Assignment 2 (Yan Lin)

1 Introduction

I will analyze the BreastCancer dataset, which involves the characterization of breast tissue samples collected from 699 women in Wisconsin using fine needle aspiration cytology (FNAC). A raw data frame contained 699 observations with 11 variables, including 1 character variable, 9 ordered or nominal variables, and a target class. This original dataset will be cleaned by me and then transformed into a new dataset, MyCancer, which can be used for exploratory studies and modelling.

First, I will fit a logistic regression model for the full model (9 predictor variables), followed by the calculation of the in-sample training error and the test error under out-of-sample k-fold cross-validation for the full model. Then, the best subset of the logistic regression model is then selected based on a combination of the three information criteria, AIC, BIC and k-fold cross-validation, and the training and testing errors for the best subset are calculated. Multiple selected predictor variables (up to 9) are used to create LDA and QDA Bayesian models, the mean of each group is estimated, their training and testing errors are calculated, the three models are compared, and conclusions are drawn. In this work, we primarily used the 10-fold cross-validation method to calculate the test error for each model and to perform the third criterion for the best subset selection. The 10-fold cross-validation method is more stable and less volatile than the validation set method.

2.1 Cleaning data

Assume that patients can be considered as a random sample from the population of women presenting with breast cancer symptoms. We could deduce that this BreastCancer dataset contains 699 rows and 11 columns, as well as summarise it. The data in columns 2 to 11 are all factorial rather than numerical and cannot be mathematically calculated or modelled, as determined by the class or is.factor functions.

Before beginning analysis, we must clean the data after loading the raw dataset. This data set contains some missing observations on predictors, encoded as NA. When we print 24th row of the BreastCancer dataset, we observe that the seventh number (i.e., the Bare.nuclei column) is NA. Additionally, we can utilise the is.na function to determine if a value is NA (true for yes, false for no). After that, I use table(is.na(BreastCancer)) to calculate how many NA values there are in the whole dataset (16 values for True). We can also use which(is.na(BreastCancer),arr.ind = TRUE) to find the coordinates of these specific 16 missing values. So, we must delete the rows where these 16 values are located.

```
table(is.na(BreastCancer)) # The full codes are in the Appendix 1
## FALSE  TRUE
## 7673    16
```

We can find that the 16 missing values NA are actually all in column 7, just in different rows. The next step is use na.omit(BreastCancer) function to delete all rows containing NA and name the new dataset MyClean. Whilst, we then use table(is.na(MyClean)) to check if there is still a true value, and if the true value is 0, then the new dataset no longer contains NA.

```
MyClean=na.omit(BreastCancer) #Delete rows containing NA
table(is.na(MyClean)) #check whether the new data set still contains NA
## FALSE
## 7513
```

The column labelled 'Id' is a record of this patient's sample code number, whereas the columns labelled 'Cl.thickness' to 'Mitoses' are calculated from digitised images of fine needle aspiration (FNA) of the breast lump and describe the characteristics of the nuclei present on the images, with values ranging from

1 to 10. The column labelled 'Class' is a factor with two levels - benign and malignant - showing the properties of the breast cancer in that patient.

We are interested in determining whether the nine sliced cell features (or fewer predictor variables) in this sample dataset may be used to diagnose benign and malignant breast cancer. In this illustration, I will disregard the 'Id' column and use the 'Class' as the response variable. Note that Class is stored as a factor, where benign is the first level and malignant the second.

```
typeof(MyClean$Class)
## [1] "integer"
head(as.integer(MyClean$Class))
## [1] 1 1 1 1 1 2
```

