MATH42715: Introduction to Statistics for Data Science
Assignment No. 2 | Student No. 001059181

Binary Classification Problem using BreastCancer Dataset

1. Introduction

Breast cancer is one of the most commonly diagnosed forms of cancer in women with one woman diagnosed with breast cancer in the UK every ten minutes.[1] It is also the second leading cause of death among women after lung cancer.[2] This high incidence of the disease as well as its high mortality rate signals an urgent need to examine available data on the disease and develop effective interventions to reduce its incidence. In this context, the present enquiry is a binary classification project. Utilising the BreastCancer data set from the `mlbench` library, it builds three classifiers, namely, a logistic regression classifier, a LDA classifier, and a QDA classifier, which, based on a 7 (of the 9) predictor variables predicts whether a tumour of a patient is malignant or not. Thereafter, the three classifiers are compared in terms of the respective test errors.

[,1]	Id	Sample code number
[,2]	Cl.thickness	Clump Thickness
[,3]	Cell.size	Uniformity of Cell Size
[,4]	Cell.shape	Uniformity of Cell Shape
[,5]	Marg.adhesion	Marginal Adhesion
[,6]	Epith.c.size	Single Epithelial Cell Size
[,7]	Bare.nuclei	Bare Nuclei
[,8]	Bl.cromatin	Bland Chromatin
[,9]	Normal.nucleoli	Normal Nucleoli
[,10]	Mitoses	Mitoses
[,11]	Class	Class

Table 1. Variables in the BreastCancer dataset

The dataset comes from the Wisconsin Breast Cancer Database and includes data from breast tissue samples collected using fine needle aspiration cytology (FNAC) as Dr. William H. Wolberg reported his clinical cases. There are 699 samples (observations) collected periodically from January 1989 to November 1991.[3] For each observation, there is data on 9 variables [see Table 1, rows 2-10]. Each observation under each variable has been converted to a value ranging from 0 to 10. The first variable provides an identification number. The last variable titled "Class" classifies whether the sample is benign (cancer-free) or malignant (cancerous).

2. Data Pre-processing

Prior to building a classifier, it is necessary to pre-process the data because the algorithms used in the classifier relies on this data to carry out its functions. The original data BreastCancer which has 699 rows and 11 columns, was cleaned in four steps, namely, conversion of all variables to quantitative variables, conversion of Class variable to 0, 1 levels, removal of missing values, and removal of ID column. Steps taken to verify the effectiveness of the data cleaning process are also detailed at each stage.

2.1. Conversion to Quantitative Variables. The dataset has 11 variables. Using the **str** function in R, the structure of each of these variables were checked. The first variable, ID, provides a sample code number and is a character variable, while the last variable, Class, is encoded as a factor/target class with two levels "benign" or "malignant". The remaining variables i.e. nine cytological characteristics in the dataset, are encoded factors as ordinal variables on a 1-10 scale.

[1] "Facts and Statistics 2021." *Breast Cancer Now.* https://breastcancernow.org/about-us/media/facts-statistics

[2] Giaquinto, A.N., Sung, H., Miller, K.D., Kramer, J.L., Newman, L.A., Minihan, A., Jemal, A. and Siegel, R.L. (2022), Breast Cancer Statistics, 2022. CA A Cancer J Clin, 72: 524-541. https://doi.org/10.3322/caac.21754

[3] Breast Cancer Wisconsin (Original) Data Set. UCI Machine Learning Repository. https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+(original)

They are classified as ordinal because there is a clear ordering (via a 1-10 scale). These variables were converted to, and treated as, quantitative variables by executing a simple loop which converts the variables using the **as.numeric** function. To ensure that the loop worked as intended, the **str** function was used, which showed that all 11 columns are now numeric.

2.2. Conversion of Class Variable to 0,1 levels. The variable "Class" is a factor with two levels "benign" and "malignant" denoted by 1 and 2 respectively. This variable was converted to numeric using dummy variables so that 0 denotes a benign tumour and 1 denotes a malignant tumour. To verify that the Class variable now utilises dummy variables, the **head** function was used. This showed the first six rows converted to 0,0,0,0,0,1 instead of "benign", "benign", "benign", "benign", "benign", "malignant".

2.3. Identification and Removal of Missing Values. The **is.na** function was used to identify missing values in the data. The **sum(is.na)** function showed that there are 16 missing values which confirmed R documentation which states that there are 16 missing attribute variables. The **colSums** function showed that it is the bare nuclei variable that is missing all 16 missing values. Using the **na.omit** function, the relevant rows with NA values were identified and removed. To ensure that the **na.omit** function was effective, **sum(is.na)** function and **colSums(is.na)** was used which showed there were now zero NA values in the dataset and in the bare nuclei column. Further, the total no. of rows reduced from 699 to 683 indicating that 16 rows have been removed.

2.4. Removal of ID column. Finally, to prepare the data for analysis, the first column was removed. This column only contains the ID number, presumably identifying the sample, and does not contain data regarding the samples themselves.

3. Preliminary Observations and Correlations: Exploratory Data Analysis

In the cleaned dataset, there are 683 rows and 10 columns (9 cytological features and the Class variable). The **table** function shows that there are 444 cases of benign breast tumours and 239 cases of malignancy. Therefore, of the 683 observations, there are more benign cases than malignant cases.

table(BC3$Class)	
0	1
444	239

Table 2.

A brief overview of each of the 9 cytological features proves useful as it provides insight into the data: (1) clump thickness indicates how the cells are grouped (if grouped in multilayers, which is often the case in malignant cells, the value given is higher), (2) cell size uniformity which relates the size of the cell indicates whether the cancer has metasized (spread), (3) cell shape uniformity indicates the varying sizes of the cells, (4) marginal adhesion (specifically, its loss) is an indication of malignancy because malignant cells lose the ability to adhere whereas benign cells stick together, (5) epithelial cell size indicates that if the cell size becomes larger, it can be indicative of malignancy, (6) bare nuclei relates to whether the nuclei has a cytoplasm coating, if it does i.e. the nuclei is not "bare", it is indicative of malignancy, (7) bland chromatin relates to the texture of the nucleus; if the texture of the chromatin is not uniform but is coarser, it can be indicative of malignancy, (8) normal nucleoli refers to the structures within the nucleus where, if the nucleoli is prominent and not small, it may be indicative of malignancy, and finally, (9) mitoses refers to how the nucleus divides (cell division) and if there is greater cell division, the greater the possibility of malignancy.[4][5] It is to be noted that the terminology used here is *indicative;* it merely

[4] Nag, A., & Sarkar, S. (2017). Identifying Patients at Risk of Breast Cancer through Decision Trees. *International Journal of Advanced Computer Research, 8.*

[5] Santiago-Montero, R., Sossa, H., Gutiérrez-Hernández, D. A., Zamudio, V., Hernández-Bautista, I., & Valadez-Godínez, S. (2020). Novel Mathematical Model of Breast Cancer Diagnostics Using an Associative Pattern Classification. *Diagnostics, 10*(3), 136. https://www.mdpi.com/2075-4418/10/3/136

remarks upon the possibility of malignancy, and does not draw a causal link between, for example, high epithelial cell size and malignancy.

As visualisation can provide insight into how the variables relate to one another, a pairwise scatterplot (black and white) of the predictor variables was plotted (see Appendix, Figure 1). However, as it is difficult to draw insights from this, the same scatterplot was replotted with malignancy being represented by a different colour (see Appendix, Figure 2, Figure 3).

