The Dissertation Committee for Vijay Aditya Tadipatri
certifies that this is the approved version of the following dissertation:

Developing Robust Movement Decoders for Local Field Potentials

Committee:

Ahmed H. Tewfik, Supervisor

Brian Evans

James Ashe

Eyal Seidemann

Grady H. Rylander

Haris Vikalo

Developing Robust Movement Decoders for Local Field Potentials

by

Vijay Aditya Tadipatri, B.E., M.E.E.

DISSERTATION

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN

May 2015

Dedicated to

Munni

Acknowledgments

I would like to express my gratitude towards several individuals who stood by me during my PhD studies.

Firstly, I would like to thank my adviser, Dr. Ahmed H. Tewfik, for motivating me and trusting in my ability. He created a very friendly atmosphere, where I could discuss my ideas and express my thoughts. His knowledge, motivation, and enthusiasm especially during times when I had particularly difficult problems, helped me arrive at swift solutions. I could not have hoped for a better adviser than him during my PhD study. I greatly appreciate his keenness to develop practically applicable projects and hope to carry this philosophy forward.

I would also like to sincerely thank Dr. James Ashe, for his numerous discussions on my project. He taught me basics of neuroscience and was patient with me during my several questions. I would like to thank my committee: Dr. Brian Evans, Dr. Grady Rylander, Dr. Eyal Seidemann, and Dr. Haris Vikalo for their encouragement, insightful comments and challenging questions. I would also like to thank Dr. Giuseppe Pellizzer for his support and discussions during my dissertation.

Graduate school gives you an opportunity to meet several likeminded people. I was lucky to have met many during my time at school. I thank my fel-

low lab mates Dr. Nuri Firat, Dr. Rahul Gupta, Dr. Vikrham Gowreesunker, Dr. Nikhil Kundargi, Vimal Singh, Youngchun Kim, Dr. Yingxi Liu, Neeraj Gaur and Megan Ruthven for their guidance, stimulating discussions, several long hours of work, and most importantly making this entire experience very joyful.

I would also like to express sincere thanks and gratitude to friends in both Minneapolis and Austin, especially Vimal Singh, Pranav Bhandarkar, Krishna Chaitanya and Doron Talegaonkar.

I would like to thank my mother and my sister for their love, support and motivation during my entire studies. I am forever indebted to my wife, Kalyani Mallela, for her unwavering and unconditional support during my graduate school. Her motivation, patience, sacrifice and love gave me strength to purse my studies.

Developing Robust Movement Decoders for Local Field Potentials

Publication No. _____

Vijay Aditya Tadipatri, Ph.D.
The University of Texas at Austin, 2015

Supervisor: Ahmed H. Tewfik

Brain Computer Interfaces (BCI) are devices that translate acquired neural signals to command and control signals. Applications of BCI include neural rehabilitation and neural prosthesis (thought controlled wheelchair, thought controlled speller etc.) to aid patients with disabilities and to augment human computer interaction. A successful practical BCI requires a faithful acquisition modality to record high quality neural signals; a signal processing system to construct appropriate features from these signals; and an algorithm to translate these features to appropriate outputs. Intracortical recordings like local field potentials provide reliable high SNR signals over long periods and suit BCI applications well. However, the non-stationarity of neural signals poses a challenge in robust decoding of subject behavior. Most BCI research focuses either on developing daily re-calibrated decoders that require exhaustive training sessions; or on providing cross-validation results. Such results

ignore the variation of signal characteristics over different sessions and provide an optimistic estimate of BCI performance. Specifically, traditional BCI algorithms fail to perform at the same level on chronological data recordings. Neural signals are susceptible to variations in signal characteristics due to changes in subject behavior and learning, and variability in electrode characteristics due to tissue interactions. While training day-specific BCI overcomes signal variability, BCI re-training causes user frustration and exhaustion.

This dissertation presents contributions to solve these challenges in BCI research. Specifically, we developed decoders trained on a single recording session and applied them on subsequently recorded sessions. This strategy evaluates BCI in a practical scenario with a potential to alleviate BCI user frustration without compromising performance. The initial part of the dissertation investigates extracting features that remain robust to changes in neural signal over several days of recordings. It presents a qualitative feature extraction technique based on ranking the instantaneous power of multichannel data. These qualitative features remain robust to outliers and changes in the baseline of neural recordings, while extracting discriminative information. These features form the foundation in developing robust decoders.

Next, this dissertation presents a novel algorithm based on the hypothesis that multiple neural spatial patterns describe the variation in behavior. The presented algorithm outperforms the traditional methods in decoding over chronological recordings. Adapting such a decoder over multiple recording sessions (over 6 weeks) provided $> 90\%$ accuracy in decoding eight movement

directions. In comparison, performance of traditional algorithms like Common Spatial Patterns deteriorates to 16% over the same time. Over time, adaptation reinforces some spatial patterns while diminishing others. Characterizing these spatial patterns reduces model complexity without user input, while retaining the same accuracy levels.

Lastly, this dissertation provides an algorithm that overcomes the variation in recording quality. Chronic electrode implantation causes changes in signal-to-noise ratio (SNR) of neural signals. Thus, some signals and their corresponding features available during training become unavailable during testing and vice-versa. The proposed algorithm uses prior knowledge on spatial pattern evolution to estimate unknown neural features. This algorithm overcomes SNR variations and provides up to 93% decoding of eight movement directions over 6 weeks. Since model training requires only one session, this strategy reduces user frustration. In a practical closed-loop BCI, the user learns to produce stable spatial patterns, which improves performance of the proposed algorithms.

Table of Contents

List of Tables

List of Figures

Chapter 1

Motivation

Tetraplegia or quadriplegia is a paralysis that results in ones inability to use their limbs due to the injury to brain or high-level spinal cord. In this condition, the patient loses both control and sensation of limbs. Brain Computer Interface (BCI) offers a solution to these patients by providing independent control and ability to move their limbs by imagining movement of those limbs. BCIs are devices that translate neural signals to control and communication signals to enable people with disabilities to control an external device such as a wheelchair or robotic arm [1, 213, 212, 98, 175, 18, 43, 36]. BCIs assist, augment, or repair human cognitive and motor functions. Practical BCI applications include neural prosthetic devices in patients with a loss of arm or limb, devices to control arm or leg in patients suffering from paralysis, brain controlled external devices like wheel chair [18, 159], and as aids in rehabilitation of people who suffered a stroke. They also augment the sensory system in applications like rapid sorting of images or interesting scenes [68], control in gaming and virtual environments [18, 149, 134, 52]. Finally, they provide a great analysis tool to understand the physiological and functional aspects of the brain. Applications of BCI are many; however, the above highlight some possibilities.

While the non-invasive nature of electroencephalography (EEG) makes it a lucrative recording modality, its spatial resolution and signal-to-noise-ratio (SNR) yield an inefficient practical BCI. As such, these characteristics restricts its use to binary decisions and provide poor information rates. On the other end of the recording spectrum, Single Unit Activity (SUA) provides high spatial resolution but is susceptible to recording losses over long time. Dickey et al. showed that during chronic recordings of single units only a fraction of them provided stable recordings over multiple days [40]. Electrocorticography (ECoG) and Local Field Potentials (LFP) are subdural recordings from the surface of the brain and offer better SNR than EEG [108, 8]. These recordings are composed of sustained currents in the brain avoiding domination by a single neuron. Due to this, ECoG and LFP offer stable recordings and suit long-term robust practical decoding applications [7]. Recent neural engineering advances show recording capabilities of sub-dural electrodes over multiple months [183, 40, 57, 56]. For example, Simeral et al. recorded LFP over 1000 days and Slutzky et al. recorded LFP for BMI over a 7 month period [183, 56]. These studies establish the long-term recording capability of LFP; however, its use in long-term decoding applications has not received much attention.

One of the main challenges in using LFP for long-term decoding is their day-to-day signal variability. Causes for the variations include:

1. Subject dependent variability in behavior, motivation, skill, and learning [190]

2. Experiment dependent variability due to micro motion of recording electrode [198]

3. Neuronal variability in the excitation of neuronal components, variability of the electrode's electrical characteristics like impedance due to accumulation of brain tissue, and scarring effects of the implantation [153]

Due to the multitude of causes, the task to characterize the variability becomes very difficult. These inconsistencies manifest in various forms including variations in signal power and change in spatial patterns resulting in variability of derived features. Figure 1.1 and 1.2 present an example of the signal variability over a week. The figures present the spatial patterns of LFP power (dB) averaged over multiple reaches in the δ-band of the pre-motor grid area during a hand movement task. The location of each sub-figure corresponds to the movement direction on the 2-D hand movement space. The LFP signal was sampled from 57 electrodes arranged in a 10×10 grid on the pre-motor area. This power is spatially filtered using an interpolation operator to provide a easily viewable pre-motor cortex.

One way to tackle these variations is to have a retraining session to calibrate the BCI to new characteristics before every application session [22, 45, 46, 91, 107, 111, 128, 138, 157, 161, 171, 204, 205]. However, these pauses to calibrate BCI increase user frustration and fatigue in using BCI [135]. Most studies use cross-validation to measure and report BCI performance [21, 20, 45, 46, 174]. Such measures provide an optimistic view of their performance as

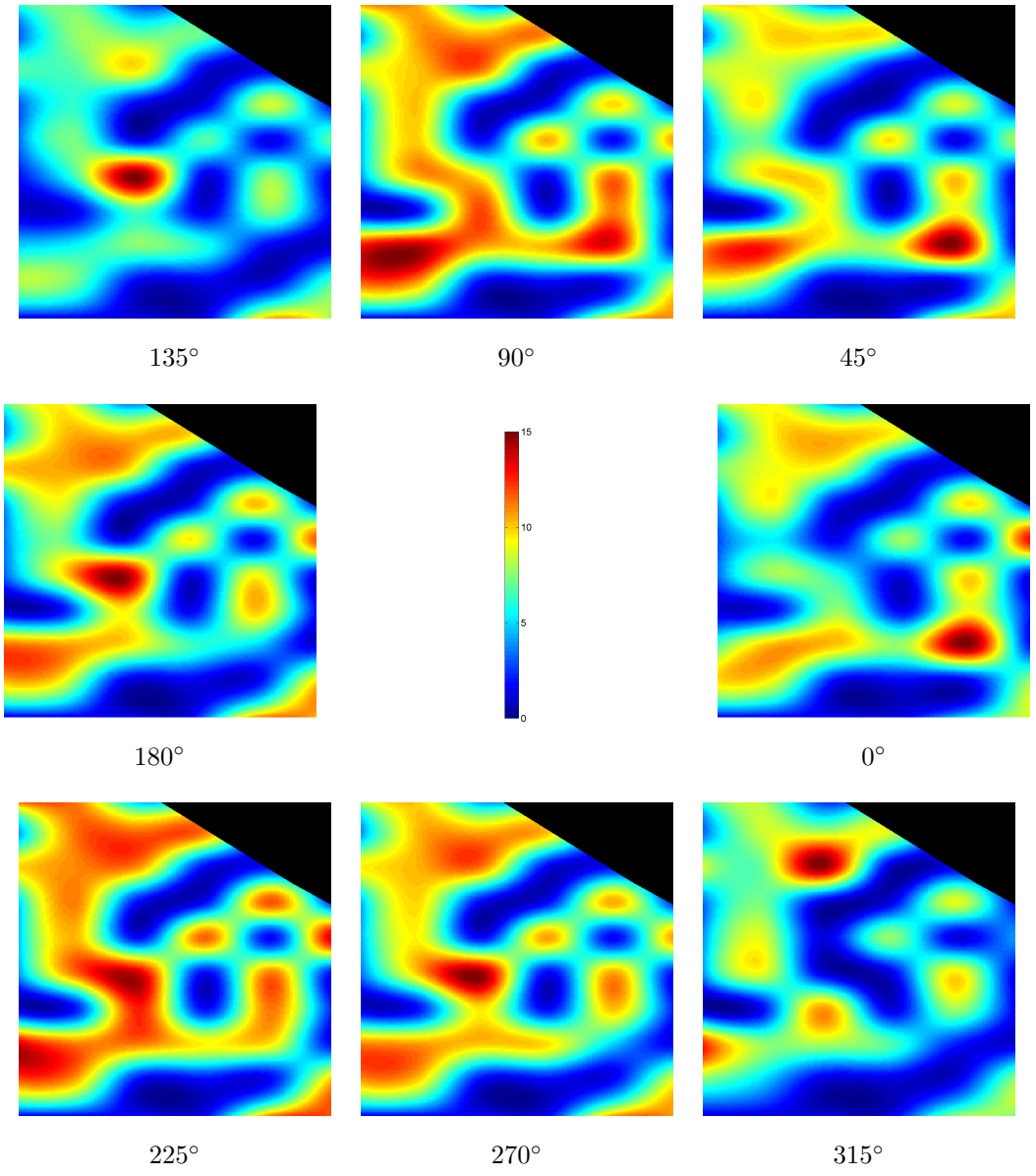


Figure 1.1: Average Neural Pattern for different directions of movement on day 0. The spatial patterns represent the activity of average Premotor LFP power (0-15dB as indicated in the color scale) for different direction reaches. The LFP is extracted from a $4mm \times 4mm$ grid placed in the Premotor area of the monkey brain. The different directions are indicated by their placement and also by the number under it.

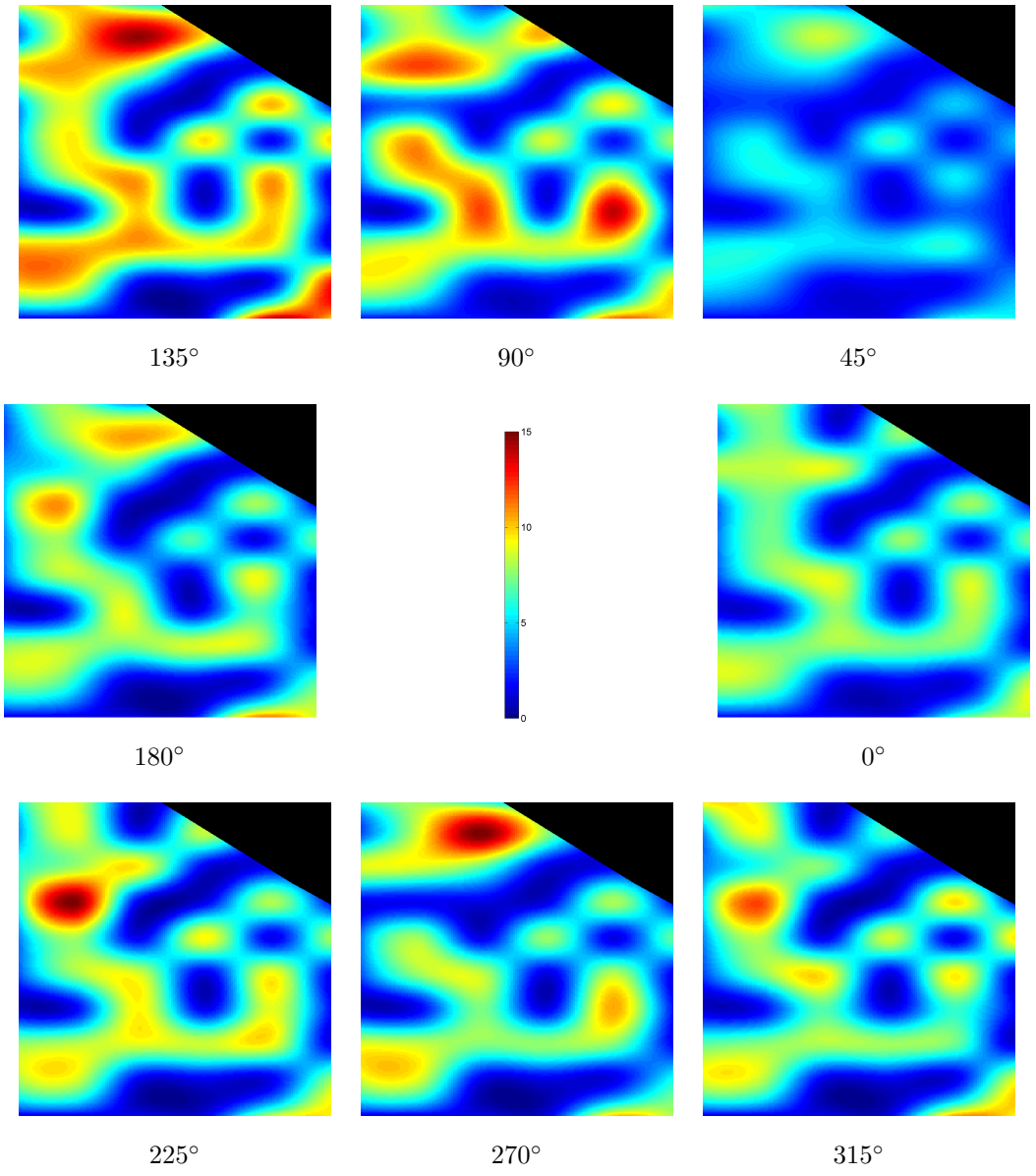


Figure 1.2: Average Neural Pattern for different directions of movement on day 8. The spatial patterns represent the activity of average Premotor LFP power (0-15dB as indicated in the color scale) for different direction reaches. The LFP is extracted from a $4mm \times 4mm$ grid placed in the Premotor area of the monkey brain. The different directions are indicated by their placement and also by the number under it.

cross-validation resamples training and testing data sets from the same data and ignore any variability or changes in the feature characteristics from the training to test data. However, these studies establish the feasibility of using the modalities for decoding applications.

Another solution to overcome variability is to derive invariant patterns from the training data [22]. Such a model would provide a general solution across all the variations of the feature space. However, this requires that all variations of the neural data be captured during training and is practically not possible due to limited data sets available during training. Therefore, it is not feasible to present all variations in behavior and changes in environments in a single training session. Even if such behavioral variations were possible, they do not account for the variability in signal characteristics over multiple days. Some studies have focused on developing robust features that overcome the signal non-stationarity [195] or look for features lying on recurring subspaces [74]. However, these approaches fail to capture the long-term variability of the signals and as such do not adapt to the day-to-day variations or variations imposed by changing environments.

A closed-loop BCI is a suitable solution as it provides reliable decoding over a long period. In closed-loop BCI, feedback about the decoders' performance is provided to the user in real time and the user incorporates this feedback by modulating the neural features [25, 141, 30, 64, 69, 95, 81, 110, 183]. Recent research shows that BCI user learns to produce a fixed neural pattern in about six to ten training sessions and daily practice improves BCI efficiency

[64]. These strategies, while useful in rehabilitation efforts, fail to adapt to changes in neural patterns due to subject-induced modulation of neural patterns. These modulations are especially noticeable when the subject learns new behavior and adapts to changing environments. Since any practical application of the BCI would eventually work in a closed-loop fashion, research suggests the use of closed-loop calibration to learn the closed-loop dynamics of the system [95]. Further, some decoders also internalize the uncertainty in the estimated behavior parameters by assuming that the BCI user corrects for any uncertainty of the estimate [69]. In this framework the burden of learning is placed on the BCI user who learns to adapt and modulate the neural signals to suit the model [203]. It is estimated that 20-25% of users fail to achieve such adaptation and control the BCI [203]. Hence, there is a need for a model that learns the users' mental state in an open-loop fashion. Such a model will also advance the understanding of learning without constraining the user to comply with a fixed pattern [132].

This thesis presents my contributions to developing a neural decoder performing in an open loop, where the user gets no feedback about the decoder's estimates. Our proposed decoder relieves the learning burden on the BCI user by mimicking a practical BCI environment. The contributions are summarized in the following thesis statements:

1. *Developing novel and time-robust neural features overcomes signal variability and improves decoding of hand movement over multiple days*

2. *Encapsulating the variability of subject behavior in multiple spatio-temporal patterns and capturing the changes in subject behavior by adapting decoding model to novel features improves long-term decoding*

3. *Estimating unobservable feature parameters by capturing prior model information overcomes day-to-day variation in channel SNR*

I developed several algorithms based on the above statements and applied these on data collected from two monkeys over multiple weeks. Specifically, I monitored the performance of these algorithms in a practical BCI framework. In such a setting, I trained decoding models on a single recording session and applied it over subsequent recordings. The algorithms presented in this thesis predicted the correct target in 90% of trials over a period of 4-6 weeks in two monkeys. The decoding performance remained stable through changes in environmental conditions and external perturbations on the hand during movement. These results indicate the applicability of LFP in a practical BCI setting with a decoder that requires minimal training.

Chapter 2

Background

2.1 Introduction

Brain Computer Interfaces (BCIs) translate neural signals acquired from the brain to command and/or control signals [1, 213, 212, 198]. BCIs offer hope of restoring functionality to patients suffering from paralysis or spinal cord injury. A Brain Computer Interface has three major components as shown in the figure 2.1.

1. A signal acquisition system to faithfully record neural signals
2. A signal processing system to convert the acquired signal to control data
3. An algorithm to interpret these signals to behavioral control or command signals for a BCI application

Generally, the BCI operates in a feedback loop providing its user with visual, audio, or perceptual feedback on its performance. This feedback enables the user to modulate the neural signals and obtain a desired effect. Signals from the brain are acquired either in the form of electrical potentials (EEG, ECoG, LFP, Spike Trains) or in magnetic fields (MEG, fMRI) from functionally active portions of the brain. This thesis focuses on the signal processing aspects of

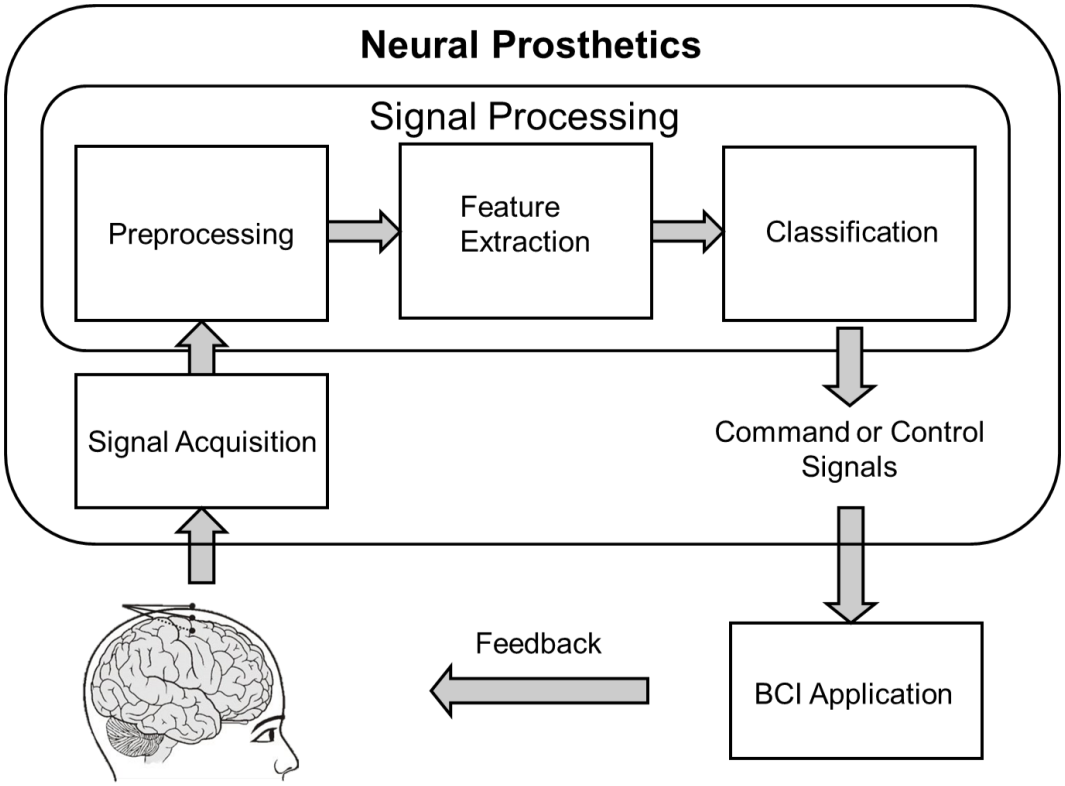


Figure 2.1: A schematic of Neural Prosthetic

the BCI i.e. preprocessing, feature extraction, and translation of these features to behavioral commands or controls. Understanding of feature extraction and signal processing decisions requires the understanding of signal acquisition modalities. The next few sections will provide a brief background to the various acquisition modalities including their relative advantages and disadvantages.

2.2 Neural Acquisition Modalities

There are various neural signal acquisition modalities ranging from non-invasive recordings like Electroencephalogram (EEG) to the invasive recordings of single neural units. Due to its non-invasive nature, EEG is the most pursued modality along with magneto-encephalogram (MEG). Since neural magnetic fields are not distorted by the skull, MEG has better spatial resolution and a temporal resolution of 10 ms (or less) [35]. However, MEG equipment requires rigorous magnetic shielding and as such cannot analyze freely moving subject behavior. EEG records neural activation in the form of electrical pulses using a skullcap wore tightly along the scalp by the subject. The activity of a single neuron is too small to be picked by an EEG electrode but the summation of all the synchronous neural activity of the brain gets recorded [114]. In addition, as the neural activity attenuates rapidly with distance, it becomes difficult to pick up neural activity in deep brain.

Most EEG based BCI applications deal with simple binary classification tasks such as Motor Imagery (MI) [98, 122, 146]. In these tasks, subjects imagine the motor tasks instead of actually performing them. Imagination of movements causes certain neural locations to oscillate with specific frequencies; generally μ-rhythms ($8 - 12$ Hz), β-rhythms ($13 - 40$ Hz) and γ-rhythms ($40 - 100$ Hz) [132, 146]. This phenomenon is also termed as Event Related Potentials (ERPs) —simultaneous oscillation of multiple EEG locations in response to a cognitive event [117]. Identifying such a spectral and spatial response of EEG allows classification of motor imagery, different motor tasks,

and motor vs. other cognitive tasks [21, 90, 89, 132, 146, 157, 172, 205]. This recording technique has several advantages including high temporal resolution ($\sim$ 1 ms), low hardware costs, subject tolerance, and is non-invasive. However, its low spatial resolution, poor signal-to-noise ratio, and inability to detect neural activity below the upper layers of the scalp limit the information rates of EEG based BCI.

Electrocorticography (ECoG) and subdural EEG use electrodes placed under the exposed surface of brain to record neural activations. ECoG is an invasive procedure requiring surgical incisions to place electrodes. The ECoG recordings are composed of synchronized postsynaptic potentials recorded directly from the surface of the cortex [207]. These potentials conduct through several layers of the cerebral cortex and cerebrospinal fluid before reaching the subdural electrodes. While EEG signal must pass though the skull, which has severe low pass filtering effects, ECoG does not conduct through it. Hence, they do not suffer from the rapid attenuation plaguing EEG signals. This allows a better spatial resolution ($\sim$ 1 cm) and a temporal resolution ($\sim$ 5 ms) for the ECoG [114]. Typical ECoG recordings use electrode arrays in various dimensions and configurations having anywhere from 4 to 256 electrodes [8, 16, 61, 81, 110, 129, 151, 169, 181, 207]. Due to the placement flexibility and greater proximity to neural activity, ECoG offers greater precision, a higher spatial resolution, and a superior signal-to-noise ratio than EEG. However, ECoG suffers from limited field of view and is dependent on the successful implant and surgery.

On the other end of the recording spectrum, single unit recordings measure the electrical responses of individual neurons by measuring and identifying their characteristic action potentials. A single-unit is defined as a single, firing neuron whose spike potentials are distinctly isolated [24]. Fine-tipped, high-impedance electrodes placed very close to a neuron record the rate of change in voltage with respect to time. A microelectrode placed close to the cell surface, measures extracellular recordings in the form of spike information. Intracellular recordings (tracing the membrane resting potentials) are possible through an intracellular electrode. Since they record at a single neuron level, Single Unit Activity (SUA) has the best signal to noise ratio and spatial resolution of all the modalities. However, it is very difficult to obtain a chronic recording of single unit activity due to tissue damage and effects of the electrode on the extracellular space [153, 198, 175, 6, 24]. In general, obtaining recordings from the same location over multiple days is extremely difficult. However, one of the important contributions of the research on SUA is that they provide a benchmark for the decoding power that can be achieved when analyzing the most detailed level of neuronal signal [28, 101, 208, 176, 196, 25]. Results from [196] showed that the decoding accuracy increases with the number of SUAs analyzed and that it can reach reasonable levels with relatively few neurons. The discussion below details the advantages of analyzing Local Field Potentials over Single Unit Activity analysis.

Local Field Potentials (LFPs) are recorded in the extracellular space of the brain typically using microelectrodes. LFP recorded from subdural elec-

trodes differs from ECoG, as they record signals from deep within the cortical tissue. Due to the invasive nature of their recordings they sample relatively localized populations of neurons (within a few hundred μm) [100]. Since they record from relatively close to the source, the signal suffers from less propagation attenuation in extracellular media like cerebrospinal fluid, dura matter, muscle, skull, and skin. However, LFP signals undergo filtering of high frequency spectrum due to the separation from the sources by portions of cortical tissues. It is hypothesized that LFP activity reflects the sum of action potentials from cells within $50 - 350$ micrometers [108, 185]. The signal is recorded using an extracellular microelectrode usually placed far from a single local neuron to prevent its domination. This signal is then low-pass filtered around 200-300 Hz to obtain the LFP data. The low-pass filtering effect partly occurs due to the complex electrical conductivity properties of the inhomogeneous extracellular space. LFP power-spectrum exhibits a characteristic $\frac{1}{f}$ frequency scaling at low frequencies explained by the filtering properties of the currents through extracellular media [17, 13, 137, 155]. Other theories including the morphology of the neuron [145] and self-organized critical phenomenon [96] might also explain the frequency scaling. It is important to note that the low-pass filtering effect of the medium also explains the non-dominance of individual single units away from the electrodes.

Interpreting LFP signals remains a difficult task, even if its neuronal genesis is well understood. A simple model hypothesizes LFP sources as electric dipoles generated by transmembrane currents that are embedded in per-

fectly resistive medium [14, 126]. Using such a model I obtained a extracellular potentials $V(r)$ at a position r due to a current element I_o located at position r_o in a medium with electrical conductivity σ as:

$$V(r) = \frac{1}{4\pi\sigma} \frac{I_o}{|r - r_0|} \tag{2.1}$$

There is still some debate over the exact nature of the sources regarding modeling them as dipoles or monopoles [14]. Many researchers studied the dipole excitation model to explain LFP recordings. Since the LFP is a synaptic activity over multiple neurons, it proves to be a stable modality providing chronic recording ability, unlike multi-unit activity, while providing higher signal to noise ratio and spatial resolution than ECoG. Recently, LFP recorded for chronic experiments like BCIs proved a good supplement to SUA in decoding behavior [128, 7, 161, 171, 43, 11, 121].

2.3 Neural Analysis Techniques

Many decoding and analysis techniques were proposed to decode and analyze neural signals. Based on the neural signal (EEG, ECoG, LFP or single units) under consideration, the analysis techniques differ to accommodate their different characteristics. Each technique was developed on observations made on the neural signal characteristics and catered well to their analysis. This section discusses some of the existing strategies to analyze neural signals and their applicability to the analysis of LFPs. Until recently, LFP was considered a supplementary signal and generally filtered out as the low frequency

component during single unit analysis [128, 7, 161, 171, 57, 56]. Researchers assumed that higher frequency bands contained most of the useful neural information to decode behavior. Hence, they either discarded or used the lower spectrum (primarily LFP signal) sparingly.

2.3.1 Analysis of Single Unit Activity

Analysis of SUA relies on different properties of its spike activity. A spike is recorded as the electrical deactivation of a neural cell in response to a particular behavior. Each recording site detects spikes from multiple neurons and each neuron has a characteristic signal that identifies it from the other activity. This knowledge is valuable in achieving spike separation. To correlate behavior and activity of the most fundamental element of the brain, researchers identified binned spike rate (also called firing rate) - calculated as the number of spikes in a time-window bin - as a reliable feature for SUA analysis. Typically a window bin of 100ms is used to calculate the number of spikes. Georgopolous et al. showed that neurons tune to their preferred directions and described them with a simple cosine tuning function of spike rate [67]. In general, researchers found that neurons fire more frequently during certain behaviors. Since the temporal resolution of firing rate is in the order of 5ms [208], they deliver precise timing information. This led towards using spike rate as a characteristic feature in SUA analysis. Time to first spike after stimulus onset and higher moments of inter spike interval distributions also encode stimulus [103, 59]. Analysis on the firing rate assumes that the firing

rate follows a Poisson distribution [9]. Thus, calculating the mean firing rate determines the relation to the stimulus.

Population coding is a popular method to encode stimuli using the activities of multiple neurons [67, 29, 75, 97, 120, 77]. It weighs the activities of several neurons by monitoring their multivariate distributions to obtain a tuning function for the stimulus. Sparse coding of the population identifies a subset of neurons that remains active during a given time window and their corresponding weights to encode the stimulus. This technique is especially useful when the dimensionality of the stimulus is very small compared to the number of sampled neurons. Recently, analysis of spike rate from a population of neurons using linear or non-linear dynamic functions has been used to detect kinematics of hand movements. In these papers, a Kalman filter relates the firing rate of a population of neurons to the state defined by the different kinematic state of the hand [179, 180, 36, 183, 63, 69]. These studies also laid foundation to the use of BCI and direct neural control of prosthetic limbs and joysticks.

LFP was used as a supplementary activity to the neural signal until recently [128, 138, 81, 113, 121, 161, 9]. The addition of LFP to the existing spike information boosts performance of the BCI, although in most cases the improvement was marginal. Most of the LFP research focused on using features similar to spike rate and inter-spike interval [9, 11, 113]. These studies used power calculated in specific bands of LFP activity as a surrogate to the spike rates. While such analysis introduced LFP and its information content,

the analysis tools were not adapted to the task. In the context of movement decoding from intra-cortical modalities, preferred direction for both LFP and SUA were used to decode direction. The distribution of preferred directions of SUAs is known to be uniformly or close to uniformly distributed [164], while the distributions of LFP's preferred directions tended to form clusters [9, 138]. Due to this fundamental dissimilarity between SUA and LFP signals, the latter cannot reliably decode movement directions using methods like cosine tuning.

2.3.2 Analysis of EEG and ECoG

Other inspirations for LFP analysis come from the EEG and ECoG analysis, where a plethora of techniques exists. These techniques have varied from using temporal features using multi-channel neural data, analysis of band-filtered signals, and identifying discriminative spatio-temporal patterns. These techniques also seem suitable for analysis as EEG, ECoG, and LFPs have similar temporal structure and their analysis usually involves data from multiple channels. Analysis of EEG and ECoG mainly focused on detecting Event Related Potentials (ERP) - a deviation of the signal from its baseline due to the onset of a stimulus [111, 132, 21, 44]. When such a deviation is below the baseline due to a stimulus it is called an event related desynchronization (ERD) and if above is called event related synchronization (ERS). ERD or ERS emerges as an important characteristic of the stimulus and extensively used in the P300 based odd ball detection [18, 149]. In several applications, interesting meaningful stimulus produce a positive going wave with a typical

latency of $300-1000$ms on one of the Pz, Fz, and Cz scalp sites of the EEG cap in the parietal lobe [68, 52, 18]. To detect these ERPs several morphological features of the electrode recordings have been proposed including temporal features (latency and amplitude of the maximum signal, slope of the deviation, peak-to-peak value of the deviation, zero crossings etc.), frequency features (frequency with the maximum energy component), and wavelet features [3]. Features that provide the best detection are identified based on their performance accuracy and the latency of detection. While the neural substrates of P300 are not fully understood, their reproducibility makes them a good choice for clinical and lab evaluations. Applications of P300 include:

1. rapid image sorting —to detect interesting images of helipads from a random collection of images [68]

2. lie detection —to detect if a presented real-life situation occurred with a subject [52]

3. speller —to spell a word or a sentence by using letters without the need of neuro-muscular movement [18]

While such initial analysis provided evidence of information in EEG, researchers postulated that analysis of multiple channels improves information content. Initial studies applied techniques described for single channel analysis and showed that band powers could be easily controlled by the subjects; achieving 2-3 times the random classification in choosing one of four targets

[45, 172, 17, 122]. These studies showed evidence that subjects could control the two channels of neural information at least in a specific frequency band - the μ rhythms - and produce consistent waveforms. In analysis of multiple channel recordings, it is advantageous to use techniques that cater to multi-channel analysis. In early analysis of EEG, most researchers analyzed if a specific frequency provided important information on the classification. The underlying principle is nevertheless, based on the detection of event related potentials in a specific frequency band of interest. The band power was calculated from fixed windows and multiple pattern recognition and machine learning techniques like artificial neural networks process the band power features to classify behavioral tasks [154]. Another set of methods developed auto-regressive models to deal with the non-stationarity of the EEG signals [172]. The model describes EEG as a stochastic process and the model parameters are estimated adaptively. This method requires no prior knowledge of the reactive frequency band and models the spectral component of the signal with updated auto regressive parameters. To maintain a satisfactory performance the model parameters need adaptation. Assuming that model parameters change slowly over time, a continuous visual feedback by the BCI user achieves adaptation. The choice of the update coefficient determines the rate of update and balances between accuracy and speed of feedback. Schlogl et al. proposed such a method in combination with a linear classifier that provided 90% accuracy in classifying right vs. left hand imagination [172].