And so benign is represented internally as a 1 and malignant as a 2. When working with logistic regression, I prefer to adopt the more standard 0/1 numerical labels. To do this, whilst also omitting the Id column, I can create a new data frame and convert the first nine cytological characteristics (predictor variables) from factorial to numerical. Finally, I use the str and head functions to check the newly generated MyCancer data frame.

```
MyCancer = data.frame(MyClean[,-c(1,11)] ,Class=as.integer(MyClean$Class)-1)
MyCancer$Cl.thickness=as.numeric(MyCancer$Cl.thickness)
# The full codes are in the Appendix 1
```

```
> str(MyCancer)
'data.frame':   683 obs. of  10 variables:
 $ Cl.thickness   : num  5 5 3 6 4 8 1 2 2 4 ...
 $ Cell.size      : num  1 4 1 8 1 10 1 1 1 2 ...
 $ Cell.shape     : num  1 4 1 8 1 10 1 2 1 1 ...
 $ Marg.adhesion  : num  1 5 1 1 3 8 1 1 1 1 ...
 $ Epith.c.size   : num  2 7 2 3 2 7 2 2 2 2 ...
 $ Bare.nuclei    : num  1 10 2 4 1 10 10 1 1 1 ...
 $ Bl.cromatin    : num  3 3 3 3 3 9 3 3 1 2 ...
 $ Normal.nucleoli: num  1 2 1 7 1 7 1 1 1 1 ...
 $ Mitoses        : num  1 1 1 1 1 1 1 1 5 1 ...
 $ Class          : num  0 0 0 0 0 1 0 0 0 0 ...
```

Figure 1: Check that the nine cytological characteristics (predictor variables) are already numerical

2.2 Exploratory data analysis

Let's explain each column of the new data set. In the clump thickness, benign cells are typically arranged in monolayers, whereas malignant cells are frequently arranged in multilayers. (Column 1) In contrast to the Uniformity of cell size/shape, the size and shape of cancer cells tend to fluctuate. Therefore, these factors are useful for evaluating whether or not the cells are malignant. (Column 2&3) In marginal adhesion, normal cells tend to adhere to one another, whereas malignant cells lose this ability. Therefore, lack of adhesion is an indication of cancer. (Column 4) The size of a single epithelial cell is connected to the aforementioned uniformity. Significantly expanded epithelial cells may be cancerous cells. (Column 5) The term bare nuclei describe nuclei that are not surrounded by cytoplasm (the rest of the cell). Typically observed in benign tumours. (Column 6) The term Bland Chromatin refers to the uniform "texture" of benign cell nuclei. In cancer cells, chromatin is typically coarser. (Column 7) The Normal nucleoli are tiny structures that can be observed within the nucleus. If visible at all, the nucleolus is typically quite tiny in normal cells. In cancer cells, the nucleoli grow more conspicuous and there may be an increase in their number. (Column 8) Lastly, Mitosis is nuclear division plus cytokines and results in the production of two identical daughter cells during prophase. It is the process of cell division and replication. Pathologists can detect the severity of cancer by counting mitoses. (Column 9)

MyCancer dataset	Columns	Variables	Values
Predictor variables	1	Clump Thickness	1–10
	2	Uniformity of Cell Size	
	3	Uniformity of Cell Shape	
	4	Marginal Adhesion	
	5	Single Epithelial Cell Size	
	6	Bare Nuclei	
	7	Bland Chromatin	
	8	Normal Nucleoli	
	9	Mitoses	
Response variable	10	Class	0 or 1

Table 1: Number of columns and values of predictor and response variables

Now that we are clear in our labelling of the response variable, we can perform some exploratory data analysis. We notice that after entering table(MyCancer$Class), the number of classes with values of 0 and 1 is counted.

```
##  0   1
## 444 239
```

There are 444 benign and 239 malignant. Therefore, breast cancers are more likely to be benign than malignant. There are a variety of graphical methods for examining the data. For instance, we may comprehend the relationship between predictor variables by making paired plots of nine predictor variables, as well as the relationship between response variables and nine predictor variables by colouring the points based on whether breast cancer is benign or malignant.

This produces the Figure 2. Observing that the majority of the black colour is centred in the lower left corner indicates that the smaller the nine cytological characteristics, the more probable they are to be benign. Whist, we can observe that, Cell.size, Cell.shape and Bare.nuclei all have a strong positive relationship with Class. The relationship between Mitoses and Class is the weakest. As for the relationship between the predictor variables, there is a strong positive linear relationship between Cell.size and Cell.shape. There was also a relatively strong positive relationship between Bare.nuclei and the first two predictor variables, which again suggests that the size of single epithelial cells is related to the aforementioned homogeneity of cell size and shape.

