ROBUST DECISION MAKING
AND ITS APPLICATIONS IN MACHINE LEARNING

Huan Xu

Department of Electrical and Computer Engineering
McGill University, Montréal

September 2009

A thesis submitted to McGill University
in partial fulfilment of the requirements of the degree of
Doctor of Philosophy

ABSTRACT

Decision making formulated as finding a strategy that maximizes a utility function depends critically on knowing the problem parameters precisely. The obtained strategy can be highly sub-optimal and/or infeasible when parameters are subject to uncertainty, a typical situation in practice. *Robust optimization*, and more generally *robust decision making*, addresses this issue by treating uncertain parameters as an arbitrary element of a pre-defined set and solving solutions based on a worst-case analysis. In this thesis we contribute to two closely related fields of robust decision making.

First, we address two limitations of robust decision making. Namely, a lack of theoretical justification and conservatism in sequential decision making. Specifically, we provide an axiomatic justification of robust optimization based on the MaxMin Expected Utility framework from decision theory. Furthermore, we propose three less conservative decision criteria for sequential decision making tasks, which include: (1) In uncertain Markov decision processes we propose an alternative formulation of the parameter uncertainty – the nested-set structured parameter uncertainty – and find the strategy that achieves maxmin expected utility to mitigate the conservatism of the standard robust Markov decision processes. (2) We investigate uncertain Markov decision processes where each strategy is evaluated comparatively by its gap to the optimum value. Two formulations, namely minimax regret and mean-variance trade-off of the regret, were proposed and their computational cost studied. (3) We propose a novel Kalman filter design based on trading-off the likely performance and the robustness under parameter uncertainty.

Second, we apply robust decision making into machine learning both theoretically and algorithmically. Specifically, on the theoretical front, we show that the concept of robustness is essential to "successful" learning. In particular, we prove that both SVM and Lasso are special cases of robust optimization, and such robustness interpretation implies consistency and sparsity naturally. We further establish a more general duality between robustness and generalizability – the former is a necessary and sufficient condition to the latter for an arbitrary learning algorithm – thus providing an answer to the fundamental question of what makes a learning algorithm work.

On the algorithmic front, we propose novel robust learning algorithms that include (1) a robust classifier with controlled conservatism by extending robust SVM to a soft notion of robustness known as comprehensive robustness; (2) a High-dimensional Robust Principal Component Analysis (HR-PCA) algorithm for reducing dimensionality in the case that outlying observation exists and the dimensionality is comparable to the number of observations.

RESUME

La prise de décision, formulée comme trouver une stratégie qui maximise une fonction de l'utilité, dépend de manière critique sur la connaissance précise des paramètres du problem. La stratégie obtenue peut être très sous-optimale et/ou infeasible quand les paramètres sont subjets à l'incertitude – une situation typique en pratique. L'*optimisation robuste*, et plus genéralement, la *prise de décision robuste*, vise cette question en traitant le paramètre incertain comme un élement arbitraire d'un ensemble prédéfini et en trouvant une solution en suivant l'analyse du pire scénario. Dans cette thèse, nous contribuons envers deux champs intimement reliés et appartenant à la prise de décision robuste.

En premier lieu, nous considérons deux limites de la prise de décision robuste: le manque de justification théorique et le conservatism dans la prise de décision séquentielle. Pour être plus spécifique, nous donnons une justifiquation axiomatique de l'optimisation robuste basée sur le cadre de l'utilité espérée MaxMin de la théorie de la prise de décision. De plus, nous proposons trois critères moins conservateurs pour la prise de décision séquentielle, incluant: (1) dans les processus incertains de décision de Markov, nous proposons un modèle alternative de l'incertitude de paramètres – l'incertitude structurée comme des ensembles emboîtées – et trouvons une stratégie qui obtient une utilité espérée maxmin pour mitiguer le conservatisme des processus incertains de décision de Markov qui sont de norme. (2) Nous considérons les processus incertains de décision de Markov où chaque stratégie est évaluée par comparaison de l'écart avec l'optimum. Deux modèles – le regret minimax et le compromis entre

l'espérance et la variance du regret – sont présentés et leurs complexités étudiées. (3) Nous proposons une nouvelle conception de filtre de Kalman basé sur le compromis entre la performance et la robustesse sujet a l'incertitude de paramètres.

En deuxième lieu, nous appliquons la prise de décision robuste à la théory et aux algorithmes de l'apprentissage par machine. En particulier, en ce qui se rapporte à la théorie, nous démontrons que le concepte de robustesse est essentiel à la réussite de l'apprentissage. Nous prouvons que la machine aux vecteurs de support et le Lasso sont des cas particuliers de l'optimisation robuste; de plus, cette interprétation implique naturellement la consistence et la creusité. Nous établisson ensuite une dualité plus génerale entre la robustesse et la possibilité de géneralisation – la robustesse est une condition nécessaire et suffisante à la possibilité de géneralisation pour un algorithme d'apprentissage arbitraire – ce qui répond à la question fondamentale du fonctionement d'un algorithme d'apprentissage.

En ce qui se rapporte aux algorithmes, nous proposons de nouveaux algorithmes d'apprentissage robustes, incluant: (1) un algorithm de classification robuste avec un conservatisme controllé obtenu par extension de la version robuste de la machine aux vecteurs de support vers une notion de robustesse appelée robustesse étendue; (2) un algorithme robuste d'analyse de la composante principale aux dimensions élevées pour reduire la dimension dans le cas où des observations éloignées existent et la dimension est comparable au nombre d'observations.

CONTRIBUTIONS OF AUTHORS

The work presented in this thesis has been carried out almost entirely by the doctoral candidate. To be more specific, the contribution of each co-author is

- The doctoral candidate conducted the research reported in Chapter 2 to Chapter 11 and wrote the corresponding manuscripts.
- Professor Shie Mannor provided advice and comments on the research reported in Chapter 2 to Chapter 11, and helped in editing the corresponding manuscripts.
- Professor Constantine Caramanis provided advice and comments on the research reported in Chapter 2, Chapter 6, Chapter 7, Chapter 9, Chapter 10 and Chapter 11, and helped in editing the corresponding manuscripts.
- Sungho Yun performed the simulation study reported in Section 10.7.

The original contributions of the thesis include:

- A theoretical justification of robust optimization based on the maxmin expected utility framework; see [**166**].
- A new decision making framework in uncertain Markov decision processes, which includes the nested-set structure of the uncertainty, optimality criterion based on maxmin expected utility and computational issues; see [**171**] and [**173**].
- Two formulations, minimax regret and mean-variance tradeoff of regret, in uncertain Markov decision processes where strategies are evaluated comparatively and related computational issues; see [**174**].

- A novel Kalman filter design that tradeoffs robustness to parameter uncertainty and likely performance; see [**170**] and [**172**].
- Equivalence of support vector machines and robust optimization, and a new proof of consistency of support vector machines using the robustness interpretation; see [**168**].
- Equivalence of Lasso and robust optimization, and new proofs of consistency and sparsity of Lasso using the robustness interpretation; see [**167**] and [**165**].
- The establishment of robustness as a sufficient and necessary condition of generalizability for arbitrary learning algorithms.
- The establishment of a theorem that sparsity and stability are contradictory; see [**176**] and [**175**].
- A novel classification algorithm in the spirit of a comprehensive robust version of support vector machines; see [**169**].
- A novel dimensionality reduction algorithm for high-dimension case with outlier observations, based on an "actor-critic" scheme; see [**164**].

LIST OF PUBLICATIONS

Journal publications:

(1) [**172**] H. Xu and S. Mannor. A Kalman filter design based on performance/robustness tradeoff. *IEEE Transactions on Automatic Control*, 54(5):1171-1175, 2009.

(2) [**168**] H. Xu, C. Caramanis, and S. Mannor. Robustness and regularization of support vector machines. *The Journal of Machine Learning Research*, 10(Jul):1485-1510, 2009.

(3) [**165**] H. Xu, C. Caramanis, and S. Mannor. Robust regression and Lasso. Submitted, 2008.

(4) [**173**] H. Xu and S. Mannor. The maxmin expected utility approach to uncertain Markov decision processes. Submitted, 2009.

(5) [**176**] H. Xu, S. Mannor, and C. Caramanis. Sparse algorithms are not stable. Submitted, 2009.

(6) [**166**] H. Xu, C. Caramanis, and S. Mannor. Robust optimization and maxmin expected utility. In preparation, 2009.

Conference proceedings:

(1) [**171**] H. Xu and S. Mannor. The robustness-performance tradeoff in Markov decision processes. In B. Schölkopf, J. C. Platt, and T. Hofmann, editors, *Advances in Neural Information Processing Systems 19*, pages 1537–1544. MIT Press, 2007.

(2) [**170**] H. Xu and S. Mannor. A Kalman filter design based on the performance/robustness tradeoff. In *Proceedings of Forty-Fifth Allerton Conference on Communication, Control, and Computing*, pages 59–63, 2007.

(3) [**164**] H. Xu, C. Caramanis, and S. Mannor. Robust dimensionality reduction for high-dimension data. In *Proceedings of Forty-Sixth Allerton Conference on Communication, Control, and Computing*, pages 1291–1298, 2008.

(4) [**175**] H. Xu, S. Mannor, and C. Caramanis. Sparse algorithms are not stable: A no-free-lunch theorem. In *Proceedings of Forty-Sixth Allerton Conference on Communication, Control, and Computing*, pages 1299 – 1303, 2008.

(5) [**167**] H. Xu, C. Caramanis, and S. Mannor. Robust regression and lasso. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 1801–1808, 2009.

(6) [**169**] H. Xu, C. Caramanis, S. Mannor and S. Yun. Risk sensitive robust support vector machines. To appear in *Forty-Eighth IEEE Conference on Decision and Control*, 2009.

(7) [**174**] H. Xu and S. Mannor. Parametric regret in uncertain markov decision processes. To appear in *Forty-Eighth IEEE Conference on Decision and Control*, 2009.

ACKNOWLEDGEMENT

First and foremost, I am deeply indebted to my advisor, Professor Shie Mannor. His support and advice have been invaluable, in terms of both personal interaction and professionalism. I have benefited from his broad range of knowledge, deep insight and thorough technical guidance in each and every step of my research during the last four years. I am particularly grateful for his emphasis on simplicity and profoundness in research, an approach that has immensely affected my development as an academic. Without his inspiration and supervision, this thesis would never have happened.

I am very grateful to Professor Constantine Caramanis of the University of Texas at Austin for helping me appreciate the beauty of robust optimization. I am fortunate to have had the chance to collaborate with him, an experience that helped produce a significant portion of this thesis.

Special thanks go to my thesis committee members, Professor Peter E. Caines, Professor David Avis and Professor Ioannis Psaromiligkos. Their support, suggestions and comments have been crucial to the steady progress of my research. I would also like to thank Professor Roussos Dimitrakopoulos, for helping me navigate the subtleties of real-word parameter uncertainties.

I would thank my friends and fellow students at CIM, Amir Danak, Yingxuan Duan, Peng Jia, Arman Kizilkale, Zhongjing Ma, Mojtaba Nourian, Zhi Qi, Vahid Raissi Dehkordi, Farzin Taringoo, etc. They have created a very pleasant atmosphere in which to conduct research and live my life. Special thanks goes to Jiayuan Yu, for being a constant stimulus to my research and for translating the abstract of this

thesis into French. I would also like to thank Sunho Yun of the University of Texas at Austin for providing the simulation results reported in Chapter 10.

I wish to thank the CIM staff, in particular, Jan Binder, Marlene Gray and Cynthia Davidson for providing a comfortable and professional research environment.

Finally, thanks to my parents for their love and support, and to my wife, Lei, for everything.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Introduction

It is better to be roughly right than precisely wrong.

—John Maynard Keynes

This thesis focuses on a problem that has attracted increasing attention in engineering, computer science, economics and operations research: how should an agent make his/her decision when the parameters that define a problem are not exactly known? Substantial research shows that neglecting this uncertainty completely and using approximated/guessed parameters instead can lead to decisions that result in dramatic performance degradation under the true parameters, or even being infeasible/unstable. These observations motivate the need for methodologies in decision making models that lead to solutions that are immune to parameter uncertainties.

Robust optimization, and more generally *robust decision making*, addresses the issue of parameter uncertainty in a computationally tractable way. This approach treats the uncertain parameters as an arbitrary element of a pre-defined set and finds solutions based on a worst-case analysis. This robust framework has experienced quickly rising popularity since the 1990's. However, in contrast to its increasingly broad application, little research regarding its theoretical justification is available in the literature. Moreover, robust models can lead to conservative solutions, especially in sequential decision making, where the effects of different uncertain parameters tend to cancel each other out.

Of particular interest is the application of such robust framework into machine learning, both theoretically and algorithmically. Indeed, many machine learning problems are inherently decision making tasks under parameter uncertainty. For example, in the binary classification problem, one observes a finite number of labeled samples, and finds a rule which with high probability will correctly predict the label given a new unlabeled sample. Such a task essentially requires computing a prediction rule with a minimal expected error, where the expectation is taken over the *unknown* generative distribution that can only be approximated using the observed samples. It is therefore natural to study the relationship between successful learning and the robustness of the learning algorithm, and further design novel learning algorithms by harnessing developments in robust decision making.

In this thesis, we investigate and contribute to two closely related aspects. First, we address the aforementioned limitations of the robust framework. In particular, we provide an axiomatic justification of robust optimization and propose new "flexible" robust decision making methodologies which can smoothly adjust the level of protection toward parameter uncertainty.

Second, we apply the robust framework to machine learning. We show that successful learning algorithms such as Support Vector Machines (SVM) and Lasso are special cases of robust optimization, and further prove that robustness is indeed the *necessary* and *sufficient* condition for a general learning algorithm to work. Finally, we propose new learning algorithms that are robust to parameter perturbation and outlying observations.

1.1. Robust Decision Making

Decision making tasks are often formulated as maximizing a certain utility function jointly determined by the strategy chosen by the decision agent and the problem parameter. That is, the decision agent attempts to solve the following problem:

$$\text{Maximize:}_{\boldsymbol{\pi}} \quad u(\boldsymbol{\pi}, \boldsymbol{\xi}). \tag{1.1}$$

Here, $\boldsymbol{\pi}$ is the strategy to take, and $\boldsymbol{\xi}$ is the problem parameter. We note that it is straightforward to include constrained problems by setting the utility to $-\infty$ for infeasible strategies.

Problem parameters may be subject to uncertainty in many real-world problems. This is due to noisy observations, estimating parameters from a finite number of samples, and over-simplification of the problem formulation. Take supply chain optimization for example. The actual demand for products, critical to evaluate the expected income of a decision, is often not precisely known when a decision has to be made, and thus has to be inferred from previous records.

It has long been known that neglecting parameter uncertainty and instead solving the decision problem (1.1) with some *roughly right* parameters often render a computed solution *precisely wrong*, i.e., highly infeasible, suboptimal or both. We quote here from the case study by Ben-Tal and Nemirovski [**13**] on linear optimization problems with parameter uncertainty:

> In real-world applications of Linear Programming, one cannot ignore the possibility that a small uncertainty in the data can make the usual optimal solution completely meaningless from a practical viewpoint.

Early efforts of addressing parameter uncertainty include *sensitivity analysis* and *stochastic programming*. Sensitivity analysis (e.g., [**24, 113, 33**]) quantifies the change in utility of the computed decision for small perturbations of problem data. However, this inherently *ex post* analysis is not particularly helpful for *computing* solutions that are robust to data changes. Stochastic programming (e.g., [**29, 123, 96**]) treats uncertain parameters as random variables with a known probabilistic description. The decision that maximizes the expected utility is thus deemed optimal. However, the assumption that the actual distributions of the uncertain parameters are available is rarely satisfied in practice. Furthermore, even if we know the distributions, finding such an optimal strategy is often computationally challenging.

A more recent approach to decision making under parameter uncertainty is *robust optimization*, in which the uncertainty is not stochastic, but rather deterministic and set-inclusive. Strategies are then ranked by their utility under the (respective) most adversarial parameter realization. The main advantages of this robust approach are two-fold. First, the set-inclusive uncertainty model is often more realistic than the assumption of knowing the distribution of the uncertainty. Second, and perhaps more importantly in practice, the resulting "robust problems" remain tractable for many decision making problems.

In the 1970s, Soyster [**141**] was among the first researchers to investigate robust optimization explicitly. He considered robust linear optimization where the column vectors of the constraint matrix were subject to set-inclusive uncertainty. Due to this column-wise uncertainty formulation, the resulting model produces overly-conservative solutions (see the comments in [**12**]).

Robust formulations of *mathematical programming* have been extensively investigated since the late 1990s following the work of Ben-tal and Nemirovski [**11, 12, 13**], El Ghaoui et.al [**64, 65**] and Bertsimas and Sim [**22, 21**]. It was shown that for a large number of optimization problems including Linear Program (LP), Quadratic Constrained Quadratic Program (QCQP) and Second Order Cone Program (SOCP), the robust formulations where the uncertainty set is either polyhedral or ellipsoidal remain polynomial time solvable.

This robust framework has been applied to sequential decision making as well. For example, in Markov Decision Processes(MDP) [**124, 16**], similarly as in the case of mathematical programming, the practical performance of a strategy in MDP can significantly differ from the model's prediction due to parameter uncertainty (cf experiments in Mannor et al. [**107**]). Most attempts to reduce such performance variation consider the robust MDP formulation (e.g., [**116, 5, 161, 91, 67**]), i.e., optimizing the worst-case performance for set-inclusive uncertainties. Such robust formulation is tractable under the assumption that parameters are state-wise independent and the uncertainty set is compact and convex. Robust Kalman filtering is another successful

application of the robust framework [**130, 63**], where the filtering problem is decomposed into a sequence of optimization problems, and each of them is then robustified using *robust optimization*.

There are two limitations to the robust approach, which we address in Chapter 2 to Chapter 5 of this thesis. First, in contrast to its increasingly broad applications, little research has been done on its theoretic justifications. One notable exception is [**17**], where the authors justified the *Robust Linear Program* by showing that a worst-case linear constraint is equivalent to a coherent risk measure constraint. However, for general robust decision making, a similar justification from decision theory seems missing from the literature.

Second, from a practitioner's perspective, the robust approach can lead to conservative solutions, partly because its set-inclusive formulation makes it hard to incorporate the distributional information of the uncertain parameters. For mathematical programming problems, such conservativeness is mitigated by constructing uncertainty sets that have adjustable probability guarantees [**23, 22**]. For the special case of linear optimization, robust-like formulations based on exploiting the risk preference of the decision maker are proposed [**17, 9**], which essentially provide flexible protection toward parameter uncertainty. However, for general robust decision making, in particular sequential decision making problems, the conservativeness of robust formulation has not been investigated.

1.2. Robustness in Machine Learning

In the last decade, a body of literature has developed to apply robust optimization into machine learning tasks such as classification (e.g., [**101, 26, 27, 137, 150, 79**]) and regression [**64**]. These works consider cases where training samples are subject to exogenous noise, and propose robustified learning algorithms that are essentially modified versions of standard learning algorithms to control the adversarial effect of such noise.

Apart from these algorithmic contributions, little has been done to relate robustness as a reason why learning algorithms work, or, more precisely, a reason for *generalizability*. Generalizability means that the expected performance should agree with the empirical error as the number of training samples increases, and is deemed to be the key requirement of a supervised learning algorithm: an algorithm that learns a mapping given a set of observed input-output pairs.

There are two classical approaches for examining generalizability in literature. The first one is based on the uniform convergence of empirical quantities to their mean (e.g., [**157, 155, 158**]). This approach provides ways to bound the gap between the risk on a test set and the empirical risk on a training set in terms of the complexity of the space of learned mappings. Examples of complexity measures are the Vapnik-Chervonenkis (VC) dimension (e.g., [**155, 70**]), the fat-shattering dimension (e.g., [**1, 6**]), and the Rademacher complexity ([**8, 7**]).

Another well-known approach is based on *stability*. An algorithm is stable if its output remains "similar" for different sets of training samples that are identical up to the removal or change of a single sample. This is in contrast to the complexity-based approach that focuses on the space that an algorithm searches, as stability analysis concentrates on *how* the algorithm searches the space. The first results that related stability to generalizability track back to [**55**] and [**56**]. Later, McDiarmid's [**110**] concentration inequalities facilitated new bounds on generalization error (e.g., [**52, 32, 100, 122, 112**]).

Both aforementioned approaches provide sufficient but not necessary conditions for generalizability. Indeed, to the best of our knowledge, a necessary and sufficient condition of generalizability for general learning algorithms has not been suggested in the literature. A notable exception is the Empirical Risk Minimization (ERM) algorithm, where it is known that both having a finite VC-dimension [**158**] and being CVEEE$_{loo}$ stable [**112**] are necessary and sufficient conditions for an ERM algorithm to generalize. However, the class of ERM algorithms is restrictive, and does not

include many algorithms that are successful in practice such as k-NN, Support Vector Machines (SVM) [**133**] and Boosting [**132, 75**].

In this thesis, we show that robustness is a critical property leading to generalizability. We first examine two widely used learning algorithms: Support Vector Machines (SVM) [**159, 157, 31**] for classification and Lasso [**146, 61**] for regression, and prove that they are indeed a robustified version of empirical risk minimization. Moreover, we use this robust optimization interpretation to prove the generalizability of SVM and Lasso. Thess are indeed special cases that display the relationship between robustness and generalizability: we prove that robustness is a *necessary and sufficient* condition of generalizability, and therefore provide an answer to the following fundamental question: "what is the reason for learning algorithms to work?"

1.3. Structure of the Thesis

This thesis is organized as follows:

Chapter 2. Robust Optimization and MaxMin Expected Utility. An axiomatic justification of robust optimization based on the MaxMin Expected Utility framework is presented in this chapter, in order to address the lack of theoretical justification of robust optimization as discussed. Furthermore, a special case in which multiple parameters all belong to the same space is investigated. The result implies that one can use robust optimization for decision making problems with distributional requirements, such as stochastic programming and machine learning. Part of the material in this chapter appears in [**166**].

Chapter 3. The MaxMin Expected Utility Approach to Uncertain Markov Decision Processes. A novel approach to handling uncertain Markov decision processes is proposed in this chapter to address the conservatism of the robust approach. In particular, the parameter uncertainty is represented by nested sets: the parameters are likely to belong to the inner set, and are guaranteed to belong to the outer set. Such formulation arises naturally from the maxmin expected utility

framework and can model the case where the decision maker knows *a priori* both the likely values and the possible deviation of the parameters. A polynomial time algorithm that computes the optimal strategy is presented. In a special case where only the reward parameters are subject to uncertainty, the optimal strategy can be interpreted as a tradeoff between the likely performance and the downside deviation. If the uncertainty sets are polyhedral, an algorithm that computes the *whole* set of optimal tradeoff strategies in a single run is proposed. This allows the decision maker to obtain the most desirable tradeoff without committing to a single tradeoff a-priori. Part of the material in this chapter appears in [**171**] and [**173**].

Chapter 4. Parametric Regret in Uncertain Markov Decision Processes. In standard Markov decision processes, each strategy is evaluated by its accumulated reward-to-go. However, there are situations where the decision maker is concerned about how the performance of a strategy compares with other strategies. Robust decision making of uncertain Markov decision processes in such a comparative setup is investigated in this chapter. Each strategy is evaluated by its *parametric regret*: the gap between the performance of the best strategy and the performance of the strategy that is chosen before the parameter realization is revealed. Two related problems – minimax regret and mean-variance tradeoff of the regret – are discussed: In the minimax regret formulation, the true parameters are regarded as deterministic but unknown, and the optimal strategy is the one that minimizes the worst-case regret under the most adversarial possible realization. The problem of computing the minimax regret strategy is shown to be NP-hard in general. Furthermore, algorithms that efficiently solve minimax regret strategy under favorable conditions are proposed. The mean-variance tradeoff formulation requires a probabilistic model of the uncertain parameters and looks for a strategy that minimizes a convex combination of the mean and the variance of the regret. An algorithm is proposed that computes such a strategy numerically in an efficient way. Part of the material in this chapter appears in [**174**].

Chapter 5. A Kalman Filter Design Based on the Performance/Robustness Tradeoff. Robust state estimation is investigated in this chapter. Based on the likely performance/worst-case performance tradeoff concept raised in Chapter 3, a new Kalman filter that solves a tradeoff problem iteratively is proposed. The proposed filter can be computed efficiently online and is steady-state stable. Simulation results show that it is less conservative than the robust filter and performs satisfactorily under a wide range of scenarios. Part of the material in this chapter appears in [**172**] and [**170**].

Chapter 6. Robustness and Regularization of Support Vector Machines. From this chapter on, we will concentrate on the application of robust decision making to machine learning. In this chapter, one of the most widely used classification algorithms, the Support Vector Machine (SVM in short), is investigated. In particular, it is shown that the regularized SVM is precisely equivalent to a new robust optimization formulation. Such an equivalence relationship provides a robust optimization interpretation for the success of regularized SVMs. A new proof of consistency of (kernelized) SVMs based on this robustness interpretation is given, thus establishing robustness as the *reason* regularized SVMs generalize well. Part of the material of this chapter appears in [**168**].

Chapter 7. Robust Regression and Lasso. In this chapter, the robustness property of Lasso (i.e., ℓ^1 regularized least squares) is investigated. It is shown that Lasso can be recast as a robust optimization problem. The implications of this are two-fold: First, robustness provides a connection of the regularizer to a physical property, namely, protection from noise, which thus allows a principled selection of the regularizer as well as generalizations of Lasso that also yield convex optimization problems. Second, robustness can be used as an avenue for exploring different properties of the solution. In particular, the robustness of Lasso explains why its solution is sparse. Furthermore, a proof that Lasso is consistent is given using robustness

directly. Finally, a theorem saying that Lasso is not stable, is presented. Part of the material in this chapter appears in [**165**] and [**167**].

Chapter 8. All Learning is Robust: On the Equivalence of Robustness and Generalizability. In Chapter 6 and Chapter 7, it is shown that some widely implemented learning algorithms have nice robustness properties that imply consistency. This chapter generalizes such results. Indeed, it is shown that robustness is a necessary and sufficient condition for an arbitrary learning algorithm to perform "well", more precisely, to generalize. This is the first "if-and-only-if" condition for the generalizability of learning algorithms other than empirical risk minimization. Conditions that ensure robustness and hence generalizability for samples that are independent and identically distributed and for samples that come from a Markov chain are also presented. This leads to new theorems of generalizability as well as novel proofs of known results.

Chapter 9. Sparse Algorithms are not Stable: A No-free-lunch Theorem. This chapter generalizes the theorem that Lasso is not stable which is presented in Chapter 7 to a more general context. In particular, it is shown that two widely used notions in machine learning, namely: *sparsity* and *algorithmic stability*, both deemed desirable in designing algorithms, contradict each other. That is, under mild technical assumptions, a sparse algorithm can not be stable and vice versa. Thus, one has to tradeoff sparsity and stability in designing a learning algorithm. Examples of stable (hence non-sparse) algorithms and sparse (hence non-stable) algorithms are presented to illustrate the implication of this theorem. Part of the material in this chapter appears in [**175**] and [**176**].

Chapter 10. Comprehensive Robust Support Vector Machines and Convex Risk Measures. In this chapter, a novel support vector machines classification algorithm based on robust optimization is proposed, one that builds in non-conservative protection to noise and controls overfitting. The formulation is based on

a softer version of robust optimization called comprehensive robustness. It is shown that this formulation is equivalent to regularization by any arbitrary convex regularizer. The connection of comprehensive robustness to convex risk-measures is explored, which can be used to design risk-measure constrained classifiers with robustness to the input distribution. The proposed formulation leads to convex optimization problems that can be easily solved and achieves promising empirical results. Part of the material in this chapter appears in [**169**].

Chapter 11. Robust Dimensionality Reduction for High-Dimension Data. This chapter investigates the dimensionality-reduction problem for *contaminated* data in the high dimensional regime, where the the number of *observations* is of the same order of magnitude as the number of *variables* of each observation, and the data set contains some (arbitrarily) corrupted observations. A High-dimensional Robust Principal Component Analysis (HR-PCA) algorithm is proposed. The HR-PCA algorithm takes an "actor-critic" form: we apply standard PCA in order to find a set of candidate directions. These directions are then subjected to a hypothesis test which determines whether the variance is due to corrupted data, or indeed the "authentic" points. In the latter case, the algorithm has found a true principal component. In the former case, a randomized point removal scheme is used that guarantees quick convergence. The HR-PCA algorithm is tractable, robust to contaminated points, and easily kernelizable. The resulting solution has a bounded deviation from the optimal one, and unlike PCA, achieves optimality in the limit case where the proportion of corrupted points goes to zero. Part of the material in this chapter appears in [**164**].

Chapter 12. Conclusion. This chapter contains some concluding remarks and discusses open issues and questions raised by this thesis.

CHAPTER 2

Robust Optimization and MaxMin Expected Utility

This Chapter address the first limitation of robust optimization as discussed in Chapter 1: a lack of theoretical justifications. We show that Robust Optimization is a special case of MaxMin Expected Utility framework, and hence giving an *axiomatic* justification of robust optimization from a decision theory perspective. A special case where multiple parameters all belong to the same space is investigated. Such a result implies that Robust Optimization can be used to handle decision making problems where some probabilistic information of unknown parameters is available, particularly problems which involve using samples to approximate the underlying generative distribution, such as stochastic programming and machine learning. Part of the material in this chapter appears in [**166**].

2.1. Introduction

Robust Optimization (RO), traced back as early as [**141**], is widely used in operations research, computer science, engineering, and many other fields (e.g., [**12, 13, 22, 64, 63, 137, 101**], see [**18**] for a detailed survey). In contrast to its increasingly broad applications, little research has been done on its theoretical justifications. One notable exception is [**17**], where the authors justified Robust Linear Program by

showing that a worst-case linear constraint is equivalent to a coherent risk measure constraint.

In this chapter, we provide a MaxMin Expected Utility (MMEU) interpretation for Robust Optimization. We show that RO (not necessarily Linear Program) is a special case of MMEU decision, i.e., the worst performance under disturbance of parameters equals to the maximum error w.r.t. a class of probability measures (hereafter referred as the *corresponding class*). We thus provide an axiomatic justification to RO from a decision making perspective. Furthermore, this relationship implies that RO can be a useful tool to deal with decision making problems with distributional requirements, which include most problems in machine learning and stochastic programming. This also helps one to determine the uncertainty set by exploring the distributional requirement of the problem.

We consider a general robust optimization problem, i.e., maximizing an arbitrary utility. Therefore, we drop constraints by setting the utility of infeasible solutions as $-\infty$. We further consider a special case where parameters $\{\mathbf{x}_i\}_{i=1}^n$ belong to the same space $\mathbb{R}^m$, and we are looking for the corresponding set in $\mathbb{R}^m$ rather than in $\mathbb{R}^{m\times n}$. This case is of interest when one is trying to approximate a continuous probability distribution with a finite number of samples, either because the distribution is not explicitly known, or the computation involved is untractable.

We use $\mathcal{P}$ and Υ in this section to denote the set of probability measures and the set of σ-finite measures, respectively, both defined on Borel algebra of $\mathbb{R}^m$. Hence $\mathcal{P} = \{\mu \in \Upsilon | \mu(\mathbb{R}^m) = 1\}$.

2.2. MaxMin Expected Utility

In [**77, 98**], the authors proposed an axiomatic framework for decision-making under uncertainty, which is often referred as Maxmini Expected Utility (MMEU). We here briefly recall their results.

Consider a general decision making problem, where each action a is defined by its outcomes under different parameter realizations. For example, if there are n possible

parameter realizations, then a can be identified by a vector $(a_1, \cdots, a_n) \in \mathbb{R}^n$. Let αe denote an action that achieves a constant outcome $\alpha \in \mathbb{R}$ under all parameter realizations. The authors prove the following existence theorem (see Theorem 1 of [**98**]).

THEOREM 2.1. *If a preference relationship among actions has a functional form $V(a)$, and satisfying the following axioms:*

(1) Quasi-Concavity: V is Quasi-Concave.

(2) Certainty Independence: for any action b, and $0 < \lambda < 1$,

$$V\big(\lambda b + (1-\lambda)\alpha e\big) = \lambda V(b) + (1-\lambda)V(\alpha e).$$

(3) Weak Dominance: V is increasing in all arguments.

Then, there exists a closed convex set $\mathcal{C}$ of probability distributions on parameter realizations, such that $V(a) = \min_{p\in\mathcal{C}} \mathbb{E}_p a$.[1]

If the outcome of an action has a functional form, i.e., some $u(\mathbf{v}, \mathbf{x})$ denotes the outcome of a decision variable $\mathbf{v}$ under a parameter realization $\mathbf{x}$, then we can rewrite Theorem 2.1 as

$$V(a) = \hat{V}(\mathbf{v}) \triangleq \min_{\mu\in\mathcal{C}} \int u(\mathbf{v}, \mathbf{x}) d\mu(\mathbf{x}),$$

i.e., each decision is evaluated w.r.t the minimum (among a set of distributions) expected utility, and the optimal decision is given by $\mathbf{v}^* = \arg\max \hat{V}(\mathbf{v})$. This can also be interpreted as a robust solution w.r.t distributions. It is straightforward to extend to cost-minimization problem.

2.3. Robustness and MMEU

In this section we investigate the relationship between robustness and MMEU. Notice that we are interested in finding the equivalence relationship given any fixed candidate decision. Hence we drop the decision variable in this section. Theorem 2.2

[1]Here $\mathbb{E}_p a$ is understood as $\sum_i p_i a_i$, where p_i is the probability that i^{th} realization happens assuming that the parameter follows a probabilistic law p, and a_i is the outcome of a under the i^{th} realization.

investigates the equivalence relationship of a general robust optimization problem. Theorem 2.3 considers the special case in which multiple parameters are assumed to belong to a same space.

THEOREM 2.2. *Given a function $f : \mathbb{R}^m \to \mathbb{R}$, and a Borel set $\mathcal{Z} \subseteq \mathbb{R}^m$, the following holds:*

$$\inf_{\mathbf{x}' \in \mathcal{Z}} f(\mathbf{x}') = \inf_{\mu \in \mathcal{P} | \mu(\mathcal{Z})=1} \int_{\mathbb{R}^m} f(\mathbf{x}) d\mu(\mathbf{x}).$$

PROOF. Let $\hat{\mathbf{x}}$ be a $\epsilon-$optimal solution to the left hand side, consider the probability measure μ' that puts mass 1 on $\hat{\mathbf{x}}$, and satisfies $\mu'(\mathcal{Z}) = 1$. Hence, we have

$$\inf_{\mathbf{x}' \in \mathcal{Z}} f(\mathbf{x}') + \epsilon \geq \inf_{\mu \in \mathcal{P} | \mu(\mathcal{Z})=1} \int_{\mathbb{R}^m} f(\mathbf{x}) d\mu(\mathbf{x}),$$

since ϵ can be arbitrarily small, this leads to

$$\inf_{\mathbf{x}' \in \mathcal{Z}} f(\mathbf{x}') \geq \inf_{\mu \in \mathcal{P} | \mu(\mathcal{Z})=1} \int_{\mathbb{R}^m} f(\mathbf{x}) d\mu(\mathbf{x}). \tag{2.1}$$

Next construct function $\hat{f} : \mathbb{R}^m \to \mathbb{R}$ as

$$\hat{f}(\mathbf{x}) \triangleq \begin{cases} f(\hat{\mathbf{x}}) & \mathbf{x} \in \mathcal{Z}; \\ f(\mathbf{x}) & \text{otherwise.} \end{cases}$$

By definition of $\hat{\mathbf{x}}$ we have $f(\mathbf{x}) \geq \hat{f}(\mathbf{x}) - \epsilon$ for all $\mathbf{x} \in \mathbb{R}^m$. Hence, for any probability measure μ such that $\mu(\mathcal{Z}) = 1$, the following holds

$$\int_{\mathbb{R}^m} f(\mathbf{x}) d\mu(x) \geq \int_{\mathbb{R}^m} \hat{f}(\mathbf{x}) d\mu(x) - \epsilon = f(\hat{\mathbf{x}}) - \epsilon \geq \inf_{\mathbf{x}' \in \mathcal{Z}} f(\mathbf{x}') - \epsilon.$$

This leads to

$$\inf_{\mu \in \mathcal{P} | \mu(\mathcal{Z})=1} \int_{\mathbb{R}^m} f(\mathbf{x}) d\mu(x) \geq \inf_{\mathbf{x}' \in \mathcal{Z}} f(\mathbf{x}') - \epsilon.$$

Notice ϵ can be arbitrarily small, we have

$$\inf_{\mu \in \mathcal{P} | \mu(\mathcal{Z})=1} \int_{\mathbb{R}^m} f(\mathbf{x}) d\mu(x) \geq \inf_{\mathbf{x}' \in \mathcal{Z}} f(\mathbf{x}') \tag{2.2}$$

Combining (2.1) and (2.2), we prove the theorem. □

THEOREM 2.3. *Given a function $f : \mathbb{R}^m \to \mathbb{R}$ and Borel sets $\mathcal{Z}_1, \cdots, \mathcal{Z}_n \subseteq \mathbb{R}^m$, denote*

$$\mathcal{P}_n \triangleq \{\mu \in \mathcal{P} | \forall S \subseteq \{1, \cdots, n\} : \mu(\bigcup_{i \in S} \mathcal{Z}_i) \geq |S|/n\}.$$

The following holds

$$\frac{1}{n} \sum_{i=1}^{n} \inf_{\mathbf{x}_i \in \mathcal{Z}_i} f(\mathbf{x}_i) = \inf_{\mu \in \mathcal{P}_n} \int_{\mathbb{R}^m} f(\mathbf{x}) d\mu(\mathbf{x}).$$

Notice the sets $\mathcal{Z}_1, \cdots, \mathcal{Z}_n$ can overlap with each other or even be identical.

PROOF. Let $\hat{\mathbf{x}}_i$ be an $\epsilon-$optimal solution to $\inf_{\mathbf{x}_i \in \mathcal{Z}_i} f(\mathbf{x}_i)$. Observe that the empirical distribution for $(\hat{\mathbf{x}}_1, \cdots, \hat{\mathbf{x}}_n)$ belongs to $\mathcal{P}_n$. Since ϵ can be arbitrarily close to zero, we have

$$\frac{1}{n} \sum_{i=1}^{n} \inf_{\mathbf{x}_i \in \mathcal{Z}_i} f(\mathbf{x}_i) \geq \inf_{\mu \in \mathcal{P}_n} \int_{\mathbb{R}^m} f(\mathbf{x}) d\mu(\mathbf{x}). \tag{2.3}$$

Without loss of generality, assume

$$f(\hat{\mathbf{x}}_1) \geq f(\hat{\mathbf{x}}_2) \geq \cdots \geq f(\hat{\mathbf{x}}_n). \tag{2.4}$$

Now construct the following function

$$\hat{f}(\mathbf{x}) \triangleq \begin{cases} \max_{i | \mathbf{x} \in \mathcal{Z}_i} f(\hat{\mathbf{x}}_i) & \mathbf{x} \in \bigcup_{j=1}^{n} \mathcal{Z}_j; \\ f(\mathbf{x}) & \text{otherwise.} \end{cases}$$

Observe that $f(\mathbf{x}) \geq \hat{f}(\mathbf{x}) - \epsilon$ for all $\mathbf{x}$.

Furthermore, given $\mu \in \mathcal{P}_n$, we have

$$\begin{aligned} & \int_{\mathbb{R}^m} f(\mathbf{x}) d\mu(\mathbf{x}) + \epsilon \\ \geq & \int_{\mathbb{R}^m} \hat{f}(\mathbf{x}) d\mu(\mathbf{x}) \\ = & \sum_{k=1}^{n} f(\hat{\mathbf{x}}_k) \Big[\mu(\bigcup_{i=1}^{k} \mathcal{Z}_i) - \mu(\bigcup_{i=1}^{k-1} \mathcal{Z}_i) \Big] \end{aligned}$$

Denote $\alpha_k \triangleq \left[\mu(\bigcup_{i=1}^{k} \mathcal{Z}_i) - \mu(\bigcup_{i=1}^{k-1} \mathcal{Z}_i)\right]$, we have

$$\sum_{k=1}^{n} \alpha_k = 1, \quad \sum_{k=1}^{t} \alpha_k \geq t/n.$$

Hence by Equation (2.4) we have

$$\sum_{k=1}^{n} \alpha_k f(\hat{\mathbf{x}}_k) \geq \frac{1}{n} \sum_{k=1}^{n} f(\hat{\mathbf{x}}_k).$$

Thus we have for any $\mu \in \mathcal{P}_n$,

$$\int_{\mathbb{R}^m} f(\mathbf{x}) d\mu(\mathbf{x}) + \epsilon \geq \frac{1}{n} \sum_{k=1}^{n} f(\hat{\mathbf{x}}_k).$$

Therefore,

$$\inf_{\mu \in \mathcal{P}_n} \int_{\mathbb{R}^m} f(\mathbf{x}) d\mu(\mathbf{x}) + \epsilon \geq \inf_{\mathbf{x}_i \in \mathcal{Z}_i} \frac{1}{n} \sum_{k=1}^{n} f(\mathbf{x}_k).$$

Notice ϵ can be arbitrarily close to 0, we proved the proposition by combining with (2.3). □

Note that in both Theorem 2.2 and 2.3, the corresponding classes of probability measures only depend on uncertainty sets. Therefore, the equivalent relationships are indeed uniform for all utility functions, in particular the set of utility functions $f_{\mathbf{v}}(\cdot)$ indexed with the decision variable $\mathbf{v}$. Thus, the optimal decision to the left-hand-side and the right-hand-side are the same.

2.4. Discussions

Theorem 2.2 and 2.3 have two-fold significance. On one hand, they provide an axiomatic justification of the widely-used RO method from a decision-theory perspective. On the other-hand, they imply a RO-based approach to handle decision making problems based on distributional requirements. For example, many decision problems can be written as

$$\text{maximize:}_{\mathbf{v}} \quad \mathbb{E}_{\mathbf{x} \sim \mathbb{P}}\{f_{\mathbf{v}}(\mathbf{x})\}, \tag{2.5}$$

where $\mathbb{P}$ is either unknown (e.g., most machine learning problems where only a set of samples generated according to $\mathbb{P}$ are given) or too complicated to evaluate (e.g., most stochastic programming problems with a continuum of scenarios). A sampling technique is often used instead, i.e., the true distribution $\mathbb{P}$ is replaced by μ_n which is an empirical distribution of n i.i.d. samples $\mathbf{x}_1, \cdots, \mathbf{x}_n$ generated according to $\mathbb{P}$, and a decision

$$\mathbf{v}_n^* \triangleq \arg\max_{\mathbf{v}} \mathbb{E}_{\mathbf{x}\sim\mu_n}\{f_{\mathbf{v}}(\mathbf{x})\} = \arg\max_{\mathbf{v}} \frac{1}{n}\sum_{i=1}^{n} f_{\mathbf{v}}(\mathbf{x}_i),$$

is taken as an approximation of the solution of Problem (2.5). However, it is widely known that such a sampling technique often yields overly optimistic solution, i.e., the empirical utility for $\mathbf{v}_n^*$ is a biased estimation of its expected utility. Even worse, it is often the case that as $n \uparrow \infty$, the sequence $\{\mathbf{v}_n^*\}$ does not converge to the optimal decision. This is often termed as "over-fitting" in machine learning literature ([**158**]), and has attracted extensive research. Briefly speaking, this is because the convergence of $\mathbb{E}_{\mu_n} f_{\mathbf{v}}(\cdot)$ to $\mathbb{E}_{\mathbb{P}} f_{\mathbf{v}}(\cdot)$ is not uniform for all $\mathbf{v}$. We propose to solve this problem by constructing uncertainty sets such that the right-hand-side of the equality in Theorem 2.3 "approximately" contains the true probability distribution $\mathbb{P}$. To be more rigorous, this is to say the right-hand-side contains a sequence of distributions which converges to $\mathbb{P}$ uniformly w.r.t. $\mathbf{v}$. One notable example of such sequence is any kernel density estimator. If in addition, the size of the uncertainty set shrinks to zero, the sequence of the min-max decisions converges to the optimal solution. This property is in fact exploited implicitly by many widely used learning algorithms, as illustrated by the following two examples. See Chapter 6 and Chapter 7 for details.

The following example is in the classical machine learning setup where a decision maker observes a set of training samples $\{\mathbf{x}_i\}_{i=1}^m$ (each sample is assumed in $\mathbb{R}^k$) and their labels $\{y_i\}_{i=1}^m$ (each label is in $\{-1, 1\}$).

EXAMPLE 2.1. *If the training samples* $\{\mathbf{x}_i, y_i\}_{i=1}^m$ *are non-separable, then the regularized SVM problem*

$$\min_{\mathbf{w},b} : \quad c\|\mathbf{w}\|^* + \sum_{i=1}^{m} \max\big[1 - y_i\big(\langle \mathbf{w},\, \mathbf{x}_i\rangle + b\big), 0\big],$$

is equivalent to the robust optimization problems

$$\min_{\mathbf{w},b} : \quad \max_{(\boldsymbol{\delta}_1,\cdots,\boldsymbol{\delta}_m)\in\mathcal{T}} \sum_{i=1}^{m} \max\big[1 - y_i\big(\langle \mathbf{w},\, \mathbf{x}_i - \boldsymbol{\delta}_i\rangle + b\big), 0\big].$$

where the uncertainty set is given by

$$\mathcal{T} \triangleq \Big\{(\boldsymbol{\delta}_1, \cdots \boldsymbol{\delta}_m) | \sum_{i=1}^{m} \|\boldsymbol{\delta}_i\| \leq c; \Big\}.$$

The next example considers a regression setup. In this setup we are given m vector in $\mathbb{R}^k$ denoted by $\{\mathbf{a}_i\}_{i=1}^m$ and m associated real values $\{b_i\}_{i=1}^m$. We are looking for a k dimensional linear regressor $\mathbf{v}$ that satisfies $\mathbf{b} \approx A\mathbf{v}$, where A is a matrix whose rows are the m vectors. There are many way to solve this regression problem and we consider a specific popular framework known as Lasso.

EXAMPLE 2.2. *The* l_1 *regularized regression problem (aka Lasso)*

$$\min_{\mathbf{v}} : \quad \|\mathbf{b} - A\mathbf{v}\|_2 + c\|\mathbf{v}\|_1,$$

is equivalent to a robust regression

$$\min_{\mathbf{v}} : \quad \max_{\Delta A\in\mathcal{U}} \|\mathbf{b} - (A + \Delta A)\mathbf{v}\|_2,$$

with the uncertainty set

$$\mathcal{U} \triangleq \Big\{(\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m)\Big| \|\boldsymbol{\delta}_i\|_2 \leq c, \;\; i = 1, \cdots, m\Big\}.$$

2.5. Chapter summary

In this chapter, we showed that Robust Optimization has a MaxMin Expected Utility interpretation, and explicitly found the set of probability measures that corresponds to the disturbance. A special case where multiple parameters belong to the same space is also considered, which is of interest in handling decision problems where some probabilistic information of unknown parameters is available.

The main thrust of this research is to embed *general* Robust Optimization into a well-established axiomatic decision making framework. This not only provides a more solid justification and motivation of RO, but also suggests a new approach on choosing the uncertainty set by exploring the distributional requirement. One such example is to use RO in approximating the generative distribution by finite samples, a process that has been implicitly used by many standard learning algorithms.

CHAPTER 3

The MaxMin Expected Utility Approach to Uncertain Markov Decision Processes

As discussed in Chapter 1, the so-called robust approach of Markov decision processes can be overly conservative. In this chapter we propose an alternative approach by modeling the uncertainty in a more flexible way. In particular, we consider Markov decision processes where there is uncertainty in the values of the parameters. This uncertainty is represented by nested sets: the parameters are likely to belong to the inner set, and are guaranteed to belong to the outer set. Our formulation models the case where the decision maker knows a-priori both the likely values and the possible deviation of the parameters and arises naturally from the maxmin expected utility framework. We devise a polynomial time algorithm for computing a strategy that maximizes the expected utility under the most adversarial distribution. When only the reward parameters are subject to uncertainty, such strategies have an interpretation as a tradeoff between the likely performance and the performance under worst case parameters. If the uncertainty sets are polyhedral, we propose an algorithm that computes the *whole* set of optimal tradeoff strategies in a single run without committing to a single tradeoff *a priori*. Part of the material in this chapter appears in [**171**] and [**173**].

3.1. Introduction

Sequential decision making in stochastic dynamic environments is often modeled using Markov decision processes (e.g., Puterman [**124**], Bertsekas and Tsitsiklis [**16**]). A strategy that achieves maximal expected accumulated reward is considered to be the optimal solution. However, in many applications, the practical performance of such a strategy can significantly differ from the model's prediction due to *parameter uncertainty* – the deviation of the modeling parameters from the true ones (cf. experiments in Mannor et al. [**107**]). Most attempts to reduce such performance variation consider the robust MDP formulation (e.g., Nilim and El Ghaoui [**116**], Bagnell et al. [**5**], White and El Deib[**161**], Iyengar [**91**], Epstein and Schneider[**67**]). In this context, it is assumed that the parameters can be any member of a known set (termed the *uncertainty set*), and solutions are ranked based on their performance under the (respective) worst parameter realizations. The optimal solution to a robust MDP is obtained in polynomial time under the assumption that parameters are state-wise independent and the uncertainty set is compact and convex.

A major disadvantage of the robust approach is that it often generates overly conservative strategies tailored to parameters with large perturbations (cf Delage and Mannor [**50**]). This is due to the fact that all parameter realizations within the uncertainty set are treated in the same manner, which leads to an unfavorable bias to adverse rare disturbances. Indeed, by replacing a convex uncertainty set with its boundary we obtain the same solution. Therefore, despite having a best downside protection, the solution of the robust MDP is often inferior under less extreme parameter realizations. A standard remedy to such conservatism is to shrink the uncertainty set, i.e., to use a smaller set of parameters. However, since certain parameter realizations are excluded completely, there is no guaranteed downside protection.

In this chapter, we propose a new formulation for MDPs under parameter uncertainty that mitigates the conservatism without losing the protection to downside deviations. In particular, we represent the parameter uncertainty with a nested-set

structure: the inner set (termed *concentration set*) stands for the "likely"[1] realizations of the parameters and the outer one (*deviation set*) is the set of all possible realizations. Such prior information of the unknown parameters is often available in practice, and the inner set and the outer set can differ significantly so that neglecting either one is not desirable.

The proposed formulation is based on the MaxMin Expected Utility (MMEU) framework that is popular in decision theory (Gilboa and Schemeidler [**77**]), which states that for a general decision problem under parameter uncertainty, a decision maker will maximize his/her expected reward under the worst parameter distribution if a set of axioms are satisfied. (See Section 2.2 for a detailed discussion.) Therefore, we treat the unknown parameters as random variables and consider the set of distributions satisfying: (1) the parameters are state-wise independent; (2) each distribution is supported by the deviation set; (3) each parameter belongs to the concentration set with a probability at least λ. Strategies are then ranked based on their expected performance under the (respective) most adversarial distribution. Observe that the robust MDP formulation is a special case of the proposed formulation by setting λ as zero, which stands for the case that the knowledge of how the distribution is concentrated is lacking.

The nested set formulation is motivated by setups where the parameters are considered to be random variables – i.e., a Bayesian approach (e.g., Strens [**143**], Dearden et al. [**49**]) is taken – and their distribution is estimated from samples. Such estimation is often imprecise especially when only a small number of samples are available. Instead, estimating uncertainty sets with high confidence can be made more accurate, which provides a lower-bound on the performance under the true distribution. A generalization to nested structures with more than two uncertainty sets is straightforward. Thus, the nested set formulation provides a framework that can model the a priori information in a flexible way. Even when the distribution of the parameters is known precisely, the nested set formulation can still be used

[1]We use the word *likely* in a loose sense here: we do not assume that we are equipped with a precise probabilistic model for the parameter uncertainty; see the discussion below.

to approximate a Bayesian setup. Such approximation is particularly useful when the decision maker is risk-sensitive, because maximizing a risk-sensitive measure in a Bayesian setup can often be computational difficult. For example, a percentile objective is known to be NP-hard (Delage and Mannor [**50**]).

We further discuss in detail an important special case where only the expected reward parameters are uncertain and the transition probabilities are precisely known. In this case, the proposed MMEU formulation can be interpreted in an intuitively appealing way: It finds Pareto efficient solutions of the "likely" performance and the performance under the worst deviation. This criterion resembles the following decision rule: "I am willing to compromise certain amount of performance, and I want to gain maximum protection toward downside deviation due to parameter perturbations". A widely known example of this decision rule is insurance policies, where one pays a small amount of premium to cover a potentially large loss if some rare event happens.

It is of interest to find the whole tradeoff relationship, because a decision maker often wants to compare different tradeoffs and choose the best one. (Consider an insurance buyer who selects an insurance policy from multiple proposals.) In general, we can only approximate this by solving finitely many tradeoffs. If the uncertainty sets are polyhedral, we propose an algorithm based on Parametric Linear Programming (PLP) that computes the *whole* set of Pareto efficient solutions in a single run. This is beneficial in practice since the decision maker's preference among different tradeoffs can be very complicated and not straightforward to incorporate as a single tradeoff parameter. Instead of arbitrarily claiming that a certain solution is a good tradeoff, our algorithm computes the whole tradeoff relationship so that the decision maker can choose the most desirable solution according to her preference. By doing this we leave the subjective decision of determining the exact tradeoff to the decision maker and hence avoid tuning of tradeoff parameters.

This chapter is organized as follows. In Section 3.2 we provide some background. We then formulate and solve the MMEU-based robust MDP for the general case in Section 3.3. The special case where the transition probabilities are known is discussed

in Section 3.4, in which we link the proposed strategy with the criterion of achieving Pareto efficient tradeoffs as well as provide an algorithm that finds the whole set of solutions. In Section 3.5, we present a computational example. Some concluding remarks are offered in Section 3.6. Finally, in Section 3.7 we provide the proof to Theorem 3.1.

NOTATION. We use capital letters to denote matrices, and bold face letters to denote column vectors. Row vectors are represented as the transpose of column vectors. We use $\mathbf{1}$ to denote the vectors of appropriate length with all elements 1, and use $\mathbf{e}_i(m)$ to denote the i^{th} elementary vector of length m. The indicator function is denoted by $\mathbf{I}(\cdot)$, i.e., the output of the function is 1 if the event inside the bracket is true, and zero otherwise.

3.2. Preliminaries

In this section, we present some background knowledge including uncertainty in Markov decision processes and parametric linear programming.

3.2.1. Uncertain Markov decision processes. A (finite) Markov Decision Process (MDP) is defined as a 6-tuple $< T, \gamma, S, A_s, \mathbf{p}, \mathbf{r} >$ where: T is the possibly infinite decision horizon; $\gamma \in (0, 1]$ is the discount factor; S is the state set; A_s is the action set of state s; both S and A_s are finite sets; $\mathbf{p}$ is the transition probability; and $\mathbf{r}$ is the expected reward. That is, for $s \in S$ and $a \in A_s$, $r(s, a)$ is the expected reward and $p(s'|s, a)$ is the probability to reach state s'. Following the notation of Puterman [**124**], we denote the set of all history-dependent randomized strategies by Π^{HR}, and the set of all Markovian randomized strategies by Π^{MR}. We use subscript s to denote the value associated with state s, e.g., $\mathbf{r}_s$ denotes the vector form of rewards associated with state s, and π_s is the (randomized) action chosen at state s for strategy π. The elements in vector $\mathbf{p}_s$ are listed in the following way: the transition probabilities of the same action are arranged in the same block, and inside each block they are listed according to the order of the next state. We use $\underline{s}$ to denote the (random) state following s, and $\Delta(s)$ to denote the probability simplex on A_s.

An Uncertain MDP (*UMDP*) is defined as an 8-tuple $< T, \gamma, S, A_s, \overline{\mathcal{P}}, \overline{\mathcal{R}}, \mathcal{P}, \mathcal{R} >$ where: $\overline{\mathcal{P}}$ and $\overline{\mathcal{R}}$ are the concentration sets: sets that the unknown parameters (transition probabilities and expected rewards respectively) are "likely" to belong to, while $\mathcal{P}$ and $\mathcal{R}$ are the deviation sets: sets that the parameters are guaranteed to belonging to. For this formulation to make sense, $\overline{\mathcal{P}} \subseteq \mathcal{P}$ and $\overline{\mathcal{R}} \subseteq \mathcal{R}$. A special case is when the concentration set is a singleton, representing the nominal (i.e., most possible) parameter realization. We use $\mathbb{E}_{\pi}^{\mathbf{p}}(\cdot)$ to represent taking expectation where the transition probability is $\mathbf{p}$, and the strategy to take is π.

We make the following assumption about the uncertainty set, which basically means that the parameters of different states are independent (we use the term "independent" but there is no probabilistic interpretation here). Such assumption is made by all papers investigating UMDPs to date, to the best of our knowledge.

ASSUMPTION 3.1. *State-wise Cartesian uncertainty sets:*

(i) $\overline{\mathcal{P}} = \prod_{s \in S} \overline{\mathcal{P}}_s$, $\overline{\mathcal{R}} = \prod_{s \in S} \overline{\mathcal{R}}_s$, $\mathcal{P} = \prod_{s \in S} \mathcal{P}_s$, $\mathcal{R} = \prod_{s \in S} \mathcal{R}_s$.

(ii) $\overline{\mathcal{P}}_s$, $\overline{\mathcal{R}}_s$, $\mathcal{P}_s$, $\mathcal{R}_s$ *are nonempty, convex and compact.*

Similarly to Nilim and El Ghaoui [**116**], we assume that when a state is visited for multiple times, each time it can take a different parameter realization (*non-stationary model*), mainly because the stationary model is generally intractable and a lower-bound on it is given by the non-stationary model. Therefore, multiple visits to a same state can be treated as visiting different states. By introducing dummy states, for finite horizon case we can make the following assumptions without loss of generality.

ASSUMPTION 3.2. *(i) Each state belongs to only one stage.*

(ii) The terminal reward equals zero.

(iii) The first stage only contains one state s^{ini}.

Using Assumption 3.2 (i), we partition S according to the stage each state belongs to. That is, we let S_t be the set of states belong to t^{th} stage. For a strategy π, we denote the expected (discounted) total-reward under parameters $\mathbf{p}$, $\mathbf{r}$ by $u(\pi, \mathbf{p}, \mathbf{r})$,

i.e.,

$$u(\pi, \mathbf{p}, \mathbf{r}) \triangleq \mathbb{E}_{\pi}^{\mathbf{p}}\{\sum_{i=1}^{T} \gamma^{i-1} r(s_i, a_i)\}.$$

3.2.2. Parametric linear programming. We briefly recall Parametric Linear Programming from linear programming textbooks (e.g., Bertsimas and Tsitsiklis [**24**], Ehrogtt [**62**], Murty [**113**]). A Parametric Linear Programming is the following set of infinitely many optimization problems:

$$\begin{aligned} &\text{For all } \lambda \in [0,1]\,, \\ &\quad \text{Minimize:} && \lambda {\mathbf{c}^{(1)}}^{\top}\mathbf{x} + (1-\lambda){\mathbf{c}^{(2)}}^{\top}\mathbf{x} \\ &\quad \text{Subject to:} && A\mathbf{x} = \mathbf{b} \\ &&& \mathbf{x} \geq 0. \end{aligned} \tag{3.1}$$

We call ${\mathbf{c}^{(1)}}^{\top}\mathbf{x}$ the first objective, and ${\mathbf{c}^{(2)}}^{\top}\mathbf{x}$ the second objective. We assume that the Linear Program is feasible and bounded for either objective. Although there are uncountably many possible λ, Problem (3.1) can be solved by a simplex-like algorithm. Here, "solve" means that for each λ, we find one optimal solution. An outline of the PLP algorithm is given in Algorithm 3.1 which is a tabular simplex method where the entering variable is determined in a specific way.

ALGORITHM 3.1. *Parametric Linear Program*

(1) Find a basic feasible optimal solution for $\lambda = 0$. If multiple solutions exist, choose one among those with minimal ${\mathbf{c}^{(1)}}^{\top}\mathbf{x}$.
(2) Record current basic feasible solution. Check the reduced cost of the first objective $\bar{c}_j^{(1)}$ for each column. Terminate if all of them are nonnegative.
(3) Among all columns with negative $\bar{c}_j^{(1)}$, choose the one with largest ratio $|\bar{c}_j^{(1)}/\bar{c}_j^{(2)}|$ as the entering variable.
(4) Pivot the base, go to 2.

This algorithm is based on the observation that for any λ, there exists an optimal *basic* feasible solution. Hence, by finding a suitable subset of all vertices of

the feasible region, we can solve the PLP. Furthermore, we can find this subset by sequentially pivoting among neighboring extreme points, and choose the one having the largest ratio between decrease of the first objective and increase of the second. The Pareto front of the two objectives (i.e., $\{(\mathbf{c}^{(1)^\top}\mathbf{x}, \mathbf{c}^{(2)^\top}\mathbf{x})|\mathbf{x}$ is Pareto efficient$\}$) is piecewise linear, and the number of pieces equals to the number of vertices pivoted. Hence if the problem is non-degenerate, then the algorithm is guaranteed to terminate within finitely many iterations. In the degenerate case, cycling can be avoided if an appropriate anti-cycling rule is followed (see [**113**]). The computational complexity is exponential in the number of the constraints and variables. That is, the number of pieces may grow exponentially, although practically this almost never happens. Such a characteristic is shared by all simplex based algorithms. It is also known that the optimal value for PLP is a continuous piecewise linear function of λ. See Bertsimas and Tsitsiklis [**24**], Ehrogtt [**62**], Murty [**113**] and other linear programming textbooks for detailed discussions.

3.3. MMEU based uncertain MDP: general case

In this section we propose an MMEU based criterion for uncertain MDPs, which incorporates prior information on how parameters spread and concentrate. We show in Section 3.3.1 that for finite horizon UMDP, a strategy defined through backward induction, which we called S-robust strategy is MMEU optimal. In addition, such strategy is computable in polynomial time under mild technical conditions. We generalize the notion of S-robust strategy to the infinite horizon case in Section 3.3.2, and show that it is MMEU optimal.

3.3.1. Finite horizon UMDP. We use the following set of distributions for our model.

$$\begin{aligned} \mathcal{C}_S(\lambda) &\triangleq \{\mu|\mu = \prod_{s\in S}\mu_s;\ \mu_s \in C_s(\lambda), \forall s \in S\}, \\ \text{where: } \mathcal{C}_s(\lambda) &\triangleq \{\mu_s|\mu_s(\mathcal{P}_s \times \mathcal{R}_s) = 1, \mu_s(\overline{\mathcal{P}}_s \times \overline{\mathcal{R}}_s) \geq \lambda\}. \end{aligned} \tag{3.2}$$

We briefly explain this set: for the unknown parameters of a state s, the condition $\mu_s(\mathcal{P}_s \times \mathcal{R}_s) = 1$ means that the parameters are restricted to the deviation set; while the condition $\mu_s(\overline{\mathcal{P}}_s \times \overline{\mathcal{R}}_s) \geq \lambda$ means that they belong to the concentration set with a high probability. Note that $\prod_{s\in S} \mu_s$ stands for the product measure generated by μ_s, which indicates that the parameters among different states are independent.

DEFINITION 3.1. *An MMEU strategy with respect to $C_S(\lambda)$ is*

$$\pi_M^* \triangleq \arg\max_{\pi\in\Pi^{HR}} \Big\{ \min_{\mu\in\mathcal{C}_S(\lambda)} \int u(\pi, \mathbf{p}, \mathbf{r})\, d\mu(\mathbf{p}, \mathbf{r}) \Big\}.$$

At first sight, computing an MMEU strategy w.r.t. $C_S(\lambda)$ seems formidable. In fact, evaluating the minimal (of all accessible priors) expected utility for a given strategy π already seems computational challenging, let alone finding an optimal strategy. Nevertheless, we show in Theorem 3.1 that the following S-robust strategy defined through a backward induction is the MMEU strategy. The proof is lengthy and hence deferred to Section 3.7.

DEFINITION 3.2. *Given $\lambda \in [0,1]$ and UMDP $< T, \gamma, S, A_s, \overline{\mathcal{P}}, \overline{\mathcal{R}}, \mathcal{P}, \mathcal{R} >$ with $T < \infty$ and $\gamma = 1$:*

(1) For $s \in S_T$, the S-robust value $\tilde{v}_T^\lambda(s) \triangleq 0$.

(2) For $s \in S_t$ where $t < T$, the S-robust value $\tilde{v}_t^\lambda(s)$ *and* S-robust action $\tilde{\pi}_s$ *are defined as*

$$\tilde{v}_t^\lambda(s) \triangleq \max_{\pi_{\mathbf{s}}\in\Delta(s)} \Big\{ \lambda \min_{\overline{\mathbf{p}}_s\in\overline{\mathcal{P}}_s, \overline{\mathbf{r}}_s\in\overline{\mathcal{R}}_s} \mathbb{E}_{\pi_s}^{\overline{\mathbf{p}}_s}[\overline{r}(s,a) + \tilde{v}_{t+1}^\lambda(\underline{s})] + (1-\lambda) \min_{\mathbf{p}_s\in\mathcal{P}_s, \mathbf{r}_s\in\mathcal{R}_s} \mathbb{E}_{\pi_s}^{\mathbf{p}_s}[r(s,a) + \tilde{v}_{t+1}^\lambda(\underline{s})] \Big\}.$$

$$\tilde{\pi}_s \in \arg\max_{\pi_{\mathbf{s}}\in\Delta(s)} \Big\{ \lambda \min_{\overline{\mathbf{p}}_s\in\overline{\mathcal{P}}_s, \overline{\mathbf{r}}_s\in\overline{\mathcal{R}}_s} \mathbb{E}_{\pi_s}^{\overline{\mathbf{p}}_s}[\overline{r}(s,a) + \tilde{v}_{t+1}^\lambda(\underline{s})] + (1-\lambda) \min_{\mathbf{p}_s\in\mathcal{P}_s, \mathbf{r}_s\in\mathcal{R}_s} \mathbb{E}_{\pi_s}^{\mathbf{p}_s}[r(s,a) + \tilde{v}_{t+1}^\lambda(\underline{s})] \Big\}.$$

(3) A strategy $\tilde{\pi}^*$ is a S-robust strategy *if $\forall s \in S$, $\tilde{\pi}_s^*$ is a S-robust action.*

THEOREM 3.1. *Under Assumptions 3.1 and 3.2, given $\lambda \in [0,1]$, $T < \infty$ and $\gamma = 1$, any S-robust strategy is a MMEU strategy w.r.t. $C_S(\lambda)$.*

REMARK:

(1) For finite horizon UMDPs, we assume that $\gamma = 1$ and that no terminal reward exists, purely for simplicity of expression. Such assumptions can be easily relaxed.

(2) A close examination of the proof of Theorem 3.1 shows that it is straightforward to generalize the S-robust strategy and equivalently the MMEU strategy to the case where the nested structure has more than two uncertainty sets.

We now investigate the computational aspect of the S-robust action. By backward induction we thus find the S-robust strategy.

THEOREM 3.2. *For $s \in S_t$ where $t < T$, the S-robust action is given by*

$$\mathbf{q}^* = \arg\max_{\mathbf{q}\in\Delta(s)} \Big\{\lambda\big[\min_{\overline{\mathbf{r}}_s\in\overline{\mathcal{R}}_s} \overline{\mathbf{r}}_s^\top\mathbf{q} + \min_{\overline{\mathbf{p}}_s\in\overline{\mathcal{P}}_s} \overline{\mathbf{p}}_s^\top\tilde{V}_s\mathbf{q}\big] + (1-\lambda)\big[\min_{\mathbf{r}_s\in\mathcal{R}_s} \mathbf{r}_s^\top\mathbf{q} + \min_{\mathbf{p}_s\in\mathcal{P}_s} \mathbf{p}_s^\top\tilde{V}_s\mathbf{q}\big]\Big\}, \tag{3.3}$$

where $m = |A_s|$, $\tilde{\mathbf{v}}_{t+1}$ is the vector form of $\tilde{v}_{t+1}(s')$ for all $s' \in S_{t+1}$, and

$$\tilde{V}_s \triangleq \begin{bmatrix} \tilde{\mathbf{v}}_{t+1}\mathbf{e}_1^\top(m) \\ \vdots \\ \tilde{\mathbf{v}}_{t+1}\mathbf{e}_m^\top(m) \end{bmatrix}.$$

PROOF. Notice that for any $\mathbf{q} \in \Delta(s)$, the following holds:

$$\begin{aligned}
&\lambda \min_{\overline{\mathbf{p}}_s\in\overline{\mathcal{P}}_s,\overline{\mathbf{r}}_s\in\overline{\mathcal{R}}_s} \mathbb{E}_{\mathbf{q}}^{\overline{\mathbf{p}}_s}[\overline{r}(s,a) + \tilde{v}_{t+1}(\underline{s})] + (1-\lambda) \min_{\mathbf{p}_s\in\mathcal{P}_s,\mathbf{r}_s\in\mathcal{R}_s} \mathbb{E}_{\mathbf{q}}^{\mathbf{p}_s}[r(s,a) + \tilde{v}_{t+1}(\underline{s})] \\
&= \lambda \min_{\overline{\mathbf{p}}_s\in\overline{\mathcal{P}}_s} \min_{\overline{\mathbf{r}}_s\in\overline{\mathcal{R}}_s} \Big[\sum_{a\in A_s} q(a)\overline{r}(s,a) + \sum_{a\in A_s}\sum_{s'\in S_{t+1}} q(a)\overline{p}(s'|s,a)\tilde{v}_{t+1}(s')\Big] \\
&\quad +(1-\lambda) \min_{\mathbf{p}_s\in\mathcal{P}_s} \min_{\mathbf{r}_s\in\mathcal{R}_s} \Big[\sum_{a\in A_s} q(a)r(s,a) + \sum_{a\in A_s}\sum_{s'\in S_{t+1}} q(a)p(s'|s,a)\tilde{v}_{t+1}(s')\Big] \\
&= \lambda\big[\min_{\overline{\mathbf{r}}_s\in\overline{\mathcal{R}}_s} \sum_{a\in A_s} q(a)\overline{r}(s,a) + \min_{\overline{\mathbf{p}}_s\in\overline{\mathcal{P}}_s} \sum_{a\in A_s}\sum_{s'\in S_{t+1}} q(a)\overline{p}(s'|s,a)\tilde{v}_{t+1}(s')\big] \\
&\quad +(1-\lambda)\big[\min_{\mathbf{r}_s\in\mathcal{R}_s} \sum_{a\in A_s} q(a)r(s,a) + \min_{\mathbf{p}_s\in\mathcal{P}_s} \sum_{a\in A_s}\sum_{s'\in S_{t+1}} q(a)p(s'|s,a)\tilde{v}_{t+1}(s')\big] \\
&= \lambda\big[\min_{\overline{\mathbf{r}}_s\in\overline{\mathcal{R}}_s} \overline{\mathbf{r}}_s^\top\mathbf{q} + \min_{\overline{\mathbf{p}}_s\in\overline{\mathcal{P}}_s} \overline{\mathbf{p}}_s^\top\tilde{V}_s\mathbf{q}\big] + (1-\lambda)\big[\min_{\mathbf{r}_s\in\mathcal{R}_s} \mathbf{r}_s^\top\mathbf{q} + \min_{\mathbf{p}_s\in\mathcal{P}_s} \mathbf{p}_s^\top\tilde{V}_s\mathbf{q}\big].
\end{aligned} \tag{3.4}$$

By definition, we have that the S-robust action is

$$\pi^* \in \arg\max_{\pi_s \in \Delta(s)} \Big\{ \lambda \min_{\overline{\mathbf{p}}_s \in \overline{\mathcal{P}}_s, \overline{\mathbf{r}}_s \in \overline{\mathcal{R}}_s} \mathbb{E}^{\overline{\mathbf{p}}_s}_{\pi_s}[\overline{r}(s,a) + \tilde{v}_{t+1}(\underline{s})] + (1-\lambda) \min_{\mathbf{p}_s \in \mathcal{P}_s, \mathbf{r}_s \in \mathcal{R}_s} \mathbb{E}^{\mathbf{p}_s}_{\pi_s}[r(s,a) + \tilde{v}_{t+1}(\underline{s})] \Big\}.$$

Hence, we establish the theorem by maximizing over $\mathbf{q} \in \Delta(s)$ on both side of Equation (3.4). □

Theorem 3.2 implies that the computation of the S-robust action at a state s critically depends on the structure of the sets $\overline{\mathcal{P}}_s$, $\overline{\mathcal{R}}_s$, $\mathcal{P}_s$ and $\mathcal{R}_s$. In fact, it can be shown that for "good" uncertainty sets, computing the S-robust action is tractable. To make this claim precise, we need the following definition.

DEFINITION 3.3. *A* polynomial separation oracle *of a convex set $\mathcal{H} \subseteq \mathbb{R}^n$ is a subroutine such that given $\mathbf{x} \in \mathbb{R}^n$, in polynomial time it decides whether $\mathbf{x} \in \mathcal{H}$, and if the answer is negative, it finds a hyperplane that separates $\mathbf{x}$ and $\mathcal{H}$.*

COROLLARY 3.3. *The S-robust action for state s can be found in polynomial-time, if each of $\overline{\mathcal{P}}_s$, $\overline{\mathcal{R}}_s$, $\mathcal{P}_s$ and $\mathcal{R}_s$ is convex and has a polynomial separation oracle.*

PROOF. It suffices to show that in polynomial time, the following optimization problem can be solved.

$$\begin{aligned} \text{Minimize:}_{\mathbf{q}} \ & \lambda \Big[\max_{\overline{\mathbf{r}}_s \in \overline{\mathcal{R}}_s} (-\overline{\mathbf{r}}_s^\top \mathbf{q}) + \max_{\overline{\mathbf{p}}_s \in \overline{\mathcal{P}}_s} (-\overline{\mathbf{p}}_s^\top \tilde{V}_s \mathbf{q}) \Big] + (1-\lambda) \Big[\max_{\mathbf{r}_s \in \mathcal{R}_s} (-\mathbf{r}_s^\top \mathbf{q}) + \max_{\mathbf{p}_s \in \mathcal{P}_s} (-\mathbf{p}_s^\top \tilde{V}_s \mathbf{q}) \Big] \\ & \text{s.t.: } \mathbf{q} \in \Delta(s). \end{aligned} \tag{3.5}$$

Notice that the objective function to be minimized is the maximum of a class of linear functions of $\mathbf{q}$, and hence convex. Therefore, if the sub-gradient of the objective function can be evaluated in polynomial time, the optimization problem (3.5) is solvable in polynomial time.

Due to the Envelope Theorem (e.g., Rockafellar [**126**]), it is known that for a function $f(\mathbf{x}) = \max_{\mathbf{y} \in \mathcal{C}} g(\mathbf{x}, \mathbf{y})$, the following holds

$$\nabla f(\mathbf{x}_0) = \nabla_{\mathbf{x}} g(\mathbf{x}_0, \mathbf{y}^*), \quad \text{where: } \mathbf{y}^* = \arg\max_{\mathbf{y} \in \mathcal{C}} g(\mathbf{x}_0, \mathbf{y}).$$

Notice that for fixed $(\overline{\mathbf{p}}_s, \overline{\mathbf{r}}_s, \mathbf{p}_s, \mathbf{r}_s)$, the objective function is linear. Hence, evaluation of the gradient is superficial. Thus, we only need to show that $\arg\max_{\overline{\mathbf{r}}_s \in \overline{\mathcal{R}}_s}(-\overline{\mathbf{r}}_s^\top \mathbf{q})$, $\arg\max_{\overline{\mathbf{p}}_s \in \overline{\mathcal{P}}_s}(-\overline{\mathbf{p}}_s^\top \tilde{V}_s \mathbf{q})$, $\arg\max_{\mathbf{r}_s \in \mathcal{R}_s}(-\mathbf{r}_s^\top \mathbf{q}_0)$ and $\arg\max_{\mathbf{p}_s \in \mathcal{P}_s}(-\mathbf{p}_s^\top \tilde{V}_s \mathbf{q}_0)$ can be found in polynomial time for any given $\mathbf{q}_0$. Notice all these problems are maximizing a linear objective over a compact set. A sufficient condition for polynomial-time solvability of such problem is that the set is convex and has a polynomial separation oracle (Grötschel et al. [**83**]). □

Having a polynomial separation oracle is a rather mild technical condition. Indeed, any convex set defined by finitely many convex constraints $g_i(\mathbf{x}) \leq 0$ has a polynomial (w.r.t. the number of constraints) separation oracle if both the value and the subgradient of $g_i(\cdot)$ can be evaluated in polynomial time (e.g., Ben-Tal and Nemirovski [**12**], Grötschel et al. [**83**]).

In practice, especially when the problem size is large, the theoretical guarantee of polynomial-time solvability may not ensure that the problem can be solved in reasonably short time. However, the following equivalence due to the convex duality (e.g., Boyd and Vandenberghe [**33**]) implies that if an uncertainty set is an intersection of a polytope and an ellipsoid (this includes trivially the case that the uncertainty case is a polytope or an ellipsoid), then finding the S-robust action is at most a second order cone programming.

EXAMPLE 3.1. *The following minimization problem*

$$\begin{aligned} \textit{Minimize:}_{\mathbf{x},\mathbf{u}} \quad & \mathbf{h}^\top \mathbf{x} \\ \textit{Subject to:} \quad & \mathbf{x} = A\mathbf{u} + b \\ & \|\mathbf{u}\|_2 \leq 1 \\ & \mathbf{C}\mathbf{x} \geq \mathbf{d}; \end{aligned}$$

is equivalent to

$$\begin{aligned} \textit{Maximize:}_{\mathbf{y},\mathbf{z}} \quad & -\|A^\top \mathbf{y}\|_2 - \mathbf{b}^\top \mathbf{y} + \mathbf{d}^\top \mathbf{z} \\ \textit{Subject to:} \quad & -\mathbf{y} + C^\top \mathbf{z} = \mathbf{h} \\ & \mathbf{z} \geq 0. \end{aligned}$$

The following example is a special case of Example 3.1 where all uncertainty sets are polytopes. Many practical setups can be formulated in this way. In particular, the arguably most "natural" uncertainty set where each parameter belongs to an interval is a polytope. Even if an uncertainty set is not a polytope, as long as it is convex it can be approximated by a polytope to arbitrarily precision.

EXAMPLE 3.2. *If $\overline{\mathcal{P}}_s$, $\overline{\mathcal{R}}_s$, $\mathcal{P}_s$ and $\mathcal{R}_s$ are polyhedral sets defined as $\overline{\mathcal{P}}_s = \{\overline{\mathbf{p}}_s | \overline{J}\overline{\mathbf{p}}_s \geq \overline{\mathbf{k}}\}$, $\overline{\mathcal{R}}_s = \{\overline{\mathbf{r}}_s | \overline{C}\overline{\mathbf{r}}_s \geq \overline{\mathbf{d}}\}$, $\mathcal{P}_s = \{\mathbf{p}_s | J\mathbf{p}_s \geq \mathbf{k}\}$ and $\mathcal{R}_s = \{\mathbf{r}_s | C\mathbf{r}_s \geq \mathbf{d}\}$, then the S-robust action equals the optimal $\mathbf{q}$ of the following Linear Program on $(\mathbf{q}, \overline{\mathbf{y}}, \overline{\mathbf{z}}, \mathbf{y}, \mathbf{z})$. In addition, the S-robust value equals its optimal value.*

$$\begin{aligned} \textit{Maximize:} \quad & \lambda\left[\overline{\mathbf{d}}^\top \overline{\mathbf{y}} + \overline{\mathbf{k}}^\top \overline{\mathbf{z}}\right] + (1-\lambda)\left[\mathbf{d}^\top \mathbf{y} + \mathbf{k}^\top \mathbf{z}\right] \\ \textit{Subject to:} \quad & C^\top \mathbf{y} = \mathbf{q}; \\ & J^\top \mathbf{z} = \tilde{V}_s \mathbf{q}; \\ & \overline{C}^\top \overline{\mathbf{y}} = \mathbf{q}; \\ & \overline{J}^\top \overline{\mathbf{z}} = \tilde{V}_s \mathbf{q}; \\ & \mathbf{1}^\top \mathbf{q} = 1; \\ & \overline{\mathbf{y}}, \overline{\mathbf{z}}, \mathbf{y}, \mathbf{z}, \mathbf{q} \geq 0. \end{aligned} \tag{3.6}$$

3.3.2. Discounted reward infinite horizon UMDP. In this subsection we consider the MMEU strategy for discounted-reward infinite-horizon UMDPs. Unlike the finite horizon case, we cannot model the system as having finitely many states, each visited at most once. In contrast, there are two different approaches to model such setup. The first approach is to treat the system as having infinite number of

states, each visited at most once (*non-stationary model*). Therefore, we use a pair (s,t) where $s \in S$ and t is the stage to define an augmented state. The set of distributions to be considered is thus

$$\mathcal{C}_S^{\infty}(\lambda) \triangleq \{\mu | \mu = \prod_{s\in S, t=1,2,\cdots} \mu_{(s,t)};\ \mu_{(s,t)} \in C_s(\lambda), \forall s \in S,\ \forall t = 1,2,\cdots\}.$$

An alternative approach is having finitely many states with multiple visits (*stationary-model*). This is equivalent to the following set of priors.

$$\mathcal{C}_S(\lambda) \triangleq \{\mu | \mu = \prod_{s\in S, t=1,2,\cdots} \mu_{(s,t)};\ \mu_{(s,t)} = \mu_s; \mu_s \in C_s(\lambda), \forall s \in S,\ \forall t = 1,2,\cdots\}.$$

It turns out that for both models the MMEU strategies are the same, and given by the S-robust strategy defined as follows.

DEFINITION 3.4. *Given* $\lambda \in [0,1]$ *and UMDP* $< T, \gamma, S, A_s, \overline{\mathbf{r}}, \overline{\mathbf{p}}, \mathcal{R}, \mathcal{P} >$ *with* $T = \infty$ *and* $\gamma < 1$*:*

(1) The S-robust value $\tilde{v}_\infty(s)$ *is the unique solution to the following set of equations:*

$$\tilde{v}_\infty(s) = \max_{\pi_{\mathbf{s}} \in \Delta(s)} \Big\{ \lambda \min_{\overline{\mathbf{p}}_s \in \overline{\mathcal{P}}_s, \overline{\mathbf{r}}_s \in \overline{\mathcal{R}}_s} \mathbb{E}_{\pi_s}^{\overline{\mathbf{p}}_s}[\overline{r}(s,a) + \gamma \tilde{v}_\infty(\underline{s})] + (1-\lambda) \min_{\mathbf{p}_s \in \mathcal{P}_s, \mathbf{r}_s \in \mathcal{R}_s} \mathbb{E}_{\pi_s}^{\mathbf{p}_s}[r(s,a) + \gamma \tilde{v}_\infty(\underline{s})] \Big\};\ \forall s \in S.$$

(2) The S-robust action $\tilde{\pi}_s$ *is given by*

$$\tilde{\pi}_s \in \arg\max_{\pi_{\mathbf{s}} \in \Delta(s)} \Big\{ \lambda \min_{\overline{\mathbf{p}}_s \in \overline{\mathcal{P}}_s, \overline{\mathbf{r}}_s \in \overline{\mathcal{R}}_s} \mathbb{E}_{\pi_s}^{\overline{\mathbf{p}}_s}[\overline{r}(s,a) + \gamma \tilde{v}_\infty(\underline{s})] + (1-\lambda) \min_{\mathbf{p}_s \in \mathcal{P}_s, \mathbf{r}_s \in \mathcal{R}_s} \mathbb{E}_{\pi_s}^{\mathbf{p}_s}[r(s,a) + \gamma \tilde{v}_\infty(\underline{s})] \Big\};\ \forall s \in S.$$

(3) A strategy $\tilde{\pi}^*$ *is a* S-robust strategy *if* $\forall s \in S$, $\tilde{\pi}_s^*$ *is a S-robust action.*

To see that the S-robust strategy is well defined, it suffices to show that the following operator $\mathcal{L} : \mathbb{R}^{|S|} \to \mathbb{R}^{|S|}$ is a γ contraction for the $\|\cdot\|_\infty$ norm.

$$\{\mathcal{L}\mathbf{v}\}(s) \triangleq \max_{\mathbf{q}\in\Delta(s)} \min_{\overline{\mathbf{p}}\in\overline{\mathcal{P}}_s, \overline{\mathbf{r}}\in\overline{\mathcal{R}}_s, \mathbf{p}\in\mathcal{P}_s, \mathbf{r}\in\mathcal{R}_s} \{\mathcal{L}^{\mathbf{q}}_{\overline{\mathbf{p}},\overline{\mathbf{r}},\mathbf{p},\mathbf{r}}\mathbf{v}\}(s);$$

where: $$\{\mathcal{L}^{\mathbf{q}}_{\overline{\mathbf{p}},\overline{\mathbf{r}},\mathbf{p},\mathbf{r}}\mathbf{v}\}(s) \triangleq \lambda\big[\sum_{a\in A_s} q(a)\overline{r}(s,a) + \gamma \sum_{a\in A_s}\sum_{s'\in S} q(a)\overline{p}(s'|s,a)v(s')\big]$$
$$+ (1-\lambda)\big[\sum_{a\in A_s} q(a)r(s,a) + \gamma \sum_{a\in A_s}\sum_{s'\in S} q(a)p(s'|s,a)v(s')\big].$$

LEMMA 3.4. *Under Assumption 3.1, $\mathcal{L}$ is a γ contraction for $\|\cdot\|_\infty$ norm.*

PROOF. Observe that $\mathcal{L}^{\mathbf{q}}_{\overline{\mathbf{p}},\overline{\mathbf{r}},\mathbf{p},\mathbf{r}}$ is a γ contraction for any given $(\mathbf{q}, \overline{\mathbf{p}}, \overline{\mathbf{r}}, \mathbf{p}, \mathbf{r})$. For arbitrary $\mathbf{v}_1$ and $\mathbf{v}_2$, let $\mathbf{q}_{1,2}$, $\overline{\mathbf{p}}_{1,2}$, $\overline{\mathbf{r}}_{1,2}$, $\mathbf{p}_{1,2}$, $\mathbf{r}_{1,2}$ be the respective maximizing and minimizing variables, we have

$$\{\mathcal{L}\mathbf{v}_1\}(s) - \{\mathcal{L}\mathbf{v}_2\}(s) = \mathcal{L}^{\mathbf{q}_1(s)}_{\overline{\mathbf{p}}_1(s),\overline{\mathbf{r}}_1(s),\mathbf{p}_1(s),\mathbf{r}_1(s)}\mathbf{v}_1(s) - \mathcal{L}^{\mathbf{q}_2(s)}_{\overline{\mathbf{p}}_2(s),\overline{\mathbf{r}}_2(s),\mathbf{p}_2(s),\mathbf{r}_2(s)}\mathbf{v}_2(s)$$
$$\le \mathcal{L}^{\mathbf{q}_1(s)}_{\overline{\mathbf{p}}_2(s),\overline{\mathbf{r}}_2(s),\mathbf{p}_2(s),\mathbf{r}_2(s)}\mathbf{v}_1(s) - \mathcal{L}^{\mathbf{q}_1(s)}_{\overline{\mathbf{p}}_2(s),\overline{\mathbf{r}}_2(s),\mathbf{p}_2(s),\mathbf{r}_2(s)}\mathbf{v}_2(s) \le \gamma\|\mathbf{v}_1 - \mathbf{v}_2\|_\infty;$$

Similarly, $\{\mathcal{L}\mathbf{v}_2\}(s) - \{\mathcal{L}\mathbf{v}_1\}(s) \le \gamma\|\mathbf{v}_2 - \mathbf{v}_1\|_\infty = \gamma\|\mathbf{v}_1 - \mathbf{v}_2\|_\infty$. Hence we establish the lemma. □

Lemma 3.4 indeed implies that by applying $\mathcal{L}$ on any initial $\mathbf{v} \in \mathbb{R}^{|S|}$ repeatedly, we can approximate the S-Robust strategy to arbitrary accuracy.

THEOREM 3.5. *Under Assumption 3.1, given $\lambda \in [0,1]$, $T = \infty$ and $\gamma < 1$,*

(1) any S-robust strategy is a MMEU strategy w.r.t. $C_S^\infty(\lambda)$;

(2) any S-robust strategy is a MMEU strategy w.r.t. $C_S(\lambda)$.

PROOF. NON-STATIONARY MODEL: We introduce the following $\hat{T}$-truncated problem: with the total reward

$$u_{\hat{T}}(\pi, \mathbf{p}, \mathbf{r}) \triangleq \mathbb{E}^{\mathbf{p}}_\pi\{\sum_{i=1}^{\hat{T}} \gamma^{i-1} r(s_i, a_i) + \gamma^{\hat{T}}\tilde{v}_\infty(s_{\hat{T}})\}.$$

That is, the problem stops at stage $\hat{T}$ with a termination reward $\tilde{v}_\infty(\cdot)$. Notice $|S|$ is finite, and all $\mathcal{R}_s$ are bounded. Hence there exists a universal constant c (independent

of $\hat{T}$) such that for any $(\pi, \mathbf{p}, \mathbf{r})$ where $\mathbf{r} \in \mathcal{R}$, the following holds:

$$\left|u_{\hat{T}}(\pi, \mathbf{p}, \mathbf{r}) - u(\pi, \mathbf{p}, \mathbf{r})\right| \leq \gamma^{\hat{T}} c.$$

This implies for any $\mu \in \mathcal{C}_S^\infty(\lambda)$,

$$\Big| \int u_{\hat{T}}(\pi, \mathbf{p}, \mathbf{r})\, d\mu(\mathbf{p}, \mathbf{r}) - \int u(\pi, \mathbf{p}, \mathbf{r})\, d\mu(\mathbf{p}, \mathbf{r})\Big| \leq \gamma^{\hat{T}} c, \tag{3.7}$$

which further leads to

$$\Big| \min_{\mu \in \mathcal{C}_S^\infty(\lambda)} \int u_{\hat{T}}(\pi, \mathbf{p}, \mathbf{r})\, d\mu(\mathbf{p}, \mathbf{r}) - \min_{\mu' \in \mathcal{C}_S^\infty(\lambda)} \int u(\pi, \mathbf{p}, \mathbf{r})\, d\mu'(\mathbf{p}, \mathbf{r})\Big| \leq \gamma^{\hat{T}} c. \tag{3.8}$$

By Theorem 3.1, it is easy to see that the S-robust strategy π^* is an MMEU strategy of the (finite horizon) $\hat{T}$ truncated problem. That is,

$$\min_{\mu \in \mathcal{C}_S^\infty(\lambda)} \int u_{\hat{T}}(\pi^*, \mathbf{p}, \mathbf{r})\, d\mu(\mathbf{p}, \mathbf{r}) \geq \min_{\mu' \in \mathcal{C}_S^\infty(\lambda)} \int u_{\hat{T}}(\pi', \mathbf{p}, \mathbf{r})\, d\mu'(\mathbf{p}, \mathbf{r}), \quad \forall \pi' \in \Pi^{HR}.$$

Combining it with Inequality (3.8), we have

$$\min_{\mu \in \mathcal{C}_S^\infty(\lambda)} \int u(\pi^*, \mathbf{p}, \mathbf{r})\, d\mu(\mathbf{p}, \mathbf{r}) \geq \min_{\mu' \in \mathcal{C}_S^\infty(\lambda)} \int u(\pi', \mathbf{p}, \mathbf{r})\, d\mu'(\mathbf{p}, \mathbf{r}) - 2\gamma^{\hat{T}} c, \quad \forall \pi' \in \Pi^{HR}.$$

Notice that this holds for arbitrary $\hat{T}$, hence we have

$$\pi^* \in \arg \max_{\pi \in \Pi^{HR}} \min_{\mu \in \mathcal{C}_S^\infty(\lambda)} \int u(\pi, \mathbf{p}, \mathbf{r})\, d\mu(\mathbf{p}, \mathbf{r}),$$

which by definition is the MMEU strategy w.r.t. $\mathcal{C}_S^\infty(\lambda)$ of the infinite horizon UMDP.

Stationary Model: Again consider the $\hat{T}$ truncated problem. Following the proof of Theorem 3.1, a Nash equilibrium (π^*, μ^*) exists for

$$\Big\{ \int u_{\hat{T}}(\pi, \mathbf{p}, \mathbf{r})\, d\mu(\mathbf{p}, \mathbf{r})\Big\},$$

while $\pi^*_{(s,t)} = \mathbf{q}^*_{(s,t)}$ is the S-robust strategy, and $\mu^*_{(s,t)}$ is the a probability measure such that $\mu^*_{(s,t)}(\mathbf{p}^*_{(s,t)}, \mathbf{r}^*_{(s,t)}) = 1 - \lambda$ and $\mu^*_{(s,t)}(\overline{\mathbf{p}}_{(s,t)}, \overline{\mathbf{r}}_{(s,t)}) = \lambda$. Here $(\pi^*_{(s,t)}, (\mathbf{p}^*_{(s,t)}, \mathbf{r}^*_{(s,t)}))$ is a Nash Equilibrium for the one-stage game

$$\Big\{\lambda \mathbb{E}^{\overline{\mathbf{p}}_s}_{\pi_s}[\overline{r}(s,a) + \gamma \tilde{v}_\infty(\underline{s})] + (1-\lambda)\mathbb{E}^{\mathbf{p}_s}_{\pi_s}[r(s,a) + \gamma \tilde{v}_\infty(\underline{s})]\Big\},$$

and hence are stationary (i.e., not dependent on t). Thus, μ^* is stationary, i.e., $\mu^* \in \mathcal{C}_S(\lambda)$. Hence,

$$\max_{\pi \in \Pi^{HR}} \min_{\mu \in \mathcal{C}_S(\lambda)} \int u(\pi, \mathbf{p}, \mathbf{r})\, d\mu(\mathbf{p}, \mathbf{r}) \leq \max_{\pi \in \Pi^{HR}} \int u(\pi, \mathbf{p}, \mathbf{r})\, d\mu^*(\mathbf{p}, \mathbf{r}).$$

Further, the fact that (π^*, μ^*) is a Nash Equilibrium implies

$$\max_{\pi \in \Pi^{HR}} \int u_{\hat{T}}(\pi, \mathbf{p}, \mathbf{r})\, d\mu^*(\mathbf{p}, \mathbf{r}) = \int u_{\hat{T}}(\pi^*, \mathbf{p}, \mathbf{r})\, d\mu^*(\mathbf{p}, \mathbf{r}) = \min_{\mu \in C_S^\infty(\lambda)} \int u_{\hat{T}}(\pi^*, \mathbf{p}, \mathbf{r})\, d\mu(\mathbf{p}, \mathbf{r}),$$

which leads to

$$\begin{aligned} &\max_{\pi \in \Pi^{HR}} \int u(\pi, \mathbf{p}, \mathbf{r})\, d\mu^*(\mathbf{p}, \mathbf{r}) \\ \leq &\min_{\mu \in \mathcal{C}_S^\infty(\lambda)} \int u(\pi^*, \mathbf{p}, \mathbf{r})\, d\mu(\mathbf{p}, \mathbf{r}) + 2\gamma^{\hat{T}} c \\ \leq &\min_{\mu \in \mathcal{C}_S(\lambda)} \int u(\pi^*, \mathbf{p}, \mathbf{r})\, d\mu(\mathbf{p}, \mathbf{r}) + 2\gamma^{\hat{T}} c. \end{aligned}$$

The first inequality holds from (3.7). The second inequality holds because $\mathcal{C}_S(\lambda) \subseteq \mathcal{C}_S^\infty(\lambda)$. Since $\hat{T}$ can be arbitrarily large, we have

$$\max_{\pi \in \Pi^{HR}} \min_{\mu \in \mathcal{C}_S(\lambda)} \int u(\pi, \mathbf{p}, \mathbf{r})\, d\mu(\mathbf{p}, \mathbf{r}) \leq \min_{\mu \in \mathcal{C}_S(\lambda)} \int u(\pi^*, \mathbf{p}, \mathbf{r})\, d\mu(\mathbf{p}, \mathbf{r}).$$

By definition, this shows that the S-robust strategy π^* is the MMEU w.r.t. $\mathcal{C}_S(\lambda)$. □

3.4. MMEU based uncertain MDP: known dynamics

This section is devoted to a special class of UMDPs: only the reward parameters are subject to uncertainty whereas the transition probabilities are precisely known. That is, the following assumption holds.

ASSUMPTION 3.3. *(i)* $\overline{\mathcal{P}} = \mathcal{P} = \{\overline{\mathbf{p}}\}$.

Such a setup can either model or approximate many practical problems. For instance, a shortest-path problem with uncertain link lengths is an UMDP with known dynamics (e.g., Puterman [**124**]). Another example is using state aggregation to solve large scale MDPs (Singh et al. [**138**]). In such case, states are grouped to a small

number of hyper-states and a reduced MDP built on these hyper-states is investigated. Typically, the transition law between hyper-states is known, but the expected reward visiting each hyper-state is uncertain due to the transitions inside each hyper-state.

The known-dynamics setup is of special interest for the following two reasons. First, the S-robust criterion has an appealing interpretation as finding Pareto efficient tradeoffs of the likely performance and the worst-case performance (L/W tradeoff for short), which we discuss in Section 3.4.1. Second, if we further assume that the uncertainty sets $\mathcal{R}_s$ are polyhedral, then a single run of an algorithm proposed in Section 3.4.2 can find the S-robust strategies for *all* $\lambda \in [0, 1]$. Consequently, we can find all efficient tradeoffs. This makes it possible for the decision maker to observe the whole tradeoff relationship and choose the most desirable solution according to her (possibly complicated) preference.

3.4.1. Likely/Worst-case tradeoff. Under Assumption 3.3, consider the following two functions of a (history-dependent) strategy $\pi \in \Pi^{HR}$:

$$L(\pi) \triangleq \min_{\overline{\mathbf{r}} \in \overline{\mathcal{R}}} \mathbb{E}_\pi \{ \sum_{i=1}^{T} \gamma^{i-1} \overline{r}(s_i, a_i) \}; \qquad W(\pi) \triangleq \min_{\mathbf{r} \in \mathcal{R}} \mathbb{E}_\pi \{ \sum_{i=1}^{T} \gamma^{i-1} r(s_i, a_i) \}.$$

Here, $L(\cdot)$ measures what is the likely performance of a strategy, and $W(\cdot)$ bounds the downside deviation due to parameters uncertainties.

Further notice that since the transition probability is known, strategy π determines the state-action frequency vector $\mathbf{x}^\pi$ defined as

$$x^\pi(s, a) \triangleq \mathbb{E}_\pi \sum_{i=1}^{T} \gamma^{i-1} \mathbf{I}(s_i = s, a_i = a).$$

For any fixed $\mathbf{r} \in \mathcal{R}$, the expected total (discounted) reward is a linear function of $\mathbf{x}^\pi$. Thus, $L(\pi)$ and $W(\pi)$ are concave functions of $\mathbf{x}^\pi$. It is widely known (e.g., Puterman [**124**]) that the set of $\mathbf{x}^\pi$ for all $\pi \in \Pi^{HR}$ is a polytope. That is, we are maximizing two concave functions in a closed convex set. Therefore, any Pareto efficient strategy for $L(\cdot)$ and $W(\cdot)$ is obtained by maximizing their convex combinations, and vice versa. Hence we have the following definition.

DEFINITION 3.5. *A Likely/Worst-case tradeoff strategy for* $\lambda \in [0, 1]$ *is defined as*

$$\pi^*_{LW} \in \arg\max_{\pi \in \Pi^{HR}} \Big\{\lambda L(\pi) + (1-\lambda) W(\pi)\Big\}.$$

Indeed, we have the following two theorems showing that the L/W tradeoff criterion coincides with the S-robust (equivalently MMEU) criterion for both finite horizon UMDPs and discounted-reward infinite horizon UMDPs. This therefore provides an alternative justification of the MMEU formulation.

THEOREM 3.6. *Under Assumptions 3.1, 3.2 and 3.3 , when* $T < \infty$ *and* $\gamma = 1$*, any S-robust strategy is a Likely/Worst-case tradeoff strategy.*

PROOF. We first define the following quantities for $t = 1, \cdots, T$, a length-t history h_t, and $\lambda \in [0, 1]$:

$$\begin{aligned} L_t(\pi, h_t) &\triangleq \min_{\overline{\mathbf{r}} \in \overline{\mathcal{R}}} \mathbb{E}_\pi \left\{ \sum_{i=t}^{T} \overline{r}(s_i, a_i) | h_t \right\}; \quad W_t(\pi, h_t) \triangleq \min_{\mathbf{r} \in \mathcal{R}} \mathbb{E}_\pi \left\{ \sum_{i=t}^{T} r(s_i, a_i) | h_t \right\} \\ c_t^\lambda(h_t) &\triangleq \max_{\pi \in \Pi^{HR}} \left\{ \lambda L_t(\pi, h_t) + (1-\lambda) W_t(\pi, h_t) \right\}. \end{aligned} \tag{3.9}$$

Note that $L_T(\cdot) \equiv W_T(\cdot) \equiv c_T(\cdot) \equiv 0$, $L(\pi) = L_1(\pi, (s^{\text{ini}}))$, and $W(\pi) = W_1(\pi, (s^{\text{ini}}))$, because $\gamma = 1$, and Assumption 3.2 holds. Thus, by definition, any strategy that achieves $c_1^\lambda((s^{\text{ini}}))$ is a L/W tradeoff strategy.

Let π^* be a S-robust strategy. We apply backward induction to prove that the following holds for any h_t and $\pi' \in \Pi^{HR}$:

$$\begin{aligned} &\lambda L_t(\pi', h_t) + (1-\lambda) W_t(\pi', h_t) \le \lambda L_t(\pi^*, h_t) + (1-\lambda) W_t(\pi^*, h_t); \\ &\tilde{v}_t^\lambda(s(h_t)) = c_t^\lambda(h_t) = \lambda L_t(\pi^*, h_t) + (1-\lambda) W_t(\pi^*, h_t), \end{aligned} \tag{3.10}$$

here $s(h_t)$ stands for the last state of the history h_t.

For $t = T$, (3.10) holds trivially. Now assume that (3.10) holds for all h_{t+1}, we show it also hold for an arbitrary length-t history h_t. Let $s = s(h_t)$. We thus have

$$
\begin{aligned}
&\lambda L_t(\pi', h_t) + (1-\lambda) W_t(\pi', h_t) \\
=&\min_{\overline{\mathbf{r}} \in \overline{\mathcal{R}}, \mathbf{r} \in \mathcal{R}} \Big\{ \big[\lambda \sum_{a \in A_s} q'(a) \overline{r}(s,a) + (1-\lambda) \sum_{a \in A_s} q'(a) r(s,a)\big] \\
&\quad + \sum_{s' \in S_{t+1}} \sum_{a \in A_s} q'(a) p(s'|s,a) \mathbb{E}_{\pi'} \big[\lambda \sum_{i=t+1}^{T} \overline{r}(s_i, a_i) + (1-\lambda) \sum_{i=t+1}^{T} r(s_i, a_i) \big| (h_t, a, s')\big] \Big\} \\
=&\min_{\overline{\mathbf{r}}_s \in \overline{\mathcal{R}}_s, \mathbf{r}_s \in \mathcal{R}_s} \Big\{ \lambda \sum_{a \in A_s} q'(a) \overline{r}(s,a) + (1-\lambda) \sum_{a \in A_s} q'(a) r(s,a) \Big\} \\
&\quad + \sum_{s' \in S_{t+1}} \sum_{a \in A_s} q'(a) p(s'|s,a) \Big\{ \min_{\overline{\mathbf{r}} \in \overline{\mathcal{R}}, \mathbf{r} \in \mathcal{R}} \mathbb{E}_{\pi'} \big[\lambda \sum_{i=t+1}^{T} \overline{r}(s_i, a_i) + (1-\lambda) \sum_{i=t+1}^{T} r(s_i, a_i) \big| (h_t, a, s')\big] \Big\} \\
\leq&\min_{\overline{\mathbf{r}}_s \in \overline{\mathcal{R}}_s, \mathbf{r}_s \in \mathcal{R}_s} \Big\{ \lambda \sum_{a \in A_s} q'(a) \overline{r}(s,a) + (1-\lambda) \sum_{a \in A_s} q'(a) r(s,a) \Big\} \\
&\quad + \sum_{s' \in S_{t+1}} \sum_{a \in A_s} q'(a) p(s'|s,a) c_{t+1}^{\lambda}((h_t, a, s')) \\
=&\min_{\overline{\mathbf{r}}_s \in \overline{\mathcal{R}}_s, \mathbf{r}_s \in \mathcal{R}_s} \Big\{ \lambda \sum_{a \in A_s} q'(a) \overline{r}(s,a) + (1-\lambda) \sum_{a \in A_s} q'(a) r(s,a) \Big\} \\
&\quad + \sum_{s' \in S_{t+1}} \sum_{a \in A_s} q'(a) p(s'|s,a) \tilde{v}_{t+1}^{\lambda}(s') \\
\leq&\max_{\mathbf{q} \in \Delta_s} \Big\{ \min_{\overline{\mathbf{r}}_s \in \overline{\mathcal{R}}_s, \mathbf{r}_s \in \mathcal{R}_s} \big[\lambda \sum_{a \in A_s} q(a) \overline{r}(s,a) + (1-\lambda) \sum_{a \in A_s} q(a) r(s,a)\big] \\
&\quad + \sum_{s' \in S_{t+1}} \sum_{a \in A_s} q(a) p(s'|s,a) \tilde{v}_{t+1}^{\lambda}(s') \Big\} \\
=&\tilde{v}_t^{\lambda}(s).
\end{aligned}
$$

The second equality holds because $\mathcal{R}$ is statewise Cartesian. Note that both inequalities hold with equality for π^* due to the backward induction assumption and the definition of a S-robust strategy, respectively. Therefore, (3.10) holds for h_t, which completes the backward induction.

