Investigating the ability of machine learning techniques to provide insight into the aetiology of complex psychiatric genetic disorders

Timothy Edward Vivian-Griffiths

A thesis presented for the degree of
Doctor of Philosophy

School of Medicine
Cardiff University
United Kingdom

April 2017

Declaration

This work has not been submitted in substance for any other degree or award at this or any other university or place of learning, nor is being submitted concurrently in candidature for any degree or other award.

Signed .... (candidate)

Date ....

Statement 1

This thesis is being submitted in partial fulfillment of the requirements for the degree of PhD.

Signed .... (candidate)

Date ....

Statement 2

This thesis is the result of my own independent work/investigation, except where otherwise stated, and the thesis has not been edited by a third party beyond what is permitted by Cardiff University's Policy on the Use of Third Party Editors by Research Degree Students. Other sources are acknowledged by explicit references. The views expressed are my own.

Signed .... (candidate)

Date ....

Statement 3

I hereby give consent for my thesis, if accepted, to be available online in the University's Open Access repository and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed .... (candidate)

Date ....

Acknowledgements

I would like to thank all of my supervisors for their support and mentoring throughout my time at Cardiff University: Prof. Sir. Michael Owen, Prof. Valentina Escott-Price, Dr. Andrew Pocklington and Dr. Andreas Artemiou. In addition to academic support, I appreciate the time eating Mexican food with margharitas in Atlanta, Thai Food in Toronto, and regular coffees in Cardiff.

I would also like to give my appreciation to all the Cardiff PhD colleagues: Nick, Hannah, Lucy, Jack, Laura and Alison.

In addition to the support of Cardiff University, I would like to thank the Wellcome Trust for their funding throughout my studies.

A very special thanks to my family members for encouraging me through both the peaks and the troughs: my parents Jim and Val, my loving wife Solveiga.

This thesis is dedicated to the memory of my grandparents: Ted and Margaret Reeves, and Oliver and Marion Vivian-Griffiths

Contents

Acronyms

ADHD Attention Deficit Hyperactivity Disorder.

ALIGATOR Association LIst Go AnnoTatOR.

ARC Activity-Regulated Cytoskeleton-associated.

ASD Autism Spectrum Disorder.

AUC Area-Under-the-Curve.

BiGS Bipolar Genome Studies.

BN Bayesian Network.

BP Base Pair.

CNS Central Nervous System.

CNV Copy Number Variant.

COGS COGnition in Schizophrenia.

CPU Central Processing Unit.

CV Cross-Validation.

DNA DeoxyriboNucleic Acid.

DSM Diagnostic and Statistical Manual of Mental Disorders.

DTI Diffusion Tensor Imaging.

DZ Dizygotic.

EEG Electroencephalography.

ExAC Exome Aggregation Consortium.

f-MRI Functional Magnetic Resonance Imaging.

FDA Federal Drug Administration.

FMRP Fragile-X Mental Retardation Protein.

FPR False Positive Rate.

GLM Generalised Linear Model.

GO Gene Ontology.

GPU Graphical Processing Unit.

GSEA Gene Set Enrichment Analysis.

GWAS Genome-Wide Association Study.

HGP Human Genome Project.
HWE Hardy-Weinberg Equilibrium.

I1M Illumina 1M.
IBD Identity By Descent.
ISC International Schizophrenia Consortium.

KB Kilo-Base.
KEGG Kyoto Encyclopedia of Genes and Genomes.

LD Linkage Disequilibrium.
LoF Loss of Function.
LOR Log Odds Ratio.
LR Logistic Regression.
LTD Long Term Depression.

MCI Mild Cognitive Impairment.
MCMC Monte Marlo Markov Chain.
MD Major Depression.
MDD Major Depressive Disorder.
MDR Multifactor Dimensionality Reduction.
MDS Multi-Dimensional Scaling.
MEG Magnetoencephalography.
MGI Mouse Genome Informatics.
mGluR Metabotropic Glutamate Receptor.
MHC Major Histocompatibility Complex.
mRNA Messenger Ribose-Nucleic Acid.
MZ Monozygotic.

NIMH National Institue of Mental Health.
NMDA N-methyl-D-aspartate.

OMNI Illumina Omni Express.
OR Odds-Ratio.

PCA Principal Component Analysis.
PET Positron Emission Topography.
PGC Psychiatric Genetics Consortium.
PGC-2 Psychiatric Genetics Consortium - 2.
PSD Post Synaptic Density.
PTV Protein Truncating Variant.

QC Quality-Control.

RA Rheumatoid Arthritis.
RAM Random Access Memory.
RBF Radial-Basis Function.
RF Random Forest.
RNA Ribose-Nucleic Acid.
ROC Receiver Operating Characteristic.

s-MRI Structural Magnetic Resonance Imaging.
SNP Single Nucleotide Polymorphism.
SNV Single Nucleotide Variation.
SVM Support Vector Machine.

T1D Type 1 diabetes.
T2D Type 2 diabetes.
TPR True Positive Rate.

UTR Un-Translated Region.

VIM Variable Importance Measures.

WTCCC Wellcome Trust Case/Control Consortium.

XOR eXclusive-OR.

List of Figures

List of Tables

Abstract

One of the biggest challenges in psychiatric genetics is examining the effects of interactions between genetic variants on the aetiologies of complex disorders. Current techniques involve looking at linear combinations of the variants, as considering all the possible combinations of interactions is computationally unfeasible. The work in this thesis attempted to address this problem by using a machine learning model called a Support Vector Machine (SVM). These algorithms are capable of either building linear models or using kernel methods to consider the effects of interactions.

The dataset used for all of the experiments was taken from a study looking into sufferers of treatment-resistant schizophrenia receiving the medication, Clozapine, with controls taken from the Wellcome Trust Case/Control Consortium study.

The first experiment used information from the individual Single Nucleotide Polymorphisms (SNPs) as inputs to the SVMs, and compared the results with a technique called a polygenic score, a linear combination of the risk contributions of the SNPs that provides a single risk score for each individual. When more SNPs were entered into the models, one of the non-linear kernels provided better results than the linear SVMs. The second experiment attempted to explain this behaviour by using simulated phenotypes made from different contributions of main effects and interactions. The results strongly suggested that interactions were making a contribution. The final experiment looked at using risk scores made from gene sets. The models identified a set involved in synaptic development that has been previously implicated in schizophrenia, and when the scores from the individual genes were entered, the non-linear kernels again showed improvement, suggesting that there are interactions occurring between these genes.

The conclusion was that using SVMs is an effective way to assess for the possible presence of interactions, before searching for them explicitly.

1. Introduction

1.1. Background of Schizophrenia

The term “schizophrenia” is used to describe a collection of symptoms that are observed in psychiatric patients who have been given this diagnosis. While it is a clinical term, this word is commonly known amongst members of the general population. It is an incredibly debilitating disorder which causes a great deal of distress to both the sufferers and their families. There is also a large economic cost to society due to the concomitant rises in rates of unemployment, homelessness and substance abuse, in addition to the high costs of medical treatment. In the United Kingdom, the costs were estimated to be £6.7 billion annually in 2004/05 (Mangalore et al., 2007), and Owen et al. (2016) point out that more recently, in 2012, the Schizophrenia Commission has estimated that the cost has elevated to £11.8 billion annually for England alone (Schizophrenia Commission, 2012). By far the most well known symptoms are the so-called *positive* symptoms, which include delusional thoughts, feelings of distorted reality and hallucinations, of which the most common are in the auditory sense, such as the hearing of voices. However, there are also debilitating *negative* symptoms of blunted emotion and cognitive functions, which can include anhedonia, apathy and reduced social function and drive (Tandon et al., 2009).

While the term of schizophrenia is relatively commonplace, the actual understanding behind what is encapsulated in this single word is far from obvious. There are ongoing debates about topics related to schizophrenia including: can it actually be classified as a single entity, or should it be considered as part of a spectrum of different disorders whose boundaries are not so clearly defined? Or is it best described as a disease, a syndrome, or a psychiatric disorder (Tandon et al., 2008b)? This introduction chapter to the thesis will discuss these issues, and how they have developed throughout psychiatric research since the 18th and 19th centuries; including a detailed description of matters regarding diagnosis that have also developed during this period. Following this, a review of past and current research pertaining to the search for any genetic aetiology will be discussed. This will provide the basis and the background to explain why the methods and techniques used in this doctoral degree were utilised in the manners that they were;

although a more detailed description of the methods is given in chapter 2. Initially, the early developments of the psychotic symptoms, and the origin of term schizophrenia will be examined. While it might not seem necessary to trace back so far to examine current developments in the field, there are, in fact, still issues arising today that were present back then as well; namely the nosological issues of whether it should be considered as a category of psychiatric symptoms, or a complex spectrum.

The beliefs and understandings concerning the taxonomy of psychiatric disorders have been debated and disputed for over a century, and continue to evolve to this day. Throughout the 18th and 19th centuries, with the advances made across many scientific fields, there was a concerted effort to classify psychiatric disorders that previously would have been referred to as “madness” or “insanity” and considered simply as impairments of intellect (Berrios and Beer, 1994). It is believed that an Austrian physician and philosopher, Baron Ernst von Feuchtersleben, was the first person to use the term “psychosis” in its current form. Utilising the ideas of mind/body duality, he used this term to describe disorders that he believed were diseases that affected the personality instead of the body or the mind alone, and stressed that they originated from the relationships between the two. This caused him to state that psychosis was a sub-category of “neurosis”, a term developed by the Scottish physician William Cullen to describe disorders pertaining to disturbances in the nervous system (Beer, 1996).

Following the introduction of the term psychosis, there was a debate raging amongst German speaking psychiatrists as to whether the observed psychiatric disorders and symptoms should be classified as separate categories or part of a single underlying entity. Wilheim Greisinger was one of the “unitarians” and developed the umbrella terminology of *Einheitspsychose*, meaning “Unitary Psychosis” to describe the belief that there is fundamentally one form of psychosis and any observable differences are due to environmental or endogenous factors (Berrios and Beer, 1994; Bruijnzeel and Tandon, 2011). Discussions and debates between the unitarians and those who believed in using different classifications continued for the remainder of the 19th century, and there is documentation of some researchers shifting opinions during this time. One psychiatrist, Karl Ludwig Kahlbaum initially took the idea that all psychiatric disorders and insanities were variants on one underlying disease, but later went on to propose ideas for different categories. It was Kahlbaum who had a large influence on Emil Kraepelin, a major figure in the 20th century, credited with providing some of the early pioneering, although controversial, ideas and developments that helped to understand more about the nature of schizophrenia to the present day (Tandon et al., 2009).

1.1.1. From the 1850s to DSM-V

It was in the latter half of the 19th century that developments that have shaped our understanding of schizophrenia began to emerge. Kahlbaum, together with another German psychiatrist, Ewald Hecker, went on to describe distinct disorders that formed the basis of Kraepelin's work. Kahlbaum is described as being a key figure in the development of the field of psychiatric nosology, and his work is notable in the fact that he based his classification methods not on the immediately observable symptoms seen in patients, but on the course, development and outcome of the symptoms. Critically, he made separations between two different clusters of disorders. He used the term *vecordia* to describe patients who presented with limited disturbance, but showed consistency across time with their symptoms. Conversely, the term *vesania* was used in cases when the symptoms were more disturbed, with progressive deterioration resulting in dementia (Angst, 2002).

Kahlbaum is arguably most renowned for being the originator of the term *catatonia*, and stated that this was a sub-type of vesania. This term is still used in modern psychiatry to describe abnormalities in motor behaviours. These abnormalities include patients having a fixed posture and stare in addition to muted movements and stupor (Fink and Taylor, 2006). In his drive to develop the nosology of disorders further, Kahlbaum is also credited for introducing the term *hebephrenia*, used to describe cases of psychosis that developed during the years of adolescence. However, it was Hecker who made strides to state that schizophrenia was a clearly defined entity with a specific symptomology and course progression that always developed in connection with the onset of puberty (Taylor et al., 2010). It was the two classifications of catatonia and hebephrenia that inspired the work of Kraepelin.

While at the time the introduction of these categories were important developments in the field of psychiatry, and provided the foundations upon which our understanding of the disorder of schizophrenia has developed; modern opinions are more critical. Taylor et al. (2010) provided some critique of how the terms were used: while Kahlbaum claimed that catatonia was a distinct disease, it is now recognised as being a symptom of behaviours that are present across a range of psychiatric disorders including mood disorders. Indeed, in the latest version of the Diagnostic and Statistical Manual of Mental Disorders (DSM) - number 5 (American Psychiatric Association, 2013), the term is used as a specifier for disorders other than schizophrenia, including schizoaffective disorder, brief psychotic disorder, and major mood disorder; and it even has the category of "catatonia not otherwise specified" to describe patients who present with the symptoms yet do not have a clearly defined underlying diagnosis of a psychiatric disorder (Tandon et al., 2013b). In addition Taylor et al. (2010) criticised the classification term of hebephrenia due to it being based on the 19th century ideas of a *tripartite mind* made up of the three areas of concept, emotion, and will; which they stated is not a valid model in the present day. They did not, however, give a great deal of information about the tripartite mind, only to state that Kahlbaum used it as a basis to develop the diagnostic category and therefore, based on this it was included as a subtype of schizophrenia (hebephrenic schizophrenia) in the earlier 4th edition of the manual (American Psychiatric Association, 1994) but no longer features in DSM-V (Tandon et al., 2013a).

Early discrete classification - Kraepelin, Bleuler and Schneider

Kraepelin's motivation was to approach the subject of classification of psychiatric disorders in the same manner that was seen in the natural sciences and had a desire for the relatively new discipline of psychiatry to truly be seen as an established medical discipline. He wanted to develop a nosology that would aid in providing sufferers with an improved prognosis, prevention from disease development and superior treatment; and in order to do this, psychiatric conditions had to be clustered in such a way as to assume common underlying aetiology within the clusters (Roelcke, 1997). Kraepelin introduced the term *dementia praecox* in the 5th edition of his textbook in 1896 and continued to develop the idea in the 6th edition in 1899. This term was used to describe a disease that he believed to have a single source of aetiology with a clear pathological trajectory (Tandon and Maj, 2008). He is probably best known in the term *Kraepelinian Dichotomy* as he stated that this disorder was a clear differentiation from disorders which he described as *manic depressive insanities*, which were more similar (but not exact) in description to the current disorder of bipolar. In dementia praecox, he collected together Kahlbaum's catatonia and Hecker's hebephrenia together with his own contribution of paranoid dementia (Angst, 2002).

Such was his desire to develop a tangible nosological system, he was noted as taking a very meticulous and methodological approach to his research; taking care to collate together his observations of various symptoms across patients. It was based on the observations of the differences of the progression and outcome of the disorders in people that he created the dichotomy, using dementia praecox for those who had an earlier onset, often in early adulthood or even in adolescence, culminating in a deteriorating outcome (Tandon and Maj, 2008), and this was contrasted with the observable symptoms that had more of a periodic nature and more favourable outcomes, which came under the category of manic-depression (Hoenig, 1983). It must however be noted that creating these diagnostic categories has resulted in criticism, both from his contemporaries and modern day researchers and clinicians. His endeavours in the field were without doubt a massive undertaking and a pioneering development to try and take the same approach to established methods in the natural sciences to the new discipline of psychiatry. He always had the aim of discovering neuropathological aetiology that would explain his two different clinical categories and indeed concerns were raised at the time by fellow researchers that any classification system that claims to be based on aetiology must come from somatic or neurological markers, not from just psychological ones. However, despite being well intentioned, he never actually succeeded in this goal, in conjunction with the work of the famous Alois Alzheimer, whose research also failed to find the necessary neuropathological biomarkers. It was for this reason that he focused on the outcome and disease progression. However, it must be noted that he never provided any definitive clear cut boundaries for the categories and was always careful to stress that they were not pathognomic in nature. Despite taking these precautions, he still received criticism from those who said that providing any differentiations between diagnostic criteria must be based on somatic and neurological observations (Jablensky, 1999).

While Kraepelin provided the initial outlines of a disorder resembling schizophrenia, he did not create the term that we use today; this was done by a Swiss psychiatrist by the name of Eugen Bleuler, who gave it this name as he believed that one of the key features of the disease was a splitting of the so-called *psychic-features*. The symptoms that he focussed on when forming his categories for diagnosis are known for giving particular attention to the negative symptoms of the disease. He proposed some fundamental symptoms of schizophrenia that are called the four As. These could be described as containing two positive symptoms, and two negative ones. Bruijnzeel and Tandon (2011) listed them as follows:

- Positive Symptoms:
    - Associational disturbances (loose thought processes)
    - Autism - used in a different manner to modern day usage. Instead used to describe moving fom reality to fantasy.
- Negative Symptoms:
    - Affective disturbances (flattening of emotion)
    - Ambivalence

Noting the difficulties associated with creating clear boundaries to a disorder, based on observed behavioural traits, he also created latent categories to deal with milder cases of the disease and actually described this whole group of disorders as “the schizophrenias” (Bruijnzeel and Tandon, 2011). It is very interesting to see that even at the time of the origin of this term, there were acknowledgements of the extreme difficulties in creating clear criteria for diagnosis that are still very much in flux today.

Another German speaking psychiatrist who is acknowledged as making a considerable contribution towards our understanding of schizophrenia was Kurt Schneider (Bruijnzeel and Tandon, 2011; Hoenig, 1983; Tandon et al., 2009). It was his belief that the fundamental signature of the disease was the impairment and lack of empathetic understanding and communication, citing the so-called “un-understandability” of the personal experience of sufferers being a pathognomic hallmark. In 1959 this lead him to develop eleven “First rank” symptoms of schizophrenia, which would now come under the description of the positive symptoms. These include: auditory hallucinations, feelings that others have control of one’s body, beliefs that the thoughts of others are being transmitted into their minds or that their minds are being actively read by others. This list of symptoms have played a major role in shaping attitudes towards psychiatric diagnosis, featuring in the criteria up to the text review of the DSM-IV in 2000 (American Psychiatric Association, 2000). In fact, the writings and classifications from all the three contributors mentioned thus far, Kraepelin, Bleuler, and Schneider have played a part in shaping the criteria of psychiatric diagnoses, albeit in different levels of emphasis played to each, up to DSM-IV-Text Review (TR) (Bruijnzeel and Tandon, 2011).

Complicating the situation - Schizoaffective Disorder

At this point it is important to give a description of another disorder that has played a role in the shaping of the ideas of psychiatric disorders alongside schizophrenia and the manic depressive illnesses: schizoaffective disorder. The development of this throughout the 20th century highlights the difficulties and controversies that can arise from attempting to creative a reliable psychiatric nosological system, and is summarised in a fairly recent review and meta-analysis of previous studies in the literature by Cheniaux et al. (2008). It was first mentioned in 1933 by Kasanin in the American Journal of Psychiatry, and while it was described as being similar to schizophrenia in that it also suggests an association with additional affective symptoms, there have been many discussions about the definition since its inception. In fact, continuing into this century, there have been debates as to whether it should even exist or not (Maier, 2006). Even if its existence is taken as true, there is still much debate as to where it fits into the picture along with schizophrenia and manic depression including a range of ideas such as the following:

- It is a variant of schizophrenia but with additional affective symptoms
- In turn - it could be a variant of manic-depression but with schizophrenia-like symptoms
- It could actually be a mixture, or a sub-selection of symptoms from both disorders
- Schizoaffective disorder could indeed be a separate entity in itself entirely

The results from the meta-analysis suggested that schizoaffective disorder might either describe a heterogenous group consisting of both schizophrenia and manic-depression patients due to the difficulties of getting a clear demarcation between the two categories when getting samples, or indeed it could fall *between* the two disorders. It should be evident that these findings do not support the continuity of referring to these diseases or disorders as separate categories and should instead be viewed as a *continuum* or a *spectrum.* It most certainly does not support the notions behind the Kraepelinian dichotomy.

Categories to spectra - moving away from Kraepelinian Dichotomy

The modern opinion of schizophrenia and other psychiatric disorders is far more open to viewing them as more of a spectrum or a continuum instead of clear-cut categories that can be easily defined. Therefore it is useful to examine why the paradigm of the Kraepelinian dichotomy has survived so long. Both Craddock and Owen (2005) and Crow (1986) state that the dichotomy is actually a convenient model for psychiatrists as it offers them the chance to make clear diagnoses from complex observed symptoms and thereby show a level of clinical expertise in the field. Crow (1986) points out that in DSM-III (American Psychiatric Association, 1980) and other diagnostical manuals at the time, the emphasis is clearly on aiding the clinicians to clearly separate the symptoms and reliably diagnose patients with one of the two disorders. The belief in the dichotomy could also have been reinforced by the responses and outcomes to different treatments and medication for the separate disorders as well as different long-term prognoses, with schizophrenia seen as having more severe and permanent changes in personality (Kendell, 1987).

While the idea of a clear dichotomy between schizophrenia and bipolar disorder is associated with the workings and ideas of Kraepelin, it must be acknowledged that he himself began to doubt this nosological methodology towards the end of his research career. As Crow (1986) pointed out, one of the last contributions of Kraepelin in 1920 was to state that many psychiatrists would be able to recall many cases that they observed, where it would be impossible to clearly delineate them into either of these two categories and even said that the whole basis for the formulation of the idea could be incorrect. Craddock and Owen (2005) also remind us that Kraepelin, along with his peers, would not have had the possibilities to carry out any form of molecular studies at that time, thereby requiring that any conclusions had to be based on observable behaviours or endophenotypes of any disorders. The belief at the time was that the diseases could be differentially classified as they appeared to "breed true". The meaning of this term is that the first-degree relative of probands tended to display symptoms of the same disorder, without any cross-overs. Remembering that there was no access to the same advanced levels of molecular and genetic testing that we have today, it can be clearly seen why observations of this manner could have led people to believe that the prevalence of symptoms among relatives would have signalled the presence of different genetic aetiologies. However, as has already been noted, the observed symptoms in these patients are from highly complex psychiatric disorders. One must use caution when examining past observational scientific studies to ensure that enough care was taken to avoid any confirmation bias to occur, and that any diagnosis of relatives was taken while being blind to the diagnoses of the probands.

The first main study that began to challenge the dichotomy of these disorders, and even the notion that they breed true came from Kendell and Gourlay (1970). Discriminant analysis of observed symptoms amongst 146 bipolar and 146 schizophrenia patients was carried out to try to identify any common differences. The samples were taken from psychiatric populations based in both the United States and the United Kingdom. The analysis involved giving lower/negative scores to the more prototypical schizophrenia symptoms, and the higher/positive scores given to the bipolar symptoms; the hypothesis being that a plot of the distribution of the disorders would be bimodal. What was observed however was a trimodal distribution, and 9% of the subjects were misclassified (19 bipolar and 7 schizophrenia). The researchers initially believed that this showed evidence for a separate group of schizo-affective disorder. However, in an addendum in this study, a very similar analysis was carried out on a further 217 patients and on this occasion a unimodal distribution resembling a normal distribution was observed, albeit with a lower misclassification rate of 7%. The authors concluded that the clearer trimodal pattern was probably just observed by chance. These results led the researchers to conclude that there was not enough evidence to support the claim that the schizophrenic and bipolar disorders are separate entities.

The same research group decided to revisit this analysis nearly a decade later, carrying out a follow up study on the UK based samples of the previous study (Brockington et al., 1979). This consisted of 134 samples in total, of whom 11 died during the follow up period and 5 did not attend the final interview, resulting in a final sample size of 118. They also included an additional population of 106 patients with schizo-affective disorder, to see how these would compare. The results for the follow up population actually showed more of a bimodal pattern on the distribution of discriminant analysis scores, but the results from the schizo-affective population showed much more an ambiguous distribution, fairly unimodal with a prominent negative skew in the data. The researchers concluded that while there was some support of the dichotomy in their results, the addition of the schizo-affective patients did not show the same pattern and therefore does not support the notion that the entire population of patients with psychotic symptoms naturally fall into these two separate categories.

Studies carried out since these dates have provided results that continue to question the validity of the dichotomy. Angst et al. (1983) performed a clustering analysis using a method called Multi-Dimensional Scaling (MDS). This is a process that combines elements of factor analysis, Principal Component Analysis (PCA) and clustering analysis, by looking at the similarities between the value of input data points, examining how many dimensions are needed to represent them with minimal loss of information, and then looking for any clusters of similarities in these reduced dimensions. The inputs to their model were lists of symptoms from 269 patients of schizophrenia and schizo-affective or affective disorders. The results showed evidence of the symptoms of affective disorders being superimposed onto schizophrenia symptoms. They concluded that there could be an underlying base affective disorder, common to all affective disorders but did not state any belief in a unitary psychosis. Even more recent literature has questioned the dichotomy further. Craddock and Owen (2010) began to question the notion that these different disorders even do, in fact, breed true, and refer to a study carried out by Lichtenstein et al. (2009) that carried out a comprehensive analysis of over 9 million people from 2 million families in Sweden between 1973 and 2004, looking at first degree relatives of probands of both bipolar disorder and schizophrenia. The results showed that there was a large amount of cross-over in diagnoses (the relatives having diagnoses of the different disorder), and suggests that the comorbidity between the two is 63%. The researchers concluded that the two disorders partly share a similar genetic cause and that the findings challenge any claim that they should have separate diagnostic categories. A meta-analysis of 66 family studies carried out between the start of 1980 and the end of 2006 by Van Snellenberg and de Candia (2009) found that there was considerable cross-over and familial coaggregation of diagnoses of schizophrenia and bipolar disorder, which the authors conclude goes against the idea of a dichotomy and call for future efforts to be given towards developing a continuum-based diagnostic model.

All of the reviews and studies presented here have shown that more recent developments in the field of psychiatric genetics have heavily contested the idea of a dichotomy of schizophrenia and bipolar disorder. The next section will outline how this drive towards using spectrum, or continuum-based criteria for diagnoses has come to include other psychiatric disorders and the effects of interactions from environmental effects.

Diagnostic and disease classification issues

Following on from these challenges to the idea of the Kraepelinian dichotomy, there have been many articles continuing to challenge the idea that it is actually possible to categorise psychiatric disorders into separate nosological entities. With the advent of newer and more advanced scientific technologies and research methods in fields including neuroscience and genetics, there have been calls for a concerted effort to make use of findings from these studies in addition to conclusions drawn from epidemiological observations. There is also the added complication of taking into account the various environmental risk factors that have been implicated in the development of psychiatric disorders. This section will provide a brief overview of the attempts that have been made so far to address these issues.

While the concerns about the categorisation between schizophrenia and bipolar disorder has already been discussed, these have widened to incorporate other psychiatric disorders and there has also been an expanding acknowledgement of the role of environmental insults and interactions into the development of these conditions. There is a common theme that the existence of categories of symptoms used in subsequent versions of the DSM have primarily been of use to facilitate diagnoses of patients, but can have the negative effect of giving the impression that these categories actually exist outside of inter-rater reliability measures, and that the assumption of delineations between disorders can have a detrimental effect on the direction and outcome of scientific research (Kendell and Jablensky, 2003; Owen et al., 2016). This has caused concern for a number of decades. As a response to these criticisms, in the early 1970s, there were suggestions made by a group of clinicians, led by a man named John Feignher, from the Department of Psychiatry in the Washington School of Medicine, to develop a set of diagnostic classification criteria that differed from the methods used by the second version of the DSM, which was in use at the time (American Psychiatric Association, 1968), where it was stated that diagnosis should be based on the judgement of a committee of experts and consultants. Their suggestions focussed on five phases of the diagnostic criteria: a clinical description giving the symptoms of the patient, any possible laboratory results of psychological or physiological observations, any descriptions of how the symptoms delineated from other disorders, and crucially, two more that focussed more on longitudinal observations: a follow-up study to examine any development in symptoms that could challenge the validity of the original diagnosis, and a family study to attempt to identify any relatives who could be suffering from symptoms related to the diagnosed disorder (Feighner et al., 1972; Robins and Guze, 1970).

Perhaps more worrying criticism of the classification methods used by the DSM is a statement made by Stephen Hyman (2010), who was the director of the National Instiue of Mental Health (NIMH) in the USA from 1996-2001, where he stated that grant applications for research studies were always required to make references to the DSM criteria, even animal studies looking to develop new models of disease were judged on how well they approximated the DSM categories. He also stated that the US Federal Drug Administration (FDA) interpreted the DSM categories as the scientific consensus when approving new medications and treatments. This can be of concern when the boundaries of these criteria are not clear, and there was so much comorbidity between criteria in the DSM-IV (Kessler et al., 2005). This is indeed a concerning revelation as it shows how a potentially erroneous classification system, albeit designed for pragmatic reasons, can result in down-stream detrimental influences that can impede future research, as well as the understanding and treatment of psychiatric disorders.

Cardno and Owen (2014) have reported that new genetics research has supported the findings made by the family and relative-based studies mentioned earlier showing that cross-over between the disorders of schizophrenia and bipolar can be seen in the first order relatives of probands (Lichtenstein et al., 2009; Van Snellenberg and de Candia, 2009). They highlighted a large study carried out by the Psychiatric Genetics Consortium (PGC) Cross Disorder Group looking at the genetic similarities between five different conditions: Autism Spectrum Disorder (ASD), Attention Deficit Hyperactivity Disorder (ADHD), Major Depressive Disorder (MDD), bipolar disorder and schizophrenia (PGC-Cross Disorder Group, 2013). The results of this study showed that a series of specific genetic variants are associated with all of the disorders considered, with most of these variants seen in regions of the genome pertaining to calcium-channel activity, and the researchers also suggested moving away from viewing these disorders as descriptive categories.

In the lead up to the release of the 5th edition of the DSM (American Psychiatric Association, 2013), there were efforts to ensure that more of a “bottom-up” approach could be developed, taking into account findings from studies making use of developing technologies and methods in the fields of neuroscience and genetics (Hyman, 2007, 2010). This is contrasted with the more traditional methods of diagnosing patients based on epidemiological observations and judgement. Since it was released in 2013, there have been some suggestions that it has moved towards achieving this goal, by removing the sub-groups of schizophrenia, basing its diagnostic aids in neuroscience, neurology and genetics research and relying less on observational epidemiological methods (Rodríguez-Testal et al., 2014). However, there still remains the concern that there are too many categories of operational criteria (Owen et al., 2016).

With these new considerations for the genetics and neurobiology becoming more prevalent, it is also very important to investigate how they can interact with any effects from the environment. It has long been acknowledged that there are certain variables that influence the rates of schizophrenia in the population including gender, socio-economic status, cannabis use, migrant status and urban living environments (McGrath et al., 2008; Tandon et al., 2008a). The increased number of resources that have allowed for new research in genetics have provided the means to examine how the environmental effects can bring about epigenetic variation (Cariaga-Martinez et al., 2016). The term "epigenetics" is used to describe changes that are made to the DeoxyriboNucleic Acid (DNA) such as methylation, and alteration to histone proteins that, while not changing the genome sequence of an individual, has effects on downstream gene expression and the examination of these changes can help to explain diseases that cannot be identified by looking at the DNA sequence alone (Urdinguio et al., 2009).

These epigenetic interactions with the environment have been of particular interest in the study of schizophrenia and the developments of the research into this field has been recently reviewed by Owen et al. (2016). While a complete and detailed review would be too long to include here, a few examples are given. The authors talk about the "neurodevelopmental hypothesis" (Fatemi and Folsom, 2009), which describes how early-life environmental effects can have an effect on the development of neural systems in later years. Some earlier research from the Nordic countries brought this to attention; Machón et al. (1983) carried out a study in Denmark that identified a higher rate of the disorder amongst winter births, and provided a possible connection to maternal infection of the influenza virus during pregnancy. This work was expanded by Mednick et al. (1988) who looked at males and females born during the Type A2 Influenza outbreak in Finland in 1957 and found that there was an increased risk seen when the infection occurred during the second trimester of pregnancy, and the observations were found to be independent of the different psychiatric clinics across Finland where these people were being treated. Articles following on from this study talk about how an environmental insult during this critical stage of gestation and development can have more severe downstream effects: the firing patterns of neural networks, specifically those related to glutamate receptors, can result in excessive synaptic pruning at another critical stage in development: adolescence. This is turn affects how an individual will react and adapt to different behavioural and social encounters and developments, thus resulting in further impaired neuronal development and possible psychotic symptoms later in life; the so-called "three-hit hypothesis" (Keshavan and Hogarty, 1999; Keshavan, 1999). Such is the focus on the possible effects of neuronal damage during these critical periods that suggestions have been made that prophylactic treatment could be given to help prevent damage to neurons caused by oxidative-stress (Sawa and Sedlak, 2016), and results from an animal based study have shown initial success when this treatment is applied to rodent models (Cabungcal et al., 2014). In order to address the important roles that the interactions between genetics and the environment present, new methods are being considered for use in studies, such as the idea of developing a “gene-environment wide interaction study”; essentially looking across the whole genomes of patients and healthy controls for evidence of genetic variants interacting with exposure to environmental factors (Thomas, 2010).

This section has highlighted how modern research and diagnosis is attempting to move away from purely observational methods towards others based on findings from genetic, neuro-biology and neuroscience research. As the techniques in this thesis make use of datasets from genetic studies, a summary of how these have developed over the past few decades is provided in the next section.

1.1.2. Genetics of Schizophrenia as a complex trait. A review of methods

This section will provide a brief summary of how the research, technology, and methods used to try and gain insight into the aetiology of schizophrenia have developed over the past decades; explaining some of the main findings that were made, as well as highlighting the weaknesses of each method. Towards the end of this chapter, some of the caveats that still remain will be highlighted, and suggest how the main techniques used in this thesis, machine learning algorithms, can hope to address some of these. A more detailed description of the algorithms and methods used in this thesis will be provided in the next chapter, as well as giving examples of how these algorithms are being adopted in the field of genetic research.

Twin Studies and the heritability of Schizophrenia

Before talking about the different methods and technologies used in the study of psychiatric genetics, it is worthwhile to provide a brief overview of the studies that have been carried out to look into the heritability of schizophrenia. The term “heritability” is used to describe the estimate of how of the variance of a disorder seen in the population can be explained by genetic factors as opposed to effects caused by any environmental interactions and insults. A common method of examining this is by performing twin-based studies, looking for the different concordance rates seen in Monozygotic (MZ) twins, those who have developed from the splitting of a fertilised ovum and are therefore *identical* and share 100% of DNA; and Dizygotic (DZ) twins, those developed from two separate fertilised ova that happened to be released at the same time, also known as *fraternal* twins, who share only 50% of DNA. The reason for using DZ twins and not just siblings of different ages is that it allows for a more plausible assumption that the environment in which development takes place is more shared, instead of events that could have happened at different ages for non-twin siblings.

This paradigm of the twin study has normally been credited to a behavioural geneticist called Francis Galton who, in 1875, wrote an article titled "The History of Twins, as a Criterion of the Relative Powers of Nature and Nurture". However, this study did not actually make any comparisons between MZ and DZ twins. This first occurred in the early 20th Century, and was mentioned as a study in two separate sources in the same year, 1924, by Curtis Merriman and Hermann Siemans (Rende et al., 1990).

Since its inception, the twin study has been used on numerous occasions to assess the rates of heritability seen in a vast range of traits and disorders. This includes schizophrenia, but the topic has been made more inscrutable by the complications in diagnostic procedure and outcome. In the latter half of the 20th Century, McGuffin et al. (1984) used the twin study to identify which diagnostic methods were best for successfully identifying a disorder that had a higher heritable component and seen more in first degree relatives of probands; developed from work carried out by Gottesman and Shields (1972). The results showed that it was the method developed by Feighner et al. (1972) that showed the superior concordance ratios for heritability seen between MZ and DZ twins.

Due to the high number of twin studies carried out in the field of schizophrenia, there have also been several meta-analyses performed on these studies. These have often yielded estimates of heritability in the region of 80% (Cardno and Gottesman, 2000; Sullivan et al., 2003), but Cardno and Owen (2014) also drew attention to family-based studies carried out in Scandinavia, which have reported lower estimates of around 65%, the aforementioned Swedish study by Lichtenstein et al. (2009) and a Danish study by Wray and Gottesman (2012). The difference in these rates could be due to the difficulties in diagnostics, as highlighted by McGuffin et al. (1984). Another weakness of many of these studies is that they make the assumption that the risk effects of heritable variants are additive in nature. This is a very practical assumption to make as Polderman et al. (2015) pointed out that considering non-linear interactions would result in a massive increase in the number of statistical tests made, and the concomitant problem of correcting for multiple comparisons. This is, in fact, a common theme that will be mentioned in this thesis, as the machine learning methods described in the next chapter were designed to circumvent this problem and find a computationally effective manner of examining the presence of genetic interactions.

All these studies have shown that there is a considerable element of heritability related to the onset of schizophrenia, which has resulted in many years worth of studies looking for the possible genetic markers that could help to identify those at risk of developing this disease. The next sections will look at how this research has developed.

Technologies and methods used to identify genetic variation

Over the past few decades, the technologies for analysing variation across the genomes of organisms have increased substantially. A major development in this field has been from the Human Genome Project (HGP), which is an enormous consortium project that began in 1988, comprised of collaborative groups from all over the world, with the aim of successfully obtaining the DNA sequence across the human genome (Lander et al., 2001; Olson, 1993). This project was deemed to have achieved their goals in 2004 (Schmutz et al., 2004). While the focus of this project was to gain the means of sequencing the human genome, another project that was launched towards the start of the century looked to use this information to provide insight into how different parts of the genome showed variation across the population, especially for different ethnic groups around the world. The name of this project was the International HapMap project, and its main aim was to identify the common genetic variants and to make these publicly available to aid future research (Gibbs et al., 2003). It focussed on looking for sources of common variation: genetic variants seen at a rate of 5% or more in the population, and looked at variants called Single Nucleotide Polymorphisms (SNPs). This term describes points of variation in the genome: places where one of the four nucleotide bases {A: Adenine, G: Guanine, T: Thymine, C: Cytosine} is more commonly seen, but for a minority of people, this is switched to another. For the vast majority of these points of variation, the switch is made to only one of the other bases, a so-called "bi-allelic" variant, although "tri-allelic" variants have also been identified, but are far rarer (Casci, 2010; Hodgkinson and Eyre-Walker, 2010).

The information about these SNPs is obtained by a process called "genotyping", and is carried out on chips containing primers to look for these points of variation across the DNA sequence. These have become very popular as advances in technology have allowed for data to be collected on very large sample sizes at a relatively low cost. Further information about the processes involved in these technologies can be read on the Illumina website[1]. At this point it should be highlighted that it is these bi-allelic SNPs that make up the variants in all of the datasets used throughout this thesis.

A statement on the website of the Broad Institute has claimed that the use of genotyping

[1] www.illumina.com/clinical/illumina_clinical_laboratory/genomics-101/reading-genome.html

techniques, and looking for the relatively common variants, is now recognised as one of the weaknesses of the HapMap project[1], and a subsequent project called the "1000 Genomes Project" was carried out that made use of a variety of more advanced techniques, including whole-genome sequencing and dense microarray genotyping, and gained information about over 88 million genetic variants at a lower rate of 1% for 2,504 samples across a variety of different ancestries, and was announced as complete in 2015 (1000 Genomes Project Consortium, 2012). This is the latest development in the technologies and resources that have enabled the advances in genetic research methods to be achieved.

All of these technological developments have allowed researchers to gain insight into the human genome with an ever increasingly fine resolution. The next sections will give an explanation of techniques into the study of genomics that have been developed in turn with these advances in technology.

Linkage Studies

The main goal in linkage studies is to carry out analyses on "genetic pedigrees", which is terminology for a group of related people organised into a family tree structure. Ideally, these pedigrees will contain a mixture of healthy individuals, and those who are patients of the disorder or disease of interest. The aim is to collect information from across the genome of these individuals, and try to look for patterns of variation or mutation that show differential tendencies to be transmitted during reproduction to the sufferers and the controls.

This technique was shown to be very successful in finding rare alleles that wholly cause diseases like Huntington's Disease, a disorder resulting from an increase in the number of repeated "CAG" trinucleotide motifs at the $3'$ end of the gene encoding the Huntingtin protein located on Chromosome 4 (Walker, 2007). However, it is not completely limited to diseases resulting from only one variant, and has been developed to help bridge the gap towards research into complex disorders resulting from a mixture of different variants by looking for multi-locus markers (Badano and Katsanis, 2002).

Despite these developments however, linkage studies have had problems with identifying the possible causes of non-Mendelian diseases, such as schizophrenia, bipolar disorder and even diabetes (Risch et al., 1996). Cariaga-Martinez et al. (2016) pointed out that this is most likely due to the low levels of *penetrance* (the effects of the variants presenting in

[1] `www.broadinstitute.org/science/projects/1000-genomes`

the phenotype) seen in the variants pertaining to these disorders, and that this could be a possible reason for why many of the findings in these studies have failed to replicate. A good example of this is a recent meta-analysis carried out by Walters et al. (2014), which looked at seven linkage studies into schizophrenia that had been funded by the NIMH. In this study, they reviewed the data preparation and analysis techniques that were used in the original studies and found that the results were not robust to modern methods of data preparation and therefore yielded different outcomes. These show that, while linkage studies have been an invaluable tool to identify certain genetic disorders, further developments were needed to address the problems related to complex diseases resulting from low penetrance variants.

Candidate Gene Analyses

The next method that was developed following on from genetic linkage studies was the candidate gene study. At this stage, before the progress made by the Human Genome, HapMap and 1000 Genomes projects, it was very expensive to genotype large sections of the genome. So instead, researchers focussed on areas located within and around genes that already had an *a priori* hypothesis to be involved in the risk of developing schizophrenia; this could come from findings in linkage studies or the known targets of pharmacological agents (Farrell et al., 2015). One of the crucial differences in this method, compared with linkage studies, is that it does not require the collection and analysis of data from related genetic pedigrees; therefore it is possible to obtain information from larger sample sizes. Given enough statistical power, this would be able to identify significant differences in rates seen between patients and controls for common variants with a lower penetrance instead of focussing on the rare mutations of large effect that the linkage studies are more suited for.

This approach proved to be successful at identifying variants associated with alcohol misuse and substance dependence (Reich et al., 1999), as well as repeated examples of success in Alzheimer's disease research, particularly pertaining to the *APOE* gene (Levy-Lahad et al., 1995; Li et al., 2008; Zou et al., 2014). However, attempts for Schizophrenia did not prove to be so fruitful; there were often poor replication rates and general criticisms of spurious claims of genetic associations with the disorder. Sullivan (2007) stated that one major flaw of these studies is that there can be too much of a role of previous opinions that could have developed more from hunches than solid evidence, which could severely bias any results, but they did actually point out that these failings did, in fact, highlight methodological weaknesses that allowed for the development of improved techniques in the field.

Despite the ability to use larger sample sizes, due to not being limited to examining sets of related individuals, Farrell et al. (2015) highlighted the fact that, ironically, it could have been a lack of suitably large sample sizes that resulted in the lack of success of these methods. It was not known at the time quite how small the contributions from the different genetic variants were to schizophrenia, which has subsequently been discovered. They provided an example calculated from information given in a medical statistics article by Gauderman (2002) stating that, even with 1,000 cases and 1,000 controls, a study would only have 0.03% power to detect a genotypic relative risk of 1.15, a value which has shown to be incredibly large for variants associated with schizophrenia. They also concluded their article by stating that, while no replicable and robust findings were found from candidate gene studies, they did lay the foundations for subsequent developments in the field. The problem of getting enough statistical power to detect variants that had very low penetrance and effect inspired the development of technique described in the next section.

Genome-Wide Association Studies

The next stage in the development of genetic studies was the Genome-Wide Association Study (GWAS). This technique can really be thought of as a natural progression that developed conjointly with the advances made in getting more information from the human genome at lower costs, and the work already performed in the linkage and candidate gene studies. With the increased possibilities of genotyping more of the variants across the genome, the idea behind a GWAS is to scan for numerous variants and to examine which of these tend to show a greater frequency in cases versus the controls. The first successful application of this technique was carried out by Klein et al. in 2005 in the study of age related macular degeneration. This study actually made use of a very small sample size (96 cases and 50 controls), especially in comparison to many of the studies that have been carried out in schizophrenia research.

As has been previously mentioned, the failure of previous techniques is most probably due to the risk of schizophrenia coming from several different variants across the genome; all individually making a very small contribution each, but as a whole, providing a certain amount of risk of developing the disorder. There are some notable exceptions, which will be discussed shortly, where certain individuals have developed schizophrenia from larger mutations in the genome that are far more penetrant and deliver more of an effect, but these individuals are the minority of people who present with symptoms of the disorder. The terminology that has been used to describe this collection of small effects of variants is to describe a disorder as being *polygenic* in nature. The first mention of this term with reference to schizophrenia was made by Gottesman and Shields in 1967 in a paper titled “A Polygenic Theory of Schizophrenia”. In this article, they explained that to consider schizophrenia as a disorder related to crossing a threshold of genetic risk, which can manifest in the development of different phenotypes related to the disorder in the presence of interactions with the environment during development, is the most probable scenario to explain the patterns in heritability seen. With the advent in the ability to take measurements of genetic variability from across the whole genome, it is now possible to effectively scan for contributions to polygenic risk, and this will continue to improve with the increased usage of whole genome sequencing that is taking place. Any analysis of polygenic risk involves making a *polygenic score* for each individual, which is a weighted average of the risk posed by the common variants across the genome. Details of how this score is made are provided in chapter 2.

At this stage, one of the main susceptibilities of a GWAS must be mentioned: a tendency for people to inherit different blocks of genome from their parents, caused by non-random mating patterns most commonly as a result of *population stratification.* This is caused due to differential patterns during the chromosome cross-over break points that are created during gametic development during meiosis. The break points for these cross-overs do not occur at random places in the genome, and this means that if an individual inherits a section of DNA from one parent, then they will also inherit the DNA regions flanking this point, unless the point of interest is alongside a break point. These blocks of DNA that are inherited are referred to as *haplotypes*, and the pattern of their distribution can vary between different ethnic populations (Daly et al., 2001; Reich et al., 2001; Wang et al., 2002). If a large enough sample is collected from people with the same ethnic heritage, then estimations of how likely the different measured variants collected (commonly referred to as *markers*) are to be inherited together in this manner can be calculated. The term that described the connectedness of inheritance between variants is called Linkage Disequilibrium (LD) (Slatkin, 2008). This is, perhaps, not the most intuitive of terms to use the phrase *disequilibrium* to describe how markers are *connected* to each other, but it is actually an off-shoot of a statistical phrase that was used to explain the opposite situation. This is concisely described by Lewontin and Kojima (1960) who state that the idea of equilibrium in linkage occurs when the joint probability of seeing the presence of two independent alleles is *equal* to the product of the probability of these alleles being seen individually, an idea that they state was first developed by Geiringer (1944). In short, this means that that, in this equilibrium situation, the chance of seeing one allele at one loci is *independent* of seeing the other one; no information about one allele can be obtained from the other. If the occurrence of the alleles at the loci is *not* independent, then this joint probability calculation cannot be performed in this way, so the presence of one allele provides the observer with some information (either small or complete) as to the presence of the other allele, and they are therefore in linkage *dis*equilibrium. The metric used to quantify the levels of LD between SNPs is the r^2 metric, and ranges from 0 to 1 to express either complete linkage equilibrium, to variants that always appear together and are therefore in total LD with each other.

This idea of being able to gain information about the probability of seeing one allele by observing its neighbours that are in high LD with it is, in fact, critical to how a GWAS functions. The measurements for the analysis are taken from genotyping markers are locations spread across the genome. If there is no prior hypothesis as to which variant could be causal towards developing a disorder, it is highly unlikely that the causal variant will be selected as the one to be measured. However, by scanning across the genome with enough resolution, it is very likely that measurements will be taken from markers that are in LD with any causal variant, and if any of these show an association to be seen more in patients than controls, then this gives an indication as to the *region* of the genome where that causal variant could be located. This shows why care must be taken to control for population stratification when making comparisons between cases and controls as the different markers could be representing different sections of haplotype blocks, which may or may not be in LD with causal variants, so this could result in erroneous results when looking for associations with disease (Cardon and Palmer, 2003).

Another issue with a GWAS is that testing so many variants for their association with a disorder results in millions of statistical comparisons. This means that a stringent procedure of *multiple comparison testing* has to be performed in order to prevent the reporting of erroneous findings. There is no easy way to do this, but the established method in the field is to perform a Bonferroni correction with a factor of 1 million. This is due to the estimation that the human genome contains 1 million blocks of Linkage Disequilibrium, and therefore this number of independent tests are being carried out in a GWAS, although it should be pointed out that this estimation is relevant for those of caucasian European descent, and can cause problems when other ethnic populations are used in analysis (Johnson et al., 2010; Risch et al., 1996). This brings the GWAS significance level to 5×10^{-8} for an α level of 0.05. This is an incredibly strict level, especially for detecting the small effects from the individual variants, and therefore requires a large amount of statistical power, which can only be achieved through the use of greater sample sizes, which can only be recruited through the coorperation of labs across the world.

Recent Large Scale Consortia Studies

A timeline of the introduction of large scale consortia studies has been provided by Kavanagh et al. in 2015. The first major project was the Wellcome Trust Case/Control

Consortium (WTCCC) (Burton et al., 2007), which looked into seven common diseases, including bipolar disorder and schizophrenia, using 2,000 samples for each disease and a shared set of 3,000 controls. One of the main conclusions from this study was that the effect sizes of the contributions from variants was very low, with odds-ratio measures less than 1.2. This confirmed that large sample sizes were needed to gain the statistical power to identify these, and Kavanagh et al. (2015) mentioned that the subsequent schizophrenia GWAS studies only identified five loci that were significantly associated (O'Donovan et al., 2008; Shi et al., 2009; Stefansson et al., 2009). The International Schizophrenia Consortium (ISC) study used 3,322 cases and 3,587 controls to identify that schizophrenia was highly polgenic in nature (Purcell et al., 2009) and showed that a signal from across the genome had to be considered. The first breakthroughs began to emerge after the establishment of the Psychiatric Genetics Consortium (PGC), and the first study by this group identified 13 novel schizophrenia risk loci (Ripke et al., 2013) using a sample size of more than 21,000 cases and 38,000 as well as 581 parent/offspring trios. However, arguably the most prolific study in this field was the second study by the PGC, which used an even larger sample size of 34,241 cases and 45,604 controls in addition to 1,235 trios. This analysis found 108 independent loci that were significantly associated at the GWAS corrected level, of which 83 were novel findings. A Manhattan plot of these can be found in figure 1 of the article on page 422 (Ripke et al, 2014).

Despite these novel findings, there is still concern that the identified loci do not explain a sufficient amount of the genetic variation seen in schizophrenia. Corvin and Sullivan (2016), pointed out that the significant loci only explain about 5%, and various collections of sub-threshold SNPs (those that were associated, but not at the stringent GWAS significance level) explain around 25%. The authors also mentioned that even larger samples could be required to gain more power and information, and that the third study from the PGC will contain information on greater than 100,000 cases, of which they expect to have 65,000 completed by the end of 2016.

This idea of "missing heritability" has been mentioned a great deal in the literature, and there have been various suggestions about how best to approach this issue. Polderman et al. (2015) carried out a meta-analysis of twin studies from 2,748 publications and examined the differences between MZ and DZ twins when looking at the *additive* effects of the genetic variations for a variety of different traits including appearance and risk of disease and disorders. They found that a third of the traits they examined did not show an additive difference in variation when comparing between the two different twin types, and gave an example of another psychiatric trait of recurrent depressive disorder. This is not just an issue for psychiatric disorders either, Zuk et al. (2012) also came to a similar conclusion when researching Crohn's disease, and estimated that up to 80% of the variation seen in the population could be due to non-additive effects. As mentioned ealier, when stating that these additive effects are not adequate, Polderman et al. (2015) raised the issue of considering the interactions that could occur between loci, but stressed that creating all of these different combinations of genetic variants would make any analysis unfeasible. As will be seen later, this is one of the topics that machine learning algorithms can be utilised to solve.

In addition to creating studies with ever larger sample sizes, there have been other methods used to try and uncover the genetic aetiology of schizophrenia, and the next two sections will discuss these. The first deals with mutations that are more rarely seen in the cases, but carry far more risk than the common variants, and the second discusses ways in which the information from the individual SNPs can be collected together into functionally related groups, referred to as *gene sets* or *pathways*.

Common variants of small effect vs. Rare mutations of large effect

At the start of this century, it was estimated that the genetic differences between two individuals amounted to approximately 0.1% of the human genome, and most of these were put down to the Single Nucleotide Polymorphisms. However, in 2004 there were two seminal studies published by Iafrate et al. and Sebat et al. that stated that there were also changes in the structure of DNA that have been given the name Copy Number Variants (CNVs). These changes are larger than the single base point variants of SNPs, with most less than 500 Kilo-Bases (KBs); but as it is estimated that there could be around 1,000 of these per individual, they can account for a difference of 4 million base pairs, and therefore deliver an additional 0.1% of genetic variation (Malhotra and Sebat, 2012). There can be many reasons for the occurrence of these CNVs in the genome and this can result in large sections of DNA being either deleted or replicated. A detailed description of the different causal mechanisms that can occur is provided in the review by Malhotra and Sebat (2012).

Since their discovery, many studies have found that CNVs can be implicated in a range of psychiatric diseases including schizophrenia and bipolar disorder (Green et al., 2015; Kirov et al., 2012), and many of these have been found to be *de novo* in nature, meaning that they are mutations that are not inherited as they are not seen in either of the parents. This finding is related to the observation that the levels of fecundity (particularly in males) is drastically reduced in sufferers of schizophrenia (Bundy et al., 2011). Many CNVs have also been shown to be enriched in the gene sets and pathways related to synaptic proteins and function (both excitatory and inhibitory) (Fromer et al., 2014; Kirov et al., 2012;

Pocklington et al., 2015), and the findings from the study by Pocklington et al. (2015) in particular provided the inspiration for the work carried out in chapter 5.

While the larger mutations have been shown to deliver a higher level of risk to those presenting with them in their genomes, they do not feature in any of the datasets that have been in the experimental chapters of this thesis. The reason for this will become clear after reading the material in chapter 2, where it is explained that the algorithms are essentially performing *pattern-matching* tasks, and are therefore looking for commonalities of genetic variation that can help differentiate between the cases and the controls. If these algorithms are presented with a mutation that only occurs in a very small fraction of all the cases, this is likely to be interpreted as an anomaly and is therefore of little or no use in these situations. It is for this reason that the primary aim of using the machine learning methods is to try and detect the presence of interactions between the SNPs, the common variants that each confer a low risk of developing the disorder.

Trying to improve on predictive power by using Gene Set analyses

To provide a brief summary on some of the critical developments so far: there have been a number of commonly seen risk SNPs that have been associated with delivering a higher risk of developing psychiatric disorders like schizophrenia. The information from these can be combined to make a polygenic score that can then be entered into predictive models to estimate the level of risk that is delivered from this ensemble of variants. In addition, regions of the genome that fall within gene regions that are highly expressed in the brain, and produce the proteins relevant to synaptic structure, have been shown to be enriched for the larger CNVs. However, the information from both of these approaches has not been sufficient to explain the levels of variation seen in disease rates in the population, leading to the question of how to best find this missing heritability.

One approach to try and isolate important genetic regions, and to provide a means of increasing statistical power, is to collect the information of genetic variants into functionally related groups of genes, which will be referred to from now as "gene sets". The main idea behind this method is that grouping the information in this manner can help explain how the increased loading from SNPs can be raising the risk of developing disorders. When looking at the vast numbers of the SNPs, it can be incredibly difficult to identify the processes that could be occurring with collective risk, and grouping them together based on *a priori* hypotheses can aid in this. It is not just in the field of psychiatric studies that this approach has been used: Subramanian et al. (2005) showed that looking at these collections of gene sets in a study of lung cancer proved to be successful at identifying association, whereas looking at the individual variants or even genes did not.

While this can be seen as an intuitive and natural progression from identifying groups of risk-associated SNPs in the genome, there is a range of complications that can have an affect on both the analysis and the interpretation of any results. As was eluded to earlier, this method is very reliant on forming hypotheses about the functions of different genes, and how these are related to each other, based on previous research on *gene annotation*. There are common databases that researchers can use to access the information on this prior research including the Kyoto Encyclopedia of Genes and Genomes (KEGG) (Kanehisa et al., 2008), the Gene Ontology (GO) database (Ashburner et al., 2000) and the Mouse Genome Informatics (MGI) database (Blake et al., 2013). The material presented in chapter 5 makes use of information from prior studies using the information from the MGI.

Wang et al. (2011) pointed out that further difficulties can arise from the different methodological issues that stem from this type of analysis. One key aspect to consider is whether to only include the protein coding regions of the genes, or to also include the various Un-Translated Regions (UTRs) into the $5'$ and $3'$ flanking DNA upsteam and downstream of the genes respectively; and if these are to be included, then the size of the window regions can play a role when considering how many different SNPs come under the different gene set groupings. One particular study by Jia et al. (2011) stated that gene set analyses must be approached with caution, as they claimed that their study found different results to a study by O'Dushlaine et al. (2011) that used the same dataset and concluded that this could have been down to different analytical methods used.

Other methodological issues are whether to perform *competitive* or *self-contained* analysis when looking at the information in the gene sets. These are two fundamentally different statistical approaches to gene set analysis. The competitive method works by comparing the levels of association in the gene sets and examining if these are greater than the levels seen in genes that do not feature in the set. Self-contained methods look to see whether any significant levels of association are in any of the genes featuring in each set (Wang et al., 2011). Additional caveats that are present can also affect the interpretation of any results; one major factor can be the size of the genes in the different sets: any set that contains genes that are larger than average could very possibly contain more significant levels of association purely because they are capturing a larger region of the genome than sets containing smaller genes. If this is not corrected for then spurious conclusions can be made.

There have been a number of different methods developed to perform gene set analysis and take the aforementioned difficulties into account. One such example is Gene Set Enrichment Analysis (GSEA)[1], which performs a sample permutation procedure to test for gene set significance, and factors in the gene set sizes in the data preprocessing procedures (Wang et al., 2007). Another method is Association LIst Go AnnoTatOR (ALIGATOR), developed by Holmans et al. (2009). A full flowchart of the procedure can be seen in figure 1 of the article on page 15, but in brief: it makes use of a series of random sampling of SNPs to create replicate lists of genes and then collects bootstrapped samples of these gene-lists to obtain an empirical p-value of the original data. The random sampling of SNPs aids in controlling for the different gene sizes.

A very recent development in this field is a piece of software called MAGMA (de Leeuw et al., 2015), and is put to use in the material presented in chapter 5. A description of the analysis procedure is given in the methods section of that chapter, but to summarise here, it makes use of a multiple logistic regression routine to examine association at the level of the individual genes, and then collates this information to the levels of the different gene sets. The sizes of the genes are entered explicitly as covariates into the model, thereby attempting to control for any confounding effects that can distort interpretations. Of note, the authors state that the competitive method is chosen as default, as this looks for statistically significant association in the sets instead of looking for the presence of any statistically associated genes within the sets as is done in the self-contained method, which they claim leads to more difficult interpretation.

As can be seen in chapter 5, both the machine learning methods and the MAGMA analysis identify a set of genes related to targets of the Fragile-X Mental Retardation Protein (FMRP), a set that has been implicated in synaptic processes. The findings are discussed in both that chapter and the main discussion chapter.

Machine Learning in Psychiatric Research

So far, all of the methods mentioned in the study of schizophrenia genetics have been about identifying associative signals from the additive effects of the mutations and variants; in statistical terms, this is looking at the main effects that are making any contributions. This is a practical approach to a problem, because looking at the effects of many thousands of SNPs would be rendered unfeasible if all of the different combinations of the interactions between these were to be considered as well, as the dimensionality would be too great, and nothing would survive statistical correction for multiple comparison testing (Polderman et al., 2015). This large scale increase in dimensionality that can occur from looking at

[1] User guide: `http://software.broadinstitute.org/gsea/doc/GSEAUserGuideFrame.html`

all the possible combinations between the features has been referred to as the *Curse of Dimensionality* (Bishop, 2006; Hastie et al., 2001).

A good example of how challenging this can be using traditional methods is a study by Hemani et al. (2014) where the pairwise interactions of SNPs were explicitly used in the analysis. This work was carried out to examine if considering these two-way interactions would explain any additional association seen in the data. The authors made clear that this would be an incredibly difficult task with SNPs that deliver very small effect sizes (as is seen in psychiatric disorders like schizophrenia), so instead, they focussed on looking for variants that were associated with differing levels of gene expression patterns, as previous research in this field has shown the effect sizes to be much larger (Powell et al., 2013). Even with a phenotype that has larger effect sizes than those seen in schizophrenia, the study still had to make use of powerful computational software and equipment that involved processing the data on Graphical Processing Units (GPUs) instead of Central Processing Units (CPUs) (Hemani et al., 2011). These GPUs are capable of carrying out large scale parallel processing of mathematical computations, and are most commonly used for the types of matrix computations required for modern computer game graphics but have recently been exploited for their powers in data analysis. The results of this study identified 501 significant pairwise combinations of SNPs that were associated with the expression levels of 238 genes.

This study highlighted some very important findings and observations that provide a clear basis for the justification of using machine learning techniques in this thesis. Their results showed that the interactions between SNPs can reveal significant association with a phenotype, so while an assumption of additive main effects allows for simpler analysis to be performed, it will never explain all of the variance if this assumption is not met. But the paper also highlighted the extreme difficulties that arise when looking for non-linear effects as they had to use variants with larger effect sizes and had to resort to using complex computational methods when looking for explicit interactions. The situation in schizophrenia is even more complicated, and other methods have to be employed that can consider non-linear effects without the need for the data on the explicit combinations of interactions. In a review of using machine learning methods to detect gene-gene interactions Cordell (2009) mentioned that there are methods in place to look for pairwise interactions, but these can be incredibly time consuming and calculated at that time that searching for these pairwise effects from millions of variants across tens of thousands of samples could take months of computational time on parallel clusters, and that any comprehensive search for higher order interactions is completely unfeasible. It is for this very purpose of looking for evidence of interactions without needing to have them explicitly entered into the models that machine learning algorithms can be used effectively.

A whole range of different algorithms has already been put into use when looking for non-linear effects in genetic studies. A review of the more common methods, describing their strengths and weaknesses can be read in Koo et al. (2013). The next chapter is dedicated to the description of how some of these algorithms work, and how they have been used in the fields of neuroscience and psychiatric genetics.

Focussing on Treatment Resistant Schizophrenia

The material in this chapter has given a review of many of the difficulties that have arisen during the debates about the classification and diagnosis of schizophrenia, and how this can have an effect on genetics studies. For this reason, all of the experiments in this thesis made use of a dataset containing samples that are believed to suffer from treatment-resistant schizophrenia. More details of these datasets are provided in chapter 3, but to summarise here, all of the cases were recruited from people receiving the medication *Clozapine* from the pharmaceutical company, Novartis (Hamshere et al., 2013). This medication is used for people who have not shown any improvement after two previous treatment programmes with different anti-psychotics before. These individuals often suffer from some of the most extreme and debilitating symptoms of schizophrenia, and therefore the assumption is made that the people presenting with this phenotype could possible have more of a cohesive genetic aetiology than other samples with less severe forms of the disease.

2. Machine learning algorithms and descriptions

2.1. The advent of Machine Learning

Ever since the 1950s, there has been a drive to research how to make use of developments in computational power and capabilities to create ever more *intelligent* systems, designed by humans. A famous example is that of Alan Turing who developed the concept of the *Turing Test*, whereby the responses of a computer to human interaction could be believed to have come from human responses (Flasiński, 2016; Turing, 1950). In the late 1950s, another pioneer of machine learning began to develop algorithms by which computers could actually learn from experience to improve at the game of draughts (Samuel, 1959). In this paper he pointed out how the main aim of machine learning is that computers can learn from experience and improve performance on a task, without being explicitly programmed to do so by humans. An often quoted phrase that is used to describe the fundamental ideas behind a computer being able to learn from experience was made by Mitchell (1997): "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E" (p. 2).

Other key developments in the field have sought to create programs and algorithms that mimic the parallel information processing that takes place within human brains via the use of computational and artificial neural networks. Probably the most famous of these developments was a collaboration between the neuropsychiatrist Warren McCulloch and mathematician named Walter Pitts to develop a system, using logical calculus, to model the "on/off" states of networks of firing neurons to transmit information (McCulloch and Pitts, 1943), and has since been recognised as a key development in the fields of cognitive neuroscience, artificial intelligence and cybernetics (Abraham, 2002).

With the modern developments of computational processing power, the use and interest in these methods has increased dramatically. A few examples of key developments can include the *recommender system* often used in online purchasing and marketing for targeted advertising based on previous purchases and reviews. A seminal event in this development was *The Netflix Prize*, when the online streaming company, Netflix, offered a prize of $1 million for the development of a system that could accurately recommend titles to people based on a dataset of 100 million ratings from 480 thousand users for nearly 18 thousand titles. Another development that has featured in the popular press has been the use of deep-learning neural networks to beat a world champion at the Chinese game of "Go" (Silver et al., 2016). This marks a key breakthrough from the early days of artificial intelligence, when Arthur Samuels was developing the aforementioned program to learn how to play draughts by experience.

This chapter will provide an outline of the algorithm used for the experiments in this thesis, the Support Vector Machine, together with detailed information on how this algorithm is applied and tailored for the given datasets. In addition, a description of the metrics used to judge the performance of these models is explained, as well as a justification of why this algorithm was chosen over another popular method used in genetics research. There will also be an example of how this algorithm can be used to predict different ethnic populations with 100% accuracy, using the same type of data featured in the later trials. The chapter will conclude with a literature review of examples of how these machine learning methods have been utilised in the fields of neuroscience and psychiatric genetics.

2.2. Support Vector Machines

The main machine learning algorithm used throughout this thesis is the Support Vector Machine (SVM). This was a method that was initially developed by Vapnik and Chervonenkis in the 1960s, and was first published in 1982. It started out as a technique to find the optimal way of separating two different classes of linearly separable data points by means of finding a *maximum margin hyperplane*. The term *hyperplane* is used to describe a plane in a dimension space that is greater than 3, and thus cannot be visualised, but still remains valid for the mathematical purposes of solving the task at hand. While it is capable of working within very high-dimensional space, it is best to focus on two dimensions in this chapter, as this can explain the conceptual ideas that are used to analyse all of the data within the experimental chapters of this thesis. The next section provides a series of diagrams and descriptions to help explain all of this clearly.

2.2.1. Finding optimal linear decision boundaries

The image in figure 2.1 shows an example of two different classes of data that display *linear separability.* The blue and orange circles represent data points from two different classes, each taking a value on the x and y axes that determine their position on the plot. In machine learning terminology, these inputs are referred to as the *features* of the model, and this term will be used throughout this thesis. These hypothetical classes could represent anything, but given the topics presented in this thesis, can be imagined as case (orange) and control (blue) status in schizophrenia, with the axes representing risk scores for two collections of variants or mutations. This example is very simple, but helps to show the conceptual idea behind the algorithm of an SVM.

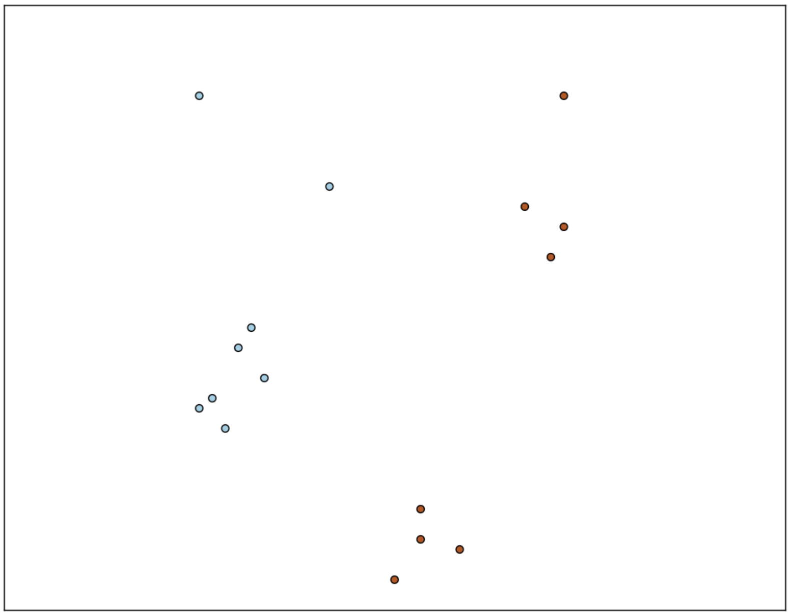


Figure 2.1.: A visual representation of two different classes that display linear separability.

The aim is to find a hyperplane (which in two dimensions can easily be represented as a straight line) that finds the optimal means of separating these two sets of data points. It is important to stress what is meant by the term “optimal” in this situation, as there are an infinite number of lines that could be drawn that separate these points. Figure 2.2 shows some examples that successfully separate the classes, but are not doing it in an optimal manner. The two black lines in particular are clipping the edge cases of the classes in two different ways, but the red line seems to represent a slightly better dividing line by appearing somewhere between the others.

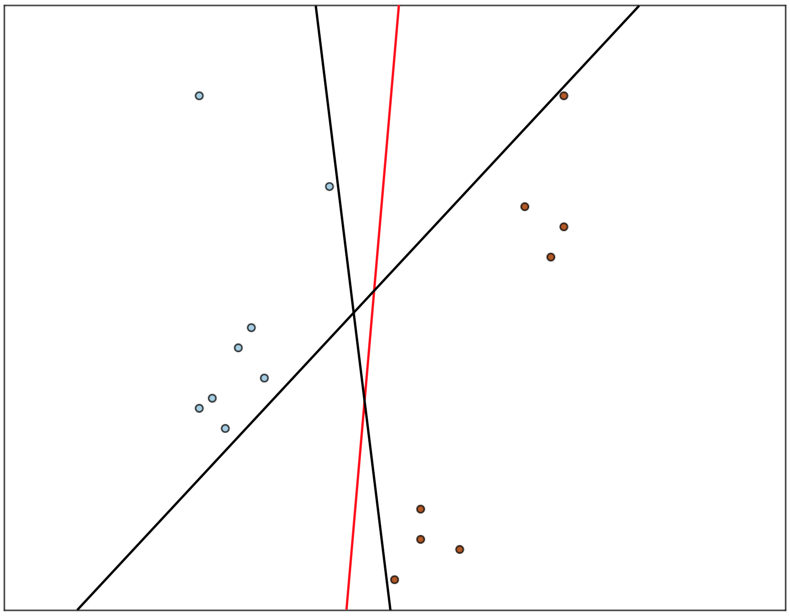


Figure 2.2.: An example showing three “sub-optimal” ways to divide the data points into two regions.