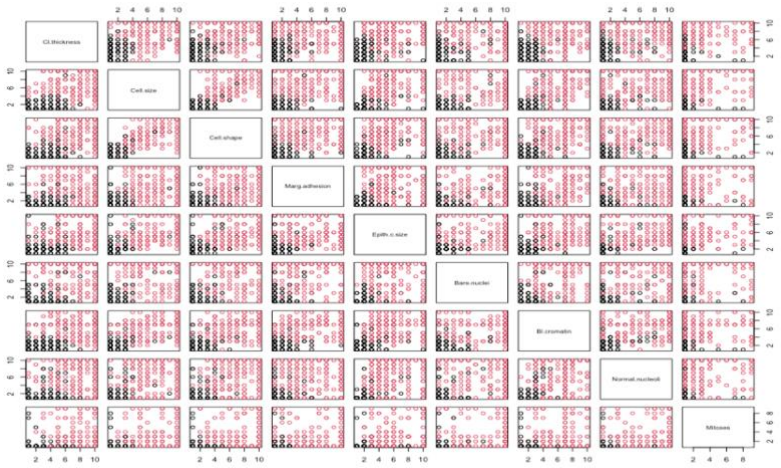

Figure 2: Scatterplot matrix for the MyCancer data. Class when the breast cancer is benign/malignant appear in black / red

Next, print out their correlation coefficients:

```
> cor(MyCancer)
               Cl.thickness Cell.size Cell.shape Marg.adhesion Epith.c.size Bare.nuclei
Cl.thickness      1.0000000 0.6424815  0.6534700     0.4878287    0.5235960   0.5930914
Cell.size         0.6424815 1.0000000  0.9072282     0.7069770    0.7535440   0.6917088
Cell.shape        0.6534700 0.9072282  1.0000000     0.6859481    0.7224624   0.7138775
Marg.adhesion     0.4878287 0.7069770  0.6859481     1.0000000    0.5945478   0.6706483
Epith.c.size      0.5235960 0.7535440  0.7224624     0.5945478    1.0000000   0.5857161
Bare.nuclei       0.5930914 0.6917088  0.7138775     0.6706483    0.5857161   1.0000000
Bl.cromatin       0.5537424 0.7555592  0.7353435     0.6685671    0.6181279   0.6806149
Normal.nucleoli   0.5340659 0.7193460  0.7179634     0.6031211    0.6289264   0.5842802
Mitoses           0.3545301 0.4654091  0.4468571     0.4249917    0.4811836   0.3490108
Class             0.7147899 0.8208014  0.8218909     0.7062941    0.6909582   0.8226959
                Bl.cromatin Normal.nucleoli   Mitoses     Class
Cl.thickness      0.5537424       0.5340659 0.3545301 0.7147899
Cell.size         0.7555592       0.7193460 0.4654091 0.8208014
Cell.shape        0.7353435       0.7179634 0.4468571 0.8218909
Marg.adhesion     0.6685671       0.6031211 0.4249917 0.7062941
Epith.c.size      0.6181279       0.6289264 0.4811836 0.6909582
Bare.nuclei       0.6806149       0.5842802 0.3490108 0.8226959
Bl.cromatin       1.0000000       0.6656015 0.3536683 0.7582276
Normal.nucleoli   0.6656015       1.0000000 0.4370424 0.7186772
Mitoses           0.3536683       0.4370424 1.0000000 0.4312971
Class             0.7582276       0.7186772 0.4312971 1.0000000
```

Figure 3: Correlation Matrix for the MyCancer data

We can see that, Cell.size and Cell.shape have high linear correlations with the response variable Class, with correlations of 0.820801 and 0.8218909, respectively. Bare.nuclei has the strongest linear correlation with Class, with a correlation of 0.8226959. Mitoses has the poorest linear correlation with Class, with a correlation of 0.4312971, indicating an extremely weak positive linear relationship. In contrast, correlations in Epith.c.size and Marg.adhesion indicated a moderately favourable linear relationship. The pair of predictor variables with the strongest linear relationship were Cell.size and Cell.shape, which had a correlation of 0.9072282, indicating a strong positive linear relationship. It is consistent with the conclusions we have drawn from observing Figure1. This suggests that when we fit the model, we can omit some of the predictor variables. Because cell size and cell shape are highly correlated, using both as predictor variables would result in a high variance, so it is unlikely that any regression model would require both sets as predictor variables.

2.3 Logistic regression with best subset selection

To find a function to model p(x) such that the output of the function is between 0 and 1 for any value of X, we choose the logistic regression model. The logistics model thus produces an S-shaped curve and always yields a reasonable prediction, regardless of the value of X. The logistic regression model can be considered as a linear model of X under a logarithmic transformation. The coefficients of the fitted logistic regression model are estimated using the great likelihood method.

