IMPROVING THE GENERALISABILITY OF BRAIN COMPUTER INTERFACE APPLICATIONS VIA MACHINE LEARNING AND SEARCH-BASED HEURISTICS

JASON ADAIR

Doctor of Philosophy

Institute of Computing Science and Mathematics

University of Stirling

March 2018

DECLARATION

I hereby declare that this thesis has been composed by me, that the work and results have not been presented for any university degree prior to this, and that the ideas that I do not attribute to others are my own

Stirling, March 2018

Jason Adair

ABSTRACT

Brain Computer Interfaces (BCI) are a domain of hardware/software in which a user can interact with a machine without the need for motor activity, communicating instead via signals generated by the nervous system. These interfaces provide life-altering benefits to users, and refinement will both allow their application to a much wider variety of disabilities, and increase their practicality. The primary method of acquiring these signals is *Electroencephalography (EEG)*. This technique is susceptible to a variety of different sources of noise, which compounds the inherent problems in BCI training data: large dimensionality, low numbers of samples, and non-stationarity between users and recording sessions. *Feature Selection* and *Transfer Learning* have been used to overcome these problems, but they fail to account for several characteristics of BCI. This thesis extends both of these approaches by the use of Search-based algorithms.

Feature Selection techniques, known as *Wrappers* use 'black box' evaluation of feature subsets, leading to higher classification accuracies than ranking methods known as *Filters*. However, Wrappers are more computationally expensive, and are prone to over-fitting to training data. In this thesis, we applied *Iterated Local Search (ILS)* to the BCI field for the first time in literature, and demonstrated competitive results with state-of-the-art methods such as Least Absolute Shrinkage and Selection Operator and Genetic Algorithms. We then developed ILS variants with guided perturbation operators. *Linkage* was used to develop a multivariate metric, *Intrasolution Linkage*. This takes into account pair-wise dependencies of features with the label, in the context of the solution. Intrasolution Linkage was then integrated into two ILS variants. The Intrasolution Linkage Score was discovered to have a stronger correlation with the solutions predictive accuracy on unseen data than *Cross Validation Error (CVE)* on the training set, the typical approach to feature subset evaluation.

Mutual Information was used to create *Minimum Redundancy Maximum Relevance Iterated Local Search (MRMR-ILS)*. In this algorithm, the perturbation

operator was guided using an existing Mutual Information measure, and compared with current Filter and Wrapper methods. It was found to achieve generally lower CVE rates and higher predictive accuracy on unseen data than existing algorithms. It was also noted that solutions found by the MRMR-ILS provided CVE rates that had a stronger correlation with the accuracy on unseen data than solutions found by other algorithms. We suggest that this may be due to the guided perturbation leading to solutions that are richer in Mutual Information.

Feature Selection reduces computational demands and can increase the accuracy of our desired models, as evidenced in this thesis. However, limited quantities of training samples restricts these models, and greatly reduces their generalisability. For this reason, utilisation of data from a wide range of users is an ideal solution. Due to the differences in neural structures between users, creating adequate models is difficult. We adopted an existing state-of-the-art ensemble technique *Ensemble Learning Generic Information (ELGI)*, and developed an initial optimisation phase. This involved using search to transplant instances between user subsets to increase the generalisability of each subset, before combination in the ELGI. We termed this *Evolved Ensemble Learning Generic Information (eELGI)*. The eELGI achieved higher accuracy than user-specific BCI models, across all eight users. Optimisation of the training dataset allowed smaller training sets to be used, offered protection against neural drift, and created models that performed similarly across participants, regardless of neural impairment.

Through the introduction and hybridisation of search based algorithms to several problems in BCI we have been able to show improvements in modelling accuracy and efficiency. Ultimately, this represents a step towards more practical BCI systems that will provide life altering benefits to users.

ACKNOWLEDGMENTS

This journey would never have been possible without the love and support of my parents, Jean and Bill, and it is a journey I would never have dared undergo, without my partner, Melanie. I can't express my thanks enough to my supervisors Gabriela, Sandy, and Fabio, for their never ending advice and inspiration over the past four years. I would also like to thank my friends: Paul, Sarah, Ken, Kevin and Saemy for their general encouragement every step of the way. And to Andrew for an uninterrupted stream of motivational cat GIFs over the final two months.

"Make it so."

Jean-Luc Picard

LIST OF PUBLICATIONS

This thesis has produced three peer-reviewed publications:

1. [1] Jason Adair, Alexander Brownlee, and Gabriela Ochoa. **Evolutionary Algorithms with Linkage Information for Feature Selection in Brain Computer Interfaces.** In *Advances in Intelligent Systems and Computing*, volume 513, pages 287-307. 2017. ISBN 9783319465616. doi: 10.1007/978-3-319-46562-3-19. This paper serves as the foundation for Chapter 5.

2. [3] Jason Adair, Alexander E. I. Brownlee, and Gabriela Ochoa. **Mutual information iterated local search: A wrapper-filter hybrid for feature selection in brain computer interfaces.** In *Applications of Evolutionary Computation*, pages 63-77, Cham, 2018. Springer International Publishing. ISBN 978-3-319-77538-8. This paper serves as the foundation for Chapter 6.

3. [2] Jason Adair, Alexander Brownlee, Fabio Daolio, and Gabriela Ochoa. **Evolving training sets for improved transfer learning in brain computer interfaces.** In M*achine Learning, Optimization, and Big Data*, pages 186-197. Springer International Publishing, 2018. ISBN 978-3-319-72926-8. This paper serves as the foundation for Chapter 7.

CONTENTS

I INTRODUCTION 1

1 INTRODUCTION 2

1.1 The Need for BCI . . . 2
1.2 The BCI Paradigm . . . 3
1.3 Problems in the Data . . . 3
1.3.1 Signal to Noise Ratio . . . 4
1.3.2 Difficulties in Collection of Training Data . . . 4
1.3.3 Non-Stationarity . . . 5
1.3.4 Curse of Dimensionality . . . 5
1.4 Possible Solutions . . . 6
1.5 Contributions of this Thesis . . . 7
1.6 Structure of this Thesis . . . 9

II BACKGROUND 11

2 CHAPTER 2 - BACKGROUND 12

2.1 Biology . . . 12
2.1.1 Neurons . . . 12
2.1.2 Signal Transmission . . . 13
2.1.3 Detectable Neural Signals . . . 14
2.2 Types of BCI Recording . . . 18
2.2.1 Invasive Methods . . . 19
2.2.2 Non-Invasive . . . 22
2.3 Non-Invasive BCI Paradigms . . . 26
2.3.1 Sensorimotor . . . 27
2.3.2 Slow Cortical Potentials (SCP) . . . 27
2.3.3 Visually Evoked Potentials (VEP) . . . 28
2.3.4 P300 . . . 28
2.3.5 Summary . . . 30
2.4 Data Preprocessing . . . 30

2.4.1 Referencing . . . 30
2.4.2 Frequency Filtering . . . 31
2.4.3 Normalisation . . . 31
2.4.4 Artefact Removal . . . 31
2.5 Types of Features . . . 33
2.5.1 Time Domain Features . . . 33
2.5.2 Frequency Domain Features . . . 34
2.5.3 Time-Frequency Domain Features . . . 37
2.5.4 Feature Vector Construction . . . 38
2.5.5 Summary . . . 38
2.6 Classifiers . . . 39
2.6.1 Classifier Taxonomy . . . 39
2.6.2 Fisher's Linear Discriminant Analysis (FLDA) . . . 40
2.6.3 Bayesian Linear Discriminant Analysis (BLDA) . . . 41
2.6.4 Support Vector Machine (SVM) . . . 42
2.6.5 k-Nearest Neighbour (KNN) . . . 43
2.6.6 Artificial Neural Networks (ANN) . . . 44

iii LITERATURE 45

3 CHAPTER 3 - LITERATURE 46
3.1 Feature Selection . . . 46
3.1.1 Filters . . . 47
3.1.2 Wrappers . . . 50
3.1.3 Embedded . . . 55
3.1.4 Hybrid Approaches . . . 55
3.1.5 Feature Selection Summary . . . 57
3.2 Transfer Learning . . . 57
3.2.1 Ensembles . . . 58
3.2.2 ELGI . . . 59
3.3 Summary . . . 60

iv METHODOLOGY 61

4 CHAPTER 4 - EXPERIMENTAL SETUP 62
4.1 Datasets . . . 62

4.1.1 Dataset D1 - Berlin BCI competition II Datasets III 63
4.1.2 Dataset D2 - Berlin BCI competition II Datasets IV 65
4.1.3 Dataset D3 - Riken - Subject A 66
4.1.4 Dataset D4 - P300 Speller (Hoffman) 68
4.1.5 Feature Extraction 70
4.2 Size of Selected Feature Subset 73
4.3 Fitness Function 73
4.4 Tools 73

V LINKAGE 75

5 CHAPTER 5 - LINKAGE 76
5.1 Introduction 76
5.2 Preliminary Algorithm Exploration 76
5.2.1 Experimental Parameters 77
5.2.2 Algorithm Performance Comparison 79
5.2.3 Evidence of Feature Interaction 80
5.2.4 Discussion of Selected Features 81
5.3 Linkage Integration Design 83
5.3.1 Linkage Map Generation 84
5.3.2 Linkage in Dataset D1 85
5.3.3 Linkage Integration 87
5.3.4 Results and Discussion 89
5.3.5 ILS with Linkage 94
5.4 Analysis 95
5.5 Conclusion 96

VI MUTUAL INFORMATION 98

6 CHAPTER 6 - MUTUAL INFORMATION 99
6.1 Mutual Information 99
6.1.1 Entropy 100
6.1.2 Mutual Information 100
6.1.3 Minimal Redundancy Maximum Relevance 101
6.2 Proposed Method - MRMR-ILS 102
6.2.1 Iterated Local Search 103

6.2.2 Minimal Redundancy Maximal Relevance-Iterated Local Search . . . 103
6.3 Methodology . . . 103
6.3.1 Classifiers . . . 104
6.3.2 Fitness Function . . . 106
6.3.3 Search Algorithm Parameters . . . 106
6.3.4 Benchmark Methods . . . 106
6.4 Results and Discussion . . . 107
6.5 Conclusion . . . 114

vii INSTANCE TRANSFER 119

7 CHAPTER 7 - INSTANCE TRANSFER 120
7.1 Transfer Learning In BCI . . . 120
7.1.1 Ensembles . . . 121
7.2 Methodology . . . 122
7.2.1 Dataset . . . 122
7.2.2 Classifier . . . 122
7.2.3 Conditions . . . 123
7.2.4 Compared Algorithms . . . 124
7.3 Evolved ELGI Ensemble . . . 126
7.4 Results . . . 127
7.5 Discussion and Conclusion . . . 131

viii SUMMARY AND CONCLUSIONS 133

8 CHAPTER 8 - SUMMARY AND CONCLUSIONS 134
8.1 Contributions . . . 137
8.2 General Conclusion . . . 139
8.2.1 Potential Impacts of our Contributions . . . 140
8.3 Summary . . . 140
8.4 Future Work . . . 141

ix APPENDICES 1

A APPENDIX 2
A.1 Feature Reference Table for Datasets D1 . . . 2

A.2 Feature Reference Table for Datasets D2 3
A.3 Feature Reference Table for Datasets D3 4
A.4 Participant Descriptions (Dataset D4: P300 Speller (Hoffman)) . 5

LIST OF FIGURES

Figure 1.1 A simplified diagram of the Brain Computer Interface paradigm 3
Figure 2.1 An example of a typical presynaptic (signal generating) neuron with its synapses making contact with a postsynaptic (signal receiving) cell [173]. 13
Figure 2.2 Differing areas of activation between imagined left and right hands squeezing a ball 15
Figure 2.3 Diagram displaying discrete sampling of an analogue signal preserving the waveform 16
Figure 2.4 Recording techniques in order of invasiveness. 18
Figure 2.5 Photograph of a NeuroScan 64-electrode EEG cap. Image courtesy of [50] 26
Figure 2.6 Visual stimuli presented to the user in (a), where each column and row are flashed randomly 29
Figure 2.7 Examples of different noise sources. (a) EEG signal with no obvious noise, (b) blink, (c) eye movement (EOG), (d) 50 Hz interference, (e) Muscle movement (EMG), (f) Heart beat (ECG) [16] 32
Figure 2.8 Review of Feature Extraction methods reported in [14] . 36
Figure 2.9 Comparison between PCA and LDA 40
Figure 2.10 A depiction of a Support Vector Machine (*SVM*). 43
Figure 3.1 Search path of the Iterated Local Search (ILS) Algorithm [64] 54
Figure 4.1 A timeline of the experimental paradigm used in Berlin BCI Competition II Dataset III. 63
Figure 4.2 The electrode configuration for D1: Berlin BCI Competition II Dataset III 64

Figure 4.3 The electrode configuration for D2: Berlin BCI Competition II Dataset IV 65
Figure 4.4 A timeline of the experimental paradigm used in the Riken - Subject A dataset. 67
Figure 4.5 The electrode configuration for the Riken - Subject A . . 67
Figure 4.6 Images presented in the P300 paradigm 69
Figure 4.7 The electrode configuration for Dataset D4 70
Figure 5.1 Box plots comparing the error rates of solutions found by each algorithm over 30 runs. 80
Figure 5.2 Selected features according to individual predictive accuracy 81
Figure 5.3 Most commonly selected channels in best performing solutions found 82
Figure 5.4 Most commonly selected frequency bandwidths in best performing solutions found 82
Figure 5.5 Most commonly selected epochs in best performing solutions found 83
Figure 5.6 Sequence diagram displaying the incorporation of Linkage in the Feature Selection phase 84
Figure 5.7 Linkage scores between all potential feature pairings . . 86
Figure 5.8 Figure 5.7 filtered to display only benign linkage 87
Figure 5.9 Figure 5.7 filtered to display only malign linkage 88
Figure 5.10 Preliminary testing of different methods of linkage guidance in Hill Climbing algorithms 90
Figure 5.11 Comparison of error rates obtained by Iterated Local Search, and Iterated Local Search with guidance via positive and negative linkage 95
Figure 6.1 Mutual Information between variables X and Y ($I(X;Y)$), seen as the over lap of the entropies of X ($H(X)$) and Y ($H(Y)$). 101
Figure 6.2 Minimum Redundancy Maximum Relevance (mRMR) diagram 102

Figure 6.3 Comparison of the most selected features of the ILS and MRMR-ILS algorithms on dataset *D1* - BCI Competition II dataset III 112
Figure 6.4 Comparison of the most selected features of the ILS and MRMR-ILS algorithms on dataset *D2* - BCI Competition II dataset IV 113
Figure 6.5 Comparison of the most selected features of the ILS and MRMR-ILS algorithms on dataset *D3* - Riken 114
Figure 6.6 Comparison between ILS and MRMR-ILS over each iteration of the algorithms for the KNN classifier on dataset *D1* - BCI Competition II dataset III 115
Figure 6.7 Comparison between ILS and MRMR-ILS over each iteration of the algorithms for the SVM classifier on dataset *D1* - BCI Competition II dataset III 116
Figure 6.8 Comparison between ILS and MRMR-ILS over each iteration of the algorithms for the KNN classifier on dataset *D2* - BCI Competition II dataset IV 116
Figure 6.9 Comparison between ILS and MRMR-ILS over each iteration of the algorithms for the SVM classifier on dataset *D2* - BCI Competition II dataset IV 117
Figure 6.10 Comparison between ILS and MRMR-ILS over each iteration of the algorithms for the KNN classifier on dataset *D3* - RIKEN Subject A 117
Figure 6.11 Comparison between ILS and MRMR-ILS over each iteration of the algorithms for the SVM classifier on *D3* - RIKEN Subject A 118
Figure 7.1 Division of paradigm into smaller sub-problems within each run. 123
Figure 7.2 ELGI approach displaying two classifiers are trained for every participant. 125
Figure 7.3 Algorithm performance by number of stimuli presentations, with differing quantities of participant-specific training data available 128

Figure 7.4 Round Accuracy over all testing sets displayed for each quantity of participant-specific training data, separated for each participant. 129
Figure 7.5 Round Accuracy over all quantities of training data for each testing set, separated for each participant. 130
Figure 7.6 Fit of hierarchical linear models, with random effects for each participant, estimating (a) the overall Round Accuracy per testing set and (b) the change in Round Accuracy over training set size. 131

LIST OF TABLES

Table 2.1 Summary of frequency bands according to each recording method 17

Table 4.1 Description of Dataset D1: Berlin BCI Competition II: Dataset III 64

Table 4.2 Description of Dataset D2: Berlin BCI Competition II: Dataset IV 66

Table 4.3 Description of Dataset D3: Riken - Subject A 68

Table 4.4 Description of Dataset D4: P300 Speller (Hoffman) . . . 70

Table 4.5 The number of features extracted from Datasets D1, D2, and D3 71

Table 4.6 Overview of datasets used in the following experiments 72

Table 5.1 Comparison of Cross Validation Error Rates between Greedy Linkage algorithms and Linkage-guided Hill Climbing algorithms 93

Table 5.2 Table comparing the correlation of solution fitness (CVE Rate) and predictive accuracy on unseen data 96

Table 6.1 Results of each feature selection algorithm while using the KNN Classifier 108

Table 6.2 Results of feature selection algorithm while using the SVM Classifier with selected subset sizes (Selected f) . . 110

Table 6.3 Correlations between Cross Validation Error Rates and Accuracy of Solution during ILS and MRMR-ILS Search 115

Table A.1 Indices of Power Spectral Density features according to the frequency, channel, and time epoch for Dataset D1: Berlin BCI Competition III dataset. 2

Table A.2 Indices of Power Spectral Density features according to the frequency, channel, and time epoch for Dataset D2: Berlin BCI Competition IV dataset. 3

Table A.3 Indices of Power Spectral Density features according to the frequency, channel, and time epoch for Dataset D3: Riken - Subject A. 4
Table A.4 Table provides a description of the participants within dataset D4: P300 Speller (Hoffman) 5

LIST OF ACRONYMS

ACO	Ant Colony Optimisation
ALS	Amyotrophic Lateral Sclerosis
ANN	Artificial Neural Networks
AR	Autoregressive Modelling
BBO	Biogeography Based Optimisation
BCI	Brain Computer Interface
BLDA	Bayesian Linear Discriminant Analysis
BOLD	Blood Oxygenation Level Dependent
CFS	Correlation-Based Feature Selection
cGA	Compact Genetic Algorithm
CSD	Current Source Density
CSP	Common Spatial Patterns
CVE	Cross Validation Error
DE	Differential Evolution
DPSO	Discrete Particle Swarm Optimisation
DWT	Discrete Wavelet Transform
EA	Evolutionary Algorithm
ECG	Electrocardiography
ecGA	Extended Compact Genetic Algorithm
ECoG	Electrocorticography
EDA	Estimation of Distribution algorithm
EEG	Electroencephalography

eELGI Evolved Ensemble Learning Generic Information

ELGI Ensemble Learning Generic Information

EMG Electromyography

EOG Electrooculography

ERD Event Related Desynchronisation

ERP Event Related Potential

ERS Event Related Synchronisation

FCBCSP Filter Bank Common Spatial Pattern

FCBF Fast Correlation Based Filter

FLDA Fisher's Linear Discriminant Analysis

fMRI Functional Magnetic Resonance Imaging

FS Fisher Score Algorithm

GA Genetic Algorithm

GA-SVM Genetic Algorithm Support Vector Machine

GLFS Greedy Linkage Feature Selection

HC Hill Climbing

HS Harmony Search

ICA Independent Component Analysis

ILS Iterated Local Search

ILS/GA Iterated Local Search Guided Mutation

IWO Invasive Weed Optimisation

L-ILS Linkage-Iterated Local Search

LASSO Least Absolute Shrinkage and Selection Operator

LDA Linear Discriminant Analysis

LED Light Emitting Diode

LFP	Local Field Potentials
KNN	K-Nearest Neighbour
MA	Memetic Algorithm
MEG	Magnetoencephalography
MIBIFS	Mutual Information Best Individual Feature Selection
MIFS	Mutual Information Feature Selection
MIRSR	Mutual Information-based Rough Set Reduction
ML	Mutual Linkage
MLFS	Maximum Linkage Feature Selection
MLP	Multilayer Perceptron
mRMR	Minimum Redundancy Maximum Relevance
mRMR-ILS	Minimum Redundancy Maximum Relevance Iterated Local Search
MRP	Movement Related Potentials
MSE	Mean Squared Error
MUA	Multiple Unit Activity
NIRS	Near Infrared Spectroscopy
NSGA-II	Non-dominated Sorting Genetic Algorithm-II
PCA	Principle Component Analysis
PILS	Population Based Iterated Local Search
PSD	Power Spectral Density
PSO	Particle Swarm Optimisation
PSOBE	Particle Swarm Optimisation Backwards Elimination
RFE	Recursive Feature Elimination
SBS	Sequential Backward Search
SCP	Slow Cortical Potential

SFFS Sequential Forward Floating Search

SFS Sequential Forward Search

SLII Standard Learning Individual Information

SNR Signal-to-Noise Ratio

SQUIDs Superconducting Quantum Interference Devices

SSVEPs Steady-State Visual Evoked Potentials

SUA Single Unit Activity

SVM Support Vector Machine

TLBO Teaching Learning Based Optimisation

TVEPs Transient Visual Evoked Potentials

VEP Visual Evoked Potential

WPT Wavelet Packet Transform

Part I

INTRODUCTION

1

INTRODUCTION

In this thesis we present our research on search techniques for improving the effectiveness of training datasets in Brain Computer Interface applications. We introduce intelligent operators that are effective in Feature Selection, and improve the performance of systems trained on a population of users (Instance Transferral). We begin by introducing BCI and the problems needing to be addressed in this challenging area, then move on to the possible solutions, before introducing the contributions of this thesis.

1.1 THE NEED FOR BCI

Brain Computer Interfaces (otherwise known as *Brain Machine Interfaces*) are a domain of hardware/software in which a user can interact with a machine without the need for motor activity [185], communicating instead via signals generated by the nervous system. In real world applications, this interface supports users in controlling artificial limbs, underpins assisted communication devices, administering psychological treatments, and finds use in recreational applications [19]. If a method of accurately measuring the structures and behaviours of the brain can be devised, a new horizon in science will open to us: from replacing missing limbs, to augmenting what is already there. However, this is far from a trivial task.

The brain is an exceptionally complex organ with a large degree of plasticity. This means that simple, catch-all models for predicting signals are all but impossible, and we must customise the models for each user to some degree. This customisation can be expensive in terms of computational costs, data requirements, and can also lead to over-fitting. For these reasons we turn to intelligent search methods to ensure that the model is customised as accurately and efficiently as possible using the limited data available.

1.2 THE BCI PARADIGM

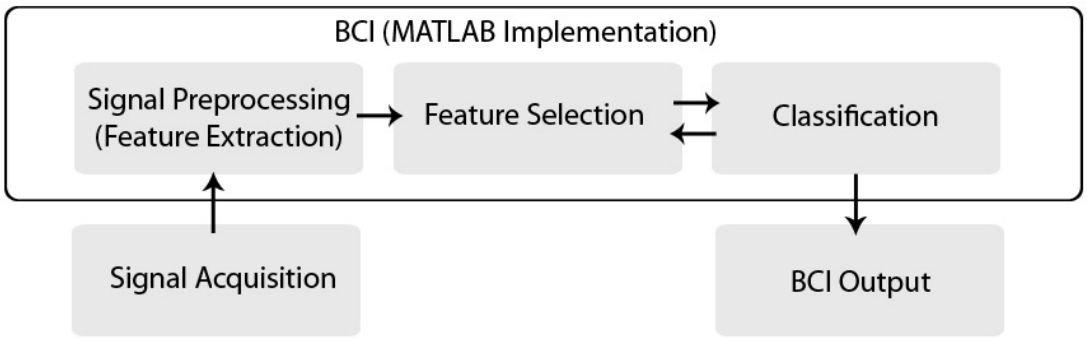


Figure 1.1: A simplified diagram of the Brain Computer Interface paradigm

Brain Computer Interface (BCI) applications typically seek to acquire neurological signals and derive a discrete classification of the user's intent. For example, allowing a user to select onscreen prompts to control a communication device. To achieve this, they rely on the architecture in Figure 1.1.

Signal Acquistion detects and records the neurological signals. *Signal Preprocessing* removes artefacts and generates numerical representations of the data, known as features, for creation of a predictive model. *Feature Selection* selects only the most relevant features to create a model; ensuring a strong representation of the patterns that are required for the system. *Classification* involves training a model, and using it to assign classes to new, unlabelled inputs. *BCI Output* can be used to control hardware or software.

1.3 PROBLEMS IN THE DATA

BCI relies on signals that originate from the electrical activity in a network of ~860 billion interconnected neurons, of more than one thousand different types [72]. Each of these individual neurons is connected to an average of seven thousand others, resulting in an estimated total of one quadrillion connections. Directly sampling the electrical activity of this entire network is impossible, and even localised sampling comes at great cost.

Approaches to obtain neurological recordings can be separated into two main groupings; invasive and non-invasive. While invasive recordings can

allow exceptional spatial and temporal resolutions, they involve sub-cranial surgery with potentially severe health risks and prohibitive financial costs [123]. With these problems in mind, we find that the non-invasive method, *Electroencephalography (EEG)*, is the most popular method of detecting neurological signals for BCI [129]. This technique avoids difficulties associated with invasive methods, but introduces and exasperates others including: Signal-to-Noise Ratios, lack of training data, non-stationarity, and increased dimensionality [140]. Each of these problematic areas are now summarised briefly.

1.3.1 *Signal to Noise Ratio*

EEG involves the placement of electrodes on the scalp surface, measuring the electrical fields of the underlying neural matter, and relaying it back to a computer for processing. This technique has become prominent over other more invasive methods due to its ease of maintenance, substantially safer lack of invasive procedures, and relative low cost [186]. However, it does present some non-trivial problems: as the electrodes that detect the electrical fields are placed on the scalp, the signal must be powerful enough to penetrate two to three centimetres of cranium, skin and other biological material [179]. For this level of energy to be generated, approximately one hundred million neurons over six square centimeters of neural matter must be active [156], resulting in low spatial resolution, contamination of signals between electrodes, and natural band passing of the frequencies when travelling through the skull. The signal is further distorted by additional electrical signals being detected from eye movements (electro-oculography), muscle movements (electromyography), and environmental noise (for example, the 50 Hz band often consists of electrical activity from nearby wall sockets [123]).

1.3.2 *Difficulties in Collection of Training Data*

User concentration is paramount to ensure that the training data acquired is of adequate quality to create good models. Unfortunately, BCI paradigms are often tediously repetitive tasks, resulting in the recording of large training

sets requiring a substantial quantity of time. This results in user concentration deterioration, and unreliable instances being introduced into our datasets [172]. The recording sessions are also limited by the expense of the equipment, the technician, and the user's willingness to continue. Due to these issues, BCI recordings contain relatively small numbers of samples, making it difficult to create an adequate model.

1.3.3 *Non-Stationarity*

One of the key issues in BCI is that the signals are non-stationary: neural patterns not only differ between users, but are also subject to temporal drift; where patterns within data obtained from a single user change drastically over time [80]. *Zero Training* systems, trained exclusively on users from previous sessions, are an ideal goal; however, this non-stationarity means highly accurate Zero Training systems may not be possible. Consequently, focus must be placed on minimising the user-specific training information required by maximising the effectiveness of the data available.

1.3.4 *Curse of Dimensionality*

In recent years, new types of recording equipment have been developed and significant increases in electrode densities have been attained. However, these further exasperate the already considerable computational load by increasing the dimensionality of the signals, and adding additional inconvenience and expense to the end user. In EEG for example, it is recommended that the sampling rate be approximately three times higher than the upper limit of the filter e.g. 70 Hz requiring no less than 200 samples per second [162]. As 32-256 channel devices are commonly used, in excess of 50,000 samples per second are to be expected which inevitably proves to be very computationally expensive. Advances in another recording technique known as intercellular recording, have doubled the number of recordable neurons every seven years since the 1950s [165], and suffers the additional issue of limited bandwidth.

1.4 POSSIBLE SOLUTIONS

Ideally, a portable BCI should be created that allows maximum degrees of control over peripherals while still maintaining a functional response time [96]. To achieve this, we must increase the Signal to Noise Ratio, while decreasing the volume of data presented to the classifier to ensure that it can respond quickly [20]. Feature Selection has been demonstrated to be an effective solution to this problem: Rejecting a substantive portion of the data can not only lower the computation requirements, but can also increase predictive accuracy [112, 145] and potentially allows additional classes to be included, increasing the Degrees Of Freedom for the user [52].

The high dimensional nature of BCI data is further complicated by the small number of training instances available, sometimes known as the 'large p, small n' problem [34]. While Feature Selection reduces this problem, additional instances are sometimes necessary. *Transfer Learning* allows knowledge to be taken from prior participants for the purpose of developing future models [80]. However, variations between different users can cause poorly fitted models, which can be overcome through *Instance Transfer* [180]. This optimisation of weights and movement of data is non-trivial and exceedingly difficult due to the aforementioned issues. The resulting large and complex search spaces mean that Search-based algorithms for use in Feature Selection and Transfer Learning are a critical area of research in BCI.

In summary, Brain Computer Interfaces provide the worst-case-scenario for machine learning: high dimensionality, low numbers of training samples, low signal-to-noise ratios, non-stationary sources, and cross-contamination between vectors. We can use search based techniques to address these problems. *Feature Selection* involves obtaining near optimal feature subsets to reduce the dimensionality of the data, thus decreasing the training and prediction time costs, creating simpler models, and increasing the predictive accuracy [190]. *Instance Selection* allows detection of relevant, participant independent instances to train models for new patients, reducing calibration times, financial expense, and user distress.

Specifically, we ask the following questions;

RQ1 Can existing feature selection methods be improved upon by integration of additional measures of solution subset relevance?

RQ2 Do solutions found by algorithms that include these measures better generalise to new, unseen data?

RQ3 Can datasets from prior users be better utilised to improve models for new users? Specifically, can they be used to: *(i)* reduce the training data required; *(ii)* increase predictive accuracy; *(iii)* mitigate difficulties in interpreting user-input with neurological damage; and *(iv)* mitigate the effects of temporal drift.

1.5 CONTRIBUTIONS OF THIS THESIS

The primary focus of this thesis is to improve search techniques for Feature Selection and Instance Transfer for EEG data in Brain Computer Interface applications. The overall contribution will be the utilisation of Information Theory based metrics, Linkage information, and classifier accuracy to obtain more generalisable feature subsets, and optimisation of data subsets between users for creation of ensemble methods. This is divided into the following more specific contributions:

1. **An exploration of search methodologies** for Feature Selection on Brain Computer Interface datasets. We have shown that Wrapper methods typically find higher quality feature subsets than Filters, giving further support to results found in literature. *Iterated Local Search (ILS)* was applied to the BCI field for the first time, and demonstrated to perform comparably with more computationally expensive techniques such as *Genetic Algorithms*, and the state-of-the-art embedded method: *LASSO*;

2. **Intelligent operators** were developed to account for feature interaction within the search space. *Linkage Information* and *Information Theory* based metrics were used to guide the permutation operators in ILS, with the aim of increasing the predictive accuracy of models on unseen data;

3. **An investigation into effective fitness measurements** for the generalisability of optimal feature subsets. A common measure of solution quality in Wrapper approaches is Cross-validation error rates achieved from training data. However, we show that it is a poor indicator of solution improvement when the model becomes over-fitted. Metrics, such as our *Intrasolution Linkage Score*, may provide better indications of a subset's generalisability to new data. This effect may also be mitigated by including additional metrics in the search process, as demonstrated by our algorithm, *Minimum Redundancy Maximum Relevance Iterated Local Search (MRMR-ILS)*;

4. **A new method of optimising datasets** for creating ensembles. BCI applications typically have small datasets with high dimensionality. While Feature Selection can reduce this dimensionality, limited sample numbers are still an issue when trying to create an adequate model. A state-of-the-art method known as *Ensemble Learning Generic Information (ELGI)* creates an ensemble of models based on recombining the current user, with data from past users. We have developed a technique for the optimisation of the datasets used in this process. Using a local search to perform instance transfer, we were able to increase the generalisibility of the dataset, before recombination with user-specific data. We called this *Evolved Ensemble Learning Generic Information (eELGI)*. We found that this technique created models that were able to achieve higher predictive accuracies even when we reduced the quantity of user-specific data available. Further improvements were seen in the models resistance to *neural drift*: over time, a user's neural patterns change, rendering well fitting past models ineffective. Using our technique, we find that BCI systems remain much more effective over the dataset's two week period.

1.6 STRUCTURE OF THIS THESIS

This thesis is structured in the following manner:

Chapter 2 - Background gives a detailed overview of *Brain Computer Interfaces*. It first begins with the *biological origin* of *neurological signals*, and how they can be *detected* for use in BCI. The methods which can be used to *elicit signals* for identification are then discussed. *Preprocessing* methods are then explained, and an overview of different *classification methods* is given.

Chapter 3 - Literature discusses publications concerning the optimisation of BCI data. *Feature Selection* is first discussed in terms of *Filter*, *Wrapper* and *Embedded* methods, after which *Hybrid methods* are introduced. *Transfer Learning* methods are then discussed, with focus given to the use of *Ensembles* in BCI.

Chapter 4 - Methodology gives details on the *Datasets*, *Preprocessing*, and *Feature Extraction* methods used in this thesis. This is followed by parameter descriptions including *Solution Size*, *Fitness Functions* and *Tools* used.

Chapter 5 - Linkage provides our first contribution chapter. In this, we provide a *preliminary exploration* of different Wrapper methods, and discuss indications of *feature interactions*. *Linkage-aware operators* and *metrics* are then developed, followed by further preliminary testing. Our first Iterated Local Search variants are introduced, in the form of a *Linkage-aware ILS*. These are then *evaluated* and *discussed*.

Chapter 6 - Mutual Information describes a well established Information Theory-based Filter, *Minimum Redundancy Maximum Relevance (mRMR)*, in terms of *Entropy* and *Mutual Information*. We then describe our ILS variant *Minimum Redundancy Maximum Relevance Iterated Local Search (MRMR-ILS)*. Following this, the *methodology* is detailed, and experimental *results* comparing it against existing *Filter* and *Wrapper* methods presented. *Conclusions* are then drawn.

Chapter 7 - Instance Transfer begins by discussing *transfer learning in BCI*, with focus on the use of *Ensembles*. The state-of-the-art approach *Ensemble Learning Generic Information (ELGI)* is described. A *methodology* for experimentation and a detailed description of our optimisation approach, *Evolved Ensemble Learning Generic Information (eELGI)*, is given. Experimental *results* are then presented and *discussed*.

Chapter 8 - Summary and Conclusions give an overview of this thesis. The motivation for Brain Computer Interfaces is given, followed by problems in their implementation. We then offer our *contributions*, before explicitly stating them. A *general conclusion* is then given to demonstrate how our contributions directly address the problems in the field, followed by *real world impacts* of these advancements. The thesis is then concluded in a *summary* with suggestions for *future work*.

Part II

BACKGROUND

2

CHAPTER 2 - BACKGROUND

In order to establish the challenges facing BCI systems, we will first consider the Biological origins of the signals in Section 2.1, and the manner in which they are measured (Section 2.2). Non-Invasive BCI systems rely on paradigms to modulate these signals, which are desribed in Section 2.3. Details of Preprocessing techniques (Section 2.4) and initial Dimensionality Reduction through Feature Extraction are then given (Section 2.5). Finally, a description of Classifiers common in BCI are given (Section 2.6).

2.1 BIOLOGY

Complex sequences of signals have been observed originating from a single cell in invertebrates, but it appears that behaviour in higher vertebrates is always governed by a larger number of processes [79]. At the heart of these processes in the human brain is a network of ~860 billion interconnected neurons, of over one thousand different types [72]. Each neuron is connected to an average of seven thousand others, resulting in an estimated total of one quadrillion connections [9]. While the connections are arranged in only a few common structures, these still allow for great complexity. With so many sources of information, recording individual neurons (*intracellular recordings*) is an impracticality. We are therefore faced with methods which listen to populations of neurons (*intercellular*) by application of electrodes. The following section describes the biological source of neural signals.

2.1.1 *Neurons*

Firstly, it is important that we describe the structure, variety, and processes of neurons as variations in these factors introduce a great deal of complexity

into neural decoding. While there are two main classifications of nerve cells, the information desired to convey 'intent' for BCI is carried through the neurons [75]. The number of connections to and from each neuron greatly varies between neuron types; motor neurons in the spine can have around ten thousand contacts while a neuron within the brain itself can exceed one million contacts [79].

As seen in Figure 2.1, a nerve cell can be divided into 4 main morphological regions; the body, dendrites, axon, and presynaptic terminals. The cell body is typically oval in shape, receiving input signals from a number of thread-like dendrites, and passing on its signal down the axon, a channel, often wrapped in a lipid sheath (myelin), that branches into terminals that end in enlarged tips known as buttons. These buttons are in close proximity with the post-synaptic (signal-receiving) cell, separated by a very small space known as the synaptic cleft.

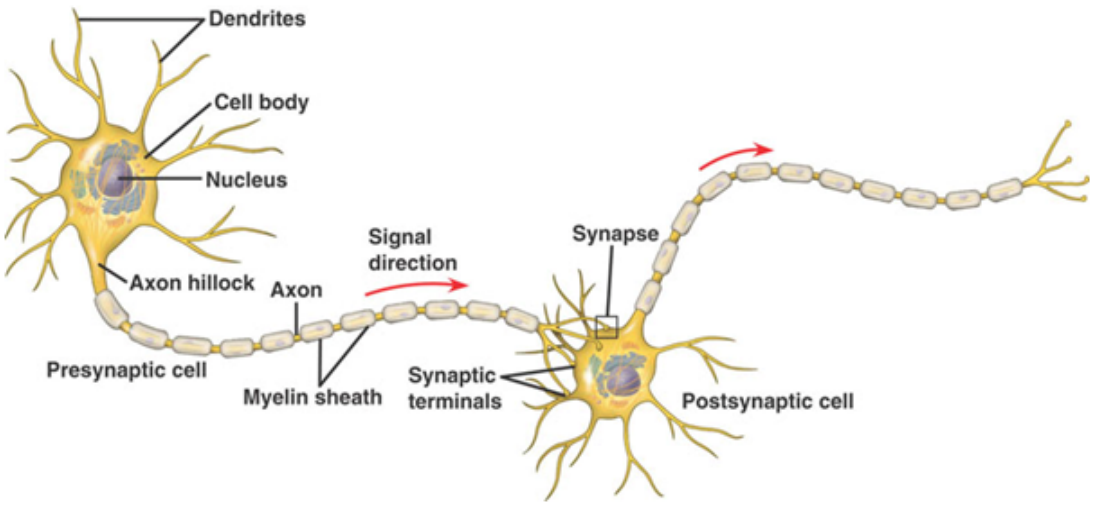


Figure 2.1: An example of a typical presynaptic (signal generating) neuron with its synapses making contact with a postsynaptic (signal receiving) cell [173].

2.1.2 *Signal Transmission*

The process for signal transmission in all nerve cells follows a similar procedure; input signal, trigger signal, conducting signal, and output signal [79], each of which generates a detectable electrical signal known as a *potential*. The input signal is typically received at the dendrites, in which neurotransmitters bind to

surface proteins and generate an electrical charge known as a *'receptor potential'*. This potential is usually faint, decaying in a matter of millimetres [79], but should the sum of all receptor potentials supersede the threshold at the trigger zone, a larger electrical potential, known as an *action potential* is generated. The action potential is then carried down the axon.At the end of the axon, this signal triggers the release of neurotransmitters which bind to the dendrites of the next (postsynaptic) cell, which generates a *receptor potential* [30]. Different behaviours are observed according to the different characteristics of neuron categories. Beating neurons are spontaneously active, firing even when there is no stimulation, whereas bursting neurons only fire when triggered. In beating neurons, a synaptic potential could trigger a single or a number of action potentials in a bursting cell, while beating neurons simply increase the frequency of theirs [79]. As action potentials are remarkably stereotyped, it is not uncommon for action potentials in sensory and motor neurons to be indistinguishable. The two key features carrying information in neurons are; quantity of fires, and their timings. The number of action potentials generated is determined by a range of different factors, largely dependent on the type of neuron.

2.1.3 *Detectable Neural Signals*

Signals are detectable variations in physical phenomena over time. In BCI, there are two main classifications of detectable signals; *hemodynamic* and *electrophysiological*. Hemodynamic responses are measured by detecting changes in the properties of the blood circulating the neural matter [19]. Demand for energy, in the form of glucose, is higher for active neurons, therefore we see an increase of blood flow within millimetres of the active region. This increase in blood flow also delivers higher levels of oxyhaemoglobin than the neurons require, resulting in a change in the ratio of oxyhaemoglobin to deoxyhaemoglobin [149]. As oxygenated blood is diamagnetic and the deoxygenated blood is paramagnetic, this process can be measured by external equipment, like that captured by Functional Magnetic Resonance Imaging in figure 2.2.

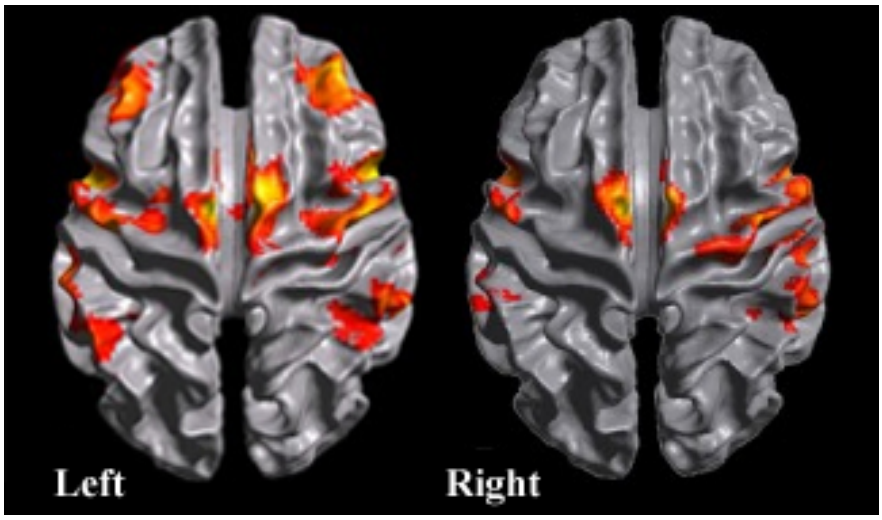


Figure 2.2: Differing areas of activation between imagined left and right hands squeezing a ball. Captured using fMRI by detecting decreases in the magnetic properties of blood in the regions. Image adapted using figures from [167].