Substituting $h_1 = (s^{\text{ini}})$ into (3.10) proves the theorem. □

THEOREM 3.7. *Under Assumptions 3.1 and 3.3 , when $T = \infty$ and $\gamma < 1$, any S-robust strategy is a Likely/Worst-case tradeoff strategy.*

PROOF. Similarly to the proof of Theorem 3.5, we consider a $\hat{T}$ truncated MDP, with $\tilde{v}_\infty(s)$ the terminal reward. Thus, the S-robust strategy maximizes for all finite $\hat{T}$:

$$\hat{u}^{\hat{T}}(\pi) \triangleq \lambda \min_{\overline{\mathbf{r}}_s \in \overline{\mathcal{R}}_s} \mathbb{E}_\pi\{\sum_{i=1}^{\hat{T}} \gamma^{i-1}\overline{r}(s_i,a_i)\}+(1-\lambda)\min_{\mathbf{r}\in\mathcal{R}} \mathbb{E}_\pi\{\sum_{i=1}^{\hat{T}} \gamma^{i-1} r(s_i,a_i)\}+\mathbb{E}_\pi(\gamma^{\hat{T}}\tilde{v}_\infty(s)).$$

Further notice that $|S|$ is finite and $\mathcal{R}_s$ is bounded. Thus there exists a universal constant c independent of $\hat{T}$ such that for all π,

$$|\lambda L(\pi) + (1-\lambda)W(\pi) - \hat{u}^{\hat{T}}(\pi)| \leq |\max_{\mathbf{r}\in\mathcal{R}} \mathbb{E}^\pi \sum_{t=\hat{T}}^{\infty} \gamma^t r(s_i,a_i)| + |\mathbb{E}^\pi \gamma^{\hat{T}} \tilde{v}_\infty(s)| \leq \gamma^{\hat{T}} c.$$

Since $\hat{T}$ is arbitrary, π^* is hence optimal to $\lambda P(\pi) + (1-\lambda)R(\pi)$, i.e., it is the L/W tradeoff strategy for discounted reward infinite horizon UMDP. □

We need to point out that Assumption 3.3 is essential. Indeed, if the system dynamics are not precisely known, the L/W strategy can be non-Markovian, which implies possible intractability. We show this with the following example.

Consider a finite horizon MDP shown in the Figure 3.1:
$S = \{s1, s2, s3, s4, s5, t1, t2, t3, t4, Terminal\}$; $A_{s1} = \{a(1,1)\}$; $A_{s2} = \{a(2,1)\}$; $A_{s3} = \{a(3,1)\}$; $A_{s4} = \{a(4,1)\}$ and $A_{s5} = \{a(5,1), a(5,2)\}$. Rewards are only available at the third stage, and are perfectly known. The set $\overline{\mathcal{P}}$ is a singleton:

$$\overline{p}\,(s2|s1,a(1,1)) = 0.5; \quad \overline{p}\,(s4|s2,a(2,1)) = 1; \quad \overline{p}\,(t3|s5,a(5,2)) = 1.$$

The set $\mathcal{P}$ is such that

$$p\,(s2|s1,a(1,1)) \in \{0.5\}; \quad p\,(s4|s2,a(2,1)) \in [0,1]; \quad p\,(t3|s5,a(5,2)) \in [0,1].$$

Observe that the worst parameter realization (for all strategies) is

$$p(s4|s2,a(2,1)) = p(t3|s5,a(5,2)) = 0.$$

We consider $\lambda = 0.5$. Since multiple actions only exist in state $s5$, a strategy is determined by the action chosen on $s5$. Let the probability of choosing action $a(5,1)$ and $a(5,2)$ be q and $1-q$, respectively.

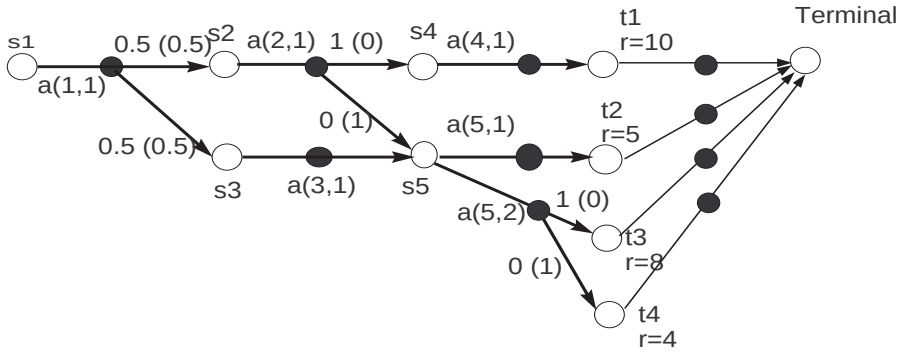


FIGURE 3.1. A general UMDP with a non-Markovian L/W strategy

Consider the transition trajectory "$s1 \to s2$". Under the nominal transition probability, this trajectory will reach $t1$ with a reward of 10, regardless of the choice of q. The worst transition is that action $a(2,1)$ leads to $s5$ and action $a(5,2)$ leads to $t4$, where the expected reward is $5q + 4(1-q)$. Hence the optimal action is choosing $a(5,1)$ deterministically.

Consider the transition trajectory "$s1 \to s3$". In this case, the nominal reward is $5q + 8(1-q)$, and the worst case reward is $5q + 4(1-q)$. Thus $q = 0$ optimizes the weighted sum, i.e., the optimal strategy is choosing $a(5,2)$.

Therefore, for this example the (unique) L/W tradeoff strategy is non-Markovian. This is due to the fact that we are taking expectations over two different probability measures, hence the smoothing property of conditional expectation cannot be used. From the computational perspective, this non-Markovian property implies a possibility that past actions affect the choice of future actions, and hence could render the problem intractable in general.

3.4.2. Finding S-robust strategies for all λ. In this subsection we make an additional assumption: the uncertainty sets are polytopes. A notable example arises when each parameter belongs to an interval.

ASSUMPTION 3.4. *For any $s \in S$, there exist $\overline{C}_s$, C_s , $\overline{\mathbf{d}}_s$ and $\mathbf{d}_s$ such that*

$$\overline{\mathbf{R}}_s = \{\overline{\mathbf{r}}_s | \overline{C}_s \overline{\mathbf{r}}_s \geq \overline{\mathbf{d}}_s\}; \quad \mathbf{R}_s = \{\mathbf{r}_s | C_s \mathbf{r}_s \geq \mathbf{d}_s\}.$$

We show how to find the whole set of S-robust strategies for *all* λ. At the mean time, we also prove by backward induction that $\tilde{v}_t^{\lambda}(s)$ is a continuous and piecewise linear function of λ. The algorithm is based on Parametric Linear Programming (PLP), which solves the following set of infinitely many optimization problems over $\mathbf{x}$:

$$\begin{aligned} &\text{For all } \lambda \in [0,1]\,, \\ &\quad \text{Minimize:} && \lambda \mathbf{c}^{(1)^\top}\mathbf{x} + (1-\lambda)\mathbf{c}^{(2)^\top}\mathbf{x} \\ &\quad \text{Subject to:} && A\mathbf{x} = \mathbf{b} \\ &&& \mathbf{x} \geq 0. \end{aligned}$$

PLP can be solved using Algorithm 3.1, which is provided in Appendix 3.2.2 for completeness.

Consider the finite horizon case first. Let $S_{t+1} = \{s^1, \cdots, s^k\}$. Assume for all $j \in \{1, \cdots, k\}$, $\tilde{v}_{t+1}^{\lambda}(s^j)$ are continuous piecewise linear functions. Thus, we can divide $[0,1]$ into finite (say n) intervals $[0, \lambda_1], \cdots [\lambda_{n-1}, 1]$ such that in each interval, for any j, $\tilde{v}_{t+1}^{(\cdot)}(s^j)$ is a linear function of λ. That is, there exist constants l_i^j and m_i^j such that $\tilde{v}_{t+1}^{\lambda}(s^j) = l_i^j \lambda + m_i^j$, for $\lambda \in [\lambda_{i-1}, \lambda_i]$[2]. By Example 3.2, and since $\overline{\mathcal{P}} = \mathcal{P} = \{\overline{\mathbf{p}}\}$, we

[2]We let $\lambda_0 = 0$ and $\lambda_n = 1$ for consistency of notation.

have that $\tilde{v}_t^{\lambda}(s)$ equals to the optimal value of the following LP on $\overline{\mathbf{y}}$, $\mathbf{y}$ and $\mathbf{q}$.

$$\begin{aligned} \text{Maximize: } & (1-\lambda)\mathbf{d}_s^{\top}\mathbf{y} + \lambda\overline{\mathbf{d}}_s^{\top}\overline{\mathbf{y}} + \sum_{j=1}^{k}\sum_{a\in A_s} p(s^j|s,a)q(a)\tilde{v}_{t+1}^{\lambda}(s^j) \\ \text{Subject to: } & C_s^{\top}\mathbf{y} = \mathbf{q}, \\ & \overline{C}_s^{\top}\overline{\mathbf{y}} = \mathbf{q}, \\ & \mathbf{1}^{\top}\mathbf{q} = 1, \\ & \mathbf{q}, \mathbf{y} \geq 0. \end{aligned} \tag{3.11}$$

Observe that the feasible set is the same for all λ. By substituting $\tilde{v}_{t+1}^{\lambda}(s^j)$ and rearranging, it follows that for $\lambda \in [\lambda_{i-1}, \lambda_i]$ the objective function equals to

$$(1-\lambda)\Big\{\sum_{a\in A_s}\big[\sum_{j=1}^{k} p(s^j|s,a)m_i^j\big]q(a) + \mathbf{d}_s^{\top}\mathbf{y}\Big\} + \lambda\Big\{\sum_{a\in A_s}\big[\sum_{j=1}^{k} p(s^j|s,a)(l_i^j + m_i^j)\big]q(a) + \overline{\mathbf{d}}_s^{\top}\overline{\mathbf{y}}\Big\}. \tag{3.12}$$

Thus, for $\lambda \in [\lambda_{i-1}, \lambda_i]$, from the optimal solution for λ_{i-1}, we can solve for all λ using Algorithm 3.1 by converting it to a PLP. Furthermore, we need not re-initiate for each interval, because the optimal solution for the end of i^{th} interval is optimal for λ_i, and hence we can solve for $\lambda \in [\lambda_i, \lambda_{i+1}]$ by simply substituting l_{i+1} and m_{i+1} into the Equation (3.12) and run Algorithm 3.1 from this solution. Observe that the resulting $\tilde{v}_t^{\lambda}(s)$ is also continuous, piecewise linear. Thus, since $\tilde{v}_T^{\lambda}(\cdot) = 0$, the assumption that the value functions are continuous and piecewise linear holds by backward induction.

Here we outline of the algorithm finding all S-robust actions for a state s. Define

$$\begin{aligned} U_i(\mathbf{q}, \overline{\mathbf{y}}, \mathbf{y}) &\triangleq -\sum_{a\in A_s}\big[\sum_{j=1}^{k} p(s^j|s,a)(l_i^j + m_i^j)\big]q(a) + \overline{\mathbf{d}}_s^{\top}\overline{\mathbf{y}}, \\ V_i(\mathbf{q}, \overline{\mathbf{y}}, \mathbf{y}) &\triangleq -\sum_{a\in A_s}\Big[\sum_{j=1}^{k} p(s^j|s,a)m_i^j\Big]q(a) + \mathbf{d}_s^{\top}\mathbf{y}, \end{aligned}$$

and let $\mathcal{F}$ denote the feasible set of Problem (3.11). For the i^{th} interval, Problem (3.11) is thus minimizing $\lambda U_i(\mathbf{q}, \overline{\mathbf{y}}, \mathbf{y}) + (1-\lambda)V_i(\mathbf{q}, \overline{\mathbf{y}}, \mathbf{y})$ in $\mathcal{F}$.

ALGORITHM 3.2.

- Input and Initialization:
  (1) Minimize $V_1(\mathbf{q}, \overline{\mathbf{y}}, \mathbf{y})$ in $\mathcal{F}$, denote the optimal value as V^*.
  (2) Minimize $U_1(\mathbf{q}, \overline{\mathbf{y}}, \mathbf{y})$ in $\mathcal{F}$ with the constraint $V_1(\mathbf{q}, \overline{\mathbf{y}}, \mathbf{y}) = V^*$. Set the optimal basic feasible solution to be $(\mathbf{q}^*, \overline{\mathbf{y}}^*, \mathbf{y}^*)$. If multiple solutions exist, arbitrarily choose one.
  (3) $i := 1$ and $\lambda := 0$.
- Iteration
  (1) Calculate the coefficients of V_i and U_i.
  (2) Calculate the reduced costs of the objective function V_i and U_i at $(\mathbf{q}, \overline{\mathbf{y}}, \mathbf{y})$ for all its non-basic variables. If all non-basic variables of $(\mathbf{q}, \overline{\mathbf{y}}, \mathbf{y})$ have nonnegative reduced cost of U_i, go to 5.
  (3) Among all non-basic variables with negative reduced cost of U_i, choose the one with the largest absolute value of the reduced cost of U_i divided by that of V_i. (We call this value the *ratio*.) Add this variable into the base and denote the new basic feasible solution as $(\mathbf{q}^{\text{new}}, \overline{\mathbf{y}}^{\text{new}}, \mathbf{y}^{\text{new}})$. If the *ratio* is smaller than $(1 - \lambda_i)/\lambda_i$, go to 5.
  (4) $(\mathbf{q}, \overline{\mathbf{y}}, \mathbf{y}) := (\mathbf{q}^{\text{new}}, \overline{\mathbf{y}}^{\text{new}}, \mathbf{y}^{\text{new}})$, go to 2.
  (5) If $i = n$, terminate. Otherwise $i := i + 1$, go to 1.

We briefly explain the iteration of the algorithm. It contains two loops. Steps (2) to (4) are the operations in one interval, which is a standard PLP algorithm. If the *ratio* falls below $(1 - \lambda_i)/\lambda_i$ or U_i can no longer be decreased, the current interval ends, and we use the current solution as the start point of the next interval. The number of iterates equals to the number of pieces of $\tilde{v}_t^{\lambda}(s)$, and hence the algorithm is guaranteed to terminate in finite steps.

Next we investigate the discounted-reward infinite-horizon case. Algorithm 3.2 essentially performs the $\mathcal{L}$ operator. Hence we can approximate the whole set of S-robust strategies by applying Algorithm 3.2 repeatedly. An alternative way that

solves the *exact* S-robust strategy for *all* λ of an infinite-horizon UMDP is converting the MDP into its linear programming form and applying PLP.

THEOREM 3.8. *Suppose Assumptions 3.1, 3.3 and 3.4 hold. Then given initial distribution $\alpha(s)$, the L/W tradeoff strategy to a γ discounted-reward infinite-horizon UMDP is the following LP*

$$\begin{aligned}
&\textit{Maximize:}\lambda \sum_{s\in S}\left[\overline{\mathbf{d}}_s^{\top}\overline{\mathbf{y}}_s\right] + (1-\lambda)\sum_{s\in S}\left[\mathbf{d}_s^{\top}\mathbf{y}_s\right] \\
&\textit{Subject to:} \sum_{a\in A_{s'}} x(s',a) - \sum_{s\in S}\sum_{a\in A_s}\gamma p(s'|s,a)x(s,a) = \alpha(s'), \quad \forall s', \\
&\qquad C_s^{\top}\mathbf{y}_s = \mathbf{x}_s \quad \forall s, \\
&\qquad \overline{C}_s^{\top}\overline{\mathbf{y}}_s = \mathbf{x}_s \quad \forall s, \\
&\qquad \mathbf{y}_s, \overline{\mathbf{y}}_s \geq \mathbf{0}, \quad \forall s, \\
&\qquad x(s,a) \geq 0, \quad \forall s, \forall a,
\end{aligned} \tag{3.13}$$

with the optimal policy at state s given by $q_s(a) \triangleq x(s,a)/\sum_{a'\in A_s} x(s,a')$ and the denominator is guaranteed to be nonzero.

PROOF. Fix λ, the maxmin problem of the L/W tradeoff strategy is essentially a zero-sum game, with the decision maker trying to maximize the weighted sum and *Nature* trying to minimize it by selecting $\overline{\mathbf{r}}$ and $\mathbf{r}$ adversely. Furthermore, the dynamics of the game (i.e., the state transition) is determined only by the decision maker. A well known result in discounted zero-sum stochastic games states that in this case even if non-stationary policies are admissible, a Nash equilibrium in which both players choose a stationary policy exists; see Bertsekas and Tsitsiklis [**16**, Proposition 7.3].

For initial state distribution $\alpha(s)$, recall that there exists a one-to-one correspondence relation between the following polytope $\mathcal{X}$ and the state-action frequencies $\mathbb{E}\sum_{i=1}^{\infty}\gamma^{i-1}(\mathbf{I}(s_i = s, a_i = a))$ for stationary strategies; see Puterman [**124**, Theorem

6.9.1].

$$\mathcal{X}: \quad \sum_{a\in A_{s'}} x(s',a) - \sum_{s\in S}\sum_{a\in A_s} \gamma p(s'|s,a)x(s,a) = \alpha(s'),\ \forall s'$$
$$x(s,a) \geq 0,\ \forall s, \forall a \in A_s. \tag{3.14}$$

Since it suffices to only consider stationary minimax policies, the L/W tradeoff (and equivalently S-robust) problem is:

$$\begin{aligned} &\text{Maximize: } \inf_{\overline{\mathbf{r}}\in\overline{\mathcal{R}},\mathbf{r}\in\mathcal{R}} \sum_{s\in S}\sum_{a\in A_s} [\lambda\overline{r}(s,a)x(s,a) + (1-\lambda)r(s,a)x(s,a)] \\ &\text{Subject to: } \mathbf{x}\in\mathcal{X}. \end{aligned} \tag{3.15}$$

By duality of LP, Problem (3.15) is equivalent to Problem (3.13). Refer to Puterman [**124**] for the conversion from $\mathbf{x}$ to q and the denominator being nonzero. □

Note that Problem (3.13) is a PLP, which can be solved using Algorithm 3.1 from Appendix 3.2.2.

3.5. A numerical example

In this section, we apply the algorithm of UMDP with known dynamics to a T-stage machine maintenance problem, and compare the tradeoff solutions with the nominal solution and the robust solution. Notice that we abuse notations in this section.

Let $S \triangleq \{1, \cdots, n\}$ denote the state set for each stage, which represents the condition of a machine. In state h, the decision maker can choose either to replace the machine which will lead to state 1 deterministically, or to continue running, which has a probability p leading to state $h+1$. If the machine is in state n, the decision maker has to replace it. The replacing cost is exactly known to be c_r. The inner set of the running cost in state h is a singleton $\{c_h\}$ (thus the likely performance is indeed the nominal performance), while the deviation set is $[c_h - \sigma_h, c_h + \sigma_h]$, with c_h and σ_h increasing with h. We set $T = 20$, $n = 7$. $p = 0.9$, $c_h = \sqrt{h} - 1$ and $\sigma_h = 2h/n$. Figure 3.2 shows the Likely/Worst-case tradeoffs of this UMDP as

computed by Algorithm 3.2. Note that each dot point is a *deterministic* L/W tradeoff strategy.

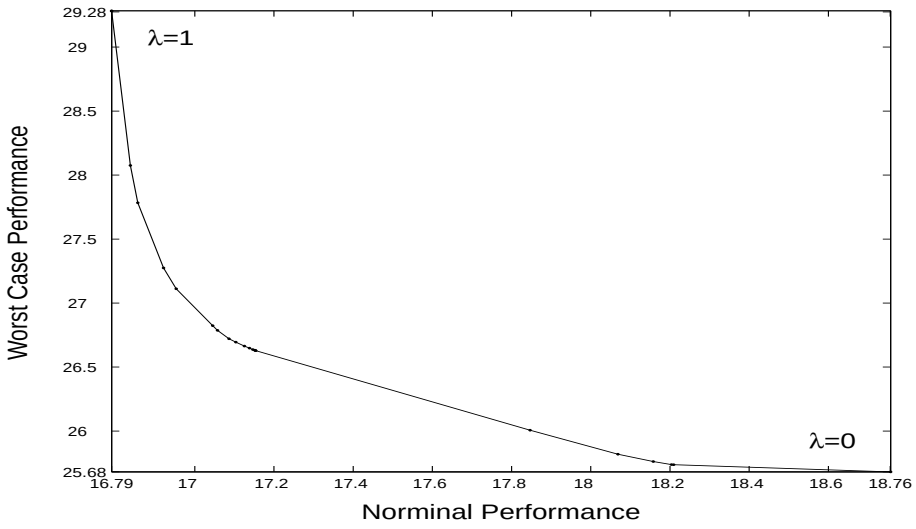


FIGURE 3.2. The Likely/Worst-case tradeoffs of the machine maintenance example.

Next we ran Monte-Carlo simulations to compare the solutions we get. In the simulation, we used a pre-defined parameter δ to control how adversarial the parameter realization is. To be more specific, the running cost in state h is generated according to the following probability distribution

$$f(x) = \begin{cases} \dfrac{\delta}{\sigma_h}, & x \in [c_h,\, c_h + \sigma_h], \\ \dfrac{1-\delta}{\sigma_h}, & x \in [c_h - \sigma_h,\, c_h), \\ 0, & \text{otherwise.} \end{cases}$$

Note that δ determines the probability that the true cost lies in the "bad" half of the uncertainty set. The value $\delta = 0.5$ means that the nominal parameter is the true mean value of the cost. And the larger the δ, the more overly-optimistic the nominal cost is. For $\delta = 0.5, 0.7, 1.0$, we generated 300 parameter realizations each. For each realization, we ran 300 tests and take their average reward as the cost for one simulation, to cancel out the internal variance (i.e., the performance variation of different runs under a same parameter realization due to the stochastic nature of the

system). We normalized the cost for each simulation, i.e., we divided the cost by the smallest expected nominal cost.

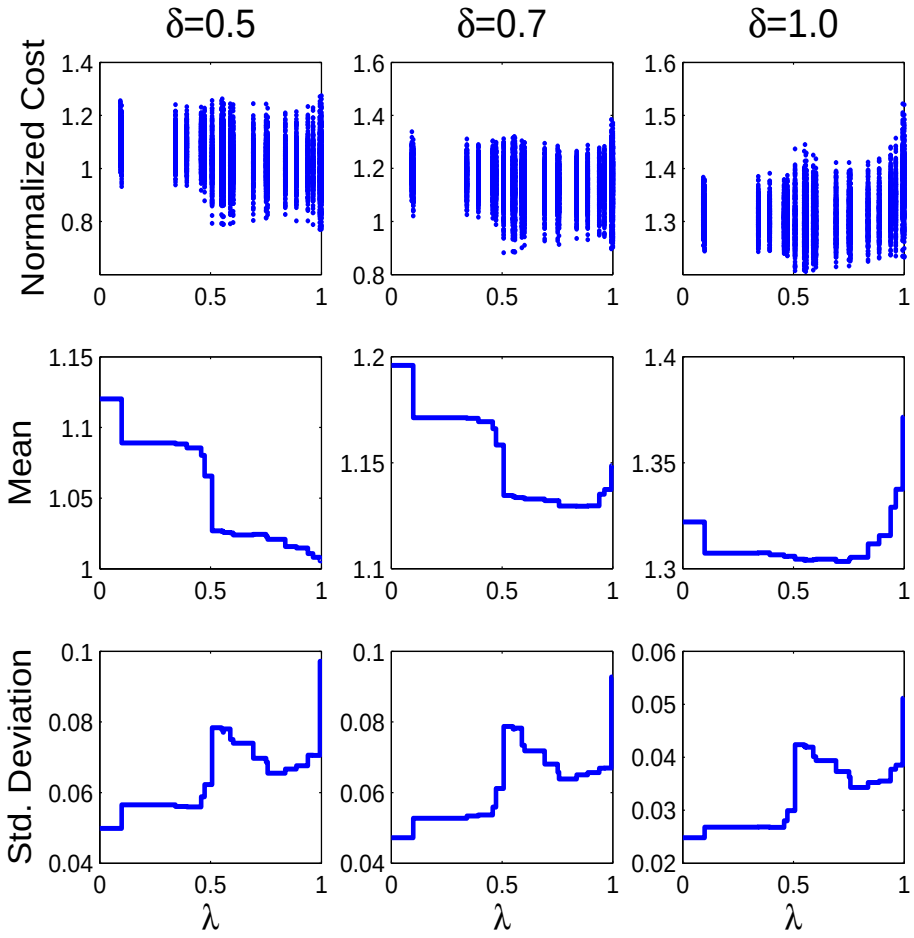


FIGURE 3.3. Simulation results of the machine maintenance problem for different δ.

The simulation results are shown in Figure 3.3. The nominal solution, i.e., solution at $\lambda = 1$ that neglecting the deviation set completely, achieves minimal mean cost at $\delta = 0.5$. However, it has a wide spread in the scatter plot, a sharp increase in the standard deviation, and a severe deterioration of the mean performance when δ increases, all of which show that it is sensitive to parameter uncertainty. On the other hand, the robust solution is overly conservative, even in the case when $\delta = 1$. In contrast to these two extreme solutions, the strategies in the middle range of λ

achieve much better tradeoffs between the nominal performance and worst-case performance. We also observe that by slightly relaxing the performance requirement (say, take $\lambda = 0.9$) we will get much better robustness to parameter uncertainty, with a marginal decrease in the mean performance.

3.6. Chapter summary

In this chapter we addressed MDPs under parameter uncertainty following the axiomatic MMEU framework. In particular, we considered the nested-set structured parameter uncertainty to model the prior information of both the likely value and the possible deviation of the parameters. We proposed to find a strategy that achieves maximum expected utility under the worst possible distribution of the parameters. Such formulation leads to an optimal strategy that is obtained through a Bellman type backward induction, and can be solved in polynomial time under mild technical conditions.

We further investigated a special case where the transition probabilities are precisely known. In such case, the MMEU strategy has an intuitively appealing interpretation as finding the Pareto efficient trade-offs of the "likely" performance and the downside-deviation protection. If the uncertainty sets are further assumed to be polyhedral, we provided a PLP based algorithm that finds the *whole* set of MMEU strategies. Thus, the decision maker can choose the strategy that achieves the most desirable tradeoff according to his/her subjective preference, by observing all possible tradeoffs. This is in contrast to the standard method where the decision maker has to guess a tradeoff parameter beforehand.

The main thrust of this chapter is to mitigate the conservatism of the robust MDP framework by incorporating additional prior information regarding the unknown parameters, and to allow flexible decision making that achieves both good performance under the normal case and reasonably robustness to possible deviations. The proposed formulation can be easily generalized to more complicated uncertainty structures, provided that the parameters of different states are independent.

3.7. Proof of Theorem 3.1

PROOF. We drop the superscript λ in the proof whenever it is clear. Let h_t denote a history up to stage t and $s(h_t)$ denote the last state of history h_t. We use $\pi_{h_t}(a)$ to represent the probability of choosing an action a at the state $s(h_t)$, following a strategy π and under a history h_t. A $t+1$ stage history, with h_t followed by action a and state s' is written as (h_t, a, s').

With an abuse of notation, we denote the expected reward-to-go under a history as:

$$u(\pi, \mathbf{p}, \mathbf{r}, h_t) \triangleq \mathbb{E}_\pi^{\mathbf{p}}\{\sum_{i=t}^{T} r(s_i, a_i) | (s_1, a_1 \cdots, s_t) = h_t\}.$$

For $\pi \in \Pi^{HR}$ and $\mu \in \mathcal{C}_S(\lambda)$, define

$$w(\pi, \mu, h_t) \triangleq \mathbb{E}_{(\mathbf{p},\mathbf{r})\sim\mu} u_s(\pi, \mathbf{p}, \mathbf{r}, h(t)) = \int u(\pi, \mathbf{p}, \mathbf{r}, h(t)) d\mu(\mathbf{p}, \mathbf{r}).$$

Note that the MMEU strategy is thus

$$\pi_M^* = \arg\max_{\pi\in\Pi^{HR}} \min_{\mu\in\mathcal{C}_S(\lambda)} w(\pi, \mu, h_1); \quad \text{where } h_1 = (s^{\text{ini}}).$$

We first establish the following two lemmas.

LEMMA 3.9. *The following updating rule holds for any h_t where $t < T$, $\pi \in \Pi^{HR}$ and $\mu \in C_S(\lambda)$:*

$$\begin{aligned} w(\pi, \mu, h_t) = \int \sum_{a\in A_{s(h_t)}} \pi_{h_t}(a) \Big[& r\big(s(h_t), a\big) \\ & + \sum_{s'\in S} p\big(s' | s(h_t), a\big) w\big(\pi, \mu, (h_t, a, s')\big) \Big] d\mu_{s(h_t)}(\mathbf{p}_{s(h_t)}, \mathbf{r}_{s(h_t)}). \end{aligned} \tag{3.16}$$

PROOF. By definition $\mu(\mathbf{p}, \mathbf{r}) \in \mathcal{C}_S(\lambda)$ implies $\mu(\mathbf{p}, \mathbf{r}) = \prod_{s\in S} \mu_s(\mathbf{p}_s, \mathbf{r}_s)$ while $\mu_s(\mathbf{p}_s, \mathbf{r}_s) \in \mathcal{C}_s(\lambda)$. Since it is clear on what variable the distribution is, we will simply write μ and μ_s. Denote $\mu(t) = \prod_{s\in\bigcup_{i=t}^{T} S_i} \mu_s$, that is, the probability distribution for

the parameters from stage t on. We thus have:

$$\begin{aligned} w(\pi,\mu,h_t) &= \mathbb{E}_{(\mathbf{p},\mathbf{r})\sim\mu} u(\pi,\mathbf{p},\mathbf{r},h_t) = \int u(\pi,\mathbf{p},\mathbf{r},h_t)d\mu \\ &= \int u(\pi,\mathbf{p},\mathbf{r},h_t)d\mu(t) = \int\int u(\pi,\mathbf{p},\mathbf{r},h_t)d\mu(t+1)d\mu_{s(h_t)}, \end{aligned} \tag{3.17}$$

due to the fact that $u\big(\pi,\mathbf{p},\mathbf{r},h(t)\big)$ only depends on the parameters from the t^{th} stage on. Notice that for a fixed parameter realization and a fixed strategy, the Bellman equation holds. That is,

$$u(\pi,\mathbf{p},\mathbf{r},h_t) = \sum_{a\in A_{s(h_t)}} \pi_{h_t}(a)\Big(r\big(s(h_t),a\big) + \sum_{s'\in S_{t+1}} p\big(s'|s(h_t),a\big)u\big(\pi,\mathbf{p},\mathbf{r},(h_t,a,s')\big)\Big).$$

Thus the right-hand-side of Equation (3.17) equals

$$\begin{aligned} &\int\int\Big\{\sum_{a\in A_{s(h_t)}} \pi_{h_t}(a)\Big[r\big(s(h_t),a\big) \\ &\qquad + \sum_{s'\in S_{t+1}} p\big(s'|s(h_t),a\big)u\big(\pi,\mathbf{p},\mathbf{r},(h_t,a,s')\big)\Big]\Big\}d\mu(t+1)d\mu_{s(h_t)} \\ =&\int\sum_{a\in A_{s(h_t)}} \pi_{h_t}(a)\Big[r\big(s(h_t),a\big) \\ &\qquad + \sum_{s'\in S_{t+1}} p\big(s'|s(h_t),a\big)\int u\big(\pi,\mathbf{p},\mathbf{r},(h_t,a,s')\big)d\mu(t+1)\Big]d\mu_{s(h_t)} \\ =&\int\sum_{a\in A_{s(h_t)}} \pi_{h_t}(a)\Big(r\big(s(h_t),a\big) + \sum_{s'\in S} p\big(s'|s(h_t),a\big)w\big(\pi,\mu,(h_t,a,s')\big)\Big)d\mu_{s(h_t)}, \end{aligned}$$

which proves the lemma. Here the first equality holds because $\mu(t+1)$ by definition is the probability distribution of parameters from stage $t+1$ on, and $s(h_t)$ belongs to stage t. □

LEMMA 3.10. *For $s\in S_t$ where $t<T$, there exists $\mu_s^*\in\mathcal{C}_s(\lambda)$ such that*

$$\begin{aligned} &(i)\quad \max_{\pi_s\in\Delta_s} v_s(\pi_s,\mu_s^*) = v_s(\pi_s^*,\mu_s^*) = \min_{\mu_s\in\mathcal{C}_s(\lambda)} v_s(\pi_s^*,\mu_s); \\ &(ii)\quad v_s(\pi_s^*,\mu_s^*) = \tilde{v}_t(s), \end{aligned} \tag{3.18}$$

holds for any S-robust action $\pi_s^* \in \Delta(s)$. *Here,* $v_s(\pi_s, \mu_s) \triangleq \mathbb{E}_{(\mathbf{p}_s, \mathbf{r}_s)\sim\mu_s}\Big\{\mathbb{E}_{\pi_s}^{\mathbf{p}_s}[r(s,a) + \tilde{v}_{t+1}(\underline{s})]\Big\}$.

PROOF. First consider the following zero-sum game.

Game 1: For $s \in S_t$, one player chooses $\mathbf{q}_s \in \Delta(s)$; the other one chooses $(\overline{\mathbf{p}}_s, \overline{\mathbf{r}}_s, \mathbf{p}_s, \mathbf{r}_s) \in \overline{\mathcal{P}}_s \times \overline{\mathcal{R}}_s \times \mathcal{P}_s \times \mathcal{R}_s$; the utility function for player one is

$$\begin{aligned}
&\hat{v}_s(\mathbf{q}_s, \mathbf{p}_s, \mathbf{r}_s) \triangleq \lambda \mathbb{E}_{\pi_s}^{\overline{\mathbf{p}}_s}[\overline{r}(s,a) + \tilde{v}_{t+1}(\underline{s})] + (1-\lambda)\mathbb{E}_{\pi_s}^{\mathbf{p}_s}[r(s,a) + \tilde{v}_{t+1}(\underline{s})] \\
=&\lambda \sum_{a\in A_s} q_s(a)\big[\overline{r}(s,a) + \sum_{s'\in S_{t+1}} \overline{p}(s'|s,a)\tilde{v}_{t+1}(s')\big] \\
&+ (1-\lambda)\sum_{a\in A_s} q_s(a)\big[r(s,a) + \sum_{s'\in S_{t+1}} p(s'|s,a)\tilde{v}_{t+1}(s')\big].
\end{aligned}$$

This game has a Nash equilibrium $\big(\mathbf{q}_s^*, (\overline{\mathbf{p}}_s^*, \overline{\mathbf{r}}_s^*, \mathbf{p}_s^*, \mathbf{r}_s^*)\big)$, because the strategy domains for both players are compact and convex, and the utility function is linear to both players. Observe that $\mathbf{q}_s^*$ can be any S-robust action by definition. Now let $\pi_s^* = \mathbf{q}_s^*$ and μ_s^* be such that $\mu_s^*(\mathbf{p}_s^*, \mathbf{r}_s^*) = 1-\lambda$ and $\mu_s^*(\overline{\mathbf{p}}_s^*, \overline{\mathbf{r}}_s^*) = \lambda$. We have $\mu_s^* \in \mathcal{C}_s(\lambda)$. It is easy to check that (π_s^*, μ_s^*) satisfy Equation (3.18). □

To prove the theorem, it suffices to show that given a S-robust strategy π^*, we have $\mu^* = \prod_{s\in S}\mu_s^*$ where μ_s^* is given by Lemma 3.10 such that following three statements hold $\forall h_t,\ t = 1, \cdots, T$:

$$\begin{aligned}
&\text{(i)} \quad w(\pi^*, \mu^*, h_t) = \tilde{v}_t\big(s(h_t)\big); \\
&\text{(ii)} \quad w(\pi^*, \mu, h_t) \geq \tilde{v}_t\big(s(h_t)\big), \quad \forall \mu \in \mathcal{C}_S(\lambda); \\
&\text{(iii)} \quad w(\pi, \mu^*, h_t) \leq \tilde{v}_t\big(s(h_t)\big), \quad \forall \pi \in \Pi^{HR}.
\end{aligned} \tag{3.19}$$

We prove this by backward induction. Suppose for all h_{t+1}, (3.19) holds. Now we show they still hold for any h_t.

Condition (i):

$$
\begin{aligned}
&w(\pi^*, \mu^*, h_t) \\
=&\int \sum_{a \in A_{s(h_t)}} \pi^*_{h_t}(a) \Big[r\big(s(h_t), a\big) + \sum_{s' \in S_{t+1}} p\big(s' | s(h_t), a\big) w\big(\pi^*, \mu^*, (h_t, a, s')\big) \Big] d\mu^*_{s(h_t)} \\
=&\int \sum_{a \in A_{s(h_t)}} \pi^*_{s(h_t)}(a) \Big[r\big(s(h_t), a\big) + \sum_{s' \in S_{t+1}} p\big(s' | s(h_t), a\big) \tilde{v}_{t+1}(s') \Big] d\mu^*_{s(h_t)} \\
=&v_{s(h_t)}(\pi^*_{s(h_t)}, \mu^*_{s(h_t)}) = \tilde{v}_t(s(h_t)).
\end{aligned}
$$

The second equality holds by Condition (i) of the history (h_t, a, s'). The last equality holds by Lemma 3.10.

Condition (ii):

$$
\begin{aligned}
&w(\pi^*, \mu, h_t) \\
=&\int \sum_{a \in A_{s(h_t)}} \pi^*_{h_t}(a) \Big[r\big(s(h_t), a\big) + \sum_{s' \in S_{t+1}} p\big(s' | s(h_t), a\big) w\big(\pi^*, \mu, (h_t, a, s')\big) \Big] d\mu_{s(h_t)} \\
\geq&\int \sum_{a \in A_{s(h_t)}} \pi^*_{s(h_t)}(a) \Big[r\big(s(h_t), a\big) + \sum_{s' \in S} p\big(s' | s(h_t), a\big) \tilde{v}_{t+1}(s') \Big) d\mu_s(h_t) \\
\geq&\min_{\mu' \in \mathcal{C}_{s(h_t)}(\lambda)} \int \sum_{a \in A_{s(h_t)}} \pi^*_{s(h_t)}(a) \Big(r\big(s(h_t), a\big) + \sum_{s' \in S_{t+1}} p\big(s' | s(h_t), a\big) \tilde{v}_{t+1}(s') \Big) d\mu' \\
=&\min_{\mu' \in \mathcal{C}_{s(h_t)}(\lambda)} v_{s(h_t)}(\pi^*_{s(h_t)}, \mu') = \tilde{v}_t\big(s(h_t)\big).
\end{aligned}
$$

The first inequality holds by Condition (ii) of the history (h_t, a, s). The second inequality holds because $\mu \in \mathcal{C}_S(\lambda)$ which leads to $\mu_{s(h_t)} \in \mathcal{C}_{s(h_t)}(\lambda)$. The last equality holds by Lemma 3.10.

Condition (iii):

$$
\begin{aligned}
& w(\pi, \mu^*, h_t) \\
=& \int \sum_{a \in A_{s(h_t)}} \pi_{h_t}(a) \Big(r\big(s(h_t), a\big) + \sum_{s' \in S_{t+1}} p\big(s' | s(h_t), a\big) w\big(\pi, \mu^*, (h_t, a, s')\big) \Big) d\mu_s^* \\
\leq& \int \sum_{a \in A_{s(h_t)}} \pi_{h_t}(a) \Big(r\big(s(h_t), a\big) + \sum_{s' \in S_{t+1}} p\big(s' | s(h_t), a\big) \tilde{v}_{t+1}(s') \Big) d\mu_s^* \\
\leq& \max_{\pi' \in \Delta(s(h_t))} \int \sum_{a \in A_{s(h_t)}} \pi'(a) \Big(r\big(s(h_t), a\big) + \sum_{s' \in S_{t+1}} p\big(s' | s(h_t), a\big) \tilde{v}_{t+1}(s') \Big) d\mu_s^* \\
=& \max_{\pi' \in \Delta(s(h_t))} v_{s(h_t)}(\pi', \mu^*_{s(h_t)}) = \tilde{v}_{t+1}\big(s(h_t)\big).
\end{aligned}
$$

The first inequality holds because of the third condition for the history (h_t, a, s). The second inequality holds because $\pi_{h_t} \in \Delta(s(h_t))$ by definition. The last equality holds by Lemma 3.10.

Observe that (3.19) holds trivially for $t = T$. This complete the backward induction. Substituting $h_1 = (s^{\text{ini}})$ into (3.19) establishes the theorem. Notice that (3.19) indeed means that (π^*, μ^*) is a Nash Equilibrium. □

CHAPTER 4

Parametric Regret in Uncertain Markov Decision Processes

In Chapter 3 we discussed uncertain Markov decision processes where each strategy is evaluated by its performance, i.e., accumulated reward-to-go. In contrast to this standard setup, there are situations in which a strategy is evaluated comparatively. That is, the decision maker is concerned about how the performance of a strategy compares with other strategies. In this chapter, we investigate robust decision making in such a setup. In particular, we consider uncertain Markov decision processes where the performance criterion is the gap between the performance of the best strategy that is chosen after the true parameter realization is revealed and the performance of the strategy that is chosen before the parameter realization is revealed. We call this gap the *parametric regret*. We consider two related problems: minimax regret and mean-variance tradeoff of the regret. A minimax regret strategy minimizes the worst-case regret under the most adversarial possible realization. We show that the problem of computing a minimax regret strategy is NP-hard and propose algorithms to efficiently finding it under favorable conditions. The mean-variance tradeoff formulation requires a probabilistic model of the uncertain parameters and looks for a strategy that minimizes a convex combination of the mean and the variance of the

regret. We prove that computing such a strategy can be done numerically in an efficient way. Part of the material in this chapter appears in [**174**].

4.1. Introduction

Sequential decision making in stochastic dynamic environments is often modeled using Markov Decision Processes (MDP, cf [**124, 16**]). In the *standard setup*, each strategy is evaluated according to its *performance*, i.e., the expected accumulated reward. An optimal strategy is one that achieves maximal performance.

In many real applications, the decision maker evaluates strategies in a comparative way. That is, given a strategy, the decision maker is interested in how its performance competes with other strategies rather than the *value* of the performance itself. For example, the objective in financial applications such as portfolio optimizations is often to "beat the market", i.e., to perform favorably compared to a strategy that holds index stocks. The same percentage of growth can be regarded as "incredible success" or "disastrous failure" purely depending on how others perform in this same market. A natural measurement of strategies in such setup, which we termed *competitive setup* hereafter, is the so-called *parametric regret*: the gap between the performance of a strategy and that of the optimal one. [1]

When the parameters of a MDP are known, minimizing the regret is equivalent to maximizing the performance of a strategy, and hence the competitive setup coincides with the standard setup. However, the formulation of a problem is often subject to *parameter uncertainty* – the deviation of the modeling parameters from the unknown true ones (cf [**116, 5, 162, 91**]). In this case, both performance and regret of a strategy are functions of parameter realizations, where in general there is no strategy that is optimal for all parameter realizations.

In the standard setup, there are two formulations to find the "optimal" strategy for MDPs with uncertain parameters. The first formulation[**116, 5, 162, 91**] takes

[1]We will use "regret" in the following for simplicity of the expression. However, it should be noted that this is different from the standard notion of regret in online learning - the gap between the average reward of a learning algorithm and the optimal strategy [**86**].

a minimax approach, i.e., the true parameters can be any element of a known set, and strategies are evaluated based on the performance under the (respectively) worst possible parameter realization. The second one takes a Bayesian approach (e.g. [**50**]): The true parameters are regarded as random variables. Thus, given a strategy, its performance is a random variable whose probability distribution can be obtained. And an optimal strategy is one that maximizes certain risk measures such as percentile loss [**50**] or mean-variance tradeoff [**102**].

In this chapter we adapt the aforementioned formulations into the competitive setup and discuss parametric regret minimizing in uncertain Markov decision processes. In particular, our contributions include the following:

- In Section 4.2 we follow the minimax approach and propose the Minimal Maximum Regret (MMR) decision criterion.
- We show in Section 4.3 that finding the MMR strategy is NP-hard in general.
- We investigate the algorithmic aspect of MMR strategy in Section 4.4. In particular, we propose in Section 4.4.1 an algorithm based on mixed integer programming that solves for the MMR strategy, and discuss in the rest of Section 4.4 two special cases where the MMR strategy can be found in polynomial time.
- We take the Bayesian approach and propose the Optimal Mean-Variance Tradeoff of Regret criterion in Section 4.5. We further show that such a formulation can be converted into a quadratic program on a polytope, and hence solved efficiently.

We need to point out that in this paper we concentrate on the case where the system dynamics are known and only reward parameters are subject to uncertainty, partly due to the prohibitive computational cost. Indeed, as shown in Section 4.3, even in this seemingly simple case finding the MMR strategy is NP-hard. In addition, the known-dynamics case can either model or approximate many practical problems. For instance, a shortest-path problem with uncertain link lengths is an uncertain MDP with known dynamics (e.g., [**124**]). Another example is using state aggregation

to solve large scale MDPs [**138**]. In such case, states are grouped to a small number of hyper-states and a reduced MDP built on these hyper-states is investigated. Typically, the transition law between hyper-states are known, but the expected reward visiting each hyper-state is uncertain due to the transitions inside each hyper-state.

4.1.1. Preliminaries and notations. Throughout this chapter, boldface letters are used for column vectors, where its elements are represented using the same but non-boldfaced letter. For example, the first element of a vector $\mathbf{v}$ is denoted as v_1. Given a function $f(\mathbf{x})$ not necessarily differentiable, we use $\nabla f(\mathbf{x})|_{\mathbf{x}_0}$ to represent the set of subgradients at point $\mathbf{x}_0$.

An uncertain Markov Decision Process (*uMDP*) is a 6-tuple $< T, \gamma, S, A, \mathbf{p}, \mathcal{R} >$ where:

- T is the (possibly infinite) decision horizon;
- $\gamma \in (0, 1]$ is the discount factor. We allow $\gamma = 1$ only when T is finite.
- S is the state set and A is the action set. Both sets are finite. As standard, in the finite-horizon case, a state that appears in multiple stages is treated as different states.
- $\mathbf{p}$ is the transition probability i.e., $p(s'|s, a)$ is the probability to reach state s' from a state s when action a is taken.
- $\mathcal{R}$ is the admissible set of reward parameter. To be more specific, the reward vector $\mathbf{r}$ is unknown to the decision maker (this is why it is called "uncertain MDP"). To make such a decision problem meaningful, some a priori information of $\mathbf{r}$ is known: it is an element of $\mathcal{R}$. In the literature of *robust optimization*, $\mathcal{R}$ is often called the *uncertainty set* (cf [**141, 12, 116**]). Since we mainly consider the planning problem, the decision maker is not allowed to adapt/learn her strategy according to different parameter realizations.

We assume that the initial state distribution is known to be $\boldsymbol{\alpha}$. All history-dependent randomized strategies are admissible, and we denote that set by Π^{HR}. We use Π^S and Π^D to denote the set of stationary Markovian random strategies and stationary Markovian deterministic strategies, respectively. For $\pi \in \Pi^S$, we use $\pi(a|s)$ to

represent the probability of choosing $a \in A$ at state s following π. Given a strategy $\pi \in \Pi^{HR}$ and a parameter realization $\mathbf{r} \in \mathcal{R}$, its expected performance (i.e., accumulated discounted reward) is denoted by $P(\pi, \mathbf{r})$, that is

$$P(\pi, \mathbf{r}) \triangleq \mathbb{E}_\pi\{\sum_{i=1}^{T} \gamma^{i-1} r(s_i, a_i)\}. \tag{4.1}$$

We focus on the case where the uncertainty set $\mathcal{R}$ is a polytope. Polytopes are probably the most "natural" formulation of an uncertainty set that can model many widely applicable cases. For example, the interval case, i.e., each reward parameter $r(s, a)$ belongs to an interval, is a polytope. We also assume that $\mathcal{R}$ is bounded, to avoid technical problems such as infinitely large regret.

4.2. MiniMax regret in MDPs

In this section we propose the MiniMax Regret criterion, i.e., minimizing the parametric regret under the most adversarial parameter realization.

DEFINITION 4.1. *Given a uMDP* $< T, \gamma, S, A, \mathbf{p}, \mathcal{R} >$ *and* $\mathbf{r}_0 \in \mathcal{R}$*, the* parametric regret *of a strategy* π *w.r.t.* $\mathbf{r}_0$ *is defined as*

$$\hat{R}(\pi, \mathbf{r}_0) \triangleq \max_{\pi' \in \Pi^{HR}} \{P(\pi', \mathbf{r}_0) - P(\pi, \mathbf{r}_0)\}.$$

In words, regret is the performance gap between a strategy and the optimal strategy. It is thus a natural performance measure in a competitive environment. Observe that for a fixed $\mathbf{r}_0$, the regret is equivalent to the expected reward up to adding a constant.

DEFINITION 4.2. *Given a uMDP* $< T, \gamma, S, A, \mathbf{p}, \mathcal{R} >$*, the* Maximum Regret *of a strategy* π *is defined as*

$$R(\pi) \triangleq \max_{\mathbf{r} \in \mathcal{R}} \hat{R}(\pi, \mathbf{r}) = \max_{\mathbf{r} \in \mathcal{R}, \pi' \in \Pi^{HR}} \{P(\pi', \mathbf{r}) - P(\pi, \mathbf{r})\}. \tag{4.2}$$

The maximum regret is the regret of a strategy under the most adversarial parameter realization. It can also be regarded as the performance gap w.r.t. an "all-mighty"

opponent strategy that can observe the parameter realization and select the respective optimal solution.

DEFINITION 4.3. *Given a uMDP* $< T, \gamma, S, A, \mathbf{p}, \mathcal{R} >$*, the* MiniMax Regret *(MMR) strategy is*

$$\pi^* \triangleq \arg \min_{\pi \in \Pi^{HR}} R(\pi). \tag{4.3}$$

The minimax regret strategy is not the same as the robust MDP (i.e., minimax performance) strategy in general, as shown in the following example: Consider the MDP as shown in Figure 4.1, where $\mathcal{R} = [0, 3] \times [1, 2]$. Observe that the minimax performance strategy is selecting $a2$, whose maximum regret equals 2. On the other hand, the minimax regret strategy is selecting either action with probability 50%, whose maximum regret is 1.

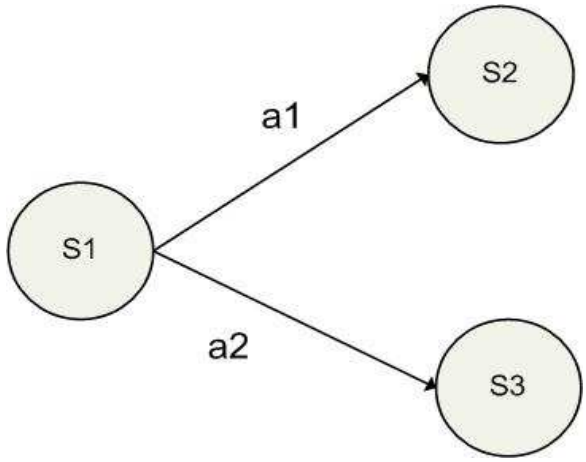


FIGURE 4.1. An example of MMR not equivalent to robust MDP.

4.2.1. Existence of stationary optimal solution. Although the definition of MMR considers history dependent strategies, in this subsection we show that without loss of generality we can concentrate on Π^S because there exists a stationary MMR strategy. We need the following lemma first.

LEMMA 4.1. *Given* $\pi_0 \in \Pi^{HR}$*, there exists* $\hat{\pi} \in \Pi^S$ *such that* $R(\hat{\pi}) = R(\pi_0)$.

PROOF. It is well known that (e.g., [**124**]) given $\pi_0 \in \Pi^{HR}$, there exists $\hat{\pi} \in \Pi^S$ such that $\forall s \in S, a \in A$

$$\mathbb{E}_{\pi_0} \sum_i \gamma^{i-1} \mathbf{1}(s_i = s, a_i = a) \equiv \mathbb{E}_{\hat{\pi}} \sum_i \gamma^{i-1} \mathbf{1}(s_i = s, a_i = a).$$

Note that the following holds for any $\pi \in \Pi^{HR}$,

$$P(\pi, \mathbf{r}) = \sum_{s \in S} \sum_{a \in A} \{r(s,a) \mathbb{E}_{\pi} \sum_i \gamma^{i-1} \mathbf{1}(s_i = s, a_i = a)\}.$$

Hence,

$$P(\pi', \mathbf{r}) - P(\pi_0, \mathbf{r}) = P(\pi', \mathbf{r}) - P(\hat{\pi}, \mathbf{r}), \ \forall \mathbf{r} \in \mathcal{R}, \pi' \in \Pi^{HR}.$$

Taking maximization over π' and $\mathbf{r}$ establishes the lemma. □

We now present the main theorem of this subsection: the existence of a stationary MMR strategy.

THEOREM 4.2. *There exists $\pi^* \in \Pi^S$ such that $R(\pi^*) \leq R(\pi), \ \forall \pi \in \Pi^{HR}$.*

PROOF. From Lemma 4.1, it suffices to prove that $R(\pi^*) \leq R(\pi), \ \forall \pi \in \Pi^S$. We define a metric $d(\pi_1, \pi_2) \triangleq \max_{s \in S, a \in A} |\pi_1(a|s) - \pi_2(a|s)|$ on Π^S and note that since S and A are finite, the set Π^S is compact. Let sequence $\{\pi_n\} \subseteq \Pi^S$ be such that $R(\pi_n) \to \inf_{\pi \in \Pi^{HR}} R(\pi)$. Due to compactness of Π^S, taking a convergent subsequence $\{\pi_{m_n}\}$ and let $\pi^* \in \Pi^S$ be its limiting point. Let

$$(\hat{\pi}', \hat{\mathbf{r}}) = \arg\max_{(\pi', \mathbf{r})} \{P(\pi', \mathbf{r}) - P(\pi^*, \mathbf{r})\}.$$

By definition of maximum regret we have

$$R(\pi_{m_n}) \geq P(\hat{\pi}', \hat{\mathbf{r}}) - P(\pi_{m_n}, \hat{\mathbf{r}}), \ \forall n.$$

Take limits on both sides and note that $P(\hat{\pi}', \hat{\mathbf{r}}) - P(\pi, \hat{\mathbf{r}})$ is a continuous function of π w.r.t. the aforementioned metric, we have

$$\inf_{\pi \in \Pi^{HR}} R(\pi) \geq R(\pi^*),$$

which establishes the theorem. □

4.3. Computational complexity

This section investigates the computational complexity of MMR strategy. We show that finding a MMR strategy is in general intractable. In fact, even evaluating the maximum regret for a given strategy can be NP-hard, as shown in the next theorem.

THEOREM 4.3. *Let $\mathcal{R}$ be a polytope defined by a set of n linear inequalities. Then deciding whether the maximum regret of a strategy is at least 1 is NP-complete with respect to $|S|$, $|A|$ and n.*

PROOF. We first show that deciding whether the maximum regret is at least 1 can not be computationally more difficult than NP. This is due to the fact that evaluating the regret of a given strategy $\hat{\pi}$ can be written as the following optimization problem on $(\mathbf{x}', \mathbf{r})$:

$$\begin{aligned} \text{max: } & \sum_{a\in A}\sum_{s\in S}\left\{r(s,a)x'(s,a) - r(s,a)\hat{x}(s,a)\right\} \\ \text{s.t. : } & \sum_{a\in A} x'(s',a) - \sum_{s\in S}\sum_{a\in A}\gamma p(s'|s,a)x'(s,a) = \alpha(s'),\ \forall s', \\ & x'(s,a) \geq 0, \quad \forall s, \forall a, \\ & \mathbf{r} \in \mathcal{R}. \end{aligned} \tag{4.4}$$

where $\hat{x}(s,a)$ is given by the $\sum_{i=1}^{T}\gamma^{i-1}\mathbb{E}(\mathbf{1}_{s_i=s,a_i=a})$ under $\hat{\pi}$. Note that Equation (4.4) is a (non-convex) quadratic program which is known to be equivalent to NP. Hence, deciding whether the maximum regret is at least 1 can not be computationally more difficult than NP.

Next we prove that deciding whether the maximum regret is at least 1 is NP-hard by showing that the *integer feasibility problem*, which is known to be NP-hard (e.g., [**117**]), can be reduced to deciding whether the maximum regret is at least 1 for a given strategy.

The integer feasibility problem is to tell for $H \in \mathbb{R}^{m\times n}$ and $\mathbf{t} \in \mathbb{R}^m$, whether there exist a vector $\mathbf{x} \in \{0,1\}^n$ such that $H\mathbf{x} \leq \mathbf{t}$. Now consider the following MDP:

Let $\mathbf{r}_a$ denote the vector form of r_{ai} and let $\mathcal{R}$ be defined by the following linear equalities/inequlities:

$$\begin{aligned} r_{ai} &= -1 - r_{bi},\ i = 1, \cdots n \\ -1 \leq r_{ai} &\leq 0,\ i = 1, \cdots n \\ r_0 &= -1, \\ -H\mathbf{r}_a &\leq \mathbf{t}. \end{aligned}$$

We claim that the integer feasibility problem is equivalent to whether the maximum regret of action b_0 is at least 1. Suppose the maximum regret is at least 1. Note that all rewards are negative and the performance of b_0 does not depend on the reward realization. Hence there exists $(\pi, \mathbf{r})$ such that $P(\pi, \mathbf{r}) = 0$, which means that the expected reward from s_i must be zero for all $i = 1, \cdots n$. Therefore, either r_{ai} or r_{bi} must equal to zero, i.e., $-r_{ai} \in \{0, 1\}$. Thus, let $x_i = -r_{ai}$, the integer feasibility problem has an affirmative answer. Now suppose that the integer feasibility problem has an affirmative answer, i.e., there exists $\mathbf{x}$ satisfying the integer feasibility, let $r_{ai} = -x_i$. Hence either r_{ai} or r_{bi} equals to zero, and the maximum expected reward equals to zero, i.e., the maximum regret of b_0 is 1. Therefore, we reduce the integer feasibility problem to deciding whether the maximum regret is at least 1, and hence the latter is NP-hard.

Combining the two steps, we conclude that deciding whether the maximum regret is at least 1 is NP complete. □

4.4. Algorithms for finding the MMR solution

Although the MMR solution is generally intractable, we propose in this section several ways to find the MMR strategy. In Subsection 4.4.1 we propose a subgradient

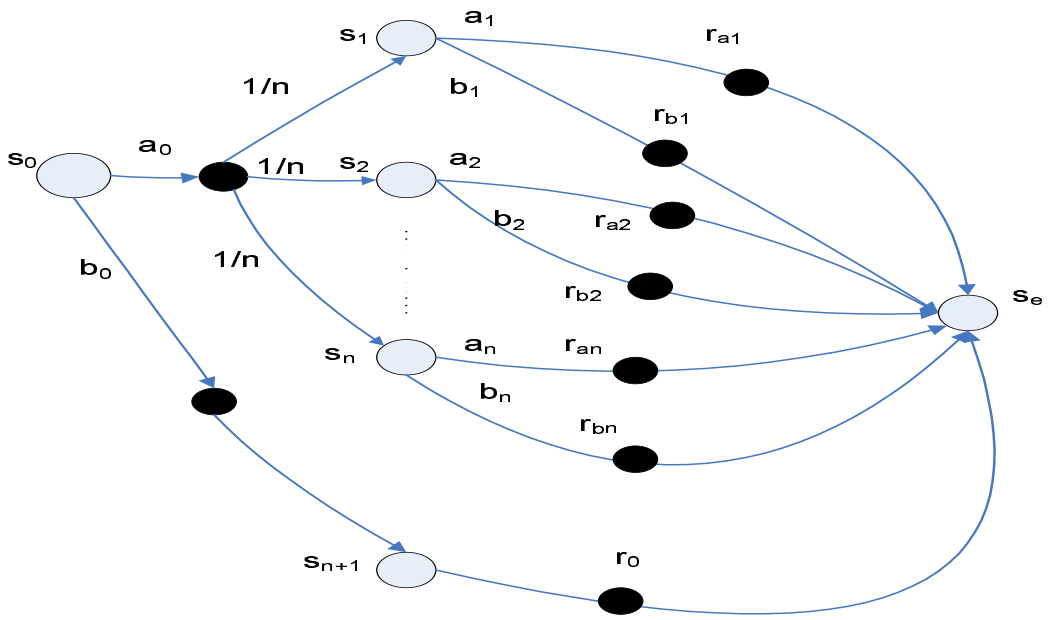


FIGURE 4.2. Regret evaluation is NP-hard.

method to find MMR, where the subgradient in each step is evaluate by a Mixed Integer Program (MIP). Due to the NP-hardness of MIP, such an algorithm is inherently non-polynomial. We further consider two special cases where polynomial algorithms are possible. (1) In Section 4.4.2 we show that when the number of vertices of $\mathcal{R}$ is small, i.e., $\mathcal{R}$ is the convex hull of a small number of parameter realizations, we can find MMR in polynomial time by solving a linear program. (2) In Section 4.4.3 we show that the MMR has a special property: it is a randomization of "efficient" (defined subsequently) strategies. Furthermore, the weighting coefficients of this randomization can be obtained by LP. Thus we are able to solve MMR in an efficient way if the set of "efficient" strategy, which can be found using action elimination methods, contains a small number of elements.

4.4.1. Subgradient approach. In this subsection, we propose a subgradient method to find the MMR solution. The subgradient for each step is indeed the reward parameter that achieves the maximum regret. We further provide an "oracle" based on mixed integer programming that computes this subgradient. This method is non-polynomial, due to the inherent NP-hardness of the problem as shown in Section 4.3.

We first show that minimizing the maximum regret is indeed a convex program (w.r.t. an equivalent form the the decision variable π). Thus, the global optimum (i.e., the MMR strategy) can be found with a subgradient descent/projection method.

Recall the well-known equivalence between a strategy of MDP and its expected state-action frequency (cf [**124**]). We thus change the decision variable $\pi \in \Pi^{HR}$ to its state-action frequency vector $\mathbf{x}$, i.e., the vector form of $x(s,a) = \mathbb{E}_\pi \sum_{i=1}^{\infty} \gamma^{i-1} \mathbf{1}(s_i = s, a_i = a)$, and recast finding MMR strategy as the following minimization problem on $\mathbf{x}$.

$$\min_{\mathbf{x}\in\mathcal{X}} G(\mathbf{x}). \tag{4.5}$$

Here, $\mathcal{X}$ is the state-action polytope:

$$\mathcal{X} : \sum_{a\in A_{s'}} x(s',a) - \sum_{s\in S}\sum_{a\in A_s} \gamma p(s'|s,a)x(s,a) = \alpha(s'),\ \forall s';$$
$$x(s,a) \geq 0, \quad \forall s, \forall a;$$

and $G(\cdot) : \mathcal{X} \to \mathbb{R}$ is defined by

$$G(\mathbf{x}) \triangleq \max_{\mathbf{r}\in\mathcal{R},\mathbf{x}'\in\mathcal{X}} (\mathbf{r}^\top \mathbf{x}' - \mathbf{r}^\top \mathbf{x}).$$

THEOREM 4.4. *(1) Problem (4.5) is a convex program;*

(2) Given $\mathbf{x}_0 \in \mathcal{X}$,

$$-\arg\max_{\mathbf{r}\in\mathcal{R}} \left\{ \max_{\mathbf{x}'\in\mathcal{X}} (\mathbf{r}^\top \mathbf{x}' - \mathbf{r}^\top \mathbf{x}_0) \right\} \in \nabla G(\mathbf{x})|_{\mathbf{x}_0}.$$

PROOF. Observe that $\mathcal{X}$ is convex. To see that the objective function (i.e., the part inside the curled bracket) is convex, we note that for a fixed pair of $(\mathbf{r}, \mathbf{x}')$, function $(\mathbf{r}^\top \mathbf{x}' - \mathbf{r}^\top \mathbf{x})$ is affine. Therefore the objective function is the maximum over a class of affine functions and hence convex. The second claim follows from the Envelope Theorem (e.g.,[**126**]). □

Therefore, we propose here a subgradient descent/project algorithm.

ALGORITHM 4.1.

(1) Initialize. $n := 1$*; choose* $\mathbf{r}_0 \in \mathcal{R}$, $\mathbf{x}^* := \arg\max_{\mathbf{x}\in\mathcal{X}} \mathbf{r}_0^\top \mathbf{x}$.

(2) Oracle. Solve $\mathbf{r}^* := \arg\max_{\mathbf{r}\in\mathcal{R}} \left\{ \max_{\mathbf{x}'\in\mathcal{X}} (\mathbf{r}^\top \mathbf{x}' - \mathbf{r}^\top \mathbf{x}^*) \right\}$.

(3) Descent. $\hat{\mathbf{x}} := \mathbf{x}^* + \frac{\mathbf{r}^*}{n}$.

(4) Projection. Solve $\mathbf{x}^* := \arg\max_{\mathbf{x}\in\mathcal{X}} \|\mathbf{x} - \hat{\mathbf{x}}\|$.

(5) $n := n + 1$. *Go to Step 2.*

Note that the Projection step is a convex quadratic program over a polytope, which can be solved in polynomial time. In contrast, Step 2 is NP-hard as shown in Section 4.3. We thus propose a MIP formulation that finds $\arg\max_{\mathbf{r}\in\mathcal{R}} \left\{ \max_{\mathbf{x}'\in\mathcal{X}} (\mathbf{r}^\top \mathbf{x}' - \mathbf{r}^\top \mathbf{x}^*) \right\}$.

The formulation is based on a "large M" method.[2] Define

$$r_{\max} \triangleq \sup_{\mathbf{r}\in\mathcal{R}} \max_{s\in S, a\in A} r(s,a); \quad M \triangleq r_{\max}\Big(\sum_{i=1}^{T} \gamma^{i-1}\Big).$$

Note that $r_{\max}$ is finite since $\mathcal{R}$ is bounded, and $\sum_{i=1}^{T} \gamma^{i-1}$ is finite because $\gamma = 1$ only when T is finite. Observe that M is larger than or equal to the reward-to-go for any $s \in S$, $\pi \in \Pi^{HR}$ and $\mathbf{r} \in \mathcal{R}$.

THEOREM 4.5. *Given initial state distribution* $\boldsymbol{\alpha}$ *and* $\mathbf{x}^*$, *let* $\mathbf{r}^*$ *be the optimal solution of the following maximization problem on* $(\mathbf{z}, \mathbf{v}, \mathbf{q}, \mathbf{r})$,

$$\begin{aligned}
&\textit{max:} \sum_s \alpha(s) v(s) - \sum_{s\in S} \sum_{a\in A} r(s,a) x^*(s,a) \\
&\textit{S.T.:} \sum_{a\in A} z_{s,a} = 1, \quad \forall s \in S, \\
&\left.\begin{aligned}
&q(s,a) = r(s,a) + \gamma \textstyle\sum_{s'\in S} p(s'|s,a) v(s'), \\
&v(s) \geq q(s,a), \\
&v(s) \leq M(1 - z_{s,a}) + q(s,a), \\
&z_{s,a} \in \{0,1\},
\end{aligned}\right\} \forall s, a \\
&\mathbf{r} \in \mathcal{R}.
\end{aligned} \tag{4.6}$$

We have $\mathbf{r}^* = \arg\max_{\mathbf{r}\in\mathcal{R}} \left\{ \max_{\mathbf{x}'\in\mathcal{X}} (\mathbf{r}^\top \mathbf{x}' - \mathbf{r}^\top \mathbf{x}^*) \right\}$.

[2] A similar method is independently proposed in [**125**].

PROOF. We establish the following lemma first.

LEMMA 4.6. *Fix* $\mathbf{r}$. *The following set of constraints*

$$\begin{aligned} v(s) &= \max_{a \in A} q(s,a); \\ q(s,a) &= r(s,a) + \gamma \sum_{s' \in S} p(s'|s,a) v(s'). \end{aligned} \tag{4.7}$$

is equivalent to

$$\begin{aligned} &\sum_{a \in A} z_{s,a} = 1, \quad \forall s \in S, \\ &\left.\begin{aligned} &q(s,a) = r(s,a) + \gamma \textstyle\sum_{s' \in S} p(s'|s,a) v(s'), \\ &v(s) \geq q(s,a), \\ &v(s) \leq M(1 - z_{s,a}) + q(s,a), \\ &z_{s,a} \in \{0,1\}, \end{aligned}\right\} \forall s, a. \end{aligned} \tag{4.8}$$

PROOF. First note that since M is larger than or equal to the reward to go of any s, π and $\mathbf{r}$, any $\mathbf{v}, \mathbf{q}$ that satisfy (4.7) also satisfy (4.8). (Let $z_{s,a^*} = 1$ when a^* maximizes $q(s,\cdot)$. If multiple a^* exist, arbitrarily pick one.)

Now consider any $\mathbf{q}, \mathbf{v}, \mathbf{z}$ satisfying (4.8). Fix a s. We have $v(s) \leq q(s,a^*)$ for some $a^* \in A$. This is because $z(s,a) \in \{0,\, 1\}$ and $\sum_a z(s,a) = 1$ implies the existence of a^* such that $z(s,a^*) = 1$. Thus,

$$v(s) \leq M(1 - z_{s,a^*}) + q(s,a^*) = q(s,a^*).$$

Combining this with $v(s) \geq q(s,a)$ for all $a \in A$ implies that $v(s) = \max_{a \in A} q(s,a)$. Therefore, $(\mathbf{q}, \mathbf{v})$ satisfies Equation (4.7). □

We now prove the theorem. Note that for a fixed $\mathbf{r}$, (4.7) uniquely determines the reward-to-go $\mathbf{v}$ (cf [**124**]). Therefore, the unique solution that of (4.8) is the reward-to-go and hence $\sum_s \alpha(s) v(s)$ is the expected performance under $\mathbf{r}$. We thus conclude that $\mathbf{r}^* = \arg\max_{\mathbf{r} \in \mathcal{R}} \left\{ \max_{\mathbf{x}' \in \mathcal{X}} (\mathbf{r}^\top \mathbf{x}' - \mathbf{r}^\top \mathbf{x}^*) \right\}$. □

4.4.2. Vertices approach. We consider a special type of uMDP: the uncertainty set $\mathcal{R}$ has a small number of vertices. That is, there exists $\mathbf{r}_1, \cdots, \mathbf{r}_t$ such that

$$\mathcal{R} = \text{conv}\{\mathbf{r}_1, \cdots, \mathbf{r}_t\} \triangleq \left\{ \sum_{i=1}^{t} c_i \mathbf{r}_i | \sum_{i=1}^{t} c_i = 1;\ c_i \geq 0,\ \forall i \right\}.$$

THEOREM 4.7. *Given uMDP* $< T, \gamma, S, A, \mathbf{p}, \mathcal{R} >$, *suppose* $\mathcal{R} = \text{conv}\{\mathbf{r}_1, \cdots, \mathbf{r}_t\}$ *and the initial state-distribution is* $\boldsymbol{\alpha}$. *Let* $\hat{x}_i(s,a) \triangleq \mathbb{E}_{\pi_i'} \sum_{i=1}^{T} \gamma^{i-1} \mathbf{1}(s_i = s, a_i = a)$ *where*

$$\pi_i' = \arg\max_{\pi' \in \Pi^D} P(\pi', \mathbf{r_i});$$

and h^*, $\mathbf{x}^*$ *be an optimal solution of the following LP,*

$$\begin{aligned} &\textit{Min: } h \\ &\textit{S. T.: } h \geq \sum_{s \in S} \sum_{a \in A} \left[r_i(s,a) \hat{x}_i(s,a) - r_i(s,a) x(s,a) \right],\ \forall i, \\ &\sum_{a \in A} x(s',a) - \sum_{s \in S} \sum_{a \in A} \gamma p(s'|s,a) x(s,a) = \alpha(s'),\ \forall s', \\ &x(s,a) \geq 0,\ \forall s, \forall a. \end{aligned} \tag{4.9}$$

Then the MMR strategy π^* *is such that* $\pi^*(a|s) = x(s,a) / \sum_{a' \in A} x(s,a')$ *for all* s, a. *Here, the denominator is guaranteed to be nonzero.*

PROOF. We establish the following lemma first.

LEMMA 4.8. *For any* $\pi \in \Pi^{HR}$ *the following holds,*

$$R(\pi) = \max_{i=1,\cdots,t} \Big\{ P(\pi_i', \mathbf{r_i}) - P(\pi, \mathbf{r_i}) \Big\}.$$

PROOF. Fix a strategy $\pi \in \Pi^{HR}$. Define the following function ranging over $\mathcal{R}$:

$$R^{\pi}(\mathbf{r}) \triangleq \max_{\pi' \in \Pi^{HR}} \{ P(\pi', \mathbf{r}) - P(\pi, \mathbf{r}) \}.$$

It is easy to see that $R^{\pi}(\cdot)$ is convex because $P(\pi', \mathbf{r}) - P(\pi, \mathbf{r})$ is a linear function of $\mathbf{r}$ for any π', and hence $R^{\pi}(\cdot)$ is convex as it is the maximum of a class of linear functions.

By convexity of $R^\pi(\cdot)$ and definition of π'_i we have

$$R(\pi) = \max_{\mathbf{r}\in\mathcal{R}}\Big\{\max_{\pi'\in\Pi^R}\big[P(\pi',\mathbf{r_i}) - P(\pi,\mathbf{r_i})\big]\Big\} = \max_{\mathbf{r}\in\mathcal{R}} R^\pi(\mathbf{r}) = \max_{i=1,\cdots,t} R^\pi(\mathbf{r_i})$$
$$= \max_{i=1,\cdots,t}\Big\{P(\pi'_i,\mathbf{r_i}) - P(\pi,\mathbf{r_i})\Big\},$$

which establishes the lemma. □

Now we prove the theorem. By Lemma 4.8, we have

$$R(\pi) = \min_h\Big\{h | h \geq P(\pi'_i,\mathbf{r}_i) - P(\pi,\mathbf{r}_i),\ i=1,\cdots,t\Big\}.$$

Taking the minimum over $\pi\in\Pi^S$ on both sides, the theorem follows immediately by writing the MDP as its dual LP form, see [**124**] for the details. □

4.4.3. Efficient-strategy approach.

DEFINITION 4.4. *A strategy $\pi\in\Pi^D$ is called* efficient *if there is no $\pi'\in\Pi^{HR}$ such that $P(\pi,\mathbf{r}) < P(\pi',\mathbf{r})$ holds for all $\mathbf{r}\in\mathcal{R}$.*

THEOREM 4.9. *Suppose $\mathcal{R} = \{\mathbf{r}|A\mathbf{r}\leq\mathbf{b}\}$ and $\{\pi'_1,\cdots,\pi'_t\}\subset\Pi^D$ is a superset of the set of efficient strategies. Let $\hat{x}_i(s,a)\triangleq\mathbb{E}_{\pi'_i}\sum_{i=1}^T\gamma^{i-1}\mathbf{1}(s_i=s,a_i=a)$, whose vector form is denoted by $\hat{\mathbf{x}}_i$. Let $\mathbf{c}^*$ be an optimal solution of the following LP on h, $\mathbf{c}$ and $\mathbf{z}(i)$,*

$$\begin{aligned}
\min:\quad & h\\
S.T.:\quad & \sum_{i=1}^{\top} c_i = 1;\\
& \mathbf{c}\geq\mathbf{0};\\
& \left.\begin{array}{l} h\geq\mathbf{b}^\top\mathbf{z}(i);\\ A^\top\mathbf{z}(i)+\hat{X}\mathbf{c}=\hat{\mathbf{x}}_i;\\ \mathbf{z}(i)\geq\mathbf{0};\end{array}\right\} i=1,\cdots,t,
\end{aligned}$$

where $\hat{X} = (\hat{\mathbf{x}}_1, \cdots, \hat{\mathbf{x}}_t)$*, then the MMR strategy* π^* *is such that*

$$\pi^*(a|s) = \frac{\sum_{i=1}^t c_i \hat{x}_i(s,a)}{\sum_{a' \in A} \sum_{i=1}^t c_i \hat{x}_i(s,a')}; \ \forall s, a.$$

Here, the denominator is guaranteed to be nonzero.

PROOF. We first show that the MMR strategy is a *randomization* over $\pi'_1, \cdots, \pi'_t$, where "randomization" stands for the following: given a pool of deterministic strategies pick one according to an exogenous stochastic source and then follow it forever. It is well known that (cf. [**124**]) for any stationary strategy, there is an equivalent randomization over all deterministic strategies and vice versa. Hence there is a MMR that is a randomization due to Theorem 4.2. Further note that the probability of picking a non-efficient strategy must be zero, or there exists a strategy that performs strictly better for all $\mathbf{r}$ which contradicts the MMR condition. Hence the MMR strategy is a randomization over $\pi'_1, \cdots, \pi'_t$.

Observe that if the probability of picking π'_i is c_i, then the state-action frequency equals $\sum_{i=1}^t c_i \hat{\mathbf{x}}_i$. Thus, the MMR strategy is the following optimization problem:

$$\min_{\mathbf{c}: \sum_{j=1}^t c_j = 1; \mathbf{c} \geq \mathbf{0}} \left\{ \max_{i \in \{1, \cdots, t\}, \mathbf{r} \in \mathcal{R}} \left[\mathbf{r}^\top \hat{x}_i - \mathbf{r}^\top \sum_{j=1}^t c_j \hat{\mathbf{x}}_j \right] \right\}.$$

This can be rewritten as

$$\begin{aligned} &\min : h \\ &\text{S.T.:} \ \sum_{i=1}^\top c_i = 1; \\ &\qquad \mathbf{c} \geq \mathbf{0}; \\ &\qquad h \geq \max_{\mathbf{r} \in \mathcal{R}} (\hat{\mathbf{x}}_i^\top - \mathbf{c}^\top X^\top)\mathbf{r}, \quad i = 1, \cdots, t. \end{aligned} \tag{4.10}$$

By duality of LP (cf [**113, 24**]) and $\mathcal{R} = \{\mathbf{r}|A\mathbf{r} \leq \mathbf{b}\}$, $\max_{\mathbf{r}\in\mathcal{R}}(\hat{\mathbf{x}}_i^\top - \mathbf{c}^\top X^\top)\mathbf{r}$ equals to the following LP on $\mathbf{z}(i)$:

$$\text{Min: } \mathbf{b}^\top \mathbf{z}(i);$$

$$\text{S.T.: } A^\top \mathbf{z}(i) + \hat{X}\mathbf{c} = \hat{\mathbf{x}}_i;$$

$$\mathbf{z}(i) \geq \mathbf{0}.$$

Substituting it into (4.10) establishes the theorem. □

Observe that if a strategy maximizes the performance $P(\cdot, \mathbf{r}_0)$ for some parameter realization $\mathbf{r}_0 \in \mathcal{R}$, then it is efficient. The following proposition shows that the reverse also holds.

PROPOSITION 4.10. *Suppose $\mathcal{R}$ is convex and its relative interior is non-empty*[3]*. If a strategy $\pi \in \Pi^{HR}$ is efficient, then there exists $\mathbf{r}_0 \in \mathcal{R}$ such that $v^\pi(\mathbf{r}_0) = v^*(\mathbf{r}_0)$.*

PROOF. We define the following to simplify the expression:

$$v^\pi(\mathbf{r}) \triangleq P(\pi, \mathbf{r}); \quad \pi \in \Pi^{HR}.$$

$$v^*(\mathbf{r}) \triangleq \max_{\pi\in\Pi^{HR}} v^\pi(\mathbf{r}).$$

Before proving the proposition, we establish the following lemma.

LEMMA 4.11. *Let $\mathcal{R}$ be convex, then*

(1) for any $\pi \in \Pi^S$, $v^\pi(\cdot)$ is an affine function;

(2) $v^*(\cdot)$ is a convex, piecewise affine function.

PROOF. Note that given strategy $\pi \in \Pi^S$, we have

$$v^\pi(\mathbf{r}) = \sum_{s\in S}\sum_{a\in A}\{r(s,a)\mathbb{E}_\pi \sum_i \gamma^{i-1}\mathbf{1}(s_i = s, a_i = a)\}.$$

The right-hand side is affine of $\mathbf{r}$, which implies the first claim.

[3]See page 23 of [**33**] for the definition of relative interior. In particular, all polytopes have non-empty relative interior.

To prove the second claim, recall that (e.g., [**124**]) for a fixed $\mathbf{r}$, the optimal strategy is determined and stationary, i.e.,

$$v^*(\mathbf{r}) = \max_{\pi \in \Pi^{HR}} P(\pi, \mathbf{r}) = \max_{\pi \in \Pi^{D}} P(\pi, \mathbf{r}).$$

Further note that Π^D is a finite set, and $v^\pi(\mathbf{r})$ is affine. Thus $v^*(\cdot)$ is convex and piecewise affine, since it is a pointwise maximum over a finite number of affine functions. □

We now prove the proposition by contradiction. Assume there exists an efficient strategy π^* which does not maximize the expect reward for any realization. Note $v^{\pi^*}(\cdot)$ is affine. We construct a function $v'(\cdot)$ such that $v'(\mathbf{r}) > v^{\pi^*}(\mathbf{r})$ for all $\mathbf{r} \in \mathcal{R}$, and show that there exists a strategy $\pi' \in \Pi^{HR}$ such that $v^{\pi'}(\mathbf{r}) \geq v'(\mathbf{r})$ for all $\mathbf{r} \in \mathcal{R}$.

Step 1: To construct $v'(\cdot)$, note that by assumption $v^{\pi^*}(\mathbf{r}) < v^*(\mathbf{r})$ for all $\mathbf{r} \in \mathcal{R}$. Hence let $c_0 \triangleq \min_{\mathbf{r} \in \mathcal{R}} \left[v^*(\mathbf{r}) - v^{\pi^*}(\mathbf{r}) \right]$ and $\mathbf{r}_0 \in \arg\min_{\mathbf{r} \in \mathcal{R}} \left[v^*(\mathbf{r}) - v^{\pi^*}(\mathbf{r}) \right]$. These two definition is valid since $v^*(\cdot)$ and $v^{\pi^*}(\cdot)$ are continuous functions and $\mathcal{R}$ is compact. Let $v'(\mathbf{r}) \triangleq v^{\pi^*}(\mathbf{r}) + c_0$, observe that $v^{\pi^*}(\mathbf{r}) < v'(\mathbf{r}) \leq v^*(\mathbf{r})$ holds for all $\mathbf{r} \in \mathcal{R}$, and we also have $v'(\mathbf{r}_0) = v^*(\mathbf{r}_0)$. Note that $v^{\pi^*}(\mathbf{r})$ is an affine function, so is $v'(\mathbf{r})$ by definition, and we can rewrite

$$v'(\mathbf{r}) = \mathbf{g}^\top \mathbf{r} + \left[v^*(\mathbf{r}_0) - \mathbf{g}^\top \mathbf{r}_0 \right].$$

Step 2: To show there exists $\pi' \in \Pi^{HR}$ such that $v^{\pi'}(\mathbf{r}) \geq v'(\mathbf{r})$ for all $\mathbf{r} \in \mathcal{R}$. Let $\mathcal{R} \subseteq \mathbb{R}^m$ and we extend $v^*(\cdot)$ into the whole space, i.e., for $\mathbf{r} \in \mathbb{R}^m$, define

$$v_f^*(\mathbf{r}) \triangleq \max_{\pi \in \Pi^{D}} P(\pi, \mathbf{r});$$

$$v_o^*(\mathbf{r}) \triangleq \begin{cases} 0 & \text{if } \mathbf{r} \in \mathcal{R}; \\ +\infty & \text{otherwise.} \end{cases}$$

Note that $v'(\mathbf{r}) \leq v^*(\mathbf{r})$ holds for all $\mathbf{r} \in \mathcal{R}$ implies $\mathbf{g}^\top \mathbf{r} + \left[v^*(\mathbf{r}_0) - \mathbf{g}^\top \mathbf{r}_0 \right] \leq v_f^*(\mathbf{r}) + v_o^*(\mathbf{r})$ holds for all $\mathbf{r} \in \mathbb{R}^m$. Hence $\mathbf{g}$ is a subgradient to convex function $v_f^*(\mathbf{r}) + v_o^*(\mathbf{r})$ at $\mathbf{r}_0$,

denote as $\mathbf{g} \in \partial\big[v_f^*(\mathbf{r}_0) + v_o^*(\mathbf{r}_0)\big]$. Hence there exists $\mathbf{g}_f$, $\mathbf{g}_o$ such that $\mathbf{g}_f \in \partial v_f^*(\mathbf{r}_0)$, $\mathbf{g}_o \in \partial v_o^*(\mathbf{r}_0)$ and $\mathbf{g} = \mathbf{g}_f + \mathbf{g}_o$ (cf Theorem 23.8 of [**126**]).

Step 2.1 To prove there exists π' such that $v^{\pi'}(\mathbf{r}) = \mathbf{g}_f^\top \mathbf{r} + \big[v^*(\mathbf{r}_0) - \mathbf{g}_f^\top \mathbf{r}_0\big]$ for all $\mathbf{r} \in \mathcal{R}$. Let set $\Pi_0 \triangleq \arg\max_{\pi \in \Pi^D} v^\pi(\mathbf{r}_0)$, i.e., the set of strategies that achieves maximal at $\mathbf{r}_0$. Note that Π_0 is a finite set since Π^D is a finite set. Hence denote $\Pi_0 = \{\pi_1, \cdots, \pi_h\}$. Note that by definition of Π_0, $v^{\pi_i}(\mathbf{r}_0) = v^*(\mathbf{r}_0)$. Hence we can rewrite

$$v^{\pi_i}(\mathbf{r}) = \mathbf{d}_i^\top \mathbf{r} + [v^*(\mathbf{r}_0) - \mathbf{d}_i^\top \mathbf{r}_0],$$

for some $\mathbf{d}_i$ since $v^{\pi_i}(\cdot)$ is a linear function.

Recall $\mathbf{g}_f \in \partial v_f^*(\mathbf{r}_0)$, hence by a standard continuity argument we have in a sufficiently small open ball around $\mathbf{r}_0$, $\mathbf{g}_f^\top \mathbf{r} + \big[v^*(\mathbf{r}_0) - \mathbf{g}_f^\top \mathbf{r}_0\big] \le \max_{\pi \in \Pi_0} v^{\pi_i}(\mathbf{r})$. Note that the left-hand side is affine, and the right-hand side is piecewise affine, hence this inequality holds for all $\mathbf{r} \in \mathbb{R}^m$. That is

$$\mathbf{g}_f^\top (\mathbf{r} - \mathbf{r}_0) \le \max_{i \in \{1, \cdots, h\}} \mathbf{d}_i^\top (\mathbf{r} - \mathbf{r}_0), \quad \forall \mathbf{r} \in \mathbb{R}^m.$$

This implies there exists no $\mathbf{y} \in \mathbb{R}^{m+1}$ such that $[\mathbf{g}_f^\top, 1]\mathbf{y} \ge \max_{i \in \{1, \cdots, h\}} [\mathbf{d}_i^\top, 1]\mathbf{y}$, hence no $\mathbf{y}$ satisfy the following conditions

$$\begin{bmatrix} \mathbf{g}_f \\ 1 \end{bmatrix}^\top \mathbf{y} > 0; \quad \begin{bmatrix} \mathbf{d}_i \\ 1 \end{bmatrix}^\top \mathbf{y} \le 0; \ i = 1, \cdots, h.$$

By Farkas Lemma, this means there exists $\lambda_1, \cdots, \lambda_h$ such that $\lambda_i \ge 0$ and

$$\begin{bmatrix} \mathbf{g}_f \\ 1 \end{bmatrix} = \sum_{i=1}^h \lambda_i \begin{bmatrix} \mathbf{d}_i \\ 1 \end{bmatrix}.$$

This implies $\sum_{i=1}^h \lambda_i = 1$ and $\sum_{i=1}^h \lambda_i \mathbf{d}_i = \mathbf{g}_f$. Now construct a strategy π' as taking strategy π_i with probability λ_i, and we have

$$v^{\pi'}(\mathbf{r}) = \sum_{i=1}^h \lambda_i v^{\pi_i}(\mathbf{r}) = \sum_{i=1}^h \lambda_i \Big\{ \mathbf{d}_i^\top \mathbf{r} + [v^*(\mathbf{r}_0) - \mathbf{d}_i^\top \mathbf{r}_0] \Big\} = \mathbf{g}_f^\top \mathbf{r} + \big[v^*(\mathbf{r}_0) - \mathbf{g}_f^\top \mathbf{r}_0\big].$$

Step 2.2: To show that $v^{\pi'}(\mathbf{r}) \geq v'(\mathbf{r})$ for all $\mathbf{r} \in \mathcal{R}$. By definition of $v_o^*(\cdot)$ and $\mathbf{g}_o \in \partial v_o^*(\mathbf{r}_0)$ we have

$$\mathbf{g}_o^\top \mathbf{r} + [v_o^*(\mathbf{r}_0) - \mathbf{g}_o^\top \mathbf{r}_0] \leq 0, \quad \forall \mathbf{r} \in \mathcal{R}.$$

Recall $\mathbf{r}_0 \in \mathcal{R}$, which implies $v_o^*(\mathbf{r}_0) = 0$. Hence substitute this into $\mathbf{g} = \mathbf{g}_f + \mathbf{g}_o$ leads to

$$\begin{aligned} v'(\mathbf{r}) =& \mathbf{g}^\top \mathbf{r} + [v^*(\mathbf{r}_0) - \mathbf{g}^\top \mathbf{r}_0] \\ =& \mathbf{g}_o^\top \mathbf{r} + [v_o^*(\mathbf{r}_0) - \mathbf{g}_o^\top \mathbf{r}_0] + \mathbf{g}_f^\top \mathbf{r} + [v_f^*(\mathbf{r}_0) - \mathbf{g}_f^\top \mathbf{r}_0] \\ \leq& \mathbf{g}_f^\top \mathbf{r} + [v_f^*(\mathbf{r}_0) - \mathbf{g}_f^\top \mathbf{r}_0] \\ =& v^{\pi'}(\mathbf{r}). \quad \forall \mathbf{r} \in \mathcal{R}. \end{aligned}$$

Hence we proved Step 2. Combining two steps, we establish the proposition. □

We may thus use *action elimination* [**124**][**103**][**68**] to find a "small" superset of efficient strategies: if an action at a state can be determined to *not* belong to optimal policy for any parameter realization, it can be discarded and disregarded. If only a small number of strategies remains after action elimination[4], then we can solve MMR in a less computational expensive way.

4.5. Mean variance tradeoff of regret

So far we regarded the true parameters as deterministic but unknown. In this section we take a Bayesian approach: we treat the true parameters as a random vector following distribution μ known a-priori. Thus, given a strategy, its regret is a random variable whose probability distribution can be evaluated. We use the mean-variance tradeoff criterion to compare such random variables. That is, the strategy that minimizes the tradeoff (i.e., the convex combination) of the mean and variance of the regret is considered optimal.

[4]Of course this is not guaranteed due to the NP-hardness of MMR.

DEFINITION 4.5. *Suppose the true reward parameter* $\mathbf{r}^t$ *follows a distribution* μ *supported by a compact* $\mathcal{R}$*. For a strategy* $\pi \in \Pi^{HR}$*:*

(1) the regret mean *is*

$$\begin{aligned} E^R(\pi) &\triangleq \mathbb{E}_{\mathbf{r}^t}\Big\{ \max_{\pi' \in \Pi^{HR}} P(\pi', \mathbf{r}^t) - P(\pi, \mathbf{r}^t)\Big\} \\ &= \int \big[\max_{\pi' \in \Pi^{HR}} P(\pi', \mathbf{r}) - P(\pi, \mathbf{r})\big]\mu(d\mathbf{r}); \end{aligned} \tag{4.11}$$

(2) the regret variance *is*

$$Var^R(\pi) \triangleq \mathbb{E}_{\mathbf{r}^t}\Big[\max_{\pi' \in \Pi^{HR}} P(\pi', \mathbf{r}^t) - P(\pi, \mathbf{r}^t)\Big]^2 - (E^R(\pi))^2. \tag{4.12}$$

DEFINITION 4.6. *Suppose the true reward parameter* $\mathbf{r}^t$ *follows a distribution* μ *supported by a compact* $\mathcal{R}$*. Fix* $\lambda \in [0, 1]$*, the* Optimal Mean-Variance Tradeoff of Regret *(OMVTR) strategy is*

$$\pi_\lambda \triangleq \arg\min_{\pi \in \Pi^{HR}} \big[\lambda E^R(\pi) + (1-\lambda) Var^R(\pi)\big].$$

To simplify notations, define function $P^*(\cdot) : \mathcal{R} \to \mathbb{R}$ as

$$P^*(\mathbf{r}) \triangleq \max_{\pi \in \Pi^{HR}} P(\pi, \mathbf{r}),$$

i.e., the optimal reward-to-go given $\mathbf{r}$. Note that $P^*(\mathbf{r})$ is easy to compute, using for example dynamic programming.

THEOREM 4.12. *For* $\lambda \in [0, 1]$*, let* $\mathbf{x}_\lambda$ *be an optimal solution to the following convex quadratic program*

$$\begin{aligned} &\textit{min:} (1-\lambda)\mathbf{x}^\top \mathbb{E}(\mathbf{r}\mathbf{r}^\top)\mathbf{x} + \Big\{[(1-\lambda)\mathbb{E}(P^*(\mathbf{r})) - \lambda]\mathbb{E}(\mathbf{r}) - (1-\lambda)\mathbb{E}[P^*(\mathbf{r})\mathbf{r}]\Big\}^\top \mathbf{x} \\ &\textit{S.T.:} \sum_{a \in A} x(s', a) - \sum_{s \in S}\sum_{a \in A} \gamma p(s'|s, a) x(s, a) = \alpha(s'), \ \forall s' \\ &\qquad x(s, a) \geq 0, \ \forall s, a. \end{aligned} \tag{4.13}$$

The OMVTR strategy π_λ is such that $\pi_\lambda(a|s) = x_\lambda(s,a)/\sum_{a' \in A} x_\lambda(s,a')$ for all s, a. Here, the denominator is guaranteed to be nonzero.

PROOF. We again use the equivalence between Π^{HR} and state-action frequency polytope. Let $\mathbf{x}(\pi)$ be the state-action vector of a a strategy π. Observe that

$$\begin{aligned} E^R(\pi) &= \mathbb{E}(P^*(\mathbf{r}^t)) - \mathbb{E}(\mathbf{r}^t)^\top \mathbf{x}(\pi); \\ Var^R(\pi) &= \mathbb{E}\Big[\max_{\pi' \in \Pi^{HR}} P(\pi', \mathbf{r}^t) - P(\pi, \mathbf{r}^t) - E^R(\pi)\Big]^2 \\ &= \mathbb{E}\Big[P^*(\mathbf{r}^t) - {\mathbf{r}^t}^\top \mathbf{x}(\pi) - \mathbb{E}(P^*(\mathbf{r}^t)) + \mathbb{E}(\mathbf{r}^t)^\top \mathbf{x}(\pi)\Big]^2. \end{aligned}$$

Thus algebra yields

$$\begin{aligned} &\lambda E^R(\pi) + (1-\lambda) Var^R \\ =&\lambda \mathbb{E}(P^*(\mathbf{r})^t) - \lambda \mathbb{E}(\mathbf{r}^t)^\top \mathbf{x}(\pi) + (1-\lambda)\mathbb{E}\big\{P^*(\mathbf{r}^t)^2 - 2P^*(\mathbf{r}^t){\mathbf{r}^t}^\top \mathbf{x}(\pi) + ({\mathbf{r}^t}^\top \mathbf{x}(\pi))^2\big\} \\ &- (1-\lambda)[\mathbb{E}(P^*(\mathbf{r}^t))]^2 + 2(1-\lambda)\mathbb{E}[P^*(\mathbf{r}^t)]\mathbb{E}(\mathbf{r}^t)^\top \mathbf{x}(\pi) - (1-\lambda)[\mathbb{E}(\mathbf{r}^t)^\top \mathbf{x}(\pi)]^2. \end{aligned}$$

Note that the right-hand-side is equivalent to the minimizing objective in Problem (4.13) up to adding a constant, which establishes the theorem. □

We denote the objective function of Problem (4.13) by $O(\mathbf{x})$, whose coefficients can be approximated using Monte Carlo sampling. The following theorem establishes an error bound of the solution to the approximated problem $\overline{O}(\mathbf{x})$.

THEOREM 4.13. *Let π^* and $\overline{\pi}$ be the OMVTR and the solution to the approximated problem using n i.i.d. samples respectively. Denote $\hat{T} \triangleq \sum_{i=1}^{T} \gamma^{i-1}$; $V \triangleq |S| \times |A|$ and $\hat{R} \triangleq \sup_{\mathbf{r} \in \mathcal{R}} \max_{s \in S, a \in A} |r(s,a)|$. Then, the following holds:*

$$\begin{aligned} &\Pr\big\{\lambda E^R(\overline{\pi}) + (1-\lambda) Var^R(\overline{\pi}) \geq \lambda E^R(\pi^*) + (1-\lambda) Var^R(\pi^*) + 2\epsilon\big\} \\ &\leq (2V^2 + 4V + 2) \exp\left(\frac{-n\epsilon^2}{2\hat{R}^2(4\hat{T}^2\hat{R} + \hat{T})^2}\right). \end{aligned}$$

PROOF. We use overline to represent the empirical average of a quantity from n iid sampling. Let $\epsilon_0 = \epsilon/(4\hat{T}^2\hat{R} + \hat{T})$. Note that each element of the $V \times V$ random matrix $\mathbf{r}(i)\mathbf{r}(i)^\top$ belongs to $[-\hat{R}^2, \hat{R}^2]$; $P^*(\mathbf{r}(i)) \in [-\hat{T}\hat{R}, \hat{T}\hat{R}]$; each element of the V

dimension random vector $\mathbf{r}(i)$ belongs to $[-\hat{R}, \hat{R}]$; each element of the V dimension random vector $P^*(\mathbf{r}(i))\mathbf{r}(i)$ belongs to $[-\hat{T}\hat{R}^2, \hat{T}\hat{R}^2]$. By Hoeffding's inequality, the followings hold:

$$\Pr\left(\left\|\overline{\mathbf{r}\mathbf{r}^\top} - \mathbb{E}(\mathbf{r}\mathbf{r}^\top)\right\|_{\max} \geq R\epsilon_0\right) \leq 2V^2 \exp\left(-\frac{n\epsilon_0^2}{2\hat{R}^2}\right). \tag{4.14}$$

$$\Pr\left(\left|\overline{P^*(\mathbf{r})} - \mathbb{E}(P^*(\mathbf{r}))\right| \geq \hat{T}\epsilon_0\right) \leq 2\exp\left(-\frac{n\epsilon_0^2}{2\hat{R}^2}\right). \tag{4.15}$$

$$\Pr\left(\left\|\overline{\mathbf{r}} - \mathbb{E}(\mathbf{r})\right\|_{\infty} \geq \epsilon_0\right) \leq 2V \exp\left(-\frac{n\epsilon_0^2}{2\hat{R}^2}\right). \tag{4.16}$$

$$\Pr\left(\left\|\overline{P^*(\mathbf{r})\mathbf{r}} - \mathbb{E}(P^*(\mathbf{r})\mathbf{r})\right\|_{\infty} \geq \hat{T}\hat{R}\epsilon_0\right) \leq 2V \exp\left(-\frac{n\epsilon_0^2}{2\hat{R}^2}\right). \tag{4.17}$$

Here, $\|\cdot\|_{\max}$ is the largest absolute value of the elements of a matrix. Now note that $\mathbf{x} \in \mathcal{X}$ implies $\|\mathbf{x}\|_\infty \leq \hat{T}$. Thus algebra manipulations lead for $\mathbf{x} \in \mathcal{X}$:

$$\begin{aligned}
&\overline{O}(\mathbf{x}) - O(\mathbf{x}) \\
\leq&(1-\lambda)\mathbf{x}^\top\left\{\overline{\mathbf{r}\mathbf{r}^\top} - \mathbb{E}(\mathbf{r}\mathbf{r}^\top)\right\}\mathbf{x} + (1-\lambda)\left\{\overline{P^*(\mathbf{r})\mathbf{r}} - \mathbb{E}(P^*(\mathbf{r})\mathbf{r})\right\}^\top\mathbf{x} \\
&+ \left\{(1-\lambda)\overline{P^*(\mathbf{r})} + \lambda - (1-\lambda)\mathbb{E}(P^*(\mathbf{r})) - \lambda\right\} \cdot \left|\mathbb{E}(\mathbf{r})^\top\mathbf{x}\right| \\
&+ \left|(1-\lambda)\overline{P^*(\mathbf{r})} + \lambda\right| \cdot \left|(\overline{\mathbf{r}} - \mathbb{E}(\mathbf{r}))^\top\mathbf{x}\right| \\
\leq&(1-\lambda)\hat{T}^2\left\|\overline{\mathbf{r}\mathbf{r}^\top} - \mathbb{E}(\mathbf{r}\mathbf{r}^\top)\right\|_{\max} + (1-\lambda)\hat{T}\left\|\overline{P^*(\mathbf{r})\mathbf{r}} - \mathbb{E}(P^*(\mathbf{r})\mathbf{r})\right\|_\infty \\
&+ (1-\lambda)\hat{T}\hat{R}|\overline{P^*(\mathbf{r})} - \mathbb{E}(P^*(\mathbf{r}))| + \hat{T}[(1-\lambda)\hat{T}\hat{R} + \lambda]\|\overline{\mathbf{r}} - \mathbb{E}(\mathbf{r})\|_\infty.
\end{aligned}$$

Combining this with Inequalities (4.14) to (4.17), we have:

$$\begin{aligned}
&\Pr\left\{\max_{\mathbf{x}\in\mathcal{X}}|\overline{O}(\mathbf{x}) - O(\mathbf{x})| \geq \epsilon\right\} \\
\leq&(2V^2 + 4V + 2)\exp\left(\frac{-n\epsilon^2}{2\hat{R}^2(4\hat{T}^2\hat{R} + \hat{T})^2}\right),
\end{aligned}$$

which implies the theorem because

$$\begin{aligned}
&\left|\lambda\left[E^R(\overline{\pi})+(1-\lambda)Var^R(\overline{\pi})\right]-\left[\lambda E^R(\pi^*)+(1-\lambda)Var^R(\pi^*)\right]\right| \\
=&\left|O(\mathbf{x}(\overline{\pi}))-O(\mathbf{x}(\pi^*))\right| \\
\leq&\left|O(\mathbf{x}(\overline{\pi}))-\overline{O}(\mathbf{x}(\overline{\pi}))\right|+\left|O(\mathbf{x}(\pi^*))-\overline{O}(\mathbf{x}(\pi^*))\right| \\
\leq&2\max_{\mathbf{x}\in\mathcal{X}}|\overline{O}(\mathbf{x})-O(\mathbf{x})|.
\end{aligned}$$

□

4.6. Chapter summary

In this chapter we investigated decision making in a Markovian setup where the reward parameters are not known in advance. In contrast to the standard setup where a strategy is evaluated by its accumulated reward-to-go, we focus on the so-called competitive setup where the criterion is the parametric regret, i.e., the gap between the performance of the best strategy that is chosen after the true parameter realization is revealed and the performance of the strategy that is chosen before the parameter realization is revealed.