2.3.1 The full model

We first fit a logistic regression model to all predictor variables and determine whether the best subset selection is required based on the fit and test error. Because we have set Class as the response variable (binary 0-1), the response variable is satisfied as binary. The MyCancer dataset was cleaned from the BreastCancer dataset, which was obtained from a random sample of breast cancer diagnoses in Wisconsin, and therefore satisfies the observation value independence.

Next, there are 683 values for MyCancer$Class, the sample size is sufficiently large to be satisfied. Therefore, we simply make the following **three assumptions**: 1. no multicollinearity between the explanatory variables; 2. no severe outliers; 3.the relationship between the explanatory variables and the Logit of the response variable is linear.

We start by recording the number of rows in our data frame and the number of predictor variables(n=nrow(MyCancer), p=ncol(MyCancer)-1). Next, we will fit a logistic regression model for Class in terms of the predictors Cl.thickness through Mitoses. Then, we can then summarise the fit of the model using the summary function.

```
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 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 4: Summary of the full logistic regression model

We can derive the intercept and nine coefficient values for the fit:

$$\beta_0 = \text{-10.110096},\ \beta_1 = 0.535256, \beta_2 = \text{-0.005943},\ \beta_3 = 0.322136, \beta_4 = 0.330694,$$
$$\beta_5 = 0.096797,\ \beta_6 = 0.383015, \beta_7 = 0.447401, \beta_8 = 0.213074\ , \beta_9 = 0.538551$$

Only cell size was negatively correlated with Class, the other eight predictor variables were all positively correlated with Class, this is because cell size and cell shape are highly correlated and using them together would result in high variance, therefore these two sets of data could not be used together in any model and **assumption 1** could not be met. And we note that only four variables have p-values less than 0.05, i.e., significantly different from the original hypothesis of a coefficient of zero. The other five variables have large p-values. the large p-values mean that they contribute little to a model that includes all other predictors. We can also calculate a training error of 0.03074671 for the full model and a testing error of 0.03367496 for the 10-folds cross-validation.

Unnecessary predictor variables can widen the variance of parameter estimates, which can have a negative impact on prediction performance. To reduce testing error, we perform best subset selection in logistic regression.

2.3.2 Best subset selection in logistic regression model

An important step in breast cancer diagnostic modelling is feature extraction. The optimal feature set should have valid and discriminative features, while primarily reducing redundancy in the feature space to avoid the "dimensional catastrophe" problem.

From 2.4.1 we see that the full model contains more predictors than it needs. This suggests that we may be able to use a subset selection approach to produce a model with fewer predictors than p and have better predictive performance. We only consider the best subset selection. We use negative log likelihoods rather than sums of squares of residuals. When comparing a model containing 0, 1, …., p predictors, various model comparison criteria can be used. Widely used examples are AIC (equivalent to Mallow's Cp in multiple linear regression), BIC and cross-validation(10-folds) test errors. In all three cases, a small value of the information criterion indicates a 'better' model.

```
> bss_fit_AIC$Subsets
   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 5: Best subset selection when the information criterion used is AIC

The AIC information criterion tells us that 7 predictor variables should be selected (1st, 3rd, 4th, 6th, 7th, 8th, 9th), whilst the BIC information criterion tells us that 5 predictor variables should be selected (1st, 4th, 6th, 7th, 8th).

```
> bss_fit_BIC$Subsets
   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 6: Best subset selection when the information criterion used is BIC

We chose to do 10-fold cross-validation by setting set.seed(5) and then wrote the logistic_reg_bss_cv function. In the first step, the test error for each fold is calculated for the different predictor variable models, in the second step, the size of the fold is found, in the third step, the average test error for each fold is calculated for the different predictor variable models, and in the fourth step, the test error is returned for the different predictor variable models. Identify the number of predictors in the model which minimises test error: `best_cv = which.min(cv_errors) - 1`. Draw a graph of the three information criteria chosen for the best subset comparison:

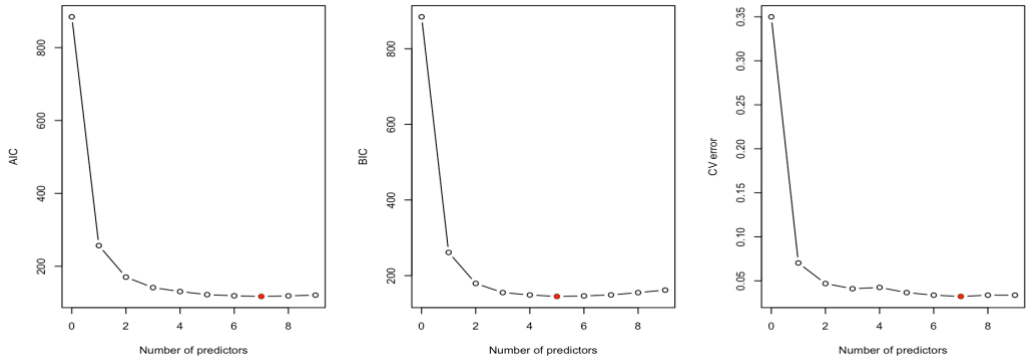

Figure 7: Selection of the best subset under three information criteria

Clearly 10-fold cross-validation for best subset selection would be a better decision than AIC and BIC. This is because the 10-fold cross-validation is a model that has tested each of the different predictor variables, and the 10-fold cross-validation does not have a high bias or a high variance.

