Learning Without Labels: A Geometric Perspective on Invariant Contrastive Image Representations

Eric Zimmermann
Department of Electrical & Computer Engineering
McGill University, Montreal
April 2023

A thesis submitted to McGill University in partial fulfillment of the requirements of the degree of Master of Science

Abstract

This thesis presents a unified geometric perspective on learning contrastive image representations without access to labels that are transferable to a variety of downstream tasks in a deep learning setting. Inspired by the literature on normalized contrastive learning, we adopt the view of learning compressed image representations on a self-supervised pretext task constrained to a hyperspherical manifold in high dimensional space. We model perturbed image representations as elements embedded on the hypersphere and aim to learn diverse and invariant element configurations by comparing and contrasting them with one another. We explore the concepts of learning perturbation invariance through an alignment objective while preserving overall expressiveness of the representations using various diversity constraints. We commence by outlining distance functions as well as additional operations required to understand learning dynamics on the hypersphere. The contrastive diversity objective is then constructed using methods from probability theory and potential theory. We analyze the behavior of all methods and investigate the failure modes associated to them by describing their limitations in terms of element coupling. Each method is evaluated through extensive experimentation to empirically demonstrate that representation expressiveness is sufficiently modelled through the lens of minimum k-energy. Finally, we show that it is possible to efficiently minimize k-energy off the hypersphere in an unnormalized space.

Abrégé

Cette thèse présente une perspective géométrique unifiée sur l'apprentissage de représentations d'images contrastives sans accès à des étiquettes qui sont transférables à une variété de tâches en aval dans un cadre d'apprentissage profond. Inspiré par la littérature sur l'apprentissage contrastif normalisé, nous adoptons la méthode d'apprentissage de représentations d'images compressées sur une tâche de prétexte auto-supervisée contrainte à un collecteur hypersphérique dans un espace de haute dimension. Nous modélisons les représentations d'images perturbées comme des éléments intégrés à l'hypersphère et nous cherchons à apprendre des configurations d'éléments diverses et invariantes en les comparant et en les contrastant les unes avec les autres. Nous explorons les concepts d'apprentissage de l'invariance des perturbations par le biais d'un objectif d'alignement tout en préservant l'expressivité globale des représentations à l'aide de diverses contraintes de diversité. Nous commençons par donner une description générale des fonctions de distance ainsi que des opérations supplémentaires nécessaires pour comprendre la dynamique d'apprentissage sur l'hypersphère. L'objectif de diversité contrastive est ensuite construit à l'aide de méthodes issues de la théorie des probabilités et de la théorie du potentiel. Nous analysons le comportement de toutes les méthodes et étudions les modes de défaillance qui leur sont associés en décrivant leurs limites en termes de couplage d'éléments. Chaque méthode est évaluée par une expérimentation approfondie afin de démontrer empiriquement que l'expressivité de la représentation est suffisamment modélisée par le biais de l'énergie-k minimale. Enfin, nous démontrons qu'il est possible de minimiser efficacement l'énergie-k en dehors de l'hypersphère dans un espace non normalisé.

Acknowledgements

I would like to express my gratitude to my mother Orit, father Adrian, and brother Daniel for their love, support, and encouragement throughout my academic journey. Their belief in me and my abilities gave me the confidence to pursue and complete this degree, despite the many challenges faced along the way.

I am also deeply grateful to my supervisor, Professor Tal Arbel. Thank you for your support and guidance. I could not have achieved this milestone without you.

To my lab members and friends, Justin Szeto, Kirill Vasilevski, and Harley Wiltzer, thank you for being patient. Thank you for the discussions, and thank you for being by my side. I attribute my growth and success as a student to the many late nights and the endless support you all provided.

Contribution of Authors

The contributions presented in this thesis are related to various perspectives on contrastive image representation learning on and off the hypersphere. In particular, we propose two learning algorithms in the form of batch hyperspherical orthogonality and minimal energy constraints. The details of these methods are found in Chapter 4.

Contents

List of Figures

List of Tables

List of Acronyms

Abbreviation	Meaning
CNN	Convolutional Neural Network
DCL	Decoupled Contrastive Learning
DL	Deep Learning
DNC	Did Not Converge
KDE	Kernel Density Estimation
LSE	Log Sum Exponent
MLP	Multi-Layer Perceptron
MoCo	Momentum Contrastive Learning
MHE	Minimum Hyperspherical Energy
NN	Neural Network
NNCLR	Nearest Neighbour Contrastive Learning
NNC	Negative-negative Coupling
NPC	Negative-Positive Coupling
PPC	Positive-Positive Coupling
PS	Power Spherical Distribution
SGD	Stochastic Gradient Descent
SimCLR	Simple Contrastive Learning Representations
vMF	von Mises-Fisher Distribution

Notation

Symbol	Meaning
$\mathbb{R}$	The set of real numbers
$\mathbb{R}^n$	The set of real numbered n-dimensional vectors
$\mathbb{R}_+$	The set of non-negative real numbers
$\mathbb{S}_R^n$	The set of n-dimensional vectors on the hypersphere sphere of radius R
$\mathbb{D}_R^n$	The set of n-dimensional vectors in stereographic space of radius R
$\mathbb{E}^n$	The set of n-dimensional vectors Euclidean
$\mathbf{v}$	Boldface symbols such as $\mathbf{v}$ represent vectors
a	Lightface symbols such as a represent scalars
f_θ	The neural network f with parameters θ
M	The generic metric space
d_M	The metric or distance function on space M
$\|\| \cdot \|\|_p$	The p-norm
$\langle \cdot, \cdot \rangle_{\mathcal{F}}$	The inner product on generic space $\mathcal{F}$
$\mathcal{L}$	The loss function
$\mathcal{T}$	The set of image augmentations
B_R^n	The n dimensional open ball of radius R
$\mathcal{U}$	The uniform distribution
$\mathbf{E}$	The expectation function
$\mathbf{V}$	The variance function
∇	The gradient operator
ν	The Borel probability measure
$K(\cdot, \cdot)$	The kernel operation
$H(p)$	The entropy of a probability distribution p

1

Introduction

Deep learning is a subfield of artificial intelligence that is inspired by the structure and function of the human brain [1]. A deep learning system or model is called intelligent as it is able to learn complex representations of data that can be used to solve specialized tasks by minimizing an associated cost function. These systems are flexible and can be used in a variety of disciplines across many modalities for tasks that include image and speech recognition as well as natural language processing. Much of the success of deep learning in the past decade can be attributed to the availability of large labelled and uncurated sets of training data [1]. The most effective means of training a model on a dataset is through supervised learning. Supervised learning is a learning paradigm where a model learns relationships between an input data sample and a target task by minimizing the error between its predictions and the ground truth task labels. The model is able to extrapolate

to unseen data samples if it belongs to a similar distribution as the training data. The quality and generalizability of a learned model is therefore a function of the quality and diversity of the training dataset and its labels.

Unfortunately, it is often not possible to have access to large labelled datasets. Moreover, the cost and time requirements of annotating sufficiently many samples is often times too large. In order to minimize the barrier of entry to train a deep learning model on large sets of data, it is desirable to formulate a learning procedure which can distill useful information without the presence of a complete label distribution. Self-supervised learning is the study of learning task-independent representations of the data without having access to its labels. This procedure aims to synthesize pseudo-labels that can be learned in a supervised manner through the use of a pretext task. A pretext task is defined using information that is implicit to the sample itself. It is designed in a way whereby the type of information learned during training can be transferred to a downstream task like classification. If the representations learned from a pretext task are correlated to the features needed in a downstream task, it is possible to tune the model on a partial set of labels that is significantly smaller than what would normally be required to solve the downstream task.

We must first ask ourselves about what kind of information is contained in our data. In this thesis, we explore topics related to computer vision, with a focus on learning image representations for recognition-based tasks. It is possible to represent image samples in terms of their context, shape, and texture features. If our end goal requires us to learn a classifier on a set of partially labelled object-centric (single foreground object) images, it is possible to design a pretext task that extracts similar types of features from unlabelled data. Once learned, these features are likely to be transferable to said downstream task. Early image based self-supervised methods aimed to specify distinct pretext tasks that targeted specific subsets of features. These methods range from re-colorization of grayscale images as a means of learning object-color relationships, all the way to predicting manually induced rotations in an attempt to learn the spatial orientation of objects [2, 3]. Other methods focused on context prediction by shuffling patches within an image

and have a model predict the spatial relationships between each patch [4, 5]. As the types of pretext tasks continued to evolve, it became clear that a mixture of all feature types were of utmost importance.

Following the introduction of the targeted methods, contrastive self-supervised methods gave rise to a new family of algorithms [6, 7]. Contrastive methods extend the idea of a pretext task by comparing and contrasting noisy samples from one another. This method decomposes the learning problem into two core components. The first component is defined in terms of an invariance objective, where two noisy copies of the same sample are forced to have similar feature representations. The second component requires that a subset of the samples are sufficiently diverse and different from one another. In particular, we aim to study the computer vision problem of learning image representations without labels using normalized invariant contrastive image representations. The invariance objective learns to extract meaningful content within noisy copies of the same image regardless of the applied perturbation. The diversity objective contrasts random samples to one another. In order to learn what an object is, it is possible to learn about what it is not. Various methods have provided countless perspectives on how to best handle the diversity objective using different means of contrasting samples. Some methods have attempted to model the learning procedure in terms of a classification problem [6, 8, 9, 10], while other methods have attempted to model elements using hyperspherical energy [11].

We adopt the perspective presented by T. Chen [6] and T. Wang [11] by modeling image representations as elements on a high dimensional normalized space like the hypersphere. Many methods have chosen to model the contrastive problem on the hypersphere due to its training stability and superior performance when transferred to a downstream task like classification [6]. On the hypersphere, the diversity task can be modelled by enforcing a uniformity constraint over the set of image representations by contrasting them to one another. Uniformity is selected since it is the distribution with the highest possible diversity on the hypersphere. Uniformity can be achieved using techniques from potential theory, where pairs of elements are contrasted through their pairwise energy [11]. Moreover, uniformity has also been enforced by maximizing the empirical entropy

over a subset of elements [6, 8, 9]. Although these methods have seen great success, many assumptions are made about how to best model the normalized space as well as how to construct these energy and entropy estimators. There are also many more ways to present the problem of learning on the hypersphere that are rooted in how the space is described and what characteristics are needed. We aim to present a geometric perspective on how to learn contrastive image representations in hyperspherical space. We model the problem using two different views of the hypersphere and introduce different means of comparing elements using distance functions. We also provide details on how to model distributions and energy functions on the hypersphere and analyze the relationship between different methods. The goal of this work is to provide a complete perspective on how these methods relate to one another, as well as where they should be improved if possible. We show extended reasoning for how to better structure a variety of diversity objectives using the notion of element coupling, since certain samples in the dataset are likely to be coupled through a hidden class label that is unavailable when learning without labels. We present the reasoning for how to best select a distance function and how to model elements on the sphere. We then return to the initial problem of learning invariant contrastive image representations in unnormalized spaces. We show that the tools used to model the hyperspherical problem can be directly applied in the unnormalized space, and demonstrate that unnormalized contrastive techniques transfer as effectively as the normalized counterpart. We present empirical evidence for sets of experiments trained on a small benchmark dataset without labels and analyze the efficacy of each technique. We evaluate each method on a downstream classification task where labels are available, in order to determine the quality of the features learned without labels.

The following thesis is organized as follows. Chapter 2 introduces background theory, mathematical tools, and definitions that are required to define and measure distances on different geometric spaces. It introduces fundamental concepts related to how deep learning and neural networks operate, and provides further detail on the learning requirements in a self-supervised framework. It also discusses different self-supervised methodologies that expand beyond contrastive paradigms. Chapter 3 defines the hyperspherical space and introduces different types of operators that are needed to perform any

kind of geometric analysis between elements in this space. We discuss different methods to project elements onto the hypersphere, given its intrinsic or extrinsic view. We then provide distance functions depending on the type of projections selected. This chapter also discusses how to represent distributions and energy functions on the hypersphere using generalized symmetric positive definite kernels. Next, Chapter 4 introduces the concept of learning invariant representations in terms of an alignment and diversity objective on the hypersphere and discusses the issues of each method using the definitions of element coupling. We discuss the alignment and diversity objectives in terms of an element-element matching, variance, orthogonality, and energy reduction. Each algorithm is broken down and analyzed in order to evaluate its strengths or weaknesses depending on the type of element coupling observed. We then demonstrate that there exists a unified energy model which can be used to recover most hyperspherical diversity objectives, and show how to better formulate the problem to avoid issues with finite batches of data. We then use theoretical results found in the hyperspherical space to motivate a stable algorithm in an unnormalized space. In Chapter 5, we define the methodology used to train and evaluate a self-supervised model to assess the quality of the features learned without labels. We train a set of experiments across different alignment and diversity objectives subject to different choices of distances functions. We alternate between different representations of the hyperspherical space in order to explore any benefits of the stereographic model. We find that stereographic models underperform compared to those that used closest point projections. It is observed that the choice of distance function only plays a minor role in how representations are learned and is dependent on the construction of the diversity objective itself. Moreover, we show that models which avoid excessive positive-positive coupling outperform methods that contain said coupling by a statistically significant margin. Finally, we show that it is possible to modify the unnormalized learning procedure and find that the reformulation is competitive with other strong contrastive methods. Finally, Chapter 6 concludes the thesis by summarizing the key theoretical contributions detailed in Chapter 4 that are supported by the empirical evidence presented in Chapter 5.

2

Background

The following chapter aims to serve as a reference for the mathematics and theory required to understand how neural networks learn distilled representations and features from images. The information provided in the section is not complete, but can be regarded as a sufficient resource for understanding subsequent work explored in the thesis.

Before delving into any formal definitions for how neural networks learn, some basic notation, axioms, definitions, and concepts are established. These components serve as the foundation for how we chose to define the learning problem. Following these definitions, we explore high level concepts from representation learning, introduce vision based neural networks and their architecture, and outline a generalized learning framework through the lens of empirical risk minimization using loss functions and gradient descent. The concept of self-supervised learning is introduced through a representation

learning framework and is explored with references to relevant related works.

2.1 Spaces, Distances, and Manifolds

The following section aims to provide the reader with a fundamental math background necessary to understand the content within this thesis. It is intended to introduce specific components used in various segments related to deep learning. We define all components in terms of vector valued functions. We introduce different spaces and their properties, however, in the interest of being concise, we aim to only introduce content which is relevant throughout the remainder of the thesis.

2.1.1 Metric Spaces

In order to be able to compare elements to each other, we must first define a space and function which systemically defines where the elements exist and a metric to evaluate their similarity or distance [12].

A **metric space** is defined by an ordered pair (M, d_M) where M is a non-empty set of elements and d_M is a distance function called a metric. The metric is defined on M such that $d_M : M \times M \to \mathbb{R}_+$ satisfying:

$\forall \mathbf{x}, \mathbf{y}, \mathbf{z} \in M$

1. $d_M(\mathbf{x}, \mathbf{y}) = 0 \iff \mathbf{x} = \mathbf{y}$ (*separation of points*)
2. $d_M(\mathbf{x}, \mathbf{y}) = d_M(\mathbf{y}, \mathbf{x})$ (*symmetry*)
3. $d_M(\mathbf{x}, \mathbf{z}) + d_M(\mathbf{z}, \mathbf{y}) \geq d_M(\mathbf{x}, \mathbf{y})$ (*triangle inequality*)

Given two metric spaces (M, d_M) and $(M', d_{M'})$, a function $f : M \to M'$ is said to be an **isometry** if $\forall \mathbf{x}, \mathbf{y} \in M$:

$$d_M(\mathbf{x}, \mathbf{y}) = d_{M'}(f(\mathbf{x}), f(\mathbf{y})). \tag{2.1}$$

Isometries are the basis for constructing learning problems where spaces have clear correspondences. These correspondences form an isomorphism that is distance preserving,

and can be used to evaluate certain properties in spaces that are more conducive to analysis. By definition, an isomorphism is bijective.

2.1.2 Sets and Balls

We define elements and bounds as a function of sets and balls. These components are of utmost importance when describing problems over spaces with particular properties.

An **open ball** $B_R^n(\mathbf{c})$ of dimension n with radius R centered at a point $\mathbf{c}$ in a metric space (M, d_M) is the set $\{\mathbf{x} : d_M(\mathbf{c}, \mathbf{x}) < R\}$. A **closed ball** has the form of $\{\mathbf{x} : d_M(\mathbf{c}, \mathbf{x}) \leq R\}$ [12]. An **open set** U in (M, d_M) is a set where for every element in U, there exists an open ball centered about the element which contains all other elements in U. A **closed set** is the complement of an open set. A **bounded set** is a set contained in a ball with finite radius [12].

2.1.3 Topological Spaces

Not all geometric structures fall under the definition of a proper vector space. Tools from topology allow us to understand and describe nuances related to elements on these structures. To define a topological space, we require a set of points and their open sets. The pairing allows us to analyze elements in terms of their proximity as a function of the open sets. This relationship is called the topology on the space and is a flexible generalization of metric spaces.

A **topological space** is an ordered pair $(X, \mathcal{O})$ where X is a non-empty set with a collection of open subsets of $\mathcal{O}$ specified on X satisfying:

Given all collection of open sets $I \in \mathcal{O}$

1. $X, \emptyset \in \mathcal{O}$ (*contains the empty set and X itself*)
2. $\{A_i : i \in I, A_i \in \mathcal{O}\} \Rightarrow \cup_i A_i \in \mathcal{O}$ (*closure under arbitrary unions*)
3. $A, B \in \mathcal{O} \Rightarrow A \cap B \in \mathcal{O}$ (*closure under finite intersections*)

If we wish to describe a space with explicit structure or geometry, we must be able to decompose it into minimal intersections of open sets that are analogous to patches that cover the space. The covering allows us to subdivide analysis of the entire structure using the open sets that define properties in a neighbourhood. An **open cover** of a topological space $(X, \mathcal{O})$ is a group of open sets U which cover $X = \cup_{U_i \in U} U_i$. A **subcover** of U is a subgroup that also covers X. A topological space is **compact** if every open cover has a finite subcover [13].

2.1.4 Inner Products

Inner products allow us to define certain angular properties between vectors. These angular properties can be used to describe pairs of elements in terms of their similarity.

Let $\mathcal{A}$ be a vector space defined over the reals $\mathbb{R}$. An **inner product** on $\mathcal{A}$ is a function $\langle \cdot, \cdot \rangle_{\mathcal{H}} : \mathcal{A} \times \mathcal{A} \to \mathbf{R}_+$ satisfying:

$\forall \mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathcal{A}, a, b \in \mathbb{R}$

1. $\langle a\mathbf{x} + b\mathbf{y}, \mathbf{z} \rangle_{\mathcal{A}} = a\langle \mathbf{x}, \mathbf{z} \rangle_{\mathcal{A}} + b\langle \mathbf{y}, \mathbf{z} \rangle_{\mathcal{A}}$ (*linearity*)
2. $\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{A}} = \langle \mathbf{y}, \mathbf{x} \rangle_{\mathcal{A}}$ (*symmetry*)
3. $\langle \mathbf{x}, \mathbf{x} \rangle_{\mathcal{A}} \geq 0$, $\langle \mathbf{x}, \mathbf{x} \rangle_{\mathcal{A}} = 0 \iff \mathbf{x} = 0$ (*positive definiteness*)

A **Hilbert space** $\mathcal{H}$ is an example of a vector space equipped with an inner product that induces a metric $\forall \mathbf{x}, \mathbf{y} \in \mathcal{H}$, $d_{\mathcal{H}}(\mathbf{x}, \mathbf{y}) = \sqrt{\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{H}}}$ [12].

2.1.5 Norms and the Induced Metric

A **norm** is a real valued function which maps elements from a vector space M to the non-negative scalars as $|| \cdot || : M \to \mathbb{R}_+$. The norm is also induced by an inner product over M denoted as $|| \cdot ||_M = \sqrt{\langle \cdot, \cdot \rangle_M}$ and thus shares most properties.

Given $\mathbf{x}, \mathbf{y} \in M$, $a \in \mathbb{R}$:

1. $||a \cdot \mathbf{x}||_M = |a| \cdot ||\mathbf{x}||_M$ (*positive homogeneity*)

2. $||\mathbf{x}||_M \geq 0, ||\mathbf{x}||_M = 0 \iff \mathbf{x} = \mathbf{0}$ (*positive definiteness*)

3. $||\mathbf{x}||_M + ||\mathbf{y}||_M \geq ||\mathbf{x} + \mathbf{y}||_M$ (*triangle inequality*)

Using the norm, we can easily construct a metric for $\mathbf{x}, \mathbf{y} \in M$ as $d_M(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}||$. More generally, we may also define the norm in relation to the space. In the case of finite vectors of dimension n, $\mathbf{x} \in M$, where $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, we define the p-norm $|| \cdot ||_p$:

$$||\mathbf{x}||_p = \Big(\sum_{i=1}^{n} |x_i|^p \Big)^{\frac{1}{p}}. \tag{2.2}$$

By definition, the p-norm of a vector $||\mathbf{x}||_p$ can be viewed as a distance between $\mathbf{x}$ and the zero vector $d_p(\mathbf{x}, \mathbf{0}) = ||\mathbf{x} - \mathbf{0}||_p = ||\mathbf{x}||_p$. Another interesting property is that the metric induced by the norm is rotation and translation invariant. Here, it is observed that setting $p = 2$ recovers the Euclidean norm and setting $p = \infty$ recovers the maximum.

2.1.6 Kernels and Similarity Functions

When dealing with high dimensional vectors, it is often convenient to work with measures of similarity rather than distances, since similarities are typically bounded between $[-1, 1]$. At times, it may not be possible to have access to an inner product. Kernel methods allow us to circumvent such issues by comparing hidden feature maps and also allows us to compute smooth functions between elements in the same set. Kernels may also be used to perform interpolation or to quantify quantities related to energy or similarities between features. A **kernel** is a function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_+$ defined over the non-empty set $\mathcal{X}$ given the existence of a real Hilbert space $\mathcal{H}$ and a feature map $\phi : \mathcal{X} \to \mathcal{H}$. The kernel is defined as [14]:

$$K(\mathbf{x}, \mathbf{y}) = \langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle_{\mathcal{H}}, \qquad \forall \mathbf{x}, \mathbf{y} \in \mathcal{X}. \tag{2.3}$$

We do not require an inner product on $\mathcal{X}$ and can skip this operation by utilizing the features associated to its elements. We informally remark that sums and products of kernels are also kernels as long as any mixing coefficients guarantee positive definiteness. In general a symmetric function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is **positive definite** if $\forall n \geq 1, (a_1, a_2, \ldots, a_n) \in$

$\mathbb{R}^n, (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n) \in \mathcal{X}^n$ [14]:

$$\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j K(\mathbf{x}_i, \mathbf{x}_j) \geq 0. \tag{2.4}$$

The result also holds for all kernels K, therefore, all kernels are positive definite. Kernels are also used to define operations $\mathcal{X}$ using functions $f \in \mathcal{H}$ [14]. A special kind of kernel exists and is defined by its reproducing property. A **Reproducing Kernel Hilbert Space** (RKHS) is a Hilbert space with the reproducing property. A Hilbert space is said to be reproducing if it has the **reproducing property** defined $\forall \mathbf{x} \in \mathcal{X}$, $\phi(\mathbf{x}) = K(\cdot, \mathbf{x})$ and $\forall f \in \mathcal{H}$, $\langle f, K(\cdot, \mathbf{x})\rangle_{\mathcal{H}} = f(\mathbf{x})$. By extension:

$$K(\mathbf{x}, \mathbf{y}) = \langle K(\cdot, \mathbf{x}), K(\cdot, \mathbf{y})\rangle_{\mathcal{H}}, \qquad \forall \mathbf{x}, \mathbf{y} \in \mathcal{X}. \tag{2.5}$$

For the remainder of this thesis, we shall refer to kernels as positive definite functions as per equation 2.4. As an additional reference, it is clear that kernels can be used to assign a pseudo-metric $\forall \mathbf{x}, \mathbf{y} \in \mathcal{X}$ using the reproducing property for bounded kernels given by:

$$\begin{aligned} d_K(\mathbf{x}, \mathbf{y}) &= ||K(\cdot, \mathbf{x}) - K(\cdot, \mathbf{y})||^2_{\mathcal{H}} \\ &= 2 - 2K(\mathbf{x}, \mathbf{y}). \end{aligned} \tag{2.6}$$

We also restrict ourselves to kernels that are invariant to translations. A kernel that is translation invariant satisfies the property $\forall \mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathcal{X}$:

$$K(\mathbf{x} + \mathbf{z}, \mathbf{y} + \mathbf{z}) = K(\mathbf{x}, \mathbf{y}). \tag{2.7}$$

There exists a family of invariant kernels called **universal kernels** which we define over metric space $(\mathcal{X}, d_{\mathcal{X}})$. They are described $\forall \mathbf{x}, \mathbf{y} \in \mathcal{X}$ and $\alpha, \beta \in \mathbb{R}_+$ as [15]:

$$K_\alpha(\mathbf{x}, \mathbf{y}) = \exp(-\alpha d_{\mathcal{X}}(\mathbf{x}, \mathbf{y})^2). \tag{2.8}$$

$$K_{\alpha,\beta}(\mathbf{x}, \mathbf{y}) = (\beta + d_{\mathcal{X}}(\mathbf{x}, \mathbf{y})^2)^{-\alpha}. \tag{2.9}$$

We call equation 2.8 the **Gaussian kernel** and equation 2.9 the **inverse polynomial kernel**. Although it is not universal, we note that for $\alpha < 0$ we recover the **polynomial kernel** and for unrestricted α and $\beta = 0$ we recover the **Riesz kernel**. We also note that using the ℓ_1 distance rather than the squared ℓ_2 distance in equation 2.8 recovers the **Laplacian kernel**. We may use the reproducing property to first embed into a space with certain properties, and then leverage a kernel to evaluate non-linear similarities between elements in the embedded space.