Electrophysiological signals in BCI are generated by the action potentials of the firing neurons. Continuous-time signals are measurable at any point in instant time, however for use in the BCI domain, it is common to create discrete-time signals by sampling the continuous-time signal at set intervals. While this sampling reduces the resolution of the signal, adequate sampling speeds can preserve the waveform as in figure 2.3. Two of the defining properties of a signal are their amplitude (the magnitude of the signal) and their frequency [113].

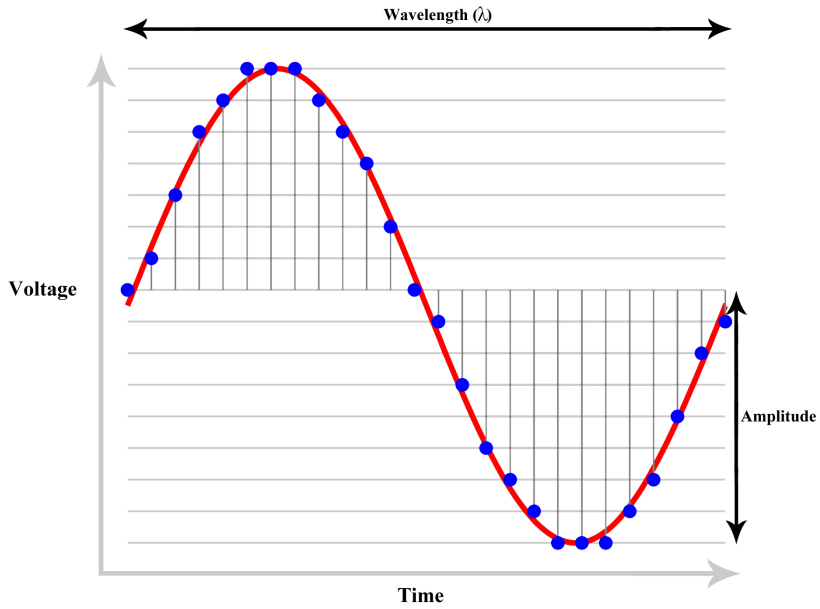


Figure 2.3: Diagram displaying discrete sampling of an analogue signal preserving the waveform. The number of complete wavelengths (λ) over a given time period (*t*) dictates the frequency (as this is typically measured in seconds, the unit tends to be Hertz (*Hz*)

2.1.3.1 *Frequency Bands*

Analogue neural recordings rely largely on differentiation between 3 main frequency bands that reflect the activities of groups of neurons.

LOW FREQUENCY BANDS: The lowest frequency band encompasses 6-13 Hz in *Local Field Potentials (LFP)*, under 2 Hz in *Electrocorticography (ECoG)* and around 67 Hz in *Electroencephalography (EEG)* and *Magnetoencephalography (MEG)* recordings [179]. Changes within this range tend to be of limited use for BCIs as it represents large populations of neurons and, due to its slow frequency, only low bit rates are possible.

MID FREQUENCY BANDS: The next band is inclusive of both the μ and β rhythms which are both highly correlated with actual and intended movement [95]. In LFPs this band consists of 16-42 Hz, in ECoG it is 6-30 Hz and EEG and MEG register it at 10-30 Hz. These frequencies are of specific interest in the field

of BCIs for prosthetics as they appear to desynchronize in the sensorimotor cortex when the user attempts a movement, and more interestingly, imagines a movement; suggesting that they may provide a potential method of control in prosthetic devices designed for users who lack muscle control [136]. While it is generally agreed that these rhythms lack the detail needed to decipher the directional intentions of limb movement [123], some researchers, such as Wolpaw [185] contend this idea and have continued to adapt EEG-based BCIs with some success including 2D cursor control.

HIGH FREQUENCY BANDS: The higher frequencies that constitute the Gamma banding are recorded as 62-87 Hz in EEG and MEG, 34-128 Hz in ECoG, and 63-200 Hz in LFPs [179]. When observing the motor cortex, it has been noted that there is a close correlation between the firing of individual neurons and the gamma band during muscle contractions [51]. This would suggest that it may provide a non-invasive source of information that is rich enough to convey directional information and is drawing increasing interest from the field. Unfortunately, this band is especially prone to noise artefacts such as electromyography (electrical signals originating in the muscles) and electrooculography (electrical signals caused by eye movements) [123].

Table 2.1: Summary of frequency bands according to each recording method

		Recording Technique			
		LFP	**ECoG**	**EEG**	**MEG**
Frequency Band (Hz)	**Low**	6-13	<2	67	67
	Mid	16-42	6-30	10-30	10-30
	High	63-200	34-128	62-87	62-87

2.1.3.2 *Amplitude*

Amplitude of the wave is most commonly used to observe changes in the energy of specific bandwidths, but can be used on its own for identification of neural activity. An example of this is the P300 wave; when a person is attentive

to a specific stimulus, and something unexpected occurs, a spike in neural activity is seen approximately 300 milliseconds later [118]. This is otherwise known as the 'oddball' paradigm [123]. This technique is a typical example of Peak Picking methodologies that depend on Event Related Potentials (*ERPs*) and is further covered in Section 2.3.

2.2 TYPES OF BCI RECORDING

Neural data can be obtained in a number of ways, each with advantages and disadvantages. One of the most distinguishing features that sets them apart is the invasiveness of the technique. This section is structured according to this premise and will begin with highly invasive methods, and move through to non-invasive. As the techniques become less invasive, the ability to detect the activity of individual neurons is lost and instead, reliance on the activity of neuronal populations is required. Figure 2.4 demonstrates the level at which each technique detects neural activity.

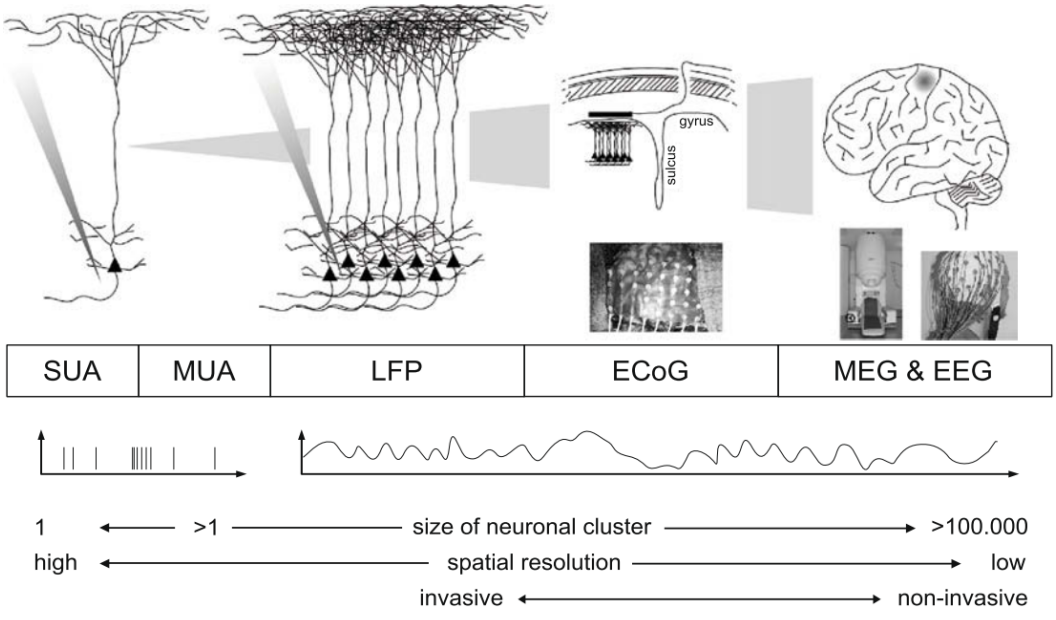


Figure 2.4: Recording techniques in order of invasiveness. Single and Multiple Unit Activity (*SUA* and *MUA*) signals shown to detect discrete firing of individual neurons, as techniques become less invasive, they rely on regions of activity, and therefore detect analogue signals. Diagram taken from [179].

2.2.1 *Invasive Methods*

Invasive methods have the benefit of being able to measure the electrophysiological activity of the neuronal population directly, with high spatial and temporal resolution, but the user must undergo invasive, expensive, and potentially dangerous surgery [185].

2.2.1.1 *Intracortical Electrodes*

Intracortical recordings are known as fully invasive as they perforate the neural matter using small electrodes. This approach comes with all the risks associated with surgical intervention and brings with it the additional risk of the user's body rejecting the foreign object, while offering a substantially richer level of information than that presented by extra-cranial approaches. One of the greatest strengths of gaining this level of detail from the neurons is the ability to match neuronal firings directly to desired movements without the need to train users with arbitrary mental associations [179]. There are 3 variations of intracortical methods; intra cellular, in which individual action potentials are recorded, inter cellular, where local action potentials are recorded, and *Local Field Potentials (LFP)* in which all local potentials are detected.

Intracortical recordings focus on deriving the signal of a single neuron's action potential via the insertion of an electrode, often within a small glass enclosure. Allowing an axon to grow through it creates an isolated environment due to the high resistance of the glass and provides an accurate signal with a very high *Signal-to-Noise Ratio (SNR)* [179]. This form of equipment means that only a few signals can be recorded due to size and invasiveness, but it has been shown by Scott [157] that individual neurons in the sensorimotor cortex can convey information such as position and velocity. While intracellular recordings demonstrate substantial potential, relying on data from such a small subset of relatively random neurons has inherent restraints.

Intercellular electrodes can be used to detect the action potentials within 100 μm of the electrode tip which can be individually identified, and this is known as the Single Unit Activity [179]. An alternative to this is to use Multi Unit Activity; by utilising the filtered, but unsorted, higher frequency signal

and using the averaged action potentials of the local neural population, it is possible to reduce the computational demand and also extend the viable recording distance from the tip [29]. Using this form of signal is a relatively new development in the field but it has already yielded interesting results allowing three dimensional control of a robotic arm with a 91.6% degree of accuracy [129]. With a spatial resolution of 100 μm and a temporal resolution between 50 and 100 Hz, intercellular recordings are undoubtedly the best candidates for a natural brain-computer interface [99].

Another extracellular recording is the local field potential (*LFP*). This consists of the lower frequencies (<250/300 Hz) of the recorded data and is composed of the local membrane currents, resulting in an analogue signal rather than the discrete spiking signals described previously [123]. These signals tend to be outperformed by as little as 12 SU recordings, but are much more robust due to being less reliant on spatial stability (electrode drift is a serious hindrance in SUAs). Two dimensional cursor control has been demonstrated using LFPs but of greater interest is their correlation with Gamma frequencies detected by electrocochleography (*ECoG*). These information rich bands are thought to be a reflection of underlying action potentials, suggesting that LFPs actually contain a significant level of actual neural firing data, hinting at a potentially robust and computationally inexpensive method of BCI control [156].

Invasive methods appear to be the most likely approach for successful, natural, and fluent BCI control but they come with serious risks. While surgery itself risks stroke, haemorrhages and anaesthesia complications, the electrodes are also subject to immune responses. After an electrode has been implanted, inflammation is to be expected. This inflammation typically subsides in a week, but chronic inflammation has been known to occur; tissue death around the implant is a severe, but uncommon occurrence [156]. Less rarely however, nerve cells known as glia have been observed to encase the electrode, shielding it from the surrounding currents, effectively disabling it from further use [89]. This is further compounded by the observed behaviour of neural circuits migrating away from the foreign object. This currently results in the need for periodic recalibration of the electrodes but neurotropic mediums are in development that will encourage neural growth in the effected regions [123].

It should be noted that, while complications are possible with this technique, over 25,000 individuals have received such implants with few negative results and 80% of their original functionality after one year of operation. Indeed, there have been some cases lasting over 7 years and still retaining usable levels of performance [44].

2.2.1.2 *Electrocorticography (ECoG)*

Electrocorticography is known as a minimally invasive technique; while it is intracranial, it does not perforate the neural tissue itself and results in a low infection potential. It is performed by removing a portion of the skull and placing a grid of electrodes on the surface of the brain, just below the dura [75]. The recordings from this grid are very similar to that of *electroencephalography (EEG)*, but are not filtered by the skull. This results in a better, wider range of frequencies (a five fold increase due to the lack of the natural low pass filter), a significantly better spatial resolution, (24 fold due to the proximity of the sensors to the actual neural activity), and a substantially better SNR due to less noise from sources such as electromyography (*EMG*) [129]. As this method is relatively new and requires surgical intervention, the majority of the data currently used results from patients with severe epilepsy [75]. Using ECoG, brain regions that trigger seizures can be identified but while some patients have been waiting for this form of data collection, BCI researchers were allowed to investigate. This typically results in limited session times within a window of 4 to 5 days but long term use has been explored in monkeys and suggests that, even after months of use, accuracy levels remain high and no recalibration is required. Even within such short time periods, it was demonstrated that users could gain a reasonable level on control over a BCI, controlling multi-dimensional cursors [123]. Perhaps more notably, Chao [37] demonstrated that asynchronous control of a prosthetic limb is possible in monkeys. This establishes that directional information can also be acquired from this method, and that it has a viable future, especially if methods of implantation can be improved to utilise smaller craniotomies.

2.2.2 *Non-Invasive*

To classify as a non-invasive method, the technique must not go beyond the epidermis of the user [42] which leads to a number of inherent advantages; primarily, the lack of need for surgical intervention, but other advantages should not be underestimated: ease of removal, cost of equipment, ease of replacement and maintenance, and the speed with which it can be deployed.

2.2.2.1 *Magnetoencephalography (MEG)*

Rather than detecting the electrical fields generated by neurons, magnetoencephalography identifies the magnetic fields created by intracellular currents in dendrites. Due to the large number of different signals being presented to the sensors on the scalp, extensive hardware is required, and only SQUIDs (superconducting quantum interference devices) are capable of the task [178]. An inherent issue with all superconductors however, is the need for temperatures close to absolute zero to function; creating a severe issue for BCI applications. The resulting setup requires liquid helium, stored within a Dewar chamber inside magnetically shielded room to protect against external sources of noise [123] creating a non-portable and a prohibitively expensive solution. However, even with these shortcomings, MEG still stands to provide valuable insight into brain activity; although it is an extra cranial method that records relatively large neuronal populations, it has an excellent temporal resolution of around 1ms (comparable to intracranial electrodes), advances in multiple coil implementations greatly increase the spatial resolution, it is less susceptible to the filtering effect of the skull than electrical fields, and is significantly better at detecting primary over secondary sources than EEG [163].

2.2.2.2 *Functional Magnetic Resonance Imaging (FMRI)*

Functional Magnetic Resonance Imaging (*fMRI*) detects the changes in the magnetic properties of haemoglobin in the blood vessels surrounding the neural tissue. When there is increased activity in an area of the brain, the demand for oxygen also increases, causing the flow of blood containing a high concentration of oxyhemoglobin (diamagnetic) to the area and causing

the blood to become paramagnetic after oxygen is extracted. This has been termed blood oxygenation level dependent (BOLD) imaging [149]. This method shares the EEG's lack of invasiveness and has the added advantage of an excellent spatial resolution over the entire brain. However, this is quickly negated by its reliance on indirect neural activity; due to this technique's inherent need to observe the after effects, rather than the direct action, of the neurons, the temporal resolution suffers a delay of at least 1 to 2 seconds at the most basic level. Eklund, Andersson, Ohlsson, Ynnerman and Knutsson [46] implemented a real-time BCI speller using fMRI and reported correct identifications of actions within the 87-90% region, but required in excess of 40 seconds per letter. When this delay is combined with the need for a large piece of nonportable equipment, fMRI is unlikely to serve as a practical BCI, but it will still prove an invaluable technique to decide where to place sensors for other interfaces.

2.2.2.3 *Near Infrared Spectroscopy (NIRS)*

NIRS is an optical technique that produces intense infrared light using an LED placed directly on the scalp. This light penetrates 1-3 cm and reflected light is detected by a photodiode (also placed on the scalp [124]. Like fMRI, NIRS relies on BOLD responses and varying blood flow, but does not require a stationary patient in a shielded room. Instead, the required equipment is comparable to EEG, but without the need for gels, which carries with it the possibility of creating a personal BCI. As this is a relatively new technique, a number of unresolved issues are still quite prohibitive; it is limited to the outer cortical layer, its spatial limit is around 1 cm, low bit rate, haemodynamic response delays, and difficulty in making a clean connection due to obstacles like hair [123].

2.2.2.4 *Electroencephalography (EEG)*

EEG is the prevalent method for implementation of BCIs for a number of reasons, but perhaps the most important comes in the form of high temporal resolutions and its almost riskless (non-invasive) application [129]. Added to this, the low cost of the equipment (when compared to other systems such as

fMRI) and potential for portability [123], renders the rapid expansion of its application in recent years unsurprising. EEG involves placing electrodes on the scalp of the user. A minimum of 3 electrodes (more typically in the region of 64 or 128) are required for this process; one ground, one reference and one active stream. This is due to EEG not just merely measuring the potentials at each pad, but the difference between them by removing the common-mode potential [163] via comparisons with the reference signal. These electrodes are placed in a standardised pattern known as the 10-20 system [74], which gives a more consistent and predictable performance, further increasing the validity of the research field.

EEG systems also have substantial downsides. One problem is that the impedance between the scalp and the electrode must be sufficiently low to allow the detection of a wide spectrum of wavelengths, but techniques for reducing resistance often introduce further complication: 'wet' electrodes rely on the introduction of a gel or saline solution between the contact and the scalp. This dries out, limiting the recording time to an hour. 'Dry' electrodes are in development but often rely on amplifiers which are susceptible to environmental noise from sources such as nearby power lines [163]. One of the greatest challenges facing EEG systems is their low signal to noise ratio, with background noise being inherited from electrocardiography (electrical activity of the heart, *ECG*), electromyography (electrical activity of the muscles, *EMG*), and electrooculography (electrical activity of the eye, *EOG*). These issues are further compounded by the low spatial resolution; the distance between the neural surface and the electrodes is naturally a minimum of 2-3 cm due to the cranium [179] which results in the detection of an 'area', rather than the ability to detect the activity of individual neurons. Due to the rate of decay of the signal power, Srinivasan [163] projected that almost $6cm^2$ of neural tissue must be activated for a measurable signal to be detected, which indicates the activity in the region of 100 million neurons [156]. This number within such an area would suggest that EEG may lack the ability to convey the fine detail needed for the interpretation of more natural cognitive processes, but instead will need to rely on Evoked Related Potentials (discussed later).

It should also be noted that the skull acts as a natural low pass filter, resulting in detections primarily within the 5-70 Hz range, while the actual frequency generated lies between 5 Hz and 10 kHz. The lower bandings pass though the barrier relatively successfully, however the richer higher bands are severely hindered, removing potentially invaluable control information for the use of BCIs in prosthetics [123]. The actual equipment itself also requires some refinement; set-up of the electrodes can be time consuming and cosmetically unattractive, but companies like Neurosky are currently bringing consumer grade EEG devices to market with a calibration time measured in tens of seconds. These consumer products are still immature, with significantly poorer signals than those of their lab counterparts and are currently unsuitable for complex BCI control [57].

With these issues in mind, the practicality of an EEG-based recording system has proven sufficiently attractive to make it, by some measure, the most common form of BCI in research [155], and the only form to venture into the commercial market. While it was initially believed to be limited to simple binary controls, Wolpaw and McFarland [184] demonstrated two-dimensional controls are possible, later to be surpassed by McFarland, Sarnacki and Wolpaw's achievement of three-dimensional control [117]. Other successful EEG-BCIs include control of communication devices [76, 84, 139, 187, 188] and restoration of movement to paralysed limbs through detection of associated neural activity and direct stimulus of the limb [137]. This demonstrates that, with refinement and further research, EEG-based recording devices have the potential to fulfil the requirements of many BCI applications, but solutions to their shortcomings must be sought.

2.2.2.5 *Summary*

Due to the properties of the skull, non-invasive (extracranial) BCIs will never have access to the intricate information flows within the brain, especially those contained within the higher frequencies. This limits the dexterity of any potential external effector, but ease of use, application, low cost and relative safety in comparison to invasive methods, will ensure that research will continue in the field, and potentially into field applications. While EEG-

Figure 2.5: Photograph of a NeuroScan 64-electrode EEG cap. Image courtesy of [50]

based BCIs are not ideal, they currently appear to be the most feasible and attractive method of getting a product ready for commercial deployment.

2.3 NON-INVASIVE BCI PARADIGMS

As non-invasive techniques cannot pass through the skull, the cranium behaves as a natural Low Pass filter. While this has the advantage of reducing some noise artefacts, it also introduces others, decreases spatial resolution, and removes higher frequencies that are potentially much richer in information than their lower counterparts. To compensate for these disadvantages, it is possible to observe Field potentials; the summation of potentials (e.g. axonal, synaptic, action) in a relatively small region. While this does not necessarily convey the same level of detail as monitoring the firing of individual neurons, it does present observable changes. These Event Related Potentials (*ERPs*) come in two forms; endogenous, internal mediation of potentials or rhythms, and exogenous, requiring an evoked response being triggered by an external stimulus [11]. The following section describes the methods which can be used to create these observable changes, with particular focus given to the paradigms used in this thesis: *sensorimotor* and *P300* based approaches.

2.3.1 *Sensorimotor*

Sensorimotor brain rhythm changes occupy the μ (8-12 Hz) and β (13-30 Hz) bands discussed previously in Section 2.1.3.1. Observable changes in these bandwidths are seen in relation to bodily movements, but do not require actual movement to occur [135]. These changes consist of two modulations; event-related desynchronisation (*ERD*), decreases in amplitude, and event-related synchronisation (*ERS*), increases in amplitude, before and after movement. To trigger the modulations for BCI, users are asked to imagine physical movements, but this can be problematic as users will often visualise movement-associated imagery instead, which elicits different activation patterns. To counteract this, user training is often required [124]. Another form of sensorimotor BCI paradigm is Movement Related Potentials (MRP). MRPs consist of changes in the lowest bandwidths (<8 Hz) beginning up to 1.5 seconds before a movement. While these potentials carry directional information regarding the movements of the user, the bit rate is very low, often requiring averaged signals over repeated trials [179].

2.3.2 *Slow Cortical Potentials (SCP)*

SCPs are voltage shifts around the 1 Hz frequencies. Negative shifts represent increased neuronal activity, while positive shifts represent a decrease, both of which last between 300 milliseconds and several seconds [56]. As with sensorimotor potentials, Slow Cortical Potentials have been shown to be present in both able and less-able bodied individuals but SCP require user training. This training can be affected by a number of factors, such as the user's pain levels, mental state, relationship with trainer, and even after several months of practice, can only achieve accuracy rates within the 70-80% range [123]. Endogenous methods are more elaborate than exogenous in that they purposefully evoke neural reactions from stimuli, and simply measure the responses.

2.3.3 *Visually Evoked Potentials (VEP)*

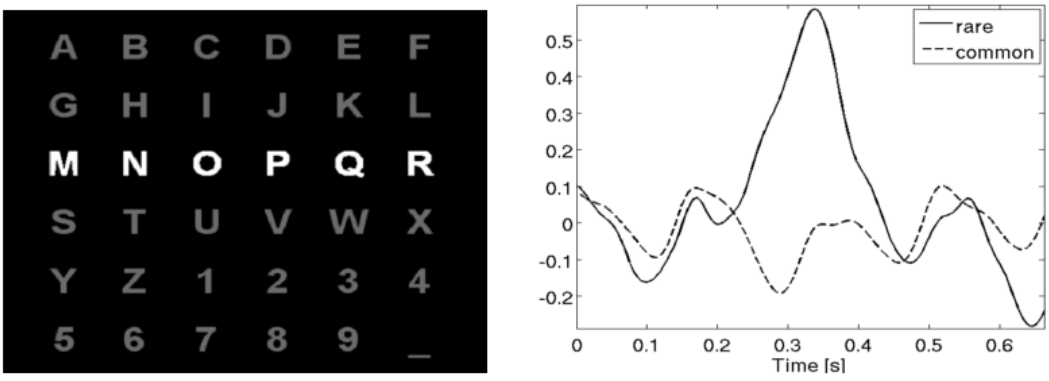
Visual Evoked Potentials (*VEPs*) are triggered when a user is presented with a visual stimulus, with the magnitude of the response greatly increasing if the stimulus is brought closer to the centre of vision and additional attention is given to it [181]. VEPs come in two primary forms: transient, and steady-state. Transient VEPS (*TVEPS*) tend to appear in response to visual changes in frequencies lower than 6 Hz, and can be triggered using flashing lights, a brief appearance of a pattern or the reversal of an existing one [123]. The measurement of this form of evoked potential is easily contaminated by EMG and EOG sources, and is rarely chosen over its counterpart, Steady-State VEPs. SSVEPs are a very common form of BCI control in the literature due to their high SNR, and classified according to the modulation of the stimuli presentation; time, frequency and order of stimulus presentation [123]. To control a SSVEP BCI, such as a speller, the user stares at the desired stimulus and its frequency modulates the frequency of the response detected, indicating the selected input. While this approach has the advantage of little or no user training, it does require a user to give complete visual focus to a stimulus (typically a screen) disqualifying its use as a natural BCI interface for controlling devices such as prosthetic limbs, and inoperable for sufferers of neuromuscular diseases that lack the ability to alter their gaze [129].

2.3.4 *P300*

Another common endogenous BCI control potential is the P300 response. If a user is observing a number of stimuli flashing seemingly randomly, focusing on one will trigger a secondary modulation in the field potential 300 milli-seconds later. This is known as an 'odd-ball' response as it appears to increase in amplitude according to how unlikely the stimulus is [123]. A problem with this method is that the P300 response is measured relatively to the responses of the non-attended or expected stimulus meaning that a number of stimuli must be presented over multiple runs (with appropriate gaps between epochs to prevent overlapping) and an average difference calculated. This substantially

decreases the maximum bit-rate of the technique but it does not require the user to be as attentive as SSVEPs, and also shows some success using auditory stimuli [84]. It should be noted however, that the amplitude of this ERP appears to be directly proportional to how often it has been shown, exemplifying the necessity to achieve reliable classifications with minimal data [119].

One of the most common applications of the P300 signal in BCI is the P300 Speller. The first account of this paradigm was in [47], in which a grid of 6x6 alphanumeric characters were displayed on screen, as seen in Figure 2.6. A user was asked to focus on a single character, and each row and column was flashed in a random sequence. Other variations of this technique have been developed, including a Single Character Speller. These involve each character being flashed alone, which requires approximately twice the time frame for a single round of the speller. This disadvantage is offset by much larger P300 responses, but still proves less accurate than the Row Column paradigm initially described [59]. This may be due to a number of issues inherent to the design of this kind of BCI; P300 responses can be missed if stimuli are presented within 500ms of each other, and are lessened if they do occur [147]. This effect is worsened when similar stimuli create the 'Crowding Effect', making the target less novel and reducing the response[48].



(a) (b) P300 evoked potentials

Figure 2.6: Visual stimuli presented to the user in (a), where each column and row are flashed randomly. When the column or row containing the user's target letter flashes, the P300 wave in (b) is observed in the signal, approximately 300 miliseconds later. Diagram taken from [148].

2.3.5 *Summary*

Two of the most important paradigms in non-invasive BCI are *Sensorimotor Rhythms* and *P300 Spellers*. Sensorimotor Rhythms can be modulated by the user, without the need for external stimulation. This renders them viable options for control of BCI devices that are intended for use in 'passive' activities, such as prosthetic limbs. P300 Spellers require a user to attend a stimulus. While a stimulus is attended, the user will be unable to accomplish other tasks, but this method allows the user to select from a wider variety of outputs. This results in a highly effective mode of communication. Datasets using these two paradigms were the focus of this thesis, and are described in Section 4.1.

2.4 DATA PREPROCESSING

After the signal is acquired from the brain, there are a number of simple techniques commonly applied to reduce artefacts within the data, before presentation to the classifier. This step is especially important when dealing with BCI signals, as they are notoriously noisy.

2.4.1 *Referencing*

Electrode referencing uses an electrode in a region near to the brain, which contains as little neural activity as possible. This allows detection of potential sources of noise from the eyes and muscles, that can be subtracted from the electrodes that are intended for neural recording. This electrode is referred to as a *common reference*, and it is normally attached to the nose, mastoids, or earlobes. Alternatives to this approach are *average referencing* (subtraction of the average activity over all electrodes), and *current source density* (*CSD*)(a method which involves subtracting the average summed signal from only the electrodes that surround the one in question). These methods, especially CSD, have been shown to increase performance in some cases. However, recordings from non-expert users may be problematic as they require a large number of electrodes [14] to be evenly spaced on a two dimensional plane [4].

2.4.2 *Frequency Filtering*

As the skull behaves as a natural bandpass filter, frequencies of above 30 Hz are likely to originate mainly from external sources. Due to this, it is common practice to set a maximum bandwidth of 30 Hz, and to almost always remove the gamma band, despite it being the richest band for motor decoding [123]. On the other hand, the lower frequencies are prone to contamination from EOG and eye-blinks. In sensorimotor tasks, it is generally accepted that a bandpass filter of between 8 and 30 Hz is effective to capture both the μ and the β bands in which the desired information resides [106].

2.4.3 *Normalisation*

It is common for different brain regions to generate markedly different amplitudes, which can mislead classifiers into over weighting this aspect and failing to discriminate between the more subtle signal dynamics. Normalisation of the signal is common by subtraction of the mean, and division by variance [93]. This must be done with care: amplitude (particularly in P300 applications) is often a powerful discriminatory feature.

2.4.4 *Artefact Removal*

Artefacts from EOG and EMG are highly problematic within BCI systems. Removal of these is often done manually in studies within neuroscience, but this is impossible for Brain Computer Interfaces, in which near spontaneous classifications are required. As seen in figure 2.7, the difficulty in identifying unwanted artefacts varies with the source. A technique known as *winsorizing* is often used to remove outliers such as those caused by eye blinks. To do this, the outliers are replaced by a value representing a predefined selected percentile of the data. As seen in Figure 2.7, a blink is quite easily identified, but electrical activity from muscle sources (EMG) is much more subtle. This problem comes from EMG signals being broadband, which includes the sensorimotor bands, β and μ. Spatial filtering (see Section 2.5.1) techniques can help to alleviate

this problem, as EMG occurs primarily in the peripheral regions of the skull, with the central regions generally being less affected [93].

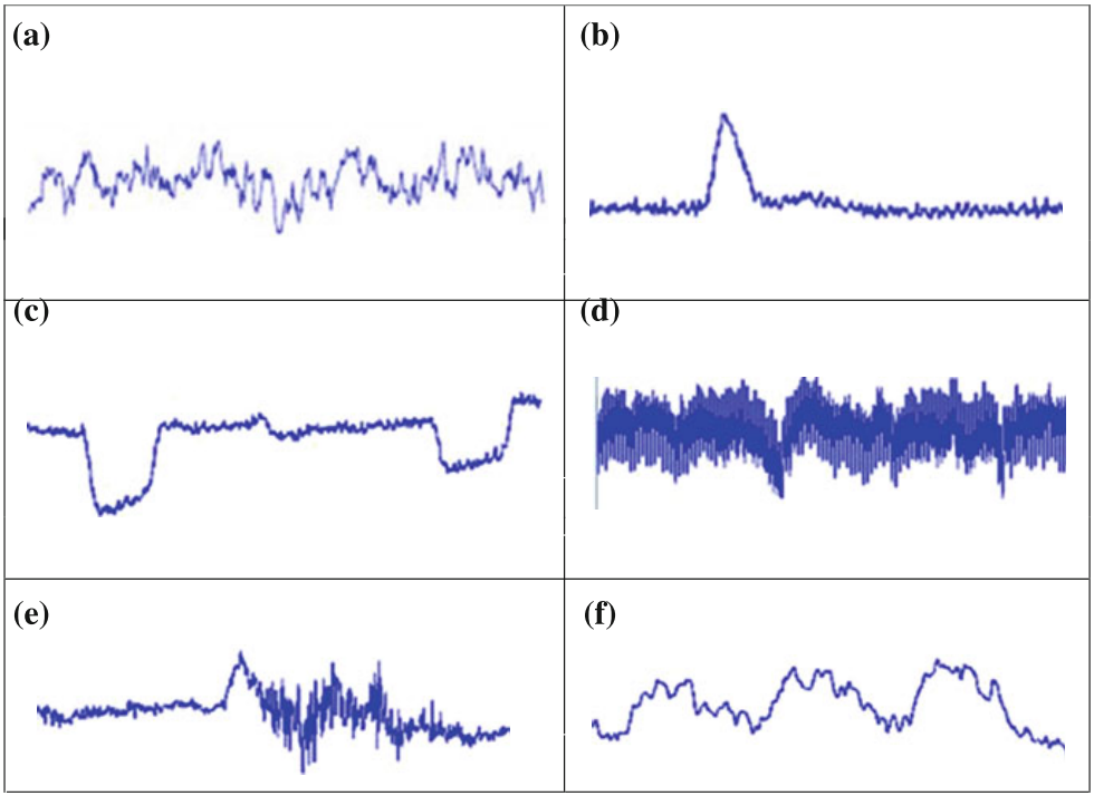


Figure 2.7: Examples of different noise sources. (a) EEG signal with no obvious noise, (b) blink, (c) eye movement (EOG), (d) 50 Hz interference, (e) Muscle movement (EMG), (f) Heart beat (ECG) [16]

2.5 TYPES OF FEATURES

Features are the underlying characteristics of a signal; sequences or simultaneous occurrence of which create a pattern and are indicative of an underlying mental process. The identification of these is performed by a classifier. It has been demonstrated that, while the raw signal is classifiable, it is of substantial benefit to undergo an enhancement phase known as *Feature Extraction* [105]. Feature extraction typically creates a new search space by decomposing the recordings in terms of time or frequency. This serves two primary purposes; dimensionality reduction and to preserve and enhance the relevant properties of the signal, reducing the computational demand while increasing the classification accuracy. This transformation of the data can affect the representation of the data, change the search space, and alter the performance of different classifiers. Extracted features are the inputs to the machine learning phase of the BCI system.

2.5.1 *Time Domain Features*

Time domain features are predominantly used for BCI paradigms that relate to temporal changes, such as P300 responses. In these cases raw signals can be sufficient, but more complex data transforms can be performed [106].

2.5.1.1 *Autoregressive Modelling (AR)*

To create an AR model, a weighted linear combination is created by combining a number of previous samples in order to predict future input samples [151]). Features are created as the weights a_i of the autoregressive parameters that are multiplied by the signal $X(t)$ when measured at time point t, while taking into account a noise term (E_t).

$$X(t) = \sum_{i=1}^{P} a_i X(t-i) + E_t \tag{2.1}$$

While AR models tend to provide good frequency resolutions, especially on short samples, selection of an incorrect model order P can cause loss in spectral detail or false spikes in the spectrum [161].

2.5.1.2 *Hjorth Parameters*

Hjorth parameters seek to quantify a signal at different timepoints in terms of *activity*, *mobility* and *complexity*. *Activity* is defined as the mean power of the signal (Equation 2.2), *Mobility* is defined as the mean frequency of the signal (Equation 2.3), and *Complexity* is defined as the deviation from a sine wave (Equation 2.4) [4].

$$\text{Activity}(X(t)) = \text{VAR}(X(t)) \tag{2.2}$$

$$\text{Mobility}(X(t)) = \sqrt{\frac{\text{Activity}\left(\frac{dX(t)}{dt}\right)}{\text{Activity}(X(t))}} \tag{2.3}$$

$$\text{Complexity}(X(t)) = \frac{\text{Mobility}\left(\frac{dX(t)}{dt}\right)}{\text{Mobility}(X(t))} \tag{2.4}$$

This form of feature is typically seen in motor imagery paradigms, but have been shown to be capable in emotional classifications as well [177].

2.5.2 *Frequency Domain Features*

To extract frequency domain features, the signal recorded at each electrode is divided into time epochs, which is then decomposed into separate bandwidths before processing [145]. These features are widespread in literature as they are readily applied, quickly computed, and easily interpreted.

2.5.2.1 *Bandpowers*

To track changes in modulations within certain frequencies, the signal may be bandpassed according to the band of interest. The signal should then be squared to ensure only positive values remain, and the peaks smoothed via low-pass filtering or integration [93]. As described in Section 2.3.1, the frequencies of most interest in sensorimotor BCIs are within the α and β

frequencies, but the more precise frequency within each of these bands is user-dependent [135]. Fast Fourier Transform methods are typically more effective when there are multiple frequency bands of interest.

2.5.2.2 *Power Spectral Density (PSD)*

Power Spectral Density is a feature extracted from the frequency domain most often found in literature. This is typically estimated using an average of the minimum and maximum densities returned by Welch's method. To do this, the signal is divided into segments and the following steps are then applied [127];

For each sample n in the signal x, divide it into K overlapping sections of length M.

$$x_i[m] = x[m + iD], \quad \begin{matrix} i = 0, ..., K-1 \\ m = 0, ..., M-1 \end{matrix} \tag{2.5}$$

where iD is the data point at the start of the ith sequence. A window is then applied to the section and a periodogram calculated.

$$P_i(f) = \frac{1}{NU} \left| \sum_{n=0}^{N-1} w[m] \cdot x_i[m] e^{-j2\pi fm} \right|^2 \tag{2.6}$$

where $U = \frac{1}{M} \sum_{m=0}^{M-1} w[m]^2$ is a normalisation constant.

Finally, the spectral density can be estimated by averaging the periodograms calculated from the K sections.

$$P^w(f) = \frac{1}{K} \sum_{i=0}^{K-1} P_i(f) \tag{2.7}$$

By overlapping these windows, it reduces the variance by averaging a number of different periodograms [5]. As shown in Figure 2.8, PSD features dominate the literature, especially that involving sensorimotor data. This is due to its high success rates and proven efficiency across a number of BCI applications [106], with Herman et al providing evidence that it is the most robust method of feature selection for motor imagery [73].

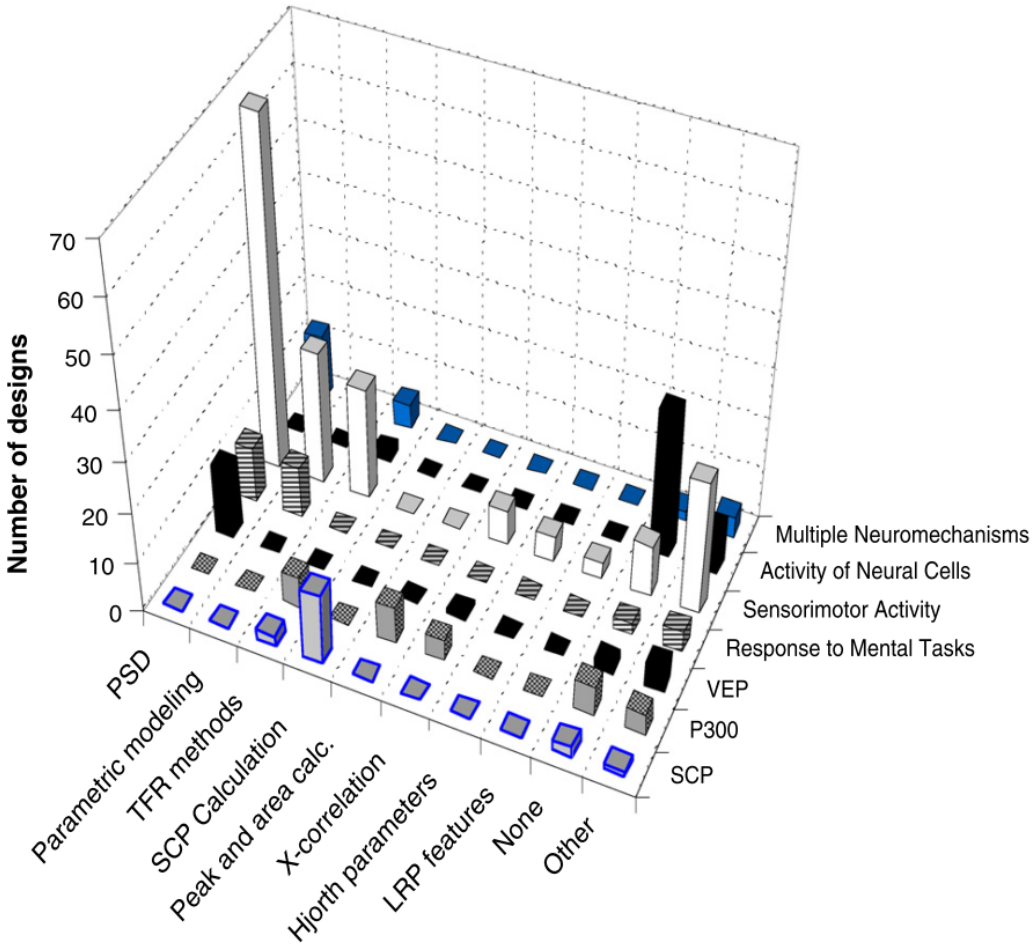


Figure 2.8: Review of Feature Extraction methods reported in [14]

2.5.2.3 *Spatial Filtering*

Spatial filtering is an especially interesting process as it can be used in a number of ways within the BCI paradigm: as a signal *preprocessing* measure to remove artefacts, *feature extraction* by collapsing the dimensionality of the raw signals, or for *feature selection*. The following methods can be used for these purposes.

PRINCIPAL COMPONENT ANALYSIS (PCA) Principal Component Analysis projects the data onto lower dimensions so that the variables are as uncorrelated as possible [164]. The first principal component explains the largest share of the variance in the dataset, with each subsequent variable explaining the remaining variation. While this is a common approach, it is limited to linear combinations, tends to under perform when compared with other methods for feature selection, and is prone to detecting noise sources

rather than the desired neurological information. That is, if the variance in a channel is predominently explained by environmental noise, such as 60 Hz electrical lines, it will be identified as the principle component [93].

INDEPENDENT COMPONENT ANALYSIS (ICA) *ICA* seeks to create statistically independent variables from the signal; a more difficult task than generating uncorrelated variables by PCA. It assumes that the recordings of neural activity are the accumulation of different, independent processes. It seeks to separate them as a 'cocktail party' problem: focusing on separate sources of information in a noisy signal. This has been shown to be particularly effective in identifying artefacts caused by eye activity [4].