2.3.3 Feature Extraction for Multi-Channel Neural Analysis

The methods mentioned above consider a single frequency band of interest and consider only features from this band. Such analysis requires efforts to determine the relevant frequency band and ignore any inter-frequency interactions that might characterize the behavior better. With this focus, researchers developed simultaneous temporal and spectral features [71, 90, 89]. The main idea behind such research is to develop a redundant dictionary with temporal and spectral features using either wavelet decomposition or frequency analysis. Wavelet decomposition of time and frequency components builds a wavelet feature dictionary. Altering the scale of decomposition changes the analyzed time-frequency components [71, 90, 89, 92]. Using a block Fourier transform, also accomplishes the same task with similar accuracy [88]. This transform substitutes wavelet decomposition with FFT frequency decomposition in a pre-determined time window. This method retains normalized energy in each time-frequency block as an analysis feature. Although using all the features would provide high accuracy in the training set due to the curse of dimensionality, they provide poor classification in the testing set. Identifying a subset of the features that generalize the accuracy is essential for neural decoders and analysis tools. Hence, the burden of classification lies in identifying a subset of features that are useful to classify the activity. Several methods of feature selection suggested in [71, 90, 89, 92] include:

1. Thresholding - Using a predetermined threshold, only features above it can be considered in the analysis. This is akin to the threshold in the

wavelet decomposition based noise removal used in the image processing and image compression literature. This process reduces the number of features drastically and quickly reduces the analysis computation time. The choice of threshold determines the tradeoff between computation time and model accuracy.

2. Greedy Subset selection - Selecting a subset of features that have high wavelet components gathers only the top features based on their wavelet coefficients. As high-energy components retain most information content, such analysis resulted in good performance. Choosing such a subset of features allows the determination of feature interaction in terms of time and frequency components. This subset of features while providing the maximum energy components do not consider any information provided by lower energy components and as such provides locally optimal solutions.

3. Subset Selection using Wrapper methods - In these methods, subsets of features are determined by the amount of information content offered by their combination. These methods are initialized with either the maximal energy component or the maximal information component. The information from a feature can be determined with the use of many measures including Area Under the Receiver Operator Curve (AUC), Genie Impurity (GI) or Fisher Discrimination criteria (FDC) [88]. The theme of these quantities is to provide a good measure of their discrimination

capabilities. In the wrapper methods, any new feature is selected by choosing the best available feature that in combination with the existing features improves discrimination.

Further analysis of multi-channel EEG revealed the existence of inter-channel interaction during brain activity. Some of the early work focused on developing features similar to the ones described above by creating time-frequency features for each recorded channel [71, 90, 89, 92]. This method results in an increased number of features and combinations with the addition of new channels. Hence new feature selection methods focused on pruning techniques that considered the spatial, spectral, and temporal origins of the features. Using one such method, all features related to that selected feature can be pruned out of further analysis. This strategy prunes features that share the same spatial, temporal, and spectral locations. It results in a quick training algorithm that uses only a fraction of the original number of features that provided similar performance; for example, subset selection wrapper methods that used 20% of the original features and provided the same performance in analyzing EEG and ECoG data sets [71]. The method required no knowledge of the spatial arrangement of channels and their neighborhoods. In fact, the spatial locations identified by the algorithm corresponded with contralateral location to the hand behavior.

2.3.4 Feature Extraction via Discriminative Spatial Patterns

Another popular technique in the BCI analysis literature is Common Spatial Patterns (CSPs) [157, 22, 34]. The motivation for this algorithm arises from the spatial filtering affected by the skull on the EEG signals. Blankertz et. al. observed that spatial filtering of signals with predefined filters such as bipolar and common average reference improves accuracy of a right vs. left imagery task [157, 22]. Further, they found that Laplace filtering of the channels provided even better accuracy. While these filters provided good accuracy, they concluded that optimizing the spatial filters to the acquired EEG data might provide subject-specific task-oriented spatial filters.

CSP is a technique used to analyze multi-channel recordings from two classes. This data driven approach maximizes the variance of spatially filtered data from one class while simultaneously minimizing the variance of the same spatial filter for the other class. Such filtering results in linearly discriminant features. This can be interpreted as maximizing the band (spatial filtering) power (variance) features from one class, while minimizing the same for the other. Thus, CSP provides an ERD/ ERS components of spatial filtering between the two conditions. Consider $x(t) \in \mathcal{R}^C$ and $y(t) \in \mathcal{R}^C$ be the multi-channel recordings from C channels at each time t and Σ_x and Σ_y be their average covariance of the signals estimated empirically from the data. The spatial filters that optimize the discrimination between these classes is given as follows:

$$w : w\Sigma_x w \gtrless w\Sigma_y w \tag{2.2}$$

CSP filters can be easily estimated with the use of a generalized Eigen vector formulation based on empirical covariance estimates derived from the training samples. Mathematically, they can be formulated as below:

$$\Sigma_D = \Sigma_x - \Sigma_y \tag{2.3}$$

$$\Sigma_C = \Sigma_x + \Sigma_y \tag{2.4}$$

$$w_{max} = \max_w \frac{w^T \Sigma_D w}{w^T \Sigma_C w} \tag{2.5}$$

$$w_{min} = \min_w \frac{w^T \Sigma_D w}{w^T \Sigma_C w} \tag{2.6}$$

Here Σ_D and Σ_C represent the discriminate and the common spatial components. Spatial weights w are identified using the generalized eigen vector formulation:

$$w : \Sigma_D w = \lambda \Sigma_C w \tag{2.7}$$

A large value of λ provides a spatial filter w that solves the maximization (2.5) and lower value of λ solves the minimization (2.6). In general, more than one spatial filter is designed to suit either end of the spectrum, i.e., one spatial filter has high variance for class x and the other for class y. The spatial filters project the raw multi-channel signals $z(t)$ onto a pseudo-channel or feature space as follows:

$$z_f(t) = wz^T z w^T \tag{2.8}$$

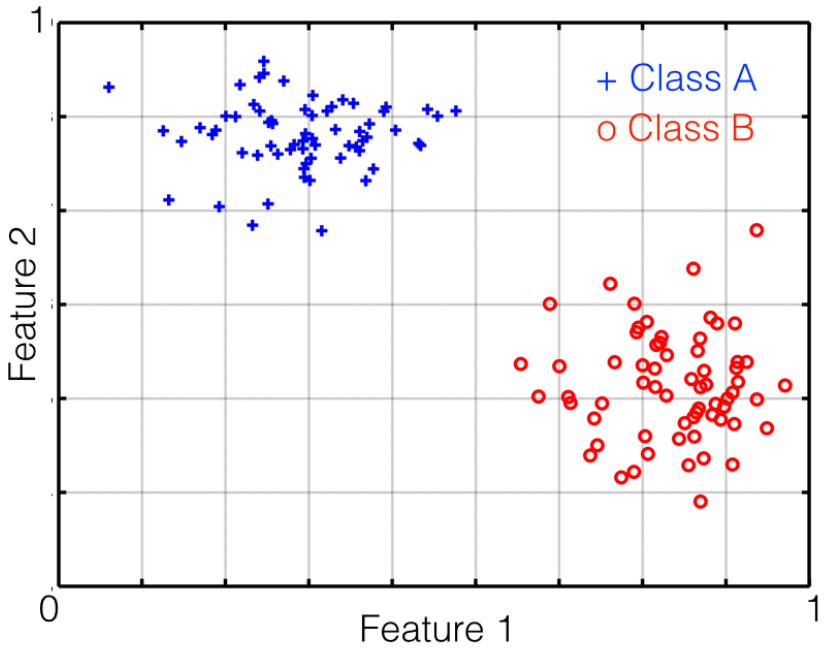


Figure 2.2: A representative Feature projection on two CSP filters. The two axis represent the normalized feature space when multiple trials are projected on the discriminative spatial filters. The different colors represent the features extracted for two different classes - here different directions.

Figure 2.2 represents the CSP feature space for two representative classes and easily classified with the use of a linear classifier. The ease and low complexity of estimating features makes CSP very popular. While some researchers use non-linear classifiers to classify the CSP features, I found that linear separability of the features achieves discrimination.

These filters provide excellent classification on various data sets with

not only binary conditions but also multiple classes via the use of multi-class classification [22, 20, 45, 205, 217, 34]. Apart from the accuracy of classification, the main advantage of CSP is its interpretability and visualization. The inverse of spatial filters can be viewed on the spatial maps as spatial patterns that discriminate the two classes. However, several parameters still need to be estimated before CSP can be applied including band-pass filtering, the time window of the data, and removal of any physiological artifact [22]. A longer time window of operation usually results in better accuracy because better estimates of channel covariance are estimated. However, longer window time results in latency of the BCI. Researchers evaluated different frequency bands for different tasks to arrive at the best functional frequency band. Some researchers focused on estimating the spectral as well as spatial components simultaneously using a similar framework. These filters, also called the Common Spectral Spatial Patterns (CSSP), are estimated by allowing delayed replicates of the channels [109, 46]. Using such a framework builds an optimal FIR filter at each electrode that provides both the spatial and spectral filtering. The number of latencies determines the order of the spectral filtering in the framework of an FIR filter.

The disadvantages of CSP include their sensitivity to outliers, artifact noise, balance of samples in the classes, and poor generalization due to overfitting to training data [160]. Artifacts such as blinking or muscle noise need to be removed before the analysis [22]. Since these artifacts result in high-energy components, they result in high eigen values and cause spurious results. The

intermittent noise that might occur only during a few trials may also affect the CSP estimation adversely. An equal or comparable number of trials per class is highly recommended to obtain a faithful result. In cases where there is an imbalance in the number of per class trials, the CSP tends to bias towards the class with higher number of samples.

Other variants of CSP filters have also been published in the literature including invariance CSP, regularized CSP , spatially regularized CSP, Tikhonov regularized CSP, weighted Tikhonov regularized CSP, and spatially regularized CSP [109, 205, 217, 115]. The underlying principle of all these regularization techniques is to embed the prior knowledge and obtain a better generalization of CSP. In general, regularization helps improve performance; while one of the regularizations provides better accuracy than the regular CSP, it is not clear which one improves performance apriori. Some researchers remarked that using a sparse CSP would aid in its generalization and performance [217, 70]. The motivation of such studies is to extract only those channels that aid in the classification of binary task by favoring sparse solutions. Further, the success of sparse solutions in other fields has favored researchers in this direction. While some studies used the ℓ^1 norm constraint, others have pursued a greedy approach to building sparse CSP.

Finally, CSP was developed as an analysis tool for a binary classification and provides channel weights that optimally classify training data. In applications that need classification of more than two classes, multiple classifiers are prescribed [91]. They offer poor performance in applications that

need tracking of continuous behavior. As such, every addition of new behavior requires retraining the entire model results in degraded performance. While initial studies with CSP filters were used with a cross-validation approach, the issue of poor generalization on chronologically recorded data has been well documented. The poor generalization of CSP is due to their data dependence in obtaining optimal filters [160]. Hence, advanced signal processing tools need to be developed to obtain better generalization.

2.3.5 Subspace Identification

Advanced signal processing algorithms extract recording subspaces from neural data. Independent Component Analysis (ICA) in BCI detects eye movement and muscle artifacts and removes noise [72, 182, 218]. Since the noise components are considered independent of the behavior related components, ICA provides a natural platform in applications like eye movement artifact removal, identifying single unit activity [149, 166]. The main principle of these approaches is to identify low-dimensional subspaces that model the non-stationarity of neural data. However, behavior relevant component identity poses a challenge in ICA. The methods build subspaces such that the signals from multiple repetitions of the behavior lie on these subspaces. Unlike Principal Component Analysis (PCA), which identifies only the highest energy component from the signal, these algorithms aim to identify the most recurrent and informative subspaces for neural data. The Iterative Subspace Identification (ISI) algorithm attempts to identify lower subspace representation of

intrinsic patterns from the new data [73]. The primary difference between ISI and Singular Valued Decomposition (SVD) is that while ISI identifies multiple subspaces of lower dimensions from the neural data, SVD tends to bundle them into a single subspace. ISI starts with an initial vector from the training set and identifies a sparse subspace from the remaining training vectors to represent the initial vector. Using this method provides a representation with minimal vectors used to represent any of the other vectors in the data set. After removing all the vectors that have been identified by the subspace, the same procedure is repeated to identify further representative subspaces. In the work done by Gowreesunker et al., ISI applied on neural data collected on a single day provided good lower dimensional subspaces [73]. However, these subspaces do not recur for multiple days and fail to provide the same level of representation in future data. In fact, the paper suggests that frequent update of the subspaces is necessary to obtain marginal improvements over existing traditional feature extraction techniques.

The above algorithms prove the existence of behavior specific spatio-temporal-spectral neural features that can be developed to accurately detect behavior. These algorithms provide benchmark decoding solutions and performance. However, these algorithms have been tested in a cross-validation environment where trials from the same data are resampled into training and testing session. Moreover, these features ignore any adaptation that might occur due to learning and other physiological reasons.

2.4 Conclusion

This chapter provided a brief background on the state-of-the-art in BCI research. This literature survey leads to following conclusions:

1. Most BCI are subject specific and need extensive training to build the decoding model and obtain satisfactory performance.

2. Feature extraction often determines the accuracy of the system. A robust feature space allows different classification tools to provide consistent performance over chronologically spaced sessions

3. Typically, spectral components of neural signals characterize specific subject behavior. Hence, neural signals are often band pass filtered to preprocess and obtain specific information

4. Spatial interaction between multiple neural locations varies with subject behavior. Identifying and modeling this interaction helps in decoding subject behavior

5. Signal Processing advances in analyzing multi-channel neural data leads to high performance in BCI

Chapter 3

Data

3.1 Introduction

This chapter provides an overview of data recording paradigm and details on experimental set up. It will first introduce the behavioral set up of the experiment including details on changes in external perturbations on monkey behavior. A practical Brain Computer Interface (BCI) should be capable of working in different environments and analyzing changes in neural patterns over different environments provide insights. The chapter also introduces neural extraction techniques and provides a brief overview on the initial analysis performed on decoding movement directions from Local Field Potential (LFP) recordings.

3.2 Behavioral Data

We (Dr. James Ashe, Dr. Giuseppe Pellizzer, Dr. Rahul Gupta, Dr. Firat Ince) trained two left-handed male rhesus monkey subjects (Macaca mulatta), H564 and H464, weighing 6.1Kg and 4.5Kg respectively to perform an instructed-delay center-out task to perform a point-to-point movement to visually displayed targets using a manipulandum (Interaction Motion Tech-

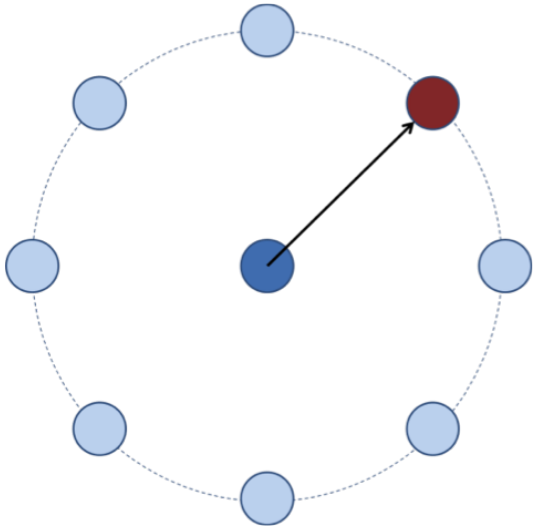


Figure 3.1: Schematic of the eight targets presented to the monkey. Each target is placed 10cm away from the center hold target. The monkey initially places the cursor in the center target and waits for one of the targets to appear, thus providing a visual cue. It then proceeds to move the the cursor in the highlighted target by moving the joystick in the appropriate direction.

nologies, Cambridge, MA). This two-joint manipulandum is commonly used in force-field studies in both human and non-human primates. The monkeys sat in front of a monitor that displayed the center and target location. When instructed by an experiment with the aid of visual cues, the monkeys were trained to reach one of the eight equally spaced targets located around a circle of around 9cm. The monitor was also a source of visual feedback to the monkey. A schematic display is shown in figure 3.1.

Each trial began when the monkey placed the cursor inside a circular window of radius around 1cm, at the center of the display. The monkey was trained to hold the cursor for at least 800ms to ensure a control period. Any

minor perturbations in this circular target were acceptable, but deviations over this circular window resulted in an error and the monkey had to restart the trial. The control period is sometimes referred to as the center-hold period. After ensuring that the cursor was held in this location, a random generator chooses one of the eight targets, highlighted, and displayed for about $500 - 700$ms on the display. This served as the visual cue for the monkey. We instructed the monkeys to hold their cursors in the center of the display during this cue and memorize the location of reach. An early start of the reach during this cue resulted in an error. We switch off the target location for a memory delay time of about $800 - 1000$ms during which the monkey memorizes the target location.

Following the memory delay, the targets reappeared on the screen and the monkeys reached the target during this cue in $800-1000$ms. Both monkeys completed all successful reaches to the peripheral circular targets ($\sim$ 1cm.) under 1000ms to avoid error. Upon reaching the target, the monkeys held the cursor in that position for about 800ms. The successful completion of each reach to the target and in the appropriate periods resulted in a correct trial. The monkey obtained a juice reward for its success. A time line for each trial with the median times spent in each period is shown in figure 3.2. The next trial began after an inter-trial interval of 500ms.

Since monkeys learn the timing of these experiments very well, all the experiment times were pseudo-randomized to ensure attention during the entire trial. In addition, the targets were presented in a pseudo-random fashion

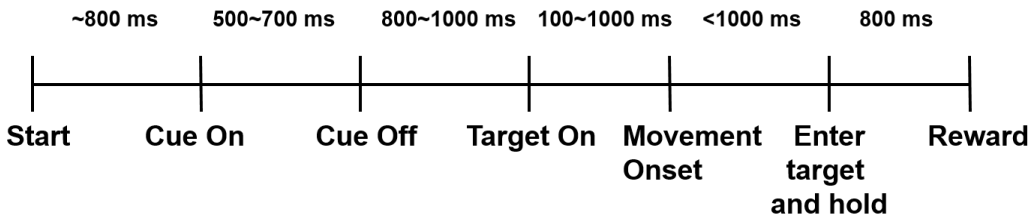


Figure 3.2: Time-line of each trial performed by the monkey. The number during each epoch indicates the median time spent during that period.

in sets of eight. To move to the next set, the monkey needed to reach all eight presented targets without any timing errors. Behavioral data in the form of position and velocity of the cursor, forces and torques applied by the monkey at the handle of the manipulandum were sampled at 200Hz and stored for analysis. These forces are different from the forces applied by the manipulandum and will be discussed later. All timing information regarding each of the single trials like the visual cues - ON and OFF, Target Go cue, Movement Onset time, and Target Reach time were also stored (see Figure 3.2). Only correctly performed trials (determined online at the time of recording) were stored for further analysis. During all the sessions, the monkeys accurately reached more than 70% of all presented targets in a timely manner.

Depending on the motivation of the monkey and its skill at reaching the targets, the number of accurate trials performed during each recording session varied from 88 to 520. Initially, we recorded only one session per day and later progressed to multiple sessions based on the monkey's motivation. We recorded over a period of 4 weeks (H564) and 6 weeks (H464). Once the monkeys grew

accustomed to the above behavior and performed target reaches with fewer errors, external forces were applied to the manipulandum to study the forces' effects on learning and behavior of the monkey. These forces diverted the cursor away from the target. To obtain the juice reward, the monkey had to work against these external field forces and reach the target. We applied the force fields only during the reach period and applied no force when the manipulandum was in the center or at the targets. Forces depended on the velocity (viscous fields) or the position (stiffness fields) of the manipulandum and acted in a direction perpendicular to the direction of motion (curl fields). Mathematically they can be expressed as

$$\mathbf{F}_x = -bv_y; \mathbf{F}_y = bv_x \tag{3.1}$$

$$\mathbf{F}_x = -ky; \mathbf{F}_y = kx \tag{3.2}$$

, where $\mathbf{F}_x$ and $\mathbf{F}_y$ are the x, y components of the forces corresponding to the v_x and v_y components of the velocity of the manipulandum, b is the viscosity coefficient, x, y are the horizontal and vertical components of the position of the manipulandum, and k is the stiffness coefficient. The stiffness and viscosity coefficients were chosen to be non-zero, and have enough effect to cause diversions in the natural movements while ensuring that the monkey could still perform the task. Depending on their direction and type, the forces were Stiff Clockwise (SCW), Viscous Clockwise (VCW), and Viscous Counter Clockwise (VCCW). However, the nature of forces against each direction in a particular session remained unchanged.

3.3 Neural Recordings

To record neural signals two silicon based Utah electrode arrays (Blackrock Microsystem, formerly Cyberkinetics, Foxboro, MA) were implanted in the contralateral arm areas of primary motor (M1) and dorsal premotor (PMd) cortices respectively. Electrodes on the array were arranged in 4mm x 4mm square with an electrode depth of 1.5mm and an inter electrode spacing of 0.4mm. Although each array was capable of acquiring up to 96 channels, the available amplifiers reduced the total capability to 64 channels per grid. Signals from the arrays were initially sampled at 44KHz. Spike sorting was performed using the Plexon Offline Sorter (Plexon Inc. Dallas, TX). Spikes were simultaneously recorded from a varying number of cells in the M1 and PMd areas of both monkeys. Offline analysis of the spike rate was used to predict behavioral data, position and velocity of the cursor, and the forces applied by the monkey on the manipulandum [77]. This analysis established that prediction of these parameters requires only a linear analysis tool in the form of linear regression and shorter neural stream of data ($\sim$ 420ms) to make accurate predictions. The analysis involved a 10-fold cross validation to obtain decoding accuracy and correlation coefficient between the estimated and actual variables. However, since the number of single units varied over different recording sessions, it is difficult to comment on the long-term applicability of such analysis.

The analysis of single units ignored the lower frequency component of the signals. However, their long-term stability is an attractive trait that

becomes very useful for practical BCI applications [135, 212, 211, 198]. In this study, the Local Field Potentials (LFPs) related to these recordings were filtered at $0.3 - 500$ Hz and then stored at 1 KHz sampling rate. We visually inspected channels to determine channels free of artifacts and any power line noise. We removed such channels from the analysis and retained 75 (H464) and 88 (H564) channels for further analysis. Figure 3.3 shows the channel locations in the M1 and PMd of the two monkeys.

This data was then filtered between $0.3 - 220$ Hz and down sampled at 500 Hz. Although the signal was filtered at 0.3Hz at the hardware level some baseline wander was observed in channels for many trials. To eliminate such wanders, we sub-band filtered and removed the DC trend by linearly de-trending the recordings. This minimized outliers due to DC shifts. After the de-trending step, we proceeded to identify the most reactive sub-band of LFP activity. During the time-frequency analysis of the waveforms, it was observed that distinct patterns existed in $0 - 4$Hz, $4 - 10$Hz, $14 - 30$Hz, and $48 - 200$Hz bands. Of these frequency bands, the $0 - 4$Hz band was modulated differently across all directions while other frequency bands provided similar modulations across all directions. We obtained close to $\frac{1}{f}$ frequency response of the LFP activity similar to other reports in the literature [158, 81, 13, 155]. This indicates a high-energy content in the lower frequency band that might aid in discrimination of directions. Other studies have also indicated that this frequency band is particularly active in direction decoding. Hence, our feature extraction was based on this frequency band component [91].

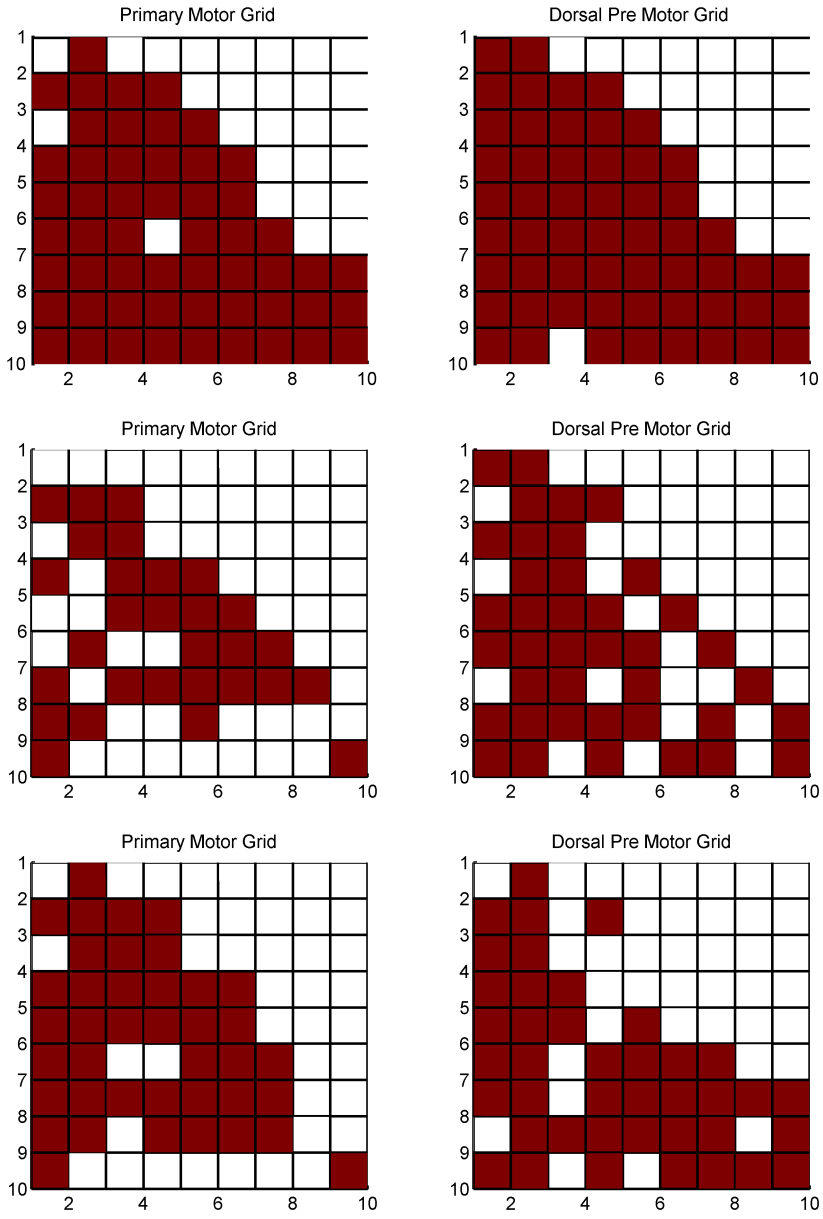


Figure 3.3: Spatial Arrangement of the electrode grids in the Primary (M1) area and Pre Motor (PMd) Area. The first row represents the spatial arrangement of channels. The second and third rows represent the locations of non-noise channels in monkeys H464 and H564 respectively. The electrodes are placed on a 4 mm $\times$ 4 mm grid with an inter-electrode separation of 400 μm. For ease of reading they are indicated with units in the figure.

3.4 Initial Analysis

Ince et al. performed the initial analysis on the data and proposed to use a regularized Common Spatial Pattern (CSP) to analyze the data [91]. The choice of using correlation based spatial patterns is motivated by slow baseline shifts observed for different directions. Using a centered covariance matrix approach removes such wanders from the signal and results in the loss of information content. Hence, using a correlation between multiple channels might result in better accuracy. Such approaches also provided better decoding in classifying slow-moving cortical event related synchronization or desynchronization in EEG [44].

For purposes of establishing use of LFP for target decoding, they proposed to decode the movement direction from features derived using multi-channel LFP recordings. Focus of the analysis was on analyzing 1s of data after movement onset; LFP activity in this 1s was used to extract CSP filters and features. They observed that the correlation between channels varied with different directions and incorporated this observation to derive CSP features from the correlation of multi-channel data. The proposed CSP model follows the below feature extraction algorithm:

CSP is inherently designed for binary classification. Since the direction decoding is a multi-class (8 directions) discrimination, they proposed the use of redundant classifiers in the form of

1. multiple pair-wise classifiers that classified each direction from the other

2. multiple hierarchical classifiers that classified a group of directions against a diametrically opposite set of directions

A schematic of such hierarchical classifiers is presented in figure 3.4. This choice of hierarchical classifiers was motivated by the observation that neighboring directions, e.g. 0°and 45°, produced spatially similar neural patterns during their reaches. Hence, trials from neighboring directions can be grouped to obtain a stable neural pattern. To decode K directions they proposed to use $K(K-1)/2$ pairwise classifiers and 12 hierarchical classifiers. During the training of these classifiers, trials belonging to the directions are collected and a spatial pattern capable of classifying them is constructed. For the hierarchical classifier this means that trials from neighboring directions were lumped into a single class. Applying these spatial filters on the band-filtered LFP data, results in a CSP feature space, where the classes can be easily separated using a Linear Discriminant Analysis (LDA) classifier. The CSP training model includes spatial filters for each classifier and the resultant weights from the LDA classifier.

To decode the direction of a new trial, all the spatial filters from the classifiers are applied. This provides class labels for each of the classifier. Using Error Correcting Output Codes (ECOCs) provides the scores for all eight directional classes. For the given hierarchical classifiers, the class could define more than one direction. Hence, combining the scores of both —pairwise and hierarchical classifiers —provides a better estimate of the direction. The estimated direction is the one with the maximum score.

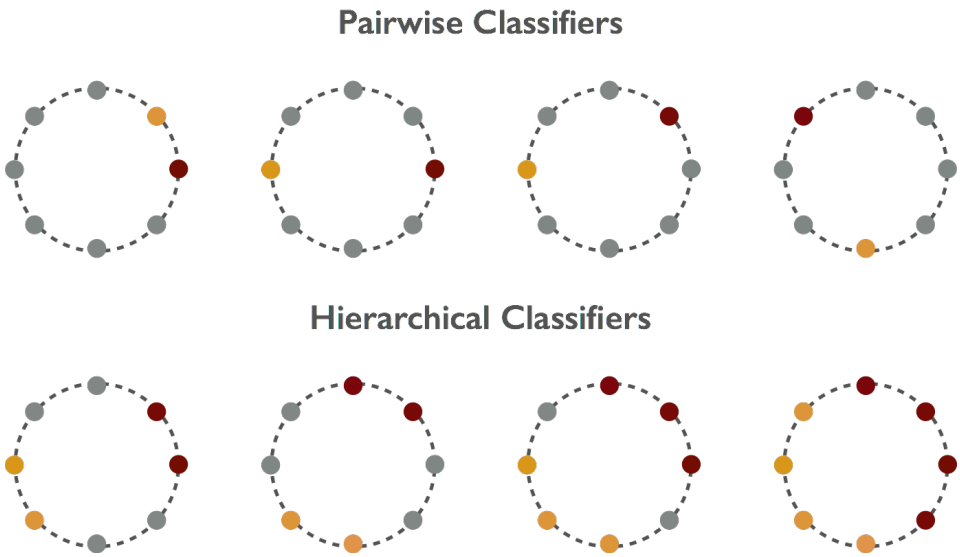


Figure 3.4: Grouping of directions using the error correction output codes. Each group represents a super class representing all the direction in the group.

To evaluate the performance of this method and to test the feasibility of using LFP for movement decoding, [91] used a 10 x 10 cross-validation over data collected from multiple days spanning over one week. Sessions 1, 2, and 3 from each monkey are used in the analysis and resulted in 508 (H564) and 1107 (H464) trials. Such a cross-validation approach allows capturing variability of spatial patterns over these days and the spatial patterns can be modeled efficiently. This analysis was carried out at multiple time instants after movement onset - each time sampling 1s of data prior to the snapshot and modeling spatial patterns for each time instant separately. The overall accuracy of the model was defined in terms of decoding power (decoding accuracy) as the fraction of accurate estimates of the direction to the total number of testing trials.

As there are eight possible directions of movement, the decoding power for a random classification is 12.5%.

The analysis obtained the best decoding results with regularized CSP for the two monkeys reaching 80% (H564) and 92% (H464). They found that the regularized CSP based on correlation of channels provided better decoding results. This shows that signals in these bands have some useful information that helps in discriminating the movement directions. Results from this analysis also proved that slowly varying signals under 4Hz. contain direction decoding information. The use of ECOC based hierarchical patterns also added an average of 10% to the pairwise classification results. We also implemented the one-vs-rest classification method and it failed to provide decoding levels close to the other methods. This shows that the use of multiple redundant classifiers produces better performance.

These analyses prove feasibility of LFP to decode movement directions under assumptions of having knowledge of entire variability of data in the form of trials from multiple sessions. Practical BCIs do not have that advantage; they generally have to be modeled on data from a single session when the BCI user provides calibration data. While daily calibration of BCI systems is an option, it increases the burden on the user to provide a training session that is not an elegant solution. Cross-validation results and analysis provide a good tool to test the feasibility of an algorithm along with providing a generously optimistic estimate of the algorithm's performance in a practical scenario. Hence, we extended the decoding analysis with just trials from session 1 for

Table 3.1: Decoding Accuracy of CSP filters over multiple days of testing. To mimic a practical BCI, I trained the decoder on Session 1 (Day 0). These results and the method provide as baseline comparisons to the proposed algorithm.

Day after Training	8	9	13	14
H464				
Number of Trials	263	325	348	88
CSP	38%	41%	16%	13%
H564				
Number of Trials	206	103		
CSP	41%	40%		

training and the rest for testing. This allowed us to evaluate the performance of different BCI algorithms in a similar setting.

Table 3.1 presents the decoding power over two weeks for monkeys H464 and H564. Decoding models used redundant CSP based feature extraction and LDA classifiers; combining their results using ECOC to obtain the final estimated direction for a test trial. Since the model was trained on a single session, it only provides a good illustration of that session. The inherent data variability of LFP recordings result in poor performance of the algorithm over multiple days.

To overcome the daily variability of LFP signals, I conducted another analysis focused on obtaining low dimensional subspaces that explain training data [73]. The motivation behind this analysis is to estimate subspaces that recur during the reaches to each direction. The model uses an iterative subspace identification approach to build multiple representative subspaces per

direction. During training, subspaces are extracted to faithfully represent the data from each direction. Projection of a trial on to these subspaces produces direction related components, and I hypothesized that accurate directions can be estimated from such information.

For a test trial, channel data is projected onto a subspace to measure its correlation with that subspace. It is assumed that the subspaces can extract direction component from the data and can be used in classification. This analysis used models based on CSP to identify if the projected component of the signal belonged to a particular class. Finally, the decision from the CSP classifiers is matched with the directionality of the subspaces. The decision is accepted only if the match is exact. Ideally, the projection onto a direction subspace would be enough to classify the direction of a trial and directionality of the subspace remains consistent over all recording sessions. The subspace extracted from directions in the training session represents same direction even in the testing session. However, in practice due to day-to-day variability of the signal, it is unclear if projection onto a given subspace could represent the same direction. In fact, I observed that the subspace fitting one direction in training session provided better representations of other directions during testing. Hence, I built a classifier that assumes knowledge of supplementary information on the directionality of the subspace.

