Using a supervised Principal Components Analysis for variable selection in high-dimensional datasets reduces False Discovery Rates

Insha Ullah*[1,2], Kerrie Mengersen[1,2], Anthony Pettitt[1,2], and Benoit Liquet[1,2,3]

[1]School of Mathematical Science, Queensland University of Technology, Australia
[2]ARC Centre of Excellence for Mathematical & Statistical Frontiers (ACEMS)
[3]Laboratoire de Mathematiques et de leurs Applications Université de Pau et des Pays de l'Adour, UMR CNRS 5142, E2S-UPPA, Pau, France

Abstract

High-dimensional datasets, where the number of variables 'p' is much larger compared to the number of samples 'n', are ubiquitous and often render standard classification and regression techniques unreliable due to overfitting. An important research problem is feature selection — ranking of candidate variables based on their relevance to the outcome variable and retaining those that satisfy a chosen criterion. In this article, we propose a computationally efficient variable selection method based on principal component analysis. The method is very simple, accessible, and suitable for the analysis of high-dimensional datasets. It allows to correct for population structure which otherwise can induce spurious associations and is less likely to overfit. We expect our method to accurately identify important features but at the same time reduce the False Discovery Rate (FDR) through accounting for the correlation between variables and through de-noising data in the training phase, which also make it robust to outliers in the training data. Being almost as fast as univariate filters, our method allows for valid statistical inference (p-values and confidence intervals). To our knowledge, none of the modern multivariate statistical tools designed for today's high-dimensional data possess this quality of valid post selection inference. The superior performance of the method is demonstrated through a semi-real gene-expression datasets, a challenging childhood acute lymphoblastic leukemia (CALL) gene expression study, and through a genome-wide association study (GWAS) that attempts to identify single-nucleotide polymorphisms (SNPs) associated with the rice grain length.

Keywords: binary classification, case-control study, feature selection, gene-expression data, GWAS, QTL mapping, SVD.

1 Introduction

Many new challenges have been raised over recent years by high-dimensional data, especially so-called 'large p small n' data with thousands or tens of thousands of features but limited number of samples. This brings the standard analysis tools to halt or deteriorate the learning process as the learning algorithm becomes prone to over-fitting due to the abundance of irrelevant and redundant features. This means that the good performance of a fitted model could be limited to the training data set and its performance could be very poor for new unseen samples. An obvious challenge in such applications is feature selection, which is to filter out the features that have little to no correlation with the response variable based on the available data. Feature selection represents an integral component of modern statistical research, and methods that are computationally less demanding and that obtain reliable results are urgently needed.

*Correspondence: i.ullah1980@gmail.com

Recently, considerable effort has been directed to identifying important variables that contribute useful information towards a response variable of interest. The approaches that are commonly used can be broadly divided into two categories: univariate filters and multivariate feature selection methods. The univariate filters independently assess each feature based on its relevance to the response and reduce the dimension by filtering out all those features that are not strongly associated with a response variable [for example, see Smyth, 2004, Yu and Liu, 2004, Ullah et al., 2019]. Owing to their simplicity, interpretability and computational efficiency, the filters are more commonly used in the analysis of high-dimensional datasets. One of the drawbacks of univariate filters, however, is that dependencies among variables are ignored because the variables are assessed one a time. As a result, a substantial amount of information contained in the data is left unutilized.

The multivariate variable selection methods overcome this drawback and use all the variables simultaneously to choose the best subset of variables. Many multivariate variable selection methods have been proposed. For example, the lasso regression [Tibshirani, 1996] and its improved variants like the elastic-net regression [Zou and Hastie, 2005] are widely used because they produce sparse solutions and hence perform the model selection simultaneously with parameter estimation. An approach that uses an ensemble of classification or regression trees known as Random Forest is also a popular algorithm [Breiman, 2001]. The algorithm returns measures of variable importance and has been used for gene selection by Díaz-Uriarte and De Andres [2006]. However, both lasso regression and Random Forest were designed for datasets of a moderate size (hundreds of variables) and rely on computationally cumbersome methods such as cross-validation for tuning penalty parameters. For high-dimensional datasets, these methods may be statistically inaccurate [He and Lin, 2010, Fan et al., 2011].

An intermediate approach reduces the dimensionality of data to a manageable size by first using a univariate filter, which is then followed by an appropriate multivariate method to do the final selection. This is adopted by Fan and Lv [2008] who introduced iteratively sure independent screening (ISIS) in the context of linear regression models and showed that it is a suitable variable selection strategy for high-dimensional settings. Their procedure couples an independent screening of a large number of variables based on their marginal correlations with the final variable selection and parameter estimation using a more sophisticated multivariate technique such as elastic-net logistic regression [Zou and Hastie, 2005]. The method has been used for variable selection with a range of high-dimensional datasets [for example, see Pi and Halabi, 2018]. The ISIS idea is also extended to logistic regression of case control data by He and Lin [2010]. However, their method still suffers from the drawbacks of computational inefficiency and statistical inaccuracy at the second stage of the final variable selection if the number of filtered variables is large compared to the number of available samples.

Principal component analysis (PCA) has been used for dimension reduction in high-dimensional data because it is statistically coherent, computationally faster and scalable [Price et al., 2010]. A disadvantage of using PCA as a dimension reduction technique is that the principal components (PCs) are linear combinations of all original variables and often may not be easy to interpret. Even if the PCs are interpretable, which is sometimes the case for the first few PCs, one often needs interpretation in terms of original variables. For example, one might be interested in identifying genes that are associated with a particular disease. Attempts have been made to use PCA for variable selection rather than variable extraction [for example, see McCabe, 1984, Jolliffe, 1972]. However, these methods are computationally infeasible for high-dimensional datasets. Shen and Huang [2008] introduced sparse PCA that produce PCs with sparse loadings (a.k.a coefficients) through forcing near zero loading to be exactly zero. The variables thus associated with the non-zero loadings in the first few PCs are considered informative. More recently, Wathen et al. [2019] have adapted PCA for variable selection for high-dimensional genotype data coded 0, 1, or 2 based on the reference allele counts. They have ranked the original variables based on their weights in the first two PCs. These approaches, however, are arguably too arbitrary to be used for general purpose variable selection.

In this study we propose to use PCA for variable selection in a binary classification problem or a case-control study. We achieve this by using data from one class to fit a PCA model. The estimated parameters thus obtained are used to reconstruct the data from another class. The variables that are not reconstructed accurately are the important predictors to distinguish between the two classes. The method is effective even for small sample sizes. The results, however, could be improved by accurately estimating the parameters in the training phase. Therefore, we use the class with the larger sample size for training so that the estimated parameters are as accurate as possible. In some cases it may be easier to acquire a larger sample size for one class compared to the other. For example, healthy subjects may be abundant while the diseased subjects may be rare. Training of a classification technique, such as penalized logistic regression and Random Forest, usually requires a sufficiently large number of samples in each class. In our case, in such case-control association studies we could use the data from healthy subjects for fitting a PCA model. A simple procedure to obtain p-values corresponding to each variable is also proposed since such a measure is relatively easier to interpret for decision making purposes. For illustration we focus on a genomics problem, emphasizing that our method is much more widely applicable. We consider the performance of this approach in a semi-real gene expression study where the true differentially expressed genes are known. We further evaluate the method through application to two publicly available high-dimensional datasets: a childhood acute lymphoblastic leukemia gene expression dataset and a genome-wide association study dataset. Our method is straightforward to understand, computationally efficient, easy to implement and empirical evaluation on several real-world datasets suggests that it is more effective in variable selection compared to existing methods.

2 Methods

PCA is traditionally used as a dimension reduction technique and is a useful tool to visualize high-dimensional data in a manageable low-dimensional space [Pearson, 1901, Hotelling, 1933, Alter et al., 2000, Jolliffe, 2002, Ringnér, 2008, Jolliffe and Cadima, 2016, McTavish et al., 2013]. It uses an orthogonal transformation to transform a set of p (possibly correlated) observed variables into another set of p uncorrelated variables termed principal components (PCs). PCs are uncorrelated linear functions of the originally observed variables that successively maximize variance such that the first PC stands for the axis along which the observed data exhibit the largest variance; the second PC stands for the axis that is orthogonal to the first PC and along which the observed data exhibit the second largest variance; the third PC stands for the axis that is orthogonal to the first two PCs and along which the observed data exhibit the third largest variance, and so on. In this way, the p orthogonal dimensions of variability that exist in the data are captured in p PCs and the proportion of variability that each PC accounts for accumulates to the total variation in the data. The sole objective of PCA is to capture as much variation as possible in the first few PCs. It is, therefore, often the case that the first q ($q \ll p$) PCs retain conceivably useful information in the observed data and the rest contain variation mainly due to noise [Jolliffe, 2002].

More formally, let y_{ij} denote a real-valued observation of the jth variable made on the ith subject, where $i = 1, \ldots, n$ and $j = 1, \ldots, p$. Assume that the n observations are arranged in n rows of an $n \times p$ data matrix Y with columns corresponding to p variables or features. We standardize columns of Y to have zero mean and unit standard deviation, and store the resultant values in a data matrix X, that is, the elements x_{ij} of X are obtained by $x_{ij} = (y_{ij} - \bar{y}_j)/s_j$, where $\bar{y}_j$ and s_j are the mean and standard deviation of the jth column of Y, respectively. The PCA can be performed by singular value decomposition (SVD) of X; that is, the $n \times p$ matrix X of rank $r \leq \mathsf{min}(n, p)$ is decomposed as

$$X = U\Gamma V^T, \tag{1}$$

where U is a $n \times r$ orthonormal matrix ($U^T U = I_r$), Γ is a $r \times r$ diagonal matrix containing r non-negative singular values in decreasing order of magnitude on the diagonal and V is a $p \times r$ matrix with orthonormal columns ($V^T V = I_r$). Denote the sample correlation matrix of X by R, which can be expressed as

$$R = \frac{1}{n-1} X^T X = V \Lambda V^T, \tag{2}$$

where $\Lambda = 1/(n-1)\Gamma^2$ is a $r \times r$ diagonal matrix containing r non-zero positive eigenvalues $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_r)^T$ of matrix R on the diagonal in decreasing order of magnitude. It follows that the r columns of matrix V

contain the eigenvectors of X^TX and hence are the desired directions of variation. The derived set of r transformed variables (the PCs) are obtained by

$$Z = XV.$$

It is important to note that the matrix U above contains standardized PCs in its columns and is a scaled version of Z, which is provided additionally in (1). To see this, multiply (1) on the right by V to obtain

$$Z = XV = U\Gamma.$$

In practice, the first $q \ll r$ major components are of greater interest since they account for most of the variation in the data. Without undue loss of information, the dimension of Z may thus be reduced from p to q, that is

$$\tilde{Z} = X\tilde{V}, \tag{3}$$

where $\tilde{V}$ is a $p \times q$ matrix that contains the first q columns of V and $\tilde{Z}$ contains the first q PCs in its columns. The set of first q PCs is a lower dimensional representation of a p-dimensional dataset and can be used to uncover trends and patterns in the data, which is probably the most prevalent application of PCA. In addition, the most informative PCs can be used in subsequent analyses of the data [for example, see Price et al., 2006]. However, there are concerns with this approach. First, like the p original variables, the q PCs (although informative because they account for most of the variation in the data) are not easy to interpret [McCabe, 1984, Zou et al., 2006] and often misleading or imaginary interpretations are associated with PCs [Jolliffe, 2002]. Second, in high-dimensional datasets the level of noise due to high-dimensionality often obscures useful reproducible patterns (true structure rather than random association) in the data [Golub et al., 1999]. Third, the PCs are developed in an unsupervised manner so may not be optimal for further analyses such as regression. In such cases it is assumed that only a small number of variables are able to uncover interpretable patterns in the observed data and variable selection (rather than extraction) methods are usually preferred.