2.1.7 Manifolds and Geodesics

In order to evaluate any kind of learning procedures in non-Euclidean space, it is imperative to define the properties of the geometry in the space, as well as a metric that quantifies distances between points. Not all spaces may be proper metric spaces and as a result, we are required to subdivide the space in components which can be analyzed in pieces and reassembled as needed. Tools from topology allow us to define and understand these spaces in terms of the open sets that cover it. The geometry is called a manifold, and it is defined as follows. A **smooth paracompact manifold** $\mathcal{M}$ is a topological space that satisfies the following [13]:

1. All pairs in $\mathcal{M}$ have at least one pair of disjoint neighbourhoods (*Hausdorff*)
2. $\mathcal{M}$ has a countable basis for its topology (*second countable*)
3. $\mathcal{M}$ is equipped with a set of charts $\{(U_i, \varphi_i)\}$ where $\varphi_i : U_i \to \mathbb{R}^n$ (*maximal atlas*)

The set of charts is called an atlas. The atlas contains the collection of open sets U_i that form a complete cover of $\mathcal{M}$. Each chart in the atlas is composed of a pair (U_i, φ_i) where U_i is an open set on $\mathcal{M}$, and φ_i is an isomorphism that preserves certain topological properties from the open set on $\mathcal{M}$ to an open set in $\mathbb{R}^n$. The open set and the map is a chart that describes the structure in a neighbourhood of the manifold and the mapping φ_i is the push-forward operation onto $\mathbb{R}^n$ which allows us to understand how elements are compared to one another in a more sensible space. Since a manifold is defined using a collection of patches (open sets), the metric on $\mathcal{M}$ that defines the total distance between

two elements $\mathbf{x}, \mathbf{y} \in \mathcal{M}$ is a function of the path $p : [\mathbf{x}, \mathbf{y}]$ taken and the patches traversed. A path is parameterized by a curve $\gamma(t)$ where $\gamma(0) = \mathbf{x}, \gamma(1) = \mathbf{y}$. Each patch may have non-constant curvature that uniquely deforms space and distances in its neighbourhood. The distortion is accounted for by defining a local metric $g_{\gamma(t)}$ tangent to the manifold denoted as $T_{\gamma(t)}\mathcal{M}$ for a position on the curve at t. We call this space and the local metric the **tangent space** and the Riemannian metric respectively. We let $\dot{\gamma}(t)$ be the instantaneous velocity of the curve at time t and define the total distance traversed along a path on the manifold using its arclength $L(\gamma)$. The resulting distance is therefore the total accumulation of distance segments along the path measured by the instantaneous velocity in the tangent space $T_{\gamma(t)}\mathcal{M}$ at all locations along the curve γ [13, 16]. Arclength or distance is therefore defined as:

$$L(\gamma) = \int_0^1 ||\dot{\gamma}(t)||dt. \tag{2.10}$$

Under the assumption that $\mathcal{M}$ is geodescially complete, meaning that there exists a geodesic for all elements on the manifold, the **geodesic** is defined as the shortest path in the set of all possible paths $\Omega(\gamma)$. The geodesic is therefore the generalized metric between the two elements on $\mathcal{M}$ as a function of the topology. We define the distance $d_{\mathcal{M}}$ as [13, 16]:

$$d_{\mathcal{M}}(\mathbf{x}, \mathbf{y}) = \min_{\gamma \in \Omega(\gamma)} L(\gamma). \tag{2.11}$$

2.2 General Representation Learning

Representation learning is the topic concerned with learning features or representations from data. This is accomplished by transforming or distilling information from higher dimensional data onto a lower dimensional manifold with distinct structure using a map. Features are described by these lower dimensional embeddings, which typically have some semantic meaning to them. Features can be learned from a variety of tasks in an attempt to influence the structure of the space. We define a finite dataset X with data distribution p_{data} where $\mathbf{x} \in \mathbb{R}^n$ and embeddings $\mathbf{z} \in Z$ where given $\mathbf{z} \in \mathbb{R}^m$ and $n \geq m$. We define a map $F : X \to Z$ which, as stated, maps samples in X into the embedded

space Z. F can be constructed for specific task using linear or non-linear maps. The map may also be invertible, meaning that there exists a map $F^{-1} : Z \to X$. A map may

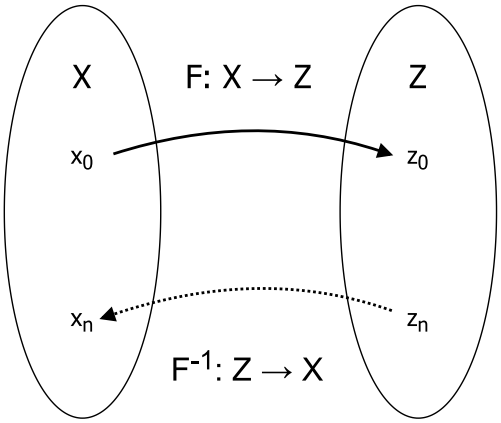

Figure 2.1: Data transformations via maps

be learned as a function that aims to provide a solution to specific kinds of problems. Consider the problem of classifying a set of data points X with a known set of finite k categorical labels $y \in \{1, 2, \ldots, k\}$ belonging to a label space Y. If an error minimizing map exists, it can be expressed as a chain of non-invertible many-to-one maps from the data space to embedding space and then again to label space $F^* : X \to Y$, where Z is an intermediate representation space that is passed through. We define a chain to explicitly decouple the label space from the embedding space, which is also called a latent space. This is done for consistency reasons, as embedding space have interesting properties to be explored in subsequent sections. An example of a linear map used for classification is the method of eigenfaces. This method learns a basis for the set of human faces that is used to classify faces seen in the dataset [17]. The map is learned using Principal Component Analysis on the covariance matrix constructed from observed images of faces in order to find a map from $\mathbb{R}^n$ to $\mathbb{R}^m$ given $n > m$. The map is learned by specifying and optimizing the task that requires minimized variance within the projected space.

While a linear map may be useful in certain situations, they are not always expressive enough for more complex problems such as image classification, regression, and segmentation, whose solutions have highly non-linear relationships. As a result, it is imperative to define requirements and frameworks on how to construct non-linear maps that learn

meaningful features in accordance to a target problem. There are many ways in which this problem can be tackled, however, the chosen method of interest is centered around the optimization of neural networks. We note that neural networks are called of universal function approximators since they are able to model arbitrarily complex functions.

2.2.1 Neural Networks, Loss Functions, and Optimization

A **neural network** is a universal function approximator $f_\theta : X \to Z$ parameterized by a diverse set of weights and biases denoted as θ. This function approximator is a map that is used to extract features or values Z from data X. In the case of computer vision, the neural networks of choice are the multi-layer perception (MLP), the convolutional neural network (CNN), and the vision transformer. CNNs are built from convolutional operators that act as translational equivalent localized linear transformations. Convolutions leverage a natural inductive bias that neighbourhoods of pixels in an image are naturally correlated and connected with one another. They are also composed of non-linear activations and positionally invariant subsampling functions. These networks build up complex representations as a function of network depth [1]. Their structure is summarized as a depth-wise stack of blocks that are each composed of convolutions, non-linearities, and subsampling operation. The ordering of the components in a block formation is motivated by the laws of linearity. Consider a set of linear matrices T. By the laws of linearity, there exists a single linear transform that is equivalent to the composition of multiple linear operations $t^* = \prod_{t \in T} t$. This can also be shown by matrix decomposition techniques like Singular Value Decomposition, where each matrix is decomposed into a product of individual scale and rotation matrices. Chaining multiple linear transformations together can therefore be summarized by a single net rotation and scale factor. Consequently, a non-linear function must be placed between linear operators in order to build up the expressiveness of a network, since expressiveness can be increased proportionally to the number of non-linear elements composed on top of one another.

It is convenient to define learning objectives using a pair of neural networks called the backbone and projector. Let $f_\theta : X \to Z$ be the backbone mapping from data X to em-

bedding space Z parameterized by θ and let $g_\phi : Z \to Y$ be the projector mapping from embedding space to label space Y parameterized by ϕ. The backbone is the map that produces features of reduced dimensionality, while the projector is a map which uses the backbone features to solve a particular task whose error can be measured and minimized. For example, a CNN may serve as a backbone and a MLP may serve as a projector. The composition of the two networks is $h_{\theta,\phi} = g_\phi \circ f_\theta$.

Given a dataset X, a neural network f_θ may produce useful embeddings $\mathbf{z} = f_\theta(\mathbf{x})$ and solve a specified task with labels Y if the task is well posed. In most scenarios, datasets and neural networks are large. Due to the size of these elements, it is often not computationally tractable to learn optimal parameters of a network for a particular task if it requires knowledge of all samples in the dataset at the same time. A loss function can be used as a surrogate task to empirically estimate the performance of the network pair on a mini-batch of data. A mini-batch is a subset of the data and is sampled randomly without replacement under the assumption that all elements in X are independent and identically distributed. Under this assumption, minimizing the empirical loss on a mini-batch over the set of many mini-batches may sufficiently minimize the error on the original task. We let an approximation of the true label $\hat{y}$ be produced by mapping a sample $\mathbf{x}$ from the joined data and label distribution $(\mathbf{x}, \mathbf{y}) \sim p_{\text{data}}$ into the target space with a neural network pair such that $\hat{y} = g_\phi(f_\theta(\mathbf{x})) = h_{\theta,\phi}(\mathbf{x})$. We measure the performance of the system from the error between the approximation and the associated ground truth $\mathbf{y}$ through the loss function $\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y})$. The learning objective aims to find optimal parameters θ^*, ϕ^* that solve the task by minimizing the associated empirical loss estimated over the mini-batches. This is posed as:

$$\theta^*, \phi^* = \underset{\theta,\phi}{\operatorname{argmin}} \underset{(\mathbf{x},\mathbf{y})\sim p_{\text{data}}}{\mathbf{E}} [\mathcal{L}(h_{\theta,\phi}(\mathbf{x}), \mathbf{y})]. \tag{2.12}$$

It should be noted that any deep learning loss function can use ground truth labels which are represented as a vector, regardless of whether the task is classifying a discrete label or regressing a higher dimensional embedding. This is due to the fact that any quantity may be encoded in a one-hot format.

There are two natural ways in which a loss function may be constructed and optimized.

It is assumed that the loss function is piecewise differentiable everywhere, in order to get feedback through the estimation of gradients. The first method relies on encoding network outputs as probabilities for a given task. Given a set of task probabilities and a target, we may model encoding networks as statistical maps that approximate an empirical label distribution conditioned on the input $p(\mathbf{y} \mid \mathbf{x})$. These maps may be optimized with any maximum likelihood estimation (MLE) or maximum a posteriori procedures. The second method constructs loss functions given a differentiable metric over the target space. In this scenario, the aim is to minimize or maximize a distance based function subject to the geometry of the space. Distance functions may be coded as empirical probabilities as well, using statistical kernels or Boltzmann distributions. All neural networks can be optimized with the backpropagation algorithm and stochastic gradient descent (SGD) under the latter conditions [1]. Let $\theta = \{W_i\}_N$ be the set of all weights in a neural network. Using chain rule, we may express the gradients for a given weight layer as a function of the gradients produced by a loss function. For any weight layer, the simple update equation at step t with step size α can be described by:

$$W_i^{(t+1)} = W_i^{(t)} - \alpha \nabla_{W_i} \mathcal{L}(\hat{\mathbf{y}}, \mathbf{y}). \tag{2.13}$$

The value of a loss function is an important metric to track the learning process over time and is a good indicator on downstream task performance. The gradient and update equations are also useful when looking to understand the learning dynamics, since the loss function dictates how the network parameters are eventually learned.

2.3 Self-Supervised Learning

The goal of self-supervised learning is to learn meaningful representations from data without any labels. These representations can be used on a variety of downstream tasks such as image classification, object detection, mass-language modeling, and multi-modal understanding. Self-supervised learning is a subset of representation learning where information about the data can be mined by specifying an objective function that is implicit

to the data itself. The objective is called a pretext task, which provides pseudo-labels to a learning algorithm in order to provide feedback in the learning process. The features learned are a function of the information required to perform the pretext task, thus, it is imperative that meaningful pretext tasks are selected given a known downstream task. If features can be learned without labels, it is then possible to finetune such features with a small subset of labels in order to solve the task associated to those labels.

In the context of feature learning in computer vision, the most prevalent pretext task of interest is the task of image representation invariance. A network f_θ is said to be **invariant** with respect to a set of operations $\mathcal{T}$ acting on a set of images X if and only if the set of output embeddings measured by a metric d_Z is unchanged under the operation. In practice, it is not always possible to learn perfectly invariant embeddings, therefore, we say a network is invariant subject to a tolerance threshold ϵ if:

$$\sup_{\mathbf{x} \in X, t \in \mathcal{T}} d_Z(f_\theta(t(\mathbf{x})), f_\theta(\mathbf{x})) < \epsilon. \tag{2.14}$$

Image representation invariance can be learned with respect to a set of perturbations called augmentations or transforms $\mathcal{T}$ with transformation distribution p_{aug}. We would like to produce invariant image embeddings subject to a set of injected noise operations. We chose a set of noise operations that should consistently preserve the majority of the information content in the image. These operations should induce noticeable changes in the image and can be lossy. Examples of augmentations include random crops, blurs, color distortions, and affine transformations and can be seen in Figure 2.2. Random crops allow features to distill knowledge from subsets of the information present in an image, blur promotes texture invariance, distortions encourage color invariance, and affine transformations allow for positional and pose invariance. An image sample $\mathbf{x} \sim p_{\mathsf{data}}$, is perturbed by an augmentation $t \sim p_{\mathsf{aug}}$ that produces a **view** of the sample $t(\mathbf{x})$.

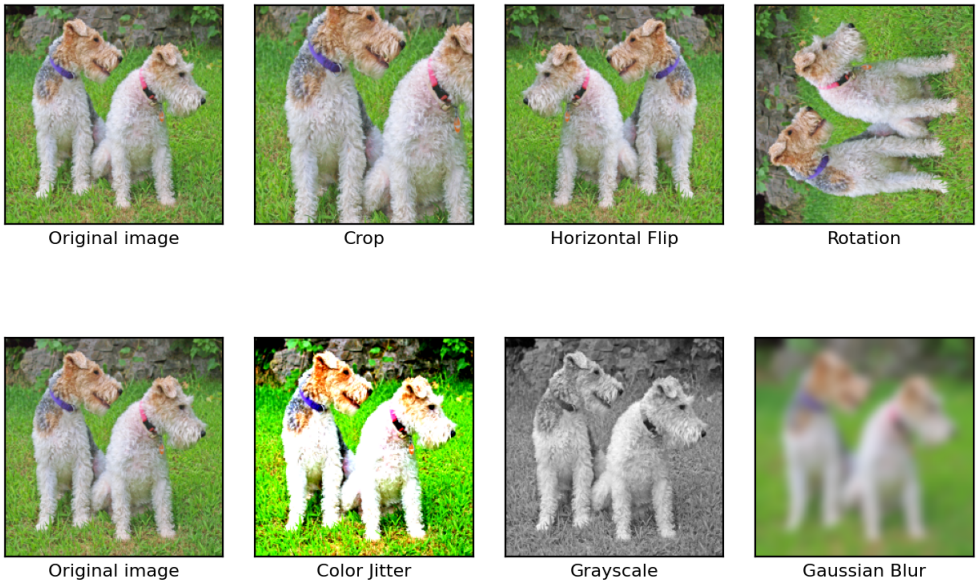

Figure 2.2: Example of invariance augmentations which include the crop, flip, rotation, color jitter, grayscale, and blur operations.

The invariance objective can be defined using distance functions in the embedding space. Given a dataset X, transformations $\mathcal{T}$, backbone and projector pair $h_{\theta,\phi} : X \to Z$, and metric $d_Z : Z \times Z \to \mathbb{R}_+$ the invariance objective is described as:

$$\theta^*, \phi^* = \underset{\theta,\phi}{\operatorname{argmin}} \underset{\mathbf{x}\sim p_{\text{data}}}{\mathbf{E}} [\underset{(t,t')\sim p_{\text{aug}}}{\mathbf{E}} [d_Z(h_{\theta,\phi}(t(\mathbf{x})), h_{\theta,\phi}(t'(\mathbf{x})))]]. \tag{2.15}$$

Here, the pairwise distance of two different views is minimized over all samples in the dataset. In the limit of the optimization, minimizing the pairwise differences is equivalent to minimizing the variance over the set of all transformations. This optimization is inherently flawed, as any variance minimization problem can be solved with a trivial solution where the network output is a constant vector $\mathbf{c}$ regardless of input ($h_{\theta,\phi}(\mathbf{x}) = \mathbf{c} \quad \forall \mathbf{x} \in X, \mathbf{c} \in \mathbb{R}^n$). If the network $h_{\theta,\phi}$ maps all representations to a constant vector, the task is solved, however no useful information has been learned. To circumvent this issue, a regularizer is added to ensure the embedding space cannot collapse or stabilize to suboptimal solution spaces. This is accomplished through the use of a diversity penalty that places a lower bound on the variance between features produced. For simplicity, we defined the modified problem statement in terms of a variance-invariance objective. Let

$\lambda > 0$ be a regularization weighting term. The reformulation of the invariance objective subject to a variance penalty $\mathbf{V}$ is:

$$\theta^*, \phi^* = \underset{\theta,\phi}{\operatorname{argmin}} \underset{\mathbf{x}\sim p(\mathbf{x})}{\mathbf{E}} [\underset{(t,t')\sim p(t)}{\mathbf{E}} [d_Z(h_{\theta,\phi}(t(\mathbf{x})), h_{\theta,\phi}(t'(\mathbf{x})))]] - \lambda \mathbf{V}[h_{\theta,\phi}(X)]. \tag{2.16}$$

It is noted that any alternative penalty which encourages a diverse set of outputs is a sufficient regularizer as well. These alternatives shall be explored in subsequent sections as a function of the geometry in the embedding space. If the variance over the set of all embeddings is non-zero, it is assumed that in order to minimize the main objective, semantic information must be learned to ensure consistency between distorted images. The reasoning behind this phenomenon is that the network is encouraged to learn a hidden latent understanding of an image that is independent to fluctuations in noise. These factors can only be rooted in distinct content based signatures within an image.

The learning framework must pay particular attention to the metric used in the optimization framework. If the embedding space is a metric space (Z, d_Z), it must be bounded, otherwise the weights of the network may collapse during training. As stated in the section 2.1.2, a metric space is bounded if the elements of Z are bounded set. In order to maximize diversity in a stable manner, an upper bound must exist. This may be achieved through the use of a bounded metric on a normalized space or a learning penalty which artificially places an upper bound on Z.

Most modern invariance based self-supervision algorithms rely on the siamese network [6, 11, 18, 19]. The siamese network is typically comprised of a backbone and projector pair (f_θ, g_ϕ). In a Siamese learning environment, a single batch of data is sampled, and two views are generated via augmentations. The two noisy batches are passed into the same network with shared weights separately in order to preserve batch statistics. The outputs are then compared across embedded views for the invariance objective, while diversity regularization can be explicitly computed as needed.

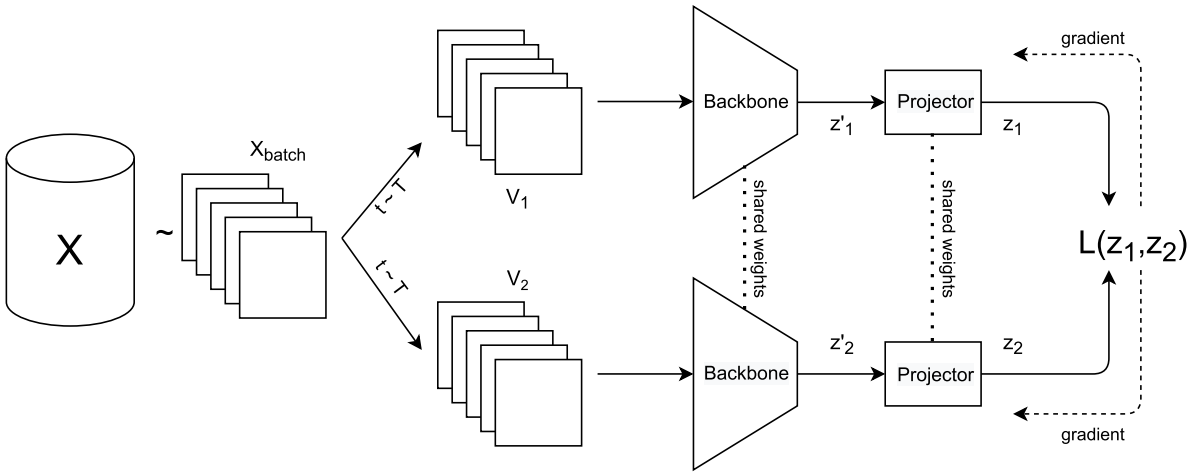

Figure 2.3: Generic siamese invariant learning framework. Images $\mathbf{v}$ are sampled from a dataset and perturbed using a bag of augmentations to generate views $\mathbf{v}$. The views are processed by a backbone-projector pair and compared using a loss function.

Once features have been learned, they must be evaluated on downstream tasks. A common strategy for evaluating the strength of the learned features is though a linear evaluation protocol [6, 19]. The projector is tossed away and backbone is frozen. A new single layer linear classifier is added to the output of the backbone. This projector is then trained on labelled data in order to verify the linear separability of the learned embedding space. An additional means of evaluating the quality of learned features is by slowly finetuning the backbone with a new linear classifier on a subset of the labelled data (1% or 10%) [6, 19]. The linear evaluation protocol is the evaluation method of choice for the work presented.

As an extension to the terminology, the set of all augmented views generated from the same source sample are called positive samples. On the contrary, the set of differing samples in a batch or across batches are called negative samples. We denote the distribution of positive samples as p_{pos} and the distribution of negatives as p_{neg}. The invariance task is defined over the set of positives and the regularization task is imposed over the set of negatives.

2.3.1 Single Instance Learning

While the focus of the latter section is on invariance objectives using siamese networks, there exists rich literature on methods involving a single network and a single view of an image at a time. Many of these methods have been critical in laying the groundwork for understanding the composition of contrastive methods, which is the main focus of this thesis.

Prior to the concept of a pretext task, pretraining a neural network was accomplished using autoencoders. The autoencoder is composed of an encoding and decoding networks that embed samples from a dataset into a latent space and then decodes them back into image space [20]. The goal of an autoencoder is to learn a lossless compressed representation that removes redundant information within an image while preserving all the semantically relevant content of the image itself. In practice, autoencoders are lossy since they cannot encode all image content into arbitrarily small latent representations. While the autoencoder falls under the realm of unsupervised learning, it can be viewed as a self-supervised algorithm where the pretext task is a reconstruction objective. Variants of the autoencoder extend it's utility by including denoising, variational, and masking processes. The denoising autoencoder aims to improve the process of encoding relevant information by learning features invariant to noise and is an early example of task invariance [21]. The variational autoencoder added an information bottleneck between the encoder and the decoder. This bottleneck is an attempt to capture the most important aspects of the data in the latent representation by leveraging the variational lower bound on the log-likelihood of the data [22]. The masked autoencoder used a masking procedure to remove random patches in an image and used the reconstruction protocol to learn the missing content [23]. This method has seen the most success in recent years and is competitive with many invariance based methods. Moreover, there exists extensive literature on disentangled autoencoders, where disentanglement is defined by the neural network's ability to isolate content and style within a representation. This isolation is with the goal of learning general and transferable features to a diverse set of downstream tasks that are directly in line with the goals of all self-supervised algorithms [24, 25, 26].