To compare the efficiency of this approach I used the same training and testing sessions as used for the CSP algorithm in the previous analysis [91]. This algorithm uses the traditional covariance based CSP as a feature extractor

Table 3.2: Decoding Accuracy of using Low Dimensional Subspace projection with CSP filters over multiple days of testing. To provide a fair comparison, I used the same parameters for the CSP filters.

Day after Training	8	9	13	14
H464				
Number of Trials	263	325	348	88
CSP	38%	41%	16%	13%
Low Dimensional Subspaces	58%	54%	27%	24%
H564				
Number of Trials	206	103		
CSP	41%	40%		
Low Dimensional Subspaces	44%	42%		

on data projected on a subspace. The subspaces contribute by extracting only direction related component from band-filtered channels. When this approach was applied in a cross validation setting, decoding powers improved slightly from 75% to 88% (H564) and 89% to 90% (H464). In scenarios with different training and testing sessions, this method provides a marginal improvement on the CSP results. Table 3.2 provides a comparison of decoding powers over the two-week period with the two algorithms. Finally, to obtain the best decoding the algorithm required trials from multiple sessions to capture data variability over days.

3.5 Conclusion

This chapter provided an overview on the data-recording paradigm and experimental setup used to collect neural data. The main objective of my thesis

is to develop a robust LFP decoder with minimal training. Initial analysis on the data provided a performance benchmark. The findings from this analysis and published work in [91, 73] lead to interesting conclusions:

1. Unlike Single Unit Activity, LFP provides consistent signals over multiple days

2. Spatial Patterns that define each behavior over a period of a week could be estimated and observed to provide good decoding of eight directions. The construction of these spatial patterns requires that most of the data variability be captured in training set

3. When such data variability is not captured by training only on a single session, the performance degrades and reaches a random classification in 2 weeks

4. Developing multiple redundant classifiers provided better performance than using pair-wise classifiers

5. Low dimensional subspaces of the data exist in a linear space, but only a few subspaces are recurrent. Specifically, apriori knowledge of the subspace clustering and labeling is needed to obtain any decoding

6. It is not clear if even the subspaces learned from the same day can provide any better decoding than those learned from the training day

7. High success in the cross-validation studies leads to the conclusion that certain spatial patterns are recurrent over multiple days, but no method to extract them exists

After studying different features for their robustness over multiple days, I hypothesized that developing robust features helps in building a robust classifier that provides consistent results over multiple days and over varying environmental conditions. To this end, I proposed to use qualitative features in the form of instantaneous inter-channel power ranks instead of raw channel data. Chapter 4 discusses motivation of using these features and analysis of using such features on the data.

CSP approach assumes Gaussian distributed spatial patterns that can discriminate a binary task. In specific, estimating mean covariances for each of the behaviors estimates the mean neural pattern for that behavior. Since this is a data dependent algorithm, variance of such a Gaussian is large enough to fit the entire data. To overcome the shortcomings of this method I theorized that more than one spatial pattern is required to describe a behavior (see Chapter 5). I propose to describe the neural spatial patterns with multiple Gaussian distributions to build such a model. The goal of model training is to accurately estimate mean and variance of the spatial pattern.

Model adaptation is essential to build a robust BCI. I propose a strategy for model adaptation that is un-intrusive and requires minimal calibration from the user. Such an approach results in improved and consistent decoding

performance over multiple days including during changes in the environments. In conjunction with a multiple spatial pattern model, I characterized model learning and spatial changes that occur during the learning process (Chapter 6). I observed that in the multi-spatial pattern model some of the spatial patterns recur over time and aid in robust classification, while some other patterns represent perturbations in the data due to day-to-day variations. Characterizing them enables efficient modeling of recurrent spatial patterns.

Chapter 4

Robust Feature Extraction

4.1 Introduction

Chapter 3 discussed the data collection paradigm and experimental set up. During the initial analysis, Common Spatial Patterns (CSP) were used to extract neural spatial patterns from Local Field Potentials (LFP) and decode eight hand movement directions. Due to changes in the spatial patterns over multiple recording sessions, these discriminating spatial patterns developed on a single session were ineffective. I hypothesized that investigating the invariance characteristics of extracted features boosts decoding performance. This chapter presents my motivation and contributions in extracting features that remain robust to variations in LFP recorded over multiple days.

4.2 Extracting Qualitative Spatial Patterns

The initial analysis of LFP data used CSP patterns to decode the eight directions of movement [91]. These results indicate existence of a direction specific spatial pattern in LFP data. This implies that when a particular directional target is reached, the neural channels are arranged in a unique manner that enables the decoder to classify these directions. To investigate

these spatial patterns, I analyzed power patterns calculated in fixed time windows in each channel. I used rectangular moving time-windows to evaluate temporal evolution of such patterns. Visualization of such patterns on the electrode grid revealed existence of patterns unique to a direction [195]. I observed that while the spatial and temporal evolutions of direction reaches were consistent over multiple trials, power level across trials was inconsistent. Such inconsistencies in power levels lead to variability in extracted features. Current state-of-the-art machine learning techniques require that features extracted from the testing session closely resemble those in the training session [80, 156]. Pattern recognition techniques depend heavily on stationarity of extracted features. While variability of power in a single session could be modeled, modeling changes in power levels across multiple days is very difficult.

Normalization techniques could be used to overcome such baseline shifts in power levels. However, normalization techniques require explicit knowledge of entire test data. In particular, normalization requires knowledge of dynamic range parameters - minimum and maximum signal deviations - over the entire training session. These kinds of techniques work well for cross-validation analysis that has such parameters readily available. Analyzing model consistency over subsequent recordings requires knowledge of parameters from the testing sessions. Hence, normalization techniques are difficult to implement and do not provide a solution to the problem [195].

In statistical analysis, variables that cannot be parametrized by a nor-

mal distribution are analyzed using non-parametric solutions [39]. Ranking is one such technique that preserves the order of the variables, while ignoring the dynamic range and actual values. They have the following advantages over normal parametrization methods:

1. Rank ordering is preferred over standard parametric methods when the assumptions of normality are grossly violated.

2. Presence of outliers affects the parametric methods, while rank-based methods ignore such outliers as they provide them with the same value as any big (or extremely small) original value.

3. Rank ordering also handles missing data better than parametric methods, by giving the missing data a zero rank or a very large rank. Ranked variables are generally uniformly distributed and offer a better power of the test in those cases where normality assumptions are violated. The relatively poorer performance of ranking occurs due to loss of details from the original variable and only when parametric assumptions are satisfied. It is generally accepted that in such scenarios power of the rank test is about 0.95 times power of the parametric test.

Ranking has been suggested to overcome variable dynamic range of multiple photographic plates [210]. Using such an approach resulted in obtaining robust features to classify objects in astronomical images. Ranking of the "raw" features in this study eliminated dependence on their probability distribution.

Finally, even if parametric ranges are normalized, the ranking of features preserves the order and allows use of parameters like maximum and minimum intensities.

I performed similar analysis to observe changes in power levels over multiple days. The distribution of raw features varies significantly across multiple days. However, I observed that the locations of high and low power ranks on electrode grid remained consistent. I analyzed the rank of channels instead of their original values. In the proposed approach, I ranked channels of each trial according to their power. At each time sample, channel power is calculated using a 250 ms rectangular time-window preceding the sample. The channel with highest power is assigned rank 1, the next channel rank 2 and so on in descending order. Ordering could also be done in ascending manner of rank powers. In this manner, the total number of channels in the analysis is the upper bound value of the rank feature. This feature extraction bypasses the dynamic range of channels, while retaining the relative information between channels. Figure 4.1 shows a typical trial and evolution of its corresponding rank features. The figure also shows that change in cognitive states (around 0.6s after movement onset) that could be inferred by analyzing the organization of ranks.

I analyzed only the top ranking channels using different ranking schemes like regular ranking and standard competition ranking. In standard competition ranking, competitors with the same score get the same rank enabling a fairer ranking. In the context of multi-channel LFP, this reduces to aggregat-

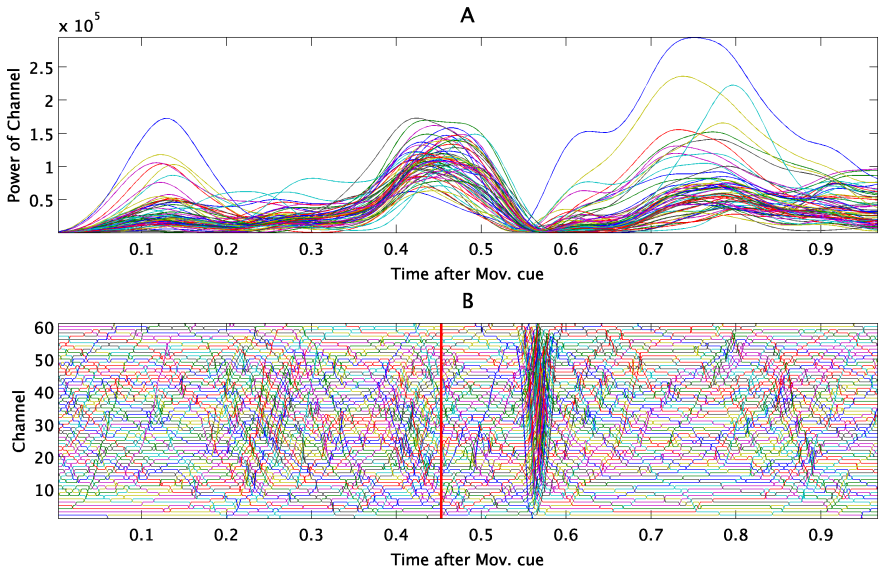


Figure 4.1: A representative trial and its rank power features calculated over its duration. The first plot shows the power modulation of the channels when a particular target is reached. The second plot shows the changes in the rank patters for the same trial.

ing channels that have similar powers and ranking them with a similar value. While this scenario often occurs in competitions, it is highly unlikely that two channels have the same power (unless they are zero). I adapt this ranking system by using a fractional threshold that measures similar channel powers. The process is described in algorithm 1.

The analysis consists of training and test phase. During the training phase, rank pattern features are extracted from single trials. To test efficacy of the proposed feature extraction methods using rank patterns, I tested its performance using Common Spatial Patterns (CSPs). This provides us a good

Algorithm 1 Calculate Ranks in Standard Competition Rank Scheme

Obtain Standard Competition Ranks $\mathbf{R}$ from N channel raw data X
Calculate Channel Power: $\mathbf{P} = \sum_{t-\tau}^{t} \mathbf{X}^2(t)$
Initialize: $r = 1, \Psi = \{1 : N\}, \mathbf{R} = 0$
while $\Psi! = \emptyset$ **do**

Find the highest powered channel in the pool

$$c^* = \arg\max_{c \in \Psi} \mathbf{P}_c$$

Select Channels in the range:

$$S_c = \arg_c \mathbf{P}_{c^*}(1 - f_{th}) \leq \mathbf{P}_c \leq \mathbf{P}_{c^*}$$

Assign selected channels rank:

$$\mathbf{R}(S_c) = r,$$

$$\Psi = \Psi - S_c,$$

$$r = r + |S_c|$$

end while

benchmark performance. I extract CSP filters on data transformed to rank space. Thus, this method provides an estimate of qualitative measurements rather than quantitative measurements. Figure 4.2 presents the flowchart to compute the discriminant spatial filters from the neural signals. As an example, I present one of the CSP filters on the pre-motor area of monkey H464 in figure 4.4. As a reference, figure 4.3 presents the electrode arrangement on the pre-motor grid. The spatial filtering operation shown in the flowchart is only used for visualizing the spatial filters.

Figure 4.4 also illustrates the consistency of the spatial patterns over

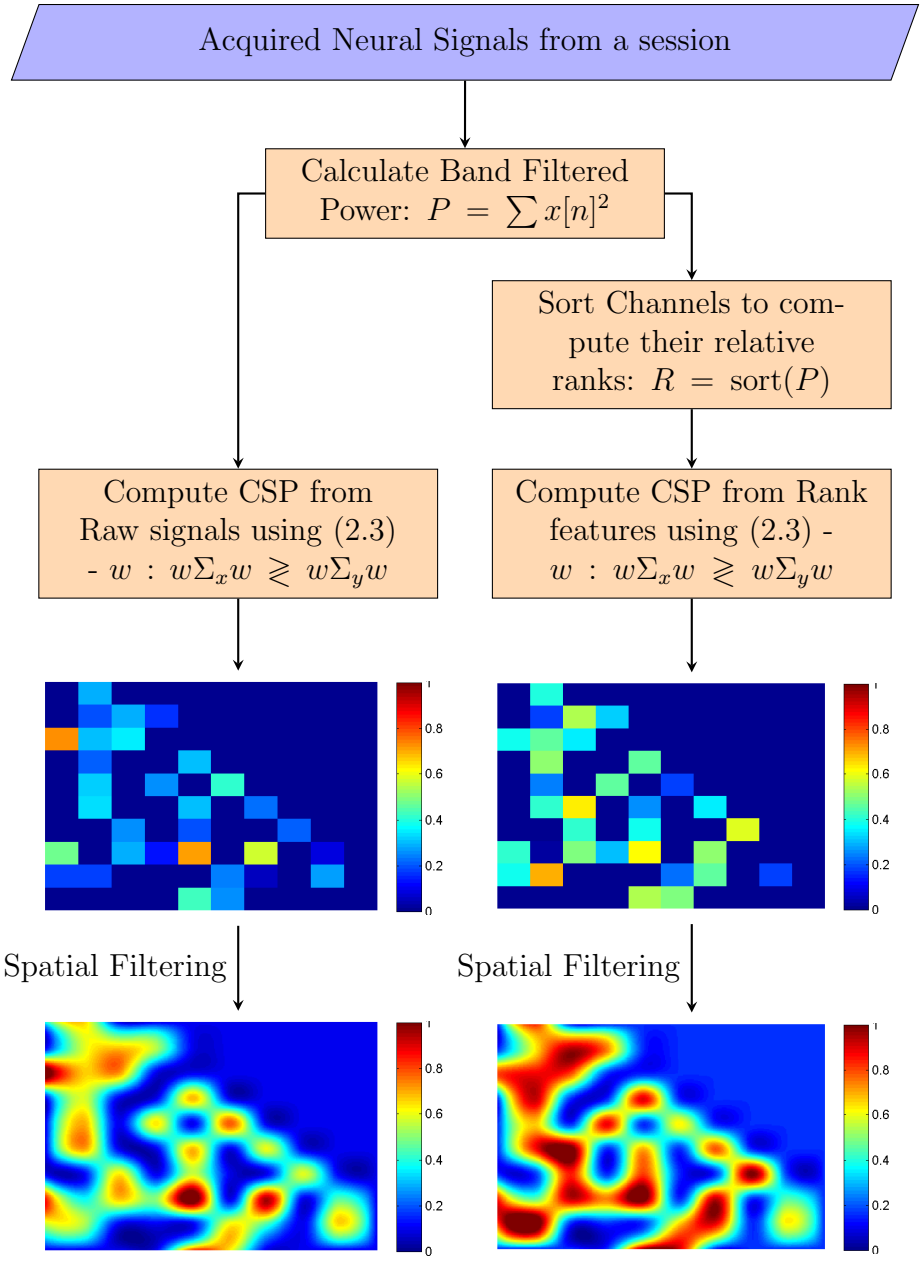


Figure 4.2: Flowchart to calculate the spatial discriminating patterns. The figures shows the spatial weights obtained from CSP on the Premotor spatial grid. The electrodes are placed on a $4mm \times 4mm$ grid. The location of each electrode is represented by a single pixel before spatial filtering. For the ease of visualization a spatially filtered version of the filter is also shown.

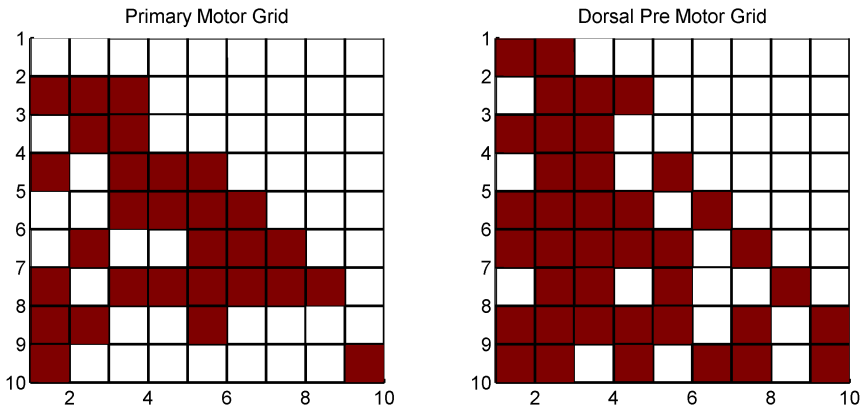
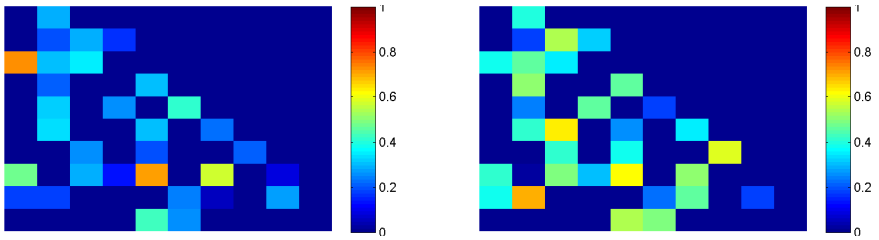


Figure 4.3: Spatial arrangement of electrodes on Primary and Pre-motor area of H464. The grids are each placed on a $4mm \times 4mm$ grid, with an inter-electrode spacing of $400\mu m$.

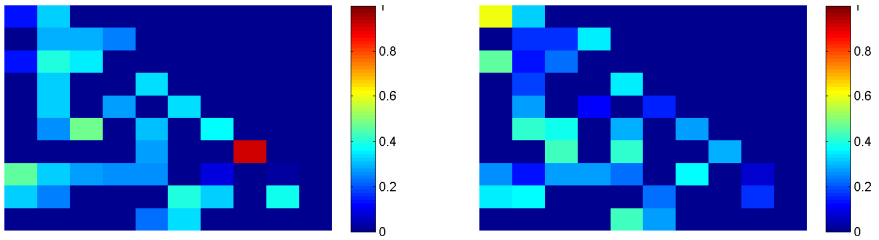
different sessions. I employed pairwise and hierarchical classifiers from an earlier analysis in chapter 3 to ensure fair comparison of methods. Linear Discriminant Analysis (LDA) classifiers are built on the Rank Features obtained from training session. During the testing session, spatial filters and their corresponding LDA classifiers are applied on each single trial. Decoding matrices similar to the ones presented in previous chapter are used here.

To evaluate the performance same training (session 1) and testing sessions (the rest) are used for both CSP and Rank CSP. While non-linear classifiers might provide better decoding, initial analysis focused on comparing the feature performance. I initially performed cross-validation tests and observed that both CSP and Rank CSP perform at the same level providing high decoding accuracies. This result is expected in a cross-validation setting courtesy of the CSP framework. Further, I tested feature performance to overcome long-



(a) Raw CSP on Day 1

(b) Rank CSP on Day 1

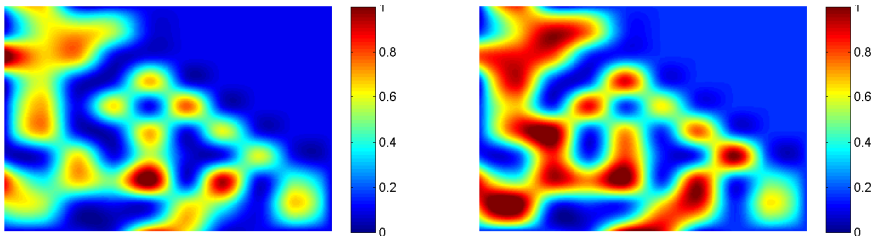


(c) Raw CSP on Day 8

(d) Rank CSP on Day 8

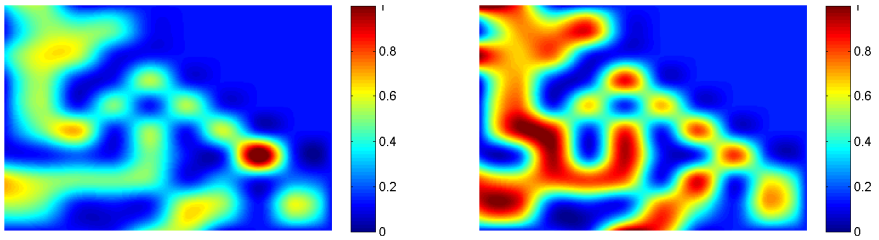
Figure 4.4: Spatial Patterns extracted on raw and rank features over two days. The spatial patterns represent the average LFP signal on the same Pre Motor grid area ($4mm \times 4mm$) when monkey H464 reached direction 0. Each pixel represents the location of each electrode on the LFP grid. (a) represents the spatial pattern derived from raw LFP on session 1. (b) represents the spatial pattern derived from rank LFP features on session 1. (c) and (d) represent the same for session 2 (collected on day 8) respectively.

term variations of LFP signal. Results of this analysis are presented in table 4.1. I find that the rank features provide robust decoding compared to the raw features ($p < 0.01$). These results suggest that feature extraction method plays an important role in the performance of CSP filters. This conclusion also holds true when training and testing sessions are selected from sessions where external field forces were applied. In this scenario, VCCW field force was applied during hand reaches of the training session and field forces varied during test sessions. I also studied performance of the algorithm with varying



(a) Raw CSP on Day 1

(b) Rank CSP on Day 1



(c) Raw CSP on Day 8

(d) Rank CSP on Day 8

Figure 4.5: Spatial Patterns extracted on raw and rank features over two days. These patterns are the filters shown in figure 4.4 but spatially filtered over the premotor cortex using an interpolation operator. The spatial patterns represent the average LFP signal on the Pre Motor grid area of $4mm \times 4mm$ when monkey H464 reached direction 0. (a) represents the spatial pattern derived from raw LFP on session 1. (b) represents the spatial pattern derived from rank LFP features on session 1. (c) and (d) represent the same for session 2 (collected on day 8) respectively.

fractional thresholds and concluded that the ranking approach provides only marginal improvements over the traditional ranking system. Further, it is very difficult to know apriori which fractional threshold provides best decoding performance.

Table 4.1: Decoding Accuracy of the time evolving spatial pattern model. Both Rank CSP and the Time Evolution Models perform better than the state-of-the-art CSP model ($p < 0.01$).

Day after Training	8	9	13	14
H464				
Number of Trials	263	325	348	88
CSP	38%	41%	16%	13%
Rank CSP	62%	61%	55%	47%
Time Evolving Spatial Pattern	71%	69%	58%	62%
H564				
Number of Trials	206	103		
CSP	41%	40%		
Rank CSP	53%	52%		
Time Evolving Spatial Pattern	66%	60%		

4.3 Temporal Evolution of Spatial Patterns

CSP algorithm assumes that the correlation of electrodes that determine the spatial weights remains constant over entire reach to the directional target. I investigated the evolution of inter-electrode correlation over the entire reach by analyzing the changes in spatial patterns over time using small time-windows [193]. Each trial is divided into non-overlapping time-windows of a small size, say Tms, determined by cross-validation. For each of these time-windows spatial patterns are computed using CSP method to discriminate the directions. During testing stage, each time-window of a trial is evaluated with the spatial pattern of its corresponding window. I compute and store the LDA distance for each time-window and obtain the final decision based on the entire trial length. The proposed algorithm presented in Algorithm 2 assumes

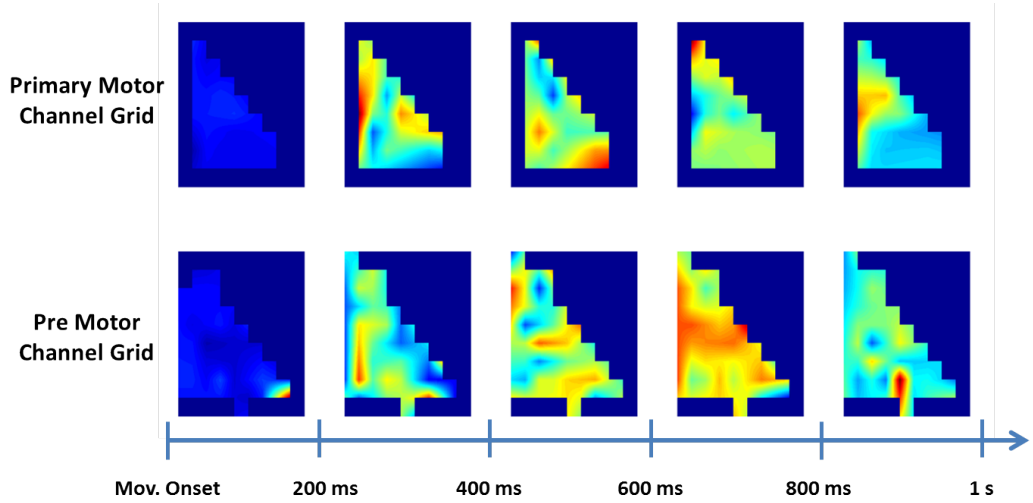


Figure 4.6: An example of time evolving spatial pattern obtained using the Algorithm 2. Each grid represents the $4mm \times 4mm$ spatial grid sampled on the primary and pre-motor area of the monkey cortex. For ease of visualization, the grid is also spatially filtered using an interpolation operator.

a binary classification problem and can be extended to multiple classes using redundant hierarchical classifiers. Figure 4.6 shows an example of a spatio temporal pattern obtained using this algorithm.

Table 4.1 shows the decoding performance of the proposed model for H464 and H564. For both these monkeys, the training was performed on the first session. I observed that all the methods perform poorly when field forces in testing session are different from those of training session. Performance returned to a stable level when the original field force was reintroduced. These results highlight the need of using evolving spatial patterns of electrode ranks in the context of robust movement decoding in LFPs.

Algorithm 2 Algorithm to identify Temporally varying Spatial Patterns

Training Stage: Learn $\mathcal{M}$ from Data:$\{\mathbf{X}_i(t), \mathbf{D}_i\}_{i=1}^N$
for $t = 0 \rightarrow$end of trial **do**

Obtain Spatial Pattern using CSP:

$$w_t = \max_w \frac{w^T \Sigma_D w}{w^T \Sigma_C w}$$

, where Σ_D and Σ_C are calculated using eq ((2.3))

end for
Compute LDA classifiers for each CSP filter: $\{w_L, b_L\}$
$\mathcal{M} \leftarrow \{w_t, w_L, b_L\}$

Testing Stage: Estimate $\hat{\mathbf{D}}_i^*$ from Data:$\{\mathbf{X}_i^*\}_{i=1}^N$, $\mathcal{M}^*$
for $t = 0 \rightarrow$end of trial **do**

Calculate CSP feature: $f = w^T \mathbf{X}^T \mathbf{X} w$

Calculate classification score: $\alpha_t = f w_L + b_L$

end for
Estimate $\hat{\mathbf{D}}^* = \text{mode}_t\ \alpha_t > 0$

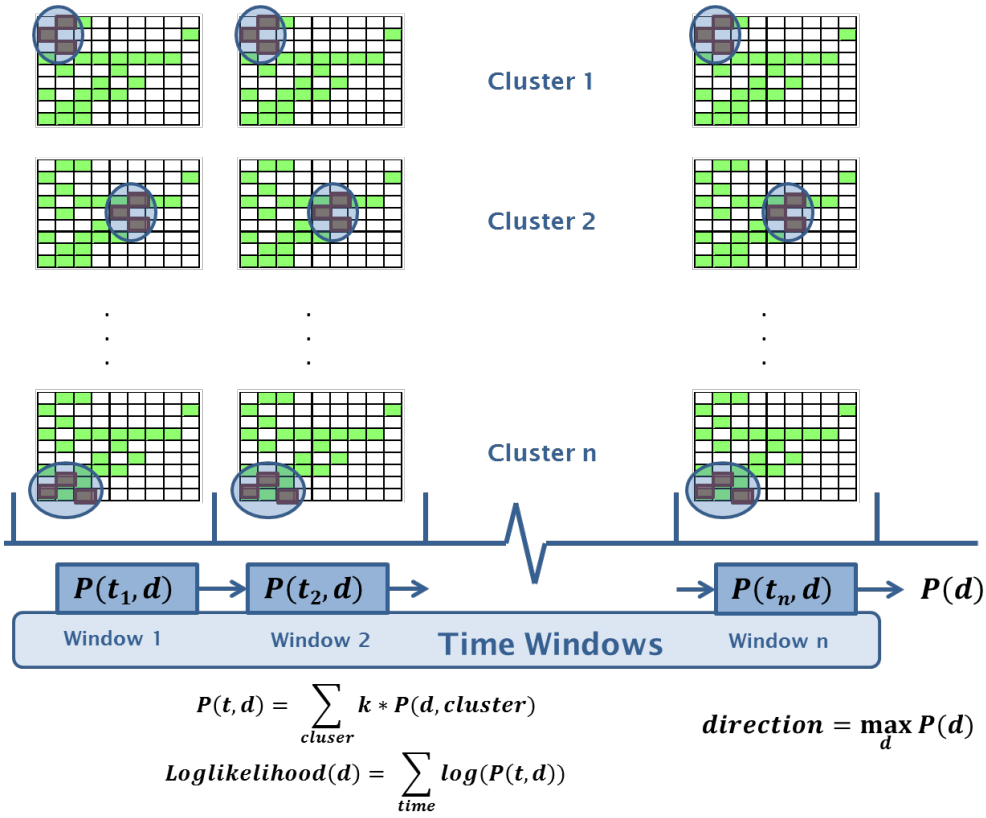


Figure 4.7: A schematic of the clustering methodology to identify locations with high probability of top ranked channels. The circle represents the channel selection operator using the topographical clustering method.

4.4 Topographical Clustering of Qualitative Patterns

The previous analysis establishes that certain spatial locations typically have relatively high power when reaching a particular direction. To understand the spatial configuration of high ranked channels, I analyzed their spatial behavior and hypothesized that the location of high ranked channels predicts movement direction [193]. To analyze this hypothesis, each temporal snapshot

of the trial is characterized by the probability of observing a high-powered spatial location. I estimate the posterior probability of spatial locations having a top ranked channel for a given direction. I tested if the spatial closeness of electrodes provides robust decoding and if the probability could be extended to spatial grouping of electrodes rather than a single electrode. I proposed two methods to cluster electrodes based on their topography. One method considered comparing a cluster of neighboring channels for their direction sensitivity. The other method considered clusters formed in a greedy fashion. For both these methods, I used a maximum likelihood prediction algorithm to estimate the directionality of a trial assuming its independence in temporal patterns. The training and testing phases are described in Algorithm 3 and a schematic is presented in figure 4.7.

The two monkeys were implanted with two electrode arrays in the premotor and primary motor areas. I analyzed spatial configuration of top-ranked channels in these two grids independently. For the initial analysis, I evaluated all possible combinations of channels by selecting a 2-electrode and a 3-electrode cluster neighborhood, modeling each direction with a probability distribution of top ranked channels. The distribution of top 10 channels in each grid is considered to determine the decoding. I observed that each electrode grid provided directional information and characterizing such spatial information would be critical [191]. I found that combining the spatial information from both the grids provided the best decoding ability. Figure 4.8 shows the spatial location of top ranked channels for different direction reaches. I that to

Algorithm 3 Algorithm to identify Clusters of Top Ranked Channels

Training Stage: Learn $\mathcal{M}$ from Data:$\{\mathbf{X}_i(t), \mathbf{D}_i\}_{i=1}^{N}$
Calculate Channel Power: $\mathbf{P}_i = \sum_{t-\tau}^{t} \mathbf{X}(t)$
Rank the Channels using channel power: $\mathbf{R}_i = \text{sort}(\mathbf{P}(t))$
Identify top ranked channels: $\mathbf{T}(t) = \begin{cases} 1, & \text{if } \mathbf{R}(t) < T \\ 0, & \text{otherwise} \end{cases}$
for each column time instant t **do**
  for each cluster k **do**
Enumerate $P(\text{cluster } k \text{ has } c \text{ Top Ranks}/\mathbf{D} = d)$:

$$P_t^d(k, c) = \frac{\sum_{\mathbf{D}(i)=d} T_i(t)}{|\mathbf{D}(i) = d|}$$

  end for
end for
$\mathcal{M} \leftarrow \{P, T\}$

Testing Stage: Estimate $\hat{\mathbf{D}}_i^*$ from Data:$\{\mathbf{X}_i^*\}_{i=1}^{N}$, $\mathcal{M}^*$
Calculate Channel Power: $\mathbf{P}_i^* = \sum_{t-\tau}^{t} \mathbf{X}^*(t)$
Rank the Channels using channel power: $\mathbf{R}_i^* = \text{sort}(\mathbf{P}^*(t))$
Identify top ranked channels: $\mathbf{T}^*(t) = \begin{cases} 1, & \text{if } \mathbf{R}(t)^* < T \\ 0, & \text{otherwise} \end{cases}$
for each cluster k **do**

$c_k^* = \sum T^*(t)$

end for
for each possible target d **do**

$p_i(D* = d/\text{cluster } k \text{ has } c \text{ Top Ranks}) = \prod_{t,k} P_t^d(k, c_k^*)$

end for
Estimate target direction: $\hat{\mathbf{D}}_i^* = \max p_i$

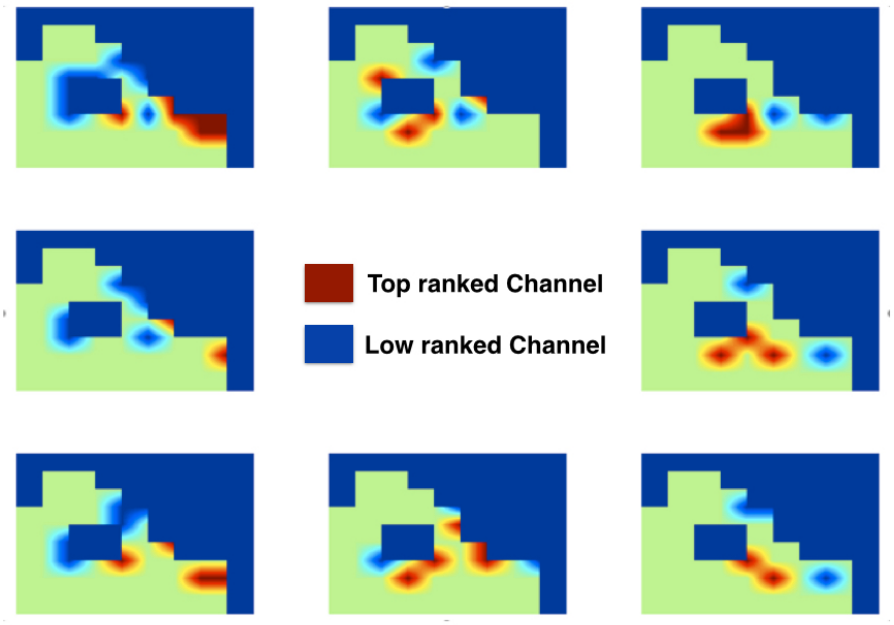


Figure 4.8: Spatial Location of Discriminative Patterns in the Primary Motor Area of monkey H464. The eight grids are placed at the location of their respective targets. Each grid represents the $4mm \times 4mm$ grid area of the primary motor area of the brain. The colors represent the

obtain consistent results, at least the top 10 channels from each grid need to be analyzed because analysis of fewer channels resulted in poorer performance. This implies that high spatial variation exists even in top ranked channels and including more channels reduces the variation.

