IMPERIAL

IMPERIAL COLLEGE LONDON

DEPARTMENT OF MATHEMATICS

SECOND-YEAR GROUP RESEARCH PROJECT

# Gene Expression Data Analysis

*Author:*
Mingfei Yi (CID: 02236292)
Kevin Pan (CID: 02218181)
Cheng Fang (CID: 02218106)
Annabel Paek (CID: 01862534)
Yangduoji Li (CID: 02207261)

*Supervisor(s):*
Marina Evangelou

November 17, 2024

**Abstract**

This research project focuses on gene expression data analysis using microarrays to understand the variations in gene expression levels among individuals with colitis and cancer. The project explores different methods including multivariate logistic regression with shrinkage methods, univariate logistic regression, and principal component regression, to build and validate predictive models for gene expression analysis. Further, complexity parameter selection, predictive model selection, and gene selection are proposed and explored.

# Contents


**1 Introduction** **3**

**2 Methodology** **4**

2.1 Multivariate Logistic Regression . . . . . 4

2.2 Shrinkage Methods . . . . . 5

2.2.1 Ridge Regression . . . . . 5

2.2.2 LASSO Regression . . . . . 6

2.2.3 Comparison between Ridge and LASSO . . . . . 7

2.2.4 Elastic Net . . . . . 7

2.3 Univariate Logistic Regression . . . . . 8

2.3.1 Multiple Testing . . . . . 9

2.4 Principal Component Analysis and Regression . . . . . 10

2.4.1 Principal Component Analysis . . . . . 10

2.4.2 Principal Component Regression . . . . . 11

2.5 Testing . . . . . 11

2.5.1 Cross-Validation . . . . . 11

2.5.2 Assessing Model . . . . . 12

**3 R package** **12**

3.1 GEOquery . . . . . 12

3.2 glmnet . . . . . 13

3.3 factoextra . . . . . 13

3.4 pls . . . . . 13

**4 Analysis: Colitis Dataset** **13**

4.1 Model Selection . . . . . 13

4.1.1 LASSO and Ridge Model . . . . . 14

4.1.2 PCR . . . . . 14

4.2 Variable Selection . . . . . 15

4.2.1 Selection . . . . . 15

4.2.2 Validation . . . . . 17

4.3 Prediction using PCR on selected variables . . . . . 19

**5 Analysis: Breast Cancer Dataset** **21**

5.1 Data Pre-processing . . . . . 21

5.1.1 Imputation Methods . . . . . 21

5.1.2 Cut-off Percentage Decision . . . . . 22

5.2 Model Selection . . . . . 23
5.2.1 LASSO and Ridge Model . . . . . 23
5.2.2 PCR . . . . . 24
5.3 Variable Selection . . . . . 25
5.3.1 Selection . . . . . 25
5.3.2 Validation . . . . . 25
5.4 Prediction using PCR on selected variables . . . . . 26

**6 Discussion** **27**
6.1 Criticism . . . . . 27
6.2 Future . . . . . 27

**Acknowledgement** **27**

**Appendix** **28**

# 1 Introduction

The study of gene expression is a crucial field within bioinformatics and computational biology, offering valuable insights into the molecular mechanisms that drive various biological processes and diseases. Gene expression is the process by which information encoded in a gene is turned into a function. This process involves transcription (copying DNA to RNA) and translation (making proteins from RNA). By quantifying gene expression levels, researchers can decipher the complex interactions among genes and their roles in the development and progression of diseases. Many investigations have demonstrated significant correlations between gene expression patterns and the emergence and progression of diseases like colitis and cancer. Understanding these connections is vital for identifying biomarkers for disease diagnosis and potential therapeutic targets.

Microarray technology, a key tool in gene expression studies, enables the simultaneous measurement of thousands of gene expression levels. This uses a grid of DNA sequences attached to a solid surface to allow parallel analysis of gene expression across different samples. This high-throughput method provides a broad overview of gene activity, making it invaluable for detecting genes that show differential expression under various conditions or treatments. [1]

For example, in the recent news scientists explored how altered gene expression can induce autism. By developing genetically engineered mice with specific gene mutations, they performed comprehensive profiling. This research revealed that changes in gene expression could lead to autistic-like behaviors, offering potential targets for therapeutic intervention. [2]

In our thesis, we analyse gene expression data from microarray experiments, focusing on differences between individuals with colitis and those with breast cancer. Colitis is a major form of inflammatory bowel disease (IBD) that causes chronic inflammation of the colon. It affects millions of people worldwide, leading to significant morbidity. We also look into patients with breast cancer which is one of the most common cancers globally, affecting millions of women worldwide. The colitis dataset comprises expression profiles from 127 individuals, including 85 colitis patients and 42 controls. Gene expression analysis is conducted using micro-arrays to measure the expression of 22,283 genes across these individuals. The cancer dataset comprises data from 60 breast cancer patients treated with tamoxifen, categorised by cancer recurrence status. The original dataset included more than 22,575 genes but exhibited a significant amount of missing data. [3]

Addressing challenges of high-dimensional data, such as missing values and potential overfitting, this study employs a robust statistical framework to pre-process the data and construct predictive models.

Our introduction outlines the significance of gene expression analysis and the study's objectives. The second chapter is about methodology which tackles two methods- multivariate and univariate logistic regression- describing the statistical techniques, highlighting the mathematical models and algorithms used for analysis. The results section presents a comprehensive analysis of the gene expression models, including performance metrics and comparative assessments of the imputation methods. The discussion interprets the findings, examines their implications for understanding disease mechanisms, and suggests potential future research directions.

By refining the methods for pre-processing and analysing the data, this work aims to enhance our understanding of the molecular foundations of colitis and cancer and improve the robustness and accuracy of genomics data analysis techniques. This advancement could lead to more reliable identification of biomarkers (measurable indicators of disease) and therapeutic targets.

# 2 Methodology

## 2.1 Multivariate Logistic Regression

Since the datasets contain multiple continuous explanatory variables and one binary response variable, we could use multivariate logistic regression to set up a model.

In this report, we use binary logistic regression as the response is separated into 2 classes: control(y = 1) or case( y = 0).

First, it involves weights which represent the importance of each predictor and an intercept. Let $z$ be the linear combinations of the predictors:

$$z = \beta_0 + \sum_{i=1}^{n} \beta_i x_i = \mathbf{X}\beta \tag{1}$$

where $x$ represents one predictor variable, $\mathbf{X}$ is the design matrix for the predictors and $\beta$ is the column vector of the weights.

Next, $z$ which can take any real value is transformed into a probability that ranges from $0$ to $1$ through a **sigmoid function**, also known as the **logistic function** (Figure 1) and leads to the eponymous regression:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{2}$$

In this way, it is particularly useful for probability estimation and binary classification tasks.

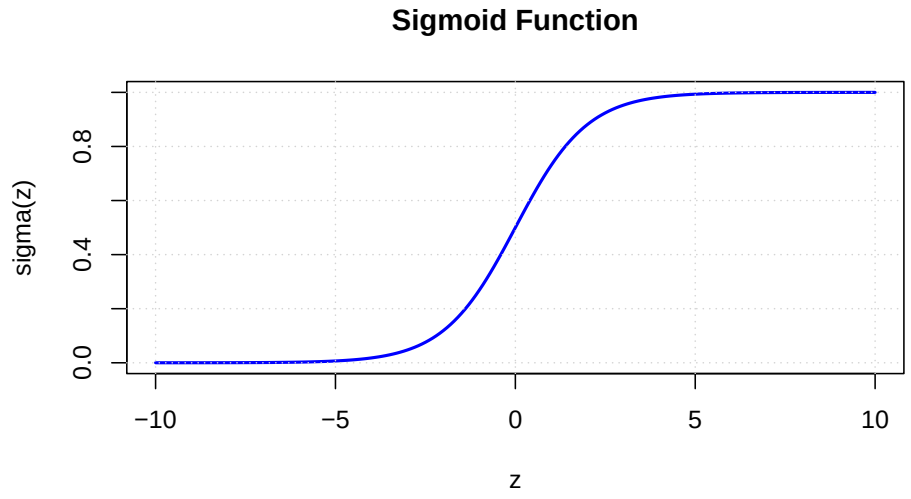


**Figure 1** It is called sigmoid because it has the shape of the letter s.

Since the response is either positive (1) or negative (0), the probabilities can be assigned as follows:

$$P(y = 1|\mathbf{x}) = \sigma(\beta_0 + \sum_{i=1}^{n} \beta_i x_i) = \frac{1}{1 + exp(\beta_0 + \sum_{i=1}^{n} \beta_i x_i)} \tag{3a}$$

$$P(y = 0|\mathbf{x}) = 1 - P(y = 1|\mathbf{x}) = \frac{exp(\beta_0 + \sum_{i=1}^{n} \beta_i x_i)}{1 + exp(\beta_0 + \sum_{i=1}^{n} \beta_i x_i)} \tag{3b}$$

Note that $1 - \sigma(z) = \sigma(-z)$ and $P(y = 0) = \sigma(-(\beta_0 + \sum_{i=1}^{n} \beta_i x_i))$.