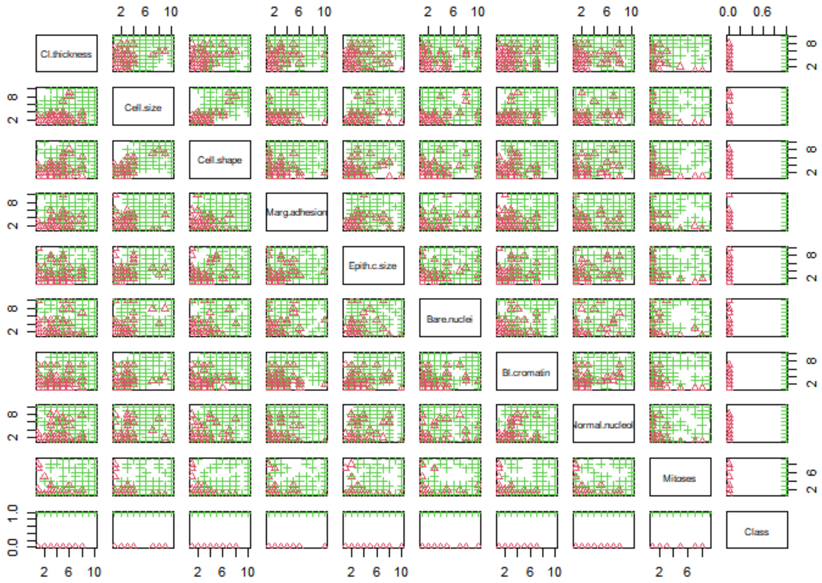
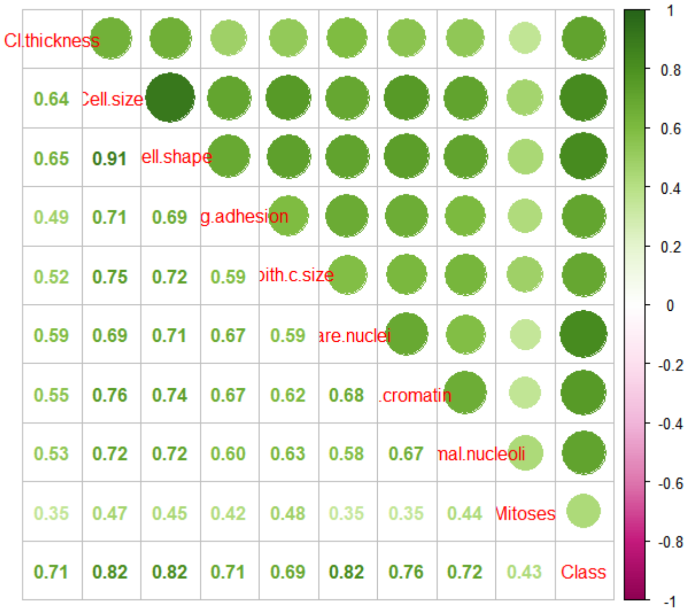


Figure 3. Pairwise Scatterplot of BreastCancer Data: Malignancy in Green

Figure 3 represents the observations which denote benign tumours in red triangles and the observations which denote malignancy in green crosses. This provides insight into how the values between 0 and 10 have been assigned to each variable. The clustering of red triangles towards the left-bottom corner of each pair provides initial insight that benign tumours score lower on the scale of 1-10 i.e. the healthier the sample, the lower the score. Similarly, the prominent presence of green crosses as the value increases along the 0-10 scale indicates that as the value of each variable is higher, the likelihood of malignancy increases. Furthermore, there appears to be a positive correlation between two predictor variables, specifically, cell size and cell shape i.e. as cell size increases, the cell shape appears to increase. This indicates the possible presence of multicollinearity.



Cl.thickness									
0.64	Cell.size								
0.65	0.91	ell.shape							
0.49	0.71	0.69	g.adhesion						
0.52	0.75	0.72	0.59	ith.c.size					
0.59	0.69	0.71	0.67	0.59	are.nuclei				
0.55	0.76	0.74	0.67	0.62	0.68	.cromatin			
0.53	0.72	0.72	0.60	0.63	0.58	0.67	mal.nucleoli		
0.35	0.47	0.45	0.42	0.48	0.35	0.35	0.44	Mitoses	
0.71	0.82	0.82	0.71	0.69	0.82	0.76	0.72	0.43	Class

Figure 4.
Correlation Panel: Correlation Coefficients of BreastCancer Variables

In order to gain further insight into the possible correlations between these variables, Figure 4 with the coefficient correlations between the variables were plotted using the `corrplot` library. In this correlation panel, the legend colour shows the correlation coefficients and the corresponding colours. High, positive correlations are indicated in a green while low, negative correlations are indicated in pink. In the BreastCancer data, it is clear that all predictor variables are positively correlated as all of the coefficients (albeit to varying degrees) are shaded in green, and all of the correlation coefficients in the lower panel are positive. If any of the coefficients were shaded in pink, it would have indicated a negative correlation between the relevant variables. The intensity of the colour, as well as the size of the circles are proportional to its correlation coefficient: where the circle is dark green and larger, the correlation coefficient is high whereas where the circle is lighter and smaller, the correlation coefficient is low.

3.1. Relationship between predictor variables. Figure 4 reveals that cell size appears highly correlated with cell shape with a correlation coefficient of 0.91 reinforcing the preliminary observation made using Figure 3. This makes sense as the size of a malignant cell increase, it would increasingly affect the cell shape as well. Thus, it appears that these two variables demonstrate multicollinearity i.e. that the two predictor variables might be correlated with each other,[6] which in turns is confirmed by checking the variation inflation factor (VIF) using the `car` library which records above 5 values for both these variables (see Appendix, Figure 4.1). Cell size also correlates to a great degree with bland cromatin (0.76), epithelial cell size (0.75), normal nucleoli (0.72) and marginal adhesion (0.71). This makes sense as the less healthy the cell is, its cell size (including single epithelial cell size) would increase, affect the nucleoli, and reduce adhesion. Similarly, cell size correlates to a continued yet somewhat lesser degree with bare nuclei (0.69) and cell thickness (0.64) while cell shape similarly correlates to bland cromatin (0.74) and normal nucleoli (0.72). This makes sense as the texture of chromatin becomes coarser, it could affect cell shape while the changes in the structure of the normal nucleoli could also affect cell shape. Further, it is likely, therefore, that only one of these variables, cell size and cell shape, is required when fitting a logistic regression model. It further appears that while there is some degree of correlation between all the variables, the least degree of positive correlation is between mitoses and the remaining predictor variables.

3.2. Relationship between response and predictor variables. There are several variables which correlate highly to the response variable, specifically, cell size, cell shape and bare nuclei, as all three variables have a correlation coefficient of 0.82 with the response variable. It is, therefore, highly likely that these variables would be essential to build the classifier, noting however, that only one of the variables cell size and cell shape may be required as these two variables appear highly correlated. In addition to these three variables, all the other predictor variables have relatively high correlation (more than 0.69) coefficients with the response variables except for Mitoses. While mitoses, the degree to which the nucleus divides, is positively correlated with findings of malignancy, with a coefficient of only 0.43, it does not appear to be as highly correlated as the other variables. This indicates that mitoses maybe the first variable dropped when selecting the most suitable variables to build a classifier.

4. Modelling

Three modelling techniques are used to build a classifier: (1) logistic regression, (2) linear discriminant analysis, and (3) quadratic discriminant analysis.

4.1. Logistic Regression. Logistic regression is generally used with a qualitative (i.e. two-class or binary) response variable (in contrast, linear regression predicts a quantitative response), and is thus a classification method. While it can be extended to instances where there are more than 2 categorical variables ($K > 2$), for example, multinomial logistic regression, where there is multiple class classification, discriminant analysis is more popularly used. Logistic regression models the *probability* that a specific observation, Y, belongs to a specific category.

[6] Gareth James, D. W., Trevor Hastie, Robert Tibshirani,. (2021). *An introduction to statistical learning : with applications in R* (Second ed.). New York : Springer, [2013] ©2013. https://search.library.wisc.edu/catalog/9910207152902121

4.1.1. Assumptions. Logistic regression assumes that the data has the Bernoulli distribution (a special case of the binomial distribution where Y takes two possible values). In the present enquiry, the values are 0 (for benign) or 1 (for malignant). This modelling technique also assumes linearity of independent variables, absence of influential outliers, absence of multicollinearity, independence of observations, and large sample size.

