Reach and Saccade Movement Direction Decoding from Monkey Local Field Potential Recordings

Victor ZhengYao Lü
Department of ECSE
Faculty of Engineering
McGill University
Montreal, Quebec, Canada

August 2015

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

1. Table of Contents

2. Abstract

Modern neural prostheses are promising medical devices that can be controlled exclusively by the user's brain. Such devices receive input signals through brain-computer interfaces (BCI). The goal of this project is to enhance BCI designs by implementing separate target direction decoders for the reach and saccade movements based on local field potential (LFP) signals. The LFP are recorded in monkey's medial intraparietal sulcus (MIP) and the dorsal premotor cortex (PMd), two cortices that are associated with arm and eye movement planning and execution. LFP is the signal of interest because it measures the neuronal ensemble activity with high spatial and temporal resolution while providing a long-term stability compared to the action potential (AP) signal. The data are recorded while monkey subjects performed center-out memory reach tasks coupled with saccade movements. Furthermore, the LFP signals are decoded using machine learning algorithms built upon the logistic regression (LR), artificial neural network (ANN), decision tree classifier (DTC), and support vector machine (SVM). While all four algorithms yielded 100% accuracy when performing binary classification for both movements, LR achieved 93% accuracy for classifying reaches to 3 directions, and 85% for 4 directions and 90% and 76% when classifying saccade to 3 and 4 directions. The results are comparable to the state-of-the-art AP based movement direction decoder while outperforming non-invasive BCIs such as EEG and fMRI. The high decoding accuracy is also attributable to the feature extraction technique that we applied, which is to use a multitaper power density estimation and to incorporate a large range of time and frequency information as opposed to the single frequency band based decoding method proposed by most researchers.

3. Abrégé

Les prothèses neurales modernes sont des dispositifs médicaux qui peuvent être contrôlées exclusivement par le cerveau d'utilisateur à travers des interfaces neuronale-directe (BCI). Le but de ce projet est d'améliorer le fonctionnement du BCI en construisant deux décodeurs de cible séparément pour la portée de la main et de la saccade oculaire à l'aide des signaux enregistrés du champ de potentiel local (LFP). Ces signaux sont recueillis dans le sillon intraparietal médial (MIP) et le cortex pré-moteur dorsal (PMd) de deux macaques. Les neurones de ces cortex sont souvent associés à la planification et à l'exécution des mouvements du bras et des yeux. En outre, LFP est le signal d'intérêt à cause qu'il mesure avec une haute résolution spatiale et temporelle tout en offrant une stabilité à longue terme contrairement aux signaux de potentiel d'action (AP). Les données sont recueillies lorsque les macaques performent des taches d'atteindre des cibles mémorisées de façons centrifuge couplée aux saccades oculaire. De plus, les signaux LFP sont décodés en utilisant des algorithmes d'apprentissage machine (ML) construits sur la régression logistique (LR), le réseau de neurones artificiels (ANN), l'arbre décisionnelle (DTC) et machine à vecteurs de support (SVM). Toutes quatre méthodes ont atteint une précision de 100% lorsqu'on distingue entre deux directions pour les deux sortes de mouvements. Dorénavant, le LR a atteint une précision de 93% pour classifier les portées à trois directions, et 85% pour quatre directions. De même, une précision 90% et 76% a été atteint pour les classifications de saccade à trois et à quatre directions. Ce résultat est comparable à la performance des décodeurs basée sur AP tout en surperforment les modalités non-invasive comme le EEG et le fMRI. La haute précision de décodage peut également être attribuée à notre technique d'extraction de caractéristique qui utilise une estimation spectrale multitaper et qui intègre une large gamme d'information temporelle et fréquentielle contrairement à la méthode de classifier les attribues de mouvement à partir d'une seule bande de fréquence.

4. Acknowledgements

Foremost, I'd like to thank my supervisor, Dr. Sam Musallam, for presenting me with the opportunity to work on this innovative project and for all the scholarly advices he provided throughout the years of my master's study. Most of all, I thank him for allowing me to explore the wonderful realm of neural science.

I'm also forever grateful to my labmates, Mr. Greg Stacey, a PhD candidate, Mr. Nathan Friedman, a recent MEng graduate, and Ms. Rubing Xu, a former PhD candidate, for always making themselves available when I'm in need of enlightenment. They introduced me to the scientific ways of the thinking via the memorable sessions of the journal club.

I'd also like to show my appreciation to the other, past or present, members of my lab: Brendan Thorn, Mohammad Poustinchi, Kelly Bullock, Rishi Rajalingham, Benjamin Lourtau, Caitlin Wharin, Pascal Labrecque, George Xereas, Mitchel Benovoy, and Alina Dragomir. They have all left their marks in making this lab a wondrous place.

Ultimately, I'd like to thank dear my parents and friends for their everlasting support through times of hardship.

5. Contributions

Mr. Greg Stacey collected all the monkeys' local field potential data, as well as the hand and eye movement recordings, prior to my arrival in the lab.

Dr. Sam Musallam, Mr. Greg Stacey, and Mr. Nathan Friedman wrote some algorithms used in the data pre-processing.

I accomplished the data analysis and thesis redaction.

6. Background

Brain-computer interfaces (BCI) present a vast array of medical applications by allowing the users to interact with the environment solely with their mind. One particularly interesting quest at the focus of our lab is to advance the research in neural motor prostheses control. These BCI can greatly improve the quality of life of the patients suffering from partial or total motor impairment. A report by Cahill *et al.* [1] assesses the prevalence of paralysis and spinal cord injuries in the United States at 1.9% in 2009, and similarly, currently 675,000 paralysed patients are living in Canada based on the latest national survey [2]. In addition, 192,000 patients are living with some degree of limb amputation, calculated based on the 0.54% prevalence of limb loss [3]. Other related research fields of neuroprosthetics are focusing on reinstating sensory systems, such as visual and hearing impairment, which are assessed to affect 3.4% and 8.2% of the population respectively [4]. The main difference between these two categories is that the former seeks to connect the brain to movement actuators, while the latter seeks to connect sensors to the brain's nervous system. This project focuses on improving the former BCI.

A typical BCI design comprises three major steps. **First**, the signal is acquired and preprocessed from the user's central or peripheral nervous system. Many modalities have been studied to record different types of brain signals. **Second**, these recorded signals are channeled towards the feature extraction and classification algorithm. The former detects and identifies the discriminative features, and the latter analyses these features and attempts to correctly label the observation based on the trained database of known observations. **Finally**, the BCI sends out a specific control command to the actuator as a function of the classification output.

Each of the abovementioned steps is essential to the overall performance of the BCI; an improvement brought to any of these steps can potentially increase the system's overall efficiency. In this project, we will make an attempt to improve the second step by implementing decoders built upon modern machine learning algorithms namely the logistic regression (LR), the artificial neural network (ANN), the decision tree classifier (DTC) and the support vector machine (SVM). But first, the readers will be introduced to the different techniques of the brain signal acquisitions and be presented the advantages and disadvantages of the recording modality at the focus of this project, i.e. the local field potential (LFP). The implementation of the control output and the actuation of the command is an entirely separate endeavour by itself, and will be reserved for a future project.

6.1 BCI Background

Two milestones in the development of BCI are widely recognized in the research community. The first is when Prof. Hans Berger, a German neurologist, recorded EEG signals in the first human subject in 1924 and identified the alpha wave [5]. He observed that the brain waves displayed dynamics that can be associated with the patient's brain diseases. The second milestone is when the term BCI was coined by Prof. Jacques J. Vidal in 1973 in his pioneering work [6], in which he reported the successful implementation of a cursor control system based on visually evoked potential. Nevertheless, the number of research groups interested in harvesting the power of cortical signals were scarce in the next few decades due to both the lack of appropriate technology and the lack of user confidence [7]. The situation turned at the beginning of the millennium when peer reviewed publications related to BCI increased exponentially. Meanwhile, the lay public gained insights to the seemingly limitless possibilities of BCI through media coverages such as the documentary entitled "Het

Gretige Brein" (The Eager Brain) produced in by VPRO, a Dutch TV company. The broadcast presented a quadriplegic patient moving a robotic arm to serve herself a cup of coffee [8] with the help of implanted sensors in the brain. Other public exposures include the most recent science-fiction blockbuster of the *RoboCop* franchise [9], where the brain of a police officer is salvaged in a cyborg armour. This media coverage stimulated the eagerness of the general public to learn more about BCI which greatly facilitated the cutting edge research that would be objectionable a decade ago [10].

6.2 Recording Modalities

The direct beneficiary of this expanding research field is the improvement of the cortical signal recording modalities. These methods can be broadly distinguished into two classes, non-invasive methods such as EEG, fMRI, and fNIRS, and invasive methods such as SUA, ECoG, and LFP. The preliminary BCI researches were almost exclusively focused on non-invasive EEG methods [11]. Later studies saw a shift towards the invasive methods to achieve a better control speed and accuracy. In this section, we will compare the two classes and describe the major constituents.

6.2.1 Non-Invasive Methods

The non-invasive methods as reviewed by Millán *et al.* [12] are sought to be the proper way of recording brain signals by virtue of their non-invasiveness as the name suggests. In other words, they do not require any physical incision to the brain, the skull, nor the scalp. Because of this user friendly form factor and the lower medical risks, not only can they aid paralyzed patients, but they can also provide a way to augment able-bodied humans with everyday activities such as driving [13] and gaming [14].

EEG The most promising method among this class of BCI is the electroencephalogram (EEG). *Niedermeyer's Electroencephalography* [15] offers an in-depth review. In essence, neurons exchange ions with the extracellular surrounding in order to maintain the cell's resting potential or to propagate action potentials. This exchange of ions reflects the activation of a neuron, and eventually the ensemble activity of thousands or millions of neurons propagates altogether to the recording sites. This electrical voltage is accumulated, both constructively and destructively, by the propagation of the waves through the layers of neurons. The outcome of this propagation is a measurable voltage oscillation at the scalp where the electrodes are located. The oscillations encompass activities in different rhythmic frequency bands, each believed to be associated with specific neurophysiological behaviours. For example, the alpha waves ($8 - 13$ Hz) in the visual cortex are associated with the synchronous activity of thalamic pacemaker cells which are present during wakeful relaxation with eyes closed [16]. In other words, one can predict whether the subject with closed eyes is awake by analysing the oscillation patterns of the alpha waves. Such analysis methods have been studied extensively in the past decades to be used in diagnosis of various disorders such as epilepsy and sleep disorders by discerning abnormalities in the patient's EEG readings. Researchers have also discovered useful properties which allow EEG to be used as a basis for BCI as reviewed by Padmavathi and Ranganathan [17].

The basic technique employed in EEG-BCI designs uses evoked potentials, a response recorded from the nervous system following the presentation of a stimulus. A realisation of this method is the Steady-State Visually Evoked Potentials (SSVEP) where the user is instructed to gaze at an LCD screen with areas flickering at different frequencies. By fixating on different targets, an increase in the EEG power of the frequency corresponding to the

flicker can be observed and thus the command associated with the flicker can be executed, e.g. an online speller [18]. The issue with these BCI is that they require the user's attention to focus on the stimulus and therefore is not suitable for control tasks such as wheelchair maneuver. For this reason, researchers called upon the ability of the brain to perform motor imagery [19] or mental tasks [20] that can be decoded with the help of algorithms. The issue with these paradigms is that the user could be performing them inattentively, which greatly affect the BCI's real world performance. Moreover, there are other significant noise sources that can affect EEG signals such as electrocardiogram (ECG), electrooculography (EOG), electromyography (EMG), just to name a few. These noise sources are caused by the electrical potential related to muscle activation corresponding to the heart, the eyes, and the limbs respectively.

fMRI Another commonly used recording modality is the blood oxygen level dependent (BOLD) functional magnetic resonance imaging (fMRI) whose physical principles are described in Huettel's book [21]. Simply put, the BOLD fMRI exploits the difference in hemoglobin's response to the induced magnetic field depending on whether it is oxygenated (oxy-Hb), or is depleted of its oxygen molecule (deoxy-Hb). Detecting this difference leads to the measurement of cerebral blood flow (CBF) which is strongly believed to be correlated with the neuronal activation level. This measurement technique has been used extensively to map the brain's functional areas for research purpose since the 1990s. Because the fMRI can realize high spatial resolution in the millimeters range and moderate temporal resolution in the seconds range, it is thought to be a practical BCI basis in the last decade as reviewed by Sitaram *et al.* [23]. This review has assessed that human subjects, with the help of real time fMRI providing visual feedback, can learn to voluntarily regulate brain activity

in a great number of brain regions. An interesting BCI application resulting from the voluntary regulation is the processing of emotions [24]. The experiment was conducted such that the patient is shown two curves representing the BOLD activity in the rostral-ventral and the dorsal part of the anterior cingulate cortex (ACC), two regions of the brain commonly associated with emotional valence and arousal. The possible extensions of this result are the study of correlation between behavioural states and cortical regions. Nevertheless, due to the large form factor and high usage cost of current fMRI equipment, along with the patient's movement limitation during recording sessions, everyday use of fMRI-BCI is yet impractical.

fNIRS An emerging brain activity recording modality is the functional near-infrared spectroscopy (fNIRS) which, similar to fMRI, measures the hemodynamic response of neuronal activity. Instead of measuring the magnetic response, fNIRS measures the difference in light (700~900 nm wavelength) absorption spectra of the oxy-Hb and deoxy-Hb. Unlike fMRI, fNIRS can be recorded with a portable and affordable instrument [25]. Considering these factors, research groups have investigated the implementation of fNIRS-BCI in several fields. An in-depth review of fNIRS-BCI has been published very recently [26]. The first category of application is a binary prediction schema, i.e. "yes-no" or "start-stop". The classification rate was about 80% in most test cases. Interestingly, the same accuracy is observed in totally locked-in ALS patients, although only 40% of the tested patients reacted to the experiment. The group also argued that fNIRS-BCI are limited in real life settings due to the slow information transfer rate and high prediction error. Nevertheless, Shin and Jeong [27] reported that they achieved a higher classification accuracy of about 90% and a faster transfer rate with their custom built fNIRS recording device. A continuous improvement in these two factors can eventually result in more advanced application of fNIRS-BCI.

6.2.2 Invasive Methods

SUA Despite the non-invasiveness of the BCI modalities described above, they all share a common drawback that is the slow control speed and relatively low spatial and temporal resolution. Henceforth, the class of invasive methods provides a greater signal specificity which translates to faster control speed and higher resolution as many researchers argued and summarized by Lebedev and Nicolelis [28]. The methods described below, along with EEG, interpret neural activity by means of electrophysiological (EP) behaviour of the neurons. The first experiment of neuronal activity recording was reported by Edgar Adrian in 1928 [29], who later won the Nobel Prize for his work in this field. The technique has since been evolving to develop better microelectrodes with greater precision. In 1998, Kennedy and Bakay successfully developed an action potential based BCI to allow an ALS patient to control a computer cursor [30], at which point the scientific community saw the potential in the invasive BCI. In single unit activity (SUA) recordings, the microelectrode measures the electrical potential adjacent to the target neuron. When the neuron is activated, it is said to "fire" an action potential down its axon to the receiving neuron. The rate at which it fires is called the "firing rate" (FR). Experiments have shown that the FR encodes pertinent information, e.g. reach movement planning from the premotor cortex neurons [31]. It is believed that since neuronal activity are at the basis of all brain process, decoding the FR can lead to the most robust BCI. However, there is an important issue that is the invasive procedure of electrode insertion which scars the cerebral matter at the insertion site. Also, in the presence of foreign objects, a sheath of glial substance develops on the electrodes as a self-healing mechanism of the central nervous system [32]. This gliotic sheath has high impedance properties that force the electrical current to flow through the surrounding

region which has a lower impedance. Therefore, the signal strength deteriorates over time, leaving the SUA-BCI not viable for long-term usage.

ECoG In order to avoid the highly invasive nature of electrode implants while recording at a high resolution, researchers have introduced the electrocorticography (ECoG) which measures the electrical potential at the surface of the cortex, either above (epidural) or below (subdural) the dura mater. Its principles are similar to EEG and is hence also called intracranial EEG (iEEG). ECoG implants were first used clinically in 1950s by Dr. Wilder Penfield and Dr. Herbert Jasper in the renowned "Montreal Procedure" to localize epilepsy tissues [33]. Despite its current clinical use in human patients, the placement of the electrodes and the duration of the implantation are carefully chosen for the brain disease pathology and not fully versatile for research purpose [34]. Nevertheless, researchers have obtained visuomotor task related ECoG recordings, with the patients' consent, and demonstrated the capability of this modality from offline analysis [35], [36]. In 2004, Leuthardt *et al.* realized the first online ECoG-BCI to decode the direction of hand movements [37]. This latter study was conducted with four participants to control a one-dimensional cursor and achieved a 74–100% accuracy after a short training period of 3 – 24 min. In 2008, another group successfully implemented an online two-dimensional cursor control system based on imagined or actual motor tasks [38]. Albeit these promising results, ECoG-BCI studies are currently limited to a restrained population of epilepsy patients for a training window of only a few hours per day. On a technical aspect, ECoG arrays provide too coarse of a spatial resolution, around 1 mm [39], to represent the activities of localized neuron populations. The article by Schalk and Leuthardt [40] offers an in depth review of ECoG-BCI.

6.2.3 Local field potential (LFP)

LFP uses similar electrode design as SUA recording and it is the modality that is studied in this project. Unlike SUA, LFP measures the summed electrical current generated from surrounding neuronal activity in a radius of $50 - 350\,\mu m$. By measuring the electrical discharge from multiple neurons, the LFP recordings mainly remediate the diminishing signal amplitude issue of SUA recordings [11]. The reason that LFP signal is reliable is due to the columnar structure of the cortex known as cortical columns [41]. In other words, neurons are generally physically grouped together to form columns and are activated by the same stimulus. Thus, the electrode that is recording from a specific column is exploiting the additive property of the electrical potential of the neuronal population. Nevertheless, the multiple sources of electrical potential make LFP more difficult to interpret than SUA signals and thus more advanced modelling and signal processing methods need to be developed in order to fully acquire the informative content. The most widely used analysis of LFP is the frequency analysis. This method constitutes of decomposing the continuous LFP signal into frequency bands with the help of mathematical tools such as the Fourier transform. It then studies the correlation between the amplitude in each frequency band with the stimulus [42]. Other studies have attempted to further improve this understanding of LFP by tackling specific bands, e.g. the gamma rhythms (30-80 Hz) [43], or specific brain areas, e.g. the premotor cortex [44]. Despite, the promising long-term usability and the high spatial resolution of LFP signals, studies of online LFP-BCI are still non-present due to the invasiveness of the electrodes and a general lack of interest among the research community.