We considered two related formulations: minimax regret and mean-variance tradeoff of the regret. In the minimax regret formulation, the true parameters are regarded as deterministic but unknown, and the optimal strategy is the one that minimizes the worst-case regret under the most adversarial possible realization. We showed that the problem of computing the minimax regret strategy is NP-hard and propose algorithms to efficiently solve it under favorable conditions. The mean-variance tradeoff formulation requires a probabilistic model of the uncertain parameters and looks for a strategy that minimizes a convex combination of the mean and the variance of the regret. We proved that computing such a strategy can be done numerically in an efficient way.

MDPs in a competitive setup can model many real applications. However, unlike the standard setup, robust decision making in such a setup has not been thoroughly

investigated. This chapter aims to address this absence by recasting solution concepts that were successfully implemented for standard setup to the competitive setup and solve them with a reasonable computation cost.

CHAPTER 5

A Kalman Filter Design Based on the Performance/Robustness Tradeoff

This chapter investigates state estimation of a linear system. State estimation can be regarded as a decision making problem, where the output is the estimated state value. Therefore, we are interested in applying robust decision making in the optimal filtering design. In particular, we apply the Likely performance/Worst-case performance (i.e., robustness) tradeoff concept proposed in Chapter 3 to the design of Kalman filter: We consider filter design of a linear system with parameter uncertainty. In contrast to the robust Kalman filter which focuses on a worst case analysis, we propose a design methodology based on iteratively solving a tradeoff problem between nominal performance and robustness to the uncertainty. Our proposed filter can be computed online efficiently, is steady-state stable, and is less conservative than the robust filter. Part of the material in this chapter appears in [**172**] and [**170**].

5.1. Introduction

The Kalman filter addresses the estimation problem for linear systems, and is widely used in many fields including control, finance, communication etc (e.g.,[**35, 95**]). One central assumption of the Kalman filter is that the underlying state-space model is exactly known. In practice, this assumption is often violated, i.e., the

parameters we use as the system dynamics (referred as *nominal parameters* hereafter) are only guesses of the unknown true parameters. It is reported (e.g.,[**76, 140, 82**]) that in this case, the performance of the Kalman filter can deteriorate significantly. In [**130**], Sayed proposed a filtering framework based on a worst-case analysis (hereafter referred to as the *robust filter*), i.e., instead of iteratively minimizing the regularized residual norm as the standard Kalman filter does, the robust filter minimizes the worst-possible regularized residual norm over the set of admissible uncertainty.

Empirical studies show that the Kalman filter and the robust filter perform well in different setups: the performance (measured by the steady-state error variance) of the robust filter is significantly better than the Kalman filter when the uncertainty is large; but under small uncertainty, its performance is not satisfactory, indicating over-conservativeness comparing to the standard Kalman filter. Furthermore, the robust filter usually has a slower transient response. Therefore, a filter that exhibits a similar performance to the better filter under all cases is desirable.

In this chapter, we present a new filter design approach to achieve this goal by interpolating the standard Kalman filter and the robust filter. To be more specific, in each iteration, the proposed filter finds a *Pareto efficient* filtered estimation by minimizing the convex combination of the nominal regularized residue (the criterion of the Kalman filter) and the worst-case regularized residue (the criterion of the robust filter). This approach leads to an optimization problem that can be solved recursively similarly to the Kalman filter and hence can be applied on-line. The proposed filter is stable and achieves bounded error-variance. Simulation results show that the proposed filter exhibits a similar performance to the better one between the Kalman filter and the robust filter. That is, the performance of the proposed filter is similar to the Kalman filter under small uncertainty, and is comparable to the robust filter under large uncertainty. Therefore, the proposed filter is suitable for a wider range of problem setups.

We need to point out that the proposed filter achieves good tradeoff because it is the only interpolating method that achieves Pareto efficiency between the *nominal performance* given by the nominal residue and the *robustness* given by the worst residue. There are several other "robust" filters designs based on $\mathcal{H}_2/\mathcal{H}_\infty$ robust control (e.g., [**114, 4, 136, 179, 87, 121**]), set-inclusive robust optimization (e.g., [**15, 63**]), and guaranteed error variance minimization (e.g., [**121, 120, 163**]). The main difference is that these methods perform de-regularization, and hence need to check certain existence conditions each iteration. If the existence conditions fail at some step, the robustness of the filter is not valid anymore. Furthermore, de-regularization leads to a computationally expensive algorithm, and hence is often not suitable in on-line application. See [**130**] for a more detailed comparison among different robust filter design methodologies.

This chapter is organized as follows. We formulate the filtering design as an optimization problem in Section 5.2, and show how to solve it in Section 5.3, which leads to the recursive formula for the proposed filter in Section 5.4. In Section 5.5 and Section 5.6 we investigate the theoretical and empirical behavior of the proposed filter respectively. Some concluding remarks are given in Section 5.7. Finally, in Section 5.8 we detail the derivation of the algorithm.

Notations: We use capital letters and boldface letters to denote matrices and column vectors respectively. Without further explanations, $\|\cdot\|$ stands for Euclidean norm for vectors, and largest singular value for matrices. The notation $col\{\mathbf{a}, \mathbf{b}\}$ stands for a column vector with entries $\mathbf{a}$ and $\mathbf{b}$, and $diag\{A, B\}$ denotes a block diagonal matrix with entries A and B. Given a column vector $\mathbf{z}$ and a positive definite matrix W, $\|\mathbf{z}\|_W^2$ stands for $\mathbf{z}^\top W \mathbf{z}$.

5.2. Filter formulation

We consider the following system:

$$\begin{aligned} \mathbf{x}_{i+1} =& (F_i + M_i \Delta_i E_{f,i})\mathbf{x}_i + (G_i + M_i \Delta_i E_{g,i})\mathbf{u}_i, \\ \mathbf{y}_i =& H_i \mathbf{x}_i + \mathbf{v}_i, \quad i = 0, 1, \cdots. \end{aligned} \tag{5.1}$$

Here, F_i , G_i, M_i, $E_{f,i}$ and $E_{g,i}$ are known matrices and Δ_i are unknown matrices with $\|\Delta_i\| \leq 1$. The variance of the initial state $\mathbf{x}_0$ is Π_0, and the driving noises $\mathbf{u}_i$ and $\mathbf{v}_i$ are white, zero mean and uncorrelated, with variance Q_i and R_i respectively. This formulation is standard in robust filter design [**130, 121**]. We denote the estimate of $\mathbf{x}_i$ given observation $\{\mathbf{y}_0, \cdots, \mathbf{y}_j\}$ by $\hat{\mathbf{x}}_{i|j}$, and denote its error variance by $P_{i|j}$. Furthermore, $\hat{\mathbf{x}}_i$ and P_i denote $\hat{\mathbf{x}}_{i|i-1}$ and $P_{i|i-1}$ respectively. We assume $P_{i|i}$ to be invertible, which can be relaxed because the final recursion form is independent of $P_{i|i}^{-1}$.

Both the Kalman filter and the Robust filter iteratively find the optimal/robust smoothing estimation and propagate them respectively (e.g., [**35, 95, 130**]), i.e.,

Kalman Filter:

$$\begin{aligned} (\hat{\mathbf{x}}_{i|i+1}, \hat{\mathbf{u}}_{i|i+1}) :=& \arg\min_{\mathbf{x}_i, \mathbf{u}_i} \Big\{ \|\mathbf{x}_i - \hat{\mathbf{x}}_{i|i}\|^2_{P^{-1}_{i|i}} + \|\mathbf{u}_i\|^2_{Q_i^{-1}} + \|\mathbf{y}_{i+1} - H_{i+1}\mathbf{x}_{i+1}\|^2_{R^{-1}_{i+1}} \big| \Delta_i = 0 \Big\}, \\ & \text{where: } \mathbf{x}_{i+1} = F_i \mathbf{x}_i + G_i \mathbf{u}_i; \\ \hat{\mathbf{x}}_{i+1|i+1} :=& F_i \hat{\mathbf{x}}_{i|i+1} + G_i \hat{\mathbf{u}}_{i|i+1}; \end{aligned}$$

Robust Filter:

$$\begin{aligned} (\hat{\mathbf{x}}_{i|i+1}, \hat{\mathbf{u}}_{i|i+1}) :=& \arg\min_{\mathbf{x}_i, \mathbf{u}_i} \max_{\|\Delta_i\| \leq 1} \Big\{ \|\mathbf{x}_i - \hat{\mathbf{x}}_{i|i}\|^2_{P^{-1}_{i|i}} + \|\mathbf{u}_i\|^2_{Q_i^{-1}} + \|\mathbf{y}_{i+1} - H_{i+1}\mathbf{x}_{i+1}\|^2_{R^{-1}_{i+1}} \Big\}, \\ & \text{where: } \mathbf{x}_{i+1} = F_i \mathbf{x}_i + G_i \mathbf{u}_i; \\ \hat{\mathbf{x}}_{i+1|i+1} :=& F_i \hat{\mathbf{x}}_{i|i+1} + G_i \hat{\mathbf{u}}_{i|i+1}. \end{aligned}$$

Notice here, the cost function for the Kalman filter is the error variance under the nominal parameters, whereas the cost function for the robust filter is the worst case error variance. Hence the former criterion stands for the nominal performance of the

smoothed estimation, and the latter represents how robust the smoothed estimation is. Ideally, a good estimation should perform well (in the sense of Pareto efficiency) for both criteria. This is equivalent to a minimizer of their convex combination, which leads to the proposed filter:

PROPOSED FILTER: Fix $\alpha \in (0,1)$

$$
\begin{aligned}
(\hat{\mathbf{x}}_{i|i+1}, \hat{\mathbf{u}}_{i|i+1}) := \arg\min_{\mathbf{x}_i, \mathbf{u}_i} \Big\{ & \alpha \big[\|\mathbf{x}_i - \hat{\mathbf{x}}_{i|i}\|^2_{P^{-1}_{i|i}} + \|\mathbf{u}_i\|^2_{Q_i^{-1}} + \|\mathbf{y}_{i+1} - H_{i+1}\mathbf{x}_{i+1}\|^2_{R^{-1}_{i+1}} \big| \Delta_i = 0 \big] \\
& + (1-\alpha) \max_{\|\Delta_i\| \le 1} \big[\|\mathbf{x}_i - \hat{\mathbf{x}}_{i|i}\|^2_{P^{-1}_{i|i}} + \|\mathbf{u}_i\|^2_{Q_i^{-1}} + \|\mathbf{y}_{i+1} - H_{i+1}\mathbf{x}_{i+1}\|^2_{R^{-1}_{i+1}} \big] \Big\}, \\
& \text{where: } \mathbf{x}_{i+1} = F_i \mathbf{x}_i + G_i \mathbf{u}_i; \\
\hat{\mathbf{x}}_{i+1|i+1} := & F_i \hat{\mathbf{x}}_{i|i+1} + G_i \hat{\mathbf{u}}_{i|i+1}.
\end{aligned}
\tag{5.2}
$$

Note that since both criteria are convex functions, not only any minimizer of the convex combination is Pareto efficient, but any Pareto efficient solution must minimize the convex combination for some α. Hence, this formulation computes all the solutions that achieve good tradeoff between the nominal performance and the robustness. This is different from other interpolation such as shrinking the uncertainty set, where the Pareto efficiency is not guaranteed.

5.3. Solving the minimization problem

To solve the minimization problem in Formulation (5.2), we denote

$$
\mathbf{z} \triangleq col\{\mathbf{x}_i - \hat{\mathbf{x}}_{i|i},\, \mathbf{u}_i\};\ \mathbf{b} \triangleq \mathbf{y}_{i+1} - H_{i+1}F_i\hat{\mathbf{x}}_{i|i};\ A \triangleq H_{i+1}[F_i,\, G_i];\ T \triangleq diag\{P^{-1}_{i|i},\, Q_i^{-1}\};
$$

$$
W \triangleq R^{-1}_{i+1}; \quad D \triangleq H_{i+1}M_i; \quad E_a \triangleq [E_{f,i},\, E_{g,i}]; \quad \mathbf{t} \triangleq -E_{f,i}\hat{\mathbf{x}}_{i|i}; \quad \phi(\mathbf{z}) \triangleq \|E_a\mathbf{z} - \mathbf{t}\|.
$$

We can rewrite Problem (5.2) as

$$
\arg\min_{\mathbf{z}} :\, C(\mathbf{z}) \triangleq \mathbf{z}^\top T\mathbf{z} + \alpha(A\mathbf{z} - \mathbf{b})^\top W(A\mathbf{z} - \mathbf{b}) + (1-\alpha) \max_{\|\mathbf{y}\| \le \phi(\mathbf{z})} \|A\mathbf{z} - \mathbf{b} + D\mathbf{y}\|^2_W,
\tag{5.3}
$$

Problem (5.3) is a bilevel optimization problem which is generally NP-hard. However, following a similar argument as [**131**], we show this special problem can be efficiently solved by converting into a *unimodal* scalar optimization problem. Before giving the main result of this section, we need to define the following functions of $\lambda \in \big[\|D^\top W D\|, +\infty\big)$:

$$\begin{aligned}
&\overline{W}(\lambda) \triangleq W + (1-\alpha) W D (\lambda I - D^\top W D)^\dagger D^\top W, \\
&\mathbf{z}^o(\lambda) \triangleq \arg\min_{\mathbf{z}} \Big\{ \mathbf{z}^\top T \mathbf{z} + (A\mathbf{z}-\mathbf{b})^\top \overline{W}(\lambda)(A\mathbf{z}-\mathbf{b}) + (1-\alpha)\lambda \phi^2(\mathbf{z}) \Big\}, \\
&G(\lambda) \triangleq \min_{\mathbf{z}} \Big\{ \mathbf{z}^\top T \mathbf{z} + (A\mathbf{z}-\mathbf{b})^\top \overline{W}(\lambda)(A\mathbf{z}-\mathbf{b}) + (1-\alpha)\lambda \phi^2(\mathbf{z}) \Big\} \\
&\qquad = \mathbf{z}^{o\top}(\lambda) T \mathbf{z}^o(\lambda) + \big(A\mathbf{z}^o(\lambda)-\mathbf{b}\big)^\top \overline{W}(\lambda) \big(A\mathbf{z}^o(\lambda)-\mathbf{b}\big) + (1-\alpha)\lambda \phi^2\big(\mathbf{z}^o(\lambda)\big).
\end{aligned}$$

Here, $(\cdot)^\dagger$ stands for the pseudo inverse of a matrix. Note that $T > 0$, $\phi(\cdot)$ is convex, and $\lambda \geq \|D^\top W D\|$ implies $\overline{W}(\lambda) \geq 0$, hence the definitions of $\mathbf{z}^o(\lambda)$ and $G(\lambda)$ are valid, because the part in the curled bracket is strictly convex on $\mathbf{z}$. Therefore, for any given λ we can evaluate $\mathbf{z}^o(\lambda)$ and $G(\lambda)$. The next theorem shows that the optimal $\mathbf{z}$ for Problem (5.3) can be evaluated by minimizing $G(\lambda)$ using line search and substituting the minimizer into $\mathbf{z}^o(\cdot)$.

THEOREM 5.1. *(1) Let* $\lambda^o \triangleq \arg\min_{\lambda \geq \|D^\top W D\|} G(\lambda)$, *we have*

$$\arg\min_{\mathbf{z}} C(\mathbf{z}) = \mathbf{z}^o(\lambda^o); \quad \min_{\mathbf{z}} C(\mathbf{z}) = G(\lambda^o).$$

(2) On $\lambda \geq \|D^\top W D\|$, $G(\lambda)$ *has only one local minimum, which is also its global minimum.*

PROOF. Define $R(\mathbf{z}, \mathbf{y}) \triangleq (A\mathbf{z} - \mathbf{b} + H\mathbf{y})^\top W (A\mathbf{z} - \mathbf{b} + H\mathbf{y})$ and $\hat{W}(\lambda) \triangleq W + W D (\lambda I - D^\top W D)^\dagger D^\top W$, for $\lambda \in \big[\|D^\top W D\|, +\infty\big)$. Hence $\overline{W}(\lambda) = \alpha W + (1-\alpha)\hat{W}(\lambda)$. Lemma 5.2 describes the property of $R(\mathbf{z}, \mathbf{y})$; its proof can be found in [**131**].

LEMMA 5.2. (a) *Function* $\max_{\|\mathbf{y}\| \leq \phi(\mathbf{z})} R(\mathbf{z}, \mathbf{y})$ *is convex on* $\mathbf{z}$.

(b) *For all* $\mathbf{z}$*,*

$$\max_{\|\mathbf{y}\|\le\phi(\mathbf{z})} R(\mathbf{z},\mathbf{y}) = \min_{\lambda\ge\|D^\top W D\|} (A\mathbf{z}-\mathbf{b})^\top \hat{W}(\lambda)(A\mathbf{z}-\mathbf{b}) + \lambda\phi^2(\mathbf{z}).$$

(c) $\lambda^o(\mathbf{z}) \triangleq \arg\min_{\lambda\ge\|D^\top W D\|}(A\mathbf{z}-\mathbf{b})^\top \hat{W}(\lambda)(A\mathbf{z}-\mathbf{b}) + \lambda\phi^2(\mathbf{z})$ *is well defined and continuous.*

Therefore, the following holds:

$$\begin{aligned}
&\min_{\mathbf{z}} C(\mathbf{z}) \\
&= \min_{\mathbf{z}} \Big\{ \mathbf{z}^\top T\mathbf{z} + \alpha(A\mathbf{z}-\mathbf{b})^\top W(A\mathbf{z}-\mathbf{b}) + (1-\alpha) \max_{\|\mathbf{y}\|\le\phi(\mathbf{z})} R(\mathbf{z},\mathbf{y}) \Big\} \\
&= \min_{\mathbf{z}} \Big\{ \mathbf{z}^\top T\mathbf{z} + \alpha(A\mathbf{z}-\mathbf{b})^\top W(A\mathbf{z}-\mathbf{b}) + (1-\alpha) \\
&\quad \times \min_{\lambda\ge\|D^\top W D\|} \big[(A\mathbf{z}-\mathbf{b})^\top \hat{W}(\lambda)(A\mathbf{z}-\mathbf{b}) + \lambda\phi^2(\mathbf{z}) \big] \Big\} \\
&= \min_{\lambda\ge\|D^\top W D\|} \min_{\mathbf{z}} \Big\{ \mathbf{z}^\top T\mathbf{z} + (A\mathbf{z}-\mathbf{b})^\top \overline{W}(\lambda)(A\mathbf{z}-\mathbf{b}) + (1-\alpha)\lambda\phi^2(\mathbf{z}) \Big\} \\
&= \min_{\lambda\ge\|D^\top W D\|} G(\lambda).
\end{aligned}$$

We now show that $G(\cdot)$ is unimodal. Denote $H(\mathbf{z},\lambda) \triangleq \mathbf{z}^\top T\mathbf{z} + (A\mathbf{z}-\mathbf{b})^\top \overline{W}(\lambda)(A\mathbf{z}-\mathbf{b}) + (1-\alpha)\lambda\phi^2(\mathbf{z})$. Observe that $C(\mathbf{z}) = \min_{\lambda\ge\|D^\top W D\|} H(\mathbf{z},\lambda)$ and

$$\begin{aligned}
\lambda^o(\mathbf{z}) &= \arg\min_{\lambda\ge\|D^\top W D\|} (A\mathbf{z}-\mathbf{b})^\top \hat{W}(\lambda)(A\mathbf{z}-\mathbf{b}) + \lambda\phi^2(\mathbf{z}) \\
&= \arg\min_{\lambda\ge\|D^\top W D\|} \Big\{ \mathbf{z}^\top T\mathbf{z} + \alpha(A\mathbf{z}-\mathbf{b})^\top W(A\mathbf{z}-\mathbf{b}) \\
&\quad + (1-\alpha)\big[(A\mathbf{z}-\mathbf{b})^\top \hat{W}(\lambda)(A\mathbf{z}-\mathbf{b}) + \lambda\phi^2(\mathbf{z}) \big] \Big\} \\
&= \arg\min_{\lambda\ge\|D^\top W D\|} H(\mathbf{z},\lambda).
\end{aligned}$$

Hence $G(\lambda) = \min_{\mathbf{z}} H(\mathbf{z},\lambda)$. Note that $C(\mathbf{z})$ is strictly convex and goes to infinity whenever $\|\mathbf{z}\| \uparrow \infty$, which implies $C(\mathbf{z})$ is unimodal and has a unique global minimum. Also note, $H(\mathbf{z},\lambda)$ has the following property: fix one variable, then it is a unimodal function of the other variable and achieves unique minimum on its domain. This,

combined with the continuity of $\lambda^o(\mathbf{z})$, establishes the unimodality of $G(\cdot)$ by applying Lemma C.2 in [**131**]. □

Note that $\phi(\mathbf{z}) = \|E_a\mathbf{z} - \mathbf{t}\|$ yields a closed form for $\mathbf{z}^o(\cdot)$:

$$\mathbf{z}^o(\lambda) = \Big(T + A^\top \overline{W}(\lambda)A + (1-\alpha)\lambda E_a^\top E_a\Big)^{-1}\Big(A^\top \overline{W}(\lambda)\mathbf{b} + (1-\alpha)\lambda E_a^\top \mathbf{t}\Big). \quad (5.4)$$

5.4. Recursive formula of the filter

Substituting Equation (5.4) into Problem (5.2) and with some algebra detailed in Section 5.8, we obtain the recursion formula of the proposed filter. We present the prediction form which propagates $\{\hat{\mathbf{x}}_i, P_i\}$, whereas the Measurement-Update form which propagates $\{\hat{\mathbf{x}}_{i|i}, P_{i|i}\}$ can be found in Section 5.8. The recursive formula of the proposed filter is a modified version of the Robust filter, where $*$ are the modifications. In addition, $G(\lambda)$ and hence λ^o are also different.

ALGORITHM 5.1. **Prediction form**

(1) Initialize: $\hat{\mathbf{x}}_0 := 0, \qquad P_0 := \Pi_0, \qquad \hat{R}_0 := R_0.$

(2) Given $\hat{R}_i, H_i, P_i$*, calculate:*

$$\begin{aligned} P_{i|i} &:= (P_i^{-1} + H_i^\top \hat{R}_i^{-1} H_i)^{-1} \\ &= P_i - P_i H_i^\top (\hat{R}_i + H_i P_i H_i^\top)^{-1} H_i P_i. \end{aligned}$$

(3) Recursion: Construct and minimize $G(\lambda)$ *over* $(\|M_i^\top H_{i+1}^\top R_{i+1}^{-1} H_{i+1} M_i\|, +\infty)$. *Let the optimal value be* λ_i^o. *Computing the following values:*

$$
\begin{aligned}
\hat{\lambda}_i :=& (1-\alpha)\lambda_i^o \qquad * \\
\overline{R}_{i+1} :=& R_{i+1} - \lambda^{o-1} H_{i+1} M_i M_i^\top H_{i+1}^\top \\
\hat{R}_{i+1}^{-1} :=& \alpha R_{i+1}^{-1} + (1-\alpha)\overline{R}_{i+1}^{-1} \qquad * \\
\hat{Q}_i^{-1} :=& Q_i^{-1} + \hat{\lambda}_i E_{g,i}^\top \big[I + \hat{\lambda}_i E_{f,i} P_{i|i} E_{f,i}^\top\big]^{-1} E_{g,i} \\
\hat{P}_{i|i} :=& (P_{i|i}^{-1} + \hat{\lambda}_i E_{f,i}^\top E_{f,i})^{-1} \\
=& P_{i|i} - P_{i|i} E_{f,i}^\top (\hat{\lambda}_i^{-1} I + E_{f,i} P_{i|i} E_{f,i}^\top)^{-1} E_{f,i} P_{i|i} \\
\hat{G}_i :=& G_i - \hat{\lambda}_i F_i \hat{P}_{i|i} E_{f,i}^\top E_{g,i} \\
\hat{F}_i :=& (F_i - \hat{\lambda}_i \hat{G}_i \hat{Q}_i E_{g,i}^\top E_{f,i})(I - \hat{\lambda}_i \hat{P}_{i|i} E_{f,i}^\top E_{f,i}) \\
\overline{H}_i^\top :=& \Big[H_i^\top \hat{R}_i^{-\top/2} \sqrt{\hat{\lambda}_i}\Big] \\
\overline{R}_{e,i} :=& I + \overline{H}_i P_i \overline{H}_i^\top \\
\overline{K}_i :=& F_i P_i \overline{H}_i^\top \\
P_{i+1} :=& F_i P_i F_i^\top - \overline{K}_i \overline{R}_{e,i}^{-1} \overline{K}_i^\top + \hat{G}_i \hat{Q}_i \hat{G}_i^\top \\
\mathbf{e}_i :=& \mathbf{y}_i - H_i \hat{\mathbf{x}}_i \\
\hat{\mathbf{x}}_{i+1} :=& \hat{F}_i \hat{\mathbf{x}}_i + \hat{F}_i P_{i|i} H_i^\top \hat{R}_i^{-1} \mathbf{e}_i \\
=& \hat{F}_i \hat{\mathbf{x}}_i + \hat{F}_i P_i H_i^\top R_{e,i}^{-1} \mathbf{e}_i.
\end{aligned}
$$

5.5. Steady-state analysis

In this section we study steady-state characteristics of the proposed filter, namely closed-loop stability and bounded error-variance. Similarly to [**130**], we restrict our discussion to uncertainty models where all parameters are stationary, except Δ_i, and drop the subscript i. Further assume the uncertainty only appears in the F matrix. Hence, we have $\hat{Q} = Q$ and $\hat{G} = G$. In addition, we approximate λ^o by setting

$\lambda^o := (1+\beta)\|M^\top H^\top R^{-1} H M\|$ for some $\beta > 0$. The next theorem shows that the proposed filter converges to a stable steady-state filter.

THEOREM 5.3. *Assume that $\{F, \overline{H}\}$ is detectable and $\{F, GQ^{1/2}\}$ is stabilizable. Then, for any initial condition $\Pi_0 > 0$, the Riccati variable P_i converges to the unique solution of*

$$P = FPF^\top - FP\overline{H}^\top (I + \overline{H} P \overline{H}^\top)^{-1} \overline{H} P F^\top + GQG^\top. \tag{5.5}$$

Furthermore, the solution P is semi-definite positive, and the steady state closed loop matrix $F_p \triangleq \hat{F}[I - PH^\top R_e^{-1} H]$ is stable.

PROOF. The closed loop formula for $\hat{\mathbf{x}}$ is

$$\begin{aligned}\hat{\mathbf{x}}_{i+1} &= \hat{F}_i \hat{\mathbf{x}}_i + \hat{F}_i P_i H^\top R_{e,i}^{-1} [\mathbf{y}_i - H\hat{\mathbf{x}}_i] \\ &= \hat{F}_i [I - P_i H^\top R_{e,i}^{-1} H] \hat{\mathbf{x}}_i + \hat{F}_i P_i H^\top R_{e,i}^{-1} \mathbf{y}_i.\end{aligned}$$

Notice that

$$\begin{aligned}&F\Big[I - P_i \overline{H}^\top \overline{R}_{e,i}^{-1} \overline{H}\Big] \\ &= F\Big[P_i - P_i \overline{H}^\top (I + \overline{H} P_i \overline{H}^\top)^{-1} \overline{H} P_i\Big] P_i^{-1} \\ &= F(P_i^{-1} + \overline{H}^\top \overline{H})^{-1} P_i^{-1}.\end{aligned}$$

Now consider the closed loop gain

$$\begin{aligned}&F_{p,i} \triangleq \hat{F}_i [I - P_i H^\top R_{e,i}^{-1} H] \\ =&F\Big[I - \hat{\lambda}(P_i^{-1} + \overline{H}^\top \overline{H})^{-1} E_f^\top E_f\Big]\Big[I - P_i H^\top R_{e,i}^{-1} H\Big] \\ =&F(P_i^{-1} + \overline{H}^\top \overline{H})^{-1}\Big[P_i^{-1} + \overline{H}^\top \overline{H} - \hat{\lambda} E_f^\top E_f\Big]\Big[I - P_i H^\top R_{e,i}^{-1} H\Big] \\ =&F(P_i^{-1} + \overline{H}^\top \overline{H})^{-1}(P_i^{-1} + H^\top \hat{R}^{-1} H)\Big[P_i - P_i H^\top R_{e,i}^{-1} H P_i\Big] P_i^{-1} \\ =&F(P_i^{-1} + \overline{H}^\top \overline{H})^{-1}(P_i^{-1} + H^\top \hat{R}^{-1} H)\Big[P_i - P_i H^\top (\hat{R}_i + H P_i H^\top)^{-1} H P_i\Big] P_i^{-1} \\ =&F(P_i^{-1} + \overline{H}^\top \overline{H})^{-1} P_i^{-1} = F\Big[I - P_i \overline{H}^\top \overline{R}_{e,i}^{-1} \overline{H}\Big].\end{aligned}$$

The positive definiteness of $\hat{R}$ guarantees that $\overline{H}$ is well defined. Hence, detectability of $\{F, \overline{H}\}$ and the stablizability of $\{F, GQ^{1/2}\}$ guarantee that P_i converges to the unique positive semi-definite solution P of Equation (5.5), which stabilizes the matrix $F[I-P\overline{H}^\top(I+\overline{H}P\overline{H}^\top)^{-1}\overline{H}]$. The stability follows for this matrix equals to the steady state closed loop gain F_p. □

Further assume that the system is quadratically stable, i.e, there exists a matrix $V > 0$ such that

$$V - [F + M\Delta E_f]^\top V[F + M\Delta E_f] > 0, \quad \forall \|\Delta\| \leq 1.$$

which is equivalent to a stable F and a bounded norm $\|E_f(zI - F)^{-1}M\|_\infty < 1$. Denote

$$\mathcal{F} \triangleq \begin{bmatrix} F - F_pPH^\top\hat{R}^{-1}H & F - F_p - F_pPH^\top\hat{R}^{-1}H \\ F_pPH^\top\hat{R}^{-1}H & F_p + F_pPH^\top\hat{R}^{-1}H \end{bmatrix},$$

$$\mathcal{G} \triangleq \begin{bmatrix} G & -F_pPH^\top\hat{R}^{-1}H \\ 0 & F_pPH^\top\hat{R}^{-1}H \end{bmatrix}.$$

The following theorem shows that the error-variance is uniformly bounded, which is equivalent to saying that the extended system is stable and has a $\mathcal{H}_\infty$ norm less than 1.

THEOREM 5.4. *Let $\tilde{x}_i$ be the estimation error. For any $\mathcal{P} > 0$ such that $\forall\|\Delta\| \leq 1$:*

$$\mathcal{P} - \left\{\mathcal{F} + \begin{bmatrix} M \\ 0 \end{bmatrix}\Delta[E_f \;\; E_f]\right\}\mathcal{P}\left\{\mathcal{F} + \begin{bmatrix} M \\ 0 \end{bmatrix}\Delta[E_f \;\; E_f]\right\}^\top - \mathcal{G}\begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix}\mathcal{G}^\top \geq 0;$$

the error variance satisfies $\lim_{i\to\infty}\mathbb{E}\tilde{x}_i\tilde{x}_i^\top \leq \mathcal{P}_{11}$, where $\mathcal{P}_{11}$ is the $(1,1)$ block entries of $\mathcal{P}$. Furthermore, such a $\mathcal{P}$ is guaranteed to exist.

PROOF. Define the estimation error $\tilde{\mathbf{x}}_i \triangleq \mathbf{x}_i - \hat{\mathbf{x}}_i$, and

$$\delta\mathcal{F}_i \triangleq \begin{bmatrix} M\Delta_iE_f & M\Delta_iE_f \\ 0 & 0 \end{bmatrix}.$$

Hence the extended state equation holds:

$$\begin{bmatrix} \tilde{\mathbf{x}}_{i+1} \\ \hat{\mathbf{x}}_{i+1} \end{bmatrix} = (\mathcal{F} + \delta\mathcal{F}_i) \begin{bmatrix} \tilde{\mathbf{x}}_i \\ \hat{\mathbf{x}}_i \end{bmatrix} + \mathcal{G} \begin{bmatrix} \mathbf{u}_i \\ \mathbf{v}_i \end{bmatrix}. \tag{5.6}$$

Introduce a similarity transformation:

$$\mathcal{T} \triangleq \begin{bmatrix} I & I \\ 0 & I \end{bmatrix}, \qquad \mathcal{T}^{-1} = \begin{bmatrix} I & -I \\ 0 & I \end{bmatrix}.$$

We have,

$$\mathcal{T}(\mathcal{F} + \delta\mathcal{F}_i)\mathcal{T}^{-1} = \begin{bmatrix} F & 0 \\ F_p P H^\top \hat{R}^{-1} H & F_p \end{bmatrix} + \begin{bmatrix} M\Delta_i E_f & 0 \\ 0 & 0 \end{bmatrix}.$$

Hence the first part (i.e., the nominal matrix, denote as $\tilde{\mathcal{F}}$) is stable since F and F_p are stable.

Furthermore, the following equality

$$E_f(zI - F)^{-1}M = [E_f \;\; 0](zI - \tilde{\mathcal{F}})^{-1} \begin{bmatrix} M \\ 0 \end{bmatrix},$$

shows that the extended system has the same $\mathcal{H}_\infty$-norm as the original system. Hence the extended system is quadratically stable. Thus, there exists a positive definite matrix $\mathcal{V}$ such that

$$\mathcal{V} - (\mathcal{F} + \delta\mathcal{F}_i)\mathcal{V}(\mathcal{F} + \delta\mathcal{F}_i)^\top > 0.$$

Scaling $\mathcal{V}$ by a sufficiently large factor, we can find a positive $\mathcal{P}$ such that

$$\mathcal{P} \geq (\mathcal{F} + \delta\mathcal{F}_i)\mathcal{P}(\mathcal{F} + \delta\mathcal{F}_i)^\top + \mathcal{G} \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix} \mathcal{G}^\top. \tag{5.7}$$

Let

$$\mathcal{M}_i \triangleq \mathbb{E}\left\{ \begin{bmatrix} \tilde{\mathbf{x}}_i \\ \hat{\mathbf{x}}_i \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{x}}_i \\ \hat{\mathbf{x}}_i \end{bmatrix}^\top \right\}.$$

Then the following recursion formula holds

$$\mathcal{M}_{i+1} = (\mathcal{F} + \delta\mathcal{F}_i)\mathcal{M}_i(\mathcal{F} + \delta\mathcal{F}_i)^\top + \mathcal{G}\begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix}\mathcal{G}^\top. \tag{5.8}$$

Subtracting Equation (5.8) from Equation (5.7) we get

$$\mathcal{P} - \mathcal{M}_{i+1} = (\mathcal{F} + \delta\mathcal{F}_i)(\mathcal{P} - \mathcal{M}_i)(\mathcal{F} + \delta\mathcal{F}_i)^\top + \mathcal{Q}_i,$$

for some $\mathcal{Q}_i \geq 0$. The quadratic stability of $\mathcal{F} + \delta\mathcal{F}_i$ implies that $\mathcal{P} - \mathcal{M}_\infty \geq 0$. □

5.6. Simulation study

In this section, we investigate the empirical performance of the proposed filter in three parameter setups that differ in the relative magnitude of the uncertainty. The following numerical example is frequently used in robust filtering design (e.g.,[**130**],[**121**]):

$$\mathbf{x}_{i+1} = \begin{bmatrix} 0.9802 & 0.0196 + 0.099\Delta_i \\ 0 & 0.9802 \end{bmatrix}\mathbf{x}_i + \mathbf{u}_i,$$

$$\mathbf{y}_i = [1 \ \ -1]\mathbf{x}_i + \mathbf{v}_i,$$

$$\text{where} \quad Q = \begin{bmatrix} 1.9608 & 0.0195 \\ 0.0195 & 1.9608 \end{bmatrix} \quad R = 1, \quad \mathbf{x}_0 \sim N(\mathbf{0}, I).$$

We note that the uncertainty only affects the $F_{1,2}$. and the magnitude of the nominal parameter and the uncertainty are of the same order. The tradeoff parameter α is set to 0.8. The error variance is averaged from 500 trajectories.

In Figure 5.1(a), the uncertainty Δ is generated according to a uniform distribution in $[-1,\ 1]$, and is fixed for the whole trajectory. In Figure 5.1(b), the uncertainty is re-generated in each step. In both cases, the proposed filter exhibits a similar steady-state performance to the robust filter, and a faster transient response (i.e., smaller error in the transient stages). We also observe that for the non-stationary

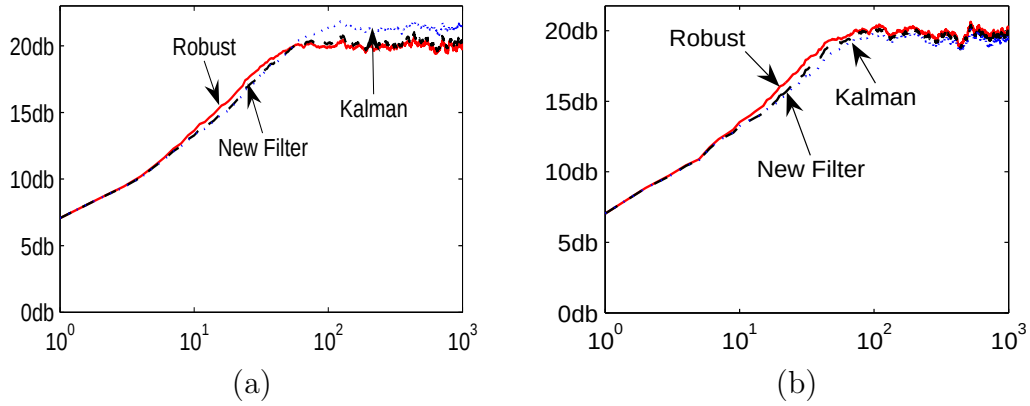


FIGURE 5.1. Error variance curves: (a) fixed uncertainty; (b) time-varying uncertainty.

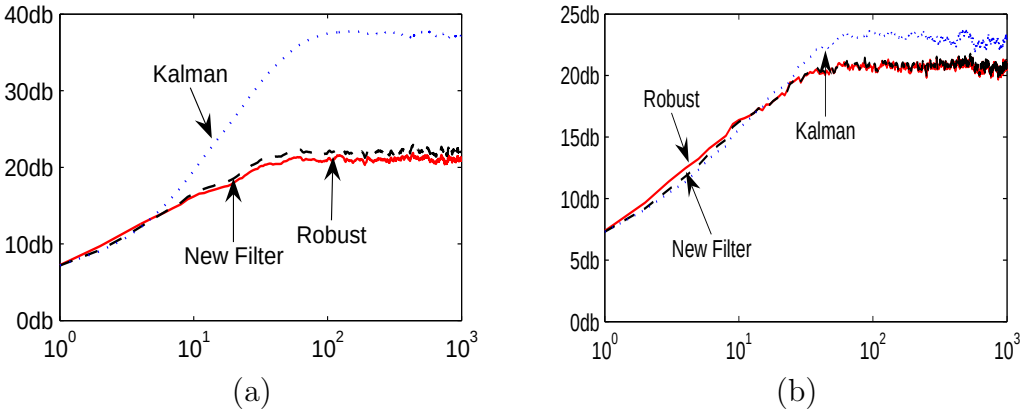


FIGURE 5.2. Error variance curves for large uncertainty: (a) fixed uncertainty; (b) time-varying uncertainty.

case, the robust filter performs worse probably due to the fact that time varying uncertainties cancel out.

In Figure 5.2, we depict the case with large uncertainty by setting $F_{1,2} = 0.0196 + 0.99\Delta_i$. In such situation, the performance of the Kalman filter degrades significantly. In contrast, the steady-state error of the proposed filter is only $1dB$ worse than the robust filter in the fixed uncertainty case, and is comparable to the robust filter in the time-varying case. This shows that the proposed filter achieves a comparable robustness as the robust filter.

In Figure 5.3, we investigate the small uncertainty case by enlarging nominal parameters, i.e., $F_{1,2} = 0.3912 + 0.099\Delta_i$. The robust filter achieves a steady-state error

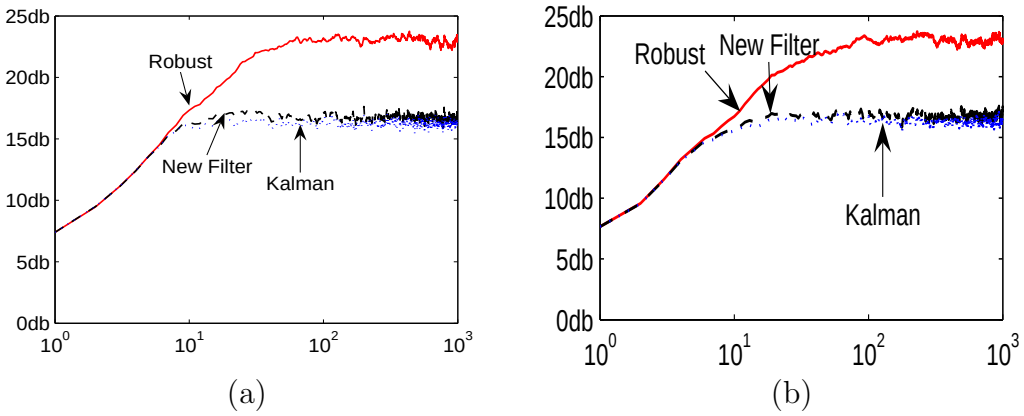


FIGURE 5.3. Error variance curves for large nominal value: (a) fixed uncertainty; (b) time-varying uncertainty.

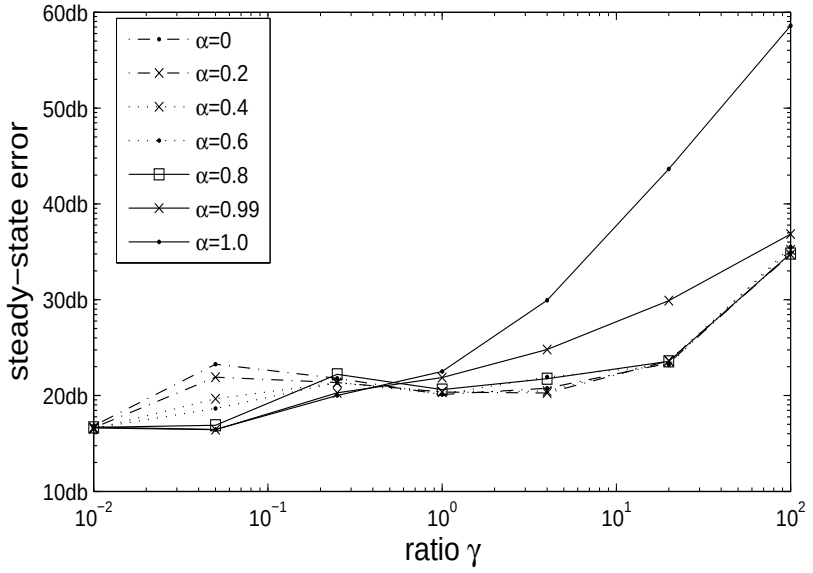


FIGURE 5.4. Effect of α on steady-state error.

variance around $23dB$, while both the Kalman filter and the proposed filter achieve a steady-state error around $16dB$. This shows that the robust filter could be overly conservative when the uncertainty is comparatively small, whereas the proposed filter does not suffer from such conservativeness.

We further simulate the steady-state error-variance for different α under different uncertainty ratio. Here, $\alpha = 0$ and $\alpha = 1$ are the robust filter and the Kalman filter,

respectively; $\gamma = 1$ is the original example. We increase the uncertainty when $\gamma > 1$, and increase the nominal parameter when $\gamma < 1$. Figure 5.4 shows that when γ is small, (i.e., uncertainty is relatively small), larger α achieves better performance. That is, for small uncertainty, focusing on robustness can degrade the performance. On the other hand, for large uncertainty, the steady-state error for the Kalman filter is large. In contrast, even for $\alpha = 0.99$ which means the robust part has a small effect, the proposed filter achieves a much better performance. The overall most-balanced filter in this example is achieved by taking $\alpha = 0.8$, which is also our suggestion for the tradeoff parameter. The exact value of α is not sensitive, for example, choosing $\alpha = 0.6$ instead works well too.

To summarize, the simulation study shows that both the Kalman filter and the robust filter are sensitive to the relative magnitude of the uncertainty. In contrast, in all three cases, the proposed filter exhibits a performance comparable to the better one, and therefore is suitable for a wider range of problems.

5.7. Chapter summary

In this chapter, we presented a new algorithm for state estimation of a linear system with uncertainty in the parameters. This filter iteratively finds a smoothed estimation that is Pareto efficient between the nominal performance and the worst performance. The resulting recursive form has a computational cost comparable to the standard Kalman filter, hence can be easily implemented on-line. Under certain technical conditions, the proposed filter converges to a stable steady-state estimator and achieves bounded error-variance. Simulation studies show that the proposed filter overcomes both the sensitivity of the Kalman filter and the overly conservativeness of the robust filter, and hence achieves satisfactory performance under a wider range of parameters.

The main motivation of the proposed approach is obtaining more flexibility in filter design while retaining the computational efficiency. As the simulation study showed, the performance of both the Kalman filter and the robust filter depend on

the parameter settings. That is, each of the filters can perform rather poorly under unsuitable parameters. Whether a problem setting is suitable for these filters may not be known beforehand, except a general guideline that small uncertainty favors the standard Kalman filter and large uncertainty favors the robust filter. Moreover, the problem parameters can be time varying. The proposed filter therefore facilitates flexibility since the quality of its performance does not vary dramatically if the magnitude of the uncertainty is not specified perfectly.

5.8. Derivation of the prediction form

In this section we show how to get the prediction form based on solving Problem (5.2). By Theorem 5.1 and Equation (5.4), we have

$$\begin{aligned} &col(\hat{\mathbf{x}}_{i|i+1} - \hat{\mathbf{x}}_{i|i}, \hat{\mathbf{u}}_{i|i+1}) = \mathbf{z}^o(\lambda^o) \\ =&\Big(T + A^\top \overline{W}(\lambda^o)A + (1-\alpha)\lambda^o E_a^\top E_a\Big)^{-1}\Big(A^\top \overline{W}(\lambda^o)\mathbf{b} + (1-\alpha)\lambda^o E_a^\top \mathbf{t}\Big), \end{aligned} \tag{5.9}$$

where λ^o is the minimizer of function $G(\lambda)$ over $\big[\|D^\top W D\|, +\infty\big)$. Since we are using a line search to find out λ^o, we exclude the boundary point $\|D^\top W D\|$. Hence, $\lambda^o I - D^\top W D$ is invertible. Denote

$$\begin{aligned} \overline{R}_{i+1} \triangleq& \hat{W}(\lambda^o)^{-1} \\ =&\big\{W + WD(\lambda^o I - D^\top W D)^{-1} D^\top W\big\}^{-1} \\ =&W^{-1} - (\lambda^o)^{-1} DD^\top = R_{i+1} - (\lambda^o)^{-1} H_{i+1} M_i M_i^\top H_{i+1}^\top. \end{aligned} \tag{5.10}$$

The second equality holds due to the matrix inversion lemma, and the last equality holds by substituting the definition of D and W. Next, define

$$\hat{R}_{i+1} \triangleq \overline{W}(\lambda^o)^{-1} = \big[\alpha W + (1-\alpha)\hat{W}(\lambda^o)\big]^{-1} = \big[\alpha R_{i+1}^{-1} + (1-\alpha)\overline{R}_{i+1}^{-1}\big]^{-1}. \tag{5.11}$$

Notice this definition makes sense since $\hat{W}^(\lambda)$ is positive for $\lambda > \|D^\top W D\|$ and W is also positive. Let

$$\hat{\lambda}_i \triangleq (1-\alpha)\lambda^o; \quad \hat{T} \triangleq \begin{pmatrix} P_{i|i}^{-1} + \hat{\lambda}_i E_{f,i}^\top E_{f,i} & \hat{\lambda}_i E_{f,i}^\top E_{g,i} \\ \hat{\lambda}_i E_{g,i}^\top E_{f,i} & Q_i^{-1} + \hat{\lambda}_i E_{g,i}^\top E_{g,i} \end{pmatrix}.$$

We then rewrite the first term of Equation (5.9):

$$\begin{aligned} & T + A^\top \overline{W}(\lambda^o) A + (1-\alpha)\lambda^o E_a^\top E_a \\ =& \begin{pmatrix} P_{i|i}^{-1} & 0 \\ 0 & Q_i^{-1} \end{pmatrix} + \hat{\lambda}_i [E_{f,i},\, E_{g,i}]^\top [E_{f,i},\, E_{g,i}] + A^\top \overline{W}(\lambda^o) A \qquad (5.12) \\ =& \hat{T} + A^\top \overline{W}(\lambda^o) A = \hat{T} + A^\top \hat{R}_{i+1}^{-1} A. \end{aligned}$$

Notice that the $(1,1)$ block of $\hat{T}$ is strictly positive. By block matrix inversion, we have

$$\hat{T}^{-1} = \begin{pmatrix} \hat{P}_{i|i} + \hat{P}_{i|i}\hat{\lambda}_i E_{f,i}^\top E_{g,i} \hat{Q}_i E_{g,i}^\top E_{f,i} \hat{\lambda}_i \hat{P}_{i|i} & -\hat{P}_{i|i}\hat{\lambda}_i E_{f,i}^\top E_{g,i} \hat{Q}_i \\ -\hat{Q}_i E_{g,i}^\top E_{f,i} \hat{\lambda}_i \hat{P}_{i|i} & \hat{Q}_i \end{pmatrix}, \qquad (5.13)$$

where $\hat{P}_{i|i} \triangleq \left(P_{i|i}^{-1} + \hat{\lambda}_i E_{f,i}^\top E_{f,i}\right)^{-1}$ is the inverse of the $(1,1)$ block of the matrix $\hat{T}$ and $\hat{Q}_i \triangleq \left(Q_i^{-1} + \hat{\lambda}_i E_{g,i}^\top E_{g,i} - \hat{\lambda}_i E_{g,i}^\top E_{f,i} \hat{P}_{i|i} E_{f,i}^\top E_{g,i} \hat{\lambda}_i\right)^{-1}$ is the inverse of the Schur complement.

We next simplify $(\hat{T} + A^\top \overline{W}(\lambda^o)A)^{-1}$ by first proving a useful equation:

$$
\begin{aligned}
&[F_i,\, G_i]\hat{T}^{-1}[F_i,\, G_i]^\top \\
=&[F_i,\, G_i] \begin{pmatrix} \hat{P}_{i|i} + \hat{P}_{i|i}\hat{\lambda}_i E_{f,i}^\top E_{g,i}\hat{Q}_i E_{g,i}^\top E_{f,i}\hat{\lambda}_i \hat{P}_{i|i} & -\hat{P}_{i|i}\hat{\lambda}_i E_{f,i}^\top E_{g,i}\hat{Q}_i \\ -\hat{Q}_i E_{g,i}^\top E_{f,i}\hat{\lambda}_i \hat{P}_{i|i} & \hat{Q}_i \end{pmatrix} [F_i,\, G_i]^\top \\
=&F_i(\hat{P}_{i|i} + \hat{P}_{i|i}\hat{\lambda}_i E_{f,i}^\top E_{g,i}\hat{Q}_i E_{g,i}^\top E_{f,i}\hat{\lambda}_i \hat{P}_{i|i})F_i^\top - F_i(\hat{P}_{i|i}\hat{\lambda}_i E_{f,i}^\top E_{g,i}\hat{Q}_i)G_i^\top \\
&- G_i(\hat{Q}_i E_{g,i}^\top E_{f,i}\hat{\lambda}_i \hat{P}_{i|i})F_i^\top + G_i(\hat{Q}_i)G_i^\top \\
=&F_i\hat{P}_{i|i}F_i^\top + F_i\hat{P}_{i|i}\hat{\lambda}_i E_{f,i}^\top E_{g,i}\hat{Q}_i E_{g,i}^\top E_{f,i}\hat{\lambda}_i \hat{P}_{i|i}F_i^\top - G_i\hat{Q}_i E_{g,i}^\top E_{f,i}\hat{\lambda}_i \hat{P}_{i|i}F_i^\top \\
&- F_i\hat{P}_{i|i}\hat{\lambda}_i E_{f,i}^\top E_{g,i}\hat{Q}_i G_i^\top + G_i\hat{Q}_i G_i^\top \\
=&F_i\hat{P}_{i|i}F_i^\top - (G_i - F_i\hat{P}_{i|i}\hat{\lambda}_i E_{f,i}^\top E_{g,i})\hat{Q}_i E_{g,i}^\top E_{f,i}\hat{\lambda}_i \hat{P}_{i|i}F_i^\top + (G_i - F_i\hat{P}_{i|i}\hat{\lambda}_i E_{f,i}^\top E_{g,i})\hat{Q}_i G_i^\top H_{i+1}^\top \\
=&H_{i+1}F_i\hat{P}_{i|i}F_i^\top + (G_i - \hat{\lambda}_i F_i\hat{P}_{i|i}E_{f,i}E_{g,i})\hat{Q}_i(G_i - \hat{\lambda}_i F_i\hat{P}_{i|i}E_{f,i}E_{g,i})^\top \\
=&F_i\hat{P}_{i|i}F_i^\top + \hat{G}_i\hat{Q}_i\hat{G}_i^\top = P_{i+1},
\end{aligned}
\tag{5.14}
$$

where

$$
\hat{G}_i \triangleq G_i - \hat{\lambda}_i F_i\hat{P}_{i|i}E_{f,i}E_{g,i}; \quad P_{i+1} \triangleq F_i\hat{P}_{i|i}F_i^\top + \hat{G}_i\hat{Q}_i\hat{G}_i^\top. \tag{5.15}
$$

Hence we can simplify $A\hat{T}^{-1}A^\top$ as

$$
\begin{aligned}
A\hat{T}^{-1}A^\top &= H_{i+1}[F_i,\, G_i]\hat{T}^{-1}[F_i,\, G_i]^\top H_{i+1}^\top \\
&= H_{i+1}(F_i\hat{P}_{i|i}F_i^\top + \hat{G}_i\hat{Q}_i\hat{G}_i^\top)H_{i+1}^\top = H_{i+1}P_{i+1}H_{i+1}^\top.
\end{aligned}
\tag{5.16}
$$

Define

$$
R_{e,i+1} \triangleq \hat{R}_{i+1} + H_{i+1}P_{i+1}H_{i+1}^\top = \hat{R}_{i+1} + A\hat{T}^{-1}A^\top. \tag{5.17}
$$

Hence

$$\begin{aligned}
&\left(T + A^\top \overline{W}(\lambda^o) A + (1-\alpha)\lambda^o E_a^\top E_a\right)^{-1} \\
=&(\hat{T} + A^\top \overline{W}(\lambda^o) A)^{-1} = (\hat{T} + A^\top \hat{R}_{i+1}^{-1} A)^{-1} \\
=&\hat{T}^{-1} - \hat{T}^{-1} A^\top (\hat{R}_{i+1} + A\hat{T}^{-1}A^\top)^{-1} A\hat{T}^{-1} \\
=&\hat{T}^{-1} - \hat{T}^{-1} A^\top R_{e,i+1}^{-1} A\hat{T}^{-1} \\
=&\hat{T}^{-1} - \hat{T}^{-1}[F_i,\, G_i]^\top H_{i+1}^\top R_{e,i+1}^{-1} H_{i+1}[F_i,\, G_i]\hat{T}^{-1} \\
=&\hat{T}^{-1} - \hat{T}^{-1} \begin{pmatrix} F_i^\top H_{i+1}^\top R_{e,i+1}^{-1} H_{i+1} F_i & F_i^\top H_{i+1}^\top R_{e,i+1}^{-1} H_{i+1} G_i \\ G_i^\top H_{i+1}^\top R_{e,i+1}^{-1} H_{i+1} F_i & G_i^\top H_{i+1}^\top R_{e,i+1}^{-1} H_{i+1} G_i \end{pmatrix} \hat{T}^{-1}.
\end{aligned} \tag{5.18}$$

The equations hold from (5.12), (5.11), (5.17), the matrix inversion lemma and the definition of A respectively.

Now consider the second term of Equation (5.9). By the definition of $\hat{\lambda}_i$, $\hat{R}_{i+1}$, and substituting A, $\mathbf{b}$, E_a and $\mathbf{t}$, we have:

$$\begin{aligned}
&A^\top \overline{W}(\lambda^o)\mathbf{b} + (1-\alpha)\lambda^o E_a^\top \mathbf{t} = A^\top \hat{R}_{i+1}^{-1}\mathbf{b} + \hat{\lambda}_i E_a^\top \mathbf{t} \\
=&[F_i,\, G_i]^\top H_{i+1}^\top \hat{R}^{-1}(\mathbf{y}_{i+1} - H_{i+1}F_i\hat{\mathbf{x}}_{i|i}) + \hat{\lambda}_i [E_{f,i},\, E_{g,i}]^\top (-E_{f,i}\hat{\mathbf{x}}_{i|i}) \\
=&\begin{pmatrix} F_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} \\ G_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} \end{pmatrix} \mathbf{y}_{i+1} + \begin{pmatrix} -F_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} H_{i+1}F_i - \hat{\lambda}_i E_{f,i}^\top E_{f,i} \\ -G_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} H_{i+1}F_i - \hat{\lambda}_i E_{g,i}^\top E_{f,i} \end{pmatrix} \hat{\mathbf{x}}_{i|i}.
\end{aligned} \tag{5.19}$$

Substitute Equation (5.12), Equation (5.19) into Equation (5.9) yields

$$\begin{aligned}
&\begin{pmatrix} \hat{\mathbf{x}}_{i|i+1} - \hat{\mathbf{x}}_{i|i} \\ \hat{\mathbf{u}}_{i|i+1} \end{pmatrix} \\
=&(\hat{T} + A^\top \hat{R}_{i+1}^{-1} A)^{-1} \left\{ \begin{pmatrix} F_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} \\ G_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} \end{pmatrix} \mathbf{y}_{i+1} \right. \\
&\left. + \begin{pmatrix} -F_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} H_{i+1}F_i - \hat{\lambda}_i E_{f,i}^\top E_{f,i} \\ -G_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} H_{i+1}F_i - \hat{\lambda}_i E_{g,i}^\top E_{f,i} \end{pmatrix} \hat{\mathbf{x}}_{i|i} \right\},
\end{aligned}$$

which implies

$$\begin{aligned}
&\begin{pmatrix} \hat{\mathbf{x}}_{i|i+1} \\ \hat{\mathbf{u}}_{i|i+1} \end{pmatrix} \\
=&(\hat{T} + A^\top \hat{R}_{i+1}^{-1} A)^{-1} \left\{ \begin{pmatrix} F_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} \\ G_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} \end{pmatrix} \mathbf{y}_{i+1} \right. \\
&\left. + \left[\begin{pmatrix} -F_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} H_{i+1} F_i - \hat{\lambda}_i E_{f,i}^\top E_{f,i} \\ -G_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} H_{i+1} F_i - \hat{\lambda}_i E_{g,i}^\top E_{f,i} \end{pmatrix} + (\hat{T} + A^\top \hat{R}_{i+1}^{-1} A) \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right] \hat{\mathbf{x}}_{i|i} \right\}.
\end{aligned} \tag{5.20}$$

Note that

$$\begin{aligned}
&(\hat{T} + A^\top \hat{R}_{i+1}^{-1} A) \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\
=&\left\{ \begin{pmatrix} P_{i|i}^{-1} + \hat{\lambda}_i E_{f,i}^\top E_{f,i} & \hat{\lambda}_i E_{f,i}^\top E_{g,i} \\ \hat{\lambda}_i E_{g,i}^\top E_{f,i} & Q_i^{-1} + \hat{\lambda}_i E_{g,i}^\top E_{g,i} \end{pmatrix} + \begin{bmatrix} F_i^\top \\ G_i^\top \end{bmatrix} H_{i+1}^\top \hat{R}_{i+1}^{-1} H_{i+1} [F_i,\, G_i] \right\} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\
=&\begin{pmatrix} P_{i|i}^{-1} + \hat{\lambda}_i E_{f,i}^\top E_{f,i} + F_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} H_{i+1} F_i \\ \hat{\lambda}_i E_{g,i}^\top E_{f,i} + G_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} H_{i+1} F_i \end{pmatrix}.
\end{aligned}$$

Substituting it back into Equation (5.20) leads to

$$\begin{pmatrix} \hat{\mathbf{x}}_{i|i+1} \\ \hat{\mathbf{u}}_{i|i+1} \end{pmatrix} = (\hat{T} + A^\top \hat{R}_{i+1}^{-1} A)^{-1} \left\{ \begin{pmatrix} F_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} \\ G_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} \end{pmatrix} \mathbf{y}_{i+1} + \begin{pmatrix} P_{i|i}^{-1} \\ 0 \end{pmatrix} \hat{\mathbf{x}}_{i|i} \right\}. \tag{5.21}$$

Substituting this into Problem (5.2), we have

$$\begin{aligned}
\hat{\mathbf{x}}_{i+1|i+1} &= F_i\hat{\mathbf{x}}_{i|i+1} + G_i\hat{\mathbf{u}}_{i|i+1} = [F_i,\, G_i] \begin{pmatrix} \hat{\mathbf{x}}_{i|i+1} \\ \hat{\mathbf{u}}_{i|i+1} \end{pmatrix} \\
&= [F_i,\, G_i](\hat{T} + A^\top \hat{R}_{i+1}^{-1} A)^{-1} \left\{ \begin{pmatrix} F_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} \\ G_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} \end{pmatrix} \mathbf{y}_{i+1} + \begin{pmatrix} P_{i|i}^{-1} \\ 0 \end{pmatrix} \hat{\mathbf{x}}_{i|i} \right\} \\
&= [F_i,\, G_i](\hat{T} + A^\top \hat{R}_{i+1}^{-1} A)^{-1} \begin{pmatrix} F_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} \\ G_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} \end{pmatrix} \mathbf{y}_{i+1} \\
&\qquad + [F_i,\, G_i](\hat{T} + A^\top \hat{R}_{i+1}^{-1} A)^{-1} \begin{pmatrix} P_{i|i}^{-1} \\ 0 \end{pmatrix} \hat{\mathbf{x}}_{i|i}
\end{aligned} \tag{5.22}$$

We compute the two term separately, the coefficient of $\mathbf{y}_{i+1}$ can be written as

$$\begin{aligned}
&[F_i,\, G_i](\hat{T} + A^\top \hat{R}_{i+1}^{-1} A)^{-1} \begin{pmatrix} F_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} \\ G_i^\top H_{i+1}^\top \hat{R}_{i+1}^{-1} \end{pmatrix} \\
=&[F_i,\, G_i] \left[\hat{T}^{-1} - \hat{T}^{-1}[F_i,\, G_i]^\top H_{i+1}^\top R_{e,i+1}^{-1} H_{i+1}[F_i,\, G_i]\hat{T}^{-1} \right] \begin{pmatrix} F_i^\top \\ G_i^\top \end{pmatrix} H_{i+1}^\top \hat{R}_{i+1}^{-1} \\
=&\left\{ [F_i,\, G_i]\hat{T}^{-1}[F_i,\, G_i]^\top - [F_i,\, G_i]\hat{T}^{-1}[F_i,\, G_i]^\top H_{i+1}^\top R_{e,i+1}^{-1} H_{i+1}[F_i,\, G_i]\hat{T}^{-1}[F_i,\, G_i]^\top \right\} \\
&\qquad \times H_{i+1}^\top \hat{R}_{i+1}^{-1} \\
=&(P_{i+1} - P_{i+1}H_{i+1}^\top R_{e,i+1}^{-1} H_{i+1} P_{i+1}) H_{i+1}^\top \hat{R}_{i+1}^{-1}.
\end{aligned} \tag{5.23}$$

The first equality holds from Equation (5.18), and the last equality holds from Equation (5.14).

The coefficient of $\hat{\mathbf{x}}_{i|i}$ can be written as:

$$\begin{aligned}
&[F_i,\, G_i](\hat{T} + A^\top \hat{R}_{i+1}^{-1} A)^{-1} \begin{pmatrix} P_{i|i}^{-1} \\ 0 \end{pmatrix} \\
=&[F_i,\, G_i] \left[\hat{T}^{-1} - \hat{T}^{-1}[F_i,\, G_i]^\top H_{i+1}^\top R_{e,i+1}^{-1} H_{i+1} [F_i,\, G_i]\hat{T}^{-1}\right] \begin{pmatrix} P_{i|i}^{-1} \\ 0 \end{pmatrix} \\
=&\left[I - [F_i,\, G_i]\hat{T}^{-1}[F_i,\, G_i]^\top H_{i+1}^\top R_{e,i+1}^{-1} H_{i+1}\right] [F_i,\, G_i]\hat{T}^{-1} \begin{pmatrix} P_{i|i}^{-1} \\ 0 \end{pmatrix} \\
=&\left[I - P_{i+1} H_{i+1}^\top R_{e,i+1}^{-1} H_{i+1}\right] [F_i,\, G_i]\hat{T}_i^{-1} \begin{pmatrix} P_{i|i}^{-1} \\ 0 \end{pmatrix}.
\end{aligned} \tag{5.24}$$

Notice by Equation (5.13), we have

$$\begin{aligned}
&[F_i,\, G_i]\hat{T}^{-1} \begin{pmatrix} P_{i|i}^{-1} \\ 0 \end{pmatrix} \\
=&[F_i,\, G_i] \begin{pmatrix} \hat{P}_{i|i} + \hat{P}_{i|i}\hat{\lambda}_i E_{f,i}^\top E_{g,i} \hat{Q}_i E_{g,i}^\top E_{f,i} \hat{\lambda}_i \hat{P}_{i|i} & -\hat{P}_{i|i}\hat{\lambda}_i E_{f,i}^\top E_{g,i} \hat{Q}_i \\ -\hat{Q}_i E_{g,i}^\top E_{f,i} \hat{\lambda}_i \hat{P}_{i|i} & \hat{Q}_i \end{pmatrix} \begin{pmatrix} P_{i|i}^{-1} \\ 0 \end{pmatrix} \\
=&\big(F_i \hat{P}_{i|i} + F_i \hat{P}_{i|i}\hat{\lambda}_i E_{f,i}^\top E_{g,i} \hat{Q}_i E_{g,i}^\top E_{f,i} \hat{\lambda}_i \hat{P}_{i|i} - G_i \hat{Q}_i E_{g,i}^\top E_{f,i} \hat{\lambda}_i \hat{P}_{i|i}\big) P_{i|i}^{-1} \\
=&\big(F_i - \hat{G}_i \hat{Q}_i E_{g,i}^\top E_{f,i} \hat{\lambda}_i\big) \hat{P}_{i|i} P_{i|i}^{-1} = \tilde{F}_i \hat{P}_{i|i} P_{i|i}^{-1},
\end{aligned} \tag{5.25}$$

where

$$\tilde{F}_i \triangleq F_i - \hat{\lambda}_i \hat{G}_i \hat{Q}_i E_{g,i}^\top E_{f,i}, \tag{5.26}$$

and the second last equality holds from Equation (5.15).

Recall definition of $\hat{P}_{i|i}$, we have

$$\begin{aligned}
&\hat{P}_{i|i} = \left(P_{i|i}^{-1} + \hat{\lambda}_i E_{f,i}^\top E_{f,i}\right)^{-1} \\
\Rightarrow\ & P_{i|i}^{-1} = \hat{P}_{i|i}^{-1} - \hat{\lambda}_i E_{f,i}^\top E_{f,i} \\
\Rightarrow\ & \hat{P}_{i|i} P_{i|i}^{-1} = \hat{P}_{i|i}(\hat{P}_{i|i}^{-1} - \hat{\lambda}_i E_{f,i}^\top E_{f,i}) = I - \hat{\lambda}_i \hat{P}_{i|i} E_{f,i}^\top E_{f,i} \\
\Rightarrow\ & \hat{F}_i \triangleq (F_i - \hat{\lambda}_i \hat{G}_i \hat{Q}_i E_{g,i}^\top E_{f,i})(I - \hat{\lambda}_i \hat{P}_{i|i} E_{f,i}^\top E_{f,i}) = \tilde{F}_i \hat{P}_{i|i} P_{i|i}^{-1}.
\end{aligned} \tag{5.27}$$

Substitute Equation (5.25) and Equation (5.27) into Equation (5.24), we have

$$[F_i,\, G_i](\hat{T} + A^\top \hat{R}_{i+1}^{-1} A)^{-1} \begin{pmatrix} P_{i|i}^{-1} \\ 0 \end{pmatrix} = \left[I - P_{i+1} H_{i+1}^\top R_{e,i+1}^{-1} H_{i+1}\right] \hat{F}_i. \tag{5.28}$$

Now substitute Equation (5.23) and Equation (5.28) into Equation (5.22), and denote

$$\hat{\mathbf{x}}_{i+1} \triangleq \hat{F}_i \hat{\mathbf{x}}_{i|i}; \quad \mathbf{e}_{i+1} \triangleq \mathbf{y}_{i+1} - H_{i+1}\hat{\mathbf{x}}_{i+1}; \quad P_{i+1|i+1} \triangleq P_{i+1} - P_{i+1} H_{i+1}^\top R_{e,i+1}^{-1} H_{i+1} P_{i+1}. \tag{5.29}$$

We have

$$\begin{aligned}
&\hat{\mathbf{x}}_{i+1|i+1} \\
=&(P_{i+1} - P_{i+1} H_{i+1}^\top R_{e,i+1}^{-1} H_{i+1} P_{i+1}) H_{i+1}^\top \hat{R}_{i+1}^{-1} \mathbf{y}_{i+1} + \left[I - P_{i+1} H_{i+1}^\top R_{e,i+1}^{-1} H_{i+1}\right] \hat{F}_i \hat{\mathbf{x}}_{i|i} \\
=&(P_{i+1} - P_{i+1} H_{i+1}^\top R_{e,i+1}^{-1} H_{i+1} P_{i+1}) H_{i+1}^\top \hat{R}_{i+1}^{-1} \left(\mathbf{e}_{i+1} + H_{i+1} \hat{F}_i \hat{\mathbf{x}}_{i|i}\right) \\
&\quad + \left[I - P_{i+1} H_{i+1}^\top R_{e,i+1}^{-1} H_{i+1}\right] \hat{F}_i \hat{\mathbf{x}}_{i|i} \\
=&P_{i+1|i+1} H_{i+1}^\top \hat{R}_{i+1}^{-1} \mathbf{e}_{i+1} + \left[I - P_{i+1} H_{i+1}^\top R_{e,i+1}^{-1} H_{i+1}\right] \left[I + P_{i+1} H_{i+1}^\top \hat{R}_{i+1}^{-1} H_{i+1}\right] \hat{F}_i \hat{\mathbf{x}}_{i|i} \\
=&P_{i+1|i+1} H_{i+1}^\top \hat{R}_{i+1}^{-1} \mathbf{e}_{i+1} \\
&\quad + \left[I - P_{i+1} H_{i+1}^\top (\hat{R}_{i+1} + H_{i+1} P_{i+1} H_{i+1}^\top)^{-1} H_{i+1}\right] \left[I + P_{i+1} H_{i+1}^\top \hat{R}_{i+1}^{-1} H_{i+1}\right] \hat{F}_i \hat{\mathbf{x}}_{i|i} \\
=&P_{i+1|i+1} H_{i+1}^\top \hat{R}_{i+1}^{-1} \mathbf{e}_{i+1} + \hat{F}_i \hat{\mathbf{x}}_{i|i} \\
=&P_{i+1|i+1} H_{i+1}^\top \hat{R}_{i+1}^{-1} \mathbf{e}_{i+1} + \hat{\mathbf{x}}_{i+1}.
\end{aligned} \tag{5.30}$$

The third equality follows form Equation (5.17), and the fourth equality holds from matrix inversion lemma.

Combining all the definitions and Equation (5.30), we get the following measurement-update form.

ALGORITHM 5.2. **Measurement-Update form**

(1) Initialize:

$$P_{0|0} := (\Pi_0^{-1} + H_0^\top R_0^{-1} H_0)^{-1}$$

$$\hat{\mathbf{x}}_{0|0} := P_{0|0} H_0^\top R_0^{-1} \mathbf{y}_0.$$

(2) Recursion:

Construct and minimize $G(\lambda)$ over $(\|M_i^\top H_{i+1}^\top R_{i+1}^{-1} H_{i+1} M_i\|, +\infty)$. Let the optimal value be λ_i^o. Compute the following values:

$$\hat{\lambda}_i := (1-\alpha)\lambda_i^o$$

$$\overline{R}_{i+1} := R_{i+1} - \lambda^{o-1} H_{i+1} M_i M_i^\top H_{i+1}^\top$$

$$\hat{R}_{i+1}^{-1} := \alpha R_{i+1}^{-1} + (1-\alpha)\overline{R}_{i+1}^{-1}$$

$$\hat{Q}_i^{-1} := Q_i^{-1} + \hat{\lambda}_i E_{g,i}^\top \left[I + \hat{\lambda}_i E_{f,i} P_{i|i} E_{f,i}^\top \right]^{-1} E_{g,i}$$

$$\hat{P}_{i|i} := (P_{i|i}^{-1} + \hat{\lambda}_i E_{f,i}^\top E_{f,i})^{-1} = P_{i|i} - P_{i|i} E_{f,i}^\top (\hat{\lambda}_i^{-1} I + E_{f,i} P_{i|i} E_{f,i}^\top)^{-1} E_{f,i} P_{i|i}$$

$$\hat{G}_i := G_i - \hat{\lambda}_i F_i \hat{P}_{i|i} E_{f,i}^\top E_{g,i}$$

$$\hat{F}_i := (F_i - \hat{\lambda}_i \hat{G}_i \hat{Q}_i E_{g,i}^\top E_{f,i})(I - \hat{\lambda}_i \hat{P}_{i|i} E_{f,i}^\top E_{f,i})$$

$$P_{i+1} := F_i \hat{P}_{i|i} F_i^\top + \hat{G}_i \hat{Q}_i \hat{G}_i^\top$$

$$R_{e,i+1} := \hat{R}_{i+1} + H_{i+1} P_{i+1} H_{i+1}^\top$$

$$P_{i+1|i+1} := P_{i+1} - P_{i+1} H_{i+1}^\top R_{e,i+1}^{-1} H_{i+1} P_{i+1}$$

$$\hat{\mathbf{x}}_{i+1} := \hat{F}_i \hat{\mathbf{x}}_{i|i}$$

$$\mathbf{e}_{i+1} := \mathbf{y}_{i+1} - H_{i+1} \hat{\mathbf{x}}_{i+1}$$

$$\hat{\mathbf{x}}_{i+1|i+1} := \hat{\mathbf{x}}_{i+1} + P_{i+1|i+1} H_{i+1}^\top \hat{R}_{i+1}^{-1} \mathbf{e}_{i+1}.$$

To derive the prediction form from the measurement-update form, we need the following two lemma.

LEMMA 5.5.

$$P_{i+1} = F_i P_i F_i^\top - \overline{K}_i \overline{R}_{e,i}^{-1} \overline{K}_i^\top + \hat{G}_i \hat{Q}_i \hat{G}_i^\top, \tag{5.31}$$

where

$$\overline{K}_i \triangleq F_i P_i \overline{H}_i^\top, \qquad \overline{R}_{e,i} \triangleq I + \overline{H}_i P_i \overline{H}_i^\top, \qquad \overline{H}_i \triangleq \begin{bmatrix} \hat{R}_i^{-1/2} H_i \\ \sqrt{\hat{\lambda}_i} E_{f,i} \end{bmatrix}.$$

PROOF. First note

$$\begin{aligned} P_{i|i}^{-1} =& (P_i - P_i H_i^\top R_{e,i}^{-1} H_i P_i)^{-1} = \Big(P_i - P_i H_i^\top (\hat{R}_i + H_i P_i H_i^\top)^{-1} H_i P_i\Big)^{-1} \\ =& \Big((P_i^{-1} + H_i^\top \hat{R}_i^{-1} H_i)^{-1}\Big)^{-1} = P_i^{-1} + H_i^\top \hat{R}_i^{-1} H_i. \end{aligned} \tag{5.32}$$

Hence we have

$$\begin{aligned} P_{i+1} =& F_i \hat{P}_{i|i} F_i^\top + \hat{G}_i \hat{Q}_i \hat{G}_i^\top = F_i \big(P_{i|i}^{-1} + \hat{\lambda}_i E_{f,i}^\top E_{f,i}\big)^{-1} F_i^\top + \hat{G}_i \hat{Q}_i \hat{G}_i^\top \\ =& F_i \big(P_i^{-1} + H_i^\top \hat{R}_i^{-1} H_i + \hat{\lambda}_i E_{f,i}^\top E_{f,i}\big)^{-1} F_i^\top + \hat{G}_i \hat{Q}_i \hat{G}_i^\top \\ =& F_i \big(P_i^{-1} + \overline{H}_i^\top \overline{H}_i\big)^{-1} F_i^\top + \hat{G}_i \hat{Q}_i \hat{G}_i \\ =& F_i P_i F_i^\top - \overline{K}_i \overline{R}_{e,i}^{-1} \overline{K}_i^\top + \hat{G}_i \hat{Q}_i \hat{G}_i^\top. \end{aligned}$$

□

LEMMA 5.6.

$$P_{i|i} H_i^\top \hat{R}_i^{-1} = P_i H_i^\top R_{e,i}^{-1}.$$

PROOF. From Equation (5.32) we have

$$\begin{aligned} & P_{i|i}^{-1} (P_i H_i^\top R_{e,i}^{-1}) = (P_i^{-1} + H_i^\top \hat{R}_i^{-1} H_i)(P_i H_i^\top R_{e,i}^{-1}) = H_i^\top (I + \hat{R}_i^{-1} H_i P_i H_i^\top) R_{e,i}^{-1} \\ =& H_i^\top \hat{R}_i^{-1} (\hat{R}_i + H_i P_i H_i^\top) R_{e,i}^{-1} = H_i^\top \hat{R}_i^{-1}. \end{aligned}$$

By left multiplying $P_{i|i}$ on both sides, the lemma follows. □

Substituting these two Lemmas into the Measurement-Update form, we get the recursive formula of the prediction form.

CHAPTER 6

Robustness and Regularization of Support Vector Machines

In Chapters 2- 5 we addressed two limitations of robust decision making, namely lack of theoretical justification and the conservatism in sequential decision making. From this chapter on, we will concentrate on exploring the relationship between robust decision making and machine learning. Note that machine learning tasks such as classification and regression can be recast as finding an optimal decision boundary with respect to an *unknown* probability distribution which can only be *approximated* by finitely many samples. Hence, it is natural to investigate how robustness may help in such decision tasks. Indeed, we will show in the sequel that robustness is *the reason* that makes learning algorithms work. We start from one of the most widely used classification algorithms – the support vector machines – in this chapter. In particular, we consider regularized support vector machines (SVMs) and show that they are precisely equivalent to a new robust optimization formulation. We show that this equivalence of robust optimization and regularization has implications for both algorithms, and analysis. In terms of algorithms, the equivalence suggests more general SVM-like algorithms for classification that explicitly build in protection to noise, and at the same time control overfitting. On the analysis front, the equivalence of robustness and regularization, provides a robust optimization interpretation for

the success of regularized SVMs. We use this new robustness interpretation of SVMs to give a new proof of consistency of (kernelized) SVMs, thus establishing robustness as the reason regularized SVMs generalize well. Part of the material of this chapter appears in [**168**].

6.1. Introduction

Support Vector Machines (SVMs for short) originated in [**31**] and can be traced back to as early as [**159**] and [**157**]. They continue to be one of the most successful algorithms for classification. SVMs address the classification problem by finding the hyperplane in the feature space that achieves maximum sample margin when the training samples are separable, which leads to minimizing the norm of the classifier. When the samples are not separable, a penalty term that approximates the total training error is considered [**14, 43**]. It is well known that minimizing the training error itself can lead to poor classification performance for new unlabeled data; that is, such an approach may have poor generalization error because of, essentially, overfitting [**158**]. A variety of modifications have been proposed to combat this problem, one of the most popular methods being that of minimizing a combination of the training-error and a regularization term. The latter is typically chosen as a norm of the classifier. The resulting regularized classifier performs better on new data. This phenomenon is often interpreted from a statistical learning theory view: the regularization term restricts the complexity of the classifier, hence the deviation of the testing error and the training error is controlled (see [**139, 70, 8, 99, 7**] and references therein).

In this chapter we consider a different setup, assuming that the training data are generated by the true underlying distribution, but some non-i.i.d. (potentially adversarial) disturbance is then added to the samples we observe. We follow a robust optimization (see [**64, 12, 22**] and references therein) approach, i.e., minimizing the worst possible empirical error under such disturbances. The use of robust optimization in classification is not new (e.g., [**137, 27, 101**]). Robust classification

models studied in the past have considered only box-type uncertainty sets, which allow the possibility that the data have all been skewed in some non-neutral manner by a correlated disturbance. This has made it difficult to obtain non-conservative generalization bounds. Moreover, there has not been an explicit connection to the regularized classifier, although at a high-level it is known that regularization and robust optimization are related (e.g., [**64, 2**]). The main contribution in this chapter is solving the robust classification problem for a class of non-box-type uncertainty sets, and providing a linkage between robust classification and the standard regularization scheme of SVMs. In particular, our contributions include the following:

- We solve the robust SVM formulation for a class of non-box-type uncertainty sets. This permits finer control of the adversarial disturbance, restricting it to satisfy aggregate constraints across data points, therefore reducing the possibility of highly correlated disturbances.
- We show that the standard regularized SVM classifier is a special case of our robust classification, thus explicitly relating robustness and regularization. This provides an alternative explanation to the success of regularization, and also suggests new physically motivated ways to construct regularization terms.
- We relate our robust formulation to several probabilistic formulations. We consider a chance-constrained classifier (i.e., a classifier with probabilistic constraints on misclassification) and show that our robust formulation can approximate it far less conservatively than previous robust formulations could possibly do. We also consider a Bayesian setup, and show that this can be used to provide a principled means of selecting the regularization coefficient without cross-validation.
- We show that the robustness perspective, stemming from a non-i.i.d. analysis, can be useful in the standard learning (i.i.d.) setup, by using it to prove consistency for standard SVM classification, *without using VC-dimension or stability arguments*. This result implies that generalization ability is a

direct result of robustness to local disturbances; it therefore suggests a new justification for good performance, and consequently allows us to construct learning algorithms that generalize well by robustifying non-consistent algorithms.

Robustness and Regularization. We comment here on the explicit equivalence of robustness and regularization. We briefly explain how this observation is different from previous work and why it is interesting. Certain equivalence relationships between robustness and regularization have been established for problems other than classification [**64, 12, 30**], but these results do not directly apply to the classification problem. Indeed, research on classifier regularization mainly discusses its effect on bounding the complexity of the function class (e.g., [**139, 70, 8, 99, 7**]). Meanwhile, research on robust classification has not attempted to relate robustness and regularization (e.g, [**101, 26, 27, 137, 150, 79**]), in part due to the robustness formulations used in those papers. In fact, they all consider robustified versions of *regularized* classifiers.[1] [**25**] considers a robust formulation for box-type uncertainty, and relates this robust formulation with regularized SVM. However, this formulation involves a non-standard loss function that does not bound the $0-1$ loss, and hence its physical interpretation is not clear.

The connection of robustness and regularization in the SVM context is important for the following reasons. First, it gives an alternative and potentially powerful explanation of the generalization ability of the regularization term. In the classical machine learning literature, the regularization term bounds the complexity of the class of classifiers. The robust view of regularization regards the testing samples as a perturbed copy of the training samples. We show that when the total perturbation is given or bounded, the regularization term bounds the gap between the classification errors of the SVM on these two sets of samples. In contrast to the standard PAC approach, this bound depends neither on how rich the class of candidate classifiers is, nor on an assumption that all samples are picked in an i.i.d. manner. In addition,

[1] [**101**] is perhaps the only exception, where a regularization term is added to the covariance estimation rather than to the objective function.

this suggests novel approaches to designing good classification algorithms, in particular, designing the regularization term. In the PAC structural-risk minimization approach, regularization is chosen to minimize a bound on the generalization error based on the training error and a complexity term. This complexity term typically leads to overly emphasizing the regularizer, and indeed this approach is known to often be too pessimistic [**97**] for problems with more structure. The robust approach offers another avenue. Since both noise and robustness are physical processes, a close investigation of the application and noise characteristics at hand, can provide insights into how to properly robustify, and therefore regularize the classifier. For example, it is known that normalizing the samples so that the variance among all features is roughly the same (a process commonly used to eliminate the scaling freedom of individual features) often leads to good generalization performance. From the robustness perspective, this simply says that the noise is anisotropic (ellipsoidal) rather than spherical, and hence an appropriate robustification must be designed to fit this anisotropy.

We also show that using the robust optimization viewpoint, we obtain some probabilistic results outside the PAC setup. In Section 6.3 we bound the probability that a noisy training sample is correctly labeled. Such a bound considers the behavior of *corrupted* samples and is hence different from the known PAC bounds. This is helpful when the training samples and the testing samples are drawn from different distributions, or some adversary manipulates the samples to prevent them from being correctly labeled (e.g., spam senders change their patterns from time to time to avoid being labeled and filtered). Finally, this connection of robustification and regularization also provides us with new proof techniques as well (see Section 6.5).

We need to point out that there are several different definitions of robustness in the literature. In this thesis, as well as the aforementioned robust classification papers, robustness is mainly understood from a Robust Optimization perspective, where a min-max optimization is performed over all possible disturbances. An alternative interpretation of robustness stems from the rich literature on Robust Statistics (e.g.,

[**90, 85, 129, 108**]), which studies how an estimator or algorithm behaves under a small perturbation of the statistics model. For example, the Influence Function approach, proposed in [**84**] and [**85**], measures the impact of an infinitesimal amount of contamination of the original distribution on the quantity of interest. Based on this notion of robustness, [**39**] showed that many kernel classification algorithms, including SVM, are robust in the sense of having a finite Influence Function. A similar result for regression algorithms is shown in [**40**] for smooth loss functions, and in [**41**] for non-smooth loss functions where a relaxed version of the Influence Function is applied. In the machine learning literature, another widely used notion closely related to robustness is the *stability*, where an algorithm is required to be robust (in the sense that the output function does not change significantly) under a specific perturbation: deleting one sample from the training set. It is now well known that a stable algorithm such as SVM has desirable generalization properties, and is statistically consistent under mild technical conditions; see for example [**32, 100, 122, 112**] for details. One main difference between Robust Optimization and other robustness notions is that the former is constructive rather than analytical. That is, in contrast to robust statistics or the stability approach that measures the robustness of a *given* algorithm, Robust Optimization can *robustify* an algorithm: it converts a given algorithm to a robust one. For example, as we show in this chapter, the RO version of a naive empirical-error minimization is the well known SVM. As a constructive process, the RO approach also leads to additional flexibility in algorithm design, especially when the nature of the perturbation is known or can be well estimated.

This chapter is organized as follows. In Section 6.2 we investigate the correlated disturbance case, and show the equivalence between robust classification and regularization. We develop the connections to probabilistic formulations in Section 6.3. The kernelized version is investigated in Section 6.4, based on which a consistency result following a robustness analysis is developed in Section 6.5. Some concluding remarks are given in Section 6.6.

NOTATION: Capital letters are used to denote matrices, and boldface letters are used to denote column vectors. For a given norm $\|\cdot\|$, we use $\|\cdot\|^*$ to denote its dual norm, i.e., $\|\mathbf{z}\|^* \triangleq \sup\{\mathbf{z}^\top\mathbf{x} | \|\mathbf{x}\| \leq 1\}$. For a vector $\mathbf{x}$ and a positive semi-definite matrix C of the same dimension, $\|\mathbf{x}\|_C$ denotes $\sqrt{\mathbf{x}^\top C\mathbf{x}}$. We use δ to denote disturbance affecting the samples. We use superscript r to denote the true value for an uncertain variable, so that $\boldsymbol{\delta}_i^r$ is the true (but unknown) noise of the i^{th} sample. The set of non-negative scalars is denoted by $\mathbb{R}^+$. The set of integers from 1 to n is denoted by $[1:n]$.

6.2. Robust classification and regularization

We consider the standard binary classification problem, where we are given a finite number of training samples $\{\mathbf{x}_i, y_i\}_{i=1}^m \subseteq \mathbb{R}^n \times \{-1, +1\}$, and must find a linear classifier, specified by the function $h^{\mathbf{w},b}(\mathbf{x}) = \text{sgn}(\langle \mathbf{w}, \mathbf{x}\rangle + b)$. For the standard regularized classifier, the parameters $(\mathbf{w}, b)$ are obtained by solving the following convex optimization problem:

$$\begin{aligned} \min_{\mathbf{w},b,\boldsymbol{\xi}}: \quad & r(\mathbf{w},b) + \sum_{i=1}^m \xi_i \\ \text{s.t.}: \quad & \xi_i \geq \left[1 - y_i(\langle \mathbf{w}, \mathbf{x}_i\rangle + b)\right] \\ & \xi_i \geq 0, \end{aligned}$$

where $r(\mathbf{w}, b)$ is a regularization term. This is equivalent to

$$\min_{\mathbf{w},b} \left\{ r(\mathbf{w},b) + \sum_{i=1}^m \max\left[1 - y_i(\langle \mathbf{w}, \mathbf{x}_i\rangle + b), 0\right] \right\}.$$

Previous robust classification work [**137, 26, 27, 25, 150**] considers the classification problem where the input are subject to (unknown) disturbances $\vec{\boldsymbol{\delta}} = (\boldsymbol{\delta}_1, \ldots, \boldsymbol{\delta}_m)$ and essentially solves the following min-max problem:

$$\min_{\mathbf{w},b} \max_{\vec{\boldsymbol{\delta}} \in \mathcal{N}_{\text{box}}} \left\{ r(\mathbf{w},b) + \sum_{i=1}^m \max\left[1 - y_i(\langle \mathbf{w}, \mathbf{x}_i - \boldsymbol{\delta}_i\rangle + b), 0\right] \right\}, \tag{6.1}$$

for a box-type uncertainty set $\mathcal{N}_{\text{box}}$. That is, let $\mathcal{N}_i$ denote the projection of $\mathcal{N}_{\text{box}}$ onto the $\boldsymbol{\delta}_i$ component, then $\mathcal{N}_{\text{box}} = \mathcal{N}_1 \times \cdots \times \mathcal{N}_m$. Effectively, this allows simultaneous worst-case disturbances across many samples, and leads to overly conservative solutions. The goal of this paper is to obtain a robust formulation where the disturbances $\{\boldsymbol{\delta}_i\}$ may be meaningfully taken to be correlated, i.e., to solve for a non-box-type $\mathcal{N}$:

$$\min_{\mathbf{w},b} \max_{\vec{\boldsymbol{\delta}} \in \mathcal{N}} \left\{ r(\mathbf{w}, b) + \sum_{i=1}^{m} \max \left[1 - y_i(\langle \mathbf{w}, \mathbf{x}_i - \boldsymbol{\delta}_i \rangle + b), 0 \right] \right\}. \tag{6.2}$$

We briefly explain here the four reasons that motivate this "robust to perturbation" setup and in particular the min-max form of (6.1) and (6.2). First, it can explicitly incorporate prior problem knowledge of local invariance (e.g., [**145**]). For example, in vision tasks, a desirable classifier should provide a consistent answer if an input image slightly changes. Second, there are situations where some adversarial opponents (e.g., spam senders) will manipulate the testing samples to avoid being correctly classified, and the robustness toward such manipulation should be taken into consideration in the training process [**79**]. Or alternatively, the training samples and the testing samples can be obtained from different processes and hence the standard i.i.d. assumption is violated [**28**]. For example in real-time applications, the newly generated samples are often less accurate due to time constraints. Finally, formulations based on chance-constraints [**27, 137**] are mathematically equivalent to such a min-max formulation.

We define explicitly the correlated disturbance (or uncertainty) which we study below.

DEFINITION 6.1. *A set* $\mathcal{N}_0 \subseteq \mathbb{R}^n$ *is called an* Atomic Uncertainty Set *if*

(I) $\mathbf{0} \in \mathcal{N}_0$;

(II) *For any* $\mathbf{w}_0 \in \mathbb{R}^n$: $\sup_{\boldsymbol{\delta} \in \mathcal{N}_0} [\mathbf{w}_0^\top \boldsymbol{\delta}] = \sup_{\boldsymbol{\delta}' \in \mathcal{N}_0} [-\mathbf{w}_0^\top \boldsymbol{\delta}'] < +\infty$.

We use "sup" here because the maximal value is not necessary attained since $\mathcal{N}_0$ may not be a closed set. The second condition of Atomic Uncertainty set basically says that the uncertainty set is bounded and symmetric. In particular, all norm balls

and ellipsoids centered at the origin are atomic uncertainty sets, while an arbitrary polytope might not be an atomic uncertainty set.

DEFINITION 6.2. *Let $\mathcal{N}_0$ be an atomic uncertainty set. A set $\mathcal{N} \subseteq \mathbb{R}^{n\times m}$ is called a* Sublinear Aggregated Uncertainty Set *of $\mathcal{N}_0$, if*

$$\mathcal{N}^- \subseteq \mathcal{N} \subseteq \mathcal{N}^+,$$

$$\textit{where:}\quad \mathcal{N}^- \triangleq \bigcup_{t=1}^{m} \mathcal{N}_t^-; \qquad \mathcal{N}_t^- \triangleq \{(\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m) | \boldsymbol{\delta}_t \in \mathcal{N}_0;\ \boldsymbol{\delta}_{i\neq t} = \mathbf{0}\}.$$

$$\mathcal{N}^+ \triangleq \{(\alpha_1\boldsymbol{\delta}_1, \cdots, \alpha_m\boldsymbol{\delta}_m) | \sum_{i=1}^{m} \alpha_i = 1;\ \alpha_i \geq 0,\ \boldsymbol{\delta}_i \in \mathcal{N}_0,\ i = 1, \cdots, m\}.$$

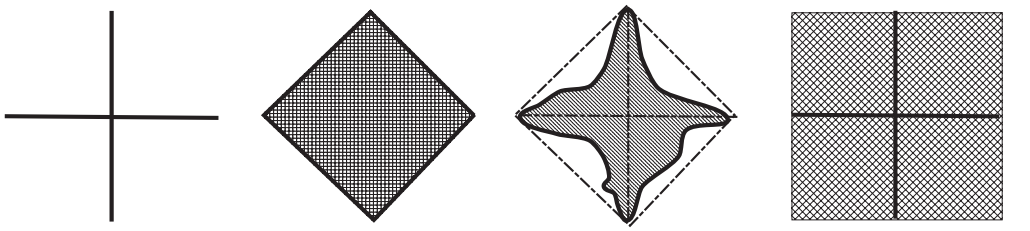
The Sublinear Aggregated Uncertainty definition models the case where the disturbances on each sample are treated identically, but their aggregate behavior across multiple samples is controlled. Some interesting examples include

$$(1)\quad \{(\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m) | \sum_{i=1}^{m} \|\boldsymbol{\delta}_i\| \leq c\};$$

$$(2)\quad \{(\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m) | \exists t \in [1:m];\ \|\boldsymbol{\delta}_t\| \leq c;\ \boldsymbol{\delta}_i = \mathbf{0}, \forall i \neq t\};$$

$$(3)\quad \{(\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m) | \sum_{i=1}^{m} \sqrt{c\|\boldsymbol{\delta}_i\|} \leq c\}.$$

All these examples have the same atomic uncertainty set $\mathcal{N}_0 = \left\{\boldsymbol{\delta} \middle| \|\boldsymbol{\delta}\| \leq c\right\}$. Figure 6.1 provides an illustration of a sublinear aggregated uncertainty set for $n = 1$ and $m = 2$, i.e., the training set consists of two univariate samples.



a. $\mathcal{N}^-$ b. $\mathcal{N}^+$ c. $\mathcal{N}$ d. Box uncertainty

FIGURE 6.1. Illustration of a *Sublinear Aggregated Uncertainty Set $\mathcal{N}$*.

THEOREM 6.1. *Assume* $\{\mathbf{x}_i, y_i\}_{i=1}^m$ *are non-separable,* $r(\cdot) : \mathbb{R}^{n+1} \to \mathbb{R}$ *is an arbitrary function,* $\mathcal{N}$ *is a Sublinear Aggregated Uncertainty set with corresponding atomic uncertainty set* $\mathcal{N}_0$*. Then the following min-max problem*

$$\min_{\mathbf{w},b} \sup_{(\boldsymbol{\delta}_1,\cdots,\boldsymbol{\delta}_m)\in\mathcal{N}} \left\{ r(\mathbf{w},b) + \sum_{i=1}^m \max\left[1 - y_i(\langle \mathbf{w}, \mathbf{x}_i - \boldsymbol{\delta}_i\rangle + b),\, 0\right] \right\} \tag{6.3}$$

is equivalent to the following optimization problem on $\mathbf{w}, b, \boldsymbol{\xi}$*:*

$$\begin{aligned} \min: \quad & r(\mathbf{w},b) + \sup_{\boldsymbol{\delta}\in\mathcal{N}_0} (\mathbf{w}^\top \boldsymbol{\delta}) + \sum_{i=1}^m \xi_i, \\ \text{s.t.}: \quad & \xi_i \geq 1 - [y_i(\langle \mathbf{w},\, \mathbf{x}_i\rangle + b)], \quad i = 1,\ldots,m; \\ & \xi_i \geq 0, \quad i = 1,\ldots,m. \end{aligned} \tag{6.4}$$

Furthermore, the minimum in Problem (6.4) is attainable when $r(\cdot,\cdot)$ *is lower semi-continuous.*

PROOF. Define:

$$v(\mathbf{w},b) \triangleq \sup_{\boldsymbol{\delta}\in\mathcal{N}_0} (\mathbf{w}^\top \boldsymbol{\delta}) + \sum_{i=1}^m \max\left[1 - y_i(\langle \mathbf{w}, \mathbf{x}_i\rangle + b),\, 0\right].$$

Recall that $\mathcal{N}^- \subseteq \mathcal{N} \subseteq N^+$ by definition. Hence, fixing any $(\hat{\mathbf{w}}, \hat{b}) \in \mathbb{R}^{n+1}$, the following inequalities hold:

$$\begin{aligned} & \sup_{(\boldsymbol{\delta}_1,\cdots,\boldsymbol{\delta}_m)\in\mathcal{N}^-} \sum_{i=1}^m \max\left[1 - y_i(\langle \hat{\mathbf{w}}, \mathbf{x}_i - \boldsymbol{\delta}_i\rangle + \hat{b}),\, 0\right] \\ \leq & \sup_{(\boldsymbol{\delta}_1,\cdots,\boldsymbol{\delta}_m)\in\mathcal{N}} \sum_{i=1}^m \max\left[1 - y_i(\langle \hat{\mathbf{w}}, \mathbf{x}_i - \boldsymbol{\delta}_i\rangle + \hat{b}),\, 0\right] \\ \leq & \sup_{(\boldsymbol{\delta}_1,\cdots,\boldsymbol{\delta}_m)\in\mathcal{N}^+} \sum_{i=1}^m \max\left[1 - y_i(\langle \hat{\mathbf{w}}, \mathbf{x}_i - \boldsymbol{\delta}_i\rangle + \hat{b}),\, 0\right]. \end{aligned}$$

To prove the theorem, we first show that $v(\hat{\mathbf{w}}, \hat{b})$ is no larger than the leftmost expression and then show $v(\hat{\mathbf{w}}, \hat{b})$ is no smaller than the rightmost expression.

Step 1: We prove that

$$v(\hat{\mathbf{w}}, \hat{b}) \leq \sup_{(\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m) \in \mathcal{N}^-} \sum_{i=1}^{m} \max\big[1 - y_i(\langle \hat{\mathbf{w}}, \mathbf{x}_i - \boldsymbol{\delta}_i \rangle + \hat{b}),\, 0\big]. \tag{6.5}$$

Since the samples $\{\mathbf{x}_i,\, y_i\}_{i=1}^m$ are not separable, there exists $t \in [1:m]$ such that

$$y_t(\langle \hat{\mathbf{w}}, \mathbf{x}_t \rangle + \hat{b}) < 0. \tag{6.6}$$

Hence,

$$\begin{aligned}
&\sup_{(\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m) \in \mathcal{N}_t^-} \sum_{i=1}^{m} \max\big[1 - y_i(\langle \hat{\mathbf{w}}, \mathbf{x}_i - \boldsymbol{\delta}_i \rangle + \hat{b}),\, 0\big] \\
&= \sum_{i \neq t} \max\big[1 - y_i(\langle \hat{\mathbf{w}}, \mathbf{x}_i \rangle + \hat{b}),\, 0\big] + \sup_{\boldsymbol{\delta}_t \in \mathcal{N}_0} \max\big[1 - y_t(\langle \hat{\mathbf{w}}, \mathbf{x}_t - \boldsymbol{\delta}_t \rangle + \hat{b}),\, 0\big] \\
&= \sum_{i \neq t} \max\big[1 - y_i(\langle \hat{\mathbf{w}}, \mathbf{x}_i \rangle + \hat{b}),\, 0\big] + \max\big[1 - y_t(\langle \hat{\mathbf{w}}, \mathbf{x}_t \rangle + \hat{b}) + \sup_{\boldsymbol{\delta}_t \in \mathcal{N}_0} (y_t \hat{\mathbf{w}}^\top \boldsymbol{\delta}_t),\, 0\big] \\
&= \sum_{i \neq t} \max\big[1 - y_i(\langle \hat{\mathbf{w}}, \mathbf{x}_i \rangle + \hat{b}),\, 0\big] + \max\big[1 - y_t(\langle \hat{\mathbf{w}}, \mathbf{x}_t \rangle + \hat{b}),\, 0\big] + \sup_{\boldsymbol{\delta}_t \in \mathcal{N}_0} (y_t \hat{\mathbf{w}}^\top \boldsymbol{\delta}_t) \\
&= \sup_{\boldsymbol{\delta} \in \mathcal{N}_0} (\hat{\mathbf{w}}^\top \boldsymbol{\delta}) + \sum_{i=1}^{m} \max\big[1 - y_i(\langle \hat{\mathbf{w}}, \mathbf{x}_i \rangle + \hat{b}), 0\big] = v(\hat{\mathbf{w}}, \hat{b}).
\end{aligned}$$

The third equality holds because of Inequality (6.6) and $\sup_{\boldsymbol{\delta}_t \in \mathcal{N}_0}(y_t \hat{\mathbf{w}}^\top \boldsymbol{\delta}_t)$ being non-negative (recall $\mathbf{0} \in \mathcal{N}_0$). Since $\mathcal{N}_t^- \subseteq \mathcal{N}^-$, Inequality (6.5) follows.