4.1.2. Full Fitted Model. First, the response variable Class was regressed using the **glm** function on *all predictor variables* to gain insight into (i) how the full model performs, and (ii) which variables may not be as useful. The argument family="binomial" is set to indicate that a logistic regression model, and not a different kind of generalised linear model, is fitted. When doing so, the **glm** function uses maximum likelihood estimation to derive confidence intervals for the regression coefficients. The summary output of the full model yields insights (see Appendix: Figure 5 for full summary). When inspecting the p-value column, it is clear that there are several variables with coefficients very close to zero when testing at a 95% confidence interval. Assuming the null and alternate hypothesis i.e. $H_0: \beta_i = 0$ versus $H_1: \beta_i \neq 0$, the null hypothesis at the 5% level would be rejected when $i = 1$, $i = 4$, $i = 7$. Therefore, in evaluating the p-values generated after fitting the full model using logistic regression, it may be surmised that the best fitted logistic regression model need not include all predictor variables. Reducing the number of variables can be useful to, on one hand, improve predictive performance, and on the other hand, improve model interpretability. Further, given its p-values, it is possible that cell thickness, marg.adhesion, Bl. cromatin maybe variables that would be more useful in fitting the model.

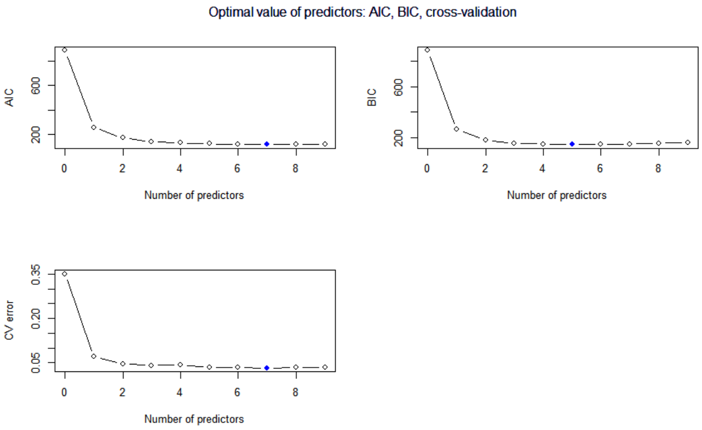
4.2. Variable Selection: Best Subset Selection. While there are a number of methods to decide which predictors to select, the method adopted in this report is the best-subset selection method. This method is a sub-category of variable selection or feature selection methods. Variable selection methods allow for greater precision. Because a reduced set of predictors reduces the variance in parameter estimates, a model with reduced set of variables has better predictive performance.

To perform best subset selection, the **bestglm** function from the `bestglm` library was used. When applied to logistic regression, the function uses negative loglikelihood whereas in multiple linear regression, it uses residual sum of squares. To use the **bestglm** function, the last column in the data frame must be the response variable (i.e. y) preceded by the predictor variables. In the present enquiry, it was verified that the last column is Class i.e. the response variable (using the **head** function). In order to compare the best-fitting models containing 0 to *p* predictors, a number of model-comparison criteria can be used. This enquiry used three commonly used criterion, namely, Akaike information criterion (AIC) which, in this context is identical to Mallow's Cp Statistic, Bayesian information Criterion (BIC), and k-fold cross-validation.

4.2.1.AIC and BIC. The best subset selection using the AIC and BIC were applied by using two applications of the **bestglm** function (specifying family = binomial as the default is gaussian, and IC (information criteria) as "AIC" and "BIC" respectively). The best subsets for models with variables 0 to 9 for both AIC and BIC are given in Appendix: Figures 6 and 7. respectively. While the AIC prefers a model with 7 predictors, the BIC prefers a model with 5 predictors.

4.2.2. k-fold cross validation. The third criterion, k-fold cross validation, is the process where the data is randomly split into a given number of sections (k) of equal size, and each model is trained on k-1 models, the predictions of each model is compared to the remaining unused section, and the process is repeated k times [further explanation is given in Section 5.1 below]. Once cross validation is applied, the test error for each model was calculated (see Appendix: Figure 8), and the number of predictors in the model which minimises test error was identified as 7 (using the **which.min** function).

Figure 9.



4.2.3. Selection of no. of Predictors. While the BIC prefers a model with 5 predictors, both AIC and cross-validation method suggests a 7-predictor model as having the lowest prediction error. This can be visually plotted as seen in Figure 9. In both AIC and BIC, the error is very close to zero where the no. of predictors is 6 upwards. This is so similarly in BIC where the error is minimised in models with 4 predictors and upwards. In the cross-validation method, the error seems to reduce most at 7 predictors as the error appears to reduce even further from a 6-predictor and 8-predictor model. These observations, coupled with the fact that two methods suggest a 7-predictor model, led to the selection of a 7-predictor model.

4.3. Logistic Regression: 7-Predictor Model. In order to select the variables for the best-fitting 7-predictor model, the AIC was used to extract the relevant subset of variables. The new 7-predictor model includes the following 7 variables: cell thickness, cell shape, marginal adhesion, bare nuclei, bland cromatin, normal nucleoli, and mitoses. The **glm** function was used to fit the logistic regression model. The coefficients for the intercept and the other variables are unknown; thus, the **glm** function uses maximum likelihood estimation to derive confidence intervals for the regression coefficients. In the summary output (see Appendix: Figure 10 for full summary), the maximum likelihood estimates of the regression coefficients (to three decimal places) were as follows:

$$\hat{\beta}_0 = \text{-}9.989 \quad \hat{\beta}_1 = 0.534 \quad \hat{\beta}_2 = 0.345 \quad \hat{\beta}_3 = 0.342 \quad \hat{\beta}_4 = 0.388 \quad \hat{\beta}_5 = 0.462 \quad \hat{\beta}_6 = 0.226 \quad \hat{\beta}_7 = 0.535$$

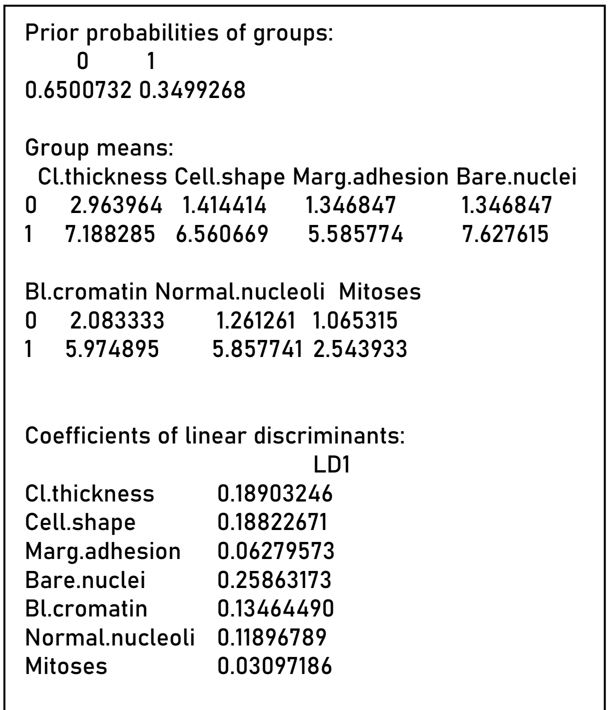
All the estimates (except for the Intercept) are positive. This indicates that as each of these variables increase, the likelihood (probability) of malignancy increases. For example, a one unit increase in $\hat{\beta}_1$ i.e. cell thickness is associated with an increase in the log odds of malignancy by 0.534 units. Further, majority of the p-values, in particular cell thickness (0.000146), marginal adhesion (0.004060), bland cromatin (0.005997), and normal nucleoli (0.041561) are quite small, indicating a possible association between each of these variables and Class.

5.4. Discriminant Analysis. The Bayes Classifiers for both Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA) were modelled.

5.4.1. Assumptions. LDA assumes that the observations in each class are (1) multivariate normal (Gaussian), (2) has a class specific mean, and (3) a common covariance matrix. QDA is a non-linear discriminant analysis which assumes, like LDA, that the observations in each class are (1) multivariate normal, and (2) has a group-specific mean vector. However, unlike LDA, QDA assumes that each has a different covariance matrix i.e. QDA does not assume equal covariance.