COMMON SPATIAL PATTERNS (CSP) To enhance signals, an approach often seen is Common Spatial Patterns (CSP). This technique is similar to PCA, but takes the predictive labels and spatial information regarding electrodes into account by calculation of a matrix in which class differences are maximised. This results in the possibility of inverting the filtering matrix, to restrospectively discover the physical origins of the neurological signals that best seperate the classes [93]. This technique is very effective when dealing with sensorimotor BCI recordings, but is subject to a number of disadvantages. Among those disadvantages are: high sensitivity to artefacts, requirement for large numbers of electrodes, and identical electrode placement in all samples due to the spatial basis of the technique [4].

2.5.3 *Time-Frequency Domain Features*

In order to retain the advantageous information provided by both temporal and frequency domains, time-frequency features were developed. The most popular of these, is the *Wavelet Transform*.

2.5.3.1 *Wavelet Transforms*

Wavelet Transforms are a spectral estimation technique in which any general function can be expressed as an infinite series of wavelets [5]. Due to

neural recordings being non-stationary, a wavelet transform is potentially more powerful than relying on the signal's power alone as it allows for variable sized time windows. This results in higher resolution in low frequencies by using larger windows, while retaining the ability to use shorter windows for high frequencies [111]. The two most common wavelet transform methods are *Discrete Wavelet Transform (DWT)* and *Wavelet Packet Transform (WPT)*. While WPT provides better high frequency resolutions, the lower computational complexity of DWT is preferred as useful EEG signals rarely exceed 60Hz [54]. After the signal has been transformed, information such as the Relative Wavelet Energy can be extracted and used to form the feature vectors [152].

2.5.4 *Feature Vector Construction*

Feature vectors are constructed in the form of a N_f x N_i matrix where N_f is the number of features and N_i is the number of instances. N_f commonly consists of a concatenation of features created from each electrode [76], but concatenation of features created from different time epochs, frequency bands, and spatial locations is possible [145]. This form of dimensionality reduction creates two dimensional data frames for presentation to the classifier, but draws attention to the need for further dimensionality reduction. For example, a 64 channel EEG recording, with features extracted from 8 frequencies, results in each sample being represented by 512 features.

2.5.5 *Summary*

As evidenced in Figure 4.7, Power Spectral Density is the feature most commonly used in literature for sensorimotor imagery BCI [14]. This is due to a number of reasons, including their demonstrated ability to create more generalisable models [150], and their ability to preserve the distinction between relevant frequency bands and time epochs, providing information of interest to clinicians [166]. Three of our datasets use this paradigm, and it was therefore selected as our Feature Extraction technique in Chapters 5 and 6. A full description is given in Section 4.1.5.

2.6 CLASSIFIERS

Classifiers assign a label to a set of inputs based on prior observed patterns. In this thesis, we focus on BCI tasks in which a discrete classification of user intent is required using supervised learning, rather than problems which involve unsupervised learning. Thus, we are concerned with tasks that require a specific outcome, such as typing, rather than those achieved using clustering techniques more commonly used in diagnostic applications such as epileptic seizure detection [17]. These patterns differ within paradigms in BCI and it is important to make selections based on the characteristics of the problem, while safeguarding against known pitfalls of each technique. Amongst the most important aspects of classifier choice for BCI is the prevention of over-fitting to the noisy, high-dimensional, and small training sets available, while retaining the ability to detect the identifying properties of each class.

This section begins by defining a taxonomy, and then moves on to describe linear classifiers (*Fisher's Linear Discriminate Analysis* - Section 2.6.2 and *Bayesian Linear Discriminate Analysis* - Section 2.6.3) . This continues onto *Support Vector Machines* (Section 2.6.4) as a bridge between linear and non-linear methods, followed by non-linear classifiers (*k Nearest Neighbours* - Section 2.6.5 and *Artificial Neural Networks* - Section 2.6.6).

2.6.1 *Classifier Taxonomy*

For the definitions below, we assume that datasets for classification consist of a set of input vectors X and corresponding labels Y; where X consists of $x_i \in \mathbb{R}^D$, $i \in \{1...N\}$, and $y_i \in \{-1, 1\}$, $C = |Y|$.

Classifiers are defined by four main properties [106]:

1. *Generative/Discriminative* - Generative classifiers learn models for each class, whereas discriminative classifiers discover a means of discrimination between them.

2. *Static/Dynamic* - Static classifiers do not take temporal information into account for classifications, whereas dynamic classifiers can.

3. *Stable/Unstable* - Stable classifiers tend towards a low level of complexity, resulting in small variations in the training set making little difference, whereas the performance of Unstable methods is more heavily impacted by outliers.

4. *Regularized* - the complexity is controlled to prevent overfitting, and protect against outliers.

2.6.2 *Fisher's Linear Discriminant Analysis (FLDA)*

FLDA is one of the most popular classifiers in EEG BCI, largely due to its efficiency, low complexity and general stability when presented with variations in datasets. To perform its classifications, the LDA separates the data into individual hyperplanes representing each class [52], and a feature vector is labelled according to which region it appears. Similarly to PCA, it seeks to explain the variance in the data, but is a *supervised* method, meaning that it takes the class labels into account, as seen in Figure 2.9. This means that it looks for a dimensional transformation that emphasises the differences between classes, rather than those that emphasise the variance.

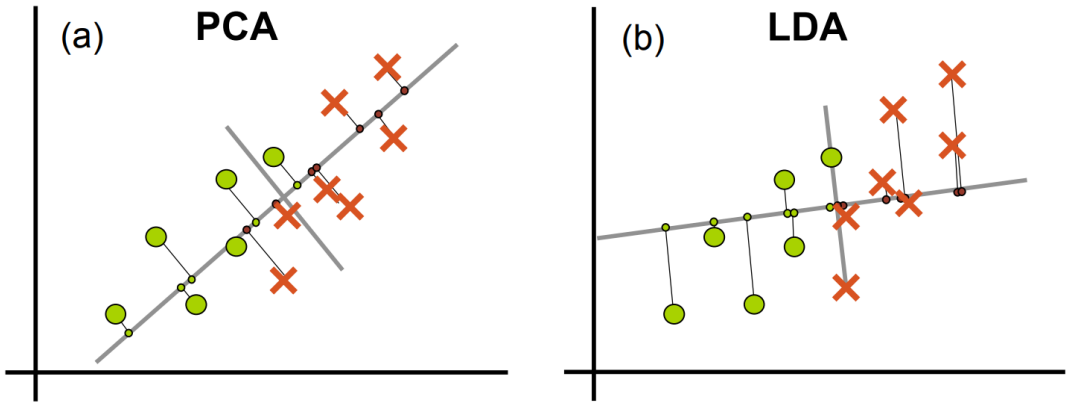


Figure 2.9: Diagrams displaying the differing intents of PCA and LDA by displaying the indended 'line of fit'. As seen in (a) the PCA seeks to explain the variance in the data, but fails to separate the classes. The LDA in (b) finds a different hyperplane than that of the PCA, encouraging a better split. Diagram taken from [41].

The LDA classifier can be expressed as:

$$g(x) = w^T x + w_0 \tag{2.8}$$

where w is a weight vector, x is the input and w_0 is a threshold. The weight vector (w) used for projection of the dimensions onto a lower dimension can be achieved by a number of methods, but FLDA optimises a cost function in the form of a Rayleigh quotient [76]

$$J(w) = \frac{w^T S_B w}{w^T S_W w} \tag{2.9}$$

where S_w is the within-class matrix:

$$S_w = \sum_{k=1}^{2} \sum_{x_i \in C_k} (x_i - m_k)(x_i - m_k)^T \tag{2.10}$$

and S_b is the between-class matrix:

$$S_b = \sum_{k=1}^{2} n_k^c (m_k - m)(m_k - m)^T \tag{2.11}$$

where m_k is the mean of class k, C_k is the training data vectors belonging to class k, and n_k^c is the number of instances in that class [131].

While this method tends to provide good results, it does not have the reported accuracies of other classifiers, often chosen for its ease of use rather than its ability. When dealing with numbers of features that exceed that number of samples, classification accuracy is noticeably reduced [13].

2.6.3 *Bayesian Linear Discriminant Analysis (BLDA)*

An extension of the FLDA approach is Bayesian Linear Discriminant Analysis (*BLDA*). Regularisation is used to address the primary issue with FLDA; overfitting due to noisy, high dimensional data [191]. By using the expectation-maximisation algorithm to optimise the hyperparameters for Bayesian Linear Regression [88], improvements in classification have been made over the FLDA. This renders the BLDA amongst one of the best classifiers in BCI, and reports the best accuracies in a number of P300 speller studies [13], even outperforming the more complex Support Vector Machine (*SVM*) [48]. For

a detailed description for the implementation of BLDA for use in the P300 paradigm, see [76].

2.6.4 *Support Vector Machine (SVM)*

SVMs are well suited to the task of decoding neural signals for BCI as they can detect linear and non-linear relationships between features and classes. They perform their classifications by projecting the dataset onto a higher dimensional space and introducing a hyperplane that maximises its distance from the most difficult to place points either side of the decision boundary, as shown in Figure 2.10.

This is known as the 'margin'. Rather than offer a 'hard decision', a soft margin is applied, allowing some points to be moved across the boundary to the correct class. This regularisation helps to mitigate the effects of outliers, to which a complex classifier such as SVM can be sensitive. To find this hyperplane we solve the minimisation problem:

$$min_{w,b}\frac{1}{2}||w||^2 + c\sum_{i=1}^{N}\xi_i \tag{2.12}$$

where ξ, are slack variables that relax the constraints to create a soft margin, and c is the regularisation parameter to control for model complexity.

SVMs are popular in BCI literature due to fast training times when compared to other approaches like the multi-layer perceptrons [134], high generalisation abilities, resistance to overfitting, and insensitivity to the curse-of-dimensionality [104]. They have been shown to be particularly effective in the classification of motor imagery data [123].

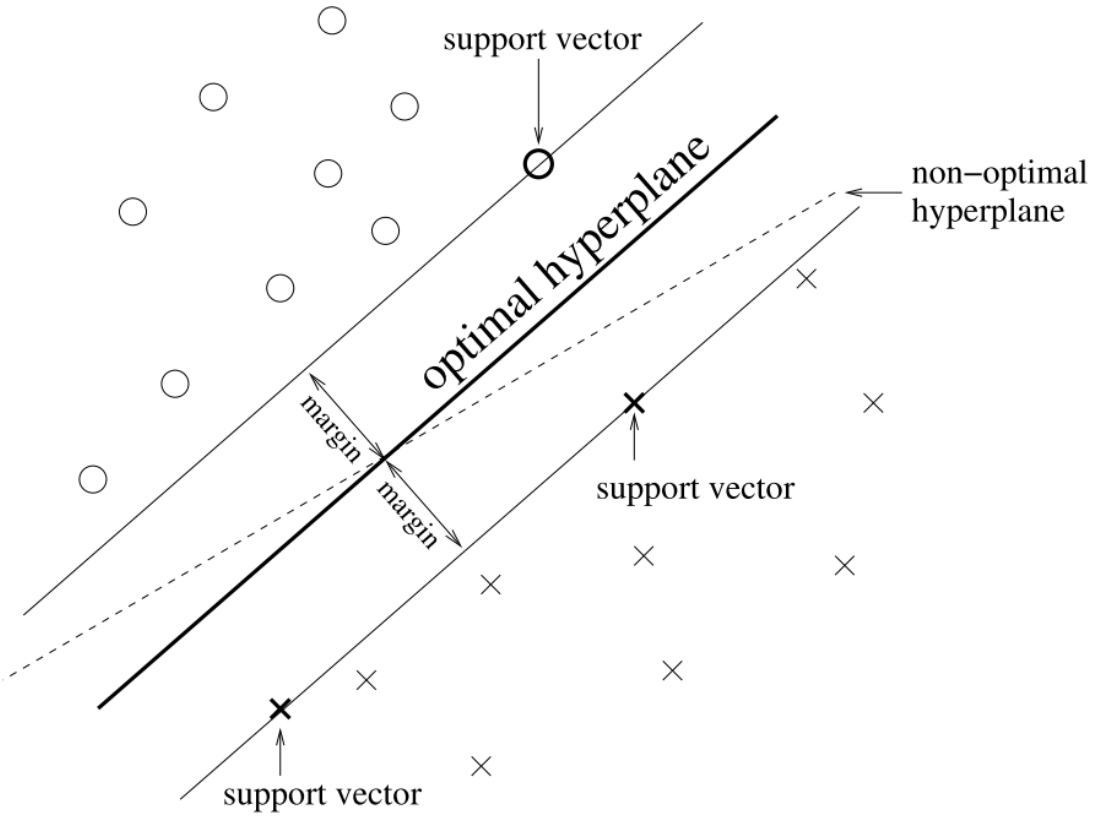


Figure 2.10: A depiction of a Support Vector Machine (*SVM*). A hyperplane is discovered that maximises the margin between the nearest support vectors of each class. Diagram taken from [104].

2.6.5 *k-Nearest Neighbour (KNN)*

Classes typically cluster in the feature space, and k-Nearest-Neighbours exploits this to perform classifications [104]. This involves using a metric to assess the distances between the features of an unlabelled instance, and that of the nearest k examples from a training set. By discovering the k nearest neighbours, misclassification due to outliers can be decreased. Each of the k neighbours are assigned a weight according to their distance as defined by:

$$w^i = \begin{cases} \frac{d^k - d^i}{d^k - d^1} & \text{if } d^k \neq d^1 \\ 1 & \text{if } d^k = d^1 \end{cases} \tag{2.13}$$

where d^i is the distance to the i-th nearest neighbour from the test instance, with d^1 being the nearest, and d^k being the furthest neighbour. The instances within the class that create the largest sum of weights is predicted to be the

test samples class. KNN is a popular classifier in many applications, but is rarely used in BCI. As EEG tends to involve high dimenional datasets [123], technqiues using Euclidian distance measures become inappropriate due to the exapansion of the space [120], leading to the failure of KNN in a number of studies [106].

2.6.6 *Artificial Neural Networks (ANN)*

Artificial Neural Networks are modelled on the structure of neurons in the brain. They are widely applied as they have been shown to be highly generalisable, discovering patterns that traditional statistical approaches struggle with [123]. ANNs consist of a network of interconnected and weighted nodes, where these weights are adjusted during the training process. A training set is presented to the network, and depending on the output, the weights are altered and the process repeated. This is continued until the output labels achieve an acceptable degree of similarity to the input labels.

The most widely used ANN in BCI is the *Multilayer Perceptron (MLP)*. An MLP is created with an input layer, one or more hidden layers, and an output layer of artificial neurons. A network such as this is a universal approximator, in that, with a large enough network, any continuous function can be represented. This results in a classifier which is vulnerable to overtraining; a substantial issue when dealing with datasets with the noisy characteristics of BCI [104]. Despite the need for expert design and regularisation, they have been applied to binary and multiclass problems using a wide range of BCI paradigms, but when compared with the classifiers described previously, typically achieve lower classification accuracies [13].

Part III

LITERATURE

3

CHAPTER 3 - LITERATURE

This chapter discusses literature relating to *Feature Selection* and *Transfer Learning* in *Brain Computer Interfaces*. Feature Selection algorithms are described in order of *Filters*, *Wrappers*, *Embedded*, and *Hybrid Methods*. These sections relate to Chapters 5 and 6. Application of Transfer Learning in BCI is then discussed in relation to Chapter 7.

3.1 FEATURE SELECTION

As discussed in previous chapters, EEG data is subject to a range of noise sources, limited quantities of training data, and substantial numbers of features. These factors expose it to the 'curse of dimensionality'. Reducing this dimensionality through feature extraction is limited in that these techniques often encompass the noise, rather than exclude it. Pruning the information sources can be a useful alternative. By selecting only the most infomative features computational load can be decreased [8], feature relevance increased [60], accuracy improved [4], and the sparsity of the feature space can be greatly reduced [130].

There are three primary divisions of feature selection techniques: Filters, Wrappers and Embedded methods. Filters utilise a ranking method and have no reliance on the classification stage of decoding, instead judging each individual feature on the basis of its relevance. Unlike Filters, Wrappers do not rank features, but instead evaluate subset effectiveness by training the classifier. This allows the classifier to serve as a 'fitness function' which results in a longer training process, but since BCIs are typically trained offline, the increase in classification accuracy takes precedence. Embedded methods involve a feature selection technique incorporated into the classifier. These techniques

have shown some promise in the field, but are limited, in that feature selection and classifier selection cannot be separated.

This chapter details the current state-of-the-art techniques used in the field of Feature Selection for Brain Computer Interfaces. It begins by describing Filter methods, Wrapper Methods, and Embedded Methods. It then extends into the foundation of our work; Filter-Wrapper hybrids.

3.1.1 *Filters*

Filter based methods rank variables according to a criterion, independently of the chosen classifier. These performance measures are traditionally defined within one of four categories: dependency (correlation), consistency, distance, and information measures [190]. More details on each of the categories are given below. The advantages of such techniques tend to be that they are typically less computationally expensive, simpler to implement, and resulting feature subsets are more generalisable as they are not tied to a specific classifier [160]. That being said, they lack the ability to exploit specific characteristics of the machine learning algorithms intended for use, and therefore rarely obtain the highest classification accuracies.

3.1.1.1 *Dependency (Correlation)*

A reasonable assumption when selecting features, is that a correlation between a feature and the class label is indicative of a 'good' feature. *Correlation-Based Feature Selection (CFS)* was introduced in [62], where features were selected on their correlation with the class labels, while also ensuring low correlation with each other. This reduces redundancy in feature subsets, a known issue in BCI feature selection [91]. Sen et al compared *Fast Correlation Based Filter (FCBF)* to *Minimum Redundancy Maximum Relevance (mRmR)* and *Fisher Score Algorithm (FS)* [158]. The authors used a two tier approach in which features were first eliminated if they failed to reach a given threshold of correlation with the class labels, and then redundant remaining features were removed by evaluating their correlation with each other.

3.1.1.2 *Consistency*

Consistency measures were introduced in [101] in which feature subsets were chosen based on the distance to their *Consistent State* (having the same values across instances for each class label). Ranking variables based on their correlations with a class label can neglect interaction effects between them, a factor that can prove important in classification tasks. Instead, Consistency of a subset is determined by evaluating the difference between variables and their class, being declared 'Inconsistent' if their only difference is the class label [160]. In [141] a greedy Consistency based algorithm was found to increase the accuracy of a motor-imagery BCI task while reducing the size of the feature set, but failed to find better solutions than other Filter methods.

3.1.1.3 *Distance*

Distance measures focus on increasing the separability between classes [190]. A popular example is the *Relief algorithm*, which engages in quality estimation of each feature based on its locality in the search space. Each instance tracks its two nearest neighbours that are a part of the same class (nearest hit), and are a part of a different class (nearest miss) [153]. To extend the classification abilities of this algorithm beyond 2 classes and increase proficiency with noisy datasets, ReliefF was developed, in which k nearest neighbours were sought, rather than just 2. The weakness of this approach however, is that the k value must be explored: if $k = 1$, noise within the data will cause reliability issues; if k is too high, an appropriate selection will fail to occur [145]. The performance of ReliefF in literature is somewhat inconsistent: In [91], ReliefF was found to under perform in comparison to other filter techniques that rely on Consistency and Information Theory, while [141] found it to perform better than the Mutual Information based technique, mRMR.

3.1.1.4 *Information Theory*

Information Theory has been shown to be a promising avenue for providing ranking criteria for Filter-based Feature Selection in BCI. A simple *maxRel* based approach achieved higher average accuracies than the *Filter Bank Common Spa-*

tial Pattern (FBCSP) Filter and FBCSP Wrapper in [61]. In emotional recognition BCIs, [12] demonstrated that *Minimum-Redundancy-Maximum-Relevance* can outperform more complex wrapper approaches such as the Genetic Algorithm-SVM (GA-SVM) in both accuracy, and dimensionality reduction. Similarly, in [115], *minimum Redundancy Maximum Relevance Feature Selection (mRMR)* was found to be slightly better than Relief, and statistically much better than CFS, PCA and *Minimal Redundancy*, achieving an increase in the region of 15% over the unfiltered feature set.

In [143] six methods were compared: CFS, ReliefF, Consistency, mRmR, C4.5, and a wrapper approach using a *Genetic Algorithm (GA)*.

The most stable accuracies were found by ReliefF and mRMR; while the highest accuracies and smallest subsets were returned by the GA. However, the mRMR performed favourably to the other filter approaches, and not far behind the GA. M*utual Information Best Individual Feature Selection (MIBIFS)* became popular after being used in the winning entry of Berlin BCI Competition IV [170]. In this work, the most relevant frequency bands were selected by MIBIFS: a simple ranking of features according to their mutual information with the label, and selecting a predefined number. A similar technique was used in [105], [126] , [63] , and in [49], where it was compared to the Mutual Information-based Rough Set Reduction (MIRSR) algorithm, a technique which uses mutual information to select highly relevant features, while using rough set theory's 'knowledge reduction' to control for redundancy. In a subsequent study, [10] used a slightly more advanced, iterative form of MIBIFS. To select salient channels, the channel that shared the highest mutual information with the label was selected. This channel was then concatenated with each remaining channel, and the channel with the highest additional mutual information was selected. This was repeated until no more increases from additional channels was possible. A further advancement on this can be seen in [58] in which features are added until no further increase can be found. A known issue with this variety of approaches is that, when new features are added to the solution, they can render earlier features redundant. To combat this, a backwards step is implemented to remove features from the currently selected subset that can increase the mutual information with the label (relevance). An intensive review

and comparison of information theoretic approaches for motor-imagery is available in [114].

3.1.2 *Wrappers*

Wrappers in the form of *Evolutionary Algorithms (EA)* have proven highly successful in the feature selection field [27, 36]. The typical approach is to use classifier accuracy as the fitness function: the EA begins by generating solutions and splitting the training set into two subsets. The classifier is then trained using the first subset, and its ability to correctly identify the labels of the second subset is used to derive the solution quality. This gives wrappers an advantage over filters: nuances in the data important for the classifier are taken into account during the subset selection. This can be seen in performance comparisons between less sophisiticated statistical dimensionality reduction techniques performing similarly [130], while being easily outperformed by the simplest of wrapper methods (sequential selection) [144].

3.1.2.1 *Sequential Selection*

Sequential selection can be implemented in two ways; *Sequential Forward Search (SFS)* and *Sequential Backward Search (SBS)*. SFS starts with one random feature, adding another, and evaluating the new subset, accepting if it improved. SBS starts with the entire search space, removing one feature, and evaluating, ensuring that there has not been a significant negative impact to the solution fitness. As the search space is often too large to attempt evaluations containing all potential features, SBS is uncommon. One of the main problems in SFS is the rigidity of its solutions: due to the correlations between features in the domain discussed in this thesis, the information provided by a particularly strong feature may be contained between several lower ranking features. That is, after adding a larger number of features, some of the first selected may prove redundant. SFS does not have a method of removing these redundant features. To overcome this, a deviation of the algorithm was developed called Sequential Forward Floating Search based on the principle Plus-L-Take-Away-R [123]. In this, each step involves removing previous features (typically at random)

while adding to the overall feature count. Although Sequential Selection has been demonstrated to be an effective method, it is consistently outperformed in Feature Selection by its counterparts GA and PSO as it lacks their ability to move around the search space and avoid becomming trapped in local optima [145].

3.1.2.2 *Particle Swarm Optimisation*

As with Sequential Selection, *Particle Swarm Optimisation (PSO)* iteratively attempts to find the best solution, but utilising a much more elaborate method. In PSO, a population of candidate solutions are created called a 'swarm'. This 'swarm' consists of individuals that move throughout the search space, eventually clustering around optima. It achieves this by having each particle keep a record of the highest quality solution it has encountered, and through each iteration, the particle's trajectory is accelerated towards that point in the search space [6]. There are a number of variations of this technique, the most prominent being the neighbourhood version in which particles communicate with each other, accelerating towards the best solution in the neighbourhood. PSO has been deployed in feature reduction in a number of studies; one notable experiment was carried out by Jin et al. [81] in which a variation known as Discrete Particle Swarm Optimisation was used for electrode selection. In a comparison against F-Score (a comparative technique) DPSO achieved an additional 8% greater accuracy. Multi Objective PSO was investigated by Hasan, Gan and Zhang [68] in a comparison against SFFS and again, a clear advantage was found using the PSO technique with less channels being required with only a 2% loss in accuracy. PSO has also demonstrated to be effective in frequency selection [189] and optimisation of CSP [154].

3.1.2.3 *Genetic Algorithm*

Genetic Algorithms (GA) are powerful tools in optimisation problems and have demonstrated considerable results in feature selection for BCIs [112]. An initial population of potential solutions is (typically randomly) generated with each solution consisting of a chain of features known as 'genes'. After initialisation, genetic algorithms utilise three operators; selection, crossover and mutation

[112]. The selection operator is modelled on the principle of 'natural selection' in which the fittest organisms will survive to pass on their genes. This is achieved by selecting the fittest individuals within the population via an objective function and using their components to create the next generation. The crossover operator then recombines the selected solutions to form the next generation. An example of this is the selection of a single point in the solution, and pairs of individuals swap their genes after this point. The limitation here is that only the original randomly selected elements can be combinatorially explored, ignoring the rest of the search space. To combat this, a mutation operator is introduced: in Feature Selection implementations, one or more genes in the solution are randomly selected and replaced with alternative genes selected at random from the entire feature space. This not only widens the scope of the exploration, but also helps prevent the algorithm becoming trapped in local optima [124].

Genetic algorithms are one of the most popular search methods used for Feature Selection in BCIs [145]. While they are somewhat more computationally demanding, offline learning of classifiers allows us to focus on improving accuracy at the expense of speed. During their earlier implementations, standard genetic algorithms reported results that produced classification accuracies of around 74–76% [112] but have since been refined to produce in excess of 90% classification accuracy [134] (in two class problems, such as 'Yes or No' and 'Left or Right') on some datasets. This superior performance over filter methods is further supported by Dias et al. [43], who reported a substantially lower rate of classification error for GA than seen in Recursive Feature Elimination, Across-Group Variance and RELIEF, a trait that appears fairly consistent across the literature. Further comparisons include [141], where it found smaller subsets with higher degrees of accuracy than CFS, Relief and mRMR were found. The substantial increase in classification accuracy obtained from genetic algorithms has arisen largely from adapting the generalised operators to better suit the BCI arena. Rejer [145] notes that a traditional GA will lean towards improving accuracy of the classifications with the minimum number of features, but it is often the case that a slight decrease in accuracy is acceptable when a significant decrease in features is possible. To realise this, they modified the

mutator function to behave in a similar fashion to SFS; preserving the GA's ability to explore the solution space while giving precedence to the smaller feature sets observed in the SFFS method. This resulted in smaller relatively consistent feature sets that markedly outperformed the state of the art LASSO embedded method.

3.1.2.4 *Memetic Algorithms*

Memetic Algorithms (MA) have recently been used, and proven to be a viable technique, in a range of feature selection problems [98]. One of the caveats with Genetic Algorithms is that they lack a mechanism which allows exploitation of the immediate search space surrounding the solutions in their population. MAs have sought to overcome this by integrating a local search technique into the overall metaheuristic. This can be achieved through a hybridised genetic algorithm, in which a random mutation Hill Climbing search is performed on each of the newly created offspring before returning them to the population [28]. This technique was further compared to a GA in [55], demonstrating a higher accuracy on NP-Hard combinatorial problems.

3.1.2.5 *Iterated Local Search*

Iterated Local Search (ILS) can be thought of as a nested Hill Climbing algorithm. A local search refines a solution, before a perturbation operator moves it into a new region of the search space. A local search is performed again, and compared against the solution found before the perturbation, as seen in Figure 3.1. This results in the comparison of two local optima: if the most recent local search has found a higher quality solution than that prior to the perturbation, it is accepted. If not, the solution found at the previous local optimum is perturbed and local search applied again. It is important that this perturbation is strong enough that it escapes the local basin of attraction, but not so strong that it resembles multi-start local search [110]. Despite wide use in other domains, ILS has not been applied to any problem within the BCI field prior to this thesis. In [64], it was used for Feature selection on simulated and real genomic datasets, performing comparably, or better than, state-of-the-art methods: LASSO, elastic net and ridge. It has also been used in gene selection

for cancer identification in [45]. In this, it was found to perform better than a Genetic Algorithm, and produced further higher accuracies when used as the local search mechanism for a Memetic Algorithm. Within ILS implementations in other application domains, guiding perturbation with problem knowledge has been found to improve performance [15, 175]. A variety of different perturbation strategies exist in the wider literature: *Population Based ILS (PILS)*, in which records of previous solutions are retained to restrict the perturbation [171]; *ILS with guided mutation (ILS/GM)* uses a technique similar to Estimation of Distribution algorithms in that it takes statistical information regarding the search space into account [194]; and μ*CHC* which uses a micro-EA for diversification [110].

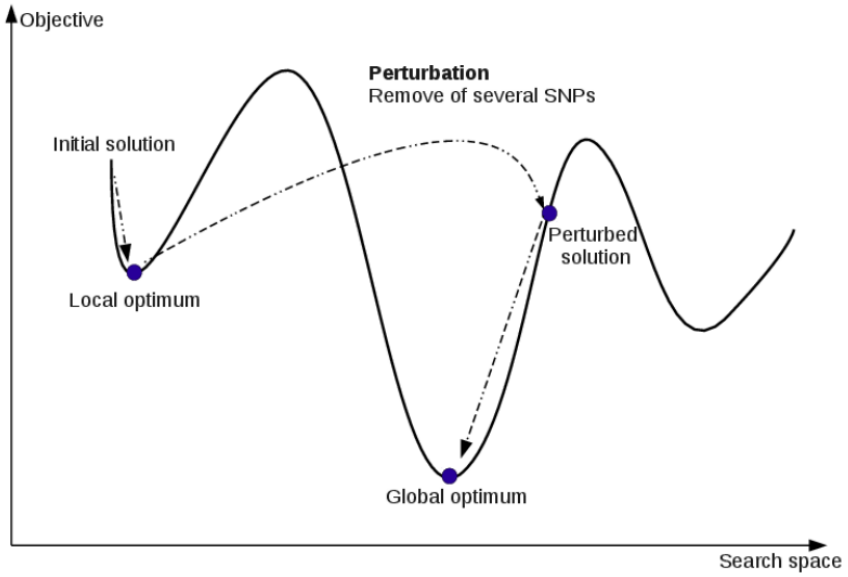


Figure 3.1: Search path of the Iterated Local Search (ILS) Algorithm [64]

3.1.2.6 *Other Evolutionary Approaches*

A comprehensive literature review on other evolutionary approaches that have been applied to Feature Selecion in BCI, such as *Differential Evolution (DE)*, *Harmony search (HS)*, *Invasive weed optimization (IWO)*, *Biogeography based optimization (BBO)*, *Teaching learning based optimization (TLBO)*, and *Non-dominated sorting genetic algorithm-II (NSGA-II)*, is provided in [168].

3.1.3 *Embedded*

Embedded methods involve a feature selection technique incorporated into the classifier. These techniques have shown some promise in the field, but are limited, by their definition, in that feature selection and classifier selection cannot be separated [16].

3.1.3.1 *Least Absolute Shrinkage and Selection Operator (LASSO)*

One of the most popular embedded methods is the *Least Absolute Shrinkage and Selection Operator (LASSO)*. By constructing a linear model that minimises the regression coefficients through penalising and utilising the residual sum of squares to calculate the error, a spatial filter bank can be created [145]. This method has been shown to be computationally less demanding than wrapper methods, yet it still provides a strong solution with few features. However, the standard LASSO method always includes the first and last eigenvectors, causing overfitting due to the commonality of outliers and the non-stationary nature of the dataset [174].

3.1.3.2 *Recursive Feature Elimination*

Recursive Feature Elimination (RFE) utilises the ranking procedure that is often contained within the classifier by simply removing the feature with the lowest ranking criterion after each run [192]. Within small training sets, RFE can successfully remove a large proportion of the features. However, it does not take in counter dependent features: individuals may score lowly, but together, they may prove invaluable to the overall classification. While authors such as Chen and Jeong [39] attempted to solve this issue with adapted RFE techniques, it still is uncompetitive when compared with wrappers.

3.1.4 *Hybrid Approaches*

A relatively uncommon approach in BCI is the combination of filters and wrappers in hybrid methods. A common form of this is a two-stage approach: a filter method is first applied to remove the most redundant individual

features, before a wrapper is applied to the remaining features. A variation of this is seen in [53], where *Sequential Forward Floating Search (SFFS)* was combined with *mRMR* by using the mutual information approach to select a set of candidate features for addition and removal at each phase. This reduced the computational training cost of utilising the classifier across all the candidate features. *Ant Colony Optimisation (ACO)* was combined with *Differential Evolution (DE)* in [86]. This technique used a mutual information evaluation function as the Selection Measure in ACO, and evaluated each of the ants using a Linear Discriminate classifier. This technique was also evaluated in [85].

In other feature selection applications, hybridised approaches involving mutual information are somewhat more prevalent. Mutual information was used to reduce the search space in advance of running a *Genetic Algorithm* in [169] and *Particle Swarm Optimisation* in [7]. It has also been successfully used within memetic algorithms as a local search method to refine the solutions found by PSO in *Particle Swarm Optimisation Backwards Elimination (PSOBE)* [122] and in Genetic Algorithms [195]. A common observation, however, is that mutual information is almost always used as a local search operator in these cases, and to this author's knowledge, has not been used as a diversification mechanism prior to this thesis.

3.1.4.1 *Linkage*

In evolutionary algorithms, *linkage* is a relationship or dependency between decision variables. As far back as 1975, Holland [77] suggested that operators aware of linkage information might be necessary for efficient GA search. The linkage model used by an EA can be implicit (e.g., linkage learning GA [66]) or explicit (e.g. multivariate Estimation of Distribution Algorithms [69, 109]). Interest in approaches that explicitly make use of the linkage and the structure that it imposes on the search space remains current, for example [25, 40, 183]. However, it has also been shown [24, 26, 70] that some aspects of linkage are essential for fully ranking all solutions to a problem and locating the global optima. Indeed, including such non-essential dependencies in the problem model used by the algorithm can hamper performance [21, 100, 138]. In [71],

a method for probing the search space for the interactions between variables was introduced. This later became known as the *Linkage Detection Algorithm*. It has previously been applied to Feature Selection [27], but not in the field of BCI.

3.1.5 *Feature Selection Summary*

In summary, Feature Selection is a known and effective manner in which classifier performance can be improved in terms of accuracy, speed, memory, and computational requirements. Furthermore, in the field of BCI, these advantages can materialise in the form of less equipment; thus reducing cost, saving time, and increasing the practicality of BCI devices.

The primary divisions of Feature Selection methods are Filters and Wrappers. Filters employ ranking measures to determine the most relevant features which typically results in fast and deterministic feature choice. Wrapper methods utilise stochastic techniques which require longer computational time, but are known to find feature subsets that create better fitting models based on the chosen classifier. The choice between Filter and Wrapper methods is often made on practical grounds: given that the limiting factors of BCI applications are often equipment expense and predictive accuracy, Wrapper methods are a worthwhile investment, but caution must be exercised as over-fitting is a known issue.

3.2 TRANSFER LEARNING

Another potential area for optimisation involves *adding* relevant data to the training set. While *Feature Selection* relies on *excluding* data that may be detrimental to the classifier performance, it has been shown that selecting data from other participants, and combining it with that of the target participant, can enhance the predictive model [188]. In BCI, Transfer Learning allows us to transfer knowledge gained from one patient, to another; potentially alleviating the *'large p, small n problem'*. This problem is defined as when the dimension-

ality p is high, and the number of samples n are low, a model can change substantially with small fluctuations in the training sample [97].

The difficulty in using multiple participants is exemplified in the non-stationary nature of brain signals: neural patterns not only differ between participants, but are also subject to *temporal drift*, where data obtained from a single participant changes drastically over time [80]. *Zero Training systems*, trained exclusively on participants from previous sessions, are an ideal goal, but this non-stationarity means highly accurate zero training systems may not be possible. Consequently, we must instead focus on minimising the participant-specific training information required by maximising the effectiveness of the data available.

Obtaining sufficient data from an individual for the creation of an accurate system comes with significant costs, so utilising databases from other participants offers an attractive avenue to alleviate this burden. Transfer Learning has been employed in a number of domains that have access to multiple data sources, allowing inferences to be made on data from previously unseen sources. For a more in-depth discussion of the wider field, [182] provides a recent and thorough survey. More specifically, BCI literature typically reports *domain adaptation* approaches [80], the most popular of which being *Common Spatial Patterns* [18]. This involves creating a transformation of the data that will allow a single classification rule to be applied across all instances. A much less commonly explored approach is *'Rule Adaptation'* [80], in which a number of rules are created from the existing datasets, and then applied to the new instances. Note that both cases rely upon the natural distribution of the data as grouped by their original participant. Some attempts have been made to group datasets by known variants such as gender [32], and others using the information extracted from the trained models [103]; but little has been done in regards to instance selection for each model.

3.2.1 *Ensembles*

One method for incorporating data from other domains is the use of *ensembles*. Ensembles typically consist of an array of different classifiers trained with the

same dataset. Each classifier makes predictions on a test set, and these are collated in a voting process. This allows multiple different relationships to be detected for the classification process, many of which may not be obvious, even to a domain expert. Another approach is to use multiple instances of the same classifier, trained with different initial datasets.

Ensembles have been used in a number of different BCI applications to increase accuracy and reduce the amount of training data required for participants. Arguably, the most well known *P300-Speller ensemble* is [139] in which an ensemble of SVMs were used to reduce variability in signal inputs by averaging classifier outputs; these, however, relied on a substantial quantity of participant-specific data. This, like most BCI ensembles [128], used naive partitioning in which the instances were divided by their associated labels, whether it be by source domain or by stimuli. This proves useful for weighting classifiers within the ensembles, allowing information regarding the appropriateness of each model and the test-domain to be extracted [103]. It was demonstrated in [128] that overlapping these naive divisions can actually increase accuracy, suggesting that having the same training data duplicated amongst the classifiers can benefit the overall performance.

3.2.2 *ELGI*

In 2015, Xu et al [187] introduced the *Ensemble Learning Generic Information (ELGI)* approach. Rather than using the small amount of training data to train a classifier, or for weighting the models within a larger ensemble trained on the data of other participants, ELGI combines the participant-dependent data with participant-independent data to form a hybrid ensemble. This is achieved by splitting the datasets of each existing patient within the database into target and non-target sets. The removed missing instance class (target or non-target) is then replaced by a copy of the corresponding class from the participant-specific training data. This results in an ensemble consisting of $2n - 1$ classifiers, where n is the number of participants within the database. An ensemble constructed in this manner allows smaller amounts of user-specific data to be supplemented by other users, while, to an extent, accounting

for the non-stationarity between their neurological patterns. This was shown to allow better generalisability between users, with high levels of accuracy, and reduced quantities of training data.

3.3 SUMMARY

In summary, previous chapters have described the need for BCI, along with the need to understand both the origin of the detected neurological signals, as well as their currently problematic classification. This has lead to the development of techniques to improve the quality of the training data including *Feature Selection* and *Instance Selection*.

Feature Selection methods are grouped into three primary divisions: Filters, a ranking method in which information is extracted based on the relationships between variables; Embedded methods, which rely directly on the classifier; and Wrappers, iteratively assessing feature subsets based on their ability to classify the training data. Hybrids of these seek to build upon the ability of Filter methods to detect relationships within the data, while utilising the classifier-aware nature of Wrapper methods.

Extending on the principle that features can be selected based on their relevance to improve a model, Instance Selection suggests that additional data can be acquired from other, related sources. As with Feature Selection, detection and assessment of the most relevant instances for models is paramount; this has been shown to reduce the amount of user-specific training data required, while increasing predictive accuracies. It is based on these assertions that we hypothesise our new hybrid methods *Benign* and *Malign Iterated Local Search*, and *Minimum Redundancy Maximum Relevance Iterated Local Search* to address research questions **RQ1** and **RQ2**. We then extend upon the motivations of Feature Selection in response to **RQ3** by developing a novel method of Instance Selection denoted as *evolved Ensemble Learning Generic Information*.

Part IV

METHODOLOGY

4

CHAPTER 4 - EXPERIMENTAL SETUP

This chapter details the experimental setup common across the forthcoming chapters, and details the BCI case studies used in these experiments.

Each case study consists of a dataset, the paradigm used, and procedure implemented for extracting appropriate features. We begin by describing the experiments in terms of participants, recording equipment, paradigm used, and preprocessing applied. The features extracted, solution size, fitness function, and tools are then described.

4.1 DATASETS

The datasets provided by the Berlin Brain Computer Interface Competitions have been some of the most prevalent in literature over the past few years. Two of these datasets (*D1* & *D2*) were used in this paper; Berlin BCI competition II, datasets III and IV[1]. Both of these datasets have proven popular for benchmarking in literature due to their challenging, but well-defined, nature. Dataset *D3* was acquired from the RIKEN Centre of Advanced Intelligence Project[2]. It does not appear as frequently in literature as the competition data, but was chosen as it is important that we explore a wider variety of state-of-the-art benchmarks. This will improve the generality of the algorithms used, better reflecting the diversity seen in real-world applications.

Dataset D4 was first provided in [76]. It uses a speller-like paradigm to elicit a P300 wave. As one of the most commonly cited datasets in BCI, it provides a structure that allows exploration of algorithm performance across different participants and time points. The following section will describe the paradigms used in each dataset, the conditions of their recording, pre-processing steps,

1 http://www.bbci.de/competition/ii/#datasets

2 http://www.bsp.brain.riken.jp/~qibin/homepage/Datasets.html

and the features extracted.

4.1.1 *Dataset D1 - Berlin BCI competition II Datasets III*

Paradigm A participant was asked to imagine left and right hand movements to control an on-screen cursor. A blank screen displayed. The first two seconds were a resting phase, followed by an auditory signal and cross being displayed in the centre of the screen to focus the participant's attention for one second, as demonstrated in Figure 4.1. The cross then became an arrow, signifying the motor-imagery (left or right hand movements) that the participant was required to imagine.