A lower rank approximation of X can be obtained by

$$\tilde{X} = \tilde{Z}\tilde{V}^T, \tag{4}$$

which is the best approximation of X in the least-squares sense by a matrix of rank q [Eckart and Young, 1936, Saporta and Niang, 2009]. The value of q is chosen using available heuristic options such as a plot of eigenvalues (or singular values) in decreasing order of magnitude, known as a scree-plot or the cumulative proportion of explained variance (CPEV), which chooses a smallest value of q for which the CPEV exceeds a pre-specified quantity ϕ that takes value between 0 and 1. Commonly used values of ϕ ranges from 0.7 to 0.9 [Jolliffe, 2002]. A value of $\phi = 0.8$ means that the chosen number of components q explains at least 80% of the total variance in the observed data.

PCA is computationally efficient and scalable to high-dimensional datasets. In this paper, we adapt PCA to variable selection in a classification problem—rather than its traditional use of variable extraction and thereby visualization. We achieve this by considering the error induced by the approximate reconstruction of an actual observation x_{ij} using the expression in (4); that is, the error is defined by

$$\hat{\varepsilon}_{ij} = x_{ij} - \tilde{x}_{ij}, \tag{5}$$

where $\tilde{x}_{ij}$ are the values contained in the ith row and jth column of $\tilde{X}$. We store the elements $\hat{\varepsilon}_{ij}$ in an $n \times p$ residual matrix $\hat{\epsilon}$.

In a case-control study, assume that Y^- is an $n_1 \times p$ data matrix containing the data from the n_1 control samples and Y^+ is an $n_2 \times p$ data matrix containing observations from the n_2 cases. We standardize the p columns of Y^- to have zero mean and unit standard deviation, and store the resultant values in a matrix X^-, that is, the elements x_{ij}^- of X^- are obtained by $x_{ij}^- = (y_{ij}^- - \bar{y}_j^-)/s_j^-$, where $\bar{y}_j^-$ and s_j^- are the mean and standard deviation of the jth column of matrix Y^-, respectively. We decompose X^- as given in (1) and estimate the parameter V^-. We also standardize the columns of Y^+ using the means and standard deviation of the corresponding columns in Y^- and store the resultant values in X^+; that is, the elements x_{ij}^+ of X^+ $x_{ij}^+ = (y_{ij}^+ - \bar{y}_j^-)/s_j^-$. We reconstruct X^+ by substituting the estimated parameter V^- in (4). The reconstruction error of X^+ is computed using the expression given in (5) as

$$\hat{\varepsilon}_{ij}^+ = x_{ij}^+ - \tilde{x}_{ij}^+, \tag{6}$$

where $\tilde{x}_{ij}^+$ is the low rank approximation of x_{ij}^+. If a variable is differentially expressed between X^- and X^+, this will be reflected in the reconstruction error. The larger the reconstruction error the larger will be the discriminative power associated with that particular variable. We compute the mean of the jth column of $\hat{\epsilon}^+$ and denote it by t_j^+; that is,

$$t_j^+ = \frac{1}{n_2}\sum_{i=1}^{n_2} \varepsilon_{ij}^+. \tag{7}$$

If a variable in X^+ is generated by a different model then its corresponding t_j^+ will follow a different distribution. In high-dimensional problems only a small subset of variables usually turns out to be useful in terms of their predictive power. In such situations the realizations of t_j^+ associated with the majority of unimportant variables will behave similarly in distribution. The differential expression of variables in two groups or the class prediction power of a variable will be reflected by its associated value of t_j^+. In other words, the values of t_j^+ that correspond to variables that are differentially expressed between the two groups will deviate more from the center—zero—of the distribution of t_j^+. The further away a t_j^+ is from zero, the better its associated variable would perform as a predictor in classification. One can thus rank the p variables based on the magnitude of the statistic t_j^+ from most important to unimportant.

One can even calculate p-values associated with t_j^+ if the elements of $\hat{\epsilon}^+$ are assumed to be independent and distributed according to the Gaussian distribution; that is,

$$\varepsilon_{ij}^+ \sim N(0, \sigma^2),$$

where σ^2 is a variance that needs to be estimated. This is a common assumption generally made when inference procedures are developed for a PCA model [for example, see Choi et al., 2017]. If X^- and X^+ are generated by the same model then the distribution of t_j^+ is easy to characterise; that is,

$$t_j^+ \sim N(0, \frac{\sigma^2}{n_2}). \tag{8}$$

This turns a variable selection problem to an outlier identification problem and a variable that is associated with an outlying t_j^+ will have discriminative power.

Note that using sample variance as an estimator of σ^2 may obscure the outlying values of t_j^+ since the sample variance is not robust against outliers. It is important to use a robust estimate of σ^2 such as mean absolute deviation (MAD), which has a 50% breakdown point [Hampel, 1974] as opposed to sample standard deviation, which has 0% breakdown point. As noticed above, in high-dimensional problems one usually does not expect to have a large number of significant variables (since MAD breaks down if the proportion of significant variables is larger than 50%). The MAD estimator of σ is

$$\hat{\sigma} = 1.4826 \ \mathsf{Median}\{|\varepsilon_{ij}^+ - \mathsf{Median}(\varepsilon_{ij}^+)|\}.$$

The above procedure could be used to identify significant variables even if we have a single case sample (many healthy baseline observations vs. a single case). It could be particularly useful when one wishes to identify differentially expressed genes in a gene expression profile of a small number of available patients. In some situations it may be that some of the samples in the case group are different with respect to some variables even though they all belong to a single group. For example, one particular type of leukaemia might have some unknown subtypes and identifying subtypes of a disease could be useful in precision medicine and personalized treatments [Collins and Varmus, 2015, Ahlqvist et al., 2018]. The one-by-one analysis makes it possible to further identify those variables which are differentially expressed and are common (or uncommon) to all patients in the same group. This makes our procedure different and more advantageous from the variable selection procedures that are commonly used in practice.

In the rest of this document we refer to our PCA based approach as Variable Selection based on PCA (VSPCA) and formalize it in the following algorithm:

1. Obtain X^- by standardizing the data matrix Y^-; that is, $x_{ij}^- = (y_{ij}^- - \bar{y}_j^-)/s_j^-$.
2. Estimate the parameter V^- using the control sample X^- and the expression in (1).
3. Obtain X^+ using $x_{ij}^+ = (y_{ij}^+ - \bar{y}_j^-)/s_j^-$.
4. Use the proportion of total variance explained criterion or scree-plot to choose q and calculate $\tilde{X}$ using the estimates $\tilde{V}^-$ and the expression in (4).
5. Compute $\hat{\varepsilon}_{ij}^+$ using (6).
6. Estimate σ using a robust MAD estimator given by

$$\hat{\sigma} = 1.4826 \ \mathsf{Median}\{|\varepsilon_{ij}^+ - \mathsf{Median}(\varepsilon_{ij}^+)|\}.$$

7. Calculate t_j^+ using (7).
8. Calculate p-values under the assumption that $t_j^+ \sim N(0, \frac{\sigma^2}{n_2})$. A smaller p-value indicates the inconsistency of associated variable in X^+ with the one in X^- and hence is the important variable.

Note that in some circumstances one may be wish to reduce the number of variables from p to a smaller number k (i.e., to filter out the $p - k$ least important variables). This is also useful when the number of variables is small, in which case one does not have enough data to estimate σ, or when the required normality assumption for the error is not valid. In such cases step-8 of the algorithm can be dropped and variables can be ranked based on the relative magnitude of t_j^+. Those variables that correspond to the k largest magnitudes of t_j^+ can be selected as the superset of predictors. Note also that the p-values obtained via VSPCA need to be adjusted for multiple testing. Unless otherwise specified, all p-values reported in this paper are adjusted for multiple testing via the 'fdr' method [Benjamini and Hochberg, 1995] implemented in the p.adjust() function of the R 'stats' package.

We recommend examining the quantile-quantile plot (qq-plot) of t_j^+ and the histograms of unadjusted p-values for diagnostic purposes. The distribution of t_j^+ is expected to be normally distributed and in typical high-dimensional problem where a large number of variables are irrelevant, the distribution of t_j^+ should be normal at least at the center. The heavier tails of the distribution in the qq-plot indicate significant variables. Similarly, under the null hypothesis of no relevant variable, one would expect the two-tailed p-values to be uniformly distributed between 0 and 1. If there are informative variables, this would be indicated in the histogram by a sharp spike near 0. The rest of the shape is still expected to be uniform.

3 Applications

3.1 Analysis of Platinum Spike Dataset

We test and illustrate our new variable selection procedure by applying it to the Platinum Spike [Zhu et al., 2010] dataset. This is one of the largest semi-real dataset for which the true differentially expressed genes are known. Hence, it has been widely used for the benchmarking of the microarray analysis methods [for example, see Hochreiter et al., 2006, Roca et al., 2017, and references therein].

The dataset is produced by a controlled experiment [Zhu et al., 2010] with two experimental conditions (condition-A and condition-B) and nine (three biological × three technical) replicates per condition. Gene expression data were obtained with Affymetrix Drosophila Genome 2.0 microarrays. The dataset has a total of 18,769 probe sets (excluding Affymetrix internal control probes), of which 5,587 were spiked-in. Among these, 1,940 (34.7%) were differentially expressed (known positives) to varying degrees between 1.2- and 4-fold (1,057 over-expressed, 883 under-expressed), while the remaining 3,406 (61.0%) were spiked-in at the same concentration in both conditions (known negatives).

