Use of the LASSO in single and multi-cohort genome-wide association studies

Thesis submitted for the degree of
Doctor of Philosophy at the
University of Leicester

Virag Patel

Department of Cardiovascular Sciences

University of Leicester

December 2017

Abstract

Over the past decade, there has been an ever growing interest in genome-wide association studies (GWAS). The role of GWAS is to discover associations between genetic variants; commonly Single Nucleotide Polymorphisms (SNPs) and complex diseases. Due to the ever increasing number of SNPs in GWAS, the commonly used association analyses tend to be univariate models rather than multivariate models. These methods are therefore unable to account for the correlation between SNPs; known as Linkage Disequilibrium (LD).

Penalised regression methods have been suggested as an alternative method in GWAS, specifically the Least Absolute Shrinkage and Selection Operator (LASSO). This method has the ability to both shrink regression coefficients and perform variable selection. In this thesis, the use of the LASSO in both single and multi-cohort GWAS is examined. In the context of the single cohort, the LASSO is applied to the GRAPHIC study in an attempt to discover novel associations with Low-density Lipoprotein. This thesis will also address some of the problems with the LASSO such the tuning parameter selection method that should be used for SNP selection and the need for pruning to reduce the dimensionality of the data in order to fit LASSO models. The literature suggests that a pruning or pre-screening method is required to fit LASSO models in GWAS due to the high computational burden of fitting such a model, yet there is little work to address how the dataset should be pruned. A SNP pruning package in R called prune is developed and is utilised in a simulation study to determine which pruning method should be used. The role of the LASSO in multi-cohort studies is also considered specifically in integrative analyses. A new penalised regression method, the Integrative LASSO, is proposed and developed which uses a combination of LASSO, ridge regression and fused LASSO penalties and tested against some of the current methods in the literature in a simulation study.

Acknowledgements

First and foremost I would like to thank my supervisors Professor John Thompson and Dr. Chris Nelson for their help and support throughout my PhD. I would also like to thank all the wonderful people in both the Cardiovascular Sciences Department and Health Sciences Department and in particular those close to me including Dr. Doly Aravani, Dr. Rhiannon Owen and Dr. Shaun Barber.

I would also like to dedicate this thesis to my parents who have always supported and shown faith in me. I would finally like to thank those who I play football with on a Wednesday, Thursday and Sunday and also the members of Mayflower cricket club. Sport has always been my stress relief and it would not have been possible without these people.

Table of Contents

List of Tables

List of Figures

1 Introduction

1.1 Background and aim

Since the completion of the human genome project (1,2) there has been an ever increasing interest in genome-wide association studies (GWAS). The role of genome-wide association studies is to identify associations between genetic variants, known as Single Nucleotide Polymorphisms (SNPs), and a complex disease (phenotype) such as Type 2 diabetes (T2D), cardiovascular disease (CVD) or different forms of cancer using a sample from the population. This is done by performing an association test between phenotype and all genotyped SNPs. An association between a SNP and a disease could be due to a number of reasons which include:

a. The SNP is truly associated with the phenotype and plays an important role in the cause or prevention of the disease (a true positive).

b. The SNP is associated with the phenotype as it highly correlated (known as Linkage Disequilibrium (LD)) with a second SNP, and this second SNP is truly associated with the disease.

c. The correlation between the SNP and phenotype is by chance and there is no real association (a false positive).

In a perfect world, only the truly causal variables (a) would be identified, although selecting the SNP in high Linkage Disequilibrium with the associated SNP (b) may also be acceptable. In reality, it is difficult to distinguish between (a) and (b) and both are deemed to be true associations. By identifying the truly associated genetic variants or regions (loci), scientists

could be able to develop new therapies, improve diagnosis and better disease prevention (3).

Since the first published GWAS in 2005 (4), the number of GWA studies has grown almost exponentially. As of December 2017, the NHGRI-EBI Catalogue of published genome-wide association studies has recorded 3,211 published studies (5). In this time, the number of SNPs in a GWAS has increased with millions of SNPs being tested for associations but also an increase in sample size. This leads to some difficulty in selecting the truly causal SNPs. The increase in sample size increases the power of a study. The power is defined as the probability of selecting an associated SNP based on a sample that has a true association in the population (i.e. a true positive). The increase in the number of SNPs available for association testing will naturally provide a better coverage of the genome and potentially allows a greater number of truly associated SNPs to be tested. The typing of millions of SNPs leads to difficultly in multivariate modelling (6) which has led to simplifying analyses and estimating each variable in a univariate model rather than a multivariate model.

The occurrence of multiple testing in GWAS may also lead to a number of false positives being selected (c). Traditional variable selection methods such as Bonferroni correction and false discovery rate (FDR) (7,8) are commonly utilised in GWAS in order to control the type 1 error rate (i.e. control the number of false positives selected). These methods are however unable to account for LD between SNPs (b).

Penalised regression methods have been suggested as an alternative, specifically for model selection in scenarios when the number of variables (*P*) is greater than the sample size (*N*) which tends to be the case in GWAS. These methods apply a penalty to regression estimates in order to shrink the estimates. In particular, the Least Absolute Shrinkage and Selection Operator (LASSO) first proposed by Tibshirani for high-dimensional data (9) has received much attention, particularly in genetics. This is due to the LASSO's ability to perform variable selection by penalizing effect estimates to 0 and thus removing the variable from the model.

The LASSO model is on a multivariate level and therefore jointly models all variables and uses computationally fast algorithms.

There are three main aims for this thesis:

1. The first aim of this thesis is to use the LASSO to find genetic associations with Low-density Lipoprotein (LDL-c) in the Genetic Regulation of Arterial Pressure of Humans in the Community (GRAPHIC) Study (10). The results of this analysis can be compared with analyses performed using Bonferroni correction, FDR and the current literature.

2. As suggested by the literature (11-16), there is a great computational burden placed on this analysis. Therefore the second aim of this thesis is to determine the best methods to reduce the dimensionality of the dataset such that the impact on variable selection is minimised.

3. In recent years, the focus has shifted from GWAS on single datasets to consortia combining multiple datasets; however there has been little research into this area in the context of the LASSO, specifically in GWAS. With this in mind, the third aim of this thesis is to compare the current penalised regression methods for integrative analysis by a simulation study. I also aim to present and test an alternative approach for integrative analysis.

1.2 Outline of the thesis

I begin with a statistical overview of the LASSO in Chapter 2; this will include statistical backgrounds of other generalisations of the LASSO such as ridge regression (17) and the

elastic net (18). In this chapter, I will also review the literature on algorithms used to fit the LASSO, tuning parameter selection methods and the application of penalised regression methods in GWAS. In Chapter 3, I show my own implementation of the LASSO using the coordinate descent algorithm (CDA) which was reviewed in Chapter 2. Additionally, I run a simulation study to determine which tuning parameter selection methods show the best performance for variable selection in a GWAS setting. Both the algorithm and a number of the tuning parameter selection methods are used later in this thesis.

One of the aims of this thesis is to apply the LASSO on a real dataset; the GRAPHIC study with Low-density Lipoprotein (LDL-c) as the phenotype of interest. This will be performed in Chapter 4. The chapter will include a literature review of previous studies that have performed GWAS on LDL to identify previously known associations. Both the Bonferroni correction method and FDR will also be used as comparisons.

Due to the computational intensity of the LASSO on the GWAS dataset, in Chapter 4, I aim to explore the use of SNP pruning by Linkage Disequilibrium to reduce the dimensionality of the dataset in order to fit a LASSO model. In Chapter 5, I describe the biological background to LD and how LD is estimated from both haplotypes and genotypes. I then compare a number of packages that estimate LD from genotypes.

In Chapter 6, I introduce my own R package called prune, which prune datasets in a variety of ways including by LD, by P-value and by LD clumping. The Prune package gives the user a great number of options which most pruning packages do not. I will also compare my Prune package and the LD pruning method used in PLINK (19,20).

In Chapter 7, I will use the Prune package and the pruning methods available in a simulation study investigating the effects of pruning on variable selection using the LASSO. The aim of the simulation study is to determine which pruning method and tuning parameter selection method performs best for variable selection.

I then return to the GRAPHIC study in Chapter 8 and apply the best combination of pruning method and tuning parameter selection method based on the simulation study in Chapter 7.

In Chapter 9, I turn my attention to the use of the LASSO in integrative analysis. I begin by reviewing the current literature that apply penalised regression methods in an integrative analysis setting. I will then run a simulation study comparing some of these methods in a GWAS setting to determine which method performs the best in terms of variable selection.

I will then offer my own alternative method, the Integrative LASSO (IL) in Chapter 10. I will explain the reasoning behind the method and provide an algorithm to fit IL models. I will illustrate an example of the IL and finally compare the IL in a simulation study against competing methods from Chapter 9.

Finally, in Chapter 11, I conclude by reviewing my findings and discussing future work.

2 Background to the LASSO

2.1 Introduction

In this chapter, I begin by introducing the statistical background to the LASSO as well as other generalisations of the LASSO and then discuss the merits and faults of each method.

Most regression functions use simple algorithms to optimise the function as they are mostly smooth and convex. The LASSO function however is non-smooth, due to the penalty, which creates some challenges. I therefore explore a number algorithms used to fit the LASSO and other penalised regression models (section 2.4). I begin by reviewing the most popular algorithms used to fit LASSO models. From the current methods, I select the most effective algorithm to write in R as a foundation for future work and show algebraically how the solution to a number of penalised regression functions is derived using the selected algorithm.

Selection of this tuning parameter is important as it is solely responsible for which variables are selected and which are removed from the model. In section 2.5, I review a number of current tuning parameter selection methods. I also discuss the methodology used when there are dual penalties.

2.2 The LASSO

Penalised regression methods attempt to minimise a function consisting of a loss function (such as ordinary least squares, logit etc.) and at least one penalty term with a tuning parameter (λ). The LASSO, as defined by Tibshirani (9), minimses the following function:

$$\hat{\beta}_{LASSO}(\lambda) = \min_{\beta}\left(L(y, x; \mu, \beta) + \lambda \sum_{j=1}^{P}|\beta_j|\right) \quad (2.1)$$

Where N is the number of subjects, P is the number of predictor variables (SNPs in GWAS studies), y is a vector of N outcome variables (known as a phenotype), x is a standardised N x P matrix of predictor variables, β is a vector of effect estimates and λ is the tuning parameter. As with all datasets, standardisation is required in order to scale the dataset correctly. The intercept is denoted by μ. $L(y, x; \mu, \beta)$ represents any loss function. For this thesis, an Ordinary Least Squares link function is used which would minimise (2.2).

$$\hat{\beta}_{LASSO}(\lambda) = \min_{\beta}\left(\frac{1}{2N}\sum_{i=1}^{N}\left(y_i - \mu - \sum_{j=1}^{P} x_{ij}\beta_j\right)^2 + \lambda \sum_{j=1}^{P}|\beta_j|\right) \quad (2.2)$$

For some t ≥ 0, equation (2.2) can be alternatively written as:

$$\hat{\beta}_{LASSO}(\lambda) = \min_{\beta}\left(\frac{1}{2N}\sum_{i=1}^{N}\left(y_i - \mu - \sum_{j=1}^{P} x_{ij}\beta_j\right)^2\right) s.t. \sum_{j=1}^{P}|\beta_j| < t \quad (2.3)$$

The LASSO performs variable selection by shrinking β estimates towards 0. The amount of shrinkage is controlled by the tuning parameter λ. If the tuning parameter is large enough for some variables, its β_j will be forced to 0, removing this variable from the model.

An example of this shrinkage is shown graphically in Figure 2.1 where 10 independent continuous variables are simulated each with a sample size of 100. Each variable was simulated from a normal distribution with mean 0 and standard deviation (S.D.) 1. The outcome variable was simulated such that:

$$y_i = 0.1X_1 + 0.2X_2 + 0.3X_3 + 0.4X_4 + 0.5X_5 + 0.6X_6 + 0.7X_7 + 0.8X_8 + 0.9X_9 + \varepsilon_i$$

Where the error term $\varepsilon_i \sim N(0,1)$. Each line on the plot represents the β coefficient for any penalty (λ). The coefficient path plot shows that as the penalty, increases on the x-axis, the β coefficients shrink towards 0 and when they reach exactly 0 the variable is removed from the model. The LASSO is able to both select variables and estimate β coefficients at the same time; however the β estimates will be biased for some λ > 0.

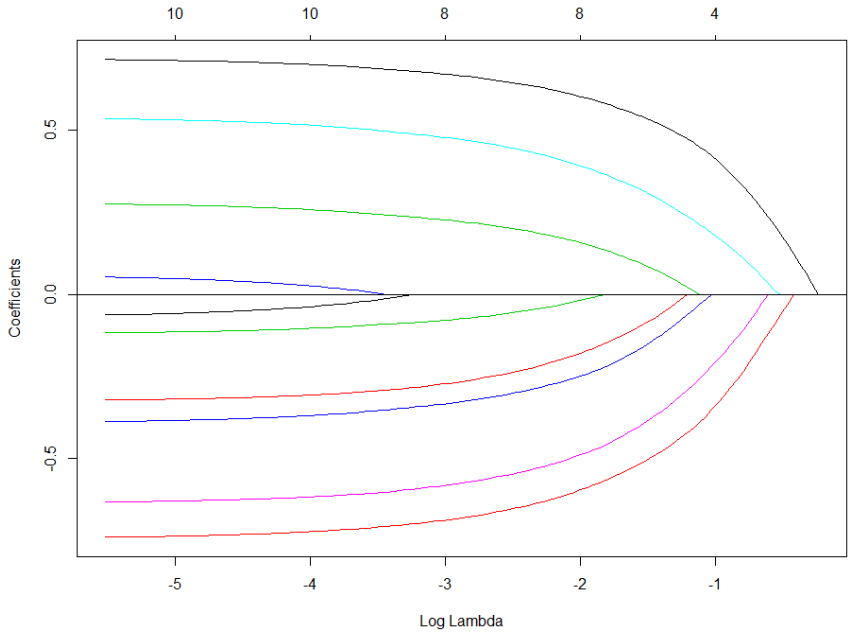


Figure 2.1 Coefficient path plot showing the shrinkage of ten variables as the tuning parameter increases. Each line represents a variable and the path shows the β coefficient on the y-axis as the penalty (on a $log(\lambda)$ scale) increases on the bottom x-axis. The top x-axis shows the number of variables remaining in the model at each $log(\lambda)$ penalty value.

There are two main reasons why the inclusion of the penalty is particularly desirable compared to a least squares model. The first is prediction accuracy and the second is model interpretation (9). In a high-dimensional dataset, a regression analysis produces estimates that have high variance and low bias. By penalising estimates, the variance is reduced and the bias will increase which increases the prediction accuracy (9,21,22). For variable selection the model interpretability is of greater importance. The aim is to select a smaller

subset of variables that is best able to explain the variance in the outcome variable. By reducing the number of variables in the model, it becomes more interpretable. Variable selection methods such as stepwise regression tend to lead to unstable models which the LASSO does not (9,23).

There are two main criticisms of the LASSO that were outlined by Zou and Hastie (18). The first is when $P > N$, then the LASSO is only able to select at most *N* variables for a model. This is less of a concern in GWAS as the number of truly associated SNPs is small compared to the ever increasing sample size of a GWA study. The second criticism of the LASSO is the inability to handle correlated data with Zou and Hastie stating that in a group of highly correlated variables, the LASSO tends to select one variable without regard for which variable is selected (18). This would be a concern in GWA studies, as SNPs in close proximity to each other tend to be highly correlated. The correlation between SNPs is known as Linkage Disequilibrium (LD) and is discussed in greater detail in Chapter 5. This is contradictory to the literature which suggests that the LASSO is able to handle the LD between SNPs (24-26).

2.3 Generalisations of the LASSO

2.3.1 Ridge regression and the elastic net

Another popular penalised regression method is ridge regression (RR) first proposed by Hoerl and Kennard in 1970 (17). Ridge regression was designed to account for correlation between variables in regression and minimises the following function:

$$\hat{\beta}_{RR}(\lambda) = \min_{\beta}\left(\frac{1}{2N}\sum_{i=1}^{N}\left(y_i - \mu - \sum_{j=1}^{P} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{P} \beta_j^2 \right) \qquad (2.4)$$

The penalty for RR is slightly different to the LASSO, while the LASSO penalty penalises the sum of the absolute value of the β estimates, ridge regression penalises the sum of the squared β estimates. The right-hand plot in Figure 2.2 shows the coefficient path plot for ridge regression. Like the LASSO, RR penalises towards 0. Unlike the LASSO however RR is unable to force variables to exactly 0 and therefore all variables remain in the model. This is shown by the top x-axis in the coefficient path plot, which displays the number of variables remaining in the model for a certain λ, which remains at 10. This means that RR is unable to perform variable selection. When comparing the coefficient path plots of the LASSO and RR in Figure 2.2, it can be seen that the LASSO tends to shrink smaller β coefficients more heavily, whereas RR penalises the larger coefficients more heavily. The RR penalty tends to shrink correlated variables towards each other creating a "grouping effect"(27).

The difference in penalties is shown in Figure 2.3 for two variables which are plotted on the x and y-axis. The point at $\hat{\beta}$ represents the OLS estimates for the two variables. The ellipses show the contours of the residual sum of squares (RSS) as the function moves away from the minimum ($\hat{\beta}$). The solid regions centred around (0, 0) are the LASSO and RR penalty constraints respectively. The size of this constraint is determined by the size of the tuning parameter λ. The penalised regression coefficient estimates for each method is at the point where the contour touches the penalty. For the LASSO, there is a greater chance that the contour would meet the penalty at a corner due to its diamond shape, and at any of these points the coefficient estimate for one of the two variables would be 0. Which coefficient is estimated as 0 would depend in which corner meets the contour. This is less likely to be the case for RR due to the circular shape of the penalty constraint.

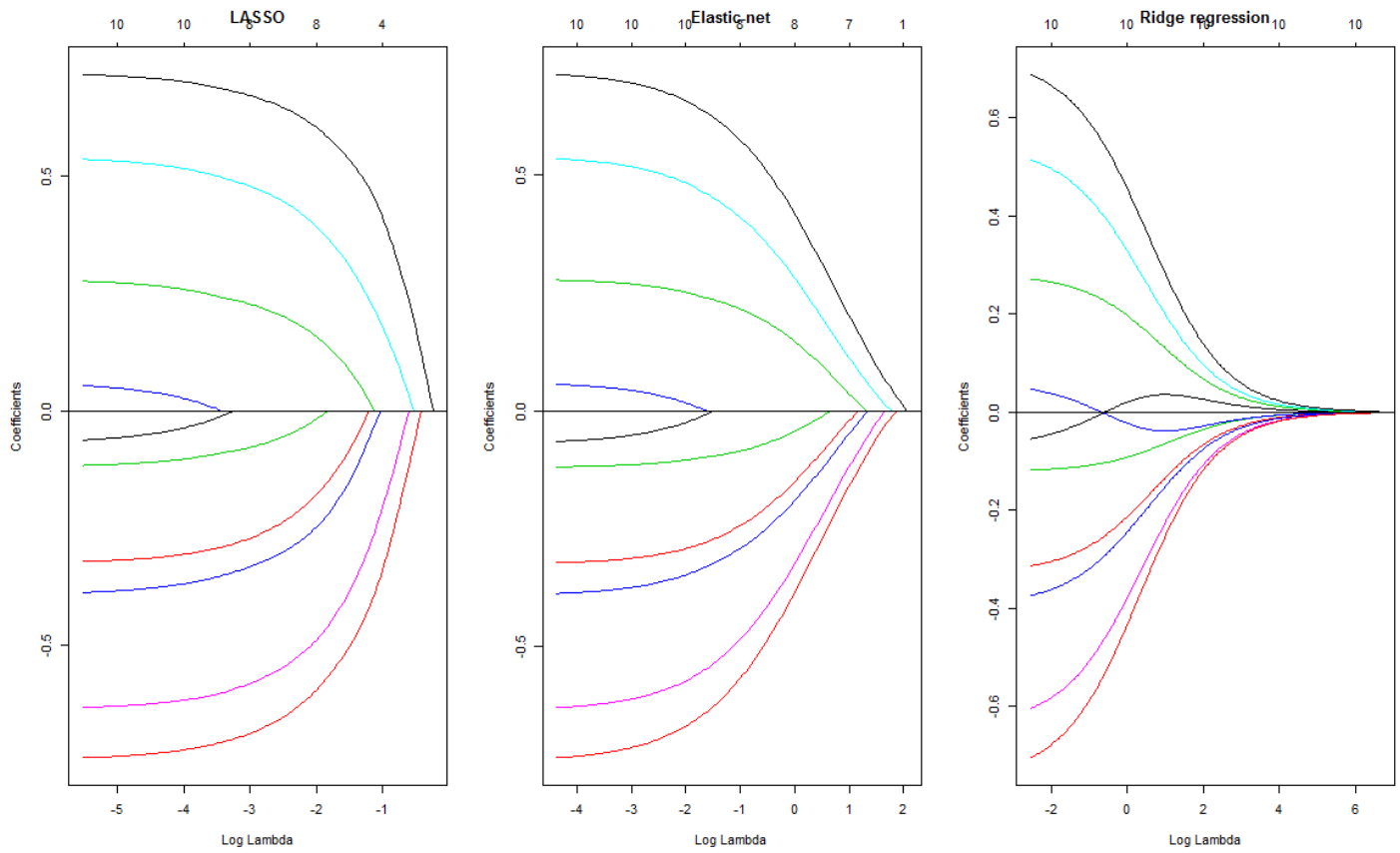


Figure 2.2 Coefficient path plot showing the shrinkage of ten variables as the tuning parameter increases for the LASSO (left), elastic net with α = 0.9 (centre) and ridge regression (right). Each line represents a variable and the path shows the β coefficient on the y-axis as the penalty (on a $log(\lambda)$ scale) increases on the bottom x-axis. The top x-axis shows the number of variables remaining in the model at each $log(\lambda)$ penalty value.

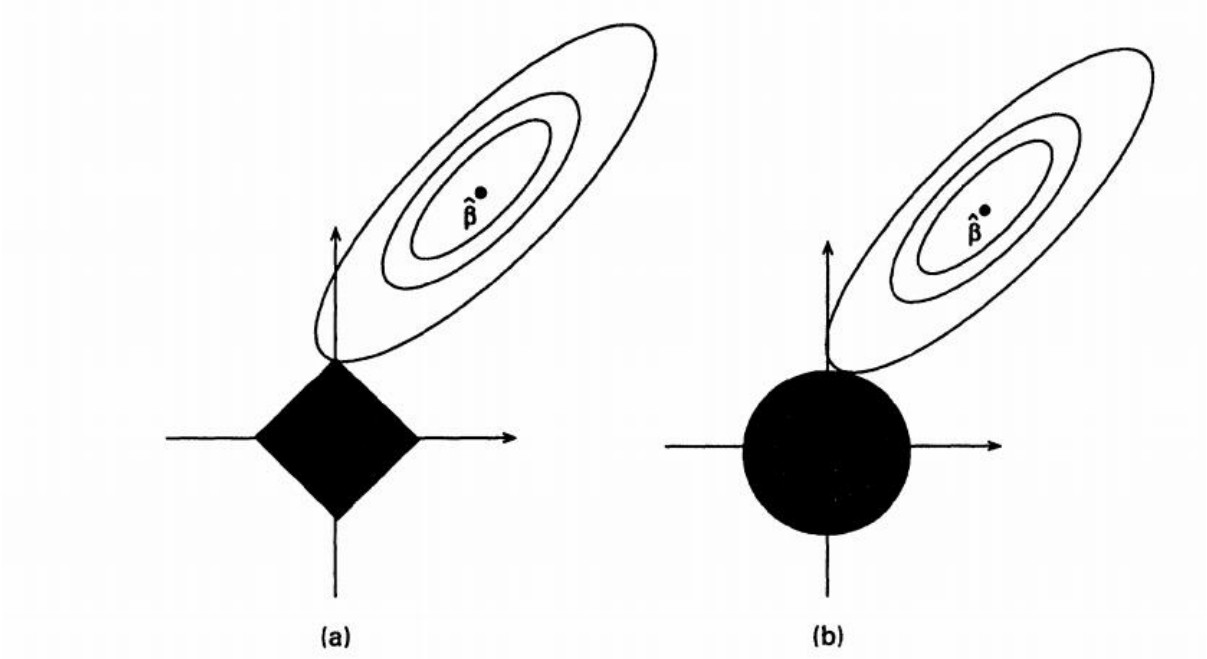


Fig. 2. Estimation picture for (a) the lasso and (b) ridge regression

Figure 2.3 Two-dimensional contour plots of the LASSO and ridge regression. Taken from Tibshirani (9).

Zou and Hastie proposed the elastic net (EN) (18) which combines both the LASSO and RR penalties (2.5). This allows the elastic net to both handle correlations and perform variable selection.

$$\hat{\beta}_{EN}(\lambda) = \min_{\beta}\left(\frac{1}{2N}\sum_{i=1}^{N}\left(y_i - \mu - \sum_{j=1}^{P} x_{ij}\beta_j\right)^2 + (\alpha-1)\lambda\sum_{j=1}^{P}|\beta_j| + \alpha\lambda\sum_{j=1}^{P}\beta_j^2\right) \quad (2.5)$$

An α term is introduced alongside the tuning parameter λ which allows the choice of varying the relative strength of each penalty. α can take any value between 0 and 1. An $\alpha = 0$ gives a LASSO model for λ, and $\alpha = 1$ produces a RR model and any α

between 0 and 1 will result in a model using both penalties. The elastic net allows both variable selection and the ability to handle correlated data. Zou and Hastie showed that in both a simulation study and example dataset, the EN outperforms the LASSO in terms of model prediction. This was mainly due to the RR's penalty being able to group correlated variables together and penalise these groups together (Figure 2.4). However this also meant that EN would select more variables in the final model compared to the LASSO which was also shown by Waldmann *et al*. in a simulation on GWAS data (27).

The increase in the number of variables selected may be detrimental in GWAS as there are a large number of correlated SNPs in a dataset. Selecting one highly associated SNP could lead to a number of correlated SNPs also being selected. In reality, it is likely that only one in a group of correlated SNPs may be truly associated. As the EN uses both LASSO and RR penalties, it's unsurprising that the shape of the penalty is somewhere between these two penalties (Figure 2.5) and its shape will be influenced by α.

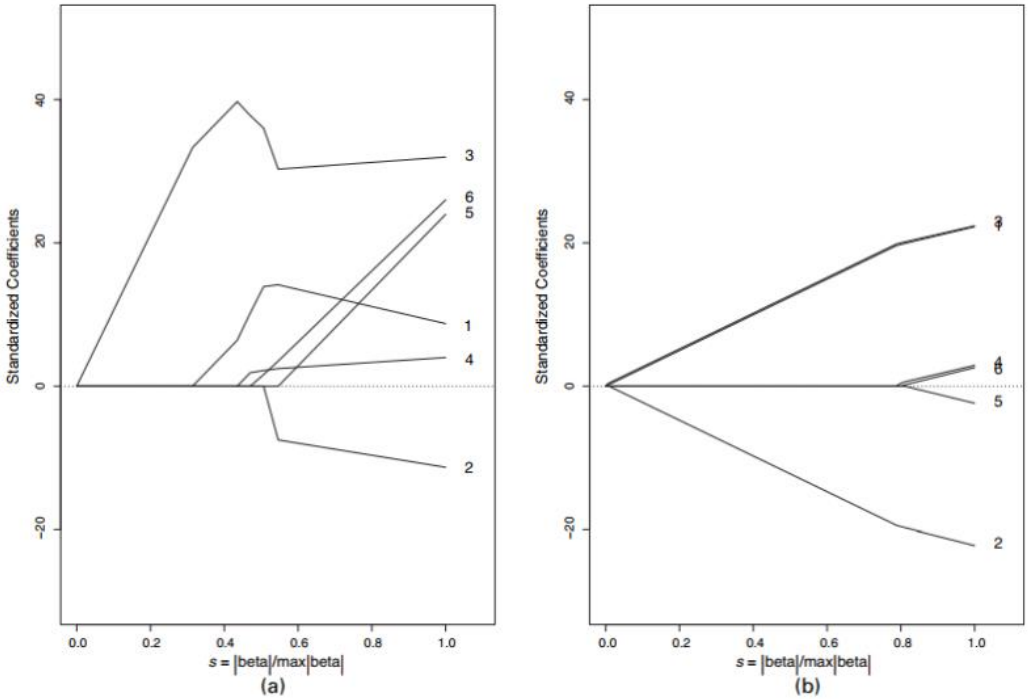


Fig. 5. (a) Lasso and (b) elastic net ($\lambda_2 = 0.5$) solution paths: the lasso paths are unstable and (a) does not reveal any correction information by itself; in contrast, the elastic net has much smoother solution paths, while clearly showing the 'grouped selection'—$\mathbf{x}_1$, $\mathbf{x}_2$ and $\mathbf{x}_3$ are in one 'significant' group and $\mathbf{x}_4$, $\mathbf{x}_5$ and $\mathbf{x}_6$ are in the other 'trivial' group; the decorrelation yields the grouping effect and stabilizes the lasso solution

Figure 2.4 The difference in coefficient path plots between the LASSO and elastic net in correlated data. Taken from Zou and Hastie (18).

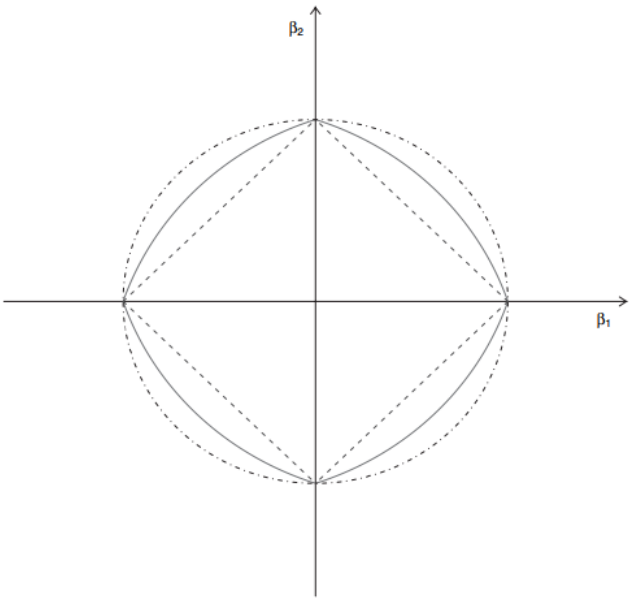


Fig. 1. Two-dimensional contour plots (level 1) (· - - - - -, shape of the ridge penalty; -------, contour of the lasso penalty; ———, contour of the elastic net penalty with $\alpha = 0.5$): we see that singularities at the vertices and the edges are strictly convex; the strength of convexity varies with α

Figure 2.5 Two-dimensional contour plots of the LASSO, elastic net and ridge regression. Taken from Zou and Hastie (18)

2.3.2 Bridge regression

The LASSO and RR are two special cases of a family of penalised regression functions known as Bridge regression (28) which minimises the following general function:

$$\hat{\beta}_{Bridge}(\lambda) = \min_{\beta}\left(\frac{1}{2N}\sum_{i=1}^{N}\left(y_i - \mu - \sum_{j=1}^{P} x_{ij}\beta_j\right)^2 + \lambda\sum_{j=1}^{P}|\beta_j|^{\gamma}\right) \qquad (2.6)$$

Any $\gamma > 0$ defines the type of penalty used. At $\gamma = 1$, LASSO function is produced (2.2) and $\gamma = 2$ produces a RR function (2.4). Variable selection can be performed for any bridge penalty $0 < \gamma \leq 1$. Figure 2.6 shows the contour plots for varying bridge regression penalties. For $\gamma \leq 1$, the contours are highly likely to intersect the penalty at a corner, which would penalise some variables to 0. For $\gamma > 2$, the circular penalty

from the ridge regression tends towards square shape, which is also unable to perform variable selection.

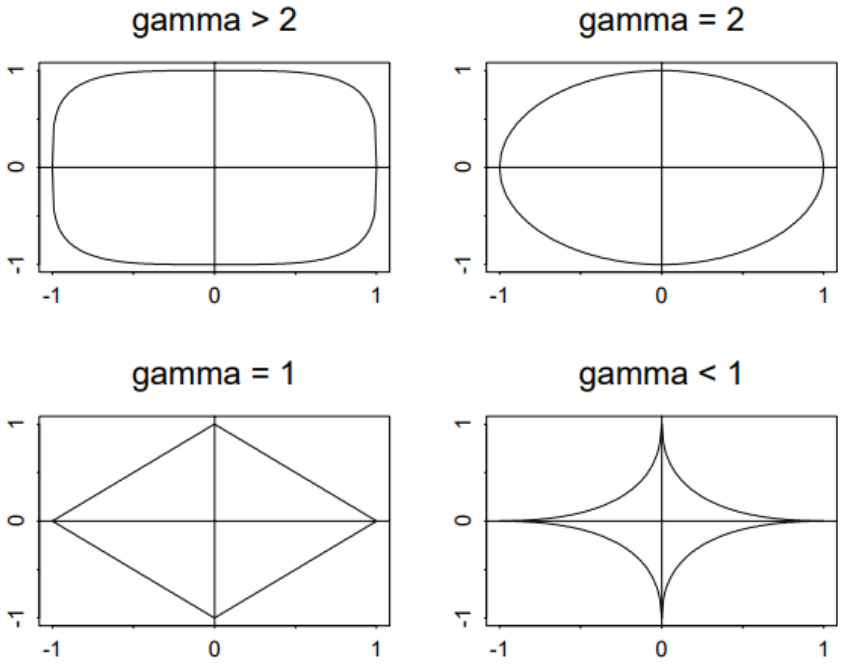


Figure 1. Constrained Areas of Bridge Regressions with t = 1.

Figure 2.6 Two-dimensional contour plots for varying Bridge regression penalties. Taken from Fu (28)

2.3.3 The group LASSO

Yuan and Lin (29) first proposed the group LASSO as a method to group desired variables together within a single dataset, shown in (2.7), where g denotes the pre-defined groups of variables. The penalty on each group is weighted by the square-root of the number of variables in that group (ρ_g), therefore for any group consisting of a single variable would be penalised in the same way as the LASSO. Groups tend to consist of variables that are correlated with each other, although this may not have been the intended design (30).

$$\hat{\beta}_{grpLASSO}(\lambda) = \min_{\beta}\left(\frac{1}{2N}\sum_{i=1}^{N}\left(y_i - \sum_{g=1}^{G} x_{ig}\beta_g\right)^2 + \lambda\sum_{g=1}^{G}\sqrt{\rho_g}\left\|\beta_g\right\|_2\right) \quad (2.7)$$

where,

$g = \{1, \dots G\}$ is a set of predefined groups from the j variables

$$\left\|\beta_g\right\|_2 = \sqrt{\sum_{j\in G, j=1}^{P}\beta_j^2}$$

ρ_g = the number of variables in group g

The group LASSO is unable to perform variable selection within groups; therefore either all variables in a group are selected or are removed from the model using the group LASSO. Friedman *et al.* proposed the sparse group LASSO which contains two penalties and allows variables to be penalised within and across groups (31) (2.8). The function is similar to that of the group LASSO (2.7); however there is no weight on group penalty and a LASSO penalty on each individual variable is included, which penalises variables independent of its grouping. As the group LASSO and sparse group LASSO use the same penalties as RR and EN respectively, it's unsurprising that the penalty takes a similar shape on a contour plot (Figure 2.7).

$$\hat{\beta}_{sgrpLASSO}(\lambda) = \min_{\beta}\left(\frac{1}{2N}\sum_{i=1}^{N}\left(y_i - \sum_{g=1}^{G} x_{ig}\beta_g\right)^2 + \lambda_1\sum_{g=1}^{G}\left\|\beta_g\right\|_2 + \lambda_2\sum_{j=1}^{P}\left|\beta_j\right|\right) \quad (2.8)$$

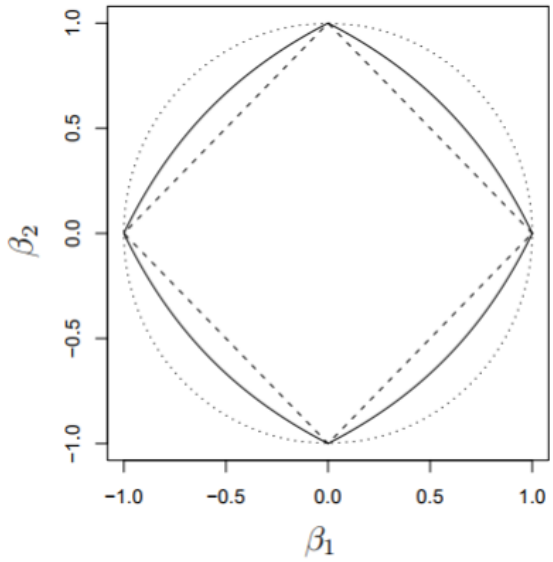


Figure 1: *Contour lines for the penalty for the group lasso (dotted), lasso (dashed) and sparse group lasso penalty (solid), for a single group with two predictors.*

Figure 2.7 Two-dimensional contour plots of the LASSO, sparse group LASSO and group LASSO. Taken from Friedman *et al*. (31)

2.3.4 The fused LASSO

The fused LASSO first proposed by Tibshirani *et al*. (32), is mainly designed for data that can be ordered in some fashion, and the ordering leads to some potential correlation, for example, SNPs in the genome are ordered along a chromosome with the correlation being LD between SNPs, especially those that are close to each other. The fused LASSO minimises the following function:

$$\hat{\beta}_{fused}(\lambda) = \min_{\beta}\left(\frac{1}{2N}\sum_{i=1}^{N}\left(y_i - \mu - \sum_{j=1}^{P} x_{ij}\beta_j\right)^2 + \lambda_1\sum_{j=1}^{P}|\beta_j| + \lambda_2\sum_{j=2}^{P}|\beta_j - \beta_{j-1}|\right) \quad (2.9)$$

There are two penalties incorporated for the fused LASSO, the first is the LASSO penalty on each individual variable which shrinks estimates towards 0. The second is a LASSO penalty on the difference in β estimates between adjacent variables in the ordered dataset. This penalty shrinks adjacent β estimates towards each other. Figure 2.8 which showed a simulated example of the fusion penalty taken from Tibshirani *et al*. (32). In each plot the black lines represent the true simulated β and the red scatter points represent the estimated β for each predictor in four scenarios; OLS (a), the LASSO (b), OLS with a fusion penalty but no LASSO penalty (c) and the fused LASSO (d). By penalising adjacent variables towards each other, the fusion methods were able to estimate the true β better than the LASSO and OLS. The fused LASSO also selects more variables than the LASSO in this case, due to the fusion penalty penalising adjacent variables into the model. This is also shown in the contour plot (Figure 2.9) where the shape of the penalty is further restricted to a small section of the LASSO penalty where $\beta_1$and β_2 take similar values.

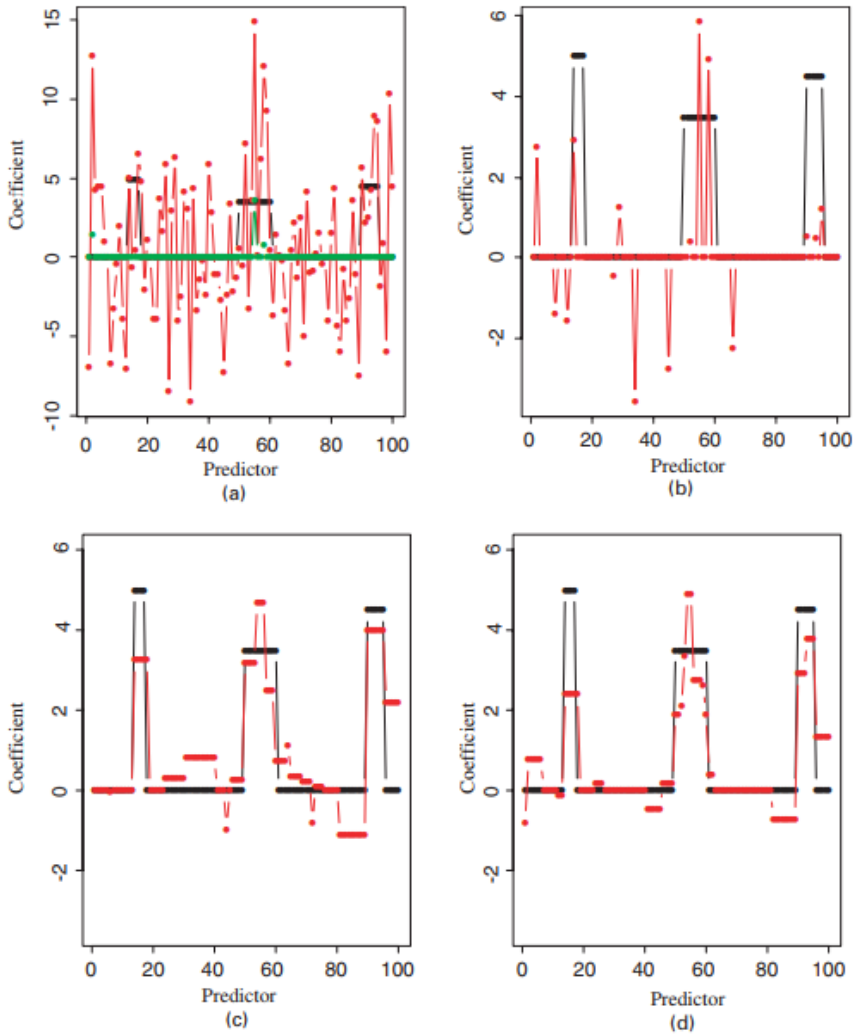


Fig. 3. Simulated example, with $p = 100$ predictors having coefficients shown by the black lines: (a) univariate regression coefficients (red) and a soft thresholded version of them (green); (b) lasso solution (red), using $s_1 = 35.6$ and $s_2 = \infty$; (c) fusion estimate, using $s_1 = \infty$ and $s_2 = 26$ (these values of s_1 and s_2 minimized the estimated test set error); (d) the fused lasso, using $s_1 = \Sigma_j |\beta_j|$ and $s_2 = \Sigma_j |\beta_j - \beta_{j-1}|$, where β is the true set of coefficients

Figure 2.8 Illustration of the fusion penalty (c) and fused LASSO (d) on a simulated dataset compared to OLS (a) and the LASSO (b). Taken from Tibshirani *et al*. (32).

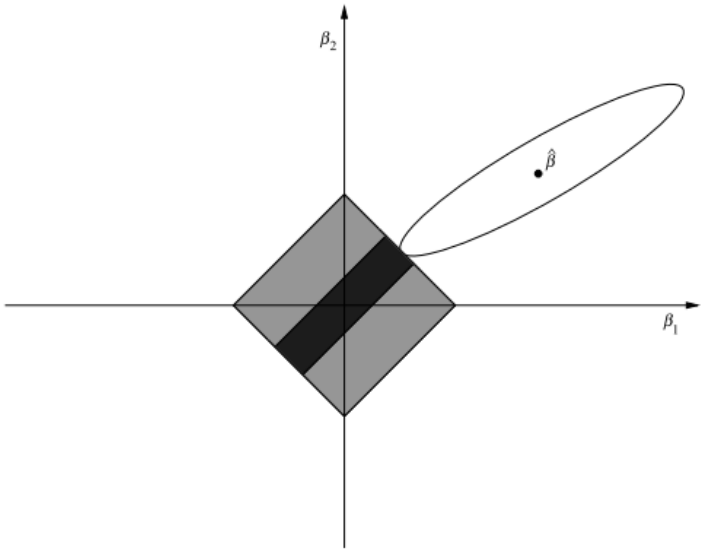


Fig. 2. Schematic diagram of the fused lasso, for the case $N > p = 2$: we seek the first time that the contours of the sum-of-squares loss function (⬭) satisfy $\Sigma_j |\beta_j| = s_1$ (◆) and $\Sigma_j |\beta_j - \beta_{j-1}| = s_2$ (◤)

Figure 2.9 Two-dimensional contour plot of the fused LASSO for two adjacent variables. Taken from Tibshirani *et al*. (32)

2.3.5 The LASSO in meta-analysis

Meta-analysis is a popular method of pooling data from several studies together (33). This is performed by pooling summary statistics from different studies to obtain a single pooled estimate, as the raw data is either not used or unavailable. By pooling a number of studies together there is an increase in the power of the study.

Meta-analyses often take one of two forms of model, fixed-effects and random-effects. A fixed effect model is fitted under the assumption that there is one constant genetic effect across all studies. A fixed-effect model using LASSO regression would involve combining all studies into one large study and then fit a LASSO model. A random effects model assumes that each study has its own genetic effect due to variation between studies, known as heterogeneity (34). These variations between studies can be due to various factors such as genotypic variation between study populations, phenotypic variations between populations due to differences in lifestyle

of differences in study design, recruitment, phenotypic or genotypic measurements. Although set in a psychology setting, Curran and Hussong detail an extensive list of possible sources of heterogeneity, many of which can be applicable in a genetics setting (35).

Meta-analyses pool together estimates, typically either effect estimates or P-values. The effect estimates typically tend to be effect estimates (β) or Odds Ratios (OR). Pooling estimates from LASSO analyses is difficult as the penalty will bias the β or OR estimates. Each dataset when analysed separately will have a different strength penalty applied to it. Even if the same penalty is applied to all datasets, this would not mean the relative strength of the penalty is the same across all datasets, as the strength of the penalty is relative to the size of the β's or ORs. In both cases the bias in the estimates across all datasets would not be consistent. So far there has been no attempt in meta-analysing studies using penalised regression methods from summary statistics or P-values although there have been suggested methods to calculate P-values using the LASSO (36).

He *et al.* have however proposed the Sparse meta-analysis (37). This method calculates regression estimates and applies a multivariate inverse-variance estimator as proposed by Lin and Zeng (38) with a penalty applied on the square-root of the absolute sum of βs across studies. The make-up of this penalty allows for heterogeneity between studies.

Another method of pooling datasets is integrative analysis. This requires the raw individual level data (ILD) for each study to pool together for analysis. This differs from meta-analysis which analyses each dataset individually then pools summary statistics together. The use of the LASSO in integrative analyses is discussed in greater detail in Chapter 9.

2.3.6 Other generalisations of the LASSO

There are a number of other generalisations of the LASSO; some of these are listed by Tibshirani (39). The list includes methods such as the adaptive LASSO (40) which uses a weighted penalty (ω_j) on each variable:

$$\hat{\beta}_{adpLASSO}(\lambda) = \min_{\beta}\left(\frac{1}{2N}\sum_{i=1}^{N}\left(y_i - \mu - \sum_{j=1}^{P} x_{ij}\beta_j\right)^2 + \lambda\sum_{j=1}^{P}\omega_j|\beta_j|\right) \quad (2.10)$$

Where , $\omega_j = \frac{1}{|\hat{\beta}_j|^{\upsilon}}$ and $\upsilon > 0$

The weight for each variable tends to be based on some initial β estimates ($\hat{\beta}_j$) such as OLS or the LASSO. A study by Zou suggests that variable selection methods could be inconsistent (40), for example due to overfitting and including a number of false positives. The adaptive LASSO was designed to overcome this problem. The weights will adjust each β coefficient differently, penalising the variables with a smaller $\hat{\beta}_j$ more heavily.

There have also been a number of Bayesian approaches to penalised regression including the LASSO (41), elastic net (42), group LASSO (43) and fused LASSO (44). These methods are summarised by Liu *et al*.(45)

2.4 Algorithms that fit LASSO models

2.4.1 A review of algorithms that fit LASSO models

The algorithms used to optimize non-smooth convex functions can be subcategorised into three main types; path following (homotopy) methods, first-order methods and alternating direction method of multipliers (ADMM) of which the first order methods seem the most popular (46).

2.4.1.1 First order methods

Tibshirani initially suggested a finite-step (2^P) convergence algorithm (9) using elements of Lawson and Hansen's work on linear least squares problems subject to inequalities (47). It stated that convergence of the model could require up to O(2^P) iterations, where P denotes the number of predictor variables. While the author estimates that most models converge between $0.5P$ and $0.75P$. However in a GWAS setting where P is large, the computational time to fit models would make the LASSO impractical using this finite step convergence algorithm.

Fu later suggested a "shooting algorithm for the LASSO" that was essentially a coordinate descent method (CD), in a study focused on comparing bridge regression with Least Squares regression, the LASSO and ridge regression (28). This was the first study to both suggest and apply a coordinate descent algorithm (CDA) as a method to optimise a form of penalised regression.

Fu compares the "shooting method" to the finite-steps algorithm and concluded that the "shooting method" is simpler to implement and a faster algorithm. It was estimated that it requires approximately $P \log P$ iterations to converge, which is faster

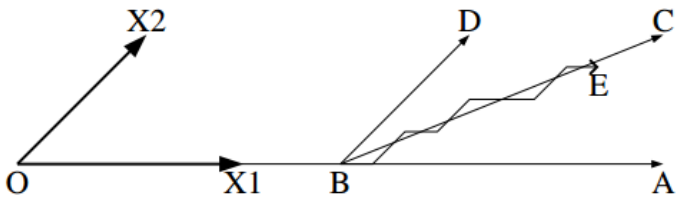
than the finite-steps algorithm. The author does state that this is an estimated result and that a theoretical result had not been obtained.

The coordinate descent algorithm is an optimisation algorithm to 'search' for a minimum of a multivariate function. It is an iterative algorithm which optimises along only one direction (or coordinate) of the multivariate space whilst keeping the remaining variables fixed, rather than attempting to minimise all variables simultaneously. This makes the algorithm simple to implement and computationally faster. CD can be used to fit models for different penalties assuming the loss function is convex, differentiable and the penalty term is both convex and separable (48). Therefore CD is flexible enough to fit a range of other penalised regression models.

Work by Shevade and Keerthi (49) and Daubachies *et al*.(50) suggested and implement CD algorithms in similar work. Thereafter a large quantity of work was done by Friedman, Hastie and Tibshirani in optimising LASSO models using CD (51-53). Coordinate descent was also implemented by the same authors into the popular R package glmnet (53). The algorithm is flexible enough to fit a range of alternative penalised regression models such as elastic net and the grouped LASSO (54).

Least Angle Regression (LARS) is another popular first-order algorithm (55). It is based on the idea of attempting to apply a forward stepwise selection process quickly and efficiently, similar to the Homotopy algorithm. LARS however produces approximate solutions and Homotopy gives exact solutions (56,57). The algorithm starts off with a set of λ, an outcome variable y and a set of predictors x_1, … x_P. Like any forward stepwise procedure, the variable x_{j1} that is the most correlated with y is selected first and enters the model. A step is taken in the direction along x_{j1} towards y until another x_{j2} variable is as correlated as x_{j1}. This first step moves the estimates along the "least angle direction" on a plane. At this point x_{j2} enters the model and a second step is taken, this time along the equiangular bisector between x_{j1} and x_{j2}. This is until a third parameter x_{j3} is as correlated as both x_{j1} and x_{j2}, at which point the next step is taken along the equiangular bisector between x_{j1},x_{j2} and x_{j3}. This is repeated until the last step, which is of the least correlated parameter x_{jp}. A graphical representation

for two predictors is shown in Figure 2.10. The algorithm requires a maximum of $O(P^3 + NP^2)$ computations to calculate the entire sequence of steps for any λ. Hesterberg *et al.* discussed the computational issues of the LARS algorithm suggesting that issues with numerical accuracies may arise in highly correlated data (58), which would be the case in GWA studies. Future studies also showed that the LARS algorithm was computationally slower than CD for a large range of N and P (53).



The LAR algorithm in the case of 2 predictors. O is the prediction based solely on an intercept. $C = \hat{Y} = \hat{\beta}_1 X_1 + \hat{\beta}_2 X_2$ is the ordinary least-squares fit, the projection of Y onto the subspace spanned by X_1 and X_2. A is the forward stepwise fit after one step; the second step proceeds to C. Stagewise takes a number of tiny steps from O to B, then takes steps alternating between the X_1 and X_2 directions, eventually reaching E; if allowed to continue it would reach C. LAR jumps from O to B in one step, where B is the point such that BC bisects the angle ABD. At the second step it jumps to C. LASSO follows a path from O to B, then from B to C. Here LAR agrees with LASSO and stagewise (as the step size $\to 0$ for stagewise). In higher dimensions additional conditions are needed for exact agreement to hold.

Figure 2.10 The LARS algorithm in the case of 2 predictors taken from Hesterberg *et al*.(58)

2.4.1.2 Homotopy

Osbourne *et al*. suggested a Homotopy algorithm to fit LASSO models as well as CD (59). The algorithm starts with an empty set of coefficients and a sufficiently large λ. The process slowly decreases λ until a 'break point' is reached. At this point the solution has changed and coefficients are included in the model by iteratively adding and deleting non-zero coefficients until convergence is met (56,57,60). Due to the algorithm starting with a large λ, it is known as a 'greedy algorithm' and is similar to a forward stepwise procedure as, at each 'break point', the next most correlated

parameter is added. Yang *et al*. compared the average run time and accuracy between five algorithms: Homotopy, primal-dual interior point method (PDIPA), truncated Newton interior-point method (TNIPM/L1LS), Iterative Shrinkage-Thresholding algorithm (ISTA/SpaRSA), Fast Iterative Shrinkage Algorithm (FISTA) and Dual Augmented Lagrange Multiplier (DALM) (57,60). Simulations showed that Homotopy had a slower average run time to convergence than most algorithms and increased linearly as the number of variables increased. Although the results on a real-life dataset (N = 249, P = 20) showed that Homotopy was the most accurate of the algorithms for facial recognition, it was also the fastest algorithm on this dataset when there is little noise (denoted by corruption %) and when P is small. In a GWAS scenario where *P* is large, the algorithm will be computationally slower than the other algorithms and may fail to converge when $P > N$ (61). Taking into account speed and accuracy the authors concluded that there was no clear winner between these methods; PDIPA was the most accurate for noise-free data, while SpaRSA, FISTA and DALM were the most efficient in noisy data.

2.4.1.3 Alternating Direction Method of Multipliers

The same authors conducted a similar study comparing the same algorithms but included accelerated version of parallel coordinate descent, where the user specifies the order of each coordinate update, Primal Augmented Lagrange Multiplier (PALM), approximate message passing (AMP) and Templates for Convex Cones Solvers (TFOCS) algorithms. They concluded that the ALM algorithms performed the best for facial recognition (60). Yang *et al*. discussed the differences between PALM and DALM algorithms noting that the efficiency can be different in real-world applications and running time would depend on the size of the dataset. For GWAS, the most computational intensive step for PALM ($O(n^2)$), would be faster than the most computational intensive step for DALM ($O(p^2 + np)$). The study showed that in this case the ADMM methods (DALM and PALM) were amongst the fastest in terms of computational time to reach an accurate estimate.

The PALM algorithm (also known as ADMM) eliminates the LASSO inequality constraint (2.3) by minimising the augmented Lagranian function (2.11). The solution to Equation (2.11) gives an approximate solution to the LASSO for some λ. Each iteration minimises β and e separately (2.12) (60). The disadvantage of this method is that the Langranian multiplier ξ is chosen and can be inefficient with a poor choice (62) and while the algorithm is guaranteed to converge, it produces approximate solutions rather than exact ones (57). The ability to remove the inequality constraint makes the ADMM algorithm flexible to fit other penalised regression models like the fused LASSO (46,63).

$$\mathcal{L}_{\xi}(\beta, e, \theta) = \sum |\beta| + \lambda \sum |e| + \frac{\xi}{2} \sum (y - \beta x - e)^2 + \theta^T (y - \beta x - e) \tag{2.11}$$

$$\begin{cases} e_{k+1} = \min_{e} \mathcal{L}_{\xi}(\beta_k, e, \theta_k) \\ \beta_{k+1} = \min_{x} \mathcal{L}_{\xi}(\beta, e_{k+1}, \theta_k) \\ \theta_{k+1} = \theta_k + \xi (y - \beta_{k+1} x - e_{k+1}) \end{cases} \tag{2.12}$$

2.4.1.4 Packages that fit the penalised regression models in R

Table 2.1 lists a number of R packages that implement penalised regression models. The descent algorithms, specifically coordinate descent, is clearly the most popular algorithm used for R packages, especially for the LASSO and group LASSO. Only genlasso (64) applies an ALM algorithm. Both glmpath (65) and lasso2 (66) use homotopy path algorithms. Other packages such as elasticnet (67), lol (68), pensim (69), relaxnet (70) and relaxo (71) are not listed as do not specify which algorithms are used.

Table 2.1 List of packages that apply penalised regression models in R

Package	Algorithm	Models options	Additional Information
genlasso (64)	DALM	LASSO, fused LASSO, trend filtering	
gglasso (72)	Blockwise-Majorization-descent	Group LASSO	
glasso (73)	Coordinate descent	Graphical LASSO	
glmmLasso (74)	Gradient descent	LASSO for Generalised linear mixed models	
glmnet (53)	Cyclic coordinate descent	LASSO, ridge regression and elastic net	For GLM and Cox proportional hazard models
glmpath (65)	Homotopy	LASSO	
grplasso (75)	Blockwise coordinate descent	Group LASSO	
grppenalty (76)	Coordinate descent	Group LASSO and group ridge regression	
grpreg (77)	Coordinate descent	Group LASSO	Includes a number of different group penalties
HDPenReg (78)	Least Angle Regression	LASSO, fused LASSO and fusion	
LARS (55)	Least Angle Regression	LASSO, LARS, Forward stagewise, stepwise	
lasso2 (66)	Homotopy	LASSO	
LassoBacktracking (79)	Coordinate descent	LASSO	Package attempts to identify variable interactions
lassoshooting (80)	Cyclic coordinate descent	LASSO	
penalized (81)	Gradient ascent	LASSO, ridge and fused LASSO	For GLM and Cox proportional hazard models
QICD (82)	Coordinate descent	LASSO	For Penalised

			Quantile regression
rqPen (83)	Coordinate descent	LASSO, SCAD and MCP functions and group penalties for each function	For Penalised Quantile regression
SGL (84)	Gradient descent	Group LASSO	
stepPlr (85)	Iterative reweighted ridge regressions (IRRR), a first-order algorithm, similar to Newton Raphson methods	Logistic ridge regression	Package attempts to identify variable interactions

2.4.1.5 Conclusion

There are four popular algorithms for non-smooth convex functions: Coordinate descent, Homotopy, ALM and LARS. Of the four, CD seems the most flexible and easiest to implement as an algorithm. Yang *et al*. showed that the ALM algorithms were computationally faster than CD (60), however this was in a scenario where $N > P$ and P was small. The ALM algorithms may not be computationally faster when P is large as the algorithm iteratively optimises both β and the dummy variable e (2.12), where CD only requires the optimisation of β. ALM also requires the selection of the Lagrange Multiplier ξ, which can be inefficient with a poor choice (62). Homotopy suffers in terms of computational speed when p is large as does LARS when compared to CD (53). CD seems to be the most popular of the algorithms with published R packages and is also flexible enough to fit a number of penalty functions (Table 2.1)

2.4.2 Algebra of coordinate descent algorithm

2.4.2.1 The LASSO

The LASSO minimises the function shown in (2.2) for an Ordinary Least Squares problem. To estimate some β_k for some $k = \{1, \dots, P\}$, the first step is to calculate the derivative of $\hat{\beta}(\lambda)$ with respect to β_k (2.13).

$$\frac{\delta\hat{\beta}(\lambda)}{\delta\beta_k} = \frac{1}{2N}.-2\sum_{i=1}^{N}\left(y_i - \mu - \sum_{j=1}^{P} x_{ij}\beta_j\right)x_{ik} + \lambda\, sign(\beta_k) \qquad (2.13)$$

Manipulation of equation (2.13) leads to equation (2.14) by separating $x_{ik}\beta_k$ from $x_{ij\neq k}\beta_{j\neq k}$, which are fixed estimates for all $j \neq k$. Only β_k is being estimated for the k[th] iteration.

$$\frac{\delta\hat{\beta}(\lambda)}{\delta\beta_k} = -\frac{1}{N}\sum_{i=1}^{N}\left(y_i - \mu - \sum_{i=1, j\neq k}^{P} x_{ij}\beta_j - x_{ik}\beta_k\right)x_{ik} \qquad (2.14)$$

$$+\lambda\, sign(\beta_k)$$

$$= -\frac{1}{N}\sum_{i=1}^{N}\left(y_i - \mu - \sum_{j=1, j\neq k}^{P} x_{ij}\beta_j\right)x_{ik} - \sum_{i=1}^{N} x_{ik}^2\beta_k \qquad (2.15)$$

$$+\lambda\, sign(\beta_k)$$

To calculate the solution to this equation, we set (2.15) to equal 0 and solve for β_k (2.16).

$$-\frac{1}{N}\sum_{i=1}^{N}\left(y_i - \mu - \sum_{j=1,j\neq k}^{P} x_{ij}\beta_j\right)x_{ik} - \sum_{i=1}^{N} x_{ik}^2\beta_k + \lambda\, sign(\beta_k) = 0$$

$$\therefore -\frac{1}{N}\sum_{i=1}^{N}\left(y_i - \mu - \sum_{j=1,j\neq k}^{P} x_{ij}\beta_j\right)x_{ik} + \lambda\, sign(\beta_k) = \sum_{i=1}^{N} x_{ik}^2\beta_k$$

$$\therefore\ \widehat{\beta_k} = \frac{-\frac{1}{N}\sum_{i=1}^{N}\left(y_i - \mu - \sum_{j=1,j\neq k}^{P} x_{ij}\beta_j\right)x_{ik} + \lambda\, sign(\beta_k)}{\sum_{i=1}^{N} x_{ik}^2} \qquad (2.16)$$

To further simplify equation (2.16), the denominator $\sum_{i=1}^{n} x_{ik}^2 = N - 1$ for any standardised x_{ik}. The derivative of the penalty function yields directional derivatives dependant on the sign of β_k. For any β_k a right (positive) and left (negative) derivative are calculated using the following steps:

$$Let\ \frac{1}{N}\sum_{i=1}^{N}\left(y_i - \mu - \sum_{j=1,j\neq k}^{p} x_{ij}\beta_j\right)x_{ik} = S(y,\mu,x,\beta)$$

$$Let\ \sum_{i=1}^{N} x_{ik}^2 = Sxx$$

$$if\ \beta_k > 0\begin{cases} rd = \dfrac{-S(y,\mu,x,\beta) + \lambda}{Sxx} \\ ld = \dfrac{-S(y,\mu,x,\beta) + \lambda}{Sxx} \end{cases} \qquad (2.17)$$

$$if\ \beta_k < 0\begin{cases} rd = \dfrac{-S(y,\mu,x,\beta) - \lambda}{Sxx} \\ ld = \dfrac{-S(y,\mu,x,\beta) - \lambda}{Sxx} \end{cases}$$

$$if\ \beta_k = 0\begin{cases} rd = \dfrac{-S(y,\mu,x,\beta) + \lambda}{Sxx} \\ ld = \dfrac{-S(y,\mu,x,\beta) - \lambda}{Sxx} \end{cases}$$

In order to update β_k for any iteration, if $ld.rd > 0$ then:

$$\widehat{\beta_k} = \beta_k - rd \tag{2.18}$$

Solutions for other generalised linear models such as logistic and poisson regression can be derived in the same manner.

2.4.2.2 Bridge penalties

Coordinate descent algorithms have been extended to other penalised regression methods. Fu (28) provided an outline to calculate solutions for bridge estimators including ridge regression. The algebra to derive the solution for ridge regression is similar to that shown in section 2.4.2.1. The entire function is differentiable in this case and therefore does not require the soft thresholding operator shown in (2.18). The solution is shown in equation (2.19). The solution to the elastic net solution can be used to derive both LASSO and ridge regression by tuning α (2.20). The solution for the elastic net is the basis of the glmnet package in R and similar solutions have been described in the Friedman *et al*. paper (53).

$$\widehat{\beta_k} = \frac{-\frac{1}{N}\sum_{i=1}^{N}\left(y_i - \mu - \sum_{j=1, j\neq k}^{P} x_{ij}\beta_j\right)x_{ik}}{\sum_{i=1}^{N} x_{ik}^2 + 2\lambda} \tag{2.19}$$

$$\widehat{\beta_k} = \frac{-\frac{1}{N}\sum_{i=1}^{N}\left(y_i - \mu - \sum_{j=1, j\neq k}^{P} x_{ij}\beta_j\right)x_{ik} + \lambda\alpha\, sign(\beta_k)}{\sum_{i=1}^{N} x_{ik}^2 + 2\lambda(1-\alpha)} \tag{2.20}$$

2.4.2.3 The group LASSO

Initially when first introducing the group LASSO, Yuan and Lin suggested applying Least Angle Regression algorithm (55) (LARS) to fit grouped LASSO (29) models. Tseng and Yun first suggested using a block coordinate descent algorithm to fit LASSO models (54) where a block is defined as a row of β estimates. Noah *et al*. provided a solution for the group LASSO using block coordinate descent (2.21). Most R packages that fit grouped LASSO models use CD (72,75-77,84) (Table 2.1).

$$\widehat{\beta_k} = \frac{1}{\|x_k\|_2^2}\left(1 - \frac{\lambda}{\left\|x_k\left(y_i - \sum_{j\neq k} x_{ij}\beta_j\right)\right\|}\right) + x_k\left(y_i - \sum_{j\neq k} x_{ij}\beta_j\right) \tag{2.21}$$

2.4.2.4 The fused LASSO

Friedman *et al*.(51) attempted to apply the coordinate descent algorithm to fit fused LASSO models. They found that the algorithm "got stuck" at local minimum rather than the global minimum for 2 of the 100 parameters. The reason for this is because the penalty is not separable from the loss function. The authors go onto outline an alternative algorithm called the fused-LASSO signal approximator (FLSA) however, as mentioned in the paper; this algorithm is an approximation and does not guarantee a precise solution. The ALM algorithm has been shown to work for the fused LASSO (63,86).

2.5 Selection of the Tuning parameter for variable selection

In the previous section, I discussed algorithms to fit LASSO models and computed a path of solutions for varying λ estimates using the coordinate descent algorithm. At this stage a user is still required to select an optimal tuning parameter estimate. This selection is vital as it will determine which variables are deemed important and which are not. The aim in this section is to review and test existing methods for selecting single tuning parameters.

LASSO models are fitted with at least one of two goals: variable selection or model prediction. The aim of variable selection is to identify a subset of the variables that is associated with the outcome variable. The aim for model prediction is to identify a subset of the variables that can be used to accurately predict the outcome variable for another dataset (87). Selected subsets for variable selection tend to be smaller than subsets for model prediction as the type I error rate is controlled (88). Therefore choosing the tuning parameter selection method that is designed with the appropriate aim is important. For GWAS, there is a large emphasis on variable selection rather than model prediction (88). There are two main groups of tuning parameter selection methods for the LASSO; Cross-validation (CV) based methods and Information criteria (IC). Cross-validation methods are usually designed for model prediction where Information criteria are more suited for variable selection.

2.5.1 Tuning parameter selection methods for single penalties

2.5.1.1 Cross-validation based methods

K-fold Cross-validation (CV) is a commonly used method for selecting tuning parameters (89,90). It is the default method in a number of popular R packages for LASSO such as glmnet (53), lars (55) and glmpath (65). First applied by Tibshirani (9), CV is a subsampling method for model prediction. Table 2.2 describes the basic algorithm for selection of the tuning parameter using Cross-validation. Figure 2.11 and Figure 2.12 show Steps 3-6 and the selection of λ in Step 9 respectively. The number of folds (K) is user selected; this determines the number of subdivisions used for Step 1 in the Cross-validation process (Table 2.2). A small number of folds produce a small and underpowered training set leading to biased estimates. As the number of folds increase, estimates increase in covariance due to overlap between training sets. A large number of folds, such as leave-one-out CV ($K = N$), will produce a high variance (91-93). The selection of the number of folds is hence a bias-variance trade off, it is suggested that 10 folds gives a balance between bias and variance (90,91) and is the default option in both glmnet (53) and lars (55) package.

Table 2.2 Selection of tuning parameter by Cross-validation

Let λ_i = A sequence of tuning parameter estimates Let K = A user-specified number of folds
1. Randomly subdivide the dataset of N subjects into K equal folds 2. Begin at λ_1 3. Remove the kth fold from the dataset 4. Fit the LASSO on the remainder of the dataset (training set) and use the model to predict the kth fold (test set) 5. Calculate the Mean Squared Error (MSE) 6. Repeat Steps 3 - 5 for each K folds 7. Calculate the average Mean Squared Error (MSE_{λ_i}) across all K folds 8. Repeat Steps 3 -7 for all λ_i 9. Select the optimal $\lambda = \ argmin_{\ \lambda \in \{\lambda_i\}}(MSE_{\lambda_i})$

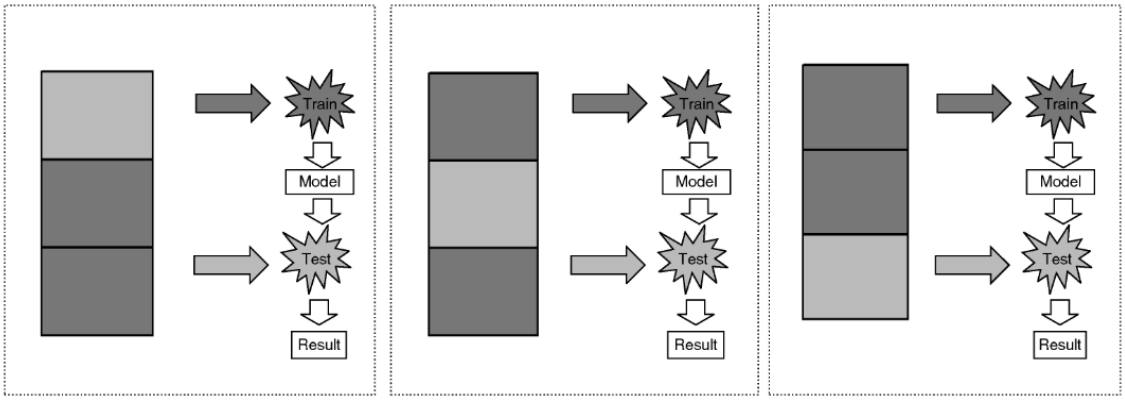


Figure 2.11 Procedure of 3-fold CV taken from Refaeilzadeh *et al*. (91). See Steps 3-6 in Table 2.2. The dataset is separated into 3 folds. One of these folds is set as the test set and removed from the dataset. The remaining two folds is the training set and are used to fit a LASSO model. Results from this model are then used to predict the test set and calculate the Mean-Squared Error (MSE). This is repeated where each fold is removed so that a mean MSE across all K-folds can be calculated in Step 7.

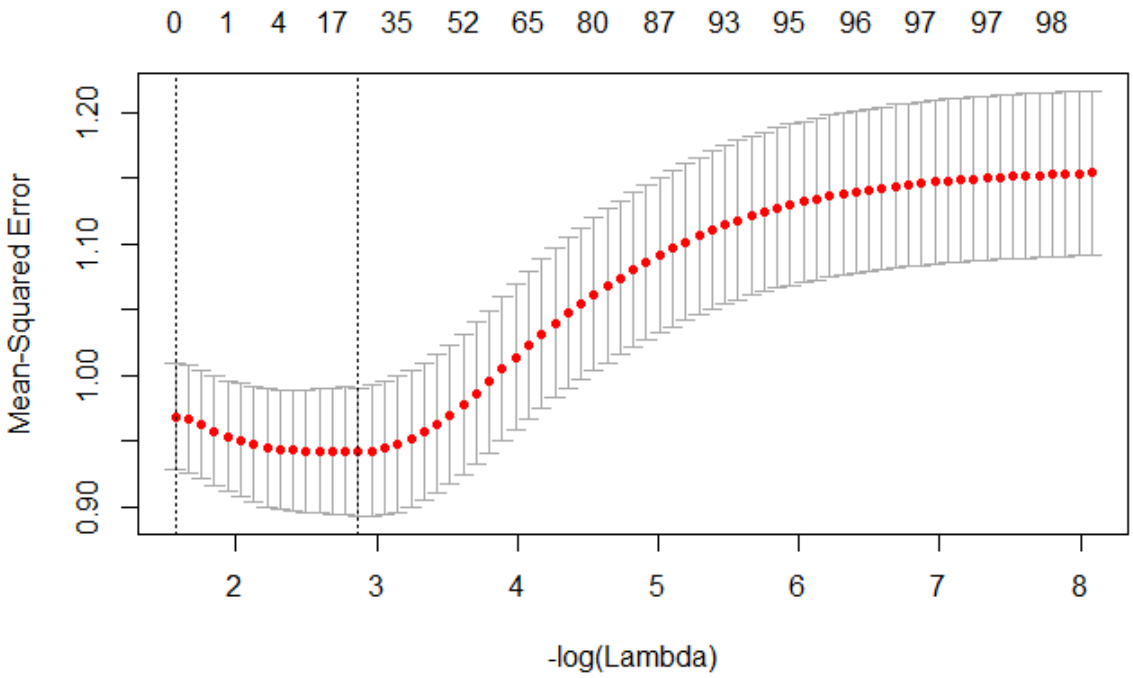


Figure 2.12 Selection of the tuning parameter from Cross-validation. Mean Squared Error is plotted against $-\log(\lambda)$ (bottom x-axis). The top x-axis counts the number of variables in the model at the corresponding $-\log(\lambda)$. The right dashed vertical line denotes the selected λ for Cross-validation. The left dashed vertical line denoted the selected λ for 1 SE Cross-validation. The simulation of the dataset for this example is described in section 2.5.4.1 with the seed = 3.

Although K-fold CV is commonly used, it tends to include a number of false positives for variable selection (87,91,94-96) and does not give consistent estimates (90) (Figure 2.13). One suggested method to reduce the false positives is the 1 Standard Error rule (92) (1SE CV). This method selects λ with the sparsest model that is no more than 1 standard error away from the optimal λ selected by CV. The 1 SE rule chooses the simplest model whose accuracy is comparable with the best model from CV (97) and is an option available in the glmnet package. Figure 2.12 shows the difference in selected λ estimates between CV and 1SE CV, the left-hand vertical line denotes the $-log(\lambda)$ estimate selected using 1SE CV compared to the right-hand vertical line which is produced using CV.

The inconsistent estimates are due to how subjects are randomised into folds leading to sample variation and therefore variation in λ estimates. Repeated CV has been suggested to produce a more stable λ estimate (97). The process repeats CV and each time recording the λ estimate which produces a distribution of λ estimates (Figure 2.13). The mean or median from this distribution can be selected as the optimal λ estimate. Due to the number of repeats of CV this method is more computationally expensive as the number of repeats increase. Other advantages and disadvantages for CV and repeated CV are described by Refaeilzadeh *et al.*(91).

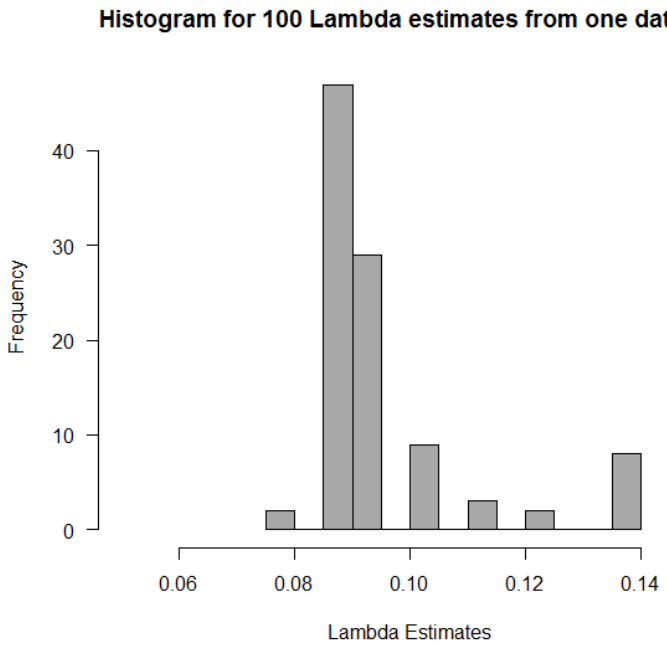


Figure 2.13 Distribution of λ estimates obtained using 10-fold Cross-validation from glmnet on one dataset repeated 100 times. The simulation of the dataset for this example is described later in section 3.2.2.1 with the seed = 3. The distribution of the estimates shows how inconsistent CV estimates can be and hence impact the final model. In this example the number of variables selected could vary between 0 and 5.

Generalised Cross-validation (GCV) is a method first suggested to tune parameters in ridge regression and also applied by Tibshirani (9). For any $\hat{\beta}$ estimates obtained by fitting a LASSO model of any $\lambda > 0$, GCV is calculated using Equation (2.22). DF_λ

denotes degrees of freedom for the model at λ. The optimal λ is selected as the minimum GCV estimate across a range of λ estimates.

$$GCV_\lambda = \frac{\left(y - x\hat{\beta}_\lambda\right)^2}{N\left(1 - \frac{DF_\lambda}{N}\right)^2} \quad (2.22)$$

2.5.1.2 Stability selection

Stability selection is a P-value based approach to model selection suggested by Meinhausen and Bühlmann (98) that also intends to address the lack of consistent estimates from CV. Table 2.3 describes the stability selection method. By selecting variables with the highest probability of selection from subsamples, stability selection will produce a stable final model. There are three considerable disadvantages to stability selection, specifically with applications to genetic data. The first is due to the use of subsampling, LASSO models are fitted on half a dataset and therefore the results will be underpowered. Rare genetic variants may not be selected if the subsample does not contain any variation in alleles and hence lowering the SNP's P-value for selection. The second is discussed by Alexander and Lange (99). In a region of SNPs with high LD, the LASSO tends to select one SNP out of the group. Within an associated genomic region with high LD the selected SNP may be different between subsamples. This will lower P-values for all SNP's in the region and lead to no SNPs meeting the probability threshold. The third disadvantage is addressed by the authors. This is the computational intensity of the method which suggests that if $P > N$ stability selection is approximately 3 times more computationally expensive and if $P < N$ this could increase to 5.5.

Table 2.3 Selection of tuning parameter by stability selection

• Let N = number of subject in the dataset • Let P = the number of variables in the dataset • Let i denote the i^{th} variable • Let Pr_i = The probability that the i^{th} variable is selected • Select t = The number of repetitions • Select Π = The probability threshold for selection
1. Without replacement, draw a random subsample from the dataset of size $\frac{N}{2}$ 2. Fit a LASSO model using K-fold Cross-validation 3. Record $Sel_i = \begin{cases} 0\ Variable\ has\ not\ been\ selected \\ 1\ Variable\ has\ been\ selected \end{cases}$ 4. Repeat Steps 1-3 t times 5. Calculate the probability that each variable is selected $Pr_i = \frac{\sum_t Sel_i}{t}$ 6. Select variables for the final model such that: $Pr_i > \Pi$

2.5.1.3 Information Criterion

Information criterions (IC) have traditionally been used for model selection in regression analyses by minimising the log likelihood function of the model plus a penalty. This penalty is based on the number of variables remaining in the model; therefore a model which contains a greater number of variables will be penalised more heavily. Recently studies have used Information criterion to select a tuning parameter (96,100,101). The advantage of IC methods over CV methods is that they take less computational time to run (102). Bayes Information Criterion (BIC) (103) is the most popular Information criterion as it is designed for variable selection and has been used by a number of studies for tuning parameter selection (90,100,102). The method calculates the residual sum of squares of the model and adds a penalty of the degrees

of freedom (DF) of the model multiplied by the log of the total number of observations (2.23) (104).

$$BIC_\lambda = N \log\left(\frac{\sum_{i=1}^{N}\left(y_i - \beta_j x_{ij}\right)^2}{N}\right) + DF_\lambda \ \log N \tag{2.23}$$

Akaike's information criterion (AIC) has also been proposed for tuning parameter selection (104). Unlike BIC, the AIC is largely used for model prediction rather than variable selection (102). It is unsurprising therefore that the proposed formula for AIC has a relaxed penalty on the degrees of freedom compared to the BIC. The difference is that the $\log N$ term is replaced with 2 which for a large N will be a smaller penalty (2.24).

$$AIC_\lambda = N \log\left(\frac{\sum_{i=1}^{N}\left(y_i - \beta_j x_{ij}\right)^2}{N}\right) + 2DF_\lambda \tag{2.24}$$

Wang *et al.* showed that the log transformation of the GCV (2.22) approximates the AIC for any given λ (100). Both AIC and BIC methods follow a similar process as described with CV (Table 2.2). A sequence of tuning parameter estimates λ_i is selected and for each λ_i the AIC or BIC is calculated. The optimal λ is one that produces the minimum AIC or BIC value (Figure 2.14). The IC methods are less computationally intensive than CV as they do not require any repetitions for each λ_i.

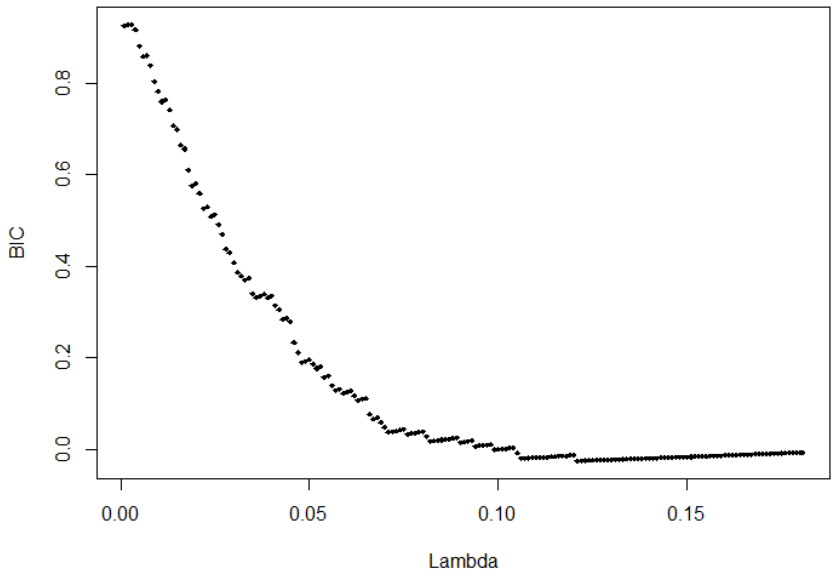


Figure 2.14 Selection of the tuning parameter using BIC. BIC values (y-axis) are plotted against the tuning parameter λ (x-axis). The selected tuning parameter is the one with the minimum BIC value. The simulation of the dataset for this example is described in section 3.2.2.1 with the seed = 3.

2.5.1.4 Permutation method

Sabourin *et al*. proposed the permutation method for tuning parameter selection (87) based on suggestions from Ayers and Cordell (88). This method (described in Table 2.4) is intended for variable selection in high-dimensional data but unlike the previously discussed Cross-validation based methods; does not use subsampling. The method uses an assumption that individual samples are exchangeable. This assumption is used to randomly exchange (permute) the outcome variable (alternatively predictor variables can also be exchanged), across subjects. This would break up any existing associations in the dataset therefore it would be expected that no variables would be selected from this permuted dataset. Therefore the LASSO is applied and the smallest λ to produce a null model is chosen. This process repeated a number of times with different random permutation each time to produce a distribution of λ estimates in

which the median is selected (Ayers and Cordell suggest using the maximum estimate (88)). From this distribution, the median λ is then the selected tuning parameter for the original dataset that has not been permutated. The number of permutations required to control the variability in $\hat{\lambda}_t$ estimates varies between authors. An increase in permutations will increase the computational time taken for the analysis but also increase the accuracy of the $\hat{\lambda}_t$ estimate. Ayers and Cordell suggest using over 25 permutations where Sabourin *et al*. suggest 100 but also admit that in some cases a lower value such as 20 can be sufficient.

Table 2.4 Selection of tuning parameter by the Permutation method

• Let N = number of subject in the dataset • Let P = the number of variables in the dataset • Let t = the t[th] permutation • Select T = The number of permutations • Let y = a vector of outcome variable values, where $y = \{y_1, \dots, y_N\}$ • Let x = an *N* x P dataset of predictor variables
1. Create a vector y_t of length N. Randomly allocate (permute) any cell from y to y_t without replacement. 2. Fit a LASSO model of X against y_t 3. Calculate and record $\hat{\lambda}_t$ which is the smallest λ that produces a null model 4. Repeat steps 1-3 T times 5. Calculate $\hat{\lambda}_{perm} = median\ (\hat{\lambda}_1, \dots, \hat{\lambda}_T)$ 6. Fit a LASSO model of x against y with tuning parameter $\hat{\lambda}_{perm}$

Both Ayers and Cordell (105) and Arbet *et al*. (106) suggest variations of the permutation method where after permutation, the estimated $\hat{\lambda}_t$ is the smallest λ that

produces a model contains a pre-specified number of variables > 0. However, the definition by Sabourin *et al*. makes more sense. If the outcome variable is permuted correctly, all associations between the variables and the outcome would be broken and therefore a null model would be expected and not a model containing some number of variables.

Yi *et al*. proposed a permutation based method to control the false discovery rate (15) (see section 4.6) and found the method to work well for a small number of causal SNP but the method became more conservative as the number of causal SNPs increased, they concluded that this was due to the permutation of the phenotype also permuting the random error in the data. This leads an over estimation in the random error in the data resulting in a loss of power. It is not known if this is the case for all permutation based methods or just the method proposed by Yi *et al*. Studies by Sabourin *et al*. (87), Ayers and Cordell (105), and Arbet *et al*. (106) do not mention such a problem occurring.

2.5.1.5 Comparisons of methods in the literature

Tibshirani (9) describes selecting the tuning parameter λ by 5-fold CV, GCV and a method based on Stein's unbiased estimate of risk, details of this method can be found here (107). Both CV methods produced a smaller median MSE estimate than Stein's method across 4 simulated examples than Stein's method. Results on the mean number of 0 coefficients is also provided although it is not clear whether these are true or false negatives in all examples. Hirose *et al*. (101) simulated the same scenarios (over 200 datasets instead of 50) as Tibshirani and included Information Criterion such as a bias corrected AIC, BIC, CV, GCV and Mallows C_p statistic for comparison. Results showed that GCV had the highest true and false positive rates across all four examples, suggesting that this method included more variables in the final model compared to the other methods (see Table 3 (101)). The probability of selecting the true model was the smallest for GVC and the largest for 10-fold CV across all methods. BIC and Cross-validation generally out-performed the other methods especially in terms of

minimising the FPR. The BIC also consistently produced a lower MSE estimate than CV suggesting that the tuning parameter estimate selected by BIC produces a better model fit. The authors also note that the C_p statistic and AIC can yield the same result which is shown in a simulation by Kirkland *et al.*(102). This simulation showed that the kappa coefficient method returned the correct model more often than the comparison methods. The authors however, noted that this method does not give consistent variable selections for the LASSO even though only 8 variables were simulated. On average the kappa coefficient tended to select the sparsest model and therefore underestimated the true model. The 1SE CV methods (5-fold and 10-fold) performed similarly to the kappa method in this simulation followed by the BIC method. The BIC method was found to have the highest rate of selecting a model that included the correct model but may also include some false positives.

The permutation method was compared with 1SE Cross-validation, BIC, the covariance test by Lockhart et al. (36) and the Scaled Sparse Linear Regression (Scalreg) method (108). Details of both these methods are described in the Sabourin *et al.* paper (87). 16 scenarios were simulated and repeated 100 times each with variables such as the dimensionality of the data (low N = 200, high N = 1,000), Signal to Noise Ratio (low SNR = 0.5, high SNR = 2) and number of causal variables (1, 5, 10 and 20) varied in each scenario. This was repeated for both Gaussian regression and logistic regression; however Scalreg results were omitted for logistic regression as the method was designed for Gaussian regression only. Performance was measured by the average power for true positives and the average false discovery rate (FDR) for false positives. Results showed that 1SE CV, BIC and the Permutation method producing similar results in most scenarios. The covariance test performed well when the number of causal variables was small but suffered when this number increased as the method tended to select the smallest model. For Gaussian models the Scalreg method was comparable in low-dimensional datasets but tended to have a higher FDR that the competing methods. The authors also compared the computational time taken for each method. In most low-dimensional scenarios, the permutation method was computationally the

fastest followed by the BIC, however in high-dimensional scenarios BIC was the fastest followed by the permutation method.

Waldmann *et al*. used CV and 1SE CV methods in a genetic setting to compare variable selection between the LASSO and EN (27). Their simulations showed that CV selected a large number of false positives whilst the 1SE method did not select many SNPs but did not select any false positives. The authors concluded that both methods do not perform well for variable selection and an alternative criterion to MSE should be used.

2.5.2 Tuning Parameter selection for dual penalties

Some penalised regression methods, such as the elastic net (18) and the grouped LASSO by Zhou *et al*. (109), apply two penalty terms rather than one. There is little literature that looks into methods to optimise dual penalties, however there are two simple suggested methods for selecting the optimal penalty across two penalties, both are briefly discussed by Zhou *et al*. (109).