The variable $z$ is frequently referred to as logit since the logit function, which gives the natural logarithm of the odds (log-odds), is the inverse of the sigmoid function:

$$logit(p) = \log \frac{p}{1-p} = \sigma^{-1}(p) = \mathbf{X}\beta \tag{4}$$

where $p = P(y = 1|\mathbf{x})$.

Usually, the decision boundary is set at 0.5, that is: the sample is predicted to be positive (1) if the predicted probability is greater than 0.5 and vice versa. [4]

The intercept gives the base log-odds when all the predictors are $0$. Without it, the model would be forced through the origin and that could make the model unreliable. The coefficient of the model can be interpreted as the change in log-odds for every unit increase in a predictor. For example, let $z = \beta_0 + 2x$, it means the log-odds goes up by $e^2 \approx 7.389$ times when $x$ is increased by $1$.

A logistic regression model is fitted using the maximum log-likelihood so that the predicted probability is as close to the observed status (0 or 1) as possible.

The likelihood function is:

$$\ell(\beta) = \prod_{i:y_i=1} p(y = 1|x_i, \beta) \prod_{i':y_{i'}=0} p(y = 0|x_{i'}, \beta) \tag{5a}$$

$$= \prod_{i=1}^{n} (y_i p(x_i) + (1 - y_i)(1 - p(x_i))) \tag{5b}$$

where $p(y = 1|x_i, \beta)$ is simplified to $p(x_i)$. The log-likelihood is given as:

$$log(\ell(\beta)) = \sum_{i=1}^{n} (y_i log(p(x_i)) + (1 - y_i) log(1 - p(x_i))) \tag{6}$$

[5]

Functions such as log loss ($-1\times$log-likelihood) or binomial deviance ($-2\times$log-likelihood) can be used to measure the performance of the fitted model.

A good logistic model will allow us to make an accurate prediction based on the values of the predictors.

## 2.2 Shrinkage Methods

Linear regression estimates tend to have low bias and high variance. Sometimes, we only want some of the parameters to set up a linear regression model. Therefore, we use shrinkage methods to improve the estimation of regression coefficients by introducing a penalty (or constraint) on the size of the coefficients. Reducing the complexity of the model (the number of parameters that need to be estimated) results in less variance at the cost of introducing more bias. If we can find the most desirable spot for the total error at which the error due to bias plus the error of the variance is minimised, we can improve the prediction of the model. Common shrinkage methods include *Ridge Regression* (L2 regularization), *LASSO Regression* (L1 regularisation).

### 2.2.1 Ridge Regression

Ridge regression shrinks all the regression coefficients by imposing a penalty equal to the square of the magnitude of the coefficients. The ridge coefficients minimise a penalised residual sum

of squares,

$$\hat{\beta}^{\text{ridge}} = \underset{\beta}{\text{argmin}} \left\{ \sum_{i=1}^{N} \left( y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}. \tag{7}$$

or

$$\hat{\beta}^{\text{ridge}} = \underset{\beta}{\text{argmin}} \sum_{i=1}^{N} \left( y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2, \tag{8}$$

$$\text{subject to} \sum_{j=1}^{p} \beta_j^2 \leq t$$

Here, $\lambda \geq 0$ is a complexity parameter which determines the amount of shrinkage: the larger the value of $\lambda$, the greater the amount of shrinkage. With special notice, the coefficients in ridge regression are shrunk toward zero (and each other), but never exactly zero.

When there exist many correlated variables in one linear regression model, their coefficients cannot be well determined as one large positive coefficient of one variable can be eliminated by another similarly large negative coefficient of its correlated variable, exhibiting high variance. By imposing a size constraint on them, this problem is solved.

The ridge solutions are not equivariant under scaling of the inputs, so inputs are normally standardised before solving (7). Noticing that the intercept $\beta_0$ is not included in penalty term, we separate the solutions to (7) into two parts after reparametrisation using *centered* inputs: every $x_{ij}$ gets replaced by $x_{ij} - \bar{x}_j$. We estimate $\beta_0$ by $\bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i$. The remaining coefficients are estimated by a ridge regression without intercept, using the centered $x_{ij}$.

Thus, we write equation (2.1) in matrix form

$$\text{RSS}(\lambda) = (\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta) + \lambda\beta^T\beta, \tag{9}$$

Then, we get ridge solutions as following

$$\hat{\beta}^{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y} \tag{10}$$

[5]

### 2.2.2 LASSO Regression

The LASSO (Least Absolute Shrinkage and Selection Operator) regression is a shrinkage method like ridge, with subtle but important differences. It imposes a penalty equal to the absolute value of the magnitude of the coefficients. The LASSO estimate is defined by

$$\hat{\beta}^{\text{LASSO}} = \underset{\beta}{\text{argmin}} \left\{ \sum_{i=1}^{N} \left( y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta| \right\}. \tag{11}$$

or

$$\hat{\beta}^{\text{ridge}} = \underset{\beta}{\text{argmin}} \sum_{i=1}^{N} \left( y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2, \tag{12}$$

$$\text{subject to} \sum_{j=1}^{p} |\beta| \leq t$$

Just the same as in ridge regression, we need reparametrise the constant $\beta_0$ by standardising the explanatory variables; the solution for $\hat{\beta}_0$ is $\bar{y}$, and then we fit a model without an intercept. With special notice, the penalty term makes the solutions nonlinear in the $y_i$, and there is no closed-form expression as in ridge regression. [5]

### 2.2.3 Comparison between Ridge and LASSO

We discuss and compare these two approaches discussed so far that restrict the linear regression model: ridge regression and LASSO regression. In the case of an orthonormal input matrix $\mathbf{X}$, each method sets a penalty term by transforming the least squares estimate $\hat{\beta}_j$. Ridge regression does a proportional shrinkage. LASSO translates each coefficient by a constant factor $\lambda$, truncating at zero.

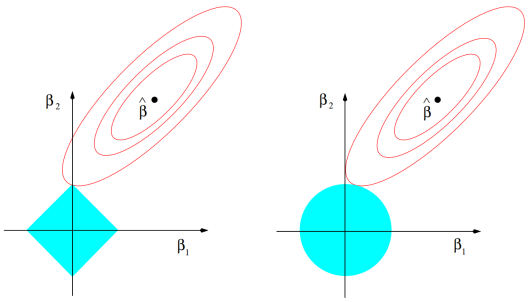


**Figure 2** Estimation pictures for LASSO(left) and Ridge(right). [5, p.71]
The residual sum of squares has elliptical contours, centered at the full least squares estimate. The constraint region for ridge regression is the disk $\beta_1^2 + \beta_2^2 \leq t$, while that for LASSO is the diamond $|\beta_1| + |\beta_2| \leq t$.

For the non-orthogonal case, we get to understand the relations between these two methods through Figure 2 depicting the LASSO (left) and ridge regression (right) when there are only two parameters. Both methods obtain the first point where the elliptical contours hit the constraint region. Unlike the disk, the diamond has corners; if the solution occurs at a corner on the axis, it has one parameter $\beta_j$ equal to zero. When p $>$ 2, the diamond becomes a rhomboid, and has many corners, flat edges and faces; there are many more opportunities for the estimated parameters to be zero. For example, based on our colitis dataset, the coefficients for ridge(Figure 3) and LASSO(Figure 4) are controlled by changing $\lambda$. Obviously, the larger the $\lambda$, the more coefficients are shrinking to exactly zero in LASSO regression while none of them would become zero in ridge regression eventually.

[5]

### 2.2.4 Elastic Net

Usually, many genes are involved in one biological pathway (series of molecular interactions), meaning that there are often high correlations between them.