Step 2: Next we prove that

$$\sup_{(\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m) \in \mathcal{N}^+} \sum_{i=1}^{m} \max\big[1 - y_i(\langle \hat{\mathbf{w}}, \mathbf{x}_i - \boldsymbol{\delta}_i \rangle + \hat{b}),\, 0\big] \leq v(\hat{\mathbf{w}}, \hat{b}). \tag{6.7}$$

Notice that by the definition of $\mathcal{N}^+$ we have

$$\begin{aligned}
&\sup_{(\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m) \in \mathcal{N}^+} \sum_{i=1}^{m} \max\big[1 - y_i(\langle \hat{\mathbf{w}}, \mathbf{x}_i - \boldsymbol{\delta}_i \rangle + \hat{b}),\, 0\big] \\
&= \sup_{\sum_{i=1}^m \alpha_i = 1;\, \alpha_i \geq 0;\, \hat{\boldsymbol{\delta}}_i \in \mathcal{N}_0} \sum_{i=1}^{m} \max\big[1 - y_i(\langle \hat{\mathbf{w}}, \mathbf{x}_i - \alpha_i \hat{\boldsymbol{\delta}}_i \rangle + \hat{b}),\, 0\big] \qquad (6.8) \\
&= \sup_{\sum_{i=1}^m \alpha_i = 1;\, \alpha_i \geq 0;} \sum_{i=1}^{m} \max\big[\sup_{\hat{\boldsymbol{\delta}}_i \in \mathcal{N}_0} \big(1 - y_i(\langle \hat{\mathbf{w}}, \mathbf{x}_i - \alpha_i \hat{\boldsymbol{\delta}}_i \rangle + \hat{b})\big),\, 0\big].
\end{aligned}$$

Now, for any $i \in [1:m]$, the following holds,

$$\begin{aligned}
&\max\big[\sup_{\hat{\boldsymbol{\delta}}_i \in \mathcal{N}_0} \big(1 - y_i(\langle \hat{\mathbf{w}}, \mathbf{x}_i - \alpha_i \hat{\boldsymbol{\delta}}_i \rangle + \hat{b})\big),\, 0\big] \\
=&\max\big[1 - y_i(\langle \hat{\mathbf{w}}, \mathbf{x}_i \rangle + \hat{b}) + \alpha_i \sup_{\hat{\boldsymbol{\delta}}_i \in \mathcal{N}_0} (\hat{\mathbf{w}}^\top \hat{\boldsymbol{\delta}}_i),\, 0\big] \\
\leq&\max\big[1 - y_i(\langle \hat{\mathbf{w}}, \mathbf{x}_i \rangle + \hat{b}),\, 0\big] + \alpha_i \sup_{\hat{\boldsymbol{\delta}}_i \in \mathcal{N}_0} (\hat{\mathbf{w}}^\top \hat{\boldsymbol{\delta}}_i).
\end{aligned}$$

Therefore, Equation (6.8) is upper bounded by

$$\begin{aligned}
&\sum_{i=1}^{m} \max\big[1 - y_i(\langle \hat{\mathbf{w}}, \mathbf{x}_i \rangle + \hat{b}),\, 0\big] + \sup_{\sum_{i=1}^m \alpha_i = 1;\, \alpha_i \geq 0;} \sum_{i=1}^{m} \alpha_i \sup_{\hat{\boldsymbol{\delta}}_i \in \mathcal{N}_0} (\hat{\mathbf{w}}^\top \hat{\boldsymbol{\delta}}_i) \\
=&\sup_{\boldsymbol{\delta} \in \mathcal{N}_0} (\hat{\mathbf{w}}^\top \boldsymbol{\delta}) + \sum_{i=1}^{m} \max\big[1 - y_i(\langle \hat{\mathbf{w}}, \mathbf{x}_i \rangle + \hat{b}), 0\big] = v(\hat{\mathbf{w}}, \hat{b}),
\end{aligned}$$

hence Inequality (6.7) holds.

Step 3: Combining the two steps and adding $r(\mathbf{w}, b)$ on both sides leads to: $\forall (\mathbf{w}, b) \in \mathbb{R}^{n+1}$,

$$\sup_{(\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_{\boldsymbol{m}}) \in \mathcal{N}} \sum_{i=1}^{m} \max\big[1 - y_i(\langle \mathbf{w}, \mathbf{x}_i - \boldsymbol{\delta}_i \rangle + b),\, 0\big] + r(\mathbf{w}, b) = v(\mathbf{w}, b) + r(\mathbf{w}, \mathbf{b}).$$

Taking the infimum on both sides establishes the equivalence of Problem (6.3) and Problem (6.4). Observe that $\sup_{\boldsymbol{\delta} \in \mathcal{N}_0} \mathbf{w}^\top \boldsymbol{\delta}$ is a supremum over a class of affine functions, and hence is lower semi-continuous. Therefore $v(\cdot, \cdot)$ is also lower semi-continuous. Thus the minimum can be achieved for Problem (6.4), and Problem (6.3) by equivalence, when $r(\cdot)$ is lower semi-continuous. □

This theorem reveals the main difference between Formulation (6.1) and our formulation in (6.2). Consider a Sublinear Aggregated Uncertainty set

$$\mathcal{N} = \{(\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m) | \sum_{i=1}^{m} \|\boldsymbol{\delta}_i\| \leq c\}.$$

The smallest box-type uncertainty set containing $\mathcal{N}$ includes disturbances with norm sum up to mc. Therefore, it leads to a regularization coefficient as large as mc that is linked to the number of training samples, and will therefore be overly conservative.

An immediate corollary is that a special case of our robust formulation is equivalent to the norm-regularized SVM setup:

COROLLARY 6.2. *Let* $\mathcal{T} \triangleq \Big\{(\boldsymbol{\delta}_1, \cdots \boldsymbol{\delta}_m) | \sum_{i=1}^m \|\boldsymbol{\delta}_i\|^* \leq c\Big\}$. *If the training sample* $\{\mathbf{x}_i, y_i\}_{i=1}^m$ *are non-separable, then the following two optimization problems on* $(\mathbf{w}, b)$ *are equivalent*[2]

$$\min : \qquad \max_{(\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m) \in \mathcal{T}} \sum_{i=1}^m \max\big[1 - y_i\big(\langle \mathbf{w},\, \mathbf{x}_i - \boldsymbol{\delta}_i\rangle + b\big), 0\big], \tag{6.9}$$

$$\min : \qquad c\|\mathbf{w}\| + \sum_{i=1}^m \max\big[1 - y_i\big(\langle \mathbf{w},\, \mathbf{x}_i\rangle + b\big), 0\big]. \tag{6.10}$$

PROOF. Let $\mathcal{N}_0$ be the dual-norm ball $\{\boldsymbol{\delta} | \|\boldsymbol{\delta}\|^* \leq c\}$ and $r(\mathbf{w}, b) \equiv 0$. Then $\sup_{\|\boldsymbol{\delta}\|^* \leq c}(\mathbf{w}^\top \boldsymbol{\delta}) = c\|\mathbf{w}\|$. The corollary follows from Theorem 6.1. Notice that indeed the equivalence holds for any $\mathbf{w}$ and b. □

This corollary explains the widely known fact that the regularized classifier tends to be more robust. Specifically, it explains the observation that when the disturbance is noise-like and neutral rather than adversarial, a norm-regularized classifier (without any robustness requirement) has a performance often superior to a *box-type* robust classifier [**150**]. On the other hand, this observation also suggests that the appropriate way to regularize should come from a disturbance-robustness perspective. The above equivalence implies that standard regularization essentially assumes that the disturbance is spherical; if this is not true, robustness may yield a better regularization-like algorithm. To find a more effective regularization term, a closer investigation of the data variation is desirable, e.g., by examining the variation of the data and solving the corresponding robust classification problem. For example, one way to regularize is by splitting the given training samples into two subsets with

[2]After a journal version of this chapter [**168**] is submitted, the author was informed that the optimization equivalence for the linear case was observed independently by [**19**].

an equal number of elements, and treating one as a disturbed copy of the other. By analyzing the direction of the disturbance and the magnitude of the total variation, one can choose the proper norm to use, and a suitable tradeoff parameter.

6.3. Probabilistic interpretations

Although Problem (6.3) is formulated without any probabilistic assumptions, in this section, we briefly explain two approaches to construct the uncertainty set and equivalently tune the regularization parameter c based on probabilistic information.

The first approach is to use Problem (6.3) to approximate an upper bound for a chance-constrained classifier. Suppose the disturbance $(\boldsymbol{\delta}_1^r, \cdots \boldsymbol{\delta}_m^r)$ follows a joint probability measure μ. Then the chance-constrained classifier is given by the following minimization problem given a confidence level $\eta \in [0,\, 1]$,

$$\begin{aligned} \min_{\mathbf{w},b,l} :\quad & l \\ \text{s.t.} :\quad & \mu\Big\{ \sum_{i=1}^{m} \max\left[1 - y_i(\langle \mathbf{w},\, \mathbf{x}_i - \boldsymbol{\delta}_i^r \rangle + b), 0\right] \leq l \Big\} \geq 1 - \eta. \end{aligned} \tag{6.11}$$

The formulations in [**137**], [**101**] and [**26**] assume uncorrelated noise and require all constraints to be satisfied with high probability *simultaneously*. They find a vector $[\xi_1, \cdots, \xi_m]^\top$ where each ξ_i is the η-quantile of the hinge-loss for sample $\mathbf{x}_i^r$. In contrast, our formulation above minimizes the η-quantile of the average (or equivalently the sum of) empirical error. When controlling this average quantity is of more interest, the box-type noise formulation will be overly conservative.

Problem (6.11) is generally intractable. However, we can approximate it as follows. Let

$$c^* \triangleq \inf\{\alpha | \mu(\sum_i \|\boldsymbol{\delta}_i\|^* \leq \alpha) \geq 1 - \eta\}.$$

Notice that c^* is easily calculate using simulation given μ. Then for any $(\mathbf{w}, b)$, with probability no less than $1-\eta$, the following holds,

$$\begin{aligned}&\sum_{i=1}^{m} \max\left[1 - y_i(\langle \mathbf{w},\, \mathbf{x}_i - \boldsymbol{\delta}_i^r\rangle + b), 0\right]\\ \leq &\max_{\sum_i \|\boldsymbol{\delta}_i\|^* \leq c^*} \sum_{i=1}^{m} \max\left[1 - y_i(\langle \mathbf{w},\, \mathbf{x}_i - \boldsymbol{\delta}_i\rangle + b), 0\right].\end{aligned}$$

Thus (6.11) is upper bounded by (6.10) with $c = c^*$. This gives an additional probabilistic robustness property of the standard regularized classifier. Notice that following a similar approach but with the constraint-wise robust setup, i.e., the box uncertainty set, would lead to considerably more pessimistic approximations of the chance constraint.

The second approach considers a Bayesian setup. Suppose the total disturbance $c^r \triangleq \sum_{i=1}^m \|\boldsymbol{\delta}_i^r\|^*$ follows a prior distribution $\rho(\cdot)$. This can model for example the case where the training sample set is a mixture of several data sets where the disturbance magnitude of each set is known. Such a setup leads to the following classifier which minimizes the Bayesian (robust) error:

$$\min_{\mathbf{w},b}: \quad \int \left\{ \max_{\sum \|\boldsymbol{\delta}_i\|^* \leq c} \sum_{i=1}^{m} \max\left[1 - y_i\big(\langle \mathbf{w},\, \mathbf{x}_i - \boldsymbol{\delta}_i\rangle + b\big), 0\right] \right\} d\rho(c). \tag{6.12}$$

By Corollary 6.2, the Bayesian classifier (6.12) is equivalent to

$$\min_{\mathbf{w},b}: \quad \int \left\{ c\|\mathbf{w}\| + \sum_{i=1}^{m} \max\left[1 - y_i\big(\langle \mathbf{w},\, \mathbf{x}_i\rangle + b\big), 0\right] \right\} d\rho(c),$$

which can be further simplified as

$$\min_{\mathbf{w},b}: \quad \overline{c}\|\mathbf{w}\| + \sum_{i=1}^{m} \max\left[1 - y_i\big(\langle \mathbf{w},\, \mathbf{x}_i\rangle + b\big), 0\right],$$

where $\overline{c} \triangleq \int c\, d\rho(c)$. This thus provides a justifiable parameter tuning method different from cross validation: simply using the expected value of c^r. We note that it is the equivalence in Corollary 6.2 that makes this possible, since it is difficult to imagine a setting where one would have a prior on regularization coefficients.

6.4. Kernelization

The previous results can be easily generalized to the kernelized setting, which we discuss in detail in this section. In particular, similar to the linear classification case, we give a new interpretation of the standard kernelized SVM as the min-max empirical hinge-loss solution, where the disturbance is assumed to lie in the feature space. We then relate this to the (more intuitively appealing) setup where the disturbance lies in the sample space. We use this relationship in Section 6.5 to prove a consistency result for kernelized SVMs.

The kernelized SVM formulation considers a linear classifier in the feature space $\mathcal{H}$, a Hilbert space containing the range of some feature mapping $\Phi(\cdot)$. The standard formulation is as follows,

$$\begin{aligned} \min_{\mathbf{w},b}: \quad & r(\mathbf{w},b) + \sum_{i=1}^{m} \xi_i \\ \text{s.t.}: \quad & \xi_i \geq \big[1 - y_i(\langle \mathbf{w}, \Phi(\mathbf{x}_i)\rangle + b)\big], \\ & \xi_i \geq 0\,. \end{aligned}$$

It has been proved in [**133**] that if we take $f(\langle \mathbf{w}, \mathbf{w}\rangle)$ for some increasing function $f(\cdot)$ as the regularization term $r(\mathbf{w}, b)$, then the optimal solution has a representation $\mathbf{w}^* = \sum_{i=1}^{m} \alpha_i \Phi(\mathbf{x}_i)$, which can further be found without knowing explicitly the feature mapping, by evaluating a kernel function $k(\mathbf{x}, \mathbf{x}') \triangleq \langle \Phi(\mathbf{x}),\, \Phi(\mathbf{x}')\rangle$ only. This is the well-known "kernel trick".

The definitions of Atomic Uncertainty Set and Sublinear Aggregated Uncertainty Set in the feature space are identical to Definition 6.1 and 6.2, with $\mathbb{R}^n$ replaced by $\mathcal{H}$. The following theorem is a feature-space counterpart of Theorem 6.1. The proof follows from a similar argument to Theorem 6.1, i.e., for any fixed $(\mathbf{w}, b)$ the worst-case empirical error equals the empirical error plus a penalty term $\sup_{\boldsymbol{\delta}\in\mathcal{N}_0} \big(\langle \mathbf{w},\, \boldsymbol{\delta}\rangle\big)$, and hence the details are omitted.

THEOREM 6.3. *Assume* $\{\Phi(\mathbf{x}_i), y_i\}_{i=1}^m$ *are not linearly separable,* $r(\cdot) : \mathcal{H} \times \mathbb{R} \to \mathbb{R}$ *is an arbitrary function,* $\mathcal{N} \subseteq \mathcal{H}^m$ *is a Sublinear Aggregated Uncertainty set with corresponding atomic uncertainty set* $\mathcal{N}_0 \subseteq \mathcal{H}$*. Then the following min-max problem*

$$\min_{\mathbf{w},b} \sup_{(\boldsymbol{\delta}_1,\cdots,\boldsymbol{\delta_m})\in\mathcal{N}} \left\{ r(\mathbf{w},b) + \sum_{i=1}^m \max\left[1 - y_i(\langle \mathbf{w}, \Phi(\mathbf{x}_i) - \boldsymbol{\delta}_i\rangle + b),\, 0\right] \right\} \tag{6.13}$$

is equivalent to

$$\begin{aligned} \min: \quad & r(\mathbf{w},b) + \sup_{\boldsymbol{\delta}\in\mathcal{N}_0} \left(\langle \mathbf{w},\, \boldsymbol{\delta}\rangle\right) + \sum_{i=1}^m \xi_i, \\ \text{s.t.}: \quad & \xi_i \geq 1 - y_i\big(\langle \mathbf{w},\, \Phi(\mathbf{x}_i)\rangle + b\big), \quad i = 1, \cdots, m; \\ & \xi_i \geq 0, \quad i = 1, \cdots, m. \end{aligned} \tag{6.14}$$

Furthermore, the minimization of Problem (6.14) is attainable when $r(\cdot,\cdot)$ *is lower semi-continuous.*

For some widely used feature mappings (e.g., RKHS of a Gaussian kernel), $\{\Phi(\mathbf{x}_i), y_i\}_{i=1}^m$ are always separable. In this case, the worst-case empirical error may not be equal to the empirical error plus a penalty term $\sup_{\boldsymbol{\delta}\in\mathcal{N}_0} \left(\langle \mathbf{w},\, \boldsymbol{\delta}\rangle\right)$. However, it is easy to show that for any $(\mathbf{w}, b)$, the latter is an upper bound of the former.

The next corollary is the feature-space counterpart of Corollary 6.2, where $\|\cdot\|_{\mathcal{H}}$ stands for the RKHS norm, i.e., for $\mathbf{z} \in \mathcal{H}$, $\|\mathbf{z}\|_{\mathcal{H}} = \sqrt{\langle \mathbf{z},\, \mathbf{z}\rangle}$. Noticing that the RKHS norm is self dual, we find that the proof is identical to that of Corollary 6.2, and hence omit it.

COROLLARY 6.4. *Let* $\mathcal{T}_{\mathcal{H}} \triangleq \left\{(\boldsymbol{\delta}_1, \cdots \boldsymbol{\delta}_m) | \sum_{i=1}^m \|\boldsymbol{\delta}_i\|_{\mathcal{H}} \leq c\right\}$*. If* $\{\Phi(\mathbf{x}_i), y_i\}_{i=1}^m$ *are non-separable, then the following two optimization problems on* $(\mathbf{w}, b)$ *are equivalent*

$$\min: \quad \max_{(\boldsymbol{\delta}_1,\cdots,\boldsymbol{\delta}_m)\in\mathcal{T}_{\mathcal{H}}} \sum_{i=1}^m \max\left[1 - y_i\big(\langle \mathbf{w},\, \Phi(\mathbf{x}_i) - \boldsymbol{\delta}_i\rangle + b\big), 0\right], \tag{6.15}$$

$$\min: \quad c\|\mathbf{w}\|_{\mathcal{H}} + \sum_{i=1}^m \max\left[1 - y_i\big(\langle \mathbf{w},\, \Phi(\mathbf{x}_i)\rangle + b\big), 0\right]. \tag{6.16}$$

Equation (6.16) is a variant form of the standard SVM, where the latter has a squared RKHS norm regularization term. And it can be shown that the two formulations are equivalent up to a changing of the tradeoff parameter c, since both the empirical hinge-loss and the RKHS norm are convex. Therefore, Corollary 6.4 essentially means that the standard kernelized SVM is implicitly a robust classifier (without regularization) with disturbance in the feature-space, where the sum of the magnitudes of the disturbance is bounded.

Disturbance in the feature-space is less intuitive than disturbance in the sample space, and the next lemma relates these two different notions.

LEMMA 6.5. *Suppose there exists* $\mathcal{X} \subseteq \mathbb{R}^n$, $\rho > 0$, *and a continuous non-decreasing function* $f : \mathbb{R}^+ \to \mathbb{R}^+$ *satisfying* $f(0) = 0$, *such that*

$$k(\mathbf{x}, \mathbf{x}) + k(\mathbf{x}', \mathbf{x}') - 2k(\mathbf{x}, \mathbf{x}') \leq f(\|\mathbf{x} - \mathbf{x}'\|_2^2), \quad \forall \mathbf{x}, \mathbf{x}' \in \mathcal{X}, \|\mathbf{x} - \mathbf{x}'\|_2 \leq \rho$$

then

$$\|\Phi(\hat{\mathbf{x}} + \boldsymbol{\delta}) - \Phi(\hat{\mathbf{x}})\|_{\mathcal{H}} \leq \sqrt{f(\|\boldsymbol{\delta}\|_2^2)}, \quad \forall \|\boldsymbol{\delta}\|_2 \leq \rho, \ \hat{\mathbf{x}}, \hat{\mathbf{x}} + \boldsymbol{\delta} \in \mathcal{X}.$$

PROOF. Expanding the RKHS norm yields

$$\begin{aligned}
&\|\Phi(\hat{\mathbf{x}} + \boldsymbol{\delta}) - \Phi(\hat{\mathbf{x}})\|_{\mathcal{H}} \\
=&\sqrt{\langle \Phi(\hat{\mathbf{x}} + \boldsymbol{\delta}) - \Phi(\hat{\mathbf{x}}), \ \Phi(\hat{\mathbf{x}} + \boldsymbol{\delta}) - \Phi(\hat{\mathbf{x}}) \rangle} \\
=&\sqrt{\langle \Phi(\hat{\mathbf{x}} + \boldsymbol{\delta}), \ \Phi(\hat{\mathbf{x}} + \boldsymbol{\delta}) \rangle + \langle \Phi(\hat{\mathbf{x}}), \ \Phi(\hat{\mathbf{x}}) \rangle - 2\langle \Phi(\hat{\mathbf{x}} + \boldsymbol{\delta}), \ \Phi(\hat{\mathbf{x}}) \rangle} \\
=&\sqrt{k\big(\hat{\mathbf{x}} + \boldsymbol{\delta}, \ \hat{\mathbf{x}} + \boldsymbol{\delta}\big) + k\big(\hat{\mathbf{x}}, \ \hat{\mathbf{x}}\big) - 2k\big(\hat{\mathbf{x}} + \boldsymbol{\delta}, \ \ \hat{\mathbf{x}}\big)} \\
\leq&\sqrt{f(\|\hat{\mathbf{x}} + \boldsymbol{\delta} - \hat{\mathbf{x}}\|_2^2)} = \sqrt{f(\|\boldsymbol{\delta}\|_2^2)},
\end{aligned}$$

where the inequality follows from the assumption. □

Lemma 6.5 essentially says that under certain conditions, robustness in the feature space is a stronger requirement that robustness in the sample space. Therefore, a classifier that achieves robustness in the feature space (the SVM for example) also achieves robustness in the sample space. Notice that the condition of Lemma 6.5

is rather weak. In particular, it holds for any continuous $k(\cdot,\cdot)$ and bounded $\mathcal{X}$. Indeed, for RBF kernels, there exists a tighter relationship between disturbances in the feature space and disturbances in the sample space. Since such a relationship is not necessary in developing consistency of SVM, we defer it to Section 6.7.

In the next section we consider a more fundamental property of robustness in the sample space: we show that a classifier that is robust in the sample space is asymptotically consistent. As a consequence of this result for linear classifiers, this implies the consistency for a broad class of kernelized SVMs.

6.5. Consistency of regularization

In this section we explore a fundamental connection between learning and robustness, by using robustness properties to re-prove the statistical consistency of the linear SVM, and then the kernelized SVM. Indeed, our proof mirrors the consistency proof found in [**142**], with the key difference that *we replace metric entropy, VC-dimension, and stability conditions used there, with a robustness condition.*

Thus far we have considered the setup where the training-samples are corrupted by certain set-inclusive disturbances. We now turn to the standard statistical learning setup, by assuming that all training samples and testing samples are generated i.i.d. according to a (unknown) probability $\mathbb{P}$, i.e., there does not exist an explicit disturbance.

Let $\mathcal{X} \subseteq \mathbb{R}^n$ be bounded, and suppose the training samples $(\mathbf{x}_i, y_i)_{i=1}^{\infty}$ are generated i.i.d. according to an unknown distribution $\mathbb{P}$ supported by $\mathcal{X} \times \{-1, +1\}$. The next theorem shows that our robust classifier and equivalently, regularized SVM asymptotically minimizes an upper-bound of the expected classification error and hinge loss.

THEOREM 6.6. *Denote* $K \triangleq \max_{x\in\mathcal{X}} \|x\|_2$. *Then there exists a random sequence* $\{\gamma_{m,c}\}$ *such that:*

(1) $\forall c > 0$, $\lim_{m\to\infty} \gamma_{m,c} = 0$ *almost surely, and the convergence is uniform in* $\mathbb{P}$*;*

(2) the following bounds of the Bayes loss and the hinge loss hold uniformly for all $(\mathbf{w}, b)$*:*

$$\mathbb{E}_{(\mathbf{x},y)\sim\mathbb{P}}(\mathbf{1}_{y\neq sgn(\langle \mathbf{w},\mathbf{x}\rangle+b)}) \leq \gamma_{m,c} + c\|\mathbf{w}\|_2 + \frac{1}{m}\sum_{i=1}^{m} \max\big[1 - y_i(\langle \mathbf{w},\, \mathbf{x}_i\rangle + b), 0\big];$$

$$\mathbb{E}_{(\mathbf{x},y)\sim\mathbb{P}}\big(\max(1 - y(\langle \mathbf{w},\, \mathbf{x}\rangle + b),\, 0)\big) \leq$$

$$\gamma_{m,c}(1 + K\|\mathbf{w}\|_2 + |b|) + c\|\mathbf{w}\|_2 + \frac{1}{m}\sum_{i=1}^{m} \max\big[1 - y_i(\langle \mathbf{w},\, \mathbf{x}_i\rangle + b), 0\big].$$

PROOF. We briefly explain the basic idea of the proof before going to the technical details. We consider the testing sample set as a perturbed copy of the training sample set, and measure the magnitude of the perturbation. For testing samples that have "small" perturbations, $c\|\mathbf{w}\|_2+\frac{1}{m}\sum_{i=1}^{m} \max\big[1-y_i(\langle \mathbf{w},\, \mathbf{x}_i\rangle+b), 0\big]$ upper-bounds their total loss by Corollary 6.2. Therefore, we only need to show that the ratio of testing samples having "large" perturbations diminishes to prove the theorem.

Now we present the detailed proof. Given a $c > 0$, we call a testing sample $(\mathbf{x}', y')$ and a training sample $(\mathbf{x}, y)$ a *sample pair* if $y = y'$ and $\|\mathbf{x} - \mathbf{x}'\|_2 \leq c$. We say a set of training samples and a set of testing samples form l pairings if there exist l sample pairs with no data reused. Given m training samples and m testing samples, we use $M_{m,c}$ to denote the largest number of pairings. To prove this theorem, we need to establish the following lemma.

LEMMA 6.7. *Given a* $c > 0$*,* $M_{m,c}/m \to 1$ *almost surely as* $m \to +\infty$*, uniformly w.r.t.* $\mathbb{P}$*.*

PROOF. We make a partition of $\mathcal{X} \times \{-1, +1\} = \bigcup_{t=1}^{T_c} \mathcal{X}_t$ such that $\mathcal{X}_t$ either has the form $[\alpha_1, \alpha_1 + c/\sqrt{n}) \times [\alpha_2, \alpha_2 + c/\sqrt{n}) \cdots \times [\alpha_n, \alpha_n + c/\sqrt{n}) \times \{+1\}$ or $[\alpha_1, \alpha_1+c/\sqrt{n})\times[\alpha_2, \alpha_2+c/\sqrt{n}) \cdots\times[\alpha_n, \alpha_n+c/\sqrt{n})\times\{-1\}$ (recall n is the dimension of $\mathcal{X}$). That is, each partition is the Cartesian product of a rectangular cell in $\mathcal{X}$ and a singleton in $\{-1, +1\}$. Notice that if a training sample and a testing sample fall into $\mathcal{X}_t$, they can form a pairing.

Let N_t^{tr} and N_t^{te} be the number of training samples and testing samples falling in the t^{th} set, respectively. Thus, $(N_1^{tr}, \cdots, N_{T_c}^{tr})$ and $(N_1^{te}, \cdots, N_{T_c}^{te})$ are multinomially distributed random vectors following a same distribution. Notice that for a multinomially distributed random vector $(N_1, \cdots, N_k)$ with parameter m and $(p_1, \cdots, p_k)$, the following holds (Bretegnolle-Huber-Carol inequality, see for example Proposition A6.6 of [**154**]). For any $\lambda > 0$,

$$\mathbb{P}\Big(\sum_{i=1}^{k} \big|N_i - mp_i\big|) \geq 2\sqrt{m}\lambda\Big) \leq 2^k \exp(-2\lambda^2).$$

Hence we have

$$\begin{aligned}
& \mathbb{P}\Big(\sum_{t=1}^{T_c} \big|N_t^{tr} - N_t^{te}\big| \geq 4\sqrt{m}\lambda\Big) \leq 2^{T_c+1} \exp(-2\lambda^2), \\
\Longrightarrow \qquad & \mathbb{P}\Big(\frac{1}{m}\sum_{t=1}^{T_c} \big|N_t^{tr} - N_t^{te}\big| \geq \lambda\Big) \leq 2^{T_c+1} \exp(\frac{-m\lambda^2}{8}), \\
\Longrightarrow \qquad & \mathbb{P}\Big(M_{m,c}/m \leq 1 - \lambda\Big) \leq 2^{T_c+1} \exp(\frac{-m\lambda^2}{8}), \qquad (6.17)
\end{aligned}$$

Observe that $\sum_{m=1}^{\infty} 2^{T_c+1} \exp(\frac{-m\lambda^2}{8}) < +\infty$, hence by the Borel-Cantelli Lemma [**60**], with probability one the event $\{M_{m,c}/m \leq 1-\lambda\}$ only occurs finitely often as $m \to \infty$. That is, $\liminf_m M_{m,c}/m \geq 1 - \lambda$ almost surely. Since λ can be arbitrarily close to zero, $M_{m,c}/m \to 1$ almost surely. Observe that this convergence is uniform in $\mathbb{P}$, since T_c only depends on $\mathcal{X}$. □

Now we proceed to prove the theorem. Given m training samples and m testing samples with $M_{m,c}$ sample pairs, we notice that for these paired samples, both the total testing error and the total testing hinge-loss is upper bounded by

$$\begin{aligned}
& \max_{(\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m) \in \mathcal{N}_0 \times \cdots \times \mathcal{N}_0} \sum_{i=1}^{m} \max\big[1 - y_i\big(\langle \mathbf{w},\, \mathbf{x}_i - \boldsymbol{\delta}_i\rangle + b\big), 0\big] \\
\leq & cm\|\mathbf{w}\|_2 + \sum_{i=1}^{m} \max\big[1 - y_i(\langle \mathbf{w},\, \mathbf{x}_i\rangle + b),\, 0\big],
\end{aligned}$$

where $\mathcal{N}_0 = \{\delta \,|\, \|\delta\| \leq c\}$. Hence the total classification error of the m testing samples can be upper bounded by

$$(m - M_{m,c}) + cm\|\mathbf{w}\|_2 + \sum_{i=1}^{m} \max\big[1 - y_i(\langle \mathbf{w},\, \mathbf{x}_i\rangle + b),\, 0\big],$$

and since

$$\max_{\mathbf{x}\in\mathcal{X}}(1 - y(\langle \mathbf{w}, \mathbf{x}\rangle)) \leq \max_{\mathbf{x}\in\mathcal{X}} \Big\{1 + |b| + \sqrt{\langle \mathbf{x}, \mathbf{x}\rangle \cdot \langle \mathbf{w}, \mathbf{w}\rangle}\Big\} = 1 + |b| + K\|\mathbf{w}\|_2,$$

the accumulated hinge-loss of the total m testing samples is upper bounded by

$$(m - M_{m,c})(1 + K\|\mathbf{w}\|_2 + |b|) + cm\|\mathbf{w}\|_2 + \sum_{i=1}^{m} \max\big[1 - y_i(\langle \mathbf{w},\, \mathbf{x}_i\rangle + b),\, 0\big].$$

Therefore, the average testing error is upper bounded by

$$1 - M_{m,c}/m + c\|\mathbf{w}\|_2 + \frac{1}{m}\sum_{i=1}^{n} \max\big[1 - y_i(\langle \mathbf{w},\, \mathbf{x}_i\rangle + b),\, 0\big], \tag{6.18}$$

and the average hinge loss is upper bounded by

$$(1 - M_{m,c}/m)(1 + K\|\mathbf{w}\|_2 + |b|) + c\|\mathbf{w}\|_2 + \frac{1}{m}\sum_{i=1}^{m} \max\big[1 - y_i(\langle \mathbf{w},\, \mathbf{x}_i\rangle + b), 0\big].$$

Let $\gamma_{m,c} = 1 - M_{m,c}/m$. The proof follows since $M_{m,c}/m \to 1$ almost surely for any $c > 0$. Notice by Inequality (6.17) we have

$$\mathbb{P}\Big(\gamma_{m,c} \geq \lambda\Big) \leq \exp\Big(-m\lambda^2/8 + (T_c + 1)\log 2\Big), \tag{6.19}$$

i.e., the convergence is uniform in $\mathbb{P}$.

We have shown that the average testing error is upper bounded. The final step is to show that this implies that in fact the random variable given by the conditional expectation (conditioned on the training sample) of the error is bounded almost surely as in the statement of the theorem. To make things precise, consider a fixed m, and let $\omega_1 \in \Omega_1$ and $\omega_2 \in \Omega_2$ generate the m training samples and m testing samples, respectively, and for shorthand let $\mathcal{T}^m$ denote the random variable consisting of the first m training samples. Let us denote the probability measures for the training by

ρ_1 and the testing samples by ρ_2. By independence, the joint measure is given by the product of these two. We rely on this property in what follows. Now fix a λ and a $c > 0$. In our new notation, Equation (6.19) now reads:

$$\begin{aligned}\int_{\Omega_1}\int_{\Omega_2} \mathbf{1}\big\{\gamma_{m,c}(\omega_1,\omega_2) \geq \lambda\big\}\, d\rho_2(\omega_2)\, d\rho_1(\omega_1) &= \mathbb{P}\Big(\gamma_{m,c}(\omega_1,\omega_2) \geq \lambda\Big) \\ &\leq \exp\Big(-m\lambda^2/8 + (T_c+1)\log 2\Big).\end{aligned}$$

We now bound $\mathbb{P}_{\omega_1}(\mathbb{E}_{\omega_2}[\gamma_{m,c}(\omega_1,\omega_2)\,|\,\mathcal{T}^m] > \lambda)$, and then use Borel-Cantelli to show that this event can happen only finitely often. We have:

$$\begin{aligned}&\mathbb{P}_{\omega_1}(\mathbb{E}_{\omega_2}[\gamma_{m,c}(\omega_1,\omega_2)\,|\,\mathcal{T}^m] > \lambda) \\ =\ & \int_{\Omega_1} \mathbf{1}\Big\{ \int_{\Omega_2} \gamma_{m,c}(\omega_1,\omega_2)\, d\rho_2(\omega_2) > \lambda\Big\}\, d\rho_1(\omega_1) \\ \leq\ & \int_{\Omega_1} \mathbf{1}\Big\{ \big[\int_{\Omega_2} \gamma_{m,c}(\omega_1,\omega_2)\mathbf{1}(\gamma_{m,c}(\omega_1,\omega_2) \leq \lambda)\, d\rho_2(\omega_2) + \\ & \int_{\Omega_2} \gamma_{m,c}(\omega_1,\omega_2)\mathbf{1}(\gamma_{m,c}(\omega_1,\omega_2) > \lambda)\, d\rho_2(\omega_2)\big] \geq 2\lambda\Big\} d\rho_1(\omega_1) \\ \leq\ & \int_{\Omega_1} \mathbf{1}\Big\{ \big[\int_{\Omega_2} \lambda\mathbf{1}(\lambda(\omega_1,\omega_2) \leq \lambda)\, d\rho_2(\omega_2) + \\ & \int_{\Omega_2} \mathbf{1}(\gamma_{m,c}(\omega_1,\omega_2) > \lambda)\, d\rho_2(\omega_2)\big] \geq 2\lambda\Big\} d\rho_1(\omega_1) \\ \leq\ & \int_{\Omega_1} \mathbf{1}\Big\{ \big[\lambda + \int_{\Omega_2} \mathbf{1}(\gamma_{m,c}(\omega_1,\omega_2) > \lambda)\, d\rho_2(\omega_2)\big] \geq 2\lambda\Big\} d\rho_1(\omega_1) \\ =\ & \int_{\Omega_1} \mathbf{1}\Big\{ \int_{\Omega_2} \mathbf{1}(\gamma_{m,c}(\omega_1,\omega_2) > \lambda)\, d\rho_2(\omega_2) \geq \lambda\Big\} d\rho_1(\omega_1).\end{aligned}$$

Here, the first equality holds because training and testing samples are independent, and hence the joint measure is the product of ρ_1 and ρ_2. The second inequality holds because $\gamma_{m,c}(\omega_1,\omega_2) \leq 1$ everywhere. Further notice that

$$\begin{aligned}&\int_{\Omega_1}\int_{\Omega_2} \mathbf{1}\big\{\gamma_{m,c}(\omega_1,\omega_2) \geq \lambda\big\}\, d\rho_2(\omega_2)\, d\rho_1(\omega_1) \\ &\geq \int_{\Omega_1} \lambda\mathbf{1}\Big\{ \int_{\Omega_2} \mathbf{1}\big(\gamma_{m,c}(\omega_1,\omega_2) \geq \lambda\big)\, d\rho(\omega_2) > \lambda\Big\}\, d\rho_1(\omega_1).\end{aligned}$$

Thus we have

$$\mathbb{P}(\mathbb{E}_{\omega_2}(\gamma_{m,c}(\omega_1,\omega_2)) > \lambda) \leq \mathbb{P}\Big(\gamma_{m,c}(\omega_1,\omega_2) \geq \lambda\Big)/\lambda \leq \exp\Big(-m\lambda^2/8+(T_c+1)\log 2\Big)/\lambda.$$

For any λ and c, summing up the right hand side over $m = 1$ to ∞ is finite, hence the theorem follows from the Borel-Cantelli lemma. □

REMARK 6.1. We note that M_m/m converges to 1 almost surely even if $\mathcal{X}$ is not bounded. Indeed, to see this, fix $\epsilon > 0$, and let $\mathcal{X}' \subseteq \mathcal{X}$ be a bounded set such that $\mathbb{P}(\mathcal{X}') > 1 - \epsilon$. Then, with probability one,

$$\#(\text{unpaired samples in}\mathcal{X}')/m \to 0,$$

by Lemma 6.7. In addition,

$$\max\big(\#(\text{training samples not in } \mathcal{X}'), \#(\text{testing samples not in } \mathcal{X}')\big)/m \to \epsilon.$$

Notice that

$$\begin{aligned} M_m \geq m &- \#(\text{unpaired samples in } \mathcal{X}') \\ &- \max\big(\#(\text{training samples not in } \mathcal{X}'), \#(\text{testing samples not in } \mathcal{X}')\big). \end{aligned}$$

Hence

$$\lim_{m\to\infty} M_m/m \geq 1 - \epsilon,$$

almost surely. Since ϵ is arbitrary, we have $M_m/m \to 1$ almost surely.

Next, we prove an analog of Theorem 6.6 for the kernelized case, and then show that these two imply statistical consistency of linear and kernelized SVMs. Again, let $\mathcal{X} \subseteq \mathbb{R}^n$ be bounded, and suppose the training samples $(\mathbf{x}_i, y_i)_{i=1}^{\infty}$ are generated i.i.d. according to an unknown distribution $\mathbb{P}$ supported on $\mathcal{X} \times \{-1, +1\}$.

THEOREM 6.8. *Denote* $K \triangleq \max_{\mathbf{x}\in\mathcal{X}} k(\mathbf{x},\mathbf{x})$. *Suppose there exists* $\rho > 0$ *and a continuous non-decreasing function* $f : \mathbb{R}^+ \to \mathbb{R}^+$ *satisfying* $f(0) = 0$, *such that:*

$$k(\mathbf{x},\mathbf{x}) + k(\mathbf{x}',\mathbf{x}') - 2k(\mathbf{x},\mathbf{x}') \leq f(\|\mathbf{x}-\mathbf{x}'\|_2^2), \quad \forall \mathbf{x},\mathbf{x}' \in \mathcal{X}, \|\mathbf{x}-\mathbf{x}'\|_2 \leq \rho.$$

Then there exists a random sequence $\{\gamma_{m,c}\}$ *such that,*

(1) $\forall c > 0$, $\lim_{m\to\infty} \gamma_{m,c} = 0$ *almost surely, and the convergence is uniform in* $\mathbb{P}$*;*

(2) the following bounds on the Bayes loss and the hinge loss hold uniformly for all $(\mathbf{w}, b) \in \mathcal{H} \times \mathbb{R}$

$$\mathbb{E}_{\mathbb{P}}(\mathbf{1}_{y\neq sgn(\langle \mathbf{w},\, \Phi(\mathbf{x})\rangle + b)}) \leq \gamma_{m,c} + c\|\mathbf{w}\|_{\mathcal{H}} + \frac{1}{m}\sum_{i=1}^{m} \max\big[1 - y_i(\langle \mathbf{w},\, \Phi(\mathbf{x}_i)\rangle + b), 0\big],$$

$$\mathbb{E}_{(\mathbf{x},y)\sim\mathbb{P}}\big(\max(1 - y(\langle \mathbf{w},\, \Phi(\mathbf{x})\rangle + b),\, 0)\big) \leq$$

$$\gamma_{m,c}(1 + K\|\mathbf{w}\|_{\mathcal{H}} + |b|) + c\|\mathbf{w}\|_{\mathcal{H}} + \frac{1}{m}\sum_{i=1}^{m} \max\big[1 - y_i(\langle \mathbf{w},\, \Phi(\mathbf{x}_i)\rangle + b), 0\big].$$

PROOF. As in the proof of Theorem 6.6, we generate a set of m testing samples and m training samples, and then lower-bound the number of samples that can form a *sample pair* in the feature-space; that is, a pair consisting of a training sample $(\mathbf{x}, y)$ and a testing sample $(\mathbf{x}', y')$ such that $y = y'$ and $\|\Phi(\mathbf{x}) - \Phi(\mathbf{x}')\|_{\mathcal{H}} \leq c$. In contrast to the finite-dimensional sample space, the feature space may be infinite dimensional, and thus our decomposition may have an infinite number of "bricks." In this case, our multinomial random variable argument used in the proof of Lemma 6.7 breaks down. Nevertheless, we are able to lower bound the number of sample pairs in the feature space by the number of sample pairs in the *sample space*.

Define $f^{-1}(\alpha) \triangleq \max\{\beta \geq 0 | f(\beta) \leq \alpha\}$. Since $f(\cdot)$ is continuous, $f^{-1}(\alpha) > 0$ for any $\alpha > 0$. Now notice that by Lemma 6.5, if a testing sample $\mathbf{x}$ and a training sample $\mathbf{x}'$ belong to a "brick" with length of each side $\min(\rho/\sqrt{n},\, f^{-1}(c^2)/\sqrt{n})$ in the *sample space* (see the proof of Lemma 6.7), $\|\Phi(\mathbf{x}) - \Phi(\mathbf{x}')\|_{\mathcal{H}} \leq c$. Hence the number of *sample pairs* in the feature space is lower bounded by the number of pairs of samples that fall in the same brick in the sample space. We can cover $\mathcal{X}$ with finitely many (denoted as T_c) such bricks since $f^{-1}(c^2) > 0$. Then, a similar argument as in Lemma 6.7 shows that the ratio of samples that form pairs in a brick converges to 1 as m increases. Further notice that for M paired samples, the total testing error and

hinge-loss are both upper-bounded by

$$cM\|\mathbf{w}\|_{\mathcal{H}} + \sum_{i=1}^{M} \max\left[1 - y_i(\langle \mathbf{w},\, \Phi(\mathbf{x}_i)\rangle + b), 0\right].$$

The rest of the proof is identical to Theorem 6.6. In particular, Inequality (6.19) still holds. □

Notice that the condition in Theorem 6.8 is satisfied by most widely used kernels, e.g., homogeneous polynominal kernels, and Gaussian RBF. This condition requires that the feature mapping is "smooth" and hence preserves "locality" of the disturbance, i.e., small disturbance in the sample space guarantees the corresponding disturbance in the feature space is also small. It is easy to construct non-smooth kernel functions which do not generalize well. For example, consider the following kernel:

$$k(\mathbf{x}, \mathbf{x}') = \begin{cases} 1 & \mathbf{x} = \mathbf{x}'; \\ 0 & \mathbf{x} \neq \mathbf{x}'. \end{cases}$$

A standard RKHS regularized SVM using this kernel leads to a decision function

$$\text{sign}(\sum_{i=1}^{m} \alpha_i k(\mathbf{x}, \mathbf{x}_i) + b),$$

which equals sign(b) and provides no meaningful prediction if the testing sample $\mathbf{x}$ is not one of the training samples. Hence as m increases, the testing error remains as large as 50% regardless of the tradeoff parameter used in the algorithm, while the training error can be made arbitrarily small by fine-tuning the parameter.

Convergence to Bayes Risk. Next we relate the results of Theorem 6.6 and Theorem 6.8 to the standard consistency notion, i.e., convergence to the Bayes Risk [**142**]. The key point of interest in our proof is the use of a robustness condition in place of a VC-dimension or stability condition used in [**142**]. The proof in [**142**] has 4 main steps. They show: (i) there always exists a minimizer to the expected regularized (kernel) hinge loss; (ii) the expected regularized hinge loss of the minimizer converges to the expected hinge loss as the regularizer goes to zero; (iii) if a sequence of functions

asymptotically have optimal expected hinge loss, then they also have optimal expected loss; and (iv) the expected hinge loss of the minimizer of the regularized *training* hinge loss concentrates around the empirical regularized hinge loss. In [**142**], this final step, (iv), is accomplished using concentration inequalities derived from VC-dimension considerations, and stability considerations.

Instead, we use our robustness-based results of Theorem 6.6 and Theorem 6.8 to replace these approaches (Lemmas 3.21 and 3.22 in [**142**]) in proving step (iv), and thus to establish the main result.

Recall that a classifier is a rule that assigns to every training set $T = \{\mathbf{x}_i, y_i\}_{i=1}^m$ a measurable function f_T. The risk of a measurable function $f : \mathcal{X} \to \mathbb{R}$ is defined as

$$\mathcal{R}_{\mathbb{P}}(f) \triangleq \mathbb{P}(\{\mathbf{x}, y : \text{sign} f(\mathbf{x}) \neq y\}).$$

The smallest achievable risk

$$\mathcal{R}_{\mathbb{P}} \triangleq \inf\{\mathcal{R}_{\mathbb{P}}(f) | f : \mathcal{X} \to \mathbb{R} \,\text{measurable}\}$$

is called the *Bayes Risk* of $\mathbb{P}$. A classifier is said to be strongly uniformly consistent if for all distributions P on $\mathcal{X} \times [-1, +1]$, the following holds almost surely.

$$\lim_{m \to \infty} \mathcal{R}_{\mathbb{P}}(f_T) = \mathcal{R}_{\mathbb{P}}.$$

Without loss of generality, we only consider the kernel version. Recall a definition from [**142**].

DEFINITION 6.3. *Let $C(\mathcal{X})$ be the set of all continuous functions defined on $\mathcal{X}$, equipped with a metric $d(f_1, f_2) = \sup_{\mathbf{x} \in \mathcal{X}} |f_1(\mathbf{x}) - f_2(\mathbf{x})|$. Consider the mapping $I : \mathcal{H} \to C(\mathcal{X})$ defined by $I\mathbf{w} \triangleq \langle \mathbf{w}, \Phi(\cdot) \rangle$. If I has a dense image, we call the kernel* universal.

Roughly speaking, if a kernel is universal, then the corresponding RKHS is rich enough to satisfy the condition of step (ii) above.

THEOREM 6.9. *If a kernel satisfies the condition of Theorem 6.8, and is universal, then the Kernel SVM with $c \downarrow 0$ sufficiently slowly is strongly uniformly consistent.*

PROOF. We first introduce some notation, largely following [**142**]. For some probability measure μ and $(\mathbf{w}, b) \in \mathcal{H} \times \mathbb{R}$,

$$R_{L,\mu}((\mathbf{w}, b)) \triangleq \mathbb{E}_{(\mathbf{x},y)\sim\mu}\big\{ \max(0, 1 - y(\langle \mathbf{w}, \Phi(\mathbf{x})\rangle + b))\big\},$$

is the expected hinge-loss under probability μ, and

$$R^c_{L,\mu}((\mathbf{w}, b)) \triangleq c\|\mathbf{w}\|_{\mathcal{H}} + \mathbb{E}_{(\mathbf{x},y)\sim\mu}\big\{ \max(0, 1 - y(\langle \mathbf{w}, \Phi(\mathbf{x})\rangle + b))\big\}$$

is the regularized expected hinge-loss. Hence $R_{L,\mathbb{P}}(\cdot)$ and $R^c_{L,\mathbb{P}}(\cdot)$ are the expected hinge-loss and regularized expected hinge-loss under the generating probability $\mathbb{P}$. If μ is the empirical distribution of m samples, we write $R_{L,m}(\cdot)$ and $R^c_{L,m}(\cdot)$ respectively. Notice $R^c_{L,m}(\cdot)$ is the objective function of the SVM. Denote its solution by $f_{m,c}$, i.e., the classifier we get by running SVM with m samples and parameter c. Further denote by $f_{\mathbb{P},c} \in \mathcal{H} \times \mathbb{R}$ the minimizer of $R^c_{L,\mathbb{P}}(\cdot)$. The existence of such a minimizer is proved in Lemma 3.1 of [**142**] (step (i)). Let

$$\mathcal{R}_{L,\mathbb{P}} \triangleq \min_{f \text{ measurable}} \mathbb{E}_{\mathbf{x},y\sim\mathbb{P}}\Big\{ \max\big(1 - yf(\mathbf{x}),\, 0\big)\Big\},$$

i.e., the smallest achievable hinge-loss for all measurable functions.

The main content of our proof is to use Theorems 6.6 and 6.8 to prove step (iv) in [**142**]. In particular, we show: if $c \downarrow 0$ "slowly", we have with probability one

$$\lim_{m\to\infty} R_{L,\mathbb{P}}(f_{m,c}) = \mathcal{R}_{L,\mathbb{P}}. \tag{6.20}$$

To prove Equation (6.20), denote by $\mathbf{w}(f)$ and $b(f)$ as the weight part and offset part of any classifier f. Next, we bound the magnitude of $f_{m,c}$ by using $R^c_{L,m}(f_{m,c}) \leq R^c_{L,m}(\mathbf{0}, 0) \leq 1$, which leads to

$$\|\mathbf{w}(f_{m,c})\|_{\mathcal{H}} \leq 1/c$$

and

$$|b(f_{m,c})| \leq 2 + K\|\mathbf{w}(f_{m,c})\|_{\mathcal{H}} \leq 2 + K/c.$$

From Theorem 6.8 (note that the bound holds uniformly for all $(\mathbf{w}, b)$), we have

$$\begin{aligned}
R_{L,\mathbb{P}}(f_{m,c}) &\leq \gamma_{m,c}[1 + K\|\mathbf{w}(f_{m,c})\|_{\mathcal{H}} + |b|] + R^c_{L,m}(f_{m,c}) \\
&\leq \gamma_{m,c}[3 + 2K/c] + R^c_{L,m}(f_{m,c}) \\
&\leq \gamma_{m,c}[3 + 2K/c] + R^c_{L,m}(f_{\mathbb{P},c}) \\
&= \mathcal{R}_{L,\mathbb{P}} + \gamma_{m,c}[3 + 2K/c] + \left\{R^c_{L,m}(f_{\mathbb{P},c}) - R^c_{L,\mathbb{P}}(f_{\mathbb{P},c})\right\} + \left\{R^c_{L,\mathbb{P}}(f_{\mathbb{P},c}) - \mathcal{R}_{L,\mathbb{P}}\right\} \\
&= \mathcal{R}_{L,\mathbb{P}} + \gamma_{m,c}[3 + 2K/c] + \left\{R_{L,m}(f_{\mathbb{P},c}) - R_{L,\mathbb{P}}(f_{\mathbb{P},c})\right\} + \left\{R^c_{L,\mathbb{P}}(f_{\mathbb{P},c}) - \mathcal{R}_{L,\mathbb{P}}\right\}.
\end{aligned}$$

The last inequality holds because $f_{m,c}$ minimizes $R^c_{L,m}$.

It is known (Proposition 3.2 [**142**]) (step (ii)) that if the kernel used is rich enough, i.e., universal, then

$$\lim_{c \to 0} R^c_{L,\mathbb{P}}(f_{\mathbb{P}}, c) = \mathcal{R}_{L,\mathbb{P}}.$$

For fixed $c > 0$, we have

$$\lim_{m \to \infty} R_{L,m}(f_{\mathbb{P},c}) = R_{L,\mathbb{P}}(f_{\mathbb{P},c}),$$

almost surely due to the strong law of large numbers (notice that $f_{\mathbb{P},c}$ is a fixed classifier), and $\gamma_{m,c}[3 + 2K/c] \to 0$ almost surely. Notice that neither convergence rate depends on $\mathbb{P}$. Therefore, if $c \downarrow 0$ sufficiently slowly,[3] we have almost surely

$$\lim_{m \to \infty} R_{L,\mathbb{P}}(f_{m,c}) \leq \mathcal{R}_{L,\mathbb{P}}.$$

Now, for any m and c, we have $R_{L,\mathbb{P}}(f_{m,c}) \geq \mathcal{R}_{L,\mathbb{P}}$ by definition. This implies that Equation (6.20) holds almost surely, thus giving us step (iv).

[3]For example, we can take $\{c(m)\}$ be the smallest number satisfying $c(m) \geq m^{-1/8}$ and $T_{c(m)} \leq m^{1/8}/\log 2 - 1$. Inequality (6.19) thus leads to $\sum_{m=1}^{\infty} P(\gamma_{m,c(m)}/c(m) \geq m^{1/4}) \leq +\infty$ which implies uniform convergence of $\gamma_{m,c(m)}/c(m)$.

Finally, Proposition 3.3. of [**142**] shows step (iii), namely, approximating hinge loss is sufficient to guarantee approximation of the Bayes loss. Thus Equation (6.20) implies that the risk of the function $f_{m,c}$ converges to the Bayes risk. □

6.6. Chapter summary

This chapter considers the relationship between robust and regularized SVM classification. In particular, we prove that the standard norm-regularized SVM classifier is in fact the solution to a robust classification setup, and thus known results about regularized classifiers extend to robust classifiers. To the best of our knowledge, this is the first explicit such link between regularization and robustness in pattern classification. This link suggests that norm-based regularization essentially builds in a robustness to sample noise whose probability level sets are symmetric, and moreover have the structure of the unit ball with respect to the dual of the regularizing norm. It would be interesting to understand the performance gains possible when the noise does not have such characteristics, and the robust setup is used in place of regularization with appropriately defined uncertainty set.

Based on the robustness interpretation of the regularization term, we re-proved the consistency of SVMs without direct appeal to notions of metric entropy, VC-dimension, or stability. Our proof suggests that the ability to handle disturbance is crucial for an algorithm to achieve good generalization ability. In particular, for "smooth" feature mappings, the robustness to disturbance in the observation space is guaranteed and hence SVMs achieve consistency. On the other-hand, certain "non-smooth" feature mappings fail to be consistent simply because for such kernels the robustness in the feature-space (guaranteed by the regularization process) does not imply robustness in the observation space.

6.7. An exact equivalence of robustness in sample space and feature space

We show in this section that we can relate robustness in the feature space and robustness in the sample space more directly for RBF kernels.

THEOREM 6.10. *Suppose the Kernel function has the form* $k(\mathbf{x}, \mathbf{x}') = f(\|\mathbf{x}-\mathbf{x}'\|)$*, with* $f : \mathbb{R}^+ \to \mathbb{R}$ *a decreasing function. Denote by* $\mathcal{H}$ *the RKHS space of* $k(\cdot,\cdot)$ *and* $\Phi(\cdot)$ *the corresponding feature mapping. Then we have for any* $\mathbf{x} \in \mathbb{R}^n$*,* $\mathbf{w} \in \mathcal{H}$ *and* $c > 0$*,*

$$\sup_{\|\boldsymbol{\delta}\| \le c} \langle \mathbf{w}, \Phi(\mathbf{x} - \boldsymbol{\delta}) \rangle = \sup_{\|\boldsymbol{\delta}_\phi\|_{\mathcal{H}} \le \sqrt{2f(0)-2f(c)}} \langle \mathbf{w}, \Phi(\mathbf{x}) + \boldsymbol{\delta}_\phi \rangle.$$

PROOF. We show that the left-hand-side is not larger than the right-hand-side, and vice versa.

First we show

$$\sup_{\|\boldsymbol{\delta}\| \le c} \langle \mathbf{w}, \Phi(\mathbf{x} - \boldsymbol{\delta}) \rangle \le \sup_{\|\boldsymbol{\delta}_\phi\|_{\mathcal{H}} \le \sqrt{2f(0)-2f(c)}} \langle \mathbf{w}, \Phi(\mathbf{x}) - \boldsymbol{\delta}_\phi \rangle. \tag{6.21}$$

We notice that for any $\|\boldsymbol{\delta}\| \le c$, we have

$$\begin{aligned}
&\langle \mathbf{w}, \Phi(\mathbf{x} - \boldsymbol{\delta}) \rangle \\
=&\Big\langle \mathbf{w}, \Phi(\mathbf{x}) + \big(\Phi(\mathbf{x} - \boldsymbol{\delta}) - \Phi(\mathbf{x})\big) \Big\rangle \\
=&\langle \mathbf{w}, \Phi(\mathbf{x}) \rangle + \langle \mathbf{w}, \Phi(\mathbf{x} - \boldsymbol{\delta}) - \Phi(\mathbf{x}) \rangle \\
\le&\langle \mathbf{w}, \Phi(\mathbf{x}) \rangle + \|\mathbf{w}\|_{\mathcal{H}} \cdot \|\Phi(\mathbf{x} - \boldsymbol{\delta}) - \Phi(\mathbf{x})\|_{\mathcal{H}} \\
\le&\langle \mathbf{w}, \Phi(\mathbf{x}) \rangle + \|\mathbf{w}\|_{\mathcal{H}} \sqrt{2f(0) - 2f(c)} \\
=& \sup_{\|\boldsymbol{\delta}_\phi\|_{\mathcal{H}} \le \sqrt{2f(0)-2f(c)}} \langle \mathbf{w}, \Phi(\mathbf{x}) - \boldsymbol{\delta}_\phi \rangle.
\end{aligned}$$

Taking the supremum over $\boldsymbol{\delta}$ establishes Inequality (6.21).

Next, we show the opposite inequality,

$$\sup_{\|\boldsymbol{\delta}\|\leq c} \langle \mathbf{w},\, \Phi(\mathbf{x}-\boldsymbol{\delta})\rangle \geq \sup_{\|\boldsymbol{\delta}_\phi\|_{\mathcal{H}}\leq\sqrt{2f(0)-2f(c)}} \langle \mathbf{w},\, \Phi(\mathbf{x})-\boldsymbol{\delta}_\phi\rangle. \tag{6.22}$$

If $f(c) = f(0)$, then Inequality 6.22 holds trivially, hence we only consider the case that $f(c) < f(0)$. Notice that the inner product is a continuous function in $\mathcal{H}$, hence for any $\epsilon > 0$, there exists a $\boldsymbol{\delta}'_\phi$ such that

$$\langle \mathbf{w},\, \Phi(\mathbf{x})-\boldsymbol{\delta}'_\phi\rangle > \sup_{\|\boldsymbol{\delta}_\phi\|_{\mathcal{H}}\leq\sqrt{2f(0)-2f(c)}} \langle \mathbf{w},\, \Phi(\mathbf{x})-\boldsymbol{\delta}_\phi\rangle - \epsilon; \quad \|\boldsymbol{\delta}'_\phi\|_{\mathcal{H}} < \sqrt{2f(0)-2f(c)}.$$

Recall that the RKHS space is the completion of the feature mapping, thus there exists a sequence of $\{\mathbf{x}'_i\} \in \mathbb{R}^n$ such that

$$\Phi(\mathbf{x}'_i) \to \Phi(\mathbf{x}) - \boldsymbol{\delta}'_\phi, \tag{6.23}$$

which is equivalent to

$$\big(\Phi(\mathbf{x}'_i) - \Phi(\mathbf{x})\big) \to -\boldsymbol{\delta}'_\phi.$$

This leads to

$$\begin{aligned}\lim_{i\to\infty} \sqrt{2f(0)-2f(\|\mathbf{x}'_i-\mathbf{x}\|)} &= \lim_{i\to\infty} \|\Phi(\mathbf{x}'_i)-\Phi(\mathbf{x})\|_{\mathcal{H}} \\ &= \|\boldsymbol{\delta}'_\phi\|_{\mathcal{H}} < \sqrt{2f(0)-2f(c)}.\end{aligned}$$

Since f is decreasing, we conclude that $\|\mathbf{x}'_i - \mathbf{x}\| \leq c$ holds except for a finite number of i. By (6.23) we have

$$\langle \mathbf{w},\, \Phi(\mathbf{x}'_i)\rangle \to \langle \mathbf{w},\, \Phi(\mathbf{x})-\boldsymbol{\delta}'_\phi\rangle > \sup_{\|\boldsymbol{\delta}_\phi\|_{\mathcal{H}}\leq\sqrt{2f(0)-2f(c)}} \langle \mathbf{w},\, \Phi(\mathbf{x})-\boldsymbol{\delta}_\phi\rangle - \epsilon,$$

which means

$$\sup_{\|\boldsymbol{\delta}\|\leq c} \langle \mathbf{w},\, \Phi(\mathbf{x}-\boldsymbol{\delta})\rangle \geq \sup_{\|\boldsymbol{\delta}_\phi\|_{\mathcal{H}}\leq\sqrt{2f(0)-2f(c)}} \langle \mathbf{w},\, \Phi(\mathbf{x})-\boldsymbol{\delta}_\phi\rangle - \epsilon.$$

Since ϵ is arbitrary, we establish Inequality (6.22).

Combining Inequality (6.21) and Inequality (6.22) proves the theorem. □

CHAPTER 7

Robust Regression and Lasso

Similarly to Chapter 6, in this chapter we consider the robustness property of another widely used learning algorithm: Lasso. Part of the material in this chapter appears in [**165**] and [**167**].

Lasso, or ℓ^1 regularized least squares, has been explored extensively for its remarkable sparsity properties. We show in this chapter that the solution to Lasso, in addition to its sparsity, has robustness properties: it is the solution to a robust optimization problem. This has two important consequences. First, robustness provides a connection of the regularizer to a physical property, namely, protection from noise. This allows a principled selection of the regularizer, and in particular, generalizations of Lasso that also yield convex optimization problems are obtained by considering different uncertainty sets.

Secondly, robustness can itself be used as an avenue to exploring different properties of the solution. In particular, it is shown that robustness of the solution explains why the solution is sparse. The analysis as well as the specific results obtained differ from standard sparsity results, providing different geometric intuition. Furthermore, it is shown that the robust optimization formulation is related to kernel density estimation, and based on this approach, a proof that Lasso is consistent is given using robustness directly. Finally, a theorem saying that Lasso is not stable, is presented.

7.1. Introduction

In this chapter we consider linear regression problems with least-square error. The problem is to find a vector $\mathbf{x}$ so that the ℓ_2 norm of the residual $\mathbf{b}-A\mathbf{x}$ is minimized, for a given matrix $A \in \mathbb{R}^{n\times m}$ and vector $\mathbf{b} \in \mathbb{R}^n$. From a learning/regression perspective, each row of A can be regarded as a training sample, and the corresponding element of b as the target value of this observed sample. Each column of A corresponds to a feature, and the objective is to find a set of weights so that the weighted sum of the feature values approximates the target value.

It is well known that minimizing the squared error can lead to sensitive solutions [**66, 81, 88, 72**]. Many regularization methods have been proposed to decrease this sensitivity. Among them, Tikhonov regularization [**147**] and Lasso [**146, 61**] are two widely known and cited algorithms. These methods minimize a weighted sum of the residual norm and a certain regularization term, $\|\mathbf{x}\|_2$ for Tikhonov regularization and $\|\mathbf{x}\|_1$ for Lasso. In addition to providing regularity, Lasso is also known for the tendency to select sparse solutions. Recently this has attracted much attention for its ability to reconstruct sparse solutions when sampling occurs far below the Nyquist rate, and also for its ability to recover the sparsity pattern exactly with probability one, asymptotically as the number of observations increases (there is an extensive literature on this subject, and we refer the reader to [**38, 71, 36, 151, 160**] and references therein).

The first result of this chapter is that the solution to Lasso has robustness properties: it is the solution to a robust optimization problem. In itself, this interpretation of Lasso as the solution to a robust least squares problem is a development in line with the results of [**64**]. There, the authors propose an alternative approach for reducing sensitivity of linear regression by considering a robust version of the regression problem, i.e., minimizing the worst-case residual for the observations under some unknown but bounded disturbance. Most of the research in this area considers either the case where the disturbance is row-wise uncoupled [**137**], or the case where the Frobenius norm of the disturbance matrix is bounded [**64**].

None of these robust optimization approaches produces a solution that has sparsity properties (in particular, the solution to Lasso does not solve any of these previously formulated robust optimization problems). In contrast, we investigate the robust regression problem where the uncertainty set is defined by feature-wise constraints. Such a noise model is of interest when the values of the features are obtained with some noisy pre-processing steps, and the magnitudes of such noises are known or bounded. Another situation of interest is where the features are meaningfully coupled. We define *coupled* and *uncoupled* disturbances and uncertainty sets precisely in Section 7.2.1 below. Intuitively, a disturbance is feature-wise coupled if the variation or disturbance across features satisfy joint constraints, and uncoupled otherwise.

Considering the solution to Lasso as the solution of a robust least squares problem has two important consequences. First, robustness provides a connection of the regularizer to a physical property, namely, protection from noise. This allows more principled selection of the regularizer, and in particular, considering different uncertainty sets, we construct generalizations of Lasso that also yield convex optimization problems.

Secondly, and perhaps most significantly, robustness is a strong property that can itself be used as an avenue to investigating different properties of the solution. We show that robustness of the solution can explain why the solution is sparse. The analysis as well as the specific results we obtain differ from standard sparsity results, providing different geometric intuition, and extending beyond the least-squares setting. Sparsity results obtained for Lasso ultimately depend on the fact that introducing additional features incurs larger ℓ^1-penalty than the least squares error reduction. In contrast, we exploit the fact that a robust solution is, by definition, the optimal solution under a worst-case perturbation. Our results show that, essentially, a coefficient of the solution is nonzero if the corresponding feature is relevant under all allowable perturbations. In addition to sparsity, we also use robustness directly to prove consistency of Lasso.

We briefly list the main contributions as well as the organization of this chapter.

- In Section 7.2, we formulate the robust regression problem with feature-wise independent disturbances, and show that this formulation is equivalent to a least-squares problem with a weighted ℓ_1 norm regularization term. Hence, we provide an interpretation of Lasso from a robustness perspective.
- We generalize the robust regression formulation to loss functions of arbitrary norm in Section 7.3. We also consider uncertainty sets that require disturbances of different features to satisfy joint conditions. This can be used to mitigate the conservativeness of the robust solution and to obtain solutions with additional properties. We mention further examples of the flexibility of the robust formulation, including uncertainty sets with both column-wise and feature-wise disturbances, as well as a class of cardinality-constrained robust-regression problems which smoothly interpolate between Lasso and a ℓ_∞-norm regularizer.
- In Section 7.4, we present new sparsity results for the robust regression problem with feature-wise independent disturbances. This provides a new robustness-based explanation to the sparsity of Lasso. Our approach gives new analysis and also geometric intuition, and furthermore allows one to obtain sparsity results for more general loss functions, beyond the squared loss.
- Next, we relate Lasso to kernel density estimation in Section 7.5. This allows us to re-prove consistency in a statistical learning setup, using the new robustness tools and formulation we introduce. Along with our results on sparsity, this illustrates the power of robustness in explaining and also exploring different properties of the solution.
- Finally, we prove in Section 7.6 a "no-free-lunch" theorem, stating that an algorithm that encourages sparsity cannot be stable.

NOTATION. We use capital letters to represent matrices, and boldface letters to represent column vectors. Row vectors are represented as the transpose of column vectors. For a vector $\mathbf{z}$, z_i denotes its i^{th} element. Throughout the chapter, $\mathbf{a}_i$ and

$\mathbf{r}_j^\top$ are used to denote the i^{th} column and the j^{th} row of the observation matrix A, respectively. We use a_{ij} to denote the ij element of A, hence it is the j^{th} element of $\mathbf{r}_i$, and i^{th} element of $\mathbf{a}_j$. For a convex function $f(\cdot)$, $\partial f(\mathbf{z})$ represents any of its sub-gradients evaluated at $\mathbf{z}$. A vector with length n and each element equals 1 is denoted as $\mathbf{1}_n$.

7.2. Robust regression with feature-wise disturbance

In this section, we show that our robust regression formulation recovers Lasso as a special case. We also derive probabilistic bounds that guide in the construction of the uncertainty set.

The regression formulation we consider differs from the standard Lasso formulation, as we minimize the norm of the error, rather than the squared norm. It is known that these two coincide up to a change of the regularization coefficient. Yet as we discuss above, our results lead to more flexible and potentially powerful robust formulations, and give new insight into known results.

7.2.1. Formulation. Robust linear regression considers the case where the observed matrix is corrupted by some potentially malicious disturbance. The objective is to find the optimal solution in the worst case sense. This is usually formulated as the following min-max problem,

$$\textbf{Robust Linear Regression:} \\ \min_{\mathbf{x}\in\mathbb{R}^m} \left\{ \max_{\Delta A\in\mathcal{U}} \|\mathbf{b} - (A+\Delta A)\mathbf{x}\|_2 \right\}, \tag{7.1}$$

where $\mathcal{U}$ is called the *uncertainty set*, or the set of admissible disturbances of the matrix A. In this section, we consider the class of uncertainty sets that bound the norm of the disturbance to each feature, without placing any joint requirements across feature disturbances. That is, we consider the class of uncertainty sets:

$$\mathcal{U} \triangleq \left\{ (\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m) \middle| \|\boldsymbol{\delta}_i\|_2 \le c_i, \;\; i = 1, \cdots, m \right\}, \tag{7.2}$$

for given $c_i \geq 0$. We call these uncertainty sets *feature-wise uncoupled*, in contrast to *coupled* uncertainty sets that require disturbances of different features to satisfy some joint constraints (we discuss these extensively below, and their significance). While the inner maximization problem of (7.1) is nonconvex, we show in the next theorem that uncoupled norm-bounded uncertainty sets lead to an easily solvable optimization problem.

THEOREM 7.1. *The robust regression problem (7.1) with uncertainty set of the form (7.2) is equivalent to the following ℓ^1 regularized regression problem:*

$$\min_{\mathbf{x}\in\mathbb{R}^m} \Big\{ \|\mathbf{b} - A\mathbf{x}\|_2 + \sum_{i=1}^m c_i |x_i| \Big\}. \tag{7.3}$$

PROOF. Fix $\mathbf{x}^*$. We prove that $\max_{\Delta A\in\mathcal{U}} \|\mathbf{b} - (A+\Delta A)\mathbf{x}^*\|_2 = \|\mathbf{b} - A\mathbf{x}^*\|_2 + \sum_{i=1}^m c_i|x_i^*|$.

The left hand side can be written as

$$\begin{aligned}
&\max_{\Delta A\in\mathcal{U}} \|\mathbf{b} - (A+\Delta A)\mathbf{x}^*\|_2 \\
=&\max_{(\boldsymbol{\delta}_1,\cdots,\boldsymbol{\delta}_m)|\|\boldsymbol{\delta}_i\|_2\leq c_i} \Big\| \mathbf{b} - \big(A + (\boldsymbol{\delta}_1,\cdots,\boldsymbol{\delta}_m)\big)\mathbf{x}^* \Big\|_2 \\
=&\max_{(\boldsymbol{\delta}_1,\cdots,\boldsymbol{\delta}_m)|\|\boldsymbol{\delta}_i\|_2\leq c_i} \|\mathbf{b} - A\mathbf{x}^* - \sum_{i=1}^m x_i^*\boldsymbol{\delta}_i\|_2 \\
\leq&\max_{(\boldsymbol{\delta}_1,\cdots,\boldsymbol{\delta}_m)|\|\boldsymbol{\delta}_i\|_2\leq c_i} \Big\| \mathbf{b} - A\mathbf{x}^* \Big\|_2 + \sum_{i=1}^m \|x_i^*\boldsymbol{\delta}_i\|_2 \\
\leq&\|\mathbf{b} - A\mathbf{x}^*\|_2 + \sum_{i=1}^m |x_i^*|c_i.
\end{aligned} \tag{7.4}$$

Now, let

$$\mathbf{u} \triangleq \begin{cases} \frac{\mathbf{b}-A\mathbf{x}^*}{\|\mathbf{b}-A\mathbf{x}^*\|_2} & \text{if } A\mathbf{x}^* \neq \mathbf{b}, \\ \text{any vector with unit } \ell^2 \text{ norm} & \text{otherwise;} \end{cases}$$

and let

$$\boldsymbol{\delta}_i^* \triangleq -c_i \mathrm{sgn}(x_i^*)\mathbf{u}.$$

Observe that $\|\boldsymbol{\delta}_i^*\|_2 \leq c_i$, hence $\Delta A^* \triangleq (\boldsymbol{\delta}_1^*, \cdots, \boldsymbol{\delta}_m^*) \in \mathcal{U}$. Notice that

$$\begin{aligned}
&\max_{\Delta A \in \mathcal{U}} \|\mathbf{b} - (A + \Delta A)\mathbf{x}^*\|_2 \\
\geq &\|\mathbf{b} - (A + \Delta A^*)\mathbf{x}^*\|_2 \\
= &\Big\|\mathbf{b} - \big(A + (\boldsymbol{\delta}_1^*, \cdots, \boldsymbol{\delta}_m^*)\big)\mathbf{x}^*\Big\|_2 \\
= &\Big\|(\mathbf{b} - A\mathbf{x}^*) - \sum_{i=1}^m \big(-x_i^* c_i \mathrm{sgn}(x_i^*)\mathbf{u}\big)\Big\|_2 \qquad (7.5) \\
= &\Big\|(\mathbf{b} - A\mathbf{x}^*) + (\sum_{i=1}^m c_i|x_i^*|)\mathbf{u}\Big\|_2 \\
= &\|\mathbf{b} - A\mathbf{x}^*\|_2 + \sum_{i=1}^m c_i|x_i^*|.
\end{aligned}$$

The last equation holds from the definition of $\mathbf{u}$.

Combining Inequalities (7.4) and (7.5), establishes the equality $\max_{\Delta A \in \mathcal{U}} \|\mathbf{b} - (A + \Delta A)\mathbf{x}^*\|_2 = \|\mathbf{b} - A\mathbf{x}^*\|_2 + \sum_{i=1}^m c_i|x_i^*|$ for any $\mathbf{x}^*$. Minimizing over $\mathbf{x}$ on both sides proves the theorem. □

Taking $c_i = c$ and normalizing $\mathbf{a}_i$ for all i, Problem (7.3) recovers the well-known Lasso [**146, 61**].

7.2.2. Uncertainty set construction. The selection of an uncertainty set $\mathcal{U}$ in Robust Optimization is of fundamental importance. One way this can be done is as an approximation of so-called *chance constraints*, where a deterministic constraint is replaced by the requirement that a constraint is satisfied with at least some probability. These can be formulated when we know the distribution exactly, or when we have only partial information of the uncertainty, such as, e.g., first and second moments. This chance-constraint formulation is particularly important when the distribution has large support, rendering the naive robust optimization formulation overly pessimistic.

For confidence level η, the chance constraint formulation becomes:

$$\begin{aligned} \text{minimize:} \quad & t \\ \text{Subject to:} \quad & \Pr(\|\mathbf{b} - (A + \Delta A)\mathbf{x}\|_2 \leq t) \geq 1 - \eta. \end{aligned}$$

Here, $\mathbf{x}$ and t are the decision variables.

Constructing the uncertainty set for feature i can be done quickly via line search and bisection, as long as we can evaluate $\Pr(\|\mathbf{a}_i\|_2 \geq c)$. If we know the distribution exactly (i.e., if we have complete probabilistic information), this can be quickly done via sampling. Another setting of interest is when we have access only to some moments of the distribution of the uncertainty, e.g., the mean and variance. In this setting, the uncertainty sets are constructed via a bisection procedure which evaluates the worst-case probability over all distributions with given mean and variance. We do this using a tight bound on the probability of an event, given the first two moments.

In the scalar case, the Markov Inequality provides such a bound. The next theorem is a generalization of the Markov inequality to $\mathbb{R}^n$, which bounds the probability where the disturbance on a given feature is more than c_i, if only the first and second moment of the random variable are known. We refer the reader to [**20**] for similar results using semi-definite optimization.

THEOREM 7.2. *Consider a random vector* $\mathbf{v} \in \mathbb{R}^n$*, such that* $\mathbb{E}(\mathbf{v}) = \mathbf{a}$*, and* $\mathbb{E}(\mathbf{v}\mathbf{v}^\top) = \Sigma$*,* $\Sigma \succeq 0$*. Then we have*

$$\Pr\{\|\mathbf{v}\|_2 \geq c_i\} \leq \left\{ \begin{array}{ll} \min_{P,\mathbf{q},r,\lambda} & \text{Trace}(\Sigma P) + 2\mathbf{q}^\top \mathbf{a} + r \\ \textit{subject to:} & \begin{pmatrix} P & \mathbf{q} \\ \mathbf{q}^\top & r \end{pmatrix} \succeq 0 \\ & \begin{pmatrix} I(m) & \mathbf{0} \\ \mathbf{0}^\top & -c_i^2 \end{pmatrix} \preceq \lambda \begin{pmatrix} P & \mathbf{q} \\ \mathbf{q}^\top & r-1 \end{pmatrix} \\ & \lambda \geq 0. \end{array} \right. \tag{7.6}$$

PROOF. Consider a function $f(\cdot)$ parameterized by $P, \mathbf{q}, r$ defined as $f(\mathbf{v}) = \mathbf{v}^\top P\mathbf{v} + 2\mathbf{q}^\top\mathbf{v} + r$. Notice $\mathbb{E}\big(f(\mathbf{v})\big) = \text{Trace}(\Sigma P) + 2\mathbf{q}^\top\mathbf{a} + r$. Now we show that $f(\mathbf{v}) \geq \mathbf{1}_{\|\mathbf{v}\|\geq c_i}$ for all $P, \mathbf{q}, r$ satisfying the constraints in (7.6).

To show $f(\mathbf{v}) \geq \mathbf{1}_{\|\mathbf{v}\|_2\geq c_i}$, we need to establish (i) $f(\mathbf{v}) \geq 0$ for all $\mathbf{v}$, and (ii) $f(\mathbf{v}) \geq 1$ when $\|\mathbf{v}\|_2 \geq c_i$. Notice that

$$f(\mathbf{v}) = \begin{pmatrix} \mathbf{v} \\ 1 \end{pmatrix}^\top \begin{pmatrix} P & \mathbf{q} \\ \mathbf{q}^\top & r \end{pmatrix} \begin{pmatrix} \mathbf{v} \\ 1 \end{pmatrix},$$

hence (i) holds because

$$\begin{pmatrix} P & \mathbf{q} \\ \mathbf{q}^\top & r \end{pmatrix} \succeq 0.$$

To establish condition (ii), it suffices to show $\mathbf{v}^\top\mathbf{v} \geq c_i^2$ implies $\mathbf{v}^\top P\mathbf{v} + 2\mathbf{q}^\top\mathbf{v} + r \geq 1$, which is equivalent to show $\big\{\mathbf{v}\big|\mathbf{v}^\top P\mathbf{v} + 2\mathbf{q}^\top\mathbf{v} + r - 1 \leq 0\big\} \subseteq \big\{\mathbf{v}\big|\mathbf{v}^\top\mathbf{v} \leq c_i^2\big\}$. Noticing this is an ellipsoid-containment condition, by S-Procedure, we see that is equivalent to the condition that there exists a $\lambda \geq 0$ such that

$$\begin{pmatrix} I(m) & \mathbf{0} \\ \mathbf{0}^\top & -c_i^2 \end{pmatrix} \preceq \lambda \begin{pmatrix} P & \mathbf{q} \\ \mathbf{q}^\top & r - 1 \end{pmatrix}.$$

Hence we have $f(\mathbf{v}) \geq \mathbf{1}_{\|\mathbf{v}\|_2\geq c_i}$, taking expectation over both side that notice that the expectation of a indicator function is the probability, we establish the theorem. □

The optimization problem (7.6) is a semi-definite programming, which can be solved in polynomial time. Furthermore, if we replace $\mathbb{E}(\mathbf{v}\mathbf{v}^\top) = \Sigma$ by an inequality $\mathbb{E}(\mathbf{v}\mathbf{v}^\top) \leq \Sigma$, the uniform bound still holds. Thus, even if our estimate of the variance is not precise, we are still able to bound the probability of having "large" disturbance.

7.3. General uncertainty sets

One reason the robust optimization formulation is powerful, is that having provided the connection to Lasso, it then allows the opportunity to generalize to efficient "Lasso-like" regularization algorithms.

In this section, we make several generalizations of the robust formulation (7.1) and derive counterparts of Theorem 7.1. In Section 7.3.1 we generalize the robust formulation in two ways: (a) to the case of arbitrary norm; and (b) to the case of coupled uncertainty sets. In Section 7.3.2 we investigate a class of uncertainty sets inspired by [**22**], that control the *cardinality* of perturbed features. The uncertainty sets are non-convex, but nevertheless we show that the resulting robust regression problem is still tractable. In the last subsection, we consider a disturbance model where both column-wise disturbance and row-wise disturbance exist simultaneously.

7.3.1. Arbitrary norm and coupled disturbance. We first consider the case of an arbitrary norm $\|\cdot\|_a$ of $\mathbb{R}^n$ as a cost function rather than the squared loss. The proof of the next theorem is identical to that of Theorem 7.1, with only the ℓ^2 norm changed to $\|\cdot\|_a$.

THEOREM 7.3. *The robust regression problem*

$$\min_{\mathbf{x}\in\mathbb{R}^m}\left\{\max_{\Delta A\in\mathcal{U}_a}\|\mathbf{b}-(A+\Delta A)\mathbf{x}\|_a\right\};\ \mathcal{U}_a\triangleq\left\{(\boldsymbol{\delta}_1,\cdots,\boldsymbol{\delta}_m)\Big|\|\boldsymbol{\delta}_i\|_a\le c_i,\ \ i=1,\cdots,m\right\};$$

is equivalent to the following regularized regression problem

$$\min_{\mathbf{x}\in\mathbb{R}^m}\left\{\|\mathbf{b}-A\mathbf{x}\|_a+\sum_{i=1}^m c_i|x_i|\right\}.$$

We next remove the assumption that the disturbances are feature-wise uncoupled. Allowing coupled uncertainty sets is useful when we have some additional information about potential noise in the problem, and we want to limit the conservativeness of the worst-case formulation. Consider the following uncertainty set:

$$\mathcal{U}'\triangleq\left\{(\boldsymbol{\delta}_1,\cdots,\boldsymbol{\delta}_m)\big|f_j(\|\boldsymbol{\delta}_1\|_a,\cdots,\|\boldsymbol{\delta}_m\|_a)\le 0;\ j=1,\cdots,k\right\},$$

where $f_j(\cdot)$ are convex functions. Note that both k and f_j can be arbitrary, hence this is a very general formulation, and provides us with significant flexibility in designing uncertainty sets and equivalently new regression algorithms (see for example

Corollary 7.5 and 7.6). The following theorem converts this formulation to tractable optimization problems.

THEOREM 7.4. *Assume that the set*

$$\mathcal{Z} \triangleq \{\mathbf{z} \in \mathbb{R}^m | f_j(\mathbf{z}) \leq 0,\ j = 1, \cdots, k;\ \mathbf{z} \geq \mathbf{0}\}$$

has non-empty relative interior. Then the robust regression problem

$$\min_{\mathbf{x} \in \mathbb{R}^m} \left\{ \max_{\Delta A \in \mathcal{U}'} \|\mathbf{b} - (A + \Delta A)\mathbf{x}\|_a \right\}$$

is equivalent to the following regularized regression problem

$$\begin{aligned} &\min_{\boldsymbol{\lambda} \in \mathbb{R}^k_+, \boldsymbol{\kappa} \in \mathbb{R}^m_+, \mathbf{x} \in \mathbb{R}^m} \Big\{ \|\mathbf{b} - A\mathbf{x}\|_a + v(\boldsymbol{\lambda}, \boldsymbol{\kappa}, \mathbf{x}) \Big\}; \\ &\textit{where: } v(\boldsymbol{\lambda}, \boldsymbol{\kappa}, \mathbf{x}) \triangleq \max_{\mathbf{c} \in \mathbf{R}^m} \Big[(\boldsymbol{\kappa} + |\mathbf{x}|)^\top \mathbf{c} - \sum_{j=1}^k \lambda_j f_j(\mathbf{c}) \Big] \end{aligned} \tag{7.7}$$

PROOF. Fix a solution $\mathbf{x}^*$. Note that

$$\mathcal{U}' = \{(\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m) | \mathbf{c} \in \mathcal{Z};\ \|\boldsymbol{\delta}_i\|_a \leq c_i,\ i = 1, \cdots, m\}.$$

Hence we have:

$$\begin{aligned} &\max_{\Delta A \in \mathcal{U}'} \|\mathbf{b} - (A + \Delta A)\mathbf{x}^*\|_a \\ =&\max_{\mathbf{c} \in \mathcal{Z}} \Big\{ \max_{\|\boldsymbol{\delta}_i\|_a \leq c_i,\, i=1,\cdots,m} \|\mathbf{b} - \big(A + (\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m)\big)\mathbf{x}^*\|_a \Big\} \\ =&\max_{\mathbf{c} \in \mathcal{Z}} \Big\{ \|\mathbf{b} - A\mathbf{x}^*\|_a + \sum_{i=1}^m c_i |x_i^*| \Big\} \\ =&\|\mathbf{b} - A\mathbf{x}^*\|_a + \max_{\mathbf{c} \in \mathcal{Z}} \Big\{ |\mathbf{x}^*|^\top \mathbf{c} \Big\}. \end{aligned} \tag{7.8}$$

The second equality follows from Theorem 7.3.

Now we need to evaluate $\max_{\mathbf{c} \in \mathcal{Z}} \{|\mathbf{x}^*|^\top \mathbf{c}\}$, which equals to $-\min_{\mathbf{c} \in \mathcal{Z}} \{-|\mathbf{x}^*|^\top \mathbf{c}\}$. Hence we are minimizing a linear function subject to a set of convex constraints. Furthermore, by assumption the Slater's condition holds. Hence the duality gap of

$\min_{\mathbf{c}\in\mathcal{Z}}\{-|\mathbf{x}^*|^\top\mathbf{c}\}$ is zero. A standard duality analysis shows that

$$\max_{\mathbf{c}\in\mathcal{Z}}\left\{|\mathbf{x}^*|^\top\mathbf{c}\right\} = \min_{\boldsymbol{\lambda}\in\mathbb{R}^k_+,\boldsymbol{\kappa}\in\mathbb{R}^m_+} v(\boldsymbol{\lambda},\boldsymbol{\kappa},\mathbf{x}^*). \tag{7.9}$$

We establish the theorem by substituting Equation (7.9) back into Equation (7.8) and taking minimum over $\mathbf{x}$ on both sides. □

REMARK: Problem (7.7) is efficiently solvable. Denote $z^{\mathbf{c}}(\boldsymbol{\lambda},\boldsymbol{\kappa},\mathbf{x}) \triangleq \Big[(\boldsymbol{\kappa}+|\mathbf{x}|)^\top\mathbf{c} - \sum_{j=1}^k \lambda_j f_j(\mathbf{c})\Big]$. This is a convex function of $(\boldsymbol{\lambda},\boldsymbol{\kappa},\mathbf{x})$, and the sub-gradient of $z^{\mathbf{c}}(\cdot)$ can be computed easily for any $\mathbf{c}$. The function $v(\boldsymbol{\lambda},\boldsymbol{\kappa},\mathbf{x})$ is the maximum of a set of convex functions, $z^{\mathbf{c}}(\cdot)$, hence is convex, and satisfies

$$\partial v(\boldsymbol{\lambda}^*,\boldsymbol{\kappa}^*,\mathbf{x}^*) = \partial z^{\mathbf{c}_0}(\boldsymbol{\lambda}^*,\boldsymbol{\kappa}^*,\mathbf{x}^*),$$

where $\mathbf{c}_0$ maximizes $\Big[(\boldsymbol{\kappa}^*+|\mathbf{x}|^*)^\top\mathbf{c} - \sum_{j=1}^k \lambda_j^* f_j(\mathbf{c})\Big]$. We can efficiently evaluate $\mathbf{c}_0$ due to convexity of $f_j(\cdot)$, and hence we can efficiently evaluate the sub-gradient of $v(\cdot)$.

The next two corollaries are a direct application of Theorem 7.4.

COROLLARY 7.5. *Suppose* $\mathcal{U}' = \left\{(\boldsymbol{\delta}_1,\cdots,\boldsymbol{\delta}_m)\middle|\big\|\|\boldsymbol{\delta}_1\|_a,\cdots,\|\boldsymbol{\delta}_m\|_a\big\|_s \le l;\right\}$ *for a symmetric norm* $\|\cdot\|_s$*, then the resulting regularized regression problem is*

$$\min_{\mathbf{x}\in\mathbb{R}^m}\left\{\|\mathbf{b}-A\mathbf{x}\|_a + l\|\mathbf{x}\|_s^*\right\}; \quad \textit{where } \|\cdot\|_s^* \textit{ is the dual norm of } \|\cdot\|_s.$$

This corollary interprets *arbitrary* norm-based regularizers from a robust regression perspective. For example, it is straightforward to show that if we take both $\|\cdot\|_\alpha$ and $\|\cdot\|_s$ as the Euclidean norm, then $\mathcal{U}'$ is the set of matrices with their Frobenius norms bounded, and Corollary 7.5 reduces to the robust formulation introduced by [**64**].

COROLLARY 7.6. *Suppose* $\mathcal{U}' = \left\{(\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m)\middle|\exists \mathbf{c} \geq \mathbf{0} : T\mathbf{c} \leq \mathbf{s};\ \|\boldsymbol{\delta}_j\|_a \leq c_j;\right\}$, *then the resulting regularized regression problem is*

$$\begin{aligned} \textit{Minimize:} \quad & \|\mathbf{b} - A\mathbf{x}\|_a + \mathbf{s}^\top \boldsymbol{\lambda} \\ \textit{Subject to:} \quad & \mathbf{x} \leq T^\top \boldsymbol{\lambda} \\ & -\mathbf{x} \leq T^\top \boldsymbol{\lambda} \\ & \boldsymbol{\lambda} \geq \mathbf{0}. \end{aligned}$$

Unlike previous results, this corollary considers general polytope uncertainty sets. Advantages of such sets include the linearity of the final formulation. Moreover, the modeling power is considerable, as many interesting disturbances can be modeled in this way.

7.3.2. A class of non-convex uncertainty sets. Theorem 7.4 deals with convex uncertainty sets. Next we consider a class of non-convex but still tractable uncertainty sets, which can be regarded as interpolations between the uncorrelated case and the fully correlated case. To be specific, we consider the case that no more than a given number of features are disturbed. This formulation is inspired by [**22**] in which a similar uncertainty set for robust LP is considered. Let

$$\begin{aligned} \mathcal{Z}_t \triangleq \{\mathbf{z} \in \mathbb{R}^m \big| & \exists S \subseteq \{1, \cdots, m\}, |S| = \lfloor t \rfloor, \forall i \in S, 0 \leq z_i \leq c_i; \\ & \exists j \in \{1, \cdots, m\} \backslash S; 0 \leq z_j \leq (t - \lfloor t \rfloor) c_j;\ \forall k \notin S \cup \{j\}, z_k = 0\}. \end{aligned}$$

$$\mathcal{U}_t \triangleq \left\{\left(\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m\right)\middle|\exists \mathbf{z} \in \mathcal{Z}_t,\ \|\boldsymbol{\delta}_i\|_a = z_i.\right\}$$

Here, $\lfloor t \rfloor$ stands for the largest integer not larger than t. $\mathcal{U}_t$ represents an uncertainty set, such that the deviation of each feature is bounded by c_i and only t features are allowed to deviate. For t being a non-integer, it is interpreted as to allow $\lfloor t \rfloor$ features to completely deviate, and one other feature to partially deviate. Neither $\mathcal{Z}_t$ nor $\mathcal{U}_t$ is a convex set. Nevertheless, the robust regression problem with $\mathcal{U}_t$ as the uncertainty set is still tractable because it is equivalent to a robust regression problem with the

following polyhedral uncertainty set:

$$\tilde{\mathcal{Z}}_t \triangleq \left\{ \mathbf{z} \in \mathbb{R}^m \middle| 0 \le z_i \le c_i;\ \sum_{i=1}^{m} z_i/c_i \le t \right\};$$
$$\tilde{\mathcal{U}}_t \triangleq \left\{ \left(\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m\right) \middle| \exists \mathbf{z} \in \tilde{\mathcal{Z}}_t,\ \|\boldsymbol{\delta}_i\|_a = z_i. \right\}.$$

Note that $\tilde{\mathcal{U}}_t$ itself has an intuitively appealing interpretation as the set of disturbances such that besides the norm bound for disturbance on each feature, there exists an extra constraint which bounds the (weighted) total disturbance.

PROPOSITION 7.7. *For any* $\mathbf{x}^*$*, and* $1 \le t \le m$*, the following holds*

$$\max_{\Delta A \in \mathcal{U}_t} \|\mathbf{b} - (A + \Delta A)\mathbf{x}^*\|_a \equiv \max_{\Delta A \in \tilde{\mathcal{U}}_t} \|\mathbf{b} - (A + \Delta A)\mathbf{x}^*\|_a$$

PROOF. Observe that $\mathcal{Z}_t \subseteq \tilde{\mathcal{Z}}_t$ and $\mathcal{U}_t \subseteq \tilde{\mathcal{U}}_t$, hence

$$\max_{\Delta A \in \mathcal{U}_t} \|\mathbf{b} - (A + \Delta A)\mathbf{x}^*\|_a \le \max_{\Delta A \in \tilde{\mathcal{U}}_t} \|\mathbf{b} - (A + \Delta A)\mathbf{x}^*\|_a.$$

To prove the proposition, it suffices to show that

$$\arg\max_{\Delta A \in \tilde{\mathcal{U}}_t} \|\mathbf{b} - (A + \Delta A)\mathbf{x}^*\|_a \in \mathcal{U}_t,$$

which is equivalent to show that

$$\arg\max_{\mathbf{z} \in \tilde{\mathcal{Z}}_t} \left\{ |\mathbf{x}^*|^\top \mathbf{z} \right\} \in \mathcal{Z}_t.$$

The left-hand side is the solution for the following linear programming

$$\begin{aligned} \text{Maximize:} \quad & |\mathbf{x}^*|^\top \mathbf{z} \\ \text{Subject to:} \quad & 0 \le z_i \le c_i \\ & \sum_{i=1}^{m} z_i/c_i \le t. \end{aligned} \tag{7.10}$$

Let $v_i \triangleq z_i/c_i$, LP (7.10) is equivalent to

$$\begin{aligned} \text{Maximize:} \quad & \sum_{i=1}^{m} c_i |x_i^*| v_i \\ \text{Subject to:} \quad & 0 \le v_i \le 1 \\ & \sum_{i=1}^{m} v_i \le t. \end{aligned}$$

Observe for this LP, there is an optimal solution $\mathbf{v}^*$ which set value 1 for $\lfloor t \rfloor$ variables, and set $t - \lfloor t \rfloor$ on another variable. It is easy to check that the corresponding $\mathbf{z}^* \in \mathcal{Z}_t$. □

Combining Proposition 7.7 and Theorem 7.4 leads to the following corollary.

COROLLARY 7.8. *The robust regression problem*

$$\min_{\mathbf{x} \in \mathbb{R}^m} \left\{ \max_{\Delta A \in \mathcal{U}_t} \|\mathbf{b} - (A + \Delta A)\mathbf{x}\|_a \right\};$$

is equivalent to the following regularized regression problem

$$\begin{aligned} \textit{Minimize:} \quad & \|\mathbf{b} - A\mathbf{x}\|_a + \sum_{i=1}^{m} c_i \lambda_i + t\xi \\ \textit{Subject to:} \quad & x_i - \lambda_i - \xi/c_i \le 0, \quad i = 1, \cdots, m \\ & -x_i - \lambda_i - \xi/c_i \le 0, \quad i = 1, \cdots, m \\ & \lambda_i \ge 0, \quad i = 1, \cdots, m \\ & \xi \ge 0. \end{aligned}$$

If all the c_i are same, the robust regression with $\mathcal{U}_m$ (a non-correlated set) is Lasso, while the robust regression with $\mathcal{U}_1$ (a fully correlated set) leads to a ℓ^∞ norm regularization, which is known to be non-sparse. Our empirical results will show that the number of allowable deviations (i.e., the correlation level of the uncertainty set) plays an important role in controlling the sparsity level.

7.3.3. Row and column uncertainty case. Next we consider a case where we have both row-wise uncertainty and column-wise uncertainty. One motivation to consider this is the well-known elastic net method ([**181**]) known to sometimes outperform Lasso, in addition to possessing other properties of interest.

Combing row-wise and column-wise uncertainty leads to the following robust optimization problem

$$\begin{aligned} &\min_{\mathbf{x}} \max_{\Delta A_1 \in \mathcal{U}_1, \Delta A_2 \in \mathcal{U}_2} \|\mathbf{b} - (A + \Delta A_1 + \Delta A_2)\mathbf{x}\|_2, \\ \text{where:} \quad &\mathcal{U}_1 = \Big\{ (\mathbf{l}_1, \cdots, \mathbf{l}_n)^\top \Big| \mathbf{l}_j^\top \Sigma_j^{-1} \mathbf{l}_j \leq 1, \quad i = 1, \cdots, n \Big\}; \\ &\mathcal{U}_2 = \Big\{ (\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m) \Big| \|\boldsymbol{\delta}_i\|_2 \leq c_i, \;\; i = 1, \cdots, m \Big\}; \end{aligned} \tag{7.11}$$

for positive definite matrices Σ_j and positive scalars c_i.

THEOREM 7.9. *Denote the j^{th} row of A as $\mathbf{r}_j^\top$. Then given $\mathbf{x}$, the following holds*

$$\max_{\Delta A_1 \in \mathcal{U}_1, \Delta A_2 \in \mathcal{U}_2} \|\mathbf{b} - (A + \Delta A_1 + \Delta A_2)\mathbf{x}\|_2 = \sqrt{\sum_{j=1}^{n} \left(|b_j - \mathbf{r}_j^\top \mathbf{x}| + \|\Sigma_j^{1/2} \mathbf{x}\|_2 \right)^2} + \sum_{i=1}^{m} c_i |x_i|,$$

and moreover, the robust regression problem (7.11) is equivalent to the following Second Order Cone Program on $(\mathbf{x}, \mathbf{z}, \mathbf{t}, w)$:

$$\begin{aligned} \textit{Minimize:} \quad & w + \sum_{i=1}^{m} c_i z_i \\ \textit{Subject to:} \quad & \mathbf{x} \leq \mathbf{z}; \\ & -\mathbf{x} \leq \mathbf{z} \\ & \|\Sigma_j^{1/2} \mathbf{x}\|_2 \leq t_j - b_j + \mathbf{r}_j^\top \mathbf{x}; \quad j = 1, \cdots, n. \\ & \|\Sigma_j^{1/2} \mathbf{x}\|_2 \leq t_j + b_j - \mathbf{r}_j^\top \mathbf{x}; \quad j = 1, \cdots, n. \\ & \|\mathbf{t}\|_2 \leq w. \end{aligned}$$

PROOF. To prove the theorem, it suffices to show that for given $\mathbf{x}$, the following holds

$$\max_{\Delta A_1 \in \mathcal{U}_1, \Delta A_2 \in \mathcal{U}_2} \|\mathbf{b}-(A+\Delta A_1+\Delta A_2)\mathbf{x}\|_2 = \sqrt{\sum_{j=1}^{n} \left(|b_j - \mathbf{r}_j^\top \mathbf{x}| + \|\Sigma_j^{1/2}\mathbf{x}\|_2\right)^2} + \sum_{i=1}^{m} c_i|x_i|.$$

Notice that

$$\begin{aligned}
&\max_{\Delta A_1 \in \mathcal{U}_1, \Delta A_2 \in \mathcal{U}_2} \|\mathbf{b} - (A + \Delta A_1 + \Delta A_2)\mathbf{x}\|_2 \\
&= \max_{\Delta A_1 \in \mathcal{U}_1} \{ \max_{\Delta A_2 \in \mathcal{U}_2} \|\mathbf{b} - (A + \Delta A_1 + \Delta A_2)\mathbf{x}\|_2 \} \\
&= \max_{\Delta A_1 \in \mathcal{U}_1} \{\|\mathbf{b} - (A + \Delta A_1)\mathbf{x}\|_2 + \sum_{i=1}^{m} c_i|x_i|\} \\
&= \max_{\Delta A_1 \in \mathcal{U}_1} (\|\mathbf{b} - (A + \Delta A_1)\mathbf{x}\|_2) + \sum_{i=1}^{m} c_i|x_i|.
\end{aligned}$$

Furthermore, the following equation proves the theorem.

$$\begin{aligned}
&\max_{\Delta A_1 \in \mathcal{U}_1} (\|\mathbf{b} - (A + \Delta A_1)\mathbf{x}\|_2) \\
=&\sqrt{\sum_{j=1}^{n} \max_{\mathbf{l}_j \Sigma_j^{-1} \mathbf{l}_j \leq 1} (b_j - \mathbf{r}_j^\top \mathbf{x} - \mathbf{l}_j^\top \mathbf{x})^2} \\
=&\sqrt{\sum_{j=1}^{n} \left(|b_j - \mathbf{r}_j^\top \mathbf{x}| + \|\Sigma_j^{1/2}\mathbf{x}\|_2\right)^2}.
\end{aligned}$$

The last equality holds because

$$\min_{\mathbf{l}_j \Sigma_j^{-1} \mathbf{l}_j} \mathbf{l}_j^\top \mathbf{x} = -\|\Sigma_j^{1/2}\mathbf{x}\|_2; \quad \& \max_{\mathbf{l}_j \Sigma_j^{-1} \mathbf{l}_j} \mathbf{l}_j^\top \mathbf{x} = \|\Sigma_j^{1/2}\mathbf{x}\|_2.$$

□

7.4. Sparsity

In this section, we investigate the sparsity properties of robust regression (7.1), and equivalently Lasso. Lasso's ability to recover sparse solutions has been extensively studied and discussed (cf [**38, 71, 36, 151**]). There are generally two approaches.

The first approach investigates the problem from a statistical perspective. That is, it assumes that the observations are generated by a (sparse) linear combination of the features, and investigates the asymptotic or probabilistic conditions required for Lasso to correctly recover the generative model. The second approach treats the problem from an optimization perspective, and studies under what conditions a pair $(A, \mathbf{b})$ defines a problem with sparse solutions (e.g., [**152**]).

We follow the second approach and do not assume a generative model. Instead, we consider the conditions that lead to a feature receiving zero weight. Our first result paves the way for the remainder of this section. We show in Theorem 7.10 that, essentially, a feature receives no weight (namely, $x_i^* = 0$) if there exists an allowable perturbation of that feature which makes it irrelevant. This result holds for general norm loss functions, but in the ℓ^2 case, we obtain further geometric results. For instance, using Theorem 7.10, we show, among other results, that "nearly" orthogonal features get zero weight (Theorem 7.11).

Substantial research regarding sparsity properties of Lasso can be found in the literature (cf [**38, 71, 36, 151, 78, 42, 104, 58**] and many others). In particular, similar results as in point (a), that rely on an *incoherence* property, have been established in, e.g., [**152**], and are used as standard tools in investigating sparsity of Lasso from the statistical perspective. However, a proof exploiting robustness and properties of the uncertainty is novel. Indeed, such a proof shows a fundamental connection between robustness and sparsity, and implies that robustifying w.r.t. a feature-wise independent uncertainty set might be a plausible way to achieve sparsity for other problems.

To state the main theorem of this section, from which the other results derive, we introduce some notation to facilitate the discussion. Given a feature-wise uncoupled uncertainty set, $\mathcal{U}$, an index subset $I \subseteq \{1, \ldots, n\}$, and any $\Delta A \in \mathcal{U}$, let ΔA^I denote the element of $\mathcal{U}$ that equals ΔA on each feature indexed by $i \in I$, and is zero elsewhere. Then, we can write any element $\Delta A \in \mathcal{U}$ as $\Delta A^I + \Delta A^{I^c}$ (where $I^c = \{1, \ldots, n\} \setminus I$). Then we have the following theorem. We note that the result

holds for any norm loss function, but we state and prove it for the ℓ^2 norm, since the proof for other norms is identical.

THEOREM 7.10. *The robust regression problem*

$$\min_{\mathbf{x}\in\mathbb{R}^m}\left\{\max_{\Delta A\in\mathcal{U}}\|\mathbf{b}-(A+\Delta A)\mathbf{x}\|_2\right\},$$

has a solution supported on an index set I if there exists some perturbation $\Delta\tilde{A}^{I^c}\in\mathcal{U}$ of the features in I^c, such that the robust regression problem

$$\min_{\mathbf{x}\in\mathbb{R}^m}\left\{\max_{\Delta\tilde{A}^I\in\mathcal{U}^I}\|\mathbf{b}-(A+\Delta\tilde{A}^{I^c}+\Delta\tilde{A}^I)\mathbf{x}\|_2\right\},$$

has a solution supported on the set I.