The first suggested method is to fix some ratio between the two penalty estimates. This is the option used for the elastic net for glmnet where the “alpha” option controls the strength of the LASSO and ridge regression penalties respectively (2.5). This method would give the user the control on how they would like to penalise any given dataset but it may not give the optimal penalty for variable selection. Given the popularity of glmnet to fit penalised regression models this method has been the most commonly used method for selecting tuning parameters for the elastic net.

The second is to perform a “grid” type search for the optimal combinations of tuning parameters, this is performed by calculating the estimate statistic such as MSE or BIC for every combination of tuning parameter penalties and selecting the minimum of these estimates as the optimal penalty. This method will be more computationally intensive than the first suggested method as it may calculate every combination of the two tuning parameters.

The two suggested methods described above are simple and easy to implement on most tuning parameter selection methods such as CV, BIC, AIC and GCV as the tuning parameter estimates are derived from some statistic estimate where the optimal penalty is the minimum. Both stability selection and the permutation method would work by fixing the ratio between two penalties, however this would not work for a grid search method. Both methods do not use a statistic that can be used to compare models with different penalties but instead a threshold is used. The permutation method uses a λ estimate that produces a null model. Some combination of the two small λ estimates will not produce a null model; therefore a ratio of λ would be required rather than a grid for this method. Both methods ultimately pose a problem in terms of selection of the best model from a number of different "best models". One potential way to overcome this using a grid search would be to calculate model fit by either MSE or BIC for each of the "best models" and select the one with the minimum value.

2.6 Genetic association

Deoxyribonucleic acid (DNA) forms the human genome, and it is composed from four different types of molecules called nucleotides: adenine (A), cytosine (C), guanine (G) and thymine (T). Nucleotides are joined by covalent bonds to form base pairs. There are approximately 3.3 billion base pairs in the human genome which are contained in 23 pairs of chromosomes. A large portion of the genome is identical for all humans in a population, for example every person in a population has an AA pair of nucleotides. There is however a number of locations in the genome where there is variation in the pair of nucleotides in a population, for example some individuals in a population may have an AA; others may have AG and some GG. This genetic variation in the population is known as a Single Nucleotide Polymorphism (SNP) and occurs through genetic mutation in an individual and spread into the population through mating. The specific variant forms of the SNP (i.e. A and G) are known as alleles and the pair of alleles collectively is known as a genotype (i.e. AA, AG or GG).

For association testing, genotype data is used and each genotype is coded by a 0, 1 or 2 depending on the pair of alleles and the relative frequency of these alleles in the population. The minor allele frequency (MAF) is the frequency of the least common allele in a population and therefore the genotype coding tends to be by the number of minor alleles for every SNP in an individual. For example, a SNP with A and G alleles where A is the minor allele would be coded by the following: GG = 0, AG = 1, AA = 2.

A genetic association test will test the phenotype of interest against the number of minor alleles assuming an additive model (110,111). For a quantitative trait such as LDL, an ordinary least squares (OLS) model is used (2.25) and for a binary trait, such as cancer, a logistic regression model is used (2.26).

$$y = \beta_0 + \beta_1 x + \beta_2 C + \varepsilon \tag{2.25}$$

$$logit(y) = \beta_0 + \beta_1 x + \beta_2 C \tag{2.26}$$

y is a vector of phenotype values and β_0 is the intercept term for the linear model. β_1 and β_2 denote a vector of effect estimates for the matrix of genotypes *x* (coded as 0,1,2) and matrix of covariates *C* respectively. The matrix of covariates contains non-genetic risk factors that the model can adjust for such as age and sex. For the quantitative trait, ε is the error term that contains the residual variance of the phenotype that is not explained by the model, where $\varepsilon \sim N(0, \sigma^2)$. In a GWAS the matrix of genotypes (*x*) can contain millions of SNPs; the null hypothesis for each SNP is that there is no effect on the trait as the number of minor alleles present in an individual in the population increases ($\beta_1 = 0$).

2.7 The LASSO in genetic epidemiology

There have been a number of studies that have applied penalised regression methods to genetic SNP data, both in simulated and real datasets. In this section, I review some of these studies and discuss their results and conclusions. As this thesis focuses on frequentist approaches to variable selection, Bayesian approaches (112-118) and studies that focus on model prediction (119-126) will not be discussed.

Studies by Yi *et al*. (15) and Sung *et al*. (127) compared penalised regression methods against single marker regression methods. The comparisons made by the Yi *et al*. study (15) include the LASSO, adaptive LASSO, fusion-type penalties, the elastic net, Bonferroni correction and false discovery date (FDR) methods on simulated single chromosome and multiple chromosome datasets. Both the Bonferroni correction and FDR methods are described in greater detail in Chapter 4. Variable selection was assessed in terms of true and false positive rates (TPR and FPR). The authors found that there was very little difference between the penalised regression methods tested although the elastic net with $\alpha = 0.5$ performed slightly better of the four methods. The penalised regression methods were more powerful than the single marker methods tested. Waldmann *et al*. compared the LASSO and against the elastic net in both simulated scenarios and in real data from cattle (128). In each case, the results showed that the switch from the LASSO penalty (2.2) to an elastic net penalty (2.5) increased the number of SNPs selected in the model and an increased in the number of false positives selected. Based on the simulated data which simulated 25 causal SNPs in a dataset of 50,000 SNPs and varying levels of LD (high, mixed and low), the elastic net with a large ridge penalty relative to LASSO penalty ($\alpha = 0.9$) showed the best performance for variable selection (see Table 1 (128)). The authors also state the Bonferroni correction method showed similar performance to the elastic net. The LASSO was able to restrict the number of false positives selected but this also limited the number of true positives selected. The number of causal SNPs selected greatly

increased when the simulated LD between SNPs was low. The main conclusion of this paper however, was that both the tuning parameter selection methods (CV and 1SE CV) used performed poorly for variable selection and other methods should be utilised.

Sung *et al*. compared the application of the LASSO against a single marker analyses (SMA) in a simulated dataset consisting of 6,857 subjects and 4,589 SNPs along a chromosome (127). Only one SNP was simulated as the causal SNP with another 12 polygenic SNPs with "smaller effects that influenced the simulated LDL phenotype" (129). The results compared the rankings of selection for both methods. The ranking for the SMA was based on the univariate P-value rank; with the SNP with the smallest P-value has a rank of 1 and regarded as the highest rank, whereas the LASSO is the rank at which the SNP enters the model. The difference in ranks showed the difference between the two methods in accounting for LD. The SMA selected the causal SNP as the top rank in most of the 200 replications, but this also produced a high rank in 10 other associated SNPs that were correlated with the causal SNP. The LASSO selected the causal SNP as the top rank in 114 out of the 200 replications. When the causal SNP was not selected as the top rank, one of three SNPs correlated with the causal SNP was selected as the top rank and the remaining correlated SNPs produced low ranks. This is consistent with observations made by Zou and Hastie (18) which is that in a group of highly correlated variables, the LASSO tends to select only one variable and does not select any others. This ability is somewhat of an advantage compared to SMAs in GWAS, in a group of highly correlated SNPs only one SNP is likely to be the causal SNP and the remaining SNPs are correlated with the causal SNP. An SMA is likely to select a number of variants from an associated region when only one SNP is causal where the LASSO tends to select one SNP from the region.

One of the advantages of penalised regression methods over SMAs is the ability to jointly model the variables. By jointly analysing SNPs, penalised regression methods are able to consider the correlation of each marker with the phenotype, conditional on all other relevant markers. This can increase the power to detect weak associations

compared to single marker methods due to the smaller residual variance and the fact that the conditional correlation of a marker with the phenotype is often higher than the marginal correlation (130). However in the Sung *et al*. study, both methods were unable to select any of the 12 SNPs with the smaller effects, suggesting both methods have a similar power to select SNPs in this analysis (127).

If allowing correlated variables into the model is particularly desirable both group penalties and fusion penalties could be used. The fused LASSO (2.9) is designed to penalise adjacent variables only, however LD between SNPs may occur over a greater number of adjacent SNPs. Bao and Wang propose an interesting fusion penalty (131) which considers a window of multiple adjacent SNPs that allows fusion across multiple SNPs rather than a pair of adjacent SNPs. Liu *et al*. proposed an extension of the grouped LASSO which penalised grouped but included a fusion penalty to smooth estimates between groups as correlation still may exist between adjacent groups (132).

The grouped LASSO (2.7) and sparse group LASSO (2.8) have been proposed for analysis of rare variants (133-135). Rare SNPs often lack power to be selected, therefore by grouping correlated rare SNPs or genes together the power to detect these rare variants increases. As penalised regression methods are designed for high-dimensional data, it is natural that some interaction models have been proposed for genetic data for both gene-environment interactions (113,118,124,136,137) and gene-gene interactions (138-141).

A number of studies have commented on the inability to fit a large number of SNPs in a LASSO model due to a large computational burden (11-16). Table 2.5 lists the studies that have applied either the LASSO or a generalisation of the LASSO in a GWAS setting on human cohorts. In all of the studies with 119,000 SNPs or more, some form of pruning (or pre-screening) method was utilised to reduce the number of SNPs for analysis. SNP Pruning is a quality control procedure that removes a number of SNPs from the dataset. In most previous studies the dataset is pruned by mostly P-value.

Carlsen *et al*. used a form of forward step-wise model as a pre-screening method (16). Ahmed *et al*. used LD pruning to remove the highly correlated SNPs from the dataset (142). Other statistics such as FDR (143,144) and test score statistics (137,145) have also been used; both are similar to pruning by P-value. Pruning SNPs by P-value is logical as only the most significantly associated SNPs will remain for analysis. However little is known how pruning by P-value or any other pruning methods affect penalised regression models and therefore one of the aims for this thesis is to investigate the effects of SNP pruning on penalised regression models. This is discussed in greater detail in Chapter 6. Selection of the tuning parameter tends to be either by Cross-validation or selecting a pre-specified number of variables (Table 2.5). Of all the penalised regression methods, the LASSO seems to be the most popular used for variable selection in GWAS.

Table 2.5 Summary of studies that have applied the LASSO or generalisations of the LASSO in genome-wide association studies

Study	Phenotype	Study population/ GWAS dataset	Sample size	Penalised regression model	Number of SNPs in dataset	Pruning method used	Number of SNPs after pruning	Tuning parameter selection method	Notes
Ahmed (142)	High-density Lipoprotein (HDL)	Finnish	Not specified	LASSO	329,091	LD pruning with $r^2 > 0.8$. Previously identified associations from this dataset were also kept in the dataset	254,748	10-fold Cross-validation and stability selection	
Assimes (138)	Coronary artery disease (CAD)	Taiwanese	8,556 (5,423 controls, 3,133 cases)	Logistic LASSO model	9,087	None	9,087	Not specified	
Cho (11)	Adult height	Korean	8,842	Elastic net	327,872	Selection top 1,000 SNPs by P-value	1,000	10-fold Cross-validation	
Denis (139)	Placental Abruption	Peruvian	524 (280 cases, 244 controls)	Logistic LASSO model	118,782	None	118,782	20-fold Cross-validation	
Frost (136)	Bladder cancer	US	1,475 (610	Logistic Elastic net	1,488	None	1,488	Cross-validation	Elastic net was used to

			cases, 865 controls)						perform variable selection, later logistic regression was used to determine gene – environment interactions
Frost (137)	Alzheimer's disease	US	572 (412 cases, 160 controls)	Logistic elastic net	398,230	Select top 20,000 SNPs based on score statistic of marginal association with phenotype	20,000	Cross-validation	Elastic net was used to perform variable selection, later logistic regression was used to determine gene – environment interactions
Hoffman (13)	Crohn's disease, rheumatoid arthritis and Type 1 diabetes	Wellcome Trust Case Control Consortium	Not specified	LASSO and adaptive LASSO	Not specified	$p > 0.01$	Not specified	Select $1.5\sqrt{n}$ variables	Other penalised methods such as MCP and NEG also used on the dataset

Hong (146)	Adult height	Korean	8,842	LASSO, elastic net, adaptive LASSO and ridge regression	327,872	Selection top 1,000 SNPs by P-value and $\lvert\beta_j\rvert$	1,000	10-fold Cross-validation	Study compared each combination of penalised regression methods and pruning methods
Jiang(147)	Bipolar disorder	Wellcome Trust Case Control Consortium	Not specified	LASSO and prior LASSO	Not specified	p > 0.001 or SNP belongs to a previously identified gene and p > 0.21	916	1SE rule 3-fold Cross-validation	
Kohannim (148)	Temporal lobe volume	North America	729	LASSO	18,284	None	18,284	Leave-one out Cross-validation	
Shi (144)	Systolic blood pressure (SBP)	US	15,792	LASSO	~2.5 million	FDR > 0.2	15	Mallows C_p	
Wu (145)	Coeliac disease	British	2,220	LASSO	310,637	Score statistic	Unknown	Pre-selected model size by authors	
Yang (141)	Rheumatoid arthritis	Wellcome Trust Case Control	Not specified	Adaptive group LASSO	Not specified	Analyses conducted on individual	Not specified	5-fold Cross-validation	Two group penalties were incorporated,

		Consortium				chromosomes			one on individual SNPs the other on interactions between SNPs
Yao (149)	Forced expiratory volume in 1 second (FEV1)/ forced vital capacity (FVC)	Hurrerite	604	LASSO	246,010	p > 0.001	312	10-fold Cross-validation	

2.8 Summary

In this chapter, I have presented a background to the LASSO and some other generalisations such as ridge regression (RR), elastic net (EN), grouping penalties such as the group LASSO and sparse group LASSO and the fused LASSO. I also reviewed algorithms that fit LASSO models and these generalisations as well as tuning parameter selection methods. In Chapter 3, I implement the coordinate descent algorithm (CDA) for the LASSO and run a simulation study comparing a number of tuning parameter selection methods in terms of variable selection for the LASSO. Both implementations will be used in future work in this thesis. I also reviewed a number of studies that have applied these penalised regression approaches in a GWAS setting. The literature suggests that the LASSO is not able to select variables that univariate analyses also do not (13), however the advantage of the LASSO is that it is able to select an associated variable and remove variables that are correlated with the selected variable (127).

3 Implementation of the LASSO

3.1 Introduction

In Chapter 2, I provided a background to the LASSO and its generalisations which included a review on algorithms and tuning parameter selection methods that can be used to implement the LASSO on a dataset and perform variable selection. In this chapter, I follow up these reviews by illustrating implementations of the coordinate descent algorithm and a number of tuning parameter selection methods in a simulation study. The aim of these implementations is so that they can be used in future work in this thesis.

In section 2.4.1, I select the most effective algorithm to write in R as a foundation for future work. After selecting this algorithm, I then derived the basic solution for the LASSO and a number of other generaliseations of the LASSO in section 2.4.2. In this chapter, I follow on this work by deriving a pseudo code for an R program to fit LASSO models. I run this program on a test dataset and compare the results with an existing R package. This program will also be used in some proceeding chapters.

I also follow up the review conducted on tuning parameter selection methods (section 2.5.1) by testing a number of these methods via a simulation study. The aim of the simulation is to determine the relative performance of these methods in terms of variable selection. The methods that show good performance will be used for tuning parameter selection in further analyses.

3.2 Fitting the LASSO by coordinate descent

3.2.1 My coordinate descent algorithm for the LASSO

Table 3.1 presents my coordinate descent algorithm to fit the LASSO. This algorithm is designed for a standard coordinate descent in which parameters are estimated one at a time starting at the first variable in the dataset and ending with the last. Three loops are incorporated in this algorithm, the outer one loops over a vector of λ's (see step 9), however only one value of λ can be used. The middle loop is for a specified number of iterations and the third loop is for across SNPs. This final loop minimises the function each variable whilst keeping the remaining variables constant. Therefore in one iteration loop all variable estimates will be updated.

Convergence of a model for a specified λ is determined at the end of an iteration loop (Step 15). The criteria used to determine if convergence is reached, is by calculating the sum of the absolute difference between the new beta estimates from the current iteration loop ($\hat{\beta}$) and the beta estimates from the previous iteration loop (Oldbeta). If this sum is less than a specified threshold then convergence is reached. This threshold value should be small in order to produce accurate β estimates but not too small as this would increase the computational time. A threshold between 0.0001 and 0.000001 would seem reasonable. If a vector of λs is given, the vector should be increasing order, to make the algorithm more efficient. Once a model has converged at λ_k then the initial estimates used for λ_{k+1} are the final $\widehat{\beta}$ estimates for λ_k. Applying these "warm starts" makes the algorithm simpler and faster than restarting the initial estimates at 0 (53). The code for my LASSO function in R is in Appendix A.

Table 3.1 My Coordinate descent algorithm to fit the LASSO

- Let N = The number of subjects in the dataset
- Let i = the i^{th} subject, where i={1, …, N}
- Let P = the number of SNPS in the dataset
- Let j = the j^{th} SNP, where j={1, …, P}
- x = The N x P standardised SNP matrix
- y = A continuous Phenotype with mean μ and standard deviation σ^2
- Let k = the k^{th} value of Lambda

1. Generate a vector of increasing penalty thresholds with length K (K ≥ 1). Call it Lambda.
2. Specify the maximum number of iterations that are to be used. Call it iterations
3. Specify convergence threshold >0. Call it THRESH
4. Calculate the intercept which is the mean of Y. Call it mu
5. Calculate sxx, where $sxx = \sum_{j=0}^{P} x_j^2 = N - 1$
6. Generate a vector of initial estimates of length P. Call it Betahat ($\hat{\beta}$)
7. Generate the same vector of initial estiamtes of length P. Call it Oldbeta
8. Start at k = 1
9. Start at iter = 1
10. For each cell in Oldbeta, replace the Oldbeta values with those in Betahat
11. Start at j = 1
12. Calculate $r = \sum_{i=1}^{N}(y_i - mu)x_j$
13. Calculate the left (ld) and right derivatives (rd)

a. If $\widehat{\beta_J} = 0$ $\begin{cases} ld = -r + N\lambda_k \\ rd = -r - N\lambda_k \end{cases}$

b. If $\widehat{\beta_J} > 0$ $\begin{cases} rd = -r + N\lambda_k \\ ld = -r + N\lambda_k \end{cases}$

c. If $\widehat{\beta_J} < 0$ $\begin{cases} rd = -r - N\lambda_k \\ ld = -r - N\lambda_k \end{cases}$

14. Let New.beta denote the updated Beta estimate. In order to calculate this:

    a. If $rd\ x\ ld \leq 0$ then $\widehat{\beta_J} = 0$

    b. If $rd\ x\ ld > 0$ then

        i. Calculate $New.beta_j = \beta_j - \frac{rd}{sxx}$

        ii. Update $mu = mu + \left(New.beta_j - \widehat{\beta_J}\right)x_j$

        iii. Replace $\widehat{\beta_J} = New.beta_j$

15. Decide if the convergence criterion has been met.

    a. If $\sum_{j=1}^{NSNP}\left|\widehat{\beta_J} - Oldbeta_j\right| < Thresh$ then convergence criterion has been met. Go to Step 16

    b. If $\sum_{j=1}^{NSNP}\left|\widehat{\beta_J} - Oldbeta_j\right| \geq Thresh$ then convergence criterion has been met.

        i. If iter = iterations. Stop. Model has not converged

        ii. If j = P, set j=1 and set iter = iter+1. Go to step 10.

        iii. If j < P & iter < iterations, set j= j+1. Go to step 12.

16. Output $Lambda_k$ and the vector $\hat{\beta}$.

17. Either move to the next value of Lambda or stop.

    a. If $Lambda_k < K$, set k=k+1 and go to step 9.

b. If $Lambda_k = K$. Stop. Estimates for all specified values of Lambda have been obtained.

3.2.2 Comparison of my coordinate descent algorithm against glmnet

3.2.2.1 Simulation of data

A dataset of 500 subjects and 100 independent SNPs was simulated. Minor allele frequencies (MAF) for SNPs were randomly generated from a uniform distribution and varied between 0.01 and 0.5. The minor allele frequencies for SNP 25 and SNP 75 were set to 0.02 and 0.2 respectively as these were simulated as the causal SNPs. Simulated β's were calculated using the percentage variance explained and the MAF of the causal SNP (3.1).

$$\beta = \sqrt{\frac{Percentage\ of\ Variance\ explained}{2\ x\ MAF\ (1 - MAF)}} \tag{3.1}$$

A continuous phenotype was simulated with both causal SNPs explained 1% of the total variance each ($\beta_{25} = 0.5051$ and $\beta_{75} = 0.1768$) (3.2).

$$y_i = \beta_{25} x_{i,75} + \beta_{75} x_{i,75} + e_i \tag{3.2}$$

where,

$$e_i \sim N(0, 0.98)$$

The smallest value of λ required for a null model was λ = 0.1346. Therefore λ was selected over a range starting at 0 and increasing in intervals of 0.005 until 0.135. The maximum number of iterations was set to 10,000. The convergence threshold was set at 0.000001 (see step 15). For this simulation a seed was set as seed = 1.

For both glmnet and my algorithm, data on the number of SNPs in the model, the number of causal SNPs in the model, estimated β values, BIC and the residual sum of squares (RSS) was collected. The computational time taken was also calculated over 1,000 repetitions of the algorithm. This was performed using the `system.time()` command in R.

3.2.2.2 Results

The results for both my algorithm and glmnet are shown in Table 3.2. My LASSO algorithm was able to converge for all values of lambda (with a threshold of 0.000001). The results show similar results across the information collected. The number of SNPs in the final model was the same with both programs with the exception of $\lambda = 0.03$ where there was a difference of one SNP. The difference in the SNP estimates was negligible, as glmnet estimated β = 0.0000218, my algorithm estimated β = 0. Although there was this one difference in the number of SNPs it was not a causal SNP. There was no difference between the number of causal SNPs remaining in the final model, in both cases the same SNP (SNP 25, MAF = 0.02) was removed from the model at λ = 0.045 and SNP 75 (MAF = 0.2) was removed at λ = 0.115. The results did show however that predicted estimates between the two programs were different in all non-zero estimates across all SNPs and tuning parameter values, the difference was small in all cases ranging between 5 and 7 decimal places. The largest difference in β estimates between the two programs is 0.00013 (λ = 0.13), at this point only one SNP is remaining in the model. The size of the difference in β estimates between the algorithms (my algorithm subtracted from the glmnet) is small when calculating the residual sum of squares (RSS) of each model. The largest difference in RSS was 0.00007 (λ = 0.045). At every λ, the results showed that although there was a small difference in RSS estimates, the β estimates from my algorithm provided a smaller RSS estimate which would suggest that my algorithm produced more optimal estimates than glmnet.

The `system.time()` command in R was used to compare the time taken to compare the computational time taken for both programs, was repeated 1,000 times to give a fairer time estimation. The glmnet took 179.25 seconds (2 minutes and 59.25 seconds) to run 1,000 times. My algorithm took considerably longer at 859.52 seconds (14 minutes and 59.52 seconds) for the same process (Table 3.3). Tests across a number of convergence thresholds were run also. Naturally as the convergence threshold became smaller, the time taken to run increased. To compare the accuracy of estimates, the estimates for each threshold were compared to the estimates from glmnet by calculating the sum of the absolute difference of all SNPs across all λ values. Table 3.3 showed that the smaller the threshold the closer estimates became to those produced by glmnet. Given the little difference between estimates from a threshold of 0.0001 and 0.0000001 compared to the large difference in running time, it would be beneficial to use a threshold of 0.0001 in future analyses for similar size datasets.

Table 3.2 Results showing the comparison of my code against glmnet

λ	No. of SNPs selected - my algorithm	No. of SNPs selected - glmnet	Causal SNPs selected - my algorithm	Causal SNPs selected - glmnet	RSS - my algorithm	RSS - glmnet	Difference in RSS	Largest β difference
0.00	100	100	2	2	490.8791	490.879	3.07E-12	0.000083
0.005	91	91	2	2	490.8973	490.897	2.85E-06	0.000045
0.01	77	77	2	2	490.9091	490.909	1.14E-05	0.000028
0.015	68	68	2	2	490.9162	490.916	2.04E-05	0.000033
0.02	65	65	2	2	490.9194	490.920	3.49E-05	0.000077
0.025	54	54	2	2	490.9195	490.920	4.32E-05	0.000069
0.03	45	46	2	2	490.9183	490.919	4.59E-05	0.000059
0.035	43	43	2	2	490.9159	490.916	5.78E-05	0.000067
0.04	40	40	2	2	490.9121	490.912	6.83E-05	0.000074
0.045	32	32	1	1	490.9076	490.908	7.40E-05	0.000080
0.05	24	24	1	1	490.903	490.903	6.61E-05	0.000083
0.055	18	18	1	1	490.8993	490.899	5.52E-05	0.000078
0.06	12	12	1	1	490.8963	490.896	4.74E-05	0.000083
0.065	11	11	1	1	490.8937	490.894	5.25E-05	0.000092
0.07	10	10	1	1	490.8909	490.891	5.33E-05	0.000092
0.075	7	7	1	1	490.8882	490.888	4.23E-05	0.000094
0.08	5	5	1	1	490.8865	490.887	3.18E-05	0.000085
0.085	2	2	1	1	490.8857	490.886	1.47E-05	0.000086
0.09	2	2	1	1	490.8852	490.885	1.65E-05	0.000091
0.095	2	2	1	1	490.8846	490.885	1.83E-05	0.000097
0.1	2	2	1	1	490.8838	490.884	2.03E-05	0.000102
0.105	2	2	1	1	490.883	490.883	2.24E-05	0.000107

0.11	2	2	1	1	490.8821	490.882	2.46E-05	0.000112
0.115	1	1	0	0	490.8813	490.881	1.33E-05	0.000115
0.12	1	1	0	0	490.8808	490.881	1.44E-05	0.000120
0.125	1	1	0	0	490.8803	490.880	1.57E-05	0.000125
0.13	1	1	0	0	490.8797	490.878	1.69E-05	0.000130
0.135	0	0	0	0	490.8791	490.879	0	0

Table 3.3 Comparison of timings between glmnet and varying thresholds of my algorithm over 1,000 loops

Program	Threshold	Time taken (s)	Sum of the absolute difference of all SNPs at every λ against glmnet
glmnet	NA	179.25	NA
My algorithm	0.01	325.65	0.03934339
	0.001	464.65	0.02353101
	0.0001	586.14	0.02342260
	0.00001	720.63	0.02340717
	0.000001	859.52	0.02340617
	0.0000001	969.17	0.02340579
	0.00000001	1078.09	0.02340576

3.2.3 Conclusion

There are a number of different algorithms that can be implemented to fit non-smooth convex functions such as the LASSO. In chapter 2, I reviewed the three main categories of algorithms: first order algorithms, path finding algorithms and ADMM algorithms. Taking into consideration computational speed, accuracy, and flexibility I concluded that coordinate descent was the best algorithm to implement into my own R code as a basis for future work. A simulation showed that my algorithm written in R gave a slightly more optimal solution but computational time was considerably longer than glmnet.

3.3 Simulation Study on LASSO tuning parameter selection methods for variable selection

In this section, I conduct a simulation study to compare a number of tuning parameter selection methods in order to determine which methods perform well for variable selection and which can be used in further analyses. The methods selected for comparison were repeated 10-fold CV, repeated 10-fold 1SE CV, BIC methods and the permutation method. Most of these methods are designed for variable selection with the exception of Cross-validation which is designed for model prediction. Cross-validation was included as it remains the most popular method for tuning parameter selection (Table 2.5). Stability selection was not considered due to the reasons stated in section 2.5.1.2. Other methods such as AIC & GCV were also not considered as results in previous simulation studies showed poor performance (101).

3.3.1 Methods

Datasets were simulated as described in the previous section (see 3.2.2.1) however a number of different scenarios were simulated. Each scenario was simulated 1,000 times. The baseline scenario is described in section 3.2.2.1. Eight other scenarios were simulated, each one varied either the number of subjects (N = 1,000 and 2,000), number of independent SNPs (NSNP = 250 and 500), number of causal variants (N Causal = 5 and 10) or the percentage variance explained by each causal variant (%Var = 2% and 5%).

For the scenario where the numbers of causal variants = 5 they were set at positions 1, 25, 50, 75 and 100 with MAFs set to 0.25, 0.02, 0.1, 0.2 and 0.4 respectively. Similarly for the scenario where the numbers of causal variants = 10 they were set at positions 1, 15, 25, 35, 50, 65, 75, 80, 95 and 100 with MAF set to 0.25, 0.05, 0.02,

0.15, 0.1, 0.3, 0.2, 0.35, 0.5 and 0.4. The simulation was run using R and seed = 1 was used.

Tuning parameter selection methods were applied to each simulated dataset for each scenario. Model fitting and Cross-validation was applied to simulated datasets using glmnet (53). Glmnet was also used to calculate the minimum λ for a null model for the permutation method. For both CV methods and the permutation method, each method was repeated 25 times and both the mean and median λ estimate was used. The BIC was calculated at intervals of 0.001 along λ. Performance was determined by true and false positive rates (TPR and FPR) as well as the proportion of times the true model was selected by the method, specifically where the number of dimensions are higher.

3.3.2 Results

3.3.2.1 Cross-validation

Table 3.4 shows the results for repeated Cross-validation. Cross-validation is the only method designed for model prediction rather than variable selection; therefore it is unsurprising that on average, this method includes the highest number of SNPs in the selected model compared to the other methods leading to a high proportion of selected SNPs that are false positive. As the number of dimensions increased the number of FP SNPs selected increased, suggesting that CV will select a high number of variables that are false positives in GWAS. The mean FPR decreased as the number of SNPs increased in this simulation. Interestingly the FPR also increases as both the number of causal SNPs and the percentage variance explained by the causal SNP increases, even if the numbers of dimensions have not changed. The TPR increases in both scenarios. Selecting the mean tuning parameter over the 25 repetitions outperforms selecting the median, as the mean produces a higher TPR along with a lower FPR.

Table 3.4 Mean and standard deviation of simulation results for the repeated 10-fold Cross-validation method averaged over 1,000 datasets

N	NSNP	No. of causal SNPs	% VAR	True positive rate	False positive rate	Mean No. SNPs	Mean No. of true SNPs	Mean No. of false SNPS
Cross-validation - Median								
500	100	2	1	0.47 ± 0.41	0.05 ± 0.06	5.44 ± 6.57	0.95 ± 0.81	4.49 ± 6.09
1,000	100	2	1	0.82 ± 0.32	0.06 ± 0.07	7.94 ± 6.79	1.63 ± 0.63	6.30 ± 6.51
2,000	100	2	1	0.99 ± 0.07	0.08 ± 0.07	9.52 ± 6.67	1.98 ± 0.14	7.54 ± 6.65
500	250	2	1	0.38 ± 0.39	0.02 ± 0.03	6.49 ± 8.88	0.76 ± 0.79	5.72 ± 8.42
500	500	2	1	0.28 ± 0.35	0.01 ± 0.02	6.81 ± 10.75	0.57 ± 0.71	6.24 ± 10.37
500	100	5	1	0.63 ± 0.31	0.08 ± 0.08	11.30 ± 8.78	3.16 ± 1.53	8.14 ± 7.76
500	100	10	1	0.82 ± 0.17	0.16 ± 0.09	23.37 ± 9.59	8.17 ± 1.73	15.20 ± 8.47
500	100	2	2	0.85 ± 0.29	0.07 ± 0.07	8.21 ± 6.80	1.69 ± 0.57	6.52 ± 6.57
500	100	2	5	1.00 ± 0.02	0.08 ± 0.07	9.74 ± 6.48	2.00 ± 0.03	7.74 ± 6.48
Cross-validation - Mean								
500	100	2	1	0.51 ± 0.38	0.04 ± 0.05	5.25 ± 5.61	1.02 ± 0.77	4.23 ± 5.21
1,000	100	2	1	0.83 ± 0.29	0.06 ± 0.06	7.66 ± 6.13	1.67 ± 0.58	5.99 ± 5.89
2,000	100	2	1	0.99 ± 0.07	0.08 ± 0.06	9.45 ± 6.26	1.98 ± 0.13	7.47 ± 6.24
500	250	2	1	0.40 ± 0.38	0.02 ± 0.03	6.16 ± 7.64	0.80 ± 0.77	5.36 ± 7.25
500	500	2	1	0.31 ± 0.35	0.01 ± 0.02	6.37 ± 8.95	0.61 ± 0.70	5.75 ± 8.61
500	100	5	1	0.64 ± 0.29	0.08 ± 0.07	10.89 ± 8.05	3.18 ± 1.43	7.71 ± 7.11
500	100	10	1	0.81 ± 0.17	0.15 ± 0.08	22.98 ± 9.16	8.14 ± 1.71	14.84 ± 8.05
500	100	2	2	0.86 ± 0.26	0.06 ± 0.06	7.91 ± 6.17	1.72 ± 0.51	6.19 ± 5.96
500	100	2	5	1.00 ± 0.02	0.08 ± 0.06	9.60 ± 6.06	2.00 ± 0.03	7.60 ± 6.05

3.3.2.2 1 Standard Error Cross-validation

Table 3.5 shows the results for repeated 1SE Cross-validation. This method shows highly conservative results by selecting on average the least number of SNPs in most simulated scenarios but also produced the lowest FPR in all but one scenario. With the exception of 2 scenarios, the mean number of SNPs selected was less than one, meaning that there were a large proportion of null models selected. The implication of the results suggests that the 1SE CV method is likely to underestimate any true model. The mean estimate out-performed the median especially when the number of dimensions increased. In these scenarios the TPR was higher with the mean estimate, while there was little difference in the FPR. Both CV methods showed similar trends across scenarios with the exception of the mean number of FP SNPS as N increases. Then overall mean for repeated CV increased the mean (4.49 to 7.54) the mean for repeated 1SE CV decreased (0.14 to 0.00). Both the results for CV and 1SE CV were similar to the simulation conducted by Waldmann *et al*. where CV selected too many false positives and 1SE CV selected too few variables (27).

Table 3.5 Mean and standard deviation of simulation results for the repeated 10-fold 1Standard-Error Cross-validation method averaged over 1,000 datasets

N	NSNP	No. of causal SNPs	% VAR	True positive rate	False positive rate	Mean No. SNPs	Mean No. of true SNPs	Mean No. of false SNPS
1 Standard Error Cross-validation - Median								
500	100	2	1	0.09 ± 0.19	0.00 ± 0.00	0.32 ± 0.47	0.18 ± 0.38	0.14 ± 0.35
1,000	100	2	1	0.13 ± 0.22	0.00 ± 0.00	0.31 ± 0.47	0.27 ± 0.45	0.04 ± 0.21
2,000	100	2	1	0.15 ± 0.24	0.00 ± 0.00	0.31 ± 0.48	0.31 ± 0.48	0.00 ± 0.04
500	250	2	1	0.07 ± 0.17	0.00 ± 0.00	0.29 ± 0.46	0.13 ± 0.34	0.16 ± 0.37
500	500	2	1	0.06 ± 0.16	0.00 ± 0.00	0.31 ± 0.50	0.11 ± 0.32	0.20 ± 0.41
500	100	5	1	0.07 ± 0.13	0.00 ± 0.00	0.44 ± 0.80	0.36 ± 0.64	0.08 ± 0.34
500	100	10	1	0.20 ± 0.25	0.01 ± 0.01	2.44 ± 3.44	2.03 ± 2.49	0.46 ± 1.30
500	100	2	2	0.19 ± 0.26	0.00 ± 0.00	0.42 ± 0.54	0.38 ± 0.53	0.04 ± 0.19
500	100	2	5	0.80 ± 0.32	0.00 ± 0.00	1.63 ± 0.72	1.59 ± 0.64	0.04 ± 0.27
1 Standard Error Cross-validation - Mean								
500	100	2	1	0.11 ± 0.21	0.00 ± 0.00	0.36 ± 0.49	0.22 ± 0.42	0.14 ± 0.35
1,000	100	2	1	0.17 ± 0.24	0.00 ± 0.00	0.38 ± 0.50	0.34 ± 0.48	0.05 ± 0.21
2,000	100	2	1	0.29 ± 0.28	0.00 ± 0.00	0.59 ± 0.57	0.59 ± 0.57	0.00 ± 0.04
500	250	2	1	0.08 ± 0.18	0.00 ± 0.00	0.33 ± 0.50	0.16 ± 0.37	0.17 ± 0.40
500	500	2	1	0.07 ± 0.17	0.00 ± 0.00	0.36 ± 0.56	0.13 ± 0.35	0.22 ± 0.47
500	100	5	1	0.10 ± 0.14	0.00 ± 0.00	0.60 ± 0.80	0.50 ± 0.68	0.10 ± 0.35
500	100	10	1	0.22 ± 0.22	0.00 ± 0.01	2.62 ± 2.97	2.23 ± 2.20	0.40 ± 1.10
500	100	2	2	0.26 ± 0.28	0.00 ± 0.00	0.58 ± 0.57	0.53 ± 0.55	0.05 ± 0.24
500	100	2	5	0.83 ± 0.25	0.00 ± 0.00	1.70 ± 0.57	1.66 ± 0.50	0.03 ± 0.23

3.3.2.3 Bayes Information Criterion

The result for the BIC is shown in Table 3.6. Results show that the BIC is another conservative method as the mean number of variables selected on average was less than one in most scenarios. The BIC outperforms both CV methods as the number of FPs selected was much lower compared to repeated CV. However with the exception of one scenario (N Causal = 10), the BIC was not as conservative as the 1SE CV method. The BIC maintained a very low FPR but a higher TPR. This method performs especially well when the number of dimensions increase compared to the CV methods.

Table 3.6 Mean and standard deviation of simulation results for the BIC averaged over 1,000 datasets

BIC								
N	**NSNP**	**N Causal**	**% Var**	**True positive rate**	**False positive rate**	**Mean No. of SNPs**	**Mean No. of true SNPs**	**Mean No. of false SNPS**
500	100	2	1	0.14 ± 0.25	0.00 ± 0.00	0.48 ± 0.69	0.28 ± 0.51	0.20 ± 0.45
1,000	100	2	1	0.40 ± 0.39	0.00 ± 0.00	0.94 ± 0.96	0.80 ± 0.77	0.14 ± 0.43
2,000	100	2	1	0.83 ± 0.32	0.00 ± 0.01	1.95 ± 0.97	1.67 ± 0.64	0.28 ± 0.60
500	250	2	1	0.11 ± 0.22	0.00 ± 0.00	0.46 ± 0.66	0.22 ± 0.45	0.24 ± 0.48
500	500	2	1	0.10 ± 0.21	0.00 ± 0.00	0.44 ± 0.65	0.19 ± 0.43	0.25 ± 0.49
500	100	5	1	0.14 ± 0.19	0.00 ± 0.00	0.82 ± 1.17	0.70 ± 0.95	0.12 ± 0.42
500	100	10	1	0.14 ± 0.18	0.00 ± 0.01	1.54 ± 2.18	1.40 ± 1.84	0.14 ± 0.53
500	100	2	2	0.45 ± 0.40	0.00 ± 0.01	1.12 ± 1.11	0.90 ± 0.80	0.21 ± 0.57
500	100	2	5	0.97 ± 0.13	0.01 ± 0.01	2.49 ± 0.95	1.95 ± 0.26	0.54 ± 0.88

3.3.2.4 Permutation method

The results for the permutation method are shown in Table 3.7. There is little difference between selecting the mean or median from the distribution of λ estimates. Selecting the mean tends to select a larger λ estimate and hence selects a smaller number of SNPs. Sabourin *et al*.(87) do not explain why the median is used over the mean although the results show that there is little difference to choose between the two. While the mean will reduce the FPR, most scenarios only select between 1 and 2 SNPs for the final model and therefore selecting the median will at least on average maximise the number of TP SNPS.

This method performed well, as it selected more SNPs on average than both the 1SE CV and BIC methods including a higher number of true SNPs while maintaining a low FPR. The FPR is well controlled in this method as shown by the consistency in mean number of false SNP estimates. Across all scenarios the mean estimate only varies between 0.7 and 0.47 for the median and between 0.41 and 0.59.

Table 3.7 Mean and standard deviation of simulation results for the permutation method averaged over 1,000 datasets

N	NSNP	N Causal	% VAR	True positive rate	False positive rate	Mean No. SNPs	Mean No. of true SNPs	Mean No. of false SNPS
Permutation method - Median								
500	100	2	1	0.32 ± 0.33	0.01 ± 0.01	1.29 ± 1.07	0.64 ± 0.66	0.65 ± 0.83
1,000	100	2	1	0.68 ± 0.34	0.01 ± 0.01	1.97 ± 1.09	1.36 ± 0.68	0.61 ± 0.83
2,000	100	2	1	0.96 ± 0.14	0.01 ± 0.01	2.58 ± 0.87	1.92 ± 0.28	0.66 ± 0.81
500	250	2	1	0.23 ± 0.30	0.00 ± 0.00	1.16 ± 1.04	0.46 ± 0.60	0.70 ± 0.87
500	500	2	1	0.16 ± 0.26	0.00 ± 0.00	0.98 ± 0.98	0.31 ± 0.52	0.67 ± 0.84
500	100	5	1	0.34 ± 0.21	0.01 ± 0.01	2.28 ± 1.31	1.69 ± 1.05	0.59 ± 0.78
500	100	10	1	0.35 ± 0.15	0.01 ± 0.01	4.08 ± 1.65	3.55 ± 1.48	0.54 ± 0.76
500	100	2	2	0.70 ± 0.32	0.01 ± 0.01	2.01 ± 1.05	1.39 ± 0.64	0.61 ± 0.83
500	100	2	5	0.99 ± 0.06	0.00 ± 0.01	2.45 ± 0.74	1.99 ± 0.12	0.47 ± 0.73
Permutation method - Mean								
500	100	2	1	0.31 ± 0.33	0.01 ± 0.01	1.86 ± 1.02	0.62 ± 0.65	0.56 ± 0.77
1,000	100	2	1	0.66 ± 0.34	0.01 ± 0.01	1.86 ± 1.04	1.32 ± 0.68	0.54 ± 0.77
2,000	100	2	1	0.96 ± 0.14	0.01 ± 0.01	2.48 ± 0.81	1.91 ± 0.29	0.56 ± 0.74
500	250	2	1	0.21 ± 0.29	0.00 ± 0.00	1.02 ± 0.96	0.43 ± 0.59	0.59 ± 0.78
500	500	2	1	0.15 ± 0.25	0.00 ± 0.00	0.86 ± 0.92	0.29 ± 0.50	0.57 ± 0.79
500	100	5	1	0.32 ± 0.21	0.01 ± 0.01	2.13 ± 1.29	1.61 ± 1.06	0.53 ± 0.74
500	100	10	1	0.34 ± 0.15	0.00 ± 0.01	3.88 ± 1.60	3.42 ± 1.46	0.46 ± 0.70
500	100	2	2	0.68 ± 0.32	0.01 ± 0.01	1.89 ± 0.99	1.36 ± 0.64	0.53 ± 0.77
500	100	2	5	0.99 ± 0.06	0.00 ± 0.01	2.40 ± 0.69	1.99 ± 0.12	0.41 ± 0.69

3.3.2.5 Comparison of simulation results

Results showed that repeated CV did not perform well for variable selection. A number of previous studies have also shown that CV includes a high number of false positives (87,91,94-96). Similar results are shown in this simulation (Table 3.8 and Table 3.9). Repeated CV tended to select a greater number of variables in its final model than the other methods with 7 or more SNPs selected over 30% of the time even though only 2 causal variables were simulated. Whilst repeated CV produced the highest TPRs and FPRs in all simulated scenarios, the 1SE CV method produced the lowest rates in most scenarios as well as the most null models of any method (Table 3.8 and Table 3.9).

Table 3.8 Table listing the number of times each tuning parameter selection method selected a number of SNPs in its final model in the first scenario with N = 500, NSNP = 100, 2 causal variants each explaining 1% of the variation.

Number of SNPs in final model	CV - Median	CV - Mean	1SE CV - Median	1SE CV - Mean	BIC	Permutation - Median	Permutation - Mean
0	216	73	683	646	604	249	276
1	185	250	317	350	332	383	396
2	74	114	0	4	47	232	221
3	74	96	0	0	14	107	90
4	41	59	0	0	1	24	12
5	52	61	0	0	2	3	3
6	43	37	0	0	0	2	2
≥7	315	310	0	0	0	0	0

Table 3.9 Table listing the number of times each tuning parameter selection method selected a number of SNPs in its final model in the fifth scenario with N = 500, NSNP = 500, 2 causal variants each explaining 1% of the variation

Number of SNPs in final model	CV - Median	CV - Mean	1SE CV - Median	1SE CV - Mean	BIC	Permutation - Median	Permutation - Mean
0	305	125	699	664	620	373	419
1	181	267	299	327	332	368	366
2	50	95	0	3	36	185	162
3	46	86	0	3	10	58	42
4	44	47	1	1	1	12	9
5	38	46	1	1	0	4	2
6	34	41	0	1	1	0	0
≥7	302	293	0	0	0	0	0

The BIC method was also conservative in terms of variable selection producing a high number of null models, although this was not as high as the 1SE CV method. The greatest difference in results between these two methods is seen when the number of variables increase. When NSNP = 2,000, TPR = 0.835 for the BIC compared to 0.295 for the 1SE CV mean method. Although the FPR increases slightly for the BIC compared to a decrease for the 1SE CV mean method the gain in TPs outweighs the small change in FPs. As NSNP increases the TPR for BIC is higher than 1SE CV mean method (0.096 vs 0.0665) and both methods have similar FPRs (0.0005 vs 0.0004) suggesting that the BIC would perform better than 1SE CV in a high-dimensional setting. BIC and 1SE CV produced the lowest standard error for the mean estimates which suggest these methods are more consistent however this is due to the methods underestimating the true model where there is a small number of causal SNPs and hence a high proportion of null models. When the number of causal SNPs increased (NCAUSAL = 10), the permutation method produces the smallest standard error for the mean estimate.

Out of the variable selection based methods, the permutation method produced the highest TPR whilst maintaining a low FPR. Although the permutation method has a higher FPR than the other variable selection methods, the mean number of false SNPs selected is consistently around 0.5 across all scenarios simulated and hence the false positive rate is still small. Table 3.10 shows the percentage of the 1,000 simulations that correctly selected the true simulated model. No true model was selected when N Causal = 10. In the majority of the other scenarios, the permutation method outperforms the other methods and selects a higher proportion of true models than the competing methods. The BIC performs well, where the 1SE CV method very rarely selects the true model. Repeated CV performs on a similar level to BIC in some scenarios however the difference is due to repeated CV including a number of false positives which the BIC does not. BIC does not select as many true positives. For each final model, the Mean Squared Error was calculated and results are shown in Table 3.11. Results show that there was little difference in MSE estimates across the methods although the results tend to suggest that the sparsest model produces on average a lower MSE estimate.

Table 3.10 The percentage of times each tuning parameter selection method selected the exact true model over 1,000 simulations.

N	NSNP	N Causal	% Var	CV - Median	CV - Mean	1SE CV - Median	1SE CV - Mean	BIC	Permutation method - Median	Permutation method - Mean
500	100	2	1	1.70	1.90	0.00	0.30	1.60	5.00	5.40
1,000	100	2	1	5.00	4.20	0.30	0.50	15.60	24.80	25.70
2,000	100	2	1	6.00	4.70	0.70	4.00	55.80	46.50	51.40
500	250	2	1	1.00	1.30	0.00	0.00	1.00	3.20	3.10
500	500	2	1	0.20	0.20	0.00	0.00	0.90	1.50	1.50
500	100	5	1	0.10	0.00	0.00	0.00	0.10	0.20	0.20
500	100	10	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00
500	100	2	2	5.20	5.20	2.10	2.70	17.50	27.50	27.60
500	100	2	5	5.40	5.30	64.80	64.90	60.50	63.90	66.70

Table 3.11 The average Mean Squared Error from the final model for each tuning parameter selection method over 1,000 simulations

N	NSNP	N Causal	% Var	CV - Median	CV - Mean	1SE CV - Median	1SE CV - Mean	BIC	Permutation method - Median	Permutation method - Mean
500	100	2	1	494.57 ± 31.41	494.57 ± 31.41	493.59 ± 31.07	493.59 ± 31.07	494.75 ± 31.92	494.57 ± 31.41	494.57 ± 31.41
1,000	100	2	1	989.57 ± 44.13	989.57 ± 44.13	987.44 ± 43.37	987.44 ± 43.37	988.72 ± 41.76	989.57 ± 44.13	989.57 ± 44.13
2,000	100	2	1	1977.78 ± 61.77	1977.78 ± 61.77	1975.32 ± 63.85	1975.32 ± 63.85	1978.79 ± 63.38	1977.77 ± 61.77	1977.77 ± 61.77
500	250	2	1	493.30 ± 31.05	493.30 ± 31.05	494.18 ± 30.88	494.18 ± 30.88	493.39 ± 31.01	493.29 ± 31.04	493.29 ± 31.04
500	500	2	1	494.58 ± 30.58	494.58 ± 30.58	494.38 ± 31.38	494.38 ± 31.38	494.25 ± 31.88	494.57 ± 30.58	494.57 ± 30.58
500	100	5	1	533.25 ± 34.14	533.25 ± 34.14	531.67 ± 33.67	531.67 ± 33.67	533.17 ± 35.10	533.23 ± 34.14	533.23 ± 34.14
500	100	10	1	749.20 ± 44.96	749.20 ± 44.96	747.66 ± 44.48	747.66 ± 44.48	749.31 ± 46.21	749.17 ± 44.96	749.17 ± 44.96
500	100	2	2	489.37 ± 31.17	489.37 ± 31.17	488.40 ± 30.80	488.40 ± 30.80	489.42 ± 31.65	489.36 ± 31.17	489.36 ± 31.17
500	100	2	5	476.03 ± 30.88	476.03 ± 30.88	475.27 ± 30.67	475.27 ± 30.67	475.83 ± 31.35	476.02 ± 30.88	476.02 ± 30.88

3.3.3 Conclusion

The permutation method showed superior performance in this simulation study compared to the repeated CV, repeated 1SE CV and the BIC. The BIC also performed well although the method did not select many variables and tended to select a high proportion number of null models (Table 3.8 and Table 3.9). Cross-validation and 1SE Cross-validation produced extreme results, CV over selected the number of variables in the final model where the 1SE method under selected. A number of methods were not run in this simulation due to poor performance in previous studies.

3.3.4 Discussion

In this section, I ran a simulation study comparing various tuning parameter selection methods. From the methods used in the simulation, the permutation method outperformed the other methods in terms of variable selection for this small-scale simulation. Although this method also worked well when the dimensions of the dataset increased, this selection method is relatively untested and it's not known how well this method will work in a high-dimensional setting such as GWAS. While other tuning parameter selection methods can be easily implemented when there is more than one penalty such as the elastic net by using a grid method, the permutation method cannot be implemented in such a way. Therefore determining the optimal combination penalties using the permutation method may be difficult. The BIC also performed well but tended to select fewer variables than the permutation method. Cross-validation is a method used for model prediction rather than model selection; it is unsurprising that it tends to over select variables. The 1SE Cross-validation method on the other hand, was highly conservative and rarely selected any variables.

3.4 Summary

In this chapter, I have provided illustrations of the LASSO that can be used in future work. I write a program in R that can fit the LASSO and produce accurate estimates compared to the popular glmnet package (53). In section 3.3, I conduct a simulation study comparing a number of tuning parameter selection methods. Results showed that although Cross-validation is a popular choice for tuning parameter selection (Table 2.5), it does not perform well for variable selection. Either the BIC or permutation method should be used instead.

4 Application of the LASSO on the GRAPHIC study

4.1 Introduction

In this chapter, I apply the LASSO in a GWAS setting using the Genetic Regulation of Arterial Pressure of Humans in the Community (GRAPHIC) study dataset (10). GRAPHIC is a family based study, however for this analysis only unrelated subjects were used. The aim is to apply the LASSO to identify SNPs associated with Low-density Lipoprotein cholesterol (LDL-c). Commonly used techniques such as Bonferroni correction and false discovery rate (150) were used as a baseline comparison in identifying associations in the GRAPHIC study.

I begin by conducting a literature search of studies that have conducted a GWAS on LDL in order to identify previously known genetic associations with LDL-c. I then introduce the GRAPHIC study and describe the quality control criteria used on the dataset. The LASSO, Bonferroni correction and false discovery rate methods are then applied to the GRAPHIC study in order to compare selected associations between the methods and the known associations found in the literature search.

4.2 Genetics of LDL-c

4.2.1 Low-density Lipoprotein

Coronary Artery Disease (CAD) is the one of the leading causes of mortality worldwide (151). One of the main risk factors associated with CAD is the cholesterol level, particularly levels of Low-density Lipoprotein (LDL) and High-density Lipoprotein (152-154). Other risk factors include cigarette smoking, hypertension, a family history of Coronary Heart Disease (CHD) and age (155). The function of LDL is to carry cholesterol molecules from the liver to cells such as the muscles (156). Too much LDL however, can lead to a build-up of cholesterol in the arterial wall which ultimately leads to Atherosclerosis if left untreated (157) hence why LDL is known as "bad cholesterol"(158). High-density Lipoprotein is in a sense a role reversal of LDL as it carries cholesterol away from muscles and back to the liver in an attempt to prevent any build-up of cholesterol and therefore is known as the "good cholesterol" (158). Therefore reduced levels of HDL also contribute to an increased risk of CAD.

LDL measurements are usually obtained using blood tests. The National Heart, Lung and Blood Institute in the United States of America published guidelines on classifying levels of LDL to a risk level which are widely used and accepted (Table 4.1). Naturally some variation in an individual's LDL level will be explained lifestyle choices such as diet and exercise, however it is estimated that between 40-50% of the variation is genetically inherited (159,160).

Table 4.1 ATP III Classification of LDL levels (155)

LDL Cholesterol (mg/dL)	
<100	Optimal
100-129	Near optimal/Above optimal
130-159	Borderline high
160-189	High
≥ 190	Very high

4.2.2 Literature search

A literature search was conducted to identify previously known SNPs and genetic regions associated with LDL. If the results produced from the GRAPHIC study analysis replicated results seen in previous GWAS studies they would be generally more accepted, particularly if the association has been replicated in a number of previous studies. The keywords of "GWAS AND LDL" were used in PubMed for this search. Unabbreviated versions of these terms such as "Genome-Wide Association Studies" and "Low-Density Lipoprotein" were also searched in the case that any studies were missed by using abbreviated terms.

Table 4.2 describes the inclusion and exclusion criteria used for the literature search. LDL-c was the only included phenotype of interest and hence any analyses that included oxidised LDL, high-density lipoprotein (HDL) or triglycerides (TG) as the phenotype were excluded. Included studies were restricted to genome-wide association studies only rather than studies that analyse certain regions or selected SNPs only. All of these excluded studies selected SNPs or regions for analysis based on previously reported associations and therefore by excluding these studies, any new loci that have not been reported elsewhere have not been excluded. Studies that used either traditional GWAS methods such as ordinary least squares or meta-analysis methods to determine association were included. Included studies were restricted to human adults above 18 years of age only as this is similar to the GRAPHIC cohort. Another reason for including an age restriction is that levels of LDL is correlated with age (Figure 4.1) therefore the effect of confounding due to age is reduced. Further to this for any studies that met the inclusion criteria, associated SNPs on chromosome 23 were excluded from the search. It would be expected that the power to detect association on this chromosome would be lower. Studies were not excluded by ancestry as potential associations may be found in either SNPs or regions across different ancestries.

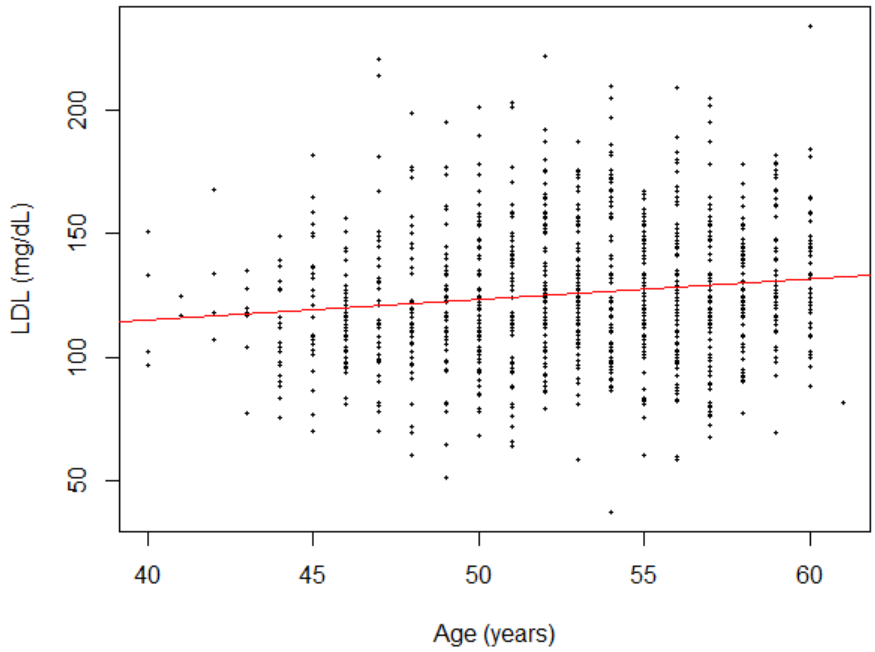


Figure 4.1 Scatter plot showing the relationship between LDL cholesterol and age, obtained from the GRAPHIC study cohort.

For studies that met the inclusion criteria the following data was collected; the name of the first author, publication date, the statistical techniques used, sample size of the study, ancestry of the cohort, the number of SNPs used in the GWAS study, the SNPs found to be associated with LDL as well as the gene, chromosome, base position and the P-value of the associated SNP (Table 4.3). The dbSNP database (161) was used in PubMed to cross-reference base positions of associated SNPs and the GRCh37.p10 assembly as a reference was used for these positions; this is the assembly reference that is used on the GRAPHIC dataset. Identifying associated SNPs in the literature search, would also indicate particular regions of interest in the genome which include a number of associations. An associated region was defined to have two or more associated SNPs that are either in the same gene or at most within 50kb of each other.

Table 4.2 Literature search inclusion/exclusion criteria

Literature search inclusion/exclusion criteria
Included any GWAS studies with LDL-C as one of its outcomes.
Included any GWAS or meta-Analysis methods with LDL-C as one of its outcomes.
Excluded any analyses with oxidised LDL as its outcome.
Included human studies only.
Included any studies that reported results by SNPs
Excluded any studies that reported results as either genes or loci only
Excluded any studies that only analysed previously known SNPs or regions
Excluded any associated SNPs on Chromosome X
Included studies conducted on adults only (≥18 years old)

Table 4.3 Data collected for included literature search studies

Data collected for included Literature search studies
Author
Publication date
Statistical methods used
Sample size
Ancestry / Population studied
Number of SNPs in dataset
Whether imputation was used or not
SNPs associated with LDL
Gene of the associated SNP
Chromosome of the associated SNP
Base position of the associated SNP
P-value of the associated SNP

4.2.2.1Results

A total 206 search results in PubMed were obtained. Table 4.4 shows the reasoning for exclusion of studies and the number of studies excluded in each case. 115 studies were excluded by title and abstract and a further 9 excluded after reading the full text as these studies had no relevance. 33 studies were excluded as they based the association testing on using previously identified SNPs or loci and a further 10 studies for reporting results by region or gene only. 11 studies were excluded as they were not

GWAS studies (4 clinical trials based on statin therapy, 3 heritability analyses, 2 simulation studies and one literature review). A final 5 studies were excluded as they either did not include LDL-c as an outcome or had a study sample consisting of children. After applying the exclusion criteria a total of 23 studies remained (162-184) and the data collected is shown in Table B.0.1 (Appendix).

Table 4.4 Literature search results and reasoning for exclusion

Reason for Exclusion	Number of studies
Search results	206
No relevance	124
Study uses previously known SNPs, regions or genes	33
Not a GWAS study	10
Reports results by region or gene rather than individual SNPs	10
Study sample were based on children	2
Did not have LDL a phenotype	3
No associations found	1
Studies Included	**23**

A total of 126 SNPs were found to have an association with LDL from the 23 studies. Each study used different methods and association levels (Table B.0.1). Associations from 28 of these SNPs were replicated in more than one study and are listed in Table 4.5. 17 of the 28 replicated SNPs were located on either chromosome 1 or 19 of genes such as SORT1/CELSR2, APOB, TOMM40 /APOE. rs6511720 on the LDLR gene (19p11.2) was replicated the most frequently (8 studies) followed by rs4420638 (19q13.2) which was replicated in 6 separate studies.

Studies were generally consistent in identifying associated SNPs across similar regions. Of the 126 SNPs identified in the literature search, 96 SNPs (76.19%) were found to be within a region of another identified SNP, 30 were not. Table 4.6 lists the SNPs identified by the literature search by gene. The commonly identified regions were the

PCSK9 (1p32.3), SORT1 (1p13.3), APOB (2p24-p23), ABCG (2p21), HMGCR5/HMGCR8 (2p21), LDLR (19p11.2) and APOE (19q13.2).

To date the largest GWAS study conducted with LDL as the phenotype was conducted by Teslovich *et al.* (180). The study combined a total of 95,454 subjects (63,274 women and 38,514 men) across 46 participating studies and approximately 2.6 million SNPs were meta-analysed across the four cholesterol based phenotypes (LDL-c, HDL-c, total cholesterol (TC) and triglycerides). Whether a SNP was significantly associated was determined by using a P -value ($P<0.0005$) obtained from a fixed-effect meta-analysis. The study identified 37 SNPs associated (Appendix B) and while 9 of these SNPs were replicated in other studies, there were a large number of SNPs identified in this study were not previously identified by either SNP or region. From the 30 SNPs that were not found to be within a region with any of the identified SNPs, 17 (56.66%) were identified by the Teslovich study (180). The other 22 studies seemed fairly consistent in terms of identifying similar regions.

Table 4.5 Identified associated SNPs that have been replicated in multiple studies.

SNPs	Gene	Chromosome	Base position	Number of replications	Studies
rs11206510	PCSK9	1	55,496,039	3	Kathiresan , Waterworth, Willer (167,182,185)
rs11591147	PCSK9	1	55,505,647	5	Chasman, Kathiresan, Musunuru, Talmund, Wu (166,171,179,184,186)
rs10889353	DOCK7	1	63,118,196	2	Aulchenko, Lettre (163,169)
rs12740374	CELSR2/ SORT1	1	109,817,590	5	Kathiresan, Lettre, Musunuru, Talmund, Wu (167,169,179,184)
rs660240	CELSR2/ SORT1	1	109,817,838	2	Middelberg, Waterworth (170,182)
rs629301	CELSR2/ SORT1	1	109,818,306	2	Talmund, Teslovich (179,180)
rs646776	CELSR2/ SORT1	1	109,818,530	5	Aulchenko, Chasman, Kathiresan, Sabatti, Saleheen (163,164,166,174,175)
rs599839	CELSR2/ SORT1	1	109,822,166	3	Kim, Roslin, Sandhu, Willer (168,173,176,183)
rs693	APOB	2	21,232,195	5	Asselbergs, Aulchenko, Kathiresan, Sabatti, Talmund (162,163,166,174,179)
rs934197	APOB	2	21,267,461	2	Musunuru, Talmund (171,179)
rs515135	APOB	2	21,286,057	2	Kathiresan, Waterworth (166,182)
rs562338	APOB	2	21,288,321	5	Lettre, Musunuru, Sandhu, Talmund, Willer (169,171,176,179,183)
rs4299376	ABCG5/ABCG8	2	44,072,576	2	Talmund, Teslovich (179,180)
rs4953023	ABCG5/ABCG8	2	44,074,000	2	Asselbergs, Musunuru (162,171)
rs12654264	HMGCR	5	74,648,603	2	Kathiresan, Kim (166,168)
rs12916	HMGCR	5	74,656,539	4	Musunuru, Talmund, Teslovich, Waterworth (171,179,180,182)
rs12670798	DNAH11	7	21,607,352	2	Aulchenko, Teslovich (163,180)

rs174546	FADS1	11	61,569,830	2	Sabatti, Teslovich (174,180)
rs2000999	HPR	16	72,108,093	2	Musunuru, Teslovich (171,180)
rs6511720	LDLR	19	11,202,306	8	Chasman, Kathiresan (x2), Lettre, Musunuru, Teslovich, Trompet, Willer (164,166,169,171,180,181,183,187)
rs2228671	LDLR	19	11,210,912	2	Aulchenko, Talmund (163,179)
rs10401969	CILP2	19	19,407,718	3	Kathiresan, Teslovich, Waterworth (166,180,182)
rs16996148	CILP2	19	19,658,472	2	Kathiresan, Willer (183,187)
rs157580	TOMM40	19	45,395,266	2	Aulchenko, Sabatti (163,174)
rs2075650	TOMM40	19	45,395,619	2	Middelberg, Talmund (170,179)
rs7412	APOE	19	45,412,079	4	Chasman, Rasmussen-Torvik, Smith, Wu (172,178,184,186)
rs12721046	APOE	19	45,421,254	2	Musunuru, Talmund (171,179)
rs4420638	APOE	19	45,422,946	6	Kathiresan (x2), Sandhu, Teslovich, Waterworth, Willer (166,176,180,182,183,187)

Table 4.6 Genes associated with LDL from the literature search. Numbers in brackets denote the number replicated associations

Gene	Chr.	Number of identified SNPs within gene	Identified SNPs within genes
PCSK9	1	5	***rs11206510 (3),*** rs2479409, ***rs11591147 (4),*** rs11806638, rs499883
DOCK7	1	3	rs10889335, rs2131925, ***rs10889353 (2)***
SORT1/ CELSR2	1	8	rs4970834, rs7528419, ***rs12740374 (5)***, ***rs660240 (2)***, ***rs629301 (2)***, ***rs646776(5),*** rs602633, ***rs599839 (4)***
APOB	2	11	rs4971516, ***rs693 (5)***, rs10199768, rs1367117, ***rs934197 (2)***, rs934197, rs7575840, ***rs515135 (2)***, ***rs562338 (5)***, rs506585, rs503662
ABCG5, ABCG8	2	5	rs6756629, ***rs4299376 (2)***, rs6544713, ***rs4953023 (2)***, rs76866386
HMGCR	5	6	***rs12654264 (2)***, rs3846662, rs3846663, ***rs12916 (4)***, rs3804231, rs258494
HAVCR1	5	3	rs6882076, rs9715911, rs1501908
LPA	6	3	rs1564348, rs3798220, rs10455872
DNAH11	7	1	***rs12670798 (2)***
NPC1L1	7	2	rs2072183, rs17725246
PPP1R3B	8	2	rs9987289, rs2126259
TRIB1	8	5	rs6982636, rs2954021, rs2954029, rs4870941, rs6987702
ABO	9	3	rs2519093, rs651007, rs635634
FADS	11	3	rs174541, ***rs174546 (2)***, rs174570
APOA	11	4	rs12272004, rs1558861, rs964184, rs2072560
HNF1A	12	2	rs2650000, rs1169288
CETP	16	2	rs3764261, rs17231506
HPR	16	2	rs72626182, ***rs2000999 (2)***
LDLR	19	10	rs1529729, rs73015011, rs11668477, rs17248720, ***rs6511720 (8)***, rs8110695, ***rs2228671 (2)***, rs5930, rs2738446, rs2738459
CILP2	19	2	***rs10401969 (3)***, ***rs16996148 (2)***
APOE	19	15	rs1531517, rs4803750, rs10402271, rs519113, rs6859, rs283813, ***rs157580(2)***, ***rs2075650 (2)***, rs1160985, rs769450, ***rs7412 (4)***, rs445925, rs389261, ***rs12721046 (2)***, ***rs4420638 (6)***
TOP1	20	2	rs1883511, rs6029526

4.3 The GRAPHIC study

Genetic Regulation of Arterial Pressure of Humans in the Community (GRAPHIC) Study: The GRAPHIC Study comprises 2,037 individuals from 520 nuclear families recruited from the general population in Leicestershire, UK between 2003-2005 for the purpose of investigating the genetic determinants of blood pressure and related cardiovascular traits in the general population. Recruitment of families was performed by invitation of women aged between 40 and 69 registered with a general practitioner in Leicestershire, UK. Families were included if both parents were aged 40-60 years and two offspring ≥18 years wished to participate. A detailed medical history was obtained from study subjects by standardized questionnaires and a clinical examination was performed by research nurses following standard procedures. Measurements obtained included height, weight, waist-hip ratio, a 12-lead ECG, lipid levels including total cholesterol, HDL and LDL and also both clinic and ambulatory blood pressure.

The subjects from the GRAPHIC study cohort were genotyped on 3 arrays, 50k Cardiochip on all samples, Exomechip that contains a large number of rare variants on both generations and HumanOmniExpress-12v1 array for the GWAS dataset which was genotyped on parental subjects only. For this analysis the GWAS dataset consisting of 1,017 parental subjects was used. Further information about recruitment and genotyping can be found here (10).

4.4 Quality Control and Exclusion Criteria

The GRAPHIC GWAS dataset consists of 1,017 parental subjects (508 males and 509 females) and 730,525 SNPs. These are mostly common SNPs however there are also some rare variants in the dataset. Any subjects with a missing phenotype (N = 35) were removed. A quality control and exclusion criterion was applied to the remaining 982 subjects. The criterion used is described in Table 4.10 below. Any individuals with a low call rate, any SNPs with a low call rate, SNPs with a small minor allele frequency (MAF), SNPs with a highly significant Hardy-Weinberg Equilibrium P-value or individuals with sex inconsistencies in the data were excluded. PLINK (version 1.07) (19,20) was used to apply the quality control procedure.

4.4.1 Low SNP call rate in individuals

A low call rate of SNPs in an individual indicates a poor DNA sample which may lead to inconsistent readings for that individual (188). Therefore it makes sense to exclude subjects with low call rates. Given that the initial sample size after removing subjects with a missing phenotype is 982 an excessive number people should not be excluded from the analysis as a reduced sample size would reduce the power to detect associations in the analysis. This is especially the case in rare SNPs as a rare variant with a low initial call rate will produce a low number of minor alleles within the population. This has a knock-on effect on association testing for that SNP as not only is the DNA sample unreliable, but also easier to produce false positive or false negative results by chance. Turner et.al (188) discusses the application and implications of quality control procedures in GWAS. A 98-99% call rate was suggested; similarly a study by Weale (189) suggests a 97-98% call rate.

Table 4.7 below shows the numbers of people that would be excluded for varying call rate cut-off points. A high cut-off point would lead to the exclusion of a high

proportion of subjects in the dataset but also reduce the power of the study. One particular subject was found to have a call rate = 86.93%, in comparison all the other subjects had call rates ≥ 94%. Any subjects with a call rate < 95% were excluded to compromise with power. Most previous GWA studies on LDL used a sample size of thousands of subjects and therefore have a greater power to detect associations with small effect sizes and can afford to exclude more subjects. The studies from the literature search with a similar sample size to this study generally found a low number of associations (172,173,177). A call rate of < 95% excludes 3 individuals and leaves 979 subjects for analysis. The mean genotyping rate across the remaining subjects was 99.32%

Table 4.7 Numbers of subjects that would be excluded for varying call rates

Individual call rate cut-off	Numbers of subjects excluded	% of people excluded
100%	982	100
99.50%	347	35.34
99%	180	18.33
98.50%	123	12.53
98%	81	8.25
97.50%	52	5.30
97%	40	4.07
96%	15	1.53
95%	**3**	**0.31**
94%	1	0.10
93%	1	0.10
92%	1	0.10
91%	1	0.10
90%	1	0.10

4.4.2 Low genotype call rate

SNPs with a low genotype call rate were excluded as these SNPs would indicate poor marker quality. An alternative approach would be to impute missing values however this can lead to error in the estimation of genotypes and could produce either false

associations or truly causal associations not being selected. It is therefore beneficial to initially exclude SNPs with a high missing call rate. Table 4.8 shows the numbers of SNPs that would be excluded for varying genotype call rates. There is a high SNP call rate across a large proportion of the GRAPHIC study. A large proportion the SNPs with lower call rates were on chromosome 23 (19% of all SNPs with a call rate < 90%). A cut-off of 97% was selected as this removed a large proportion of poor quality SNPs whilst not removing too many needlessly as the remaining missing genotypes would be imputed for analysis. In total 39,302 SNPs (5.38%) were excluded due to a low genotyping call rate.

Table 4.8 Numbers of SNPs that would be excluded for varying call rates

SNP call rate cut-off	Number of SNPs excluded	% of SNPs excluded
100%	324,372	44.40
99.50%	121,238	16.60
99%	82,357	11.27
98.50%	64,435	8.82
98%	53,162	7.28
97.50%	45,333	6.21
97%	**39,302**	**5.38**
96.50%	34,667	4.75
96%	30,965	4.24
95.50%	27,795	3.80
95%	25,615	3.51
94%	21,205	2.90
93%	17,796	2.44
92%	15,144	2.07
91%	10,037	1.37
90%	11,525	1.58
85%	6,249	0.86
80%	3,909	0.54
75%	2,837	0.39
70%	2,302	0.32
60%	1,918	0.26
50%	1,799	0.25

4.4.3 Minor allele frequency

SNPs with a low minor allele frequency (MAF) were also excluded. Rare variants will lack power to detect causality (188); this will especially be the case in this analysis as the sample size only consisted of 979 subjects compared to most previous GWA studies with LDL that use many thousands of subjects. Consider a scenario of a rare SNP with MAF of 1%, in a dataset of 1,000 people with 100% genotype call rate for this SNP. It would be expected that on average 10 subjects will have at least one minor allele and only one person to have both two minor alleles for that SNP. It is difficult to detect any true association unless there is a large effect size from the SNP on the phenotype. Conversely another issue is that with a large number of rare variants in the dataset there may be a large number of false positive associations by chance. For example, it would be easy to imagine a coincidental scenario where these ten subjects that have the minor allele of a rare SNP have a higher than average LDL than the rest of the population by chance and therefore potentially leading to a false association. This would particularly be the case for rarer SNPs as there is not enough data to otherwise reject any association.

Figure 4.2 and Figure 4.3 shows histograms of the MAFs of SNPs from the GRAPHIC study. There was a large number of monomorphic SNPs (N = 44,335), 40,042 of which have been set to a MAF of 0 due to missing allele readings. The remaining 4,293 SNPs were found to be homozygous across all subjects in the cohort. Any SNPs with a MAF < 2% were removed (N = 91,224). This cut-off was selected as it would exclude the rarest variants and therefore removing some of the issues with rare variants previously discussed whilst simultaneously attempting not to exclude too many SNPs needlessly. Due to the way commands are run in PLINK, the 91,224 SNPs excluded by MAF may have some SNPs that have also been excluded due to a low genotype call rate and therefore these exclusions are not on top of those excluded for call rate.

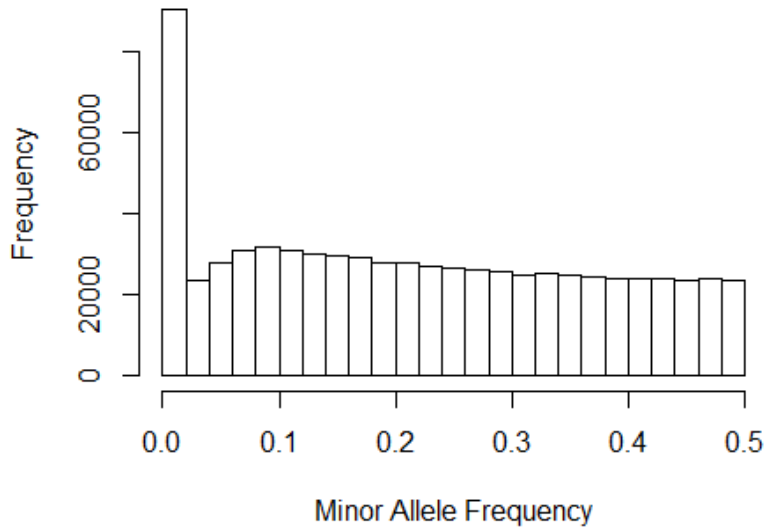


Figure 4.2 Histogram of minor allele frequencies of all SNPs

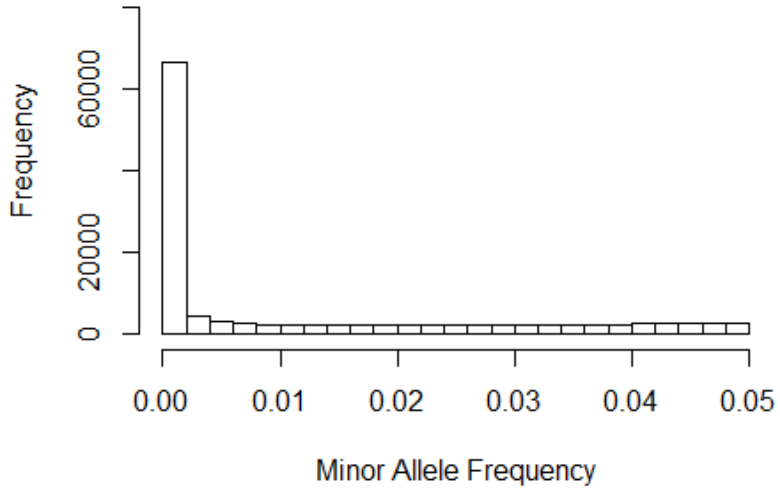


Figure 4.3 Histogram of minor allele frequencies of SNPs with MAF < 0.05

4.4.4 Hardy-Weinberg Equilibrium

Finally SNPs were excluded due to deviances from Hardy-Weinberg Equilibrium (HWE), which is based on comparing observed and expected frequency rates using chi-squared tests (190). A departure from HWE (i.e. a small P-value) suggests error in genotype calling. Figure 4.4 shows a histogram of the HWE P-values of all SNPs. There is a high number of SNPs (N = 154,087) with a P-value equal to 1. However the 40,042 SNPs that have previously identified as having missing allele (and hence will be excluded) all have a HWE P-value of 1 because observed and expected frequencies cannot be calculated and are included in Figure 4.4. The aim is to exclude SNPs with a small P-value; however as the HWE test is based on P-values the number of false positives for any given P-value cut-off point should be considered. Table 4.9 shows the number of SNPs excluded for the specified P-value cut-offs compared to the expected number of false positives assuming P-values follow a uniform distribution. A HWE threshold of $P < 0.0001$ was selected which excludes 3,089 SNPs. This threshold was chosen as it removed the most significant SNPs in disequilibrium (Figure 4.5) while minimising the number of false positives. Increasing the threshold to $P < 0.001$ would exclude a further 1,438 SNP however the number of false positives would increase by around 657 SNPs. If the threshold was to decrease to $P < 0.00001$ only include 662 SNPs for the analysis for only 66 less false positive SNPs.

Table 4.9 Numbers of SNPs for exclusion for varying HWE P-values

HWE P-value	Number of SNP's excluded	Expected number of false positives
0.1	65,955	73052.50
0.09	60,396	65747.25
0.08	54,509	58442.00
0.07	48,469	51136.75
0.06	42,501	43831.50
0.05	36,176	36526.25
0.04	30,321	29221.00
0.03	24,086	21915.75
0.025	20,932	18263.13
0.02	17,865	14610.50
0.015	14,638	10957.88
0.01	11,366	7305.25
0.001	4,527	730.53
0.0001	**3,089**	**73.05**
0.00001	2,427	7.31
0.000001	2,057	0.73
0.0000001	1,793	0.07
0.00000001	1,598	0.007

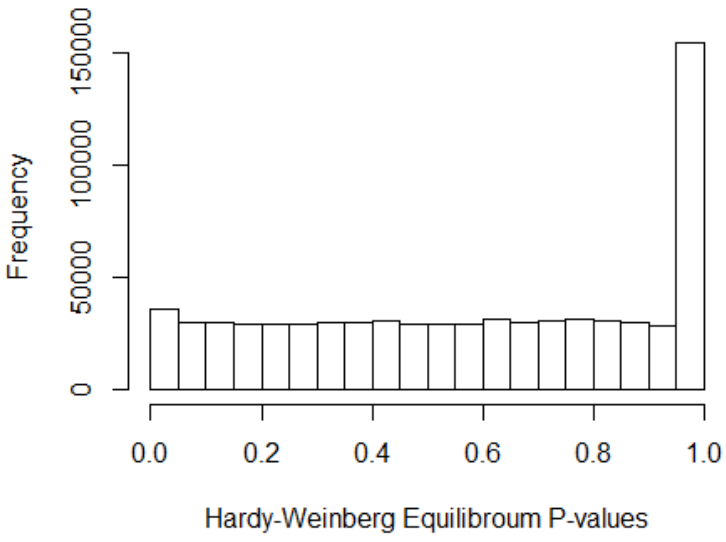


Figure 4.4 Histogram of Hardy-Weinberg Equilibrium P-values

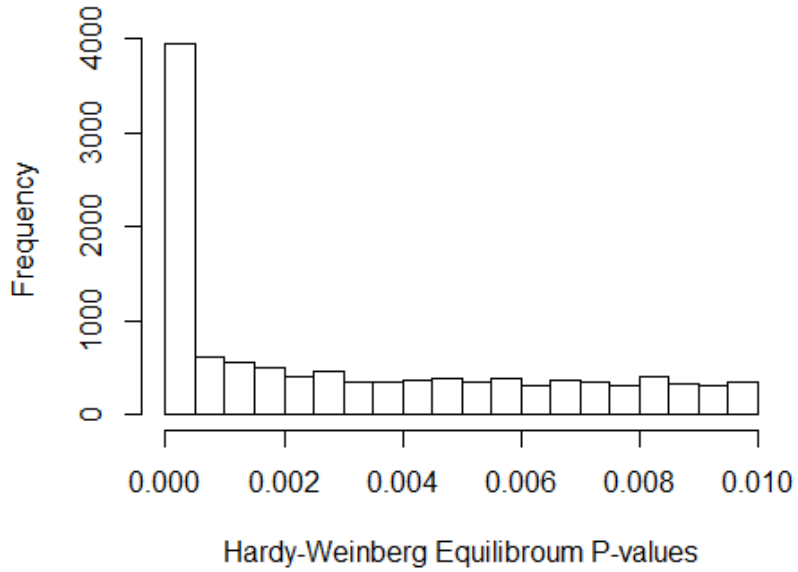


Figure 4.5 Histogram of Hardy-Weinberg Equilibrium P-values (P < 0.01)

4.4.5 Other exclusion criteria

In total 38 subjects and 117,584 SNPs were excluded from quality control. Following this some exclusion criteria were applied to the remaining 612,941 SNPs. Any SNPs with a missing base position or chromosome were removed as the region the SNP was located could not be identified (N = 1,230) and excluded all remaining SNPs on the X chromosome (N = 19,937). There were two reasons for excluding the X chromosome, the first was the significantly reduced power on this chromosome. The second was due to the low genotype call rate found in SNPs on this chromosome.

This left 979 individuals and 591,774 SNPs remaining for analysis (Table 4.10) and the characteristics of these individuals is described in Table 4.11. Of all the subjects included in the analysis, 489 were males and 490 were females with an overall mean age of 52.87 years (S.D. = 4.411) and the mean LDL cholesterol across all subjects was 125.85 mg/dL (S.D. = 25.06).

Table 4.10 Quality Control and Exclusion Criteria

Criteria	Criterion used	Numbers excluded
Across humans		
Low call rate in individuals	< 95%	3 People
Missing LDL values	-	35 People
Sex inconsistencies	-	0 People
Across SNPs		
Low call rate in SNPs	< 97%	38,129 SNPs
Minor allele frequency (MAF)	<2%	91,224 SNPs
Hardy-Weinberg Equilibrium (HWE)	P < 0.0001	3,089 SNPs
Missing Chromosome/Base position	-	1,230 SNPs
SNPs on X Chromosome	-	19,937 SNPs
A total of **979 subjects** and **591,774 SNPs** remain after quality control		

Table 4.11 Summary statistics of GRAPHIC study GWAS dataset after quality control

	Male				Female				P-value
	Mean	S.D	95% C.I.		Mean	S.D	95% C.I.		
Age (years)	53.86	4.24	53.48	54.23	51.89	4.36	51.50	52.28	< 0.001
BMI (kg/m^2)	27.81	0.17	27.47	28.15	27.09	0.21	26.68	27.49	0.008
Waist Girth (cm)	97.85	10.96	96.87	98.82	85.49	11.23	84.49	86.49	< 0.001
Hip Girth (cm)	105.04	7.35	104.39	105.70	105.05	10.19	104.14	105.95	0.994
Total Cholesterol (mg/dL)	219.88	39.72	216.35	223.41	227.21	39.31	223.72	230.70	0.004
Triglycerides (mg/dL)	185.45	96.78	176.85	194.05	141.48	75.46	134.78	148.18	< 0.001
LDL Cholesterol (mg/dL)	125.71	27.21	123.30	128.13	125.98	28.91	123.42	128.55	0.881
HDL Cholesterol (mg/dL)	50.11	11.83	49.05	51.16	62.64	14.58	61.35	63.93	< 0.001

Figure 4.6 shows a Quantile-Quantile plot of the calculated P-values against the expected P-values under a uniform distribution, using linear regression assuming an additive genetic model. Each scatter point represents a SNP. The red diagonal line shows the expected line the scatter plot would follow under the assumption that there are no associated SNPs. Any points far above this line suggest that the SNP may be associated with LDL. The plot show that there are a number of SNPs that may be associated with LDL, however both the Q-Q plot and Bonferroni correction method do not take into account of the LD between SNPs and therefore a number of these possible associated SNPs may be associated due to correlation with another highly associated SNP.

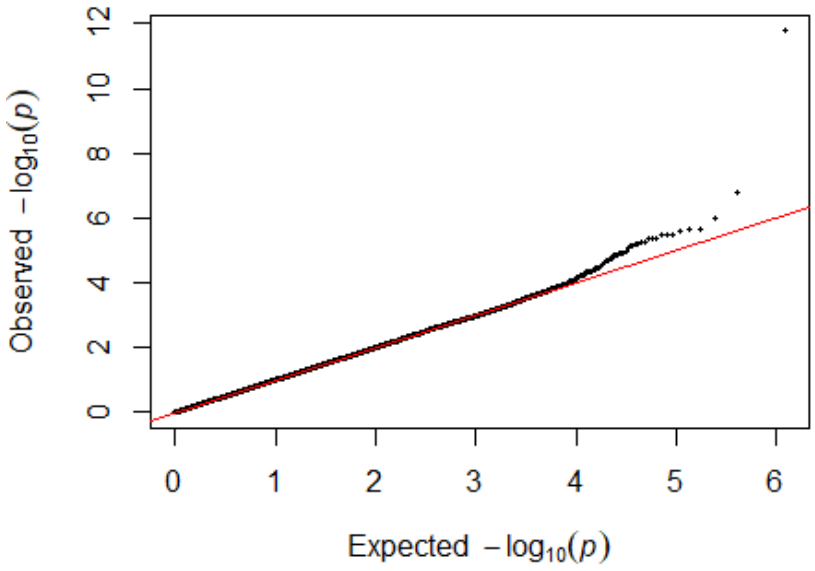


Figure 4.6 Quantile-Quantile plot for P-values from the GRAPHIC study. Each SNP's univariate $-log_{10}$ P-value on the y-axis is plotted against the expected $-log_{10}$ P-value under a uniform distribution on the x-axis. The diagonal line in red denotes the expected values the plotted SNPs would take assuming that there are no significant associations with the phenotype.

4.5 Bonferroni Correction

Bonferroni correction is a simple and commonly used statistical technique to account for multiple testing in GWAS. It controls the Type I Error rate when conducting a large number of tests. It assumes that all tests are independent of one another; an assumption that is not valid in this analysis due to Linkage Disequilibrium (LD) between SNPs. Significance testing is based on the use of P-values. Assuming that all statistical tests were null using a significance level of α_0 = 0.05, approximately 5% of all tests to be found to be statistically 'significant' by chance and therefore produce a false positive result. This would not be an issue with a low number of tests as it would lead a small number of false positives. However using a significance level of 0.05 on a dataset consisting of with 591,774 SNPs, would lead to approximately 29,589 statistically significant associations, assuming no SNPs were truly associated. In reality it would be expected that only a handful of SNPs to have a true association with LDL as shown in Table 4.5. The Bonferroni correction is a naïve method for correcting the significance level by dividing the original significance level (α_0) by the number of tests to obtain an adjusted significance level (α_a).

4.5.1 Methods

Using a significance level of $\alpha_0 = 0.05$, an adjusted genome-wide significance level of 8.449x10^{-8} was obtained. This significance level is similar but slightly less strict to the suggested genome-wide significance level of 5x10^{-8} (191).

$$\alpha_a = \frac{\alpha_o}{Number\ of\ tests} = \frac{0.05}{591{,}774} = 8.449\ x10^{-8} \qquad (4.1)$$

By scaling the significance level, the number significant SNPs that are false positives is reduced however it is likely to exclude a number of true positives at the same time which is one of the main criticisms of the Bonferroni correction method (192-194). After quality control an association test between the remaining SNPs and LDL was conducted by calculating univariate P-values. From the list of P-values obtained, any SNPs with a univariate P-value of less than 8.449×10^{-8} (4.1) were considered to be statistically significant and therefore associated with LDL.