LASSO is indifferent to the highly correlated predictors while ridge tends to shrink the coefficients of the correlated variables towards each other. The elastic net penalty can combine the benefits of the two methods: performing feature (same as predictor) selection while controlling

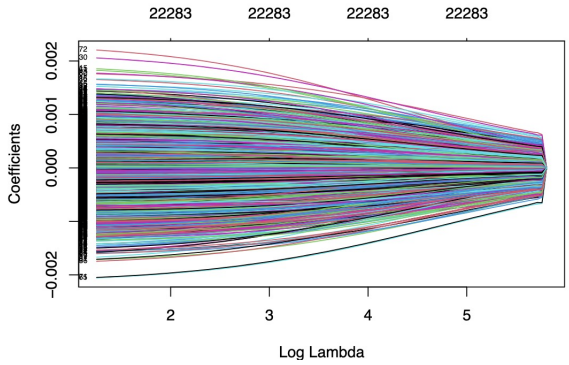


**Figure 3** Coefficients-Lambda Plot for Ridge based on Colitis Dataset. The larger the $\lambda$, the more coefficients are shrinking to near zero, but never exactly zero.

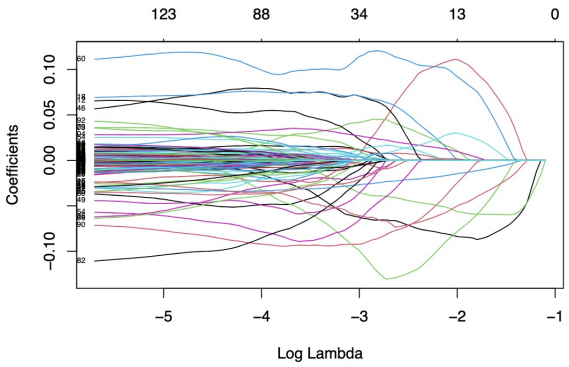


**Figure 4** Coefficients-Lambda Plot for LASSO based on Colitis Dataset. The larger the $\lambda$, the more coefficients are shrinking to exactly zero.

the magnitude of the coefficients. The penalty has the form:

$$\sum_{i=1}^{n}(\alpha|\beta_i| + (1-\alpha)\beta_i^2) \tag{13}$$

In the case where $p(no.of predictors) > N(no.of samples)$, the elastic net penalty can give more than $N$ predictors with non-zero coefficients and that could be more useful than LASSO, which can only select up to $p$ predictors. We can see the difference between LASSO and elastic net from Figure 5.[5, 6]

## 2.3 Univariate Logistic Regression

Besides multivariate logistic regression, we can use another method to select significant explanatory variables and compare the results.

The genes can be tested individually under a univariate logistic regression: $\log \frac{p}{1-p} = \beta_0 + \beta_1 x$ with $H_0 : \beta_1 = 0$, $H_1 : \beta_1 \neq 0$, where $p = P(y = 1|\mathbf{x})$. This can be carried out with Wald test on a single parameter.

In this scenario, we are given a log-likelihood, an unrestricted maximum likelihood estimator $\hat{\beta}_1$ of a parameter $\beta_1$, a testing value $0$ and the information matrix $I$, the Wald statistic is defined as:

$$W = \frac{(\hat{\beta}_1 - 0)^2}{I(\hat{\beta}_1)^{-1}} \sim \chi_1^2 \tag{14}$$

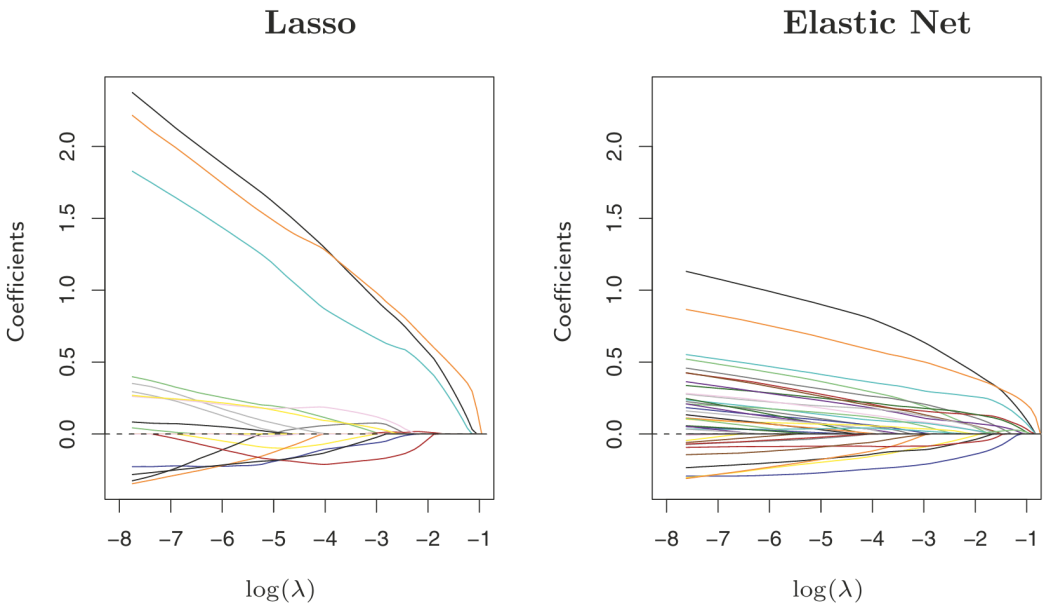


**Figure 5** Regularised logistic regression paths for the leukemia data. [5, p.662] The panels show how the coefficients vary with different $\lambda$ value. Using the smallest $\lambda$ value results in 19 and 39 non-zero coefficients for LASSO and elastic net respectively. Note that the magnitude of the coefficients for elastic net is smaller due to the averaging property

or equivalently (z-test),

$$\sqrt{W} = \frac{\hat{\beta}_1}{SE(\hat{\beta}_1)} \sim \mathbb{Z} = \mathbb{N}(0,1) \tag{15}$$

The $z$-value can be computed and the corresponding $p$-value can be found. [7]

For example,

| | Estimate | SE | $z$-value | Pr$(>\|z\|)$ |
|---|---|---|---|---|
| Intercept | 2.3042 | 3.9882 | 0.578 | 0.563 |
| MIR4640 | -0.4802 | 0.6364 | -0.755 | 0.451 |

**Table 1** GDS1615: MIR4640 (a gene), computed in R using logistic regression

Since Pr$(>|z|)$, i.e. the $p$-value, is greater the 0.05 (the usual level of significance $\alpha$), there is insufficient evidence to reject $H_0$, and therefore the first gene is unlikely to be significant.

### 2.3.1 Multiple Testing

The more hypothesis tests we run without correction, the more false positives (type I error) we could get by chance. One could use Bonferroni correction to control the family-wise error rate (FWER), i.e. the probability of type I error, but it is very conservative, meaning it can miss out on features that are significant (the more tests we run simultaneously, the lower the size and power of the individual test). Alternatively, one can use the Benjamini-Hochberg procedure to control the false discovery rate (FDR) and improve the power of the test, at the cost of admitting some false features.

T/F: true/false, D/N: discovery/no discovery

| | $H_0$ not rejected | $H_0$ rejected |
|---|---|---|
| $H_0$ true | TN | FD (type I error) |
| $H_0$ false | FN (type II error) | TD |
| total | N | D |

**Table 2** Hypothesis test result classification

In FWER control methods, the probability of having at least one false discovery is less than or equal to $\alpha$, whereas FDR $= \mathbb{E}(\frac{\text{FD}}{\text{D}}) < \alpha$, i.e. for a 5% FDR, the expected number of false discoveries is 5% of the total number of discoveries. Please refer to the reference for more details on Benjamini-Hochberg procedure. [8]

For the GDS1615 (colitis) dataset, Bonferroni and FDR pick up 554 and 6429 significant genes respectively.

## 2.4 Principal Component Analysis and Regression

### 2.4.1 Principal Component Analysis

Following the data selection process, we would like to validate our results. Principal Component Analysis (PCA) is a useful technique for visualising and validating variable selections.