Further, the frequency domain decomposition of the LFP signal observes two challenges as described by Babadi [45]. First, an infinite length time window is required to

accurately describe the frequency content. Second, there are some degree of stochasticity due to both the inherent property of brain signal and the measurement noise. The latter issue would require an infinite number of repeated observation to properly nullify the stochasticity within the signal, e.g. by taking the sample average. However, the actual recording is limited in both a finite time window and a single realization of the output. Thus researchers have designed spectral estimation techniques to extract the power content using a Multitaper Method (MTM) [46] that remediates both the high bias and high error variance in finite estimates. Such method is used in this project to perform the feature extraction step of LFP signals.

6.3 Brain Signal Decoding

Brain signal decoding is often regarded as beyond reach by the lay public and the mere idea of "mind reading" incites fear of privacy infringement in some people. Nevertheless, the current technology provides the possibility for researchers to discriminate specific mental states often with high accuracy. The rationale is that if the brain activity differs, ever so slightly, when it is occupying distinguishable mental states, then the decoder should be able to detect this disparity and to discern the underlying mental state. As mentioned previously, a vast array of BCI are available to interact with the brain and to record different signals (electrical, magnetic, or optical) that reflect the brain activity. More specifically, within the electrical modalities, different methods are often necessary to exploit the available data in order to achieve the highest decoding accuracy. While early attempts to classify brain signals are almost exclusively rule-based, e.g. thresholding [47], the advancement in the field of machine learning allowed the implementation of more accurate decoders. One of the main reasons is that the latter has access to a larger set of decision boundaries unattainable by

human capability. As the number of features increases, it is more difficult, if not impossible, for humans to correctly assign a threshold at the boundary of two classes. Computers, on the other hand, are only limited by the Moore's law and the complexity of the mathematical model.

Supervised Learning The first class of machine learning algorithms is known as supervised learning. In this group, the training model has access to a set of labeled data that is derived from observed behaviour of the subject, e.g. hand or eye coordinates. The prediction of the decoder is then restricted to this label space, which can be discrete, e.g. reach direction, or continuous, e.g. movement velocity. A number of mathematical models are available to describe the supervised learning process. A few all-purpose algorithms that are widely used in the research community are presented here along with their applications in the brain signal decoder implementation.

Linear regression, a product of the regression analysis, is possibly the most straightforward model to be employed in a continuous labeling scheme. The model fits a linear relationship between the input data vector, and its mapping onto higher dimensions, and the target data by means of error minimisation, often using a least squares approach. A group of researchers used this model to study the information content of brain signals by regressing kinematic measurements such as position and velocity on LFP power spectrum [48]. Another group took a step further and applied the model to predict planar and 3D limb kinematics from EEG signals [49]. These examples illustrate the possibility of brainwave analysis using the linear regression method.

An analogous model that exploits the linear relationship is the Logistic Regression (LR) [50] which is used to predict discrete label sets. In simple terms, the model calculates the

probability that the instance is associated with a certain outcome among the available classes by mapping a linear function of the feature vector to the probability space using a logistic sigmoid function. This method has allowed researchers to use EEG signals to perform motor imagery classification [51] and reach target prediction [52], [53]. Another group used this method to classify grasping motions in monkeys [54].

Taking a further step from the logistic regression, scientists have built the Artificial Neural Network (ANN) algorithm. Mimicking the brain's neural hierarchical architecture, each node, or neuron, within the network assigns numeric weights to nodes found at the lower layer and outputs an activation value. The weight assigned to each node are updated using an error backpropagation updating scheme. The potential of this method was first reported in the recognition of handwritten digits (0~9) [55]. A study of ANN's capability is conducted to predict the onset of Parkinson's disease tremor from recorded LFP signals [56] and to detect the presence of index finger movement attempts from tetraplegic patient's EEG and MEG signals [57].

To tackle the issue of large length of feature vectors, researchers have developed a set of ensemble techniques that only learn on subsets of the feature space. The best known model amongst this category is the random forests algorithm developed by Breiman [58] which constructs decision tree classifiers (DTC) [59] based on randomly selected subsets of the features. These trees are then trained with randomly selected samples. This injected randomness effectively remediates DTC's tendency to overfit as argued by Breiman. The method has been reported successful in detecting anomalies in neonatal EEG signals [60] and brain region activation patterns in fMRI data [61]. Although the research on the

suitability of the random forest on LFP signals is yet vacant, the method is a pertinent candidate for this classification study.

Built on another set of mathematical models is the (soft margin) Support Vector Machine (SVM). This model identifies the support vectors defining the boundary between two classes; new observations are labeled uniquely using this boundary. The soft margin variant of the model [62] allows mislabeled data points to be present on either side of boundary that would otherwise forbid the outlining of a clean boundary. This trait is essential in the application of SVM on realistic data. In addition, a practical advantage of the SVM is its relative robustness against the "curse of dimensionality" [63], also known as the Hughes phenomenon, named after G. Hughes (1968) who showed that the number of required training samples grow exponentially as the dimensionality increases. Further, groups of researchers have used this method to classify behavioural tasks from LFP signals of Parkinson's disease patients [65] and reach-and-saccade targets from EEG signals [66].

Recent development in machine learning led to the rise of a second class of algorithms known as the unsupervised learning, whose goal is to classify the input signals into distinct classes without, or with only partial, prior knowledge of the class labels. Its employment in the field of neural science is still in its infancy and further research has yet to be done. Nevertheless, its promises are countless as mentioned in the book titled "Unsupervised Learning: Foundations of Neural Computation" [67]. Despite its values, the data available for this project is not fully adequate for applying unsupervised learning and therefore no results are available to be shown.

6.4 Brain Areas involved in Visually Guided Reach

Reaching for objects is a seemingly simple task in the everyday living. However, the brain must incorporate a vast array of information in order to guide the arm from the current position to the location of the reach target. In order for the subject to successfully reach the target, its brain processes the information by passing signals from one area of the brain to the next along the so-called neural pathways. Each area along this pathway encodes the signal differently and thus signal transformation needs to take place, i.e. the *sensorimotor transformation*.

To realize the visually guided reaches paradigm that is studied in this project, the subject's brain must go through the following steps. First, the visual stimulus activates the photoreceptors located within the retina at the rear end of the eye. Then the information is passed along the optic nerves, a chain of neurons, to the visual cortex (V1). Here, the information is encoded in a retinocentric reference frame [68] considering that the retina shifts with the direction of the gaze. Henceforth, a transformation from the eye centered to the hand centered reference frame needs to take place for the reach to occur. After analyzing the neuronal activity in the posterior parietal cortex (PPC) and the premotor cortex (PMC), researchers found that they encode coordinates in both the abovementioned reference frames [69] which is consistent with their role in the sensorimotor transformation process.

Furthermore, within the PPC is located the parietal reach region (PRR) which is associated with the reach process [70]. PRR then sends signals directly to the dorsal premotor cortex (PMd) [71], which is located within the premotor cortex (PMC). Chang *et al.* suggests that PRR is involved in the preparation for the reach movement of the contralateral (opposite side) limb while the PMd further prepares both the contralateral and the

ipsilateral limbs. Finally, the reach movement information is processed in the primary motor cortex (M1), where the coordinates are encoded in a body centered reference frame [72] allowing the reach to be performed effectively.

Two cortices were mentioned to be associated with reach preparation: PPC and PMC. These areas are chosen for the experiment's data acquisition sites for the reason that both encodes coordinates in eye centered and hand centered reference frames. Readers are invited to read the following sections in order to familiarize themselves with the basic functions of these two regions.

6.4.1 Posterior Parietal Cortex (PPC)

The PPC is a section of the parietal lobe posterior to the primary somatosensory cortex consisting of Brodmann area 5 and Brodmann area 7 of the primate brain. Historically, lesion studies helped the discovery of the role of the PPC by assessing deficits in patients with lesions in this brain area. The most noteworthy deficit is the apraxia: an acquired difficulty in motor planning while coordination and senses are unaffected [73]. More recent studies on brain mapping indicate that the PPC is composed of sub-areas with more specialized roles of movement planning; the lateral intraparietal area (LIP) specialises in saccadic eye movement, the PRR, which contains the medial intraparietal area (MIP), specialises in reaching, and the anterior intraparietal area (AIP) specialises in grasping [74]. To dissociate the functions of these brain areas, Snyder *et al.* designed a delayed tasks experiment that implied a time delay in the execution of the tasks (saccade or reach). Noticing the preferential neuron tuning during the planning or the memory stage, that is the epoch prior to the task execution, the authors draw the conclusion that LIP and PRR are specialized differently with the abovementioned roles.

Zhang and Barash [75] designed an experiment to further distinguish whether the neuronal response is purely visually triggered or did movement planning actually occurred. The subject was trained to perform a saccade either towards the cue (pro-saccade) or opposite to the location of the cue (anti-saccade) depending on the color of the visual cue. They discovered that the tuning of the neurons were first biased towards the location of the cue, but the tuning switched rapidly, within 50 ms of the signal's arrival in the LIP, towards the instructed movement direction. This result support the hypothesis that LIP encodes the movement plan rather than the visually stimulus. Gail and Andersen later designed a similar anti-reach experiment to show that PRR encodes the reach plan rather the stimulus memory [76]. In addition, Steenrod *et al.* showed that LIP encodes the target location rather than the movement vector towards the target [77] by dissociating the stimulus location and the saccade end-point.

6.4.2 Premotor Cortex

The PMC is part of the motor cortex corresponding to part of the Brodmann area 6 in the primate brain. Its function is not yet fully understood, but results show that it is responsible for movement planning using high level abstract rules [78]. The PMC is further divided into sub-regions, each having specific roles; the frontal eye fields (FEFs) are associated with saccades [79], the dorsal part (PMd) is associated with reaching [80] while the ventral part (PMv) is associated with grasping [81].

Evidence show that the PMd is responsible for the reference frame transformation in the reach movement planning [82]. Functional analysis of the brain places this area at the affluent of the PRR and at the influent of M1 and the spinal cord. Unlike the target location tuned neurons of the PRR, PMd neurons are tuned to the direction of the reach movement;

this region employs a shoulder-centered reference frame. To show this, Caminiti *et al.* [83] designed an experiment which dissociated reach direction and reach endpoint by varying the arm's initial position.

6.5 Summary

The above results provide ample reasons for our experiment to record brain signals from the PRR and the PMd and to decode the reach movement plan from the LFP data. Also it is of interest to simultaneously decode saccade direction from these regions in order to confirm the specificity of the regions. The goal of this project is to provide an offline feasibility study of LFP-BCI based on various popular machine learning algorithms for the research community to eventually develop a reliable online BCI that can be implanted for long term use.

This paper is structured as follows. In the Materials and Methods section, the experimental setup is explained which emphases on the description of the delayed memory reach task. Then the models of various machine learning algorithms studied in this project are exposed. In the Results section, we demonstrate the properties of the LFP signal related to the saccade and reach movement as well as the decoding accuracy with respect to each decoding algorithm as a function of the modeling parameters. Finally, we study some interesting downfalls of the decoders and possible adjustments in the Discussions section.

7. Materials and Methods

7.1 Test Subjects

The training and data collections were performed in 2010 on two awake and behaving male rhesus monkeys (*Macaca mulatta*), M and H, weighing 6.5 and 11.9 kg respectively. Both monkeys are right-handed. The monkeys were implanted with head fixation posts (MRI Chair by Rogue Research). Then microelectrode arrays (MicroProbe, Gaithersburg, MD) were implanted in the left medial intraparietal (MIP) and left dorsal premotor cortex (PMd). The sites were localized using a MRI-based BrainSight (Rogue Research).

Monkeys were given Glycopyrrolate intramuscularly at a dose of 5 $\mu g/kg$ followed by a dose of Ketamine at 10 mg/kg for the induction of general anesthesia. Isoflurane gas $(1 - 4\%)$ was given to maintain general anesthesia. All surgical procedures were carried out under sterile conditions. After the surgical procedures, analgesia was administered to the monkeys for pain control for 10 days. The monkeys were given 14 days to recover before starting experiments. Daily body weight of the monkeys were measured to monitor health and growth. Throughout the testing period the animals were pair-housed and could exercise in a jungle gym. Monkeys were given fresh fruits after all completed sessions. All experimental procedures complied with the Canadian Council of Animal Care guidelines and were approved by the McGill animal care committee.

7.2 Electrophysiological Recordings

We used multielectrode array recordings to record neural activity from monkeys M and H. Monkey M was implanted with four 16-channel Floating Microelectrode Arrays (FMA) in MIP and one 32-channel FMA in PMd. Monkey H was implanted with two 16-channel FMA,

in both MIP and PMd. Refer to Figure 1C for array placement. Three arrays (48 channels) were recorded simultaneously in each session from MIP and PMd. The lengths of electrodes in MIP ranged from $1-9\,mm$ in both monkeys in order to sample the depth of the intraparietal sulcus. Electrodes in PMd varied between $0.5-1.5\,mm$. Electrode impedance is in the range $0.4-0.6\,M\Omega$. Since both animals were trained to reach with the right arm, all arrays were implanted in the left hemisphere. Signals were initially amplified using 20x headstage (Plexon, Inc) and sampled at 40 kHz using a Plexon acquisition system (Plexon, Inc) and continuously recorded to the disk for offline analysis. The local field potential (LFP) signals were extracted from the waveform by first lowpass filtering to 8 kHz using a third order Butterworth filter and then down-sampled to a rate of 1 kHz.

7.3 Behavioural Task

The two monkeys were trained to perform a modified center-out memory reach task as shown in Figure 1A. The monkeys were seated inside a grounded copper Faraday cage while their heads were held stationary by the implanted headpost. The behavioral tasks were run by a real-time system (LabVIEW RT, National Instruments). An infrared reflection camera monitored gaze direction (ISCAN, Boston). The monkeys' two-dimensional hand position was monitored by an acoustic touchscreen (ELO Touch, California) coupled to an LCD monitor. The monkeys were only allowed to touch the monitor with their right hands while their left hands were constrained inside the seating apparatus. The monkeys sat approximately 45 cm in front of the touchscreen oriented along the fronto-parallel plane. The monkeys were trained to follow the eye targets and to touch the reach targets as indicated on the monitor. The monkey were rewarded with juice for completing each successful trial. Once the trial was initiated, the eye and hand positions were constrained

within 2 cm radius windows of the central target until the reward was delivered; otherwise, the trial was aborted and no reward was administered.

Monkeys were trained on three tasks differed by the eye movement condition: fixation, saccade, and pursuit. Each trial of the three tasks contained one reach movement to four targets. Trials from all conditions were initiated when the animal visually fixated a red central eye target and touched a green central reach target for 700 ms (Figure 1A, panel I). A peripheral reach cue was then flashed for 600 ms, indicating one of four reach target locations (panel II). The time from the disappearance of the peripheral reach cue until the instruction to reach was defined as the memory period (panel III). After holding the correct reach target for 500 ms, monkeys were rewarded with juice. All cues and targets were solid color circles approximately 1 cm in diameter.

Fixation condition. During fixation trials, monkeys performed a memory reach task with their eyes fixed centrally throughout the trial (Figure 1A, panel IIIa,b). Trials were initiated as described above. In all fixation trials, the memory period lasted between $1.2 - 1.6\ s$. Monkeys were instructed to reach to the remembered location of the reach cue by extinguishing the central reach target (Figure 1A, panel IVa).

Saccade condition. On saccade trials, trials were initiated as described above. $600 - 800\ ms$ after the start of the memory period (Figure 1A, panel IIIc) animals were instructed to perform a saccade during the memory period (panel IIId). The central eye target was extinguished and an eye target appeared at a location in one of the four reach target directions: up, down, right or left (Figure 1B). The saccade targets were 10 cm from the central eye target. The monkey had 250 ms to initiate the saccade or the trial was aborted.

The memory period lasted the same length of time as in the fixation condition, and all other trial periods were identical.

Pursuit condition. Pursuit trials were initiated as described above. In the Pursuit condition, the animals were instructed to perform a pursuit eye movement during the memory period. $700 - 900\ ms$ after the start of the memory period (Figure 1A, panel IIIe) the central eye target moved smoothly to an endpoint 10 cm away (panel IIIf). The pursuit endpoints were the same as the saccade targets used in the Saccade condition. The pursuit eye movement lasted approximately $1250\ ms$. The memory period ended $200 - 700\ ms$ after the pursuit ended.

7.3.1 Trials distribution

The number of successful trials is approximately evenly distributed among conditions and reach directions. All reach cues were presented at a distance of 7 cm from the central fixation (Figure 1B). All datasets consisted of at least 240 successful trials (3 trial conditions x 4 directions x 20 trials each). The pursuit conditions are only analysed in terms of spectral analysis and not decoding analysis.

7.3.2 Event Alignment

Each trial is first aligned in time to the onset of hand or eye movement accordingly. For hand movements, the onset is observed when the touch signal is lost as the monkeys lifted their hands to reach towards the target (Figure 1A, panel IV). The fixation trials are aligned to the middle of the memory period. The saccade alignment is achieved by locating the sharpest change in gaze direction after the saccade cue onset. The pursuit alignments were performed manually as the eye traces do not display any consistent trends.