Although the sample size of the training set (whichever of condition-A and condition-B one wishes to choose as a training set of data) is very small ($n_1 = 9$), this dataset is worth considering here because it has been widely used to benchmark differentially expressed genes methods [Dembélé and Kastner, 2014]. The analysis of this dataset will help understand the performance of VSPCA under this common scenario of very small sample size.

We normalized the dataset using the Affymetrix MAS5 algorithm implemented in the *mas5* function of the R package 'affy' [Gautier et al., 2004]. We applied our variable selection procedure to the processed data. The PCA model was trained on data produced under condition-B. The scree-plot is shown in Figure 1(a) and the CPEV plot is give in Figure 1(b). The scree-plot showed a mild elbow at $q = 3$. These three components together explained 42.9% of the total variation in the training data. We set $\phi = .7$ to exclude three components for the purpose of denoising (the variance explained by component-9 is zero), which led us to choose $q = 6$ to obtain the low-rank approximation of the data produced under condition-A. For comparison purposes we also present the results obtained from the univariate moderated t-test [Smyth, 2004], which is a popular choice for the analysis of data from microarrays experiments [Caiazzo et al., 2011] and is implemented in the 'limma' package of R [Ritchie et al., 2015] (We will refer to this test as MTT for the rest of the paper.)

The qq-plot is given in Figure 2 with heavy tails indicating a large number of differentially expressed genes between the two experimental conditions. The VSPCA led to lower False Positive Rate (FPR) (see Figure 3(a)) in comparison to MTT (see Figure 3(c)). The two methods, however, performed almost equally well in identifying true differentially expressed genes between the two experimental conditions (see Figures 3(b) and 3(d)). The p-values of VSPCA are shown in Figure 4 in comparison to the p-values obtained using MTT. The p-values of VSPCA for the true negatives were larger than the p-values based on MTT (see Figure 4a) while the p-values of VSPCA for true positives are smaller compared to the p-values of MTT (see Figure 4b). In addition, for the true positives the VSPCA further demarcated the genes that have the highest discriminative power (the super predictors). Based on the ROC curve in Figure 5, our multivariate approach appears to be performing equally as well as the univariate MTT.

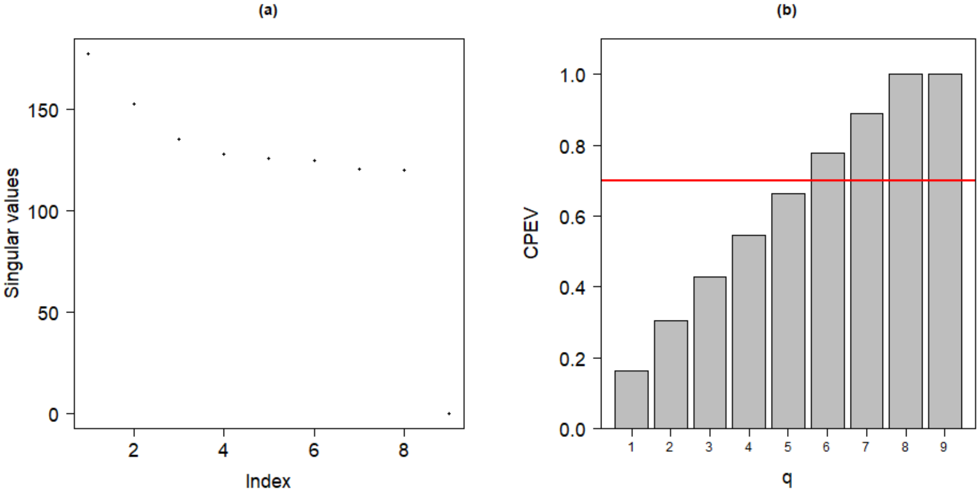

Figure 1: The plot in (a) shows the scree plot for the training data (9 samples produced under condition-B) of Platinum spike dataset and the plot in (b) shows the cumulative proportion of explained variance (CPEV) for each value of q with the horizontal line in red indicating $\phi = .7$.

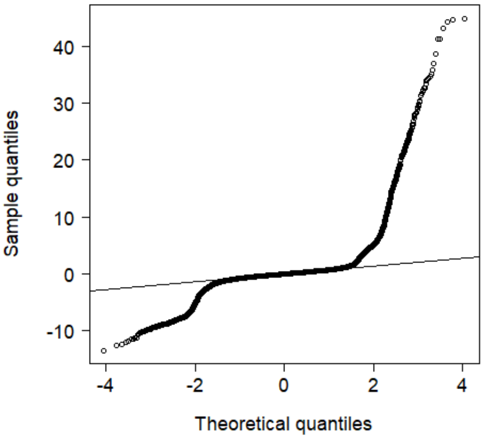

Figure 2: Platinum spike dataset. Quantiles of $\sqrt{n_2}t_j^+/\hat{\sigma}$ against the quantiles of standard normal distribution.

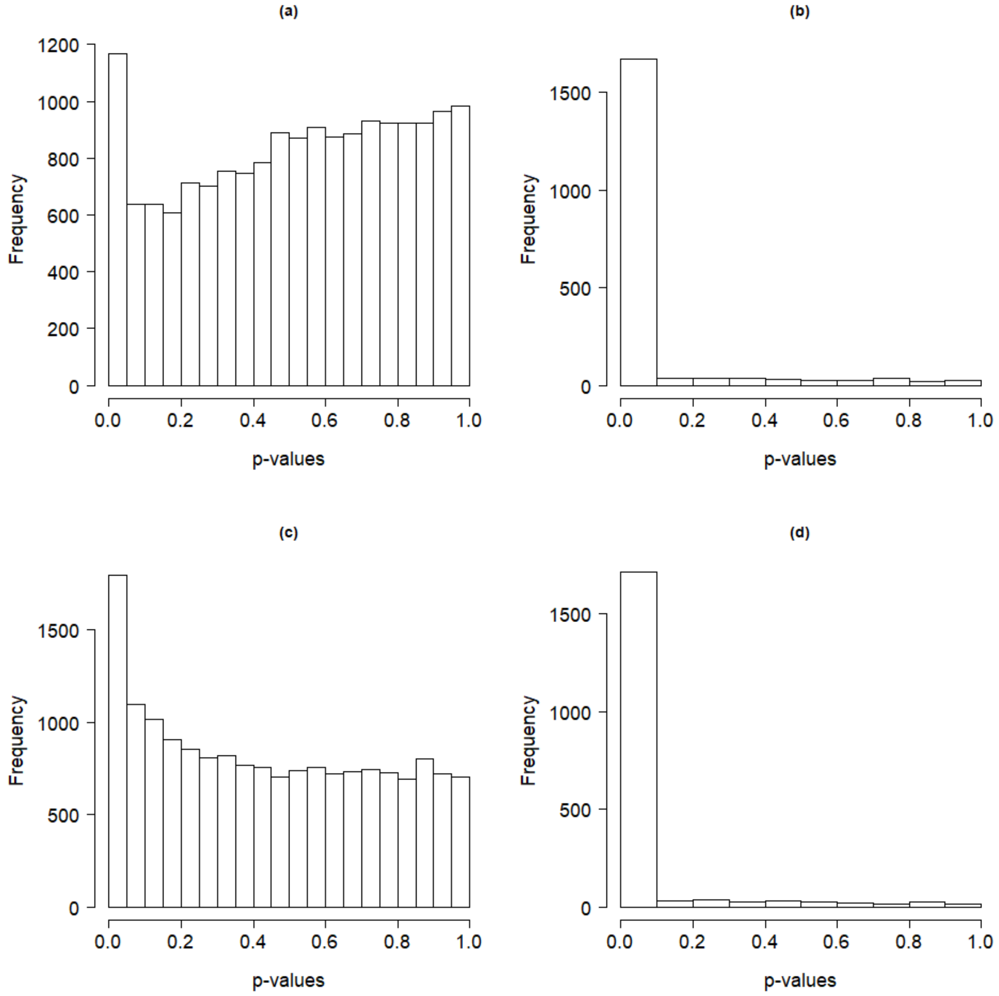

Figure 3: Histograms of unadjusted p-values for Platinum spike dataset: p-values calculated for (a) known negatives using VSPCA, (b) known positives using VSPCA, (c) known negatives using MTT, and (d) known positives using MTT.

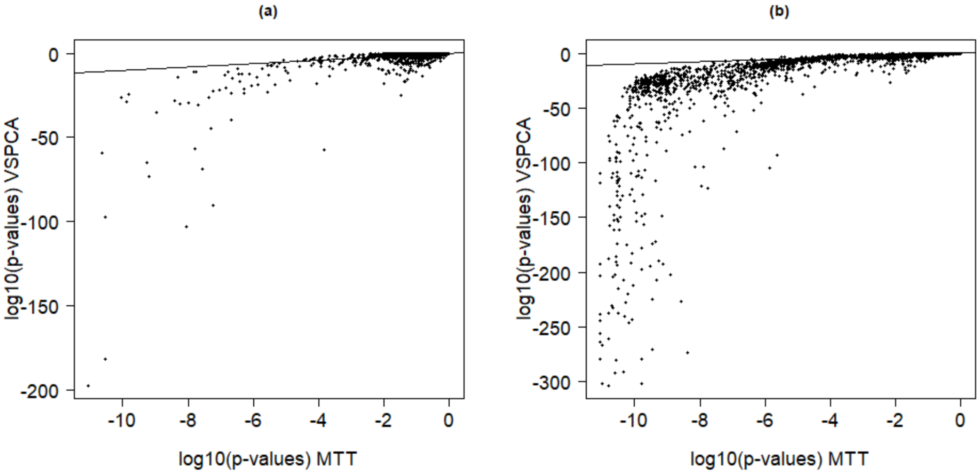

Figure 4: Logarithm of p-values obtained using VSPCA against logarithm of p-values obtained using MTT for Platinum spike dataset: (a) p-values for known negatives (b) p-values for known positives. The equality, $x = y$ values, reference lines are added.

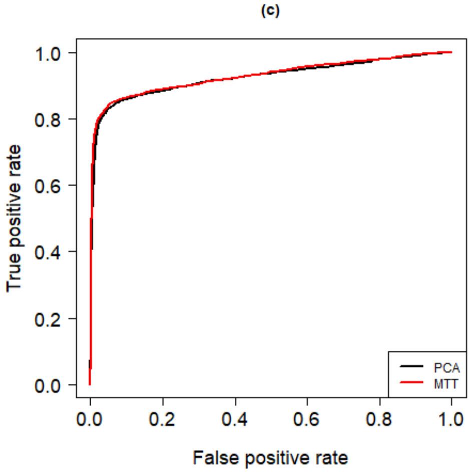

Figure 5: ROC curve based on VSPCA in comparison to MTT for Platinum spike dataset.