Thus, a robust regression has an optimal solution supported on a set I, if *any* perturbation of the features corresponding to the complement of I makes them irrelevant. Theorem 7.10 is a special case of the following theorem with $c_j = 0$ for all $j \notin I$:

THEOREM 7.10'. *Let $\mathbf{x}^*$ be an optimal solution of the robust regression problem:*

$$\min_{\mathbf{x}\in\mathbb{R}^m}\left\{\max_{\Delta A\in\mathcal{U}}\|\mathbf{b}-(A+\Delta A)\mathbf{x}\|_2\right\},$$

and let $I\subseteq\{1,\cdots,m\}$ be such that $x_j^*=0\ \forall\, j\notin I$. Let

$$\tilde{\mathcal{U}}\triangleq\left\{(\boldsymbol{\delta}_1,\cdots,\boldsymbol{\delta}_m)\middle|\|\boldsymbol{\delta}_i\|_2\le c_i,\ \ i\in I;\ \ \|\boldsymbol{\delta}_j\|_2\le c_j+l_j,\ j\notin I\right\}.$$

Then, $\mathbf{x}^*$ is an optimal solution of

$$\min_{\mathbf{x}\in\mathbb{R}^m}\left\{\max_{\Delta A\in\tilde{\mathcal{U}}}\|\mathbf{b}-(\tilde{A}+\Delta A)\mathbf{x}\|_2\right\},$$

for any $\tilde{A}$ that satisfies $\|\tilde{\mathbf{a}}_j-\mathbf{a}_j\|\le l_j$ for $j\notin I$, and $\tilde{\mathbf{a}}_i=\mathbf{a}_i$ for $i\in I$.

PROOF. Notice that

$$\begin{aligned}
&\max_{\Delta A \in \tilde{\mathcal{U}}} \left\| \mathbf{b} - (A + \Delta A)\mathbf{x}^* \right\|_2 \\
&= \max_{\Delta A \in \mathcal{U}} \left\| \mathbf{b} - (A + \Delta A)\mathbf{x}^* \right\|_2 \\
&= \max_{\Delta A \in \mathcal{U}} \left\| \mathbf{b} - (\tilde{A} + \Delta A)\mathbf{x}^* \right\|_2 .
\end{aligned}$$

These equalities hold because for $j \notin I$, $x_j^* = 0$, hence the j^{th} column of both $\tilde{A}$ and ΔA has no effect on the residual.

For an arbitrary $\mathbf{x}'$, we have

$$\begin{aligned}
&\max_{\Delta A \in \tilde{\mathcal{U}}} \left\| \mathbf{b} - (A + \Delta A)\mathbf{x}' \right\|_2 \\
&\geq \max_{\Delta A \in \mathcal{U}} \left\| \mathbf{b} - (\tilde{A} + \Delta A)\mathbf{x}' \right\|_2 .
\end{aligned}$$

This is because, $\|\mathbf{a}_j - \tilde{\mathbf{a}}_j\| \leq l_j$ for $j \notin I$, and $\mathbf{a}_i = \tilde{\mathbf{a}}_i$ for $i \in I$. Hence, we have

$$\left\{ A + \Delta A \middle| \Delta A \in \mathcal{U} \right\} \subseteq \left\{ \tilde{A} + \Delta A \middle| \Delta A \in \tilde{\mathcal{U}} \right\}.$$

Finally, notice that

$$\max_{\Delta A \in \mathcal{U}} \left\| \mathbf{b} - (A + \Delta A)\mathbf{x}^* \right\|_2 \leq \max_{\Delta A \in \mathcal{U}} \left\| \mathbf{b} - (A + \Delta A)\mathbf{x}' \right\|_2 .$$

Therefore we have

$$\max_{\Delta A \in \tilde{\mathcal{U}}} \left\| \mathbf{b} - (\tilde{A} + \Delta A)\mathbf{x}^* \right\|_2 \leq \max_{\Delta A \in \tilde{\mathcal{U}}} \left\| \mathbf{b} - (\tilde{A} + \Delta A)\mathbf{x}' \right\|_2 .$$

Since this holds for arbitrary $\mathbf{x}'$, we establish the theorem. □

We can interpret the result of this theorem by considering a generative model[1] $b = \sum_{i \in I} w_i a_i + \tilde{\xi}$ where $I \subseteq \{1 \cdots, m\}$ and $\tilde{\xi}$ is a random variable, i.e., b is generated by features belonging to I. In this case, for a feature $j \notin I$, Lasso would assign zero weight as long as there exists a perturbed value of this feature, such that the optimal regression assigned it zero weight.

[1]While we are not assuming generative models to establish the results, it is still interesting to see how these results can help in a generative model setup.

When we consider ℓ^2 loss, we can translate the condition of a feature being "irrelevant" into a geometric condition, namely, orthogonality. We now use the result of Theorem 7.10 to show that robust regression has a sparse solution as long as an incoherence-type property is satisfied. This result is more in line with the traditional sparsity results, but we note that the geometric reasoning is different, and ours is based on robustness. Indeed, we show that a feature receives zero weight, if it is "nearly" (i.e., within an allowable perturbation) orthogonal to the signal, and all relevant features.

THEOREM 7.11. *Let $c_i = c$ for all i and consider ℓ^2 loss. If there exists $I \subset \{1, \cdots, m\}$ such that for all $\mathbf{v} \in \text{span}\big(\{\mathbf{a}_\text{i}, \text{i} \in \text{I}\} \bigcup \{\mathbf{b}\}\big)$, $\|\mathbf{v}\| = 1$, we have $\mathbf{v}^\top \mathbf{a}_j \leq c$, $\forall j \notin I$, then any optimal solution $\mathbf{x}^*$ satisfies $x_j^* = 0$, $\forall j \notin I$.*

PROOF. For $j \notin I$, let $\mathbf{a}_j^=$ denote the projection of $\mathbf{a}_j$ onto the span of $\{\mathbf{a}_i,\ i \in I\} \bigcup \{\mathbf{b}\}$, and let $\mathbf{a}_j^+ \triangleq \mathbf{a}_j - \mathbf{a}_j^=$. Thus, we have $\|\mathbf{a}_j^=\| \leq c$. Let $\hat{A}$ be such that

$$\hat{\mathbf{a}}_i = \begin{cases} \mathbf{a}_i & i \in I; \\ \mathbf{a}_i^+ & i \notin I. \end{cases}$$

Now let

$$\hat{\mathcal{U}} \triangleq \{(\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m) | \|\boldsymbol{\delta}_i\|_2 \leq c,\, i \in I; \|\boldsymbol{\delta}_j\|_2 = 0,\, j \notin I\}.$$

Consider the robust regression problem $\min_{\hat{\mathbf{x}}} \left\{ \max_{\Delta A \in \hat{\mathcal{U}}} \left\|\mathbf{b} - (\hat{A} + \Delta A)\hat{\mathbf{x}}\right\|_2 \right\}$, which is equivalent to $\min_{\hat{\mathbf{x}}} \left\{ \left\|\mathbf{b} - \hat{A}\hat{\mathbf{x}}\right\|_2 + \sum_{i \in I} c|\hat{x}_i| \right\}$. Note that the $\hat{\mathbf{a}}_j$ are orthogonal to the span of $\{\hat{\mathbf{a}}_i,\ i \in I\} \bigcup \{\mathbf{b}\}$. Hence for any given $\hat{\mathbf{x}}$, by changing $\hat{x}_j$ to zero for all $j \notin I$, the minimizing objective does not increase.

Since $\|\hat{\mathbf{a}} - \hat{\mathbf{a}}_j\| = \|\mathbf{a}_j^=\| \leq c\ \forall j \notin I$, (and recall that $\mathcal{U} = \{(\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m) | \|\boldsymbol{\delta}_i\|_2 \leq c, \forall i\}$) applying Theorem 7.10 concludes the proof. □

7.5. Density estimation and consistency

In this section, we investigate the robust linear regression formulation from a statistical perspective and rederive *using only robustness properties* that Lasso is

asymptotically consistent. The basic idea of the consistency proof is as follows. We show that the robust optimization formulation can be seen to be the maximum error w.r.t. a class of probability measures. This class includes a kernel density estimator, and using this, we show that Lasso is consistent.

7.5.1. Robust optimization, worst-case expected utility and kernel density estimator. In this subsection, we present some notions and intermediate results. In particular, we link a robust optimization formulation with a worst expected utility (w.r.t. a class of probability measures); we then briefly recall the definition of a kernel density estimator. Such results will be used in establishing the consistency of Lasso, as well as providing some additional insights on robust optimization.

We recall a result on the equivalence between a robust optimization formulation and a worst-case expected utility from Chapter 2:

PROPOSITION 7.12. *Given a function* $g : \mathbb{R}^{m+1} \to \mathbb{R}$ *and Borel sets* $\mathcal{Z}_1, \cdots, \mathcal{Z}_n \subseteq \mathbb{R}^{m+1}$*, let*

$$\mathcal{P}_n \triangleq \{\mu \in \mathcal{P} | \forall S \subseteq \{1, \cdots, n\} : \mu(\bigcup_{i \in S} \mathcal{Z}_i) \geq |S|/n\}.$$

The following holds

$$\frac{1}{n} \sum_{i=1}^{n} \sup_{(\mathbf{r}_i, b_i) \in \mathcal{Z}_i} h(\mathbf{r}_i, b_i) = \sup_{\mu \in \mathcal{P}_n} \int_{\mathbb{R}^{m+1}} h(\mathbf{r}, b) d\mu(\mathbf{r}, b).$$

This leads to the following corollary for Lasso, which states that for a given $\mathbf{x}$, the robust regression loss over the training data is equal to the worst-case expected *generalization error*.

COROLLARY 7.13. *Given* $\mathbf{b} \in \mathbb{R}^n$*,* $A \in \mathbb{R}^{n \times m}$*, the following equation holds for any* $\mathbf{x} \in \mathbb{R}^m$*,*

$$\|\mathbf{b} - A\mathbf{x}\|_2 + \sqrt{n} c_n \|\mathbf{x}\|_1 + \sqrt{n} c_n = \sup_{\mu \in \hat{\mathcal{P}}(n)} \sqrt{n \int_{\mathbb{R}^{m+1}} (b' - \mathbf{r}'^{\top} \mathbf{x})^2 d\mu(\mathbf{r}', b')}. \quad (7.12)$$

Here[2]*,*

$$\hat{\mathcal{P}}(n) \triangleq \bigcup_{\|\boldsymbol{\sigma}\|_2 \leq \sqrt{n}c_n; \forall i: \|\boldsymbol{\delta}_i\|_2 \leq \sqrt{n}c_n} \mathcal{P}_n(A, \Delta, \mathbf{b}, \boldsymbol{\sigma});$$

$$\mathcal{P}_n(A, \Delta, \mathbf{b}, \boldsymbol{\sigma}) \triangleq \{\mu \in \mathcal{P} | \mathcal{Z}_i = [b_i - \sigma_i, b_i + \sigma_i] \times \prod_{j=1}^{m} [a_{ij} - \delta_{ij}, a_{ij} + \delta_{ij}];$$

$$\forall S \subseteq \{1, \cdots, n\} : \mu(\bigcup_{i \in S} \mathcal{Z}_i) \geq |S|/n\}.$$

REMARK 7.1. Before proving Corollary 7.13, we briefly explain to avoid possible confusion. Equation (7.12) is a non-probabilistic equality. That is, it holds without any assumption (e.g., i.i.d. or generated by certain distributions) on $\mathbf{b}$ and A. And it does not involve any probabilistic operation such as taking expectation on the left-hand-side, instead, it is an equivalence relationship which hold for an arbitrary set of samples. Note that the right-hand-side also depends on the samples since $\hat{\mathcal{P}}(n)$ is defined through A and $\mathbf{b}$. Indeed, $\hat{\mathcal{P}}(n)$ represents the union of classes of distributions $\mathcal{P}_n(A, \Delta, \mathbf{b}, \boldsymbol{\sigma})$ such that the norm of each column of Δ is bounded, where $\mathcal{P}_n(A, \Delta, \mathbf{b}, \boldsymbol{\sigma})$ is the set of distributions corresponds to (see Proposition 7.12) disturbance in hyper-rectangle Borel sets $\mathcal{Z}_1, \cdots, \mathcal{Z}_n$ centered at $(b_i, \mathbf{r}_i^\top)$ with lengths $(2\sigma_i, 2\delta_{i1}, \cdots, 2\delta_{im})$.

PROOF. The right-hand-side of Equation (7.12) equals

$$\sup_{\|\boldsymbol{\sigma}\|_2 \leq \sqrt{n}c_n; \forall i: \|\boldsymbol{\delta}_i\|_2 \leq \sqrt{n}c_n} \Big\{ \sup_{\mu \in \mathcal{P}_n(A, \Delta, \mathbf{b}, \boldsymbol{\sigma})} \sqrt{n \int_{\mathbb{R}^{m+1}} (b' - \mathbf{r}'^\top \mathbf{x})^2 d\mu(\mathbf{r}', b')} \Big\}.$$

[2]Recall that a_{ij} is the j^{th} element of $\mathbf{r}_i$

Notice that by the equivalence to robust formulation, the left-hand-side equals to

$$\begin{aligned}
&\max_{\|\boldsymbol{\sigma}\|_2\le\sqrt{n}c_n;\forall i:\|\boldsymbol{\delta}_i\|_2\le\sqrt{n}c_n} \Big\|\mathbf{b}+\boldsymbol{\sigma}-\big(A+[\boldsymbol{\delta}_1,\cdots,\boldsymbol{\delta}_m]\big)\mathbf{x}\Big\|_2 \\
=&\sup_{\|\boldsymbol{\sigma}\|_2\le\sqrt{n}c_n;\forall i:\|\boldsymbol{\delta}_i\|_2\le\sqrt{n}c_n} \left\{\sup_{(\hat{b}_i,\hat{\mathbf{r}}_i)\in[b_i-\sigma_i,b_i+\sigma_i]\times\prod_{j=1}^m[a_{ij}-\delta_{ij},a_{ij}+\delta_{ij}]} \sqrt{\sum_{i=1}^n(\hat{b}_i-\hat{\mathbf{r}}_i^\top\mathbf{x})^2}\right\} \\
=&\sup_{\|\boldsymbol{\sigma}\|_2\le\sqrt{n}c_n;\forall i:\|\boldsymbol{\delta}_i\|_2\le\sqrt{n}c_n} \sqrt{\sum_{i=1}^n\sup_{(\hat{b}_i,\hat{\mathbf{r}}_i)\in[b_i-\sigma_i,b_i+\sigma_i]\times\prod_{j=1}^m[a_{ij}-\delta_{ij},a_{ij}+\delta_{ij}]}(\hat{b}_i-\hat{\mathbf{r}}_i^\top\mathbf{x})^2}.
\end{aligned}$$

Furthermore, applying Proposition 7.12 yields

$$\begin{aligned}
&\sqrt{\sum_{i=1}^n\sup_{(\hat{b}_i,\hat{\mathbf{r}}_i)\in[b_i-\sigma_i,b_i+\sigma_i]\times\prod_{j=1}^m[a_{ij}-\delta_{ij},a_{ij}+\delta_{ij}]}(\hat{b}_i-\hat{\mathbf{r}}_i^\top\mathbf{x})^2} \\
=&\sqrt{\sup_{\mu\in\mathcal{P}_n(A,\Delta,\mathbf{b},\boldsymbol{\sigma})} n\int_{\mathbb{R}^{m+1}}(b'-\mathbf{r}'^\top\mathbf{x})^2d\mu(\mathbf{r}',b')} \\
=&\sup_{\mu\in\mathcal{P}_n(A,\Delta,\mathbf{b},\boldsymbol{\sigma})}\sqrt{n\int_{\mathbb{R}^{m+1}}(b'-\mathbf{r}'^\top\mathbf{x})^2d\mu(\mathbf{r}',b')},
\end{aligned}$$

which proves the corollary. □

We will later show that the set $\hat{P}_n$ includes a kernel density estimator. Hence we recall here its definition. The *kernel density estimator* for a density $\hat{h}$ in $\mathbb{R}^d$, originally proposed in [**127, 118**], is defined by

$$h_n(\mathbf{x})=(nc_n^d)^{-1}\sum_{i=1}^n K\left(\frac{\mathbf{x}-\hat{\mathbf{x}}_i}{c_n}\right),$$

where $\{c_n\}$ is a sequence of positive numbers, $\hat{\mathbf{x}}_i$ are i.i.d. samples generated according to $\hat{f}$, and K is a Borel measurable function (kernel) satisfying $K\ge 0$, $\int K=1$. See [**53, 135**] and the reference therein for detailed discussions. Figure 7.1 illustrates a kernel density estimator using Gaussian kernel for a randomly generated sample-set. A celebrated property of a kernel density estimator is that it converges in $\mathcal{L}^1$ to $\hat{h}$ when $c_n\downarrow 0$ and $nc_n^d\uparrow\infty$ [**53**].

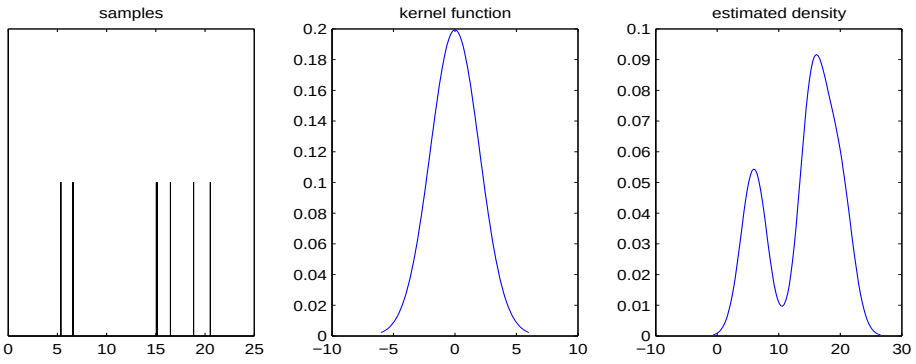


FIGURE 7.1. Illustration of a *Kernel Density Estimator*.

7.5.2. Consistency of Lasso. We restrict our discussion to the case where the magnitude of the allowable uncertainty for all features equals c, (i.e., the standard Lasso) and establish the statistical consistency of Lasso from a distributional robustness argument. Generalization to the non-uniform case is straightforward. Throughout, we use c_n to represent c where there are n samples (we take c_n to zero).

Recall the standard generative model in statistical learning: let $\mathbb{P}$ be a probability measure with bounded support that generates i.i.d samples $(b_i, \mathbf{r}_i)$, and has a density $f^*(\cdot)$. Denote the set of the first n samples by $\mathcal{S}_n$. Define

$$\begin{aligned}\mathbf{x}(c_n, \mathcal{S}_n) &\triangleq \arg\min_{\mathbf{x}} \left\{ \sqrt{\frac{1}{n}\sum_{i=1}^{n}(b_i - \mathbf{r}_i^\top \mathbf{x})^2} + c_n\|x\|_1 \right\} \\ &= \arg\min_{\mathbf{x}} \left\{ \frac{\sqrt{n}}{n} \sqrt{\sum_{i=1}^{n}(b_i - \mathbf{r}_i^\top \mathbf{x})^2} + c_n\|x\|_1 \right\}; \\ \mathbf{x}(\mathbb{P}) &\triangleq \arg\min_{\mathbf{x}} \left\{ \sqrt{\int_{b,\mathbf{r}} (b - \mathbf{r}^\top \mathbf{x})^2 d\mathbb{P}(b, \mathbf{r})} \right\}.\end{aligned}$$

In words, $\mathbf{x}(c_n, \mathcal{S}_n)$ is the solution to Lasso with the tradeoff parameter set to $c_n\sqrt{n}$, and $\mathbf{x}(\mathbb{P})$ is the "true" optimal solution. We have the following consistency result. The theorem itself is a well-known result. However, the proof technique is novel. This technique is of interest because the standard techniques to establish consistency in statistical learning including Vapnik-Chervonenkis (VC) dimension (e.g., [**158**]) and

algorithmic stability (e.g., [**32**]) often work for a limited range of algorithms, e.g., the k-Nearest Neighbor is known to have infinite VC dimension, and we show in Section 7.6 that *Lasso is not stable*. In contrast, a much wider range of algorithms have robustness interpretations, allowing a unified approach to prove their consistency.

THEOREM 7.14. *Let* $\{c_n\}$ *be such that* $c_n \downarrow 0$ *and* $\lim_{n\to\infty} n(c_n)^{m+1} = \infty$. *Suppose there exists a constant* H *such that* $\|\mathbf{x}(c_n, \mathcal{S}_n)\|_2 \leq H$. *Then,*

$$\lim_{n\to\infty} \sqrt{\int_{b,\mathbf{r}} (b - \mathbf{r}^\top \mathbf{x}(c_n, \mathcal{S}_n))^2 d\mathbb{P}(b, \mathbf{r})} = \sqrt{\int_{b,\mathbf{r}} (b - \mathbf{r}^\top \mathbf{x}(\mathbb{P}))^2 d\mathbb{P}(b, \mathbf{r})},$$

almost surely.

PROOF. **Step 1**: We show that the right hand side of Equation (7.12) includes a kernel density estimator for the true (unknown) distribution. Consider the following kernel estimator given samples $\mathcal{S}_n = (b_i, \mathbf{r}_i)_{i=1}^n$ and tradeoff parameter c_n,

$$\begin{aligned} f_n(b, \mathbf{r}) &\triangleq (nc_n^{m+1})^{-1} \sum_{i=1}^n K\left(\frac{b - b_i, \mathbf{r} - \mathbf{r}_i}{c_n}\right), \\ &\text{where: } K(\mathbf{x}) \triangleq I_{[-1,+1]^{m+1}}(\mathbf{x})/2^{m+1}. \end{aligned} \tag{7.13}$$

Let $\hat{\mu}_n$ denote the distribution given by the density function $f_n(b, \mathbf{r})$. Easy to check that $\hat{\mu}_n$ belongs to $\mathcal{P}_n(A, (c_n\mathbf{1}_n, \cdots, c_n\mathbf{1}_n), \mathbf{b}, c_n\mathbf{1}_n)$ and hence belongs to $\hat{\mathcal{P}}(n)$ by definition.

Step 2: Using the $\mathcal{L}^1$ convergence property of the kernel density estimator, we prove the consistency of robust regression and equivalently Lasso.

First note that $\|\mathbf{x}(c_n, \mathcal{S}_n)\|_2 \leq H$ and $\mathbb{P}$ has a bounded support implies that there exists a universal constant C such that

$$\max_{b,\mathbf{r}} (b - \mathbf{r}^\top \mathbf{w}(c_n, \mathcal{S}_n))^2 \leq C.$$

By Corollary 7.13 and $\hat{\mu}_n \in \hat{\mathcal{P}}(n)$ we have

$$\begin{aligned}
&\sqrt{\int_{b,\mathbf{r}} (b - \mathbf{r}^\top \mathbf{x}(c_n, \mathcal{S}_n))^2 d\hat{\mu}_n(b, \mathbf{r})} \\
\leq &\sup_{\mu \in \hat{\mathcal{P}}(n)} \sqrt{\int_{b,\mathbf{r}} (b - \mathbf{r}^\top \mathbf{x}(c_n, \mathcal{S}_n))^2 d\mu(b, \mathbf{r})} \\
= &\frac{\sqrt{n}}{n} \sqrt{\sum_{i=1}^{n} (b_i - \mathbf{r}_i^\top \mathbf{x}(c_n, \mathcal{S}_n))^2} + c_n \|\mathbf{x}(c_n, \mathcal{S}_n)\|_1 + c_n \\
\leq &\frac{\sqrt{n}}{n} \sqrt{\sum_{i=1}^{n} (b_i - \mathbf{r}_i^\top \mathbf{x}(\mathbb{P}))^2} + c_n \|\mathbf{x}(\mathbb{P})\|_1 + c_n,
\end{aligned}$$

the last inequality holds by definition of $\mathbf{x}(c_n, \mathcal{S}_n)$.

Taking the square of both sides, we have

$$\begin{aligned}
&\int_{b,\mathbf{r}} (b - \mathbf{r}^\top \mathbf{x}(c_n, \mathcal{S}_n))^2 d\hat{\mu}_n(b, \mathbf{r}) \\
\leq &\frac{1}{n} \sum_{i=1}^{n} (b_i - \mathbf{r}_i^\top \mathbf{x}(\mathbb{P}))^2 + c_n^2 (1 + \|\mathbf{x}(\mathbb{P})\|_1)^2 \\
&\qquad + 2c_n (1 + \|\mathbf{x}(\mathbb{P})\|_1) \sqrt{\frac{1}{n} \sum_{i=1}^{n} (b_i - \mathbf{r}_i^\top \mathbf{x}(\mathbb{P}))^2}.
\end{aligned}$$

Note that the right-hand side converges to $\int_{b,\mathbf{r}} (b - \mathbf{r}^\top \mathbf{x}(\mathbb{P}))^2 d\mathbb{P}(b, \mathbf{r})$ as $n \uparrow \infty$ and $c_n \downarrow 0$ almost surely. Furthermore, we have

$$\begin{aligned}
&\int_{b,\mathbf{r}} (b - \mathbf{r}^\top \mathbf{x}(c_n, \mathcal{S}_n))^2 d\mathbb{P}(b, \mathbf{r}) \\
\leq &\int_{b,\mathbf{r}} (b - \mathbf{r}^\top \mathbf{x}(c_n, \mathcal{S}_n))^2 d\hat{\mu}_n(b, \mathbf{r}) \\
&\qquad + \big[\max_{b,\mathbf{r}} (b - \mathbf{r}^\top \mathbf{x}(c_n, \mathcal{S}_n))^2 \big] \int_{b,\mathbf{r}} |f_n(b, \mathbf{r}) - f^*(b, \mathbf{r})| d(b, \mathbf{r}) \\
\leq &\int_{b,\mathbf{r}} (b - \mathbf{r}^\top \mathbf{x}(c_n, \mathcal{S}_n))^2 d\hat{\mu}_n(b, \mathbf{r}) + C \int_{b,\mathbf{r}} |f_n(b, \mathbf{r}) - f^*(b, \mathbf{r})| d(b, \mathbf{r}),
\end{aligned}$$

where the last inequality follows from the definition of C. Notice that $\int_{b,\mathbf{r}} |f_n(b,\mathbf{r}) - f^*(b,\mathbf{r})| d(b,\mathbf{r})$ goes to zero almost surely when $c_n \downarrow 0$ and $nc_n^{m+1} \uparrow \infty$ since $f_n(\cdot)$ is a kernel density estimation of $f^*(\cdot)$ (see e.g. Theorem 3.1 of [**53**]). Hence the theorem follows. □

We can remove the assumption that $\|\mathbf{x}(c_n, \mathcal{S}_n)\|_2 \leq H$, and as in Theorem 7.14, the proof technique rather than the result itself is of interest.

THEOREM 7.15. *Let $\{c_n\}$ converge to zero sufficiently slowly. Then*

$$\lim_{n\to\infty} \sqrt{\int_{b,\mathbf{r}} (b - \mathbf{r}^\top \mathbf{x}(c_n, \mathcal{S}_n))^2 d\mathbb{P}(b,\mathbf{r})} = \sqrt{\int_{b,\mathbf{r}} (b - \mathbf{r}^\top \mathbf{x}(\mathbb{P}))^2 d\mathbb{P}(b,\mathbf{r})},$$

almost surely.

PROOF. To prove the theorem, we need to consider a set of distributions belonging to $\hat{\mathcal{P}}(n)$. Hence we establish the following lemma first.

LEMMA 7.16. *Partition the support of $\mathbb{P}$ as $V_1, \cdots, V_T$ such the ℓ^∞ radius of each set is less than c_n. If a distribution μ satisfies*

$$\mu(V_t) = \Big|\big\{i | (b_i, \mathbf{r}_i) \in V_t\big\}\Big|/n; \quad t = 1, \cdots, T, \tag{7.14}$$

then $\mu \in \hat{\mathcal{P}}(n)$.

PROOF. Let $\mathcal{Z}_i = [b_i - c_n, b_i + c_n] \times \prod_{j=1}^m [a_{ij} - c_n, a_{ij} + c_n]$; recall that a_{ij} the j^{th} element of $\mathbf{r}_i$. Notice V_t has ℓ^∞ norm less than c_n we have

$$(b_i, \mathbf{r}_i \in V_t) \Rightarrow V_t \subseteq \mathcal{Z}_i.$$

Therefore, for any $S \subseteq \{1, \cdots, n\}$, the following holds

$$\begin{aligned} &\mu\Big(\bigcup_{i\in S} \mathcal{Z}_i\Big) \geq \mu\Big(\bigcup V_t | \exists i \in S : b_i, \mathbf{r}_i \in V_t\Big) \\ &= \sum_{t | \exists i \in S : b_i, \mathbf{r}_i \in V_t} \mu(V_t) = \sum_{t | \exists i \in S : b_i, \mathbf{r}_i \in V_t} \#\big((b_i, \mathbf{r}_i) \in V_t\big)/n \geq |S|/n. \end{aligned}$$

Hence $\mu \in \mathcal{P}_n(A, \Delta, b, c_n)$ where each element of Δ is c_n, which leads to $\mu \in \hat{\mathcal{P}}(n)$. □

Now we proceed to prove the theorem. Partition the support of $\mathbb{P}$ into T subsets such that ℓ^∞ radius of each one is smaller than c_n. Denote $\tilde{\mathcal{P}}(n)$ as the set of probability measures satisfying Equation (7.14). Hence $\tilde{\mathcal{P}}(n) \subseteq \hat{\mathcal{P}}(n)$ by Lemma 7.16. Further notice that there exists a universal constant K such that $\|\mathbf{x}(c_n, \mathcal{S}_n)\|_2 \leq K/c_n$ due to the fact that the square loss of the solution $\mathbf{x} = \mathbf{0}$ is bounded by a constant only depends on the support of $\mathbb{P}$. Thus, there exists a constant C such that $\max_{b,\mathbf{r}}(b - \mathbf{r}^\top \mathbf{x}(c_n, \mathcal{S}_n))^2 \leq C/c_n^2$.

Follow a similar argument as the proof of Theorem 7.14, we have

$$\begin{aligned}
&\sup_{\mu_n \in \tilde{\mathcal{P}}(n)} \int_{b,\mathbf{r}} (b - \mathbf{r}^\top \mathbf{x}(c_n, \mathcal{S}_n))^2 d\mu_n(b, \mathbf{r}) \\
\leq &\frac{1}{n} \sum_{i=1}^n (b_i - \mathbf{r}_i^\top \mathbf{x}(\mathbb{P}))^2 + c_n^2 (1 + \|\mathbf{x}(\mathbb{P})\|_1)^2 \\
&\qquad + 2c_n (1 + \|\mathbf{x}(\mathbb{P})\|_1) \sqrt{\frac{1}{n} \sum_{i=1}^n (b_i - \mathbf{r}_i^\top \mathbf{x}(\mathbb{P}))^2},
\end{aligned} \tag{7.15}$$

and

$$\begin{aligned}
&\int_{b,\mathbf{r}} (b - \mathbf{r}^\top \mathbf{x}(c_n, \mathcal{S}_n))^2 d\mathbb{P}(b, \mathbf{r}) \\
\leq &\inf_{\mu_n \in \tilde{\mathcal{P}}(n)} \Big\{ \int_{b,\mathbf{r}} (b - \mathbf{r}^\top \mathbf{x}(c_n, \mathcal{S}_n))^2 d\mu_n(b, \mathbf{r}) \\
&\qquad + \max_{b,\mathbf{r}} (b - \mathbf{r}^\top \mathbf{x}(c_n, \mathcal{S}_n))^2 \int_{b,\mathbf{r}} |f_{\mu_n}(b, \mathbf{r}) - f(b, \mathbf{r})| d(b, \mathbf{r}) \\
\leq &\sup_{\mu_n \in \tilde{\mathcal{P}}(n)} \int_{b,\mathbf{r}} (b - \mathbf{r}^\top \mathbf{x}(c_n, \mathcal{S}_n))^2 d\mu_n(b, \mathbf{r}) \\
&\qquad + 2C/c_n^2 \inf_{\mu_n' \in \tilde{\mathcal{P}}(n)} \Big\{ \int_{b,\mathbf{r}} |f_{\mu_n'}(b, \mathbf{r}) - f(b, \mathbf{r})| d(b, \mathbf{r}) \Big\},
\end{aligned}$$

here f_μ stands for the density function of a measure μ. Notice that $\tilde{\mathcal{P}}_n$ is the set of distributions satisfying Equation (7.14), hence $\inf_{\mu_n' \in \tilde{\mathcal{P}}(n)} \int_{b,\mathbf{r}} |f_{\mu_n'}(b, \mathbf{r}) - f(b, \mathbf{r})| d(b, \mathbf{r})$ is upper-bounded by $\sum_{t=1}^T |\mathbb{P}(V_t) - \#(b_i, \mathbf{r}_i \in V_t)|/n$, which goes to zero as n increases

for any fixed c_n (see for example Proposition A6.6 of [**154**]). Therefore,

$$2C/c_n^2 \inf_{\mu'_n \in \tilde{\mathcal{P}}(n)} \left\{ \int_{b,\mathbf{r}} |f_{\mu'_n}(b,\mathbf{r}) - f(b,\mathbf{r})| d(b,\mathbf{r}) \right\} \to 0,$$

if $c_n \downarrow 0$ sufficiently slow. Combining this with Inequality (7.15) proves the theorem. □

7.6. Stability

Knowing that the robust regression problem (7.1) and in particular Lasso encourage sparsity, it is of interest to investigate another desirable characteristic of a learning algorithm, namely, stability. Indeed, it can be shown that Lasso *is not stable*. This is a special case of a more general result that will be presented in Chapter 9, and hence to avoid replication we refer the readers to Chapter 9.

7.7. Chapter summary

In this chapter, we considered robust regression with a least-square-error loss. In contrast to previous work on robust regression, we considered the case where the perturbations of the observations are in the features. We show that this formulation is equivalent to a weighted ℓ^1 norm regularized regression problem if no correlation of disturbances among different features is allowed, and hence provide an interpretation of the widely used Lasso algorithm from a robustness perspective. We also formulated tractable robust regression problems for disturbance coupled among different features and hence generalize Lasso to a wider class of regularization schemes.

The sparsity and consistency of Lasso are also investigated based on its robustness interpretation. In particular we present a "no-free-lunch" theorem saying that sparsity and algorithmic stability contradict each other. This result shows, although sparsity and algorithmic stability are both regarded as desirable properties of regression algorithms, it is not possible to achieve them simultaneously, and we have to tradeoff these two properties in designing a regression algorithm.

The main thrust of this work is to treat the widely used regularized regression scheme from a robust optimization perspective, and extend the result of [**64**] (i.e., Tikhonov regularization is equivalent to a robust formulation for Frobenius norm bounded disturbance set) to a broader range of disturbance set and hence regularization scheme. This provides us not only with new insight of why regularization schemes work, but also offer solid motivations for selecting regularization parameter for existing regularization scheme and facilitate designing new regularizing schemes.

CHAPTER 8

All Learning is Robust: On the Equivalence of Robustness and Generalizability

As shown in Chapter 6 and Chapter 7, some successfully implemented learning algorithms have nice robustness properties. In fact, in this chapter we show that such a relationship is not a coincidence: for an arbitrary learning algorithm, robustness is a necessary and sufficient condition for it to work. In particular: We consider robustness of learning algorithms and prove that robustness is a necessary and sufficient condition for learning algorithms to generalize. To the best of our knowledge, this is the first "if-and-only-if" condition for the generalizability of learning algorithms other than empirical risk minimization. We provide conditions that ensure robustness and hence generalizability for samples that are independent and identically distributed and for samples that come from a Markov chain. Our results lead to new theorems of generalizability as well as novel proofs of known results.

8.1. Introduction

In supervised learning—the task of learning a mapping given a set of observed input-output pairs—the key property of a learning algorithm is it *generalizability*: the expected performance should agree with the empirical error as the number of training

samples increases. An algorithm with good generalizability is guaranteed to predict well if the empirical error is small. In particular, if a learning algorithm achieves minimal training error asymptotically (e.g., Empirical Risk Minimization (ERM)) and generalizes, then it is *consistent*: the expected risk on test data converges to the minimum risk achievable. Roughly speaking, this means that the algorithm recovers the optimal solution in the long run.

One of the most prominent approaches examining generalizability is based on the uniform convergence of empirical quantities to their mean (e.g., [**157, 155**]). This approach provides ways to bound the gap between the risk on a test set and the empirical risk on a training set by the complexity of the space of learned mappings. Examples to complexity measures are the Vapnik-Chervonenkis (VC) dimension (e.g., [**155, 70**]), the fat-shattering dimension (e.g., [**1, 6**]), and the Rademacher complexity ([**8, 7**]).

Another well-known approach is based on *stability*. An algorithm is stable if its output remains "similar" for different sets of training samples that are identical up to removal or change of a single sample. In contrast to the complexity-based approach that focuses on the space that an algorithm searches, stability analysis concentrates on *how* the algorithm searches the space. The first results that relate stability to generalizability track back to [**55**] and [**56**] that obtained bounds of generalization error for "local" algorithms such as k-Nearst Neighbor (k-NN). Later, McDiarmid's [**110**], concentration inequalities facilitated new bounds on generalization error (e.g., [**52, 32, 100, 122, 112**]).

Both aforementioned approaches provide sufficient but not necessary conditions for generalizability. It is easy to construct generalizable algorithms that have unbounded complexity (e.g., k-NN) or are unstable (e.g., it is shown in [**100**] that all classification algorithms with good generalizability are not uniformly stable). Indeed, to the best of our knowledge, a necessary and sufficient condition of generalizability for general learning algorithms has not been suggested in the literature. A notable exception is the ERM algorithm, where it is known that both having a finite VC-dimension

[**158**] and being CVEEE$_{loo}$ stable [**112**] are necessary and sufficient conditions for an ERM algorithm to generalize. However, the class of ERM algorithms is restrictive, and does not include many algorithms that are successful in practice such as k-NN, Support Vector Machines (SVM) [**133**] and Boosting [**132, 75**].

In this chapter we investigate generalizability based on the *robustness* of a learning algorithm. We show that robustness is a *necessary and sufficient* condition for the generalizability of a learning algorithm. Roughly speaking, an algorithm is robust if the solutions it produces achieve "similar" performance on testing samples that are "close" to the training samples.[1] This notion was first introduced to handle exogenous noise in learning (e.g., [**27, 137, 79**]). Recently, it was discovered that regularized algorithms such as support vector machines [**168**] and Lasso [**167**] have desirable robustness properties that further imply statistical consistency. Such an observation motivated us to explore a more fundamental relationship between robustness and generalizability. In particular, our main contributions include the following:

- We propose in section 8.2 a notion of robustness for learning algorithms and show that our notion is a necessary and sufficient condition for generalizability. Our basic result holds in a very general setup with essentially no explicit assumptions on the data source.
- We discuss the problem of how to establish that a given learning algorithm is robust in Section 8.3, under an iid assumption. We consider two different conditions: (1) the solution produced by an algorithm belongs to a function class with a finite bracketing number; and (2) the solution is "smooth" in the training samples; and show that either condition implies robustness of the algorithm.
- We demonstrate in Section 8.4 the relative simplicity of using robustness to establish generalizability when the data source is not IID. We do that by

[1]While stability and robustness are similar on an intuitive level, there is a difference between the two: stability requires that similar training sets lead to similar prediction rules, whereas robustness requires that a prediction rule has comparable performance if tested on a sample set close to the training set.

investigating the case where data are sampled from a Markov chain. We show that in that case, the equivalence of robustness and generalizability still holds and that smoothness directly implies robustness.

8.1.1. Preliminaries and notations. We consider a general supervised learning setup: a learning algorithm outputs a predictor (classifiers are considered to be a special case in our analysis) given a set of training samples, and evaluates the model based on a newly generated testing sample set. To make this precise, we will use the following terminology: Let $\mathcal{X}$ and $\mathcal{Y}$ be two sets, a *prediction rule* $\mathbb{O}$ is a mapping from $\mathcal{X}$ to $\mathcal{Y}$. A *learning algorithm* $\mathbb{A}$ is a mapping from $\bigcup_{n=1}^{\infty}(\mathcal{X}\times\mathcal{Y})^n$ to the set of prediction rules, and we use $\mathbb{A}(\mathcal{S})$ to represent the prediction rule generated by $\mathbb{A}$ given the sample set $\mathcal{S}\in\bigcup_{n=1}^{\infty}(\mathcal{X}\times\mathcal{Y})^n$. A prediction rule $\mathbb{O}$ is called *admissible* (w.r.t. $\mathbb{A}$) if there exists $\mathcal{S}\in\bigcup_{n=1}^{\infty}(\mathcal{X}\times\mathcal{Y})^n$ such that $\mathbb{O}=\mathbb{A}(\mathcal{S})$. We use a subscript n when the cardinality of the sample set is n, i.e., $\mathcal{S}_n\in(\mathcal{X}\times\mathcal{Y})^n$. We alternatively use $\{\mathcal{S}_n\}$ or $\{\mathcal{S}_n\}_{n=1}^{\infty}$ to represent an increasing sequence of samples, i.e., $\mathcal{S}_1\in\mathcal{X}\times\mathcal{Y}$, and for all $i\geq 2$, $\mathcal{S}_i=(\mathcal{S}_{i-1},\mathbf{s}_i)$ where $\mathbf{s}_i\in\mathcal{X}\times\mathcal{Y}$. Given a prediction rule $\mathbb{O}$ and a set of samples $\mathcal{T}\in\bigcup_{n=1}^{\infty}(\mathcal{X}\times\mathcal{Y})^n$ the corresponding loss is denoted by $\mathcal{L}(\mathbb{O},\mathcal{T})$ (we consider loss functions that depend on the whole sample in a general way). Thus, $\mathcal{L}(\mathbb{A}(\mathcal{S}),\mathcal{T})$ is the loss over $\mathcal{T}$ for a learning algorithm $\mathbb{A}$ trained on $\mathcal{S}$. We use $\mathcal{S}$ and $\mathcal{T}$ (sometimes with a subscript n) to denote the testing samples and training samples, respectively. We denote the probability distribution that generates $\{\mathcal{T}_n\}_{n=1}^{\infty}$ by μ, and use μ_n to represent the marginal of μ on the first n samples. We say $\mathcal{L}(\cdot,\cdot)$ is the average loss of $l(\cdot,\cdot)$ if for all n, admissible $\mathbb{O}$ and $\mathcal{T}_n=(\mathbf{t}_1,\cdots,\mathbf{t}_n)\in(\mathcal{X}\times\mathcal{Y})^n$,

$$\mathcal{L}(\mathbb{O},\mathcal{T}_n)=\frac{1}{n}\sum_{i=1}^{n}l(\mathbb{O},\mathbf{t}_i).$$

We ignore the issue of measurability and assume that all functions and sets being considered are measurable w.r.t. corresponding probability spaces.

Notice that convergence results are not possible for $\mathcal{L}$ that are not integrable since laws of large numbers fail to hold. Thus, certain assumptions on the loss function

is necessary. In particular, we define the following two conditions, namely *uniform boundedness* and *uniform envelopedness*. Observe that the former implies the latter.

DEFINITION 8.1. *Loss function $\mathcal{L}(\cdot,\cdot)$ is called* uniformly bounded *if it is non-negative and there exists a constant C such that $\mathcal{L}(\mathbb{O},\mathcal{T}_n) \le C$ for all n, admissible $\mathbb{O}$ and $\mathcal{T}_n \in (\mathcal{X}\times\mathcal{Y})^n$.*

DEFINITION 8.2. *Loss function $\mathcal{L}(\cdot,\cdot)$ is called* uniformly enveloped *if it is non-negative and for each n, there exists f_n such that $\mathcal{L}(\mathbb{O},\mathcal{T}_n) \le f_n(\mathcal{T}_n)$ for all admissible $\mathbb{O}$ and $\mathcal{T}_n \in (\mathcal{X}\times\mathcal{Y})^n$, and*

$$\lim_{M\to\infty}\left\{\limsup_n \mathbb{E}\Big\{f_n(\mathcal{T}_n)\cdot\mathbf{1}\big[f_n(\mathcal{T}_n) > M\big]\Big\}\right\} = 0.$$

Here the expectation is taken over different draws of $\{\mathcal{T}_n\}$.

In the standard setup where the loss function is the average loss of $l(\cdot,\cdot)$, it is often assumed that $l(\cdot,\cdot)$ is upper bounded by an integrable function $f(\cdot)$, so that the average loss exists. The uniform envelopedness is an extension of such a condition when the $\mathcal{L}$ is not necessarily an average loss. In particular, the aforementioned condition is uniform enveloped as we will show in Corollary 8.4.

8.2. Robustness and generalizability

In this section we prove the main result of this paper: *generalizability* of an algorithm is equivalent to the *robustness* of its output prediction rules. To make this precise, we first define these concepts.

DEFINITION 8.3. *Given $\{\mathcal{S}_n\}$, Algorithm $\mathbb{A}$* generalizes *w.r.t. $\{\mathcal{S}_n\}$ if*

$$\limsup_n \Big\{\mathbb{E}_{\mathcal{T}_n}\Big(\mathcal{L}(\mathbb{A}(\mathcal{S}_n),\mathcal{T}_n)\Big) - \mathcal{L}(\mathbb{A}(\mathcal{S}_n),\mathcal{S}_n)\Big\} \le 0.$$

DEFINITION 8.4. *Given $\{\mathcal{S}_n\}$, Algorithm $\mathbb{A}$ is* robust w.r.t. *$\{\mathcal{S}_n\}$ if there exists $\{D_n(\mathcal{S}_n)\}$ such that $D_n(\mathcal{S}_n) \subseteq (\mathcal{X}\times\mathcal{Y})^n$, $\mu_n(D_n(\mathcal{S}_n)) \to 1$, and*

$$\limsup_{n\to\infty}\Big\{\sup_{\hat{\mathcal{S}}_n\in D_n(\mathcal{S}_n)} \mathcal{L}(\mathbb{A}(\mathcal{S}_n),\hat{\mathcal{S}}_n) - \mathcal{L}(\mathbb{A}(\mathcal{S}_n),\mathcal{S}_n)\Big\} \le 0.$$

In both definitions we make no assumptions on the algorithm, the loss function and the probabilistic process that generates the data. Indeed, the equivalence between robustness and generalizability holds in a very general setup. As an example to the generality of the framework, we establish results for Markovian data in Section 8.4 and percentile loss-functions in Corollary 8.5.

The following theorem that establishes the equivalence between robustness and generalizability is the main result of this section.

THEOREM 8.1. *Suppose that $\mathcal{L}(\cdot,\cdot)$ is uniformly enveloped, and for any sequence of admissible prediction rules $\{\mathbb{O}_n\}$,*

$$\mathcal{L}(\mathbb{O}_n, \mathcal{T}_n) - \mathbb{E}\mathcal{L}(\mathbb{O}_n, \mathcal{T}_n) \xrightarrow{\Pr} 0. \tag{8.1}$$

Then algorithm $\mathbb{A}$ generalizes w.r.t. $\{\mathcal{S}_n\}$ if and only if it is robust w.r.t. $\{\mathcal{S}_n\}$.

The theorem is established by combining the following two propositions.

PROPOSITION 8.2. *If $\mathcal{L}(\cdot,\cdot)$ is uniformly enveloped, then algorithm $\mathbb{A}$ generalizes w.r.t. $\{\mathcal{S}_n\}$ if it is robust w.r.t. $\{\mathcal{S}_n\}$.*

PROOF. Given a $M > 0$, denote the M-truncation of $\mathcal{L}(\cdot,\cdot)$ as $\mathcal{L}_M(\cdot,\cdot)$. That is,

$$\mathcal{L}_M(\mathbb{O}, \mathcal{T}_n) \triangleq \min\left(\mathcal{L}(\mathbb{O}, \mathcal{T}_n),\ M\right).$$

When $\mathbb{A}$ is robust w.r.t. $\{\mathcal{S}_n\}$, by definition there exists $\{D_n(\mathcal{S}_n)\}$ such that $\mu_n(D_n(\mathcal{S}_n)) \to 1$ and

$$\limsup \left\{ \sup_{\hat{\mathcal{S}}_n \in D_n(\mathcal{S}_n)} \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \hat{\mathcal{S}}_n) - \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n) \right\} \leq 0.$$

Thus, for any $\delta,\ \epsilon > 0$, there exists $N(\delta,\epsilon)$ such that for all $n > N(\delta,\epsilon)$, $\mu_n(D_n(\mathcal{S}_n)) > 1-\delta$, and

$$\sup_{\hat{\mathcal{S}}_n \in D_n(\mathcal{S}_n)} \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \hat{\mathcal{S}}_n) - \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n) < \epsilon.$$

By definition of $\{\mu_n\}$, we have $\mu_n(H) = \Pr(\mathcal{T}_n \in H)$ holds for all $n \geq 1$ and every measurable set $H \subseteq (\mathcal{X}\times\mathcal{Y})^n$. Thus, for any $n > N(\delta,\epsilon)$ we have $\Pr\big(\mathcal{T}_n \notin D_n(\mathcal{S}_n)\big) \leq$

δ. Therefore, the following holds for any $n > N(\delta, \epsilon)$,

$$\begin{aligned}
&\mathbb{E}\Big(\mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{T}_n)\Big) - \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n) \\
\leq &\mathbb{E}\Big\{ f_n(\mathcal{T}_n) \cdot \mathbf{1}\big[f_n(\mathcal{T}_n) > M\big]\Big\} + \mathbb{E}\Big\{\mathcal{L}_M(\mathbb{A}(\mathcal{S}_n), \mathcal{T}_n)\Big\} - \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n) \\
\leq &\mathbb{E}\Big\{ f_n(\mathcal{T}_n) \cdot \mathbf{1}\big[f_n(\mathcal{T}_n) > M\big]\Big\} + \Pr(\mathcal{T}_n \notin D_n(\mathcal{S}_n))\mathbb{E}\Big(\mathcal{L}_M(\mathbb{A}(\mathcal{S}_n),\, \mathcal{T}_n)|\mathcal{T}_n \notin D_n(\mathcal{S}_n)\Big) \\
&\quad + \Pr(\mathcal{T}_n \in D_n(\mathcal{S}_n))\mathbb{E}\Big(\mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{T}_n)|\mathcal{T}_n \in D_n(\mathcal{S}_n)\Big) - \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n) \\
\leq &\mathbb{E}\Big\{ f_n(\mathcal{T}_n) \cdot \mathbf{1}\big[f_n(\mathcal{T}_n) > M\big]\Big\} + \delta M + \sup_{\hat{\mathcal{S}}_n \in D_n(\mathcal{S}_n)} \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \hat{\mathcal{S}}_n) - \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n) \\
\leq &\mathbb{E}\Big\{ f_n(\mathcal{T}_n) \cdot \mathbf{1}\big[f_n(\mathcal{T}_n) > M\big]\Big\} + \delta M + \epsilon.
\end{aligned}$$

Here all expectations are taken over different draws of $\mathcal{T}_n$. By setting M large enough, the first term of the right-hand-side can be made arbitrarily small for every large enough n. We thus conclude that

$$\limsup_n \Big\{\mathbb{E}_{\mathcal{T}_n}\Big(\mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{T}_n)\Big) - \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n)\Big\} \leq 0,$$

i.e., the algorithm $\mathbb{A}$ generalizes for $\{\mathcal{S}_n\}$, because ϵ, δ can be arbitrary. □

PROPOSITION 8.3. *Given $\{\mathcal{S}_n\}$, if algorithm $\mathbb{A}$ is not robust w.r.t. $\{\mathcal{S}_n\}$, then there exists ϵ^*, $\delta^* > 0$ such that the following holds for infinitely many n,*

$$\Pr\Big(\mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{T}_n) \geq \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n) + \epsilon^*\Big) \geq \delta^*. \tag{8.2}$$

PROOF. We prove the proposition by contradiction. Assume that such ϵ^* and δ^* do not exist. Let $\epsilon_t = \delta_t = 1/t$, then there exists $\{N(t)\}_{t=1}^{\infty}$ such that for all t we have $N(t-1) \leq N(t)$ and $n \geq N(t)$ implies $\Pr\Big(\mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{T}_n) \geq \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n) + \epsilon_t\Big) < \delta_t$. For each n, define the following set:

$$D_n^t(\mathcal{S}_n) \triangleq \{\hat{\mathcal{S}}_n | \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \hat{\mathcal{S}}_n) - \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n) < \epsilon_t\}.$$

Thus, for $n \geq N(t)$ we have

$$\begin{aligned}\mu_n(D_n^t(\mathcal{S}_n)) &= 1 - \Pr\Big(\mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{T}_n) \geq \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n) + \epsilon_t\Big) \\ &> 1 - \delta_t.\end{aligned}$$

For $n \geq N(1)$, define $D_n(\mathcal{S}_n)$ as $D_n(\mathcal{S}_n) \triangleq D_n^{t(n)}(\mathcal{S}_n)$, where: $t(n) \triangleq \max\big(t | N(t) \leq n;\ t \leq n\big)$. Thus we have for all $n \geq N(1)$ we have that $\mu_n(D_n(\mathcal{S}_n)) > 1 - \delta_{t(n)}$ and $\sup_{\hat{\mathcal{S}}_n \in D_n(\mathcal{S}_n)} \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \hat{\mathcal{S}}_n) - \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n) < \epsilon_{t(n)}$. Since $t(n) \uparrow \infty$ it follows that $\delta_{t(n)} \to 0$ and $\epsilon_{t(n)} \to 0$. Therefore, $\mu_n(D_n(\mathcal{S}_n)) \to 1$, and

$$\limsup_{n\to\infty} \Big\{ \sup_{\hat{\mathcal{S}}_n \in D_n(\mathcal{S}_n)} \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \hat{\mathcal{S}}_n) - \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n) \Big\} \leq 0.$$

That is, $\mathbb{A}$ is robust w.r.t. $\{\mathcal{S}_n\}$, which is a desired contradiction. □

Now we prove Theorem 8.1.

Proof of Theorem 8.1. Sufficiency (robustness leads to generalizability) was established in Proposition 8.2. Hence we prove necessity. Since algorithm $\mathbb{A}$ is not robust, Proposition 8.3 implies that (8.2) holds for infinite many n. Further note that under (8.1), when inequality (8.2) holds for infinite many n we have for infinitely many n,

$$\mathbb{E}_{\mathcal{T}_n}\big[\mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{T}_n)\big] \geq \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n) + \frac{\epsilon^*}{2},$$

which means that $\mathbb{A}$ does not generalize. Thus, the necessity is established. □

We apply Theorem 8.1 to specific loss functions under the assumptions that the testing samples are Independently and Identically Distributed (IID). The first corollary considers the nearly ubiquitous loss function in machine learning: the average loss.

Corollary 8.4. *Suppose $\mathcal{L}(\cdot,\cdot)$ is the average loss of $l(\cdot,\cdot)$ and testing samples are IID and independent of training samples. If there exists $f(\cdot) : (\mathcal{X} \times \mathcal{Y}) \to \mathbb{R}$ such that $\int f(\mathbf{t})\, \mu_1(d\mathbf{t}) < +\infty$ and $0 \leq l(\mathbb{O}, \mathbf{t}) \leq f(\mathbf{t})$ for all admissible $\mathbb{O}$ and $\mathbf{t} \in (\mathcal{X} \times \mathcal{Y})$, then algorithm $\mathbb{A}$ generalizes w.r.t. $\{\mathcal{S}_n\}$ if and only if it is robust w.r.t. $\{\mathcal{S}_n\}$.*

PROOF. We need to show that $\mathcal{L}(\cdot,\cdot)$ is uniformly enveloped and (8.1) holds for all admissible $\{\mathbb{O}_n\}$.

Step 1: We show that $\mathcal{L}(\cdot,\cdot)$ is uniformly enveloped. Define

$$f_n(\mathcal{T}_n) \triangleq \frac{1}{n}\sum_{i=1}^{n} f(\mathbf{t}_i),$$

where $\mathcal{T}_n = (\mathbf{t}_1, \cdots, \mathbf{t}_n)$. Observe that $\mathcal{L}(\mathbb{O}, \mathcal{T}_n) \leq f_n(\mathcal{T}_n)$ for all admissible $\mathbb{O}$. Fix a $M > \mathbb{E}f(\mathbf{t})$. We have

$$\begin{aligned}
&\mathbb{E}\Big\{ f_n(\mathcal{T}_n) \cdot \mathbf{1}[f_n(\mathcal{T}_n) > M] \Big\} \\
=&\mathbb{E}\Big\{ (f_n(\mathcal{T}_n) - M) \cdot \mathbf{1}[f_n(\mathcal{T}_n) > M] \Big\} + \mathbb{E}\Big\{ M \cdot \mathbf{1}[f_n(\mathcal{T}_n) > M] \Big\} \\
\leq&\mathbb{E}\Big\{ \frac{1}{n}\sum_{i=1}^{n} f(\mathbf{t}_i) - \mathbb{E}f(\mathbf{t}) \Big\} + M\mathrm{Pr}[\frac{1}{n}\sum_{i=1}^{n} f(\mathbf{t}_i) > M].
\end{aligned}$$

As $n \uparrow \infty$ the first term of the right-hand-side converges to 0 and the second term converges to $M\mathrm{Pr}(\mathbb{E}(f(\mathbf{t})) > M)$, both due to strong law of large numbers. Hence,

$$\begin{aligned}
&\lim_{M\to\infty} \left\{ \limsup_n \mathbb{E}\Big\{ f_n(\mathcal{T}_n) \cdot \mathbf{1}\big[f_n(\mathcal{T}_n) > M\big] \Big\} \right\} \\
\leq &\lim_{M\to\infty} \{M\mathrm{Pr}(\mathbb{E}(f(\mathbf{t})) > M)\} = 0.
\end{aligned}$$

Thus, $\mathcal{L}(\cdot,\cdot)$ is uniformly enveloped (observe that the left-hand-side is non-negative).

Step 2: We show that (8.1) holds for all admissible $\{\mathbb{O}_n\}$. To simplify the representation, we define the following functions for $M > 0$:

$$l_n(\mathbf{t}) \triangleq l\left(\mathbb{O}_n,\, \mathbf{t}\right); \qquad \hat{l}_n^M(\mathbf{t}) \triangleq \min\left(l_n(\mathbf{t}),\, M\right).$$

Observe that $0 \leq l_n^M(\mathbf{t}) \leq l_n(\mathbf{t}) \leq f(\mathbf{t})$. Hence $l_n^M(\cdot)$ and $l_n(\cdot)$ are integrable. Thus, with some algebra we have

$$\begin{aligned}
&\Big|\frac{1}{n}\sum_{i=1}^{n} l_n(\mathbf{t}_i) - \mathbb{E}(l_n(\mathbf{t}))\Big| \\
\leq&\Big\{\frac{1}{n}\sum_{i=1}^{n}\big\{[f(\mathbf{t}_i) - M]\cdot \mathbf{1}(f(\mathbf{t}_i) > M)\big\}\Big\} + \Big\{\mathbb{E}\big\{[f(\mathbf{t}) - M]\mathbf{1}(f(\mathbf{t}) > M)\big\}\Big\} \\
&+ \Big|\frac{1}{n}\sum_{i=1}^{n} \hat{l}_n^M(\mathbf{t}_i) - \mathbb{E}(\hat{l}_n^M(\mathbf{t}))\Big|.
\end{aligned}$$

Here, the expectation is taken over μ_1. We now bound each term separately. Given $\epsilon > 0$, there exists M_ϵ such that the second term is smaller than $\epsilon/4$ due to the integrability of $f(\cdot)$. Further, for a fixed M the first term is a summation of IID integrable random variables and hence the strong law of large number holds. Thus, given M_ϵ as defined above and $\delta > 0$, there exists n_1^* such that for any $n > n_1^*$, with probability at least $1 - \delta/2$,

$$\begin{aligned}
&\frac{1}{n}\sum_{i=1}^{n}\big\{[f(\mathbf{t}_i) - M_\epsilon]\cdot \mathbf{1}(f(\mathbf{t}_i) > M_\epsilon)\big\} \\
\leq&\mathbb{E}\big\{[f(\mathbf{t}) - M_\epsilon]\cdot \mathbf{1}(f(\mathbf{t}) > M_\epsilon)\big\} + \epsilon/4 \leq \epsilon/2.
\end{aligned}$$

Finally, given M_ϵ, the last term can be bounded using Hoeffding's inequality since $l_n^M(\cdot)$ are uniformly bounded. That is, there exists n_2^* such that for any $n > n^*/2$, with probability at least $1 - \delta/2$,

$$\Big|\frac{1}{n}\sum_{i=1}^{n} \hat{l}_n^{M_\epsilon}(\mathbf{t}_i) - \mathbb{E}(\hat{l}_n^{M_\epsilon}(\mathbf{t}))\Big| \leq \epsilon/4.$$

Combining all three terms, we conclude that for $n > \max(n_1^*, n_2^*)$, the following holds with probability at least $1 - \delta$,

$$\Big|\frac{1}{n}\sum_{i=1}^{n} l_n(\mathbf{t}_i) - \mathbb{E}(l_n(\mathbf{t}))\Big| \leq \epsilon.$$

Observe that $\mathbb{E}_{\mathcal{T}_n}\mathcal{L}(\mathbb{O}, \mathcal{T}_n) \equiv \mathbb{E}(l(\mathbb{O}, \mathbf{t}))$. Hence, (8.1) holds. The corollary follows by applying Theorem 8.1. □

The next corollary considers an interesting while less extensively investigated loss function: quantile loss.

COROLLARY 8.5. *Suppose the training samples are IID and independent to the training samples and that the loss function $\mathcal{L}(\cdot,\cdot)$ is the κ-quantile loss of $l(\cdot,\cdot)$ with $\kappa \in (0,1)$. That is*

$$\mathcal{L}(\mathbb{O}, \mathcal{T}_n) = \inf\big\{c \big| \sum_{i=1}^{n} \mathbf{1}(l(\mathbb{O}, \mathbf{t}_i) \leq c) \geq \kappa n\}.$$

Assume further that $l(\cdot,\cdot)$ is non-negative and upper bounded by a constant M and for any $\epsilon > 0$,

$$\inf_{\mathbb{O}} \Big\{\kappa - \Pr\big[l(\mathbb{O}, \mathbf{t}) \leq \nu(\mathbb{O}, \kappa) - \epsilon\big]\Big\} > 0;$$

$$\textit{and: } \inf_{\mathbb{O}} \Big\{\Pr\big[l(\mathbb{O}, \mathbf{t}) \leq \nu(\mathbb{O}, \kappa) + \epsilon\big] - \kappa\Big\} > 0;$$

$$\textit{where: } \nu(\mathbb{O}, \theta) \triangleq \sup\Big\{c\Big|\Pr\big[l(\mathbb{O}, \mathbf{t}) \leq c\big] \leq \theta\Big\}.$$

Then algorithm $\mathbb{A}$ generalizes w.r.t. $\{\mathcal{S}_n\}$ if and only if it is robust w.r.t. $\{\mathcal{S}_n\}$.

PROOF. First notice that by assumption $0 \leq l(\cdot,\cdot) \leq M$, which implies that $\mathcal{L}(\cdot,\cdot)$ is (trivially) uniformly enveloped. Now we show that (8.1) holds for all admissible $\{\mathbb{O}_n\}$. Fix $\epsilon > 0$, define $c_\kappa > 0$ as

$$c_\kappa \triangleq \min\left\{\inf_{\mathbb{O}}\Big[\kappa - \Pr\big[l(\mathbb{O}, \mathbf{t}) \leq \nu(\mathbb{O}, \kappa) - \epsilon\big]\Big], \quad \inf_{\mathbb{O}}\Big[\Pr\big[l(\mathbb{O}, \mathbf{t}) \leq \nu(\mathbb{O}, \kappa) + \epsilon\big] - \kappa\Big]\right\}.$$

Given $\{\mathbb{O}_n\}$, define $F_n(\cdot) : \mathbb{R} \to [0,\, 1]$ as

$$F_n(c) \triangleq \mathbb{E}_{\mathbf{t}}\big[\mathbf{1}(l(\mathbb{O}_n, \mathbf{t}) \leq c)\big],$$

i.e., the cumulative distribution function of $l(\mathbb{O}_n, \mathbf{t})$. Let $\nu^n \triangleq \sup\{c|F_n(c) \leq \kappa\} = \nu(\mathbb{O}_n, \kappa)$. Since $\mathcal{L}(\cdot,\cdot)$ is the quantile loss, we have

$$\Pr\big(\mathcal{L}(\mathbb{O}_n, \mathcal{T}_n) \leq \nu^n - \epsilon\big) = \Pr\big(\frac{1}{n}\sum_{i=1}^{n} \mathbf{1}(l(\mathbb{O}_n, \mathbf{t}_i) \leq \nu^n - \epsilon) \geq \kappa\big).$$

Notice that $\mathbf{1}(l(\mathbb{O}_n, \mathbf{t}_i) \leq \nu_\kappa - \epsilon)$ are IID binomial random variables for $i = 1, \cdots, n$. Therefore, by Hoeffding's inequality and the definition of $F_n(\cdot)$ we have

$$\begin{aligned}
&\Pr\big(\frac{1}{n}\sum_{i=1}^{n}\mathbf{1}(l\mathbb{O}_n, \mathbf{t}_i) \leq \nu^n - \epsilon) \geq \kappa\big) \\
=&\Pr\Big\{\frac{1}{n}\sum_{i=1}^{n}\mathbf{1}\big[l(\mathbb{O}_n, \mathbf{t}_i) \leq \nu^n - \epsilon\big] \geq F_n(\nu^n - \epsilon) + (\kappa - F_n(\nu^n - \epsilon))\Big\} \\
\leq& \exp[-2n^2(\kappa - F_n(\nu^n - \epsilon))^2] \leq \exp[-2n^2 c_\kappa^2].
\end{aligned}$$

Similarly,

$$\Pr\big(\mathcal{L}(\mathbb{O}_n, \mathcal{T}_n) \geq \nu^n + \epsilon) \leq \exp[-2n^2 c_\kappa^2].$$

Leading to

$$\mathcal{L}(\mathbb{O}_n, \mathcal{T}_n) - \nu^n \xrightarrow{\Pr} 0. \tag{8.3}$$

Since $\mathcal{L}(\cdot, \cdot)$ is uniformly bounded, (8.3) implies that

$$\mathbb{E}(\mathbb{O}_n, \mathcal{T}_n) - \nu^n \to 0.$$

Hence (8.1) holds. The corollary follows from Theorem 8.1. □

8.3. Robustness for algorithms with IID samples

Proving that a learning algorithm is robust has merit beyond the robustness property itself since it implies generalizability. In this section, we consider the case where training samples are IID and provide simple conditions for robustness. In Section 8.3.1 we investigate the case where the set of admissible prediction rules has a finite bracketing number. In Section 8.3.2 we consider the case where the output prediction rules are smooth in an appropriately defined way. One example of a setup with smooth classifiers is large margin classifiers.

8.3.1. Finite bracketing number. In this subsection we investigate the case where the set of admissible prediction rules has a finite bracketing number. Recall the following standard definition [**154**]:

DEFINITION 8.5. *Given two functions l and u, the* bracket $[l, u]$ *is set of functions f with $l \leq f \leq u$. An* ϵ-bracket *is a bracket $[l, u]$ such that $\|l-u\| \leq \epsilon$. The* bracketing number $N_{[]}\big(\epsilon, \mathcal{F}, \|\cdot\|\big)$ *is the minimum number of ϵ-brackets that cover $\mathcal{F}$.*

It is well known (e.g.,[**154, 2, 54**]) that a learning algorithm whose output belongs to a function class with finite bracketing number generalize well. Here we show that this can be attributed to the robustness of such algorithms.

PROPOSITION 8.6. *Let $\mathbf{s}_i$, $\mathbf{t}_i$ be IID draws from μ_1 and let $\mathcal{L}(\cdot,\cdot)$ be the average loss of a non-negative function $l(\cdot,\cdot)$. Suppose that there exists $\mathcal{F}$ such that for all admissible $\mathbb{O}$, $l(\mathbb{O},\cdot) \in \mathcal{F}$ and the bracketing number $N_{[]}\big(\epsilon, \mathcal{F}, L_1(\mu_1)\big)$ is finite for all ϵ. Then $\mathbb{A}$ is robust w.r.t. almost every $\{\mathcal{S}_n\}$.*

PROOF. First observe that a finite bracketing number implies that $\mathcal{L}(\cdot,\cdot)$ is uniformly enveloped. Fix $\epsilon > 0$, choose finitely many (say I) $\epsilon/4$-brackets $[f_i^-, f_i^+]$ that cover $\mathcal{F}$. Thus, given S_n, there is a i^* such that

$$f_{i^*}^-(\mathbf{t}) \leq l(\mathbb{A}(\mathcal{S}_n), \mathbf{t}) \leq f_{i^*}^+(\mathbf{t}); \quad \forall \mathbf{t} \in \mathcal{X} \times \mathcal{Y}. \tag{8.4}$$

For $i = 1, \cdots, I$, define

$$D_n^i(\mathcal{S}_n) \triangleq \Big\{(\hat{\mathbf{s}}_1, \cdots, \hat{\mathbf{s}}_n) \Big| \frac{1}{n}\sum_{j=1}^n [f_i^+(\hat{\mathbf{s}}_j) - f_i^-(\hat{\mathbf{s}}_j)] \leq \epsilon/2;\ \frac{1}{n}\sum_{j=1}^n [f_i^-(\hat{\mathbf{s}}_j) - f_i^-(\mathbf{s}_j)] \leq \epsilon/2\Big\}.$$

Let $D_n(\mathcal{S}_n) \triangleq \bigcap_{i=1}^I D_n^i(\mathcal{S}_n)$. We have that

$$\begin{aligned}
&\sup_{\hat{\mathcal{S}}_n \in D_n(\mathcal{S}_n)} l(\mathbb{A}(\mathcal{S}_n), \hat{\mathcal{S}}_n) - l(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n) \\
\overset{(a)}{\leq} &\sup_{\hat{\mathcal{S}}_n \in D_n^{i^*}(\mathcal{S}_n)} \Big\{\frac{1}{n}\sum_{j=1}^n f_{i^*}^+(\hat{\mathbf{s}}_j) - \frac{1}{n}\sum_{j=1}^n f_{i^*}^-(\hat{\mathbf{s}}_j) + \frac{1}{n}\sum_{j=1}^n f_{i^*}^-(\hat{\mathbf{s}}_j) - \frac{1}{n}\sum_{j=1}^n f_{i^*}^-(\mathbf{s}_j)\Big\} \\
\leq &\epsilon.
\end{aligned}$$

Here, (a) follows from (8.4) and from the fact that $D_n(\mathcal{S}_n) \subseteq D_n^{i^*}(\mathcal{S}_n)$.

Next we show that for almost all $\{\mathcal{S}_n\}$, $\Pr_{\mathcal{T}_n}\big(\mathcal{T}_n \in D_n(\mathcal{S}_n)\big) \to 1$. It suffices to show that with probability 1 on $\{\mathcal{S}_n, \mathcal{T}_n\}_{n=1}^{\infty}$ the following event

$$\big\{\mathcal{T}_j \notin D_j(\mathcal{S}_j)\big\}; \tag{8.5}$$

happens for finite many j. For each $i \in \{1, \cdots, I\}$, by strong law of large numbers (notice that all functions involved are non-negative and integrable) with probability 1 both of the following events happen for finitely many n:

$$\Big\{\frac{1}{n}\sum_{j=1}^{n}[f_i^+(\mathbf{t}_j) - f_i^-(\mathbf{t}_j)] - \mathbb{E}_{\mathbf{t}\sim\mu_1}\big(f_i^+(\mathbf{t}) - f_i^-(\mathbf{t})\big) \geq \epsilon/4\Big\};$$

$$\Big\{\frac{1}{n}\sum_{j=1}^{n} f_i^-(\mathbf{t}_j) - \frac{1}{n}\sum_{j=1}^{n} f_i^-(\mathbf{s}_j) \geq \epsilon/2\Big\}.$$

Notice $\mathbb{E}_{\mathbf{t}\sim\mu_1}\big(f_i^+(\mathbf{t}) - f_i^-(\mathbf{t})\big) \leq \epsilon/4$ since $[f_i^-, f_i^+]$ is a $\epsilon/4$-bracket. Thus, $\big\{\mathcal{T}_j \notin D_j^i(\mathcal{S}_j)\big\}$ holds for finite many j. Since I is finite, (8.5) holds for finite many j. Therefore, $\mathbb{A}$ is robust w.r.t almost every $\{\mathcal{S}_n\}$. □

8.3.2. Smooth solutions. In this subsection, we consider a less extensively investigated case: the solutions to a learning algorithm are "smooth," and show that such property implies robustness and hence generalizability.

Equip the space $(\mathcal{X} \times \mathcal{Y})$ with a metric ρ. For $\gamma > 0$, let $K(\gamma)$ be the minimal number of subsets which partitions $(\mathcal{X} \times \mathcal{Y})$ such that any two points belonging to one subset has a distance at most γ. Recall the definition of covering number from [**154**]:

DEFINITION 8.6. *The* covering number $N(\epsilon, \mathcal{Z}, \rho)$ *is the minimal number of balls* $\{g : \rho(g, f) \leq \epsilon\}$ *of radius* ϵ *that covers* $\mathcal{Z}$*.*

It is easy to see that $K(\gamma) \leq N(\gamma/2, (\mathcal{X} \times \mathcal{Y}), \rho)$. We now define the pair function between two sets of identical size which denote the maximal fraction of the points that can be matched (up to γ).

DEFINITION 8.7. *The* pairing fraction *between* $\mathcal{S}_n$ *and* $\mathcal{T}_n$ *at* γ *is the function* $\text{pair}(\cdot) : (\mathcal{X} \times \mathcal{Y})^n \times (\mathcal{X} \times \mathcal{Y})^n \times \mathbb{R}^+ \to [0, 1]$*:*

$$\text{pair}(\mathcal{S}_n, \mathcal{T}_n, \gamma) \triangleq \max_{\pi \in \Pi_n} \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}(\rho(\mathbf{s}_i, \pi(\mathcal{T}_n)_i) \leq \gamma),$$

where Π_n *is the set of permutations of rank* n*.*

The next lemma shows that the pairing fraction can be bounded using $K(\gamma)$.

LEMMA 8.7. *Given* $\gamma > 0$*. If* $\mathbf{s}_1^*, \cdots, \mathbf{s}_n^*$, $\mathbf{t}_1^*, \cdots, \mathbf{t}_n^*$ *are independent draws from* μ_1*, then the event* $\{\mathcal{T}_n^* \in \tilde{D}_n^{\gamma,\delta}(\mathcal{S}_n^*)\}$ *holds with (joint) probability at least* $1 - \delta$*. Here*

$$\tilde{D}_n^{\gamma,\delta}(\mathcal{S}_n^*) \triangleq \left\{ (\mathbf{t}_1, \cdots, \mathbf{t}_n) \,|\text{pair}(\mathcal{S}_n^*, \mathcal{T}_n, \gamma) \geq 1 - \sqrt{\frac{8}{n}[(K(\gamma) + 1) \ln 2 + \ln \frac{1}{\delta}]} \right\}.$$

PROOF. We partition $(\mathcal{X} \times \mathcal{Y})$ into $K(\gamma)$ subsets $H_1, \cdots, H_{K(\gamma)}$ such that the distance between any two points belonging to one subset is at most γ. For $j = 1, \cdots, K(\gamma)$, let $N_j^{\mathcal{S}}$ and $N_j^{\mathcal{T}}$ denote the number of points of $\mathcal{S}_n^*$ and $\mathcal{T}_n^*$ that fall into the j^{th} partition. Observe that

$$\frac{1}{n} \sum_{j=1}^{K(\gamma)} |N_j^{\mathcal{S}} - N_j^{\mathcal{T}}| \leq 1 - \text{pair}(\mathcal{S}_n^*, \mathcal{T}_n^*, \gamma).$$

Notice that $(N_1^{\mathcal{S}}, \cdots, N_{K(\gamma)}^{\mathcal{S}})$ and $(N_1^{\mathcal{T}}, \cdots, N_{K(\gamma)}^{\mathcal{T}})$ are IID multinomial random variables with parameters n and
$(\mu_1(H_1), \cdots, \mu_1(H_{K(\gamma)}))$. The following holds by the Bretegnolle-Huber-Carol inequality

$$\Pr\left\{ \frac{1}{n} \sum_{j=1}^{K(\gamma)} |N_j^{\mathcal{S}} - N_j^{\mathcal{T}}| \geq 2\lambda \right\} \leq 2^{K(\gamma)+1} \exp(\frac{-n\lambda^2}{2}).$$

Thus

$$\Pr\{1 - \text{pair}(\mathcal{S}_n^*, \mathcal{T}_n^*, \gamma) \geq 2\lambda\} \leq 2^{K(\gamma)+1} \exp(\frac{-n\lambda^2}{2}).$$

Taking δ equals the right hand side establishes the lemma. □

Lemma 8.7 holds for a fixed γ that is independent of the samples. However, in practice we are often interested in data-dependent γ. Thus we make the statement uniform over all value of γ at the expense of an additional $\ln(1/\delta)$ factor.

LEMMA 8.8. *If $\mathbf{s}_1^*, \cdots, \mathbf{s}_n^*$, $\mathbf{t}_1^*, \cdots, \mathbf{t}_n^*$ are independent draws from μ_1, then the event $\{\mathcal{T}_n^* \in \bigcap_{\gamma\in(0,1]} D_n^{\gamma,\delta}(\mathcal{S}_n^*)\}$ holds with (joint) probability at least $1-\delta$. Here*

$$D_n^{\gamma,\delta}(\mathcal{S}_n^*) \triangleq \left\{(\mathbf{t}_1, \cdots, \mathbf{t}_n) \,|\mathrm{pair}(\mathcal{S}_n^*, \mathcal{T}_n, \gamma) \geq 1 - \sqrt{\frac{8}{n}[(K(\frac{\gamma}{2})+1)\ln 2 + \ln\frac{2}{\delta\gamma}]}\right\}.$$

PROOF. We recall the following Lemma which is adapted from Lemma 15.5 of [**2**]:

LEMMA 8.9. *Let $(X, \mathcal{F}, P)$ be a probability space, and let*

$$\{E(\alpha_1, \alpha_2, \delta) : 0 < \alpha_1, \alpha_2, \delta \leq 1\}$$

be a set of events satisfying the following conditions:

(1) for all $0 \leq \alpha \leq 1$ and $0 \leq \delta \leq 1$, $P(E(\alpha, \alpha, \delta)) \leq \delta$;

(2) for all $0 \leq a \leq 1$ and $0 \leq \delta \leq 1$

$$\bigcup_{\alpha\in(0,\,1]} E\big(\alpha a, \alpha, \delta\alpha(1-a)\big)$$

is measurable.

(3) for all $0 < \alpha_1 \leq \alpha \leq \alpha_2 \leq 1$ and $0 \leq \delta_1 \leq \delta < 1$

$$E(\alpha_1, \alpha_2, \delta_1) \subseteq E(\alpha, \alpha, \delta).$$

Then, for $0 < a, \delta < 1$

$$P\left(\bigcup_{\alpha\in(0,\,1]} E\big(\alpha a, \alpha, \delta\alpha(1-a)\big)\right) \leq \delta.$$

Let $E(\gamma_1, \gamma_2, \delta)$ be the set of $\mathcal{S}_n, \mathcal{T}_n$ such that

$$\mathrm{pair}(\mathcal{S}_n, \mathcal{T}_n, \gamma_2) \leq 1 - \sqrt{\frac{8}{n}[(K(\gamma_1)+1)\ln 2 + \ln\frac{1}{\delta}]}.$$

Lemma 8.8 follows by taking $a = 1/2$ and apply Lemma 8.9. □

Next we prove the main theorem of this section, which states that if the solution for a learning algorithm is sufficiently smooth, then it is robust and hence generalizes well.

THEOREM 8.10. *Suppose that* $\mathbf{s}_i$, $\mathbf{t}_i$ *are IID draws from* μ_1 *and that* $\mathcal{L}(\cdot,\cdot)$ *is the average loss of* $l(\cdot,\cdot)$ *and is uniformly enveloped. Given* $\{\mathcal{S}_n\}$, *if there exist* $\{g_n(\cdot) : \mathbb{R}_+ \to \mathbb{R}_+\}$ *and* $\{c_n > 0\}$ *such that*

(1) For any n, $g_n(\cdot)$ *is non-decreasing, and* $g_n^{-1}(\epsilon)$ *defined as* $g_n^{-1}(\epsilon) = \sup\{c | g_n(c) \leq \epsilon\}$ *exists for every* $\epsilon > 0$; *(this include the case that* $g_n(c) \leq \epsilon$ *for all* c, *where we denote* $g_n^{-1}(\epsilon) = +\infty$*).*

(2) $\{\lambda_n\} \downarrow 0$ *where*

$$\lambda_n \triangleq \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}\Big(\sup_{\mathbf{y}:\rho(\mathbf{y},\mathbf{s}_i)\leq c_n} |l(\mathbb{A}(\mathcal{S}_n),\mathbf{s}_i) - l(\mathbb{A}(\mathcal{S}_n),\mathbf{y})| > g_n(\rho(\mathbf{s}_i,\mathbf{y}))\Big). \tag{8.6}$$

(3) For all $\epsilon > 0$

$$\begin{aligned} &\lim_{n\to\infty} \frac{K(\frac{\min(g^{-1}(\epsilon),c_n)}{2})}{n} = 0; \\ &\limsup_{n\to\infty} \frac{-\ln \min(g^{-1}(\epsilon), c_n)}{n} \leq 0. \end{aligned} \tag{8.7}$$

Then $\mathbb{A}$ *is robust w.r.t. almost every* $\{\mathcal{S}_n\}$.

The conditions of Theorem 8.10 mean that the output solution $\mathbb{A}(\mathcal{S}_n)$ is smooth (i.e., upper bounded by $g_n(\cdot)$) within the neighborhood (i.e., in a ball of radius c_n) of the majority $(1 - \lambda_n)$ of training samples. Algorithms that ensure smoothness include regression with a bounded norm of the output, where $g_n(\cdot)$ is a linear function depending on the norm [**167**] and large-margin classifiers, where the upper-bound function $g_n(\cdot) \equiv 0$. Theorem 8.10 provides a new approach for investigating norm-constrained or margin-based algorithms such as support vector machines [**168**].

PROOF OF THEOREM 8.10. Assume for now that the loss function is uniformly bounded by a constant M. Condition (8.7) implies that there exists $\{\epsilon(n)\} \downarrow 0$ such

that

$$\lim_{n\to\infty} \frac{K(\frac{\min(g_n^{-1}(\epsilon(n)),c_n)}{2})}{n} = 0;$$
$$\limsup_{n\to\infty} \frac{-\ln \min(g_n^{-1}(\epsilon(n)), c_n)}{n} \leq 0.$$

Let $\gamma_n = \min(g_n^{-1}(\epsilon(n)), c_n)$, $\tilde{\gamma}_n = \min(\gamma_n, 1)$. Given a permutation $\pi \in \Pi_n$, suppose that i satisfies

$$\sup_{\mathbf{y}:\rho(\mathbf{y},\mathbf{s}_i)\leq c_n} |l(\mathbb{A}(\mathcal{S}_n), \mathbf{s}_i) - l(\mathbb{A}(\mathcal{S}_n), \mathbf{y})| \leq g_n(\rho(\mathbf{s}_i, \mathbf{y}));$$

and $\rho(\mathbf{s}_i, \pi(\mathcal{T}_n)_i) \leq \gamma_n$. Then the following holds

$$|l(\mathbb{A}(\mathcal{S}_n), \mathbf{s}_i) - l(\mathbb{A}(\mathcal{S}_n), \pi(\mathcal{T}_n)_i)| \leq g_n(\rho(\mathbf{s}_i, \pi(\mathcal{T}_n)_i) \leq g_n(g_n^{-1}(\epsilon(n))) = \epsilon(n).$$

We therefore have

$$\begin{aligned}
&|\mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{T}_n) - \mathcal{L}(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n)| \\
\leq& M\big[\lambda(n) + (1 - \text{pair}(\mathcal{S}_n, \mathcal{T}_n, \gamma_n))\big] + \epsilon(n)\text{pair}(\mathcal{S}_n, \mathcal{T}_n, \gamma_n),
\end{aligned}$$

since the number of unpaired samples is upper bounded by $\lambda(n)+(1-\text{pair}(\mathcal{S}_n, \mathcal{T}_n, \gamma_n))$. Let

$$D_n(\mathcal{S}_n) = \left\{ (\mathbf{t}_1, \cdots, \mathbf{t}_n) \,|\text{pair}(\mathcal{S}_n, \mathcal{T}_n, \gamma_n) \geq 1 - \sqrt{\frac{8}{n}[(K(\frac{\tilde{\gamma}_n}{2}) + 1)\ln 2 + \ln \frac{2n^2}{\tilde{\gamma}_n}]} \right\}.$$

We have

$$\begin{aligned}
&\limsup_{n\to\infty} \Big\{ \sup_{\hat{\mathcal{S}}_n \in D_n(\mathcal{S}_n)} l(\mathbb{A}(\mathcal{S}_n), \hat{\mathcal{S}}_n) - l(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n) \Big\} \\
\leq& \limsup_{n\to\infty} \left\{ M \left[\lambda(n) + \sqrt{\frac{8}{n}[(K(\frac{\tilde{\gamma_n}}{2}) + 1)\ln 2 + \ln \frac{2n^2}{\tilde{\gamma}_n}]} \right] + \epsilon(n) \right\} = 0.
\end{aligned}$$

Furthermore,

$$\begin{aligned}&\Pr_{\mathcal{S}_n,\mathcal{T}_n}\left(\mathcal{T}_n \notin D_n(\mathcal{S}_n)\right)\\ \leq&\Pr_{\mathcal{S}_n,\mathcal{T}_n}\Big(\mathcal{T}_n^* \notin \bigcap_{\gamma\in(0,1]} D_n^{\gamma,1/n^2}(\mathcal{S}_n^*)\Big) \leq 1/n^2.\end{aligned}$$

Thus, by Borel-Cantelli lemma, w.p.1 on $\{\mathcal{S}_n,\mathcal{T}_n\}_{n=1}^{\infty}$ this happens finitely many times. This further implies that except for a set of measure zero on $\{\mathcal{S}_i\}_{i=1}^n$, $\mu_n(D_n(\mathcal{S}_n)) = \Pr_{\mathcal{T}_n}(\mathcal{T}_n \in D_n(\mathcal{S}_n)) \to 1$. Hence, under the uniform boundedness assumption, $\mathbb{A}$ is robust w.r.t. almost every $\{\mathcal{S}_n\}$.

We now relax the assumption of uniform boundedness. Let $f(\cdot)$ be the integrable envelope function of $l(\cdot,\cdot)$. Fix κ and $M > 0$. Let $l^M(\mathbb{O},\mathbf{t}) \triangleq \min(l(\mathbb{O},\mathbf{t}), M)$ and $\mathcal{L}^M(\mathbb{O},\mathcal{T}_n) \triangleq \frac{1}{n}\sum_{i=1}^n l^M(\mathbb{O},\mathbf{t}_i)$. Observe that $|l^M(\mathbb{O},\mathbf{x}) - l^M(\mathbb{O},\mathbf{y})| \leq |l(\mathbb{O},\mathbf{x}) - l(\mathbb{O},\mathbf{y})|$, hence (8.6) holds for $l^M(\cdot,\cdot)$. We therefore have the following:

$$\begin{aligned}&|\mathcal{L}(\mathbb{A}(\mathcal{S}_n),\mathcal{T}_n) - \mathcal{L}(\mathbb{A}(\mathcal{S}_n),\mathcal{S}_n)|\\ \leq&M\big[\lambda(n) + (1-\text{pair}(\mathcal{S}_n,\mathcal{T}_n,\gamma_n))\big] + \epsilon(n)\text{pair}(\mathcal{S}_n,\mathcal{T}_n,\gamma_n)\\ &+\frac{1}{n}\sum_{i=1}^n (f(\mathbf{s}_i)-M)\mathbf{1}(f(\mathbf{s}_i)>M) + \frac{1}{n}\sum_{i=1}^n (f(\mathbf{t}_i)-M)\mathbf{1}(f(\mathbf{t}_i)>M).\end{aligned} \tag{8.8}$$

Let $D_n^M(\mathcal{S}_n) \triangleq D_n(\mathcal{S}_n)\bigcap H_n^M$ where

$$H_n^M \triangleq \left\{(\mathbf{t}_1,\cdots,\mathbf{t}_n)|\frac{1}{n}\sum_{i=1}^n (f(\mathbf{t}_i)-M)\mathbf{1}(f(\mathbf{t}_i)>M) \leq \mathbb{E}[(f(\mathbf{t})-M)\mathbf{1}(f(\mathbf{t})>M)] + \kappa\right\}.$$

Notice that $\mu(H_n^M) \xrightarrow{n} 1$ due to law of large numbers. Thus, $\mu(D_n^M(\mathcal{S}_n)) \xrightarrow{n} 1$.

Furthermore, denote the following random variable which is determined by $\{\mathcal{S}_n\}$,

$$Z(M) \triangleq \limsup_{n\to\infty}\left\{\frac{1}{n}\sum_{i=1}^n (f(\mathbf{s}_i)-M)\mathbf{1}(f(\mathbf{s}_i)>M)\right\}.$$

Observe that $Z(M)$ is decreasing w.r.t. M. Furthermore, for a fixed M, by the strong law of large number, with probability 1 we have $Z(M) = \mathbb{E}\{(f(\mathbf{t})-M)\mathbf{1}(f(\mathbf{t})>M)\}$. Hence take a sequence of countably many M's that go to infinity and we easily

establish that except for a set of $\{\mathcal{S}_n\}$ with measure 0,

$$\lim_{M\to\infty} Z(M) = \lim_{M\to\infty} \mathbb{E}\{(f(\mathbf{t}) - M)\mathbf{1}(f(\mathbf{t}) > M)\} = 0.$$

Thus the following holds

$$\begin{aligned}
&\limsup_{n\to\infty}\Big\{\sup_{\hat{\mathcal{S}}_n\in D_n^M(\mathcal{S}_n)} l(\mathbb{A}(\mathcal{S}_n),\hat{\mathcal{S}}_n) - l(\mathbb{A}(\mathcal{S}_n),\mathcal{S}_n)\Big\}\\
\overset{(a)}{\leq}&\limsup_{n\to\infty}\Big\{M\left[\lambda(n) + \sqrt{\frac{2}{n}[(K(\frac{\tilde{\gamma}_n}{2})+1)\ln 2 + \ln\frac{2n^2}{\tilde{\gamma}_n}]}\right]\\
&+\epsilon(n) + \frac{1}{n}\sum_{i=1}^{n}(f(\mathbf{s}_i) - M)\mathbf{1}(f(\mathbf{s}_i) > M) + \mathbb{E}[(f(\mathbf{t}) - M)\mathbf{1}(f(\mathbf{t}) > M)] + \kappa\Big\}\\
\leq& Z(M) + \mathbb{E}[(f(\mathbf{t}) - M)\mathbf{1}(f(\mathbf{t}) > M)] + \kappa,
\end{aligned}$$

where (a) follows from (8.8). Since $\lim_{M\to\infty}(Z(M)) = 0$, we can make the right hand side arbitrarily small by taking M sufficiently large. Hence we establish that $\mathbb{A}$ is robust w.r.t. almost every $\{\mathcal{S}_n\}$. □

The rest of this section considers large-margin classifiers. Theorem 8.10 requires a metric on the labeled sampling space $\mathcal{X}\times\{-1, +1\}$ while the notion of "margin" is generally defined through a metric on $\mathcal{X}$. We thus introduce the following metric extension.

Given $\mathcal{X}$ with metric $\hat{\rho}$, we equip space $\mathcal{X}\times\{-1, +1\}$ with metric ρ defined as

$$\rho(\mathbf{u},\mathbf{v}) = \begin{cases} \hat{\rho}(\mathbf{u}_{|\mathcal{X}}, \mathbf{v}_{|\mathcal{X}}) & \text{if } \mathbf{u}_{|\{-1,+1\}} = \mathbf{v}_{|\{-1,+1\}} \\ +\infty & \text{otherwise.} \end{cases}$$

Here, the subscripts $|\mathcal{X}$ and $|\{-1,+1\}$ stand for the projections onto $\mathcal{X}$ and $\{-1, +1\}$, respectively.

Similarly to Theorem 8.10, we only require that "most" training samples have a large margin. We thus define following λ-margin, where λ is the fraction of samples that are too close to the boundary (i.e., within the margin).

DEFINITION 8.8. *Given* $\lambda \in [0, 1]$, $\mathcal{S}_n \in (\mathcal{X} \times \{-1, +1\})^n$, *and* $\mathbb{O} : \mathcal{X} \to \{-1, +1\}$, *the* λ*-margin of* $\mathbb{O}$ *w.r.t.* $\mathcal{S}_n$ *is*

$$M_\lambda(\mathbb{O}, \mathcal{S}_n) \triangleq \sup\Big\{c \Big| \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}\big[dist((\mathbf{s}_i)_{|\mathcal{X}}, \mathbb{O}) \le c\big] \le \lambda\Big\};$$

$$\textit{where}: \quad dist(\mathbf{x}, \mathbb{O}) \triangleq \inf\{\hat{\rho}(\mathbf{x}', \mathbf{x}) : \mathbf{x}' \in \mathcal{X}, \mathbb{O}(\mathbf{x}) \neq \mathbb{O}(\mathbf{x}')\}.$$

COROLLARY 8.11. *Suppose that* $\mathbf{s}_i$, $\mathbf{t}_i$ *are IID draws from* μ_1 *and let the loss function be the average classification error:*

$$l(\mathbb{O}, \mathbf{t}) = \mathbf{1}\big[\mathbb{O}(\mathbf{t}_{|\mathcal{X}}) \neq \mathbf{t}_{|\{-1,+1\}}\big].$$

Suppose further that $\{\mathcal{S}_n\}$ *satisfies that for all* $\lambda > 0$

$$\begin{aligned} &\lim_{n\to\infty} \frac{K(\frac{M_\lambda(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n)}{2})}{n} = 0 \text{ and} \\ &\limsup_{n\to\infty} \frac{-\ln M_\lambda(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n)}{n} \le 0. \end{aligned} \tag{8.9}$$

Then $\mathbb{A}$ *is robust w.r.t. almost every* $\{\mathcal{S}_n\}$.

PROOF. Equation (8.9) is equivalent to: $\exists\{\lambda(n)\} \downarrow 0$, such that

$$\begin{aligned} &\lim_{n\to\infty} \frac{K(\frac{M_{\lambda(n)}(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n)}{2})}{n} = 0; \\ &\limsup_{n\to\infty} \frac{-\ln M_{\lambda(n)}(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n)}{n} \le 0. \end{aligned}$$

The loss function is upper bounded by 1, hence is uniformly enveloped. Let $c_n = M_{\lambda(n)}(\mathbb{A}(\mathcal{S}_n), \mathcal{S}_n)$; $g_n(\cdot) \equiv 0$; and $\lambda_n = \lambda(n)$. All the conditions of Theorem 8.10 are satisfied, and hence we establish the corollary. □

8.4. Robustness of algorithms with Markovian samples

The robustness approach is not restricted to the IID setup. In many applications of interest, such as reinforcement learning and time series forecasting, the IID assumption is violated. In such applications there is a time driven process that generates

samples that depend on the previous samples (e.g., the observations of trajectory of a robot). Such a situation can be modeled by stochastic process such as a Markov processes. In this section we establish similar result to the IID case for samples the samples are drawn from a Markov chain. The state space can be general, i.e., it is not necessarily finite or countable. Thus, a certain ergodic structure of the underlying Markov chain is needed. We focus on chains that converge to equilibrium exponentially fast and uniformly in the initial condition. It is known that this is equivalent to the class of of Doeblin chains [**111**]. Recall the following definition (cf. [**111**][**59**]):

DEFINITION 8.9. *A Markov chain* $\{\mathbf{z}_i\}_{i=1}^{\infty}$ *on a state space* $\mathcal{Z}$ *is a* Doeblin chain *(with* α *and* m*) if there exists a probability measure* φ *on* $\mathcal{Z}$*,* $\alpha > 0$*, an integer* $m \geq 1$ *such that*

$$\Pr(\mathbf{z}_m \in H | \mathbf{z}_0 = z) \geq \alpha\varphi(H);\ \forall \textit{ measurable } H \subseteq \mathcal{Z};\ \forall z \in \mathcal{Z}.$$

The class of Doeblin chains is probably the "nicest" class of general state-space Markov chains. We notice that such assumption is not overly restrictive, since by requiring that an ergodic theorem holds for all bounded functions uniformly in the initial distribution itself implies that a chain is Doeblin [**111**]. In particular, an ergodic chain defined on a finite state-space is a Doeblin chain.

We first establish the equivalence between generalizability and robustness for Markovian samples, i.e., a counterpart of Corollary 8.4.

THEOREM 8.12. *Let* $\mathcal{L}(\cdot,\cdot)$ *be the average loss of a uniform bounded function* $l(\cdot,\cdot)$*. Suppose that the testing samples are drawn from a Doeblin chain and are independent of the training samples. Then algorithm* $\mathbb{A}$ *generalizes w.r.t.* $\{\mathcal{S}_n\}$ *if and only if it is robust w.r.t.* $\{\mathcal{S}_n\}$*.*

PROOF. The loss $\mathcal{L}(\cdot,\cdot)$ is uniformly enveloped trivially. To apply Theorem 8.1, we need to show that that (8.1) holds for all admissible $\{\mathcal{O}_n\}$ where $\{\mathcal{T}_n\}$ is drawn from a Doeblin chain. Recall the following lemma adapted from Theorem 2 of [**80**]:

LEMMA 8.13. *Let $\{\mathbf{x}_i\}$ be a Doeblin chain as in Definition 8.9. Fix a function $f : \mathcal{X} \to \mathbb{R}$ such that $\|f\|_\infty \leq C$. Then for $n > 2Cm/\epsilon\alpha$ the following holds*

$$\Pr\left(\sum_{i=1}^{n} f(\mathbf{x}_i) - \mathbb{E}\big[\sum_{i=1}^{n} f(\mathbf{x}_i)\big] \geq n\epsilon\right) \leq \exp\left(-\frac{\alpha^2(n\epsilon - 2Cm/\alpha)^2}{2nC^2m^2}\right).$$

Since the training samples are independent to the testing samples, for a fixed n, we can treat $\mathbb{A}(\mathcal{S}_n)$ as a fixed function and apply Lemma 8.13. Thus, the convergence in probability as stated in (8.1) holds. Applying Theorem 8.1 we complete the proof. □

We now show that similarly to Theorem 8.10, algorithms that generate smooth solutions are robust, where $\{\mathcal{S}_n\}$ and $\{\mathcal{T}_n\}$ are independent evolving according to the same Doeblin chain. To this end, we establish following lemmas first.

LEMMA 8.14. *Fix $\gamma > 0$. If $\{\mathcal{S}_n\}$ and $\{\mathcal{T}_n\}$ are independent sequences of a Doeblin chain (with α and m) on $\mathcal{X} \times \mathcal{Y}$, then for $n > 2m/\alpha^2$ the event $\{\mathcal{T}_n^* \in \tilde{D}_n^{\gamma,\delta}(\mathcal{S}_n^*)\}$ has a (joint) probability at least $1 - \delta$. Here*

$$\tilde{D}_n^{\gamma,\delta}(\mathcal{S}_n^*)$$

$$\triangleq \left\{(\mathbf{t}_1, \cdots, \mathbf{t}_n) \,|\text{pair}(\mathcal{S}_n^*, \mathcal{T}_n, \gamma) \geq 1 - 2\sqrt{\sqrt{\frac{2m^2(2(K(\gamma)+1)\ln 2 + \ln(\frac{1}{\delta}))}{\alpha^2 n}} + \frac{2m}{\alpha n}}\right\}.$$

PROOF. Similarly to the proof of Lemma 8.7, we bound the term $\frac{1}{n}\sum_{j=1}^{K(\gamma)} |N_j^{\mathcal{S}} - N_j^{\mathcal{T}}|$. Let π be the invariant measure of the Doeblin chain that generates $\mathcal{S}_n$ and $\mathcal{T}_n$. The invariant measure uniquely exists for all Doeblin chain. We have

$$\Pr(\frac{1}{n}\sum_{j=1}^{K(\gamma)} |N_j^{\mathcal{S}} - N_j^{\mathcal{T}}| \geq 2\lambda) \leq 2\Pr(\frac{1}{n}\sum_{j=1}^{K(\gamma)} |N_j^{\mathcal{S}} - \pi(H_j)| \geq \lambda).$$

Consider the set of functions $\mathcal{H} = \{\mathbf{1}(\mathbf{x} \in C) | C = \bigcup_{i\in I} H_i;\ \forall I \subseteq \{1, \cdots, K(\gamma)\}\}$, i.e., the set of indicator functions of all combinations of H_i. Then $|\mathcal{H}| = 2^{K(\gamma)}$.

Furthermore, fix a $h_0 \in \mathcal{H}$,

$$\begin{aligned}
&\Pr(\frac{1}{n}\sum_{j=1}^{K(\gamma)} |N_j^{\mathcal{S}} - \pi(H_j)| \geq \lambda) \\
=&\Pr\Big\{ \sup_{h\in\mathcal{H}} [\frac{1}{n}\sum_{i=1}^{n} h(\mathbf{s}_i) - \mathbb{E}_\pi h(\mathbf{s})] \geq \lambda \Big\} \\
\leq& 2^{K(\gamma)} \Pr[\frac{1}{n}\sum_{i=1}^{n} h_0(\mathbf{s}_i) - \mathbb{E}_\pi h_0(\mathbf{s}) \geq \lambda].
\end{aligned}$$

Since $\|h_0\|_\infty = 1$, we apply Lemma 8.13 to get for $n > 2m/\lambda\alpha$

$$\Pr[\frac{1}{n}\sum_{i=1}^{n} h_0(\mathbf{s}_i) - \mathbb{E}_\mu h_0(\mathbf{s}) \geq \lambda] \leq \exp\left(-\frac{\alpha^2(n\lambda^2 - 2m/\alpha)^2}{2nm^2}\right).$$

Leting $\delta = 2^{K(\gamma)+1}\exp[\alpha^2(n\lambda^2 - 2m/\alpha)^2/2nm^2]$ establishes the lemma. (Since $\lambda > \sqrt{2m/\alpha n}$, we have that $n > 2m/\sqrt{2m/\alpha n}\alpha$, which is equivalent to $n > 2m/\alpha^2$, implying that $n > 2m/\lambda\alpha$,as required for applying Lemma 8.13.) □

Similarly to the IID case we establish following results. The proofs are similar to those of Lemma 8.8 and Theorem 8.10 and are hence omitted.

LEMMA 8.15. *Suppose that $\{\mathcal{S}_n\}$ and $\{\mathcal{T}_n\}$ are independent sequences of a Doeblin chain (with α and m) on a state space $(\mathcal{X}\times\mathcal{Y})$. Then for $n > 2m/\alpha^2$ the event $\{\mathcal{T}_n^* \in \bigcap_{\gamma\in(0,1]} D_n^{\gamma,\delta}(\mathcal{S}_n^*)\}$ holds with (joint) probability at least $1-\delta$. Here*

$$\begin{aligned}
&\tilde{D}_n^{\gamma,\delta}(\mathcal{S}_n^*) \\
&\triangleq \left\{ (\mathbf{t}_1, \cdots, \mathbf{t}_n) \,|\mathrm{pair}(\mathcal{S}_n^*, \mathcal{T}_n, \gamma) \geq 1 - 2\sqrt{\sqrt{\frac{2m^2(2(K(\frac{\gamma}{2})+1)\ln 2 + \ln(\frac{1}{\gamma\delta}))}{\alpha^2 n}} + \frac{2m}{\alpha n}} \right\}.
\end{aligned}$$

THEOREM 8.16. *Suppose that $\{\mathcal{S}_n\}$ and $\{\mathcal{T}_n\}$ are two independent sequences sampled from a Doeblin chain on $\mathcal{X}\times\mathcal{Y}$ and that $\mathcal{L}(\cdot,\cdot)$ is the average loss of $l(\cdot,\cdot)$ and is uniformly enveloped. Given $\{\mathcal{S}_n\}$, if there exist $\{g_n(\cdot) : \mathbb{R}_+ \to \mathbb{R}_+\}$ and $\{c_n > 0\}$ such that*

(1) For any n, $g_n(\cdot)$ is non-decreasing, and $g_n^{-1}(\epsilon)$ defined as $g_n^{-1}(\epsilon) = \sup\{c|g_n(c) \leq \epsilon\}$ exists for every $\epsilon > 0$; (this include the case that $g_n(c) \leq \epsilon$ for all c, where we denote $g_n^{-1}(\epsilon) = +\infty$).