4.5.2 Results

Figure 4.7 shows the Manhattan plot for the included SNPs after quality control. The x-axis plots each SNP in order of the chromosome and base position against the univariate $-\log_{10}$ P-value on the y-axis. The horizontal red line indicates the Bonferroni corrected significance level of 8.499×10^{-8}. As seen in the literature search (Table B.0.1), a number of the most statistically significant SNPs were found on chromosome 1 (Figure 4.8) and 19 (Figure 4.9).

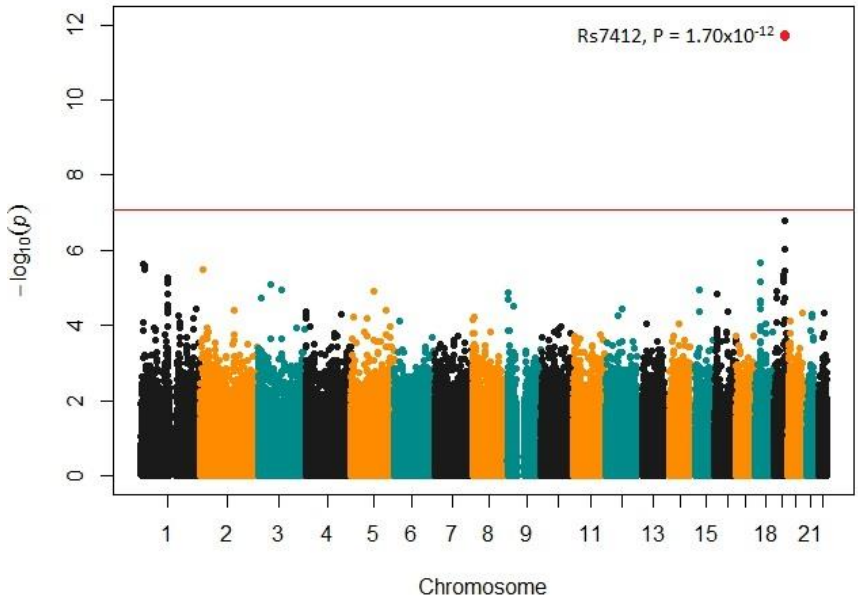


Figure 4.7 Manhattan plot of SNPs in the GRAPHIC study. Each SNP is plotted in order of chromosome and base position along the x-axis against the univariate $-log_{10}$ P-value on the y-axis. The horizontal line in red denotes the Bonferroni corrected P-value threshold of 8.499×10^{-8}

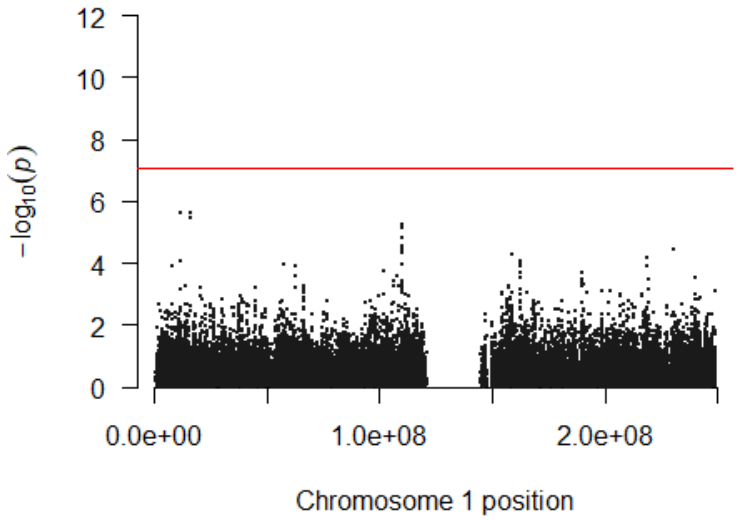


Figure 4.8 Manhattan pot of Chromosome 1. Each SNP on this chromosome is plotted in order of base position along the x-axis against the univariate $-log_{10}$ P-value on the

y-axis. The horizontal line in red denotes the Bonferroni corrected P-value threshold of $8.499x10^{-8}$

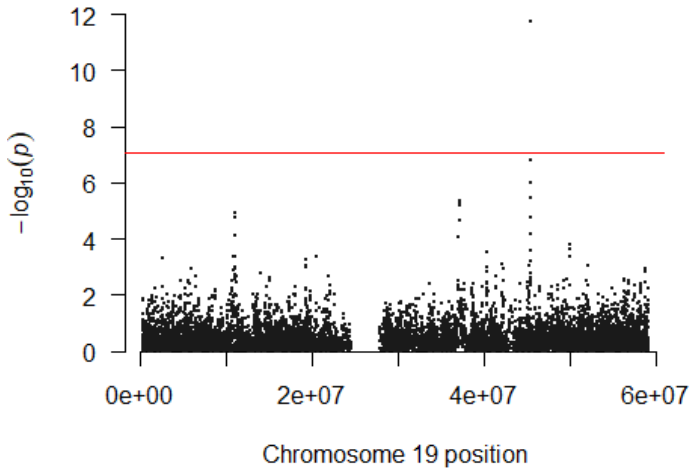


Figure 4.9 Manhattan plot of Chromosome 19. Each SNP on this chromosome is plotted in order of base position along the x-axis against the univariate $-log_{10}$ P-value on the y-axis. The horizontal line in red denotes the Bonferroni corrected P-value threshold of $8.499x10^{-8}$

Only rs7412 on the APOE gene on chromosome 19 (BP = 45,412,079), was found to have a statistically significant association with LDL-c ($p = 1.70x10^{-12}$) after applying the Bonferroni adjusted significance level. The effect estimate showed that LDL levels for individuals with the minor T allele for rs7412 decreased on average by 16.28 mg/dL (S.E. = 2.28) per allele compared to those who did not (Figure 4.10). This association between rs7412 and LDL was previously identified in the literature and has been replicated in four other studies (172,178,184,186). These previous studies showed the minor T allele of rs7412 also decreased the level of LDL in individuals. The effect size varied between -0.505 and -69.74 mg/dL.

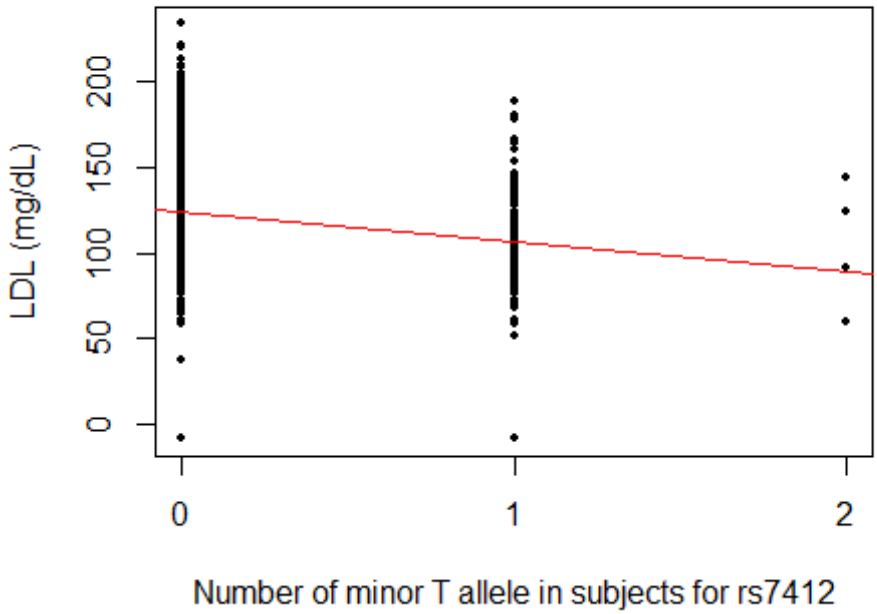


Figure 4.10 Scatter plot showing the effect of rs7412 on LDL cholesterol.

4.6 False discovery rate

One of the long standing criticisms of the Bonferroni correction method in GWAS is that it is a conservative method and may exclude a number of true positives (192-194). This could very well be the case in the previous analysis as only one SNP was selected whilst both the Manhattan plot (Figure 4.7) and Q-Q plot (Figure 4.6) showed that there may be other associations that could be selected. The false discovery rate (FDR) is another technique in GWAS that is based on Q-values which is a measure of the false discovery rate (7,8).

Table 4.12 shows a generalised scenario of testing m null hypotheses, where R tests are were found to be statistically significant. The false discovery rate estimates the expected proportion of false positives (4.2) (150).

Table 4.12 Number of errors committed when testing m null hypotheses. Taken from Benjamini *et al*.(150)

	Declared non-significant	Declared significant	Total
True null hypothesis	U	V	m_0
Non-true null hypothesis	T	S	m - m_0
Total	m-R	R	m

$$False\ Discovery\ Rate = E\left[\frac{V}{V+S}\right] = E\left[\frac{V}{R}\right] \quad (4.2)$$

A P-value is a measure of the minimum false positive rate whereas a Q-value is a measure of the minimum false discovery rate. This leads to differing interpretations in the statistics. Suppose a false positive rate of 5% was used, this is the same as setting a P-value significance level of 0.05 for each test. 5% of all null tests would therefore be statistically significant by chance. If a false discovery rate significance level of 5% was used, then 5% of all tests that are already statistically significant are false positive. A study by Storey describes a quick and efficient way of calculating Q-values based on obtained P-values (7). Given a list of P-values that can be obtained from a regression method analysis on single SNPs, the proportion of tests that are truly null (π_0) can be estimated for any tuning parameter κ (4.3).

$$\pi_0(\kappa) = \frac{m_0}{m} = \frac{\#\{p_i > \kappa; i = 1, \ldots, m\}}{m\,(1-\kappa)} \quad (4.3)$$

Using this $\pi_0$estimate for a chosen tuning parameter κ, the FDR can be estimated for any threshold *t* (4.4).

$$\widehat{FDR}(t) = \frac{\widehat{\pi_0} \,.\, m \,.\, t}{\#\{\, p_i \,\leq t \,\}} \tag{4.4}$$

And thus a Q-value can be obtained (4.5).

$$\hat{q}(p_i) = \min_{t \geq p_i} \widehat{FDR}(t) \tag{4.5}$$

Studies have shown that the FDR is a more powerful testing method compared to the Bonferroni correction method (150). The only decision that is required for this analysis is to select a FDR significance level much like selecting a significance level threshold. Like the Bonferroni method the FDR method assumes independence in all tests which cannot be assumed due to LD between SNPs.

4.6.1 Methods

For this analysis the same univariate P-values that were used in the Bonferroni correction were used to obtain a set of Q-values. The qvalue package in R was used for the FDR analysis (195). To estimate the proportion of tests that are truly null (π_0), the "specify lambda" option in the package was selected and the range of π_0 was allowed to vary between 0 and 0.99 increasing by 0.01 and the "smoother" option was also used. A Q-value threshold of 0.05 was selected. Any SNPs with q ≤ 0.05 was deemed to be associated with LDL.

4.6.2 Results

FDR analysis included all SNPs after quality control using a $q \leq 0.05$ as an FDR significance threshold. The proportion of null hypotheses (π_0) was estimated to be 0.9972. Table 4.13 shows the number of SNPs selected for varying Q-value thresholds and their respective P-value thresholds. While the distribution of P-values is fairly uniform across the range of values (Figure 4.11), it is clear that this is not the case with the Q-values as the distribution is heavily skewed towards p = 1. Of the 591,774 SNPs in the dataset, only 136 were found to have Q-value less than 0.9. The difference in distribution of P-values and Q-values is because Q-values estimate the false discovery rate rather than false positive rate.

Table 4.13 Comparison of numbers of SNPs selected for varying P-value and Q-value thresholds

Q-value	P-value	Number of SNPs
0.01	1.70E-12	1
0.05	1.58E-07	2
0.2	9.66E-07	3
0.3	8.37E-06	19
0.35	1.41E-05	26
0.4	1.95E-05	30
0.5	2.82E-05	34
0.6	4.74E-05	48
0.7	7.03E-05	60
0.8	9.74E-05	72
0.9	0.0002035	136
0.95	0.0002961	186
1	1.00E+00	591,774

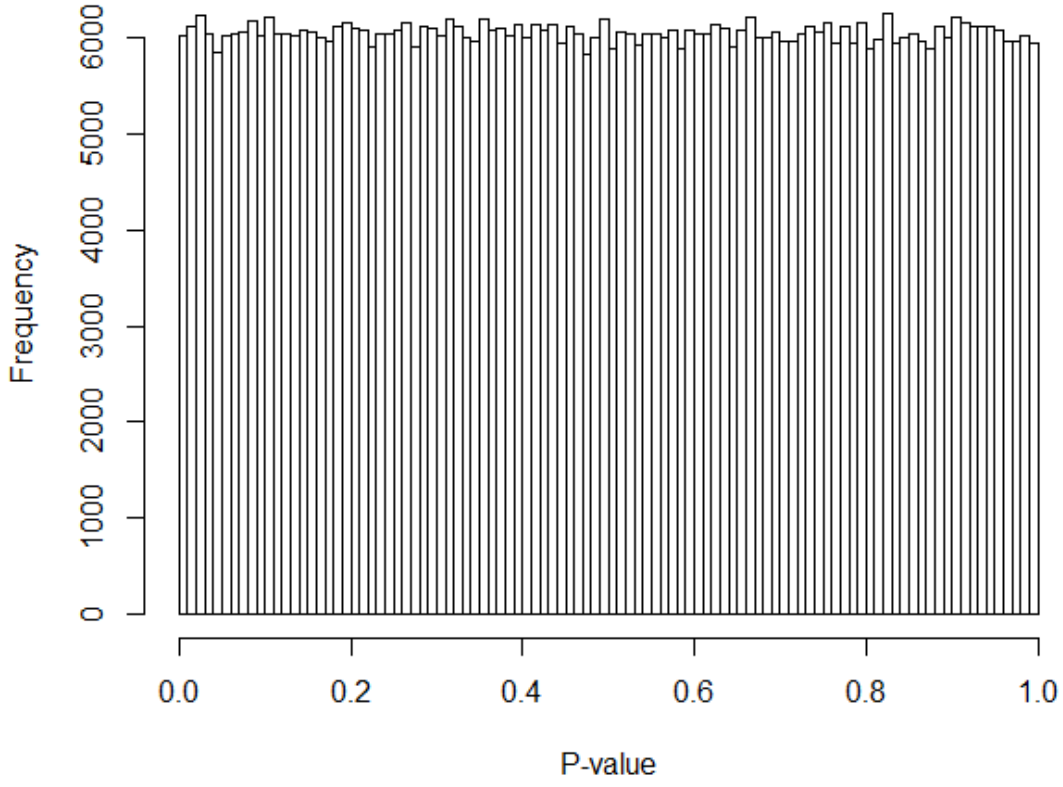


Figure 4.11 Histogram showing the distribution of univariate P-values for each SNP in the GRAPHIC study against LDL

Table 4.14 shows the top 25 SNPs by Q-value. The majority of these SNPs were on chromosome 1 (N = 6) or 19 (N = 9). These 25 SNPs are also the top 25 SNPs by P-value. Of the strongest associated regions identified from these SNPs (Table 4.15) in this study only the CELSR2 and APOE genes were previously identified in the literature search (Table 4.6). The strongest associated region was around the APOE gene on chromosome 19; including the top 3 SNPs by P-value. The FDR method found 2 of these SNPs rs7412 and rs4420638 to be associated with LDL ($q < 0.05$). Figure 4.12 and Figure 4.13 show regional plots around the APOE gene where these two SNPs are located. The figures show that while these two SNPs are in the same region to each other, there is little LD between the SNPs ($r^2 = 0.025$). The effect estimates (Table 4.14) shows that these two leads SNPs have opposite effects estimates of LDL levels on

subjects as the minor allele of rs7412 decreases the mean LDL levels where minor allele of rs4420638 increases the mean LDL level by 8.09 mg/dL (S.E. = 1.52).

Table 4.14 Top 20 selected SNPs by Q-value

CHR	SNP	Base position	Beta	S.E.	MAF (%)	P-value	Q-value
19	rs7412	45,412,079	-16.28	2.28	8.60	1.70E-12	1.00E-06
19	rs4420638	45,422,946	8.09	1.52	21.0	1.58E-07	0.0466
19	rs2075650	45,395,619	8.45	1.71	15.7	9.66E-07	0.1901
1	rs2745291	11,607,932	-7.17	1.51	21.8	2.35E-06	0.2208
1	rs12026701	16,123,199	5.86	1.25	44.5	3.29E-06	0.2208
1	rs12569079	16,124,438	5.93	1.25	44.5	2.46E-06	0.2208
2	rs1728149	10,617,598	6.56	1.40	28.0	3.20E-06	0.2208
18	rs17223656	23,100,817	-6.24	1.31	38.9	2.15E-06	0.2208
19	rs445925	45,415,640	-9.37	2.00	11.4	3.37E-06	0.2208
19	rs10402182	37,160,529	6.29	1.36	30.1	4.53E-06	0.2229
19	rs17272386	37,180,297	6.29	1.36	30.1	4.53E-06	0.2229
19	rs1525133	37,199,250	6.29	1.36	30.1	4.53E-06	0.2229
1	rs3120625	109,768,889	-6.13	1.34	33.7	5.41E-06	0.2383
1	rs7528419	109,817,192	-6.70	1.48	21.8	6.68E-06	0.2383
18	rs4800637	23,093,219	-5.95	1.32	38.9	6.87E-06	0.2383
19	rs2967442	37,064,240	6.26	1.37	30.0	5.87E-06	0.2383
19	rs1035777	37,094,435	6.19	1.37	30.1	6.64E-06	0.2383
1	rs660240	109,817,838	-6.77	1.50	21.3	7.54E-06	0.2470
3	rs646929	54,695,763	5.81	1.30	36.4	8.37E-06	0.2601
3	rs1717608	99,401,638	-5.72	1.30	35.2	1.15E-05	0.3207

Table 4.15 Associated SNPs from FDR analysis that are within regions of other associated SNPs

Gene	Chr.	SNP	Base position	Beta	S.E.	MAF (%)	P-value	Q-value
FBLIM1	1	rs12026701	16,123,199	5.855	1.251	44.5	3.29E-06	0.2208
		rs12569079	16,124,438	5.931	1.251	44.5	2.46E-06	0.2208
CELSR2	1	rs3120625	109,768,889	-6.128	1.34	33.7	5.41E-06	0.2383
		rs7528419	109,817,192	-6.695	1.478	21.8	6.68E-06	0.2383
		rs660240	109,817,838	-6.768	1.503	21.3	7.54E-06	0.2470
		rs646776	109,818,530	-6.444	1.479	22.0	1.46E-05	0.3309
ZNF521 - SS18	18	rs4800637	23,093,219	-5.964	1.319	38.9	6.87E-06	0.2383
		rs17223656	23,100,817	-6.242	1.309	38.9	2.15E-06	0.2208
ZNF520 -ZNF567	19	rs2967442	37,064,240	6.225	1.366	30.0	5.87E-06	0.2383
		rs1035777	37,094,435	6.192	1.367	30.1	6.64E-06	0.2383
		rs10402182	37,160,529	6.285	1.363	30.1	4.53E-06	0.2229
		rs17272386	37,180,297	6.285	1.363	30.1	4.53E-06	0.2229
		rs1525133	37,199,250	6.285	1.363	30.1	4.53E-06	0.2229
APOE / TOMM40	19	rs2075650	45,395,619	8.452	1.714	15.7	9.66E-07	0.1901
		rs7412	45,412,079	-16.28	2.277	8.6	1.70E-12	1.00E-06
		rs445925	45,415,640	-9.366	2.004	11.4	3.37E-06	0.2208
		rs4420638	45,422,946	8.009	1.516	21.0	1.58E-07	0.0466

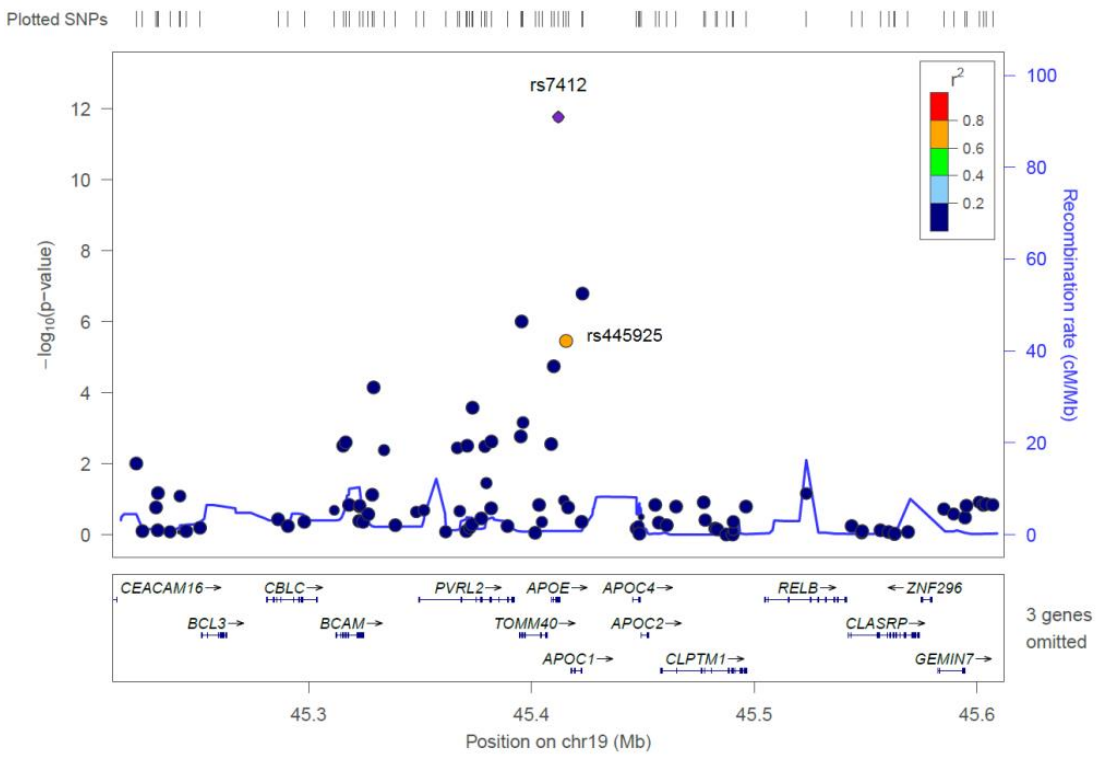


Figure 4.12 Regional plot around the APOE gene and SNPs in Linkage Disequilibrium with rs7412. SNPs are plotted in order of base position along the x-axis against the univariate $-log_{10}$ P-value on the left-hand y-axis. The blue line shows the recombination rate across this region. Colours for each SNP represent the correlation (r^2) between this SNP and the lead SNP in purple.

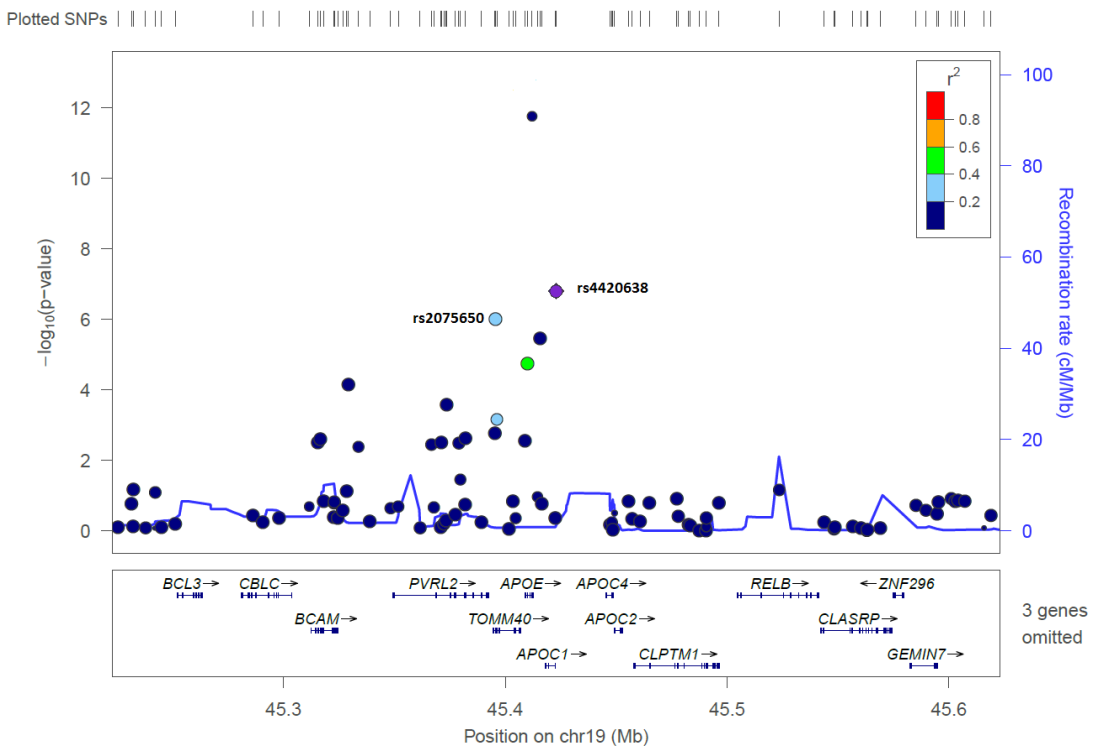


Figure 4.13 Regional plot around the APOE gene and SNPs in Linkage Disequilibrium with rs4420638. SNPs are plotted in order of base position along the x-axis against the univariate $-log_{10}$ P-value on the left-hand y-axis. The blue line shows the recombination rate across this region. Colours for each SNP represent the correlation (r^2) between this SNP and the lead SNP in purple.

4.7 The LASSO on the GRAPHIC study

An attempt to apply the LASSO on a full GRAPHIC dataset was made in R (196) using the glmnet package (53). The idea was to apply three tuning parameter selection methods on the dataset; repeated 10-fold Cross-validation, BIC and the permutation method. However two issues arose with this investigation.

The first was the lack of memory to load the dataset into R. To counter this issue the ALICE High Performance Computing Facility at the University of Leicester was used. This resource allowed a much greater memory limit to be used so that the data could be loaded and analysed. The caveat of using the High Performance Computing Facility

is that each analysis must include a time limit and if the analysis goes over this time limit the analysis is aborted. Even at the maximum time limit available (200 hours) this analysis failed to finish and therefore the results were unobtainable. Instead an analysis was performed on a single chromosome from the GRAPHIC study. The analysis should not be performed on each chromosome separately and then combined, as the estimated λ would be different on each chromosome rather than a fixed λ across all datasets which would allow a larger number of variables selected. Fixing the same λ across all dataset is another option; however selecting the λ in this scenario would be difficult. Yi *et al*. also concluded that analyses on each chromosome separately would be "prudent" (15).

4.8 Application of the LASSO on chromosome 19 of the GRAPHIC study

Chromosome 19 was selected for this analysis as there is a number of associations in both the literature (Table 4.5) and the most significant SNPs from the GRAPHIC study were also on this chromosome (Table 4.14 and Figure 4.1). This chromosome consists of 12,376 SNPs and did not cause any problems in terms of computational time taken to analyse the data as it is a smaller subset of the overall dataset. The Bonferroni correction, FDR and LASSO was applied to this chromosome for comparison. The Q-Q plot again showed that there are some associations with LDL on this chromosome (Figure 4.14).

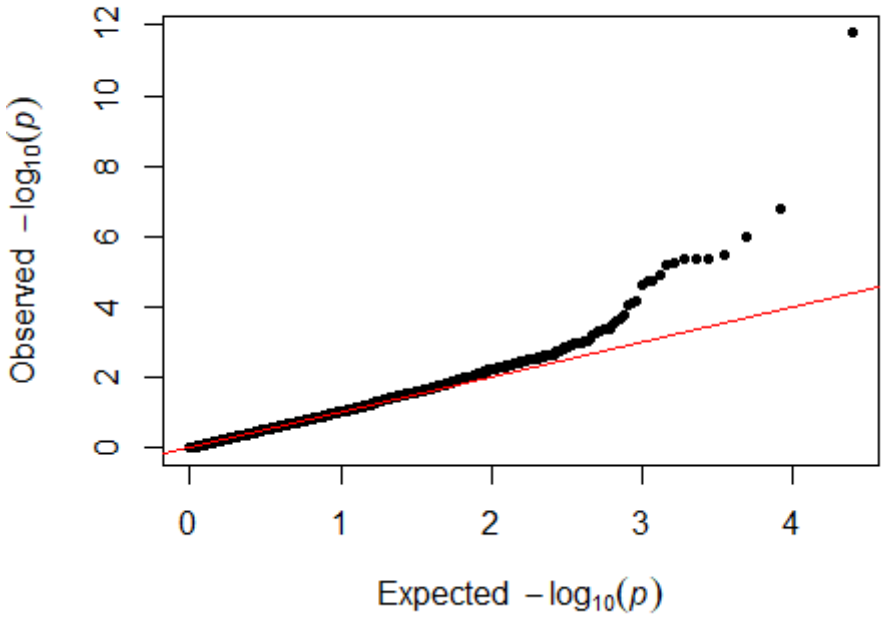


Figure 4.14 Quantile-Quantile plot for P-values from chromosome 19 of the GRAPHIC study. Each SNP's univariate $-log_{10}$ P-value on the y-axis is plotted against the expected $-log_{10}$ P-value under a uniform distribution. The diagonal line in red denotes the expected values the plotted SNPs would take assuming that there are no significant associations with the phenotype.

4.8.1 Methods

The same procedures for Bonferroni correction method and FDR analyses were used as described in sections 4.5.1 and 4.6.1. As the number of SNPs is reduced, the number of tests conducted also decreases leading to different results and most likely more SNPs being selected. The adjusted Bonferroni threshold changes as does the Q-value for each SNP. The Bonferroni adjusted P-value threshold was calculated as $p = 4.04 \times 10^{-6}$. A Q-value threshold of 0.05 was again used.

The glmnet package does not allow any missing values in the dataset therefore imputation was required for the missing genotype data to fit the LASSO. For each SNP with a missing genotype for an individual, median number of minor alleles across the population was imputed. This means that each missing genotype was imputed with the most common genotype in the population. The FDR and Bonferroni correction analyses were performed on the dataset without imputation. In order to check if the results in both datasets were comparable, the P-values were plotted before and after imputation look for any major changes in P-values after imputation. Figure 4.15 shows the scatter plot comparing P-values before and after imputation for all SNPs on chromosome 19. The plot shows that is little difference for the majority of SNPs, especially for the most statistically significant SNPs (Figure 4.16) and therefore it seems reasonable to compare SNPs selected between the LASSO and both the Bonferroni correction and FDR methods. There was little difference when comparing Q-values before and after imputation (Figure 4.17).

Three tuning parameter selection methods were used; repeated 10-fold Cross-validation, BIC and the permutation method. Both repeated CV and the permutation was repeated 100 times for greater accuracy and the median of these λ estimates would be selected as the optimum λ. Repeated CV used a range of 200 λ estimates. While the BIC a range of 625 λ estimates with an interval of 0.01 between each estimate.

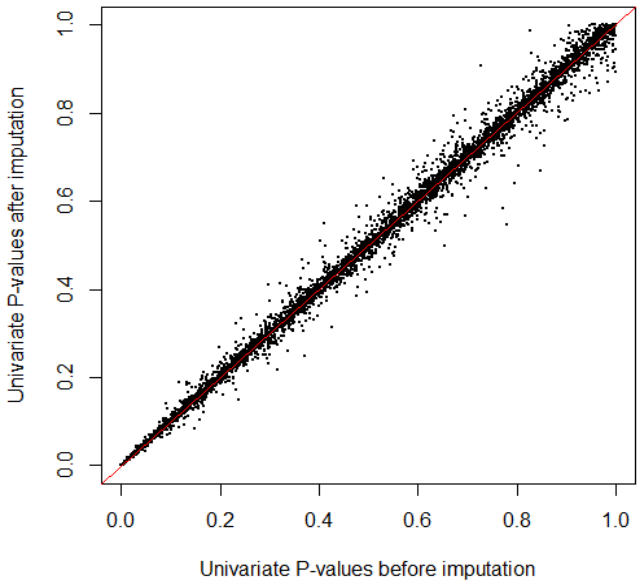


Figure 4.15 Scatter plot comparing P-values for each SNP on chromosome 19 before and after imputation. Imputation was conducted by replacing missing genotype with the median genotype from the population. The red diagonal line represents the line if there is no change in P-values.

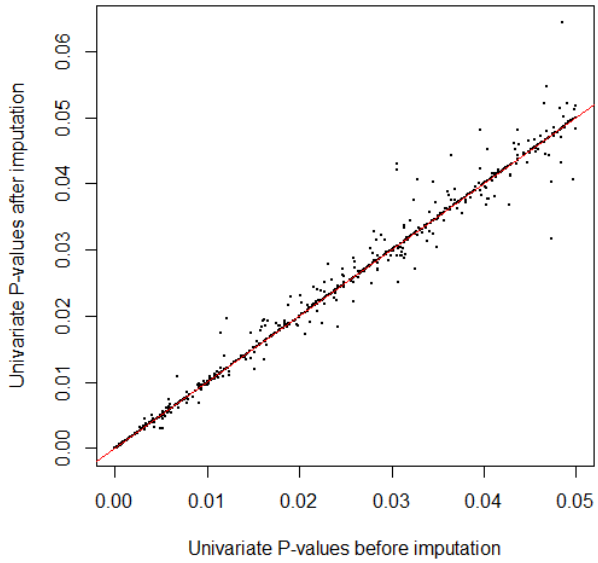


Figure 4.16 Scatter plot comparing P-values for each SNP on chromosome 19 before and after imputation for P-values ≤ 0.05 before imputation. Imputation was conducted by replacing missing genotype with the median genotype from the population. The red diagonal linerepresents the line if there is no change in P-values.

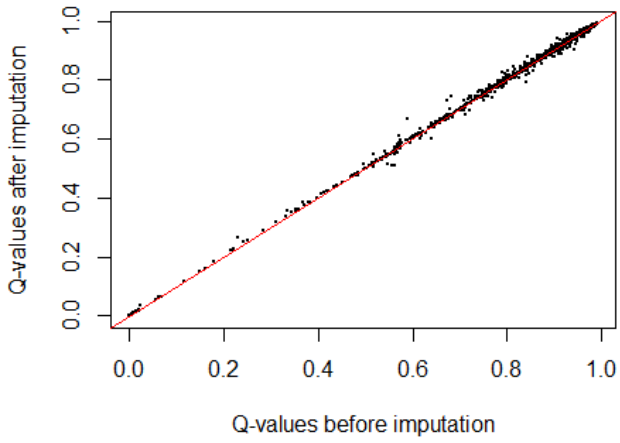


Figure 4.17 Scatter plot comparing Q-values for each SNP on chromosome 19 before and after imputation. Imputation was conducted by replacing missing genotype with the median genotype from the population. The red diagonal linerepresents the line if there is no change in Q-values.

4.8.2 Results

4.8.2.1 Bonferroni correction

Table 4.16 shows the SNPs selected by the Bonferroni correction method using the adjusted P-value threshold of 4.04×10^{-6}. The four SNPs that selected; rs7412, rs4420638, rs2075650 and rs445925 are all located in the same region on APOE gene. As shown in Figure 4.12 and Figure 4.13, r27412 and rs4420638 are independent signals. However the figures show there is correlation between these two SNPs and the two other SNPs selected. There is a high correlation between rs7412 and rs445925 ($r^2 = 0.712$, Figure 4.12) and also some correlation between rs4420638 and rs2075650 ($r^2 = 0.416$, Figure 4.13). These correlations with the top two SNPs in Table 4.16 aid the latter two SNPs to become more statistically significant associated with LDL.

Table 4.16 SNPs selected by the Bonferroni correction method on chromosome 19 of the GRAPHIC study.

SNP	Base position	Beta	S.E.	MAF	P-value	Q-value
rs7412	45412079	-16.28	2.28	0.0861	1.70E-12	2.09E-08
rs4420638	45422946	8.01	1.52	0.2106	1.58E-07	0.000968
rs2075650	45395619	8.45	1.71	0.1573	9.66E-07	0.003951
rs445925	45415640	-9.37	2.00	0.1144	3.37E-06	0.007255

4.8.2.2 False discovery rate

The FDR analysis performed on chromosome 19 selected 13 SNPs (Table 4.17). Two new regions were identified in this analysis. The first is a region between ZNF520 and ZNF567 on chromosome 19, selected six SNPs (rs2967442, rs1035777, rs1525133, rs10402182, rs1727386 and rs2967440) across a region of around 135kb (Figure 4.18). This region has not been identified by previous studies. There is little difference

between the effect size (beta), MAF, P-values and Q-values in all six SNPs (Table 4.17) which shows that there is high LD between them. The correlation between rs1525133, rs10402182 and rs1727386 was estimated as $r^2 = 1$ and therefore were in perfect correlation with each other. Given that there are multiple SNPs identified within these regions, it can be seen that neither the Bonferroni correction nor the FDR methods are unable to handle correlated SNP data. Generally the top SNP by P-value in a region tends to be selected as the associated SNP in any GWAS dataset although it may not be the causal SNP; however it becomes more difficult in the region between ZNF529 and ZNF567 on chromosome 19 as there are 3 SNPS with the same P-value (Table 4.17).

Two further SNPs were identified between the DNM2 and CARM1 genes (Figure 4.19). This region has also not been identified in previous studies however this region is approximately 200kb from the LDLR gene which has shown a number of associations in the literature (Table B.0.1). There was little statistical significance on the LDLR gene in this study however (Figure 4.19).

Table 4.17 SNPs selected by false discovery rate on chromosome 19 of the GRAPHIC study.

SNP	Base position	Beta	S.E.	MAF	P-value	Q-value
rs7412	45412079	-16.28	2.28	0.0861	1.70E-12	2.09E-08
rs4420638	45422946	8.01	1.52	0.2106	1.58E-07	0.000968
rs2075650	45395619	8.45	1.71	0.1573	9.66E-07	0.003951
rs445925	45415640	-9.37	2.00	0.1144	3.37E-06	0.007255
rs10402182	37160529	6.29	1.36	0.3013	4.53E-06	0.007943
rs1525133	37199250	6.29	1.36	0.3013	4.53E-06	0.007943
rs17272386	37180297	6.29	1.36	0.3013	4.53E-06	0.007943
rs2967442	37064240	6.23	1.37	0.3008	5.87E-06	0.008707
rs1035777	37094435	6.19	1.37	0.3011	6.64E-06	0.00905
rs17001002	10948031	-6.97	1.59	0.1839	1.25E-05	0.014524
rs769449	4541,0002	7.76	1.80	0.1386	1.79E-05	0.018291
rs11881156	10950125	-6.92	1.60	0.1855	1.79E-05	0.018328
rs2967440	37059215	5.84	1.37	0.3046	2.33E-05	0.021957

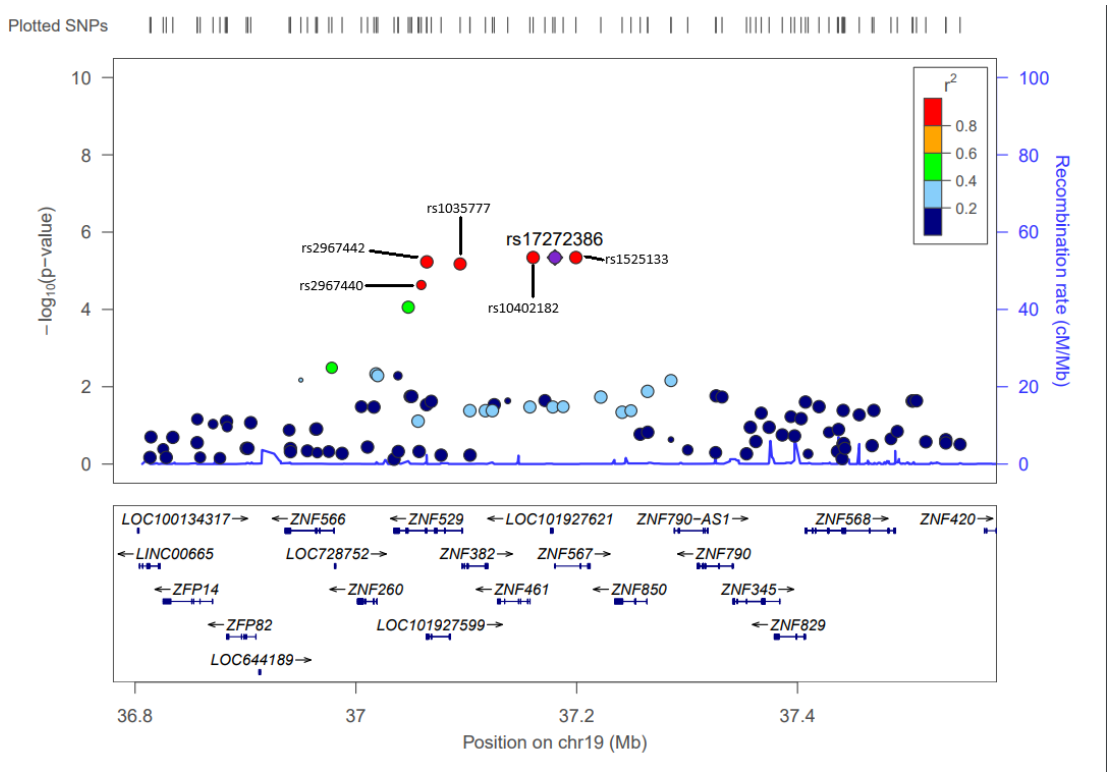


Figure 4.18 Regional plot of identified region between the ZNF529 - ZNF567 genes. SNPs are plotted in order of base position along the x-axis against the univariate $-log_{10}$ P-value on the left-hand y-axis. The blue line shows the recombination rate across this region. Colours for each SNP represent the correlation (r^2) between this SNP and the lead SNP in purple.

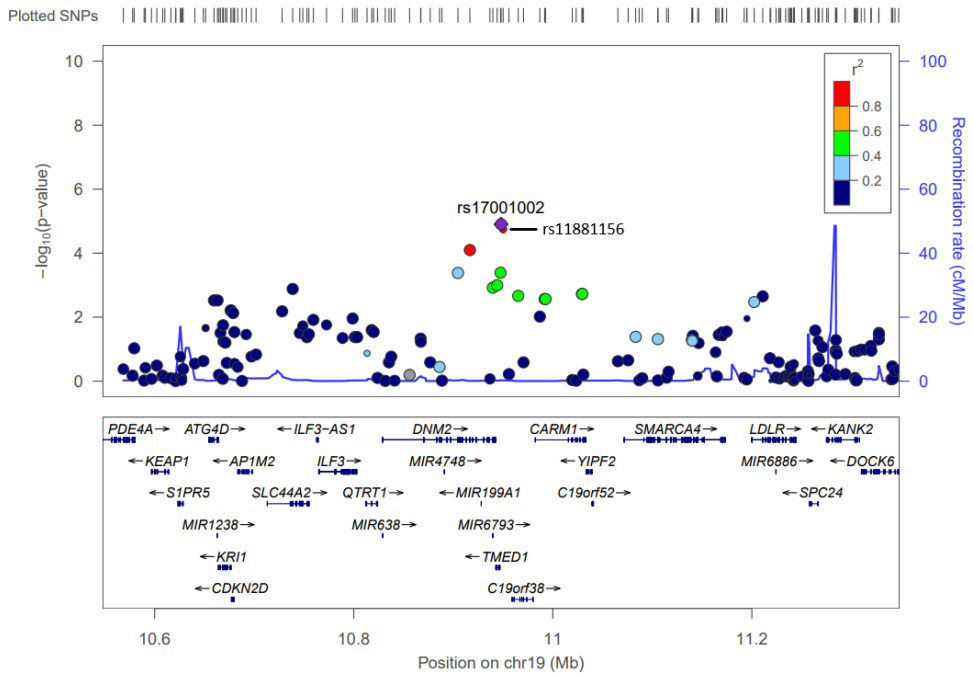


Figure 4.19 Regional plot of identified region between the DNM2 – CARM1 genes. SNPs are plotted in order of base position along the x-axis against the univariate $-log_{10}$ P-value on the left-hand y-axis. The blue line shows the recombination rate across this region. Colours for each SNP represent the correlation (r^2) between this SNP and the lead SNP in purple.

4.8.2.3 LASSO

The SNPs selected by repeated 10-fold CV on chromosome 19 are shown in (Appendix C). The tuning parameter estimates varied between 2.058 (selecting 22 SNPs) and 2.655 (selecting 85 SNPs) with mean = 2.389 (S.D = 0.12) and median = 2.365 (Figure 4.20). A total of 44 SNPs were selected on chromosome. This is unsurprising as shown in section 3.3.2.1 CV tends to select a large model with a number of false positives (87,91,94-96).

Previous studies have shown the correlations between SNPs may able to accommodate for LD between SNPs (24-26) where the Bonferroni correction and FDR methods are unable to. There is evidence of this on the GRAPHIC study. For example, six SNPs (rs2967442, rs1035777, rs1525133, rs10402182, rs1727386 and rs2967440)

were selected using the FDR method (Table 4.17) the region around ZNF529 – ZNF567 genes (Figure 4.18). Of these six SNPs, three were in perfect LD with each other with another three SNPs in high LD ($r^2 \geq 0.8$). The LASSO selected the three SNPs in perfect LD (rs1525133, rs10402182 and rs1727386) but not the remaining three SNPs ($r^2 < 1$). Further inspection of the three SNPs in perfect LD showed that there was a big difference in β estimates produced by the LASSO at the selected λ. The estimate for rs10402182 was β = 2.314, while the estimates for both rs1525133 and rs1727386 were $\beta < 1x10^{-14}$. This shows that the LASSO selects rs10402182 over the other SNPs as that the beta estimates for these two SNPs are so small and are close to being removed from the model.

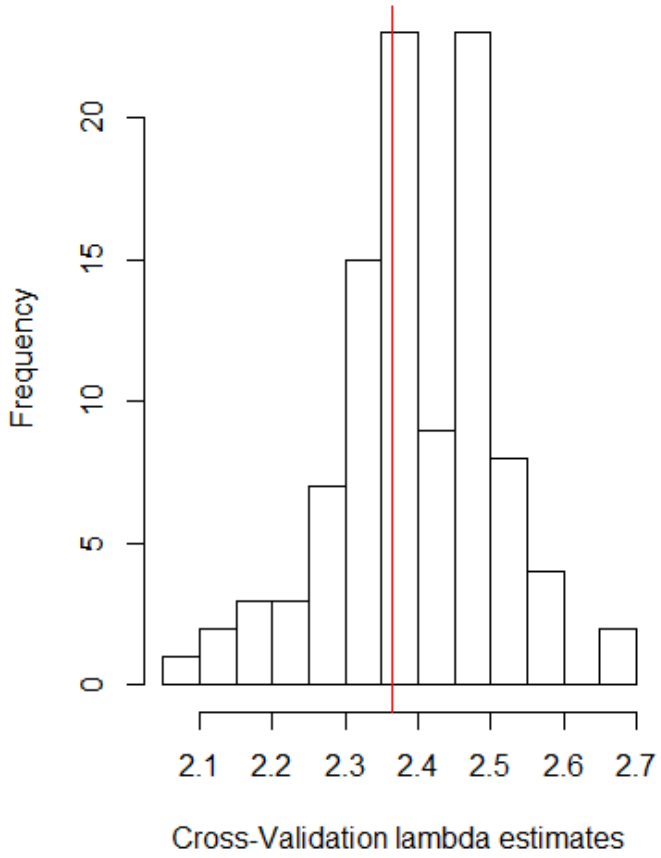


Figure 4.20 Histogram of lambdas estimates using Cross-validation for each of the 100 repetitions. The red vertical line represents the median estimate.

Both the BIC and permutation methods selected the same model of 4 SNPs. The 100 tuning parameter estimates for the permutation method varied between 3.119 (selecting 7 SNPs) and 4.315 (selecting 2 SNPs) with mean = 3.536 (S.D. = 0.26) and median = 3.494 (Figure 4.21). The tuning parameter estimate for the BIC was $\lambda = 3.50$. The SNPs selected were from the four regions selected using the FDR method (Table 4.17). Only one SNP was selected from each region, again showing that the LASSO is able to handle LD by selecting the top association and removing the remaining correlated SNPs.

Table 4.18 SNPs selected by the LASSO using both BIC and 100 repeats of the permutation method for tuning parameter selection

SNP	Base position	Beta	S.E.	MAF	P-value	Q-value
rs7412	45412079	-16.28	2.28	0.0861	1.70E-12	2.09E-08
rs4420638	45422946	8.01	1.52	0.2106	1.58E-07	0.000968
rs10402182	37160529	6.29	1.36	0.3013	4.53E-06	0.007943
rs17001002	10948031	-6.97	1.59	0.1839	1.25E-05	0.014524

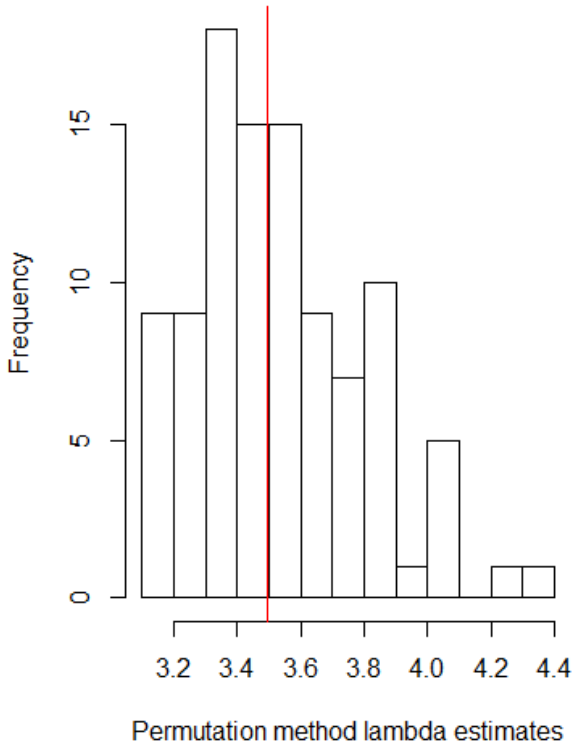


Figure 4.21 Histogram of lambdas estimates using permutation method for each of the 100 repetitions. The red vertical line represents the median estimate.

4.9 Discussion

The aim of this chapter was to apply the LASSO to the GWAS dataset from the GRAPHIC study in order to select associations with Low-density Lipoprotein (LDL-c). However the time taken to fit the model reached time limits using the ALICE High Performance Computing Facility at the University of Leicester, and therefore, was unable to analyse the full dataset due to the computational intensiveness of the process including selection of the tuning parameter. The analysis was not performed genome-wide, but instead on a single chromosome leading to a number of potential associations not being discovered from the dataset.

A way around this issue would be to fit the LASSO on each chromosome individually and combine the results; however this method would be crude as each chromosome varies in the number of SNPs and significant associations, therefore fitting the LASSO individually on each chromosome separately produces 22 analyses that are on different scales to one another and hence it is unlikely to produce the same results as a genome-wide analysis. This was shown when comparing the Bonferroni correction and FDR methods between the genome-wide and chromosome 19 analyses. The Bonferroni correction method selected one SNP on the whole dataset and selected four for the single chromosome analysis. Likewise the FDR method selected two SNPs on the whole dataset and selected thirteen from the single chromosome analysis. It is not known if this increase in the number of selected will occur when comparing with the genome-wide dataset, however if it is the case, this would lead to a large number of variables selected. A high number of associations is unlikely from this dataset as shown literature search where studies with a similar sample size to GRAPHIC generally found a low number of associations (172,173,177).

An alternative approach which has been used in previous studies for high-dimensional data (Table 2.5) would be to reduce the number of SNPs across the whole genome to fit the LASSO in one analysis. This process is known as Pruning and is a logical step given the number of SNPs that are not associated with LDL in the dataset that could be removed without much potential impact on the results. It is also unknown however how pruning would affect LASSO models.

The analysis was performed on the GWAS dataset which consisted of only parental subjects. A number of previous studies that have performed GWAS on single datasets using regression have adjusted for other factors such as age, Body Mass Index (BMI) and sex (162,164,173,174,177-179,181). There was no association with sex in the GRAPHIC study ($p = 0.8845$), there were significant associations with BMI ($p = 2.13 \times 10^{-7}$) and age ($p < 2 \times 10^{-16}$) and could have been adjusted for. For the LASSO, this would require the adjusted variables to be in the regression equation but not included in the

penalty (4.6). Another simpler approach would be to fit a linear model between the phenotype and the variables that will be adjusted for. The residuals can then be predicted from its model and can be used as the phenotype for the LASSO. These residuals would make up the remaining unexplained variance of the phenotype.

$$\hat{\beta}(\lambda) = \frac{1}{2N}\sum_{i=1}^{n}\left(y_i - \mu - \sum_{j=1}^{p} x_{ij}\beta_j - age_i\upsilon - BMI_i\phi\right)^2 + \lambda\sum_{j=1}^{p}|\beta_j| \quad (4.6)$$

where,

υ = the effect estimate of age on the phenotype y
ϕ = the effect estimate of BMI on the phenotype y

Imputation of the dataset was required and was performed by replacing any missing genotypes with the median genotype for that SNP from the dataset. Although this is a crude method, Figure 4.15 and Figure 4.16 both show that there was little difference in P-values before and after imputation, especially for the most statistically significant SNPs. This small difference is mostly due to the quality control procedure where SNPs with a genotype call rate < 97% were removed and therefore the effect of imputation on missing data would be minimal on the results.

Both the Bonferroni and FDR methods selected previously associated SNPs identified in the literature search (Table 4.5), however for the single chromosome analysis both methods selected a greater number of SNPs including some in LD with other selected SNPs. In contrast the LASSO was able to handle the correlation between SNPs and selected mostly independent associations. This was even the case with SNPs in perfect correlation however on close inspection the LASSO selected the first SNP by base position and penalised any other SNP in perfect LD with this SNP. The reasoning for this may be due to the algorithm used to fit the LASSO. The glmnet package uses the coordinate descent algorithm (see section 2.4.2) which updates β estimate a SNP at a time, starting from the first SNP in the dataset, therefore for any pair of SNPs with r^2 =

1, the first SNP by base position will estimate a true β for a given λ but the second will not as the model will have been adjusted for the first SNP. One of the criticisms of the LASSO is that it selects one in a group of highly correlated SNPs and removes the remaining SNPs from the model (18); however this is less of a disadvantage in GWAS. As long as at least one SNP in the associated region is selected it is not a concern if other SNPs in this region are not selected. If other SNPs in the region are of more interest, follow-up analyses can be conducted.

The FDR and LASSO analyses identified two novel regions between ZNF529 and ZNF567 (Figure 4.18) and between DNM2 and CARM1 genes (Figure 4.19). Both these regions have not been previously identified in the literature as associated regions with LDL and require further study.

4.10 Conclusion

In this chapter, analyses were conducted on both a single chromosome and the whole genome from the GRAPHIC study. The results on a single chromosome showed that the Bonferroni correction method to be a conservative method, selecting two regions. Both the FDR and LASSO selected four regions on chromosome 19; however the LASSO was able to handle the LD between SNPs but, the FDR and Bonferroni correction methods were not.

Both the FDR and Bonferroni correction methods on the whole genome selected SNPs, rs7412 and rs4420638, with previously known associations with the phenotype LDL (164,166,172,176,178,180,182,183,187). However the application of the LASSO on the whole genome failed due to computational limits and SNP pruning should be considered to overcome these problems. Although there were problems with the computational time taken, the single chromosome analysis showed that the LASSO may work well in a GWAS setting for variable selection if these problems could be

overcome. With a greater number of variables the LASSO would be recommended over the Bonferroni correction and FDR methods as it is able to remove correlated variables and keep the most statistically important variable.

5 Linkage Disequlibrium estimation

5.1 Introduction

Estimation of LD statistics is the first step when pruning SNPs by LD and therefore is important for future work examining LD SNP pruning. In this chapter, I begin by describing the biological background of LD and how it occurs. I then discuss how LD is estimated in haplotype and genotype data and compare a number of packages that estimate LD. The comparison of different packages is required as they use different algorithms to estimate LD from genotype data which could produce very different results.

5.2 Biology of Linkage Disequilibrium

Linkage Disequilibrium is defined as the non-random statistical association of alleles at different loci (197,198). It occurs due to the co-inheritance of alleles and erodes over generations due to recombination (199). This is illustrated in Figure 5.1. At point (**a**) there are two locus, the first is a polymorphic SNP with respective alleles *A* and *a* the second is a monomorphic SNP. At this point there are only two allele combinations in the population (*A*, *B*) and (*a*, *B*). At some point a mutation may occur on a chromosome at the second locus as shown in green at point (**b**) resulting in a third combination (*A*, *b*) being present in the population. Offspring inherit a pair of chromosomes from the parents with one chromatid inherited from each parent, therefore the alleles on each chromosome become co-inherited. The third combination of alleles also leads to a correlation between the *b* allele and the *A* allele as the presence of the former will always be co-inherited with the latter and therefore producing a statistical association between these alleles.

Over generations the LD may be broken down by recombination. This is where sections of the parental chromatids swap. This is shown at point (**c**) in Figure 5.1, where recombination occurs between the two loci, (*A*, *b*) in green and (*a*, *B*) in blue resulting in the fourth and final combination to be produced (point (**d**)). There are a number of factors that can affect LD in a population which are discussed in greater detail by Slatking which includes natural selection, genetic drift, population subdivisions and inbreeding (198).

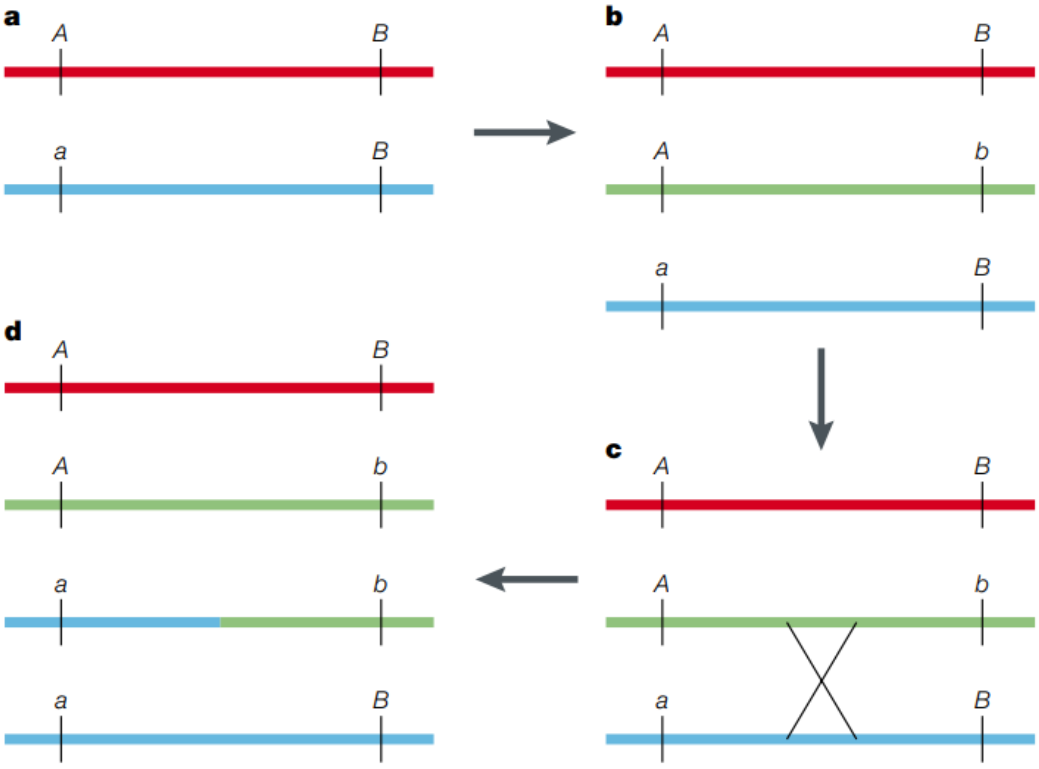


Figure 3 | **The erosion of linkage disequilibrium by recombination. a** | At the outset, there is a polymorphic locus with alleles *A* and *a*. **b** | When a mutation occurs at a nearby locus, changing an allele *B* to *b*, this occurs on a single chromosome bearing either allele *A* or *a* at the first locus (*A* in this example). So, early in the lifetime of the mutation, only three out of the four possible haplotypes will be observed in the population. The *b* allele will always be found on a chromosome with the *A* allele at the adjacent locus. **c** | The association between alleles at the two loci will gradually be disrupted by recombination between the loci. **d** | This will result in the creation of the fourth possible haplotype and an eventual decline in LD among the markers in the population as the recombinant chromosome (*a*, *b*) increases in frequency.

Figure 5.1 The erosion of linkage disequilibrium by recombination taken from Ardle *et al*.(199).

5.3 Linkage Disequilibrium measures and estimation

There are two measures commonly used to calculate linkage disequilibrium, r-squared (r^2) and D-prime (D'). Both measures are standardised measures for the difference between expected and observed haplotype frequencies. *D* and *r* can be interpreted as the covariance and the correlation between loci and across gametes (200). The r-squared statistic is particularly popular as it is related to the statistical power to detect disease associations (201). Both measures are calculated by comparing observed and expected haplotype frequencies. A haplotype is defined as a group of alleles inherited together from a single parent, as LD calculations are based on a pair of SNPs a haplotype is considered for a pair of SNPs rather than a group. The calculation is made based on the deviation statistic (D) between the expected and actual haplotype frequencies is then standardised to produce these measures.

Let $SNP\ A$ and $SNP\ B$ be a pair of SNPS with respective alleles A_1, A_2, B_1 and B_2. Haplotype (z_{11}, z_{12}, z_{21} and z_{22}) and allele frequencies (p_1, p_2, q_1 and q_2) can be calculated from the data (Table 5.1 and Table 5.2)

Table 5.1 Definition of haplotype frequencies for two SNPs with two alleles

Haplotype	Frequency
$\boldsymbol{A_1B_1}$	z_{11}
$\boldsymbol{A_1B_2}$	z_{12}
$\boldsymbol{A_2B_1}$	z_{21}
$\boldsymbol{A_2B_2}$	z_{22}

Where,

$$\sum_{j=1}^{2}\sum_{i=1}^{2} z_{ij} = 1$$

Table 5.2 Definition of allele frequencies based on haplotype frequencies.

Allele	Frequency
$\boldsymbol{A_1}$	$p_1 = z_{11} + z_{12}$
$\boldsymbol{A_2}$	$p_2 = z_{21} + z_{22}$
$\boldsymbol{B_1}$	$q_1 = z_{11} + z_{21}$
$\boldsymbol{B_2}$	$q_2 = z_{12} + z_{22}$

With these allele and haplotype frequencies the deviation statistic (D) can be obtained, by calculating the difference between the expected and actual frequencies. The expected haplotype frequencies can be calculated by multiplying the two respective allele frequencies together (5.1).

$$Expected\ haplotype\ frequency\ E(A_iB_j) = p_i\, q_j \tag{5.1}$$

The expected haplotype frequency assumes the SNPs are in Linkage equilibrium, hence there is no statistical correlation between the combinations of alleles to form haplotypes. The deviation statistic can then be calculated by finding the difference between the haplotype frequency and expected haplotype frequency (Table 5.3) or alternatively the statistic can directly be calculated from haplotype frequencies (5.3).

Table 5.3 Relationship between haplotype frequencies, allele frequencies and the deviation statistic

Allele	$\boldsymbol{A_1}$	$\boldsymbol{A_2}$	Total
$\boldsymbol{B_1}$	$z_{11} = p_1q_1 + D$	$z_{21} = p_2q_1 - D$	q_1
$\boldsymbol{B_2}$	$z_{12} = p_1q_2 - D$	$z_{22} = p_2q_2 + D$	q_2
Total	p_1	p_2	1

$$D = z_{11} - p_1 q_1$$

$$= z_{11} - (z_{11} + z_{12})(z_{11} + z_{21})$$

$$= z_{11}(1 - z_{11} - z_{12} - z_{21}) - (z_{12} z_{21})$$

$$D = z_{11} z_{22} - z_{12} z_{21} \quad (5.2)$$

If the SNPs are in linkage equilibrium, it is expected that there is no difference between the expected and actual frequencies ($D = 0$). If a pair of SNPs are in LD then $D \neq 0$. The r-squared statistic can then be calculated by dividing D with the multiplication of the square root of the four allele frequencies p_1, p_2, q_1 and q_2 and squaring (5.3).

$$r^2 = \left(\frac{D}{\sqrt{p_1 p_2 q_1 q_2}} \right)^2 \quad (5.3)$$

The D-prime statistic also uses the deviance statistic (5.2) and its calculation is dependent on the sign of the deviation statistic (5.4).

$$D' = \frac{D}{D_{min}} \quad (5.4)$$

where

$$\begin{cases} D_{min} = min(p_1 q_2, p_2 q_1) & if\ D < 0 \\ D_{min} = min(p_1 q_1, p_2 q_2) & if\ D > 0 \end{cases}$$

While both r-squared and D-prime are standardised measures for the difference between expected and observed haplotype frequencies (*D*), they have different interpretations. r^2 is a measure of the correlation between haplotypes whereas D-prime measures the largest covariance between a pair of haplotypes.

Variance Inflation factor (VIF) is another method to calculate the correlation between SNPs. In a linear regression analysis, SNPs with a high LD will produce an inflated variance due to the correlation. The variance of each correlated predictor variable is

inflated by a factor shown in (5.5). The R^2 statistic is based on the coefficient of determination calculated by linear regression of a SNP onto another SNP and not the r^2 statistic shown in (5.3). VIF values range between 1 and ∞. A $VIF = 1$ is obtained when $R^2 = 0$ and hence SNPs are in Linkage Equilibrium.

$$VIF = \frac{1}{1 - R^2} \tag{5.5}$$

5.4 Linkage Disequilibrium estimation from genotype data

While estimation by haplotypes will produce the true LD measure, in reality estimation by genotypes is more commonly used, as it is technically very difficult and expensive to measure alleles on a single chromosome and therefore genotype measure are used instead. Datasets in genotype form are such that each SNP for an individual is assigned a number; 0, 1 or 2 depending on its allelic count of the minor allele. LD estimation between SNPs cannot be estimated directly using genotype data as the haplotypes are unknown. If the allele count is 0 or 2 the alleles on each chromatid are known. A 0 count on an individual at a particular SNP would mean the individual has a major allele on each chromatid, where a count of 2 means an individual has a minor allele on each chromatid. However, with a genotype count of 1 it is unclear which chromatid contains the major and minor alleles. Therefore if a pair of SNPs both have a genotype count of 1, it will be unclear whether the same parental chromatid contains both contain the major or minor allele or if each chromatid contains one major and minor allele respectively. The estimation of haplotypes from genotypes is known as phasing. Phasing is also required for imputation of missing genotype data in GWAS studies (202,203).

There have been a number of methods proposed to estimate haplotypes from genotype data which include Clark's algorithm (204), the Expectation-Maximisation

(EM) algorithm (205,206) and hidden Markov Models (207) which require iterative approaches. Browning and Browning discuss these methods, and the software that implement these algorithms in detail (208). Clayton and Leung also proposed a numerical approach for LD estimation from genotypes, which is implemented in the R package snpStats (209).

5.5 Comparison of R packages that estimate LD between SNPs

5.5.1 R functions that calculate Linkage Disequilibrium

Three commonly used packages for genetic analyses in R that can calculate LD statistics between multiple SNPs were used; GenABEL (210), genetics (211) and snpStats (212). These packages are not specifically designed for LD calculations but all include functions that calculate LD for genetic markers. All three packages estimate LD by genotypes rather than haplotypes. GenABEL uses an Expectation-Maximisation (EM) algorithm as described by Hao and Crawley (206). The genetics package uses "maximum likelihood estimation" to estimate LD and snpStats uses a numerical approach (209).

All three R packages are able to calculate pairwise LD statistics based on both r-squared and D-prime statistics for large numbers of SNPs. The GenABEL package has two separate functions to calculate these statistics; `r2fast` for r-squared statistics and `drpfast` for D-prime statistics. These functions return a P x P matrix; where P is the number of SNPs defined in the dataset. The matrix contains two separate readings; the r-squared or D-prime statistics are stored above the diagonal in the matrix and the numbers of SNP genotypes measured for both SNPs that have been used to calculate the LD statistic are stored below the diagonal.

The genetics package uses an `LD` function to calculate LD between SNPs. The P x P matrix returned stores the LD statistics above the diagonal and the cells below the diagonal are set to missing. The function does not include any options and only requires the input of a dataset. The dataset can be in the form of allele pairs (i.e. A\T) or genotype form (i.e. 0, 1 and 2). For any user-specified dataset the genetics package automatically calculates and stores both r-squared and D-prime estimates for any user supplied set of genotypes. The user must then extract the desired LD statistic. Users can return a number of other statistics from this function that include a correlation coefficient, the number of observations or a chi-squared test for linkage disequilibrium which tests the hypothesis of a pair of SNPs in linkage equilibrium (D-prime = 0) against the pair of SNPs in linkage disequilibrium (D-prime ≠ 0) and a P-value from the chi-squared test. The chi-squared tests are applied to D-prime statistics and not r-squared statistics. All these statistics are automatically calculated by the package regardless of if they are required or not.

The snpStats package includes an `ld` function to calculate LD statistics. There is a greater choice in LD measures using snpStats that include log likelihood ratio, odds ratio, Yule's Q statistic, covariance, and r as well as the more commonly used measures of D-prime and r-squared. The function also includes a “depth” option which is a numeric argument that forces the function to only calculate LD statistics across a certain number of adjacent SNPs from any SNP. This option essentially creates an LD window of adjacent SNPs in the LD matrix, similar to a pruning window implemented in PLINK. Both the GenABEL and genetics packages do not include this option, nor do they include a wider variety of LD measures to choose from.

5.5.2 Methods to compare the R functions

To compare the estimated LD statistics between the three packages and PLINK, the first 2,000 SNPs on chromosome 1 from 1,014 parents in the GRAPHIC GWAS dataset were taken (see section 4.3). These 2,000 SNPs span over 7.8Mb and give 1,999,000 pairwise LD estimations for comparison. The quality control procedures applied were the same as those in section 4.4 (low call rate in individuals < 95%, low call rate in SNPs < 97%, MAF < 2%, HWE < 0.0001).

PLINK version 1.07 (19,20) was used as a baseline comparison for the LD statistics calculated in the R packages. PLINK can calculate LD statistics by both haplotypes and genotypes; however calculation by haplotypes can only be applied when calculating LD between a single pair of SNPs and not multiple pairs of SNPs, therefore an LD matrix cannot be produced from the estimation of haplotypes. Calculation from genotype data is implemented using an EM algorithm. PLINK does not have an option to calculate an LD matrix for D-prime statistics, though this option can be used when calculating by haplotypes. Only R and r-squared LD matrices can be calculated from genotypes. The PLINK command also includes options to restrict the number of LD calculations to a specific window size. The window can be implemented by either a number of adjacent SNPs or genetic distance (kb). There is also an option that returns an LD matrix that only report LD statistics above a user-supplied threshold, the remaining statistics are set to 0.

For each R package an LD matrix was calculated for both r-squared and D-prime measures. An LD matrix from PLINK was obtained using the --r2 --matrix command. No LD window was implemented for this analysis; LD statistics for all pairwise combinations were calculated. The r-squared statistics from each package were plotted against the statistics calculated in PLINK to show the variation in estimates between packages. PLINK does not calculate LD statistics by D-prime and therefore the statistics calculated in each package were plotted against each other as a comparison.

To compare the computational time taken to calculate the LD matrix between the three packages and PLINK, the same dataset of SNPs on chromosome 1 from 1,014 parents in the GRAPHIC study was used. The size of the dataset varied between the first 2,000 SNPs to the first 20,000 SNPs on chromosome 1. Five datasets were used with totals of 2,000, 5,000, 10,000, 15,000 and 20,000 SNPs respectively. This was to gauge the effect on computational time as the number of SNPs increases. The quality control procedures applied were again the same as the previous chapter (see section 4.3). LD matrices were calculated using the same commands as the analysis comparing LD statistics. The computational time taken was recorded in hours, minutes and seconds.

5.5.3 Comparison of Linkage Disequilibrium statistics

Figure 5.2, Figure 5.3 and Figure 5.4 show scatter plots of the estimated LD statistics between PLINK and the three packages: GenABEL, genetics and snpStats. While the LD values between PLINK and both genetics and snpStats seem fairly similar and consistent (Figure 5.3 and Figure 5.4), with the snpStats estimate being slightly more accurate than genetics. The algorithm implemented in GenABEL tends to overestimate r^2 statistics compared to PLINK (Figure 5.2); however the overestimation was relatively consistent in all pairwise LD statistics. The r-squared statistics between the genetics and snpStats packages were almost identical (Figure 5.5).

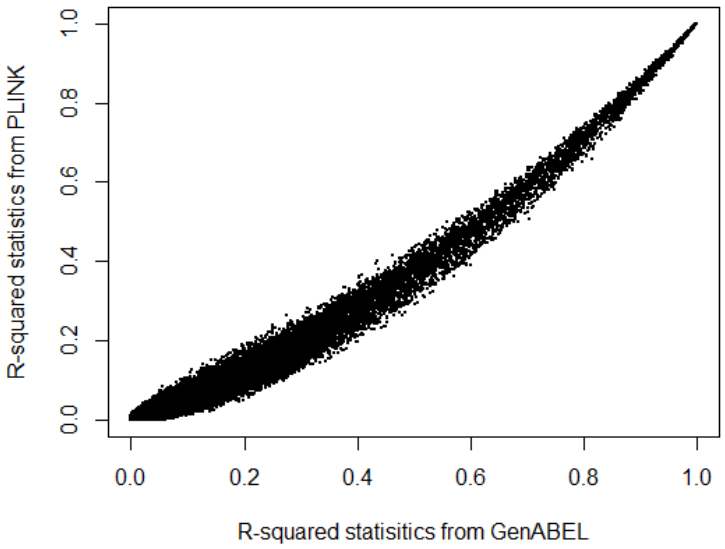


Figure 5.2 Comparison of r^2 statistics between PLINK and GenABEL

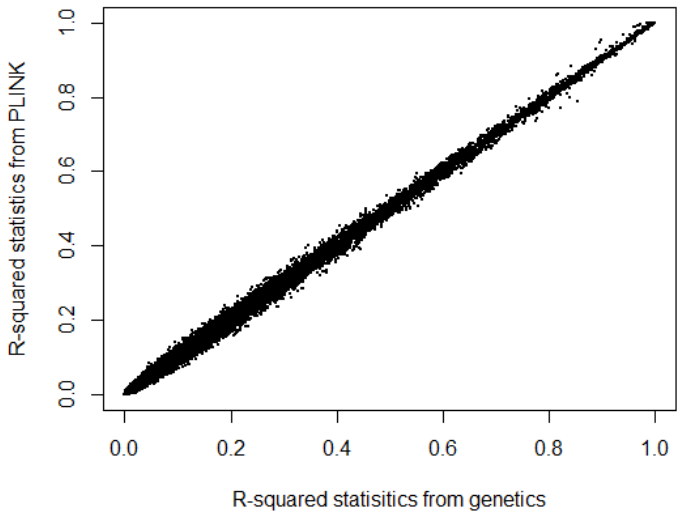


Figure 5.3 Comparison of r^2 statistics between PLINK and genetics

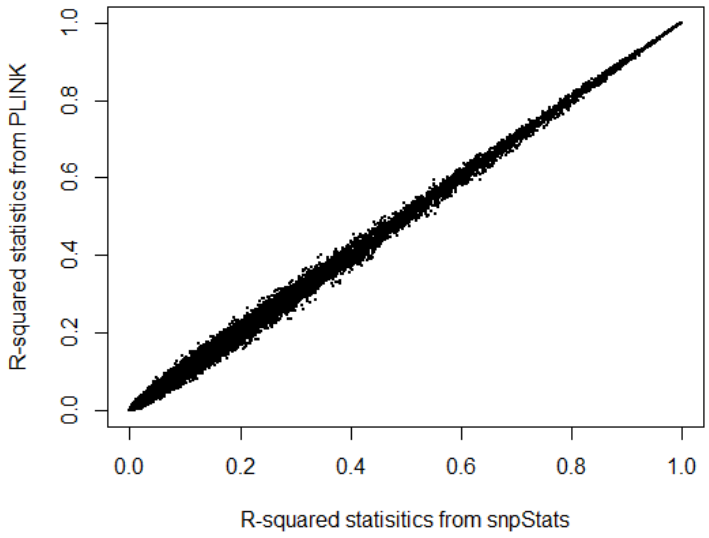


Figure 5.4 Comparison of r^2 statistics between PLINK and snpStats

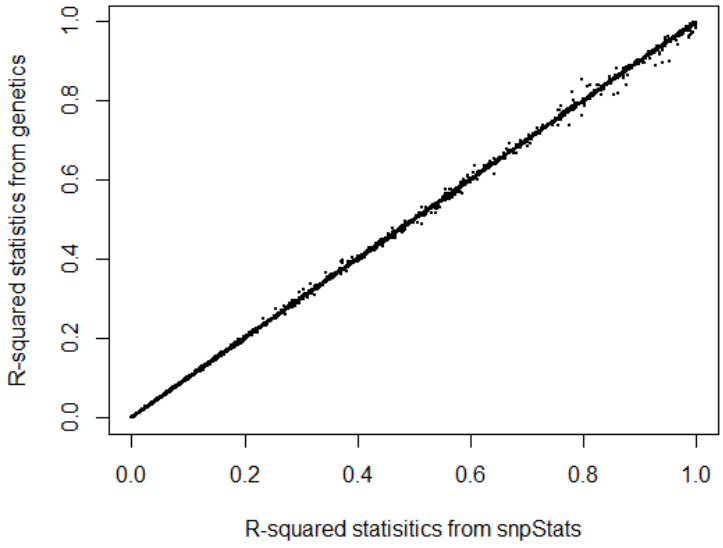


Figure 5.5 Comparison of r^2 statistics between genetics and snpStats

When comparing the D-prime matrices between the three R packages, there were again discrepancies between GenABEL and both the genetics (Figure 5.6) and snpStats (Figure 5.7) packages. Unlike the r-squared statistics where there was a

slight but consistent overestimation of LD statistics, there was no consistency in D-prime statistics. There was both under and over estimation of D-prime statistics in the GenABEL package compared to genetics and snpStats. There is a high amount of variability in the overestimated SNPs with some pairs of SNPs that are estimated with D-prime = 0 in GenABEL estimated with D-prime = 1 in genetics and snpStats. There is a smaller variability in the underestimated SNPs; the extreme difference in D-prime estimates is 0.35 (estimated as D-prime = 0 in both genetics and snpStats and D-prime ≈ 0.38 in GenABEL). The GenABEL help file does acknowledge that there is a difference between itself and the genetics package in both LD statistics but does not explain why there is this difference. D-prime statistics between genetics and snpStats were similar as they were for r-squared (Figure 5.8).

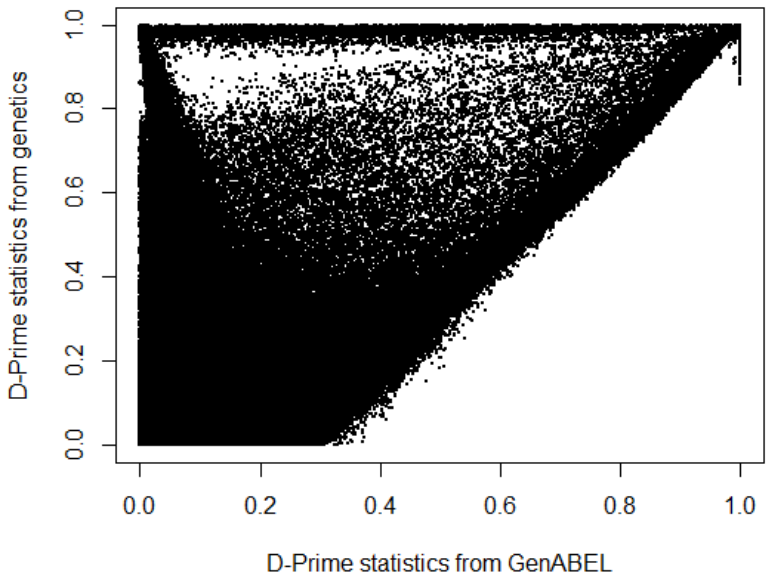


Figure 5.6 Comparison of D-prime statistics between genetics and GenABEL

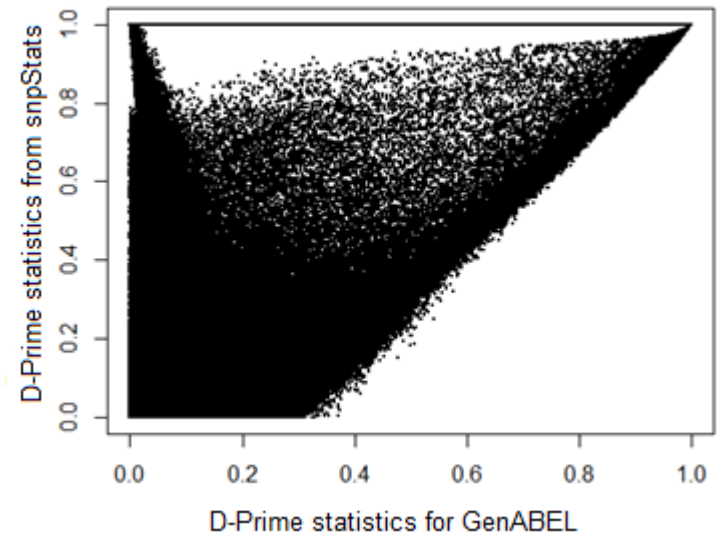


Figure 5.7 Comparison of D-prime statistics between snpStats and GenABEL

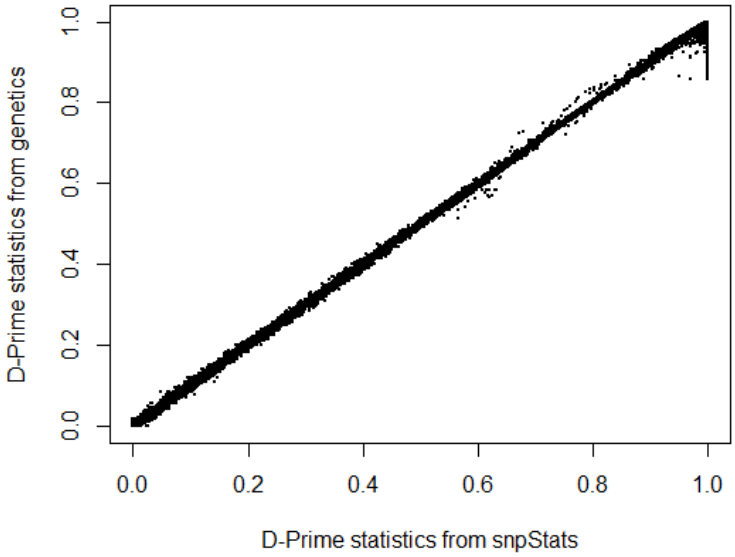


Figure 5.8 Comparison of D-prime statistics between genetics and snpStats

5.5.4 Comparison of time taken to compute Linkage Disequilibrium estimates

Table 5.4 shows the time taken to compute different size LD matrices with both r-squared and D-prime measures. The snpStats package was shown to be consistently faster to calculate these matrices for both measures, which is unsurprising considering the snpStats package is the only package that does not use the EM algorithm which requires a number of iterations to converge to the true LD estimate. The GenABEL package was faster than PLINK in calculating r-squared statistics. The genetics package took the longest to compute LD statistics with datasets of 10,000 SNPs or more reaching the computer time limit before the matrix was calculated. It is unsurprising that the genetics package takes so much computational time as the function calculates a number of statistics including both r-squared and D-prime statistics. The timings between r-squared and D-prime measures for each package were similar.

Table 5.4 Time taken (hh:mm:ss) to compute different size LD matrices in PLINK, GenABEL, genetics and snpStats packages

Time taken to calculate an LD matrix in hours, minutes and seconds										
No. of SNPs	**2,000**		**5,000**		**10,000**		**15,000**		**20,000**	
LD Measure	**r^2**	**D'**	**r^2**	**D'**	**r^2**	**D'**	**r^2**	**D'**	**r^2**	**D'**
PLINK	00:04:15	-	00:23:36	-	01:30:10	-	03:24:00	-	05:47:46	-
GenABEL	00:00:33	00:00:32	00:01:55	00:01:55	00:07:31	00:07:32	00:16:47	00:16:50	00:29:48	00:29:54
genetics	10:08:50	10:24:58	64:48:34	64:31:04	> 200 hrs	> 200 hrs	> 200 hrs	> 200 hrs	> 200 hrs	> 200 hrs
snpStats	00:00:14	00:00:14	00:00:51	00:00:52	00:02:39	00:02:46	00:05:24	00:05:25	00:08:59	00:09:06

5.5.5 Conclusion

In comparing R packages to calculate an LD matrix, both accuracy of estimates compared to PLINK and computational time were considered. Both the snpStats and genetics packages showed good accuracy in LD statistics for the r-squared measure when compared to PLINK (Figure 5.3 and Figure 5.4) and when compared to each other for both statistics (Figure 5.5 and Figure 5.8). However the GenABEL package tended to overestimate r-squared statistics (Figure 5.2) and showed large inconsistencies in D-prime statistics compared to the other R packages (Figure 5.6 and Figure 5.7). snpStats was computationally the quickest of all the packages to calculate both types of LD measures while genetics was computationally the longest. The function took over 200 hours for a dataset of 10,000 SNPs compared to 7 minutes 30 second in snpStats. Other advantages of using snpStats include the ability to prune by alternative LD measures such as log likelihood ratio, odds ratio, Yule's Q, covariance, and R as well as the ability to use the “depth” option that can be used to implement a pruning window in adjacent SNPs.

5.6 Summary

This chapter has introduced the biological background and calculation of LD statistics. These LD statistics can then be used for pruning SNPs from a dataset which is discussed in greater detail in Chapter 5. I tested a number of packages that estimate the LD measures D' and r^2 and found that the snpStats package showed superior performance in terms of both estimation accuracy and computational time taken.

6 SNP pruning

6.1 Introduction

SNP Pruning is a quality control procedure that removes a number of SNPs from the dataset. There are a number of reasons that pruning is used such as removing SNPs in LD so that a dataset of independent SNPs remains or saving computational time. In section 4.7, I described how due to the lack of computational time and memory I was unable to perform a GWAS using the LASSO on the GRAPHIC GWAS dataset. The computational intensity of the method requires the number of SNPs to be controlled. This is likely to be the case for larger GWAS datasets as well as studies that perform meta-analyses or integrative analyses which will combine a number of datasets.

In this chapter, I review a number of current pruning methods and discuss the advantages and disadvantages of each method. I then apply a number of these methods in my own R program. The algorithm uses manipulation of the LD matrix to prune rather than other algorithms that prune combinations of SNPs in a pairwise fashion. I then compare my pruning algorithm to the PLINK pruning program (19,20).

6.2 Linkage Disequilibrium pruning

Datasets are commonly pruned by LD to both reduce the number of dimensions and to remove correlations between SNPs. Principle Component Analysis (PCA) is used to investigate population structure across different ancestries (213-216). Differences in ancestry can lead to confounding by population stratification (217), however clustering in regions of high LD can be difficult as the LD can obscure patterns of population structure (213,218). LD pruning is therefore used as a quality control step by removing regions of high LD in order to perform PCA analyses.

LD pruning is also commonly used in genetic risk scores, also known as polygenic risk scores (219,220). The reason pruning is utilised here is to avoid any duplicated information due to LD in the score however this could lead to causal variants being pruned (219). To avoid such a scenario, some studies now prune SNPs based on P-value, known as LD clumping (221-223). The LD clumping method prunes SNPs based on P-value, allowing the top signals in a dataset to remain after pruning.

There are a three main LD statistics that can be used for pruning, D-prime, r^2 and R^2. Calculations of these statistics are described in section 5.3. If the LD statistic is above a given threshold, one of the pair of SNPs is pruned. Most commonly used pruning programmes implement a pruning window and step size options for pruning (19,20,201,224-226). Both options are designed to reduce time for pruning. A window of length M and a step size H, will take the next F adjacent SNPs and prune only within this window and then move along H SNPs and repeat the process. The pruning algorithm will only prune SNPs within this window. The step option then moves the window by a number of SNPs and repeats until the end of the dataset is reached.