7.4 Feature Extraction

For the decoders to achieve optimal performance, it is important to extract relevant features from the data that are suited predictors of the movement directions. In this project, we selected the MTM as proposed by the literature for temporal-frequency decomposition. The method takes in the time series of the LFP signal and outputs a two-dimensional spectral power matrix for each trial

$$MTM: \mathbb{R}^{T\times N\times L} \rightarrow \mathbb{R}^{T\times (N*F*W)} \quad (1)$$

where $\boldsymbol{T}$ represents the number of trials and $\boldsymbol{N}$, the number of channels. For each of channel and trial pair, the LFP time series of length $\boldsymbol{L}$ is decomposed into a matrix of $\boldsymbol{F}$ frequency bins by $\boldsymbol{W}$ time bins. The four-dimensional spectral power matrix is further unrolled into a two-dimensional matrix of size T by $N * F * W$ such that each row represents an observation and each column represent an independent feature dimension.

Nevertheless, the dimension of such feature space grows exponentially as the number of channel, time bins and frequency bins increase, and not all dimensions contribute to the prediction with equal significance; some features are effectively noise that deteriorate the decoding performance as shown in the results. Therefore, it is necessary to implement a ranking scheme to reduce the number of features to maximize the decoding accuracy. Two feature reduction methods are evaluated, the principle component analysis (PCA) and a correlation coefficient based method.

7.4.1 Multitaper Method (MTM)

The LFP signals were transformed using the multitaper method [45]. Let the sampling rate be $F_s = 1/\Delta$. Thus the sample data in discrete time is $x_k = x(k\Delta), k = 0, \dots, N-1$. Its discrete time Fourier transform (DTF) is thus

$$X(f) = \Delta \sum_{i=0}^{N-1} x_k \, exp(-j2\pi k f \Delta) \tag{2}$$

and its power in the frequency band $f \pm \frac{1}{2}\delta f$ is approximately

$$P = \frac{1}{N\Delta} |X(f)|^2 \delta f \tag{3}$$

Assuming an infinite sample size $N \to \infty$, the average of the power around f is

$$\lim_{N\to\infty} \mathbb{E}\left\{\frac{1}{N\Delta} |X(f)|^2\right\} \delta f \tag{4}$$

However, due to the limited signal length, the common practice is to assume a second-order stationary process

$$\begin{aligned} \mu_i &= \mathbb{E}\{x_i\} = \mu, \qquad i = 0, 1, \dots \\ S_{i,j} &= \mathbb{E}\{(x_i - \mu_i)(x_j - \mu_j)\} = S_{|i-j|}, \qquad i, j = 0,1, \dots \end{aligned} \tag{5}$$

Thus the power spectral density (PSD) can be approximated by virtue of the Wiener-Khinchin theorem [84]

$$\lim_{N\to\infty} \mathbb{E}\left\{\frac{1}{N\Delta} \left|X_p(f)\right|^2\right\} = S(f) = \Delta \sum_{k=-\infty}^{\infty} s_k \, exp(-j2\pi k f \Delta) \tag{6}$$

The next step is to estimate the autocovariance sequence s_k given a finite length signal x_k. Let $\hat{S}(f)$ be an estimate of $S(f)$ that satisfies both unbiasedness and low variance. Such approximation is realized by averaging the periodogram calculated from multiple tapers. Given a random sample x_k of size N

$$\begin{aligned} \hat{\mu} &= \frac{1}{N} \sum_{i=0}^{N-1} x_i \\ \hat{s}_k &= \frac{1}{N} \sum_{i=0}^{N-k-1} (x_i - \hat{\mu})(x_{i+k} - \hat{\mu}) \end{aligned} \tag{7}$$

its Fourier transform is given by the periodogram

$$\hat{S}(f) = \Delta \sum_{k=-N+1}^{N-1} \hat{s}_k \, exp(-j2\pi kf\Delta) = \frac{\Delta}{N} \left| \sum_{k=0}^{N-1} x_k \, exp(-j2\pi kf\Delta) \right|^2 \tag{8}$$

The next step is to apply tapering, an effective trade-off between broadband and narrowband bias of spectral estimates. Consider tapers $h_k, k = 0,1, \dots, N-1$

$$\hat{S}^t(f) = \Delta \left| \sum_{k=0}^{N-1} h_k x_k \, exp(-j2\pi kf\Delta) \right|^2 \tag{9}$$

whose expectation is a smoothed version of the true probability density function of the recorded signal depending on the shape of the taper. One of the design by Hann is

$$h_k = \frac{1}{2}\left(1 - cos\left(\frac{2\pi k}{N-1}\right)\right), \qquad k = 0,1, \dots, N-1 \tag{10}$$

Multiples of such taper estimates are constructed as such

$$\begin{aligned} \hat{S}^{mt}(f) &= \frac{1}{L} \sum_{i=1}^{L} \hat{S}^{(i)}(f) \\ \hat{S}^{(i)}(f) &= \Delta \left| \sum_{k=0}^{N-1} h_k^{(i)} x_k \, exp(-j2\pi kf\Delta) \right|^2 \end{aligned} \tag{11}$$

By imposing orthonormality among the tapers the tapered estimates $\hat{S}^{(i)}(f)$ are approximately uncorrelated, and $\mathbb{E}\{\hat{S}^{mt}(f)\} \approx S(f)$. The calculation of the MTM spectral estimator is performed by calling a function in the *Chronux* software package [85].

7.4.2 Principle Components Analysis (PCA)

PCA, as described in Ian's book [86], transforms a set of correlated data into linearly uncorrelated variables called *principle components* (PCs). In addition, the PCs are sorted such that the top PC covers the largest variance within the data, and each subsequent PC covers the largest variance that is orthogonal to the preceding PCs.

The method maps a zero-mean data matrix $\boldsymbol{X}$ of T observations by d dimensions onto a matrix $\boldsymbol{Y}$ of T observations by m dimensions, where $m \leq d$, by applying a linear transformation through matrix multiplication with the weight matrix $\boldsymbol{W}$

$$\boldsymbol{Y} = \boldsymbol{W} \cdot \boldsymbol{X} \tag{12}$$

The weight matrix $\boldsymbol{W}$ is obtained as such.

First, obtain the $d \times d$ covariance of the data matrix $\boldsymbol{X}$**:**

$$\Sigma = \frac{1}{T-1} X^T X \tag{13}$$

Next, calculate the eigenvectors of the covariance matrix which are defined as such

$$\begin{aligned} |\Sigma - \lambda I| &= 0 \\ \Sigma x &= \lambda x \end{aligned} \tag{14}$$

where I is the identity matrix with 1's on the diagonal and 0's elsewhere, λ denotes the eigenvalue of the matrix Σ and x denotes the corresponding eigenvector. We can build a diagonal matrix of eigenvalues and the corresponding eigenvectors:

$$\begin{aligned} \Sigma\Phi &= \Lambda\Phi \\ \Lambda &= \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_d \end{bmatrix} \\ \Phi &= \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,d} \\ x_{2,1} & x_{2,2} & & x_{2,d} \\ \vdots & & \ddots & \vdots \\ x_{d,1} & x_{d,2} & \cdots & x_{d,d} \end{bmatrix} \end{aligned} \tag{15}$$

where λ_i is the *i*th eigenvalue and $x_{i,j}$ is the *j*th element of the *i*th eigenvector.

The eigenvectors can be normalized such that

$$\Phi \cdot \Phi^T = \Phi^T \cdot \Phi = I \tag{16}$$

We can then decompose the covariance matrix C

$$\begin{aligned} \Phi^T \cdot \Sigma \cdot \Phi &= \Lambda \\ \Sigma &= \Phi \cdot \Lambda \cdot \Phi^T \end{aligned} \tag{17}$$

The matrix Φ is also called the PCA scores matrix, and it is an orthonormal matrix where each vector represents the projection of the features onto a new basis. The feature reduction can be realized by choosing the top m eigenvectors.

7.4.3 Correlation Coefficient Squared (CC^2)

Contrary to the unsupervised PCA method, we proposed an alternative supervised feature reduction method that can not only effectively select the most discriminating features, but also helps to understand the inherent properties of the temporal-frequency structures in the LFP data.

Given the target labels, choose any two directions to form a binary classification set:

$$\boldsymbol{Y} = \{(0,1)\}_T \tag{18}$$

where the vector $\boldsymbol{Y}$ of length T has as entry the Boolean value of the trial's direction corresponding to the reference direction. Next, calculate the correlation coefficient between each feature dimension X_d and the vector $\boldsymbol{Y}$:

$$\begin{aligned} \rho_d(X,Y) &= \frac{cov(X_d,Y)}{\sigma_{X_d}\sigma_Y} \\ &= \frac{E\left[\left(X_d-\mu_{X_d}\right)(Y-\mu_Y)\right]}{\sigma_{X_d}\sigma_Y} \end{aligned} \tag{19}$$

where $cov(\cdot)$ denotes the covariance function and μ, σ denote the mean and standard deviation respectively. The expectation function $E(\cdot)$ denotes the sample average. The obtained vector $\boldsymbol{\rho}$ of length can then be squared to obtain the CC^2 score. The motivation is that a feature dimension that is highly correlated ($\rho \to 1$) has the same discriminative property as a feature dimension that is highly anti-correlated ($\rho \to -1$). Finally, the CC^2 can be decreasingly sorted so that the m most discriminative features can be selected.

7.5 Movement Direction Decoding

7.5.1 Logistic Regression (LR)

The logistic regression [50] is based on the logistic function defined as

$$g(z) = \frac{1}{1 + e^{-z}} \tag{20}$$

where z is a linear function of the input vector $\boldsymbol{x} = [1, x_1, x_2, \dots, x_m]$ of $m + 1$ dimensions.

$$z = \theta_0 + \sum_{j=1}^{n} \theta_j x_j = \boldsymbol{\theta}^T \boldsymbol{x} \tag{21}$$

$g(z)$ tends towards 1 as $z \to +\infty$ and 0 as $z \to -\infty$, and is bounded within (0,1).

We can thus rewrite the logistic function with respect to $\boldsymbol{x}$ as

$$h_{\boldsymbol{\theta}}(\boldsymbol{x}) = g(\boldsymbol{\theta}^T \boldsymbol{x}) = \frac{1}{1 + exp(-\boldsymbol{\theta}^T \boldsymbol{x})} \tag{22}$$

$h_{\boldsymbol{\theta}}(\boldsymbol{x})$ defined above is the hypothesis function representing the probability of "success" given $\boldsymbol{x}$ and some parameters θ_j for $0 \leq j \leq m$. The values of θ_j can be predefined or can trained when the true value of the output is known. In this project, we used the optimization technique summarized in the following algorithm.

Let us first define the derivative of $g(z)$

$$\begin{aligned} g'(z) &= \frac{d}{dz}\left(\frac{1}{1 + e^{-z}}\right) = \frac{1}{(1 + e^{-z})^2} e^{-z} \\ &= \frac{1}{1 + e^{-z}}\left(1 - \frac{1}{1 + e^{-z}}\right) \\ &= g(z)\big(1 - g(z)\big) \end{aligned} \tag{23}$$

as well as some notation

$$\begin{aligned} P(y = 1|x; \theta) &= h_\theta(x) \\ P(y = 0|x; \theta) &= 1 - h_\theta(x) \\ p(y|x; \theta) &= \big(h_\theta(x)\big)^y \big(1 - h_\theta(x)\big)^{1-y} \end{aligned} \tag{24}$$

Assuming the instances of $\boldsymbol{x}$ are independent, we can set the likelihood function

$$\begin{aligned} L(\theta) &= p(\boldsymbol{y}|X;\theta) = \prod_{i=1}^{m} p(y_i|x_i;\theta) \\ &= \prod_{i=1}^{m} \big(h_\theta(x_i)\big)^{y_i} \big(1-h_\theta(x_i)\big)^{1-y_i} \end{aligned} \tag{25}$$

or in log-likelihood notation

$$\begin{aligned} \ell(\theta) &= log\, L(\theta) \\ &= \sum_{i=1}^{m} y_i\, log\, h_\theta(x_i) + (1-y_i)\, log\big(1-h_\theta(x_i)\big) \end{aligned} \tag{26}$$

where x_i and y_i are the ith element of X and $\boldsymbol{y}$ respectively, in other words the ith observation. The parameter optimization can now be solved as a likelihood maximization problem with objective function $\ell(\theta)$. The update rule for the parameters θ is defined as

$$\theta \coloneqq \theta + \alpha \nabla_\theta \ell(\theta) \tag{27}$$

where α is the learning rate parameter and the gradient $\nabla_\theta \ell(\theta)$ is defined as

$$\begin{aligned} \frac{\partial}{\partial \theta_j}\ell(\theta) &= \left(y\frac{1}{g(\theta^T x)} - (1-y)\frac{1}{1-g(\theta^T x)}\right)\frac{\partial}{\partial \theta_j} g(\theta^T x) \\ &= \left(y\frac{1}{g(\theta^T x)} - (1-y)\frac{1}{1-g(\theta^T x)}\right) g(\theta^T x)\big(1 \\ &\qquad - g(\theta^T x)\big)\, \frac{\partial}{\partial \theta_j}(\theta^T x) \\ &= \big(y\big(1-g(\theta^T x)\big) - (1-y)\big(g(\theta^T x)\big)x_j \\ &= \big(y - h_\theta(x)\big)x_j \end{aligned} \tag{28}$$

where θ_j, x_j denote jth element respectively. Hence the stochastic gradient *ascent* rule is

$$\theta_j \coloneqq \theta_j + \alpha\big(y_i - h_\theta(x_i)\big)x_{i,j} \tag{29}$$

Note that the update uses the error factor $y_i - h_\theta(x_i)$ which is the difference between the actual value of the output and the activation hypothesis function. We may also optimize the θ parameters differently using other search methods that are available in mathematical

toolboxes such as *fminunc* (unconstrained minimization function built-into MATLAB). To do so, we need to provide the cost function $J(\theta)$ and its derivative $\frac{\partial}{\partial\theta}J(\theta)$.

$$\begin{aligned} J(\theta) &= -\frac{1}{m}\ell(\theta) \\ &= \frac{1}{m}\sum_{i=1}^{m} -y_i \, log\, h_\theta(x_i) - (1-y_i)\, log\big(1-h_\theta(x_i)\big) \\ \frac{\partial J(\theta)}{\partial\theta_j} &= \frac{1}{m}\sum_{i=1}^{m}\big(h_\theta(x_i)-y_i\big)\,(x_i) \end{aligned} \tag{30}$$

It is of interest to avoid overfitting by adding a regularization term λ.

$$\begin{aligned} J(\theta) &= \frac{1}{m}\sum_{i=1}^{m} -y_i \, log\, h_\theta(x_i) - \big(1-y_i\big)\, log\big(1-h_\theta(x_i)\big) + \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2 \\ \frac{\partial J(\theta)}{\partial\theta_j} &= \begin{cases} \left(\frac{1}{m}\sum_{i=1}^{m}\big(h_\theta(x_i)-y_i\big)\,(x_i)\right) & , \quad j=0 \\ \left(\frac{1}{m}\sum_{i=1}^{m}\big(h_\theta(x_i)-y_i\big)\,(x_i)\right)+\frac{\lambda}{m}\theta_j & , \quad j\geq 1 \end{cases} \end{aligned} \tag{31}$$

7.5.2 Artificial Neural Network (ANN)

The artificial neural network refers to a configuration of inter-related neurons where the value of each neuron is a function of the values of the lower layer of neurons as depicted in the figure on the right. Many parameters of this network can be customized to best solve the problem: the number of layers, the number of neurons within each layer, the function mapping between the layers, just to name a few. In this project, we will explore the capability of a specific ANN: a feedforward neural network with one hidden layer. The hypothesis function connecting the layers is set to the same logistic function as described in section 7.5. That is the j^{th} neuron in each layer will have activation value

$$
\begin{aligned}
a_0^{(1)} &= 1 \\
a_j^{(1)} &= x_j \\
a_j^{(2)} &= g\left(\sum_{i=1}^{m} \theta_{i,j}^{(1)} a_i^{(1)}\right) \\
h_\theta(x)_j = a_j^{(3)} &= g\left(\sum_{i=1}^{m} \theta_{i,j}^{(2)} a_i^{(2)}\right)
\end{aligned}
\tag{32}
$$

where,

- $a_0^{(1)}$ defines the bias term, and $a_j^{(1)}$, $a_j^{(2)}$, $a_j^{(3)}$ represent the activation value of the j^{th} neuron from the input layer, the hidden layer, and the output layer respectively. The input layer is set to the values of the input vector $\boldsymbol{x}$, and the output layer defines the hypothesis function $h_\theta(x)_j$.
- $\theta_{i,j}^{(1)}$ and $\theta_{i,j}^{(2)}$ are the weights assigned to the i^{th} neuron from the input layer and the hidden layer respectively
- $g(\cdot)$ is the logistic function

The training process of this network consists of two sets of values: $\theta_{i,j}^{(1)}$ and $\theta_{i,j}^{(2)}$. The optimization procedure is described in the following algorithm. Let us first define the cost function and the regularization term

$$
\begin{aligned}
J(\theta) = \frac{1}{m}\sum_{i=1}^{m}\sum_{k=1}^{K}\Big[& -y_k^{(i)} \log\left(\left(h_\theta\left(x^{(i)}\right)\right)_k\right) \\
& - \left(1 - y_k^{(i)}\right) \log\left(1 - \left(h_\theta\left(x^{(i)}\right)\right)_k\right)\Big] \\
& + \frac{\lambda}{2m}\left[\sum_{j=1}^{m}\sum_{k=1}^{K}\left(\Theta_{j,k}^{(1)}\right)^2 + \sum_{j=1}^{m}\sum_{k=1}^{K}\left(\Theta_{j,k}^{(2)}\right)^2\right]
\end{aligned}
\tag{33}
$$

where m is the total number of instances and K is the total number of output variables. The value of $\left(h_\theta\left(x^{(i)}\right)\right)_k$ is calculated as per equation (8). The parameter variables $\left(\Theta_{j,k}^{(1)}\right)$ and

$\left(\Theta_{j,k}^{(2)}\right)$ denotes the $(j,k)^{th}$ entry of the weights in the first and second layers respectively. Next, let's define error functions

$$\begin{aligned}\delta_k^{(3)} &= \left(a_k^{(3)} - y_k\right) \\ \delta^{(2)} &= \left(\Theta^{(2)}\right)^T \delta^{(3)} .* g'\left(z^{(2)}\right)\end{aligned} \tag{34}$$

where $\delta_k^{(3)}$ is the difference between the hypothesis function and the target value. $\delta^{(2)}$ is the error derived by backpropagation of $\delta_k^{(3)}$ scaled by the weights $\Theta^{(2)}$ and the derivative of the hidden layer activation values. Finally, we need to derive the gradient of the cost function.

$$\begin{aligned}\Delta^{(l)} &= \Delta^{(l)} + \delta^{(l+1)}\left(a^{(l)}\right)^T \\ \frac{\partial}{\partial \Theta_{i,j}^{(l)}} J(\Theta) = D_{ij}^{(l)} &= \begin{cases} \frac{1}{m}\Delta_{ij}^{(l)} & , \quad j = 0 \\ \frac{1}{m}\Delta_{ij}^{(l)} + \frac{\lambda}{m}\Theta_{ij}^{(l)} & , \quad j \geq 1 \end{cases}\end{aligned} \tag{35}$$

7.5.3 Decision Tree Classifier (DTC)

The motivation is to split the input space into subsets recursively that maximizes the prediction accuracy of the target classes, as shown in the figure on the right. Each *node*, including the *root*, corresponds to a specific dimension of the input vector and the *leaf* predicts the class. The most popular method to *grow* the tree is to choose the feature that minimizes the Gini index of the split

$$Gini_{split}(T) = \frac{N_1}{N} gini(T_1) + \frac{N_2}{N} gini(T_2) \tag{36}$$

where T denotes the dataset and T_1, T_2 denote the subsets after the split. N_1, N_2 denote the number of instances found in each subset. The $Gini(T)$ is the Gini index of each subset

$$Gini(T) = \sum_{i=1}^{m} f_i(1 - f_i) = 1 - \sum_{i=1}^{m} f_i^2 \tag{37}$$

where f_i denotes the fraction of items labeled with the value i, and there is a total of m classes. The tree that is grown as such until all features are exhausted is said to be *fully grown*.