3.2 Childhood acute lymphoblastic leukemia (CALL) gene expression study

To provide additional evidence of the performance of VSPCA, we chose a childhood acute lymphoblastic leukemia (CALL) high-throughput gene-expression dataset that is studied in detail by Den Boer et al. [2009] and accessible through GEO Series accession number GSE13425. The data had 22,283 genes and 190 samples in total. The 190 samples are from different subtypes of CALL: BCR-ABL (4 samples), BCR.ABL.hyperdiploidy (1 sample), E2A-rearranged.E (1 sample), E2A-rearranged.Esub, (4 samples), E2A-rearranged (8 samples), hyperdiploid (44 samples), MLL (4 samples), pre.B.ALL (44 samples), T.ALL (36 samples), TEL.AML1 (43 samples), and TEL.AML1.hyperdiploidy (1 sample). This allows us to evaluate the performance of the method under a variety of sample sizes.

We normalized the data using the Affymetrix MAS5 algorithm implemented in the R package 'limma' [Ritchie et al., 2015]. The normalized data was then used to rank and identify genes that are associated with the class labels using our VSPCA method and the MTT.

We used hyperdiploid subtype samples ($n_1 = 44$) as a control group to train our PCA model. The T.ALL subtype samples ($n_2 = 36$) were used as cases to identify genes that are differentially expressed from hyperdiploid subtype samples. A scree plot based on hyperdiploid subtype samples is shown in Figure 6, which suggested $q = 7$. However, we used the proportion of explained variance criteria which we believe protects against unreasonably smaller choices of q. We set $\phi = .8$, which yielded $q = 33$ to reconstruct the test set of data (cases group). The qq-plot of t_j^+ is given in Figure 7 and the histograms of unadjusted p-values are shown in Figure 8(a) and Figure 8(b). The heavier tails of the distribution of t_j^+ in the qq-plot indicate differentially expressed genes and the distribution seems to be normal around the center. The histograms of unadjusted p-values obtained using VSPCA and MTT revealed a large number of null genes with a spike near zero indicated a number of differentially expressed genes. The p-values obtained using VSPCA are plotted against the p-values obtained using the MTT in Figure 8(c). The significant p-values obtained via VSPCA were found to be smaller in comparison to the significant values obtained via MTT. This has an advantage when it comes to correction for multiple testing; that is, the smaller a p-value, the higher the chance of remaining in the set of significant p-values once corrected for multiple testing.

The MTT identified a large number of differentially expressed genes (4,980 at 0.01 level of significance) compared to VSPCA, which identified 582 differentially expressed genes. To explore the discriminative power of the large number of additional genes identified by MTT, we performed post-selection PCA analysis using only the 582 genes selected based on VSPCA and also using the selected superset of 4,980 genes based on MTT. It turns out that the smaller superset of 582 genes selected using VSPCA has higher discriminative power than the much larger superset of 4,980 genes selected using MTT (see Figure 9). The PCA plot based on the superset selected using VSPCA is dominated by between-group variation (see Figure 9(a)). It can also be observed that the hyperdiploid samples (the training data) are tightly clustered together in comparison to the T.ALL samples (the test data). This is because we took into account 33 components and dropped the rest which in general are believed to represent noise and possibles outlying effects of some of the observations. This makes the procedure robust to outliers and unusual observations in the training data. In contrast, the plot based on the superset selected using MTT is dominated by within-group variation (see Figure 9(b)). This indicates that the MTT performance is affected by the noise, which led to a higher FPR. As expected, our new approach can substantially reduce the FPR by accounting for correlation between variables and by dropping the minor components that are often believed to stand for noise and unusual observations in the data.

To see the stability of the above results and to further test the performance of VSPCA under a smaller n_2 (smaller set of test data), we re-sampled the 36 T.ALL subtype samples. As above, we used hyperdiploid samples ($n_1 = 44$) as a training set of data and T.ALL samples as a test set of data. The value of n_2 was reduced gradually from 36 to 6 by successively subtracting 5 each time. For all values of $n_2 < 36$ that were considered here, we selected a sample without replacement from the test set of data and performed variable selection based on VSPCA and MTT using a level of significance $\alpha = 0.01$. The experiment was repeated 1000 times for each value of $n_2 < 43$. The variables that were significant under smaller n_2 were matched with the ones that were selected under the entire available test data sample (VSPCA selected 582 genes and MTT selected 4,980 genes as above) were counted. We compared the proportions of significant variables under entire sample size that were also significant under smaller sample sizes in more than 50% of the replicates (see Figure 10). Note that the choice of 50% ($> 500/1000$) is an arbitrary threshold but a similar pattern could be achieved by changing the threshold. The number of selected variables dropped much faster for MTT than it did for VSPCA as we reduced n_2. With $n_2 = 6$ the superset of 4,980 genes selected through MTT dropped by 63.5%. However, the superset of 582 genes selected through VSPCA dropped by 21.6%, which not only showed the stability of VSPCA but also showed that VSPCA could be more useful when the number of samples in the test class is smaller.

The VSPCA together with the post-selection PCA is useful when one is interested in identifying possible subgroups in the test data. To this end, we used the hyperdiploid subtype samples as training data and kept the training phase as it was above. However, this time we used all the rest of the subtypes as control data. Based on the VSPCA 293 genes were found to be differentially significant at the 0.01 level of significance while the number of differentially expressed genes were 2582 when MTT was used. The PCA plots based on the identified differentially expressed genes are shown in Figure 11. The 293 genes chosen based on VSPCA reveal similarities and differences between the subtypes more clearly as compared with the 2582 genes chosen based on MTT.

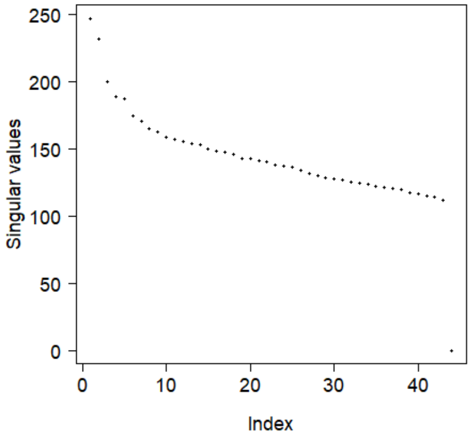

Figure 6: Scree plot based on hyperdiploid subtype samples of CALL dataset.

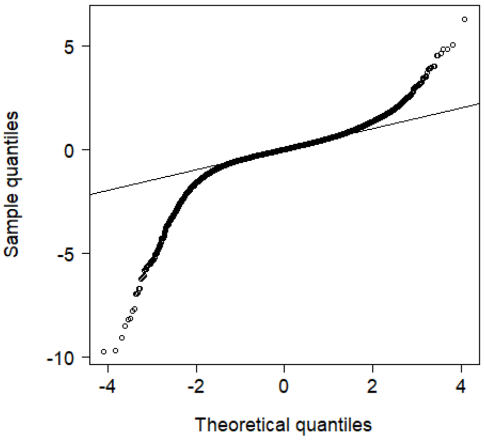

Figure 7: Quantiles of $\sqrt{n_2}t_j^+/\hat{\sigma}$ against the quantiles of standard normal distribution.

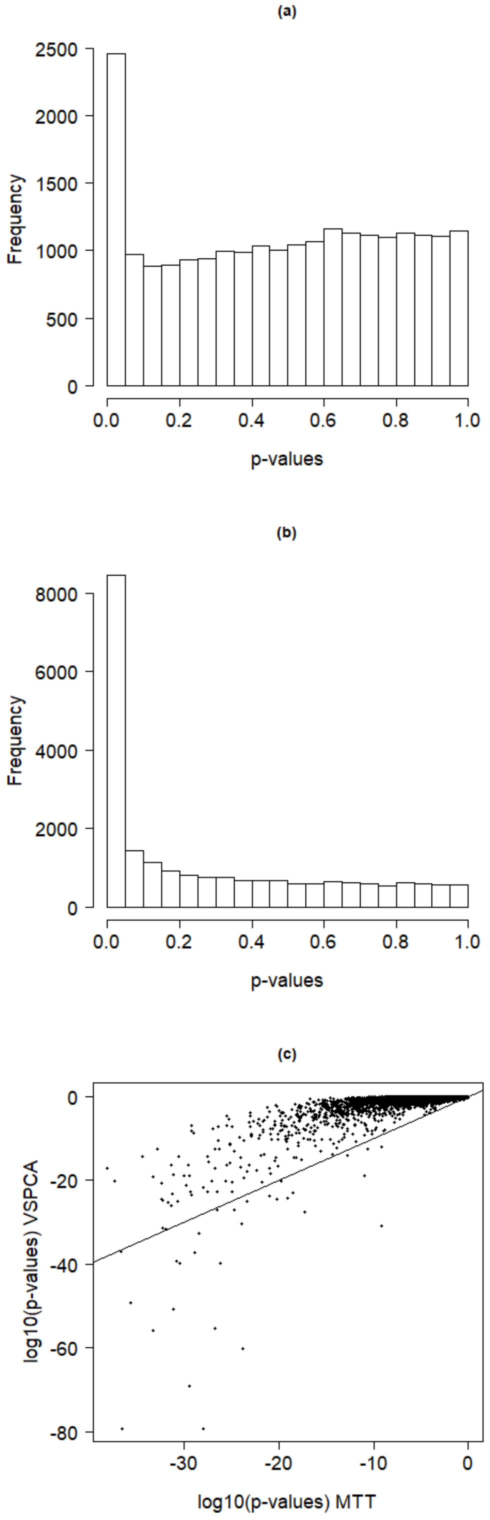

Figure 8: Plots of p-values: (a) Unadjusted p-values obtained via VSPCA, (b) Unadjusted p-values obtained via MTT, and (c) Logarithm of adjusted p-values obtained using VSPCA against logarithm of p-value obtained using the MTT (the equality, $x = y$ values, reference line is added).