5.4.2. Bayes Classifier for LDA and QDA, and Output. The Bayes Classifiers for both LDA and QDA built using the 7-predictor dataset. Both these discriminant methods were used as the number of variables are high and it is difficult to ascertain the presence or absence of variance.

The **lda** function was used to fit the model for LDA, and the **qda** function was used to fit the model for QDA, both from the MASS library. The complete function outputs for both models are given in Figures 11 and 12 (see Appendix). In these outputs, the prior probabilities of groups and the group means are insightful. At the outset, it is clear that the prior probabilities and group means are the same for both models. The prior probabilities of groups 0 and 1 indicate that 65% of the training observations are benign while 34.99% are malignant. The group means (i.e. the average of each predictor variable in each class 0, 1) suggest that, on average, each of the predictors score a lower value when the sample is benign in comparison to when there is malignancy. For instance, where the sample is malignant, cell thickness averages 7.188 whereas where the sample is benign, cell thickness is less than half (2.963). Because there are only two response classes, there is only one set of coefficients of linear discriminants (LD1) in LDA. These coefficients provide the linear combination of the variables which are used to form the LDA decision rule[7] i.e. if 0.18903246 x cl.thickness – 0.18822671 x cell.shape - 0.06279573 x marg.adhesion (and so on) is large, the LDA classifier will predict malignancy and vice versa. The QDA output does not provide coefficients of the linear discriminants because, naturally, QDA is a quadratic (and not linear) function of the predictor variables.



```
Prior probabilities of groups:
        0         1
0.6500732 0.3499268

Group means:
  Cl.thickness Cell.shape Marg.adhesion Bare.nuclei
0     2.963964   1.414414      1.346847    1.346847
1     7.188285   6.560669      5.585774    7.627615

  Bl.cromatin Normal.nucleoli  Mitoses
0    2.083333        1.261261 1.065315
1    5.974895        5.857741 2.543933


Coefficients of linear discriminants:
                       LD1
Cl.thickness    0.18903246
Cell.shape      0.18822671
Marg.adhesion   0.06279573
Bare.nuclei     0.25863173
Bl.cromatin     0.13464490
Normal.nucleoli 0.11896789
Mitoses         0.03097186
```

Figure 11. LDA Fitted Model output

5. Model Comparison

In order to compare the three models, the test error for each model was calculated. The test error is a measure that is used to check the accuracy of the model. Unlike the training error (which is calculated using the same data used to train the model), the test error is calculated using disjoint data sets. To do this, k-fold cross validation was adopted.

5.1. K-fold validation. This is the process where the data is randomly split into a given number of sections or subsets (k) of equal size. Thereafter, the first fold is treated as the validation set, and the method is fit on the remaining k – 1 folds i.e. each model is trained on k-1 models, and the predictions of each model is compared to the remaining unused section. This process is repeated k times. While there is no specific rule as to how many folds can be set, because there is a bias-variance trade-off associated with the choice of k, often k = 5 or k = 10 is selected because these values have shown to provide test error estimates that do not have an excessively high bias nor variance.[8] Therefore, in the present exercise, a value of k = 10 was selected. The data was randomly split into ten sections, each model trained on 9 sections and the prediction of each model was thereafter compared to the remaining 10th section, and repeated 10 times.

[7] (ibid, p.188)

[8] (ibid, p.184)

5.2. Calculation of Test Errors and Fair Comparison Considerations. To calculate the test error estimate for the logistic regression model, the (custom) general_cv function was used. To calculate the test error estimate for the LDA model (given a specific split of the data into both training and test data), the (custom) logistic_reg_fold_error function (used for logistic regression) was modified. Once modified, it was passed an argument to the general_cv function. A similar procedure was adopted to calculate the test error for the QDA model. In order to ensure a fair comparison between the test error of the logistic regression model, the LDA model, and the QDA model, (1) the same (number of) predictors were used to fit all three models, and (2) the test errors were calculated using the same fold index. This ensures that the same partition of the data into folds was used in computing each test error. In comparing these models, whether the underlying assumptions of each are met plays a role in interpreting the test errors. While LDA has a lower variance than QDA and can, therefore, have improved predictive performance, if the assumption of common covariance is not met, the LDA model can highly biased.

5.3. Test Errors. The test error for the logistic regression classifier was 3.51% which is, as expected, higher than the training error for the same classifier which was 3.07%. This is to be expected as the training error is usually an overestimate, and thus, the test error is likely a better indicator of predictive performance. The confusion matrix (see Appendix: Figure 13(a)) shows that it made incorrect predictions for 10 samples which should have been classified as benign and 11 samples which should have been classified as malignant. The test error of the LDA classifier was 4.24% (higher than the training error which was 3.95%). The confusion matrix (see Appendix: Figure 13(b)) shows that the LDA classifier made incorrect predictions for 8 samples which should have been classified as benign and 19 samples which should have been classified as malignant. The test error of the QDA classifier was 4.83% (higher than the training error which was 4.53%). The confusion matrix (see Appendix: Figure 13(c)) showed that QDA made incorrect predictions for 25 samples which ought have been classified as benign and 6 samples which ought to have been classified as malignant. While the QDA classifier misclassified 6 out of 425 samples as malignant which appears minor, it also misclassified 25 observations out of 258 which should have been identified as malignant i.e. when it comes to classifying malignancy, the QDA classifier misclassified 10.7% of the observations. From the perspective of a pathologist or an oncologist, a 10.7% misclassification error is likely to be extremely unacceptable.

Given that the QDA classifier has the highest test error and a more than 10% misclassification rate in the malignancy class, it does not perform as well as the other two classifiers. The logistic regression classifier has the lowest test error at 3.5% indicating that where it classifies 100 samples, only 3.51% would be misclassified which is lower than the 4.34% the LDA classifier will misclassify.

6. Conclusion

This exercise utilised the BreastCancer dataset to build a classifier to predict whether, given a set of parameters, a sample of breast tissue would be benign or malignant. In building the classifiers, the best subset selection method highlighted that while almost all predictor variables were important to varying degrees, out of the 9 predictor variables, the cell thickness, and bare nuclei were among the more important predictors of malignancy while epithelial cell size and mitoses were among the least important. While three classifiers, namely, logistic, LDA, and QDA classifiers were modelled, the test errors of each of the classifiers indicate that the logistic classifier has the lowest error rate.