Foundational self-supervised algorithms are rooted in single objective pretext tasks that are categorized through the use of color or positional transformations. Color based methods originated with the goal of restoring color to old photos. These methods learn to regress color histograms from grayscale images. The process is simulated by applying the grayscale transformation to a colored image and using its input color distribution as the supervisory signal for training [2]. Colorization as a pretext task allows a neural network to implicitly learn structural and color based relationships, since there are known correspondences between objects and their colors. Structural methods focus on developing pretext tasks as a function of local and global content. A successful method that incorporated global content is the method of quantized rotations [3]. An image is rotated by factors of $90°$ and a neural network is tasked with learning the angle that the original image is rotated by. This method learns reasonable representations that are covariant to the pretext task due to the inherent relationships between the pose of certain objects and its orientation. Methods that incorporate local structural knowledge use image patches that are generated using cropping protocols. Pretext tasks are formulated around the goal of learning spatial correlations between each of the generated image patches. The relative position of each patch is regressed by neural network by solving jigsaw puzzles or by predicting their context [4, 5].

2.3.2 Contrastive Learning

Contrastive learning in the context of deep learning is the study of learning representations by contrasting pairs of samples against one another. Contrastive learning has been formulated over a variety of learning objects and can be applied in both supervised and self-supervised settings. In a contrastive learning framework, we define two categories of sample pairs. We say that positive pairs are samples that should be similar to each other, and negative pairs are those that should be dissimilar to each other. Similarity can be defined in a number of ways, however, we quantify similarity in terms of sample closeness measured by a metric.

One of the first methods to inspire learning invariant embedding is the Exemplar al-

gorithm [27]. Exemplar learns invariant embeddings by first preprocessing the entire dataset with multiple augmentations and then learning to classify each image index across the dataset. This method did not explicitly compare images across indices, however, a classification loss such as cross-entropy implicitly compares the logit of each sample. The method is problematic because it requires a classifier equal to the size of the dataset and is not flexible for larger, more realistic datasets. This method set the groundwork for more flexible methods which directly contrast samples with samples in a self-supervised setting.

Early contrastive methods were formulated for supervised classification type problems where class labels are used as a means of aggregating sets of positives and negatives. The contrastive loss distinguishes between samples of different classes using a hinge loss, where the distance between two samples from the same class is minimized and the distance between two different classes are maximized up to a cutoff threshold [28]. Thresholds are required because stable optimization procedures require all the embedded samples to form a bounded set. The hinge loss guarantees the set is bounded. The contrastive loss was later extended to the triplet loss [29]. The triplet loss is a hinge loss that maximizes inter-class separation between a positive pair and a random negative pair in an attempt to balance the push and pull between different classes. These methods are made more flexible by incorporating information between multiple negative pairs to learn better separation between multiple classes using the N-pair loss [30]. Statistical tools like noise contrastive estimation were developed and used to create entropy based loss functions that are able to distinguish between positive pairs and a noise distribution that is modelled using large sets of negative pairs. The method inspired by mutual information is called information noise contrastive estimation (InfoNCE) [31, 32].

Contrastive methods have direct implications for self-supervised learning, as it is possible to model noisy estimates of negative pairs by independently sampling from the data distribution. Positive pairs are simulated without having access to the underlying class label distribution by inducing a positive pair through a distortion process. The pair is then contrasted against samples drawn uniformly from the data distribution. The con-

trastive process aims to learn structural similarities between an image and itself. This is done by maximizing the likelihood it is most similar to itself compared to the noisy negative pairs, regardless of the applied distortions. The method of Simple Contrastive Learning Representations (SimCLR) uses this latter formulation of the InfoNCE loss and defines similarities between samples using normalized and unnormalized inner products [6]. The process is called noisy because the sampling procedure does not have access to the class label distribution and is thus a noisy mixture from the dataset. As a result, the distribution of negative samples may contain elements that are similar to a positive sample. It is noted that the noisy estimates can be smoothed out by sampling large sets of negative samples. The likelihood of drawing biased samples that share similar semantic information to that of the positive approaches the true distribution statistics as the number of samples grow. SimCLR demonstrates that normalization plays a key role in learning strong features, which has been adopted in all subsequent contrastive algorithms. Alignment and uniformity plays on the idea that normalization is necessary and models samples as elements on the unit sphere. It shows that a similar contrastive algorithm is achieved by minimizing the logarithm of the average pairwise potential between samples [11]. Memory bottlenecks have played a major role in limiting the total number of online negative samples that can be processed at a given time. To circumvent this issue, Momentum Contrastive Learning (MoCo) uses an offline running first-in first-out queue filled with previously seen negative samples to minimize the variability from the sampled noise distribution and expand the number of comparable negatives [8]. The running queue allows contrastive algorithms to keep track of approximate running estimation of the entire data distribution and allows for more meaningful samples to be compared against one another. Nearest Neighbour Contrastive Learning Representations (NNCLR) makes minor modifications to the idea of the running queue and proposes that the most informative negative samples are the samples that lie close to the positive itself. NNCLR uses the queue to find informative nearest neighbours instead of blindly comparing a positive to the entire set of running negatives [9]. A decoupled version of SimCLR, called DCL analyzes the asymptotic properties of the latter contrastive algorithms. It shows that typical contrastive learning problems should not be modelled using likelihoods, since

they have an implicit coupling term which impedes learning the best separation of features [10]. DCL also shows that the method of alignment and uniformity benefits from modeling the system using pairwise potentials, since it inherently dodges any induced problematic coupling terms as a result of normalizing densities.

There also exists a subset of methods inspired by clustering. DeepCluster leverages an iterative unsupervised clustering algorithm to find naturally occurring clusters in the data using the K-means algorithm. Each sample is assigned a pseudo-label corresponding to the cluster assignment, which is used as a supervisory signal in a classification problem [33]. SwAV explicitly models the contrastive learning problem between positive pairs and a set of prototypes that approximate the data distribution. Pairwise similarity is learned by contrasting individual cluster assignments to the set of prototypes. Clusters are assigned using Sinkhorn divergences, where the entropy regularization term ensures that the prototypes and samples are well distributed over the unit sphere [34].

2.3.3 Non-Contrastive Learning

Non-contrastive self-supervised techniques aim to learn the same invariant representations as their sample-contrastive counterparts. The goal of non-contrastive methods is to avoid directly contrasting samples with one another in an attempt to limit the noise associated with pushing negative pairs apart that share the same semantic information that is not available due to the lack of any class labels. These methods use various tricks to avoid dimensionality collapse. Bootstrap Your Own Latent is a method that uses an asymmetric siamese architecture with the addition of a predictor to process two sample views, where an online network is updated normally using SGD and an offline network is only updated using a slow moving average of the online network's weights [35]. The asymmetry allows the network to avoid collapsing to trivial solutions while avoiding any unnecessary regularization. SimSiam defaults back to the original siamese architecture, but uses an additional predictor to induce bidirectional architectural asymmetries when comparing views to one another. The asymmetry ensures that batch statistics have sufficient variance by standardizing the data before projecting onto the sphere to compare

views [7]. The second paradigm of non-contrastive uses dimension contrastive operations rather than sample contrastive operations. Dimension contrastive methods solve the dual problem with less computational constraints. In this setting, methods like Barlow Twins compute cross correlation matrices between features rather than Gram matrices between samples. This method aims to learn independence and is shown to be robust to small batch sizes [18]. Finally, variance-invariance-covariance uses an explicit mixture of variance and covariance regularization without any normalization. A hinge style loss is used to place an upper bound on how much standard deviation can be regularized to ensure a lower bound is maintained throughout training [19]. The mixture of variance, invariance, and covariance aims to balance three core objectives to learn a diverse set of features in a variety of flexible tasks.

2.4 Summary

In this chapter, we have reviewed important mathematical tools that include metric and topological spaces, as well as inner products, norms, and kernels. These tools enable us to define complex geometric manifolds which can be evaluated over its open sets using its charts. We have also introduced how deep learning works and how a neural network can learn from a training signal. Furthermore, we introduced the concept of learning invariant image representations without labels using self-supervision. We discussed what kind of augmentations could be used and what role each one of them play in generalizing to different downstream tasks. In the following chapter, we narrow down the type of geometry we are interested in. We define the hypersphere and many of its properties. We introduce different metrics on the space as well as projection maps that can be integrated into a neural network. Finally, we define how to specify distributions and energy functions using kernels on the hypersphere.

3

Operations on the Hypersphere

The following chapter is concerned with definitions, operations, and properties on the hypersphere. These components set the groundwork for learning invariant representations using contrastive methods. It is presented to the reader as a primer for understanding how to learn representations on high dimensional spheres defined by its constant curvature. The chapter introduces the space as a manifold and defines its tangent space and geodesics. We define the hypersphere in terms of its chart and provide tools to map to and from the manifold and the bundle of tangent spaces. Since the end goal of this body of work is to analyze optimization problems in this space, we also define practical means of representing spaces and introduce different types of projection models to map from a neural network's output space to sphere. We provide details on which metrics can be defined as a function of the projection model and show how these maps modify the metric

on the sphere.

Using these standardized operations and metrics, we are able to define distributions on the sphere and show their analogue to the unimodal standard normal in Euclidean space. We then extend the definition of a distribution on the sphere by defining them in terms of any bounded symmetric kernel. We provide additional tools to model pairwise potentials between elements on the sphere. We demonstrate that these potentials also have a kernel analogue, which can used to summarize configurations of elements on the sphere.

When combining these properties and operation, we are then able to analyze the problem of learning representations in terms of a system of elements on the sphere with particular learning dynamics as a function of the chosen metric and projection. Finally, we introduce assumptions related to these dynamics under a gradient descent optimization framework.

3.1 Metrics and the Hypersphere

Consider the case where all elements in a set M exist on a closed and compact spherical manifold $\mathbb{S}^n_{R,\mathbf{c}}$ embedded in $\mathbb{R}^{n+1}$. This manifold is called an n-sphere or **hypersphere** of radius R and centering $\mathbf{c}$. The hypersphere is commonly described using the ℓ_2 Euclidean norm. For simplicity, we refer to the zero centered hypersphere $\mathbb{S}^n_{R,\mathbf{0}}$ as $\mathbb{S}^n_R$.

$$\mathbb{S}^n_{R,\mathbf{c}} = \{\mathbf{x} \in \mathbb{R}^{n+1} : ||\mathbf{x} - \mathbf{c}||_2 = R\}. \tag{3.1}$$

The hypersphere $\mathbb{S}^n_R$ is a closed manifold and is not a proper vector space, since there is no clear means of adding vectors on the hypersphere. Although it is defined using an ℓ_2 inner product, a proper metric must be imposed using the definition of the geodesic. Unlike Euclidean space, spherical space is defined by its constant positive curvature, where curvature is related to the radius through the inverse square proportion R^{-2}. It is known that any two points taking arbitrary straight paths on the hypersphere will always have two intersections due to the constant positive curvature. All straight paths along the hypersphere are called great circles and the geodesic between two points on the hypersphere is defined by the minimal arc long the great circle that forms an intersection with them as

per figure 3.1.

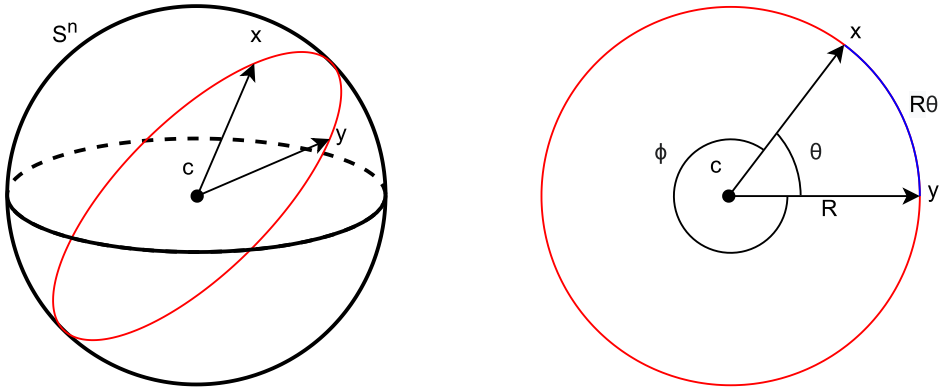

Figure 3.1: Intersection between two points embedded into $\mathbb{S}^3$ visualized as the great circle pictured in red. The minimum distance between the pair is the pictured in blue, defined by the smaller of the two angles $\theta < \phi$ on the great circle.

It is possible to solve for this distance analytically using a closed-form function that is bounded on $[0, R\pi]$. The resulting distance function is a modification built upon the Euclidean inner product. Given $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, the angular relationship given by the Euclidean inner product between vectors is:

$$\langle \mathbf{x}, \mathbf{y} \rangle_2 = ||\mathbf{x}||_2 ||\mathbf{y}||_2 \cos \theta_{\mathbf{x},\mathbf{y}}. \tag{3.2}$$

We condition on the hypersphere such that $\mathbf{x}, \mathbf{y} \in \mathbb{S}_R^n$ which implies $||\mathbf{x}||_2$ and $||\mathbf{y}||_2$ are both equal to R. Rearranging the angular relationship and taking the radial scale into account yields the **angular distance** $d_{\mathbb{S}_R^n}$:

$$d_{\mathbb{S}_R^n}(\mathbf{x}, \mathbf{y}) = R\theta_{\mathbf{x},\mathbf{y}} = R \cos^{-1}\left(\frac{\langle \mathbf{x}, \mathbf{y} \rangle_2}{R^2}\right). \tag{3.3}$$

A useful property related to the angular distance on the hypersphere is in relation to the asymptotic behavior with respect to its curvature. As the space becomes more flat and the curvature tends to zero from the right $R^{-2} \to 0^+$, distances between elements on the surface explode to infinity $d_{\mathbb{S}_R^n} \to \infty$. This property is useful since it is possible to select and thus control the dynamics, interactions, and energy of a system of elements on the surface by adding and removing density as a function of radius.

Another convenient measure is the cosine similarity $\cos \theta_{\mathbf{x},\mathbf{y}}$ recovered from the angular

relationship between vector pairs. Cosine similarity is not a distance, however, it is easily computed as the normalized ℓ_2 inner product between a set of observations and is bounded on $[-1, 1]$. Cosine similarity has an inherent relationship with the Euclidean metric and is used to construct metric through the hypersphere. We note that this is not a valid geodesic since it passes through points that are not on the manifold. We define $d_{\ell_2,R}$ as the **Euclidean distance** on the hypersphere by conditioning on its properties:

$$\begin{aligned} d_{\mathbb{S}_R^n}(\mathbf{x}, \mathbf{y}) &= d_{\ell_2,R}(\mathbf{x}, \mathbf{y}) \\ &= ||\mathbf{x} - \mathbf{y}||_2 \\ &= \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle_2 + \langle \mathbf{y}, \mathbf{y} \rangle_2 - 2\langle \mathbf{x}, \mathbf{y} \rangle_2} \\ &= R\sqrt{2 - 2\cos\theta_{\mathbf{x},\mathbf{y}}}. \end{aligned} \tag{3.4}$$

Both $d_{\mathbb{S}_R^n}$ and $d_{\ell_2,R}$ satisfy the triangle inequality and are positive definite. We omit the details in this section and note that $d_{\ell_2,R}$ has non-linear gradients as a result of it taking shortcuts through the hypersphere. The angular distance does not suffer from the same phenomenon. For more detail, refer to section 4.2.1.

It is imperative to define a framework where elements can be compared to one another. Let the tangent space at a reference point on the hypersphere $\mathbf{x} \in \mathbb{S}_R^n$ be denoted as $T_{\mathbf{x}}\mathbb{S}_R^n$. We say that the tangent space contains the set of all tangent vectors referenced at $\mathbf{x}$. Mapping onto the tangent space can be accomplished using an orthogonal projection at the reference, followed by a subtraction of the normal component. The **tangential projection** matrix at $\mathbf{x}$ is denoted as $P_{\mathbf{x}}$ where:

$$P_{\mathbf{x}} = I - \frac{\mathbf{x}\mathbf{x}^T}{||\mathbf{x}||_2^2}. \tag{3.5}$$

Note that for convenience, the outer product $\mathbf{x}\mathbf{x}^T$ is normalized. This formulation is the generalized projection matrix for all vectors, regardless of length or spherical constraint. It is clear that the tangent vector is orthogonal to its reference vector $\mathbf{x}$, therefore $P_{\mathbf{x}}\mathbf{x} = \mathbf{0}$.

A tangent vector $\mathbf{v} \in T_{\mathbf{x}}\mathbb{S}_R^n$ with a reference point $\mathbf{x}$ is said to have a map from the tangent space to the hypersphere, called the exponential map $\exp_{\mathbf{x}}^R : T_{\mathbf{x}}\mathbb{S}_R^n \to \mathbb{S}_R^n$. The inverse

analog is the logarithmic map $\log_{\mathbf{x}}^{R} : \mathbb{S}_{R}^{n} \to T_{\mathbf{x}}\mathbb{S}_{R}^{n}$ which maps a point on the hypersphere to the tangent space at $\mathbf{x}$. Given a tangent vector $\mathbf{v}$, the exponential and logarithmic maps on the hypersphere are defined as [13]:

$$\exp_{\mathbf{x}}^{R}(\mathbf{v}) = \cos\left(\frac{||\mathbf{v}||_2}{R}\right)\mathbf{x} + R\sin\left(\frac{||\mathbf{v}||_2}{R}\right)\frac{\mathbf{v}}{||\mathbf{v}||_2}. \tag{3.6}$$

$$\log_{\mathbf{x}}^{R}(\mathbf{y}) = d_{\mathbb{S}_{R}^{n}}(\mathbf{x},\mathbf{y})\frac{P_{\mathbf{x}}\mathbf{y}}{||P_{\mathbf{x}}\mathbf{y}||_2}. \tag{3.7}$$

We can observe these operations visually as per figure 3.2. These operations are required due to the presence of positive curvature. Consider an element moving along a trajectory. The trajectory can only be modelled in the tangent space and is only valid in a differential neighbourhood at the observed point. As the element moves along the tangent space in the direction of the tangent vector, the distance between it's updated position and the surface diverges as it is progressively pushed off the manifold. The exponential map allows us to circumvent such an issue by providing a map back from the updated position to the hypersphere at a new and correct position. The projection matrix, exponential map, and logarithmic map enable us to compute and analyze dynamics between elements co-existing on the hypersphere.

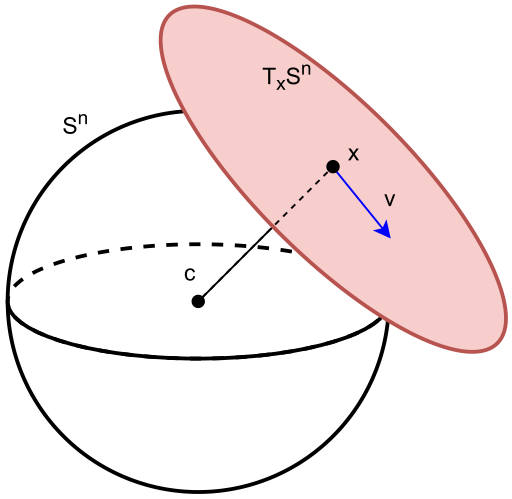

Figure 3.2: Tangent space depicted in red at point $\mathbf{x}$ with tangent vector $\mathbf{v}$ depicted in blue. Tangent vectors can be mapped from the hypersphere referenced at $\mathbf{x}$ using the logarithmic map and back using the exponential map.

One curious property of the hypersphere is in regard to its surface area as a function of

dimension. For the hypersphere $\mathbb{S}_R^n$ of dimension n and radius R, the surface area $S_{\mathbb{S}_R^n}$ is defined as [36]:

$$S_{\mathbb{S}_R^n} = 2R^{n+1}\frac{\pi^{\frac{n+1}{2}}}{\Gamma(\frac{n+1}{2})}. \tag{3.8}$$

Peculiarly, as the dimension of the hypersphere increases, the surface area rapidly decreases, which may lead to unstable computations since there is a limit to how small numbers can be modelled using numerical floating point precision. This property may limit learning in this space, since optimization for high dimensional images often have lower but still high dimensional latent representations [36].

3.2 Mapping onto the Hypersphere

The problem of learning invariant embedding structure using a neural network h is subject to the geometry of the embedding space. The problem can be posed using the intrinsic or extrinsic view of the manifold. The intrinsic view requires that we define the manifold in terms of its charts, and the extrinsic view assumes that the manifold is embedded directly into $\mathbb{R}^n$. We note the definition of the hypersphere in section 3.1 is in accordance with the extrinsic view.

Defining the output of a neural network in terms of a manifold requires that all embeddings produced must be mapped directly to the geometry of choice. In order to do so, structural guidance is necessary in terms of a loss function. We can penalize these embeddings using a zero-centered hypersphere loss or with a ball loss of radius R. These losses would be specified for h, given a data distribution p_{data} as:

$$\mathcal{L}_{\mathsf{sphere}}(h; R, p, q) = \mathop{\mathbf{E}}_{\mathbf{x}\sim p_{\mathsf{data}}}[(R - ||h(\mathbf{x})||_2^p)^q], \qquad R, p, q > 0. \tag{3.9}$$

$$\mathcal{L}_{\mathsf{ball}}(h; R, p) = \mathop{\mathbf{E}}_{\mathbf{x}\sim p_{\mathsf{data}}}[(R - \max(R, ||h(\mathbf{x})||_2))^p)], \qquad R, p > 0. \tag{3.10}$$

It would be nearly impossible to learn a model with zero embedding error, especially when considering that this loss must be integrated in on top of a regularized invariance based loss. In order to circumvent any issues related to the geometry of choice, we define

the output of a neural network in terms of an ambient space and leverage a mapping onto the hypersphere as a function of that space.

Let the ambient space be defined as a subspace of $\mathbb{R}^{n+1}$. In the context of learning representations, we define the ambient space as the output of a neural network. In order to analyze behavior on the hypersphere, we require a map $\pi_R : \mathbb{R}^{n+1} \to \mathbb{S}^n_R$ from the ambient space to the hypersphere. There are a few different methods of describing a map and each come with their own benefits and limitations. We present and derive metrics for extrinsic and intrinsic views of the hypersphere using two different mappings as projections onto the hypersphere. The first projection is defined using the extrinsic view of the hypersphere using the closes point model. The second projection is defined using the intrinsic view of the hypersphere and is defined using the stereographic model.

3.2.1 Closest Point Projection

The **closest point** projection in the context of this thesis in is built on the Euclidean distance model. It is defined by an injective linear projection mapping from an extrinsically defined ambient space to a target hyperspherical space as $\rho_R : \mathbb{R}^{n+1} \setminus \{\mathbf{0}\} \to \mathbb{S}^n_R$. It is defined using a normalization operation that is equivalent to projecting an element in the ambient space to the hypersphere using the shortest Euclidean distance between the point and the hypersphere [37]. This operation is valid everywhere in the ambient space, excluding the origin, since it is equidistant to the entire manifold. For a sample $\mathbf{z} \in \mathbb{R}^{n+1} \setminus \mathbf{0}$ the mapping is defined as:

$$\rho_R(\mathbf{z}) = R\frac{\mathbf{z}}{||\mathbf{z}||_2}. \tag{3.11}$$

A suitable metric on this space is formed by pushing the projection operation forward through the angular metric operation. Given two samples in the ambient space $\mathbf{u}, \mathbf{v} \in$

$\mathbb{R}^{n+1} \setminus \mathbf{0}$ the proper metric pushed onto the hypersphere is:

$$\begin{aligned} d_{\mathbb{R}^{n+1},R}(\mathbf{u},\mathbf{v}) &= d_{\mathbb{S}^n_R}(\rho_R(\mathbf{u}),\rho_R(\mathbf{v})) \\ &= d_{\mathbb{S}^n_R}\left(R\frac{\mathbf{u}}{||\mathbf{u}||_2}, R\frac{\mathbf{v}}{||\mathbf{v}||_2}\right) \\ &= R\cos^{-1}\left(\frac{\langle \mathbf{u},\mathbf{v}\rangle_2}{||\mathbf{u}||_2||\mathbf{v}||_2}\right). \end{aligned} \tag{3.12}$$

Since this operation is non-injective, it fails to preserve certain topological properties of the original space. Additionally, it is possible to lift an ambient vector onto the hypersphere in a higher dimension, assuming that for all $\mathbf{z} \in B_R^{n+1}(\mathbf{0})$. The missing component is found implicitly as $\sqrt{R - ||\mathbf{z}||_2^2}$ [37]. In the case where this is possible, topological properties of the space are preserved. In practice, this method requires an additional constraint on the norm, similar to that presented at the beginning of 3.2 or a clipping of the norm, which can lead to biased computations. Since we assign a projection model to a learning process as a function of the choice of space, we choose to denote a generic projection as π_R rather than ρ_R.

