DEPARTMENT OF STATISTICS 2017

Feature screening in ultrahigh-dimensional data

Candidate Number: 64466

Submitted for the Master of Science, London School of Economics, University of London

August 2017

Contents

List of Figures

List of Tables

Summary

High-dimensional data are gathered in various different fields, ranging from genomics to finance. A problematic feature across all of these data sets is that the number of covariates, p, is large relative to the the number of available observations, the sample size n. In this case, many of our finite sample inference tools, such as regression, cannot be applied. Consequently, there is a growing need to develop new statistical methodologies and techniques to analyse and interpret high-dimensional data.

Variable selection is a fundamental aspect of high-dimensional data analysis. Several procedures have been proposed in the literature, such as the lasso (Tibshirani, 1996) and the SCAD (Fan and Li, 2001). However, these methods are difficult to implement computationally, when p is very large (say, $\exp(O(n^\epsilon))$ for $\epsilon > 0$). In this new world of ultrahigh-dimensional data, alternative methods of variable selection must be considered.

The first aim of this project is to review the methods published for the feature screening of ultrahigh-dimensional data. For completion, we provide an overview of techniques that are used to handle linear models as well as those that are suitable in a generalised linear model context. The second aim of this project is to adapt the High-dimensional Ordinary Least-squares Projection (HOLP) method, first proposed by Wang and Leng (2015) for a linear setting, to a generalised linear model framework. In order to test the sample performance of the new GLM-HOLP approach, we set up simulation studies to explore both computational efficiency and screening accuracy. We also apply the new procedure to a gene expression data set.

Keywords: variable selection, feature screening, generalised linear models, high-dimensional ordinary least squares projection, ultrahigh-dimensional data

Chapter 1

Introduction

The technological revolution has brought with it an unprecedented volume of information. Typical data-intensive applications include genomics, climate satellites, high-frequency financial markets and brain imaging. As a result, there has been a growing demand from industry for competitive and efficient techniques to make sense of the 'big data' collected.

A high-dimensional data set usually consists of a collection of p features, $X_1, ..., X_p$, observed over a sample size of n. The problematic feature across all of these data sets is that the number of covariates, p, is large relative to the the number of available observations, the sample size n. Additionally, we also have to consider the statistical problem of how to model the relationship between the covariates and the response variable Y. In this case, many of our finite sample inference tools cannot be applied. For instance, when working with gene expression microarray data, the number of arrays may be in the order of a hundred, but the gene expression profiles could be up to the order of tens or hundreds of thousands. Consequently, there is a growing need to develop new statistical methodologies and machine learning techniques to analyse high-dimensional data.

In order to make sense of the scale of these data sets, the implementation of variable selection techniques has become a fundamental cornerstone of high-dimensional data analysis. It is incredibly important to select the most important features in the model, with the view to streamline interpretation, estimate model parameters, and provide a more parsimonious representation.

The most common type of high-dimensional data collected is linear in nature. This is also the case that has been studied the most by statisticians. One particularly popular class of methods is based off the notion of penalised variable selection. Several procedures in this family have been proposed, including the lasso (Tibshirani, 1996), the Dantzig selector (Candes and Tao, 2007) and the SCAD (Fan and Li, 2001). In the literature, these methods have garnered a lot of theoretical interest and several extensions have been proposed. While this family of 'one-stage' methods allows for simultaneous parameter estimation and variable selection, they are generally difficult to implement computationally, when p is very large (say, $\exp(O(n^{\epsilon}))$ for $\epsilon > 0$). This extreme case is what we refer to as ultrahigh-dimensional data. Within this new world, alternative methods of variable selection must be considered. These new procedures must be computationally efficient, stable and statistically accurate.

To meet the demand for techniques applicable to ultrahigh-dimensional data, Fan and Lv (2008) propose the sure independence screening (SIS) method. Within the context of least-squares regression, a feature's marginal utility can be defined as its marginal correlation with the response. Then, SIS selects those features ranked with the largest marginal utility. Under certain regularity conditions, Fan and Lv (2008) show that SIS obeys the *sure screening property*; namely, as $n \to \infty$, the probability that the chosen submodel does not miss any truly significant predictors tends to one. However, there are two instances where these regularity conditions may fail to choose the best submodel: first, if important features that are jointly correlated with the response, but marginally uncorrelated, are excluded, and second, if highly ranked features that are jointly uncorrelated with the response are chosen over other important covariates. To extend the SIS method to these cases, Fan and Lv outline an additional iterated sure independence screening (ISIS) procedure. As the scope of a linear model is limited, (I)SIS can also be extended to other parametric settings and this is developed fully in Fan et al. (2009). Here, Fan et al. consider a new general pseudo-likelihood framework for marginal utility ranking, which includes generalised linear models as a special case.

As an alternative to the original (I)SIS procedure outlined for linear models (Fan and Lv, 2008), Wang and Leng (2015) propose the High-dimensional Ordinary Least-squares Projection (HOLP) method, which is heavily inspired by the ordinary least-squares estimator. In this paper, a ridge regression extension of the HOLP estimator is also outlined. Theoretically, the authors also sucessfully prove that both the HOLP estimator and its ridge regression counterpart obey the sure screening property, without any limiting assumptions being placed on the marginal correlation of important variables.

In this project, we develop an extension of the High-dimensional Ordinary Least-squares Projection method (Wang and Leng, 2015) that is suitable for feature screening in a generalised linear model framework (GLM-HOLP). Like (I)SIS, the resulting procedure is straightforward and efficient to compute.

The thesis is organised as follows. In Chapter 2, we provide an overview of techniques that are used to handle feature screening in a linear setting. Chapter 3 presents an introduction to generalised linear models, a discussion of some of the relevant feature screening techniques, and outlines the GLM-HOLP procedure. We illustrate the performance of our proposed method via various numerical studies in Chapter 4. An analysis of a gene expression data set confirms the usefulness of the method. Finally, Chapter 5 provides concluding remarks and a discussion on future research.

Chapter 2

Feature Screening for Linear Models

In this chapter, we outline two categories of linear model techniques that are commonly used to handle high-dimensional data sets. The first category (of one-stage methods) conducts model selection and estimation simultaneously, by applying a shrinkage penalty to the regression coefficients. These techniques include the lasso (Tibshirani, 1996), the SCAD (Fan and Li, 2001), and the Dantzig selector (Candes and Tao, 2007). Usually, these methods are succesfully applied to a reduced feature space, where p is approximately $O(n)$. If p is much larger than n, the first class of approaches may start to be less effective. The second category of techniques (of two-stage methods), including the work put forward by Fan and Lv (2008), are designed to handle these ultrahigh-dimensional cases. Here, a two-stage approach consists of employing a specialised variable screener to initially reduce the dimensionality of the data to a manageable size, and then applying (if desired) a one-stage approach for further model refinement.

2.1 Background

2.1.1 Notation

Consider a response variable Y and a set of p features, $X_1, ..., X_p$. In the linear model, we assume that the relationship between the response and the features is explained by a $p \times 1$ parameter vector $\boldsymbol{\beta}$, such that,

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}. \tag{2.1}$$

Let $\mathbf{y} = (y_1, ..., y_n)^T \in \mathbb{R}^n$ be the response vector, $\mathbf{X} = (\mathbf{x_1}, ..., \mathbf{x_n})^T = (\mathbf{X_1}, ..., \mathbf{X_p})$ be an $n \times p$ design matrix, $\boldsymbol{\beta} = (\beta_1, ..., \beta_p)^T \in \mathbb{R}^p$ be the parameter vector, and $\boldsymbol{\epsilon} = (\epsilon_1, ..., \epsilon_n)^T \in \mathbb{R}^n$ be a vector of i.i.d random errors, such that $\boldsymbol{\epsilon} \sim MVN(\mathbf{0}, \sigma^2\mathbf{I}_n)$. Denote x_{ij} as the $i-$th observation of the j-th predictor. Thus, we write $\mathbf{x}_i = (x_{i1}, ..., x_{ip})^T$. Using this new notation, the linear regression model can be written alternatively as,

$$y_i = \mathbf{x}_i^T \boldsymbol{\beta} + \epsilon_i = \sum_{j=1}^{p} \beta_j x_{ij} + \epsilon_i \text{ for } i = 1, ..., n.$$

When the dimension of the data, p, is high, we tend to assume that only a handful of predictors out of $X_1, ..., X_p$ are important to the model and influence the response. This is equivalent to assuming that the coefficient vector $\boldsymbol{\beta}$ is *sparse*. Under the sparsity assumption, we can develop the notion of a *true model*, $\mathcal{M}*$, with size $s = |\mathcal{M}*|$, which contains of all of the significant covariates. For equation (2.1), the true model can be written as,

$$\mathcal{M}_* = \{j : \beta_j \neq 0, j = 1, ..., p\}, \tag{2.2}$$

with size $s < n$. In order to find an approximation of the true model, either one-stage or two-stage techniques can be used to select a subset of covariates. If a variable selection technique can also estimate the chosen model, we will want to set the coefficients of the irrelevant variables to zero. Define the chosen subset as the *selected model*, $\widehat{\mathcal{M}}$. The manner in which $\widehat{\mathcal{M}}$ is defined depends on the technique used and the context it is applied in (Liu et al., 2015).

2.1.2 The problem with least squares estimation

Traditionally, a linear regression model, as given by equation (2.1), is fitted using ordinary least squares estimation. The least squares estimator is given by,

$$\widehat{\boldsymbol{\beta}}_{OLS} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}. \tag{2.3}$$

In situations where $n \ll p$, the least squares estimator tends to perform well, with low variance. However, the least squares estimator is susceptible to the problem of multicollinearity (Friedman et al., 2009; James et al., 2013).

When multicollinearity occurs, the parameter estimates themselves are unbiased, but their variances are very large. Unsurprisingly, as the value of p grows, the least squares approach begins to perform poorly. An ‘over-defined’ model (i.e. too many variables) is a source of multicollinearity and the estimates will require large prediction intervals, in order to capture the variability of coefficient estimates. When $p > n$, the matrix $\mathbf{X}^T\mathbf{X}$ becomes degenerate and the least squares estimator cannot be solved for uniquely. The breakdown of the least squares approach in high dimensions has led to the development of alternative fitting procedures for high-dimensional data. James et al. (2013) point out that there are two important points to consider when deciding upon a linear model fitting procedure: model interpretability and prediction accuracy.

Variable selection techniques can be used to improve the interpretability of a proposed fitting procedure and provide a parsimonious representation. A plethora of well-established methods will be discussed in this chapter. In addition, *shrinkage* techniques can be used to improve the accuracy of coefficient estimates. This type of approach (also commonly known as *regularisation*) entails fitting a penalised linear model to all p covariates. The penalty term will have the effect of shrinking the parameter estimates down toward zero, controlling the variance of the fitted model. Certain shrinkage techniques have the added advantage of performing variable selection (James et al., 2013).

2.1.3 An aside on ridge regression

Hoerl and Kennard (1970) propose to resolve the potential instability in the least squares estimator by adding a ridge parameter, $\lambda \geq 0$, to the diagonal elements of the matrix, $\mathbf{X}^T\mathbf{X}$, before inverting it. The resulting estimator is known as the *ridge regression* estimator,

$$\widehat{\boldsymbol{\beta}}(\lambda) = (\lambda \mathbf{I}_p + \mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}. \tag{2.4}$$

Typically, the inputs are standardised before solving for the ridge regression estimator.

We can additionally think of ridge regression as a type of regularisation technique. The inclusion of the ridge parameter places an ℓ_2-penalty on the parameter estimates. Thus, we can reformulate equation (2.4) as,

$$\widehat{\boldsymbol{\beta}}(\lambda) = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}}\{\sum_{i=1}^{n}(y_i - \sum_{j=1}^{p}\beta_j x_{ij})^2 + \lambda\sum_{j=1}^{p}\beta_j^2\} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}}||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||_2^2 + \lambda||\boldsymbol{\beta}||_2^2,$$

where λ is determined separately (Friedman et al., 2009; James et al., 2013).