(2) $\{\lambda_n\} \downarrow 0$ where

$$\lambda_n \triangleq \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}\Big(\sup_{\mathbf{y}:\rho(\mathbf{y},\mathbf{s}_i)\leq c_n} |l(\mathbb{A}(\mathcal{S}_n), \mathbf{s}_i) - l(\mathbb{A}(\mathcal{S}_n), \mathbf{y})| > g_n(\rho(\mathbf{s}_i, \mathbf{y}))\Big).$$

(3) For all $\epsilon > 0$

$$\lim_{n\to\infty} \frac{K(\frac{\min(g^{-1}(\epsilon), c_n)}{2})}{n} = 0;$$
$$\limsup_{n\to\infty} \frac{-\ln \min(g^{-1}(\epsilon), c_n)}{n} \leq 0.$$

Then $\mathbb{A}$ is robust w.r.t. almost every $\{\mathcal{S}_n\}$.

PROOF. The proof is identical to that of Theorem 8.10, with the following two modifications:

(1) $Z(M)$, defined same as in the proof of Theorem 8.10, admits the following equation:

$$Z(M) = \mathbb{E}_\pi\{(f(\mathbf{t}) - M)\mathbf{1}(f(\mathbf{t}) > M)\},$$

due to the fact that the strong law of large numbers holds for Doeblin chain.

(2) H_n^M is now defined as

$$H_n^M \triangleq \Big\{(\mathbf{t}_1, \cdots, \mathbf{t}_n)|\frac{1}{n}\sum_{i=1}^{n}(f(\mathbf{t}_i) - M)\mathbf{1}(f(\mathbf{t}_i) > M)$$
$$\leq \mathbb{E}_\pi[(f(\mathbf{t}) - M)\mathbf{1}(f(\mathbf{t}) > M)] + \kappa\Big\}.$$

Notice that law of large numbers holds for Doeblin chain, hence $\mu(H_n^M) \uparrow 1$. Plugging these modifications into the proof of Theorem 8.10, we establish the theorem. □

Note that the conditions in Theorem 8.16 are identical to these of Theorem 8.10, except that the samples are drawn from a Doeblin chain.

8.5. Chapter summary

The main message of this chapter is that robustness of learning algorithms is a *necessary and sufficient* condition for generalizability. To the best of our knowledge, this is the first "if and only if" condition for algorithms other than ERM. Examples of conditions that ensure robustness were investigated, which resulted in novel generalizability results as well as new proofs of known results. In addition to the standard IID setup, the proposed approach was also applied to the case where the samples are generated according to a Markov chain.

Both robustness and generalizability of learning algorithms have been extensively investigated. However, their relationship has not been explored until recently. The main thrust of this work is to formalize the observation that good learning algorithms tend to be robust and provide another answer to the following fundamental question: "what is the reason that makes learning algorithms work?"

CHAPTER 9

Sparse Algorithms are not Stable: A No-free-lunch Theorem

In Chapter 7 we showed that Lasso as a sparse algorithm is not stable. Indeed, such relationship holds in a more broader context, as we show in this chapter, any sparse algorithm is non-stable. To be more specified: We consider two widely used notions in machine learning, namely: *sparsity* and *algorithmic stability*. Both notions are deemed desirable in designing algorithms, and are believed to lead to good generalization ability. In this paper, we show that these two notions contradict each other. That is, a sparse algorithm can not be stable and vice versa. Thus, one has to tradeoff sparsity and stability in designing a learning algorithm. We further present some examples of stable (hence non-sparse) algorithms and sparse (hence non-stable) algorithms to illustrate the implication of this theorem. Part of the material in this chapter appears in [**175**] and [**176**].

9.1. Introduction

Regression and classification are important problems with impact in a broad range of applications. Given data points encoded by the rows of a matrix A, and observations or labels $\mathbf{b}$, the basic goal is to find a (linear) relationship between A and $\mathbf{b}$. Various objectives are possible, for example in regression, on may consider

minimizing the least squared error, $||A\mathbf{w} - \mathbf{b}||_2$, or perhaps in case of a generative model assumption, minimizing the generalization error, i.e., the expected error of the regressor $\mathbf{x}$ on the next sample generated: $\mathbb{E}||\mathbf{a}^\top \mathbf{w} - b||$. In addition to such objectives, one may ask for solutions, $\mathbf{w}$, that have additional structural properties. In the machine learning literature, much work has focused on obtaining solutions with special properties.

Two properties of particular interest are *sparsity* of the solution, and the *stability* of the algorithm. Stability in this context, refers to the property that when given two very similar data sets, an algorithm's output varies little. More specifically, an algorithm is stable if its output changes very little when given two data sets differing on only one sample (this is known as the leave-one-out error). When this difference decays in the number of samples, that decay rate can be used directly to prove good generalization ability [**32**]. This stability property is also used extensively in the statistical learning community. For example, in [**142**] the author uses stability properties of ℓ^2-regularized SVM to establish its consistency.

Similarly, numerous algorithms that encourage sparse solutions have been proposed in virtually all fields in machine learning, including Lasso, ℓ_1-SVM, Deep Belief Network, Sparse PCA [**146, 180, 89, 46, 105**, and many others], mainly because of the following reasons: (i) a sparse solution is less complicated and hence generalizes well [**78**]; (ii) a sparse solution has good interpretability or less cost [**42, 104, 36, 58**]; and (iii) sparse algorithms may be computationally much easier to implement, store, compress, etc.

In this chapter, we investigate the mutual relationship of these two concepts. In particular, we show that sparse algorithms are not stable: if an algorithm "encourages sparsity" (in a sense defined precisely below) then its sensitivity to small perturbations of the input data remains bounded away from zero, i.e., it has no uniform stability properties. We define these notions exactly and precisely in Section 9.2.

We prove this no-free-lunch theorem by constructing an instance where the leave-one-out error of the algorithm is bounded away from zero by exploiting the property that a sparse algorithm can have non-unique optimal solutions.

This chapter is organized as follows. We make necessary definitions in Section 9.2 and provide the no-free-lunch theorem based on these definitions in Section 9.3. Sections 9.2 and 9.3 are devoted to regression algorithms; in Section 9.4 we generalize the theorem to arbitrary loss functions. In Section 9.5 we discuss the justification of the particular notions of stability and sparsity considered in this paper. Concluding remarks are given in Section 9.6.

9.2. Definitions and Assumptions

The first part of the chapter considers regression algorithms that find a weight vector, $\mathbf{w}^*$ in the *feature space*. The goal of any algorithm we consider is to minimize the loss given a new observation $(\hat{b}, \hat{\mathbf{a}})$. Initially we consider the loss function $l(\mathbf{w}^*, (\hat{b}, \hat{\mathbf{a}})) = |\hat{b} - \hat{\mathbf{a}}^\top \mathbf{w}^*|$. Here $\mathbf{a}$ is the vector of feature values of the observation. In the standard regression problem, the learning algorithm $\mathbb{L}$ obtains the candidate solution $\mathbf{w}^*$ by minimizing the empirical loss $||A\mathbf{w} - \mathbf{b}||_2$, or the regularized empirical loss. For a given objective function, we can compare two solutions $\mathbf{w}^1, \mathbf{w}^2$ by considering their empirical loss. We adopt a somewhat more general framework, considering only the partial ordering induced by any learning algorithm $\mathbb{L}$ and training set $(\mathbf{b}, A)$. That is, given two candidate solutions, $\mathbf{w}^1, \mathbf{w}^2$, we write

$$\mathbf{w}^1 \preceq_{(\mathbf{b},A)} \mathbf{w}^2,$$

if on input $(\mathbf{b}, A)$, the algorithm $\mathbb{L}$ would select $\mathbf{w}^2$ before $\mathbf{w}^1$. In short, given an algorithm $\mathbb{L}$, each sample set $(\mathbf{b}, A)$ defines an order relationship $\preceq_{(\mathbf{b},A)}$ among all candidate solutions $\mathbf{w}$. This order relationship defines a family of "best" solutions, and one of these, $\mathbf{w}^*$ is the output of the algorithm. We denote this by writing $\mathbf{w}^* \in \mathbb{L}_{(\mathbf{b},A)}$.

Thus, by defining a data-dependent partial ordering on the space of solutions, we can speak more generically of algorithms, their stability, and their sparseness. As we define below, an algorithm $\mathbb{L}$ is sparse if the set $\mathbb{L}_{(\mathbf{b},A)}$ of optimal solutions contains a sparse solution, and an algorithm is stable if the sets $\mathbb{L}_{(\mathbf{b},A)}$ and $\mathbb{L}_{(\hat{\mathbf{b}},\hat{A})}$ do not contain solutions that are very far apart, when $(\mathbf{b}, A)$ and $(\hat{\mathbf{b}}, \hat{A})$ differ on only one point.

We make a few assumptions on the preference ordering, and hence on the algorithms that we consider:

ASSUMPTION 9.1. *(i) Given j, $\mathbf{b}$, A, $\mathbf{w}^1$ and $\mathbf{w}^2$, if*

$$\mathbf{w}^1 \preceq_{(\mathbf{b},A)} \mathbf{w}^2,$$

and

$$w_j^1 = w_j^2 = 0,$$

then for any $\hat{\mathbf{a}}$,

$$\mathbf{w}^1 \preceq_{(\mathbf{b},\hat{A})} \mathbf{w}^2,$$

where

$$\hat{A} = (\mathbf{a}_1, \cdots, \mathbf{a}_{j-1}, \hat{\mathbf{a}}, \mathbf{a}_{j+1}, \cdots, \mathbf{a}_m)\,.$$

(ii) Given $\mathbf{b}$, A, $\mathbf{w}^1$, $\mathbf{w}^2$, b' and $\mathbf{z}$, if

$$\mathbf{w}^1 \preceq_{(\mathbf{b},A)} \mathbf{w}^2,$$

and

$$b = \mathbf{z}^\top \mathbf{w}^2,$$

then

$$\mathbf{w}^1 \preceq_{(\overline{\mathbf{b}},\overline{A})} \mathbf{w}^2,$$

where

$$\overline{\mathbf{b}} = \begin{pmatrix} \mathbf{b} \\ b' \end{pmatrix}; \quad \overline{A} = \begin{pmatrix} A \\ \mathbf{z}^\top \end{pmatrix}.$$

(iii) Given j, $\mathbf{b}$, A, $\mathbf{w}^1$ *and* $\mathbf{w}^2$, *if*

$$\mathbf{w}^1 \preceq_{(\mathbf{b},A)} \mathbf{w}^2,$$

then

$$\hat{\mathbf{w}}^1 \preceq_{(\mathbf{b},\tilde{A})} \hat{\mathbf{w}}^2,$$

where

$$\hat{\mathbf{w}}^i = \begin{pmatrix} \mathbf{w}^i \\ 0 \end{pmatrix}, \ i = 1, 2; \quad \tilde{A} = (A, \mathbf{0}).$$

(iv) Given $\mathbf{b}$, A, $\mathbf{w}^1$, $\mathbf{w}^2$ *and* $P \in \mathbb{R}^{m \times m}$ *a permutation matrix, if*

$$\mathbf{w}^1 \preceq_{(\mathbf{b},A)} \mathbf{w}^2,$$

then

$$P^\top \mathbf{w}^1 \preceq_{(\mathbf{b},AP)} P^\top \mathbf{w}^2.$$

Part (i) says that the value of a column corresponding to a non-selected feature has no effect on the ordering; (ii) says that adding a sample that is perfectly predicted by a particular solution, cannot decrease its place in the partial ordering; (iii) says the order relationship is preserved when a trivial (all zeros) feature is added; (iv) says that the partial ordering and hence the algorithm, is feature-wise symmetric. These assumptions are intuitively appealing and satisfied by most algorithms including, for instance, standard regression, and regularized regression.

In what follows, we will define precisely what we mean by stability and sparseness. We recall the definition of uniform (algorithmic) stability first, as given in [**32**]. We let $\mathcal{Z}$ denote the space of points and labels (typically this will be a compact subset of $\mathbb{R}^{m+1}$) so that $S \in \mathcal{Z}^n$ denotes a collection of n labelled training points. For regression problems, therefore, we have $S = (\mathbf{b}, A) \in \mathcal{Z}^n$. We let $\mathbb{L}$ denote a learning algorithm, and for $(\mathbf{b}, A) \in \mathcal{Z}^n$, we let $\mathbb{L}_{(\mathbf{b},A)}$ denote the output of the learning algorithm (i.e., the regression function it has learned from the training data). Then given a loss function l, and a labelled point $s = (\mathbf{z}, b) \in \mathcal{Z}$, $l(\mathbb{L}_{(\mathbf{b},A)}, s)$ denotes the loss of the

algorithm that has been trained on the set $(\mathbf{b}, A)$, on the data point s. Thus for squared loss, we would have $l(\mathbb{L}_{(\mathbf{b},A)}, s) = \|\mathbb{L}_{(\mathbf{b},A)}(\mathbf{z}) - b\|_2$.

DEFINITION 9.1. *An algorithm $\mathbb{L}$ has uniform stability β_n with respect to the loss function l if the following holds:*

$$\forall(\mathbf{b}, A) \in \mathcal{Z}^n, \forall i \in \{1, \cdots, n\}, \qquad \|l(\mathbb{L}_{(\mathbf{b},A)}, \cdot) - l(\mathbb{L}_{(\mathbf{b},A)^{\setminus i}}, \cdot)\|_\infty \leq \beta_n.$$

Here $\mathbb{L}_{(\mathbf{b},A)^{\setminus i}}$ stands for the learned solution with the i^{th} sample removed from $(\mathbf{b}, A)$, i.e., with the i^{th} row of A and the i^{th} element of $\mathbf{b}$ removed.

At first glance, this definition may seem too stringent for any reasonable algorithm to exhibit good stability properties. However, as shown in [**32**], *Tikhonov-regularized regression has stability that scales as* $1/n$. Stability can be used to establish strong PAC bounds. For example, in [**32**] they show that if we have n samples, β_n denotes the uniform stability, and M a bound on the loss, then

$$R \leq R_{\text{emp}} + 2\beta_n + (4n\beta_n + M)\sqrt{\frac{\ln 1/\delta}{2n}},$$

where R denotes the Bayes loss, and R_{emp} the empirical loss.

Since Lasso is an example of an algorithm that yields sparse solutions, one implication of the results of this chapter is that while ℓ^2-regularized regression yields sparse solutions, ℓ^1-regularized regression does not. We show that the stability parameter of Lasso does not decrease in the number of samples (compared to the $O(1/n)$ decay for ℓ^2-regularized regression). In fact, we show that Lasso's stability is, in the following sense, as bad as it gets. To this end, we define the notion of the trivial bound, which is the worst possible error a training algorithm can have for arbitrary training set and testing sample labelled by zero.

DEFINITION 9.2. *Given a subset from which we can draw n labelled points, $\mathcal{Z} \subseteq \mathbb{R}^{n\times(m+1)}$ and a subset for one unlabelled point, $\mathcal{X} \subseteq \mathbb{R}^n$, a trivial bound for a learning*

algorithm $\mathbb{L}$ *w.r.t.* $\mathcal{Z}$ *and* $\mathcal{X}$ *is*

$$\mathfrak{b}(\mathbb{L}, \mathcal{Z}, \mathcal{X}) \triangleq \max_{(\mathbf{b},A)\in\mathcal{Z},\mathbf{z}\in\mathcal{X}} l\big(\mathbb{L}_{(\mathbf{b},A)}, (\mathbf{z}, 0)\big).$$

As above, $l(\cdot,\cdot)$ *is a given loss function.*

Notice that the trivial bound does not diminish as the number of samples, n, increases, since by repeatedly choosing the worst sample, the algorithm will yield the same solution.

Our next definition makes precise the notion of sparsity of an algorithm which we use.

DEFINITION 9.3. *An algorithm* $\mathbb{L}$ *is said to* Identify Redundant Features *(I.R.F. for short) if given* $(\mathbf{b}, A)$*, there exists* $\mathbf{x}^* \in \mathbb{L}_{(\mathbf{b},A)}$ *such that if* $\mathbf{a}_i = \mathbf{a}_j$*, then not both* w_i *and* w_j *are nonzero. That is,*

$$\forall i \neq j, \quad \mathbf{a}_i = \mathbf{a}_j \Rightarrow w_i^* w_j^* = 0.$$

I.R.F. means that at least one solution of the algorithm does not select both features if they are identical. We note that this is a quite weak notion of sparsity. An algorithm that achieves reasonable sparsity (such as Lasso) should be able to I.R.F.

9.3. Main Theorem

The next theorem is the main contribution of this chapter. It says that if an algorithm is sparse, in the sense that it identifies redundant features as in the definition above, then that algorithm *is not stable*. One notable example that satisfies this theorem is Lasso.

THEOREM 9.1. *Let* $\mathcal{Z} \subseteq \mathbb{R}^{n\times(m+1)}$ *denote the domain of sample sets of* n *points each with* m *features, and* $\mathcal{X} \subseteq \mathbb{R}^{m+1}$ *the domain of new observations consisting of a point in* $\mathbb{R}^m$*, and its label in* $\mathbb{R}$*. Similarly, let* $\hat{\mathcal{Z}} \subseteq \mathbb{R}^{n\times(2m+1)}$ *be the domain of sample sets of* n *points each with* $2m$ *features, and* $\hat{\mathcal{X}} \subseteq \mathbb{R}^{2m+1}$ *be the domain of new*

observations. Suppose that these sets of samples and observations are such that:

$$(\mathbf{b}, A) \in \mathcal{Z} \Longrightarrow (\mathbf{b}, A, A) \in \hat{\mathcal{Z}}$$
$$(0, \mathbf{z}^\top) \in \mathcal{X} \Longrightarrow (0, \mathbf{z}^\top, \mathbf{z}^\top) \in \hat{\mathcal{X}}.$$

If a learning algorithm $\mathbb{L}$ satisfies Assumption 9.1 and identifies redundant features, its uniform stability bound β is lower bounded by $\mathfrak{b}(\mathbb{L}, \mathcal{Z}, \mathcal{X})$, and in particular does not go to zero with n.

PROOF. Let $(\mathbf{b}, A)$ and $(0, \mathbf{z}^\top)$ be the sample set and the new observation such that they jointly achieve $\mathfrak{b}(\mathbb{L}, \mathcal{Z}, \mathcal{X})$, i.e., for some $\mathbf{w}^* \in \mathbb{L}(\mathbf{b}, A)$, we have

$$\mathfrak{b}(\mathbb{L}, \mathcal{Z}, \mathcal{X}) = l\big(\mathbf{w}^*, (0, \mathbf{z})\big). \tag{9.1}$$

Let $0^{n\times m}$ be the $n \times m$ 0-matrix, and $\mathbf{0}$ stand for the zero vector of length m. We denote

$$\hat{\mathbf{z}} \triangleq (\mathbf{0}^\top, \mathbf{z}^\top); \qquad \hat{A} \triangleq (A,\ A);$$
$$\tilde{\mathbf{b}} \triangleq \begin{pmatrix} \mathbf{b} \\ b' \end{pmatrix}; \qquad \tilde{A} \triangleq \begin{pmatrix} A, & A \\ \mathbf{0}^\top, & \mathbf{z}^\top \end{pmatrix}.$$

We first show that

$$\begin{pmatrix} \mathbf{0} \\ \mathbf{w}^* \end{pmatrix} \in \mathbb{L}_{(\mathbf{b},\hat{A})}; \quad \begin{pmatrix} \mathbf{w}^* \\ \mathbf{0} \end{pmatrix} \in \mathbb{L}_{(\tilde{\mathbf{b}},\tilde{A})}. \tag{9.2}$$

Notice that $\mathbb{L}$ is feature-wise symmetric and identifies redundant features, hence there exists a $\mathbf{w}'$ such that

$$\begin{pmatrix} \mathbf{0} \\ \mathbf{w}' \end{pmatrix} \in \mathbb{L}_{(\mathbf{b},\hat{A})}.$$

Since $\mathbf{w}^* \in \mathbb{L}_{(\mathbf{b},A)}$, we have

$$\begin{aligned}
& \mathbf{w}' \preceq_{(\mathbf{b},A)} \mathbf{w}^* \\
\Rightarrow \quad & \begin{pmatrix} \mathbf{0} \\ \mathbf{w}' \end{pmatrix} \preceq_{(\mathbf{b},(0^{n\times m},A))} \begin{pmatrix} \mathbf{0} \\ \mathbf{w}^* \end{pmatrix} \\
\Rightarrow \quad & \begin{pmatrix} \mathbf{0} \\ \mathbf{w}' \end{pmatrix} \preceq_{(\mathbf{b},\hat{A})} \begin{pmatrix} \mathbf{0} \\ \mathbf{w}^* \end{pmatrix} \\
\Rightarrow \quad & \begin{pmatrix} \mathbf{0} \\ \mathbf{w}^* \end{pmatrix} \in \mathbb{L}_{(\mathbf{b},\hat{A})}.
\end{aligned}$$

The first implication follows from Assumption 9.1(iii), and the second from (i).

Now notice by feature-wise symmetry, we have

$$\begin{pmatrix} \mathbf{w}^* \\ \mathbf{0} \end{pmatrix} \in \mathbb{L}_{(\mathbf{b},\hat{A})}.$$

Furthermore,

$$0 = (\mathbf{0}^\top, \mathbf{z}^\top) \begin{pmatrix} \mathbf{w}^* \\ \mathbf{0} \end{pmatrix},$$

and thus by Assumption 9.1(ii) we have

$$\begin{pmatrix} \mathbf{w}^* \\ \mathbf{0} \end{pmatrix} \in \mathbb{L}_{(\tilde{\mathbf{b}},\tilde{A})}.$$

Hence (9.2) holds. This leads to

$$l\big(\mathbb{L}_{(\mathbf{b},\hat{A})}, (0,\hat{\mathbf{z}})\big) = l(\mathbf{w}^*, (0,\mathbf{z})); \quad l\big(\mathbb{L}_{(\tilde{\mathbf{b}},\tilde{A})}, (0,\hat{\mathbf{z}})\big) = 0.$$

By definition of the uniform bound, we have

$$\beta \geq l\big(\mathbb{L}_{(\mathbf{b},\hat{A})}, (0,\hat{\mathbf{z}})\big) - l\big(\mathbb{L}_{(\tilde{\mathbf{b}},\tilde{A})}, (0,\hat{\mathbf{z}})\big).$$

Hence by (9.1) we have $\beta \geq \mathfrak{b}(\mathbb{L}, \mathcal{Z}, \mathcal{X})$, which establishes the theorem. □

Theorem 9.1 not only means that a sparse algorithm is not stable, it also states that if an algorithm is stable, there is no hope to achieve satisfactory sparsity, since it cannot even identify redundant features. Note that indeed, l_2 regularized regression is stable, and does not identify redundant features.

9.4. Generalization to Arbitrary Loss

The results derived so far can easily be generalized to algorithms with arbitrary loss function $l(\mathbf{x}^*, (\hat{b}, \hat{\mathbf{a}})) = f_m(\hat{b}, \hat{a}_1 w_i^*, \cdots, \hat{a}_m w_m^*)$ for some f_m. Here, $\hat{a}_i$ and w_i^* denote the i^{th} component of $\hat{\mathbf{a}} \in \mathbb{R}^m$ and $\mathbf{w}^* \in \mathbb{R}^m$, respectively. We assume that the function $f_m(\cdot)$ satisfies the following conditions

$$\begin{aligned} &(a) \quad f_m(b, v_1, \cdots, v_i, \cdots, v_j, \cdots v_m) = f_m(b, v_1, \cdots, v_j, \cdots, v_i, \cdots v_m);\ \forall b, \mathbf{v}, i, j. \\ &(b) \quad f_m(b, v_1, \cdots, v_m) = f_{m+1}(b, v_1, \cdots, v_m, 0);\ \forall b, \mathbf{v}. \end{aligned} \tag{9.3}$$

We require following modifications of Assumption 9.1(ii) and Definition 9.2.

ASSUMPTION 9.2. *(ii) Given* $\mathbf{b}$, A, $\mathbf{w}^1$, $\mathbf{w}^2$, b' *and* $\mathbf{z}$ *if*

$$\mathbf{w}^1 \preceq_{(\mathbf{b},A)} \mathbf{w}^2, \quad l(\mathbf{w}^2, (b', \mathbf{z})) \leq l(\mathbf{w}^1, (b', \mathbf{z}))$$

then

$$\mathbf{w}^1 \preceq_{(\overline{\mathbf{b}},\overline{A})} \mathbf{w}^2, \quad \textit{where } \overline{\mathbf{b}} = \begin{pmatrix} \mathbf{b} \\ b' \end{pmatrix}; \quad \overline{A} = \begin{pmatrix} A \\ \mathbf{z}^\top \end{pmatrix}.$$

DEFINITION 9.4. *Given* $\mathcal{Z} \subseteq \mathbb{R}^{n \times (m+1)}$ *and* $\mathcal{X} \subseteq \mathbb{R}^{m+1}$, *a trivial bound for a learning algorithm* $\mathbb{L}$ *w.r.t.* $\mathcal{Z}$ *and* $\mathcal{X}$ *is*

$$\hat{\mathfrak{b}}(\mathbb{L}, \mathcal{Z}, \mathcal{X}) \triangleq \max_{(\mathbf{b},A) \in \mathcal{Z}, (b,\mathbf{z}) \in \mathcal{X}} \Big\{ l\big(\mathbb{L}_{(\mathbf{b},A)}, (b, \mathbf{z})\big) - l\big(\mathbf{0}, (b,\ \mathbf{z})\big) \Big\}.$$

These modifications account for the fact that under an arbitrary loss function, there may not exist a sample that can be perfectly predicted by the zero vector. With these modifications, we have the same no-free-lunch theorem.

THEOREM 9.2. *As before, let $\mathcal{Z} \subseteq \mathbb{R}^{n\times(m+1)}$, and $\hat{\mathcal{Z}} \subseteq \mathbb{R}^{n\times(2m+1)}$ be the domain of sample sets, and $\mathcal{X} \subseteq \mathbb{R}^{m+1}$, and $\hat{\mathcal{X}} \subseteq \mathbb{R}^{2m+1}$ be the domain of new observations, with m and $2m$ features respectively. Suppose, as before, that these sets satisfy*

$$(\mathbf{b}, A) \in \mathcal{Z} \Longrightarrow (\mathbf{b}, A, A) \in \hat{\mathcal{Z}}$$
$$(b', \mathbf{z}^\top) \in \mathcal{X} \Longrightarrow (b', \mathbf{z}^\top, \mathbf{z}^\top) \in \hat{\mathcal{X}}.$$

If a learning algorithm $\mathbb{L}$ satisfies Assumption 9.2 and identifies redundant features, its uniform stability bound β is lower bounded by $\hat{\mathfrak{b}}(\mathbb{L}, \mathcal{Z}, \mathcal{X})$.

PROOF. This proof follows a similar line of reasoning as the proof of Theorem 9.1. Let $(\mathbf{b}, A)$ and $(b', \mathbf{z}^\top)$ be the sample set and the new observation such that they jointly achieve $\hat{\mathfrak{b}}(\mathbb{L}, \mathcal{Z}, \mathcal{X})$, i.e., let $\mathbf{w}^* \in \mathbb{L}(\mathbf{b}, A)$, and

$$\begin{aligned}\hat{\mathfrak{b}}(\mathbb{L}, \mathcal{Z}, \mathcal{X}) &= l\big(\mathbf{w}^*, (b', \mathbf{z})\big) - l\big(\mathbf{0}, (b', \mathbf{z})\big) \\ &= f_m(b', w_1^* z_1, \cdots, w_m^* z_m) - f(b', 0, \cdots, 0).\end{aligned}$$

Let $0^{n\times m}$ be the $n \times m$ 0-matrix, and $\mathbf{0}$ stand for the zero vector of length m. We denote

$$\hat{\mathbf{z}} \triangleq (\mathbf{0}^\top, \mathbf{z}^\top); \qquad \hat{A} \triangleq (A, A);$$
$$\tilde{\mathbf{b}} \triangleq \begin{pmatrix} \mathbf{b} \\ b' \end{pmatrix}; \qquad \tilde{A} \triangleq \begin{pmatrix} A, & A \\ \mathbf{0}^\top, & \mathbf{z}^\top \end{pmatrix}.$$

To prove the theorem, it suffices show that there exists $\mathbf{w}^1$, $\mathbf{w}^2$ such that

$$\mathbf{w}^1 \in \mathbb{L}_{(\mathbf{b}, \hat{A})}; \quad \mathbf{w}^2 \in \mathbb{L}_{(\tilde{\mathbf{b}}, \tilde{A})},$$

and

$$l\big(\mathbf{w}^1, (b', \hat{\mathbf{z}})\big) - l\big(\mathbf{w}^2, (b', \hat{\mathbf{z}})\big) \geq \hat{\mathfrak{b}}(\mathbb{L}, \mathcal{Z}, \mathcal{X})$$

where again,

$$\hat{\mathfrak{b}}(\mathbb{L}, \mathcal{Z}, \mathcal{X}) = f_m(b', w_1^* z_1, \cdots, w_m^* z_m) - f_m(b', 0, \cdots, 0).$$

By an identical argument as that of Proof 9.1, we have

$$\begin{pmatrix} \mathbf{0} \\ \mathbf{w}^* \end{pmatrix} \in \mathbb{L}_{(\mathbf{b},\hat{A})}.$$

Hence there exists $\mathbf{w}^1 \in \mathbb{L}_{(\mathbf{b},\hat{A})}$ such that

$$\begin{aligned} l\left(\mathbf{w}^1, (b', \hat{\mathbf{z}})\right) &= l\left(\begin{pmatrix} \mathbf{0} \\ \mathbf{w}^* \end{pmatrix}, (b', \hat{\mathbf{z}})\right) \\ &= f_m(b', w_1^* z_1, \cdots, w_m^* z_m). \end{aligned} \tag{9.4}$$

The last equality follows from Equation (9.3) easily.

Now notice that by feature-wise symmetry, we have

$$\begin{pmatrix} \mathbf{w}^* \\ \mathbf{0} \end{pmatrix} \in \mathbb{L}_{(\mathbf{b},\hat{A})}.$$

Hence there exists $\mathbf{w}^2 \in \mathbb{L}_{(\tilde{\mathbf{b}},\tilde{A})}$ such that

$$\begin{aligned} l\left(\mathbf{w}^2, (b', \hat{\mathbf{z}})\right) &\leq l\left(\begin{pmatrix} \mathbf{w}^* \\ \mathbf{0} \end{pmatrix}, (b', \hat{\mathbf{z}})\right) \\ &= f_m(b', 0, \cdots, 0). \end{aligned} \tag{9.5}$$

The last equality follows from Equation (9.3). The inequality here holds because by Assumption 9.2(ii), if there is no $\mathbf{w}^2 \in \mathbb{L}_{(\tilde{\mathbf{b}},\tilde{A})}$ that satisfies the inequality, then we have

$$\mathbf{w}^2 \preceq_{(\tilde{\mathbf{b}},\tilde{A})} \begin{pmatrix} \mathbf{w}^* \\ \mathbf{0} \end{pmatrix}$$

which then implies that

$$\Rightarrow \quad \begin{pmatrix} \mathbf{w}^* \\ \mathbf{0} \end{pmatrix} \in \mathbb{L}_{(\tilde{\mathbf{b}},\tilde{A})},$$

which is absurd.

Combining (9.4) and (9.5) proves the theorem. □

9.5. Discussions

To see that the two notions that we consider are not too restrictive, we list in this section some algorithms that either admit a diminishing uniform stability bound or identify redundant features. Thus, by applying Theorem 9.2 we conclude that they are either non-sparse or non-stable. We also quota some empirical results to show that algorithms that identify redundant features do achieve sparsity.

9.5.1. Stable algorithms. All algorithms listed in this subsection has a uniform stability bound that decreases as $O(\frac{1}{n})$, and is hence stable. Example 9.1 to 9.5 and adapted from [**32**].

EXAMPLE 9.1 (Bounded SVM regression). *Assume k is a bounded kernel, that is $k(\mathbf{x}, \mathbf{x}) \leq \kappa^2$. Let $\mathcal{F}$ denote the RKHS space of k. Consider $\mathcal{Y} = [0,\, B]$ and the loss function*

$$l(f, (y, \mathbf{x})) = |f(\mathbf{x}) - y|_\epsilon = \begin{cases} 0 & \text{if } |f(\mathbf{x}) - y| \leq \epsilon; \\ |f(\mathbf{x}) - y| - \epsilon & \text{otherwise.} \end{cases}$$

The SVM regression algorithm with kernel k is defined as

$$\mathbb{L}_S = \arg\min_{g \in \mathcal{F}} \Big\{ \sum_{i=1}^{n} l(g, (y_i, \mathbf{x}_i)) + \lambda n \|g\|_\kappa^2 \Big\}; \quad \text{here: } S = ((y_1, \mathbf{x}_1), \cdots, (y_n, \mathbf{x}_n)).$$

Then, its uniform stability satisfies

$$\beta_n \leq \frac{\kappa^2}{2\lambda n}.$$

EXAMPLE 9.2 (Soft-margin SVM classification). *Assume k is a bounded kernel, that is $k(\mathbf{x}, \mathbf{x}) \leq \kappa^2$. Let $\mathcal{F}$ denote the RKHS space of k. Consider $\mathcal{Y} = \{0,\, 1\}$[1] and the loss function*

$$l(f, (y, \mathbf{x})) = (1 - (2y - 1)f(\mathbf{x}))^+ = \begin{cases} 1 - (2y - 1)f(\mathbf{x}) & \text{if } 1 - (2y - 1)f(\mathbf{x}) > 0; \\ 0 & \text{otherwise.} \end{cases}$$

[1]This is slightly different from but equivalent to the standard setup where $\mathcal{Y} = \{-1,\, 1\}$.

The soft-margin SVM classification algorithm with kernel k is defined as

$$\mathbb{L}_S = \arg\min_{g\in\mathcal{F}} \Big\{ \sum_{i=1}^{n} l(g,(y_i,\mathbf{x}_i)) + \lambda n \|g\|_\kappa^2 \Big\}; \quad \textit{here: } S = ((y_1,\mathbf{x}_1),\cdots,(y_n,\mathbf{x}_n)).$$

Then, its uniform stability satisfies

$$\beta_n \le \frac{\kappa^2}{2\lambda n}.$$

EXAMPLE 9.3 (RKHS regularized least square regression). *Assume k is a bounded kernel, that is $k(\mathbf{x},\mathbf{x}) \le \kappa^2$. Let $\mathcal{F}$ denote the RKHS space of k. Consider $\mathcal{Y} = [0,\, B]$ and the loss function*

$$l(f,(y,\mathbf{x})) = (f(\mathbf{x}) - y)^2.$$

The regularized least square regression algorithm with kernel k is defined as

$$\mathbb{L}_S = \arg\min_{g\in\mathcal{F}} \Big\{ \sum_{i=1}^{n} l(g,(y_i,\mathbf{x}_i)) + \lambda n \|g\|_\kappa^2 \Big\}; \quad \textit{here: } S = ((y_1,\mathbf{x}_1),\cdots,(y_n,\mathbf{x}_n)).$$

Then, its uniform stability satisfies

$$\beta_n \le \frac{2\kappa^2 B^2}{\lambda n}.$$

Next example is the relative entropy regularization. In this case, we are given a class of base hypotheses, and the output of the algorithm is a mixture of them, or more precisely a probability distribution over the class of base hypothese.

EXAMPLE 9.4 (Relative Entropy Regularization). *Let $\mathcal{H} = \{h_\theta : \theta \in \Theta\}$ be the class of base hypotheses, where Θ is a measurable space with a reference measure. Let $\mathcal{F}$ denote the set of probability distributions over Θ dominated by the reference measure. Consider the loss function*

$$l(f,\mathbf{z}) = \int_\Theta r(h_\theta,\mathbf{z}) f(\theta) d\theta;$$

where $r(\cdot,\cdot)$ is a loss function bounded by M. Further let f_0 be a fixed element of $\mathcal{F}$ and $K(\cdot,\cdot)$ denote the Kullback-Leibler divergence. The relative entropy regularized

algorithm is defined as

$$\mathbb{L}_S = \arg\min_{g\in\mathcal{F}} \Big\{ \sum_{i=1}^{n} l(g, \mathbf{z}_i) + \lambda n K(g, f_0) \Big\}; \quad \textit{here: } S = (\mathbf{z}_1, \cdots, \mathbf{z}_n).$$

Then, its uniform stability satisfies

$$\beta_n \leq \frac{M^2}{\lambda n}.$$

A special case of relative entropy regularization is the following *maximum entropy discrimination* proposed in [**92**].

EXAMPLE 9.5 (Maximum entropy discrimination). *Let* $\mathcal{H} = \{h_{\theta,\gamma} : \theta \in \Theta, \gamma \in \mathbb{R}\}$ *with* $h_{\theta,\gamma} = h_\theta$. *Consider* $\mathcal{Y} = \{0, 1\}$ *and the loss function*

$$l(f, \mathbf{z}) = \left(\int_{\Theta,\mathbb{R}} [\gamma - (2y-1)h_\theta(\mathbf{z})] f(\theta) d\theta \right)_+;$$

where $[\gamma - (2y-1)h_\theta(\mathbf{z})]$ *is bounded by* M. *The maximum entropy discrimination is a real-valued classifier defined as*

$$\mathbb{L}_S = \arg\min_{g\in\mathcal{F}} \Big\{ \sum_{i=1}^{n} l(g, \mathbf{z}_i) + \lambda n K(g, f_0) \Big\}; \quad \textit{here: } S = (\mathbf{z}_1, \cdots, \mathbf{z}_n).$$

Then, its uniform stability satisfies

$$\beta_n \leq \frac{M}{\lambda n}.$$

If an algorithm is not stable, one way to stabilize it is to averaging its solutions trained on small bootstrap subsets of the training set, a process called subbagging [**69**], which we recall in the following example.

EXAMPLE 9.6. *let* $\mathbb{L}$ *be a learning algorithm with a stability* β_n, *and consider the following algorithm*

$$\hat{\mathbb{L}}_{\mathcal{D}}^{k}(\mathbf{x}) \triangleq \mathbb{E}_{\mathcal{S}} \left(\mathbb{L}_{\mathcal{S}}(\mathbf{x}) \right).$$

where $\mathbb{E}_{\mathcal{S}}$ *is the expectation of with respect to* k *points sampled in* $\mathcal{D}$ *uniformly* without replacement. *Then* $\hat{\mathbb{L}}^k$ *has a stability* $\hat{\beta}_n$ *satisfying*

$$\hat{\beta}_n \leq \frac{k}{n}\beta_k.$$

9.5.2. Sparse Algorithms. Next we list some algorithms that identify redundant features.

EXAMPLE 9.7 (ℓ_0 Minimization). *Subset selection algorithms based on minimizing* ℓ_0 *norm identifies redundant features. One example of such algorithm is the* canonical selection procedure [**74**]*, which is defined as*

$$\mathbf{w}^* = \arg\min_{\mathbf{w}\in\mathbb{R}^m} \{\|A\mathbf{w} - \mathbf{b}\|_2 + \lambda\|\mathbf{w}\|_0\}. \tag{9.6}$$

PROOF. Note that if a solution $\mathbf{w}^*$ achieves minimum and has non-zero weights on two redundant features i and i', then by constructing a $\hat{\mathbf{w}}$ such that $\hat{w}_i = w_i^* + w_{i'}^*$ and $\hat{w}_{i'} = 0$ we get a strictly better solution, which is a contradiction. Hence ℓ_0 minimizing algorithms IRF. □

It is known that in general finding the minimum of (9.6) is NP-hard [**115**]. Therefore, a convex relaxation, the ℓ_1 norm, is used instead to find a sparse solution. These algorithms either minimize the ℓ_1 norm of the solution under the constraint of a regression error, or minimize the convex combination of some regression error and the ℓ_1 norm of the solution.

EXAMPLE 9.8 (ℓ_1 Minimization). *The following subset selection algorithms based on minimizing* ℓ_1 *norm identify redundant features. There algorithms include:*

(1) Lasso [**146**] *defined as*

$$\mathbf{w}^* = \arg\min_{\mathbf{w}\in\mathbb{R}^m} \{\|A\mathbf{w} - \mathbf{b}\|_2 + \lambda\|\mathbf{w}\|_1\}.$$

(2) Basis Pursuit [**38**] *defined as the solution of the following optimization problem on* $\mathbf{w} \in \mathbb{R}^m$*:*

$$\begin{aligned} &\textit{Minimize:} \quad \|\mathbf{w}\|_1 \\ &\textit{Subject to:} \quad A\mathbf{w} = \mathbf{b}. \end{aligned}$$

(3) Dantzig Selector [**37**] *defined as*

$$\begin{aligned} &\textit{Minimize:} \quad \|\mathbf{w}\|_1 \\ &\textit{Subject to:} \quad \|A^*(A\mathbf{w} - \mathbf{b})\|_\infty \leq c. \end{aligned}$$

Here, A^* *is the complex conjugate of* A*, and* c *is some positive constant.*

(4) 1-norm SVM [**180, 105**] *defined as the solution of the following optimization problem on* $\boldsymbol{\alpha}$*,* $\boldsymbol{\xi}$ *and* γ*.*

$$\begin{aligned} &\textit{Minimize:} \quad \|\boldsymbol{\alpha}\|_1 + C\sum_{i=1}^{n} \xi_i \\ &\textit{Subject to:} \quad y_i \left\{ \sum_{j=1}^{n} \alpha_i k(\mathbf{x}_i, \mathbf{x}_j) + \gamma \right\} \geq 1 - \xi_i; \quad i = 1, \cdots, n; \\ &\qquad\qquad \xi_i \geq 0; \quad i = 1, \cdots, n. \end{aligned}$$

(5) ℓ_1 norm SVM regression [**133**] *defined as the solution of the following optimization problem on* $\boldsymbol{\alpha}$*,* $\boldsymbol{\xi}$ *and* γ*:*

$$\begin{aligned} &\textit{Minimize:} \quad \|\boldsymbol{\alpha}\|_1 + C\sum_{i=1}^{n} \xi_i \\ &\textit{Subject to:} \quad \left\{ \sum_{j=1}^{n} \alpha_i k(\mathbf{x}_i, \mathbf{x}_j) + \gamma \right\} - y_i \leq \varepsilon + \xi_i; \quad i = 1, \cdots, n; \\ &\qquad\qquad y_i - \left\{ \sum_{j=1}^{n} \alpha_i k(\mathbf{x}_i, \mathbf{x}_j) + \gamma \right\} \leq \varepsilon + \xi_i; \quad i = 1, \cdots, n \\ &\qquad\qquad \xi_i \geq 0; \quad i = 1, \cdots, n, \end{aligned}$$

where $\varepsilon > 0$ *is a fixed constant.*

PROOF. Given an optimal $\mathbf{w}^*$ we construct a new solution $\hat{\mathbf{w}}$ such that for any subset of redundant features I, $\sum_{i\in I} \mathbf{1}(\hat{w}_i \neq 0) \leq 1$ and $\sum_{i\in I} \hat{w}_i = \sum_{i\in I} w_i^*$. Thus, $\hat{\mathbf{w}}$ and $\mathbf{w}^*$ are equally good, which implies that any ℓ_1 minimizing algorithm has at leat one optimal solution that I.R.F. Hence such algorithm I.R.F. by definition. □

Empirical results also show that the outputs of algorithms that identify redundant features are much sparser than algorithms that do not identify redundant features. In [**106**], the authors reported the empirical performance (testing error and number of SV) of three algorithms: soft-margin SVM, 1-norm SVM and ℓ_0 minimizing SVM[2] on five UCI data sets [**3**], namely *Ionosphere, Pima, Wbc, Bupa* and *Sona*. Their empirical results clearly show that to achieve a similar testing error, soft-margin SVM, which does not identify redundant features, requires significantly more support vectors. We quote their results in Section 9.7 for completeness.

9.6. Chapter summary

In this chapter, we prove a no-free-lunch theorem show that sparsity and stability are at odds with each other. We show that if an algorithm is sparse, then its uniform stability is lower bounded by a nonzero constant. This also shows that any stable algorithm cannot be sparse. Thus we show that these two widely used concepts, namely *sparsity* and *algorithmic stability* conflict with each other. At a high level, this theorem provides us with additional insight into these concepts and their interrelation, and it furthermore implies that a tradeoff between these two concepts is unavoidable in designing learning algorithms. On the other hand, given that both sparsity and stability are desirable properties, one interesting direction is to understand the full implications of having one of them. That is, what other properties must a sparse solution have? Given that sparse algorithms often perform well and have strong statistical behavior, one may further ask for other significant notions of stability that are not in conflict with sparsity.

[2]The ℓ_0 minimization is approximated using a Cross Entropy method due its NP-hardness.

9.7. Empirical results

The following empirical results are adapted from [**106**].

TABLE 9.1. The error and SV percentage (in parentheses) with a linear kernel. The number after $\pm$ represents the standard deviation of either the error or the percentage of SVs.

Data Set	soft-margin SVM	1-norm SVM	ℓ_0-SVM
Ionosphere	14.7 ± 2.0 (36.4 ± 8.9)	13.0 ± 1.8 (15.1 ± 2.5)	14.6 ± 1.8 (7.7 ± 2.6)
Pima	24.3 ± 1.4 (51.2 ± 6.2)	24.6 ± 1.1 (4.9 ± 0.5)	24.8 ± 1.5 (3.9 ± 1.2)
Wbc	5.7 ± 1.2 (10.1 ± 2.5)	5.9 ± 1.4 (4.8 ± 1.1)	5.9 ± 0.8 (3.7 ± 1.4)
Bups	32.6 ± 2.1 (71.9 ± 3.8)	32.5 ± 1.7 (4.0 ± 0.0)	33.4 ± 2.8 (3.1 ± 0.6)
Sona	25.9 ± 3.7 (53.7 ± 7.9)	25.5 ± 4.7 (14.7 ± 2.4)	25.5 ± 4.7 (10.3 ± 1.9)

TABLE 9.2. The error and SV percentage with a polynomial kernel of degree 5.

Data Set	soft-margin SVM	1-norm SVM	ℓ_0-SVM
Ionosphere	15.2 ± 2.7 (36.1 ± 3.7)	13.7 ± 2.6 (20.5 ± 8.4)	12.5 ± 1.3 (7.1 ± 1.1)
Pima	33.2 ± 1.5 (48.8 ± 5.2)	30.6 ± 1.8 (29.5 ± 4.6)	30.2 ± 2.4 (11.2 ± 6.6)
Wbc	6.0 ± 2.1 (21.9 ± 2.7)	8.5 ± 2.8 (15.1 ± 3.2)	5.6 ± 1.3 (2.5 ± 0.7)
Bups	33.7 ± 5.2 (58.0 ± 6.0)	36.3 ± 2.2 (33.9 ± 3.5)	37.9 ± 4.4 (14.4 ± 9.9)
Sona	15.9 ± 4.7 (70.3 ± 1.7)	20.3 ± 7.0 (51.1 ± 6.8)	23.3 ± 5.3 (6.9 ± 1.6)

TABLE 9.3. The error and SV percentage with a Gaussian kernel and $C = 1$.

Data Set	soft-margin SVM	1-norm SVM	ℓ_0-SVM
Ionosphere	9.8 ± 2.3 (76.3 ± 2.2)	6.2 ± 1.5 (19.3 ± 3.1)	6.6 ± 2.3 (14.1 ± 2.6)
Pima	27.5 ± 1.7 (67.9 ± 5.1)	25.2 ± 3.0 (12.9 ± 4.9)	25.4 ± 3.3 (8.5 ± 1.7)
Wbc	7.5 ± 0.8 (42.4 ± 3.4)	4.6 ± 1.5 (14.4 ± 1.5)	4.7 ± 1.4 (9.7 ± 1.2)
Bups	34.4 ± 3.0 (93.4 ± 1.6)	36.9 ± 3.9 (28.3 ± 25.5)	36.9 ± 4.6 (10.4 ± 4.3)
Sona	46.7 ± 6.3 (100.0 ± 0.0)	24.3 ± 3.5 (41.7 ± 6.2)	24.5 ± 3.7 (22.5 ± 2.5)

CHAPTER 10

Comprehensive Robust Support Vector Machines and Convex Risk Measures

In Chapters 6- 8 we showed that robustness played an important role in machine learning tasks. In fact, we can actively exploit this relationship by designing robust learning algorithms, which is the main theme of Chapters 10 and 11. In this chapter, we propose a new classification algorithm in the spirit of support vector machines based on robust optimization, that builds in non-conservative protection to noise and controls overfitting. Our formulation is based on a softer version of robust optimization called comprehensive robustness. We show that this formulation is equivalent to regularization by any arbitrary convex regularizer. We explain how the connection of comprehensive robustness to convex risk-measures can be used to design risk-measure constrained classifiers with robustness to the input distribution. The proposed formulation leads to convex optimization problems that can be easily solved. Finally, we provide some empirical results that show the promise of comprehensive robust classifiers. Part of the material of this chapter appears in [**169**].

10.1. Introduction

SVMs are among the most successful algorithms for classification (see for example [**2, 156, 133**]). The standard SVM setup relies on an iid assumption, that is, all

training samples and testing samples are assumed to be independently generated according to an unknown underlying distribution, and finds a hyperplane (in the Reproducing Kernel Hilbert Space) that minimizes some regularized empirical loss.

In this chapter we follow a different approach, proposed originally by [**137, 27, 101**]. The training data are assumed to be generated by the true underlying distribution, but some non-iid (potentially adversarial) disturbance is then added to the samples we observe. Previous works on robust SVMs are all based on a (often too conservative) worst-case analysis, i.e., the training error under the most *adversarial* disturbance realization is considered. This worst-case approach provides a solution with but one guarantee: feasibility and worst-case performance control for *any* realization of the disturbance within the bounded uncertainty set. If the disturbance realization turns out favorable (e.g., close to mean behavior), no improved performance is guaranteed, while if the realization occurs outside the assumed uncertainty set, all bets are off: the error is not controlled. This makes it difficult to address noise with heavy tails: if one takes a small uncertainty set, there is no guarantee for potentially high probability events; on the other hand, if one seeks protection over large uncertainty sets, the robust setting may yield overly pessimistic solutions.

We harness new developments in robust optimization [**64, 12, 22**], in particular the softer notion of "comprehensive robustness" [**10**], and derive a new robust SVM formulation that addresses this problem explicitly. The key idea to comprehensive robustness is to discount lower-probability noise realizations by reducing the loss incurred. This allows us to construct classifiers with improved empirical performance together with probability bounds for *all* magnitudes of constraint violations. In particular, our contributions include the following:

- We use comprehensive robustness to construct "soft robust" classifiers with performance guarantees *that depend on the level of disturbance* affecting the training data – that is, the performance guarantee is noise-level-dependent. This is in contrast to robust classification which provides the same guarantees uniformly inside the uncertainty set, and no guarantees outside. We

show that this richer class of robustness is equivalent to a much broader class of regularizers, including, e.g., standard norm-based SVM and Kullback-Leibler divergence based SVM regularizers. Moreover, we provide computational complexity results for these comprehensive robust classifiers.

- We next show the connection to risk theory [**73, 9**], at the same time extending past work on chance constraints, and also opening the door for constructing classifiers with different risk-based guarantees. Although the connection seems natural, to the best of our knowledge this is the first attempt to view classification from a risk-hedging perspective.
- Lastly, we illustrate the performance of our new classifiers through simulation. In particular we show that the comprehensive robust classifier, which can be viewed as a generalization of the standard SVM and the robust SVM, provides superior empirical results.

Structure of the chapter: This chapter is organized as follows. In Section 10.2 we investigate the comprehensive robust classification framework, particularly a formulation where the loss incurred decreases in an additive way, depending on the disturbance. We discuss a special class of discounts, namely norm discounts, and derive probability bounds for such discounts in Section 10.3. In Section 10.4 we briefly investigate the tractability of the multiplicative discount formulation, i.e., the loss incurred decreases in a multiplicative manner, depending on the disturbance. We relate comprehensive robust classification with convex risk theory in Section 10.5. The kernelized version of comprehensive robust classification is given in Section 10.6. We provide numerical simulation results comparing robust classification and comprehensive robust classification in Section 10.7. Some concluding remarks are given in Section 10.8.

Notation: Capital letters are used to denote matrices, and boldface letters are used to denote column vectors. For a given norm $\|\cdot\|$, we use $\|\cdot\|^*$ to denote its dual norm. Similarly, for a function $f(\cdot)$ defined on a set $\mathcal{H}$, $f^*(\cdot)$ denotes its conjugate

function, i.e., $f^*(\mathbf{y}) = \sup_{\mathbf{x}\in\mathcal{H}}\{\mathbf{y}^\top\mathbf{x} - f(\mathbf{x})\}$. For a vector $\mathbf{x}$ and a positive semi-definite matrix C of the same dimension, $\|\mathbf{x}\|_C$ denotes $\sqrt{\mathbf{x}^\top C\mathbf{x}}$. We use δ to denote disturbance affecting the samples. We use superscript r to denote the true value for an uncertain variable, so that $\boldsymbol{\delta}_i^r$ is the true (but unknown) noise of the i^{th} sample. The set of non-negative scalars is denoted by $\mathbb{R}^+$. The set of integers from 1 to n is denoted by $[1:n]$.

10.2. Comprehensive robust classification

We consider the standard binary-class classification setup, where we are given a finite number of training samples $\{\mathbf{x}_i, y_i\}_{i=1}^m \subseteq \mathbb{R}^n \times \{-1, +1\}$, and must find a linear classifier, specified by the function $h^{\mathbf{w},b}(\mathbf{x}) = \text{sgn}(\langle\mathbf{w},\, \mathbf{x}\rangle + b)$. For a standard regularized classifier, the parameters are obtained by solving the following convex optimization problem:

$$\min_{\mathbf{x},b}\Big\{r(\mathbf{w},b) + \sum_{i=1}^m \big[1 - y_i(\langle\mathbf{w},\, \mathbf{x}_i\rangle + b), 0\big]\Big\},$$

where $r(\mathbf{w}, b)$ is a regularization term. Notice here, $[1 - y_i(\langle\mathbf{w},\, \mathbf{x}_i\rangle + b), 0]$ is the hinge-loss incurred for the i^{th} sample. The standard robust SVM (e.g., [**27, 137**]) considers the case where samples are corrupted by some noise $\vec{\boldsymbol{\delta}} = (\boldsymbol{\delta}_1, \cdots, \boldsymbol{\delta}_m) \in \mathcal{N}$, and solve the following mini-max problem

$$\min_{\mathbf{w},b}\ \max_{(\boldsymbol{\delta}_1,\cdots,\boldsymbol{\delta}_m)\in\mathcal{N}}\Big\{r(\mathbf{w},b) + \sum_{i=1}^m \big[1 - y_i(\langle\mathbf{w},\, \mathbf{x}_i - \boldsymbol{\delta}_i\rangle + b), 0\big]\Big\}. \tag{10.1}$$

The *uncertainty set* $\mathcal{N}$ is called box-typed if $\mathcal{N} = \prod_{i=1}^m \mathcal{N}_i$, where $\mathcal{N}_i$ is the projection of $\mathcal{N}$ onto the i^{th} component. This essentially implies that the disturbances for different observations are uncorrelated, and is an assumption made by virtually all robust SVM works. For box-typed uncertainty set, the robust classifier (10.1) can be

rewritten as

$$\begin{aligned} \min_{\mathbf{w},b}: \quad & r(\mathbf{w},b) + \sum_{i=1}^{m} \xi_i \\ \text{s.t.}: \quad & \xi_i \geq \big[1 - y_i(\langle \mathbf{w}, \mathbf{x}_i - \boldsymbol{\delta}_i \rangle + b)\big], \quad \boldsymbol{\delta}_i \in \mathcal{N}_i, \\ & \xi_i \geq 0. \end{aligned}$$

If we denote the hinge loss of a sample under a certain noise realization as $\xi_i(\boldsymbol{\delta}_i) \triangleq \max\big[1 - y_i(\langle \mathbf{w}, \mathbf{x}_i - \boldsymbol{\delta}_i \rangle + b), 0\big]$, the robust classifier (10.1) can be rewritten as:

$$\min_{\mathbf{w},b} \max_{(\boldsymbol{\delta}_1,\cdots,\boldsymbol{\delta}_m)\in\mathcal{N}} \big\{ r(\mathbf{w},b) + \sum_{i=1}^{m} \xi_i(\boldsymbol{\delta}_i) \big\}.$$

There are two potential problems with this robust classifier. First, it treats all disturbances belonging to $\mathcal{N}$ in exactly the same manner, which can lead to an unfavorable bias to rare disturbances. In fact, it can be shown that replacing $\mathcal{N}$ with its boundary we obtain the same classifier. Second, it provides no protection against disturbances outside $\mathcal{N}$, which makes it inappropriate for disturbances with unbounded support, particularly in the heavy-tailed case.

Instead, we formulate the comprehensive robust classifier by introducing a discounted loss function depending not only on the nominal hinge loss, *but also on the noise realization itself.* Let $h_i(\cdot,\cdot) : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}$ satisfy $0 \leq h_i(\alpha,\beta) \leq h_i(\alpha,\mathbf{0}) = \alpha$. We use h to denote our discounted loss function: it discounts the loss depending on the realized data, yet is always nonnegative, and provides no discount for samples with zero disturbance. Thus, the comprehensive robust classifier is given by:

$$\min_{\mathbf{w},b} \sup_{(\boldsymbol{\delta}_1,\cdots,\boldsymbol{\delta}_m)\in\mathcal{N}} \Big\{ r(\mathbf{w},b) + \sum_{i=1}^{m} h_i\big(\xi_i(\boldsymbol{\delta}_i), \boldsymbol{\delta}_i\big) \Big\}. \tag{10.2}$$

We primarily investigate additive discounts of the form $h_i(\alpha,\boldsymbol{\beta}) \triangleq \max(0, \alpha - f_i(\boldsymbol{\beta}))$ in this chapter, with a short detour to consider multiplicative discounts in Section 10.4. Additive structure provides a rich class of discount functions, while

remaining tractable. Moreover, this additive structure provides the link to risk theory and convex risk measures which we pursue in Section 10.5.

We formulate comprehensive robust classification with an additive discount function in Section 10.2.1 and establish an equivalence relation between comprehensive robust classification and a broad class of regularization schemes in Section 10.2.2. In particular, we show that the standard norm-regularized SVM has a comprehensive robust representation, and so do many regularized SVMs with non-norm regularizers. In Section 10.2.3 we investigate the tractability of comprehensive robust classification.

10.2.1. Problem formulation. We consider box uncertainty sets throughout. Substituting $h_i(\alpha, \boldsymbol{\beta}) \triangleq \max(0, \alpha - f_i(\boldsymbol{\beta}))$ and $\mathcal{N} = \prod_i \mathcal{N}_i$ into Equation (10.2) and extending $f_i(\cdot)$ to take the value $+\infty$ for $\delta_i \notin \mathcal{N}_i$, we obtain a formulation of the comprehensive robust classifier that has uncountably many constraints:

Comprehensive Robust Classifier:

$$\begin{aligned} \min: \quad & r(\mathbf{w}, b) + \sum_{i=1}^{m} \xi_i, \\ \text{s.t.}: \quad & y_i(\langle \mathbf{w}, \mathbf{x}_i - \boldsymbol{\delta}_i \rangle + b) \geq 1 - \xi_i - f_i(\boldsymbol{\delta}_i), \quad \forall \boldsymbol{\delta}_i \in \mathbb{R}^n,\ i = 1, \cdots, m \\ & \xi_i \geq 0; \qquad i = 1, \cdots, m. \end{aligned} \tag{10.3}$$

This $f_i(\cdot)$ (extended real) function controls the disturbance discount, and therefore must satisfy

$$\inf_{\boldsymbol{\beta} \in \mathbb{R}^n} f_i(\boldsymbol{\beta}) = f_i(\mathbf{0}) = 0. \tag{10.4}$$

Notice that if we set $f_i(\cdot)$ to be the indicator function of a set, we recover the standard robust classifier formulation. Thus the comprehensive robust classifier is a natural generalization of the robust classifier with more flexibility on setting $f_i(\cdot)$.

The function $f_i(\cdot)$ has a physical interpretation as controlling the margin of the resulting classifier under *all disturbances*. That is, when $\xi_i = 0$, the resulting classifier guarantees a margin $1/\|\mathbf{w}\|$ for the observed sample $\mathbf{x}_i$ (the same as the standard

classifier), together with a guaranteed margin $(1 - f_i(\boldsymbol{\delta}_i))/\|\mathbf{w}\|$ when the sample is perturbed by $\boldsymbol{\delta}_i$.

10.2.2. Comprehensive robustness and regularization. In this section we show that any convex regularization term in the constraint is equivalent to a comprehensive robust formulation, and vice versa. Moreover, the standard regularized SVM is equivalent to a (non-regularized) comprehensive robust classifier where $f_i(\boldsymbol{\delta}_i) = \alpha\|\boldsymbol{\delta}_i\|$.

Given a function $f(\cdot)$, let f^* denote its Legendre-Fenchel transform or conjugate function, given by $f^*(s) = \sup_x\{\langle s, y\rangle - f(x)\}$ (see, e.g., [**126**] for details). Then we have the following, that shows that if f is a disturbance discount that satisfies (10.4), then so does its conjugate, and vice versa. We use this below to establish the equivalence between convex regularization and comprehensive robustness.

LEMMA 10.1. *(i) If $f(\cdot)$ satisfies (10.4), then so does $f^*(\cdot)$.*
(ii) If $g(\cdot)$ is closed and convex, and $g^*(\cdot)$ satisfies (10.4), then so does $g(\cdot)$.

PROOF. (i) By definition we have $f^*(\mathbf{y}) \geq \mathbf{y}^\top\mathbf{0} - f(\mathbf{0}),\ \forall\mathbf{y} \in \mathbb{R}^n$. Hence $\inf_{\mathbf{y}\in\mathbf{R}^n} f^*(\mathbf{y}) \geq 0$, since $f(\mathbf{0}) = 0$. Furthermore, $f^*(\mathbf{0}) = \sup_{\mathbf{x}\in\mathbb{R}^n}(\mathbf{0}^\top\mathbf{x} - f(\mathbf{x})) = -\inf_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}) = 0$ completes the proof of the first part.
(ii) For $g(\cdot)$ closed and convex, $g(\cdot) = (g(\cdot)^*)^*$ [**126, 33**]. The second part follows from the first part by setting $f(\cdot) = g^*(\cdot)$. □

THEOREM 10.2. *The Comprehensive Robust Classifier (10.3) is equivalent to the following convex program:*

$$\begin{aligned} \min: \quad & r(\mathbf{w}, b) + \sum_{i=1}^{m} \xi_i, \\ \text{s.t.}: \quad & y_i(\langle \mathbf{w},\, \mathbf{x}_i\rangle + b) - f_i^*(y_i\mathbf{w}) \geq 1 - \xi_i,\ \ i = 1, \cdots, m, \\ & \xi_i \geq 0,\ \ i = 1, \cdots, m. \end{aligned} \tag{10.5}$$

PROOF. Simple algebra yields

$$\begin{aligned}
& y_i(\langle \mathbf{w},\, \mathbf{x}_i - \boldsymbol{\delta}_i \rangle + b) \geq 1 - \xi_i - f_i(\boldsymbol{\delta}_i),\ \forall \boldsymbol{\delta}_i \in \mathbb{R}^n \\
\iff\quad & y_i(\langle \mathbf{w},\, \mathbf{x}_i \rangle + b) - y_i \mathbf{w}^\top \boldsymbol{\delta}_i + f_i(\boldsymbol{\delta}_i) \geq 1 - \xi_i,\ \forall \boldsymbol{\delta}_i \in \mathbb{R}^n \\
\iff\quad & y_i(\langle \mathbf{w},\, \mathbf{x}_i \rangle + b) - \sup_{\boldsymbol{\delta}_i \in \mathbb{R}^n} \left[y_i \mathbf{w}^\top \boldsymbol{\delta}_i - f_i(\boldsymbol{\delta}_i) \right] \geq 1 - \xi_i \\
\iff\quad & y_i(\langle \mathbf{w},\, \mathbf{x}_i \rangle + b) - f_i^*(y_i \mathbf{w}) \geq 1 - \xi_i.
\end{aligned}$$

Finally, note that the problem convexity follows immediately from the (generic) convexity of the conjugate function. □

Theorem 10.2 has two implications. First, it gives an equivalent and finite representation for the infinite program of the Comprehensive robust classifier. Second, the robustness for a given regularizer $f^*(\cdot)$ can be obtained by investigating the corresponding discount function $f(\cdot)$.

From Lemma 10.1(i),

$$\inf_{\mathbf{w} \in \mathbb{R}^n} f_i^*(y_i \mathbf{w}) = f_i^*(\mathbf{0}) = 0,$$

and therefore $f_i^*(\cdot)$ "penalizes" $y_i \mathbf{w}$ and is thus a regularization term. A classifier that has a convex regularization term $g(\cdot)$ in each constraint is equivalent to a comprehensive robust classifier with disturbance discount $f(\cdot) = g^*(\cdot)$ (Lemma 10.1(ii)). Therefore, the comprehensive robust classifier is equivalent to the constraint-wise regularized classifier with general convex regularization. This equivalence gives an alternative explanation for the generalization ability of regularization: intuitively, the set of testing data can be regarded as a "disturbed" copy of the set of training samples where the penalty on large (or low-probability) disturbance is discounted. Empirical results show that a classifier that handles noise well has a good performance for testing samples.

As an example of this equivalence, set $f_i(\boldsymbol{\delta}_i) = \alpha\|\boldsymbol{\delta}_i\|$ for $\alpha > 0$ and $r(\mathbf{w}, b) \equiv 0$. Here,

$$f_i^*(y_i\mathbf{w}) = \begin{cases} 0 & \|\mathbf{w}\|^* \leq \alpha, \\ +\infty & \text{otherwise;} \end{cases}$$

which is the indicator function of the dual-norm ball with radius α. Thus (10.5) is equivalent to

$$\begin{array}{rl} \min: & \sum_{i=1}^m \xi_i, \\ \text{s.t.}: & y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) \geq 1 - \xi_i, \;\; i = 1, \cdots, m, \\ & \|\mathbf{w}\|^* \leq \alpha, \\ & \xi_i \geq 0, \;\; i = 1, \cdots, m. \end{array} \tag{10.6}$$

We notice that Problem (10.6) is the standard regularized classifier. Hence, the comprehensive robust classification framework is a general framework which includes both robust SVMs and regularized SVMs as special cases. Hence, the results obtained for the comprehensive robust classifier (e.g., the probabilistic bound in Section 10.3) can be easily applied to robust SVMs and standard SVMs.

10.2.3. Tractability. We now give a sufficient condition on the discount, so that the resulting comprehensive robust classification problem (10.5) is computationally tractable.

DEFINITION 10.1. *A function $f(\cdot) : \mathbb{R}^n \to \mathbb{R}$ is called* Efficiently Conjugatable *if there exists a sub-routine such that for arbitrary $\mathbf{h} \in \mathbb{R}^n$ and $\alpha \in \mathbb{R}$, in polynomial time it either reports*

$$\sup_{\mathbf{x} \in \mathbb{R}^n} \left(\mathbf{h}^\top \mathbf{x} - f(\mathbf{x}) \right) \leq \alpha,$$

or reports $\mathbf{x}_0$ such that

$$\mathbf{h}^\top \mathbf{x}_0 - f(\mathbf{x}_0) > \alpha.$$

THEOREM 10.3. *Suppose*

(1) $f_i(\cdot)$ is efficiently conjugatable, $\forall i \in [1 : m]$.

(2) Both $r(\mathbf{w}, b)$ and $\partial r(\mathbf{w}, b)$ can be evaluated in polynomial time $\forall (\mathbf{w}, b) \in \mathbb{R}^{n+1}$, where ∂ stands for any sub-gradient.

Then, Problem (10.5) can be solved in polynomial time.

PROOF. Rewrite Problem (10.5) as

$$\begin{aligned} \min: \quad & t \\ \text{s.t.}: \quad & r(\mathbf{w}, b) + \sum_{i=1}^{m} \xi_i - t \leq 0 \\ & f_i^*(y_i \mathbf{w}) - y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) - \xi_i + 1 \leq 0, \;\; i = 1, \cdots, m, \\ & -\xi_i \leq 0, \;\; i = 1, \cdots, m. \end{aligned} \tag{10.7}$$

This is a special case of $\min_{\mathbf{z} \in \mathcal{U}} \mathbf{c}^\top \mathbf{z}$ for a convex $\mathcal{U}$. It is known [**83**] that for this problem to be efficiently solvable, it suffices to have a "*Separation Oracle*" for $\mathcal{U}$, i.e., a subroutine which in polynomial time reports either $\mathbf{z} \in \mathcal{U}$, or a separating hyperplane of $\mathbf{z}$ and $\mathcal{U}$ when $\mathbf{z} \notin \mathcal{U}$ for any $\mathbf{z}$.

We can construct a separation oracle for $\mathcal{U}$ as long as we can construct a separation oracle for the feasible set of each individual constraint.

Constraint Type 1: $r(\mathbf{w}, b) + \sum_{i=1}^{m} \xi_i - t \leq 0$.

For any $(\mathbf{w}^*, \boldsymbol{\xi}^*, t^*, b^*)$, since $r(\mathbf{w}^*, b^*)$ can be evaluated efficiently, we can report whether this constraint holds or not in polynomial time. Furthermore, when the constraint is violated, any sub-gradient of the left-hand side evaluated at $(\mathbf{w}^*, \boldsymbol{\xi}^*, t^*, b^*)$ is a separating hyperplane. Finding such sub-gradient can also be done efficiently since $\partial r(\mathbf{w}^*, b)$ can be evaluated efficiently.

Constraint Type 2: $f_i^*(y_i \mathbf{w}) - y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) - \xi_i + 1 \leq 0$.

For given $(\mathbf{w}^*, \boldsymbol{\xi}^*, t^*, b^*)$, let $\alpha = y_i(\langle \mathbf{w}^*, \mathbf{x}_i \rangle + b^*) + \xi_i^* - 1$, and $\mathbf{h} = y_i \mathbf{w}^*$. Since $f_i(\cdot)$ is efficiently conjugatable, in polynomial time we can either confirm

$$\sup_{\mathbf{c} \in \mathbb{R}^n} \left(\mathbf{h}^\top \mathbf{c} - f(\mathbf{c}) \right) \leq \alpha,$$

which means the constraint holds, or report a $\mathbf{c}_0$ such that

$$\mathbf{h}^\top \mathbf{c_0} - f(\mathbf{c_0}) > \alpha.$$

Substituting back α, $\mathbf{h}$ and rearranging the terms yields

$$(y_i\mathbf{c}_0 - y_i\mathbf{x}_i)^\top \mathbf{w}^* - y_i b^* - \xi_i^* > f(\mathbf{c}_0) - 1.$$

Notice that, for any feasible $(\hat{\mathbf{w}}, \hat{\boldsymbol{\xi}}, \hat{t}, \hat{b})$, the following holds:

$$\begin{aligned}
& f_i^*(y_i\hat{\mathbf{w}}) - y_i(\langle \hat{\mathbf{w}},\, \mathbf{x}_i\rangle + \hat{b}) - \hat{\xi}_i + 1 \le 0 \\
\Longrightarrow \quad & \sup_{\mathbf{c}\in\mathbb{R}^n} \big(y_i\hat{\mathbf{w}}^\top\mathbf{c} - f(\mathbf{c})\big) - y_i(\langle \hat{\mathbf{w}},\, \mathbf{x}_i\rangle + \hat{b}) - \hat{\xi}_i + 1 \le 0 \\
\Longrightarrow \quad & \big(y_i\hat{\mathbf{w}}^\top\mathbf{c}_0 - f(\mathbf{c}_0)\big) - y_i(\langle \hat{\mathbf{w}},\, \mathbf{x}_i\rangle + \hat{b}) - \hat{\xi}_i + 1 \le 0 \\
\Longrightarrow \quad & (y_i\mathbf{c}_0 - y_i\mathbf{x}_i)^\top\hat{\mathbf{w}} - y_i\hat{b} - \hat{\xi}_i \le f(\mathbf{c}_0) - 1.
\end{aligned}$$

Hence $(y_i\mathbf{c}_0 - y_i\mathbf{x}_i, -y_i, -1)$ is a separation Oracle.

Constraint Type 3: $-\xi_i \le 0$.

The separation oracle for this constraint is trivial.

Combining all three steps, we conclude that a separation oracle exists for each individual constraint, and hence we have a separation Oracle for $\mathcal{U}$. Therefore, Problem (10.5) can be solved in polynomial time. □

This theorem guarantees polynomial time solvability, but much stronger complexity requirements may be needed for large scale problems. While this is a topic of future research, in the nest section we provide some discount function examples that are of practical interest.

10.3. Norm discount

In this section, we discuss a class of discount functions based on certain ellipsoidal norms of the noise, i.e.,

$$f_i(\boldsymbol{\delta}_i) = t_i(\|\boldsymbol{\delta}\|_V),$$

for a nondecreasing $t_i : \mathbb{R}^+ \to \mathbb{R}^+$. Simple algebra yields $f_i^*(\mathbf{y}) = t_i^*(\|\mathbf{y}\|_{V^{-1}})$, where $t_i^*(y) = \sup_{x\ge 0}\big[xy - t(x)\big]$, and thus conjugation is easy. This formulation has two natural probabilistic interpretations: (1) it provides tight bounds on the probability

of *all* magnitude of constraint violations when only the first two moments of the disturbance are known (Theorem 10.4); (2) it explicitly computes the probabilities of *all* magnitude of constraint violations when the disturbance is Gaussian (Theorem 10.5).

THEOREM 10.4. *Suppose the random variable* $\boldsymbol{\delta}_i^r$ *has mean* $\mathbf{0}$ *and variance* Σ. *Then the constraint*

$$y_i(\langle \mathbf{w},\, \mathbf{x}_i - \boldsymbol{\delta}_i \rangle + b) \geq 1 - \xi_i - t_i(\|\boldsymbol{\delta}_i\|_{\Sigma^{-1}}),\ \forall \boldsymbol{\delta}_i \in \mathbb{R}^n, \tag{10.8}$$

is equivalent to

$$\inf_{\boldsymbol{\delta}_i^r \sim (0,\Sigma)} Pr\big(y_i(\langle \mathbf{w},\, \mathbf{x}_i^r \rangle + b) - 1 + \xi_i \geq -s\big) \geq 1 - \frac{1}{\big(t_i^{-1}(s)\big)^2 + 1},\ \forall s \geq 0 \tag{10.9}$$

Here, the infimum is taken over all random variables with mean zero and variance Σ, *and* $t_i^{-1}(s) \triangleq \sup\{r | t(r) \leq x\}$.

PROOF. [**137**] studied the robust formulation and showed that for a fixed γ_0, the following three inequalities are equivalent:

- $\inf_{\boldsymbol{\delta}_i^r \sim (0,\Sigma)} Pr\big(y_i(\langle \mathbf{w},\, \mathbf{x}_i^r \rangle + b) - 1 + \xi_i \geq 0\big) \geq 1 - \frac{1}{\gamma_0^2 + 1}$,
- $y_i(\langle \mathbf{w},\, \mathbf{x}_i \rangle + b) - 1 + \xi_i \geq \gamma_0 \|\mathbf{w}\|_\Sigma$,
- $y_i(\langle \mathbf{w},\, \mathbf{x}_i - \boldsymbol{\delta}_i \rangle + b) - 1 + \xi_i \geq 0, \quad \forall \|\boldsymbol{\delta}_i\|_{\Sigma^{-1}} \leq \gamma_0$.

Observe that Equation (10.9) is equivalent to

$$\inf_{\boldsymbol{\delta}_i^r \sim (0,\Sigma)} Pr\big(y_i(\langle \mathbf{w},\, \mathbf{x}_i^r \rangle + b) - 1 + \xi_i \geq -t_i(\gamma)\big) \geq 1 - \frac{1}{\gamma^2 + 1},\ \forall \gamma \geq 0.$$

Hence, it is equivalent to:

$$y_i(\langle \mathbf{w},\, \mathbf{x}_i - \boldsymbol{\delta}_i \rangle + b) - 1 + \xi_i \geq -t_i(\gamma),\ \forall \|\boldsymbol{\delta}_i\|_{\Sigma^{-1}} \leq \gamma, \quad \forall \gamma \geq 0.$$

Since $t_i(\cdot)$ is nondecreasing, this is equivalent to (10.8). □

Theorem 10.4 shows that the comprehensive robust formulation bounds the probability of *all* magnitudes of constraint violation. It is of interest to compare this bound

with the bound given by the robust formulation. Indeed,

$$\begin{aligned} & y_i(\langle \mathbf{w},\, \mathbf{x}_i\rangle + b) - 1 + \xi_i \geq \gamma_0 \|\mathbf{w}\|_\Sigma \\ \iff \quad & y_i(\langle \mathbf{w},\, \mathbf{x}_i\rangle + b) - 1 + \xi_i + s \geq \big(\gamma_0 + \frac{s}{\|\mathbf{w}\|_\Sigma}\big)\|\mathbf{w}\|_\Sigma,\ \forall s \geq 0 \\ \iff \quad & \inf_{\boldsymbol{\delta}_i^r \sim (0,\,\Sigma)} Pr\big(y_i(\langle \mathbf{w},\, \mathbf{x}_i^r\rangle + b) - 1 + \xi_i \geq -s\big) \geq 1 - \frac{1}{(\gamma_0 + \frac{s}{\|\mathbf{w}\|_\Sigma})^2 + 1}. \end{aligned}$$

Hence the probability of large violation depends on $\|\mathbf{w}\|_\Sigma$, and is impossible to bound without knowing $\|\mathbf{w}\|_\Sigma$ *a priori*.

REMARK 10.1. Notice the derived bound for the robust formulation is tight, in the sense that if

$$y_i(\langle \mathbf{w},\, \mathbf{x}_i\rangle + b) - 1 + \xi_i < \gamma_0 \|\mathbf{w}\|_\Sigma,$$

then there exists a zero-mean random variable $\boldsymbol{\delta}_i^r$ with variance Σ such that

$$Pr\big(y_i(\langle \mathbf{w},\, \mathbf{x}_i^r\rangle + b) - 1 + \xi_i \geq -s\big) < 1 - \frac{1}{(\gamma_0 + \frac{s}{\|\mathbf{w}\|_\Sigma})^2 + 1}.$$

This is because the multivariate Chebyshev inequality [**109, 33, 83**] states that

$$\begin{aligned} & \sup_{\mathbf{z} \sim (\bar{\mathbf{z}}, \sigma)} Pr\{\mathbf{a}^\top \mathbf{z} \leq c\} = (1 + d^2)^{-1} \\ \text{where} \quad & d^2 = \inf_{\mathbf{z}_0 | \mathbf{a}^\top \mathbf{z}_0 \leq c} \inf (\mathbf{z}_0 - \bar{\mathbf{z}})^\top \Sigma^{-1} (\mathbf{z}_0 - \bar{\mathbf{z}}). \end{aligned}$$

Here $\mathbf{z} \sim (\bar{\mathbf{z}}, \sigma)$ stands for $\mathbf{z}$ is a random variable with mean $\bar{\mathbf{z}}$ and variance σ. Letting $\mathbf{a} = y_i \mathbf{w}$, $\mathbf{z} = -\boldsymbol{\delta}_i^r$ and $c = 1 - \xi_i - s - y_i(\langle \mathbf{w},\, \mathbf{x}_i\rangle + b)$, we have

$$\begin{aligned} & \sup_{\boldsymbol{\delta}_i^r \sim (0,\,\Sigma)} Pr\big(y_i(\langle \mathbf{w},\, \mathbf{x}_i^r\rangle + b) - 1 + \xi_i \leq -s\big) = (1 + d_0^2)^{-1} \\ \text{where:} \quad & d_0 = \frac{y_i(\langle \mathbf{w},\, \mathbf{x}_i\rangle + b) - 1 + \xi_i + s}{\sqrt{\mathbf{w}^\top \Sigma \mathbf{w}}}. \end{aligned}$$

Hence,

$$\begin{aligned} & y_i(\langle \mathbf{w},\, \mathbf{x}_i\rangle + b) - 1 + \xi_i < \gamma_0 \|\mathbf{w}\|_\Sigma \\ \Longrightarrow \quad & d_0 < \gamma_0 + s/\|\mathbf{w}\|_\Sigma \\ \Longrightarrow \quad & \sup_{\boldsymbol{\delta}_i^r \sim (0,\,\Sigma)} Pr\big(y_i(\langle \mathbf{w},\, \mathbf{x}_i^r\rangle + b) - 1 + \xi_i \le -s\big) > \big[1 + (\gamma_0 + s/\|\mathbf{w}\|_\Sigma)^2\big]^{-1}, \end{aligned}$$

showing that the bound is tight.

With a similar argument, we can derive probability bounds under a Gaussian noise assumption.

THEOREM 10.5. *If $\boldsymbol{\delta}_i^r \sim \mathcal{N}(0, \Sigma)$, then the constraint*

$$y_i(\langle \mathbf{w},\, \mathbf{x}_i - \boldsymbol{\delta}_i\rangle + b) \ge 1 - \xi_i - t_i(\|\boldsymbol{\delta}_i\|_{\Sigma^{-1}}),\ \forall \boldsymbol{\delta}_i \in \mathbb{R}^n, \tag{10.10}$$

is equivalent to

$$Pr\big(y_i(\langle \mathbf{w},\, \mathbf{x}_i^r\rangle + b) - 1 + \xi_i \ge -s\big) \ge \Phi\big(t_i^{-1}(s)\big), \qquad \forall s \ge 0. \tag{10.11}$$

Here, $\Phi(\cdot)$ is the cumulative distribution function of $\mathcal{N}(0, 1)$.

PROOF. For fixed $k \ge 1/2$ and constant l, the following constraints are equivalent:

$$\begin{aligned} & Pr(y_i \mathbf{w}^\top \boldsymbol{\delta}_i^r \ge l) \ge k \\ \Longleftrightarrow \quad & l \le \Phi^{-1}(k)\big(\mathbf{w}^\top \Sigma \mathbf{w}\big)^{1/2} \\ \Longleftrightarrow \quad & l \le y \mathbf{w}^\top \boldsymbol{\delta}_i,\ \forall \|\boldsymbol{\delta}_i\|_{\Sigma^{-1}} \le \Phi^{-1}(k). \end{aligned}$$

Notice that (10.11) is equivalent to

$$Pr\Big(y_i(\langle \mathbf{w},\, \mathbf{x}_i^r\rangle + b) - 1 + \xi_i \ge -t_i(\gamma)\Big) \ge \Phi(\gamma),\ \forall \gamma \ge 0,$$

and hence it is equivalent to: $\forall \gamma \ge 0$,

$$y_i(\langle \mathbf{w},\, \mathbf{x}_i - \boldsymbol{\delta}_i\rangle + b) - 1 + \xi_i \ge -t_i(\gamma), \quad \forall \|\boldsymbol{\delta}_i\|_{\Sigma^{-1}} \le \Phi^{-1}\big(\Phi(\gamma)\big) = \gamma.$$

	Original Function	Conjugate function
Affine Fun.	$t_i(x) = ax + b$	$t_i^*(y) = I_\alpha - b$
Indicate Fun.	$t_i(x) = I_\alpha + b.$	$t_i^*(y) = ay - b$
Power Fun.	$t_i(x) = ax^n + b$	$t_i^*(y) = a^{\frac{-1}{n-1}}(n^{\frac{-1}{n-1}} - n^{\frac{-n}{n-1}})y^{\frac{n}{n-1}} - b$
Quadratic Fun.	$t_i(x) = ax^2 + b$	$t_i^*(y) = \frac{1}{4a}y^2 - b$
Neg. Entropy	$t_i(x) = ax\log x + b$	$t_i^*(y) = ae^{y/a-1} - b$
Exponential Fun.	$t_i(x) = ae^x + b$	$t_i^*(y) = y\log(y/a) - y - b$
Point-wise Min.	$t_i(x) = \min_{j=1,\cdots,l} t_{ij}(x)$	$t_i^*(x) = \max_{j=1,\cdots,l} t_{ij}^*(y)$
Non-convex Fun.	$t_i(x)$ non-convex	$t_i^*(y) = \big(conv(t_i(x))\big)^*(y)$

TABLE 10.1. Some functions and their conjugates.

Since $t_i(\cdot)$ is nondecreasing, this is equivalent to (10.10). □

We list in Tables 10.1 and 10.2 some examples of $t_i(\cdot)$ and their conjugate functions. Notice that both $t_i(\cdot)$ and $t_i^*(\cdot)$ are defined on $\mathbb{R}^+$. Here, $I_\alpha : \mathbb{R}^+ \to \mathbb{R}^+ \bigcup\{+\infty\}$ is the indicator function of set α, and $conv(t(\cdot)) \triangleq \sup\{f(\cdot)|f(\cdot)$ is convex, $f(\cdot) \leq t(\cdot)\}$. Standard robustness uses an indicator function of a set. Table 10.2 shows several different relaxations of this indicator function allowing the increase of $f(\cdot)$ to be more smooth.

Notice that, all conjugate functions can be written as $t^*(x) = \max_{1,2}(s_1(x), s_2(x))$, where $s_i = \inf_{\boldsymbol{\lambda}\in\mathcal{S}_i} q_i(\lambda, x)$ for some "simple" functions q_i and polytope $\mathcal{S}_i$. Here by "simple" we mean the function is a quadratic function, or a linear function, or an indicator function. Hence the constraint $t_i^*(x) \leq \alpha$ is equivalent to

$$q_1(x, \boldsymbol{\lambda}_1) \leq \alpha;$$

$$\boldsymbol{\lambda}_1 \in \mathcal{S}_1;$$

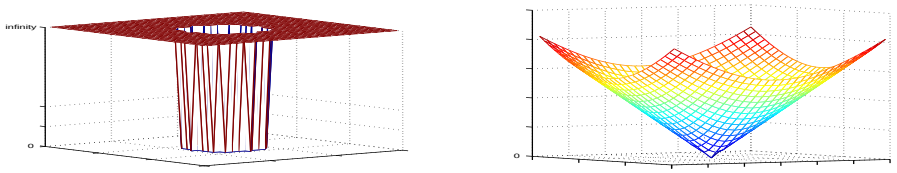
$$q_2(x, \boldsymbol{\lambda}_2) \leq \alpha;$$

$$\boldsymbol{\lambda}_2 \in \mathcal{S}_2.$$

Since a "simple" function leads to a Second Order Cone constraint, the resulting classifier is a SOCP. This means that the comprehensive robust classification with the relaxations listed above has a comparable computational cost to robust classification.

Original Function	Conjugate function
$t_i(x) = \begin{cases} 0 & x \le c, \\ \alpha(x-c) & x > c. \end{cases}$	$t_i^*(y) = \begin{cases} cy & y \le \alpha, \\ +\infty & y > \alpha. \end{cases}$ $= \max(I_\alpha,\, cy)$
$t_i(x) = \begin{cases} \alpha x & x \le c, \\ +\infty & x > c. \end{cases}$	$t_i^*(y) = \begin{cases} 0 & y \le \alpha, \\ c(y-\alpha) & y > \alpha. \end{cases}$ $= \max(0,\, c(y-\alpha))$
$t_i(x) = \begin{cases} 0 & x \le c_1, \\ \alpha(x-c_1) & c_1 < x \le c_2, \\ +\infty & x > c_2. \end{cases}$	$t_i^*(y) = \begin{cases} c_1 y & y \le \alpha, \\ c_2(y-\alpha)+\alpha c_1 & y > \alpha. \end{cases}$ $= \max(c_1 y,\, c_2 y + \alpha(c_1 - c_2))$
$t_i(x) = \begin{cases} 0 & x \le c, \\ \alpha(x-c)^2 & x > c. \end{cases}$	$t_i^*(y) = y^2/4\alpha + cy.$
$t_i(x) = \begin{cases} \alpha x^2 & x \le c, \\ +\infty & x > c. \end{cases}$	$t_i^*(y) = \begin{cases} y^2/4\alpha & y \le 2\alpha c, \\ cy - \alpha c^2 & y > 2\alpha c. \end{cases}$ $= \inf_{\lambda \ge 0} \left((y-\lambda)^2/4\alpha + c\lambda \right)$
$t_i(x) = \begin{cases} 0 & x \le c_1, \\ \alpha(x-c_1)^2 & c_1 < x \le c_2, \\ +\infty & x > c_2. \end{cases}$	$t_i^*(y) = \begin{cases} y^2/4\alpha + yc_1 & y \le 2\alpha(c_2-c_1), \\ c_2 y - \alpha(c_2-c_1)^2 & y > 2\alpha(c_2-c_1). \end{cases}$ $= \max\Big(c_1 y,\, \inf_{\lambda_1,\lambda_2 \ge 0} [\frac{(y+\lambda_1-\lambda_2)^2}{4\alpha}$ $+c_1 y + (c_2-c_1)\lambda_2] \Big)$

TABLE 10.2. Piecewise-defined functions and their conjugates.



(r) (r')

FIGURE 10.1. The robust discount function and its conjugate: note that the discount function provides uniform protection inside the uncertainty set, and no protection outside.

Figure 10.1 illustrates the discount function for the standard robust formulation, and Figure 10.2 illustrates the respective conjugate functions for the first four relaxations in Table 10.2.

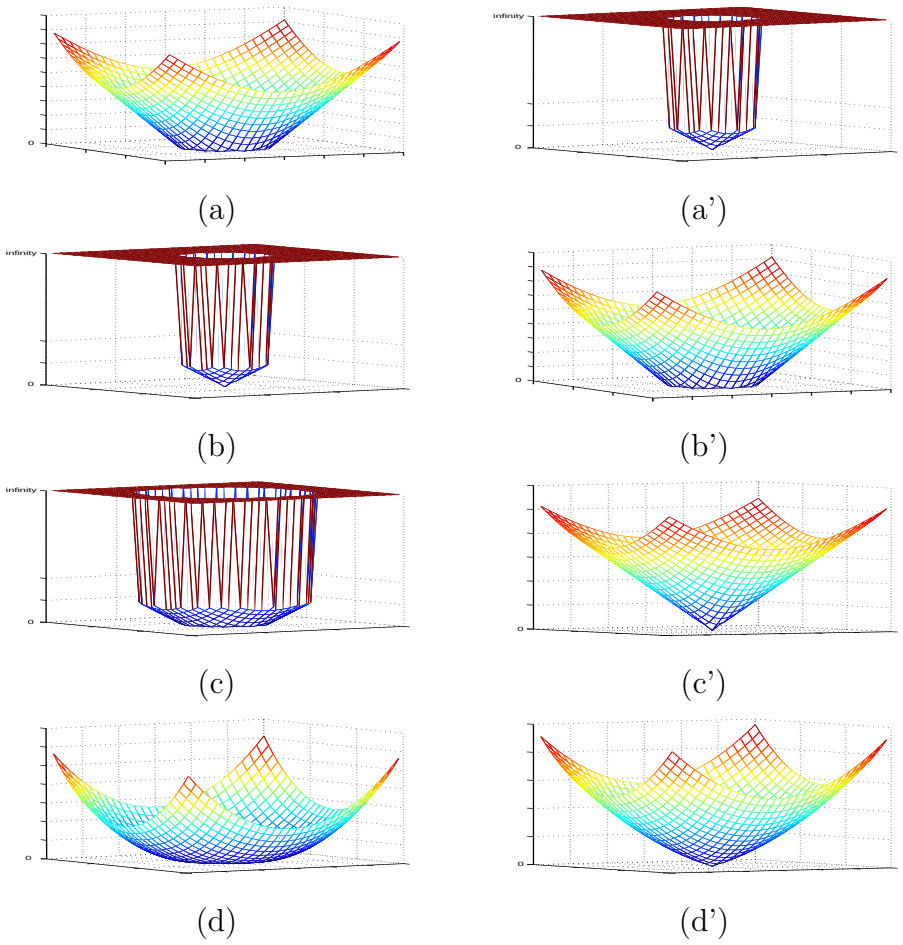


FIGURE 10.2. Piecewise-defined Functions (first four functions in Table 10.2) and their Conjugates: note the flexibility in controlling the discount given the realization of the disturbance.

10.4. Multiplicative discount

In this section we consider a multiplicative structure for the disturbance discount, and investigate its tractability. To multiply a random function with certain values based on the probability of realizations seems to be a very natural way to reduce the effect of rare event, and indeed it has the following probabilistic interpretation as a method to bound the expected loss.

THEOREM 10.6. *Consider a non-negative function $L(\cdot) : \mathbb{R}^n \to \mathbb{R}^+$, such that there exists a non-negative function $\alpha(\cdot)$ and constant c satisfying*

$$L(\mathbf{x})\alpha(\mathbf{x}) \leq c\,; \quad \forall \mathbf{x} \in \mathbb{R}^n.$$

Further assume that a random variable $\hat{\mathbf{x}}$ has a density $f(\cdot)$ satisfying $\alpha(\mathbf{x}) = 0 \Rightarrow f(\mathbf{x}) = 0$. Then we have

$$\mathbb{E}\big\{L(\hat{\mathbf{x}})\big\} \leq c \int_{\alpha(\mathbf{x})\neq 0} \frac{f(\mathbf{x})}{\alpha(\mathbf{x})} d\mathbf{x}.$$

PROOF. Notice that

$$\begin{aligned}\mathbb{E}\big\{L(\hat{\mathbf{x}})\big\} &= \int_{f(\mathbf{x})\neq 0} L(\mathbf{x}) f(\mathbf{x}) d\mathbf{x} \\ &= \int_{\alpha(\mathbf{x})\neq 0} L(\mathbf{x})\alpha(x)\frac{f(\mathbf{x})}{\alpha(\mathbf{x})} d\mathbf{x} \leq \int_{\alpha(\mathbf{x})\neq 0} c\frac{f(\mathbf{x})}{\alpha(\mathbf{x})} d\mathbf{x} = c\int_{\alpha(\mathbf{x})\neq 0} \frac{f(\mathbf{x})}{\alpha(\mathbf{x})} d\mathbf{x}.\end{aligned}$$

□

The Comprehensive robust classifier with multiplicative discount has the form:

$$\min_{\mathbf{w},b} \max_{(\boldsymbol{\delta}_1,\cdots\boldsymbol{\delta}_m)\in\mathcal{N}} \quad \left\{ r(\mathbf{w},b) + \sum_{i=1}^m c_i(\boldsymbol{\delta}_i) \max\big[1 - y_i(\langle \mathbf{w},\, \mathbf{x}_i - \boldsymbol{\delta}_i\rangle + b),\, 0\big] \right\}$$

where $c(\cdot) : \mathbb{R}^n \to \mathbb{R}$ satisfies

$$0 \leq c_i(\boldsymbol{\delta}) \leq c_i(\mathbf{0}) = 1; \quad \forall \boldsymbol{\delta} \in \mathbb{R}^n.$$

By adding slack variables, we get the following optimization problem:

Comprehensive Robust Classifier (Multiplicative):

$$\begin{array}{ll} \min: & r(\mathbf{w},b) + \sum_{i=1}^m \xi_i, \\ \text{s.t.}: & \xi_i \geq c_i(\boldsymbol{\delta})\big[1 - y_i(\langle \mathbf{w},\, \mathbf{x}_i - \boldsymbol{\delta}_i\rangle + b)\big], \qquad \forall \boldsymbol{\delta}_i \in \mathbb{R}^n,\ i = 1,\cdots,m, \\ & \xi_i \geq 0, \quad i = 1,\cdots,m. \end{array} \tag{10.12}$$

Define

$$g_i(\boldsymbol{\delta}) \triangleq \begin{cases} \frac{1}{c_i(\boldsymbol{\delta})} & \text{if } c(\boldsymbol{\delta}) > 0, \\ +\infty & \text{otherwise.} \end{cases}$$

Problem (10.12) can be rewritten as:

$$\begin{aligned}
\min: \quad & r(\mathbf{w}, b) + \textstyle\sum_{i=1}^{m} \xi_i, \\
\text{s.t.}: \quad & g_i(\boldsymbol{\delta}_i)\xi_i \geq \big[1 - y_i(\langle \mathbf{w},\, \mathbf{x}_i - \boldsymbol{\delta}_i \rangle + b)\big], \qquad \forall \boldsymbol{\delta}_i \in \mathbb{R}^n,\ i = 1, \cdots, m, \\
& \xi_i \geq \epsilon, \quad i = 1, \cdots, m.
\end{aligned}$$

We perturb the constraint $\xi_i \geq 0$ to $\xi_i \geq \epsilon$ for small $\epsilon > 0$ to avoid the case that both $\xi_i = 0$ and $g_i(\boldsymbol{\delta}_i) = \infty$ hold simultaneously. Under this modification, we have the following tractability theorem:

THEOREM 10.7. *Suppose*

(1) $g_i(\cdot)$ is efficiently conjugatable, $\forall i \in [1 : m]$

(2) Both $r(\mathbf{w}, b)$, $\partial r(\mathbf{w}, b)$ can be evaluated in polynomial time $\forall (\mathbf{w}, b) \in \mathbb{R}^{n+1}$, where ∂ stands for any sub-gradient.

Then, Problem (10.12) can be solved in polynomial time.

PROOF. Rewrite Problem (10.12) as

$$\begin{aligned}
\min: \quad & t \\
\text{s.t.}: \quad & r(\mathbf{w}, b) + \textstyle\sum_{i=1}^{m} \xi_i - t \leq 0 \\
& 1 - y_i(\langle \mathbf{w},\, \mathbf{x}_i \rangle + b) + y_i \mathbf{w}^\top \boldsymbol{\delta}_i - \xi_i g_i(\boldsymbol{\delta}_i) \leq 0, \qquad \forall \boldsymbol{\delta}_i \in \mathbb{R}^n,\ i = 1, \cdots, m, \\
& -\xi_i \leq -\epsilon, \quad i = 1, \cdots, m.
\end{aligned}$$

Following a similar argument as in the proof of Theorem 10.3 we derive a separation oracle for each constraint. Constraint Type 1 and Type 3 are exactly the same as in Theorem 10.3, hence we only discuss Constraint Type 2, i.e.,

$$1 - y_i(\langle \mathbf{w},\, \mathbf{x}_i \rangle + b) + y_i \mathbf{w}^\top \boldsymbol{\delta}_i - \xi_i g_i(\boldsymbol{\delta}_i) \leq 0,\ \forall \boldsymbol{\delta}_i.$$

Now suppose we are given a solution $(\mathbf{w}^*, \boldsymbol{\xi}^*, t^*, b^*)$, with $\xi_i^* \geq \epsilon$ (otherwise we get a separation oracle from Type 3). Letting $\mathbf{h} = y_i \mathbf{w}^* / \xi_i^*$ and $\alpha = (1 - y_i(\langle \mathbf{w}^*,\, \mathbf{x}_i \rangle + b^*))/\xi_i^*$, the constraint is equivalent to:

$$\sup_{\boldsymbol{\delta}_i \in \mathbb{R}^n} \big\{ \mathbf{h}^\top \boldsymbol{\delta}_i - g_i(\boldsymbol{\delta}_i) \big\} \leq \alpha.$$

Since $g_i(\cdot)$ is efficiently conjugatable, then in polynomial time we either conclude the constraint is satisfied, or find a $\boldsymbol{\delta}^*$ such that $\mathbf{h}^\top \boldsymbol{\delta}^* - g_i(\boldsymbol{\delta}^*) > \alpha$, which is equivalent to

$$\begin{aligned} & 1 - y_i(\langle \mathbf{w}^*, \mathbf{x}_i \rangle + b^*) + y_i \mathbf{w}^{*\top} \boldsymbol{\delta}^* - \xi_i^* g_i(\boldsymbol{\delta}^*) > 0 \\ \iff \quad & y(\mathbf{x}_i - \boldsymbol{\delta}^*)^\top \mathbf{w}^* + y_i b^* + g_i(\boldsymbol{\delta}^*)\xi_i < 1. \end{aligned} \tag{10.13}$$

Hence, $(\mathbf{x}_i - \boldsymbol{\delta}^*, y_i, g_i(\boldsymbol{\delta}^*))$ is a separation oracle. □

10.5. Comprehensive robustness and convex risk Measures

In this section we investigate the relationship between comprehensive robustness and convex risk measures, a notion adapted form decision theory. A risk measure is a mapping from a random variable to the real numbers, that, at a high level, captures some valuation of that random variable. Simple examples of risk measures include expectation, standard deviation, and conditional value-at-risk (CVaR). Risk measure constraints represent a natural way to express risk aversion, corresponding to particular risk preferences.

In Section 10.5.1, we briefly recall the notion of a convex risk measure, formulate classifiers based on risk-measure constraints and show that they are equivalent to comprehensive robust classifiers. In Section 10.5.2, we give examples of tractable Risk-measure constrained classifiers.

10.5.1. Convex risk measure and risk-measure constrained classifier.

The theory of (convex) risk[1] measures was developed in response to the observation that the preference of a decision maker among random losses (aka gambles) can be quite complicated. Still, under mild conditions, it can be proved that for any gamble, there exists a constant such that the decision maker will feel indifferent between the gamble and the constant. Therefore, the preference between the random losses is converted to comparing the respective constants. To be more precise, given a

[1]This is a term used in decision theory to represent a random loss, which is different from what is often used in machine learning literature, i.e., a certain loss of the classifier.

probability space $(\Omega, \mathcal{F}, \mathbb{P})$, let $\mathcal{X}$ denote the set of random variables on Ω. Each elements of $\mathcal{X}$ represents an uncertain *loss*. We have the following definition.

DEFINITION 10.2. *A* risk measure *is a function* $\rho : \mathcal{X} \to \mathbb{R}$.

A risk measure essentially defines a preference relationship among random variables: X_1 is preferable over X_2 if and only if $\rho(X_1) \leq \rho(X_2)$. Alternatively, we can regard $\rho(\cdot)$ as the measurement of how risky a random variable is: X_1 is a less risky decision than X_2 when $\rho(X_1) \leq \rho(X_2)$.

DEFINITION 10.3. *A risk measure is called* convex *if it satisfies the following three conditions:*

(1) Convexity: $\rho(\lambda X + (1-\lambda)Y) \leq \lambda\rho(X) + (1-\lambda)\rho(Y)$*;*

(2) Monotonicity: $X \leq Y \Rightarrow \rho(X) \leq \rho(Y)$*;*

(3) Translation Invariance: $\rho(X + a) = \rho(X) + a, \forall a \in \mathbb{R}$.

In words, Convexity means that diversification reduces risk. Monotonicity says that if one random loss is always less than another, the first is preferable. Translation invariance says that if a fixed penalty a is going to be paid in addition to X, we are indifferent to whether we will pay it before or after X is realized. These properties are intuitively appealing when considering risk-hedging.

A convex risk measure $\rho(\cdot)$ is called *normalized* if it satisfies $\rho(0) = 0$ and $\forall X \in \mathcal{X}$, $\rho(X) \geq \mathbb{E}_{\mathbb{P}}(X)$, which essentially says that the risk measure $\rho(\cdot)$ represents risk aversion. Many widely used criteria comparing random variables are normalized convex risk measures, including expected value, Conditional Value at Risk (CVaR), and the exponential loss function [**10, 17**].

Equipped with a normalized convex risk measure $\rho(\cdot)$, we can formulate a classification problem as follows:

Risk-Measure Constrained Classifier

$$\begin{aligned} \min: \quad & r(\mathbf{w}, b) + \sum_{i=1}^{m} \xi_i, \\ \text{s.t.}: \quad & \rho_i(\xi_i) \geq \rho_i(1 - y_i(\langle \mathbf{w},\, \mathbf{x}_i^r \rangle + b)), \;\; i = 1, \cdots, m, \\ & \xi_i \geq 0, \;\; i = 1, \cdots, m. \end{aligned} \tag{10.14}$$

Notice that $\mathbf{x}_i^r$ is a random variable, hence $1 - y_i(\langle \mathbf{w},\, \mathbf{x}_i^r \rangle + b)$ is a random loss, and ξ_i is the constant "equivalent" to this random loss.