The implementation of a window allows the user to reduce the computational time. A smaller window reduces the number of pairwise LD calculations required to prune the entire dataset; however there may also be a risk that the dataset is not pruned thoroughly as a small window would not cover a dense region of SNPs in high LD of each other. Likewise a large step size would lead to a computationally faster but less thorough pruning of a dataset.

The disadvantage of pruning by LD is that SNPs tend to be pruned without any user control of which SNPs are kept and which are pruned (219). A particularly desirable option that packages do not contain could be to fix certain SNPs, for example, SNPs with previously known associations, to prevent them from being pruned. This would involve a similar approach to LD-clumping without the use of P-values but instead prior knowledge.

6.2.1 Tag SNPs

Genotyping millions of SNPs for an association analysis can be time consuming and costly. To reduce time and money, individual SNPs known as TagSNPs are used to represent dense regions of SNPs in LD and thus every SNP in the region does not require genotyping. Methods in selecting these TagSNPs tend to be based on accounting the LD structure using a reference population (201,227-229), prediction based methods (229) or PCA have also been suggested (230).

6.2.2 Pruning by P-value

As shown in Table 2.5, quite a few studies have used P-value pruning to reduce the dimensionality of the dataset before fitting the LASSO. Unlike LD pruning, pruning a dataset by P-value would ensure that the associated SNPs would be kept and the SNPs with no association would be pruned. This would leave regions of high associations left in the dataset rather than a genome-wide dataset. This is similar to studies that select regions or genes for association tests.

The method guarantees that the most statistically significant variants remain after pruning while reducing the number of SNPs for analysis. Pruning by P-value however will leave a dataset that will consist of highly associated regions and not one that covers the whole genome. Figure 6.1 shows plotted univariate β estimates against their respective P-values for the 591,774 SNPs in the GRAPHIC study (see section 4.4). The plot shows that as the β estimate increases, the P-value of the SNP decreases therefore by pruning by P-value, only the large effects remain and smaller effects will be pruned.

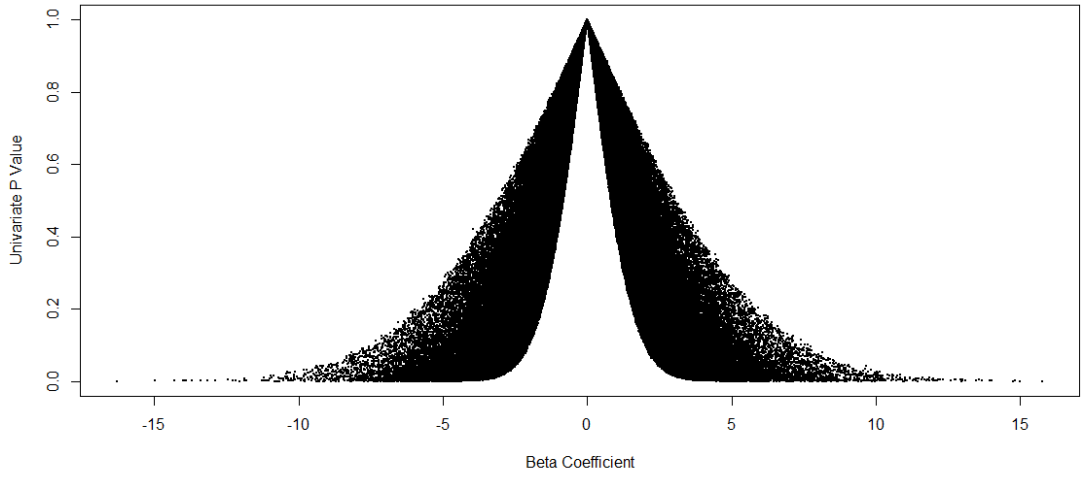


Figure 6.1 Volcano plot showing beta coefficients against P-value of each of the 591,774 SNPs calculated on 979 subjects from the GRAPHIC study with LDL as the phenotype.

6.2.3 Pruning by LD clumping

LD clumping combines both P-value pruning and LD pruning and keeps the advantages of both methods. SNPS are pruned by LD; however the position of the starting SNP is dictated by P-value. The SNP with the smallest P-value is the starting point and SNPs in LD with this top SNP are pruned. Unlike the LD pruning algorithm, next SNP included is not the next adjacent SNP but the next SNP with the smallest P-value that has not been pruned. This continues until the last SNP with the highest P-value is reached. The method ensures an independent set of SNPs while fixing the most statistically significant SNPs in each region. A handful of studies have used this approach to pruning for certain analyses (231,232).

6.3 Current SNP pruning software

6.3.1.1 PLINK

PLINK (20) is the most commonly used SNP pruning program. The program requires 4 options: a sliding window size, a window step size, a desired LD statistic and its pruning threshold. PLINK allows calculation of LD by r^2 (5.3) using the `–indep` option and VIF (5.5) using the `--indep-pairwise` option. Both window and step size are given in terms of numbers of adjacent SNPs. The algorithm used by PLINK is described in Table 6.1. VIF pruning uses the same algorithm but with threshold values > 1. A VIF threshold between 1.5 ($R^2 = 0.33$) and 2 ($R^2 = 0.5$) is suggested in PLINK.

Table 6.1 PLINK algorithm for LD pruning

• Let Window size = M • Let step size = H • Let LD statistic = r^2 • Let LD pruning threshold = T
1. Begin at the first SNP in the first chromosome 2. Take a window of the next M adjacent SNPs 3. Calculate LD between each pair of SNPs in the window 4. For a pair of SNPs in the window with $r^2 > T$, prune the SNP with the lowest MAF. When pairs of SNPs in LD have the same MAF the first SNP is kept and the second is pruned. 5. Move window along by H SNPs (steps) 6. Repeat steps 2-6 until the end of the chromosome 7. Repeat 1-6 for each chromosome

The PLINK pruning algorithm is simple but is computationally slow (Table 5.4) and lacks a number of useful options. D-prime is not an allowed LD measure option for pruning although PLINK can be used to calculate D-prime estimates. For a pair of SNPs in LD, one of a pair is pruned; the SNP with a lower MAF from the pair is pruned and therefore leading to a pruned datasets of mostly common variants. When pairs of SNPs in LD have the same MAF the first SNP is kept and the second is pruned. The choice of which SNP is pruned is systematic rather than at random, repeating the pruning algorithm with the same commands on the same dataset will prune the same SNPs. The program does not allow user selected SNPs to be kept in the dataset.

6.3.1.2 SNPRelate

SNPRelate (225) is an R package primarily designed for principle component analysis on SNP data. The package includes a command called `snpgdsLDpruning` that prunes SNPs by LD. The command also performs some quality control such as removing monomorphic SNPs, removing SNPs with low MAF and remove SNPs by missing rates. The command prunes by D-prime, r and R. The more commonly used measures of r^2 and R^2 are not options. The package uses a sliding window like PLINK but not a step option which is fixed to 1. There is a default option to implement a sliding window by base position rather than adjacent SNPs. The pruning window by genetic distance is particularly advantageous as the window can cover dense regions of SNPs in high LD which a window in of adjacent SNPs may not.

The pruning of SNPs is at random rather than systematic. The pruning algorithm randomly selects starting position on each chromosome for pruning. From this starting position SNPs are pruned to the right until the last SNP in the chromosome is reached and then to the left until the first SNP in the chromosome is reached. For a pair of SNPs in LD, the SNP that is closest to the starting SNP is pruned. Therefore the choice of which SNP is pruned is dependent of the starting SNP which is selected at random.

6.4 My Prune package

In this section, I describe my algorithm for LD pruning, which I have implemented in as an R package called 'prune'. I also describe my other pruning algorithms such as P-value pruning, LD pruning while fixing the top SNPs by P-value and LD clumping. My algorithm is slightly different to that applied by PLINK. The traditional LD pruning method implemented in PLINK selects the first SNP in the genome and removes any adjacent SNPs in LD above the threshold. This process then moves onto the next available SNP and repeats until the last SNP in the genome is reached. My algorithm for LD pruning is novel and requires the manipulation of a P x P LD matrix and attempts to prune all SNPs in a single step rather than continuously repeating for each SNP. The algorithm also implements a range of options including a random starting position for pruning. The algorithm uses the snpStats package to calculate desired LD statistics (212).

6.4.1 The Prune package LD pruning algorithm

The choice of which SNPs are pruned from the data in this algorithm is entirely dependent on the starting position. If the starting position is the first SNP in the dataset then the algorithm prunes along the genome towards the last SNP, similar to PLINK. If the starting position is the last SNP in the dataset then the algorithm prunes in the opposite direction, towards the first SNP. If the starting position is in neither of these then, the algorithm prunes in both directions from the starting position. This leads to different manipulations of the LD matrix for the algorithm. If the starting position is the first SNP in the dataset, the LD matrix required is a half-matrix with the upper diagonal cells containing the LD statistics and both the cells along the diagonal and below set to 0 (see step 8a, Table 6.2). To prune from the last SNP in the dataset, an LD matrix where the cells below the diagonal contain the LD statistics and the upper diagonal and diagonal cells are set to 0 is required (see step 8b, Table 6.2). For a central starting position some manipulation of the LD matrix is required in order for

the algorithm to prune the correct SNPs (see step 8c, Table 6.2). The basic LD pruning algorithm for my Prune package is described below with an illustration of the LD matrix manipulation for a random starting position is described in Table 6.2.

Table 6.2 Instructions for the Prune LD pruning algorithm implementing a random starting position for pruning

• Let P = the number of SNPs in the dataset • Let THRESHOLD = LD pruning threshold, where 0 ≤ THRESHOLD ≤ 1 – 0.5 in this example
1. Choose either a specified or random starting position. Call it START.SNP. 2. Create an LD matrix, with a desired window size and LD measure using snpStats. 3. Reflect this half-matrix to obtain a full P by P LD matrix (snpStats only calculates a half-matrix). 4. Set the diagonal of the matrix to 0. This will prevent SNPs being pruned out due to the LD between a SNP and itself = 1. 5. If the starting position is random, then select a random number between 1 and P and set to START.SNP. 6. If the user has specified the start position, set this to START.SNP. 7. Set the LD statistics for the START.SNP column to 0 (Figure 6.2 where the random starting position is highlighted in green). This will prevent the SNP from being pruned from the dataset. 8. To manipulate the LD matrix for pruning: a. If the START.SNP = 1 then SNPs are be to be pruned to the right only. Therefore the lower triangle of the matrix is set to 0. b. If the START.SNP = P then SNPs are be to be pruned to the left only. Therefore the upper triangle of the matrix is set to 0. c. If 1 < START.SNP < P then the data will be pruned to the right of START.SNP then to the left. The LD matrix is manipulated by: i. Take the upper-right quadrant of the matrix with rows between

1 and START.SNP - 1 and columns between START.SNP + 1 and P and set the LD statistics to 0 (Figure 6.3).

ii. Take the upper-right quadrant of the matrix with rows between 1 and START.SNP - 1 and columns between START.SNP + 1 and P and set the LD statistics to 0 (Figure 6.3).

iii. Take the upper triangle of the upper-left quadrant with rows between 1 and START.SNP - 1 and columns between 1 and START.SNP - 1 and set the LD statistics to 0 (Figure 6.4). Only the LD statistics of the lower triangle are required to prune from START.SNP towards the first SNP (to the left of START.SNP).

iv. Take the lower triangle of the lower-right quadrant with rows between START.SNP + 1 and P and columns between START.SNP + 1 and P set the LD statistics to 0 (Figure 6.5). The LD values of the upper triangle only are required to prune from START.SNP towards P (to the right of START.SNP).

9. Set all cells in the matrix with an LD > THRESHOLD to "NA" (Figure 6.6). This marks the SNPs that will be pruned.
10. Create a vector of 1's called MARK of length P. Each cell represent a SNP and its value will determine if it is pruned or not.
11. To mark SNPs for pruning:
    a. Start at START.SNP.
    b. If the cell in MARK that denotes this SNP is "NA" move to step e. If the cell in MARK = 1 move to step c.
    c. Take the row from the LD matrix that corresponds to this SNP.
    d. If any SNPs in this row are marked as "NA", replace the cell in the corresponding column of MARK with an "NA".
    e. Move onto the next SNP to the right.
    f. Repeat steps b-e until the last SNP in the dataset is reached.
    g. Move to the SNP directly to the left of START.SNP.
    h. If the cell in the corresponding column in MARK that denotes this SNP is "NA" move to step k. If the cell in MARK = 1 move to step i.

i. Take the row from the LD matrix that corresponds to this SNP.

j. If any SNPs are marked as “NA”, replace the cell in the corresponding column of MARK with an “NA”.

k. Move to the next SNP to the left.

l. Repeat steps h-k until the first SNP in the dataset is reached. This will produce a vector of SNPs in MARK with 1’s and NA’s. The SNPs marked with an “NA” are marked for pruning. By skipping SNPs in MARK already marked with and “NA”, SNPs cannot be pruned due to LD with a SNP that has already been marked for pruning (step10b & 10h).

12. Rowbind MARK with the genotype matrix.
13. Remove any columns of SNPs from this matrix with an “NA”. This will prune all marked SNPs.
14. Unbind MARK from the genotype matrix.
15. An LD pruned genotype matrix will remain.

	rs1	rs2	rs3	rs4	rs5	rs6	rs7	rs8	rs9	rs10	rs11	rs12	rs13	rs14	rs15	rs16	rs17	rs18	rs19	rs20
rs1	0	0.40	0.00	0.05	0.03	0.00	0.00	0.04	0.03	0	0.00	0.00	0.00	0.01	0.01	0.01	0.00	0.00	0.00	0.00
rs2	0.40	0	0.05	0.01	0.00	0.05	0.04	0.01	0.02	0	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00
rs3	0.00	0.05	0	0.03	0.16	0.98	0.65	0.13	0.14	0	0.01	0.01	0.01	0.01	0.01	0.00	0.00	0.00	0.14	0.02
rs4	0.05	0.01	0.03	0	0.49	0.03	0.05	0.28	0.29	0	0.03	0.00	0.00	0.03	0.02	0.02	0.00	0.00	0.00	0.00
rs5	0.03	0.00	0.16	0.49	0	0.16	0.24	0.66	0.56	0	0.05	0.02	0.02	0.03	0.03	0.03	0.00	0.00	0.05	0.01
rs6	0.00	0.05	0.98	0.03	0.16	0	0.67	0.13	0.14	0	0.01	0.01	0.01	0.00	0.01	0.00	0.00	0.00	0.14	0.02
rs7	0.00	0.04	0.65	0.05	0.24	0.67	0	0.20	0.14	0	0.01	0.01	0.01	0.00	0.00	0.00	0.00	0.01	0.10	0.03
rs8	0.04	0.01	0.13	0.28	0.66	0.13	0.20	0	0.90	0	0.08	0.00	0.00	0.01	0.01	0.01	0.00	0.01	0.02	0.00
rs9	0.03	0.02	0.14	0.29	0.56	0.14	0.14	0.90	0	0	0.09	0.00	0.00	0.01	0.01	0.01	0.00	0.01	0.02	0.00
rs10	0.00	0.02	0.01	0.02	0.05	0.01	0.01	0.09	0.11	0	0.91	0.00	0.00	0.01	0.01	0.01	0.00	0.04	0.01	0.03
rs11	0.00	0.02	0.01	0.03	0.05	0.01	0.01	0.08	0.09	0	0	0.00	0.00	0.01	0.01	0.01	0.00	0.04	0.01	0.03
rs12	0.00	0.00	0.01	0.00	0.02	0.01	0.01	0.00	0.00	0	0.00	0	1.00	0.05	0.05	0.05	0.00	0.06	0.48	0.10
rs13	0.00	0.00	0.01	0.00	0.02	0.01	0.01	0.00	0.00	0	0.00	1.00	0	0.05	0.05	0.05	0.00	0.06	0.48	0.10
rs14	0.01	0.00	0.01	0.03	0.03	0.00	0.00	0.01	0.01	0	0.01	0.05	0.05	0	0.95	0.99	0.05	0.12	0.08	0.02
rs15	0.01	0.00	0.01	0.02	0.03	0.01	0.00	0.01	0.01	0	0.01	0.05	0.05	0.95	0	0.96	0.05	0.11	0.10	0.03
rs16	0.01	0.00	0.00	0.02	0.03	0.00	0.00	0.01	0.01	0	0.01	0.05	0.05	0.99	0.96	0	0.05	0.12	0.08	0.02
rs17	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0	0.00	0.00	0.00	0.05	0.05	0.05	0	0.06	0.00	0.08
rs18	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0	0.04	0.06	0.06	0.12	0.11	0.12	0.06	0	0.12	0.33
rs19	0.00	0.01	0.14	0.00	0.05	0.14	0.10	0.02	0.02	0	0.01	0.48	0.48	0.08	0.10	0.08	0.00	0.12	0	0.21
rs20	0.00	0.00	0.02	0.00	0.01	0.02	0.03	0.00	0.00	0	0.03	0.10	0.10	0.02	0.03	0.02	0.08	0.33	0.21	0

Figure 6.2 LD matrix of 20 SNPs with the diagonal highlighted in yellow and random starting SNP highlighted in green set to 0.

	rs1	rs2	rs3	rs4	rs5	rs6	rs7	rs8	rs9	rs10	rs11	rs12	rs13	rs14	rs15	rs16	rs17	rs18	rs19	rs20
rs1	0	0.40	0.00	0.05	0.03	0.00	0.00	0.04	0.03	0	0	0	0	0	0	0	0	0	0	0
rs2	0.40	0	0.05	0.01	0.00	0.05	0.04	0.01	0.02	0	0	0	0	0	0	0	0	0	0	0
rs3	0.00	0.05	0	0.03	0.16	0.98	0.65	0.13	0.14	0	0	0	0	0	0	0	0	0	0	0
rs4	0.05	0.01	0.03	0.00	0.49	0.03	0.05	0.28	0.29	0	0	0	0	0	0	0	0	0	0	0
rs5	0.03	0.00	0.16	0.49	0	0.16	0.24	0.66	0.56	0	0	0	0	0	0	0	0	0	0	0
rs6	0.00	0.05	0.98	0.03	0.16	0	0.67	0.13	0.14	0	0	0	0	0	0	0	0	0	0	0
rs7	0.00	0.04	0.65	0.05	0.24	0.67	0	0.20	0.14	0	0	0	0	0	0	0	0	0	0	0
rs8	0.04	0.01	0.13	0.28	0.66	0.13	0.20	0	0.90	0	0	0	0	0	0	0	0	0	0	0
rs9	0.03	0.02	0.14	0.29	0.56	0.14	0.14	0.90	0	0	0	0	0	0	0	0	0	0	0	0
rs10	0.00	0.02	0.01	0.02	0.05	0.01	0.01	0.09	0.11	0	0.91	0.00	0.00	0.01	0.01	0.01	0.00	0.04	0.01	0.03
rs11	0.00	0.02	0.01	0.03	0.05	0.01	0.01	0.08	0.09	0	0	0.00	0.00	0.01	0.01	0.01	0.00	0.04	0.01	0.03
rs12	0.00	0.00	0.01	0.00	0.02	0.01	0.01	0.00	0.00	0	0.00	0	1.00	0.05	0.05	0.05	0.00	0.06	0.48	0.10
rs13	0.00	0.00	0.01	0.00	0.02	0.01	0.01	0.00	0.00	0	0.00	1.00	0	0.05	0.05	0.05	0.00	0.06	0.48	0.10
rs14	0.01	0.00	0.01	0.03	0.03	0.00	0.00	0.01	0.01	0	0.01	0.05	0.05	0	0.95	0.99	0.05	0.12	0.08	0.02
rs15	0.01	0.00	0.01	0.02	0.03	0.01	0.00	0.01	0.01	0	0.01	0.05	0.05	0.95	0	0.96	0.05	0.11	0.10	0.03
rs16	0.01	0.00	0.00	0.02	0.03	0.00	0.00	0.01	0.01	0	0.01	0.05	0.05	0.99	0.96	0	0.05	0.12	0.08	0.02
rs17	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0	0.00	0.00	0.00	0.05	0.05	0.05	0	0.06	0.00	0.08
rs18	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0	0.04	0.06	0.06	0.12	0.11	0.12	0.06	0	0.12	0.33
rs19	0.00	0.01	0.14	0.00	0.05	0.14	0.10	0.02	0.02	0	0.01	0.48	0.48	0.08	0.10	0.08	0.00	0.12	0	0.21
rs20	0.00	0.00	0.02	0.00	0.01	0.02	0.03	0.00	0.00	0	0.03	0.10	0.10	0.02	0.03	0.02	0.08	0.33	0.21	0

Figure 6.3 Set all cells upper-right quadrant of the matrix between the row and column representing the random starting location, highlighted in green, to 0.

	rs1	rs2	rs3	rs4	rs5	rs6	rs7	rs8	rs9	rs10	rs11	rs12	rs13	rs14	rs15	rs16	rs17	rs18	rs19	rs20
rs1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs2	0.40	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs3	0.00	0.05	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs4	0.05	0.01	0.03	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs5	0.03	0.00	0.16	0.49	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs6	0.00	0.05	0.98	0.03	0.16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs7	0.00	0.04	0.65	0.05	0.24	0.67	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs8	0.04	0.01	0.13	0.28	0.66	0.13	0.20	0	0	0	0	0	0	0	0	0	0	0	0	0
rs9	0.03	0.02	0.14	0.29	0.56	0.14	0.14	0.90	0	0	0	0	0	0	0	0	0	0	0	0
rs10	0.00	0.02	0.01	0.02	0.05	0.01	0.01	0.09	0.11	0	0.91	0.00	0.00	0.01	0.01	0.01	0.00	0.04	0.01	0.03
rs11	0.00	0.02	0.01	0.03	0.05	0.01	0.01	0.08	0.09	0	0	0.00	0.00	0.01	0.01	0.01	0.00	0.04	0.01	0.03
rs12	0.00	0.00	0.01	0.00	0.02	0.01	0.01	0.00	0.00	0	0.00	0	1.00	0.05	0.05	0.05	0.00	0.06	0.48	0.10
rs13	0.00	0.00	0.01	0.00	0.02	0.01	0.01	0.00	0.00	0	0.00	1.00	0	0.05	0.05	0.05	0.00	0.06	0.48	0.10
rs14	0.01	0.00	0.01	0.03	0.03	0.00	0.00	0.01	0.01	0	0.01	0.05	0.05	0	0.95	0.99	0.05	0.12	0.08	0.02
rs15	0.01	0.00	0.01	0.02	0.03	0.01	0.00	0.01	0.01	0	0.01	0.05	0.05	0.95	0	0.96	0.05	0.11	0.10	0.03
rs16	0.01	0.00	0.00	0.02	0.03	0.00	0.00	0.01	0.01	0	0.01	0.05	0.05	0.99	0.96	0	0.05	0.12	0.08	0.02
rs17	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0	0.00	0.00	0.00	0.05	0.05	0.05	0	0.06	0.00	0.08
rs18	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0	0.04	0.06	0.06	0.12	0.11	0.12	0.06	0	0.12	0.33
rs19	0.00	0.01	0.14	0.00	0.05	0.14	0.10	0.02	0.02	0	0.01	0.48	0.48	0.08	0.10	0.08	0.00	0.12	0	0.21
rs20	0.00	0.00	0.02	0.00	0.01	0.02	0.03	0.00	0.00	0	0.03	0.10	0.10	0.02	0.03	0.02	0.08	0.33	0.21	0

Figure 6.4 Set the cells in the upper triangle of the upper-left quadrant with rows and columns between the first SNP in the dataset and the random starting location, highlighted in green, to 0.

	rs1	rs2	rs3	rs4	rs5	rs6	rs7	rs8	rs9	rs10	rs11	rs12	rs13	rs14	rs15	rs16	rs17	rs18	rs19	rs20
rs1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs2	0.40	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs3	0.00	0.05	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs4	0.05	0.01	0.03	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs5	0.03	0.00	0.16	0.49	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs6	0.00	0.05	0.98	0.03	0.16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs7	0.00	0.04	0.65	0.05	0.24	0.67	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs8	0.04	0.01	0.13	0.28	0.66	0.13	0.20	0	0	0	0	0	0	0	0	0	0	0	0	0
rs9	0.03	0.02	0.14	0.29	0.56	0.14	0.14	0.90	0	0	0	0	0	0	0	0	0	0	0	0
rs10	0.00	0.02	0.01	0.02	0.05	0.01	0.01	0.09	0.11	0	0.91	0.00	0.00	0.01	0.01	0.01	0.00	0.04	0.01	0.03
rs11	0.00	0.02	0.01	0.03	0.05	0.01	0.01	0.08	0.09	0	0	0.00	0.00	0.01	0.01	0.01	0.00	0.04	0.01	0.03
rs12	0.00	0.00	0.01	0.00	0.02	0.01	0.01	0.00	0.00	0	0	0	1.00	0.05	0.05	0.05	0.00	0.06	0.48	0.10
rs13	0.00	0.00	0.01	0.00	0.02	0.01	0.01	0.00	0.00	0	0	0	0	0.05	0.05	0.05	0.00	0.06	0.48	0.10
rs14	0.01	0.00	0.01	0.03	0.03	0.00	0.00	0.01	0.01	0	0	0	0	0	0.95	0.99	0.05	0.12	0.08	0.02
rs15	0.01	0.00	0.01	0.02	0.03	0.01	0.00	0.01	0.01	0	0	0	0	0	0	0.96	0.05	0.11	0.10	0.03
rs16	0.01	0.00	0.00	0.02	0.03	0.00	0.00	0.01	0.01	0	0	0	0	0	0	0	0.05	0.12	0.08	0.02
rs17	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0	0	0	0	0.06	0.00	0.08
rs18	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0	0	0	0	0	0	0	0	0	0.12	0.33
rs19	0.00	0.01	0.14	0.00	0.05	0.14	0.10	0.02	0.02	0	0	0	0	0	0	0	0	0	0	0.21
rs20	0.00	0.00	0.02	0.00	0.01	0.02	0.03	0.00	0.00	0	0	0	0	0	0	0	0	0	0	0

Figure 6.5 Set the cells in the lower triangle of the lower-right quadrant with rows and columns between the random starting location and the last SNP in the dataset, highlighted in green, to 0.

	rs1	rs2	rs3	rs4	rs5	rs6	rs7	rs8	rs9	rs10	rs11	rs12	rs13	rs14	rs15	rs16	rs17	rs18	rs19	rs20
rs1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs2	0.40	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs3	0.00	0.05	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs4	0.05	0.01	0.03	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs5	0.03	0.00	0.16	0.49	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs6	0.00	0.05	NA	0.03	0.16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs7	0.00	0.04	NA	0.05	0.24	NA	0	0	0	0	0	0	0	0	0	0	0	0	0	0
rs8	0.04	0.01	0.13	0.28	NA	0.13	0.20	0	0	0	0	0	0	0	0	0	0	0	0	0
rs9	0.03	0.02	0.14	0.29	NA	0.14	0.14	NA	0	0	0	0	0	0	0	0	0	0	0	0
rs10	0.00	0.02	0.01	0.02	0.05	0.01	0.01	0.09	0.11	0	NA	0.00	0.00	0.01	0.01	0.01	0.00	0.04	0.01	0.03
rs11	0.00	0.02	0.01	0.03	0.05	0.01	0.01	0.08	0.09	0	0	0.00	0.00	0.01	0.01	0.01	0.00	0.04	0.01	0.03
rs12	0.00	0.00	0.01	0.00	0.02	0.01	0.01	0.00	0.00	0	0	0	NA	0.05	0.05	0.05	0.00	0.06	0.48	0.10
rs13	0.00	0.00	0.01	0.00	0.02	0.01	0.01	0.00	0.00	0	0	0	0	0.05	0.05	0.05	0.00	0.06	0.48	0.10
rs14	0.01	0.00	0.01	0.03	0.03	0.00	0.00	0.01	0.01	0	0	0	0	0	NA	NA	0.05	0.12	0.08	0.02
rs15	0.01	0.00	0.01	0.02	0.03	0.01	0.00	0.01	0.01	0	0	0	0	0	0	NA	0.05	0.11	0.10	0.03
rs16	0.01	0.00	0.00	0.02	0.03	0.00	0.00	0.01	0.01	0	0	0	0	0	0	0	0.05	0.12	0.08	0.02
rs17	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0	0	0	0	0	0	0	0	0.06	0.00	0.08
rs18	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0	0	0	0	0	0	0	0	0	0.12	0.33
rs19	0.00	0.01	0.14	0.00	0.05	0.14	0.10	0.02	0.02	0	0	0	0	0	0	0	0	0	0	0.21
rs20	0.00	0.00	0.02	0.00	0.01	0.02	0.03	0.00	0.00	0	0	0	0	0	0	0	0	0	0	0

Figure 6.6 Set all cells with an LD estimate above the treshold to “NA”

The matrix in Figure 6.5 is the general matrix form required for pruning from a central starting position. From a start position, the column for that SNP (rs10) is set to 0 to protect the SNP from being pruned. Certain LD statistics have been set to 0. For any

starting position, these will be for the cells that are both above the diagonal and above the row of START.SNP and the cells that are in the lower diagonal of the matrix between START.SNP and P. These are set to 0 to ensure SNPs are not pruned in the wrong direction. The matrix still contains the LD statistics for each combination of SNPs, meaning all pairs of SNPs can still be pruned including a pair of SNPs that are either side of the starting position.

SNPs are marked for pruning if the LD along any particular row of a matrix is above the specified threshold, which have all been set to NA (Figure 6.6). For the example above, an r-squared threshold of 0.5 was used. The algorithm will then start at rs10 and look along the row for any cells set to NA. In this case the cell for rs11 = NA, therefore the corresponding cell in MARK is set to NA. The process then moves to the next SNP to the right, rs11. This SNP has already been marked for pruning due to LD with rs10 and therefore the algorithm skips this SNP and moves onto the next SNP, rs12. This SNP has not been marked for pruning therefore the algorithm will look along the row of rs12 and mark any SNPs with a missing cell which is rs13. This is repeated until the last SNP is reached. The algorithm then moves back to the SNP to the left of the starting position, rs9 and repeats but instead of moving one SNP to the right, it now moves one SNP to the left until the first SNP is reached. The vector MARK will then contain a list of 1's and missing cells denoted by "NA". The missing cells mark the SNPs that will be pruned from the dataset by simply deleting the corresponding row from the genotype matrix, leaving a pruned matrix of SNPs.

6.4.2 Available options on LD the pruning algorithm

There are a number of different variations that can be implemented into the LD pruning algorithm that give the user a greater number of options for pruning a dataset. The options I discuss include a choice of LD measures, types of pruning window, size of the pruning window and an option that can fix certain SNPs into the dataset.

6.4.2.1 Pruning window

The snpStats package in R is selected to calculate LD matrices for pruning as the package showed good accuracy in estimates as well as a fast computational time, see section 5.5. One of the major advantages of this package is that it includes a "depth" option in the `ld` command that will calculate LD between a SNP and all other SNPs within a user-specified window. Therefore pruning within a window of adjacent SNPs is simple to implement into the algorithm.

Another method of implementing a pruning window would be by genetic distance as this would take into account the distribution of SNPs along the genome. A dataset could easily include a dense region of thousands of SNPs with the majority of SNPs in the region in strong LD. By implementing a window of adjacent SNPs there is a risk that the window may not cover the whole LD region and there could be strong LD between SNPs outside the window. Conversely there will be a number of sparse regions where there is a low frequency of SNPs and a high recombination rate leading to a region of independent SNPs. No SNPs will be pruned in this region, regardless of the window size. By implementing a window by distance the user can take into account the distribution of SNPs in the genome and still implement a pruning window, reducing the number of pairwise LD calculations.

To prune by genetic distance a user-supplied vector of base positions for each SNP is required. The algorithm to create an LD matrix with a window by distance is described below (Table 6.3).

Table 6.3 Instructions for the Prune algorithm implementing a sliding window by genetic distance

• Let P be the number of SNPs in the dataset. • Let Window.size be the window size in base-pairs.
1. Supply a vector POS of base-pair positions for each SNP. 2. Create a P by P matrix called DIST with all cells = 1. 3. Replace each cell (i,j) in DIST with the absolute difference in base-pairs between i^{th} SNP and j^{th} SNP in POS. 4. For all cells in DIST with an absolute difference in distance > Window.size, replace the cell with "0". 5. Create a P by P LD matrix using snpStats with depth = P. 6. For any cell in DIST = 0, replace the corresponding cell in the LD matrix with 0. 7. An LD matrix with a pruning window based on genetic distance will remain ready for pruning.

This option can be computationally time consuming for larger datasets as the algorithm requires the calculation of the full P x P LD matrix before LD statistics outside the distance threshold are set to 0.

6.4.2.2 LD Measures

As discussed in section 5.5.3, snpStats includes a number of LD measure options therefore by using this package, the algorithm can also prune from any of these measures. The measures that can be implements are: r, r-squared, D-prime, log likelihood ratio, odds ratio, Yule's Q and covariance. The Prune package can also prune VIF which is calculated using the `summary(lm())$r.squared` command in R to calculate R^2 statistics between pairwise combinations of SNPs.

6.4.2.3 Fixing SNPs into the dataset

In any dataset of SNPs that requires pruning, each SNP is at risk of being pruned. There can be circumstances where the user requires a certain SNP or set of SNPs not to be pruned and remain for any analyses that are conducted. For any supplied vector of SNPs the algorithm can set the LD matrix column vectors for these SNPs to 0, similar to step 7 in the LD pruning algorithm, see section 6.4.1. This can be performed once the LD matrix is calculated, between steps 4 and 5 in the algorithm. Each cell in the vector supplied should be the relative position of the SNP in the LD matrix. For example, to fix the first SNP, the vector would contain a cell with a '1' denoting this SNP; to fix the fifth SNP a '5' would be required.

The problem with fixing SNPs in this way is that there still may be LD between SNPs that have been fixed into the dataset and other SNPs that have not been pruned. It can be argued however, that depending on the make-up the SNPs that require fixing, there may be strong LD between SNPs in this vector too.

6.5 Other pruning methods

There are a number of alternative pruning methods that can be implemented by the algorithm. In this section, I describe these methods of pruning such as P-value pruning, LD pruning while fixing the top SNPs by P-value and LD clumping. All of these approaches implement a P-value based approach. The algorithm does not calculate P-values and must be supplied by the user. This gives the user greater flexibility, for example some analyses may that require the P-values to be adjusted for other non-genetics covariates such as age and sex. It also allows users to use P-values from other studies if desired.

6.5.1 P-value pruning

The P-value pruning algorithm applies the following instructions:

Table 6.4 Instructions for the Prune LD P-value pruning algorithm

• Let P = the number of SNPS in the dataset • Let THRESHOLD = P-value pruning threshold, where 0 ≤ THRESHOLD ≤ 1
1. Replace any P-values > THRESHOLD = “NA”. This will mark SNPs for pruning. 2. Rowbind the vector of P-values to the genotype matrix. 3. Remove any columns of SNPs with an “NA”. This will prune all marked SNPs. 4. Unbind the vector of P-values from the genotype matrix. 5. A P-value pruned genotype dataset will remain.

6.5.2 Top SNP Pruning

Top SNP pruning is a similar approach to the Fix option in the LD pruning algorithm (see section 6.4.2.3). The difference in this method is that the SNPs that are fixed are the top hits by P-value. The remaining data is pruned by LD, see section 6.4.1. The user can supply the number of top hits that are required to be fixed however it is likely that LD between the fixed SNPs and some remaining SNPs after pruning, as discussed in section 6.4.2.3. As this method prunes by LD, the options included in the LD pruning algorithm (section 6.4.2) as well as the choice of the pruning starting position, is implemented with this method.

6.5.3 LD clumping

The LD clumping algorithm applies the following instructions:

Table 6.5 Instructions for the Prune LD clumping pruning algorithm

- Let P = the number of SNPs in the dataset
- Let THRESHOLD = LD pruning threshold where 0 ≤ THRESHOLD ≤ 1

1. Create an LD matrix, with a desired window size and LD measure using snpStats.
2. Reflect the LD matrix to obtain a full P by P LD matrix. snpStats only calculates a half-matrix.
3. Set the diagonal of the matrix to 0. This will prevent SNPs being pruned out due to the LD between a SNP and itself = 1.
4. Set all cells in the matrix with an LD > THRESHOLD to "NA". This marks the SNPs are above the threshold.
5. Order the P-values from smallest to largest.
6. Create a vector of 1's called MARK of length P. Each cell denotes a SNP.
7. To mark SNPs for pruning:
    a. Start at the top SNP by P-value.
    b. If the cell in MARK that denotes this SNP is "NA" move to step e. If the cell in MARK that denotes this SNP is "1" move to step c.
    c. Take the row from the LD matrix that corresponds to this SNP.
    d. If any SNPs in this row are marked as "NA", replace the corresponding cell of MARK with an "NA".
    e. Move to the next SNP in the list of ordered SNPs.
    f. Repeat steps b – e until the last SNP in the list of ordered SNPs is reached.
8. Rowbind MARK with the genotype matrix.
9. Remove any columns (SNPS) from this matrix with an "NA". This will prune all marked SNPs.
10. Unbind MARK from the genotype matrix.
11. An LD pruned genotype matrix by ordered P-value will remain.

Like the Top SNP pruning method, the algorithm prunes by LD therefore the options discussed in section 6.4.2 are included in this algorithm. The choice of starting position cannot be used for this method as the pruning positions and respective order are based on the P-values for each SNP.

6.6 The R code

To implement the pruning algorithms described in sections 6.4 and 6.5, I have written into an R function called prune. My function calculates an LD matrix using the ld function in snpStats(212) with the desired LD measure between SNPs and window size. The Prune function then applies one of the described pruning methods as described in sections 6.4 and 6.5 and includes the user options previously discussed. The function requires the user to specify a genotype matrix (in allele dosage form) with a subject as each row and each column as a SNP. The function will produce three outputs:

- A matrix of pruned SNPs in the same format as the input genotype matrix
- A list of SNPs that remain in the dataset after pruning and their relative position in the input matrix
- A list of SNPs that are pruned out of the dataset and their relative position in the input matrix

In this section, I describe the Prune function in more detail using the R help file. The help file includes a list of all the commands and default options for this function.

6.6.1 Prune help file

Description.

Prunes a SNP dataset, given a genotype object in matrix class. Can prune by Linkage Disequilibrium (LD), P-value, LD pruning while fixing certain SNPs SNPs or LD clumping.

Usage

```
prune(geno,Pruning.Method = c("LD", "Pvalue", "Topsnp", "Clumping"), Ld.Measure = c("R.squared","D.prime"), Threshold = 0.5, Window.type = c("Position","Distance"), Window.size = 100, Start.SNP = 0, Distance = NULL, Top.SNPs = NULL, Pvalue, Fix)
```

Arguments

`geno`	Input genotype matrix to be pruned, of dimension nobs x nSNPs. Each row is a subject and each column is a SNP in genotype form (0, 1, 2). Genotype matrix must be in an object of class "*matrix*" or "*double matrix*".
`Pruning.Method`	The method required for SNP pruning. Default is `Pruning.Method = c("LD").`
`Threshold`	The pruning threshold; a number between 0 and 1 that denotes the minimum LD at which a SNP may be pruned for LD-based pruning methods i.e. `Pruning.Method = c("LD","Topsnp", "Clumping").` For `Pruning.Method = c("Pvalue")Threshold` denotes the minimum P-value at which a SNP may be pruned. If Threshold is below 0, above 1 or missing, then `Threshold` will be set to 0.5 as a default.
`Ld.Measure`	LD measures for LD-based pruning. If `LD.Measure = c("R.squared")`, r-squared statistics will be calculated for pruning. If `LD.Measure = c("D.prime")`, D-prime statistics will be calculated for pruning. Other options available

	are "LLR", "OR","Q", "Covar", "VIF" and "R". Default is LD.Measure = c("R.squared").
Window.type	Type of LD window required for pruning. Window.type = c("Position") will implement a window of adjacent SNPs. Window.type = c("Distance") will implement a window by genetic distance.
Window.size	The size of the LD window required. For Window.type = c("Position"), Window.size will implement a window of adjacent SNPs. Pruning occurs in both directions, so the specified window size is the number of adjacent SNPs included in the window in one direction. The total window size will be double the specified window size. Default is 100. For Window.type = c("Distance"), Window.size will implement a window by genetic distance in base-pairs. Default is 200,000. Pruning occurs in both directions, so the specified window size includes a number of SNPs in the window in one direction. The total window size will be double the specified window size.
Start.SNP	A number to denote initial position to start pruning. Argument is required if Pruning.Method = c("LD","Topsnp"). If Start.SNP = 0 then a starting position will randomly be selected. If 1≤ Start.SNP ≤ P then this position will the starting position for pruning. Where P is the total number of SNPs in geno. If Start.SNP < 0 or > number of SNPs in the dataset or missing then then Start.SNP will be set to a default of 1.

`Distance`	A user supplied vector of base-pair positions for each SNP. Required if `Window.type = c("Distance")`.
`Pvalue`	A user supplied vector of P-values for each SNP. Required if `Pruning.Method = c("Pvalue","Topsnp","Clumping")`.
`Top.SNPs`	A number to denote the number of top SNPs by P-value that will be fixed into the dataset and will not be pruned. If `Top.SNPs` is below 0 or above the number of SNPs in the dataset or missing then `Top.SNPs` will be set to a default of 10.
`Fix`	A vector of numerical values to denote the SNPs that require fixing into the dataset and will not be pruned. Each value in the vector will denote the column SNP that requires fixing. If a value in `Fix` < 1, > number of SNPs in the dataset or the argument `Fix` is missing then then `Fix` will be set to NULL as default.

Details

This command prunes a genotype dataset based on desired pruning method. Four methods can be used to prune the dataset using the Pruning.Method argument: “LD” for LD pruning, “Pvalue” for P-value pruning, “Topsnp” for LD pruning while ensuring the top SNPs by P-value are not pruned and “Clumping” for LD clumping. The LD based pruning algorithms gives a choice of LD measures for pruning which are calculated using the snpStats package. The “LD” and “Topsnp” pruning algorithms also include an option that gives the user a choice of starting position for pruning.

Value

Prune returns a list of the following results:

`Pruned.data`	Output genotype matrix that has been pruned, of dimension nobs x nSNPs. Each row is a subject and each column is a SNP in genotype form (0,1,2). Genotype matrix is returned in an object of class "*matrix*" or "*double matrix*".
`SNP.IN`	A list of SNPs that have not been pruned and the SNP location from the input matrix.
`SNP.OUT`	A list of SNPs that have been pruned and the SNP location from the input matrix.

6.7 Comparison of the Prune program against PLINK

In this section I compare the Prune package against PLINK. PLINK is only able to prune by r^2 and VIF measure therefore these LD statistics will be the basis for comparison. After quality control procedure applied on the GRAPHIC study described in section 4.4, the first 500 SNPs on chromosome 1 (rs3094315 to rs2493278) was used as a dataset. Both programs used a sliding window of 100 SNPs and a step size = 1. PLINK prunes from the first SNP in the dataset and its selection if SNPs is systematic and therefore repeating the pruning process on the same dataset using PLINK produces the same

pruned dataset. Prune allows the user to vary the starting location which can influence the number of SNPs in the pruned dataset therefore prune was repeated 500 times with each SNP chosen as a starting location without replacement. The LD threshold for both r^2 and R^2 varied between 0.1 and 0.9.

Results are shown in Figure 6.7. The PLINK program prune SNPs more heavily compared to my Prune package for both r^2 and VIF. As the threshold for r^2 increases, the number of SNPS remaining after pruning becomes similar between the two datasets. This is unsurprising given that most of the variation between the r^2 calculations in PLINK and snpStats occurs when $r^2 < 0.5$ (Figure 5.4). The r-squared threshold of 0.1 also showed the largest variation in the number of SNPs pruned across the 500 starting points (S.E. = 0.1225, Figure 6.8). For VIF however, the difference in number of SNPs pruned between the two programs increases as the R^2 threshold increases. PLINK does not allow LD matrices to be calculated for R^2 or VIF therefore it is not possible to conclude how PLINK calculates VIF for pruning.

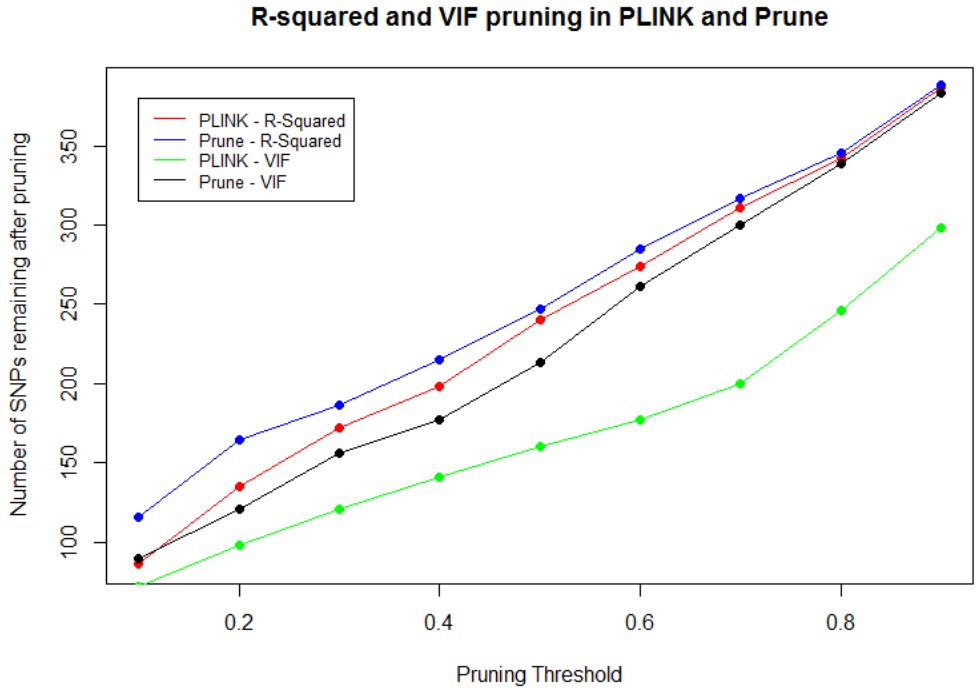


Figure 6.7 Line graph showing the number of SNPs remaining after pruning. Each line represents combinations of either PLINK or my Prune package with r-squared or VIF as the LD measure. The dataset is based on the first 500 SNPs on chromosome 1 and

1,014 subjects from the GRAPHIC study. Prune package was repeated 500 times such that each SNP was the Start SNP for pruning. The number of SNPs plotted for the Prune package is the mean number of SNPS remaining after pruning.

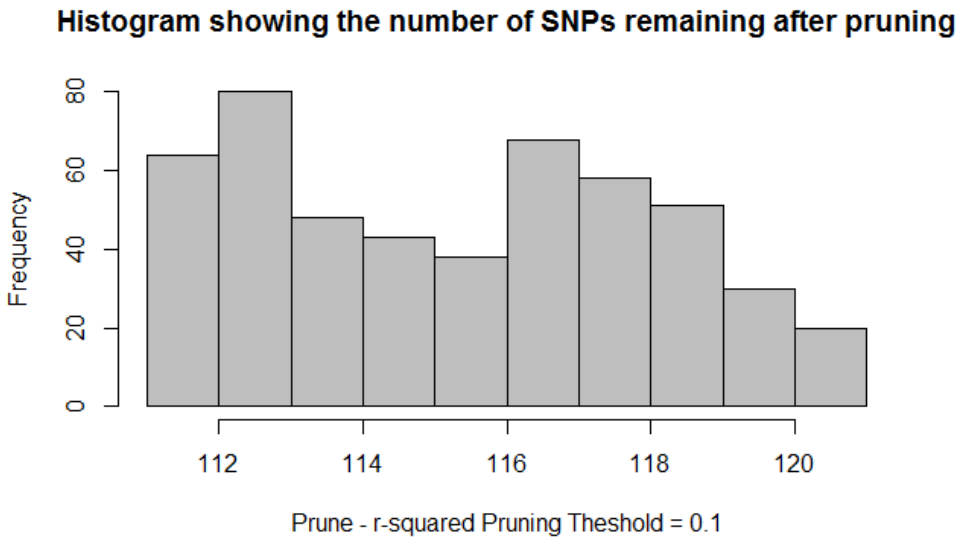


Figure 6.8 Histogram showing the number of SNPs remaining after pruning with the Prune package using an r-squared pruning threshold = 0.1 was repeated 500 times such that each SNP was the Start SNP for pruning. The dataset is based on the first 500 SNPs on chromosome 1 and 1,014 subjects from the GRAPHIC study.

Table 5.4 shows the difference between PLINK and snpStats in calculating r^2 statistics with snpStats being able to calculate an LD matrix of any size much faster than PLINK. This is naturally reflected in the computational time taken to prune each dataset between the two methods. The pruning by VIF is considerably computationally quicker in PLINK than Prune (Figure 6.9). However as the threshold increases for PLINK the computational time increases, the Prune algorithm remains stable regardless of the pruning threshold.

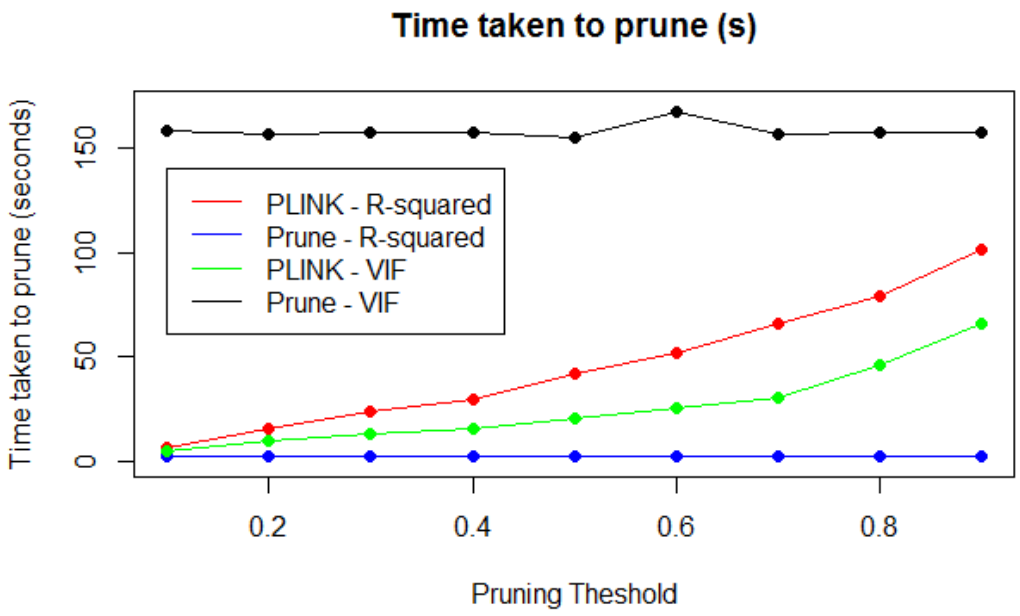


Figure 6.9 Line graph showing the time taken to prune a dataset, in seconds, which is based on the first 500 SNPs on chromosome 1 and 1,014 subjects from the GRAPHIC study. Each line represents combinations of either PLINK or Prune package with r-squared or VIF as the LD measure. The time taken plotted for the Prune package is the mean time spent pruning.

6.8 Conclusion

In this chapter, I have briefly reviewed the methods used for SNP pruning in genetic studies, including pruning by a number of LD measure, pruning by P-value and methods of pruning that combine LD and P-value pruning. These pruning methods are all combined into an R package called Prune. To current knowledge, no pruning package apart from Prune allows pruning for more than one method. Prune allows LD based pruning using a wider variety the LD pruning measures than any other package as well as an option to fix any desirable SNPs that users wish not to prune from the dataset. The program also allows an option to prune using a random starting location to eliminate any bias in pruning between a pair of SNPs but also produces some variation in the number of SNPs pruned (Figure 6.8).

Comparisons between Prune and PLINK in section 6.7 showed that PLINK prunes SNPs more heavily than Prune although the number of SNPs were similar for r^2 there was a greater difference for VIF. It is unknown why this difference occurs as the Prune algorithm calculated VIF statistics as described in PLINK (19,20). Prune uses the snpStats package to calculate most LD statistics apart from VIF, the use of this package allows significantly less computational time spent pruning compared to PLINK, the VIF method in Prune is computationally slower than PLINK.

7 Simulation study on the effects of SNP pruning on variable selection using penalised regression

7.1 Introduction

In Chapter 6, I discussed the need for SNP pruning for either removing correlations between SNPs or reducing the dimensions of a dataset as was required in application of the LASSO on the GRAPHIC study (see section 4.7). Both previous studies (24-26) and my analysis on chromosome 19 of the GRAPHIC study (see section 4.8.2) have shown the correlations between SNPs may not be a problem using penalised regression methods as they are able to accommodate for LD and select a single SNP from a highly correlated region. Given the motivation is towards reducing the number of dimensions due to the lack of computational memory or time, the question arises how varying SNP pruning methods, pruning thresholds and tuning parameter selection methods effects variable selection when fitting LASSO models.

In this chapter, I conduct a simulation study to determine the effects of various SNP pruning methods (discussed in section 6.4) on variable selection using the LASSO. The pruning thresholds vary for each dataset so that the effects of pruning on the final LASSO model can be seen between pruning method, threshold and tuning parameter selection method.

7.2 Previous literature on the effects of pruning on penalised regression

To current knowledge there has been little research done to address how pruning effects penalised regression models, specifically for variable selection. Abraham *et al*. assessed the effect of SNP independence on model prediction performance with the use of pruning on the Celiac-UK1 dataset (25). Predictive ability was assessed using the Area Under the Curve (AUC) measure which takes into account the sensitivity and specificity rates with a number of methods compared including the LASSO, elastic net, variable selection using logistic regression followed by pruning selected SNPs by LD (r^2 = 0.8), a polygenic risk score as performed in PLINK(19,20) and Genome-Wide Complex Trait Analysis (GCTA) as described by Yang *et al*.(233). The tuning parameter estimate was selected by 30 repetitions of 10-fold CV. AUC was compared between a full dataset and a dataset pruned using VIF pruning in PLINK (see section 6.3.1.1) with a sliding window size = 100, step size = 5 and VIF pruning threshold = 1.5 which pruned approximately 74% of the dataset. The results (Figure 7.1) showed that the maximum AUC for both LASSO and elastic net dropped from 0.88 to 0.85 however a larger model was required to reach the maximum AUC after pruning.

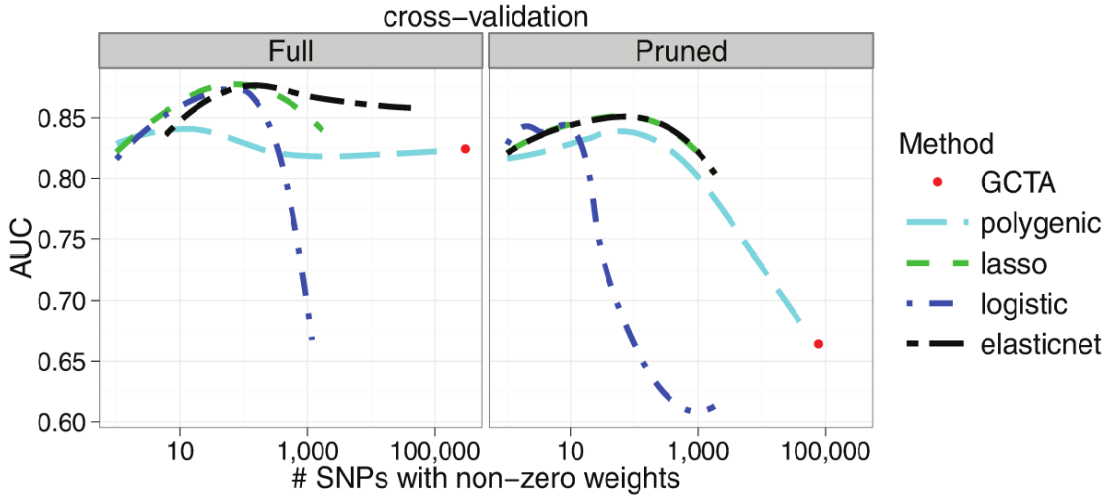


Figure 7.1 LOESS-smoothed AUC estimated in 30 x 10-fold Cross-validation within the Celiac-UK1 dataset, either the full dataset or pruned version. For GCTA, the average over the Cross-validation replications is shown. Taken from Abraham *et al.*(25)

Hong *et al*. compared two methods of pruning across four different penalised regression methods; the LASSO, ridge regression, elastic net and the adaptive LASSO in two separate datasets (146). Both studies used adult height as the phenotype. The two pruning methods used were P-value pruning and pruning by the absolute β estimate. In both cases the top 1,000 SNPs were used as the pruned dataset. Pruning by $|\beta_j|$ was not considered in this simulation nor the Prune package. As Hong *et al*. discuss, this method tends to prune common SNPs, and only keeps rare SNPs in the dataset (See Figure 1 (146)). 10-fold CV was used as the tuning parameter selection method. Over 500 of the 1,000 SNPs were selected for each combination of pruning method and penalised regression method. It is not known if this is due to the pruning method, the tuning parameter selection method or a combination of both.

7.3 Simulation of data

This simulation study looks at how various methods of pruning a dataset, discussed in section 6.4, effects variable selection using the LASSO models compared to not pruning the dataset. Sensitivity and specificity were used as measures to determine the performance for variable selection in each case. Sensitivity is defined and calculated as the true positive rate (TPR) while specificity is calculated as 1 – FPR. Both the sample size (N = 250, 500 and 1,000) and the percentage of variance explained (%VAR) by the causal SNPs (1%, 2.75% and 5.5% (3.1)) were varied in each scenario (Table 7.4).

SNPs were simulated from a single chromosome of 20,000 SNPs. In order to simulate a realistic LD pattern, SNPs were generated using the genotype data from the GRAPHIC study (see section 4.3). Only one chromosome was used for this simulation, as each chromosome is independent and LD does not occur across chromosomes. Therefore a genome-wide analysis would show similar results in each chromosome. 20,000 SNPs were selected rather than a large number in order to save time computationally in the simulation when calculating an LD matrix for the pruning algorithm (Table 5.4). The

number of SNPs on each chromosome in the GRAPHIC study varies between 48,494 SNPs on chromosome 1 to 8,741 SNPs on chromosome 21 (Table 7.1). The first 20,000 SNPs on Chromosome 13 (P = 23,049) was selected for this simulation as this was the closet to a full chromosome to 20,000 SNPs.

Table 7.1 Number of SNPs in each chromosome from the GRAPHIC study after quality control

Chromosome	No. of SNPs	Chromosome	No. of SNPs
1	48,494	**12**	29,614
2	47,899	**13**	23,049
3	39,615	**14**	19,492
4	34,217	**15**	18,258
5	35,870	**16**	19,114
6	40,655	**17**	16,761
7	32,235	**18**	17,962
8	31,753	**19**	12,376
9	28,139	**20**	15,560
10	32,500	**21**	8,741
11	30,629	**22**	8,841
Total = 591,774			

As with previous analyses, only parental subjects were used in this simulation and the same quality control procedures described in section 4.4 were applied. The 35 subjects that were removed for missing phenotypes in the GRAPHIC study analysis were included in this simulation as the phenotype was simulated.

7.3.1. Phasing haplotypes

In order to simulate new genotype datasets, haplotype data was required from each of the 1,014 subjects from the GRAPHIC study. This was done by estimating from the genotype data using fastPHASE version 1.1 (234). Haplotype estimation in fastPHASE is based on the Expectation-Maximization algorithm (205,206). fastPHASE is also able to impute missing genotype data. The package uses cluster-based modelling to estimate

the missing data. Each cluster is defined as a group of closely related haplotypes. It assumes that all haplotypes from a dataset originate from a number of clustered populations. As subjects recruited for the GRAPHIC study were all of European descent and live within a local population, this assumption would be valid. By estimating the true underlying number of clusters in the GRAPIC study, the algorithm is able to predict the missing haplotypes with a greater certainty. Scheet and Stephens tested the error rate in predicting missing genotypes using fastPHASE (234). This was performed by masking between 10% - 90% (P = 93,476 – 837,853) for CEPH HapMap data across chromosome 22 and applying the fastPHASE package to the dataset to predict the missing genotype. The calculated error rate (Table 7.2) was the proportion of masked genotypes that were not correctly estimated. While the error rate is small (3.3%) when 10% of genotypes are missing, a threshold of 3% missing call rate in SNPs is already applied on GRAPHIC study, will further reduce the error in estimating the remaining missing genotypes.

Table 7.2 Error rate for estimation of missing genotypes using fastPHASE for CEPH HapMap data, Chromosome 22, taken from Scheet & Stephens(234)

Missing Data (%)	fastPHASE Error
10	0.033
20	0.037
30	0.042
40	0.051
50	0.064
60	0.089
70	0.137
80	0.227
90	0.358

To estimate the number of clusters the 20,000 SNPs were divided into 20 blocks of 1,000 adjacent SNPs. The upper and lower limits of considered number of clusters (Kl) varied between 30 and 70 in intervals of 2. In each block of 1,000 SNPs, fastPHASE would randomly select at most 100 consecutive SNPs; mask approximately 5% of the observed genotypes among all individuals at 1,000 SNPs; impute missing genotypes at each value of Kl and tabulate errors. This was repeated 50 times for each block of

1,000 SNPs with the best estimated number of clusters as the one to produce the lowest error rate. The mean number of clusters with the smallest error rate from the 20 blocks was 60.8 (median = 62, range = 38 - 70), therefore K = 61 was selected for the number of clusters for estimation of missing data and haplotypes.

Scheet and Stephens (234) noted that the EM algorithm tends to find local maximums rather than global maximums for phasing, therefore different starting estimates for the algorithm will lead to different estimates. To deal with this issue the algorithm was repeated 50 times, each with a different starting point. Of these 50 repetitions, the repetition that produced the highest likelihood estimate was selected as the estimated haplotypes.

7.3.2. Simulation of data

Subjects were simulated by randomly combining a pair of haplotypes with replacement from phased subjects to form genotypes. Any simulated SNPs that were monomorphic were pruned from the dataset before the phenotype was simulated to avoid these SNPs being simulated as causal SNPs. 10 causal SNPs were selected at random in each dataset. The causal β's were simulated using the percentage of variance explained (%VAR) (3.1) with the MAF of the causal SNP calculated from the simulated dataset rather than from the GRAPHIC study dataset.

A simulation was run to calculate the approximate %VAR required for the simulated β's to have sufficient power. A power level of 90% was used to allow selection of SNPs in LD with the causal SNPs to have power for selection also. 50 independent SNPs from 500 unrelated subjects were simulated. From these 50 SNPs, one causal SNP was simulated. The MAF of the causal SNP was varied between 2% and 50% and the effect size of the causal SNP was also varied between 0.01 and 1 and increased by 0.01 for each MAF. Each combination MAF and effect size was repeated over 1,000 repetitions,

in each case, the LASSO was applied and the tuning parameter was selected using 10-fold Cross-validation. The power calculated by the percentage of times the causal SNP was selected over the 1,000 repetitions and is plotted against the β values for varying MAFs in Figure 7.2. Table 7.3 shows the β values and calculated %VAR for each MAF for 90% power. The %VAR varies between 2.619% and 2.967% with a mean of 2.75%. Therefore N = 500 and %VAR = 2.75% was used as a baseline scenario, other scenarios considered were at a lower power (i.e. a lower sample size or %VAR) or higher power (i.e. a higher sample size or %VAR) as shown in Table 7.4.

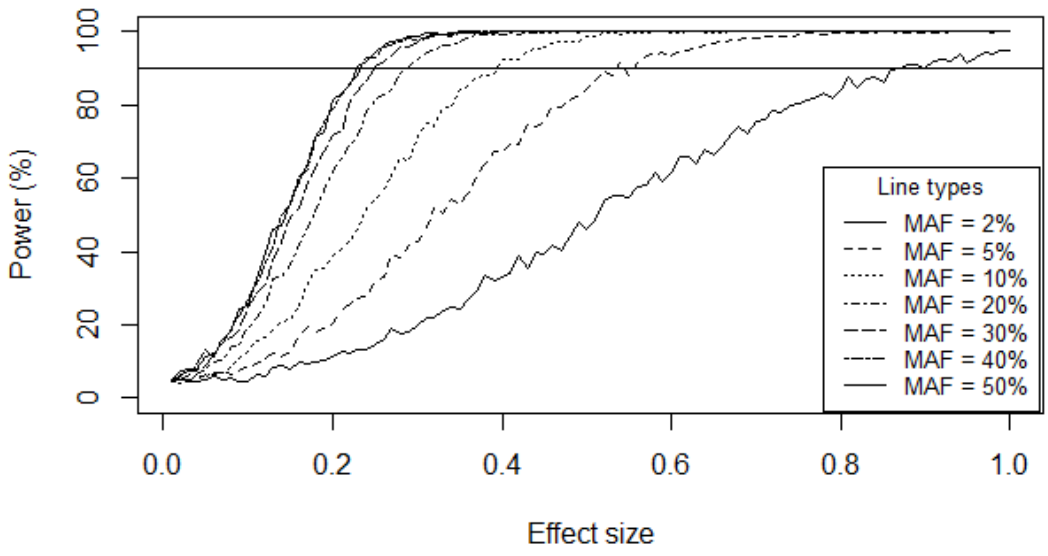


Figure 7.2 Power curves for varying MAFs and a sample size of 500

Table 7.3 The effect size and percentage variance explained by the causal SNP required for 90% power for varying MAFs

MAF (%)	N = 500	
	β	Variance explained (%)
2	0.87	2.967
5	0.54	2.770
10	0.4	2.880
15	0.33	2.777
20	0.29	2.691
25	0.28	2.940
30	0.25	2.625
35	0.24	2.621
40	0.24	2.765
45	0.23	2.619
50	0.23	2.645

Table 7.4 Simulated scenarios

Scenario	Sample size varies	Percentage variance explained varies
Low powered	N = 250, Percentage variance explained = 2.75%	N = 500, Percentage variance explained = 1%
Mid powered	N = 500 & Percentage variance explained = 2.75%	
High powered	N = 1,000, Percentage variance explained = 2.75%	N = 500, Percentage variance explained = 5.5%

Figure 7.3 shows the simulated beta estimates for each MAF and level of the %VAR by the causal SNP used in the simulation. Each causal SNP was given either a positive or negative effect at random. The residual variance of the phenotype followed a normal distribution with mean = 0 and S.E. = 1 – (Number of causal SNPS x %VAR).

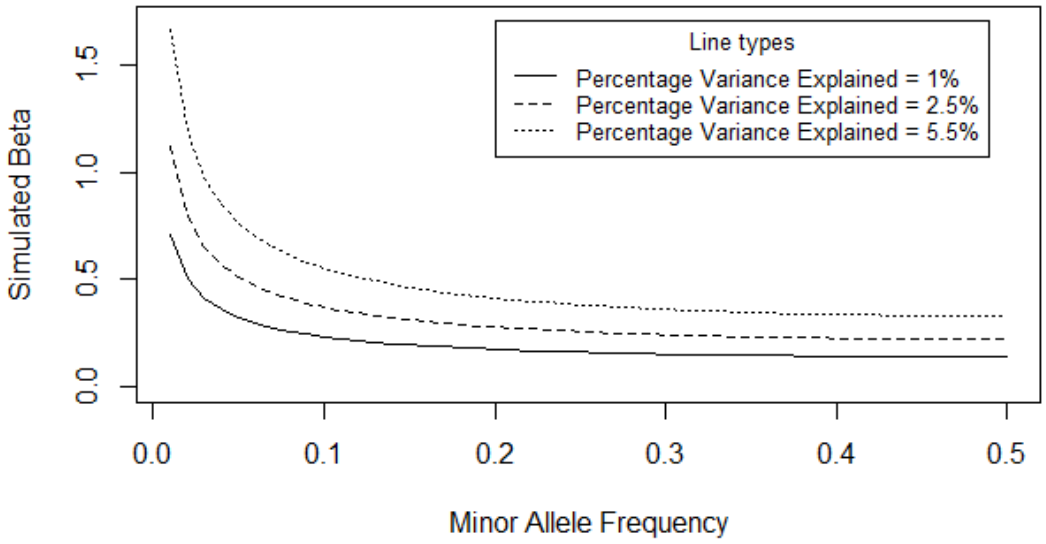


Figure 7.3 Simulated beta estimates against minor allele frequency for differing levels of percentage of variance explained

7.3.3. SNP Pruning methods

My Prune package (see section 6.4) was used to prune simulated datasets. Three pruning methods were used LD pruning, P-value pruning and LD clumping. For the LD pruning and LD clumping methods, r^2 was used as the LD measure with thresholds set between 0.9 and 0.2 and at intervals of 0.1 between these limits. For P-value pruning the threshold was set at $p < 0.2$ to $p < 0.02$ and at every 0.02 interval between these limits. No sliding window was used for this simulation. A random starting location was used to prune each dataset for LD pruning and causal SNPs were allowed to be pruned as they could be in any real study. An increase in pruning threshold was defined as a decrease in the value of pruning threshold.

7.3.4. Fitting the LASSO model

BIC, repeated 10-fold Cross-validation and the permutation method were used as tuning parameter selection methods. The glmnet package (53) was again used for variable selection for both the permutation method and repeated CV, with 25 repetitions used for both methods. To provide good accuracy in λ, a range of 200 λ values was used for repeated CV. BIC values were taken at every step of 0.001 until the smallest λ for a null model is reached, which gave a range approximately 300 λ values. Each simulation recorded the sensitivity and specificity rates and was repeated 1,000 times. A true positive result was defined as any selected SNP that is either the causal SNP or a selected SNP with $r^2 \geq 0.5$ with a causal SNP. A false positive result was defined as any selected SNP with $r^2 < 0.5$ with all 10 causal SNPs.

7.4 Results

7.4.1. LD Pruning

Table 7.5 and Table 7.6 show the results for LD pruning using CV as the method for tuning parameter selection. As the pruning threshold increased the mean number of SNPs selected in each model decreased, as did both the number of selected true positives and false positives. The sensitivity rate across all scenarios between not pruning the dataset and pruning with $r^2 = 0.2$ was approximately halved. This was expected as the number of SNPs after pruning decreased as the threshold increased. The increase in pruning threshold leads to more causal SNPs being pruned from the dataset leading to a loss of power in selecting the causal SNPs. The number of false positives selected also decreased as the pruning threshold increased with the exception of the N = 1,000 scenario where the number of false positives selected increased. Naturally as both the sample size and %Var increased, the sensitivity rate increased due to the increase in power. However as seen in section 3.3.2.1, CV tends

to select a number of false positives which also increased as the power to detect causal SNPs increased. This may partly be due to the SNPs in low LD ($0 < r^2 < 0.5$) with the causal SNP also having an increase in power. However it would not explain such a large increase in the number of false positive SNPs selected in this scenario.

Table 7.5 Mean and standard deviation results for LD pruning using repeated Cross-validation for tuning parameter selection for differing sample sizes with the percentage of variance explained = 2.75%

Pruning threshold	No. of SNPs after pruning	No. of SNPs selected	No. of true positive SNPs	No. of false positive SNPS	Sensitivity	Specificity
			N = 250			
None	20000.00 ± 0.00	34.15 ± 33.93	2.95 ± 2.06	30.31 ± 32.01	0.29 ± 0.21	1.00 ± 0.00
0.9	14992.78 ± 23.33	31.22 ± 31.29	2.91 ± 2.05	27.95 ± 29.70	0.29 ± 0.21	1.00 ± 0.00
0.8	13271.68 ± 24.81	30.19 ± 30.37	2.86 ± 2.05	27.05 ± 28.85	0.29 ± 0.20	1.00 ± 0.00
0.7	11619.73 ± 25.63	29.33 ± 30.00	2.68 ± 2.02	26.45 ± 28.54	0.27 ± 0.20	1.00 ± 0.00
0.6	10055.15 ± 25.55	27.71 ± 29.73	2.53 ± 1.99	25.05 ± 28.25	0.25 ± 0.20	1.00 ± 0.00
0.5	8589.57 ± 22.20	28.70 ± 30.22	2.45 ± 1.91	26.19 ± 28.92	0.24 ± 0.19	1.00 ± 0.00
0.4	7214.89 ± 23.30	26.67 ± 30.86	2.15 ± 1.85	24.50 ± 29.59	0.21 ± 0.18	1.00 ± 0.00
0.3	5894.42 ± 19.67	25.54 ± 31.09	1.86 ± 1.68	23.64 ± 30.07	0.19 ± 0.17	1.00 ± 0.01
0.2	4586.50 ± 19.64	22.43 ± 27.43	1.52 ± 1.51	20.91 ± 26.49	0.15 ± 0.15	1.00 ± 0.01
			N = 500			
None	20000.00 ± 0.00	78.18 ± 32.15	8.32 ± 1.36	66.78 ± 31.25	0.83 ± 0.14	1.00 ± 0.00
0.9	15041.80 ± 18.78	73.02 ± 30.46	8.28 ± 1.38	63.09 ± 29.72	0.83 ± 0.14	1.00 ± 0.00
0.8	13329.33 ± 20.41	72.81 ± 30.43	8.19 ± 1.42	63.24 ± 29.71	0.82 ± 0.14	1.00 ± 0.00
0.7	11666.30 ± 20.94	71.45 ± 30.52	8.01 ± 1.48	62.37 ± 29.77	0.80 ± 0.15	0.99 ± 0.00
0.6	10092.29 ± 22.06	70.87 ± 32.48	7.73 ± 1.57	62.44 ± 31.74	0.77 ± 0.16	0.99 ± 0.00
0.5	8619.05 ± 19.98	68.06 ± 31.39	7.33 ± 1.76	60.39 ± 30.48	0.73 ± 0.18	0.99 ± 0.00
0.4	7243.81 ± 21.98	65.65 ± 33.06	6.49 ± 1.84	59.00 ± 32.24	0.65 ± 0.18	0.99 ± 0.00
0.3	5920.86 ± 16.83	61.51 ± 34.23	5.50 ± 1.80	55.97 ± 33.51	0.55 ± 0.18	0.99 ± 0.01
0.2	4616.83 ± 18.60	57.23 ± 34.98	4.41 ± 1.84	52.82 ± 34.28	0.44 ± 0.18	0.99 ± 0.01

			N = 1,000			
None	20000.00 ± 0.00	91.83 ± 28.20	9.92 ± 0.33	76.94 ± 27.83	0.99 ± 0.03	1.00 ± 0.00
0.9	15063.67 ± 15.55	86.79 ± 26.81	9.91 ± 0.34	73.95 ± 26.59	0.99 ± 0.03	1.00 ± 0.00
0.8	13358.90 ± 16.64	87.21 ± 27.85	9.90 ± 0.35	74.84 ± 27.67	0.99 ± 0.04	0.99 ± 0.00
0.7	11688.27 ± 16.63	87.78 ± 27.20	9.88 ± 0.39	75.82 ± 27.01	0.99 ± 0.04	0.99 ± 0.00
0.6	10110.87 ± 20.92	90.33 ± 29.10	9.80 ± 0.47	78.97 ± 28.88	0.98 ± 0.05	0.99 ± 0.00
0.5	8631.07 ± 15.87	92.89 ± 30.86	9.67 ± 0.62	82.21 ± 30.70	0.97 ± 0.06	0.99 ± 0.00
0.4	7254.88 ± 20.37	94.39 ± 30.74	8.73 ± 1.09	85.18 ± 30.76	0.87 ± 0.11	0.99 ± 0.00
0.3	5934.77 ± 15.06	95.03 ± 32.99	7.45 ± 1.33	87.49 ± 33.05	0.74 ± 0.13	0.99 ± 0.01
0.2	4631.62 ± 17.96	94.97 ± 35.11	5.91 ± 1.51	89.06 ± 35.23	0.59 ± 0.15	0.98 ± 0.01

Table 7.6 Mean and standard deviation results for LD pruning using repeated Cross-validation for tuning parameter selection for differing percentage of variance explained with N = 500

Pruning threshold	No. of SNPs selected	No. of true positive SNPs	No. of false positive SNPs	Sensitivity	Specificity
Variance Explained = 1%					
None	19.57 ± 25.72	1.45 ± 1.52	17.83 ± 24.42	0.15 ± 0.15	1.00 ± 0.00
0.9	18.92 ± 24.47	1.44 ± 1.52	17.36 ± 23.35	0.14 ± 0.15	1.00 ± 0.00
0.8	18.81 ± 24.81	1.39 ± 1.52	17.32 ± 23.69	0.14 ± 0.15	1.00 ± 0.00
0.7	18.60 ± 24.60	1.38 ± 1.49	17.14 ± 23.52	0.14 ± 0.15	1.00 ± 0.00
0.6	18.64 ± 24.74	1.28 ± 1.42	17.32 ± 23.74	0.13 ± 0.14	1.00 ± 0.00
0.5	16.43 ± 22.72	1.13 ± 1.29	15.29 ± 21.84	0.11 ± 0.13	1.00 ± 0.00
0.4	16.83 ± 24.23	1.07 ± 1.29	15.76 ± 23.36	0.11 ± 0.13	1.00 ± 0.00
0.3	15.95 ± 23.73	0.94 ± 1.19	15.00 ± 22.96	0.09 ± 0.12	1.00 ± 0.00
0.2	14.67 ± 22.58	0.78 ± 1.04	13.89 ± 21.96	0.08 ± 0.10	1.00 ± 0.00
Variance Explained = 2.75%					
None	78.18 ± 32.15	8.32 ± 1.36	66.78 ± 31.25	0.83 ± 0.14	1.00 ± 0.00
0.9	73.02 ± 30.46	8.28 ± 1.38	63.09 ± 29.72	0.83 ± 0.14	1.00 ± 0.00
0.8	72.81 ± 30.43	8.19 ± 1.42	63.24 ± 29.71	0.82 ± 0.14	1.00 ± 0.00
0.7	71.45 ± 30.52	8.01 ± 1.48	62.37 ± 29.77	0.80 ± 0.15	0.99 ± 0.00
0.6	70.87 ± 32.48	7.73 ± 1.57	62.44 ± 31.74	0.77 ± 0.16	0.99 ± 0.00
0.5	68.06 ± 31.39	7.33 ± 1.76	60.39 ± 30.48	0.73 ± 0.18	0.99 ± 0.00
0.4	65.65 ± 33.06	6.49 ± 1.84	59.00 ± 32.24	0.65 ± 0.18	0.99 ± 0.00
0.3	61.51 ± 34.23	5.50 ± 1.80	55.97 ± 33.51	0.55 ± 0.18	0.99 ± 0.01
0.2	57.23 ± 34.98	4.41 ± 1.84	52.82 ± 34.28	0.44 ± 0.18	0.99 ± 0.01
Variance Explained = 5.5%					
None	96.51 ± 29.61	9.81 ± 0.48	81.70 ± 29.47	0.98 ± 0.05	1.00 ± 0.00
0.9	90.34 ± 27.99	9.81 ± 0.47	77.84 ± 27.89	0.98 ± 0.05	0.99 ± 0.00
0.8	89.90 ± 27.95	9.77 ± 0.51	77.87 ± 27.83	0.98 ± 0.05	0.99 ± 0.00
0.7	90.01 ± 28.24	9.68 ± 0.63	78.48 ± 28.04	0.97 ± 0.06	0.99 ± 0.00
0.6	90.88 ± 29.75	9.56 ± 0.68	79.99 ± 29.54	0.96 ± 0.07	0.99 ± 0.00
0.5	89.46 ± 28.54	9.35 ± 0.82	79.36 ± 28.38	0.94 ± 0.08	0.99 ± 0.00
0.4	89.86 ± 31.17	8.40 ± 1.18	81.11 ± 31.11	0.84 ± 0.12	0.99 ± 0.00
0.3	88.85 ± 34.62	7.10 ± 1.42	81.68 ± 34.55	0.71 ± 0.14	0.99 ± 0.01
0.2	84.33 ± 36.74	5.65 ± 1.59	78.68 ± 36.65	0.57 ± 0.16	0.98 ± 0.01

The results for BIC are shown in Table 7.7 and Table 7.8. The BIC tended to select a sparser model than CV when pruning by LD. This tuning parameter selection method also selected less truly causal SNPs however it also selected a significantly lower

number of false positive SNPs in all scenarios. The number of SNPs selected as well as the number of true positives selected decreases as the dataset becomes more heavily pruned however there was a small increase in these statistics compared to not pruning at all. This is illustrated in Figure 7.4 and Figure 7.5. Where no pruning has occurred, is the LD pruning threshold = 1 in these Figures. The mean number of SNPs (Figure 7.5) and causal SNPs (Figure 7.4) selected increased compared to not pruning between the LD thresholds of 0.9 and 0.6 in most scenarios. This suggests that a low LD pruning threshold using the BIC as tuning parameter selection method increases the numbers of SNPs and true positives selected. The two high powered scenarios (N = 1,000 and %Var = 5.5%) showed an increase in the numbers of false positives selected compared to a lower powered scenario, however unlike the results shown using repeated CV the increase in number of false positives selected was small and some could be explained by SNPs in low LD with the causal SNP being selected.

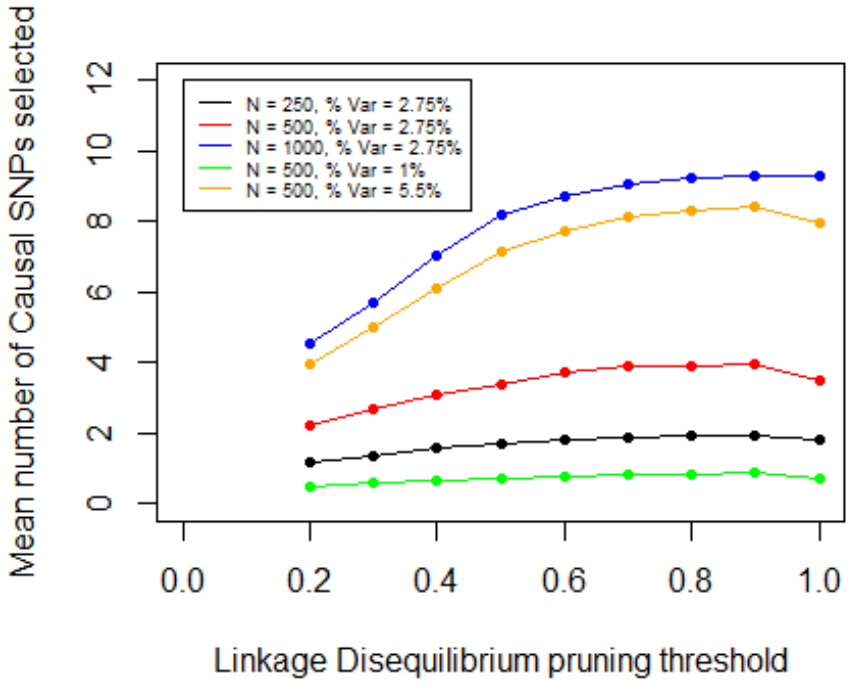


Figure 7.4 Line graph showing the mean number of causal SNPs selected against the varying Linkage Disequilibrium pruning thresholds. LD pruning threshold = 1 denotes no pruning has occurred.

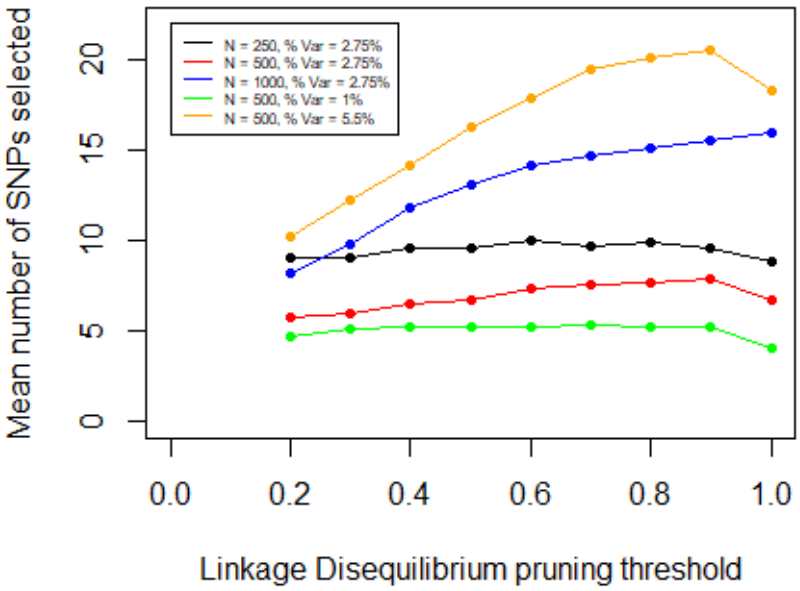


Figure 7.5 Line graph showing the mean number of SNPs selected against varying Linkage Disequilibrium pruning thresholds. LD pruning threshold = 1 denotes no pruning has occurred.