In this alternative formulation, the first term is the residual sum of squares (RSS) and the second term represents the shrinkage penalty applied to $\boldsymbol{\beta}$. The ridge parameter, λ, controls the relative influence of the two components. If $\lambda = 0$, the ridge regression estimator collapses to the ordinary least squares estimator provided in (2.3). As λ grows, the impact of the penalty term grows and the coefficient estimates shrink to zero. It is important to note, however, that each new value of λ produces a different set of parameter estimates. Therefore, it is vital that we carefully choose an apppropriate value of λ. Luckily, cross-validation provides a quick and easy solution to this issue.

Overall, the main disadvantage of ridge regression is that it includes all p variables in the final model. While estimation accuracy is not directly affected by this, it is difficult to obtain a clear interpretation of the model in a high-dimensional setting. The penalty term may shrink the value of the coefficients, but it does not actually set any of them to zero (unless $\lambda = \infty$). Therefore, ridge regression cannot be used as a tool for variable selection. In the next section, we discuss various penalised likelihood techniques, such as the lasso and its variants (Tibshirani, 1996), which offer the options of variable selection and shrinkage estimation.

2.2 One-stage methods

In this section, we discuss one-stage methods. Penalisation is a commonly used idea in variable selection. This class of techniques conducts model selection and estimation simultaneously, by applying a shrinkage penalty to the regression coefficients. We review the lasso (Tibshirani, 1996), the SCAD (Fan and Li, 2001), and the Dantzig selector (Candes and Tao, 2007) in detail. For all of these one-stage methods, the selected submodel, $\widehat{\mathcal{M}}$, is defined as, $\{j : \widehat{\beta}_j \neq 0, j = 1, ..., p\}$.

2.2.1 Penalised likelihood

The generalised form of the penalised likelihood, as defined by Fan and Lv (2010), can be written as,

$$n^{-1}\ell_n(\boldsymbol{\beta}) - \sum_{j=1}^{p} p_\lambda(|\beta_j|), \tag{2.5}$$

where $\ell_n(\boldsymbol{\beta})$ is the log-likelihood function and $p_\lambda(.)$ is the penalty function, indexed via the regularisation parameter λ. In the literature, a great amount of work has been conducted to develop a unified framework for feature selection, using the penalised likelihood as a starting point. By maximising the penalised likelihood function from (2.5), we hope to be able to simultaneously select significant variables and estimate their corresponding regression coefficients.

The ordinary least squares estimate is obtained by minimising $||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||_2^2$ over $\boldsymbol{\beta} \in \mathbb{R}^p$. The penalised likelihood given in (2.5) is equivalent (up to an affine transformation) to a form of penalised least squares (PLS) regression problem,

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^p} \{\frac{1}{2n}||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||_2^2 + \sum_{j=1}^{p} p_\lambda(|\beta_j|)\}. \tag{2.6}$$

Several penalties have been introduced for shrinkage estimation and variable selection (Fan and Lv, 2010). Often, various cases of the ℓ_q-penalty are used, where $p_\lambda(|\beta|) = \lambda|\beta|^q$ for $0 < q \leq 2$. For instance, penalised ℓ_1-regression is equivalent to the the lasso (Tibshirani, 1996) in a linear model setting, although it can also be identified as a specific case of the penalised ℓ_1-likelihood. Similarly, penalised ℓ_2-regression is equivalent to ridge regression. However, as discussed in the previous section, the ℓ_2-penalty is not suitable for feature selection. In general, Fan and Li (2001) prefer penalty functions that provide estimators, possessing the following three properties:

1. **Sparsity:** The estimator automatically sets small estimated coefficients to zero, in order to select variables and reduce the complexity of the model.
2. **Unbiasedness:** The estimator is almost unbiased, especially if the true unknown coefficients are large.
3. **Continuity:** The estimator is continuous in the data, with the aim of avoiding instability in model prediction.

Fan and Li also require $p_\lambda(|\beta|)$ to be nondecreasing in $|\beta|$. Note that all of these penalised likelihood methods will produce an entire path of solutions, for various values of λ. Usually, statisticians use k-fold cross validation to select the most appropriate value of λ.

2.2.2 The lasso and its variants

Tibshirani (1996) proposes the lasso, otherwise known as the *least absolute shrinkage and selection operator*, as a shrinkage and variable selection technique. As discussed briefly above in Section 2.2.1, the lasso applies an ℓ_1-penalty to the parameters in the linear regression model. Tibshirani defines the lasso estimate by,

$$\widehat{\boldsymbol{\beta}}_{lasso} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}}\Big\{\sum_{i=1}^{n}(y_i - \sum_{j=1}^{p}\beta_j x_{ij})^2 + \lambda\sum_{j=1}^{p}|\beta_j|\Big\}, \tag{2.7}$$

where λ is determined separately (Friedman et al., 2009; James et al., 2013). This can be equivalently written as,

$$\widehat{\boldsymbol{\beta}}_{lasso} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}}||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||_2^2 + \lambda||\boldsymbol{\beta}||_1.$$

Computationally, the lasso solution is a quadratic programming problem, with the same operational cost as ridge regression. The penalty term, λ, should be chosen in a such a way as to minimise the expected prediction error.

The elastic-net penalty

From above, we know that the general version of the ℓ_q-penalty can be written as $p_\lambda(|\beta|) = \lambda|\beta|^q$ for $0 < q \leq 2$. For $q > 1, |\beta|^q$ is now differentiable at the origin. Therefore, any ℓ_q-penalty function with $q > 1$ is not able to set parameter coefficients exactly to zero. For this reason, Zou and Hastie (2005) propose the *elastic-net* penalty,

$$\lambda\sum_{j=1}^{p}((1-\alpha)\beta_j^2 + \alpha|\beta_j|), \tag{2.8}$$

where $\alpha \in [0, 1]$. The elastic-net is a mixture of the lasso and ridge regression, while retaining the best of both techniques. The first term shrinks together the coefficients of correlated covariates. Whereas, the second term allows for a sparse solution in the parameter vector (Friedman et al., 2009).

The adaptive lasso

The lasso penalty is widely used because of its convexity. However, it generates model bias. To overcome this problem, Zou (2006) develops the *adaptive lasso*. The idea in this variant of the lasso is to use a weighted version of the ℓ_1-penalty for the penalisation term,

$$\lambda \sum_{j=1}^{p} \omega_j |\beta_j|, \tag{2.9}$$

where λ is the shrinkage parameter and $\boldsymbol{\omega} = (\omega_1, ..., \omega_d)^T$ is a weight vector. Zou (2006) also suggests using $\omega_j = 1/|\widehat{\beta}_j|^\nu$, where $\widehat{\boldsymbol{\beta}}$ is the ordinary least squares estimate and $\nu > 0$ (Friedman et al., 2009; Fan and Lv, 2008).

2.2.3 The SCAD penalty

Recall the three conditions listed in Section 2.2.1, proposed by Fan and Li (2001), for penalised least squares estimators. It is widely known that the ℓ_q-penalty fails the sparsity condition for values of $q > 1$ and that it also fails the continuity condition for $0 \leq q < 1$. Therefore, it holds that none of the ℓ_q-penalty functions are capable of satisfying all three of the conditions at the same time. To remedy this, Fan and Li propose a continously differentiable alternative, known as the *smooth clipped absolute deviation* (SCAD) penalty function. The derivative of the SCAD penalty is defined as,

$$p'_\lambda(|\theta|) = \lambda\big\{I(|\theta| \leq \lambda) + \frac{(a\lambda - |\theta|)_+}{(a-1)\lambda} I(|\theta| > \lambda)\big\} \text{ for some } a > 2. \tag{2.10}$$

Fan and Li advise setting $a = 3.7$. The main difference between the SCAD and the lasso is that larger coefficients are shrunken less severely.

The *minimum concave penalty* (MCP), proposed by Zhang (2007), is another viable alternative to the ℓ_q-penalty. The derivative of the MCP, given by,

$$p'_\lambda(|\theta|) = \frac{(a\lambda - |\theta|)_+}{a},$$

shifts the flat portion of the derivative of the SCAD penalty to the origin. Both the SCAD penalty and the MCP satisfy the three conditions outlined above by Fan and Li (2001).

2.2.4 Dantzig selector

Candes and Tao (2007) propose a slightly modified version of the lasso. The *Dantzig selector* obtains the solution $\widehat{\boldsymbol{\beta}}_{DS}$ to the problem,

$$\min_{\boldsymbol{\beta}\in\mathbb{R}^p} ||\boldsymbol{\beta}||_1, \text{ subject to } ||\mathbf{X}^T(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})||_\infty \leq \lambda, \tag{2.11}$$

where λ is the tuning parameter. Note, $||.||_\infty$ is the ℓ_∞-norm, which takes the maximum absolute value of the components of a vector. The condition above can be rewritten as an ℓ_1-regularisation problem,

$$\min_{\boldsymbol{\beta}\in\mathbb{R}^p} \{||\mathbf{X}^T(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})||_\infty\}, \text{ subject to } ||\boldsymbol{\beta}||_1 \leq \lambda.$$

In this form, the Dantzig selector starts to resemble the lasso. Let's assume that $p > n$. As the value of the regularisation parameter grows, the Dantzig selector obtains a least squares solution with an ℓ_1-norm, just like the lasso. However, for small values of λ, the Dantzig selector obtains a solution path that is different to that of the lasso (Friedman et al., 2009).

2.3 Two-stage methods

In a two-stage procedure, we are interested in reducing the dimensionality of the data from p, which is usually of a large scale (say, $\exp(O(n^\epsilon))$ for some $\epsilon > 0$), to some relatively smaller value, d, of $O(n)$. In the first stage, we can use a feature screener to select a submodel, $\widehat{\mathcal{M}}$, of size d. If desired, we can then apply any of the one-stage approaches outlined in the previous section to the reduced feature space for further second-stage analysis. We will discuss four appropriate feature screening techniques: SIS (Fan and Lv, 2008), the HOLP procedure (Wang and Leng, 2015), tilting (Cho and Fryzlewicz, 2012) and forward regression (Wang, 2009).

2.3.1 Sure screening property

Recall the definitions of the true model, $\mathcal{M}_*$, and the selected model, $\widehat{\mathcal{M}}$. Under certain regularity conditions, the *sure screening property* states that as $n \to \infty$,

$$P(\mathcal{M}_* \subset \widehat{\mathcal{M}}) \to 1. \tag{2.12}$$

This property, initially developed by Fan and Lv (2008), ensures that all of the important features are retained after screening. Thus, any one-stage method employed on a screened submodel would be much more effective. In general, all proposed feature screeners should uphold this result.

2.3.2 Sure independence screening

By introducing the seminal theory of *sure independence screening* (SIS), Fan and Lv (2008) are the first to truly pioneer feature screening techniques in an ultrahigh-dimesional data setting. Let us initally motivate SIS from a ridge regression perspective. Recall equation (2.4),

$$\widehat{\boldsymbol{\beta}}(\lambda) = (\lambda \mathbf{I}_p + \mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y},$$

where λ is the ridge parameter. We centre all of the variables with mean 0 and ensure that they are scaled to have variance 1. If $\lambda \to 0$, it is observed that $\widehat{\boldsymbol{\beta}}(\lambda)$ becomes the ordinary least squares estimator, provided that it is non-degenerate. In comparison, if $\lambda \to \infty$, $\lambda\widehat{\boldsymbol{\beta}}(\lambda)$ becomes $\mathbf{X}^T\mathbf{y}$. This implies that the ridge regression estimator is proportional to $\mathbf{X}^T\mathbf{y}$. Thus, $\frac{1}{n}\mathbf{X}^T\mathbf{y}$ is the vector of the sample marginal correlations, between the p features in the data and the response vector. Inspired by this idea, Fan and Lv (2008) propose the use of sample correlation as a tool for screening features.

Let $\boldsymbol{\omega} = (\omega_1, ..., \omega_p)^T$ be the $p \times 1$ vector obtained via componentwise regression. In other words, we set $\boldsymbol{\omega} = \mathbf{X}^T\mathbf{y}$. So, $\boldsymbol{\omega}$ is the vector of marginal correlations of the predictors with the response variable, scaled by the standard deviation of the response. We are interested in using this information to select d predictors. For a given threshold $\gamma \in (0, 1)$, we rank the marginal correlations of the predictors, in order to select the following submodel,

$$\widehat{\mathcal{M}}_\gamma = \{1 \leq j \leq p : |\omega_j| \text{ is among the largest } [\gamma n] \text{ of them all}\}.$$

The success of the screening procedure depends on whether $\widehat{\mathcal{M}}_\gamma$ contains all of the important variables from the true model, $\mathcal{M}_*$. Theoretically, Fan and Lv are able to demonstrate that the SIS procedure obeys the sure screening property. In order to prove this, a sufficient condition is placed on the data, which requires the marginal correlations of the important predictors to be bounded away from zero. This assumption is henceforth referred to as the *marginal correlation* assumption.