Substituting $\rho_i(0) = 0$ and $\mathbf{x}_i^r = \mathbf{x}_i - \boldsymbol{\delta}_i^r$ where $\mathbf{x}_i = \mathbb{E}_{\mathbb{P}}(\mathbf{x}_i^r)$, the constraint can be rewritten as

$$\xi_i \geq 1 - y_i(\langle \mathbf{w},\, \mathbf{x}_i \rangle + b) + \rho_i(y_i \mathbf{w}^\top \boldsymbol{\delta}_i^r). \tag{10.15}$$

This formulation seeks a classifier whose total risk is minimized. When $\mathbf{x}_i^r$ is precisely known, this formulation reduces to the standard SVM.

The following theorem states that the risk-constrained classifier and the comprehensive robust classifier are equivalent. The proof is postponed to the Appendix.

THEOREM 10.8. *(1) A Risk-Measure Constrained Classifier with normalized convex risk measures $\rho_i(\cdot)$ is equivalent to a Comprehensive Robust Classifier where*

$$\begin{aligned} f_i(\boldsymbol{\delta}) &= \inf\{\alpha_i^0(Q) | \mathbb{E}_Q(\boldsymbol{\delta}_i^r) = \boldsymbol{\delta}\}, \\ \alpha_i^0(Q) &\triangleq \sup_{X' \in \mathcal{X}} \big(\mathbb{E}_Q(X') - \rho_i(X')\big). \end{aligned}$$

(2) A Comprehensive Robust Classifier with convex discount functions $f_i(\cdot)$ is equivalent to a Risk-Constrained Classifier where

$$\begin{aligned} \rho_i(X) &= \inf\{m \in \mathbb{R} | X - m \in \mathcal{A}_i\}, \\ \mathcal{A}_i &\triangleq \{X \in \mathcal{X} | X(\omega) \leq f_i\big(\boldsymbol{\delta}_i^r(\omega)\big), \; \forall \omega \in \Omega\}, \end{aligned}$$

assuming that $\boldsymbol{\delta}_i^r$ has support $\mathbb{R}^n$.

PROOF. Before proving Theorem 10.8, we establish the following two lemmas. Lemma 10.9 is adapted from [**73**], and the reader can find the proof there.

LEMMA 10.9. *Let $\mathcal{X}$ be the set of random variables for $(\Omega, \mathcal{F}, \mathbb{P})$, $\mathcal{P}$ be the set of probability measures absolutely continuous with respect to $\mathbb{P}$, and $\rho : \mathcal{X} \to \mathbb{R}$ be a convex risk measure satisfying $X_n \downarrow X \Rightarrow \rho(X_n) \to \rho(X)$, then there exists a convex function $\alpha : \mathcal{P} \to (-\infty, +\infty]$ such that*

$$\rho(X) = \sup_{Q \in \mathcal{P}} \big(\mathbb{E}_Q(X) - \alpha(Q) \big) \quad \forall X \in \mathcal{X}. \tag{10.16}$$

Furthermore, $\alpha^0(Q) \triangleq \sup_{X' \in \mathcal{X}} \big(\mathbb{E}_Q(X') - \rho(X') \big)$ satisfies (10.16), and it is minimal in the sense that $\alpha^0(Q) \leq \alpha(Q)$ for all $Q \in \mathcal{P}$, if $\alpha(\cdot)$ also satisfies (10.16).

We call $\alpha^0(\cdot)$ the minimal representation of a convex risk measure.

LEMMA 10.10. *For a normalized convex risk measure $\rho(\cdot)$, its minimal representation satisfies:*

$$0 = \alpha^0(\mathbb{P}) \leq \alpha^0(Q), \, \forall Q \ll \mathbb{P}.$$

PROOF. First, since $\mathbb{E}_Q(0) \equiv 0$, we have

$$\rho(0) = 0 \to \inf_{Q \in \mathcal{P}} \alpha^0(Q) = 0. \tag{10.17}$$

Next, by definition $\alpha^0(\mathbb{P}) = \sup_{X \in \mathcal{X}} \big(\mathbb{E}_{\mathbb{P}}(X) - \rho(X) \big)$, and $\mathbb{E}_{\mathbb{P}}(X) \leq \rho(X)$ by assumption. Hence taking the supremum leads to $\alpha_0(\mathbb{P}) \leq 0$. Combining this with Equation (10.17) establishes the lemma. □

Now we proceed to prove Theorem 10.8.

(1) By Lemma 10.10, $f_i(\boldsymbol{\delta}_i) \geq 0$ since $\alpha^0(Q) \geq 0, \forall Q \in \mathcal{P}$. In addition, $\mathbb{E}_{\mathbb{P}}(\boldsymbol{\delta}_i) = \mathbf{0}$ and $\alpha^0(\mathbb{P}) = 0$ together imply $f_i(\mathbf{0}) = 0$. Hence $f_i(\cdot)$ satisfies (10.4).

Inequality (10.15) can be rewritten as

$$\begin{aligned}
&\xi_i + y_i(\langle \mathbf{w}, \mathbf{x}_i\rangle + b) - 1 \geq \sup_{Q\in\mathcal{P}} \big(\mathbb{E}_Q(y_i \mathbf{w}^\top \boldsymbol{\delta}_i^r) - \alpha(Q)\big) \\
\iff \quad & \xi_i + y_i(\langle \mathbf{w}, \mathbf{x}_i\rangle + b) - 1 \geq \sup_{\boldsymbol{\delta}_i\in\mathbb{R}^n} \sup_{Q\in\mathcal{P}|\mathbb{E}_Q(\boldsymbol{\delta}_i^r)=\boldsymbol{\delta}_i} \big(y_i \mathbf{w}^\top \boldsymbol{\delta}_i - \alpha(Q)\big) \\
\iff \quad & y_i(\langle \mathbf{w}, \mathbf{x}_i - \boldsymbol{\delta}_i\rangle + b) \geq 1 - \xi_i - \inf\{\alpha(Q)|\mathbb{E}_Q(\boldsymbol{\delta}_i^r) = \boldsymbol{\delta}_i\},\ \forall \boldsymbol{\delta}_i \in \mathbb{R}^n, \\
\iff \quad & y_i(\langle \mathbf{w}, \mathbf{x}_i - \boldsymbol{\delta}_i\rangle + b) \geq 1 - \xi_i - f_i(\boldsymbol{\delta}_i),\ \forall \boldsymbol{\delta}_i \in \mathbb{R}^n,
\end{aligned}$$

which proves the first part.

(2) First we show $\rho_i(\cdot)$ is a convex risk measure. Notice $f_i(\mathbf{0})$ is finite, hence, $\rho_i(X) > -\infty$. Observe that $\rho_i(\cdot)$ satisfies Translation Invariance. To prove Monotonicity, suppose $X \leq Y$ and $Y - s \in \mathcal{A}_i$ for some $s \in \mathbb{R}$, then $X - s \in \mathcal{A}_i$, hence $\inf\{m|X - m \in \mathcal{A}_i\} \leq s$, which implies $\rho_i(X) \leq \rho_i(Y)$. To prove Convexity, suppose $X - m$ and $Y - n$ belong to $\mathcal{A}_i$ for $m, n \in \mathbb{R}$. Given $\lambda \in [0, 1]$, we have $\lambda(X(\omega) - m) + (1 - \lambda)(Y(\omega) - n) \leq f_i\big(\boldsymbol{\delta}_i^r(\omega)\big)$ and hence $\big(\lambda X + (1 - \lambda)Y\big) - (\lambda m + (1 - \lambda)n) \in \mathcal{A}_i$ which implies $\rho_i(\lambda X + (1 - \lambda)Y) \leq \lambda m + (1 - \lambda)n$, hence the convexity holds. Therefore $\rho_i(\cdot)$ is a convex risk measure.

Inequality (10.15) can be rewritten as

$$\begin{aligned}
&\inf\{m \in \mathbb{R}|y_i \mathbf{w}^\top \boldsymbol{\delta}_i^r - m \in \mathcal{A}_i\} \leq \xi_i + y_i(\langle \mathbf{w}, \mathbf{x}_i\rangle + b) - 1 \\
\iff \quad & y_i \mathbf{w}^\top \boldsymbol{\delta}_i^r - \xi_i - y_i(\langle \mathbf{w}, \mathbf{x}_i\rangle + b) + 1 - \epsilon \in \mathcal{A}_i, \quad \forall \epsilon > 0 \\
\iff \quad & y_i \mathbf{w}^\top \boldsymbol{\delta}_i^r(\omega) - \xi_i - y_i(\langle \mathbf{w}, \mathbf{x}_i\rangle + b) + 1 - \epsilon \leq f_i(\boldsymbol{\delta}_i^r(\omega)), \qquad \forall \omega \in \Omega, \forall \epsilon > 0 \\
\iff \quad & y_i \mathbf{w}^\top \boldsymbol{\delta}_i - \xi_i - y_i(\langle \mathbf{w}, \mathbf{x}_i\rangle + b) + 1 - \epsilon \leq f_i(\boldsymbol{\delta}_i), \qquad \forall \boldsymbol{\delta}_i \in \mathbb{R}^n.
\end{aligned}$$

The last equivalence holds from the assumption that $\boldsymbol{\delta}_i^r$ has support $\mathbb{R}^n$.

□

Note that for the first part of Theorem 10.8, the assumption that $\rho_i(\cdot)$ is normalized can be relaxed to $\rho_i(0) = 0$ and $\inf\{\alpha_i^0(Q)|\mathbb{E}_Q(\boldsymbol{\delta}_i^r) = \mathbf{0}\} = 0$.

10.5.2. Risk-measure constrained classifier and distribution deviation. Let $\mathcal{P}$ be the set of probability measures absolutely continuous w.r.t. $\mathbb{P}$. It is known [**73, 9**] that any convex risk measure $\rho(\cdot)$ can be represented as $\rho(X) = \sum_{Q\in\mathcal{P}}[\mathbb{E}_Q(X) - \alpha(Q)]$ for some convex function $\alpha(\cdot)$; conversely, given any such convex function α, the resulting function $\rho(\cdot)$ is indeed a convex risk measure. Given $\alpha(\cdot)$, $\rho(\cdot)$ is called the corresponding risk measure. The function $\alpha(\cdot)$ can be thought of as a penalty function on probability distributions. This gives us a way to directly investigate classifier robustness with respect to distributional deviation. As an example, suppose we want to be robust over distributions that are nowhere more than a factor of two greater than a nominal distribution, $\mathbb{P}$. This can be captured by the risk constraint using risk measure $\rho(\cdot)$, where ρ corresponds to the convex function α given by letting $\alpha(\cdot)$ satisfy $\alpha(Q) = 0$ for $dQ/d\mathbb{P} \leq 2$, and $\alpha(Q) = +\infty$ for all other Q.

A natural notion of distributional divergence is the Kullback-Leibler divergence. The next result derives the corresponding risk measure when the reference noise, $\boldsymbol{\delta}_i^r$, is Gaussian.

THEOREM 10.11. *Suppose $\boldsymbol{\delta}_i^r \sim \mathcal{N}(0, \Sigma_i)$ and let $\rho(\cdot)$ be the corresponding risk measure of*

$$\alpha(Q) = \begin{cases} \int \frac{dQ}{d\mathbb{P}} \log \frac{dQ}{d\mathbb{P}} d\mathbb{P} & Q \ll \mathbb{P}, \\ +\infty & \textit{otherwise.} \end{cases}$$

Then the Risk-Measure Constrained Classifier is equivalent to

$$\begin{aligned} \min: \quad & r(\mathbf{w}, b) + \sum_{i=1}^{m} \xi_i, \\ \text{s.t.}: \quad & y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) - \mathbf{w}^\top \Sigma_i \mathbf{w}/2 \geq 1 - \xi_i, \;\; i = 1, \cdots, m, \\ & \xi_i \geq 0, \;\; i = 1, \cdots, m. \end{aligned}$$

PROOF. We first show that for the KL divergence, its corresponding convex risk measure equals $\log \mathbb{E}_{\mathbb{P}}[e^X]$ by applying the following theorem adapted from [**73**].

THEOREM 10.12. *Suppose a convex risk measure can be represented as*

$$\rho(X) = \inf\{m \in \mathbb{R} | \mathbb{E}_{\mathbb{P}}[l(X-m)] \leq x_0\},$$

for an increasing convex function $l : \mathbb{R} \to \mathbb{R}$ and scalar x_0. Then $\rho(\cdot)$ is the corresponding risk measure of

$$\alpha_0(Q) = \inf_{\lambda>0} \frac{1}{\lambda}\left(x_0 + \mathbb{E}_{\mathbb{P}}\big[l^*(\lambda\frac{dQ}{d\mathbb{P}})\big]\right).$$

Note that $\log \mathbb{E}_{\mathbb{P}}[e^X] = \inf\{m \in \mathbb{R} | \mathbb{E}_{\mathbb{P}}\big[e^{X-m}\big] \leq 1\}$, and hence the risk measure $\log \mathbb{E}_{\mathbb{P}}[e^X]$ can be represented as in the theorem, with $l(x) = e^x$, and $x_0 = 1$. The conclusion of the theorem tells us that $\log \mathbb{E}_{\mathbb{P}}[e^X]$ is the corresponding risk measure of

$$\begin{aligned}
\alpha_0(Q) &= \inf_{\lambda>0} \frac{1}{\lambda}\left(1 + \mathbb{E}_{\mathbb{P}}\big[\lambda\frac{dQ}{d\mathbb{P}}\log(\lambda\frac{dQ}{d\mathbb{P}}) - \lambda\frac{dQ}{d\mathbb{P}}\big]\right) \\
&= \mathbb{E}_{\mathbb{P}}\big[\frac{dQ}{d\mathbb{P}}\log\frac{dQ}{d\mathbb{P}}\big] + \inf_{\lambda>0}\big[\frac{1}{\lambda} + \mathbb{E}_{\mathbb{P}}(\frac{dQ}{d\mathbb{P}})(\log\lambda - 1)\big] \\
&= \begin{cases} \int \frac{dQ}{d\mathbb{P}}\log\frac{dQ}{d\mathbb{P}}d\mathbb{P} & Q \ll \mathbb{P}, \\ +\infty & \text{otherwise}, \end{cases}
\end{aligned}$$

where the last equation holds since $\mathbb{E}_{\mathbb{P}}(dQ/d\mathbb{P}) = 1$ and $\inf_{\lambda>0}(1/\lambda + \log\lambda - 1) = 0$. Therefore $\rho(X) = \log \mathbb{E}_{\mathbb{P}}[e^X]$ is indeed the corresponding risk measure to KL-divergence. Now we evaluate $\log \mathbb{E}_{\mathbb{P}}(e^{y_i \mathbf{w}^\top \boldsymbol{\delta}_i^r})$. Since $\boldsymbol{\delta}_i^r \sim N(0, \Sigma_i)$, $y_i\mathbf{w}^\top\boldsymbol{\delta}_i^r \sim N(0, \mathbf{w}^\top\Sigma_i\mathbf{w})$, which leads to

$$\begin{aligned}
\mathbb{E}_{\mathbb{P}}(e^{y_i\mathbf{w}^\top\boldsymbol{\delta}_i^r}) &= \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}}\exp\Big[-t^2/2\sqrt{\mathbf{w}^\top\Sigma_i\mathbf{w}}\Big]e^t dt \\
&= \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}}\exp\Big\{-(t-\sqrt{\mathbf{w}^\top\Sigma_i\mathbf{w}})^2\Big/2\sqrt{\mathbf{w}^\top\Sigma_i\mathbf{w}}\Big\}e^{\mathbf{w}^\top\Sigma_i\mathbf{w}/2}dt \\
&= e^{\mathbf{w}^\top\Sigma_i\mathbf{w}/2}\int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}}\exp\Big\{-(t-\sqrt{\mathbf{w}^\top\Sigma_i\mathbf{w}})^2\Big/2\sqrt{\mathbf{w}^\top\Sigma_i\mathbf{w}}\Big\}dt = e^{\mathbf{w}^\top\Sigma_i\mathbf{w}/2}.
\end{aligned}$$

Thus $\log \mathbb{E}_{\mathbb{P}}(e^{y_i\mathbf{w}^\top\boldsymbol{\delta}_i^r}) = \mathbf{w}^\top\Sigma_i\mathbf{w}/2$, proving the theorem. □

Observe that here we get a regularizer (in each constraint) that is the *square* of an ellipsoidal norm, and hence is different from the norm regularizer obtained from the robust classification framework. In fact, recalling the result from Section 10.3, we notice that the new regularizer is the result of a quadratic discount function, instead of the indicator discount function used by robust classification.

For general $\boldsymbol{\delta}_i^r$ and $\alpha(\cdot)$, it is not always straightforward to find and optimize the explicit form of the regularization term. Hence we sample, approximating $\mathbb{P}$ with its empirical distribution $\mathbb{P}_n$. This is equivalent to assuming $\boldsymbol{\delta}_i^r$ has finite support $\{\boldsymbol{\delta}_i^1, \cdots, \boldsymbol{\delta}_i^t\}$ with probability $\{p_1, \cdots, p_t\}$. We note that the distribution of the noise is often unknown, where only some samples of the noise are given. Therefore, the finite-support approach is often an appropriate method in practice.

THEOREM 10.13. *For $\boldsymbol{\delta}_i^r$ with finite support, the risk-measure constrained classifier is equivalent to*

$$\begin{aligned} \min: \quad & r(\mathbf{w}, b) + \sum_{i=1}^{m} \xi_i, \\ \text{s.t.}: \quad & y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) - \alpha^*\big(y_i \Delta_i^\top \mathbf{w} + \lambda_i \mathbf{1}\big) + \lambda_i \geq 1 - \xi_i, \ i = 1, \cdots, m; \\ & \xi_i \geq 0, \ i = 1, \cdots, m; \end{aligned}$$

where $\alpha^*(\mathbf{y}) \triangleq \sup_{\mathbf{x} \geq \mathbf{0}} \{\mathbf{y}^\top \mathbf{x} - \alpha(\mathbf{x})\}$ and $\Delta_i \triangleq \{\boldsymbol{\delta}_i^1, \cdots, \boldsymbol{\delta}_i^t\}$.

PROOF. It suffices to prove that Constraint (10.15) is equivalent to

$$y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) - \alpha^*\big(y_i \Delta_i^\top \mathbf{w} + \lambda_i \mathbf{1}\big) + \lambda_i \geq 1 - \xi_i,$$

which is the same as showing that the conjugate function of

$$f_i(\boldsymbol{\delta}) \triangleq \inf\{\alpha(\mathbf{q}) | \sum_{j=1}^{t} q_j \delta_i^j = \boldsymbol{\delta}\}$$

evaluated at $y_i \mathbf{w}$ equals

$$\min_{\lambda} \{\alpha^*\big(y_i \Delta_i^\top \mathbf{w} + \lambda \mathbf{1}\big) - \lambda\}.$$

By definition, $f^*(y_i\mathbf{w}) = \sup_{\boldsymbol{\delta}\in\mathbb{R}^n}\big\{y_i\mathbf{w}^\top\boldsymbol{\delta} - f(\boldsymbol{\delta})\big\}$, which equals

$$\begin{aligned} \text{maximize on } \boldsymbol{\delta}, \mathbf{q}\text{:} \quad & y_i\mathbf{w}^\top\boldsymbol{\delta} - \alpha(\mathbf{q}) \\ \text{subject to:} \quad & \Delta_i\mathbf{q} - \boldsymbol{\delta} = 0, \\ & \mathbf{1}^\top\mathbf{q} = 1 \\ & \mathbf{q} \geq \mathbf{0}. \end{aligned} \tag{10.18}$$

Notice that (10.18) equals

$$\mathcal{L}(\boldsymbol{\delta}, \mathbf{q}, \mathbf{c}, \lambda) \;\triangleq\; \max_{\boldsymbol{\delta};\, \mathbf{q}\geq\mathbf{0}} \min_{\mathbf{c},\lambda} \big\{y_i\mathbf{w}^\top\boldsymbol{\delta} - \alpha(\mathbf{q}) + \mathbf{c}^\top\Delta_i\mathbf{q} - \mathbf{c}^\top\boldsymbol{\delta} + \lambda\mathbf{1}^\top\mathbf{q} - \lambda\big\}.$$

Since Problem (10.18) is convex and all constraints are linear, Slater's condition is satisfied and the duality gap is zero. Hence, we can exchange the order of minimization and maximization:

$$\begin{aligned} \mathcal{L}(\boldsymbol{\delta}, \mathbf{q}, \mathbf{c}, \lambda) &= \min_{\mathbf{c},\lambda} \max_{\boldsymbol{\delta},\mathbf{q}\geq\mathbf{0}} \big\{y_i\mathbf{w}^\top\boldsymbol{\delta} - \alpha(\mathbf{q}) + \mathbf{c}^\top\Delta_i\mathbf{q} - \mathbf{c}^\top\boldsymbol{\delta} + \lambda\mathbf{1}^\top\mathbf{q} - \lambda\big\} \\ &= \min_{\mathbf{c},\lambda} \big\{ \max_{\boldsymbol{\delta}} \big(y_i\mathbf{w}^\top\boldsymbol{\delta} - \mathbf{c}^\top\boldsymbol{\delta}\big) + \max_{\mathbf{q}\geq\mathbf{0}} \big(\mathbf{c}^\top\Delta_i\mathbf{q} + \lambda\mathbf{1}^\top\mathbf{q} - \alpha(\mathbf{q})\big) - \lambda\big\} \\ &= \min_{\lambda} \big\{ \max_{\mathbf{q}\geq\mathbf{0}} \big(y_i\mathbf{w}^\top\Delta_i\mathbf{q} + \lambda\mathbf{1}^\top\mathbf{q} - \alpha(\mathbf{q})\big) - \lambda\big\} \\ &= \min_{\lambda} \alpha^*\big(y_i\Delta_i^\top\mathbf{w} + \lambda\mathbf{1}\big) - \lambda. \end{aligned}$$

The third equality holds because $\mathbf{c} = y_i\mathbf{w}$ is the necessary condition to make $\max_{\boldsymbol{\delta}} \big(y_i\mathbf{w}^\top\boldsymbol{\delta} - \mathbf{c}^\top\boldsymbol{\delta}\big)$ finite. □

Example. Let $\alpha(\mathbf{q}) = \sum_{j=1}^t q_j \log(q_j/p_j)$, the KL divergence for discrete probability measures. By applying Theorem 10.13, Constraint (10.15) is equivalent to

$$\begin{aligned} & y_i(\langle\mathbf{w},\, \mathbf{x}_i\rangle + b) - \log\big(\sum_{j=1}^t p_j \exp(y_i\mathbf{w}^\top\boldsymbol{\delta}_i^j)\big) \geq 1 - \xi_i, \\ \iff \quad & \sum_{j=1}^t p_j \exp\Big(y_i\mathbf{w}^\top\boldsymbol{\delta}_i^j - y_i(\langle\mathbf{w},\, \mathbf{x}_i\rangle + b) + 1 - \xi_i\Big) \leq 1. \end{aligned}$$

This is a geometric program, which is a well-studied class of convex problems with specialized and efficient algorithms for their solution [**33**].

There is substantial research on how a convex risk measure is approximated by a finite number of samples. For example, [**34**] proved the following result for CVaR.

THEOREM 10.14. *Suppose a random variable X satisfies* $\text{support}(X) \subseteq [0, U]$. *$X_1, \cdots, X_N$ are independent realizations of X. Denote $\overline{X}$ as a random variable with probability $1/N$ on X_i, and let*

$$\overline{\text{CVaR}}_\alpha(X_1, \cdots, X_N) \triangleq \text{CVaR}_\alpha(\overline{X}),$$

i.e., the CVaR estimated according to N samples. For $\alpha \in (0, 1]$ we have

$$\mathbb{P}\big(\overline{\text{CVaR}}_\alpha(X_1, \cdots, X_N) \geq \text{CVaR}_\alpha(X) + \epsilon\big) \leq \exp\big(\frac{-2N\alpha^2\epsilon^2}{U^2}\big);$$
$$\mathbb{P}\big(\overline{\text{CVaR}}_\alpha(X_1, \cdots, X_N) \leq \text{CVaR}_\alpha(X) - \epsilon\big) \leq 3\exp\big(\frac{-N\alpha\epsilon^2}{5U^2}\big).$$

10.6. Kernelized comprehensive robust classifier

Much of the previous development can be extended to the kernel space. The main contributions in this section are (i) in Section 10.6.1 we provide a representer theorem in the case where we have discount functions in the feature space; and (ii) in Section 10.6.2 we provide a sufficient condition for approximation in the case that we have discount functions in the original sample space.

We use $k(\cdot, \cdot) : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ to represent the kernel function, and K to denote the Gram matrix with respect to $(\mathbf{x}_1, \cdots, \mathbf{x}_m)$. We assume that K is a non-zero matrix without loss of generality.

10.6.1. Comprehensive robustness in feature space. We first investigate the case where the noise exists explicitly in the feature space. Let $\phi(\cdot)$ be the mapping from the sample space $\mathbb{R}^n$ to the feature space Φ. Let $\hat{\Phi} \subseteq \Phi$ be the subspace spanned by $\{\phi(\mathbf{x}_1), \cdots, \phi(\mathbf{x}_m)\}$. For a vector $\mathbf{z} \in \Phi$, denote $\mathbf{z}^=$ as its projection on $\hat{\Phi}$, and $\mathbf{z}^\perp \triangleq \mathbf{z} - \mathbf{z}^=$ as its residual. The following theorem states that we can focus on $\mathbf{w} \in \hat{\Phi}$ without loss of generality.

THEOREM 10.15. *If $f_i(\cdot)$ is such that*

$$f_i(\boldsymbol{\delta}) \geq f_i(\boldsymbol{\delta}^{=}), \quad \forall \boldsymbol{\delta} \in \Phi,$$

and $\mathbf{w} \in \Phi$ satisfies

$$y(\langle \mathbf{w}, \phi(\mathbf{x}_i) - \boldsymbol{\delta}_i \rangle + b) \geq 1 - \xi_i - f_i(\boldsymbol{\delta}_i), \quad \forall \boldsymbol{\delta}_i \in \Phi, \tag{10.19}$$

then its projection $\mathbf{w}^{=}$ also satisfies (10.19).

PROOF. Before proving this theorem, we first establish the following two lemmas.

LEMMA 10.16. *If $\mathbf{w} \in \Phi$ satisfies*

$$y(\langle \mathbf{w}, \phi(\mathbf{x}_i) - \boldsymbol{\delta}_i \rangle + b) \geq 1 - \xi_i - f_i(\boldsymbol{\delta}_i), \quad \forall \boldsymbol{\delta}_i \in \hat{\Phi}, \tag{10.20}$$

then its projection $\mathbf{w}^{=}$ also satisfies (10.20).

PROOF. Decompose $\mathbf{w} = \mathbf{w}^{=} + \mathbf{w}^{\perp}$. By definition, $\mathbf{w}^{\perp}$ is orthogonal to $\hat{\Phi}$. Since $\boldsymbol{\delta}_i \in \hat{\Phi}$ and $\phi(\mathbf{x}_i) \in \hat{\Phi}$, we have

$$\langle \mathbf{w}^{\perp}, \phi(\mathbf{x}_i) - \boldsymbol{\delta}_i \rangle = 0, \quad \forall \boldsymbol{\delta}_i \in \hat{\Phi},$$

which establishes the lemma. □

LEMMA 10.17. *If $f_i(\cdot)$ is such that*

$$f_i(\boldsymbol{\delta}) \geq f_i(\boldsymbol{\delta}^{=}), \quad \forall \boldsymbol{\delta} \in \Phi,$$

and $\mathbf{w} \in \hat{\Phi}$ satisfies

$$y(\langle \mathbf{w}, \phi(\mathbf{x}_i) - \hat{\boldsymbol{\delta}}_i \rangle + b) \geq 1 - \xi_i - f_i(\hat{\boldsymbol{\delta}}_i), \quad \forall \hat{\boldsymbol{\delta}}_i \in \hat{\Phi}, \tag{10.21}$$

then $\mathbf{w}$ satisfies

$$y(\langle \mathbf{w}, \phi(\mathbf{x}_i) - \boldsymbol{\delta}_i \rangle + b) \geq 1 - \xi_i - f_i(\boldsymbol{\delta}_i), \quad \forall \boldsymbol{\delta}_i \in \Phi. \tag{10.22}$$

PROOF. We prove this lemma by deriving a contradiction. Assume that there exists $\boldsymbol{\delta}' \in \Phi$ such that Inequality (10.22) does not hold, i.e.,

$$y(\langle \mathbf{w}, \phi(\mathbf{x}_i) - \boldsymbol{\delta}' \rangle + b) < 1 - \xi_i - f_i(\boldsymbol{\delta}').$$

Decompose $\boldsymbol{\delta}' = \boldsymbol{\delta}'^{=} + \boldsymbol{\delta}'^{\perp}$. Hence we have $f_i(\boldsymbol{\delta}'^{=}) \leq f_i(\boldsymbol{\delta}')$ by assumption, and $\langle \mathbf{w}, \boldsymbol{\delta}'^{\perp} \rangle = 0$ since $\mathbf{w} \in \hat{\Phi}$. This leads to

$$\begin{aligned} y(\langle \mathbf{w}, \phi(\mathbf{x}_i) - \boldsymbol{\delta}'^{=} \rangle + b) &= y(\langle \mathbf{w}, \phi(\mathbf{x}_i) - \boldsymbol{\delta}' \rangle + b) \\ &< 1 - \xi_i - f_i(\boldsymbol{\delta}') \leq 1 - \xi_i - f_i(\boldsymbol{\delta}'^{=}), \end{aligned}$$

which contradicts (10.21) and hence we prove the lemma. □

Now we proceed to prove Theorem 10.15. Since $\mathbf{w}$ satisfies (10.19), then it also satisfies

$$y(\langle \mathbf{w}, \phi(\mathbf{x}_i) - \boldsymbol{\delta}_i \rangle + b) \geq 1 - \xi_i - f_i(\boldsymbol{\delta}_i), \quad \forall \boldsymbol{\delta}_i \in \hat{\Phi}.$$

Thus by Lemma 10.16, $\mathbf{w}^{=}$ satisfies

$$y(\langle \mathbf{w}^{=}, \phi(\mathbf{x}_i) - \boldsymbol{\delta}_i \rangle + b) \geq 1 - \xi_i - f_i(\boldsymbol{\delta}_i), \quad \forall \boldsymbol{\delta}_i \in \hat{\Phi}.$$

By Lemma 10.17, this implies that $\mathbf{w}^{=}$ satisfies

$$y(\langle \mathbf{w}^{=}, \phi(\mathbf{x}_i) - \boldsymbol{\delta}_i \rangle + b) \geq 1 - \xi_i - f_i(\boldsymbol{\delta}_i), \quad \forall \boldsymbol{\delta}_i \in \Phi,$$

which establishes the theorem. □

The kernelized comprehensive robust classifier can be written as:

Kernelized Comprehensive Robust Classifier:

$$\begin{aligned}
\min: \quad & r\big(\sum_{j=1}^{m}\alpha_j\phi(\mathbf{x}_j), b\big) + \sum_{i=1}^{m}\xi_i, \\
\text{s.t.}: \quad & y_i(\langle\sum_{j=1}^{m}\alpha_j\phi(\mathbf{x}_j),\ \phi(\mathbf{x}_i) - \sum_{j=1}^{m}c_j\phi(\mathbf{x}_j)\rangle + b) \geq \\
& \qquad 1 - \xi_i - f_i\big(\sum_{j=1}^{m}c_j\phi(\mathbf{x}_j)\big), \quad \forall\big(c_1,\cdots,c_m\big)\in\mathbb{R}^m,\ i=1,\cdots,m, \\
& \xi_i \geq 0,\ i=1,\cdots,m,
\end{aligned} \tag{10.23}$$

Define $\mathbf{c} \triangleq (c_1,\cdots,c_m)$, $g_i(\mathbf{c}) \triangleq f_i(\sum_{i=1}^{m}c_i\phi(\mathbf{x}_i))$, and $\tilde{r}(\boldsymbol{\alpha}, b) \triangleq r\big(\sum_{j=1}^{m}\alpha_j\phi(\mathbf{x}_j), b\big)$. Let $\mathbf{e}_i$ denote the i^{th} basis vector. Then Problem (10.23) can be rewritten as

$$\begin{aligned}
\min: \quad & \tilde{r}(\boldsymbol{\alpha}, b) + \sum_{i=1}^{m}\xi_i, \\
\text{s.t.}: \quad & y_i(\mathbf{e}_i^\top K\boldsymbol{\alpha} + b) - y_i\boldsymbol{\alpha}^\top K\mathbf{c} \geq 1 - \xi_i - g_i(\mathbf{c}), \qquad \forall\mathbf{c}\in\mathbb{R}^m,\ i=1,\cdots,m, \\
& \xi_i \geq 0,\ i=1,\cdots,m,
\end{aligned}$$

where the constraint can be further simplified as

$$y_i(\mathbf{e}_i^\top K\boldsymbol{\alpha} + b) - g_i^*(y_iK\boldsymbol{\alpha}) \geq 1 - \xi_i,\ i=1,\cdots,m.$$

Notice that generally $g^*(\cdot)$ depends on the exact formulation of the feature mapping $\phi(\cdot)$. However, for the following specific class of $f(\cdot)$, we can determine $g^*(\cdot)$ from K without knowing $\phi(\cdot)$.

THEOREM 10.18. *If there exists $h_i : \mathbb{R}^+ \to \mathbb{R}^+$ such that*

$$f_i(\boldsymbol{\delta}) = h_i(\sqrt{\langle\boldsymbol{\delta},\ \boldsymbol{\delta}\rangle}), \forall\boldsymbol{\delta}\in\Phi,$$

then

$$g_i^*(y_iK\alpha) = h_i^*(\|\boldsymbol{\alpha}\|_K).$$

PROOF. By definition,

$$\begin{aligned} g_i^*(y_i K\alpha) &= \sup_{\mathbf{c}\in\mathbb{R}^m} \left\{ y_i \boldsymbol{\alpha}^\top K\mathbf{c} - g_i(\mathbf{c}) \right\} \\ &= \sup_{\mathbf{c}\in\mathbb{R}^m} \left\{ y_i \boldsymbol{\alpha}^\top K\mathbf{c} - f_i\big(\sum_{j=1}^m c_j \phi(\mathbf{x}_j) \big) \right\} \\ &= \sup_{\mathbf{c}\in\mathbb{R}^m} \left\{ y_i \boldsymbol{\alpha}^\top K\mathbf{c} - h_i \left(\sqrt{\langle \sum_{j=1}^m c_j \phi(\mathbf{x}_j), \sum_{j=1}^m c_j \phi(\mathbf{x}_j) \rangle} \right) \right\} \\ &= \sup_{\mathbf{c}\in\mathbb{R}^m} \left\{ y_i \boldsymbol{\alpha}^\top K\mathbf{c} - h_i(\sqrt{\mathbf{c}^\top K\mathbf{c}}) \right\}. \end{aligned}$$

Notice the right-hand side can be written as

$$\begin{aligned} \sup_{\mathbf{c}\in\mathbb{R}^m} \left\{ (y_i K^{1/2}\boldsymbol{\alpha})^\top (K^{1/2}\mathbf{c}) - h_i(\|K^{1/2}\mathbf{c}\|_2) \right\} &= \sup_{\mathbf{c}\in\mathbb{R}^m} \left\{ \|y_i K^{1/2}\boldsymbol{\alpha}\|_2 \|K^{1/2}\mathbf{c}\|_2 - h_i(\|K^{1/2}\mathbf{c}\|_2) \right\} \\ &= \sup_{s\in\mathbb{R}^+} \left\{ s\|(K^{1/2}\boldsymbol{\alpha})\|_2 - h_i(s) \right\} = h_i^*(\|\boldsymbol{\alpha}\|_K). \end{aligned}$$

Here, the first equality holds since

$$(y_i K^{1/2}\boldsymbol{\alpha})^\top (K^{1/2}\mathbf{c}) \le \|y_i K^{1/2}\boldsymbol{\alpha}\|_2 \|K^{1/2}\mathbf{c}\|_2$$

by Hölder's inequality. And the equality can be reached by taking $\mathbf{c}$ equal to $y_i\boldsymbol{\alpha}$ multiplied by a constant. The third equality holds because when K is non-zero, $\|K^{1/2}\mathbf{c}\|_2$ ranges over $\mathbb{R}^+$. □

Notice that when h_i is an increasing function, then $f_i(\boldsymbol{\delta}) \ge f_i(\boldsymbol{\delta}^=)$ is automatically satisfied $\forall \boldsymbol{\delta} \in \Phi$.

10.6.2. Comprehensive robustness in sample space. The previous results hold for the case where we have explicit discount functions in the feature space. However, in certain cases the discount functions naturally lie in the original sample space. The next theorem gives a sufficient alternative in this case.

THEOREM 10.19. *Suppose* $h_i : \mathbb{R}^+ \to \mathbb{R}^+$ *satisfies*

$$h_i\big(\sqrt{k(\mathbf{x}_i,\mathbf{x}_i) + k(\mathbf{x}_i-\boldsymbol{\delta},\mathbf{x}_i-\boldsymbol{\delta}) - 2k(\mathbf{x}_i,\mathbf{x}_i-\boldsymbol{\delta})}\big) \le f_i(\boldsymbol{\delta}), \quad \forall \boldsymbol{\delta} \in \mathbb{R}^n \tag{10.24}$$

Then

$$y_i\big(\langle \mathbf{w},\, \phi(\mathbf{x}_i) - \boldsymbol{\delta}_{\boldsymbol{\phi}}\rangle + b\big) \geq 1 - \xi_i - h_i(\sqrt{\langle \boldsymbol{\delta}_{\boldsymbol{\phi}}, \boldsymbol{\delta}_{\boldsymbol{\phi}}\rangle}), \quad \forall \boldsymbol{\delta}_{\boldsymbol{\phi}} \in \Phi, \quad (10.25)$$

implies

$$y_i\big(\langle \mathbf{w}, \phi(\mathbf{x}_i - \boldsymbol{\delta})\rangle + b\big) \geq 1 - \xi_i - f_i(\boldsymbol{\delta}), \quad \forall \boldsymbol{\delta} \in \mathbb{R}^n. \quad (10.26)$$

PROOF. Notice that (10.25) implies that

$$y_i\Big(\big\langle \mathbf{w},\, \phi(\mathbf{x}_i) - \big[\phi(\mathbf{x}_i) - \phi(\mathbf{x}_i - \boldsymbol{\delta})\big]\big\rangle + b\Big) \geq 1 - \xi_i - h_i\big(\sqrt{\big\langle \phi(\mathbf{x}_i) - \phi(\mathbf{x}_i - \boldsymbol{\delta}),\, \phi(\mathbf{x}_i) - \phi(\mathbf{x}_i - \boldsymbol{\delta})\big\rangle}\big),$$

$\forall \boldsymbol{\delta} \in \mathbb{R}^n$. The right-hand side is equal to

$$1 - \xi_i - h_i\big(\sqrt{k(\mathbf{x}_i,\, \mathbf{x}_i) + k(\mathbf{x}_i - \boldsymbol{\delta}, \mathbf{x}_i - \boldsymbol{\delta}) - 2k(\mathbf{x}_i, \mathbf{x}_i - \boldsymbol{\delta})}\big) \geq 1 - \xi_i - f_i(\boldsymbol{\delta}).$$

Since this holds for all $\boldsymbol{\delta} \in \mathbb{R}^n$, (10.26) holds for $(\mathbf{w},\, b)$. □

In fact, when Equation (10.24) holdswith equality, this sufficient condition is also necessary, as the next theorem states.

THEOREM 10.20. *Suppose $h_i : \mathbb{R}^+ \to \mathbb{R}^+$ is upper semi-continuous and satisfies*

$$h_i\big(\sqrt{k(\mathbf{x}_i, \mathbf{x}_i) + k(\mathbf{x}_i - \boldsymbol{\delta}, \mathbf{x}_i - \boldsymbol{\delta}) - 2k(\mathbf{x}_i, \mathbf{x}_i - \boldsymbol{\delta})}\big) \geq f_i(\boldsymbol{\delta}), \quad \forall \boldsymbol{\delta} \in \mathbb{R}^n, \quad (10.27)$$

and Φ is the Reproducing Kernel Hilbert Space. Then Condition (10.26) implies Condition (10.25).

PROOF. Condition (10.26) and Inequality (10.27) implies that

$$y_i\big(\langle \mathbf{w},\, \phi(\mathbf{x}_i) - \boldsymbol{\delta}_{\boldsymbol{\phi}}\rangle + b\big) \geq 1 - \xi_i - h_i(\sqrt{\langle \boldsymbol{\delta}_{\boldsymbol{\phi}}, \boldsymbol{\delta}_{\boldsymbol{\phi}}\rangle}), \quad \forall \boldsymbol{\delta}_{\boldsymbol{\phi}} : \exists \boldsymbol{\delta} \in \mathbb{R}^n, \boldsymbol{\delta}_{\boldsymbol{\phi}} = \phi(\mathbf{x}_i) - \phi(\mathbf{x}_i - \boldsymbol{\delta}).$$

Denote $\mathbf{z} = \phi(\mathbf{x}_i) - \boldsymbol{\delta}_{\boldsymbol{\phi}}$, we have

$$y_i\big(\langle \mathbf{w},\, \mathbf{z}\rangle + b\big) \geq 1 - \xi_i - h_i(\sqrt{\langle \phi(\mathbf{x}_i) - \mathbf{z}, \phi(\mathbf{x}_i) - \mathbf{z}\rangle}), \quad \forall \mathbf{z} : \exists \mathbf{x}' \in \mathbb{R}^n, \mathbf{z} = \phi(\mathbf{x}').$$

Notice that the Reproducing Kernel Hilbert Space is the completing of the image of the feature mapping, i.e., $\overline{\phi(\mathbb{R}^n)}$, hence for any $\mathbf{z}' \in \Phi$, there exists a sequence of $\mathbf{z}_t \to \mathbf{z}'$ where $\mathbf{z}_t = \phi(\mathbf{x}'_t)$. By continuity of the dot product, we have

$$\begin{aligned} y_i\big(\langle \mathbf{w}, \mathbf{z}' \rangle + b\big) &= \lim_{t\to\infty} y_i\big(\langle \mathbf{w}, \mathbf{z}_t \rangle + b\big) \\ &\geq \lim_{t\to\infty} \Big\{1 - \xi_i - h_i(\sqrt{\langle \phi(\mathbf{x}_i) - \mathbf{z}_t, \phi(\mathbf{x}_i) - \mathbf{z}_t \rangle})\Big\} \\ &\geq 1 - \xi_i - h_i(\sqrt{\langle \phi(\mathbf{x}_i) - \mathbf{z}', \phi(\mathbf{x}_i) - \mathbf{z}' \rangle}), \end{aligned} \tag{10.28}$$

where the last inequality follows from the assumption that $h_i(\cdot)$ is upper semi-continuous. Notice Inequality (10.28) holds for arbitrary $\mathbf{z}' \in \Phi$, which is equivalent to Condition (10.25). □

Notice the condition in Theorem 10.19 and Theorem 10.20 only involves the kernel function $k(\cdot, \cdot)$ and is independent of the explicit feature mapping. Hence this theorem applies for abstract mappings, and specifically mappings into infinite-dimensional spaces.

THEOREM 10.21. *Equip the sample space with a metric $d(\cdot, \cdot)$, and suppose there exist $\hat{k}_i : \mathbb{R}^+ \to \mathbb{R}$, and $\hat{f}_i : \mathbb{R}^+ \to \mathbb{R} \bigcup \{+\infty\}$ such that,*

$$\begin{aligned} k(\mathbf{x}, \mathbf{x}') &= \hat{k}(d(\mathbf{x}, \mathbf{x}')), \quad \forall \mathbf{x}, \mathbf{x}' \in \mathbb{R}^n; \\ f_i(\boldsymbol{\delta}) &= \hat{f}_i(d(\mathbf{x}_i, \mathbf{x}_i - \boldsymbol{\delta}_i)), \forall \boldsymbol{\delta} \in \mathbb{R}^n. \end{aligned} \tag{10.29}$$

Then $h_i : \mathbb{R}^+ \to \mathbb{R}^+ \bigcup \{+\infty\}$ defined as

$$h_i(x) = \inf_{y | \exists \mathbf{z} \in \mathbb{R}^n : y = d(\mathbf{x}_i, \mathbf{z}),\, \hat{k}(y) = \hat{k}(0) - x^2/2} \hat{f}_i(y) \tag{10.30}$$

satisfies Equation (10.24), and for any $h'(\cdot)$ that satisfies Equation (10.24), $h'(x) \leq h(x), \forall x \geq 0$ holds. Here, we take $\inf_{y\in\emptyset} \hat{f}_i(y)$ to be $+\infty$.

PROOF. Rewrite Inequality (10.24) as

$$h_i\big(\sqrt{\hat{k}(d(\mathbf{x}_i, \mathbf{x}_i)) + \hat{k}(d(\mathbf{x}_i - \boldsymbol{\delta}, \mathbf{x}_i - \boldsymbol{\delta})) - 2k(d(\mathbf{x}_i, \mathbf{x}_i - \boldsymbol{\delta}))}\big) \leq \hat{f}_i(d(\mathbf{x}_i, \mathbf{x}_i - \boldsymbol{\delta})),\ \forall \boldsymbol{\delta} \in \mathbb{R}^n,$$

which is equivalent to

$$h_i(x) \le \hat{f}_i(y); \quad \forall x, y, \mathbf{z} : x = \sqrt{2\hat{k}(0) - y};\ y = d(\mathbf{x}_i, \mathbf{z}).$$

Observe that the function defined by (10.30) is the maximal function that satisfies this inequality, thus proving the theorem. □

REMARK 10.2. In many cases, $\hat{f}_i$ is increasing and piecewise continuous, $d(\cdot,\cdot)$ satisfies that for any $y \ge 0$, there exists $\mathbf{z} \in \mathbb{R}^n$ such that $d(\mathbf{x}_i, \mathbf{z}) = y$. Equation (10.30) can be simplified to

$$h_i(x) = \begin{cases} +\infty & \{y|\hat{k}(y) = \hat{k}(0) - x^2/2\} \text{ is empty} \\ \hat{f}_i\Big(\hat{k}^{-1}\big(\hat{k}(0) - x^2/2\big)\Big) & \min\{y|\hat{k}(y) = \hat{k}(0) - x^2/2\} \text{ exists} \\ \hat{f}_i\Big(\hat{k}^{-1}\big(\hat{k}(0) - x^2/2\big)^+\Big) & \text{otherwise.} \end{cases}$$

Here, $\hat{k}^{-1}(x) \triangleq \inf\{y|\hat{k}(y) = x\}$, and $\hat{f}_i(c^+)$ stands for the right limit at c of $f_i(\cdot)$.

Consider the Gaussian Kernel $k(\mathbf{x}, \mathbf{x}') = \exp(-\|\mathbf{x} - \mathbf{x}'\|^2/2\sigma^2)$ as an example. We have $d(\mathbf{x}, \mathbf{x}') = \|\mathbf{x} - \mathbf{x}'\|$ and $\hat{k}(x) = \exp(-x^2/2\sigma^2)$. Hence $\hat{k}^{-1}(y) = \sqrt{-2\sigma^2 \ln y}$ yields

$$h_i(x) = \begin{cases} \hat{f}_i\Big(\sqrt{-2\sigma^2 \ln(1 - x^2/2)}\Big) & x < \sqrt{2} \\ +\infty & \text{otherwise.} \end{cases}$$

Taking $\hat{f}_i(x) = I_c$, the corresponding $h_i(x) = I_{\sqrt{2-2\exp(-c^2/2\sigma^2)}}$. Taking $\hat{f}_i(x) = cx^2$, the corresponding $h_i(x)$ is

$$h_i(x) = \begin{cases} -2c\sigma^2 \ln(1 - x^2/2) & x < \sqrt{2} \\ +\infty & \text{otherwise.} \end{cases}$$

It is easy to check that the condition of Theorem 10.20 also holds, hence the corresponding robustness requirement in the feature space is both necessary and sufficient.

10.7. Numerical simulations

In this section, we use empirical experiments to gain further insight into the performance of the comprehensive robust classifier. To this end, we compare the performance of three classification algorithms: the standard SVM, the standard robust SVM with ellipsoidal uncertainty set, and comprehensive robust SVM with ellipsoidal uncertainty set with linear discount function from the center of the ellipse to its boundary (see below). The simulation results show that a comprehensive robust classifier with the discount function appropriately tuned has a performance superior to both the robust classifier and the standard SVM. The empirical results show that this soft formulation of robustness builds in protection to noise, without being overly conservative.

We use the non-kernelized version for both the robust classification and the comprehensive robust classification. We use a linear discount function for the comprehensive robust classifier. That is, noise is bounded in the same ellipsoidal set as for the robust SVM, $\{\boldsymbol{\delta} | \|\boldsymbol{\delta}\|_{\Sigma^{-1}} \leq 1\}$, and the discount function is

$$f_i(\boldsymbol{\delta}) = \begin{cases} \alpha \|\boldsymbol{\delta}\|_{\Sigma^{-1}} & \|\boldsymbol{\delta}\|_{\Sigma^{-1}} \leq 1, \\ +\infty & \text{otherwise.} \end{cases}$$

The parameter α controls the disturbance discount. As α tends to zero, there is no discount inside the uncertainty set, and we recover the robust classifier. As α tends to $+\infty$, the discount increases until effectively the constraint is only imposed at the center of the ellipse, hence recovering the standard SVM classifier.

We use SeduMi 1.1R3 [**144**] to solve the resulting convex programs. We first compare the performance of the three algorithms on the Wisconsin-Breast-Cancer data set from the UCI repository [**3**]. In each iteration, we randomly pick 50% of the samples as training samples and the rest as testing samples. Each sample is corrupted by i.i.d. noise, which is uniformly distributed in an ellipsoid $\{\boldsymbol{\delta} | \|\boldsymbol{\delta}\|_{\Sigma^{-1}} \leq 1\}$. Here, the matrix Σ is diagonal. For the first 40% of features, $\Sigma_{ii} = 16$, and for the remaining features, $\Sigma_{ii} = 1$. This ellipsoidal uncertainty set captures the setup

where noise is skewed toward part of the features. We repeat 30 such iterations to get the average empirical error of the three different algorithms. Figure 10.3 shows that for appropriately chosen discount parameter α, the comprehensive robust classifier outperforms both the robust and standard SVM classifiers. As anticipated, when α is small, comprehensive robust classification has a testing error rate comparable to robust classification. For large α, the classifier's performance is similar to that of the standard SVM. This figure essentially shows that protection against noise is beneficial as long as it does not become overly conservative, and comprehensive robust classification provides a more flexible approach to handle the noise.

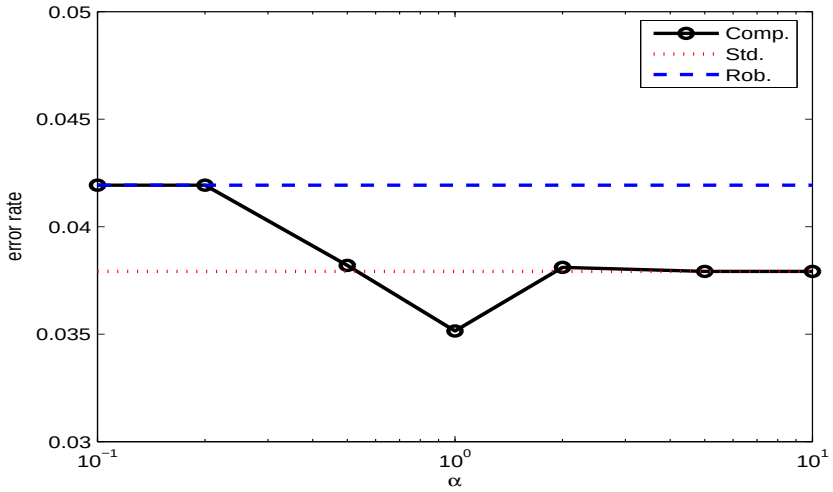


FIGURE 10.3. Empirical error for WBC data.

We run similar simulations on Ionosphere and Sonar data sets from the UCI repository [**3**]. To fit the variability of the data, we scale the uncertainty set: for 40% of the features, Σ_{ii} equals 0.3 for Ionosphere and 0.01 for Sonar; for the remaining features, Σ_{ii} equals 0.0003 for Ionosphere and 0.00001 for Sonar. Figure 10.4 and Figure 10.5 show the respective simulation results. Similarly to the WBC data set, comprehensive robust classification achieves its optimal performance for mid-range α, and is superior to both the standard SVM and the robust SVM.

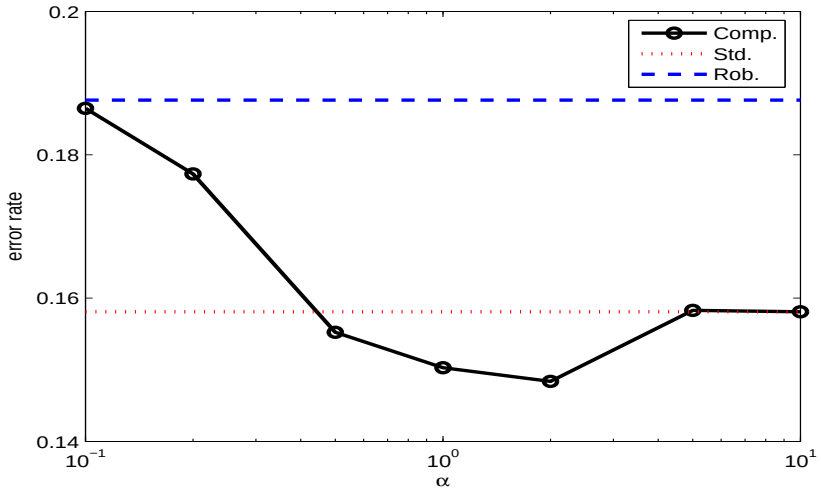


FIGURE 10.4. Empirical error for Ionosphere data.

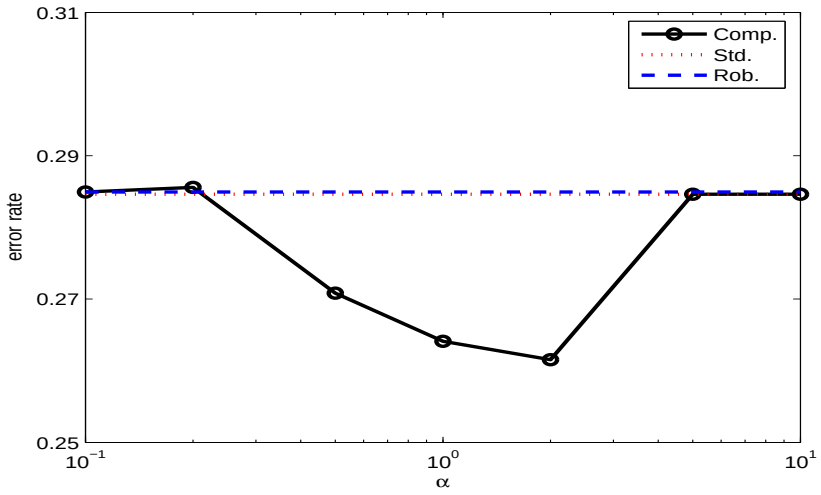


FIGURE 10.5. Empirical error for Sonar data.

The noise resistance ability of the resulting classifiers is also of interest, especially in the case where the noise is adversarial, or non i.i.d. This is measured using percentile performance: for each testing sample, we generate 100 independent noise realization and measure the probability (i.e., confidence) that this testing sample is correctly classified. The percentage of testing samples that achieves each confidence threshold is reported in Figure 10.6, Figure 10.7 and Figure 10.8. The standard SVM

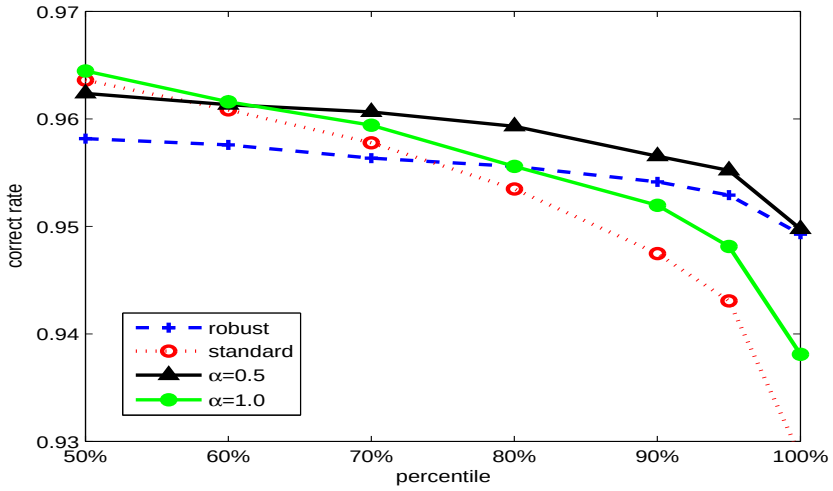


FIGURE 10.6. Percentile performance for WBC data.

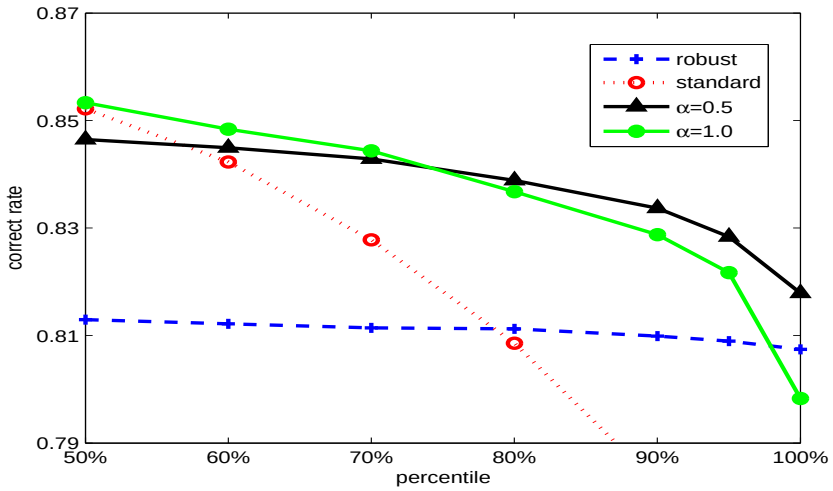


FIGURE 10.7. Percentile performance for Ionosphere data.

has a good performance for the 50% threshold, but it degrades significantly as the threshold increases, indicating a lack of noise-protection. The robust classifier tends to be overly conservative. The comprehensive robust classifier with α appropriately tuned performs well at all thresholds, especially in the 60% to 80% range, indicating good noise resistance without being overly conservative.

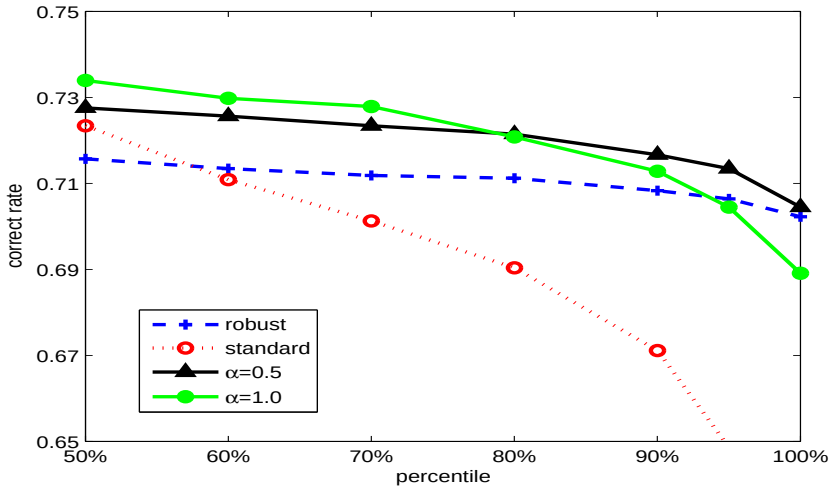


FIGURE 10.8. Percentile performance for Sonar data.

10.8. Chapter summary

This chapter extends the robust classification to a soft notion of robustness known as comprehensive robustness, and seeks to develop robust classifiers with controlled conservatism. The resulting classifier overcomes the conservatism inherent to the standard robust formulation and provides extra flexibility on handling the observation noise. We further show that any arbitrary convex constraint regularization, including the standard regularized SVM, is equivalent to a comprehensive robust classifier. This leads to a connection to convex risk measures, a notion widely used in decision making theory, from which we develop risk-constrained classifiers.

At a high level, our contribution is the introduction of a more geometric notion of hedging and controlling complexity (robust and comprehensive robust classifiers integrally depend on the uncertainty set and structure of the discount function) and the link to probabilistic notions of hedging, including chance constraints and convex risk constraints. We believe that in applications, particularly when distribution-free PAC-style bounds are pessimistic, the design flexibility of such a framework will yield superior performance. A central issue on the application front is to understand how to effectively use the additional degrees of freedom and flexibility since now we

are designing uncertainty sets and discount functions, rather than simply choosing regularization parameters that multiply a norm.

CHAPTER 11

Robust Dimensionality Reduction for High-Dimension Data

Similarly to Chapter 10 we propose new robust learning algorithm in this chapter. More precisely, we consider the dimensionality-reduction problem (finding a subspace approximation of observed data) for contaminated data in the high dimensional regime, where the the number of *observations* is of the same order of magnitude as the number of *variables* of each observation, and the data set contains some (arbitrarily) corrupted observations. We propose a High-dimensional Robust Principal Component Analysis (HR-PCA) algorithm that is tractable, robust to contaminated points, and easily kernelizable. The resulting subspace has a bounded deviation from the desired one, and unlike ordinary PCA algorithms, achieves optimality in the limit case where the proportion of corrupted points goes to zero. Part of the material in this chapter appears in [**164**].

11.1. Introduction

The analysis of very high dimensional data – data sets where the dimensionality of each observation is comparable to or even larger than the number of observations – has drawn increasing attention in the last few decades [**57, 93**]. Today, it is common practice that observations on individual instances are curves, spectra, images or even

movies, where a single observation has dimensionality ranging from thousands to billions. Practical high dimensional data examples include DNA Microarray data, financial data, climate data, web search engine, and consumer data. In addition, the nowadays standard "Kernel Trick" [**133**], a pre-processing routine which non-linearly maps the observations into a (possibly infinite dimensional) Hilbert space, transforms virtually every data set to a high dimensional one. Efforts of extending traditional statistical tools (designed for low dimensional case) into this high-dimensional regime are generally unsuccessful. This fact has stimulated research on formulating fresh data-analysis techniques able to cope with such a "dimensionality explosion."

In this chapter, we consider a high-dimensional counterpart of Principal Component Analysis (PCA) that is robust to the existence of corrupted or contaminated data. In our setup, a low dimensional Gaussian signal is mapped to a very high dimensional space, *after which point* high-dimensional Gaussian noise is added, to produce points that no longer lie on a low dimensional subspace. Then, *a constant fraction of the points are arbitrarily corrupted* in a perhaps non-probabilistic manner. We refrain from calling these "outliers" to emphasize that their distribution is entirely arbitrary, rather than from the tails of any particular distribution, e.g., the noise distribution. We call the remaining points "authentic."

Work on PCA dates back as early as [**119**], and has become one of the most important techniques for data compression and feature extraction. It is widely used in statistical data analysis, communication theory, pattern recognition, and image processing [**94**]. The standard PCA algorithm constructs the optimal (in a least-square sense) subspace approximation to observations by computing the eigenvectors or Principal Components (PCs) of the sample covariance or correlation matrix.

It is well known that such analysis is extremely sensitive to outlying, or corrupted, measurements. Indeed, one aberrant observation is sufficient to cause arbitrarily large changes in the covariance or correlation matrix, and hence the corresponding PCs.

In the low-dimensional regime where the observations significantly outnumber the variables of each observation, several robust PCA algorithms have been proposed

(e.g.,[**51, 177, 178, 45, 47, 48, 44**]). These algorithms can be roughly divided into two classes: (i) performing a standard PCA on a robust estimation of the covariance or correlation matrix; (ii) maximizing (over all unit-norm $\mathbf{w}$) some $r(\mathbf{w})$ that is a robust estimate of the variance of univariate data obtained by projecting the observations onto direction $\mathbf{w}$. Both approaches encounter serious difficulties when applied to high-dimensional data-sets:

- There are not enough observations to robustly estimate the covariance or correlations matrix. For example, the widely-used MVE estimator [**128**], which treats the Minimum Volume Ellipsoid that covers half of the observations as the covariance estimation, is ill-posed in the high-dimensional case. Indeed, to the best of our knowledge, the assumption that observations far outnumber dimensionality seems crucial for those robust variance estimators to achieve statistical consistency.
- Unlike standard PCA that has a polynomial computation time, the maximization of $r(\mathbf{w})$ is generally a non-convex problem, and becomes extremely hard to solve or approximate as the dimensionality of $\mathbf{w}$ increases. In fact, the number of the local maxima grows so fast that it is effectively impossible to find a sufficiently good solution using gradient-based algorithms with random re-initialization.

In contrast to these approaches, we propose a High-dimensional Robust PCA (HR-PCA) algorithm that takes into account the inherent difficulty in analyzing the high dimensional data. In particular, the algorithm we propose here is tractable, provably robust to corrupted points, easily kernelizable, and asymptotically optimal.

The proposed algorithm takes an "actor-critic" form: we apply standard PCA in order to find a set of candidate directions. These directions are then subjected to a hypothesis test, that uses a computationally efficient one-dimensional robust variance estimate. This hypothesis test determines if the variance is due to corrupted data, or indeed the "authentic" points. In case of the latter, the algorithm has found a true PC. In case of the former, we use a randomized point removal scheme, that

guarantees quick termination of our algorithm with deviation guarantees on the PCs it ultimately reports, from the true PCs.

One notable difference between this work and previous robust PCA work is how we measure the robustness of an algorithm. The traditional robustness measurement is the so-called "breakdown point" [**90**], i.e., the percentage of corrupted points that can make the output of the algorithm *arbitrarily* bad. This is an indirect measurement: except that the error is not unlimited, there is no guarantee that the output is *good enough*, even when the algorithm does not break down. In contrast, we directly investigate the "robust performance" of the algorithm, i.e., the performance gap between the output of the algorithm and the optimum, as a function of the fraction of corrupted points. Therefore, such a direct measurement provides an explicit guarantee of the performance of the algorithm, which we regard to be of importance in practice.

The chapter is organized as follows: In Section 11.2 we present the setup of the problem, the hypothesis test, and then the HR-PCA algorithm including the randomized point removal scheme. Based on some technical results established in Section 11.3, we show the validity of HR-PCA in Section 11.4 by providing a bound on the probability that our algorithm removes a corrupted point at any given iteration, and then using this to bound the running time of the algorithm, and finally to give finite sample and asymptotic performance guarantees. Section 11.5 is devoted to the kernelization of HR-PCA. We provide some numerical experiment results in Section 11.6.

Notation: Capital letters and boldface letters are used to denote matrices and vectors, respectively. $\Phi(\cdot)$ stands for the cumulative distribution function of $\mathcal{N}(0,1)$ and we let $\Phi^{-1}(c)$ be $-\infty$ and $+\infty$ for $c \leq 0$ and $c \geq 1$ respectively. $\Psi(\cdot)$ is the *Tracy-Widom* distribution of order one (c.f [**149, 93**]), implemented using a numerical lookup table. A $k \times k$ unit matrix is denoted by I_k. The largest eigenvalue of a symmetric matrix C is represented as $\lambda_{\max}(C)$. For $c \in \mathbb{R}$, $[c]^+ \triangleq \max(0, c)$.

As this chapter is notationally very heavy, we introduce a number of parameters to simplify long expressions, in the hopes of highlighting the key elements. While we introduce the parameters at the appropriate parts of the text, we keep the following convention, to facilitate the reader's job. Parameters which have a physical meaning, such as the largest singular value of a matrix, or the fraction of corrupted points, etc., all have a '$^-$'. Parameters which are introduced as "slack" factors in order to deal with finite-sample estimates (and go to zero asymptotically) all have a '$\hat{}$'. Finally, parameters synthesized from the two categories above, and introduced simply to shorten and simplify expressions, all have a '$\tilde{}$'.

11.2. HR-PCA: the algorithm

The algorithm of HR-PCA is presented in this section. We start with the mathematical setup of the problem in Section 11.2.1. As discussed in the Introduction, HR-PCA follows an "actor-critic" approach in which a robust univariate variance estimator serves as a hypothesis test, to evaluate the robustness of PCs found. We call this the "Sensitivity Test" and provide its formulation in Section 11.2.2. The HR-PCA algorithm is then given in Section 11.2.3.

11.2.1. Problem setup. We consider the following problem:

- The "authentic samples" $\mathbf{z}_1, \ldots, \mathbf{z}_t \in \mathbb{R}^m$ are generated by $\mathbf{z}_i = A\mathbf{x}_i + \mathbf{n}_i$, where $\mathbf{x}_i$ (the "signal") and $\mathbf{n}_i$ (the "noise") are independent realizations of random variables $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, I_d)$ and $\mathbf{n} \sim \mathcal{N}(\mathbf{0}, I_m)$ respectively. The matrix $A \in \mathbb{R}^{m\times d}$ is unknown.
- The corrupted data are denoted $\mathbf{o}_1, \ldots, \mathbf{o}_{n-t} \in \mathbb{R}^m$ and they are arbitrary (even maliciously chosen).
- We only observe the contaminated data set

$$\mathcal{Y} \triangleq \{\mathbf{y}_1 \ldots, \mathbf{y}_n\} = \{\mathbf{z}_1, \ldots, \mathbf{z}_t\} \bigcup \{\mathbf{o}_1, \ldots, \mathbf{o}_{n-t}\}.$$

  An element of $\mathcal{Y}$ is called a "point".

We denote the fraction of corrupted points by $\overline{\eta} \triangleq n - t/n$. In this chapter, we focus on the case where $n \sim m \gg d$ and $\lambda_{\max}(A^\top A) \gg 1$. That is, the number and dimensionality of observations are of the same magnitude, and much larger than the dimensionality of $\mathbf{x}$; the leading eigenvalue of $A^\top A$ is significantly larger than 1.

For a set of orthogonal vectors $\mathbf{v}_1, \ldots, \mathbf{v}_d$, performance is measured by the *Expressed Variance*

$$\text{E.V.} \triangleq \frac{\sum_{i=1}^{d} \mathbf{v}_i^\top A A^\top \mathbf{v}_i}{\sum_{i=1}^{d} \mathbf{v}_i^{*\top} A A^\top \mathbf{v}_i^*},$$

where $\{\mathbf{v}_1^*, \ldots, \mathbf{v}_d^*\}$ are the largest d eigenvectors of $AA^\top$ (i.e., the desired PCs). Notice that the maximum of E.V. equals 1, and is achieved by recovering the span of the true PCs $\{\mathbf{v}_1^*, \ldots, \mathbf{v}_d^*\}$. In addition, when $d = 1$, the Expressed Variance relates to another natural performance metric — the angle between $\mathbf{v}_1$ and $\mathbf{v}_1^*$ — since $E.V.(\mathbf{v}_1) = \cos^2(\angle(\mathbf{v}_1, \mathbf{v}_1^*))$ (see Figure 11.1). When $d > 1$, such geometric interpretation no longer exists since the angle between two subspaces is not well defined. The Expressed Variance represents the portion of signal $A\mathbf{x}$ being expressed by $\mathbf{v}_1, \ldots, \mathbf{v}_d$. Equivalently, $1 - E.V.$ is the reconstruction error of the signal.

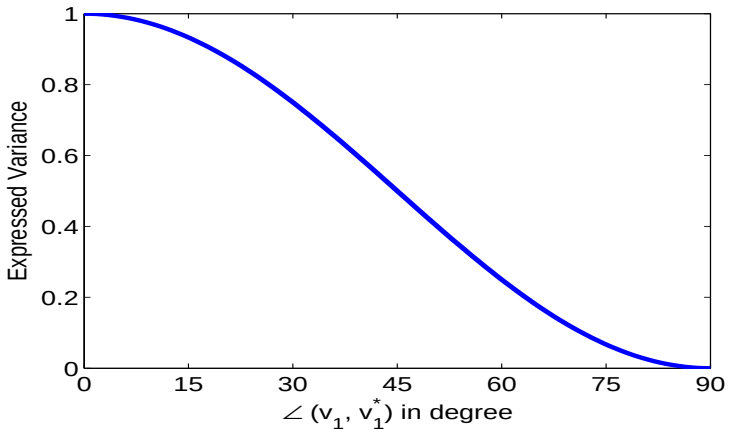


FIGURE 11.1. Expressed variance vs angle for d=1.

While we give finite-sample results, our main theorem gives the asymptotic performance of HR-PCA when *the dimension and the number of observations grow together* to infinity. To be more precise, our asymptotic setting is as follows. Suppose there

exists a sequence of sample sets $\{\mathcal{Y}(j)\}_j = \{\mathcal{Y}(1), \mathcal{Y}(2), \dots\}$, where for $\mathcal{Y}(j)$, $n(j)$, $m(j)$, $A(j)$, $d(j)$, etc., denote the corresponding values of the quantities defined above. Then the following must hold for some positive constants c_1, c_2:

$$\begin{aligned} &\lim_{j\to\infty} \frac{n(j)}{m(j)} = c_1; \quad d(j) \leq c_2; \quad m(j) \uparrow +\infty; \\ &\lambda_{\max}(A(j)^\top A(j)) \uparrow +\infty. \end{aligned} \tag{11.1}$$

11.2.2. Sensitivity test. We present the formulation of the "Sensitivity Test" in this subsection. This test is based on evaluating the "θ confidence interval" of a collection of scalar values, i.e., the shortest interval containing a θ fraction of the scalars. For unit-norm $\mathbf{w} \in \mathbb{R}^m$, let

$$\overline{l}_{\mathbf{w}} \triangleq l(0.5 + \frac{\overline{\eta}}{2}, \mathbf{w}^\top \mathbf{y}_1, \dots, \mathbf{w}^\top \mathbf{y}_n),$$

which is the $0.5 + \overline{\eta}/2$ confidence interval for the points projected on the direction $\mathbf{w}$. This confidence interval $\overline{l}_{\mathbf{w}}$ is an estimator of standard deviation robust to the existence of corrupted points [**90**]. Once PCA outputs directions of largest variance, for each direction we use this estimator to determine if the confidence interval is consistent with the observed variance, and hence if the variance is a phenomenon due to the authentic points, or due to the corrupted data. We refer to Figure 11.2 for an illustration. This figure illustrates a sensitivity test used often in practice, whereby

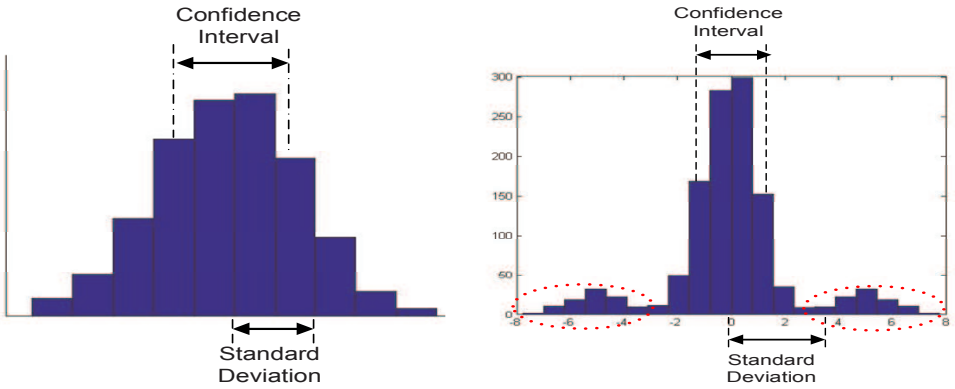


FIGURE 11.2. Illustration of the *Sensitivity Test*

the variance, σ^2, of a corrupted sample set along a direction $\mathbf{w}$ is deemed consistent with the confidence interval or not, according to the rule:

$$\begin{cases} \text{consistent} & \text{if } (1+\sqrt{\overline{\eta}})(1-\overline{\eta})\mathbf{H}_{\mathbf{w}} \geq \sigma^2 \\ \text{inconsistent} & \text{if } (1+\sqrt{\overline{\eta}})(1-\overline{\eta})\mathbf{H}_{\mathbf{w}} < \sigma^2 \end{cases},$$

where $\mathbf{H}_{\mathbf{w}} \triangleq \left(\frac{l_{\mathbf{w}}}{2\Phi^{-1}(0.75)}\right)^2$.

Because we are interested in a sensitivity test at each iteration, when some number s of the points have been removed in previous iterations, and because we require finite sample bounds in the sequel, we need a modification of the above sensitivity test that incorporates some slack factors. We note that asymptotically, our slack factors disappear, and our sensitivity test corresponds with the one given above. Before providing the exact form of the Sensitivity Test, we define the following terms to simplify the expressions. Again we use our convention whereby parameters with a physical meaning have a '$^-$', slack parameters which go to zero asymptotically have a '$\hat{}$' and parameters synthesized from the two categories above, and introduced simply to shorten and simplify expressions, all have a '$\tilde{}$'.

In the quantities below, the subscript 't' corresponds to the number of authentic points, and therefore is a quantity that in the asymptotic analysis will go to infinity. The subscript 'δ' will be a probability parameter we use in the sequel to control the probability of finite sample deviation results, and therefore will be taken to be a very small positive number.

$$\overline{\sigma}_1 \triangleq \sqrt{\lambda_{\max}(AA^\top)};$$
$$\overline{\lambda}_{t,\delta} \triangleq \frac{4}{1-\overline{\eta}} + \frac{2[\Psi^{-1}(1-\delta)]^+}{\sqrt{(1-\overline{\eta})t}}.$$

The first quantity above implicitly has a t-index, since the size of the matrix A is fixed to t. Note further that by assumption, $\overline{\sigma}_1$ goes to infinity as $t \to \infty$. The second quantity above bounds the variance of the noise realizations, as shown in

Theorem 11.2 below.

$$\hat{c}_{t,\delta} \triangleq \sqrt{\frac{6d^2}{t}\left(\ln 2d^2 + \ln\frac{1}{\delta}\right)};$$

$$\hat{h}_{t,\delta} \triangleq \sqrt{\frac{8d+8}{t}\ln\frac{t}{d+1} + \frac{8}{t}\ln\frac{8}{\delta}};$$

$$\hat{\phi}_{t,\delta} \triangleq 2\Phi^{-1}(0.75) - 2\Phi^{-1}(0.75 - \hat{h}_{t,\delta});$$

$$\tilde{v}_{t,\delta} \triangleq \hat{c}_{t,\delta}\overline{\sigma}_1^2 + [1 + \hat{c}_{t,\delta}/2]\sqrt{\overline{\lambda}_{t,\delta}}\overline{\sigma}_1 + \overline{\lambda}_{t,\delta};$$

$$\overline{O}_{\mathbf{w}} \triangleq \min_{c \geq \sqrt{2\overline{\lambda}_{t,\delta}}} \left\{ \frac{l_{\mathbf{w}} + \hat{\phi}_{t,\delta}\overline{\sigma}_1 + 2c}{2\Phi^{-1}(0.75 - \frac{\overline{\lambda}_{t,\delta}}{2c^2})} \right\};$$

$$\overline{H}_{\mathbf{w}} \triangleq \overline{O}_{\mathbf{w}}^2 + \tilde{v}_{t,\delta}.$$

Finally, we can define our sensitivity test.

Definition 11.1. *If s points have been removed, and the empirical variance in a direction $\mathbf{w}$ is σ^2, then the* Sensitivity Test $\mathbb{H}$ *is defined as*

$$\mathbb{H}(\mathbf{w}, \sigma, s) \triangleq \begin{cases} \textit{consistent}, & \textit{if } \frac{(1+\sqrt{\overline{\eta}})(1-\overline{\eta})n}{n-s}\overline{H}_{\mathbf{w}} \geq \sigma^2; \\ \textit{inconsistent}, & \textit{if } \frac{(1+\sqrt{\overline{\eta}})(1-\overline{\eta})n}{n-s}\overline{H}_{\mathbf{w}} < \sigma^2. \end{cases}$$

Note that $\overline{O}$ takes the place of the term $\frac{l_{\mathbf{w}}}{2\Phi^{-1}(0.75)}$ in the expression given for the first sensitivity test above.

11.2.3. Main algorithm. The main algorithm of HR-PCA is as given below.

Algorithm 11.1. **HR-PCA**

Input: Contaminated sample-set $\mathcal{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_n\} \subset \mathbb{R}^m$, $\overline{\eta}$, d, δ, $\overline{\sigma}_1$.

Output: $\mathbf{v}_1, \ldots, \mathbf{v}_d$.

Algorithm:

(1) Let $\hat{\mathbf{y}}_i := \mathbf{y}_i$ for $i = 1, \ldots n$; $s := 0$.

(2) Compute the empirical variance matrix

$$\hat{\Sigma} := \frac{1}{n-s}\sum_{i=1}^{n-s} \hat{\mathbf{y}}_i \hat{\mathbf{y}}_i^\top.$$

(3) Let $\hat{\sigma}_1^2, \ldots, \hat{\sigma}_d^2$ and $\mathbf{v}_1, \ldots, \mathbf{v}_d$ be the d largest eigenvalues and the corresponding eigenvectors of $\hat{\Sigma}$.

(4) If there is a $j \in \{1, \ldots, d\}$ such that *Sensitivity Test* $\mathbb{H}(\mathbf{v}_j, \hat{\sigma}_j, s)$ fails, do the following:

- randomly remove a point from $\{\hat{\mathbf{y}}_i\}_{i=1}^{n-s}$ according to

$$\Pr(\hat{\mathbf{y}}_i \text{ is removed}) \propto (\mathbf{v}_j^\top \hat{\mathbf{y}}_i)^2;$$

- denote the remaining points by $\{\hat{\mathbf{y}}_i\}_{i=1}^{n-s-1}$;
- $s := s + 1$, go to Step 2.

(5) Output $\mathbf{v}_1, \ldots, \mathbf{v}_d$. End.

In each iteration, HR-PCA finds a set of directions maximizing the empirical variance of the points (i.e., of authentic and corrupted samples). If all directions pass the Sensitivity Test, then the variances of "authentic samples" projected on them must be *close to being the largest*, and hence the chosen directions are close to the true PCs. If the Sensitivity Test fails, then the corrupted points must have a large influence on the variance in this direction. In this case, the PC is not selected, and a point is removed in proportion to its variance. We show that this proportional removal guarantees a minimum probability that a corrupted point will be removed.

The correctness of HR-PCA is shown in the following sections. We outline here the main theorem providing an asymptotic lower bound of the performance (illustrated in Figure 11.3). This is based on a finite-sample result, which we state and prove in Section 11.4.3.

THEOREM 11.1. *If Equation (11.1) holds, and* $\overline{\eta}(j) \to \overline{\eta}^*$*, then the following holds in probability with* $j \uparrow \infty$*,*

$$\frac{\sum_{q=1}^{d} \mathbf{v}_q(j)^\top (A(j)A(j)^\top) \mathbf{v}_q(j)}{\sum_{q=1}^{d} \mathbf{v}_q^*(j)^\top (A(j)A(j)^\top) \mathbf{v}_q^*(j)} \geq \frac{\int_{-\tilde{\zeta}^*}^{\tilde{\zeta}^*} \frac{x^2}{\sqrt{2\pi}} \exp(\frac{-x^2}{2}) dx}{1 + \sqrt{\overline{\eta}^*}} \left(\frac{\Phi^{-1}(0.75)}{\Phi^{-1}(0.75 + \frac{\overline{\eta}^*}{2 - 2\overline{\eta}^*})} \right)^2, \tag{11.2}$$

where $\tilde{\zeta}^* \triangleq \Phi^{-1}\left(1 - \frac{\sqrt{\overline{\eta}^*}}{1-\sqrt{\overline{\eta}^*}}\right)$.

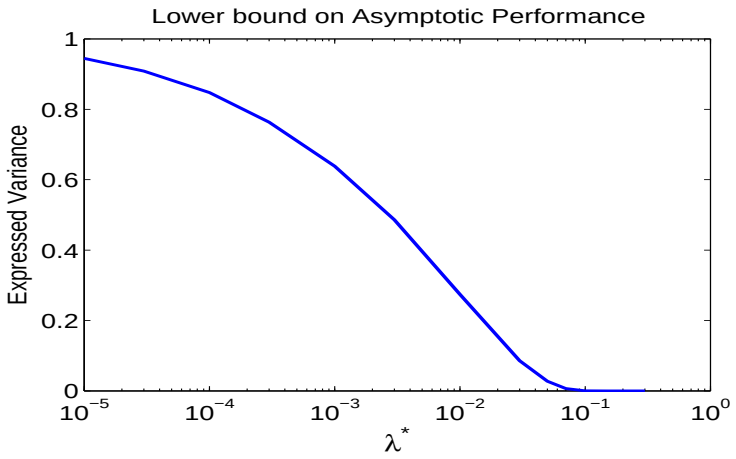


FIGURE 11.3. Lower bound of asymptotic performance of HR-PCA.

REMARK 11.1. If $\overline{\eta}(j) \downarrow 0$ (e.g., there are a fixed *number* of corrupted points), then the right-hand-side of Inequality (11.2) equals 1, i.e., HR-PCA is asymptotically optimal. This is in contrast to PCA, where the existence of *even a single* corrupted point is sufficient to bound the output *arbitrarily* away from the optimum.

11.3. Technical results: uniform convergence

This section is devoted to establishing the uniform (w.r.t. all directions $\mathbf{w} \in \mathbb{R}^m$) convergence properties of the sample variance and the "confidence interval" for authentic samples $\{\mathbf{z}_1, \ldots, \mathbf{z}_t\}$ (Theorems 11.3 and 11.4 respectively). These results are used as technical lemmas in proving the validity of HR-PCA in Section 11.4. Due to space constraints, all proofs in this section are deferred to Section 11.9.

Consider the following events:

Condition (A): $\{\lambda_{\max}(1/t\sum_{i=1}^{t}\mathbf{n}_i\mathbf{n}_i^\top) \le \overline{\lambda}_{t,\delta}\}$

Condition (B):

$$\left\{\sup_{\mathbf{w}\in\mathbb{R}^d,\,\|\mathbf{w}\|=1}\left|\mathbf{w}^\top(\frac{1}{t}\sum_{i=1}^{t}\mathbf{x}_i\mathbf{x}_i^\top - I_d)\mathbf{w}\right| \le \hat{c}_{t,\delta}\right\};$$

Condition (C):

$$\left\{\forall\|\mathbf{w}\|_2=1, \forall\theta\in[0,1]:\ 2\Phi^{-1}\left(\frac{1+\theta}{2}-\hat{h}_{t,\delta}\right)\right.$$
$$\left. \le l(\theta, \mathbf{w}^\top\mathbf{x}_1,\ldots,\mathbf{w}^\top\mathbf{x}_t) \le 2\Phi^{-1}\left(\frac{1+\theta}{2}+\hat{h}_{t,\delta}\right)\right\}.$$

THEOREM 11.2. *For sufficiently large t and m:*

(a) Condition (A) holds with probability at least $1-\delta$;

(b) Condition (B) holds with probability at least $1-\delta$;

(c) Condition (C) holds with probability at least $1-\delta$.

The validity of Theorem 11.2(a) follows from a lemma in [**93**]; Theorem 11.2(b) and Theorem 11.2(c) are finite dimensional uniform convergence results that follow from VC-dimension style arguments. The detailed proof is deferred to Section 11.9.

The next two theorems give analogs of Condition (B) and Condition (C) but for the high dimensional points.

THEOREM 11.3. *Under Conditions (A) and (B), and $n \ge 4$, the average variance along direction $\mathbf{w}$, of the authentic points, has distance from the size of $(AA^\top + I)$ in the $\mathbf{w}$ direction bounded as follows:*

$$\begin{aligned}&\sup_{\mathbf{w}\in\mathbb{R}^m,\|\mathbf{w}\|_2=1}\left|\mathbf{w}^\top\left(\frac{1}{t}\sum_{i=1}^{t}\mathbf{z}_i\mathbf{z}_i^\top\right)\mathbf{w} - \mathbf{w}^\top(AA^\top+I_m)\mathbf{w}\right| \\ \le&\hat{c}_{t,\delta}\overline{\sigma}_1^2 + \left[1+\frac{\hat{c}_{t,\delta}}{2}\right]\sqrt{\overline{\lambda}_{t,\delta}}\overline{\sigma}_1 + \overline{\lambda}_{t,\delta} - 1.\end{aligned} \tag{11.3}$$

THEOREM 11.4. *Under Conditions (A) and (C), for any* $\|\mathbf{w}\|_2 = 1$ *and* $\theta \in (0,1)$*, the* θ*-confidence interval of all the authentic points projected on direction* $\mathbf{w}$ *is sandwiched as follows:*

$$\begin{aligned}
&\sup_{c>0}\left\{2\Phi^{-1}(\frac{1+\theta}{2}-\frac{\overline{\lambda}_{t,\delta}}{2c^2})\sqrt{\mathbf{w}^\top AA^\top\mathbf{w}}-(2\Phi^{-1}(\frac{1+\theta}{2})-2\Phi^{-1}(\frac{1+\theta}{2}-\hat{h}_{t,\delta}))\overline{\sigma}_1-2c\right\}\\
&\le l(\theta,\mathbf{w}^\top\mathbf{z}_1,\ldots,\mathbf{w}^\top\mathbf{z}_t)\\
&\le\inf_{c>0}\left\{2\Phi^{-1}(\frac{1+\theta}{2}+\frac{\overline{\lambda}_{t,\delta}}{2c^2})\sqrt{\mathbf{w}^\top AA^\top\mathbf{w}}+(2\Phi^{-1}(\frac{1+\theta}{2}+\hat{h}_{t,\delta})-2\Phi^{-1}(\frac{1+\theta}{2}))\overline{\sigma}_1+2c\right\}.
\end{aligned}\tag{11.4}$$

Note that these bounds are, up to addition of appropriate slack factors, the θ-confidence bounds for the original low-dimensional points $\{\mathbf{x}_i\}$ as given in Condition C, elongated by $\sqrt{\mathbf{w}AA^\top\mathbf{w}}$.

11.4. Correctness of HR-PCA

Based on the results presented in the previous section, we show in this section the correctness of HR-PCA, i.e., with a high probability, the subspace spanned by the output $\mathbf{v}_1,\ldots,\mathbf{v}_d$ is a good approximation (in the sense of the Expressed Variance) of that spanned by $\mathbf{v}_1^*,\ldots,\mathbf{v}_d^*$.

In Section 11.4.1 we lower bound the probability of removing a corrupted point in each iteration. We then show that the number of iterations is small with high probability in Section 11.4.2. We complete the argument in Section 11.4.3 by showing that when HR-PCA stops within a small number of iterations, the PCs found are good approximations of the desired ones.

Throughout this section, we assume without explicitly stating it in each theorem that Conditions (A), (B) and (C) hold simultaneously. As shown in Section 11.3, this occurs with probability is at least $1-3\delta$. We further assume $n \ge 4$.

11.4.1. Probability of removing a corrupted point. In this section, we lower bound the probability of removing a corrupted point in each iteration of HR-PCA.

THEOREM 11.5. *If the Sensitivity Test fails (i.e., it returns "inconsistent") in the s^{th} iteration, then*

$$\Pr(\text{The next removed point is corrupted}) > \frac{\sqrt{\overline{\eta}}}{1+\sqrt{\overline{\eta}}}.$$

To prove Theorem 11.5, we need the following lemma.

LEMMA 11.6. *For all $\|\mathbf{w}\|_2 = 1$, $\frac{1}{t}\sum_{i=1}^{t}(\mathbf{w}^\top \mathbf{z}_i)^2 \le \overline{H}_{\mathbf{w}}$.*

PROOF. By definition, an interval with length $l_{\mathbf{w}}$ covers $(0.5+\overline{\eta}/2)n$ points in $\mathcal{Y}$, which implies that it covers at least $(0.5-\overline{\eta}/2)n = 0.5t$ authentic samples. Therefore,

$$l(0.5, \mathbf{w}^\top \mathbf{z}_1, \ldots, \mathbf{w}^\top \mathbf{z}_t) \le l_{\mathbf{w}}. \tag{11.5}$$

Now, from Theorem 11.4, when (B) and (C) hold, we have for all θ and $\|\mathbf{w}\|_2 = 1$,

$$\begin{aligned}
&2\Phi^{-1}(\frac{1+\theta}{2} - \frac{\overline{\lambda}_{t,\delta}}{2c^2})\sqrt{\mathbf{w}^\top AA^\top \mathbf{w}} \\
\le& l(\theta, \mathbf{w}^\top \mathbf{z}_1, \ldots, \mathbf{w}^\top \mathbf{z}_t) + (2\Phi^{-1}(\frac{1+\theta}{2}) - 2\Phi^{-1}(\frac{1+\theta}{2} - \hat{h}_{t,\delta}))\overline{\sigma}_1 + 2c.
\end{aligned}$$

Taking $\theta = 0.5$, by applying (11.5) we get for all $c \ge \sqrt{2\overline{\lambda}_t}$,

$$\sqrt{\mathbf{w}^\top AA^\top \mathbf{w}} \le (l_{\mathbf{w}} + \hat{\phi}_{t,\delta}\overline{\sigma}_1 + 2c)/2\Phi^{-1}(0.75 - \frac{\overline{\lambda}_{t,\delta}}{2c^2}).$$

Minimizing over c implies $\mathbf{w}^\top(AA^\top + I_m)\mathbf{w} \le \overline{O}_{\mathbf{w}}^2 + 1$. Applying Theorem 11.3 completes the proof. □

PROOF OF THEOREM 11.5. First recall that our point removal strategy implies

$$\Pr(\text{The next removed point is corrupted}) = \frac{\sum_{\mathbf{y}_i \in \hat{I}_s}(\mathbf{v}_q{}^\top \mathbf{y}_i)^2}{\sum_{\mathbf{y}_i \in I_s}(\mathbf{v}_q{}^\top \mathbf{y}_i)^2 + \sum_{\mathbf{y}_i \in \hat{I}_s}(\mathbf{v}_q{}^\top \mathbf{y}_i)^2},$$

where I_s and $\hat{I}_s$ denote the set of remaining authentic samples and remaining corrupted points, respectively. We prove that

$$\sqrt{\overline{\eta}}\sum_{\mathbf{y}_i \in I_s}(\mathbf{v}_q{}^\top \mathbf{y}_i)^2 < \sum_{\mathbf{y}_i \in \hat{I}_s}(\mathbf{v}_q{}^\top \mathbf{y}_i)^2,$$

which will conclude the proof.

By definition, if the Sensitivity Test fails it must be that for some $q \in \{1, \ldots, d\}$

$$(1 + \sqrt{\overline{\eta}})(1 - \overline{\eta})n\overline{H}_{\mathbf{v}_q}/(n - s) < \hat{\sigma}_q^2. \tag{11.6}$$

At the s^{th} iteration, there are $n - s$ remaining points. We have $|I_s| + |\hat{I}_s| = n - s$, and $I_s \subseteq \{\mathbf{z}_1, \ldots, \mathbf{z}_t\}$. Therefore,

$$\sum_{\mathbf{y}_i \in I_s} (\mathbf{v}_q{}^\top \mathbf{y}_i)^2 \leq \sum_{i=1}^{t} (\mathbf{v}_q{}^\top \mathbf{z}_i)^2 \leq (1 - \overline{\eta})n\overline{H}_{\mathbf{v}_q}, \tag{11.7}$$

where the last inequality follows from Lemma 11.6. Furthermore,

$$\begin{aligned} \hat{\sigma}_q^2 &= \frac{1}{n-s} \sum_{\mathbf{y}_i \in I_s \bigcup \hat{I}_s} (\mathbf{v}_q{}^\top \mathbf{y}_i)^2 \\ \implies \quad (n-s)\hat{\sigma}_q^2 &= \sum_{\mathbf{y}_i \in I_s} (\mathbf{v}_q{}^\top \mathbf{y}_i)^2 + \sum_{\mathbf{y}_i \in \hat{I}_s} (\mathbf{v}_q{}^\top \mathbf{y}_i)^2. \end{aligned} \tag{11.8}$$

Substituting (11.7) and (11.8) into Inequality (11.6) leads to

$$\begin{aligned} &(1 + \sqrt{\overline{\eta}}) \sum_{\mathbf{y}_i \in I_s} (\mathbf{v}_q{}^\top \mathbf{y}_i)^2 < \sum_{\mathbf{y}_i \in I_s} (\mathbf{v}_q{}^\top \mathbf{y}_i)^2 + \sum_{\mathbf{y}_i \in \hat{I}_s} (\mathbf{v}_q{}^\top \mathbf{y}_i)^2 \\ \Rightarrow \quad &\sqrt{\overline{\eta}} \sum_{\mathbf{y}_i \in I_s} (\mathbf{v}_q{}^\top \mathbf{y}_i)^2 < \sum_{\mathbf{y}_i \in \hat{I}_s} (\mathbf{v}_q{}^\top \mathbf{y}_i)^2. \end{aligned}$$

□

Theorem 11.5 implies that if the Sensitivity Tests fails, then there is at least one corrupted point remaining.

Since the probability of removing a corrupted point at any given iteration of the algorithm is $\sqrt{\overline{\eta}}/(1 + \sqrt{\overline{\eta}})$, each iteration decreases the "expected number of corrupted points" by that amount. In the next section, we bound the probability that the algorithm fails to terminate before some particular iteration. For this we use twice the number of corrupted points divided by the expected reduction at a given iteration:

$$\bar{s}_0 \triangleq 2 \left(\frac{1 + \sqrt{\overline{\eta}}}{\sqrt{\overline{\eta}}} \right) \overline{\eta} n.$$

11.4.2. Number of iterations. In this section, we show that with high probability, HR-PCA terminates quickly: within s_0 iterations. The key to the proof is the previous theorem, that each step removes a corrupted point with probability at least $\sqrt{\overline{\eta}}/(1+\sqrt{\overline{\eta}})$. If the event of corrupted point removal at subsequent iterations were independent, then the expected number of points removed by s iterations would be $s \cdot \sqrt{\overline{\eta}}/(1+\sqrt{\overline{\eta}})$, and since there are only $\overline{\eta}n$ corrupted points in total, the result would be straightforward. Instead, we must use a Martingale argument to arrive at the desired result.

By definition, HR-PCA terminates at step s if the Sensitivity Test succeeds after $s-1$ points have been removed. Let random variable $V(s)$ denote the number of corrupted points removed up to iteration s, inclusive. Define the following stochastic process:

$$X_s \triangleq \begin{cases} V(T) - \frac{\sqrt{\overline{\eta}}(T-1)}{1+\sqrt{\overline{\eta}}}, & \text{HR-PCA stoped at } T \leq s; \\ V(s) - \frac{\sqrt{\overline{\eta}}s}{1+\sqrt{\overline{\eta}}}, & \text{Otherwise.} \end{cases}$$

Let $\mathcal{F}_s$ be the filtration generated by the set of events up to iteration s. Hence, X_s is measurable w.r.t. $\mathcal{F}_s$.

LEMMA 11.7. *$\{X_s, \mathcal{F}_s, s = 1, \ldots, n\}$ is a sub-martingale.*

PROOF. Observe that $X_s \in \mathcal{F}_s$ by definition of $\mathcal{F}_s$. We show that $\mathbb{E}(X_s|\mathcal{F}_{s-1}) \geq X_{s-1}$ by enumerating the following three cases.

Case 1: the algorithm has not terminated up to step $s-1$, and the hypothesis test of the s^{th} iteration fails. Thus by Theorem 11.5,

$$\mathbb{E}(X_s - X_{s-1}|\mathcal{F}_{s-1}) = \Pr(\text{The next removed point is corrupted}) - \frac{\sqrt{\overline{\eta}}}{1+\sqrt{\overline{\eta}}} \geq 0.$$

Case 2: the algorithm has not terminated up to step $s-1$, and the hypothesis test of the s^{th} iteration succeeds. Thus, the algorithm terminates at step s, i.e., no extra point will be removed. Hence $V(s) = V(s-1)$. By definition of X we have $X_s = X_{s-1}$ in this case.

Case 3: the algorithm terminates at step $T \leq s-1$. Observe that $X_s = X_{s-1}$ in this case.

Combining all three cases shows that $\mathbb{E}(X_s|\mathcal{F}_{s-1}) \geq X_{s-1}$, which proves the lemma. □

THEOREM 11.8. *Denote* $\kappa = \sqrt{\overline{\eta}}$. *For all* $s \geq (1+\kappa)\lambda n/\kappa$, *we have*

$$\Pr(\text{the algorithm does not terminate up to step } s) \leq \exp\Big(\frac{-(\lambda n - \frac{\kappa s}{1+\kappa})^2}{8s}\Big).$$

PROOF. We prove the theorem by exploiting the deviation bound of a martingale process. Let $y_s \triangleq X_s - X_{s-1}$, where recall that $X_0 = 0$. Consider the following sequence:

$$y'_s \triangleq y_s - \mathbb{E}(y_s|y_1, \cdots, y_{s-1}).$$

Observe that $\{y'_s\}$ is a martingale difference process w.r.t. $\{\mathcal{F}_s\}$. Since $\{X_s\}$ is a sub-martingale, $\mathbb{E}(y_s|y_1, \cdots, y_{s-1}) \geq 0$ a.s. Therefore, the following holds a.s.,

$$X_s = \sum_{i=1}^{s} y_i = \sum_{i=1}^{s} y'_i + \sum_{i=1}^{s} \mathbb{E}(y_i|y_1, \cdots, y_{i-1}) \geq \sum_{i=1}^{s} y'_i. \tag{11.9}$$

By definition, $|y_s| \leq 1$, and hence $|y'_s| \leq 2$. Now for any $\theta > 0$,

$$\begin{aligned}
&\mathbb{E}\big\{\exp(\theta \sum_{i=1}^{s}(-y'_i))\big\} \\
&= \mathbb{E}\big\{\exp(\theta \sum_{i=1}^{s-1}(-y'_i))\big\}\mathbb{E}(\theta - y'_s| - y'_1, \cdots, -y'_{s-1}) \\
&\leq \mathbb{E}\big\{\exp(\theta \sum_{i=1}^{s-1}(-y'_i))\big\}\exp(\theta^2| - y'_s|^2/2) \\
&= \mathbb{E}\big\{\theta \exp(\sum_{i=1}^{s-1}(-y'_i))\big\}\exp(2\theta^2).
\end{aligned}$$

The inequality follows from Lemma 8.1 of [**54**]. By iteration we have

$$\mathbb{E}\big\{\exp(\theta \sum_{i=1}^{s}(-y'_i))\big\} \leq \exp(2s\theta^2).$$

Using the Markov inequality, we have that for any $\epsilon > 0$,

$$\Pr(\sum_{i=1}^{s}(-y_i') \geq s\epsilon) \leq \exp(2s\theta^2 - \theta s\epsilon).$$

Taking the minimum over θ of the right hand side and applying (11.9) leads to

$$\Pr(X_s \leq -s\epsilon) \leq \exp(-s\epsilon^2/8).$$

Now note that if the algorithm does not terminate up to step s, we have $X_s \leq \lambda n - \kappa s/(1+\kappa)$, because there are only λn outliers. Thus we have for all $s \geq (1+\kappa)\lambda n/\kappa$,

$$\begin{aligned}
&\Pr(\text{the algorithm does not terminate up to step } s)\\
\leq &\Pr\Big(X_s \leq \lambda n - \frac{\kappa s}{1+\kappa}\Big)\\
\leq &\exp\Big(\frac{-(\lambda n - \frac{\kappa s}{1+\kappa})^2}{8s}\Big),
\end{aligned}$$

which establishes the theorem. □

The probability that the algorithm does not terminate up to $\bar{s}_0 = 2(1+\sqrt{\overline{\eta}})\overline{\eta}n/\sqrt{\overline{\eta}}$ is hence bounded by

$$e^{-\left(\frac{n\sqrt{\overline{\eta}}\overline{\eta}}{8(1+\sqrt{\overline{\eta}})}\right)},$$

which goes to zero exponentially in n.