What is meant by the “optimal” solution is the line that retains the *largest possible margin* between the line and the nearest points of each class. This is shown in figure 2.3. In this figure, the centre line shows the optimum fit, and the dotted lines represent the margins which are now at their maximum possible width. The areas on either side have been coloured to represent the class of the data points belonging to each respective side. This is a visual representation of the main aim of building categorical models, which is to create models that allow the categorisation of new, previously unseen, data points. Any new data points in this example model would be plotted onto the graph, and be assigned the class relating to the colour of its position.

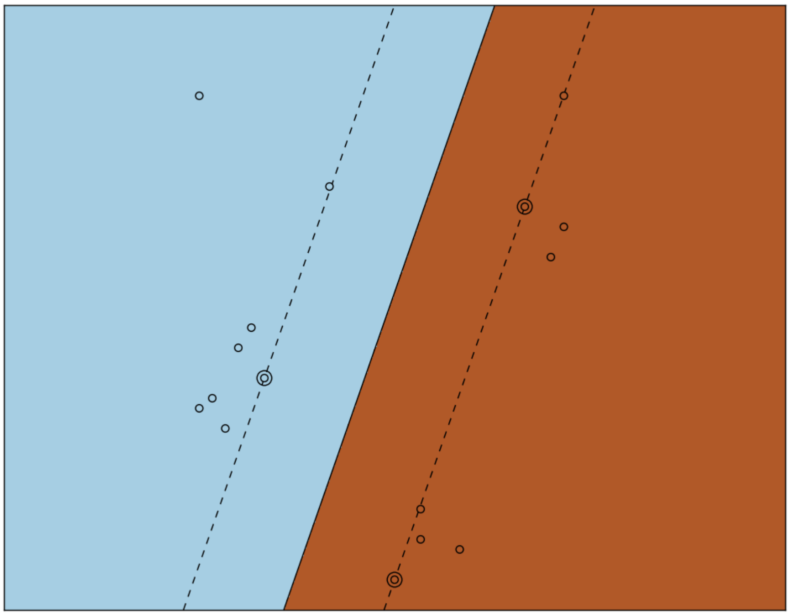


Figure 2.3.: In this figure, the optimum separating line is shown together with the dotted margin lines, the colours representing the regions assigned to each class, and circles around the data points on the margin - the *support vectors*.

Another feature in this figure is that some of the points have a circle around them. These represent the points that are closest to the margin, and it is these points that are referred to as the *support vectors*, hence the name of the whole algorithm. Due to the nature of the mathematics involved in an SVM, it is in fact only the information from these points that is used to find the optimum hyperplane. In short, it is only the “ambiguous” data points that play a role in building the model, which can lead to greater efficiency during the optimisation procedure. Of course, if there is no clear divide between classes, as is often the case with psychiatric genetic datasets, then there are many ambiguous data points, and hence an equally large number of support vectors, which results in a longer computational running time.

One of the outputs of a linear SVM is the coefficients that are assigned to the different features. What these coefficients are describing is a vector that runs from the origin of the figure (where both feature values are 0) and is orthogonal to the hyperplane. This can be seen in figure 2.4, represented as a red arrow. In this situation, the origin of the figure is in the top left (and therefore, the value assigned to the y axis feature in this circumstance would be negative). In order to calculate which side of the line a new data point falls, its feature values are *projected* orthogonally onto this coefficient vector to see how far along it lands. This can also be seen in figure 2.4 as the green dotted arrow.

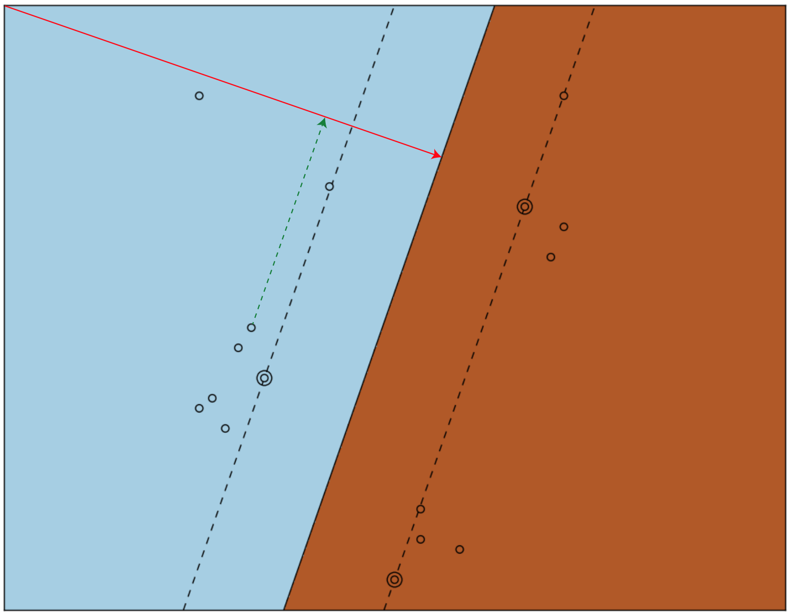


Figure 2.4.: The red arrow is this figure represents the coefficient vector, which runs from the origin of the graph, in the top left, and is orthogonal to the separating line. The dotted green arrow represents a projection of one of the data points onto the coefficient vector, and shows that this point does not pass the hyperplane boundary.

In this example, the projection clearly falls within the blue area. This projection is calculated by taking what is known as the *dot-product* between the data point vector and the coefficient vector. If the data point vector is written as $\mathbf{u}$ and the coefficient vector as $\mathbf{w}$, then the dot product is written as $\mathbf{u} \cdot \mathbf{w}$[1]. The letter "w" is used in this vector as the coefficients are also referred to as the "weights" of the model. When the SVM is *linear* in nature, then the values in this weight vector can provide some very useful interpretability to the model, in the sense that they represent the *importance* of the different features, and how much of a role they play in the categorisation. This is put into extensive use in chapter 5 when the importance metrics are assigned to gene sets.

[1] Of note, in the field of linear algebra, this can often be written as $\langle \mathbf{u}, \mathbf{w} \rangle$, which will be used in later chapters when the issues of different *kernels* is raised

Finding the *hyper-parameters* of the models

Another aspect that plays a vital role when building any machine learning model is that of *hyper-parameters* and a conceptual overview is provided with diagrams here.

The situation presented in figure 2.5 is very similar to that already presented but with one minor change: one of the blue points has now moved position. The data is still linearly separable, but the situation is more unclear as it looks as though this point should be orange, based on its proximity to the other orange data points. This brings about some ambiguity as to whether the point's colour should be interpreted as correct, or is it better to assume that this point could be a mistake? This matter is handled by setting the value of the hyper-parameter **C**, which represents the *cost-parameter*, and in essence, states how seriously the model should be concerned with not classifying the point as incorrect.

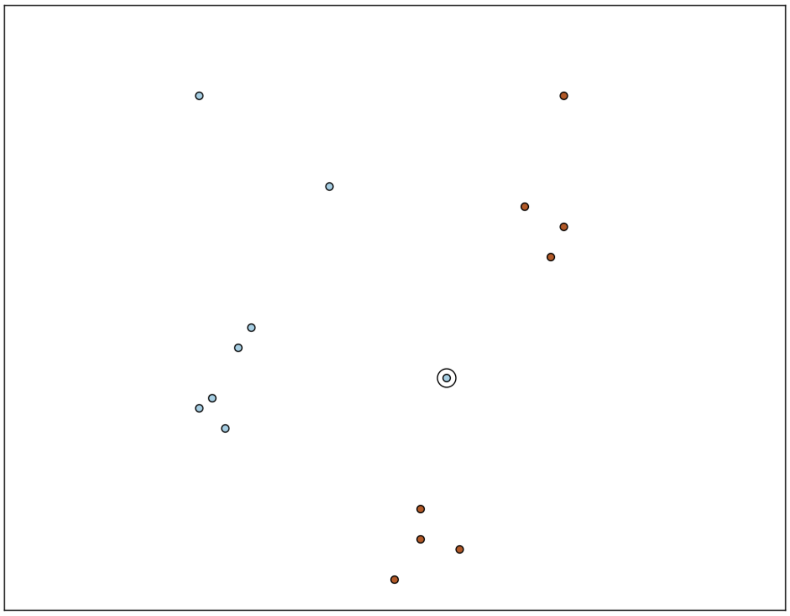


Figure 2.5.: In this example, there is an unclear datapoint, shown as circled, which while it is blue, looks like it would be better classified as orange based on its proximity to the others.

In the first example, this value has been set to 100, which forces the model to get the classification of the data provided correct. The effect of this can be seen in figure 2.6, where forcing the model to classify all of the data points correctly has resulted in dramatically reducing the width of the margins. This is a clear demonstration of what is termed *overfitting* of the model to the training set; while the model is technically correct, it looks like the line is not in the optimum position, and therefore any new data points could be possibly misclassified.

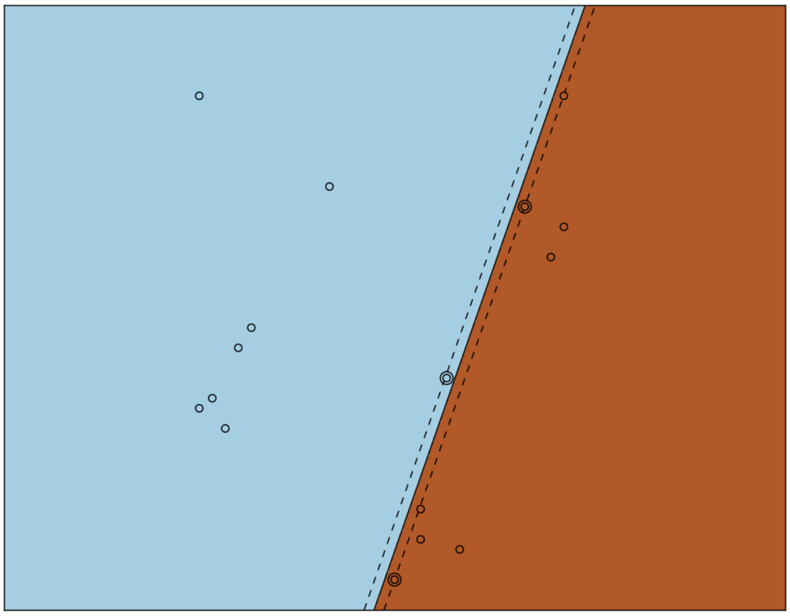


Figure 2.6.: The effects of setting the **C** parameter to 100, no incorrect classifications are made, but the width between margins is considerably decreased.

Due to this problem of overfitting, it is often far more preferable to build a model that is permitted to make mistakes on the data used to train it, in preparation for better performance on any new data points assigned for categorisation. In this example, this can be achieved by setting the parameter to **C** to 1 instead of 100, and the effects of this can be seen in figure 2.7, resulting in a model that can better generalise to new data.

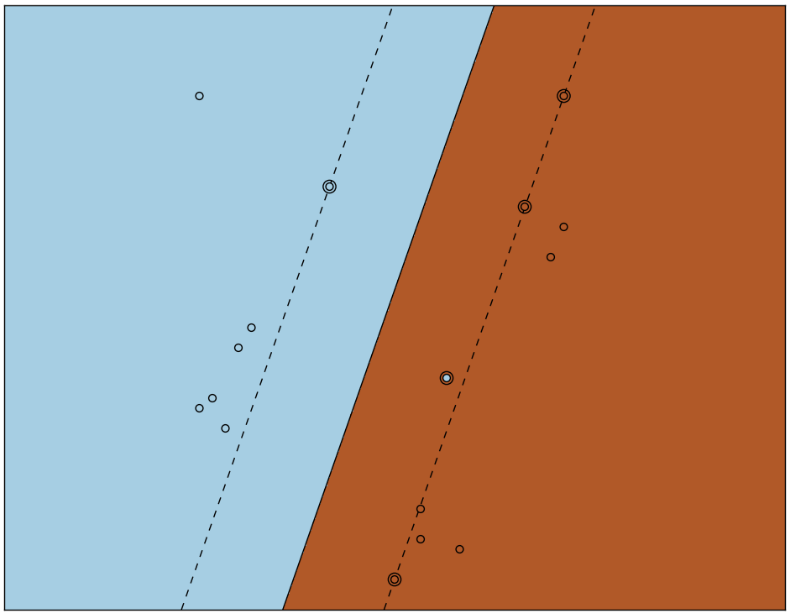


Figure 2.7.: When the **C** parameter is set to 1, the unclear data point in the training set is misclassified, but the wide margin is regained, which allows the model to better generalise to new data points.

This example also highlights another important ability and characteristic of an SVM in that it is permitted to make mistakes; the algorithm is described as a *soft margin classifier* in the sense that the dividing boundary does not have to make totally correct predictions on the training data, a concept first described by Cortes and Vapnik (1995). This is crucial as, unlike in the simple example provided here, most real datasets that are presented to machine learning algorithms are not linearly separable, and that is certainly the case with psychiatric genetic datasets.

When linearity fails to make suitable classifications

While an SVM is capable of building these soft margins to allow it to make mistakes with the training data in order to improve generalisability later, there are frequent occasions when a linear classifier is simply not suitable for use in the model. Another simple example showing this can be seen in figure 2.8. This example is a variant of the eXclusive-OR (XOR) problem, whereby the only way for a data point to be classified as positive is if it has a high value on one *OR* the other feature; not having either, or having both, will result in belonging to the negative class.

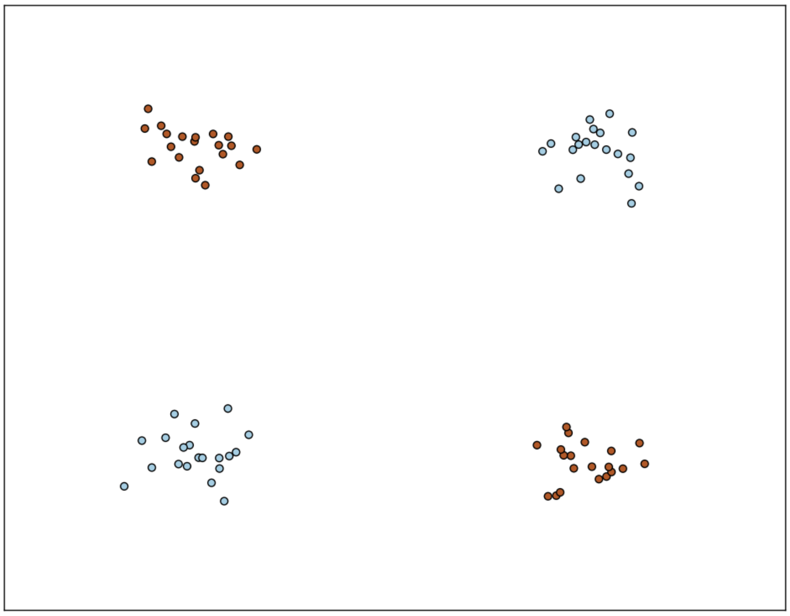


Figure 2.8.: An example showing four different clusters of points belonging to two different classes. There is no way to fit a linear decision boundary to discriminate between the two classes in this situation.

This is shown in figure 2.8 and can be related to a very simple genetic example: if the x and y axes represent scores on two different sets of mutations, a high score in either would result in developing a disorder, but the presence of both could result in the epistatic effect of each of them cancelling out the contribution of the other. There is no way that these different sets of classes can be separated by a single straight line. An attempt to do so can be seen in figure 2.9, where all of the data points in the lower left hand side have been misclassified.

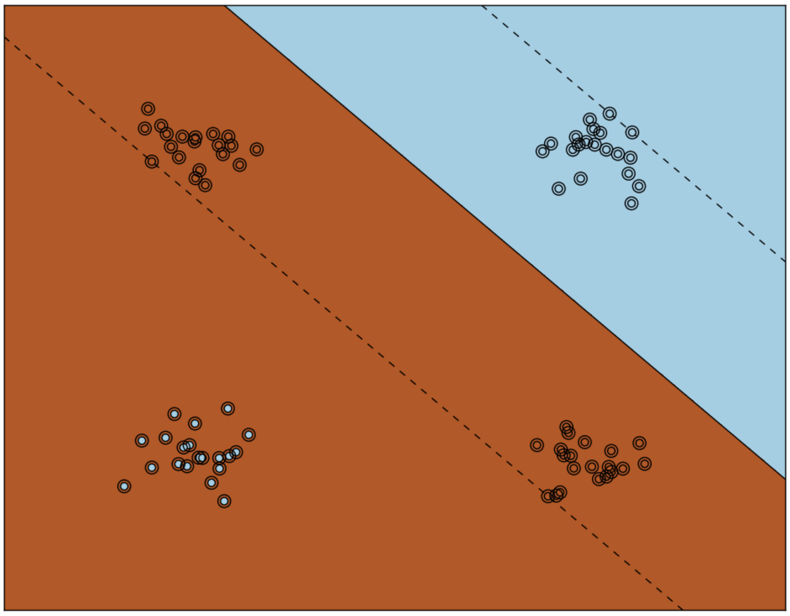


Figure 2.9.: An example showing an attempt to fit a linear decision boundary to a situation that resembles the XOR problem. It is simply not possible and many wrong classifications will be made.

Finding a suitable model to accurately predict patterns like this involves applying one of the most powerful machine learning techniques that are available to be used with an SVM, and involves mapping the original input space of the data points into a higher dimensional space using the *kernel trick*.

2.2.2. Using "The Kernel Trick" to identify non-linear patterns

This techniques involves looking at the relationships between the data points (more accurately the support vectors), which is not just a linear relationship. The example given in figure 2.10 makes use of *Euclidean distances* between the points. In this section, only a conceptual overview is provided and an example of some of the mathematics involved is given in chapter 3.

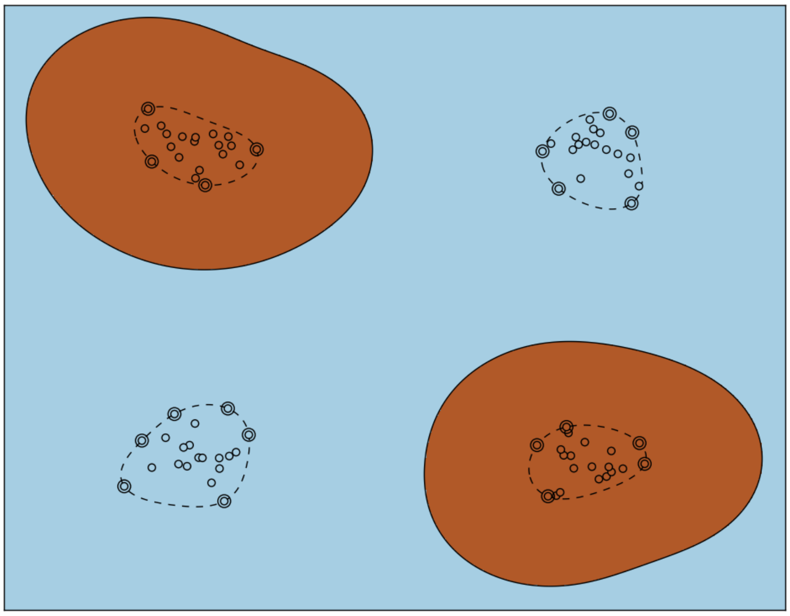


Figure 2.10.: Here, the RBF kernel has been used to find clusters of similarly placed data points, measured on their Euclidean distances from each other. This technique has allowed for perfect classification of the data.

This method, using the Radial-Basis Function (RBF) kernel is clearly very effective at dealing with the XOR problem. When using this kernel, the data points are actually being mapped into a higher dimensional space than the two provided by each data point (the x and y axes values). The act of mapping into a higher dimensional space allows data that were not linearly separable to actually become linearly separable, a concept described in *Cover's Theorem*. The full details of the proof of this theorem are not given here, but can be read about in the original publication (Cover, 1965).

Instead, a graphical representation is provided that helps to clarify this concept. The example in figure 2.11 shows a warping of the canvas used to plot the points in the XOR problem into three dimensional space. If the blue and orange points are now plotted onto the warped canvas, the orange points would be located towards the top of the higher peaks; this means that a linear separating plane can be drawn that allows for perfect classification. This idea works in much higher dimensions, but these are not possible to visualise, so this simple example is given instead.

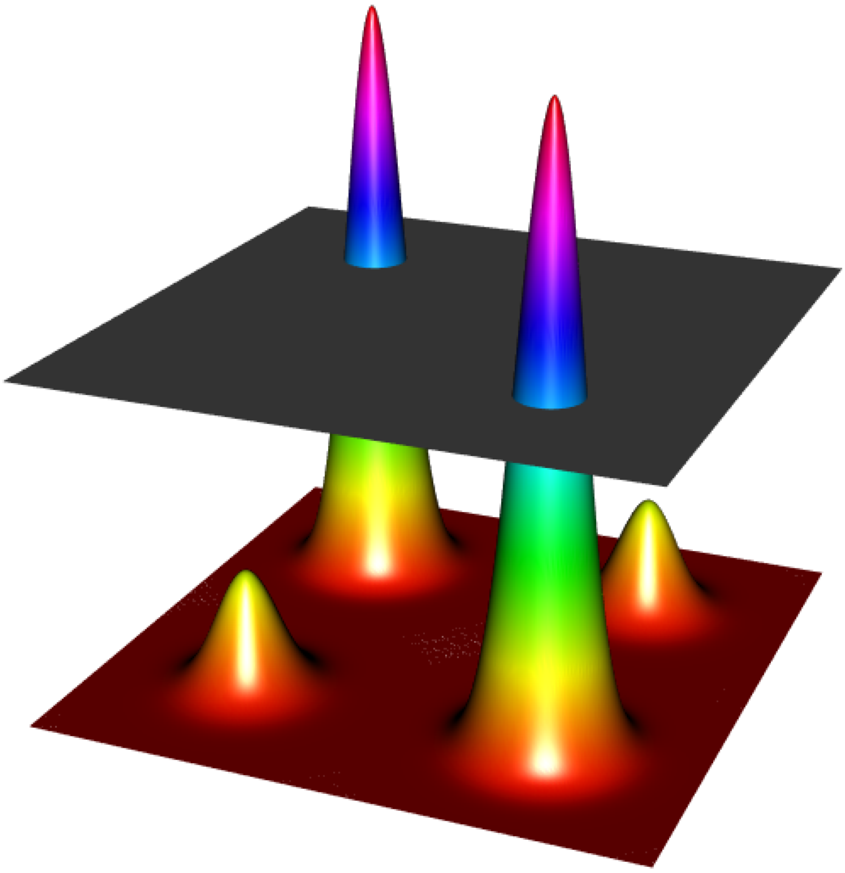


Figure 2.11.: A example showing how the mapping of the XOR problem into 3-dimensional space allows for a linear plane to be drawn that successfully separates between the classes.

This example can also be used to explain an important, if unfortunate, effect of using the kernel methods in an SVM. As was mentioned earlier, the values in the coefficient (or weight) vector actually provide some useful interpretation information because the values assigned to the features can be used to assess the importance of each respective feature, as this value determines how each data point is projected and whether this results in them crossing the decision plane. When using a kernel however, it is not the position in the original input space that determines the classification, but the position in the higher dimension. In the example provided above, it is actually the height of the peak where the data point lands that states which class it belongs to. This is a classic example of how increasing the complexity of the model results in the decrease of its *interpretability*; in other words: some models focus on delivering predictive power while others focus on increasing the interpretation of what could be driving the performance and results, and care must be taken when deciding which to use (Shmueli, 2010). This is one of the reasons why the SVM was chosen to be used in this thesis as it can provide an element of both: if interpretability is required, then the focus can be placed onto the linear model, but if an increase in predictive power is required, especially when looking for the non-linear effects of interactions, then the kernel methods can be employed.

The use of kernels bring about the additional complication of extra hyperparameters that also require tuning to find the optimum values. For the RBF kernel, there is an additional parameter: $\boldsymbol{\gamma}$ (the Greek letter pronounced “gamma”), which sets the width of the Gaussian kernel to use. The value used in the example in figure 2.10 was 3, but an additional example is shown here using a different value to highlight the effect of changing this. In figure 2.12, $\boldsymbol{\gamma}$ is set to 50, and it can be clearly seen that this results in a tightening of the kernel around the orange points to make the region far more specific to the areas close to these data points. This is another example of overfitting, as it is unlikely that new data will be classified correctly in this example.

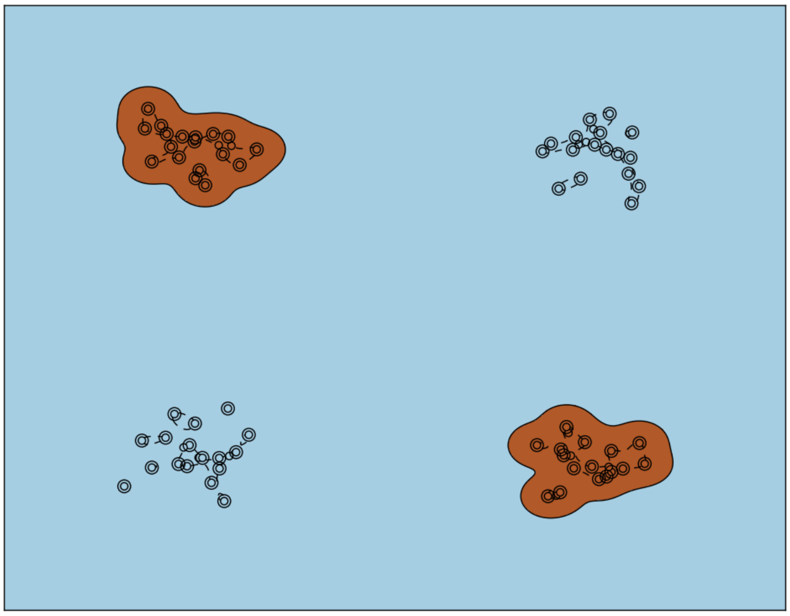


Figure 2.12.: An example of the RBF kernel being used on the XOR related problem, but this time using a $\boldsymbol{\gamma}$ value of 50. The effect is that the kernel width has been tightened to become more specific to the data points, and could very well result in the overfitting to the training set and misclassification on unseen data

All of the machine learning modelling performed in this thesis was carried out using the *scikit-learn* machine learning library (Pedregosa et al., 2011), written for use with the *Python* programming language. When performing kernel based modelling using SVMs, the default value for $\boldsymbol{\gamma}$ is set to the reciprocal of the number of data points in the training set: $\frac{1}{N}$, which attempts to avoid the problem of overfitting. The examples of the RBF kernel shown in this chapter were actually using unnaturally high values of the $\boldsymbol{\gamma}$ parameter for the benefit of the visualisations, but as can be read about in chapter 3, when applied to the real-life datasets, it is these lower values of $\boldsymbol{\gamma}$ that perform much better.

A comment on the scaling of the features

Another important point to note about the SVM algorithm is that it is sensitive to the scale, or range, of the values for each of the input features. This is because the algorithm looks for patterns that the data points are projected into in *common dimensional space* between all of the features. To clarify this point, an extreme example is given here: if the feature represented on the x axis were in a very small range of 0.001 to 0.01, but the feature on the y axis was in an extremely large range from 10,000 to 100,000,000, then any plotting of these points onto a common set of axes would show almost no variability along the x direction compared to the y direction and it would be almost impossible for the SVM to find any optimum boundary in this situation.

It is therefore necessary to carry out some *pre-processing* steps of the data before the model is built and fitted. One of these common steps to remove the problem of different feature ranges is called *feature scaling*, and if often performed by transforming each feature so that the distribution of its values in the dataset have a mean of 0 with a variance of 1.

2.2.3. Finding the correct hyper-parameters through Cross-Validation

One of the difficulties in finding the optimum value of these different hyper-parameters, is that there is often no analytical means of calculating, or even estimating, which will perform best in advance. This means that the values have to be found by using a trial and error approach in a technique called Cross-Validation (CV). One fundamental issue that is important to remember when building any machine learning model is that *the data points used to train a model cannot play any part in the assessment of the performance of the same model.* The whole aim of any model is that it can be used to classify *new* information provided to it; there is every possibility that the performance of a model on information that was used to build it could approach very high levels, especially in cases of overfitted models. In order to counteract this, procedures have to be carried out to ensure that the model is always tested on a subset of the data, which was held out when it was being built.

During the CV process, all of the data points are repeatedly split into two different sets: the *training* set, and the *validation* set. The training set takes up the majority of the data points, with a subset being held out in the validation set for testing the performance of the model. Splits of 90/10% or 75/25% are common, but it is often favoured to choose the latter of these as it results in less erratic performances in noisy datasets where there is no clear decision boundary to be found. Of note, all of the CV procedures in this thesis use a 75/25% split. There are primarily two different ways that the data can be divided: using set *folds* of the data, or by performing random shuffled splits. In the first method, often referred to as *k-fold* Cross-Validation, the data is split into k sections of equal size, with k representing as many folds as desired for the size of the CV partition: $k = 10$ for 10% and $k = 4$ for 25%. Each partition plays the role of the validation set once, and all of the others are used to train the model. This procedure is then iterated over all of the partitions in turn. In the other method, there are no pre-set splits of the data, but at each iteration, a random split based on the desired quantities is made, and this is repeated for as many times as desired[1]. An example of k-fold CV is demonstrated in chapter 3 and the shuffle technique is used in chapter 5. An additional feature of CV that can be carried out if desired is ensuring that every split is *stratified* in nature. This term means that the proportion of classes (case/control status in all of the studies in this thesis) is maintained at every split. This is highly desirable, as it prevents any splits from having a large imbalance between the classes, which could affect the performance.

It is also important to point out that any pre-processing steps that occur during the building of the model must also take place within the CV process. The common pre-processing step mentioned so far is the scaling of the features before the model is built. It is important that this is done in the correct manner. A naïve approach would be to simply take all of the input features in the entire dataset, and transform them so that they all have a mean of 0 and a variance of 1 before starting the CV process. This is incorrect, as there is information about the distribution of the scores from data points that are involved in building models that has an effect on the scaling transformation of the values in the validation data points. The correct approach is to build the pre-processing steps into the entire CV process by means of building a *pipeline*. This term means that for every split carried out during the CV process, the training data is transformed to have mean 0 and variance of 1, but the parameters used for this transformation are then applied to the validation data, without any additional information from those points. In short, the validation data points are scaled according to the distribution of the training data points. This is performed at every split, and as this is such a common task in machine learning,

[1] Information on these methods can be read about on the software web-pages: `http://scikit-learn.org/stable/modules/cross_validation.html`

the functionality has been built into the software packages.

Searching through the hyper-parameter space

Now that a method is set up to test the hyper-parameters, there must be a way of setting out which ranges and distributions of these are to be tested in the whole CV routine. There are two common methods of doing this. The first is called a *grid-search* routine, which performs an exhaustive search of every possible permutation of preset, user-defined, values for all of the different hyper-parameters. This procedure is thorough, and is often very useful in cases when only a few values of each hyper-parameter need to be tested, but the computational running time can often be too great if too many values are to be used. If a wider search is required, another technique can be employed that makes use of *Monte Carlo* simulation methods. For each selection procedure, new values are drawn from pre-defined parameter distributions that have been chosen on the assumption that they give greater weighting towards values that have an a-priori estimate of being more suitable. This procedure allows for many different combinations to be tested, without the need to dwell in areas of hyper-parameter space that could be completely unsuitable. An example of this procedure can be seen in chapter 3.

A visualisation of the whole CV procedure can be seen in the flowchart in figure 2.13. This shows that for every set of parameters chosen (either by the grid or Monte-Carlo methods), all of the split-permutations are carried out, represented with the inner loop, and the average performance of these recorded before new parameter combinations are chosen again, shown with the outer loop. At the very end of this, the best performing parameters are chosen to be used in the final model.

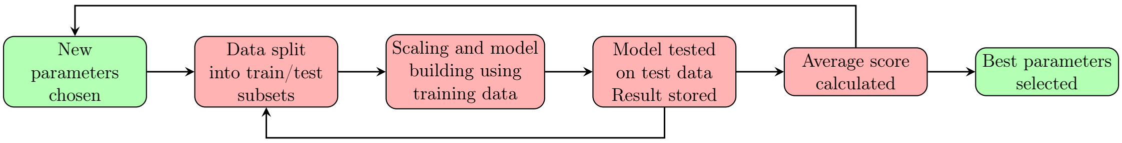

New parameters chosen
Data split into train/test subsets
Scaling and model building using training data
Model tested on test data Result stored
Average score calculated
Best parameters selected

Figure 2.13.: A flowchart showing the different repeated iterations that occur during the procedure to find the optimum hyper-parameter values with Cross-Validation.

A comment on the use of addition test sets

So far, it has been mentioned that the original dataset is always split up into training and validation sets to build the best model. However, if the aim is to build a truly predictive model, then an additional *test* dataset must be used. These data must have played *absolutely no role* in the building of the model, and this includes the whole CV routine that was used to select the best hyper-parameters. An example of this being put into effect can be seen in chapter 3, when the initial intention was to see how effective a predictive model could be built with the machine learning algorithms. However, in later chapters, the focus shifted towards examining what extra information about the genetic aetiology could be obtained from the different performances of different algorithms, which focussed on *comparative* measures of performance instead of trying to obtain the most accurate overall model. Because of this, the use of a specific test set was no longer required.

2.2.4. A summary of the optimisation routines: LibSVM and LibLinear

The computational methods used for finding the optimum decision boundary (or more specifically, the largest margin classifier), is an example of *quadratic programming* (Cortes and Vapnik, 1995; Murty and Yu, 1988) whereby the minimisation of the *error*: essentially any deviation from the optimum solution, takes a quadratic form. The full details of this process cannot be described here, but the key important feature of a quadratic optimisation problem is that it takes on a *convex* form. This shape can be imagined in two dimensions by looking at the shape of the curve in the function $y = x^2$. In this curve there is a *single minimum point* at $x = 0$; and it is this feature of only having a single or *global* minimum point that defines a quadratic problem. This is a fundamental advantage of SVMs as any decrease in the error of the model is *always* moving towards the most optimum solution possible. For some types of algorithm, for example artificial neural networks, the error function can contain *local minima* points that make the isolation of the best possible solution very difficult to find (Hastie et al., 2001).

In the optimisation procedure for an SVM, there are two main programs that have been incorporated into most of the software written to build the SVMs: the *LibSVM* algorithm by Chang and Lin (2011) and the *LibLinear* algorithm by Fan et al. (2008a). The former can deal with the non-linear kernel methods, while the latter focusses on linear boundaries but is optimised for efficiency in extremely large datasets, especially those which display more input features than the number of available samples.

In addition to finding the optimal solutions, the LibLinear algorithm is capable of introducing additional *constraints* on the coefficients given to the different features. This prevents the overfitting of the models to the training data, by not allowing the coefficients to become too *tailor-made* to the training data (Hastie et al., 2001). There are different methods for how to make such constraints, and more detail is provided in chapter 5 when the results from these two methods are compared.

2.2.5. How the performance of the models was measured: the Area-under-the-curve metric

Throughout all of the experimental chapters in this thesis, the target outcome for all of the models is *binary*, meaning that there are only two classes being represented: cases with schizophrenia and controls. In order to assess the results from models built for binary inputs, a particularly good metric to use is called the Area-Under-the-Curve (AUC) metric, which can also be referred to as the Receiver Operating Characteristic (ROC) curve, due to its origins in signal processing (Metz, 1978).

The advantage of this method is that it is capable of dealing with imbalanced class sizes in models. This gives a real benefit over metrics that just look at the percentage of correct answers given across both classes. While it is preferable to include an equal number of representations from both classes in a binary classifier, it is not always possible, especially in fields like psychiatric genetics where the number of cases in the whole population can be less than 1%. An extreme example here is provided to highlight this point: if one of the classes makes up 90% of the data points in the model training set, and every subsequent CV and test set provided, it would be possible to have the model simply assign that class label to every single data point that it sees. This would provide an accuracy score of 90%, as it is technically getting this number of predictions correct. However, this model is clearly of no use as it is simply a naïve classifier.

While in reality, it is unlikely that a model would be trained on such imbalanced data, it is common that there could be slight imbalances, and any change in performance levels could be affected by these, so it is important to use a suitable metric like the AUC. Of note, for all of the experiments carried out in this thesis, there are slight imbalances between class proportions, but the results will not have been affected due to the use of a suitable performance metric, and in addition, these imbalances were taken into account when the models are built. More details about how this was done is provided in the methods section of chapter 3.

The AUC method works by making incremental changes to the threshold at which a data point is classed as a positive case, and then looks for the trade off between the True Positive Rate (TPR) and False Positive Rate (FPR) at each stage. These trade offs are then recorded and plotted for each step. All of this will be explained in a series of diagrams and examples. The formulae in equation 2.1 show how these are calculated, and can be related to the orange and blue data points in the earlier figures. Recall that the orange points were the positive classes, so it is desirable for these to be labelled as so; the blue points are the negative classes, so ideally as few of these as possible will be labelled as positive. The top equation refers to the orange points; those that have been correctly classified are the *True Positives* and those that have not are the *False Negatives*, so this equation represents the number of orange points that have been *correctly* classified. The second equation is the converse for the blue points; the *False Positives* are the negative points that have been mistakenly classed as members of the positive class, so this equation represents the proportion of blue points that have been *incorrectly* classified. Ideally, the output of the first equation should be as high as possible, and that of the second one as low as possible.

$$\begin{aligned}\text{TPR} &= \frac{TruePositives}{TruePositives + FalseNegatives}\\ \text{FPR} &= \frac{FalsePositives}{FalsePositives + TrueNegatives}\end{aligned} \tag{2.1}$$

A similar linear graphical example to those presented earlier is now shown in figure 2.15 to demonstrate how this is put into effect. The only change that is made here is that the points are no longer linearly separable, so a perfect linear decision boundary is not possible, and a soft margin has to be used.

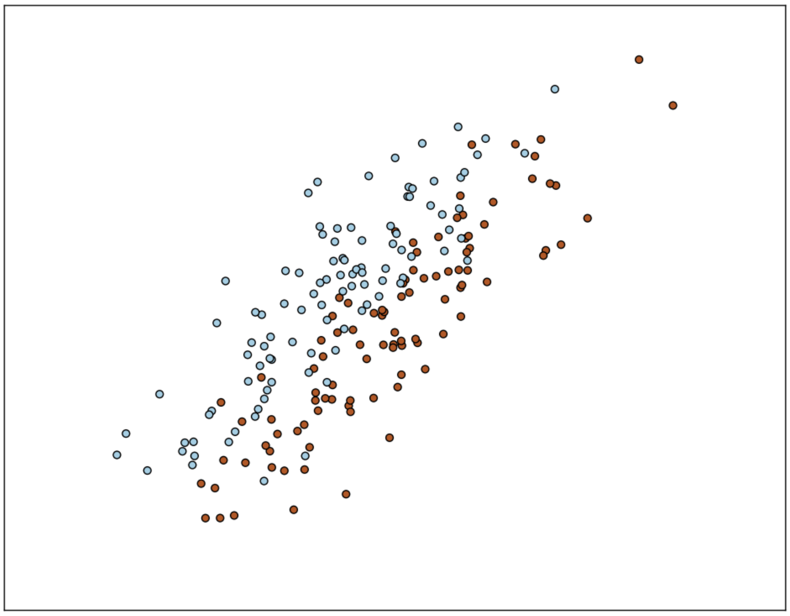


Figure 2.14.: An example showing data points that still show a definite trend for being differentially placed on the graph, but are not completely linearly separable.

While there are some ambiguous and overlapping points, it is still clear that there is a trend for the orange points to be further towards the bottom right, away from the origin in the top left corner of the axes. In figure 2.15, the optimum decision plane has been added together with the support vectors and the margin regions. There are now far more support vectors due to the increased ambiguity in the model.

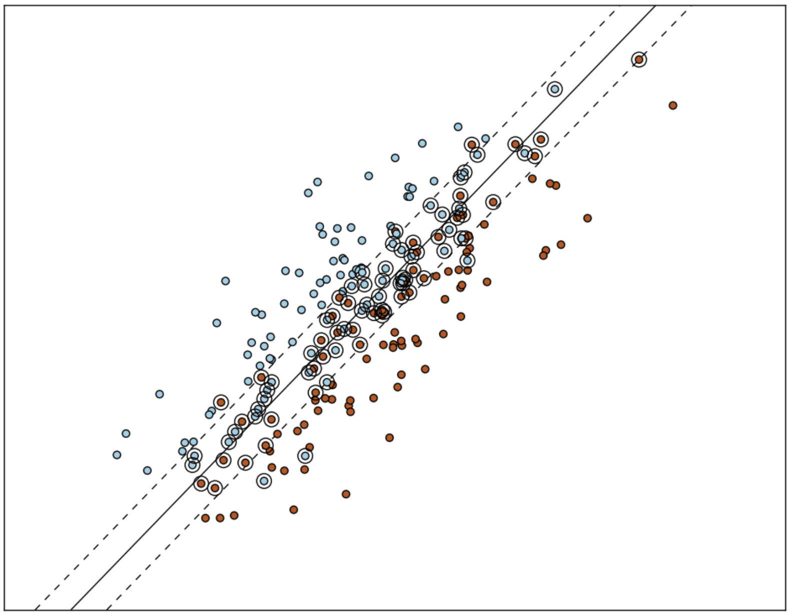


Figure 2.15.: An example showing data points that still show a definite trend for being differentially placed on the graph, but are not completely linearly separable. Also shown are the optimum decision plane, the margins and the support vectors.

First, an example of the ROC curve for this dataset is shown, and then an explanation of how it is calculated will be provided. This can be seen in figure 2.16.

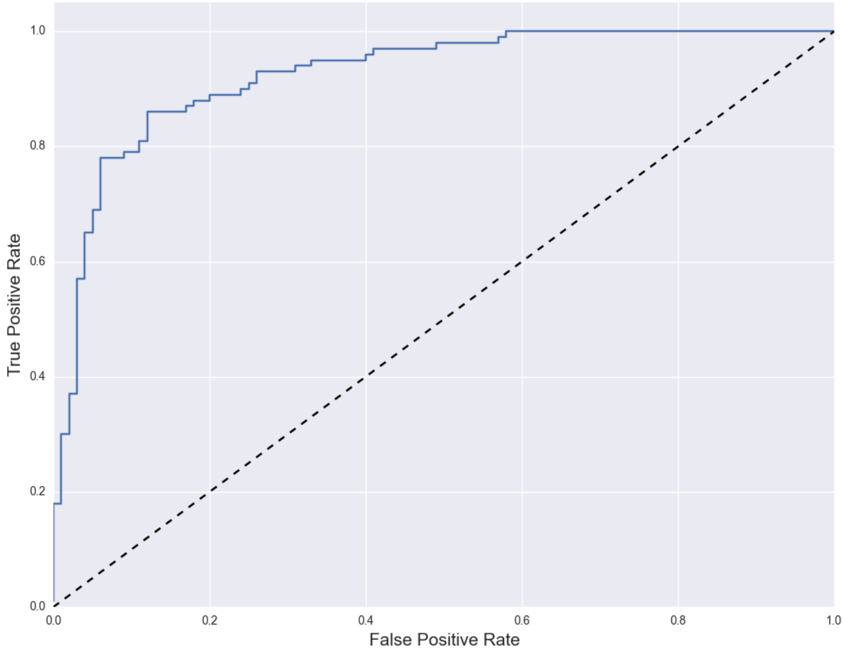


Figure 2.16.: An example of an ROC curve with a high level of performance. The FPR and TPR are shown along the x and y axes respectively.

In this figure, the FPR is represented along the x axis, and the TPR along the y axis. The *ideal* location on this figure is therefore the top left hand corner, which represents full identification of the positive classes, with no false positives from the negative classes at all. This point is only obtainable when the data displays perfect separability. The dotted line from the bottom left to the top right outlines the pattern that would be seen with total chance performance, as any desirable increase up the y axis comes with the exact same increase along the x axis, which represents a greater number of false positive classifications. This also explains the ranges of this metric; it does not range from 0 to 1, but from 0.5 to 1 as the value is calculated by literally taking the area under the curve. In the example in figure 2.16, this value was 0.925, showing a good level of performance. To explain how this curve is calculated, some additional lines have been added to the previous plot, and can be seen in figure 2.17

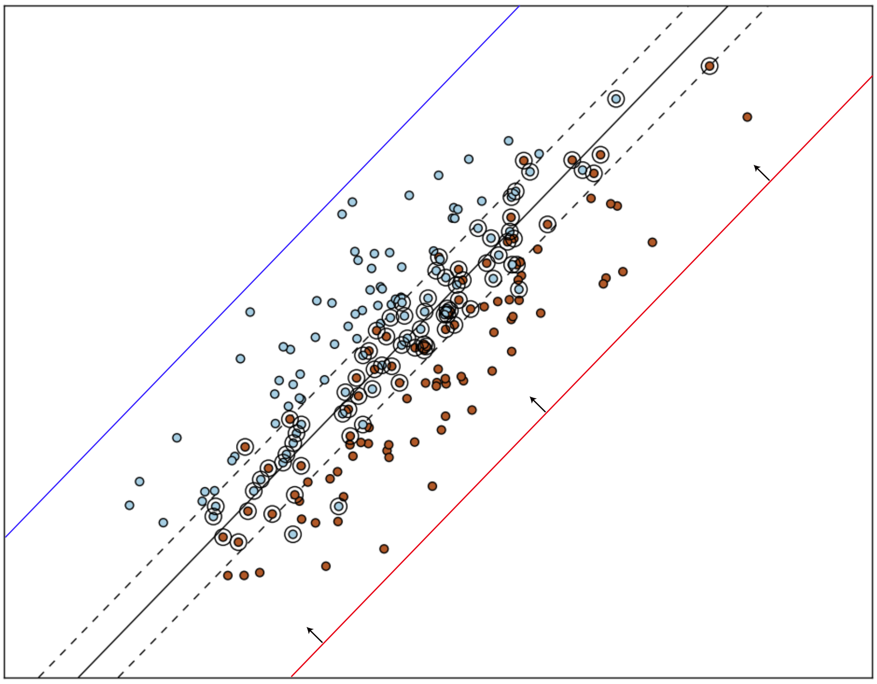


Figure 2.17.: In this example, the cross-over data points, together with the optimum decision line and margin regions, can be seen together with two extra coloured lines representing the extrema of the moving threshold positions. The threshold starts at the red line, where no point is classified as positive, and moves in the direction of the arrows shown to the blue line, at which point all data points are labelled as positive.

In this plot, there are two extra lines which represent the extreme points of the moving thresholds used in the calculation of the metric. These lines are parallel to the optimum decision plane. The red line represents the starting point; at this position, none of the data points is classified as being positive, so this position would be represented as the bottom left hand corner of the curve in figure 2.16. The blue line represents the opposite situation: where all of the points are classified as positive, and takes the position of the top right corner of the curve. The iterative changes in the threshold position move in the direction of the arrows shown on the red line, and new classifications are made at each iteration. As can be seen, as this line moves, it initially starts to pick up correct classifications, so the curve starts to be drawn up the y axis, but as it starts misclassifying the blue points as positive, the curve moves along the x axis. At the end point of the blue line, it is inevitable that there will be 100% false positives displayed, but the important information is how the proportion of the TPR and FPR has varied throughout the threshold change.

To see how the ROC curve changes with different levels of AUC performance, a final example is given. In this situation there is still a trend for the orange classes to be further to the bottom right, but the different classes cross-over far more, so there is far more ambiguity regarding where to place a linear classifier. A scatter plot of this can be seen in figure 2.18.

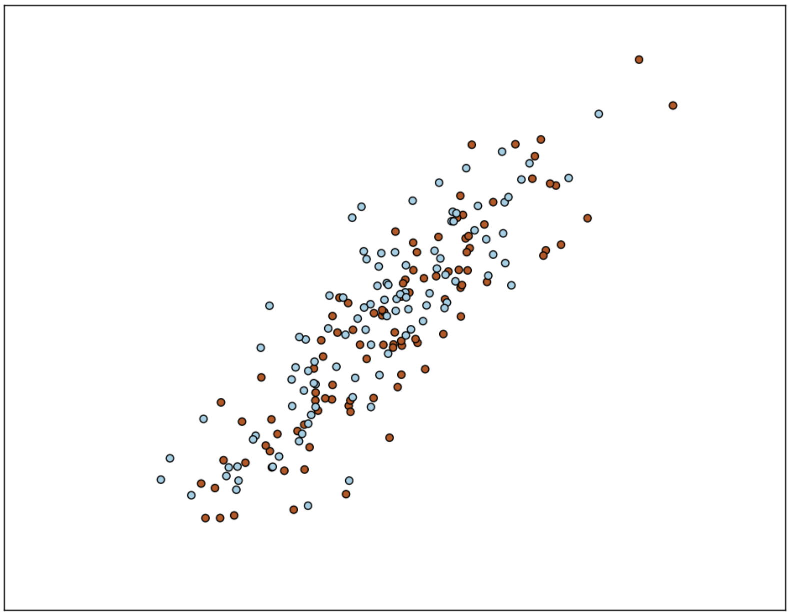


Figure 2.18.: An example of two classes of data points that show a high level of cross-over. With these data, it is very difficult to accurately discriminate between the classes with a linear decision boundary

Here it is clearly seen that any attempt to move the boundary to classify more of the orange data points correctly comes with the additional misclassification of several blue points. A plot of this new curve is shown in figure 2.19 and the difference in shape is obvious. The AUC score for this dataset is 0.637.

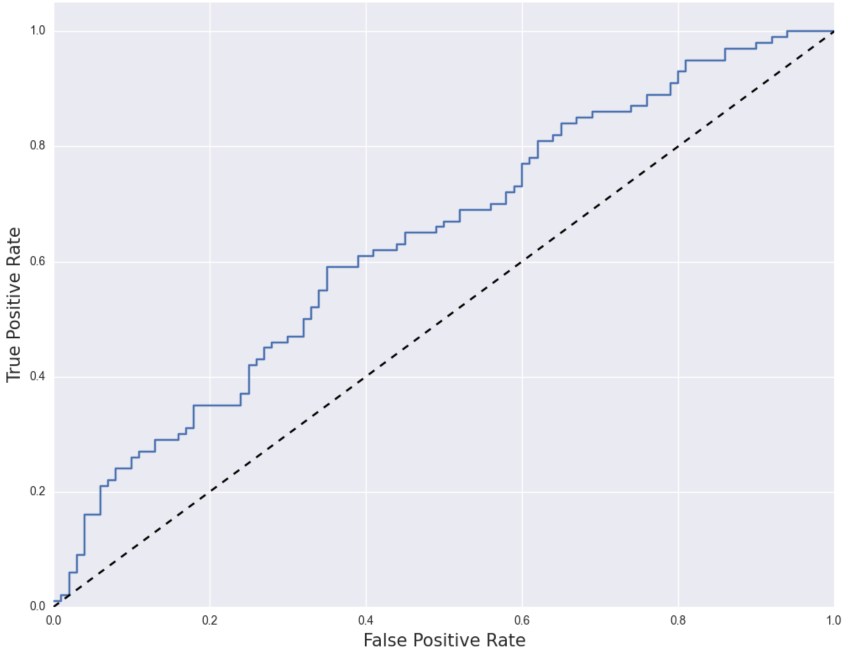


Figure 2.19.: The ROC curve for the data shown in figure 2.18. As expected, the performance is far lower in this example.

2.2.6. Correcting for the effects of Linkage Disequilibrium

As was mentioned in chapter 1, one of the major complications that can occur with genomics research is ensuring that the information provided from each mutation or variant, which will now be collectively referred to using machine learning terminology as features, is not correlated with the signal from other features due to being in a state of Linkage Disequilibrium (LD) with each other (Slatkin, 2008). There are two main methods to deal with the presence of LD, and both are dealt with in the specialist genetics software PLINK (Chang et al., 2014; Purcell et al., 2007). This software contains a set of command line functions to perform a whole suite of different techniques used in genomic research.

The first of these techniques is called *pruning*. In this process, a sliding window of a size in DeoxyriboNucleic Acid (DNA) Base Pairs (BPs), defined by the user, is passed over the genome information, with a user-defined degree of overlap, for all of the samples. The algorithm used in this process can identify Single Nucleotide Polymorphisms (SNPs) that are in LD with each other, and one of these SNPs is chosen to represent each of the regions. The threshold used to determine whether they are in LD is calculated by the r^2 metric, which is also defined by the user. In the pruning process, the choice of the SNP to represent each LD block is not important; the only information that is required is a representation of the features for all of the blocks in the genome.

The second process is called *clumping*, and this method is used when there is additional information from a Genome-Wide Association Study (GWAS). The main difference in this technique is that the GWAS results provide additional information about which of the SNPs are associated with the disorder in question, and therefore this will affect the choice of the SNP used to represent each block of LD, as it gives preference to the most highly associated SNP. In this method, the structure is a little less rigid as the user defines the r^2 threshold, as well as a minimum window size; this window is not used in the same sliding method used in pruning, but states that all SNPs within a certain region of each other are, by default, in LD, regardless of the r^2 calculation. Further information about how this process is performed can be read about on the PLINK website [1].

2.2.7. A brief description of Random Forests

The Random Forest (RF) algorithm is another very popular method that has been used extensively in many areas of research and predictive analytics. It is a form of *ensemble learning* as it is constructed by building a whole series of weak predictive models, which can combine their results to form accurate models. Examples of its usage to identify complex genetic relationships will be provided in the literature review at the end of this chapter.

The original idea of the RF algorithm is developed from the use of Decision Trees in classification tasks. These methods work by looking for dichotomous splits in the different features based on certain information criteria such as the Gini Index and the Gain Ratio, a full description of these methods is not described here, but can be read about in Rokach and Maimon (2005). The structure of the decision trees results in an algorithm that takes the form: “If feature x is greater than a set threshold, move to branch 1, if not move to branch two etc. . . . ”. This then plays across the different features, always looking for which feature to split based on the information criteria in the data. When the branches of the tree have been built, a new data point can be classified by traversing through the branches using the criteria and thresholds to get to the end of a certain branch that will return the classification.

[1] `https://www.cog-genomics.org/plink2`

While decision trees can be effective models, they come under two main criticisms: they often cannot deal with complex feature interactions as they can only consider a threshold break point in one feature at a time, and another serious criticism is that they have a tendency to overfit to the data that was used to train them, and not perform well at generalising to new data. These issues have been solved by using ensemble methods on the decision trees, hence the terminology that they are now turned into *forests*.

The first development to build these forests was developed in the 1990s, developing on the idea of random subspace sampling methods (Ho, 1995, 1998). The idea behind this is to build separate decision trees on different samples of the features; this means that none of the trees has the full access to all of the input features when building their branches. This idea of sub-sampling was further developed by Breiman in 2001, when he expanded the idea of a forest to incorporate his previous development of the *bagging* algorithm, which stands for "bootstrapped aggregation" (Breiman, 1996). This method works by not only taking sub-samples of the features for the different trees, but also bootstrapped collections of the *samples*; meaning that at each tree, a collection of samples is taken from the training data and these are used to make the forest of the decision trees. This effectively results in an ensemble of weak predictors - the individual trees. However, when all of the information from the results from different trees is aggregated together, this has been shown to be effective in many situations at achieving a high level of predictive performance while avoiding overfitting. Also, while the models cannot take into account any explicit interactions occurring between the features, as the binary decisions are always made on individual features, by making a series of predictors that use different combinations of the features, these methods are capable of solving complex categorisation tasks.

In addition to reports of good predictive performance, there are some other very useful aspects present in the RF methods. When the threshold-based decision splits are made for all of the different branches, for all of the trees in the ensemble, the information criteria used to make these, as well as the features chosen for the splits, are stored. This means that when the final ensemble model is built, there is additional information on the *feature importance* of all of the different inputs. While the SVM can gain this information when a linear decision boundary is made; this ability is lost with the use of the complex non-linear kernels. Some examples of how this has been put into effect when looking for important SNPs in genetic studies when interactions have been taken into account show that Variable Importance Measuress (VIMs) can be assigned to the variants by a range of different techniques including Random Forests (Nicodemus et al., 2007; Nicodemus and Malley, 2009; Nicodemus et al., 2010).

Another advantage of using an RF is that it is not sensitive to the different scaling of the features. This means that if the raw data contains information from different sources with vastly different ranges in their distributions of values, these inputs can be implemented into the models; no additional scaling in the pre-processing steps is required.

Why the Support Vector Machine was chosen

At the start of the project, there was a strong amount of consideration given to using the RF algorithm, due to its proven effectiveness in many different fields and the two advantages of presenting the importance of the features and the resilience to the different scalings of the features. Unfortunately, all of the attempts at any pilot trials of these algorithms on the genetic datasets yielded models that were no superior to random chance; the AUC scores were always in the region of 0.5. This was the case for all of the trials carried out in the experimental chapters. It is not fully known as to why this could have occurred, but a few suggestions are presented here.

It is possible that the ensemble method to find the different interactions between the genetic features was not sufficient at identifying any of the possibly very weak interaction effects which could be occurring. The kernel based SVMs have an advantage here by actually taking into account these explicit interactions when building their decision boundaries, albeit with a loss in the interpretability of the models.

However, the RFs still did not perform as well as the linear SVMs, and this cannot be explained by the different methods of analysis of the interactions. The only way to get an understanding of this is to examine the different types of boundary which are drawn when using the RF algorithm on similar examples to those seen already.

In figure 2.20, the boundary lines for an ensemble of ten trees in a forest can be seen. The plot has been designed so that all of the different boundaries are laid on top of each other and the overall pattern can be seen. It is immediately noticeable that all of the lines of the boundaries are parallel with at least one of the axes, this is due to the binary splits being based on the single features. The overall pattern shows a tendency to follow the similar direction to the optimum hyperplane in the SVM.

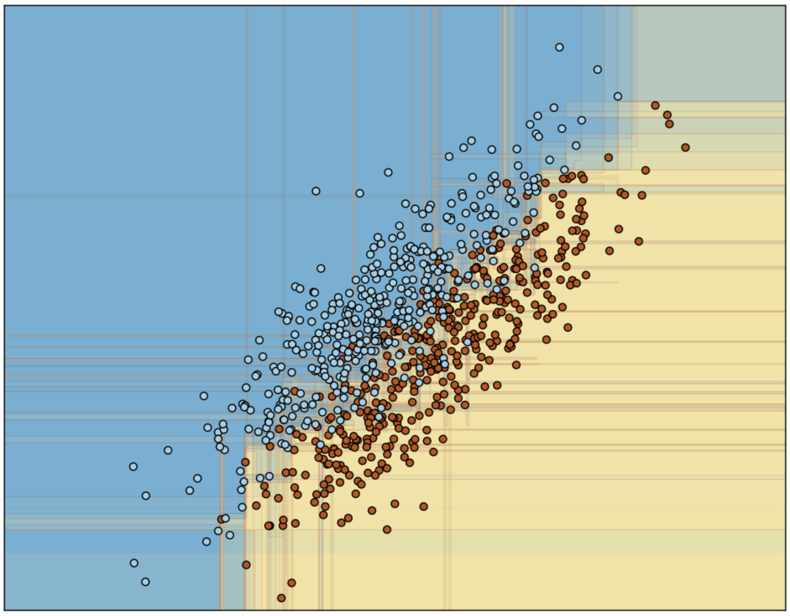


Figure 2.20.: An example of the decision boundary made by a Random Forest algorithm on data that can be reasonably separated with a soft margin linear classifier, using ten different bootstrapped estimators. Each different boundary made has been shaded in turn to create the final aggregation, which would be used for the classification of the data points.

A possible reason for why the RFs are not performing as well, could be due to the incredibly ambiguous nature of the genetic datasets used in the experiments. For all of the trials, the prediction metrics were not particularly high, so there will have been a lot of cross-over of the points. When looking at the image in figure 2.20, it is reasonable to think that there could be some confusion in these areas with a great deal of overlap; the cut-off point is not so clear as it is with the single line in the SVM.

The decision boundary for the XOR example can be seen in figure 2.21. It is clear from this image that the RF is doing a successful job at classification.

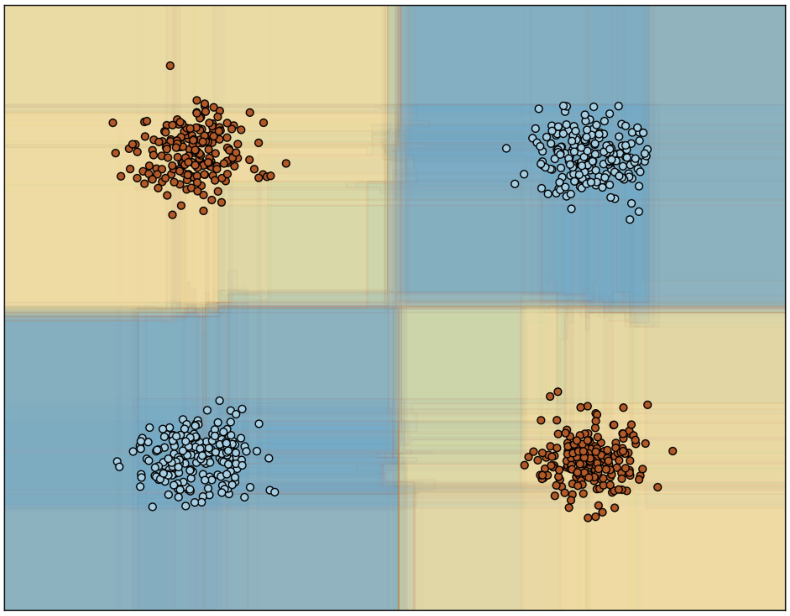


Figure 2.21.: An example of the decision boundary made by a Random Forest algorithm on the XOR example, using ten different bootstrapped estimators. Each different boundary made by these has been shaded in turn to create the final aggregation, which would be used in the final classification of the data points.

It is possible that the ambiguous nature of the classification tasks in the genetic datasets used in this thesis proved to be unsuitable for the RF algorithm, as many of the points will be crossed over in the manner seen near the decision boundary in figure 2.20. In situations like those seen in figure 2.21, it is clear that RFs are capable of dealing with interactions in the datasets, despite looking at the feature inputs individually.

2.2.8. Comparing the performance of the Support Vector Machine and Random Forest Algorithms on an ambiguous dataset

At this point, it was only an assumption that the RF algorithm would not perform as well. Therefore a comparison trial between an SVM and an RF had to be made, using the same dataset for each model.

The algorithms used were the same as those illustrated in the examples so far, with the same parameters: SVM: linear kernel, $\mathbf{C} = 1$, RF: using ten bootstrapped decision trees. The data points were very similar to those seen earlier in figure 2.18, but on this occasion, more data points were added as it is important for the bootstrapping procedure of the RF to have enough data, and the desire was to have a fair comparison between these two algorithms.

In order to make the comparison, a shuffled Cross-Validation procedure was used, with 75/25% train/validation splits, with a total of 100 splits. These splits of the data were the same for both algorithms to allow for ease of performance comparison. This was achieved by setting a *random-seed* number for the splits in the Python program. The ROC performance metric was used, and the results provided a total of 200 metric scores; 100 for each algorithm. A box plot of these distributions can be seen in figure 2.22.

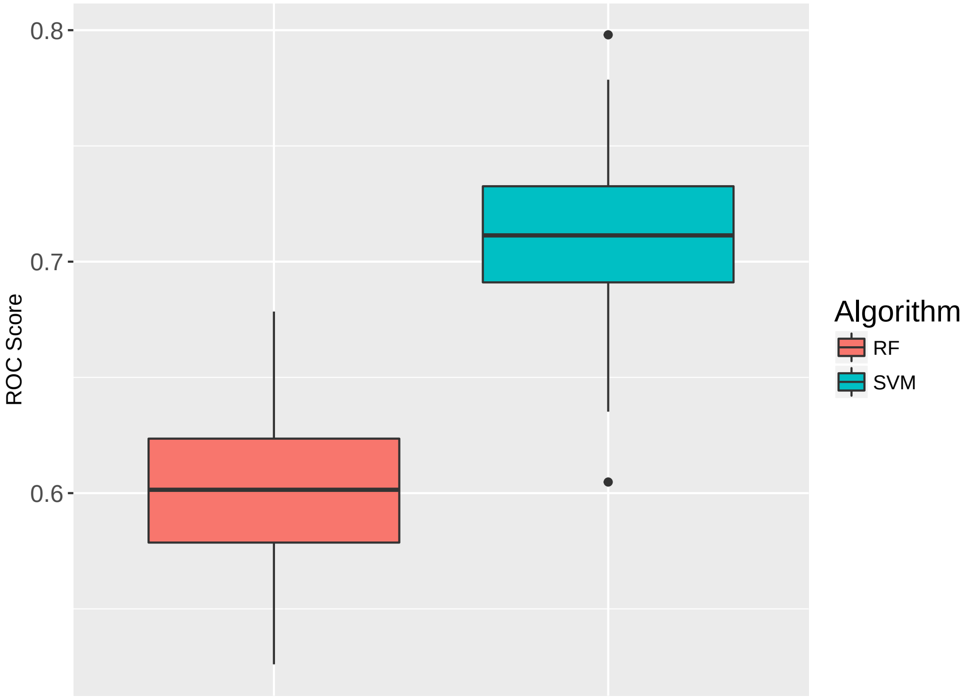


Figure 2.22.: Box plot showing the comparison between the distribution of ROC scores for both the SVM and RF algorithms. The SVM is showing superior performance on this ambiguous data.

This box plot shows that the linear SVM is displaying superior performance over the RF, given the information in this dataset, and confirms the assumption that the type of decision boundary drawn by the RF does not cope well with overlapping data points from both classes. There is one particular caveat to bear in mind with these results: an extensive CV and hyper-parameter tuning procedure was not performed, but the large difference in performance, coupled with the failed attempts to get the RF algorithm working with the main genetic datasets meant that the conclusion was made that the SVM was more suitable for use in this thesis. The next section describes how an SVM was used on a large genetic dataset with clear boundaries between the different class categories, and was performed as a "proof-of-concept" that both the algorithm, and the software, could deal with genotyped data.

2.3. A Proof of Concept example of the effectiveness of a Support Vector Machine, predicting population from genetic structure

Before the Support Vector Machine algorithm was put into effect on the schizophrenia datasets, an example experiment was carried out to ensure that this algorithm was indeed capable of dealing with genotyped information from multiple SNPs. This initial experiment is referred to as a *Proof of Concept*, because it was expected that the performance on this would, in fact, be 100% accuracy. The dataset used in this experiment was taken from the International HapMap project, which was introduced in chapter 1 (Gibbs et al., 2003). The original raw dataset contained information on 76,035 SNPs from 483 individuals. The pruning method in PLINK was used to isolate out those not in LD with each other. This was done using the following parameters:

• Window size	500 Kilo-Bases (KBs)
• Movement of window	250 KBs (resulting in 50% overlap)
• r^2 threshold	0.2

This procedure left a total of 66,396 SNPs remaining. This data then had to be prepared to represent the inputs needed for the SVM. The information that was required was the minor-allele count for each SNP. For this study, the identity of the minor-allele was identified by looking at the distribution of the alleles in the dataset provided to PLINK, and denoting the less frequent allele as the minor-allele, with no additional information. This is perfectly viable in this study, but for the additional studies carried out in the experimental chapters that make use of GWAS data, it is important that the identified minor-allele is the same as that reported in the GWAS. The procedure to do this is described in the respective methods sections for all of the experiments. This recoding was performed by using the `--recodeA` command in PLINK, and results in a file containing a large matrix of numbers in the set $\{0, 1, 2\}$, as this represents the possible number of minor-alleles at each SNP. This data could then be used in the SVMs.

The target, or outcome, variable in this case was not a case/control status for any disorder, but the ethnic population group that the individuals belong to. One of the achievements of the HapMap project was to identify genomic differences that occur between different populations, so the samples were carefully selected to avoid any possible effects of inter-racial reproduction that could have occurred. This task was therefore a *multi-class* problem for the SVMs to solve, and not a binary one. As such, the AUC metric could not be used, but as can be seen in the results, this was not necessary. Out of all of the different ethnic populations used in the HapMap study, the following five were used here. These can be seen in table 2.1.

Table 2.1.: Table showing the different populations used in the positive example study, together with their respective HapMap codes, and the number of samples used.

Population	**HapMap Code**	**Number of Samples**
Han Chinese in Beijing, China	CHB	84
Japanese in Tokyo, Japan	JPT	86
Utah residents with Northern and Western European Ancestry	CEU	165
Yoruba in Ibadan, Nigeria	YRI	113
Gujarati Indians in Houston, Texas	GIH	88

For this study, the information from the first two populations were combined to make one group of 170 samples, with the code **ASI**: Asian. This was done because these populations were very similar to each other, and this can be seen in figure 2.23. The main task here was to provide the SVM algorithm with a problem that is known to be easy to solve, to see how it performs. If these two populations had been kept together, then it might not have been possible to assign a lack of performance to any possible insufficiency in the algorithm for this type of data.

In order to gain a meaningful graphical representation of the distribution of the genetic variations in this dataset, a Principal Component Analysis (PCA) was carried out to identify the two top components that explain most of the variation seen in the data. A plot of how the samples map onto these two components can be seen in figure 2.23.