If the marginal correlation condition is not valid, either of these two problems may occur:

1. If an important covariate, X_j, for some $j \in \{1, ..., p\}$, is jointly correlated but marginally uncorrelated with the response, the SIS procedure may fail to recruit it.

2. If there is an unimportant covariate, X_j, that is jointly uncorrelated with the response, but has a high marginal correlation, the SIS procedure may accidentally select it.

To address these screening issues, Fan and Lv (2008) propose an iterative SIS procedure (ISIS), which repeatedly replaces the response variable with the residual obtained from fitting the model selected in the previous step. For the sake of brevity, we do not include the specific details of the ISIS algorithm here. Instead, a generalisation of the (I)SIS procedure, proposed by Fan et al. (2009), is discussed in detail in Section 3.4.1. Further generalisations of correlation learning are discussed in the detailed review published by Liu et al. (2015).

2.3.3 High-dimensional Ordinary Least Squares Projection

As an alternative to sure independence screening, Wang and Leng (2015) propose a variable screener inspired by ordinary least squares (OLS) estimation. To discuss the idea behind the procedure, let us initially explore a general class of estimates of $\boldsymbol{\beta}$, where,

$$\tilde{\boldsymbol{\beta}} = \mathbf{A}\mathbf{y},$$

such that $\mathbf{A} \in \mathbb{R}^{p \times n}$. Here, $\mathbf{A}$ acts as the 'screening' matrix, mapping the response vector to the parameter estimates. In the original SIS procedure, Fan and Lv (2008) set the screening matrix to be $\mathbf{X}^T$. Even if $\tilde{\boldsymbol{\beta}}$ is not an accurate estimator, it should at least preserve the rank order of the entries in the original parameter vector, $\boldsymbol{\beta}$. This is to ensure that the largest entries in $\tilde{\boldsymbol{\beta}}$ correspond to the nonzero entries in $\boldsymbol{\beta}$.

If we expand the expression given above, we can see that,

$$\tilde{\boldsymbol{\beta}} = \mathbf{A}\mathbf{y} = \mathbf{A}(\mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}) = (\mathbf{A}\mathbf{X})\boldsymbol{\beta} + \mathbf{A}\boldsymbol{\epsilon}.$$

Note that, $(\mathbf{A}\mathbf{X})\boldsymbol{\beta}$ is the signal term and that $\mathbf{A}\boldsymbol{\epsilon}$ is the noise term. Ideally, the signal part of the estimator should be amplified as much as possible, relative to a stochastically negligible noise term. In order to do this, Wang and Leng (2015) suggest that $\mathbf{A}\mathbf{X}$ should ideally be the identity matrix, $\mathbf{I}$, thereby allowing the rank order to be maintained.

This discussion leads us to naturally consider using some kind of inverse of $\mathbf{X}$ for the screening matrix. If $p < n$, the easiest solution is to set $\mathbf{A} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$, giving us the ordinary least squares estimator. However, in a high-dimensional setting, $\mathbf{X}^T\mathbf{X}$ is a degenerate matrix. Instead, Wang and Leng propose the use of a generalised inverse of $\mathbf{X}$.

When $p > n$, we can use the Moore-Penrose generalised inverse of $\mathbf{X}$, such that $\mathbf{A} = \mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}$. While $\mathbf{A}\mathbf{X}$ is now no longer an identity matrix, it is still diagonally dominant. This will ensure that the rank order of $\boldsymbol{\beta}$ is more or less preserved. Under various simulation settings, Wang and Leng demonstrate that diagonal dominance exists for $\mathbf{A}\mathbf{X} = \mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{X}$, under a variety of different correlation structures.

Wang and Leng (2015) define the *High-dimensional Ordinary Least Squares Projection*

(HOLP) estimator as,

$$\widehat{\boldsymbol{\beta}} = \mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{y}. \tag{2.13}$$

For variable screening, the components of $\widehat{\boldsymbol{\beta}}$ are ranked in descending absolute value and the largest are selected. Specifically, the procedure selects the d variables corresponding to the d largest coefficients. The submodel chosen is,

$$\widehat{\mathcal{M}}_d = \{j : |\hat{\beta}_j| \text{ are among the largest } d \text{ of all coefficients, for } j = 1, ..., p\}.$$

The HOLP procedure can also be derived from a ridge regression perspective. Recall the ridge regression estimator from equation (2.4),

$$\widehat{\boldsymbol{\beta}}(\lambda) = (\lambda\mathbf{I}_p + \mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y},$$

where λ is the ridge parameter. As discussed in Section 2.3.2, if we let $\lambda \to \infty$, $\lambda\widehat{\boldsymbol{\beta}}(\lambda) \to \mathbf{X}^T\mathbf{y}$. Whereas, if $\lambda \to 0$, the ridge regression estimator tends to,

$$(\mathbf{X}^T\mathbf{X})^{+}\mathbf{X}^T\mathbf{y},$$

where $\mathbf{A}^+$ denotes the Moore-Penrose generalised inverse. Applying the Sherman-Morrison-Woodbury formula, we obtain,

$$(\lambda\mathbf{I}_p + \mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} = \mathbf{X}^T(\lambda\mathbf{I}_p + \mathbf{X}\mathbf{X}^T)^{-1}\mathbf{y}.$$

Thus, the ridge version of HOLP is $\mathbf{X}^T(\lambda\mathbf{I}_p + \mathbf{X}\mathbf{X}^T)^{-1}\mathbf{y}$. Then, if we let $\lambda \to 0$, we obtain the HOLP estimator from equation (2.13). For the full theoretical derivation of the ridge HOLP estimator, please see the appendix of Wang and Leng (2015).

Within their paper, Wang and Leng provide various asymptotic properties of their proposed procedure. Under mild conditions, it is possible to prove the sure screening property. In addition, the authors are able to establish the *screening consistency* of HOLP estimator and its ridge regression version. Screening consistency ensures that the important and unimportant variables can be separated from each other, by simply using the values of their corresponding coefficients in $\widehat{\boldsymbol{\beta}}$.

Overall, there are three distinct advantages of the HOLP approach (Wang and Leng, 2015). First, it is a computationally efficient procedure. The computional complexity of HOLP is of $O(n^2p)$, while SIS is $O(np)$. Second, the HOLP procedure has scale invariance in the signal term, $\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{X}\boldsymbol{\beta}$. In comparison, SIS does not have scale invariance in its corresponding signal term, $\mathbf{X}^T\mathbf{X}\boldsymbol{\beta}$, which implies that variable scaling can impact its performance. Third, the proof of the screening property for the HOLP procedure does not rely upon the marginal correlation assumption. This assumption is often violated in high-dimensional data. By removing the need for the marginal correlation assumption, the HOLP approach is not susceptible to the problems associated with the original SIS procedure.

2.3.4 Alternative iterative procedures

Building upon SIS (Fan and Lv, 2008), there are several other feature screeners built for ultrahigh-dimensional data that compete with the HOLP procedure. Indeed,Wang and Leng compare the sample performance of some of these alternative screeners to their own approach in extensive numerical studies. We briefly discuss two of these techniques.

Wang (2009) proposes an extension of the classical variable screening method, *forward regression* (FR), to the ultrahigh-dimensional setting. Starting initially with an empty set, FR sequentially introduces covariates into the model that most improve the fit. As a greedy algorithm, the FR procedure produces a nested sequence of models. Theoretical analysis demonstrates that FR is able to identify relevant predictors consistently (i.e. sure screening), even in cases where p is significantly larger than n. In particular, it is shown that if the size of the true model, $\mathcal{M}_*$, is finite, the algorithm can recover all of the important covariates within a finite number of steps. To select the best candidate from the nested models produced by the algorithm, Wang makes use of the extended BIC criterion (Chen and Chen, 2008).

Cho and Fryzlewicz (2012) propose a novel way to study the contribution of each covariate in the data to the response, taking account of the existing correlation structure within the data. To do this, the authors define the notion of *tilted correlation* to measure the relationship between each covariate and the response. Each column, X_j of the design matrix, for $j = 1, ..., p$, can be transformed or *tilted* (so that it becomes X_j^*), in order to reduce the impact of other covariates X_k, for $k \neq j$, on the tilted correlation. In the final procedure, Cho and Fryzlewicz provide an adaptive choice between using marginal correlation or tilting correlation to investigate the impact of each variable. Theoretical analyis demonstrates that, under mild conditions, the tilting correlation measure can successfully identify important and unimportant variables in a data set, making it a useful tool for variable selection. The authors also develop an iterative procedure for tilting.

Chapter 3

Feature Screening for Generalised Linear Models

In this chapter, we formally introduce generalised linear models (GLMs) and consider some of the relevant variable selection techniques for a high-dimensional setting. The primary purpose of this chapter is to extend the High-dimensional Ordinary Least Squares Projection (HOLP) estimator (Wang and Leng, 2015) to the generalised linear model framework. Our proposed procedure is known as the GLM-HOLP method. To develop the theory, we use Agresti (2015) and McCullagh and Nelder (1996) as our primary references.

3.1 Introduction to GLMs

3.1.1 Components of a GLM

Generalised linear models (GLMs) are extensions of standard linear regression models, designed to cover a wide range of response distributions. Let us consider a vector of n independent observations, $\mathbf{y} = (y_1, ..., y_n)^T$, which are the sample realisations of the response variable Y. We can provide a three-part specification of a generalised linear model (Agresti, 2015; McCullagh and Nelder, 1996).

1. **The random component**
   This consists of n independent realisations, $(y_1, ..., y_n)^T$, of the response variable Y. The observations are drawn from a distribution in the exponential family.

2. **The systematic component**
   Given an observation i, we denote x_{ij} to be the realised value of the covariate X_j, for each of $j = 1, ..., p$. The *linear predictor* relates the covariates $X_1, ..., X_p$ to the coefficient vector, $\boldsymbol{\beta}$, through the formula,

$$\eta_i = \sum_{j=1}^{p} \beta_j x_{ij} = \mathbf{x}_i^T \boldsymbol{\beta} \text{ for } i = 1, ..., n. \tag{3.1}$$

In matrix form, we can express the linear predictor as,

$$\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\beta}, \tag{3.2}$$

where $\boldsymbol{\eta} = (\eta_1, .., \eta_n)^T$, $\boldsymbol{\beta} = (\beta_1, .., \beta_p)^T$ is a parameter vector, and $\mathbf{X}$ is an $n \times p$ design matrix of the realised values $\{x_{ij}\}$. For simplicity, it is common to treat Y_i as random and the vector $\mathbf{x}_i = (x_{i1}, ..., x_{ip})^T$ as fixed. As a result, the linear predictor is often referred to as the *systematic component*. In practice, $\mathbf{x}_i$ is often random, and inferences about covariates need to be made conditional on the observed values.