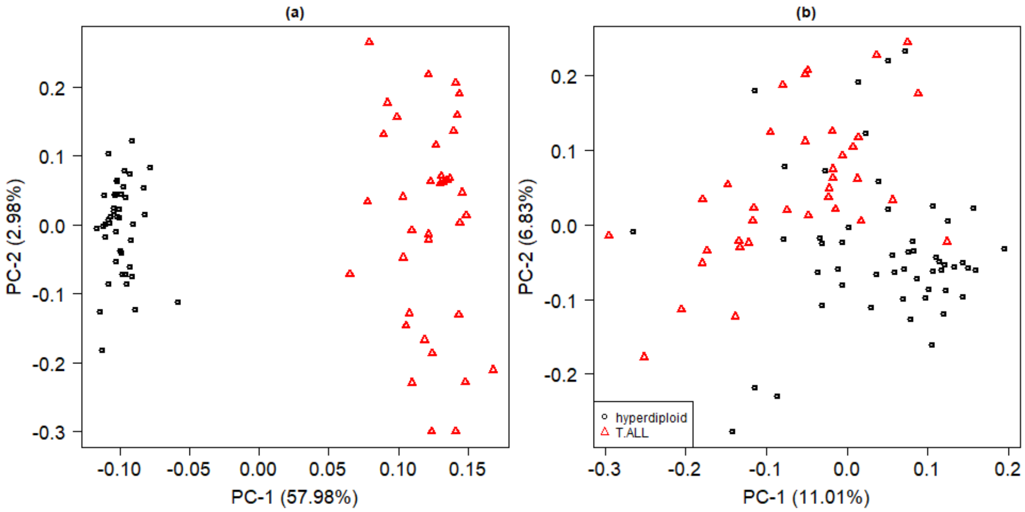

Figure 9: PCA plots based on the identified differentially expressed genes between hyperdiploid and T.ALL subtypes of the CALL dataset using: (a) VSPCA and (b) MTT.

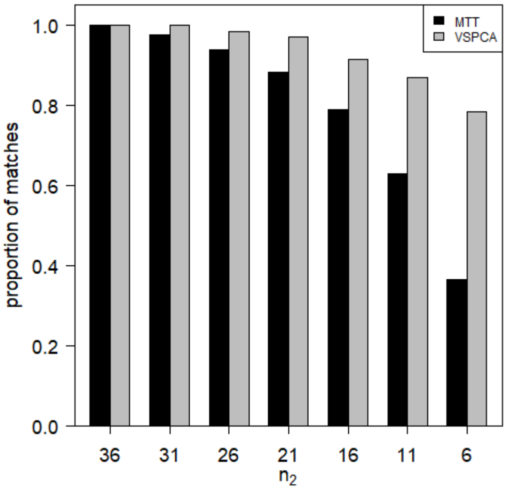

Figure 10: We used hyperdiploid samples ($n_1 = 44$) as a training set of data and T.ALL samples ($n_2 = 36$) as a test set of data. The value of n_2 was reduced from 36 to 6 by successively subtracting 5 (as shown on the x-axis from left to right). For all values of $n_2 < 36$ that are considered, we selected a sample without replacement from the test set of data and performed variable selection based on VSPCA and MTT using the nominal level of significance $\alpha = 0.01$. The experiment is repeated 1000 times for each value of $n_2 < 36$. The variables that turned out to be significant under smaller sample sizes and matched the ones that were selected under the entire available sample size for the test data are counted. Shown are the proportions of significant variables under the entire sample size that were also significant under smaller sample sizes in more than 50% (500/1000) of the replicates.

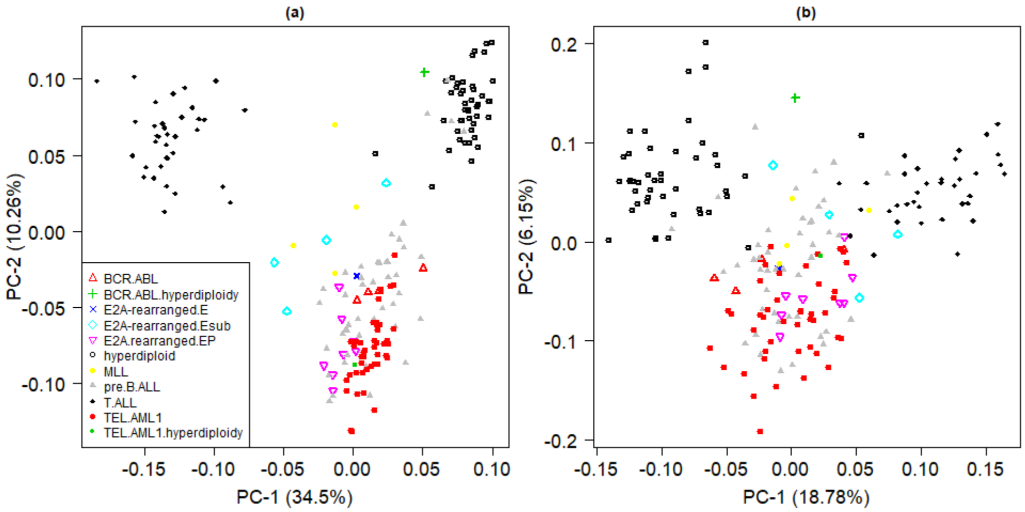

Figure 11: PCA plots based on the identified differentially expressed genes between hyperdiploid and the rest of the subtypes of the CALL dataset using: (a) VSPCA and (b) MTT.

To further evaluate our method, we changed the test set of data; that is, to train the PCA model we again used hyperdiploid samples but this time we used TEL.AML1 subtype ($n_2 = 43$) as a test set of data. The number of genes that were differentially expressed between hyperdiploid subtype and TEL.AML1 subtype, at 0.01 level of significance, was 373 for VSPCA and 2196 for MTT. The MTT again identified a larger number of genes comapred to VSPCA. The histograms of unadjusted p-values are provided in Figure 12(a) and in Figure 12(b) for VSPCA and MTT, respectively. The p-values under VSPCA are plotted against the p-values under MTT in Figure 12(c). As above, we performed post-selection PCA analysis using only the superset of 373 genes selected by VSPCA and also using the superset of 2,196 genes selected by MTT. The PCA analysis presented in Figure 13 shows a clear split between the observations for the two groups for both supersets. However, the VSPCA superset of 373 genes separates the two groups better (more between-group variation and less within-group variation) and therefore contains stronger predictors. The less clear separation with a much larger superset of predictors identified by MTT is evidence of higher FPR.

We again performed the analysis to evaluate the stability of the chosen superset of predictors where we reduce the number of samples in the test set of data. We used hyperdiploid samples ($n_1 = 44$) as a training set of data and re-sampled the 43 TEL.AML1 subtype samples. The value of n_2 was reduced gradually from 43 to 3 by successively subtracting 5. For all values of $n_2 < 43$ that are considered, we selected a sample without replacement from the test set of data and conducted variable selection based on both the VSPCA and the MTT. The experiment was repeated 1000 times for each value of $n_2 < 43$. The variables that turned out significant (at 0.01 level of significance) under smaller sample sizes that matched the ones that were selected under the entire available sample size (VSPCA selected 487 genes and MTT selected 2,196 genes as above) for the test data were counted. We display in Figure 14 the proportions of significant variables for the original complete sample with$n_2 = 43$ that were also significant under a smaller value of n_2 in more than 50% of the replicates. Again, the number of selected variables dropped much more quickly for MTT than it did for VSPCA as we reduced n_2. This highlights the stability of VSPCA as well as the usefulness of VSPCA under a smaller sample size of the test set of data.

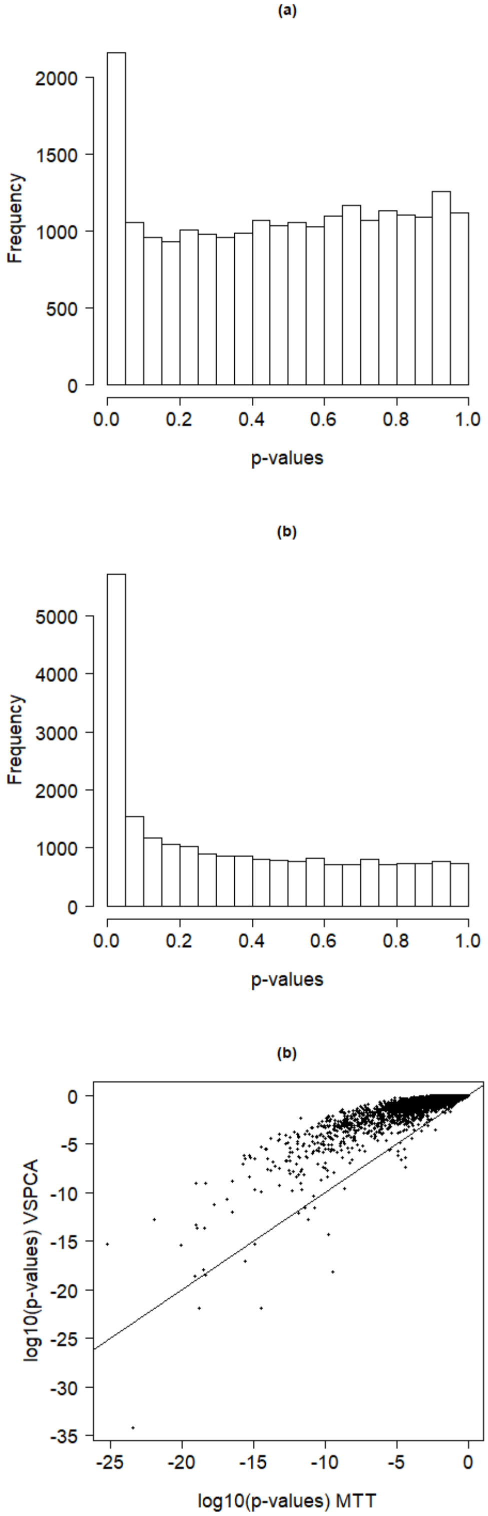

Figure 12: Plots of p-values: (a) Unadjusted p-values obtained via VSPCA, (b) Unadjusted p-values obtained via MTT, and (c) Logarithm of adjusted p-values obtained using VSPCA against logarithm of p-value obtained using the MTT (the equality, $x = y$ values, reference line is added).