Table 7.7 Mean and standard deviation results for LD pruning using BIC for tuning parameter selection for differing sample sizes with the percentage of variance explained = 2.75%

Pruning threshold	No. of SNPs after pruning	No. of SNPs selected	No. of true positive SNPs	No. of false positive SNPS	Sensitivity	Specificity
N = 250						
None	20000.00 ± 0.00	8.81 ± 6.73	1.82 ± 1.20	6.63 ± 6.19	0.18 ± 0.12	1.00 ± 0.00
0.9	14992.78 ± 23.33	9.62 ± 7.05	1.95 ± 1.23	7.52 ± 6.55	0.20 ± 0.12	1.00 ± 0.00
0.8	13271.68 ± 24.81	9.89 ± 7.25	1.96 ± 1.28	7.81 ± 6.74	0.20 ± 0.13	1.00 ± 0.00
0.7	11619.73 ± 25.63	9.68 ± 6.63	1.88 ± 1.24	7.71 ± 6.17	0.19 ± 0.12	1.00 ± 0.00
0.6	10055.15 ± 25.55	10.06 ± 7.27	1.82 ± 1.27	8.20 ± 6.75	0.18 ± 0.13	1.00 ± 0.00
0.5	8589.57 ± 22.20	9.59 ± 7.21	1.70 ± 1.20	7.87 ± 6.84	0.17 ± 0.12	1.00 ± 0.00
0.4	7214.89 ± 23.30	9.60 ± 7.18	1.56 ± 1.16	8.04 ± 6.84	0.16 ± 0.12	1.00 ± 0.00
0.3	5894.42 ± 19.67	9.03 ± 6.92	1.34 ± 1.07	7.69 ± 6.55	0.13 ± 0.11	1.00 ± 0.00
0.2	4586.50 ± 19.64	9.06 ± 6.62	1.17 ± 1.02	7.88 ± 6.39	0.12 ± 0.10	1.00 ± 0.00
N = 500						
None	20000.00 ± 0.00	6.77 ± 6.09	3.48 ± 2.18	2.59 ± 4.04	0.35 ± 0.22	1.00 ± 0.00
0.9	15041.80 ± 18.78	7.89 ± 6.41	3.99 ± 2.19	3.50 ± 4.51	0.40 ± 0.22	1.00 ± 0.00
0.8	13329.33 ± 20.41	7.64 ± 6.28	3.91 ± 2.16	3.41 ± 4.51	0.39 ± 0.22	1.00 ± 0.00
0.7	11666.30 ± 20.94	7.63 ± 5.95	3.88 ± 2.09	3.52 ± 4.34	0.39 ± 0.21	1.00 ± 0.00
0.6	10092.29 ± 22.06	7.39 ± 5.90	3.71 ± 2.07	3.56 ± 4.30	0.37 ± 0.21	1.00 ± 0.00
0.5	8619.05 ± 19.98	6.74 ± 5.35	3.39 ± 1.93	3.29 ± 3.99	0.34 ± 0.19	1.00 ± 0.00
0.4	7243.81 ± 21.98	6.53 ± 5.05	3.08 ± 1.86	3.43 ± 3.79	0.31 ± 0.19	1.00 ± 0.00
0.3	5920.86 ± 16.83	6.03 ± 4.71	2.66 ± 1.62	3.37 ± 3.73	0.27 ± 0.16	1.00 ± 0.00
0.2	4616.83 ± 18.60	5.75 ± 4.75	2.24 ± 1.49	3.51 ± 3.98	0.22 ± 0.15	1.00 ± 0.00
N = 1,000						
None	20000.00 ± 0.00	15.94 ± 4.59	9.29 ± 1.26	4.19 ± 3.21	0.93 ± 0.13	1.00 ± 0.00

0.9	15063.67 ± 15.55	15.56 ± 4.45	9.33 ± 1.11	4.67 ± 3.44	0.93 ± 0.11	1.00 ± 0.00
0.8	13358.90 ± 16.64	15.11 ± 4.37	9.25 ± 1.25	4.62 ± 3.35	0.92 ± 0.12	1.00 ± 0.00
0.7	11688.27 ± 16.63	14.72 ± 4.58	9.05 ± 1.50	4.64 ± 3.42	0.90 ± 0.15	1.00 ± 0.00
0.6	10110.87 ± 20.92	14.16 ± 4.62	8.74 ± 1.69	4.71 ± 3.38	0.87 ± 0.17	1.00 ± 0.00
0.5	8631.07 ± 15.87	13.14 ± 4.81	8.19 ± 1.96	4.51 ± 3.40	0.82 ± 0.20	1.00 ± 0.00
0.4	7254.88 ± 20.37	11.80 ± 5.13	7.03 ± 2.22	4.61 ± 3.64	0.70 ± 0.22	1.00 ± 0.00
0.3	5934.77 ± 15.06	9.86 ± 5.04	5.68 ± 2.22	4.16 ± 3.61	0.57 ± 0.22	1.00 ± 0.00
0.2	4631.62 ± 17.96	8.23 ± 4.53	4.52 ± 2.06	3.71 ± 3.24	0.45 ± 0.21	1.00 ± 0.00

Table 7.8 Mean and standard deviation results for LD pruning using BIC for tuning parameter selection for differing percentage of variance explained with N = 500

Pruning threshold	No. of SNPs selected	No. of true positive SNPs	No. of false positive SNPs	Sensitivity	Specificity
Variance Explained = 1%					
None	4.12 ± 3.79	0.73 ± 0.80	3.28 ± 3.50	0.07 ± 0.08	1.00 ± 0.00
0.9	5.24 ± 4.09	0.87 ± 0.83	4.33 ± 3.91	0.09 ± 0.08	1.00 ± 0.00
0.8	5.28 ± 4.11	0.83 ± 0.83	4.41 ± 3.91	0.08 ± 0.08	1.00 ± 0.00
0.7	5.32 ± 4.19	0.83 ± 0.85	4.47 ± 4.00	0.08 ± 0.08	1.00 ± 0.00
0.6	5.25 ± 4.13	0.77 ± 0.83	4.47 ± 3.91	0.08 ± 0.08	1.00 ± 0.00
0.5	5.23 ± 3.96	0.69 ± 0.76	4.53 ± 3.81	0.07 ± 0.08	1.00 ± 0.00
0.4	5.22 ± 4.10	0.66 ± 0.75	4.55 ± 3.93	0.07 ± 0.07	1.00 ± 0.00
0.3	5.14 ± 3.95	0.59 ± 0.73	4.54 ± 3.82	0.06 ± 0.07	1.00 ± 0.00
0.2	4.75 ± 3.42	0.50 ± 0.66	4.25 ± 3.33	0.05 ± 0.07	1.00 ± 0.00
Variance Explained = 2.75%					
None	6.77 ± 6.09	3.48 ± 2.18	2.59 ± 4.04	0.35 ± 0.22	1.00 ± 0.00
0.9	7.89 ± 6.41	3.99 ± 2.19	3.50 ± 4.51	0.40 ± 0.22	1.00 ± 0.00
0.8	7.64 ± 6.28	3.91 ± 2.16	3.41 ± 4.51	0.39 ± 0.22	1.00 ± 0.00
0.7	7.63 ± 5.95	3.88 ± 2.09	3.52 ± 4.34	0.39 ± 0.21	1.00 ± 0.00
0.6	7.39 ± 5.90	3.71 ± 2.07	3.56 ± 4.30	0.37 ± 0.21	1.00 ± 0.00
0.5	6.74 ± 5.35	3.39 ± 1.93	3.29 ± 3.99	0.34 ± 0.19	1.00 ± 0.00
0.4	6.53 ± 5.05	3.08 ± 1.86	3.43 ± 3.79	0.31 ± 0.19	1.00 ± 0.00
0.3	6.03 ± 4.71	2.66 ± 1.62	3.37 ± 3.73	0.27 ± 0.16	1.00 ± 0.00
0.2	5.75 ± 4.75	2.24 ± 1.49	3.51 ± 3.98	0.22 ± 0.15	1.00 ± 0.00
Variance Explained = 5.5%					
None	18.37 ± 9.45	7.97 ± 2.37	7.88 ± 6.91	0.80 ± 0.24	1.00 ± 0.00
0.9	20.58 ± 9.81	8.44 ± 1.97	10.63 ± 8.06	0.84 ± 0.20	1.00 ± 0.00
0.8	20.15 ± 9.79	8.31 ± 2.02	10.64 ± 8.10	0.83 ± 0.20	1.00 ± 0.00
0.7	19.48 ± 9.74	8.12 ± 2.12	10.45 ± 8.03	0.81 ± 0.21	1.00 ± 0.00
0.6	17.89 ± 9.33	7.71 ± 2.24	9.58 ± 7.53	0.77 ± 0.22	1.00 ± 0.00
0.5	16.35 ± 9.12	7.13 ± 2.41	8.92 ± 7.30	0.71 ± 0.24	1.00 ± 0.00
0.4	14.19 ± 8.55	6.13 ± 2.41	7.94 ± 6.84	0.61 ± 0.24	1.00 ± 0.00
0.3	12.28 ± 8.03	5.00 ± 2.19	7.26 ± 6.53	0.50 ± 0.22	1.00 ± 0.00
0.2	10.26 ± 7.16	3.94 ± 2.06	6.32 ± 5.80	0.39 ± 0.21	1.00 ± 0.00

Both repeated CV and BIC methods showed variations in the mean number of true and false positive SNPs selected as well as the number of SNPs in the final model as the pruning threshold changes. The permutation method however showed a stable mean and S.D. estimate in each scenario regardless of the LD pruning threshold that was used (Table 7.9 and Table 7.10). This suggests that there is little or no effect of pruning if the permutation method is used for tuning parameter selection. The permutation method however selected sparser models that both repeated CV and BIC methods and therefore a lower number of true and false positives were selected.

Table 7.9 Mean and standard deviation results for LD pruning using the permutation method for tuning parameter selection for differing sample sizes with the percentage of variance explained = 2.75%

Pruning threshold	No. of SNPs after pruning	No. of SNPs selected	No. of true positive SNPs	No. of false positive SNPS	Sensitivity	Specificity
			N = 250			
None	20000.00 ± 0.00	2.13 ± 1.50	0.87 ± 0.86	1.13 ± 1.16	0.09 ± 0.09	1.00 ± 0.00
0.9	14992.78 ± 23.33	2.16 ± 1.51	0.90 ± 0.88	1.14 ± 1.17	0.09 ± 0.09	1.00 ± 0.00
0.8	13271.68 ± 24.81	2.17 ± 1.58	0.91 ± 0.88	1.13 ± 1.23	0.09 ± 0.09	1.00 ± 0.00
0.7	11619.73 ± 25.63	2.14 ± 1.56	0.89 ± 0.89	1.12 ± 1.17	0.09 ± 0.09	1.00 ± 0.00
0.6	10055.15 ± 25.55	2.17 ± 1.52	0.89 ± 0.88	1.13 ± 1.18	0.09 ± 0.09	1.00 ± 0.00
0.5	8589.57 ± 22.20	2.17 ± 1.55	0.88 ± 0.87	1.14 ± 1.19	0.09 ± 0.09	1.00 ± 0.00
0.4	7214.89 ± 23.30	2.17 ± 1.55	0.90 ± 0.88	1.14 ± 1.19	0.09 ± 0.09	1.00 ± 0.00
0.3	5894.42 ± 19.67	2.20 ± 1.55	0.90 ± 0.89	1.16 ± 1.20	0.09 ± 0.09	1.00 ± 0.00
0.2	4586.50 ± 19.64	2.18 ± 1.52	0.89 ± 0.87	1.13 ± 1.17	0.09 ± 0.09	1.00 ± 0.00
			N = 500			
None	20000.00 ± 0.00	5.94 ± 2.31	3.80 ± 1.43	1.37 ± 1.32	0.38 ± 0.14	1.00 ± 0.00
0.9	15041.80 ± 18.78	5.93 ± 2.37	3.79 ± 1.46	1.35 ± 1.32	0.38 ± 0.15	1.00 ± 0.00
0.8	13329.33 ± 20.41	5.89 ± 2.29	3.79 ± 1.43	1.32 ± 1.27	0.38 ± 0.14	1.00 ± 0.00
0.7	11666.30 ± 20.94	5.94 ± 2.34	3.79 ± 1.46	1.36 ± 1.32	0.38 ± 0.15	1.00 ± 0.00
0.6	10092.29 ± 22.06	5.97 ± 2.34	3.82 ± 1.43	1.36 ± 1.30	0.38 ± 0.14	1.00 ± 0.00
0.5	8619.05 ± 19.98	5.92 ± 2.37	3.81 ± 1.47	1.35 ± 1.33	0.38 ± 0.15	1.00 ± 0.00
0.4	7243.81 ± 21.98	5.99 ± 2.37	3.81 ± 1.44	1.37 ± 1.31	0.38 ± 0.14	1.00 ± 0.00
0.3	5920.86 ± 16.83	5.94 ± 2.34	3.80 ± 1.47	1.34 ± 1.31	0.38 ± 0.15	1.00 ± 0.00
0.2	4616.83 ± 18.60	5.97 ± 2.35	3.81 ± 1.44	1.37 ± 1.29	0.38 ± 0.14	1.00 ± 0.00

N = 1,000						
None	20000.00 ± 0.00	12.25 ± 2.25	8.81 ± 1.06	1.29 ± 1.24	0.88 ± 0.11	1.00 ± 0.00
0.9	15063.67 ± 15.55	12.30 ± 2.34	8.81 ± 1.06	1.30 ± 1.26	0.88 ± 0.11	1.00 ± 0.00
0.8	13358.90 ± 16.64	12.28 ± 2.25	8.82 ± 1.05	1.30 ± 1.25	0.88 ± 0.11	1.00 ± 0.00
0.7	11688.27 ± 16.63	12.21 ± 2.28	8.81 ± 1.06	1.28 ± 1.24	0.88 ± 0.11	1.00 ± 0.00
0.6	10110.87 ± 20.92	12.25 ± 2.32	8.81 ± 1.06	1.29 ± 1.23	0.88 ± 0.11	1.00 ± 0.00
0.5	8631.07 ± 15.87	12.21 ± 2.29	8.79 ± 1.06	1.28 ± 1.26	0.88 ± 0.11	1.00 ± 0.00
0.4	7254.88 ± 20.37	12.24 ± 2.28	8.80 ± 1.07	1.28 ± 1.25	0.88 ± 0.11	1.00 ± 0.00
0.3	5934.77 ± 15.06	12.27 ± 2.31	8.82 ± 1.04	1.29 ± 1.24	0.88 ± 0.10	1.00 ± 0.00
0.2	4631.62 ± 17.96	12.26 ± 2.33	8.79 ± 1.07	1.28 ± 1.25	0.88 ± 0.11	1.00 ± 0.00

Table 7.10 Mean and standard deviation results for LD pruning using the permutation method for tuning parameter selection for differing percentage of variance explained with N = 500

Pruning threshold	No. of SNPs selected	No. of true positive SNPs	No. of false positive SNPs	Sensitivity	Specificity
Variance Explained = 1%					
None	1.53 ± 1.38	0.43 ± 0.63	1.05 ± 1.15	0.04 ± 0.06	1.00 ± 0.00
0.9	1.56 ± 1.39	0.44 ± 0.64	1.06 ± 1.16	0.04 ± 0.06	1.00 ± 0.00
0.8	1.52 ± 1.36	0.42 ± 0.62	1.04 ± 1.14	0.04 ± 0.06	1.00 ± 0.00
0.7	1.55 ± 1.39	0.43 ± 0.64	1.06 ± 1.15	0.04 ± 0.06	1.00 ± 0.00
0.6	1.56 ± 1.39	0.44 ± 0.64	1.06 ± 1.17	0.04 ± 0.06	1.00 ± 0.00
0.5	1.54 ± 1.38	0.42 ± 0.63	1.06 ± 1.13	0.04 ± 0.06	1.00 ± 0.00
0.4	1.52 ± 1.33	0.43 ± 0.62	1.03 ± 1.12	0.04 ± 0.06	1.00 ± 0.00
0.3	1.54 ± 1.36	0.44 ± 0.63	1.04 ± 1.13	0.04 ± 0.06	1.00 ± 0.00
0.2	1.60 ± 1.43	0.45 ± 0.66	1.09 ± 1.16	0.04 ± 0.07	1.00 ± 0.00
Variance Explained = 2.75%					
None	5.94 ± 2.31	3.80 ± 1.43	1.37 ± 1.32	0.38 ± 0.14	1.00 ± 0.00
0.9	5.93 ± 2.37	3.79 ± 1.46	1.35 ± 1.32	0.38 ± 0.15	1.00 ± 0.00
0.8	5.89 ± 2.29	3.79 ± 1.43	1.32 ± 1.27	0.38 ± 0.14	1.00 ± 0.00
0.7	5.94 ± 2.34	3.79 ± 1.46	1.36 ± 1.32	0.38 ± 0.15	1.00 ± 0.00
0.6	5.97 ± 2.34	3.82 ± 1.43	1.36 ± 1.30	0.38 ± 0.14	1.00 ± 0.00
0.5	5.92 ± 2.37	3.81 ± 1.47	1.35 ± 1.33	0.38 ± 0.15	1.00 ± 0.00
0.4	5.99 ± 2.37	3.81 ± 1.44	1.37 ± 1.31	0.38 ± 0.14	1.00 ± 0.00
0.3	5.94 ± 2.34	3.80 ± 1.47	1.34 ± 1.31	0.38 ± 0.15	1.00 ± 0.00
0.2	5.97 ± 2.35	3.81 ± 1.44	1.37 ± 1.29	0.38 ± 0.14	1.00 ± 0.00
Variance Explained = 5.5%					
None	9.28 ± 2.44	6.66 ± 1.43	1.13 ± 1.23	0.67 ± 0.14	1.00 ± 0.00
0.9	9.33 ± 2.41	6.69 ± 1.43	1.14 ± 1.21	0.67 ± 0.14	1.00 ± 0.00
0.8	9.31 ± 2.43	6.68 ± 1.42	1.11 ± 1.20	0.67 ± 0.14	1.00 ± 0.00
0.7	9.34 ± 2.43	6.69 ± 1.45	1.16 ± 1.21	0.67 ± 0.14	1.00 ± 0.00
0.6	9.35 ± 2.44	6.70 ± 1.42	1.12 ± 1.20	0.67 ± 0.14	1.00 ± 0.00
0.5	9.32 ± 2.46	6.68 ± 1.43	1.13 ± 1.23	0.67 ± 0.14	1.00 ± 0.00
0.4	9.34 ± 2.38	6.69 ± 1.41	1.14 ± 1.22	0.67 ± 0.14	1.00 ± 0.00
0.3	9.35 ± 2.41	6.70 ± 1.42	1.15 ± 1.24	0.67 ± 0.14	1.00 ± 0.00
0.2	9.30 ± 2.43	6.68 ± 1.43	1.13 ± 1.20	0.67 ± 0.14	1.00 ± 0.00

7.4.2. P-value Pruning

The results for P-value pruning using repeated CV for the tuning parameter selection are shown in Table 7.11 and Table 7.12. Like the results for LD pruning, the results show that a large number of false positives were selected for P-value pruning. However the number of false positives selected was considerably greater for P-value pruning compared to LD pruning, leading to a lower specificity rate. The increase in the mean number of SNPs selected in a model leads to an increase in the sensitivity rate. As CV is predominantly used for model prediction, it is unsurprising that a large number of SNPs is selected. Pruning by P-value will produce a dataset of the most significant SNPs without regard of LD and therefore would over-select models to a greater extent. Pruning by LD would at least remove a number of SNPs in LD with causal SNP which P-value pruning would not.

Table 7.11 Mean and standard deviation results for P-value pruning using repeated Cross-validation for tuning parameter selection for differing sample sizes with the percentage of variance explained = 2.75%

Pruning threshold	No. of SNPs after pruning	No. of SNPs selected	No. of true positive SNPs	No. of false positive SNPS	Sensitivity	Specificity
			N = 250			
None	20000.00 ± 0.00	34.15 ± 33.93	2.95 ± 2.06	30.31 ± 32.01	0.29 ± 0.21	1.00 ± 0.00
0.2	4240.82 ± 215.10	56.14 ± 82.12	3.18 ± 2.16	51.93 ± 80.37	0.32 ± 0.22	0.99 ± 0.02
0.18	3836.78 ± 208.17	67.94 ± 95.92	3.26 ± 2.20	63.56 ± 93.98	0.33 ± 0.22	0.98 ± 0.02
0.16	3430.12 ± 199.71	94.12 ± 119.39	3.44 ± 2.25	89.41 ± 117.17	0.34 ± 0.23	0.97 ± 0.03
0.14	3022.39 ± 190.14	152.41 ± 141.17	3.91 ± 2.27	146.59 ± 138.73	0.39 ± 0.23	0.95 ± 0.05
0.12	2612.69 ± 179.20	234.39 ± 126.73	4.44 ± 2.06	227.40 ± 124.94	0.44 ± 0.21	0.91 ± 0.05
0.10	2200.74 ± 164.71	298.39 ± 53.58	4.84 ± 1.66	291.02 ± 52.99	0.48 ± 0.17	0.87 ± 0.03
0.08	1783.59 ± 149.36	299.57 ± 13.83	4.70 ± 1.58	292.50 ± 13.83	0.47 ± 0.16	0.83 ± 0.01
0.06	1362.86 ± 129.83	277.82 ± 16.47	4.38 ± 1.59	271.38 ± 16.25	0.44 ± 0.16	0.80 ± 0.02
0.04	934.28 ± 104.45	233.61 ± 18.84	4.48 ± 1.62	227.39 ± 18.69	0.45 ± 0.16	0.75 ± 0.02
0.02	494.68 ± 70.89	168.62 ± 14.87	4.81 ± 1.61	161.95 ± 14.79	0.48 ± 0.16	0.66 ± 0.03
			N = 500			
None	20000.00 ± 0.00	78.18 ± 32.15	8.32 ± 1.36	66.78 ± 31.25	0.83 ± 0.14	1.00 ± 0.00
0.2	4465.04 ± 217.16	135.49 ± 149.13	8.29 ± 1.35	123.95 ± 148.96	0.83 ± 0.13	0.97 ± 0.03
0.18	4055.86 ± 210.67	183.34 ± 184.65	8.18 ± 1.40	171.79 ± 184.67	0.82 ± 0.14	0.96 ± 0.05
0.16	3644.21 ± 202.70	249.90 ± 202.72	8.07 ± 1.45	238.39 ± 202.95	0.81 ± 0.15	0.93 ± 0.06
0.14	3229.57 ± 193.26	328.62 ± 187.21	7.88 ± 1.49	317.26 ± 187.60	0.79 ± 0.15	0.90 ± 0.06
0.12	2809.99 ± 181.02	387.15 ± 135.91	7.72 ± 1.48	375.99 ± 136.19	0.77 ± 0.15	0.87 ± 0.05
0.10	2385.39 ± 167.49	401.91 ± 79.41	7.67 ± 1.45	390.80 ± 79.71	0.77 ± 0.14	0.83 ± 0.04
0.08	1954.96 ± 150.16	386.80 ± 31.81	7.76 ± 1.38	375.54 ± 32.14	0.78 ± 0.14	0.81 ± 0.02
0.06	1516.37 ± 131.59	351.01 ± 18.58	7.96 ± 1.32	339.42 ± 18.92	0.80 ± 0.13	0.77 ± 0.02

0.04	1065.22 ± 107.50	299.95 ± 17.66	8.16 ± 1.29	287.88 ± 17.89	0.82 ± 0.13	0.73 ± 0.02
0.02	592.66 ± 75.57	221.39 ± 16.72	8.45 ± 1.20	208.63 ± 16.64	0.84 ± 0.12	0.64 ± 0.03
			N = 1,000			
None	20000.00 ± 0.00	91.83 ± 28.20	9.92 ± 0.33	76.94 ± 27.83	0.99 ± 0.03	1.00 ± 0.00
0.2	4835.41 ± 221.45	98.46 ± 42.31	9.92 ± 0.33	83.53 ± 41.89	0.99 ± 0.03	0.98 ± 0.01
0.18	4420.08 ± 216.65	101.49 ± 50.59	9.92 ± 0.33	86.54 ± 50.20	0.99 ± 0.03	0.98 ± 0.01
0.16	3999.41 ± 208.94	110.50 ± 71.87	9.92 ± 0.33	95.54 ± 71.54	0.99 ± 0.03	0.98 ± 0.02
0.14	3571.95 ± 200.65	130.96 ± 106.70	9.93 ± 0.32	115.92 ± 106.29	0.99 ± 0.03	0.97 ± 0.03
0.12	3137.60 ± 191.64	192.92 ± 167.98	9.91 ± 0.35	177.80 ± 167.63	0.99 ± 0.04	0.94 ± 0.05
0.10	2694.61 ± 181.44	304.33 ± 192.06	9.88 ± 0.38	288.98 ± 191.63	0.99 ± 0.04	0.89 ± 0.07
0.08	2242.02 ± 165.49	410.76 ± 129.95	9.85 ± 0.41	395.11 ± 129.77	0.98 ± 0.04	0.82 ± 0.06
0.06	1774.57 ± 146.65	424.13 ± 43.71	9.84 ± 0.44	408.40 ± 43.54	0.98 ± 0.04	0.77 ± 0.03
0.04	1282.82 ± 122.97	372.09 ± 21.22	9.84 ± 0.41	356.16 ± 21.08	0.98 ± 0.04	0.72 ± 0.02
0.02	755.97 ± 88.98	278.73 ± 19.55	9.86 ± 0.39	262.39 ± 19.36	0.99 ± 0.04	0.65 ± 0.03

Table 7.12 Mean and standard deviation results for P-value pruning using repeated Cross-validation for tuning parameter selection for differing percentage of variance explained with N = 500

Pruning threshold	No. of SNPs selected	No. of true positive SNPs	No. of false positive SNPs	Sensitivity	Specificity
Variance Explained = 1%					
None	19.57 ± 25.72	1.45 ± 1.52	17.83 ± 24.42	0.15 ± 0.15	1.00 ± 0.00
0.2	69.68 ± 157.51	1.70 ± 1.70	67.56 ± 156.27	0.17 ± 0.17	0.98 ± 0.04
0.18	117.51 ± 203.78	1.92 ± 1.78	115.05 ± 202.32	0.19 ± 0.18	0.97 ± 0.05
0.16	214.21 ± 237.91	2.44 ± 1.89	211.03 ± 236.21	0.24 ± 0.19	0.94 ± 0.07
0.14	315.20 ± 219.49	2.95 ± 1.89	311.19 ± 217.94	0.29 ± 0.19	0.90 ± 0.07
0.12	386.41 ± 149.64	3.53 ± 1.71	381.72 ± 148.65	0.35 ± 0.17	0.85 ± 0.06
0.1	405.27 ± 69.37	3.86 ± 1.55	400.22 ± 68.92	0.39 ± 0.16	0.81 ± 0.03
0.08	382.49 ± 22.03	3.98 ± 1.57	377.24 ± 21.97	0.40 ± 0.16	0.78 ± 0.02
0.06	341.84 ± 17.51	4.05 ± 1.56	336.46 ± 17.51	0.41 ± 0.16	0.74 ± 0.02
0.04	286.67 ± 17.19	4.16 ± 1.52	281.18 ± 17.13	0.42 ± 0.15	0.68 ± 0.03
0.02	200.89 ± 17.13	4.12 ± 1.46	195.40 ± 16.86	0.41 ± 0.15	0.57 ± 0.04
Variance Explained = 2.75%					
None	78.18 ± 32.15	8.32 ± 1.36	66.78 ± 31.25	0.83 ± 0.14	1.00 ± 0.00
0.2	135.49 ± 149.13	8.29 ± 1.35	123.95 ± 148.96	0.83 ± 0.13	0.97 ± 0.03
0.18	183.34 ± 184.65	8.18 ± 1.40	171.79 ± 184.67	0.82 ± 0.14	0.96 ± 0.05
0.16	249.90 ± 202.72	8.07 ± 1.45	238.39 ± 202.95	0.81 ± 0.15	0.93 ± 0.06
0.14	328.62 ± 187.21	7.88 ± 1.49	317.26 ± 187.60	0.79 ± 0.15	0.90 ± 0.06
0.12	387.15 ± 135.91	7.72 ± 1.48	375.99 ± 136.19	0.77 ± 0.15	0.87 ± 0.05
0.1	401.91 ± 79.41	7.67 ± 1.45	390.80 ± 79.71	0.77 ± 0.14	0.83 ± 0.04
0.08	386.80 ± 31.81	7.76 ± 1.38	375.54 ± 32.14	0.78 ± 0.14	0.81 ± 0.02
0.06	351.01 ± 18.58	7.96 ± 1.32	339.42 ± 18.92	0.80 ± 0.13	0.77 ± 0.02
0.04	299.95 ± 17.66	8.16 ± 1.29	287.88 ± 17.89	0.82 ± 0.13	0.73 ± 0.02

0.02	221.39 ± 16.72	8.45 ± 1.20	208.63 ± 16.64	0.84 ± 0.12	0.64 ± 0.03
			Variance Explained = 5.5%		
None	96.51 ± 29.61	9.81 ± 0.48	81.70 ± 29.47	0.98 ± 0.05	1.00 ± 0.00
0.2	131.68 ± 99.80	9.80 ± 0.49	116.80 ± 99.78	0.98 ± 0.05	0.97 ± 0.02
0.18	156.45 ± 127.77	9.78 ± 0.53	141.52 ± 127.82	0.98 ± 0.05	0.97 ± 0.03
0.16	197.22 ± 153.21	9.72 ± 0.63	182.07 ± 153.27	0.97 ± 0.06	0.95 ± 0.04
0.14	252.31 ± 164.58	9.62 ± 0.74	237.28 ± 164.79	0.96 ± 0.07	0.93 ± 0.05
0.12	314.41 ± 150.63	9.47 ± 0.88	299.57 ± 150.95	0.95 ± 0.09	0.90 ± 0.05
0.1	351.50 ± 112.48	9.38 ± 0.91	336.81 ± 112.93	0.94 ± 0.09	0.87 ± 0.05
0.08	362.63 ± 62.97	9.36 ± 0.89	348.01 ± 63.40	0.94 ± 0.09	0.83 ± 0.03
0.06	344.43 ± 26.40	9.42 ± 0.83	329.65 ± 26.84	0.94 ± 0.08	0.80 ± 0.02
0.04	299.99 ± 18.30	9.47 ± 0.77	284.84 ± 18.58	0.95 ± 0.08	0.75 ± 0.02
0.02	226.60 ± 16.91	9.60 ± 0.67	211.03 ± 16.91	0.96 ± 0.07	0.68 ± 0.03

Table 7.13 and Table 7.14 show results for variable selection by BIC using P-value pruning. Although it is difficult to compare results with the LD pruning method, as the pruned subset contained a different number and combination of SNPs, the results showed a similar pattern to the LD pruning results. The BIC method produced similar sized models with higher specificity rate than the repeated CV method. The mean number of SNPs selected and sensitivity rate again increased slightly in models that were pruned with a low pruning threshold but decreases as the dataset becomes heavily pruned. The two high powered scenarios (N = 1,000 and %Var = 5.5%) show the selection of true positives to be consistent regardless of the P-value pruning threshold whilst the number of false positives decreased, suggesting that this tuning parameter method may work well with a large sample size when very heavy pruning is required.

Results for the permutation method (Table 7.15 and Table 7.16) also show the same patterns to the LD pruning method. In fact the sensitivity and specificity rates as well as the mean numbers of SNPs selected were nearly the same as the LD pruning method (Table 7.9 and Table 7.10) and did not vary much with across different pruning threshold.

Table 7.13 Mean and standard deviation results for P-value pruning using BIC for tuning parameter selection for differing sample sizes with the percentage of variance explained = 2.75%

Pruning threshold	No. of SNPs after pruning	No. of SNPs selected	No. of true positive SNPs	No. of false positive SNPS	Sensitivity	Specificity
			N = 250			
None	20000.00 ± 0.00	8.81 ± 6.73	1.82 ± 1.20	6.63 ± 6.19	0.18 ± 0.12	1.00 ± 0.00
0.2	4240.82 ± 215.10	9.49 ± 7.31	1.88 ± 1.23	7.20 ± 6.70	0.19 ± 0.12	1.00 ± 0.00
0.18	3836.78 ± 208.17	9.31 ± 7.20	1.88 ± 1.23	7.04 ± 6.61	0.19 ± 0.12	1.00 ± 0.00
0.16	3430.12 ± 199.71	9.41 ± 7.31	1.89 ± 1.24	7.17 ± 6.70	0.19 ± 0.12	1.00 ± 0.00
0.14	3022.39 ± 190.14	9.69 ± 7.29	1.91 ± 1.24	7.39 ± 6.75	0.19 ± 0.12	1.00 ± 0.00
0.12	2612.69 ± 179.20	10.06 ± 7.66	1.95 ± 1.27	7.72 ± 6.99	0.19 ± 0.13	1.00 ± 0.00
0.10	2200.74 ± 164.71	9.91 ± 7.61	1.93 ± 1.24	7.59 ± 6.98	0.19 ± 0.12	1.00 ± 0.00
0.08	1783.59 ± 149.36	10.25 ± 7.94	1.97 ± 1.25	7.86 ± 7.31	0.20 ± 0.12	1.00 ± 0.00
0.06	1362.86 ± 129.83	10.52 ± 8.01	1.98 ± 1.26	8.13 ± 7.38	0.20 ± 0.13	0.99 ± 0.01
0.04	934.28 ± 104.45	11.72 ± 8.99	2.07 ± 1.28	9.21 ± 8.36	0.21 ± 0.13	0.99 ± 0.01
0.02	494.68 ± 70.89	12.88 ± 12.03	2.00 ± 1.43	10.46 ± 11.04	0.20 ± 0.14	0.98 ± 0.02
			N = 500			
None	20000.00 ± 0.00	6.77 ± 6.09	3.48 ± 2.18	2.59 ± 4.04	0.35 ± 0.22	1.00 ± 0.00
0.2	4465.04 ± 217.16	7.33 ± 6.09	3.66 ± 2.10	2.89 ± 4.12	0.37 ± 0.21	1.00 ± 0.00
0.18	4055.86 ± 210.67	7.26 ± 5.96	3.66 ± 2.11	2.83 ± 3.91	0.37 ± 0.21	1.00 ± 0.00
0.16	3644.21 ± 202.70	7.31 ± 6.14	3.68 ± 2.06	2.88 ± 4.27	0.37 ± 0.21	1.00 ± 0.00
0.14	3229.57 ± 193.26	7.37 ± 6.37	3.69 ± 2.12	2.91 ± 4.39	0.37 ± 0.21	1.00 ± 0.00
0.12	2809.99 ± 181.02	7.06 ± 5.93	3.60 ± 2.07	2.71 ± 4.00	0.36 ± 0.21	1.00 ± 0.00
0.10	2385.39 ± 167.49	6.79 ± 5.88	3.50 ± 2.10	2.55 ± 3.84	0.35 ± 0.21	1.00 ± 0.00
0.08	1954.96 ± 150.16	6.23 ± 5.72	3.32 ± 2.09	2.27 ± 3.77	0.33 ± 0.21	1.00 ± 0.00
0.06	1516.37 ± 131.59	4.97 ± 4.72	2.91 ± 2.03	1.54 ± 2.85	0.29 ± 0.20	1.00 ± 0.00

0.04	1065.22 ± 107.50	3.51 ± 3.41	2.38 ± 2.00	0.80 ± 1.45	0.24 ± 0.20	1.00 ± 0.00
0.02	592.66 ± 75.57	2.80 ± 3.13	2.06 ± 2.06	0.55 ± 1.14	0.21 ± 0.21	1.00 ± 0.00
			N = 1,000			
None	20000.00 ± 0.00	15.94 ± 4.59	9.29 ± 1.26	4.19 ± 3.21	0.93 ± 0.13	1.00 ± 0.00
0.2	4835.41 ± 221.45	15.91 ± 4.51	9.29 ± 1.26	4.19 ± 3.14	0.93 ± 0.13	1.00 ± 0.00
0.18	4420.08 ± 216.65	15.90 ± 4.49	9.28 ± 1.26	4.17 ± 3.13	0.93 ± 0.13	1.00 ± 0.00
0.16	3999.41 ± 208.94	15.87 ± 4.52	9.28 ± 1.26	4.16 ± 3.13	0.93 ± 0.13	1.00 ± 0.00
0.14	3571.95 ± 200.65	15.82 ± 4.49	9.28 ± 1.27	4.14 ± 3.12	0.93 ± 0.13	1.00 ± 0.00
0.12	3137.60 ± 191.64	15.78 ± 4.44	9.28 ± 1.26	4.09 ± 3.07	0.93 ± 0.13	1.00 ± 0.00
0.10	2694.61 ± 181.44	15.77 ± 4.47	9.28 ± 1.27	4.08 ± 3.08	0.93 ± 0.13	1.00 ± 0.00
0.08	2242.02 ± 165.49	15.74 ± 4.44	9.28 ± 1.27	4.07 ± 3.07	0.93 ± 0.13	1.00 ± 0.00
0.06	1774.57 ± 146.65	15.74 ± 4.44	9.28 ± 1.27	4.07 ± 3.07	0.93 ± 0.13	1.00 ± 0.00
0.04	1282.82 ± 122.97	15.73 ± 4.43	9.28 ± 1.26	4.07 ± 3.07	0.93 ± 0.13	1.00 ± 0.00
0.02	755.97 ± 88.98	15.58 ± 4.37	9.25 ± 1.32	3.95 ± 3.00	0.92 ± 0.13	0.99 ± 0.00

Table 7.14 Mean and standard deviation results for P-value pruning using BIC for tuning parameter selection for differing percentage of variance explained with N = 500

Pruning threshold	No. of SNPs selected	No. of true positive SNPs	No. of false positive SNPs	Sensitivity	Specificity
Variance Explained = 1%					
None	4.12 ± 3.79	0.73 ± 0.80	3.28 ± 3.50	0.07 ± 0.08	1.00 ± 0.00
0.2	5.41 ± 4.47	0.85 ± 0.85	4.43 ± 4.23	0.08 ± 0.09	1.00 ± 0.00
0.18	5.56 ± 4.68	0.88 ± 0.87	4.54 ± 4.42	0.09 ± 0.09	1.00 ± 0.00
0.16	5.55 ± 4.67	0.87 ± 0.87	4.55 ± 4.40	0.09 ± 0.09	1.00 ± 0.00
0.14	5.56 ± 4.84	0.87 ± 0.84	4.57 ± 4.60	0.09 ± 0.08	1.00 ± 0.00
0.12	5.64 ± 4.89	0.88 ± 0.85	4.63 ± 4.62	0.09 ± 0.09	1.00 ± 0.00
0.1	5.66 ± 5.15	0.87 ± 0.89	4.66 ± 4.81	0.09 ± 0.09	1.00 ± 0.00
0.08	4.63 ± 4.80	0.74 ± 0.80	3.77 ± 4.50	0.07 ± 0.08	1.00 ± 0.00
0.06	2.98 ± 4.08	0.56 ± 0.72	2.33 ± 3.79	0.06 ± 0.07	1.00 ± 0.00
0.04	1.21 ± 1.28	0.32 ± 0.53	0.87 ± 1.13	0.03 ± 0.05	1.00 ± 0.00
0.02	0.53 ± 0.64	0.19 ± 0.43	0.34 ± 0.53	0.02 ± 0.04	1.00 ± 0.00
Variance Explained = 2.75%					
None	6.77 ± 6.09	3.48 ± 2.18	2.59 ± 4.04	0.35 ± 0.22	1.00 ± 0.00
0.2	7.33 ± 6.09	3.66 ± 2.10	2.89 ± 4.12	0.37 ± 0.21	1.00 ± 0.00
0.18	7.26 ± 5.96	3.66 ± 2.11	2.83 ± 3.91	0.37 ± 0.21	1.00 ± 0.00
0.16	7.31 ± 6.14	3.68 ± 2.06	2.88 ± 4.27	0.37 ± 0.21	1.00 ± 0.00
0.14	7.37 ± 6.37	3.69 ± 2.12	2.91 ± 4.39	0.37 ± 0.21	1.00 ± 0.00
0.12	7.06 ± 5.93	3.60 ± 2.07	2.71 ± 4.00	0.36 ± 0.21	1.00 ± 0.00
0.1	6.79 ± 5.88	3.50 ± 2.10	2.55 ± 3.84	0.35 ± 0.21	1.00 ± 0.00
0.08	6.23 ± 5.72	3.32 ± 2.09	2.27 ± 3.77	0.33 ± 0.21	1.00 ± 0.00
0.06	4.97 ± 4.72	2.91 ± 2.03	1.54 ± 2.85	0.29 ± 0.20	1.00 ± 0.00
0.04	3.51 ± 3.41	2.38 ± 2.00	0.80 ± 1.45	0.24 ± 0.20	1.00 ± 0.00
0.02	2.80 ± 3.13	2.06 ± 2.06	0.55 ± 1.14	0.21 ± 0.21	1.00 ± 0.00
Variance Explained = 5.5%					
None	18.37 ± 9.45	7.97 ± 2.37	7.88 ± 6.91	0.80 ± 0.24	1.00 ± 0.00
0.2	19.03 ± 9.81	8.04 ± 2.34	8.37 ± 7.38	0.80 ± 0.23	1.00 ± 0.00
0.18	19.03 ± 9.79	8.05 ± 2.32	8.33 ± 7.22	0.81 ± 0.23	1.00 ± 0.00
0.16	18.83 ± 9.60	8.03 ± 2.30	8.18 ± 7.11	0.80 ± 0.23	1.00 ± 0.00
0.14	18.80 ± 9.71	7.99 ± 2.33	8.16 ± 7.17	0.80 ± 0.23	1.00 ± 0.00
0.12	18.36 ± 9.45	7.97 ± 2.35	7.81 ± 6.93	0.80 ± 0.24	1.00 ± 0.00
0.1	17.87 ± 9.28	7.91 ± 2.39	7.44 ± 6.75	0.79 ± 0.24	1.00 ± 0.00
0.08	16.50 ± 8.68	7.76 ± 2.45	6.37 ± 5.98	0.78 ± 0.25	1.00 ± 0.00
0.06	15.17 ± 8.27	7.58 ± 2.56	5.40 ± 5.50	0.76 ± 0.26	1.00 ± 0.00
0.04	13.01 ± 6.90	7.30 ± 2.73	3.90 ± 3.93	0.73 ± 0.27	1.00 ± 0.00
0.02	12.01 ± 6.39	7.16 ± 2.84	3.31 ± 3.35	0.72 ± 0.28	0.99 ± 0.01

Table 7.15 Mean and standard deviation results for P-value pruning using the permutation method for tuning parameter selection for differing sample sizes with the percentage of variance explained = 2.75%

Pruning threshold	No. of SNPs after pruning	No. of SNPs selected	No. of true positive SNPs	No. of false positive SNPS	Sensitivity	Specificity
			N = 250			
None	20000.00 ± 0.00	2.13 ± 1.50	0.87 ± 0.86	1.13 ± 1.16	0.09 ± 0.09	1.00 ± 0.00
0.2	4240.82 ± 215.10	2.22 ± 1.54	0.91 ± 0.88	1.16 ± 1.20	0.09 ± 0.09	1.00 ± 0.00
0.18	3836.78 ± 208.17	2.18 ± 1.53	0.91 ± 0.87	1.14 ± 1.19	0.09 ± 0.09	1.00 ± 0.00
0.16	3430.12 ± 199.71	2.14 ± 1.49	0.89 ± 0.86	1.11 ± 1.13	0.09 ± 0.09	1.00 ± 0.00
0.14	3022.39 ± 190.14	2.20 ± 1.56	0.89 ± 0.87	1.16 ± 1.23	0.09 ± 0.09	1.00 ± 0.00
0.12	2612.69 ± 179.20	2.15 ± 1.51	0.89 ± 0.88	1.13 ± 1.18	0.09 ± 0.09	1.00 ± 0.00
0.1	2200.74 ± 164.71	2.11 ± 1.54	0.88 ± 0.88	1.09 ± 1.16	0.09 ± 0.09	1.00 ± 0.00
0.08	1783.59 ± 149.36	2.16 ± 1.53	0.89 ± 0.88	1.13 ± 1.18	0.09 ± 0.09	1.00 ± 0.00
0.06	1362.86 ± 129.83	2.17 ± 1.53	0.89 ± 0.87	1.14 ± 1.18	0.09 ± 0.09	1.00 ± 0.00
0.04	934.28 ± 104.45	2.20 ± 1.55	0.90 ± 0.87	1.14 ± 1.20	0.09 ± 0.09	1.00 ± 0.00
0.02	494.68 ± 70.89	2.17 ± 1.54	0.89 ± 0.88	1.14 ± 1.16	0.09 ± 0.09	1.00 ± 0.00
			N = 500			
None	20000.00 ± 0.00	5.94 ± 2.31	3.80 ± 1.43	1.37 ± 1.32	0.38 ± 0.14	1.00 ± 0.00
0.2	4465.04 ± 217.16	5.91 ± 2.32	3.79 ± 1.44	1.35 ± 1.29	0.38 ± 0.14	1.00 ± 0.00
0.18	4055.86 ± 210.67	5.92 ± 2.41	3.79 ± 1.46	1.35 ± 1.33	0.38 ± 0.15	1.00 ± 0.00
0.16	3644.21 ± 202.70	5.95 ± 2.36	3.81 ± 1.45	1.34 ± 1.34	0.38 ± 0.14	1.00 ± 0.00
0.14	3229.57 ± 193.26	5.94 ± 2.32	3.80 ± 1.44	1.35 ± 1.30	0.38 ± 0.14	1.00 ± 0.00
0.12	2809.99 ± 181.02	5.96 ± 2.32	3.82 ± 1.43	1.37 ± 1.30	0.38 ± 0.14	1.00 ± 0.00
0.1	2385.39 ± 167.49	5.95 ± 2.35	3.80 ± 1.45	1.33 ± 1.28	0.38 ± 0.14	1.00 ± 0.00
0.08	1954.96 ± 150.16	5.87 ± 2.27	3.77 ± 1.43	1.33 ± 1.27	0.38 ± 0.14	1.00 ± 0.00
0.06	1516.37 ± 131.59	5.92 ± 2.35	3.81 ± 1.47	1.33 ± 1.28	0.38 ± 0.15	1.00 ± 0.00

0.04	1065.22 ± 107.50	5.95 ± 2.35	3.80 ± 1.45	1.36 ± 1.35	0.38 ± 0.14	1.00 ± 0.00
0.02	592.66 ± 75.57	5.92 ± 2.33	3.80 ± 1.44	1.36 ± 1.32	0.38 ± 0.14	1.00 ± 0.00
			N = 1,000			
None	20000.00 ± 0.00	12.25 ± 2.25	8.81 ± 1.06	1.29 ± 1.24	0.88 ± 0.11	1.00 ± 0.00
0.2	4835.41 ± 221.45	12.32 ± 2.33	8.82 ± 1.05	1.31 ± 1.25	0.88 ± 0.10	1.00 ± 0.00
0.18	4420.08 ± 216.65	12.28 ± 2.41	8.81 ± 1.06	1.28 ± 1.24	0.88 ± 0.11	1.00 ± 0.00
0.16	3999.41 ± 208.94	12.25 ± 2.32	8.83 ± 1.04	1.29 ± 1.26	0.88 ± 0.10	1.00 ± 0.00
0.14	3571.95 ± 200.65	12.24 ± 2.29	8.81 ± 1.05	1.27 ± 1.24	0.88 ± 0.11	1.00 ± 0.00
0.12	3137.60 ± 191.64	12.22 ± 2.31	8.80 ± 1.06	1.29 ± 1.26	0.88 ± 0.11	1.00 ± 0.00
0.1	2694.61 ± 181.44	12.25 ± 2.28	8.80 ± 1.05	1.28 ± 1.24	0.88 ± 0.11	1.00 ± 0.00
0.08	2242.02 ± 165.49	12.26 ± 2.28	8.82 ± 1.05	1.30 ± 1.24	0.88 ± 0.10	1.00 ± 0.00
0.06	1774.57 ± 146.65	12.25 ± 2.30	8.79 ± 1.08	1.28 ± 1.23	0.88 ± 0.11	1.00 ± 0.00
0.04	1282.82 ± 122.97	12.24 ± 2.31	8.79 ± 1.04	1.29 ± 1.26	0.88 ± 0.10	1.00 ± 0.00
0.02	755.97 ± 88.98	12.27 ± 2.27	8.81 ± 1.05	1.29 ± 1.24	0.88 ± 0.10	1.00 ± 0.00

Table 7.16 Mean and standard deviation results for P-value pruning using the permutation method for tuning parameter selection for differing percentage of variance explained with N = 500

Pruning threshold	No. of SNPs selected	No. of true positive SNPs	No. of false positive SNPs	Sensitivity	Specificity
Variance Explained = 1%					
None	1.53 ± 1.38	0.43 ± 0.63	1.05 ± 1.15	0.04 ± 0.06	1.00 ± 0.00
0.2	1.54 ± 1.36	0.43 ± 0.63	1.05 ± 1.16	0.04 ± 0.06	1.00 ± 0.00
0.18	1.55 ± 1.38	0.44 ± 0.64	1.05 ± 1.14	0.04 ± 0.06	1.00 ± 0.00
0.16	1.55 ± 1.37	0.44 ± 0.64	1.05 ± 1.16	0.04 ± 0.06	1.00 ± 0.00
0.14	1.55 ± 1.35	0.43 ± 0.63	1.06 ± 1.12	0.04 ± 0.06	1.00 ± 0.00
0.12	1.54 ± 1.38	0.43 ± 0.63	1.05 ± 1.17	0.04 ± 0.06	1.00 ± 0.00
0.1	1.58 ± 1.35	0.45 ± 0.64	1.06 ± 1.12	0.04 ± 0.06	1.00 ± 0.00
0.08	1.55 ± 1.35	0.44 ± 0.64	1.05 ± 1.13	0.04 ± 0.06	1.00 ± 0.00
0.06	1.56 ± 1.41	0.43 ± 0.63	1.06 ± 1.16	0.04 ± 0.06	1.00 ± 0.00
0.04	1.53 ± 1.34	0.43 ± 0.64	1.05 ± 1.13	0.04 ± 0.06	1.00 ± 0.00
0.02	1.55 ± 1.40	0.45 ± 0.64	1.04 ± 1.15	0.04 ± 0.06	1.00 ± 0.00
Variance Explained = 2.75%					
None	5.94 ± 2.31	3.80 ± 1.43	1.37 ± 1.32	0.38 ± 0.14	1.00 ± 0.00
0.2	5.91 ± 2.32	3.79 ± 1.44	1.35 ± 1.29	0.38 ± 0.14	1.00 ± 0.00
0.18	5.92 ± 2.41	3.79 ± 1.46	1.35 ± 1.33	0.38 ± 0.15	1.00 ± 0.00
0.16	5.95 ± 2.36	3.81 ± 1.45	1.34 ± 1.34	0.38 ± 0.14	1.00 ± 0.00
0.14	5.94 ± 2.32	3.80 ± 1.44	1.35 ± 1.30	0.38 ± 0.14	1.00 ± 0.00
0.12	5.96 ± 2.32	3.82 ± 1.43	1.37 ± 1.30	0.38 ± 0.14	1.00 ± 0.00
0.1	5.95 ± 2.35	3.80 ± 1.45	1.33 ± 1.28	0.38 ± 0.14	1.00 ± 0.00
0.08	5.87 ± 2.27	3.77 ± 1.43	1.33 ± 1.27	0.38 ± 0.14	1.00 ± 0.00
0.06	5.92 ± 2.35	3.81 ± 1.47	1.33 ± 1.28	0.38 ± 0.15	1.00 ± 0.00
0.04	5.95 ± 2.35	3.80 ± 1.45	1.36 ± 1.35	0.38 ± 0.14	1.00 ± 0.00
0.02	5.92 ± 2.33	3.80 ± 1.44	1.36 ± 1.32	0.38 ± 0.14	1.00 ± 0.00
Variance Explained = 5.5%					
None	9.28 ± 2.44	6.66 ± 1.43	1.13 ± 1.23	0.67 ± 0.14	1.00 ± 0.00
0.2	9.34 ± 2.43	6.68 ± 1.43	1.14 ± 1.22	0.67 ± 0.14	1.00 ± 0.00
0.18	9.32 ± 2.43	6.66 ± 1.42	1.14 ± 1.22	0.67 ± 0.14	1.00 ± 0.00
0.16	9.31 ± 2.36	6.69 ± 1.40	1.13 ± 1.20	0.67 ± 0.14	1.00 ± 0.00
0.14	9.33 ± 2.40	6.69 ± 1.42	1.15 ± 1.20	0.67 ± 0.14	1.00 ± 0.00
0.12	9.33 ± 2.41	6.70 ± 1.41	1.12 ± 1.20	0.67 ± 0.14	1.00 ± 0.00
0.1	9.31 ± 2.47	6.66 ± 1.43	1.14 ± 1.21	0.67 ± 0.14	1.00 ± 0.00
0.08	9.28 ± 2.39	6.67 ± 1.42	1.12 ± 1.18	0.67 ± 0.14	1.00 ± 0.00
0.06	9.30 ± 2.42	6.68 ± 1.42	1.12 ± 1.22	0.67 ± 0.14	1.00 ± 0.00
0.04	9.33 ± 2.43	6.68 ± 1.41	1.14 ± 1.21	0.67 ± 0.14	1.00 ± 0.00
0.02	9.38 ± 2.36	6.72 ± 1.40	1.14 ± 1.22	0.67 ± 0.14	1.00 ± 0.00

7.4.3. LD clumping

There were some differences between the results for LD pruning and for LD clumping when CV was used for tuning parameter selection. The mean number of SNPs selected along with the sensitivity rate decreased as the LD pruning threshold increased, but both the sensitivity and specificity rates increased as the LD clumping pruning threshold increased. The LD clumping method ensures that the most statistically significant variants remain, producing a dataset similar to that of pruning by P-value without the SNPs in LD of the most significant SNPs. Therefore the mean number of false positives selected was likely to lie between the values of LD pruning and P-value pruning. As seen with the other pruning methods, the increase in power of the causal SNPs leads to an increase to the model size and decrease in both sensitivity and specificity rates.

The results for the BIC using LD clumping (Table 7.19 and Table 7.20) showed slightly better results compared to the LD pruning method (Table 7.7 and Table 7.8). This is expected as LD clumping will ensure the most statistically significant SNPs will remain in the dataset and not be pruned. This was also illustrated when the datasets were more heavily pruned, the mean model size and number of true positives selected decreased using LD pruning, but there is little change using LD clumping. However the mean number of false positive selected also increased when the dataset is more heavily pruned by LD clumping. As seen with all the pruning methods, using the BIC the datasets that were pruned selected a slightly higher number of SNPs than the same dataset that was not pruned.

For all three tuning parameter selection methods the permutation selected the least number of SNPs in each scenario however the method was not affected by both SNP pruning method or the pruning threshold used. The results for LD clumping (Table 7.21 and Table 7.22) were again the same as the results for both LD pruning (Table 7.9 and Table 7.10) and P-value pruning (Table 7.15 and Table 7.16).

Table 7.17 Mean and standard deviation results for LD clumping using repeated Cross-validation for tuning parameter selection for differing sample sizes with the percentage of variance explained = 2.75%

Pruning threshold	No. of SNPs after pruning	No. of SNPs selected	No. of true positive SNPs	No. of false positive SNPS	Sensitivity	Specificity
			N = 250			
None	20000.00 ± 0.00	31.32 ± 31.43	2.99 ± 2.07	28.13 ± 30.15	0.29 ± 0.21	1.00 ± 0.00
0.9	14973.03 ± 23.12	32.84 ± 31.93	3.04 ± 2.08	29.45 ± 30.41	0.30 ± 0.21	1.00 ± 0.00
0.8	13219.21 ± 24.81	33.82 ± 32.14	3.08 ± 2.06	30.43 ± 30.64	0.31 ± 0.21	1.00 ± 0.00
0.7	11538.73 ± 26.19	36.65 ± 34.40	3.18 ± 2.09	33.23 ± 32.98	0.32 ± 0.21	1.00 ± 0.00
0.6	9944.99 ± 26.67	39.82 ± 36.36	3.31 ± 2.09	36.38 ± 35.00	0.33 ± 0.21	1.00 ± 0.00
0.5	8454.05 ± 25.53	43.89 ± 39.58	3.39 ± 2.10	40.47 ± 38.30	0.34 ± 0.21	1.00 ± 0.00
0.4	7062.36 ± 23.52	51.31 ± 44.33	3.57 ± 2.07	47.72 ± 43.14	0.36 ± 0.21	0.99 ± 0.01
0.3	5729.32 ± 22.12	61.29 ± 51.08	3.75 ± 2.05	57.54 ± 50.00	0.38 ± 0.20	0.99 ± 0.01
0.2	4388.77 ± 20.16	79.56 ± 59.70	3.96 ± 2.00	75.63 ± 58.77	0.40 ± 0.20	0.98 ± 0.01
			N = 500			
None	20000.00 ± 0.00	73.13 ± 30.37	8.31 ± 1.34	61.97 ± 30.03	0.83 ± 0.13	1.00 ± 0.00
0.9	15019.94 ± 19.62	74.83 ± 30.96	8.34 ± 1.34	64.90 ± 30.27	0.83 ± 0.13	1.00 ± 0.00
0.8	13272.66 ± 21.06	76.35 ± 31.15	8.37 ± 1.34	66.69 ± 30.50	0.84 ± 0.13	0.99 ± 0.00
0.7	11582.75 ± 22.54	78.63 ± 32.85	8.38 ± 1.33	69.29 ± 32.25	0.84 ± 0.13	0.99 ± 0.00
0.6	9978.88 ± 22.15	81.41 ± 33.27	8.44 ± 1.31	72.40 ± 32.78	0.84 ± 0.13	0.99 ± 0.00
0.5	8480.43 ± 23.58	85.87 ± 34.90	8.49 ± 1.29	77.27 ± 34.53	0.85 ± 0.13	0.99 ± 0.00
0.4	7086.22 ± 23.19	91.00 ± 37.42	8.41 ± 1.29	82.55 ± 37.10	0.84 ± 0.13	0.99 ± 0.01
0.3	5755.76 ± 21.39	99.20 ± 40.40	8.32 ± 1.30	90.87 ± 40.10	0.83 ± 0.13	0.98 ± 0.01
0.2	4416.63 ± 20.10	113.45 ± 47.34	8.18 ± 1.31	105.27 ± 47.11	0.82 ± 0.13	0.98 ± 0.01

			N = 1,000			
None	20000.00 ± 0.00	87.69 ± 27.31	9.92 ± 0.33	73.62 ± 26.55	0.99 ± 0.03	1.00 ± 0.00
0.9	15040.79 ± 16.01	87.69 ± 27.31	9.92 ± 0.33	74.94 ± 27.10	0.99 ± 0.03	1.00 ± 0.00
0.8	13299.53 ± 17.21	88.19 ± 27.62	9.92 ± 0.33	76.08 ± 27.41	0.99 ± 0.03	0.99 ± 0.00
0.7	11602.81 ± 19.44	89.94 ± 28.75	9.92 ± 0.33	78.47 ± 28.61	0.99 ± 0.03	0.99 ± 0.00
0.6	9996.51 ± 20.79	91.89 ± 29.37	9.92 ± 0.33	81.05 ± 29.24	0.99 ± 0.03	0.99 ± 0.00
0.5	8489.93 ± 20.96	94.69 ± 29.61	9.92 ± 0.34	84.60 ± 29.57	0.99 ± 0.03	0.99 ± 0.00
0.4	7096.75 ± 21.11	99.27 ± 32.18	9.84 ± 0.42	89.37 ± 32.16	0.98 ± 0.04	0.99 ± 0.00
0.3	5769.18 ± 20.28	105.69 ± 34.20	9.76 ± 0.51	95.92 ± 34.18	0.98 ± 0.05	0.98 ± 0.01
0.2	4429.58 ± 20.11	116.76 ± 39.76	9.66 ± 0.58	107.10 ± 39.75	0.97 ± 0.06	0.98 ± 0.01

Table 7.18 Mean and standard deviation results for LD clumping using repeated Cross-validation for tuning parameter selection for differing percentage of variance explained with N = 500

Pruning threshold	No. of SNPs selected	No. of true positive SNPs	No. of false positive SNPs	Sensitivity	Specificity
Variance Explained = 1%					
1	18.67 ± 24.35	1.45 ± 1.50	17.10 ± 23.13	0.15 ± 0.16	1.00 ± 0.00
0.9	19.89 ± 25.55	1.50 ± 1.55	18.27 ± 24.41	0.15 ± 0.16	1.00 ± 0.00
0.8	20.91 ± 26.57	1.53 ± 1.57	19.28 ± 25.44	0.15 ± 0.16	1.00 ± 0.00
0.7	23.11 ± 28.25	1.63 ± 1.62	21.40 ± 27.11	0.16 ± 0.16	1.00 ± 0.00
0.6	25.54 ± 30.17	1.71 ± 1.65	23.78 ± 29.00	0.17 ± 0.17	1.00 ± 0.00
0.5	29.30 ± 33.29	1.82 ± 1.69	27.48 ± 32.16	0.18 ± 0.17	1.00 ± 0.00
0.4	33.67 ± 36.88	1.92 ± 1.70	31.76 ± 35.75	0.19 ± 0.17	1.00 ± 0.01
0.3	41.20 ± 42.92	2.08 ± 1.74	39.12 ± 41.76	0.21 ± 0.17	0.99 ± 0.01
0.2	55.12 ± 53.31	2.30 ± 1.77	52.82 ± 52.22	0.23 ± 0.18	0.99 ± 0.01
Variance Explained = 2.75%					
1	73.13 ± 30.37	8.31 ± 1.34	61.97 ± 30.03	0.83 ± 0.13	1.00 ± 0.00
0.9	74.83 ± 30.96	8.34 ± 1.34	64.90 ± 30.27	0.83 ± 0.13	1.00 ± 0.00
0.8	76.35 ± 31.15	8.37 ± 1.34	66.69 ± 30.50	0.84 ± 0.13	0.99 ± 0.00
0.7	78.63 ± 32.85	8.38 ± 1.33	69.29 ± 32.25	0.84 ± 0.13	0.99 ± 0.00
0.6	81.41 ± 33.27	8.44 ± 1.31	72.40 ± 32.78	0.84 ± 0.13	0.99 ± 0.00
0.5	85.87 ± 34.90	8.49 ± 1.29	77.27 ± 34.53	0.85 ± 0.13	0.99 ± 0.00
0.4	91.00 ± 37.42	8.41 ± 1.29	82.55 ± 37.10	0.84 ± 0.13	0.99 ± 0.01
0.3	99.20 ± 40.40	8.32 ± 1.30	90.87 ± 40.10	0.83 ± 0.13	0.98 ± 0.01
0.2	113.45 ± 47.34	8.18 ± 1.31	105.27 ± 47.11	0.82 ± 0.13	0.98 ± 0.01
Variance Explained = 5.5%					
1	89.98 ± 26.89	9.81 ± 0.48	76.73 ± 25.61	0.98 ± 0.05	0.99 ± 0.00
0.9	90.90 ± 28.11	9.81 ± 0.48	78.53 ± 27.99	0.98 ± 0.05	0.99 ± 0.00
0.8	91.96 ± 29.32	9.82 ± 0.48	80.09 ± 29.23	0.98 ± 0.05	0.99 ± 0.00
0.7	93.09 ± 29.50	9.82 ± 0.48	81.80 ± 29.39	0.98 ± 0.05	0.99 ± 0.00
0.6	95.55 ± 31.43	9.82 ± 0.48	84.86 ± 31.37	0.98 ± 0.05	0.99 ± 0.00
0.5	98.59 ± 32.38	9.82 ± 0.47	88.59 ± 32.33	0.98 ± 0.05	0.99 ± 0.00
0.4	102.46 ± 35.31	9.70 ± 0.59	92.69 ± 35.30	0.97 ± 0.06	0.99 ± 0.00
0.3	108.81 ± 37.56	9.56 ± 0.68	99.23 ± 37.55	0.96 ± 0.07	0.98 ± 0.01
0.2	120.73 ± 43.14	9.39 ± 0.79	111.24 ± 43.01	0.94 ± 0.08	0.97 ± 0.01

Table 7.19 Mean and standard deviation results for LD clumping using BIC for tuning parameter selection for differing sample sizes with the percentage of variance explained = 2.75%

Pruning threshold	No. of SNPs after pruning	No. of SNPs selected	No. of true positive SNPs	No. of false positive SNPS	Sensitivity	Specificity
N = 250						
None	20000.00 ± 0.00	9.69 ± 7.53	2.00 ± 1.28	7.54 ± 6.98	0.20 ± 0.13	1.00 ± 0.00
0.9	14973.03 ± 23.12	9.77 ± 7.44	1.99 ± 1.29	7.63 ± 6.84	0.20 ± 0.13	1.00 ± 0.00
0.8	13219.21 ± 24.81	9.90 ± 7.28	1.99 ± 1.28	7.77 ± 6.73	0.20 ± 0.13	1.00 ± 0.00
0.7	11538.73 ± 26.19	10.04 ± 7.26	2.02 ± 1.29	7.92 ± 6.74	0.20 ± 0.13	1.00 ± 0.00
0.6	9944.99 ± 26.67	9.60 ± 6.93	2.02 ± 1.27	7.55 ± 6.46	0.20 ± 0.13	1.00 ± 0.00
0.5	8454.05 ± 25.53	9.67 ± 6.87	2.02 ± 1.28	7.64 ± 6.35	0.20 ± 0.13	1.00 ± 0.00
0.4	7062.36 ± 23.52	9.82 ± 6.99	2.05 ± 1.29	7.77 ± 6.47	0.20 ± 0.13	1.00 ± 0.00
0.3	5729.32 ± 22.12	9.61 ± 6.84	2.01 ± 1.27	7.60 ± 6.37	0.20 ± 0.13	1.00 ± 0.00
0.2	4388.77 ± 20.16	9.56 ± 6.93	1.99 ± 1.29	7.57 ± 6.43	0.20 ± 0.13	1.00 ± 0.00
N = 500						
None	20000.00 ± 0.00	7.90 ± 6.63	4.02 ± 2.25	3.27 ± 4.57	0.40 ± 0.23	1.00 ± 0.00
0.9	15019.94 ± 19.62	8.02 ± 6.62	4.08 ± 2.26	3.55 ± 4.76	0.41 ± 0.23	1.00 ± 0.00
0.8	13272.66 ± 21.06	8.30 ± 6.80	4.17 ± 2.26	3.84 ± 4.97	0.42 ± 0.23	1.00 ± 0.00
0.7	11582.75 ± 22.54	8.36 ± 6.69	4.23 ± 2.27	3.91 ± 4.88	0.42 ± 0.23	1.00 ± 0.00
0.6	9978.88 ± 22.15	8.79 ± 7.18	4.36 ± 2.31	4.29 ± 5.46	0.44 ± 0.23	1.00 ± 0.00
0.5	8480.43 ± 23.58	8.79 ± 6.89	4.40 ± 2.29	4.36 ± 5.24	0.44 ± 0.23	1.00 ± 0.00
0.4	7086.22 ± 23.19	8.85 ± 6.79	4.45 ± 2.27	4.40 ± 5.18	0.44 ± 0.23	1.00 ± 0.00
0.3	5755.76 ± 21.39	8.93 ± 6.87	4.45 ± 2.29	4.49 ± 5.23	0.44 ± 0.23	1.00 ± 0.00
0.2	4416.63 ± 20.10	9.03 ± 6.83	4.46 ± 2.27	4.58 ± 5.16	0.45 ± 0.23	1.00 ± 0.00

			N = 1,000			
None	20000.00 ± 0.00	15.80 ± 4.21	9.36 ± 1.11	4.49 ± 3.23	0.94 ± 0.11	1.00 ± 0.00
0.9	15040.79 ± 16.01	15.50 ± 4.18	9.37 ± 1.10	4.63 ± 3.27	0.94 ± 0.11	1.00 ± 0.00
0.8	13299.53 ± 17.21	15.26 ± 4.16	9.39 ± 1.07	4.75 ± 3.34	0.94 ± 0.11	1.00 ± 0.00
0.7	11602.81 ± 19.44	14.98 ± 4.08	9.40 ± 1.07	4.84 ± 3.38	0.94 ± 0.11	1.00 ± 0.00
0.6	9996.51 ± 20.79	14.87 ± 4.02	9.41 ± 1.04	5.03 ± 3.43	0.94 ± 0.10	1.00 ± 0.00
0.5	8489.93 ± 20.96	14.73 ± 3.97	9.44 ± 1.00	5.22 ± 3.54	0.94 ± 0.10	1.00 ± 0.00
0.4	7096.75 ± 21.11	14.59 ± 3.75	9.39 ± 0.96	5.19 ± 3.37	0.94 ± 0.10	1.00 ± 0.00
0.3	5769.18 ± 20.28	14.57 ± 3.92	9.34 ± 0.96	5.24 ± 3.59	0.93 ± 0.10	1.00 ± 0.00
0.2	4429.58 ± 20.11	14.40 ± 3.75	9.27 ± 0.98	5.14 ± 3.44	0.93 ± 0.10	1.00 ± 0.00

Table 7.20 Mean and standard deviation results for LD clumping using BIC for tuning parameter selection for differing percentage of variance explained with N = 500

Pruning threshold	No. of SNPs selected	No. of true positive SNPs	No. of false positive SNPs	Sensitivity	Specificity
Variance Explained = 1%					
None	4.83 ± 3.75	0.84 ± 0.84	3.98 ± 3.60	0.09 ± 0.09	1.00 ± 0.00
0.9	5.08 ± 3.83	0.86 ± 0.85	4.17 ± 3.62	0.09 ± 0.09	1.00 ± 0.00
0.8	5.24 ± 3.89	0.88 ± 0.85	4.32 ± 3.68	0.09 ± 0.09	1.00 ± 0.00
0.7	5.20 ± 3.89	0.89 ± 0.84	4.29 ± 3.69	0.09 ± 0.08	1.00 ± 0.00
0.6	5.34 ± 4.19	0.91 ± 0.86	4.42 ± 4.00	0.09 ± 0.09	1.00 ± 0.00
0.5	5.13 ± 3.65	0.90 ± 0.86	4.23 ± 3.45	0.09 ± 0.09	1.00 ± 0.00
0.4	5.23 ± 4.02	0.89 ± 0.87	4.34 ± 3.81	0.09 ± 0.09	1.00 ± 0.00
0.3	5.17 ± 3.84	0.90 ± 0.86	4.28 ± 3.60	0.09 ± 0.09	1.00 ± 0.00
0.2	4.91 ± 3.57	0.86 ± 0.84	4.06 ± 3.38	0.09 ± 0.08	1.00 ± 0.00
Variance Explained = 2.75%					
None	7.90 ± 6.63	4.02 ± 2.25	3.27 ± 4.57	0.40 ± 0.23	1.00 ± 0.00
0.9	8.02 ± 6.62	4.08 ± 2.26	3.55 ± 4.76	0.41 ± 0.23	1.00 ± 0.00
0.8	8.30 ± 6.80	4.17 ± 2.26	3.84 ± 4.97	0.42 ± 0.23	1.00 ± 0.00
0.7	8.36 ± 6.69	4.23 ± 2.27	3.91 ± 4.88	0.42 ± 0.23	1.00 ± 0.00
0.6	8.79 ± 7.18	4.36 ± 2.31	4.29 ± 5.46	0.44 ± 0.23	1.00 ± 0.00
0.5	8.79 ± 6.89	4.40 ± 2.29	4.36 ± 5.24	0.44 ± 0.23	1.00 ± 0.00
0.4	8.85 ± 6.79	4.45 ± 2.27	4.40 ± 5.18	0.44 ± 0.23	1.00 ± 0.00
0.3	8.93 ± 6.87	4.45 ± 2.29	4.49 ± 5.23	0.44 ± 0.23	1.00 ± 0.00
0.2	9.03 ± 6.83	4.46 ± 2.27	4.58 ± 5.16	0.45 ± 0.23	1.00 ± 0.00
Variance Explained = 5.5%					
None	20.18 ± 9.35	8.39 ± 1.90	10.03 ± 8.07	0.84 ± 0.19	1.00 ± 0.00
0.9	20.70 ± 9.75	8.52 ± 1.91	10.71 ± 8.10	0.85 ± 0.19	1.00 ± 0.00
0.8	21.17 ± 9.99	8.62 ± 1.82	11.43 ± 8.56	0.86 ± 0.18	1.00 ± 0.00
0.7	21.09 ± 9.25	8.70 ± 1.73	11.59 ± 8.02	0.87 ± 0.17	1.00 ± 0.00
0.6	21.17 ± 9.21	8.75 ± 1.72	11.96 ± 8.09	0.87 ± 0.17	1.00 ± 0.00
0.5	21.50 ± 9.31	8.80 ± 1.64	12.63 ± 8.34	0.88 ± 0.16	1.00 ± 0.00
0.4	21.50 ± 9.20	8.73 ± 1.62	12.76 ± 8.33	0.87 ± 0.16	1.00 ± 0.00
0.3	21.32 ± 9.01	8.65 ± 1.63	12.67 ± 8.15	0.86 ± 0.16	1.00 ± 0.00
0.2	21.05 ± 8.61	8.58 ± 1.57	12.47 ± 7.83	0.86 ± 0.16	1.00 ± 0.00

Table 7.21 Mean and standard deviation results for LD clumping using the permutation method for tuning parameter selection for differing sample sizes with the percentage of variance explained = 2.75%

Pruning threshold	No. of SNPs after pruning	No. of SNPs selected	No. of true positive SNPs	No. of false positive SNPS	Sensitivity	Specificity
N = 250						
None	20000.00 ± 0.00	2.20 ± 1.61	0.91 ± 0.88	1.13 ± 1.20	0.09 ± 0.09	1.00 ± 0.00
0.9	14973.03 ± 23.12	2.19 ± 1.60	0.90 ± 0.88	1.14 ± 1.22	0.09 ± 0.09	1.00 ± 0.00
0.8	13219.21 ± 24.81	2.21 ± 1.59	0.89 ± 0.88	1.17 ± 1.23	0.09 ± 0.09	1.00 ± 0.00
0.7	11538.73 ± 26.19	2.17 ± 1.50	0.90 ± 0.86	1.12 ± 1.15	0.09 ± 0.09	1.00 ± 0.00
0.6	9944.99 ± 26.67	2.18 ± 1.61	0.89 ± 0.88	1.15 ± 1.21	0.09 ± 0.09	1.00 ± 0.00
0.5	8454.05 ± 25.53	2.18 ± 1.53	0.90 ± 0.87	1.14 ± 1.19	0.09 ± 0.09	1.00 ± 0.00
0.4	7062.36 ± 23.52	2.17 ± 1.54	0.89 ± 0.86	1.14 ± 1.20	0.09 ± 0.09	1.00 ± 0.00
0.3	5729.32 ± 22.12	2.19 ± 1.57	0.89 ± 0.87	1.17 ± 1.22	0.09 ± 0.09	1.00 ± 0.00
0.2	4388.77 ± 20.16	2.17 ± 1.56	0.89 ± 0.87	1.14 ± 1.18	0.09 ± 0.09	1.00 ± 0.00
N = 500						
None	20000.00 ± 0.00	5.93 ± 2.40	3.80 ± 1.43	1.37 ± 1.37	0.38 ± 0.14	1.00 ± 0.00
0.9	15019.94 ± 19.62	5.95 ± 2.39	3.80 ± 1.43	1.37 ± 1.37	0.38 ± 0.14	1.00 ± 0.00
0.8	13272.66 ± 21.06	5.99 ± 2.37	3.84 ± 1.45	1.38 ± 1.36	0.38 ± 0.15	1.00 ± 0.00
0.7	11582.75 ± 22.54	5.98 ± 2.33	3.82 ± 1.45	1.37 ± 1.31	0.38 ± 0.15	1.00 ± 0.00
0.6	9978.88 ± 22.15	5.95 ± 2.32	3.80 ± 1.44	1.35 ± 1.30	0.38 ± 0.14	1.00 ± 0.00
0.5	8480.43 ± 23.58	5.92 ± 2.32	3.80 ± 1.46	1.33 ± 1.26	0.38 ± 0.15	1.00 ± 0.00
0.4	7086.22 ± 23.19	5.91 ± 2.36	3.78 ± 1.48	1.33 ± 1.29	0.38 ± 0.15	1.00 ± 0.00
0.3	5755.76 ± 21.39	6.03 ± 2.39	3.84 ± 1.47	1.38 ± 1.32	0.38 ± 0.15	1.00 ± 0.00
0.2	4416.63 ± 20.10	5.95 ± 2.37	3.83 ± 1.47	1.33 ± 1.27	0.38 ± 0.15	1.00 ± 0.00

			N = 1,000			
None	20000.00 ± 0.00	12.29 ± 2.23	8.81 ± 1.02	1.29 ± 1.25	0.88 ± 0.10	1.00 ± 0.00
0.9	15040.79 ± 16.01	12.29 ± 2.28	8.81 ± 1.03	1.29 ± 1.27	0.88 ± 0.10	1.00 ± 0.00
0.8	13299.53 ± 17.21	12.28 ± 2.35	8.81 ± 1.06	1.29 ± 1.25	0.88 ± 0.11	1.00 ± 0.00
0.7	11602.81 ± 19.44	12.28 ± 2.32	8.81 ± 1.04	1.30 ± 1.26	0.88 ± 0.10	1.00 ± 0.00
0.6	9996.51 ± 20.79	12.24 ± 2.29	8.82 ± 1.04	1.29 ± 1.27	0.88 ± 0.10	1.00 ± 0.00
0.5	8489.93 ± 20.96	12.31 ± 2.28	8.84 ± 1.04	1.28 ± 1.25	0.88 ± 0.10	1.00 ± 0.00
0.4	7096.75 ± 21.11	12.27 ± 2.31	8.81 ± 1.06	1.30 ± 1.25	0.88 ± 0.11	1.00 ± 0.00
0.3	5769.18 ± 20.28	12.25 ± 2.29	8.80 ± 1.05	1.30 ± 1.24	0.88 ± 0.11	1.00 ± 0.00
0.2	4429.58 ± 20.11	12.30 ± 2.34	8.81 ± 1.06	1.32 ± 1.26	0.88 ± 0.11	1.00 ± 0.00

Table 7.22 Mean and standard deviation results for LD clumping using the permutation method for tuning parameter selection for differing percentage of variance explained with N = 500

Pruning threshold	No. of SNPs selected	No. of true positive SNPs	No. of false positive SNPs	Sensitivity	Specificity
Variance Explained = 1%					
None	1.51 ± 1.38	0.42 ± 0.62	1.05 ± 1.14	0.04 ± 0.06	1.00 ± 0.00
0.9	1.56 ± 1.39	0.43 ± 0.64	1.06 ± 1.15	0.04 ± 0.06	1.00 ± 0.00
0.8	1.58 ± 1.39	0.44 ± 0.63	1.08 ± 1.17	0.04 ± 0.06	1.00 ± 0.00
0.7	1.54 ± 1.38	0.44 ± 0.63	1.04 ± 1.17	0.04 ± 0.06	1.00 ± 0.00
0.6	1.58 ± 1.40	0.44 ± 0.64	1.08 ± 1.16	0.04 ± 0.06	1.00 ± 0.00
0.5	1.55 ± 1.36	0.43 ± 0.63	1.06 ± 1.14	0.04 ± 0.06	1.00 ± 0.00
0.4	1.53 ± 1.37	0.43 ± 0.64	1.04 ± 1.13	0.04 ± 0.06	1.00 ± 0.00
0.3	1.58 ± 1.39	0.45 ± 0.63	1.07 ± 1.17	0.04 ± 0.06	1.00 ± 0.00
0.2	1.58 ± 1.43	0.44 ± 0.63	1.08 ± 1.19	0.04 ± 0.06	1.00 ± 0.00
Variance Explained = 2.75%					
None	5.93 ± 2.40	3.80 ± 1.43	1.37 ± 1.37	0.38 ± 0.14	1.00 ± 0.00
0.9	5.95 ± 2.39	3.80 ± 1.43	1.37 ± 1.37	0.38 ± 0.14	1.00 ± 0.00
0.8	5.99 ± 2.37	3.84 ± 1.45	1.38 ± 1.36	0.38 ± 0.15	1.00 ± 0.00
0.7	5.98 ± 2.33	3.82 ± 1.45	1.37 ± 1.31	0.38 ± 0.15	1.00 ± 0.00
0.6	5.95 ± 2.32	3.80 ± 1.44	1.35 ± 1.30	0.38 ± 0.14	1.00 ± 0.00
0.5	5.92 ± 2.32	3.80 ± 1.46	1.33 ± 1.26	0.38 ± 0.15	1.00 ± 0.00
0.4	5.91 ± 2.36	3.78 ± 1.48	1.33 ± 1.29	0.38 ± 0.15	1.00 ± 0.00
0.3	6.03 ± 2.39	3.84 ± 1.47	1.38 ± 1.32	0.38 ± 0.15	1.00 ± 0.00
0.2	5.95 ± 2.37	3.83 ± 1.47	1.33 ± 1.27	0.38 ± 0.15	1.00 ± 0.00
Variance Explained = 5.5%					
None	9.32 ± 2.43	6.67 ± 1.41	1.13 ± 1.24	0.67 ± 0.14	1.00 ± 0.00
0.9	9.32 ± 2.44	6.67 ± 1.41	1.14 ± 1.24	0.67 ± 0.14	1.00 ± 0.00
0.8	9.39 ± 2.41	6.69 ± 1.43	1.17 ± 1.24	0.67 ± 0.14	1.00 ± 0.00
0.7	9.32 ± 2.43	6.67 ± 1.45	1.15 ± 1.21	0.67 ± 0.14	1.00 ± 0.00
0.6	9.32 ± 2.44	6.67 ± 1.45	1.16 ± 1.24	0.67 ± 0.15	1.00 ± 0.00
0.5	9.34 ± 2.40	6.71 ± 1.42	1.12 ± 1.19	0.67 ± 0.14	1.00 ± 0.00
0.4	9.28 ± 2.34	6.68 ± 1.40	1.12 ± 1.17	0.67 ± 0.14	1.00 ± 0.00
0.3	9.36 ± 2.50	6.69 ± 1.42	1.13 ± 1.22	0.67 ± 0.14	1.00 ± 0.00
0.2	9.31 ± 2.42	6.68 ± 1.43	1.13 ± 1.21	0.67 ± 0.14	1.00 ± 0.00

7.5 Conclusion

In this chapter, a simulation study was conducted to assess the performance of a number of pruning methods implemented in my Prune package for variable selection. Data was simulated from the GRAPHIC dataset from a single chromosome. The results showed that the tuning parameter selection method was more influential on variable selection than the pruning method itself. Repeated 10-fold Cross-validation selected a large number of false positives regardless of pruning method (Table 7.5, Table 7.6, Table 7.11, Table 7.12,Table 7.17 and Table 7.18), however the mean number of false positives selected was greater with P-value pruning (Table 7.11 and Table 7.12). This particular combination of pruning method and tuning parameter selection method yielded the lowest specificity rate across all scenarios. This is unsurprising as CV is designed for model prediction rather than variable selection, however the mean number of false positives selected is concerning as the simulation was conducted on a single chromosome rather than genome-wide where the number of false positives selected is likely to increase substantially. Both Cho *et al*. (11) and Yao *et al*. (149) have used a combination of P-value pruning and Cross-validation for tuning parameter selection and both studies also selected a high number of SNPs (129 and 80 respectively). Hong *et al*. also showed that the combination of P-value pruning and tuning parameter selection by CV selects a large number of variables; in this case over 500 of the 1,000 SNPs were selected across four penalised regression methods (146).

10 casual SNPs were simulated which may be a reasonable number of causal variants in a genome-wide study rather than there being 10 causal SNPs in each chromosome. Therefore it is likely that the number of true positive SNPs selected is more representative of a genome-wide study whilst the mean number of false positives may not be as representative. Of the three pruning methods LD pruning produces the highest specificity rate.

The BIC also selected a high proportion of false positive SNPs (Table 7.7, Table 7.8, Table 7.13, Table 7.14, Table 7.19 and Table 7.20). This was particularly the case in the underpowered scenarios (N = 250 or %VAR = 1%) where mean number of false positives selected increased as the pruning threshold increased. With the exception of some high powered scenarios, results showed that the LD based pruning methods increased the mean number of SNPs selected compared to not pruning at all which, in turn increased both the number of true and false positive SNPs selected. This increase was gradually countered by the increase in pruning.

In most scenarios, for both repeated CV and the BIC methods, the mean number of false positives selected increased between the mid-powered scenario and high powered scenario (i.e. between N = 500 and N = 1,000 and between %VAR = 2.5% and 5.5%). This could be partially due to defining any selected SNP with an $r^2 > 0.5$ with a causal as a true positive rather than a lower threshold. It may be the case that SNPs with $0 < r^2 < 0.5$ may be selected due to the LD in the high powered scenario especially at the higher LD pruning thresholds. In this simulation, no pruning window was used which may affect the sensitivity rate. By not implementing a pruning window, there is a small chance that the simulated causal SNP may be in LD with another SNP with a large distance between them by chance and not a true association between alleles. Therefore selection of this SNP may also increase the sensitivity rate with false positives although the $r^2 > 0.5$ threshold helps protect against this situation.

It is recommended that to prune a GWAS dataset to apply the LASSO, LD clumping should be used and the tuning parameter should be selected by the permutation method. The permutation method produced similar results regardless of pruning method or pruning threshold (Table 7.9, Table 7.10, Table 7.15, Table 7.16, Table 7.21 and Table 7.22). The method selected the lowest number of false positives; an average of 1 false positive SNP was selected in every scenario, where both the BIC and repeated CV select a higher number of false positives. Due to the number of false

positives selected on a simulation on a single chromosome for BIC and repeated CV, the permutation method outperforms the other two tuning parameter selection methods in terms of variable selection. Of the three pruning methods LD clumping seems to select a slightly higher mean number of true positive SNPs in most scenarios using the permutation method.

7.6 Summary

In this chapter, I ran a simulation study on the effects of SNP pruning methods, the pruning threshold and the tuning parameter selection method on variable selection. To current knowledge this is the first study that looks at the effects of pruning on variable selection. Results showed that pruning with LD clumping and using the permutation method produced the best performance for variable selection due to the high number of false positive SNPs selected by other methods, especially as both repeated 10-fold CV and the BIC selected a number of false positive SNPs across a single chromosome.

8 Application of the LASSO on the GRAPHIC study with SNP pruning

8.1 Introduction

In Chapter 4, I was unable to apply the LASSO to the full GWAS dataset of the GRAPHIC study (10) due to computational constraints (see section 4.7). I suggested SNP pruning as a step to reduce the number of dimensions in order to fit LASSO models on a genome-wide scale. In Chapter 6, I discussed a number of SNP pruning methods that could be utilised as well as my Prune package which applies these SNP pruning methods. I then followed up by conducting a simulation study applying these SNP pruning methods and to see the effect each pruning method has on variable selection using the LASSO.

In this chapter, I return to the GRAPHIC study and apply the LASSO to the GWAS dataset after pruning SNPs. I firstly re-run the analysis on chromosome 19 to compare the number of SNPs selected by varying tuning parameter selection methods and pruning thresholds. This was to check whether results between the simulation study and the GRAPHIC dataset were consistent with each other, as the phenotype in the simulation study was different to the LDL phenotype in terms of effect sizes and variation. I then select the best combination pruning method and tuning parameter selection methods based on both the simulation study and the application on chromosome 19 and apply them on the full GRAPHIC study dataset. LDL-c was again used as the phenotype and the same quality control procedures discussed in section 4.4 were applied to both analyses.

8.2 Application of the LASSO on chromosome 19 after pruning the dataset

8.2.1 Methods for chromosome 19 study

The dataset consists of 12,376 SNPs and 979 subjects. My Prune package was used to prune the datasets. The three pruning methods were used, LD, P-value and clumping with three pruning thresholds for each pruning method. The LD pruning and clumping methods pruned the dataset with the r^2 measure using thresholds of 0.8, 0.5 and 0.2 and a pruning window of 500 adjacent SNPs. The P-value pruning method used thresholds of 0.2, 0.05 and 0.02. A random starting position for LD pruning was selected and the same start position was used for all pruning thresholds. Section 4.8.1 outlines the procedures for fitting the LASSO using the three tuning parameter selection methods, repeated 10-fold Cross-validation, BIC and the permutation method. An increase in the pruning threshold was again defined as a decrease in the threshold value. The same imputation procedures were also implemented (see section 4.8.1). The LASSO model was fitted using glmnet (53).

8.2.2 Results of chromosome 19 study

Table 8.1, Table 8.3 and Table 8.4 show the number of SNPs selected for each combination of SNP pruning method, tuning parameter selection method and pruning threshold. The results for LD pruning were consistent with the SNP pruning simulation. The number of SNPs selected by both repeated CV and BIC decreased as the pruning threshold increased, while the permutation method remained stable (Table 8.1). LD pruning allows any SNP to be pruned out therefore selected SNPs using a lower LD threshold could be pruned as the pruning threshold increases. This was the case in this analysis reducing the number of associations in the dataset but did not affect the number of SNPs and which regions were selected using the permutation method.

Table 8.2 shows the SNPs selected using the permutation method for LD pruning. The table shows that as pruning increases and if the selected SNP was pruned out of the dataset, this SNP was replaced with another SNP in the same region and in LD with the pruned out SNP ($r^2 > 0.53$ in all cases). The BIC method selected the same four SNPs as the permutation method shown in Table 8.2 for the pruning thresholds of $r^2 < 0.8$ and $r^2 < 0.5$. For the $r^2 < 0.2$ threshold the two SNPs on the APOE gene were selected; rs4420638 (p = 1.58E-07) and rs445925 (p = 3.37E-06).

Table 8.1 Number of SNPs selected on chromosome 19 after pruning the dataset by LD using various forms of tuning parameter selection methods and LD pruning thresholds

LD pruning threshold (r^2)	Number of SNPs in dataset after pruning	Cross-validation	BIC	Permutation method
No pruning	12,376	41	4	4
0.8	8,615	41	4	4
0.5	6,190	31	4	4
0.2	3,655	11	2	4

Table 8.2 SNPs selected on chromosome 19 for varying levels of LD pruning and the permutation method for tuning parameter selection.

LD pruning threshold	Selected SNP	Base position	P-value	LD between SNPs (r^2)
DNM2 - CARM1				
No pruning	rs17001002	10,948,031	1.25E-05	-
0.8	rs11881156	10,950,125	1.79E-05	0.993
0.5	rs11881156	10,950,125	1.79E-05	-
0.2	rs11881156	10,950,125	1.79E-05	-
ZNF529 - ZNF567				
No pruning	rs10402182	37,160,529	4.53E-06	-
0.8	rs1525133	37,199,250	4.53E-06	1
0.5	rs2967440	37,059,215	5.87E-06	0.981
0.2	rs2967436	37,059,215	5.87E-06	0.539
APOE				
No pruning	rs7412	45,412,079	1.70E-12	-
0.8	rs7412	45,412,079	1.70E-12	-
0.5	rs7412	45,412,079	1.70E-12	-
0.2	rs445925	45,415,640	3.37E-06	0.712
APOE				
No pruning	rs4420638	45,422,946	1.58E-07	-
0.8	rs4420638	45,422,946	1.58E-07	-
0.5	rs4420638	45,422,946	1.58E-07	-
0.2	rs4420638	45,422,946	1.58E-07	-

Like the simulation study in the previous chapter the combination of P-value pruning and repeated CV performed particularly poorly (Table 8.3). Using a $p < 0.2$ threshold, the LASSO model selected 770 SNPs from a dataset of 2,600 SNPs (29.62% of all SNPs after pruning). The proportion of SNPs selected increased as the pruning threshold increased and with a threshold of $p < 0.02$, 59.80% of SNPs (183 SNPs from a dataset of 306) were selected. In fact, the numbers of SNPs selected in this case were comparable to a similar scenario in the simulation study (see N = 1,000 in Table 7.11) even though this analysis consisted of a smaller dataset after pruning than the simulation study. Like the results of the simulation study, the number of SNPs selected by the BIC

method was similar regardless of P-value pruning threshold. The same four SNPs (rs17001002, rs10402182, rs7412 and rs4420638, see Table 4.18) were selected regardless of P-value pruning threshold. Interestingly however, the number of SNPs selected increased using the permutation method as the pruning threshold increased compared to the simulation study, where the numbers of SNPs selected remained stable regardless of pruning threshold.