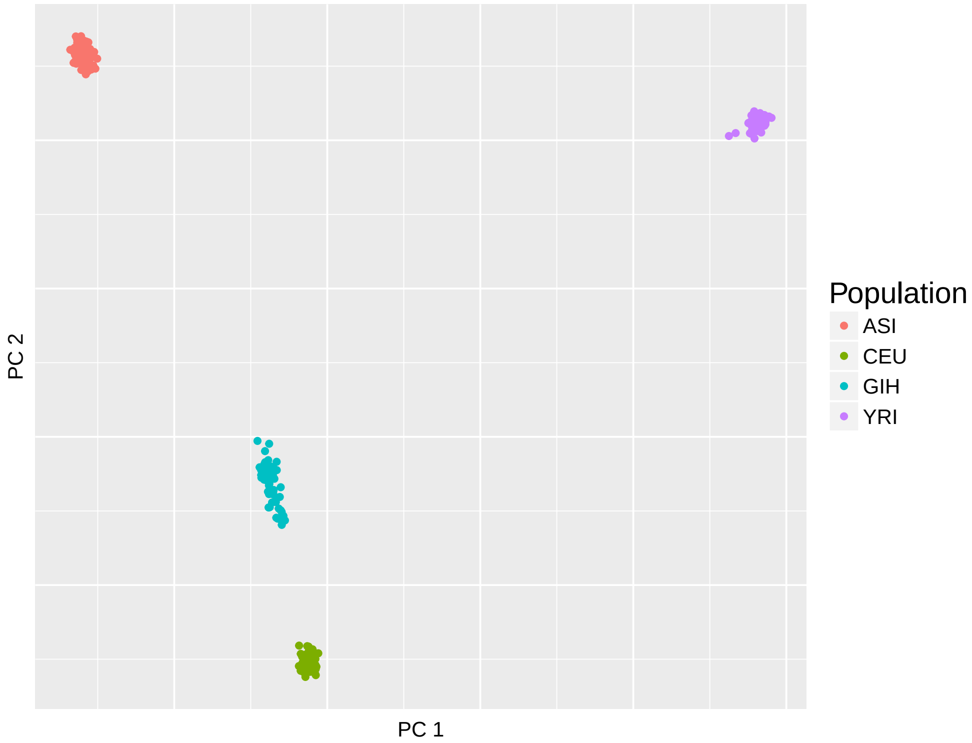


Figure 2.23.: Scatter plot showing the distribution of the different populations across the two top principal components

This plot shows that the four different populations really do cluster into separate regions, even when the information from all 66,396 SNPs have been combined into the top two principal components. It also shows the need to combine the two Asian populations, as they show a complete overlap in two dimensions. It is possible that taking into account information from higher order principal components would separate these 2, but this exercise was not relevant to the task at hand.

The next stage was to fit the SVM to this problem, to see if it could identify differences between the populations. In this task, a linear SVM was built, and provided with all of the 66,396 SNP features. The only parameter for a linear kernel is **C**, and this was kept at the *scikit-learn* default value of 1. No adjustments for the different class sizes were made. The inputs were scaled as part of a pipelined procedure, so the mean and standard deviations from the training points were used to transform those data in the validation set at each split.

For the CV procedure, a stratified shuffle split was made, with splits of 75/25% train/test proportions, for a total of ten iterations. As was mentioned earlier, as this is a multi-class problem, the AUC metric could not be used, instead, the proportion of correct answers was given for the held out samples at each split. The slight variation in the algorithm that is made with a multi-class problem is that the model is built several times using a "one-vs-rest" method, that iterates through all the labels and classifies them as "positive" for their respective turn.

The outcome of these trials was that a linear SVM obtained 100% prediction accuracy at identifying all of the four separate class labels. It is important to note that, while the image in figure 2.23 would suggest that these are not linearly separable, as they resemble the clustering of patterns seen in the XOR example, it must be remembered that this image is only showing the two top principal components. These data points must be completely linearly separable in their original dimension space in order for the algorithm to display the perfect performance that it did.

The result showed that the SVM algorithm is capable of working with minor-allele information from genotyped datasets, and was suitable for use in the experimental chapters.

2.4. How a Polygenic Score is calculated, and examples of its use in recent studies

As was mentioned in chapter 1, a polygenic score is a single score that is given to each sample that captures the level of genetic risk that is provided from all of the possible common risk variants. As the use of this score features heavily in the experimental chapters to provide a baseline of comparison for the performance of the SVMs, a detailed example of how it is created is provided here.

The score is made from two pieces of information: the number of alleles per variant of interest, and the Log Odds Ratio of these variants, which is that output from a GWAS study. Genotyping is used to identify how many reference alleles are seen at each variant. As was mentioned earlier, a Single Nucleotide Polymorphism (SNP) represents a point in DNA where one of the nucleotide bases has switched to another. In the vast majority of occasions, the switch moves from the nucleotide that is more commonly seen in a human population, to one other nucleotide. These differences that are seen are referred to as alleles. The switch to one other nucleotide only is referred to as a bi-allelic SNP. There are examples of tri-allelic SNPs, where a nucleotide can switch to one of two other nucleotides, but these are much rarer (Casci, 2010; Hodgkinson and Eyre-Walker, 2010) and do not feature in any of the datasets used in this thesis. For these alleles, there are three possible values for each variant: $\{0, 1, 2\}$. This is due to the double stranded DNA in humans which means that a variant can be seen on either none, both, or one of strands. A GWAS is then performed to see if any of these less common alleles are seen either more, or less, frequently in cases and controls. One method that is used is called “allelic-association”, and counts how many of the less common allele is seen in both cases and controls. From this information, an Odds-Ratio (OR) can be calculated. As this is a ratio score, it is unevenly distributed around 1, so a common approach is to take the natural logarithm of this taken to give a score that is equally distributed around zero. This is referred to as the Log Odds Ratio (LOR), as is measure of the association that the minor-allele has with a disorder. The p-value of this procedure is the information that is provided for the clumping routine for LD correction.

It is the LOR that is used when making a polygenic score. It is extremely important to note that this value must have been calculated from samples that *do not* feature in the samples for whom the polygenic score is being made. This is because the polygenic score is often used in predictive modelling, and it is not feasible to use the case/control status of a group of people being used to influence a metric, that it turn will be used to try and predict the same statuses of those individuals. Therefore, the GWAS results from different samples for the same disorder must be obtained in order to proceed. The dataset used to make the LORs is often called the *discovery set* and the dataset containing the samples for whom the polygenic score is being made is called the *testing set*.

Once all the information is available, the LORs are then used to perform a clump procedure to gain a list of SNPs that are not in LD with each other. The allele counts of these SNPs are then weighted by their respective LOR values, and then the average of these values is taken for each sample. This is the polygenic score for each person, which represents their overall risk score for a certain disorder of interest. One small note to make here is that the least common allele seen at a variant can differ depending on different samples. A GWAS will assign the OR to the least common allele in the dataset, so it is important to be sure which is being used as the so-called *reference allele* for each variant.

The use of this score featured in the risk-profiling section of the Psychiatric Genetics Consortium - 2 (PGC-2) study (Ripke et al, 2014). As this was a large consortia study, these scores could be assigned to the different datasets that featured in the final analysis, by using the information from all the other datasets as the discovery set. The results from the trials using the same dataset of treatment-resistant schizophrenia samples showed results with an ROC score of 0.704. This finding is replicated in the results in chapter 3.

2.5. Review of Machine Learning uses in genetic and neuroscience research

The increased ability to perform more computationally intensive tasks has provided the possibilities for machine learning algorithms to be implemented with real world scientific datasets and experiments. This concluding section will outline some examples of how they have been used in the fields of neuroscience, genetics, and oncology research; across a variety of different scientific methods including brain imaging, micro-array, and GWAS datasets.

One of the areas that has probably seen the most amount of developments and implementations using machine learning techniques has been brain imaging. This has been the case across a whole range of different imaging modalities including Structural Magnetic Resonance Imaging (s-MRI), Functional Magnetic Resonance Imaging (f-MRI), Diffusion Tensor Imaging (DTI), Positron Emission Topography (PET), Electroencephalography (EEG), and Magnetoencephalography (MEG). One of the reasons why brain imaging research might have been so receptive to the use of machine learning techniques, according to a review paper published by Lemm et al. (2011), is their ability to extract information from very large and high dimensional data, with very low signal-to-noise ratios. The authors, however, stressed that these methods must be used in the correct way, and mention a series of pitfalls that can occur in the analysis (shown in Table 1 of the article on p. 397), of which those relevant to the analysis have been taken account in this thesis: dealing with class imbalances correctly, and ensuring that proper CV processes are observed, such as making sure that any pre-processing of the data is included in these, and not performed on a global scale.

A review of studies using SVMs for brain imaging was carried out by Orrù et al. in 2012, together with a description of the possibilities and challenges for future research. A brief summary of some of the key studies is provided here. Using regional volumetric maps from s-MRI and PET data on 15 patients of Mild Cognitive Impairment (MCI) (a common precursor to dementia seen in Alzheimer's disease), and 15 healthy controls, Fan et al. (2008b) found that linear SVMs were capable of classifying between the two groups with an AUC result of 0.978. This was a very impressive score, but relied on an *a priori* selection of brain regions in advance. Without this information, the score fell to 0.875, and no classification was achieved on the PET data (AUC 0.5). In a study of Major Depression (MD), Hahn et al. (2011) using a linear SVM to classify between cases and controls using f-MRI data with stimuli of neutral faces, large rewards and safety clues, found an accuracy of around 87% and concluded that the results had identified brain regions involved with processing emotional stimuli. It must be noted however that the main focus of this paper was to use another predictive algorithm called a Gaussian Process, but the results were compared with the SVM and it was concluded that the predictive performance was the same. In a study looking at the white matter structure in the brain using DTI, a perfect classification was made between Alzheimers patients and healthy controls (20/25 samples respectively) using a linear SVM (Graña et al., 2011).

There were two particular studies that focussed on topics relevant to the material in this thesis, both looking at data of neuronal firing patterns in schizophrenia. Khodayari-Rostamabad et al. (2010) used a machine learning algorithm called Kernel Based Partial Least Squares Regression (Rosipal and Krämer, 2006), a method that used a similar kernel approach to that in a Radial-Basis Function kernel, to make predictions about responses to clozapine treatment from EEG recordings made prior to medication. The results showed that the algorithm had an 85% similarity with reports made by clinical experts blind to any of the imaging results. In a study using word based, and non-word-based tasks with MEG recording, Ince et al. (2008) made use of the coefficients assigned to the input features from the MEG recordings to isolate those that were important for the classification task, and found that they were able to discriminate between 15 schizophrenia patients and 23 healthy controls with an accuracy level of 91.9%. However, they did not make it clear which metric was used in the evaluation, or if there were any corrections made for the imbalances between the class numbers. It is still a high level of performance, but these factors could have inflated it slightly.

In the field of cancer research, there was interest from the early 2000s to make use of the capabilities of the SVM algorithm. Cruz and Wishart (2006) stressed how the method could be used to help with the prediction and prognostic analysis of the disease, and poses a distinct advantage concerning interpretability when compared with earlier "black box" machine learning algorithms like artificial neural networks. But they also expressed concern that it is too easy and common for studies not to make use of the full CV procedure when building their models. They did mention one study which they believe did take the correct approach: Listgarten et al. (2004) used information from 98 SNPs from 45 genes relevant to the development of breast cancer and found that an SVM using a polynomial kernel of degree 2 (also termed a *quadratic* kernel) on a subset of three SNPs had a specificity score of 83% (±7%). This score represents how well the model protects against false negatives; the sensitivity result, showing how many of the true positive were found, was 53% (±2%), so over half of these were identified. It is interesting to contrast the number of SNPs used in these older studies with the numbers used in modern analysis (as can be seen in the later experimental chapters) which make use of hundreds of thousands, or even millions, of SNPs. The quadratic kernel is another method of mapping the data into higher dimensional space, but specifically looking for pairwise interactions. There were attempts to use different versions of the polynomial kernel in chapters 3 and 5, but they were not shown to deliver high performance.

Another popular area in which machine learning has played a role due to the high dimensionality of the datasets is in the field of the analysis of gene expression using micro-array techniques or epigenetic profiling using methods like Chip-Seq. A PubMed search carried out on the 4th of October 2016 for the term "Support Vector Machine Microarray" returned 341 results. The most recent one to feature, by Xi et al. (2016) used a combination of an SVM using an RBF kernel together with an iterative feature elimination method to find the optimum sets of genes for analysis called a "particle swarm optimisation algorithm". This was carried out for microarray data of five subtypes of cancer: leukemia, prostate, colon, lung, and lymphoma. The results showed a perfect classification technique, but seeing as this is a very recent publication, it would be very interesting to see the reception of these results in the community. Fernández and Miranda-Saavedra (2012) also used an SVM with an RBF kernel to try and predict gene enhancers from chromatin modification. Another optimisation method called a "genetic algorithm" that finds results by iterating through procedures and seeing which parameters or features tend to "survive" the process, was used to find the optimum window size for the epigenetic profiles. The authors called the combination of the two algorithms "ChromaGenSVM". One of the successful results from this study was to use ChipSeq data on 38 distinct histone methylation and acetylation marks in human CD4$^+$ T cells, and identified five epigenetic marks that predicted active enhancers, of which 90% were backed up with experimental results.

2.5.1. Critical review of machine learning in GWAS studies

This last section of the literature review will focus on examples of how machine learning has been used to process data from GWAS trials. These studies provide the type of high dimensional data that the algorithms have been designed to tackle, hence this review is included as all of the data in this thesis comes from GWAS analyses. It will focus on three different methods of examining the effects of genetic interactions in high dimensional data: exhaustive search techniques, Random Forests and Support Vector Machines.

In addition, examples of different software packages that have been developed to tackle machine learning, either in general, or with special consideration to GWAS data, will be provided.

Exhaustive search methods

As the name of this type of algorithm suggests, exhaustive search methods attempt to explicitly look at the effects of interactions on a case/control outcome being measured. However, because this can involve the analysis of a massive number of dimensions, it can quickly become unfeasible when using traditional statistical methods like logistic regression. Instead, it is possible to carry out extensive pre-processing of the data, in order to capture the signal from interactions before they are entered into the models. This section will focus on one such method called Multifactor Dimensionality Reduction (MDR), which was first implemented by Ritchie et al. (2001) when looking at the involvement of interactions of SNPs in estrogen metabolism genes in the development of sporadic breast cancer.

The pre-processing steps involve looking at the different possible combinations of bi-allelic SNPs and assigning these to either *high-risk* or *low-risk* status. A short example, taken from Ritchie et al. (2001) is provided here. A bi-allelic SNP can be represented in the following manner: {**aa**, **aA**, **AA**}, showing the two homozygous states together with the heterozygous state. Another SNP for the interaction comparison can be represented in a very similar manner: {**bb**, **bB**, **BB**}, and from this it follows that there are nine different combinations of values that any individual could have for these two SNPs. MDR seeks to reduce these nine values down to essentially a one dimensional, binary value to indicate high or low risk status. It does this by looking at the counts of cases and controls for each combination, and setting them as high-risk if this exceeds a set threshold. This threshold is the main parameter that needs to be tuned in MDR. In order to prevent overfitting, these relative counts of cases and controls are not calculated on the whole dataset, but analysed using k-fold Cross-Validation. The original article suggests that the value of k should be set to ten, so that the process be repeated ten times with different splits to avoid the overfitting. This process is repeated for every different interaction combination of interest, and then a subset of those found to be most informative can be used to predict new data samples.

As is apparent from the description, this process can be incredibly lengthy, and requires access to a high level of computational power, especially when looking at higher-order interactions. However, the authors state in their study that this method allowed them to identify a particular interaction pattern of four SNPs in a dataset that had been shown to contain no statistically significant main-effects. The same research team developed some software, which they made freely available two years later (Hahn et al., 2003). In this article however, they acknowledge the computational demands of their method, and state that the software can take a maximum of 500 input features from a maximum sample size of 4,000, and only analyse up to four-way interactions; but of course, it must be remembered that this software was developed over a decade ago and would have been limited to the power of computers at that time. Indeed, a recent study has shown the implementation of this technique was used to identify a series of pairwise combinations of SNPs involved in the development of bladder cancer in two different GWAS datasets consisting of 4,759 samples with 620,901 SNPs and 2,228 samples with 1,072,820 SNPs, where every possible pairwise combination was explicitly processed (Cheng et al., 2016).

It must be stated that the name given to this algorithm could, in fact, be slightly misleading. It gives the impression that the number of *dimensions* to be analysed is reduced, but what is actually occurring is that it is the number of different *values* of a combination of SNPs that is being reduced. Admittedly, the number of possible interactions being considered can be reduced by taking a subset of the most informative interactions from the pre-processing, but the authors state that the example given above has reduced the analysis from nine dimensions down to one, which is not correct. The actual reduction in dimensions comes from using this process to perform various feature selections to select the SNPs used for further analysis. It has also been pointed out that this original method is dependent on there being a balanced sample with equal numbers of cases and controls (Uppu et al., 2016). However, there have been successful attempts at dealing with such cases. By using simulated datasets with unbalanced ratios of cases and controls, Velez et al. (2007) showed that this problem could be addressed by either upsampling or downsampling the minority or majority class (case or control) respectively, or by setting the high-risk/low-risk threshold based on taking the average of the *sensitivity* and *specificity* of the models built during the CV procedure, instead of just looking at the proportions of cases to controls for each combination of SNP values.

While the technique of MDR has been shown to identify relationships between interactions of SNPs and phenotypes, it seems as though this method is best designed to be used when the genetic signal does not consist of so many higher order interactions, and when the important interactions themselves have a large effect on the disorder outcome. This is most likely not the case with psychiatric genetic disorders, where a series of highly complex interactions between variants in and around genes involved in many different pathways could be taking place, each contributing to a small effect on the overall outcome of very complex and varied phenotypes. It is for this reason that more promising research in this area will come from algorithms that take these issues into account, like RFs and SVMs.

Random-Forests and Support Vector Machines in GWAS analysis

Random Forests and Support Vector Machines have been included into one section, as there are examples in the scientific literature where both have been used in the same study, so it makes sense to describe how these have developed together for use in GWAS analysis. There have been several cases when both of these algorithms have been used in the field of genetic research, but this review will focus on a subset of studies. These include attempts to fit models to real data, and also cases where simulated data has been used in order to identify how to develop the algorithms, and how to use them to identify important features.

A large comparison of different machine learning algorithms, where their performance was contrasted with that of the polygenic score, was carried out by Pirooznia et al. (2012). The work focussed on building a predictive model to discriminate between cases of bipolar disorder and healthy controls. The authors provide a rich description of the datasets used, and the procedures that they employed to identify the SNPs of interest for the analysis. They stated that they took the data for the training set from the Bipolar Genome Studies (BiGS) consortium (Mahon et al., 2009) and the test data from the Wellcome Trust Case/Control Consortium (WTCCC) (Burton et al., 2007). Care was taken to ensure that proper Quality-Control (QC) measures were put in place for both the SNPs and samples that were selected, paying attention to minor allele frequency ($< 1\%$), missing data rate ($> 5\%$) and Hardy-Weinburg equilibrium p-values ($< 10^{-6}$). Principle components analyses using the Eigenstrat method (Price et al., 2006) were used to control for the effects of population stratification, and to remove any outliers based on ethnicity. They also used the PLINK software (Purcell et al., 2007) to clump the SNPs to control for the effects of Linkage Disequilibrium (a very similar procedure is used to select SNPs of interest in chapter 3). They also prepared two different datasets for analysis: one looking across the whole genome, and another focussing on areas of the genome falling in the region of approximately 13,000 genes that have been shown to be widely expressed in brain regions (Johnson et al., 2009). In addition to the datasets with separate SNP information, they also used PLINK to prepare polygenic scores, and analysed these using logistic regression.

The software used to build all of the machine learning models in the comparison study was the WEKA data mining and machine learning software (Hall et al., 2009), developed by the machine learning group of the University of Waikato in New Zealand. This software is very popular, and the paper from 2009 has been cited over 13,800 times[1]. The researchers

[1] As of 8th April 2017

built a series of different learning algorithms, including the linear and RBF SVM, and the RF; in addition to another method called a Bayesian Network (BN). Their results showed that, out of the machine learning algorithms, the BN performed best, but that none of the algorithms outperformed the polygenic score, on both of the datasets. The smaller dataset, focussing on the brain expressed genes, also showed a drop in performance over that of the whole genome. The authors suggested that these algorithms were therefore not suitable for any diagnostic or clinical use, at the time of writing.

However, there are some points of concern with this conclusion: for all of the implementations of the machine learning algorithms, there were no attempts to perform any tuning of the hyper-parameters, and only the default values in the WEKA software were used. It is unfortunate that no attempts to tune the models were performed, especially after such care and attention was given to the preparation of the genetic datasets. These algorithms can be very sensitive to the values given to these parameters; a particular example is that the value of **C** in an SVM should be reduced when dealing with the type of noisy and ambiguous data that can often occur in the study of psychiatric genetics. As will be seen in chapter 3, it is often the far lower values of **C** that perform best. The default value in WEKA for this parameter is 1. This is an unfortunate conclusion to make of, what is otherwise, a very carefully controlled study. If a properly planned parameter tuning exercise had been performed, it is possible that the authors could have seen improved results in the machine learning when compared with the polygenic score.

SVMs have also been used for the analysis of GWAS data in the field of diabetes research, both for Type 1 diabetes (T1D) and Type 2 diabetes (T2D). A good example of this is a study by Wei et al. (2009), who used this algorithm with the Radial-Basis Function (RBF) kernel to include the effects of interactions. They decided to focus on T1D because it has more genetic heritability that T2D, which has much more influence from epigenetic and environmental factors, a difference that they stated is $\sim 90\%$ to $\sim 50\%$. The researchers focussed on two different datasets for this analysis: one where the information from many variants across the genome was considered, and another where the focus was on a few dozen SNPs, which had been previously shown to display high association with the disorder. Their main focus was on looking at a greater number of SNPs, as they stated that other previous studies had focussed only on those variants where there had been a more significant signal, identified by the GWAS. In fact, they criticise some of the attempts to use an SVM in studies looking at T2D, as they only used about 20 SNPs.

The first experiment described made use of a single dataset from an Affymetrix chip with 1,963 cases and 1,480 controls, and focussed on looking at the ROC scores from five-fold CV within this dataset. There are a couple of points that need to be mentioned about how the researchers selected the subset of SNPs used in the model. Firstly, no matter which variants were chosen after the LD-clumping and filtering procedure, they always added an additional 45 SNPs, which were identified as known risk markers in an earlier study by Barrett et al. (2009), many of which appear in the Major Histocompatibility Complex (MHC) region on chromosome 6. Secondly, the selection of the SNPs was built into the CV procedure as part of pipeline, by carrying out a GWAS on the training data and getting the p-values to select the SNPs under different thresholds. The inclusion of the feature selection process in the CV pipeline ensured that none of the information from the test samples played any role in the building of the models. By using this method, they found that the best p-value threshold was 1×10^{-5}, which consisted of between 399 - 443 SNPs in the different folds, and returned a high ROC score of ~ 0.9. Other findings from these trials found that the majority of the signal came from the MHC, in line with previous findings in the field. When the SNPs from this region were removed, the performance of the models dropped considerably (ROC ~ 0.64). However, they also noticed that not including the other SNPs had a detrimental effect on the models. In fact, one of the key points of the article is that including a greater number of SNPs, and not just focussing on the tens of sites that had previously been implicated in T1D, resulted in better predictive models; a finding from which they conclude that these SNPs that do not show as much main-effect association with disorders can make a contribution when used in machine learning algorithms.

The researchers did not just look at the performances of the SVMs on that first dataset; they also built the models with a logistic regression model for comparison, which performed at a similarly high level for the CV procedures on the first dataset. Further examinations were performed to see if the models could also generalise to two different, independent datasets; one using the same Affymetrix chip, and the other using samples genotyped on Illumina chips. They note that the data from the Illumina chip had to have more SNP permutation performed, as it did not contain raw information on the same SNPs that were used to build in each model. The results from these trials showed that the performance levels did drop slightly for the SVM models, but remained high (~ 0.86). The logistic regression model, on the other hand, only kept the higher levels of performance for the Affymetrix data. The authors concluded from this that the SVMs models were more robust and generalisable to new datasets.

In addition to building these predictive models on T1D, they also looked into the specificity of these models by examining if sufferers of other disorders would be incorrectly classed. These disorders were bipolar disorder, coronary heart disease, Crohn's disease, hypertension, Rheumatoid Arthritis (RA) and T2D. The only one of these that showed itself to get misclassified for T1D in the models was RA. The authors stated that, while this could be a confusing result, it is not unexpected, seeing as RA has been shown to have risk loci present in the MHC as well. However, there is a slight concern with this explanation: as the model for T1D lost most predictive power when the MHC SNPs were removed, it could be the case that they had primarily built a model that identifies MHC risk disorders. However, this is a minor criticism, as the test of specificity to a particular disorder when compared with others is a valuable piece of analysis, and is not often featured in studies.

This study by Wei et al. (2009) presents a very thorough approach to using SVMs on GWAS data. Care was taken to ensure that valid models, with clear separation of training and test data, were built, and two separate independent datasets were also used to check the reliability and robustness of these. The only point lacking in the article is that there is no information given to which hyperparameters were used in the model tuning process. As they used the RBF kernel, information on both $\mathbf{C}$ and $\boldsymbol{\gamma}$ should really have been provided.

It is not just real-life GWAS datasets that have been the focus of machine learning studies, there are also many examples of how researchers have created simulated datasets to convey different types of information and relationships between both inputs and disorders and within the input features. This is an efficient technique to identify how different algorithms behave under a variety of different circumstances; to find their relative strengths and weaknesses. In the field of GWAS, this has been particularly prevalent in Random Forests.

One particular recent study of RFs by Szymczak et al. (2016) looked at both simulated and real data. This study describes a new piece of software that the authors wrote, which plays a role in the selection of SNPs for RF models called *r2VIM*. As was mentioned earlier, the acronym VIM stands for Variable Importance Measures, and refers to the level of importance that is given to each feature (SNPs in the case of GWAS), and is one of the areas where the RF algorithm has strength over the SVM, as they can only assign similar measures to features in the case of linear models.

However, the value assigned to these VIMs can be prone to fluctuation under different circumstances. Szymczak et al. (2016) talked about how they can change with the different hyper-parameter values that are used in the model-building, and other studies have highlighted how these can be detrimentally impacted by correlations that occur between the input features, especially when the features are conveying small individual effects towards the phenotype target, as is often the case with GWAS studies of complex psychiatric disorders (Nicodemus and Malley, 2009; Nicodemus et al., 2010). Measures to try and correct for these issues involve finding new ways of calculating VIMs from the usual method, which makes use of the Gini-Index, by performing permutation routines of the values within the different predictors. The idea behind this is that the permutation of important input features will result in a more measurable decrease in the predictive power of the model than those predictors that are conveying little information (Strobl et al., 2008).

These permutations can take two different forms: unconditional and conditional. In the unconditional method, the permutations of all of the variables are performed, and then the downstream effects on the predictive performance of the model are assessed. This method is very computationally efficient, but there have been concerns that different results could be observed in correlated predictors, relative to uncorrelated predictors (Nicodemus et al., 2010). The conditional method, proposed by Strobl et al. (2008), seeks to address this concern by finding collections, or different strata, of predictors that are more correlated with each other. This new conditional VIM for a feature is now calculated by seeing how the permutation of its values effect the model when compared with the other predictors in its collection. All of the permutation comparisons take place *within* each of the strata, therefore the importance value attributed to a predictor is made by comparing how its performance is affected by the permutations relative to the other predictors in the strata, and not all of the predictors in the model. This means that each feature's VIM score is now *conditional* on those other correlated variables. A further calculation on either of these VIMs is to either perform *scaling* or not. Scaled VIMs are calculated by effectively taking the z-score of each VIM in relation to the other features by dividing each by the standard deviation of all VIM values (Nicodemus et al., 2010).

In their study of calculating these different VIMs on simulated data, Nicodemus et al. (2010) stated that the unconditional, unscaled VIMs are a computationally feasible choice for use in large datasets like those seen in GWAS. The results also showed that, instead of an inflation of importance being given to any predictors that were correlated with each other, the opposite effect was observed. The scores were actually lower for these predictors relative to the uncorrelated ones as their signal was being shared during the different splitting procedures during the model building process. The conditional VIMs were also able to identify the signal from the uncorrelated predictors, but the authors concluded that the computational requirements were not feasible for use in large datasets. In summary, when suitable parameters were used, the unconditional VIMs were able to perform well, without the same levels of computational complexity of the conditional VIMs.

Based on these findings, Szymczak et al. (2016) made use of the unconditional and unscaled VIM in their r2VIM algorithm to identify influential SNPs. In this method, several Forest models are built, each using different starting *seeds* for the random selection of participants and features for the different trees. The VIMs for the features are calculated for these different models; the rationale being that the most important and influential features on the outcome will show consistent high VIMs. The lowest average feature VIM is then used as the baseline. The features are then selected by using the *factor* level for each feature: $\frac{featureVIM}{baselineVIM}$. The threshold of this factor value is the main parameter to be tuned in this method. The models were built using the *Random Jungle* software (Schwarz et al., 2010), which was designed to apply Random Forests to large scale GWAS data, and includes functionality to create the permutation based VIM scores. This software has recently been superseded by a new R-package under the name of *Ranger* (Wright and Ziegler, 2015), which uses C++ libraries to optimise the Random Jungle algorithm and apply it to large scale datasets.

In their study, Szymczak et al. (2016) used both simulated and real datasets. First, the simulated data was used to measure the performance of their feature selection method, and then the models were assessed on their application on the real data. In order to add extra rigour to their simulation procedures, two different trials using different simulation software were used. The first study used a program called GWAsimulator (Li and Li, 2008), with information on haplotypes from 381 Europeans from the 1000 genomes study (1000 Genomes Project Consortium, 2012), and the second used one called SeqSIMLA2 (Chung et al., 2015). Both of these methods allow for the creation of simulated samples, both cases and controls, based on different minor-allele frequency of SNPs and LD structure.

A minor point of concern in these simulations is that both of them made use of only nine causal SNPs, which is too few when considering complex disorders like schizophrenia. However, the researchers must be credited for included a null-simulation where the phenotype bore no relation to any of the simulated SNP features. This showed to be a very effective manner at identifying what level of threshold for the VIM ratio factor should be used in order to prevent the identification of false positives.

For the real GWAS data, two different and contrasting datasets were used. The first was selected as it had contained a strong signal related to folate and B12 vitamin metabolism in adults, and the second was related to age-related eye disease and was chosen as it is contained a weak signal. The authors stated that this was chosen to be "negative control data", with the primary aim to check that the r2VIM did not return any false positives. From the perspective of a researcher in the field of psychiatric genetics, it would have been of more interest if this dataset had been used to see how well it could detect those weak effects, as this is the common situation in complex disorders; but this cannot be stated as a criticism of the study, as protecting against Type I errors is clearly of great importance.

The results showed that the method could protect against reporting false positives in both the real and simulated datasets, and did identify all of the nine causal simulated SNPs when a large enough sample size was used. However, the r2VIM method did not perform quite as well as a logistic regression model applied to the same data. While this is a slightly disappointing result, it must be remembered that these simulations were made using only main-effects in the data, which logistic regression is designed to identify. Another study by the same research group, presented at the Pacific Symposium of Biocomputing (Holzinger et al., 2015), applied the r2VIM method to simulated data making use of interactions, and these showed very different results where the method greatly outperformed the logistic regression model at finding the causal SNPs. However, the simulations used only a pair of interacting SNPs, both with high levels of minor-allele frequency. It would be very interesting to see if this model is soon applied to the type of data seen in psychiatric disorders, with many interactions of weak effects.

All of the simulation studies mentioned in this sections have shown that making simulations can provide vital information as to the performance and behaviour of the algorithms, and the use of simulated phenotypes can be seen in chapter 4. However, care must be taken to ensure that these simulations do not deliver overly inflated performance levels to any model building procedure. An example of this can be seen in a study by Aguiar-Pulido et al. (2010) where they looked at SNPs in the HTR2A and DRD3 genes to build an SVM classifier on 260 schizophrenia patients and 354 controls. When using the data from these samples, the models achieved an ROC score of 0.649; but when new simulated samples, made using the HAP-SAMPLE algorithm (Wright et al., 2007) were included, this score increased to 0.942. This level of performance is far higher than anything that has been observed in other schizophrenia studies, so it is almost certain that the simulation was introducing some bias into the samples that was easily detected by the classifier.

2.6. Machine Learning Summary

This chapter has given a thorough background to the algorithms and performance metrics used throughout this thesis. A justification of the use of Support Vector Machines has been provided, as well as a positive experimental example on population data, showing that the SVMs are capable of dealing with genetic data containing information on thousands of SNPs. Examples have also been given of how these methods have been used in previous research. The next chapter provides the first attempt at building SVM classification models to predict case/control status in treatment-resistant schizophrenia.

3. Using Individual Single Nucleotide Polymorphism scores as the inputs to the Support Vector Machines

3.1. Introduction

This experimental chapter describes the first attempt in this thesis at using machine learning techniques for the analysis of Genome-Wide Association Study (GWAS) data. Throughout this chapter, the input values to the models will be referred to as *features*. A more detailed description of these features will be provided in the methods section. The main intention was to compare the predictive performances of models built using the Support Vector Machine (SVM) algorithm (with both linear, and non-linear kernels) with a more traditional risk-profile model, using the polygenic scores. This score was initially described in chapter 2 and is essentially a linear combination of the individual Single Nucleotide Polymorphism (SNP) signals from across the genome, which can then be assessed using a single-feature Logistic Regression (LR) model. In contrast, the machine learning approach will use the minor allele count information that each of the selected SNPs provides separately. One of the main points of interest in this chapter was to see if the SVMs are capable of matching, or improving, on the performance of the polygenic score using only the information about the minor-allele counts.

While using an LR model of the polygenic score is a simpler and more parsimonious model, it has one particular weakness in that it cannot account for any possible interactions between the polymorphisms, which could contribute a possible risk towards developing the disorder that is greater than the linear sum of their individual contributions. By using the various kernel methods that were outlined in chapter 2, an attempt at finding any of these complex interactions can be made. This is a topic that needs to be addressed. As mentioned in chapter 1, a study by Hemani et al. (2014) found that patterns of gene expression were seen to be associated with the effects of epistasis between polymorphisms, both short range (*cis*) and long range (*trans*) in the genome.

While the kernel methods used here do pose a problem for later interpretation of results, they provide an opportunity to search for possible effects of interaction without specifying these as features into the model. Any attempt to do so would only result in a massive increase in dimensionality of the input features.

Background to the current study: CLOZUK and PGC-2

All of the datasets used in this chapter were taken from a study by Hamshere et al. (2013) at Cardiff University, which carried out a GWAS comparing sufferers of treatment-resistant schizophrenia receiving Clozapine medication with control samples that were largely taken from the Wellcome Trust Case/Control Consortium (WTCCC) (Burton et al., 2007). This dataset is called the *CLOZUK* dataset. This dataset was one of many which featured in the recent Psychiatric Genetics Consortium - 2 (PGC-2) study (Ripke et al, 2014) which was described in the Introduction chapter.

The work in this chapter was based on the trials using the polygenic score for the risk profile scoring section of the PGC-2 study. The description of how the polygenic score is made was provided in section 2.4 in chapter 2. The GWAS Log Odds Ratio (LOR) results that were used to perform the clump procedure and weight the reference allele counts were calculated from all the samples in the PGC-2 study with European ancestry minus those samples that were used for the CLOZUK study, and another performed in Cardiff University called the Cardiff COGnition in Schizophrenia (COGS) study (by JTR. Walters). It was important that these samples were held out from this calculation because it is the case/control status of the samples that is used to create the LOR score, so it is not feasible to allow one dataset to influence a metric that would then be used in a predictive model of the same data as this could very possibly lead to erroneously high performance. The additional samples from the Cardiff COGS study were removed to be certain of excluding anyone that could have been related to another sample, due to the close proximity of people in the study recruitment areas.

In the PGC-2 study, the procedure of building the polygenic score was repeated using different thresholds of significance for the index SNPs. The results of these can be seen in Supplementary Table 7 (Ripke et al, 2014), and show that the optimum p-value threshold for the CLOZUK dataset was 0.05, therefore this was the value used in this study. All of the case information was obtained via collaboration with the pharmaceutical company Novartis, the manufacturer of the branded and proprietary form of clozapine, *Clozaril.* Blood samples were acquired from the patients receiving clozapine, who were suffering from treatment-resistant schizophrenia according to the reports completed by the treating psychiatrists. All samples were genotyped by the Broad Institute. The data for the controls came from the WTCCC studies (taken from the 1958 British Birth Cohort, the UK Blood Service Control Group and the UK National Blood Transfusion Service), and they were not screened for psychiatric illness.

Descriptions of genotyping chips used

All of the case/control information were genotyped on two different arrays, the samples for both the cases and controls had been carefully prepared to ensure that they were population matched to try and minimise the confounding effects of population stratification. The original number of samples in the dataset for the different chips were: Illumina 1M (I1M) - 7,763, Illumina Omni Express (OMNI) - 4,233; however, 143 samples had to be removed due to evidence of them being closely related to other samples. This was determined by them having an Identity By Descent (IBD) value of 0.9 or greater. The numbers shown in the list below are those that were used in the different datasets for this chapter, and all remaining experimental chapters.

- I1M - 7,731 total samples
  - 3,446 Cases
  - 4,285 Controls
- OMNI - 4,122 total samples
  - 2,108 Cases
  - 2,014 Controls

Of note, there are some differences between these figures and those reported in the table 1 in the supplementary material of the study (p. 21). The values listed there are: I1M chip 3,426 cases and 4,085 controls; OMNI chip 2,105 cases and 1,975 controls. The reason for this difference is that additional samples will have been removed due to them having IBD values greater than 0.9 with individuals *in other datasets* that were included in the PGC-2 consortium. This can often occur due to the geographical locations of areas used for schizophrenia recruitment.

For the risk-profiling section in the PGC-2 study, the information from the two datasets were combined together. However, in this study, as the machine learning models were given access to the individual information from each SNP, it was decided that the datasets should be initially kept apart, in case there were any confounding effects of the different chips that could have had a detrimental effect on the model. By doing this, not only was there an attempt to control for the differences between the chips, but it also allowed one of the chip's data to be used as a separate test set to evaluate whether a successful model would be transferable and robust to information from different sources. As the I1M sample size was larger than the OMNI sample size, it was chosen as the main training set in this study. The OMNI chip data was used for two tests: firstly to see if the machine learning models would show good performance on different data, and then if this proved to be the case, all of the data could be combined together to make a larger dataset to see if building the models with more samples would result in better predictive performance.

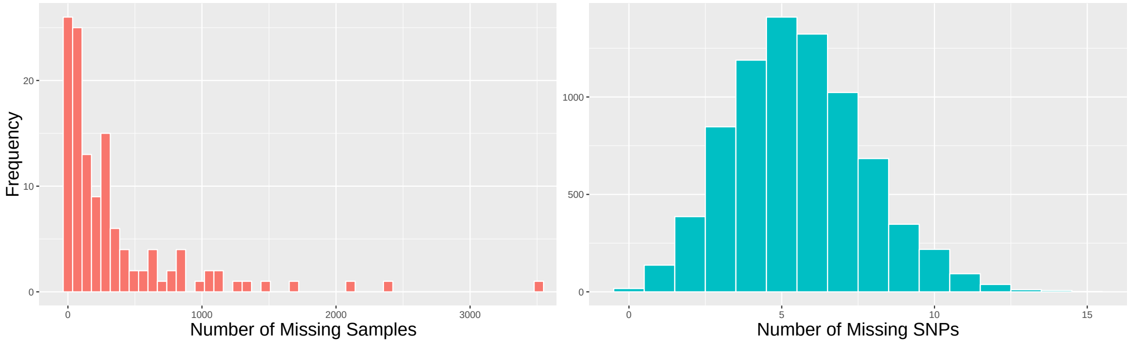
The I1M dataset has fewer cases than controls; as discussed in chapter 2, unequal class sizes can cause problems for machine learning algorithms. However, in the *scikit-learn* package, there is a procedure to help correct for any bias. The model can accept a `class_weight = 'balanced'` parameter, which effectively changes the cost penalty given to getting a classification wrong, which is inversely proportional to the size of each class.

3.1.1. Data preparation and dataset descriptions

Due to the added complexity that the machine learning models introduce, it was imperative that the data inputs to the models was managed correctly. There were two different datasets used: a smaller set consisting of only those SNPs found to be GWAS significant in the PGC-2 study, and a larger one that was created by performing a PLINK clumping procedure on the SNPs and selecting the index SNPs below a p-value threshold of 0.05 from the GWAS. Throughout this chapter, these two datasets will be referred to as the "GWAS significant dataset" and the "threshold dataset". Ideally, both of these preparations of the data would have come from a dataset that had the same Quality-Control (QC) procedures carried out on them, but unfortunately, this was not possible; the more stringent QC procedure used on the threshold dataset ended up removing 49.6% of the GWAS significant SNPs. A full description of the QC procedures, together with possible caveats are described here. Each of the two datasets are described separately.

The GWAS significant dataset

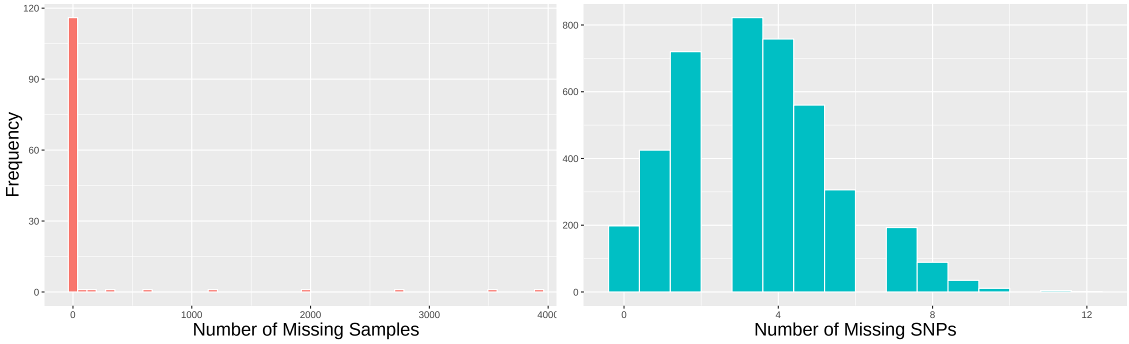
The only QC procedure carried out on the GWAS significant dataset was an INFO score threshold of 0.9 for the SNP imputation. All of the trials carried out in this chapter excluded the sex-chromosomes, and this dataset contained 125 out of the 128 GWAS significant SNPs located in the autosomes (Ripke et al, 2014) (for details on the individual SNPs and their locations, refer to table 2 in the supplementary material of their paper). However, the missingness rates of the SNPs were not considered. The effect of this can be seen for both the I1M and OMNI data in figures 3.1 and 3.2. While the distribution of the number of missing SNPs per sample seems acceptable, it can be seen in the two histograms showing the number of missing samples per SNP that these datasets did not undergo any QC procedure for missingness. In some cases, not all of the SNPs were present for many thousands of samples for both chips. However, while this is clearly not desirable, nothing else could have been done here if these GWAS significant SNPs were to be studied.



(a) Missing data points per SNP

(b) Missing data points per sample

Figure 3.1.: Histograms showing missing data information for the I1M, 125 GWAS SNPs dataset



(a) Missing data points per SNP

(b) Missing data points per sample

Figure 3.2.: Histograms showing missing data information for the OMNI, 125 GWAS SNPs dataset

The threshold dataset

As the best performances in the risk profile scoring section of the PGC-2 study for the CLOZUK dataset were achieved when the GWAS significance threshold was set at 0.05 (Ripke et al, 2014), it was decided that this threshold would be used again here. The initial data from which the SNPs were selected had been through the additional QC steps:

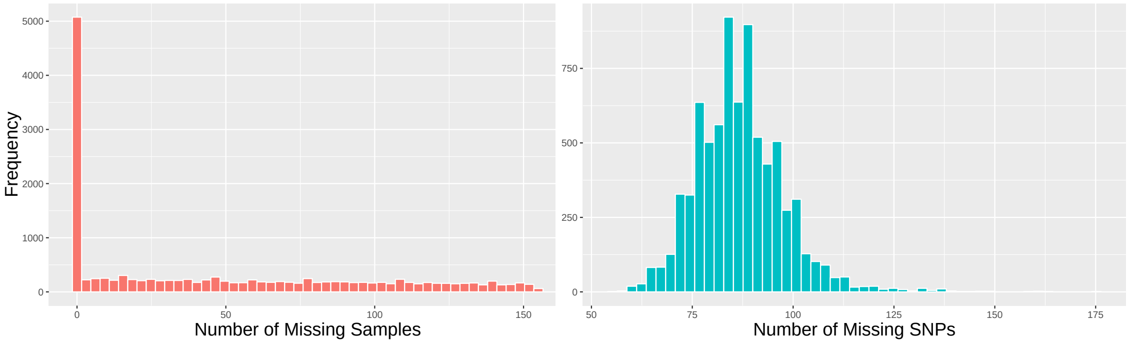
• Genotyping missingness rate	2%
• Minor Allele Frequency	10%
• Hardy-Weinberg Equilibrium (HWE) significance level	1×10^{-4}

These additional QC steps could be carried out in this case as there were no specific SNPs that had to be included in the analysis. In addition, the Major Histocompatibility Complex (MHC) region (chr6: 25-34 Mb) was removed due to the highly correlated nature of the SNPs in this region.

At this point 2,680,814 SNPs remained. However, these were not all independent markers due to the effects of Linkage Disequilibrium (LD), so the data had to be clumped to find the desired index SNPs. The same parameters outlined in PGC-2 were used here:

p1	The significance threshold of the Index SNPs	0.05
kb	Kilo-Bases (KBs) between Index SNPs	500
r2	The r^2 value for the LD threshold	0.1

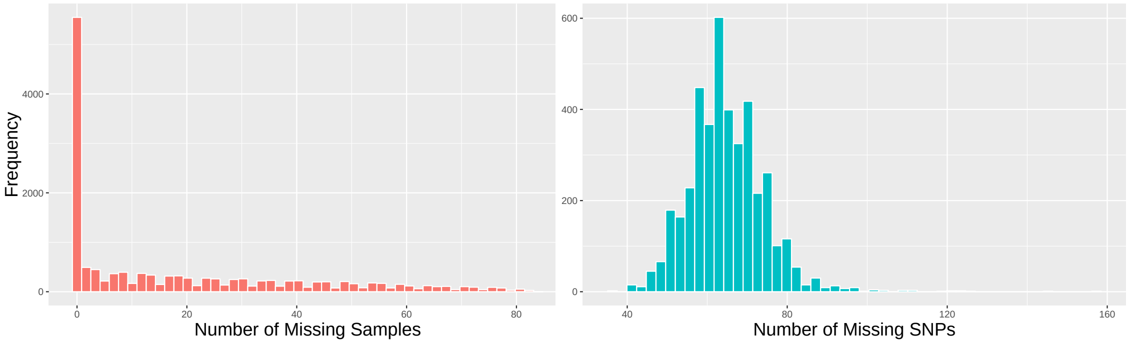
After the clumping procedure, which was carried out in PLINK, a total of 14,462 SNPs remained. As the genotyping missingness rate was far more stringent on this occasion, the distribution for the missing number of samples for each SNP is not so positively skewed as in the previous example. This can be seen in figures 3.3 and 3.4. The maximum number of missing samples per SNP is far lower than was seen in the GWAS significant dataset, and while the distribution for the number of missing SNPs per sample shows higher numbers, this is due to the greatly increased numbers of SNPs being used.



(a) Missing data points per SNP

(b) Missing data points per sample

Figure 3.3.: Histograms showing missing data information for the full I1M dataset



(a) Missing data points per SNP

(b) Missing data points per sample

Figure 3.4.: Histograms showing missing data information for the full OMNI dataset

Reformatting the data for the machine learning models

After selecting the SNPs as described above, the data had to be transformed into a format suitable for being entered in the Python scripts for the machine learning. The information required was how many of the *reference alleles* an individual had for each SNPs. The term reference allele is used to signify which of the mutations was considered in the calculation of the Odds-Ratio in the GWAS statistical tests. In a GWAS study data format, the reference allele is normally assumed to be the least frequent allele seen at each location; in effect, the minor allele. However, in the case of the data used in this study, the GWAS results were calculated from all of the different datasets contributed by the teams in the Schizophrenia Consortium apart from CLOZUK and Cardiff COGS (as this data had to be held out), so the more infrequent alleles seen in the different positions could well have been different in the CLOZUK dataset. In order to control for this, the PLINK command `--recodeA` was used, which can be prefixed with the flag `--referenceallele` and a text file instructing which allele should be used as the reference for each SNP. This results in a flat text file, showing how many of these reference alleles each sample had for all the SNPs.

Data Analysis in Python

All of the data analysis was carried out in Python, using two main packages for machine learning and data processing. The *scikit-learn* package (Pedregosa et al., 2011) is an ongoing development of a variety of different algorithms for a vast array of machine learning requirements, and provides an interface to allow all of the algorithms used throughout this thesis. The *Pandas* package (McKinney, 2011) is another ongoing development that facilitates the application of a wide selection of data processing procedures, often used to prepare and shape the data to make it ready for use in *scikit-learn*.

The output of the recoding procedure results in a file showing (amongst other information) the number of counts of the various reference alleles for the SNPs for each sample in the analysis. This is essentially a large matrix of integers in the set $\{0, 1, 2\}$, because, due to the existence of two chromosomes for each SNP position, each sample must have one of these number of reference alleles at each location. This information was used to create the three main types of input information that were used in this chapter:

Raw Allele Counts A matrix of the integers $\{0, 1, 2\}$ only

Weighted Allele Counts As above, weighted by the LOR values from the GWAS results

Polygenic Score The mean of all the weighted allele counts for each individual

Creating these inputs was done with relative ease using *Pandas*; the information for allele counts could be extracted from the output of the `--recodeA` operation. The weighting was carried out by getting the correct order of the SNPs from the columns of the recode output file, and using this to get the LOR from the GWAS results file. This resulted in a matrix of the allele counts as well as a vector of the respective weight values. In a single operation, these two can be multiplied together to give the final matrix of the weighted values. The polygenic score was then calculated by taking the mean of the weighted values for each sample.

3.1.2. Dealing with Missing Data Points

Due to the nature of how the SVM algorithms work, the missing values had to be replaced. While the *scikit-learn* package does provide functionality to replace missing values, this method was not deemed adequate as it can only replace them with a single value per SNP, using either the mean, the median or the most frequent value. Instead, it was decided that the values should be replaced in a probabilistic manner. This meant that the various proportion of 0s, 1s and 2s occurring in the non-missing data would be preserved. As the data was stored in a *Pandas* dataframe, with the rows representing the samples and the columns representing the SNPs, this was done by shuffling the order of the rows and then replacing the missing values with the nearest non-missing value that occurs in the respective column. After the values were replaced, the rows were then returned to their original order[1]. This ensured that the missing values contained no information about case/control status.

While this replacement procedure had to be carried out in order to use the algorithms, it was still of some concern that this could have had an unknown detrimental effect of the performance levels of the models. It is impossible to know for sure what the full extent of these effects could be when using the SVMs, but it was possible to carry out a comparison of performance for the polygenic score. As this score is the weighted average of the non-missing allele counts per individual, it was possible to create two set of these scores: one from the original data and another from the data with the missing values replaced. The results of this can be seen in section 3.3.1.

[1] For any desired attempts at reproduction, this procedure was carried out with the following command (making use of the random number modules within *numpy*) "`df = df.reindex(numpy.random.permutation(df.index)).ffill().bfill().sort()`"

In addition to replacing the missing values, a quick analysis was carried out to ensure that the distributions of the missing values were equal for both the cases and the controls. If there is a noticeable difference between the two, then this can lead to the results being biased in favour of higher predictive performance. While the replacement of the missing values could well ameliorate this effect, it was still desirable to check for any differential missingness that could have been present in the data. As can be seen in table 3.1, in three out of the four different datasets, this rate was below 2%. The only case where it was higher was in the data for the OMNI chip for the GWAS significant dataset. This was slightly unfortunate as it was these particular SNPs that were needed in this instance, and in another situation, the SNPs with the higher rate of missingness would have been dropped from the analysis. However, since the main work for the training of the model was done in the I1M chip data, with the OMNI data used only for final testing, this did not affect the study.

Table 3.1.: The percentages of differential missingness for both chips in the two datasets. All but the data for the OMNI chip for the 125 GWAS study had a rate less than 2%.

Dataset	**Chip**	**Maximum Missingness (%)**
125 GWAS	I1M	1.8
	OMNI	4
Full dataset	I1M	1.2
	OMNI	1.4

3.2. Methods - Outline of Chapter and Aims of the Study

3.2.1. Description of Algorithms

As this chapter represents the first attempts at using the SVMs on the genotyped data, a wide array of different procedures were carried out. The intention here was to not make any assumptions about which methods would either perform better, or encounter various problems in their execution, but to experiment with a variety of different methods and algorithms in order to find a method that would be most suitable for use in the later experiments and chapters. Due to the sheer scale of the differences in sizes between the two datasets, alternative approaches had to be employed for use in the different situations. While the GWAS significant dataset was comparatively simple in that there were only 125 input features to be considered, the threshold dataset had the problem of having vastly more features than the number of samples present.

The main outline for both datasets was as follows:

- Carry out the training and testing of SVMs with the following kernels:
    1. Linear
    2. Polynomial
    3. Radial-Basis Function (RBF)
- Use the following variations of the data for the inputs to the SVMs:
    1. **Use the minor allele counts only**
    2. **Use the *weighted* minor allele counts**
- Finally - compare the performances of these models against that of the polygenic score using predictive performance metrics

Scaling of the input data

As mentioned in chapter 2, it is often considered good practice to perform *scaling* of the features before entering them into any algorithm that maps inputs into a common space like an SVM. This term is commonly used in machine learning, and essentially means normalising the data to have 0 mean with a variance of 1, so this was carried out for all the features across all the samples. However, it might not be the best approach in this situation. The allele counts are in the set $\{0, 1, 2\}$ and are therefore always positive or 0. The sign of the LOR, on the other hand, contains a great deal of information. If a variant is seen more often in cases than in controls, then the sign of the LOR will be positive as the odds-ratio will have been greater than 1. The converse is true for variants seen more frequently in controls. In effect, one can think of alleles with positive LORs as *damaging* and those with negative LORs as *protective.* This information is vital to the formation of the polygenic score in order to gain any meaningful value of risk. Any *z-score* scaling procedure performed on the features will erase the information given by the sign.

While the case can be made that the scaling would be detrimental to the performance of the model, the presence of zero values in the data were of concern when using the kernel based models. This was relevant when using the polynomial kernel as it makes use of multiplicative terms between the features. The presence of zero values would cause any interaction information to be lost. For clarity, the example of a quadratic kernel (a polynomial kernel with degree 2) is given. Equation 3.1 shows a breakdown of the kernel transformation being carried out on a pair of two dimensional vectors $\boldsymbol{x}$ and $\boldsymbol{z}$. In these equations, the subscript number denote the elements of each vector. In essence, this kernel is effectively calculating the square of the dot product between the two vectors. If any of the values in these vectors is 0, then it would knock out at least two of the three terms shown on the far right side of the equation, and therefore would not be a true representation of any interactions occurring.

$$\langle \boldsymbol{x}, \boldsymbol{z} \rangle^2 = (x_1 z_1 + x_2 z_2)^2 = x_1^2 z_1^2 + x_2^2 z_2^2 + 2 x_1 x_2 z_1 z_2 \tag{3.1}$$

In comparison, the RBF kernel does not suffer from this feature of the data. If the same two vectors $\boldsymbol{x}$ and $\boldsymbol{z}$ are considered, the expansion of the kernel mapping can be seen in equation 3.2. Here it can be seen that, unlike in the quadratic kernel in equation 3.1, the elements of each vector are subtracted from each other. The presence of any zero values would therefore not cause the same deleterious effect on any calculation.

$$\exp(-\gamma \|\boldsymbol{x} - \boldsymbol{z}\|^2) = \exp(-\gamma \sqrt{(x_1 - z_1)^2 + (x_2 - z_2)^2}) \tag{3.2}$$

Trials of performance using a simple XOR model

In light of the concerns about the scaling of the feature variables, and the presence of zero values in the data, an attempt was made to see if this would have an effect on a simple example where the outcomes of the models should be known in advance. The case in question is that of the eXclusive-OR (XOR) model, which was described in chapter 2 and recapped briefly here. A quick schematic to illustrate this is shown in example 3.3. All of the rows in the matrices represent the four different types of inputs, and the desired output, with 1 classified as a “hit” and 0 as a “miss”. As can be seen, the output should only be a 1 if the input is exclusively one input *OR* the other.

$$\underbrace{\begin{pmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \\ 1 & 1 \end{pmatrix}}_{\text{Inputs}} \rightarrow \underbrace{\begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}}_{\text{Targets}} \tag{3.3}$$

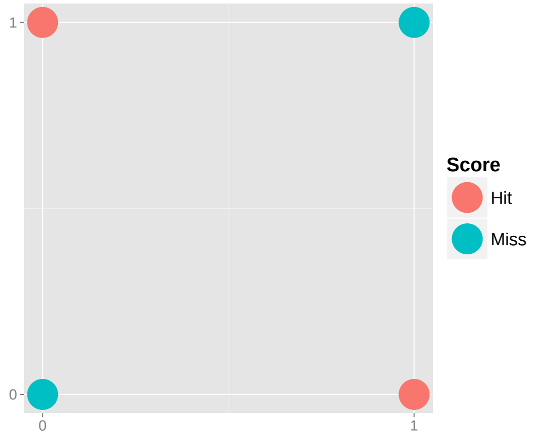


Figure 3.5.: A graphical representation of the unscaled XOR problem.

Figure 3.5 shows that it is not possible to put a single dividing line between these points, hence we would not expect the linear kernel to work, regardless of any scaling. However, the non-linear kernels should be able to cope with this without any problems, so it was of interest to see whether the quadratic kernel could deal with the zero values in the inputs. The results in example 3.4 below show that these suspicions were confirmed; the RBF kernel is delivering the correct answer while the quadratic kernel is not.

$$\begin{aligned} \text{Linear} &\rightarrow \begin{pmatrix}1\\1\\1\\1\end{pmatrix} \\ \text{Quadratic} &\rightarrow \begin{pmatrix}0\\1\\1\\1\end{pmatrix} \\ \text{RBF} &\rightarrow \begin{pmatrix}0\\1\\1\\0\end{pmatrix} \end{aligned} \tag{3.4}$$

When the inputs are scaled, the structure of the problem is retained, but in this case the zero values have been changed to -1, making the values for the features have zero mean and a variance of 1. The inputs are now shown in example 3.5 and figure 3.6.

$$\underbrace{\begin{pmatrix} -1 & -1 \\ -1 & 1 \\ 1 & -1 \\ 1 & 1 \end{pmatrix}}_{\text{Inputs}} \rightarrow \underbrace{\begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}}_{\text{Targets}} \tag{3.5}$$

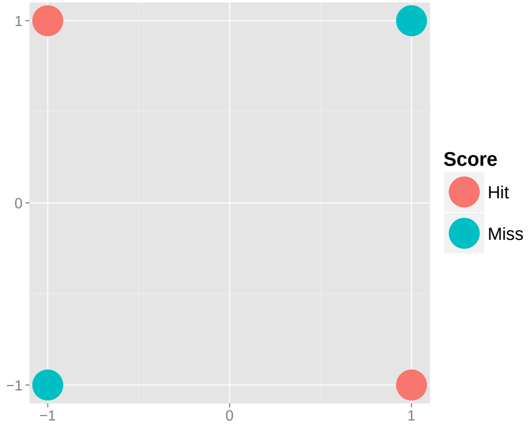


Figure 3.6.: The scaled version of the XOR problem.

Now the results from the both non-linear kernels are performing correctly, as seen in 3.6.

$$
\begin{aligned}
\text{Linear} \rightarrow &\begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \\
\text{Quadratic} \rightarrow &\begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} \\
\text{RBF} \rightarrow &\begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}
\end{aligned}
\tag{3.6}
$$

Due to these findings, it was presumed that the best representation of the patterns in the data would be to use the scaled values in the models.

While it was suspected that the scaling would build better models, the analysis was also performed on the allele-counts, both weighted and non-weighted, just to see how this would affect the performance.

There is a slight caveat with this method for removing zero values in the allele count information, as there is no guarantee that all of the zero values will be removed in every Cross-Validation split of the data. When the scaling was performed across the samples, all of the zero values were removed, but there is no guarantee of this always being the case during the Cross-Validation (CV) procedure. However, as will be seen later in the chapter, and throughout the thesis, the main important information came from the linear and RBF kernels, which are not affected by the zero values in the same way as the polynomial kernels.

Feature Scaling as part of a Pipeline

As stressed before, an important factor to keep in mind when building any form of predictive model is that the data within the CV or test splits should not contain any information from the training data. Therefore, during the scaling it is not desirable to just perform this transformation on the whole dataset before the procedure is run. Instead, the scaling should be carried out using only the data in the training split: this is then used to calculate the mean and variance of that particular subset of the data and subsequently these values are used for any transformation on the CV or test splits. Fortunately, as this is such a common practice within machine learning, the *scikit-learn* libraries contain simple functionality to carry this out by the use of the `Pipeline` function. This allows the different stages of the model building process to be carried out throughout the whole process to achieve the result that is desired.

3.2.2. Hyper-Parameter Search

As mentioned in chapter 2, the best hyper-parameters for each model cannot be calculated analytically beforehand. There is simply no way to be able to accurately predict what is best from the data provided. Therefore, the parameters have to be found, in essence, by trial and error. This involves carrying out a CV procedure by building and testing the models on subsections of the data with different parameter values.

The hyper-parameters that were searched at this initial stage were:

C Represents the "cost" accrued for classifying a training sample incorrectly. Used for all kernels

$\boldsymbol{\gamma}$ Used to adjust the width and shape of the kernels when using non-linear kernels

degree Used to specify the degree of the polynomial kernel, which in turn makes the models look for different types of interaction

The CV procedure carried out in this study worked as follows:

1. The dataset was initially split into training (90%) and test (10%) subsets

2. The training set was then sorted using 4-fold, stratified splits

3. The search through the different parameters was then carried out using the Monte Carlo sampling method mentioned in chapter 2, using exponential probability distributions

4. The following distributions were used for the different hyper-parameters:

    - **C** - expon($\lambda = 1$)

    - $\boldsymbol{\gamma}$ - expon($\lambda = 0.01$)

- **degree** - Values 2, 3, and 4

5. Each selection of hyper-parameters was assessed by taking the mean score across the four folds of the training data.

6. After the best model was chosen, the respective hyper-parameters were then used to train a model using *all* of the training data, and this model was then used to discover the performance on the originally held out, 10% test set.

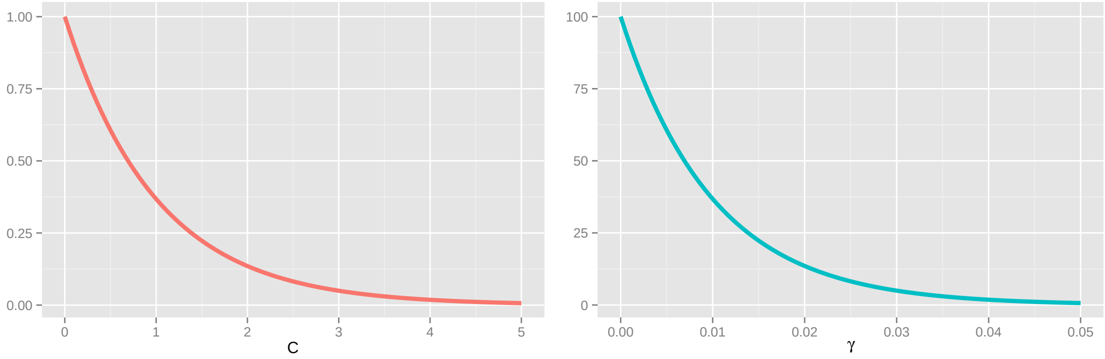
7. The metric used for all of the performance assessment was the Area-Under-the-Curve (AUC) method described in chapter 2, also known as the Receiver Operating Characteristic (ROC) score.

This train/cross-validate/test procedure was run five times for the different types of dataset input. The hyper-parameter search consisted of 100 iterations, each with different samples from the probability distributions. As each selection involved 4-folds of the data, there were a total of 400 models built for every train/test split. Fortunately, the software used was able to handle parallelisation of the program execution.

One very important aspect to point out here is that all of the splits that were made, both for the initial training/test splits, and the stratified 4-fold CV selections, were given the same random number generator seed. This ensured that the splits were always done in exactly the same way for all of the trials across all of the kernels. Also, the Monte Carlo search through the hyper-parameters was given a fixed seed; this ensured that all of the combinations of the parameters would be the same across the trials.

The formula for the exponential distribution is given in equation 3.7. The probability distribution was selected because the lower values (shown on the x axes) have a higher probability of being chosen. Initial trials showed that it was these low values that resulted in better outcomes, and these distributions ensure that not too many trials are spent testing potential sub-optimal parameter values. The distributions for both $\mathbf{C}$ and $\boldsymbol{\gamma}$ are shown in figure 3.7. The values chosen for $\boldsymbol{\lambda}$ for these distributions were adjusted to alter the scale of the parameters, as shown on the x axes.

$$\frac{1}{\lambda}e^{-x/\lambda} \tag{3.7}$$



(a) Exponential distribution of C.

(b) Exponential distribution of γ.

Figure 3.7.: The probability distributions of the **C** and $\boldsymbol{\gamma}$ parameters, note the change of scales in the x-axes.

3.2.3. Choosing the features for the threshold dataset

The rationale for building the threshold dataset with all the index SNPs with a significance level of under 0.05 from the GWAS results was that this was the best performing level for the polygenic score predictive models for the CLOZUK dataset (Ripke et al, 2014). The main interest was to see whether entering all of the features individually into a machine learning model would result in better performance than using the amalgamated polygenic score. However, there was a problem with this strategy as it resulted in 14,462 features for only 7,731 samples. This is a clear case of the dimensionality of the model exceeding the number of available data points. This can pose a particular problem for models using non-linear kernels. As was described in chapter 2, these kernels are already mapping the data points into a higher dimensional space (to where the classes may be linearly separable), and to have more features than samples means that the models will be prone to overfitting to the training data (Noble, 2006). Because of this, not all of the information from the SNPs could be used.

There were a few different ways to deal with this situation. One method involves incorporating a *feature-selection* process into the pipeline of the model building. There are functions in *scikit-learn* to carry this out, and they involve selecting a certain number of best features at every cross-validation fold, based on statistical tests of association with the case/control outcome. The advantage of this method is that it provides the model with access to all of the features. The disadvantage however is that, unless there are a clear sub-group of features that are carrying most of the signal, then different splits of the data could result in a great disparity in the features that are being chosen every time. This makes comparing the performance of any model much more difficult as different information was used to train and test them. The other solution is to make the significance threshold more stringent than 0.05, in order to get fewer features.

When these models were built, both methods were tried, and it was concluded that the second method would be preferable to report here. The feature selection method was found to have very sporadic performance due to the different subsets of SNPs being chosen at each iteration. The material in chapter 5 discusses how the data can be prepared to include more information from a greater number of SNPs, but for the rest of this chapter, only a subset was used.

The next thing to decide was how many features were to be retained in these models. When calculating this, both the train/test (90%/10%) and train/cross-validation splits (75%/25% - due to 4-fold cross-validation) had to be taken into account, as all of these steps reduce the number of samples available to build the model. As shown in equation 3.8, the maximum number of features would be 5,218.

$$7731 \times 0.9 \times 0.75 = 5218.425 \tag{3.8}$$

As this number was the absolute maximum that would be used, it was decided that the desirable number would be rounded down to 5,000. However, as there was not a clear p-value cut-off threshold for exactly 5,000 features (due to many features having equal values), the cut-off was made at 4,998 with a p-value of 7.34×10^{-3}.

3.3. Results

3.3.1. Performance of the polygenic score with and without the missing data imputation

This first analysis was carried out to ensure that the probabilistic imputation method described in section 3.1.2 did not have a dramatic effect on the performance of the models. In order to test this, two different versions of the polygenic score were made: one using the original data with no replacement of the missing data, and the other with the replaced values. The process to test this was done using a built-in function in *scikit-learn* called `cross_val_score`. This method is slightly different and simpler than the full training, CV and testing procedure carried out earlier, and is described briefly here. For this, all of the samples were used, none was held out for testing purposes. Instead several permutations of CV were performed to get a distribution of results. The sequence was as follows:

- A stratified split of 75%/25% was made

- A LR model was trained on the larger set and tested on the other

- No hyper-parameter tuning was necessary as it was only a single-feature model ($C = 1$)

- Each permutation was scored using the ROC metric

- This procedure was carried out 200 times for each of the two datasets

Boxplots of these distributions of results can be seen in figure 3.8. The results show that in the case of the threshold dataset, the imputation had a much smaller effect on the outcome. But in the GWAS significant dataset, the effect of these is quite pronounced. It is possible that this result was seen due to the larger amount of missing data points present. Pairwise related t-tests showed that in both the GWAS significant and threshold dataset, the differences seen when replacing the missing values were in fact highly significant (Full dataset: $t = 2.78$, $p < 0.006$, 125 dataset: $t = 37.77$, $p < 1.035 \times 10^{-92}$). In the smaller dataset, it is clear that that there is a significant difference, but for the threshold dataset, the significance appears to be due to the high number (200) repeats making the t-test extremely sensitive to very small effects sizes.

Because of this observed difference, for the remainder of this chapter, whenever the polygenic score is used, it will always have been created from the data with the missing values replaced to make a fair comparison with the machine learning.

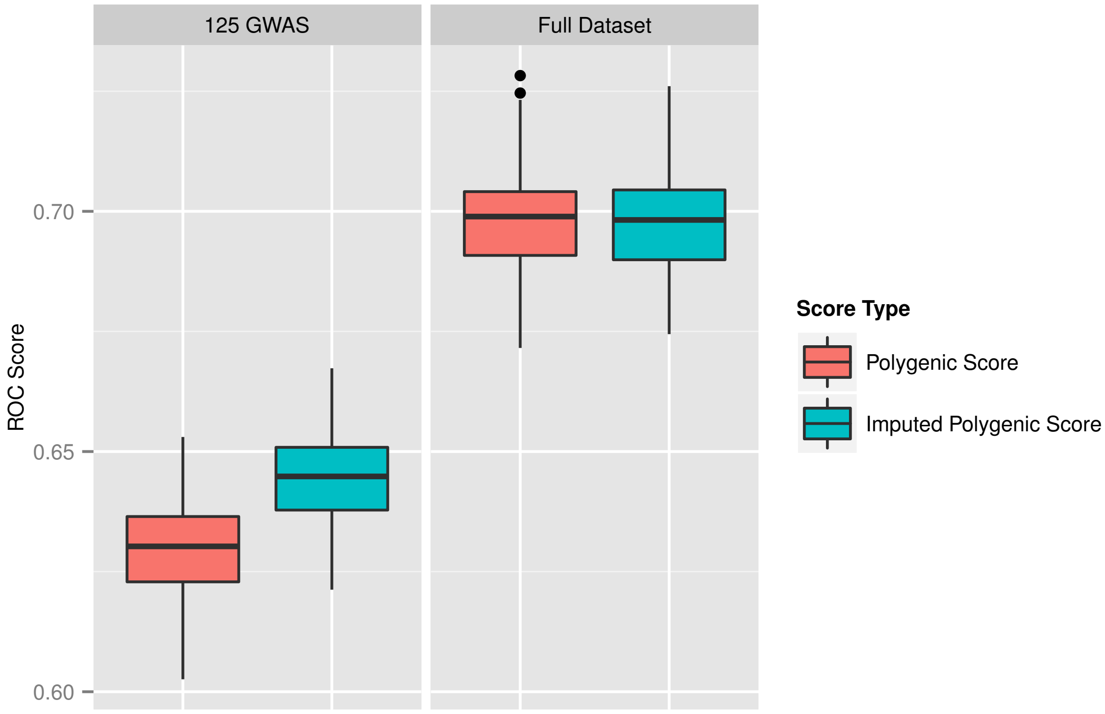


Figure 3.8.: Boxplot showing the distribution of scores from 100 permutations using the polygenic scores from the original data with missing values and the data with the imputed values.