Since the learning of such algorithm does not depend on any specified hyper-parameter, the method is said to be non-parameterized. The classification of a new entry is performed by comparing its features against the partitions to finally arrive at the corresponding leaf.

7.5.4 Support Vector Machine (SVM)

Let's first define the linear SVM as it provides the basic formulation of the algorithm. The idea is to partition a p-dimensional space into two subsets using $(p-1)$ hyperplanes. While there are an infinite number of hyperplanes to choose from, the logical choice is to choose the one that separates the two classes of data points with a maximum margin as shown in Figure 5 . The classifier constructed as such is called the *maximum margin classifier*.

Given some training data $\mathcal{D}$, a set of n points of the form

$$\mathcal{D} = \left\{(\boldsymbol{x}_i, y_i) | \boldsymbol{x}_i \in \mathbb{R}^p, y_i \in \{-1,1\}\right\}_{i=1}^{n} \tag{38}$$

where $\boldsymbol{x}_i$ denotes the i^{th} instance of the input vector and y_i denotes the label corresponding to the i^{th} instance. We want to find a hyperplane that maximizes the margin separating the instances with $y_i = 1$ from the instances with $y_i = -1$. Note the different labelling scheme of the target variable between SVM and previous methods.

These hyperplanes are defined by

$$\boldsymbol{w} \cdot \boldsymbol{x} - b = 0 \tag{39}$$

with the margins defined by

$$\boldsymbol{w} \cdot \boldsymbol{x} - b = \pm 1 \tag{40}$$

The distance between the two margin equals $2/\|\boldsymbol{w}\|$. To maximize the margin is to minimize $\|\boldsymbol{w}\|$ while making sure that no instance falls in between the margins. The optimization problem can thus be described

$$arg\min_{\boldsymbol{w},b}\frac{1}{2}\|\boldsymbol{w}\|^2 \tag{41}$$

Subject to

$$y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i - b) \geq 1, \qquad 1 \leq i \leq n \tag{42}$$

or written with Lagrange multipliers (λ):

$$arg\min_{\boldsymbol{w},b}\max_{\lambda\geq 0}\left\{\frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_{i=1}^{n}\lambda_i[y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i - b) - 1]\right\} \tag{43}$$

The Karush–Kuhn–Tucker condition [87] implies that the solution for $\boldsymbol{w}$ can be expressed as a linear combination of the training vectors.

$$\boldsymbol{w} = \sum_{i=1}^{n}\lambda_i y_i \boldsymbol{x}_i \tag{44}$$

However as only the data points positioned on the margin vectors ($y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i - b) = 1$) are defining the hyperplanes we can rewrite the equation

$$\boldsymbol{w}\cdot\boldsymbol{x}_i - b = \frac{1}{y_i} = y_i \Leftrightarrow b = \boldsymbol{w}\cdot\boldsymbol{x}_i - y_i \tag{45}$$

Note that the offset b depends on $\boldsymbol{x}_i$ and y_i, so it varies for each instance. The practice is to take the sample average in order to approximate the statistical mean.

$$b = \frac{1}{N}\sum_{i=1}^{N}(\boldsymbol{w}\cdot\boldsymbol{x}_i - y_i) \tag{46}$$

Now we shall define the *dual form* of the SVM as it reveals insights into the Kernel definition of the method. Using the property that $\|\boldsymbol{x}\|^2 = \boldsymbol{x}^T\cdot\boldsymbol{x}$, and $\boldsymbol{w} = \sum_i \lambda_i y_i \boldsymbol{x}_i$, equation (36) can be rewritten in terms of λ_i

$$\tilde{L}(\lambda_i) = \sum_{i=1}^{n}\lambda_i - \frac{1}{2}\sum_{i,j}\lambda_i\lambda_j y_i y_j \boldsymbol{x}_i^T\boldsymbol{x}_j \tag{47}$$

$$= \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i,j} \lambda_i \lambda_j y_i y_j k(\boldsymbol{x}_i, \boldsymbol{x}_j), \forall \lambda_i \geq 0$$

The kernel function $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is defined as the inner product between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. But it can be defined elsewise, e.g. polynomial and RBF

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \begin{cases} (\boldsymbol{x}_i^T \cdot \boldsymbol{x}_j)^d \\ exp\left(-\gamma \left\| \boldsymbol{x}_i - \boldsymbol{x}_j \right\|^2\right), \qquad \gamma = \frac{1}{2\sigma^2} > 0 \end{cases} \tag{48}$$

where d denotes the degree of the polynomial and σ denotes the width of the RBF kernel. These kernel functions allow the mapping of the input vector onto a higher dimensional space that may better reflect the relationship between the instances. The first definition above is called the polynomial kernel, and the second is called the *radial basis function*, a popular choice in kernelized learning algorithms. The last addition to the SVM method we employed is the introduction of the *soft-margin*. Due to a multitude of factors in the real world data, the separation of the data points is not always possible. Thus, we need to allow certain number of data points to be mislabeled by updating the optimization constraints

$$y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i - b) \geq 1 - \xi_i, \qquad 1 \leq i \leq n \tag{49}$$

where the slack variable ξ_i measure the degree of misclassification of the instance i. The optimization problem is then modified to

$$arg \min_{\boldsymbol{w}, \xi, b} \left\{ \frac{1}{2} \|\boldsymbol{w}\|^2 + C \sum_{i=1}^{n} \xi_i \right\}, \qquad \xi_i \geq 0 \tag{50}$$

where the penalty is controlled by C, a hyper-parameter.

The overall equation can be rewritten with Lagrange multiplier

$$arg \min_{\boldsymbol{w}, b} \max_{\boldsymbol{\alpha} \geq 0} \left\{ \frac{1}{2} \|\boldsymbol{w}\|^2 + C \sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \lambda_i [y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i - b) - 1 + \xi_i] - \sum_{i=1}^{n} \beta_i \xi_i \right\} \tag{51}$$

Interestingly, by recalling the dual form of the optimization problem, the slack variable ξ_i vanishes and the misclassification penalty is parametrized by the constant C.

$$\tilde{L}(\lambda_i) = \sum_{i=1}^{n} \lambda_i - \frac{1}{2}\sum_{i,j} \lambda_i \lambda_j y_i y_j k(\boldsymbol{x}_i, \boldsymbol{x}_j), \qquad 0 \le \lambda_i \le C \tag{52}$$

In the project, we have called upon the highly optimized SVM algorithm provided by MATLAB toolboxes.

7.5.5 Testing and Validation

Due to the small number of training instances, the prediction accuracy for each learning algorithm is determined using a *leave-one-out* cross-validation scheme. In essence, for each instance m_i, train the algorithm with the subset $M - \{m_i\}$ and test the prediction against $\{m_i\}$. If the predicted class is identical to the labeled class, then the prediction was correct.

$$accuracy\_rate = \frac{\#\ correct\ prediction}{\#\ instances}\ \% \tag{53}$$

Such measurement is the primary criterion used to compare the performance of the machine learning methods.

For each learning algorithm, the prediction chance level equals $(100/\#class)\%$. For instance, $chance = 50\%$ for binary classification. Such rate is enforced by selecting the training sets uniformly amongst the available training data. A random shuffling confirms the chance level and it consists of assigning a random label to each training instance and obtain the overall prediction accuracy.

8. Results

In this section, the decoding performance for each of the previously described methods will be presented. First, the feature vector selection process will be shown. Then, the parameter training of each parametrized classifier will be explained. Finally, the overall decoding accuracy will be compared.

8.1 Features Reduction

The vectors in the feature space are largely uncorrelated as shown in . Such property make PCA inappropriate as a feature extraction method. In fact, Figure 6 shows that the principle components are not statistically discriminative towards the classification labels. In such a situation, a more appropriate feature extraction method is to analyse the individual features and compare them against the classification labels to rank the most discriminative features. The correlation coefficient between each feature vector and the classification label is calculated and the squared correlation coefficient (CC^2) value of the entire feature space is ranked to extract the most discriminative indices as shown in Figure 7. The correlation coefficients also allow the visualization of the feature vector mapping across the time and frequency space as shown in Figure 8. Two characteristics can be observed from this last figure. First, the lower frequency bands are maximally correlated $\mathbf{50}\ \boldsymbol{ms}$ after the movement onset in the PRR region and around $\mathbf{0}\ \boldsymbol{ms}$ in the PMd region for both movements. It is also to be noted that reach trials demonstrate correlation around $\mathbf{250}\ \boldsymbol{ms}$ prior to the movement onset in the $(15 - 40\ Hz)$ frequency band, while this band is absent in saccade trials. This result suggests that the LFP signals within PRR and PMd regions contribute to the reach movement preparation and not the saccade preparations.

8.2 Decoding Accuracy

As discussed in the methods section, the decoding accuracy of each algorithm depend on a number of factors. The previous subsection described the feature space dimensionality reduction process. The next step is to train classifiers such that the parameters are optimized for minimizing training error while achieving a high degree of generalization.

8.2.1 Parameter Training

Table 2 displays the effects of varying the regularization parameter λ on the cross-validated decoding accuracy of the LR method. It shows that for λ in the range $(10^{-2} - 10^{2})$, the model's performance is near uniform. The effect of varying λ can be seen in detail in Figure 9 – training and testing error vs. regularization parameter (λ) for three-directional saccade movement decoding over the range of $(10^{-10} - 10^{10})$. The two curves shown in the figure are the training and testing error as calculated per eq. 21. The training error asymptotically approaches zero as the model's complexity increases, which also increases the probability of overfitting the model on the training data, represented by the test error approaching the chance level of 33.33%. The overfitting can be regularized by optimizing the value of λ such that it minimizes the testing error. Such minimization occurs at $\lambda \cong 100$. Similarly, both the training and testing error approaches the chance level as the value of $\lambda \rightarrow \infty$, in which case the model is underfitting the data. In other words, it does not have enough complexity to describe data space.

Table 3 displays the effects of varying the regularization parameter λ on the cross-validated decoding accuracy of the ANN method. It shows that for $\lambda \geq 100$, the model's performance sharply drops to the chance level for all three testing cases. Similar to the LR model, the training and testing error over a wider range is shown in Figure 9 – training and

testing error vs. regularization parameter (λ) for three-directional saccade movement decoding. For $\lambda > 10$, the model is underfitted. Unlike LR, the training error is not asymptotically zero. A possible reason for such observation is the hidden layer's property of *compressing* the information content in input layer. As a result, some information are filtered similarly to the dimensionality reduction process. On the other hand, the testing error is globally lower compared to the logistic regression model, and no trend of overfitting is present as $\lambda \to 0$, although the variance increases. Hence the value of $\lambda \cong 0.1$ ensures a more stable performance of the model although producing the global minimum.

Table 4 displays the effects of varying the width (σ) of the RBF kernel used in the SVM algorithm as described in eq. 41 on p.41. It shows that for $\sigma < 2.15$, the model's performance is near chance level for all three testing cases. Also, for $\sigma > 46.42$, the performance is decreasing monotonously. The reason for the difference in performance is illustrated in Figure 10 (RBF). When $\sigma = 0.1$, the decision boundary is overfitted for both linearly and non-linearly separable data whereas when $\sigma = 10$, the model is underfitted. As comparison, when $\sigma = 1$, the boundary seems to be well fitted. However, the figure only displays the data in two dimensions. The actual performance of the algorithm with respect to the value of width depends on all the dimensions within the training set, and hence the optimal width is determined based on *Table 4*: $\sigma \cong 100$.

Table 5 displays the effects of varying the degree of the polynomial kernel used in the SVM algorithm as described in eq. 41 on p.41. It shows that for $d \geq 4$, the model's performance is near chance level for all three testing cases. Similarly to SVM with RBF kernel, the decision boundary is illustrated in Figure 10 (Polynomial). Note that when $d = 1$, the polynomial kernel is identical to a linear kernel, and thus does not fit well on non-linearly

separable data. On the other hand, a fifth degree polynomial kernel displays indices of data overfitting. Finally, the third degree polynomial approximates the linear separation while providing a more accurate decision boundary than linear kernel on non-linearly separable data. Nevertheless, the LFP data has a much higher dimensionality and the polynomial degree to optimize the decoding accuracy is $d \cong 3$ according to *Table 5*.

8.2.2 Overall Performance

Table 6 and *Table 7* illustrate the effect of varying the dimensionality of the input data on the accuracy for saccade and reach movement direction decoding respectively. The results include the parameterized methods as described previously as well as a non-parameterized method, DTC. The latter method's performance does not depend on any specific hyper-parameter and thus no parameter training was required.

The number of dimensions are exponentially added onto the feature space, according to the PCA values. In other words, the first feature to be used as input is the PCA score that *explains* the highest correlation from the highest ranked CC[2] feature reduction method.

$$\tilde{L}(\lambda_i) = \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i,j} \lambda_i \lambda_j y_i y_j k(\boldsymbol{x}_i, \boldsymbol{x}_j), \qquad 0 \leq \lambda_i \leq C \tag{54}$$

The first interesting result is that a single dimension can produce 100% accuracy for binary saccade decoding algorithms as well as approximately 95% accuracy for binary reach decoding. Such accuracy is unsurprising based on the result shown in Figure 7, i.e. the labels are linearly separable with respect to certain features.

Still referring to the binary movement decoding, the non-parametrized method, i.e. the DTC's performance quickly drops to chance level of 50% as the dimensionality increases to

100 features. On the other hand, the rest of the decoding models' performance are largely unaffected by the introduction of new dimensions. Additionally, these latter models' decoding accuracy displayed higher accuracy for reach decoding with additional features.

The ternary and quaternary decoding, where linearly separation of all labels are not always possible, offer a more in-depth analysis of the models' characteristics. All the models, with the exception of LR, displayed a decrease in decoding accuracy as the dimensionality increases from 20 to 100. A possible explanation for such observation is that the added features are non-descriptive to the target labels. In other words, they behave as noise and the decoders' performance do not scale well to the number of dimensions. In fact, even SVM, which is described as robust against the curse dimensionality, cannot sustain its performance. On the other hand, LR's performance highly robust as compared to the other methods. Such result can be attributed to the regularized property of LR which can assign near zero weights to non-descriptive features.

8.3 Comparative Review

Disregarding the effect of dimensionality, the decoding accuracy of all evaluated machine learning methods will be compared. The saccade movement direction can be decoded with accuracy as high as 100% for binary, 90.60% for ternary (LR), and 82.14% for quaternary (SVM-RBF and LDA) decoding. The reach movement direction can be decoded with accuracy of 99.50% for binary (ANN), 93.01% for ternary (LR and SVM-RBF), and 85.05% for quaternary (LR) decoding. These levels are well above the chance level for each setting. It is hence possible to conclude that the LFP signal, with power spectrum density extracted with the multi-taper method is an accurate predictor to the movement direction for reach and saccade.

9. Discussions

The main objective of this research was to compare the performance of different machine learning algorithms for saccade and reach movement decoding based on local field potential signals. The results show that most algorithms are able to decode the movement direction with an accuracy much higher than the chance level. A particularly well performing algorithm is the logistic regression which displayed robustness against noisy features and a high decoding rate in general.

To our knowledge, this is the first comparative study of machine learning algorithms performed on LFP signal decoding. Nevertheless, there were previous studies on informational continent encoded in various BCI modalities such as SUA and EEG. This section will compare the results we obtained against other groups published results.

9.1 LFP vs. Other BCI Modalities

As shown in Figure 8, the LFP signal in PRR and PMd areas encode movement direction information. Moreover, these regions are tuned with respect to reach movement planning and not to saccade movement planning. Such results were published for neuron's action potential signals [74],[80]. However, the current result confirms that LFP signal also encode movement paradigm specialization among motor cortices. The same figure displayed a discrepancy in time between the PMd and PRR regions which suggest that PMd reacts before PRR. Such results contradicts the signal pathway which maps the output of PRR to the input of PMd [71]. Nevertheless, this time lag is also reported in Stetson's paper [88]. A possible explanation could be that both cortices receive a motor image prior to the movement onset. Taking into account the directional tuning map of LFP signal, our results confirm that LFP

signal encode similar information as the action potentials with respect to movement planning in the two areas studied in this project.