I extended this analysis to the use of a nonlinear classifier. I built a Support Vector Machine (SVM) classifier on this feature set because they are

Table 4.2: Decoding Accuracy of the Support Vector Machine model trained only on the locations of the top ranked channels. The proposed Rank CSP and the Time Evolution models performed significantly better ($p < 0.01$) than the CSP method.

Day after Training	8	9	13	14
H464				
Number of Trials	263	325	348	88
CSP	38%	41%	16%	13%
Rank CSP	62%	61%	55%	47%
Time Evolving Spatial Pattern	71%	69%	58%	62%
Using Location of Top Ranks	76%	81%	70%	70%
H564				
Number of Trials	206	103		
CSP	41%	40%		
Rank CSP	53%	52%		
Time Evolving Spatial Pattern	66%	60%		
Using Location of Top Ranks	61%	49%		

known to provide better classification than simple linear classifiers, especially on a high dimensional feature set [21]. The basic idea in using SVM is to project the feature set onto a higher dimensional vector space using a kernel and classifying this projected feature space using a hyperplane. I chose Radial-Basis kernel to project features and extract the non-linear trend. Since these classifiers are mainly used for binary classification, I have incorporated an Error Correction Output Code (ECOC) to tackle our multiple class scenarios similar to previous analysis.

A comparison of results for various algorithms is provided in table 4.2. I compare our results with those from CSP and ECOC, which uses the raw data

to construct CSP; and Rank CSP and ECOC, which use the rank patterns, time-evolving spatial patterns, and top ranked channels. As I can see, this method provides better decoding across the two weeks of testing in H464 ($p < 0.01$ calculated using McNemar's Test). Further, the error remains consistent as shown in the figure 4.9. On an average, only 20% of trials were more than 45°away from the original target. I observed that the location of the top rank carried most information regarding directionality. While it requires a non-linear classifier like SVM, the method provides consistent decoding over the evaluated sessions. In monkey H564, however, analyzing just the top ranked channels resulted in decreased performance. This shows that for H564, the top rank location is insufficient to obtain robustness over multiple days. In this monkey, analysis of all the channels might provide a better decoding performance.

4.5 Conclusion

This chapter discussed the use of robust feature extraction techniques for long-term BCI. Specifically, I evaluated the use of qualitative spatial patterns based on relative inter-channel powers and the analysis provided following conclusions:

1. Using ranking method improves the decoding performance ($p < 0.01$) of a static decoder from 36% to 56%, on an average, over two weeks of recording.

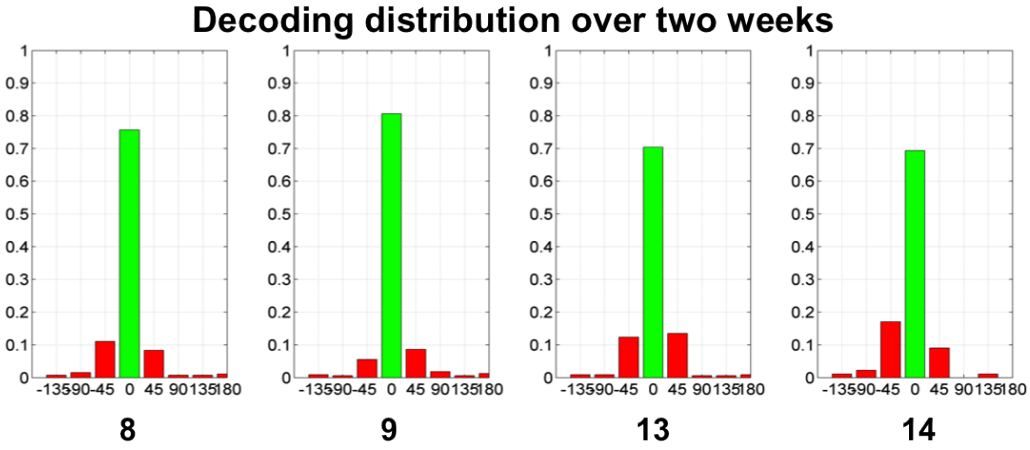


Figure 4.9: Error Distribution using the Spatio Temporal Pattern Algorithm on multiple testing sessions indicated by the recording day.

2. This method also provides a consistent decoder when there are changes in the external field forces, establishing their robustness to environmental variances.

3. Ranking the raw signals provides a robust way to capture behavior specific invariant spatial patterns.

4. Since ranking ignores the details of the raw signal, I can conclude that the spatial patterns over multiple days have similar structures. While changes in the actual values might vary during different sessions, the relative power of the channels remains consistent.

5. Evolution of the rank patterns provides a better model and robust performance ($p < 0.01$) of the direction decoder.

6. Most of the directionality information could be inferred from top ranked channels.

7. Similar to the evolution of spatial patterns, evolution of the top-ranked channels provides robust performance.

Although these features show promise of good performance, the decoder still carries disadvantages of the CSP method and fails to provide consistent decoding over a long period. This requires advanced classification techniques either linear or non-linear that provide robust decoding.

Chapter 5

Movement Direction Decoding

5.1 Introduction

Traditional feature extraction techniques like Common Spatial Patterns (CSP) assume that the neural behavior is captured by a single spatial pattern. However, these techniques fail to capture the variability particularly over chronological recordings. I hypothesized that human behavior is inherently variable and multiple neural patterns describe this variability better than a single pattern. This chapter presents an algorithm, based on the above hypothesis, that overcomes signal variability over time and due to different environmental conditions. I show that such a strategy improved decoding capability and accurately predicts 90% of hand reaches to eight directions over 4-6 weeks.

5.2 Background

The previous chapter discussed the use of qualitative patterns in the form of instantaneous inter-channel ranks. These features were tested with comparable algorithms like Common Spatial Patterns (CSPs) to establish their superior performance. The results show evidence that using non-linear classi-

fiers provide better decoding. The assumption behind CSP is that two classes could each be defined by a single gaussian distribution of their inter-channel covariance. The two classes are assumed to have the same feature variance. However, these variances are wide enough to cover all the training data. Based on this assumption, CSP proceeds to find a mean spatial pattern that could sufficiently describe a behavior. Since the mean spatial pattern is learned from the training data set, it provides excellent discrimination for the training set and in studies where characteristics of the data do not change from training to test. While it proves as a useful analysis tool, it overfits the training data and generalizes poorly over the entire range of testing data [160]. This is a major drawback of using CSP on testing data that differs from training data. Many versions of regularization of CSP have been proposed to add prior knowledge of data to the CSP [115, 70, 217]. However, it is not clear when any of the proposed regularization provides better classification.

Human behavior, including movement, is inherently variable and non-repetitive [188]. Each repetitive movement of the arm involves a unique set of motor patterns. As multiple ways of performing a same task exist, I hypothesize that arm position is encoded effectively by multiple neural patterns that generate multiple motor patterns. In a given set of neural data and corresponding behavioral tasks, I intend to identify prototypical neural patterns. This chapter discusses the use of such templates and their use in decoding arm direction.

Consider $x(t) \in \mathcal{R}^C$ and $y(t) \in \mathcal{R}^C$ be the multi-channel recordings from

C channels at each time t and $\mathbf{X}_i \in \mathcal{R}^{C\times T}$ and $\mathbf{Y}_i \in \mathcal{R}^{C\times T}$ be their corresponding spatio-temporal patterns. The decision on a new spatio-temporal pattern $\mathbf{Z}$ for a recording $z(t)$ is obtained using multiple spatial patterns modeled using the following equation (5.1)

$$\sum_{i\in x} w_i \Phi(\mathbf{Z}, \mathbf{X}_i^*) \gtrless \sum_{i\in y} w_i \Phi(\mathbf{Z}, \mathbf{Y}_i^*) \tag{5.1}$$

where $\Phi()$ represents a suitable kernel, either linear or non-linear, to project the spatial patterns on decision space. The weights w_i are optimally estimated to provide the least training error. Using a maximum likelihood estimate to obtain estimates of w provides a good fit on training data but suffers from poor generalization [200]. Constraints on the nature of w ensure a better generalization. For model optimality all parameters of the model w, $\mathbf{X}_i^*$, and $\mathbf{Y}_i^*$ should be estimated [104]. That is a tedious task and therefore in this proposed model spatial patterns from the training set were used to build the model. Using a single spatial pattern similar to the sample average spatial patterns results in a CSP formulation.

The proposed model decodes intended target direction by estimating hand position in terms of its horizontal and vertical components from LFP data. Regression is well suited for such analysis since it provides continuous control and generalizes to novel targets and environments [124]. This model uses a kernel based regression method called Relevance Vector Machine (RVM) that approximates the target by measuring similarity between acquired neural features and a feature basis. RVMs are trained in a Bayesian framework and

provide a generalized sparse solution on training data [200]. They work on the premise that a few relevant vectors describe training feature space accurately and can generalize beyond the training set. However, these methods need the input space to be relatively stationary and fail to perform when trained relevant vectors partially (or cannot) describe the new features [136]. This situation arises in BCI where feature non-stationarity is typical. While retraining the entire model to suit the necessities of a new feature set is an option, this frustrates the user and diminishes usability of BCI. This model offers a novel solution that updates the model online in an unsupervised learning framework without needing to collect calibration data. This model update is a step closer to practical BCI and provides robust long-term performance even during different behavioral environments.

To decode the intended path, proposed decoder estimates multiple hand kinetic parameters at each instant. Most regression techniques, including RVM, assume a single target vector and prescribe a separate regression to estimate each dimension. Such estimates are spurious and fail to provide good estimation in all dimensions simultaneously, especially when these dimensions are correlated [10]. Further, good estimates of one dimension do not translate to other dimensions. To avoid this I propose the use of a Kernel Dependency Estimation (KDE) framework. This framework employs kernel functions to measure the correlations in target dimensions and encodes prior information about the target in an elegant way [209]. The kernel can be decomposed into its singular vectors, resulting in multiple redundant regression models to

estimate the multi-dimensional target vector.

The proposed model decodes intended arm target direction by tracking the hand position continuously. In BCI, faithful decoding of hand movement requires the model to estimate multiple kinetic parameters - horizontal (x-) and vertical (y-) components. The main innovations of the proposed solution are:

1. Use of abstraction levels to estimate multiple kinematic parameters (5.4).
2. Semi-supervised adaptation to address short-term non-stationarities (5.5).
3. Unsupervised model-pruning to improve computational complexity while maintaining robustness against long-term variability (6.2).

I estimate multiple parameters simultaneously using multiple kernel regression in a Kernel Dependency Estimation (KDE) framework. Hence, I choose a RVM regression framework as it provides superior generalization through sparse formulation. The next sections discuss these methods.

5.3 Relevance Vector Machines

RVM is a set of general models in the form of equation (5.2)

$$y(\mathbf{X}) = \sum w_i \Phi(\mathbf{X}, \mathbf{Y}_i) + w_0 \tag{5.2}$$

where $\mathbf{Y}_i$ are the different feature basis vectors and $\Phi()$ is the kernel function that measures similarity between input neural feature vector $\mathbf{X}$ and basis vectors. The task in RVM is to estimate $\mathbf{w}$ when training input and target data

are provided. Maximum likelihood estimation of $\mathbf{w}$ without any constraints leads to over-fitting on the training set [200]. The RVM framework obtains generalization via sparse formulation under the assumption that $\mathbf{w}$ is derived from a zero mean Gaussian distribution. The search for 'relevant' vectors leads to the best subset of input feature vectors that can represent a given input space. RVM introduces a new set of hyper-parameters α to set a Gaussian prior of the form:

$$p(\mathbf{w}/\alpha) = \prod \mathcal{N}(w_i/0, \alpha_i^{-1}) \tag{5.3}$$

$\mathbf{w}$ is estimated in an iterative fashion to optimize the marginal likelihood over α [199]. Usually, the chosen basis vectors $\mathbf{Y}_i$ are a set of prototypical examples from the input training vector set. The relevant vectors (support vectors) chosen by RVM are significantly different from SVM. SVM gathers vectors closer to the decision boundary and causes the number of support vectors to grow linearly with the training set [200]. In contrast, RVM gathers prototypical examples from training data that lie at the center of it and retains sparsity even in a growing data set.

I chose a Gaussian radial basis function, shown in (5.4), since an RBF-SVM technique provided successful decoding in a similar setting [192]:

$$\Phi(\mathbf{X}, \mathbf{Y}) = \exp(-\frac{||\mathbf{X} - \mathbf{Y}||_2^2}{\sigma^2}) \tag{5.4}$$

where σ is the basis width that determines the spread of each basis vector $\mathbf{Y}$. A smaller value of σ indicates that the vector has a smaller coverage of the feature space and by allowing a larger σ a single vector can cover a larger

feature space. Thus, a kernel with a larger σ will provide a sparser solution than a kernel with a smaller σ. In their original form all the basis vectors have the same spread; however, variations of kernels can be formulated with multiple basis width [51].

5.4 Multi-output Regression

The RVM framework is formulated to provide a sparse solution for a single target. This model extends it to estimate multiple target components. One approach to estimate a multi-dimensional target is to use an independent model for each dimension. Such models provide a good correlation of individual dimensions when independent and provide spurious results when dependence exists [10]. I observed that the estimated absolute position correlated well with the actual value (>0.92) than the independent estimates of x- and y-positions (0.82). I would want to leverage this high correlation in one parameter to others and achieve better estimates of all parameters. The technique proposed here is based on Kernel Dependence Estimation (KDE) to take advantage of such dependence and obtain a better overall performance [209]. KDE uses kernel technique to encapsulate the multi-dimensional output in a similar fashion as the input kernel.

A suitable target kernel function reflects the non-linear dependence of target dimensions and chosen target basis vectors. Each target basis vector of this function represents a unique point in the hand movement space as shown

by (5.5) below:

$$\mathbf{\Psi}(\mathbf{y}, \mathbf{y}_i) = \exp(-(\mathbf{y} - \mathbf{y}_i)\Sigma_y^{-1}(\mathbf{y} - \mathbf{y}_i)^T) \tag{5.5}$$

The above equation is a gaussian kernel evaluated at each point on the target space, where $\mathbf{y}_i$ denotes hand movement space in the form of its horizontal, vertical, and absolute positions: $\{y_x, y_y, y_r\}$, and $y_r = \sqrt{(y_x^2 + y_y^2)}$. This kernel assesses the similarity between actual hand position and any location on the grid as a function of their euclidean distance. Building this target kernel basis allows decomposing the obtained basis into its independent singular vectors, and approximating them individually in an RVM framework. By formulating a regression model for the left singular vectors of $\mathbf{\Psi}$, an equivalent regression model is obtained as follows:

$$\mathbf{\Psi} = USV^T \tag{5.6}$$

$$U_b = \sum w_{bi}\mathbf{\Phi}(X) + w_{b0}$$

where U_b is the b^{th} column of U and $(.)^T$ is the transpose operator. This results in multiple redundant approximations and provides high correlation in all the target dimensions. Choosing only top K singular vectors that represent more than 90% of the basis energy reduces the number of regression models. During the training phase, I learn regression parameters for each component. In the testing phase neural features are projected on the input basis $\mathbf{\Phi}$ and each abstract component of the output basis is estimated using independent regression models. Output basis, $\hat{\Psi}$, is estimated by combining all the calculated

output bases. The model can be described by the kernel function parameters, basis vectors and the corresponding $\mathbf{w}$ as $\mathcal{M} := \{\mathbf{\Phi}(.), \mathbf{X}, \mathbf{w}\}$.

To obtain the estimates for new test samples $\mathbf{X}_{new}$, model $\mathcal{M}$ can be applied as

$$\hat{y}_{new} = \sum \mathbf{w}_b^T \mathbf{\Phi}(\mathbf{X}_{new}, \mathbf{X}_b) + \mathbf{w}_{b0} \tag{5.7}$$

One of the challenges for KDE during testing stage is pre-image identification that refers to identifying the correct basis when value of the target kernel is given [209]. In the current setting, this refers to identifying hand position from the estimates of the output kernel. Generally, finding the pre-image of a Gaussian kernel is difficult and pre-images might not necessarily exist [173]. I chose a maximum likelihood approach by identifying the target basis that provides the maximum kernel value.

5.5 Adaptation

The above-described model provides good approximation and generalization, if the feature basis vectors can represent the entire input feature space. However, when there are changes in the feature characteristics in noisy and non-stationary time-series environments, RVM fails to capture the dynamics and provides unstable approximations [136]. It has been well documented that neural recordings and their features change due to learning and other environmental conditions. In a typical BCI, the new neural feature vectors acquired on a different day (or session) tend to align poorly with previously collected

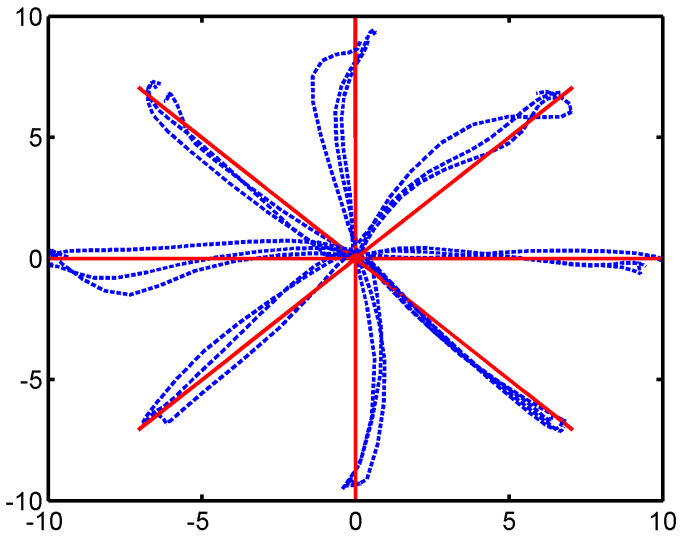


Figure 5.1: Actual ($\mathbf{t}$, dashed blue), and the used intended movement ($\tilde{\mathbf{t}}$, solid red) of example trials overlaid on a 10cm $\times$ 10cm workspace.

feature basis vectors $\mathbf{X}_b$. Hence, updates are required to adapt the model to the new data.

If the target $y_{desired}$ corresponding to $\mathbf{X}_{new}$ was available, adaptation only requires a correction that provides a good estimate for the residual trajectory:

$$e = y_{desired} - \hat{y}_{desired}$$

$$\mathbf{w}_u : \min ||e - \mathbf{w}_u^T \mathbf{\Phi}(\mathbf{X}_{new}, \mathbf{X}_{new})|| + \lambda ||\mathbf{\Phi}(\mathbf{X}_b, \mathbf{X}_{new})|| \tag{5.8}$$

$$\mathcal{M}_u := \{\mathbf{\Phi}(.), \mathbf{X_{new}}, \mathbf{w}_u\}$$

The first part of equation (5.8) can be estimated using RVM learning algorithm with similar constraints shown above in section 5.3. The constraint on the

included basis ensures that the baseline model remains unaffected with the update ($\mathbf{w}_b$ needs no update). In the absence of such a constraint, the fit on the training data would suffer due to the addition of new feature bases.

While it is desirable for BCI to build the model update on actual hand kinetics, they are unavailable in practice. Our key innovation is to mimic a practical BCI even when no prior knowledge of the actual hand trajectories $y_{desired}$ exists. For this, I incorporate general principles of natural movements by assuming that the monkey reaches intended target direction in a straight line from the center to the target. I propose to use estimated target direction from baseline model to obtain the intended path. Thus, I construct the intended linear trajectory from the center to the target as $\tilde{y} = \mathcal{F}(\hat{\theta})$ shown in figure 5.1.

The overall model used for succeeding trials will be $\mathcal{M}_* = \mathcal{M}_0 \oplus \mathcal{M}_u$, where $\oplus$ is a suitable appending function and $\mathcal{M}_0 = \{\mathbf{\Phi}(.), \mathbf{X}_0, \mathbf{w}_0\}$ is model before this update. Since the current model structure is linear in the kernel function space, updated model is obtained as follows:

$$\mathcal{M}_* = \{\mathbf{w}_0 \cup \mathbf{w}_u, \mathbf{\Phi}(.), \mathbf{X}_0 \cup \mathbf{X}_u\} \tag{5.9}$$

$$\hat{y}_*(\mathbf{X}_*) = \sum \mathbf{w}_0^T \mathbf{\Phi}(\mathbf{X}_*, \mathbf{X}_0) + \sum \mathbf{w}_u^T \mathbf{\Phi}(\mathbf{X}_*, \mathbf{X}_u) \tag{5.10}$$

Since the model update involves only a few sample correction, it takes less time for the update and can be performed online as soon as a successful trial has finished. The algorithm for the model training, testing, and adaptation is presented in Algorithm 4.

Algorithm 4 Algorithm to model multiple spatial patterns to decode movement direction

Training Stage: Learn $\mathcal{M}$ from Data:$\{\mathbf{X}_i, \mathbf{t}_i\}_{i=1}^{N}$
Build Input Kernel : $\mathbf{\Phi}(\mathbf{X_i}, \mathbf{X_i})$ using eq ((5.4))
Build Output Kernel : $\mathbf{\Psi}(\mathbf{t_i}, \mathbf{y_j})$ using eq ((5.5))
$\mathbf{\Psi} = USV^T$
for each column k of U **do**

Estimate $\mathbf{w}_k : U_k = \sum_{i=1}^{N} w_{ki}\Phi(X) + w_{k0}$

end for
Basis Vectors : $\mathbf{D} \leftarrow \mathbf{X}$
Store Model: $\mathcal{M}_0 \leftarrow \{S, V, \mathbf{D}, \mathbf{w}\}$

Initialize $\mathcal{M}^* \leftarrow \mathcal{M}_0$
Testing Stage: Estimate $\hat{\mathbf{t}}_i$ from Data:$\{\mathbf{X}_i^*\}_{i=1}^{N}$, $\mathcal{M}^*$
Build Input Kernel : $\mathbf{\Phi}(\mathbf{X_i^*}, \mathbf{D_j})$ using eq ((5.4))
for each column k of U **do**

Calculate $\hat{U_k} = \sum_{j=1}^{K} w_{kj}\Phi_j(X) + w_{k0}$

end for
$\hat{\mathbf{\Psi}} = \hat{U}SV^T$
$\hat{\mathbf{t}}_i = \max_i \hat{\mathbf{\Psi}}_i$, $\hat{\theta}_i = \arctan \frac{t_y}{t_x}$

Adaptation Stage: Update Model $\mathcal{M}^*$ after L trials using $\{\mathbf{X}_i^*, \hat{\theta}_i^*\}_{i=1}^{L}$
$\tilde{\mathbf{t}} = \mathcal{F}(\hat{\theta}_i)$
$\tilde{\Psi} = \mathbf{\Psi}(\hat{\mathbf{t}}_\mathbf{i}, \mathbf{y_j})$ using eq ((5.5))
$\tilde{U} = \tilde{\mathbf{\Psi}}VS^{-1}$
for each column k of U **do**

Update $\mathbf{w}_k^u : \tilde{U_k} - \hat{U_k} = \sum_{i=1}^{N} w_{ki}^u \mathbf{\Phi}(\mathbf{X}_i^*) + w_{k0}^u$ using eq ((5.8))

end for
Basis Vectors : $\mathbf{D}^u \leftarrow \mathbf{X}^*$
$\mathcal{M}^* \leftarrow \{S, V, \mathbf{D}|\mathbf{D}^u, \mathbf{w}|\mathbf{w}^u\}$

5.6 Results

The regression model learned from data collected on a single day and this static decoder is applied on future sessions. I observed that the basis-width plays a significant role as it modulates the error in the prediction. Although a narrow target basis width implies less variable output estimation, representation of such a kernel requires considerably high number of singular vectors. On the other hand, a wide target basis width allows the decomposition of the output kernel $\mathbf{\Psi}$ with lesser number of components. In the current analysis, choice of a wide basis width obtains a computationally efficient algorithm. A narrow width kernel boosts decoding by sacrificing computational complexity. Table 5.1 presents the results obtained using the suggested model without adaptation. To obtain decoding accuracy, angle made by the trajectory is measured as $\Theta = \tan^{-1}(y/x)$. The target closest to the decoded angle is assigned to the trial.

Next, I present the performance of a continuously adapting model. The model is adapted to account for varying signal characteristics after decoding 25 trials and selecting only successfully predicted trials. Updating the model after each successful trial, results in an oversensitive update. While such an update presents a new model at every trial, it also requires an additional processing time (to update the model) at the end of every trial. Conversely, update after a large number of trials - passive model - might not track the fluctuations fast enough and will be ineffective in the short term. Thus, the update process must choose an optimal number of trials to update the model. In the current

Table 5.1: Correlation coefficients of hand position predictions and the actual hand positions measured while the monkey performed the tasks. These correlations represent the average value over multiple trials conducted during a particular session (as indicated in the Table).

Decoder Age	# of trials	ρ_x	ρ_y	Decoding Accuracy
H464				
8	263	0.94	0.91	93%
9	325	0.93	0.91	89%
13	348	0.88	0.87	82%
14	88	0.76	0.75	66%
H564				
8	206	0.89	0.85	80%
9	103	0.79	0.75	66%

analysis, decoding performance varied little (<2%) with different number of adaptation trials. The model stores only accurately predicted trials for the next update. Thus, updated performance depends on the quality of the initial model. If the initial model decodes poorly, there will be lesser number of trials for update and this results in an inefficient update. Figure 5.2 shows stable performance of the adaptation algorithm over multiple blocks and over days. The vertical axis shows decoding accuracy measured as the fraction of all correct predictions up to the current instant over successive adaptation blocks (25 trials). The adaptation algorithm maintains the decoding consistently across multiple blocks and multiple days, while the performance baseline algorithm slowly decays over the adapting blocks.

The update process involves obtaining new basis vectors that fit the er-

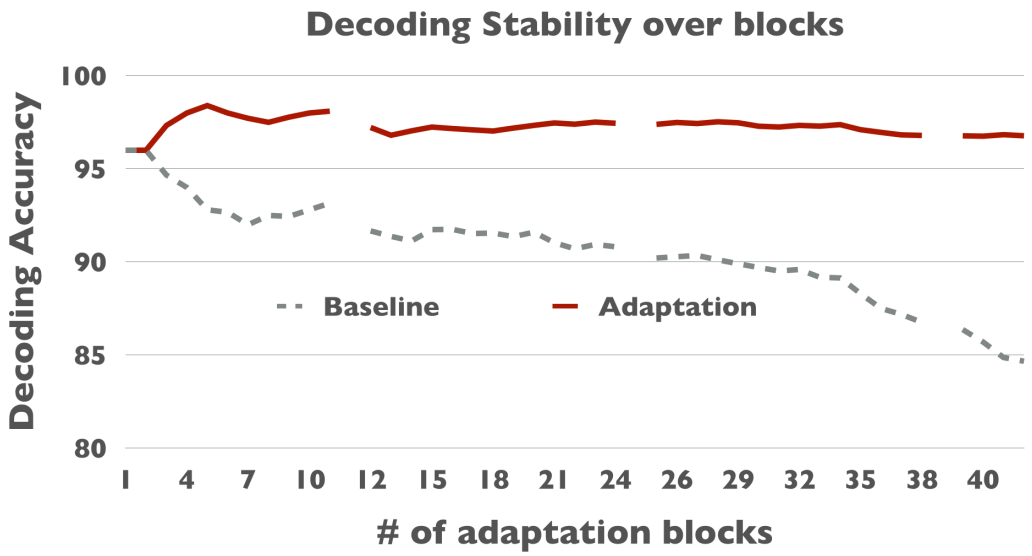


Figure 5.2: Cumulative decoding accuracy with and without adaptation across multiple adaptation blocks of 25 trials each. The gaps in the curves represent the end of day.

rors from original model, increasing the number of basis vectors in the updated model. Here, I present a model pruning strategy that eliminates redundant feature basis. This process includes all estimates despite their proximity to the intended trajectory. However, I can improve the computational performance (reduce number of basis vectors) by selecting only those trajectories, where estimated value ($\hat{t}$) deviates more than a threshold from the expected trajectory ($\tilde{t}$). By allowing a deviation of 1 cm between the expected and intended trajectories, I observed that the number of basis vectors drastically reduces without affecting the decoding accuracy.

Table 5.2 compares the results of adaptation to the decoding perfor-

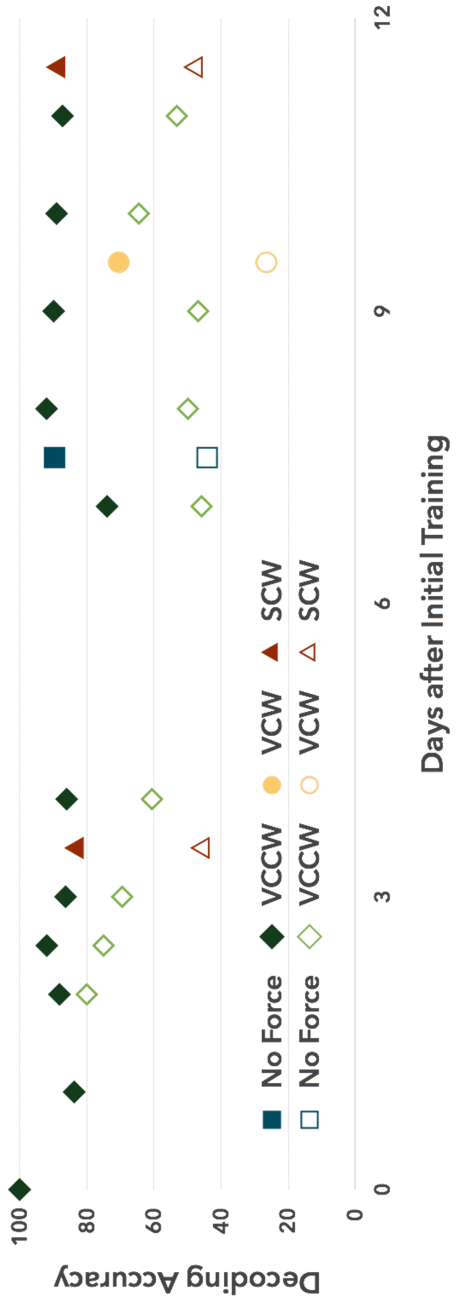


Figure 5.3: Decoding Accuracy in sessions with varying field forces. The filled icons represent accuracy with adapting model and unfilled icons represent the accuracy of the baseline model. For ease of reading, different field forces are represented with different shapes.

Table 5.2: Decoding Accuracy (in %) across decoder age. We observe that the accuracy of the static decoder falls to 66% over two weeks, while an evolving decoder performs significantly better ($p < 0.01$) at 97% over the same period.

Decoder Age	8	9	13	14
H464				
Number of Trials	263	325	348	88
No update	93	89	82	66
Daily model Reset	98	96	92	82
Continuous Update	98	97	96	97
H564				
Number of Trials	206	103		
No Update	80	66		
Daily model Reset	81	70		
Continuous Update	81	80		

mance over the decoder age. To observe long-term effects of adaptation, I adapted one model continuously over the two weeks of test data. Another model adapted only to current test session and ignored any previous adaptation by resetting the model to $\mathcal{M}_0$ at the beginning of each day. I observed that adaptation improves target decoding accuracy over the two weeks ($p < 0.01$ using McNemar's Test). When the model was not adapted, accuracy drastically reduced around day 14 but adaptation results in an accuracy over 95%. While model adaptation only on the current day improves decoding accuracy, its performance gradually decreases with the decoder age due to evolution of new neural patterns. These results show that learning modulates neural activity continuously (rather than daily) and decoders benefit from the adaptation to variations introduced by this learning.

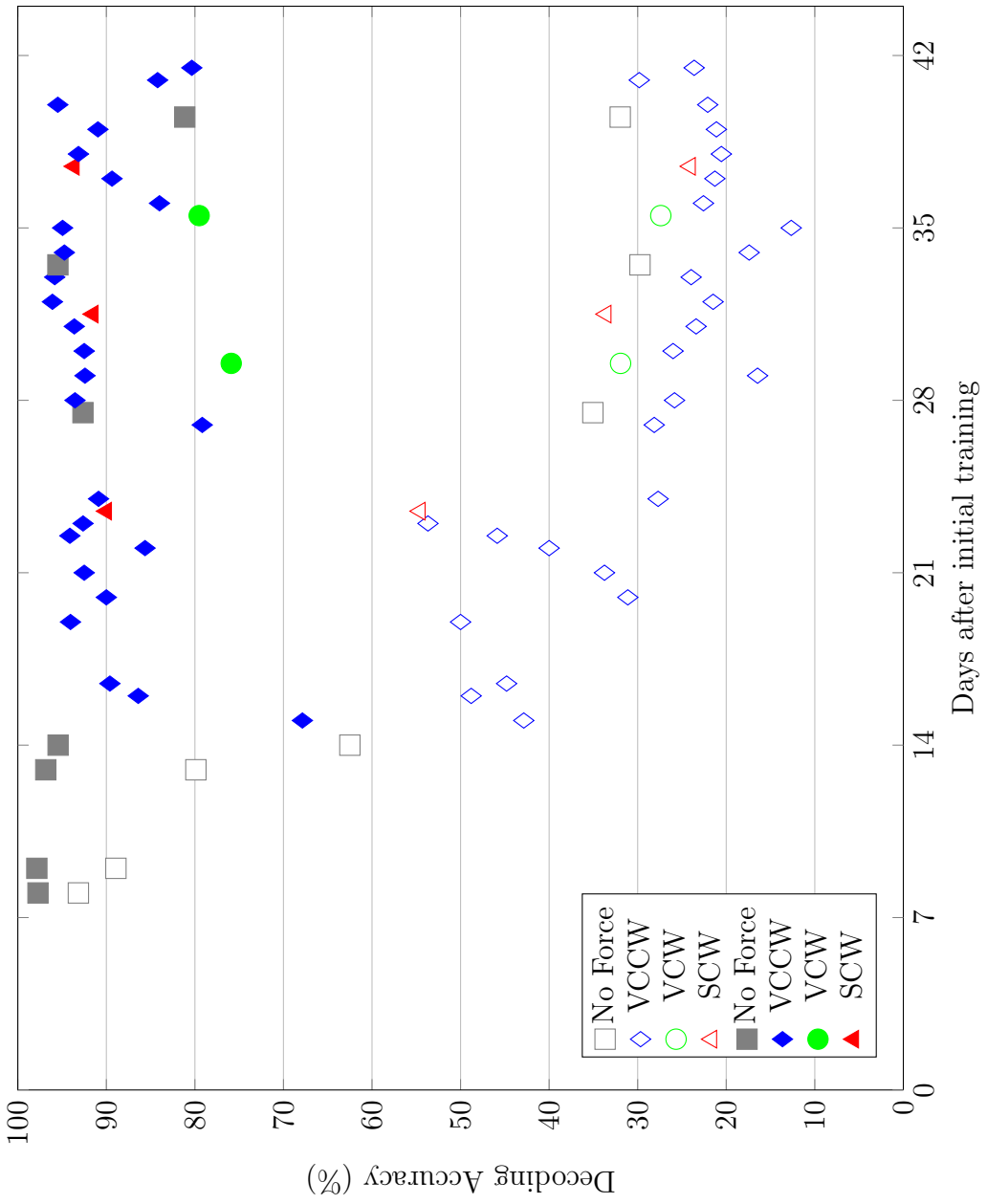


Figure 5.4: Decoding Accuracy in all recorded sessions from monkey H464. The filled icons represent accuracy of adapting model and the unfilled icons represent the accuracy of the baseline model. For ease of reading, different field forces have been represented with different shapes. The adaptive decoder performs significantly better ($p < 0.01$) than the static decoder over the entire evaluation period.

Table 5.3: Comparison of Decoding algorithms for H464. The proposed method performs better than the state-of-the-art CSP algorithm. Further, even when the CSP method is adapted to the changes in the neural pattern, the adaptive method presented here performs stably over 95%.