3.3.2. Support Vector Machine Results

As many different variations of algorithms and inputs were tested in this chapter, only the selected findings that were deemed to have been of most interest are shown here. Those results not shown here can be seen in appendix A. For the GWAS significant dataset, the results for the scaled allele counts is the only information shown, but for the threshold dataset, it was not so clear cut which was the best example to use. For this reason, an example of performance for the scaled and unscaled allele counts, as well as the weighted data is shown as the results could lead to different conclusions. However, in the interest of interpreting the results, due to the results seen in the XOR example earlier, more focus was given to the scaled data.

3.3.3. 125 GWAS SNP dataset

The results from the different parameter values for the three kernels can be seen in figures 3.9 to 3.13. In all of these figures, the vertical panels represent the five separate train/test splits of the data described in section 3.2.2, the x axis represents the different values of the parameters chosen and the y axis shows the ROC score achieved. For the polynomial kernels only, the colours of the points represent the different polynomial degrees used. It can be seen that the performance in these models did not match the median level seen for the different permutations of the imputed polygenic score, shown by the centre bar in the blue box plot in the left panel of figure 3.8, although at this stage, it was a considered a possibility that this could be due to the SVM having information from only 90% of the data. The linear kernel shows fairly robust performance levels across the different values of **C**. But there is a much greater variety in performance levels for the non-linear kernels.

The polynomial kernel showed very dissimilar patterns for the different degree levels. Degree 2 performed the worst, with most performance not too far above the chance level of 0.5. Degree 3 performed the best for this kernel, and it seems as though it was most sensitive to the $\boldsymbol{\gamma}$ parameter, with a small peak seen at around 0.01 (see figure 3.11). This peak value however does not surpass the performance of the linear kernel. Degree 4 also performed poorly; in general not as badly as degree 2, but for some hyper-parameter combinations, doing substantially worse than chance levels.

Performance levels in the RBF kernel also varied greatly, with some values looking as if they are marginally improving on the linear kernel, and others performing very poorly.

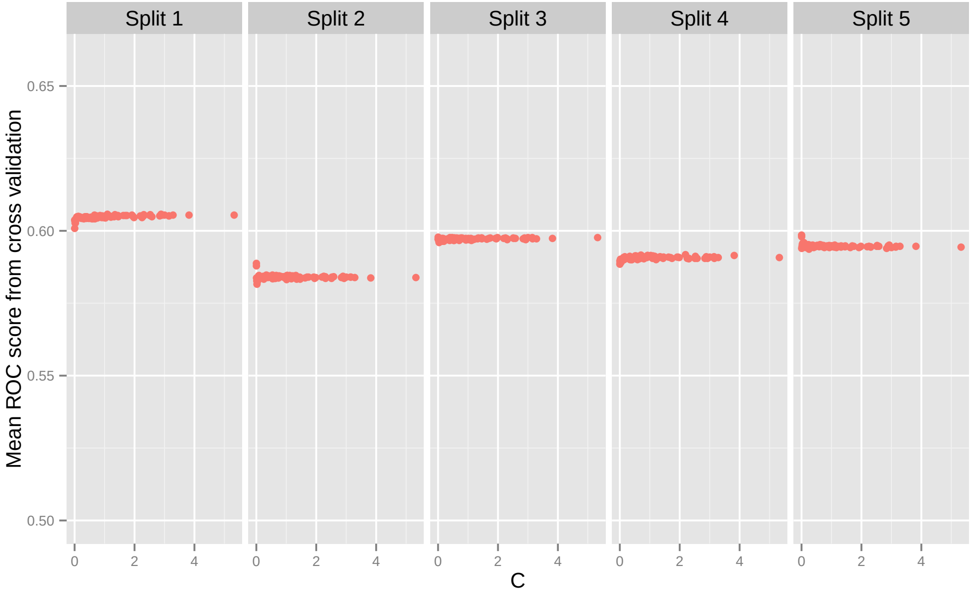


Figure 3.9.: Scatter plots showing the distribution of the ROC values for the different ranges of **C** when using the linear kernel.

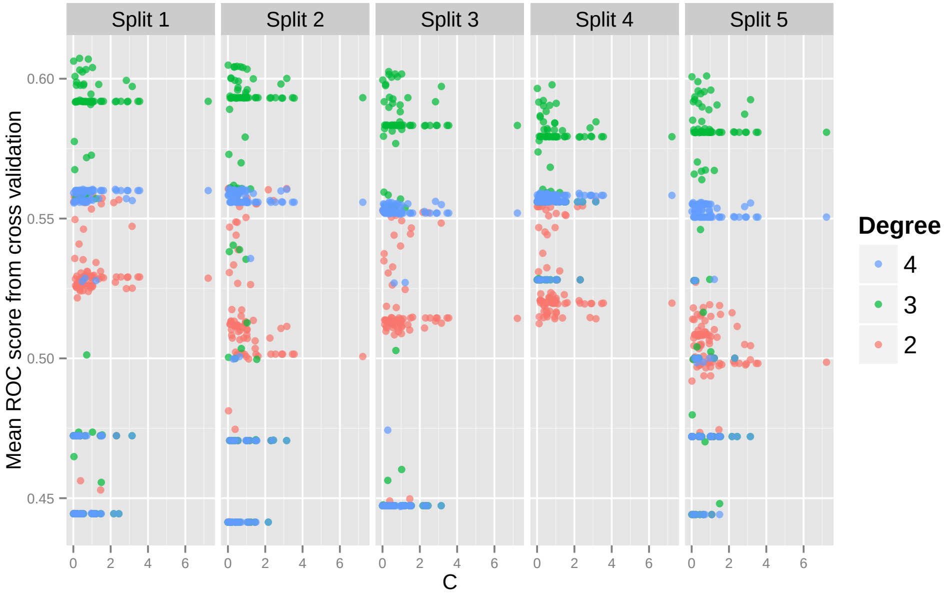


Figure 3.10.: The distribution of scores across the values of **C** for the different degrees of the polynomial kernel.

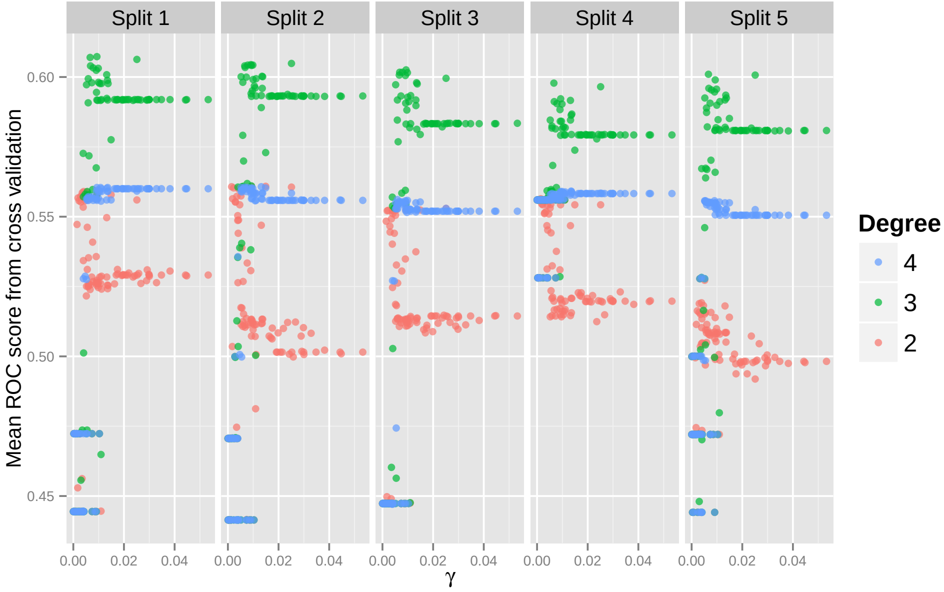


Figure 3.11.: The distribution of scores across the values of $\boldsymbol{\gamma}$ for the different degrees of the polynomial kernel.

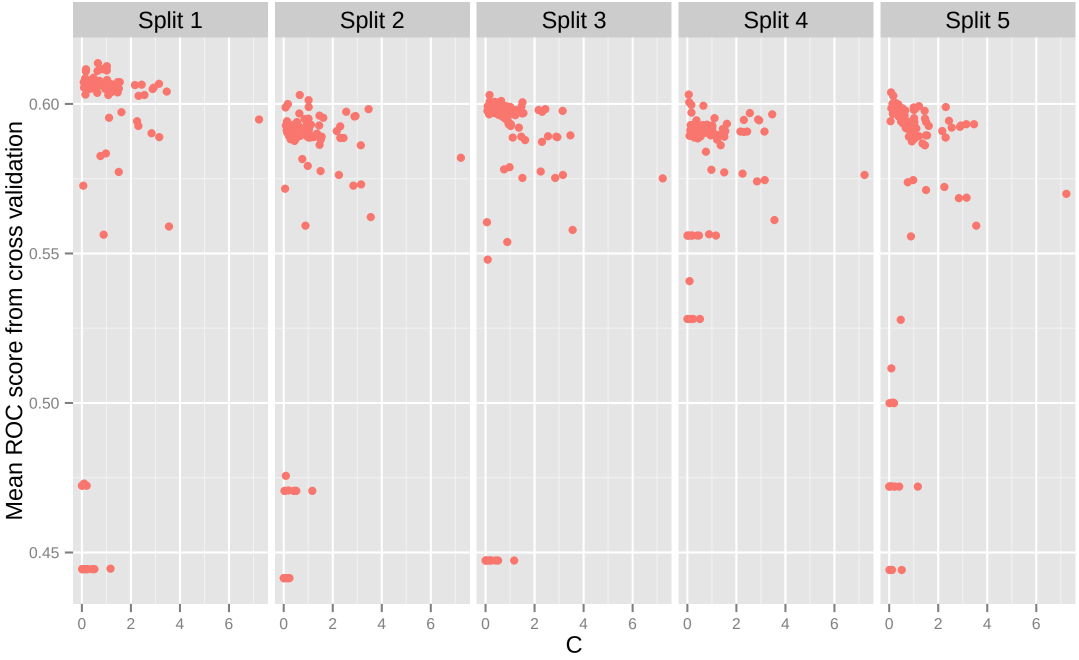


Figure 3.12.: Distributions of scores for RBF kernel for **C**.

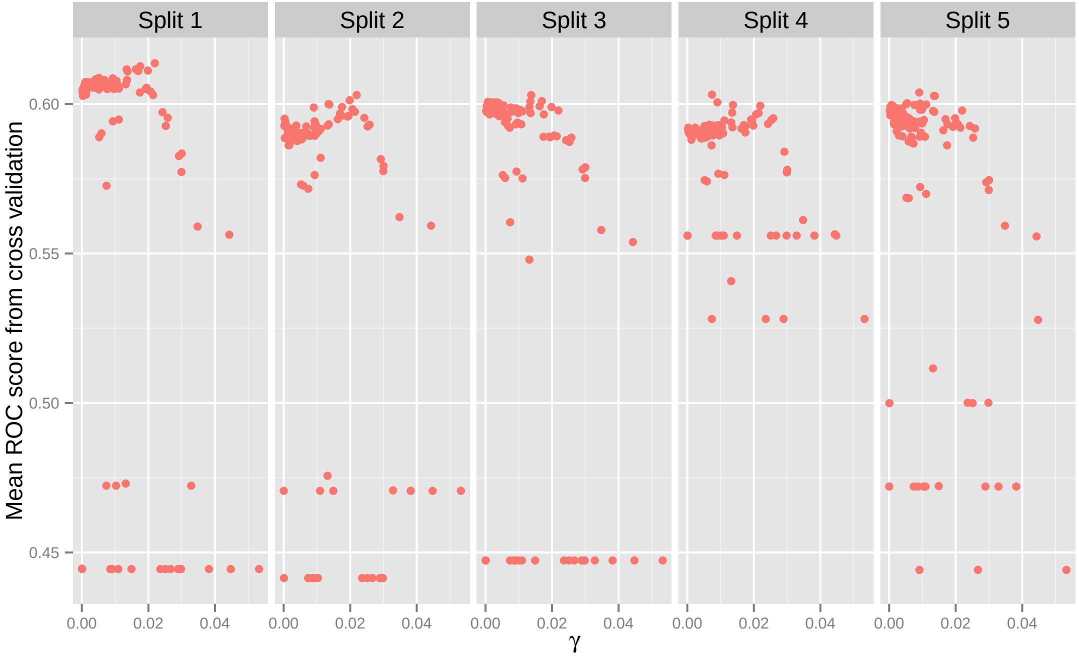


Figure 3.13.: Distributions of scores for RBF kernel for γ.

As there is a large amount of information present in these figures, a summary of the best performing models for each split is shown in table 3.2. This table shows the optimum values of all the hyper-parameters, with the columns showing the kernels and their relevant hyper-parameters, and the rows representing the five different train/test splits of the data. What is of interest is that there are often some erratic patterns for the best values of the hyper-parameters for the different splits of the data. This is probably down to the specific samples that were being used in those particular splits. But what was of more interest here was that the best performance scores of the RBF kernel was actually outperforming those for the linear kernel. This could suggest that maybe there is a minor role of interactions between features. However, it is not feasible to come to that conclusion from just a single, best performing model per split. It must also be taken into account that the performance of the RBF kernel was very sensitive to the hyper-parameter values whereas the linear kernel had more robust performance.

Table 3.2.: All results and hyper-parameters for the best performing models for the 125 GWAS SNPs dataset.

	Linear		Poly-2			Poly-3			Poly-4			RBF		
Split	Score	C	Score	C	γ	Score	C	γ	Score	C	γ	Score	C	γ
1	0.6057	2.87	0.559	0.439	0.004	0.607	0.334	0.009	0.56	2.25	0.009	0.614	0.654	0.0219
2	0.5888	0.002	0.561	0.135	0.009	0.6048	0.015	0.025	0.56	0.088	0.013	0.603	0.654	0.0219
3	0.5978	0.003	0.553	0.015	0.025	0.603	0.334	0.009	0.556	2.84	0.006	0.603	0.162	0.014
4	0.5918	2.19	0.5576	0.213	0.005	0.598	0.8	0.007	0.559	0.167	0.013	0.603	0.057	0.007
5	0.5985	0.002	0.5278	0.12	0.003	0.6	0.8	0.007	0.556	0.543	0.0055	0.604	0.076	0.009

Performance on held out data

As there was some held out test data for each of these five splits, it was of interest to see how the models actually performed on these. These are shown for the five different splits in figure 3.14. In this figure, the vertical panels represent the five train/test splits and the bars represent the single score that was achieved when using the models with the optimum hyper-parameters for each split being used in the models to test the held out test data for each respective split. The different bars represent the different models used, identified by the colour legend. The reassuring results from this verification process is that the same patterns are being seen here that were seen in the hyper-parameter search, which demonstrates that these modest findings are in fact relevant across the whole dataset. But again, these results only show a few points for different sample splits, and as certain kernel perform better for different splits, it is difficult to see which provides the most consistent high performance. To examine this, a new series of experiments had to be carried out.

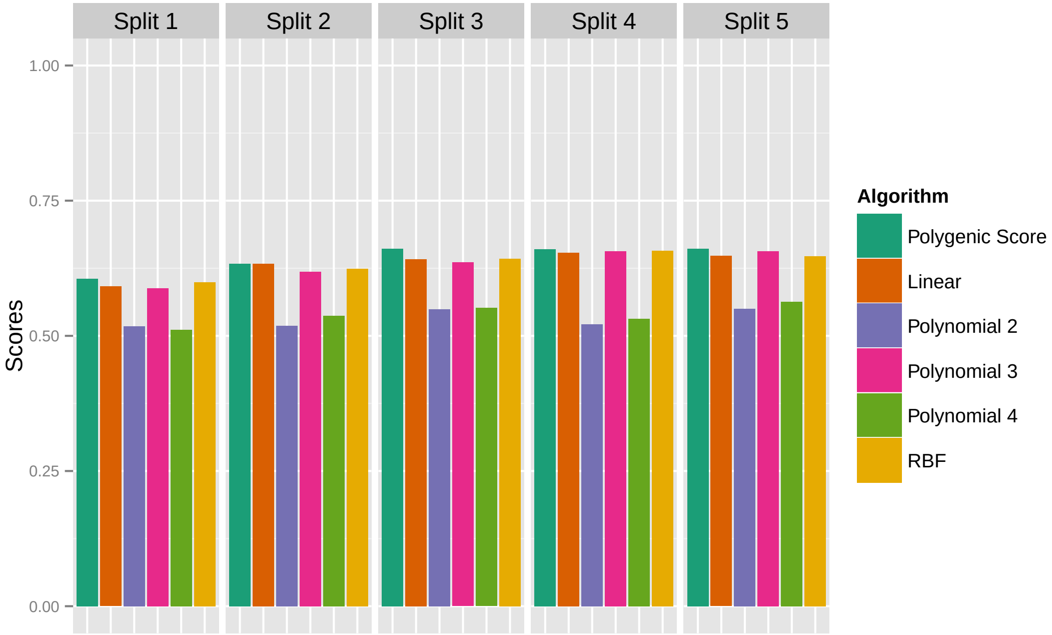


Figure 3.14.: Barplot showing the performance of the best models on the held out test data.

Permutation procedure for all algorithms

Table 3.3 shows comparison of the test performance with the CV procedures. The **Mean CV Score** is the mean of the values seen in the respective columns of table 3.2, and the **Mean test score** is the mean of the peaks of the bars in figure 3.14. As the performance had shown consistent patterns for the kernels across the different splits of the data, but it was difficult to tell which of the algorithms were consistently performing better given the similar performance; a series of permutation procedures were carried out. The main difference with this approach was that there was no focus on any search for the best hyper-parameters (as they had already been found) and there was no splitting of the data to include a final held out test set.

Table 3.3.: The mean results of the scores from the CV procedure and the 10% test data splits for all of the algorithms using the 125 GWAS significant SNPs.

Algorithm	Mean CV Score	Mean test score
Polygenic Score	N/A	0.644
Linear	0.597	0.633
RBF	0.605	0.634
Polynomial-2	0.56	0.53
Polynomial-3	0.615	0.6
Polynomial-4	0.558	0.634

The main reason for holding out some data completely separate from the CV procedure is for verification purposes of the accuracy of predictive models. Given these initial results, it seems unlikely that any reliable model for prediction of schizophrenia will be made from this data, and instead, the models should be used as tool to try and understand what genetic processes could be occurring that drive any increase in predictive power. This means that the 10% test set is not needed, and instead the full datasets were provided to the models. The process for these trials was the same as the missing value check for the polygenic scores described in section 3.3.1: the data was split into train/test splits using 75%/25% proportions, then built and tested on a large number of these splits for each of the different kernels to get distributions of scores for each model.

As the interest was to see the best performance from each kernel, the hyper-parameters were chosen to represent the models that had performed best so far. The parameters chosen to be used can be seen in table 3.4 and were based on where the best peaks of performance were seen in the earlier scatter plots.

Table 3.4.: The hyper-parameters used for the permutation procedure.

Kernel	C	γ
Linear	1	N/A
Polynomial (all degrees)	1	0.01
RBF	0.5	0.02

A random number seed was used to ensure that each kernel was given the same splits of the data, and the procedure was repeated 100 times for each model type. In addition to all of the kernels, the performance of the polygenic score was also recorded. The results can be seen in figure 3.15. Of note, the RBF kernel has been positioned next to the linear results as the performance levels of these were quite similar and this positioning made any comparison of performance easier to visualise.

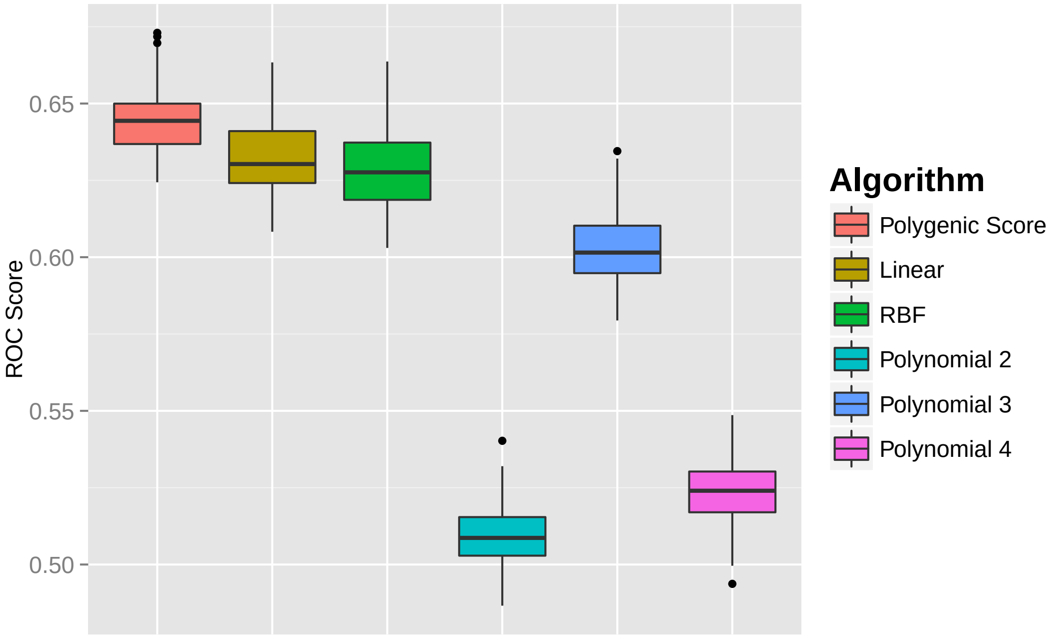


Figure 3.15.: Boxplots showing the distribution of ROC scores for the different model types.

From this box plot showing the distribution of the results, it can be clearly seen that the the RBF kernel is not consistently outperforming the linear kernel, and that the linear kernel is not improving on the performance of the polygenic score model. These differences in performance were statistically significantly when assessed with pairwise t-tests: polygenic score to linear kernel - $t(99) = 17.571, p < 2.2 \times 10^{-16}$; linear kernel to RBF kernel - $t(99) = 9, p < 1.6 \times 10^{-14}$. One other notable outcome from this is that the performance levels have increased appreciably from those seen in the scatter plots of the hyper-parameter search. This is probably due to the fact that the models were being trained and tested on 100% of the data instead of 90% as was done earlier, as on this occasion no data is set aside as test hold out. These additional samples have clearly enabled better models to be built.

Inclusion of the OMNI data

At this stage the information from the different OMNI chip was introduced into the analysis. Firstly, the intention was to see if the models built from the I1M data would have similar performance when applied to a completely different dataset. Secondly, if indeed the performance was similar, then the samples could be grouped together to perform a similar permutation style analysis on. The hope was that, as the addition of the extra 10% in the I1M trials resulted in an increase in performance, maybe the performance could be further improved with more samples.

Each of the different models were built using the GWAS significant SNPs data for all of the samples sequenced on the I1M chip, using the hyper-parameters outlined in table 3.4. After each model was built, it was tested once on the samples from the OMNI chip giving the results seen in table 3.5. The third column shows the median score for all of the permutations in the I1M chip data, which represent the centre lines of the boxes in figure 3.15. These results clearly show the same pattern of effect in this extra data.

Table 3.5.: The performance of the models built from the scaled allele counts of the 125 GWAS SNPs on the OMNI dataset.

Algorithm	**Score on OMNI data**	**Median score on I1M data**
Polygenic Score	0.644	0.644
Linear	0.6385	0.63
Polynomial 2	0.5139	0.51
Polynomial 3	0.5981	0.6
Polynomial 4	0.5314	0.524
RBF	0.6275	0.627

This meant that it was feasible to combine the data from both chips together and perform the same permutation procedure for all of the different kernels and the polygenic score. This was done in exactly the same manner as the permutation procedure with only the I1M sample, with 100 train/test splits, and the results are shown in figure 3.16. The layout of this figure is slightly different as the information from figure 3.15 is included as well. This was done so that the difference in performance when adding the additional data can be visualised more easily. This extra data did not deliver the increase in performance that was expected, but seems to have narrowed the distribution of scores slightly. Once again, none of the SVM algorithms outperformed the polygenic score.

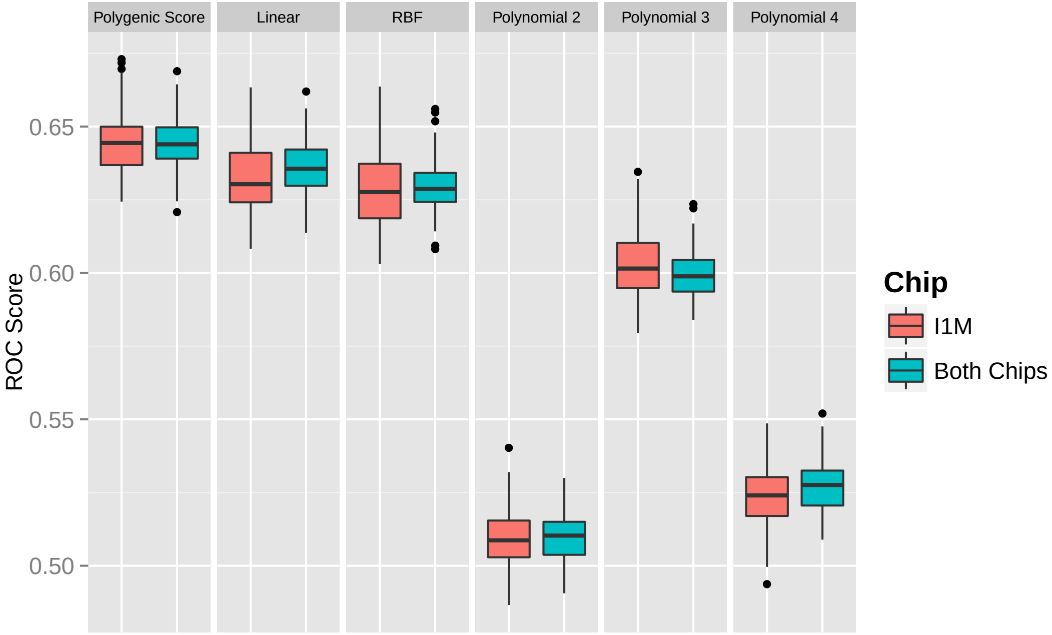


Figure 3.16.: Boxplots showing the results from the permutation procedures for all the different algorithms on the two different datasets: one with only the I1M data, and the other with the addition samples from the OMNI chip.

Table 3.6 shows the results of independent t-tests between the performance levels seen between using on the I1M chip with using both chips for all of the different algorithms. The significant differences were seen for the linear kernel, and degrees 3 and 4 or the polynomial kernel. All of the others were non significant. However, from the box plot in figure 3.16, there is still very similar performance levels seen in for the significantly different results. When looking at all the pairwise differences between the different algorithms using the information from both chips, with Bonferroni correction, the results were as follows: polygenic score to linear kernel $p < 4.5 \times 10^{-9}$, linear kernel to RBF kernel $p < 2.9 \times 10^{-5}$, all the other comparisons were even more significantly different at $p < 2 \times 10^{-16}$.

Table 3.6.: Results of independent t-tests looking at the differences between using information from both chips, and the information from the I1M chip only, for the 125 GWAS significant SNPs.

Algorithm	**t-value(198)**	**p-value**
Polygenic Score	-0.13	0.9
Linear	2.9	0.004
RBF	1.29	0.2
Polynomial-2	0.365	0.7
Polynomial-3	-2.55	0.01
Polynomial-4	2.95	0.0035

Summary from the 125 GWAS SNP dataset

The results from these trials show that unfortunately, no improvement could be made to the model using only the polygenic score. All of the differences between the results from these kernels and the polygenic score were statistically significant, even after Bonferroni correction for multiple comparison testing. A table of these results can be seen in table A.1 in appendix A. What was particularly striking was that the two polynomial kernels with even degrees had incredibly low performance, barely above chance levels, while using a degree value of three had far better performance. It is difficult to ascertain what is occurring here, and if any effect of interactions between the features could be taking place. The simulation procedures described in chapter 4 were carried out to attempt to explain this, and will therefore not be discussed here.

Another noticeable feature is that the models built on the OMNI chip proved to show the same patterns of performance on the I1M chip data, as seen in table 3.5. This was reassuring, but perhaps not completely surprising seeing as this subset of SNPs was shown to be GWAS significant from a large consortium study. Unfortunately, including all of the samples from both chips together did not result in any improvements in performance over the polygenic score.

3.3.4. Threshold Data Set

When the same hyper-parameter search was carried out on the threshold dataset consisting of the top 4,998 Index SNPs, the interpretation of the findings was a lot more inscrutable. There was a marked difference in performance across all different algorithms depending on the types of inputs that were entered (raw allele count, scaled allele counts, and weighted allele counts). Because of this, when the results are presented, the information from all of the separate five splits has been omitted in favour of showing the different patterns of performance for the different inputs. All of the results can be seen in figures 3.17 to 3.21. Unlike the figures for the GWAS significant dataset, only one of the five splits is displayed per figure, and the panels represent the different types of input that were entered into the models. As before, the x axes represent the different values of the hyper-parameters, and the y axes show the ROC score achieved.

The results for the linear kernel can be seen in figure 3.17. This was probably the most stable of the kernels, and the plots show that there was no variation in performance seen for either the raw, or the scaled allele count inputs. For the weighted inputs however, the performance did increase as $\mathbf{C}$ reduced in size. It did in fact reach an asymptote at the higher value seen, so there was no point in reducing the parameter further. This suggests that, for this input type, the model performs best when it is allowed to generalise more and allow more incorrect classifications for the training set when building the model.

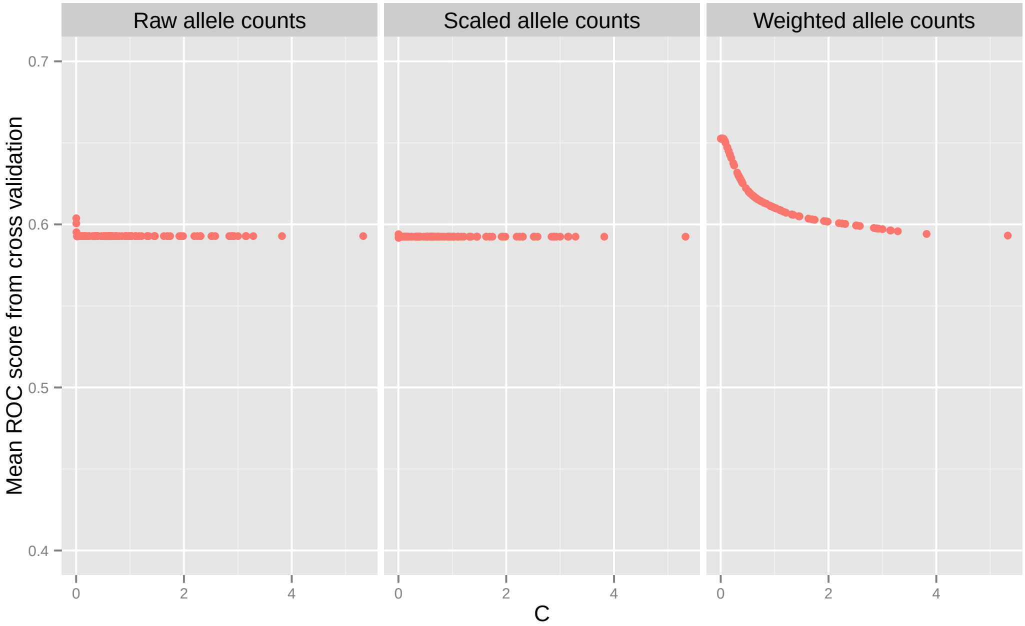


Figure 3.17.: The performance levels of the linear kernel across the different values of **C** for the three different types of input: raw allele counts, scaled allele counts and the weighted allele counts.

The results for the polynomial kernel across different values of **C** and $\boldsymbol{\gamma}$ are displayed in figures 3.18 and 3.19. From these it can be seen that the non-scaled inputs, both weighted and non-weighted, result in similar performances across the different degree values, whereas the scaled inputs result in a very similar pattern that was seen before in the GWAS significant dataset with the even degree values performing very poorly, and the value of three performing better. The better performances of the even degrees for the non-scaled inputs could well have been a result of an anomaly in the model building which was highlighted in the XOR example. It is probably safer to assume that the scaled values are more reliable because of this reason, but it was considered of interest to report the vast differences in performance which can happen with the different data types. Of note, when some of the points overlap in the plots, the differences can be quite hard to spot. The reason for this is that the software package used for the plotting (the *ggplot2* package for the R language (Wickham, 2009)) plots the points in a layered fashion. The points for the higher degrees are therefore placed over the lower degrees and while an attempt has been made to increase the transparency, there was no other way to plot this tight data and still display the required information. In order to see that this overlapping of points is happening, a figure of the weighted values for the polynomial kernel across different values of **C**, with separate panels for the different degree values can be seen in figure A.6 in appendix A.

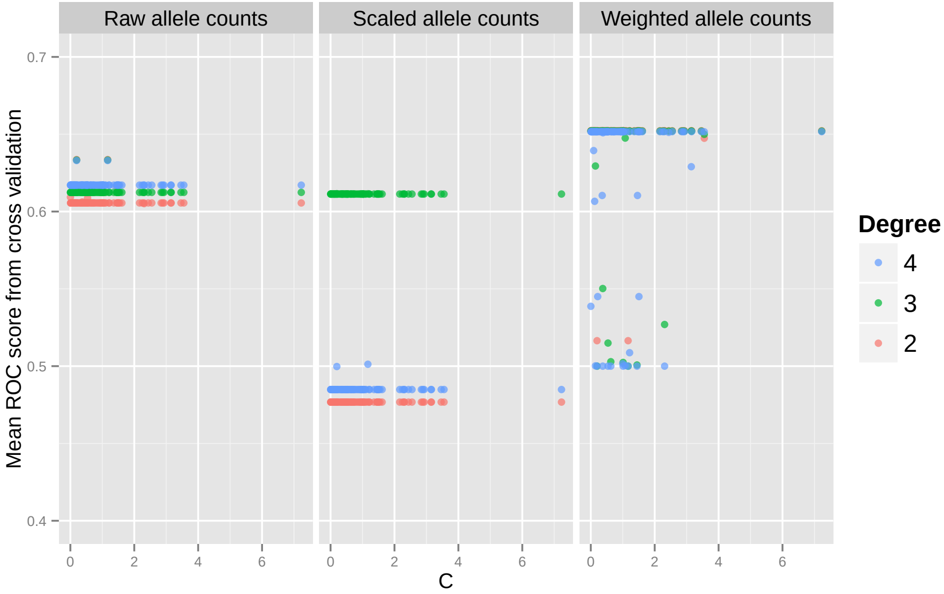


Figure 3.18.: The performance of the polynomial kernel across the different values of **C** the three different types of input: raw allele counts, scaled allele counts and the weighted allele counts.

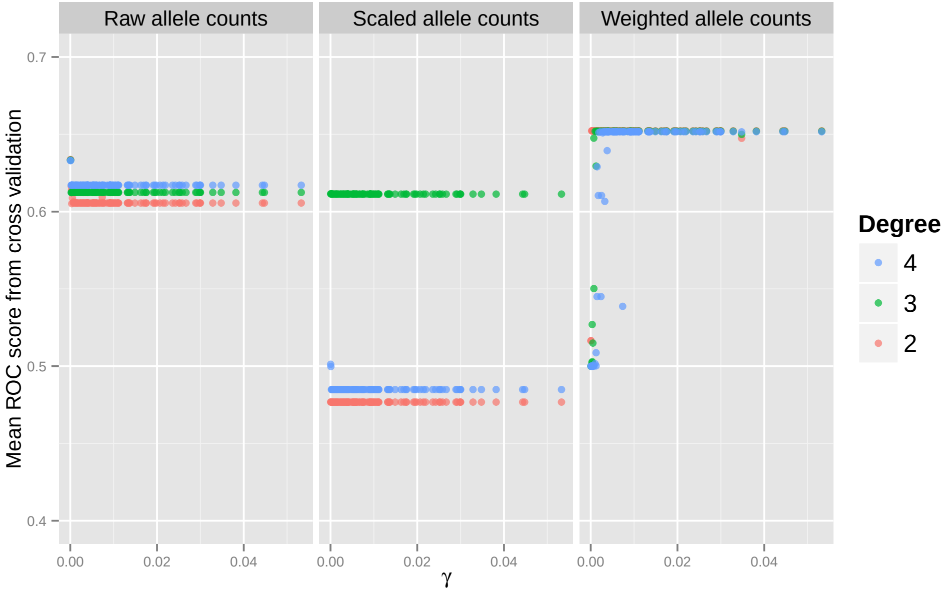


Figure 3.19.: The performance of the polynomial kernel across the different values of γ the three different types of input: raw allele counts, scaled allele counts and the weighted allele counts.

The RBF kernel performance is shown in figures 3.20 and 3.21, and it can be seen that only the weighted inputs were not highly sensitive to the γ parameter. The other two input types only performed above chance levels for the lowest values allowed of this parameter. It is difficult to know which of the input types would be best to use when interpreting the results from this kernel, as it is more robust at dealing with the presence of zero values in the data than the polynomial kernel. This was seen in section 3.2.1 using the XOR examples. It certainly showed more robust performance when the weighted inputs were used, but these levels do not seem to surpass those seen in the linear kernel. In the interest of consistency with the previous dataset, the main focus will be on the scaled allele counts.

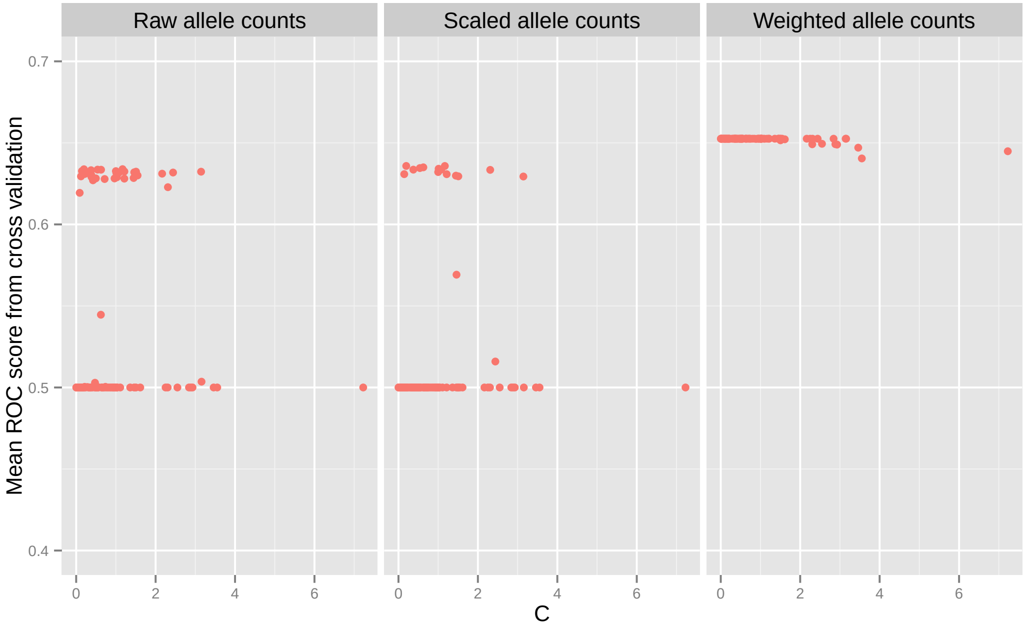


Figure 3.20.: The performance of the RBF kernel across the different values of **C** the three different types of input: raw allele counts, scaled allele counts and the weighted allele counts.

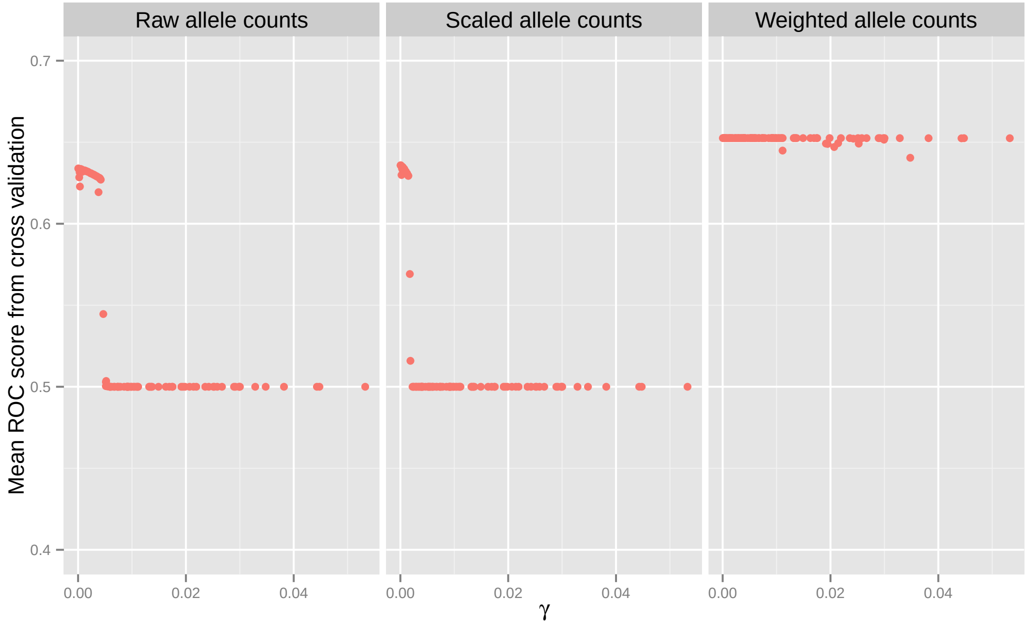


Figure 3.21.: The performance of the RBF across the different values of γ the three different types of input: raw allele counts, scaled allele counts and the weighted allele counts.

The scores and hyper-parameters chosen for the scaled inputs can be seen in table 3.7. Probably the most notable point of interest here is how low the γ values are for the non-linear kernels. Due to these extreme values being used, it was of great interest to see how they performed on the held out data, and if the performance levels replicated for the OMNI chip data.

Table 3.7.: All results and hyper-parameters for the best performing models for the 4,998 alleles in the larger dataset.

	Linear		Poly-2			Poly-3			Poly-4			RBF		
Split	Score	C	Score	C	γ	Score	C	γ	Score	C	γ	Score	C	γ
1	0.594	0.002	0.4768	1.168	<.0001	0.6115	0.195	<.0001	0.501	1.168	<.0001	0.6359	0.195	<.0001
2	0.594	0.167	0.4899	1.168	<.0001	0.6169	1.168	<.0001	0.502	0.195	<.0001	0.627	0.195	<.0001
3	0.603	0.002	0.494	2.44	0.002	0.6189	0.195	<.0001	0.505	0.195	<.0001	0.642	0.195	<.0001
4	0.603	0.002	0.484	1.168	<.0001	0.614	0.195	<.0001	0.503	1.168	<.0001	0.63	0.195	<.0001
5	0.601	0.002	0.489	1.168	<.0001	0.6116	1.168	<.0001	0.503	0.195	<.0001	0.6329	1.168	<.0001

Performance on the held out test data

As was done in the GWAS significant dataset, the models for all of the splits were tested on the 10% of the samples that were held out each time. Again, the panels represent the five different splits, and the bars show the value of the single score that was made when testing the optimum models on the 10% held out test data. This was done with the models built using the scaled allele counts. The barplots of these can be seen in figure 3.22. It is immediately evident that none of the kernels performed at the level of the polygenic score (shown with the green bars). Once again, the even degrees of the polynomial are showing a distinct drop in performance, but the linear kernel is no longer showing the same dominance over the others. In every split, is it being beaten by either the RBF kernel, the polynomial-3 kernel, or both.

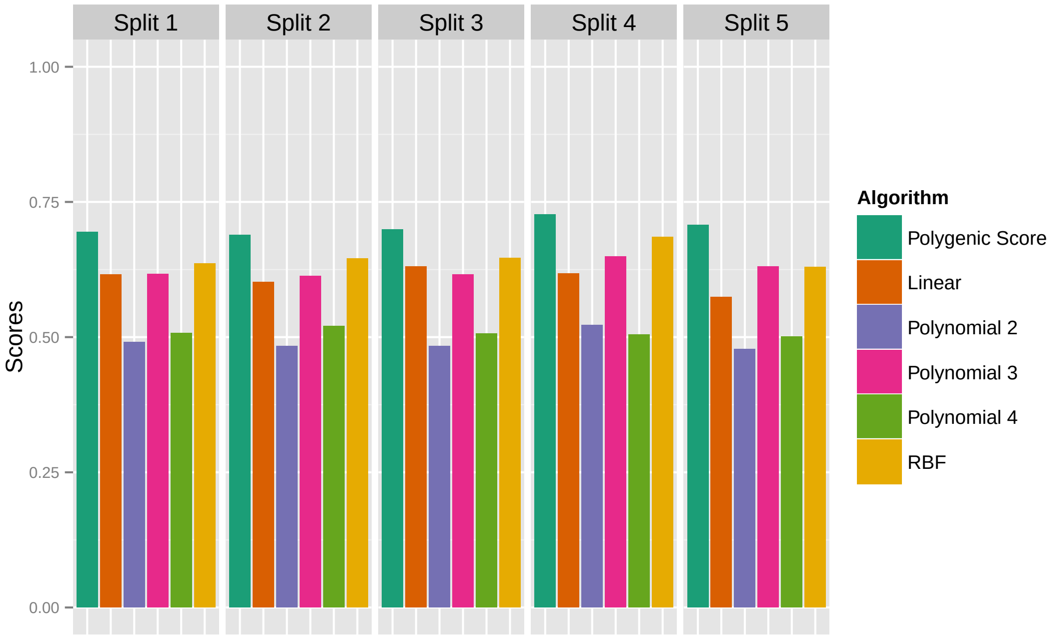


Figure 3.22.: The performance of the best models on the 10% of held out data from each splits for the larger dataset.

Permutation procedure for all algorithms

Again, the results seen for the held out data showed the same pattern between the different algorithms as those seen during the CV process. The similarities between the test and CV results can be seen in table 3.8. The **Mean CV Score** column is the mean score of the score columns for the respective algorithms in table 3.7, and the **Mean test score** column shows the averages of the test scores across the five splits; the numbers represented by the bar heights in figure 3.22. As all of the data so far has presented only the results of the hyper-parameter search, and the single point scores from the held out data for the five splits, a permutation procedure similar to that seen in the GWAS significant dataset was performed, in order to show distributions of performance for the different algorithms.

Table 3.8.: The mean results of the scores from the CV procedure and the 10% test data splits for all of the algorithms using the 4,998 SNPs.

Algorithm	Mean CV Score	Mean test score
Polygenic Score	N/A	0.703
Linear	0.599	0.6
RBF	0.634	0.649
Polynomial-2	0.487	0.491
Polynomial-3	0.615	0.625
Polynomial-4	0.503	0.508

The parameters for the permutations were set to match the best performances that were observed. For this threshold dataset, for the scaled allele counts, the performances were actually quite robust across the different parameters, but there were still trends showing higher scores. The linear kernel displayed a small preference for very small values of $\mathbf{C}$; the non-linear kernels favoured lower values of $\boldsymbol{\gamma}$ but were indifferent to the values of $\mathbf{C}$. In *scikit-learn*, the default value for $\mathbf{C}$ is 1, and the default for $\boldsymbol{\gamma}$ is the reciprocal of the sample size used in the training set, $\frac{1}{n} = \frac{1}{0.75 \times 7731} = 0.000172$. This value matches the best performing values so was deemed to be sufficient. Therefore, for this procedure, the default parameters were used with the exception of the linear kernel, which had a value of 0.002 for $\mathbf{C}$, as this was most frequently the best parameter (table 3.7). The main intention here was to see if the increase in performance for the RBF and polynomial-3 kernels would be consistent for different train/test permutations. As can be seen in figure 3.23, this was indeed the case, with the green and blue boxes of the RBF and polynomial-3 kernels positioned higher than the mustard-yellow coloured box for the linear kernels. The actual median values shown by the centre lines in the boxes were as follows: linear - 0.6008, RBF - 0.6354, polynomial 3 - 0.6164. All of the differences between all pairs of algorithms were shown to be highly statistically significant on repeated pairwise t-tests, with all combinations showing $p < 2 \times 10^{-16}$, after Bonferroni correction for multiple comparison testing.

The performances for all of the SVMs are greatly outperformed by the polygenic score analysis. This is in itself a linear combination of the features, so while the results of the machine learning alone suggests a role of interactions between the SNPs, the superior performance of the polygenic score method makes it difficult to claim that the inclusion of information from interactions is improving on predictive power.

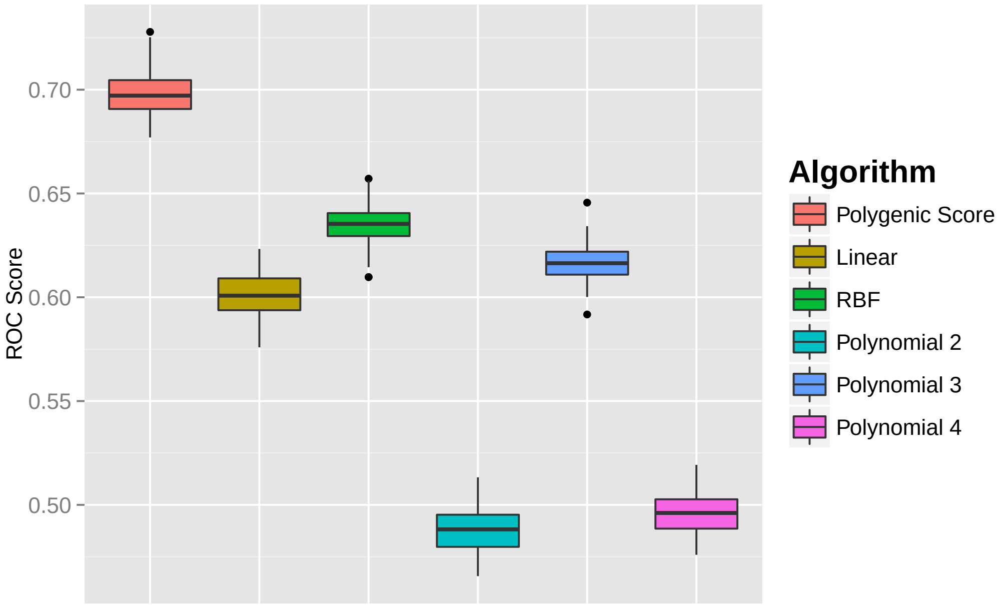


Figure 3.23.: Box plots showing the distributions of the 100 train/test permutations for the larger dataset for the I1M chip only.

Inclusion of OMNI chip information

As was done with the GWAS significant dataset, these models were then tested on the samples of the OMNI dataset. The results show a similar pattern to those seen in the held out data and can be seen in table 3.9. The median score from the permutation procedure shown in figure 3.23 is shown in the third column. These values are the numbers represented by the centre line in the boxes.

Table 3.9.: The performance of the models for the 4998 SNP on the OMNI data, and show how this compares with the median scores seen in the box plot.

Algorithm	Score on OMNI data	Median Score on I1M data
Polygenic Score	0.694	0.697
Linear	0.614	0.6
Polynomial 2	0.497	0.49
Polynomial 3	0.62	0.616
Polynomial 4	0.506	0.496
RBF	0.6487	0.635

As these results suggest that it is safe to combine the data from the two chips together for the increased number of SNPs, the same permutation procedure was carried out with the combined data and is shown in figure 3.24. An immediate aspect to note in this figure is that the performance of the SVM models increases substantially with the inclusion of the extra OMNI data. This effect is not seen at all for the permutations of the polygenic score in the leftmost panel. In fact, the only effect seen there is that the variance of the distribution of scores has narrowed. In this plot, the RBF kernel has been moved to be positioned next to the linear kernel to allow for easier comparison.

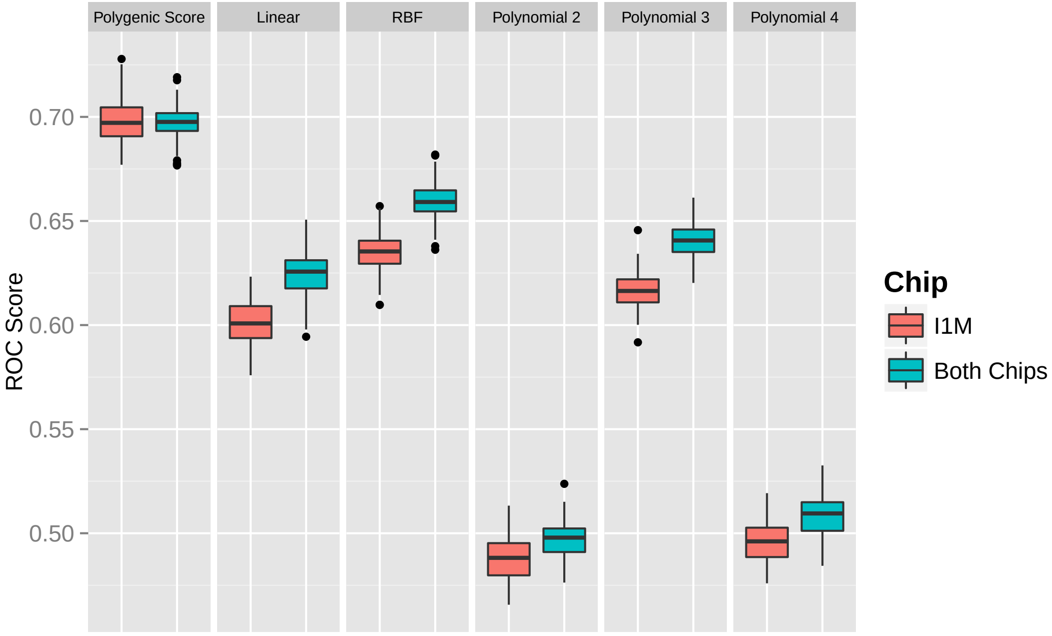


Figure 3.24.: Box plot showing the distributions of results for all the algorithms on the larger dataset of 4998 SNPs. The results show the performances for both the I1M chip only and the combined chip data.

This figure shows that all of the SVM kernels benefited from the additional samples from the OMNI chip, but the polygenic score did not. This observation was supported by the results of independent t-tests for all of the different algorithms, as seen in table 3.10. There was also significant differences seen between all pairwise comparisons of the different algorithms for the information from both chips, using repeated pairwise t-tests. All of these tests showed $p < 2.2 \times 10^{-16}$, after Bonferroni correction.

Table 3.10.: Results of independent t-tests looking at the differences between using information from both chips, and the information from the I1M chip only, for the 4,998 SNPs.

Algorithm	**t-value(198)**	**p-value**
Polygenic Score	-0.6	0.55
Linear	16.227	$< 2.2 \times 10^{-16}$
RBF	20.03	$< 2.2 \times 10^{-16}$
Polynomial-2	7	$< 3.97 \times 10^{-11}$
Polynomial-3	19.94	$< 2.2 \times 10^{-16}$
Polynomial-4	9.5	$< 2.2 \times 10^{-16}$

3.4. Summary of findings from the threshold dataset

The overall pattern of the behaviour of the results seen in the dataset is very similar to that of the GWAS significant one, but with some noticeable differences. The polygenic score showed a distinct increase in performance over the GWAS significant dataset in terms of predictive accuracy. This was not seen to the same level in the SVM models. The performance for the linear kernel decreased while the RBF and polynomial-3 kernel showed a modest increase, but not to the levels approaching the polygenic score. The even-valued polynomial kernels still displayed chance level performance.

Another notable difference is that the addition of the OMNI data samples has resulted in a clear increase in performance. This increased benefit of including the OMNI samples was not seen in the models that were using the GWAS significant SNPs, but these results suggest that this is of real benefit to models using a far higher number of input features. This finding is to be expected as the GWAS significant SNPs have a higher level of association with the disorder than other SNPs so a good level of performance can be obtained from smaller sample sizes. The sub-threshold SNPs will naturally show more variable effects across cases and controls and therefore the larger set will be better at providing a reliable estimate.

It is impossible at this stage to claim that the addition of extra samples might lead to increases in performance, but the results suggest that this could be a possibility. It would be interesting to examine this further, as there were signs that the RBF kernel was showing superior performance over the linear kernel, hinting at some possible evidence for interactions between the features. However, as stated earlier, this increased performance still did not match that of the polygenic score, so in essence, a model using a linear/additive combination of the features is still outperforming a model which is considering the interactions. This being considered, the polygenic score analysis is a far simpler model that only makes use of a single feature. Also, the polygenic score has information about the odds ratios that were made from the larger sample sizes of all the other studies that went into the discovery set, so this could explain why its level of performance was higher, even when only the I1M chip was used. The section immediately following from here actually attempts to address this issue by re-running the polygenic score analysis, but only using the allele counts, without any weighting from the LORs.

The progression that has been made in predictive performance in the different consortia studies is most likely due to the larger sample sizes providing more accurate odds-ratio information about the SNPs. In order for any improvement to be seen in the polygenic score, more accurate estimates of these odds-ratios need to be provided, which hopefully will be delivered in the results of the upcoming third study from the Psychiatric Genetics Consortium (PGC).

Maybe with the addition of even further samples, the RBF kernel could have more power to detect any possible interactions occurring. This does make some intuitive sense because the SNPs in these models represent common mutations of small effect, so it is reasonable to assume that the interactions occurring would also be of small effect. Therefore it would always be advantageous to have as large a sample size as possible.

3.5. Re-running analysis with an *unweighted* Polygenic Score

One of the main findings from the results presented so far in this chapter is that the polygenic score has consistently shown superior performance to that of the SVM algorithms. There are a few different conclusions that one can come to in light of these findings; it could be the case that it has the advantage of being a simpler, more parsimonious model; but it could have the unfair advantage of having extra information from a far larger sample of people by making use of the LOR weights from the PGC-2 findings. To recap on how the data were prepared for the results presented so far, the polygenic score was made by taking the mean of the weighted allele counts for each sample, but the SVMs were provided with the allele-counts only, which were then scaled during the Cross-Validation (CV) process. At this point, it is also worth mentioning some of the findings that were not focussed on in the main conclusions: in section 3.3.4 on page 108, the results from using the weighted inputs in the SVM models for the larger dataset with 4,998 inputs was shown in the third panels of figures 3.17 to 3.21. At the time when these results were being interpreted, it was decided that these inputs would not be the main focus, due to the issue of feature scaling to avoid having zero valued inputs into the model. It was pointed out that any scaling procedure would remove the information provided by the LOR weighting, so the allele counts were focussed on instead.

However, when referring to the information in the third panels of figures 3.17 to 3.21, an interesting observation can be made: while the behaviour of the polynomial kernels is probably unreliable, due to the presence of zero valued inputs, both the linear and RBF kernels show slightly elevated levels of performance for these types of inputs; so this does lend some support to the interpretation that the polygenic score is showing superior performance in the algorithm comparison trials due to the extra information provided by the weightings. In fact, in the later experiments using the same genotyped data in chapter 5, the datasets for all of the different algorithms are prepared using the weighted allele counts.

In order to make the interpretation of the superior performance of the polygenic score more clear, an additional series of trials were carried out, but using an *unweighted* polygenic score as the input. This score is essentially an average count of the *risk* allele count per sample, and therefore contains no additional weighting information. It can still be argued that there is still *some* information from the PGC-2 findings present here, as these weights were used in the clumping procedure to find these particular SNPs of interest, but all of the SVM algorithms make use of this same information as well, allowing a direct comparison to be made. The next section will describe how this score was made and implemented into the analysis.

3.5.1. Method to create the unweighted polygenic score

To make this score, the allele count data was rearranged to ensure that each count referred to the particular allele that had a LOR value greater than one, indicating that they all represented a *deleterious* allele. The score was then made by taking the mean value of the counts per individual. This procedure was carried out for both of the datasets, the 125 GWAS significant SNPs, and the 4,998 that were used in the threshold dataset. In the smaller dataset, the values for 62 of the 125 alleles had to be switched, and for the larger set, 2,574 alleles were changed.

3.5.2. Permutation results from the unweighted polygenic score

After the new unweighted polygenic scores were calculated, these values were entered into exactly the same CV procedure that was done in the previous trials, with 100 permutations of train/test splits at proportions of 75% and 25% respectively. The same random number seeds were also used to enable ease of comparison. This was done for both the information from the I1M chip only, and then from the data with the additional sample from the OMNI chip. The results of these permutations can be seen in figures 3.25 and 3.26. Of note, in both of these figures, as the panels are narrower than before, the information for the polygenic score and unweighted polygenic score has been labelled “PS” and “U-PS” respectively.

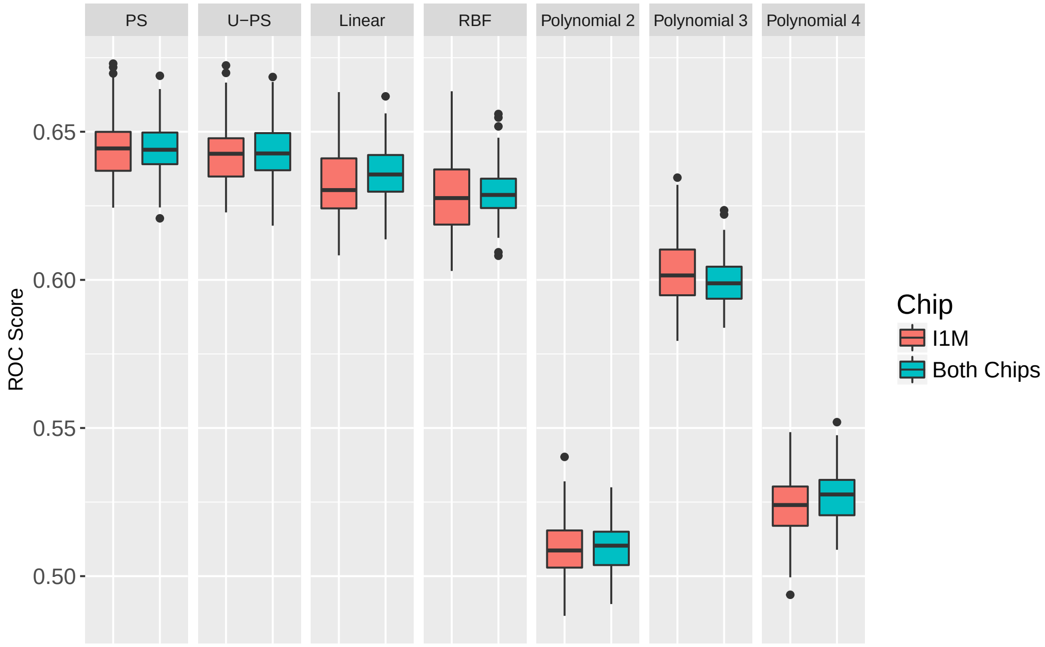


Figure 3.25.: Results showing the additional distributions from the CV permutation procedure carried out on the unweighted polygenic score from the 125 GWAS significant SNPs

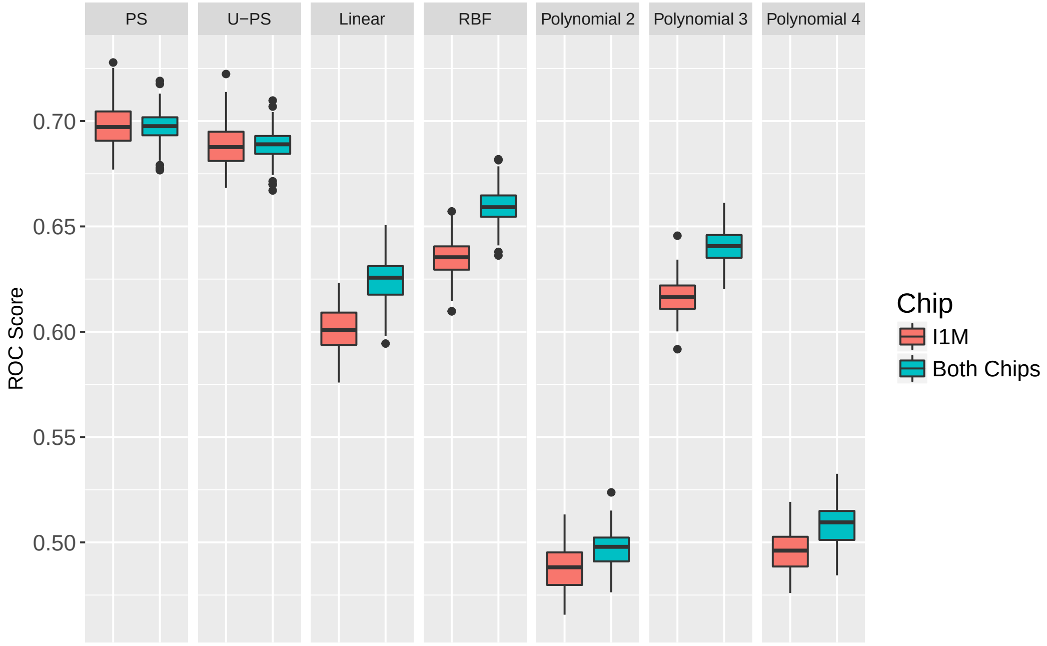


Figure 3.26.: Results showing the additional distributions from the CV permutation procedure carried out on the unweighted polygenic score from the 4,998 SNPs in the threshold dataset

Both of these figures show a very interesting pattern in the findings. Despite not having the LOR weighted information, the unweighted polygenic scores are showing a similarly high level of performance that was seen in the regular polygenic score. The information in tables 3.11 and 3.12 shows that the regular, weighted, polygenic score is still performing better, but it is very interesting to see that this difference is quite small for the smaller 125 GWAS significant SNPs. These were the SNPs that were shown to survive the multiple comparison testing procedure at the GWAS level of 5×10^{-8}. The difference is a bit more apparent in the larger dataset with 4,998 SNPs, so it seems that if those SNPs with lower levels of significance with a disorder are included, then the LOR weighting does provide some important information, but for the most significant SNPs, the knowledge of the minor allele count in itself carries a great deal of information.

Table 3.11.: The results of pairwise t-tests comparing the differences in performance between the weighted and unweighted polygenic scores for the I1M chip only, and the information from both chips combined for the 125 GWAS SNPs

Chip Type	t-value(99)	p-value
I1M chip only	7.084	$< 2.07 \times 10^{-10}$
Both chips	2.734	0.0074

Table 3.12.: The results of pairwise t-tests comparing the differences in performance between the weighted and unweighted polygenic scores for the I1M chip only, and the information from both chips combined for the 4,998 SNPs

Chip Type	t-value(99)	p-value
I1M chip only	44.572	$< 2.2 \times 10^{-16}$
Both chips	19.35	$< 2.2 \times 10^{-16}$

This greatly facilitates the interpretation of the original findings in the previous sections of this chapter. Now, it seems that the increased performance of the single feature polygenic score over the multi-feature SVM models is due to its more simple and parsimonious nature. The unweighted polygenic score has no more information provided to it than the SVMs, and it is still showing consistently higher performance than all of the different kernels.

3.6. Discussion

This study was designed to build on the findings made by the Psychiatric Genetics Consortium - 2 (PGC-2) group (Ripke et al, 2014). All of the samples used were taken from a study into people suffering from treatment-resistant schizophrenia, who were receiving Clozapine medication (Hamshere et al., 2013) and the Wellcome Trust Case/Control Consortium (WTCCC). The information used to provide the SNP weightings when making the polygenic scores were taken from the GWAS results from the PGC-2 with all the Clozapine samples held out. For the weighted version of this score, the reference allele counts were weighted by multiplying the count with the LOR results from the GWAS, and for the non-weighted version, the average count of the *deleterious* alleles was calculated.

The trials assessed how the Support Vector Machine algorithm could perform when given information about the individual variants. This was done for two different datasets, the GWAS significant dataset with information on 125 SNPs and a threshold dataset using information from 4,998 SNPs. These datasets were prepared from samples genotyped on two different chips: the I1M and OMNI chips. After some trials using a simple XOR task, it was decided that the best input features to use for the model would be the scaled reference allele counts. These trials were carried out to examine if these algorithms, specifically the kernels, which take interactions between the variants into account, could improve on the modest predictive power of the polygenic scoring method.

The main result from all of these trials is that it was not possible to improve on the results of the polygenic score. The weighted polygenic score performed the best, but the non-weighted also showed levels of performance above all of the different SVMs. However, despite this, a number of interesting aspects within the data were discovered. A positive finding was that all of the information made from all of the models (the Logistic Regression of the polygenic scores and the SVMs) was transferable from one genotyping chip to another, as the models built on the I1M chip performed equally well when tested on the OMNI chip. A particularly interesting negative finding was that the SVM models built using the polynomial-2 and polynomial-4 kernels were incredibly unsuccessful, delivering predictive performances barely above the ROC chance level of 0.5. At this stage, the reasons for this were unclear, but it was suspected that these particular models must not be good at detecting small main effects within datasets. It is also worth noting that, if the bulk of the signal is being delivered by the main effects, which is suggested by the higher performance of the polygenic scores, then this linear effect is essentially a polynomial of degree 1. This is an odd-valued degree, and can help to explain the differences seen. If plots are made of $y = x$, $y = x^2$, $y = x^3$, and $y = x^4$, then it can be seen that the cubic function is more similar in shape to the linear function, while the other two show more of a U shape to their plots, with no points below the x axis. This drop in performance was not seen when the non-scaled inputs were used, for both the weighted, and non-weighted allele counts. However, the results of the trials using the XOR information in section 3.2.1 show that the presence of many zeros in the inputs can have a detrimental effect on the expected outcome, so it was concluded that any findings using these inputs had to be interpreted with caution.