3. **The link function**

   The link function brings together the random and systematic parts of the model. For a given observation i, let's consider the mean, $\mu_i = E(Y_i)$. The model connects η_i (as specified in (3.1)) to μ_i, through a specified function $g(.)$, which is differentiable and monotonic. In other words, the link function,

$$g(\mu_i) = \eta_i = \sum_{j=1}^{p} \beta_j x_{ij}. \tag{3.3}$$

3.1.2 The exponential dispersion family

In a generalised linear model, we draw independent observations $\mathbf{y} = (y_1, ..., y_n)^T$ from the same exponential family distribution. We say that Y_i is of *exponential dispersion family* form, if its probability density can be written in the following manner,

$$f(y_i; \theta_i, \phi) = \exp\left(\frac{(y_i\theta_i - b(\theta_i))}{a(\phi)} + c(y, \phi)\right). \tag{3.4}$$

Here, θ_i is the canonical parameter and ϕ is the dispersion parameter. The functions $a(.)$, $b(.)$ and $c(.)$ are all known. Often, GLMs take $a(\phi) = \phi$. In general, expressions for E(Y_i) and Var(Y_i) can be found in terms of the parameters from equation (3.4). To save space, the derivations of the following properties are excluded. We know that,

$$\mu_i = \mathrm{E}(Y_i) = b'(\theta_i), \text{ for } i = 1, ..., n, \text{ and} \tag{3.5}$$

$$Var(Y_i) = a(\phi)b''(\theta_i) = a(\phi)\nu(\mu_i), \text{ for } i = 1, ..., n. \tag{3.6}$$

3.1.3 The canonical link function of a GLM

As mentioned above, the link function of a GLM connects the random component and the linear predictor. The link function that maps the mean, μ_i, to the canonical parameter, θ_i, is called the canonical link. From equation (3.5), we know that $\theta_i = \theta_i(\mu_i)$. So, the direct relationship can be written as,

$$\theta_i(\mu_i) = g(\mu_i) = \sum_{j=1}^{p} \beta_j x_{ij}, \text{ for } i = 1, ..., n. \tag{3.7}$$

3.1.4 GLMs for Normal, Binomial, and Poisson responses

We continue to illustrate the exponential dispersion family by discussing the three most popular distributions - namely the Normal, the Poisson, and the Binomial - and the generalised linear models associated with them (Agresti, 2015; McCullagh and Nelder, 1996).

Normal Linear Model

The class of GLMs includes models for continuous response variables. These models assume a Normal distribution for the random component, such that $Y_i \sim N(\mu_i, \sigma^2)$ for $i = 1, ..., n$. For the Normal distribution, observation i has probability density function,

$$\begin{aligned} f(y_i; \mu_i, \sigma^2) &= \frac{1}{\sigma\sqrt{2\pi}} \exp\Big(- \frac{(y_i - \mu_i)^2}{2\sigma^2}\Big) \\ &= \exp\Big(\frac{y_i\mu_i - \frac{1}{2}\mu_i^2}{\sigma^2} - \frac{1}{2}\log(2\pi\sigma^2) - \frac{y_i^2}{2\sigma^2}\Big). \end{aligned}$$

We can see that this satisfies the exponential dispersion family form, with the canonical parameter set as $\theta_i = \mu_i$. We know that,

$$b(\theta_i) = \frac{1}{2}\mu_i^2 = \frac{1}{2}\theta_i^2, a(\phi) = \sigma^2, \text{ and } c(y_i, \phi) = -\frac{1}{2}\log(2\pi\sigma^2) - \frac{y_i^2}{2\sigma^2}.$$

So,

$$E(Y_i) = b'(\theta_i) = \theta_i \text{ and } Var(Y_i) = b''(\theta_i)a(\phi) = \sigma^2.$$

As $\theta_i(\mu_i) = \mu_i$, we know that the canonical link function is the identity link, $g(\mu_i) = \mu_i$. Thus, $\eta_i = \mu_i$. A GLM that uses the identity link function is the *linear model*. Therefore we can define the linear model as,

$$\mu_i = \sum_{j=1}^{p} \beta_j x_{ij} \text{ for } i = 1, ..., n. \tag{3.8}$$

If we assume that $\epsilon_i \sim N(0, \sigma^2)$ for $i = 1, ...n$, the linear model can alternatively be written as,

$$y_i = \mathbf{x}_i^T \boldsymbol{\beta} + \epsilon_i \text{ for } i = 1, ..., n.$$

Binomial Logistic Model

Let us assume that $R_i = n_i Y_i$ has a $Bin(n_i, \pi_i)$ distribution. Thus, $Y_i = \frac{R_i}{n_i} = \frac{Bin(n_i,\pi_i)}{n_i}$ is the sample proportion of successes, such that $E(Y_i) = \pi_i$. For the Binomial distribution, observation i has probability mass function,

$$f(y_i; \pi_i) = \binom{n_i}{n_i y_i} \pi_i^{n_i y_i}(1 - \pi_i)^{n_i - n_i y_i},$$

for $y = 0, \frac{1}{n_i}, ..., 1$. The density function can be rewritten as,

$$f(y_i; \pi_i) = \exp\Big(\frac{y_i \log(\frac{\pi_i}{1-\pi_i}) + \log(1-\pi_i)}{1/n_i} + \log\binom{n_i}{n_i y_i}\Big).$$

We set the canonical parameter to be $\theta_i = \log(\frac{\pi_i}{1-\pi_i})$. So in terms of θ_i,

$$f(y_i; \theta_i) = \exp\Big(\frac{y_i \log(\theta_i) - \log(1 + exp(\theta_i))}{1/n_i} + \log\binom{n_i}{n_i y_i}\Big).$$

Thus, the binomial density function takes the required exponential family form. Then, we know that,

$$b(\theta_i) = \log(1 + \exp(\theta_i)), a(\phi) = \frac{1}{n_i}, \text{ and } c(y_i, \phi) = \log\binom{n_i}{n_i y_i}.$$

So,

$$E(Y_i) = b'(\theta_i) = \frac{\exp(\theta_i)}{1 + \exp(\theta_i)} \text{ and } Var(Y_i) = b''(\theta_i)a(\phi) = \frac{\exp(\theta_i)}{(1 + \exp(\theta_i))^2 n_i} = \frac{\pi_i(1-\pi_i)}{n_i}.$$

For the rest of the chapter, we assume that each Y_i is a binary response variable (i.e. $n_i = 1$). As $\theta_i(\pi_i) = \log(\frac{\pi_i}{1-\pi_i})$, we know that the canonical link function is the logit link. Thus, $\eta_i = \log(\frac{\pi_i}{1-\pi_i})$. A GLM that uses the logit link is the *logistic regression model*. This GLM can be defined as,

$$\text{logit}(\pi_i) = \log(\frac{\pi_i}{1-\pi_i}) = \sum_{j=1}^{p} \beta_j x_{ij} = \mathbf{x}_i^T \boldsymbol{\beta} \text{ for } i = 1, ..., n. \tag{3.9}$$

Alternatively, the model can be written in terms of the binary response probability,

$$\pi_i = \frac{\exp(\theta_i)}{1 + \exp(\theta_i)} = \frac{\exp(\sum_{j=1}^{p} \beta_j x_{ij})}{1 + \exp(\sum_{j=1}^{p} \beta_j x_{ij})} \text{ for } i = 1, ..., n. \tag{3.10}$$

Poisson Loglinear Model

Some response variables have counts as their possible outcomes. Let us assume that Y_i has a $Poisson(\mu_i)$ distribution. For the Poisson distribution, observation i has probability mass function,

$$f(y_i; \mu_i) = \frac{\exp(-\mu_i)\mu_i^{y_i}}{y_i!} = \exp(y_i \log(\mu_i) - \mu_i - \log(y_i!)) \text{ for } y_i = 0, 1, 2, ...\text{etc.}$$

We set the canonical parameter to be $\theta_i = \log(\mu_i)$. In terms of θ_i,

$$f(y_i; \theta_i) = \exp(y_i\theta_i - \exp(\theta_i) - \log(y_i!)),$$

As the above function takes the required exponential family form, we know that,

$$b(\theta_i) = \exp(\theta_i), a(\phi) = 1, \text{ and } c(y_i, \phi) = -\log(y_i!).$$

So,

$$E(Y_i) = b'(\theta_i) = \exp(\theta_i) = \mu_i \text{ and } Var(Y_i) = b''(\theta_i)a(\phi) = \exp(\theta_i) = \mu_i.$$

Since $\theta(\mu_i) = \log(\mu_i)$, we know that the canonical link function is the log link. Thus, $\eta_i = \log(\mu_i)$. A GLM that uses the log link is the *Poisson loglinear model.* This GLM can be defined as,

$$\log(\mu_i) = \sum_{j=1}^{p} \beta_j x_{ij} = \mathbf{x}_i^T \boldsymbol{\beta} \text{ for } i = 1, ..., n. \tag{3.11}$$

3.2 Estimation using likelihood methods ($p < n$)

In traditional modelling, the parameter vector $\boldsymbol{\beta}$ is generally estimated using maximum likelihood methods. The idea is to simultaneously solve p score equations to obtain $\boldsymbol{\beta}$. However, this can be difficult to accomplish analytically. Typically, the score equations are numerically solved using the Fisher Scoring method, which can be easily transformed into an iterative weighted least squares algorithm (Agresti, 2015; McCullagh and Nelder, 1996).

3.2.1 Maximum likelihood estimation

For n independent observations, we know that,

$$l(\boldsymbol{\beta}) = \sum_{i=1}^{n} l_i = \sum_{i=1}^{n} \log f(y_i; \theta_i, \phi) = \sum_{i=1}^{n} \frac{(y_i\theta_i - b(\theta_i))}{a(\phi)} + \sum_{i=1}^{n} c(y_i, \phi). \tag{3.12}$$

For a GLM such that $\eta_i = \sum_{j=1}^{p} \beta_j x_{ij}$, the score equations are,

$$s_j = \sum_{i=1}^{n} \frac{\partial l_i}{\partial \beta_j} = 0, \text{ for } j = 1, .., p. \tag{3.13}$$

Using the chain rule, we know that,

$$s_j = \sum_{i=1}^{n} \frac{\partial l_i}{\partial \beta_j} = \sum_{i=1}^{n} \frac{\partial l_i}{\partial \theta_i} \frac{d\theta_i}{d\mu_i} \frac{d\mu_i}{d\eta_i} \frac{\partial \eta_i}{\partial \beta_j}.$$

We now derive expressions for each of the derivatives in the expression for the j-th score equation. First, we know that,

$$\frac{\partial l_i}{\partial \theta_i} = \frac{(y_i - \mu_i)}{a(\phi)}.$$

As $\mu_i(\theta_i) = b'(\theta_i)$ and $Var(Y_i) = b'(\theta_i)a(\phi)$,

$$\frac{d\mu_i}{d\theta_i} = b''(\theta_i) = V_i,$$

where $V_i = \frac{Var(Y_i)}{a(\phi)}$. Thus,

$$\frac{d\theta_i}{d\mu_i} = \frac{1}{V_i}.$$

Also, since $\eta_i = \sum_{j=1}^{p} \beta_j x_{ij}$, we know that

$$\frac{\partial \eta_i}{\partial \beta_j} = x_{ij}.$$

Finally, as $g(\mu_i) = \eta_i$, we know that $\frac{d\mu_i}{d\eta_i}$ depends on the specific link function used. So, the score equations can be written as,

$$s_j = \sum_{i=1}^{n} \frac{(y_i - \mu_i)}{a(\phi)} \frac{1}{V_i} \frac{d\mu_i}{d\eta_i} x_{ij}, \text{ for } j = 1, .., p. \tag{3.14}$$

If we then define the quadratic weights,

$$w_i = \frac{1}{V_i} (\frac{d\mu_i}{d\eta_i})^2, \tag{3.15}$$

we can rewrite the score equations as

$$s_j = \sum_{i=1}^{n} (\frac{y_i - \mu_i}{a(\phi)}) w_i \frac{d\eta_i}{d\mu_i} x_{ij}, \text{ for } j = 1, ...p. \tag{3.16}$$

Under the canonical link (i.e. $\theta_i = g(\mu_i) = \eta_i$), we know that

$$\frac{d\theta_i}{d\mu_i} = \frac{dg(\mu_i)}{d\mu_i} = \frac{1}{V_i},$$

and,

$$w_i (\frac{d\eta_i}{d\mu_i}) = w_i (\frac{dg(\mu_i)}{d\mu_i}) = \frac{1}{V_i} (\frac{d\mu_i}{d\theta_i})^2 \frac{1}{V_i} = 1.$$

In this case, the score equations collapse to

$$s_j = \sum_{i=1}^{n} (\frac{y_i - \mu_i}{a(\phi)}) x_{ij}.$$

Often, $a(\phi)$ is constant for all observations. In this case, the score equations simplify further such that,

$$s_j = \sum_{i=1}^{n} (y_i - \mu_i) x_{ij}.$$

We now discuss how to obtain parameter estimates for $\boldsymbol{\beta}$. Let's define the score vector as

$$\mathrm{s}(\boldsymbol{\beta}) = (s_1(\boldsymbol{\beta}), ..., s_p(\boldsymbol{\beta}))^T.$$