Decoder Age	8	9	13	14
CSP	50%	46%	22%	14%
Rank CSP	64%	56%	43%	26%
Proposed Method	93%	89%	80%	66%
Adaptive CSP	50%	70%	26%	16%
Adaptive Rank CSP	78%	85%	76%	69%
Proposed Method with Adaptation	98%	98%	97%	97%

To investigate the effect of adaptation on changing environments, I applied a similar strategy on sessions where external field forces against movement were applied. Even for this experiment, I trained the model on a session where a field force VCCW was applied and updated on successive sessions. The model here needs to tackle both variations due to learning over time and due to varying external field forces. These results presented in figure 5.3 show that the model is robust to both and achieves an average decoding of 85% on sessions even with different external field forces. I observe interesting results when a) latency is present between sessions (between days 4 and 7), and b) new field force VCW is introduced (on day 12). In these sessions, adaptation boosts decoding (especially on day 12). These results motivate the use of LFP for practical BCI even under varying environment conditions.

In a final simulation, I trained a model on day 1 of all recordings and allowed it to learn variations of the neural data over time and external field

forces without interruption. Figures 5.4 and 5.5 show the decoding performance of such an adaptation over all the sessions in comparison to the same algorithm without adaptation. I observed that decoding performance reduced when there is a delay between two successive sessions or when a novel field force is introduced to the subject. I observe this during the first VCCW session in both monkeys. This drop in performance is anticipated because field forces were introduced to the monkeys for the first time. In fact, the monkeys were unable to cope with this abrupt introduction of force applied to the manipulandum and needed two sessions to adjust to the change. In these early sessions, the monkeys were still adjusting to behavioral change. I can infer that once they became accustomed to these forces, they generated stable neural patterns and hence the decoding accuracy stabilized.

Table 5.3 compares the proposed method with state-of-the-art Common Spatial Patterns (CSPs) method and its variant [20, 192]. To provide a fair comparison, decoding model uses multiple redundant linear classifiers and the final output obtained through an Error Correcting Output Code (ECOC) [41]. I also applied a similar adaptation strategy to CSP decoders and adapted them after every 25 trials. CSP fails to incorporate the signal variability over time and thus reports lower performance, while CSP applied on rank features provides a stable performance by virtue of using stable features. However, the proposed method outperforms both the versions of CSP and provides robust performance over time and external forces ($p < 0.01$). Table 5.3 also shows the performance of algorithms when they are adapted using techniques de-

Table 5.4: Decoding Power (DP) of other literature in comparison with the proposed method. Note that while the other studies used cross-validation for their analysis, the proposed algorithm provides higher accuracy than the existing algorithms. (These algorithms need to be evaluated on the same data to provide comparative significance).

Algorithm	Decoding Accuracy
Bayesian Decoding, SVM [128]	40%
Directional Tuning [138]	50%
Bayesian Classification [171]	81%
Directional Tuning [9]	75%
Proposed Method	95%

scribed in 5.5. I see that even when the traditional decoding algorithms are adapted to new data they fail to capture novel patterns. These results show that robust performance needs a robust baseline model for adaptation. The algorithm presented here decodes the intended movement successfully without adaptation and its performance is enhanced with adaptation.

Table 5.4 compares the decoding power of proposed method with those of other studies using LFP. A direct comparison between these studies is difficult as they use different modalities, behavior and learning paradigm. In addition, these studies used cross-validation to obtain the decoding power and ignore any non-stationarity between training and testing samples. Recent publications show long-term decoding in an online setting, where the user adjusts to the decoder and stabilizes neural patterns, to perform tasks consistently with a target reach accuracy under 80% [58]. Results from this model suggest that a similar adaptive decoder may perform well in a closed-loop setting, as

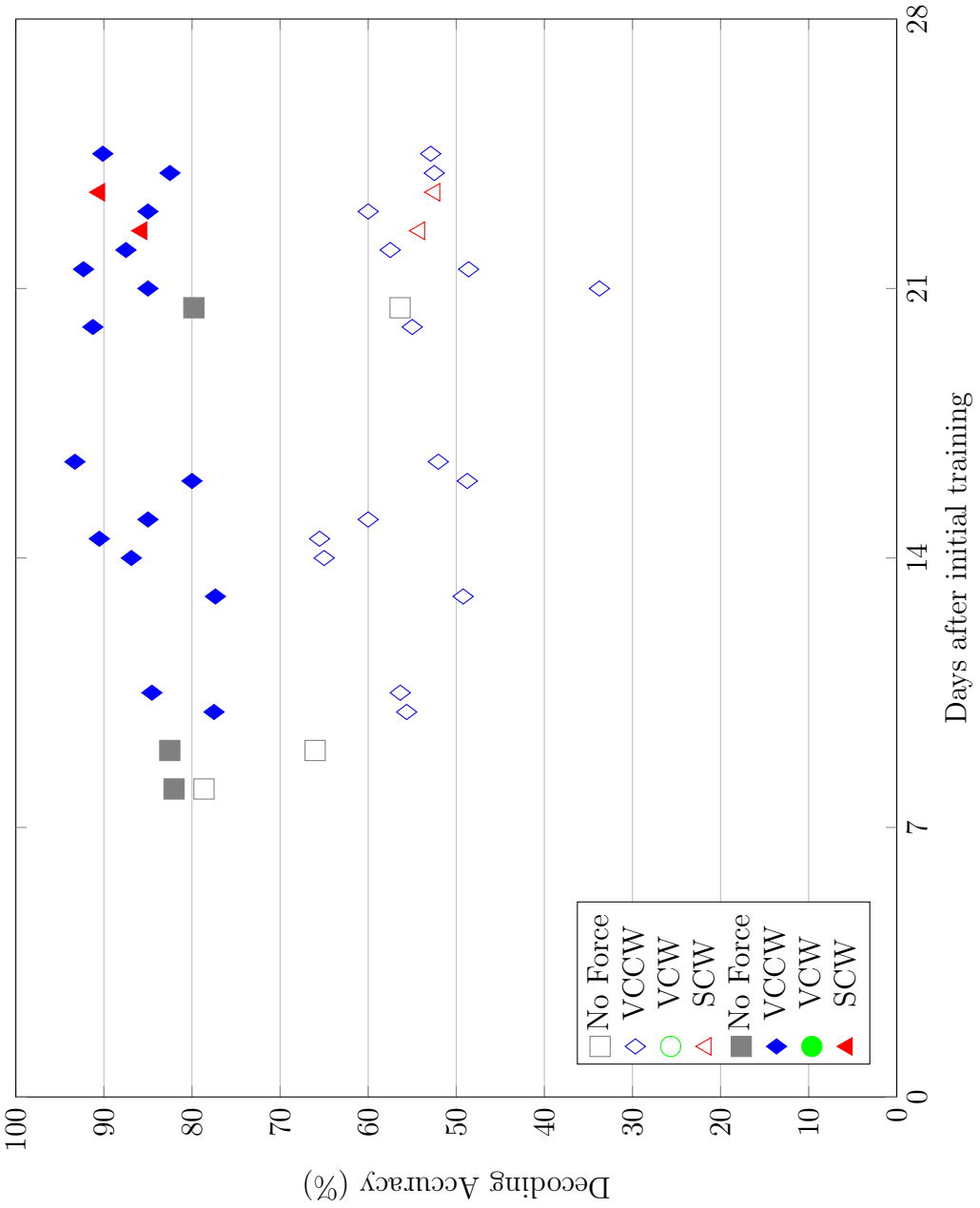


Figure 5.5: Decoding Accuracy in all recorded sessions from monkey H564. The filled icons represent accuracy with adapting model and the unfilled icons represent the accuracy of the baseline model similar to Figure 5.4. The adaptive decoder performs significantly better ($p < 0.01$) than the static decoder over the entire evaluation period.

both the user and the model co-adapt continuously. Such a decoder alleviates user fatigue and 'illiteracy' associated with using BCIs.

5.7 Conclusion

This chapter highlights our analysis of the use of multiple spatial patterns to interpret neural data. The main hypothesis for this analysis is that subject behavior is defined by multiple spatial patterns. To test this hypothesis I trained a model to identify prototypical spatial patterns defining a behavior. I extended this model from decoding discrete targets to tracking hand movements. Adapting the model to changes in neural data provided further improvement and achieved a stable decoding level. I draw the following conclusions:

1. The model provides a robust and accurate decoding of multiple targets over a two week period.

2. Adapting the model over new recording sessions with minimal feedback, improves the decoding to over 95% on the two week period.

3. The model also remained robust to external perturbations and required 1 session to reach a consistent decoding performance (>85%).

4. I observed that adapting the model continuously over all sessions provided better decoding ($p < 0.01$ over all sessions) than just adapting the model on a particular day (Table 5.2).

5. Since there is no requirement of a separate calibration session, this decoder provides results similar to practical BCI.

6. Allowing deviations in the hand estimates provides a model with lesser complexity (number of added spatial patterns), without losing decoding accuracy.

7. The adaptive component of the model is only as good as the baseline model. This is evident from the studies of adapting traditional decoding models (Table 5.3). Although adaptation improves over their original decoding level, they do not reach the decoding ability of the model with multiple spatial patterns.

The evolution of these spatial patterns over multiple days provides further insights to the learning behavior of both the monkey and the model. The next chapter discusses our analysis on characterizing the added spatial patterns.

Chapter 6

Characterizing Spatial Patterns

6.1 Introduction

Chapter 5 presented a model that provides robust decoding using an adaptive model. During the adaptation, I observed certain key components on the spatial patterns of the decoding and hand-tracking model. This chapter discusses characterization of the model and, then applies the characterization to model learning. I introduce a pruning algorithm to remove redundant spatial patterns and reduce model complexity. Further, I provide model characterization in terms of changes in spatial patterns and a strategy to improve decoding using such characterization.

6.2 Model Pruning

The model update suggested in Chapter 5 adds new, relevant feature vectors at each update stage, thereby increasing the model size. The addition of these new bases helps the model to adapt to changes in spatial patterns and provides robust decoding over multiple days, including those under different external field forces. While the accuracy of the model improves with the number of bases, it will result in higher computational and memory cost to

store and process the model. If the number of bases is not limited, the model grows over time and results in the collection of redundant spatial patterns. I observed that the number of new bases added to the model is a factor of

1. Number of accurate decisions (see Section 5.5)
2. Number of redundant output kernels used (see Section 5.4)
3. Error allowed during the pruning stage

The number of new basis added to the model increase with the accurate decisions and output kernels. The more the error allowed, the lesser the new patterns in the model. Thus, the number of total bases could be limited by adjusting the number of output kernels and error parameters. However, such adjustments also affect the accuracy of the model and in general, the decoding accuracy decreases. Hence, I need other methods that could limit the newly added basis.

Further, I observed that as the basis feature vectors evolve over time, older basis vectors have lesser impact on newer sessions. Thus, I can remove some of them and compress the model while retaining performance. For such compression to be useful in a practical BCI setting, it needs to be done without interrupting its user - without any new training, or calibration data. The objective of this compression algorithm is to reduce the number of feature vectors without loss of its functionality. One strategy to reduce relevant bases is to remove those with a smaller weight like a wavelet denoising strategy.

However, this method is blind to the quality of the basis and results in loss of performance. To tackle this, I propose an unsupervised model-pruning algorithm.

Consider a model $\mathcal{M} := \{\mathbf{\Phi}(.), \mathbf{X}, \mathbf{w}\}$, where $\mathbf{X}$ are the neural feature basis vectors and $\mathbf{w}$ their corresponding weights. Output corresponding to these basis vectors could be calculated as

$$\hat{y}_{\mathbf{X}} = \mathbf{w}^T \mathbf{\Phi}(\mathbf{X}, \mathbf{X}) + \mathbf{w}_0$$

The pruned model can be estimated as

$$\mathbf{w}_* : \min ||\hat{y}_{\mathbf{X}} - \mathbf{w}_*^T \mathbf{\Phi}(\mathbf{X}, \mathbf{X})|| + \lambda ||\mathbf{w}_*||_1 \tag{6.1}$$

and ensures that the model retains only relevant feature vectors via the ℓ^1-norm constraint in equation (6.1). The updated pruned model contains only those basis vectors with non-zero weights. This model-pruning algorithm can be used after every session for effective use. As this pruning removes redundant basis vectors, it results in a computationally efficient model. This strategy ensures that the updated algorithm captures short-term (within session) non-stationarities and long-term variability by revising the model.

The model-pruning algorithm effectively reduces model complexity without losing any performance. Since this algorithm does not involve any user input, it can be practically implemented at a) certain pre-determined update interval - after the number of bases exceeds a limit; or b) at the end of a recording session. The first solution allows a control on the memory of the

model by interrupting user during model update. If no user interruption is intended, then the second option presents a good solution with more memory requirements to store the model parameters. Figure 6.1 presents the results of the model compression at the end of every recording session. The left part of the figure shows the number of spatial patterns accumulated during in-session update of the model. After each session a new set of bases (indicated by a new color) are added to the model. Using the compression technique total number of basis used in the model is managed within a limit. Since this is a continuous update, memory requirements to store the model parameters are also alleviated. In our analysis there was no statistical difference between the results from model without update, model updated after K basis, and model updated after a session.

6.3 Characterization

The methods mentioned in chapter 5 allow us to model neural data using multiple prototypical trials. These trials define the behavior of the monkey in that session. During the analysis, I observe that adapting these spatial patterns to new trials improves decoding performance. This means that the new model includes elements that are prototypes of both the training session and the unobserved testing session. Characterizing spatial patterns allows us to preempt the need of adaptation. If the observed spatial patterns are very close to the previously existing patterns, then the need for adaptation could be avoided as the model already provides good decoding.

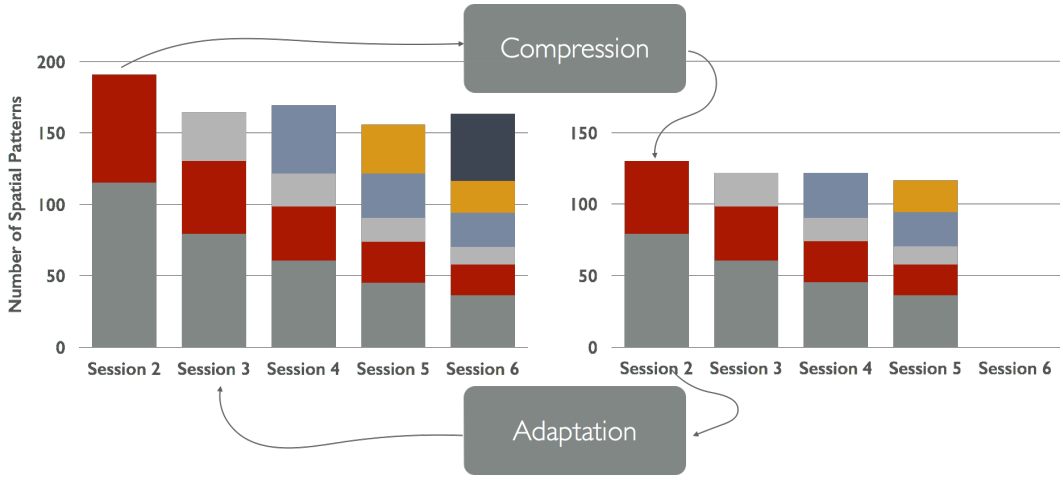


Figure 6.1: The number of spatial patterns in the model before and after compression. This figure shows that by using this pruning strategy the number of spatial patterns in the model is limited. Each color in this figure represents the number of spatial patterns from a new session. This figure shows that using the pruning model limits the number of spatial patterns to around 130.

Neuronal plasticity is the ability of the neural system to adapt to changes in environment, behavior and neural processes, and those resulting after an injury [43, 47, 48, 163, 62, 86, 97, 102, 143, 120, 166]. Neural plasticity is fundamental to the development of novel neuronal pathways in response to demands from the external environments. While the flexibility of the neural pathways allows learning new behaviors, they also need to be rigid enough not to change all at once [94]. During the slow changes in neural patterns, brain reaches an equilibrium state defined by the new set of habits. Some habits are easily acquired while others need training on part of the subject. Characterizing changes in the neural activity and the level of neural activity may provide bio-markers for neuropyschiatric disease [143]. Rehabilitating by conditioning

μ-rhythms in patients suffering from autistic symptoms could induce changes with positive implications on electrophysiology and behavior.

Adaptation to newly learned behavior - reaching a target under external perturbations - alters the underlying topology of primary motor neurons [219]. However, the average strength of their interactions is conserved. Zhu et al., validated the applicability of multi-variate autoregressive modeling to spike trains and found that during adaptation (1 week after introducing the perturbations) the firing rates of primary motor neurons varied [219]. Some had increased firing rates and returned to normal, while others had decreased firing rates. They use the interaction (defined by the coupling matrix) between eight representative neurons to identify the changes in behavior. Similarly [62, 186] measure the causal interactions of multiple neurons in the Granger causality framework and track changes in the neuronal firing rates during adaptation

In this thesis, I propose that multiple prototypical spatial patterns represent a neural behavior. During adaptation, some of the spatial patterns are strengthened while other patterns are lost. I propose to characterize the learning of spatial patterns in terms of

(a) Changes in behavior due to adaptation - in terms of changes in the hand positions defined by the x-,y- positions.

(b) Changes in the spatial patterns - in terms of the Kullback Leibler divergence of the original and new spatial patterns.

6.3.1 Characterizing Changes in Behavior

To characterize spatial patterns, I cluster the hand workspace to multiple areas based on target direction and proximity to target as shown in Figure 6.2. This division of the workspace leads to 17 clusters. I designed multiple prototypical spatial patterns to represent each of these clusters. To identify such spatial pattern I use a one-vs-rest classification algorithm using the formulation below:

$$\sum_{i \in x} w_i \Phi(\mathbf{Z}, \mathbf{X}_i^*) \gtrless \sum_{i \in y} w_i \Phi(\mathbf{Z}, \mathbf{Y}_i^*) \tag{6.2}$$

,where $\mathbf{X}_i$ and $\mathbf{Y}_i$ are neural spatial patterns describing two different clusters. For our analysis, I choose ten prototypes for each cluster. For a new hand reach and its spatial pattern, comparing this pattern with that of the cluster could identify the physical location of the hand. Each of these 17 clusters represents a physical location of the monkey hand defined by its x- and y- positions. One location estimate for the new trial is defined by its cluster. Similarly I estimate hand position from the model discussed in Chapter 5. These approaches provide two measures in terms of the projections of neural data on the workspace. Changes in the neural spatial patterns can be characterized by measuring these differences between the two representative models.

To characterize the variability of spatial patterns, I modeled cluster prototypes on data from session 1. Since the clustering model does not update over the new sessions, it provides a static benchmark comparison to the adapting models. The adapting model evolves, learning new spatial patterns

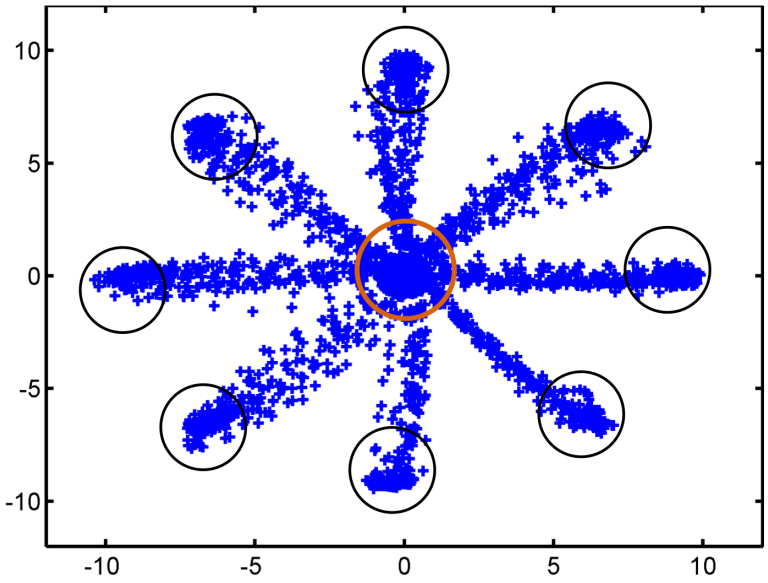


Figure 6.2: Clustering the workspace to multiple areas in order to characterize the changes in spatial location during learning. The spatial workspace represents a $10cm \times 10cm$ area for moving the cursor.

continuously and provides a better estimate on location of the hand position. Measuring the distance between these two projections provides a good estimate of the adaptation. The difference between these estimates shows the changes in spatial patterns over time. One can observe the migration of spatial patterns that represented a particular target in the original session. Figure 6.3 shows the changes in the location of spatial patterns over five recording sessions. Each row of the figure represents clusters corresponding to the center and different targets. I can infer that spatial patterns belonging to the targets in training session tend to move away from it during the testing sessions, and spatial patterns representing one direction during day 0 represent another

direction during other days.

The analysis also allows us to estimate changes in neural patterns during the learning phase of monkey and model. Figure 6.4 shows the histogram and the changes in the spatial patterns over testing sessions. Similar to the above analysis, I modeled clusters on data from session 1, while the adaptive model evolves continuously. From this figure, I can infer that during initial sessions (when new field forces were not introduced) the histograms had a similar structure. While these histograms change when new field forces are introduced, once the new field force is learned, distributions remain consistent with the field force.

6.3.2 Characterizing Changes in Spatial Patterns

Model update involves addition of spatial patterns and pruning out patterns that are non-representative. Such evolution of the model varies over different sessions and is reflected in the total number of spatial patterns added to the model after each session. Figure 6.5 shows the number of spatial patterns added over multiple sessions as a percentage of the total spatial patterns. The figure is color-coded to represent the initial trials in blue and later trials in red. I can infer that during initial sessions (when the field forces were absent) spatial patterns from the first session contributed $> 30\%$ of bases. As field forces are introduced, the number of these initial spatial patterns decreases and new spatial patterns are added to the model. This effect is clearly visible towards the end of the sessions where $< 5\%$ of used bases originate from the

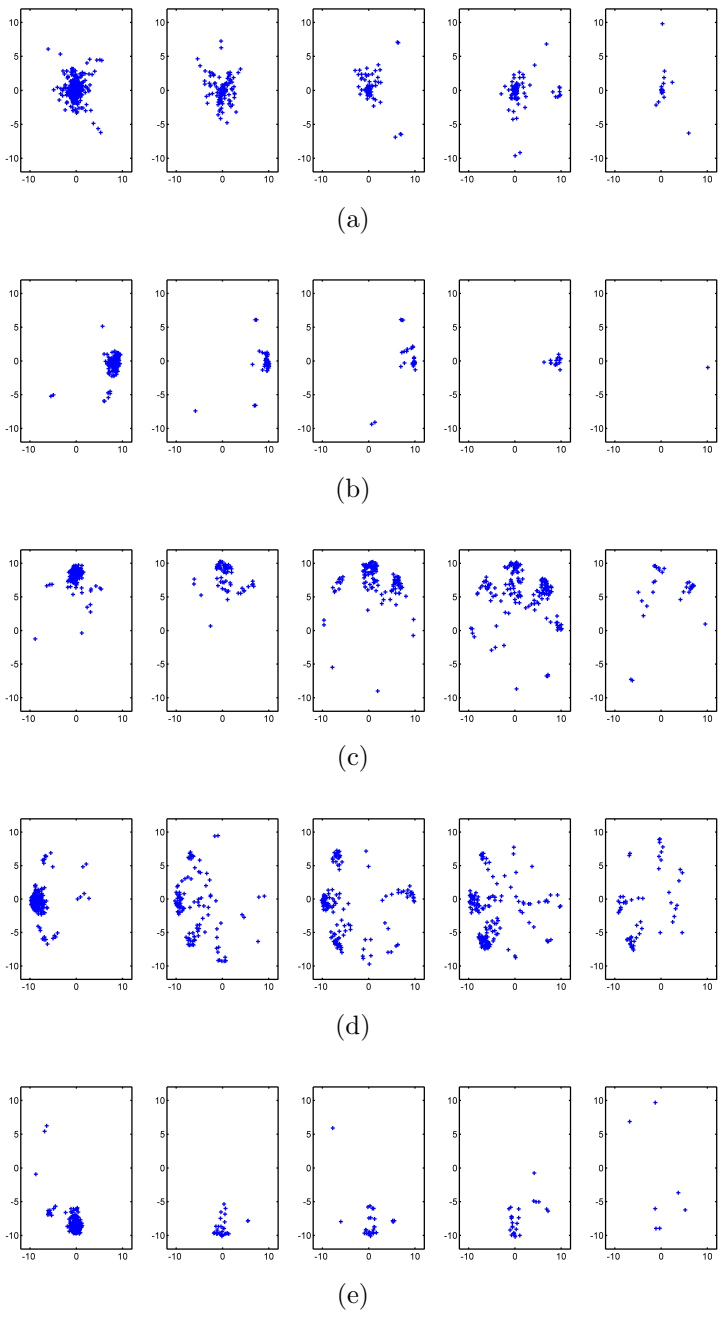


Figure 6.3: Changes in the spatial patterns and the behavioral descriptions over 5 sessions recorded in monkey H464. Row (a) represents the cluster at the center of the hand movement space. Row (b)-(e) represent a different target placed at 0°, 90°, 180°and 270°.

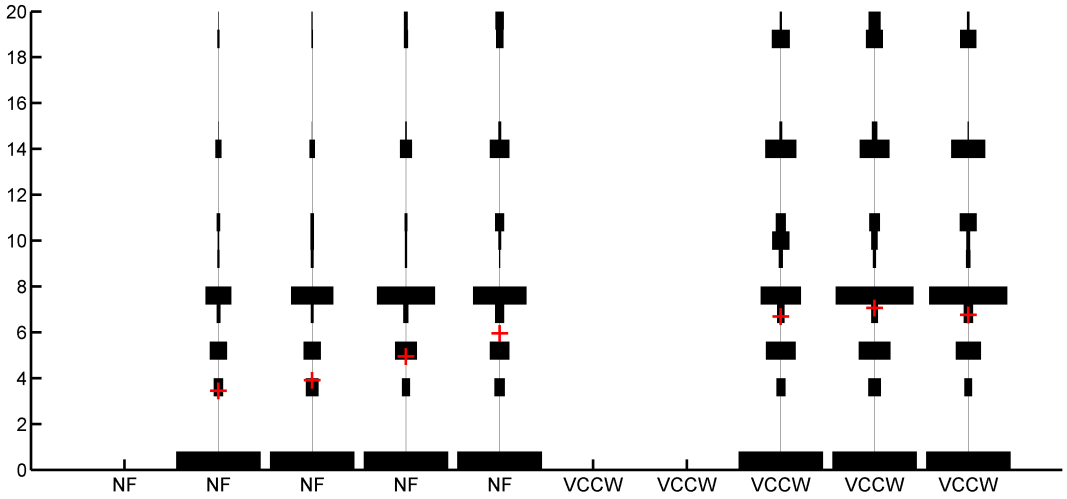


Figure 6.4: Histogram of the difference between estimates from a constant spatial pattern and an evolving model. The y-axis represents the distance between estimates in cm. The x-axis shows the sessions performed on different days and field forces.

first sessions.

I also analyzed quality of the spatial patterns and their changes over time. Since each behavior is characterized by the distribution centered on spatial patterns, I chose to characterize them with the differences in the distributions, in terms of Kullback Leibler divergence (KL) shown in eq (6.3).

$$D_{KL}(P//Q) = \int_{-\inf}^{\inf} \log(\frac{p(x)}{q(x)})p(x)dx \tag{6.3}$$

where P and Q are two distributions on x. I measured the divergence of the models at the end of every session by adapting the above equation.

$$D_{KL}(\mathcal{M}_1//\mathcal{M}_2) = \sum_{\mathbf{X}} \log(\frac{p(\mathcal{M}_1(\mathbf{X}))}{p(\mathcal{M}_2(\mathbf{X}))}p(\mathcal{M}_1(\mathbf{X})) \tag{6.4}$$

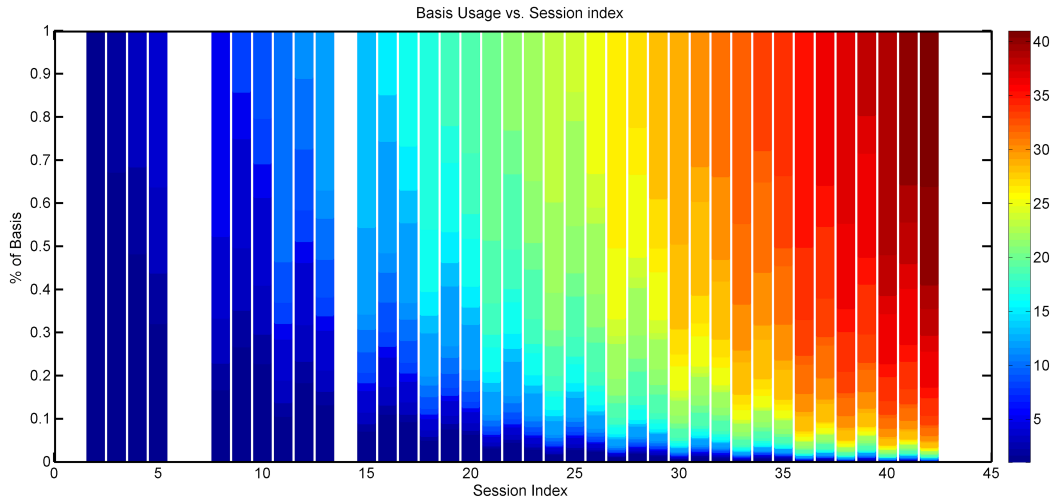


Figure 6.5: Fraction of basis used at each session as a function of their age. The figure is color coded to indicate basis from earlier sessions in blue and those from later sessions in red.

,where $p(.)$ measures the projection of spatial pattern $\mathbf{X}$ on the model $\mathcal{M}$. Such analysis provides insight on the learning rate of the model. Figure 6.6 shows evolution of this estimate over entire sessions performed by monkey H464. From this figure, I can infer that the spatial patterns evolve over time and remain consistent when the behavior remains consistent. This is evident from the rate of increase in first few sessions when new spatial patterns evolve; once the behavior is established the change in spatial patterns remains stable. These results show that, while continuous adaptation adds new spatial patterns and obtains a robust decoder, these spatial patterns appended after an established behavior provide only an incremental change. In the absence of new environments, the subject learns to create consistent spatial patterns.

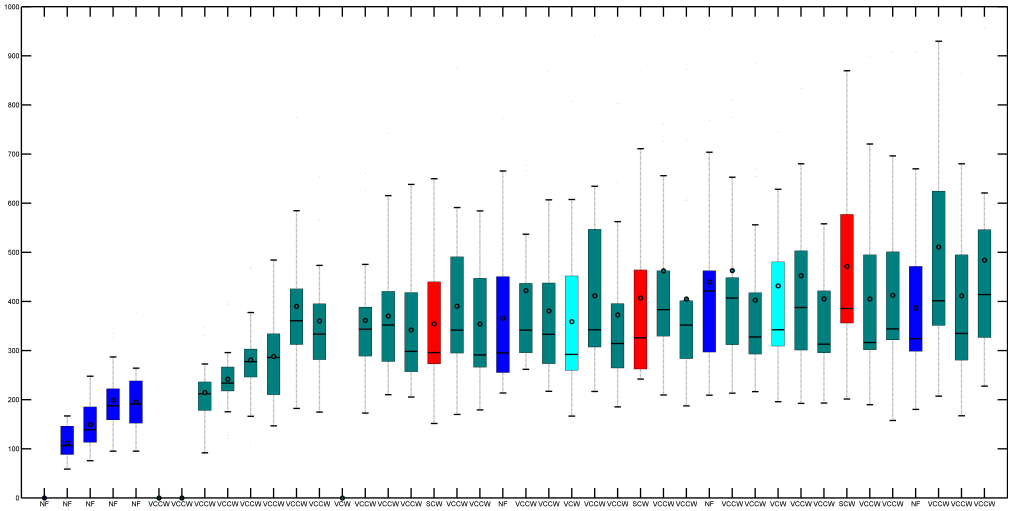


Figure 6.6: Characterizing spatial patterns using KL Divergence metric. The figure shows the changes in spatial pattern models represented by the KL Divergence metric calculated between the model trained on session 1 and the model adapted at the end of each recording session.

6.4 Application of Model Characterization to Model Pruning

Adapting to daily changes in spatial patterns achieves robust decoding. The adaptation of these spatial patterns is based on the fit of the current model on the new data and appends the model with all relevant spatial patterns. The added bases reduce modeling error on the behavior and hence provide better fit on the current data. I observed that such adaptation sometimes adds redundant information to the model. New candidates (spatial patterns) that closely resemble existing spatial patterns are added to the model in addition to novel spatial patterns. Since the updated model adds redundant information, it causes a) Computation overhead, and; b) Introduces noise into the model.

In our observation, model update of spatial patterns with similar information is unnecessary. In this chapter (see 6.2) I proposed pruning the model after each session to effectively reduce the number of basis functions. However, such a strategy does not benefit trials in the current session as model pruning occurs only after all the trials in the session are completed. While the same strategy could be applied at each model update stage during a current session, this requires intensive computation and increases the model update delay.

To avoid such a scenario, I propose to characterize new spatial pattern $\mathbf{X}^*$ and compare it with spatial patterns in the model $\mathbf{X}_0$. Such a strategy needs an effective measure that decides if an updated candidate model $\mathcal{M}^*$ adds value. The adapted model $\mathcal{M}^*$ fits the new behavior better than the initial model $\mathcal{M}$ by virtue of its modeling and hence, any proposed strategy needs to be independent of such fitness measures. The proposed strategy considers spatial characteristics of the candidate spatial patterns and measures distance between spatial patterns of candidate $\mathbf{X}^* \in \mathcal{M}^*$ and patterns in the original model $\mathbf{X}_0 \in \mathcal{M}_0$. The model update occurs only if spatial characteristics of candidates are different from the existing model. To characterize and measure the changes in spatial characteristics of the models, I use Kullback-Leibler divergence between the original model $\mathcal{M}_0$ and the update candidate model $\mathcal{M}^*$. The proposed strategy updates the model only if the divergence estimate exceeds a threshold. Figure 6.7 presents the flowchart of this proposed method.