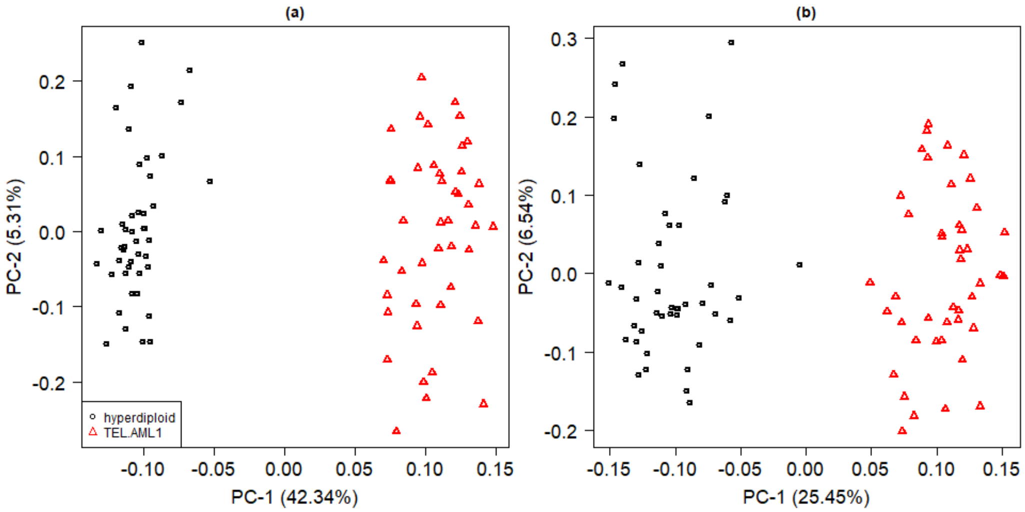

Figure 13: PCA plots based on the identified differentially expressed genes between hyperdiploid and TEL.AML1 subtypes of the CALL dataset using: (a) VSPCA and (b) MTT.

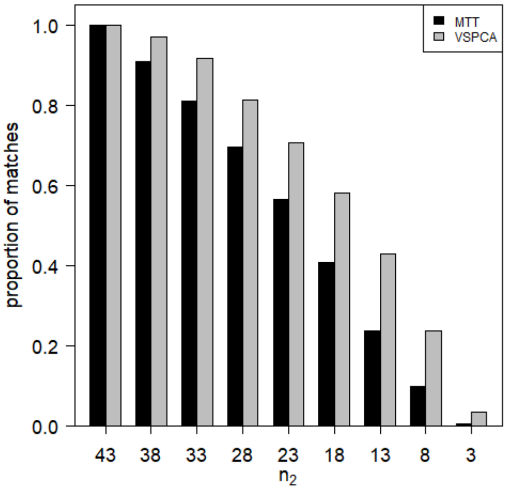

Figure 14: We used hyperdiploid samples ($n_1 = 44$) as a training set of data and TEL.AML1 samples ($n_2 = 43$) as a test set of data. The value of n_2 was reduced gradually from 43 to 3 by successively subtracting 5 (as shown on the x-axis from left to right). For all values of $n_2 < 43$ that were considered, we selected a sample without replacement from the test set of data and conducted variable selection based on VSPCA and MTT using the nominal level of significance $\alpha = 0.01$. The experiment was repeated 1000 times for each value of $n_2 < 43$. The variables that were significant under smaller sample sizes matched with the ones that were selected under the entire available sample size for the test data are counted. Shown are the proportions of significant variables under the entire sample size that were also significant under a smaller sample sizes in more than 50% (500/1000) of the replicates.

3.3 GWAS to identify SNPs associated with rice grain length

We used a publicly available High Density Rice Array (HDRA, 700k SNPs) [McCouch et al., 2016] to further evaluate our method on high-dimensional and non-normal data. The genotype and phenotype (average grain length) data for 1,568 accessions could be downloaded from the Rice Diversity database[1]. Each of the 1,568 accessions was genotyped. The genotypic dataset of 700,000 SNPs that was assayed amounted to one SNP approximately every 0.54 kb across the rice genome. For more details about the dataset readers are referred to McCouch et al. [2016]. Note that our aim here was not to re-analyse the data but to show how well our method performs on very high-dimensional non-normal data.

We limited the genotype data to include accessions for which phenotype data are available. Out of these 1,146 accessions, we were only able to use 1,127 accessions due to the missing IDs across the phenotypes and genotypes files. These included 47 admixed, 23 admixed-indica, 102 admixed-japonica, 30 aromatic, 158 aus, 332 indica, 185 temperate-japonica, and 250 tropical-japonica sub-populations as categorized by McCouch et al. [2016]. We removed SNP loci from our analysis if the call rate was less than 95% or if minor allele frequency (MAF) is less than 5% or deviated from Hardy–Weinberg equilibrium. A total of 158,956 markers were included in the analysis none of them have more than 5% missing values. All the missing values for a SNP were set to the mean of the values pertaining to varieties whose values were non-missing for that particular SNP. The resultant genotype data, after scaling for each SNP to have zero mean and unit standard deviation, were stored in an $n \times p$ matrix G. We performed pre-selection PCA. The PCA plots in McCouch et al. [2016] are reproduced in Figure 16 for completeness, which shows a strong population structure. The PC-1 captured the variation between indica and japonica varieties, the PC-2 highlighted the variation between aus and indica subpopulations, and the PC-3 highlighted the variation among the three japonica subpopulations.

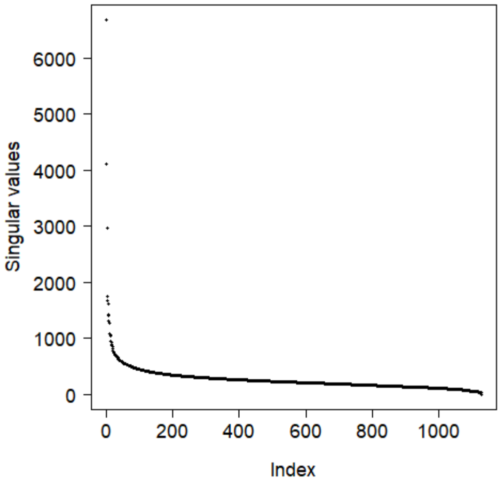

Figure 15: Scree plot based on 158,956 filtered markers of 1,127 accessions genotype data.

[1] http://www.ricediversity.org/

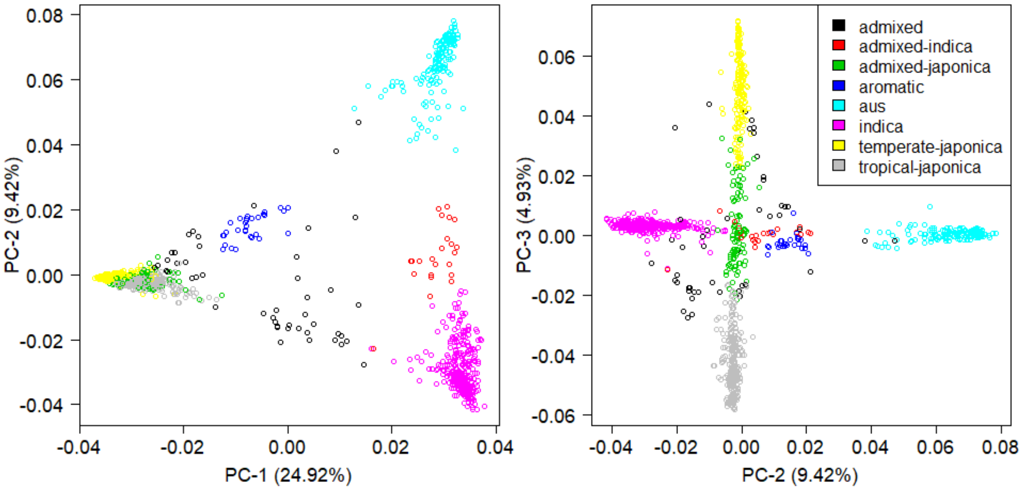

Figure 16: PCA plots based on the 158,956 filtered markers of 1,127 varieties of rice. The categorization into sub-populations is in accordance with McCouch et al. [2016].

The problem of confounding by population structure in genome-wide association studies (GWAS) is widely recognized [Devlin and Roeder, 1999, Price et al., 2006, Yu et al., 2006]. One need to correct for confounding factors induced by relatedness of observations to minimize FDR [Devlin and Roeder, 1999]. To correct for population stratification, we decompose G using singular value decomposition and reconstruct it using $q = 10$; that is,

$$\tilde{G} = G - G\tilde{V}^T\tilde{V}.$$

The phenotype of the 1,127 varieties had a bimodal distribution (see Figure 17) with easily distinguishable peaks and with a complete separation around the average grain length of 6 mm. Therefore, we partitioned the genotype data, contained in $Y = \tilde{G}$, into two classes based on the phenotype variable such that the varieties of rice having average grain length of less than 6 mm fell into class-A and those having average grain length of more than 6 mm fell into class-B. We used the class-A data ($n_1 = 692$) to obtain the estimates of the parameters. We used total variance explained criterion and set $\phi = .8$. This led us to use $q = 436$ to reconstruct the genotype data from class-B ($n_2 = 435$). The quantiles of $\sqrt{n_2}t_j^+/\hat{\sigma}$ deviated from the expected quantiles under null distribution $N(0, 1)$ (see Figure 19), which indicated that some SNPs are associated with the phenotype of interest. This is equally reflected in the histogram of unadjusted p-values, where the p-values are uniformly distributed with a spike near zero (see Figure 20(a)).

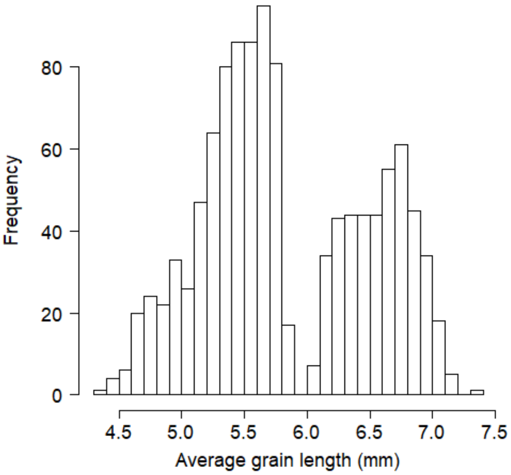

Figure 17: Histogram of average grain length for 1,127 varieties of rice.

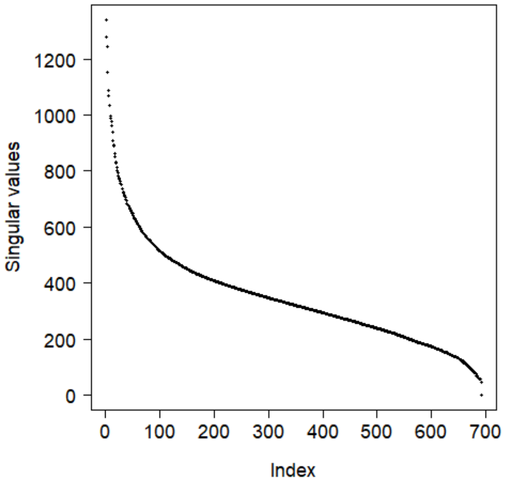

Figure 18: Scree plot based on 158,956 filtered markers of class-A genotype data.