PCA reduces dimensionality while minimising information loss, by transforming the original variables into a new set of uncorrelated variables that maximises the variance, called principal components (PCs), which are linear combinations of all $p$ original features. [9]

Suppose $X$ is the centred $n \times p$ matrix of the data set, then the principal components decomposition of $X$ can therefore be given as

$$X = SV^T \tag{16}$$

where $V$ is the $p \times p$ loading matrix of PCs, whose each column is the contributions of variables to the corresponding PC; and $S$ is the $n \times p$ score matrix of samples, whose each column is the position of each sample point in the new coordinate system consisting of PCs.

Loading and score matrices can be derived in several different ways. Below are the methods used by two common PCA functions in R. [10]

The R function `prcomp` uses SVD (singular value decomposition). It derives $V$ by an SVD on $X$, $X = UDV^T$. Then $S = XV = UD$.

The R function `princomp` uses eigendecomposition. It derives $V$ by an eigendecomposition on $X$ if $n = p$, $X = VDV^T$, or on the covariate matrix $\Sigma$ (up to a factor of $N$) if $n \neq p$, $n\Sigma = X^TX = VD^2V^T$. Then $S = XV$.

The columns of $V$ and $S$ are ranked from high to low by the variances explained, given by $Var(s_i) = \frac{d_i^2}{n}$, where $s_i$ is the variable consisting of the $i$th column of $S$, and $d_i$ is the $i$th diagonal element of $D$. [5]

Below are some of the most frequently used plots for PCA, their applications, and some guidelines for analysis.

The score plot depicts the transformed samples on selected PCs, it is the most important plot for visualising PCA results. Although there is no metric to quantify the significance of cluster separation, a complete cluster separation typically indicates high statistical significance,

suggesting that the variables effectively differentiate the samples based on the response. In contrast, partial cluster separation does not consistently indicate significance. [11]

The scree plot graphs the PCs against the variances they explain, it can help determine the number of PCs used in the score plot. A dataset suitable for score plot visualisation would have the first few PCs explaining more than 80% of the variance, or at least containing an elbow point. Note that cluster separations do not always appear in the first few PCs, and if they do, it supports one of the assumptions for PCR: the PCs that explain the largest variation in the predictor data should also be the most informative for predicting the response. [12]

A contribution plot displays the variables against their contributions to the selected PCs. Ideally, this plot helps identify which variables capture more variance with less noise. However, its performance requires accurate choices of PCs and the previously alluded assumption to be satisfied.

### 2.4.2 Principal Component Regression

PCR performs a regression on the principal components derived from PCA. Although PCR does not perform feature selection since the PCs are linear combinations of all $p$ original features, it can handle dimensionality reduction, because we can choose the first $q$ out of $p$ PCs to use for regression. Note that if we choose all the PCs, we would just get back the usual least squares estimates. [5]

Suppose $S_q$ is the $n \times q$ score matrix of samples on the first $q$ PCs, then the regression $\hat{y}$ is given by [13]

$$\hat{y} = S_q \beta \tag{17}$$

The solution is given by

$$\hat{\beta} = (S^T S)^{-1} S^T y \tag{18}$$

There are many different ways to determine $p$, one common approach is to choose the model that minimises RMSEP (root mean square error of prediction),

$$RMSEP = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y_i})^2} \tag{19}$$

where $y_i$ is the observed value, $\hat{y_i}$ is the predicted value and $n$ is the number of samples. [**?** ]

## 2.5 Testing

### 2.5.1 Cross-Validation

Cross-validation is a statistical method that provides a more unbiased assessment of a predictive model. By running the modeling process on different subsets of the dataset, cross-validation generates multiple measures of model quality.

For example, we can divide the dataset into 5 parts, each comprising 20% of the entire dataset. We say that we have broken the data into 5 "folds".

Then, we run one experiment for each fold:

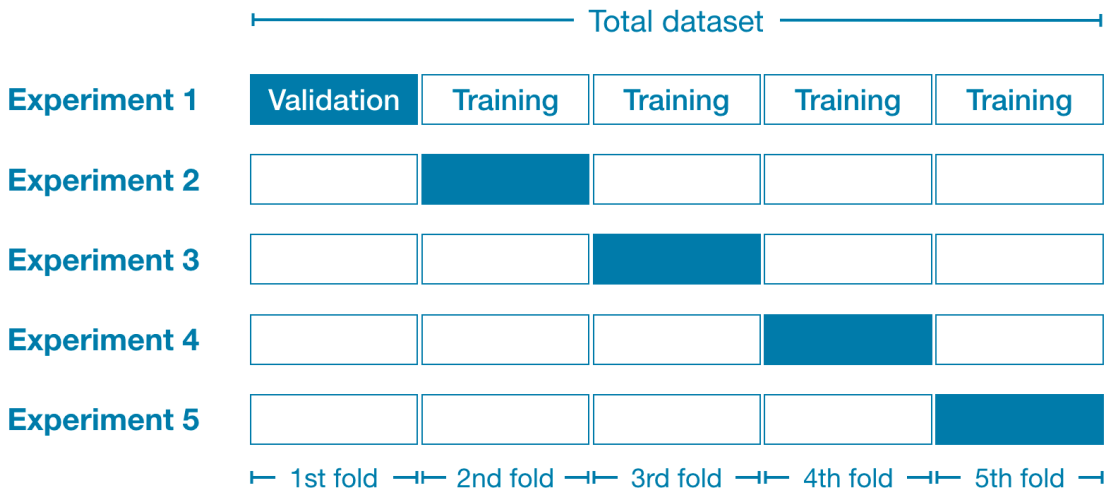


**Figure 6** Process of Cross-validation [14]

- In Experiment 1, we use the first fold as a validation set and everything else as training data. This gives us a measure of model quality based on a 20% validation set.
- In Experiment 2, we hold out data from the second fold (and use everything except the second fold for training the model). The validation set is then used to get a second estimate of model quality.
- We repeat this process until all the data is used as validation at some point, and we end up with a measure of model quality that is based on all of the rows in the dataset. [14]

### 2.5.2 Assessing Model

To assess our regression model, We randomly divided the dataset into two parts: 60% for training and 40% for testing, ensuring approximately the same proportion of the control group and the case group. We train the logistic regression model on the training dataset, and then we compare the result predicted by the fitted model on the testing dataset and the true data. We then use the misclassification rate (i.e. the number of incorrect predictions / the total number of predictions) as the metric to assess the accuracy of a model. To mitigate the effect of randomness, we repeat the experiment 100 times and use a box plot to visualise the distribution of accuracy for each model.

# 3 R package

## 3.1 GEOquery

Package `GEOquery` is primarily used for loading the gene database. Here is some of the R commands we used:

| | |
|---|---|
| `getGEO` | Download and parse a GEO SOFT format file into an R data structure |
| `GDS2eSet` | Convert the dataset to an ExpressionSet |
| `pData` | Convert the ExpressionSet to a data frame |

[15]

### 3.2 glmnet

Package `glmnet` is primarily used for fitting generalised linear and similar model using maximum likelihood with penalty. Here is some of the R commands we used:

| | |
|---|---|
| `cv.glmnet` | Fit the model using cross-validation, two models with different complexity parameters with return by default. *lambda.min* is the complexity parameter at which the smallest measure (such as MSE (mean square error), misclassification rate) is achieved, *lambda.1se* is the largest complexity parameter at which the measure is within one standard error of the smallest measure (default). |
| `predict` | Use the fitted model to make prediction on new dataset |

[16]

### 3.3 factoextra

Package `factoextra` is used to extract and visualise the output of PCA, Here are some of the R commands we used:

| | |
|---|---|
| `fviz_eig` | Create scree plot for all samples |
| `fviz_pca_ind` | Create score plot for all samples |
| `fviz_contrib` | Create contribution plot for all variables on selected PCs |

[17]

### 3.4 pls

Package `pls` is used to extract and visualize the output of PCA, Here are some of the R commands we used:

| | |
|---|---|
| `validationplot` | Plot validation statistics, such as RMSEP |

[18]

## 4 Analysis: Colitis Dataset

### 4.1 Model Selection

Gene expression analysis is crucial to discovering links between genes and diseases. Selecting the appropriate model to fit the dataset becomes a critical problem. In the following section,

we employed multivariate logistic regression with Ridge and LASSO penalties to fit the dataset and evaluate the accuracy of each model.