Figure 6.8 shows decoding accuracy on multiple sessions when I apply the proposed algorithm. I observe that the decoding accuracy improves from

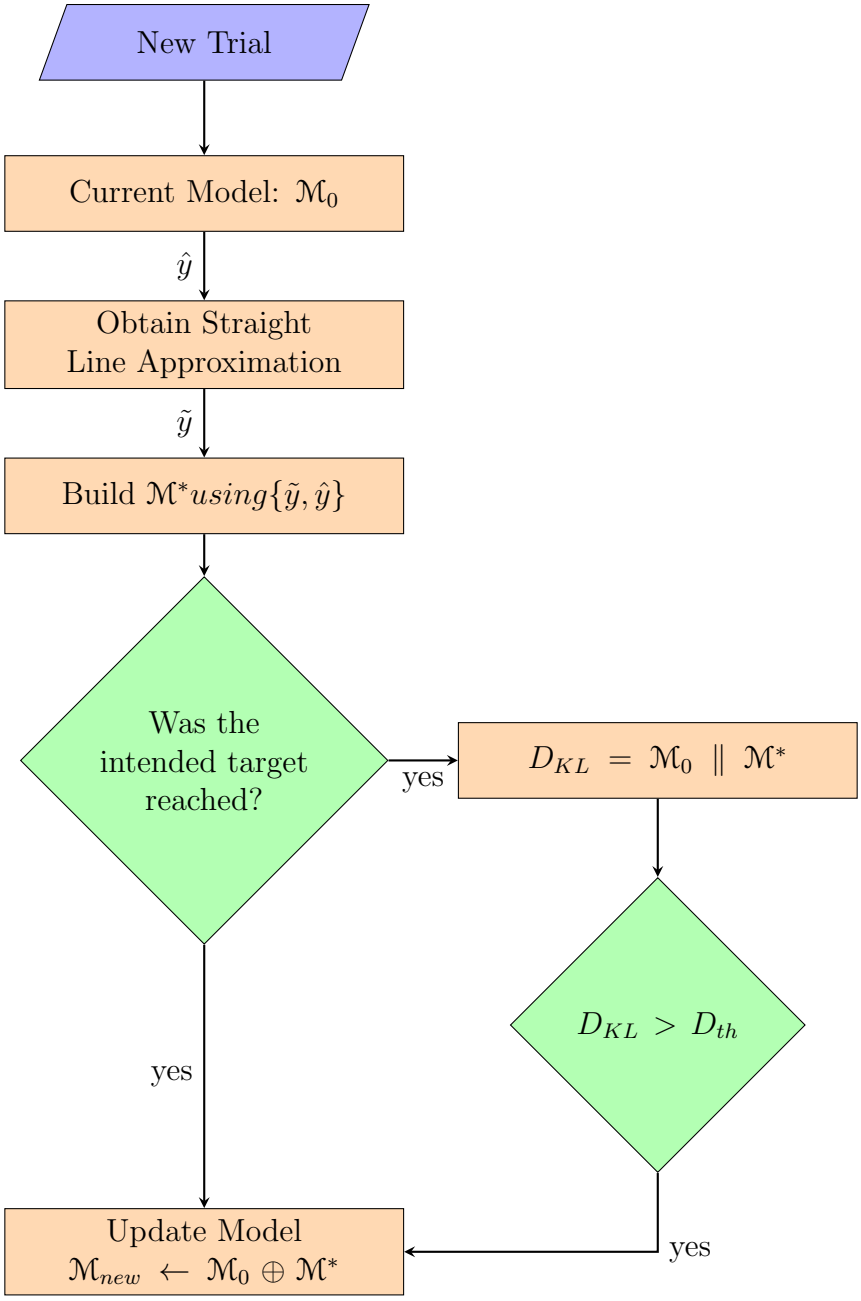


Figure 6.7: Flowchart showing the model update strategy based on KL divergence metric. The left branch of the flow chart requires user input to decide if the intended target was reached. On the right branch, the new model is updated based on the KL-Divergence metric (only if the candidate model adds new information).

85% when redundancy was tolerated, to an average of 90% over six weeks of model testing. From these initial results, I can infer that updates of the model often include redundant information resulting in short-term improvement in performance at the cost of higher model complexity and long-term performance. Removing redundant spatial patterns improves the decoding performance.

6.5 Conclusion

This chapter introduced techniques to prune redundant models and reduce model complexity. In chapter 5 I showed that allowing small errors in the hand position estimates decreases the learning onus on the model and reduces model complexity. The following are some conclusions:

1. A strategy to retrain the decoding model at the end of each recording session is investigated. Such a strategy removed redundant spatial patterns and reduced the model size to a stable level.

2. I characterized the changes in behavior, and also changes in neural patterns.

3. I introduced a metric based on KL divergence to measure changes in the model and used this metric to characterize adapted models.

4. I observed that during adaptation, redundant spatial patterns are added; By using a threshold on the model divergence metric, I reduced model

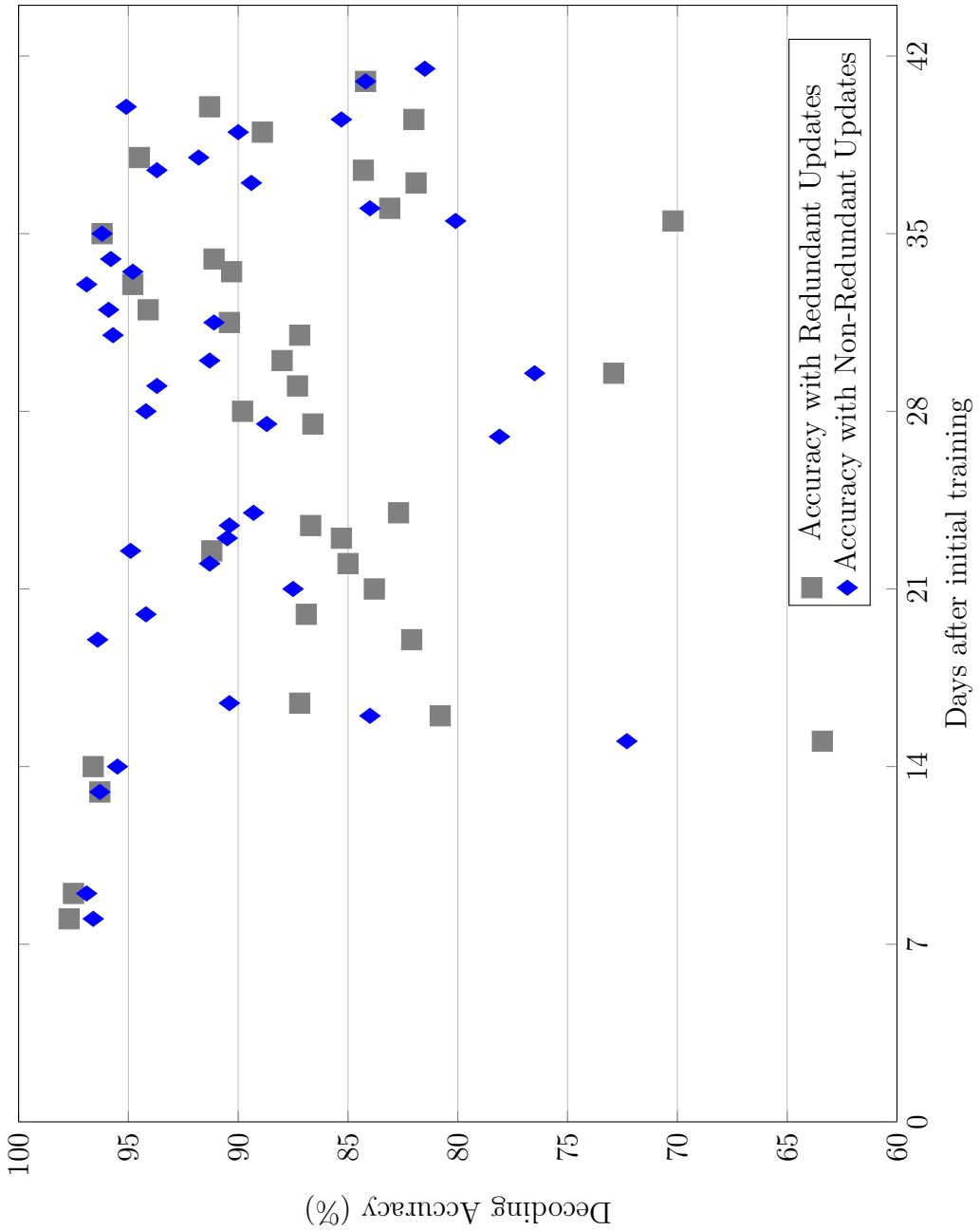


Figure 6.8: Decoding Accuracy of a model using non-redundant spatial patterns in comparison to that of a model with redundant spatial patterns.

complexity and obtained decoding accuracy levels of up to $90\% \pm 6$ over six weeks of testing from $85\% \pm 10$ using a redundant model.

Chapter 7

Estimating Unobserved neural features

7.1 Introduction

Tissue reaction to chronic implantation of electrodes causes changes in electrode impedance and acquired signal quality. Specifically, signals recorded from the same location might have different signal-to-noise-ratio even over consecutive recording sessions resulting in some unreliable channels and unobservable features. This chapter presents a novel approach that overcomes signal variability by identifying reliable channels and features in any given trial. This method also estimates features from the unobserved and unreliable channels and adapts the neural classifier with no user input in real time. The proposed decoder predicts arm movements to one of eight directions in different environmental conditions at an unmatched accuracy of above 90% in two monkeys over 4-6 weeks. The performance improvement is particularly pronounced during sessions with varying external conditions. Since the decoder requires only one dedicated calibration session to train the BCI, it reduces user frustration and improves the practicality and usability of BCI.

7.2 Background

Long-term decoding requires consistency in extracted features across training and testing sessions. But, day-to-day fluctuation in signal characteristics causes variability in extracted features [140, 145, 103]. This remains a major challenge in using LFP for long-term decoding. Causes of such signal variations include subject-induced variations like changes in behavior, motivation, skill, and learning. Other subject independent causes include variability in the location of recording electrode, variability in excitation of neuronal components, and variability in the electrode's electrical characteristics like impedance [97]. Such changes occur due to accumulation of brain tissue around the electrode and scarring effects of the implantation [97]. These variations manifest in different forms: signal quality, power, changes in spatial patterns, etc., and result in inconsistency in derived features [145, 195]. Recent studies have indicated the use of advanced signal processing techniques to overcome such long-term variability [194, 57, 56, 73]. These studies ignore the variability of LFP signal-to-noise ratio (SNR) over multiple days. Specifically, these studies select a set of LFP channels during training and learn the decoder on these channels. LFPs suffer from long-term changes in electrode and tissue properties that affect their SNR [165, 131, 42]. Hence, a practical Brain Computer Interface system should be capable of addressing changes in signal SNR over multiple days. In this chapter, I present a model that filters the neural features by tracking the local feature correlations. Since the actual "signal" and "noise" components of the LFP are unknown, I estimated a

surrogate measure of signal quality, SNR_s.

The main contributions of this chapter are 1) introducing an arm direction decoder that automates channel selection by virtue of SNR; 2) estimating unknown feature parameters by modeling prior information and; 3) adapting the obtained decoder across multiple sessions to overcome variability. These include channel quality and variability in subject behavior due to model latency and changes in environmental effects. Such an adaptive decoder obtained above 93% decoding of eight movement directions over 6 weeks of neural recordings. Further capturing prior information in terms of auto regressive models estimates unobservable channel information and improves decoding performance. Decoders capable of adapting to the above changes reduce user frustration with BCI and increase their practicality [135].

7.3 Challenge with varying SNR

I hypothesized that a behavior is estimated by not just one spatio-temporal pattern but by multiple patterns. The idea is that behavior could be described more accurately by using multiple patterns than by using a single pattern. Using multiple spatial patterns also allows us to understand and characterize the variability in spatial patterns. The algorithm used Relevance Vector Machines (RVM) to obtain the best neural patterns that describe a

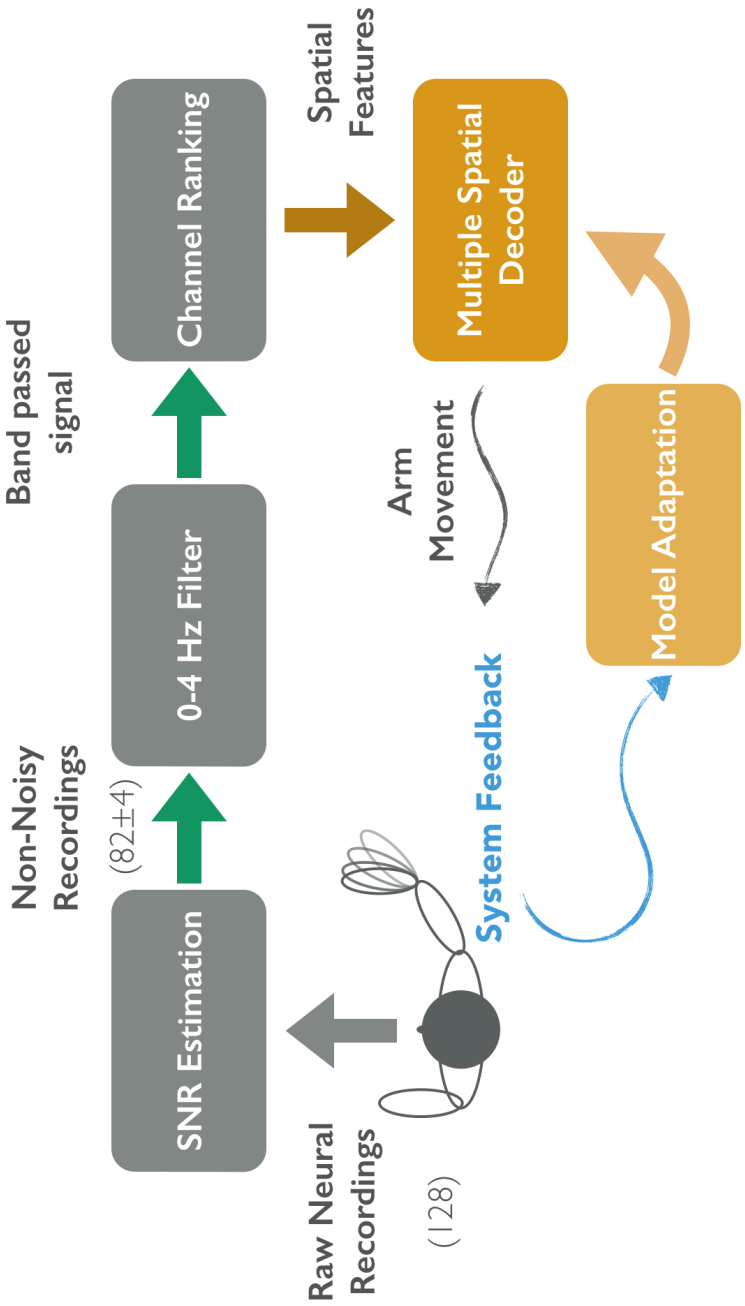


Figure 7.1: A schematic overview of the proposed BCI. The initial decoder is trained using a training data. Model adaptation occurs continuously over the testing sessions by identifying suitable spatial patterns.

particular task, by solving (7.1) [200].

$$\min_{X_b, w_b} \sum_i \parallel p_i - \sum_b w_b \phi(X_i, X_b) \parallel_2 + \lambda |\mathbf{w}|_1$$
$$\mathcal{M} = \{X_b, w_b, \phi\} \tag{7.1}$$

The function $\phi()$ measures the similarity between the two neural spatial patterns X_i, X_b and λ controls the sparsity enforced on the model. For example, ϕ could be a radial basis function (used in this work), or a linear correlation model. p_i represents the two dimensional hand position vector corresponding to the neural feature X_i. To obtain faithful decoding of arm movement, the BCI decodes multiple arm kinematic parameters such as the horizontal (x-) and vertical(y-) arm positions. I proposed to decode these parameters by estimating low level abstract parameters and translating these estimates to arm positions via Kernel Dependency Estimation [209]. This framework employs kernel functions to measure the correlations in hand position and encodes prior information about the target in an elegant way [209].

While such a multi-spatial pattern model provides accurate decoding over training sessions and over initial testing days (spread over 1 week), its performance tapers down over time. Variability of neural patterns between sessions (even conducted on the same day) causes most pattern-recognition algorithms to fail across sessions. Most BCI applications rely on recalibrating the decoder using daily calibration sessions before using it [141, 95, 183, 58]. Such delays fatigue the BCI users and lead to frustration with it. Reducing calibration session time remains a challenge in translating BCI into a prac-

tical application [135, 212]. Since neural adaptation changes the spatial and temporal patterns of brain activity, I proposed that the decoding model also needs a suitable adaptation strategy to track neural adaptation. Identifying suitable neural patterns during BCI use and intelligently incorporating them in the decoder accomplishes decoder adaptation (7.2) [194].

$$\mathcal{M}_u = \{X^* \parallel X^u, w^* \parallel w^u, \phi\} \tag{7.2}$$

, where $\parallel$ is the concatenation operator applied on appropriate dimensions.

Figure 7.1 presents the evaluation scheme of the proposed decoder. The identified noise-affected channels and removes them from analysis as discussed in section 7.3.2. Next, I filter the recorded LFP channels in the delta-band to extract instantaneous qualitative features (Section 7.3.1) and finally decode them to obtain arm movements. Initial calibration of the decoder occurs on the data collected during a training session, where the BCI user provides neural data corresponding to calibration routines. During the evaluation sessions, the decoder undergoes continuous adaptation, based on its performance on the evaluated trials. Adaptation to new neural features provides previously unknown information to the BCI. The proposed structure of the neural decoder allows easy adaptation of the model to incorporate new features.

7.3.1 Neural Features

Neural features used for the decoder were extracted from the δ-band (0-4Hz). Since LFPs follow a $\frac{1}{f}$ frequency response, most of the signal power is

retained in lower frequencies and analyzing this frequency band gives advantage in decoding movement parameters like direction [9, 138]. Further, qualitative measurements like inter-channel ranking extract robust features against variability in non-stationary signal characteristics, and dynamic ranges of LFP power [195, 23, 210]. In cases where data does not follow a normal distribution, ranking methods have distinctive advantages over normal methods [23]. While the distribution of raw features varies significantly across multiple days, I observed that the locations of high and low power ranks on electrode grid remained consistent. I analyzed the rank of channels instead of their original values. At each time sample, channel power was calculated using a rectangular time-window (250 ms) preceding the sample. The channel with the highest power is assigned rank 1, the next channel rank 2 and so on. I propose to use these robust instantaneous rank features to decode arm position.

As a general trend, neural features from training and testing sessions are derived from recording locations selected apriori. However, due to variations in the electrode impedance over time, the quality of recordings is impacted and a loss of recording locations might occur. The current analysis evaluates impact of loss of channels on the performance of an adaptive decoder. Figure 7.2 shows the variation in the quality of three different electrodes. As shown in this figure, some electrodes (orange) present variable signal quality that result in unobservable features during some sessions. This paper presents a model that estimate unobservable feature values, during sessions with poor SNR_s, and improve the decoding performance.

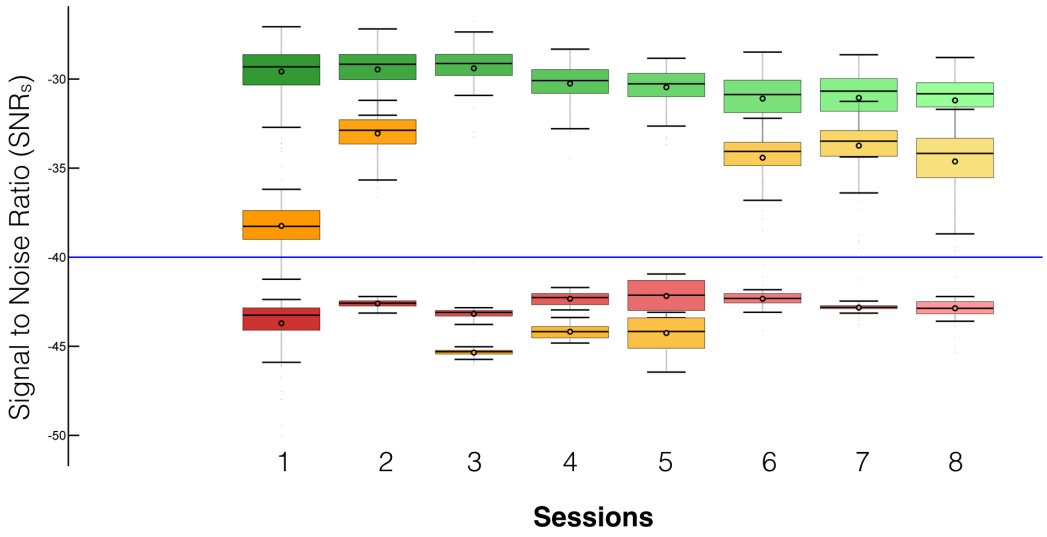


Figure 7.2: Variability in the channel quality of three LFP channels measured in terms of SNR_s over multiple sessions. These electrodes are color coded to show always high quality channel in green, always low quality channel in red and a variable channel in orange. The box plot on each session represents the SNR_s variation in that single session.

7.3.2 Reliable channel identification

Our initial analysis of the LFP signals involved several signal pre-processing steps like time-frequency analysis and histogram analysis to identify noisy LFP channels. Only those channels that passed visual confirmation of these features were deemed high quality channels and were used in further analysis. I performed BCI training and testing on the fixed set of channels to analyze their efficacy. They provided $> 89\%$ decoding accuracy over multiple recording days (6 weeks) including sessions with external field forces.

While such an analysis is useful in establishing the performance of LFP based BCI, the pre-processing step requires manual intervention in the form of visual inspection. Further, only a fixed number of LFP channels were used over the entire recording sessions. In general, it is possible that the quality of LFP signals varies over different experimental sessions and LFP signal having high/poor SNR in one session might provide poor/high signal SNR in a future session. Hence, a practical BCI decoder should estimate movement directions regardless of the loss (or gain) of LFP channels. Recently, Sanchez et. al., used a reinforcement learning method to overcome lost spiking activity during continuous BCI recording [152]. This chapter presents analysis on LFP channels with signal degradation.

Since there is no direct way to estimate the LFP signal strength and a noise estimate I estimate a surrogate measure for SNR, SNR_s, by measuring a channel deviation from the average LFP recording. Consider x_{tr} be the signal recorded on an LFP channel during a single trial. SNR_s is calculated as a function of the deviation from the signal averaged over multiple trials conducted in a given session, $\langle x_{tr} \rangle$, as shown in (7.3).

$$\begin{aligned} \sigma_{tr} &= \sqrt{\frac{1}{T} \sum x_{tr} - \langle x_{tr} \rangle)} \\ \text{SNR}_s &= 20 \log(\frac{1}{\sigma_{tr}}) \end{aligned} \tag{7.3}$$

SNR_s of different channels over the session is presented in Figure 7.3. All channels that have an SNR_s more than -50dB are deemed non-noisy signals and used to train and test the decoding models. Using this scheme, I

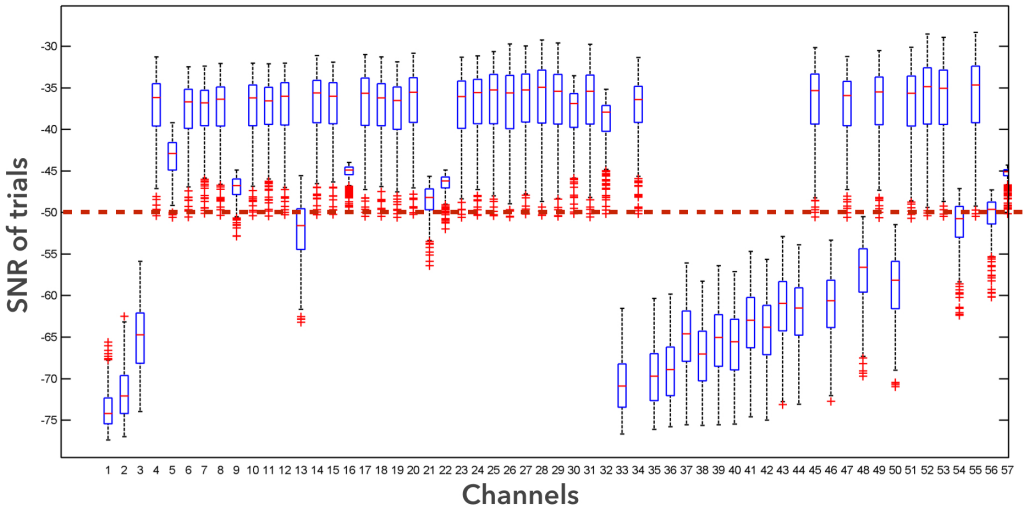


Figure 7.3: SNR_s of LFP channels recorded on the first session in H464. The box plot for each channel represents the variation over the trials recorded in this session. Using a threshold of -50dB, the channels can be classified as high quality channels (> -50db) and low quality channels (< -50db).

observed that 82 ± 5 channels had no noise in H464 and 120 ± 2 in H564.

7.4 Robustness against loss of recordings

The decoding model $\mathcal{M}$ is trained only over a single session and its scope is limited to the electrode locations identified in that session. The model can be written as $\mathcal{M} = \{X, w, \phi_{\mathbf{x}}\}$, where $\phi_{\mathbf{x}}$ calculates the similarity only over the channels ($\mathbf{x}$) in training data X. When trials of the training session need to be evaluated, the decoder possesses knowledge of all channels and making the pattern similarity calculation simple. During model evaluation, the arm position p corresponding to a neural data Y is computed as

$$p = \sum_i w_i \phi_{\mathbf{x}}(X_i, Y) \tag{7.4}$$

Without loss of generality, neural data extracted on the testing day Y could be decomposed as

$$X = \begin{bmatrix} X_c \\ X_x \end{bmatrix}, Y = \begin{bmatrix} Y_c \\ Y_y \end{bmatrix} \tag{7.5}$$

,where $._c$ represents common recordings from the training and testing spatial patterns. The other subscripts represent the electrode locations observed only on that particular session. Thus the pattern similarity ϕ_X could be calculated only over these common channels and the arm position calculated as

$$p = \sum_i w_i \phi_c(X, Y) \tag{7.6}$$

$$\text{,where } \phi_c(X, Y) = \phi(X_c, Y_c)$$

Including noisy channels over the training or testing sessions adds noise to the estimates of hand positions. By selecting channels with high SNR_s, the

proposed method ensures improved estimates of spatial similarity and results in increased direction decoding accuracy. The features X_c and Y_c need to be re-normalized to compute ϕ_X. This normalization only requires the re-ranking of selected channel features. Decoder adaptation follows a similar algorithm to (7.2), by appending appropriate neural features to the existing decoder. As it is possible that channels corresponding to Y_y might be active during a future session, all channels with high SNR_s identified during a session are stored in the model. Thus, the decoder incorporates new information on unknown features.

$$\mathcal{M}_u = \{X^* \parallel_* Y^u, w^* \parallel w^u, \phi\} \tag{7.7}$$

, where the concatenation operator appends the new spatial pattern Y^u to the updated model.

7.4.1 Estimating Partial Observations

The above method discussed in 7.4, estimates the similarity between two neural patterns by selecting channels with high SNR_s in both sessions. By ignoring any channels with low SNR_s, the model improves decoding accuracy. However, this strategy also ignores any information from the remaining high quality channels. I propose that prior knowledge gained from the channel and spatial pattern interaction aids in denoising pattern similarity and adding decoding information. I estimate the similarity measure estimated over all

locations x as $\phi_X(X, Y)$, shown in eq (7.8)

$$\phi_{\mathbf{x}}(X, Y) = \phi(X_c, Y_c) + \phi(X_x, \hat{Y}_x)$$
$$\phi_{\mathbf{x}}(X, Y) = \phi(X_c, Y_c) + \hat{\phi}(X_x, Y_x) \tag{7.8}$$

I observed that the similarity calculated using the common channels is a fraction of the total estimate. To estimate the unobserved portion of the feature, I propose to gain knowledge from the feature evolutions. I track the local correlations between spatial patterns in the form of auto-regressive functions as:

$$\phi_c(t) = \mathbf{H}\phi_{\mathbf{x}}(t) + \vartheta \tag{7.9}$$
$$\phi_{\mathbf{x}}(t+1) = \mathbf{F}\phi_{\mathbf{x}}(t) + \eta \tag{7.10}$$

, where t represents the time step of analysis. $\mathbf{F}$ is the autoregressive parameter that describes the evolution of spatial patterns. $\mathbf{H}$ is the observation parameter that represents the observation, ϕ_c, which is modeled as a fraction of the variable $\phi_{\mathbf{x}}$. ϑ and η are zero mean gaussian white noise variables representing the noise in the measurement and observation models. These equations follow the Kalman filtering dynamical model system that improves the observations based on prior knowledge. Using the "**Predict**" and "**Update**" phases of the Kalman filter, the observation could be corrected closer to the model estimates [99]. The design of Kalman filter parameters, involves calculating the auto-regressive parameters, $\mathbf{F}$ and the respective noise covariance on the neural samples recorded on the training session. For this application I design the

observation matrix, **H**, as a scalar under the assumption that all spatial filters are partially observed.

7.4.2 Decoder Adaptation

As mentioned above, an important element of the decoder is the adaptation strategy presented in equations (7.2) and (7.7). Under this strategy, selected neural patterns are added to the decoding model to improve performance over future trials (from the same as well as next sessions). Since the channels of added bases and new testing trials remain the same, no ambiguity about the similarity measure exists. However, measures corresponding to basis from a previous session are denoised using Kalman filtering.

$$\hat{\Phi}_{new} = \begin{bmatrix} \hat{\Phi}_{\mathbf{x}} \\ \Phi_u \end{bmatrix} \tag{7.11}$$

, where $\hat{\Phi}_x$ is the filtered version of the features estimated by the Kalman filter.

Adaptation by assimilation ensures the stable performance of neural decoder during the evaluation session. Pruning of the accumulated basis by removing redundant neural features constrains the size of the decoder. I use the unsupervised pruning algorithm presented in Section 6.2. The next section details the results achieved using the methods described in this section.

7.5 Results and Discussion

The objective of this project is to design long-term decoding capability that provides stable performance, with minimal re-training sessions to miti-

gate BCI user frustration. To evaluate this, I train the movement decoder on a single session and evaluate its performance over the rest of the sessions. I measure performance as decoding accuracy (DA): percentage of accurately predicted targets in a session, and measure the correlation between actual hand movement and the prediction estimated from neural data. During each evaluation session, the model is adapted after every K (= 25) trials. Adaptation of the model begins first by predicting the direction associated with the trial's neural patterns. Under the assumption that the BCI user intends to reach the target in a straight path, I compare the prediction to an expected signal, modeled as a straight line from the center to the predicted target [194]. The adaptation strategy uses only accurate reaches to adapt the decoder by selecting neural patterns that can reduce the error between the prediction and the desired straight line approximation (7.7). Feedback to the BCI system could be delivered via multiple modes including vocal cues, error related potentials, or any residual muscle activity [152, 50, 53, 60, 84]. Such a binary feedback is enough to inform the decoder if the intended target was reached and improves accuracy in the future sessions.

I measured and compared the decoding accuracy of different decoders. I trained decoding models on the first session (with no field forces) and applied over chronological sessions spread over 4-6 weeks (including sessions with novel external field forces). I should note that the subjects were unfamiliar with these perturbations and required multiple sessions to learn and perform the target reaching tasks. I expect that monkeys counter the perturbations in

the behavior by modulating neural patterns to adapt to the dynamics. Model adaptation learns these changes in neural patterns. I performed initial analysis on a fixed set of channels selected via visual inspection. This set of channels remained consistent over all the training and testing sessions and did not require the estimation of unknown features. This decoder predicts one of eight arm directions at $> 89\%$ over the evaluation sessions. At a random classification, the decoding would achieve only 12.5%.

I trained and tested all the decoders on the same training, testing and adaptation scheme to ensure fair comparison. Any additional model parameters like $\mathbf{F}$ and the error covariance matrix were also trained on the same training data, assuming no knowledge of future sessions. This parameter can be estimated using multiple training reaches. By calculating the $\phi(t+1)$ and $\phi(t)$ respectively, $\mathbf{F}$ can be calculated as

$$\mathbf{F} = \phi_{\mathbf{x}}(t+1)\phi_{\mathbf{x}}(t)^{\dagger}$$

, where † is the pseudo inverse of the signal. Since, ϕ_x can be completely determined during training, the matrix $\mathbf{F}$ can be characterized before the testing and evaluation phase. Decoder adaptation (7.4.2) filters only observations corresponding to previous day, and requires no update of $\mathbf{F}$. After evaluating and adapting during the test session, I update $\mathbf{F}$ to correspond to the auto-correlation of the new neural features $\hat{\Phi}_{\text{new}}$.

Parameter $\mathbf{H}$ represents the relation between observation and the learned correlation model. In this analysis, I begin by designing $\mathbf{H}$ as a scalar multi-

ple of an identity matrix under the assumption that all spatial features have similar uncertainty. A lower **H** assumes higher uncertainty between observations and the auto-correlation model and vice-versa. During application of new field forces, I expect that uncertainty increases causing the observations to drift away from the modeled values. Hence, I expect that adapting **H** over different field forces sessions is advantageous. Our heuristic modifies **H** based on the recent history of the sessions. **H** is tuned for a lower value after a change in field forces, reflecting the lower correlation between observation and auto-regressive modeling.

The decoding results are presented in the Figures 7.4 and 7.5 for monkeys H464 and H564 respectively. I introduced field forces to monkey H464 after two weeks and H564 after 10 days. Decoders must overcome not only the neural variability due to time lag but also modulations in neural patterns due to environmental effects. Results from the figures show the improvement of decoding results when decoders consider only channels with high SNR. The improved accuracy ($> 10\%$ with $p < 0.01$) is especially noticeable on days 15, 28 and multiple sessions after day 35 for monkey H464 (Figure 7.4) and on days 16 and 20 for monkey H564 (Figure 7.5). Overall, the presented decoders improved up to 3% (not significant improvement over all session at $p < 0.01$) over decoders with fixed channels. Table 7.1 presents the performance of the three decoders in different phases of the experiments. In H564, the fraction of common channels between sessions is $98\% \pm 1$. This implies that the auto-correlation model represents the observation accurately. Due to this, I observe

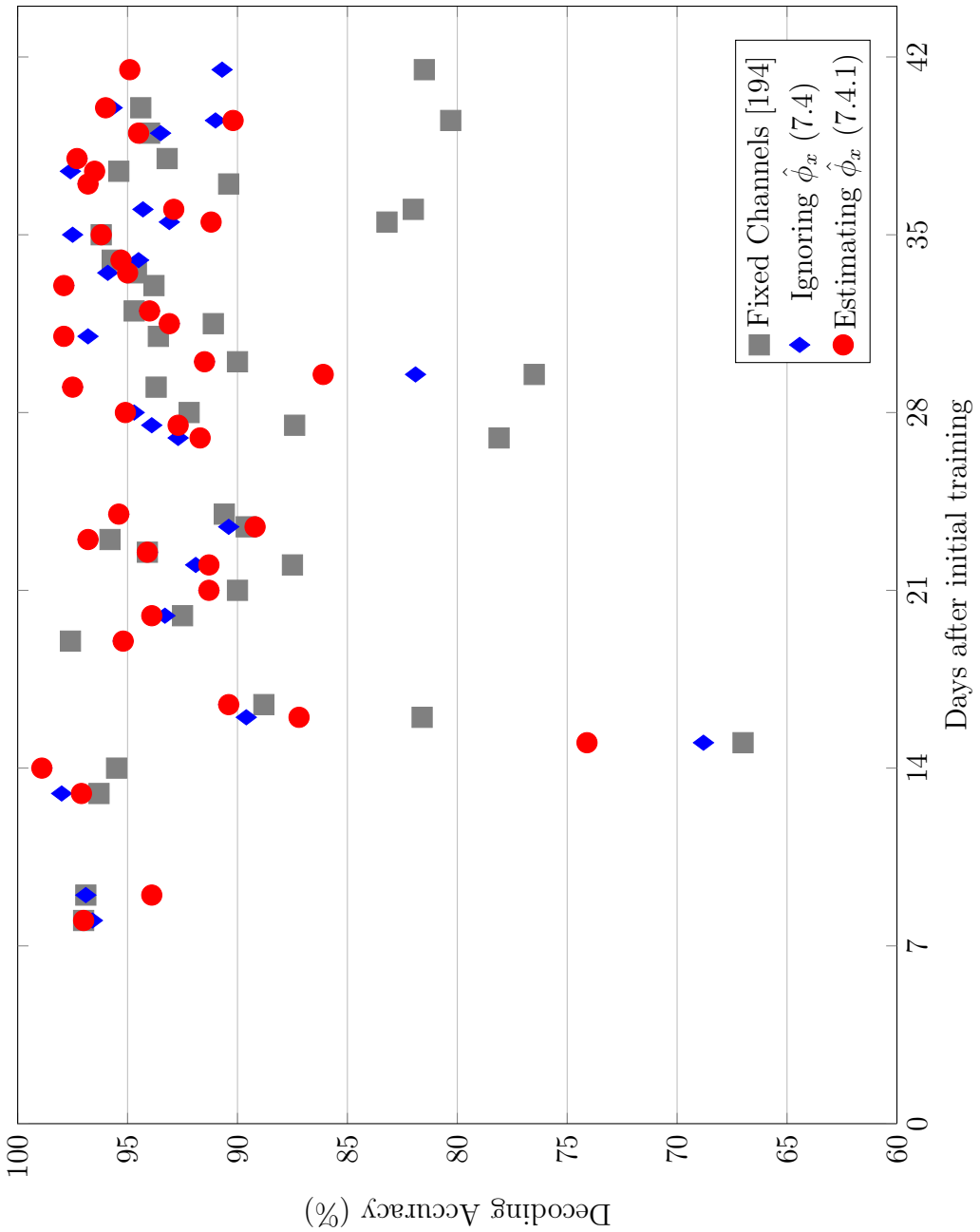


Figure 7.4: Decoding Accuracy over multiple testing sessions recorded from subject H464 for decoders presented in the chapter. I train the decoders on neural data recorded on day 0. Field forces are applied on sessions after day 14 and vary on different sessions.