Then, we obtain the MLE $\hat{\boldsymbol{\beta}}$, by solving $\mathrm{s}(\boldsymbol{\beta}) = \mathbf{0}$. This is equivalent to solving the following set of p simultaneous equations,

$$\sum_{i=1}^{n} (\frac{y_i - \mu_i}{a(\phi)}) w_i \frac{d\eta_i}{d\mu_i} x_{ij} = 0, \text{ for } j = 1, ...p. \tag{3.17}$$

3.2.2 ML as Iteratively Reweighted Least Squares

Estimation via Fisher Scoring

The score equations can be iteratively solved using the Fisher Scoring method (Agresti, 2015; McCullagh and Nelder, 1996),

$$\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^{(t)} + [\mathcal{I}(\boldsymbol{\beta}^{(t)})]^{-1}\text{s}(\boldsymbol{\beta}^{(t)}), \text{ for } t = 0, 1, \text{...etc.} \tag{3.18}$$

In order to apply this method, we need to obtain the (j, k)-th element of the information matrix $[\mathcal{I}(\boldsymbol{\beta})]$. To do this, we make use of a useful result which holds for the exponential family. For the partial contribution l_i,

$$-E[\frac{\partial^2 l_i}{\partial\beta_j\partial\beta_k}] = E[(\frac{\partial l_i}{\partial\beta_j})(\frac{\partial l_i}{\partial\beta_k})]$$

We know then (after omitting the dispersion factor),

$$-E[\frac{\partial^2 l_i}{\partial\beta_j\partial\beta_k}] = E\big[\big((y_i-\mu_i)w_i\frac{d\eta_i}{d\mu_i}x_{ij}\big)\big((y_i-\mu_i)w_i\frac{d\eta_i}{d\mu_i}x_{ik}\big)\big] = w_i x_{ij} x_{ik}$$

Since $l(\boldsymbol{\beta}) = \sum_{i=1}^{n} l_i$,

$$[\mathcal{I}(\boldsymbol{\beta})]_{jk} = -E[\frac{\partial^2 l(\boldsymbol{\beta})}{\partial\beta_j\partial\beta_k}] = \sum_{i=1}^{n} w_i x_{ij} x_{ik}. \tag{3.19}$$

Then, the entire information matrix can be written in the following manner,

$$[\mathcal{I}(\boldsymbol{\beta})] = \mathbf{X}^T\mathbf{W}\mathbf{X}, \tag{3.20}$$

where $\mathbf{X}$ is the design matrix and $\mathbf{W} = diag\big(w_1, .., w_n\big)$. To compute an updated estimate $\boldsymbol{\beta}^{(t+1)}$ using equation (3.18), we can complete the following steps.

1. Evaluate every element of $\text{s}(\boldsymbol{\beta}^{(t)})$ and $[\mathcal{I}(\boldsymbol{\beta}^{(t)})]$,
2. Invert the information matrix to obtain $[\mathcal{I}(\boldsymbol{\beta}^{(t)})]^{-1}$.

Estimation via Weighted Least Squares

We can premultiply equation (3.18) by $\mathcal{I}(\boldsymbol{\beta}^{(t)})$ to obtain,

$$\mathcal{I}(\boldsymbol{\beta}^{(t)})\boldsymbol{\beta}^{(t+1)} = \mathcal{I}(\boldsymbol{\beta}^{(t)})\boldsymbol{\beta}^{(t)} + \text{s}(\boldsymbol{\beta}^{(t)}) \tag{3.21}$$

Omitting the t and $t+1$ subscripts for now, the j-th element of the LHS is

$$[\mathcal{I}(\boldsymbol{\beta})\boldsymbol{\beta}]_j = \sum_{k=1}^{p}\sum_{i=1}^{n} w_i x_{ij} x_{ik}\beta_k.$$

Thus,

$$[\mathcal{I}(\boldsymbol{\beta})\boldsymbol{\beta}] = \mathbf{X}^T\mathbf{W}\mathbf{X}\boldsymbol{\beta}.$$

On the RHS, the j-th element is,

$$\begin{aligned}[\mathcal{I}(\boldsymbol{\beta})\boldsymbol{\beta} + \mathrm{s}(\boldsymbol{\beta})]_j &= \sum_{k=1}^{p}\sum_{i=1}^{n} w_i x_{ij} x_{ik} \beta_k + \sum_{i=1}^{n}(y_i - \mu_i) w_i \frac{d\eta_i}{d\mu_i} x_{ij} \\ &= \sum_{i=1}^{n} w_i x_{ij} \big[\sum_{k=1}^{p} x_{ik}\beta_k + (y_i - \mu_i)\frac{d\eta_i}{d\mu_i} \big] = \sum_{i=1}^{n} w_i x_{ij} z_i.\end{aligned}$$

Thus,

$$[\mathcal{I}(\boldsymbol{\beta})\boldsymbol{\beta} + \mathrm{s}(\boldsymbol{\beta})] = \mathbf{X}^{\mathbf{T}}\mathbf{W}\mathbf{z}.$$

Putting this together, we get,

$$\mathbf{X}^T\mathbf{W}\mathbf{X}\boldsymbol{\beta} = \mathbf{X}^T\mathbf{W}\mathbf{z}, \tag{3.22}$$

which are the *weighted least squares equations*. If we solve these equations, we will obtain the MLE $\widehat{\boldsymbol{\beta}}$, but this must be done iteratively, as both $\mathbf{W}$ and $\mathbf{z}$ depend on the coefficients, $\boldsymbol{\beta}$. Let $\eta_i^{(t)} = \theta_i^{(t)}$ be the current estimate of the linear predictor and let $\mu_i^{(t)} = g^{-1}(\eta_i^{(t)})$ denote the current fitted mean. The iterative procedure to estimate $\boldsymbol{\beta}$ is provided below (Agresti, 2015; McCullagh and Nelder, 1996).

1. Form the adjusted dependent variable $z_i^{(t)} = \eta_i^{(t)} + (y_i - \mu_i^{(t)})\frac{d\eta_i^{(t)}}{d\mu_i^{(t)}} = \eta_i^{(t)} + \frac{(y_i - \mu_i^{(t)})}{V_i}$ for $i = 1, ...n$ where $V_i = b''(\theta_i^{(t)})$.

2. Regress the new dependent variable $\mathbf{z}^{(t)}$ upon the covariates $X_1, ..., X_p$ with the weights $\mathbf{W}^{(t)}$, to obtain,

$$\boldsymbol{\beta}^{(t+1)} = (\mathbf{X}^T\mathbf{W}^{(t)}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}^{(t)}\mathbf{z}^{(t)} \tag{3.23}$$

3. Form a new estimate of $\boldsymbol{\eta}^{(t+1)} = \mathbf{X}\boldsymbol{\beta}^{(t+1)}$ and a new dependent variable $\mathbf{z}^{(t+1)}$.

4. Repeat Steps 1-3 until the changes are sufficiently small. Upon convergence,

$$\widehat{\boldsymbol{\beta}} = \big(\mathbf{X}^T\hat{\mathbf{W}}\mathbf{X}\big)^{-1}\mathbf{X}^T\hat{\mathbf{W}}\hat{\mathbf{z}}, \tag{3.24}$$

for the estimated adjusted response $\hat{\mathbf{z}} = \mathbf{X}\widehat{\boldsymbol{\beta}} + \mathbf{V}\big(\mathbf{y} - \widehat{\boldsymbol{\mu}}\big)$, where $\mathbf{V} = diag\big(\frac{1}{V_1}, ..., \frac{1}{V_n}\big)$.

The ‘true’ maximum likelihood estimator can be approximated by a few cycles of weighted least squares, where the weight matrix changes at each iteration. This procedure is called *iteratively reweighted least squares*.

3.3 Extending HOLP to GLMs

When p is large relative to n, the maximum likelihood approach detailed above becomes impractical. First, simultaneously solving the score equations will not produce any parameter estimates that are identically zero. Therefore, variable selection cannot occur. Second, for large p, the variance of the estimated coefficients will grow, leading to an imprecise model. When $p > n$, the p score equations can no longer be solved uniquely for $\boldsymbol{\beta}$. These problems must be addressed with alternative methods that are specifically designed for high-dimensional data.

In this section, we propose the GLM-HOLP, as a generalisation of the HOLP procedure (Wang and Leng, 2015) reviewed in Section 2.3.3.

3.3.1 Computing the GLM-HOLP

An analogous approach to the iteratively reweighted least squares method can be used to compute the generalised HOLP solution.

1. At the $(t+1)$-th iteration, let $V_i = b''(\theta_i^{(t)})$ and $z_i^{(t)} = \sum_{j=1}^{p} x_{ij}\beta_j^{(t)} + \frac{(y_i-\mu_i)}{V_i}$, where $\theta_i^{(t)} = g^{-1}(\eta_i^{(t)})$ and $\eta_i^{(t)} = \mathbf{x}_i^T\boldsymbol{\beta}^{(t)}$.
2. Let $z_i^* = z_i^{(t)}\sqrt{V_i}$ and $x_{ij}^* = x_{ij}\sqrt{V_i}$.
3. Apply the HOLP method to compute $\boldsymbol{\beta}^{(t+1)}$ with $\mathbf{z}^*$ as the response and $\mathbf{X}^*$ as the design matrix., i.e.
$$\boldsymbol{\beta}^{(t+1)} = \mathbf{X}^{*T}(\mathbf{X}^*\mathbf{X}^{*T})^{-1}\mathbf{z}^* \tag{3.25}$$
4. Repeat Steps 1-3 until convergence and the changes are sufficiently small. Upon convergence, use $\widehat{\boldsymbol{\beta}}$ to select the d best variables.

3.3.2 Special cases for GLM-HOLP

Normal Linear Model

For this model, $V_i = b''(\theta_i) = 1$. Thus, at time t, the adjusted response variable will be

$$z_i^{(t)} = \sum_{j=1}^{p} x_{ij}\beta_j^{(t)} + (y_i - \mu_i^{(t)}) = \sum_{j=1}^{p} x_{ij}\beta_j^{(t)} + e_i, \tag{3.26}$$

where $e_i = y_i - \mu_i \sim N(0,\sigma^2)$. This is exactly the form of the model provided in Section 3.1.4. Given that g(.) is the identity link, we expect there to be little to no difference between the GLM-HOLP estimates and the traditional HOLP estimates.

Binomial Logistic Model

For the logistic regression model, we know that $V_i = b''(\theta_i) = \frac{\exp(\theta_i)}{(1+\exp(\theta_i))^2} = \pi_i(1-\pi_i)$. So, at time t, the adjusted response variable will be

$$z_i^{(t)} = \sum_{j=1}^{p} x_{ij}\beta_j^{(t)} + \frac{(y_i - \pi_i^{(t)})}{V_i} = \sum_{j=1}^{p} x_{ij}\beta_j^{(t)} + \frac{(y_i - \pi_i^{(t)})}{\pi_i^{(t)}(1-\pi_i^{(t)})}. \tag{3.27}$$

We also know that $z_i^* = z_i^{(t)}\sqrt{\pi_i^{(t)}(1-\pi_i^{(t)})}$ and $x_{ij}^* = x_{ij}\sqrt{\pi_i^{(t)}(1-\pi_i^{(t)})}$, at time t.

Poisson Loglinear Model

For the Poisson regression model, we know that $V_i = b''(\theta_i) = \exp(\theta_i) = \mu_i$. So, at time t, the adjusted response variable will be

$$z_i^{(t)} = \sum_{j=1}^{p} x_{ij}\beta_j^{(t)} + \frac{(y_i - \mu_i^{(t)})}{V_i} = \sum_{j=1}^{p} x_{ij}\beta_j^{(t)} + \frac{(y_i - \mu_i^{(t)})}{\mu_i^{(t)}}. \tag{3.28}$$

We also know that $z_i^* = z_i^{(t)}\sqrt{\mu_i^{(t)}}$ and $x_{ij}^* = x_{ij}\sqrt{\mu_i^{(t)}}$, at time t.

3.4 Variable selection for high-dimesional GLMs ($p > n$)

Variable selection for generalised linear models has not been as extensively studied as variable selection for linear models. In this section, we briefly review some of the existing work in this area. We assume that all of the GLMs have an intercept, β_0, for consistency with the literature.