### 4.1.1 LASSO and Ridge Model

The `cv.glmnet` function returns two complexity parameters: *lambda.1se* and *lambda.min*. It is natural to ask which parameter is preferable when we choose the model. To address this, we created box plots to illustrate the distributions of accuracy under four different scenarios: Ridge with *lambda.1se*, Ridge with *lambda.min*, LASSO with *lambda.1se*, and LASSO with *lambda.min*. Note that accuracy is measured in misclassification rate.

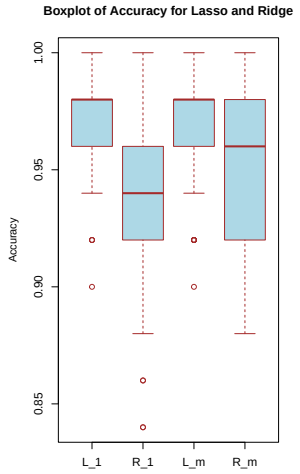


**Figure 7** Box plots of the accuracy of 4 regression models. L stands for LASSO, 1 stands for *lambda.1se*, and m stands for *lambda.min*

Based on the box plots(Figure 7), it is evident that the model with LASSO penalty performs better, but we still cannot decide which complexity parameter we should use on the model. Therefore, we plotted the histogram of the accuracy of two models with LASSO penalty and calculated the mean and variance.

From these two histograms(Figure 8 & Figure 9), we can see that while the LASSO model with *lambda.min* complexity parameter would yield more results with low bias, its variance is high compared to the LASSO model with *lambda.1se* complexity parameter. As a result, the biased-variance trade-off is important when we decide to use *lambda.1se* complexity parameter or *lambda.min* complexity parameter.

### 4.1.2 PCR

We first performed PCA on the entire dataset to see if we can perform PCR directly. However, the results are not good enough, the score plot does not demonstrate clear clustering (Figure 10), additionally, the percentage of variance explained by each principal component is low.

This result is mainly because PCR does not perform variable selection or apply any penalties. Despite this, PCA offers clear visualisation and is beneficial for validation purposes. This motivates further exploration to see the potential improvements when performing PCA and PCR on selected variables only.

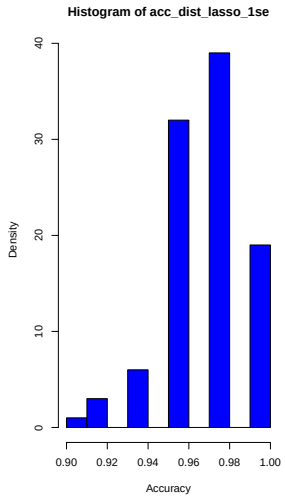


**Figure 8** Histogram of the accuracy of the model with LASSO penalty and complexity parameter *lambda.1se*

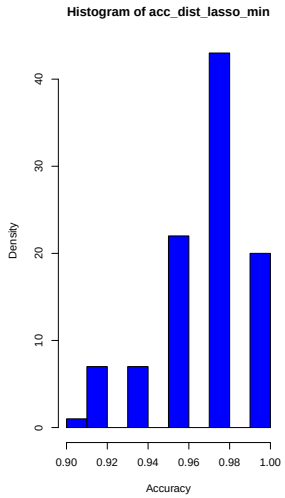


**Figure 9** Histogram of the accuracy of the model with LASSO penalty and complexity parameter *lambda.min*

## 4.2 Variable Selection

A critical aspect of gene expression data analysis is identifying genes that are significantly associated with the disease states. In the following section, we used univariate and multivariate logistic regression with LASSO penalty to identify these significant genes. Subsequently, we employed PCA and gene databases to validate and confirm our findings.

### 4.2.1 Selection

First, we utilised multivariate logistic regression with the LASSO penalty to identify genes whose corresponding coefficients do not shrink to zero. We considered using the complexity parameter *lambda.min* as it retains more coefficients in the model. This approach identified over 30 genes that are deemed important to the Colitis.

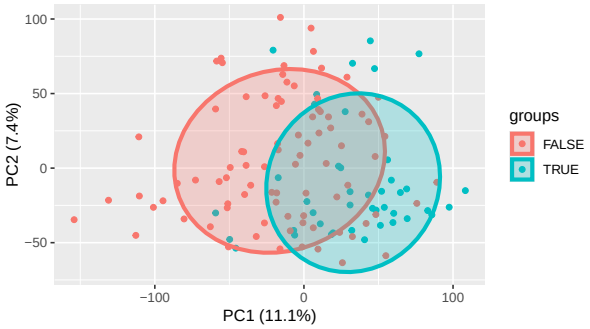


**Figure 10** Score plot for the dataset before variable selection
groups: TRUE = control, FALSE = case, subsequent graphs use the same group classification

Then we employed univariate logistic regression with Bonferroni correction to identify genes where the coefficient of their fitting model is significantly different from zero. This method identified around 500 genes that are deemed important to the Colitis.

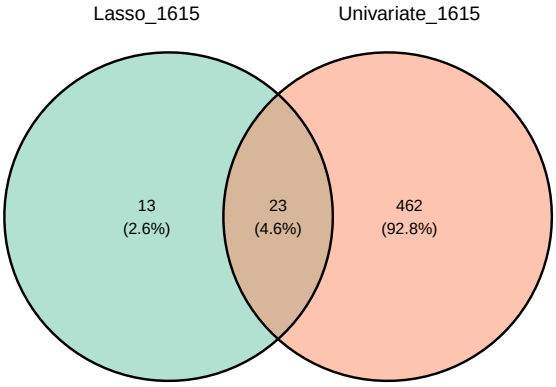


**Figure 11** Venn diagram of genes selected by LASSO model and Univariate model

To visually illustrate the relationship between these two sets of genes, we constructed a Venn diagram. Interestingly, the univariate model selected more genes than the LASSO method. Based on this observation, we hypothesised that the univariate method with Bonferroni correction selected all genes that are correlated with the truly significant genes.

To explore the correlation structure within the selected gene lists, we generated correlation heatmaps(Figure 12 & Figure 13). The heatmap for genes selected by the univariate method showed stronger correlations among the genes, indicating potential relationships among these

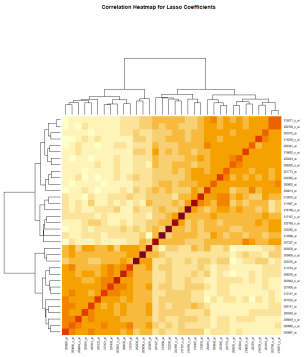


**Figure 12** Correlation heatmap of genes selected by LASSO model

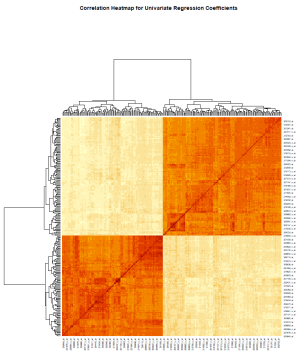


**Figure 13** Correlation heatmap of genes selected by Univariate model

genes. In contrast, the heatmap of genes selected by the LASSO method does not exhibit obvious correlations among each other, confirming our hypothesis.

### 4.2.2 Validation

We performed PCA on variables selected by LASSO Regression, Univariate Logistic Regression, and their intersection, to visualise and validate our results.

In order to determine the number of PCs used in the score plots, we first created scree plots for variables selected by LASSO, Univariate, and their intersection (Figures 14-16). In each of the three plots, we can see that although the first 2 PCs explain less than 80% of the variance, an elbow-like point appears at PC2, moreover, PC1 explains significantly more variance than the others. Therefore, we decided to try PC1 vs PC2 for the score plots to visualise the data.

Then, we created score plots on variables selected by LASSO, Univariate, and their intersection. In each of the PC1-PC2 score plots (Figures 17-19), the two clusters are almost totally separated. This suggests that it is strongly statistically significant that the selected variables are effective in distinguishing the samples by disease states.