Table 8.3 Number of SNPs selected on chromosome 19 after pruning the dataset by P-value using various forms of tuning parameter selection methods and P-value pruning thresholds

P-value pruning threshold	Number of SNPs in dataset after pruning	Cross-validation	BIC	Permutation method
No pruning	12,376	41	4	4
0.2	2,600	770	4	8
0.05	719	358	4	16
0.02	306	183	4	19

The results for LD clumping (Table 8.4) also showed similar results with the simulation study. The number of SNPs selected greatly increased using repeated CV as the pruning threshold increased while both the BIC and permutation methods remained relatively stable. Both methods selected the same SNPs for each pruning threshold. On top of the four SNPs (rs17001002, rs10402182, rs7412 and rs4420638, see Table 4.18) selected by these two methods a fifth SNP; rs10853810 ($p = 0.000166$) was selected when a LD clumping threshold of $r^2 < 0.2$ was applied.

Table 8.4 Number of SNPs selected on chromosome 19 after pruning the dataset by LD clumping using various forms of tuning parameter selection methods and LD clumping pruning thresholds

LD clumping threshold (r^2)	Number of SNPs in dataset after pruning	Cross-validation	BIC	Permutation
No pruning	12376	41	4	4
0.8	8595	41	4	4
0.5	6109	45	4	4
0.2	3539	93	5	5

8.2.3 Conclusion from the chromosome 19 study

With the exception of using the permutation method after P-value pruning the results of the analysis of various SNP pruning methods and tuning parameter selection methods showed similarities to the simulation study in the previous chapter.

The combination of P-value pruning and permutation method increased the number of SNPs selected as the pruning threshold increased while the number of SNPs selected in the simulation study remained stable, much like the number of SNPs selected using the permutation method with LD pruning and clumping. This is illustrated in Figure 8.2 which shows the histogram for each combination of pruning method and threshold for the permutation method. The vertical red line represents the median λ selected across the 100 repetitions. The median estimate is very similar for the LD pruning and clumping methods as the pruning threshold changes whereas the median λ decreases as the pruning threshold increases for P-value pruning. A similar pattern is seen for repeated CV after pruning by LD clumping (Figure 8.1) where the number of SNPs selected also increased.

Both the BIC and permutation method showed consistent selection of SNPs and/or regions for most SNP pruning methods seem to be similar for variable selection. P-value pruning however should not be considered as a method of pruning for the LASSO as a number of false positives may be selected (Table 8.3). LD clumping is likely to select a larger model for LD pruning, as this method ensures that highly associated SNPs are not pruned although the results of this analysis show that SNPs in high LD with the top SNP in a region may be selected instead using LD pruning (Table 8.2).

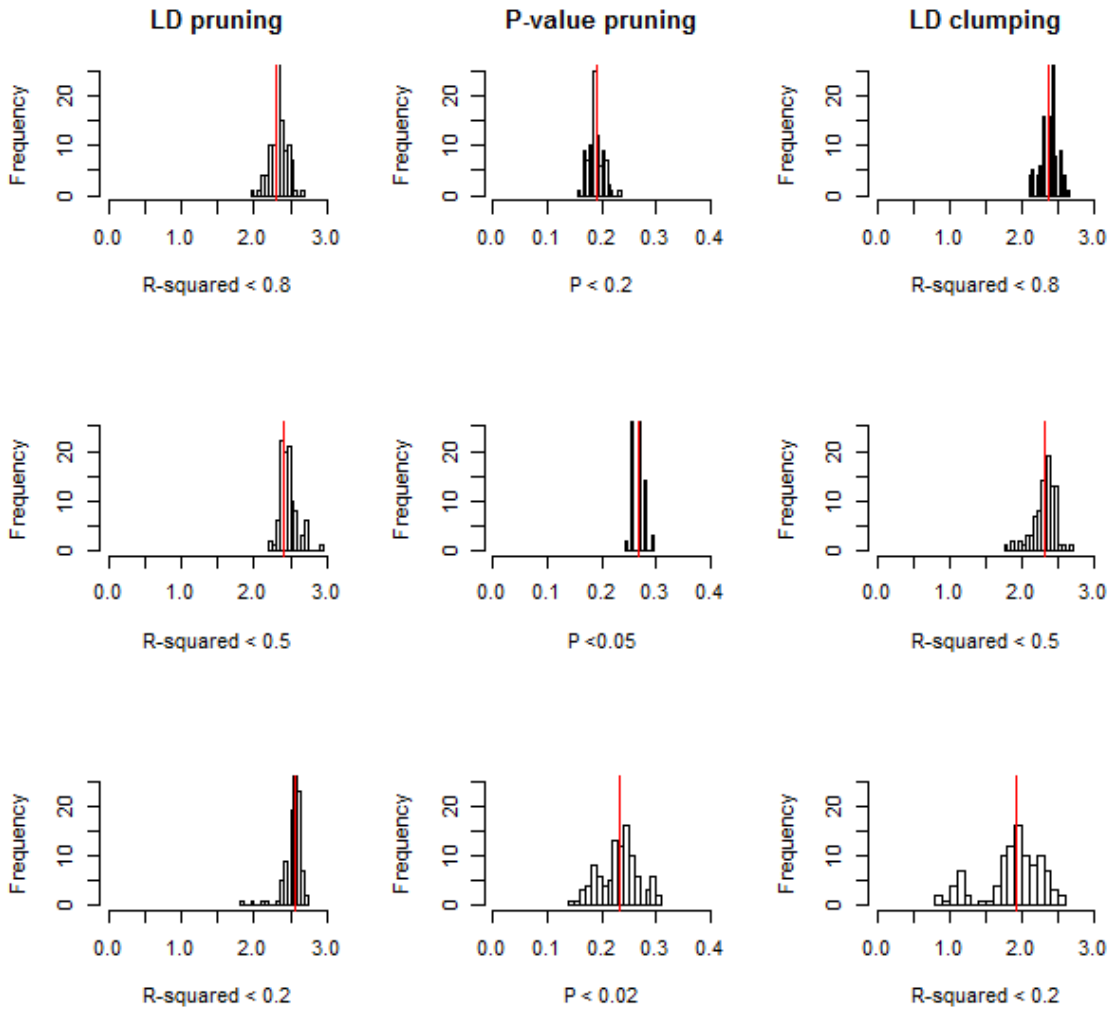


Figure 8.1 Histogram of 100 lambdas estimates using Cross-validation for each pruning method and pruning threshold. The red vertical line represents the median estimate.

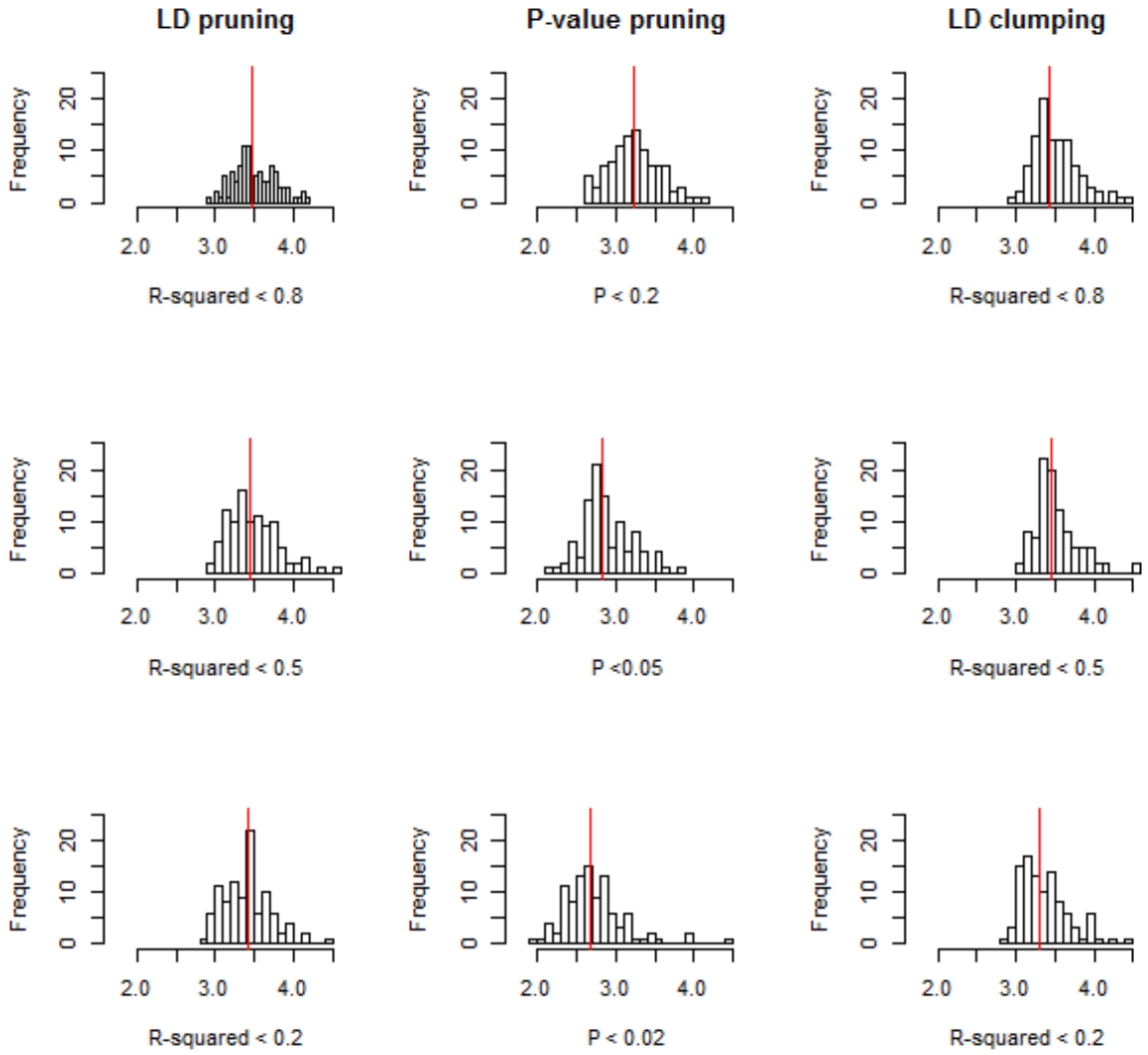


Figure 8.2 Histogram of 100 lambdas estimates using the permutation method for each pruning method and pruning threshold. The red vertical line represents the median estimate.

8.3 Genome-wide association study on the GRAPHIC using the LASSO

8.3.1 Methods for GWAS

After quality control the GWAS dataset consisted of 591,774 SNPs and 979 subjects (see section 4.4), LDL-c was again used as the phenotype. LD clumping was used to prune the GRAPHIC GWAS dataset as this method ensures that the most statistically significant signals remain in the dataset. Both BIC and the permutation methods were selected as tuning parameter selection methods. These methods have shown that there is little difference in terms of variable selection across varying levels of pruning thresholds when pruning by LD clumping. For this reason a pruning threshold of $r^2 < 0.2$ was selected with a window size of 500 SNPs. Each chromosome was pruned separately to allow a random starting position for pruning on each chromosome. The time taken to prune each chromosome varied between 7 minutes 43 seconds for chromosome 1 and 1 minute and 53 seconds for chromosome 22. Table 8.5 shows the number of SNPs remaining in each chromosome after pruning. A total of 138,812 remained after pruning (23.46% of all SNPs). Missing genotypes were again imputed with the median genotype value. Figure 8.3 plots the univariate P-values of each SNP before and after imuputation, again there is little difference between the P-values in most SNPS (mean absolute difference = 0.004). 162 SNPs were removed as they had an absolute difference P-value > 0.1 after imputation. The BIC values were calculated across 625 λ values with an increase 0.01 each time. The permutation method used 200 repetitions. The LASSO model was fitted using glmnet (53).

Table 8.5 Number of SNPs remaining after pruning the GRAPHIC study dataset using the Prune package with window size = 500 and pruning threshold of $r^2 < 0.2$

Chromosome	No. of SNPs on chromosome	No. of SNPs remaining after LD clumping
1	48,494	11,304
2	47,899	11,193
3	39,615	9,023
4	34,217	8,193
5	35,870	8,354
6	40,655	8,403
7	32,235	7,466
8	31,753	6,982
9	28,139	6,535
10	32,500	7,254
11	30,629	6,832
12	29,614	6,987
13	23,049	5,302
14	19,492	4,661
15	18,258	4,451
16	19,114	4,765
17	16,761	4,444
18	17,962	4,515
19	12,376	3,539
20	15,560	3,934
21	8,741	2,255
22	8,841	2,420
Total	591,774	138,812

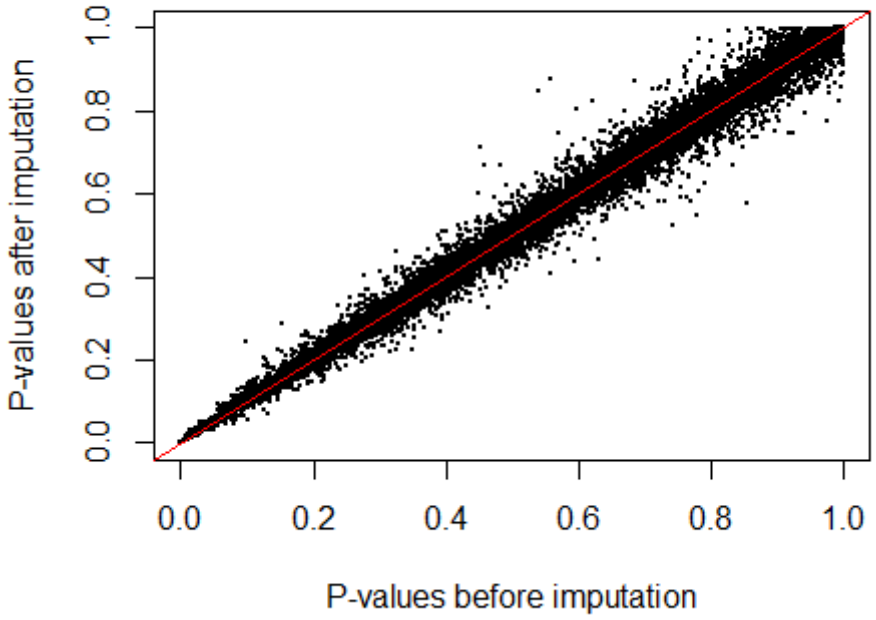


Figure 8.3 Scatter plot comparing P-values for each SNP before and after imputation. Imputation was conducted by replacing missing genotype with the median genotype from the population. The red diagonal line represents the line if there is no change in P-values.

8.3.2 Results of GWAS

Figure 8.4 shows the coefficient path plot for the LASSO model fitted to the GRAPHIC dataset. There is clearly one large SNP with a large association with LDL compared to the remaining SNPs. This SNP was rs7412, the top SNP by P-value (Figure 4.7) and the only SNP selected by the Bonferroni correction method in section 4.5.2. The BIC method however selected a $\lambda = 6.25$ which returned a null model, therefore no SNPs were selected by this method. The permutation method selected a median $\lambda = 4.733$ (mean = 4.770, S.E. = 0.225) which selected one SNP; rs7412. Figure 8.5 shows the histogram of the 200 λ estimates using the permutation method. For the next SNP

(rs4420638) to enter the model a λ < 4.672 was required which was close to the median estimate.

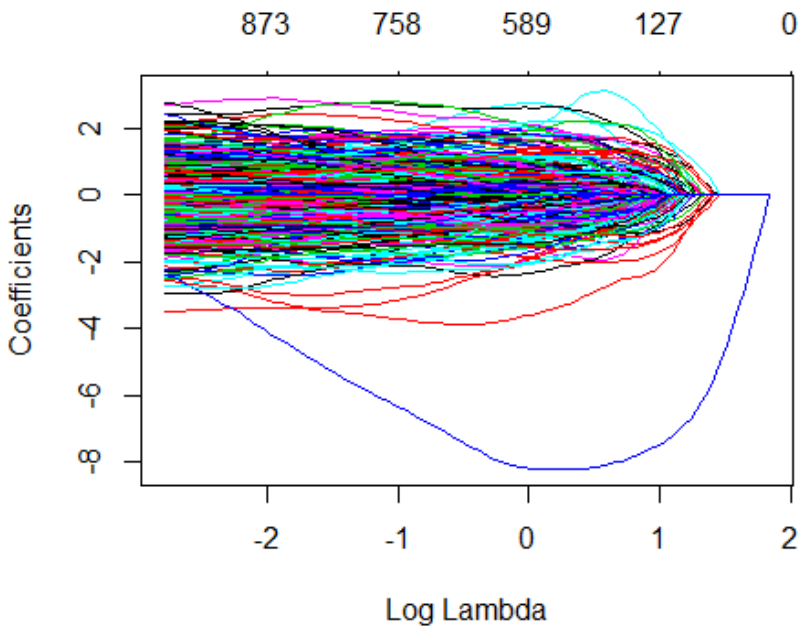


Figure 8.4 Coefficient path plot for the LASSO on the GRAPHIC study. Each line represents a SNP and the path shows the βcoefficient on the y-axis as the penalty (on a $log(\lambda)$ scale) increases on the bottom x-axis. The top x-axis shows the number of SNPs remaining in the model at each $log(\lambda)$ penalty value.

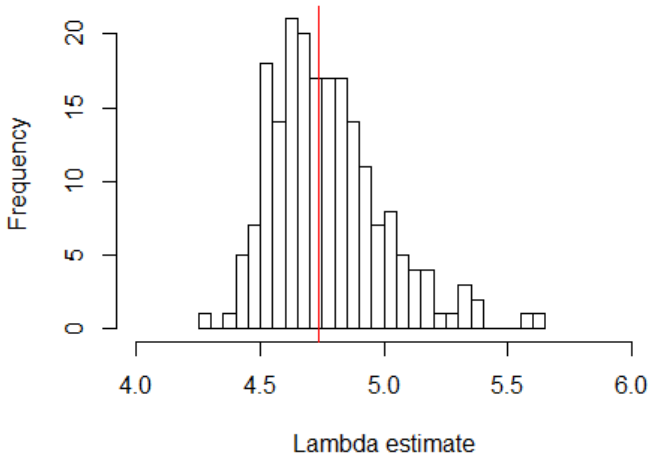


Figure 8.5 Histogram of 200 lambdas estimates using the permutation method. The red vertical line represents the median estimate.

8.4 Discussion

In this chapter, I conducted a GWAS study on the GRAPHIC study using the LASSO after pruning the dataset. Pruning was required to reduce the number of SNPs in the dataset in order to make the analysis more computationally viable (see section 4.7). The pruning was not needed to remove SNPs in correlation as there is evidence that the LASSO is able to handle correlated data in both the literature (24-26) and in previous chapters (see section 4.8.2.3).

LD clumping was used as the method for pruning the dataset from my Prune package. This allowed greater flexibility for pruning as each chromosome was allowed to be pruned separately, and each with a random starting position for pruning. The pruning process in total took just over an hour in time, in contrast pruning in PLINK (19,20) using the same options (window size = 500, step size = 1, threshold = $r^2 < 0.2$) took over 6 days, again highlighting the usefulness of the Prune package in terms of time

taken to prune a large dataset. Pruning if performed on each chromosome separately could be run in parallel with each other, further reducing the computational time.

Pruning by LD clumping ensured that the top SNPs by P-value remained in the dataset especially given the high pruning threshold used. Results on chromosome 19 showed that the LASSO performed well in selecting associated regions when the top SNP by P-value was removed from the data by LD pruning (Table 8.2).

The BIC and permutation methods were used for tuning parameter selection as these methods have performed well in terms of variable selection. The BIC method, which is the more conservative method of the two did not select any SNP while the permutation method after selecting the median value of 200 repetitions selected a single SNP; rs7412 which showed the strongest association with LDL in the dataset. The λ estimate permutation method was close to selecting more SNPs; 97 of the 200 λ estimates selected more than one SNP therefore a greater number of repetitions could have been used to provide greater accuracy. In Chapter 4, the Bonferroni correction and FDR methods were applied to GRAPHIC study dataset without pruning. The Bonferroni correction method also selected rs7412 (see section 4.5.2) while the FDR method selected both rs7412 and rs4420638 (see section 4.6.2). Both associations for rs7412 and rs4420638 have been replicated in previous studies (164,166,172,176,178,180,182,183,187).

9 Applications of integrative analyses in penalised regression

9.1 Introduction

In recent years there has been a shift from GWAS on single datasets to work performed in consortia that collaborate to combine multiple datasets in order to increase the sample size and power to detect associations. The large increase in sample size would again lead to computational issue using the LASSO and pruning would certainly be required. In section 2.3.5, I briefly discuss the meta-analysis in penalised regression and in particular the difficulty in combing summary statistics from penalised estimates. Given the difficulty in combining LASSO summary estimates into a meta-analysis, there has been very little work done in this field (37). An alternative method to meta-analysis is integrative analysis. Integrative analyses require individual level data (ILD) for each study to pool together for analysis. This differs from meta-analysis which analyses each dataset individually then pools summary statistics together.

In this chapter, I firstly review the current literature for integrative analysis based methods using penalised regression. I follow up by conducting a simulation study comparing meta-LASSO method against the stacked LASSO and separate LASSO. The stacked LASSO pools all datasets together and fits a LASSO model without regard of heterogeneity and the separate LASSO that applies the LASSO individually first then pools the summary results together.

9.2 Integrative analysis in penalised regression

Although it is difficult and costly to obtain individual level data (ILD) analyses are considered the gold standard (235). Curran and Hussong discuss the potential advantages of integrating data which include the ability to replicate results in studies, increased statistical power and ability to explore between-study heterogeneity (35). Lambert *et al.* ran simulations comparing meta-regressions using summary statistics and individual patient data (IPD) and concluded that the IPD analyses result in a higher power to detect interaction effects (236). Berlin *et al.* showed that meta-analysis based methods also may fail to detect differences between subgroups that IPD analyses are able to find (237).

There have been a number of methods that integrate datasets and analyse using penalised regression in gene expression data. There is high heterogeneity between studies in gene expression due to varying experimental factors and arrays leading to varying outcome measurements. Often transformations of the expression values is required (238), these transformations known as "intensity approaches". Huang *et al.* showed by simulation on gene expression data that an intensity approach which combines all the dataset and applies the elastic net, outperforms meta-analysis that applies the elastic net in individual datasets (239).

9.2.1 Variations of the group LASSO for integrative analysis

In section 2.3.3, I reviewed the group LASSO (29) as a method to group desired variables within a single dataset, shown in (2.7). $G_1, \dots, G_K$ denotes the pre-defined groups of variables and $i = \{1, \dots N\}$ denotes the ith subject.

$$\hat{\beta}(\lambda) = arg\min_{\beta} \frac{1}{2N} \sum_{i=1}^{N} \left(y_i - \sum_{k=1}^{K} x_{ik} G_k \right)^2 + \lambda \sum_{k=1}^{K} \|G_k\|_2 \qquad (9.1)$$

Ma *et al.* proposed an extension to the group LASSO by grouping β estimates across multiple datasets $H_1, \ldots, H_D$ and applied this to pancreatic cancer studies (240). The study proposed two methods (9.2); the group LASSO (δ = 1) and group bridge LASSO (δ = 0.5).

$$\hat{\beta}(\lambda) = \arg\min_{\beta} \frac{1}{2N} \sum_{d=1}^{D} \sum_{i=1}^{N} \left(y_{di} - \sum_{d=1}^{D} x_{dij} \beta_{dj} \right)^2 + \lambda \sum_{j=1}^{P} \|\beta_j\|_2^{\delta} \qquad (9.2)$$

where,

$\delta =$ the bridge penalty

$$\|\beta_j\|_2 = \left[\sum_{d=1}^{D} (\beta_{dj})^2 \right]^{\frac{1}{2}}$$

$j = \{1, \ldots P\}$ denotes the j[th] SNP and $d = \{1, \ldots D\}$ represents the d[th] dataset.

The penalty allows each β_{dj} to be estimated within individual datasets, and then grouped and penalised across datasets. The paper provides an algorithm to compute the group bridge LASSO that can be solved using Least Angle Regression (LARS), the group LASSO model can be fitted using the coordinate descent algorithm (241) (See section 2.4.2). The study compares integrative analysis of both the group LASSO and bridge group LASSO. The LASSO and bridge penalties are also applied on individual studies and variable selection is defined as when a gene is selected in at least one study. K-fold Cross-validation was used for variable selection in this study. It is therefore unsurprising that the results showed that the analysis on individual datasets over selects the number of variables and hence includes a large number of false positives (See Table 1). There integrative analyses methods show slightly superior performance than the intensity approaches in most simulated scenarios. The bridge

penalty outperformed the LASSO penalty as the bridge consistently selected a lower number of genes in the final model while maintaining a similar number of true positives. Integrative analysis with the bridge penalty also consistently showed the lowest prediction errors.

A previous study by a number of the same authors also compared the group bridge penalty (δ = 0.5) with AIC and BIC as the tuning parameter selection methods against the group LASSO, and another of other grouping methods, including the group LASSO, which used BIC and Mallows C_p as the tuning parameter selection method (242). Results were similar to the Ma *et al.* study (240) with the group bridge LASSO outperforming the competing methods in terms of both variable selection and prediction with the BIC especially performing well (See Table 1 (242)).

As discussed by the authors, the aim of the bridge group LASSO is to identify "a common set of covariates across multiple studies" by pooling β_{dj} estimates across studies. Therefore the method does not allow any selection within studies (31,240). Another grouping method that has been suggested is the sparse group LASSO (84). This method uses two penalties and is similar to the elastic net, the first is a group penalty that penalises across datasets and the second is a LASSO penalty on the variables within each dataset (9.3).

$$\hat{\beta}(\lambda) = \arg\min_{\beta} \frac{1}{2N} \sum_{d=1}^{D} \sum_{i=1}^{N} \left(y_{di} - \sum_{j=1}^{P} x_{dij} \beta_{dj} \right)^2 + \lambda_1 \sum_{j=1}^{P} \left\| \beta_{dj} \right\|_2 + \lambda_2 \left\| \beta_{dj} \right\|_1 \quad (9.3)$$

This proposed method, allows for variables to be penalised both within and across datasets and can be solved using coordinate descent (243). Lin *et al.* used the sparse group LASSO to pool together multiple diverse datasets, in this case, SNP datasets and

gene expression studies (243). 3 datasets of 15,235 SNPs were simulated from chromosome 22 with the 3 phenotype and 3 expression datasets simulated based from the simulated SNP data. The authors use a 30 x 30 search grid to find the optimal combination of λ_1 and λ_2.The study compares the sparse group LASSO with a sparse group ridge regression and meta-analysis, each method was applied to the SNP data and gene expression datasets separately and were then combined. Variable selection was based on a gene level although the causal variants were simulated on individual SNPs. Results show that the sparse group LASSO performed better than the competing methods, especially when combining all datasets (See Figure 1 and 2 (243)). The group bridge method was not considered as previous simulations suggested poor performance compared to sparse group ridge in a single SNP dataset setting (244). Park *et al*. used the sparse group LASSO with latent variables to account for the overlapping between groups in single dataset analyses (245).

9.2.2 The meta-LASSO method for integrative analysis

The meta-LASSO method, proposed by Li *et al*. (246) incorporates a dual penalty much like other variations of the LASSO such as the elastic net (18). The study applies the meta-LASSO to gene expression data analysing the expression 88 genes across 5 datasets of immune cells of subjects with either atherosclerosis or cardiac events such as myocardial infarction or stroke. Therefore each dataset is composed of a binary phenotype y_{di} and a vector of gene expression profiles of *P* genes. By assuming the conditional probability that $y_{di} = 1$ given the vector of gene expression, then y_{di} follows a logistic regression model (9.4). To account for heterogeneity, the effect estimate β_{dj} is parametrized (9.5). γ_j denotes the overall effect of the j^{th} gene across all datasets and the ζ_{dj} term is the effect difference estimate that accounts for heterogeneity on the j^{th} gene in the d^{th} dataset. β_{dj} is reparameterized by multiplying γ_j and ζ_{dj}. Therefore if there is no heterogeneity then ζ_{dj} = 1 and $\beta_{dj} = \gamma_j$. As gene expression values only take positive values, a constraint is placed such that $\zeta_{dj} \geq 0$.

The overall estimate, γ_j take positive values also therefore produce positive β_{dj} estimates only.

$$log\left(\frac{\Pr(y_{di}=1|x_{di})}{\Pr(y_{di}=0\,|x_{di})}\right) = \beta_{0d} + x_{di}\beta_{dj} \tag{9.4}$$

$$\beta_{dj} = \gamma_j\zeta_{dj} \qquad d=1,\dots,D; j=1,\dots,P \tag{9.5}$$

such that:

$$\zeta_{dj} \geq 0$$

$$\max_{\beta_0,\gamma,\zeta}\left\{\sum_{d=1}^{D} L_d(\alpha_d,\gamma,\zeta_d) - \lambda_\gamma \sum_{j=1}^{P}|\gamma_j| - \lambda_\zeta \sum_{j=1}^{P}\sum_{d=1}^{D}|\zeta_{dj}|\right\} \tag{9.6}$$

where,

$\gamma = \{\gamma_j\}$ and $L_d(\alpha_d,\gamma,\zeta_d)$ is the log-likelihood function s.t.:

$$L_d(\alpha_d,\gamma,\zeta_d) = \sum_{i=1}^{N_d} y_{di}\{\alpha_d + x_{di}^T(\gamma.\zeta_d)\} - log[1 + exp\{\alpha_d + x_{di}^T(\gamma.\zeta_d)\}]$$

The meta-LASSO analysis can then solved by applying a penalty on both of the γ_j and ζ_{dj} components each with a separate tuning parameter λ_γ and λ_ζ(9.6). The loss function $L_d(\alpha_d,\gamma,\zeta_d)$ can take the form of other distributions such as normal or poisson distribution. The $\lambda_\gamma$tuning parameter controls variable on the overall gene effect across all *m* datasets and therefore can remove genes from all datasets if they are deemed no to be associated. The λ_ζ tuning parameter controls variable selection at an individual dataset level. The authors also show that (9.6) can be further simplified into one tuning parameter (9.7) where the penalty applied on $|\gamma_j|$ is 1.

$$\max_{\alpha,\gamma,\zeta}\left\{\sum_{d=1}^{D} l_d(\alpha_d,\gamma,\zeta_d) - 1\sum_{j=1}^{P}|\gamma_j| - \lambda\sum_{j=1}^{P}\sum_{d=1}^{D}|\zeta_{dj}|\right\} \tag{9.7}$$

where

$$\lambda = \lambda_{\gamma}.\lambda_{\zeta}$$

The studies compared the meta-LASSO to a number of other methods such as the "separate LASSO", the "stack LASSO", the group LASSO, the adaptively weighting (AW) method, Fisher's method and both Fixed (FEM) and Random (RAM) effects meta-analysis models in a simulation study. The separate LASSO method used in the study fits a LASSO model to each dataset separately, however it is not clear how results from each dataset are combined or if they are combined at all. The stacked LASSO method assumes that there is no heterogeneity between studies and hence β_{dj} is the same for all *d*. Datasets are combined together to fit a "stacked LASSO" model which is in effect a standard LASSO model as the data has been pooled together (9.8).

$$\sum_{d=1}^{D} l_d(\alpha_d, \beta) \; - \lambda \sum_{j=1}^{P} |\beta_j| \tag{9.8}$$

where

β_j = The effect estimate across all *d* datasets

For the meta-LASSO and other penalised regression methods used in the simulation study, the tuning parameter was selected by minimizing the BIC (9.9). For the AW method, Fisher's method, FEM and RAM methods, a gene is selected if the gene is found to be significant across all studies

$$BIC(\lambda) = \sum_{d=1}^{D} \{-2l_d(\hat{\beta}_{d,\lambda}) + \; DF_d \log(N_d)\} \tag{9.9}$$

The study used sensitivity and specificity as its outcome variables across varying levels of heterogeneity among datasets over 100 repetitions. Ten studies each with 1,000 genes were simulated each with a sample size of 50 subjects. To simulate heterogeneity between datasets, the authors slightly modified the parameterization equation (9.5) to include γ_{dj}^{*} (9.10). The modification is to allow some variance between the overall estimates between studies.

$$\beta_{dj}^{*} = \gamma_{dj}^{*}\zeta_{dj}^{*} \qquad 1, \dots, M; j = 1, \dots, 10 \qquad (9.10)$$

This was performed by allowing $\gamma_{dj}^{*} \sim \mathrm{N}(3, 0.5)^2$ and ζ_{dj}^{*} follows a Bernoulli distribution with probability (π_0). The inclusion of the Bernoulli distribution allowed each of the 10 important genes to either have an effect $\beta_{dj}^{*} \sim \mathrm{N}(3, 0.5)^2$ with probability π_0 or $\beta_{dj}^{*} = 0$ with probability $1 - \pi_0$. π_0 took three values: 0.2, 0.5 and 0.9 that denote high, mid and low levels of heterogeneity.

For the high and mid ranged levels of heterogeneity ($\pi_0 = 0.2 \text{ and } 0.5$ respectively) the meta-LASSO clearly outperformed the other methods (See Table 2 (246)). At a low level of heterogeneity ($\pi_0 = 0.9$) the stack LASSO and FEM methods outperformed the meta-LASSO as both had higher rates of sensitivity and specificity. When homogeneity is strong there is very little variance between datasets and hence each simulated dataset would produce similar summary statistics such as beta effect coefficients and P-values. The combination of *m* datasets would increase the power in the analysis and with the beta effect coefficients and P-values being similar across the datasets, a high sensitivity and specificity would be expected. The meta-LASSO would suffer in this scenario as the ζ_{dj} penalty would have very little impact as there is very little heterogeneity between the datasets and there is little need to penalise within a dataset. The group LASSO also performed well against competing methods for the high and mid ranged levels of heterogeneity, especially in terms of sensitivity, but did not perform well in the low heterogeneity setting. The high sensitivity and specificity of most of the methods in the simulation in the low heterogeneity scenario suggests that large effect sizes have been simulated and brings into question if this method works when smaller effect sizes, or lower powered associations are simulated.

9.2.2.1 Algorithm to fit meta-LASSO models

The authors provide a brief algorithm to fit meta-LASSO models to solve the function shown in equation (9.7). The algorithm is based to optimising γ_j and ζ_{dj} separately, both can be optimised using coordinate descent algorithm (see section 2.5). γ_j is optimised using the "stacked" LASSO method discussed in the section above, where as ζ_{dj} is optimised separately for each *d*

The solutions for γ_j and ζ_{dj} and are derived using the same calculations shown in section 2.5.2. The algorithm is outlined in Table 9.1.

Table 9.1 Algorithm to fit meta-LASSO models for logistic regression

- Let $y_{id} = \text{A set of phenotype}$
- Let γ_j = A 1 X P matrix of overall estimates
- Let ζ_{dj} = A D X P matrix of effect difference estimates
- Let β_{dj} = A D X P matrix of effect estimates s.t. $\beta_{dj} = \gamma_j \zeta_{dj}$
- Let β_{0d} = A vector of intercept estimates

1. Set $\begin{cases} \beta_{dj} = 0 \\ \gamma_j = 0 \\ \zeta_{dj} = 1 \end{cases}$

   $for\ all\ \mathrm{d} = \{1, \dots, \mathrm{D}\}, \mathrm{j} = \{1, \dots, \mathrm{P}\}$

2. Set $Old.Beta_{dj} = \beta_{dj}$
3. Calculate $\tilde{x}_i = x_{ijd}\, \zeta_{dj}$
4. Update γ_j using coordinate descent (section 2.5.2) by "stacking" $\tilde{x}_{ij1}$, … $\tilde{x}_{ijd}$ and y_{i1}, …,y_{id}
5. Calculate $\hat{x}_{id} = x_{ijd}\, \gamma_j$
6. Update ζ_{dj} using coordinate descent (section 2.5.2) by setting x = $\hat{x}_{id}$ for each *d* and setting a constraint s.t. $\zeta_{dj} \geq 0$
7. Update $\beta_{dj} = \gamma_j \zeta_{dj}$
8. Repeat steps 2 – 7 until $max\left|\beta_{dj} - Old.Beta_{dj}\right| < 0.00001$

9.2.3 The Data Shared LASSO for integrative analysis

Gross and Tibshirani propose the Data Shared LASSO (DSL) (247) which is a similar approach to the meta-LASSO. Following the notation from the previous section, the reparameterisation used assumes an additive relationship between the overall estimate and the heterogeneity estimate (9.11) and is solved by minimising (9.12).

$$\beta_{dj} = \gamma_j + \zeta_{dj} \qquad d = 1, \dots, D; j = 1, \dots, P \qquad (9.11)$$

$$min_\beta f(\theta) = \frac{1}{2N} \sum_{d=1}^{D} \sum_{i=1}^{N} \left(y_{id} - \sum_{j=1}^{P} x_{ijd} \beta_{jd} \right)^2 + \lambda \left(\|\gamma_j\|_1 + \sum_{d=1}^{D} r_d \|\zeta_{dj}\|_1 \right) \qquad (9.12)$$

The r_d term controls the strength of the penalty of the heterogeneity penalty relative to the overall penalty, similar to the α term used in elastic net (18). $r_d = \frac{1}{\sqrt{D}}$ is suggested assuming $D > 3$. The authors aim is to identify common variables across and within subgroups from a larger dataset. The DSL is not tested by simulation but only on a real-life dataset. Although this study uses a dataset of movie reviews, sub grouped into genres of drama, comedy and horror, it can easily be applied in an integrative analysis setting where each "subgroup" is a separate dataset and hence penalise both within and across datasets. Likewise the meta-LASSO could potentially combine subgroups as "datasets" for analysis.

The authors used a real-life dataset for analysis for variable prediction rather than selection and were compared to a stacked LASSO ($r = \infty$) and the separate LASSO$\left(r = \frac{1}{4}\right)$. The dataset was split into a training set (n = 16,386) and test set (n = 18,109) with Mean Squared Error (MSE) calculated from the test set. Results showed that the DSL produced the lowest MSE in the all, drama and horror genres with the stacked LASSO performing slightly better in comedy. With the exception of the horror genre however the difference in MSE between the 3 methods is small (See Table 1).

9.2.3.1 Fitting the Data Shared LASSO

The DSL can be easily fitted using coordinate descent by using an augmented data approach. This approach pools together all the y_d phenotypes into a single vector, an augmented matrix Z that is created using the predictor variables (9.13) where each cell in the matrix represents either an N x P matrix of $0's$ or a dataset x_d of the same dimensions. This makes the total dimensions of $Z = ND\ x\ P(D+1)$.

$$Y = \begin{pmatrix} y_1 \\ y_2 \\ \mathrm{M} \\ \mathrm{M} \\ y_D \end{pmatrix} \quad Z = \begin{pmatrix} \mathrm{x}_1 & r_1 x_1 & 0 & & \Lambda & 0 \\ x_2 & 0 & r_2 x_2 & & \Lambda & 0 \\ \mathrm{M} & \mathrm{M} & \mathrm{M} & \mathrm{O} & & \mathrm{M} \\ \mathrm{M} & \mathrm{M} & \mathrm{M} & & \mathrm{O} & \mathrm{M} \\ x_D & 0 & 0 & \Lambda & \Lambda & r_D x_D \end{pmatrix} \tag{9.13}$$

The augmentation allows the DSL method to be fitted using coordinate descent and can be fitted using the glmnet package. However in a GWAS setting this may prove difficult especially as both N and P are large for each dataset. Some GWAS datasets can potentially contain millions of SNPs and therefore lead to problems with memory in R (see section 4.7) and therefore is likely to be an unviable method with this particular algorithm.

9.3 Simulation study comparing the meta-LASSO against the LASSO

In this section, I run a simulation comparing the meta-LASSO against the stacked LASSO and separate methods on SNP datasets. The aim of this simulation is to assess the performance of the meta-LASSO in a genetic association setting where the power to select a causal SNP is often low. The stacked LASSO pools all datasets together without

regard for heterogeneity. The separate LASSO fits the LASSO separately on each dataset. The comparison against the stacked LASSO is to determine whether there is any advantage of the meta-LASSO in a GWAS setting as opposed to just combining all the datasets together.

The meta-LASSO method can be applied to a GWAS study by redefining the gene effect estimates to SNP effect estimates. Using the parametrization (9.5), γ_j is defined as the overall SNP effect of the j^{th} SNP across all *d* datasets, ζ_{dj} still accounts for heterogeneity across datasets on the j^{th} SNP in the d^{th} dataset. β_{dj} denotes the SNP effect of the j^{th} SNP in the d^{th} dataset. However as β_{dj} can now take negative values, the constraint placed on ζ_{dj} such that $\zeta_{dj} \geq 0$ is not applicable and is not included.

9.3.1 Methods

5 datasets of 50 independent SNPs and 100 subjects were simulated. SNP 10, 20, 30, 40 and 50 were simulated as causal SNPs with MAFs of 0.02, 0.1, 0.2, 0.25 and 0.4 respectively. The MAFs for remaining SNPs were randomly generated from a uniform distribution with ranging between 0.01 and 0.5. The MAF for each SNP was the same across all 5 datasets. SNPs that contained the same combination of alleles across all individuals in a dataset were re-simulated until this was not the case. Each dataset was standardised separately.

1,000 simulations were run for each analysis with the seed varying between 1 and 101, with the exclusion of seed 56 and 10 repetitions for each seed. One dataset failed to converge for seed 56 and therefore results for that seed was not included. Closer inspection running the simulation for this dataset showed that the algorithm was converging however it was taking a greater number of iterations than the default setting of 10,000 iterations. In contrast the other datasets took < 10 iterations to converge. Li *et al*. proposed a simplification of the dual penalty by fixing the penalty on

γ_j to 1 (9.7). This penalty however is too strong for the simulated dataset and returns a null model. The penalty was therefore fixed to $0.004\left(\frac{1}{\sum_d N_d}\right)$.

10-fold CV, BIC and the permutation method were all used for tuning parameter selection. The permutation method used 100 repetitions with the optimal tuning parameter selected as the median from the 100 repeated estimates. The CV and BIC methods used a range of 100 lambda values. These were determined by calculating the smallest lambda value required for a null model as the largest lambda value, and the remaining 98 values were equidistant values between this value and 0. For the meta-LASSO the sub-splitting required for CV and the permutations on the phenotype for the permutation method were both performed within respective datasets. The BIC for the meta-LASSO was calculated as shown in (9.9).

For the stacked LASSO all 5 datasets are pooled together into on larger dataset and then fitted with each tuning parameter selection method across all datasets. For the separate LASSO, tuning parameter selection was applied to the 5 datasets separately, allowing a different optimum λ in each dataset.

Heterogeneity between datasets was simulated by varying the percentage variance explained of each causal SNP (3.1) between datasets (Table 9.2). Each causal SNP was simulated with a positive β. Firstly a scenario to compare how both Cross-validation and the permutation would perform with the meta-LASSO was simulated in a high powered setting, similar to the Li study (246). In this scenario each causal SNP explained 5% of the variation with no heterogeneity between datasets.

For the remaining scenarios, a baseline scenario was used that simulates 1% variance explained and across all datasets. In the remaining scenarios heterogeneity between datasets is increased. Each dataset varied in the percentage of variance explained but still averages 1% across all 5 datasets.

Table 9.2 Simulation of heterogeneity in datasets and the percentage of variance explained in each dataset

Heterogeneity	Dataset 1	Dataset 2	Dataset 3	Dataset 4	Dataset 5
High variance explained	5	5	5	5	5
Baseline	1	1	1	1	1
Low	0.7	1.3	1	0.7	1.3
Mid	0.5	1.5	1	0.5	1.5
High	0	2	1	0	2

Variable selection performance is evaluated by two measures; the first is sensitivity and specificity which is based on the proportion of truly positive and negative SNPs as described by Li *et al.* (246). In this case sensitivity is defined as the proportion of truly causal SNPs that are selected with a non-zero β_{dj}. For the remainder of this chapter this measure will be known as the single selection measure. Specificity is defined as the proportion of truly non-causal SNPs estimated as $\beta_{dj} = 0$. The second measure considered is based on the meta-LASSO and separate LASSO's ability to replicate selection of any SNPs across the 5 datasets. In this case a replication defined as a SNP that is selected in more than one dataset. Sensitivity is defined as the proportion of truly causal SNPs that are replicated. Specificity is defined as the proportion of non-causal SNPs estimated as $\beta_{dj} = 0$ in at least 4 of the 5 datasets (i.e. does not replicate). LASSO models for the stacked and separate method were fitted using glmnet. The meta-LASSO function was written in R using the algorithm described in Table 9.1 and using the coordinate descent algorithm described and written (see section 3.2.1).

9.3.2 Results in the high variance explained scenario

Sensitivity and specificity rates in the high power scenario are shown in Table 9.3. The results for the BIC are similar to those shown by Li *et al.* in low heterogeneity (246). Both the meta-LASSO and stacked LASSO performed well with high sensitivity (0.972 and 1.00) and specificity rates (0.995 and 0.982). In both this and the Li *et al.* simulation the stacked LASSO produced a higher sensitivity rate than meta-LASSO and in both simulations the separate LASSO produced a high specificity but the sensitivity were lower than the competing methods.

For Cross-validation, the meta-LASSO performed well producing a high sensitivity rate (0.986) compared to the competing tuning parameter selection methods, but also the lowest specificity rate (0.992). Both the stacked and separate LASSO methods produce lower specificity rates (0.764 and 0.803) for CV than the BIC (0.982 and 0.948) and permutation method (0.995 and 0.991). This suggests that the stacked and separate LASSO based methods will select a greater number of false positives than the meta-LASSO method. The stacked LASSO selected on average 10.72 FPs in each model where the separate LASSO selected on average 8.87 FPs using CV. The meta-LASSO however does not select as many false positives (mean number of false positives selected = 0.36) suggesting that CV is a more conservative method for variable selection when using the meta-LASSO. While the stacked LASSO selected every true SNP using CV, the separate LASSO produced a lower sensitivity rate.

Across the 1,000 simulations the stacked LASSO selected the most causal SNPs and produced the highest sensitivity with every tuning parameter selection method. The meta-LASSO method performed well using the permutation method in terms of reducing the number of false positive SNPs selected and produced the highest specificity rate (0.998) across all methods and tuning parameter selection methods. In fact all three methods produced the highest specificity using the permutation method compared BIC or CV suggesting that this method should be the preferred choice in any

study where the number of false positives is to be reduced as much as possible. While the sensitivity rate for the permutation method was similar with the other tuning parameter selection methods using the meta-LASSO and stacked LASSO, this was not the case for the separate LASSO which produced the lowest sensitivity rate (0.486) amongst the three tuning parameter selection methods.

The sensitivity rates for each of the 5 simulated causal SNPs were similar for each method and tuning parameter selection method (Table 9.4), although the sensitivity rate for the rarest causal SNP (MAF = 2%) was slightly lower than the other causal SNPs.

Table 9.3 Mean and standard deviation of sensitivity and specificity results using single selection measure in a high variance explained scenario using the meta-LASSO, stacked LASSO and separate LASSO with Cross-validation, BIC and permutation method as tuning parameter selection methods over 1,000 simulations.

Method	Cross-validation		BIC		Permutation method	
	Sens	**Spec**	**Sens**	**Spec**	**Sens**	**Spec**
Meta-LASSO	0.986 ± 0.028	0.992 ± 0.016	0.972 ± 0.058	0.995 ± 0.008	0.985 ± 0.029	0.998 ± 0.005
Stacked LASSO	1.000 ± 0.000	0.764 ± 0.122	1.000 ± 0.006	0.982 ± 0.021	0.999 ± 0.015	0.995 ± 0.011
Separate LASSO	0.857 ± 0.096	0.803 ± 0.064	0.702 ± 0.129	0.948 ± 0.023	0.486 ± 0.092	0.991 ± 0.006

Table 9.4 Sensitivity rates of the 5 causal SNPs in the high variance explained scenario using the meta-LASSO, stacked LASSO and separate LASSO and Cross-validation, BIC and permutation method as tuning parameter selection methods over 1,000 simulations.

Method	Minor allele frequency	Meta - LASSO	Stacked LASSO	Separate LASSO
Cross-validation	MAF = 2%	0.972	1.000	0.833
	MAF = 10%	0.989	1.000	0.860
	MAF = 20%	0.989	1.000	0.859
	MAF = 25%	0.989	1.000	0.866
	MAF = 40%	0.993	1.000	0.868
BIC	MAF = 2%	0.948	0.999	0.682
	MAF = 10%	0.975	1.000	0.706
	MAF = 20%	0.978	1.000	0.703
	MAF = 25%	0.977	1.000	0.710
	MAF = 40%	0.981	1.000	0.709
Permutation method	MAF = 2%	0.970	0.996	0.472
	MAF = 10%	0.988	1.000	0.490
	MAF = 20%	0.988	0.999	0.490
	MAF = 25%	0.986	0.999	0.489
	MAF = 40%	0.992	1.000	0.489

The sensitivity and specificity results based on replication across datasets for the meta-LASSO and separate LASSO is shown in Table 9.5. For the meta-LASSO, SNP selection using this measure increased the sensitivity for all tuning parameter selection methods with only a small decrease (between 0.003 and 0.004) in specificity for each tuning parameter selection method. There was a significantly large increase in the sensitivity rates for the separate LASSO using the repetition measure compared to the single selection measure. The specificity again slightly decreases for CV but increases for the BIC and permutation method.

The results for both LASSO based methods suggest that the replication measure may be a better measure to use for SNP selection compared to the single selection measure as this measure selects a larger proportion of true positives and at worst, a small increase in the number of false positives.

Table 9.5 Mean and standard deviation of sensitivity and specificity results for the proportion of replicated results in a high variance explained scenario using the meta-LASSO, stacked LASSO and separate LASSO with Cross-validation, BIC and permutation method as tuning parameter selection methods over 1,000 simulations.

Method	Cross-validation		BIC		Permutation method	
	Sens	Spec	Sens	Spec	Sens	Spec
Meta-LASSO	0.998 ± 0.018	0.988 ± 0.024	0.989 ± 0.047	0.992 ± 0.014	0.999 ± 0.083	0.989 ± 0.007
Separate LASSO	0.997 ± 0.023	0.744 ± 0.136	0.973 ± 0.082	0.977 ± 0.028	0.793 ± 0.177	0.999 ± 0.004

9.3.3 Results on varying levels of heterogeneity

9.3.3.1 Results based on the single selection measure

Table 9.6 shows the results for Cross-validation for the varying levels of heterogeneity. The three methods produce very different results. As heterogeneity increased the sensitivity for the meta-LASSO and stacked LASSO decreased. The greatest decrease occurred between the mid and high levels of heterogeneity. The specificity rate remained mostly consistent between heterogeneity levels for all three methods using CV. The sensitivity rate for the separate LASSO increases slightly as the heterogeneity increases.

The stacked LASSO again performs well in terms of sensitivity compared to the competing methods however, the specificity remains low (> 0.83 across all levels of heterogeneity) with on average 7.56 false positive selected at baseline (specificity = 0.832) and decreasing to 4.82 in the high heterogeneity scenario (specificity = 0.893). In contrast both the meta-LASSO and separate LASSO performed poorly in terms of sensitivity where between a quarter and a fifth all truly causal SNPs were selected (0.174 - 0.264 for meta-LASSO and between 0.177 – 0.181 for separate LASSO). While

the separate LASSO still selects a number of false positives, the specificity rate is higher than the stacked LASSO and both the sensitivity and specificity for this method remain stable regardless of heterogeneity. The use of CV for the meta-LASSO works well in terms of reducing the number of false positives (sensitivity between 0.992 and 0.989) and in fact seems a relatively conservative method.

Table 9.6 Mean and standard deviation of sensitivity and specificity results using the single selection measure for varying levels of heterogeneity using the meta-LASSO, stacked LASSO and seperate LASSO with Cross-vaildation over 1,000 simulations.

Heterogeneity	Meta - LASSO		Stacked LASSO		Separate LASSO	
	Sens	Spec	Sens	Spec	Sens	Spec
Baseline	0.264 ± 0.197	0.992 ± 0.016	0.741 ± 0.291	0.832 ± 0.137	0.177 ± 0.104	0.929 ± 0.048
Low	0.257 ± 0.191	0.992 ± 0.016	0.732 ± 0.294	0.834 ± 0.138	0.178 ± 0.103	0.929 ± 0.048
Mid	0.250 ± 0.188	0.991 ± 0.016	0.705 ± 0.306	0.840 ± 0.138	0.179 ± 0.103	0.929 ± 0.049
High	0.174 ± 0.158	0.989 ± 0.018	0.432 ± 0.329	0.893 ± 0.126	0.181 ± 0.104	0.927 ± 0.049

The results for BIC (Table 9.7) show similar patterns to those shown for the CV. As heterogeneity increased the sensitivity for the meta-LASSO and stacked LASSO decreases. The sensitivity for the separate LASSO increases as heterogeneity increases. The specificity of the meta-LASSO and separate LASSO decreases while the specificity of the stacked LASSO increases. These patterns for the BIC replicate the patterns shown by Li *et al*. in their simulation (see Table 2 (246)).

BIC has been shown to be a conservative method for variable selection (see section 3.3.2.3). It is unsurprising therefore that all three LASSO methods produce high specificity rates using BIC as only a small number of variables are selected. Much like the simulation performed by Li *et al*. the meta-LASSO outperforms the competing methods using the BIC for tuning parameter selection. Both the meta-LASSO and stacked LASSO returned high specificity rates with little difference between them but

the meta-LASSO produced a slightly higher sensitivity rate. The separate LASSO returns the lowest sensitivity and specificity of the three methods using BIC. For the permutation method the stacked LASSO outperforms the competing methods (Table 9.8). All three methods produce a high specificity rate however; the stacked LASSO has a higher sensitivity rate than both the meta-LASSO and separate LASSO. The poor performance of the meta-LASSO compared to the stacked LASSO could be attributed to the strength of the penalty chosen. As previously discussed, each dataset will have a different minimum λ estimate to produce a null model. However by permuting with the same λ across all datasets, variable selection may be restricted in some datasets.

In 3.3.2, the permutation method performed well against the competing tuning parameter selection methods in terms of variable selection. All 3 methods produced similar results in this simulation for the three tuning parameter selection methods. Each method allowed a number of false positives using CV while the BIC does not select many true positives. The permutation method has similar specificity rates as BIC but a higher sensitivity rate. The separate LASSO produces the lowest sensitivity rate of all three methods regardless of which tuning parameter method is used. This is unsurprising as models are fitted individually on datasets rather than together therefore the analysis will lack power to select causal SNPs. For each tuning parameter selection method, the sensitivity increased slightly in the high heterogeneity scenario, while they decreased for the meta-LASSO and stacked LASSO. While the separate LASSO may lack the power to select truly causal SNPs, the increase in variance explained in 2 of the 5 datasets will increase the power within these datasets. In general meta-LASSO seems to be a conservative method compared to the stacked LASSO regardless of which tuning parameter is used as shown by the high specificity and low sensitivity for all three tuning parameter selection methods (Table 9.6, Table 9.7 and Table 9.8).

Given the lack of true positives selected, the meta-LASSO would work best with the permutation method for tuning parameter selection. The method already controls the FPR well and the use of permutation method will allow a higher TPR than BIC or CV. The stacked LASSO using the permutation method however showed superior

performance across all methods as the sensitivity rate was higher and specificity was only slightly lower than the meta-LASSO.

Table 9.7 Mean and standard deviation of sensitivity and specificity using the single selection measure for varying levels of heterogeneity using the meta-LASSO, stacked LASSO and seperate LASSO with BIC over 1,000 simulations.

Heterogeneity	Meta - LASSO		Stacked LASSO		Separate LASSO	
	Sens	Spec	Sens	Spec	Sens	Spec
Baseline	0.095 ± 0.105	0.999 ± 0.005	0.076 ± 0.117	0.999 ± 0.004	0.049 ± 0.047	0.992 ± 0.006
Low	0.098 ± 0.107	0.998 ± 0.005	0.081 ± 0.121	0.999 ± 0.004	0.049 ± 0.045	0.992 ± 0.005
Mid	0.096 ± 0.105	0.998 ± 0.005	0.071 ± 0.113	0.999 ± 0.004	0.050 ± 0.048	0.993 ± 0.006
High	0.086 ± 0.100	0.997 ± 0.007	0.049 ± 0.088	0.998 ± 0.006	0.055 ± 0.054	0.992 ± 0.006

Table 9.8 Mean and standard deviation of sensitivity and specificity results using the single selection measure for varying levels of heterogeneity using the meta-LASSO, stacked LASSO and seperate LASSO with the permutation method over 1,000 simulations.

Heterogeneity	Meta - LASSO		Stacked LASSO		Separate LASSO	
	Sens	Spec	Sens	Spec	Sens	Spec
Baseline	0.307 ± 0.173	0.992 ± 0.011	0.423 ± 0.217	0.987 ± 0.017	0.073 ± 0.051	0.987 ± 0.007
Low	0.299 ± 0.171	0.992 ± 0.011	0.416 ± 0.217	0.987 ± 0.017	0.073 ± 0.050	0.987 ± 0.007
Mid	0.286 ± 0.167	0.992 ± 0.011	0.397 ± 0.216	0.987 ± 0.017	0.073 ± 0.051	0.987 ± 0.017
High	0.171 ± 0.141	0.992 ± 0.011	0.224 ± 0.183	0.986 ± 0.018	0.074 ± 0.050	0.987 ± 0.007

9.3.3.2 Results based on the proportion of the replication measure

Table 9.9, Table 9.10 and Table 9.11 show the results for Cross-validation, BIC and the permutation methods respectively for both the meta-LASSO and separate LASSO using the replication measure. In each case the meta-LASSO performed well using this measure compared to the single selection measure. There is little difference in specificity rates between the two measures however there is an increase in sensitivity for each tuning parameter selection method. These results suggest that when a true positive is selected, there tends to be a replication in another dataset but this is not the case when a false positive is selected. Given the implications of these results the meta-LASSO with a replication measure should be used variable selection over the single selection measure proposed by Li *et al.* Both the sensitivity and specificity rates increased using the separate LASSO with CV. The sensitivity for permutation method and BIC both increased for the replication measure compared to the single selection measure but the specificity rate decreased slightly. This shows that the separate LASSO rarely selects SNPs using BIC and permutation method and when a SNP is selected it is rarely replicated, again this can be attributed to the lack of power.

Table 9.9 Mean and standard deviation of sensitivity and specificity results for both selection measures across varying levels of heterogeneity using the meta-LASSO and seperate LASSO with Cross-validation over 1,000 simulations.

Heterogeneity	Meta - LASSO				Separate LASSO			
	Single selection		Replication		Single selection		Replication	
	Sensitivity	Specificity	Sensitivity	Specificity	Sensitivity	Specificity	Sensitivity	Specificity
Baseline	0.264 ± 0.197	0.992 ± 0.016	0.331 ± 0.240	0.988 ± 0.023	0.177 ± 0.104	0.929 ± 0.048	0.212 ± 0.234	0.958 ± 0.065
Low	0.257 ± 0.191	0.992 ± 0.016	0.325 ± 0.237	0.988 ± 0.023	0.178 ± 0.103	0.929 ± 0.048	0.215 ± 0.235	0.957 ± 0.064
Mid	0.250 ± 0.188	0.991 ± 0.016	0.320 ± 0.237	0.987 ± 0.024	0.179 ± 0.103	0.929 ± 0.049	0.215 ± 0.234	0.957 ± 0.064
High	0.174 ± 0.158	0.989 ± 0.018	0.231 ± 0.201	0.983 ± 0.026	0.181 ± 0.104	0.927 ± 0.049	0.219 ± 0.243	0.955 ± 0.066

Table 9.10 Mean and standard deviation of sensitivity and specificity results for both selection measures across varying levels of heterogeneity using the meta-LASSO and seperate LASSO and BIC over 1,000 simulations.

Heterogeneity	Meta - LASSO				Separate LASSO			
	Single selection		Replication		Single selection		Replication	
	Sensitivity	Specificity	Sensitivity	Specificity	Sensitivity	Specificity	Sensitivity	Specificity
Baseline	0.095 ± 0.105	0.999 ± 0.005	0.129 ± 0.144	0.998 ± 0.008	0.049 ± 0.047	0.992 ± 0.006	0.020 ± 0.064	1.000 ± 0.003
Low	0.098 ± 0.107	0.998 ± 0.005	0.136 ± 0.151	0.997 ± 0.008	0.049 ± 0.045	0.992 ± 0.005	0.020 ± 0.063	0.999 ± 0.004
Mid	0.096 ± 0.105	0.998 ± 0.005	0.132 ± 0.146	0.997 ± 0.008	0.050 ± 0.048	0.993 ± 0.006	0.019 ± 0.062	0.999 ± 0.004
High	0.086 ± 0.100	0.997 ± 0.007	0.125 ± 0.146	0.995 ± 0.011	0.055 ± 0.054	0.992 ± 0.006	0.027 ± 0.077	0.999 ± 0.004

Table 9.11 Mean and standard deviation of sensitivity and specificity results for both selection measures across varying levels of heterogeneity using the meta-LASSO and seperate LASSO and permutation method over 1,000 simulations.

Heterogeneity	Meta - LASSO				Separate LASSO			
	Single selection		Replication		Single selection		Replication	
	Sensitivity	Specificity	Sensitivity	Specificity	Sensitivity	Specificity	Sensitivity	Specificity
Baseline	0.307 ± 0.173	0.992 ± 0.011	0.388 ± 0.212	0.989 ± 0.015	0.073 ± 0.051	0.987 ± 0.007	0.046 ± 0.092	0.998 ± 0.006
Low	0.299 ± 0.171	0.992 ± 0.011	0.381 ± 0.212	0.989 ± 0.015	0.073 ± 0.050	0.987 ± 0.007	0.045 ± 0.091	0.998 ± 0.006
Mid	0.286 ± 0.167	0.992 ± 0.011	0.365 ± 0.208	0.989 ± 0.015	0.073 ± 0.051	0.987 ± 0.017	0.044 ± 0.090	0.998 ± 0.006
High	0.171 ± 0.141	0.992 ± 0.011	0.229 ± 0.183	0.988 ± 0.015	0.074 ± 0.050	0.987 ± 0.007	0.043 ± 0.090	0.998 ± 0.006

9.3.4 Sensitivity analysis

Results of the simulation showed that the meta-LASSO seems to be a conservative method which produced a low sensitivity and high specificity regardless of the tuning parameter selection method that is used. Perhaps surprisingly, this is even the case when using Cross-validation which tends to over select variables. The reason for this must be attributed to the penalty on ζ_{dj} as removing this penalty from equation (9.6) reduces the equation to the stacked LASSO. Two reasons were considered as to why the meta-LASSO did not select many variables. The first is the nature of the penalty on ζ_{dj}, and the second is the strength of the fixed penalty on γ_j. To check that this was not due to the strength of the fixed penalty, a sensitivity analysis was run by varying the fixed λ. The results are shown in Table 9.12, Table 9.13 and Table 9.14 for the three respective tuning parameter selection methods. The results show that there is little difference in sensitivity and specificity rates and that the conclusion to this simulation study would remain the same if a different fixed λ value was selected. The specificity rate remained similar across all levels heterogeneity, tuning parameter selection methods and fixed lambdas (between 0.987 and 0.999). The sensitivity rate for CV and permutation method was highest for λ = 0.002 suggesting a smaller penalty would work best, however there is only a 2-3% difference in sensitivity compared to the λ = 0.004. A λ = 0.006 seemed to perform best for the BIC however this increase in sensitivity compared to λ = 0.004 was also small (<1% at most levels of heterogeneity). A grid search method could have been used in this simulation rather than fixing one penalty in order to find the optimum penalty. The meta-LASSO using the permutation method performed well compared to the other tuning parameter selection methods. This method produced a higher sensitivity rate while maintaining a similar specificity rate compared to the BIC. Results suggest that λ = 0.002 produced the best results as the specificity rate remained high but the sensitivity rate increased. These results however still do not outperform those using the stacked LASSO regardless of the heterogeneity level.

Given the strength of the fixed penalty has little effect on the sensitivity and specificity rates it seems the nature of the penalty on ζ_{dj} leads to the meta-LASSO producing conservative models. This is because the shrinkage of the heterogeneity estimates is towards 0 as does the fixed penalty on γ_j. While the meta-LASSO is able to perform well when there are large effect estimates (Table 9.3) and when the effects sizes for a causal SNP across studies are small, the heterogeneity between SNPs will also be small. If the heterogeneity between SNPs is small then little shrinkage is required to remove the SNP from the model and hence lower the sensitivity rate.

Table 9.12 Mean and standard deviation of sensitivity and specificity rates for the meta-LASSO using Cross-validation across varying levels of heterogeneity and fixed λ estimates over 1,000 simulations.

Heterogeneity	Measure	Stacked LASSO	λ = 0.002	λ = 0.004	λ = 0.006	λ = 0.010	λ = 0.020
Baseline	Sensitivity	0.741 ± 0.291	0.292 ± 0.213	0.264 ± 0.197	0.242 ± 0.189	0.224 ± 0.190	0.224 ± 0.197
	Specificity	0.832 ± 0.137	0.990 ± 0.019	0.992 ± 0.016	0.992 ± 0.015	0.992 ± 0.014	0.994 ± 0.012
Low	Sensitivity	0.732 ± 0.294	0.288 ± 0.211	0.257 ± 0.191	0.237 ± 0.185	0.218 ± 0.186	0.217 ± 0.189
	Specificity	0.834 ± 0.138	0.991 ± 0.018	0.992 ± 0.016	0.992 ± 0.015	0.992 ± 0.014	0.994 ± 0.012
Mid	Sensitivity	0.705 ± 0.306	0.275 ± 0.204	0.250 ± 0.188	0.230 ± 0.181	0.214 ± 0.182	0.212 ± 0.190
	Specificity	0.840 ± 0.138	0.990 ± 0.018	0.991 ± 0.016	0.992 ± 0.015	0.992 ± 0.014	0.993 ± 0.013
High	Sensitivity	0.432 ± 0.329	0.177 ± 0.169	0.174 ± 0.158	0.166 ± 0.155	0.156 ± 0.152	0.146 ± 0.147
	Specificity	0.893 ± 0.126	0.987 ± 0.021	0.989 ± 0.018	0.990 ± 0.016	0.991 ± 0.016	0.993 ± 0.012

Table 9.13 Mean and standard deviation of sensitivity and specificity rates for the meta-LASSO using BIC across different levels of heterogeneity and fixed λ estimates over 1,000 simulations.

Heterogeneity	Measure	Stacked LASSO	λ = 0.002	λ = 0.004	λ = 0.006	λ = 0.010	λ = 0.020
Baseline	Sensitivity	0.076 ± 0.117	0.099 ± 0.109	0.095 ± 0.105	0.106 ± 0.105	0.097 ± 0.097	0.092 ± 0.092
	Specificity	0.999 ± 0.004	0.999 ± 0.005	0.999 ± 0.005	0.998 ± 0.006	0.998 ± 0.005	0.998 ± 0.005
Low	Sensitivity	0.081 ± 0.121	0.092 ± 0.106	0.098 ± 0.107	0.101 ± 0.107	0.098 ± 0.096	0.217 ± 0.189
	Specificity	0.999 ± 0.004	0.999 ± 0.004	0.998 ± 0.005	0.998 ± 0.005	0.998 ± 0.005	0.994 ± 0.012
Mid	Sensitivity	0.071 ± 0.113	0.096 ± 0.104	0.096 ± 0.105	0.102 ± 0.107	0.096 ± 0.095	0.091 ± 0.093
	Specificity	0.999 ± 0.004	0.999 ± 0.005	0.998 ± 0.005	0.998 ± 0.005	0.998 ± 0.005	0.998 ± 0.005
High	Sensitivity	0.049 ± 0.088	0.079 ± 0.096	0.086 ± 0.100	0.093 ± 0.100	0.086 ± 0.089	0.075 ± 0.083
	Specificity	0.998 ± 0.006	0.998 ± 0.006	0.997 ± 0.007	0.996 ± 0.007	0.996 ± 0.006	0.997 ± 0.006

Table 9.14 Mean and standard deviation of sensitivity and specificity rates for the meta-LASSO using permutation method across different levels of heterogeneity and fixed λ estimates over 1,000 simulations.

Heterogeneity	Measure	Stacked LASSO	λ = 0.002	λ = 0.004	λ = 0.006	λ = 0.010	λ = 0.020
Baseline	Sensitivity	0.423 ± 0.217	0.357 ± 0.192	0.307 ± 0.173	0.273 ± 0.161	0.243 ± 0.154	0.273 ± 0.173
	Specificity	0.987 ± 0.017	0.990 ± 0.013	0.992 ± 0.011	0.993 ± 0.010	0.994 ± 0.009	0.993 ± 0.011
Low	Sensitivity	0.416 ± 0.217	0.349 ± 0.189	0.299 ± 0.171	0.268 ± 0.159	0.238 ± 0.152	0.267 ± 0.169
	Specificity	0.987 ± 0.017	0.990 ± 0.013	0.992 ± 0.011	0.993 ± 0.010	0.994 ± 0.009	0.993 ± 0.011
Mid	Sensitivity	0.397 ± 0.216	0.332 ± 0.186	0.286 ± 0.167	0.254 ± 0.156	0.228 ± 0.150	0.260 ± 0.168
	Specificity	0.987 ± 0.017	0.990 ± 0.013	0.992 ± 0.011	0.993 ± 0.010	0.994 ± 0.009	0.993 ± 0.011
High	Sensitivity	0.224 ± 0.183	0.185 ± 0.156	0.171 ± 0.141	0.161 ± 0.132	0.156 ± 0.128	0.172 ± 0.142
	Specificity	0.986 ± 0.018	0.990 ± 0.013	0.992 ± 0.011	0.990 ± 0.001	0.994 ± 0.009	0.992 ± 0.011

9.4 Discussion

In the previous section, a simulation study was conducted to assess the relative performance of the meta-LASSO when the simulated effect sizes are not overpowered. The Li study simulated effect sizes of $\beta_{dj} \sim \mathrm{N}(3, 0.5)^2$, in my high-powered simulation the β_{dj} effect estimates varied between 1.13 and 0.32 depending on the MAF of the causal SNP. Even with a smaller effect size the high powered scenario produced similar results as the Li *et al*. simulation. This simulation showed the same patterns for the three LASSO based methods as the Li *et al*. simulation, with the sensitivity and specificity rates decreasing and heterogeneity increased for both meta-LASSO and stacked LASSO. The separate LASSO saw an increase in sensitivity and a decrease in specificity as heterogeneity increased. These patterns were apparent in the lower powered simulations and across all three tuning parameter selection methods. To test

whether the selection of a fixed λ had any effect on the simulation results, a sensitivity analysis was conducted to test varying values of the fixed penalty. The results showed that the fixed penalty in fact had little effect on the overall result and that the stacked LASSO using the permutation method was the best performing method overall.

Two defined measures were used in this simulation; the first was the single selection measure which calculated the total proportion of true and false SNPs that were selected, and the second was the replication measure which calculated the proportion of true and false SNPs replicated the correct result. The simulation showed that replication measure produced a higher sensitivity while maintaining a similar specificity as the single selection measure. The replication measure can be used to protect against selecting variables that are only selected in one dataset and may in fact be a false positive. This simulation showed that when a true positive is selected, there tends to be a replication in at least one dataset but this is not the case when a false positive is selected.

9.5 Conclusion

In this chapter, I have reviewed a number integrative analysis based methods that incorporate penalised regression. I followed up by conducting a simulation study comparing the meta-LASSO method (246) against both the stacked LASSO and separate LASSO in a SNP study setting. The results showed that the meta-LASSO performed well for both CV and BIC compared to the stacked and separate methods but did not perform well using the permutation method. For each tuning parameter selection method the meta-LASSO produced a low sensitivity and high specificity rates suggesting that the method is quite conservative for SNP selection. Of the three tuning parameter selection methods the permutation method performs the best but is outperformed by the stacked LASSO using the permutation method.

Results of the simulation showed that the meta-LASSO seems to be a conservative method which produced a low sensitivity and high specificity regardless of the tuning parameter selection method that is used. There were two potential reasons for this, first is the nature of the penalty on ζ_{dj}, the second is the strength of the fixed penalty on γ_j. A sensitivity analysis was performed to see the influence of the fixed penalty; the results showed that there was little difference in varying this penalty (Table 9.12, Table 9.13 and Table 9.14). Therefore the conservative nature of this method seems to be due to the ζ_{dj} penalty as it shrinks to 0 further removing variables from the model.

I also suggest an alternative measure for variable selection for both the meta-LASSO and separate LASSO which selects variables only if the SNP is selected in more than one dataset (i.e. replicated). Li *et al.* suggest variable selection to be based on if any SNP from any one dataset is selected (246). Results showed that the replication measure produces a similar specificity rates but a higher sensitivity rates which suggests a more powerful measure for variable selection.

To current knowledge this is the first study that applies the meta-LASSO in a SNP study and also tests if the method works in a setting where the causal variables not overpowered. This is also the first study that tests other tuning parameter selection methods such as Cross-validation and the permutation method for the meta-LASSO.

10 The Integrative LASSO

10.1 Introduction

Results from the simulation study in Chapter 9 comparing a number of methods for integrative analysis produced low sensitivity rate in the lower powered scenario where the level of heterogeneity was varied. The stacked LASSO method was the exception to this rule; however the method is unable to allow for heterogeneity between datasets. In this chapter, I propose an alternative method, the Integrative LASSO (IL). The purpose of the Integrative LASSO is to penalise SNPs within datasets but also penalise some SNPs into the model by averaging β estimates across datasets and therefore potentially increasing the sensitivity rate.

In this chapter, I firstly describe the Integrative LASSO and explain the reasoning behind the penalties that are used. I provide an algorithm to apply the IL method to datasets by coordinate descent. I then provide an example of how the Integrative LASSO works using a test dataset and finally conduct a simulation study comparing the IL to meta-LASSO, stacked LASSO and separate LASSO which were discussed in greater detail in Chapter 9.

10.2 The Integrative LASSO

Consider the same scenario described in the previous chapter where there are *D* datasets $H_1, \dots, H_D$ for an integrative analysis. Each dataset of the *D* datasets consists of N_d subjects and the same *P* SNPs. The following notation is used, let $i = \{1, \dots N\}$ denote the i^{th} subject, $j = \{1, \dots P\}$ denote the j^{th} SNP and $d = \{1, \dots D\}$ represents the d^{th} dataset. For simplicity it is assumed that each dataset consists of the same

number of subjects contains the same SNPs in each dataset. Each dataset is standardised separately. The Integrative LASSO (IL) minimises the following function:

$$\hat{\beta}(\lambda) = \arg\min_{\beta} \sum_{d=1}^{D} \frac{1}{2N_d} \sum_{i=1}^{N_d} \left(y_{di} - \sum_{j=1}^{P} x_{dij}\beta_{dj} \right)^2 + \lambda_1 \sum_{d=1}^{D} \sum_{j=1}^{P} |\beta_{dj}| + \lambda_2 \sum_{d=1}^{D} \sum_{j=1}^{P} \left(\beta_{dj} - \frac{1}{D} \sum_{d=1}^{D} \beta_{dj} \right)^2 \quad (10.1)$$

The IL incorporates two penalty terms with elements of both the fused LASSO (32) and elastic net (18) in the second penalty. The first penalty penalises SNPs within each dataset which will produce separate β_j for each of the D datasets, similar to the penalty on ζ for the meta-LASSO method (246). The second penalty attempts to average β_{dj} estimates across datasets by penalising the squared difference between each β_{dj} estimates towards the mean estimate of these estimates across all datasets. For this chapter, the λ_1 penalty is called the LASSO penalty and the λ_2 penalty will be referred to as the variance penalty. The variance penalty penalises β_{dj} estimates towards the mean across all D datasets. The logic behind this penalty is that a casual SNP is likely to have a non-zero beta estimates with the same sign in most, if not all, datasets leading to a non-zero mean. In contrast mean beta across all D datasets for a non-causal SNP is likely to be zero. Therefore by forcing the SNPs towards the mean across all datasets the IL attempts to retain the causal SNPs by penalising these SNPs away from zero while non-causal SNPs are penalised further towards zero. An increase in λ_2 penalises SNPs towards the mean β_j across all *D* datasets therefore for any large penalty on λ_2 combined with a small penalty on λ_1, the estimates are forced towards a OLS regression model, where the estimates across all datasets will be the same and none or a very small number of the SNPs are removed from the model. The square difference was used rather than the absolute difference for two reasons; the first was that removing SNPs from the datasets has greater importance (i.e variable selection is performed). The second reason is that the squared term in the elastic net encourages a

grouping effect that the absolute penalty does not (18), therefore this would aid in grouping SNP estimates across datasets.

10.3 Fitting the Integrative LASSO via coordinate descent

The Integrative LASSO can be fitted using the coordinate decent algorithm (CDA). This section presents the algebra and my own algorithm for the IL. Both the algebra and algorithm presented are similar for to those I show for the LASSO in sections 2.4.2 and 3.2.1.

10.3.1 The algebra for fitting the Integrative LASSO via coordinate descent

The Integrative LASSO minimises the function shown in (10.1). Removing the λ_2 penalty from this function, a function that fits the separate LASSO on each dataset is obtained which can be performed by coordinate descent and the solution to minimising the LASSO is shown in section 2.5.2.1. We can therefore concentrate on minimising the variance penalty. We wish to minimise this function of λ_2 ($f(\lambda_2)$) for some β_{lk} where $l = \{1, \dots, D\}$, $k = \{1, \dots, P\}$ and therefore differentiate w.r.t β_{lk} to produce a solution.

We firstly expand summation of the penalty over the *D* datasets (10.2).

$$f(\lambda_2) = \sum_{d=1}^{D}\sum_{k=1}^{P}\left(\beta_{dk} - \frac{1}{D}\sum_{d=1}^{D}\beta_{dk}\right)^2$$
$$= \sum_{k=1}^{P}\left(\beta_{1k} - \frac{1}{D}\sum_{d=1}^{D}\beta_{dk}\right)^2 + \cdots + \left(\beta_{Dk} - \frac{1}{D}\sum_{d=1}^{D}\beta_{dk}\right)^2 \quad (10.2)$$

For any $l = \{1, \dots, D\}$ equation (10.2) can be generalised as:

$$f(\lambda_2) = \sum_{k=1}^{P}\left(\left(\beta_{lk} - \frac{1}{D}\sum_{d=1}^{D}\beta_{dk}\right)^2 + \sum_{m=1,m\neq l,}^{D}\left(\beta_{mk} - \frac{1}{D}\sum_{d=1}^{D}\beta_{dk}\right)^2\right) \quad (10.3)$$

We now differentiate w.r.t. β_{lk} and simplify:

$$\frac{\delta f(\lambda_2)}{\delta \beta_{lk}} = 2\frac{D-1}{D}\left(\beta_{lk} - \frac{1}{D}\sum_{d=1}^{D}\beta_{dk}\right) - 2\frac{1}{D}\sum_{m=1,m\neq l,}^{D}\left(\beta_{mk} - \frac{1}{D}\sum_{d=1}^{D}\beta_{dk}\right)$$

$$= \frac{2}{D}\left[(D-1)\left(\beta_{lk} - \frac{1}{D}\sum_{d=1}^{D}\beta_{dk}\right) - \left(\sum_{m=1,m\neq l}^{D}\beta_{mk}\right) + \frac{D-1}{D}\sum_{d=1}^{D}\beta_{dk}\right]$$

$$= \frac{2}{D}\left[(D-1)\beta_{lk} - \left(\sum_{m=1,m\neq l}^{D}\beta_{mk}\right) + \left(\frac{D-1}{D} - \frac{D-1}{D}\right)\sum_{d=1}^{D}\beta_{dk}\right]$$

$$\therefore \frac{\delta f(\lambda_2)}{\delta \beta_{lk}} = \frac{2}{D}\left[(D-1)\beta_{lk} - \left(\sum_{m=1,m\neq l}^{D} \beta_{mk}\right)\right] \tag{10.4}$$

Therefore the Integrative LASSO can be solved for any $\hat{\beta}_{lk}$ by:

$$\begin{aligned}\hat{\beta}_{lk} = \frac{1}{\sum_{i=1}^{N} x_{lik}^2}\Bigg(&-\frac{1}{N}\sum_{i=1}^{N}\left(y_{li} - \mu_l - \sum_{j=1,j\neq k}^{P} x_{lij}\beta_{lj}\right)x_{lik} \\ &+ \lambda_1\, sign(\beta_{lk}) \\ &+ \lambda_2 \frac{2}{D}\left[(D-1)\beta_{lk} - \left(\sum_{m=1,m\neq l}^{D} \beta_{mk}\right)\right]\Bigg)\end{aligned} \tag{10.5}$$

The derivative of the LASSO penalty function again yields directional derivatives dependant on the sign of β_k. For any β_k a right (positive) and left (negative) derivatives are calculated using the following steps:

$$Let\ \frac{1}{N}\sum_{i=1}^{N}\left(y_{li} - \mu_l - \sum_{j=1, j\neq k}^{p} x_{lij}\beta_{lj}\right)x_{lik} + \lambda_2\frac{2}{D}\left[(D-1)\beta_{lk} - \left(\sum_{m=1, m\neq l}^{D}\beta_{mk}\right)\right] = S(l, y, \mu, x, \beta, \lambda_2)$$

$$Let\ \sum_{i=1}^{N} x_{lik}^2 = Sxx_l$$

$$if\ \beta_k > 0 \begin{cases} rd = \dfrac{-S(l, y, \mu, x, \beta, \lambda_2) + \lambda_1}{Sxx_l} \\ ld = \dfrac{-S(l, y, \mu, x, \beta, \lambda_2) + \lambda_1}{Sxx_l} \end{cases} \tag{10.6}$$

$$if\ \beta_k < 0 \begin{cases} rd = \dfrac{-S(l, y, \mu, x, \beta, \lambda_2) - \lambda_1}{Sxx_l} \\ ld = \dfrac{-S(l, y, \mu, x, \beta, \lambda_2) - \lambda_1}{Sxx_l} \end{cases}$$

$$if\ \beta_k = 0 \begin{cases} rd = \dfrac{-S(l, y, \mu, x, \beta, \lambda_2) + \lambda_1}{Sxx_l} \\ ld = \dfrac{-S(l, y, \mu, x, \beta, \lambda_2) - \lambda_1}{Sxx_l} \end{cases}$$

In order to update β_{lk} for any iteration, if $ld.rd > 0$ then:

$$\widehat{\beta_{lk}} = \beta_{lk} - rd \tag{10.7}$$

10.3.2 My coordinate descent algorithm for the Integrative LASSO

Table 10.1 shows the pseudo code for my coordinate decent algorithm to fit the Integrative LASSO. The algorithm is very similar to the coordinate descent algorithm for the LASSO described in section 2.4.2. Each dataset is optimised separately and repeated over a loop across datasets until convergence is met. Convergence is based on the sum of the absolute differences between iterations from the current iteration loop ($\hat{\beta}$) and the beta estimates from the previous iteration loop (Oldbeta) both within (step 19) and across datasets (step 20). However the convergence threshold can be different for the two loops. An alternative convergence threshold that could be used is the maximum absolute difference across all SNPs. The inclusion of the variance penalty is shown in step 16.

Table 10.1 My pseudo code to fit the Integrative LASSO using the coordinate descent algorithm

• D = the number of datasets • Let d = the dth dataset, where d = {1, …, D} • Let N = the number of subjects in the dataset • Let i = the ith subject, where i = {1, …, N} • Let P = the number of SNPS in each dataset • Let j = the jth SNP, where j = {1, …, P} • x_d = The N x P standardised SNP matrix for the dth dataset • y_d = A continuous phenotype with mean μ_d and standard deviation σ^2 for the dth dataset
1. Specify two penalty thresholds for the LASSO penalty and variance penalty. Call them λ_1 and λ_2. 2. Specify the maximum number of iterations that are to be used within each dataset. Call it NIter1 3. Specify the maximum number of iterations that are to be used across dataset. Call it NIter2

4. Specify convergence threshold > 0 for convergence within a dataset. Call it THRESH1
5. Specify convergence threshold > 0 for convergence across datasets. Call it THRESH2
6. Calculate the intercept which is the mean of y_d. Call it mu_d for all d
7. Calculate sxx_d for all d, where $sxx_d = \sum_{j=0}^{P} x_{dj}^2$
8. Generate a $D\ x\ P$ matrix of initial estimates of length P. Call it Betahat ($\hat{\beta}$)
9. Generate the same $D\ x\ P$ matrix of initial estimates of length P. Call it Oldbeta
10. Set $d = 1$
11. Set $iter2 = 1$
12. Set $iter1 = 1$
13. For each cell in the d^{th} row of Oldbeta, replace the Oldbeta values with those in the d^{th} row of Betahat ($\hat{\beta}$)
14. Set $j = 1$
15. Take the j^{th} column of $\hat{\beta}$, $\hat{\beta}_j$ and remove the d^{th} row. Call this vector OtherBetas
16. Calculate

$$r = \sum_{i=1}^{N}(y_{di} - mu_d)x_{dj} - \lambda_2 \frac{2N}{D}\left((D-1)\,\hat{\beta}_{dj} - \sum OtherBetas\right)$$

17. Calculate the left (ld) and right derivatives (rd)

a. If $\hat{\beta}_{dj} = 0$ $\begin{cases} ld = -r + N\lambda_1 \\ rd = -r - N\lambda_1 \end{cases}$

b. If $\hat{\beta}_{dj} > 0$ $\begin{cases} rd = -r + N\lambda_1 \\ ld = -r + N\lambda_1 \end{cases}$

c. If $\hat{\beta}_{dj} < 0$ $\begin{cases} rd = -r - N\lambda_1 \\ ld = -r - N\lambda_1 \end{cases}$

18. Let $New.beta$ denote the updated Beta estimate. In order to calculate this:
    a. If $rd\ x\ ld \leq 0$ then $\hat{\beta}_{dj} = 0$
    b. If $rd\ x\ ld > 0$ then
        i. Calculate $New.beta_{dj} = \hat{\beta}_{dj} - \frac{rd}{sxx_d}$
        ii. Update $mu_d = mu_d + \left(New.beta_{dj} - \hat{\beta}_{dj}\right)x_{dj}$

iii. Replace $\hat{\beta}_{dj} = New.beta_{dj}$

19. Decide if the convergence criterion within dataset has been met.
    a. If $\sum_{j=1}^{P} \left| \hat{\beta}_{dj} - Oldbeta_{dj} \right| < THRESH1$ and $j = P$ then convergence criterion has been met. Go to Step 20
    b. If $\sum_{j=1}^{P} \left| \hat{\beta}_{dj} - Oldbeta_{dj} \right| \geq Thresh1$ then convergence criterion has not been met.
        i. If $iter1 = NIter1$. Stop. Model has not converged
        ii. If $j = P$, set $iter1 = iter1 + 1$. Go to step 14.
        iii. If $j < P$ & $iter1 < NIter$, set $j = j + 1$. Go to step 15.
20. Decide if the convergence criterion across datasets has been met.
    a. If $\sum_{j=1}^{D} \sum_{j=1}^{P} \left| \hat{\beta}_{dj} - Oldbeta_{dj} \right| < THRESH2$ and $d = D$ then convergence criterion has been met. $\hat{\beta}_{dj}$ contains a matrix of beta estimates obtained by the Integrative LASSO for tuning parameter λ_1 and λ_2.
    b. If $\sum_{j=1}^{D} \sum_{j=1}^{P} \left| \hat{\beta}_{dj} - Oldbeta_{dj} \right| > THRESH2$ then convergence criterion has not been met.
        i. If $iter2 = NIter2$. Stop. Model has not converged
        ii. If $d = D$, set $iter2 = iter2 + 1$. Go to step 12
        iii. If $d < D$, set $d = d + 1$. Go to step 12

10.4 Example of the Integrative LASSO on a test dataset

For this example, a dataset was simulated as described in section 9.3.1 with 5 datasets of 50 independent SNPs and 100 subjects in each dataset. Each of the five causal SNPs explained 5% of the total variance of the phenotype. Each casual SNP was simulated with a positive effect estimate. A seed was set to 1 using the `set.seed()` command in R. 10,000 iterations (NIter1 in Table 10.1) and a convergence threshold of 0.0001 (THRESH1 in Table 10.1) was used for convergence within each dataset. 40 iterations

(NIter2 in Table 10.1) and a convergence threshold of 0.001 (THRESH2 in Table 10.1) was used for convergence within each dataset.

10.4.1 Illustration of the variance penalty

The smallest λ_1 required for a null model was 0.52513. Values of λ_2 ranged between 0 and 0.62. When $\lambda_2 > 0.62$ the model failed to converge within each dataset and is discussed futher in section 10.4.2.

Figure 10.1 shows the coefficient path plot for the 45 non-causal SNPs (top left) and the 5 causal SNPs in all 5 datasets across a range of λ_2 values with no LASSO penalty, $\lambda_1 = 0$. When $\lambda_2 = 0$ the β estimates obtained are the OLS regression estimates for each SNP in each dataset. As the variance penalty increases, these estimates were forced towards the mean β across all datasets which is the β_j produced by pooling all datasets together and fitting an OLS regression. It is clear in Figure 10.1 that the non-causal SNPs were generally being shrunk towards 0, which is the approximate simulated mean for a non-causal SNP. The causal SNPs do not shrink towards 0 but to some value > 0 as their simulated effect is also some positive non-zero value. The shrinkage for each SNP is shown in greater detail in Appendix D.