3.2.2 Stereographic Projection

In order to circumvent the non-injective mapping onto the hypersphere, an isomorphism preserving certain topological properties is defined between the hypersphere and the ambient space. This isometry corresponds to the hypersphere's chart given an intrinsic view. Fortunately, only one chart is required to represent the hypersphere, and the map is called the **stereographic projection**. The stereographic projection is a non-linear conformal (angle preserving) mapping from the hypersphere to the ambient space and is valid everywhere on $\mathbb{S}^n_R$ except at its north pole $\boldsymbol{\mu}_0 = (R, 0, \ldots, 0)$. Geometrically, the stereographic projection of a point from the hypersphere onto the ambient hyperplane is formed by the line passing from the north pole through the point and onto the hyperplane seen in figure 3.3. Since this operation is a non-linear warping of the space, a conformal factor must be applied to the metric tensor when measuring distances. This factor is accounted for when describing the projection itself.

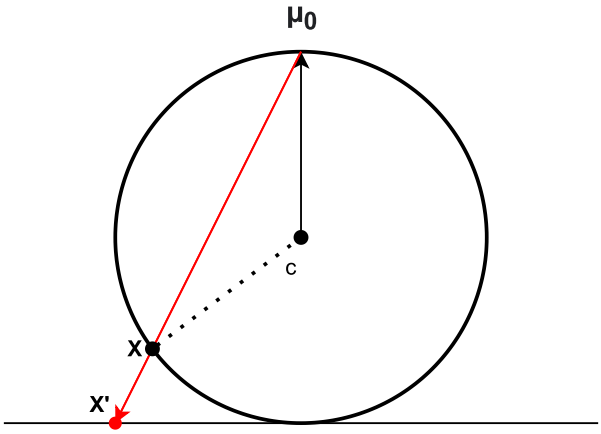

Figure 3.3: A stereographic projection from the hypersphere onto the plane measured from the north pole $\boldsymbol{\mu}_0$. The point $\mathbf{x}$ is mapped to $\mathbf{x}'$ in ambient space

Given that the projection is defined from the hypersphere to the ambient space, the inverse mapping is required. The inverse stereographic mapping for a constant positive curvature space to an ambient vector is $\rho_R^{*-1} : \mathbb{D}_R^n \to \mathbb{S}_R^n \setminus \boldsymbol{\mu}_0$. $\mathbb{D}_R^n$ is viewed as the mapping from the hypersphere, excluding the north pole, back to the ambient space $\rho_R^*(\mathbb{S}_R^n \setminus \boldsymbol{\mu}_0)$. The inverse map lifts an ambient vector into a higher dimension, similar to the missing component method. It also has the benefit of being defined everywhere, thus avoiding the restrictions on the ambient space's domain. The forwards mapping $\rho_R^{*-1}(\mathbf{z})$ is defined as follows, using a tuple to denote the additional component and has conformal scaling factor applied [37].

$$\rho_R^{*-1}(\mathbf{z}) = \Big(\underbrace{2R^2 \frac{\mathbf{z}}{R^2 + ||\mathbf{z}||_2^2}}_{\mathbb{R}^n}, \underbrace{R \frac{R^2 - ||\mathbf{z}||_2^2}{R^2 + ||\mathbf{z}||_2^2}}_{\mathbb{R}} \Big). \tag{3.13}$$

A suitable metric on this space is formed by pulling the projection operation backwards. Given two samples in the ambient space $\mathbf{u}, \mathbf{v}$ the proper metric pulled onto the hypersphere is:

$$\begin{aligned} d_{\mathbb{D}_R^n}(\mathbf{u}, \mathbf{v}) &= d_{\mathbb{S}_R^n}(\rho_R^{*-1}(\mathbf{u}), \rho_R^{*-1}(\mathbf{v})) \\ &= d_{\mathbb{S}_R^n}\Big(\Big(2R^2 \frac{\mathbf{u}}{R^2 + ||\mathbf{u}||_2^2}, R \frac{R^2 - ||\mathbf{u}||_2^2}{R^2 + ||\mathbf{u}||_2^2} \Big), \Big(2R^2 \frac{\mathbf{v}}{R^2 + ||\mathbf{v}||_2^2}, R \frac{R^2 - ||\mathbf{v}||_2^2}{R^2 + ||\mathbf{v}||_2^2} \Big) \Big) \\ &= R \cos^{-1} \Big(1 - \frac{2R^2 ||\mathbf{u} - \mathbf{v}||_2^2}{(R^2 + ||\mathbf{u}||_2^2)(R^2 + ||\mathbf{v}||_2^2)} \Big). \end{aligned} \tag{3.14}$$

Even though this method has the benefit of being bijective and smooth, it does suffer from exploding norms of the ambient vectors as backprojected elements approach the north pole on the hypersphere. Learning dynamics can also be impacted through weight decay, since this may inhibit how much learning can actually be done on the upper half of the hypersphere as the exploding norms become bounded. Once again, since we assign a projection model to a learning process as a function of the choice of space, we choose to denote a generic projection as π_R rather than ρ_R^*.

3.3 Distributions on the Hypersphere

It is convenient to understand distributions of elements on the hypersphere so that we may compare and contrast elements in terms of their likelihoods when trying to learn invariant embeddings. Probability densities can be described on the hypersphere using methods from directional statistics. These distributions are categorized by their symmetry with respect to the distribution mean [38]. Given that the hypersphere is circular and has periodic angular measurements, all elements are measured with respect to the mean, which is the relative reference point that summarizes the distribution. In order to frame any learning problem on the hypersphere, it is imperative that we understand the benefits and limitations of these distributions once imposed on the observed set of samples.

Consider a unimodal Gaussian in Euclidean space $\mathbb{E}^n$ with known mean vector $\boldsymbol{\mu}$ and scalar variance σ^2. The probability density $p(\mathbf{x} \mid \boldsymbol{\mu}, \sigma^2)$ of an element $\mathbf{x} \in \mathbb{E}^n$ is described as:

$$p(\mathbf{x} \mid \boldsymbol{\mu}, \sigma^2) = \frac{1}{\sqrt{2^n \pi^n} \sigma} \exp\Big(-\frac{||\mathbf{x} - \boldsymbol{\mu}||_2^2}{2\sigma^2}\Big). \tag{3.15}$$

If we reduce and condition the space of elements to be on the zero centered hypersphere with radius R where $\mathbf{x}, \boldsymbol{\mu} \in \mathbb{S}_R^n$ it is observed that the density is proportional to the similarity between the mean direction and an observed sample with normalization coefficient C:

$$p_{\mathbb{S}_R^n}(\mathbf{x} \mid \boldsymbol{\mu}, \sigma^2) = \frac{p(\mathbf{x} \mid \boldsymbol{\mu}, \sigma^2)}{\int_{\mathbb{S}_R^n} p(\mathbf{y} \mid \boldsymbol{\mu}, \sigma^2) d\mathbf{y}} = C \exp\Big(\frac{\langle \boldsymbol{\mu}, \mathbf{x} \rangle_2}{2\sigma^2}\Big). \tag{3.16}$$

The resulting distribution is called the von Mises-Fisher (vMF). It has been studied in great length and has asymptotic behavior that is of value depending on a learning problem's requirements. It is noted that the integral of the Gaussian along the hypersphere is non-trivial. This non-triviality is amplified when trying to generalize the result to a multivariate distribution with a covariance. This component is also proportional to the surface area of the hypersphere, which vanishes in higher dimensions. Due to the latter, we study the utility of the univariate formulation.

In the following two sections, the vMF and its counterpart, called the Power Spherical distribution (PS) is presented and decomposed as needed in order to lay the foundations for learning densities on the hypersphere in a self-supervised setting. We omit sections regarding sampling procedures and differential entropy, as it is not relevant in subsequent sections.

3.3.1 The von Mises-Fisher Distribution

As demonstrated in the section above, the vMF distribution is a natural parameterization of a Gaussian on the hypersphere. The vMF does have a minor modification, that is to say, it is induced on the unit hypersphere $\mathbb{S}^n$ where $R = 1$. The vMF also introduces a new parameter proportional to the inverse of the variance called the **concentration** κ. The concentration is a positive constant that naturally describes how elements are spread about a mean direction on the hypersphere. Since we have already described how distances between points vanish as a function of curvature in section 2.1, we say that the concentration can also be viewed as an expansion or contraction of the radius of the hypersphere. Using this new definition, we denote the vMF density $p_{\text{vMF}}(\mathbf{z}, \mid \boldsymbol{\mu}, \kappa)$ with known mean $\boldsymbol{\mu}$, concentration κ, and normalization coefficient $C_n(\kappa)$ for a random variable $\mathbf{z} \in \mathbb{S}^n$ with unit norm as [39].

$$p_{\text{vMF}}(\mathbf{z} \mid \boldsymbol{\mu}, \kappa) = C_n(\kappa) \exp(\kappa \langle \boldsymbol{\mu}, \mathbf{z} \rangle_2). \tag{3.17}$$

Here, the normalization coefficient is explicitly defined in terms of the concentration and dimension n using the modified Bessel function of the first kind order v as $I_v(\kappa)$.

$$C_n(\kappa) = \frac{\kappa^{\frac{n}{2}-1}}{(2\pi)^{\frac{n}{2}} I_{\frac{n}{2}-1}(\kappa)}. \tag{3.18}$$

Given the complex nature of this distribution, the mean and concentration can be inferred from samples taken from the data distribution $\mathbf{z} \sim p_{\text{data}}$ defined on $\mathbb{S}^n$ using MLE. The concentration is a non-linear function and can be refined using Newtons method, however, Sra's approximation is sufficient for all intents and purposes [39]. Let $\bar{\boldsymbol{\mu}}$ denote the unnormalized estimated mean direction and $\boldsymbol{\mu}$ denote the mean renormalized onto the hypersphere using the closest point projection. By Sra's approximation:

$$\bar{\boldsymbol{\mu}} = \mathop{\mathbf{E}}_{\mathbf{z}\sim p_{\text{data}}}[\mathbf{z}], \qquad \boldsymbol{\mu} = \frac{\bar{\boldsymbol{\mu}}}{||\bar{\boldsymbol{\mu}}||_2}, \qquad \kappa = \frac{||\bar{\boldsymbol{\mu}}||_2(n - ||\bar{\boldsymbol{\mu}}||_2^2)}{1 - ||\bar{\boldsymbol{\mu}}||_2^2}. \tag{3.19}$$

Although it is not shown, the concentration involves computations with respect to the inverse of the modified Bessel function of the first kind. Such an operation is intractable. In general, numeric estimation of the Bessel function in higher dimensions is also unstable and should be avoided when possible. Moreover, by the triangle inequality, we show that the estimate of the mean is smooth and bounded. As it follows, the concentration parameter is non-linearly proportional to the unnormalized estimate of the mean and thus yields interesting optimization utilities.

$$0 \leq ||\mathop{\mathbf{E}}_{\mathbf{z}\sim p_{\text{data}}}[\mathbf{z}]||_2 \leq \mathop{\mathbf{E}}_{\mathbf{z}\sim p_{\text{data}}}[||\mathbf{x}||_2] \leq 1. \tag{3.20}$$

The vMF distribution has two asymptotic behaviours of interest as a function of κ. Taking the limits demonstrate that the vMF converges to a uniform distribution or Dirac point mass as depicted in figure 3.4.

$$\lim_{\kappa\to 0^+} p(\mathbf{z} \mid \boldsymbol{\mu}, \kappa) = \mathcal{U}(\mathbb{S}^n). \tag{3.21}$$

$$\lim_{\kappa\to \infty} p(\mathbf{z} \mid \boldsymbol{\mu}, \kappa) = \delta_{\boldsymbol{\mu}}(\mathbf{z}). \tag{3.22}$$

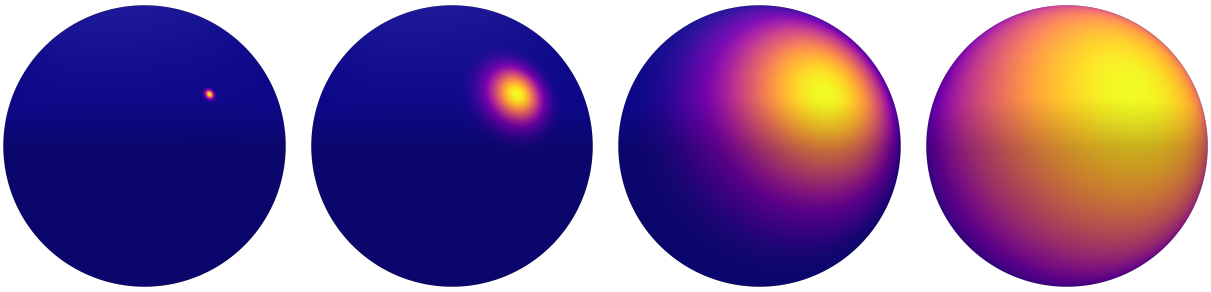

Figure 3.4: Diffusion of density on the 3-sphere as a function of distribution concentration. The asymptotic behavior of the concentration starts from a tight point mass (leftmost sphere) and symmetrically diffuses over the sphere towards a uniform distribution (rightmost sphere). Regions of low relative density are colored in blue, while regions of higher relative density are colored in yellow.

We may also define $||\bar{\boldsymbol{\mu}}||_2$ as linear estimator for concentration which can be used to control the learning dynamics on the hypersphere since it is analogous to the variance of the distribution. Using Sra's approximation, we derive loose upper and lower bounds for the concentration.

$$||\bar{\boldsymbol{\mu}}||_2 \leq \frac{||\bar{\boldsymbol{\mu}}||_2(n - ||\bar{\boldsymbol{\mu}}||_2^2)}{1 - ||\bar{\boldsymbol{\mu}}||_2^2} \leq \frac{n}{1 - ||\bar{\boldsymbol{\mu}}||_2}. \tag{3.23}$$

Clearly, maximizing the lower bound with upper limit of 1 is proportional to aligning all the points in the distribution towards the Dirac point mass. Minimizing the upper bound towards 0 is therefore the same as pushing the distribution to be uniform on the hypersphere. These results are sensible since by definition, the concentration is analogous to the unimodal variance on the hypersphere.

3.3.2 The Power Spherical Distribution

The PS distribution is another distribution defined on the unit hypersphere $\mathbb{S}^n$. The PS distribution is created to mitigate stability issues with respect to the vMF distribution. The PS distribution has the same benefits as the vMF in terms of its symmetry and known Kullback-Leibler divergence. It also benefits from being numerically stable in high dimensions and high concentration regimes. The PS distribution is defined using the power family rather than the exponential family. It too is described using a mean and concentration parameter and has the benefit of being formulated in terms of a marginal affine Beta distribution with α, β parameters as a function of dimensionality of the hypersphere and

its concentration [40].

$$\alpha = \frac{n-1}{2} + \kappa, \qquad \beta = \frac{n-1}{2}. \tag{3.24}$$

$$p_{PS}(\mathbf{z} \mid \boldsymbol{\mu}, \kappa) = \frac{\Gamma(\alpha+\beta)}{2^{\alpha+\beta}\pi^{\beta}\Gamma(\alpha)}(1 + \langle \boldsymbol{\mu}, \mathbf{z} \rangle_2)^{\kappa}. \tag{3.25}$$

A major benefit of the PS distribution is in relation to the normalization coefficient. Gamma functions are commonly available in auto-differentiation packages, are stable, and computationally tractable.

3.3.3 General Kernel Distributions

We note that the unnormalized vMF and PS can be rewritten in terms of the Gaussian and polynomial kernels using a Euclidean metric instead of the ℓ_2 inner product. These kernels satisfy the requirement that any distribution on the hypersphere must be symmetric about its mean $\boldsymbol{\mu}$. As a result, we extend the family of distributions on the hypersphere to all positive bounded symmetric kernels with parameters $\boldsymbol{\sigma}$. Given a valid unnormalized kernel $K_{\boldsymbol{\sigma}}$ and a mean $\boldsymbol{\mu}$, a **kernel density** $p_{\mathsf{K}}(\mathbf{z} \mid \boldsymbol{\mu}, \boldsymbol{\sigma})$ can be constructed for samples $\mathbf{z} \in \mathbb{S}^n_R$ as:

$$p_{\mathsf{K}}(\mathbf{z} \mid \boldsymbol{\mu}, \boldsymbol{\sigma}) = \frac{K_{\boldsymbol{\sigma}}(\mathbf{z}, \boldsymbol{\mu})}{\int_{\mathbb{S}^n_R} K_{\boldsymbol{\sigma}}(\mathbf{z}', \boldsymbol{\mu}) d\mathbf{z}'}, \qquad \int_{\mathbb{S}^n_R} K_{\boldsymbol{\sigma}}(\mathbf{z}', \boldsymbol{\mu}) d\mathbf{z}' < \infty. \tag{3.26}$$

The result of this generalization implies that we may now construct distributions in terms of kernels that are a function of different metrics like the angular metric [41]. Each metric has slightly different learning dynamics since their gradients on the sphere may differ from one another. These gradients let us design and select how elements push and pull each other on the sphere and shall allow us to better tune the learning dynamics for the invariance problem.

3.4 Energy on the Hypersphere

It is possible to model how a set of embeddings interact with each other on the sphere in terms of their pairwise potentials. We may then optimize the configuration of points subject to a desired energy state. Let ν denote a Borel probability measure that assigns

probability density to each region on $\mathbb{S}_R^n$ and let $G(\mathbf{x}, \mathbf{y}) : \mathbb{S}_R^n \times \mathbb{S}_R^n \to \mathbb{R}_+$ model the pairwise potential between elements $\forall \mathbf{x}, \mathbf{y} \in \mathbb{S}_R^n$. The **local energy** function $V_\nu(\mathbf{x})$ about an element $\mathbf{x}$ is defined and approximated as [42, 43, 44]:

$$V_\nu(\mathbf{x}) = \int_{\mathbb{S}_R^n} G(\mathbf{x}, \mathbf{y}) d\nu(\mathbf{y}) = \mathbf{E}_\mathbf{y}[G(\mathbf{x}, \mathbf{y})]. \tag{3.27}$$

We define the **global energy** function U_ν as an aggregation over local energy functions V_ν as [11, 42]:

$$U_\nu = \int_{\mathbb{S}_R^n} V_\nu(\mathbf{x}) d\nu(\mathbf{x}) = \int_{\mathbb{S}_R^n} \int_{\mathbb{S}_R^n} G(\mathbf{x}, \mathbf{y}) d\nu(\mathbf{x}) d\nu(\mathbf{y}) = \mathbf{E}_{(\mathbf{x},\mathbf{y})}[G(\mathbf{x}, \mathbf{y})]. \tag{3.28}$$

Similar to classical electrostatic potential theory, each element on the hypersphere induces a field along its surface. The local field measured at a point $E_\nu(\mathbf{x})$ is induced by the set of alternate points on the hypersphere. The interactions between such points are measured by the local potentials and the field can be recovered using the Euclidean gradient operator [42]. An element is said to flow along the field until it reaches a stationary point where the net flow is zero. Since elements are restricted to the hypersphere, gradients observed along the radial direction measured from the center of the hypersphere do not contribute to any component of the flow. We therefore apply the tangential projection to the field in order to recover the gradients in the tangent space referenced to the element.

$$E_\nu(\mathbf{x}) = -P_\mathbf{x} \nabla_\mathbf{x} V_\nu(\mathbf{x}). \tag{3.29}$$

In order to use any potential model, it is important to define a pairwise potential function G. It is known that solving Poisson's equation in free space yields the Coulomb potential function which is a special case of Riesz's potential, however it is more convenient to define a few general requirements for all potential functions. For any compact set $\mathcal{X} \subseteq \mathbb{R}^n$ there exists a positive definite kernel that defines the **kernel energy** (k-energy) [43, 44]. Since the hypersphere is compact, we may replace the definition of G using any valid potential kernel $K_\mathbf{s}$ with parameters $\mathbf{s}$. Kernels used to define potential functions must also be symmetric, however there are no constraints on boundedness. Potential kernels

are subdivided into two categories. Long range kernels like Riesz's kernel, also known as Riesz's s-kernel have slow decay rates inversely proportional to the degree of the chosen polynomial. Short range kernels like the Gaussian kernel have fast decay rates. It is possible to tune the density of elements on the sphere as a function of how each element interacts with each other. As a result, it is important to understand the range of each kernel when constructing a learning problem where elements receive feedback from the system as a function of its range

3.5 Gradients on the Hypersphere

Gradient descent is an essential tool required to minimize loss functions in a deep learning setting. Gradients are typically computed in a Euclidean space and thus require some consideration when moving to the hypersphere. Spherical gradients can be computed from their Euclidean counterpart using an appropriate remapping protocol [45]. Consider an element on the sphere $\mathbf{z} \in \mathbb{S}_R^n$ with an objective function $\mathcal{L}$. Let $\nabla_{\mathbf{z}}^{\mathbb{E}^n}$ denote the Euclidean gradient operator and $\nabla_{\mathbf{z}}^{\mathbb{S}_R^n}$ be the spherical gradient operator both at $\mathbf{z}$. The updated position of an element undergoing motion due to a gradient descent update with step size α is described by the Euclidean update rule at time t as:

$$\mathbf{z}^{(t+1)} = \mathbf{z}^{(t)} - \alpha \nabla_{\mathbf{z}}^{\mathbb{E}^n} \mathcal{L}. \tag{3.30}$$

We may loosely view the gradients as a force acting on an element in the system. Since the elements are restricted to movement on the hypersphere, the force is restricted to the tangent space observed at the position of the element. The radial components are eliminated as they are not valid degrees of freedom. In other words, we can map Euclidean gradients to the tangent space $T_{\mathbf{z}^{(t)}}\mathbb{S}_R^n$ at the point $\mathbf{z}^{(t)}$ using the tangential projection $P_{\mathbf{z}^{(t)}}$ such that $\nabla_{\mathbf{z}}^{\mathbb{S}_R^n} = P_{\mathbf{z}^{(t)}} \nabla_{\mathbf{z}}^{\mathbb{E}^n}$. The spherical operator can now be written in terms of the Euclidean operator. The update rule is modified to use the exponential map and guarantees

that for each iteration of gradient descent, the elements stay on the hypersphere.

$$\mathbf{z}^{(t+1)} = \exp^R_{\mathbf{z}^{(t)}}(-\alpha P_{\mathbf{z}^{(t)}} \nabla^{\mathbb{E}^n}_{\mathbf{z}} \mathcal{L}). \tag{3.31}$$

The same principle can be applied to weights of a neural network, however, it is not necessarily clear how to properly apply the exponential map. If the map is applied to samples on the hypersphere, it is not clear how to back-propagate their gradients to the weights of a neural network. As a means to circumvent this issue, we analyze the behaviour of the exponential map under differential update steps of size ϵ.

$$\lim_{\alpha \to \epsilon} \exp^R_{\mathbf{z}^{(t)}}(-\alpha P_{\mathbf{z}^{(t)}} \nabla^{\mathbb{E}^n}_{\mathbf{z}} \mathcal{L}) \approx \mathbf{z}^{(t)} - \epsilon P_{\mathbf{z}^{(t)}} \nabla^{\mathbb{E}^n}_{\mathbf{z}} \mathcal{L}. \tag{3.32}$$

The above demonstrates that under infinitesimally small updates, a path taken by an element along a path in a direction tangent to the manifold is still on the manifold. As the number of steps increases, the error is expected to increase as well. It is assumed that the error accumulated is marginal if the step size is sufficiently small. If we further modify the space to leverage the Euclidean distance mapping back onto the hypersphere at the end of each iteration, the accumulated error becomes trivial. This is also equivalent to adding the radial loss mentioned in section 3.2, since the update recovers the radial component lost to the tangent space operation. In theory, we may circumvent these issues by trying to solve the problem in continuous time, however, this is not computationally feasible. For the remainder of this work, we shall presume that under sufficiently small learning rate conditions, the path taken by an embedding mapped to the hypersphere is not problematic. Moreover, we note that neural networks are not smooth and we care most about the direction of the update in the tangent space.

3.6 Summary

In this chapter, we introduced and defined the hypersphere as well as the metrics that can be imposed on it. We also categorized the hypersphere in terms of its intrinsic and extrinsic views and showed how to map onto the space using the closest point or inverse

stereographic projection depending on the view point taken. We introduced the concepts of a distribution and a pair potential on the hypersphere and showed their generalized kernel analogue. Finally, we showed it is possible to learn on the hypersphere with Euclidean gradients if the tangent space is properly integrated into an update step. In the next chapter, we specify the self-supervised problem on the hypersphere in terms of an alignment and diversity objective. We discuss the presence of hidden coupling mechanisms and analyze various optimization procedures in terms of the type of observable coupling.

4

Learning on Hyperspheres

Now that we have established the tools to learn on spherical spaces, we pose the problem of learning image representations in a self-supervised setting without access to the true label distribution on the hypersphere given a distortion-invariant pretext task. We define a set of objectives required to learn invariant representations from a geometric perspective on a manifold with particular structure. While it is possible to model the problem over any manifold, the hypersphere is closed, bounded, and has a variety of meaningful metrics that can be used. The latter makes the hypersphere ideal, as we do not have to worry about vanishing densities. We are able to control the density of the space by manually tuning its radius, since all elements becoming close to far from one another at a rate proportional to the curvature of the hypersphere.