Our GWAS identified one major locus on the rice chromosome-3 associated with the grain length and several minor loci (see Manhattan plot in Figure 20(b)). These findings were consistent with the previous findings. For example, the SNP-3.16732086, a functional SNP in the GS3 gene on the rice chromosome-3 [for example, see Fan et al., 2006, Mao et al., 2010], were found to be highly associated with grain length in McCouch et al. [2016] and is also highly significant (p-value $= 2.47e - 26$) in our analysis together with some more significant SNPs in the close vicinity on chromosome-3. The locus correspond to the peak on chromosome-5 was found to be associated with grain size [Li et al., 2011] and is also mentioned in McCouch et al. [2016] for its possible association with grain length. Other well pronounced peaks were located on chromosome-4 and chromosome-8, which were also observed in McCouch et al. [2016]. One reason for identifying loci on almost every chromosome could be that we have dichotomised the phenotype variable while it is used as a continuous variable in other studies [for example, see McCouch et al., 2016].

Dichotomisation of average length of grain might have confounded the association of other traits of grain with the grain length. Several quantitative trait loci (QTLs) for grain traits have been previously identified located all over the rice genome. [For an overview, see Huang et al., 2013, Zuo and Li, 2014].

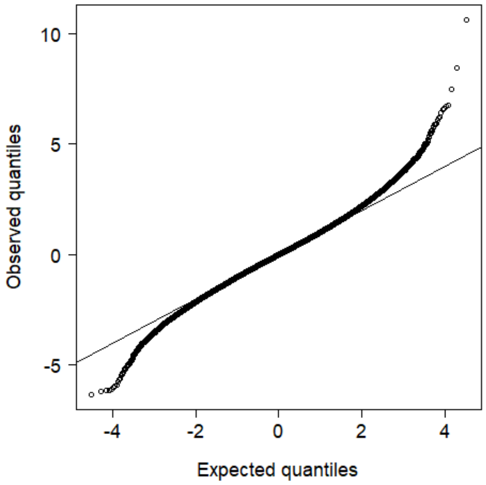

Figure 19: Quantiles of $\sqrt{n_2}t_j^+/\hat{\sigma}$ against the quantiles of standard normal distribution.

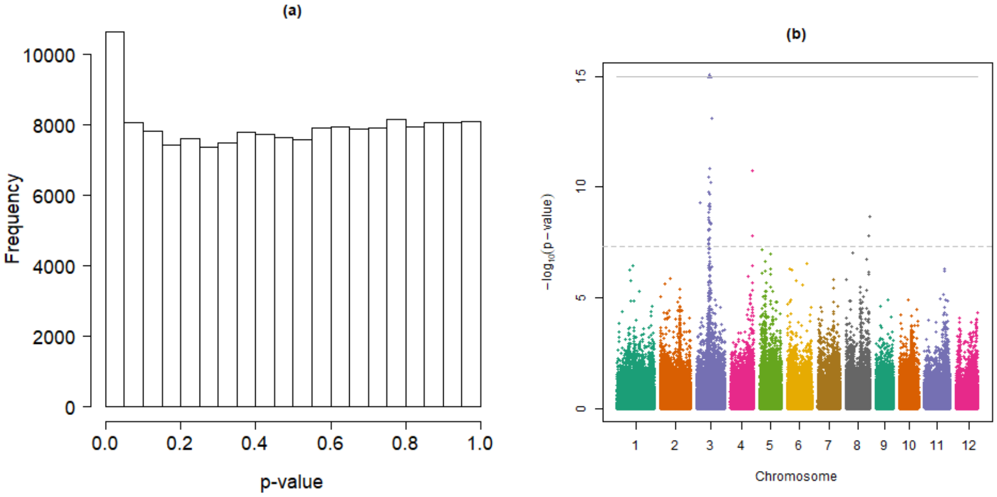

Figure 20: Plots of p-values: (a) Histogram of unadjusted p-values and (b) Manhattan plot based on unadjusted p-values obtained using VSPCA. The gray horizontal dashed-line indicates genome-wide significant threshold corresponding to p-value of $5e-08$ and the gray solid-line indicates the upper-limit of the plot along y-axis. The $-\log_{10}$(p-value) for some SNPs are beyond this upper-limit and are indicated by solid-triangles.

4 Conclusion and discussion

PCA is a classical statistical method traditionally used for dimension reduction. It is computationally manageable even for a very large dataset. A drawback of PCA, however, is that it tries to uncovers the underlying structure in the data in an unsupervised manner. We overcome this drawback of PCA by taking into account the target variable. We propose to use PCA in a supervised manner for variable selection in high-dimensional datasets when the objective is class prediction in a binary classification problem such as a case-control study.

A training dataset that comes from the control group (or a group with more samples) is used to first construct a PCA model. The test dataset obtained from the cases group (or a group with less samples) is reconstructed using the low-rank approximation based on the estimates of the parameters obtained during the training phase. The induced error obtained via a low-rank reconstruction of the test set of data is used to identify and rank the variables according to their relevance to the target of interest. All variables that have the ability to predict class labels accurately will have a larger average reconstruction error and are the most informative predictors. Often we are interested in p-values associated with each variable since they are readily interpretable. Assuming that the errors are normally distributed with mean zero and with an unknown variance σ^2 that we estimate using a robust estimator, we propose to obtain p-values associated with each variable. These p-values, once corrected for multiple testing, are then used for the selection of a superset of variables.

We tested our proposed method through the analysis of a well-known semi-real gene expression dataset with known differentially expressed genes and through the analysis of a real childhood acute lymphoblastic leukemia gene expression dataset. The results from a state-of-the-art independent filtering test (MTT) are also presented for comparison purposes. The proposed method identifies the superset of variables that have the ability to discriminate the classes. In contrast, the MTT is a filtering approach that evaluates feature relevance independently for each feature. Our analysis showed that the MTT is prone to choosing many redundant feature as is criticised in Yu and Liu [2004]. The method is further evaluated through the analysis of a genome-wide association sudy of a publically available High Density Rice Array. The results were found consistent with the previous findings where computationally intensive mixed models were used to identity the trait loci. We expect our method to perform better than the mixed models because it uses all variables simultaneously. However, to be more certain, intensive comparative evaluation of the two methods on other datasets would be of great interest.

PCA seeks to capture the important correlation patterns among a set of variables in the first few components. The rest of the components are believed to stand for the peculiarities and unusual observations in the data; therefore, it has been used as a core method to de-noise high-dimensional datasets. We expect our VSPCA to reduce the False Discovery Rate (FDR) through accounting for the correlation between variables and through de-noising data by dropping a number of minor components in the training data. Disregarding the minor components will reduce the individual specific influence on the effect size since some of them are believed to represent outliers in the data.

The VSPCA is simple and computationally efficient for a high-dimensional dataset. For computational time and memory requirements of the R implementation of SVD the readers are referred to Anderson et al. [1999]. We even reduce these requirements by performing SVD only on the training data. Modern variable selection approaches such as LASSO regression require tuning parameters and computationally cumbersome methods such as cross-validation are used to tune these parameters. In contrast, our approach needs only a few lines of R code, as given in the appendix, and unlike LASSO regression it does not rely on tuning parameters that need to be chosen using techniques such as cross-validation, which may not be justifiable for a 'large p small n' data. However, VSPCA requires one to choose the number of components, q (indeed one can use $q = r$, however, we recommend to drop some minor components). We recommend to choose a value of ϕ ranges from 0.7 to 0.9, which is a popular heuristic approach to choose q. This criterion usually favours larger values of q and is safer. In our experience, changing the value of ϕ from 0.7 to 0.9 did not make a noticeable difference to the final results. Other heuristics such as looking for an elbow point in a scree plot could be used as well. However, we found that the scree plot is less safe because it favours smaller values of q and an extra-small value of q might have a large effect on the results therefore, increase the dependency of the results on the value of q. We found it safe to use a slightly larger value of q (larger value than what the elbow point suggest) since it does not make a notable change to the results.

Acknowledgements

This research was supported by an ARC Australian Laureate Fellowship under Grant No. FL150100150. The authors also acknowledge the support of the Australian Research Council (ARC) Centre of Excellence for Mathematical and Statistical Frontiers (ACEMS).

References

E. Ahlqvist, P. Storm, A. Käräjämäki, M. Martinell, M. Dorkhan, A. Carlsson, P. Vikman, R. B. Prasad, D. M. Aly, P. Almgren, et al. Novel subgroups of adult-onset diabetes and their association with outcomes: a data-driven cluster analysis of six variables. *The Lancet Diabetes & Endocrinology*, 6(5):361–369, 2018.

O. Alter, P. O. Brown, and D. Botstein. Singular value decomposition for genome-wide expression data processing and modeling. *Proceedings of the National Academy of Sciences*, 97(18):10101–10106, 2000.

E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen. *LAPACK Users' guide*, volume 9. Siam, 1999.

Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(1):289–300, 1995.

L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

M. Caiazzo, M. T. Dell'Anno, E. Dvoretskova, D. Lazarevic, S. Taverna, D. Leo, T. D. Sotnikova, A. Menegon, P. Roncaglia, G. Colciago, et al. Direct generation of functional dopaminergic neurons from mouse and human fibroblasts. *Nature*, 476(7359):224, 2011.

Y. Choi, J. Taylor, R. Tibshirani, et al. Selecting the number of principal components: Estimation of the true rank of a noisy matrix. *The Annals of Statistics*, 45(6):2590–2617, 2017.

F. S. Collins and H. Varmus. A new initiative on precision medicine. *New England Journal of Medicine*, 372 (9):793–795, 2015.

D. Dembélé and P. Kastner. Fold change rank ordering statistics: a new method for detecting differentially expressed genes. *BMC Bioinformatics*, 15(1):14, 2014.

M. L. Den Boer et al. A subtype of childhood acute lymphoblastic leukaemia with poor treatment outcome: a genome-wide classification study. *The Lancet Oncology*, 10(2):125–134, 2009.

B. Devlin and K. Roeder. Genomic control for association studies. *Biometrics*, 55(4):997–1004, 1999.

R. Díaz-Uriarte and S. A. De Andres. Gene selection and classification of microarray data using random forest. *BMC bioinformatics*, 7(1):3, 2006.