11.4.3. Deviation bound of output PCs. In this section, we show that when HR-PCA terminates, i.e., when all $\mathbf{v}_1, \ldots, \mathbf{v}_d$ pass the Sensitivity Test, the output is close to optimal, in the sense that we bound the distance of $\sum_{i=1}^{d} \mathbf{v}_i^\top AA^\top \mathbf{v}_i$ to $\sum_{i=1}^{d} \mathbf{v}_i^{*\top} AA^\top \mathbf{v}_i^*$, where recall the $\{\mathbf{v}_i^*\}$ are the true PCs. We state some technical lemmas, then prove a finite sample result (Theorem 11.12) and finally go on to prove the asymptotic result stated above in Theorem 11.1. The proofs are omitted due to space constraints, and deferred to the appendix. To simplify the expressions, we

define the following terms:

$$\bar{\theta} \triangleq \frac{1+\overline{\eta}}{2-2\overline{\eta}};$$
$$\hat{\psi}_{t,\delta} \triangleq 2\Phi^{-1}(\frac{1+\bar{\theta}}{2}+\hat{h}_{t,\delta}) - 2\Phi^{-1}(\frac{1+\bar{\theta}}{2});$$
$$\tilde{\zeta}_{t,\delta} \triangleq \Phi^{-1}\left(1-\frac{\bar{s}_0}{2t}-\sqrt{\frac{1}{8t}\log\frac{d}{\delta}}\right).$$

We sometimes drop the subscript of $\tilde{\zeta}_{t,\delta}$ and simply write $\tilde{\zeta}$. It should be understood that $\tilde{\zeta}$ depends on t and δ.

LEMMA 11.9. *For all* $\|\mathbf{w}\| \leq 1$, $l_{\mathbf{w}} \leq l(\bar{\theta}, \mathbf{w}^\top\mathbf{z}_1, \ldots, \mathbf{w}^\top\mathbf{z}_t)$.

PROOF. This follows from the definition of $\overline{\theta}$. □

LEMMA 11.10. *For all unit-norm* $\mathbf{w}$*, the following holds,*

$$\sqrt{\mathbf{w}^\top AA^\top\mathbf{w}} \geq \max_{c\geq\sqrt{2\overline{\lambda}_t}}\left\{\frac{\Phi^{-1}(0.75-\frac{\overline{\lambda}_{t,\delta}}{2c^2})\overline{O}_{\mathbf{w}} - 2c - (\hat{\phi}_{t,\delta}+\hat{\psi}_{t,\delta})\frac{\overline{\sigma}_1}{2}}{\Phi^{-1}(\frac{1+\bar{\theta}}{2}+\frac{\overline{\lambda}_{t,\delta}}{2c^2})}\right\}.$$

PROOF. Let c_0 maximize the right-hand-side. From Lemma 11.9 and Theorem 11.4, we have the following inequality under Conditions (B) and (C),

$$l_{\mathbf{w}} \leq l(\overline{\theta}, \mathbf{w}^\top\mathbf{z}_1, \cdots, \mathbf{w}^\top\mathbf{z}_t) \leq 2\Phi^{-1}(\frac{1+\overline{\theta}}{2}+\frac{\overline{\lambda}_{t,\delta}}{2c_0^2})\sqrt{\mathbf{w}^\top AA^\top\mathbf{w}} + \hat{\psi}_{t,\delta}\overline{\sigma}_1 + 2c_0.$$

Rearranging terms leads to

$$\begin{aligned}
\sqrt{\mathbf{w}^\top AA^\top\mathbf{w}} &\geq \frac{l_{\mathbf{w}} - \hat{\psi}_{t,\delta} - 2c_0}{2\Phi^{-1}(\frac{1+\overline{\theta}}{2}+\frac{\overline{\lambda}_{t,\delta}}{2c_0^2})} \\
&= \frac{\left\{\frac{l_{\mathbf{w}}+\hat{\phi}_{t,\delta}\overline{\sigma}_1+2c_0}{2\Phi^{-1}(0.75-\frac{\overline{\lambda}_{t,\delta}}{2c_0^2})}\right\}\times 2\Phi^{-1}(0.75-\frac{\overline{\lambda}_{t,\delta}}{2c_0^2}) - \hat{\phi}_{t,\delta}\overline{\sigma}_1 - 2c_0 - \hat{\psi}_{t,\delta}\overline{\sigma}_1 - 2c_0}{2\Phi^{-1}(\frac{1+\overline{\theta}}{2}+\frac{\overline{\lambda}_{t,\delta}}{2c_0^2})} \\
&\geq \frac{\overline{O}_{\mathbf{w}}\times 2\Phi^{-1}(0.75-\frac{\overline{\lambda}_{t,\delta}}{2c_0^2}) - \hat{\phi}_{t,\delta}\overline{\sigma}_1 - 2c_0 - \hat{\psi}_{t,\delta}\overline{\sigma}_1 - 2c_0}{2\Phi^{-1}(\frac{1+\overline{\theta}}{2}+\frac{\overline{\lambda}_{t,\delta}}{2c_0^2})}.
\end{aligned}$$

The last inequality holds because by definition of $\overline{O}_{\mathbf{w}}$,

$$\overline{O}_{\mathbf{w}} \leq \Big\{ \frac{l_{\mathbf{w}} + \hat{\phi}_{t,\delta}\overline{\sigma}_1 + 2c_0}{2\Phi^{-1}(0.75 - \frac{\overline{\lambda}_{t,\delta}}{2c_0^2})} \Big\},$$

and $\Phi^{-1}(0.75 - \frac{\overline{\lambda}_{t,\delta}}{2c_0^2})$ and $\Phi^{-1}(\frac{1+\overline{\theta}}{2} + \frac{\overline{\lambda}_{t,\delta}}{2c_0^2})$ are non-negative for $c_0 \geq \sqrt{2\overline{\lambda}_{t,\delta}}$. □

LEMMA 11.11. *Let $a_1, \ldots, a_t$ be i.i.d. realizations of a scalar random variable $\overline{a} \sim \mathcal{N}(0, \omega^2)$. Then, for any fixed $\gamma \in [0, 1)$, the following holds with probability at least $1 - 2\delta$:*

$$\min_{I \subseteq \{1,\ldots,t\},\, |I| \geq (1-\gamma)t} \frac{1}{t} \sum_{i \in I} a_i^2 \geq \omega^2 \Big\{ \int_{-\tilde{\tau}}^{\tilde{\tau}} \frac{x^2}{\sqrt{2\pi}} \exp(\frac{-x^2}{2}) dx - \tilde{\tau}^2 \sqrt{\frac{1}{2t} \log \frac{1}{\delta}} \Big\},$$

where $\tilde{\tau} \triangleq \Phi^{-1}\left(1 - \frac{\gamma}{2} - \sqrt{\frac{1}{8t} \log \frac{1}{\delta}}\right)$.

PROOF. Define function $\hat{f}(a) \triangleq (\mathbf{1}_{|a| \leq \omega\tau})a^2$. The following holds for any $\epsilon > 0$ by Hoeffding's inequality:

$$\Pr(\frac{1}{t} \sum_{i=1}^{t} \hat{f}(a_i) - \mathbb{E}_{\overline{a} \sim \mathcal{N}(0,\omega^2)} \hat{f}(\overline{a}) < -\epsilon\omega^2) \leq \exp(-\frac{2t\epsilon^2}{\tau^4}).$$

That is, with probability at least $1 - \delta$, the following holds

$$\begin{aligned} &\frac{1}{t} \sum_{i=1}^{t} \hat{f}(a_i) - \mathbb{E}_{\overline{a} \sim \mathcal{N}(0,\omega^2)} \hat{f}(\overline{a}) \geq -\tau^2\omega^2 \sqrt{\frac{1}{2t} \log \frac{1}{\delta}} \\ \Longrightarrow &\frac{1}{t} \sum_{i=1}^{t} \hat{f}(a_i) \geq \omega^2 \int_{-\tau}^{\tau} \frac{x^2}{\sqrt{2\pi}} \exp(\frac{-x^2}{2}) dx - \omega^2\sigma^2 \sqrt{\frac{1}{2t} \log \frac{1}{\delta}} \end{aligned} \tag{11.10}$$

Next define function $\hat{g}(a) \triangleq \mathbf{1}_{|a| \leq \omega\tau}$. By Hoeffding's inequality the following holds,

$$\Pr(\frac{1}{t} \sum_{i=1}^{t} \hat{g}(a_i) - \mathbb{E}_{\overline{a} \sim \mathcal{N}(0,\omega^2)} \hat{g}(\overline{a}) > \epsilon) \leq \exp(-2t\epsilon^2).$$

That is, with probability at least $1 - \delta$,

$$\frac{1}{t} \sum_{i=1}^{t} \hat{g}(a_i) - \mathbb{E}_{\overline{a} \sim \mathcal{N}(0,\omega^2)} \hat{g}(\overline{a}) \leq \sqrt{\frac{1}{2t} \log \frac{1}{\delta}}.$$

Notice that $\mathbb{E}_{\overline{a}\sim\mathcal{N}(0,\omega^2)}\hat{g}(\overline{a}) = \Phi(\tau)-\Phi(-\tau) = 1-\gamma-\sqrt{\frac{1}{2t}\log\frac{1}{\delta}}$. Hence with probability at least $1-\delta$,

$$\frac{1}{t}\sum_{i=1}^{t}\hat{g}(a_i) \leq 1-\gamma. \tag{11.11}$$

Notice that when Inequality (11.10) and (11.11) both hold, we have

$$\min_{I\subseteq\{1,\cdots,t\},|I|\geq(1-\gamma)t}\frac{1}{t}\sum_{i\in I}a_i^2 \geq \omega^2\Big\{\int_{-\tau}^{\tau}\frac{x^2}{\sqrt{2\pi}}\exp(\frac{-x^2}{2})dx - \tau^2\sqrt{\frac{1}{2t}\log\frac{1}{\delta}}\Big\},$$

which establishes the lemma. □

We now prove a finite-sample result, which we subsequently use to prove the main theorem of this chapter.

THEOREM 11.12. *Under (A), (B) and (C), if Algorithm 11.1 terminates at the s^{th} iteration, where $s \leq \bar{s}_0$, and $\int_{-\tilde{\zeta}}^{\tilde{\zeta}}\frac{x^2}{\sqrt{2\pi}}\exp(\frac{-x^2}{2})dx - \tilde{\zeta}^2\sqrt{\frac{1}{2t}\log\frac{d}{\delta}} > 0$, then with probability at least $1-2\delta$ the following holds*

$$\begin{aligned}
&\sum_{q=1}^{d}\mathbf{v}_q^{*\top}(AA^\top+I_m)\mathbf{v}_q^* \\
&\leq \frac{1+\sqrt{\overline{\eta}}}{\int_{-\tilde{\zeta}}^{\tilde{\zeta}}\frac{x^2}{\sqrt{2\pi}}\exp(\frac{-x^2}{2})dx - \tilde{\zeta}^2\sqrt{\frac{1}{2t}\log\frac{d}{\delta}}}\min_{c\geq\sqrt{2\overline{\lambda}_{t,\delta}}}\Bigg\{\left(\frac{\Phi^{-1}(\frac{1+\bar{\theta}}{2}+\frac{\overline{\lambda}_{t,\delta}}{2c^2})}{\Phi^{-1}(0.75-\frac{\overline{\lambda}_{t,\delta}}{2c^2})}\right)^2(\sum_{q=1}^{d}\mathbf{v}_q^\top AA^\top\mathbf{v}_q) \\
&+\frac{\sqrt{d}\Phi^{-1}(\frac{1+\bar{\theta}}{2}+\frac{\overline{\lambda}_{t,\delta}}{2c^2})(4c+\hat{\phi}_{t,\delta}\overline{\sigma}_1+\hat{\psi}_{t,\delta}\overline{\sigma}_1)}{[\Phi^{-1}(0.75-\frac{\overline{\lambda}_{t,\delta}}{2c^2})]^2}\sqrt{\sum_{q=1}^{d}\mathbf{v}_q^\top AA^\top\mathbf{v}_q}+\frac{(4c+\hat{\phi}_{t,\delta}\overline{\sigma}_1+\hat{\psi}_{t,\delta}\overline{\sigma}_1)^2}{4[\Phi^{-1}(0.75-\frac{\overline{\lambda}_{t,\delta}}{2c^2})]^2}+\tilde{v}_{t,\delta}\Bigg\}.
\end{aligned} \tag{11.12}$$

PROOF. By definition, if HR-PCA terminates at the s^{th} iteration, then for all $q\in\{1,\ldots,d\}$

$$\frac{(1+\sqrt{\overline{\eta}})(1-\overline{\eta})n}{n-s}\overline{H}_{\mathbf{v}_q} \geq \hat{\sigma}_q^2.$$

Substituting the definitions of $\tilde{v}_{t,\delta}$ and $\overline{H}_{\mathbf{w}}$ into Lemma 11.10, with some algebra we have

$$\begin{aligned}
&\frac{n-s}{(1+\sqrt{\overline{\eta}})(1-\overline{\eta})n}\hat{\sigma}_q^2 \\
&\leq \min_{c\geq\sqrt{2\overline{\lambda}_{t,\delta}}}\left\{\left(\frac{\Phi^{-1}(\frac{1+\bar{\theta}}{2}+\frac{\overline{\lambda}_{t,\delta}}{2c^2})}{\Phi^{-1}(0.75-\frac{\overline{\lambda}_{t,\delta}}{2c^2})}\right)^2 \mathbf{v}_q^\top AA^\top\mathbf{v}_q\right. \\
&\left.+\frac{\Phi^{-1}(\frac{1+\bar{\theta}}{2}+\frac{\overline{\lambda}_{t,\delta}}{2c^2})(4c+\hat{\phi}_{t,\delta}\overline{\sigma}_1+\hat{\psi}_{t,\delta}\overline{\sigma}_1)}{[\Phi^{-1}(0.75-\frac{\overline{\lambda}_{t,\delta}}{2c^2})]^2}\sqrt{\mathbf{v}_q^\top AA^\top\mathbf{v}_q}+\frac{(4c+\hat{\phi}_{t,\delta}\overline{\sigma}_1+\hat{\psi}_{t,\delta}\overline{\sigma}_1)^2}{4[\Phi^{-1}(0.75-\frac{\overline{\lambda}_{t,\delta}}{2c^2})]^2}+\tilde{v}_{t,\delta}\right\}.
\end{aligned}$$

Summing up for $q=1,\ldots,d$, noticing that the minimal value of the sum is no less than the summation of the minimal value of each term and using the inequality $\sum_{i=1}^d a_i \leq \sqrt{d\sum_{i=1}^d a_i^2}$ for any $a_i\in\mathbb{R}$, we have

$$\begin{aligned}
&\frac{n-s}{(1+\sqrt{\overline{\eta}})(1-\overline{\eta})n}\sum_{q=1}^d\hat{\sigma}_q^2 \\
&\leq \min_{c\geq\sqrt{2\overline{\lambda}_{t,\delta}}}\left\{\left(\frac{\Phi^{-1}(\frac{1+\bar{\theta}}{2}+\frac{\overline{\lambda}_{t,\delta}}{2c^2})}{\Phi^{-1}(0.75-\frac{\overline{\lambda}_{t,\delta}}{2c^2})}\right)^2 (\sum_{q=1}^d \mathbf{v}_q^\top AA^\top\mathbf{v}_q)\right. \\
&+\frac{\sqrt{d}\Phi^{-1}(\frac{1+\bar{\theta}}{2}+\frac{\overline{\lambda}_{t,\delta}}{2c^2})(4c+\hat{\phi}_{t,\delta}\overline{\sigma}_1+\hat{\psi}_{t,\delta}\overline{\sigma}_1)}{[\Phi^{-1}(0.75-\frac{\overline{\lambda}_{t,\delta}}{2c^2})]^2}\sqrt{\sum_{q=1}^d\mathbf{v}_q^\top AA^\top\mathbf{v}_q} \\
&\left.+\frac{(4c+\hat{\phi}_{t,\delta}\overline{\sigma}_1+\hat{\psi}_{t,\delta}\overline{\sigma}_1)^2}{4[\Phi^{-1}(0.75-\frac{\overline{\lambda}_{t,\delta}}{2c^2})]^2}+\tilde{v}_{t,\delta}\right\}.
\end{aligned} \tag{11.13}$$

Since $\mathbf{v}_q$ are the PCs of the remaining points at the s^{th} iteration, then for all orthonormal $\{\hat{\mathbf{v}}_1,\ldots,\hat{\mathbf{v}}_d\}$,

$$\sum_{q=1}^d\hat{\mathbf{v}}_q^\top\hat{\Sigma}\hat{\mathbf{v}}_q\leq\sum_{q=1}^d\mathbf{v}_q^\top\hat{\Sigma}\mathbf{v}_q=\sum_{q=1}^d\hat{\sigma}_q^2,$$

where $\hat{\Sigma}$ is the covariance matrix of the remain points.

Recall that $\mathbf{v}_1^*,\ldots,\mathbf{v}_d^*$ are orthonormal and are independent to $\mathcal{Y}$. Hence for a fixed $q\in\{1,\ldots,d\}$, the projection of the authentic samples onto $\mathbf{v}_q^*$ follows a

Gaussian distribution with variance $\mathbf{v}_q^{*\top}(AA^\top + I_m)\mathbf{v}_q^*$. Therefore, by Lemma 11.11 and since $s \leq \bar{s}_0$, we have with probability $1 - 2\delta/d$

$$\begin{aligned}\mathbf{v}_q^{*\top}\hat{\Sigma}\mathbf{v}_q^* \geq & \frac{1}{n-s}\min_{I' \subseteq \{1,\ldots,n\}, |I'| \geq n-s} \sum_{i \in I'} (\mathbf{v}_q^{*\top}\mathbf{y}_i)^2 \\ \geq & \frac{t}{n-s}\mathbf{v}_q^{*\top}(AA^\top + I_m)\mathbf{v}_q^* \Big\{ \int_{-\tilde{\zeta}}^{\tilde{\zeta}} \frac{x^2}{\sqrt{2\pi}} \exp(\frac{-x^2}{2}) dx - \tilde{\zeta}^2 \sqrt{\frac{1}{2t}\log\frac{d}{\delta}} \Big\}.\end{aligned}$$

Summing up over q and substituting it into Inequality (11.13), we establish the theorem. □

Notice that Conditions (A), (B) and (C) hold simultaneously with probability at least $1 - 3\delta$, and Algorithm 11.1 terminates at $s < \bar{s}_0$ with probability at least $1 - \exp\left(\frac{-n\sqrt{\overline{\eta}\eta}}{8(1+\sqrt{\overline{\eta}})}\right)$. Hence, Theorem 11.12 implies that this finite sample bound holds with probability at least $1 - 5\delta - \exp\left(\frac{-n\sqrt{\overline{\eta}\eta}}{8(1+\sqrt{\overline{\eta}})}\right)$.

Finally, we prove Theorem 11.1, which provides bounds on the asymptotic performance of the algorithm. To simplify the expressions, let

$$\begin{aligned}\bar{\theta}^* &\triangleq \frac{1+\overline{\eta}^*}{2-2\overline{\eta}^*}; \\ \rho(j) &\triangleq \sum_{q=1}^{d} \mathbf{v}_q(j)^\top (A(j)A(j)^\top)\mathbf{v}_q(j); \\ \rho^*(j) &\triangleq \sum_{q=1}^{d} \mathbf{v}_q^*(j)^\top (A(j)A(j)^\top)\mathbf{v}_q^*(j).\end{aligned}$$

PROOF OF THEOREM 11.1. Taking $c = \sqrt{\overline{\sigma}_1(j)}$ and dividing both sides of Theorem 11.12 by $\rho^*(j)$, we have:

$$\begin{aligned}
&\frac{\int_{-\tilde{\zeta}(j)}^{\tilde{\zeta}(j)} \frac{x^2}{\sqrt{2\pi}} \exp(\frac{-x^2}{2}) dx - \tilde{\zeta}(j)^2 \sqrt{\frac{1}{2t(j)} \log \frac{d}{\delta}}}{1 + \sqrt{\overline{\eta}}(j)} \\
&\leq \left(\frac{\Phi^{-1}(\frac{1+\bar{\theta}(j)}{2} + \frac{\overline{\lambda}_{t(j),\delta}}{2\overline{\sigma}_1(j)})}{\Phi^{-1}(0.75 - \frac{\overline{\lambda}_{t(j),\delta}}{2\overline{\sigma}_1(j)})} \sqrt{\frac{\rho(j)}{\rho^*(j)}} + \frac{\sqrt{d}(4\sqrt{\overline{\sigma}_1(j)} + \hat{\phi}_{t(j),\delta}\overline{\sigma}_1(j) + \hat{\psi}_{t(j),\delta}\overline{\sigma}_1(j))}{2\sqrt{\rho^*(j)}} \right)^2 \\
&\quad + \frac{(4\sqrt{\overline{\sigma}_1(j)} + \hat{\phi}_{t(j),\delta}\overline{\sigma}_1(j) + \hat{\psi}_{t(j),\delta}\overline{\sigma}_1(j))^2}{4[\Phi^{-1}(0.75 - \frac{\overline{\lambda}_{t(j),\delta}}{2\overline{\sigma}_1(j)})]^2 \rho^*(j)} + \frac{\tilde{v}_{t(j),\delta}}{\rho^*(j)}.
\end{aligned} \tag{11.14}$$

Notice that by definition $\sqrt{\rho^*(j)} \geq \overline{\sigma}_1(j) \uparrow +\infty$. Furthermore, we have

$$\hat{\phi}_{t(j),\delta} \downarrow 0; \quad \hat{\psi}_{t(j),\delta} \downarrow 0; \quad \tilde{v}_{t(j),\delta} \downarrow 0; \quad \overline{\lambda}_{t(j),\delta} \downarrow \frac{4}{1-\overline{\eta}}.$$

Thus the right-hand-side of Inequality (11.14) converges to

$$\left(\frac{\Phi^{-1}(\frac{1+\bar{\theta}(j)}{2} + \frac{\overline{\lambda}_{t(j),\delta}}{2\overline{\sigma}_1(j)})}{\Phi^{-1}(0.75 - \frac{\overline{\lambda}_{t(j),\delta}}{2\overline{\sigma}_1(j)})} \sqrt{\frac{\rho(j)}{\rho^*(j)}} \right)^2, \tag{11.15}$$

since all other terms go to zero. Notice that (11.15) further converges to

$$\left(\frac{\Phi^{-1}(\frac{1+\bar{\theta}^*}{2})}{\Phi^{-1}(0.75)} \sqrt{\frac{\rho(j)}{\rho^*(j)}} \right)^2,$$

as $\overline{\sigma}_1(j)$ increases and $\overline{\eta}(j) \to \overline{\eta}^*$. On the other hand, noticing that $\tilde{\zeta}(j) \to \tilde{\zeta}^*$ and $\sqrt{\overline{\eta}}(j) \to \sqrt{\overline{\eta}}^*$, we have that the left-hand-side of Inequality (11.14) converges to

$$\frac{\int_{-\tilde{\zeta}^*}^{\tilde{\zeta}^*} \frac{x^2}{\sqrt{2\pi}} \exp(\frac{-x^2}{2}) dx}{1 + \sqrt{\overline{\eta}}^*}.$$

The corollary follows by definition of $\sqrt{\overline{\eta}}^*$ and $\bar{\theta}^*$. □

11.5. Kernelization

We consider kernelizing HR-PCA in this section: given a feature mapping $\Upsilon(\cdot) : \mathbb{R}^m \to \mathcal{H}$ equipped with a kernel function $k(\cdot,\cdot)$, i.e., $\langle \Upsilon(\mathbf{a}), \Upsilon(\mathbf{b}) \rangle = k(\mathbf{a},\mathbf{b})$ holds for all $\mathbf{a},\mathbf{b} \in \mathbb{R}^m$, we perform the dimensionality reduction in the feature space $\mathcal{H}$ without knowing the explicit form of $\Upsilon(\cdot)$.

We assume that $\{\Upsilon(\mathbf{y}_1),\ldots,\Upsilon(\mathbf{y}_n)\}$ is centered at origin without loss of generality, since we can center any $\Upsilon(\cdot)$ with the following feature mapping

$$\hat{\Upsilon}(\mathbf{x}) \triangleq \Upsilon(\mathbf{x}) - \frac{1}{n}\sum_{i=1}^{n}\Upsilon(\mathbf{y}_i),$$

whose kernel function is

$$\begin{aligned}\hat{k}(\mathbf{a},\mathbf{b}) = k(\mathbf{a},\mathbf{b}) &- \frac{1}{n}\sum_{j=1}^{n}k(\mathbf{a},\mathbf{y}_j) \\ &- \frac{1}{n}\sum_{i=1}^{n}k(\mathbf{y}_i,\mathbf{b}) + \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}k(\mathbf{y}_i,\mathbf{y}_j).\end{aligned}$$

Notice that HR-PCA involves finding a set of PCs $\{\mathbf{v}_1,\ldots,\mathbf{v}_d\}$, and evaluating $l(\cdot)$ that is a function of $\{\mathbf{v}_q^\top\mathbf{y}_1,\ldots,\mathbf{v}_q^\top\mathbf{y}_n\}$ for $q = 1,\ldots,d$. The former can be kernelized by applying Kernel PCA introduced by [**134**], where each of the output PCs admits a representation

$$\mathbf{v}_q = \sum_{i=1}^{n-s}\alpha_i(q)\Upsilon(\hat{\mathbf{y}}_i), \quad q = 1,\ldots,d.$$

Thus, $l(\cdot)$ is easily evaluated by

$$\mathbf{v}_q^\top\Upsilon(\mathbf{y}_j) = \sum_{i=1}^{n-s}\alpha_i(q)k(\hat{\mathbf{y}}_i,\mathbf{y}_j).$$

Therefore, HR-PCA is kernelizable since both steps are easily kernelized, and we have the following Kernel HR-PCA.

ALGORITHM 11.2. **Kernel HR-PCA**

Input: Contaminated sample-set $\mathcal{Y} = \{\mathbf{y}_1,\ldots,\mathbf{y}_n\} \subset \mathbb{R}^m$, $\overline{\eta}$, d.

Output: $\boldsymbol{\alpha}(1), \ldots, \boldsymbol{\alpha}(d)$.

Algorithm:

(1) Let $\sqrt{\overline{\eta}} := \sqrt{\overline{\eta}}$; $\hat{\mathbf{y}}_i := \mathbf{y}_i$ for $i = 1, \ldots n$; $s := 0$.

(2) Compute the Gram matrix of $\{\hat{\mathbf{y}}_i\}$:

$$K_{ij} := k(\hat{\mathbf{y}}_i, \hat{\mathbf{y}}_j); \quad i, j = 1, \ldots, n - s.$$

(3) Let $\hat{\sigma}_1^2, \ldots, \hat{\sigma}_d^2$ and $\hat{\boldsymbol{\alpha}}(1), \ldots, \hat{\boldsymbol{\alpha}}(d)$ be the d largest eigenvalues and the corresponding eigenvectors of K.

(4) Normalize: $\boldsymbol{\alpha}(q) := \hat{\boldsymbol{\alpha}}(q)/\hat{\sigma}_q$.

(5) If there is a $q \in \{1, \ldots, d\}$ such that *Kernel Sensitivity Test* $\mathbb{H}_k(\boldsymbol{\alpha}(q), \hat{\sigma}_q, s)$ fails, do the following:

- randomly remove a point from $\{\hat{\mathbf{y}}_i\}_{i=1}^{n-s}$ according to

$$\Pr(\hat{\mathbf{y}}_i \text{ is removed}) \propto (\sum_{j=1}^{n-s} \alpha(q)_j k(\hat{\mathbf{y}}_j, \hat{\mathbf{y}}_i))^2;$$

- denote the remaining points by $\{\hat{\mathbf{y}}_i\}_{i=1}^{n-s-1}$;
- $s := s + 1$, go to Step 2.

(6) Output $\boldsymbol{\alpha}(1), \ldots, \boldsymbol{\alpha}(d)$. End.

We next define the Kernel Sensitivity Test. For $\boldsymbol{\alpha} \in \mathbb{R}^{n-s}$,

$$l_{\boldsymbol{\alpha}} \triangleq l\big(0.5 + \frac{\overline{\eta}}{2}, \sum_{i=1}^{n-s} \alpha_i k(\hat{\mathbf{y}}_i, \mathbf{y}_1), \ldots, \sum_{i=1}^{n-s} \alpha_i k(\hat{\mathbf{y}}_i, \mathbf{y}_n)\big);$$

$$\overline{O}_{\boldsymbol{\alpha}} \triangleq \min_{c \geq \sqrt{2\overline{\lambda}_{t,\delta}}} \Big\{ \frac{l_{\boldsymbol{\alpha}} + \hat{\phi}_{t,\delta} \overline{\sigma}_1 + 2c}{2\Phi^{-1}(0.75 - \frac{\overline{\lambda}_{t,\delta}}{2c^2})} \Big\};$$

$$\overline{H}_{\boldsymbol{\alpha}} \triangleq \overline{O}_{\boldsymbol{\alpha}}^2 + \tilde{v}_{t,\delta}.$$

Definition 11.2. Kernel Sensitivity Test $\mathbb{H}_k$ *is defined as*

$$\mathbb{H}_k(\boldsymbol{\alpha}, \sigma, s) \triangleq \begin{cases} \textit{false}, & \textit{if } \frac{(1+\sqrt{\overline{\eta}})(1-\overline{\eta})n}{n-s} \overline{H}_{\boldsymbol{\alpha}} < \sigma^2; \\ \textit{true}, & \textit{if } \frac{(1+\sqrt{\overline{\eta}})(1-\overline{\eta})n}{n-s} \overline{H}_{\boldsymbol{\alpha}} \geq \sigma^2. \end{cases}$$

11.6. Numerical illustrations

We report in this section some numerical experimental results. We let $n = m = 100$, i.e., 100 points, each with 100 dimensions. Each element of A is generated according to a uniform distribution; A is then scaled so that its leading eigenvalue equals the given $\overline{\sigma}_1$. All corrupted points are generated on a randomly selected direction. We compare the performance of PCA and HR-PCA for different ratios of corrupted points, magnitudes of corrupted points and $\overline{\sigma}_1$. For each set of parameters, we report the average result of 100 tests.

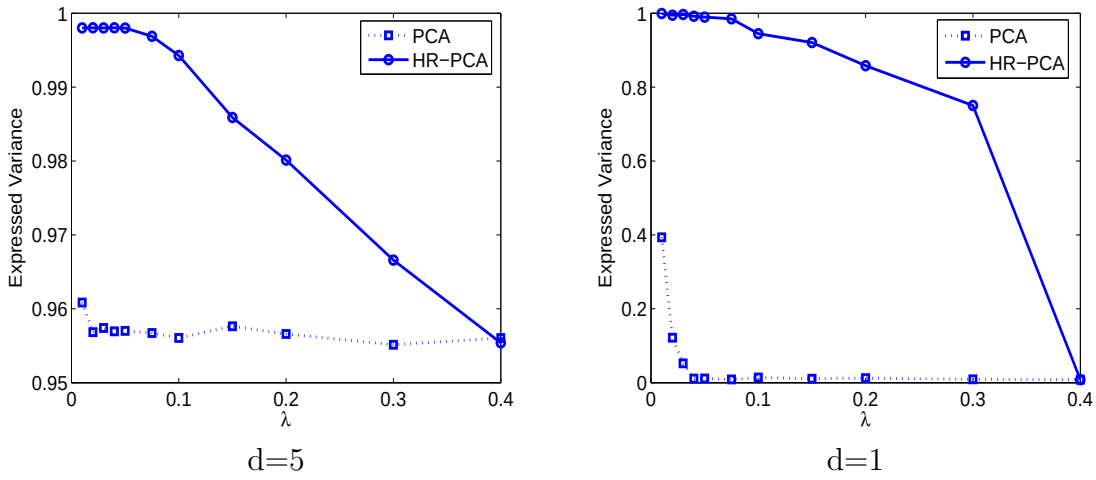


FIGURE 11.4. Performance for different ratios of corrupted points

The performance of PCA and HR-PCA of different $\overline{\eta}$ is reported in Figure 11.4, where both $\overline{\sigma}_1$ and the magnitude of the corrupted points are fixed as 50. As one would expect, HR-PCA outperforms PCA for all $\overline{\eta}$. The performance of HR-PCA breaks only for $\overline{\eta}$ as large as 0.4, i.e., 40% of points are corrupted. We notice that the empirical performance is much better than predicted by the theoretical lower-bound, which is to be expected since the lower bound is derived from a very pessimistic analysis. We also observe that PCA performs much better in the case $d = 5$ than for $d = 1$. This is mainly due to the fact that corrupted points are generated in only one direction. Thus even though PCA wrongly picks the corrupted point direction as a PC, for $d = 5$, the other 4 directions PCA picks are correct, and hence the

total Expressed Variance seems to be acceptable. Figure 11.5 shows a significant performance degradation of PCA in the $d = 5$ case when the corrupted points are generated in 5 random directions.

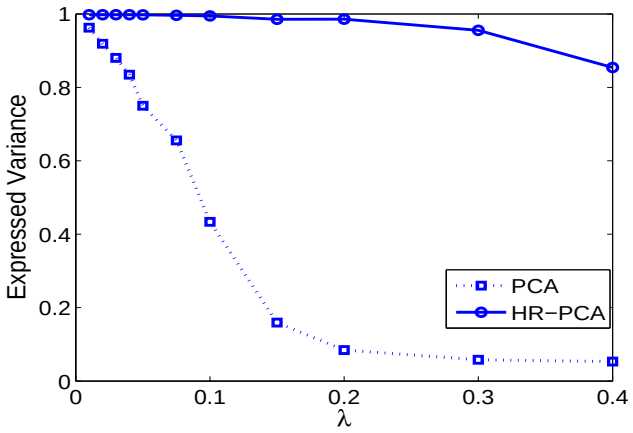


FIGURE 11.5. Performance for different ratios of corrupted points: corrupted points generated in multiple directions

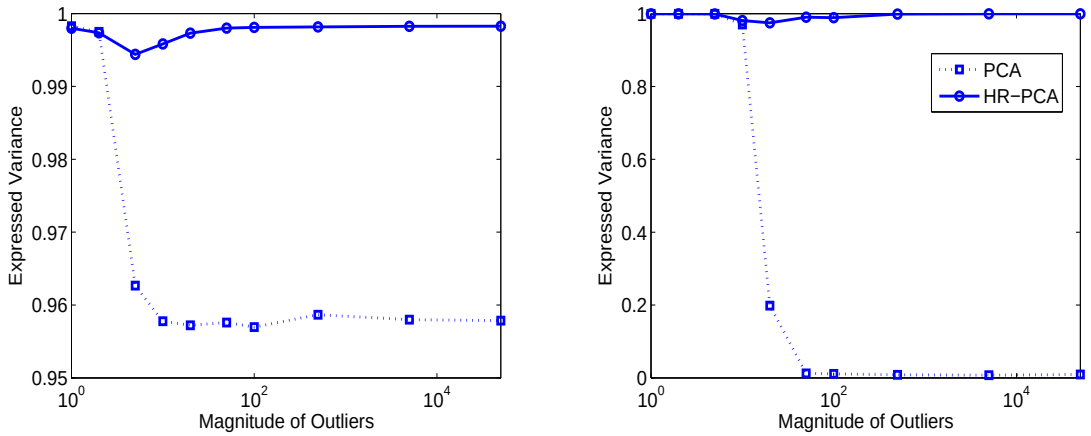


FIGURE 11.6. Performance for different magnitudes of corrupted points

Figure 11.6 shows the performance of HR-PCA and PCA for different magnitudes of corrupted points, with $\overline{\sigma}_1 = 50$ and $\overline{\eta} = 0.05$. One interesting observation is the performance of HR-PCA seems to be quite consistent for different magnitudes of the corrupted points. Indeed, when the corrupted points are large, the performance of

HR-PCA is as good as the no-corruption case, mainly because the corrupted points become easier to remove.

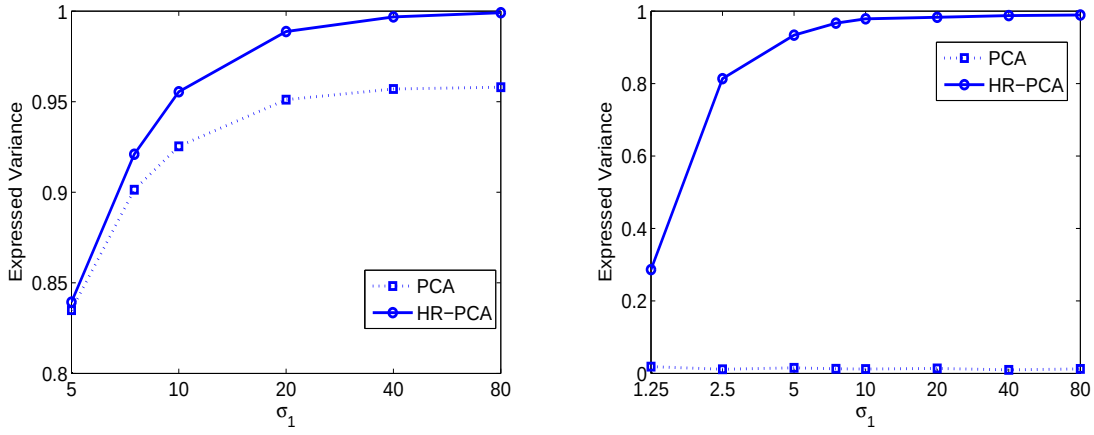


FIGURE 11.7. Performance for different $\overline{\sigma}_1$

Figure 11.7 shows that the performance of HR-PCA becomes satisfactory for reasonably large $\overline{\sigma}_1$ ($\overline{\sigma}_1 \geq 5$ for 1-d case and $\overline{\sigma}_1 \geq 20$ for 5-d case).

11.7. Concluding remarks

In this chapter, we investigated the dimensionality-reduction problem in the case where the number and the dimensionality of samples are of the same magnitude, and a constant fraction of the points are arbitrarily corrupted (perhaps maliciously so). We proposed a High-dimensional Robust Principal Component Analysis algorithm that is tractable, robust to corrupted points, easily kernelizable and asymptotically optimal. The algorithm takes an "actor-critic" form: iteratively finding a set of PCs using standard PCA and subsequently validating the robustness of those PCs using the confidence interval, using a point removal procedure in case of validation failure. We provided both theoretical guarantees and favorable simulation results about the performance of the proposed algorithm.

To the best of our knowledge, previous efforts to extend existing robust PCA algorithms into the high-dimensional case remain unsuccessful. Such algorithms are

designed for low dimensional data sets where the observations significantly outnumber the variables of each dimension. When applied to high-dimensional data sets, they either lose statistical consistency due to lack of sufficient observations, or become highly intractable. This motivates our work of proposing a new robust PCA algorithm that takes into account the inherent difficulty in analyzing high-dimensional data.

11.8. The Tracy-Widom distribution

The *Tracy-Widom* distribution of order 1, denoted Ψ, is defined as

$$\Psi(w) = \exp\left\{-\frac{1}{2}\int_w^\infty q(x) + (x-w)q^2(x)dx\right\}, \;\; w \in \mathbb{R},$$

where q solves $\ddot{q}(x) = xq(x) + 2q^3(x)$, $q(x) \sim Ai(x)$ as $x \to +\infty$, and $Ai(x)$ denotes the Airy function. This distribution was first identified and characterized in [**148**]. Numerical work (e.g., [**149**]) shows that the Ψ distribution has mean ≈ -1.21, and standard deviation ≈ 1.27. The density of Ψ is asymmetric ([**93**]): its left tail decays exponentially according to $\exp(-|w|^3/24)$, while its right tail decays exponentially as $\exp(-\frac{2}{3}w^{3/2})$. We quote the following numerical table from ([**93**]):

w	-2.78	-1.91	-1.27	-0.59	0.45	0.98	2.02
$\Psi(w)$	0.10	0.30	0.50	0.70	0.90	0.95	0.99

11.9. Proofs of technical results

11.9.1. Proof of Theorem 11.2. We prove in this subsection Theorem 11.2.

Theorem 11.2 (a): For sufficiently large t and m, Condition (A) holds with probability at least $1 - \delta$.

PROOF. To prove the theorem, we need the following lemma, stated and proved in [**93**].

LEMMA 11.13. *Let* $\mathbf{n}_1, \cdots, \mathbf{n}_t$ *be i.i.d. realizations of* $\overline{\mathbf{n}} \sim \mathcal{N}(0, I_m)$. *Define*

$$\ell_1 \triangleq \lambda_{\max}(\sum_{i=1}^{t} \mathbf{n}_i \mathbf{n}_i^\top), \quad \mu_{tm} \triangleq (\sqrt{t-1}+\sqrt{m})^2; \quad \sigma_{tm} \triangleq (\sqrt{t-1}+\sqrt{m})(\frac{1}{\sqrt{t-1}}+\frac{1}{\sqrt{m}})^{1/3}.$$

If $t/m \to c \geq 1$, *we have*

$$\frac{\ell_1 - \mu_{tm}}{\sigma_{tm}} \to W_1 \sim \Psi,$$

and if $m/t \to c > 1$, *we have*

$$\frac{\ell_1 - \mu_{mt}}{\sigma_{mt}} \to W_1 \sim \Psi,$$

(where Ψ *denotes the Tracy-Widom distribution of order one) i.e., the theorem still holds by simply reversing* t *and* m.

Now we proceed to prove the theorem. Lemma 11.13 implies that for large m and t, the following holds with probability at least $1-\delta$:

$$\ell_1 \leq \max\Big(\mu_{mt} + \Psi^{-1}(1-\delta)\sigma_{mt},\ \ \mu_{tm} + \Psi^{-1}(1-\delta)\sigma_{tm}\Big).$$

Using this inequality, we obtain the bound that is the statement of the theorem:

$$\begin{aligned}
&\lambda_{\max}\Big(\frac{1}{t}\sum_{i=1}^{t} \mathbf{n}_i \mathbf{n}_i^\top\Big) = \frac{1}{t}\ell_1 \leq \max\Big(\frac{\mu_{mt} + \Psi^{-1}(1-\delta)\sigma_{mt}}{t}, \frac{\mu_{tm} + \Psi^{-1}(1-\delta)\sigma_{tm}}{t}\Big) \\
\leq &\frac{(\sqrt{n}+\sqrt{m})^2 + \Psi^{-1}(1-\delta)(\sqrt{n}+\sqrt{m})(\frac{1}{\sqrt{m-1}} + \frac{1}{\sqrt{t-1}})^{1/3}}{(1-\overline{\eta})n} \\
\leq &\frac{4}{1-\overline{\eta}} + \frac{2\Psi^{-1}(1-\delta)}{(1-\overline{\eta})\sqrt{n}} \leq \frac{4}{1-\overline{\eta}} + \frac{2[\Psi^{-1}(1-\delta)]^+}{(1-\overline{\eta})\sqrt{n}} = \bar{\lambda}_{t,\delta}.
\end{aligned}$$

The third inequality holds when $m, t \geq 5$, and by assumption $n \geq m$. This concludes the proof of the theorem. □

Theorem 11.2 (b): For sufficiently large t, Condition (B) holds with probability at least $1-\delta$.

PROOF. Let x_{ij} denote the j^{th} element of $\mathbf{x}_i$. Recall that these are i.i.d. scalar random variables, following a standard normal distribution $\mathcal{N}(0,1)$. We prove the following lemma before proving the main theorem.

LEMMA 11.14. *For any $j, j' = 1, \cdots, d$ such that $j \neq j'$, the following holds:*

$$\frac{1}{\sqrt{t}} \sum_{i=1}^{t} (x_{ij}^2 - 1) \rightarrow W_2 \sim \mathcal{N}(0,3);$$

$$\frac{1}{\sqrt{t}} \sum_{i=1}^{t} x_{ij} x_{ij'} \rightarrow W_3 \sim \mathcal{N}(0,1).$$

PROOF. Notice that $\{x_{ij}^2\}$ are i.i.d. random variables because the $\{x_{ij}\}$ are i.i.d., and they have mean 1 and variance $\mathbb{E}(x_{11}^4) = 3$. Now, the first convergence result follows directly from the Central Limit Theorem. For any $j \neq j'$, the products $\{x_{ij}x_{ij'}\}$ are i.i.d. random variables. Furthermore, $\mathbb{E}(x_{ij}x_{ij'}) = \mathbb{E}x_{1j}\mathbb{E}x_{1j'} = 0$, and $\mathbb{E}((x_{ij}x_{ij'})^2) = \mathbb{E}(x_{ij}^2)\mathbb{E}(x_{ij'}^2) = 1$, since the $\{x_{ij}\}$ are i.i.d. From here, the second equation follows from the Central Limit Theorem. □

Now we prove Theorem 11.2 (b). This is equivalent to showing that for any fixed $\epsilon > 0$, the following holds:

$$\Pr\{ \sup_{\|\mathbf{w}\|=1} \left| \mathbf{w}^\top (\frac{1}{t} \sum \mathbf{x}_i \mathbf{x}_i^\top - I_d) \mathbf{w} \right| > \epsilon \} \leq 2d^2 \exp(-\frac{\epsilon^2 t}{6d^2}). \tag{11.16}$$

Letting w_j stand for the j^{th} component of $\mathbf{w}$, we have

$$\begin{aligned} &\Pr \left\{ \sup_{\|\mathbf{w}\|=1} \left| \mathbf{w}^\top (\frac{1}{t} \sum \mathbf{x}_i \mathbf{x}_i^\top - I_d) \mathbf{w} \right| \geq \epsilon \right\} \\ =&\Pr \left\{ \sup_{\|\mathbf{w}\|=1} \left| \frac{1}{t} \sum_{i=1}^{t} \sum_{j=1}^{d} (w_j)^2 (x_{ij}^2 - 1) + \frac{1}{t} \sum_{i=1}^{t} \sum_{j,j' | j \neq j'} w_j w_{j'} x_{ij} x_{ij'} \right| \geq \epsilon \right\}. \end{aligned}$$

Notice that $\|\mathbf{w}\| = 1 \Rightarrow \sum_j (w_j)^2 + \sum_{j,j'|j\neq j'} w_j w_{j'} \leq d$, hence

$$\begin{aligned}
&\sup_{\|\mathbf{w}\|=1} \Big|\frac{1}{t}\sum_{i=1}^{t}\sum_{j=1}^{d}(w_j)^2(x_{ij}^2-1) + \frac{1}{t}\sum_{i=1}^{t}\sum_{j,j'|j\neq j'} w_j w_{j'} x_{ij}x_{ij'}\Big| \\
\leq & d\max\left\{\max_j \Big|\frac{1}{t}\sum_{i=1}^{t}(x_{ij}^2-1)\Big|, \max_{j\neq j'}\Big|\frac{1}{t}\sum_{i=1}^{t}x_{ij}x_{ij'}\Big|\right\}.
\end{aligned}$$

Thus we have for sufficiently large t,

$$\begin{aligned}
&\Pr\left\{\sup_{\|\mathbf{w}\|=1} \Big|\frac{1}{t}\sum_{i=1}^{t}\sum_{j=1}^{d}(w_j)^2(x_{ij}^2-1) + \frac{1}{t}\sum_{i=1}^{t}\sum_{j,j'|j\neq j'} w_j w_{j'} x_{ij}x_{ij'}\Big| \geq \epsilon\right\} \\
\leq & \Pr\left\{\max_j \Big|\frac{1}{t}\sum_{i=1}^{t}(x_{ij}^2-1)\Big| \geq \frac{\epsilon}{d}\right\} + \Pr\left\{\max_{j\neq j'}\Big|\frac{1}{t}\sum_{i=1}^{t}x_{ij}x_{ij'}\Big| \geq \frac{\epsilon}{d}\right\} \\
\leq & d\times \Pr\left\{\Big|\frac{1}{t}\sum_{i=1}^{t}(x_{i1}^2-1)\Big| \geq \frac{\epsilon}{d}\right\} + (d^2-d)\times \Pr\left\{\Big|\frac{1}{t}\sum_{i=1}^{t}x_{i1}x_{i2}\Big| \geq \frac{\epsilon}{d}\right\} \\
\leq & 2d(1-\Phi(\frac{\epsilon\sqrt{t}}{\sqrt{3}d})) + 2d(d-1)(1-\Phi(\frac{\epsilon\sqrt{t}}{d})) \\
\leq & 2d\exp(-\frac{\epsilon^2 t}{6d^2}) + (2d^2-2d)\exp(-\frac{\epsilon^2 t}{2d^2}) \leq 2d^2\exp(-\frac{\epsilon^2 t}{6d^2}).
\end{aligned}$$

The third inequality follows from Lemma 11.14, and the fourth inequality follows from $1-\Phi(u) \leq \exp(-u^2/2)$. □

Theorem 11.2 (c): With probability at least $1-\delta$, Condition (C) holds.

PROOF. We start from the following lemma.

LEMMA 11.15. *With probability at least $1-\delta$, the following holds*

$$\sup_{\mathbf{w}\in\mathbb{R}^d,\, \|\mathbf{w}\|_2=1;\, b\in\mathbb{R}} \Big|\frac{1}{t}\sum_{i=1}^{t}\mathbf{1}_{\mathbf{w}^\top\mathbf{x}_i\leq b} - \Phi(b)\Big| \leq \hat{h}_{t,\delta}.$$

PROOF. Consider the set of indicator functions of half-spaces of $\mathbb{R}^d$

$$\mathcal{F} \triangleq \{f(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R} | f(\mathbf{x}) = \mathbf{1}_{\mathbf{w}^\top\mathbf{x}\leq b};\ \mathbf{w}\in\mathbb{R}^d,\ \|\mathbf{w}\|_2 = 1;\ b\in\mathbb{R}\}.$$

It is well known that the VC dimension of $\mathcal{F}$ is $d+1$ (cf [**133**]). Therefore, a standard uniform convergence argument shows that for t i.i.d. random variables $\mathbf{x}_i$ that follow some law $\mathbb{P}$, then with probability at least $1-\delta$ we have:

$$\begin{aligned}&\sup_{f\in\mathcal{F}}|\frac{1}{t}\sum_{i=1}^{t}f(\mathbf{x}_i)-\mathbb{E}_{\mathbb{P}}f(\mathbf{x})|\\ \leq&\sqrt{\frac{8}{t}\left(\mathrm{VC}(\mathcal{F})\ln\frac{t}{\mathrm{VC}(\mathcal{F})}+\ln\frac{8}{\delta}\right)}=\sqrt{\frac{8d+8}{t}\ln\frac{t}{d+1}+\frac{8}{t}\ln\frac{8}{\delta}}=\hat{h}_{t,\delta}.\end{aligned}\tag{11.17}$$

Note that when $\mathbb{P}$ is a d-dimensional Normal distribution $\mathcal{N}(0,I_d)$, for any $f\in\mathcal{F}$ defined by $\|\mathbf{w}\|_2=1$ and b, we have

$$\mathbb{E}_{\mathbb{P}}f(\mathbf{x})=\mathbb{E}_{\mathcal{N}(0,I_d)}(\mathbf{1}_{\mathbf{w}^\top\mathbf{x}\leq b})=\Phi(b).$$

Substituting this into Inequality (11.17) proves the lemma. □

Now we proceed to prove Theorem 11.2 (c). By Lemma 11.15, we have that with probability at least $1-\delta$, the following holds:

$$\Phi(b)-\hat{h}_{t,\delta}\leq\frac{1}{t}\sum_{i=1}^{t}\mathbf{1}_{\mathbf{w}^\top\mathbf{x}_i\leq b}\leq\Phi(b)+\hat{h}_{t,\delta};\quad\forall\|\mathbf{w}\|_2=1,\,b\in\mathbb{R}.\tag{11.18}$$

Inequality (11.18) implies that $\forall\|\mathbf{w}\|_2=1,\ \theta\in[0,1]$

$$\begin{aligned}&\frac{1}{t}\sum_{i=1}^{t}\mathbf{1}_{-\Phi^{-1}(\frac{1+\theta}{2}+\hat{h}_{t,\delta})\leq\mathbf{w}^\top\mathbf{x}_i\leq\Phi^{-1}(\frac{1+\theta}{2}+\hat{h}_{t,\delta})}\\ \geq&\Phi\left(\Phi^{-1}(\frac{1+\theta}{2}+\hat{h}_{t,\delta})\right)-\Phi\left(-\Phi^{-1}(\frac{1+\theta}{2}+\hat{h}_{t,\delta})\right)-2\hat{h}_{t,\delta}\\ =&\Phi\left(\Phi^{-1}(\frac{1+\theta}{2}+\hat{h}_{t,\delta})\right)-\Phi\left(\Phi^{-1}(\frac{1-\theta}{2}-\hat{h}_{t,\delta})\right)-2\hat{h}_{t,\delta}=\theta.\end{aligned}$$

Hence by definition of $l(\cdot)$, Inequality (11.18) implies

$$l(\theta,\mathbf{w}^\top\mathbf{x}_1,\cdots,\mathbf{w}^\top\mathbf{x}_t)\leq2\Phi^{-1}(\frac{1+\theta}{2}+\hat{h}_{t,\delta});\quad\forall\|\mathbf{w}\|_2=1,\ \theta\in[0,1].$$

Next, notice that Inequality (11.18) also implies

$$\frac{1}{t}\sum_{i=1}^{t}\mathbf{1}_{b^-\le\mathbf{w}^\top\mathbf{x}_i\le b^+}\le\Phi(b^+)-\Phi(b^-)+2\hat{h}_{t,\delta};\quad\forall\|\mathbf{w}\|_2=1,\,b^-\le b^+.$$

Thus, by definition of $l(\cdot)$, Inequality (11.18) implies $\forall\|\mathbf{w}\|_2=1$, $\theta\in[0,1]$,

$$l(\theta,\mathbf{w}^\top\mathbf{x}_1,\cdots,\mathbf{w}^\top\mathbf{x}_t)=\min\{b^+-b^-\big|\frac{1}{t}\sum_{i=1}^{t}\mathbf{1}_{b^-\le\mathbf{w}^\top\mathbf{x}_i\le b^+}\ge\theta\}$$

$$\ge\min\{b^+-b^-\big|\Phi(b^+)-\Phi(b^-)+2\hat{h}_{t,\delta}\ge\theta\}=2\Phi^{-1}\left(\frac{1+\theta}{2}-\hat{h}_{t,\delta}\right).$$

Finally, recalling that Inequality (11.18) holds with probability at least $1-\delta$, the proof of the theorem is complete. □

11.9.2. Proof of Theorem 11.3.

PROOF. Recall that $\mathbf{z}_i=A\mathbf{x}_i+\mathbf{n}_i$. Then we have,

$$\sup_{\mathbf{w}\in\mathbb{R}^m,\|\mathbf{w}\|_2=1}\big|1/t\sum_{i=1}^{t}(\mathbf{w}^\top\mathbf{z}_i)^2-\mathbf{w}^\top(AA^\top+I_m)\mathbf{w}\big|$$

$$=\sup_{\mathbf{w}\in\mathbb{R}^m,\|\mathbf{w}\|_2=1}\big|1/t\sum_{i=1}^{t}[\mathbf{w}^\top(A\mathbf{x}_i+\mathbf{n}_i)]^2-\mathbf{w}^\top(AA^\top+I_m)\mathbf{w}\big|$$

$$=\sup_{\mathbf{w}\in\mathbb{R}^m,\|\mathbf{w}\|_2=1}\big|1/t\sum_{i=1}^{t}\left\{\mathbf{w}^\top A\mathbf{x}_i\mathbf{x}_i^\top A^\top\mathbf{w}+\mathbf{w}^\top(A\mathbf{x}_i\mathbf{n}_i^\top+\mathbf{n}_i\mathbf{x}_i^\top A^\top)\mathbf{w}+\mathbf{w}^\top\mathbf{n}_i\mathbf{n}_i^\top\mathbf{w}\right\}$$
$$-\mathbf{w}^\top(AA^\top+I_m)\mathbf{w}\big|$$

$$\le\sup_{\mathbf{w}\in\mathbb{R}^m,\|\mathbf{w}\|_2=1}\big|1/t\sum_{i=1}^{t}\mathbf{w}^\top A\mathbf{x}_i\mathbf{x}_i^\top A^\top\mathbf{w}-\mathbf{w}^\top AA^\top\mathbf{w}\big|$$
$$+\sup_{\mathbf{w}\in\mathbb{R}^m,\|\mathbf{w}\|_2=1}\big|1/t\sum_{i=1}^{t}\mathbf{w}^\top\mathbf{n}_i\mathbf{n}_i^\top\mathbf{w}-\mathbf{w}^\top I_m\mathbf{w}\big|+\sup_{\mathbf{w}\in\mathbb{R}^m,\|\mathbf{w}\|_2=1}\big|2/t\sum_{i=1}^{t}\mathbf{w}^\top A\mathbf{x}_i\mathbf{n}_i^\top\mathbf{w}\big|.$$

Next, we bound each of the three terms in the last expression above.

Bounding Term 1: When Condition (B) holds, we have

$$\begin{aligned}
&\sup_{\mathbf{w}\in\mathbb{R}^m,\|\mathbf{w}\|_2=1} \left|1/t\sum_{i=1}^{t}\mathbf{w}^\top A\mathbf{x}_i\mathbf{x}_i^\top A^\top\mathbf{w} - \mathbf{w}^\top AA^\top\mathbf{w}\right| \\
=&\sup_{\mathbf{w}\in\mathbb{R}^m,\|\mathbf{w}\|_2=1} \left|\mathbf{w}^\top A(1/t\sum\mathbf{x}_i\mathbf{x}_i^\top - I_d)A^\top\mathbf{w}\right| \\
\leq&\lambda_{\max}(AA^\top)\sup_{\mathbf{w}'\in\mathbb{R}^d,\|\mathbf{w}'\|_2=1}\left|\mathbf{w}'^\top(1/t\sum\mathbf{x}_i\mathbf{x}_i^\top - I_d)\mathbf{w}'\right| \leq \hat{c}_{t,\delta}\lambda_{\max}(AA^\top).
\end{aligned}$$

Here, the first inequality holds due to the fact that

$$\|A^\top\mathbf{w}\|_2 \leq \|A^\top\|_2 = \sigma_{\max}(A) = \sqrt{\lambda_{\max}(AA^\top)}, \quad \forall\mathbf{w}\in\mathbb{R}^m, \|\mathbf{w}\|_2 = 1;$$

and Condition (B) implies the second inequality.

Bounding Term 2: When Condition (A) holds, we have

$$\begin{aligned}
&\sup_{\mathbf{w}\in\mathbb{R}^m,\|\mathbf{w}\|_2=1} \left|1/t\sum_{i=1}^{t}\mathbf{w}^\top\mathbf{n}_i\mathbf{n}_i^\top\mathbf{w} - \mathbf{w}^\top I_m\mathbf{w}\right| \\
=&\sup_{\mathbf{w}\in\mathbb{R}^m,\|\mathbf{w}\|_2=1} \left|1/t\sum_{i=1}^{t}\mathbf{w}^\top\mathbf{n}_i\mathbf{n}_i^\top\mathbf{w} - 1\right| \\
=&\max\left[\lambda_{\max}(\frac{1}{t}\sum_{i=1}^{t}\mathbf{n}_i\mathbf{n}_i^\top) - 1, 1\right] \\
\leq&\overline{\lambda}_{t,\delta} - 1.
\end{aligned}$$

Here, the last inequality holds since $\overline{\lambda}_{t,\delta} \geq 4/(1-\overline{\eta}) > 2$.

Bounding Term 3: By Hölder's inequality, for unit-norm $\mathbf{w}_0$ we have,

$$
\begin{aligned}
&1/t\sum_{i=1}^{t}|(\mathbf{w}_0^\top A\mathbf{x}_i)(\mathbf{n}_i^\top\mathbf{w}_0)| \le 1/t\sqrt{\sum_{i=1}^{t}|\mathbf{w}_0^\top A\mathbf{x}_i|^2}\sqrt{\sum_{i=1}^{t}|\mathbf{n}_i^\top\mathbf{w}_0|^2}\\
=&\sqrt{1/t\sum_{i=1}^{t}|\mathbf{w}_0^\top A\mathbf{x}_i|^2}\sqrt{1/t\sum_{i=1}^{t}|\mathbf{n}_i^\top\mathbf{w}_0|^2}\\
\le&\sqrt{\sup_{\|\mathbf{w}_1\|_2=1}(1/t\sum_{i=1}^{t}\mathbf{w}_1^\top A\mathbf{x}_i\mathbf{x}_i^\top A^\top\mathbf{w}_1)}\sqrt{\sup_{\|\mathbf{w}_2\|_2=1}(1/t\sum_{i=1}^{t}\mathbf{w}_2^\top\mathbf{n}_i\mathbf{n}_i^\top\mathbf{w}_2)}\\
\le&\sqrt{\lambda_{\max}(AA^\top)\lambda_{\max}(1/t\sum_{i=1}^{t}\mathbf{x}_i\mathbf{x}_i^\top)}\sqrt{\lambda_{\max}(1/t\sum_{i=1}^{t}\mathbf{n}_i\mathbf{n}_i^\top)}\\
\le&\sqrt{\lambda_{\max}(AA^\top)}\sqrt{1+\sqrt{\frac{6d^2}{t}\left(\ln 2d^2+\ln\frac{1}{\delta}\right)}}\sqrt{\overline{\lambda}_{t,\delta}}\\
\le&\sqrt{\lambda_{\max}(AA^\top)}\left[1+\sqrt{\frac{3d^2}{2t}\left(\ln 2d^2+\ln\frac{1}{\delta}\right)}\right]\left[\sqrt{\overline{\lambda}_{t,\delta}}\right]\\
=&\Big[1+\frac{\hat{c}_{t,\delta}}{2}\Big]\sqrt{\overline{\lambda}_{t,\delta}\lambda_{\max}(AA^\top)}.
\end{aligned}
$$

The last two inequalities hold due to the assumption that Conditions (A) and (B) hold and $n \ge 4$.

The theorem follows by summing up all three terms. □

11.9.3. Proof of Theorem 11.4.

PROOF. It suffices to show the inequalities hold for any $c > 0$. We prove the following lemma before proceeding to prove the theorem.

LEMMA 11.16. *Given* $\mathbf{v}_1, \cdots, \mathbf{v}_t \in \mathbb{R}^m$, $\theta \in (0,1)$, *unit-norm* $\mathbf{w}$, *and* $c > 0$, *Condition (A) implies*

$$
\begin{aligned}
&l(\theta, \mathbf{w}^\top(\mathbf{v}_1+\mathbf{n}_1), \cdots, \mathbf{w}^\top(\mathbf{v}_t+\mathbf{n}_t)) - 2c \le l(\theta+\overline{\lambda}_{t,\delta}/c^2, \mathbf{w}^\top\mathbf{v}_1, \cdots, \mathbf{w}^\top\mathbf{v}_t);\\
&l(\theta, \mathbf{w}^\top(\mathbf{v}_1-\mathbf{n}_1), \cdots, \mathbf{w}^\top(\mathbf{v}_t-\mathbf{n}_t)) - 2c \le l(\theta+\overline{\lambda}_{t,\delta}/c^2, \mathbf{w}^\top\mathbf{v}_1, \cdots, \mathbf{w}^\top\mathbf{v}_t).
\end{aligned}
\tag{11.19}
$$

PROOF. Consider the following optimization problem:

$$\begin{aligned} \text{minimize:} \quad & l^+ - l^- \\ \text{subject to:} \quad & \sum_{i=1}^{t} \mathbf{1}_{l^- \le \mathbf{w}^\top \mathbf{v}_i \le l^+} \ge (\theta + \overline{\lambda}_{t,\delta}/c^2)t. \end{aligned}$$

Let l_0^- and l_0^+ be the optimal solution. By definition of $l(\cdot)$,

$$l_0^+ - l_0^- = l(\theta + \overline{\lambda}_{t,\delta}/c^2, \mathbf{w}^\top \mathbf{v}_1, \cdots, \mathbf{w}^\top \mathbf{v}_t).$$

Notice that for all i satisfying both $l_0^- \le \mathbf{w}^\top \mathbf{v}_i \le l_0^+$, and $|\mathbf{w}^\top \mathbf{n}_i| \le c$, the following holds:

$$l_0^- - c \le \mathbf{w}^\top (\mathbf{v}_i \pm \mathbf{n}_i) \le l_0^+ + c.$$

That is,

$$\sum_{i=1}^{t} \mathbf{1}_{l_0^- \le \mathbf{w}^\top \mathbf{v}_i \le l_0^+ \& |\mathbf{w}^\top \mathbf{n}_i| \le c} \le \sum_{i=1}^{t} \mathbf{1}_{l_0^- - c \le \mathbf{w}^\top (\mathbf{v}_i \pm \mathbf{n}_i) \le l_0^+ + c},$$

which implies

$$\sum_{i=1}^{t} \mathbf{1}_{l_0^- \le \mathbf{w}^\top \mathbf{v}_i \le l_0^+} - \sum_{i=1}^{t} \mathbf{1}_{|\mathbf{w}^\top \mathbf{n}_i| > c} \le \sum_{i=1}^{t} \mathbf{1}_{l_0^- - c \le \mathbf{w}^\top (\mathbf{v}_i \pm \mathbf{n}_i) \le l_0^+ + c}.$$

Notice that by definition, for all unit norm $\mathbf{w} \in \mathbb{R}^m$

$$\lambda_{\max}(\sum_{i=1}^{t} \mathbf{n}_i \mathbf{n}_i^\top) \ge \sum_{i=1}^{t} (\mathbf{w}^\top \mathbf{n}_i)^2.$$

Hence,

$$\sum_{i=1}^{t} \mathbf{1}_{|\mathbf{w}^\top \mathbf{n}_i| \ge c} \le \lambda_{\max}(\sum_{i=1}^{t} \mathbf{n}_i \mathbf{n}_i^\top)/c^2.$$

Thus, Condition (A) implies

$$\sum_{i=1}^{t} \mathbf{1}_{|\mathbf{w}^\top \mathbf{n}_i| \ge c} \le t\overline{\lambda}_{t,\delta}/c^2, \quad \Longrightarrow \quad \sum_{i=1}^{t} \mathbf{1}_{l_0^- - c \le \mathbf{w}^\top (\mathbf{v}_i \pm \mathbf{n}_i) \le l_0^+ + c} \ge \theta.$$

By definition of $l(\cdot)$, we have

$$l(\theta, \mathbf{w}^\top (\mathbf{v}_1 \pm \mathbf{n}_1), \cdots, \mathbf{w}^\top (\mathbf{v}_t \pm \mathbf{n}_t)) \le l_0^+ + c - (l_0^- - c) = v_0 + 2c,$$

which establishes the lemma. □

Now we proceed to prove the theorem. By a straight-forward application of Lemma 11.16, we have that Conditions (A) and (C) imply

$$\begin{aligned} & l(\theta - \overline{\lambda}_{t,\delta}/c^2, \mathbf{w}^\top A\mathbf{x}_1, \cdots, \mathbf{w}^\top A\mathbf{x}_t) - 2c \\ \leq\ & l(\theta, \mathbf{w}^\top \mathbf{z}_1, \cdots, \mathbf{w}^\top \mathbf{z}_t) \\ \leq\ & l(\theta + \overline{\lambda}_{t,\delta}/c^2, \mathbf{w}^\top A\mathbf{x}_1, \cdots, \mathbf{w}^\top A\mathbf{x}_t) + 2c. \end{aligned} \tag{11.20}$$

Next, notice that

$$\begin{aligned} & \sup_{\mathbf{w}\in\mathbb{R}^m, \|\mathbf{w}\|_2=1} \left\{ 2\Phi^{-1}(\frac{1+\theta}{2} - \frac{\overline{\lambda}_{t,\delta}}{2c^2})\|\mathbf{w}^\top A\|_2 - l(\theta - \frac{\overline{\lambda}_{t,\delta}}{c^2}, \mathbf{w}^\top A\mathbf{x}_1, \cdots, \mathbf{w}^\top A\mathbf{x}_t) \right\} \\ \leq & \sup_{\|\mathbf{w}^\top A\|_2 \leq \sqrt{\lambda_{\max}(AA^\top)}} \left\{ 2\Phi^{-1}(\frac{1+\theta}{2} - \frac{\overline{\lambda}_{t,\delta}}{2c^2})\|\mathbf{w}^\top A\|_2 - l(\theta - \frac{\overline{\lambda}_{t,\delta}}{c^2}, \mathbf{w}^\top A\mathbf{x}_1, \cdots, \mathbf{w}^\top A\mathbf{x}_t) \right\} \\ \leq & \sqrt{\lambda_{\max}(AA^\top)} \sup_{\mathbf{w}'\in\mathbb{R}^d, \|\mathbf{w}'\|_2\leq 1} \left\{ 2\Phi^{-1}(\frac{1+\theta}{2} - \frac{\overline{\lambda}_{t,\delta}}{2c^2}) - l(\theta - \frac{\overline{\lambda}_{t,\delta}}{c^2}, \mathbf{w}'^\top \mathbf{x}_1, \cdots, \mathbf{w}'^\top \mathbf{x}_t) \right\}, \end{aligned} \tag{11.21}$$

where the last inequality holds because $l(\delta, \lambda c_1, \cdots, \lambda c_t) = \lambda(\delta, c_1, \cdots, c_t)$ for any $\lambda > 0$.

Condition (C) implies that for any unit norm $\mathbf{w}' \in \mathbb{R}^d$,

$$2\Phi^{-1}(\frac{1+\theta}{2} - \frac{\overline{\lambda}_{t,\delta}}{2c^2} - \hat{h}_{t,\delta}) \leq l(\theta - \frac{\overline{\lambda}_{t,\delta}}{c^2}, \mathbf{w}'^\top \mathbf{x}_1, \cdots, \mathbf{w}'^\top \mathbf{x}_t).$$

Substituting it into Equation (11.21) leads to the following inequality

$$\begin{aligned} & l(\theta - \frac{\overline{\lambda}_{t,\delta}}{c^2}, \mathbf{w}^\top A\mathbf{x}_1, \cdots, \mathbf{w}^\top A\mathbf{x}_t) \\ \geq & 2\Phi^{-1}(\frac{1+\theta}{2} - \frac{\overline{\lambda}_{t,\delta}}{2c^2})\|\mathbf{w}^\top A\|_2 \\ & - \sqrt{\lambda_{\max}(AA^\top)}\Big(2\Phi^{-1}(\frac{1+\theta}{2} - \frac{\overline{\lambda}_{t,\delta}}{2c^2}) - 2\Phi^{-1}(\frac{1+\theta}{2} - \frac{\overline{\lambda}_{t,\delta}}{2c^2} - h_{t,\delta})\Big). \end{aligned}$$

Substituting it into Inequality (11.20) implies the first part of Inequality (11.4). The proof of the second part is identical and hence omitted. □

CHAPTER 12

Conclusion

This thesis studies decision making methodologies in the spirit of Robust Optimization and investigates both theoretic and algorithmic applications of robust decision making into machine learning field. Section 1.3 gives a fairly detailed account of the contribution of this thesis. In this chapter we provide a brief overview of what we have learnt and what issues are open, and need to be addressed in future research.

12.1. Summary of contributions

In Chapter 2- 5 we addressed two limitations of robust optimization, namely, a lack of theoretical justification and conservatism in sequential decision making.

We provided an axiomatic justification of robust optimization based on the MaxMin Expected Utility framework from decision theory. This not only provides a more solid justification and motivation of robust optimization, but also suggests a new approach for choosing the uncertainty set by exploring the distributional requirement.

We studied a less conservative decision criterion for uncertain Markov decision processes. In particular, we considered the nested-set structured parameter uncertainty to model the probabilistic information of the parameters and proposed to find the strategy that achieves maxmin expected utility. Such formulation leads to

tractable solutions that have an appealing interpretation as trading-off the likely performance and robustness, hence mitigating the conservatism of the standard robust Markov decision processes.

We investigated a sequential decision making setup that can be modeled using Markov decision processes whereas each strategy is evaluated comparatively by its *parameter regret*, i.e., the gap between its performance and the optimum. Under parameter uncertainty, two formulations – minimax regret and mean-variance tradeoff of the regret – were proposed and their computational cost studied.

We proposed a Kalman filter design based on trading-off the likely performance and the robustness under parameter uncertainty. The proposed filter can be computed efficiently online, is steady-state stable, and is less conservative than the robust filter proposed in [**130**]. Simulation studies showed that the proposed filter achieves satisfactory performance under a wider range of parameters than both the standard Kalman filter and the robust filter.

In Chapter 6- 11 we applied robust decision making into machine learning on both the theoretic and the algorithmic front.

On the theoretic front, we showed that the concept of robustness is essential to "successful" learning. In particular, we proved that both SVM and Lasso are special cases of robust optimization, and such robustness interpretation implies consistency and sparsity naturally. We further established a more general duality between robustness and generalizability: indeed, the former is a necessary and sufficient condition to the latter for an arbitrary learning algorithm. Thus, we provided an answer to the fundamental question of what makes a learning algorithm work.

We proved a theorem saying that two widely used concepts, namely *sparsity* and *algorithmic stability* conflict with each other. This theorem provides us with additional insight into these concepts and their interrelation, and it furthermore implies that a tradeoff between these two concepts is unavoidable in designing learning algorithms.

On the algorithmic front, we designed novel robust learning algorithms. For the binary classification task, we developed a robust classifier with controlled conservatism by extending robust SVM [**27**][**137**] to a soft notion of robustness known as comprehensive robustness. For the dimensionality reduction task, we investigated the case that outlying observation exists and the dimensionality is comparable to the number of observations, a case where standard robust PCA algorithms break. We proposed a HR-PCA algorithm based on an "actor-critic" scheme. The HR-PCA algorithm is tractable, robust to outlier, easily kernelizable, and has a bounded deviation that converges to zero in the limit case where the proportion of corrupted points goes to zero.

12.2. Open problems and future research

The work reported in this thesis has raised many problems to be studied in the future. We list in this section some of the immediate questions to direct future works after this thesis.

Parameter regret for MDPs with uncertain transition probabilities. In Chapter 4 we investigated parametric regret for MDPs where only the reward parameters are subject to uncertainty. A more general case where the transition parameters are uncertain surely merits study. As pointed out in Chapter 4, parameter regret in the general uncertain MDP incurs significant computational issues. Therefore, we want to investigate under what conditions the optimal strategy can be found or approximated in polynomial-time.

Applications of the MMEU-based uMDP and regret-based uMDP. In Chapter 3 and Chapter 4 we proposed decision criteria for uncertain Markov decision processes based on MaxMin Expected Utility and parameter regret, respectively. Under favorable conditions, both formulations can be solved in a computational efficient manner. In this case, the formulations and algorithms would have applications to handle real-life decision making. One notable example is portfolio optimization in

finance, where the problem is inherently comparative, and hence particularly suitable for a regret-based formulation.

Tighter PAC bound using the robust interpretation. In Chapter 6 we proved that SVM is a special case of robust optimization, and hence provided an interpretation of regularization schemes from a robustness perspective. We further showed that such interpretation leads to statistical consistency. Consistency essentially means that the algorithm converges to the optimal solution asymptotically. An immediate question afterward, is how fast is the convergence, i.e., the rate. In machine learning, the rate is quantified using the Probably Approximately Correct (PAC) framework proposed in [**153**]. For SVM, PAC bounds have been extensively studied based on approaches such as algorithmic stability and Rademacher complexity (e.g., [**8, 7, 32**]). It is of significant interest to investigate whether tighter PCA bound can be obtained through the robustness argument.

General relationship between sparsity and robustness. In Chapter 7 we established the sparsity of Lasso from its robustness interpretation. The proof is based on the fact that the uncertainty set of the robust formulation that is equivalent to Lasso is feature-wise uncoupled. On the other hand, feature-wise coupled uncertainty set often leads to non-sparse solutions. For example, in [**64**], it is shown that a similar robust regression formulation with the uncertainty set constrained by the Frobenius norm of the perturbation is equivalent to the Tikhonov regularized regression, which is known to be non-sparse. Therefore, we conjecture that being feature-wise uncoupled is crucial to achieve sparseness for more general algorithms. In particular, future research along this line should include both theoretic developments and new sparse algorithms based on robust optimization w.r.t. feature-wise uncertainty set.

New robust learning algorithms. In Chapter 10 and Chapter 11 we proposed learning algorithms that are robust to input perturbations and outlying observations for classification and dimensionality reduction tasks. Indeed, robustness is desirable

for a number of other machine learning problems including regression, clustering, rank learning and active learning. Therefore, we are interested in applying robust decision making to these tasks and designing new learning algorithms that have good empirical performance and are robust to perturbations and outliers.

REFERENCES

[1] N. Alon, S. Ben-David, N. Cesa-Bianchi, and D. Haussler. Scale-sensitive dimension, uniform convergence, and learnability. *Journal of the ACM*, 44(4):615–631, 1997.

[2] M. Anthony and P. L. Bartlett. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, 1999.

[3] A. Asuncion and D. J. Newman. UCI machine learning repository, 2007.

[4] T. Başar and P. Bernhard. *H_∞-Optimal Control and Related Minimax Design Problems: A Dynamic Game Approach*. Birkhauser, 1991.

[5] A. Bagnell, A. Ng, and J. Schneider. Solving uncertain Markov decision problems. Technical Report CMU-RI-TR-01-25, Carnegie Mellon University, August 2001.

[6] P. L. Bartlett. The sample complexity of pattern classification with neural networks: The size of the weight is more important than the size of the network. *IEEE Transactions on Information Theory*, 44(2):525–536, 1998.

[7] P. L. Bartlett, O. Bousquet, and S. Mendelson. Local Rademacher complexities. *The Annals of Statistics*, 33(4):1497–1537, 2005.

[8] P. L. Bartlett and S. Mendelson. Rademacher and Gaussian complexities: Risk bounds and structural results. *The Journal of Machine Learning Research*, 3:463–482, November 2002.

[9] A. Ben-Tal, D. Bertsimas, and D. B. Brown. A soft robust model for optimization under ambiguity. Submitted, September 2006.

[10] A. Ben-Tal, S. Boyd, and A. Nemirovski. Extending scope of robust optimization: Comprehensive robust counterparts of uncertain problems. *Mathematical Programming, Series B*, 107:63–89, 2006.

[11] A. Ben-tal and A. Nemirovski. Robust convex optimization. *Mathematics of Operations Research*, 23:769–805, 1998.

[12] A. Ben-Tal and A. Nemirovski. Robust solutions of uncertain linear programs. *Operations Research Letters*, 25(1):1–13, August 1999.

[13] A. Ben-Tal and A. Nemirovski. Robust solutions of linear programming problems contaminated with uncertain data. *Mathematical Programming, Serial A*, 88:411–424, 2000.

[14] K. P. Bennett and O. L. Mangasarian. Robust linear programming discrimination of two linearly inseparable sets. *Optimization Methods and Software*, 1(1):23–34, 1992.

[15] D. P. Bertsekas and I. B. Rhodes. Recursive state estimation for a set-membership description of uncertainties. *IEEE Transactions on Automatic Control*, 16:117–128, 1971.

[16] D. P. Bertsekas and J. N. Tsitsiklis. *Neuro-Dynamic Programming*. Athena Scientific, 1996.

[17] D. Bertsimas and D. B. Brown. Constructing uncertainty sets for robust linear optimization. To appear in *Operations Research*, 2009.

[18] D. Bertsimas, D. B. Brown, and C. Caramanis. Theory and applications of robust optimization. Submitted, available from `http://users.ece.utexas.edu/~cmcaram/pubs/RobustOptimizationLV.pdf`, 2007.

[19] D. Bertsimas and A. Fertis. Personal Correspondence, March 2008.

[20] D. Bertsimas and I. Popescu. Optimal inequalities in probability theory: A convex optimization approach. *SIAM Journal of Optimization*, 15(3):780–800, 2001.

[21] D. Bertsimas and M. Sim. Robust discrete optimization and network flows. *Mathematical Programming, Series B*, 98:49–71, 2003.

[22] D. Bertsimas and M. Sim. The price of robustness. *Operations Research*, 52(1):35–53, January 2004.

[23] D. Bertsimas and M. Sim. Tractable approximations to robust conic optimization problems. *Mathematical Programming, Serial B*, 107(1):5–36, 2006.

[24] D. Bertsimas and J. N. Tsitsiklis. *Introduction to Linear Optimization*. Athena Scientific, 1997.

[25] C. Bhattacharyya. Robust classification of noisy data using second order cone programming approach. In *Proceedings International Conference on Intelligent Sensing and Information Processing*, pages 433–438, Chennai, India, 2004.

[26] C. Bhattacharyya, L. R. Grate, M. I. Jordan, L. El Ghaoui, and I. S. Mian. Robust sparse hyperplane classifiers: Application to uncertain molecular profiling data. *Journal of Computational Biology*, 11(6):1073–1089, 2004.

[27] C. Bhattacharyya, K. S. Pannagadatta, and A. J. Smola. A second order cone programming formulation for classifying missing data. In Lawrence K. Saul, Yair Weiss, and Léon Bottou, editors, *Advances in Neural Information Processing Systems (NIPS17)*, Cambridge, MA, 2004. MIT Press.

[28] J. Bi and T. Zhang. Support vector classification with input data uncertainty. In Lawrence K. Saul, Yair Weiss, and Léon Bottou, editors, *Advances in Neural Information Processing Systems (NIPS17)*, Cambridge, MA, 2004. MIT Press.

[29] J. R. Birge and F. Louveaux. *Introduction to Stochastic Programming*. Springer-Verlag, New York, 1997.

[30] C. M. Bishop. Training with noise is equivalent to Tikhonov regularization. *Neural Computation*, 7(1):108–116, 1995.

[31] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory*, pages 144–152, New York, NY, 1992.

[32] O. Bousquet and A. Elisseeff. Stability and generalization. *The Journal of Machine Learning Research*, 2:499–526, 2002.

[33] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[34] D. B. Brown. Large deviations bounds for estimating conditional value-at-risk. *Operations Research Letters*, 35:722–730, 2007.

[35] A. Bryson Jr. and Y. C. Ho. *Applied Optimal Control: Optimization, Estimation and Control*. John Willey & Sons, 1975.

[36] E. J. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2):489–509, 2006.

[37] E. J. Candès and T. Tao. The Dantzig selector: Statistical estimation when p is much larger than n. *The Annals of Statistics*, 35(6):2313–2351, 2007.

[38] S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20(1):33–61, 1999.

[39] A. Christmann and I. Steinwart. On robustness properties of convex risk minimization methods for pattern recognition. *The Journal of Machine Learning Research*, 5:1007–1034, 2004.

[40] A. Christmann and I. Steinwart. Consistency and robustness of kernel based regression in convex risk minimization. *Bernoulli*, 13(3):799–819, 2007.

[41] A. Christmann and A. Van Messem. Bouligand derivatives and robustness of support vector machines for regression. *The Journal of Machine Learning Research*, 9:915–936, 2008.

[42] R. R. Coifman and M. V. Wickerhauser. Entropy-based algorithms for best basis selection. *IEEE Transactions on Information Theory*, 38(2):713–718, 1992.

[43] C. Cortes and V. N. Vapnik. Support vector networks. *Machine Learning*, 20:1–25, 1995.

[44] C. Croux, P. Filzmoser, and M. Oliveira. Algorithms for Projection−Pursuit robust principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 87(2):218–225, 2007.

[45] C. Croux and G. Hasebroeck. Principal component analysis based on robust estimators of the covariance or correlation matrix: Influence functions and efficiencies. *Biometrika*, 87(3):603–618, 2000.

[46] A. d'Aspremont, L El Ghaoui, M. I. Jordan, and G. R. Lanckriet. A direct formulation for sparse PCA using semidefinite programming. *SIAM Review*, 49(3):434–448, 2007.

[47] F. De la Torre and M. J. Black. Robust principal component analysis for computer vision. In *Proceedings of the Eighth International Conference on Computer Vision (ICCV'01)*, pages 362–369, 2001.

[48] F. De la Torre and M. J. Black. A framework for robust subspace learning. *International Journal of Computer Vision*, 54(1/2/3):117–142, 2003.

[49] R. Dearden, N. Friedman, and S. Russell. Bayesian Q-learning. In *Proceedings of the fifteenth national/tenth conference on Artificial intelligence/Innovative applications of artificial intelligence*, pages 761–768, Menlo Park, CA, USA, 1998. American Association for Artificial Intelligence.

[50] E. Delage and S. Mannor. Percentile optimization for Markov decision processes with parameter uncertainty. To appear in *Operations Research*, 2009.

[51] S. J. Devlin, R. Gnanadesikan, and J. R. Kettenring. Robust estimation of dispersion matrices and principal components. *Journal of the American Statistical Association*, 76(374):354–362, 1981.

[52] L. Devroye. Exponential inequality in nonparametric estimation. In *Nonparametric Functional Estimation and Related Topics*, pages 31–44, 1991.

[53] L. Devroye and L. Györfi. *Nonparametric Density Estimation: the l_1 View*. John Wiley & Sons, 1985.

[54] L. Devroye, L. Györfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*. Springer, New York, 1996.

[55] L. Devroye and T. Wagner. Distribution-free inequalities for the deleted and holdout error estimates. *IEEE Transactions of Information Theory*, 25(2):202–207, 1979.

[56] L. Devroye and T. Wagner. Distribution-free performance bounds for potential function rules. *IEEE Transactions of Information Theory*, 25(2):601–604, 1979.

[57] D. L. Donoho. High-dimensional data analysis: The curses and blessings of dimensionality. American Math. Society Lecture—Math. Challenges of the 21st Century, 2000.

[58] D. L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, 2006.

[59] J. L. Doob. *Stochastic Processes*. Wiley, New York, 1953.

[60] R. Durrett. *Probability: Theory and Examples*. Duxbury Press, 2004.

[61] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *The Annals of Statistics*, 32(2):407–499, 2004.

[62] M. Ehrgott. *Multicriteria Optimization*. Springer-Verlag Berlin Heidelberg, 2000.

[63] L. El Ghaoui and G. Calafiore. Robust filtering for discrete-time systems with bounded noise and parametric uncertaitny. *IEEE Transactions on Automatic Control*, 46(7):1084–1089, 2001.

[64] L. El Ghaoui and H. Lebret. Robust solutions to least-squares problems with uncertain data. *SIAM Journal on Matrix Analysis and Applications*, 18:1035–1064, 1997.

[65] L. El Ghaoui, F. Oustry, and H. Lebret. Robust solutions to uncertain semidefinite programs. *SIAM Journal on Optimization*, 9(1):33–52, 1998.

[66] L. Elden. Perturbation theory for the least square problem with linear equality constraints. *SIAM Journal on Numerical Analysis*, 17(3):338–350, 1980.

[67] L. G. Epstein and M. Schneider. Learning under ambiguity. *Review of Economic Studies*, 74(4):1275–1303, 2007.

[68] E. Even-Dar, S. Mannor, and Y. Mansour. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of Machine Learning Research*, 7:1079–1105, 2006.

[69] T. Evgeniou, M. Pontil, and A. Elisseeff. Leave one out error, stability, and generalization of voting combinations of classifiers. *Machine Learning*, 55(1):71–97, 2004.

[70] T. Evgeniou, M. Pontil, and T. Poggio. Regularization networks and support vector machines. In A. J. Smola, P. L. Bartlett, B. Schölkopf, and D. Schuurmans, editors, *Advances in Large Margin Classifiers*, pages 171–203, Cambridge, MA, 2000. MIT Press.

[71] A. Feuer and A. Nemirovski. On sparse representation in pairs of bases. *IEEE Transactions on Information Theory*, 49(6):1579–1581, 2003.

[72] R. D. Fierro and J. R. Bunch. Collinearity and total least squares. *SIAM Journal on Matrix Analysis and Applications*, 15:1167–1181, 1994.

[73] H. Föllmer and A. Schied. Convex measures of risk and trading constraints. *Finance and Stochastics*, 6:429–447, 2002.

[74] D. P. Foster and E. I. George. The risk inflation criterion for multiple regression. *The Annals of Statistics*, 22:1947–1975, 1994.

[75] Y. Freund and R. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

[76] A. Gelb, editor. *Applied Optimal Estimation*. MIT Press, 1974.

[77] I. Gilboa and D. Schmeidler. Maxmin expected utility with a non-unique prior. *Journal of Mathematical Economics*, 18:141–153, 1989.

[78] F. Girosi. An equivalence between sparse approximation and support vector machines. *Neural Computation*, 10(6):1445–1480, 1998.

[79] A. Globerson and S. Roweis. Nightmare at test time: Robust learning by feature deletion. In *ICML '06: Proceedings of the 23rd International Conference on Machine Learning*, pages 353–360, New York, NY, USA, 2006. ACM Press.

[80] P. W. Glynn and D. Ormoneit. Hoeffding's inequality for uniformly ergodic Markov chains. *Statistics and Probability Letters*, 56:143–146, 2002.

[81] G. H. Golub and C. F. Van Loan. *Matrix Computations*. John Hopkins University Press, Baltimore, 1989.

[82] M. S. Grewal and A. P. Andrews. *Kalman Filtering: Theory and Practice*. Prentice-Hall, 1993.

[83] M. Grötschel, L. Lovasz, and A. Schrijver. *The Ellipsoid Method and Combinatorial Optimization*. Springer, Heidelberg, 1988.

[84] F. Hampel. The influence curve and its role in robust estimation. *Journal of the American Statistical Association*, 69(346):383–393, 1974.

[85] F. R. Hampel, E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel. *Robust Statistics: The Approach Based on Influence Functions*. John Wiley & Sons, New York, 1986.

[86] J. Hannan. Approximation to Bayes risk in repeated play. *Contributions to the Theory of Games*, 3:97–139, 1957.

[87] B. Hassibi, A. H. Sayed, and T. Kailath. *Indefinite Quadratic Estimation and Control: A Unified Approach to $\mathcal{H}_2$ and $\mathcal{H}_\infty$ Theories*. SIAM, Philadelphia, 1999.

[88] D. J. Higham and N. J. Higham. Backward error and condition of structured linear systems. *SIAM Journal on Matrix Analysis and Applications*, 13:162–175, 1992.

[89] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with nerual networks. *Science*, 313:504–507, 2006.

[90] P. J. Huber. *Robust Statistics*. John Wiley & Sons, New York, 1981.

[91] G. N. Iyengar. Robust dynamic programming. *Mathematics of Operations Research*, 30(2):257–280, 2005.

[92] T. Jaakkola, M. Meila, and T. Jebara. Maximum entropy discrimination. In *Advances in Neural Information Processing Systems 12*, pages 470–476. MIT Press, 1999.

[93] I. M. Johnstone. On the distribution of the largest eigenvalue in principal components analysis. *The Annals of Statistics*, 29(2):295–327, 2001.

[94] I. T. Jolliffe. *Principal Component Analysis*. Springer Series in Statistics, Berlin: Springer, 1986.

[95] T. Kailath, A. H. Sayed, and B. Hassibi. *Linear Estimation*. Prentice-Hall, 2000.

[96] P. Kall and S. W. Wallace. *Stochastic Programming*. John Wiley & Sons, 1994.

[97] M. Kearns, Y. Mansour, A. Ng, and D. Ron. An experimental and theoretical comparison of model selection methods. *Machine Learning*, 27:7–50, 1997.

[98] D. Kelsey. Maxmin expected utility and weight of evidence. *Oxford Economic Papers*, 46:425–444, 1994.

[99] V. Koltchinskii and D. Panchenko. Empirical margin distributions and bounding the generalization error of combined classifiers. *The Annals of Statistics*, 30(1):1–50, 2002.

[100] S. Kutin and P. Niyogi. Almost-everywhere algorithmic stability and generalization error. In *UAI-2002: Uncertainty in Artificial Intelligence*, pages 275–282, 2002.

[101] G. R. Lanckriet, L. El Ghaoui, C. Bhattacharyya, and M. I. Jordan. A robust minimax approach to classification. *The Journal of Machine Learning Research*, 3:555–582, 2003.

[102] H. Levy and H. M. Markowtiz. Approximating expected utility by a function of mean and variance. *The American Economic Review*, 69(3):308–17, 1979.

[103] J. MacQueen. A modified dynamic programming method for Markov decision problems. *Journal of Mathematical Analysis and Application*, 14:38–43, 1966.

[104] S. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415, 1993.

[105] O. L. Mangasarian. Generalized support vector machines. In A. J. Smola, P. L. Bartlett, B. Schölkopf, and D. Schuurmans, editors, *Advances in Large Margin Classifiers*, pages 135–146. MIT Press, 2000.

[106] S. Mannor, D. Peleg, and R. Rubinstein. The cross entropy method for classification. In *Proceedings of the 22nd international conference on Machine learning*, pages 561–568, 2005.

[107] S. Mannor, D. Simester, P. Sun, and J. N. Tsitsiklis. Bias and variance approximation in value function estimates. *Management Science*, 53(2):308–322, 2007.

[108] R. A. Maronna, R. D. Martin, and V. J. Yohai. *Robust Statistics: Theory and Methods.* John Wiley & Sons, New York, 2006.

[109] A. W. Marshall and I. Olkin. Multivariate Chebyshev inequalities. *Annuals of Mathematical Statistics*, 31(4):1001–1014, 1960.

[110] C. McDiarmid. On the method of bounded differences. In *Surveys in Combinatorics*, pages 148–188, 1989.

[111] S. P. Meyn and R. L. Tweedie. *Markov Chains and Stochastic Stability.* Springer, New York, 1993.

[112] S. Mukherjee, P. Niyogi, T. Poggio, and R. Rifkin. Learning theory: Stability is sufficient for generalization and necessary and sufficient for consistency of empirical risk minimization. *Advances in Computational Mathematics*, 25(1-3):161–193, 2006.

[113] K. G. Murty. *Linear Programming.* John Wiley & Sons, 1983.

[114] K. M. Nagpal and P. P. Khargonekar. Filtering and smoothing in an h_∞-setting. *IEEE Transactions on Automatic Control*, 36:151–166, 1991.

[115] B. K. Natarajan. Sparse approximate solutions to linear systems. *SIAM Journal of Computation*, 24:227–234, 1995.

[116] A. Nilim and L. El Ghaoui. Robust control of Markov decision processes with uncertain transition matrices. *Operations Research*, 53(5):780–798, September 2005.

[117] C. H. Papadimitriou. *Computational Complexity.* Addison Wesley, 1994.

[118] E. Parzen. On the estimation of a probability density function and the mode. *The Annals of Mathematical Statistics*, 33:1065–1076, 1962.

[119] K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2(6):559–572, 1901.

[120] I. R. Petersen and D. C. McFarlane. Robust state estimation for uncertain systems. In *Proceedings: IEEE Conference on Decision and Control*, 1991.

[121] I. R .Petersen and A. V. Savkin. *Robust Kalman Filtering for Signals and Systems with Large Uncertainties*. Birkauser, 1999.

[122] T. Poggio, R. Rifkin, S. Mukherjee, and P. Niyogi. General conditions for predictivity in learning theory. *Nature*, 428(6981):419–422, 2004.

[123] A. Prékopa. *Stochastic Programming*. Kluwer, 1995.

[124] M. L. Puterman. *Markov Decision Processes*. John Wiley & Sons, New York, 1994.

[125] K. Regan and C. Boutlier. Regret-based reward elicitation for Markov decision processes. In *Proceedings of the Twenty-fifth Conference on Uncertainty in Artificial Intelligence (UAI-09),*, 2009.

[126] R.T. Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, N.J., 1970.

[127] M. Rosenblatt. Remarks on some nonparametric estimates of a density function. *The Annals of Mathematical Statistics*, 27:832–837, 1956.

[128] P. J. Rousseeuw. Multivariate estimation with high breakdown point. In W. Grossman, G. Pflug, I. Vincze, and W. Wertz, editors, *Mathematical Statistics and Applications*, pages 283–297. Reidel, Dordrecht, 1985.

[129] P. J. Rousseeuw and A. M. Leroy. *Robust Regression and Outlier Detection*. John Wiley & Sons, New York, 1987.

[130] A. H. Sayed. A framework for state-space estimation with uncertain models. *IEEE Transactions on Automatic Control*, 46(7):998–1013, July 2001.

[131] A. H. Sayed, V. H. Nascimento, and F. A. Cipparrone. A regularized robust design criterion for uncertain data. *SIAM Journal on Matrix Analysis and Its Applications*, 23(4):1120–1142, 2002.

[132] R. Schapire. Strength of weak learnability. *Machine Learning*, 5:192–227, 1990.

[133] B. Schölkopf and A. J. Smola. *Learning with Kernels*. MIT Press, 2002.

[134] B. Schölkopf, A. J. Smola, and K. R. Müller. Kernel principal component analysis. In B. Schölkopf, C. Burges, and A. J. Smola, editors, *Advances in kernel Methods – Support Vector Learning*, pages 327–352. MIT Press, Cambridge, MA, 1999.

[135] D. W. Scott. *Multivariate Density Estimation: Theory, Practice and Visualization*. John Wiley & Sons, New York, 1992.

[136] U. Shaked and Y. Theodor. $\mathcal{H}_\infty$-optimal estimation: A tutorial. In *Proceedings: IEEE Conference on Decision and Control*, pages 2278–2286, 1992.

[137] P. K. Shivaswamy, C. Bhattacharyya, and A. J. Smola. Second order cone programming approaches for handling missing and uncertain data. *The Journal of Machine Learning Research*, 7:1283–1314, July 2006.

[138] S. Singh, T. Jaakkola, and M. I. Jordan. Reinforcement learning with soft state aggregation. In G. Tesauro, D. Touretzky, and T. Leen, editors, *Advances in Neural Information Processing Systems*, volume 7, pages 361–368. The MIT Press, 1995.

[139] A. J. Smola, B. Schölkopf, and K. Müller. The connection between regularization operators and support vector kernels. *Neural Networks*, 11:637–649, 1998.

[140] H. W. Sorenson, editor. *Kalman Filtering: Theory and Application*. IEEE Press, 1985.

[141] A. L. Soyster. Convex programming with set-inclusive constraints and applications to inexact linear programming. *Operations Research*, 21:1154–1157, 1973.

[142] I. Steinwart. Consistency of support vector machines and other regularized kernel classifiers. *IEEE Transactions on Information Theory*, 51(1):128–142, 2005.

[143] M. Strens. A Bayesian framework for reinforcement learning. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 943–950, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.

[144] J.F. Sturm. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11–12:625–653, 1999. Special issue on Interior Point Methods (CD supplement with software).

[145] C. H. Teo, A. Globerson, S. Roweis, and A. J. Smola. Convex learning with invariances. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 1489–1496, Cambridge, MA, 2008. MIT Press.

[146] R. Tibshirani. Regression shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society, Series B*, 58(1):267–288, 1996.

[147] A. N. Tikhonov and V. Arsenin. *Solutions of Ill-Posed Problems*. Wiley, New York, 1977.

[148] C. A. Tracy and H. Widom. On orthogonal and symplectic matrix ensembles. *Communications in Mathematical Physics*, 177(3):727–754, 1996.

[149] C. A. Tracy and H. Widom. The distribution of the largest eigenvalue in the Gaussian ensembles. In J. van Diejen and L. Vinet, editors, *Calogero-Moser-Sutherland Models*, pages 461–472. Springer, New York, 2000.

[150] T. Trafalis and R. Gilbert. Robust support vector machines for classification and computational issues. *Optimization Methods and Software*, 22(1):187–198, February 2007.

[151] J. A. Tropp. Greed is good: Algorithmic results for sparse approximation. *IEEE Transactions on Information Theory*, 50(10):2231–2242, 2004.

[152] J. A. Tropp. Just relax: Convex programming methods for identifying sparse signals in noise. *IEEE Transactions on Information Theory*, 51(3):1030–1051, 2006.

[153] L. G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

[154] A. W. van der Vaart and J. A. Wellner. *Weak Convergence and Empirical Processes*. Springer-Verlag, New York, 2000.

[155] V. N. Vapnik. *Estimation of Dependences Based on Empirical Data*. Springer-Verlag, 1982.

[156] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, New York, 2000.

[157] V. N. Vapnik and A. Chervonenkis. *Theory of Pattern Recognition*. Nauka, Moscow, 1974.

[158] V. N. Vapnik and A. Chervonenkis. The necessary and sufficient conditions for consistency in the empirical risk minimization method. *Pattern Recognition and Image Analysis*, 1(3):260–284, 1991.

[159] V. N. Vapnik and A. Lerner. Pattern recognition using generalized portrait method. *Automation and Remote Control*, 24:744–780, 1963.

[160] M. J. Wainwright. Sharp thresholds for noisy and high-dimensional recovery of sparsity using ℓ_1-constrained quadratic programming. Technical Report Available from: `http://www.stat.berkeley.edu/tech-reports/709.pdf`, Department of Statistics, UC Berkeley, 2006.

[161] C. C. White III and H. K. El Deib. Markov decision processes with imprecise transition probabilities. *Operations Research*, 42(4):739–748, July 1992.

[162] C. C. White III and H. K. El-Deib. Parameter imprecision in finite state, finite action dynamic programs. *Operations Research*, 34(1):120–128, January 1986.

[163] L. Xie, Y. C. Soh, and C. E. De Souza. Robust Kalman filtering for uncertain discrete-time systems. *IEEE Transactions on Automatic Control*, 39:1310–1314, 1994.

[164] H. Xu, C. Caramanis, and S. Mannor. Robust dimensionality reduction for high-dimension data. In *Proceedings of Forty-Sixth Allerton Conference on Communication, Control, and Computing*, pages 1291–1298, 2008.

[165] H. Xu, C. Caramanis, and S. Mannor. Robust regression and Lasso. Submitted, 2008.

[166] H. Xu, C. Caramanis, and S. Mannor. Robust optimization and maxmin expected utility. In preparation, 2009.

[167] H. Xu, C. Caramanis, and S. Mannor. Robust regression and Lasso. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 1801–1808, 2009.

[168] H. Xu, C. Caramanis, and S. Mannor. Robustness and regularization of support vector machines. *The Journal of Machine Learning Research*, 10(Jul):1485–1510, 2009.

[169] H. Xu, C. Caramanis, S. Mannor, and S. Yun. Risk sensitive robust support vector machines. To appear in *Forty-Eighth IEEE Conference on Decision and Control*, 2009.

[170] H. Xu and S. Mannor. A Kalman filter design based on the performance/robustness tradeoff. In *Proceedings of Forty-Fifth Allerton Conference on Communication, Control, and Computing*, pages 59–63, 2007.

[171] H. Xu and S. Mannor. The robustness-performance tradeoff in Markov decision processes. In B. Schölkopf, J. C. Platt, and T. Hofmann, editors, *Advances in Neural Information Processing Systems 19*, pages 1537–1544. MIT Press, 2007.

[172] H. Xu and S. Mannor. A Kalman filter design based on performance/robustness tradeoff. *IEEE Transactions on Automatic Control*, 54(5):1171–1175, 2009.

[173] H. Xu and S. Mannor. The maxmin expected utility approach to uncertain Markov decision processes. Submitted, 2009.

[174] H. Xu and S. Mannor. Parametric regret in uncertain markov decision processes. To appear in *Forty-Eighth IEEE Conference on Decision and Control*, 2009.

[175] H. Xu, S. Mannor, and C. Caramanis. Sparse algorithms are not stable: A no-free-lunch theorem. In *Proceedings of Forty-Sixth Allerton Conference on Communication, Control, and Computing*, pages 1299 – 1303, 2008.

[176] H. Xu, S. Mannor, and C. Caramanis. Sparse algorithms are not stable. Submitted, 2009.

[177] L. Xu and A. L. Yuille. Robust principal component analysis by self-organizing rules based on statistical physics approach. *IEEE Transactions on Neural Networks*, 6(1):131–143, 1995.

[178] T. N. Yang and S. D. Wang. Robust algorithms for principal component analysis. *Pattern Recognition Letters*, 20(9):927–933, 1999.

[179] K. Zhou, J. C. Doyle, and K. Glover. *Robust and Optimal Control*. Prentice-Hall, Upper Saddle River, NJ, 1996.

[180] J. Zhu, S. Rosset, T. Hastie, and R. Tibshirani. 1-norm support vector machines. In *Advances in Neural Information Processing Systems 16*, 2003.

[181] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal Of The Royal Statistical Society Series B*, 67(2):301–320, 2005.

Document Log:

Manuscript Version 1 — 14 September 2009

Typeset by $\mathcal{AMS}$-LaTeX — 14 September 2009

Huan Xu

Center for Intelligent Machines, McGill University, 3480 University Street, Montréal (Québec) H3A 2A7, Canada

E-mail address: xuhuan@cim.mcgill.ca

Typeset by $\mathcal{AMS}$-LaTeX