The regularized invariant-diversity problem outlined in section 2.3 is solved with differ-

ent optimization dynamics as a function of the metric chosen on the hypersphere as well as the choice of diversity constraint. Each choice imposes specific contrastive properties which impact how elements interact with one another. In this chapter, we reintroduce the invariance object in terms of a feature alignment objective, since elements are measured as a function of their angle when remapped to the hypersphere. We then explore different perspectives of feature diversity on the hypersphere by imposing constraints that include variance, entropy, energy, and orthogonality all on samples embedding in this space.

4.1 Learning Objectives on the Hypersphere

Before diving into any concepts related to learning, we define the three main distributions required to subdivide the contrastive learning problem. We assume there exists a set of images that are independently and identically distributed (*i.i.d*). Let p_{data} be the **data distribution** and let p_{aug} be the **augmentation distribution**. Let p_{pos} be the **positive distribution** where a n-tuple (group of n elements) sampled from p_{pos} is the set of augmented views of a single sample drawn form p_{data}. Here, $(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n) \sim p_{\mathsf{data}}$ is equivalent to sampling $\mathbf{x} \sim p_{\mathsf{data}}$ and then generating a set of views as $(t_1(\mathbf{x}), t_2(\mathbf{x}), \ldots, t_n(\mathbf{x}))$ given n augmentations sampled from $t_i \sim p_{\mathsf{aug}}$. We chose to define p_{neg} as the **negative distribution**. Samples drawn from the negative distribution are *i.i.d* samples from the data distribution that are then augmented using perturbations sampled from the augmentation distribution. 2-tuples drawn from the positive distribution are called a positive pair, while 2-tuples drawn from the negative distribution are called a negative pair.

Learning structure on the hypersphere can be decomposed into two core objectives. The main objective can be described through the lens of **feature alignment** which is the invariance analogue on the hypersphere. Since all elements existing on the hypersphere have measurable distances proportional to the angle measured between them, minimizing their distances is the same as aligning their directional vectors referenced from the spherical origin. Spherical alignment plays a crucial role in learning invariant embeddings, since a sample perturbed by various transformations are invariant if and only if the angles spanned between their mapped embeddings is minimal. Given a set of pair-

wise angles $\theta \in \Theta$ and sufficient alignment condition ϵ_θ, we say the set of embeddings is aligned if $\sup_{\theta \in \Theta} \theta < \epsilon_\theta$. In practice, this is accomplished by minimizing the variance over the set of paired positive samples subject to a selected metric. In contrast to the alignment objective, the secondary objective can be described through the lens of **feature diversity**. Feature diversity is the complement view of feature alignment. Here, it is desirable to have a set of features over the negatives with sufficient angular differences between them. There are countless different means of accomplishing diversity. For instance, diversity can be described using variance. Here, it would be desirable that the set of pairwise angles over the negatives $\theta \in \Theta$ subject to a sufficient diversity condition ϵ_θ is satisfied using the variance $\mathbf{V}[\Theta] > \epsilon_\theta$. Similarly, we may also desire some maximum entropy or uniformity over the set of observed random variables, noting that the distribution of points that maximizes the entropy on the hypersphere is the uniform distribution.

The self-supervised object is restated in terms of feature alignment and diversity as a function of positive and negative distributions. We learn weights of neural network defined using a backbone-projector pair equipped with a projection map and metric d_* as $h : X \to \mathbb{S}^n_R$ where $h_{\pi_*,\theta,\phi} = \pi_* \circ g_\phi \circ f_\theta$. In this section, the goal is to minimize the **global objective** of feature alignment and diversity subject to the loss functions $\mathcal{L}_{\mathsf{a}}(h; \mathbf{a})$ with parameter set $\mathbf{a}$ and $\mathcal{L}_{\mathsf{d}}(h; \mathbf{b})$ with parameter set $\mathbf{b}$ and balancing term λ as:

$$\mathcal{L}_{\mathsf{g}}(h; \mathbf{a}, \mathbf{b}, \lambda) = \mathcal{L}_{\mathsf{a}}(h, \mathbf{b}) + \lambda \mathcal{L}_{\mathsf{d}}(h; \mathbf{b}). \tag{4.1}$$

4.2 Learning Dynamics

The main objective of learning aligned and diverse features is vague and requires more understanding about how learning dynamics take place on the hypersphere. We understand that samples from p_{pos} are embedded onto the hypersphere using h must be aligned subject to a metric d_Z. This view is quite simple as there is little dependence on understanding the entire system of elements, rather, it only requires the understanding over the distribution of augmentations. As a result, we can summarize the entire alignment

objective by minimizing the p-power pairwise distances where $p = 2$ corresponded to the variance:

$$\mathcal{L}_{\mathsf{align}}(h; p) = \underset{(\mathbf{x},\mathbf{y})\sim p_{\mathsf{pos}}}{\mathbf{E}}[d_Z(h(\mathbf{x}), h(\mathbf{y}))^p], \qquad p > 0. \tag{4.2}$$

The main choice in terms of the alignment objective is related to the choice metric on the space, as well as the power. On the other hand, there is no clear objective related to feature diversity and special care must be taken when constructing its loss functions. Both objectives introduce the idea of **element coupling**, which specifies the interactions between elements that have pairwise relationships depending on their source distributions. We note that comparisons between positives is defined as **positive-positive** coupling (PPC), comparisons between negatives is defined as **negative-negative coupling** (NNC) and comparisons across the two distributions is defined as **negative-positive coupling** (NPC). These components may be introduced in different ways depending on the structure of the loss [10]. Here, we chose to define coupling in terms of the pairwise comparisons rather than a multiplicative factor, as specified by DCL. We discuss the importance of coupling at a high level and investigate its function across various diversity objectives.

4.2.1 Gradients of the Metric

In order to understand the pros and cons of the angular or Euclidean metric on the hypersphere, it is imperative that we analyze their gradients. Given two elements $\mathbf{u}, \mathbf{v} \in \mathbb{R}^{n+1}$, we show that the closest point projection map π_R to $\mathbb{S}^n_R$ produces gradients in the tangent space and does not require additional projection. We apply the Euclidean gradient operator to the projection itself and show that it exists in the proper tangent space as follows:

$$\begin{aligned} \nabla_{\mathbf{u}} \pi_R(\mathbf{u}) &= \nabla_{\mathbf{u}} \frac{\mathbf{u}}{||\mathbf{u}||_2} \\ &= \frac{1}{||\mathbf{u}||_2}\Big(I - \frac{\mathbf{u}\mathbf{u}^T}{||\mathbf{u}||_2^2}\Big) \\ &= \frac{1}{||\mathbf{u}||_2} P_{\mathbf{u}}. \end{aligned} \tag{4.3}$$

This operation produces vectors in the tangent space proportional to the length of the initial vector and as a result, additional operations to account for the tangent space are

not necessary. We chose to analyze properties related to the square metric, since mean squared errors are smooth and more often than not, they are easier to optimize than their non-squared counterpart. Using chain rule, we analyze the behaviour of the squared Euclidean metric as a function of its gradients and its magnitude as follows:

$$\begin{aligned}\nabla_{\mathbf{u}} d_{\ell_2,R}(\pi_R(\mathbf{u}), \pi_R(\mathbf{v}))^2 &= \nabla_{\mathbf{u}} \Big|\Big| \frac{\mathbf{u}}{||\mathbf{u}||_2} - \frac{\mathbf{v}}{||\mathbf{v}||_2} \Big|\Big|_2^2 \\ &= \frac{2}{||\mathbf{u}||_2} P_{\mathbf{u}} \Big(\frac{\mathbf{u}}{||\mathbf{u}||_2} - \frac{\mathbf{v}}{||\mathbf{v}||_2} \Big) \\ &= -2 P_{\mathbf{u}} \frac{\mathbf{v}}{||\mathbf{v}||_2 ||\mathbf{u}||_2}.\end{aligned} \tag{4.4}$$

$$||\nabla_{\mathbf{u}} d_{\ell_2,R}(\pi_R(\mathbf{u}), \pi_R(\mathbf{v}))^2||_2 \propto \sin \theta_{\mathbf{u},\mathbf{v}}. \tag{4.5}$$

It is observed that in equation 4.4, the tangent operator $P_{\mathbf{u}}$ annihilates the direction of the gradient associated to $\mathbf{u}$ and the components of $\mathbf{v}$ in the direction of $\mathbf{u}$, leaving the alternate component to dominate the direction with a sine dependency. This dependency is outlier resistant, as it is at its maximum when the vector pair is orthogonal. It is smooth and decreasing as the vector pair approach each other, and contributes less as they are farther apart. This can be beneficial in the case where a bad perturbation is applied to a positive sample. Ideally, in this case, the two elements would be far apart by virtue of the extreme perturbation. The metric avoids biasing the update step by minimizing the magnitude of the gradients associated to the bad perturbation. An example of this is seen when applying random crops to an input image. If the crop is faulty, it may result in the background being compared to a foreground object which is undesirable. The built-in resistance makes this situation less problematic. On the other hand, for settings where there are no issues with the perturbations, the sine dependency minimizes the importance of good samples which should not be far apart which is contrary to the alignment objective. These sets of pairs should be considered more important than others. We compare and

contrast these dynamics with those of the squared angular metric:

$$\begin{aligned}\nabla_{\mathbf{u}} d_{\mathbb{S}_R^n}(\pi_R(\mathbf{u}), \pi_R(\mathbf{v}))^2 &= \nabla_{\mathbf{u}} R^2 \cos^{-1}\Big(\frac{\langle \mathbf{u}, \mathbf{v}\rangle_2}{||\mathbf{u}||_2||\mathbf{v}||_2}\Big)^2 \\ &= \frac{-2R^2}{\sqrt{1-\frac{\langle \mathbf{u},\mathbf{v}\rangle_2}{||\mathbf{u}||_2||\mathbf{v}||_2}^2}} \cos^{-1}\Big(\frac{\langle \mathbf{u}, \mathbf{v}\rangle_2}{||\mathbf{u}||_2||\mathbf{v}||_2}\Big) P_{\mathbf{u}} \frac{\mathbf{v}}{||\mathbf{u}||_2||\mathbf{v}||_2} \qquad (4.6) \\ &= \frac{R^2\theta_{\mathbf{u},\mathbf{v}}}{\sin\theta_{\mathbf{u},\mathbf{v}}} \nabla_{\mathbf{u}} d_{\ell_2,R}(\pi_R(\mathbf{u}), \pi_R(\mathbf{v})).\end{aligned}$$

$$||\nabla_{\mathbf{u}} d_{\mathbb{S}_R^n}(\pi_R(\mathbf{u}), \pi_R(\mathbf{v}))^2||_2 \propto \theta_{\mathbf{u},\mathbf{v}}. \qquad (4.7)$$

Here, the gradients vary with a linear proportion to the angle between them. This setting is outlier sensitive and as seen, has a scaled correction factor applied to the Euclidean metric removing the sine bias. This metric has the inverted benefits of the Euclidean metric. In this setting, it considers far away points to be more important. Clearly, one method may provide more benefit than another depending on the upstream problem specification, since the user determines how challenging the problem can be as a function of augmentation settings. A major note for use cases involving the angular metric is with respect to its numerical stability due to the presence of poles. This occurs when vectors are perfectly aligned or misaligned. In practice, this operation contains some numeric instabilities and must be clamped. While this does yield biased gradients in the clamped regimes, the samples corresponding to these situations are not of value since they are either sufficiently, or far enough away where the bias is minimal.

4.2.2 Smooth Extrema Operations

When dealing with a learning problem, we wish to model a set of samples in terms of a system of elements with particular densities. It may be of interest to control and penalize the system as a function of its densities and modes as opposed to its expected value. We may try to describe the system in terms of its modes which can be approximated using various minimum and maximum operations. One major problem with using these operations is that they are not smooth and are sparse, as they only index a single value at a time. In order to circumvent these issues, a continuously differentiable smooth estima-

tor can be used to extract these extremal values. The smooth approximator is called the **LogSumExp** or softplus (LSE) and is defined over a set of elements X with scaling T as:

$$\mathsf{LSE}(X;T) = \frac{1}{T}\log\sum_{x\in X}\exp(Tx). \tag{4.8}$$

The scaling drives the separation of values and approximates the minimum or maximum in its limiting behavior:

$$\lim_{T\to\infty}\mathsf{LSE}(X;T) = \max X, \qquad \lim_{T\to-\infty}\mathsf{LSE}(X;T) = \min X. \tag{4.9}$$

Its gradient is also the softmax over the elements of the set:

$$\nabla_{x_i}\mathsf{LSE}(X;T) = \frac{\exp(Tx_i)}{\sum_{x_j\in X}\exp(Tx_j)}. \tag{4.10}$$

We can analyze learning behavior over a set of elements by looking at a distribution of pairwise interaction found in the contrastive learning problem. We may then optimize these extremal values in a smooth manner. Since the operator is smooth, samples sharing similar values have similar contributions to each other. This operator will be of utmost importance when investigating the properties of the diversity loss in contrastive algorithms in the sections to follow.

We note that the for any composite function, the LSE introduces a batch dependent coupling. Let f be a function acting on the input of the LSE. Expanding the gradients for a sample in the set using chain rule yields:

$$\nabla_{x_i}\mathsf{LSE}(X;T,f) = \underbrace{\frac{1}{\sum_{x_j\in X}\exp(Tf(x_j))}}_{\text{system scale}}\underbrace{\exp(Tf(x_i))}_{\text{sample weight}}\underbrace{\nabla_{x_i}f(x_i)}_{\text{chained gradient}}. \tag{4.11}$$

The per sample weight is constant for a fixed scale term, however, the rescaling term is a function of the entire set of observations. If samples from a distribution are too noisy, then there may be a chance that the scaling term introduced excessive coupling and places too much emphasis on certain sets of samples and may be inconsistent from batch to

batch. This phenomenon is expected to diminish as the number of samples in the system increases.

4.2.3 Hidden Coupling Mechanisms

We claim that in an invariance based self-supervised setting, a neural network can learn important features by uncovering hidden latent factors that are invariant to the perturbations applied to an input sample. We introduce an alignment loss which directly maximizes the similarity between the representations across perturbations, knowing that the sample should be similar to itself given some unknown invariant latent factors which can be extracted regardless of the noise applied. We assume that the neural network is able to uncover these latent factors, and we assume that they belong to a non-unique hidden class distribution present across a dataset. Non-uniqueness is desirable because it allows us to learn clusters of data that have similar semantic information without ever requiring access to the ground truth class label distribution when training. It is also problematic because in a contrastive setting, diversity regularization implicitly penalize pairs of negatives or pairs across the positive and negative distributions. It is highly possible that some pairs share similar latent factors. As a result, the diversity penalty introduces a competing and counterproductive goal in relation to the invariant alignment objective. This cost cannot be mitigated as we require diversity to avoid a non-homogeneous solution, however, special care can be taken to construct diversity penalties which limit the amount of interference with the alignment objective. As introduced in section 4.2, we define coupling as the counterproductive influence present across alignment and diversity tasks in terms of element comparisons. Depending on the design of the diversity loss, it is possible to introduce all three types of coupling. In particular, positive-positive coupling (PPC) may be present as a result of computing normalization coefficients and is avoidable. It can often times be removed by design. Alternatively, negative-positive coupling (NPC) and negative-negative coupling (NNC) exists in the presence of multiple samples with the sample hidden class variables. As the number of samples compared increases, the probability that some of the pairs share similar latent factors increases. This type of

coupling is unavoidable, however, it is possible to mitigate its effects.

4.3 Optimizing Distributions on the Hypersphere

In the following section, we aim to solve the alignment and diversity problem using techniques from contrastive learning that are centered around modeling elements as densities on the hypersphere. We present an initial view of the diversity objective by leveraging the asymptotic relationship of the vMF's concentration parameter and build an estimator for the task that is analogous to enforcing uniformity on the hypersphere. We extend the literature based on maximizing the log probability that positive pairs are most similar to each other when contrasted against sets of negatives. This is accomplished by generalizing the types of distributions that are used to parameterized embeddings on the hypersphere. The generalization is accomplished by modeling embeddings as densities using a statistical kernel built on top of different metrics, each of which is defined in sections 3.3 and 3.1. We present the interpretation of the cross-entropy task using a fixed parameter distribution in terms of a cluster matching assignment task and show that it is a generalization of SimCLR [6]. We then decompose cluster formulation and present a similar objective from the perspective of kernel density estimation (KDE) [11]. Finally, we address the challenges and weaknesses of each method based on the presence of NPC and propose an alternate reformulation to reduce its impact.

4.3.1 Optimizing Concentration

As shown in section 3.3.1, the concentration κ has two asymptotic properties related to its distribution. In particular, the limiting case for uniformity is of interest as it is the distribution with maximal entropy and diversity on the hypersphere. We've shown that κ has a loose upper bound that is minimized as a function of the mean. Based on this result, we let $\hat{\kappa} = ||\,\mathbf{E}_{\mathbf{x}\sim p_{\text{neg}}}[h(\mathbf{x})]||_2$ be an estimator of the concentration. We construct a loss

function that minimizes the estimator which is a surrogate for hyperspherical uniformity:

$$\mathcal{L}_{\mathsf{u},\kappa}(h;p) = || \underset{\mathbf{x}\sim p_{\mathsf{neg}}}{\mathbf{E}} [h(\mathbf{x})]||_2^p, \qquad p > 0. \tag{4.12}$$

We may define a similar analogue for the alignment objective by maximizing the lower bound of the linear estimator. The alignment objective is therefore defined as:

$$\mathcal{L}_{\mathsf{a},\kappa}(h;p,q) = (1 - || \underset{\mathbf{x}\sim p_{\mathsf{pos}}}{\mathbf{E}} [h(\mathbf{x})]||_2^p)^q, \qquad p,q > 0. \tag{4.13}$$

We note that the concentration alignment objective is proportional to $\mathcal{L}_{\mathsf{align}}$. This relationship is found by applying the polarization identity given the Euclidean metric. As a result, we chose to omit any further discussion related to the concentration alignment loss and strictly evaluate the behavior of the diversity penalty.

The concentration is a highly non-linear unbounded function and the estimator is a poor representation of its true value. It is observed that the estimator is tightly related to the empirical variance and energy on the hypersphere. These concepts are further explored in sections 4.4.1, 4.5.1.

4.3.2 Matching Models and Kernel Density Estimators

We wish to learn hidden latent factors associated with an image in a contrastive setting where positive perturbed samples are compared against other negative samples in a finite mini-batch. We assume the latent factors of interest are contained in the embeddings mapped to the hypersphere and assign each sample a class index associated to its position in the mini-batch. The class index is defined using an indicator function for the i^{th} sample as $c_i = \mathbb{1}_{[c \neq c_i]}$. We introduce a set of probability density functions centered about each element of the distribution on the hypersphere. We denote the per-sample density for each sample $\mathbf{z}_i$ on the hypersphere with distribution parameters $\boldsymbol{\sigma}_i$ conditioned on the class index c_i as $p(\cdot \mid c_i, \mathbf{z}_i, \boldsymbol{\sigma}_i)$. We represent the generalized density using any normalized symmetric kernel $K_{\boldsymbol{\sigma}_i}$ with normalization coefficient $n(\boldsymbol{\sigma}_i)$. We define the class prior $p(c_i)$ for each sample distribution. Using Bayes rule, we define the posterior for the i^{th} sample

over the set of all observed samples [46]. This posterior estimates the probability that a sample $\mathbf{z}$ is related to another sample in the mini-batch and is defined as:

$$p(c_i \mid \mathbf{z}, \mathbf{z}_i, \boldsymbol{\sigma}_i) = \frac{p(\mathbf{z} \mid c_i, \mathbf{z}_i, \boldsymbol{\sigma}_i)p(c_i)}{\sum_j p(\mathbf{z} \mid c_j, \mathbf{z}_j, \boldsymbol{\sigma}_j)p(c_j)} = \frac{n(\boldsymbol{\sigma}_i)K_{\boldsymbol{\sigma}_i}(\mathbf{z}, \mathbf{z}_i)p(c_i)}{\sum_j n(\boldsymbol{\sigma}_j)K_{\boldsymbol{\sigma}_j}(\mathbf{z}, \mathbf{z}_j)p(c_j)}. \tag{4.14}$$

Since we are working in an unsupervised setting under *i.i.d* assumptions with no knowledge about the underlying hidden class distributions, we can assume a constant uniform prior $p(c)$ such that $p(c_i) = p(c_j)$ for i, j pairs. We further simplify the problem by selecting a fixed set of kernel parameters $\boldsymbol{\sigma}$. The simplified posterior is therefore written as:

$$p(c_i \mid \mathbf{z}, \mathbf{z}_i, \boldsymbol{\sigma}_i) = \frac{K_{\boldsymbol{\sigma}}(\mathbf{z}, \mathbf{z}_i)}{\sum_j K_{\boldsymbol{\sigma}}(\mathbf{z}, \mathbf{z}_j)}. \tag{4.15}$$

We wish to maximize the probability that a sample is most similar to itself under a set of perturbations. This maximization is done by minimizing the distances between positives while maximizing the distances to the negatives. Geometrically, we assign each positive in the set of all combination pairs of positives to each other's distribution. Once assigned, we maximize the probability that this assignment is most correct. We construct the objective by minimizing the negative log-posterior over a finite set of observed samples. In practice, we sample positives and negatives jointly from the data distribution. We denote the set of positives and negatives with M perturbations and N unique samples as $Q = \{\mathbf{x}_{i,j}\}_{i,j}^{N,M}$ where the i^{th} index corresponds to the sample index and the j^{th} denotes the perturbation index. The fixed kernel matching loss is defined as:

$$L_k^{\mathsf{pos}} = \sum_{\substack{i,j \\ j>i}}^{M} K_\sigma(h(\mathbf{x}_{k,i}), h(\mathbf{x}_{k,j})), \qquad L_k^{\mathsf{neg}} = \sum_{\substack{l \\ l \neq k}}^{N} \sum_{i,j}^{M} K_\sigma(h(\mathbf{x}_{k,i}), h(\mathbf{x}_{l,j})). \tag{4.16}$$

$$\mathcal{L}_{\mathsf{mch}}(h; \boldsymbol{\sigma}) = -\frac{1}{NM(M-1)} \sum_{k}^{N} \sum_{\substack{i,j \\ j \neq i}}^{M} \log \frac{K_{\boldsymbol{\sigma}}(h(\mathbf{x}_{k,i}), h(\mathbf{x}_{k,j}))}{L_k^{\mathsf{pos}} + L_k^{\mathsf{neg}}}. \tag{4.17}$$

Rather than modeling the density from the perspective of a temperature, we model the problem from the perspective of the space's curvature. If the kernel parameter σ is con-

stant across all samples, it is equivalent to selecting the radius of the hypersphere that the elements exist on. As a result, we tune this parameter to modulate the distances as a function of the space. The space fills proportionally to the number of elements sampled in a batch and we may manually modify the density of elements on the space by selecting a specific radius. The choice of radius determines how close elements on the hypersphere are to one another and this closeness dictates how much element pairs push or pull on each other.