Recording and Preprocessing Three electrodes were placed at positions C3, C4, and Cz (Figure 4.2), and sampled at 128Hz over a set of 280 9-second trails with one participant. The signal was then bandpass filtered between 0.5 and 30Hz.

Data Structure There were 280 instances recorded over 7 sessions with breaks of a only a few minutes. 140 of those instances were randomly assigned as 'training data', and the remaining 140 as 'testing data'.

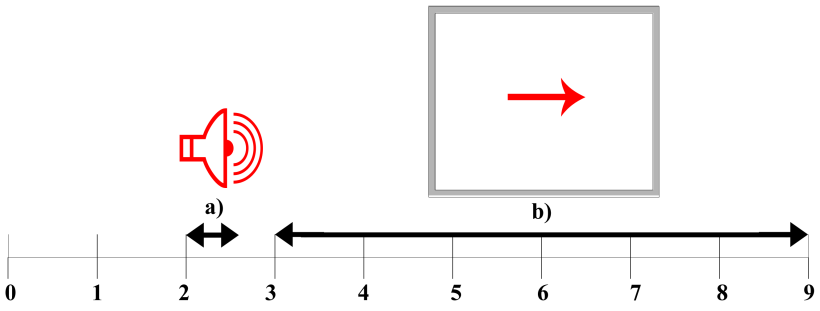


Figure 4.1: A timeline of the experimental paradigm used in Berlin BCI Competition II Dataset III. Over each 9 second trial, an auditory cue was played at 2 seconds and a cross displayed to focus the participants attention (a). An arrow then appeared onscreen (b), instructing the participant of which hand (left or right) they were required to imagine moving.

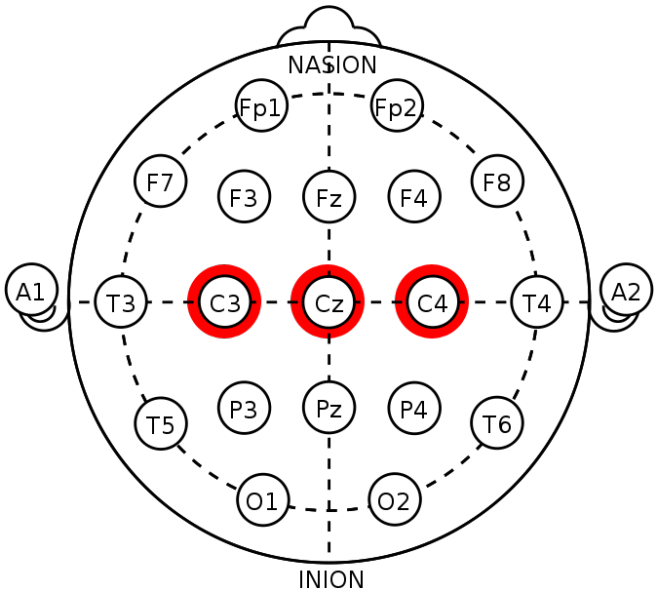


Figure 4.2: The electrode configuration for D1: Berlin BCI Competition II Dataset III followed the International 10-20 System [74] and placed 3 electrodes at C3, Cz, and C4

Table 4.1: Description of Dataset D1: Berlin BCI Competition II: Dataset III

Dataset Reference	D1
Dataset Name	Berlin BCI Competition II: Dataset III
Paradigm	Imagined Sensorimotor
Recorded Frequencies	0.5-30Hz
Time Epochs	9
Electrode Count	3
Training Instances	140
Testing Instances	140
Recording Sessions	Same Day - Randomly assigned to training/testing

4.1.2 *Dataset D2 - Berlin BCI competition II Datasets IV*

Paradigm The participant was asked to sit at a computer with their hands in a typical position at the keyboard. The participant was then allowed to press keys at a rate of one per second, in a self-determined order.

Recording and Preprocessing A set of 28 EEG electrodes performed sampling at 1000 Hz, band-pass filtered between 0.05 and 200 Hz, before being down-sampled to 100 Hz. The electrodes were arranged according to the international 10/20-system with electrodes being placed on Rows F, FC, C, and CP, and O1 and O2 (Figure 4.3).

Data Structure Three sessions consisting of a one minute rest period, 6 minutes of data collection, and a one minute rest period were recorded on a single day. In total, 416 instances were collected: 316 of which were designated as training, and 100 were provided, unlabelled, as testing data. This resulted in 416 instances of 500 ms, stopping 130 ms before the key-press, each labelled with either 'right' or 'left' hand, summarised in Table 4.2.

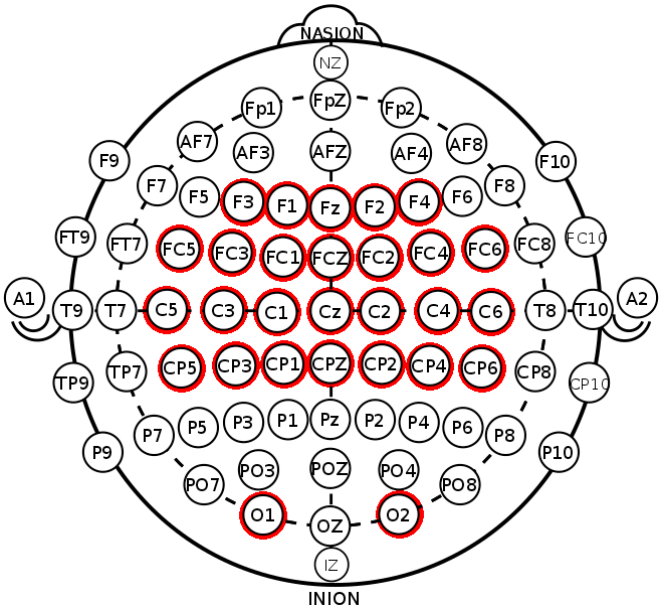


Figure 4.3: The electrode configuration for D2: Berlin BCI Competition II Dataset IV followed the International 10-20 System [74] and placed 28 electrodes at Rows F, FC, C, and CP, and O1 and O2.

Table 4.2: Description of Dataset D2: Berlin BCI Competition II: Dataset IV

Dataset Reference	D2
Dataset Name	Berlin BCI Competition II: Dataset IV
Paradigm	Intended Sensorimotor
Recorded Frequencies	0.5-100Hz
Time Epochs	1
Electrode Count	28
Training Instances	316
Testing Instances	100
Recording Sessions	Same Day - Randomly assigned to training/testing

4.1.3 *Dataset D3 - Riken - Subject A*

Paradigm Sessions one and two from Subject A were taken from the RIKEN EEG Datasets homepage[3]. A participant was asked to sit in a chair and pay attention to a blank screen. After 2 seconds, an arrow pointing left or right appeared and for the following three seconds, the participant imagined the corresponding left or right hand movements, as shown in Figure 4.4.

Recording and Preprocessing The recording was obtained via six channels, sampled at a rate of 256Hz, which was then band-pass filtered between 2 and 30Hz (Figure 4.5).

Data Structure In total, 264 instances were recorded: session one was selected as the training dataset with 130 trials, and the 134 trials from session two serving as the testing data. Unlike Dataset D1, Sessions 1 and 2 are recorded on different days, and the preceding two second rest period is not included in the data.

3 http://www.bsp.brain.riken.jp/~qibin/homepage/Datasets.html

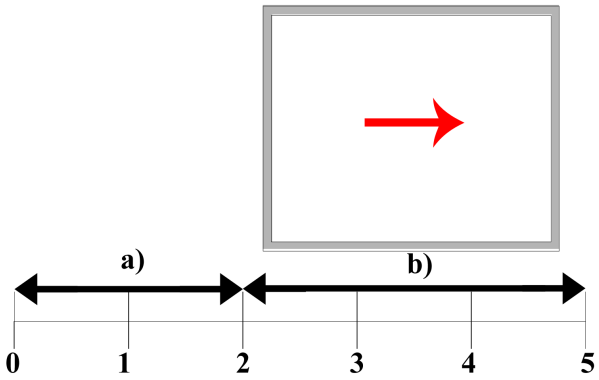


Figure 4.4: A timeline of the experimental paradigm used in the Riken - Subject A dataset. A non-recorded 2 second resting window (a) preceded a 3 second epoch of imagined hand movement. An arrow appeared on screen during the recording window (b) to indicate to the participant which hand to imagine moving.

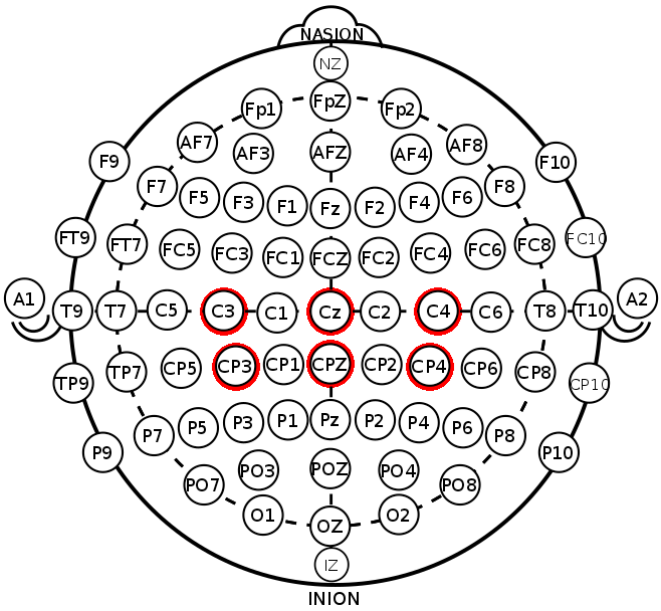


Figure 4.5: The electrode configuration for the Riken - Subject A dataset followed the International 10-20 System [74] and placed 28 electrodes at C3, Cz, C4, CP3, CPZ, and CP4.

Table 4.3: Description of Dataset D3: Riken - Subject A

Dataset Reference	D3
Dataset Name	Riken - Subject A
Paradigm	Imagined Sensorimotor
Recorded Frequencies	2-30Hz
Time Epochs	5
Electrode Count	6
Training Instances	130
Testing Instances	134
Recording Sessions	Different Days: Training from Day 1, Testing from Day 2

4.1.4 *Dataset D4 - P300 Speller (Hoffman)*

Paradigm This dataset was obtained from [76]. Much like the P300 speller described in 2.3.4, a series of images were shown on a screen, but in this case, images of objects were used instead of alphanumeric characters. These images were: a television, telephone, lamp, door, window, and radio. After a warning tone, the images were flashed by increasing their brightness randomly, one at a time, and the participant counted the number of times a target object flashed. Each flash lasted for 100ms with 300ms intervals.

Recording and Preprocessing A 32 electrode configuration was used in line with the international 10-20 system, but a 4 electrode configuration of this dataset was used for the purposes of increasing the challenge, and creating a more economical and deployable BCI. This sampling frequency was 2048 Hz and down sampled to 32 Hz. The signal was referenced against the mastoid electrodes, and bandpassed filtered between 1-12Hz. Winsorizing was also applied to remove noise sources, as described in Section 2.4.4.

Participants Unlike Datasets D1, D2, and D3, Dataset D4 contains multiple participants. D4 originally included datasets from 9 participants, but Hoffmann et al [76] suggests that one participant's dataset cannot be used, due

to fluctuations in their consciousness during the recording. The dataset used included four participants with varying neurological impairments, and four able-bodied PhD students. Participants 1, 2 and 4 were able to speak with some dysarthria, but participant 3 was unable to communicate verbally due to the symptoms of late stages of amyotrophic lateral sclerosis. All four disabled participants were wheelchair users, with limited or no control over their upper limbs. Participants 5-8 were PhD students with no known neurological issues.

Data Structure The dataset obtained for each participant follows a common hierarchical structure: each participant recorded 4 sessions of 6 runs. A 'run' is equated to 6 rounds, and a 'round' is the flash of all 6 images, 20 times. The first two sessions were recorded on one day, with the following two being recorded not more than two weeks later. This results in approximately 3240 trials for each participant, with 810 trials in each session.

Figure 4.6: Images presented in the P300 paradigm for the dataset used in [76]

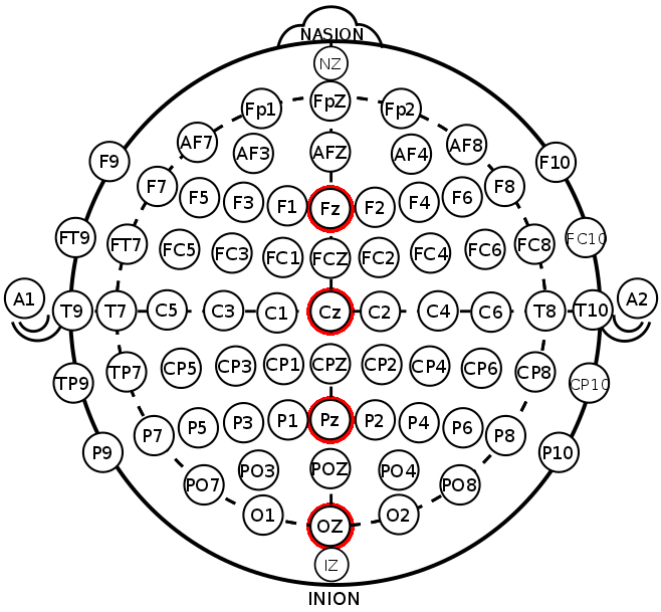


Figure 4.7: The electrode configuration for Dataset D4 followed the International 10-20 System [74] and placed 4 electrodes at Fz, Cz, Pz, and OZ

Table 4.4: Description of Dataset D4: P300 Speller (Hoffman)

Dataset Reference	D4
Dataset Name	P300 Speller (Hoffman)
Paradigm	P300 Speller
Recorded Frequencies	1-12Hz
Time Epochs	-
Electrode Count	4
Participants	9 (8 usable)
Training/Testing Instances	$\approx$ 3240 per participant
Recording Sessions	Different Days: 4 sessions over 2 weeks

4.1.5 *Feature Extraction*

As discussed in Section 2.5.2.2, the most common and appropriate type of feature extraction for motor imagery-based BCI is *Power Spectral Density (PSD)*. As the Berlin BCI and Riken datasets (Datasets D1, D2 and D3) fall into this

paradigm category, PSD features were extracted. In Section 2.3.1 it was shown that the most appropriate frequency range for this is between 8 and 30 Hz. This can be further decomposed into α and β bands, and [145] utilised a further 5 subdivisions within each: 2 primary bandwidths of 8-13Hz (α) and 13-30Hz (β), and 5 sub-bands within each (8-9, 9-10, 10-11, 11-12, 12-13Hz and 13-17, 17-20, 20-23, 23-26, 26-30Hz). To ensure temporal features could still be detected, signals were split into different 1 second *epochs (time segments)* for the Berlin BCI Competition II Dataset III and Riken datasets (Datasets D1 and D3), and a 0.5 second epoch for Berlin BCI Competition Dataset IV (Dataset D2). This results in each feature representing the PSD of a single epoch, recorded on one channel, over each of the frequency bandings.

Dataset D4 (Hoffman's P300 Speller) is used for experimentation in Chapter 6. This dataset differs substantially from the previous as it utilises a P300 evoked potential, which renders PSD features less effective. As a *BLDA* classifier was used, feature extraction was not necessary due to its tolerance of high dimensionality-to-instances (large p, small n problem) datasets.

Dataset	Number of Frequency Bands	Epoch Length Length (seconds)	Number of Epochs	Number of Channels	Number of Features
D1	12	1	9	3	*324*
D2	12	0.5	1	28	*336*
D3	12	1	3	6	*216*

Table 4.5: The number of features extracted from Datasets D1, D2, and D3 is determined by the number of frequency bands, number of epochs, and number of channels (electrodes)

Table 4.6: Overview of datasets used in the following experiments

Dataset	D1	D2	D3	D4
	Berlin BCI Competition II:		Riken -	P300 Speller
	Dataset III	Dataset IV	Subject A	(Hoffman)
Experimental Use		Feature Selection		Instance Transferral
Participants	1	1	1	9 (with 1 excluded)
Paradigm	Sensorimotor	Sensorimotor	Sensorimotor	P300
Task	Imagined hand movements	Finger movement	Imagined hand movements	Onscreen image selection
Electrodes	3	28	6	4
Classes	2	2	2	6
Frequency Range	0.5-30	0.5-100	2-30	1-12
Features Extracted	324	336	216	4
Training Instances	140	316	130	($\approx$ 3240 instances
Testing Instances	140	100	134	per participant)
Recording Time Frame	Same Day	Same Day	Different Days	Varied
Onset	Cued	Self-Determined	Cued	Cued

4.2 SIZE OF SELECTED FEATURE SUBSET

As noted in Chandrashekar and Sahin [35], there are no ideal methods to choose the size of the subset for selection. For this reason, we based the number of selected features, or 'solution size', on [145] for Dataset D1 when using an SVM so as to be consistent with the precedent set in literature. As there is no background literature for Datasets D2 and D3 that utilise the PSD features extracted, preliminary exploration was required. An upper limit was selected as 10% of the complete feature space as recommended in [82]. A Sequential Forward Search was then performed on each dataset to determine an appropriate number of features to seed the other algorithms.

4.3 FITNESS FUNCTION

Fitness functions for wrappers in feature selection typically consist of *k-fold cross-validation* on the training set [92]. This involves randomly splitting the instances into k sets. One set is then designated as a test set, while the remaining are used to train the model. The labels of this test set are recorded, and this process is repeated for all k sets. The accuracies of all k tests are averaged, giving the *Cross Validation Error (CVE)*. Although some publications determine the performance of algorithms based on the entire dataset, it is essential that an 'unseen' subset of the data is withheld from the feature selection algorithm. This is to ensure that the selected subset is generalisable to future tasks. In the following experiments, we set $k = 10$, as leave-one-out cross validation is prone to over fitting, and smaller k results in training subsets that contain too few samples for the feature subsets being tested [146].

4.4 TOOLS

All software was implemented in MATLAB. The experiments in Chapters 5 and 7 were performed on a machine with an Intel i7-3770 and 16GB of memory. Experimentation in Chapter 6 was performed using the EPSRC

funded ARCHIE-WeSt High Performance Computer (www.archie-west.ac.uk). EPSRC grant no. EP/K000586/1.

Part V

LINKAGE

5

CHAPTER 5 - LINKAGE

5.1 INTRODUCTION

While many different algorithms have been applied to the problem of feature selection in BCI, they often assume the features are independent, lacking the ability to exploit relationships that may exist between features. A technique that has been useful in other Feature Selection problems has been to utilise linkage information [27, 38, 132]. By probing the features to determine their separate and joint contribution to fitness, we can reveal 'linkage' between them. Linkage aware operators can be devised to exploit this information and potentially increase performance.

This chapter proposes a method in which operators in evolutionary algorithms can be guided using linkage to increase the classification accuracy of EEG data. To this end, we initially compare four base algorithms: Hill Climbing (HC), Iterated Local Search (ILS), Genetic Algorithm (GA), and Memetic Algorithm (MA) (Section 5.2). Thereafter, linkage is incorporated into both Hill Climbing (Section 5.3.3.1) and ILS algorithms (Section 5.3.4.2). These techniques were applied to the dataset provided by the second Berlin BCI competition, in track three (motor-imagery - Dataset D1). Potential explanations for the behaviours observed are also explored in detail (Section 5.4).

The main contribution of this chapter is to assess the viability of guiding metaheuristics for the feature selection phase of brain computer interfaces, using knowledge of pairwise interactions (linkage) between features.

5.2 PRELIMINARY ALGORITHM EXPLORATION

Information detected from pairwise interactions between variables can be integrated into a variety of different search based algorithms. We deemed it

important to verify the applicability of said algorithms within the problem field before adopting one for further experimentation. To this end, four algorithms were investigated: a *Hill Climbing (HC)*, *Iterated Local Search (ILS)*, *Genetic Algorithm (GA)*, and a *Memetic Algorithm (MA)*.

5.2.1 *Experimental Parameters*

In this section, parameters that were used to govern the execution of the experiments, and the algorithms within, are outlined. The dataset, classifier, search algorithms and their parameters are defined.

Dataset: The Berlin BCI Competition II Dataset III dataset (as detailed in Section 4.1, Dataset D1) was used for experimentation in this chapter.

Solution Representation: An integer representation was used for the solutions: an array of integers of a set size, representing the selected feature vectors.

Classifier: As this dataset is based on motor-imagery, a Support Vector Machine (SVM) was used, as supported by literature in Section 2.6. 10-fold Cross Validation using the training set was used as a fitness function. Solutions with lower error rates were deemed to be fitter.

Search Algorithms: All runs were restricted to 100,000 evaluations of the classifier. The following algorithms were compared:

HILL CLIMBING ALGORITHM (HC)

- *Description:* A Hill Climbing algorithm is a local search algorithm in which a solution can be subjected to a mutation, and this new mutation evaluated. If it is deemed to have improved on the previous solution, it is accepted as the new, current, solution.
- *Mutation Operator:* single point mutation
- *Acceptance Criterion:* first improvement

- *Iteration Limit:* As each iteration requires only one evaluation of the solution, 100,000 iterations were used.

ITERATED LOCAL SEARCH (ILS)

- *Description:* An Iterated Local Search (ILS) algorithm encompasses a local search in the form of a Hill Climbing algorithm within a larger, exploratory search.
- *Outer Perturbation 'kick':* single point mutation
- *Perturbation Operator:* multi-point mutation (50% of solution)
- *Mutation Operator:* single point mutation
- *Acceptance Criteria:* first improvement
- *Iteration Limit:* Preliminary tests suggested that 'kicks' were only required when the inner local search became stuck in a local optimum. This was demonstrated by higher performance in experiments which used 1000 Hill Climbing iterations and 100 kicks, compared with 100 iterations and 1000 kicks. Hence, the former was selected for comparison.

GENETIC ALGORITHM (GA)

- *Description:* A Genetic Algorithm (GA) is a population based approach which uses selection criteria to determine the solutions which propagate (through cross over and mutation) the next generation.
- *Population Size:* 20 solutions, as selected in [107], which specifically sought to addess overfitting in wrapper-based Feature Selection. This is further justified in reference to a GA, with a population of only 10 solutions, returning some of the highest accuracies reported for this dataset [145].
- *Population Type:* A steady state model was used, with a pair of offspring replacing the losing solutions in each tournament.
- *Selection Type:* Tournament Selection with a tournament size of 2.
- *Crossover Operator:* Random single point crossover.

- *Mutation Operator:* Random single point mutation.
- *Iteration Limit:* only 2 runs of the classifier were needed per iteration and therefore the GA ran for 50,000 iterations.

MEMETIC ALGORITHM (MA)

- *Description:* A Memetic Algorithm (MA) is a relation of the Genetic Algorithm that also includes a method of local refinement. In this case, a Hill Climbing algorithm is used for local search.
- *Population Size:* 20 solutions, initialised at random.
- *Population Type:* A steady state model was used: a pair of offspring replacing the losing solutions in each tournament.
- *Selection Type:* Tournament Selection with a tournament size of 2.
- *Crossover Operator:* Random single point crossover.
- *Mutation Operator:* Random single point mutation.
- *Inner Hill Climbing algorithm:* Prior to child solutions being added to the population, a Hill Climbing algorithm was applied
  - *Mutation Operator:* single point mutation
  - *Acceptance Criteria:* first improvement
- *Iteration Limit:* each child solution was subject to a 100 iteration Hill Climbing algorithm, and with 2 children produced per generation/iteration, the MA ran for 500 iterations.

5.2.2 *Algorithm Performance Comparison*

Preliminary tests were performed using the aforementioned algorithms, for the purposes of selecting an appropriate algorithm for modification to exploit linkage information. After 30 runs of each algorithm, GAs were found to produce

consistently lower error rates, closely followed by Iterated Local Search (Figure 5.1). Despite a relatively weak configuration for the GA, it outperformed the other algorithms in the main. This is not unexpected as an investigation into the effects of population size demonstrated a steep improvement between 5 and 20 chromosomes [107], with a lesser improvement seen between 30 and 100, and with no improvement found in larger populations. The Hill Climbing algorithm performed inconsistently, typically producing inferior solutions to the other techniques.

Iterated Local Search was chosen for modification as it produces solutions that are competitive with those of the Genetic Algorithm, but does not require a cross-over operator which might disrupt linkage and complicate analysis. This choice was similar to that of [64].

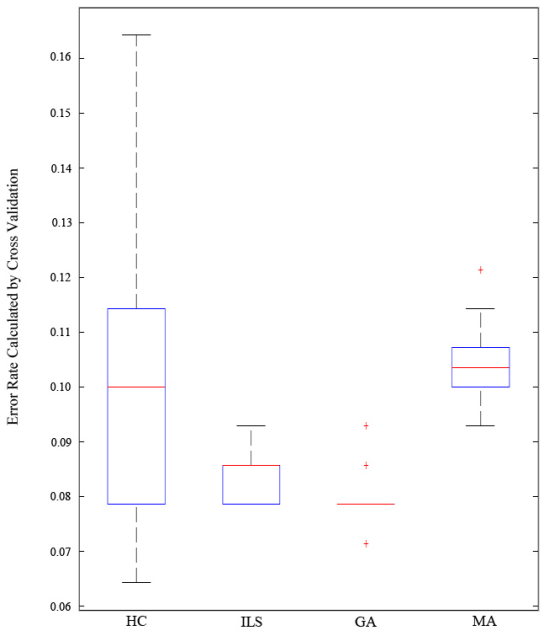


Figure 5.1: Box plots comparing the error rates of solutions found by each algorithm over 30 runs.

5.2.3 *Evidence of Feature Interaction*

In the algorithm selection experiment, 104 solutions with an error rate of less than 10% were found. As these are high quality solutions, they were analysed for the strength of each feature's contribution to the fitness in isolation. A

plot of feature selection rates for the features, sorted by descending classifier accuracy is shown in Figure 5.2. Given that the most predictive features were the most commonly selected, it would suggest that individual abilities are highly important for feature selection in this problem. However, it should be noted that there are significant gaps in the selected feature space, suggesting some feature linkage and that simply choosing the most predictive individual features would be a less than optimal approach. It is this interaction (or 'linkage') that the work in this chapter sought to detect and exploit in the following experiments.

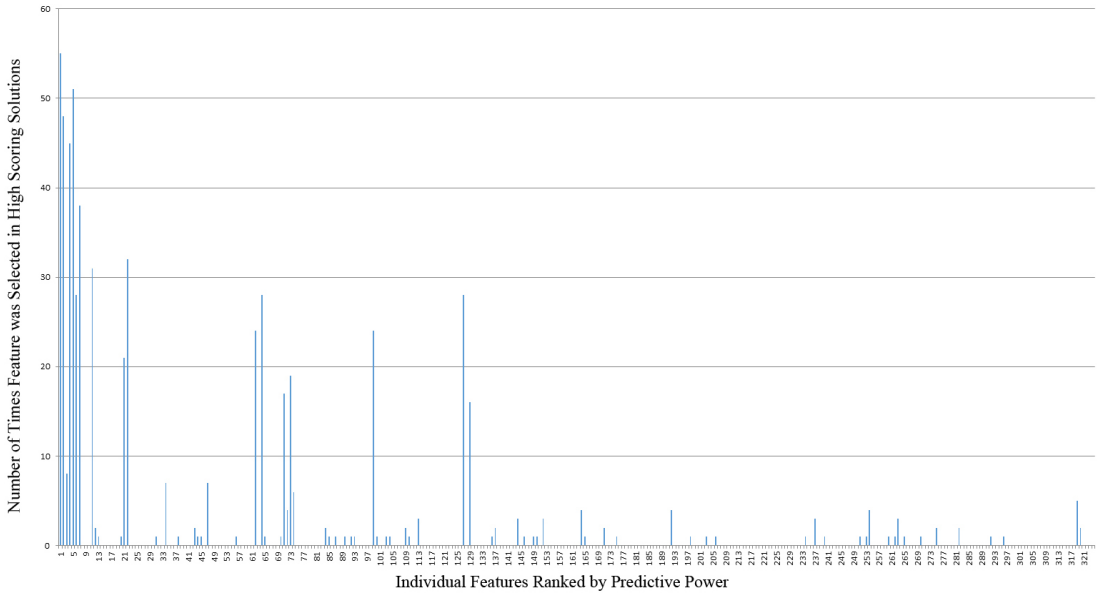


Figure 5.2: Selected features according to individual predictive accuracy. Each feature was independently tested as a single feature solution to train the classifier and cross validation was performed. This allowed features to be 'ranked' according to their individual power. 104 solutions with <10% error rate were detected in the earlier experimental phase and the occurrence of each feature was tallied.

5.2.4 *Discussion of Selected Features*

As seen in Figure 5.3, the most commonly selected channel is over the left hemisphere (C3), followed by the right (C4). The central electrode (Cz) is much less commonly selected. This is an interesting, but again not unexpected, result as the left-hemisphere has been shown to be especially important in motor-

control tasks [87]. More recent studies suggest it is particularly important in novel motor tasks, which are common in BCI paradigms [121].

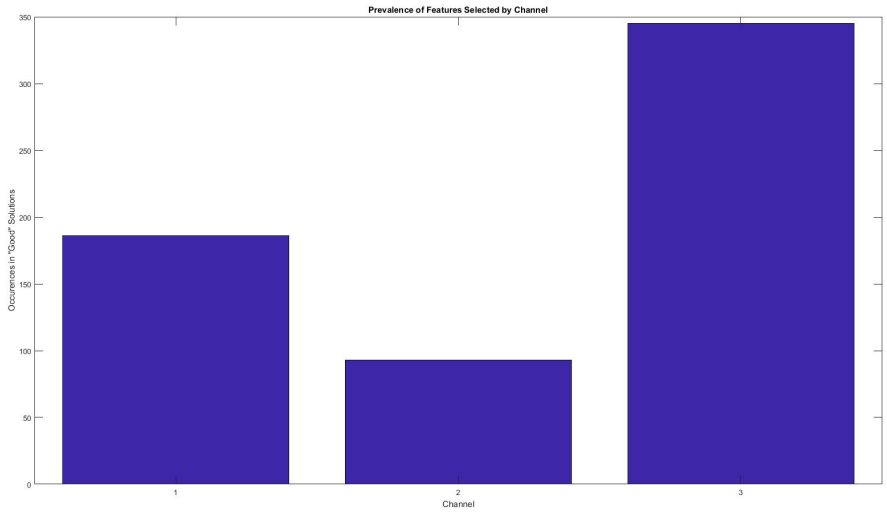


Figure 5.3: Most commonly selected channels in best performing solutions found

In regards to bandwidth selection (Figure 5.4), an important observation can be made; the α frequency band (8-13Hz, and more importantly, the lowest frequency band (8-9Hz) were most commonly selected. This suggests that lower, unused frequency bands, such as θ (4-7Hz) or δ (< 4Hz), may be of interest, despite being often discarded in EEG BCI tasks [125, 144].

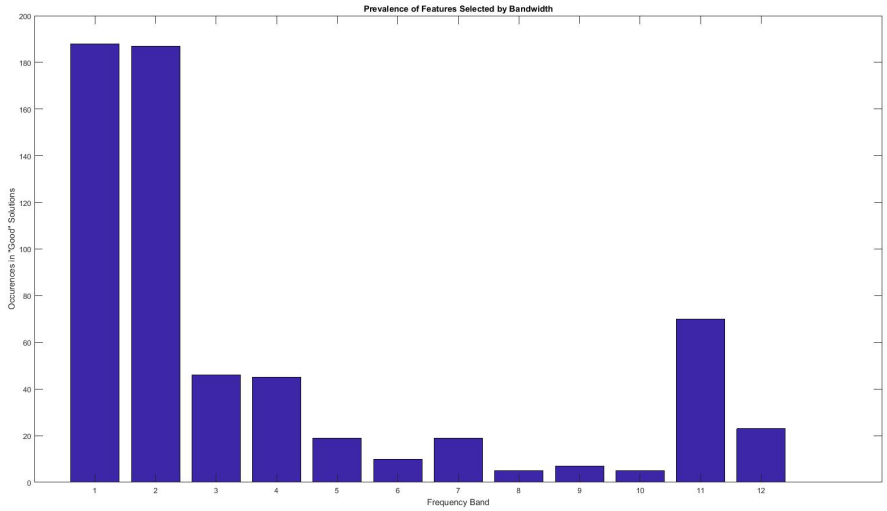


Figure 5.4: Most commonly selected frequency bandwidths in best performing solutions found

The most commonly selected time points are shown in Figure 5.5. The first second after the directional arrow was displayed onscreen was rarely selected (epoch 4); however, epochs 5 and 6 appear to contain the features richest in information. Other notable observations are that the mental status of the participant just before the auditory stimulus (epoch 2) appears to be of some interest; and epoch 7 is rarely chosen.

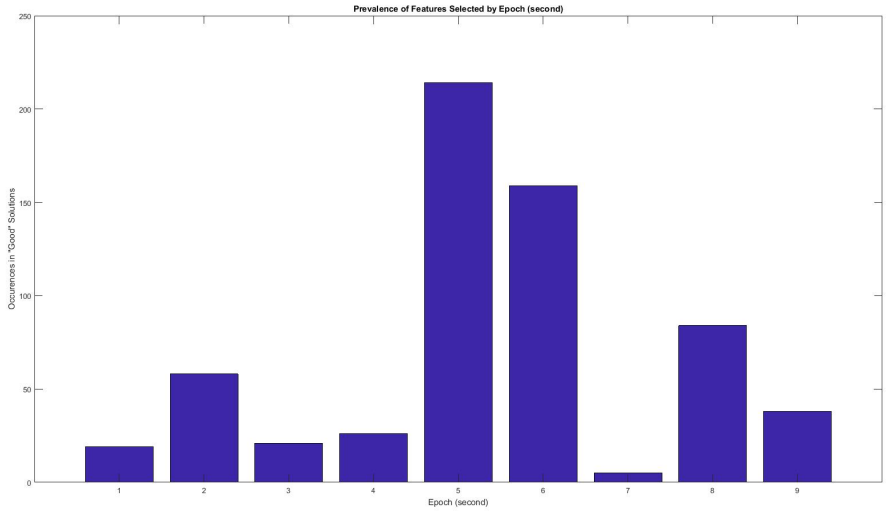


Figure 5.5: Most commonly selected epochs in best performing solutions found

5.3 LINKAGE INTEGRATION DESIGN

Figure 5.6 displays the data flow within the Feature Selection phase of the proposed metaheuristics with Linkage. At **(1)**, the training data is used to create a mapping of all pairwise linkages within the feature set, which is then passed to the metaheuristic. The metaheuristic **(2)** then selects features and performs cross-validation using the training data. The fitness returned by the cross-validation is then used by the metaheuristic to guide the next iteration of Feature Selection. After stopping criteria have been met, the Feature Selection phase is ended, and the selected features at that point are passed on to be used on the testing data **(3)**. As this is a black box optimisation problem, the classifier accuracy was utilised as a fitness function.

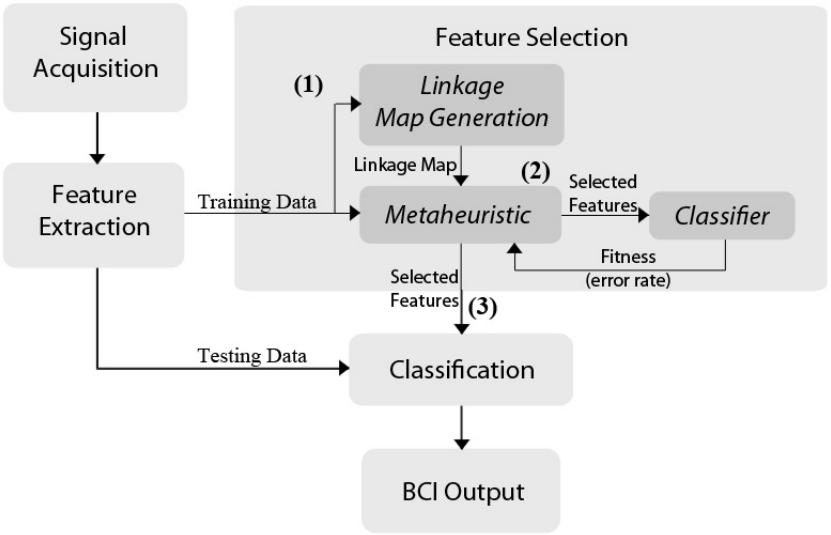


Figure 5.6: Sequence diagram displaying the incorporation of Linkage in the Feature Selection phase

5.3.1 *Linkage Map Generation*

Linkage between features was determined by applying the Linkage Detection Algorithm [71] to the training data, which has been used in linkage-aware Estimation of Distribution Algorithms (e.g [23] and [193]). The algorithm starts with no features selected: the classifier accuracy f_ϕ is determined. The accuracy f_a is calculated having selected only feature a. From this we have a change in accuracy from the baseline $\delta_a = f_a - f_\phi$. This is then repeated to find δ_b when selecting only feature b, and δ_{ab} when selecting features a and b. For a pair of features a and b, the change in classifier accuracy is measured while selecting the two features separately δ_a, δ_b and both together δ_{ab}. We have called the difference in these changes in accuracy the *Linkage Score*, $s_{ab} = \delta_{ab} - (\delta_a + \delta_b)$. If s is non-zero, there is deemed to be linkage between the variables. This method can be expanded to higher levels of interaction but its complexity grows rapidly with the level of interaction.

The Linkage Score was calculated for every pair of the 324 features. Dependencies (linkage) were classified as benign and malign in [83]. *Benign* linkage is that for which the combined change in fitness is in the same direction as the independent changes (i.e. the signs of $\delta_a + \delta_b$ and δ_{ab} are the same). *Malign* linkage shows a combined change in the opposite direction to the independent

changes (i.e. the signs of $\delta_a + \delta_b$ and δ_{ab} are the opposite). These terms were adopted in the following way. If a pair yields a positive Linkage Score, it reflects an increase in error rate over the combination of the individual scores and is deemed 'malign'. A negative score suggests that there is a reduction in error rate when the features are combined and is hence a 'benign' linkage. We would expect a 'good' solution to include low levels of malign linkage, and high levels of benign linkage. The operators were designed accordingly.

5.3.2 *Linkage in Dataset D1*

The Linkage Score (as described in Section 5.3.1) was calculated for all pairings of the 324 features and is illustrated in the heat map in Figure 5.7. Heat maps showing only benign and malign linkage scores are also provided (Figures 5.8 and 5.9). Darker regions represent strong levels of linkage, lighter regions being weakly linked. Linkage scores were more pronounced in the broader frequency ranges, as seen in Figure 5.7: features 1 to 27 (Frequency Band f_1) and 163 to 189 (Frequency Band f_7) have clear bonds, showing that these features are strongly linked. This was especially noticeable in the malign linkage scores map, Figure 5.9. The information presented by these maps was then provided to the linkage exploitation algorithms in the following experiments.

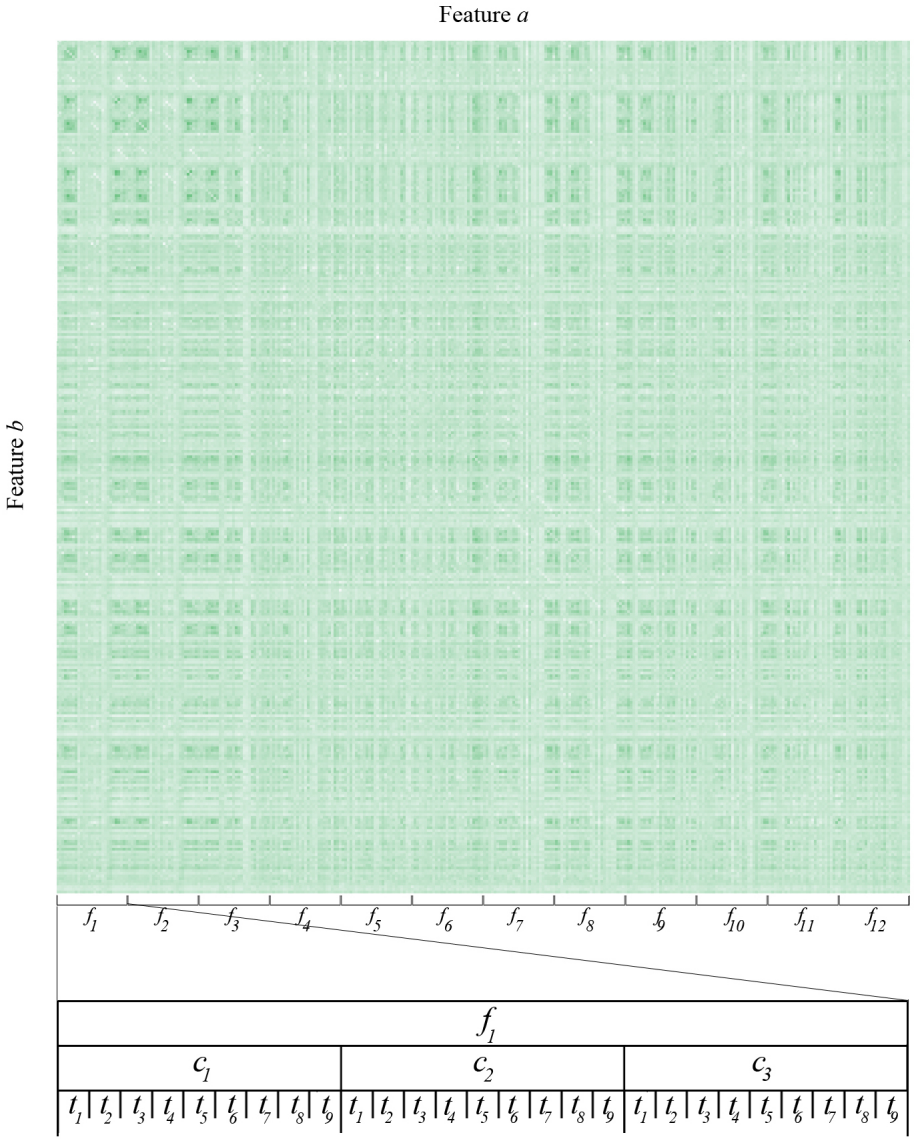


Figure 5.7: Linkage scores between all potential feature pairings (a_m,b_n) extracted from Dataset D1: Berlin BCI II Competition III Dataset. Darker points represent stronger linkage between pairs of features. Each axis represents a concatenation of all 324 features, with an example of the breakdown within each feature given above. Each of the 12 frequency bands (f_i) consists of the Power Spectral Densities extracted from 9 time points (t_j) simultaneously recorded over 3 channels (c_k). The lower section of the image demonstrates the breakdown of the features within a frequency band.