LFP vs. SUA, MUA Furthermore, the decoding accuracy of the LFP signal from this study achieved comparable rate as the reported results from other groups carrying out similar experiments on SUA and MUA signal. In 2007, Stark and Abeles [89] reported an overall six directional decoding accuracy of 76% for SUA and 94% for MUA. The paper also reported results from a LFP implementation and reported 43% accuracy rate. The details of the decoding algorithm were omitted, hence it is beyond our ability to assess their computational optimization. The comparable results between our decoding rate and the SUA based decoding rate are confirmed in an earlier paper by Pesaran *et al*. [90] where they reported 87% accuracy rate for both LFP and SUA based binary reach movement direction decoding. The similar result between these two modalities, according to the authors, is attributed to the coherency between the signals' tuning with respect to the movement direction.

LFP vs. EEG Other non-invasive BCI modalities were sought to solve the same purpose of movement direction decoding and predicting. The most promising method, namely the EEG, is being studied by multiple groups. A paper by Wang and Makeig published in 2009 [66] reported a classification rate of 80.25% for binary movement direction, eye and hand combined, using EEG signals from the human PPC, the same recording cortex as this project. Although their decoding rate is lower than the LFP signal, it is possible ask the users to issue movement command based on multi-trials prediction, thus lowering the error probability as long as the single trials' prediction are mutually independent. Additionally, the group simply decoded with time based EEG signals while leaving out the frequency components. It is

theoretically possible to apply the methods developed in this project on EEG recordings because the signal source of the latter is identical to the LFP signals.

LFP vs. MEG Another group of researchers, Buch *et al.*, took interest in the magneto-encephalography (MEG) based BCI to facilitate stoke patients with limb control rehabilitation. In their paper published in 2008 [91], the group studied the binary decoding rate of grasp type, a similar venture as the reach direction decoding. In fact, the two motor control signals are analyzed together in many studies, e.g. the paper mentioned previously by Wang and Makeig [66]. Their method is to monitor the μ-rhythm ($9 - 12\ Hz$) of the MEG signal recorded from sensorimotor cortex, and compare the time signal's strength to a pre-defined baseline. Most test subjects were able to achieve a 72.48% performance rate on average after training compared to 52.84% on first trial. Since the MEG signal is a fluctuation in magnetic current resulting from neuronal activity [92], as opposed to the electric current in EEG signal, it is reasonable to expect a similar, if not superior result, by applying the methods developed in this project.

LFP vs. ECoG As introduced in the background, ECoG is a recent modality at the heart of interest for many neuroscientists for human studies. One group of researchers, Wang *et al.*, designed an experiment to study online (real-time) prediction of movement direction from ECoG signals of epilepsy patients. The group reported in 2012 [93] to achieved directional prediction with 54.81^{o} error rate on average for an eight targets setup, i.e. 45^{o} separation between targets. Since the group's method is a regression based model, i.e. continuous target space, instead of classification, i.e. discrete target space as in this present project, there is no result available for direct comparison of classification rate. Nevertheless, it is reasonable to assert that Wang *et al.* achieved comparable result as this experiment by

converting the metric to classification rate: $(360 - 54.81)^o/360^o = 84.78\%$ octonary. The group selected a SVM based decoding algorithm and time and frequency decomposition method.

LFP vs. ALL, a trend can be observed by examination of the above state-of-the-art methods. As the signal's recording radius becomes broader, more noise are introduced and thus the decoding rate is lowered, with the exception of MUA. The high decoding rate of MUA can be attributed to the digital encoding of the signal in the form of action potentials as oppose to the other signals which are all in analog form. Despite SUA is also decoding from action potentials, the source signal is generated from a single neuron, thus does not provide as much feature as population neurons.

9.2 Machine Learning Methods

This subsection discusses the implementation of the various machine learning algorithm studied in this project. It is important to note that the performance of each algorithm depends on various aspects such data structure, parameter selection. More importantly, the computational and time complexity of each algorithm depends heavily on the software implementation. Considering these aspects, the decoding rate reported in this project are subject to changes if the features were extracted differently, or if the hyper-parameter optimization were performed alternatively. Another important characteristic to consider when choosing the decoding algorithm is the scalability of the method. As the amount of data grows exponentially, the process becomes impossible to fit on a single computer's active memory. Therefore it is of interest to choose an algorithm that can be paralleled and take advantage of the clustered computing paradigm such as MapReduce [94].

On the other hand, the computing time should be online realizable to provide the user with the most natural user-experience.

LR The logistic regression algorithm displayed the highest robustness against the feature space dimension and the amount of noisy features. It is also the method offering the clearest human interpretation such as feature importance (weight parameters) and prediction error (probabilistic interpretation). Most importantly, the algorithm can easily altered to take advantage of the MapReduce architecture. Indeed, the weights of each instance can be learned in parallel and all the weights can be accumulated afterwards. Such characteristic also facilitates the weight update method if an online implementation of the decoder is desired. Otherwise, if training and decoding are to be performed separately, the weights parameter can be learned offline with extensive complexity and stored onto the decoder for fast online decoding. Nonetheless, the performance of the algorithm depends heavily on a single hyper-parameter, the regulation term, which in turn depends on the data structure. Thus to achieve true robustness, the algorithm must be able implemented with adaptive regularization term updating methods.

ANN The artificial neural network implemented in this project is based on the backpropagation network with a single hidden layer as described in the methods section. While the results showed less than optimal performance, the network structure is believed to be the deciding factor. Indeed, if the network is implemented with a hidden layer of neurons whose activation values are in a one-to-one relationship with the output layer, then the ANN is functionally identical to the LR, and thus should offer identical performance. Following this reasoning, the decoding rate of ANN should only be greater than or equal to LR. A possible explanation for the inferior performance is the effect of the hidden layer size

and the value of the regularization term, whose value depends on the hidden layer size. Nevertheless, the ANN is a very adaptable structure, where each aspect of the algorithm may be substituted by a counterpart if doing so can improve the performance. An example of such modularity is the possibility to substitute the sigmoid function linking the hidden layer to the output layer by a rectilinear function or a radial basis function. One important remark is that one should not blindly increase the number of layers or the number of neurons within each layer in hope of improving the performance. Doing so can easily overfit the model on the training data and deteriorate the generalization.

SVM The support vector machine algorithm implemented in this study displayed poor result and inconsistent robustness against feature dimensionality for all three kernels (RBF, linear, and polynomial). The poor decoding rate can be attributed to the unfitness of the method on the data. Indeed SVM with RBF and polynomial kernels did not produce higher decoding rate than simple linear kernel for the given feature space. The main cause of this result is the way feature reduction was conducted, i.e. ranking of features as a function correlation coefficient between feature dimension and target label. The features selected as such are linearly separable and thus promote kernels to behave linearly. On the other hand, the poor performance against dimensionality contradicts the description of SVM in the literature. A possible explanation for such result is the fashion that SVM decodes the data. In fact, SVM fits a hyperplane through all the given feature dimensions, which are projected onto hyperspace through kernel functions, and is thus inevitably affected by noisy dimensions. Figure 12 illustrates this hypothesis with fictitious data where only one dimension is a valid predictor for the classification label and the additional dimensions are simply noise. In such situation, the performance of SVM algorithm decreases by

approximately 20% for every 10-fold of noisy data. The decoding rate approaches chance level after 1000 noisy dimensions have been added. As a comparison, decision tree classifiers were able to decode with perfect rate given that there is one dimension with perfect predictability of the target label.

DTC The decision tree classifier is a non-parameterized method as described in the methods section. Such a trait allows the algorithm to perform independently of any hyper-parameter optimization. The disadvantage is that the method is more prone to overfitting. The result from this study shows that DTC has the lowest performance amongst all the methods. Indeed, its decoding rate approaches chance level after the least number of dimensions, which seemingly contradicts the result shown in Figure 12. Nevertheless, it is to be noted that unlike the fictitious data, the features of LFP signal do not guarantee linear separability. Thus, the features must be examined jointly to increase the decoding rate. DTC, on the other hand, follows a hierarchical decision structure which limits the interaction among the feature space.

10. Conclusions and Future Research

This thesis presented a set of machine learning algorithms that decode the reach and saccade movement directions based on the local field potential signals of the monkey MIP and PMd brain areas. This final section will review the contributions of this project and discuss the possible directions for future research.

10.1 Contributions

In this project, a few machine learning algorithms were studied for their decoding accuracy with respect to the algorithms' setup and parameter selections. Furthermore, in order to achieve the best accuracy, a novel feature extraction technique using the multi-taper method was introduced. These results can be used by fellow researchers to better understand the decoding of brain signals or other rhythmic signals in general.

MIP and PMd – As the results confirm, signals acquired from these two brain areas offer enough information to decode the reach and saccade movement directions as well as reach movement preparation.

Decoding – The decoding accuracies presented in the results section demonstrate a high decoding rate from all the algorithms studied in this project. Tables 6 and 7 present the decoding rate with respect to the hyper-parameter selection. Fellow researchers can refer to such results to initialize their parameter search algorithm.

Feature extraction – This project introduced the use of multi-taper method as a feature extraction method which serves to provide a spectral density estimation. This method proved several advantages over standard techniques. It effectively attenuate the noise within the signal.

10.2 Future Research

This project only studied a selection of well known machine learning techniques. The promising results suggest that further research is required to design better algorithms which can yield higher decoding accuracy in general. Further, the methods presented in this project may produce interesting results when applied to other BCI modalities such as EEG or ECoG. Lastly, the signal analysis were performed solely in an offline setting; a practical BCI will require online decoding capabilities.

Multi-layer ANN – The artificial neural network studied in this project only includes a single hidden layer with varying number of neurons. The decoding rate of such method only yields comparable result logistic regression, which can be regarded as a neural network without the hidden layer. Due to the restraints of training samples and computing power, a more elaborate neural network was not tested in this project. It is of interest to see how much improvement, if any, can addition hidden layers bring to the decoder. It is also interesting to study how alternate neural network setup, such as recurrent neural network or the long short-term memory (LSTM), perform with the given neural data.

DTC with pruning – In this project, the decision tree classifiers were allowed to be grown without pruning. Such a model has the advantage of not requiring any explicit hyper-parameter. Nevertheless, the result show that decoding accuracy drop drastically as the number of features increases. A standard technique to regularize the algorithm by limiting the number of decision nodes with a hyper-parameter. The advantage of such regularization is that overfitting can be restricted at the cost of the need to search for a parameter value. A derivative of the DTC is the Random Forest algorithm. The latter two algorithms can be the subject of further research.

Other ML methods – There are countless published and proven machine learning algorithms available for studying the decoding analysis. The set presented in this project are chosen based on the available documentation.

Other BCI modalities – As discussed in section 6.2, LFP is a rhythmic brain signal who shares the common signal source with EEG and ECoG signals. The latter two are measured at a location closer to the scalp. Nevertheless, these two less invasive modalities encode signals in the frequency space just like LFP. It is therefore interesting to see if the same methods studied in this project are applicable to EEG and ECoG, and to what extent.

Online decoding – The study is conducted based on recorded brain signals which are analyzed in an offline environment; that is each trial is regarded as a unique entity. In order for the decoder to be useful as a real world BCI, it is important that the algorithm can be adapted to an online environment; that is predictions are generated as the signals are acquired. An important challenge in realizing this feature is how to label the classes. As the subject conduct the experiment, the brain processes myriad other motor planning and execution commands. It is thus uneasy to categorize the brain signals with respect to solely the eye or hand movement. A possible solution is implement a regression-based decoder that yield prediction in a continuous space, e.g. the two or three-dimensional velocity of the reach or saccade movement. Another proposal is to realize an unsupervised learning method which continuously process the brain signal and attempt to categorize the mind state. Such a method could yield unexpected results that can introduce new research orientations.

11. Figures

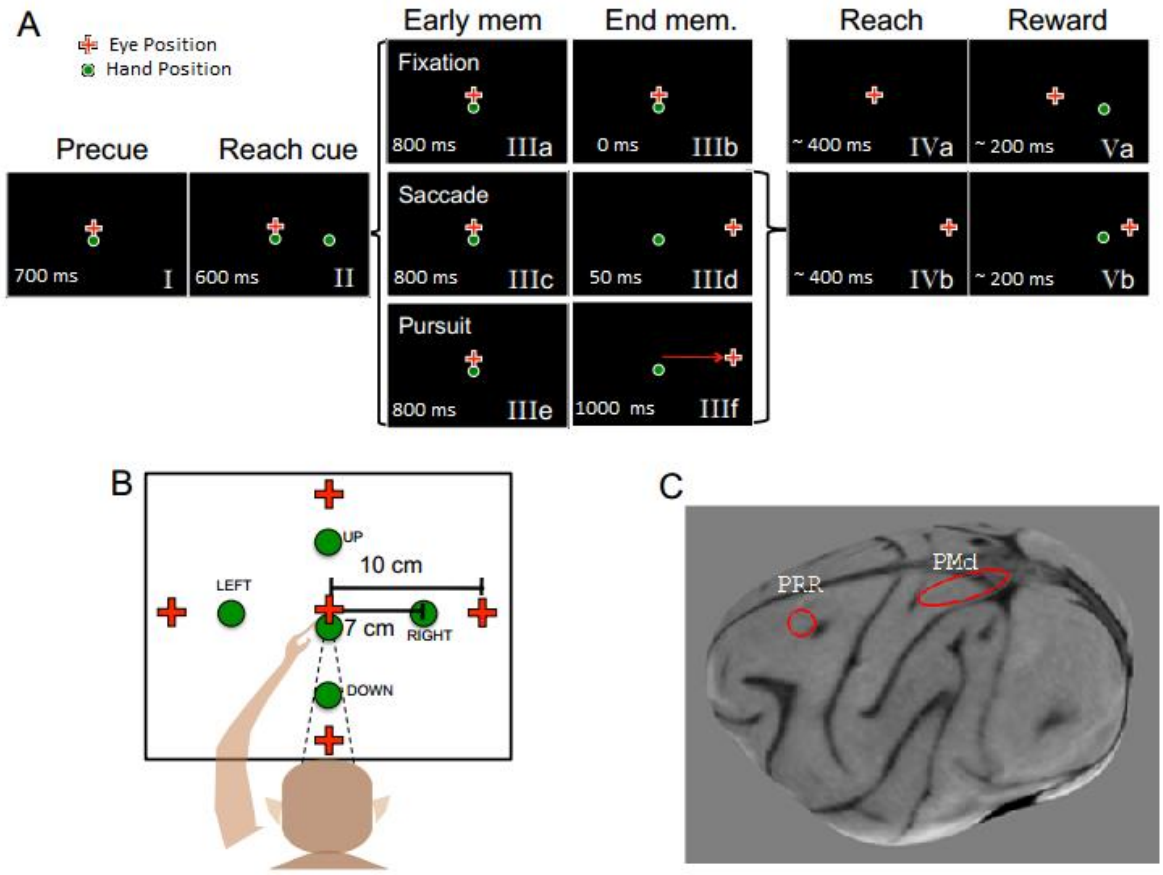

Figure 1 - Experimental Setup. **A** Time laps of delayed memory reach task. (I) The initial position of gaze and hand cues are shown for 700 *ms*. (II) The reach cue is shown among one of the four targets for 600 *ms*. (IIIa,c,e) During the early memory period lasting 800 *ms*, the reach cue is extinguished. (IIIb) For fixation condition, no eye movement cue is shown. (IIId) For saccade condition, the eye cue at one of the four targets is lit promptly and the subject shifts its gaze in a saccadic eye movement to the target; the action takes less than 50 *ms*. (IIIf) For pursuit condition, the eye cue moves linearly towards the final location and the subject visually follows the target; the duration of the task is approximately 1000 *ms*. (IVa,b) The hand cue is extinguished which notifies the subject to start the reach action. (Va,b) The reward is administered if the hand successfully touches the memorized reach target. **B** The figure shows the relative location of the cues and the monkey. **C** The figures indicates the location of the electrode implantation, namely in the PRR and PMd areas.

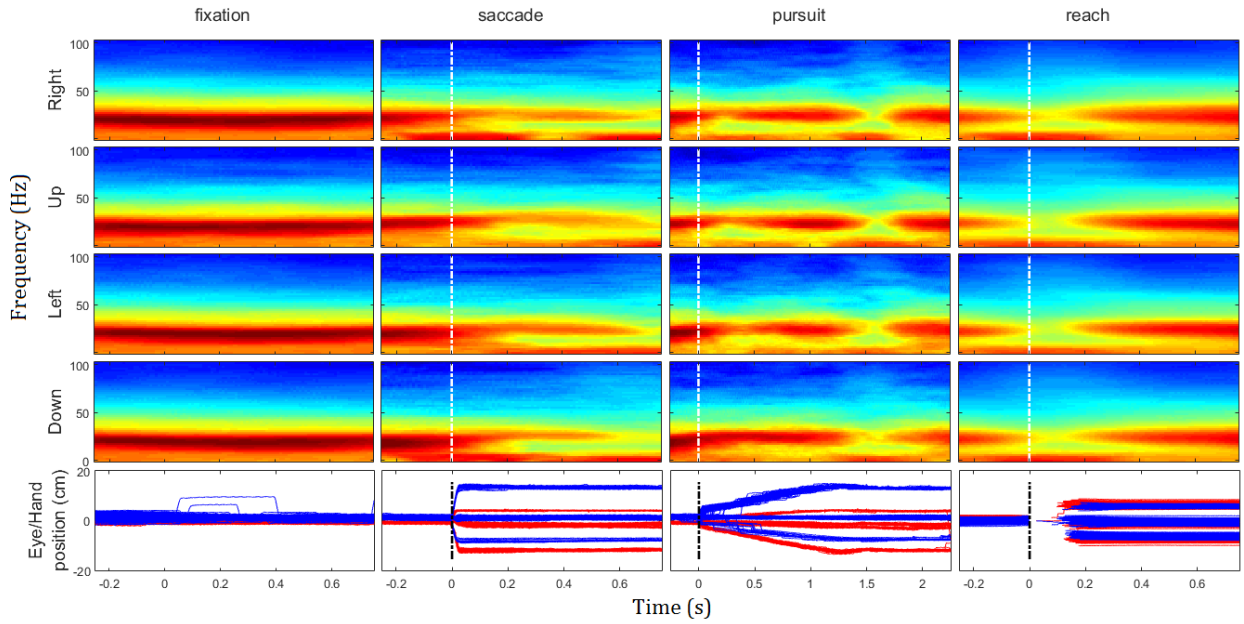

Figure 2 – PSD of sample channel in PRR. The figure shows a reconstruction of the power spectral density (PSD) of LFP from a sample channel in the PRR area. Each column represents a specific condition from left to right: fixation, saccade, pursuit, and reach task. Each row corresponds to the movement target from top to bottom: right, up, left, and down. The rows of fixation trials are separated according to the reach direction of the trial. The last row displays the eye or hand position recorded with the apparatus mentioned in the Materials and Methods section. The onset alignment is indicated by the dotted line at $T = 0$. As shown in the figure, there exists some dynamics in the frequency bands that are coupled with the eye or hand movement onset as compared to the relatively uniform PSD during the fixation epoch. Further, there are visible separations across the frequency distribution: $(0 - 10), (10 - 40), (40 - 100)\ Hz$. Notice the change in power in saccade trials around the 700 ms and in pursuit trials around 1500 ms are overlapped with reach movement onset.