There was one particular finding, seen in the threshold dataset, which did show some promising, albeit slightly enigmatic, information. This was the fact that, given the larger number of SNP features, the RBF and polynomial-3 kernels both showed highly significant improvements over the results from the linear kernel. Based on this information alone, this would suggest evidence for interactions taking place between the variants. However, the results of the polygenic score analysis were superior to both the RBF and polynomial-3 kernels, and this score cannot take interactions into account as it is a weighted linear combination of all the SNP features in the data. The superior performance of the original, weighted polygenic score cannot be explained by claiming that it has extra information from many other samples by using the LOR weights, as the unweighted version performed at similarly high levels, all above any of the SVM models. It seems as though, at this stage, the most feasible explanation for its superior performance is that it is a simpler and more parsimonious model, which is more effective with this sample size. Maybe as sample sizes grow in new consortia studies, and with the development of increased computational power, machine learning models will begin to show improvements over linear methods like the polygenic score. But given the results of comparing the two different types of polygenic score, as the weighted polygenic score did perform slightly better, it was decided that all of the inputs for the SVMs in the remaining experimental chapter should be weighted with the GWAS LORs.

With this information, there are no more conclusions that can be drawn from the results. Therefore, it was decided that a study should be carried out to investigate how the different models perform *when the causes of the outcomes are known in advance.* In order to do this, artificial phenotypes were created from aspects (either main effects or simulated interactions between variants) present in the genetic data of the samples; this way the models can be assessed on how efficiently they can predict these phenotypes created under differing circumstances. These experiments are described in full in the next chapter.

4. Comparing algorithm performances on simulated phenotypes, created from the information in the 125 GWAS significant SNPs

4.1. Introduction

In light of the findings from chapter 3, the work in this chapter was carried out in order to gain some insight into what could have been causing the algorithms to perform in the way that they were. The main intention was to try and examine what could have been driving the algorithms to perform the way in which they were seen to with the reference allele inputs. This is, in fact, very difficult to figure out when using the real case/control status phenotype, as the aetiology of schizophrenia is largely unknown. However, if these case/control phenotypes are *created* from known aspects of the data, then the performance of the machine learning algorithms can be scrutinised by observing their outputs under different situations.

To recap the findings from the previous chapter, the polygenic score, both weighted and un-weighted, was the most successful at case/control prediction, and in the smaller set with 125 Single Nucleotide Polymorphisms (SNPs) as inputs, the linear kernel was performing at a similar level. Of the non-linear kernels, there was a similar level, albeit slightly lower, of performance seen in the Radial-Basis Function (RBF) and polynomial-3 kernels, but the polynomial kernels with even valued degrees performed much worse. For the larger dataset with 4,998 SNPs, the RBF and polynomial-3 kernels were performing better than the linear kernel. As the output from the non-linear kernels is very difficult to interpret, it would be useful to see if these patterns could be replicated with simulated phenotypes to examine what could be driving this behaviour. The simulated phenotypes can either be made from both the main effects of the inputs, or researcher-constructed interactions between them. The main point here is that in each case, the discriminatory properties that cause the samples to be classified as “mock” cases or controls is known.

The dataset used in this chapter is that of the 125 Genome-Wide Association Study (GWAS) significant SNPs from chapter 3. These were the genotyped data from the CLOZUK study carried out by Hamshere et al. (2013). In that chapter, the inputs used were the counts of the reference alleles for each polymorphism, scaled (or normalised) to have a 0 mean and variance of 1. Both the weighted and non-weighted polygenic scores showed higher performance levels than the Support Vector Machines (SVMs), but as the weighted score was slightly superior, it was decided that in this chapter, all of the analyses would use the weighted allele counts, also scaled to have 0 mean, variance 1. This meant that, all of the analyses using a polygenic score made use of the weighted version.

The allele count data that was used to create this was exactly the same as used previously, with the same replacements of the missing values described earlier. The reason that this dataset was chosen over the larger threshold dataset with the 22,568 index SNPs was due to the fact that with only 125 inputs, it is perfectly feasible to manually construct all of the possible pairwise interactions that can take place. In fact, this is exactly 7,750, calculated using the binomial coefficient shown in equation 4.1, $n = 125, k = 2$.

$$\binom{n}{k} = \frac{n!}{k!\,(n-k)!} \tag{4.1}$$

It is therefore very simple to create phenotypes that are either made from aspects of the original 125 features, the 7,750 pairwise interaction features, or even different combinations of the two. In this study, only the smaller dataset was used. If the same procedure was done with the larger threshold dataset, this would have resulted in $\binom{22{,}567}{2} = 254,623,461$ features, which would have been unfeasible. In fact, as outlined in chapter 2, the advantage of using kernel methods in machine learning is that it allows for analysis of interactions without their explicit calculation.

The first set of experiments look at the separate contributions from the main effects and interactions separately and then a series of trials using the joint contributions were performed.

4.2. Method for making phenotypes from the main effects and interactions

The aim of this initial trial was to create different phenotypes that were either wholly or probabilistically dependent on features within the data. This involved a series of different steps to be carried out to prepare this new data for analysis. The two main procedures that were carried out were:

- Create pairwise interactions between the features, either all of them, or only a subset.
- Determining what features in the data classify a sample as case or control.

The steps to create this require a fair amount of explanation, and all are outlined with a mock example. The data had to be prepared in such a way that the main effects and the pairwise interactions between them could be used to create mock phenotypes, either in a deterministic, or probabilistic manner. In this chapter, the inputs used were the Log Odds Ratio (LOR) weighted allele counts of the 125 GWAS significant SNPs from the Illumina 1M (I1M) chip (n = 7,731 samples). An outline of the steps is as follows:

1. The LOR weighted allele counts are normalised to have mean 0 and variance 1.
    - These are the main effects - either all of these are used or a subset is chosen.
2. Pairwise interactions are made from either all, or a subset of the main effects.
    - Done by multiplying the pairwise combinations of main effects.
    - Output - the interactions; either all can be used, or a subset chosen.
3. For both the main effects and interactions. Samples are selected as being "hits" or "misses".
    - Done by seeing if a sample's mean score is above or below the global mean for all samples.
4. Probabilities of being a case or a control are assigned to the hit/miss status of the main effects and interactions.
5. Either an average, or a weighted average of these two probabilities is calculated.
    - This represents each sample's probability of being a case or a control.
6. Each sample is assigned a case/control status based on a random Bernoulli trial.

An example is now outlined using five samples, showing six main effect features, of which three are used to make the interactions. This example will place heavy influence on interactions, so they will have a probability of 0.9, while the main effects will have a probability of 0.8. In addition, the interactions will be weighted as double that of the main effects. All of these steps will be made clear in the examples. The rows of the matrices represent the different samples, and the columns represent the features - in this case, the LOR weighted reference allele counts.

Step 1: The initial LOR weighted main effects are scaled:

$$\underset{\text{LOR weighted allele counts}}{\begin{pmatrix} 0.0 & 1.62 & 1.87 & 1.18 & 1.1 & 1.41 \\ 0.0 & 1.62 & 1.87 & 0.0 & 1.1 & 2.82 \\ 0.0 & 0.0 & 3.74 & 0.0 & 0.55 & 2.82 \\ 0.76 & 1.62 & 1.87 & 0.59 & 0.55 & 2.82 \\ 0.38 & 0.0 & 1.87 & 1.18 & 0.55 & 1.41 \end{pmatrix}} \rightarrow \underset{\text{Scaled LOR weighted allele counts - The main effects}}{\begin{pmatrix} -0.75 & 0.82 & -0.5 & 1.12 & 1.22 & -1.22 \\ -0.75 & 0.82 & -0.5 & -1.12 & 1.22 & 0.82 \\ -0.75 & -1.22 & 2.0 & -1.12 & -0.82 & 0.82 \\ 1.75 & 0.82 & -0.5 & 0.0 & -0.82 & 0.82 \\ 0.5 & -1.22 & -0.5 & 1.12 & -0.82 & -1.22 \end{pmatrix}}$$

Step 2: A subset of these (columns 1, 3 and 5) are chosen, and the three different pairwise interactions are made by multiplying the columns together:

$$\underset{\text{Subset of main effects}}{\begin{pmatrix} -0.75 & -0.5 & 1.22 \\ -0.75 & -0.5 & 1.22 \\ -0.75 & 2.0 & -0.82 \\ 1.75 & -0.5 & -0.82 \\ 0.5 & -0.5 & -0.82 \end{pmatrix}} \rightarrow \underset{\text{Pairwise Interactions}}{\begin{pmatrix} 0.38 & -0.92 & -0.61 \\ 0.38 & -0.92 & -0.61 \\ -1.5 & 0.61 & -1.63 \\ -0.88 & -1.43 & 0.41 \\ -0.25 & -0.41 & 0.41 \end{pmatrix}}$$

Step 3: For both the main effects and the interactions, each sample is classified as being a "hit" or a "miss" by their mean score being either above or below the global mean for each matrix.

Main effects: Global Mean = 0:

$$\underset{\text{Main Effects}}{\begin{pmatrix} -0.75 & 0.82 & -0.5 & 1.12 & 1.22 & -1.22 \\ -0.75 & 0.82 & -0.5 & -1.12 & 1.22 & 0.82 \\ -0.75 & -1.22 & 2.0 & -1.12 & -0.82 & 0.82 \\ 1.75 & 0.82 & -0.5 & 0.0 & -0.82 & 0.82 \\ 0.5 & -1.22 & -0.5 & 1.12 & -0.82 & -1.22 \end{pmatrix}} \rightarrow \underset{\text{Sample Means}}{\begin{pmatrix} 0.11 \\ 0.08 \\ -0.18 \\ 0.34 \\ -0.36 \end{pmatrix}} \rightarrow \underset{\text{Hit or Miss status}}{\begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}}$$

Step 4: The interactions are themselves scaled - and the hits and misses are calculated in the same manner.

Interactions: Global Mean = 0:

$$\underset{\text{Scaled Interactions}}{\begin{pmatrix} 1.03 & -0.44 & -0.27 \\ 1.03 & -0.44 & -0.27 \\ -1.54 & 1.77 & -1.6 \\ -0.69 & -1.18 & 1.07 \\ 0.17 & 0.29 & 1.07 \end{pmatrix}} \rightarrow \underset{\text{Sample Means}}{\begin{pmatrix} 0.11 \\ 0.11 \\ -0.46 \\ -0.27 \\ 0.51 \end{pmatrix}} \rightarrow \underset{\text{Hit or Miss status}}{\begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}}$$

At this point it is interesting to notice that samples 4 and 5 each have a different status level for the main effects and interactions.

Step 5: The samples are assigned probability values based on their status levels for the main effects and interactions. To recall, the main effects are 0.8 and the interactions are 0.9.

Main effects:

$$\underset{\text{Main effect status}}{\begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}} \rightarrow \underset{\text{Main effect probabilities}}{\begin{pmatrix} 0.8 \\ 0.8 \\ 0.2 \\ 0.8 \\ 0.2 \end{pmatrix}}$$

Interactions:

$$\underset{\text{Interaction status}}{\begin{pmatrix}1\\1\\0\\0\\1\end{pmatrix}} \rightarrow \underset{\text{Interaction probabilities}}{\begin{pmatrix}0.9\\0.9\\0.1\\0.1\\0.9\end{pmatrix}}$$

These probabilities then have to be combined together, and the average values taken. As mentioned earlier, the emphasis in this mock example is the interactions, so they have double the weight, which means that they are counted twice when the average is calculated:

$$\underset{\text{Weighted probabilities}}{\begin{pmatrix}0.8 & 0.9 & 0.9\\0.8 & 0.9 & 0.9\\0.2 & 0.1 & 0.1\\0.8 & 0.1 & 0.1\\0.2 & 0.9 & 0.9\end{pmatrix}} \rightarrow \underset{\text{Final probabilities}}{\begin{pmatrix}0.87\\0.87\\0.13\\0.33\\0.67\end{pmatrix}}$$

The final values are then used to calculate the simulated phenotype status. This is done using the `random.binomial` function in Python. As can be seen, the first two samples have a high chance of being chosen as they are hits for both main effects and interactions. Sample 3 has a low probability, as this individual was a miss for both main effects and interactions. Samples 4 and 5 are interesting as they had a hit for one and a miss for another, but as the weighting was biased towards the interactions, sample 5 has a far higher chance of being chosen as a case in this situation.

The initial experiments were carried out to examine how the different algorithms would perform on phenotypes that had been simulated from either exclusively main effects or interactions. A series of trials were carried out to test these exclusive effects, and to see what happens when changes are made to the probability values, and the number of features used to make the main effects or the interactions.

The following combinations were performed, for both the interactions and the main effects: probabilities - {0.8, 0.9, 1}, proportion of features used - $\{\frac{1}{3}, \frac{1}{2}$, all features}. These combinations were carried out exclusively from each other, so when the probabilities were changing, the proportions were fixed at 1, and when the proportions were changing, the probabilities were fixed at 1. For each simulated phenotype, a similar permutation procedure that has been described before was employed, with stratified shuffle splits using train/test proportions of 75%/25%, again with random seeds to make sure that each algorithm was given the same information for comparison reasons. Seeing as these trials were carried out to examine the main effects and interactions separately, there was no requirement to set any weighting when the averages were taken. In the examples shown here, 20 different train/test permutations were performed for each algorithm.

An additional note is that in this chapter, as the focus was on seeing how the models changed in performance based on the different nature of how the phenotypes are made, a search to find the best hyper-parameters were not of interest, so the defaults from *scikit-learn* were used with $\mathbf{C}$ set to 1 and $\boldsymbol{\gamma}$ set to $\frac{1}{nfeatures} = \frac{1}{125}$. In order to correct for any difference in numbers between cases and controls, the `class_weight='balanced'` argument was used.

For all of the trials performed in this chapter, the inputs to the models were as follows:

Polygenic Score Same as in chapter 3. Made from all the weighted SNP counts

SVM inputs The weighted allele counts, which were scaled during a pipeline process

4.2.1. Initial results from exclusive main effect and interactions

Main Effects

Box plots showing the results from the 20 split permutations for the main effects can be seen in figures 4.1 and 4.2. There are many findings presented here, which will be discussed in turn. The first is that the performance for the even degrees of the polynomial kernels display the recognised drop in performance. The second main finding is that the performance of the polygenic score suffers when only subsets of the features are used to determine the phenotypes. This is perhaps not surprising as this score is meant to represent the combined contributions from each feature, as determined by large scale GWAS, so it is unrealistic to assume that large sets of the inputs would not make any contribution. Also, this polygenic score was made from all of the features, and then fixed, so was not flexible to adapt to the different inputs. But from a machine learning perspective, this really highlights the strength of using these models in that they are capable of finding the relevant features to make their classifications.

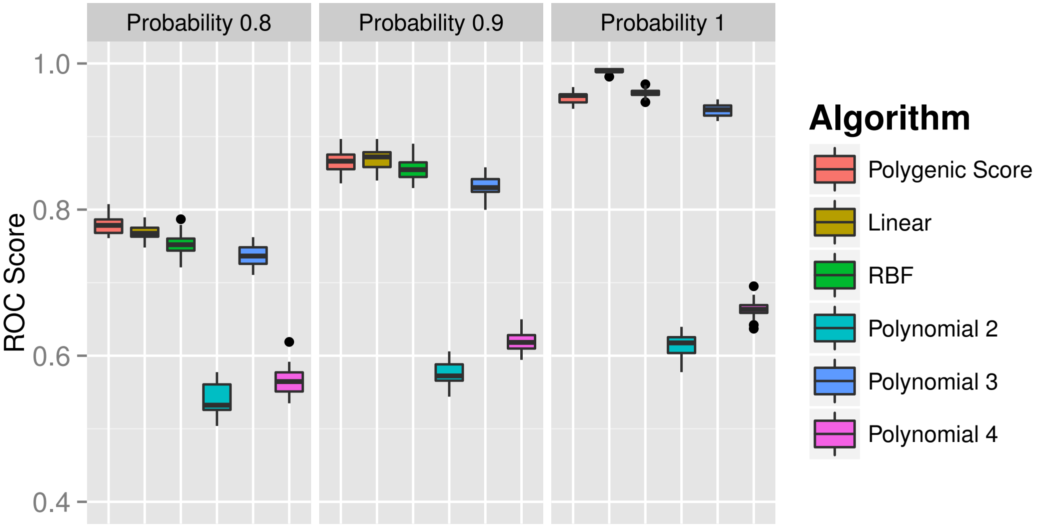


Figure 4.1.: Box plot showing the performance of the algorithms for phenotypes simulated from different levels of penetrance of the main effects. As the tight performance makes the colours difficult to see, the algorithms listed top to bottom in the legend go from left to right on the plots.

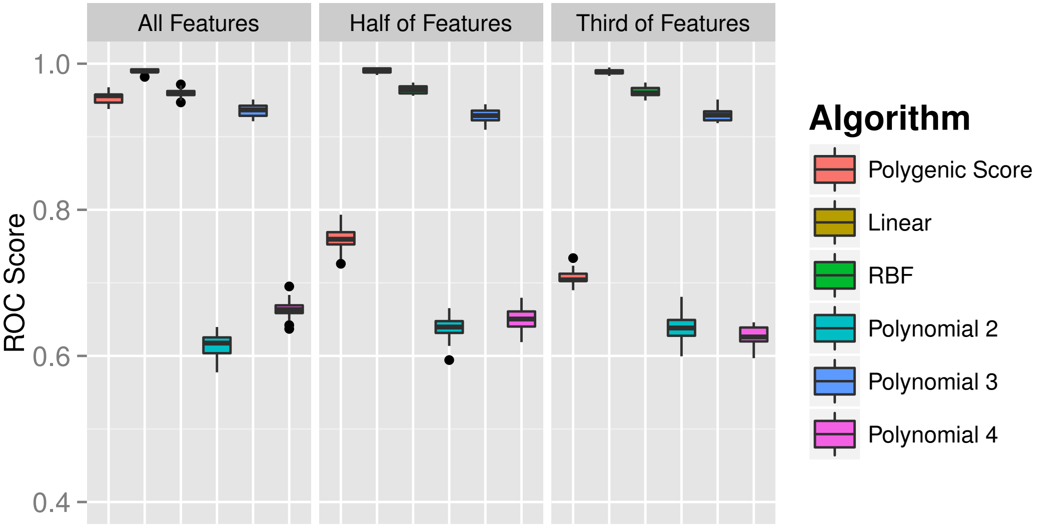


Figure 4.2.: The performances of the algorithms for simulated phenotypes made with different subsets of the main effects. As the tight performance makes the colours difficult to see, the algorithms listed top to bottom in the legend go from left to right on the plots.

Interactions

Similar plots to the main effects, showing the different performance of the algorithms on the prediction of the simulated phenotypes made from the pairwise interactions are shown in figures 4.3 and 4.4. These data reveal a very different pattern of behaviours, not seen in the main effect simulations or the results in chapter 3. As might be expected, the performance of the polygenic score and the linear kernel is very poor, at the chance level of 0.5. There is a clear improvement seen in non-linear kernels, but the RBF, and the polynomial-2 especially, are performing better. It makes logical sense that the polynomial-2 should be performing particularly well as it is based around looking for pairwise multiplicative interactions between the features - exactly the method used in these simulations. However, the performance is far from perfect, as even at the maximum probability level, it does not get better than 0.8 Receiver Operating Characteristic (ROC).

As seen in the main effects examples, varying the probability levels results in the largest changes seen in performance. The models are once again quite resilient to there only being subsets of the features used to make the interactions. This suggests that the Support Vector Machine algorithm is particularly good at identifying contributing features from those that are not providing any information and are therefore just nuisance variables in the models.

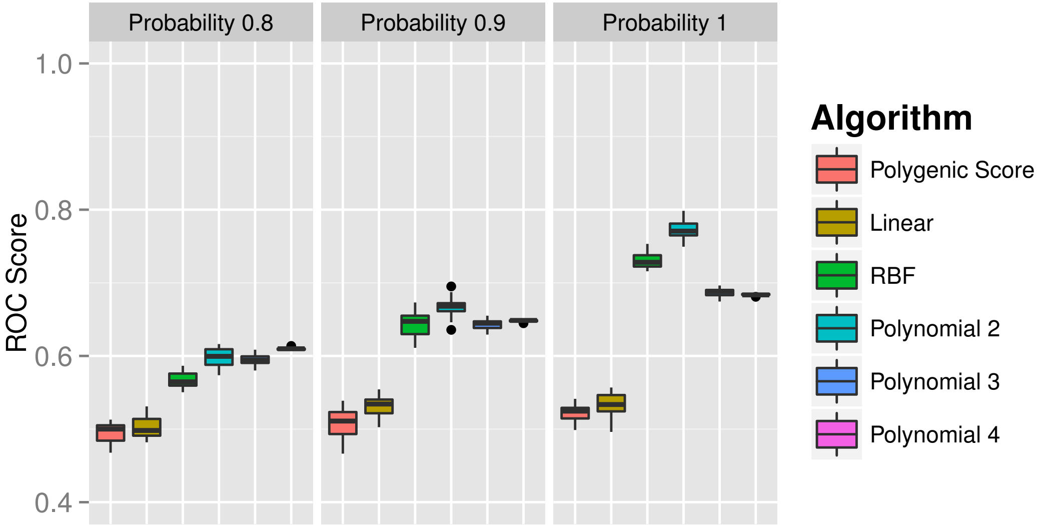


Figure 4.3.: Boxplot showing the performance of the algorithms for phenotypes simulated from different levels of probability of the interactions.

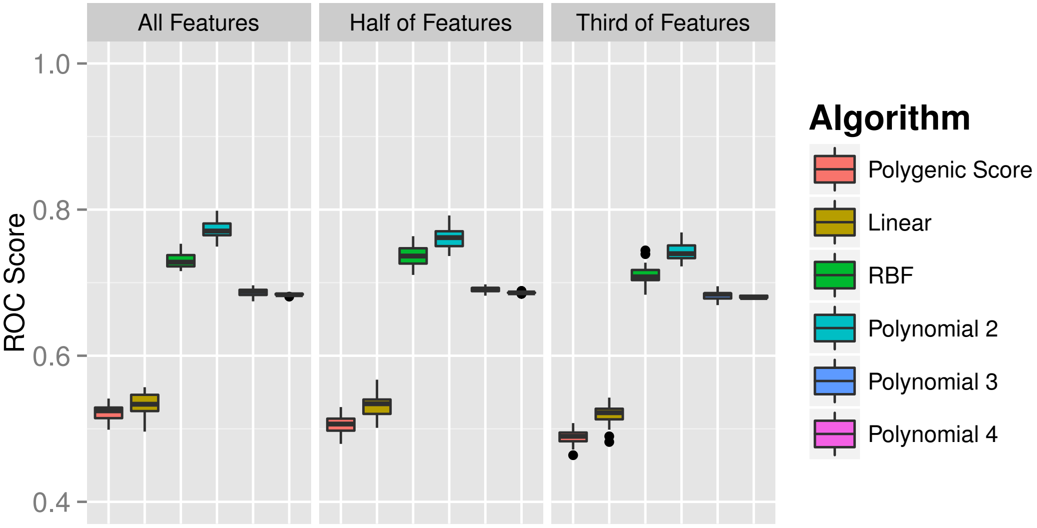


Figure 4.4.: The performances of the algorithms for simulated phenotypes made with different subsets of the input features. As the tight performance makes the colours difficult to see, the algorithms listed in the legend go from left to right on the plots.

Based on the findings of these simulations, it looks as though the pattern of results that most resembles that seen in the real data in chapter 3 is the performance seen on the phenotypes simulated from the main effects, using all of the input features. While the performance seen in the models does not exactly match that seen in the real data (as in the real data, it is the polygenic score which performs best and not the linear kernel), it is showing the very characteristic drops in performances for the even-valued polynomial kernels. Also, it is assumed from these results that all of the main effects should be considered when building the simulations. This is because, in the real dataset in the previous chapter, the polygenic score was always performing better than the SVMs. In these simulations, it can be seen that only providing a subset of the main effects has a detrimental outcome on the performance of the polygenic score, a pattern of behaviour not previously seen. In addition, it makes logical sense that all of the main effects should play a role, as they were selected due to their high association with schizophrenia from the GWAS results (Ripke et al, 2014).

Another interesting finding was that the RBF kernel seemed to show quite robust performance levels for both the main effects and the interactions. While the linear kernel was best for the main effects, and the polynomial-2 kernel was best for the pairwise interactions, both of these performed extremely badly when the conditions were swapped. The conclusion from this is that, if the desired outcome is to gain predictive performance without the need for interpretability of the model, the RBF kernel is a good first step, based on the versatile levels of performance.

4.3. Examining differing contributions of combined main effects and interactions on the simulated phenotypes

The next stage of the analysis was to vary the amount of contribution provided by the main effects and the interactions into the simulation of the phenotypes. This was done by making adjustments to the probabilities and weight values of the main effects and interactions.

For all of the trials, all of the features were used for the main effects, due to the assumption just mentioned earlier. However, it was not so easy to decide on the correct proportion of features to use for the interactions. Therefore, for all of the trials described below, the following proportions of the 125 SNP features were used, in all cases, the number of features chosen was rounded to the nearest integer value:

- 5% = 6 features - 15 pairwise interactions
- 10% = 12 features - 66 pairwise interactions
- 20% = 25 features - 300 pairwise interactions

These values were chosen as they represented a variety between very few interactions, to what is possibly quite a high number of interactions.

For all of the different proportions of interacting features, the simulations procedure was run with the following parameter values for the probabilities and weights for both the main effects and interactions. This was done to gain a variety of different phenotypes, simulated from varying contributions from the main effects and interactions.

- Main Effect Probabilities - {0.7, 0.9}
- Interaction Probabilities - {0.7, 0.9}
- Main Effect Weights - {1, 2, 3}
- Interaction Weights - {1, 2, 3}

Using all of these values results in 36 different final probability values to create the simulated phenotypes ($2 \times 2 \times 3 \times 3$). As the phenotypes are created from these probability values, in order to try and minimise the chances of any performance results being due to patterns that could have arisen by chance, each combination of parameter values was run five times (by using random-seeds in the programs), resulting in 180 different simulations carried out for each of the three different proportions of interactions.

All of these different simulated phenotypes were used to assess the different levels of performance when using the polygenic score, the linear kernel, and the RBF kernel. These algorithms were chosen as they seemed to show the most stable levels of performance in the earlier trials. While the linear kernel and the polygenic score did not perform well on the phenotypes made exclusively from the interactions, it must be remembered that this is to be expected. The RBF kernel on the other hand showed a more stable level of performance, and was therefore chosen as the algorithm that could potentially detect contributions from the interactions. While it performed very well on the pairwise interactions, the polynomial-2 kernel (and the polynomial-4 kernel) was not chosen to be used in this trial as its volatility in performance seen when comparing the main effects and interactions raised the concern that any results would be difficult to interpret. The polynomial-3 kernel was also not chosen as it did not perform to the same level as the RBF kernel.

As in the initial study using the exclusive main effects or interactions, each algorithm was run with 20 different stratified Cross-Validation (CV) permutations with a train/test proportion of 75%/25%, using a random number seed to ensure that each algorithm had exactly the same splits to allow easier comparison of results. The default parameters for the SVMs were used again: $\mathbf{C} = 1$, $\boldsymbol{\gamma} = \frac{1}{nfeatures} = \frac{1}{125}$. To take account of any discrepancies in proportions between cases and controls, the `class_weight='balanced'` argument was used again.

Main aims of this study

The main goal of this study was not to only assess the predictive performances of the algorithms, but to examine what levels of increased contributions from the interactions are required in order to see a benefit in using a non-linear kernel, in this case the RBF kernel. This would not just be of use for the interpretation of the patterns seen in for the larger threshold dataset in chapter 3 (where the RBF kernel outperformed the linear kernel, but not the polygenic score), but also to other researchers who might be interested in assessing whether it is worth implementing any machine learning algorithm with a non-linear kernel, given the sort of patterns that they assume could be occurring in their datasets.

The procedure to create the simulated phenotypes, described in section 4.2, is a method of creating the binary outcome for the models only. It does not provide any means of assessing the different relative contributions of the main effects and interactions used, and it is impossible to do this from examining the parameter values by themselves. Because of this, an additional stage of data analysis had to be carried out, making use of the logistic regression algorithm under the Generalised Linear Model (GLM) framework. This is a process that requires more statistical modelling and was therefore more appropriate to be performed using the R programming language instead of Python. R is another open-source language, often used in the field of data-science and statistical modelling in particular (R Core Team, 2013).

When the simulation trials were carried out, the different simulated phenotypes and the selected features used to make the interactions were saved, in addition to the ROC scores obtained from each CV permutation. This allowed the GLM analyses to be carried out on the specific main effects and interactions used to create each simulated phenotype. All the steps in the process carried out to examine how different relative effect sizes of the main effects and interactions were related to the different levels of performance for the different algorithms is described below. This process was carried out on every simulated phenotype used, across all of the different parameters, proportion of features, and the five repeats of this process (n = 540); together with their respective list of interaction features and the ROC scores from the predictive algorithm models. A description of this process is described here, and then an example will be provided that was taken from the real data analysis.

1. The 125 LOR weighted reference allele inputs were scaled to have mean 0, variance 1.
2. A set of simulated phenotypes was loaded, together with:
   - Details of which SNPs were used to make the interactions.
   - The ROC scores for all three algorithms.
3. All of the pairwise combinations of the features were explicitly calculated.
4. For each of the pairwise combinations:
   a) A logistic regression model was built using the `glm` function in R, looking at the two main effects and the interaction between them together with the phenotype[1].

   b) The $\boldsymbol{\beta}$ coefficients of the two main effects and the interaction term were obtained, together with their p-values.

   c) The ratio of relative effect sizes was calculated by dividing the $\boldsymbol{\beta}$ coefficient of the interaction term by the largest of the $\boldsymbol{\beta}$ coefficients of the main effects.

   d) As is usual practice with ratio values, the natural logarithm was taken of this value.
5. Once all of the log ratios had been calculated, the median value of these were recorded to protect against any possible outlier effects
6. The ROC scores from the 20 different CV permutations were then analysed:
7. Pairwise t-tests were carried out between the RBF and linear kernels; and the RBF kernel and the polygenic score[2].
8. Both the t-values and p-values from these analyses were stored.

An example is now provided that gives a breakdown of the analysis used to create a single datapoint for one of the simulations that made use of 5% of the main effects inputs, which resulted in six SNPs and therefore 15 pairwise interactions. This example will include the output from the logistic regression for one pair of SNPs, and then an outline of how these results were then used to calculated the different ratio values.

[1] `glm(phenotype ~ main_effect1 * main_effect2, family='binomial', data=dataframe)`

[2] Pairwise tests could be made due to the use of the exact same train/test splits at each permutation

At first, the identities of the six SNPs used for this particular simulation were loaded, together with the respective phenotypes, and the 15 pairwise combinations were calculated. The weighted counts for these two SNPs were then extracted from the data and scaled to have 0 mean and a variance of 1. This resulted in two vectors of values that were then entered into the logistic regression model, together with their interaction, and the dependent variable of the simulated phenotypes. The $\boldsymbol{\beta}$ coefficients were then extracted from this model. The output from one pair was as follows:

$$\underset{\text{Both main effect coefficients}}{\begin{pmatrix}0.041 & -0.031\end{pmatrix}} \qquad \underset{\text{Interaction coefficient}}{\begin{pmatrix}0.099\end{pmatrix}}$$

As can be seen, it is the first of the main effects that provides the largest relative effect size, so this one was recorded, along with the value for the interaction coefficient.

This model building and coefficient selection was performed on all of the 14 remaining pairwise combinations of SNPs, resulting in two vectors each of length 15 for the main effects and interactions, notice that the top values represent the findings just shown:

$$\underset{\text{All main effect coefficients}}{\begin{pmatrix}0.041\\0.042\\0.052\\0.047\\0.041\\-0.03\\0.05\\0.045\\-0.031\\0.051\\0.046\\0.018\\0.052\\0.05\\0.047\end{pmatrix}} \qquad \underset{\text{All interaction coefficients}}{\begin{pmatrix}0.099\\0.1\\0.045\\0.054\\0.03\\0.062\\0.079\\0.051\\0.092\\0.094\\0.037\\0.075\\0.033\\0.035\\0.077\end{pmatrix}}$$

It can be seen here there there are two values in the main effects that have negative values. As the main focus of interest was the ratio of the relative sizes of the coefficients, the absolute values were always taken. Also, if negative values had remained, then it would have caused errors when attempting to calculate logarithm values.

After these values were recorded, the logarithm of the ratio between the interaction and main effect coefficients was calculated. This was done by taking *element-wise* division of the 15 interaction absolute values by the 15 main effect absolute values, and taking the natural logarithm (Ln) of that outcome. Notice that there are no negative values in the following columns of information:

$$\underset{\text{All main effect coefficients}}{\begin{pmatrix} 0.041 \\ 0.042 \\ 0.052 \\ 0.047 \\ 0.041 \\ 0.03 \\ 0.05 \\ 0.045 \\ 0.031 \\ 0.051 \\ 0.046 \\ 0.018 \\ 0.052 \\ 0.05 \\ 0.047 \end{pmatrix}} \quad \underset{\text{All interaction coefficients}}{\begin{pmatrix} 0.099 \\ 0.1 \\ 0.045 \\ 0.054 \\ 0.03 \\ 0.062 \\ 0.079 \\ 0.051 \\ 0.092 \\ 0.094 \\ 0.037 \\ 0.075 \\ 0.033 \\ 0.035 \\ 0.077 \end{pmatrix}} \xrightarrow{\text{Inter} \div \text{Main}} \underset{\text{Coefficient ratios}}{\begin{pmatrix} 2.427 \\ 2.376 \\ 0.879 \\ 1.147 \\ 0.738 \\ 2.037 \\ 1.565 \\ 1.131 \\ 2.965 \\ 1.856 \\ 0.787 \\ 4.124 \\ 0.628 \\ 0.699 \\ 1.64 \end{pmatrix}} \xrightarrow{\text{Ln}} \underset{\text{Log ratios}}{\begin{pmatrix} 0.886 \\ 0.865 \\ -0.129 \\ 0.137 \\ -0.304 \\ 0.712 \\ 0.448 \\ 0.123 \\ 1.087 \\ 0.618 \\ -0.239 \\ 1.417 \\ -0.466 \\ -0.359 \\ 0.495 \end{pmatrix}}$$

The right column shows all of the 15 different log ratios for this particular simulated phenotype; all of the positive values show where the interaction coefficient was larger, and it can be clearly seen that this is the case for the majority here. The final value taken from this information was the median of this column, which in this case is 0.448. In the plots shown in the results, this information is shown as points on different scatter diagrams, and this value represents the points on the x axes.

After this median ratio value was calculated, the ROC results from the 20 CV permutations that were carried out for both the linear and RBF kernels, as well as the polygenic score for the respective simulated phenotypes were loaded, and pairwise t-tests performed between the RBF and linear kernels, as well as the RBF kernel and polygenic score. The t values from these tests represent the values on the y axes of the scatter plots.

This process resulted in a series of sets of information data points: the t-test results (together with their p-values) of the comparisons of the RBF kernel with the linear methods, and the median of the log ratios between all of the possible pairwise interactions, and the largest respective main effect; calculated for each simulated phenotype. The "effect-sizes" of the main effects and interactions shown by the $\boldsymbol{\beta}$ coefficients in a logistic regression model of a GLM actually represents the change in the *log odds* of the output that is made by having a unit change in the respective feature for that coefficient. More information of the interpretations of the output of a logistic regression model can be read in Kleinbaum and Klein (2010). The reason for this is that the logistic regression model works by performing a *Logit* transformation of a linear model.

However, before the results of this procedure are shown, it is worth examining the proportion of cases and controls created for each simulated phenotype. This is a good check to carry out, to make sure that any results seen cannot be confounded by a large skew in the proportion of cases and controls in the GLM procedure. The distribution of the proportions of the samples which were simulated as being cases across the different proportion of features can be seen in figure 4.5. This figure shows that while there was not an even split of cases and controls in these simulations, there are also no extreme outliers, so all of these values were deemed suitable for use with the `glm` function in R. It also shows that as the number of features used to make the interactions increased, there was a slight tendency for more of the samples to be classified as controls instead of cases.

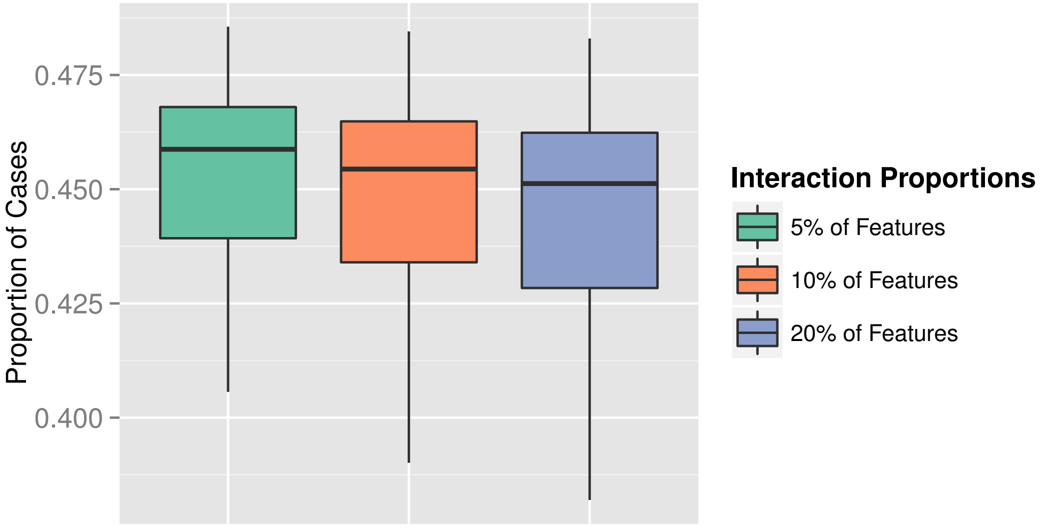


Figure 4.5.: Box plots showing the distribution of the proportions of the samples that were classified as being cases for the simulation procedures for the three different levels of feature proportions used to make these simulated phenotypes.

4.3.1. Results of the effect size study

With this summary data at hand, it is now possible to compare how the different algorithms performed with different contributions of the interactions and main effects, when different proportions of SNPs were used to make the permutations. To recap, the comparison is made by looking at the log ratios of the interaction coefficients to the largest main effect coefficient for each possible pairwise comparison, taking the median of these scores and seeing if they have any relationship between the ROC scores, when comparing the RBF kernel with both of the linear methods (the polygenic score and the linear kernel), assessed with pairwise t-tests.

RBF to Linear kernel comparison

A scatter plot for the comparison of the RBF kernel with the linear kernel, showing the relationship between the two measurements for the different proportions features used to make the pairwise interactions can be seen in figure 4.6. In all of the panels, the x axis represents the natural log of the coefficient ratio of the interactions and largest main effects, and the y axis represents the t-value statistic from the pairwise t-test of the ROC scores achieved in the CV permutation procedure, which, to recap, consisted of 20 train/test splits of the data with 75%/25% respective proportions, always using a random number seed to ensure that the different algorithms were provided with exactly the same splits of samples. The three colours represent whether the t-tests were statistically significant or not ($p < .05$), and which algorithm was performing better if there was a significant difference. It can be seen that the non-significant findings cluster in the regions where the t-values are close to 0. The vertical lines show the point where the median of the 20 log ratios between the main effects and interactions is 0.

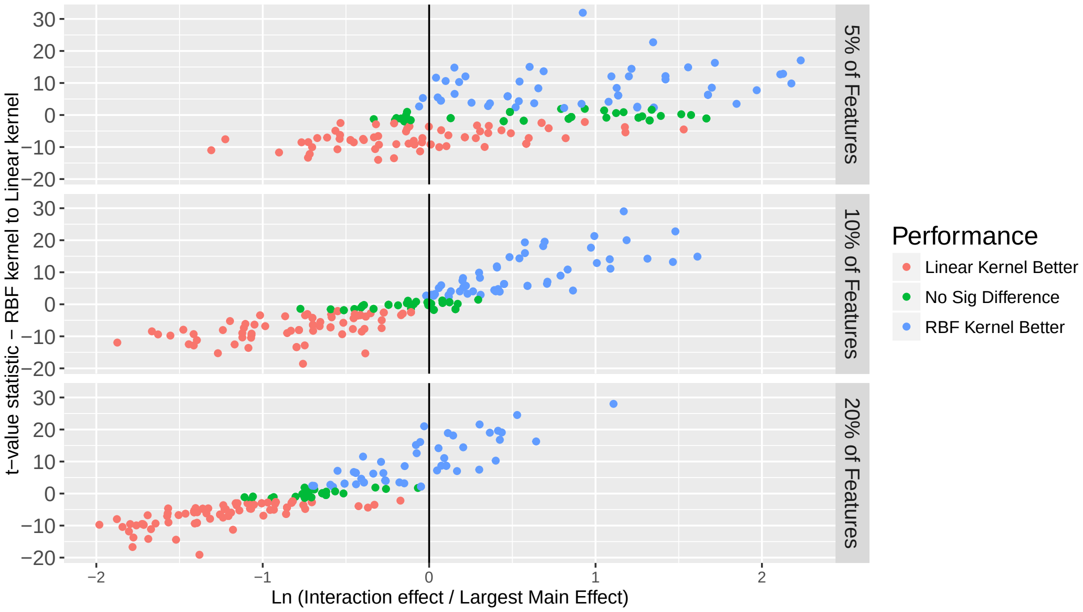


Figure 4.6.: Scatter plot showing the relationship between the median values of the log ratios of the interaction and the respective largest main effect coefficients and the difference in performance of the RBF and linear kernels, assessed by pairwise t-tests. The colours represent if the difference in performance for the t-tests was significant at p $<$ .05, and which algorithm was performing better if the difference was significant.

There is a great deal of information provided in this figure. It seems apparent that when more features are used to make the interactions, the relative effect sizes of the interactions compared to the main effects are reduced. This is the reason why the dots do not extend so far to the right along the x axis for the 10% and 20% panels. Attempts were made to increase these by making further adjustments to the probability and weight parameters when the simulations were carried out, which were unsuccessful.

Also, when there are sufficient interactions contributing (20% case), they do not have to be larger than main effects for RBF kernel to do better. However, when there are interactions between fewer of the features, while the relative differences in effect sizes between the interactions and main effects increases, this does not necessarily result in a superior RBF performance. For the 10% panel, it is still apparent that once these effect sizes become larger, then the superior performance is resumed. The largest log ratio that still shows a non-significant result occurs when the log ratio is 0.4144325, meaning that, at this point, the interaction is 1.514 ($e^{0.4144325}$) times larger than the main effects. So while the RBF kernel is not reliably detecting any interaction contribution, once these contributions go above $\sim$ 1.5 that of the main effects, it is likely that this will be discovered. For the top panel however, where only 5% of the features are interacting with each other, the story is very different. There are still non-significant differences between the kernels seen when the log ratio is as high as 1.848. At this point, the interactions are 6.34 ($e^{1.848}$) times larger than the main effects. This shows that when so few features are interacting with each other, the effects of these interactions need to be substantially larger, by a number of magnitudes, in order to see any benefit from using the RBF kernel.

RBF kernel to Polygenic Score comparison

In the study carried out for the larger threshold dataset in chapter 3 (section 3.3.4), the results showed that the RBF kernel was outperforming the linear kernel, but not the polygenic score. The difference between the RBF and linear kernels was even more apparent when the additional samples were added from the Illumina Omni Express (OMNI) chip (refer to figure 3.24), but yet again, there was no increase in performance over the polygenic score. This led to the difficult conclusion that the performances of the different SVM algorithms suggested a role of interactions in that dataset, but the scores seen in the polygenic score contradicted this conclusion as it is itself a linear combination of the input features.

The results from this simulation study provide the opportunity to elucidate what could be occurring in this situation, as the different simulations were tested for classification performance using the polygenic score as well as the SVMs. This means that the same comparison between the log ratio of the effect sizes and the differences in performance between the RBF kernel and the polygenic score can be displayed in exactly the same manner. These results can be seen in figure 4.7.

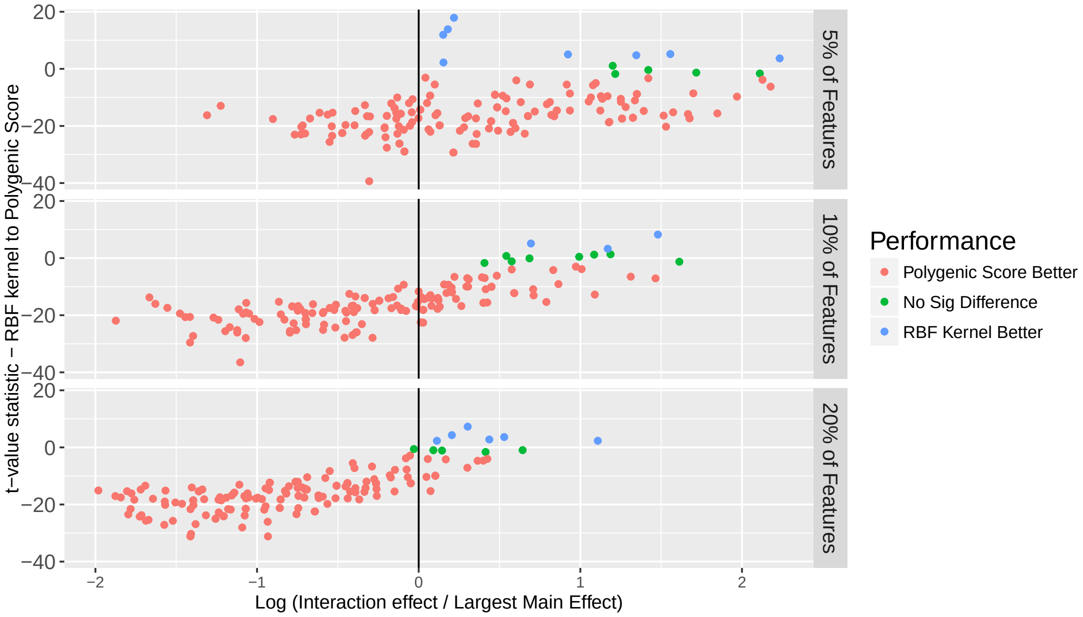


Figure 4.7.: Scatter plot showing the relationship between the median values of the log ratios of the interaction and the respective largest main effect coefficients and the difference in performance of the RBF kernel and polygenic score, assessed by pairwise t-tests. The colours represent if the difference in performance for the t-tests was significantly different at p < .05, and which algorithm was performing better if the difference was significant.

This figure shows a very different pattern to that seen in figure 4.6. In all three of the different panels representing the feature proportions, there is no significant increase in performance when using the RBF kernel seen with the increase in the interaction coefficients. There is still an upwards linear trend seen in the performance, and there are isolated incidents where the RBF kernel is showing increased performance, but these are most likely due to chance alone. This is a very interesting finding, as it suggests that the polygenic score is still capable of capturing the contribution from interactions in the data, despite it being a linear combination of the features.

Distributions of ROC Scores

So far, the figure have only showed how the results of the t-tests have changed with the different contributions of the main effects and interactions and not the changes of the actual ROC scores themselves. Therefore, in this section, the distributions of how these scores change across the log ratio values for the three different algorithms are shown. Because the scores were often quite tight together, all three are not shown together; instead, the three different pairwise comparisons of RBF kernel to linear kernel, RBF kernel to polygenic score, and linear kernel to polygenic score are shown in figures 4.8 to 4.10

respectively. All three figures display the information in the following format: the individual points in the scatter plots show the median ROC scores from the 20 Cross-Validation (CV) permutations for each phenotype. The coloured solid lines represent a smoothed Loess regression curve for each algorithm, with the shaded boundaries showing the 95% confidence intervals calculated from the standard errors. Loess regression (Cleveland et al., 1992) is a method of fitting a polynomial curve to the data and was used to show the general trend in the scores and the log ratios, as there was a fair amount of variance seen in the scatter points that made these difficult to identify. Carrying this out was done with the `geom_smooth` function in the *ggplot2* package in R using the default parameters.

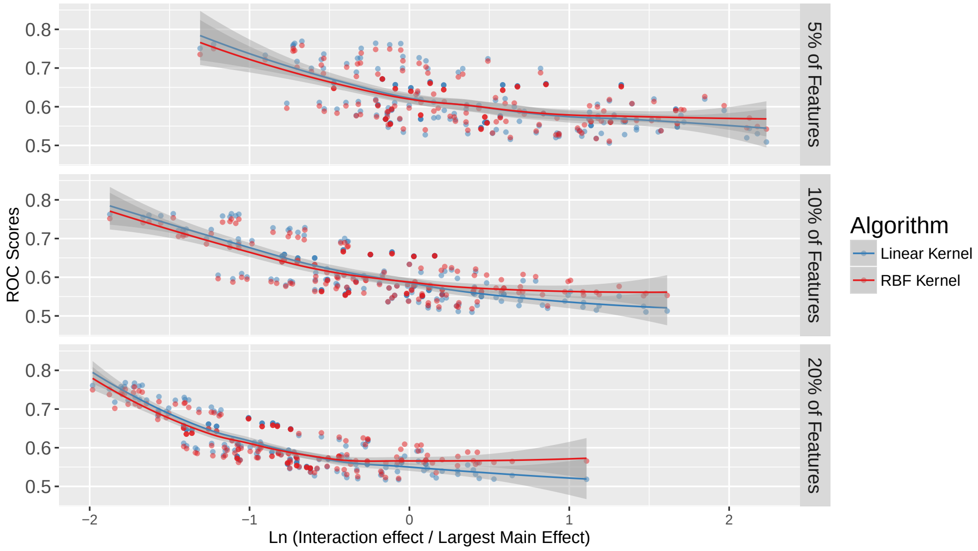


Figure 4.8.: Scatter plot showing the distribution of the median ROC scores for the RBF and linear kernels from the 20 CV permutations. The coloured lines represent the Loess regression curve, with the shaded 95% confidence interval region.

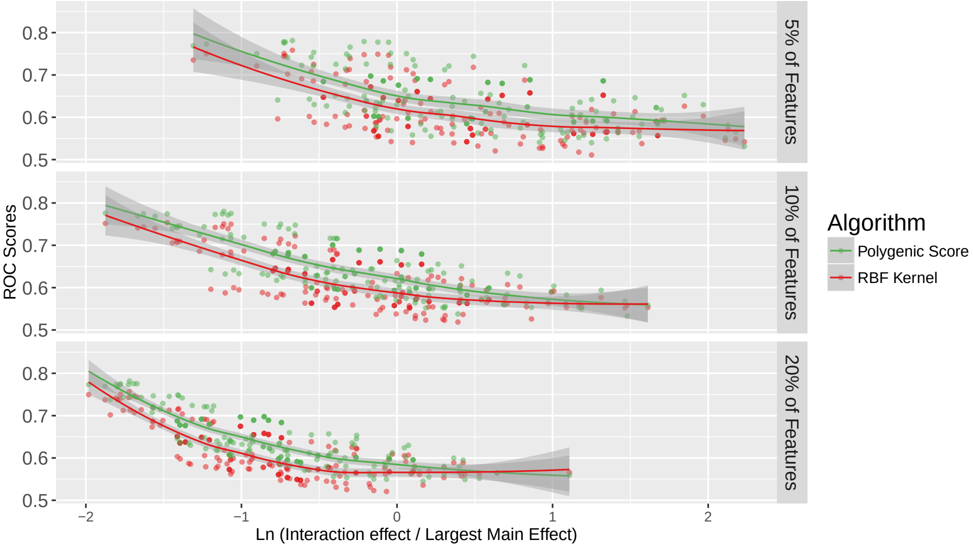


Figure 4.9.: Scatter plot showing the distribution of the median ROC scores for the RBF kernel and polygenic score from the 20 CV permutations. The coloured lines represent the Loess regression curve, with the shaded 95% confidence interval region.

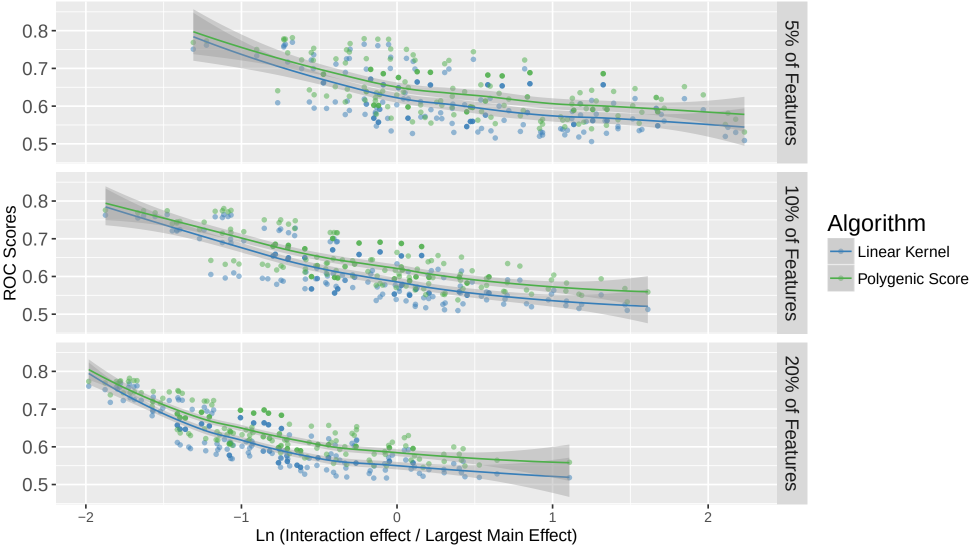


Figure 4.10.: Scatter plot showing the distribution of the median ROC scores for the polygenic score and linear kernel from the 20 CV permutations. The coloured lines represent the Loess regression curve, with the shaded 95% confidence interval region.

These figures convey some very useful information that aids in the interpretation of the results. The most apparent aspect is that all of the models show a decreasing level of performance with the increased contributions from the interactions. This effect is increased as more of the inputs features contribute to these interactions, as shown by the regression curves dropping more steeply, especially in the lower panels for the 20%

contributions. While this is interesting, it could be argued that this makes intuitive sense as these trials where the performance is dropping represent situations where the signal determining the outcome is contained in the interactions rather than presenting itself in the main effects. The sample space of the different combinations of interactions could be very large, and greatly exceed that of the number of samples available, which could explain the lower levels of performance seen. The more useful information can be seen in the Loess regression curves for all the models. In figures 4.8 and 4.9 where the RBF kernel is compared with the linear models, there is often a cross-over, from the RBF kernel performing worse for the main effects, but better for the interactions (this happens in all situations apart from when only 5% of the inputs were interacting for the RBF/polygenic score comparison, topmost panel of figure 4.9).

For the comparison of the RBF and linear kernels, the Loess line shows that the general trend of performance is reached quite early on for all the algorithms, and as more features are interacting, the RBF kernel shows improvements (albeit not statistically significant), even when the interaction coefficients are slightly smaller. The cross-over point in the 20% case happens at around -0.5. The cross-over points for the RBF kernel and polygenic scores, for the 10% and 20% panels, only appear at the very right end of the curves, so it must be considered that this could have happened purely due to chance. For the comparison of the two linear methods in figure 4.10, there are no cross-overs in performance. The only occasion where the linear kernel approaches the same level of performance is when the main effects are carrying far more of the signal, shown on the leftmost side of each panel. As the lines move towards the right sides, the linear kernel shows a consistent slight inferior performance to the polygenic score, so it is being more adversely affected by the presence of the interactions.

Inclusion of the OMNI chip samples

So far, the analysis in this chapter has only shown the results from using the samples genotyped on the Illumina 1M (I1M) chip. In chapter 3, for the trials using the larger threshold dataset, it was shown that the addition of extra samples from the OMNI chip aided the performance of the SVM models, but not for the polygenic score. A recap of this finding is shown in figure 4.11. Also in those trials, the RBF kernel showed superior performance over the linear kernel. With the addition of the extra chip information, there were an additional 4,233 samples, bringing the total up to 11,964.

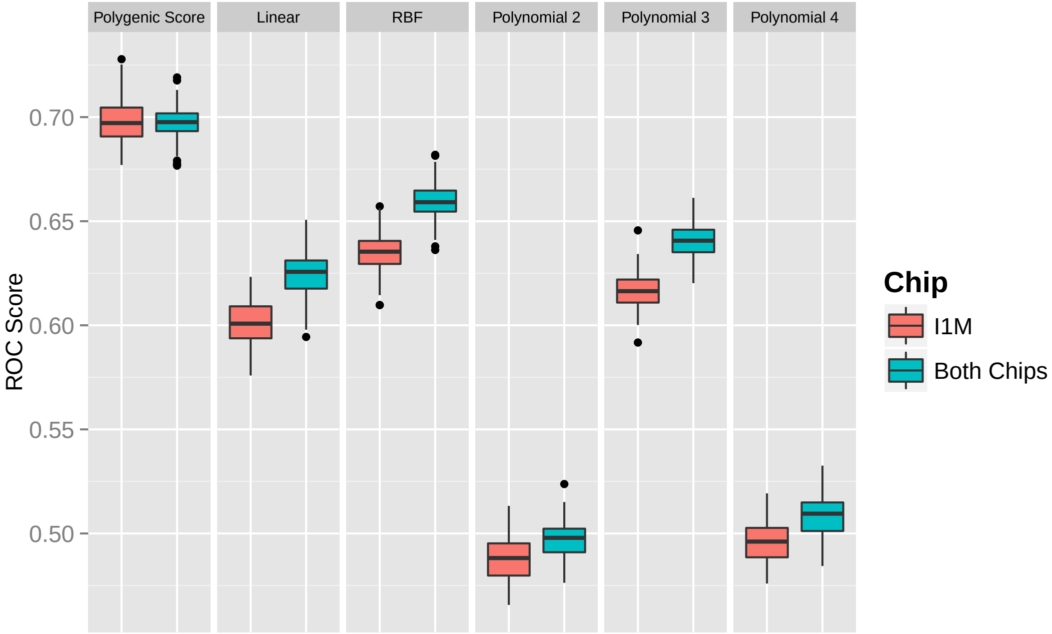


Figure 4.11.: Box plot showing the distributions of results for all the algorithms on the larger dataset of 4998 SNPs. The results show the performances for both the I1M chip only and the combined chip data.

Due to the pattern seen with the additional samples, it was of great interest to see if the inclusion of these would have an effect on the performances for the simulated data. All of the results are shown in the same format as earlier. The comparisons of the relative effect sizes of the main effects and interactions with the t-test performances can be seen in figures 4.12 and 4.13.

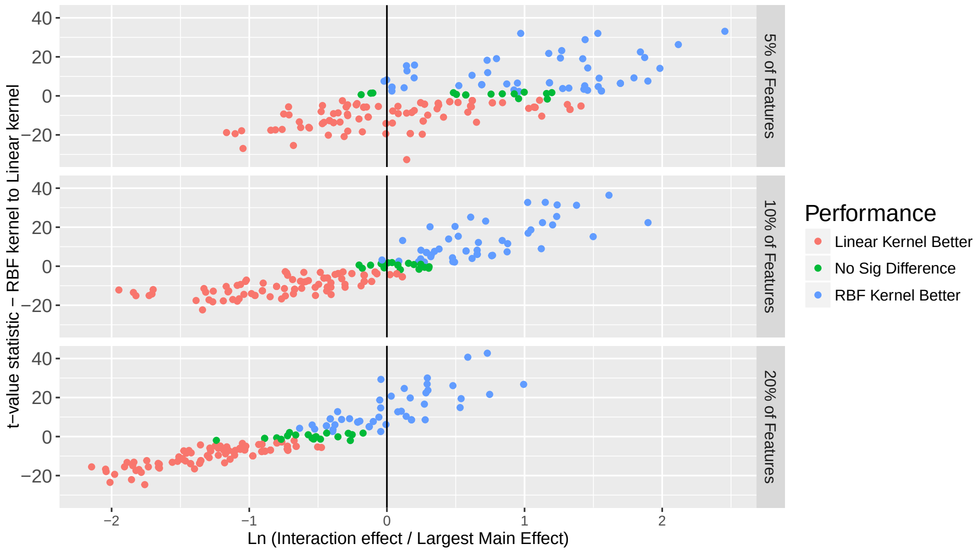


Figure 4.12.: Scatter plot showing the relationship between the median values of the log ratios of the interaction and respective largest main effect coefficients and the difference in performance of the RBF and linear kernels, assessed by pairwise t-tests; using the samples from both the I1M and OMNI chips. The colours represent if the difference in performance for the t-tests was significantly different at $p < .05$, and which algorithm was performing better if the difference was significant.

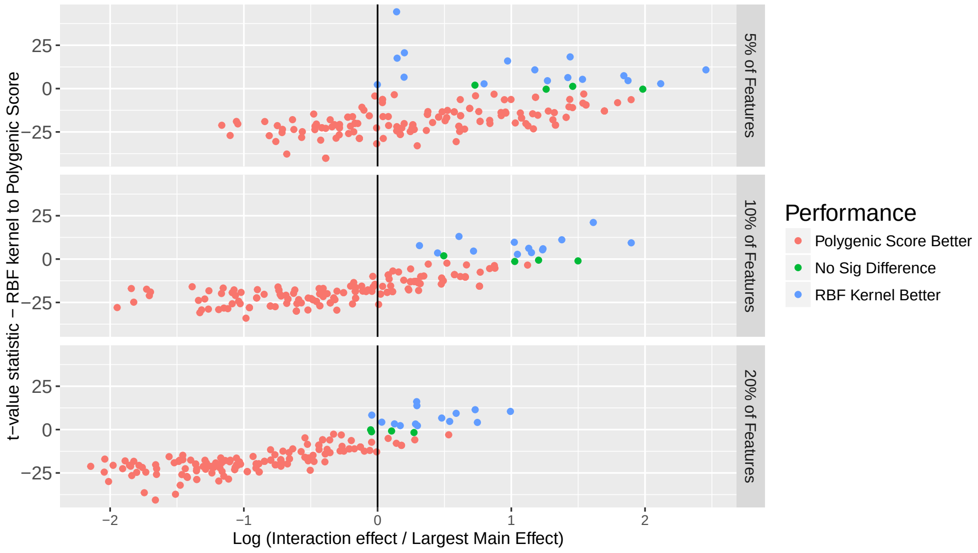


Figure 4.13.: Scatter plot showing the relationship between the median values of the log ratios of the interaction and respective largest main effect coefficients and the difference in performance of the RBF kernel and polygenic score, assessed by pairwise t-tests; using the samples from both the I1M and OMNI chips. The colours represent if the difference in performance for the t-tests was significantly different at $p < .05$, and which algorithm was performing better if the difference was significant.

Most of the differences between these two figures and the similar versions in the I1M only trials are quite subtle, so they are summarised in table 4.1. The layout of this table has been constructed to make comparing the results as easy as possible. The values for the relevant comparisons are placed together, with the information from both chips highlighted in grey. The value in the right column shows the exponent of the highest value seen in each trial where a non-significant value was still seen (the green points in the figures). The exponent is taken as the values on the x axis are on a logarithmic scale, and therefore these numbers reflect the relative size of the median interaction coefficients (the effect sizes) to the median main effects coefficients. The assumption is made that for all of these different trials, if the relative interaction values are greater than these listed here, then the RBF kernel will perform to a superior level.

Table 4.1.: Table showing the differences in the maximum points at which there was no difference seen between the RBF kernel with the linear methods for the two different sizes of samples. The values in the right column show the relative size of the interactions compared to the main effects at these threshold points. The information for the I1M chip only is plain, and the values for both chips is highlighted.

Comparison	Proportion	Chip Info	Max Non-Sig Value
RBF Kernel to Linear Kernel	5%	I1M only	5.29
		Both chips	4.1
	10%	I1M only	1.34
		Both chips	1.36
	20%	I1M only	0.93
		Both chips	0.84
RBF Kernel to Polygenic Score	5%	I1M only	8.81
		Both chips	7.27
	10%	I1M only	5.01
		Both chips	4.48
	20%	I1M only	1.9
		Both chips	1.7

The pattern is quite different for the various algorithm comparisons. When contrasting the RBF kernel with the linear kernel, there is a benefit seen for both the 5% and 20% proportions when including the extra samples, but the 10% proportion actually results in a slightly higher threshold. While this pattern does make the interpretation of the results difficult, this does not mean that there is no benefit to using the extra samples for the 10% cases, because if the scales of the y axis are compared between figure 4.6 and figure 4.12, then it can be seen that the t-values are so much greater when the additional samples are included, suggesting that when the RBF kernel is better, it is showing much more superior performance.

For the comparison between the RBF kernel and the polygenic score, there is always a benefit seen when using the additional samples of the OMNI chip. While the same pattern is present of the polygenic score showing strong performance in all cases except when the interactions are very strong, these threshold points are decreased; and in the case when 20% of the features are interacting, the RBF is showing superior performance for fairly moderate relative increases in interaction contributions, and there are only five red points (showing where the polygenic score performed significantly better) of the lower pane of figure 4.13 when the interaction contributions were larger than the main effects.

While these images show the changes in t-test results across the different coefficient ratios, it is again useful to examine how the actual ROC scores for the various algorithms change. This is shown in exactly the same manner as before in figures 4.14 to 4.16, with the points representing the median ROC score for each round of 20 CV permutations performed on each phenotype, and the trend lines showing the Loess regression curves.

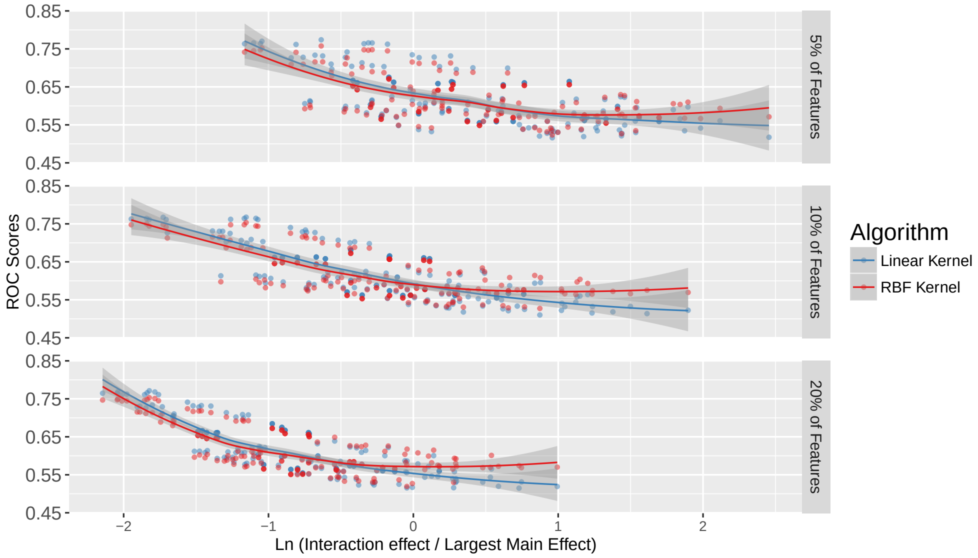


Figure 4.14.: Scatter plot showing the distribution of the median ROC scores for the RBF and linear kernels from the 20 CV permutations for both chips. The coloured lines represent the Loess regression curve, with the shaded 95% confidence interval region.

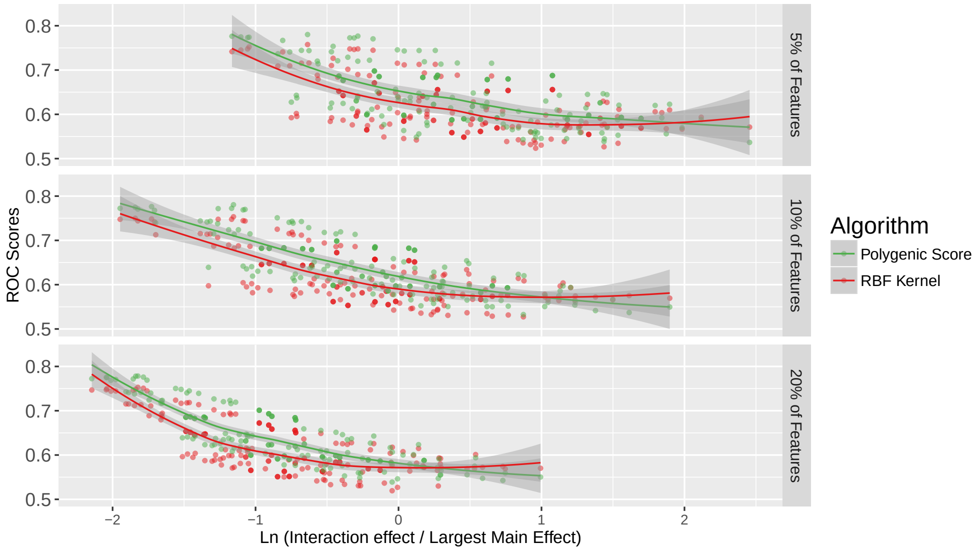


Figure 4.15.: Scatter plot showing the distribution of the median ROC scores for the RBF kernel and polygenic score from the 20 CV permutations for both chips. The coloured lines represent the Loess regression curve, with the shaded 95% confidence interval region.

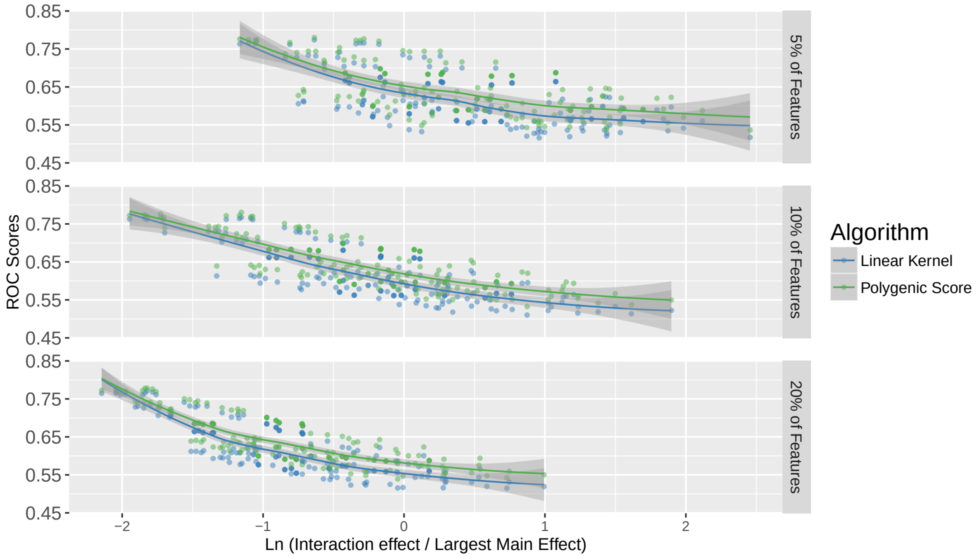


Figure 4.16.: Scatter plot showing the distribution of the median ROC scores for the polygenic score and linear kernel from the 20 CV permutations for both chips. The coloured lines represent the Loess regression curve, with the shaded 95% confidence interval region.