Appendix 1. Graphical and Tabular Outputs

Figure 1. Pairwise Scatterplot of BreastCancer Data

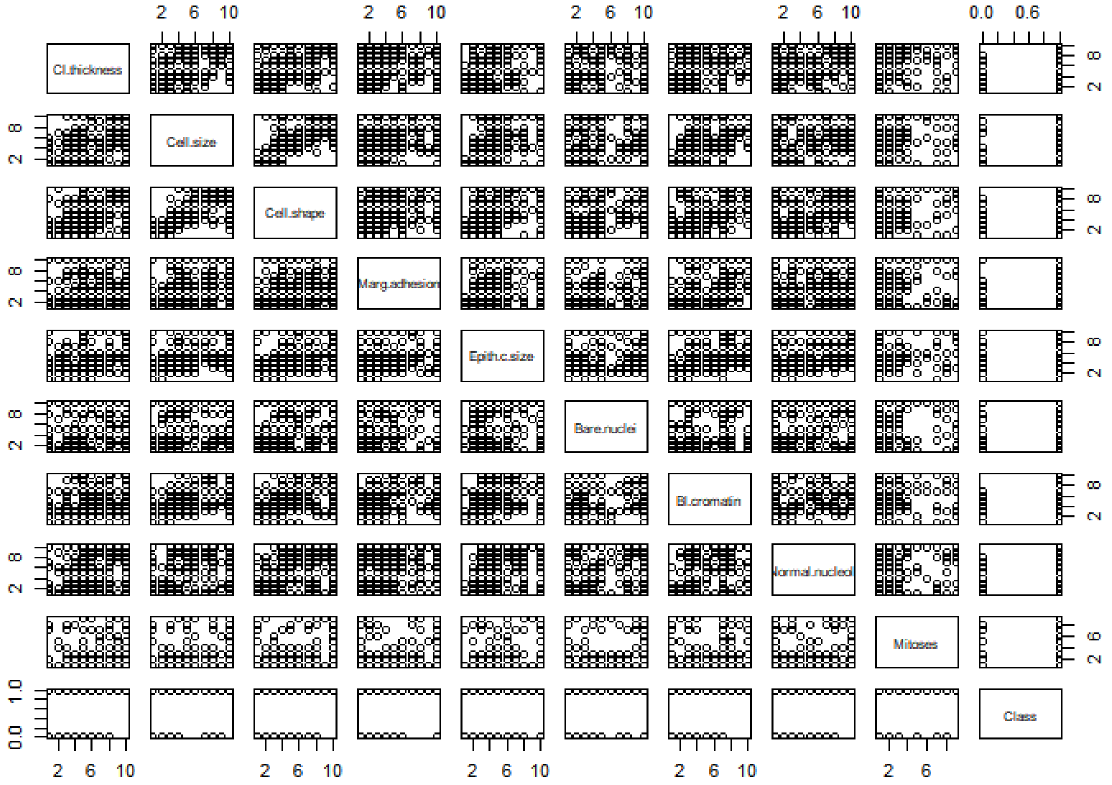


Figure 1. Pairwise Scatterplot of BreastCancer Data

Figure 2. Pairwise Scatterplot of Breast Cancer Data: Malignancy in Red

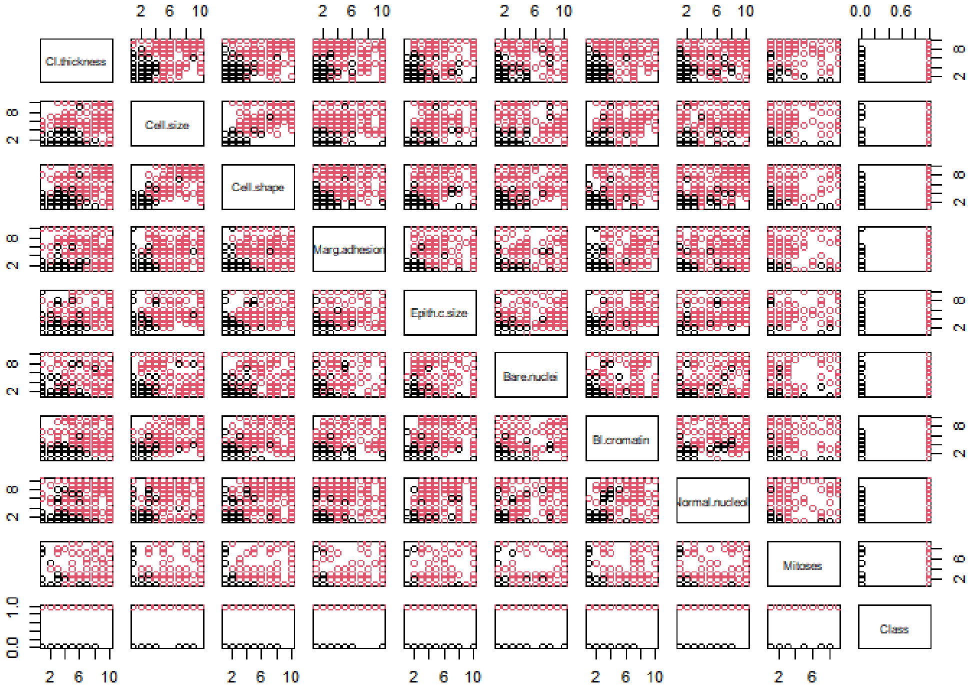


Figure 2. Pairwise Scatterplot of BreastCancer Data: Malignancy in Red

Figure 3. Pairwise Scatterplot of BreastCancer Data: Malignancy in Green Crosses, Benign samples in Red Triangles [included in Report]

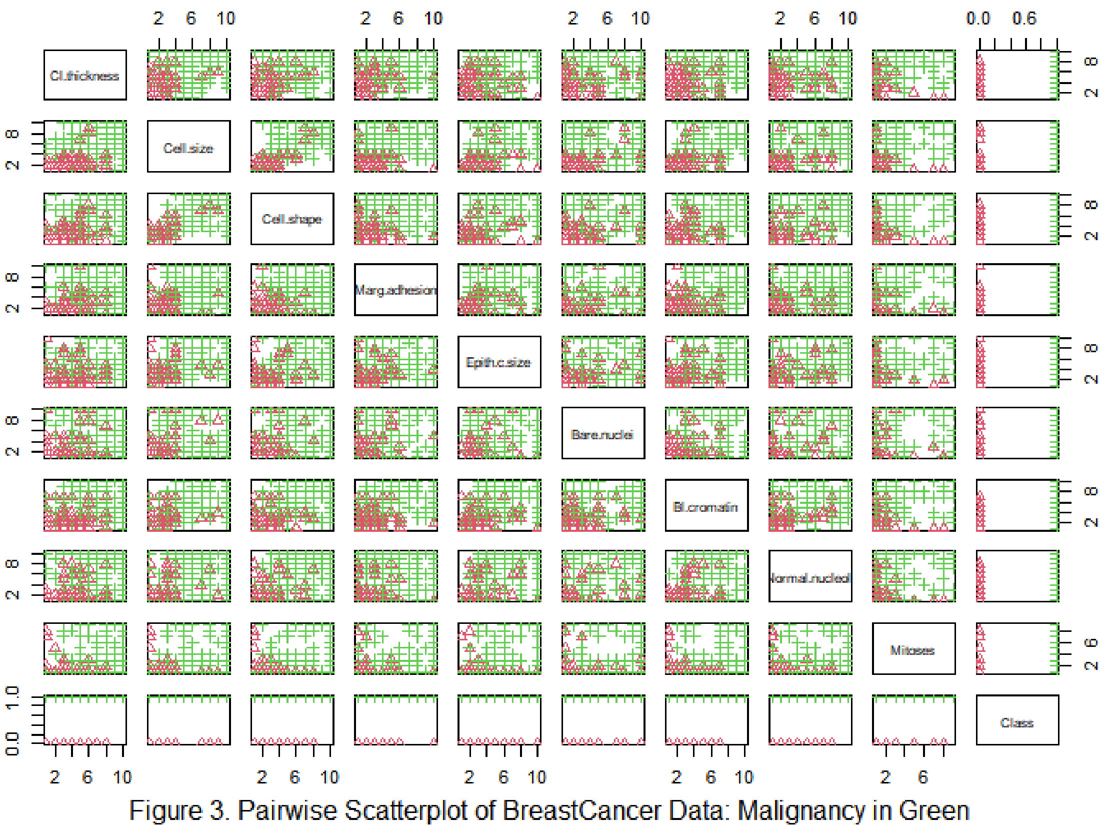


Figure 4. Correlation Panel: Correlation Coefficients of BreastCancer [included in the Report]