Figure 5.8: Figure 5.7 filtered to display only benign linkage

5.3.3 *Linkage Integration*

In order to verify the efficacy of incorporating Linkage Information in wrapper approaches for this domain, it is necessary to perform preliminary experimentation to answer two questions: "can Linkage Information be used to inform feature subset selection?"; and "in what manner can it be most effectively utilised?"

The first question is addressed by suggesting a greedy linkage-based search method in Section 5.3.4.1, with the second addressed by evaluation of Linkage incorporation methods in the following section.

For the purposes of exploring how Linkage could be utilised during the exploitation phase of the Iterated Local Search algorithm, six Hill Climbing algorithms were devised and tested, as described in Section 5.3.3.1. One hundred experiments were repeated for each of the six Hill Climbing variations, with a termination of 1000 iterations using a single point mutation.

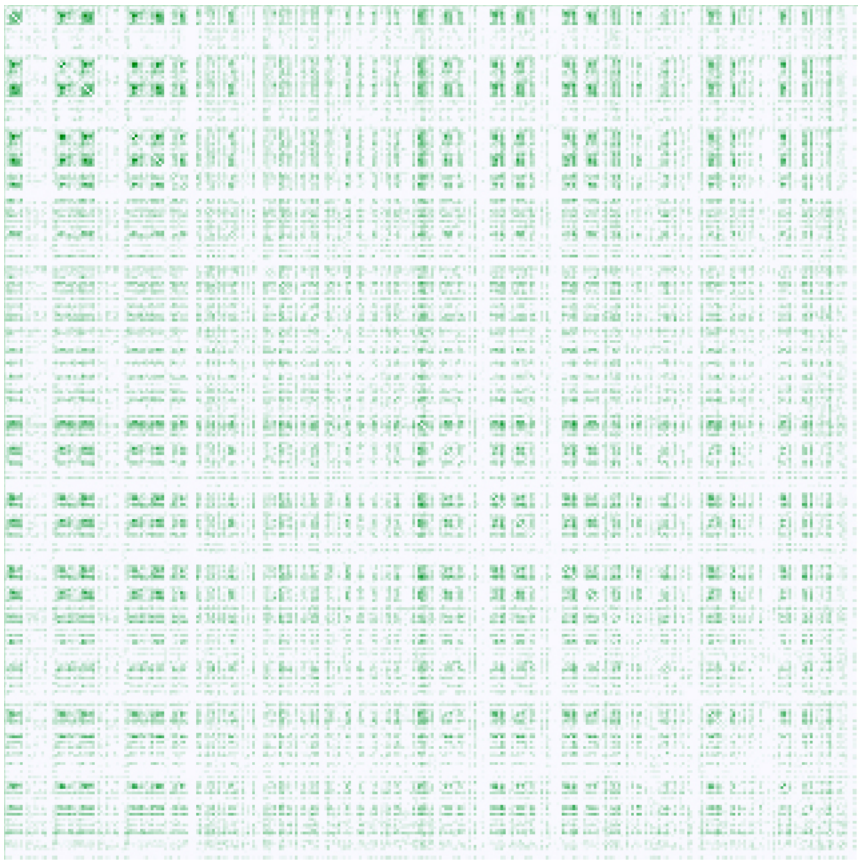


Figure 5.9: Figure 5.7 filtered to display only malign linkage

Each repeat experiment began by randomly generating a single solution; all algorithms were seeded with this same solution. Each of the proposed operators considers Linkage among the selected features in a solution, and whether replacing a feature increases or decreases this.

5.3.3.1 *Hill Climbing Algorithm with Linkage Integration*

We created a range of Linkage-aware operators to incorporate into the ILS algorithm. These were first used to create Hill Climbing algorithms in order to determine which would be best suited for further, more intensive, exploration.

H1. Basic Hill Climbing Algorithm - A simple Hill Climbing algorithm in which the mutation point and a replacement feature were both randomly selected was required as a control.

H2. Selection of Mutation Point - Target Most Malign Feature Pair - All pairs of selected features within the current solution are compared. One of the features

in the pair that reflects the largest malign linkage score is selected at random for deselection and replacement with another feature chosen at random.

H3. Selection of Mutation Point - Target Most Malign Feature - Both features of the pair with the largest malign linkage score in the solution are compared with the other selected features in the solution. The feature with the most malign linkages is deselected and replaced with an unselected feature chosen at random.

H4. Selection of Mutation Point - Spare the Most Benign Pair - The mutation point is chosen at random, but the feature pair within the solution that have the largest benign linkage score are excluded from possible mutation.

H5. Selection of Replacement - Good Mutation - A feature is chosen for deselection, and 20 features are chosen at random from the unselected features as potential replacements. Each of these potential replacement features are paired with the remaining solution features, and the one with the highest benign linkage score is selected.

H6. Selection of Replacement - Best Mutation - As in the 'Good Mutation' condition H5, but *all* unselected features are assessed as potential replacement candidates.

H7. Selection of Mutation Point - Target Most Benign Feature - To ensure that the linkage information was being used appropriately, a counter-intuitive method which deselected the feature with the most benign linkage scores with other selected features was also used.

5.3.4 *Results and Discussion*

Figure 5.10 shows boxplots for the error rates of the final solutions found by the Hill Climbing algorithm, using the seven different mutation operators (with and without linkage guidance - see Section 5.3.3.1). Counter-intuitively, using

the linkage-guided mutation operators appears to hinder the performance of the simple Hill Climbing algorithm in all conditions. Notably, operator H6 returns, by a large margin, the worst results. Two operators that produced solutions competitive with those of the unguided algorithm (H1) were both related to the target of the mutation operator: selecting the most benign (H7) and most malign (H3) features from within the solution. This suggests that feature subsets with high degrees of Linkage, albeit benign or malign, may be harmful to the fitness of a solution. This could be explained by the additional degrees of freedom introduced to the search via the calculation of Linkage. Specifically, cross validation using random splits is preferable to using the same folds, as it helps to avoid over-fitting. However, this introduces additional variance into the fitness function via three separate CVE evaluations: when each feature is evaluated individually, and then as a pair.

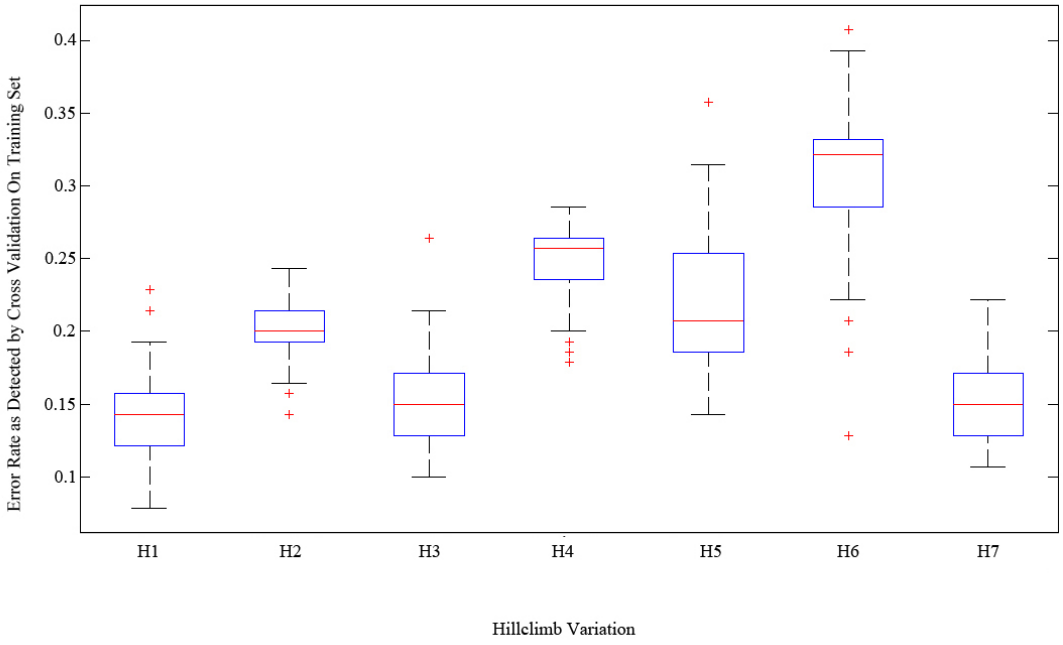


Figure 5.10: Preliminary testing of different methods of linkage guidance in Hill Climbing algorithms

5.3.4.1 *Greedy Linkage-based Feature Selection*

We design two Linkage-aware algorithms to assess a greedy ranking-based approach, rather than incorporation into more costly wrapper-based algorithms. These new algorithms take inspiration from the approaches used in Mutual Information Feature Selection (MIFS) and Minimum Redundancy Maximum

Algorithm 1 Greedy Linkage Feature Selection

Input: $linkageRankedFeatures \leftarrow$
sort(summedColumns(generateLinkageMap(trainingData, labels)))

Output: Final solution is S_{best}

Let maximumSubsetSize $= 100$
$S_{best} \leftarrow$ linkageRankedFeatures[1]
$S_{bestErrorRate} \leftarrow$ evaluteSolution(S_{best})
for $x = 2 \rightarrow$ maximumSubsetSize **do**
  $S_{candidate} \leftarrow$ linkageRankedFeatures[1..x]
  $S_{candidateErrorRate} \leftarrow$ evaluteSolution($S_{candidate}$)
  if ($S_{candidateErrorRate} < S_{bestErrorRate}$) **then**
    $S_{best} \leftarrow S_{candidate}$
    $S_{bestErrorRate} \leftarrow S_{candidateErrorRate}$
  end if
end for

Relevance (mRMR). As discussed in [133], Mutual Information can be used as a metric to measure the relevance of each feature, and the highest ranking of which can be selected as a solution subset. In our first algorithm, hereby referred to as Greedy Linkage Feature Selection (GLFS), we replace the MI based measure with that of Linkage Information (Section 5.3.4.1). The Linkage Score of each feature is calculated and ranked, after which, the top n features can be selected, or appended to a solution until a threshold fitness has been achieved. This algorithm is presented as pseudo-code in Algorithm 1.

The second of our greedy Linkage-based algorithms is inspired by mRMR (described in Section 6.1.3), which seeks to iteratively take into account the relevance of features already selected, before appending another. To achieve this we designed Maximum Linkage Feature Selection (MLFS), see Algorithm 2, where the feature with the highest Linkage Information is initially selected. Further features are selected based on their linkage with features already in the solution, as described by the following pseudo-code:

Algorithm 2 Maximum Linkage Feature Selection

Input: $linkageMap \leftarrow$ generateLinkageMap(trainingData, labels)

Output: Final solution is S_{best}

Let $n =$ numberOfFeatures
Let maximumSubsetSize $= 100$
% Select Initial Feature
$S_{bestLinkage} \leftarrow 0$
for $i = 1 \rightarrow n$ **do**
  $S_{candidateLinkage} \leftarrow \text{sum}(linkageMap[:, i])$
  if $(S_{candidateLinkage} < S_{bestLinkage})$ **then**
    $S_{maxLinkageFeatures} \leftarrow i$
    $S_{bestLinkage} \leftarrow S_{candidateLinkage}$
  end if
end for
$S_{best} \leftarrow 0$
$S_{bestErrorRate} \leftarrow \text{evaluteSolution}(S_{maxLinkageFeatures})$
% Select Additional Features
for $x = 2 \rightarrow$ maximumSubsetSize **do**
  $S_{bestLinkage} \leftarrow 0$
  for $i = 1 \rightarrow n$ **do**
    $S_{candidateLinkage} \leftarrow \text{sum}(linkageMap[S_{best}, i])$
    if $(S_{candidateLinkage} > S_{bestLinkage})$ **then**
      $S_{bestCandidate} \leftarrow i$
      $S_{bestLinkage} \leftarrow S_{candidateLinkage}$
    end if
  end for
  $S_{maxLinkageFeatures} \leftarrow [S_{maxLinkageFeatures} i]$
  $S_{candidateErrorRate} \leftarrow \text{evaluteSolution}(S_{maxLinkageFeatures})$
  if $(S_{candidateErrorRate} < S_{bestErrorRate})$ **then**
    $S_{best} \leftarrow S_{maxLinkageFeatures}$
    $S_{bestErrorRate} \leftarrow S_{candidateErrorRate}$
  end if
end for

To ensure the integrity of the results, we select our solutions and their parameters based on metrics obtained exclusively from the training data. For this reason, we calculate the CVE rate for each size of solution of up to 100 features; subsequently, the solution with the smallest observed error rate is used to determine the size of the selected subset.

Table 5.1: Comparison of Cross Validation Error Rates between Greedy Linkage algorithms and Linkage-guided Hill Climbing algorithms

Algorithm	**H1**	**H2**	**H3**	**H4**	**H5**	**H6**	**H7**	**GLFS**	**MLFS**
CVE	0.1401	0.2012	0.1517	0.2484	0.2209	0.3105	0.1532	0.4206	0.3014

When compared to the seven Hill Climbing variations, the Greedy Linkage-based algorithms produced solutions with substantially greater error rates (GLFS: 0.4206 and MLFS: 0.3014), and are not further investigated in this chapter. However, they have been included as a baseline for comparison in Chapter 6.

5.3.4.2 *Iterated Local Search with Linkage Integration*

Iterated Local Search (ILS) is a little explored algorithm in BCI and, to our best knowledge, has not been tested on feature selection for EEG. ILS has been selected as it is less convoluted than other EA methods, lacking the need for a population or cross-over, which should help emphasise the effects of the guided mutation operator. In essence, it is a nested Hill Climbing algorithm: In a traditional Hill Climber, a small mutation, replacing a selected feature with an unselected one, is performed on the initial solution to create a new potential solution. This new solution is scored via a fitness function and then accepted if it is deemed to be 'fitter' than the initial solution. This process is repeated to find increasingly optimal solutions, but can often become trapped in local optima. In an ILS, a 'kick' is performed by mutating a large portion of the solution (3 of the 6 features in this case). A Hill Climbing algorithm is then performed on this new, heavily mutated solution, and the resulting solution from this is then compared to the original, 'pre-kicked' feature set.

Iterated Local Search was selected for modification to explore the exploitation of linkage information in a more sophisticated algorithm. ILS has a two tiered iterative structure, from which we chose to provide guidance to the 'kick' function. For each selected feature in the solution, we calculate its *Mutual Linkage (ML)*; the mean linkage score between that feature and the other selected features in the solution. The three features with the highest ML were retained in the solution, and the remaining three were removed and replaced with randomly selected features.

Two variations of this method were tested:

I1 - Benign-preservation - The ML was computed using only benign linkage scores between features.

I2 - Malign-preservation - The ML was computed using only malign linkage scores between features.

5.3.5 *ILS with Linkage*

When considering the preliminary testing phase in which linkage was used to exploit Hill Climbing algorithms, it appears that selection of the mutation targets in a solution may be beneficial (*H3* and *H7*), and that interfering with the selection of their replacements is detrimental (*H5* and *H6*). This led to the selection of a modified 'kick' phase, in which only the targets for mutation were manipulated. The results of these tests are displayed in Figure 5.11. Performance of the guided and unguided ILSs were not found to be statistically significantly different (analysis performed by a two-tailed t-test, $p > 0.05$).

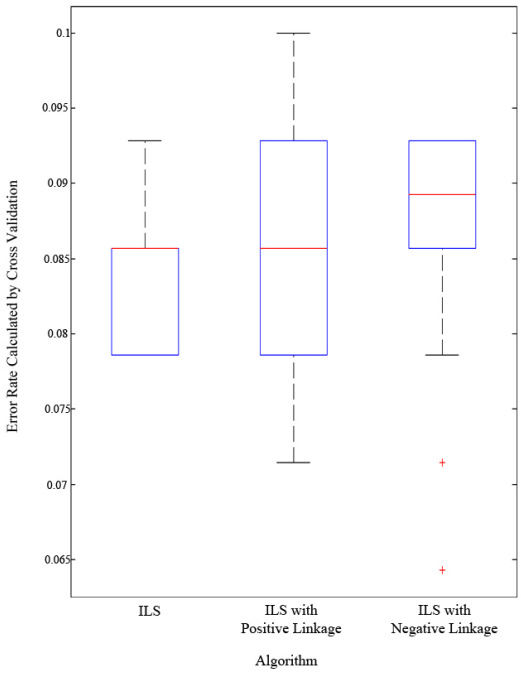


Figure 5.11: Comparison of error rates obtained by Iterated Local Search, and Iterated Local Search with guidance via positive and negative linkage

5.4 ANALYSIS

To further explore the reasons as to why Linkage exploitation did not prove effective in the previous experiments, further analysis was performed on the most optimal solutions found over the course of this Chapter. For each solution, 3 scores were calculated; the Cross Validation Error (CVE) from the training set, the predictive accuracy from the testing set and what we term the *'Intra-solution Linkage' score*. This *'Intra-solution Linkage' score* quantifies the strength of the linkages between features within a solution by summing the Mutual Linkage scores for each selected feature, as described in section 5.3.4.2. It is a measure of how much Linkage is present between the selected features in a solution.

Table 5.2 shows the Pearson's correlation coefficients between the solutions' predictive accuracy on the test set, and the measurements derived from the training set; intra-solution linkage and cross-validation error. The solutions are divided into 2 groups; low quality solutions (15-50% error rates on the test set - drawn from all stages of the runs) and high quality solutions (<15 % error

Table 5.2: Table comparing the correlation of solution fitness (CVE Rate) and predictive accuracy on unseen data. In the later stages of the search algorithm (<15% CVE), changes in CVE become less correlated with predictive accuracy. This is contrasted by an increase in correlation between predictive accuracy and Intra-Solution Linkage Scores.

	Score Derived from Training Set			
	Cross Validation Error Rates		**Intra-Solution Linkage Score**	
	Poor Solutions *(>15% CVE)*	**Good Solutions** *(<15% CVE)*	**Poor Solutions** *(>15% CVE)*	**Good Solutions** *(<15% CVE)*
Correlation with Predictive Accuracy on Testing Data	0.7543	0.2411	-0.2263	-0.4296

rates on the test set - solutions found in the final stages of the runs). For the low quality solutions, the correlation between CVE and predictive accuracy is 0.7543. This drops to 0.2411 in the higher quality solutions, which we suspect is due to over-fitting of the test data. The correlation between the intra-solution linkage score and predictive accuracy scores for low quality solutions is low (-0.2263). However, unlike CVE, the correlation magnitude increases in the higher quality solutions (-0.4296). This infers that intra-solution linkage scores may be a better indicator of the generality of solutions than CVE in higher quality solutions (later stages of search algorithms).

In summary, 'good' solutions (that is, with a low error rate on the validation data) have no, or, weak linkage between their selected features. Overfitted solutions (this is, those with low error rate on the training data but high error rate on the validation data), tend to have stronger linkage between their selected features.

5.5 CONCLUSION

The integration of Linkage information in the evolutionary algorithms described in this Chapter provided no significant improvement in the results, or performance of the algorithm as intended by Research Questions RQ1 & RQ2.

When we consider the computational load required to calculate the linkage scores in advance, we would not recommend this form of implementation in real world systems. This is not to say that linkage should be dismissed as a form of guidance in BCI: While this Chapter failed to find a successful application, it was based on only one dataset. It should be noted that further analysis on solutions found by the evolutionary algorithms shows that the correlation between the training set's cross validation error rate, and prediction accuracy, declines in the higher scoring solutions. While this is something that we fully expect as over-fitting occurs, more interestingly, the negative correlation between the solutions predictive accuracy on the test set and the linkage scores within these solutions (derived from the training set) actually increases. This makes sense: we might expect that the classifier would be able to gain more information from features that are not linked (or correlated with each other) than those that are. This suggests that it may be possible to mitigate some of the effects of over-fitting by developing a multi-objective fitness function that gives increasing weight to the solutions that minimise linkage, while concurrently continuing to minimise cross validation error rates.

Part VI

MUTUAL INFORMATION

6

CHAPTER 6 - MUTUAL INFORMATION

Filter based feature selection methods rank variables according to a criterion, independently of the classifier. Examples of these criteria include the Pearson correlation coefficient [176], Fisher score [31], and measures based in Information Theory [10]. The advantages of such techniques are typically less computationally expensive, simpler to implement, and resulting feature subsets are more generalisable as they are not tied to a specific classifier [8]. That being said, they lack the ability to exploit specific characteristics of the machine learning algorithms intended for use, and therefore rarely obtain the highest classification accuracies.

This chapter describes the *Minimum Redundancy Maximum Relevance Iterated Local Search (MRMR-ILS)* algorithm, one of the contributions of this thesis. MRMR-ILS is intended to incorporate mutual information into the operators of ILS, with the goal of finding feature subsets for the creation of models that yield higher predictive accuracies on unseen data. This Chapter has the following structure: a description of Mutual Information is given 6.1. The newly proposed technique MRMR-ILS is described in 6.2 and the methodology used for its testing detailed in 6.3. Results and Discussions are presented in 6.4, followed by Conclusions in 6.5.

6.1 MUTUAL INFORMATION

One of the most prominent and well established measures of a feature's relevance originates from Information Theory, and is known as *Mutual Information*. The following concept definitions explain the mutual information aspects of the algorithm presented by this chapter.

6.1.1 *Entropy*

Entropy is an integral concept within Information Theory, defining the uncertainty of a variable. A well-known measurement of this is Shannon's entropy [159], which measures the number of bits required to represent a variable;

$$H(X) = -\sum_{x} p(x) \log p(x) \,. \tag{6.1}$$

Entropy is calculated by the summation of all the probability distributions, *p(x)*, of values $x \in X$, multiplied by the natural log of those probability distributions. This can be most easily understood when considering a common 6 sided dice. A dice (X) has six sides ($|X|$), where each side (x) is unique. This results in a probability distribution for each of the sides as $1/6$. Using the above equation (6.1), we can see that it results in -6($1/6$.log.$1/6$) = 2.585. That is, we need 2.585 bits to represent all possible values observable from a single dice.

6.1.2 *Mutual Information*

Mutual Information is the unique information shared between two variables. Using entropy, it is possible to quantify the conveyable information from a variable; however, what is often of interest, is how much variables 'overlap' in the information that they convey. This is especially useful when we want to consider how effective one variable is at predicting another; higher shared information suggests that they are measuring a similar source of information.

$$I(X : Y) = H(X) - H(X|Y) = H(X) + H(Y) - H(X, Y) \,. \tag{6.2}$$

To do this, we consider how much information is conveyed by each variable as individuals, in comparison with how much information is conveyed when they are paired. That is, the joint entropy of X and Y, *H(X,Y)*, subtracted from the summed entropies of X, *H(X)* and Y, $H(Y)$. This can also be seen as the amount of uncertainty that can be removed from a variable, when another one is known.

Mutual Information Feature Selection (MIFS) is a technique which selects the top k features after being ranked according to their mutual information with the class label.

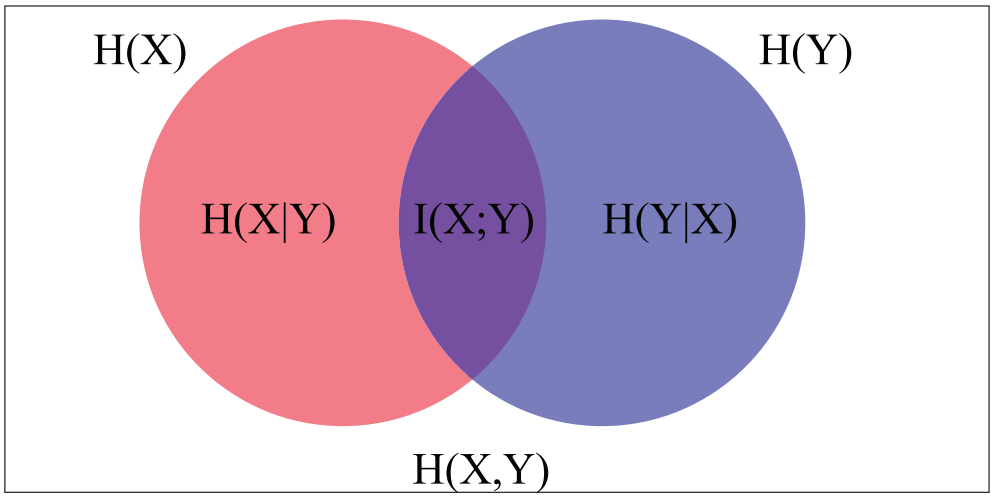


Figure 6.1: Mutual Information between variables X and Y ($I(X;Y)$), seen as the over lap of the entropies of X ($H(X)$) and Y ($H(Y)$).

6.1.3 *Minimal Redundancy Maximum Relevance*

Mutual Information can capture even non-linear interactions between variables, but it is limited due to it being a univariate approach. This is a source of weakness in applications such as feature selection, as we frequently find multivariate interactions between variables and their labels. To solve this, Peng et al. introduced the mRMR approach [133]. This algorithm seeks to address two conditions; maximisation of selected features *Relevance*, and minimisation of their *Redundancy*:

Relevance is defined as:

$$\max D(S,c), \quad D = \frac{1}{|S|} \sum_{x_i \in S} I(x_i;c)\,. \tag{6.3}$$

where $I(x_i;c)$ is the mutual information between each selected feature (x_i) in the subset (S) and the class (c).

Redundancy is defined as:

$$\min R(S), \quad R = \frac{1}{|S|^2} \sum_{x_i,x_j \in S} I(x_i;x_j)\,. \tag{6.4}$$

where $I(x_i;x_j)$ is the mutual information between each pair of selected features within the selected subset (S).

So that *mRMR* can be defined as:

$$\max \Phi(D,R), \quad \Phi = D - R\,. \tag{6.5}$$

mRMR seeks to maximise the distance between the Relevance (D) and Redundancy (R). Figure 6.2 illustrates this as overlapping entropies of features X and Y with class C [133].

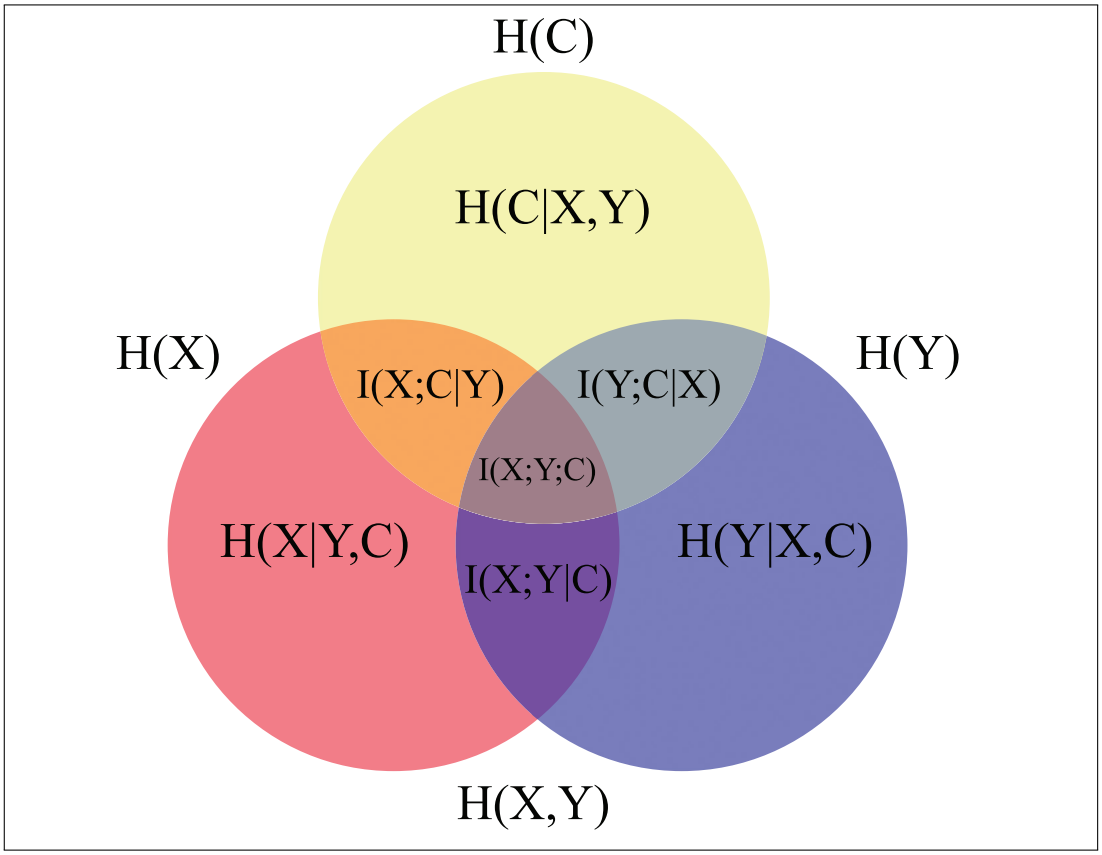


Figure 6.2: Minimum Redundancy Maximum Relevance (mRMR) attempts to maximise the Mutual Information between a variable and its class label ($I(X;C)$), while minimising the Redundant information ($I(X;Y;C)$); or, in other words, information that has already been provided by another variable. If seeking to select feature X, mRMR will seek to maximise $I(X;C \mid Y)$.

6.2 PROPOSED METHOD - MRMR-ILS

We now replicate the existing Iterated Local Search (ILS) algorithm (as defined in Chapter 5), followed by detailing our contribution, the MRMR-ILS.

6.2.1 *Iterated Local Search*

Iterated Local Search is an iterative search-based algorithm that has demonstrated interesting results across a variety of domains [108], but with almost no application to BCI domain. The ILS used in this Chapter consists of a layered search: (*i*) a local search, in the form of a Hill Climbing algorithm; and (*ii*) a diversification mechanism, in the form of a strong mutation, known as a perturbation. A solution is either randomly generated or provided to the algorithm. A Hill Climbing algorithm is then used to search the local space; a candidate solution is created by performing a single point mutation on the current solution. This is achieved by randomly choosing one of the selected features in the current solution, and replacing it with an unselected feature. This is then evaluated by performing 10-fold cross-validation using the training set, which obtains the average prediction error rates on each of the folds.

6.2.2 *Minimal Redundancy Maximal Relevance-Iterated Local Search*

In the MRMR-ILS algorithm proposed in this work, the stochastic perturbation stage of the ILS (as seen in Algorithms 3 and 4) is replaced by an information-measure based selection process (as seen in Algorithm 5). Instead of randomly selecting features for replacement, features are selected for retention based on the information they share with each other, and the label. The mRMR score for each feature is calculated, and those that score most highly (that is, those that have the highest relevance with the label), and have the lowest information overlap with other features within the selected solution, are retained. The remaining features are replaced with unselected features chosen at random.

6.3 METHODOLOGY

The experimental methodology is presented in the following order; classification algorithms used, fitness function, search algorithm parameters, and benchmark methods for comparison.

Algorithm 3 Iterated Local Search

Input: Initial solution is $S_{input} \leftarrow$ generateInitialSolution()

Output: Final solution is S_{best}

```
Let outerLoopLimit = 100
Let innerLoopLimit = 1000
S_best ← S_input
S_bestErrorRate ← evaluteSolution(S_best)
for x = 1 → outerLoopLimit do
    S_best*← perturbateSolution(S_best)
    S_bestErrorRate* ← evaluteSolution(S_best*)
    for x = 1 → innerLoopLimit do
        S'_best ← mutateSolution(S_best*)
        S'_bestErrorRate ← evaluteSolution(S_best*)
        if (S'_bestErrorRate < S_bestErrorRate*) then
            S_best*← S'_best
            S_bestErrorRate*← S'_bestErrorRate
        end if
    end for
    if (S_bestErrorRate* < S_bestErrorRate) then
        S_best ← S_best*
        S_bestErrorRate ← S_bestErrorRate*
    end if
end for
```

6.3.1 *Classifiers*

The key aim of BCI paradigms is to produce an effective model to classify some aspect of neural recordings. The creation of such a model relies heavily on the selection of machine learning algorithm used. In this Chapter, we evaluate two such algorithms:

- *K-Nearest-Neighbours (KNN):* while commonly used in other fields, KNN has been largely neglected within the BCI literature due to its known sensitivity to

Algorithm 4 Iterated Local Search - perturbateSolution

Input: Initial solution is $S_{initial}$

Output: Final solution is $S_{perturbated}$

1: $mutation\_points \leftarrow$ randomIndicies($size\_of\_perturbation$)
2: $new\_features \leftarrow$ randomIndicies($size\_of\_perturbation$)
3: $S_{perturbated} \leftarrow S_{initial}$)
4: $S_{perturbated}(mutation\_points) \leftarrow new\_feature$)

Algorithm 5 MRMR Iterated Local Search - perturbateSolution

Input: Initial solution is $S_{initial}$

Output: Final solution is $S_{perturbated}$

1: $S_{perturbated}(1) \leftarrow$ selectHighestRelevanceFeature($S_{initial}$)
2: **for** $x = 2 \rightarrow size\_of\_perturbation$ **do**
3: feature_scores $\leftarrow$ emptyArray()
4: **for** $y = x \rightarrow size(S_{initial})$ **do**
5: feature_relevance $\leftarrow$ getRelevance($S_{initial}(y)$)
6: feature_redundancy $\leftarrow$ getRedundancy($S_{initial}(y)$, $S_{perturbated}$)
7: feature_scores(y) $\leftarrow feature\_relevance - feature\_redundancy$)
8: **end for**
9: $S_{perturbated}(x) \leftarrow S_{inital}(minimum(feature\_scores))$
10: **end for**
11: $new\_features \leftarrow$ randomIndicies($size\_of\_perturbation$)
12: $S_{perturbated} \leftarrow S_{perturbated} + new\_features$

the 'Curse of Dimensionality' [104]. KNN was selected for use in this work for exploration, and to support our deliberate selection of small feature subsets.

- *Support Vector Machines (SVM):* commonly used in BCI literature, and often obtain the best accuracies. This is thought to be due to their ability to operate with feature sets of a higher dimensionality, and their resistance to overfitting [142].

6.3.2 *Fitness Function*

The fitness of a proposed feature subset was evaluated using k-fold cross-validation on the training data. $k = 10$ was selected due to preliminary experimentation revealing a noisy fitness function originating mainly from the randomly chosen splits in cross-validation. While 10-fold cross-validation creates an expensive fitness function, it is required in such datasets where we find high-dimensionality coupled with low number of samples and poor signal-to-noise ratios [90].

6.3.3 *Search Algorithm Parameters*

Each algorithm was executed 25 times, with 100,000 evaluations of the classifier set as the termination criteria. In each run, there were 100 perturbation 'kicks', and local searches were limited to 1000 evaluation first-improvement Hill Climbing.

6.3.4 *Benchmark Methods*

A selection of benchmark algorithms from the literature were used as comparisons for our algorithm: Mutual Information based filter methods, wrappers, and a state-of-the-art embedded method.

6.3.4.1 *Filters*

Two Mutual Information filter methods were evaluated using a greedy forward-search to select the feature subset size, as used in [94]. Mutual Information Feature Selection (*MIFS*), relies on selecting features that increase the selected subsets' Mutual Information with the class label. *mRMR* seeks to maximise the selected subsets' Mutual Information with the class label (relevance), while minimising the Mutual Information between features (redundancy).

6.3.4.2 *Wrappers*

Two wrapper approaches were selected for comparison: *Sequential Forward Search* (SFS) - a greedy algorithm that selects the next best feature as evaluated by the classifier; and *Iterated Local Search* - a two layer search involving perturbations and local searches.

SFS is a very popular technique, and is often used as an exploratory measure in feature selection. ILS has been used in a wide variety of different search areas, but has not been used in BCI literature prior to this thesis.

6.3.4.3 *Embedded*

Least Absolute Shrinkage and Selection Operator (LASSO) (or L1 regularisation) performs feature selection by reducing the sum of the absolute values of the model parameters below an upper bound. It does this by shrinking the coefficients of the features, often to zero, effectively deselecting them. It can provide two feature subsets: Sparse, and Mean Squared Error (MSE). This method provides relatively poor cross-validation error rates on the training set, but tends to be reasonably more generalisable.

6.4 RESULTS AND DISCUSSION

Table 6.1 and 6.2 present results obtained using the KNN and SVM classifiers respectively. The list of measures are: the number of features selected by each algorithm (**Selected f**); the average final solutions' fitnesses (cross-validation error rate on training data; **CVE**, where lower is better); and their **Accuracy** on the unseen, testing data. The datasets were labeled: *D1* - Berlin BCI Competition II Dataset III; *D2* - Berlin BCI Competition II Dataset IV; *D3* - Subject A from the Riken dataset.

When using a KNN classifier, it is observed in Table 6.1 that the MRMR-ILS finds solutions with the lowest cross-validation error rates on two datasets: D1 (10.6%) and D2 (27.23%). On dataset D3, it achieved the second lowest (14.92%), only just behind the SFS (13.85%). In all three cases, the MRMR ILS outperformed the unguided ILS. These cross validation error rates reflect the

Table 6.1: Results of each feature selection algorithm while using the KNN Classifier. Number of selected features, CVE rates, and accuracy is shown for Datasets D1, D2 and D3. Values in bold denote the highest performing algorithm for each measure.

Dataset	Algorithm	Selected f	CVE	Accuracy
D1	**GLFS**	96	0.3194	0.6786
	MLFS	60	0.3114	0.6
	MIFS	20	0.4105	0.6000
	MRMR	43	0.3295	0.7286
	LASSO (Sparse)	8	0.2186	0.7143
	LASSO (MSE)	29	0.1993	0.7143
	SFS	14	0.1357	0.7357
	ILS	**6**	0.1114±0.0082	**0.7926±0.0366**
	MRMR ILS	**6**	**0.106±0.0073**	0.792±0.027
D2	**GLFS**	100	0.433	0.56
	MLFS	82	0.4376	0.56
	MIFS	10	0.4839	0.5600
	MRMR	34	0.4754	0.5200
	LASSO (Sparse)	11	0.4269	0.5500
	LASSO (MSE)	13	0.4222	0.5500
	SFS	15	0.2816	0.6200
	ILS	**6**	0.2738±0.0148	0.6148±0.0474
	MRMR ILS	**6**	**0.2723±0.0087**	**0.6412±0.037**
D3	**GLFS**	17	0.3858	0.4403
	MLFS	18	0.4089	0.4925
	MIFS	6	0.5172	0.6194
	MRMR	30	0.4772	0.5224
	LASSO (Sparse)	**4**	0.2408	0.6045
	LASSO (MSE)	15	0.2615	0.5672
	SFS	14	**0.1385**	0.5896
	ILS	**4**	0.1539±0.0107	0.5997±0.0258
	MRMR ILS	**4**	0.1492±0.0104	**0.6085±0.0198**

algorithms' performance on unseen data by achieving the highest accuracy on datasets D2 (64.12%) and D3 (60.85%), with the second highest accuracy on dataset D1 (79.2%).

In Table 6.2, the SVM classifer produces results with a similar pattern as using the KNN, with the MRMR-ILS achieving the lowest cross-validation error rates in dataset D1 and D3 (8.843% and 7.72% respectively), and behind the ILS by just 0.17% on dataset D2. Classification accuracies on unseen datasets in this case are slightly more nuanced; the MRMR-ILS achieved the highest accuracy on dataset D2 (69.48%). and in dataset D1 it achieved the second highest accuracy of 82.69%, just behind the ILS (84.23%). Dataset D3 presented slightly more unusual results, specifically in regards to the Greedy Linkage algorithms introduced in Section 5.3.4.1. These see generally poor performance across all datasets, except for D3 when using a SVM. In this case, the GLFS and MLFS outperformed the ILS variants.

In order to assess if there was a significant difference between the performance of the ILS and MRMR-ILS algoritms, a One-Way MANCOVA was performed. There was no statistically significant difference between the ILS and MRMR ILS on the combined dependent variables (cross validation error rate and accuracy on unseen data) after controlling for Datasets (D1, D2, D3) and Classifiers (KNN , SVM), $F(2, 295) = 1.893$, $p = .152$, Wilks' $\lambda = .987$, partial $\eta^2 = .013$.

6.4.0.1 *Post Hoc Analysis*

To further analyse the resulting behaviours of the ILS and MRMR-ILS algorithms, 2 additional avenues were explored; comparison of the features most commonly selected by each algorithm, and the relation between the expected model performance (cross validation error rate) and its performance on unseen testing data.

A comparison of selected features was necessary to ensure that the final features selected by each algorithm differ when an Information Measure is included in the wrapper. For explanations of what each feature index on the x axis represents in the graphs 6.3, 6.4, and 6.5, please see Appendix A.1,A.2, and A.3. When comparing the features selected by the algorithms in experiments

Table 6.2: Results of feature selection algorithm while using the SVM Classifier with selected subset sizes (Selected f). Values in bold denote the highest performing algorithm for each measure.