C. Eckart and G. Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3): 211–218, 1936.

C. Fan, Y. Xing, H. Mao, T. Lu, B. Han, C. Xu, X. Li, and Q. Zhang. Gs3, a major qtl for grain length and weight and minor qtl for grain width and thickness in rice, encodes a putative transmembrane protein. *Theoretical and Applied Genetics*, 112(6):1164–1171, 2006.

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

J. Fan, Y. Feng, and R. Song. Nonparametric independence screening in sparse ultra-high-dimensional additive models. *Journal of the American Statistical Association*, 106(494):544–557, 2011.

L. Gautier, L. Cope, B. M. Bolstad, and R. A. Irizarry. affy—analysis of affymetrix genechip data at the probe level. *Bioinformatics*, 20(3):307–315, 2004.

T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, et al. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531–537, 1999.

F. R. Hampel. The influence curve and its role in robust estimation. *Journal of the American Statistical Association*, 69(346):383–393, 1974.

Q. He and D.-Y. Lin. A variable selection method for genome-wide association studies. *Bioinformatics*, 27 (1):1–8, 2010.

S. Hochreiter et al. A new summarization method for affymetrix probe level data. *Bioinformatics*, 22(8): 943–949, 2006.

H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of educational psychology*, 24(6):417, 1933.

R. Huang, L. Jiang, J. Zheng, T. Wang, H. Wang, Y. Huang, and Z. Hong. Genetic bases of rice grain shape: so many genes, so little known. *Trends in plant science*, 18(4):218–226, 2013.

I. Jolliffe. *Principal component analysis*. Springer Verlag, Springer Series in Statistics. New York, 3 edition, 2002.

I. T. Jolliffe. Discarding variables in a principal component analysis. i: Artificial data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 21(2):160–173, 1972.

I. T. Jolliffe and J. Cadima. Principal component analysis: a review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374(2065): 20150202, 2016.

Y. Li, C. Fan, Y. Xing, Y. Jiang, L. Luo, L. Sun, D. Shao, C. Xu, X. Li, J. Xiao, et al. Natural variation in gs5 plays an important role in regulating grain size and yield in rice. *Nature genetics*, 43(12):1266, 2011.

H. Mao, S. Sun, J. Yao, C. Wang, S. Yu, C. Xu, X. Li, and Q. Zhang. Linking differential domain functions of the gs3 protein to natural variation of grain size in rice. *Proceedings of the National Academy of Sciences*, 107(45):19579–19584, 2010.

G. P. McCabe. Principal variables. *Technometrics*, 26(2):137–144, 1984.

S. R. McCouch, M. H. Wright, C.-W. Tung, L. G. Maron, K. L. McNally, M. Fitzgerald, N. Singh, G. DeClerck, F. Agosto-Perez, P. Korniliev, et al. Open access resources for genome-wide association mapping in rice. *Nature communications*, 7:10532, 2016.

E. J. McTavish, J. E. Decker, R. D. Schnabel, J. F. Taylor, and D. M. Hillis. New world cattle show ancestry from multiple independent domestication events. *Proceedings of the National Academy of Sciences*, 110 (15):E1398–E1406, 2013.

K. Pearson. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.

L. Pi and S. Halabi. Combined performance of screening and variable selection methods in ultra-high dimensional data in predicting time-to-event outcomes. *Diagnostic and prognostic research*, 2(1):21, 2018.

A. L. Price, N. J. Patterson, R. M. Plenge, M. E. Weinblatt, N. A. Shadick, and D. Reich. Principal components analysis corrects for stratification in genome-wide association studies. *Nature Genetics*, 38 (8):904, 2006.

A. L. Price, N. A. Zaitlen, D. Reich, and N. Patterson. New approaches to population stratification in genome-wide association studies. *Nature Reviews Genetics*, 11(7):459, 2010.

M. Ringnér. What is principal component analysis? *Nature Biotechnology*, 26(3):303, 2008.

M. E. Ritchie et al. limma powers differential expression analyses for rna-sequencing and microarray studies. *Nucleic Acids Research*, 43(7):e47–e47, 2015.

C. P. Roca, S. I. Gomes, M. J. Amorim, and J. J. Scott-Fordsmand. Variation-preserving normalization unveils blind spots in gene expression profiling. *Scientific Reports*, 7:42460, 2017.

G. Saporta and N. Niang. Principal component analysis: application to statistical process control. *Data Analysis*, pages 1–23, 2009.

H. Shen and J. Z. Huang. Sparse principal component analysis via regularized low rank matrix approximation. *Journal of multivariate analysis*, 99(6):1015–1034, 2008.

G. K. Smyth. Linear models and empirical bayes methods for assessing differential expression in microarray experiments. *Statistical Applications in Genetics and Molecular Biology*, 3(1):1–25, 2004.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

I. Ullah, S. Paul, Z. Hong, and Y.-G. Wang. Significance tests for analyzing gene expression data with small sample sizes. *Bioinformatics*, 2019.

M. J. Wathen, Y. Gautam, S. Ghandikota, M. B. Rao, and T. B. Mersha. Lei: A novel allele frequency-based feature selection method for multi-ancestry admixed populations. *Scientific reports*, 9(1):11103, 2019.

J. Yu, G. Pressoir, W. H. Briggs, I. V. Bi, M. Yamasaki, J. F. Doebley, M. D. McMullen, B. S. Gaut, D. M. Nielsen, J. B. Holland, et al. A unified mixed-model method for association mapping that accounts for multiple levels of relatedness. *Nature genetics*, 38(2):203, 2006.

L. Yu and H. Liu. Efficient feature selection via analysis of relevance and redundancy. *Journal of Machine Learning Research*, 5(Oct):1205–1224, 2004.

Q. Zhu, J. C. Miecznikowski, and M. S. Halfon. Preferred analysis methods for affymetrix genechips. ii. an expanded, balanced, wholly-defined spike-in dataset. *BMC Bioinformatics*, 11(1):285, 2010.

H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320, 2005.

H. Zou, T. Hastie, and R. Tibshirani. Sparse principal component analysis. *Journal of Computational and Graphical Statistics*, 15(2):265–286, 2006.

J. Zuo and J. Li. Molecular genetic dissection of quantitative trait loci regulating rice grain size. *Annual review of genetics*, 48:99–118, 2014.

A R code to perform simulations

```r
# Required packages for generating data and performing MTT
library(MASS)
library(limma)

# number of observation in control group or the group with largest sample size
n1 <- 40
# number of variables
p <- 500

# I use exchangeable covariance structure for both groups
# I add a bit of noise to the daigonal different for the two groups
Sigma=matrix(0.9,p,p);diag(Sigma)=1;
Sigmax =Sigma + diag(c(rgamma(p,1,1)))
Sigmay=Sigma + diag(c(rgamma(p,1,1)))

# level of significance
alpha <- 0.01
# different levels of shift
c0=0 # No shft. Null is true
c1=0.8 # smallest shift
c2=1.5 # middle level shift
c3=2.0 # largest shift

# We repeat 1000 independent experiment of similar nature
res <- replicate(1000,{

  # generate n1 X p matrix of data. This is the training data
  Y_minus <- mvrnorm(n=n1, mu= rep(0,p), Sigma=Sigmax)

  # number of observations in the cases group
  n2 <- 10
  # below
  # shifts sizes are randomly chosen for 10% of variables
  mu <- rbinom(p,1,prob=.10)*sample(c(-c3,-c2,-c1,c1,c2,c3), p, replace=T)
  # generate n2 X p matrix of data. This is the cases data.
  Y_plus <- mvrnorm(n=n2, mu=mu, Sigma=Sigmay)

  # standardize columns of training data
  X_minus <- scale(Y_minus)
  # standardize columns of cases data using the mean
  # and standard deviations of the columns of training data.
  # R is clever! It stores means and sds for you in X_minus
  X_plus <- scale(Y_plus, center = attr(X_minus,"scaled:center"),
                  scale=attr(X_minus,"scaled:scale"))

  # perform SVD using training data
  SVDec <- svd(X_minus)
  V <- SVDec$v
  # scree plot to decide how many components to use to recover cases data.
  # plot(SVDec$d)

  # number of components to use to recover cases data.
  # We specify phi=.80
  nc=1
  while (cumsum(sum(SVDec$d[1:nc])/sum(SVDec$d))<.80){
    nc=nc+1
  }
  #nc

  X_plus_tilde <- X_plus%*%V[,1:nc]%*%t(V[,1:nc])
  # induced error
  err=(X_plus-X_plus_tilde)
  # average error
  tjplus <- apply(err,2,mean) # assumed to follow a normal distribution
  pvalues <- p.adjust(2*pnorm(abs(tjplus*sqrt(n2)/mad(tjplus*sqrt(n2))),
                              lower.tail = F), method = "fdr")

  # perform MTT
  YY <- t(rbind(Y_minus, Y_plus))
  design <- cbind(Grp1=1,Grp2vs1=rep(c(0,1),c(n1,n2)))
  fit <- lmFit(YY,design)
  efit <- eBayes(fit)
  de.table <- topTable( efit, coef=2, number=Inf, sort.by="none",
                        adjust.method="fdr" )

  # finaly results
  c(mean(pvalues[abs(mu)==c0] < alpha), # null with VSPCA
    mean(de.table$adj.P.Val[abs(mu)==c0] < alpha), # null with MTT
    mean(pvalues[abs(mu)==c1] < alpha), # identify shift c1 with VSPCA
    mean(de.table$adj.P.Val[abs(mu)==c1] < alpha), # identify shift c1 with MTT
    mean(pvalues[abs(mu)==c2] < alpha), # identify shift c2 with VSPCA
    mean(de.table$adj.P.Val[abs(mu)==c2] < alpha), # identify shift c2 with MTT
    mean(pvalues[abs(mu)==c3] < alpha), # identify shift c3 with VSPCA
    mean(de.table$adj.P.Val[abs(mu)==c3] < alpha)) # identify shift c3 with MTT
})

# compare results for different levels of shifts
par(mar=c(6, 6, 4, 2))
bcol=rep(c("darkgoldenrod","deepskyblue"), 4)
boxplot(res~row(res), xlab="", ylab="", main="", cex.axis=1.5, las=1, ylim=c(0,1),
        xaxt='n', col=bcol)
axis(1, at=c(1.5,3.5,5.5,7.5),labels=c(c0,c1,c2,c3),
     line=0,tck=-.05, cex.axis=1)
legend("topleft",
       legend=c("VSPCA", "MTT"), horiz = F ,
       cex = 1, bg = "white",box.lty = 1,
       fill = unique(bcol))
title(xlab="shift size", cex.lab=1.5, line = 3.5)
title(ylab="power", cex.lab=1.5, line=3.5)
```