The generalization also reduces to SimCLR when selecting a single pair of positives, Gaussian kernel with Euclidean metric that corresponds to the vMF distribution, and kernel scalar parameter σ that is inversely proportional to the distribution temperature $2\sigma = \tau^{-1}$ [6, 11]. Let Q'_k denote the combinations of pairs used to compute the k^{th} normalization terms $L_k^{\text{pos}}, L_k^{\text{neg}}$. We use the formulation in equation 4.17 given the SimCLR settings to show equivalencies and demonstrate its relationship to both alignment and diversity objectives:

$$\begin{aligned}
\mathcal{L}_{\text{mch}}(h;\tau^{-1}) &= -\frac{1}{N}\sum_{i}^{N}\log\frac{\exp(-\tau^{-1}||h(\mathbf{x}_{i,0})-h(\mathbf{x}_{i,1})||_2^2)}{L_k^{\text{pos}}+L_k^{\text{neg}}} \\
&\propto \underbrace{\sum_{k}^{N}||h(\mathbf{x}_{i,0})-h(\mathbf{x}_{i,1})||_2^2}_{\text{alignment}} + \underbrace{\sum_{k}^{N}\text{LSE}(Q'_k;-\tau^{-1},d_{\ell_2,1})}_{\text{diversity}} \\
&\propto \mathcal{L}_{\text{SimCLR}}(h,\tau).
\end{aligned} \tag{4.18}$$

The alignment objective is recovered as expected and the diversity objective is stated in terms of a maximization of distances as a function of the most similar pairs in the set of observed samples. The SimCLR loss is known to perform poorly in low sample regimes and requires enormous batch sizes in order to stay competitive to more modern algorithms [10, 7]. We explore the relationship between the large sample dependency and the implicit formulation of the diversity objective as a result of modeling the problem in terms of the posterior. Taking a step back, it is clear that the goal of the diversity penalty is to maximize the separation of between pairs of elements in Q'_k proportionally to their distance. Separating all pairs is equivalent to maximizing the entropy of the empirical distribution

estimated by the batch. With any comparative method, there exists coupling mechanisms at play which control the amount of total separation possible. The LSE rescales the gradients of each term across the entire batch proportionally to the minimal distances in the set of all pairs observed. The consequence of this result is that the learning dynamics are always dictated by these similar pairs. Depending on the dataset, it is highly possible that multiple negative samples or cross positive-negative pairs in the batch share the same hidden latent factors as each other and indirectly introduce NPC and NNC via the L^{neg} term. The presence of these two types of coupling is problematic, however, we cannot avoid this phenomenon without major modifications to the loss function itself. We do however note that the set of element pairs is constructed with terms that also belong to the alignment objective. As the neural network learns to better align the positive pairs, the magnitude of their influence increases and introduces PPC via the L^{pos} as part of the diversity loss. This term is guaranteed to have the same latent factors by construction. The introduction of PPC explains why methods like SimCLR require large batch sizes, since there will always be factors introduced by the posterior normalization component of diversity objective that interferes with the alignment objective. As the number of samples in a batch increases, the likelihood that a pair of negatives share the same hidden latent factors increases. If there are sufficiently many negative-negative terms that dominate the diversity penalty, the impact of a few positive-positive terms vanishes. This is further supported by the literature on noise contrastive estimation. As the number of samples tend to infinity, the gradient of the noisy loss converges to the smooth gradients related to MLE of the true distribution [47]. In practice, real datasets have diverse class distributions and it is not possible to extend the number of samples indefinitely. Moreover, the computational cost of scaling the number of samples is often times too expensive. Many methods have tried to circumvent the requirements and cost of processing large batches of data by leveraging external offline tools to simulate the dynamics of a larger batch size. MoCo introduced large queues that cached previously seen samples. These samples are reintroduced in the diversity loss and extend the set of negatives to the entire data distribution in an attempt to reduce noise [8]. NNCLR improves the utility of the queue by querying it for the nearest neighbour of a sample, given that the LSE is dominated by

that neighbour to begin with. While these methods see some success, DCL proposes to remove PPC altogether by design [10]. Although the removal of the PPC clearly helps the objective, it is grounded in empirical and observational evidence.

We aim to reformulate the diversity loss as a means of circumventing PPC altogether without having to increase batch sizes all from the perspective of kernel density estimation (KDE). Given infinitely many samples, the diversity component of the matching loss is asymptotically equivalent to an entropy of the empirical distribution formed by the negatives. Fortunately, Wang and Isola have already demonstrated that KDE using a vMF-like kernel on the hypersphere in a contrastive setting can be used to penalize the entropy of the empirical distribution approximated by a batch of data [11]. We extend this method for the negative distribution across the set of all fixed parameter positive definite bounded symmetric kernels. We start by defining the kernel density estimate where we approximate the empirical distribution $\hat{p}_{\mathsf{KDE}}$ over the embedded hypersphere. Given a kernel with parameters $\boldsymbol{\sigma}$, an embedding neural network h and a negative data distribution p_{neg}, we construct the empirical distribution as [48, 49]:

$$\hat{p}_{\mathsf{KDE}}(\cdot \mid h, \boldsymbol{\sigma}) = \mathop{\mathbf{E}}_{\mathbf{y} \sim p_{\mathsf{neg}}}[K_{\boldsymbol{\sigma}}(h(\cdot), h(\mathbf{y}))]. \tag{4.19}$$

We note that the empirical entropy $\hat{H}$ is estimated using kernelized empirical distribution as [11, 48]:

$$\begin{aligned} \hat{H}_{\mathsf{neg}}(h; \boldsymbol{\sigma}) &= - \mathop{\mathbf{E}}_{\mathbf{x} \sim p_{\mathsf{neg}}}[\log \hat{p}_{\mathsf{KDE}}(\mathbf{x} \mid h, \boldsymbol{\sigma})] \\ &\approx - \mathop{\mathbf{E}}_{\mathbf{x} \sim p_{\mathsf{neg}}}[\log \mathop{\mathbf{E}}_{\mathbf{y} \sim p_{\mathsf{neg}}}[K_{\boldsymbol{\sigma}}(\mathbf{x}, \mathbf{y})]]. \end{aligned} \tag{4.20}$$

We can reformulate the matching loss by writing the alignment and diversity objectives in terms the entropy alignment and the asymptotic system entropy estimator. We assume the asymptotic formulation can be used instead of the finite element method, knowing that it implicitly avoids the problematic PPC components. We note that this formulation only contains NNC which cannot be avoided, however, the removal of PPC is progress nonetheless. In practice, we estimate the empirical distribution across sets of mini-batches

and use KDE as the estimator for the empirical entropy. We minimize the bias for this estimator in three ways. The first way is to construct the empirical estimator with a different set of samples that are used to evaluate the empirical entropy as specified in equation 4.20. The second method requires us to use a leave-one-out method, where each sample is compared to all other non-positive samples in the batch. Finally, we can operate under the assumption that all positive pairs are perfectly aligned with zero distance and compute the estimator directly from a single batch. The resulting loss does not produce counterproductive gradients (gradients opposite to the alignment goal) and introduces a smoother variant of the LSE which may be desirable during optimization [50]. The entropy variant of the matching loss is therefore written as the decoupled objective pair with a balancing parameter $\lambda > 0$. We expand the logarithm and extend the alignment and diversity objective for two types of kernels with parameters $\boldsymbol{\sigma}_{\mathsf{pos}}, \boldsymbol{\sigma}_{\mathsf{neg}}$. The two sample entropic matching loss is:

$$\mathcal{L}_{\mathsf{mch,e}}(h; \boldsymbol{\sigma}_{\mathsf{pos}}, \boldsymbol{\sigma}_{\mathsf{neg}}, \lambda) = \underbrace{- \mathop{\mathbf{E}}_{(\mathbf{x},\mathbf{y})\sim p_{\mathsf{pos}}} [\log K_{\boldsymbol{\sigma}_{\mathsf{pos}}}(h(\mathbf{x}), h(\mathbf{y}))]}_{\text{expected pairwise kernel entropy}} - \underbrace{\lambda \hat{H}_{\mathsf{neg}}(h; \boldsymbol{\sigma}_{\mathsf{neg}})}_{\text{empirical system entropy}}. \tag{4.21}$$

The entropy representation of the matching loss allows us to view the alignment in terms of the entropy over the augmentation distribution. Dimensionality collapse is a big topic of discussion in self-supervised learning, and the latter entropy loss implies that the alignment loss implicitly contracts the embedding space in an attempt to reduce the variance related to the applied perturbations. It also allows us to view the empirical entropy as a regularizer of the entire space which avoids collapsing to these trivial solution spaces.

In summary, it can be beneficial to model the interactions of elements on the hypersphere with a specific radius in terms of a fixed parameter normal distribution. This method provides intuition as to what is being mixed and matched on the space. We have also demonstrated where, how, and why elements interact with each other to learn stable invariant representation in as smooth of a means as possible. If we understand how the system operates in the limiting case, we can use the optimal expected behavior to better formulate the problem. These representations are thus guaranteed to be transferable to

other tasks with reasonable results given a proper selection of diversity loss.

4.3.3 A Note on Kernel Parameters

In the prior section, an entropy estimator is formulated under a fixed kernel parameter constraint. This assumption comes at a cost, since the solution space is restricted to element distributions whose interactions are driven by the parameter itself. It is possible to provide a global parameter or a per-sample parameter that is regressed from a neural network that are both optimized throughout the learning procedure, however, each option comes with its own problems. In the per-sample case, kernel normalization parameters must be computed. As seen with the vMF distribution, the normalization coefficients require us to compute modified Bessel functions. Not only are these coefficients expensive to compute, they are proportional to the surface area of the hypersphere, which is shown to vanish in higher dimensions in section 3.1. As a result, numerical underflow hinder the ability to adequately estimate normalized density used to drive the learning process. Moreover, we note that the diversity objective is artificially solved by inflating the concentration to high enough levels where the set of elements barely interact with one another. Due to the latter conditions, we restrict kernel parameters and perform a hyperparameter search to determine the optimal balance between attractive and repulsive operations that drive learning dynamics on the hypersphere.

4.4 Optimizing Distances on the Hypersphere

An alternative way to model the problem of learning invariant representations on the hypersphere is in terms of element separation measured using various distance functions. It is also possible to model the problem in terms of the variance and set orthogonality. We consider the joint objective of alignment and diversity by first defining the Fréchet variance $\mathbf{V}_{d_Z}$ in terms of a metric d_Z for a set of elements $Z \in \mathbb{S}_R^n$ on the hypersphere as:

$$\mathbf{V}_{d_Z}[Z] = \frac{1}{|Z|(|Z|-1)} \sum_{\substack{i,j \\ j \neq i}}^{|Z|} d_Z(\mathbf{z}_i, \mathbf{z}_j)^2. \tag{4.22}$$

We build on the definition of alignment and diversity given the definition of the generalized variance and attempt to learn sufficient separation of elements that will result useful invariant representation learned without supervision.

4.4.1 Maximal Variance on the Hypersphere

We would like to construct a diversity based optimization procedure that maximizes variance over the set of observations. Given the Euclidean metric on $\mathbb{S}_R^n$ the variance over a set *i.i.d* samples Z drawn from p_{sample} on the hypersphere is:

$$\begin{aligned}
\mathbf{V}_{d_{\ell_2,S_R^n}}[Z] &= \frac{1}{|Z|(|Z|-1)} \sum_{\substack{i,j \\ j \neq i}}^{|Z|} d_Z(\mathbf{z}_i, \mathbf{z}_j)^2 \\
&= \frac{1}{|Z|(|Z|-1)} \sum_{\substack{i,j \\ j \neq i}}^{|Z|} ||\mathbf{z}_i - \mathbf{z}_j||_2^2 \\
&= \mathop{\mathbf{E}}_{\mathbf{z} \sim p_{\text{sample}}} [||\mathbf{z}||]^2 - || \mathop{\mathbf{E}}_{\mathbf{z} \sim p_{\text{sample}}} [\mathbf{z}]||^2 \\
&= R^2 - ||\bar{\boldsymbol{\mu}}||_2^2.
\end{aligned} \tag{4.23}$$

The variance is shown to have an upper bound of R^2 since the space is bounded. The variance is therefore maximized subject to the minimization of the Euclidean sample mean. We note this result is identical to the concentration maximization in section 4.3.1. It is observed that the concentration is an implicit formulation of the variance. Moreover, we revisit the concept of uniformity in relation to the maximum variance problem. It is observed that maximal variance and minimal mean is a necessary but not sufficient condition for uniformity on the hypersphere. We prove that elements of the hypersphere that maximize the Euclidean variance is not a sufficient condition by counter example. Let us assume there exists an infinite set of uniformly distributed elements on a spherical cap $\Omega(R_0)$ with radius $R_0 < R - \epsilon$ of a hypersphere with radius R. We construct a new set by adding a complementary set $\bar{\Omega}(R_0)$ that is the mirror of $\Omega(R_0)$ on the opposite pole of the hypersphere. The union of the two sets $\Omega(R_0) \cup \bar{\Omega}(R_0)$ must have zero mean and maximum variance by design, however, there exists an $\epsilon > 0$ that guarantees a region

between the two caps is uncovered with zero density. As a result, all uniform distributions on the hypersphere maximize the Euclidean variance, however not all maximum variance configurations are uniform. Given this conclusion, it is observed that the concentration diversity loss is a relaxed constraint on uniformity. It has infinitely unstable saddle points that correspond to maximum variance but non-uniform configurations on the hypersphere.

We redefine the variance subject to the angular metric and propose the pairwise angular distance loss. Since the separation of angles is a function of the number of elements on the hypersphere, we cannot leverage the presence of an upper bound, as we did for the Euclidean formulation. The angular variance penalty is maximized with no restrictions for samples distributed on the unit hypersphere as:

$$\mathcal{L}_{\mathsf{v},\theta}(h) = - \mathop{\mathbf{E}}_{(\mathbf{x},\mathbf{y})\sim p_{\text{neg}}} [\cos^{-1}\left(\frac{\langle h(\mathbf{x}), h(\mathbf{y})\rangle_2}{||h(\mathbf{x})||_2||h(\mathbf{y})||_2)}\right)^2]. \tag{4.24}$$

In order to use the angular or Euclidean metric for the maximal variance problem, it is imperative to understand the learning dynamics subject to its gradients. By definition, the optimization procedure aims to compare pairwise distances. If the goal is to spread out elements on the hypersphere, it is essential that regions of high density are penalized to a greater extent. Using the results presented in section 4.2.1, it is clear that using both angular and Euclidean metrics have gradients that vanish as elements approach one another, which is contrary to the diversity objective as a whole. In the following section, we modify the task in order to circumvent the problem of vanishing gradients.

4.4.2 Pairwise Metric Optimizations

We observe that all pairwise distances have eventually decreasing gradients as the elements approach one another. If we wish to maximize pairwise distances as a diversity penalty, a loss function must be constructed to avoid placing emphasis on pairs of elements that are already sufficiently far apart. Since we observe many points in high dimensional space with hidden coupling mechanisms, we cannot assume that a perfect packing

is possible. We mitigate both issues by generating a minimum margin penalty based on a distance threshold γ which is strictly decreasing and does not vanish in regimes where elements are close to one another. We introduce the minimum margin loss given a metric d as:

$$\mathcal{L}_{\mathsf{m}}(h; d, \gamma, p) = \mathop{\mathbf{E}}_{(\mathbf{x},\mathbf{y})\sim p_{\mathsf{neg}}}[(\min(0, \gamma - d(h(\mathbf{x}), h(\mathbf{y})))^p], \qquad p > 0. \tag{4.25}$$

In order to be flexible, there are no restrictions on the margin. In the case where the margin is set to zero, we aim to solve an over-regularized problem where stationary solutions can only be reached if all elements are perfectly packed.

Another means of mitigating issues related to vanishing gradients in regimes with high element density is through the use of the LSE as the smooth minimum estimator. We can unroll the optimization procedure from the global perspective of the system or from a local perspective for each element. When applied as a global operation over all pairwise combinations from the negative distribution we aim to minimize the dominant modes of the distribution where major densities bias the operation. When applied on the per sample basis, we aim to minimize the regional modes of the system, where we analyze each element in terms of its nearest neighbours. We define the global minimum and local minimum diversity loss subject to a choice of metric d and temperature T:

$$\mathcal{L}_{\mathsf{g,m}}(h; d, t, p) = \log \mathop{\mathbf{E}}_{(\mathbf{x},\mathbf{y})\sim p_{\mathsf{neg}}}[\exp(-td(h(\mathbf{x}), h(\mathbf{y}))^p)], \qquad p, t > 0. \tag{4.26}$$

$$\mathcal{L}_{\mathsf{l,m}}(h; d, t, p) = \mathop{\mathbf{E}}_{\mathbf{x}\sim p_{\mathsf{neg}}}[\log \mathop{\mathbf{E}}_{\mathbf{y}\sim p_{\mathsf{neg}}}[\exp(-td(h(\mathbf{x}), h(\mathbf{y}))^p)]], \qquad p, t > 0. \tag{4.27}$$

The behavior of each loss function is best analyzed from the NNC perspective. The global optimization procedure penalizes subsections of the system with the highest density. If there exist negatively coupled elements that cannot be separated, the global optimization will bias the gradients of the entire system to inseparable pairs. On the contrary, it is observed that the local optimization is density insensitive, since it is analogous to maximizing the distances between elements and their neighbours. This process shares the contrastive cost over all pairs and is therefore not biased by the whole system, only its local neighbourhood. As a result, it is clear that the local formulation is more robust to

system-wide NNC. Both methods circumvent PPC as a whole since there is no need to normalize any components of the algorithm.

Given the latter formulation, we note that the local model is identical to DCL and has the same properties which arise when analyzing the problem with infinitely many samples KDE. We also note that this formulation is related to the k-energy problem given a Gaussian kernel which is explored in section 4.5.1

4.4.3 Orthogonal Systems

It is also convenient to model the problem of learning invariant embeddings by imposing stronger constraints on global properties of the system that still involve contrasting elements on the unit hypersphere. We study the expected behavior of a system given the case where we observe many samples on $\mathbb{S}^n$. Let $\mathcal{B}(\mathbb{S}^n)$ be the set of all Borel probability measures on the hypersphere. We assume that for the uniform Borel probability measure ν^*, the expected pairwise inner products I_{ν^*} over the entire space converges to zero. This implies that on average, elements are orthogonal to one another [51]. We define I_ν as:

$$I_\nu = \int_{\mathbb{S}^n} \int_{\mathbb{S}^n} \langle \mathbf{x}, \mathbf{y} \rangle_2 d\nu(\mathbf{x}) d\nu(\mathbf{y}) = \mathbf{E}_{(\mathbf{x},\mathbf{y})}[\langle \mathbf{x}, \mathbf{y} \rangle_2]. \tag{4.28}$$

Where $I_\nu \to 0$ as $\nu \to \nu^*$. We show that sets with well distributed points in accordance to the uniform probability measure are orthogonal in expectation, knowing that the Euclidean mean $\bar{\boldsymbol{\mu}} = \mathbf{0}$ for any uniform configuration of elements. For ν^*:

$$\mathbf{E}_{(\mathbf{x},\mathbf{y})}[\langle \mathbf{x}, \mathbf{y} \rangle_2] = \langle \mathbf{E}_{\mathbf{x}}[\mathbf{x}], \mathbf{E}_{\mathbf{y}}[\mathbf{y}] \rangle_2 = \langle \bar{\boldsymbol{\mu}}, \bar{\boldsymbol{\mu}} \rangle_2 = ||\bar{\boldsymbol{\mu}}||_2^2 = 0 \iff \bar{\boldsymbol{\mu}} = \mathbf{0}. \tag{4.29}$$

We require a loss function that enforces orthogonality given infinite elements on $\mathbb{S}^n$ but is also robust to the flaws of the variance optimization objective. We can construct such a loss by also ensuring sufficient separation between all the pairs. Given the positive definite properties of the variance, we define a loss function which imposes an orthogonal

and pairwise separation constraint as:

$$\begin{aligned} \mathbf{E}_{(\mathbf{x},\mathbf{y})}[\langle \mathbf{x}, \mathbf{y} \rangle_2^2] - \mathbf{E}_{(\mathbf{x},\mathbf{y})}[\langle \mathbf{x}, \mathbf{y} \rangle_2]^2 &\geq 0 \\ \mathbf{E}_{(\mathbf{x},\mathbf{y})}[\langle \mathbf{x}, \mathbf{y} \rangle_2^2] &\geq \mathbf{E}_{(\mathbf{x},\mathbf{y})}[\langle \mathbf{x}, \mathbf{y} \rangle_2]^2 \\ \underbrace{\sqrt{\mathbf{E}_{(\mathbf{x},\mathbf{y})}[\langle \mathbf{x}, \mathbf{y} \rangle_2^2]}}_{\text{pairwise separation}} &\geq \underbrace{|\, \mathbf{E}_{(\mathbf{x},\mathbf{y})}[\langle \mathbf{x}, \mathbf{y} \rangle_2]|}_{\text{orthogonal error}}. \end{aligned} \tag{4.30}$$

Therefore, we may use the per-sample mean squared penalty to construct an orthogonality loss since it is an upper bound on the absolute orthogonality error. It is a stronger constraint that also aims to separate values that are similar. Using this result, we construct the Euclidean and angular orthogonality losses over the set of negatives as:

$$\mathcal{L}_{\mathsf{o},\mathsf{e}}(h) = \mathbf{E}_{(\mathbf{x},\mathbf{y})\sim p_{\mathsf{neg}}}[\langle h(\mathbf{x}), h(\mathbf{y}) \rangle_2^2]. \tag{4.31}$$

$$\mathcal{L}_{\mathsf{o},\mathsf{s}}(h) = \mathbf{E}_{(\mathbf{x},\mathbf{y})\sim p_{\mathsf{neg}}}[(\frac{R\pi}{2} - d_{\mathbb{S}_R^n}(h(\mathbf{x}), h(\mathbf{y})))^2]. \tag{4.32}$$

4.5 Optimizing Potentials on the Hypersphere

We have already demonstrated that diverse sets of elements on the hypersphere can be achieved by separating pairs based on their distances. Elements that are uniformly distributed have also been shown to have maximum entropy. These techniques have direct implications of the types of features that can be learned in a contrastive self-supervised setting when used as a diversity regularizer. An alternative way to enforce the diversity objective is by directly optimizing for uniformity on the hypersphere. Uniformity can be achieved using k-energy defined in section 3.4 which is equivalent to finding the best packing configurations of elements on $\mathbb{S}^n$. Best packing in terms of k-energy is defined as the problem of minimizing pairwise potentials. We are able to formulate a variety of optimization procedures and properties based on the selection of energy kernel. We also investigate the problem by considering the learning dynamics from a global perspective on the entire set of observed elements and then from a local perspective which measures the dynamics of an element as a function of all its neighbours.

4.5.1 Minimum Hyperspherical Energy

Minimum hyperspherical energy (MHE) is concerned with finding a configuration of points that minimizes the expected pairwise potentials on $\mathbb{S}_R^n$. MHE has been studied at length when looking to find configurations of repelling charged particles and is often referred to as Whyte's problem for the Riesz 0-potential, Thomson's problem for the Riesz 1-potential, and Tammes problem for the Riesz ∞-potential [52, 53].

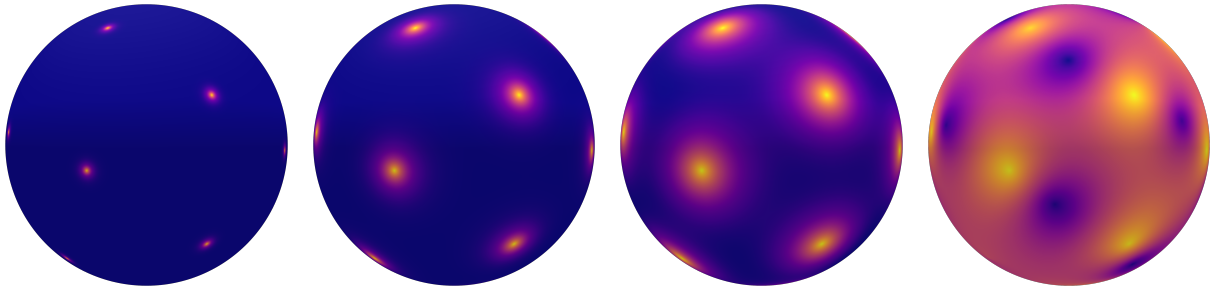

Figure 4.1: The relative magnitude of a potential field induced by a set of particles on the sphere. Regions in blue measure a smaller potential field than those in yellow. The amount of interaction between particles is dependent on the kernel parameters or bandwidth that define the range of the field about each particle. Neighbouring particle interactions parameterized with smaller bandwidths (leftmost sphere) are less than those larger bandwidths (rightmost sphere).