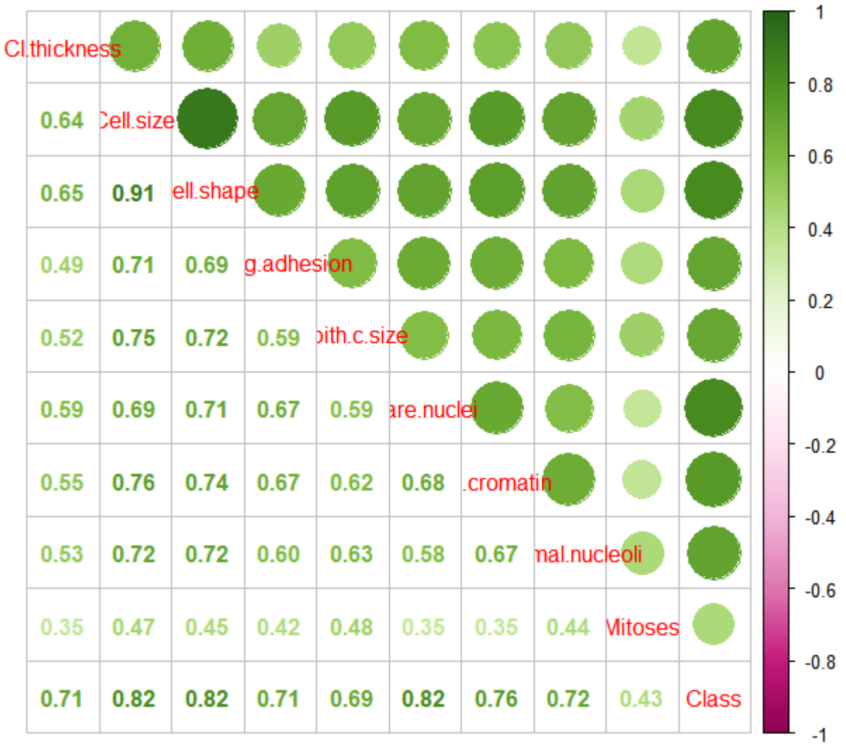


Figure 4.1. Variation Inflation Factor output (VIF)

```
Cl.thickness    Cell.size    Cell.shape  Marg.adhesion   Epith.c.size    Bare.nuclei
  1.905223      7.194043      6.549892      2.466854       2.550007       2.595167
Bl.cromatin     Normal.nucleoli   Mitoses
2.876814        2.430306            1.40059
```

Figure 5. Summary Output: Logistic Regression Full Model (all predictors)

```
Call:
glm(formula = Class ~ ., family = "binomial", data = BreastCancer2)

Deviance Residuals:
Min              1Q               Median           3Q               Max
-3.4855          -0.1152          -0.0619          0.0222           2.4702

Coefficients:
                 Estimate         Std. Error       z value          Pr(>|z|)
(Intercept)      -10.110096       1.173774          -8.613          < 2e-16
Cl.thickness     0.535256         0.141938         3.771            0.000163
Cell.size        -0.005943        0.209158         -0.028           0.977332
Cell.shape       0.322136         0.230644         1.397            0.162510
Marg.adhesion    0.330694         0.123462         2.679            0.007395
Epith.c.size     0.096797         0.156568         0.618            0.536415
Bare.nuclei      0.383015         0.093865         4.080            4.49e-05
Bl.cromatin      0.447401         0.171392         2.610            0.009044
Normal.nucleoli  0.213074         0.112894         1.887            0.059109
Mitoses          0.538551         0.325615         1.654            0.098138

(Intercept)      ***
Cl.thickness     ***
Cell.size
Cell.shape
Marg.adhesion    **
Epith.c.size
Bare.nuclei      ***
Bl.cromatin      **
Normal.nucleoli  .
Mitoses          .
---
Signif. codes:
0 ‘***’          0.001 ‘**’       0.01 ‘*’         0.05 ‘.’         0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

   Null deviance: 884.35          on 682  degrees of freedom
Residual deviance: 102.90         on 673  degrees of freedom
AIC: 122.9

Number of Fisher Scoring iterations: 8
```

Figure 6. Best Subsets: Akaike Information Criterion (AIC)

	Intercept	Cl.thickness	Cell.size	Cell.shape	Marg.adhesion	Epith.c.size	Bare.nuclei	Bl.cromatin	Normal.nucleoli	Mitoses	logLikelihood	**AIC**
0	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	-442.17509	884.3502
1	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	-127.37980	256.7596
2	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	-83.15598	170.3120
3	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	-67.77778	141.5556
4	TRUE	TRUE	FALSE	TRUE	FALSE	FALSE	TRUE	TRUE	FALSE	FALSE	-61.37155	130.7431
5	TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	FALSE	-56.13177	122.2635
6	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	FALSE	-53.57186	119.1437
7*	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	-51.63998	**117.2800**
8	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	-51.45031	118.9006
9	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	-51.44991	120.8998

Figure 7. Best Subsets: Bayesian Information Criterion (BIC)

	Intercept	Cl.thickness	Cell.size	Cell.shape	Marg.adhesion	Epith.c.size	Bare.nuclei	Bl.cromatin	Normal.nucleoli	Mitoses	logLikelihood	**BIC**
0	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	-442.17509	884.3502
1	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	-127.37980	261.2861
2	TRUE	FALSE	TRUE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	-83.15598	179.3649
3	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	FALSE	-67.77778	155.1351
4	TRUE	TRUE	FALSE	TRUE	FALSE	FALSE	TRUE	TRUE	FALSE	FALSE	-61.37155	148.8491
5*	TRUE	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	FALSE	-56.13177	**144.8960**
6	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	FALSE	-53.57186	146.3027
7	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	-51.63998	148.9654
8	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	-51.45031	155.1126
9	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	-51.44991	161.6383

Figure 8. Cross Validation Errors for each model (k=10)

0.34992679 0.07613470 0.04831625 0.03806735 0.04099561

0.03367496 0.03367496 0.03367496 0.03367496 0.03367496

Figure 9. Optimal Value of Predictors: AIC, BIC, and Cross Validation

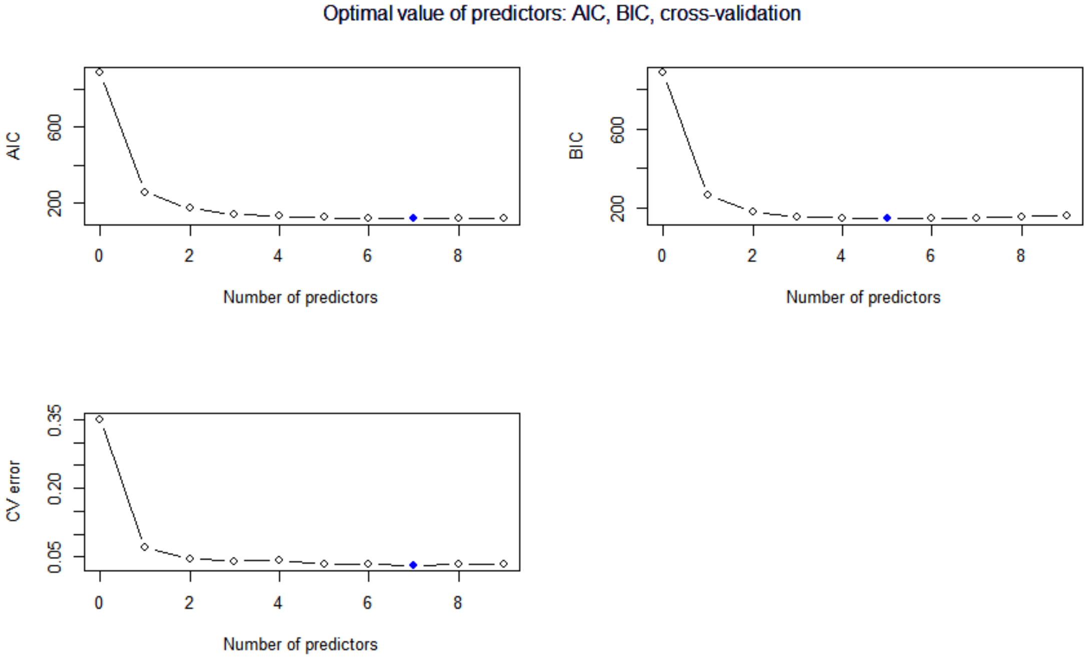


Figure 10. Output for Logistic Regression (7-predictor Model)

```
Call: glm(formula = Class ~ ., family = "binomial", data = BreastCancer3)

Deviance Residuals:
Min             1Q      Median  3Q      Max
-3.5252         -0.1148 -0.0627 0.0218  2.4118

Coefficients:
                Estimate        Std. Error      z value  Pr(>|z|)
(Intercept)     -9.98954        1.12478         -8.881   < 2e-16   ***
Cl.thickness    0.53425         0.14070         3.797    0.000146***
Cell.shape      0.34503         0.17162         2.010    0.044393 *
Marg.adhesion   0.34261         0.11923         2.873    0.004060 **
Bare.nuclei     0.38830         0.09359         4.149    3.34e-05 ***
Bl.cromatin     0.46222         0.16820         2.748    0.005997 **
Normal.nucleoli 0.22618         0.11099         2.038    0.041561 *
Mitoses         0.53536         0.32088         1.668    0.095237 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 884.35       on 682  degrees of freedom
Residual deviance: 103.28       on 675  degrees of freedom
AIC: 119.28

Number of Fisher Scoring iterations: 8
```