The addition of the LASSO penalty will also penalise SNPs towards 0, much like the separate LASSO method discussed in Chapter 9. Figure 10.2 shows the coefficient path plot for the 45 non-causal SNPs and the 5 causal SNPs in all 5 datasets across a range of λ_2 values with the LASSO penalty $\lambda_1 = 0.1$. A number of SNP estimates have shrunk to 0 and have been removed from the model. The increase in the variance penalty is further forcing the remaining non-causal SNPs with larger effect estimates towards 0. When $\lambda_2 = 0$, 67 of the 250 SNPs remain in the model including 24 of the 25 casual SNPs (sensitivity = 0.960, specificity = 0.809). By increasing the variance penalty a number of SNPs were forced out of the model but also some SNPs were forced back

into the model. This is all dependant on the mean β estimate for the SNP across the datasets when $\lambda_1 = 0.1$. A total of 71 SNPs were selected when $\lambda_2 = 0.62$ (sensitivity = 1.000, specificity = 0.796), an overall increase of 4 SNPs being selected. The aim for this method is to attempt include a larger number of true positives in the model and the increase in the variance penalty forces the single causal SNP estimated as 0 when $\lambda_2 = 0$ on SNP 10 into the model (Figure 10.2). Of the remaining SNPs forced into the model, the β estimates remain small ($|\beta_{dj}| < 0.0011$).

When $\lambda_1 = 0.2$ and $\lambda_2 = 0$, the model selects 17 SNP, 13 of which are causal SNPs (sensitivity = 0.520, specificity = 0.982). Figure 10.4 shows the effect of the variance penalty as some causals SNPs are forced back into the model. When $\lambda_2 = 0.62$, the model selects 27 SNPs of which 21 are causal SNPs resulting in a higher sensitivity rate and only slightly lower specificity rate (sensitivity = 0.778, specificity = 0.973). In this case, no SNPs were removed from the model when the variance penalty was increased. Further increasing the LASSO penalty to $\lambda_1 = 0.3$ (Figure 10.5) shows similar results, where $\lambda_2 = 0$, 7 causal SNPs were selected (sensitivity = 0.280, specificity = 1.000) whereas $\lambda_2 = 0.62$ selects a further two causal SNPs (sensitivity = 0.360, specificity = 1.000). None of the non-causal SNPs are selected at this point.

In this example, $\lambda_1 = 0.3$ is the largest λ_1 value that will penalise SNPs into the model. After this point increasing λ_1 will shrink the mean β estimates across all datasets towards 0. As the LASSO penalty increases, the β_{dj} estimates shrink towards 0 as does the mean across all datasets. Therefore for $\lambda_1 > 0.3$, the mean β estimates across all datasets becomes too small to be able to force more SNPs into the dataset.

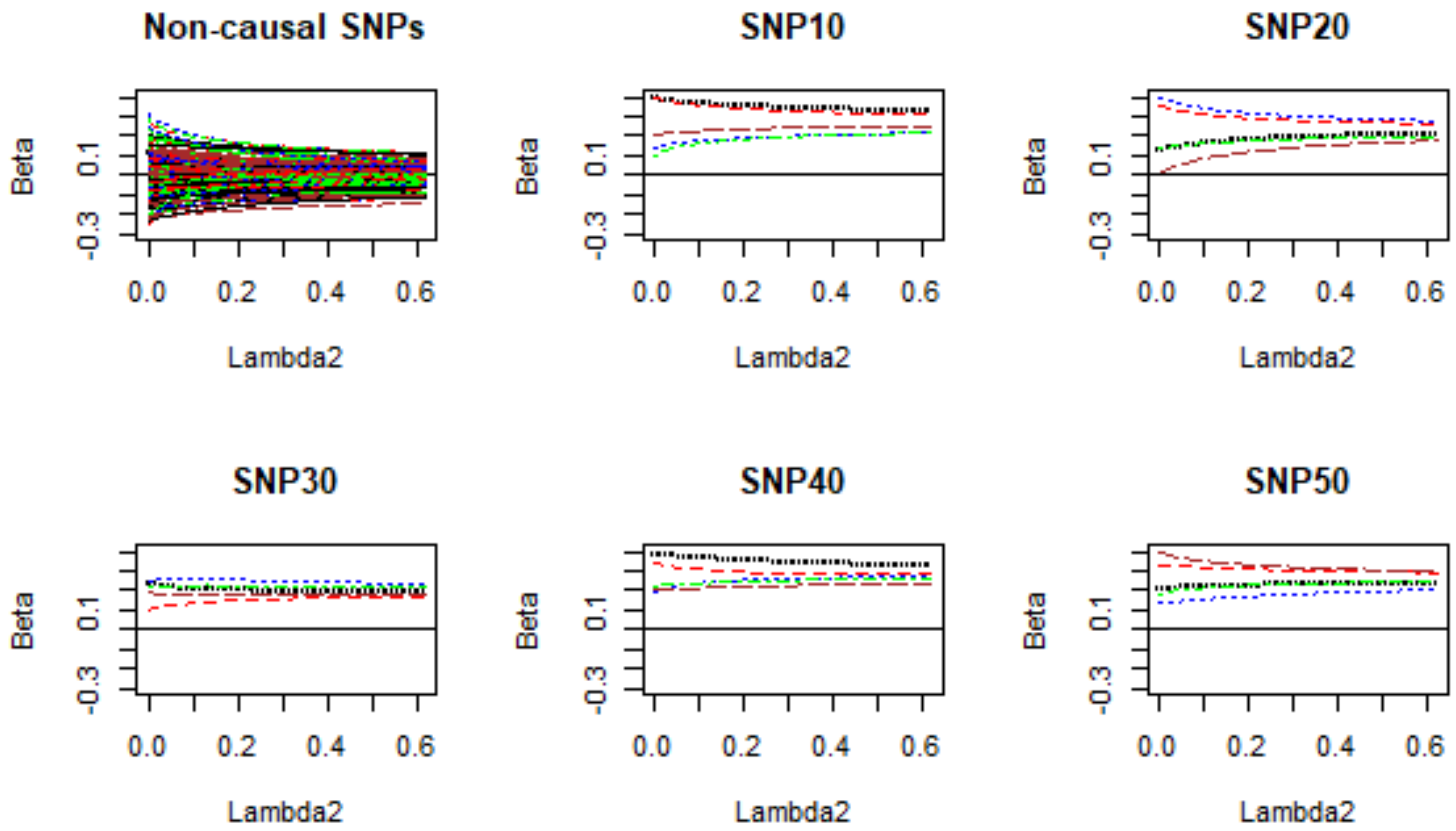


Figure 10.1 Coefficient path plots for the Integrative LASSO when $\lambda_1 = 0$. The top left plot shows the forty-five non-causal SNPs in each of the five datasets. The remaining five plots show the five causal SNPs. Each line represents a SNP from a dataset and the path shows the βcoefficient on the y-axis as the λ_2 penalty increases on the bottom x-axis.

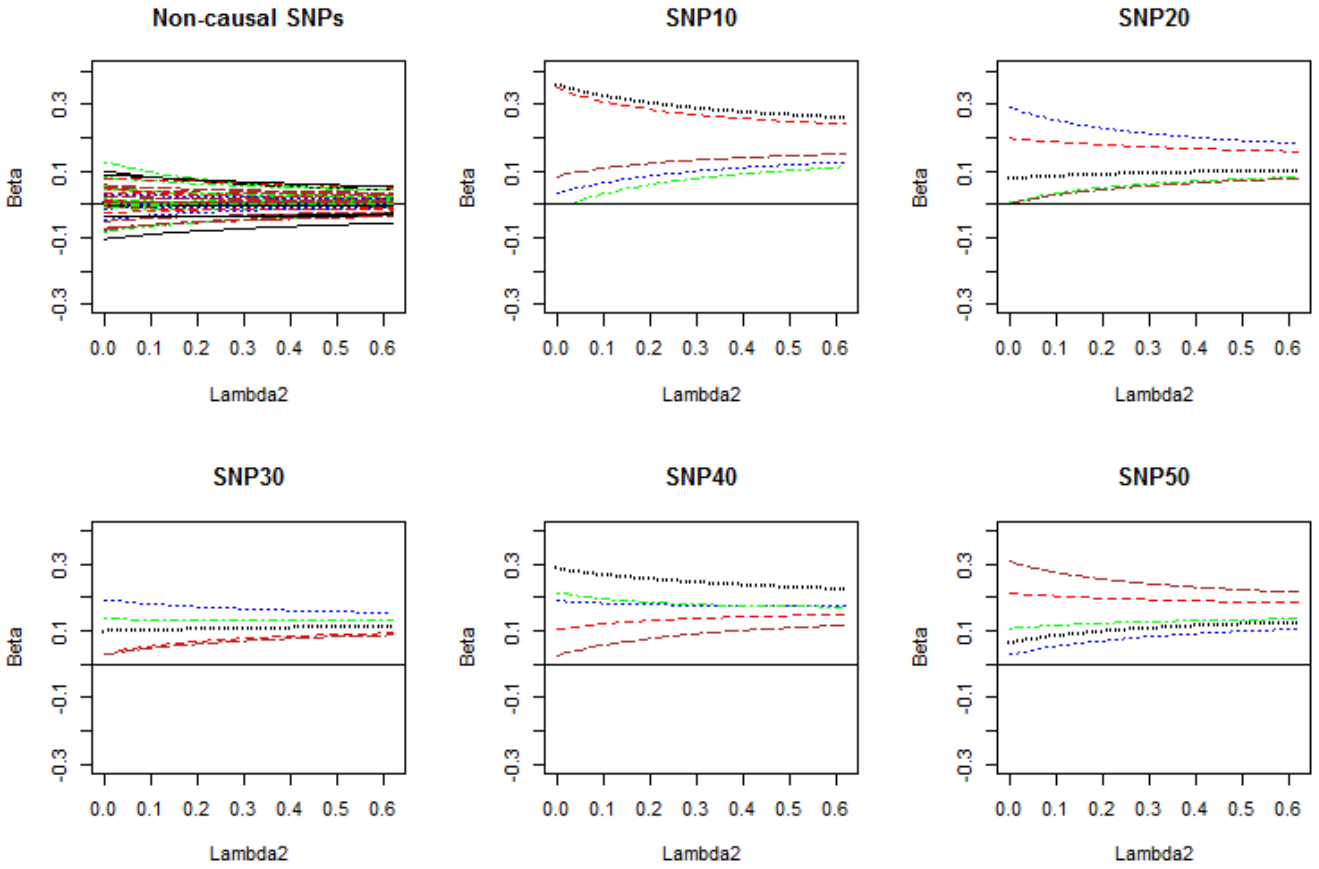


Figure 10.2. Coefficient path plots for the Integrative LASSO when $\lambda_1 = 0.1$. The top left plot shows the forty-five non-causal SNPs in each of the five datasets. The remaining five plots show the five causal SNPs. Each line represents a SNP from a dataset and the path shows the βcoefficient on the y-axis as the λ_2 penalty increases on the bottom x-axis.

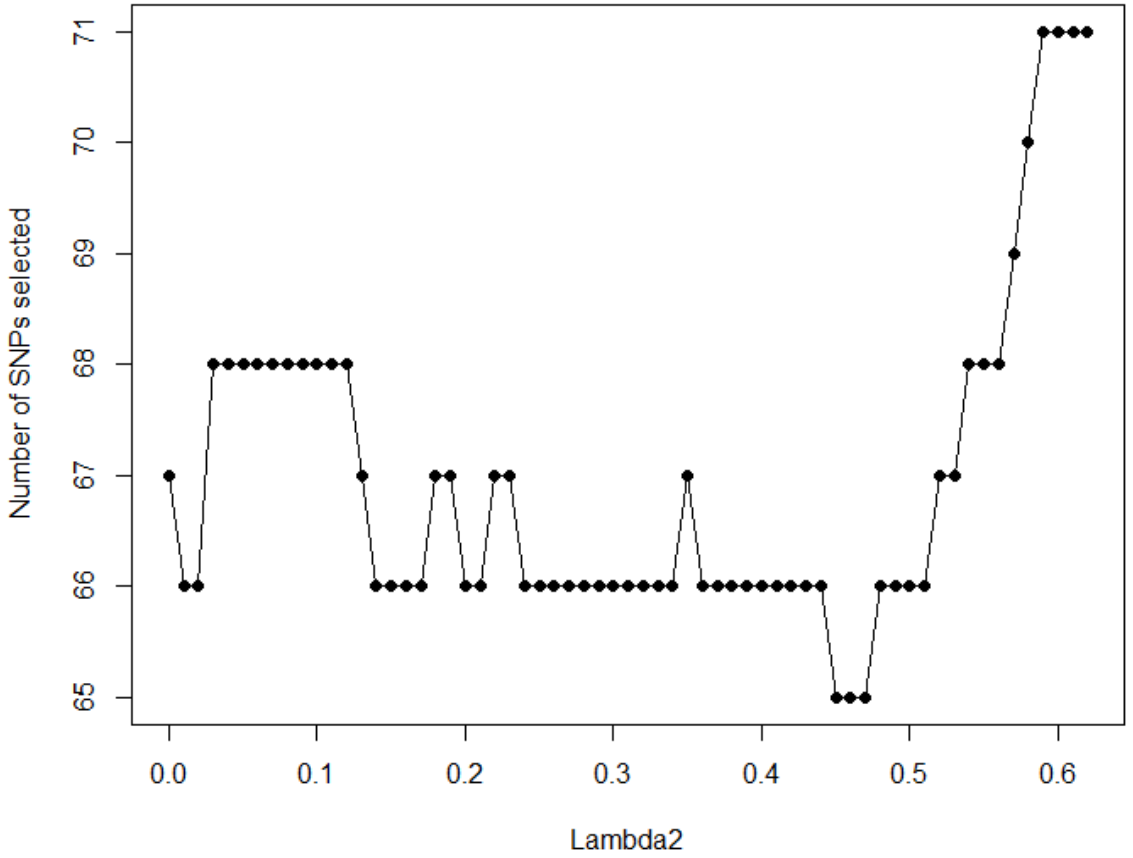


Figure 10.3 Number of SNPs selected by the Integrative LASSO for varying λ_2 penalties with $\lambda_1 = 0.1$

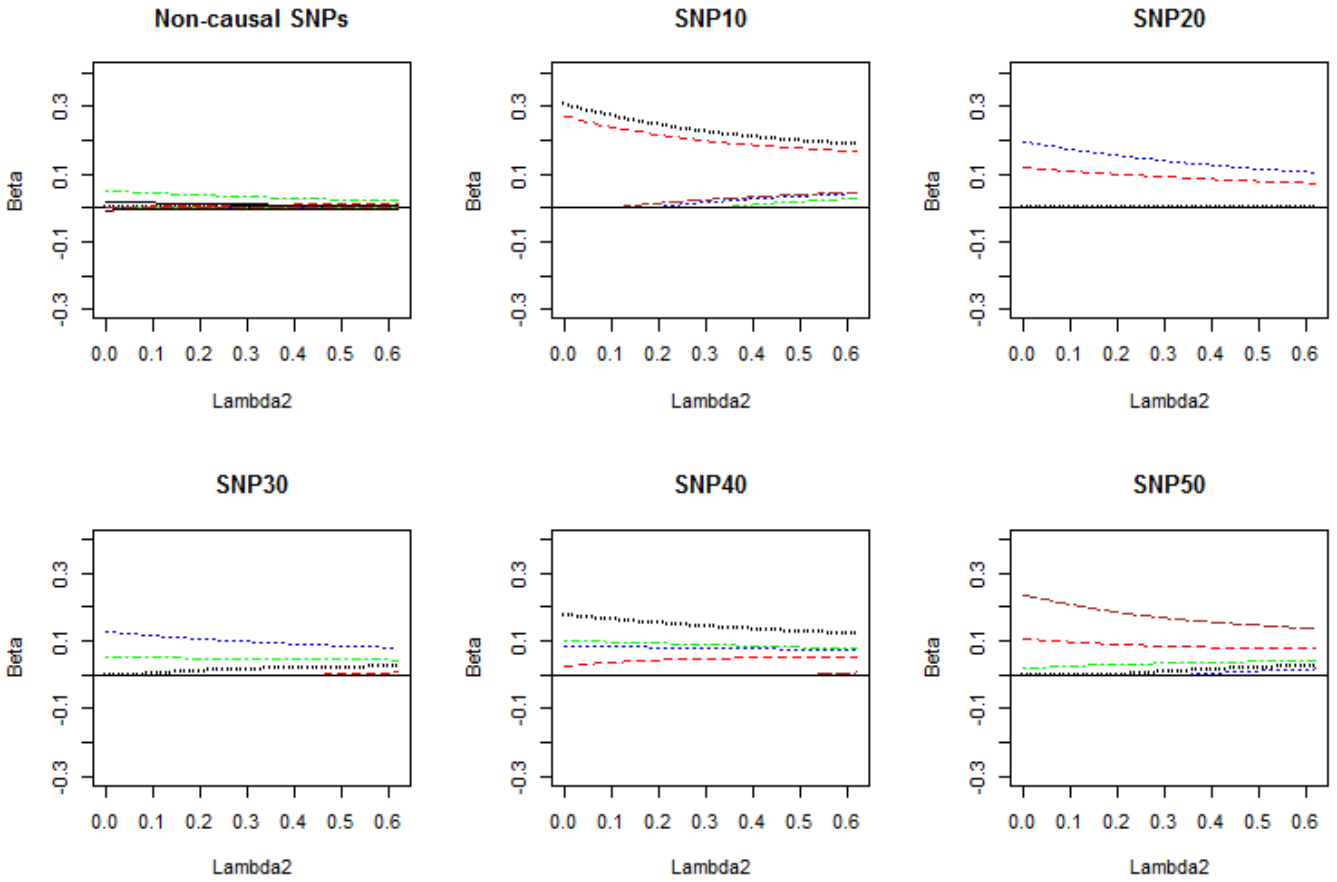


Figure 10.4 Coefficient path plots for the Integrative LASSO when $\lambda_1 = 0.2$. The top left plot shows the forty-five non-causal SNPs in each of the five datasets. The remaining five plots show the five causal SNPs. Each line represents a SNP from a dataset and the path shows the βcoefficient on the y-axis as the λ_2 penalty increases on the bottom x-axis.

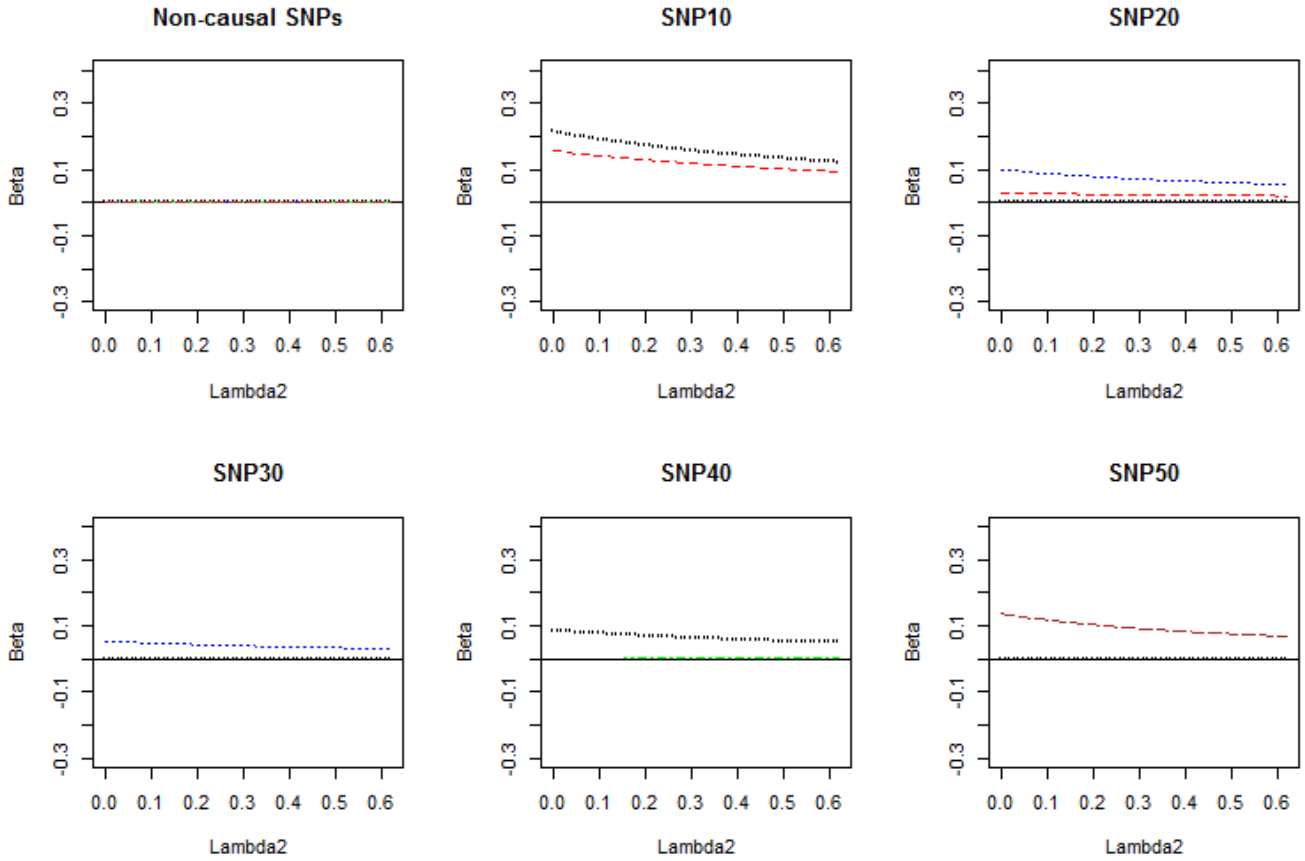


Figure 10.5 Coefficient path plots for the Integrative LASSO when $\lambda_1 = 0.3$. The top left plot shows the forty-five non-causal SNPs in each of the five datasets. The remaining five plots show the five causal SNPs. Each line represents a SNP from a dataset and the path shows the βcoefficient on the y-axis as the λ_2 penalty increases on the bottom x-axis

10.4.2 Convergence issues of the Integrative LASSO

For all values of λ_1, a variance penalty of $\lambda_2 > 0.62$, the model failed to converge within each dataset. This was the case for all seeds although the λ_2 varied. When $\lambda_1 = 0$ and $\lambda_2 = 0.63$, the sum of the absolute difference in β estimates in the last iteration in a single dataset varied between 0.2985 and 0.5867 and increasing the number of iterations to 100,000 did not make any difference to these values. To see if the model would converge across datasets the stop rule in step 19bi (Table 10.1) was removed. Convergence in this case was also not reached with the sum of the absolute beta estimates across all datasets varying between 0.70 and 1.38 after the fourth iteration (Figure 10.6). The lack of convergence was not due to any single SNP in a dataset but for most SNPs in each dataset. Figure 10.7 shows the maximum value of the absolute difference of beta estimates at each iteration was > 0.028 after the fourth iteration. To ensure this was not due to a small sample size, it was increased to 500 in each dataset. Using the same combination of penalties the model did not converge. The plots of the sum and maximum of the absolute difference across all datasets are shown in the Appendix E (Figure E.0.1 and Figure E.0.2). The sum of the absolute difference varied between 0.678 and 1.326 while the maximum absolute difference varied between 0.011 and 0.028.

As the LASSO penalty increases the sum of the absolute difference in β estimates decreases as a number of SNPs are removed from the dataset and these SNPs do converge. Therefore as the λ_1 increases the sum of the absolute difference within and across datasets steadily decreases however, the model still does not converge within dataset but does converge across datasets for $\lambda_1 < 0.38$.

Although the model fails to convergence in every dataset for at some value of λ_2, this becomes less important when considering the Integrative LASSO in terms of variable selection. For variable selection the LASSO penalty has greater importance than the variance penalty. As seen in section 10.4.1, as the λ_2 penalty increases, the number of

SNPs in the model also increases. A large λ_2 penalty may over select the number SNPs in a model. Therefore while the variance penalty is desirable it should be restricted to an extent such that models to not over select too many variables. The example illustrated in section 10.4.1, shows that at the limit where the model fails to converge ($\lambda_2 = 0.62$), the IL selects a few more SNPs into the model but does not select too many more SNPs. Most of these extra SNPs are truly causal SNPs.

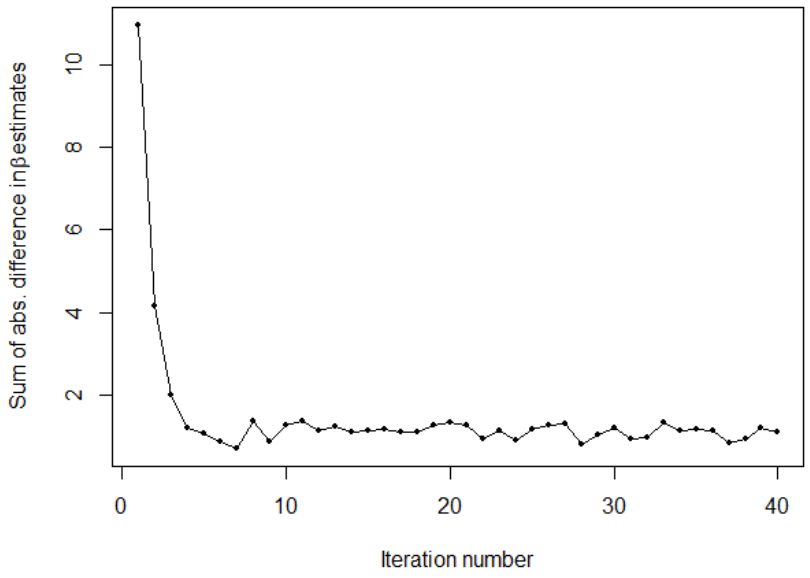


Figure 10.6 Plot of the sum of absolute difference after each iteration across all datasets against its iteration number

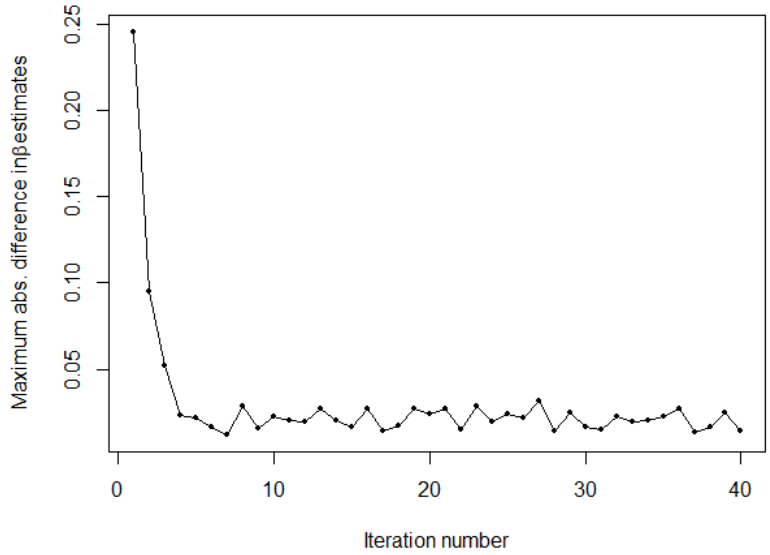


Figure 10.7 Plot of the maximum value in the absolute difference across all SNPs in all datasets after each iteration across all datasets against its iteration number

10.5 Simulation study comparing the Integrative LASSO with the LASSO and meta-LASSO

In this section, I run a simulation comparing the Integrative LASSO against the meta-LASSO, separate LASSO and stacked LASSO. The results of the competing methods are shown in section 9.3.2 and 9.3.3.

10.5.1 Methods

Simulation of datasets is described in section 9.3.1 with the same levels of heterogeneity used in each scenario as shown in Table 9.2. Repeated 10-fold Cross-validation, BIC and the permutation methods were used for tuning parameter selection. For each dataset the minimum λ_1 penalty for a null model was calculated, this value was then rounded up to 2 decimal places and was used as the largest λ_1 for

that dataset. Testing showed that models were failing to converge for some λ_2 values varying between 0.52 and 0.59.Therefore for BIC and the permutation method the λ_2 ranged between 0 and 0.50 in order to allow convergence in every simulated dataset. BIC was calculated using a grid search of all combinations of λ_1 and λ_2 penalties at intervals of 0.01 for each penalty.

A grid search cannot be used for the permutation method as unlike the BIC and CV methods as there is no measure, such as BIC and MSE that is attributed to the selection of tuning parameters. The permutation method selects the tuning parameter based on the smallest λ_1 penalty required for a null model after permutation of the dataset. This value of λ_1 was obtained for each value of λ_2 by selecting the median penalty across 25 permutations. For each value of λ_2, the minimum λ_1 required for a null model was estimated. As the aim is to attempt to select as many true positive SNPs as possible, with the λ_2 corresponding to the smallest λ_1 penalty was selecting as the optimum tuning parameters.

The upper limit for the variance penalty was reduced to 0.35 for 10-fold CV. CV divides the dataset further into smaller sets, this reduction in sample size again lead to issues with convergence in the training set for larger values of λ_2. There were also issues with convergence for rare SNPs using CV, again due to the smaller sample size. Simulated SNPs with a low minor allele count (MAC) in any dataset particularly struggled to converge regardless of the λ_2 penalty applied. The smallest MAF for simulated SNPs was increased to 0.05 from 0.01. In each case, if a SNP previously had a $MAF < 0.05$, 0.04 was added to the MAF before simulating the dataset to allow the SNP to remain relatively rare. The MAF for the causal SNP which previously had a MAF of 0.02 was set to 0.05. Increasing the sample size was not considered as this would increase the power to select causal SNPs and therefore would not be a fair as a comparison with the previous simulation. For repeated CV and BIC, the optimum combination of tuning parameters was performed using a grid search of all combinations of λ_1 and λ_2 penalties at intervals of 0.01 for each penalty. The combination of λ_1 and λ_2 that

produced the smallest MSE across the 10 folds was selected for CV, likewise the smallest BIC value and its corresponding combination of λ_1 and λ_2 was selected for the BIC. A total of 1,000 simulations were run with 10 repetitions for each seed 1-101. Seed 56 was omitted as it was omitted in the simulation study conducted in section 9.3. Both the single selection and replication measures were considered as measures for variable selection (section 9.3.1).

10.5.2 Results

10.5.2.1 Results for high variance explained scenario

Table 10.2 compares the sensitivity and specificity rate the Integrative LASSO against the meta-LASSO, stacked LASSO and separate LASSO methods (see section 9.3.2) for the single selection measure. In this scenario, the IL performs well in terms of sensitivity rates compared to the competing methods using 10-fold CV. Only the stacked LASSO, which selected all causal SNPs in the 1,000 simulations, produced a higher sensitivity rate. The specificity rate however was much lower than the competing methods (0.538). As shown in Table 10.3 this is due to a large mean number of SNPs selected. On average half of all SNPs were selected in the model (mean = 128.850, S.D. = 27.424). Figure 10.4 plots all the estimated λ_1 and λ_2 penalties over the 1,000 simulations and shows that CV consistently selected a relatively small λ_1 and relatively λ_2 large penalty. A combination of these penalties will include a large number of SNPs in the model. The permutation method also performs poorly. Although not many non-causal SNPs are selected, the IL does not select may true positives either and produces the lowest sensitivity rate across all methods and tuning parameter selection methods.

The BIC worked well for the IL in this scenario selecting every causal SNP across the 1,000 simulations. Unlike CV, the BIC maintained a high specificity rate (0.960) and on

average selected 9 non-causal SNPs (Table 10.3). The stacked LASSO produced a similar sensitivity rate but a slightly higher specificity rate (0.982).

Table 10.2 Mean and standard deviation of sensitivity and specificity results using the single selection measure in a high variance explained scenario using the meta-LASSO, stacked LASSO, seperate LASSO and Integrative LASSO with Cross-vaildation, BIC and permutation method as tuning parameter selection methods over 1,000 simulations.

Method	Cross-validation		BIC		Permutation method	
	Sens	Spec	Sens	Spec	Sens	Spec
Meta-LASSO	0.986 ± 0.028	0.992 ± 0.016	0.972 ± 0.058	0.995 ± 0.008	0.512 ± 0.211	1.000 ± 0.000
Stacked LASSO	1.000 ± 0.000	0.764 ± 0.122	1.000 ± 0.006	0.982 ± 0.021	0.999 ± 0.015	0.995 ± 0.011
Separate LASSO	0.857 ± 0.096	0.803 ± 0.064	0.702 ± 0.129	0.948 ± 0.023	0.486 ± 0.092	0.991 ± 0.006
Integrative LASSO	0.997 ± 0.013	0.538 ± 0.121	1.000 ± 0.000	0.960 ± 0.020	0.390 ± 0.103	0.996 ± 0.004

Table 10.3 Mean and standard deviation summary statistics for variable selection using the Integrative LASSO over 1,000 simulations in the high variance explained scenario.

Tuning parameter selection method	Lambda1	Lambda2	Number of SNPs selected	Number of true positive SNPs selected	Number of false positive SNPs selected
Cross-validation	0.056 ± 0.015	0.314 ± 0.036	128.850 ± 27.424	24.914 ± 0.324	103.936 ± 27.337
BIC	0.075 ± 0.007	0.102 ± 0.166	34.002 ± 4.445	25.000 ± 0.000	9.002 ± 4.445
Permutation method	0.279 ± 0.003	0.254 ± 0.147	10.595 ± 2.721	9.761 ± 2.581	0.834 ± 0.936

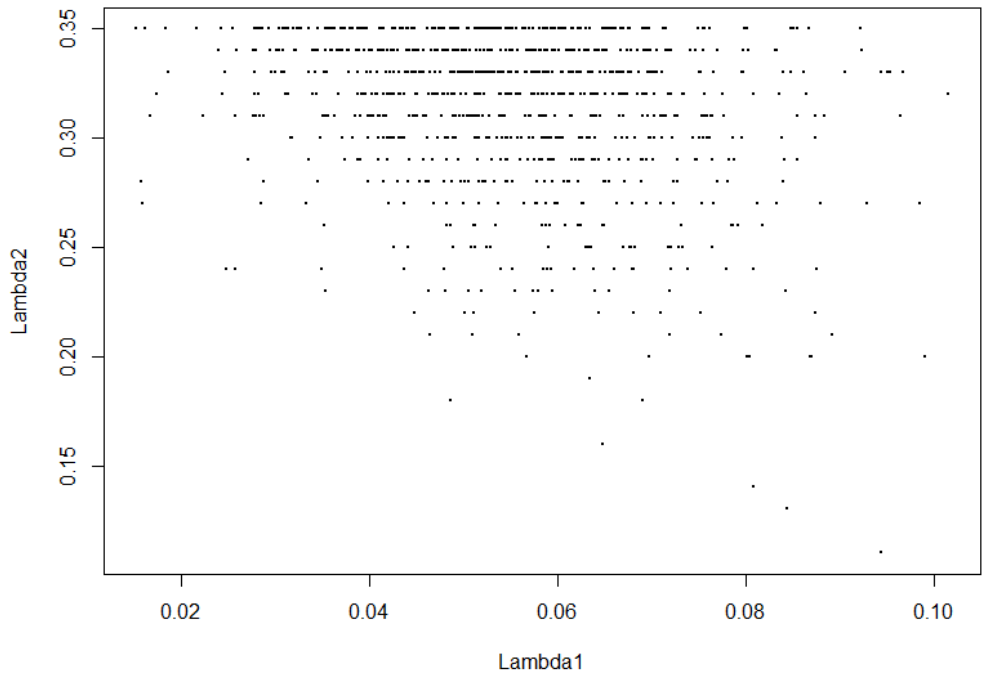


Figure 10.8 Scatter plot of the selected $\lambda_1$and λ_2 values for each of the 1,000 simulations using repeated 10-fold Cross-validation in the high variance explained scenario

Table 10.4 compares the sensitivity and specificity rate for each method using the replication measure. For CV the specificity rate is lower using this measure (0.272) compared to single selection measure (0.538). In comparison the meta-LASSO produces high sensitivity and specificity rates (0.998 and 0.988) and is clearly the best method for variable selection by the replication measure using CV. The IL shows superior performance to the meta-LASSO using the permutation method with a higher sensitivity rate. This sensitivity rate, however is lower than the separate LASSO which shows the best performance as all three methods produce the same specificity rates however the separate LASSO has the highest sensitivity rate (0.793). Of all combinations of integrative analysis methods and tuning parameter selection methods the IL with BIC shows the best performance as it selects all causal SNPs and has a high specificity rate (0.984) using the replication measure.

Table 10.4 Mean and standard deviation of sensitivity and specificity results for the proportion of replicated results in a high variance explained scenario using the meta-LASSO, stacked LASSO, seperate LASSO and Integrative LASSO with Cross-vaildation, BIC and permutation method as tuning parameter selection methods over 1,000 simulations.

Method	Cross-validation		BIC		Permutation method	
	Sens	Spec	Sens	Spec	Sens	Spec
Meta-LASSO	0.998 ± 0.018	0.988 ± 0.024	0.989 ± 0.047	0.992 ± 0.014	0.517 ± 0.213	1.000 ± 0.000
Separate LASSO	0.997 ± 0.023	0.744 ± 0.136	0.973 ± 0.082	0.977 ± 0.028	0.793 ± 0.177	0.999 ± 0.004
Integrative LASSO	1.000 ± 0.000	0.272 ± 0.167	1.000 ± 0.000	0.984 ± 0.022	0.628 ± 0.207	1.000 ± 0.002

10.5.2.2 Results for varying levels of heterogeneity

In this simulation all three tuning parameter selection methods for varying levels of heterogeneity performed poorly compared to the simulation results shown in section 9.3.3.

Table 10.5, Table 10.6 and Table 10.7 show the mean and standard deviation sensitivity and specificity rates for the IL method using 10-fold CV, BIC and permutation method. In each case the IL produced the lowest sensitivity rate of all methods. This suggest that the IL lacks power to detect associations compared to the competing methods and that the stacked LASSO with the permutation method for tuning parameter selection may be the best method for variable selection in integrative analyses.

As seen in previous section, CV tends to select large models with the IL which were also the case in this simulation. A mean of 18 SNPs were selected for the lower heterogeneity levels (Table 10.8). Most of the selected SNPs were non-causal SNPs (specificity = 0.933 – 0.934) and only an average of 3 causal SNPs were selected (sensitivity = 0.147 – 0.149). The number of SNPs selected produced a high standard deviation of the mean, which was also seen in the high variance explained scenario (Table 10.3.) Figure 10.9 plots the selected λ_1 and λ_2 values against each other. The plots show that in each case there seem to be two distinct groupings in λ_1. This is in contrast to the high variance explained scenario where there were no divisions in the distribution of λ_1 (Figure 10.8). One grouping selects a small value of λ_1 between 0 and 0.20, the other larger values of λ_1 between 0.25 and 0.35. The majority of the simulations selected a large λ_1, as the median number of SNPs selected at each level of heterogeneity was approximately 1. The group of small λ_1 values seemed to be a correlated with λ_2 as larger averaging penalties were being selected alongside small LASSO penalties. This combination of penalties would increase the numbers of SNPs in the model. A large LASSO penalty will select a small number of SNPs in the model regardless of the variance penalty that is selected, as shown in section 10.4.1. These two contrasts in λ_1 selection lead to a large variance in the number of SNPs selected. In order to reduce the variance repeated CV could be used, however this is at the expense of computational time.

Variable selection by the BIC produced the lowest sensitivity rate across all simulations. One of the problems with the BIC in these scenarios was that, nearly all (n ≥ 938) of the final models did not utilise the variance penalty, instead selecting $\lambda_2 = 0$. This is because the BIC penalises on the number of parameters in the model and therefore is likely to select the simplest model possible. This was also the case in the high variance explained scenario where 531 of the 1,000 models selected $\lambda_2 = 0$. Of the three tuning parameter selection methods, the permutation method shows the best performance for variable selection using the IL as there is a higher sensitivity rate and similar specificity rate than the BIC.

Table 10.5 Mean and standard deviation of sensitivity and specificity results using the single selection measure for varying levels of heterogeneity using the meta-LASSO, stacked LASSO and seperate LASSO with Cross-vaildation over 1,000 simulations.

Heterogeneity	Meta - LASSO		Stacked LASSO		Separate LASSO		Integrative LASSO	
	Sensitivity	**Specificity**	**Sensitivity**	**Specificity**	**Sensitivity**	**Specificity**	**Sensitivity**	**Specificity**
Baseline	0.264 ± 0.197	0.992 ± 0.016	0.741 ± 0.291	0.832 ± 0.137	0.177 ± 0.104	0.929 ± 0.048	0.147 ± 0.233	0.934 ± 0.126
Low	0.257 ± 0.191	0.992 ± 0.016	0.732 ± 0.294	0.834 ± 0.138	0.178 ± 0.103	0.929 ± 0.048	0.148 ± 0.230	0.933 ± 0.126
Mid	0.250 ± 0.188	0.991 ± 0.016	0.705 ± 0.306	0.840 ± 0.138	0.179 ± 0.103	0.929 ± 0.049	0.149 ± 0.228	0.933 ± 0.125
High	0.174 ± 0.158	0.989 ± 0.018	0.432 ± 0.329	0.893 ± 0.126	0.181 ± 0.104	0.927 ± 0.049	0.125 ± 0.189	0.944 ± 0.105

Table 10.6 Mean and standard deviation of sensitivity and specificity using the single selection measure for varying levels of heterogeneity using the meta-LASSO, stacked LASSO and seperate LASSO with BIC over 1,000 simulations.

Heterogeneity	Meta - LASSO		Stacked LASSO		Separate LASSO		Integrative LASSO	
	Sensitivity	**Specificity**	**Sensitivity**	**Specificity**	**Sensitivity**	**Specificity**	**Sensitivity**	**Specificity**
Baseline	0.095 ± 0.105	0.999 ± 0.005	0.076 ± 0.117	0.999 ± 0.004	0.049 ± 0.047	0.992 ± 0.006	0.014 ± 0.025	0.998 ± 0.003
Low	0.098 ± 0.107	0.998 ± 0.005	0.081 ± 0.121	0.999 ± 0.004	0.049 ± 0.045	0.992 ± 0.005	0.014 ± 0.023	0.998 ± 0.003
Mid	0.096 ± 0.105	0.998 ± 0.005	0.071 ± 0.113	0.999 ± 0.004	0.050 ± 0.048	0.993 ± 0.006	0.014 ± 0.025	0.998 ± 0.003
High	0.086 ± 0.100	0.997 ± 0.007	0.049 ± 0.088	0.998 ± 0.006	0.055 ± 0.054	0.992 ± 0.006	0.016 ± 0.026	0.998 ± 0.003

Table 10.7 Mean and standard deviation of sensitivity and specificity results using the single selection measure for varying levels of heterogeneity using the meta-LASSO, stacked LASSO and seperate LASSO with the permutation method over 1,000 simulations.

Heterogeneity	Meta - LASSO		Stacked LASSO		Separate LASSO		Integrative LASSO	
	Sensitivity	**Specificity**	**Sensitivity**	**Specificity**	**Sensitivity**	**Specificity**	**Sensitivity**	**Specificity**
Baseline	0.307 ± 0.173	0.992 ± 0.011	0.423 ± 0.217	0.987 ± 0.017	0.073 ± 0.051	0.987 ± 0.007	0.038 ± 0.038	0.995 ± 0.005
Low	0.299 ± 0.171	0.992 ± 0.011	0.416 ± 0.217	0.987 ± 0.017	0.073 ± 0.050	0.987 ± 0.007	0.038 ± 0.037	0.995 ± 0.005
Mid	0.286 ± 0.167	0.992 ± 0.011	0.397 ± 0.216	0.987 ± 0.017	0.073 ± 0.051	0.987 ± 0.017	0.038 ± 0.037	0.995 ± 0.005
High	0.171 ± 0.141	0.992 ± 0.011	0.224 ± 0.183	0.986 ± 0.018	0.074 ± 0.050	0.987 ± 0.007	0.041 ± 0.039	0.995 ± 0.005

Table 10.8 Mean and standard deviation summary statistics for variable selection using the Integrative LASSO with Cross-Vaildation as the tuning parameter selection method over 1,000 simulations across varying levels of heterogeneity.

Heterogeneity	Lambda1	Lambda2	Number of SNPs selected	Number of true positive SNPs selected	Number of false positive SNPs selected
Baseline	0.246 ± 0.086	0.192 ± 0.104	18.644 ± 33.937	3.686 ± 5.817	14.958 ± 28.425
Low	0.245 ± 0.086	0.191 ± 0.105	18.708 ± 33.848	3.691 ± 5.761	15.017 ± 28.399
Mid	0.244 ± 0.086	0.190 ± 0.105	18.862 ± 33.529	3.724 ± 5.707	15.138 ± 28.117
High	0.250 ± 0.082	0.185 ± 0.106	15.647 ± 28.022	3.128 ± 4.736	12.519 ± 23.573

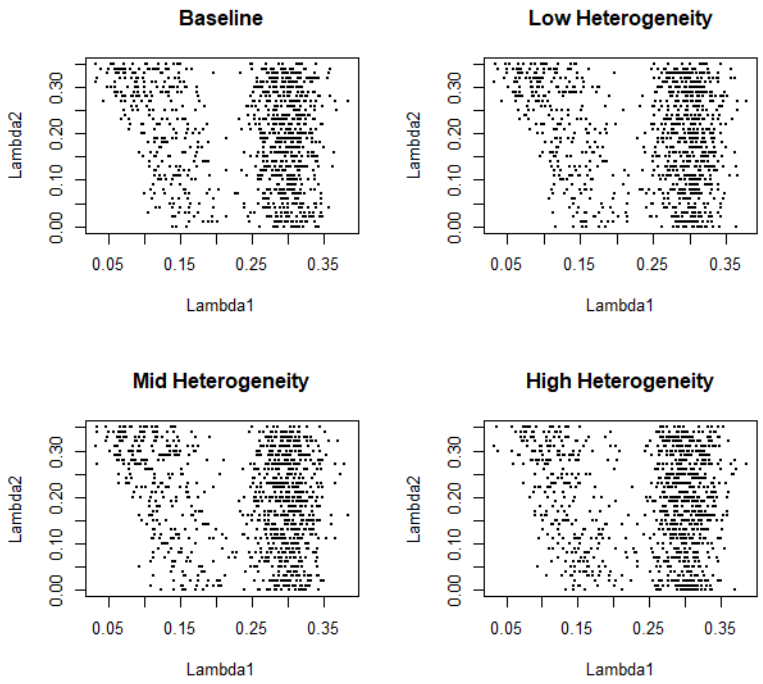


Figure 10.9 Scatter plot of the selected $\lambda_1$and λ_2 values for each of the 1,000 simulations using 10-fold Cross-validation

Table 10.9 shows the results of the simulation using the replication measure. Due to the lack of power to select variables in this scenario the sensitivity rate decreases and specificity rate increases for the BIC and permutation methods. Of the SNPs selected using these methods there were not many SNPs that were replicated. In only 11 of the 1,000 models were one of the causal SNPs replicated, and in one of these cases, two causal SNPs were replicated. Of the three methods, the meta-LASSO performed the best for variable selection using the replication measure as it produced the highest sensitivity rate as well as a high specificity rate (≤ 0.983) for all tuning parameter selection methods.

Table 10.9 Mean and standard deviation of sensitivity and specificity results for the proportion of replicated results in across varying levels of heterogeneity using the Integrative LASSO with Cross-vaildation, BIC and permutation method as tuning parameter selection methods over 1,000 simulations.

Heterogeneity	Cross-validation				BIC				Permutation method			
	Single selection		Replication		Single selection		Replication		Single selection		Replication	
	Sens	Spec	Sens	Spec	Sens	Spec	Sens	Spec	Sens	Spec	Sens	Spec
Baseline	0.147 ± 0.233	0.934 ± 0.126	0.200 ± 0.346	0.911 ± 0.196	0.014 ± 0.025	0.998 ± 0.003	0.002 ± 0.024	1.000 ± 0.000	0.038 ± 0.038	0.995 ± 0.005	0.015 ± 0.054	1.000 ± 0.003
Low	0.148 ± 0.230	0.933 ± 0.126	0.202 ± 0.343	0.910 ± 0.197	0.014 ± 0.023	0.998 ± 0.003	0.002 ± 0.019	1.000 ± 0.001	0.038 ± 0.037	0.995 ± 0.005	0.013 ± 0.050	1.000 ± 0.003
Mid	0.149 ± 0.228	0.933 ± 0.125	0.202 ± 0.340	0.910 ± 0.195	0.014 ± 0.025	0.998 ± 0.003	0.002 ± 0.024	1.000 ± 0.001	0.038 ± 0.037	0.995 ± 0.005	0.014 ± 0.051	1.000 ± 0.002
High	0.125 ± 0.189	0.944 ± 0.105	0.178 ± 0.312	0.929 ± 0.161	0.016 ± 0.026	0.998 ± 0.003	0.003 ± 0.025	1.000 ± 0.000	0.041 ± 0.039	0.995 ± 0.005	0.016 ± 0.054	1.000 ± 0.002

10.6 Discussion

The simulation study showed that the Integrative LASSO performed poorly compared to the competing methods of the meta-LASSO, stacked LASSO and separate LASSO, especially in terms of the sensitivity rate. The exception to this is when applying the BIC in a high powered scenario where the IL was able to select all causal SNPs. The lack of power to select causal SNPs were for a number of reasons. The first is the LASSO penalty (λ_1) which is applied separately on each dataset individually. The simulations have shown that stacked LASSO and meta-LASSO were the best performing methods for variable selection and both of these methods include a penalty across all datasets rather than individually. This provides greater power to select causal SNPs as the sample size increases. The penalty on individual datasets is required however in order to allow for averaging across datasets.

The second reason was the lack of flexibility for the LASSO penalty. Both the separate LASSO and IL apply penalties to datasets separately; however in most cases the separate LASSO outperformed the IL in terms of selecting causal SNPs. The difference was that the separate LASSO was allowed to have a different λ in each dataset whereas the IL was forced to have the same λ penalty in each dataset and therefore was less flexible. By fixing the same penalty in all datasets the IL will restrict variable selection in certain datasets where the λ_1 required for a null model. The IL could be allowed to have different λ_1 penalties in each dataset as shown in (10.8). This would essentially make the IL the same as the separate LASSO with a variance penalty. However allowing separate λ_1 penalties for each dataset comes at great computation cost, when selecting the optimum combination of λ_{1d} and λ_2. The BIC and CV methods both use a two-dimensional grid search to calculate the optimum combination of λ_1 and λ_2. By allowing each dataset a separate λ_{1d} the grid search would be in six dimensions for five datasets and the number of combinations required to find the optimal tuning parameter increases.

$$\hat{\beta}(\lambda) = \arg\min_{\beta} \frac{1}{2N} \sum_{d=1}^{D} \sum_{i=1}^{N} \left(y_{di} - \sum_{j=1}^{P} x_{dij} \beta_{dj} \right)^2 + \sum_{d=1}^{D} \lambda_{1d} \sum_{j=1}^{P} |\beta_{dj}| + \lambda_2 \sum_{d=1}^{D} \sum_{j=1}^{P} \left(\beta_{dj} - \frac{1}{D} \sum_{d=1}^{D} \beta_{dj} \right)^2 \quad (10.8)$$

where,

λ_{1d} is a vector of D λ_1 penalties for each of the D datasets

The range of averaging penalties that were used was restricted due to issues with convergence for some large λ_2. As the variance penalty is based on the fused LASSO other algorithms that fit the fused LASSO such as augmented Lagrangian method (ALM) could be considered (248). Allowing models to converge for larger λ_2 potentially allows more SNPs to be selected. The example shown in section 10.4.1 only penalised SNPs back into the model for a lower LASSO penalty, with the ability for the model to converge at a larger variance penalty, this may allow SNPs to enter the model for a larger LASSO penalty.

Results in the high variance explained scenario showed that the BIC performed well, selecting all causal SNPs (Table 10.2). The LASSO penalty for the permutation method was selected as the smallest median λ_1 over 25 permutations with the corresponding λ_2 penalty was selecting as the optimum tuning parameters. Allowing λ_1 to be the smallest possible value would allow a greater number of SNPs and potentially true positives to be selected. The permutation method however produced a lower sensitivity rates using both single SNP and replication measures compared to most competing methods. 10-fold CV also performed poorly in both measure, selecting a large mean number of SNPs in both scenarios which included a number of false positives (Table 10.3 and Table 10.8). Repeated CV should be used due to the high variance in both λ_1 and λ_2 estimates.

10.7 Conclusion

In this chapter, I proposed a novel approach to penalised regression in integrative analyses. The Integrative LASSO method applies two penalties, the first is a LASSO penalty on each dataset individually, and the second penalises SNPs towards the mean β across all datasets. Testing showed that the model was unable to converge for a large variance penalty using coordinate descent algorithm and therefore other algorithms such as augmented Lagrangian method (ALM) (see section 2.4.1) could instead be used. A large variance penalty is of less importance to the LASSO penalty for variable selection as it penalty tends to increase the number of SNPs selected in a model. As the IL uses a fusion penalty the fused-LASSO signal approximator (FLSA) by Friedman *et al*. (51) could also be considered, however this would only give approximate solutions rather than exact ones.

The simulation study showed that the sensitivity rate for the IL was lower than the competing methods with the exception of BIC in the high variance explained scenario. The poor performance of the IL compared to the meta-LASSO and stacked LASSO is because the IL penalises datasets individually rather than pooled across all dataset. The poor performance relative to the separate LASSO is likely to be because the IL applies the same penalty in all datasets where the separate LASSO allows each dataset to have a different tuning parameter. By fixing the same penalty in all datasets the IL will restrict variable selection in certain datasets where the λ_1 required for a null model. The IL method could however be modified to allow a different LASSO penalty in each dataset which would give the method similar or superior performance to the separate LASSO.

Of all the methods used in the simulation study, the stacked LASSO using the permutation method for tuning parameter selection showed the best performance in terms of variable selection in both the high variance explained scenario (Table 10.2) and the lower powered scenario where the heterogeneity varied (Table 10.7) using the single selection measure. The performance of the stacked LASSO is only beaten by the meta-LASSO for the BIC in the low powered scenario with varying levels of heterogeneity (Table 10.6); however the permutation method outperforms the BIC in terms of variable selection.

11 Conclusions and future work

In the past decade, genome-wide association studies play a key role in understanding complex diseases by identifying many casual genetic variants or regions associated with disease. The aim of these studies is to identify causal SNPs of regions in order to develop new therapies, improve diagnosis and better disease prevention (3). For example, four GWAS studies have identified the rs12916 SNP (MAF ~ 0.4) on the HMGCR gene (Table 4.5) to have a small but significant effect with LDL (171,179,180,182). The SNP has been shown to increase the LDL levels by 2.8 mg/dL for every minor C allele (180), and has been the target for therapeutic drugs such as statins that are designed to lower LDL cholesterol and are used by tens of millions of people world-wide (249).

Current methods to identify associated SNPs in GWAS are not without their flaws however. Methods such as Bonferroni correction and FDR are performed on a univariate level and as shown both in the literature (127) and this thesis (see section 4.8.2) these methods are also unable to account for LD therefore select multiple associated SNPs within a region. In such regions of high LD it is often difficult to determine which of a group of SNPs the causal variant is. The causal SNP may not necessarily be the top SNP by P-value which could be due to a combination of random error and LD with the truly causal SNP. There is also a case that the causal SNP may not be present in the GWAS dataset and all SNPs in an associated region are associations rather than the causal SNP. In each case however, further investigation is often required to determine the truly associated SNP.

In this thesis, I consider penalised regression, specifically the LASSO (9) as an alternative method for variable selection in both single and multi-cohort datasets. The three main aims of this thesis were:

1. Apply the LASSO to discover genetic associations with Low-density Lipoprotein (LDL-c) in the Genetic Regulation of Arterial Pressure of Humans in the Community (GRAPHIC) study (10). Compare the results of this analysis with analyses performed using Bonferroni correction, FDR and the current literature.

2. Determine the best methods to reduce the dimensionality of the dataset such that the impact on variable selection is minimised.

3. Compare the current penalised regression methods for integrative analysis by a simulation study and also present and test an alternative approach for integrative analysis.

11.1 Summary of findings

11.1.1 Aim 2: Determining the best methods to reduce the dimensionality of the dataset

In order to fully address Aim 1, Aim 2 needed to be addressed. In section 4.7, I described that due to the computational intensivity of the LASSO on such a large dataset (591,774 SNPs and 979 subjects), I was unable to fit the LASSO to the GRAPHIC study dataset initially due to first memory restrictions and then time restrictions. Therefore to be able to perform a GWAS on the GRAPHIC cohort SNP pruning was required. In Chapter 7, I conducted a simulation study to determine the effects of SNP pruning on LASSO models. As was the case throughout this thesis, three tuning parameter selection methods were considered; repeated 10-fold CV, BIC and the permutation method. I considered three pruning approaches, P-value pruning, LD

pruning and LD clumping. All three of these approaches were implemented in my Prune package written in R and is described in section 6.4.

Both the repeated 10-fold CV and BIC performed poorly regardless of pruning method. Both tuning parameter selection methods were influenced by pruning and both selected larger sized models and therefore as the pruning threshold increased. This was particularly the case when pruning by P-value and selecting the tuning parameter by CV. This particular combination of pruning and tuning parameter selection methods yielded the lowest specificity rate across all scenarios. This has also been seen in previous studies in which the combination of P-value pruning and CV has selected a large model (11,146,149). LD clumping ensures that the top independent associations remain in the dataset and the SNPs that are correlated with the top associations are removed.

In contrast the permutation method showed good performance for each pruning method. Although it was the most conservative of the three tuning parameter selection methods, this greatly reduced the number of false positives selected with an average of 1 false positive SNP selected in every scenario. Unlike the BIC and CV methods, the permutation method was robust against pruning. There was little difference in the sensitivity and specificity rate regardless of the pruning method and threshold used. Of the three pruning methods, LD clumping was able to select a slightly higher number of true positives and therefore I concluded this was the best combination of pruning method and tuning parameter selection method for variable selection in GWAS data.

11.1.2 Aim 1: Application of LASSO to the GRAPHIC study to identify associations with LDL

11.1.2.1 The LASSO on chromosome 19

A genetic association study was conducted on chromosome 19, as the GRAPHIC dataset (Table 4.14 and Figure 4.1) and previous literature (Table 4.5) both suggested that this chromosome contained a number of associations.

The Bonferroni correction method, selected four SNPs, rs7412, rs4420638, rs2075650 and rs445925. However as SMAs are unable to account for the LD between SNPs and therefore some correlation between the four selected SNP were discovered. Correlations were found between both rs7412 and rs445925 (r^2 = 0.712, Figure 4.12) and rs4420638 and rs2075650 (r^2 = 0.416, Figure 4.13), resulting in only two independent signals being selected from the same region. Both rs7412 and rs4420638 have been identified to have an association with LDL in previous studies.

The FDR method selected 13 SNPs (Table 4.17) and two novel genetic regions, one between ZNF520 and ZNF567 genes (Figure 4.18) and the second between DNM2 and CARM1 genes (Figure 4.19). Both these regions have not been identified in previous studies and require further investigation. The identified region between DNM2 and CARM1 is close (~200kb) to the LDLR gene which has been identified in multiple previous studies (Table 4.6). It is not known if these two regions are truly associated and therefore requires further investigation.

The LASSO using the BIC and permutation methods was able to select the top associations in the four regions selected by the FDR analysis. The difference between the two methods was that the LASSO eliminates the correlated SNPs from the model selecting four SNPs compared to thirteen and therefore produces a simpler model. This suggested that for this analysis that the LASSO using either the BIC or permutation method produces a similar performance to the FDR method in terms of variable selection but is able to remove correlated data. Repeated 10-fold CV selected 41 SNPs.

The analysis using the LASSO was repeated using a range of pruning methods and threshold that were used in the simulation study discussed in the previous section. Results showed that both the BIC and permutation method where relatively consistent in most cases and the same regions were identified regardless of pruning method and threshold. Unlike the simulation study, the number of SNPs selected increased as the pruning threshold increased after P-value pruning and tuning parameter selection was performed by the permutation method. Despite this, it reinforced my earlier conclusion that the permutation method showed the best performance for variable selection after pruning. Repeated CV also showed similar performance to the simulation with the number of SNPs selected decreasing after pruning by LD but increasing after pruning by either P-value or LD clumping. Previous studies that have applied penalised regression to GWAS datasets have commonly used P-value based pruning or CV for tuning parameter selection. This combination was shown to have the worst performance for variable selection as the model selected included a high number of false positives, which was also noticed in previous literature (11,149). Overall the results between the applications on this real dataset were consistent with the results from the simulation study which validates my conclusion for Aim 2.

11.1.2.2 The LASSO on the GRAPHIC study

In Chapter 4, I applied the Bonferroni correction and FDR methods to the GRAHPIC study dataset. The Bonferroni correction method only selected the top association with LDL-c, rs7412 on the APOE gene on chromosome 19 (BP = 45,412,079, p = 1.70×10^{-12}). The FDR method also selected rs7412 as well as a second SNP on the APOE gene rs4420638 (BP = 45,422,946, p = 1.58×10^{-7}). Although in close proximity to one another, both SNPs have independent effects.

As stated in a number of studies (11-16), the LASSO is unable to fit a whole genome-wide dataset therefore SNP pruning is required to reduce the dimensionality of the

dataset. Based on the conclusions from the single chromosome analysis and for Aim 2, both the BIC and permutation methods were used for tuning parameter selection and the dataset was pruned by LD clumping. The BIC, which is the more conservative of the two tuning parameter selection methods, selected a null model. The permutation method selected the top SNP, rs7412 with the λ estimate being close to also selecting rs4420638. Both the Manhattan plot (Figure 4.7) and Q-Q plot (Figure 4.6) showed that there were not many association in this dataset and it is therefore unsurprising that not many associations were discovered in this analysis. To current knowledge, this is the first study that had used LD clumping as a form of pre-screening in a GWA study and also the first time either the BIC or permutation method been used in human GWA study.

11.1.3 Aim 3: The LASSO in integrative analysis

In Chapter 10, I considered the role of penalised regression in the context of a multi-cohort study, in particular integrative analyses. I began by comparing a number of proposed methods in the current literature and then selected the meta-LASSO method proposed by Ma *et al*. (240) as the basis for a simulation study. This study was proposed in the context of gene expression data, which often has a small sample size and number of genes. I adapted this method for GWAS data and tested the method against the stacked LASSO and separate LASSO in terms of variable selection.

The results of the meta-LASSO showed that it was a relatively conservative method in terms of variable selection, while not many false positive SNPs were selected, not many true positive SNPs were selected either. This was the case for all three tuning parameter selection methods. The separate LASSO performed poorly as it lacked power. Overall the stacked LASSO showed superior performance using the permutation method selecting the highest sensitivity rate across the simulation and maintaining a high specificity rate at the same time (Table 9.8). The meta-LASSO outperformed the stacked LASSO using the BIC for tuning parameter selection. I also

proposed an alternative definition for variable selection; the replication measure. The sensitivity and specificity rates were high using the replication measure compared to the single selection measure.

Although the stacked LASSO showed the best overall performance in this simulation, there were two problems with this method; the first is that the method is unable to account for heterogeneity between dataset. The second is that the method still selected a low number of true positives using the permutation method. I therefore suggested an alternative penalised approach; the Integrative LASSO. This approach penalises the effect estimates in each dataset and also penalises the variance of the estimates across cohorts. I provide an algorithm that can be used to fit the IL. In section 10.4.1, I showed an example of how the IL method works and this example showed some potential promise for the use of the variance penalty. However as discussed in section 10.4.2 there are issues concerning convergence when the variance penalty is large. Two reasons were considered for the lack of convergence, the first was due to the opposing penalties that for some SNPs are penalising in two different directions leading to a “tug of war” type situation. The second is that for a large variance penalty there are many solutions and the algorithm is simply moving from one solution to the next across iterations. More work is required to determine the reason why the IL does not converge in these situations.

The simulation study showed that the IL method did not perform well compared to the competing methods that were used in Chapter 10. In fact the IL performed worst of the four methods in terms of variable selection. This is likely due to the nature of the penalty that penalizes each dataset separately, and by doing so the method lacks power compared to the stacked LASSO and meta-LASSO which both have a penalty penalizing all SNPs across all datasets.

11.1.4 Addressing the criticisms of the LASSO in GWAS

There are two main criticisms of the LASSO outlined by Zou *et al.* net (18).

1. When $P > N$, then the LASSO is only able to select at most N variables for a model.

2. In a group of highly correlated variables, the LASSO tends to select one variable without regard for which variable is selected.

The first criticism is of little concern in GWAS, study sample sizes tend to be large, typically containing thousands or hundreds of thousands of subjects, meanwhile the number of truly causal variants is very small in comparison to these sample sizes.

The second criticism can be argued as an advantage for the LASSO in GWAS and was illustrated in the application of the LASSO on chromosome 19 (see section 4.8.2). Both Bonferroni correction and FDR methods are unable to account for LD between SNPs and therefore selected a number of false positive SNPs that are correlated with the lead SNP in the region. My analyses show that the LASSO does select only one SNP out of a group of highly correlated SNPs, however it is not "without regard for which variable is selected" (18), rather the LASSO consistently selects the SNP with the highest correlation with the phenotype. All other SNPs that were correlated with this top SNP were removed from the model. This has also been seen in other studies that conclude the LASSO is able to handle LD in GWAS (24-26). The removal of these false positives from the model is particularly desirable for variable selection in GWA studies. When an association in a region happens to be perfectly correlated ($r^2 = 1$) with one or more SNPs, the assertion that the selected SNP is selected at random is somewhat true as discussed in section 4.8.2.3. However, regardless of whether one or all of these SNPs were selected, further investigation would be required to determine which SNP is the truly causal variant.

Hoffmann *et al*. also listed a number of criticisms of penalised regression methods in GWA studies (13) which included:

3. The inability to scale for very large GWAS datasets
4. Poor performance on simulated data
5. Finding too many 'hits' to be biologically plausible for a given GWAS sample size
6. They do not identify novel, well-supported associations that are not detectable by standard methods

Amongst others, all four criticisms reference a study by Hoggart *et al*. who uses penalised Bayesian approaches (250), which I did not consider in my thesis. Not all these criticisms may extend to the frequentist methods such as the LASSO, especially criticism 5 which only references the Hoggart study. The issue regarding the inability to scale for large GWAS is well established and was one of the aims of my thesis. As concluded in section 11.1.1, this can be overcome using LD clumping as a SNP pruning method.

Hoffmann *et al*. cites a study by Wu *et al*. regarding the poor performance in simulated data (145). In my thesis, I showed that performance in terms of variable selection is heavily influenced by the tuning parameter selection method used (see section 3.3.2). Wu *et al*. did not use any tuning parameter selection method to select a model; rather they pre-selected a model size therefore, the poor performance could easily be due to the author's choice of model size rather than the method itself. Throughout this thesis, I have demonstrated that, in both simulations and real data, the permutation method and to a lesser extent the BIC perform well for variable selection in GWA studies however, neither have previously used in a GWAS.

One of the conclusions made in section 11.1.2, was that the LASSO showed similar performance to FDR when both methods were applied to the GRAPHIC study. This conclusion is consistent with criticism 6 as the FDR can be considered as a standard method and both the FDR and LASSO selected similar models in terms of the genetic regions selected in both the single chromosome analysis and the genome-wide analysis after pruning.

11.2 Limitations and future work

This thesis limited the work to the application of the LASSO in GWAS data in terms of variable selection and therefore any conclusions made are only applicable in this context. There are plenty of alternative penalised regression approaches that could also be used. Given the criticism that the LASSO is not able to select novel variants that standard analyses are also unable to select (13), the elastic net could be a potential alternative to the LASSO. Waldmann *et al.* showed that the elastic net tends to select a large model than the LASSO in genetic studies (128), this would at least allow more variables to be selected and potentially some novel associations. This would require both tuning parameter selection methods and pruning methods to be optimised for the elastic net. Although this is straight forward using a grid search for CV and BIC, it is less clear how to optimise for the permutation method. My analyses showed that the permutation method performed particularly well relative to competing methods such as CV and BIC, however as the elastic net has two penalties it may be difficult to select the tuning parameter unless the α controlling the relative strength between the LASSO and ridge regression penalties is pre-specified. Other dual penalty methods such as the sparse group LASSO and the fused LASSO also use dual penalties and unlike the elastic net, do not have use an α to control the relative strength of penalties. Therefore some methodology does need to be developed for the permutation method for dual penalties.

In Chapter 3, I ran a simulation comparing a number of tuning parameter selection methods to determine the relative performance of each method in terms of variable selection. Stability selection (251) was not considered due to concerns with the subsampling used for this method may lack power to detect rare variants. Further work is required to determine whether this is the case or not. The analysis on the GRAPHIC study did not select rare variants either. Two approaches could be considered to increase the power to detect rare variants, the first is the use of grouping penalties (133-135), and the second by using the adaptive LASSO with weightings that favour selection of rarer SNPs over common SNPs.

Further applications of the LASSO are required on other real datasets to see whether the analyses draw the same conclusions in terms of tuning parameter selection and pruning methods. Given that the GRAPHIC study is a relatively small dataset, it would be of particular interest to test these methods hold up in a large GWAS study such as the UK Biobank data which consists of over 500,000 subjects and 73,000,000 variants. For such a large dataset, the pruning required to fit a LASSO model would need to be very heavy, in fact, even after quality control, pruning by just LD or LD clumping may not be sufficient to produce a small enough dataset to fit the LASSO. In this case, another pruning step may be required. The effect of imputation should also be considered as to how this may affect the LASSO model. In Chapter 4, I imputed missing genotypes with the median genotype value for that SNP. In Chapter 7, I used fastPhase to impute missing genotypes and it is not known if or how these imputation methods may affect the LASSO model.

There were main two limitations when applying the LASSO to the GRAPHIC study in both Chapters 4 and 8. The first was small sample size (N = 979) as only parental subjects were used which may have contributed towards low number of associations with LDL in the dataset (Figure 4.6 and Figure 4.7). The GRAPHIC study itself is a family based study consisting of over 2,000 subjects in 520 nuclear families, therefore in order to increase the power to select causal SNPs the offspring could also be included

for analysis. Papachristou *et al*. proposed a version of the LASSO for familial data (252) that could be used as a start point for this analysis. One of the limitations of my application of the LASSO to the GRAPHIC study is that I did not adjust for non-genetic factors such as age and sex or potential confounding factors. Ethnicity and differences between local populations were controlled for in this study by recruiting families from a local area, all with a white European ancestry. In my thesis, I applied the LASSO to a continuous phenotype, and binary phenotypes were not considered. Association testing on binary outcomes tends to be less powerful than continuous and requires testing to see if this is the case.

Added extensions to the Prune package are currently being considered. This includes pruning by stepwise models. Pruning by a stepwise model would allow pruning on a multivariate level, pruning by this method would require a specified cut-off point to select a certain number of SNPs for analysis. The concern with this method is that it may be computationally intensive in high-dimensional data and requires further work to see if this is the case. As discussed in section 7.2, pruning by effect estimates is not considered as it tends to prune common variants. If pruning by MAF is particularly desirable, this can be done with the current Prune package in one of two ways. The first is to use the `Fix` option to fix rare variants into the dataset. The second is to prune by LD clumping and set all the rare variants to have a P-value = 0. In terms of the simulation study on the effects of pruning, only the r^2 measure was used. This is the most commonly used measure for LD and pruning in general. The Prune package allows a number of other LD measures that could have been used (see section 6.6) and an extension to this work is to see how other measures such a D-prime and VIF perform.

The Integrative LASSO method had two main limitations. The first was the inability to converge for a large variation penalty and the second was the low power associated with the method, compared to competing methods. It is difficult to say why the IL does not converge ‘so suddenly’ without further testing, although I suggest two reasons why this may be the case in section 11.1.3. As the variance penalty is based on the fused

LASSO other algorithms that fit the fused LASSO such as augmented Lagrangian method (ALM) could be considered instead of the CDA (248). However, even if the algorithm could work for a large variance penalty, it is not likely to improve the performance compared to the stacked LASSO and meta-LASSO. One variation of the stacked LASSO that could be considered is the use of data-splitting approaches as recently described by Lu *et al.* (253). This is a similar approach to stability selection with a pooled dataset.

I suggested a slight variation on the IL that allows a different λ_d for each dataset. This gives the method greater flexibility and should show better performance compared to the separate LASSO. However, without a penalty on β across all datasets it is unlikely to outperform the meta-LASSO and stacked LASSO. Therefore, I consider two alternative approaches. For the first approach, I consider replacing the LASSO penalty on each dataset with one across all datasets therefore penalising across all datasets. The variance penalty is also slightly modified. Now I propose fitting the LASSO separately in each dataset using the same penalty that was applied across all datasets $\lambda = \lambda_1$ and the resulting β_{dj} estimates are then used in the variance penalty. Thus the proposed method would minimise the following:

$$\hat{\beta}(\lambda) = \arg\min_{\beta} \sum_{d=1}^{D} \frac{1}{2N_d} \sum_{i=1}^{N_d} \left(y_{di} - \sum_{j=1}^{P} x_{dij}\beta_{dj} \right)^2 + \lambda_1 \sum_{d=1}^{D} \sum_{j=1}^{P} |\beta_j| + \lambda_2 \sum_{d=1}^{D} \sum_{j=1}^{P} \left(\beta_{dj\lambda_1} - \frac{1}{D} \sum_{d=1}^{D} \beta_{dj\lambda_1} \right)^2 \quad (11.1)$$

If λ_1 is not scaled the same across all datasets, as it is in individual datasets, the penalty applied in separate datasets may need adjusting.

The second is the Data Shared LASSO discussed in section 9.2.3. This was not used as a comparative method as the algorithm the authors use is unviable in GWAS. Either very heavy pruning or an alternative algorithm that does not require such a high-dimensional dataset would be required for this method to be viable. Although it would need extensive testing, an algorithm similar to that proposed by Ma *et al*. (240) may work in this case.

These methods along with the meta-LASSO and stacked LASSO should be compared to traditional meta-analyses to gauge the difference between the methods for variable selection in multi-cohort studies across a much wider range of sample sizes, number of SNPs and different sources and levels of heterogeneity.

Appendix A: My LASSO function using the coordinate descent algorithm

Table A.1 My LASSO function using the coordinate descent algorithm

```
#-----------------------------------------------------------------------------------------------------------------#
# Set the seed
#-----------------------------------------------------------------------------------------------------------------#

set.seed(1)

#-----------------------------------------------------------------------------------------------------------------#
# Function to create a dataset of independent SNPs
#-----------------------------------------------------------------------------------------------------------------#

Make.Dataset <- function(NVAR, N, pExplained, MAF, MAF.min, MAF.max, Causal.MAF,
Causal.location, Error.Mean){

#-----------------------------------------------------------------------------------------------------------------#
# Default and error settings for Make.Dataset function
#-----------------------------------------------------------------------------------------------------------------#
  if( missing( NVAR ) )stop( "Must specify NVAR" )

  if( missing( N ))stop( "Must specify N" )

  if( missing( pExplained ) )stop( "Must specify pExplained" )

  if( missing( Causal.location ) )stop( "Must specify Causal.location" )

  if( missing( MAF.min ) ){

    message( paste( "MAF.min is missing: MAF.min set to 0.01" ) )

    MAF.min <- 0.01
  }

  if( missing( MAF.max )){

    message( paste( "MAF.max is missing: MAF.max set to 0.5" ) )

    MAF.max <- 0.5
  }
```

```
  if( length( Causal.MAF ) != length( Causal.location ) ) { stop( "Causal.MAF and
Causal.location are not of same length" ) }

  if( pExplained < 0 | pExplained > 1) { stop( "Percentage Explained must be between 0
and 1" ) }

  if( MAF.min < 0.001 | MAF.min > 0.999) { stop( "MAF.min must be between 0 and 1" )
}

  if( MAF.max < 0.001 | MAF.max > 0.999) { stop( "MAF.max must be between 0 and 1"
) }

  if( MAF.min >  MAF.max ) { stop( "MAF.max must be greater than MAF.min" ) }

  if( length(Causal.location ) < 1) { stop( "Must have at least 1 causal SNP" ) }

  if(length(Causal.location ) > NVAR) { stop( "Number of causal SNPs is greater than the
number of SNPs" ) }

  if( missing( Error.Mean ) ) { message(paste( "Error.Mean is missing: Error.Mean set to
0" ))

  Error.Mean <- c(0)
  }

#-----------------------------------------------------------------------------------------------------------#
# Create vectors for x, y, Beta and residual variance
#-----------------------------------------------------------------------------------------------------------#

  MAF.err <- matrix(0, nrow = 1, ncol = NVAR)

  Error.SD <- 1 - (length( Causal.location )*pExplained)

  x <- matrix(0, nrow = N, ncol = NVAR)

  y <- matrix(0, nrow = N, ncol = 1)

  Beta <- matrix(0, nrow = 1, ncol = length(Causal.location))
```

```
#-----------------------------------------------------------------------------------------------------------#
# Fix the MAF for causal SNPs and simulate a vector of minor allele frequencies
#-----------------------------------------------------------------------------------------------------------#
 if(missing(MAF)){

   message(paste("MAF is missing: MAF randomly generated"))

   MAF <- matrix( 0, nrow = 1, ncol = NVAR)

   maf <- runif(NVAR, min = MAF.min, max = MAF.max)

   for (j in 1:NVAR){

     MAF[,j] <- maf[j] + MAF.err[,j]
   }
 }

 for (j in 1:length(Causal.MAF)){
   MAF[ ,Causal.location[j]] <- Causal.MAF[j]
 }

#-----------------------------------------------------------------------------------------------------------#
# Simulate the genotype matrix X based on the MAF of each SNP
#-----------------------------------------------------------------------------------------------------------#

 for (j in 1:NVAR){

   x[, j] <- rbinom(N, 2, MAF[,j])

 }

#-----------------------------------------------------------------------------------------------------------#
# Simulate effect estimates for the causal SNPs
#-----------------------------------------------------------------------------------------------------------#

 for (i in 1:length(Causal.MAF)){
   Beta[i] <- (sqrt(pExplained/(2*maf[Causal.location[i]]*(1 - maf[Causal.location[i]]))))
 }
```

```
#-----------------------------------------------------------------------------------------------------------#
# Simulate the phenotype y
#-----------------------------------------------------------------------------------------------------------#

  y <- rnorm(N, mean = Error.Mean, sd = Error.SD)

  for (i in 1:length(Causal.MAF)){
    y <- y + Beta[i]*x[ ,Causal.location[i]]
  }

#-----------------------------------------------------------------------------------------------------------#
# Standardise X
#-----------------------------------------------------------------------------------------------------------#

  x <- scale(x)

#-----------------------------------------------------------------------------------------------------------#
# List of outputs
#-----------------------------------------------------------------------------------------------------------#

  return(list(X = x, Y = y, SNP.MAF = MAF, Causal.Beta = Beta))

}

#-----------------------------------------------------------------------------------------------------------#
# Function to fit the LASSO using coordinate descent
#-----------------------------------------------------------------------------------------------------------#

LASSO <- function(z, y, Convergence.Threshold, Iterations, Lambda){
#-----------------------------------------------------------------------------------------------------------#
# Default and error settings for the LASSO function
#-----------------------------------------------------------------------------------------------------------#

if( missing(z))stop( "Must specify z" )

  if( missing(y))stop( "Must specify y" )

  if( missing(Lambda))stop( "Must specify Lambda" )

  if( missing(Convergence.Threshold) ){

    message(paste( "Convergence.Threshold is missing: Convergence.Threshold set to
0.0001" ))

    Convergence.Threshold <- 0.0001
}
```

```
  if( missing(Iterations) ){

    message( paste( "Iterations is missing: Iterations set to 10000" ) )

    Iterations <- 10000}

#-----------------------------------------------------------------------------------------------------------#
# Simulate results matrices
#-----------------------------------------------------------------------------------------------------------#

  N <- nrow(z)

  results <- matrix(0, nrow = length(Lambda), ncol = ncol(z))

  a0 <- matrix(0, nrow = length(Lambda), ncol = 1)

  rss <- matrix(0, nrow = length(Lambda), ncol = 1)

  beta.hat <- as.numeric(matrix(0, nrow = 1, ncol = ncol(z)))

  old.beta <- as.numeric(matrix(0, nrow = 1, ncol = ncol(z)))

#-----------------------------------------------------------------------------------------------------------#
# calculate SXX
#-----------------------------------------------------------------------------------------------------------#

  sxx <-matrix(0, nrow = 1,ncol = ncol(z))
  for(i in 1:ncol(X)){sxx[i] <- sum(z[,i]^2)}

#-----------------------------------------------------------------------------------------------------------#
# Coordinate descent algorithm
#-----------------------------------------------------------------------------------------------------------#


#-----------------------------------------------------------------------------------------------------------#
# Loop over Lambda
#-----------------------------------------------------------------------------------------------------------#

  for (k in 1:length(Lambda)){
```

```
#-----------------------------------------------------------------------------------------------------------------#
# Loop over number of iterations
#-----------------------------------------------------------------------------------------------------------------#
   for (i in 1:Iterations){

#-----------------------------------------------------------------------------------------------------------------#
# Let the beta estimate from the previous iteration be called Oldbeta
#-----------------------------------------------------------------------------------------------------------------#

     old.beta <- beta.hat

#-----------------------------------------------------------------------------------------------------------------#
# Loop over number of SNPs
#-----------------------------------------------------------------------------------------------------------------#

     for(j in 1:ncol(z)){

#-----------------------------------------------------------------------------------------------------------------#
# In the case of a monomorphic SNP, do not change Beta
#-----------------------------------------------------------------------------------------------------------------#


       if (sxx[j]==0){

         new.beta <- beta.hat[j]

       } else {
#-----------------------------------------------------------------------------------------------------------------#
# Calculate r
#-----------------------------------------------------------------------------------------------------------------#


         beta.hat <- as.vector(beta.hat)

         r <- sum((y - a0[k] - z%*%beta.hat)*z[,j])

#-----------------------------------------------------------------------------------------------------------------#
# Calculate left and right derivatives
#-----------------------------------------------------------------------------------------------------------------#

         if(beta.hat[j]==0){

           right.der <- - r + (N - 1)*Lambda[k]

           left.der <- - r - (N - 1)*Lambda[k]
         }
```

```
        if(beta.hat[j] > 0){

          right.der <- - r + (N - 1)*Lambda[k]

          left.der <- - r + (N - 1)*Lambda[k]
        }

        if(beta.hat[j] < 0){

          right.der <- - r - (N - 1)*Lambda[k]

          left.der <- - r - (N - 1)*Lambda[k]
        }

#-------------------------------------------------------------------------------------------------------------#
# Calculate the new beta estimate
#-------------------------------------------------------------------------------------------------------------#


        if (right.der*left.der > 0){

          if (beta.hat[j]==0 & r < 0){

            new.beta <- beta.hat[j] - (left.der/sxx[j])

          } else {

            new.beta <- beta.hat[j] - (right.der/sxx[j])

          }

          if (new.beta*beta.hat[j] < 0){new.beta <- 0}

          beta.hat[j] <- new.beta

#-------------------------------------------------------------------------------------------------------------#
# Estimate the intercept Beta0
#-------------------------------------------------------------------------------------------------------------#

          a0[k] <- mean(y - z%*%beta.hat)

        }
      }
    }
```

```
#-----------------------------------------------------------------------------------------------------------#
# Check if convergence criteria has been met, if so stop. If not check the maximum
# number of iterations has not been met
#-----------------------------------------------------------------------------------------------------------#
    if(sum(abs(old.beta - beta.hat)) < Convergence.Threshold){break}

    if(i==Iterations){stop("LASSO failed to converge")}

  }

#-----------------------------------------------------------------------------------------------------------#
# Record the beta estimates for the k'th lambda and residual sum of squares
#-----------------------------------------------------------------------------------------------------------#
  results[k, ] <- beta.hat

  rss[k,] <- sum((y - a0[k] - z%*%beta.hat)^2)+((N - 1)*Lambda[k]*sum(abs(beta.hat)))

 }
#-----------------------------------------------------------------------------------------------------------#
# List of outputs
#-----------------------------------------------------------------------------------------------------------#


 return(list(Beta = results, RSS = rss, b0 = a0))
}

#-----------------------------------------------------------------------------------------------------------#
# Command to produce a dataset
#-----------------------------------------------------------------------------------------------------------#


data <- Make.Dataset(NVAR = 25, N = 50, pExplained = 0.01, MAF.min = 0.01, MAF.max
= 0.5, Causal.MAF = c(0.02, 0.2), Causal.location = c(15, 25))

#-----------------------------------------------------------------------------------------------------------#
# Output the genotype matrix and vector of phenotypes
#-----------------------------------------------------------------------------------------------------------#

X <- data$X
Y <- data$Y
```

```
#-----------------------------------------------------------------------------------------------------------#
# Run LASSO function, input genotype matrix, pehotype vector, convergence
# threshold, number of iterations and a sequence of lambda estimates
#-----------------------------------------------------------------------------------------------------------#

lasso <- LASSO(z = X, y = Y, Convergence.Threshold = 0.0001, Iterations = 10000,
Lambda = seq(from = 0, to = 0.2, by = 0.01))

#-----------------------------------------------------------------------------------------------------------#
# Output a matrix of beta estimates for varying lambda estimates
#-----------------------------------------------------------------------------------------------------------#

betas <- lasso$Beta
```

Appendix B: Summary of studies that have performed GWAS on the LDL

Table B.0.1 Summary of studies that have performed GWAS on the Low-density Lipoprotein phenotype

Author	Pub. Date	Methods used	Sample Size	Ancestry	No. of SNPs	Chr.	Gene	SNP	Position	P-value
Asselbergs (162)	Nov-12	Meta-analysis. Corrected for population stratification, age & lipid medication.	66,420	European	49,227	1	PCSK9	rs499883	55,519,174	1.92E-09
						2	APOB	rs693	21,232,195	1.81E-12
						2	ABCG5, ABCG8	rs4953023	44,074,000	1.15E-05
						6	LPA	rs3798220	160,961,137	5.55E-06
						11	SPT2	rs3781799	19,208,319	3.16E-05
						19	LDLR	rs5930	11,224,265	7.84E-09
						19	APOE	rs769450	45,410,444	2.62E-

										10
Aulchenko (163)	Dec-08	Meta-analysis	17,797	European (Scandanvian)	~600,000	1	DOCK7	rs10889353	63,118,196	8.00E-06
						1	CELSR2, SORT1	rs646776	109,818,530	8.00E-23
						2	APOB	rs693	21,232,195	4.00E-17
						2	ABCG5, ABCG8	rs6756629	44,065,090	3.00E-10
						5	HMGCR	rs3846662	74,651,084	2.00E-11
						7	DNAH11	rs12670798	21,607,352	6.00E-09
						8	TRIB1	rs6987702	126,504,726	3.00E-06
						11	FADS2, FADS3	rs174570	61,597,212	4.00E-13
						11	APOA1	rs12272004	116,603,724	5.00E-13
						19	LDLR	rs2228671	11,210,912	4.00E-14
						19	NCAN	rs2304130	19,789,528	3.00E-06
						19	APOE	rs157580	45,395,266	2.00E-19
Chasman (164)	Jun-08	Additive regression	6,382 (all	American	341,518	1	CELSR2, SORT1	rs646776	109,818,530	4.90E-19

		model adjusting for age, BMI, smoking status, menopausal status & HRT status	female cohort)			2	APOB	rs506585	21,397,182	9.30E-09
						19	LDLR	rs6511720	11,202,306	5.20E-15
						19	APOE	rs4803750	45,247,627	3.60E-14
Chasman (186)	Jan-12	Additive regression model.	6,989	European	814,418	1	PCSK9	rs11591147	55,505,647	4.70E-09
						19	APOE	rs7412	45,412,079	1.60E-47
Kathiresan (166)	Jan-08	Meta-analysis	2,758	European	389,878	1	PCSK9	rs11591147	55,505,647	2.00E-44
						1	CELSR2, SORT1	rs646776	109,818,530	3.00E-29
						2	APOB	rs693	21,232,195	1.00E-21
						5	HMGCR	rs12654264	74,648,603	1.00E-20
						19	LDLR	rs6511720	11,202,306	2.00E-51
						19	CILP2	rs16996148	19,658,472	3.00E-08
						19	APOE	rs4420638	45,422,946	1.00E-60
Kathiresan (167)	Dec-08	Meta-analysis	19,840	European	~2.6 million	1	PCSK9	rs11206510	55,496,039	4.00E-08