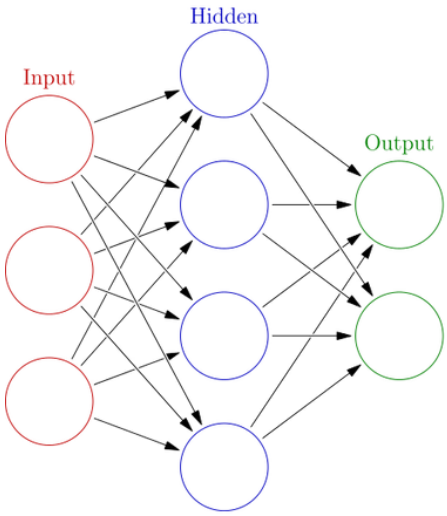

Figure 3 – Artificial Neural Network [95]. The figure illustrates the internal structure of the artificial neural network. Each dimension of the feature vector corresponds to a neuron, or node of the input layer. These neurons propagate the values to the hidden layer, whose size is arbitrary and can be optimized through parameter searching. Finally, the hidden layer's activation values propagate to the output layer, whose value equals to the classification label. The propagation algorithm is a linear combination of the weights, represented by the arrows, and the activation value of each node from the lower layer. The inner product is then mapped to the range (0,1) using the sigmoid function. Thus, the training involves optimizing the weight parameters, subject to regularization, that maximize the classification accuracy.

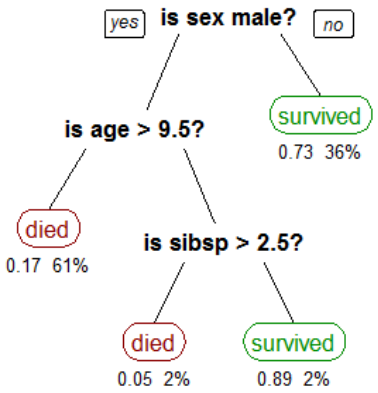

Figure 4 – Decision Tree Classifier [96]. The figure illustrates the decision tree classifier that may help in predicting the survival of a passenger aboard the Titanic. To obtain a prediction,

one traverses the tree from top to bottom and choose subsequent node according to the decision criteria. In the case of direction decoding using LFP signal, each node can represent the spectral power of a given time and frequency. The DTC can be trained such that the top node divides the trials with the greatest information gain based on the Gini factor. Each subsequent node can be trained on the residual features recursively using the same algorithm until the exhaustion of the feature space. The decision is produced at the bottom layer, also called the leaves, which are labeled as the classification targets.

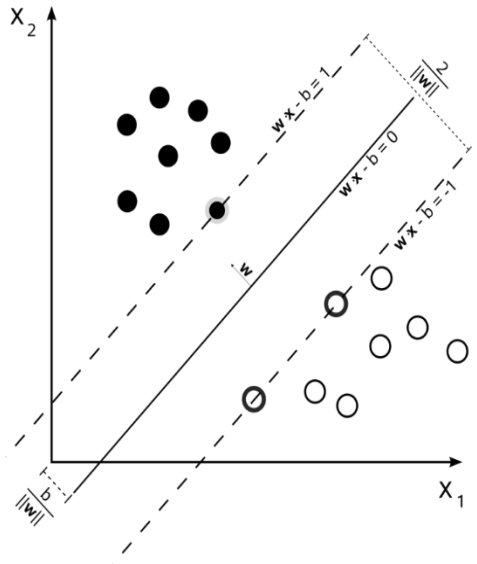

Figure 5 – Support Vector Machine [97]. The figure illustrates the SVM method. The band between the lines $\boldsymbol{wx} - b \pm 1$ represents the decision boundary. The training algorithm is to maximize the width of such band, or hyper-plane in the case of multidimensional data. However, the data cannot always be separated into such clusters. Thus, the soft-margin SVM allow the presence of mislabeled data on each side of the decision boundary controlled by a regularization term. In addition to the linear kernel shown in the figure, one can in fact select other kernel function that map the features onto higher dimensions, where the separability is increased. Examples of popular kernels are the radial basis function (RBF) and polynomials.

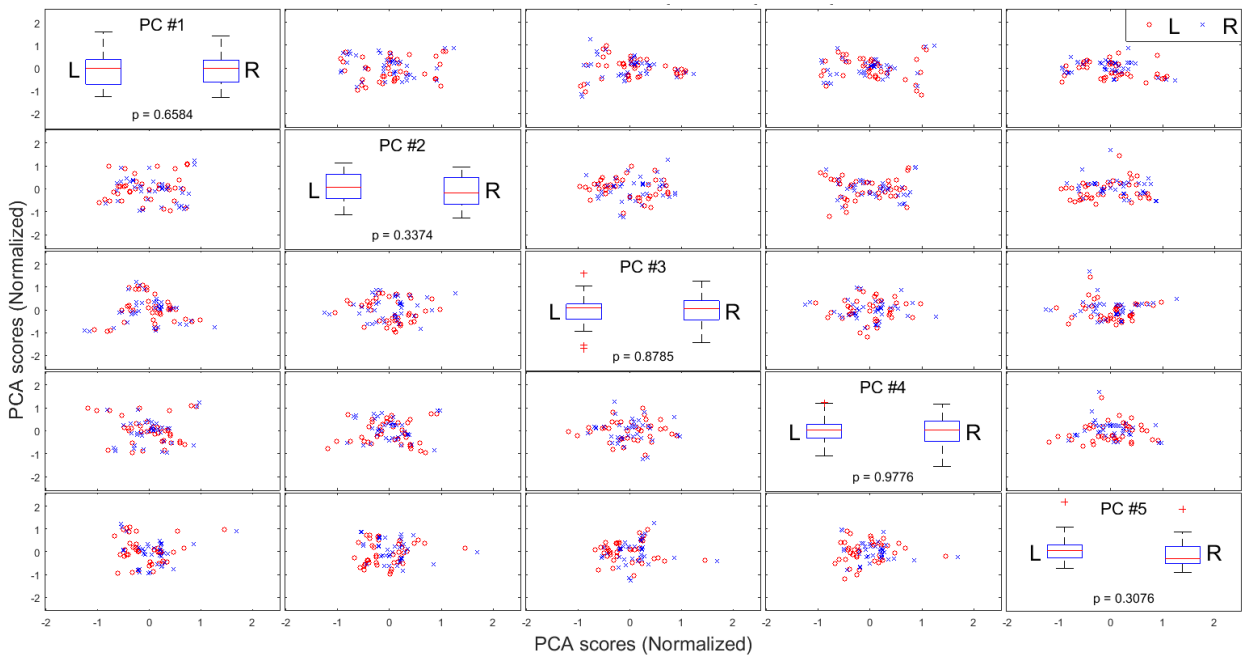

Figure 6 – Scatter Plot and Box Plot of Top 5 Principle Components. The figure illustrate the relationship among the top five principle components (PC), i.e. the dimensions that capture the most variances among the feature space. The boxplots along the diagonal illustrate the mean and variance of the data points within each PC; the data is separated by the movement target, (L)eft and (R)ight. It can be seen that the PC shows a poor separability with the lowest p-value of 0.3076. The scatter plots show the two-dimensional relationship between each pair of PC. As shown, the two target labels are highly homogeneous. The effect of such result is a poor decoding performance.

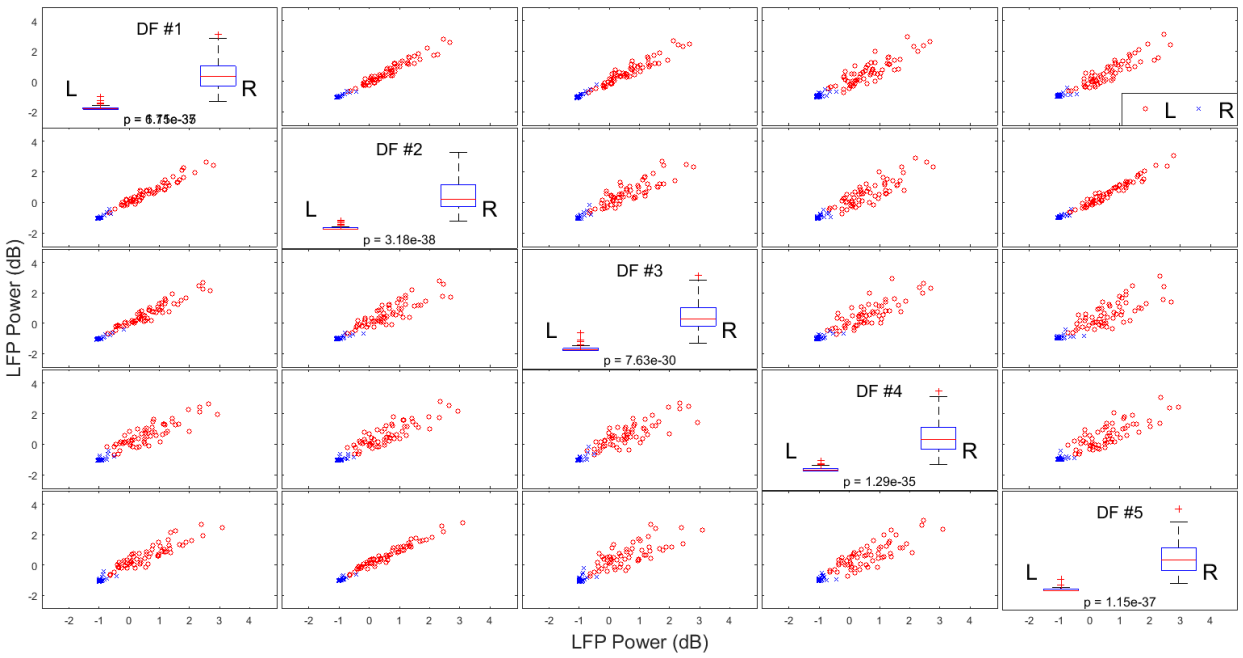

Figure 7 – Scatter Plot and Box Plot of Top 5 Discriminative Features. The figure illustrate the relationship among the top five discriminative features (DF), i.e. the dimensions that are the most (anti-)correlated with the target label. The boxplots along the diagonal illustrate the mean and variance of the data points within each DF; the data is separated by the movement target, (L)eft and (R)ight. It can be seen that the DF shows a high separability with p-values less than 10^{-30}. The scatter plots show the two-dimensional relationship between each pair of PC. As shown, the two target labels are highly separable. The effect of such result is a high decoding performance even with linear decoders. Note the correlation with the top DF. One may apply PCA to the top DF to further reduce the dimensionality.

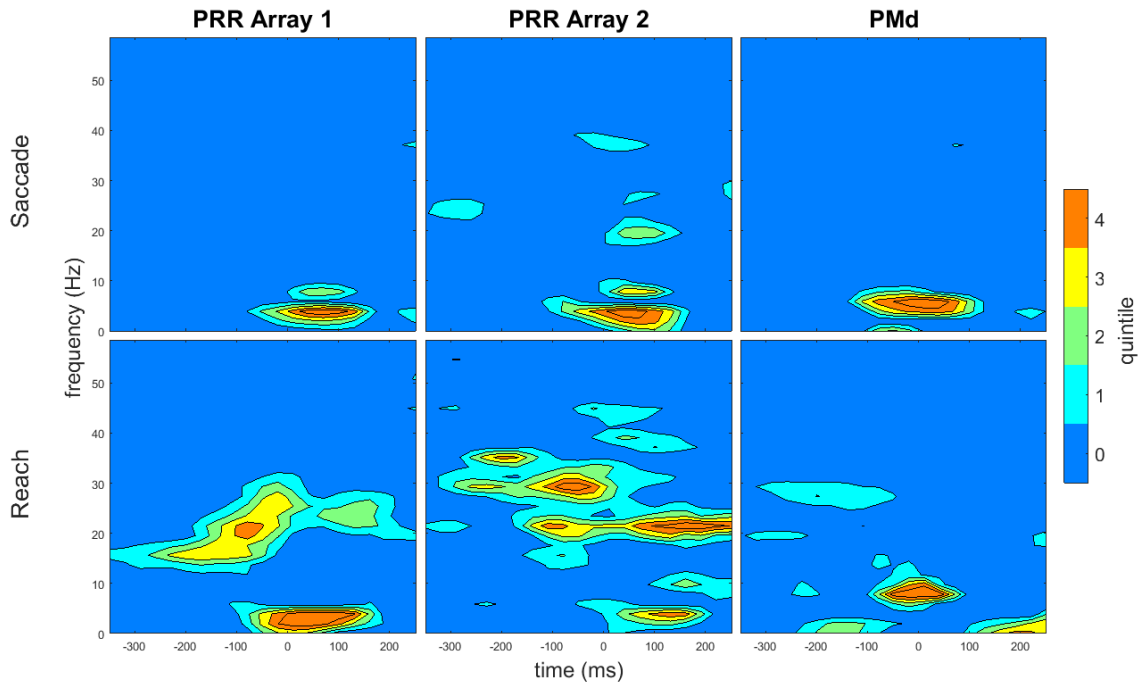

Figure 8 – Time and frequency map of squared correlation coefficient between each feature and the target label by electrode array. The figure illustrate the map in time (x-axis) and frequency (y-axis) space. The color scale represents the squared correlation coefficient (CC-2) divided into quintiles. First, the lower frequency bands are maximally correlated $\mathbf{50}\ \boldsymbol{ms}$ after the movement onset in the PRR region and around $\mathbf{0}\ \boldsymbol{ms}$ in the PMd region. The discrepancy in time suggests that PMd regions react before PRR. Second, reach trials demonstrate correlation around $\mathbf{250}\ \boldsymbol{ms}$ prior to the movement onset in the $(15 - 40\ Hz)$ frequency band, while this band is absent in saccade trials. This result suggests that the LFP signals within PRR and PMd regions contribute to the reach movement preparation and not the saccade preparations.

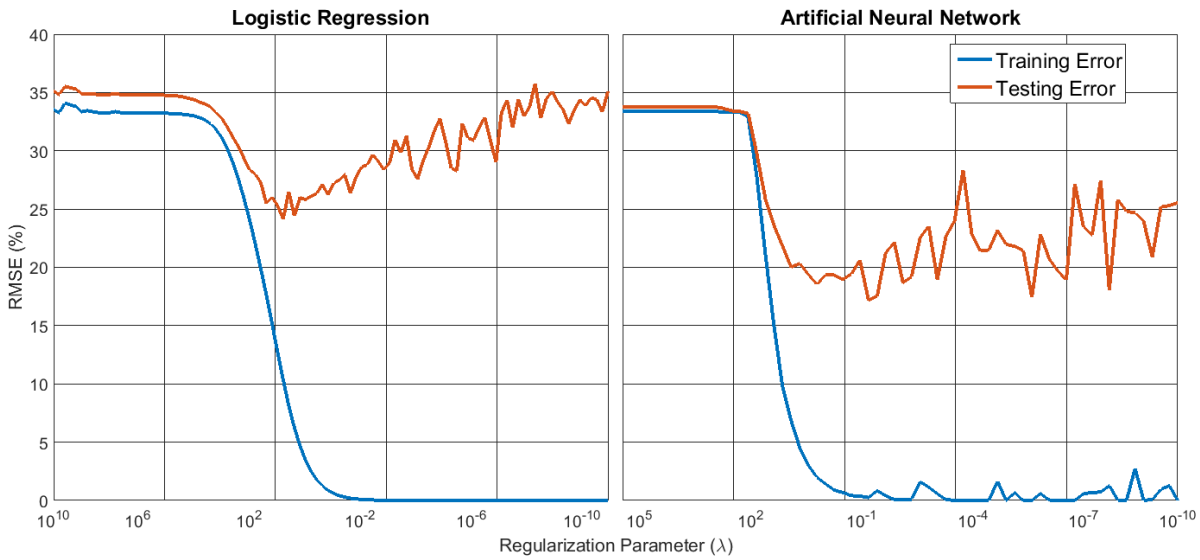

Figure 9 – training and testing error vs. regularization parameter (λ) for three-directional saccade movement decoding. The figure on the left displays the error diagram of the logistic regression method and the figure on the right, the artificial neural network implemented with a hidden layer of 10 neurons. The LR's training error asymptotically approaches zero as λ decreases while the testing error increases back to chance level. Such trend suggests the presence of overfitting. $\lambda \cong 100$ is hence the optimal value for the LR model. For the ANN model, no trend of overfitting is present however the variance increases as λ decrease towards zero. $\lambda \cong 0.1$ is hence the optimal value for regularization which locally minimizes the testing error as well as minimizing the variance and thus improve stability.