Given the set of all Borel measures $\mathcal{B}(\mathbb{S}_R^n)$ supported on the hypersphere, there exists a measure $\nu^* \in \mathcal{B}(\mathbb{S}_R^n)$ which minimizes its k-energy corresponding to the uniform measure given a kernel $K_{\boldsymbol{\sigma}}$ with parameters $\boldsymbol{\sigma}$ [11, 43]:

$$\nu^* = \underset{\nu \in \mathcal{B}(\mathbb{S}_R^n)}{\operatorname{argmin}} \int_{\mathbb{S}_R^n} \int_{\mathbb{S}_R^n} K_{\boldsymbol{\sigma}}(\mathbf{x}, \mathbf{y}) d\nu(\mathbf{x}) d\nu(\mathbf{y}) = \mathop{\mathbf{E}}_{\mathbf{x},\mathbf{y}} [K_{\boldsymbol{\sigma}}(\mathbf{x}, \mathbf{y})]. \tag{4.33}$$

We can therefore learn the configuration of elements which minimize the k-energy as a means of enforcing uniformity on the hypersphere. k-energy is defined using universal kernels or Riesz's kernel, called the s-potential. We solve the minimization problem over the expected pairwise potentials in the finite data regime given sufficiently many independent samples $\{\mathbf{x}\}_i^N$ embedded in $\mathbb{S}^n$. If elements do not share hidden latent factors, it is possible to find a reasonable packing of embeddings uniformly distributed over a subspace of the hypersphere using gradient descent [11, 43, 44]. We define the average

pairwise potential U of the system as:

$$U(h) = \frac{1}{N(N-1)} \sum_{\substack{i,j \\ i \neq j}}^{N} K_{\boldsymbol{\sigma}}(h(\mathbf{x}_i), h(\mathbf{x}_j)). \tag{4.34}$$

It is also possible to model the system potential using a logarithmic representation of its energy instead. We denote the logarithmic energy as a special case of the s-potential when $s = 0$. We define the 0-potential for an element pair $\forall \mathbf{x}, \mathbf{y} \in \mathbb{S}^n_R$ given a distance function d_s as:

$$K_s(\mathbf{x}, \mathbf{y}) = -\log d_s(\mathbf{x}, \mathbf{y}). \tag{4.35}$$

The total logarithmic system potential $U_{\log}$ is therefore defined in terms of a product of the pairwise potentials observed in the embedding space generated from $\{\mathbf{x}\}_i^N$ samples as [44, 51, 54]:

$$U_{\log}(h) = \sum_{\substack{i,j \\ i > j}}^{N} \log d_s(\mathbf{x}_i, \mathbf{x}_j)^{-1} = \log \prod_{\substack{i,j \\ i > j}}^{N} d_s(\mathbf{x}_i, \mathbf{x}_j)^{-1}. \tag{4.36}$$

Further work extended the notion of product configurations for higher order s-potentials. It is said that minimizing the logarithmic 0-potential is equivalent to solving a relaxed version of all higher order representations of itself [54]. This is demonstrated by the lower bound:

$$U_{\log}(h; s) = \sum_{\substack{i,j \\ i > j}}^{N} \log d_s(\mathbf{x}_i, \mathbf{x}_j)^{-s} = s U_{\log}(h), \qquad s > 0. \tag{4.37}$$

$$U_{\log}(h; 0) \leq U_{\log}(h; s), \qquad s > 0. \tag{4.38}$$

Moreover, it is possible to model the system in terms of its expected Dirichlet energy over the hypersphere. We note that a set of all elements moving along a field are stationary if the Dirichlet energy is zero. This can be illustrated using the local potential V_ν as:

$$\int_{\mathbb{S}^n_R} ||P_{\mathbf{x}} \nabla_{\mathbf{x}} V_\nu(\mathbf{x})||_2^2 d\nu(\mathbf{x}) = 0. \tag{4.39}$$

All the methods presented operate under the assumption that elements on the hypersphere are separable. Here, separability is defined by the ability to move elements apart

without any restrictions. This implies that there are no restrictions on the maximum distance between any two points. In practice, we are aiming to minimize pairwise potentials between image representations that have fixed and non-unique hidden class information that is shared across many samples within a dataset. As a result, the major assumption about separability cannot hold. It may be possible to separate certain sets of images that share minimal hidden class information, but there will always be some form of restrictions on subsets of the dataset. It is possible to eliminate certain kernels and methods knowing that the likelihood of sampling two images that have minimal distance in the hyperspherical embedding space is high. If two samples are too similar, methods that require s-potentials for $s > 0$ will break down, since they are unbounded and highly unstable in this regime. This phenomenon is also extended to the set of logarithmic product potentials since they are likely to vanish due to a singular bad pair of elements. Fortunately, Gaussian and inverse polynomial kernels have very desirable properties that account for problematic samples. These kernels are bounded and therefore avoid singularities found in s-potentials. Moreover, their gradients vanish for similar pairs when conditioning the kernel on any squared metric. Vanishing gradients allow the embedding space to freely group similar samples without any heavy penalties for similar samples.

If we naively operate under the conditions that all elements in the dataset are fully separable, it makes sense to minimize the logarithm of the pairwise potentials (not to be confused with the logarithmic potential). As per section 4.4.2, the logarithmic problem can be analyzed in terms of the entire joint system potential, or in terms of neighbourhoods contained in the system. This formulation assumes the elements are fully separable. We define the expected MHE, log-local MHE, and log-global MHE loss functions in accordance to the different unrolling strategies over sets of negatives given a k-energy kernel with parameters $\boldsymbol{\sigma}$:

$$\mathcal{L}_{\mathsf{mhe}}(h) = \mathbf{E}_{(\mathbf{x},\mathbf{y})\sim p_{\mathsf{neg}}}[K_{\boldsymbol{\sigma}}(h(\mathbf{x}), h(\mathbf{y}))]. \tag{4.40}$$

$$\mathcal{L}_{\mathsf{ll,mhe}}(h) = \mathbf{E}_{\mathbf{x}\sim p_{\mathsf{neg}}}[\log \mathbf{E}_{\mathbf{y}\sim p_{\mathsf{neg}}}[K_{\boldsymbol{\sigma}}(h(\mathbf{x}), h(\mathbf{y}))]]. \tag{4.41}$$

$$\mathcal{L}_{\mathsf{lg,mhe}}(h) = \log \mathbf{E}_{(\mathbf{x},\mathbf{y})\sim p_{\mathsf{neg}}}[K_{\boldsymbol{\sigma}}(h(\mathbf{x}), h(\mathbf{y}))]. \tag{4.42}$$

It is clear that the log-local is identical to the decoupled empirical entropy KDE loss introduced in section 4.3.2 and is equivalent to the method proposed by DCL[10, 11]. On the other hand, the log-global loss is the formulation introduced by Wang and Isola's uniformity decomposition [11]. Interestingly enough, all energy based models implicitly avoid PPC and do not have to rely on asymptotic properties of probabilistic methods in contrastive learning that rely on infinitely many samples.

Since it is highly likely that mini-batches contain samples with similar latent information content, we argue that the simple MHE loss should be used in place of the local-global counterparts due to its robustness to NNC. We show this result by analyzing the gradients for a single embedding $\mathbf{z}_i$ in a set of embeddings $\{\mathbf{z}_j\}_j^N \in \mathbb{S}^n$ between the MHE loss and the global logarithmic loss given a Gaussian kernel with scale parameter α with distance function d:

$$\overbrace{\underbrace{\frac{1}{\sum_{j,k,k>j}^{N} \exp(-\alpha d(\mathbf{z}_j, \mathbf{z}_k)^2)}}_{\text{global rescaling}} \nabla_{\mathbf{z}_i} \sum_{k}^{N} \exp(-\alpha d(\mathbf{z}_i, \mathbf{z}_k)^2)}^{\text{mhe}}. \tag{4.43}$$
$$\text{global mhe}$$

The additional rescaling term present in the global MHE acts as a coupling term that recalibrates the gradients as a function of the batch observed. If the cross batch statistics have high variance, the gradients between update steps for the global model are all treated with vastly different weights. This can be problematic depending on the data and the batch size selected. If samples in one batch are all relatively dissimilar, they will be rescaled and treated as important. If we eliminate this term and replace it with a constant, as per the MHE loss, it is possible to share the diversity cost fairly across update steps. The optimization benefits from this sharing procedure and reduces the effect of NNC since samples that cannot be separated but have large similarities do not dominate the signal from batch to batch. This is further reinforced by methods like MoCo and NNCLR. We argue that extending the number of samples in an offline manner reduces the variance of the rescaling term across batches and provides fairness amongst update steps. We implicitly avoid the additional infrastructure by avoiding the rescaling altogether. We further note that KDE is known to fail in high dimensional space. This is because the number

of samples required to estimate the distribution grows exponentially as the dimension increases, which supports the problem that penalizing entropy may be problematic.

A Note on s-Potential

Once again, we find a bridge between the s-potential and maximal variance on the hypersphere. It is observed that the selection of s determines the learning dynamics for the diversity objective. For the Euclidean or angular metric and $s = -2$, we recover the maximal variance objective which has been shown to be a suboptimal method for distributing points on the hypersphere. It is observed that minimizing the mean 2-potential is equivalent to maximizing the sample variance given a metric d [43, 51]:

$$U(h; 2) = -\mathbf{E}_{(\mathbf{x},\mathbf{y})\sim p_{\text{neg}}}[d(h(\mathbf{x}), h(\mathbf{y}))^2] = -\mathbf{V}_d(h) \tag{4.44}$$

4.6 Understanding Non-Spherical Optimizations

Many works on learning contrastive image representations have stated the importance and benefits of specifying the optimization problem in a normalized space that corresponds to the hypersphere [6]. In an unnormalized space such as $\mathbb{R}^n$, contrastive methods default to Euclidean optimization methods that rely on ℓ_2 distances or inner products. As seen in section 4.4, maximizing a posterior using a symmetric kernel on the hypersphere is analogous to maximizing entropy on the hypersphere. When moving off the hypersphere, maximizing the same kernel is equivalent to maximizing the entropy of a Gaussian distribution. Since the Gaussian distribution in $\mathbb{R}^n$ is unbounded, the learning procedure is subject to optimization instabilities depending on how it is specified. In this section, we aim to extend previously discussed losses without any normalization. We empirically find that not all methods are numerically stable and demonstrate that it is possible to stabilize an optimization procedure if the problem is well posed and an external regularization technique is used. For any kind of convergence, we require that all elements are bounded $\mathbb{R}^n \in B_R^n(\mathbf{0})$ where $R < \infty$ regardless of the number of optimization update steps taken.

We start by assuming that for any set of data whose embeddings are modeled in $\mathbb{R}^n$, there are no hidden coupling mechanisms between its elements. This implies that it is possible to extend the distance between any embedding pair without restrictions. As a result, gradients must eventually be decreasing for the embeddings to converge. Given a set of elements Z and loss function $\mathcal{L}$, we require the gradients of $\mathcal{L}$ to vanish for all boundary points contained in Z. We model an exterior regularizer Q and say an optimization converges after infinitely many update steps if there exists a configuration of elements where:

$$\sup_{\mathbf{z}_i \in Z} ||\nabla_{\mathbf{z}_i} \mathcal{L}||_2 < ||\nabla_{\mathbf{z}_i} Q||_2 \tag{4.45}$$

While we do not explicitly derive a lower bound on the gradients of the local uniformity objective, we note that for a fixed set of elements being pushed apart, the gradients for a specific pair that contains an element on boundary of the set increases proportionally to the distance between the closest interior point. Since there is no PPC, elements on the boundary conditions continue to be extended outwards in space along a direction that guarantees the radius of $B_R^n(\mathbf{0})$ will increase. We show this phenomenon using the gradients of a toy example with only two elements $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ subject to the local uniformity loss:

$$\nabla_{\mathbf{u}} \log(\exp(-||\mathbf{u} - \mathbf{v}||_2^2) + \exp(-||\mathbf{v} - \mathbf{u}||_2^2)) = -2(\mathbf{u} - \mathbf{v}). \tag{4.46}$$

If we keep $\mathbf{v}$ constant and construct a sequence that models the position of $\mathbf{u}$ moving along the direction opposite of is gradients, we observe that the position of $\mathbf{u}$ increases indefinitely and diverges. This can be remedied by imposing an external field on the entire embedding space that is greater than the outward force from the two elements pushing on each other. Since the gradients for this example are increasing, the external field placed on the space must be strong and will restrict the optimization in an undesirable way. As a result, we see that this class of optimization procedure is not well suited in an unbounded space.

Based on the latter observation, we can look at the problem specified using MHE since it does not contain the logarithmic transform that reweights the gradients of a batch. Here, we can derive an upper bound for the norm of the gradients of the diversity loss for the

set of elements Z. For simplicity, we chose to model the k-energy using the Gaussian kernel and show the upper bound as:

$$\begin{aligned}\Big|\Big|\nabla_z \sum_{\substack{i,j,\\ i\neq j}}^{|Z|} \exp(-||\mathbf{z}_i - \mathbf{z}_j||_2^2)\Big|\Big|_2 &= 2\Big|\Big| \sum_{\substack{i,j,\\ i\neq j}}^{|Z|} (\mathbf{z}_i - \mathbf{z}_j)\exp(-||\mathbf{z}_i - \mathbf{z}_j||_2^2)\Big|\Big|_2 \\ &\leq 2\sum_{\substack{i,j,\\ i\neq j}}^{|Z|} ||\mathbf{z}_i - \mathbf{z}_j||_2 \exp(-||\mathbf{z}_i - \mathbf{z}_j||_2^2) \\ &\leq \underbrace{\sup_{\mathbf{z}_i,\mathbf{z}_j\in Z} 2|Z|^2||\mathbf{z}_i - \mathbf{z}_j||_2 \exp(-||\mathbf{z}_i - \mathbf{z}_j||_2^2)}_{\text{decreasing}}.\end{aligned} \tag{4.47}$$

The above representation of the MHE diversity loss has eventually decreasing gradients. It is clear that these gradients diverge, however, the rate of divergence is slow. In practice, these gradients diverge so slowly that we do not need to enforce any kind of additional regularization. In theory, any non-decreasing regularizer will guarantee a convergent configuration of elements in the closed system. For example, regularizers like weight decay is a sufficient condition for convergence. In the context of contrastive learning, we also note that the hidden coupling mechanism implicitly regularizes the space since not all elements can be easily pushed apart.

We propose a regularization framework beyond weight decay to further constrain the system and control the tightness of elements in a space. We model the regularizer using an external field. We denote the global field as Q_{g} and a local field as Q_{l}. The local and global constrained diversity objectives are therefore restated using the same kernel energy model $K_{\boldsymbol{\sigma}}$ as:

$$U_{\mathsf{c},\mathsf{l}}(h) = \mathbf{E}_{(\mathbf{x},\mathbf{y})\sim p_{\mathsf{neg}}}[K_{\boldsymbol{\sigma}}(h(\mathbf{y}), h(\mathbf{x})) + Q_{\mathsf{g}}(h(\mathbf{x}), h(\mathbf{y}))]. \tag{4.48}$$

$$U_{\mathsf{c},\mathsf{g}}(h) = \mathbf{E}_{(\mathbf{x},\mathbf{y})\sim p_{\mathsf{neg}}}[K_{\boldsymbol{\sigma}}(h(\mathbf{x}), h(\mathbf{y})) + Q_{\mathsf{l}}(h(\mathbf{x})) + Q_{\mathsf{l}}(h(\mathbf{y}))]. \tag{4.49}$$

An example of a local field Q_{l} is an embedding norm penalty that is analogous to a gravitational force towards the origin along the radial direction. A local field can be interpreted in terms of an implicit element coupling model similar to the Mie potential. The Mie po-

tential models elements using soft repulsive and attractive forces. The repulsive forces are fast decreasing while the attractive force is slow decreasing with gradients in opposing directions. We chose to model a representation of the Mie potential using a mixture of two energy kernels with different parameters. Let $\boldsymbol{\sigma}_{\mathsf{a}}, \boldsymbol{\sigma}_{\mathsf{r}}$ denote the attractive and repulsive kernel parameters with weighting term β. We define the diversity objective in terms of the kernel Mie potential as:

$$\mathcal{L}_{\mathsf{mie}}(h; \beta) = \mathbf{E}_{(\mathbf{x},\mathbf{y})\sim p_{\mathsf{neg}}}[K_{\boldsymbol{\sigma}_{\mathsf{r}}}(h(\mathbf{x}), h(\mathbf{y})) - \beta K_{\boldsymbol{\sigma}_{\mathsf{a}}}(h(\mathbf{x}), h(\mathbf{y}))]. \tag{4.50}$$

As per the regular k-energy problem, we note that Riesz kernels should be avoided. We can also further tune how close elements can get to one another using a hinge loss on top of a kernel energy. The addition of the hinge term also guarantees that all separable points have a limit to how far they can be pushed apart. We define the hinge potential for the diversity objective with hinge threshold γ as:

$$U_{\mathsf{h}}(h, \gamma) = \mathbf{E}_{(\mathbf{x},\mathbf{y})\sim p_{\mathsf{neg}}}[\min(\gamma, G(h(\mathbf{x}), h(\mathbf{y})))] \tag{4.51}$$

Before moving on, it is important that we also understand how SimCLR operates in an unbounded space. It is observed that inner products are computed rather than distance functions. As a result, the problem is being treated in terms of a linear classification problem, where one sample is used to define a decision boundary when compared to another. We claim that this is not well specified for the goal of bringing elements together. This abstraction removes our ability to understand the problem in terms of hyperspherical energy as well. We also theorize that these decision boundaries learn to encode information into the lengths of the embeddings which may be problematic.

The main contribution of this section is with respect to coupled Mie potential model. We have demonstrated that it is much more stable to optimize than the other types of methods in an unnormalized space due to its decreasing gradients. It is well posed in terms of how it compares and contrasts embeddings. Another major benefit of this method is that we can leverage variance reduction tools to learn smoother solutions because we are

in $\mathbb{R}^n$. The cost of learning pairwise invariance of the set of all possible perturbations is usually amortized over the entirety of model training. Each sample is likely to encounter sufficiently many types of perturbations over time, however, this pairwise dependence is sensitive to the noise with respect to the augmentation distribution. Since samples are now generated in $\mathbb{R}^n$, there is a smooth, unbiased variance estimator that can be used to contrast negative pairs. We assume that all perturbed samples belong to a Gaussian distribution centered about a sample mean. We restate the entire contrastive learning objective in terms of a contrastive clustering problem over a set of many positives. We can sample a set of N views and minimize the variance over the pairwise views for each sample. We may then compute the diversity penalty using the means estimated in the previous step.

4.7 Summary

In this chapter, we presented different methods and their designs for how to learn in hyperspherical space using an invariant alignment and diversity objectives. We pose the problem of learning on the hypersphere by modeling it in terms of fixed parameter distributions, orthogonality, and potential energy and relate the perspective back to known work where applicable. We analyze the asymptotic properties of each method and break down where they struggle from the perspective of hidden coupling mechanisms. We give insights on how each method compares and contrasts elements on the hypersphere using different properties of the LSE operation, and propose an alternative minimal hyperspherical energy formulation that aims to circumvent these issues. Using the knowledge amassed from our analysis on hyperspherical space, we proposed an unnormalized contrastive learning optimization protocol in terms of a coupled pair potential that circumvented many issues present when modeling the diversity objective off the hypersphere. In the following chapter, present the methodology required to evaluate each method and present our findings.

5

Evaluation

In the following chapter, we aim to verify the efficacy of pre-training a neural network in a contrastive learning framework on a set of unlabelled images using different specifications of the invariance and diversity objectives presented throughout the thesis. The performance of each method is evaluated based on the linear separability between hidden class labels that are not made available during training. We explore the impacts associated with modeling the problem on different representation spaces based on the projective maps to $\mathbb{S}, \mathbb{D}$ or $\mathbb{R}$. Moreover, we investigate the impacts of selecting a Euclidean or angular metric on $\mathbb{S}$ and quantify the benefits of removing positive-positive coupling and reducing the net impact of negative-negative coupling.

We introduce a standardized methodology and framework across all experiments. Results are compared and contrasted by their **relative performance** to each other based on

empirical evidence found through experimentation. Linear separability of the features learned during training is quantified by finetuning a linear classifier on top of the features produced by the pre-trained neural network using the same set of labelled training images. Performance is evaluated on a fixed holdout test set and accuracy is reported between each experiment. Results demonstrate that stereographic spaces are too restrictive, decoupled methods outperform their coupled equivalents, and certain energy models specified in $\mathbb{R}$ can be as viable of a solution as any other method.

5.1 Evaluating Learned Features

The goal of any self-supervised algorithm is to learn features without access to labels. These features should have utility on a variety of downstream tasks. Downstream tasks common to the field of computer vision include, but are not limited to: classification, regression, segmentation, detection. In this thesis, we focus on assessing feature quality as a function of the classification task. We evaluate features learned from a self-supervised pretext task by probing the neural network with a linear classifier head trained on top of the features produced by the pre-trained network. This process allows the practitioner to evaluate the linear separability of samples using linear decision boundaries in an embedding space. We use the classifier's accuracy on an unseen test set to quantitatively evaluate the strength of the upstream self-supervised algorithm.

5.1.1 Dataset

All models are trained and evaluated on the public dataset CIFAR-10 [55]. CIFAR-10 is composed of small 32×32 Red-Green-Blue images encoded as 3-tuples, each with pixel values ranging from 0 to 255 inclusive. There are 10 evenly divided classes available in the dataset. These classes include airplanes, automobiles, birds, cats, deer, dogs, frogs, horses, ships, and trucks. The dataset contains $60,000$ images which are divided into a training set containing $50,000$ images and an evaluation set with the remaining $10,000$ images. CIFAR-10 is a well-studied dataset and is ideal for the exploratory work done

throughout this thesis.

5.1.2 Augmentations

The driving force for learning invariant features is the selection of augmentations. The set of augmentations and their relative parameters dictate the type of downstream task a practitioner can attempt to solve. Different augmentations have different resulting effects on the features learned. In the case of small image classification, we adopt four core augmentations. One augmentation of use is the random crop. The random crop allows a neural network to learn substructure consistencies across different images, even when they are not explicitly compared. Images belonging to identical classes typically have similar substructures. For example, all cats have fur, and thus taking patches of fur forces the network to learn textural relationships. Learning from substructures like texture minimize biases from positional information that may exist in the encoded image. Moreover, since CIFAR-10 contains small images without immense detail, we must introduce color distortions to avoid strictly learning weak features due to the presence of class-correlated color variables. Color distortions are introduced in the form of a grayscale transformation, or color jitter which jointly perturbs the brightness, contrast, saturation, and hue of an image. We also apply random horizontal flips to minimize the presence of positional pose information. While there are many more important augmentations that should be leveraged for larger images (solarization, blur), however, those listed above are sufficient and well explored in many related works [6, 7].

5.1.3 Experimental Framework and Configuration

As an overview, the experiment pipeline is divided into the training and evaluation protocols. The training protocol is standardized to use a backbone-projector architecture. The training protocol is decomposed in accordance to the invariance pretext task. We sample a mini-batch from the CIFAR-10 training dataset and randomly apply a set of perturbations depending on the number of required views. All perturbed images are normalized using a whitening operation whose values are precomputed on CIFAR-10 before train-

ing. We encode the perturbed and normalized samples using the joint siamese network and embed them into a target manifold using a choice of projection map (closest point, stereographic, identity). We show an example of the siamese architecture in figure 2.3. Once encoded, samples are compared using a selection of similarity and diversity loss functions. The loss function compares the encoded outputs at a given step. The gradients of the loss are used to update the networks using a backpropagation based optimizer. The process is repeated for a fixed number of epochs.

The linear evaluation protocol aims to evaluate the robustness of the backbone neural network. The projector is tossed aside and the weights on the backbone are frozen so that they cannot be updated during a secondary training phase required by the evaluation protocol. A new linear classifier is trained on top of the frozen backbone using the same training data available during the training phase. We apply the same set of training augmentations and normalization to the input samples and tune the classifier for a set number of epochs using a supervised cross entropy loss with the ground truth labels. Once completed, the backbone and classifier are evaluated on the test portion of the CIFAR-10 dataset. Top-1 accuracy is reported and used as a metric for how well the self-supervised features are able to transfer to a classification task.

Train time augmentations for CIFAR-10 include a random resize crop, color distortion, grayscale, and horizontal flips. Random crops are applied to all images. They are produced using by cropping an edge of an image using minimum and maximum scale in range $(0.2, 1)$. The image crop is then resampled to its original size. Color distortion is applied in the form of color jitter with a probability of 0.8. Color jitter modifies brightness, contrast, saturation, and hue altogether. The scale factors selected for each value is $[0.4, 0.4, 0.4, 0.1]$ respectively as per SimCLR [6]. These factors are uniformly sampled as well before being applied to an image. Horizontal flips are applied with a probability of 0.5 and images are converted to grayscale with a probability of 0.2. Both train and test time augmentations apply a whitening transformation with mean along the color channels of $[0.4914, 0.4822, 0.4465]$ and standard deviation of $[0.2023, 0.1994, 0.2010]$.

In all experiments, the backbone is selected to be a ResNet18 which is a specific type of

CNN architecture [56]. It is modified to have its final fully connected layer removed. The projector is selected to be a two layer MLP with batch normalization and a rectified linear unit as its non-linear activation. The output dimension of the backbone is $\mathbb{R}^{512}$ and the output dimension of the projector is $\mathbb{R}^{128}$. These networks are optimized using the Adam optimizer for a fixed number of epochs. Adam is initialized with a fixed learning rate of $1e^{-3}$ and weight decay regularization of $1e^{-5}$. The networks are trained for a total of 500 epochs with 2 views, each with a batch size of 512. Here, we sample a single batch of 512 samples, augment them two times to generate the set of views, and process the entire set of 1024 samples. The training portion of the evaluation protocol uses the same optimizer and batch size and is trained for a total of 100 epochs. Experiments are processed using an automatic mixed precision training protocol that mixes float32 and float16 numeric precision to minimize computational memory and speed bottlenecks.

5.2 Results

We present the results for a set of experiments in table 5.1. All experiments are run using the same sets of random seeds, and results presented are the best results across multiple runs. For each experiment, the best hyperparameter setting is presented. We specify the method as well as the distance or similarity function used to compare samples embedded into the space. For simplicity, we define angular metrics using θ, Euclidean metrics using ℓ_2 and similarity using s for the inner product. Experiments that did not converge during training are denoted as DNC. We also define experiments over a mixture of spaces. We denote the closest point experiments using $\mathbb{S}$, stereographic experiments using $\mathbb{D}$, and Euclidean experiments as $\mathbb{R}$.