					(Imputed)	1	CELSR2, SORT1	rs12740374	109,817,590	2.00E-42
						2	APOB	rs515135	21,286,057	5.00E-29
						2	ABCG5, ABCG8	rs6544713	44,073,881	2.00E-20
						5	HMGCR	rs3846663	74,655,726	8.00E-12
						5	HAVCR1	rs1501908	156,398,169	1.00E-11
						12	HNF1A	rs2650000	121,388,962	2.00E-08
						19	LDLR	rs6511720	11,202,306	2.00E-26
						19	CILP2	rs10401969	19,407,718	2.00E-08
						19	APOE	rs4420638	45,422,946	4.00E-27
						20	MAFB	rs6102059	39,228,784	4.00E-09
Kim (254)	Oct-11	Meta-analysis	23,921	Korean	-	1	CELSR2, SORT1	rs599839	109,822,166	1.84E-19
			25,098			5	HMGCR	rs12654264	74,648,603	1.21E-20
			25,112			9	ABO	rs651007	136,153,875	6.00E-09
			25,120			19	LDLR	rs2738446	11,227,326	2.02E-12

Lettre (169)	Feb-11	Linear regression model	8,090	African American	909,622	1	PCSK9	rs10493178	55,597,067	4.76E-12
						1	DOCK7	rs10889335	62,960,101	1.26E-04
						1	DOCK7	rs10889353	63,118,196	4.00E-03
						1	CELSR2, SORT1	rs12740374	109,817,590	1.36E-16
						2	APOB	rs562338	21,288,321	3.16E-07
						2	APOB	rs503662	21,414,142	2.56E-09
						19	LDLR	rs6511720	11,202,306	7.26E-08
						19	APOE	rs1160985	45,403,412	7.26E-21
Middelberg (170)	Sep-11	Multivariate analysis	11,693	Australian	-	1	CELSR2	rs660240	109,817,838	7.00E-09
						2	APOB	rs10199768	21,244,000	2.30E-08
						19	APOE	rs2075650	45,395,619	5.70E-10
Musunuru (171)	May-12	Additive regression model.	44,957	European	49,320	1	PCSK9	rs11591147	55,505,647	1.75E-28
				African American		1	PCSK9	rs11806638	55,518,160	5.06E-11
				African American &		1	CELSR2, SORT1	rs7528419	109,817,192	1.70E-51

	European					
	African American & European	1	CELSR2, SORT1	rs12740374	109,817,590	2.91E-51
	European	2	APOB	rs934197	21,267,461	2.89E-34
	African American & European	2	APOB	rs562338	21,288,321	7.61E-34
	European	2	ABCG5, ABCG8	rs4953023	44,074,000	2.42E-08
	European	5	HMGCR	rs12916	74,692,295	5.51E-13
	European	6	LPA	rs10455872	161,010,118	2.82E-12
	European	7	NPC1L1	rs17725246	44,581,986	5.57E-07
	European	8	TRIB1	rs6982636	126,479,315	4.30E-09
	European	16	HPR	rs2000999	72,108,093	2.40E-08
	African American	19	ICAM1	rs5030359	10,388,462	1.25E-08
	African American & European	19	LDLR	rs6511720	11,202,306	1.39E-49
	African	19	APOE	rs389261	45,420,343	1.11E-

				American						14
				European		19	APOE	rs12721046	45,421,254	2.25E-29
Rasmussen-Torvik (172)	Oct-12	-	1249	African American	910,341	19	APOE	rs7412	45,412,079	2.00E-09
Roslin (173)	Dec-09	Growth Curve Model adjusted for sex, baseline age, diabetes, BMI & smoking (per day) as time-varying covariates	1,659	United States of America	348,053	1	CELSR2, SORT1	rs599839	109,822,166	3.04E-10
Sabatti (174)	Dec-08	OLS Regression adjusted for various traits	4,763	Finland	329, 091	1	CELSR2, SORT1	rs646776	109,818,530	2.00E-12
						2	APOB	rs693	21,232,195	3.00E-11
						11	FADS1	rs174546	61,569,830	1.00E-07
						19	LDLR	rs11668477	11,195,030	2.00E-07
						19	APOE	rs157580	45,395,266	5.00E-08
Saleheen (175)	Jun-10	Linear regression	5,576	Pakistani	31,883	1	CELSR2, SORT1	rs646776	109,818,530	7.19E-10

		model								
Sandhu (176)	Feb-08	Meta-analysis. Corrected for population stratification	11,685	UK	461,986	1	CELSR2, SORT1	rs599839	109,822,166	1.00E-33
						2	APOB	rs562338	21,288,321	1.00E-09
						19	APOE	rs4420638	45,422,946	1.00E-20
Shen (255)	Nov-10	Multi-level model. Corrected for sex, age and age^2	841	Amish	369,241	2	APOB	rs4971516	20,903,015	2.00E-52
Smith (178)	Sep-10	Linear mixed mode adjusting for age & sex with random slope and random intercept	525	European	545,821	5	KIF4B, SGCD	rs10044666	155,128,570	4.76E-07
						6	-	rs7738656	121,714,848	2.56E-07
						19	APOE	rs7412	45,412,079	1.66E-08
						21	MRPS6, KCNE2	rs8131349	35,571,906	1.46E-08
Talmund (179)	Nov-09	Additive regression model adjusting for age and sex	5,059	UK	48,032	1	PCSK9	rs11591147	55,505,647	9.28E□-12
						1	CELSR2, SORT1	rs4970834	109,814,880	5.18E□-09
						1	CELSR2, SORT1	rs12740374	109,817,590	1.82E□-09

	1	CELSR2,SORT1	rs629301	109,818,306	6.50E⍰-09
	2	APOB	rs693	21,232,195	3.86E⍰-08
	2	APOB	rs934197	21,267,461	3.43E⍰-08
	2	APOB	rs562338	21,288,321	1.21E-11
	2	ABCG5, ABCG8	rs4299376	44,072,576	8.70E⍰-10
	5	HMGCR	rs12916	74,692,295	6.66E⍰-09
	5	HMGCR	rs3804231	74,696,779	5.15E-06
	11	APOA5	rs2072560	116,661,826	2.36E-07
	16	CETP	rs17231506	56,994,528	5.02E-07
	19	LDLR	rs1529729	11,163,562	6.71E-06
	19	LDLR	rs17248720	11,198,187	7.86E-25
	19	LDLR	rs8110695	11,206,530	1.12E-08
	19	LDLR	rs2228671	11,210,912	6.52E⍰-10
	19	BCL3	rs1531517	45,242,173	6.26E-08

Teslovich (180)	Feb-11	Meta-analysis	95,454	European	~2.6 million (Imputed)	19	BCL3/PVRL2	rs10402271	45,329,214	2.06E-12
						19	APOE	rs519113	45,376,284	9.37E-07
						19	APOE	rs6859	45,382,034	3.53E-08
						19	APOE	rs283813	45,389,174	2.51E□-06
						19	APOE	rs2075650	45,395,619	1.14E□-14
						19	APOE	rs12721046	45,421,254	7.58E-14
						19	APOE	rs12721109	45,447,221	5.06E□-14
						1	TMEM57,LDLRAP1	rs12027135	25,775,733	1.00E-10
						1	PCSK9	rs2479409	55,504,650	2.00E-28
						1	DOCK7	rs2131925	63,025,942	3.00E-18
						1	CELSR2, SORT1	rs629301	109,818,306	1.00E-170
						1	MOSC1	rs2642442	220,973,563	6.00E-11
						1	TOMM20	rs514230	234,858,597	9.00E-12
						2	APOB	rs1367117	21,263,900	4.00E-114

2	ABCG5,ABCG8	rs4299376	44,072,576	2.00E-47
5	HMGCR	rs12916	74,656,539	5.00E-45
5	HAVCR1	rs6882076	156,390,297	2.00E-22
6	IDOL	rs3757354	16,127,407	1.00E-11
6	HFE,HIST1H4C	rs1800562	26,093,141	6.00E-10
6	HLA	rs3177928	32,412,435	2.00E-15
6	FRK	rs9488822	116,312,893	3.00E-09
6	LPA	rs1564348	160,578,860	2.00E-17
7	DNAH11	rs12670798	21,607,352	7.00E-10
7	NPC1L1	rs2072183	44,579,180	4.00E-11
8	PPP1R3B	rs9987289	9,183,358	7.00E-15
8	CYP7A1	rs2081687	59,388,565	4.00E-09
8	TRIB1	rs2954029	126,490,972	3.00E-29
8	PLEC1	rs11136341	145,043,543	4.00E-13

	9	ABO	rs635634	136,155,000	8.00E-22
	10	GPAM	rs2255141	113,933,886	2.00E-09
	11	FADS1	rs174546	61,569,830	1.00E-21
	11	APOA1	rs964184	116,648,917	1.00E-26
	11	ST3GAL4	rs11220462	126,243,952	1.00E-15
	12	BRAP	rs11065987	112,072,424	2.00E-09
	12	HNF1A	rs1169288	121,416,650	1.00E-15
	14	CBLN3,KIAA0323	rs8017377	24,883,887	4.00E-11
	16	CETP	rs3764261	56,993,324	9.00E-13
	16	HPR	rs2000999	72,108,093	2.00E-22
	17	OSBPL7	rs7206971	45,425,115	4.00E-09
	19	LDLR	rs6511720	11,202,306	4.00E-117
	19	CILP2	rs10401969	19,407,718	7.00E-22
	19	APOE	rs4420638	45,422,946	9.00E-147

						20	MAFB	rs2902940	39,091,487	1.00E-08
						20	TOP1	rs6029526	39,672,618	3.00E-19
Trompet (181)	Oct-11	Linear regression model adjusted for Age, Sex & Country	5,244	European	557,192	1	CELSR5	rs602633	109,821,511	5.00E-08
						5	HMGCR	rs258494	75,038,718	1.30E-09
						11	FADS2	rs174541	61,565,908	1.10E-08
						19	LDLR	rs6511720	11,202,306	5.20E-15
						19	APOE	rs445925	45,415,640	2.80E-30
Waterworth (182)	Sep-10	Meta-analysis	17,243	European	2,155,369 (Imputed)	1	PCSK9	rs11206510	55,496,039	1.00E-10
						1	CELSR2	rs660240	109,817,838	1.00E-26
						2	APOB	rs515135	21,286,057	2.00E-20
						5	HMGCR	rs12916	74,656,539	1.00E-11
						6	MYLIP,GMPR	rs2142672	16,197,194	2.00E-08
						8	PPP1R3B	rs2126259	9,185,146	7.00E-12
						8	TRIB1	rs2954021	126,482,077	1.00E-07

						11	APOA1	rs1558861	116,607,437	2.00E-06
						19	LDLR	rs2738459	11,238,473	7.00E-06
						19	CILP2	rs10401969	19,407,718	1.00E-11
						19	APOE	rs4420638	45,422,946	2.00E-40
Willer (183)	Apr-12	-	8,589	-	~2,261,000 (Imputed)	1	PCSK9	rs11206510	55,496,039	4.00E-11
						1	CELSR2, SORT1	rs599839	109,822,166	6.00E-33
						2	APOB	rs562338	21,288,321	6.00E-22
						6	COL11A2	rs2254287	33,143,948	5.00E-08
						19	LDLR	rs6511720	11,202,306	4.00E-26
						19	CILP2	rs16996148	19,658,472	3.00E-09
						19	APOE	rs4420638	45,422,946	3.00E-43
Wu (184)	Mar-13	Trans-ethic Meta-analysis	20,278	African American, East Asian and White European	15,000+ SNP's over 58 different Locus	1	PCSK9	rs11591147	55,505,647	4.60E-32
						1	CELSR2, SORT1	rs12740374	109,817,590	7.60E-37
						2	-	rs7575840	21,273,490	4.20E-19

	2	ABCG5, ABCG8	rs76866386	44,075,482	4.00E-11
	5	-	rs7722186	74,576,285	4.00E-08
	5	HAVCR1	rs9715911	156,394,441	5.70E-06
	8	TRIB1	rs4870941	126,498,828	4.80E-06
	9	ABO	rs2519093	136,141,870	2.20E-13
	16	HPR	rs72626182	72,078,990	2.00E-06
	19	LDLR	rs73015011	11,189,764	7.40E-38
	19	APOE	rs7412	45,412,079	5.90E-209
	20	-	rs1883511	39,641,517	6.50E-06

Appendix C: Summary of SNPs selected using the LASSO on chromosome 19 of the GRAPHIC study using repeated Cross-validation

Table C.1 SNPs selected by the LASSO on chromosome 19 of the GRAPHIC study dataset using 100 repeats of 10-fold Cross-validation for tuning parameter selection

SNP	Base position	Beta	S.E.	MAF	P-value	Q-value	LASSO Beta
rs7412	45412079	-16.28	2.28	0.0861	1.70E-12	2.09E-08	-9.58638
rs4420638	45422946	8.01	1.52	0.2106	1.58E-07	0.000968	2.13897
rs2075650	45395619	8.45	1.71	0.1573	9.66E-07	0.003951	0.45674
rs10402182	37160529	6.29	1.36	0.3013	4.53E-06	0.007943	2.31366
rs17272386	37180297	6.29	1.36	0.3013	4.53E-06	0.007943	4.20E-15
rs1525133	37199250	6.29	1.36	0.3013	4.53E-06	0.007943	6.00E-16
rs17001002	10948031	-6.97	1.59	0.1839	1.25E-05	0.014524	-2.91778
rs10402271	45329214	5.44	1.36	0.3121	6.78E-05	0.055104	0.17373
rs10853810	49969085	-4.92	1.3	0.3683	0.000166	0.114392	-1.31339
rs12975624	40335037	-4.51	1.24	0.4847	0.000304	0.178521	-0.86207
rs7253937	20474655	-6.15	1.75	0.167	0.000459	0.230804	-1.21638
rs887030	2528705	5.02	1.44	0.284	0.000501	0.242565	0.48490
rs10896	19287928	4.5	1.29	0.3698	0.000531	0.250326	0.89130
rs1860328	42136626	-4.93	1.46	0.2579	0.000773	0.310623	-0.60908
rs846866	45134110	6.06	1.82	0.142	0.000871	0.330124	1.27082

rs6509544	52003331	-5.6	1.69	0.1741	0.000946	0.343679	-1.15941
rs7253451	40259342	4.55	1.38	0.2845	0.001029	0.357392	0.58311
rs10407413	5947871	7.04	2.15	0.0947	0.001116	0.370545	1.30276
rs260415	58715944	-4.38	1.37	0.324	0.001414	0.407971	-1.18267
rs2607272	13912062	4.26	1.36	0.3176	0.001743	0.439339	0.17139
rs2233152	41281016	-5.4	1.76	0.1494	0.002202	0.471841	-0.34642
rs2287692	41289756	-5.4	1.76	0.1494	0.002202	0.471841	-5.10E-14
rs2228671	11210912	-5.56	1.81	0.1361	0.002234	0.473747	-0.23144
rs10425830	6352936	9.08	2.97	0.0473	0.002291	0.477047	1.02056
rs10854133	14955029	-4.3	1.41	0.266	0.002415	0.483828	-0.06386
rs734570	49606449	-5.22	1.73	0.1726	0.002651	0.495412	-0.18926
rs8110944	51909821	8.33	2.81	0.0538	0.003069	0.512514	0.00226
rs3826838	49183284	-5.48	1.85	0.1369	0.00312	0.514355	-0.38901
rs10417957	4647231	4.46	1.51	0.2265	0.003177	0.516357	0.05071
rs11084440	56782965	3.91	1.32	0.4573	0.003187	0.516703	0.11474
rs806711	5618654	-7.03	2.4	0.07	0.003531	0.527629	-1.06273
rs4251950	44153255	13.58	4.69	0.0202	0.003848	0.536283	0.25613
rs17833533	33630693	5.3	1.84	0.1371	0.004038	0.540934	0.07895
rs2278434	38713568	-5.1	1.78	0.1534	0.004254	0.545801	-0.05245
rs12986307	57519466	-4.28	1.51	0.2244	0.004725	0.555109	-0.31246
rs11665818	39768216	-4.42	1.57	0.2045	0.005005	0.559934	-0.05974
rs12327843	18004912	3.96	1.41	0.2572	0.005124	0.561847	0.05206
rs4807284	2493811	5.54	1.98	0.119	0.005197	0.562984	0.01917
rs10500293	46431638	3.52	1.27	0.4433	0.005566	0.568335	0.00162
rs11083567	41318306	-4.34	1.57	0.1881	0.005841	0.57194	-0.19856
rs3745245	10676423	4.13	1.5	0.2352	0.006174	0.575927	0.11756
rs3218222	10676681	4.13	1.5	0.2352	0.006174	0.575927	4.60E-15
rs28433973	8660890	-4.38	1.61	0.195	0.006691	0.589904	-0.16029

rs10426401	45147719	3.92	1.46	0.2406	0.00752	0.610096	0.09552
rs1466448	8289519	-3.99	1.57	0.1967	0.01135	0.67282	-0.01455

Appendix D: Coefficient path plots all 50 simulated SNPs using the Integrative LASSO when $\lambda_1 = 0$

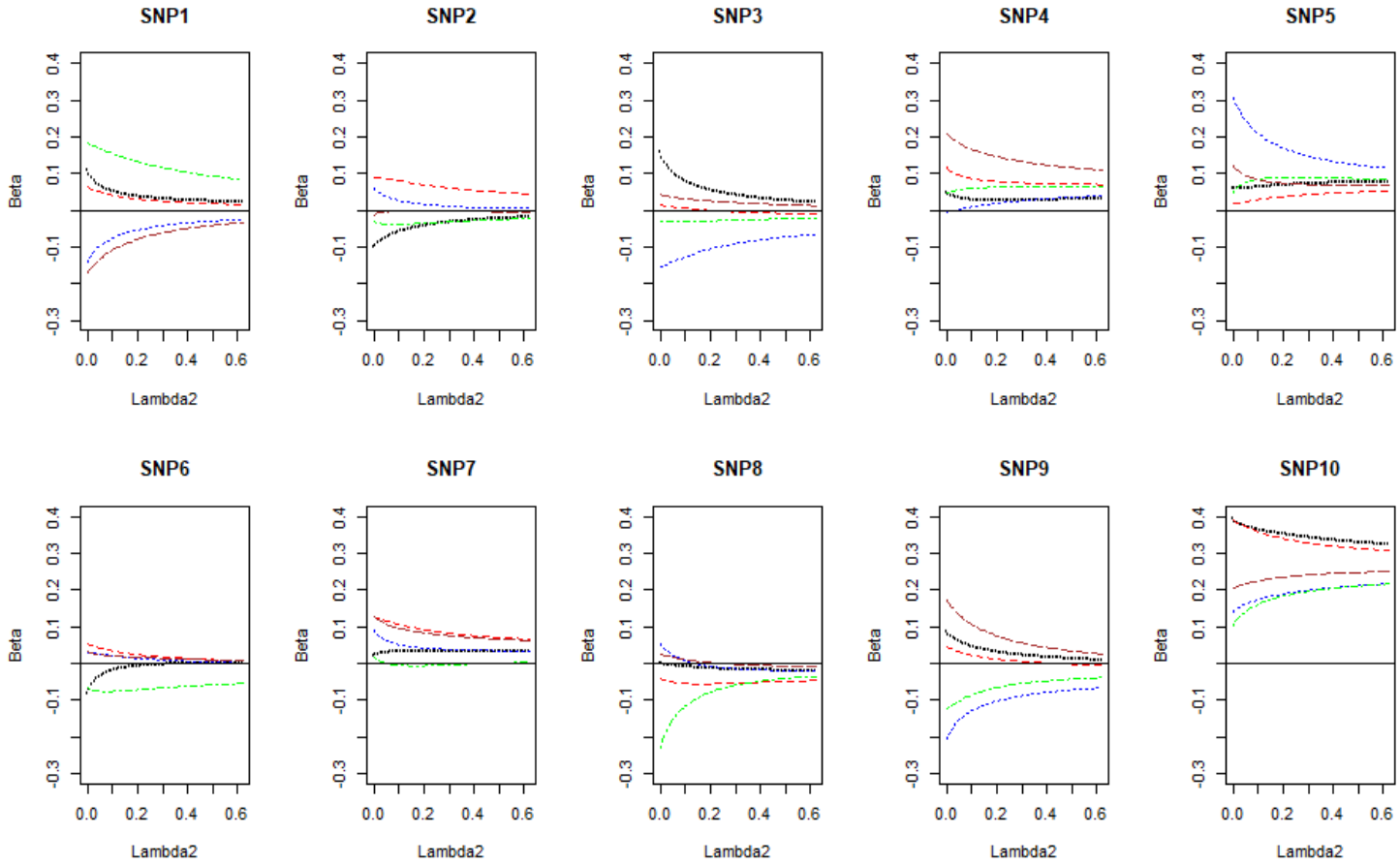


Figure D.0.1 Coefficient path plots for SNP1 to SNP10 using the Integrative LASSO when $\lambda_1 = 0$. Each line represents the SNP from a dataset and the path shows the βcoefficient on the y-axis as the λ_2 penalty increases on the bottom x-axis.

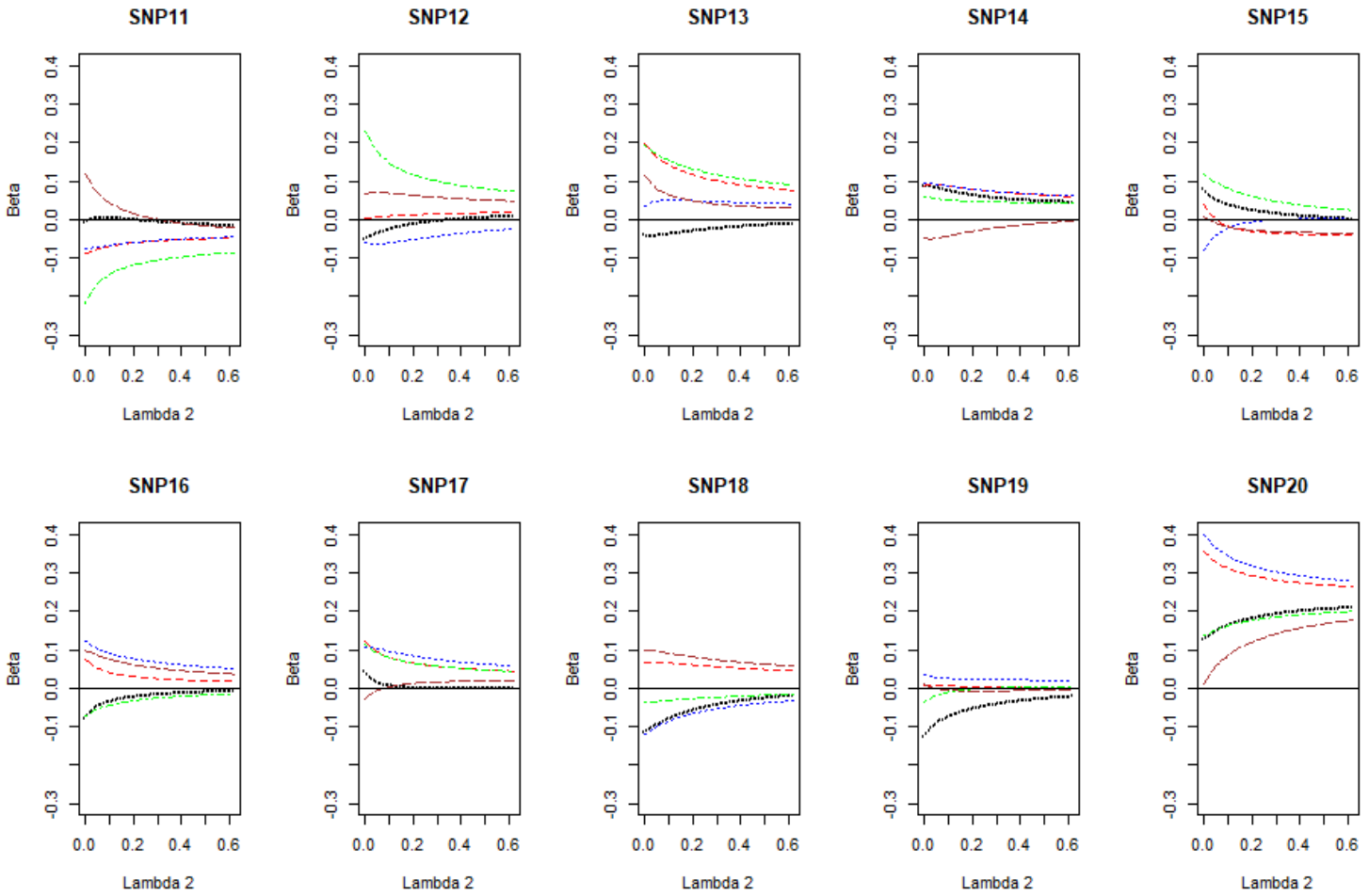


Figure D.0.2 Coefficient path plots for SNP11 to SNP20 using the Integrative LASSO when $\lambda_1 = 0$. Each line represents the SNP from a dataset and the path shows the βcoefficient on the y-axis as the λ_2 penalty increases on the bottom x-axis.

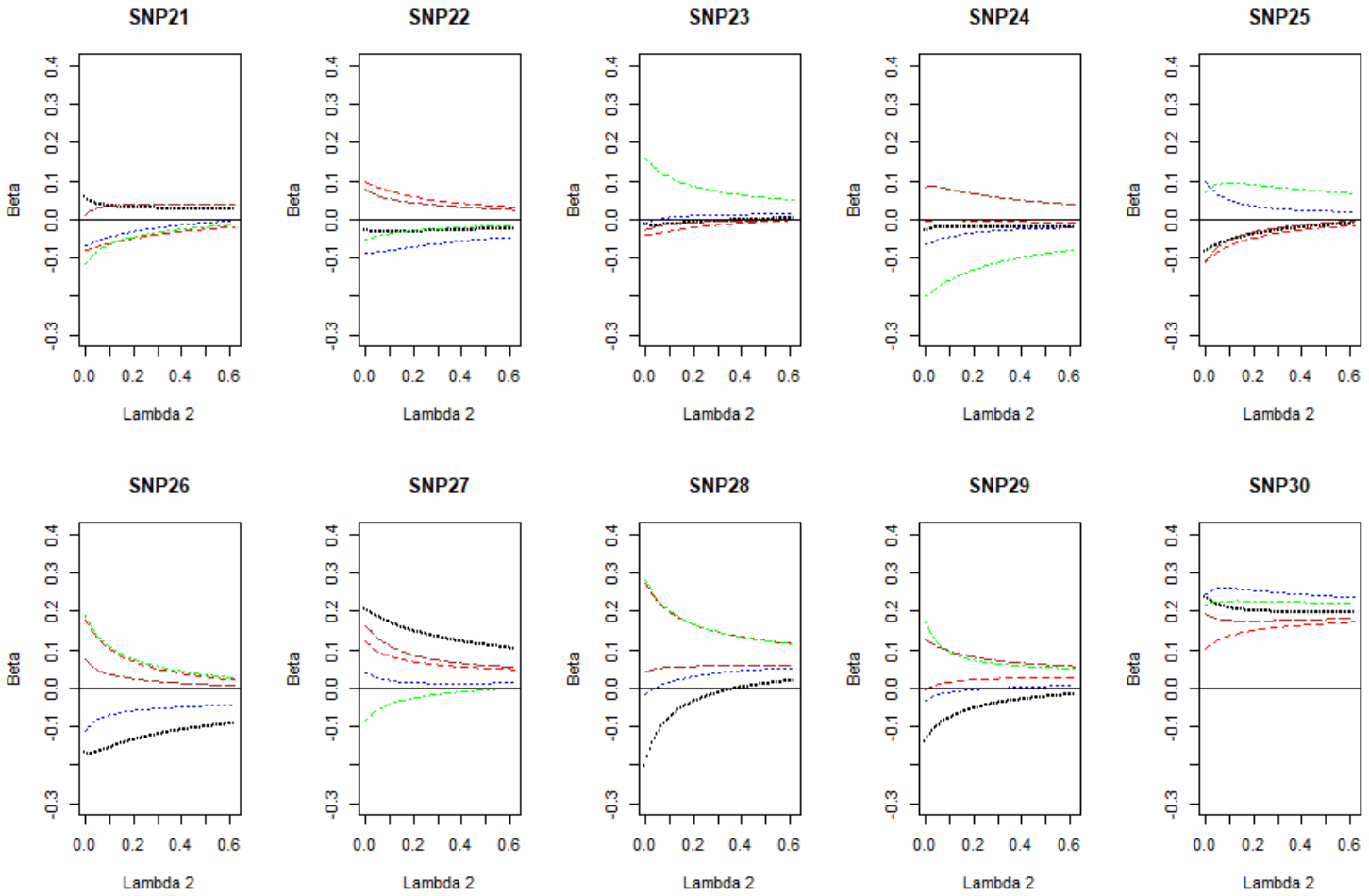


Figure D.0.3 Coefficient path plots for SNP21 to SNP30 using the Integrative LASSO when $\lambda_1 = 0$. Each line represents the SNP from a dataset and the path shows the βcoefficient on the y-axis as the λ_2 penalty increases on the bottom x-axis.

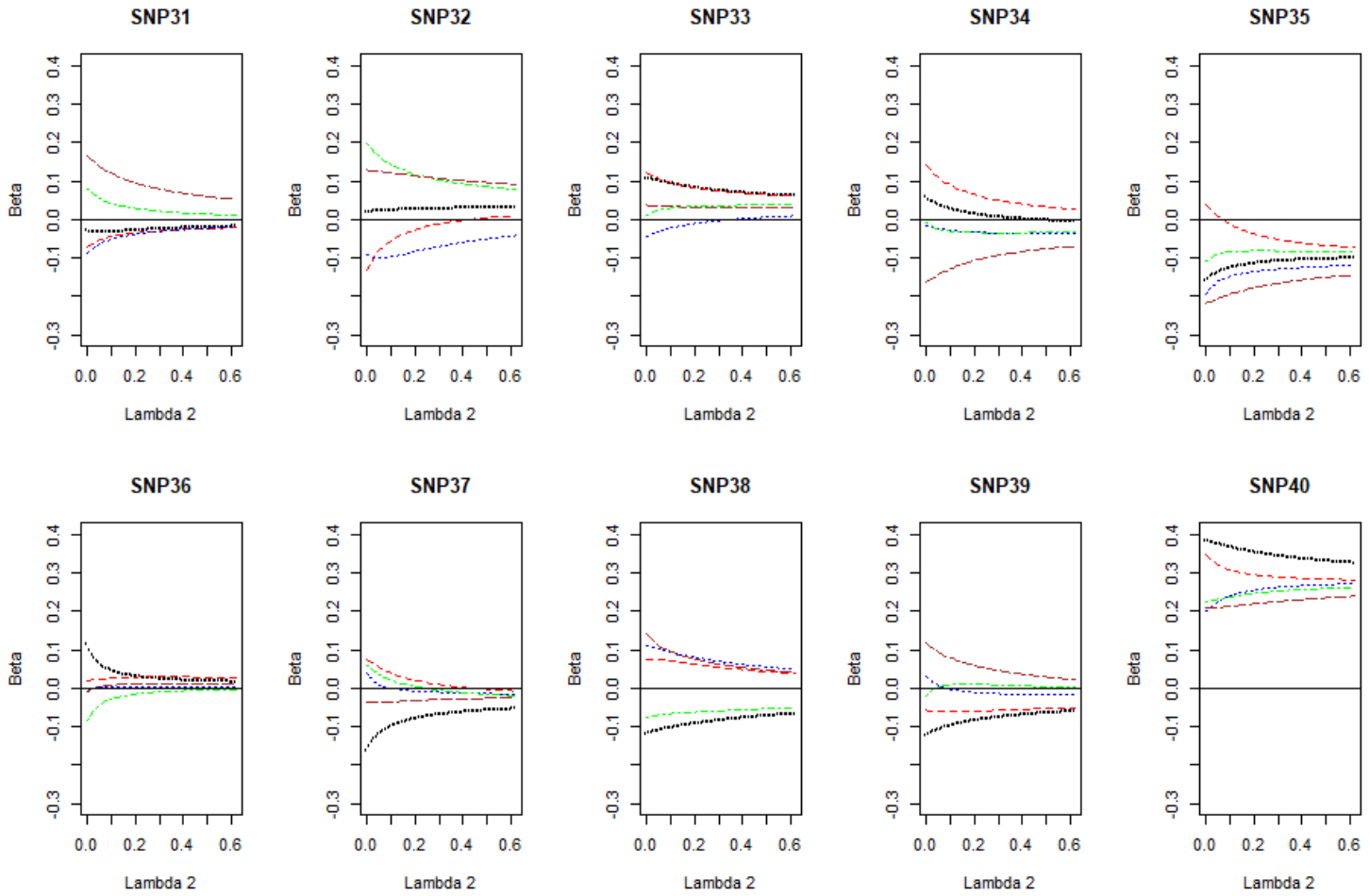


Figure D.0.4 Coefficient path plots for SNP31 to SNP40 using the Integrative LASSO when $\lambda_1 = 0$. Each line represents the SNP from a dataset and the path shows the βcoefficient on the y-axis as the λ_2 penalty increases on the bottom x-axis.

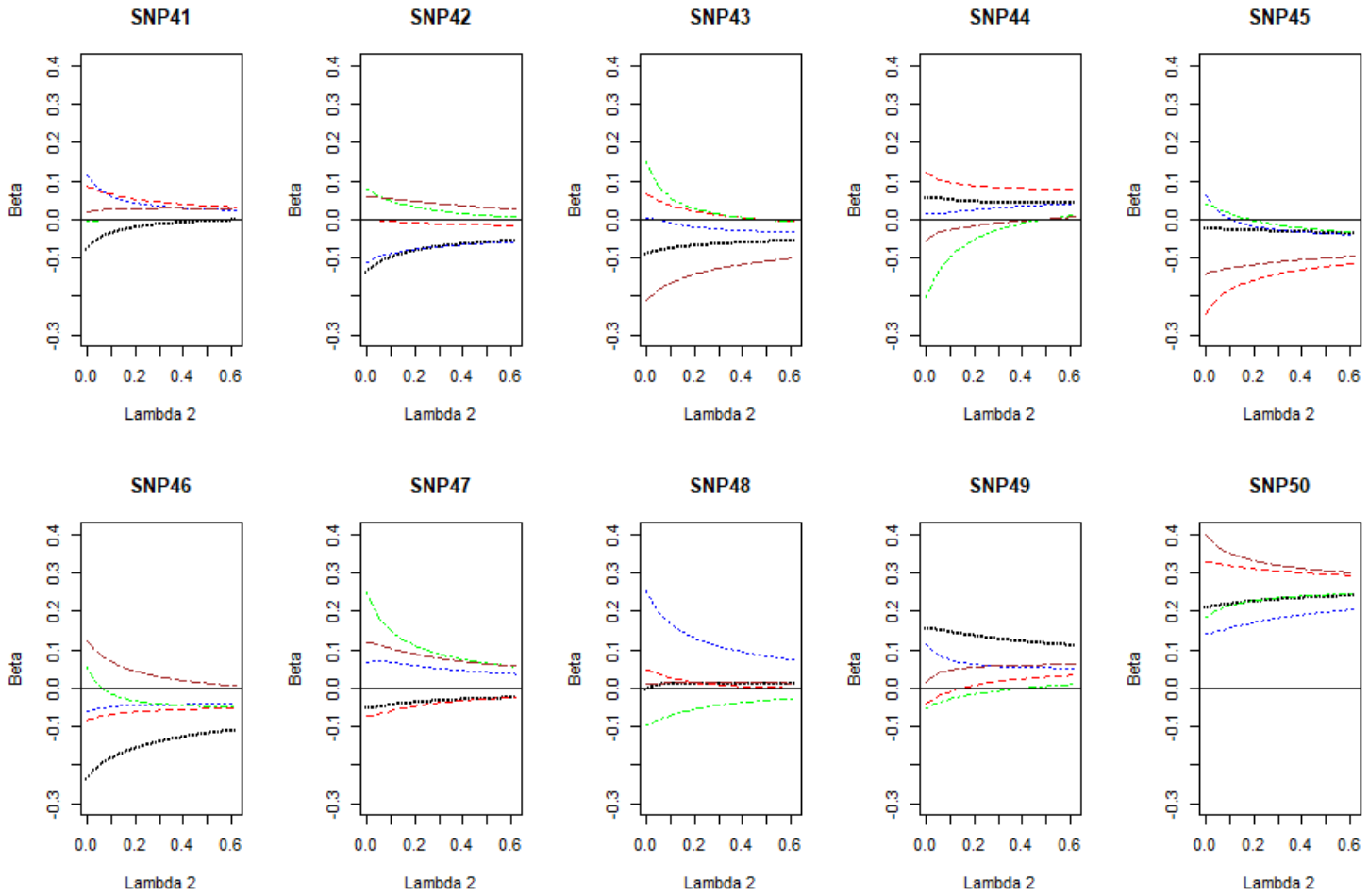


Figure D.0.5 Coefficient path plots for SNP41 to SNP50 using the Integrative LASSO when $\lambda_1 = 0$. Each line represents the SNP from a dataset and the path shows the βcoefficient on the y-axis as the λ_2 penalty increases on the bottom x-axis.

Appendix E: Convergence of the Integrative LASSO

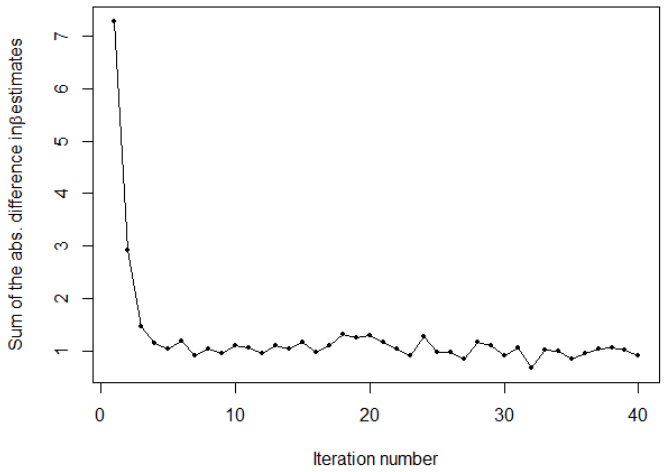


Figure E.0.1 Plot of the sum of absolute difference after each iteration across all datasets against its iteration number for a sample size of 500 in each dataset

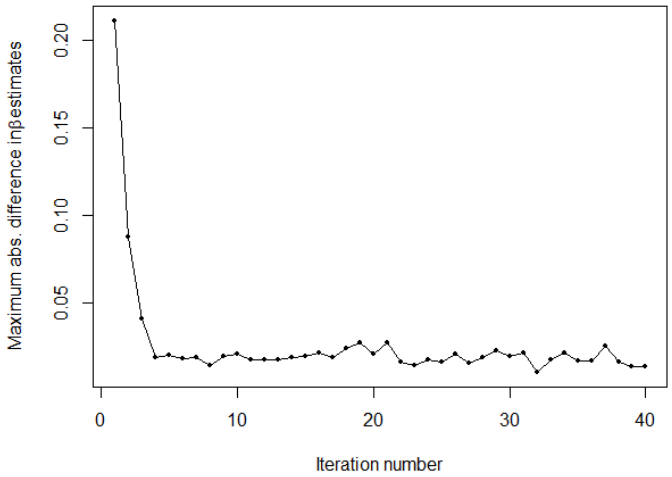


Figure E.0.2 Plot of the maximum value in the absolute difference across all SNPs in all datasets after each iteration across all datasets against its iteration number for a sample size of 500 in each dataset

Bibliography

(1) Lander ES, Linton LM, Birren B, Nusbaum C, Zody MC, Baldwin J, et al. Initial sequencing and analysis of the human genome. Nature 2001 Feb 15;409(6822):860-921.

(2) Venter JC, Adams MD, Myers EW, Li PW, Mural RJ, Sutton GG, et al. The sequence of the human genome. Science 2001 Feb 16;291(5507):1304-1351.

(3) The Wellcome Trust Case,Control Consortium. Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls. Nature 2007 06/07;447:661.

(4) Klein RJ, Zeiss C, Chew EY, Tsai JY, Sackler RS, Haynes C, et al. Complement factor H polymorphism in age-related macular degeneration. Science 2005 Apr 15;308(5720):385-389.

(5) Burdett T, Hall PN, Hastings E, Hindorff LA, Junkins HA, Klemm AK, MacArthur J, Manolio TA, Morales J, Parkinson H and Welter D. The NHGRI-EBI Catalog of published genome-wide association studies. 11/2017; Available at: www.ebi.ac.uk/gwas. Accessed 12/15, 2017.

(6) Wu TT, FAU CY, Hastie TF, Sobel EF, Lange K. Genome-wide association analysis by lasso penalized logistic regression. Bioinformatics (Oxford, England) JID - 9808944 .

(7) Storey JD. A direct approach to false discovery rates. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 2002;64(3):479-498.

(8) Storey JD, Tibshirani R. Statistical significance for genomewide studies. Proc Natl Acad Sci U S A 2003 Aug 5;100(16):9440-9445.

(9) Tibshirani R. Regression Shrinkage and Selection via the Lasso. Journal of the Royal Statistical Society.Series B (Methodological) 1996;58(1):267-288.

(10) Tobin MD, Tomaszewski M, Braund PS, Hajat C, Raleigh SM, Palmer TM, et al. Common Variants in Genes Underlying Monogenic Hypertension and Hypotension and Blood Pressure in the General Population. Hypertension 2008 June 01;51(6):1658-1664.

(11) Cho S, Kim K, Kim YJ, Lee JK, Cho YS, Lee JY, et al. Joint identification of multiple genetic variants via elastic-net variable selection in a genome-wide association analysis. Ann Hum Genet 2010 Sep 1;74(5):416-428.

(12) Hibar DP, Kohannim O, Stein JL, Chiang MC, Thompson PM. Multilocus genetic analysis of brain images. Front Genet 2011 Oct 21;2:73.

(13) Hoffman GE, Logsdon BA, Mezey JG. PUMA: a unified framework for penalized multiple regression analysis of GWAS data. PLoS Comput Biol 2013;9(6):e1003101.

(14) Szymczak S, Biernacka JM, Cordell HJ, Gonzalez-Recio O, Konig IR, Zhang H, et al. Machine learning in genome-wide association studies. Genet Epidemiol 2009;33 Suppl 1:S51-7.

(15) Yi H, Breheny P, Imam N, Liu Y, Hoeschele I. Penalized multimarker vs. single-marker regression methods for genome-wide association studies of quantitative traits. Genetics 2015 Jan;199(1):205-222.

(16) Carlsen M, Fu G, Bushman S, Corcoran C. Exploiting Linkage Disequilibrium for Ultrahigh-Dimensional Genome-Wide Data with an Integrated Statistical Approach. Genetics 2016 Feb;202(2):411-426.

(17) Hoerl AE, Kennard RW. Ridge Regression: Biased Estimation for Nonorthogonal Problems. Technometrics 1970 02/01; 2013/11;12(1):55-67.

(18) Zou H, Hastie T. Regularization and variable selection via the elastic net. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 2005;67(2):301-320.

(19) Purcell S. PLINK (Version 1.07). Available at: http://pngu.mgh.harvard.edu/purcell/plink/.

(20) Purcell S, Neale B, Todd-Brown K, Thomas L, Ferreira MA, Bender D, et al. PLINK: a tool set for whole-genome association and population-based linkage analyses. Am J Hum Genet 2007 Sep;81(3):559-575.

(21) Hastie T, Tibshirani R, Wainwright M. Statistical Learning with Sparsity: The Lasso and Generalizations. : Chapman \& Hall/CRC; 2015.

(22) Pavlou M, Ambler G, Seaman S, DeÂ Iorio M, Omar RZ. Review and evaluation of penalised regression methods for risk prediction in lowâ€dimensional data with few events. Stat Med 2015 10/06;35(7):1159-1177.

(23) Hesterberg T, Choi NH, Meier L, Fraley C. Least angle and L_1 penalized regression: A review. Statist Surv 2008;2(1):61-93.

(24) Malo N, Libiger O, Schork NJ. Accommodating Linkage Disequilibrium in Genetic-Association Analyses via Ridge Regression. Am J Hum Genet 2007 10/15;82(2):375-385.

(25) Abraham G, Kowalczyk A, Zobel J, Inouye M. Performance and robustness of penalized and unpenalized methods for genetic prediction of complex human disease. Genet Epidemiol 2013 Feb;37(2):184-195.

(26) Ayers KL, Cordell HJ. SNP selection in genome-wide and candidate gene studies via penalized logistic regression. Genet Epidemiol 2010 Dec;34(8):879-891.

(27) Waldmann P, Meszaros G, Gredler B, Fuerst C, Solkner J. Evaluation of the lasso and the elastic net in genome-wide association studies. Frontiers in Genetics 2013;4:270.

(28) Fu WJ. Penalized Regressions: The Bridge versus the Lasso. Journal of Computational and Graphical Statistics 1998 09/01;7(3):397-416.

(29) Yuan M, Lin Y. Model selection and estimation in regression with grouped variables. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 2006;68(1):49-67.

(30) Simon N, Tibshirani R. Standardization and the Group Lasso Penalty. Statistica Sinica 2012 07;22(3):983-1001.

(31) Friedman J, Hastie T, Tibshirani R. A note on the group lasso and a sparse group lasso. 2010;arXiv:1001.0736.

(32) Tibshirani R, Saunders M, Rosset S, Zhu J, Knight K. Sparsity and smoothness via the fused lasso. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 2005;67(1):91-108.

(33) Attia J, Thakkinstian A, D'Este C. Meta-analyses of molecular association studies: Methodologic lessons for genetic epidemiology. J Clin Epidemiol 2003 2017/05;56(4):297-303.

(34) Thompson JR, Attia J, Minelli C. The meta-analysis of genome-wide association studies. Briefings in Bioinformatics 2011 05/01;12(3):259-269.

(35) Curran PJ, Hussong AM. Integrative Data Analysis: The Simultaneous Analysis of Multiple Data Sets. Psychol Methods 2009 06;14(2):81-100.

(36) Lockhart R, Taylor J, Tibshirani RJ, Tibshirani R. A significance test for the lasso. Ann Statist 2014;42(2):413-468.

(37) He Q, Zhang HH, Avery CL, Lin DY. Sparse meta-analysis with high-dimensional data. Biostatistics 2016 04/01;17(2):205-220.

(38) Lin DY, Zeng D. On the relative efficiency of using summary statistics versus individual-level data in meta-analysis. Biometrika 2009 revised;97(2):321-332.

(39) Tibshirani R. Regression shrinkage and selection via the lasso: a retrospective. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 2011;73(3):273-282.

(40) Zou H. The Adaptive Lasso and Its Oracle Properties. Journal of the American Statistical Association 2006 12/01; 2013/11;101(476):1418-1429.

(41) Park T, Casella G. The Bayesian Lasso. Journal of the American Statistical Association 2008 06/01; 2013/05;103(482):681-686.

(42) Li Q, Lin N. The Bayesian elastic net. Bayesian Anal 2010 03;5(1):151-170.

(43) Xu X, Ghosh M. Bayesian Variable Selection and Estimation for Group Lasso. Bayesian Anal 2015 12;10(4):909-936.

(44) Kyung M, Gill J, Ghosh M, Casella G. Penalized regression, standard errors, and Bayesian lassos. Bayesian Anal 2010 06;5(2):369-411.

(45) Liu F, Chakraborty S, Li F, Liu Y, Lozano AC. Bayesian Regularization via Graph Laplacian. Bayesian Anal 2014 06;9(2):449-474.

(46) Zhu Y. An augmented ADMM algorithm with application to the generalized lasso problem. Journal of Computational and Graphical Statistics 2015 11/17:0-0.

(47) Lawson C, Hanson R. Solving Least Squares Problems. 15th ed.: Society for Industrial and Applied Mathematics (SIAM); 1995.

(48) Vidaurre D, Bielza C, Larrañaga P. A Survey of L1 Regression. International Statistical Review 2013;81(3):361-387.

(49) Shevade SK, Keerthi SS. A simple and efficient algorithm for gene selection using sparse logistic regression. Bioinformatics 2003 November 22;19(17):2246-2253.

(50) Daubechies I, Defrise M, De Mol C. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. Communications on Pure and Applied Mathematics 2004;57(11):1413-1457.

(51) Friedman J, Hastie T, Hofling H, Tibshirani R. Pathwise coordinate optimization. 2007 12:302-332.

(52) Friedman J, Hastie T, Tibshirani R. Sparse inverse covariance estimation with the graphical lasso. Biostatistics 2008 July 01;9(3):432-441.

(53) Friedman J, Hastie T, Tibshirani R. Regularization Paths for Generalized Linear Models via Coordinate Descent. J Stat Softw 2010;33(1):1-22.

(54) Tseng P, Yun S. A coordinate gradient descent method for nonsmooth separable minimization. Math Program 2007;117(1):387-423.

(55) Efron B, Hastie T, Johnstone I, Tibshirani R. Least angle regression. Ann Statist 2004 04;32(2):407-499.

(56) D. Donoho, Y. Tsaig. Fast Solution of *l1*-norm Minimization Problems When the Solution May be Sparse.
2006;(preprint):http://dsp.rice.edu/sites/dsp.rice.edu/files/cs/FastL1.pdf.

(57) A. Y. Yang, A. G. Balasubramanian, Z. Zhou, S. S. Sastry, Y. Ma. A review of Fast *l1*-Minimization Algorithms for Robust Face Recognition. (preprint):http://ecovision.mit.edu/~sai/12S990/Yang10-SIAM.pdf.

(58) Hesterberg T, Choi NH, Meier L, Fraley C. Least angle and L_1 penalized regression: A review. Statist Surv 2008;2(1):61-93.

(59) Osborne M, Presnell B, Turlach B. A new approach to variable selection in least squares problems. IMA Journal of Numerical Analysis 2000 July 01;20(3):389-403.

(60) A. Y. Yang, Z. Zhou, A. G. Balasubramanian, S. S. Sastry, Y. Ma. Fast -Minimization Algorithms for Robust Face Recognition. IEEE Transactions on Image Processing 2013;22(8):3234-3246.

(61) Kim. Y, Kim. Y, Kim. J. Gradient LASSO algorithm, Technical Report, Seoul National University. 2006:http://www.cs.cmu.edu/afs/cs/project/link-3/lafferty/www/ml-stat2/talks/YondaiKimGLasso-SLIDE-YD.pdf.

(62) Curtis FE, Jiang H, Robinson DP. An adaptive augmented Lagrangian method for large-scale constrained optimization. Math Program 2014;152(1):201-245.

(63) Wahlberg. B, Boyd. S, Annergren. M, Wang. Y. An ADMM Algorithm for a Class of Total Variation Regularized Estimation Problems. Proc 16th IFAC Symposium on System Identificationation 2012:p83-88.

(64) Arnold TB, Tibshirani RJ. genlasso: Path algorithm for generalized lasso problems. 2014;R package version 1.3:http://CRAN.R-project.org/package=genlasso.

(65) Mee Young Park and Trevor Hastie. glmpath: L1 Regularization Path for Generalized Linear Models and Cox Proportional Hazards Model. 2013;R package version 0.97:http://CRAN.R-project.org/package=glmpath.

(66) Justin Lokhorst, Bill Venables, Berwin Turlach. lasso2: L1 constrained estimation aka `lasso'. 2014;R package version 1.2-19:http://CRAN.R-project.org/package=lasso2.

(67) Hui Zou and Trevor Hastie. elasticnet: Elastic-Net for Sparse Estimation and Sparse PCA. 2012;R package version 1.1:http://CRAN.R-project.org/package=elasticnet.

(68) Yinyin Yuan. lol: Lots Of Lasso. 2011;R package version 1.10.0:https://www.bioconductor.org/packages/release/bioc/html/lol.html.

(69) Waldron L, Pintilie M, Tsao M, Shepherd FA, Huttenhower C, Jurisica I. Optimized application of penalized regression methods to diverse genomic data. Bioinformatics 2011;27(24):3399-3406.

(70) Stephan Ritter and Alan Hubbard. relaxnet: Relaxation of glmnet models. 2013;R package version 0.3-2:http://CRAN.R-project.org/package=relaxnet.

(71) Nicolai Meinshausen. relaxo: Relaxed Lasso. 2012;R package version 0.1-2:http://CRAN.R-project.org/package=relaxo.

(72) Yi Yang and Hui Zou. gglasso: Group Lasso Penalized Learning Using A Unified BMD Algorithm. 2014;R package version 1.3:http://CRAN.R-project.org/package=gglasso.

(73) Friedman J, Hastie T, Tibshirani R. Sparse inverse covariance estimation with the graphical lasso. Biostatistics 2008 July 01;9(3):432-441.

(74) Schelldorfer J, Meier L, Bühlmann P. GLMMLasso: An Algorithm for High-Dimensional Generalized Linear Mixed Models Using ℓ1-Penalization. Journal of Computational and Graphical Statistics 2014 04/03;23(2):460-477.

(75) Lukas Meier. grplasso: Fitting user specified models with Group Lasso penalty. 2015;R package version 0.4-5:http://CRAN.R-project.org/package=grplasso.

(76) Dingfeng Jiang. grppenalty: Concave 1-norm and 2-norm group penalty in linear and logistic regression. 2014;R package version 2.1-0:http://CRAN.R-project.org/package=grppenalty.

(77) Breheny P, Huang J. Group descent algorithms for nonconvex penalized linear and logistic regression models with grouped predictors. Statistics and Computing 2013;25(2):173-187.

(78) Quentin Grimonprez. HDPenReg: High-Dimensional Penalized Regression. 2015;R package version 0.91:http://CRAN.R-project.org/package=HDPenReg.

(79) Rajen Shah. LassoBacktracking: Modelling Interactions in High-Dimensional Data with Backtracking. 2016;R package version 0.1.1:http://CRAN.R-project.org/package=LassoBacktracking.

(80) Tobias Abenius. lassoshooting: L1 regularized regression (Lasso) solver using the Cyclic Coordinate Descent algorithm aka Lasso Shooting. 2012;R package version 0.1.5-1:http://CRAN.R-project.org/package=lassoshooting.

(81) Goeman JJ. L1 Penalized Estimation in the Cox Proportional Hazards Model. Biometrical Journal 2010;52(1):70-84.

(82) Peng. B. QICD: Estimate the Coefficients for Non-Convex Penalized Quantile Regression Model by using QICD Algorithm. 2016;R package version 1.0.1:http://CRAN.R-project.org/package=QICD.

(83) Sherwood. B, Maidman. A. rqPen: Penalized Quantile Regression. 2016;R package version 1.3:http://CRAN.R-project.org/package=rqPen.

(84) Simon N, Friedman J, Hastie T, Tibshirani R. A Sparse-Group Lasso. Journal of Computational and Graphical Statistics 2013 04/01;22(2):231-245.

(85) Mee Young Park and Trevor Hastie. stepPlr: L_2 penalized logistic regression with a stepwise variable selection. 2010;R package version 0.92:http://CRAN.R-project.org/package=stepPlr.

(86) Tibshirani RJ, Taylor J. The Solution Path of the Generalized Lasso. Annals of Statistics 2011 JUN;39(3):1335-1371.

(87) Sabourin JA, Valdar W, Nobel AB. A permutation approach for selecting the penalty parameter in penalized model selection. Biometrics 2015;71(4):1185-1194.

(88) Ayers KL, Cordell HJ. SNP Selection in genome-wide and candidate gene studies via penalized logistic regression. Genet Epidemiol 2010;34(8):879-891.

(89) Homrignausen D, McDonald D. The lasso, persistence, and cross-validation. JMLR W&CP 2013;28:1031-1039.

(90) S. Chand. On tuning parameter selection of lasso-type methods - a monte carlo study. Proceedings of 2012 9th International Bhurban Conference on Applied Sciences & Technology (IBCAST) 2012;1(1):120-129.

(91) Refaeilzadeh P, Tang L, Liu H. Cross-Validation. In: LIU L, Ã–ZSU MT, editors. Encyclopedia of Database Systems Boston, MA: Springer US; 2009. p. 532-538.

(92) Hastie T, Tibshirani R, Friedman J. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. : Springer; 2003.

(93) A Study of Cross-validation and Bootstrap for Accuracy Estimation and Model Selection. Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2 San Francisco, CA, USA: Morgan Kaufmann Publishers Inc; 1995.

(94) Kaneko S, Hirakawa A, Hamada C. Gene Selection using a High-Dimensional Regression Model with Microarrays in Cancer Prognostic Studies. Cancer Informatics 2012 02/27;11:29-39.

(95) Feng Y, Yu Y. Consistent Cross-Validation for Tuning Parameter Selection in High-Dimensional Variable Selection. eprint arXiv:1308 5390 2013.

(96) Yu Y, Feng Y. Modified Cross-Validation for Penalized High-Dimensional Linear Regression Models. Journal of Computational and Graphical Statistics 2014 10/02;23(4):1009-1027.

(97) Krstajic D, Buturovic LJ, Leahy DE, Thomas S. Cross-validation pitfalls when selecting and assessing regression and classification models. Journal of Cheminformatics 2014 03/25;6(1):10.

(98) Meinshausen N, Bühlmann P. Stability selection. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 2010;72(4):417-473.

(99) Alexander DH, Lange K. Stability selection for genome-wide association. Genet Epidemiol 2011;35(7):722-728.

(100) Wang H, Li R, Tsai C. Tuning parameter selectors for the smoothly clipped absolute deviation method. Biometrika 2007 August 01;94(3):553-568.

(101) Hirose K, Tateishi S, Konishi S. Tuning parameter selection in sparse regression modeling. Comput Stat Data Anal 2013 3;59:28-40.

(102) Kirkland L, Kanfer F, Millard S. LASSO tuning parameter selection. Annual Proceedings of the South African Statistical Association Conference 2015:49-56.

(103) Schwarz G. Estimating the Dimension of a Model. Ann Statist 1978 03;6(2):461-464.

(104) Zou H, Hastie T, Tibshirani R. On the "degrees of freedom" of the lasso. Ann Statist 2007 10;35(5):2173-2192.

(105) Ayers KL, Cordell HJ. SNP selection in genome-wide and candidate gene studies via penalized logistic regression. Genet Epidemiol 2010 Dec;34(8):879-891.

(106) Arbet J, McGue M, Chatterjee S, Basu S. Resampling-based tests for Lasso in genome-wide association studies. BMC Genet 2017 Jul 24;18(1):70-017-0533-3.

(107) Stein CM. Estimation of the Mean of a Multivariate Normal Distribution. Ann Statist 1981;6:1135--1151.

(108) Sun T, Zhang C. Scaled sparse linear regression. Biometrika 2012 2018/07;99(4):879-898.

(109) Zhou H, Sehl ME, Sinsheimer JS, Lange K. Association screening of common and rare genetic variants by penalized regression. Bioinformatics 2010 Oct 1;26(19):2375-2382.

(110) Devlin B, Roeder K. Genomic control for association studies. Biometrics 1999 Dec;55(4):997-1004.

(111) Lin DY, Zeng D. Meta-analysis of genome-wide association studies: no efficiency gain in using individual participant data. Genet Epidemiol 2010 Jan;34(1):60-66.

(112) Biswas S, Lin S. Logistic Bayesian LASSO for identifying association with rare haplotypes and application to age-related macular degeneration. Biometrics 2012 Jun;68(2):587-597.

(113) Biswas S, Xia S, Lin S. Detecting rare haplotype-environment interaction with logistic Bayesian LASSO. Genet Epidemiol 2014 Jan;38(1):31-41.

(114) Gonzalez-Recio O, Forni S. Genome-wide prediction of discrete traits using Bayesian regressions and machine learning. Genet Sel Evol 2011 Feb 17;43:7-9686-43-7.

(115) Li J, Das K, Fu G, Li R, Wu R. The Bayesian lasso for genome-wide association studies. Bioinformatics 2011 Feb 15;27(4):516-523.

(116) Li J, Wang Z, Li R, Wu R. Bayesian Group Lasso for Nonparametric Varying-Coefficient Models with Application to Functional Genome-Wide Association Studies. Ann Appl Stat 2015 Jun;9(2):640-664.

(117) Zhang X, Xue F, Liu H, Zhu D, Peng B, Wiemels JL, et al. Integrative Bayesian variable selection with gene-based informative priors for genome-wide association studies. BMC Genet 2014 Dec 10;15:130-014-0130-7.

(118) Zhang Y, Biswas S. An Improved Version of Logistic Bayesian LASSO for Detecting Rare Haplotype-Environment Interactions with Application to Lung Cancer. Cancer Inform 2015 Feb 9;14(Suppl 2):11-16.

(119) Abraham G, Kowalczyk A, Zobel J, Inouye M. Performance and robustness of penalized and unpenalized methods for genetic prediction of complex human disease. Genet Epidemiol 2013 Feb;37(2):184-195.

(120) Austin E, Pan W, Shen X. Penalized Regression and Risk Prediction in Genome-Wide Association Studies. Stat Anal Data Min 2013 Aug 1;6(4):10.1002/sam.11183.

(121) Cheng Y, Jiang T, Zhu M, Li Z, Zhang J, Wang Y, et al. Risk assessment models for genetic risk predictors of lung cancer using two-stage replication for Asian and European populations. Oncotarget 2016 Jul 5;8(33):53959-53967.

(122) Choi S, Bae S, Park T. Risk Prediction Using Genome-Wide Association Studies on Type 2 Diabetes. Genomics Inform 2016 Dec;14(4):138-148.

(123) Gim J, Kim W, Kwak SH, Choi H, Park C, Park KS, et al. Improving Disease Prediction by Incorporating Family Disease History in Risk Prediction Models with Large-Scale Genetic Data. Genetics 2017 Nov;207(3):1147-1155.

(124) Huls A, Ickstadt K, Schikowski T, Kramer U. Detection of gene-environment interactions in the presence of linkage disequilibrium and noise by using genetic risk scores with internal weights from elastic net regression. BMC Genet 2017 Jun 12;18(1):55-017-0519-1.

(125) Kooperberg C, LeBlanc M, Obenchain V. Risk prediction using genome-wide association studies. Genet Epidemiol 2010 Nov;34(7):643-652.

(126) Ogutu JO, Piepho HP. Regularized group regression methods for genomic prediction: Bridge, MCP, SCAD, group bridge, group lasso, sparse group lasso, group

MCP and group SCAD. BMC Proc 2014 Oct 7;8(Suppl 5):S7-6561-8-S5-S7. eCollection 2014.

(127) Sung YJ, Rice TK, Shi G, Gu CC, Rao D. Comparison between single-marker analysis using Merlin and multi-marker analysis using LASSO for Framingham simulated data. BMC Proc 2009;3(Suppl 7):S27.

(128) Waldmann P, Meszaros G, Gredler B, Fuerst C, Solkner J. Evaluation of the lasso and the elastic net in genome-wide association studies. Front Genet 2013 Dec 4;4:270.

(129) Kraja AT, Culverhouse R, Daw EW, Wu J, Van Brunt A, Province MA, et al. The Genetic Analysis Workshop 16 Problem 3: simulation of heritable longitudinal cardiovascular phenotypes based on actual genome-wide single-nucleotide polymorphisms in the Framingham Heart Study. BMC Proceedings 2009 12/15;3:S4-S4.

(130) Hastie T, Tibshirani R, Friedman J. The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition (Springer Series in Statistics). ; 2009.

(131) Bao M, Wang K. Genome-wide association studies using a penalized moving-window regression. Bioinformatics 2017 Aug 17.

(132) Liu J, Wang K, Ma S, Huang J. Accounting for linkage disequilibrium in genome-wide association studies: A penalized regression method. Stat Interface 2013 Jan 1;6(1):99-115.

(133) Ayers KL, Cordell HJ. Identification of grouped rare and common variants via penalized logistic regression. Genet Epidemiol 2013 Sep;37(6):592-602.

(134) Larson NB, Schaid DJ. Regularized rare variant enrichment analysis for case-control exome sequencing data. Genet Epidemiol 2014 Feb;38(2):104-113.

(135) Zhou H, Sehl ME, Sinsheimer JS, Lange K. Association screening of common and rare genetic variants by penalized regression. Bioinformatics 2010 Oct 1;26(19):2375-2382.

(136) Frost HR, Andrew AS, Karagas MR, Moore JH. A screening-testing approach for detecting gene-environment interactions using sequential penalized and unpenalized multiple logistic regression. Pac Symp Biocomput 2015:183-194.

(137) Frost HR, Shen L, Saykin AJ, Williams SM, Moore JH, Alzheimer's Disease Neuroimaging Initiative. Identifying significant gene-environment interactions using a combination of screening testing and hierarchical false discovery rate control. Genet Epidemiol 2016 Nov;40(7):544-557.

(138) Assimes TL, Lee IT, Juang JM, Guo X, Wang TD, Kim ET, et al. Genetics of Coronary Artery Disease in Taiwan: A Cardiometabochip Study by the Taichi Consortium. PLoS One 2016 Mar 16;11(3):e0138014.

(139) Denis M, Enquobahrie DA, Tadesse MG, Gelaye B, Sanchez SE, Salazar M, et al. Placental genome and maternal-placental genetic interactions: a genome-wide and candidate gene association study of placental abruption. PLoS One 2014 Dec 30;9(12):e116346.

(140) Li J, Dan J, Li C, Wu R. A model-free approach for detecting interactions in genetic association studies. Brief Bioinform 2014 Nov;15(6):1057-1068.

(141) Yang C, Wan X, Yang Q, Xue H, Yu W. Identifying main effects and epistatic interactions from large-scale SNP data via adaptive group Lasso. BMC Bioinformatics 2010 01/18;11(1):S18.

(142) Ahmed I, Hartikainen AL, Jarvelin MR, Richardson S. False discovery rate estimation for stability selection: application to genome-wide association studies. Stat Appl Genet Mol Biol 2011 Nov 28;10(1):10.2202/1544-6115.1663.

(143) Armstrong DL, Zidovetzki R, Alarcon-Riquelme ME, Tsao BP, Criswell LA, Kimberly RP, et al. GWAS identifies novel SLE susceptibility genes and explains the association of the HLA region. Genes Immun 2014 Sep;15(6):347-354.

(144) Shi G, Boerwinkle E, Morrison AC, Gu CC, Chakravarti A, Rao D. Mining Gold Dust under the Genome Wide Significance Level: A Two-Stage Approach to Analysis of GWAS. Genet Epidemiol 2011 Feb;35(2):111-118.

(145) Wu TT, Chen YF, Hastie T, Sobel E, Lange K. Genome-wide association analysis by lasso penalized logistic regression. Bioinformatics 2009 Mar 15;25(6):714-721.

(146) Hong S, Kim Y, Park T. Practical issues in screening and variable selection in genome-wide association analysis. Cancer Inform 2015 Jan 14;13(Suppl 7):55-65.

(147) Jiang Y, He Y, Zhang H. Variable Selection with Prior Information for Generalized Linear Models via the Prior LASSO Method. J Am Stat Assoc 2016;111(513):355-376.

(148) Kohannim O, Hibar DP, Stein JL, Jahanshad N, Hua X, Rajagopalan P, et al. Discovery and Replication of Gene Influences on Brain Structure Using LASSO Regression. Front Neurosci 2012;6:10.3389/fnins.2012.00115.

(149) Yao TC, Du G, Han L, Sun Y, Hu D, Yang JJ, et al. Genome-Wide Association Study of Lung Function Phenotypes in a Founder Population. J Allergy Clin Immunol 2014 Jan;133(1):248-55.e1-10.

(150) Benjamini Y, Hochberg Y. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. Journal of the Royal Statistical Society.Series B (Methodological) 1995;57(1):289-300.

(151) Murray CJ, Lopez AD. Global mortality, disability, and the contribution of risk factors: Global Burden of Disease Study. Lancet 1997 May 17;349(9063):1436-1442.

(152) Castelli WP. Lipids, risk factors and ischaemic heart disease. Atherosclerosis 1996 7;124, Supplement(0):S1-S9.

(153) Castelli WP, Anderson K, Wilson PWF, Levy D. Lipids and risk of coronary heart disease The Framingham Study. Ann Epidemiol 1992 0;2(1–2):23-28.

(154) Wilson PWF, D'Agostino RB, Levy D, Belanger AM, Silbershatz H, Kannel WB. Prediction of Coronary Heart Disease Using Risk Factor Categories. Circulation 1998 May 01;97(18):1837-1847.

(155) National Heart, Lung and Blood Institute. Third Report of the Expert Panel on Detection, Evaluation, and Treatment of High Blood Cholesterol in Adults (Adult Treatment Panel III). 2002.

(156) Gennest J, Libby P. Lipoprotein disorders and cardiovascular disease. Braunwald's Heart Disease: A Textbook of Cardiovascular Medicine. 9th ed.: Saunders Elsevier; 2011. p. chap 47.

(157) Bui QT, Prempeh M, Wilensky RL. Atherosclerotic plaque development. Int J Biochem Cell Biol 2009 11;41(11):2109-2113.

(158) NHS. High cholesterol. 27/08/2015; Available at: https://www.nhs.uk/Conditions/Cholesterol/Pages/Introduction.aspx. Accessed 10/22, 2017.

(159) Pilia G, Chen WM, Scuteri A, Orru M, Albai G, Dei M, et al. Heritability of cardiovascular and personality traits in 6,148 Sardinians. PLoS Genet 2006 Aug 25;2(8):e132.

(160) de Oliveira CM, Pereira AC, de Andrade M, Soler JM, Krieger JE. Heritability of cardiovascular risk factors in a Brazilian population: Baependi Heart Study. BMC Med Genet 2008 Apr 22;9:32-2350-9-32.

(161) Smigielski EM, Sirotkin K, Ward M, Sherry ST. dbSNP: a database of single nucleotide polymorphisms. Nucleic Acids Res 1999 11/01;28(1):352-355.

(162) Asselbergs FW, Guo Y, van Iperen EP, Sivapalaratnam S, Tragante V, Lanktree MB, et al. Large-scale gene-centric meta-analysis across 32 studies identifies multiple lipid loci. Am J Hum Genet 2012 Nov 2;91(5):823-838.

(163) Aulchenko YS, Ripatti S, Lindqvist I, Boomsma D, Heid IM, Pramstaller PP, et al. Loci influencing lipid levels and coronary heart disease risk in 16 European population cohorts. Nat Genet 2009 Jan;41(1):47-55.

(164) Chasman DI, Pare G, Zee RY, Parker AN, Cook NR, Buring JE, et al. Genetic loci associated with plasma concentration of low-density lipoprotein cholesterol, high-density lipoprotein cholesterol, triglycerides, apolipoprotein A1, and Apolipoprotein B

among 6382 white women in genome-wide analysis with replication. Circ Cardiovasc Genet 2008 Oct;1(1):21-30.

(165) Chasman DI, Pare G, Mora S, Hopewell JC, Peloso G, Clarke R, et al. Forty-three loci associated with plasma lipoprotein size, concentration, and cholesterol content in genome-wide analysis. PLoS Genet 2009 Nov;5(11):e1000730.

(166) Kathiresan S, Melander O, Guiducci C, Surti A, Burtt NP, Rieder MJ, et al. Six new loci associated with blood low-density lipoprotein cholesterol, high-density lipoprotein cholesterol or triglycerides in humans. Nat Genet 2008 Feb;40(2):189-197.

(167) Kathiresan S, Willer CJ, Peloso GM, Demissie S, Musunuru K, Schadt EE, et al. Common variants at 30 loci contribute to polygenic dyslipidemia. Nat Genet 2009 Jan;41(1):56-65.

(168) Kim YJ, Go MJ, Hu C, Hong CB, Kim YK, Lee JY, et al. Large-scale genome-wide association studies in East Asians identify new genetic loci influencing metabolic traits. Nat Genet 2011 Sep 11;43(10):990-995.

(169) Lettre G, Palmer CD, Young T, Ejebe KG, Allayee H, Benjamin EJ, et al. Genome-wide association study of coronary heart disease and its risk factors in 8,090 African Americans: the NHLBI CARe Project. PLoS Genet 2011 Feb 10;7(2):e1001300.

(170) Middelberg RP, Ferreira MA, Henders AK, Heath AC, Madden PA, Montgomery GW, et al. Genetic variants in LPL, OASL and TOMM40/APOE-C1-C2-C4 genes are associated with multiple cardiovascular-related traits. BMC Med Genet 2011 Sep 24;12:123-2350-12-123.

(171) Musunuru K, Romaine SP, Lettre G, Wilson JG, Volcik KA, Tsai MY, et al. Multi-ethnic analysis of lipid-associated loci: the NHLBI CARe project. PLoS One 2012;7(5):e36473.

(172) Rasmussen-Torvik LJ, Pacheco JA, Wilke RA, Thompson WK, Ritchie MD, Kho AN, et al. High density GWAS for LDL cholesterol in African Americans using electronic medical records reveals a strong protective variant in APOE. Clin Transl Sci 2012 Oct;5(5):394-399.

(173) Roslin NM, Hamid JS, Paterson AD, Beyene J. Genome-wide association analysis of cardiovascular-related quantitative traits in the Framingham Heart Study. BMC Proc 2009 Dec 15;3 Suppl 7:S117.

(174) Sabatti C, Service SK, Hartikainen AL, Pouta A, Ripatti S, Brodsky J, et al. Genome-wide association analysis of metabolic traits in a birth cohort from a founder population. Nat Genet 2009 Jan;41(1):35-46.

(175) Saleheen D, Soranzo N, Rasheed A, Scharnagl H, Gwilliam R, Alexander M, et al. Genetic determinants of major blood lipids in Pakistanis compared with Europeans. Circ Cardiovasc Genet 2010 Aug;3(4):348-357.

(176) Sandhu MS, Waterworth DM, Debenham SL, Wheeler E, Papadakis K, Zhao JH, et al. LDL-cholesterol concentrations: a genome-wide association study. Lancet 2008 Feb 9;371(9611):483-491.

(177) Shen H, Damcott CM, Rampersaud E, Pollin TI, Horenstein RB, McArdle PF, et al. Familial defective apolipoprotein B-100 and increased low-density lipoprotein cholesterol and coronary artery calcification in the old order amish. Arch Intern Med 2010 Nov 8;170(20):1850-1855.

(178) Smith EN, Chen W, Kahonen M, Kettunen J, Lehtimaki T, Peltonen L, et al. Longitudinal genome-wide association of cardiovascular disease risk factors in the Bogalusa heart study. PLoS Genet 2010 Sep 9;6(9):10.1371/journal.pgen.1001094.

(179) Talmud PJ, Drenos F, Shah S, Shah T, Palmen J, Verzilli C, et al. Gene-centric association signals for lipids and apolipoproteins identified via the HumanCVD BeadChip. Am J Hum Genet 2009 Nov;85(5):628-642.

(180) Teslovich TM, Musunuru K, Smith AV, Edmondson AC, Stylianou IM, Koseki M, et al. Biological, clinical and population relevance of 95 loci for blood lipids. Nature 2010 Aug 5;466(7307):707-713.

(181) Trompet S, de Craen AJ, Postmus I, Ford I, Sattar N, Caslake M, et al. Replication of LDL GWAs hits in PROSPER/PHASE as validation for future (pharmaco)genetic analyses. BMC Med Genet 2011 Oct 6;12:131-2350-12-131.

(182) Waterworth DM, Ricketts SL, Song K, Chen L, Zhao JH, Ripatti S, et al. Genetic variants influencing circulating lipid levels and risk of coronary artery disease. Arterioscler Thromb Vasc Biol 2010 Nov;30(11):2264-2276.

(183) Willer CJ, Mohlke KL. Finding genes and variants for lipid levels after genome-wide association analysis. Curr Opin Lipidol 2012 Apr;23(2):98-103.

(184) Wu Y, Waite LL, Jackson AU, Sheu WH, Buyske S, Absher D, et al. Trans-ethnic fine-mapping of lipid loci identifies population-specific signals and allelic heterogeneity that increases the trait variance explained. PLoS Genet 2013 Mar;9(3):e1003379.

(185) Willer CJ, Sanna S, Jackson AU, Scuteri A, Bonnycastle LL, Clarke R, et al. Newly identified loci that influence lipid concentrations and risk of coronary artery disease. Nat Genet 2008 Feb;40(2):161-169.

(186) Chasman DI, Giulianini F, MacFadyen J, Barratt BJ, Nyberg F, Ridker PM. Genetic determinants of statin-induced low-density lipoprotein cholesterol reduction: the Justification for the Use of Statins in Prevention: an Intervention Trial Evaluating Rosuvastatin (JUPITER) trial. Circ Cardiovasc Genet 2012 Apr 1;5(2):257-264.

(187) Kathiresan S, Manning AK, Demissie S, D'Agostino RB, Surti A, Guiducci C, et al. A genome-wide association study for blood lipid phenotypes in the Framingham Heart Study. BMC Med Genet 2007 Sep 19;8 Suppl 1:S17.

(188) Turner S, Armstrong LL, Bradford Y, Carlson CS, Crawford DC, Crenshaw AT, et al. Quality control procedures for genome-wide association studies. Curr Protoc Hum Genet 2011 Jan;Chapter 1:Unit1.19.

(189) Weale ME. Quality control for genome-wide association studies. Methods Mol Biol 2010;628:341-372.

(190) Guo SW, Thompson EA. Performing the exact test of Hardy-Weinberg proportion for multiple alleles. Biometrics 1992 Jun;48(2):361-372.

(191) Panagiotou OA, Ioannidis JP, Genome-Wide Significance Project. What should the genome-wide significance threshold be? Empirical replication of borderline genetic associations. Int J Epidemiol 2012 Feb;41(1):273-286.

(192) Duggal P, Gillanders EM, Holmes TN, Bailey-Wilson JE. Establishing an adjusted p-value threshold to control the family-wide type 1 error in genome wide association studies. BMC Genomics 2008 Oct 31;9:516-2164-9-516.

(193) Johnson RC, Nelson GW, Troyer JL, Lautenberger JA, Kessing BD, Winkler CA, et al. Accounting for multiple comparisons in a genome-wide association study (GWAS). BMC Genomics 2010 Dec 22;11:724-2164-11-724.

(194) Hendricks AE, Dupuis J, Logue MW, Myers RH, Lunetta KL. Correction for multiple testing in a gene region. Eur J Hum Genet 2013 07/10.

(195) Dabney A, Storey JD, Warnes GR. qvalue: Q-value estimation for false discovery rate control. ;R package version 1.32.0.

(196) R Development Core Team. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing 2012 Vienna, Austria;{ISBN} 3-900051-07-0:URL http://www.R-project.org/.

(197) Rogers AR. How Population Growth Affects Linkage Disequilibrium. Genetics 2014 06/06.

(198) Slatkin M. Linkage disequilibrium - understanding the evolutionary past and mapping the medical future. Nature reviews.Genetics 2008 06;9(6):477-485.

(199) Ardlie KG, Kruglyak L, Seielstad M. Patterns of linkage disequilibrium in the human genome. Nat Rev Genet 2002 Apr;3(4):299-309.

(200) Rogers AR, Huff C. Linkage Disequilibrium Between Loci With Unknown Phase. Genetics 2009 05/03;182(3):839-844.

(201) Liu G, Wang Y, Wong L. FastTagger: an efficient algorithm for genome-wide tag SNP selection using multi-marker linkage disequilibrium. BMC Bioinformatics 2010;11(1):66.

(202) Howie B, Fuchsberger C, Stephens M, Marchini J, Abecasis GR. Fast and accurate genotype imputation in genome-wide association studies through pre-phasing. Nat Genet 2012 Jul 22;44(8):955-959.

(203) Marchini J, Howie B. Genotype imputation for genome-wide association studies. Nat Rev Genet 2010 Jul;11(7):499-511.

(204) Clark AG. Inference of haplotypes from PCR-amplified samples of diploid populations. Mol Biol Evol 1990 Mar;7(2):111-122.

(205) Excoffier L, Slatkin M. Maximum-likelihood estimation of molecular haplotype frequencies in a diploid population. Mol Biol Evol 1995 Sep;12(5):921-927.

(206) Hao K, Di X, Cawley S. LdCompare: rapid computation of single- and multiple-marker r2 and genetic coverage. Bioinformatics 2007 01/15;23(2):252-254.

(207) McVean GAT, Cardin NJ. Approximating the coalescent with recombination. Philosophical Transactions of the Royal Society B: Biological Sciences 2005 07/07;360(1459):1387-1393.

(208) Browning SR, Browning BL. Haplotype phasing: existing methods and new developments. Nat Rev Genet 2011 Sep 16;12(10):703-714.

(209) Clayton D, Leung HT. An R package for analysis of whole-genome association studies. Hum Hered 2007;64(1):45-51.

(210) GenABEL project developers. GenABEL: genome-wide SNP association analysis. 2013;R package version 1.8-0:http://CRAN.R-project.org/package=GenABEL.

(211) Warnes GR, Gorjanc G, Leisch F, Man M. genetics: Population Genetics. 2013;R package version 1.3.8.1:http://CRAN.R-project.org/package=genetics.

(212) Clayton D. snpStats: SnpMatrix and XSnpMatrix classes and methods. 2012;R package version 1.8.2.

(213) Laurie CC, Doheny KF, Mirel DB, Pugh EW, Bierut LJ, Bhangale T, et al. Quality control and quality assurance in genotypic data for genome-wide association studies. Genet Epidemiol 2010;34(6):591-602.

(214) Reed E, Nunez S, Kulp D, Qian J, Reilly MP, Foulkes AS. A guide to genome-wide association analysis and post-analytic interrogation. Stat Med 2015;34(28):3769-3792.

(215) Solovieff N, Hartley SW, Baldwin CT, Perls TT, Steinberg MH, Sebastiani P. Clustering by genetic ancestry using genome-wide SNP data. BMC Genetics 2010;11(1):108.

(216) Abraham G, Inouye M. Fast Principal Component Analysis of Large-Scale Genome-Wide Data. PLoS ONE 2014 03/07;9(4):e93766.

(217) Price AL, Zaitlen NA, Reich D, Patterson N. New approaches to population stratification in genome-wide association studies. Nature reviews.Genetics 2010 07;11(7):459-463.

(218) Novembre J, Johnson T, Bryc K, Kutalik Z, Boyko AR, Auton A, et al. Genes mirror geography within Europe. Nature 2008 08/31;456(7218):98-101.

(219) Dudbridge F, Newcombe PJ. Accuracy of Gene Scores when Pruning Markers by Linkage Disequilibrium. Hum Hered 2015;80(4):178-186.

(220) Ware EB, Schmitz LL, Faul JD, Gard A, Mitchell C, Smith JA, et al. Heterogeneity in polygenic scores for common human traits. bioRxiv 2017 Cold Spring Harbor Laboratory Press.

(221) So H, Sham PC. Improving polygenic risk prediction from summary statistics by an empirical Bayes approach. Scientific Reports 2017;7.

(222) Pasaniuc B, Price AL. Dissecting the genetics of complex traits using summary association statistics. Nat Rev Genet 2017 print;18(2):117-127.

(223) Shi J, Park J, Duan J, Berndt S, Moy W, Wheeler W, et al. Winners curse correction and variable thresholding improve performance of polygenic risk modeling based on summary-level data from genome-wide association studies. bioRxiv 2016 01/10.

(224) Hao K. Genome-wide selection of tag SNPs using multiple-marker correlation. Bioinformatics 2007 12/01;23(23):3178-3184.

(225) Zheng X, Levine D, Shen J, Gogarten SM, Laurie C, Weir BS. A high-performance computing toolset for relatedness and principal component analysis of SNP data. Bioinformatics 2012 10/06;28(24):3326-3328.

(226) Wang WB, Jiang T. A new model of multi-marker correlation for genome-wide tag SNP selection. Genome Inform 2008;21:27-41.

(227) Stram DO. Tag SNP selection for association studies. Genet Epidemiol 2004;27(4):365-374.

(228) Silesian AP, Szyda J. Population parameters incorporated into genome-wide tagSNP selection. animal 2013;7(8):1227-1230.

(229) Badke YM, Bates RO, Ernst CW, Schwab C, Fix J, Van Tassell C,P., et al. Methods of tagSNP selection and other variables affecting imputation accuracy in swine. BMC Genetics 2013 01/29;14:8-8.

(230) Frommlet F. Tag SNP selection based on clustering according to dominant sets found using replicator dynamics. Advances in Data Analysis and Classification 2010;4(1):65-83.

(231) Long T, Hicks M, Yu H, Biggs WH, Kirkness EF, Menni C, et al. Whole-genome sequencing identifies common-to-rare variants associated with human blood metabolites. Nat Genet 2017 print;49(4):568-578.

(232) Stringer S, Minica,C.C., Verweij KJH, Mbarek H, Bernard M, Derringer J, et al. Genome-wide association study of lifetime cannabis use based on a large meta-analytic sample of 32330 subjects from the International Cannabis Consortium. Translational Psychiatry 2015 12/21;6(3):e769.

(233) Yang J, Lee SH, Goddard ME, Visscher PM. GCTA: a tool for genome-wide complex trait analysis. Am J Hum Genet 2011 Jan 7;88(1):76-82.

(234) Scheet P, Stephens M. A fast and flexible statistical model for large-scale population genotype data: applications to inferring missing genotypes and haplotypic phase. Am J Hum Genet 2006 Apr;78(4):629-644.

(235) Stewart LA. Practical methodology of meta-analyses (overviews) using updated individual patient data. Statist Med 1995 10/15;14(19):2057-2079.

(236) Lambert PC, Sutton AJ, Abrams KR, Jones DR. A comparison of summary patient-level covariates in meta-regression with individual patient data meta-analysis. J Clin Epidemiol 2002 1;55(1):86-94.

(237) Berlin JA, Santanna J, Schmid CH, Szczech LA, Feldman HI. Individual patient- versus group-level data meta-regressions for the investigation of treatment effect modifiers: ecological bias rears its ugly head. Stat Med 2002;21(3):371-387.

(238) Ma S, Huang J. Regularized gene selection in cancer microarray meta-analysis. BMC Bioinformatics 2009;10(1):1.

(239) Huang Y, Huang J, Shia BC, Ma S. Identification of cancer genomic markers via integrative sparse boosting. Biostatistics 2012 Jul;13(3):509-522.

(240) Ma S, Huang J, Song X. Integrative analysis and variable selection with multiple high-dimensional data sets. Biostatistics 2011 Oct;12(4):763-775.

(241) Simon N, Friedman J, Hastie T. A Blockwise Descent Algorithm for Group-penalized Multiresponse and Multinomial Regression. arXiv preprint 2013.

(242) Huang J, Ma S, Xie H, Zhang C. A group bridge approach for variable selection. Biometrika 2009 06/01;96(2):339-355.

(243) Lin D, Zhang J, Li J, He H, Deng HW, Wang YP. Integrative analysis of multiple diverse omics datasets by sparse group multitask regression. Front Cell Dev Biol 2014 Oct 27;2:62.

(244) Chen LS, Hutter CM, Potter JD, Liu Y, Prentice RL, Peters U, et al. Insights into Colon Cancer Etiology via a Regularized Approach to Gene Set Analysis of GWAS Data. Am J Hum Genet 2010 04/28;86(6):860-871.

(245) Park H, Niida A, Miyano S, Imoto S. Sparse overlapping group lasso for integrative multi-omics analysis. J Comput Biol 2015 Feb;22(2):73-84.

(246) Li Q, Wang S, Huang CC, Yu M, Shao J. Meta-analysis based variable selection for gene expression data. Biometrics 2014 Dec;70(4):872-880.

(247) Gross SM, Tibshirani R. Data Shared Lasso: A novel tool to discover uplift. Comput Stat Data Anal 2016 9;101:226-235.

(248) Wang L, You Y, Lian H. A simple and efficient algorithm for fused lasso signal approximator with convex loss function. Computational Statistics 2013 08/01;28(4):1699-1714.

(249) Lander ES. Initial impact of the sequencing of the human genome. Nature 2011 02/09;470:187.

(250) Hoggart CJ, Whittaker JC, De Iorio M, Balding DJ. Simultaneous Analysis of All SNPs in Genome-Wide and Re-Sequencing Association Studies. PLOS Genetics 2008 07/25;4(7):e1000130.

(251) Meinshausen N, Bühlmann P. Stability selection. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 2010;72(4):417-473.

(252) Papachristou C, Ober C, Abney M. A LASSO penalized regression approach for genome-wide association analyses using related individuals: application to the Genetic Analysis Workshop 19 simulated data. BMC Proceedings 2016 10/18;10:221-226.

(253) Lu C, O'Connor GT, Dupuis J, Kolaczyk ED. Meta-Analysis for Penalized Regression Methods with Multi-Cohort Genome-Wide Association Studies. Hum Hered 2016;81(3):142-149.

(254) Kim S, Sohn KA, Xing EP. A multivariate regression approach to association analysis of a quantitative trait network. Bioinformatics 2009 Jun 15;25(12):i204-12.

(255) Shen X, Wang W, Wang L, Houde C, Wu W, Tudor M, et al. Identification of genes affecting apolipoprotein B secretion following siRNA-mediated gene knockdown in primary human hepatocytes. Atherosclerosis 2012 May;222(1):154-157.