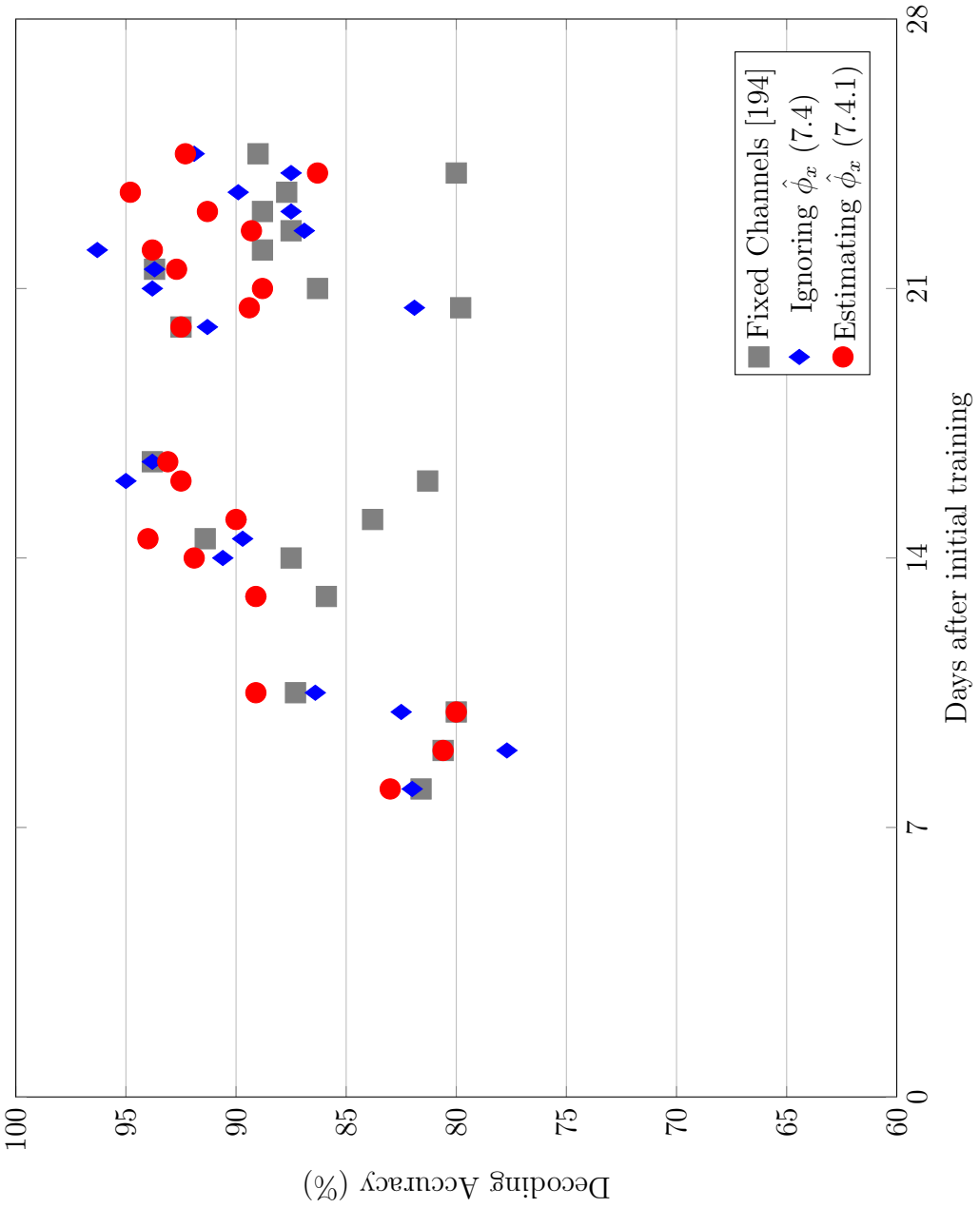


Figure 7.5: Decoding Accuracy(DA) over multiple testing sessions recorded from subject H564 for decoders presented in the chapter. Decoders trained on day 0 are applied on sessions after day 10. I observed that DA remains stable over 20 sessions and using high quality LFP signals improves decoding especially on day 15,16 and 20 (by 10%).

3% decoding accuracy improvement by estimating unknown features over a fixed channel decoder. In H464 this fraction is only $87\% \pm 1$, resulting in an improvement of 1% (not significant at $p < 0.05$) of decoding accuracy.

I can infer that using high SNR channels from a session improves decoding accuracy ($p < 0.01$ for some sessions using McNemar's test). Removing low SNR channels eliminates any creep of noise in the model without affecting the quality of the neural patterns. On an average $80\% \pm 3$ of the channels were common across the sessions in both the monkeys. Since the model training selects a few prototypical trials to represent each behavioral task, selecting common high SNR channels leads to improving the decoding accuracy. Further, estimating missing unknown neural feature data by using prior knowledge improves accuracy to the decoder by 1%. The decoder performance, in particular, improved in sessions that experience a change in the field force direction, as shown in the last row of Table 7.1.

7.6 Analysis of BCI system

I analyzed the different sub-systems of the decoder, shown in Figure 7.1, and their effects of decoding performance. Below are some of our findings:

1. Removing the channel SNR estimation and selection module results in using a fixed set of channels analyzed on the training session. Since the channel SNR estimation is not performed on every session, the decoder uses only those channels selected during the initial training session. This

Table 7.1: Decoder performance and comparison across different phases of the recordings. For monkey H564, the average decoding is presented across all 20 recording sessions spread of 4 weeks. For monkey H464, the average decoding is calculated over 37 sessions spread over 6 weeks.

Session	Using Fixed Electrodes	Updating Electrodes	**Estimating Partial Observations**
H464			
Average Decoding (6 weeks)	89.8	93.5	**93.5**
Before Field Forces (2 weeks)	96.6	97.6	**96.7**
During Field Forces (4 weeks)	89	93	**93.1** ($p < 0.01$ **for some sessions**)
Novel Field Forces (9 sessions)	85	89.5	**89.8** ($p < 0.01$)
H564			
Average Decoding (4 weeks)	86.3	88.9	**89.7**
Before Field Forces (1 weeks)	81.1	79.9	**81.8**
During Field Forces (3 weeks)	86.9	89.9	**90.6** ($p < 0.01$ **for some sessions**)
Novel Field Forces (4 sessions)	83.8	85.3	**88.4** ($p < 0.01$)

Table 7.2: Impact of Feedback Accuracy on Adaptive decoder performance. Performance of the algorithm during different recording sessions is compared. For this comparison, the model was trained on neural data from day 0 and its performance evaluated on future sessions.

Session	before introducing field forces (1-2 weeks)	during varying field forces (3-4 weeks)	All Sessions (4-6 weeks)
H464			
No update	81	31	36
Update w/o feedback	96	76	78
75% Accurate feedback	96	86	87
80% Accurate feedback	97	87	88
90% Accurate feedback	96	88	89
100% Accurate feedback	96	89	90
H564			
No update	72	54	56
Update w/o feedback	70	72	72
75% Accurate feedback	77	81	81
80% Accurate feedback	78	84	83
90% Accurate feedback	79	86	85
100% Accurate feedback	81	87	86

results in the model described in section 7.4. While this model fails to adapt to changes in channel SNR, due to the robustness of the feature extraction and adaptation, it still provides stable decoding (at accuracy $\sim 90\%$).

2. I analyzed the band-features used for direction decoding. The current analysis focused on δ-band, since prior works provided encouraging results in this band. I also noticed that the high-γ band added some sup-

plementary decoding information. Since LFP signals have a $\frac{1}{f}$ frequency response, most of their power is contained in the lower frequency bands like δ-band. Hence features based on raw power (without any band pass filtering) inter-channel ranks aligned closely with δ-band power inter-channel ranks. Analyzing features extracted from the raw signals provided similar neural patterns. Removing the band-pass filtering module resulted in similar (no statistically significant change) decoding accuracy over all sessions.

3. The strategy assumes a feedback on the accuracy of the intended target reach. I realize that errors in the feedback mechanism might induce performance degradation to the overall system. State-of-the-art EEG based error detection techniques perform at 75% accuracy to identify human induced errors [84, 170, 197]. I performed monte-carlo simulations varying the feedback accuracy and measuring its impact on overall decoding. The results, presented in Figure 7.6, indicate that feedback at even 75% accuracy reduces the decoding by 3-5%.

4. Table 7.2 presents the average decoding of 10 such simulations. I make two interesting observations. Firstly, providing any feedback improves the performance in comparison to without feedback. Of course the best performance is observed with the most accurate feedback. I compare the results with the performance of a static decoder and a decoder that assumes its previous version is always accurate in reaching the target.

The impact of feedback is evident in monkey H564, where the static decoder accuracy is 70%, the feedback improves the accuracy to above 77%. Secondly, this strategy is robust to erroneous feedback and reduces the impact of error propagation. Even when all reaches are used for update (irrespective of errors), the strategy improves accuracy (from 36% to 78% in H464 and 56% to 72% in H564 as shown in Table 7.2). In the presence of both accurate and inaccurate samples (due to feedback), the decoder provides higher weight to the accurate feedback samples and lower weights to inaccurate samples. Further, these samples from the inaccurate feedback get pruned out during the Model pruning stage of the algorithm. This ensures that the strategy remains robust to any inaccuracies in the feedback.

5. The adaptation strategy uses a straight line trajectory as the desired output. I replaced the straight line with a curved trajectory (parabolic) from the center to the target. Our analysis shows that, as long as the same desired trajectory is consistently used over all the recording sessions, decoding accuracy remains the (significantly at $p = 0.01$) same with either a linear or a non-linear trajectory.

7.7 Conclusion

In this chapter, I presented neural decoders that provide robust arm decoding. These decoders are robust against LFP variabilities over time, en-

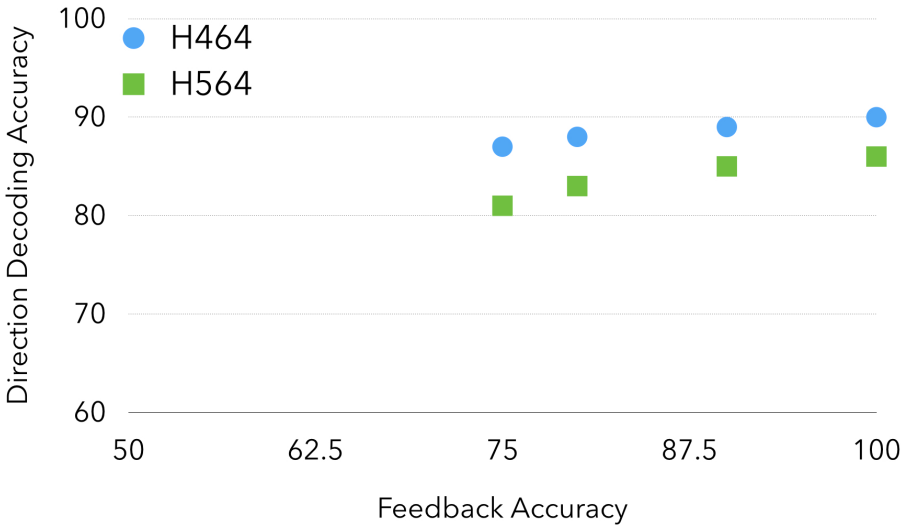


Figure 7.6: Impact of Feedback Accuracy on Adaptive decoder performance. Performance of the algorithm during different recording sessions is compared. For this comparison, the model was trained on neural data from day 0 and its performance evaluated on future sessions.

vironmental conditions like external field forces, and changing channel SNR. Since multiple motor patterns accomplish the hand reach, I proposed that multiple task-related neural patterns encode the reaching task. The decoder identifies such task related neural patterns to predict arm-movement direction. I presented an adaptive strategy to incorporate new neural feature patterns, observed during evaluation, into the decoder.

1. I observed that channels exhibit different SNR over multiple recording sessions and present a decoder that identifies channels with high SNR and uses them for direction prediction.

2. The decoder presents a novel way of estimating unobservable neural patterns by modeling feature correlations and system dynamics. This model provided up to 94% direction decoding accuracy in one monkey and 89% accuracy in another over 6 and 4 weeks respectively.

3. The adaptation strategy requires only a binary feedback input on the performance of the decoder and improves model performance.

4. I should note that recordings occurred in an open-loop fashion, where the monkey was unable to learn the decoders. I anticipate that the decoder performance would improve in a closed-loop setting, where the subject learns the dynamics of the model.

In a practical setting, I anticipate that the adaptation occurs in a symbiotic fashion allowing both human and machine to learn from each other. Decoders with such characteristics need very few calibration sessions and improve BCI usability in practical applications.

Chapter 8

Conclusion

This thesis focuses on designing robust neural decoders for arm movement direction decoding. I trained neural decoders on a single recording session and evaluated their performance over subsequent sessions. Through this analysis, the following thesis statements were tested:

(a) *Developing novel and time-robust neural features overcomes signal variability and improves decoding of hand movement over multiple days.*

(b) *Encapsulating the variability of subject behavior in multiple spatio-temporal patterns and capturing the changes in subject behavior by adapting decoding model to novel features improves long-term decoding*

(c) *Estimating unobservable feature parameters by capturing prior model information overcomes day-to-day variation in channel SNR.*

Below is a summary of my contributions: I developed robust neural features in the form of instantaneous power ranks in the sub-band filtered (0-4 Hz) local field potentials. These features were initially compared using a traditional neural analysis approach - Common Spatial Patterns. These features provided robust decoding over multiple days, especially when there were no

external perturbations, and exhibited improvements over existing traditional methods. The robust performance of features arises by overcoming daily variations of signal power level. Use of rank features instead of raw power values provided an average improvement of 20% (36% using traditional methods to 56% using rank features) over the first two weeks of decoder testing. Combining multiple evolutions of spatial patterns provided further improvement of 6% over the same two weeks ($p < 0.01$ using McNemar's test). These results prove the feasibility of ranked spatial patterns in local field potential and their evolutions to decoding movement directions.

In Chapter 5, I introduced a new methodology to decoder training. I hypothesized that subject behavior is described by multiple neural patterns and their evolutions. To test this hypothesis, I developed a model that extracts multiple spatio-temporal patterns based on rank features for each movement direction. This model was extended to tracking hand movement during the entire trial rather than just detecting the final target. The model provided an average decoding of 82.5% over two weeks from a model trained on a single session. I observed that reduced decoding was due to changes in spatial patterns, and monitoring these variations provided further improvement. By using a decoding model that adapted to the changes in neural patterns, there was improved decoding performance to above 95% over two weeks of model evaluation ($p < 0.01$ using McNemar's test). Improved decoding performance is also observed when external perturbations were applied on the hand. The adapting model provided a robust decoding of above 85% on sessions where

field forces were applied on the hand.

In Chapter 6 I provided multiple strategies to limit the model redundancy during adaptation. I observed that model size increased after each adaptation and resulted in redundant spatial patterns, thereby affecting the computational complexity of the model. To limit this redundancy, firstly I proposed to compress the model after each session in an unsupervised fashion without the need of user interaction. Next, I proposed the detection of redundant spatial patterns by evaluating their similarity to the existing model. Such detection enabled a smart update by allowing only distinct spatial patterns to be appended to the model. This strategy provided a decoder with 17% less complexity in terms of new spatial patterns added to the decoder, and with 5% improvement of average decoding over six weeks of model testing ($p < 0.01$).

Finally, in Chapter 7 I presented decoders that remain robust to changes in LFP channel SNR. I observed that channels exhibit different SNR over multiple recording sessions and present a decoder that identifies channels with high SNR and uses them for direction prediction. The decoder presents a novel way of estimating unobservable neural patterns by modeling feature correlations and system dynamics. This model provided up to 94% direction decoding accuracy in one monkey and 89% accuracy in another over 6 and 4 weeks respectively.

8.1 Future Work

1. **Application in an online BCI**: The analysis in this thesis shows that adapting a model in an open-loop environment provides robust performance. In a closed-loop environment, BCI user has the ability to monitor BCI performance and alter the neural patterns accordingly. Existing research shows the ability of BCI users to adapt to a stable pattern and generate spatial patterns consistently. Incorporating the adaptive nature of the proposed decoder enables both the BCI user and the system to co-adapt.

2. **Learning New Targets**: In this thesis, I have evaluated performance of the decoder when monkeys experienced new filed forces. The analysis showed that decoders could be built for more than one neural task of reaching the target and compensating the perturbation. Further analysis on the introduction of new targets improves our knowledge of neural learning. Specifically, one can examine if new spatial patterns are generated for any new target or if the existing spatial patterns and decoders identify the target. Successful models imply that only a few directions are required during training to enable quicker and faster training sessions.

3. **Adaptation without feedback**: In the proposed adaptation strategy, knowledge of an accurate reach is required to decide if a neural pattern helps in the successful adaptation of the model. This feedback could be

received from hand movement and position of the hand. For example, in a closed-loop BCI, since the user attempts to reach the target, a static hand location informs the BCI system that the desired target was achieved. This information could be used as a feedback surrogate. Alternatively, multiple decoders and experts could generate feedback to the BCI system. This increases the complexity of the system and is dependent on the accuracy of all involved experts. Another suggestion is to monitor the changes in spatial characteristics of neural patterns and characterize these patterns during an accurate and inaccurate reach.

Bibliography

[1] *Brain-Computer Interfaces - An international assessment of research and development trends.*

[2] "Schwartz A.B., Taylor D.M., and Tillery S.I.H.". Extraction algorithms for cortical control of arm prosthetics. *Current Opinion in Neurobiology*, 11:701–708(8), 1 December 2001.

[3] Vahid Abootalebi, Mohammad Hassan Moradi, and Mohammad Ali Khalilzadeh. A new approach for eeg feature extraction in p300-based lie detection. *Comput. Methods Prog. Biomed.*, 94(1):48–57, April 2009.

[4] Mehdi Adibi, James S. McDonald, Colin W. G. Clifford, and Ehsan Arabzadeh. Adaptation improves neural coding efficiency despite increasing correlations in variability. *The Journal of Neuroscience*, 33(5):2108–2120, 2013.

[5] Setare Amiri, Reza Fazel-Rezai, and Vahid Asadpour. A review of hybrid brain-computer interface systems. *Adv. in Hum.-Comp. Int.*, 2013:1:1–1:1, January 2013.

[6] R.A. Andersen, J.W. Burdick, S. Musallam, H. Scherberger, B. Pesaran, D. Meeker, B. D. Corneil, I. Fineman, Z. Nenadic, E. Branchaud, J.G. Cham, B. Greger, Y.C. Tai, and M.M. Mojarradi. Recording advances

for neural prosthetics. In *Engineering in Medicine and Biology Society, 2004. IEMBS '04. 26th Annual International Conference of the IEEE*, volume 2, pages 5352–5355, Sept 2004.

[7] Richard A Andersen, Sam Musallam, and Bijan Pesaran. Selecting the signals for a brainmachine interface. *Current Opinion in Neurobiology*, 14(6):720 – 726, 2004.

[8] Eishi Asano, Csaba Juhsz, Aashit Shah, Otto Muzik, Diane C. Chugani, Jagdish Shah, Sandeep Sood, and Harry T. Chugani. Origin and propagation of epileptic spasms delineated on electrocorticography. *Epilepsia*, 46(7):1086–1097, 2005.

[9] Itay Asher, Eran Stark, Moshe Abeles, and Yifat Prut. Comparison of Direction and Object Selectivity of Local Field Potentials and Single Units in Macaque Posterior Parietal Cortex During Prehension. *Journal of Neurophysiology*, 97(5):3684–3695, 2007.

[10] G BakIr. Extension to Kernel Dependency Estimation with Applications to Robotics, 2005.

[11] Arjun K. Bansal, Carlos E. Vargas-Irwin, Wilson Truccolo, and John P. Donoghue. Relationships among low-frequency local field potentials, spiking activity, and three-dimensional reach and grasp kinematics in primary motor and ventral premotor cortices. *Journal of Neurophysiology*, 105(4):1603–1619, April 2011. PMID: 21273313 PMCID: PMC3075284.

[12] ?lvaro Barbero and Moritz Grosse-Wentrup. Biased feedback in brain-computer interfaces. *Journal of NeuroEngineering and Rehabilitation*, 7(1):1–4, 2010.

[13] C. Bédard, H. Kröger, and A. Destexhe. Model of low-pass filtering of local field potentials in brain tissue. *Phys. Rev. E*, 73:051911, May 2006.

[14] Claude Bedard and Alain Destexhe. A generalized theory for current-source density analysis in brain tissue. *arXiv:1101.1094 [physics, q-bio]*, January 2011. Physical Review E 84: 041909 (2011).

[15] Yoshua Bengio, Aaron C. Courville, and Pascal Vincent. Unsupervised feature learning and deep learning: A review and new perspectives. *CoRR*, abs/1206.5538, 2012.

[16] H.L. Benz, Huaijian Zhang, A. Bezerianos, S. Acharya, N.E. Crone, Xioaxiang Zheng, and N.V. Thakor. Connectivity analysis as a novel approach to motor decoding for prosthesis control. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 20(2):143–152, March 2012.

[17] J Bhattacharya and H Petsche. Universality in the brain while listening to music. *Proceedings. Biological sciences / The Royal Society*, 268(1484):2423–2433, December 2001. PMID: 11747560 PMCID: PMC1088896.

[18] N. Birbaumer, N. Ghanayim, T. Hinterberger, I. Iversen, B. Kotchoubey, A. Kübler, J. Perelmouter, E. Taub, and H. Flor. A Spelling Device for the Paralyzed. *Nature*, 398, 1999.

[19] Tim Blakely, Kai Miller, Jeffrey Ojemann, and Rajesh Rao. Exploring the cortical dynamics of learning by leveraging bci paradigms. In Christoph Guger, Brendan Z. Allison, and Gnter Edlinger, editors, *Brain-Computer Interface Research*, SpringerBriefs in Electrical and Computer Engineering, pages 53–60. Springer Berlin Heidelberg, 2013.

[20] B. Blankertz, R. Tomioka, S. Lemm, M. Kawanabe, and K. R. Muller. Optimizing Spatial filters for Robust EEG Single-Trial Analysis. *Signal Processing Magazine, IEEE*, 25(1):41–56, 2008.

[21] Benjamin Blankertz, Gabriel Curio, and Klaus R. Müller. Classifying Single Trial EEG: Towards Brain Computer Interfacing. In Thomas G. Dietterich, Suzanna Becker, and Zoubin Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, pages 157–164, Cambridge, MA, September 2002. MIT Press.

[22] Benjamin Blankertz, Motoaki Kawanabe Ryota Tomioka, Friederike U. Hohlefeld, Vadim Nikulin, and Klaus-robert Müller. Invariant common spatial patterns: Alleviating nonstationarities in brain-computer interfacing. In *In Ad. in NIPS 20*, volume 20, 2008.

[23] Allan Bluman. *Elementary Statistics: A Step By Step Approach with*

Data CD and Formula Card. McGraw-Hill Science/Engineering/Math, 8 edition edition, January 2011.

[24] Alan A. Boulton, Glen B. Baker, and Case H. Vanderwolf. *Neurophysiological Techniques, II: Applications to Neural Systems*. Humana Press, September 1990.

[25] Jose M. Carmena, Mikhail A. Lebedev, Roy E. Crist, Joseph E. O'Doherty, David M. Santucci, Dragan F. Dimitrov, Parag G. Patil, Craig S. Henriquez, and Miguel A. L. Nicolelis. Learning to Control a BrainMachine Interface for Reaching and Grasping by Primates. *PLoS Biol*, 1(2):e42+, October 2003.

[26] Jose M. Carmena, Mikhail A. Lebedev, Craig S. Henriquez, and Miguel A. L. Nicolelis. Stable Ensemble Performance with Single-Neuron Variability during Reaching Movements in Primates. *The Journal of Neuroscience*, 25(46):10712–10716, November 2005.

[27] Benedetta Cesqui, Peppino Tropea, Silvestro Micera, and Hermano I. Krebs. EMG-based pattern recognition approach in post stroke robot-aided rehabilitation: a feasibility study. *Journal of NeuroEngineering and Rehabilitation*, 10(1):75, July 2013.

[28] J. K. Chapin, K. A. Moxon, R. S. Markowitz, and M. A. Nicolelis. Real-time control of a robot arm using simultaneously recorded neurons in the motor cortex. *Nature neuroscience*, 2(7):664–670, July 1999.

[29] Steven M. Chase, Robert E. Kass, and Andrew B. Schwartz. Behavioral and neural correlates of visuomotor adaptation observed through a brain-computer interface in primary motor cortex. *Journal of Neurophysiology*, 108(2):624–644, 2012.

[30] Steven M. Chase, Andrew B. Schwartz, and Robert E. Kass. Bias, optimal linear estimation, and the differences between open-loop simulation and closed-loop performance of spiking-based braincomputer interface algorithms. *Neural Networks*, 22(9):1203 – 1213, 2009. Brain-Machine Interface.

[31] Samprit Chatterjee, Ali S. Hadi, and Bertram Price. *Regression Analysis by Example, 3rd Edition.* Wiley-Interscience, 3 edition, November 1999.

[32] Febo Cincotti, Laura Kauhanen, Fabio Aloise, Tapio Palomäki, Nicholas Caporusso, Pasi Jylänki, Donatella Mattia, Fabio Babiloni, Gerolf Vanacker, Marnix Nuttin, Maria Grazia Marciani, and José del R. Millán. Vibrotactile feedback for brain-computer interface operation. *Intell. Neuroscience*, 2007:7–7, January 2007.

[33] SamuelT. Clanton, AngusJ.C. McMorland, Zohny Zohny, SMorgan Jeffries, RobertG Rasmussen, SharleneN Flesher, and Meel Velliste. Seven degree of freedom cortical control of a robotic arm. In Christoph Guger, Brendan Z. Allison, and Gnter Edlinger, editors, *Brain-Computer Inter-*

face Research, SpringerBriefs in Electrical and Computer Engineering, pages 73–81. Springer Berlin Heidelberg, 2013.

[34] M Congedo, F Lotte, and A Lcuyer. Classification of movement intention by spatially filtered electromagnetic inverse solutions. *Physics in Medicine and Biology*, 51(8):1971, 2006.

[35] Rongqing Cui, Ross Cunnington, Roland Beisteiner, and Lder Deecke. Effects of forceload on cortical activity preceding voluntary finger movement: Whole-scalp magnetoencephalography of the bereitschaftsfeld. *Neurology, Psychiatry and Brain Research*, 18(3):97 – 104, 2012.

[36] John P. Cunningham, Paul Nuyujukian, Vikash Gilja, Cindy A. Chestek, Stephen I. Ryu, and Krishna V. Shenoy. A closed-loop human simulator for investigating the role of feedback control in brain-machine interfaces. *Journal of Neurophysiology*, 105(4):1932–1949, 2011.

[37] Shalom Darmanjian and Jose C. Principe. Boosted and linked mixtures of hmms for brain-machine interfaces. *EURASIP J. Adv. Sig. Proc.*, 2008, 2008.

[38] Paul R. Davidson, R.D. Jones, J.H. Andreae, and H.R. Sirisena. Simulating closed- and open-loop voluntary movement: a nonlinear control-systems approach. *Biomedical Engineering, IEEE Transactions on*, 49(11):1242–1252, Nov 2002.

[39] DeWayne R. Derryberry, Sue B. Schou, and W. J. Conover. Teaching rank-based tests by emphasizing structural similarities to corresponding parametric tests. *Journal of Statistics Education*, 18(1), January 2010.

[40] Adam S. Dickey, Aaron Suminski, Yali Amit, and Nicholas G. Hatsopoulos. Single-unit stability using chronically implanted multielectrode arrays. *Journal of Neurophysiology*, 102(2):1331–1339, August 2009. PMID: 19535480 PMCID: PMC2724357.

[41] Thomas G. Dietterich and Ghulum Bakiri. Solving multiclass learning problems via error-correcting output codes. *J. Artif. Int. Res.*, 2:263–286, January 1995.

[42] J. DiGiovanna, B. Mahmoudi, J. Fortes, J.C. Principe, and J.C. Sanchez. Coadaptive brain machine interface via reinforcement learning. *Biomedical Engineering, IEEE Transactions on*, 56(1):54–64, Jan 2009.

[43] John P. Donoghue. Bridging the brain to the world: A perspective on neural interface systems. *Neuron*, 60(3):511–521, June 2008. PMID: 18995827.

[44] G. Dornhege, B. Blankertz, and G. Curio. Speeding up classification of multi-channel brain-computer interfaces: common spatial patterns for slow cortical potentials. In *Neural Engineering, 2003. Conference Proceedings. First International IEEE EMBS Conference on*, pages 595–598, March 2003.

[45] G. Dornhege, B. Blankertz, G. Curio, and K.-R. Müller. Boosting bit rates in noninvasive eeg single-trial classifications by feature combination and multiclass paradigms. *Biomedical Engineering, IEEE Transactions on*, 51(6):993 –1002, june 2004.

[46] G. Dornhege, B. Blankertz, M. Krauledat, F. Losch, G. Curio, and K.-R. Müller. Combined optimization of spatial and temporal filters for improving brain-computer interfacing. *Biomedical Engineering, IEEE Transactions on*, 53(11):2274 –2281, nov. 2006.

[47] Michael C. Dorris, Martin Par\'8e, and Douglas P. Munoz. Immediate neural plasticity shapes motor performance. *Journal of Neuroscience*, 20\(1\):RC52\, 2000.

[48] Julien Doyon, Virginia Penhune, and Leslie G Ungerleider. Distinct contribution of the cortico-striatal and cortico-cerebellar systems to motor skill learning. *Neuropsychologia*, 41(3):252–262, 2003.

[49] Robert F Engle and Clive W J Granger. Co-integration and error correction: Representation, estimation, and testing. *Econometrica*, 55(2):251–76, March 1987.

[50] M Falkenstein, J Hoormann, S Christ, and J Hohnsbein. ERP components on reaction errors and their functional significance: a tutorial. *Biological psychology*, 51(2-3):87–107, January 2000.

[51] Geng Fan, Dengwu Ma, Xiaoyan Qu, and Xiaofeng Lv. Multi-scale relevance vector machine classification based on intelligent optimization. In *Systems and Informatics (ICSAI), 2012 International Conference on*, pages 2411–2414, 2012.

[52] L A Farwell and S S Smith. Using brain MERMER testing to detect knowledge despite efforts to conceal. *Journal of forensic sciences*, 46(1):135–143, January 2001. PMID: 11210899.

[53] Pierre W. Ferrez and Jos Del R. Milln. You are wrong!automatic detection of interaction errors from brain waves. In *In Proceedings of the 19th International Joint Conference on Artificial Intelligence*, 2005.

[54] P.W. Ferrez and J. del R.Millan. Error-related eeg potentials generated during simulated brain; computer interaction. *Biomedical Engineering, IEEE Transactions on*, 55(3):923–929, March 2008.

[55] Ronald A. Fisher. The use of multiple measurements in taxonomic problems. *Annals Eugen.*, 7:179–188, 1936.

[56] R.D. Flint, Z.A. Wright, and M.W. Slutzky. Control of a biomimetic brain machine interface with local field potentials: Performance and stability of a static decoder over 200 days. In *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, pages 6719–6722, Aug 2012.

[57] Robert D Flint, Eric W Lindberg, Luke R Jordan, Lee E Miller, and Marc W Slutzky. Accurate decoding of reaching movements from field potentials in the absence of spikes. *Journal of Neural Engineering*, 9(4):046006, 2012.

[58] Robert D Flint, Zachary A Wright, Michael R Scheid, and Marc W Slutzky. Long term, stable brain machine interface performance using local field potentials and multiunit spikes. *Journal of Neural Engineering*, 10(5):056005, 2013.

[59] Guglielmo Foffani, Banu Tutunculer, and Karen A. Moxon. Role of spike timing in the forelimb somatosensory cortex of the rat. *The Journal of Neuroscience*, 24(33):7266–7271, August 2004. PMID: 15317852.

[60] Michael J Frank, Brion S Woroch, and Tim Curran. Error-related negativity predicts reinforcement learning and conflict biases. *Neuron*, 47(4):495–501, August 2005.

[61] Walter J Freeman, Linda J Rogers, Mark D Holmes, and Daniel L Silbergeld. Spatial spectral analysis of human electrocorticograms including the alpha and gamma bands. *Journal of Neuroscience Methods*, 95(2):111 – 121, 2000.

[62] Jonathan Fritz, Shihab Shamma, Mounya Elhilali, and David Klein. Rapid task-related plasticity of spectrotemporal receptive fields in primary auditory cortex. *Nature Neuroscience*, 6(11):1216–1223, November 2003.

[63] Gregory J Gage, Kip A Ludwig, Kevin J Otto, Edward L Ionides, and Daryl R Kipke. Nave coadaptive cortical control. *Journal of Neural Engineering*, 2(2):52, 2005.

[64] Karunesh Ganguly and Jose M. Carmena. Emergence of a stable cortical map for neuroprosthetic control. *PLoS Biol*, 7(7):e1000153, 07 2009.

[65] Stefanos Georgiadis, Perttu Ranta-aho, Mika Tarvainen, and Pasi Karjalaine. State-space modeling for single-trial evoked potential estimation. In Domenico Campolo, editor, *New Developments in Biomedical Engineering*. InTech, January 2010.

[66] AP Georgopoulos, JF Kalaska, R Caminiti, and JT Massey. On the relations between the direction of two-dimensional arm movements and cell discharge in primate motor cortex. *J. Neurosci.*, 2(11):1527–1537, 1982.

[67] AP Georgopoulos, AB Schwartz, and RE Kettner. Neuronal population coding of movement direction. *Science*, 233:1416–1419, 1986.

[68] A.D. Gerson, L.C. Parra, and P. Sajda. Cortically coupled computer vision for rapid image search. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 14(2):174–179, June 2006.

[69] Vikash Gilja, Paul Nuyujukian, Cindy A. Chestek, John P. Cunningham, Byron M. Yu, Joline M. Fan, Mark M. Churchland, Matthew T. Kaufman, Jonathan C. Kao, Stephen I. Ryu, and Krishna V. Shenoy. A

high-performance neural prosthesis enabled by control algorithm design. *Nature Neuroscience*, 15(12):1752–1757, December 2012.

[70] F. Goksu, N.F. Ince, and A.H. Tewfik. Sparse common spatial patterns in brain computer interface applications. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pages 533–536, May 2011.

[71] Fikri Goksu, Nuri Firat Ince, Vijay Aditya Tadipatri, and Ahmed H. Tewfik. Classification of eeg with structural feature dictionaries in a brain computer interface. In *Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE*, pages 1001 –1004, aug. 2008.

[72] G. Gomez-Herrero, W. De Clercq, H. Anwar, O. Kara, K. Egiazarian, S. Van Huffel, and W. Van Paesschen. Automatic removal of ocular artifacts in the eeg without an eog reference channel. In *Signal Processing Symposium, 2006. NORSIG 2006. Proceedings of the 7th Nordic*, pages 130–133, June 2006.

[73] B. Gowreesunker, A. Tewfik, Vijay Aditya Tadipatri, J. Ashe, G. Pellizzer, and R. Gupta. A subspace approach to learning recurrent features from brain activity. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, PP(99):1, 2011.

[74] B Gowreesunker, AH Tewfik, Vijay Aditya Tadipatri, NF Ince, J Ashe,

and G Pellizzer. Overcoming measurement time variability in brain machine interface. *Conf Proc IEEE Eng Med Biol Soc*, 1:3134–7, 2009.