The trend lines in these figures possibly provide a better indication of the pattern of behaviour that is happening. For the comparisons between the RBF kernel and the two linear methods, there is always cross-over in performance for these trials, even for the 5% panel in the polygenic score comparison, and these cross-over points are occurring further to the left on the x axes, suggesting that the RBF kernels are detecting the presence of interactions when their effect sizes are smaller with the additional samples.

4.3.2. Using Nagelkerke's R^2 to examine effect size differences

The results presented so far have highlighted that the RBF kernel is showing an increase in performance over the linear kernel, and to a lesser extent the polygenic score, when there are greater contributions from the pairwise interactions terms. However, the methods used were somewhat fragmented, because separate logistic regression models were fitted for each pairwise combination of SNPs. In addition, the main effects were only ever calculated from the SNPs that were also involved in making the interaction terms; whereas the simulations were made from the main effects of all the SNPs, and the pairwise interactions from the subsets.

Due to these concerns, a new series of analyses were performed that took a more wholistic approach to the task of examining the main effects and interactions, and how these affected the performance of the different algorithms. This method involved making only two models: one using the main effects from all 125 SNPs, and another using all of the pairwise interaction terms from the different subsets of SNPs. The interaction terms were constructed in R using the `model.matrix` function.

This improved approach was considered to be the best way to examine the different levels of contribution from the main effects and interactions. However, the assessment method could not be the same as the one used in the previous trials. In those trials a separate model was built for every SNP pair, and the metric used was the ratio of the coefficients assigned to the largest main effect and interaction term by the model. The same cannot be done in these new trials, as all of the SNPs or interactions are being used at the same time for both models. Therefore, a metric that assesses a model as a whole was required. In linear regression, this can be easily achieved by using the R^2 metric, but as this was a categorical task, with a binary dependent variable, a *pseudo* R^2 term had to be used instead. One such example of this is Nagelkerke's R^2 (Nagelkerke, 1991), a very commonly used metric to assess the performance of logistic regression models, which was also featured in the Psychiatric Genetics Consortium - 2 (PGC-2) study (Ripke et al, 2014) and the earlier study by the International Schizophrenia Consortium (Purcell et al., 2009). It must be remembered however, that this is not an ideal metric that represents the same meaning as R^2 used in linear regression, and care must be taken in interpreting its value. It does not truly reflect the proportion of variance explained by the model, as is the case with R^2 in linear regression (Peng et al., 2002). Despite these concerns however, its use was deemed to be suitable in this situation as it was the relative difference in the values for the main effects and interactions that was of interest.

Method for the Nagelkerke's R^2 trials

These trials followed a very similar, iterative design pattern to the previous trials. The following procedure was carried out for each of the three different proportion levels of the number of SNPs in the different subsets. Each of the 180 different phenotype simulations per proportion level were analysed in turn. All 125 SNPs were featured in the main effect models and the same respective subsets of SNPs that were used to make the interactions were used to make the interaction models. This information was combined with the respective performances of the polygenic score, linear kernel and RBF kernels obtained from the 20 cross-validation shuffles for each simulated phenotype. The two different models of the main effects and interactions were made, and the Nagelkerke's R^2 was obtained for each model. Then pairwise t-tests were performed to compare the results of the RBF kernel with those from the linear kernel and the polygenic score. The main information obtained from this procedure was: the R^2 values for both the main effect and interaction models for each simulated phenotype, and the t-test results from the algorithm comparisons. This resulted in 540 different points of information across the three different subset proportions of 5, 10, and 20% of SNPs.

Results

The results for these trials are displayed in a different format to those that were shown earlier. In these plots, for each of the three proportion levels, the R^2 values for the main effects models are shown on the x axes, while the values for the pairwise interaction models are shown on the y axes. The two different values of R^2 are shown as points on a scatterplot on these axes. As the main focus of interest is how the results of the t-tests are affected by the different relative contributions of the main effects and interactions, this information must also be displayed. As the two dimensions are already being used, this was best shown using colour, very much in the fashion of a heatmap plot. The diagrams use colour gradients, which allow the same colour-mapping to be used across all the different algorithms comparisons. In all of the graphs shown here, the higher t-values, which represent a superior performance of the RBF kernel, are shown in red shades, the lower value shown in blue shades with green representing the values close to zero. The information in figures 4.17 and 4.18 show the results for the I1M chip only, and that in figures 4.19 and 4.20 show the results from using the information from both chips.

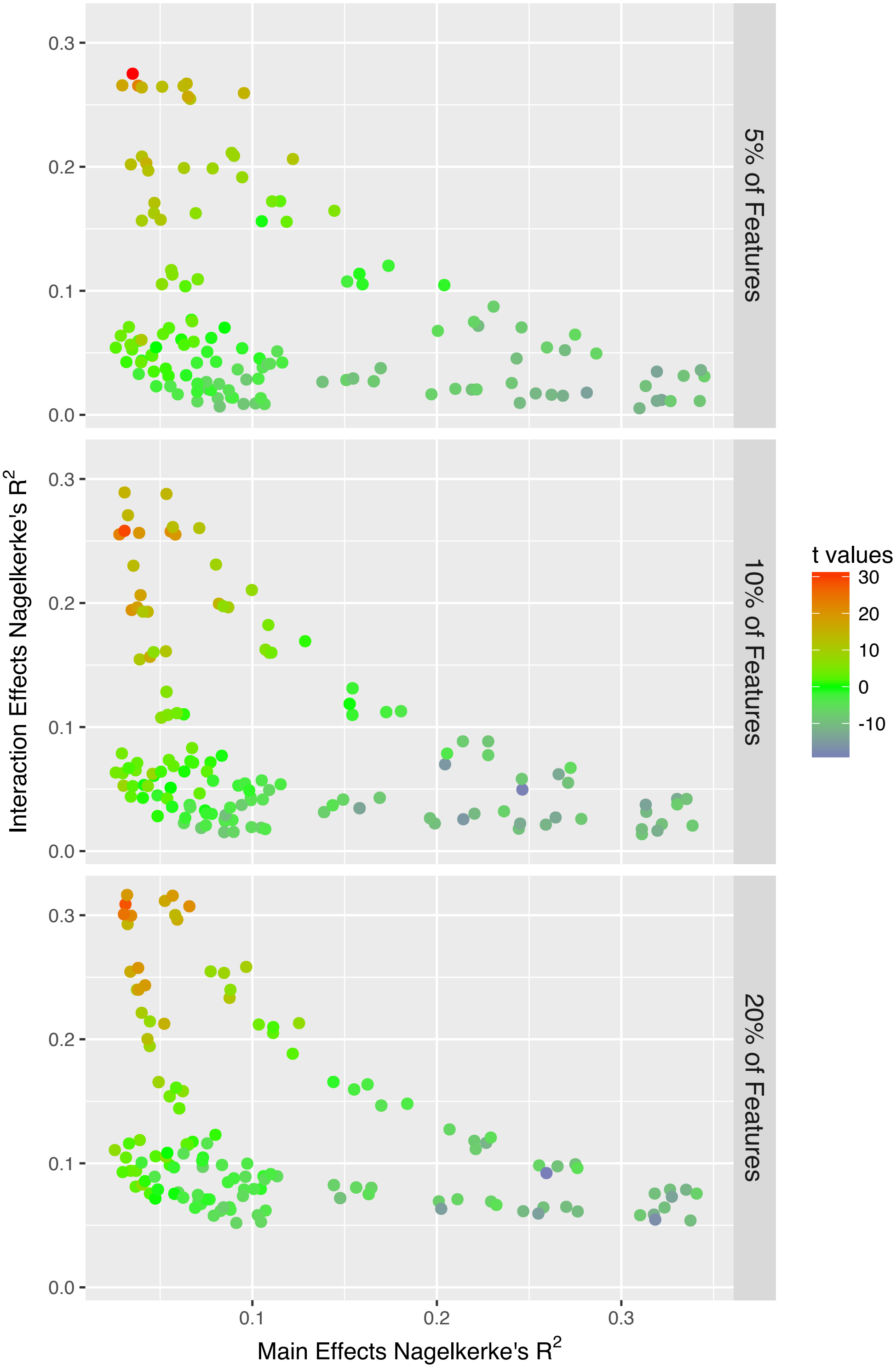


Figure 4.17.: Scatterplots showing how the values of the t-tests between the RBF and linear kernels change for the different values of Nagelkerke's R^2 for the main effects and interactions using the information from the I1M chip only

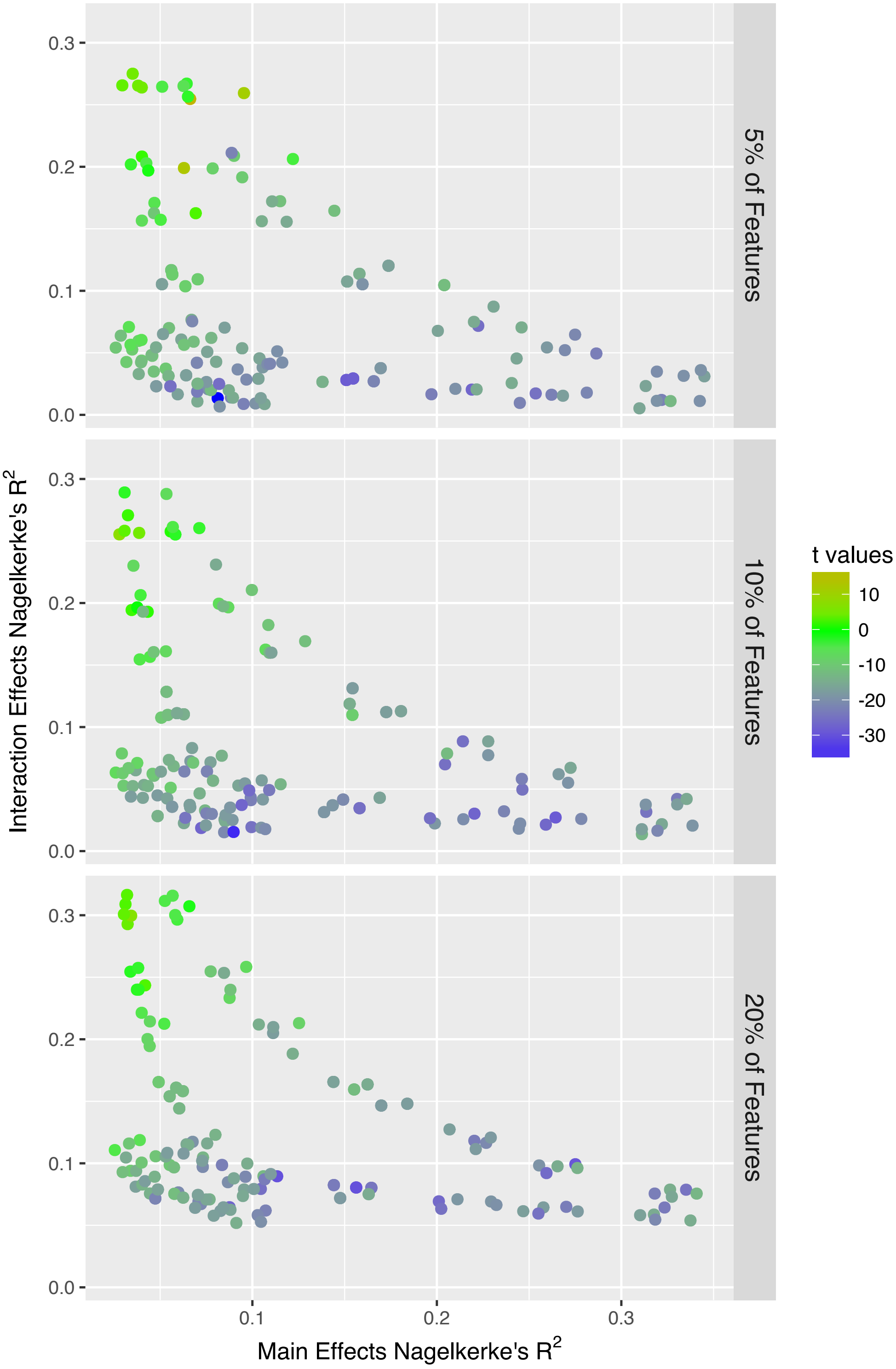


Figure 4.18.: Scatterplots showing how the values of the t-tests between the RBF kernel and polygenic score change for the different values of Nagelkerke's R^2 for the main effects and interactions using the information from the I1M chip only

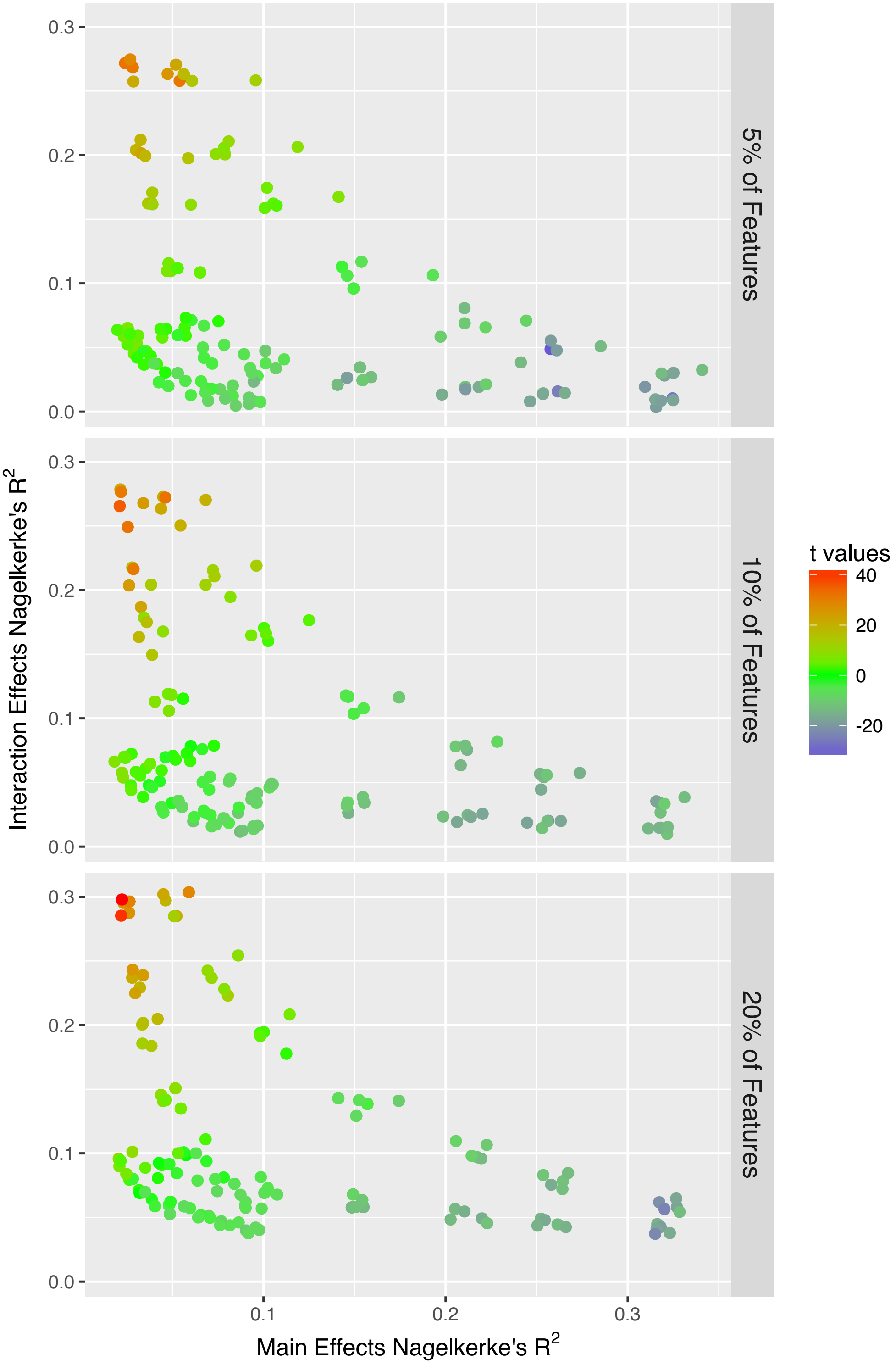


Figure 4.19.: Scatterplots showing how the values of the t-tests between the RBF and linear kernels change for the different values of Nagelkerke's R^2 for the main effects and interactions using the information from both chips

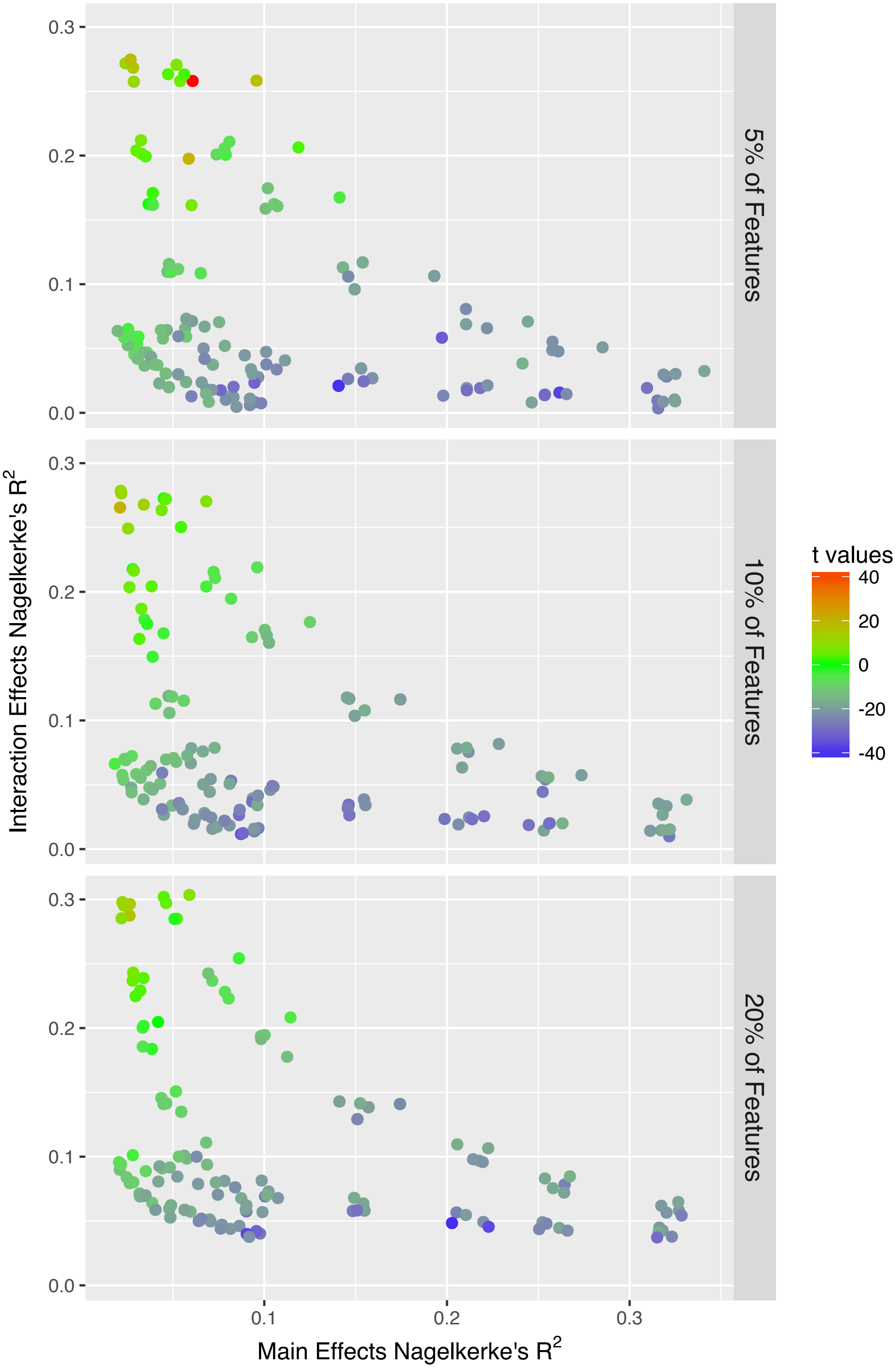


Figure 4.20.: Scatterplots showing how the values of the t-tests between the RBF kernel and polygenic score change for the different values of Nagelkerke's R^2 for the main effects and interactions using the information from both chips

There are a number of interesting patterns that can be seen in all of these four graphs. First of all, the general pattern is the same across all algorithm comparisons and interaction subset proportions. This is the "boomerang" shape of clumped curves that follows a shape similar to that of an exponential distribution. Also, if one looks carefully, then it appears to be the case that there are three groupings of these curves, one which is closer to the origin of the axes, and the other two, which move outwards and display a straighter pattern, and contain larger values on both axes. These patterns are undoubtedly due to the different combinations of parameters that were used to make the simulated phenotypes. However, the main result of interest is seen in the pattern of colours representing the t values from the pairwise t-tests between the different algorithms. While there are some deviations and fluctuations, the t values are generally higher in the favour of the RBF kernel when the main effects have a low R^2 value and the interactions have a high value. In the opposite situation, the lower values of t are shown, signalling that the RBF kernel is performing worse. This fits in with the same patterns that were seen in the earlier trials looking at each pair of SNPs individually, whereby it is only in the situation where the interactions are playing a more prominent role that the RBF begins to show its advantages. Also, the colour shading shows that when compared with the linear kernel, the RBF is showing a clear improvement, due to the greater presence of red shaded points; but when compared to the polygenic score, it only shows a slight improvement due to the large amount of green shaded points with high y axis and low x axis values. It is worth noting that there is a single point in top panel of figure 4.20, showing the comparison between the RBF kernel and the polygenic score for the data from both chips using a subset of 5% of SNPs to make the interactions. This point is red, and suggests a single point where the RBF kernel outperformed the polygenic score greatly. The data was inspected for this single point, and it is indeed an outlier when this particular combination of SNPs were used to create the simulated phenotypes.

Despite this single outlier, the main pattern observed in these wholistic analyses supports those found in the previous trials: the RBF kernel shows a relative increase in performance over both linear based models, when the pairwise interaction terms contributed more towards the creation of the phenotypes, which therefore resulted in higher Nagelkerke's R^2 values in the logistic regression models.

Investigating the clumped curves

As there was a clear pattern of clumped curves in the results that resemble a "boomerang" shape, some further investigations were carried out to ascertain what aspects of the parameters used in the simulations could have caused this effect. This process identified the different combinations of the probability values of 0.7 and 0.9 that were given to the main effects and interactions as being the cause. This effect can be seen in figure 4.21 below.

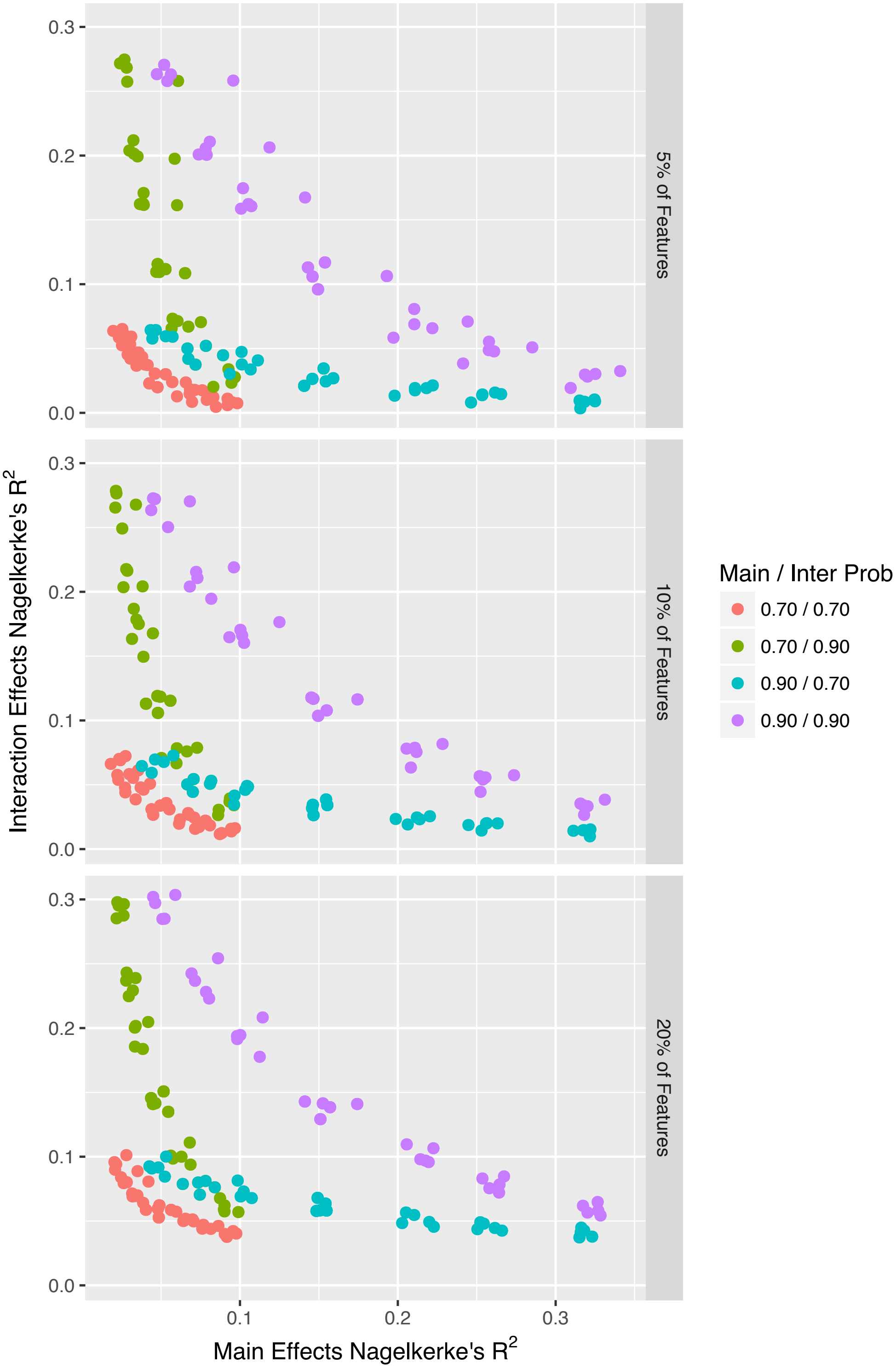


Figure 4.21.: Scatterplots showing how the four different combinations of probability values for the main effects and interactions have an effect on the Nagelkerke's R^2 values. Four different clumps can clearly be seen

These results show that there are actually four different clumped curves instead of three. The outer curves show the situations when both the main effects and interactions had the same probability values, and the inner curves show when they had different values. As would be expected, the increase in probability for an input type increases the potential effect size of these inputs in the models. So when both the main effects and interactions had a probability value of 0.7 in the simulations, the maximum R^2 effect was around 0.1. For this reason, in order to see an effective spread of results, and how they relate to different levels of performance, it is advised that any future attempts to replicate this study make use of the higher probability values in the simulations.

4.4. Discussion

The trials presented in this chapter were initially intended to try and explain if the main effects alone were enough to explain the common pattern seen for all the datasets, where the polygenic score performed best, with the linear, RBF and polynomial-3 kernels all performing at a similar level, but with a drastic drop in performance for the even value degrees of the polynomial kernel. This pattern was confirmed in section 4.2.1, and then the different combinations of main effects and interactions were compared in the remainder of the study.

As has been mentioned throughout this thesis, one of the key advantages to using the non-linear kernel methods in SVM algorithms is that they allow for the analysis of interactions without their explicit calculation. The disadvantage of using these kernel methods is that the interpretation of the results becomes more challenging, so for this reason, these interactions are explicitly calculated in the experiments in this chapter. In this situation, the curse of dimensionality could quickly become a serious problem, so for this reason, the dataset used was the smaller one with only the 125 GWAS significant SNPs, weighted by the Log Odds Ratio (LOR) obtained from the study carried out by the Psychiatric Genetics Consortium - 2 (PGC-2) (Ripke et al, 2014).

The simulation procedures have provided some rich information to help explain, to some level, what might be happening within the actual data. The first trials in section 4.2.1 dealt with simulated phenotypes that were made exclusively from either the main effects or the pairwise interactions between features. The main effect trials contained two main valuable sources of information: firstly, the aforementioned finding that having the signal determining the case/control status arising solely from main effects results in very poor performance of the even-valued degrees of the polynomial kernels, leading to the conclusion that these only work in situations where the input features are interacting between themselves, at least to some degree. But secondly, it was shown in figure 4.2 that if not all of the inputs were contributing some meaningful signal to the outcome, then the performance of the polygenic score drops considerably. This makes intuitive sense as the polygenic score is a single value representing the linear combination of all the small risk mutations, and if this value is in any way tampered with by completely nuisance inputs and variables, then there is no way to retrospectively separate out the noise from the signal. On the other hand, the SVM algorithms that are performing well (mainly the linear and RBF kernels) are not being affected by these non-contributing variables and are successfully filtering out their influence from the analysis.

This all means that the non-contributing SNPs are not being ignored completely, but they be assigned a low enough coefficient value that a high level of performance is maintained. These results also show a very useful application of using machine learning algorithms: the ability to separate out the relevant signals from a relatively high number of features that only contain noise. While this is not relevant so much in the work that has been shown so far, as in all the trials, the features are always selected based on their relevance to the target outcome (the case/control status), the material in chapter 5 provides examples of where new features are made from the mutations by combining them together in biologically meaningful ways, and using the SVM models to assess if any of these new features can be discarded as not carrying signal.

The trials using exclusively the pairwise interactions did not show any type of pattern in performance across the different algorithms that has been seen before in real data, so it is safe to assume that this does not resemble the signal present in the aetiology of schizophrenia at all. It did, however, point out that this is where the non-linear kernels do have a distinct advantage over the linear kernel and the polygenic score. It was of particular interest to see that these SVMs were showing resilience to the presence of noisy inputs, as similar levels of performance were seen when only a half or a third of the SNPs were used to make the interactions. Understandably, the polynomial-2 kernel performed best as it is designed to pick up exactly the types of interactions that were engineered in these trials, but it was extremely poor at detecting signal in main effects. The RBF kernel was performing to a very satisfactory level for the interactions, and also for the main effects, so for this reason of showing less volatility in performance than the polynomial-2 kernel, it was chosen as the best method to represent the non-linear techniques for the trials in the remainder of the chapter.

Following the experiments using exclusive main effects or interactions came the trials using different contributions of both as described in section 4.2. These simulation procedures were carried out in conjunction with statistical analysis using a GLM framework to assess the varying levels of the interactions and main effects, and how these differences related to the varying levels of performance seen in the three algorithms: the polygenic score, linear kernel and RBF kernel. These trials provided a rich array of information, which can aid in the interpretation of other results seen throughout this thesis. The analysis of the effects of the interactions were carried out in two different stages. The first looked at the main effects and interactions from only the SNPs used to make the interactions, and did this by looking at the isolated pairs of interacting SNPs. The second took a more wholistic approach; building one model for all of the main effects, and another for all of the interaction terms used to create the phenotypes. Both models were assessed using the Nagelkerke's R^2 metric.

In the first method, when a fewer number of the features were allowed to interact with each other, then the effect sizes of the interactions could often be much larger than those of the main effects, but despite the larger values, they did not offer much assistance in allowing the non-linear methods to show any significant improvements in performance. This can be best seen in figure 4.6, where the RBF kernel's performance is compared with that of the linear kernel. In the topmost panel, representing the trials when only 5% of the SNPs were used to make the interactions, there are still many data points to the right of the vertical line (where the interactions have larger effect size on average), which still show superior performance for the linear kernel. This was not the case when 10% of the SNPs were allowed to interact as in this situation, the RBF kernel was either performing at the same level or better than the linear kernel. When 20% of the SNPs were used to make the interactions, then having larger effect sizes of these *always* resulted in statistically significantly better performance of the RBF kernel. When the ROC scores were compared directly, it showed that the linear kernel was superior for signals dominated by the main effects, but this was soon replaced by superior performance of the RBF kernel with the increasing roles of the interactions. When higher proportions of the features are interacting, then this cross-over happens much sooner, with the RBF kernel showing an advantage, even when the coefficients of the interactions tend to be smaller than those of the main effects.

In the comparisons between the RBF kernel and the polygenic score however, a different pattern was seen. The RBF kernel did not show the same dominance as the influence of the interactions increased. This was a very surprising result, as it was expected that the polygenic score would really suffer in these situations in light of the initial findings seen in the exclusive interactions study (figures 4.3 and 4.4). When the actual ROC scores were compared with each other, there were some signs that there might have been a cross-over in performance to the RBF kernel doing better when the interaction coefficients were much larger, but this was not statistically significant. The main conclusion from these findings is that, while the polygenic score suffers when *only* the interactions are playing a role, it should be accepted that this situation is unlikely to happen this way in a biological process. In the more realistic situation of a combination of influences from main effects and interactions, the manner in which the polygenic score is created can actually capture the overall signal quite effectively.

The findings from the second method show a similar pattern of findings. In all of the results, figures 4.17 to 4.20, it can be seen that when the R^2 value is higher for the interaction effects, the relative performance of the RBF kernel over the alternative linear methods is increased. As the colour range is the same for all of these figures, it can be seen that the polygenic score is showing higher levels of performance over the linear kernel, as the RBF only approaches similar levels of performance with its favourable parameters over the polygenic score, whereas it shows far higher performance over the linear kernel at these same levels.

These mixed trials were then re-run with the additional 4,233 samples from the OMNI chip, and the results showed that the RBF kernel models really seemed to benefit from having the extra information. These results are best seen using the first method of looking at the interacting pairs separately, instead of taking the wholistic approach. The t-test comparisons showed that when the RBF kernel was performing better, the difference between its performance and the linear methods was increased, as seen by the greater scale in the y axes in figures 4.12 and 4.13. The Loess regression curves showed that the RBF kernel always benefited from the extra samples, especially when the comparison is made with the polygenic score.

One of the main caveats in this chapter is that it was restricted to examining the effects of pairwise interactions between input features, and this restriction is not realistic in a biological process whereby many higher order, complex interactions could be taking place. These higher order interactions are, however, very difficult to explicitly create and model, and the decision was taken to focus on completing a thorough analysis of the pairwise effects in order to get an impression of how the algorithms would behave under different situations. The assumption is made at this point that, if superior performance is seen in the RBF kernel when compared with the linear kernel, then there is strong possibility that some forms of interactions are occurring between the input features. Another interesting observation was that the performance of the linear kernel approached that of the polygenic score when there was less of a contribution from the interactions. It seems as though the polygenic score is more resilient to the presence of these interactions, so another assumption, albeit slightly cautious one, is made that if the performance of the linear kernel is inferior to the polygenic score, then this could be an indication of the presence of interactions in the data. This also occurred in the real data for the threshold dataset, as seen in figure 4.11; therefore the conclusion is made here that interactions are indeed occurring between this larger collection of variants in the real datasets for treatment-resistant schizophrenia.

The earlier comparisons between the linear kernel and the polygenic score also revealed an interesting pattern; when only subsets of the exclusive main effects were used, the linear kernel was able to identify those features that offered no information and continued to have high performance, while the polygenic score suffered. This finding led to the inspiration to carry out the work in the next chapter whereby the SVMs are used to assess the *relative importance* of input features into the models, when these input features are not the individual SNPs, but combinations of them, based on previous research, into collections of gene regions and gene sets.

5. Exploring the CLOZUK Genes and Gene sets

5.1. Introduction

The work carried out in the previous chapter on simulations aided in the interpretation of the findings from chapter 3, as well as providing justifications for new experiments. The studies carried out in this chapter build on these findings and interpretations, and attempt to gain a greater insight into the aetiology of treatment-resistant schizophrenia by preparing the genotyped data across the genomes of the samples into biologically meaningful collections of either genes or functionally related groups of genes, referred to as *gene sets*. As was the case in the previous chapters, the input values that are entered into the Support Vector Machine (SVM) models will be referred to as the *features* of the models, and a clear explanation of how these are made will be explained in detail.

As a brief recap of the critical results so far, the findings from chapter 3 showed that when only the 125 Genome-Wide Association Study (GWAS) significant Single Nucleotide Polymorphisms (SNPs) were used in the analysis, the linear kernel performed roughly at the same level as the polygenic score, but the non-linear kernels showed inferior performance. When the 4,998 SNPs were used in the larger threshold dataset, a different pattern was seen: none of the SVM kernels performed as well as the polygenic score, but a key difference was that the Radial-Basis Function (RBF) kernel was performing better than the linear kernel. Also, the performance for the SVMs was greatly improved with the addition of the extra samples from the Illumina Omni Express (OMNI) chip, while the scores seen in the polygenic score did not improve. In all of the trials carried out, the even-valued degrees of the polynomial kernels showed very poor performance, and while the polynomial-3 kernel performed better, it did not reach the same levels as the RBF kernel. The simulation procedures in chapter 4 provided some useful information to help interpret these findings. The poor performance of the even-valued polynomial kernels, and the high performance of the polygenic score suggested that the main effects of the SNPs were playing a prominent role in the signal, as this resembles the patterns seen in the simulations. In the smaller, GWAS significant dataset, the linear kernel performed at nearly the same level as the polygenic score, which suggests that there are very limited, or even no interactions occurring between the variants, as this also resembled the pattern seen in the simulations. This supports the conclusions made in the Psychiatric Genetics Consortium - 2 (PGC-2) study (Ripke et al, 2014) that there is no evidence for the presence of pair-wise interactions occurring between the SNPs. The simulation procedures suggest that the pattern of performances seen in the larger threshold dataset, both the superior RBF kernel performance and the difference in performance between the polygenic score and the linear kernel, could show evidence for the presence of interactions.

There is, however, one particular caveat with this conclusion. In this dataset, there were 4,998 SNPs featured, which is a very large increase in dimensionality. It is still presumed that the increased performance of the RBF kernel supports the presence of interactions, but it could be wise to treat the difference in performance between the polygenic score and the linear kernel with caution, as this could be due to the sheer number of input features. It can be safely assumed that all of the features were contributing to the signal, and none were introducing noise, as when this was this case in the simulations, the linear kernel was able to identify the non-used inputs and improve on the predictive power of the polygenic score, which is not occurring in the real data. This should not be surprising as the SNPs were selected for inclusion in the analysis based on their association with schizophrenia seen in the GWAS results of the PGC-2 study. In order to address the concern of high dimensionality, the work in this chapter looked to reduce it by combining collections of variants together into functionally related groups. As the simulation procedures showed that the linear kernel was particularly effective at identifying non-informative input features, the hypothesis for this chapter was that this kernel could be used to identify which collections of SNPs, grouped together into functional groups, were carrying more of a collective signal, and if any could be labelled as not contributing to any signal.

The task of the experiments carried out in this chapter was therefore to try and find a means of capturing more of the genetic signal from across the genome, but in such a way that does not result in very high-dimensional input to the machine learning algorithms. In order to successfully achieve this, the information from the SNP information had to be combined together in a biologically meaningful manner. Once these groupings are made, the SNPs contained within them can be used to make a score for this region in the same way that the polygenic score is calculated when the whole genome is taken as a group: calculating the mean GWAS Log Odds Ratio (LOR) weighted reference allele count per region, for each sample. This was done by grouping together SNPS in and around the gene-encoding regions, and in turn, the functional collections of gene sets (also referred to as pathways in the literature) to which these genes have been assigned to based on previous studies. A literature review of work in this area was provided in chapter 1 and a brief recap is provided here.

The work in this chapter builds upon recent findings made by a study looking into the enrichment of rare Copy Number Variant (CNV) mutations in protein coding regions grouped together by functional similarities (Pocklington et al., 2015). In this study, the researchers stated that they identified 134 gene sets pertaining to the function and development of the Central Nervous System (CNS). Most were derived from the Mouse Genome Informatics (MGI) database[1], an open resource to provide a means of gathering together findings from laboratory mice experiments in order to aid studies into human health and disease (Blake et al., 2013), but those pertaining to sets under the class category of "Subcellular Neuronal" were taken from a series of proteomic studies, which are listed in the paper. The genes in the Fragile-X Mental Retardation Protein (FMRP) set was derived from a study by Darnell et al. (2011) about ribosomal translocation on Messenger Ribose-Nucleic Acid (mRNA). These gene sets all fall into six broader categories, which will be referred to as the gene set *classes*. A summary of these, together with the number of gene sets that they each contain can be seen in table 5.1. This information on the gene sets and their classes can be used in conjunction with the locations of the protein coding regions of the human reference genome to collect SNPs together based on genes, gene sets, and gene set classes, all with decreasing numbers of resultant input features into the SVMs.

Table 5.1.: Table showing the six different classes of gene set, and the number of sets contained in each.

Gene set class Processes	**Number of Sets**
Behaviour	33
Cellular Morphology	12
Cellular Physiology	18
Development (CNS)	17
Region Tract Morphology	38
Subcellular Neuronal	16

The main caveat of using SNPs that fall within protein-coding gene regions is that they do not take into account any contributions of signal from the inter-genic regions of the genome. While these regions were once considered as areas of "junk" DNA, more recent studies have found that elements within these regions (especially the areas upstream and downstream of genes known as the $5'$ Un-Translated Region (UTR) and $3'$ UTR respec-

[1] Website providing information and list of related publications: `www.informatics.jax.org`

tively), can affect the regulation of gene-expression levels, particular among eukaryotes and other more complex organisms (Barrett et al., 2012; Liu et al., 2012). Therefore, to examine the possible contribution of any signal lying in these UTRs, two different versions of each dataset were created: one with only the variants strictly within the boundaries of the genes, and another including variants lying within the UTRs, flanking the genes (details provided in methods: section 5.2).

The focus of the trials carried out in this chapter was not just to try and improve the power to predict treatment-resistant schizophrenia status in samples, but also to try and glean some additional information from preparing the data in this manner, and to examine if certain genes and gene sets provide a greater level of contribution than simply the sum of the individual SNP components contained within them. As was mentioned in chapter 2, the SVM algorithm was chosen for the machine learning models as it can be modified to focus on either interpretability or predictive power; if a linear kernel is used, then the coefficient values assigned to the input features can be used as a measure of the importance of those features, while making use of non-linear kernels can improve predictability by looking for possible interactions at the expense of losing interpretability. For this reason, the main focus in this chapter will be on the linear kernels and the coefficients assigned to the input features.

Due to heightened interest in examining the roles of different gene sets, the results from the SVMs were compared with those from a recently developed software program that specifically examines the association of gene sets with disorders from GWAS data, called *MAGMA* (de Leeuw et al., 2015).

5.1.1. LibLinear SVM algorithm

As the main attention of this chapter was to focus on the linear kernels, a different version of the underlying optimisation algorithm for an SVM was used here: the *LibLinear* algorithm (Fan et al., 2008a). This algorithm is carried out from a collection of source code scripts written in the C++ programming language (Stroustrup, 1986), but can be interfaced to other languages such as Python. This is all managed by the source code in *scikit-learn*. This algorithm is a development of the *LIBSVM* algorithms used so far (Chang and Lin, 2011). It can only be used to build linear kernel based models, and therefore cannot be used to examine the effects of interactions; but, if this is the focus of interest, then there are several advantages to using it. It is optimised to be far more efficient at dealing with larger datasets, both in terms of sample size and the number of input features. But most importantly, it can use use two different types of coefficient regularisation penalty. When building a predictive model, it is important that it is not too specified, or *overfitted* to the data that was used to build it. One of the ways of avoiding this is to prevent too much importance being given to a small set of features, which could be playing a particular large contribution in the training samples. Another advantage of this is that it allows any high coefficient values assigned to the features to be interpreted as being important in a robust sense, in that they are large due to genuine importance and not because of any idiosyncrasies present in the training samples.

The LibLinear algorithm builds on this technique by allowing two different varieties of penalisation to be used, by minimising either the l^2 or l^1 norms of the vector of coefficients. More about this can be read in Hastie et al. (2001); who were the first researchers to develop the l^1 norm, also known as the "Lasso" penalty. Both of these are shown in equations 5.1 and 5.2. As can be seen, the l^1 method seeks to minimise the total of the absolute values of the vector elements, whereas the l^2 method is seeking to minimise the squares of the elements. The advantage of using the l^1 norm is that it allows zero values to be assigned to the features if they are deemed by the algorithm to genuinely contribute no valuable information. This can also be of use if the intention is to eliminate any features from further analysis.

$$l^1 = \sum_i^n |x_i| \tag{5.1}$$

$$l^2 = \sqrt{\sum_i^n x_i^2} \tag{5.2}$$

In all the trials in this chapter, both of these penalisation techniques were used to compare their performances and to gain information on the importance of the features, but in addition, the non-linear kernels were used to assess for possible interactions between genes and gene sets.

5.1.2. A note about predictive model intentions

In light of the findings presented so far, the conclusion was drawn that it is extremely unlikely that a suitable predictive model of treatment-resistant schizophrenia can be made from the genotyped data. Indeed, these are the same conclusions that were drawn concerning the polygenic score analysis in the PGC-2 study (Ripke et al, 2014). In fact, due to low prevalence of sufferers ($\sim 1\%$) in the general population, a model would have to show a predictive accuracy of very near to 100% in order for it to be of any clinical use. The reason for this is that if there is any error in predictive power, and the test were applied to members of the general population; then due to sampling bias towards controls instead of cases, the vast majority of errors would appear as false-positives, and would therefore cause undue concern to anyone receiving this information about their genetic risk load.

This means that the focus is now shifted onto trying to ascertain if using machine learning methods can aid researchers to gain more insight into the sources of possible genetic signal which could be contributing more to the disease. Indeed, by looking at functional groupings of protein-coding regions, this could possibly provide useful information about new directions for lab-based studies to identify novel medication targets. The predominant targets of anti-psychotic medication so far has been the dopamine D2 receptor (Collier et al., 2016; Ginovart and Kapur, 2012), so finding other potential medication targets would be a desirable outcome because up to a third of schizophrenia patients do not respond well to initial treatments and around 75% present with relapses and continued illness symptoms (Smith et al., 2009).

5.2. Methods

The preparation steps to carry out this work involved quite a number of additional steps to those described in chapter 3 and will be described in turn here. Of note, all of the inputs in this chapter make use of the LOR weighted allele counts, and therefore, whenever the polygenic score is mentioned, this refers to the weighted polygenic score. The main aim was to create a dataset for all of the samples with the following information:

- The weighted allele counts of the SNPs of interest
- The corresponding:
    - Gene
    - Gene set
    - Gene set group

5.2.1. Identification of the SNPs of interest

As before, the SNPs were selected by the clumping procedure in PLINK (Chang et al., 2014; Purcell et al., 2007), based on the same parameters used in both chapter 3 and the PGC-2 consortia study (Ripke et al, 2014) in the risk profile scoring section. The main difference in how this was performed in this chapter is that instead of clumping all the SNPs in the datasets, only those falling in gene regions (the so-called *genic SNPs*) were first selected before the clumping procedure was carried out, and the non-genic SNPs were removed. There was a specific reason for doing it in this manner: if clumping is performed before selecting based on location, it is extremely likely that many genic SNPs would not be selected due to more highly associated SNPs in a non-coding region nearby being in Linkage Disequilibrium (LD) with them. If only the genic SNPs from those that survived the clumping process were chosen, then there could be large regions of the genome not accounted for as genic SNPs would have been dropped in favour of a more highly associated non-genic SNP in the same LD region. By forcing the clump procedure to only act on genic SNPs to begin with, this issue is circumvented as the non-genic SNPs can no longer be selected as the index SNPs.

As mentioned in section 5.1, the analysis was carried out on two different types of dataset: one looking at only gene regions; and another looking at additional flanking regions both upstream and downstream of the gene borders. The size of the windows was chosen to replicate the methods carried out by the Network and Pathway Analysis Subgroup of the Psychiatric Genetics Consortium (O'Dushlaine et al, 2015), and set at 35 Kilo-Bases (KBs) upstream into the 5′ UTR and 10 KB downstream into the 3′ UTR. From this point, these two datasets will be referred to as the *genic-region* and *window-region* datasets. The genes were identified from the 134 gene sets that were chosen for the study of CNVs in gene sets by Pocklington et al. (2015).

The information on the location of the genes was obtained from Build 37.3 of the human genome from the NCBI database[1]. From this file, only the information with `'feature_type = GENE'` and `'group_label = GRCh37.p5-Primary Assembly'` were kept, to remove any information about Ribose-Nucleic Acid (RNA), UTR, and pseudo-gene regions. This file was then filtered to contain only the genes of interest that featured in the 134 gene sets. This provided an information file for the exact chromosome and start/stop locations for the genes of interest, and a modified file was made for the flanking window regions. This was done explicitly in Python to ensure that these regions were being specified correctly. The files were then used to identify all of the SNPs that fell within

[1] `ftp://ftp.ncbi.nlm.nih.gov/genomes/H_sapiens/ARCHIVE/BUILD.37.3/mapview/seq_gene.md.gz`

the genic areas of interest.

At this point, new sets of PLINK files were made with only the selected SNPs, so the clumping procedure to identify the index SNPs with the same parameters as before was carried out. Once this was complete, the PLINK files were filtered to contain only the index SNPs of interest and these were then recoded to show the reference allele count for each of the variants (using the `--recodeA` command again). A recap of the clumping parameters used is provided here:

p1	The significance threshold of the index SNPs	0.05
kb	KBs between index SNPs	500
r2	The r^2 value for the LD threshold	0.2

Combining this information with the Gene and Gene set information

As stated earlier, the aim of this study was to find a way of making a series of polygenic-type scores for different coding regions of the genome. The data had to therefore be presented in such a way that this information would be easy to obtain from the large amount of SNP based information. The LOR weighted SNP information was merged with the data about the gene and gene set locations. This whole process resulted in a large dataframe with the information on the weighted SNP counts, together with their corresponding genes and gene sets for each sample. The structure of this dataframe allowed database style `groupby` operations on the data, allowing polygenic scores, based on either the genes, gene sets, or gene set groups to be made very efficiently. One other aspect to point out here is that is was very common that SNPs could fall into multiple, overlapping genes; and that genes could feature in more than one gene set. There was a fair deal of duplication of the SNP data to account for all these combinations. Therefore, care had to be taken to ensure that duplicates were removed when making the scores.

For both of the datasets, all of the 134 gene sets from the Pocklington et al. (2015) study were featured, but they contained different numbers of index SNPs and genes. This was due to some genes only having index SNPs within their flanking regions and therefore only being represented in the window-region dataset. A summary of these values is provided in table 5.2.

As was the case with all the machine learning algorithms in this thesis, the feature values for these scores were always normalised or *scaled* to have mean 0 and and variance of 1 as part of a Cross-Validation (CV) pipeline when the models were being built.

Table 5.2.: The different numbers of index SNPs and genes featuring in the genic-region and window-region datasets.

	Genic-Region	**Window-Region**
Index SNPs	3,601	4,629
Genes	1,660	2,182

In all of the gene sets, there were a total of 4,144, so the genic region dataset captured 40% of these, and the window region captured 52.6%.

5.2.2. New Datasets providing extra SNP information

There were two further experiments carried out in this chapter to attempt to provide the models with additional information about the SNPs. These methods were simple to perform, and provided a means of enriching the data with information beyond that of the odds ratios from a GWAS. The first method tried to make use of the extra information about the SNPs in LD with the Index SNPs that is provided in the output from PLINK. The second was inspired by recent studies that have looked into the roles that the SNPs can play in the Loss of Function (LoF) of different genes, and the attempts to rank these by how deleterious the resulting mutation could be.

Weighting SNPs by Linkage Disequilibrium neighbours

The output from a PLINK clumping procedure not only provides the information on the SNP chosen to be the index for its LD block; it also contains information on the other SNPs that meet a p-value threshold, but are in LD with the index SNP, and are therefore not selected to be index SNPs themselves. The p-value threshold can be specifically set in the PLINK command with an additional command line flag: `--clump-p2`. As the threshold for the index SNPs was 0.05, this same value was used for the SNPs that will be referred to here as “secondary SNPs”. One can think of the resulting additional information as the number of SNPs which could have been selected, but fall within the LD region of a more highly significantly associated SNP.

The number of these secondary SNPs can vary greatly, from zero to several hundred; and none of this information is taken into account when making a polygenic score. However, an assumption could be made that: if there is a higher number of SNPs associated with a disorder, all falling in a region of the genome that is a single LD block; then there could be a higher chance of a true signal falling within this region. One must remember that these SNPs are most likely not the causal mutations of any pathogenicity, but instead are markers that could indicate the presence of deleterious mutations nearby in the genome. Therefore, it could be wise to not treat index SNPs, which have several secondary SNPs in their blocks of LD, in the same manner as index SNPs that stand isolated in their LD regions.

The process to incorporate this data was very simple: the LOR weighted allele counts were further weighted by the square roots of the total number of variants falling within their respective clumped LD region. The decision to use the square root instead of the original count number was due to the large variation seen in the data, and this prevented some of the input values from becoming too large.

As before, these new gene set scores were made by taking the mean score of all of the additionally weighted allele counts falling within each gene set region.

Incorporating Loss of Function information

The modified dataset used in these trials was created by incorporating information from a recent study examining predictions of probabilities that the function of different genes will be Loss of Function (LoF) intolerant (Lek et al., 2015). This is an attempt to describe how deleterious different mutations could be; possibly resulting in more serious diseases and phenotypes.

This work involved looking at the variant calls in the exomes of 60,707 individuals across different ethnic populations, all taken from the Exome Aggregation Consortium (ExAC) based at the Broad Institute. Their main focus was to look at the distribution of Protein Truncating Variants (PTVs), effectively: mutations that result in early stop-codons, therefore cutting short the production of the protein. Previous research by the same researchers has shown that the more damaging mutations are less frequently seen in the population as they are removed by negative selection and estimated that around 95% of mutations found to be more functionally relevant were rare (Tennessen et al., 2012).

Their method of finding the genes that were more LoF intolerant was to identify mutations in protein coding regions that show a much lower than expected rate of PTVs. The authors mentioned that this was a non-trivial task as the rates of these mutations are not uniformly distributed across the genome and gave the example that they cannot occur at CpG dinucleotide sites. In addition, different patterns and mutation rates had to be taken into account for the different populations across the samples. The authors describe using an Expectation Maximisation type algorithm to identify mutations that showed observed rates that were lower than expected rates. 3,230 mutations with an LoF probability ≥ 0.9 were described as being LoF intolerant.

Reassuringly, these results support previous findings that many of these PTVs occurred in known genes associated with haploinsufficiency, meaning that the copies from both strands of the DNA are needed to create the required amounts of the protein gene product, and a lack of protein can result in abnormal phenotypes. However, 79% of the mutations have not yet been identified as being associated with known phenotypes, and the suggestion is made that this could either be due to undiscovered disease associations or that the mutations could be so deleterious that they prove to be fatal during embryonic development. As could possibly be expected, the genes showing the highest probabilities of being LoF intolerant are those related to core biological mechanisms; some examples provided in the article include the splisosome, ribosome and proteosome components.

While these findings provide a rich insight into the possible effects of mutations, the authors mentioned some caveats: the samples were selected from previous clinical studies, based on either specifically having, or not having different disorders, so they do not reflect a random sample from across the whole population, and in addition, they pointed out that future work should look beyond the protein coding regions in sequencing data but also acknowledge the increased sample sizes that would be needed to do so.

In light of these findings, the researchers have made their data available to other researchers and was downloaded for use in the current study[1]. The values for the LoF probability used in this study were calculated from a subset of the 60,707 samples who were not patients of psychiatric disorders ($n = 45,376$). This dataset provided the information about the probability of the genes being deleteriously affected by any LoF mutation, and can be interpreted as a proxy for the functional importance of each gene.

The prepared data so far already contains information on all of the index SNPs and which of the different genes they can be assigned to. As was mentioned earlier, many of these SNPs can lie in multiple overlapping genes, but all of this information is contained within the dataframe. As the data has been prepared in this way, the SNPs can easily

[1] `ftp://ftp.broadinstitute.org/pub/ExAC_release/release0.3.1/functional_gene_constraint/`

be weighted by the LoF probability of the genes that they are situated in. Of course, many SNPs in multiple genes will be given different values based on those genes, but as the values are represented separately in the dataframe, this poses no problem, and they can still be grouped together during the aggregation procedure.

The motivation behind using this technique is that the genes with a lower LoF probability may not be playing so much of a role in the development of any disorder and therefore this would down-regulate the value of the contribution that their SNPs will make to the gene and gene set scores.

The LoF information did not contain data on all of the genes used in the other analyses in this chapter, so some could not be included in this analysis. A summary of these numbers is provided in table 5.3.

Table 5.3.: The different numbers of index SNPs and genes featuring in the genic-region and window-region datasets when the Loss of Function (LoF) information is included.

	Genic-Region	**Window-Region**
Index SNPs	3,513	4,498
Genes	1,615	2,112

5.2.3. Cross-Validation Parameter tuning and Permutation testing

The same standard machine learning practice of using CV to find the best parameters had to be carried out during the model building process. As the main aim of these models was now to gain extra insight from finding the different ways of preparing the data, there was less focus on really fine-tuning the hyper-parameters as was the case in chapter 3. For the trials using only the linear kernels, this step was naturally made simpler as there was only the **C** parameter to tune. So instead of sampling values from an exponential distribution, a set of fixed parameters for a grid-based search were provided[1]. These were the values in the set $\{0.001, 0.01, 0.1, 1, 10\}$.

As there were fewer parameter value samples to chose from this time, instead of performing 4-fold CV, a stratified shuffle-split procedure was carried out - maintaining the relative sizes of train–75%/test–25%. To recap from chapter 2, this is very similar to k-fold CV, but the main difference is: instead of just partitioning the data into equal size blocks, with each block playing the role of the test-set once, a random selection of samples are

[1] Details of grid searching for CV procedures were discussed in chapter 2

chosen at each iteration. This means that a sample can now be in the testing set on multiple occasions. As this was still carried out in a stratified manner, the proportion of cases to controls was maintained throughout. This procedure was carried out using the `StratifiedShuffleSplit` function in *scikit-learn*. As the values for the inputs (the combined scores per gene, gene set and gene set class) were scaled in z-scores based on the distribution of values in each training split, the `Pipeline` function was again used.

For the parameter-search CV, 10 shuffled splits of the data were performed for each value, again using the random number seed in Python to ensure that each value was tested in the same manner. Once the suitable values had been found, the model was then run on different train/test permutations of the whole dataset 100 times - with a random number seed set again. It is these distributions of scores that are presented in the results.

The best performing model from this procedure was then used to examine the coefficients that were assigned to the gene set input features. A similar shuffled CV procedure with splits of 75%/25% was used for 100 iterations. The main difference in these trials was that the coefficient values of the gene sets were recorded instead of the Receiver Operating Characteristic (ROC) scores.

5.3. Results

5.3.1. Gene sets and Gene set classes – Linear Kernels

The first trials were performed looking at using the scores for either the 134 gene sets or the 6 gene set classes. This was carried out for the polygenic score, as well as the linear models of the Linear SVM and the Multiple Logistic Regression model. For the two multi-input models, both types of coefficient penalty method were performed (l^1 and l^2). The results for all of these algorithms, used on both of the input types can be seen in figure 5.1. As before, the metric used to measure the performance was the ROC curve. The vertical panels represent the algorithms used, the horizontal ones show the two different types of input used and the different coloured boxes represent the two different types of border region used when selecting the index SNPs. One particular point of importance is that the polygenic score is the same for both of the input types (the gene sets and the gene set classes) as they were made from exactly the same collection of SNPs, and explains why the spread of scores looks the same in both cases. The results show the distribution of scores from the 100 train/test permutations of the samples. These were all done using the same hyper-parameters which were obtained during the CV process. The values of these parameters can be seen in table 5.4, which shows that when the greater number of features were used for the gene set inputs, the values of **C** were much lower.

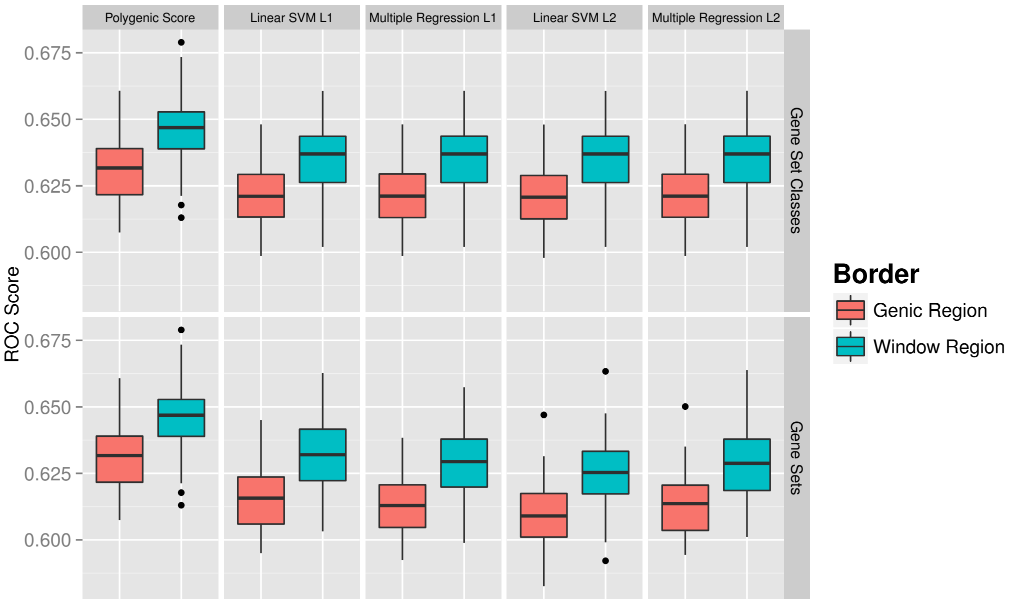


Figure 5.1.: Box plot showing the distribution of scores for both the gene sets and gene set classes for all of the different algorithms used. The colours represent the different boundary regions used. Note that the polygenic score used is the same in both cases.

Table 5.4.: The values of **C** chosen during the cross validation procedure.

Input Type	Algorithm	Border Region	C Value
Gene set classes	Linear SVM l^1	Genes Only	0.1
		UTR Window	1
	Multiple Regression l^1	Genes Only	1
		UTR Window	10
	Linear SVM l^2	Genes Only	10
		UTR Window	1
	Multiple Regression l^2	Genes Only	0.1
		UTR Window	1
Gene sets	Linear SVM l^1	Genes Only	0.01
		UTR Window	0.01
	Multiple Regression l^1	Genes Only	0.01
		UTR Window	0.01
	Linear SVM l^2	Genes Only	0.001
		UTR Window	0.001
	Multiple Regression l^2	Genes Only	0.001
		UTR Window	0.001

There are some points of interest in this figure. The polygenic score is still performing above the multi-input models; however, it is not performing to the same levels seen in chapter 3 due to the fact that only a subset of genes have been used in this study, and therefore the overall signal in the data is from a smaller proportion of the genome. When using the six features of the gene set classes, the performance is very constant across all the algorithms, but this is not the case when all of the 134 separate gene set features are used with the l^1 SVM showing the better results. Probably the clearest result from this trial is that including the SNPs in the flanking UTRs regions results in a marked improvement in performance across all of the algorithms.

As the slight differences in the median lines for the box plots can be difficult to see in the figure, the actual values are shown in table 5.5.

Table 5.5.: Table showing the median result for the performance of all the algorithms on both the gene set classes and the gene sets for the two different border regions.

Input Type	**Algorithm**	**Border Region**	**Median**
Gene set classes	Polygenic Score	Genes Only	0.631714
		UTR Window	0.646868
	Linear SVM l^1	Genes Only	0.621072
		UTR Window	0.636987
	Multiple Regression l^1	Genes Only	0.621140
		UTR Window	0.636982
	Linear SVM l^2	Genes Only	0.620737
		UTR Window	0.636981
	Multiple Regression l^2	Genes Only	0.621128
		UTR Window	0.636982
Gene sets	Polygenic Score	Genes Only	0.631714
		UTR Window	0.646868
	Linear SVM l^1	Genes Only	0.615686
		UTR Window	0.632007
	Multiple Regression l^1	Genes Only	0.612915
		UTR Window	0.629418
	Linear SVM l^2	Genes Only	0.608994
		UTR Window	0.625341
	Multiple Regression l^2	Genes Only	0.613642
		UTR Window	0.628788

Care should be taken when comparing the polygenic score to the linear kernel. In the simulations reported in chapter 4, it was suggested that a drop in performance of the linear kernel in respect to the polygenic score could be due to the effects of interactions between the features. The situation is slightly different here as the relationship between the two models is not quite the same. In the previous chapters, the polygenic score was made from exactly the same features which were entered into the SVMs, and each feature was only included once. In this chapter however, while the polygenic score only took the information from each individual SNP once, for the inputs to the SVMs, there were many cases where a SNP featured in multiple gene set and gene set classes. This makes the comparison between the two more complicated. In fact, when these trials were first carried out, an error was made in that the polygenic score was incorrectly made by taking the mean of all the gene set scores. This resulted in the linear kernel showing a superior level of performance that had to be dismissed, but highlighted the fact that the polygenic score and the SVMs were not being given the same inputs.

Examining the Coefficients as an importance measure for the gene sets

It was not surprising to see that the ROC scores seen here were similar to those seen in the SNP based analysis (in fact, a little lower given the fact that less of the genome was considered), as these machine learning models are performing a linear combination of the features. The main point of interest however, was the relative importance that these models gave to the input features provided. By examining these, it could be possible to glean more information as to which functionally connected variants could be contributing, in a manner that is not just a simple summation of their individual contributions.

A summary of the results from pairwise t-tests, looking at the different levels of performance between the l^1 and l^2 penalty methods can be seen in table B.1 in appendix B. The most significant differences were seen for the linear SVMs, using the l^1 penalty for the gene set inputs. This was therefore chosen as the penalty method to be used for these permutations, using the respective values of $\mathbf{C}$ shown in table 5.4.

All of the results for these trials can be seen in figures 5.2 to 5.5. There is quite a lot of information given here as the results are shown for both of the algorithms, for both different types of input feature, across the two different boundary regions (genic only and those with UTR flanking regions). For the gene set classes, all six of them are shown, but in the case of the gene sets, only the 15 with the highest absolute values (ie. not dependent on being positive or negative) based on the SVM algorithm are shown. The coefficients from the SVM are chosen instead of the regression model for two reasons: the main focus of this thesis was to examine the potentials of using machine learning models over traditional statistical methods, but more importantly, the linear SVM is attempting to fit a linear hyperplane in a non-transformed input space, therefore the coefficients can be directed interpreted as a vector pointing towards this decision boundary. In comparison, logistic regression models perform an exponential transform of the linear combination of inputs which, while allowing for classification tasks to be performed, makes the interpretation of the coefficients less clear.

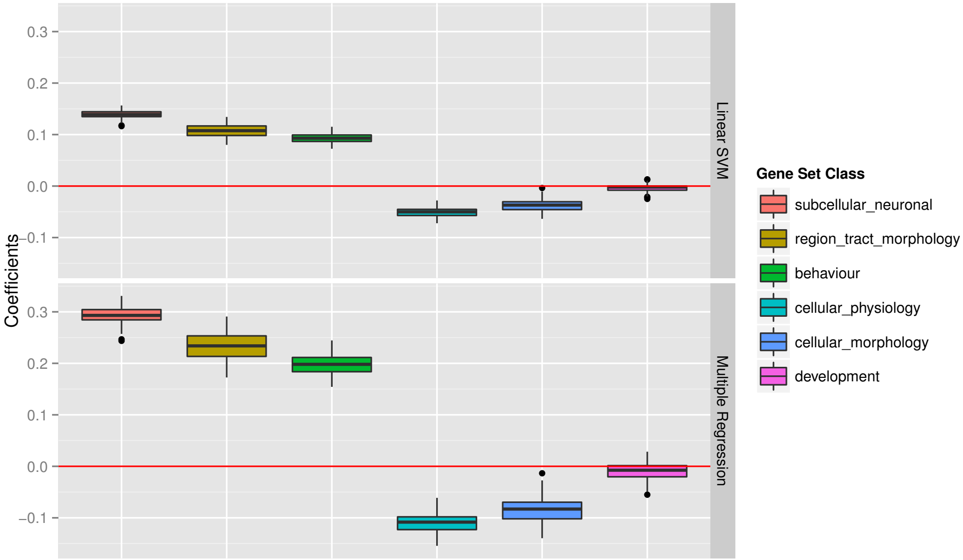


Figure 5.2.: Box plots showing the distribution of coefficients assigned to the gene set classes for both algorithms using the l^1 penalty for the scores based on SNPs in gene regions only.

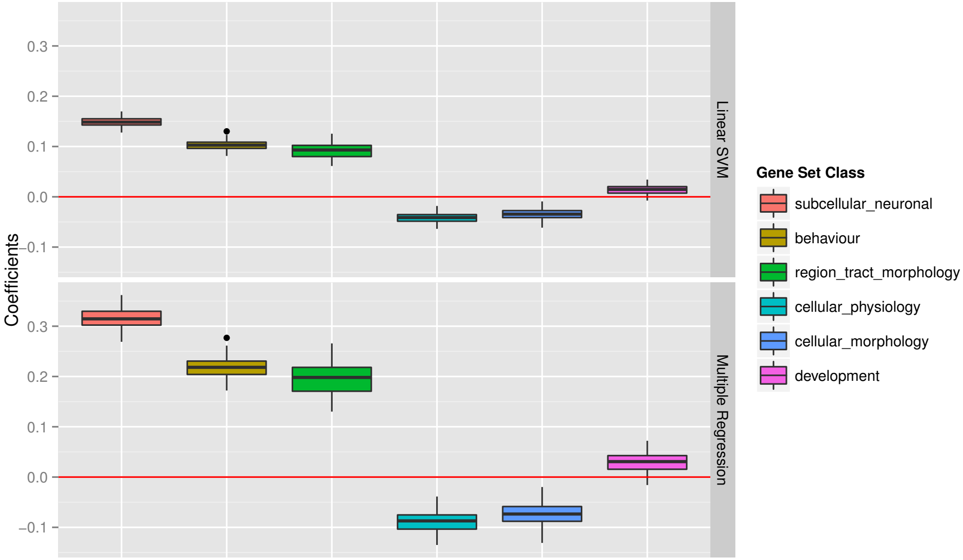


Figure 5.3.: Box plots showing the distribution of coefficients assigned to the gene set classes for both algorithms using the l^1 penalty for the scores based on SNPs in the gene regions and the flanking UTRs.