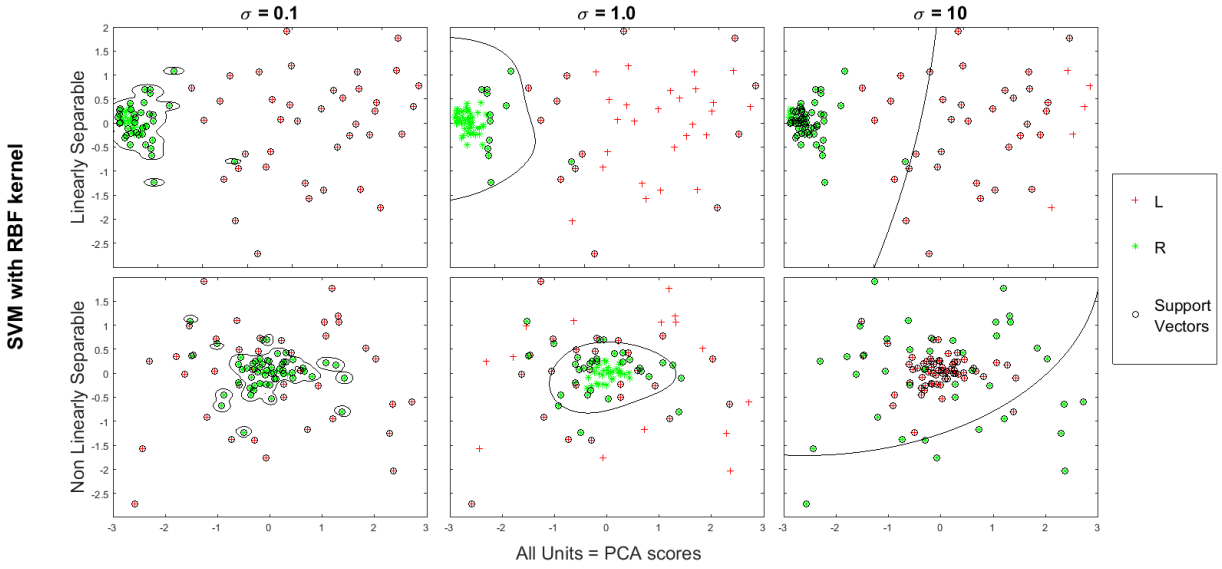

Figure 10 – SVM RBF kernel illustration. The RBF kernel's decision boundary depends on the width parameter σ. Too small of a width forces a narrow margin around the data points and thus overfitting the model on the training data as illustrated by $\sigma = 0.1$, while too large of a width will set the margins to be unrepresentative of the data and thus underfitting the features as illustrated by $\sigma = 10$. Finding the optimal width depends on the value of the data set, and thus the process involves a heuristic parameter searching. Other than the goodness of fit, the dataset's information content can be encoded in higher dimensions than linearly separable features, as shown in the bottom row. Similar to the top row, the model is overfitting when the width is narrow: the decision boundary is forming around the isolated datapoints. On the other hand, the model is underfitting with a large width: the model cannot model the labels closer to the center. An optimal width is illustrated in the middle where the model contains enough complexity to locate the central data points, while maintaining a simplicity to achieve generalization.

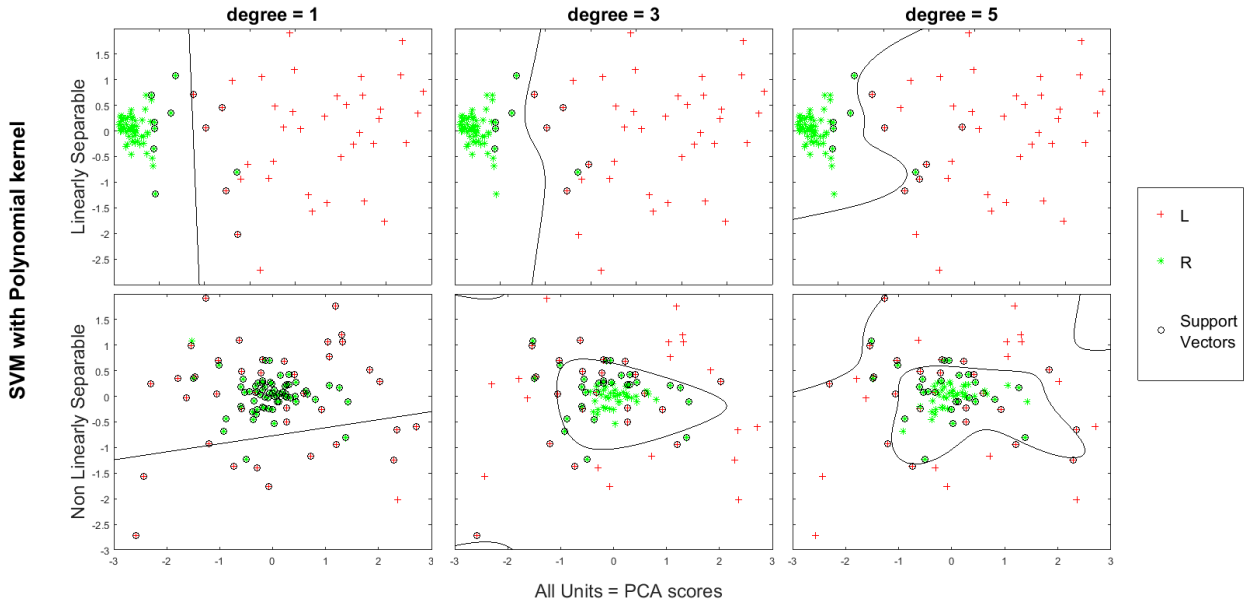

Figure 11 – SVM polynomial kernel illustration. The polynomial kernel's decision boundary depends on the degree of the polynomial. The model is identical to linear kernel with degree equals one and the model becomes more overfitting on the training data for increasing degree. This figure illustrates the effect of the polynomial degree on the decision boundary. While the linear kernel can describe the linearly separable features with a good accuracy, it cannot model the more complex case of higher dimensional data as shown on the bottom row. On the other hand, with an increasing degree of polynomial, the decision boundary can overfit for both separability cases.

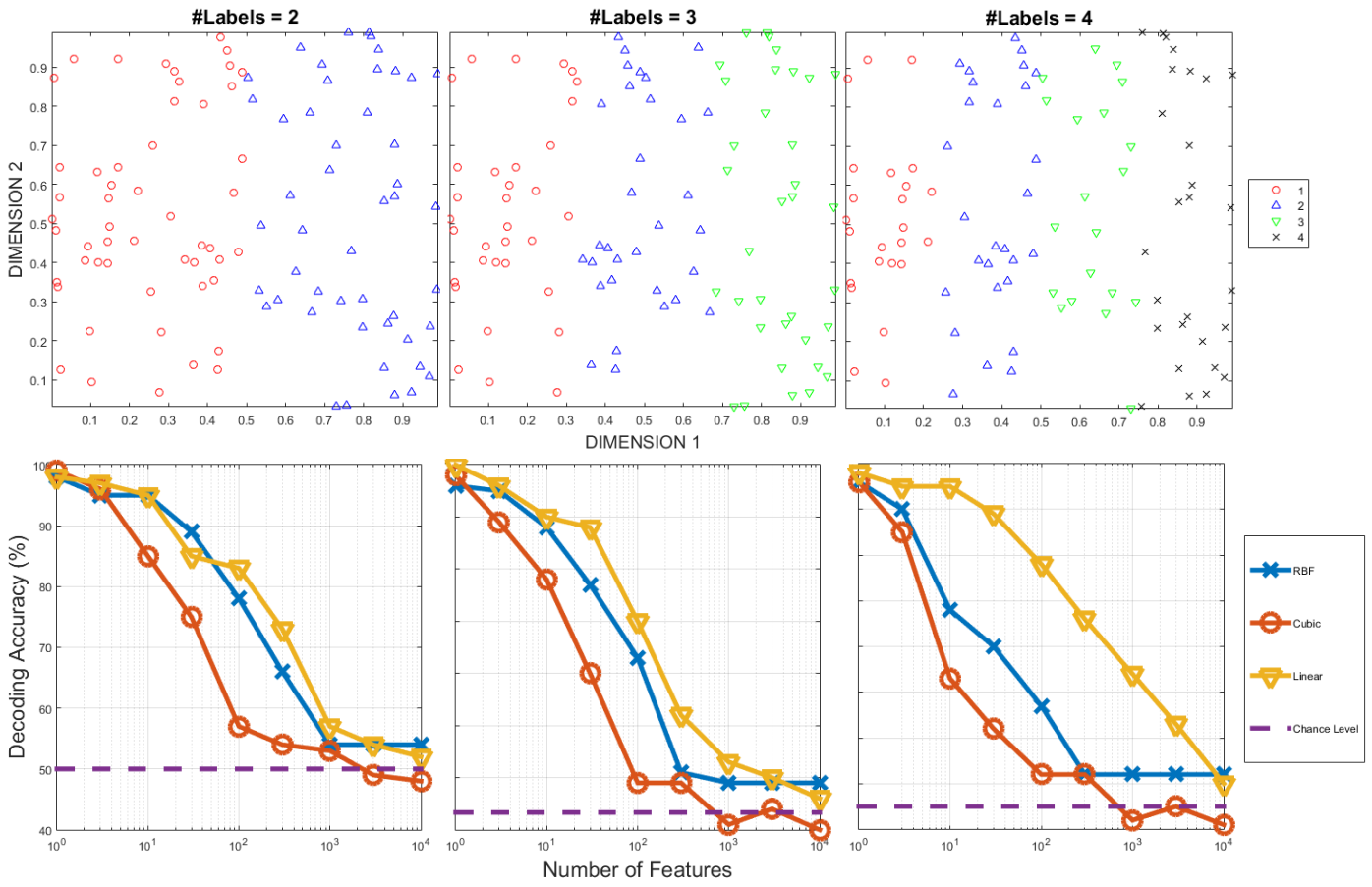

Figure 12 – (top) Fictitious data where the first dimension is a perfect linear predictor to the label and the rest of the dimensions are noise. (bottom) Decoding rate of SVM method with different kernels vs. number of dimensions of the fictitious data. The number of noisy features greatly affect the decoding performance of SVM. In fact, when there is only one dimension with discriminative data, the SVM's performance decreases at a rate of approximately $20\%/decade$.

12. Tables

Table 1 - Mean and Variance of Cross-Correlation from all Feature Vectors by Array and Number of Directions

Number of Directions	Microelectrode Array #1	Microelectrode Micro Array #2	Microelectrode Array #3	All Electrodes
2	0.0096 ± 0.0150	0.0049 ± 0.0147	0.0051 ± 0.0141	0.0050 ± 0.0131
3	0.0086 ± 0.0105	0.0054 ± 0.0105	0.0048 ± 0.0097	0.0043 ± 0.0095
4	0.0078 ± 0.0202	0.0065 ± 0.0203	0.0038 ± 0.0191	0.0027 ± 0.0183

Table 2 – Effects of varying LR regularization parameter (λ) on decoding accuracy

λ (%)	0.01	0.03	0.08	0.22	0.60	1.67	4.64	12.92	35.94	100.00
2	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00
3	87.18	86.32	85.47	86.32	87.18	87.18	85.47	83.76	85.47	84.62
4	71.43	69.64	71.43	71.43	71.43	71.43	78.57	76.79	76.79	73.21

Table 3– Effects of varying ANN regularization parameter (λ) on decoding accuracy

λ (%)	0.01	0.03	0.08	0.22	0.60	1.67	4.64	12.92	35.94	100.00
2	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	51.28
3	90.60	88.89	87.18	88.89	90.60	88.03	86.32	88.03	69.23	33.33
4	82.14	76.79	82.14	80.36	78.57	75.00	75.00	55.36	39.29	23.21

Table 4 – Effects of varying SVM RBF Kernel width (σ) on decoding accuracy

σ (%)	0.10	0.22	0.46	1.00	2.15	4.64	10.00	21.54	46.42	100.00
2	50.00	48.72	48.72	48.72	48.72	84.62	100.00	100.00	100.00	100.00
3	33.33	33.33	33.33	33.33	33.33	53.85	86.32	82.05	70.09	67.52
4	25.00	25.00	25.00	25.00	25.00	25.00	71.43	67.86	50.00	48.21

Table 5 – Effects of varying SVM Polynomial degree on decoding accuracy

degree (%)	1	2	3	4	5
2	100.00	100.00	100.00	60.26	50.00
3	65.81	79.49	83.76	46.15	35.90
4	44.64	67.86	76.79	37.50	26.79

Table 6 – Comprehensive Saccade Movement Direction Decoding Accuracy

	#DIR	CROSS-VALIDATED DECODING ACCURACY (%) VS. NUMBER OF FEATURE VECTORS						
		1	2	5	10	20	50	100
TREE	2	100.00	100.00	100.00	97.44	91.03	53.85	48.72
	3	76.92	80.34	82.05	82.91	70.09	36.75	33.33
	4	66.07	71.43	76.79	78.57	44.64	25.00	19.64
SVM RBF ($\sigma = 1.0$)	2	100.00	100.00	100.00	100.00	98.72	92.31	100.00
	3	76.92	83.76	88.89	88.89	87.18	76.07	41.88
	4	60.71	57.14	82.14	80.36	69.64	25.00	25.00
SVM LINEAR KERNEL	2	100.00	100.00	100.00	100.00	100.00	94.87	92.31
	3	74.36	84.62	86.33	88.89	86.33	82.05	40.17
	4	57.14	69.64	71.43	62.50	64.29	25.00	25.00
SVM POLYNOMIAL (D=3)	2	100.00	100.00	100.00	100.00	100.00	100.00	100.00
	3	71.80	71.80	87.18	88.89	87.18	81.20	81.20
	4	55.36	62.50	62.50	66.07	67.86	69.64	67.86
LOGISTIC REGRESSION ($\lambda = 1.0$)	2	100.00	100.00	100.00	100.00	100.00	100.00	100.00
	3	78.63	82.91	87.18	88.03	88.03	89.74	90.60
	4	60.71	60.71	73.21	73.21	73.21	75.00	76.79
ARTIFICIAL NEURAL NETWORK ($N = 10$)	2	100	100	100	100	100	92.308	89.744
	3	73.504	84.615	88.034	84.615	86.325	84.615	58.974
	4	50	60.714	76.786	73.214	67.857	46.429	28.571

Table 7 – Comprehensive Reach Movement Direction Decoding Accuracy

	#DIR	CROSS-VALIDATED DECODING ACCURACY (%) VS. NUMBER OF FEATURE VECTORS						
		1	2	5	10	20	50	100
TREE	2	95.46	94.95	84.85	94.95	92.93	53.54	50.00
	3	65.05	79.03	69.36	75.81	70.43	48.39	42.47
	4	47.98	57.26	55.65	59.68	65.73	40.73	25.00
SVM RBF ($\sigma = 1.0$)	2	94.95	94.95	92.93	98.99	96.97	98.99	94.44
	3	61.29	81.72	81.72	82.26	93.01	88.71	82.26
	4	47.98	56.86	65.32	65.73	74.19	76.21	78.63
SVM LINEAR KERNEL	2	94.95	94.95	93.43	97.48	98.99	98.99	94.95
	3	65.59	83.33	79.03	78.50	88.71	88.17	66.13
	4	47.98	56.05	56.86	67.74	74.19	76.61	55.24
SVM POLYNOMIAL (D=3)	2	91.41	90.91	91.41	94.95	92.42	90.40	90.40
	3	61.83	81.18	82.26	82.80	83.33	82.26	77.96
	4	41.53	55.65	54.84	54.84	53.63	58.87	52.42
LOGISTIC REGRESSION ($\lambda = 1.0$)	2	94.95	93.43	92.93	97.48	98.49	98.99	98.49
	3	61.29	80.11	80.11	84.41	92.47	93.01	92.47
	4	45.57	59.68	62.10	68.55	79.03	83.47	85.08
ARTIFICIAL NEURAL NETWORK ($N = 10$)	2	96.47	96.47	95.46	96.47	97.98	99.50	95.46
	3	63.98	80.65	80.11	86.02	89.79	88.17	67.74
	4	45.97	58.07	60.48	63.31	68.95	78.23	76.21

13. References

[1] Reeve Foundation, "One degree of separation: paralysis and spinal cord injury in the United States," *Christopher Dana Reeve Paralys. Resour. Cent.*, pp. 1–28, 2009.

[2] Statistics Canada, "Table 051-0001 Estimates of population, by age group and sex for July 1, Canada, provinces and territories," 2014. [Online]. Available: http://www.statcan.gc.ca/tables-tableaux/sum-som/l01/cst01/demo02a-eng.htm.

[3] K. Ziegler-Graham, E. J. MacKenzie, P. L. Ephraim, T. G. Travison, and R. Brookmeyer, "Estimating the Prevalence of Limb Loss in the United States: 2005 to 2050," *Arch. Phys. Med. Rehabil.*, vol. 89, no. 3, pp. 422–429, 2008.

[4] K. R. Vanderheiden, Gregg C. Vanderheiden, "A Brief Introduction to Disabilities," *Access. Des. Consum. Prod. Guidel. Des. Consum. Prod. to Increase Their Access. to People with Disabil. or Who are Aging*, vol. 1934, 1991.

[5] H. Berger, "Über das Elektrenkephalogramm des Menschen," *Arch. Psychiatr. Nervenkr.*, vol. 87, no. 1, pp. 527–570, Dec. 1929.

[6] J. J. Vidal, "Toward direct brain-computer communication.," *Annu. Rev. Biophys. Bioeng.*, vol. 2, pp. 157–180, 1973.

[7] S. Blain-Moraes, R. Schaff, K. L. Gruis, J. E. Huggins, and P. A. Wren, "Barriers to and mediators of brain–computer interface user acceptance: focus group findings," *Ergonomics*, vol. 55, no. 5, pp. 516–525, 2012.

[8] R. van Hattum, "Tegenlicht - Het gretige brein," VPRO, Netherlands, 2014.

[9] J. Padilha, *RoboCop*. USA, 2014.

[10] G. Tamburrini, "Brain to computer communication: Ethical perspectives on interaction models," *Neuroethics*, vol. 2, no. 3, pp. 137–149, 2009.

[11] D. Moran, "Evolution of brain-computer interface: Action potentials, local field potentials and electrocorticograms," *Curr. Opin. Neurobiol.*, vol. 20, no. 6, pp. 741–745, 2010.

[12] J. DEL R. MILLÁN, P. W. FERREZ, F. GALÁN, E. LEW, and R. CHAVARRIAGA, "NON-INVASIVE BRAIN-MACHINE INTERACTION," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 22, no. 05. pp. 959–972, 2008.