Figure 11. LDA Fitted Model output

```
Call:
lda(Class ~ ., data = BreastCancer3)

Prior probabilities of groups:
        0         1
0.6500732 0.3499268

Group means:
  Cl.thickness Cell.shape Marg.adhesion Bare.nuclei Bl.cromatin Normal.nucleoli  Mitoses
0     2.963964   1.414414      1.346847    1.346847    2.083333        1.261261 1.065315
1     7.188285   6.560669      5.585774    7.627615    5.974895        5.857741 2.543933

Coefficients of linear discriminants:
                       LD1
Cl.thickness           0.18903246
Cell.shape             0.18822671
Marg.adhesion          0.06279573
Bare.nuclei            0.25863173
Bl.cromatin            0.13464490
Normal.nucleoli        0.11896789
Mitoses                0.03097186
```

Figure 12. QDA Fitted Model output

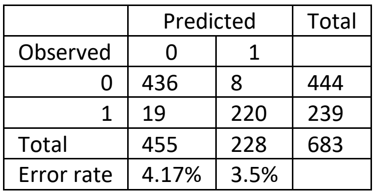
```
Call:
qda(Class ~ ., data = BreastCancer3)

Prior probabilities of groups:
        0         1
0.6500732 0.3499268

Group means:
  Cl.thickness Cell.shape Marg.adhesion Bare.nuclei Bl.cromatin Normal.nucleoli  Mitoses
0     2.963964   1.414414      1.346847    1.346847    2.083333        1.261261 1.065315
1     7.188285   6.560669      5.585774    7.627615    5.974895        5.857741 2.543933
```

Figure 13. Confusion Matrices

Figure 13(a). Confusion Matrix of Logistic Regression Model

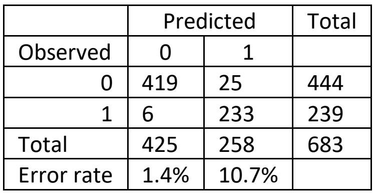
	Predicted		Total
Observed	0	1	
0	434	10	444
1	11	228	239
Total	445	238	683
Error rate	2.47%	4.2%	

Figure 13(b). Confusion Matrix of LDA Model



	Predicted		Total
Observed	0	1	
0	436	8	444
1	19	220	239
Total	455	228	683
Error rate	4.17%	3.5%	

Figure 13(c). Confusion Matrix of QDA Model



	Predicted		Total
Observed	0	1	
0	419	25	444
1	6	233	239
Total	425	258	683
Error rate	1.4%	10.7%	