Dataset	Algorithm	Selected f	CVE	Accuracy
D1	**GLFS**	79	0.4206	0.6143
	MLFS	100	0.3014	0.6643
	MIFS	20	0.3740	0.6071
	MRMR	43	0.2581	0.7929
	LASSO (Sparse)	8	0.1493	0.7929
	LASSO (MSE)	29	0.1757	0.7929
	SFS	8	0.0857	0.8071
	ILS	**6**	0.0846±0.0053	0.8423±0.0287
	MRMR ILS	**6**	**0.0843±0.0071**	**0.8269±0.0258**
D2	**GLFS**	70	0.3992	0.57
	MLFS	92	0.3705	0.59
	MIFS	10	0.4153	0.5200
	MRMR	34	0.3997	0.5800
	LASSO (Sparse)	11	0.3095	0.6700
	LASSO (MSE)	13	0.3168	0.6200
	SFS	**9**	0.2532	0.6200
	ILS	12	**0.2422±0.0074**	0.6836±0.0384
	MRMR ILS	12	0.2439±0.0095	**0.6948±0.0347**
D3	**GLFS**	95	0.3009	**0.6642**
	MLFS	98	0.2929	0.6493
	MIFS	6	0.4077	0.5373
	MRMR	30	0.2800	0.5672
	LASSO (Sparse)	**4**	0.2377	0.6045
	LASSO (MSE)	15	0.1508	0.6567
	SFS	17	0.1000	0.5970
	ILS	15	**0.0735±0.0094**	0.6197±0.043
	MRMR ILS	15	0.0772±0.0093	0.6391±0.0287

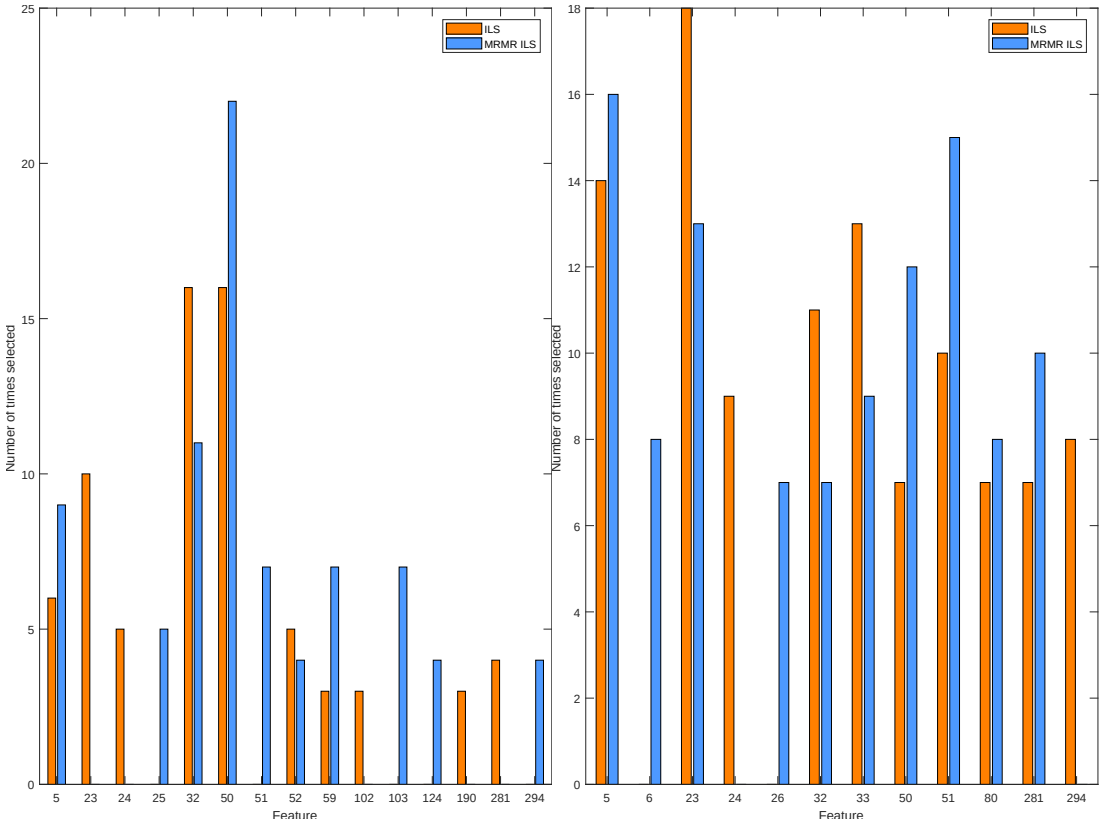
involving Dataset D1 (Berlin BCI Competition II dataset III), the KNN based searches shared 5 of the top 10 most commonly selected features, while the SVM based searches did so in 8.

For features selected from Dataset D2 (Berlin BCI Competition II dataset IV), the KNN based searches selected the same features in 7 of the top 10 features, while the SVM based searches did so in 5. A notable difference in the features selected by the SVM based algorithms is the preference for Channel C5 (features 146 and 153), Cz (191) and C2 (200) in the beta band (8-30Hz), whereas the MRMR ILS algorithm more commonly selected CP2 (281), CP4 (294) and O1 (313 and 317) of the alpha band (8-13Hz).

Features selected by KNN based algorithms in Dataset D3 (Riken) were dominated by 4 features - epoch 2 and 3 of the higher frequencies within the beta band, 20-26 Hz. Outside these 4 primary features, features 179 and 202 were commonly selected by both algorithms which are also from channels and frequencies neighbouring the 4 main features. The SVM based ILS demonstrates somewhat less stability in its feature selection, with its 10 most commonly selected features being selected a similar number of times. The MRMR ILS on the other hand, has a strong preference for features 107 and 179 which correspond to a narrow frequency band of 23-26 Hz at seconds 2 and 3, closely followed by the same frequency at second 3 on channel CP4.

As we can see from the features selected by the ILS and MRMR ILS, certain features are found by both algorithms. Where differences can be observed, is what additional features are selected. In other words, what 'supporting features are selected'. This is perhaps a logical outcome to the inclusion of the multivariate measure of Information; taking the relationship between a feature and a label into account will give preference to certain features. Including the inter-dependencies within the solution, as in mRMR, will give preference to features that compliment those with the highest shared information with the label.

Figures 6.6a, 6.7a, 6.8a, 6.9a, 6.10a, and 6.11a show the average incumbent solution fitness based on the cross-validation error rates over each iteration of the ILS and MRMR-ILS algorithms. In a post-hoc analysis, we extracted these incumbent solutions and re-evaluated their predictive accuracy on the testing



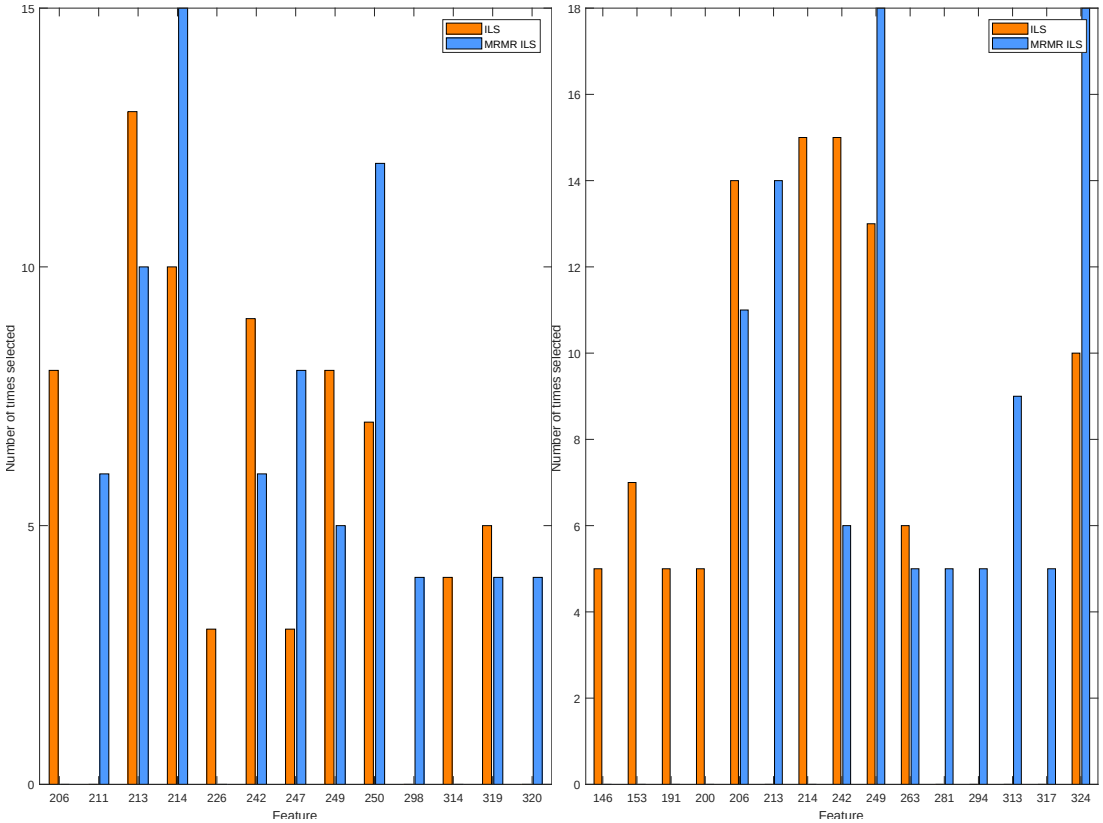
(a) Selected using KNN

(b) Selected using SVM

Figure 6.3: Comparison of the most selected features of the ILS and MRMR-ILS algorithms on dataset *D1* - BCI Competition II dataset III

data, plotted in Figures 6.6b, 6.7b, 6.8b, 6.9b, 6.10b, and 6.11b. We can see that the relationship between the MRMR-ILS fitness function, and the performance on unseen data is much stronger than that observed in the ILS.

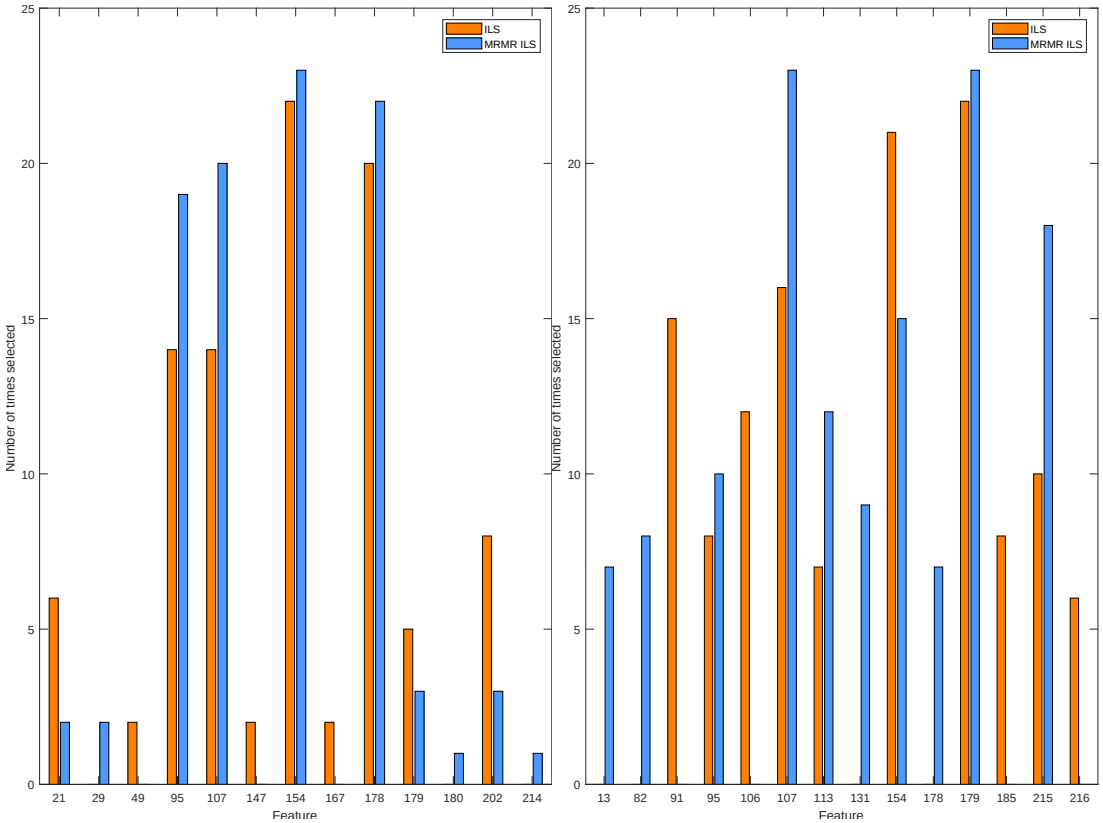
In order to find a real-world feature subset for BCI applications, it is imperative that the estimated accuracy provided by the fitness function in our algorithms correlates as closely as possible to accuracy rates obtained from new, unseen data. We further explore this in Table 6.3, in which the Pearson's correlation coefficient is calculated for the cross-validation error rates and accuracies of the incumbent solutions. In five of the six test cases, there is a substantially higher correlation between the predicted accuracy (CVE rate) and the accuracy on the unseen data in the MRMR ILS than that of the ILS. The most notable examples of this is the use of KNN in dataset D1, and the use of SVM in dataset D3, where the correlations seen within the solutions of the ILS have weak negative correlations (-0.1512 and -0.3787), which is heavily



(a) Selected using KNN (b) Selected using SVM

Figure 6.4: Comparison of the most selected features of the ILS and MRMR-ILS algorithms on dataset *D2* - BCI Competition II dataset IV

contrasted against the strong negative correlations in those of the MRMR ILS (-0.9275 and -0.7203).



(a) Selected using KNN (b) Selected using SVM

Figure 6.5: Comparison of the most selected features of the ILS and MRMR-ILS algorithms on dataset *D3* - Riken

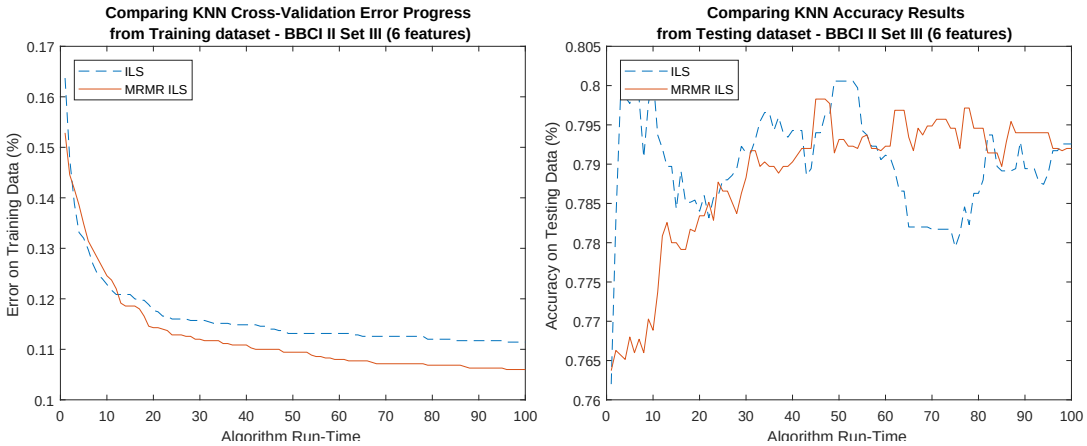
6.5 CONCLUSION

In this chapter MRMR-ILS was proposed, a hybrid Filter-Wrapper method involving Mutual Information for feature selection. Evaluations over three datasets using KNN and SVM classifiers demonstrated that feature subsets found by our method were typically of higher quality with lower error rates on training sets and higher accuracy on testing data, than those found by the compared traditional methods.

What is of additional interest is the quality of the solutions found during the search process of the MRMR-ILS in comparison to those of the ILS. Relying solely on the cross-validation error rates allowed feature subsets to be discovered that were highly effective for creating models that represent the training data. However, when these feature subsets were tested on unseen data, the predictive accuracies did not reflect those acquired from the cross-

Table 6.3: Correlations between Cross Validation Error Rates and Accuracy of Solution during ILS and MRMR-ILS Search. Figures in bold denote the highest performing algorithm for each measure.

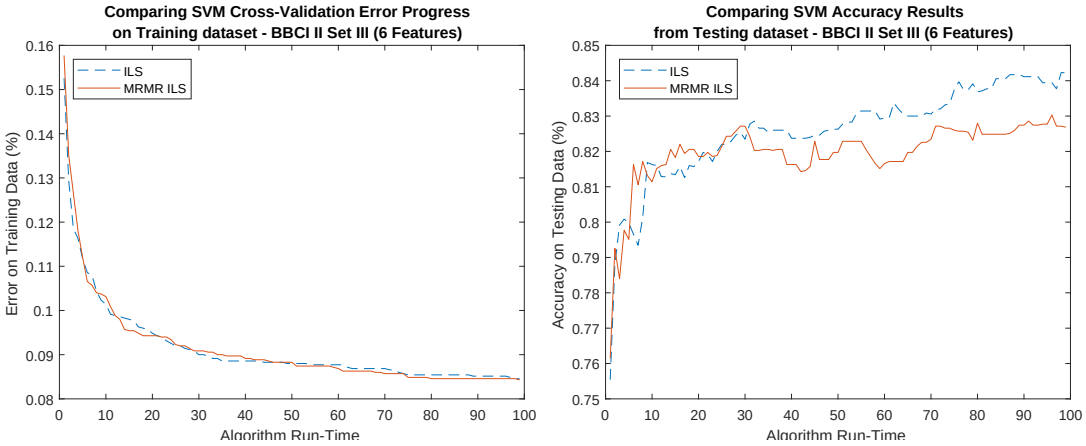
Classifier	Dataset	Algorithm	
		ILS	**MRMR ILS**
KNN	**D1**	-0.1512	**-0.9275**
	D2	-0.7131	**-0.9116**
	D3	-0.9224	**-0.9686**
SVM	**D1**	**-0.9370**	-0.9100
	D2	-0.8348	**-0.8619**
	D3	-0.3787	**-0.7203**



(a) Cross-Validation Error Rates

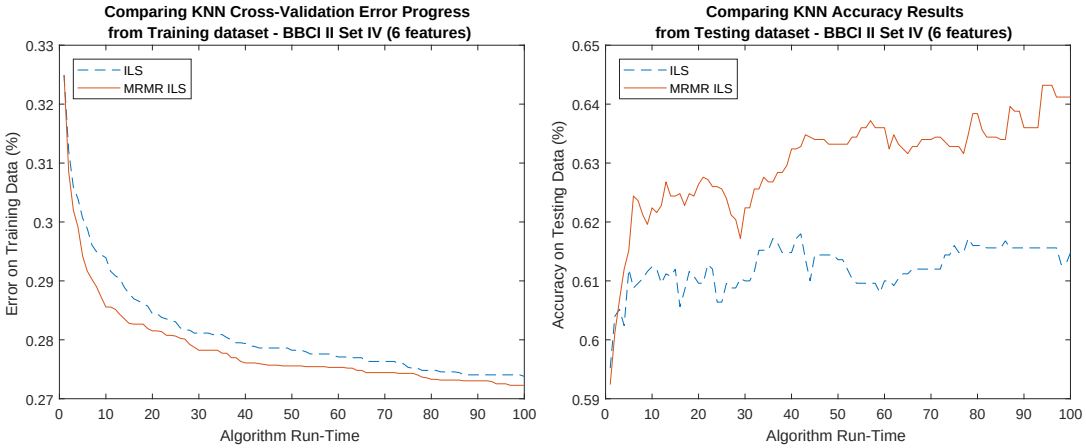
(b) Accuracy on unseen data

Figure 6.6: Comparison between ILS and MRMR-ILS over each iteration of the algorithms for the KNN classifier on dataset *D1* - BCI Competition II dataset III



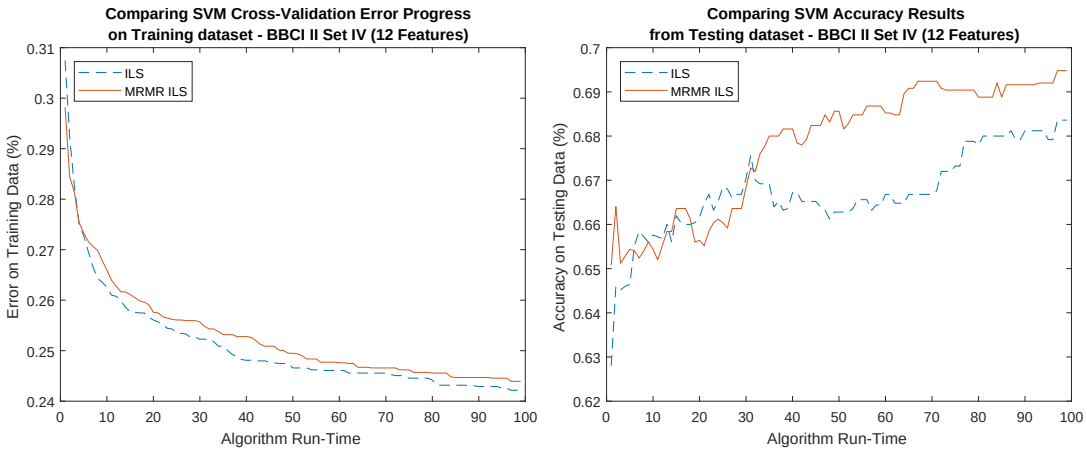
(a) Cross-Validation Error Rates (b) Accuracy on unseen data

Figure 6.7: Comparison between ILS and MRMR-ILS over each iteration of the algorithms for the SVM classifier on dataset *D1* - BCI Competition II dataset III



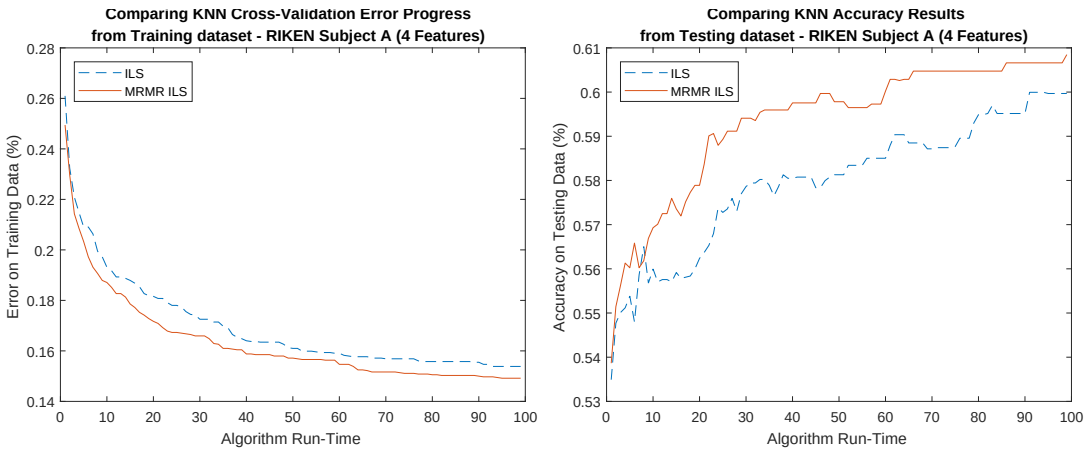
(a) Cross-Validation Error Rates (b) Accuracy on unseen data

Figure 6.8: Comparison between ILS and MRMR-ILS over each iteration of the algorithms for the KNN classifier on dataset *D2* - BCI Competition II dataset IV



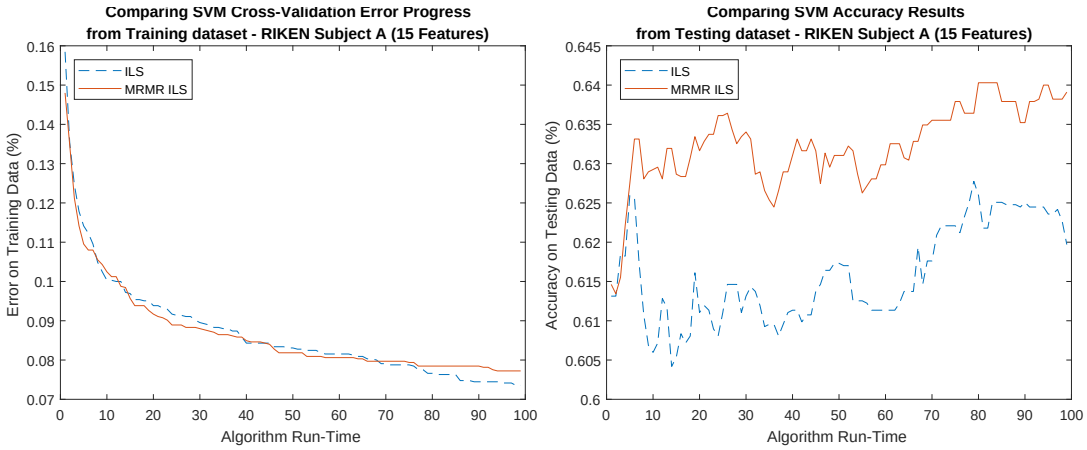
(a) Cross-Validation Error Rates (b) Accuracy on unseen data

Figure 6.9: Comparison between ILS and MRMR-ILS over each iteration of the algorithms for the SVM classifier on dataset *D2* - BCI Competition II dataset IV



(a) Cross-Validation Error Rates (b) Accuracy on unseen data

Figure 6.10: Comparison between ILS and MRMR-ILS over each iteration of the algorithms for the KNN classifier on dataset *D3* - RIKEN Subject A



(a) Cross-Validation Error Rates

(b) Accuracy on unseen data

Figure 6.11: Comparison between ILS and MRMR-ILS over each iteration of the algorithms for the SVM classifier on *D3* - RIKEN Subject A

validation error rates. This is likely due to over-fitting; creating models that are highly fitted to the training data leads to poor generalisation on new datasets. The end result is a classifier which may not be fit for purpose.

When MRMR was incorporated into the algorithm, the search was partially constrained to areas of the search space rich in mutual information. This resulted in models that generalised to unseen data with predictive accuracies that were much more consistent with the CVE from the training data. Further experimentation should seek to compare the MRMR-ILS with other Mutual Information based hybrid methods from the wider feature selection literature, and investigate the relationship between Mutual Information, cross-validation error rates, and predictive accuracy on unseen data.

Part VII

INSTANCE TRANSFER

7

CHAPTER 7 - INSTANCE TRANSFER

Previous chapters confirm the varying performance of feature selection on different datasets within Brain Computer Interfaces (BCI). It has previously been shown that models can be weakened by low numbers of training samples, and that appropriate feature selection can improve this.

In this Chapter we introduce a new proof-of-concept method for optimising the performance of BCI while minimising the quantity of required training data. This Chapter also proposes that instances from one participant may be used in the modelling of another. This is achieved by using an evolutionary approach to rearrange the distribution of training instances, prior to the construction of an *Ensemble Learning Generic Information (ELGI)* model. The training data from a population can be optimised to emphasise generality of the models derived from the data, prior to a re-combination with participant-specific data via the ELGI approach, as well as the training of classifiers. Evidence is given to support the adoption of this approach in the more difficult BCI conditions: smaller training sets, and those sets suffering from temporal drift.

7.1 TRANSFER LEARNING IN BCI

As described in Section 1, BCIs are difficult to calibrate due to recordings having a low signal to noise ratio. This is further compounded by the non-stationary nature of brain signals: neural patterns not only differ between participants, but are also subject to *temporal drift*, where data obtained from a single participant changes drastically over time [80]. *Zero Training systems*, trained exclusively on participants from previous sessions, are an ideal goal but this non-stationarity means highly accurate zero training systems may not be possible. Consequently, we must instead focus on minimising the participant-

specific training information required by optimising the applicability of data available from alternative sources.

This Chapter proposes a novel method for the optimisation of the distribution of instances within a database of sets recorded from previous participants, in a manner that ensures that they can be used to create an ensemble that is maximally general to the population. This database is then used to seed a previously established method, ELGI, that recombines instances obtained from different participants with small quantities of participant-specific data, to create a robust participant-specific ensemble. The aim is to create a BCI that requires only a small amount of training data, and should retain accuracy over time in a way that a traditional BCI does not. This is achieved by moving instances between previously obtained datasets via a random mutation Hill Climbing algorithm.

7.1.1 *Ensembles*

Ensembles have been used in a number of different BCI applications to increase accuracy and reduce the amount of training data required for participants. Most BCI ensembles use naive partitioning in which the instances are divided by their associated labels, whether it be by source domain or by stimuli. This proves useful for weighting classifiers within the ensembles; allowing information regarding the appropriateness of each model and the test-domain to be extracted [103]. It was demonstrated by Onishi et al. [128] that overlapping these naive divisions can actually increase accuracy, suggesting that having the same training data duplicated amongst the classifiers can benefit the overall performance.

In 2015, Xu et al. [187] introduced the ELGI approach. Rather than using the small amount of available training data to train a classifier, or for weighting the models within a larger ensemble trained on the data of other participants, ELGI combines the participant-dependent data with participant-independent data to form a hybrid ensemble. This is achieved by splitting the datasets of each existing participant within the database into target and non-target sets. The removed missing instance class (target or non-target) is then replaced by

a copy of the corresponding class from the participant-specific training data. This results in an ensemble consisting of $2n - 1$ classifiers, where n is the number of participants within the database.

This chapter proposes a new technique in which the database containing the previously recorded participants' datasets are optimised to create an ensemble that is maximally generalised for the population, prior to the combination process of ELGI. The procedure is outlined fully in Section 7.3.

7.2 METHODOLOGY

This section defines the BCI Paradigm used and also describes the datasets. It then goes on to describe the offline filtering applied to the data and finally defines the algorithms to be compared in the experiments.

7.2.1 *Dataset*

This chapter uses Dataset D4, detailed in Section 4.1. Hoffmann [76] provided a dataset using the P300 paradigm, in which 8 participants were recorded. Each participant (P), was recorded over 4 sessions (S), each with 6 runs (R), each run consisting of 20 rounds (I), and each round consisting of 6 binary tasks (t). This dataset is ideal for experimentation within this chapter as it includes characteristics that allow us to investigate: Different participants; varying levels of neural impairment; sub-divisions of training data; and, recordings carried out over a series of time periods.

7.2.2 *Classifier*

A *Bayesian Linear Discriminate Analysis (BLDA)* classifier (as by Hoffmann [76]) was used. Each stimulus presentation was treated as a binary problem, and the Bayesian probability of the prediction was recorded. Due to the paradigm structure, every subdivision of 6 stimuli presentations has 1 target and 5 non-target. These groupings are deemed as a 'round'. A prediction is made

based on the highest probability within each round. In each run, 20 rounds of all 6 stimuli are presented. This allows the Bayesian probabilities of each round to be summed with previous predictions, increasing predictive accuracy over the course of the run. This can be seen in Figure 7.1.

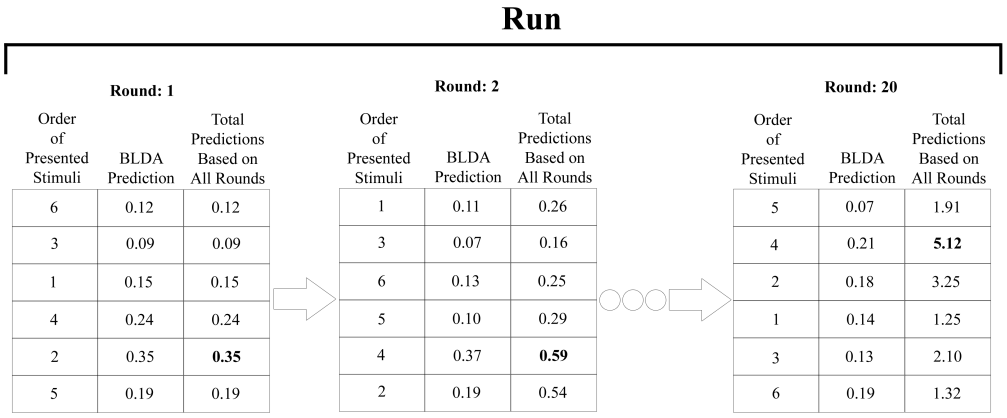


Figure 7.1: Diagram describing how the paradigm is divided into smaller sub-problems within each run. Twenty rounds of all 6 stimuli are presented to the participant and the Bayesian probability of a positive label assigned. The probabilities of each label are summed over twenty rounds to make a prediction for the run.

7.2.3 *Conditions*

The complex nature of BCI allows a number of different factors to be considered:

7.2.3.1 *Quantity of Participant-Specific Data*

As a primary aim in BCI is to minimise the required participant-specific training data, the impact of training set size was explored. The datasets follow a common hierarchical structure; each participant recording 4 sessions of 6 runs. All models were trained with data from the first session and 3 training set sizes of 3, 4, and 5 runs were used.

7.2.3.2 *Time Between Testing Sessions*

A major challenge in BCI, other than between-participant transference, is between-session transference for single participants. As neural drift occurs over time, highly fitted models tend to lose accuracy. All models were tested on data acquired from 3 sessions, recorded over 2 days; session 2 on the same day as the training data, and sessions 3 and 4 on a day no more than 2 weeks later.

7.2.4 *Compared Algorithms*

Three approaches were compared in our experiments, two taken from the literature (SLII and ELGI) and the following proposed new method (eELGI):

STANDARD LEARNING INDIVIDUAL INFORMATION (SLII) : a Bayesian LDA model trained using participant-specific data exclusively. The binary task with the highest probability in each round was selected as the target, and the rest, assumed to be non-targets [187].

ENSEMBLE LEARNING GENERIC INFORMATION (ELGI) : the ELGI method [188] creates an array of classifiers by utilising the participant-specific (data recorded from the 'current' participant) and participant-independent datasets (data recorded from 'previous' participants) in the following manner:

$$[C_{2N}] = \sum_{i=1}^{N} [C(P_i^T + P_k^{NT}), C(P_i^{NT} + P_k^T)]$$

The training data P from each participant P_i is split into two subgroups; target T and non-target NT. A copy of the target instances from the test-participant k (P_k^T) are then added to the non-target subgroup P_i^{NT}, and conversely, a copy of the test-participant's non-target instances P_k^{NT} are added to the target subgroup P_i^T. Each of these new subgroups are used to train an ensemble of classifiers C. Predictions Pr are made by each classifier in the ensemble based on the unseen data from the test-participant P_k^x, and these predictions are collated. This is done using the Sum Rule voting method where the Bayesian

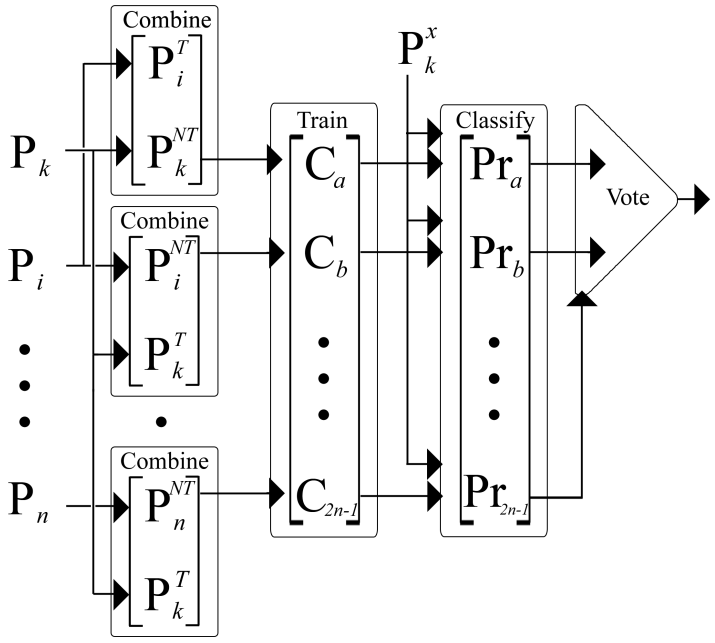


Figure 7.2: ELGI approach displaying that two classifiers are trained for every participant in the database P_i by a splitting and recombination of their target P_i^T and non-target P_i^{NT} instances with the corresponding instances from the test-participant's training data P_k. These classifiers are then used to make predictions on the test-participants unseen data P_k^x. Finally, these predictions are collated via voting.

posterior probabilities are summed for each class. This is further depicted in Figure 7.2.

EVOLVED ENSEMBLE LEARNING GENERIC INFORMATION (EELGI) : the novel approach proposed in this Chapter, is described fully in Section 7.3. In this, we assume that the natural grouping of instances by participant is not optimal. Instead, an evolutionary algorithm transplants instances between datasets taken from each participant, aiming to maximise the generalisability of each set in reference to other previously recorded participants, prior to their combination with participant-specific data via the ELGI.

7.3 EVOLVED ELGI ENSEMBLE

We propose a new approach whereby the database containing the previous participants' datasets is optimised, with the goal of creating an ELGI ensemble that better generalises to the population. This is achieved by a *leave-one-out* technique in which a participant's bin, the subset containing all data from that participant, is selected at random and a portion of the instances obtained from that participant are moved into the bin of another randomly selected participant. Two models are then trained: one using the data from the bin that was selected for transfer, and one from the bin that was selected as the destination. These models make predictions on the data in the remaining unselected bins. The resulting overall predictive accuracy is used as the fitness function for a random mutation Hill Climbing algorithm. This seeks the allocation of training data to bins that maximises the predictive accuracy within the database.

The implementation is now described in more detail. The procedure is given formally in Algorithm 6. The search is seeded with a solution consisting of 7 bins; each consisting of an individual's data but excluding any information from the new participant, as in the Zero Training Model. A 500 iteration Hill Climbing algorithm was then applied with the following mutation operator and fitness function.

MUTATION (MOVE OPERATOR) The move operator selects a target bin, a, and a destination bin, b, at random from the training set bins; a subset m with 10% of the target bin's instances are moved into the destination bin. Subsets P_{ea} and P_{eb} are created by removing subset m from P_a and appending it to P_b, respectively.

FITNESS FUNCTION To assess the fitness of the candidate solution, 2 classifiers C_{ea} and C_{eb} were trained from the subsets P_{ea} and P_{eb}. These were then used to make predictions on the remaining instances within all subsets P, excluding the participant datasets selected for mutation (P_a and P_b). The average round accuracy over all the non-selected bins was calculated for both

Algorithm 6 Evolution of instances in eELGI

Input: Initial solution is $P = \mathcal{P}(P_i)$

Output: Final solution is Modified $P = \mathcal{P}(P_i)'$

```
for x = 1 → 500 do
    Choose a and b from 1 : N where N is the |P|
    Create m ⊂ P_a
    P_ea ← P_a with m removed
    P_eb ← P_b appended with m
    Train classifiers C_a and C_b with P_a and P_b
    Train classifiers C_ea and C_eb with P_ea and P_eb
    f_a = 0, f_b = 0, f_ea = 0, f_eb = 0
    for i = 1 → N do
        if i ≠ a && i ≠ b then
            f_a = f_a + C_a(P_i), f_b = f_b + C_b(P_i)
            f_ea = f_ea + C_ea(P_i), f_eb = f_eb + C_eb(P_i)
        end if
    end for
    if f_a < f_ea && f_b < f_eb then
        P_a = P_ea, P_b = P_eb
    end if
end for
```

models affected by the mutation (f_{ea} and f_{eb}); a solution was deemed successful if the fitnesses obtained were an increase over the fitness (f_a and f_b) of both models created from the incumbent solution (C_a and C_b). The mutation was rejected if it caused a decrease in accuracy within either model.

This evolved dataset was then used to seed the original ELGI from [187].

7.4 RESULTS

Figure 7.3 presents the performance of the SLII, ELGI and eELGI algorithms averaged across all 8 participants. Rows 1, 2 and 3 show performance of models with 3, 4 and 5 runs (see Section 7.2.3) of training data available, respectively.

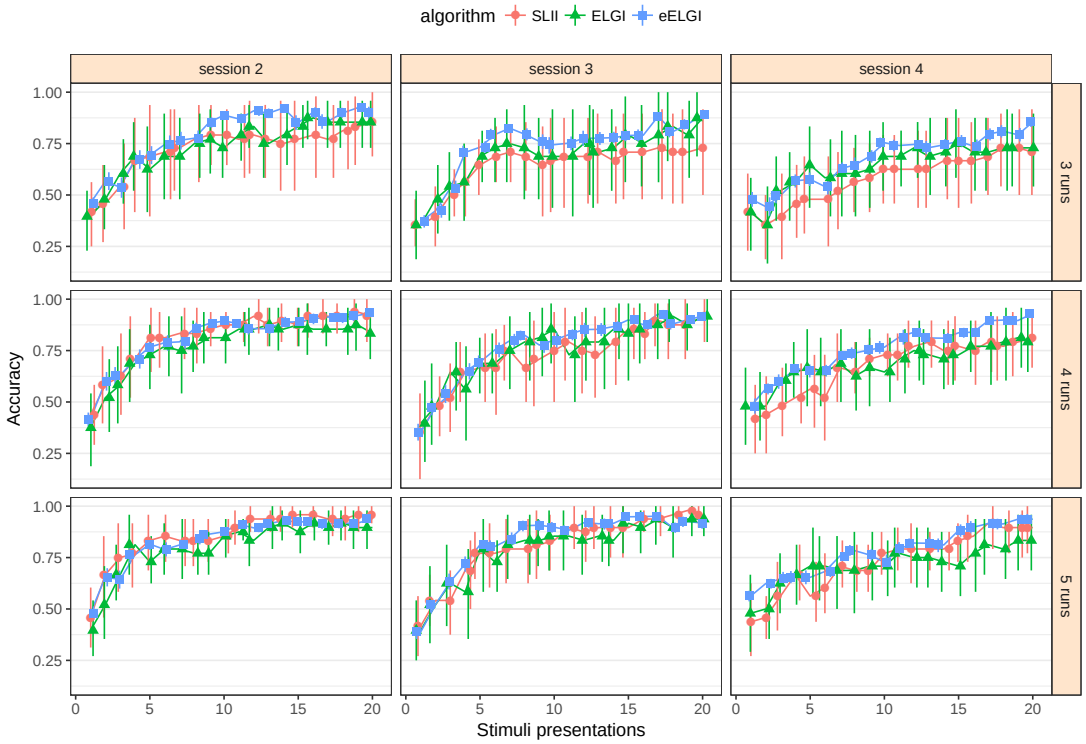


Figure 7.3: Algorithm performance by number of stimuli presentations, with differing quantities of participant-specific training data available. Error bars show the confidence intervals around the means. Horizontal jitter has been added to improve discernibility.

Columns display performance over three different testing sessions. While the confidence intervals of the different approaches vary due to differing sample sizes, the SLII and ELGI are almost indiscernible. The mean line of the eELGI is typically higher than that of the other algorithms, with its smaller confidence interval often visibly higher. The instances in which notable improvements are made are in the extremity conditions: low availability of participant specific data (row 1) and the testing session farthest from the training session (column 3).