3.4.1 (I)SIS for GLMs

In Section 2.3.2, we discuss the sure independence screening (SIS) procedure for linear models proposed by Fan and Lv (2008). The primary goal of Fan et al. (2009) is to extend SIS and ISIS to much more general models. Fan et al. develop an alternative iterative procedure, which bypasses the issue of defining a model residual altogether. Indeed, this new (I)SIS procedure now incorporates feature deletion in the selection process.

Feature ranking by marginal utilities

If we are interested in modelling the general relationship between the response for observation i, Y_i, and the covariates $X_1, ..., X_p$, we can use a pseudo-likelihood function. Fan et al. (2009) define $Q(\beta_0, \boldsymbol{\beta})$ to be a negative pseudo-likelihood function of the form,

$$Q(\beta_0, \boldsymbol{\beta}) = n^{-1}\sum_{i=1}^{n} L(y_i, \beta_0 + \mathbf{x}_i^T\boldsymbol{\beta}). \tag{3.29}$$

Fitting the model is equivalent to minimising the negative pseudo-likelihood. So, Fan et al. set L to be the loss function of using $\beta_0 + \mathbf{x}_i^T\boldsymbol{\beta}$ to predict the response, Y_i. Then,

$$L_j = \min_{\beta_0,\beta_j}\Big\{n^{-1}\sum_{i=1}^{n} L(y_i, \beta_0 + x_{ij}\beta_j)\Big\}, \tag{3.30}$$

can be seen as the marginal utility of the j-th feature, for $j = 1, .., p$. The idea for SIS is to compute a vector of marginal utilities, $\mathbf{L} = (L_1, ..., L_p)^T$ and rank the features. A smaller marginal utility implies that a predictor is more important. In order to compute each L_j, we only need to fit two model parameters, so the feature ranking can be completed efficiently, even in an ultrahigh-dimensional setting. The covariate, X_j, for $j \in \{1, .., p\}$, is selected by SIS if its corresponding marginal utility, L_j, is among the d smallest elements from the vector $\mathbf{L}$.

Penalised pseudo-likelihood

After crudely selecting d features using marginal utility ranking, Fan et al. (2009) propose the use of a penalised pseudo-likelihood method to simultaneously complete further variable selection and estimate the model parameters. We assume that $X_1, ..., X_d$ are the features selected by SIS. Let us define $\mathbf{x}_{i,d} = (x_{i1}, ..., x_{id})^T$ and also redefine $\boldsymbol{\beta} = (\beta_1, ..., \beta_d)^T$. Then, we are looking to minimise the following penalised pseudo-likelihood,

$$\ell(\beta_0, \boldsymbol{\beta}) = n^{-1} \sum_{i=1}^{n} L(y_i, \beta_0 + \mathbf{x}_{i,d}^T \boldsymbol{\beta}) + \sum_{j=1}^{d} p_\lambda(|\beta_j|). \tag{3.31}$$

As in Section 2.2.1, $p_\lambda(|\beta|)$ is referred to as the penalty function, with λ as the shrinkage parameter. Fan et al. recommend the use of the SCAD (Fan and Li, 2001), the lasso (Tibshirani, 1996) or the MCP (Zhang, 2007) for $p_\lambda(|\beta|)$.

GLMs in a pseudo-likelihood framework

Recall the form of the exponential dispersion family density function from equation (3.4),

$$f(y_i; \theta_i, \phi) = \exp\left(\frac{(y_i\theta_i - b(\theta_i))}{\phi} + c(y, \phi) \right), \text{ for } i = 1, ..., n,$$

where $\theta(.), b(.)$ and $c(.)$ are all known. For convenience, we set $a(\phi) = \phi$. The linear predictor is connected to the mean for observation i, μ_i, through the link function $g(.)$, i.e. $g(\mu_i) = \beta_0 + \mathbf{x}_i^T \boldsymbol{\beta}$. If we look at the form of the likelihood function for a GLM, we see that it fits neatly into the negative pseudo-likelihood framework devised by Fan et al. (2009). Indeed, we can set,

$$L(y_i, \beta_0 + \mathbf{x}_i^T \boldsymbol{\beta}) = \sum_{i=1}^{n} \{b(\theta_i(g^{-1}(\beta_0 + \mathbf{x}_i^T \boldsymbol{\beta}))) - y_i\theta_i(g^{-1}(\beta_0 + \mathbf{x}_i^T \boldsymbol{\beta}))\}.$$

If we use the canonical link, the pseudo-likelihood becomes,

$$L(y_i, \beta_0 + \mathbf{x}_i^T \boldsymbol{\beta}) = \sum_{i=1}^{n} \{b(\beta_0 + \mathbf{x}_i^T \boldsymbol{\beta}) - y_i(\beta_0 + \mathbf{x}_i^T \boldsymbol{\beta})\}.$$

Iterative feature selection

We now describe the ISIS procedure that Fan et al. provide to handle the screening issues presented by the original SIS (Fan and Lv, 2008) method. By implementing an iterative process for feature selection, ISIS seeks to retain more of the joint covariate structure of the data, while still being computationally efficient. The proposed procedure contains the following steps (Liu et al., 2015; Fan et al., 2009).

1. Obtain the marginal utility vector, $\mathbf{L}$, and use it to select the set $\widehat{\mathcal{A}_1} = \{1 \leq j \leq p : L_j$ is among the first k_1 smallest of all $\}$. Apply a penalised likelihood method to the submodel indexed by $\widehat{\mathcal{A}_1}$ and select a new refined subset, $\widehat{\mathcal{M}}$.

2. For each $j \in \{1, .., p\} \setminus \widehat{\mathcal{M}}$, calculate,

$$L_j^{(2)} = \min_{\beta_0, \beta_{\widehat{\mathcal{M}}}, \beta_j} \Big\{ n^{-1} \sum_{i=1}^{n} L(y_i, \beta_0 + \mathbf{x}_{i,\widehat{\mathcal{M}}}^T \boldsymbol{\beta}_{\widehat{\mathcal{M}}} + x_{ij}\beta_j) \Big\},$$

   where $\mathbf{x}_{i,\widehat{\mathcal{M}}}$ is the subvector of $\mathbf{x}_i$ containing the specific covariates selected for $\widehat{\mathcal{M}}$. Here, $L_j^{(2)}$ is the extra contribution of the j-th covariate, after accounting for the predictors in $\widehat{\mathcal{M}}$. After ordering $\{j \in \widehat{\mathcal{M}}^C : L_j^{(2)}\}$, we select the set, $\widehat{\mathcal{A}_2} = \{j \in \widehat{\mathcal{M}}^C : L_j^{(2)}$ is among the first k_2 smallest of all$\}$.

3. Apply a penalised likelihood method to the submodel indexed by $\widehat{\mathcal{M}} \cup \widehat{\mathcal{A}_2}$ and select a new $\widehat{\mathcal{M}}$. In other words, we want,

$$\widehat{\boldsymbol{\beta}_2} = \underset{\beta_0, \beta_{\widehat{\mathcal{M}}}, \beta_{\widehat{\mathcal{A}_2}}}{\operatorname{argmin}} \Big\{ n^{-1} \sum_{i=1}^{n} L(y_i, \beta_0 + \mathbf{x}_{i,\widehat{\mathcal{M}}}^T \boldsymbol{\beta}_{\widehat{\mathcal{M}}} + \mathbf{x}_{i,\widehat{\mathcal{A}_2}}^T \boldsymbol{\beta}_{\widehat{\mathcal{A}_2}} + \sum_{j \in \widehat{\mathcal{M}} \cup \widehat{\mathcal{A}_2}} p_\lambda(|\beta_j|) \Big\}.$$

   Thus, the indices of the coefficients that are non-zero in $\widehat{\boldsymbol{\beta}_2}$ form the updated $\widehat{\mathcal{M}}$.

4. Repeat steps 2 and 3 until $|\widehat{\mathcal{M}}| \leq d$, for some predetermined value of d less than n. The set $\widehat{\mathcal{M}}$ will be the final submodel.

This version of ISIS bypasses the issue of defining a model residual altogether and has the added advantage of incorporating feature deletion in the selection process. Two other variants of ISIS are provided by Fan et al., aimed at reducing false selection rates. Please see the original paper Fan et al. (2009) for further details.

3.4.2 Using the lasso to fit GLMs

Friedman et al. (2010) discuss the extension of the elastic-net penalty (Zou and Hastie, 2005) to a generalised linear model setting, using penalised maximum likelihood. The accompanying R-package, **glmnet**, is able to computationally fit a GLM by solving the following problem,

$$\min_{\beta_0, \beta} \{ n^{-1} \sum_{i=1}^{n} L(y_i, \beta_0 + \mathbf{x}_i^T \boldsymbol{\beta}) - \lambda((1-\alpha)||\beta||_2^2/2 + \alpha||\beta||_1) \}. \qquad (3.32)$$

Here, $L(y, \eta)$ is the negative log-likelihood of the model. Recall that the elastic-net penalty is a compromise between the lasso and the ridge regression methods. In order to use the lasso to fit GLMs, we need to set $\alpha = 1$. The algorithm uses an iterative coordinate descent approach, where the objective function is optimised over each model parameter (while keeping the others fixed), cycling repeatedly until convergence.

Chapter 4

Numerical Studies

Within this chapter, we provide three numerical experiments to evaluate the performance of the GLM-HOLP method. In the first part, we compare the screening accuracy of GLM-HOLP (there is a detailed discussion of the procedure in Section 3.3) to that of (I)SIS (Fan et al., 2009) and the lasso (Tibshirani, 1996). We do this within the context of three special cases: linear regression, logistic regression and Poisson regression. Specifically, we consider three different configurations of $p = 1000$ features. In the second part, we evaluate the computational time of various methods. This is to judge computational efficiency. Finally, in the third part, we consider a real data example, taken from a prostate cancer study conducted by Singh et al. (2002).

4.1 Simulation study I: Screening accuracy

For this study, we set $(p, n) = (1000, 100)$ and let all of the variables, $X_1, ..., X_p$, be marginally $N(0, 1)$. We simulate covariates with the aim of exploring the applicability of GLM-HOLP in three different model contexts: linear regression, logistic regression, and Poisson regression. For each of these three generalised linear models, we will consider three different configurations of $X_1, ..., X_p$:

Case (i): Independent predictors $X_1, ..., X_p$ are independent and identically distributed $N(0, 1)$ random variables
Case (ii): Compound symmetry $X_1, .., X_p$ are equally correlated with correlation ρ, and we set $\rho = 0.3, 0.6$ or 0.9.
Case (iii): Autoregressive correlation We assume that each X_i follows a multivariate normal, where $cov(x_i, x_j) = \rho^{|i-j|}$ and $\rho = 0.3, 0.6$ or 0.9. This type of correlation structure is particularly common when the covariates are ordered.

Case (i), with fully independent predictors, is the easiest correlation structure for variable selection. In cases (ii) and (iii), there is greater dependency between the predictors. We expect feature selection to be more challenging in these situations, especially for larger values of ρ. In all cases, we are interested in reporting the probablity of including the true model when selecting a submodel of size n.

Linear Regression Model

In our first example, we assume a linear model. The response variable Y for realisation i is set to be $y_i = \mathbf{x}_i^T \boldsymbol{\beta} + \epsilon_i$, where $\epsilon_i \sim N(0,1)$. Here, $\mathbf{x}_i$ is the vector of realisations of the p-covariates, $(x_{i1}, ..., x_{ip})^T$, for $i = 1, .., n$. The simulation settings for this example are drawn from Wang and Leng (2015) and the simulation results are displayed in Table 4.1. For each simulation case, the coefficients were chosen as follows:

Case (i): This simulation is drawn directly from Fan and Lv (2008). We set the true model $\mathcal{M}_* = \{1, 2, 3, 4, 5\}$ and the corresponding coefficients are generated as,

$\beta_i = (-1)^{u_i}(|N(0,1)| + 4\log(n)/\sqrt{n}),$ with $u_i \sim Ber(0.4)$ for $i \in \mathcal{M}_*$ and $\beta_i = 0$ otherwise.