Appendix 2. R Code

```
library(mlbench)
library(corrplot)
library(car)
library(bestglm)
library(MASS)

data(BreastCancer)
?BreastCancer
dim(BreastCancer)
head(BreastCancer)
str(BreastCancer)

#PART 1: DATA CLEANING
#all data are in factors; conversion to quantitative variables/numeric
for(i in 1:10) {
  BreastCancer[,i] <- as.numeric(BreastCancer[,i])
}

levels(BreastCancer$Class)
#BreastCancer£Class categorical variable with two levels (benign,malignant)
#convert categorical variables to numeric using dummy variables: 0,1
BreastCancer$Class <- ifelse(BreastCancer$Class == "malignant", 1, 0)
head(BreastCancer) #check that class is 0,1

#verify that all columns are numeric
str(BreastCancer)
unlist(lapply(BreastCancer, class))

#check for missing values
sum(is.na(BreastCancer))
colSums(is.na(BreastCancer))
BreastCancer1 <- na.omit(BreastCancer) #remove all rows with missing values

#verify that all rows with missing values are removed
sum(is.na(BreastCancer1))
colSums(is.na(BreastCancer1))
nrow(BreastCancer1) #rows reduced to 683 (699-16)

#remove ID column
BreastCancer2 <-BreastCancer1 [,-1]
head(BreastCancer2) #verify that ID column was removed
ncol(BreastCancer2) #columns reduced to 10 (from 11)

#PART 2: EXPLORATORY DATA ANALYSIS
#check how many cases of malignancy and benign
table(BreastCancer2$Class)

#pairwise scatterplot
pairs(BreastCancer2) #basic black and white pairs plot
mtext("Figure 1. Pairwise Scatterplot of BreastCancer Data", side = 3, line = -27.5, outer =
TRUE)

pairs(BreastCancer2[,1:10], col=BreastCancer2[,10]+1)
```

```
mtext("Figure 2. Pairwise Scatterplot of BreastCancer Data: Malignancy in Red", side = 3, line = -27.5, outer = TRUE)

pairs(BreastCancer2[,1:10], col=BreastCancer2[,10]+10, pch=BreastCancer2[,10]+2)
mtext("Figure 3. Pairwise Scatterplot of BreastCancer Data: Malignancy in Green", side = 3, line = -27.5, outer = TRUE)

#correlation panel

#add correlation coefficients through a correlation panel
panel.cor <- function(x, y){
  usr <- par("usr"); on.exit(par(usr))
  par(usr = c(0, 1, 0, 1))
  r <- round(cor(x, y), digits=2)
  txt <- paste0("r = ", r)
  cex.cor <- 0.8/strwidth(txt)
  text(0.5, 0.5, txt, cex = cex.cor * r)
}
#customize upper panel
upper.panel<-function(x, y){
  points(x,y, pch = 19)
}
#create the plots
pairs(BreastCancer2,
      lower.panel = panel.cor,
      upper.panel = upper.panel)

#check correlation between variables
corrplot(cor(BreastCancer2), method = "number", type = "upper", diag = FALSE, col = COL2('PiYG'))
corrplot.mixed(cor(BreastCancer2), upper = "circle", lower = "number", number.cex = 1.0, upper.col= COL2("PiYG"), lower.col = COL2("PiYG"))

mtext("Figure 4. Correlation Panel: Correlation Coefficients", side = 3, line = -27.5, outer = TRUE)

model <- lm(Class ~., data=BreastCancer2)
vif(model)

#PART 3: MODELLING
##3.1.LOGISTIC REGRESSION

#store rows and columns
n=nrow(BreastCancer2)
p=ncol(BreastCancer2)-1

#logistic regression on all variables
BClr_fit <- glm(Class~., data=BreastCancer2, family="binomial")
summary(BClr_fit)

#best subset selection
head(BreastCancer2) #verify that response variable is the last column

#best subset selection: AIC
BCbss_AIC <- bestglm(BreastCancer2, family=binomial, IC="AIC")
```

```
best_AIC <- BCbss_AIC$ModelReport$Bestk
best_AIC #7
BCbss_AIC$Subsets #best subsets AIC
#best subset selection: BIC
BCbss_BIC <- bestglm(BreastCancer2, family=binomial, IC="BIC")
best_BIC <- BCbss_BIC$ModelReport$Bestk
best_BIC #5
BCbss_BIC$Subsets #best subsets BIC

#best subset selection: k-fold cross validation
## Sample the fold-assignment index
nfolds = 10
fold_index = sample(nfolds, n, replace=TRUE)

logistic_reg_fold_error = function(X, y, test_data) {
  Xy = data.frame(X, y=y)
  if(ncol(Xy)>1) tmp_fit = glm(y ~ ., data=Xy[!test_data,], family="binomial")
  else tmp_fit = glm(y ~ 1, data=Xy[!test_data,,drop=FALSE], family="binomial")
  phat = predict(tmp_fit, Xy[test_data,,drop=FALSE], type="response")
  yhat = ifelse(phat > 0.5, 1, 0)
  yobs = y[test_data]
  test_error = 1 - mean(yobs == yhat)
  return(test_error)
}

general_cv = function(X, y, fold_ind, fold_error_function) {
  p = ncol(X)
  Xy = cbind(X, y=y)
  nfolds = max(fold_ind)
  if(!all.equal(sort(unique(fold_ind)), 1:nfolds)) stop("Invalid fold partition.")
  fold_errors = numeric(nfolds)
  # Compute the test error for each fold
  for(fold in 1:nfolds) {
    fold_errors[fold] = fold_error_function(X, y, fold_ind==fold)
  }
  # Find the fold sizes
  fold_sizes = numeric(nfolds)
  for(fold in 1:nfolds) fold_sizes[fold] = length(which(fold_ind==fold))
  # Compute the average test error across folds
  test_error = weighted.mean(fold_errors, w=fold_sizes)
  # Return the test error
  return(test_error)
}
logistic_reg_bss_cv = function(X, y, fold_ind) {
  p = ncol(X)
  Xy = data.frame(X, y=y)
  X = as.matrix(X)
  nfolds = max(fold_ind)
  if(!all.equal(sort(unique(fold_ind)), 1:nfolds)) stop("Invalid fold partition.")
  fold_errors = matrix(NA, nfolds, p+1) # p+1 because M_0 included in the comparison
  for(fold in 1:nfolds) {
    # Using all *but* the fold as training data, find the best-fitting models
    # with 0, 1, ..., p predictors, i.e. identify the predictors in M_0, M_1, ..., M_p
    tmp_fit = bestglm(Xy[fold_ind!=fold,], family=binomial, IC="AIC")
    best_models = as.matrix(tmp_fit$Subsets[,2:(1+p)])
    # Using the fold as test data, find the test error associated with each of
    # M_0, M_1,..., M_p
    for(k in 1:(p+1)) {
      fold_errors[fold, k] = logistic_reg_fold_error(X[,best_models[k,]], y, fold_ind==fold)
    }
  }
  # Find the fold sizes
  fold_sizes = numeric(nfolds)
  for(fold in 1:nfolds) fold_sizes[fold] = length(which(fold_ind==fold))
  # For models with 0, 1, ..., p predictors compute the average test error across folds
  test_errors = numeric(p+1)
  for(k in 1:(p+1)) {
    test_errors[k] = weighted.mean(fold_errors[,k], w=fold_sizes)
  }
  # Return the test error for models with 0, 1, ..., p predictors
  return(test_errors)
}

#abest subset selection: cross-validation
BCbss_cve = logistic_reg_bss_cv(BreastCancer2[,1:p], BreastCancer2[,p+1], fold_index)
BCbss_cve
#Identify the number of predictors in the model which minimises test error
best_BCcve = which.min(BCbss_cve - 1) #7
best_BCcve

#visualise bss: AIC, BIC, cross-validation
## Create multi-panel plotting device
par(mfrow=c(2, 2))
## Produce plots, highlighting optimal value of k
plot(0:p, BCbss_AIC$Subsets$AIC, xlab="Number of predictors", ylab="AIC", type="b")
points(best_AIC, BCbss_AIC$Subsets$AIC[best_AIC+1], col="blue", pch=16)
plot(0:p, BCbss_BIC$Subsets$BIC, xlab="Number of predictors", ylab="BIC", type="b")
points(best_BIC, BCbss_BIC$Subsets$BIC[best_BIC+1], col="blue", pch=16)
plot(0:p, BCbss_cve, xlab="Number of predictors", ylab="CV error", type="b")
points(best_BCcve, BCbss_cve[best_BCcve+1], col="blue", pch=16)

pstar = 7 #define no.of predictors
#check which predictors are in the 7-predictor model
BCbss_AIC$Subsets[pstar+1,]
BCbss_BIC$Subsets[pstar+1,]

#construct a reduced data set containing only the 7 selected predictors
BCbss_AIC$Subsets[pstar+1, 2:(p+1)]
indices = which(BCbss_AIC$Subsets[pstar+1, 2:(p+1)]==TRUE)
BreastCancer3 <- BreastCancer2[,c(indices, p+1)]
head(BreastCancer3)
dim(BreastCancer3)

#obtain regression coefficients for 7-predictor model
BClogreg_fit = glm(Class ~ ., data=BreastCancer3, family="binomial")
summary(BClogreg_fit) #shows that all variables are somewhat useful

#building Bayes classifier for LDA
#Apply  LDA
BClda_fit <- lda(Class~., data=BreastCancer3)
```

```
BClda_fit
plot(lda_fit1)

#Building Bayes classifier for QDA
#Apply  QDA to 7-predictor model
BCqda_fit <- qda(Class~., data=BreastCancer3)
BCqda_fit

#perform predictions
##data frame of predictor variables
bc <-
data.frame(Cl.thickness=3,Cell.shape=4,Marg.adhesion=5,Bare.nuclei=2,Bl.cromatin=3,Normal
.nucleoli=4,Mitoses=1)

#predict using logistic regression model
p_lg <- predict(BClogreg_fit,bc,type="response")
y_lg <- as.numeric(ifelse(p_lg>0.5,1,0))
p_lg
y_lg #0

#predict using LDA model
p_lda <- predict(BClda_fit,bc,type="response")
p_lda #0

#predict using QDA model
p_qda <- predict(BCqda_fit,bc,type="response")
p_qda #1

#PART 4: MODEL COMPARISON

#training and test errors of logistic regression model
BClg_phat <-predict(BClogreg_fit, BreastCancer3, type="response")
BClg_yhat <- as.numeric(ifelse(BClg_phat > 0.5,1,0))
#confusion matrix
confusion_lg <- table(Observed=BreastCancer3$Class, Predicted=BClg_yhat)
confusion_lg
#calculate training error
lgtraining_error <- 1 - sum(diag(confusion_lg)/sum(confusion_lg))
lgtraining_error #0.03074671
#or
1 - mean(BreastCancer3$Class == BClg_yhat)
#calculate test error
lgtest_error = general_cv(BreastCancer3[,1:pstar], BreastCancer3[,pstar+1], fold_index,
logistic_reg_fold_error)
lgtest_error #0.03513909

#training and test errors of LDA model
BClda_predict = predict(BClda_fit, BreastCancer3)
BClda_yhat = BClda_predict$class
#confusion matrix
confusion_lda <- table(Observed=BreastCancer3$Class,Predicted=BClda_yhat)
#calculate training error
1 - mean(BreastCancer3$Class == BClda_yhat) #0.03953148
#to calculate test error: cross-validation
lda_fold_error = function(X, y, test_data) {
  Xy = data.frame(X, y=y)
  if(ncol(Xy)>1) tmp_fit = lda(y ~ ., data=Xy[!test_data,])
  tmp_predict = predict(tmp_fit, Xy[test_data,])
  yhat = tmp_predict$class
  yobs = y[test_data]
  test_error = 1 - mean(yobs == yhat)
  return(test_error)
}
#calculate test error of LDA model
ldatest_error = general_cv(BreastCancer3[,1:pstar], BreastCancer3[,pstar+1], fold_index,
lda_fold_error)
ldatest_error #0.04245974

#training and test errors of QDA model
BCqda_predict = predict(BCqda_fit, BreastCancer3)
BCqda_yhat = BCqda_predict$class
## Calculate confusion matrix:
(confusion_qda = table(Observed=BreastCancer3$Class, Predicted=BCqda_yhat))
#calculate training error of QDA model
1 - mean(BreastCancer3$Class == BCqda_yhat) #0.04538799
#to calculate test error of QDA model: cross-validation
qda_fold_error = function(X, y, test_data) {
  Xy = data.frame(X, y=y)
  if(ncol(Xy)>1) tmp_fit = qda(y ~ ., data=Xy[!test_data,])
  tmp_predict = predict(tmp_fit, Xy[test_data,])
  yhat = tmp_predict$class
  yobs = y[test_data]
  test_error = 1 - mean(yobs == yhat)
  return(test_error)
}

qdatest_error = general_cv(BreastCancer3[,1:pstar], BreastCancer3[,pstar+1], fold_index,
qda_fold_error)
qdatest_error #0.04831625
```