[13] S. F. Liang, C. T. Lin, R. C. Wu, Y. C. Chen, T. Y. Huang, and T. P. Jung, "Monitoring driver's alertness based on the driving performance estimation and the EEG power spectrum analysis.," *Conf. Proc. IEEE Eng. Med. Biol. Soc.*, vol. 6, pp. 5738–5741, 2005.

[14] L.-D. Liao, C.-Y. Chen, I.-J. Wang, S.-F. Chen, S.-Y. Li, B.-W. Chen, J.-Y. Chang, and C.-T. Lin, "Gaming control using a wearable and wireless EEG-based brain-computer interface device with novel dry foam-based sensors," *Journal of NeuroEngineering and Rehabilitation*, vol. 9, no. 1. p. 5, 2012.

[15] D. L. Schomer and F. Lopes da Silva, *Niedermeyer's Electroencephalography: Basic Principles, Clinical Applications, and Related Fields*, vol. 1. 2012.

[16] J. Łaszczyk, A. Mączko, W. Walas, and A. J. Nowak, *Information Technologies in Biomedicine*, vol. 7339. 2012.

[17] R. Padmavathi and V. Ranganathan, "A Review on EEG Based Brain Computer Interface Systems," vol. 4, no. 4, pp. 683–696, 2014.

[18] T. Cao, X. Wang, B. Wang, C. M. Wong, F. Wan, P. U. Mak, P. I. Mak, and M. I. Vai, "A high rate online SSVEP based brain-computer interface speller," in *2011 5th International IEEE/EMBS Conference on Neural Engineering, NER 2011*, 2011, pp. 465–468.

[19] G. Pfurtscheller and C. Neuper, "Motor imagery and direct brain-computer communication," *Proc. IEEE*, vol. 89, no. 7, 2001.

[20] Z. A. Keirn and J. I. Aunon, "A new mode of communication between man and his surroundings," *IEEE Trans. Biomed. Eng.*, vol. 37, no. 12, pp. 1209–1214, 1990.

[21] S. A. Huettel, A. W. Song, and G. McCarthy, *Functional magnetic resonance imaging*. 2004.

[22] P. M. Matthews, G. D. Honey, and E. T. Bullmore, "Applications of fMRI in translational medicine and clinical practice.," *Nat. Rev. Neurosci.*, vol. 7, no. 9, pp. 732–744, 2006.

[23] R. Sitaram, A. Caria, R. Veit, T. Gaber, G. Rota, A. Kuebler, and N. Birbaumer, "FMRI brain-computer interface: A tool for neuroscientific research and treatment," *Comput. Intell. Neurosci.*, vol. 2007, 2007.

[24] N. Weiskopf, R. Veit, M. Erb, K. Mathiak, W. Grodd, R. Goebel, and N. Birbaumer, "Physiological self-regulation of regional brain activity using real-time functional magnetic resonance imaging (fMRI): Methodology and exemplary data," *Neuroimage*, vol. 19, no. 3, pp. 577–586, 2003.

[25] H. Ayaz, P. A. Shewokis, S. Bunce, and B. Onaral, "An optical brain computer interface for environmental control," in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, 2011, pp. 6327–6330.

[26] N. Naseer and K.-S. Hong, "fNIRS-based brain-computer interfaces: a review," *Front. Hum. Neurosci.*, vol. 9, 2015.

[27] J. Shin and J. Jeong, “Multiclass classification of hemodynamic responses for performance improvement of functional near-infrared spectroscopy-based brain-computer interface.,” *J. Biomed. Opt.*, vol. 19, no. 6, p. 067009, 2014.

[28] M. a. Lebedev and M. A. L. Nicolelis, “Brain-machine interfaces: past, present and future,” *Trends Neurosci.*, vol. 29, no. 9, pp. 536–546, 2006.

[29] Adrian, “The Basis of Sensation,” *Christophers*, 1928.

[30] P. R. Kennedy and R. A. Bakay, *Restoration of neural output from a paralyzed patient by a direct brain connection.*, vol. 9, no. 8. 1998, pp. 1707–1711.

[31] B. Pesaran, M. J. Nelson, and R. A. Andersen, “Dorsal Premotor Neurons Encode the Relative Position of the Hand, Eye, and Goal during Reach Planning,” *Neuron*, vol. 51, no. 1, pp. 125–134, 2006.

[32] J. W. Fawcett and R. A. Asher, “The glial scar and central nervous system repair,” *Brain Research Bulletin*, vol. 49, no. 6. pp. 377–391, 1999.

[33] A. Palmini, “The concept of the epileptogenic zone: A modern look at Penfield and Jasper’s views on the role of interictal spikes,” *Epileptic Disord.*, vol. 8, no. SUPPL. 2, pp. 10–15, 2006.

[34] E. E. Sutter, “The brain response interface: communication through visually-induced electrical brain responses,” *Journal of Microcomputer Applications*, vol. 15, no. 1. pp. 31–45, 1992.

[35] N. E. Crone, D. L. Miglioretti, B. Gordon, J. M. Sieracki, M. T. Wilson, S. Uematsu, and R. P. Lesser, “Functional mapping of human sensorimotor cortex with electrocorticographic spectral analysis. I. Alpha and beta event-related desynchronization.,” 1998.

[36] N. E. Crone, D. L. Miglioretti, B. Gordon, and R. P. Lesser, “Functional mapping of human sensorimotor cortex with electrocorticographic spectral analysis. II. Event-related synchronization in the gamma band.,” *Brain*, vol. 121 (Pt 1, pp. 2301–2315, 1998.

[37] E. C. Leuthardt, G. Schalk, J. R. Wolpaw, J. G. Ojemann, and D. W. Moran, “A brain-computer interface using electrocorticographic signals in humans.,” 2004.

[38] G. Schalk, K. J. Miller, N. R. Anderson, J. A. Wilson, M. D. Smyth, J. G. Ojemann, D. W. Moran, J. R. Wolpaw, and E. C. Leuthardt, “Two-dimensional movement control using electrocorticographic signals in humans.,” *J. Neural Eng.*, vol. 5, no. 1, pp. 75–84, 2008.

[39] W. J. Freeman, L. J. Rogers, M. D. Holmes, and D. L. Silbergeld, “Spatial spectral analysis of human electrocorticograms including the alpha and gamma bands,” *J. Neurosci. Methods*, vol. 95, no. 2, pp. 111–121, 2000.

[40] G. Schalk and E. C. Leuthardt, “Brain-computer interfaces using electrocorticographic signals,” *IEEE Rev. Biomed. Eng.*, vol. 4, pp. 140–154, 2011.

[41] D. P. Buxhoeveden and M. F. Casanova, “The minicolumn hypothesis in neuroscience.,” *Brain*, vol. 125, no. Pt 5, pp. 935–951, 2002.

[42] A. Belitski, A. Gretton, C. Magri, Y. Murayama, M. a Montemurro, N. K. Logothetis, and S. Panzeri, “Low-frequency local field potentials and spikes in primary visual cortex convey independent visual information.,” *J. Neurosci.*, vol. 28, no. 22, pp. 5696–5709, 2008.

[43] X. Jia and A. Kohn, “Gamma rhythms in the brain,” *PLoS Biol.*, vol. 9, no. 4, 2011.

[44] Y. Li, Y. Hao, D. Wang, Q. Zhang, Y. Liao, X. Zheng, and W. Chen, “Decoding grasp types with high frequency of local field potentials from primate primary dorsal premotor cortex,” in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, 2012, pp. 1691–1694.

[45] B. Babadi and E. N. Brown, “A review of multitaper spectral analysis.,” *IEEE Trans. Biomed. Eng.*, vol. 61, no. 5, pp. 1555–64, 2014.

[46] D. B. Percival and A. T. Walden, *Spectral Analysis for Physical Applications: Multitaper and Conventional Univariate Techniques*, vol. 38. 1993.

[47] J. Donoghue, N. Hatsopoulos, and M. Serruya, “Methods and systems for processing of brain signals,” 2004.

[48] D. A. Heldman, W. Wang, S. S. Chan, and D. W. Moran, “Local field potential spectral tuning in motor cortex during reaching,” in *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2006, vol. 14, no. 2, pp. 180–183.

[49] J. M. Antelis, L. Montesano, A. Ramos-Murguialday, N. Birbaumer, and J. Minguez, “On the Usage of Linear Regression Models to Reconstruct Limb Kinematics from Low Frequency EEG Signals,” *PLoS One*, vol. 8, no. 4, 2013.

[50] D. R. Cox, “The Regression Analysis of Binary Sequences,” *J. R. Stat. Soc. Ser. B*, vol. 20, no. 2, pp. 215–242, 1958.

[51] R. Tomioka, K. Aihara, and K. Muller, “Logistic regression for single trial EEG classification,” *Adv. Neural Inf. Process. Syst.*, vol. 19, p. 1377, 2007.

[52] P. S. Hammon, S. Makeig, H. Poizner, E. Todorov, and V. R. de Sa, “Predicting reaching targets from human EEG,” *IEEE Signal Process. Mag.*, vol. 25, no. 1, pp. 69–77, 2008.

[53] S. Ryali, K. Supekar, D. A. Abrams, and V. Menon, “Sparse logistic regression for whole-brain classification of fMRI data,” *Neuroimage*, vol. 51, no. 2, pp. 752–764, 2010.

[54] J. Zhuang, W. Truccolo, C. Vargas-Irwin, and J. P. Donoghue, "Reconstructing grasping motions from high-frequency local field potentials in primary motor cortex," in *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC'10*, 2010, vol. 32, no. 5, pp. 4347–4350.

[55] W. L. Goh, D. P. Mital, and H. A. Babri, "An artificial neural network approach to handwriting recognition," *Proc. 1st Int. Conf. Conv. Knowl. Based Intell. Electron. Syst. KES '97*, vol. 1, 1997.

[56] D. Wu, K. Warwick, Z. Ma, J. G. Burgess, S. Pan, and T. Z. Aziz, "Prediction of Parkinson's disease tremor onset using radial basis function neural networks," *Expert Syst. Appl.*, vol. 37, no. 4, pp. 2923–2928, 2010.

[57] L. Kauhanen, T. Nykopp, J. Lehtonen, P. Jylänki, J. Heikkonen, P. Rantanen, H. Alaranta, and M. Sams, "EEG and MEG brain-computer interface for tetraplegic patients," in *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2006, vol. 14, no. 2, pp. 190–193.

[58] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[59] S. R. Safavian and D. Landgrebe, "A survey of decision tree classifier methodology," *IEEE Trans. Syst. Man Cybern.*, vol. 21, no. 3, pp. 660–674, 1991.

[60] W. Chen, Y. Wang, G. Cao, G. Chen, and Q. Gu, "A random forest model based classification scheme for neonatal amplitude-integrated EEG," *Biomed. Eng. Online*, vol. 13, no. Suppl 2, p. S4, 2014.

[61] G. Langs, B. H. Menze, D. Lashkari, and P. Golland, "Detecting stable distributed patterns of brain activation using Gini contrast," *Neuroimage*, vol. 56, no. 2, pp. 497–507, 2011.

[62] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[63] N. Cristianini and J. Shawe-Taylor, *An introduction to Support Vector Machines*, vol. 47, no. 2. 2000.

[64] G. Hughes, "On the mean accuracy of statistical pattern recognizers," *IEEE Trans. Inf. Theory*, vol. 14, no. 1, 1968.

[65] S. Niketeghad, A. O. Hebb, J. Nedrud, S. J. Hanrahan, and M. H. Mahoor, "Single trial behavioral task classification using subthalamic nucleus local field potential signals," in *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2014, pp. 3793–3796.

[66] Y. Wang and S. Makeig, “Predicting Intended Movement Direction Using {EEG} from Human Posterior Parietal Cortex,” in *Foundations of Augmented Cognition. Neuroergonomics and Operational Neuroscience*, vol. 5638, 2009, pp. 437–446.

[67] G. E. Hinton and T. J. Sejnowski, *Unsupervised Learning: Foundations of Neural Computation*. MIT Press, 1999.

[68] O. A. Mullette-Gillman, Y. E. Cohen, and J. M. Groh, “Eye-centered, head-centered, and complex coding of visual and auditory targets in the intraparietal sulcus.,” *J. Neurophysiol.*, vol. 94, no. 4, pp. 2331–2352, 2005.

[69] A. P. Batista, C. A. Buneo, L. H. Snyder, and R. A. Andersen, “Reach plans in eye-centered coordinates.,” *Science*, vol. 285, no. 5425, pp. 257–260, 1999.

[70] R. P. Dum and P. L. Strick, “Motor areas in the frontal lobe of the primate,” *Physiol. Behav.*, vol. 77, no. 4–5, pp. 677–682, 2002.

[71] J. Tanné, D. Boussaoud, N. Boyer-Zeller, and E. M. Rouiller, “Direct visual pathways for reaching movements in the macaque monkey.,” *Neuroreport*, vol. 7, no. 1, pp. 267–272, 1995.

[72] S. P. Wise, D. Boussaoud, P. B. Johnson, and R. Caminiti, “Premotor and parietal cortex: corticocortical connectivity and combinatorial computations.,” *Annu. Rev. Neurosci.*, vol. 20, pp. 25–42, 1997.

[73] R. C. Leiguarda and C. D. Marsden, “Limb apraxias: higher-order disorders of sensorimotor integration.,” *Brain*, vol. 123 (Pt 5, pp. 860–879, 2000.

[74] Y. E. Cohen and R. A. Andersen, “A common reference frame for movement plans in the posterior parietal cortex.,” *Nat. Rev. Neurosci.*, vol. 3, no. 7, pp. 553–562, 2002.

[75] M. Zhang and S. Barash, “Neuronal switching of sensorimotor transformations for antisaccades.,” *Nature*, vol. 408, no. 6815, pp. 971–975.

[76] A. Gail and R. A. Andersen, “Neural dynamics in monkey parietal reach region reflect context-specific sensorimotor transformations.,” *J. Neurosci.*, vol. 26, no. 37, pp. 9376–9384, 2006.

[77] S. C. Steenrod, M. H. Phillips, and M. E. Goldberg, “The lateral intraparietal area codes the location of saccade targets and not the dimension of the saccades that will be made to acquire them.,” *J Neurophysiol*, vol. 109, no. 10, pp. 2596–605, 2013.

[78] E. Hoshi and J. Tanji, “Integration of target and body-part information in the premotor cortex when planning action.,” *Nature*, vol. 408, no. 6811, pp. 466–470, 2000.

[79] E. Bizzi, “Discharge of frontal eye field neurons during eye movements in unanesthetized monkeys.,” *Science*, vol. 157, no. 796, pp. 1588–1590, 1967.

[80] S. P. Wise, “The primate premotor cortex: past, present, and preparatory.,” *Annu. Rev. Neurosci.*, vol. 8, pp. 1–19, 1985.

[81] L. Fogassi, V. Gallese, G. Buccino, L. Craighero, L. Fadiga, and G. Rizzolatti, “Cortical mechanism for the visual guidance of hand grasping movements in the monkey: A reversible inactivation study.,” *Brain*, vol. 124, no. Pt 3, pp. 571–586, 2001.

[82] D. Boussaoud and F. Bremmer, “Gaze effects in the cerebral cortex: Reference frames for space coding and action,” in *Experimental Brain Research*, 1999, vol. 128, no. 1–2, pp. 170–180.

[83] R. Caminiti, P. B. Johnson, C. Galli, S. Ferraina, and Y. Burnod, “Making arm movements within different parts of space: the premotor and motor cortical representation of a coordinate system for reaching to visual targets.,” *J. Neurosci.*, vol. 11, no. 5, pp. 1182–1197, 1991.

[84] C. Chatfield, *The Analysis of Time Series: An Introduction*. 2003.

[85] H. Bokil, P. Andrews, J. E. Kulkarni, S. Mehta, and P. P. Mitra, “Chronux: A platform for analyzing neural signals,” *J. Neurosci. Methods*, vol. 192, no. 1, pp. 146–151, 2010.

[86] I. Jolliffe, *Principal Component Analysis*. New York: Springer-Verlag, 2002.

[87] W. Karush and A. W, “Karush–Kuhn–Tucker conditions,” *Optimization*, pp. 1–3, 2008.

[88] C. Stetson and R. A. Andersen, “The Parietal Reach Region Selectively Anti-Synchronizes with Dorsal Premotor Cortex during Planning,” *J. Neurosci.*, vol. 34, no. 36, pp. 11948–11958, Sep. 2014.

[89] E. Stark and M. Abeles, “Predicting Movement from Multiunit Activity,” *J. Neurosci.*, vol. 27, no. 31, pp. 8387–8394, Aug. 2007.

[90] B. Pesaran, J. S. Pezaris, M. Sahani, P. P. Mitra, and R. a Andersen, “Temporal structure in neuronal activity during working memory in macaque parietal cortex.,” *Nat. Neurosci.*, vol. 5, pp. 805–811, 2002.

[91] E. Buch, C. Weber, L. G. Cohen, C. Braun, M. A. Dimyan, T. Ard, J. Mellinger, A. Caria, S. Soekadar, A. Fourkas, and N. Birbaumer, “Think to Move: a Neuromagnetic Brain-Computer Interface (BCI) System for Chronic Stroke,” *Stroke*, vol. 39, no. 3, pp. 910–917, Mar. 2008.

[92] G. Barnes, A. Hillebrand, and M. Hirata, “Magnetoencephalogram,” *Scholarpedia*, vol. 5, no. 7, p. 3172, 2010.

[93] Z. Wang, A. Gunduz, P. Brunner, A. L. Ritaccio, Q. Ji, and G. Schalk, “Decoding onset and direction of movements using Electrocorticographic (ECoG) signals in humans,” *Front. Neuroeng.*, vol. 5, 2012.

[94] J. Dean and S. Ghemawat, “MapReduce,” *Commun. ACM*, vol. 51, no. 1, p. 107, Jan. 2008.

[95] Wikipedia, “Artificial neural network,” 2015. [Online]. Available: http://en.wikipedia.org/wiki/Artificial_neural_network.

[96] Wikipedia, “Decision Tree.” [Online]. Available: http://en.wikipedia.org/wiki/Decision_tree_learning.

[97] Wikipedia, “Support Vector Machine,” 2015.