We compare contrastive self-supervised methods outlined in the thesis against one another and to the fully supervised example. We present the maximum variance based diversity experiment equivalent to the concentration and 2-potential problem and explore the effects of three different kernels for the matching loss. The kernels explored are the Gaussian, Laplacian, and polynomial kernels. We note that the Gaussian and polynomial kernels reduce to the vMF and PS representation of the problem using Euclidean distance,

and is thus the same as using a similarity function on the hypersphere. We call the set of experiments exploring orthogonality as orthogonal. We also explore a maximum cutoff threshold seen in equation 4.25 which we define as the dissimilar experiment. We denote experiments that maximize uniformity using log potentials from the local and global viewpoints as the global uniform and local uniform experiments. We present the general potential experiments without any logarithmic terms as the energy or pair energy for the Mie potential formulation. These results are presented in table 5.1 found below.

Name	Space	Comparison	Accuracy (%)
Supervised	$\mathbb{R}$	s	93.23
Variance	$\mathbb{S}$	ℓ_2	72.12
Match Polynomial	$\mathbb{S}$	s	88.83
	$\mathbb{S}$	s	88.81
	$\mathbb{D}$	θ	85.92
Match Gaussian	$\mathbb{S}$	s	88.81
	$\mathbb{S}$	θ	88.90
	$\mathbb{D}$	s	85.89
	$\mathbb{R}$	s	78.65
Match Laplacian	$\mathbb{S}$	ℓ_2	88.92
	$\mathbb{S}$	θ	88.96
Orthogonal	$\mathbb{S}$	s	89.26
	$\mathbb{S}$	θ	88.85
Dissimilar	$\mathbb{S}$	s	89.12
	$\mathbb{D}$	θ	88.56
Global Uniform	$\mathbb{S}$	ℓ_2	89.10
	$\mathbb{S}$	θ	89.11
	$\mathbb{R}$	ℓ_2	68.02
Local Uniform	$\mathbb{S}$	ℓ_2	89.29
	$\mathbb{S}$	θ	89.37
	$\mathbb{R}$	ℓ_2	DNC
Energy	$\mathbb{S}$	ℓ_2	89.19
	$\mathbb{S}$	θ	89.26
Pair Energy	$\mathbb{R}$	ℓ_2	89.49

Table 5.1: Linear evaluation results on CIFAR-10. Top-1 accuracy reported as the comparative metric, depending on the embedding space and similarity or distance function.

As seen in the table 5.1, the majority of the experiments achieve competitive classification performance that closely approach the fully supervised baseline. There are a few outliers in the table that should be noted. In particular, all stereographic experiments

underperform by a significant margin when compared to the same experiments that use closest-point projection instead. Moreover, pure variance regularization was found to be inefficient, and produced results that are in excess of 15% lower to most other forms of diversity regularization. It is noted that all experiments that circumvented PPC have marginal but statistically significant improvements over their coupled counterpart. These experiments include the uniformity, energy, and orthogonality models. We observe that for contrastive methods in $\mathbb{R}$, the SimCLR-like set up using a Gaussian kernel performed poorly, and decoupled methods on $\mathbb{R}$ did even worse. We also note that using pair energy in $\mathbb{R}$ was amongst the most competitive methods observed and contradicts the claim that contrastive representations in a unnormalized space underperform. We discuss in detail the implications of the results seen above in the section below.

5.3 Discussion

There are a few significant results presented in table 5.1 that confirm much of the theoretical statements made throughout the thesis. We first address the performance related to the stereographic models $\mathbb{D}$. As a whole, it is possible to learn good representations in a stereographic space with an approximate classification accuracy of 85.90% for the polynomial and Gaussian kernel matching objective. While this score is good, it is clear that imposing some implicit constraint on the curvature of a neural network's output limits how expressive the learned set of features can be. This is supported by an absolute drop in performance of 3% when compared to the closest point projection model with the identical kernels. We theorize that this limitation is an artifact of how the neural network groups points in the ambient space. Elements that are far away from the origin are close together in the backprojected hyperspherical space. This should not be a problem if we modify the linear classifier to account for the geometry, however, any kind of weight decay will limit the maximum distance of an embedding measured from the origin and will make the problem of aligning points impossible on the upper half space of the hypersphere. The stereographic space also requires that elements on the upper half space be extremely far away from the origin which is not numerically stable. Due to these issues, we conclude

that the stereographic representation of data is not worth the additional computational burden and provides no added benefits.

When looking at the results for the matching objective in hyperspherical space $\mathbb{S}$, kernels built on top of quadratic measures of distance (Gaussian, polynomial) performed marginally worse (0.10%) than those with linear measures of distance (Laplacian). It is theorized that quadratic dependencies decay at a different rate which leads to different separation between elements computed in the LSE as part of the diversity objective. By changing the rate, neighbours that are slightly farther or closer away play a different role in the optimization. These methods were also shown to have PPC and it is clear that all objectives that avoid PPC performed better. All models containing PPC on the hypersphere are found to perform under 89.00%, while the vast majority of decoupled experiments on the hypersphere performed above this margin.

There are two experiment categories that avoid PPC on the hypersphere, but do not always perform as expected. By definition, the dissimilar and orthogonal experiments remove any presence of PPC since the losses are only computed between elements that are randomly sampled from the dataset. It is observed that the methods which aim to make samples orthogonal or dissimilar perform better with the Euclidean inner product rather than with the angular metric. Referring back to equation 4.4, it is clear that Euclidean inner products produce gradients that decrease as elements on the hypersphere approach each other. What this implies is that even with a squared loss, samples that are similar to one another are not pulled apart as much as samples that are farther apart, as decided by sine dependency. The sine dependency introduced a softness in the gradients which is more conducive to the downstream classification task. It is beneficial because it is possible to gather samples with similar hidden latent factors in clusters without the penalty over imposing separation of elements in these regions. This phenomenon holds for both kinds of losses, however, the dissimilarity objective amplifies the gradients more than the orthogonality constraint. Moreover, it is not possible to make every vector perfectly dissimilar in hyperspherical space and as a result, the optimization procedure penalizes pairs of elements which may already be in an optimal position. Optimizing for orthogo-

nality and dissimilarity with the Euclidean inner product yielded a classification accuracy of 89.26% and 89.12% respectively. These results are competitive with the best methods presented and are above the relative threshold of 89.00% seen to divide experiments with and without PPC. We note that orthogonality outperformed dissimilarity due to the over-constraint optimization specification in the dissimilarity experiment. As for the angular variant of the two objectives, referring back to equation 4.6, it is observed that gradients are maximal for pairs of elements that are most similar. In theory, this is a desirable property only if the samples being compared have minimal shared latent factors. In practice, depending on the dataset and the batch size, there can be a high likelihood of elements belonging to the same class distributions being compared. This redundancy between classes is what introduces the NNC that is overly penalized by the angular metric. In the case of CIFAR-10, this is guaranteed with a batch size of 512. As a result, both orthogonality and dissimilarity objectives are likely to be over constrained. As a result, it is more challenging for the neural network to form proper clusters of data. This phenomenon is seen in the results, where both experiments with the angular metric had downstream classification accuracy of 88.85% and 88.56. These results are well below the Euclidean formulation with a drop in performance of approximately 0.50%. Moreover, these methods performed worse than the matching objectives despite having avoided PPC on the hypersphere.

The second set of experiments that avoid PPC on the hypersphere are the *logarithm* based global and local uniformity experiments. It is noted that these experiments follow the same global and local definition as outlined in the pairwise distance section 4.4.2. In particular, we note that the local uniform objective is identical to DCL [10] and the KDE empirical entropy penalty [11] presented in section 4.3.2. The global uniformity objective is by definition the method of alignment and uniformity [11]. Both methods performed better than the matching models. The local objective accuracy is reported to be 89.29% and 89.37% while the global objective was found to be 89.10% and 89.11% across Euclidean and angular metrics. Based on these results, it is seen that there exists a clear advantage of removing PPC. The removal of PPC is observed to benefit these algorithms by approximately 0.30% compared to their coupled counterpart. In both of these experiments, it is hard to distinguish between the impacts associated with the choice of metric

for the global objective. On the other hand, the angular distance produced a boost in performance of 0.08% for the local objective. Furthermore, we note that the local formulation outperformed the global objective by approximately 0.20%. We assume this change in performance is due to the local objective's robustness to NNC. In the local formulation, the gradients of a sample are dictated by its nearest neighbours, while the global representation of the problem penalizes the modes of the entire mini-batch. Since we run experiments with a relatively large batch size for CIFAR-10, we know that there will be many modes in the embedding space correlated to the different hidden class features learned throughout training. Because these correlated samples cannot be fully pulled apart, the gradient of the diversity penalty on important pairs that are not contained in the modes of the embedding space are dampened as a result of the LSE computation.

The final formulation of the problem on the hypersphere is proposed by our modification for how energy should be applied to samples. This formulation aims to mitigate the effects of global NNC by better sharing the total contrastive cost over the entire set of update steps. This method also avoids PPC and is directly compared to the global and local uniformity objectives previously analyzed. The minimum hyperspherical energy objective was found to work better than the global objective but worse than the local objective. Accuracy for the Euclidean and angular metrics are 89.19% and 89.26% respectively. We theorize that it performed better than the global objective by being slightly more robust to inseparable modes in the data by sharing the minimization cost over all samples without the coupled rescaling term. While this is beneficial, it is tough to find the constant recalibration term which enforces sufficient push and pull of elements in the space, which is where the local formulation seems to shine.

We note that out of all the methods specified on the hypersphere, those that only penalized the sample variance performed significantly worse than every other method at 72.12%. This is because the penalty does not directly enforce any element-element comparisons and is a weak pairwise regularizer. There exist too many poor and unstable configurations of elements that provide minimal benefit to the learning problem. Even with this kind of regularization, it is still too weak to avoid collapsing to a lower than usual

subspace. The major conclusion from running this experiment is that extreme diversity is a necessary condition for learning useful features.

Finally, we investigate the effects of learning invariance in $\mathbb{R}$ where there is no upper bound on the pairwise distances. We run the SimCLR equivalent experiment using unnormalized pairwise inner products to measure similarity between pairs of embeddings. We observe that this formulation of the invariance and diversity objective yields poor results with an accuracy of 78.65%. We then run the decoupled version of the experiments expressed by the global and local uniformity objectives with Euclidean distances rather than inner products. The PPC decoupling present in the global objective leads to drastic reduction in performance of 68.02% which is over 10% below the original coupled objective. What is even more surprising is that the local objective does not converge. The network's weights explode and become numerically unstable early into experiment training. We can understand the poor performance of the coupled inner product experiment from the perspective of energy. Elements are compared to one another using the equivalent of a Boltzmann distribution where one element is used as a linear decision boundary to classify the other. This is problematic because we are attempting to force elements to be close to one another, and not just reside in some subdivision of the embedding space. Moreover, the use of an inner product cannot be related to any kernel energy model. This is because the information of each embedding norm is not included. The impact of specifying the wrong metric means that the model can learn to encode similarities as a function of the embedding's magnitude. The normalization of the distribution introduces coupling that constricts these magnitudes however, we argue that the composition of the space which is not inline with our main objective. Once we remove the PPC contained in the normalization, the network can freely inflate the length of each embedding which explains why the local decoupled model could not converge. We theorize that the global method was more stable because the normalization is conditioned on the entire space which means the network has a harder time inflating the norm of the samples due to NNC. Since the introduction of the logarithm renormalizes the diversity statistics in a batch, it is able to push far away samples ever further if the space isn't sufficiently filled. These issues were present even with weight decay. Based on the ill-posed nature of the diversity problem,

it is possible to select a pair potential model which will have guarantees on convergence. We run the unbounded version of k-energy with a paired potential and find that the minimum energy model converges with a downstream classification accuracy of 89.49 which is an enormous 11% increase compared to the next best method in $\mathbb{R}$. Not only is this the best result across all unnormalized self-supervised experiments, it is competitive and outperforms the normalized counterparts contrary to initial claims made about the benefit of normalization [6]. We theorize that an unnormalized representation of the data can be more expressive since comparisons are made based on embedding distances rather than embedding angles. These distances can capture how close elements truly are to one another which may lead to stronger invariances to the set of augmentations.

While we have presented many interesting results about the relative performances of each method, it is important to comment on the limitations of the experimental framework. CIFAR-10 has little class diversity, and a batch size of 512 is considered to be large for this type of data. As a result, we are most likely operating in a regime where noisy contrastive estimates are not so noisy. We note that a lot of the behavior may change when operating with much smaller batch sizes. We also acknowledge that the removal of PPC plays a smaller than expected role due to the large amount of NNC that cannot be avoided with large batch sizes and a dataset with a small amount of classes. It is possible that the choice between the Euclidean and angular metric can play a larger role on datasets where a mini-batch contains minimal samples that all belong to the same class. If there is greater class diversity, the angular metric may be more conducive to finding larger margins between classes. Finally, like all deep learning algorithms, performance is highly dependent on the hyperparameters for the experiment. We did an extensive hyperparameter grid search, however, by no means can we guarantee certain results presented cannot be improved with further exploration. Based on the above factors, the main takeaways from this thesis is that there is a clear benefit to understanding what kind of coupling mechanisms are at play and it is possible to learn good contrastive image representations without explicit normalization.

5.4 Summary

In this chapter, we presented a framework to train and evaluate a neural network in self-supervised setting on a downstream classification task. We presented the experimental setup common to all methods and showed empirical results in the form of evaluation accuracy on a holdout test set. Results are presented across different spaces, distance functions, and objectives. We concluded that the stereographic space was not an effective means of representing data, strong diversity regularization is needed, and that experiments which avoided explicit positive-positive coupling outperformed methods that did not. We also showed that it is possible to effectively learn contrastive image representations off the hypersphere if we take a step back to ensure the problem is well specified.

6

Conclusion

In this thesis, we studied the problem of learning useful image representations without having access to labels at training time through contrastive learning in hyperspherical space. We first provided theory on how to define hyperspherical geometry and measure distances between elements on the hypersphere.

We introduced different ways of describing distributions and energy on the hypersphere and used these tools to decompose the self-supervised problem in terms of an invariance and diversity objective. We investigated how elements on the hypersphere interact with one another as part of the diversity objective and observed various sources of noise in the learning process. We categorized these sources of noise as a function of different kinds of hidden coupling mechanisms. We analyzed the asymptotic properties related to learning diverse sets of features and proposed a variation to current methods based

on the orthogonal properties present for elements that are uniformly distributed over the hypersphere. We then introduced a modification on the diversity objective using a non-logarithmic formulation of minimal hyperspherical energy and showed that many diversity methods can be recovered depending on the energy perspective taken. Using the knowledge gathered throughout the thesis, we deconstructed the problem of learning contrastive image representations off the hypersphere and present a set of requirements that should be used in this setting.

We outlined how to train and evaluate a neural network in a self-supervised framework and present our empirical findings. We demonstrate that there are minimal impacts associated to the choice of distance function on the hypersphere and conclude that stereographic representations are too restrictive for the problem. Moreover, we showed that avoiding certain sources of element coupling is beneficial and proved that is it possible to learn competitive representations off the hypersphere.

Bibliography

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–44, 05 2015.

[2] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," *CoRR*, vol. abs/1603.08511, 2016. [Online]. Available: http://arxiv.org/abs/1603.08511

[3] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," *CoRR*, vol. abs/1803.07728, 2018. [Online]. Available: http://arxiv.org/abs/1803.07728

[4] C. Doersch, A. Gupta, and A. A. Efros, "Unsupervised visual representation learning by context prediction," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1422–1430.

[5] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving jigsaw puzzles," *CoRR*, vol. abs/1603.09246, 2016. [Online]. Available: http://arxiv.org/abs/1603.09246

[6] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, "A simple framework for contrastive learning of visual representations," *CoRR*, vol. abs/2002.05709, 2020. [Online]. Available: https://arxiv.org/abs/2002.05709

[7] X. Chen and K. He, "Exploring simple siamese representation learning," *CoRR*, vol. abs/2011.10566, 2020. [Online]. Available: https://arxiv.org/abs/2011.10566

[8] K. He, H. Fan, Y. Wu, S. Xie, and R. B. Girshick, "Momentum contrast for unsupervised visual representation learning," *CoRR*, vol. abs/1911.05722, 2019. [Online]. Available: http://arxiv.org/abs/1911.05722

[9] D. Dwibedi, Y. Aytar, J. Tompson, P. Sermanet, and A. Zisserman, "With a little help from my friends: Nearest-neighbor contrastive learning of visual representations," *CoRR*, vol. abs/2104.14548, 2021. [Online]. Available: https://arxiv.org/abs/2104.14548

[10] C.-H. Yeh, C.-Y. Hong, Y.-C. Hsu, T.-L. Liu, Y. Chen, and Y. LeCun, "Decoupled contrastive learning," 2021. [Online]. Available: https://openreview.net/forum?id=JzdYX8uzT4W

[11] T. Wang and P. Isola, "Understanding contrastive representation learning through alignment and uniformity on the hypersphere," *CoRR*, vol. abs/2005.10242, 2020. [Online]. Available: https://arxiv.org/abs/2005.10242

[12] E. Kreyszig, *Introductory Functional Analysis with Applications*, ser. Wiley Classics Library. Wiley, 1991. [Online]. Available: https://books.google.ca/books?id=AQtMEAAAQBAJ

[13] J. M. Lee, *Introduction to Smooth Manifolds*. Springer New York, 2012. [Online]. Available: https://doi.org/10.1007/978-1-4419-9982-5

[14] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*. Cambridge University Press, Jun. 2004. [Online]. Available: https://doi.org/10.1017/cbo9780511809682

[15] C. A. Micchelli, Y. Xu, and H. Zhang, "Universal kernels," *J. Mach. Learn. Res.*, vol. 7, p. 2651–2667, dec 2006.

[16] X. Pennec, "Intrinsic statistics on riemannian manifolds: Basic tools for geometric measurements," *Journal of Mathematical Imaging and Vision*, vol. 25, no. 1, pp. 127–154, Jul. 2006. [Online]. Available: https://doi.org/10.1007/s10851-006-6228-4

[17] M. Turk and A. Pentland, "Face recognition using eigenfaces," in *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1991, pp. 586–591.

[18] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny, "Barlow twins: Self-supervised learning via redundancy reduction," *CoRR*, vol. abs/2103.03230, 2021. [Online]. Available: https://arxiv.org/abs/2103.03230

[19] A. Bardes, J. Ponce, and Y. LeCun, "Vicreg: Variance-invariance-covariance regularization for self-supervised learning," *CoRR*, vol. abs/2105.04906, 2021. [Online]. Available: https://arxiv.org/abs/2105.04906

[20] D. E. Rumelhart and J. L. McClelland, *Learning Internal Representations by Error Propagation*, 1987, pp. 318–362.

[21] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th international conference on Machine learning - ICML '08*. ACM Press, 2008. [Online]. Available: https://doi.org/10.1145/1390156.1390294

[22] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," 2013. [Online]. Available: https://arxiv.org/abs/1312.6114

[23] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," 2021. [Online]. Available: https://arxiv.org/abs/2111.06377

[24] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework," in *ICLR*, 2017.

[25] R. T. Q. Chen, X. Li, R. Grosse, and D. Duvenaud, "Isolating sources of disentanglement in variational autoencoders," 2019.

[26] H. Kim and A. Mnih, "Disentangling by factorising," 2019.

[27] A. Dosovitskiy, J. T. Springenberg, M. A. Riedmiller, and T. Brox, "Discriminative unsupervised feature learning with convolutional neural networks," *CoRR*, vol. abs/1406.6909, 2014. [Online]. Available: http://arxiv.org/abs/1406.6909

[28] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, 2005, pp. 539–546 vol. 1.

[29] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," *CoRR*, vol. abs/1503.03832, 2015. [Online]. Available: http://arxiv.org/abs/1503.03832

[30] K. Sohn, "Improved deep metric learning with multi-class n-pair loss objective," in *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., vol. 29. Curran Associates, Inc., 2016. [Online]. Available: https://proceedings.neurips.cc/paper/2016/file/6b180037abbebea991d8b1232f8a8ca9-Paper.pdf

[31] M. Gutmann and A. Hyvärinen, "Noise-contrastive estimation: A new estimation principle for unnormalized statistical models," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, Y. W. Teh and M. Titterington, Eds., vol. 9. Chia Laguna Resort, Sardinia, Italy: PMLR, 13–15 May 2010, pp. 297–304. [Online]. Available: https://proceedings.mlr.press/v9/gutmann10a.html

[32] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *CoRR*, vol. abs/1807.03748, 2018. [Online]. Available: http://arxiv.org/abs/1807.03748

[33] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep clustering for unsupervised learning of visual features," *CoRR*, vol. abs/1807.05520, 2018. [Online]. Available: http://arxiv.org/abs/1807.05520

[34] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," *CoRR*, vol. abs/2006.09882, 2020. [Online]. Available: https://arxiv.org/abs/2006.09882

[35] J. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. Á. Pires, Z. D. Guo, M. G. Azar, B. Piot, K. Kavukcuoglu, R. Munos, and M. Valko, "Bootstrap your own latent: A new approach to self-supervised learning," *CoRR*, vol. abs/2006.07733, 2020. [Online]. Available: https://arxiv.org/abs/2006.07733

[36] T. R. Davidson, L. Falorsi, N. De Cao, T. Kipf, and J. M. Tomczak, "Hyperspherical variational auto-encoders," 2018. [Online]. Available: https://arxiv.org/abs/1804.00891

[37] O. Skopek, O. Ganea, and G. Bécigneul, "Mixed-curvature variational autoencoders," *CoRR*, vol. abs/1911.08411, 2019. [Online]. Available: http://arxiv.org/abs/1911.08411

[38] A. Banerjee, I. Dhillon, J. Ghosh, and S. Sra, "Clustering on the unit hypersphere using von mises-fisher distributions," *Journal of Machine Learning Research*, vol. 6, pp. 1345–1382, 09 2005.

[39] S. Sra, "A short note on parameter approximation for von mises-fisher distributions: And a fast implementation of i s(x)," *Computational Statistics*, vol. 27, pp. 177–190, 03 2012.

[40] N. De Cao and W. Aziz, "The power spherical distribution," 2020. [Online]. Available: https://arxiv.org/abs/2006.04437

[41] P. Honeine and C. Richard, "The angular kernel in machine learning for hyperspectral data classification," in *2010 2nd Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing*, 2010, pp. 1–4.

[42] Y. Xu, Z. Liu, M. Tegmark, and T. Jaakkola, "Poisson flow generative models," 2022. [Online]. Available: https://arxiv.org/abs/2209.11178

[43] S. Borodachov, D. Hardin, and E. Saff, *Discrete Energy on Rectifiable Sets*, 01 2019.

[44] J. Brauchart and P. Grabner, "Distributing many points on spheres: Minimal energy and designs," *Journal of Complexity*, vol. 31, 07 2014.

[45] B. Wilson and M. Leimeister, "Gradient descent in hyperbolic space," 2018. [Online]. Available: https://arxiv.org/abs/1805.08207

[46] M. A. Hasnat, J. Bohné, J. Milgram, S. Gentric, and L. Chen, "von mises-fisher mixture model-based deep learning: Application to face verification," 2017. [Online]. Available: https://arxiv.org/abs/1706.04264

[47] A. Mnih and Y. W. Teh, "A fast and simple algorithm for training neural probabilistic language models," 2012. [Online]. Available: https://arxiv.org/abs/1206.6426

[48] I. Ahmad and P.-E. Lin, "A nonparametric estimation of the entropy for absolutely continuous distributions (corresp.)," *IEEE Transactions on Information Theory*, vol. 22, no. 3, pp. 372–375, 1976.

[49] J. Kim and C. D. Scott, "Robust kernel density estimation," *Journal of Machine Learning Research*, vol. 13, no. 82, pp. 2529–2565, 2012. [Online]. Available: http://jmlr.org/papers/v13/kim12b.html

[50] F. Nielsen and G. Hadjeres, "Monte carlo information geometry: The dually flat case," 2018. [Online]. Available: https://arxiv.org/abs/1803.07225

[51] W. Liu, R. Lin, Z. Liu, L. Xiong, B. Schölkopf, and A. Weller, "Learning with hyperspherical uniformity," 2021. [Online]. Available: https://arxiv.org/abs/2103.01649

[52] S. Smale, "Mathematical problems for the next century," *The Mathematical Intelligencer*, vol. 20, no. 2, pp. 7–15, Mar. 1998. [Online]. Available: https://doi.org/10.1007/bf03025291

[53] J. Thomson, "XXIV. on the structure of the atom: an investigation of the stability and periods of oscillation of a number of corpuscles arranged at equal intervals

around the circumference of a circle; with application of the results to the theory of atomic structure," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 7, no. 39, pp. 237–265, Mar. 1904. [Online]. Available: https://doi.org/10.1080/14786440409463107

[54] W. Liu, R. Lin, Z. Liu, L. Liu, Z. Yu, B. Dai, and L. Song, "Learning towards minimum hyperspherical energy," 2018. [Online]. Available: https://arxiv.org/abs/1805.09298

[55] A. Krizhevsky, "Learning multiple layers of features from tiny images," 2009.

[56] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385