We also tried score plots on other pairs of principal components, but none of them showed a clear cluster separation. This supports the assumption that the PCs that explain the largest variation in the predictor data (genes) are also the most informative for predicting the response

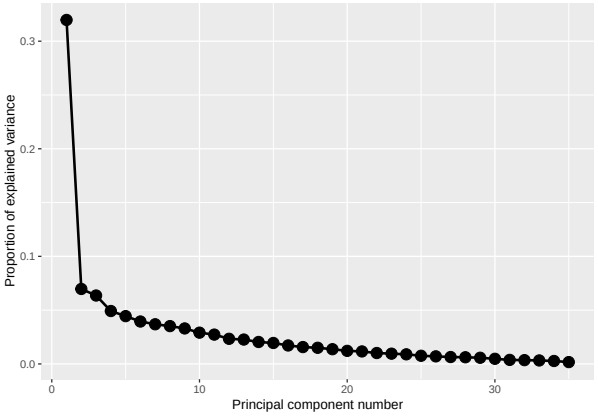


**Figure 14** Scree plot for variables selected by LASSO Regression

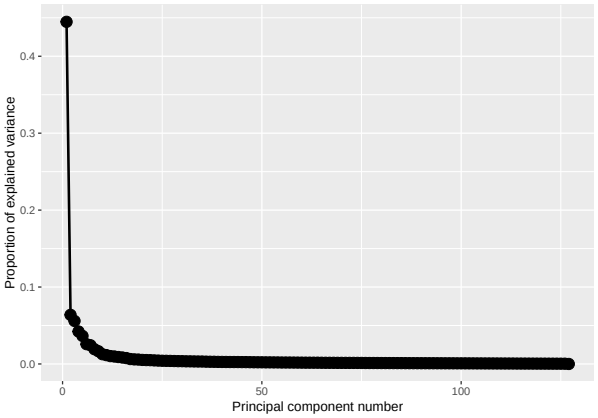


**Figure 15** Scree plot for variables selected by Univariate Regression

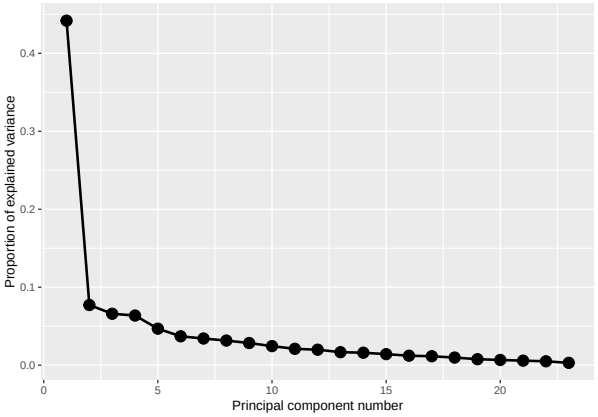


**Figure 16** Scree plot for variables selected by both LASSO and Univariate Regression

(disease state).

The loadings are very low for all variables selected by Univariate regression and therefore not suitable for contribution analysis. The contribution plots for variables selected by LASSO and their intersections (Figures 20-21) show that the variables with higher contributions identified by LASSO tend to be in the intersection set as well. This indicates that these variables are consistently identified as important across different selection methods, highlighting their potential significance.

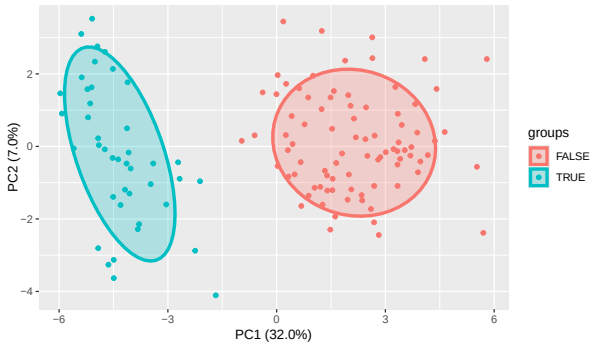


**Figure 17** Score plot for variables selected by LASSO Regression

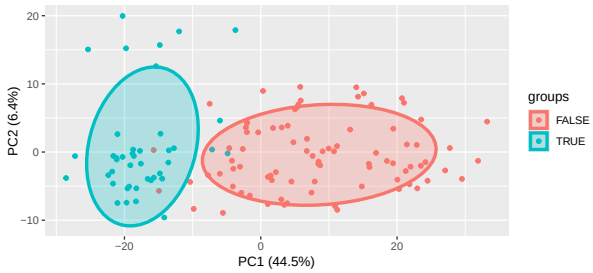


**Figure 18** Score plot for variables selected by Univariate Regression

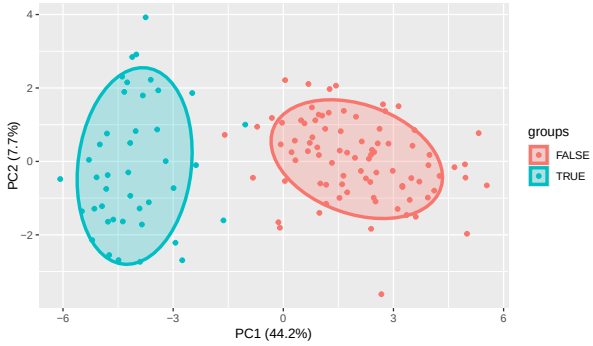


**Figure 19** Score plot for variables selected by both LASSO and Univariate Regression

## 4.3 Prediction using PCR on selected variables

Finally, we performed PCR on variables selected by both the LASSO model and the Univariate model.

We decided on choosing the number of components that minimise RMSEP to make predictions. In the RMSEP validation plot (Figure 21), the error is initially high due to the low number of components, indicating under-fitting. Adding the first few components decreases the error

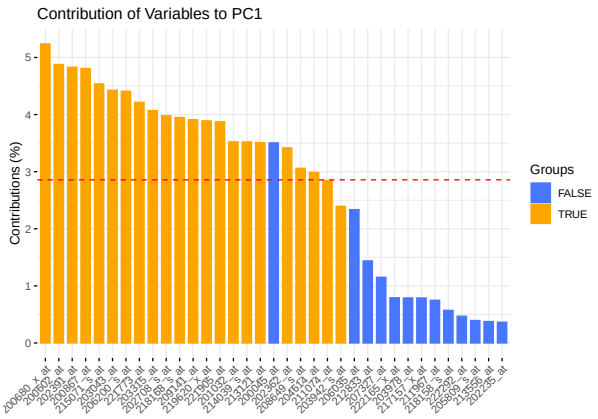


**Figure 20** Contribution plot for variables selected by LASSO Regression
groups: TRUE = variables selected by both LASSO and Univariate Regression, FALSE = variables selected by LASSO but not Univariate Regression, next graph uses the same group classification

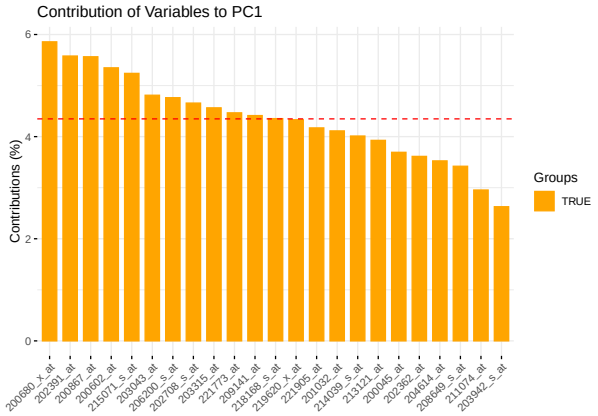


**Figure 21** Contribution plot for variables selected by both LASSO and Univariate Regression

sharply, indicating a significant performance improvement, however, adding further components only slightly enhances the model. Finally, after including 20 components, over-fitting occurs as the model begins to capture noise, leading to reduced performance. The minimum RMSEP occurs at 8 PCs.

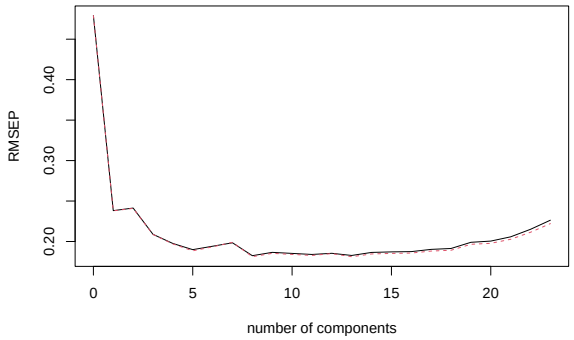


**Figure 22** RMSEP validation plot for variables selected by both LASSO and Univariate Regression, black line: ordinary CV estimate, red dotted line: bias-corrected CV estimate [19]

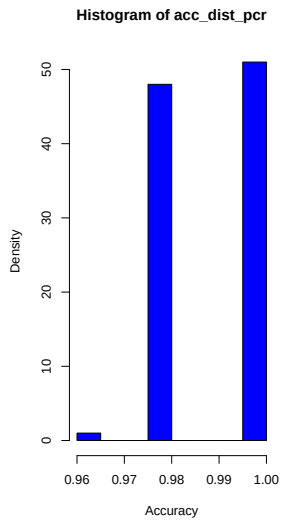


**Figure 23** Histrogram of the accuracy predicted by PCR

In the PCR predictive accuracy plot (Figure 23), we found that the predictive accuracy of the combined method outperformed the models we previously assessed. This suggests that using a model that combines LASSO and PCR might be a good choice for making predictions.