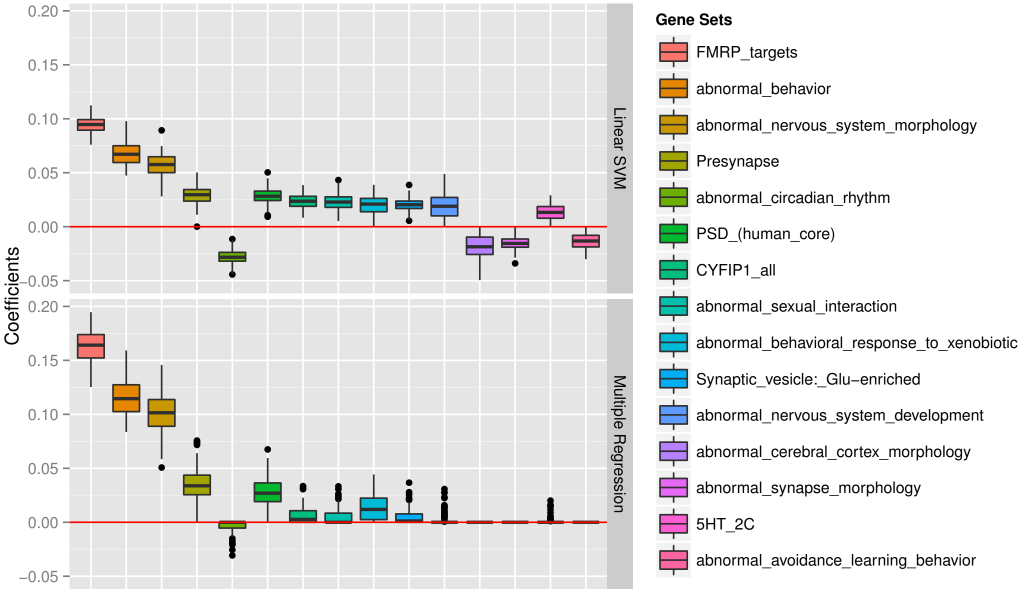


Figure 5.4.: Box plots showing the distribution of coefficients assigned to the individual gene sets for both algorithms using the l^1 penalty for the scores based on SNPs in the gene regions only.

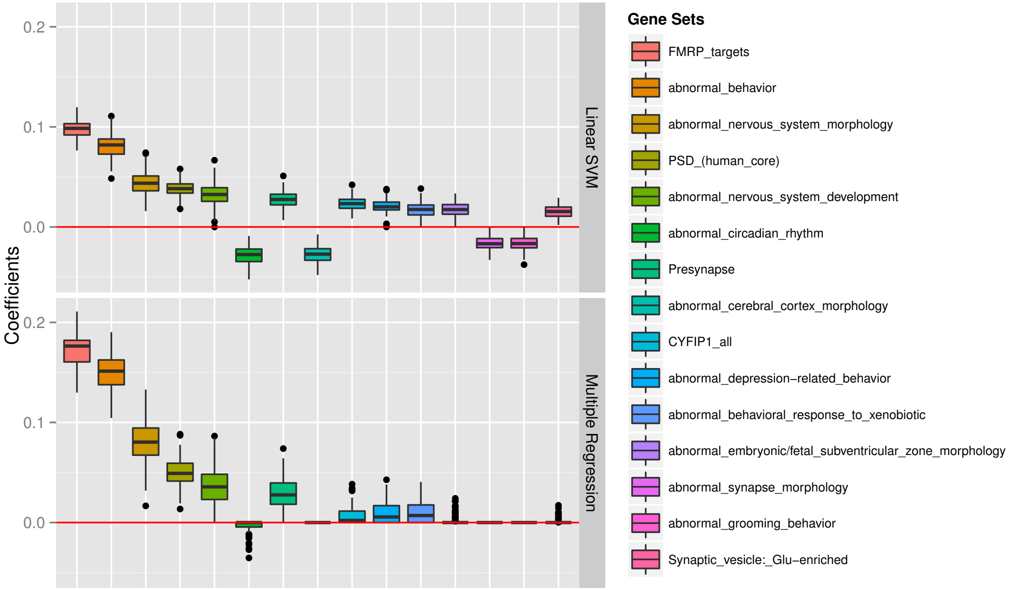


Figure 5.5.: Box plots showing the distribution of coefficients assigned to the individual gene sets for both algorithms using the l^1 penalty for the scores based on SNPs in the gene regions and the flanking UTRs.

There are many findings raised in these results that will be discussed in turn. For the gene set classes, there is a positive contribution from the three classes - subcellular neuronal, behaviour, and region tract morphology. The two classes of cellular morphology and cellular physiology provide negative coefficients while the development class seems to play the least important role of all. It is not clear at this stage what the differences between the positive and negative coefficients mean, but this is discussed in more detail in the next section.

Figures 5.4 to 5.5 show that for both boundary regions, it is the gene set labelled "FMRP targets" which always performs best. The second best performing set is the "abnormal behaviour" set, although this is less surprising as it is in itself a kind of "meta-set" comprising all of the sets pertaining to behaviour. The set, "abnormal nervous system morphology", also ranks highly in all cases, although this effect is diminished when the flanking UTRs are included in the analysis, suggesting that most of the relevant signal lies within the boundaries of the genes themselves.

After seeing these results, a few checks had to be made before more concrete interpretations were drawn. The fact that some features for both input types were assigned negative values is interesting, and therefore the distributions of the combined SNP scores that made these values had to be examined. It could have been the case that the scores for these features were negative themselves, thus requiring negative coefficients to make their signal relevant.

Key further analysis was required however to examine what the source of the stronger signal from within the FMRP targets set was. This set refers to the genes whose mRNA are targets of the Fragile-X Mental Retardation Protein (FMRP). A more detailed review of this protein and its functions are provided in the discussion in section 5.4; but in brief, it is a protein implicated in the transportation of mRNA from the cell nucleus to the synapse, a lack of which is responsible for one of the most common forms of mental retardation (Huber et al., 2002; Feng et al., 1997; Fromer et al., 2014).

Investigating the Negative Coefficients

In order to investigate the reason behind the negative coefficients, an examination of the average values of the inputs was made. This was done by looking first at the mean values across all samples for the six gene set classes and the four gene sets of FMRP targets, abnormal behaviour, abnormal circadian rhythm and abnormal cerebral cortex morphology, as these four sets showed noticeable positive or negative coefficient values. The values in tables 5.6 to 5.7 show the mean LOR values for the cases and the controls for the trials using the flanking UTRs. In these tables it is the mean value that is shown, in contrast to the median that is shown in the figures. The reason for this is that it was

Table 5.7.: A comparison of the mean-average LOR values assigned to four of the gene sets, selected as they show examples of larger positive and negative values.

Gene set	Mean LOR - Cases	Mean LOR - Controls
FMRP Targets	-0.002572	-0.002843
Abnormal Behaviour	-0.001881	-0.002086
Abnormal Circadian Rhythm	0.002509	0.002529
Abnormal Cerebral Cortex Morphology	-0.001868	-0.002026

the mean value per class or gene set that was entered into the models whereas the middle line of the box plots shows the median score of each distribution. In both tables, the gene sets and gene set classes that were given negative coefficients are shaded.

Table 5.6.: A comparison of the mean LOR values assigned to the gene set classes. Values shown are for the cases and controls separately.

Gene set class	Mean LOR - Cases	Mean LOR - Controls
Subcellular Neuronal	-0.001266	-0.001513
Behaviour	-0.001881	-0.002086
Region Tract Morphology	-0.002395	-0.002604
Cellular Physiology	-0.001969	-0.002154
Cellular Morphology	-0.002524	-0.002725
Development	-0.002152	-0.002350

The only example from the two tables that shows an instance where the controls have a larger mean value than the cases for the abnormal circadian rhythm gene set; in all of the other gene sets and gene set classes, the opposite is seen. This pattern likely explains why that particular gene set has negative values, but not the other three situations being considered. It is difficult to fully dissect what is happening inside the model calculations; it could be that the models are interpreting high LOR values as being protective. But it must be remembered that the ROC values did not show very high levels of predictive performance, so maybe those samples that the models incorrectly classified did show the opposing patterns for the LOR.

5.3.2. Examining the SNPs and genes within the gene sets and gene set classes

Before conclusions can be drawn about the results seen in the coefficient values, other aspects of the gene sets and gene set classes must be taken into account. A recent article has pointed out that care must be taken when interpreting findings involving FMRP results, as the genes involved in brain function tend to be large, and any enriched findings in FMRP could actually just be variants seen in overlapping highly-expressed brain genes (Ouwenga and Dougherty, 2015).

In order to examine this, a series of histograms were made showing selected summary information from all of the different gene sets, and how FMRP specifically fit into these distributions of scores. By doing this, it was possible to narrow down the reasons for this gene set performing particularly well.

The following data summaries were performed, all were calculated for each gene set:

Mean Gene Size The mean size in Base Pair (BP) of all the genes in the set

Gene Number The number of genes in the set containing at least one index SNP

Mean LOR The score that was entered into the models

SNP Number The total number of separate index SNPs in the set

SNPs per Gene $\frac{\text{SNP Number}}{\text{Gene Number}}$ - Average number of index SNPs per gene

SNP Density $\frac{\text{SNP Number}}{\text{Mean Gene Size}}$ - Average proportion of SNPs that are index SNPs

The distributions of all of these scores across all the 134 gene sets can be seen in figures 5.6 to 5.11. In all of these cases, the vertical black lines show each of the particular values for FMRP for that information.

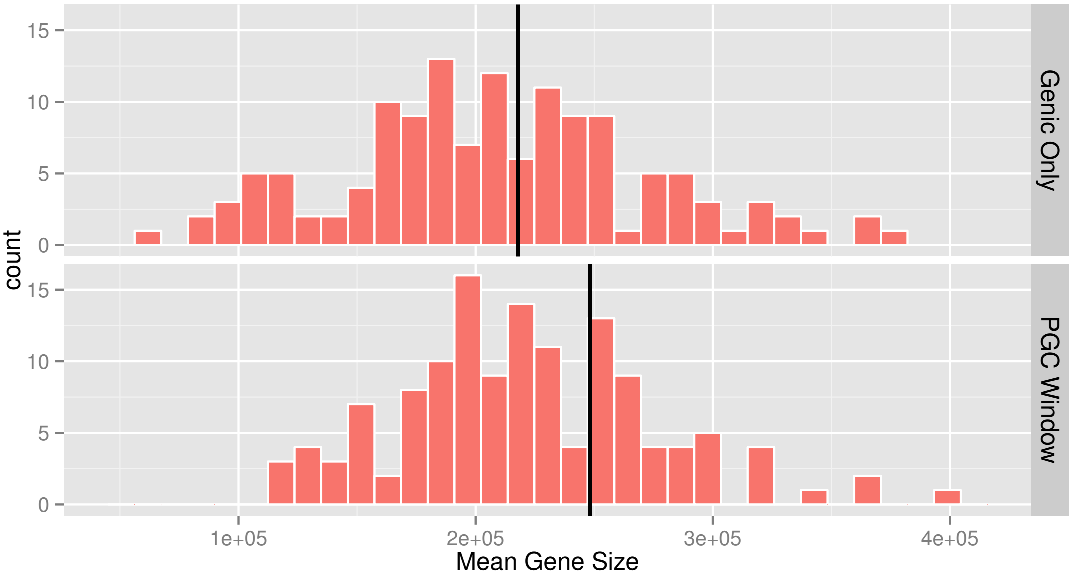


Figure 5.6.: Histogram showing the distribution of mean genes across all the gene sets.

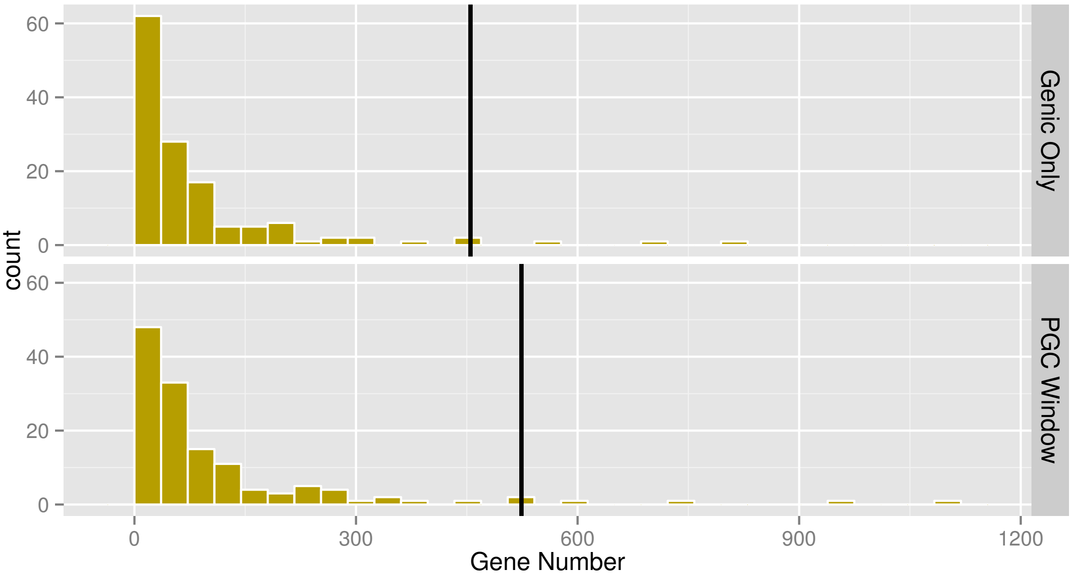


Figure 5.7.: Histogram showing the distribution of total gene number across all the gene sets.

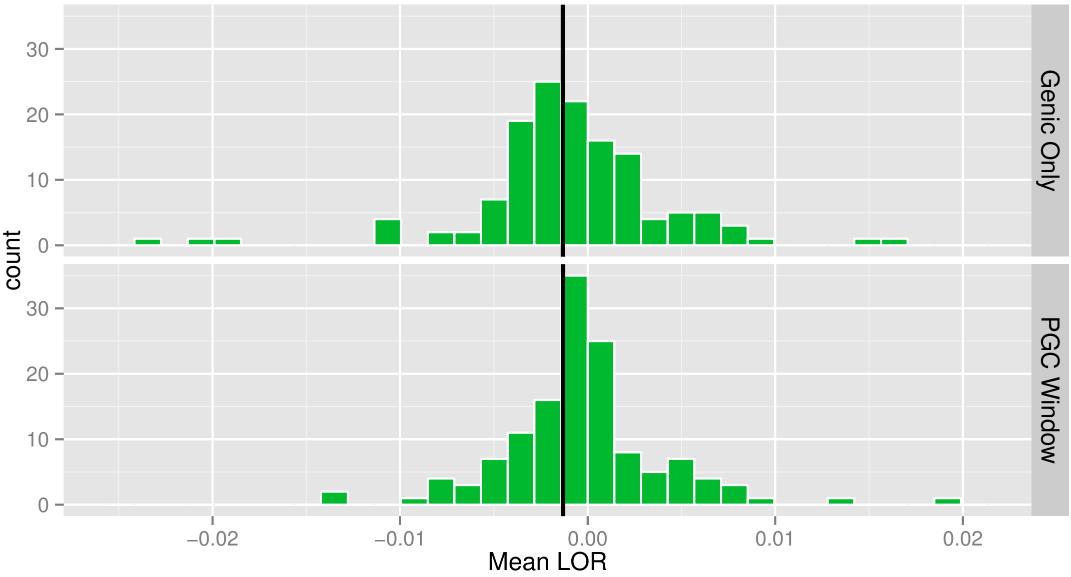


Figure 5.8.: Histogram showing the distribution of mean LOR across all the gene sets. This is the mean of all the values entered into the models for each set.

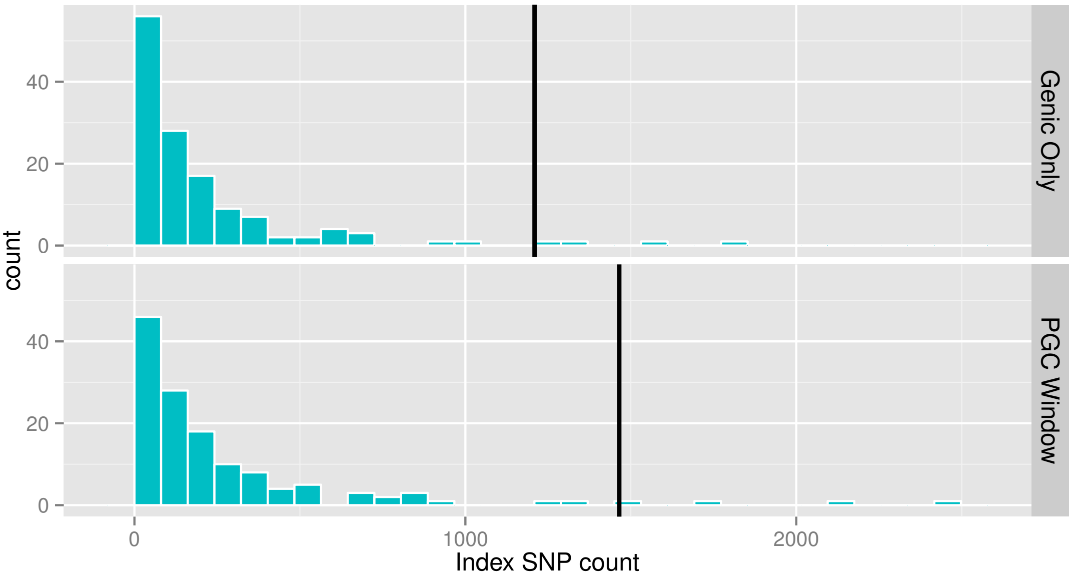


Figure 5.9.: Histogram showing the distribution of total number of index SNPs across all the gene sets.

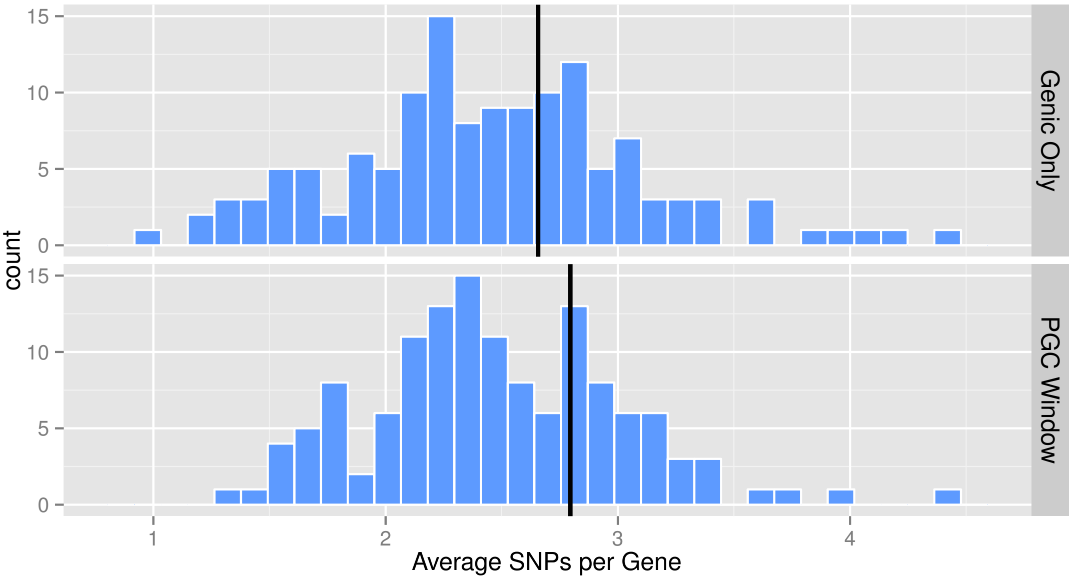


Figure 5.10.: Histogram showing the distribution of the average number of index SNPs per gene across all the gene sets.

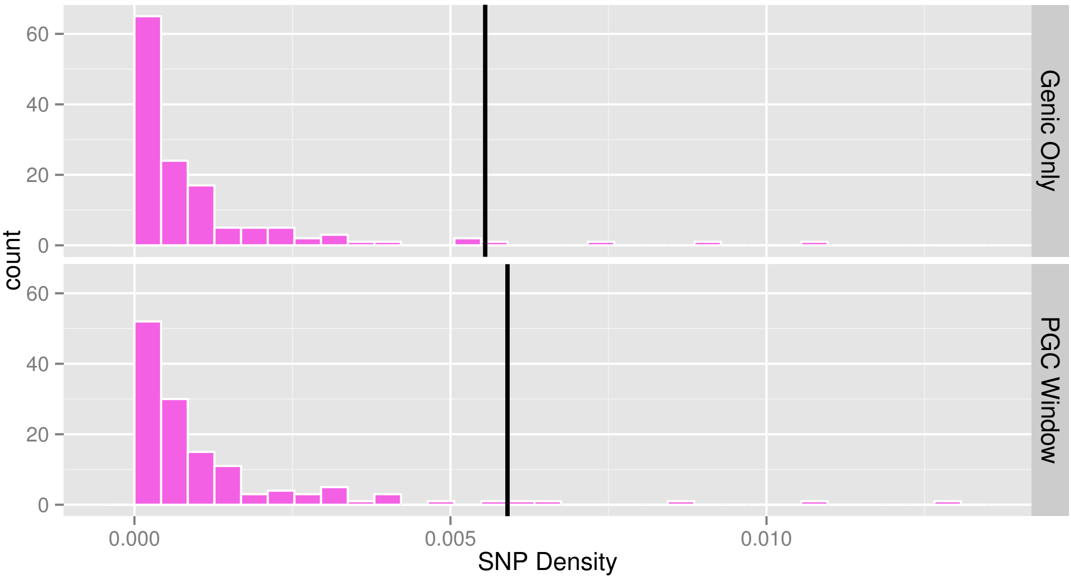


Figure 5.11.: Histogram showing the distribution of the average density of index SNPs in the genes. This was calculated by dividing the number of index SNPs by the average gene size for each gene set.

These figures provide information about what could be driving the high performance of the FMRP gene set. In three of the six summaries, the FMRP set does not stand out at all, sitting in the centre of the distributions. These three were the gene size, the mean LOR of the SNPs and the average number of index SNPs seen per gene in the set. It was very reassuring to see that the FMRP was not a high outlier for the gene size. This lends support to the notion that there is signal from this set, and it is not just an artifact of larger genes casting more of a net over the genome regions and allays some of the concerns raised by Ouwenga and Dougherty (2015). For the other three summaries however, this set really does stand out as a high performer. Interestingly, the shapes of these three summaries more resemble a geometric distribution, instead of a normal distribution. The tail of these distributions show that there are quite a few high performers; but interestingly there are some that show higher values than FMRP.

What is most likely the main cause for the high values, and this shape of distribution is that there is clearly a non-uniform distribution of index SNPs, whose p-values from the GWAS are below the threshold of 0.05. This means that there is a selection bias behind the unequal number of index SNPs seen in different gene sets. This in turn explains the greater number of relevant genes, as well as the increased SNP density.

A summary of the six gene sets with the high values seen in the tails of the three distributions can be seen in table 5.8. What is immediately apparent here is how large a contribution all of these particular gene sets are making to the total number of index SNPs used. As a recap to table 5.2, the genic data had 3,601 genes and the dataset with the flanking UTRs had 4,629. the FMRP targets account for about a third of these in each case. However, while this provides an explanation as to why this set was treated so importantly by the models, it is very interesting to see that other larger sets were not; even the abnormal behaviour set, which actually represents everything in the behaviour gene set class. In order to investigate this further, a permutation analysis was carried out.

Table 5.8.: Summary of the five gene sets with the highest values in the three summaries where FMRP scored highly.

Border Region	Gene set	Gene Number	Index SNP count	SNP Density
Genic Only	Abnormal Behaviour	817	1822	0.010553815
	Abnormal Nervous System Morphology	705	1595	0.008997016
	Abnormal Motor Capabilities/Coordination /Movement	561	1310	0.007190612
	FMRP Targets	455	1209	0.005549435
	Abnormal Brain Morphology	434	993	0.005331572
	Abnormal Neuron Morphology	386	958	0.005077326
PGC Window	Abnormal Behaviour	1089	2424	0.012676432
	Abnormal Nervous System Morphology	949	2097	0.010860446
	Abnormal Motor Capabilities/Coordination /Movement	729	1697	0.008471615
	Abnormal Brain Morphology	584	1328	0.008471615
	FMRP Targets	524	1465	0.008471615
	Abnormal Neuron Morphology	522	1244	0.006191827

5.3.3. Permutation analysis to assess the significance of the SNPs in the FMRP set

This section describes the analysis that was carried out to examine if the relative importance put onto the FMRP targets set is due to the high proportion of the total SNPs being contained within this set, or if there is something about those particular SNPs within its boundaries that, when taken as a collective group, is delivering a clear signal that is being picked up by the models. For each dataset (genic only and flanking UTRs), the polygenic score for all samples in the FMRP targets was recorded. Then a further 249 polygenic scores were made for each sample, but by using random selections of all the available SNPs, equalling the total number in the FMRP set. This created a total of 250 different polygenic scores, made up using equal numbers of SNPs. These were then tested with a logistic regression with 250 stratified splits of the samples with the same train/test splits of 75%/25%. From these trials for each score, the median value was recorded. The median was used instead of the mean, just in case there were certain splits of the data which resulted in erroneously high or low scores. The metric was, as before, the Receiver Operating Characteristic curve.

Histograms showing the distributions of these scores are shown in figure 5.12. The black vertical lines on each histogram show the level of performance of the original FMRP target set score.

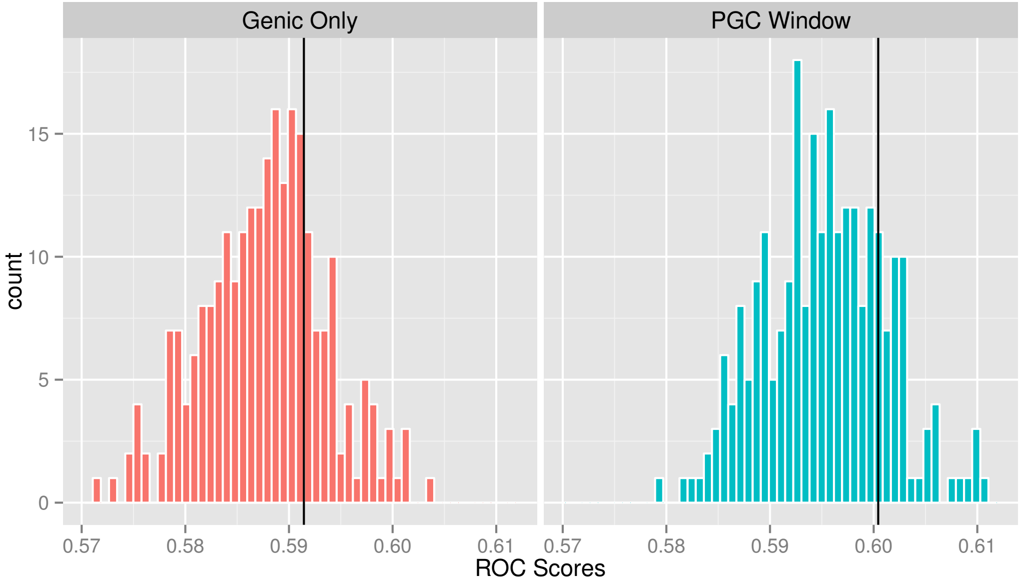


Figure 5.12.: Histograms of the distributions of median ROC scores for both datasets. The black vertical line shows the performance of the FMRP target specific SNPs.

These histograms show that, while the general performance of the FMRP target specific polygenic score definitely seemed to be better than most of the SNP selections, it was far from showing empirical statistical significance. This suggests that the sheer size of the number of SNPs present in this gene set could account for much of why this set is considered as being important.

The FMRP target set contained roughly a third of all the available SNPs, so it is very unlikely that there would be any significant results in these permutations as there must be so much cross-over of the SNP selections for each permutation. While there is no statistically significant result here, the fact that this set outperformed other sets, which are shown in table 5.8 to actually carry *more* SNPs and signal still suggests a particular role for this set. It also must be remembered that the very reason why these index SNPs were chosen was because they specifically showed association with schizophrenia in the PGC-2 GWAS.

5.3.4. Testing for gene set association and significance using the MAGMA software

At this point, after all of the analysis concerning the linear kernels has been described and the FMRP set has been highlighted as playing a prominent role, the results from the MAGMA gene set annotation software are described (de Leeuw et al., 2015)[1]. The method used here was the one described in the manual that makes use of the p-value significance scores from the GWAS results. The instructions state that this requires the use of an additional reference panel dataset, which the authors suggest should be the results from the 1000 genomes project (1000 Genomes Project Consortium, 2012), and this is made available for download on the software website. This is used to correct for LD as the SNP inputs contain information on all of the SNPs, not just the index SNPs. Because of this reason, the number of genes in each gene set will differ from those reported in the analysis so far. These gene numbers are provided in the results tables. The stages required to perform this analysis are described here in turn:

The first stage was to perform a gene annotation procedure, which provides MAGMA with a file containing the locations of all of the genes used in the analysis together with the SNPs contained within each gene. This procedure requires the following sets of information:

- The locations of the SNPs - the .bim file from the PLINK analysis
- Information about the locations and DNA strand directions of the genes: a file with the following information: gene number, chromosome, start position, end position, direction, and gene name (optional)
- Any information about any desired window for the flanking UTR regions if desired

This was used to create two different annotations: the genic regions only, and another with flanking UTR regions with the same 35/10 KB upstream/downstream sizes.

The next stage creates association scores for the genes, and is a necessary step that has to be completed before these gene set associations can be calculated. The inputs required for this procedure are:

- The genome reference information - taken from the 1000 genomes .bed, .bim, .fam files.

[1] The full instruction manual for the different analytical procedures can be seen at `http://ctg.cncr.nl/software/MAGMA/doc/manual_v1.05a.pdf`

- The p-value information - a file of the SNP names and their respective p-values (taken from the GWAS results). This requires some additional information, described below.

- The gene annotation information, which is the output from the previous procedure.

In addition to the p-value information, the software requires that the number of samples used to create these p-values is provided. This number was calculated as the number of samples used to make up the total number of samples with European ancestry used in the PGC-2 study, minus the number of samples in the CLOZUK study and the samples from another study carried out in Cardiff University called the "Cardiff COGS" study (carried out by JTR. Walters) - as this represented the number of samples that were used to perform the GWAS that created all of the LOR scores used in this analysis. This number was calculated from the information in table 1 in the supplementary material of the paper, on pages 21 and 22 (Ripke et al, 2014) by summing the columns for cases and controls for "Total EUR CC" and subtracting off the case and control counts for the two different chips used in the CLOZUK study, and those from the Cardiff COGS samples. This final number came to 61,817 samples.

After the second stage is complete, there is now an output with the association information on the genes, which is then used to perform the gene set association analysis. This final stage of the process requires the following:

- The gene association output from the previous stage

- A file with the gene set information - A text file with two columns: the name of the gene set and a list of all of the gene numbers in that set.

- A number specifying the number of random gene permutations to perform to gain a "corrected p-value" - more information below.

The third point mentions the creation of a corrected p-value. This is done because MAGMA does not correct for the multiple comparison testing of the gene set associations. The information on this procedure can be seen in full in the MAGMA manual under the heading *Empirical multiple testing correction*, but is summarised here. As no correction is made, there are two possible solutions: apply a Bonferroni correction to all of the p-values or carry out permutations to gain an empirically corrected p-value. The MAGMA authors describe the first option as too conservative, so advise the use of the second. This requires the following command to be included at this stage of the analysis: `--model fwer=N` with `N` as the number of permutations to perform. The value of `N` was set at 100,000 for the analysis. This gives a new column in the output with the heading `COMP_P_CORR` and is the one reported in table 5.9 shown below.

Table 5.9.: The results of the MAGMA analysis for the top three associated gene sets found for the genic only regions, and those with the flanking windows - labelled "PGC Window".

Border Region	Gene set	Gene Number	Corrected p-value
Genic Only	FMRP Targets	777	0.00034
	Abnormal Grooming Behaviour	63	0.077
	PSD (Human Core)	627	0.145
PGC Window	FMRP Targets	807	< 0.0001
	PSD (Human Core)	651	0.03
	5HT-2C	16	0.0627

The results of the MAGMA process show that the FMRP gene set is reported as the most significant set for both the genic regions, and those with the flanking UTRs. The only other set to show any permutation-corrected significance at the α level of 0.05 is the Post Synaptic Density (PSD) when the UTRs are included. This set pertains to genes that also have a role in synaptic processes and development (Meyer et al., 2014). More of the interpretations of these findings can be read in the discussion of this chapter.

5.3.5. Exploring possible interactions with non-linear kernels

While this work so far has shown that the linear SVM algorithm has been able to rank the importance of different gene sets, the full power of the machine learning methods would not be utilised if the same analysis was not done with the non-linear kernels in order to identify any possible interactions that could be taking place.

In order to examine this, the exact same procedure was carried out as per the linear kernels (described in section 5.2.3), but with the same non-linear kernels that were used in chapter 3, namely the polynomial (with degrees 2, 3, and 4) and the RBF kernels. The same CV routines were carried out, the only difference being that a series of values for γ had to be included into the grid search of the hyperparameters: $\{0.001, 0.01, 0.1\}$. As the non-linear algorithms cannot use LibLinear optimisation, LibSVM was used instead, the same optimisation technique used in chapter 3. As the performance of the non-linear techniques is being compared with the performance of the linear kernel, the linear procedure was also re-run using LibSVM in order to make easier comparisons. The results can be seen in figure 5.13.

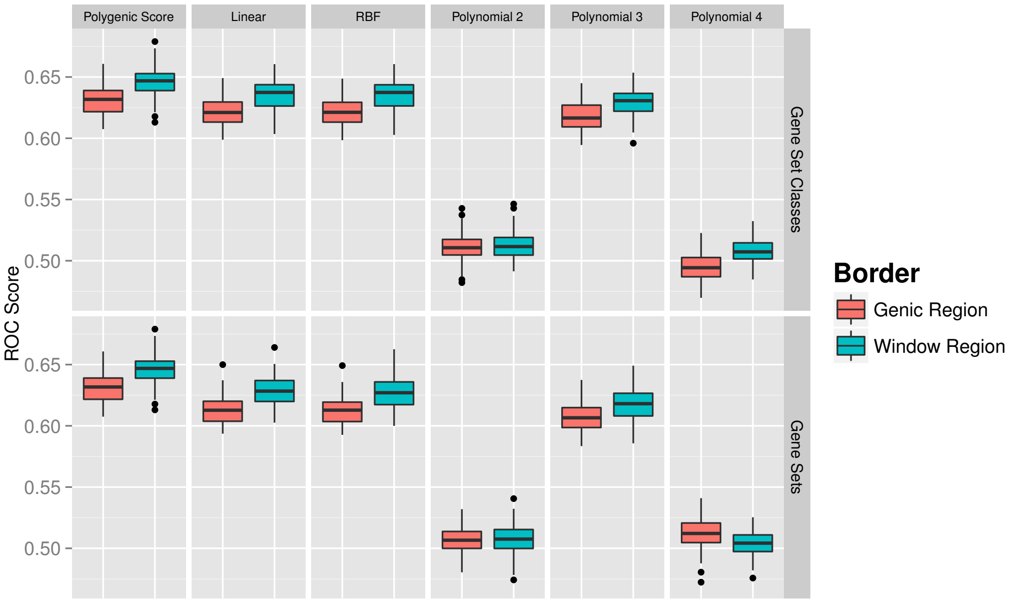


Figure 5.13.: Box plot showing the distribution of scores for both the gene sets and gene set classes for the linear and non-linear algorithms. The different colours represent the different boundary regions used. Note that the polygenic score is the same for the gene sets and the gene set classes.

Using the additional FMRP gene scores as inputs

As the results so far had identified a prevalent role for the FMRP gene set, this was an opportunity to examine if there was any evidence of interactions between the individual genes contained within this gene set. The procedure to create gene-based scores was carried out in exactly the same manner that the gene set classes and gene sets were created (section 5.2.1), where any duplicate SNPs within genes were removed, and the score calculated by taking the mean of the LOR weighted reference allele counts per gene using the database style `groupby` operation in Pandas. This score can be thought of as a "mini polygenic-score-per-gene". Referring to table 5.8, this provided information on 455/524 genes containing 1209/1465 SNPs for the genic and UTR flanking regions respectively. These scores were then tested using exactly the same procedure and the results can be seen in figure 5.14.

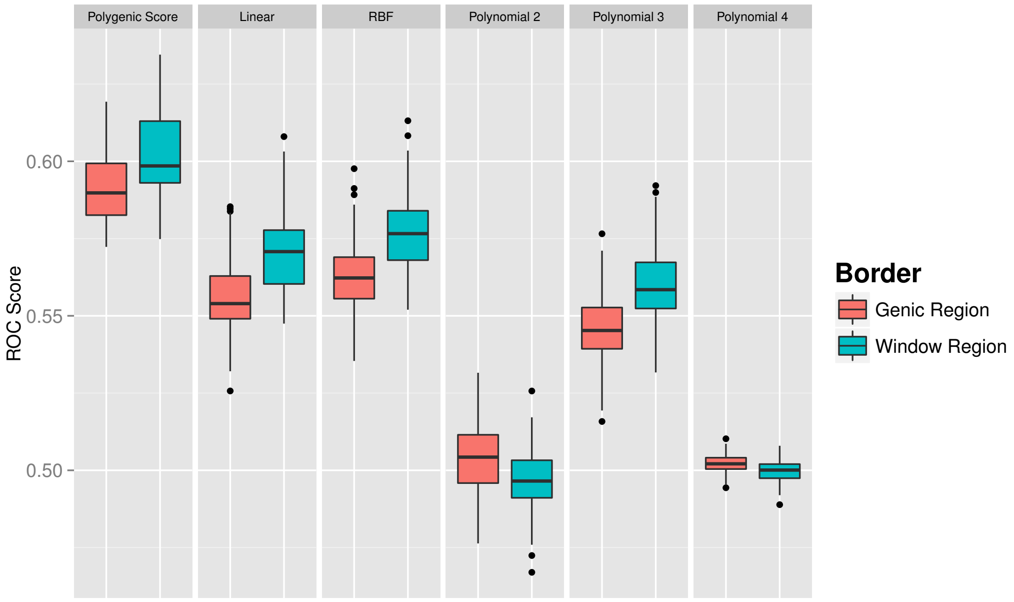


Figure 5.14.: Box plot showing the distribution of scores for the FMRP gene scores for linear and non-linear kernels. The different colours represent the different boundary regions used.

Summary of non-linear results

The spread of the results for the gene sets and classes in figure 5.13 resembles those seen for the SNP studies in chapter 3, with the characteristic drop in performance for the even-valued degrees of the polynomial kernels. In these results, the median values of the linear and RBF kernels are so close that it is is not possible to see any difference. Because of this, the values for the polygenic score, the linear kernel and the RBF kernel are shown in table 5.10. The information for the polynomial kernels is not included as it is clear that these performed at a much lower level. The conclusion from these results is that it does not appear to show any strong evidence of the presence of interactions between either the gene sets or the gene set classes. For the gene set classes, the RBF kernel is performing slightly better than the linear kernel, but only by an incredibly modest amount, and for the gene sets, the results of the two kernels are either the same, or in the case of the UTR region, show a superior linear kernel performance. If the results of the simulation procedures from chapter 4 are trusted completely then there *could* be some interactions occurring, as the results of these trials showed that if there was no contribution from the interactions, then the RBF kernel always had inferior performance, but no firm conclusions about this can be made.

Table 5.10.: Table showing the median result for the performance of the polygenic score, linear kernel and RBF kernel on both the gene set classes and the gene sets for the two different border regions.

Input Type	Border Region	Algorithm	Median
Gene set classes	Genic Region	Polygenic Score	0.6317
		Linear Kernel	0.6210
		RBF Kernel	0.6212
	Window Region	Polygenic Score	0.6469
		Linear Kernel	0.6374
		RBF Kernel	0.6374
Gene sets	Genic Region	Polygenic Score	0.6317
		Linear Kernel	0.6127
		RBF Kernel	0.6128
	Window Region	Polygenic Score	0.6469
		Linear Kernel	0.6283
		RBF Kernel	0.6270

The situation is different for the trials on the FMRP genes. There is much more evidence for the presence of interactions as the RBF kernel is clearly outperforming the linear kernel for both the genic-region and window-region datasets. The results from pairwise[1] t-test comparisons showed that these differences were highly statistically significant, the results are summarised in table 5.11.

Table 5.11.: Results of pairwise t-tests comparing the performance of the linear and RBF kernels on the FMRP gene scores for the genic-region and window-region datasets.

Border Region	t-value	p-value
Genic Region	19.02	$< 2.2 \times 10^{-16}$
Window Region	16.271	$< 2.2 \times 10^{-16}$

5.3.6. Including the OMNI chip samples in the gene set and gene analyses

As the results from both the real and simulated data showed that the all of the SVM kernels benefited from the additional samples of the OMNI chip, the extra samples were now incorporated into the analyses and different input features described in this chapter already. In chapter 3, checks were carried out to ascertain if the models developed on the Illumina 1M (I1M) chip were compatible with the OMNI chip data for the SNP input

[1] A pairwise test could be used due to the use of a random seed for the CV procedure which ensured that the train/test splits were the same for all algorithms.

features, and as this was shown to be the case, it was not deemed necessary to repeat the process in this chapter.

The results provided in this section make use of exactly the same procedure as was described in the previous section using the non-linear kernels in addition to the polygenic score and the linear kernel using the LibSVM optimisation method. As the intention in this experiment was to see how the extra samples affected the patterns of performance across the linear and non-linear methods, the procedure using the LibLinear optimisation technique was not repeated here. Also, to make the result figures easier to interpret, only the results from the window region dataset are shown here.

The distributions of these results for the gene set classes, gene sets and FMRP genes can be seen in figures 5.15 and 5.16. A different colour scheme is used here to emphasise that the comparisons made are now between using the I1M samples only and the dataset with the samples from both the I1M and OMNI chips.

The results for the gene set classes and gene sets show only very marginal changes in performance. The performance for the gene set classes appears slightly decreased for the additional samples, and there is only a modest increase in performance for the gene sets. However, the results for the FMRP gene scores show a very different pattern: the performance for the linear, RBF and polynomial-3 kernel displays a clear increase with the additional samples. The opposite is seen for the even-degree polynomial kernels, but their performance was always very poor in the real datasets. It is also interesting to see that the polygenic score has not benefited at all from these extra samples either. The pattern across the algorithms is very similar to that seen with the threshold dataset with 4,998 SNPs in chapter 3. The differences between the RBF and linear kernels with the extra samples was still significant when assessed with a pairwise t-test ($t = 16.023, p < 2.2 \times 10^{-16}$).

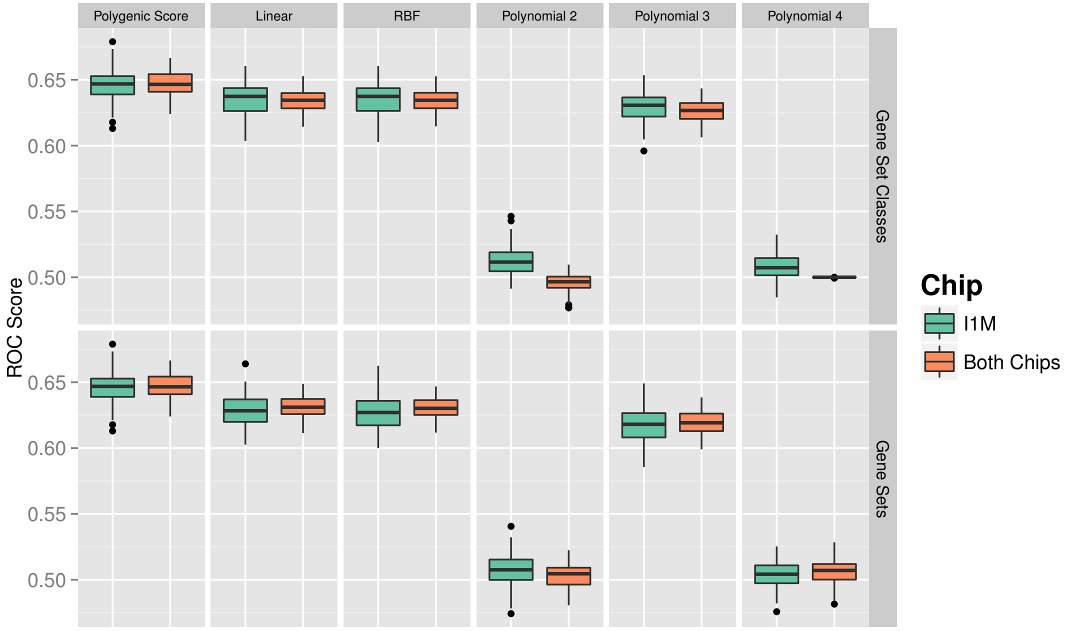


Figure 5.15.: The different distributions of ROC scores for the gene set classes and gene sets, for the two different datasets: I1M chip only and both chips combined.

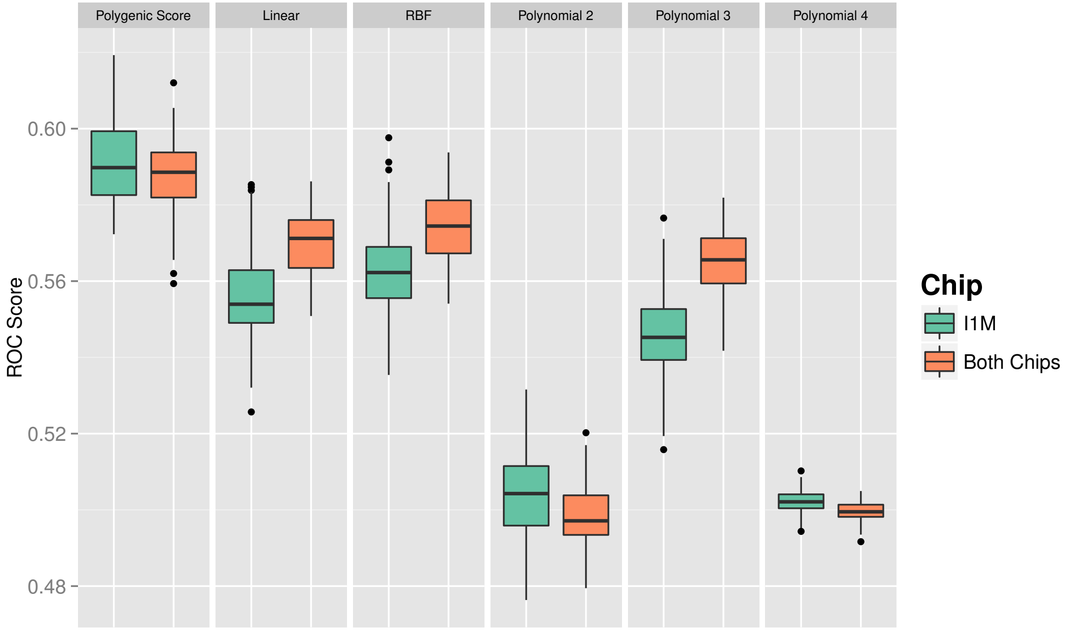


Figure 5.16.: The different distributions of ROC scores for the FMRP gene scores, for the two different datasets: I1M chip only and both chips combined.

5.3.7. Non-linear kernel analysis of the Additional Datasets with annotation information

The final pieces of analyses carried out in this chapter concerned the two extra datasets that were described in section 5.2.2. As a quick recap, the one dataset was made by providing the additional weighting to the LOR weighted reference SNP counts of the square root of the number of SNPs that fall within the LD region of the different index SNPs; and the other dataset was made by weighted the LOR weighted reference SNP counts by the probability of the protein coding region where they are located being Loss of Function (LoF) intolerant.

These two analyses were done to try and bring some extra information to the models beyond what is normally included in the study of psychiatric genetics studies, namely: information gathered from the LOR outcome of GWAS studies. The trials with the LoF intolerant information were of particular interest in light of the recent work that has been carried out in this area (Lek et al., 2015). As this information also only concerns the SNPs that lie in protein coding regions, it was deemed an ideal addition to the work carried out in this chapter, as focussing on the gene set regions meant that the SNPs in the non-coding regions were not featured in these analyses anyway.

It was difficult to state a clear hypothesis for the outcome of these experiments as there has been no information found about these sources of annotation being used in a machine learning experiment on genotyped data before. In order to see how the models would respond to this extra information, both of the datasets were tested using the procedure carried out for all of the methods that included the non-linear kernels into the analysis. This was considered to be the best approach to see if any differences in the patterns of performance would be seen. These first trials only included the information from the I1M chip samples.

The box plots of the results from these two datasets can be seen in figure 5.17 and figure 5.18.

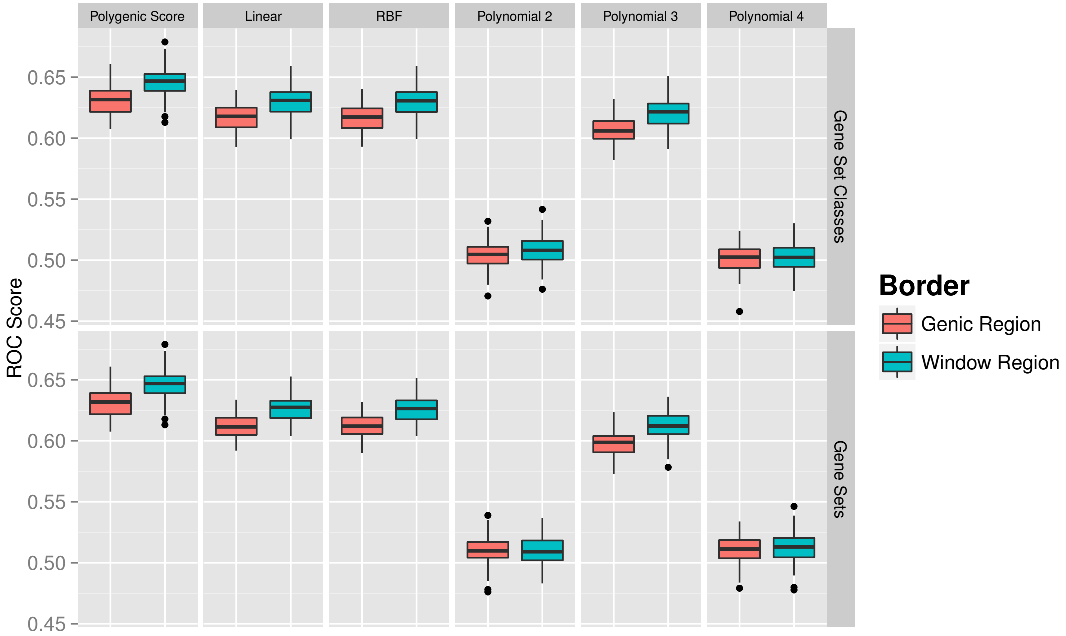


Figure 5.17.: The distributions of ROC scores for the different algorithms on the gene set and gene set classes datasets that were made with the additional weighting of the count of the SNPs lying in the LD region of the index SNPs.

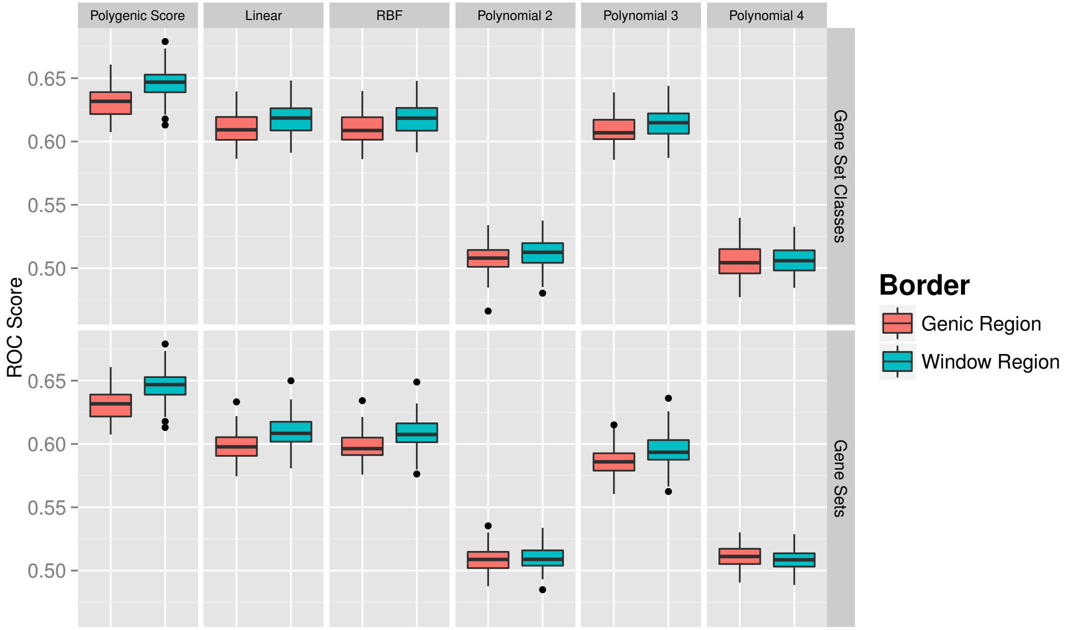


Figure 5.18.: The distributions of ROC scores for the different algorithms on the gene set and gene set classes datasets that had the additional information of the LoF intolerance probabilities of the various protein coding regions.

As can be seen in both of these figures, there is a modest increase in performance for both of the datasets with the additional information from the extra samples of the OMNI chip. However, none of the results of the SVM algorithm on these shows any improvement over the original performance of the polygenic score. The results of pairwise t-tests on the gene set input values between the new datasets, and the originals for the gene-set scores for the linear and RBF kernels are shown in the following tables. The negative t-values show that these new datasets are not performing as well as the original. Table 5.12 shows that weighting by the SNP count in the LD region makes little difference, with only the linear kernel for the UTR region showing a significant drop in performance. The LoF dataset, in table 5.13, shows a highly significant drop in performance from the original scores.

Table 5.12.: Results of pairwise t-tests between the LD region weighted gene sets, and the original gene set scores for the linear and RBF kernel.

Algorithm	Border Region	t-value(99)	p-value
Linear Kernel	Genes Only	-0.82	0.41
	UTR Window	-2.99	0.004
RBF Kernel	Genes Only	-0.06	0.95
	UTR Window	-1.28	0.2

Table 5.13.: Results of pairwise t-tests between the LoF weighted gene sets, and the original gene set scores for the linear and RBF kernel.

Algorithm	Border Region	t-value(99)	p-value
Linear Kernel	Genes Only	-21.7	2.2×10^{-39}
	UTR Window	-28.6	1.2×10^{-49}
RBF Kernel	Genes Only	-20.9	4.3×10^{-38}
	UTR Window	-25.9	7.5×10^{-46}

At this point, it is clear that neither of these methods of providing the models with additional annotation information about the SNPs and genes is effective at increasing predictive performance.

5.4. Discussion

The material and trials described in this chapter were carried out to establish if the SVM algorithm could be used to identify any contribution from signals in the genome, which had been summarised to the levels of genes, gene sets and gene set classes. The information from the individual SNPs was prepared in this way to examine if it would have an effect on the outcome of the models. This was examined by building a range of linear kernel models and observing the changes in the coefficients that were assigned to the input features, and using these values as an assessment of the importance of their contributions.

Using this method, some of the gene sets were identified as making higher contributions to the models as the distributions of their assigned coefficients were consistently higher than the distributions of the other gene sets. This was particularly relevant for those SNPs which fell in regions of gene that coded for proteins that were targets of the Fragile-X Mental Retardation Protein (FMRP). This gene set showed a higher level of contribution in the situations when only the protein coding regions were considered, and when the flanking windows into the UTRs were included as well. The identification of the FMRP set as playing a prominent role was replicated with the analysis using the MAGMA gene set association software. For both of the input types, this set showed the most significant association with treatment-resistant schizophrenia and survived the permutation correction procedure for multiple comparison testing on both occasions. The high association levels of the FMRP set is also mentioned in the extended material of the PGC-2 study (Extended table 1) (Schizophrenia Working Group of the Psychiatric Genomics Consortium et al., 2014). When the flanking UTRs were included, MAGMA also identified a (corrected) significant role of the Post Synaptic Density (PSD) set. The genes that were selected by Pocklington et al. (2015) to be included in this set were taken from a proteomic study into the diseases and evolution of the PSD by Bayés et al. (2011).

The area of the genome encoding for the Fragile-X Mental Retardation Protein is the *FMR1* gene located on the X-chromosome (Verhelj et al., 1993; Weiler and Greenough, 1999). An abnormally long repeat of the CGG trinucleotide sequence in the $5'$ UTR of the gene, in the promoter region, results in hyper-methylation that can silence the expression of the gene entirely, and result in no FMRP product being translated at all. It is interesting to note that the abnormalities related to this gene are located in a flanking UTR and shows the justification for considering these regions in the analysis. The complete loss of this protein results in an individual developing fragile X-syndrome, a disorder that is characterised by mild physical abnormalities and cognitive impairments including a long face, large ears, a tendency to hand flap, intellectual disabilities, learning impairments, eye-contact avoidance and social anxiety (Bear et al., 2004; Hagerman et al., 2009; Rogers et al., 2001). It takes the name of “fragile-X” as it was observed by Lubs in 1969 that the X-chromosome in cells cultured from sufferers had a tendency to be fragile when replication occurred in folate deficient medium (Weiler and Greenough, 1999).

For many years now, there has been a great deal of interest in FMRP as it has been shown to have implications in subjects who do not have the 5′ FMR1 mutation. There is now evidence that there are signs of reduced expression of FMR1 in individuals with autism, schizophrenia, bipolar disorder and major depression (Folsom et al., 2015), and as was mentioned earlier, FMRP is involved in the transportation of select mRNA from the cell nucleus to the synapse (Antar and Bassell, 2003; Huber et al., 2002; Feng et al., 1997; Fromer et al., 2014). Purcell et al. (2014) carried out an analysis of small rare mutations (seen in less than 1 in 10,000 individuals) using exome sequencing of gene regions that have been previously been found to show association with schizophrenia from CNVs studies (Kirov et al., 2012; Sullivan et al., 2012), GWAS reports (Ripke et al, 2014; PGC-Cross Disorder Group, 2013) and investigations into *de novo* mutations using exome sequencing (Fromer et al., 2014; Girard et al., 2011; Xu et al., 2012). These rare point mutations are referred to as Single Nucleotide Variations (SNVs), a term used to differentiate them from the acronym SNPs, which represents more common mutations. As these SNVs are rarer, exome sequencing techniques have to be used instead of genotyping techniques. The results of these analyses showed that the targets of FMRP represented one of the gene sets that was enriched for SNVs associated with schizophrenia. The authors point out that none of the individual mutations, or the genes in these pathways showed association that survived multiple comparison testing. This demonstrates the power of collecting any mutations and variants together into these functionally related groups like gene sets as was done in this chapter. The other gene sets found to show association were those involved in voltage-gated calcium ion channels and the Activity-Regulated Cytoskeleton-associated (ARC) protein of the Post Synaptic Density. All of these are associated with synaptic function and plasticity, and in a review of these processes by Hall et al. (2015), it was pointed out that this is an extremely large topic, and a brief review of how the findings of this chapter are related to these developments in the literature is reported in the main discussion in chapter 6.

After the linear SVM models had identified the FMRP set as playing a prominent role, further investigations were carried out to establish what could be driving the signal contained in this collection of SNPs. Ouwenga and Dougherty (2015) in particular had pointed out that the genes that are expressed in brain regions have a tendency to be very large, so any enrichment for disease associated SNPs could be due to the mere fact of these protein coding regions covering proportionally larger regions of the genome than other genes. A number of exploratory data summary procedures were done to assess the patterns seen within the genes in the FMRP set, and how these compared with all of the other gene sets: the mean gene size, the number of genes, the mean Log Odds Ratio of all of the index SNPs in the set, the total number of index SNPs, the mean number of index SNPs per gene in the set, and finally, the density of the SNPs in the set that were index SNPs. This analysis showed that the signal was not explained by the size of the genes as the FMRP set fell in the centre of this particular distribution (figure 5.6). This set did, however, prove to be an outlier for three of these features: the number of index SNPs, the number of genes, and the density of index SNPs. It is highly likely that the explanation for all of these findings comes from the fact that many SNPs associated with schizophrenia are located in genes that make up the FMRP set.

As the results from the MAGMA trials also identified the FMRP set as being significantly associated with treatment-resistant schizophrenia, the conclusion is made here that if the aim of any research procedure is to identify associated signals that are essentially main effects, it is far preferable to use alternative methods like MAGMA over machine learning approaches. The amount of data pre-processing involved is far less, and the overall time to complete the analysis is drastically reduced. The developers of software like MAGMA and PLINK have successfully optimised their code to carry out this type of analysis in a highly efficient and effective manner. However, the aim of the work carried out in this thesis, and the justification for using the more complex methods involved in machine learning, and the SVM in particular, was to look for any interactions in the genetic data, which could be delivering a signal larger than simply the linear sum of their individual contributions. In light of the findings from the simulation experiments carried out in chapter 4, further analysis of the genes present in the FMRP set was carried out. A series of trials were performed using the non-linear kernels of the SVMs to assess if there would be any suggestion of a role of interactions between these genes. The distributions of ROC scores for some of these kernels, particularly the Radial-Basis Function (RBF) kernel, showed superior performance over the linear kernel, and this effect was increased with the additional information of the samples from the OMNI chip. Following on from the results seen in the simulation procedures of the previous chapter, this suggests that interactions are indeed occurring between the genes in this set; a finding that warrants further statistical and experimental investigation.

This chapter has described the final experimental procedures that were performed for this thesis. The findings showed that the SVM algorithm is capable of identifying key features that are driving the signal in the results that are also replicated when using other, more traditional, statistical techniques for finding association. The use of the non-linear kernel showed that, once again, the machine learning techniques are capable of finding patterns that suggest a role of complex interactions between the genetic signals and features. These findings would not be possible without the use of these algorithms, as the traditional statistical approaches require the identification of explicitly defined interactions within the data, and they are simply too numerous and would result in an increase in dimensionality, which would render any required computational time as unfeasible. The implications of all of the findings from these three experimental chapters is summarised in the main discussion in the following chapter.

6. Discussion

6.1. Summary of the background and basis for the investigations carried out in this thesis

The main aim of the work carried out for this thesis was to examine if machine learning methods could be used to contribute to the understanding of the genetic mechanisms and aetiologies and explain more of the variation seen in complex psychiatric disorders like schizophrenia. Initially it was hoped that machine learning might have been able to improve on the predictive powers of traditional methods that already exist in the field, namely the Logistic Regression (LR) of the polygenic score, made from collections of genotyped Single Nucleotide Polymorphisms (SNPs) and the results from a Genome-Wide Association Study (GWAS). Any possible contribution to this field would be of valuable use to the study of schizophrenia to try and help alleviate the patients suffering and the enormous costs that it poses to the economy and society (Owen et al., 2016; Mangalore et al., 2007).

The first part of the introduction in chapter 1 explained how the debates and discussions about the diagnostic issues pertaining to schizophrenia and other psychiatric disorders has developed over time. Descriptions of how the beliefs about the nature of these disorders has moved from considering them as discrete categories towards treating them more as having positions on different spectra of symptoms and severity were provided (Crow, 1986; Craddock and Owen, 2005, 2010; Hyman, 2010).

The second half of the introduction focussed on the different methods and techniques that have been developed over the past decades to study a whole range of genetic disorders, how movements have been made throughout international co-operation to learn more about the architecture of the genome such as the Human Genome Project, the International HapMap consortium and 1,000 Genomes Project, and how different research groups from around the world have collaborated to create large consortia studies to increase sample sizes and statistical powers of the experiments. This section also highlighted how GWAS techniques showed better performance than linkage studies due to their abilities to look for variants of small effect from large sample sizes without having to rely on finding related subjects.

It was due to the concerns about the heterogenous nature of schizophrenia, and how this could result from different collections of genetic anticedents, that the experiments in this thesis were performed on a genotyped dataset from sufferers of treatment-resistant schizophrenia who were receiving the Clozapine medication from the pharmaceutical company Novartis. This dataset was called the CLOZUK dataset and was taken from an earlier study carried out at Cardiff University (Hamshere et al., 2013). This dataset also featured in the large scale study by the Psychiatric Genetics Consortium (PGC) that identified 128 SNPs on the autosomes (125) and sex-chromosomes (3), representing 108 independent loci, that were found to be associated with schizophrenia at the GWAS significance level of 5×10^{-8}. The basis for choosing this dataset was that it might hopefully represent a more homogenous phenotype, with a more similar set of genetic aetiologies.

Searching for evidence of interactions between variants

The material presented in chapter 2 gave an introduction to the machine learning algorithms that have been used in a range of fields in the life sciences, with particular attention to the Support Vector Machine (SVM) due to its ability to build either linear models for increased interpretability or kernel-based methods that can increase predictive power. As has been stressed throughout this thesis, the main reason for using SVM methods is to examine the possible contribution of *interactions* between variants, and to see if they explain any of the missing heritability that is so often mentioned in the field (Hemani et al., 2014; Polderman et al., 2015). Searching for interactions explicitly is extremely computationally expensive and can result in the *Curse of Dimensionality* problem (Bishop, 2006; Hastie et al., 2001) due to the sheer numbers of input variables to models that can come from looking at all the combinations of an already large number of variants.

Despite these practical difficulties, previous findings have shown that this is still a worthwhile endeavour. Using algorithms that carry out large scale parallel processing of data on Graphical Processing Units (GPUs), Hemani et al. (2014) found evidence of a number of pairwise SNP interactions that were significantly associated with observed levels of gene expression. However, the main difference here is that the effect sizes of the individual SNPs alone on the expression levels are already far greater than those seen in complex psychiatric disorders like schizophrenia (Kavanagh et al., 2015).

The increase in complexity that can occur when performing a similar comprehensive search for interactions in polygenic disorders, especially those of higher orders than two can soon become unfeasible (Cordell, 2009). The hope when starting out this thesis was that using machine learning would result in observable patterns that could be used to infer the evidence of interactions occurring within the data. Through a process of carefully building and tuning the models, with linear and non-linear kernels on real and simulated data, the conclusion is made here that this process has been a success. Key differential patterns of behaviour, especially between the linear and Radial-Basis Function (RBF) kernels, when looking for signals in the simulations performed in chapter 4 driven by either main effects or interactions showed a very similar pattern to the distributions of results seen in the real datasets.

6.2. Summary of key results from experiments

Patterns seen in the SNP and gene set datasets

This section will outline some of the key patterns of findings that were observed in the results on the real datasets of the SNPs and gene sets that were carried out in chapters 3 and 5. In all of the trials performed, none of the machine learning algorithms (both the SVM and the multiple logistic regression models) showed any improvement over that of the polygenic score analysis. In chapter 3 it was shown that this pattern also remained when the *unweighted* polygenic score was used.

For the 125 GWAS significant SNPs, the gene sets and the gene set classes, the next best performances were seen from the linear kernel SVMs. These outperformed the results from any of the non-linear kernels. For these particular datasets, there was a general pattern seen of a slight drop in performance for the RBF kernel, and a further small drop for the polynomial kernels with degree 3. In all occasions, for all of the datasets, the even-valued degrees of the polynomial kernels showed extremely poor performance, very often only slightly above the chance Receiver Operating Characteristic (ROC) values of 0.5. This is not surprising, as the evidence suggest that the main bulk of the signal is being delivered by the main effects, and this explains why the linear methods like the polygenic score and the linear kernel are doing well. So any even-valued polynomial will be particularly bad at matching this. However, the main points of interest were observed in the SVM results from the datasets representing the larger group of 4,998 sub-threshold SNPs and the gene based scores for the genes that were the targets of the Fragile-X Mental Retardation Protein (FMRP). There were two particular patterns of interest: firstly the RBF kernel was showing superior performance over the linear kernels and secondly, the SVM models benefited from the additional sample of the Illumina Omni Express (OMNI) chip samples, whereas this was not seen for the polygenic score. The lack of improved performance of the polygenic score suggests that the level seen here is the asymptotic level of its performance, due to the fact that it is looking at the main effects only. There is the possibility that, with greater samples sizes and developments in computational power, the algorithms that take into account the effects of interactions will begin to improve on this level of predictive power.

One of the aims of the work in chapter 3 was to examine if the SVMs could build models from the allele count information only. In hindsight, it may have been better to have worked with the weighted allele counts at all stages, and to have performed the scaling routines on these, as it was the weighted polygenic score that provided the overall best predictive model in this thesis. However, the conclusion is made here that using the weighted allele counts would not have made so much of a difference in the results, or contributed more information to the SVM models, as the performance of the unweighted polygenic score was very similar to that of the weighted score.

The other main conclusion was that the superior performance seen by the RBF kernel for certain datasets suggested the role of interactions between the input features. In order to provide more evidence to support this, the simulation trials in chapter 4 were carried out.

Simulating the phenotypes to test the role of interactions

In order to try and explain the patterns seen in the SNP data, the experiments with the phenotypes that were simulated from combinations of main effects and pairwise interactions from the 125 GWAS SNPs were performed. Initially, models were built using the phenotypes using only the main effects or the interactions exclusively. These showed that the pattern that most resembled those seen in the main datasets was the results from the main effect simulations, with the characteristic drop in performance for the even-degree polynomial kernels and the high performance for the others. One key observation was that the ROC values for the polygenic score dropped considerably when a proportion of the inputs were not contributing to the outcome; the SVMs on the other hand were capable of filtering out this unwanted information. This finding confirmed that the main genetic signals were being driven by the main effects from all of the SNPs, which makes intuitive sense, seeing as the input SNPs were chosen based on their association with the schizophrenia phenotype.