The Round Accuracy is presented in Figure 7.4 for the SLII, ELGI and eELGI algorithms. It is displayed by participant with each point representing the accuracy achieved with 3, 4 and 5 runs of training data provided for training. Increases in the quantity of participant-specific training data increases the predictive accuracy in each participant, except 6. Participant 5 is the outlier in terms of variance; increases in participant-specific training data makes a much more substantial change to this classifier's accuracy than others. When

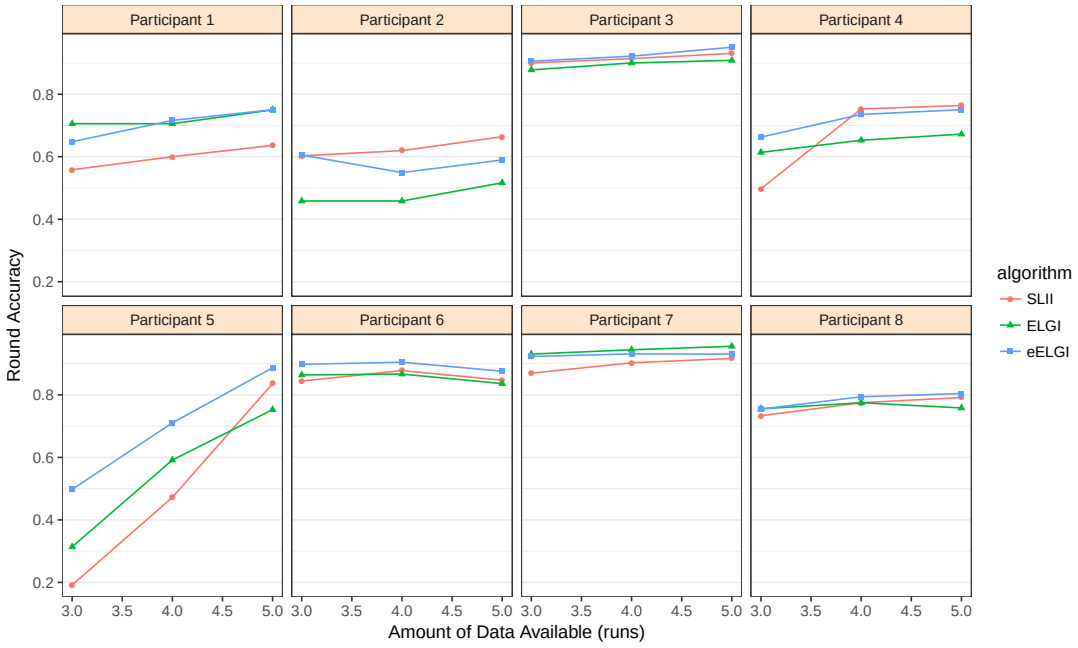


Figure 7.4: Round Accuracy over all testing sets displayed for each quantity of participant-specific training data, separated for each participant.

considering overall round accuracies across differing training set sizes, eELGI performed better than the SLII and ELGI in 62.5% of cases, and obtained the second best results in the remainder. In no cases was eELGI the worst performer.

Figure 7.5 demonstrates each algorithm's resilience to neural drift over time. The round accuracy of the SLII, ELGI and eELGI over each of the testing sessions is given. A decrease in predictive accuracy was observed between session 2 and session 3 in 62.5% of the cases, and a decrease between session 3 and 4 in 58.3%. Overall, a decrease in predictive accuracy between session 2 and 4 was observed in 79.2% of the cases, as expected due to temporal neural drift. For 5 of the 8 participants, the eELGI retained the highest round accuracy after two weeks, while still maintaining relativity high accuracy in the remaining three.

To analyse the differences between each algorithm's effectiveness in mitigating the effects of neural drift over time, hierarchical linear models were used as recommended by Locascio [102]. The results of these are given in Figures 7.6a and 7.6b. In Figure 7.6a, lines show the expected average behaviour when considering the variation across participants, with points representing the residual deviation of each participant from the estimated common behaviour.

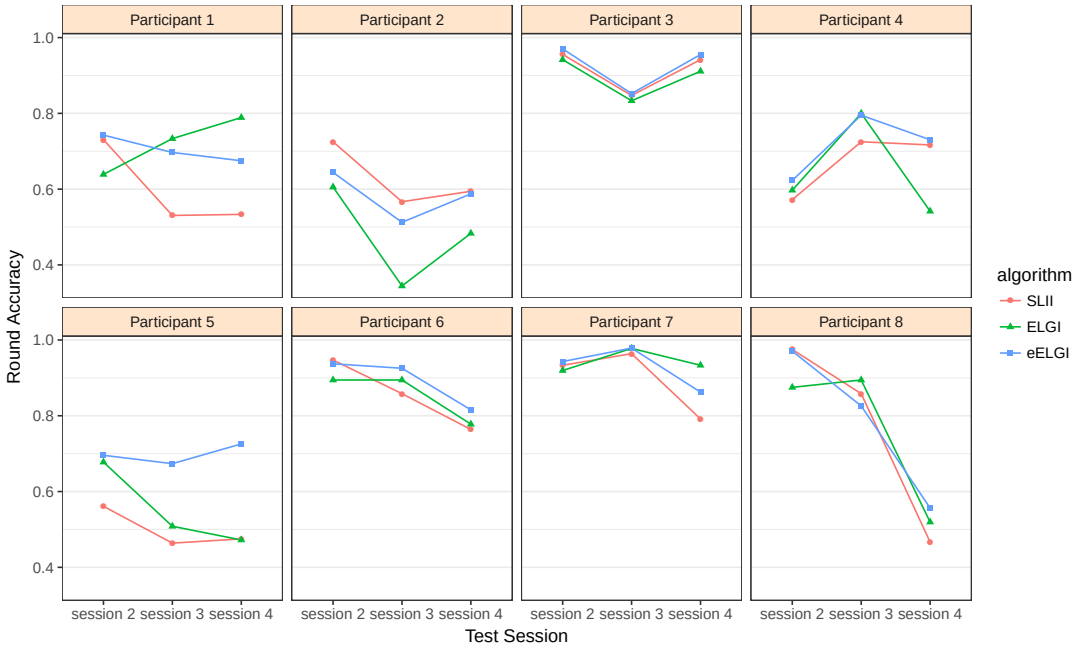
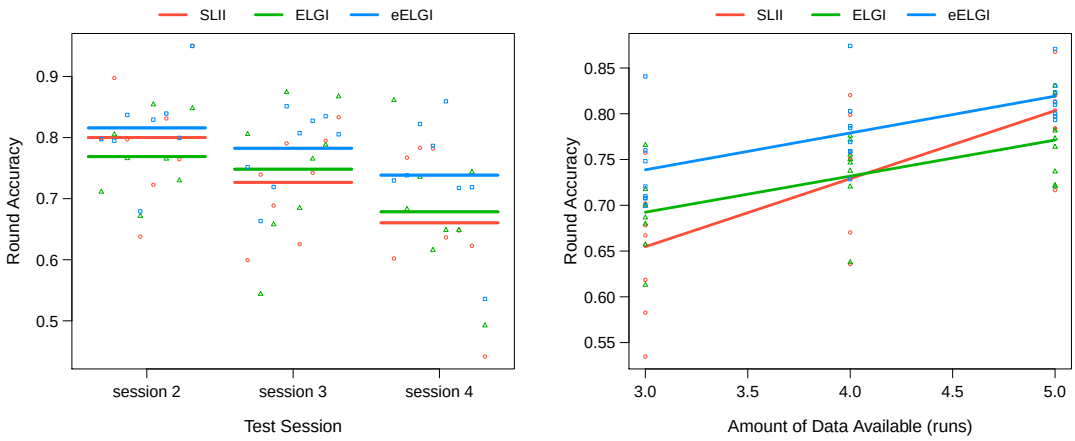


Figure 7.5: Round Accuracy over all quantities of training data for each testing set, separated for each participant.

Although no statistical significance can be claimed here, the trends suggest that in all 3 testing sessions, the eELGI performed better than both the SLII and ELGI. It should also be noted that there appears to be less variance within and between testing sets for the eELGI. This suggests that the eELGI not only performs better than the other algorithms, but is also less susceptible to neural drift over time.

As seen in Figure 7.6b, the round accuracy of all 3 algorithms increases with the amount of participant-specific data available. The SLII is most dependent on the quantity of participant-specific data, with ELGI performing much better when fewer training instances are available. However, this advantage is lost as volume of training data increases. The eELGI line has a similar slope to the ELGI (0.0402 and 0.0394, respectively) but with a higher y-intercept (0.618 to 0.574), resulting in better overall performance than both the SLII and ELGI in all 3 conditions. In fact, a post-hoc Tukey's comparison of the model estimates, averaging over algorithm-data interactions [78], showed that the eELGI produced a statistically significant increase in round accuracy over the SLII ($p = 0.0387$) while the ELGI did not ($p = 0.1483$). Therefore, with respect to the ELGI, the effect of evolving the base dataset appears to increase the



(a) Accuracy over time from training.

(b) Accuracy over available training data.

Figure 7.6: Fit of hierarchical linear models, with random effects for each participant, estimating (a) the overall Round Accuracy per testing set and (b) the change in Round Accuracy over training set size.

intercept, without having any adverse affects to the rate of improvement seen when increasing participant-specific data.

7.5 DISCUSSION AND CONCLUSION

This chapter proposed the eELGI approach and demonstrated its effectiveness in a case study. However, statistical significance can be difficult to determine with such small datasets. This being said, even with small samples, we have demonstrated that there is a visible advantage to optimisation of the participant database for use in transfer learning techniques. We can see that an evolved database has 3 primary advantages:

1. *A higher classification accuracy*, regardless of quantity of training data. As seen in Figure 7.4, 62.5% of cases see eELGI performing better than ELGI and SLII, with the remaining still close to the optimal. In Figure 7.6b we observe, in the majority of cases, a marked improvement over the non-evolved ELGI.

2. *A reduction in variance* in performance across not only sessions, but participants as well. When comparing sessions in Figure 7.6a, and training set size in Figure 7.6b, the groupings of round accuracies are noticeably more dense. Figure 7.3, is perhaps the most dramatic demonstration of this. By including

all participants over all test sets, the error bars for both the SLII and ELGI are substantial, while the eELGI provides a modest difference.

3. *A means for protection against temporal drift*. Figure 7.6b demonstrates that the traditional BCI approach (SLII) is highly susceptible to the neural drift seen over time. While ELGI alleviates that to a degree, eELGI provides a much more linear, and slower degradation in predictive accuracy over the testing sessions.

As this chapter focused on a small dataset, with an equal number of able and disabled patients, further work should investigate the effects of optimising different base datasets. For example, further work should contain substantially more participants and, in more commonly observed situations, contain disproportionately more able-bodied participants. In terms of algorithms, while a simple Hill Climbing algorithm has provided some promising results, it would be prudent to apply more complex heuristics to the problem. A potentially promising direction would be utilisation of a genetic algorithm with an encoding that would allow oversampling of the more prototypical instances.

Part VIII

SUMMARY AND CONCLUSIONS

8

CHAPTER 8 - SUMMARY AND CONCLUSIONS

This chapter is arranged in the following manner: First we recap the motivation for BCI and known problems are summarised. The contributions of this thesis are then introduced, and explicitly stated. This is followed by how these contributions addressed the problems highlighted, and summarises their results. Finally, avenues for future work are discussed.

Brain Computer Interfaces currently provide life-altering benefits to users, but refinement will allow their application to a much wider variety of disabilities, and increase their practicality. BCI most commonly use electroencephalography for the detection of the neural signals used to communicate between mind and machine. This modality of signal detection is highly problematic:

- **Sources of information often overlap different channels** - to detect the activity of neurons outside the skull, large numbers must be active. An electrode will not only detect the electrical activity of neurons directly below it, but some from neighbouring populations. This reduces the spatial information available.

- **These channels are incredibly noisy** - the information of interest is generated within the brain, but electrodes will detect information from a range of different sources e.g. cardiac rhythms and eye movement.

- **Inconsistency of use** - electrodes are applied to the scalp using a standardised method (International 10-20 system), but the exact location, and the conductivity of the electrode will vary between users and systems. This further adds to the non-stationarity of the data.

These problems further compound the inherent issues in neural recordings for BCI applications:

- **Non-stationarity between sessions -** neural drift occurs when plasticity causes alterations in the brain's structure that results in patterns, that were previously observed, no longer being present. This creates the need for new training data to be recorded if a model is unable to generalise sufficiently.

- **Non-stationarity between participants -** neural differences between individuals are sufficiently large that a model trained on an individual is a very poor fit for future participants. Generalisable models require large numbers of participants, and even then, benefit greatly from refinement to the new user.

- **Small datasets -** recording sessions involve slow, tedious and repetitive tasks. This inevitably results in user frustration, altering the neural patterns observed and limiting the quantity of training instances possible.
- **Invalid instances -** some dataset instances are invalid due to user inattention to the tasks in the paradigm. If these outliers are used in the training of the predictive model, a poor fit may occur.

Classification problems involving EEG data are difficult, but it is possible to improve models through a range of different optimisation techniques. Search based techniques have been demonstrated to be particularly effective in selecting near-optimal subsets of features to better represent the neural activity of interest to the application, in a process called *Feature Selection*. In this thesis, we integrated *Linkage* and *Mutual Information* into the metaheuristic *Iterated Local Search*. We discovered that guidance of perturbation operators that use pair-wise linkage can restrict search. Perturbation operators that take into account Mutual Information however, decrease the cross validation error rate from training sets, and increases the predictive accuracy on unseen data. These experiments also revealed the perils of over fitting solutions on training data through the use of *cross validation error rates*. It was found that a metric developed in this thesis, *Intrasolution Linkage* was a better indicator of a solutions fitness in the latter iterations of search.

To further develop search within BCI, an algorithm was developed to optimise a multiple participant database to aid in *Transfer Learning*. Using data from previous participants is an effective method of increasing the quantity of training data available for new users. However, brain signals differ substantially between different persons, and therefore not all data obtained is useful. To ensure the best use of available data, we employed a search technique to transplant instances between datasets in an ensemble. This was performed as an adaption of an existing state-of-the-art Transfer Learning approach called *Ensemble Learning Generic Information (ELGI)*, which we termed *Evolved Ensemble Learning Generic Information (eELGI)*.

8.1 CONTRIBUTIONS

The contributions of this thesis are explicitly stated as:

1. **Application of Existing Techniques To a New Domain:**
    - **Iterated Local Search -** ILS has not been previously applied to any problem within Brain Computer Interfaces. Here, it was compared with *Hill Climbing*, *Sequential Forward Search*, *Genetic Algorithm*, *Memetic Algorithm*, *Mutual Information Feature Selection*, *minimum Redundancy Maximum Relevance*, and *LASSO*.
    - **Linkage Detection Algorithm -** Like ILS, LDA has not previously appeared within the Brain Computer Interface literature.
2. **New variations of Iterated Local Search:**
    - **MRMR-Iterated Local Search -** a variation of ILS in which the perturbation operator was guided by a Mutual Information measure was introduced and evaluated.
    - **Linkage-Iterated Local Search -** the Linkage Detection Algorithm was used to guide the perturbation operator of ILS in two new algorithms:
        - **Benign L-ILS -** benign perturbations involved retaining features which provided the most *benign intrasolution linkage* within the solution.
        - **Malign L-ILS -** malign perturbations involved discarding features which provided the most *malign intrasolution linkage* within the solution.
3. **New Metrics:**
    - **Intrasolution Linkage Measures -** a method in which the Linkage Detection Algorithm could be utilised to rank features within a solution was introduced. Its design was intended to take into account the already selected features, and their relationship with the class label.

4. **Created a New Technique for Transfer Learning:**
    - **eELGI -** *Evolved Ensemble Learning Generic Information (eELGI)* was created by optimisation of the participant database, in which *instance transfer* was performed using a Hill Climbing algorithm. The performance of the eELGI was then compared against the existing state-of-the-art technique: *Ensemble Learning Generic Information (ELGI)*. A further contribution was the application and evaluation of the ELGI approach to a dataset with fewer participants than previously seen in the literature.
5. **Classifier Comparisons -** *k-Nearest-Neighbours* is uncommon in the BCI literature. We have shown that it can be applied, successfully, to some datasets. In some cases, it produces comparable performance to the state-of-the-art *Support Vector Machine*.
6. **Insight to Overfitting -** further evidence was found to support that overfitting is a common problem in feature selection. We have also found that our new metric (*Intrasolution Linkage*) may be able to mitigate it.

8.2 GENERAL CONCLUSION

This thesis explored feature selection techniques to isolate **sources of information** that best predict the neurological patterns that relate to BCI tasks. This allowed **noise** to be discarded, helped compensate for **inconsistencies** between user sessions, and reduced computational load. The efficacy of dimensional reduction in BCI is evidenced in Chapters 5 and 6. In these, feature selection algorithms were used to find feature subsets which achieved lower cross-validation-error rates, and increased predictive accuracies on unseen data. We compared a number of different search algorithms, including *Iterated Local Search*, which had not been applied to BCI before.

The newly introduced algorithm MRMR-ILS was shown to perform better than ILS with an unguided perturbation operator over almost all datasets and classifiers. Furthermore, it also performed better than well established Filter techniques such as MIFS and mRMR, and the state-of-the-art embedded method, LASSO.

Chapters 5 and 6 identify the poor correlation between the training set's Cross Validation Error rate and the accuracies obtained on unseen data. It was found in Chapter 5 that, while this correlation decreases as the search progresses, the correlation between the accuracy on unseen data and the intra-solution linkage score actually increases. This trend is further seen in Chapter 6, in which the solutions produced by wrapper algorithms that included Mutual Information in their search produced higher CVE and accuracy correlations. This suggests that wrapper algorithms in this field should incorporate additional information measures to help mitigate the affect of over-fitting.

As previously stated in this Chapter, inconsistencies in the application of equipment can lead to poor session-to-session transfer, but the most prominent problem is **non-stationarity between sessions**. We can see that, in Chapter 7, the predictive accuracy of a BCI deteriorates depending on the length of time since the training data was acquired. A state-of-the-art approach known as ELGI was applied to utilise data from a range of users to mitigate this *neural drift*. Positive results were observed in that it substantially improved upon the single-user BCI, SLII. A problem with using additional users in this manner

is **non-stationarity between participants**. This is somewhat mitigated by the recombination technique seen in ELGI, but we sought to further optimise the existing database by transferring instances between datasets within an ensemble by using a local search. We found that applying this technique achieved three primary advantages:

1. Increased classification accuracy rates were found, even when the quantity of user specific data is restricted to the **smallest of datasets**.
2. Increased stability of BCI performance, with similar performances observed across time, and participants with varying neurological impairments.
3. A reduction in performance degradation due to neural drift.

8.2.1 *Potential Impacts of our Contributions*

It is important to emphasise that optimisation in this field has real world implications. From the results stated above, advances in deployed BCI would include:

- A reduction in the training data required, causing less distress to new users.
- Higher accuracy predictions, making deployment of BCIs more practical.
- Faster BCI response times, allowing faster communication with devices like the P300 speller.
- Increased periods of practical use without retraining, giving more independence to the individual and reducing financial costs.

8.3 SUMMARY

In summary, this thesis has contributed a selection of new search techniques, tested on a series of state-of-the-art benchmark BCI datasets. These algorithms sought to, and achieved, their intended purposes of: reducing computational demand in optimisation, reducing user-specific training data, increasing predictive accuracy of feature subsets, increasing the robustness of BCI systems

in terms of user impairment, and mitigating the effects of neural drift. Furthermore, a surprising discovery was finding evidence to support the adoption of new metrics for prevention of over fitting to training data.

The contribution of this thesis represents an advancement in BCI systems. Ultimately, this offers potential application to a much wider variety of disabilities, while also increasing their practicality — providing life altering benefits to users.

8.4 FUTURE WORK

Experiments in this thesis have offered potential improvements over existing techniques. However, there are still avenues in which they can be further explored.

eELGI Variations

In our experiments, we chose to use local search to perform instance transfer between different training datasets for an ensemble. The appropriateness of an instance transplantation was assessed by using the altered dataset to train a model, which was then evaluated by making predictions on data obtained from a population of participants. This was chosen as it allows the optimisation process to occur prior to the introduction of new users. However, it may be prominent to attempt to evaluate the instance transfer on training data provided by the new participant, customising the model to their neural patterns, rather than that of the general population.

Additional work on this algorithm should include the incorporation of a more advanced search technique such as a genetic algorithm, addition of *instance deletion* and *instance duplication* operators, assessment of alternative metrics to assess solution fitness such as Pearson's Correlation Coefficient and Mutual Information, and experimention using datasets containing increased numbers of participants.

Improved Fitness Functions

An issue identified in this thesis is the reliability of the fitness function. As

shown in Chapters 5 and 6, the correlation between predicted solution fitness achieved from cross validation of training sets and the predictive accuracy on unseen data quickly declines when error rates are minimal. With low correlations, it calls into question whether the fitness function is valid in this phase of the optimisation algorithm. Section 5 shows that the correlation between the solutions predictive accuracy on unseen data, and the new measurements introduced by this thesis are actually stronger than the traditional k-fold C.V approach. Expanding on this, we would like to investigate the relationships between these new measurements and the fitness of the solutions: Specifically, is there a point in search in which the fitness function could be replaced by these alternative metrics? Is it possible, using Genetic Programming, to create a function which incorporates Linkage, Mutual Information, and Cross Validation?

Artificial Data Ensembles

As wrapper methods select features by subdividing the available data, and training the classifier on those subdivisions, we may find that those subsets may contain a noisy and insufficient subset of instances to create a good model. To overcome this, we have shown that increases in generality can be achieved by taking data from other participants. We then demonstrate that this additional data can be optimised by instance transfer. This can be taken further: future work should take into account the possibility of generating artificial EEG data with the express purpose of training nodes within an ensemble. It may prove possible to generate data for this purpose that achieves two goals: Accentuation of the participants detectable patterns, and increasing the generalisability of the resulting models.

Additional Datasets

Finally, future work should include evaluation of our algorithms suggested in Chapters 5, 6, and 7 on additional datasets. It is evident from the literature review that an algorithm's performance is highly varied depending on not just the problem, but the dataset itself. Our conclusions can only be made based

on the limitations of the datasets explored.

Building upon Estimation of Distribution Algorithms

A prominent next step is development of our Linkage-aware heuristics through comparison to similar techniques known as Estimation of Distribution Algorithms (*EDA*). This family of algorithms seek to extend upon Genetic Algorithms by replacing the population with a probabilistic sampling technique [22]. Univariate EDAs, for example, the Compact Genetic Algorithm (*cGA*) [67], propose a vector of probabilistic values, using it to create and evaluate solutions. In response to the evaluated fitness, the probabilities are then updated. A pitfall of this technique is that it does not allow for interactions between variables, something multivariate EDAs have sought to address through *linkage learning*.

A linkage-aware EDA known as extended compact Genetic Algorithm (*ecGA*) explicitly evaluates the interaction between variables and its impact on the resulting fitness of the solution, rather than the implicit linkage found in simple GAs [65]. This is achieved by calculating the product of the marginal distributions of a partition of the features. This differs from our approaches in that we assign a 'fitness' to pairs of features, rather than a probability for their selection. In our approach, a pair that have a higher *linkage score* will be selected over a pair with a lower score. This over reliance on exploitation of the classifiers error rate may suggest why our linkage-aware algorithms failed to explore the space adequately. A future work could involve a probabilistic sampling based on our linkage score metric. We additionally note that, although linkage has been shown to improve heuristic search [116], in some cases of feature selection, it was found to make no significant impact [33]. This suggests that we must also determine that linkage information is appropriate for use in this field.

BIBLIOGRAPHY

[1] Jason Adair, Alexander Brownlee, and Gabriela Ochoa. Evolutionary Algorithms with Linkage Information for Feature Selection in Brain Computer Interfaces. In *Advances in Intelligent Systems and Computing*, volume 513, pages 287–307. 2017. ISBN 97833194656l6. doi: 10.1007/978-3-319-46562-3_19.

[2] Jason Adair, Alexander Brownlee, Fabio Daolio, and Gabriela Ochoa. Evolving training sets for improved transfer learning in brain computer interfaces. In *Machine Learning, Optimization, and Big Data*, pages 186–197. Springer International Publishing, 2018. ISBN 978-3-319-72926-8. doi: 10.1007/978-3-319-72926-8_16.

[3] Jason Adair, Alexander E. I. Brownlee, and Gabriela Ochoa. Mutual information iterated local search: A wrapper-filter hybrid for feature selection in brain computer interfaces. In *Applications of Evolutionary Computation*, pages 63–77, Cham, 2018. Springer International Publishing. ISBN 978-3-319-77538-8. doi: 10.1007/978-3-319-77538-8_5.

[4] Tarik Al-ani, Dalila Trad, and Dalila Tr. *Signal processing and classification approaches for brain-computer interface*. Number January. 2010. ISBN 9789537619589. doi: 10.5772/7032.

[5] Amjed S Al-Fahoum and Ausilah A Al-Fraihat. Methods of EEG signal features extraction using linear analysis in frequency and time-frequency domains. *ISRN neuroscience*, 2014:730218, jan 2014. ISSN 2314-4661. doi: 10.1155/2014/730218.

[6] Ahmed Fouad Ali and Aboul-ella Hassanien. *Applications of Intelligent Optimization in Biology and Medicine*, volume 96 of *Intelligent Systems Reference Library*. Springer International Publishing, Cham, 2016. ISBN 978-3-319-21211-1. doi: 10.1007/978-3-319-21212-8.

[7] Syed Imran Ali and Waseem Shahzad. A feature subset selection method based on symmetric uncertainty and Ant Colony Optimization. In *2012 International Conference on Emerging Technologies*, pages 1–6. IEEE, oct 2012. ISBN 978-1-4673-4451-7. doi: 10.1109/ICET.2012.6375420.

[8] Turky Alotaiby, Fathi E Abd El-Samie, Saleh A Alshebeili, and Ishtiaq Ahmad. A review of channel selection algorithms for EEG signal processing. *EURASIP Journal on Advances in Signal Processing*, 2015(1):66, 2015. ISSN 1687-6180. doi: 10.1186/s13634-015-0251-9.

[9] and and and. The structure of the nervous system of the nematode caenorhabditis elegans. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 314(1165):1–340, 1986. ISSN 0080-4622. doi: 10.1098/rstb.1986.0056.

[10] Kai Keng Ang, Zheng Yang Chin, Chuanchu Wang, Cuntai Guan, and Haihong Zhang. Filter bank common spatial pattern algorithm on BCI competition IV datasets 2a and 2b. *Frontiers in Neuroscience*, 6(MAR):1–9, 2012. ISSN 16624548. doi: 10.3389/fnins.2012.00039.

[11] K. Aswinseshadri and V. Thulasi Bai. Evaluation of feature selection in Brain Computer Interface. *Proceeding of IEEE - 2nd International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics, IEEE - AEEICB 2016*, pages 93–97, 2016. doi: 10.1109/AEEICB.2016.7538404.

[12] John Atkinson and Daniel Campos. Improving BCI-based emotion recognition by combining EEG feature selection and kernel classifiers. *Expert Systems with Applications*, 47(November 2015):35–41, apr 2016. ISSN 09574174. doi: 10.1016/j.eswa.2015.10.049.

[13] Ali Bakhshi. A Comparison among Classification Accuracy of Neural Network , FLDA and BLDA in P300-based BCI System. 46(19):11–15, 2012.

[14] Ali Bashashati, Mehrdad Fatourechi, Rabab K Ward, and Gary E Birch. A survey of signal processing algorithms in brain-computer interfaces

based on electrical brain signals. *Journal of neural engineering*, 4:R32–R57, 2007. ISSN 1741-2560. doi: 10.1088/1741-2560/4/2/R03.

[15] Una Benlic and Jin-Kao Hao. A study of adaptive perturbation strategy for iterated local search. In Martin Middendorf and Christian Blum, editors, *Evolutionary Computation in Combinatorial Optimization*, pages 61–72, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. ISBN 978-3-642-37198-1.

[16] Tushar Kanti Bera. *Noninvasive electromagnetic methods for brain monitoring: A technical review*, volume 74. 2015. ISBN 9783642017988. doi: 10.1007/978-3-319-10978-7_3.

[17] Javad Birjandtalab, Maziyar Baran Pouyan, and Mehrdad Nourani. Unsupervised eeg analysis for automated epileptic seizure detection. *Proc.SPIE*, 10011:10011 – 10011 – 5, 2016. doi: 10.1117/12.2243622.

[18] Benjamin Blankertz, Ryota Tomioka, Steven Lemm, Motoaki Kawanabe, and Klaus Robert Müller. Optimizing spatial filters for robust EEG single-trial analysis. *IEEE Signal Processing Magazine*, 25(1):41–56, 2008. ISSN 10535888. doi: 10.1109/MSP.2008.4408441.

[19] Robert Bogue. Exoskeletons and robotic prosthetics: a review of recent developments. *Industrial Robot: An International Journal*, 36(5):421–427, 2009. ISSN 0143-991X. doi: 10.1108/01439910910980141.

[20] Henry Brighton and Chris Mellish. Advances in instance selection for instance-based learning algorithms. *Data Mining and Knowledge Discovery*, 6(2):153–172, 2002. ISSN 13845810. doi: 10.1023/A:1014043630878.

[21] A. E. I. Brownlee, M. Pelikan, J. A. W. McCall, and A. Petrovski. An application of a multivariate estimation of distribution algorithm to cancer chemotherapy. In *Proc. GECCO*, pages 463–464, Atlanta, GA, USA, 2008. ACM Press. ISBN 978-1-60558-130-9. doi: http://doi.acm.org/10.1145/1389095.1389179.

[22] A E I Brownlee, M Pelikan, John McCall, and A Petrovski. An application of a multivariate estimation of distribution algorithm to cancer chemo-

therapy. *Proceedings of the Genetic and Evolutionary Computation Conference GECCO-2008*, pages 463–464, 2008. doi: 10.1145/1389095.1389179.

[23] A. E. I. Brownlee, J. A. W. McCall, S. K. Shakya, and Q. Zhang. Structure Learning and Optimisation in a Markov-network based Estimation of Distribution Algorithm. In *Proc. IEEE CEC*, pages 447–454, Trondheim, Norway, 2009. IEEE Press. doi: 10.1007/978-3-642-12834-9_3.

[24] A. E. I. Brownlee, J. A. W. McCall, and L. A. Christie. Structural coherence of problem and algorithm: An analysis for EDAs on all 2-bit and 3-bit problems. In *Proc. IEEE CEC*, pages 2066–2073, Sendai, Japan, 2015. IEEE Press. doi: 10.1109/CEC.2015.7257139.

[25] A.E.I. Brownlee, J.A.W. McCall, and Q. Zhang. Fitness modeling with Markov networks. *IEEE T. Evolut. Comput.*, 17(6):862–879, 2013. ISSN 1089-778X. doi: 10.1109/TEVC.2013.2281538.

[26] Alexander E. I. Brownlee, John A. W. McCall, and Martin Pelikan. Influence of selection on structure learning in Markov network EDAs: An empirical study. In *Proc. GECCO*, pages 249–256. ACM Press, 2012. doi: 10.1145/2330163.2330200.

[27] Alexander E. I. Brownlee, Olivier Regnier-Coudert, John A. W. McCall, Stewart Massie, and Stefan Stulajter. An application of a GA with Markov network surrogate to feature selection. *Int. J. Syst. Sci.*, 44(11): 2039–2056, 2013. doi: 10.1080/00207721.2012.684449.

[28] E K Burke, J P Newall, and R F Weare. A memetic algorithm for university exam timetabling. *Practice and Theory of Automated Timetabling SE - 15*, 1153:241–250, 1996. ISSN 16113349. doi: 10.1007/3-540-61794-9{_}63.

[29] Samuel P Burns, Dajun Xing, and Robert M Shapley. Comparisons of the Dynamics of Local Field Potential and Multiunit Activity Signals in Macaque Visual Cortex. *Journal of Neuroscience*, 30(41):13739–13749, oct 2010. ISSN 0270-6474. doi: 10.1523/JNEUROSCI.0743-10.2010.

[30] György Buzsáki, Costas a. Anastassiou, and Christof Koch. The origin of extracellular fields and currents - EEG, ECoG, LFP and spikes. *Nature*

Reviews Neuroscience, 13(June):407–420, 2012. ISSN 1471-003X. doi: 10.1038/nrn3241.

[31] Alvaro Fuentes Cabrera, Dario Farina, and Kim Dremstrup. Comparison of feature selection and classification methods for a brain-computer interface driven by non-motor imagery. *Medical and Biological Engineering and Computing*, 48(2):123–132, 2010. ISSN 01400118. doi: 10.1007/s11517-009-0569-2.

[32] Jessica Cantillo-Negrete, Josefina Gutierrez-Martinez, Ruben Carino-Escobar, Paul Carrillo-Mora, and David Elias-Vinas. An approach to improve the performance of subject-independent BCIs-based motor imagery allocating subjects by gender. *Biomed eng*, 13(1), 2014. ISSN 1475-925X. doi: 10.1186/1475-925X-13-158.

[33] E Cantú-Paz. Feature Subset Selection by Estimation of Distribution Algorithms. *Proceedings of the 6th Genetic and Evolutionary Computation Conference, GECCO 2004*, pages 303–310, 2004. doi: 10.1007/978-1-4615-1539-5_13.

[34] Sounak Chakraborty, Malay Ghosh, and Bani K. Mallick. Bayesian nonlinear regression for large p small n problems. *Journal of Multivariate Analysis*, 108:28 – 40, 2012. ISSN 0047-259X. doi: https://doi.org/10.1016/j.jmva.2012.01.015.

[35] Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. *Computers and Electrical Engineering*, 40(1):16–28, 2014. ISSN 00457906. doi: 10.1016/j.compeleceng.2013.11.024.

[36] Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. *Comput. Electr. Eng.*, 40(1):16–28, January 2014. ISSN 0045-7906. doi: 10.1016/j.compeleceng.2013.11.024.

[37] Chao. Long-term asynchronous decoding of arm motion using electrocorticographic signals in monkey. *Frontiers in Neuroengineering*, 3(March), 2010. ISSN 16626443. doi: 10.3389/fneng.2010.00003.

[38] Benhui Chen and Jinglu Hu. *Exploitation of Linkage Learning in Evolutionary Algorithms*, chapter Protein Structure Prediction Based on HP Model Using an Improved Hybrid EDA, pages 193–214. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010. ISBN 978-3-642-12834-9. doi: 10.1007/978-3-642-12834-9_9.

[39] Xue-wen Chen and Jong Cheol Jeong. Enhanced recursive feature elimination. In *Sixth International Conference on Machine Learning and Applications (ICMLA 2007)*, number January 2008, pages 429–435. IEEE, dec 2007. ISBN 978-0-7695-3069-7. doi: 10.1109/ICMLA.2007.35.

[40] Francisco Chicano, Darrell Whitley, and Andrew M. Sutton. Efficient identification of improving moves in a ball for pseudo-boolean problems. In *Proceedings of the 2014 Conference on Genetic and Evolutionary Computation*, GECCO '14, pages 437–444, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2662-9. doi: 10.1145/2576768.2598304.

[41] Pádraig Cunningham. Dimension Reduction. *Ucd-Csi*, 1(7):15–35, 2007. ISSN 1471-2105. doi: 10.1007/978-0-387-69942-4_2.

[42] Jose del R. Milan and Jose Carmena. Invasive or Noninvasive: Understanding Brain-Machine Interface Technology [Conversations in BME. *IEEE Engineering in Medicine and Biology Magazine*, 29(1):16–22, jan 2010. ISSN 0739-5175. doi: 10.1109/MEMB.2009.935475.

[43] N. S. Dias, L. R. Jacinto, P. M. Mendes, and J. H. Correia. Feature down-selection in brain-computer interfaces dimensionality reduction and discrimination power. *2009 4th International IEEE/EMBS Conference on Neural Engineering, NER '09*, pages 323–326, 2009. doi: 10.1109/NER.2009.5109298.

[44] John P Donoghue. Bridging the Brain to the World: A Perspective on Neural Interface Systems. *Neuron*, 60(3):511–521, nov 2008. ISSN 08966273. doi: 10.1016/j.neuron.2008.10.037.

[45] Béatrice Duval, Jin-Kao Hao, and Jose Crispin Hernandez Hernandez. A memetic algorithm for gene selection and molecular classification of

cancer. *Proceedings of the 11th Annual conference on Genetic and evolutionary computation - GECCO '09*, (May 2014):201, 2009. doi: 10.1145/1569901.1569930.

[46] Anders Eklund, Mats Andersson, Henrik Ohlsson, Anders Ynnerman, and Hans Knutsson. A Brain Computer Interface for Communication Using Real-Time fMRI. In *2010 20th International Conference on Pattern Recognition*, pages 3665–3669. IEEE, aug 2010. ISBN 978-1-4244-7542-1. doi: 10.1109/ICPR.2010.894.

[47] L. A. Farwell and E. Donchin. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalography and Clinical Neurophysiology*, 70(6):510–523, 1988. ISSN 00134694. doi: 10.1016/0013-4694(88)90149-6.

[48] Reza Fazel-Rezai and Waqas Ahm. P300-based Brain-Computer Interface Paradigm Design. In *Recent Advances in Brain-Computer Interface Systems*. InTech, feb 2011. ISBN 9789533070940. doi: 10.5772/14858.

[49] Sombut Foithong, Ouen Pinngern, and Boonwat Attachoo. Feature subset selection wrapper based on mutual information and rough sets. *Expert Systems with Applications*, 39(1):574–584, jan 2012. ISSN 09574174. doi: 10.1016/j.eswa.2011.07.048.

[50] M. J. Fu, J. J. Daly, and M. C. Cavusoglu. Assessment of eeg event-related desynchronization in stroke survivors performing shoulder-elbow movements. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pages 3158–3164, May 2006. doi: 10.1109/ROBOT.2006.1642182.

[51] Agnes P. Funk and Charles M. Epstein. Natural rhythm: evidence for occult 40Hz gamma oscillation in resting motor cortex. *Neuroscience Letters*, 371(2-3):181–184, nov 2004. ISSN 03043940. doi: 10.1016/j.neulet.2004.08.066.

[52] Suraj R Gaikwad and Shruti S Kshirsagar. A Review : Analysis of EEG Signal based Brain- Computer Interface. 3(6):2447–2451, 2014.

[53] John Q. Gan, Bashar Awwad Shiekh Hasan, and Chun Sing Louis Tsui. A filter-dominating hybrid sequential forward floating search method for feature subset selection in high-dimensional space. *International Journal of Machine Learning and Cybernetics*, 5(3):413–423, 2014. ISSN 1868808X. doi: 10.1007/s13042-012-0139-z.

[54] Girisha Garg, Vijander Singh, J. R. P. Gupta, and A. P. Mittal. Wrapper based wavelet feature optimization for eeg signals. *Biomedical Engineering Letters*, 2(1):24–37, Mar 2012. ISSN 2093-985X. doi: 10.1007/s13534-012-0044-0.

[55] Poonam Garg. A Comparison between Memetic algorithm and Genetic algorithm for the cryptanalysis of Simplified Data Encryption Standard algorithm. *International Journal of Network Security & Its Applications*, 1(1): 34–42, 2009.

[56] Holger Gevensleben, BjÃ¶rn Albrecht, Henry LÃÂ¼tcke, Tibor Auer, Wan Ilma Dewiputri, Renate Schweizer, Gunther Moll, Hartmut Heinrich, and Aribert Rothenberger. Neurofeedback of slow cortical potentials: neural mechanisms and feasibility of a placebo-controlled design in healthy adults. *Frontiers in Human Neuroscience*, 8(December):1–13, dec 2014. ISSN 1662-5161. doi: 10.3389/fnhum.2014.00990.

[57] Mick Grierson and Chris Kiefer. Better brain interfacing for the masses. In *Proceedings of the 2011 annual conference extended abstracts on Human factors in computing systems - CHI EA '11*, page 1681, New York, New York, USA, 2011. ACM Press. ISBN 9781450302685. doi: 10.1145/1979742.1979828.

[58] Carlos Guerrero-Mosquera, Michel Verleysen, and Angel Navia Vazquez. EEG feature selection using mutual information and support vector machine: A comparative analysis. In *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, volume 2010, pages 4946–4949. IEEE, aug 2010. ISBN 978-1-4244-4123-5. doi: 10.1109/IEMBS.2010.5627239.

[59] Christoph Guger, Shahab Daban, Eric Sellers, Clemens Holzner, Gunther Krausz, Roberta Carabalona, Furio Gramatica, and Guenter Edlinger. How many people are able to control a P300-based brain-computer interface (BCI)? *Neuroscience Letters*, 462(1):94–98, sep 2009. ISSN 03043940. doi: 10.1016/j.neulet.2009.06.045.

[60] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *J. Mach. Learn. Res.*, 3:1157–1182, March 2003. ISSN 1532-4435.

[61] Haihong Zhang, Kai Keng Ang, Cuntai Guan, and Chuanchu Wang. Spatio-spectral feature selection based on robust mutual information estimate for brain computer interfaces. In *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, volume 2009, pages 4978–4981. IEEE, sep 2009. ISBN 9781424432967. doi: 10.1109/IEMBS.2009.5334093.

[62] Mark A. Hall. Correlation-based feature selection for discrete and numeric class machine learning. In *Proceedings of the Seventeenth International Conference on Machine Learning*, ICML '00, pages 359–366, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc. ISBN 1-55860-707-2.

[63] Brahim Hamadicharef, Haihong Zhang, Cuntai Guan, Chuanchu Wang, Kok Soon Phua, Keng Peng Tee, and Kai Keng Ang. Learning EEG-based spectral-spatial patterns for attention level measurement. In *2009 IEEE International Symposium on Circuits and Systems*, pages 1465–1468. IEEE, may 2009. ISBN 978-1-4244-3827-3. doi: 10.1109/ISCAS.2009.5118043.

[64] Julie Hamon, Clarisse Dhaenens, Gaël Even, and Julien Jacques. Feature selection in high dimensional regression problems for genomic. *Tenth International Meeting on Computational Intelligence Methods for Bioinformatics and Biostatistics*, pages 1–9, 2013.

[65] Georges Harik, Fernando Lobo, and Kumara Sastry. Linkage learning via probabilistic modeling in the extended compact genetic algorithm (ecga). volume 33, pages 39–61, 01 2007. doi: 10.1007/978-3-540-34954-9_3.

[66] Georges R Harik and David E Goldberg. Learning linkage. In *FOGA*, volume 4, pages 247–262, 1996.

[67] G.R. Harik, F.G. Lobo, and D.E. Goldberg. The compact genetic algorithm. In *1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360)*, volume 3, pages 523–528. IEEE, 1999. ISBN 0-7803-4869-9. doi: 10.1109/ICEC.1998.700083.

[68] Bashar Awwad Shiekh Hasan, John Q. Gan, and Qingfu Zhang. Multi-objective evolutionary methods for channel selection in Brain-Computer Interfaces: Some preliminary experimental results. In *IEEE Congress on Evolutionary Computation*, pages 1–6. IEEE, jul 2010. ISBN 978-1-4244-6909-3. doi: 10.1109/CEC.2010.5586411.

[69] Mark Hauschild and Martin Pelikan. An introduction and survey of estimation of distribution algorithms. *Swarm Evol. Comput.*, 1(3):111 – 128, 2011. ISSN 2210-6502. doi: 10.1016/j.swevo.2011.08.003.