Case (ii): For each value of ρ, we set the coefficients to β_i for $i = 1, .., 5$ and $\beta_i = 0$ otherwise.

Case (iii): For each value of ρ, we set the coefficients as $\beta_1 = 3, \beta_4 = 1.5, \beta_7 = 2$ and $\beta_i = 0$ otherwise.

Example		GLM-HOLP	SIS	ISIS	lasso
(i) Independent predictors		1.00	1.00	1.00	1.00
(ii) Compound symmetry	$\rho = 0.3$	1.00	1.00	1.00	1.00
	$\rho = 0.6$	1.00	1.00	0.95	1.00
	$\rho = 0.9$	1.00	0.00	0.00	0.85
(iii) Autoregressive	$\rho = 0.3$	1.00	1.00	1.00	1.00
	$\rho = 0.6$	1.00	1.00	1.00	1.00
	$\rho = 0.9$	1.00	1.00	1.00	1.00

Table 4.1: Probability to include true model when $(p, n) = (1000, 100)$ in a linear regression setting

Logistic Regression Model

In our second example, the generic response variable Y is assumed to follow the Bernoulli distribution, with probability of success $\pi \in (0, 1)$. The logistic regression model for observation i can be written as, $\log(\frac{\pi_i}{1-\pi_i}) = \mathbf{x}_i^T \boldsymbol{\beta}$. The coefficients used in each of the three simulation cases are as follows:

Case (i): This example is drawn from Fan et al. (2009). Let $\beta_1 = 1.2439, \beta_2 = -1.3416, \beta_3 = -1.3500, \beta_4 = -1.7971, \beta_5 = -1.5810, \beta_6 = -1.5967$ and for $i > 6, \beta_i = 0$.
Case (ii): For each value of ρ, we set $\beta_1 = 5, \beta_2 = -5, \beta_3 = 5, \beta_6 = -5$, and $\beta_i = 0$ otherwise.
Case (iii): For each value of ρ, we set $\beta_1 = 3, \beta_4 = 2, \beta_7 = 2, \beta_10 = 3$, and $\beta_i = 0$ otherwise.

The results for this example are detailed in Table 4.2 below.

Example		GLM-HOLP	SIS	ISIS	lasso
(i) Independent predictors		1.00	0.99	0.99	0.99
(ii) Compound symmetry	$\rho = 0.3$	1.00	1.00	1.00	1.00
	$\rho = 0.6$	1.00	0.90	0.80	1.00
	$\rho = 0.9$	1.00	0.01	0.02	0.99
(iii) Autoregressive	$\rho = 0.3$	1.00	0.96	0.94	1.00
	$\rho = 0.6$	1.00	1.00	1.00	1.00
	$\rho = 0.9$	0.98	0.95	0.95	1.00

Table 4.2: Probability to include true model when $(p, n) = (1000, 100)$ in a logistic regression setting

Poisson Regression Model

Our final example requires the response variable Y to be Poisson distributed, with mean $\mu > 0$. The Poisson regression model for observation i can be written as, $\log(\mu_i) = \mathbf{x}_i^T \boldsymbol{\beta}$. The coefficients used in each of the three simulation cases are as follows:

Case (i): We set $\beta_1 = -0.5243, \beta_2 = 0.5314, \beta_3 = -0.5012, \beta_4 = -0.4850, \beta_5 = -0.4133, \beta_6 = 0.5234$ and for $i > 6, \beta_i = 0$.
Case (ii): For each value of ρ, we set $\beta_1 = 0.5, \beta_2 = -0.5, \beta_10 = 0.6, \beta_11 = 0.5$, and $\beta_i = 0$ otherwise.
Case (iii): For each value of ρ, we set $\beta_1 = 0.75, \beta_4 = 0.6, \beta_7 = -1.0, \beta_10 = 0.9$, and $\beta_i = 0$ otherwise.

The results for this example are detailed in Table 4.3.

Example		GLM-HOLP	SIS	ISIS	lasso
(i) Independent predictors		1.00	1.00	1.00	1.00
(ii) Compound symmetry	$\rho = 0.3$	1.00	0.20	1.00	1.00
	$\rho = 0.6$	1.00	0.00	0.96	1.00
	$\rho = 0.9$	0.10	0.00	0.00	0.00
(iii) Autoregressive	$\rho = 0.3$	1.00	1.00	1.00	1.00
	$\rho = 0.6$	1.00	1.00	1.00	1.00
	$\rho = 0.9$	0.40	0.00	0.00	0.00

Table 4.3: Probability to include true model when $(p, n) = (1000, 100)$ in a Poisson regression setting

Summary of Simulation Study I

Predictably, all four methods perform near-perfectly across each of the generalised linear models, when screening fully independent predictors in case (i). However, case (i) is not a realistic scenario in the context of high-dimensional data. In addition, if faced with an autoregressive correlation structure, as in case (iii), it appears that all of the methods generally perform well, even for larger values of ρ.

Case (ii) seems to be the most challenging situation for variable selection. For higher values of ρ, the compound symmetry correlation structure proves to be very difficult for SIS and ISIS. In comparison, the lasso and GLM-HOLP tend to fare better with case (ii), especially when applied to a linear or logistic model.

However, all four methods perform unfavourably in a Poisson regression setting for cases (ii) and (iii), when $\rho = 0.9$. This could be examined in further detail, by extending cases (ii) and (iii) to fit a variety of different coefficient specifications.

Overall, through this study, we have established that the proposed GLM-HOLP method has a wide range of applicability, across various correlation structures and generalised linear model settings.

4.2 Simulation study II: Computational efficiency

Computational efficiency is an incredibly important concern for variable selection techniques. As dimensions scale up, it is vital that we develop flexible tools that allow for quick and accurate model selection and representation. In this study, we simulate a dataset with a binary response, using case (ii) from Simulation I. We set $n = 100$ and $\rho = 0.3$, with a view

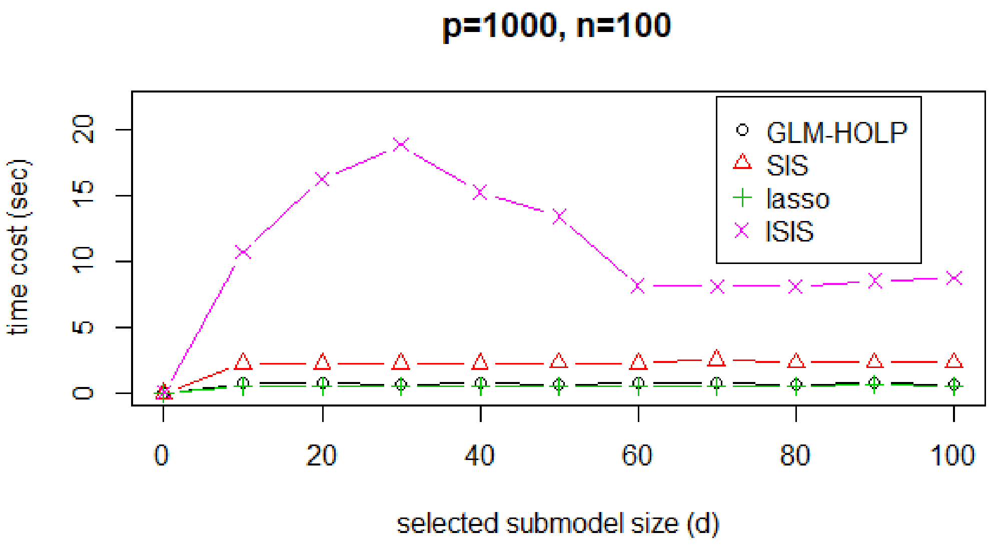


Figure 4.1: Computational time against the submodel size when $(p, n) = (1000, 100)$

to demonstrate the computational efficiency of GLM-HOLP as compared to SIS, ISIS and the lasso.

In Figure 4.1, we set the data dimension at $p = 1000$ and alter the submodel size from $d = 1, .., 100$, recording the time it takes to run each variable screener. Whereas in Figure 4.2, after fixing the submodel size at $d = 50$, we are interested in obtaining the running times for various values of p, ranging from 50 to 2500.

When the value of p is fixed, SIS and GLM-HOLP are the most efficient algorithms, with very similar running times. In comparison, the growth trajectory of ISIS in Figure 4.1 is markedly different. Even though we can keeping asking for larger submodels, at some point, the ISIS algorithm will recruit fewer features than the specified value of d, due to its feature deletion properties (there are further details in Section 3.4.1). For this reason, we see that for values of d greater than 30, the ISIS running time decreases and then stays relatively constant. While the lasso running times are provided here for the sake of completion, the algorithm does not specifically rely on a value of d.

Similarly, when the submodel size, d, is fixed, SIS, Lasso and GLM-HOLP vastly outperform ISIS in Figure 4.2, as the value of p grows. SIS has a marginally slower running time than Lasso and GLM-HOLP, but this difference is not noticeably significant. While we can see that lasso is the most preferrable algorithm in terms of speed, it may not provide the most accurate results on its own, when faced with more complex data (see Section 4.3). After taking into account the performance of GLM-HOLP in Simulation I, we can conclude that our proposed method is reasonably robust and competitive.

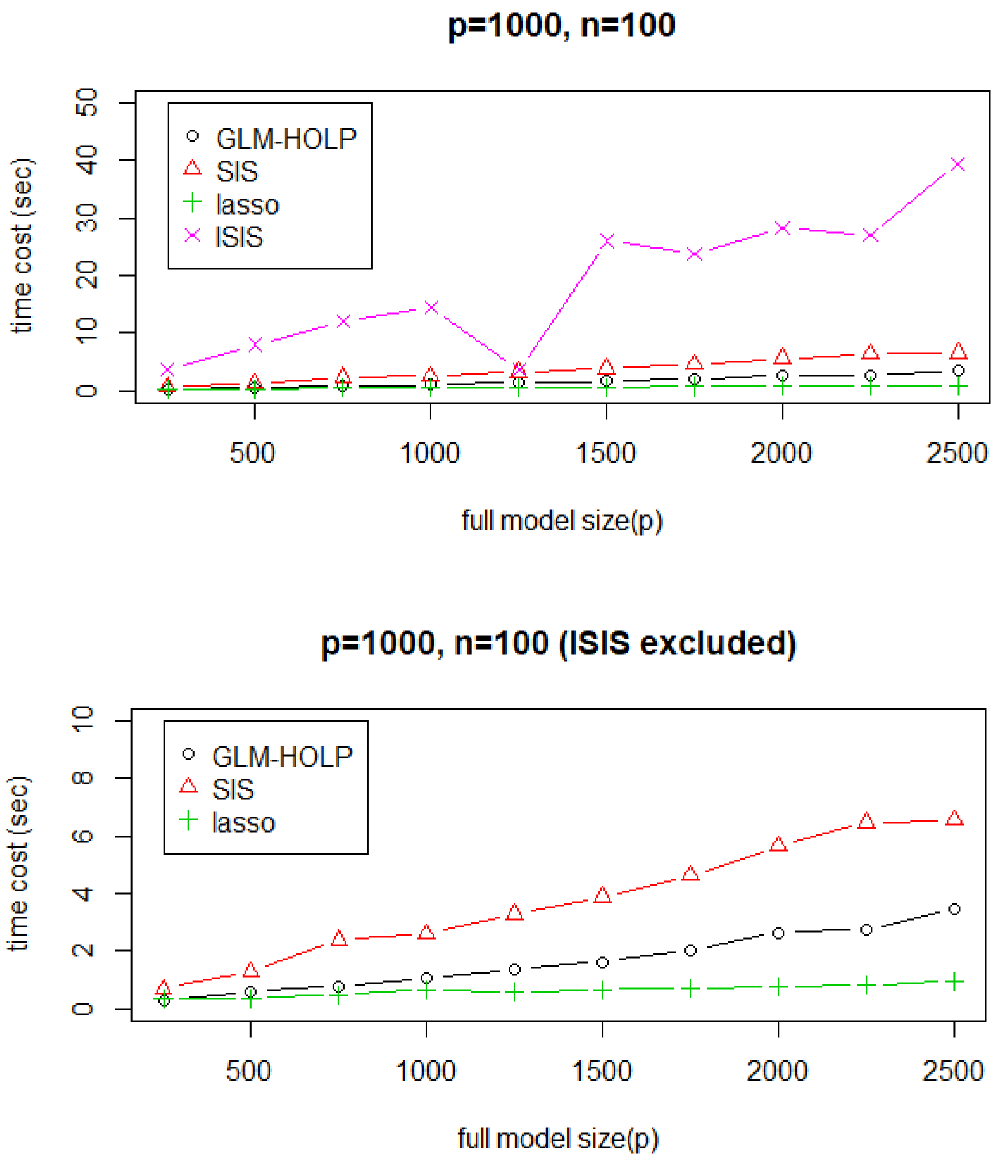


Figure 4.2: Computational time against the total number of covariates when $(d, n) = (50, 100)$

4.3 A real data application

As discussed in Chapter 2, feature screening can be used as a preselection step to reduce dimensionality, before performing further refined parameter estimation and variable selection. To fully explore the impact of using screening on second stage analysis, we compare two different two-stage procedures with a one-stage method on a real data set.