The results from the phenotypes made from only the pairwise interactions showed a completely different pattern, with the linear kernel and polygenic score performing worst, and the RBF and polynomial-2 kernels performing best. The fact that this kernel was performing the best is not surprising seeing as it is designed to look specifically for pairwise interactions. The main conclusions made from these trials was that the signal in the real datasets was clearly not primarily driven by pairwise interactions, and that the RBF kernel showed an impressive ability to adapt to the different types of signal provided to it.

At this point, simulations were performed using differing contributions of main effects and interactions. The results from the polygenic score method and the linear kernel were compared with that of the RBF kernel. The RBF kernel was used specifically instead of the polynomial-2 due to its decent levels of performance for both input types. It could be argued that it would have been preferable to use the polynomial-2, as it looks for pairwise interactions; but its erratic performance between these different inputs led to the conclusion that any results would be too difficult to interpret.

The results of these simulations trials were vital in helping to understand the patterns seen in the real data. The RBF kernel showed superior performance over that of the linear kernel with the larger influence of interactions, especially when these interactions occur between more of the input features. The RBF kernel also showed a tendency to outperform the polygenic score, if enough of the features are interacting and the interactions are large enough in comparison to the main effects. This pattern was seen when looking at the individual pairs of interacting SNPs and when using the more wholistic approach by looking at the different Nagelkerke's R^2 coefficient values given to the main effects and interactions. Another critical similarity was that the RBF kernel seemed to show a better differential performance when the additional samples from the OMNI chip were included, exactly as was seen in the real data.

The robustness of the polygenic score

Perhaps the most surprising result, from both the original data and the simulations was the finding that the polygenic score was showing good performance for both the main effects and the interactions. This was even seen when the Log Odds Ratio (LOR) information was not used, and only the allele count provided in the unweighted version of the polygenic score. On reflection however, this might not be so surprising. It could be the case that the interactions in the real data occur in the sense that a level of risk from two alleles could be slightly greater than the sum of their individual contributions. This can be contrasted with the type of situation seen in the eXclusive-OR (XOR) examples that have been presented on several situations, where the presence of risk alleles would actually completely nullify the effect of the other SNPs, and have the reciprocal effect from those others as well. However, this seems like quite an extreme and unlikely situation to occur, especially as all of the inputs were chosen based on their *main effect* association with schizophrenia in a GWAS analysis.

6.2.1. Results from the Gene Set analyses

There were three main conclusions that were made from the results found in chapter 5:

- The gene set of the FMRP targets seems to be playing a particularly important role. This finding was seen using both the machine learning methods, and the MAGMA gene set association software.
- There was a significant amount of signal provided by the SNPs in the flanking Un-Translated Regions (UTRs).
- The pattern of performance from the linear and RBF kernels for the gene scores in the FMRP set suggests strongly that interactions are occurring between these genes.

The FMRP gene set

Both the machine learning trials, and those using the MAGMA software, identified the targets of the Fragile-X Mental Retardation Protein as playing the most prominent role in this dataset of treatment-resistant schizophrenia samples. The word "prominent" is used here as these two methods reported the same finding in different ways: the linear SVMs always assigned the highest valued coefficient to the input values of the FMRP set, whereas MAGMA reported it as the most significantly associated gene set, after correcting for multiple comparisons. Recalling the information from chapter 2 about linear SVMs, the algorithm assigns coefficient values to all of the input features, which describes the position and size of the vector that is orthogonal to the decision hyper-plane and can be used to describe the relevant importance that the model is placing on the different inputs. In short, if a feature is providing no information it will be assigned a value close to zero (or even exactly zero if there is no information and a Lasso penalty is being used (Hastie et al., 2001)). When the inputs of the gene sets and the gene set classes were provided to the SVMs, then the models *always* assigned the largest coefficient to the FMRP set, or the class containing the FMRP set: "subcellular neuronal". Additional studies into the nature of the genes and SNPs falling within this dataset showed that the association signal could not be explained by the size of the genes (a concern that was raised by (Ouwenga and Dougherty, 2015)), as these were shown to be average compared to all of the other sets.

FMRP is the product of the FMR1 gene, and is involved in the transportation of target Ribose-Nucleic Acid (RNA) molecules from the nuclei of neurons out to the synapses to develop their functionality and structure, and a complete loss of the protein production results in Fragile-X syndrome (Antar and Bassell, 2003; Bear et al., 2004; Huber et al., 2002; Feng et al., 1997; Fromer et al., 2014; Weiler and Greenough, 1999). As was pointed out in chapter 5, the mechanisms involved in these processes, and the effect on further transcription of FMRP are very complicated (Hall et al., 2015). It has recently been reported that this protein can have an effect on Long Term Depression (LTD) during neuronal network development. This results in a long lasting decrease in synaptic strength between neurons; the most well known of these processes is regulated by the N-methyl-D-aspartate (NMDA) receptors, but there is another mechanism that is the result of the activation of group 1 Metabotropic Glutamate Receptors (mGluRs) (Bear et al., 2004; Oliet et al., 1997). The main difference between these is that the NMDA mechanism is believed to be reversible, whereas the mGluR is not, and can therefore be a precursor to the elimination of synapses (Snyder et al., 2001). In a study of this mechanism using mouse models, Bear et al. (2004) found that this mGluR LTD process was augmented with the absence of FMRP, resulting in a decrease in synaptic connections. They hypothesised that the production of FMRP is required to diminish the amount of LTD from the mGluR process. It must be highlighted, however, that it is not the actual FMR1 gene that was identified in the experiments, but the genes encoding for the RNA binding targets of FMRP, but it has been suggested that there could be some type of feedback mechanism between the FMRP targets and the FMR1 gene, possibly involving an effect on the expression levels of the protein.

These FMRP targets have already been shown to be enriched for risk signal from a variety of different methodologies including previous GWAS results, Copy Number Variant (CNV) studies and data from exome sequencing (Purcell et al., 2014), but the results in chapter 5 suggest that there are interactions taking place between the genes contained within the FMRP set, a finding that is believed to be first reported here. This conclusion was drawn after observing the increased performance of the RBF kernel over the linear kernel for the gene score study. The simulated phenotype procedures showed that the increased performance of the RBF kernel *only* happened when interactions were taken into account. This finding could be very valuable in helping to direct future research into the processes of FMRP and possible concomitant effects, such as the mGluR effects on LTD. Any identification of interacting genes would allow future lab work to be directed towards investigating the possible mechanisms that take place between these genes, and if they could be manipulated in future therapy and medication.

6.2.2. Caveats of findings

It is important to highlight the possible caveats that could have arisen in these studies with any interpretation of the possible implications of the results. The main conclusion from all of the trials carried out is that a superior performance of a SVM model using a RBF kernel over another using a linear kernel strongly suggests a role of interactions in the dataset, due to the fact that these were the only conditions in which this pattern was seen in the simulated phenotypes. The largest caveat with this finding is that the simulation trials only involved *pairwise* interactions. It is possible that higher order interactions could be having a very different effect on outcomes. However, the only way that this could have an adverse effect on the conclusions made thus far is that the presence of higher order interactions *could* result in the improvement of the performance of the linear kernel. It still does not nullify the finding that in these trials, the interactions had to be present for the RBF kernel to perform at a higher level.

The findings of these trials must also be replicated in another dataset involving treatment-resistant schizophrenia. While it was very reassuring to see that these findings were relevant for the data on both of the genotyping chips, a follow-up study is still crucial before any concrete conclusions can be made. This can actually be put into immediate effect as there is already data available at Cardiff University from a second CLOZUK study, using an increased number of samples.

6.2.3. Suggestions for future studies

This section will outline some possible follow-up work that could be carried out in light of the findings made in this thesis. The most important topics to tackle are the two main caveats pointed out in the previous section. A replication study could be performed very quickly using the new CLOZUK data, so this should be the priority as it poses no methodological problems, but could result in a more robust conclusion if the results are replicated. However, tackling the issue of higher order interactions could be more challenging.

Any attempt to do so would have to make use of new software and hardware that could deal with the very large amount of data that would be made from looking at the different combinations of interactions. All of the trials in this thesis were run on a Linux node on a cluster that contained 16 Central Processing Units (CPUs), using the inbuilt parallel processing functionality that is in the *scikit-learn* package for Python (Pedregosa et al., 2011). However, it is doubtful that this would be suitable for any larger datasets, as the software was not able to use further nodes with additional CPUs. These new trials would most probably have to make use of new developments in the field of cluster and cloud computing, with a specific example being the *Apache Spark*[1] software, which is gaining in popularity because of its ability to carry out distributed parallel processes that can be completed at a high speed due to how the software is able to cache any reused data in Random Access Memory (RAM). Following the initial success of this software, there have also been implementations developed that allow for machine learning algorithms to be performed (Meng et al., 2016; Venkataraman et al.), without the specific knowledge of how to distribute these routines and data across computers. Another advantage is that there are already processes in place to make use of this software using researcher-friendly languages like Python and R, and this will continue to develop (Armbrust et al., 2015). The use of this type of technology would be required to both run the models with the interactions and to perform the interpretation analysis to look at the different effect sizes of these, similar to the procedures performed in chapter 4, but on a far larger scale.

Other directions for future work could involve trying to find the subsets of the features provided to the models that are, in fact, interacting with each other. There are systems and methods that have been developed to try and deal with this situation, as it is completely unfeasible to perform a brute force search of all possible combinations of different subsets of the input features. One such system, proposed by Liu et al. (2011), seeks to actively examine the contribution of the different features after the mapping of the RBF kernel has been performed, via a complex mathematical process of expanding the kernel function out into a "Maclaurin" series (a variation of a "Taylor" series, which is used to find polynomial approximations of mathematical functions) and then looking for the information gained and lost by each feature when it is either added or removed from the model. Other methods can involve searching for subsets of the features in a non-random and intelligent manner. These approaches often adopt a Monte Marlo Markov Chain (MCMC) approach, whereby a stochastic process to select new features is based on the performance of the model at each iteration. An example of this approach can be seen in Dramiński et al. (2008), where various decision tree classifiers were built by sampling many different sets for training from the original data, with each set only containing a small selection of the original number of features. The authors concluded that the approach does help to identify subsets of interacting genes, but requires a high computational demand. Another approach is to use implementations of the "Simulated Annealing" algorithm (Metropolis et al., 1953), whereby different selections of subsets are chosen based on the changing levels of performance. In this process, if a new selection of features results in a higher level of performance, then this subset is automatically selected; however, if the new selection results in a *lower* level of performance, then it can still be selected, based on a certain

[1] Website: `http://spark.apache.org/`

probability value. This probability value decreases at every iteration, meaning that there is less chance of poorly performing subsets being chosen at a later stage. The idea behind this method is that many different subsets can be tested out at the start of the process, without dwelling on any sets that happen to perform well at an early stage, but later on in the process, only the better features tend to survive. An example of this being put into place for gene-selection using micro-array data, but with the stated intention of being used for different types of dataset, can be seen in Lin et al. (2008).

In order for any of these methods to be used in research, they need to be implemented into software packages that researchers can readily incorporate into their analysis. Unfortunately, at the time of writing, none of these was implemented into *scikit-learn* for Python. However, they have been incorporated into a suite of machine learning tools for the R language in the *caret* package (Kuhn, 2008). There was an attempt to use this package in the early stages of this thesis, but it was deemed to be too slow for the size of the dataset. This could, however, change in the near future as there have been very recent developments by Microsoft, who have acquired a company called Revolution Analytics, making the R language more memory efficient and capable of dealing with larger datasets[1].

6.3. Final conclusions

The main conclusion from this thesis is that machine learning algorithms, when properly tuned and built, can provide a useful means for a researcher to carry out an initial search for any evidence of interactions taking place within their datasets. It is incredibly computationally expensive and time consuming to carry out manual searches for these, so even a negative finding, with the linear methods outperforming the RBF kernels, can be incredibly useful and hopefully prevent a fruitless search. The machine learning methods cannot improve on current techniques used to identify association signals in main effects, as they have shown superior performance and greater ease of application with examples like the polygenic score methods and the MAGMA software, but they can be used to detect non-linear effects and hopefully future research will allow these to be more accurately identified to help guide any proposals for future research in the field of psychiatric genetics.

[1] Webpage at: `https://mran.microsoft.com/open/`

A. CLOZUK SNP Study

Different input types for the 125 GWAS significant SNPs dataset

The following figures show the spread of the different Receiver Operating Characteristic (ROC) scores from the Cross-Validation (CV) procedure for the different types of inputs in the 125 Genome-Wide Association Study (GWAS) significant Single Nucleotide Polymorphisms (SNPs) study. The scaled inputs were included in the main chapter, but the raw allele counts and Log Odds Ratio (LOR) weighted inputs were not.

The different kernels will be shown in turn:

Linear Kernel

The scatter plots for the different values of **C** for the different inputs types can be seen in figure A.1. The only marked feature of interest is that there is a distinct drop off in performance for the lower values of **C** for the weighted inputs.

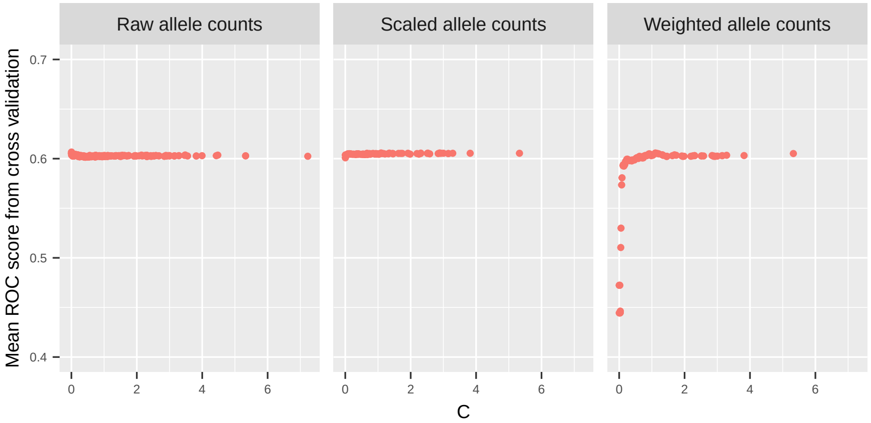


Figure A.1.: The scatter plots of the different ROC scores across the different values of the **C** parameter for the different input types, with the linear kernel.

Polynomial Kernel

The scatter plots for the distributions of performance across the different values of **C** and $\boldsymbol{\gamma}$ are shown in figures A.2 and A.3.

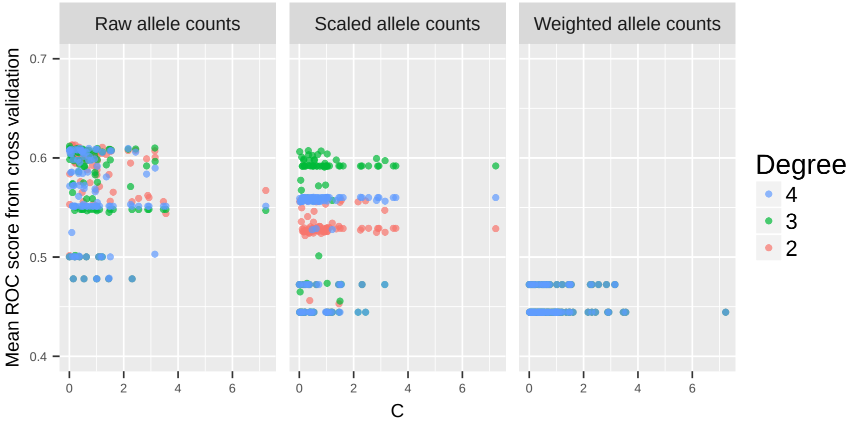


Figure A.2.: The scatter plots of the different ROC scores across the different values of the **C** parameter for the different input types, with the polynomial kernel.

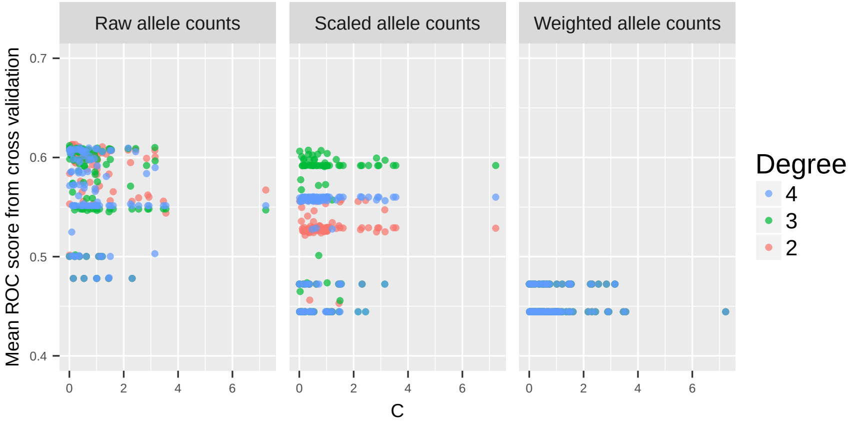


Figure A.3.: The scatter plots of the different ROC scores across the different values of the **gamma** parameter for the different input types, with the polynomial kernel.

RBF Kernel

The scatter plots for the distributions of performance across the different values of **C** and $\boldsymbol{\gamma}$ are shown in

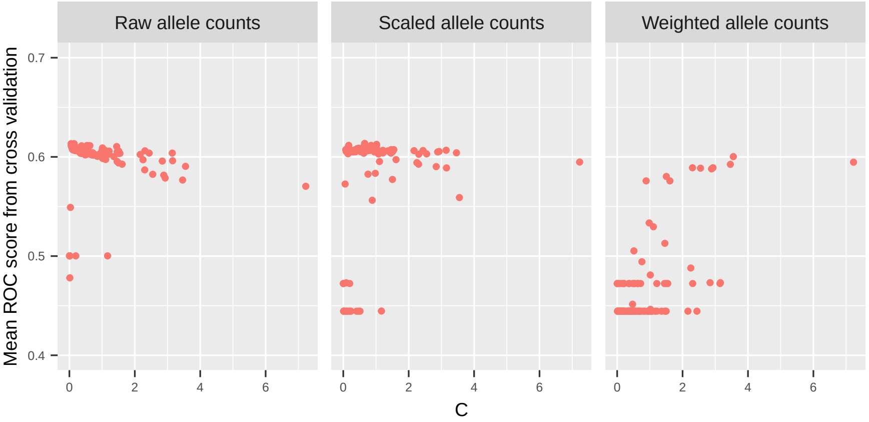


Figure A.4.: The scatter plots of the different ROC scores across the different values of the **C** parameter for the different input types, with the RBF kernel.

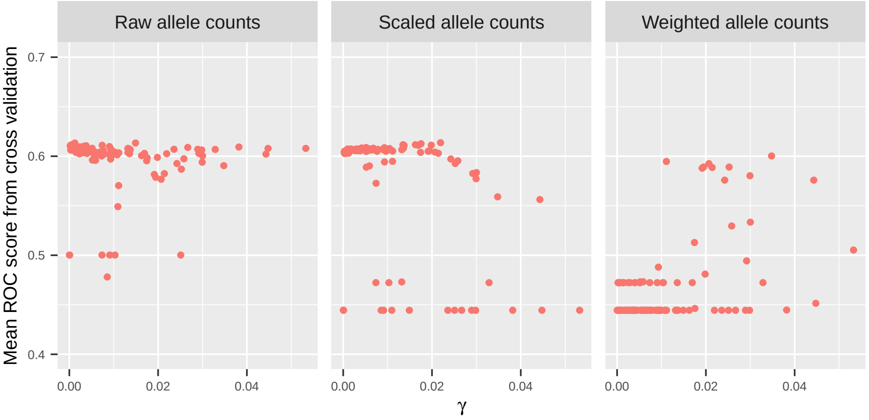


Figure A.5.: The scatter plots of the different ROC scores across the different values of the **gamma** parameter for the different input types, with the RBF kernel.

Pairwise t-test results for 125 GWAS dataset using information from both chips

Table A.1 shows the p-values of pairwise t-test between the polygenic score and all of the SVM kernels for the 125 GWAS significants SNPs, using the samples from both chips. All of these values have had Bonferroni correction for multiple comparison testing, and are still all highly significant.

Table A.1.: The p-values for the pairwise t-tests between the polygenic score and all of the SVM kernels for the information on the 125 GWAS significant SNPs obtained from both chips.

Algorithm	Comparison with polygenic score
Linear	$p < 4.5 \times 10^{-9}$
Polynomial 2	$p < 1 \times 10^{-1000}$
Polynomial 3	$p < 4.3 \times 10^{-148}$
Polynomial 4	$p < 1 \times 10^{-1000}$
RBF	$p < 1.9 \times 10^{-25}$

Demonstration of overlapping points for the polynomial kernels for the weighted inputs

The information shown in figure A.6 shows the different performance values across the values of **C** for the different degrees of the polynomial kernel when the weighted inputs from the 4,998 SNPs in the threshold datasets are used. In the figure shown in the main chapter, it was mainly the blue points that could be seen, and this is because these points are layered on top of each other in turn.

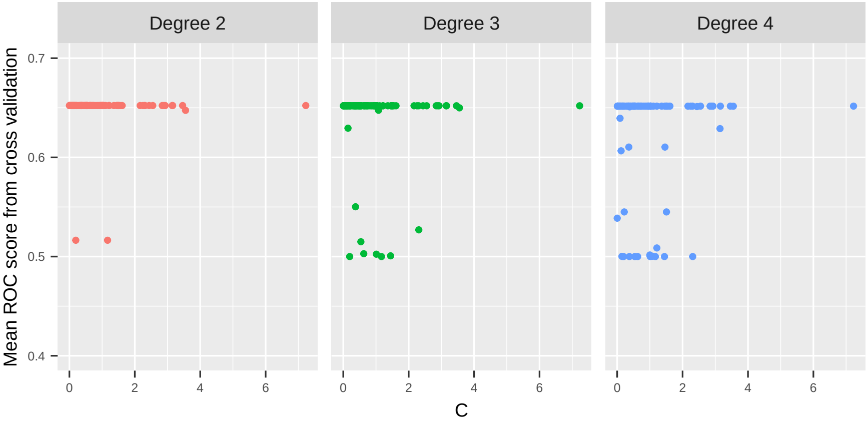


Figure A.6.: Scatterplot of the performances of the different polynomial kernel degrees for the weighted inputs using the threshold dataset. The places where the points overlap can clearly be seen.

B. CLOZUK Gene Set Study

Difference between l^1 and l^2 performance

The information shown in table B.1 shows the results from pairwise t-tests comparing the difference in performance for the different penalisation methods for all the inputs. A higher t-value suggests that the l^1 penalty is doing better.

Table B.1.: Table showing the median results for the performance of all the algorithms on both the gene set classes and the gene sets for the two different border regions.

Input Type	Algorithm	Border Region	t-value(99)	p-value
Gene set classes	Linear SVM	Genes Only	1.65	0.1
		UTR Window	-2.02	0.046
	Multiple Regression	Genes Only	-0.43	0.67
		UTR Window	-2.05	0.04
Gene sets	Linear SVM	Genes Only	15.4	2.2×10^{-16}
		UTR Window	19.22	2.2×10^{-16}
	Multiple Regression	Genes Only	0.36	0.7
		UTR Window	1.71	0.09

Bibliography

1000 Genomes Project Consortium and others (2012). An integrated map of genetic variation from 1,092 human genomes. *Nature*, 491(7422):56–65.

Abraham, T. H. (2002). (Physio)Logical Circuits: The intellectual origins of the McCullock-Pitts Neural Networks. *Journal of the History of the Behavioral Sciences*, 38(1):3–25.

Aguiar-Pulido, V., Seoane, J. A., Rabuñal, J. R., Dorado, J., Pazos, A., and Munteanu, C. R. (2010). Machine learning techniques for Single Nucleotide Polymorphism—Disease classification models in schizophrenia. *Molecules*, 15(7):4875–4889.

American Psychiatric Association (1968). *Diagnostic and statistical manual of mental disorders: DSM-2*. American Psychiatric Association, Washington, DC, 2nd edition.

American Psychiatric Association (1980). *Diagnostic and statistical manual of mental disorders: DSM-3*. American Psychiatric Association, Washington, DC, 3rd edition.

American Psychiatric Association (1994). *Diagnostic and statistical manual of mental disorders: DSM-4*. American Psychiatric Association, Washington, DC, 4th edition.

American Psychiatric Association (2000). *Diagnostic and statistical manual of mental disorders - Text Review: DSM-4-TR*. American Psychiatric Association, Washington, DC, 4th - text review edition.

American Psychiatric Association (2013). *Diagnostic and statistical manual of mental disorders: DSM-5*. American Psychiatric Association, Washington, DC, 5th edition.

Angst, J. (2002). Historical aspects of the dichotomy between manic–depressive disorders and schizophrenia. *Schizophrenia research*, 57(1):5–13.

Angst, J., Scharfetter, C., and Stassen, H. (1983). Classification of schizo-affective patients by multidimensional scaling and cluster analysis. *Psychopathology*, 16(2-4):254–264.

Antar, L. and Bassell, G. (2003). Sunrise at the synapse: the FMRP mRNP shaping the synaptic interface. *Neuron*, 37(4):555–558.

Armbrust, M., Xin, R. S., Lian, C., Huai, Y., Liu, D., Bradley, J. K., Meng, X., Kaftan, T., Franklin, M. J., Ghodsi, A., et al. (2015). Spark SQL: Relational data processing in Spark. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, pages 1383–1394. ACM.

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., et al. (2000). Gene Ontology: Tool for the unification of biology. *Nature genetics*, 25(1):25–29.

Badano, J. L. and Katsanis, N. (2002). Beyond Mendel: An evolving view of human genetic disease transmission. *Nature Reviews Genetics*, 3(10):779–789.

Barrett, J. C., Clayton, D. G., Concannon, P., Akolkar, B., Cooper, J. D., Erlich, H. A., Julier, C., Morahan, G., Nerup, J., Nierras, C., et al. (2009). Genome-wide association study and meta-analysis find that over 40 loci affect risk of type 1 diabetes. *Nature genetics*, 41(6):703–707.

Barrett, L. W., Fletcher, S., and Wilton, S. D. (2012). Regulation of eukaryotic gene expression by the untranslated gene regions and other non-coding elements. *Cellular and Molecular Life Sciences*, 69(21):3613–3634.

Bayés, À., van de Lagemaat, L. N., Collins, M. O., Croning, M. D., Whittle, I. R., Choudhary, J. S., and Grant, S. G. (2011). Characterization of the proteome, diseases and evolution of the human postsynaptic density. *Nature neuroscience*, 14(1):19–21.

Bear, M. F., Huber, K. M., and Warren, S. T. (2004). The mGluR theory of fragile X mental retardation. *Trends in neurosciences*, 27(7):370–377.

Beer, M. D. (1996). Psychosis: A history of the concept. *Comprehensive psychiatry*, 37(4):273–291.

Berrios, G. and Beer, D. (1994). The notion of unitary psychosis: A conceptual history. *History of Psychiatry*, 5(17):013–36.

Bishop, C. M. (2006). *Pattern recognition and machine learning (Information science and statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Blake, J. A., Bult, C. J., Eppig, J. T., Kadin, J. A., Richardson, J. E., Group, M. G. D., et al. (2013). The Mouse Genome Database: Integration of and access to knowledge about the laboratory mouse. *Nucleic acids research*, 42(1):810–817.

Breiman, L. (1996). Bagging predictors. *Machine learning*, 24(2):123–140.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Brockington, I., Kendell, R., Wainwright, S., Hillier, V., and Walker, J. (1979). The distinction between the affective psychoses and schizophrenia. *The British Journal of Psychiatry*, 135(3):243–248.

Bruijnzeel, D. and Tandon, R. (2011). The concept of schizophrenia: From the 1850s to the DSM-5. *Psychiatric Annals*, 41(5):289.

Bundy, H., Stahl, D., and MacCabe, J. (2011). A systematic review and meta-analysis of the fertility of patients with schizophrenia and their unaffected relatives. *Acta Psychiatrica Scandinavica*, 123(2):98–106.

Burton, P. R., Clayton, D. G., Cardon, L. R., Craddock, N., Deloukas, P., Duncanson, A., Kwiatkowski, D. P., McCarthy, M. I., Ouwehand, W. H., Samani, N. J., et al. (2007). Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls. *Nature*, 447(7145):661–678.

Cabungcal, J.-H., Counotte, D. S., Lewis, E. M., Tejeda, H. A., Piantadosi, P., Pollock, C., Calhoon, G. G., Sullivan, E. M., Presgraves, E., Kil, J., et al. (2014). Juvenile antioxidant treatment prevents adult deficits in a developmental model of schizophrenia. *Neuron*, 83(5):1073–1084.

Cardno, A. G. and Gottesman, I. I. (2000). Twin studies of schizophrenia: From bow-and-arrow concordances to Star Wars Mx and functional genomics. *American journal of medical genetics*, 97(1):12–17.

Cardno, A. G. and Owen, M. J. (2014). Genetic relationships between schizophrenia, bipolar disorder, and schizoaffective disorder. *Schizophrenia bulletin*, 40(3):504–515.

Cardon, L. R. and Palmer, L. J. (2003). Population stratification and spurious allelic association. *The Lancet*, 361(9357):598–604.

Cariaga-Martinez, A., Saiz-Ruiz, J., and Alelú-Paz, R. (2016). From linkage studies to epigenetics: What we know and what we need to know in the neurobiology of schizophrenia. *Frontiers in neuroscience*, 10.

Casci, T. (2010). Population genetics: SNPs that come in threes. *Nature Reviews Genetics*, 11(1):8–8.

Chang, C. C., Chow, C. C., Tellier, L. C., Vattikuti, S., Purcell, S. M., and Lee, J. J. (2014). Second-generation PLINK: Rising to the challenge of larger and richer datasets. *ArXiv e-prints*.

Chang, C.-C. and Lin, C.-J. (2011). LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):1–27. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Cheng, S., Andrew, A. S., Andrews, P. C., and Moore, J. H. (2016). Complex systems analysis of bladder cancer susceptibility reveals a role for decarboxylase activity in two genome-wide association studies. *BioData Mining*, 9(1):40.

Cheniaux, E., Landeira-Fernandez, J., Telles, L. L., Lessa, J. L. M., Dias, A., Duncan, T., and Versiani, M. (2008). Does schizoaffective disorder really exist? A systematic review of the studies that compared schizoaffective disorder with schizophrenia or mood disorders. *Journal of Affective Disorders*, 106(3):209–217.

Chung, R.-H., Tsai, W.-Y., Hsieh, C.-H., Hung, K.-Y., Hsiung, C. A., and Hauser, E. R. (2015). Seqsimla2: Simulating correlated quantitative traits accounting for shared environmental effects in user-specified pedigree structure. *Genetic epidemiology*, 39(1):20–24.

Cleveland, W., Grosse, E., and Shyu, W. (1992). Local regression models. In Chambers, J. and Hastie, T., editors, *Statistical Models in S*, chapter 8. Wadsworth & Brooks/Cole.

Collier, D. A., Eastwood, B. J., Malki, K., and Mokrab, Y. (2016). Advances in the genetics of schizophrenia: Toward a network and pathway view for drug discovery. *Annals of the New York Academy of Sciences*, 1366(1):61–75.

Commission, S. et al. (2012). The abandoned illness: A report from the Schizophrenia Commission. *London: Rethink Mental Illness*.

Cordell, H. J. (2009). Detecting gene–gene interactions that underlie human diseases. *Nature Reviews Genetics*, 10(6):392–404.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.

Corvin, A. and Sullivan, P. F. (2016). What next in schizophrenia genetics for the psychiatric genomics consortium? *Schizophrenia bulletin*, 42(3):538–541.

Cover, T. M. (1965). Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. *IEEE transactions on electronic computers*, (3):326–334.

Craddock, N. and Owen, M. J. (2005). The beginning of the end for the Kraepelinian dichotomy. *The British Journal of Psychiatry*, 186(5):364–366.

Craddock, N. and Owen, M. J. (2010). The Kraepelinian dichotomy–going, going... but still not gone. *The British Journal of Psychiatry*, 196(2):92–95.

Cross-Disorder Group of the Psychiatric Genomics Consortium (2013). Identification of risk loci with shared effects on five major psychiatric disorders: A genome-wide analysis. *The Lancet*, 381(9875):1371–1379.

Crow, T. (1986). The continuum of psychosis and its implication for the structure of the gene. *The British Journal of Psychiatry*, 149(4):419–429.

Cruz, J. A. and Wishart, D. S. (2006). Applications of machine learning in cancer prediction and prognosis. *Cancer informatics*, 2.

Daly, M. J., Rioux, J. D., Schaffner, S. F., Hudson, T. J., and Lander, E. S. (2001). High-resolution haplotype structure in the human genome. *Nature genetics*, 29(2):229–232.

Darnell, J. C., Van Driesche, S. J., Zhang, C., Hung, K. Y. S., Mele, A., Fraser, C. E., Stone, E. F., Chen, C., Fak, J. J., Chi, S. W., et al. (2011). FMRP stalls ribosomal translocation on mRNAs linked to synaptic function and autism. *Cell*, 146(2):247–261.

de Leeuw, C. A., Mooij, J. M., Heskes, T., and Posthuma, D. (2015). MAGMA: Generalized gene-set analysis of GWAS data. *PLoS Comput Biol*, 11(4):e1004219.

Dramiński, M., Rada-Iglesias, A., Enroth, S., Wadelius, C., Koronacki, J., and Komorowski, J. (2008). Monte Carlo feature selection for supervised classification. *Bioinformatics*, 24(1):110–117.

Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., and Lin, C.-J. (2008a). LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874.

Fan, Y., Resnick, S. M., Wu, X., and Davatzikos, C. (2008b). Structural and functional biomarkers of prodromal Alzheimer's disease: A high-dimensional pattern classification study. *Neuroimage*, 41(2):277–285.

Farrell, M., Werge, T., Sklar, P., Owen, M., Ophoff, R., O'donovan, M., Corvin, A., Cichon, S., and Sullivan, P. F. (2015). Evaluating historical candidate genes for schizophrenia. *Molecular psychiatry*, 20(5):555–562.

Fatemi, S. H. and Folsom, T. D. (2009). The neurodevelopmental hypothesis of schizophrenia, revisited. *Schizophrenia bulletin*, page sbn187.

Feighner, J. P., Robins, E., Guze, S. B., Woodruff, R. A., Winokur, G., and Munoz, R. (1972). Diagnostic criteria for use in psychiatric research. *Archives of general psychiatry*, 26(1):57–63.

Feng, Y., Absher, D., Eberhart, D. E., Brown, V., Malter, H. E., and Warren, S. T. (1997). FMRP associates with polyribosomes as an mRNP, and the I304N mutation of severe fragile X syndrome abolishes this association. *Molecular cell*, 1(1):109–118.

Fernández, M. and Miranda-Saavedra, D. (2012). Genome-wide enhancer prediction from epigenetic signatures using genetic algorithm-optimized support vector machines. *Nucleic acids research*, 40(10):e77–e77.

Fink, M. and Taylor, M. A. (2006). *Catatonia: A clinician's guide to diagnosis and treatment*. Cambridge University Press.

Flasiński, M. (2016). History of artificial intelligence. In *Introduction to Artificial Intelligence*, chapter 1, pages 3–13. Springer.

Folsom, T. D., Thuras, P. D., and Fatemi, S. H. (2015). Protein expression of targets of the FMRP regulon is altered in brains of subjects with schizophrenia and mood disorders. *Schizophrenia research*, 165(2):201–211.

Fromer, M., Pocklington, A. J., Kavanagh, D. H., Williams, H. J., Dwyer, S., Gormley, P., Georgieva, L., Rees, E., Palta, P., Ruderfer, D. M., et al. (2014). De novo mutations in schizophrenia implicate synaptic networks. *Nature*, 506(7487):179–184.

Gauderman, W. J. (2002). Sample size requirements for matched case-control studies of gene–environment interaction. *Statistics in medicine*, 21(1):35–50.

Geiringer, H. (1944). On the probability theory of linkage in Mendelian heredity. *The Annals of Mathematical Statistics*, 15(1):25–57.

Gibbs, R. A., Belmont, J. W., Hardenbol, P., Willis, T. D., Yu, F., Yang, H., Ch'ang, L.-Y., Huang, W., Liu, B., Shen, Y., et al. (2003). The international HapMap project. *Nature*, 426(6968):789–796.

Ginovart, N. and Kapur, S. (2012). Role of dopamine D2 receptors for antipsychotic activity. In *Current antipsychotics*, pages 27–52. Springer.

Girard, S. L., Gauthier, J., Noreau, A., Xiong, L., Zhou, S., Jouan, L., Dionne-Laporte, A., Spiegelman, D., Henrion, E., Diallo, O., et al. (2011). Increased exonic de novo mutation rate in individuals with schizophrenia. *Nature genetics*, 43(9):860–863.

Gottesman, I. I. and Shields, J. (1967). A polygenic theory of schizophrenia. *International Journal of Mental Health*, 1(1-2):107–115.

Gottesman, I. I. and Shields, J. (1972). *Schizophrenia and genetics: A twin study vantage point*. Academic Press.

Graña, M., Termenon, M., Savio, A., Gonzalez-Pinto, A., Echeveste, J., Pérez, J., and Besga, A. (2011). Computer aided diagnosis system for Alzheimer disease using brain diffusion tensor imaging features selected by Pearson's correlation. *Neuroscience letters*, 502(3):225–229.

Green, E., Rees, E., Walters, J., Smith, K., Forty, L., Grozeva, D., Moran, J., Sklar, P., Ripke, S., Chambert, K., et al. (2015). Copy number variation in bipolar disorder. *Molecular psychiatry*.

Hagerman, R. J., Berry-Kravis, E., Kaufmann, W. E., Ono, M. Y., Tartaglia, N., Lachiewicz, A., Kronk, R., Delahunty, C., Hessl, D., Visootsak, J., et al. (2009). Advances in the treatment of fragile X syndrome. *Pediatrics*, 123(1):378–390.

Hahn, L. W., Ritchie, M. D., and Moore, J. H. (2003). Multifactor dimensionality reduction software for detecting gene–gene and gene–environment interactions. *Bioinformatics*, 19(3):376–382.

Hahn, T., Marquand, A. F., Ehlis, A.-C., Dresler, T., Kittel-Schneider, S., Jarczok, T. A., Lesch, K.-P., Jakob, P. M., Mourao-Miranda, J., Brammer, M. J., et al. (2011). Integrating neurobiological markers of depression. *Archives of general psychiatry*, 68(4):361–368.

Hall, J., Trent, S., Thomas, K. L., O'Donovan, M. C., and Owen, M. J. (2015). Genetic risk for schizophrenia: Convergence on synaptic pathways involved in plasticity. *Biological psychiatry*, 77(1):52–58.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18.

Hamshere, M. L., Walters, J. T. R., Smith, R., Richards, A., Green, E., Grozeva, D., Jones, I., Forty, L., Jones, L., Gordon-Smith, K., et al. (2013). Genome-wide significant associations in schizophrenia to ITIH3/4, CACNA1C and SDCCAG8, and extensive replication of associations reported by the Schizophrenia PGC. *Molecular psychiatry*, 18(6):708–712.

Hastie, T., Tibshirani, R., and Friedman, J. (2001). *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA.

Hemani, G., Shakhbazov, K., Westra, H.-J., Esko, T., Henders, A. K., McRae, A. F., Yang, J., Gibson, G., Martin, N. G., Metspalu, A., et al. (2014). Detection and replication of epistasis influencing transcription in humans. *Nature*, 508(7495):249–253.

Hemani, G., Theocharidis, A., Wei, W., and Haley, C. (2011). EpiGPU: Exhaustive pairwise epistasis scans parallelized on consumer level graphics cards. *Bioinformatics*, 27(11):1462–1465.

Ho, T. K. (1995). Random decision forests. In *Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on*, volume 1, pages 278–282. IEEE.

Ho, T. K. (1998). The random subspace method for constructing decision forests. *IEEE transactions on pattern analysis and machine intelligence*, 20(8):832–844.

Hodgkinson, A. and Eyre-Walker, A. (2010). Human triallelic sites: Evidence for a new mutational mechanism? *Genetics*, 184(1):233–241.

Hoenig, J. (1983). The concept of schizophrenia. Kraepelin-Bleuler-Schneider. *The British Journal of Psychiatry*, 142(6):547–556.

Holmans, P., Green, E. K., Pahwa, J. S., Ferreira, M. A., Purcell, S. M., Sklar, P., Owen, M. J., O'Donovan, M. C., Craddock, N., Consortium, W. T. C.-C., et al. (2009). Gene ontology analysis of GWA study data sets provides insights into the biology of bipolar disorder. *The American Journal of Human Genetics*, 85(1):13–24.

Holzinger, E. R., Szymczak, S., Dasgupta, A., MALLEY, J., Li, Q., and Bailey-Wilson, J. E. (2015). Variable selection method for the identification of epistatic models. In *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, volume 20, page 195. NIH Public Access.

Huber, K. M., Gallagher, S. M., Warren, S. T., and Bear, M. F. (2002). Altered synaptic plasticity in a mouse model of fragile X mental retardation. *Proceedings of the National Academy of Sciences*, 99(11):7746–7750.

Hyman, S. E. (2007). Can neuroscience be integrated into the DSM-V? *Nature Reviews Neuroscience*, 8(9):725–732.

Hyman, S. E. (2010). The diagnosis of mental disorders: The problem of reification. *Annual review of clinical psychology*, 6:155–179.

Iafrate, A. J., Feuk, L., Rivera, M. N., Listewnik, M. L., Donahoe, P. K., Qi, Y., Scherer, S. W., and Lee, C. (2004). Detection of large-scale variation in the human genome. *Nature genetics*, 36(9):949–951.

Ince, N. F., Goksu, F., Pellizzer, G., Tewfik, A., and Stephane, M. (2008). Selection of spectro-temporal patterns in multichannel MEG with support vector machines for schizophrenia classification. In *2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 3554–3557. IEEE.

Jablensky, A. (1999). The conflict of the nosologists: Views on schizophrenia and manic-depressive illness in the early part of the 20th century. *Schizophrenia Research*, 39(2):95–100.

Jia, P., Wang, L., Meltzer, H. Y., and Zhao, Z. (2011). Pathway-based analysis of GWAS datasets: Effective but caution required. *The International Journal of Neuropsychopharmacology*, 14(04):567–572.

Johnson, M. B., Kawasawa, Y. I., Mason, C. E., Krsnik, Ž., Coppola, G., Bogdanović, D., Geschwind, D. H., Mane, S. M., State, M. W., and Šestan, N. (2009). Functional and evolutionary insights into human brain development through global transcriptome analysis. *Neuron*, 62(4):494–509.

Johnson, R. C., Nelson, G. W., Troyer, J. L., Lautenberger, J. A., Kessing, B. D., Winkler, C. A., and O'Brien, S. J. (2010). Accounting for multiple comparisons in a genome-wide association study (GWAS). *BMC genomics*, 11(1):1.

Kanehisa, M., Araki, M., Goto, S., Hattori, M., Hirakawa, M., Itoh, M., Katayama, T., Kawashima, S., Okuda, S., Tokimatsu, T., et al. (2008). KEGG for linking genomes to life and the environment. *Nucleic acids research*, 36(suppl 1):D480–D484.

Kavanagh, D., Tansey, K., O'donovan, M., and Owen, M. (2015). Schizophrenia genetics: Emerging themes for a complex disorder. *Molecular psychiatry*, 20(1):72–76.

Kendell, R. (1987). Diagnosis and classification of functional psychoses. *British medical bulletin*, 43(3):499–513.

Kendell, R. and Gourlay, J. (1970). The clinical distinction between the affective psychoses and schizophrenia. *The British Journal of Psychiatry*, 117(538):261–266.

Kendell, R. and Jablensky, A. (2003). Distinguishing between the validity and utility of psychiatric diagnoses. *American journal of psychiatry*.

Keshavan, M. S. (1999). Development, disease and degeneration in schizophrenia: A unitary pathophysiological model. *Journal of psychiatric research*, 33(6):513–521.

Keshavan, M. S. and Hogarty, G. E. (1999). Brain maturational processes and delayed onset in schizophrenia. *Development and psychopathology*, 11(03):525–543.

Kessler, R. C., Chiu, W. T., Demler, O., and Walters, E. E. (2005). Prevalence, severity, and comorbidity of 12-month DSM-IV disorders in the National Comorbidity Survey Replication. *Archives of general psychiatry*, 62(6):617–627.

Khodayari-Rostamabad, A., Hasey, G. M., MacCrimmon, D. J., Reilly, J. P., and de Bruin, H. (2010). A pilot study to determine whether machine learning methodologies using pre-treatment electroencephalography can predict the symptomatic response to clozapine therapy. *Clinical Neurophysiology*, 121(12):1998–2006.

Kirov, G., Pocklington, A., Holmans, P., Ivanov, D., Ikeda, M., Ruderfer, D., Moran, J., Chambert, K., Toncheva, D., Georgieva, L., et al. (2012). De novo CNV analysis implicates specific abnormalities of postsynaptic signalling complexes in the pathogenesis of schizophrenia. *Molecular psychiatry*, 17(2):142–153.

Klein, R. J., Zeiss, C., Chew, E. Y., Tsai, J.-Y., Sackler, R. S., Haynes, C., Henning, A. K., SanGiovanni, J. P., Mane, S. M., Mayne, S. T., et al. (2005). Complement factor H polymorphism in age-related macular degeneration. *Science*, 308(5720):385–389.

Kleinbaum, D. G. and Klein, M. (2010). *Logistic Regression: A self-learning text*. Springer, 3 edition.

Koo, C. L., Liew, M. J., Mohamad, M. S., and Mohamed Salleh, A. H. (2013). A review for detecting gene-gene interactions using machine learning methods in genetic epidemiology. *BioMed research international*, 2013.

Kuhn, M. (2008). Caret package. *Journal of Statistical Software*, 28(5).

Lander, E. S., Linton, L. M., Birren, B., Nusbaum, C., Zody, M. C., Baldwin, J., Devon, K., Dewar, K., Doyle, M., FitzHugh, W., et al. (2001). Initial sequencing and analysis of the human genome. *Nature*, 409(6822):860–921.

Lek, M., Karczewski, K., Minikel, E., Samocha, K., Banks, E., Fennell, T., O'Donnell-Luria, A., Ware, J., Hill, A., Cummings, B., et al. (2015). Analysis of protein-coding genetic variation in 60,706 humans. *bioRxiv*, page 030338.

Lemm, S., Blankertz, B., Dickhaus, T., and Müller, K.-R. (2011). Introduction to machine learning for brain imaging. *Neuroimage*, 56(2):387–399.

Levy-Lahad, E., Wasco, W., Poorkaj, P., Romano, D. M., et al. (1995). Candidate gene for the chromosome 1 familial Alzheimer's disease locus. *Science*, 269(5226):973.

Lewontin, R. and Kojima, K.-i. (1960). The evolutionary dynamics of complex polymorphisms. *Evolution*, pages 458–472.

Li, C. and Li, M. (2008). Gwasimulator: a rapid whole-genome simulation program. *Bioinformatics*, 24(1):140–142.

Li, H., Wetten, S., Li, L., Jean, P. L. S., Upmanyu, R., Surh, L., Hosford, D., Barnes, M. R., Briley, J. D., Borrie, M., et al. (2008). Candidate single-nucleotide polymorphisms from a genomewide association study of Alzheimer disease. *Archives of neurology*, 65(1):45–53.

Lichtenstein, P., Yip, B. H., Björk, C., Pawitan, Y., Cannon, T. D., Sullivan, P. F., and Hultman, C. M. (2009). Common genetic determinants of schizophrenia and bipolar disorder in Swedish families: A population-based study. *The Lancet*, 373(9659):234–239.

Lin, S.-W., Lee, Z.-J., Chen, S.-C., and Tseng, T.-Y. (2008). Parameter determination of support vector machine and feature selection using simulated annealing approach. *Applied soft computing*, 8(4):1505–1512.

Listgarten, J., Damaraju, S., Poulin, B., Cook, L., Dufour, J., Driga, A., Mackey, J., Wishart, D., Greiner, R., and Zanke, B. (2004). Predictive models for breast cancer susceptibility from multiple single nucleotide polymorphisms. *Clinical Cancer Research*, 10(8):2725–2737.

Liu, H., Yin, J., Xiao, M., Gao, C., Mason, A. S., Zhao, Z., Liu, Y., Li, J., and Fu, D. (2012). Characterization and evolution of $5'$ and $3'$ untranslated regions in eukaryotes. *Gene*, 507(2):106–111.

Liu, Q., Chen, C., Zhang, Y., and Hu, Z. (2011). Feature selection for support vector machines with RBF kernel. *Artificial Intelligence Review*, 36(2):99–115.

Lubs, H. A. (1969). A marker X chromosome. *American journal of human genetics*, 21(3):231.

Machón, R., Mednick, S. A., Schulsinger, F., et al. (1983). The interaction of seasonality, place of birth, genetic risk and subsequent schizophrenia in a high risk sample. *The British Journal of Psychiatry*, 143(4):383–388.

Mahon, P. B., Payne, J. L., MacKinnon, D. F., Mondimore, F. M., Goes, F. S., Schweizer, B., Jancic, D., Coryell, W. H., Holmans, P. A., Shi, J., et al. (2009). Genome-wide linkage and follow-up association study of postpartum mood symptoms. *American Journal of Psychiatry*, 166(11):1229–1237.

Maier, W. (2006). Do schizoaffective disorders exist at all? *Acta Psychiatrica Scandinavica*, 113(5):369–371.

Malhotra, D. and Sebat, J. (2012). CNVs: Harbingers of a rare variant revolution in psychiatric genetics. *Cell*, 148(6):1223–1241.

Mangalore, R., Knapp, M., et al. (2007). Cost of schizophrenia in England. *Journal of Mental Health Policy and Economics*, 10(1):23.

McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133.

McGrath, J., Saha, S., Chant, D., and Welham, J. (2008). Schizophrenia: A concise overview of incidence, prevalence, and mortality. *Epidemiologic reviews*, 30(1):67–76.

McGuffin, P., Farmer, A. E., Gottesman, I. I., Murray, R. M., and Reveley, A. M. (1984). Twin concordance for operationally defined schizophrenia: Confirmation of familiality and heritability. *Archives of General Psychiatry*, 41(6):541–545.

McKinney, W. (2011). pandas: a foundational Python library for data analysis and statistics. *Python for High Performance and Scientific Computing*, pages 1–9.

Mednick, S. A., Machon, R. A., Huttunen, M. O., and Bonett, D. (1988). Adult schizophrenia following prenatal exposure to an influenza epidemic. *Archives of general psychiatry*, 45(2):189–192.

Meng, X., Bradley, J., Yuvaz, B., Sparks, E., Venkataraman, S., Liu, D., Freeman, J., Tsai, D., Amde, M., Owen, S., et al. (2016). Mllib: Machine learning in Apache Spark. *JMLR*, 17(34):1–7.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092.

Metz, C. E. (1978). Basic principles of ROC analysis. In *Seminars in nuclear medicine*, volume 8, pages 283–298. Elsevier.

Meyer, D., Bonhoeffer, T., and Scheuss, V. (2014). Balance and stability of synaptic structures during synaptic plasticity. *Neuron*, 82(2):430–443.

Mitchell, T. (1997). *Machine Learning*. McGraw-Hill.

Murty, K. G. and Yu, F.-T. (1988). *Linear complementarity, linear and nonlinear programming*. Citeseer.

Nagelkerke, N. J. (1991). A note on a general definition of the coefficient of determination. *Biometrika*, 78(3):691–692.

Network, T., of the Psychiatric Genomics Consortium, P. A. S., et al. (2015). Psychiatric genome-wide association study analyses implicate neuronal, immune and histone pathways. *Nature neuroscience*, 18(2):199–209.

Nicodemus, K. K. and Malley, J. D. (2009). Predictor correlation impacts machine learning algorithms: Implications for genomic studies. *Bioinformatics*, 25(15):1884–1890.

Nicodemus, K. K., Malley, J. D., Strobl, C., and Ziegler, A. (2010). The behaviour of random forest permutation-based variable importance measures under predictor correlation. *BMC bioinformatics*, 11(1):1.

Nicodemus, K. K., Wang, W., and Shugart, Y. Y. (2007). Stability of variable importance scores and rankings using statistical learning tools on single-nucleotide polymorphisms and risk factors involved in gene×gene and gene×environment interactions. In *BMC proceedings*, volume 1, page 1. BioMed Central.

Noble, W. S. (2006). What is a support vector machine? *Nature biotechnology*, 24(12):1565–1567.

O'Donovan, M. C., Craddock, N., Norton, N., Williams, H., Peirce, T., Moskvina, V., Nikolov, I., Hamshere, M., Carroll, L., Georgieva, L., et al. (2008). Identification of loci associated with schizophrenia by genome-wide association and follow-up. *Nature genetics*, 40(9):1053–1055.

O'Dushlaine, C., Kenny, E., Heron, E., Donohoe, G., Gill, M., Morris, D., and Corvin, A. (2011). Molecular pathways involved in neuronal cell adhesion and membrane scaffolding contribute to schizophrenia and bipolar disorder susceptibility. *Molecular psychiatry*, 16(3):286–292.

Oliet, S. H., Malenka, R. C., and Nicoll, R. A. (1997). Two distinct forms of long-term depression coexist in CA1 hippocampal pyramidal cells. *Neuron*, 18(6):969–982.

Olson, M. V. (1993). The human genome project. *Proceedings of the National Academy of Sciences*, 90(10):4338–4344.

Orrù, G., Pettersson-Yeo, W., Marquand, A. F., Sartori, G., and Mechelli, A. (2012). Using support vector machine to identify imaging biomarkers of neurological and psychiatric disease: A critical review. *Neuroscience & Biobehavioral Reviews*, 36(4):1140–1152.

Ouwenga, R. L. and Dougherty, J. (2015). FMRP targets or not: Long, highly brain-expressed genes tend to be implicated in autism and brain disorders. *Molecular autism*, 6(1):1.

Owen, M., Sawa, A., and Mortensen, P. (2016). Schizophrenia. *The Lancet*, 388(10039):86–97.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Peng, C.-Y. J., Lee, K. L., and Ingersoll, G. M. (2002). An introduction to logistic regression analysis and reporting. *The journal of educational research*, 96(1):3–14.

Pirooznia, M., Seifuddin, F., Judy, J., Mahon, P. B., Potash, J. B., Zandi, P. P., Consortium, B. G. S. B., et al. (2012). Data mining approaches for genome-wide association of mood disorders. *Psychiatric genetics*, 22(2):55.

Pocklington, A. J., Rees, E., Walters, J. T., Han, J., Kavanagh, D. H., Chambert, K. D., Holmans, P., Moran, J. L., McCarroll, S. A., Kirov, G., et al. (2015). Novel findings from CNVs implicate inhibitory and excitatory signaling complexes in schizophrenia. *Neuron*, 86(5):1203–1214.

Polderman, T. J., Benyamin, B., De Leeuw, C. A., Sullivan, P. F., Van Bochoven, A., Visscher, P. M., and Posthuma, D. (2015). Meta-analysis of the heritability of human traits based on fifty years of twin studies. *Nature genetics*, 47(7):702–709.

Powell, J. E., Henders, A. K., McRae, A. F., Kim, J., Hemani, G., Martin, N. G., Dermitzakis, E. T., Gibson, G., Montgomery, G. W., and Visscher, P. M. (2013). Congruence of additive and non-additive effects on gene expression estimated from pedigree and SNP data. *PLoS Genet*, 9(5):e1003502.

Price, A. L., Patterson, N. J., Plenge, R. M., Weinblatt, M. E., Shadick, N. A., and Reich, D. (2006). Principal components analysis corrects for stratification in genome-wide association studies. *Nature genetics*, 38(8):904–909.

Purcell, S., Neale, B., Todd-Brown, K., Thomas, L., Ferreira, M. A., Bender, D., Maller, J., Sklar, P., De Bakker, P. I., Daly, M. J., et al. (2007). PLINK: A tool set for whole-genome association and population-based linkage analyses. *The American Journal of Human Genetics*, 81(3):559–575.

Purcell, S. M., Moran, J. L., Fromer, M., Ruderfer, D., Solovieff, N., Roussos, P., O'Dushlaine, C., Chambert, K., Bergen, S. E., Kähler, A., et al. (2014). A polygenic burden of rare disruptive mutations in schizophrenia. *Nature*, 506(7487):185–190.

Purcell, S. M., Wray, N. R., Stone, J. L., Visscher, P. M., O'Donovan, M. C., Sullivan, P. F., Sklar, P., Ruderfer, D. M., McQuillin, A., Morris, D. W., et al. (2009). Common polygenic variation contributes to risk of schizophrenia and bipolar disorder. *Nature*, 460(7256):748–752.

R Core Team (2013). *R: A Language and environment for statistical computing.* R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.

Reich, D. E., Cargill, M., Bolk, S., Ireland, J., Sabeti, P. C., Richter, D. J., Lavery, T., Kouyoumjian, R., Farhadian, S. F., Ward, R., et al. (2001). Linkage disequilibrium in the human genome. *Nature*, 411(6834):199–204.

Reich, T., Hinrichs, A., Culverhouse, R., and Bierut, L. (1999). Genetic studies of alcoholism and substance dependence. *The American Journal of Human Genetics*, 65(3):599–605.

Rende, R. D., Plomin, R., and Vandenberg, S. G. (1990). Who discovered the twin method? *Behavior genetics*, 20(2):277–285.

Ripke, S., O'Dushlaine, C., Chambert, K., Moran, J. L., Kähler, A. K., Akterin, S., Bergen, S. E., Collins, A. L., Crowley, J. J., Fromer, M., et al. (2013). Genome-wide association analysis identifies 13 new risk loci for schizophrenia. *Nature genetics*, 45(10):1150–1159.

Risch, N., Merikangas, K., et al. (1996). The future of genetic studies of complex human diseases. *Science*, 273(5281):1516–1517.

Ritchie, M. D., Hahn, L. W., Roodi, N., Bailey, L. R., Dupont, W. D., Parl, F. F., and Moore, J. H. (2001). Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer. *The American Journal of Human Genetics*, 69(1):138–147.

Robins, E. and Guze, S. B. (1970). Establishment of diagnostic validity in psychiatric illness: Its application to schizophrenia. *American Journal of Psychiatry*, 126(7):983–987.

Rodríguez-Testal, J. F., Senín-Calderón, C., and Perona-Garcelán, S. (2014). From DSM-IV-TR to DSM-5: Analysis of some changes. *International Journal of Clinical and Health Psychology*, 14(3):221–231.

Roelcke, V. (1997). Biologizing social facts: An early 20th century debate on Kraepelin‘s concepts of culture, neurasthenia, and degeneration. *Culture, medicine and psychiatry*, 21(4):383–403.

Rogers, S. J., Wehner, E. A., and Hagerman, R. (2001). The behavioral phenotype in fragile X: Symptoms of autism in very young children with fragile X syndrome, idiopathic autism, and other developmental disorders. *Journal of developmental & behavioral pediatrics*, 22(6):409–417.

Rokach, L. and Maimon, O. (2005). Top-down induction of decision trees classifiers-a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 35(4):476–487.

Rosipal, R. and Krämer, N. (2006). Overview and recent advances in partial least squares. In *Subspace, latent structure and feature selection*, pages 34–51. Springer.

Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. *IBM Journal of research and development*, 3(3):210–229.

Sawa, A. and Sedlak, T. W. (2016). Oxidative stress and inflammation in schizophrenia. *Schizophrenia Research*.

Schizophrenia Working Group of the Psychiatric Genomics Consortium et al. (2014). Biological insights from 108 schizophrenia-associated genetic loci. *Nature*, 511(7510):421–427.

Schmutz, J., Wheeler, J., Grimwood, J., Dickson, M., Yang, J., Caoile, C., Bajorek, E., Black, S., Chan, Y. M., Denys, M., et al. (2004). Quality assessment of the human genome sequence. *Nature*, 429(6990):365–368.

Schwarz, D. F., König, I. R., and Ziegler, A. (2010). On safari to random jungle: a fast implementation of random forests for high-dimensional data. *Bioinformatics*, 26(14):1752–1758.

Sebat, J., Lakshmi, B., Troge, J., Alexander, J., Young, J., Lundin, P., Månér, S., Massa, H., Walker, M., Chi, M., et al. (2004). Large-scale copy number polymorphism in the human genome. *Science*, 305(5683):525–528.

Shi, J., Levinson, D. F., Duan, J., Sanders, A. R., Zheng, Y., Pe'Er, I., Dudbridge, F., Holmans, P. A., Whittemore, A. S., Mowry, B. J., et al. (2009). Common variants on chromosome 6p22. 1 are associated with schizophrenia. *Nature*, 460(7256):753–757.

Shmueli, G. (2010). To explain or to predict? *Statistical science*, pages 289–310.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489.

Slatkin, M. (2008). Linkage disequilibrium—understanding the evolutionary past and mapping the medical future. *Nature Reviews Genetics*, 9(6):477–485.

Smith, E., Weston, C. A., and Lieberman, A. (2009). Schizophrenia (maintenance treatment). *BMJ Clinical Evidence*, 2009.

Snyder, E. M., Philpot, B. D., Huber, K. M., Dong, X., Fallon, J. R., and Bear, M. F. (2001). Internalization of ionotropic glutamate receptors in response to mGluR activation. *Nature neuroscience*, 4(11):1079–1085.

Stefansson, H., Ophoff, R. A., Steinberg, S., Andreassen, O. A., Cichon, S., Rujescu, D., Werge, T., Pietiläinen, O. P., Mors, O., Mortensen, P. B., et al. (2009). Common variants conferring risk of schizophrenia. *Nature*, 460(7256):744–747.

Strobl, C., Boulesteix, A.-L., Kneib, T., Augustin, T., and Zeileis, A. (2008). Conditional variable importance for random forests. *BMC bioinformatics*, 9(1):307.

Stroustrup, B. (1986). *The C++ programming language*. Pearson Education India.

Subramanian, A., Tamayo, P., Mootha, V. K., Mukherjee, S., Ebert, B. L., Gillette, M. A., Paulovich, A., Pomeroy, S. L., Golub, T. R., Lander, E. S., et al. (2005). Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences*, 102(43):15545–15550.

Sullivan, P. F. (2007). Spurious genetic associations. *Biological psychiatry*, 61(10):1121–1126.

Sullivan, P. F., Daly, M. J., and O'Donovan, M. (2012). Genetic architectures of psychiatric disorders: The emerging picture and its implications. *Nature Reviews Genetics*, 13(8):537–551.

Sullivan, P. F., Kendler, K. S., and Neale, M. C. (2003). Schizophrenia as a complex trait: Evidence from a meta-analysis of twin studies. *Archives of general psychiatry*, 60(12):1187–1192.

Szymczak, S., Holzinger, E., Dasgupta, A., Malley, J. D., Molloy, A. M., Mills, J. L., Brody, L. C., Stambolian, D., and Bailey-Wilson, J. E. (2016). r2vim: A new variable selection method for random forests in genome-wide association studies. *BioData mining*, 9(1):7.

Tandon, R., Gaebel, W., Barch, D. M., Bustillo, J., Gur, R. E., Heckers, S., Malaspina, D., Owen, M. J., Schultz, S., Tsuang, M., et al. (2013a). Definition and description of schizophrenia in the DSM-5. *Schizophrenia research*, 150(1):3–10.

Tandon, R., Heckers, S., Bustillo, J., Barch, D. M., Gaebel, W., Gur, R. E., Malaspina, D., Owen, M. J., Schultz, S., Tsuang, M., et al. (2013b). Catatonia in DSM-5. *Schizophrenia research*, 150(1):26–30.

Tandon, R., Keshavan, M. S., and Nasrallah, H. A. (2008a). Schizophrenia,"just the facts" what we know in 2008. 2. Epidemiology and etiology. *Schizophrenia research*, 102(1):1–18.

Tandon, R., Keshavan, M. S., and Nasrallah, H. A. (2008b). Schizophrenia,"just the facts": What we know in 2008: Part 1: Overview. *Schizophrenia research*, 100(1):4–19.

Tandon, R. and Maj, M. (2008). Nosological status and definition of schizophrenia: Some considerations for DSM-V and ICD-11. *Asian Journal of Psychiatry*, 1(2):22–27.

Tandon, R., Nasrallah, H. A., and Keshavan, M. S. (2009). Schizophrenia,"just the facts" 4. Clinical features and conceptualization. *Schizophrenia research*, 110(1):1–23.

Taylor, M. A., Shorter, E., Vaidya, N. A., and Fink, M. (2010). The failure of the schizophrenia concept and the argument for its replacement by hebephrenia: Applying the medical model for disease recognition. *Acta Psychiatrica Scandinavica*, 122(3):173–183.

Tennessen, J. A., Bigham, A. W., O'Connor, T. D., Fu, W., Kenny, E. E., Gravel, S., McGee, S., Do, R., Liu, X., Jun, G., et al. (2012). Evolution and functional impact of rare coding variation from deep sequencing of human exomes. *science*, 337(6090):64–69.

Thomas, D. (2010). Gene–environment-wide association studies: Emerging approaches. *Nature Reviews Genetics*, 11(4):259–272.

Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, 59(236):433–460.

Uppu, S., Krishna, A., and Gopalan, R. (2016). A review of machine learning and statistical approaches for detecting snp interactions in high-dimensional genomic data. *IEEE/ACM transactions on computational biology and bioinformatics*.

Urdinguio, R. G., Sanchez-Mut, J. V., and Esteller, M. (2009). Epigenetic mechanisms in neurological diseases: Genes, syndromes, and therapies. *The Lancet Neurology*, 8(11):1056–1072.

Van Snellenberg, J. X. and de Candia, T. (2009). Meta-analytic evidence for familial coaggregation of schizophrenia and bipolar disorder. *Archives of General Psychiatry*, 66(7):748–755.

Vapnik, V. N. and Chervonenkis, A. Y. (1982). Necessary and sufficient conditions for the uniform convergence of means to their expectations. *Theory of Probability & Its Applications*, 26(3):532–553.

Velez, D. R., White, B. C., Motsinger, A. A., Bush, W. S., Ritchie, M. D., Williams, S. M., and Moore, J. H. (2007). A balanced accuracy function for epistasis modeling in imbalanced datasets using multifactor dimensionality reduction. *Genetic epidemiology*, 31(4):306–315.

Venkataraman, S., Yang, Z., Davies Liu, E. L., Falaki, H., Meng, X., Xin, R., Ghodsi, A., Franklin, M., Stoica, I., and Zaharia, M. SparkR: Scaling R programs with Spark.

Verhelj, C., Bakker, C., de Graaff, E., Keulemans, J. L., Willemsen, R., Verkerk, A., Galjaard, H., Reuser, A., Hoogeveen, A., and Oostra, B. (1993). Characterization and localization of the FMR-1 gene product associated with fragile X syndrome. *Nature*, 363(6431):722–724.

Walker, F. O. (2007). Huntington's disease. *The Lancet*, 369(9557):218–228.

Walters, K. A., Huang, Y., Azaro, M., Tobin, K., Lehner, T., Brzustowicz, L. M., and Vieland, V. J. (2014). Meta-analysis of repository data: Impact of data regularization on NIMH schizophrenia linkage results. *PloS one*, 9(1):e84696.

Wang, K., Li, M., and Bucan, M. (2007). Pathway-based approaches for analysis of genomewide association studies. *The American Journal of Human Genetics*, 81(6):1278–1283.

Wang, L., Jia, P., Wolfinger, R. D., Chen, X., and Zhao, Z. (2011). Gene set analysis of genome-wide association studies: Methodological issues and perspectives. *Genomics*, 98(1):1–8.

Wang, N., Akey, J. M., Zhang, K., Chakraborty, R., and Jin, L. (2002). Distribution of recombination crossovers and the origin of haplotype blocks: The interplay of population history, recombination, and mutation. *The American Journal of Human Genetics*, 71(5):1227–1234.

Wei, Z., Wang, K., Qu, H.-Q., Zhang, H., Bradfield, J., Kim, C., Frackleton, E., Hou, C., Glessner, J. T., Chiavacci, R., et al. (2009). From disease association to risk assessment: an optimistic view from genome-wide association studies on type 1 diabetes. *PLoS Genet*, 5(10):e1000678.

Weiler, I. J. and Greenough, W. T. (1999). Synaptic synthesis of the Fragile X protein: Possible involvement in synapse maturation and elimination. *American journal of medical genetics*, 83(4):248–252.

Wickham, H. (2009). *ggplot2: Elegant graphics for data analysis*. Springer-Verlag New York.

Wray, N. R. and Gottesman, I. I. (2012). Using summary data from the danish national registers to estimate heritabilities for schizophrenia, bipolar disorder, and major depressive disorder. *Frontiers in genetics*, 3:118.

Wright, F. A., Huang, H., Guan, X., Gamiel, K., Jeffries, C., Barry, W. T., de Villena, F. P.-M., Sullivan, P. F., Wilhelmsen, K. C., and Zou, F. (2007). Simulating association studies: A data-based resampling method for candidate regions or whole genome scans. *Bioinformatics*, 23(19):2581–2588.

Wright, M. N. and Ziegler, A. (2015). ranger: A fast implementation of random forests for high dimensional data in c++ and r. *arXiv preprint arXiv:1508.04409*.

Xi, M., Sun, J., Liu, L., Fan, F., and Wu, X. (2016). Cancer feature selection and classification using a binary quantum-behaved particle swarm optimization and support vector machine. *Computational and Mathematical Methods in Medicine*, 2016.

Xu, B., Ionita-Laza, I., Roos, J. L., Boone, B., Woodrick, S., Sun, Y., Levy, S., Gogos, J. A., and Karayiorgou, M. (2012). De novo gene mutations highlight patterns of genetic and neural complexity in schizophrenia. *Nature genetics*, 44(12):1365–1369.

Zou, Z., Liu, C., Che, C., and Huang, H. (2014). Clinical genetics of Alzheimer's disease. *BioMed research international*, 2014.

Zuk, O., Hechter, E., Sunyaev, S. R., and Lander, E. S. (2012). The mystery of missing heritability: Genetic interactions create phantom heritability. *Proceedings of the National Academy of Sciences*, 109(4):1193–1198.