# 5 Analysis: Breast Cancer Dataset

## 5.1 Data Pre-processing

We found many missing values in the dataset. Before analysing the dataset, we need to pre-process the dataset so that we can continue our research.

### 5.1.1 Imputation Methods

Imputation is a statistical technique used to handle missing data in datasets, which is crucial in maintaining the integrity of data. In gene expression analysis, missing data might exist due to various reasons such as experimental errors, limitations in detection technologies, or sampling method issues. The imputation method aims to estimate and replace the missing data points with plausible values based on the observed data, allowing for more accurate and comprehensive analyses.

For columns with less than 50% missing values, we experimented with five different imputation methods to estimate the missing entries and compared the accuracy of the fitted model created by the imputed design matrix. We selected the model with LASSO penalty and complexity parameter *lambda.1se* as the LASSO model behaves better on the Colitis dataset and complexity parameter *lambda.1se* yields more consistent results. We chose the number 50% because it is intuitive to eliminate the columns with half of the missing data. Here are our methods:

1. Impute missing values with zero
2. Impute missing values with column mean

3. Impute missing values with column mean grouped in the same category(case or control)
4. Impute missing values by sampling in the same category(case or control)
5. Impute missing values by sampling for normal distribution with the same category(case or control)

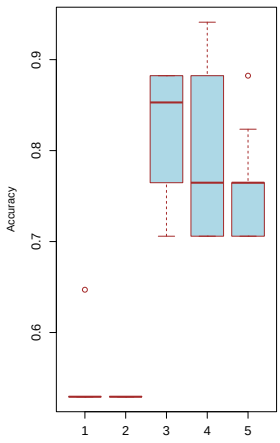


**Figure 24** Box plots of accuracy between different imputation methods on Breast Cancer dataset

Based on the result of the boxplot, method 3 performs the best, providing the most accurate and consistent results across other imputation methods. In subsequent analyses, we will choose method 3 as our imputation method.

### 5.1.2 Cut-off Percentage Decision

What percentage of missing values should genes have to be removed from the design matrix? To effectively manage missing data in our analysis, we began by creating a histogram to visualize the distribution of missing data across the dataset. This graphical representation allowed us to understand the extent and pattern of missing values.

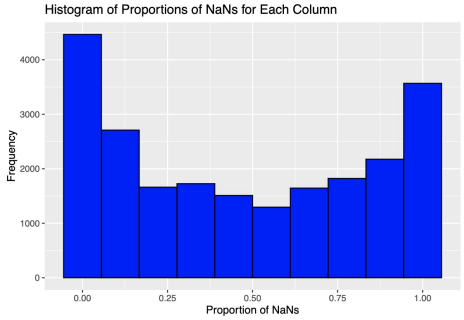


**Figure 25** Histogram of the distribution of missing values

Following this, we generated boxplots to compare the accuracy of different cut-off thresholds for missing data, ranging from 10% to 50%. We used the model with LASSO penalty and complexity parameter *lambda.1se* to calculate the accuracy for the same reason as section 5.1.1.

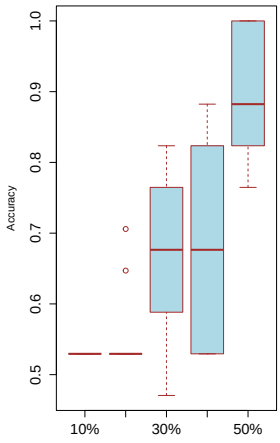


**Figure 26** Box plots of accuracy between different cut-off percentages on Breast Cancer dataset

Our findings indicated that a 50% cut-off threshold provided the best balance between retaining sufficient data for analysis and minimising the influence of missing values. This optimal threshold ensures the robustness and reliability of the subsequent steps in our data processing and analysis.

## 5.2 Model Selection

### 5.2.1 LASSO and Ridge Model

Using the same approach, we determined which model and complexity parameter perform best on the cancer dataset. Here is the distribution of four models:

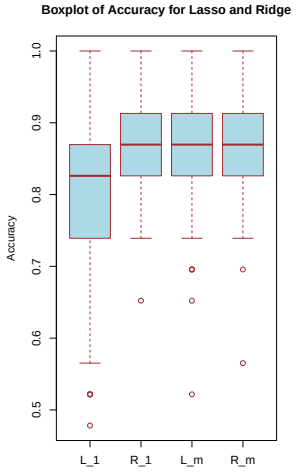


**Figure 27** Box plots of the accuracy of 4 regression models. L stands for LASSO, 1 stands for *lambda.1se*, and m stands for *lambda.min*

Based on the graphs, the models with the Ridge penalty perform better. To further determine the complexity parameter, we plotted the histogram of these two models and calculated their

mean and variance. From the histogram and our calculation, the Ridge model with *lambda.min* complexity parameter has lower bias and higher variance compared to the Ridge model with *lambda.1se* complexity parameter. Similar to the colitis dataset, there is also a bias-variance trade-off when determining the complexity parameter for the Breast Cancer dataset.

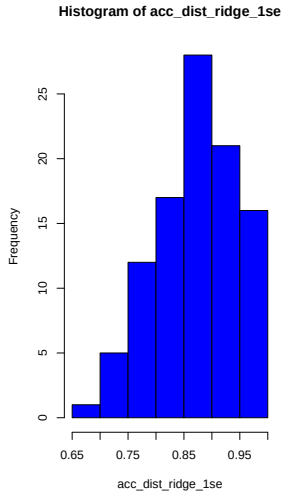


**Figure 28** Histogram of the accuracy of the model with Ridge penalty and complexity parameter *lambda.1se*

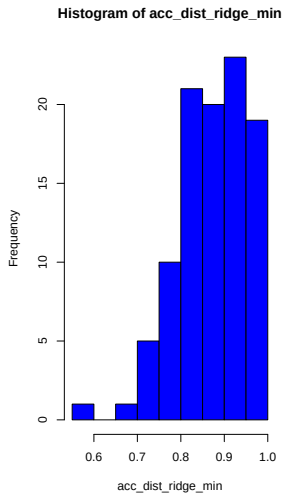


**Figure 29** Histogram of the accuracy of the model with Ridge penalty and complexity parameter *lambda.min*

### 5.2.2 PCR

We first performed PCA on the entire dataset to see if we could perform PCR directly. However, the results are not good enough, the score plot does not demonstrate clear clustering (Figure 30), additionally, the percentage of variance explained by each principal component is low.

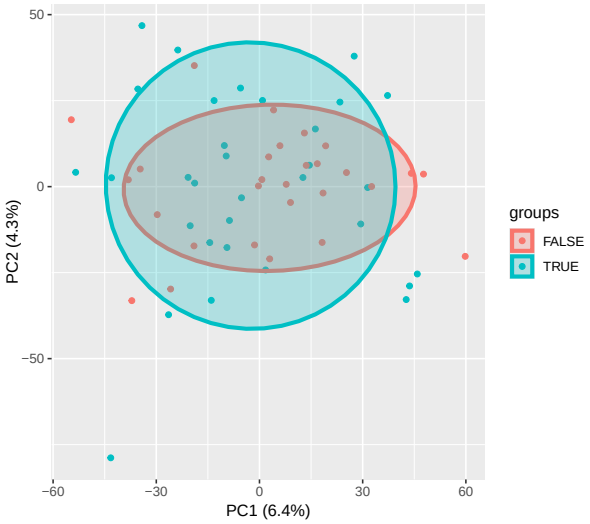


**Figure 30** Score plot for the dataset before variable selection

## 5.3 Variable Selection

### 5.3.1 Selection

When we employed univariate logistic regression with Bonferroni correction on the breast cancer dataset, we were surprised to find that the number of selected genes was zero. As a result, we turned to using False Discovery Rate (FDR) correction, which is less stringent. However, the number of genes selected with FDR correction was also zero.