We consider the prostate cancer study conducted by Singh et al. (2002) as our example.

This microarray study obtained gene expression data from 52 patients with prostate cancer and 50 healthy men. The data was collected with the view to develop a gene-based classification rule for prostate cancer patients, in order to predict the predisposition of an individual developing the disease.

In this example, we are using a modified version of the original data set, obtained from the **sda** R-package. Overall, we have measurements for 6033 genes, across a total sample of 102 individuals. This set of genes exhibits sufficient signal for the purposes of our analysis. The binary response variable that we are interested in predicting is whether or not a patient is healthy (level 1: cancer, level 2: healthy). A logistic regression model is the most appropriate GLM to apply to this data set.

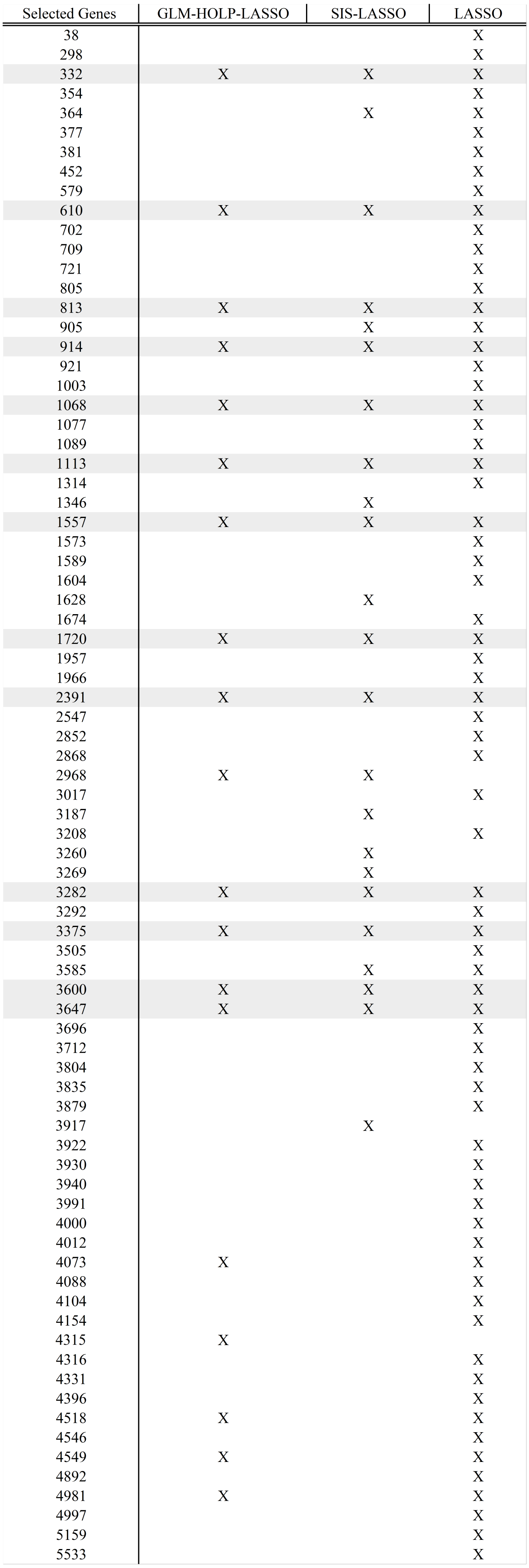
The two feature screeners that we want to compare within the framework of a two-stage procedure are the proposed GLM-HOLP and the extended SIS procedure outlined by Fan et al. (2009) for generalised linear models. To construct our two-stage procedures, we first use a screener to select an initial submodel of size $d = 30$ from the dataset and then apply the lasso to this submodel to output the final result. In this case study, we compare the performance of GLM-HOLP-Lasso and SIS-Lasso to the stand-alone lasso.

Methods	Mean errors	Standard errors	Final model size
GLM-HOLP-LASSO	0.222	0.086	19
SIS-LASSO	0.116	0.121	23
LASSO	0.279	0.080	71

Table 4.4: The 10-fold cross validation error for three different methods

For completion, these three methods are examined via 10-fold cross validation for predictive accuracy. In Table 4.4, we provide the means and standard errors of the mean square errors for prediction and the final model size.

Overall, it can be seen that models selected by GLM-HOLP-Lasso and SIS-Lasso are more competitive than the lasso on its own, with respect to cross validation error and final model size. We can see that a two-stage approach allows for a far-more refined and parsimonious representation of the data. Using Figure 4.3, we can read off the common features selected by all three methods: V332, V610, V813, V914, V1068, V1113, V1557, V1720, V2391, V3282, V3375, V3600, and V3647. To extend this case study further in the future, it would also be interesting to compare the thirteen highlighted genes in Figure 4.3 across the original, full data set.



Selected Genes	GLM-HOLP-LASSO	SIS-LASSO	LASSO
38			X
298			X
332	X	X	X
354			X
364		X	X
377			X
381			X
452			X
579			X
610	X	X	X
702			X
709			X
721			X
805			X
813	X	X	X
905		X	X
914	X	X	X
921			X
1003			X
1068	X	X	X
1077			X
1089			X
1113	X	X	X
1314			X
1346		X	
1557	X	X	X
1573			X
1589			X
1604			X
1628		X	
1674			X
1720	X	X	X
1957			X
1966			X
2391	X	X	X
2547			X
2852			X
2868			X
2968	X	X	
3017			X
3187		X	
3208			X
3260		X	
3269		X	
3282	X	X	X
3292			X
3375	X	X	X
3505			X
3585		X	X
3600	X	X	X
3647	X	X	X
3696			X
3712			X
3804			X
3835			X
3879			X
3917		X	
3922			X
3930			X
3940			X
3991			X
4000			X
4012			X
4073	X		X
4088			X
4104			X
4154			X
4315	X		
4316			X
4331			X
4396			X
4518	X		X
4546			X
4549	X		X
4892			X
4981	X		X
4997			X
5159			X
5533			X

Figure 4.3: Selected genes for the the diagnosis endpoint

Chapter 5

Conclusion

In this project, we propose a simple and efficient procedure for screening features within a high-dimensional generalised linear model framework. In building upon the HOLP screener proposed by Wang and Leng (2015), we seek to retain computational efficiency and flexibility, while extending the method's range of applicability. Indeed, the GLM-HOLP procedure proves to be competitive and robust, when its performance is compared to existing methods for generalised linear models, such as the (I)SIS variants proposed by Fan et al. (2009).

Various numerical experiments demonstrate that, especially under challenging circumstances, the GLM-HOLP is often among the best methods for feature screening, without being too computationally intensive.

There are several possible opportunities to extend the work conducted on the GLM-HOLP procedure. For instance, it would be of great interest to explore the applicability of the ridge regression version of HOLP to the class of generalised linear models. It would also be interesting to pursue extensions of the theoretical results presented in the original Wang and Leng (2015) paper, such as *screening consistency* and the *sure screening property*.

The main advantages that the (I)SIS procedure holds over the GLM-HOLP are its abilities to automatically determine a submodel size and to delete features iteratively. Indeed, certain improvements are made in this direction in the original HOLP paper. Building upon the work of Chen and Chen (2008) and Wang (2009), Wang and Leng (2015) make use of the extended BIC to determine an appropriate submodel. Therefore, it would be worthwhile to implement this extension in a generalised linear model setting, from where it would be possible to apply relevant one-stage methods.

Finally, alongside further simulation studies, there would also be merit in applying the GLM-HOLP procedure to other high-dimensional datasets, especially to a case study appropriate for analysis with Poisson regression. Following the results of the first simulation study, we are keen to further examine and compare the sensitivity of the GLM-HOLP method to other variable screeners in a Poisson regression setting.

Acknowledgements

I would like to express my gratitude to Dr. Xinghao Qiao for agreeing to supervise this project and for all the guidance provided. In addition, I would like to personally thank all of the staff and academics at the Department of Statistics for supporting my studies in the MSc. programme.

Bibliography

Agresti, A. (2015), *Foundations of linear and generalized linear models*, John Wiley & Sons.

Candes, E. and Tao, T. (2007), 'The dantzig selector: Statistical estimation when p is much larger than n', *The Annals of Statistics* pp. 2313–2351.

Chen, J. and Chen, Z. (2008), 'Extended bayesian information criteria for model selection with large model spaces', *Biometrika* **95**(3), 759–771.

Cho, H. and Fryzlewicz, P. (2012), 'High dimensional variable selection via tilting', *Journal of the Royal Statistical Society: series B (statistical methodology)* **74**(3), 593–622.

Fan, J. and Li, R. (2001), 'Variable selection via nonconcave penalized likelihood and its oracle properties', *Journal of the American Statistical Association* **96**(456), 1348–1360.

Fan, J. and Lv, J. (2008), 'Sure independence screening for ultrahigh dimensional feature space', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **70**(5), 849–911.

Fan, J. and Lv, J. (2010), 'A selective overview of variable selection in high dimensional feature space', *Statistica Sinica* **20**(1), 101.

Fan, J., Samworth, R. and Wu, Y. (2009), 'Ultrahigh dimensional feature selection: beyond the linear model', *Journal of Machine Learning Research* **10**(Sep), 2013–2038.

Friedman, J., Hastie, T. and Tibshirani, R. (2009), *The elements of statistical learning*, Springer series in statistics, 2nd ed. edn, Springer, New York.

Friedman, J., Hastie, T. and Tibshirani, R. (2010), 'Regularization paths for generalized linear models via coordinate descent', *Journal of statistical software* **33**(1), 1.

Hoerl, A. E. and Kennard, R. W. (1970), 'Ridge regression: applications to nonorthogonal problems', *Technometrics* **12**(1), 69–82.

James, G., Witten, D., Hastie, T. and Tibshirani, R. (2013), *An introduction to statistical learning*, Vol. 112, Springer.

Liu, J., Zhong, W. and Li, R. (2015), ‘A selective overview of feature screening for ultrahigh-dimensional data’, *Science China Mathematics* **58**(10), 1–22.

McCullagh, P. P. and Nelder, J. A. (1996), *Generalized linear models*, Monographs on statistics and applied probability (Series) ; 37, second edn.

Singh, D., Febbo, P. G., Ross, K., Jackson, D. G., Manola, J., Ladd, C., Tamayo, P., Renshaw, A. A., D’Amico, A. V., Richie, J. P. et al. (2002), ‘Gene expression correlates of clinical prostate cancer behavior’, *Cancer cell* **1**(2), 203–209.

Tibshirani, R. (1996), ‘Regression shrinkage and selection via the lasso’, *Journal of the Royal Statistical Society. Series B (Methodological)* pp. 267–288.

Wang, H. (2009), ‘Forward regression for ultra-high dimensional variable screening’, *Journal of the American Statistical Association* **104**(488), 1512–1524.

Wang, X. and Leng, C. (2015), ‘High dimensional ordinary least squares projection for screening variables’, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* .

Zhang, C. H. (2007), ‘Penalized linear unbiased selection’, *Department of Statistics and Bioinformatics, Rutgers University* **3**.

Zou, H. (2006), ‘The adaptive lasso and its oracle properties’, *Journal of the American statistical association* **101**(476), 1418–1429.

Zou, H. and Hastie, T. (2005), ‘Regularization and variable selection via the elastic net’, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **67**(2), 301–320.