[75] Arnulf B A Graf, Adam Kohn, Mehrdad Jazayeri, and J Anthony Movshon. Decoding the activity of neuronal populations in macaque primary visual cortex. *Nature neuroscience*, 14(2):239–245, February 2011. PMID: 21217762 PMCID: PMC3081541.

[76] Aysegul Gunduz, Justin C. Sanchez, Paul R. Carney, and Jose C. Principe. Mapping broadband electrocorticographic recordings to two-dimensional hand trajectories in humans: Motor control features. *Neural Networks*, 22(9):1257–1270, 2009.

[77] R. Gupta and J. Ashe. Offline decoding of end-point forces using neural ensembles: Application to a brain; machine interface. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 17(3):254–262, June 2009.

[78] Tayfun Gürel and Carsten Mehring. Unsupervised adaptation of brain machine interface decoders. *CoRR*, abs/1206.3666, 2012.

[79] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The weka data mining software: an update. *SIGKDD Explor. Newsl.*, 11(1):10–18, 2009.

[80] James J Heckman. Sample selection bias as a specification error. *Econometrica*, 47(1):153–61, January 1979.

[81] Dustin Allen Heldman. *Epidural Electrocorticography and Intra-cortical Local Field Potentials in Motor Cortex During Volitional Arm Movements and Their Applications to Neural Prosthetic Control*. Washington University in St. Louis, 2007.

[82] R. Heliot, K. Ganguly, J. Jimenez, and J.M. Carmena. Learning in closed-loop brain; machine interfaces: Modeling and experimental validation. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 40(5):1387–1397, Oct 2010.

[83] Leigh R. Hochberg, Mijail D. Serruya, Gerhard M. Friehs, Jon A. Mukand, Maryam Saleh, Abraham H. Caplan, Almut Branner, David Chen, Richard D. Penn, and John P. Donoghue. Neuronal ensemble control of prosthetic devices by a human with tetraplegia. *Nature*, 442(7099):164–171, July 2006.

[84] Clay B. Holroyd and G. H. The neural basis of human error processing: Reinforcement learning, dopamine, and the error-related negativity. *Psychological Review*, 109(4):679–709, 2002.

[85] M.L. Homer, J.A. Perge, M.J. Black, M.T. Harrison, S.S. Cash, and L.R. Hochberg. Adaptive offset correction for intracortical brain; computer interfaces. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 22(2):239–248, March 2014.

[86] Peter R. HUTTENLOCHER. *NEURAL PLASTICITY*. Harvard University Press, June 2009.

[87] Eun Jung Hwang and Richard A Andersen. The utility of multichannel local field potentials for brainmachine interfaces. *Journal of Neural Engineering*, 10(4):046005, 2013.

[88] N.F. Ince, Vijay Aditya Tadipatri, Fikri Goksu, and Ahmed H. Tewfik. Denoising of multiscale/multiresolution structural feature dictionaries for rapid training of a brain computer interface. In *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*, pages 21 –24, sept. 2009.

[89] Nuri F. Ince, Sami Arica, and Ahmed Tewfik. Classification of single trial motor imagery EEG recordings with subject adapted non-dyadic arbitrary timefrequency tilings. *J. Neural Eng.*, 3(3):235+, September 2006.

[90] Nuri F. Ince, Sami Arica, and Ahmed Tewfik. Motor imagery based brain computer interface with subject adapted time-frequency tiling. In *14th European Signal Processing Conference*, 2006.

[91] Nuri F. Ince, R. Gupta, S. Arica, A.H. Tewfik, J. Ashe, and G. Pellizzer. Movement direction decoding with spatial patterns of local field potentials. In *Neural Engineering, 2009. NER '09. 4th International IEEE/EMBS Conference on*, pages 291 –294, May 2009.

[92] Nuri F. Ince, A.H. Tewfik, and S. Arica. A space-time-frequency analysis approach for the classification motor imagery eeg recordings in a

brain computer interface task. In *Engineering in Medicine and Biology Society, 2006. EMBS '06. 28th Annual International Conference of the IEEE*, pages 2581 –2584, 30 2006-sept. 3 2006.

[93] Nuri F. Ince, A.H. Tewfik, and S. Arica. Extraction subject-specific motor imagery time-frequency patterns for single trial eeg classification. *Computers in Biology and Medicine*, 37(4):499 – 508, 2007. Wavelet-based Algorithms for Medical Problems.

[94] William James. *The Principles of Psychology, Vol. 1.* Dover Publications, New York, reprint edition edition, June 1950.

[95] Beata Jarosiewicz, Nicolas Y Masse, Daniel Bacher, Sydney S Cash, Emad Eskandar, Gerhard Friehs, John P Donoghue, and Leigh R Hochberg. Advantages of closed-loop calibration in intracortical braincomputer interfaces for people with tetraplegia. *Journal of Neural Engineering*, 10(4):046012, 2013.

[96] Henrik Jeldtoft Jensen. *Self-organized criticality: emergent complex behavior in physical and biological systems.* Cambridge University Press, 1998.

[97] Winnie Jensen and Patrick J. Rousche. On variability and use of rat primary motor cortex responses in behavioral task discrimination. *Journal of Neural Engineering*, 3(1):L7+, March 2006.

[98] J. Kalcher, D. Flotzinger, Ch. Neuper, S. Glly, and G. Pfurtscheller. Graz brain-computer interface ii: towards communication between humans and computers based on online classification of three different eeg patterns. *Medical and Biological Engineering and Computing*, 34:382–388, 1996. 10.1007/BF02520010.

[99] R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Fluids Engineering*, 82(1):35–45, March 1960.

[100] Steffen Katzner, Ian Nauhaus, Andrea Benucci, Vincent Bonin, Dario L Ringach, and Matteo Carandini. Local origin of field potentials in visual cortex. *Neuron*, 61(1):35–41, January 2009. PMID: 19146811 PMCID: PMC2730490.

[101] P. R. Kennedy, R. A. E. Bakay, M. M. Moore, K. Adams, and J. Goldwaithe. Direct control of a computer from the human central nervous system. *IEEE Transactions on Rehabilitation Engineering*, 8(2):198–202, June 2000.

[102] Jeffrey A Kleim and Theresa A Jones. Principles of experience-dependent neural plasticity: implications for rehabilitation after brain damage. *Journal of speech, language, and hearing research: JSLHR*, 51(1):S225–239, February 2008. PMID: 18230848.

[103] Lubomir Kostal, Petr Lansky, and Jean-Pierre Rospars. Review article: Neuronal coding and spiking randomness. *European Journal of Neuroscience*, 26(10):2693–2701, 2007.

[104] Balaji Krishnapuram, Alexander J. Hartemink, Lawrence Carin, and Mrio A. T. Figueiredo. A bayesian approach to joint feature selection and classifier design. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 26:1105–1111, 2004.

[105] Yoomi Kwon and Sung-Phil Kim. A closed-loop brain-machine interface simulator based on computer mouse control. In *Computer-Aided Control System Design (CACSD), 2010 IEEE International Symposium on*, pages 2262–2266, Sept 2010.

[106] Andrea Kbler, Boris Kotchoubey, Jochen Kaiser, Jonathan R. Wolpaw, and Niels Birbaumer. Brain-computer communication: unlocking the locked in. *Psychological Bulletin.*, 127:358–375, 2001.

[107] Felix Lee, Reinhold Scherer, Robert Leeb, Christa Neupe, Horst Bischof, and Gert Pfurtscheller. A comparative analysis of multi-class eeg classification for brain computer interface, 2005.

[108] Alan D. Legatt, Joseph Arezzo, and Herbert G. Vaughan Jr. Averaged multiple unit activity as an estimate of phasic changes in local neuronal activity: effects of volume-conducted potentials. *Journal of Neuroscience Methods*, 2(2):203 – 217, 1980.

[109] S. Lemm, B. Blankertz, G. Curio, and K. Muller. Spatio-spectral filters for improving the classification of single trial eeg. *Biomedical Engineering, IEEE Transactions on*, 52(9):1541–1548, Sept 2005.

[110] Eric C. Leuthardt, Gerwin Schalk, Jonathan R. Wolpaw, Jeffrey G. Ojemann, and Daniel W. Moran. A braincomputer interface using electrocorticographic signals in humans. *Journal of Neural Engineering*, 1(2):63–71, June 2004.

[111] S. P. Levine, J. E. Huggins, S. L. Bement, R. K. Kushwaha, L. A. Schuh, M. M. Rohde, E. A. Passaro, D. A. Ross, K. V. Elisevich, and B. J. Smith. A direct brain interface based on event-related potentials. *Rehabilitation Engineering, IEEE Transactions on*, 8(2):180–185, 2000.

[112] Zheng Li, Joseph E. O'Doherty, Mikhail A. Lebedev, and Miguel A. L. Nicolelis. Adaptive decoding for brain-machine interfaces through bayesian parameter updates. *Neural Comput.*, 23(12):3162–3204, December 2011.

[113] Henrik Lindn, Tom Tetzlaff, Tobias C. Potjans, Klas H. Pettersen, Sonja Grn, Markus Diesmann, and Gaute T. Einevoll. Modeling the spatial reach of the LFP. *Neuron*, 72(5):859–872, August 2011. PMID: 22153380.

[114] Nikos K. Logothetis. The underpinnings of the BOLD functional magnetic resonance imaging signal. *The Journal of Neuroscience*, 23(10):3963–3971, May 2003. PMID: 12764080.

[115] F. Lotte and Cuntai Guan. Regularizing common spatial patterns to improve bci designs: Unified theory and new algorithms. *Biomedical Engineering, IEEE Transactions on*, 58(2):355–362, Feb 2011.

[116] F. Lotte, A. Lecuyer, and B. Arnaldi. Furia: An inverse solution based feature extraction algorithm using fuzzy set theory for brain computer interfaces. *Signal Processing, IEEE Transactions on*, 57(8):3253–3263, Aug 2009.

[117] Steven J. Luck. *An Introduction to the Event-Related Potential Technique*. A Bradford Book, Cambridge, Mass, 1 edition edition, August 2005.

[118] B. Mahmoudi, J.C. Principe, and J.C. Sanchez. Symbiotic brain-machine interface decoding using simultaneous motor and reward neural representation. In *Neural Engineering (NER), 2011 5th International IEEE/EMBS Conference on*, pages 597–600, April 2011.

[119] W.Q. Malik, W. Truccolo, E.N. Brown, and L.R. Hochberg. Efficient decoding with steady-state kalman filter in neural interface systems. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 19(1):25–34, Feb 2011.

[120] Yael Mandelblat-Cerf, Itai Novick, Rony Paz, Yuval Link, Sharon Freeman, and Eilon Vaadia. The neuronal basis of long-term sensorimotor learning. *The Journal of Neuroscience*, 31(1):300–313, 2011.

[121] David A Markowitz, Yan T Wong, Charles M Gray, and Bijan Pesaran. Optimizing the decoding of movement goals from local field potentials in macaque cortex. *The Journal of neuroscience: the official journal of the*

Society for Neuroscience, 31(50):18412–18422, December 2011. PMID: 22171043 PMCID: PMC3315593.

[122] D. J. Mcfarland, A. T. Lefkowicz, and J. R. Wolpaw. Design and operation of an EEG-based brain-computer interface with digital signal processing technology. *Behav. Res. Methods Instrum. Comput.*, 29:337–345, 1997.

[123] Dennis J. McFarland, Dean J. Krusienski, and Jonathan R. Wolpaw. Braincomputer interface signal processing at the wadsworth center: mu and sensorimotor beta rhythms. In Christa Neuper and Wolfgang Klimesch, editors, *Event-Related Dynamics of Brain Oscillations*, volume 159 of *Progress in Brain Research*, pages 411 – 419. Elsevier, 2006.

[124] D.J. McFarland and J.R. Wolpaw. Sensorimotor rhythm-based brain-computer interface (bci): feature selection by regression improves performance. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 13(3):372–379, Sept 2005.

[125] Sam McKenzie, Nick T. M. Robinson, Lauren Herrera, Jordana C. Churchill, and Howard Eichenbaum. Learning causes reorganization of neuronal firing patterns to represent related experiences within a hippocampal schema. *The Journal of Neuroscience*, 33(25):10243–10256, June 2013. PMID: 23785140.

[126] Donald L. Schomer MD and Fernando Lopes da Silva MD PhD. *Niedermeyer's Electroencephalography: Basic Principles, Clinical Applications,*

and Related Fields. LWW, Philadelphia, sixth edition edition, December 2010.

[127] Attila Medl, Doris Flotzinger, and G. Pfurtscheller. Hilbert-transform based predictions of hand movements from eeg measurements. In *Engineering in Medicine and Biology Society, 1992 14th Annual International Conference of the IEEE*, volume 6, pages 2539–2540, Oct 1992.

[128] Carsten Mehring, Jörn Rickert, Eilon Vaadia, Simone Cardosa de Oliveira, Ad Aertsen, and Stefan Rotter. Inference of hand movements from local field potentials in monkey motor cortex. *Nature neuroscience*, 6(12):1253–1254, December 2003.

[129] Tomislav Milekovic, Jrg Fischer, Tobias Pistohl, Johanna Ruescher, Andreas Schulze-Bonhage, Ad Aertsen, Jrn Rickert, Tonio Ball, and Carsten Mehring. An online brain-machine interface using decoding of movement direction from the human electrocorticogram. *Journal of neural engineering*, 9(4):046003, August 2012. PMID: 22713666.

[130] Noboru Murata, Motoaki Kawanabe, Andreas Ziehe, Klaus-Robert Müller, and Shun-ichi Amari. On-line learning in changing environments with applications in supervised and unsupervised learning. *Neural Netw.*, 15:743–760, June 2002.

[131] Xavier Navarro, Thilo B. Krueger, Natalia Lago, Silvestro Micera, Thomas Stieglitz, and Paolo Dario. A critical review of interfaces with the peripheral nervous system for the control of neuroprostheses and hybrid

bionic systems. *Journal of the Peripheral Nervous System*, 10(3):229–258, 2005.

[132] C. Neuper and G. Pfurtscheller. Event-related dynamics of cortical rhythms: frequency-specific features and functional correlates. *Int J Psychophysiol*, 43(1):41–58, 2001.

[133] Miguel A L Nicolelis. Brain-machine interfaces to restore motor function and probe neural circuits. *Nature Reviews Neuroscience*, 4(5):417–422, 2003.

[134] F. Nijboer, E.W. Sellers, J. Mellinger, M.A. Jordan, T. Matuz, A. Furdea, S. Halder, U. Mochty, D.J. Krusienski, T.M. Vaughan, J.R. Wolpaw, N. Birbaumer, and A. Kbler. A p300-based braincomputer interface for people with amyotrophic lateral sclerosis. *Clinical Neurophysiology*, 119(8):1909 – 1916, 2008.

[135] A. Nijholt and D. Tan. Brain-computer interfacing for intelligent systems. *Intelligent Systems, IEEE*, 23(3):72–79, 2008.

[136] N. Nikolaev and P. Tino. Sequential relevance vector machine learning from time series. In *Neural Networks, 2005. IJCNN '05. Proceedings. 2005 IEEE International Joint Conference on*, volume 2, pages 1308–1313 vol. 2, 2005.

[137] E. Novikov, A. Novikov, D. Shannahoff-Khalsa, B. Schwartz, and J. Wright. Scale-similar activity in the brain. *Phys. Rev. E*, 56:R2387–R2389, Sep

1997.

[138] John G. O'Leary and Nicholas G. Hatsopoulos. Early Visuomotor Representations Revealed From Evoked Local Field Potentials in Motor and Premotor Cortical Areas. *J Neurophysiol*, 96(3):1492–1506, September 2006.

[139] Bruno A. Olshausen and David J. Field. Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision Research*, 37(23):3311 – 3325, 1997.

[140] Bence P. Olveczky, Timothy M. Otchy, Jesse H. Goldberg, Dmitriy Aronov, and Michale S. Fee. Changes in the neural control of a complex motor sequence during learning. *Journal of Neurophysiology*, 106(1):386–397, jul 2011.

[141] A.L. Orsborn, S. Dangi, H.G. Moorman, and J.M. Carmena. Closed-loop decoder adaptation on intermediate time-scales facilitates rapid bmi performance improvements independent of decoder initialization conditions. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 20(4):468–477, 2012.

[142] A.L. Orsborn, S. Dangi, H.G. Moorman, and J.M. Carmena. Closed-loop decoder adaptation on intermediate time-scales facilitates rapid bmi performance improvements independent of decoder initialization conditions. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 20(4):468–477, July 2012.

[143] Alvaro Pascual-Leone, Catarina Freitas, Lindsay Oberman, JaredC. Horvath, Mark Halko, Mark Eldaief, Shahid Bashir, Marine Vernet, Mouhshin Shafi, Brandon Westover, AndrewM. Vahabzadeh-Hagh, and Alexander Rotenberg. Characterizing brain cortical plasticity and network dynamics across the age-span in health and disease with tms-eeg and tms-fmri. *Brain Topography*, 24(3-4):302–315, 2011.

[144] P. G. Patil, J. M. Carmena, M. A. Nicolelis, and D. A. Turner. Ensemble recordings of human subcortical neurons as a source of motor control signals for a brain-machine interface. *Neurosurgery*, 55(1), July 2004.

[145] Klas H. Pettersen and Gaute T. Einevoll. Amplitude variability and extracellular low-pass filtering of neuronal spikes. *Biophysical Journal*, 94(3):784–802, January 2008. PMID: 17921225.

[146] G. Pfurtscheller and C. Neuper. Motor imagery and direct brain-computer communication. *Proceedings of the IEEE*, 89(7):1123–1134, July 2001.

[147] G. Pfurtscheller, A. Stanck Jr, and C. Neuper. Post-movement beta synchronization. a correlate of an idling motor area? *Electroencephalogr Clin Neurophysiol*, 98(4):281–93, 1996.

[148] Gert Pfurtscheller and et al. Functional meaning of event-related desynchronization (erd) and synchronization (ers). *Handbook of EEG and Clinical Neurophysiology.,rev ed*, 6:51–65, 1999.

[149] F. Piccione, F. Giorgi, P. Tonin, K. Priftis, S. Giove, S. Silvoni, G. Palmas, and F. Beverina. P300-based brain computer interface: reliability and performance in healthy and paralysed participants. *Clinical neurophysiology*, 117(3):531–537, March 2006.

[150] John R. Pierce. *An Introduction to Information Theory.* Dover Publications, subsequent edition, November 1980.

[151] Tobias Pistohl, Tonio Ball, Andreas Schulze-Bonhage, Ad Aertsen, and Carsten Mehring. Prediction of arm movement trajectories from ECoG-recordings in humans. *Journal of Neuroscience Methods*, 167(1):105–114, January 2008.

[152] Eric A. Pohlmeyer, Babak Mahmoudi, Shijia Geng, Noeline W. Prins, and Justin C. Sanchez. Using reinforcement learning to provide stable brain-machine interface control despite neural input reorganization. *PLoS ONE*, 9(1):e87253, 01 2014.

[153] V. Polikov, P. Tresco, and W. Reichert. Response of brain tissue to chronically implanted neural electrodes. *Journal of Neuroscience Methods*, 148(1):1–18, October 2005.

[154] M. Pregenzer and G. Pfurtscheller. Frequency component selection for an EEG-based brain to computer interface. *Rehabilitation Engineering, IEEE Transactions on*, 7(4):413 –419, December 1999.

[155] W S Pritchard. The brain in fractal time: 1/f-like power spectrum scaling of the human electroencephalogram. *The International journal of neuroscience*, 66(1-2):119–129, September 1992. PMID: 1304564.

[156] Joaquin Quionero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D. Lawrence. *Dataset Shift in Machine Learning*. The MIT Press, 2009.

[157] H. Ramoser, J. Muller-Gerking, and G. Pfurtscheller. Optimal spatial filtering of single trial EEG during imagined hand movement. *IEEE Transactions on Rehabilitation Engineering*, 8(4):441–446, December 2000.

[158] Malte Rasch, Nikos K. Logthetis, and Gabriel Kreiman. From neurons to circuits: linear estimation of local field potentials. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 29(44):13785–13796, November 2009. PMID: 19889990 PMCID: PMC2924964.

[159] Brice Rebsamen, Chee Leong Teo, Qiang Zeng, VMH Ang, Etienne Burdet, Cuntai Guan, Haihong Zhang, and Christian Laugier. Controlling a wheelchair indoors using thought. *Intelligent Systems, IEEE*, 22(2):18–24, 2007.

[160] B. Reuderink and M. Poel. Robustness of the common spatial patterns algorithm in the bci-pipeline, July 2008.

[161] Jörn Rickert, Simone Cardoso de C. Oliveira, Eilon Vaadia, Ad Aertsen, Stefan Rotter, and Carsten Mehring. Encoding of movement direction

in different frequency ranges of motor cortical local field potentials. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 25(39):8815–8824, September 2005.

[162] P. J. Rousche and R. A. Normann. Chronic recording capability of the Utah Intracortical Electrode Array in cat sensory cortex. *J Neurosci Methods*, 82(1):1–15, July 1998.

[163] Laura M Rowland, Reza Shadmehr, Dwight Kravitz, and Henry H Holcomb. Sequential neural changes during motor learning in schizophrenia. *Psychiatry research*, 163(1):1–12, May 2008. PMID: 18407471 PMCID: PMC2562703.

[164] Emilio Salinas and L. F. Abbott. Vector reconstruction from firing rates. *Journal of Computational Neuroscience*, 1:89–107, 1994.

[165] Justin C. Sanchez and Jose C. Principe. Brainmachine interface engineering. *Synthesis Lectures on Biomedical Engineering*, 2(1):1–234, January 2007.

[166] Cristina Savin, Prashant Joshi, and Jochen Triesch. Independent component analysis in spiking neurons. *PLoS Comput Biol*, 6(4):e1000757, 04 2010.

[167] L P Savtchenko, P Gogan, S M Korogod, and S Tyc-Dumont. Imaging stochastic spatial variability of active channel clusters during excitation

of single neurons. *Neuroscience research*, 39(4):431–446, April 2001. PMID: 11274742.

[168] S Schaffelhofer and H Scherberger. A new method of accurate hand- and arm-tracking for small primates. *Journal of Neural Engineering*, 9(2):026025, 2012.

[169] G. Schalk, J. Kubánek, K. J. Miller, N. R. Anderson, E. C. Leuthardt, J. G. Ojemann, D. Limbrick, D. Moran, L. A. Gerhardt, and J. R. Wolpaw. Decoding two-dimensional movement trajectories using electrocorticographic signals in humans. *J. Neural Eng.*, 4(3):264+, September 2007.

[170] Gerwin Schalk, Jonathan R. Wolpaw, Dennis J. McFarland, and Gert Pfurtscheller. EEG-based communication: presence of an error potential. *Clinical Neurophysiology*, 111(12):2138–2144, December 2000.

[171] Hansjörg Scherberger, Murray R. Jarvis, and Richard A. Andersen. Cortical local field potential encodes movement intentions in the posterior parietal cortex. *Neuron*, 46(2):347–354, April 2005.

[172] A. Schlögl, D. Flotzinger, and G. Pfurtscheller. Adaptive autoregressive modeling used for single-trial EEG classification. *Biomedizinische Technik. Biomedical engineering*, 42(6):162–167, June 1997.

[173] Bernhard Scholkopf and Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*.

MIT Press, Cambridge, MA, USA, 2001.

[174] Michael Schröder, Thomas Navin Lal, Thilo Hinterberger, Martin Bogdan, N. Jeremy Hill, Niels Birbaumer, Wolfgang Rosenstiel, and Bernhard Schölkopf. Robust eeg channel selection across subjects for brain-computer interfaces. *EURASIP J. Appl. Signal Process.*, 2005:3103–3112, January 2005.

[175] A. Schwartz, X. Cui, D. Weber, and D. Moran. Brain-Controlled Interfaces: Movement Restoration with Neural Prosthetics. *Neuron*, 52(1):205–220, October 2006.

[176] Mijail D. Serruya, Nicholas G. Hatsopoulos, Liam Paninski, Matthew R. Fellows, and John P. Donoghue. Brain-machine interface: Instant neural control of a movement signal. *Nature*, 416(6877):141–142, March 2002.

[177] Maryam M Shanechi, Rollin C Hu, Marissa Powers, Gregory W Wornell, Emery N Brown, and Ziv M Williams. Neural population partitioning and a concurrent brain-machine interface for sequential motor function. *Nature neuroscience*, 15(12):1715–1722, December 2012. PMID: 23143511 PMCID: PMC3509235.

[178] C. E. Shannon. A mathematical theory of communication. *Bell system technical journal*, 27, 1948.

[179] Krishna V. Shenoy. Neural prosthetic control signals from plan activity. *Neuroreport*, 14(4):591–596, March 2003.

[180] Pradeep Shenoy and Rajesh P. N. Rao. Dynamic Bayesian Networks for Brain-Computer Interfaces. In Lawrence K. Saul, Yair Weiss, and léon Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 1265–1272, Cambridge, MA, 2005. MIT Press.

[181] Kentaro Shimoda, Yasuo Nagasaka, Zenas C Chao, and Naotaka Fujii. Decoding continuous three-dimensional hand trajectories from epidural electrocorticographic signals in japanese macaques. *Journal of neural engineering*, 9(3):036015, June 2012. PMID: 22627008.

[182] Ng Siew Cheok and P. Raveendran. Removal of eog artifacts using ica regression method. In NoorAzuan Abu Osman, Fatimah Ibrahim, WanAbuBakar Wan Abas, HermanShah Abdul Rahman, and Hua-Nong Ting, editors, *4th Kuala Lumpur International Conference on Biomedical Engineering 2008*, volume 21 of *IFMBE Proceedings*, pages 226–229. Springer Berlin Heidelberg, 2008.

[183] J. D. Simeral, S.-P. Kim, M. J. Black, J. P. Donoghue, and L. R. Hochberg. Neural control of cursor trajectory and click by a human with tetraplegia 1000 days after implant of an intracortical microelectrode array. *Journal of Neural Engineering*, 8(2):025027, April 2011.

[184] Dylan Alexander Simon and Nathaniel D. Daw. Neural correlates of forward planning in a spatial decision task in humans. *The Journal of Neuroscience*, 31(14):5526–5539, April 2011. PMID: 21471389.

[185] W Singer and C M Gray. Visual feature integration and the temporal correlation hypothesis. *Annual Review of Neuroscience*, 18(1):555–586, 1995. PMID: 7605074.

[186] Kelvin So. *Changes in Neural Activity during Brain-Machine Interface Control: from Information Encoding and Connectivity to Local Field Potentials*. PhD thesis, EECS Department, University of California, Berkeley, Dec 2013.

[187] M Stephane, NF Ince, M Kuskowski, A Leuthold, AH Tewfik, K Nelson, K McClannahan, CR Fletcher, and VA Tadipatri. Neural oscillations associated with the primacy and recency effects of verbal working memory. *Neurosci Lett*, 473(3):172–7, 2010.

[188] Nicholas Stergiou and Leslie M. Decker. Human movement variability, nonlinear dynamics, and pathology: Is there a connection? *Human Movement Science*, 30(5):869 – 888, 2011. {EWOMS} 2009: The European Workshop on Movement Science.

[189] Aaron J. Suminski, Dennis C. Tkach, Andrew H. Fagg, and Nicholas G. Hatsopoulos. Incorporating feedback from multiple sensory modalities enhances BrainMachine interface control. *The Journal of Neuroscience*, 30(50):16777–16787, December 2010. PMID: 21159949.

[190] S. Suner, M.R. Fellows, C. Vargas-Irwin, G.K. Nakata, and J.P. Donoghue. Reliability of signals from a chronically implanted, silicon-based elec-

trode array in non-human primate primary motor cortex. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 13(4):524–541, 2005.

[191] V.A. Tadipatri, A.H. Tewfik, J. Ashe, G. Pellizer, and R. Gupta. Using topographical channel distribution to decode movement directions from local field potentials. In *Biomedical Imaging: From Nano to Macro, 2011 IEEE International Symposium on*, pages 1835 –1838, 30 2011-april 2 2011.

[192] V.A. Tadipatri, A.H. Tewfik, J. Ashe, and G. Pellizzer. Robust movement direction decoders from local field potentials using spatio-temporal qualitative patterns. In *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, pages 4623–4626, 2012.

[193] Vijay Aditya Tadipatri, A.H. Tewfik, J. Ashe, G. Pellizzer, and R. Gupta. Movement direction decoding of local field potentials using time-evolving spatial patterns. In *Neural Engineering (NER), 2011 5th International IEEE/EMBS Conference on*, pages 392 –395, 27 2011-may 1 2011.

[194] Vijay Aditya Tadipatri, Ahmed Tewfik, and James Ashe. Long-term movement tracking from local field potentials with an adaptive open-loop decoder. In *ICASSP2014 - Bio Imaging and Signal Processing (ICASSP2014 - BISP)*, pages 5911–5915, Florence, Italy, May 2014.

[195] Vijay Aditya Tadipatri, Ahmed H. Tewfik, Vikrham B. Gowreesunker, J. Ashe, G. Pellizer, and R. Gupta. Time robust movement direction decoding in local field potentials using channel ranking. In *Engineering in Medicine and Biology Magazine*, volume 29. IEEE, 2010.

[196] Dawn M. Taylor, Stephen I. Tillery, and Andrew B. Schwartz. Direct Cortical Control of 3D Neuroprosthetic Devices. *Science*, 296(5574):1829–1832, June 2002.

[197] Stephan F Taylor, Emily R Stern, and William J Gehring. Neural systems for error monitoring: recent findings and theoretical perspectives. *The Neuroscientist: a review journal bringing neurobiology, neurology and psychiatry*, 13(2):160–172, April 2007.

[198] Nitish V. Thakor. Translating the brain-machine interface. *Science Translational Medicine*, 5(210):210ps17, 2013.

[199] M. E. Tipping and A. C. Faul. Fast marginal likelihood maximization for sparse Bayesian models. In C. M. Bishop and B. J. Frey, editors, *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*, 2003.

[200] Michael E. Tipping. Sparse bayesian learning and the relevance vector machine. *J. Mach. Learn. Res.*, 1:211–244, September 2001.

[201] Michael E. Tipping and Anita Faul. Fast marginal likelihood maximisation for sparse bayesian models. In *Proceedings of the Ninth Inter-*

national Workshop on Artificial Intelligence and Statistics, pages 3–6, 2003.

[202] Ryota Tomioka, Kazuyuki Aihara, and Klaus-Robert Müller. Logistic regression for single trial eeg classification. In B. Schölkopf, J.C. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, pages 1377–1384. MIT Press, 2007.

[203] Carmen Vidaurre and Benjamin Blankertz. Towards a cure for BCI illiteracy. *Brain topography*, 23(2):194–198, June 2010.

[204] Stephan Waldert, Hubert Preissl, Evariste Demandt, Christoph Braun, Niels Birbaumer, Ad Aertsen, and Carsten Mehring. Hand movement direction decoded from MEG and EEG. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 28(4):1000–1008, January 2008.

[205] Yijun Wang, Shangkai Gao, and Xiaorong Gao. Common spatial pattern method for channel selelction in motor imagery based brain-computer interface. In *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the*, pages 5392–5395, Jan 2005.

[206] Zhisong Wang, Alexander Maier, David A. Leopold, Nikos K. Logothetis, and Hualou Liang. Single-trial evoked potential estimation using wavelets. *Comput. Biol. Med.*, 37(4):463–473, April 2007.

[207] R Wennberg, F Quesney, A Olivier, and T Rasmussen. Electrocorticography and outcome in frontal lobe epilepsy. *Electroencephalography and Clinical Neurophysiology*, 106(4):357 – 368, 1998.

[208] Johan Wessberg, Christopher R. Stambaugh, Jerald D. Kralik, Pamela D. Beck, Mark Laubach, John K. Chapin, Jung Kim, S. James Biggs, Mandayam A. Srinivasan, and Miguel A. L. Nicolelis. Real-time prediction of hand trajectory by ensembles of cortical neurons in primates. *Nature*, 408(6810):361–365, November 2000.

[209] Jason Weston, Olivier Chapelle, Andr Elisseeff, Bernhard Schlkopf, and Vladimir Vapnik. Kernel dependency estimation. In Suzanna Becker, Sebastian Thrun, and Klaus Obermayer, editors, *NIPS*, pages 873–880. MIT Press, 2002.

[210] Richard L. White. Object classification in astronomical images. In G. Jogesh Babu and Eric D. Feigelson, editors, *Statistical Challenges in Modern Astronomy II*, pages 135–151. Springer New York, January 1997.

[211] Jonathan Wolpaw and Elizabeth Winter Wolpaw. *Brain-Computer Interfaces: Principles and Practice*. Oxford University Press, USA, Oxford ; New York, 1 edition edition, January 2012.

[212] Jonathan R. Wolpaw, Niels Birbaumer, Dennis J. McFarland, Gert Pfurtscheller, and Theresa M. Vaughan. Braincomputer interfaces for communication and control. *Clinical Neurophysiology*, 113(6):767–791, June 2002.

[213] J.R. Wolpaw, N. Birbaumer, W.J. Heetderks, D.J. McFarland, P.H. Peckham, G. Schalk, E. Donchin, L.A. Quatrano, C.J. Robinson, and T.M. Vaughan. Brain-computer interface technology: a review of the first international meeting. *Rehabilitation Engineering, IEEE Transactions on*, 8(2):164 –173, June 2000.

[214] DM Wolpert, Z Ghahramani, and MI Jordan. An internal model for sensorimotor integration. *Science*, 269(5232):1880–1882, 1995.

[215] Wei Wu, Zhe Chen, Shangkai Gao, and E.N. Brown. A probabilistic framework for learning robust common spatial patterns. In *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*, pages 4658–4661, Sept 2009.

[216] Dajun Xing, Chun-I. Yeh, and Robert M. Shapley. Spatial spread of the local field potential and its laminar variation in visual cortex. *The Journal of Neuroscience*, 29(37):11540–11549, September 2009.

[217] Xinyi Yong, R.K. Ward, and G.E. Birch. Sparse spatial filter optimization for EEG channel reduction in brain-computer interface. In *IEEE International Conference on Acoustics, Speech and Signal Processing, 2008. ICASSP 2008*, pages 417 –420, April 2008.

[218] Weidong Zhou and Jean Gotman. Automatic removal of eye movement artifacts from the {EEG} using {ICA} and the dipole model. *Progress in Natural Science*, 19(9):1165 – 1170, 2009.

[219] Liqiang Zhu, Ying-Cheng Lai, Frank C Hoppensteadt, and Jiping He. Characterization of neural interaction during learning and adaptation from spike-train data. *Mathematical biosciences and engineering: MBE*, 2(1):1–23, January 2005. PMID: 20369909.

[220] Jun Zhuang, W. Truccolo, C. Vargas-Irwin, and J.P. Donoghue. Decoding 3-d reach and grasp kinematics from high-frequency local field potentials in primate primary motor cortex. *Biomedical Engineering, IEEE Transactions on*, 57(7):1774 –1784, july 2010.

Vita

Vijay Aditya Tadipatri received his B.E. degree from Osmania University, Hyderabad in 2005 and his M.S degree in Electrical Engineering from the University of Minnesota, at Twin Cities in 2010. From 2008 to 2010, he has been a Research Assistant under Dr. Ahmed H. Tewfik at University of Minnesota at Twin Cities and since 2010 at The University of Texas at Austin.

During the summers of 2010, 2011 and 2012 he interned in the Research Department of Boston Scientific Inc. He worked on "Noise Induced Arrhythmia Detection in Implantable Cardioverter Defibrillator", and holds a US patent: US 8744556 B2: granted on Jun 3, 2014. He also worked on "Identifying responders and non-responders to Cardiac Resynchronization Therapy".

His current research interests include statistical signal processing and machine learning applications to Brain Computer Interfaces.

Permanent address: vijay.tadipatri@utexas.edu

This dissertation was typeset with LaTeX[†] by the author.

[†]LaTeX is a document preparation system developed by Leslie Lamport as a special version of Donald Knuth's TeX Program.