We then used multivariate logistic regression with the LASSO penalty to identify the important genes. This approach pinpointed over 30 genes that are deemed important to the disease.

### 5.3.2 Validation

We performed PCA on variables selected by LASSO Regression to visualise and validate our results.

In order to determine the number of PCs used in the score plots, we first created the scree plot (Figure 31). We see that although the first 2 PCs explain less than 80% of the variance, an elbow-like point appears at PC2, moreover, PC1 explains significantly more variance than the others. Therefore, we decided to try PC1 vs PC2 for the score plots to visualise the data.

Then, we created score plots on variables selected by LASSO. In the PC1-PC2 score plot (Figure 32), the two clusters are almost totally separated. This suggests that it is strongly statistically significant that the selected variables are effective in differentiating the samples by disease states.

We also tried score plots on other pairs of principal components, but none of them showed a clear cluster separation. This supports the assumption that the PCs that explain the largest variation in the predictor data (genes) are also the most informative for predicting the response

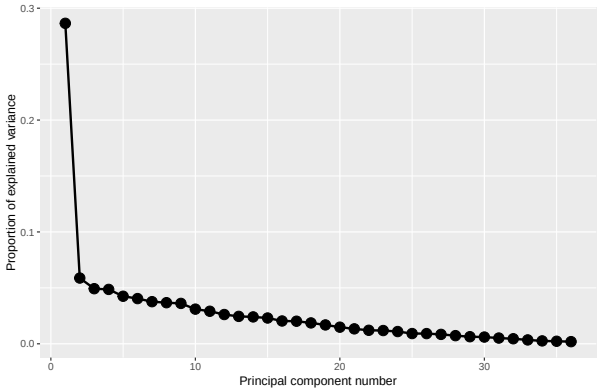


**Figure 31** Scree plot for variables selected by LASSO Regression

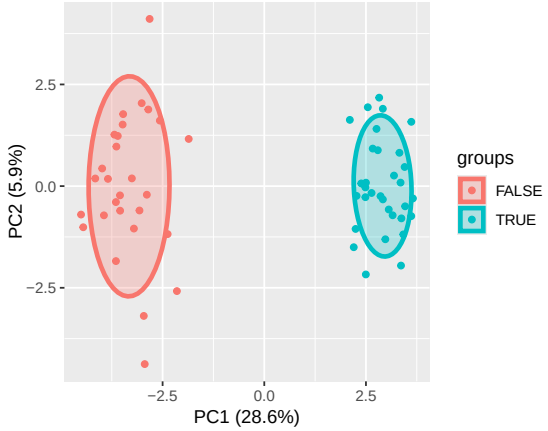


**Figure 32** Score plot for variables selected by LASSO Regression

(disease state).

## 5.4 Prediction using PCR on selected variables

Finally, we performed PCR on variables selected by LASSO Regression.

We decided on choosing the number of components that minimise RMSEP to make predictions. In the RMSEP validation plot (Figure 33), the error is initially high due to the low number of components, indicating under-fitting. Adding the first few components decreases the error sharply, indicating a significant performance improvement, however, adding further components only slightly enhances the model. Finally, after including 27 components, over-fitting occurs as the model begins to capture noise, leading to reduced performance. The minimum RMSEP occurs at 13 PCs.

A similar result is found in the Breast Cancer dataset for the model that combined LASSO and PCR. The accuracy for the combined model reached 100%, which also behaves better than the previous models, which confirms that people should consider using the combined model instead of using a single model when making predictions.

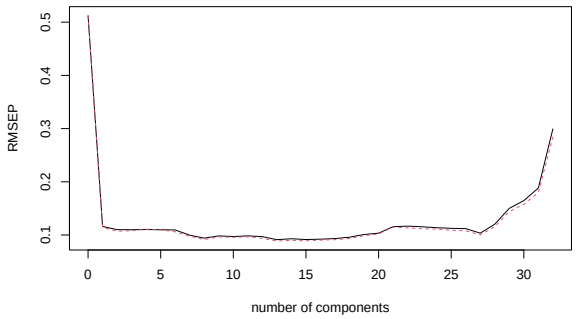


**Figure 33** RMSEP validation plot for variables selected by LASSO Regression

# 6 Discussion

## 6.1 Criticism

We set up our research based on two datasets from GEOquery, and some limitations due to the datasets were introduced.

We can see clearly that the sample size is much smaller than the number of genes in both two datasets. If more samples are used in the model, we can improve our model for gene selection.

Since 50% of the data in the breast cancer dataset(GDS807) is missing, all research done based on this dataset would be not as perfect as those done on the dataset for colitis (GDS1615), which does not have any missing data. Not all data we used in our project is measured as all the missing parts of data are imputed with method 3, i.e. with column mean grouped in the same category(case or control). Even though this imputation method has the highest accuracy among all the methods we considered, the data is still not as perfect as if it had been fully measured. Some bias might be introduced from this imputation method and the model we set up might be influenced by this.

## 6.2 Future

For future research, we would like to try additional imputation methods. One such method is multiple imputation, which generates multiple sets of plausible values for the missing data and creates several complete datasets. This statistical approach can lead to a more accurate model. We could also consider applying and comparing different ensemble learning methods in modeling, which will make more accurate predictions.

# Acknowledgments

We would like to thank Marina Evangelou, our project supervisor, for her invaluable suggestions and insights throughout our work.

## Appendix

Our GitHub repository: https://github.com/fzfzzfzzzfzzzz/Gene-Analysis-M2R

## References


[1] U. of California Los Angeles Health Sciences. Groundbreaking study connects genetic risk for autism to changes observed in the brain, May 23, 2024.

[2] B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. Molecular biology of the cell. 4th edition., 2002.

[3] F. Fabio and E. Marina. Sparse-group slope: adaptive bi-level selection with fdr-control. Technical report, May 16 2023.

[4] D. Jurafsky and J. Martin. *Logistic Regression*, pages 1–4. Speech and Language Processing. Feb 3, 2024.

[5] T. Hastie, R. Tibshirani, and J. H. Friedman. *The Elements of Statistical Learning*. Springer, New York [u.a.], 2. ed., corrected at 12. print. edition, 2009.

[6] S. Rosset and J. Zhu. Piecewise linear regularized solution paths. *The Annals of statistics*, 35(3):1012–1030, Jun 1, 2007.

[7] M. D. Ward and J. S. Ahlquist. *Maximum likelihood for social science : strategies for analysis*. Cambridge University Press, Cambridge, 2018.

[8] 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.

[9] I. T. Jolliffe and J. Cadima. Principal component analysis: a review and recent developments. *Philosophical transactions of the Royal Society of London. Series A: Mathematical, physical, and engineering sciences*, 374(2065):1–16, Apr 13, 2016.

[10] B. A. Hanson and D. T. Harvey. *LearnPCA: Functions, Data Sets and Vignettes to Aid in Learning Principal Components Analysis (PCA)*, 2024.

[11] A. M. Goodpaster and M. A. Kennedy. Quantification and statistical significance analysis of group separation in nmr-based metabonomics studies. *Chemometrics and Intelligent Laboratory Systems*, 109(2):162–170, 2011. ID: 271351.

[12] G. James, D. Witten, T. Hastie, R. Tibshirani, and J. E. Taylor. *An introduction to statistical learning*. Springer, Cham, 1 edition, 2023.

[13] K. Dunn. Principal component regression (pcr), Feb 1, 2023.

[14] A. Cook. Cross-validation. https://www.kaggle.com/code/alexisbcook/cross-validation, 2020.

[15] S. Davis and P. Meltzer. Geoquery: a bridge between the gene expression omnibus (geo) and bioconductor. *Bioinformatics*, 14, 2007.

[16] J. K. Tay, B. Narasimhan, and T. Hastie. Elastic net regularization paths for all generalized linear models. *Journal of Statistical Software*, 106(1):1–31, 2023.

[17] A. Kassambara and F. Mundt. *factoextra: Extract and Visualize the Results of Multivariate Data Analyses*, 2020.

[18] K. H. Liland, B.-H. Mevik, and R. Wehrens. *pls: Partial Least Squares and Principal Component Regression*, 2023.

[19] B.-H. Mevik and H. R. Cederkvist. Mean squared error of prediction (msep) estimates for principal component regression (pcr) and partial least squares regression (plsr). *Journal of Chemometrics*, 18(9):422–429, Sep 2004.