[70] Mark Hauschild, Martin Pelikan, Claudio F. Lima, and Kumara Sastry. Analyzing probabilistic models in hierarchical BOA on traps and spin glasses. In *Proc. GECCO*, pages 523–530. ACM Press, 2007. ISBN 978-1-59593-697-4.

[71] Robert B. Heckendorn and Alden H. Wright. Efficient linkage discovery by limited probing. *Evolutionary Computation*, 12(4):517–545, 2004. doi: 10.1162/1063656043138914. URL `https://doi.org/10.1162/1063656043138914`.

[72] Suzana Herculano-Houzel. The human brain in numbers: a linearly scaled-up primate brain. *Frontiers in Human Neuroscience*, 3:31, 2009. ISSN 1662-5161. doi: 10.3389/neuro.09.031.2009.

[73] Pawel Herman, Girijesh Prasad, Senior Member, Thomas Martin Mcginnity, and Damien Coyle. Comparative Analysis of Spectral Approaches to Feature Extraction for EEG-Based Motor Imagery Classification. 16 (4):317–326, 2008. doi: 10.1109/TNSRE.2008.926694.

[74] Uwe Herwig, Peyman Satrapi, and Carlos Schönfeldt-Lecuona. Using the international 10-20 eeg system for positioning of transcranial magnetic stimulation. *Brain Topography*, 16(2):95–99, Dec 2003. ISSN 1573-6792. doi: 10.1023/B:BRAT.0000006333.93597.9d.

[75] L.R. Hochberg and J.P. Donoghue. Sensors for brain-computer interfaces. *IEEE Engineering in Medicine and Biology Magazine*, 25(5):32–38, sep 2006. ISSN 0739-5175. doi: 10.1109/MEMB.2006.1705745.

[76] Ulrich Hoffmann, Jean-Marc Vesin, Touradj Ebrahimi, and Karin Diserens. An efficient P300-based brain-computer interface for disabled subjects. *Journal of Neuroscience Methods*, 167(1):115–125, jan 2008. ISSN 01650270. doi: 10.1016/j.jneumeth.2007.03.005.

[77] John H. Holland. *Adaptation in natural and artificial systems*. University of Michigan Press, Ann Arbor, 1975. ISBN 0472084607. by John H. Holland.; Includes index.; Bibliography: p. 175-177.

[78] Torsten Hothorn, Frank Bretz, and Peter Westfall. Simultaneous inference in general parametric models. *Biometrical journal. Biometrische Zeitschrift*, 50:346–63, 06 2008. doi: 10.1002/bimj.200810425.

[79] A James Hudspeth, Thomas M Jessell, Eric R Kandel, James Harris Schwartz, and Steven A Siegelbaum. *Principles of neural science*. McGraw-Hill, Health Professions Division, 2013. doi: 10.1002/mus.880050615.

[80] Vinay Jayaram, Morteza Alamgir, Yasemin Altun, Bernhard Scholkopf, and Moritz Grosse-Wentrup. Transfer Learning in Brain-Computer Interfaces. *IEEE Comp Intell Mag*, 11(1):20–31, 2016. ISSN 1556603X. doi: 10.1109/MCI.2015.2501545.

[81] Jing Jin, Xingyu Wang, and Jianhua Zhang. Optimal selection of eeg electrodes via dpso algorithm. In *2008 7th World Congress on Intelligent Control and Automation*, pages 5095–5099, June 2008. doi: 10.1109/WCICA.2008.4593756.

[82] A K. Jain and B Chandrasekaran. Dimensionality and sample size considerations in pattern recognition practice. *Handbook of Statistics*, 2: 835–855, 12 1982. doi: 10.1016/S0169-7161(82)02042-2.

[83] L. Kallel, B. Naudts, and C. R. Reeves. Properties of fitness functions and search landscapes. In Leila Kallel, Bart Naudts, and Alex Rogers, editors, *Theoretical Aspects of Evolutionary Computing*, pages 175–206. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001. ISBN 978-3-662-04448-3. doi: 10.1007/978-3-662-04448-3_8. URL https://doi.org/10.1007/978-3-662-04448-3_8.

[84] Ivo Käthner, Carolin A. Ruf, Emanuele Pasqualotto, Christoph Braun, Niels Birbaumer, and Sebastian Halder. A portable auditory P300 brain computer interface with directional cues. *Clinical Neurophysiology*, 124 (2):327–338, feb 2013. ISSN 13882457. doi: 10.1016/j.clinph.2012.08.006.

[85] Rami Khushaba, Akram Alsukker, Ahmed Al-Ani, and Adel Al-Jumaily. Enhanced feature selection algorithm using ant colony optimization and fuzzy memberships. *Proceedings of the 6th IASTED International Conference on Biomedical Engineering, BioMED 2008*, pages 34–39, 02 2008.

[86] Rami N. Khushaba, Ahmed Al-Ani, Akram AlSukker, and Adel Al-Jumaily. A combined ant colony and differential evolution feature selection algorithm. In Marco Dorigo, Mauro Birattari, Christian Blum, Maurice Clerc, Thomas Stützle, and Alan F. T. Winfield, editors, *Ant Colony Optimization and Swarm Intelligence*, pages 1–12, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg. ISBN 978-3-540-87527-7.

[87] D Kimura. Acquisition of a motor skill after left hemisphere damage. *Brain a journal of neurology*, 100(3):527–542, September 1977. ISSN 0006-8950. doi: 10.1093/brain/100.3.527.

[88] P.-J Kindermans, H Verschore, David Verstraeten, and B Schrauwen. A p300 bci for the masses: Prior information enables instant unsupervised spelling. *Adv. Neural Inform. Process. Syst.*, 25:719–727, 01 2012.

[89] Kevin J. Otto Kip A. Ludwig Daryl R. Kipke. *Acquiring Brain Signals from within the Brain*, volume 16. 2010. ISBN 9780199600458. doi: 10.1093/acprof.

[90] Ron Kohavi. A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. *Appears in the International Joint Conference on Articial Intelligence (IJCAI)*, 5:1–7, 1995. ISSN 10450823. doi: 10.1067/mod.2000.109031.

[91] Irena Koprinska. Feature selection for brain-computer interfaces. In Thanaruk Theeramunkong, Cholwich Nattee, Paulo J. L. Adeodato, Nitesh Chawla, Peter Christen, Philippe Lenca, Josiah Poon, and Graham Williams, editors, *New Frontiers in Applied Data Mining*, pages 106–117, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg. ISBN 978-3-642-14640-4.

[92] Pavel Krizek. *Feature selection: stability, algorithms, and evaluation*. PhD thesis, 2008.

[93] Principe Krusienski, McFarland. *BCI Signal Processing: Feature Extraction*. Number January. 2012. doi: 10.1093/acprof:oso/9780195388855.003.0007.

[94] Tian Lan, Deniz Erdogmus, Andre Adami, Misha Pavel, and Santosh Mathan. Salient EEG channel selection in brain computer interfaces by mutual information maximization. *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, 7: 7064–7, 2005. ISSN 1557-170X. doi: 10.1109/IEMBS.2005.1616133.

[95] Mikhail a Lebedev and Miguel a L Nicolelis. Brain-machine interfaces: past, present and future. *Trends in neurosciences*, 29(9):536–46, sep 2006. ISSN 0166-2236. doi: 10.1016/j.tins.2006.07.004.

[96] Mikhail a. Lebedev and Miguel a L Nicolelis. Brain-machine interfaces: past, present and future. *Trends in Neurosciences*, 29(9):536–546, 2006. ISSN 01662236. doi: 10.1016/j.tins.2006.07.004.

[97] Eun-Kyung Lee and Dianne Cook. A projection pursuit index for large p small n data. *Statistics and Computing*, 20(3):381–392, jul 2010. ISSN 0960-3174. doi: 10.1007/s11222-009-9131-1.

[98] Jaesung Lee and Dae-Won Kim. Memetic feature selection algorithm for multi-label classification. *Information Sciences*, 293:80–96, 2015. ISSN 00200255. doi: 10.1016/j.ins.2014.09.020.

[99] Eric C Leuthardt, Gerwin Schalk, D Ph, Jarod Roland, and Daniel W Moran. Evolution of brain-computer interfaces: going beyond classic motor physiology. 27(1):1–21, 2010. doi: 10.3171/2009.4.FOCUS0979. Evolution.

[100] Claudio F. Lima, Fernando G. Lobo, Martin Pelikan, and David E. Goldberg. Model accuracy in the bayesian optimization algorithm. *Soft Computing*, 15(7):1351–1371, Jul 2011. ISSN 1433-7479. doi: 10.1007/s00500-010-0675-y. URL `https://doi.org/10.1007/s00500-010-0675-y`.

[101] Huan Liu and Rudy Setiono. A Probabilistic Approach to Feature Selection - A Filter Solution. *Proceedings of International Conference on Machine Learning*, pages 319–327, 1996.

[102] Joseph J Locascio and Alireza Atri. An overview of longitudinal data analysis methods for neurological research. *Dement Geriatr Cogn Dis Extra*, 1(1):330–57, 2011. ISSN 1664-5464. doi: 10.1159/000330228.

[103] Fabien Lotte. Signal Processing Approaches to Minimize or Suppress Calibration Time in Oscillatory Activity-Based Brain-Computer Interfaces. *Proceedings of the IEEE*, 103(6):871–890, jun 2015. ISSN 0018-9219. doi: 10.1109/JPROC.2015.2404941.

[104] Fabien Lotte, Marco Congedo, Anatole LÃ©cuyer, Lamarche Fabrice, and Bruno Arnaldi. A review of classification algorithms for eeg-based brain-computer interfaces. *Journal of Neural Engineering*, 4, 07 2007. doi: 10.1088/1741-2560/4/2/R01.

[105] Fabien Lotte, Cuntai Guan, and Kai Keng Ang. Comparison of designs towards a subject-independent brain-computer interface based on motor imagery. In *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, number January 2016, pages 4543–4546. IEEE, sep 2009. doi: 10.1109/IEMBS.2009.5334126.

[106] Fabien Lotte, Electroencephalographic Signal, and Classification Tech. Study of Electroencephalographic Signal Processing and Classification Techniques towards the use of Brain-Computer Interfaces in Virtual Reality Applications Thèse l ' Institut National des Sciences Appliquées de Rennes Study of Electroencephalographic Si. 2009.

[107] John Loughrey and Pádraig Cunningham. Overfitting in Wrapper-Based Feature Subset Selection: The Harder You Try the Worse it Gets. *Research and Development in Intelligent Systems XXI SE - 3*, pages 33–43, 2005. doi: 10.1007/1-84628-102-4_3.

[108] Helena R. Lourenco, Olivier C. Martin, and Thomas Stutzle. *Iterated Local Search: Framework and Applications*, pages 363–397. Springer US, Boston, MA, 2010. ISBN 978-1-4419-1665-5. doi: 10.1007/978-1-4419-1665-5_12.

[109] J. A. Lozano, P. Larrañaga, I. Inza, and E. Bengoetxea. *Towards a New Evolutionary Computation: Advances on Estimation of Distribution Algorithms (Studies in Fuzziness and Soft Computing)*. Springer-Verlag, 2006. ISBN 3540290060.

[110] M. Lozano and C. García-Martínez. Hybrid metaheuristics with evolutionary algorithms specializing in intensification and diversification: Overview and progress report. *Computers & Operations Research*, 37(3): 481–497, mar 2010. ISSN 03050548. doi: 10.1016/j.cor.2009.02.010.

[111] Z. Mahmoodin, W. Mansor, Khuan Y Lee, and N. B. Mohamad. An analysis of EEG signal power spectrum density generated during writing in children with dyslexia. In *2015 IEEE 11th International Colloquium on Signal Processing & Its Applications (CSPA)*, pages 156–160. IEEE, mar 2015. ISBN 978-1-4799-8249-3. doi: 10.1109/CSPA.2015.7225637.

[112] Andrzej Majkowski, Marcin Kolodziej, and Remigiusz J. Rak. Implementation of automatic feature selection methods for BCI realization. In *2012 IEEE International Instrumentation and Measurement Technology Conference Proceedings*, pages 1286–1289. IEEE, may 2012. ISBN 978-1-4577-1772-7. doi: 10.1109/I2MTC.2012.6229340.

[113] Mrinal Kr Mandal and Amir Asif. *Continuous and discrete time signals and systems*. 2007. ISBN 0521854555 (hbk.)\r9780521854559 (hbk.).

[114] Rubén Martín-Clemente, Javier Olias, Deepa Thiyam, Andrzej Cichocki, and Sergio Cruces. Information Theoretic Approaches for Motor-Imagery BCI Systems: Review and Experimental Comparison. *Entropy*, 20(2):7, jan 2018. ISSN 1099-4300. doi: 10.3390/e20010007.

[115] Ankita Mazumder, Poulami Ghosh, Anwesha Khasnobish, Saugat Bhattacharyya, and D. N. Tibarewala. Selection of relevant features from cognitive eeg signals using relieff and mrmr algorithm. In Somsubhra Gupta, Sandip Bag, Karabi Ganguly, Indranath Sarkar, and Papun Biswas, editors, *Advancements of Medical Electronics*, pages 125–136, New Delhi, 2015. Springer India. ISBN 978-81-322-2256-9.

[116] J McCall, A E I Brownlee, and S Shakya. Applications of distribution estimation using markov network modelling (deum). *Markov Networks in Evolutionary Computation*, pages 193–207, 2012. ISSN 18674534 18674542.

[117] Dennis J McFarland, William A Sarnacki, and Jonathan R Wolpaw. Electroencephalographic (EEG) control of three-dimensional movement. *Journal of Neural Engineering*, 7(3):036007, jun 2010. ISSN 1741-2560. doi: 10.1088/1741-2560/7/3/036007.

[118] Peter Meinicke, Matthias Kaper, Manfred Heumann, and Helge Ritter. Improving transfer rates in brain computer interfacing: A case study. In *Advances In Neural Information Processing Systems 15*, pages 1107–1114. MIT Press, 2003.

[119] José del R. Millán. Brain-Machine Interfaces. *Principles of Tissue Engineering: Fourth Edition*, pages 1343–1352, 2013. doi: 10.1016/B978-0-12-398358-9.00063-X.

[120] Data Mining and Jerome H Friedman. On Bias, Variance, 0 1 - Loss , and the Curse of Dimensionality. (March 1997), 2014. doi: 10.1023/A.

[121] Pratik Mutha and Haaland. the effects of brain lateralization on motor control and adaption. 44(6):455–469, 2013. doi: 10.1080/00222895.2012.747482.THE.

[122] Hoai Bach Nguyen, Bing Xue, Ivy Liu, and Mengjie Zhang. Filter based backward elimination in wrapper based PSO for feature selection in classification. *Proceedings of the 2014 IEEE Congress on Evolutionary Computation, CEC 2014*, pages 3111–3118, 2014. doi: 10.1109/CEC.2014.6900657.

[123] Luis Fernando Nicolas-Alonso and Jaime Gomez-Gil. Brain computer interfaces, a review. *Sensors*, 12:1211–1279, 2012. ISSN 14248220. doi: 10.3390/s120201211.

[124] Luis Fernando Nicolas-Alonso and Jaime Gomez-Gil. Brain computer interfaces, a review. *Sensors (Basel, Switzerland)*, 12(2):1211–79, jan 2012. ISSN 1424-8220. doi: 10.3390/s120201211.

[125] Shahryar Noei, Pooya Ashtari, Mehran Jahed, and Bijan Vosoughi Vahdat. Classification of eeg signals using the spatio-temporal feature selection via the elastic net. pages 232–236, 01 2016. doi: 10.1109/ICBME.2016.7890962.

[126] Quentin Noirhomme, R.I. Kitney, and Benoît Macq. Single-Trial EEG Source Reconstruction for Brain-Computer Interface. *IEEE Transactions on Biomedical Engineering*, 55(5):1592–1601, may 2008. ISSN 0018-9294. doi: 10.1109/TBME.2007.913986.

[127] Vangelis P Oikonomou, Kostas Georgiadis, George Liaros, Spiros Nikolopoulos, and Ioannis Kompatsiaris. A Comparison Study on EEG Signal Processing Techniques Using Motor Imagery EEG Data. In *2017 IEEE 30th International Symposium on Computer-Based Medical Systems (CBMS)*, number 1, pages 781–786. IEEE, jun 2017. ISBN 978-1-5386-1710-6. doi: 10.1109/CBMS.2017.113.

[128] Akinari Onishi and Kiyohisa Natsume. Overlapped partitioning for ensemble classifiers of P300-based brain-computer interfaces. *PLoS ONE*, 9(4), 2014. ISSN 19326203. doi: 10.1371/journal.pone.0093045.

[129] Alexis Ortiz-Rosario and Hojjat Adeli. Brain-computer interface technologies: from signal to action. *Reviews in the Neurosciences*, 24(5):537–552, jan 2013. ISSN 2191-0200. doi: 10.1515/revneuro-2013-0032.

[130] Subash Padmanaban, Justin Baker, and Bradley Greger. Feature Selection Methods for Robust Decoding of Finger Movements in a Non-human Primate. *Frontiers in Neuroscience*, 12(FEB):1–15, feb 2018. ISSN 1662-453X. doi: 10.3389/fnins.2018.00022.

[131] Rajesh C. Panicker, Sadasivan Puthusserypady, and Ying Sun. An Asynchronous P300 BCI With SSVEP-Based Control State Detection. *IEEE Transactions on Biomedical Engineering*, 58(6):1781–1788, jun 2011. ISSN 0018-9294. doi: 10.1109/TBME.2011.2116018.

[132] Joshua L. Payne, Casey S. Greene, Douglas P. Hill, and Jason H. Moore. *Exploitation of Linkage Learning in Evolutionary Algorithms*, chapter Sensible Initialization of a Computational Evolution System Using Expert Knowledge for Epistasis Analysis in Human Genetics, pages 215–226. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010. ISBN 978-3-642-12834-9. doi: 10.1007/978-3-642-12834-9_10.

[133] Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(8):1226–1238, 2005. ISSN 01628828. doi: 10.1109/TPAMI.2005.159.

[134] David A. Peterson, James N. Knight, Michael J. Kirby, Charles W. Anderson, and Michael H. Thaut. Feature selection and blind source separation in an eeg-based brain-computer interface. *EURASIP Journal on Advances in Signal Processing*, 2005(19):218613, Nov 2005. ISSN 1687-6180. doi: 10.1155/ASP.2005.3128.

[135] G. Pfurtscheller, Ch Neuper, D. Flotzinger, and M. Pregenzer. EEG-based discrimination between imagination of right and left hand movement.

Electroencephalography and Clinical Neurophysiology, 103(6):642–651, dec 1997. ISSN 00134694. doi: 10.1016/S0013-4694(97)00080-1.

[136] G. Pfurtscheller, B. Graimann, J. E. Huggins, S. P. Levine, and L. A. Schuh. Spatiotemporal patterns of beta desynchronization and gamma synchronization in corticographic data during self-paced movement. *Clinical Neurophysiology*, 114(7):1226–1236, 2003. ISSN 13882457. doi: 10.1016/S1388-2457(03)00067-1.

[137] Gert Pfurtscheller, Gernot R. Müller, Jörg Pfurtscheller, Hans Jürgen Gerner, and Rüdiger Rupp. Thought-control of functional electrical stimulation to restore hand grasp in a patient with tetraplegia. *Neuroscience Letters*, 351(1):33–36, nov 2003. ISSN 03043940. doi: 10.1016/S0304-3940(03)00947-9.

[138] Elizabeth Radetic and Martin Pelikan. Spurious dependencies and EDA scalability. In *Proc. GECCO*, pages 303–310, 2010. doi: 10.1145/1830483.1830543.

[139] Alain Rakotomamonjy and Vincent Guigue. BCI Competition III: Dataset II- Ensemble of SVMs for BCI P300 Speller. *IEEE Trans Biomed Eng*, 55(3):1147–1154, mar 2008. ISSN 0018-9294. doi: 10.1109/TBME.2008.915728.

[140] Rabie A. Ramadan and Athanasios V. Vasilakos. Brain computer interface: control signals review. *Neurocomputing*, 223(August 2016):26–44, 2017. ISSN 18728286. doi: 10.1016/j.neucom.2016.10.024.

[141] Alimed Celecia Ramos and Marley Vellasco. Feature Selection Methods Applied to Motor Imagery Task Classification. Number Mi, 2016. ISBN 9781509051052.

[142] Alimed Celecia Ramos, Rene Gonzalez Hernandez, and Marley Vellasco. Feature Selection methods applied to Motor Imagery task classification. In *2016 IEEE Latin American Conference on Computational Intelligence (LA-CCI)*, number Mi, pages 1–6. IEEE, nov 2016. ISBN 978-1-5090-5105-2. doi: 10.1109/LA-CCI.2016.7885731.

[143] Alimed Celecia Ramos, Rene Gonzalez Hernandez, and Marley Vellasco. Feature Selection methods applied to Motor Imagery task classification. In *2016 IEEE Latin American Conference on Computational Intelligence (LA-CCI)*, number Mi, pages 1–6. IEEE, nov 2016. ISBN 978-1-5090-5105-2. doi: 10.1109/LA-CCI.2016.7885731.

[144] Izabela Rejer. EEG Feature Selection for BCI Based on Motor Imaginary Task. *Foundations of Computing and Decision Sciences*, 37(4):283–292, jan 2012. ISSN 2300-3405. doi: 10.2478/v10209-011-0016-7.

[145] Izabela Rejer. Genetic algorithm with aggressive mutation for feature selection in BCI feature space. *Pattern Analysis and Applications*, 18(3): 485–492, 2014. ISSN 14337541. doi: 10.1007/s10044-014-0425-3.

[146] Juha Reunanen. Overfitting in Making Comparisons Between Variable Selection Methods. *Journal ofMachine Learning Research*, 3(Mar):1371–1382, 2003. ISSN 0003-6951. doi: 10.1162/153244303322753715.

[147] B. Rivet, A. Souloumiac, G. Gibert, and V. Attina. "p300 speller" brain-computer interface: Enhancement of p300 evoked potential by spatial filters. In *2008 16th European Signal Processing Conference*, pages 1–5, Aug 2008.

[148] Bertrand Rivet, Antoine Souloumiac, Guillaume Gibert, and Virginie Attina. "p300 speller" brain-computer interface: Enhancement of p300 evoked potential by spatial filters. In *EUSIPCO*, 2008.

[149] D Robert, M Andra, B Bruce, L Alexander, and P Kenneth. Functional magnetic resonance imaging : the basics of blood-oxygen-level dependent (BOLD) imaging. 5(Oct):1–12, 1998.

[150] Roca-González & Roca-Dorda Rodríguez-Bermúdez, García-Laencina. Efficient feature selection and linear discrimination of EEG signals. 2013.

[151] GermÃ¡n RodrÃguez-BermÃºdez, Pedro J. GarcÃa-Laencina, JoaquÃn Roca-GonzÃ¡lez, and JoaquÃn Roca-Dorda. Efficient feature selection and linear discrimination of eeg signals. *Neurocomputing*, 115:161 – 165,

2013. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2013.01.001.

[152] M. D. Salwani and Y. Jasmy. Relative wavelet energy as a tool to select suitable wavelet for artifact removal in eeg. In *2005 1st International Conference on Computers, Communications, Signal Processing with Special Track on Biomedical Engineering*, pages 282–287, Nov 2005. doi: 10.1109/CCSP.2005.4977207.

[153] Noelia Sánchez-Maroño, Amparo Alonso-Betanzos, and María Tombilla-Sanromán. Filter Methods for Feature Selection A Comparative Study. *Intelligent Data Engineering and Automated Learning - IDEAL 2007*, pages 178–187. ISSN 0302-9743. doi: 10.1007/978-3-540-77226-2_19.

[154] A. R. Satti, D. Coyle, and G. Prasad. Spatio-spectral and temporal parameter searching using class correlation analysis and particle swarm optimization for a brain computer interface. In *2009 IEEE International Conference on Systems, Man and Cybernetics*, pages 1731–1735, Oct 2009. doi: 10.1109/ICSMC.2009.5346679.

[155] Gerwin Schalk and Jürgen Mellinger. *A Practical Guide to Brain-Computer Interfacing with BCI2000*. Springer London, London, 2010. ISBN 978-1-84996-091-5. doi: 10.1007/978-1-84996-092-2.

[156] Andrew B Schwartz, X Tracy Cui, Douglas J Weber, and Daniel W Moran. Brain-controlled interfaces: movement restoration with neural prosthetics. *Neuron*, 52(1):205–20, oct 2006. ISSN 0896-6273. doi: 10.1016/j.neuron.2006.09.019.

[157] Stephen H. Scott. Inconvenient Truths about neural processing in primary motor cortex. *The Journal of Physiology*, 586(5):1217–1224, mar 2008. ISSN 00223751. doi: 10.1113/jphysiol.2007.146068.

[158] Baha Sen, Peker Musa, Cavusoglu Abdullah, and Celebi Fatih. A Comparative Study on Classification of Sleep Stage Based on EEG Signals Using Feature Selection and Classification Algorithms. *Journal of Medical Systems*, 38(3):18, mar 2014. ISSN 0148-5598. doi: 10.1007/s10916-014-0018-0.

[159] Claude E Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(July 1928):379–423, 1948. ISSN 07246811. doi: 10.1145/584091.584093.

[160] Kilho Shin, Danny Fernandes, and Seiya Miyazaki. Consistency measures for feature selection: A formal definition, relative sensitivity comparison and a fast algorithm. *IJCAI International Joint Conference on Artificial Intelligence*, pages 1491–1497, 2011. ISSN 10450823. doi: 10.5591/978-1-57735-516-8/IJCAI11-251.

[161] D. M. Simpson, A. F. C. Infantosi, J. F. C. Junior, A. J. Peixoto, and L. M. d. Abrantes. On the selection of autoregressive order for electroencephalographic (eeg) signals. In *38th Midwest Symposium on Circuits and Systems. Proceedings*, volume 2, pages 1353–1356 vol.2, Aug 1995. doi: 10.1109/MWSCAS.1995.510349.

[162] American Electroencephalographic Society. Guideline 8: Guidelines for recording clinical eeg on digital media. In *Journal of Clinical Neurophysiology*, volume 23, pages 122–124, 2006.

[163] Ramesh Srinivasan. *Acquiring Brain Signals from Outside the Brain*, volume 16. 2010. ISBN 9780199600458. doi: 10.1093/acprof.

[164] Philipp Stankevich and Vladimir Spitsyn. A review of Brain-Computer Interface technology. *2015 International Siberian Conference on Control and Communications (SIBCON)*, (January 2016):1–6, 2015. doi: 10.1109/SIBCON.2015.7147225.

[165] Ian H Stevenson and Konrad P Kording. How advances in neural recording affect data analysis. *Nature Neuroscience*, 14(2):139–142, feb 2011. ISSN 1097-6256. doi: 10.1038/nn.2731.

[166] Matthew Sybeldon, Lukas Schmit, and Murat Akcakaya. Transfer Learning for SSVEP Electroencephalography Based Brain Computer Interfaces Using Learn.NSE and Mutual Information. *Entropy*, 19(1):41, 2017. ISSN 1099-4300. doi: 10.3390/e19010041.

[167] Andrea Tacchino, Catarina Saiote, Giampaolo Brichetto, Giulia Bommarito, Luca Roccatagliata, Christian Cordano, Mario A. Battaglia, Gian L. Mancardi, and Matilde Inglese. Motor Imagery as a Function of Disease Severity in Multiple Sclerosis: An fMRI Study. *Frontiers in Human Neuroscience*, 11(January):1–10, jan 2018. ISSN 1662-5161. doi: 10.3389/fnhum.2017.00628.

[168] Farajollah Tahernezhad-Javazm, Vahid Azimirad, and Maryam Shoaran. A review and experimental study on the application of classifiers and evolutionary algorithms in EEG based brain machine interface systems. *Journal of Neural Engineering*, 15(2):021007, apr 2018. ISSN 1741-2560. doi: 10.1088/1741-2552/aa8063.

[169] Feng Tan, Xuezheng Fu, Yanqing Zhang, and Anu G. Bourgeois. A genetic algorithm-based method for feature subset selection. *Soft Computing*, 12(2):111–120, 2008. ISSN 14327643. doi: 10.1007/s00500-007-0193-8.

[170] Michael Tangermann, Klaus-Robert Müller, Ad Aertsen, Niels Birbaumer, Christoph Braun, Clemens Brunner, Robert Leeb, Carsten Mehring, Kai J. Miller, Gernot R. Müller-Putz, Guido Nolte, Gert Pfurtscheller, Hubert Preissl, Gerwin Schalk, Alois Schlögl, Carmen Vidaurre, Stephan Waldert, and Benjamin Blankertz. Review of the BCI Competition IV. *Frontiers in Neuroscience*, 6(JULY):1–31, 2012. ISSN 1662-4548. doi: 10.3389/fnins.2012.00055.

[171] Dirk Thierens. Population-based iterated local search: Restricting neighborhood search by crossover. In Kalyanmoy Deb, editor, *Genetic and Evolutionary Computation – GECCO 2004*, pages 234–245, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg. ISBN 978-3-540-24855-2.

[172] Eoin Thomas, Matthew Dyson, and Maureen Clerc. An analysis of performance evaluation for motor-imagery based BCI. *Journal of neural engineering*, 10(3):031001, 2013. ISSN 1741-2552. doi: 10.1088/1741-2560/10/3/031001.

[173] Dattaprasad Torse, Raghavendra Maggavi, and S A. Pujari. Nonlinear blind source separation for eeg signal pre-processing in brain computer

interface system for epilepsy. *International Journal of Computer Applications*, 50:12–19, 07 2012. doi: 10.5120/7838-0911.

[174] Wenting Tu and Shiliang Sun. Spatial filter selection with lasso for eeg classification. In Longbing Cao, Jiang Zhong, and Yong Feng, editors, *Advanced Data Mining and Applications*, pages 142–149, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg. ISBN 978-3-642-17313-4.

[175] Ayad Turky, I. Moser, and Aldeida Aleti. An iterated local search with guided perturbation for the heterogeneous fleet vehicle routing problem with time windows and three-dimensional loading constraints. In Markus Wagner, Xiaodong Li, and Tim Hendtlass, editors, *Artificial Life and Computational Intelligence*, pages 279–290, Cham, 2017. Springer International Publishing. ISBN 978-3-319-51691-2.

[176] Roberto Vega, Touqir Sajed, Kory Wallace Mathewson, Kriti Khare, Patrick M. Pilarski, Russ Greiner, Gildardo Sanchez-Ante, and Javier M. Antelis. Assessment of feature selection and classification methods for recognizing motor imagery tasks from electroencephalographic signals. *Artificial Intelligence Research*, 6(1):37–51, 2016. ISSN 1927-6982. doi: 10.5430/air.v6n1p37.

[177] M. Vourkas, S. Micheloyannis, and G. Papadourakis. Use of ann and hjorth parameters in mental-task discrimination. In *2000 First International Conference Advances in Medical Signal and Information Processing (IEE Conf. Publ. No. 476)*, pages 327–332, 2000. doi: 10.1049/cp:20000356.

[178] Jiri Vrba and Stephen E. Robinson. Signal Processing in Magnetoencephalography. *Methods*, 25(2):249–271, oct 2001. ISSN 10462023. doi: 10.1006/meth.2001.1238.

[179] Stephan Waldert, Tobias Pistohl, Christoph Braun, Tonio Ball, Ad Aertsen, and Carsten Mehring. A review on directional information in neural signals for brain-machine interfaces. *Journal of Physiology-Paris*, 103(3-5):244–254, may 2009. ISSN 09284257. doi: 10.1016/j.jphysparis.2009.08.007.

[180] Peitao Wang, Jun Lu, Bin Zhang, and Zeng Tang. A review on transfer learning for brain-computer interface classification. *2015 5th International Conference on Information Science and Technology, ICIST 2015*, pages 315–322, 2015. doi: 10.1109/ICIST.2015.7288989.

[181] Y. Wang, Ruiping Wang, X. Gao, Bo Hong, and Shangkai Gao. A Practical VEP Based Brain Computer Interface. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14(2):234–240, jun 2006. ISSN 1534-4320. doi: 10.1109/TNSRE.2006.875576.

[182] Karl Weiss, Taghi M. Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big Data*, (1):9, dec 2016. ISSN 2196-1115. doi: 10.1186/s40537-016-0043-6.

[183] Darrell Whitley and Wenxiang Chen. Constant time steepest descent local search with lookahead for NK-landscapes and MAX-kSAT. In *Proceedings of the 14th Annual Conference on Genetic and Evolutionary Computation*, GECCO '12, pages 1357–1364, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1177-9. doi: 10.1145/2330163.2330351.

[184] J. R. Wolpaw and D. J. McFarland. Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans. *Proceedings of the National Academy of Sciences*, 101(51):17849–17854, dec 2004. ISSN 0027-8424. doi: 10.1073/pnas.0403504101.

[185] Jonathan R. Wolpaw. Brain-computer interfaces as new brain output pathways. *Journal of Physiology*, 579(3):613–619, 2007. ISSN 00223751. doi: 10.1113/jphysiol.2006.125948.

[186] Ran Xiao and Lei Ding. Evaluation of EEG Features in Decoding Individual Finger Movements from One Hand. *Computational and mathematical methods in medicine*, 2013:243257, jan 2013. ISSN 1748-6718. doi: 10.1155/2013/243257.

[187] Minpeng Xu, Jing Liu, Long Chen, Hongzhi Qi, Feng He, Peng Zhou, Xiaoman Cheng, Baikun Wan, and Dong Ming. Inter-subject information contributes to the ERP classification in the P300 speller. *Int'l IEEE/EMBS*

Conf. on Neural Engineering, 2015-July:206–209, 2015. ISSN 19483554. doi: 10.1109/NER.2015.7146596.

[188] Minpeng Xu, Jing Liu, Long Chen, Hongzhi Qi, Feng He, Peng Zhou, Baikun Wan, and Dong Ming. Incorporation of Inter-Subject Information to Improve the Accuracy of Subject-Specific P300 Classifiers. *International Journal of Neural Systems*, 26(3):1–12, 2016. ISSN 01290657. doi: 10.1142/S0129065716500106.

[189] Peng Xu, Tiejun Liu, Rui Zhang, Yangsong Zhang, and Dezhong Yao. Using particle swarm to select frequency band and time interval for feature extraction of eeg based bci. *Biomedical Signal Processing and Control*, 10:289 – 295, 2014. ISSN 1746-8094. doi: https://doi.org/10.1016/j.bspc.2013.08.012.

[190] Bing Xue, Mengjie Zhang, Will N. Browne, and Xin Yao. A Survey on Evolutionary Computation Approaches to Feature Selection. *IEEE Transactions on Evolutionary Computation*, 20(4):606–626, aug 2016. ISSN 1089-778X. doi: 10.1109/TEVC.2015.2504420.

[191] Ashkan Yazdani, Jong-Seok Lee, and Touradj Ebrahimi. Implicit emotional tagging of multimedia using EEG signals and brain computer interface. In *Proceedings of the first SIGMM workshop on Social media - WSM '09*, page 81, New York, New York, USA, 2009. ACM Press. ISBN 9781605587592. doi: 10.1145/1631144.1631160.

[192] J. Yu, S. Peng, Z. Wan, X. Liu, and X. Peng. A new implementation of recursive feature elimination algorithm for gene selection from microarray data. In *2009 WRI World Congress on Computer Science and Information Engineering, CSIE(CSIE)*, volume 03, pages 665–669, 03 2009. doi: 10.1109/CSIE.2009.75.

[193] Tian-Li Yu, Kumara Sastry, and David E. Goldberg. Linkage learning, overlapping building blocks, and systematic strategy for scalable recombination. In *Proc. of the 2005 Conf. on Genetic and evolutionary computation*, GECCO '05, pages 1217–1224, New York, NY, USA, 2005. ACM. ISBN 1-59593-010-8. doi: 10.1145/1068009.1068209.

[194] Qingfu Zhang and Jianyong Sun. Iterated local search with guided mutation. In *2006 IEEE International Conference on Evolutionary Computation*, pages 924–929, 2006. doi: 10.1109/CEC.2006.1688410.

[195] Zexuan Zhu, Sen Jia, and Zhen Ji. Towards a memetic feature selection paradigm. *IEEE Computational Intelligence Magazine*, 5(2):41–53, 2010. ISSN 1556603X. doi: 10.1109/MCI.2010.936311.

Part IX

APPENDICES

APPENDIX

A.1 FEATURE REFERENCE TABLE FOR DATASETS D1

Channel:	C3									Cz									C4								
Second:	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
8-13	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
8-9	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54
9-10	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81
10-11	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100	101	102	103	104	105	106	107	108
11-12	109	110	111	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	128	129	130	131	132	133	134	135
12-13	136	137	138	139	140	141	142	143	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	160	161	162
13-30	163	164	165	166	167	168	169	170	171	172	173	174	175	176	177	178	179	180	181	182	183	184	185	186	187	188	189
13-17	190	191	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216
17-20	217	218	219	220	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240	241	242	243
20-23	244	245	246	247	248	249	250	251	252	253	254	255	256	257	258	259	260	261	262	263	264	265	266	267	268	269	270
23-26	271	272	273	274	275	276	277	278	279	280	281	282	283	284	285	286	287	288	289	290	291	292	293	294	295	296	297
26-30	298	299	300	301	302	303	304	305	306	307	308	309	310	311	312	313	314	315	316	317	318	319	320	321	322	323	324

Frequencies (Hz)

Table A.1: Indices of Power Spectral Density features according to the frequency, channel, and time epoch for Dataset D1: Berlin BCI Competition III dataset.

Channel:		F3	F1	Fz	F2	F4	FC5	FC3	FC1	FCz	FC2	FC4	FC6	C5	C3	C1	Cz	C2	C4	C6	CP5	CP3	CP1	CPz	CP2	CP4	CP6	O1	O2
Frequencies (Hz)	**8-13**	1	13	25	37	49	61	73	85	97	109	121	133	145	157	169	181	193	205	217	229	241	253	265	277	289	301	313	325
	8-9	2	14	26	38	50	62	74	86	98	110	122	134	146	158	170	182	194	206	218	230	242	254	266	278	290	302	314	326
	9-10	3	15	27	39	51	63	75	87	99	111	123	135	147	159	171	183	195	207	219	231	243	255	267	279	291	303	315	327
	10-11	4	16	28	40	52	64	76	88	100	112	124	136	148	160	172	184	196	208	220	232	244	256	268	280	292	304	316	328
	11-12	5	17	29	41	53	65	77	89	101	113	125	137	149	161	173	185	197	209	221	233	245	257	269	281	293	305	317	329
	12-13	6	18	30	42	54	66	78	90	102	114	126	138	150	162	174	186	198	210	222	234	246	258	270	282	294	306	318	330
	13-30	7	19	31	43	55	67	79	91	103	115	127	139	151	163	175	187	199	211	223	235	247	259	271	283	295	307	319	331
	13-17	8	20	32	44	56	68	80	92	104	116	128	140	152	164	176	188	200	212	224	236	248	260	272	284	296	308	320	332
	17-20	9	21	33	45	57	69	81	93	105	117	129	141	153	165	177	189	201	213	225	237	249	261	273	285	297	309	321	333
	20-23	10	22	34	46	58	70	82	94	106	118	130	142	154	166	178	190	202	214	226	238	250	262	274	286	298	310	322	334
	23-26	11	23	35	47	59	71	83	95	107	119	131	143	155	167	179	191	203	215	227	239	251	263	275	287	299	311	323	335
	26-30	12	24	36	48	60	72	84	96	108	120	132	144	156	168	180	192	204	216	228	240	252	264	276	288	300	312	324	336

Table A.2: Indices of Power Spectral Density features according to the frequency, channel, and time epoch for Dataset D2: Berlin BCI Competition IV dataset.

A.3 FEATURE REFERENCE TABLE FOR DATASETS D3

Epoch:		1						2						3					
Channel:		**C3**	**Cz**	**C4**	**CP3**	**CPZ**	**CP4**	**C3**	**Cz**	**C4**	**CP3**	**CPZ**	**CP4**	**C3**	**Cz**	**C4**	**CP3**	**CPZ**	**CP4**
Frequencies (Hz)	**8-13**	1	13	25	37	49	61	73	85	97	109	121	133	145	157	169	181	193	205
	8-9	2	14	26	38	50	62	74	86	98	110	122	134	146	158	170	182	194	206
	9-10	3	15	27	39	51	63	75	87	99	111	123	135	147	159	171	183	195	207
	10-11	4	16	28	40	52	64	76	88	100	112	124	136	148	160	172	184	196	208
	11-12	5	17	29	41	53	65	77	89	101	113	125	137	149	161	173	185	197	209
	12-13	6	18	30	42	54	66	78	90	102	114	126	138	150	162	174	186	198	210
	13-30	7	19	31	43	55	67	79	91	103	115	127	139	151	163	175	187	199	211
	13-17	8	20	32	44	56	68	80	92	104	116	128	140	152	164	176	188	200	212
	17-20	9	21	33	45	57	69	81	93	105	117	129	141	153	165	177	189	201	213
	20-23	10	22	34	46	58	70	82	94	106	118	130	142	154	166	178	190	202	214
	23-26	11	23	35	47	59	71	83	95	107	119	131	143	155	167	179	191	203	215
	26-30	12	24	36	48	60	72	84	96	108	120	132	144	156	168	180	192	204	216

Table A.3: Indices of Power Spectral Density features according to the frequency, channel, and time epoch for Dataset D3: Riken - Subject A.

	Participant					
	1	**2**	**3**	**4**	**5**	**6-9**
Diagnosis	Cerebral palsy	Multiple sclerosis	Late-stage ALS	Traumatic brain spinal-cord injury	Post-anoxic encephalopathy	N/A
Age	56	51	47	33	43	27.7-32.3
Age at illness onset	0	37	39	27	37	
Sex	M	M	M	F	M	M
Speech	Mild disarthria	Mild disarthria	Severe disarthria	Mild disarthria	Severe hypophony	
Limb control	Weak	Weak	Very weak	Weak	Very weak	
Respiration	Normal	Normal	Weak	Normal	Normal	
Voluntary eye movement	Normal	Mild nystagmus	Normal	Normal	Balint's syndrome	
Notes				Only Female	Excluded	PhD Students

Table A.4: Table provides a description of the participants within dataset D4: P300 Speller (Hoffman)