```
> ## Construct a reduced data set containing only the 7 selected predictors
> bss_fit_AIC$Subsets[pstar+1, 2:(p+1)]
   Cl.thickness Cell.size Cell.shape Marg.adhesion Epith.c.size Bare.nuclei Bl.cromatin Normal.nucleoli Mitoses
7*         TRUE     FALSE       TRUE          TRUE        FALSE        TRUE        TRUE             TRUE    TRUE
> indices = which(bss_fit_AIC$Subsets[pstar+1, 2:(p+1)]==TRUE)
> indices
[1] 1 3 4 6 7 8 9
```

Figure 8: Best subset of 7 predictor variables taken

Since the best subset is taken from the same data set, we still make the same 3 assumptions: 1. no multicollinearity between the explanatory variables; 2. no severe outliers; 3.the relationship between the explanatory variables and the Logit of the response variable is linear.

```
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
```

Figure 9: Summarize the coefficients of the best subset of models (7 variables) in logistic regression

We can derive the intercept and seven coefficient values for the fit:

$$\beta_0 = \text{-9.98954},\ \beta_1 = 0.53425, \beta_2 = 0.34503,\ \beta_3 = 0.34261$$
$$\beta_4 = 0.38830,\ \beta_5 = 0.46222,\ \beta_6 = 0.22618,\ \beta_7 = 0.53536$$

Consistent with the Figure 2 and Figure 3, all coefficients, except for the intercept, have a positive correlation with the response variable Class. None of the p-values are extremely large, showing that each predictor variable is essential. The fitted model is consistent with three assumptions.

In the next step, we still take 10-fold cross-validation to calculate the test error of the logistic regression model with 7 predictor variables:

```
test_error_red = general_cv(MyCancer_red[,1:pstar], MyCancer_red[,pstar+1], fold_index, logistic_reg_fold_error)
```

We get a test error of 0.03221 < 0.03367 (full-model test error). It shows that the model with seven predictor variables fits better and fits more data correctly than the full model.

2.5 LDA Bayesian classifier and QDA Bayesian classifier

In this MyCancer dataset, p=7 and K=2, so we must consider not only logistic regression, but also LDA and QDA. Logistic regression models the conditional distribution of Y given X, whereas LDA and QDA model the conditional distribution of X given the prior probabilities of Y, and then apply the Bayes' principle to find the conditional probabilities of Y given X.

2.5.1 LDA Bayesian classifier

When p=1, K=2, LDA approximates logistic regression. However, the coefficients in logistic regression are calculated by the maximum likelihood estimation method, whereas the coefficients in LDA are calculated from the estimated mean and variance of the normal distribution.

Assume that the conditional distributions for the predictor variables $\boldsymbol{X} = (x1, x2, x3, x4, x5, x6, x7)^T$ are multivariate normal, with a group-specific mean vector and a common covariance matrix, multicollinearity, and independence.

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

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 10: Summary of the LDA fitting model

We can get the prior probabilities: $\Pr(Class = 0) = 0.6500732,\ \Pr(Class = 1) = 0.3499268.$

When the response variable Class = 0 (i.e. benign breast lesions), the group means for Cl.thickness, Cell.shape, Marg.adhesion, Bare.nuclei, Bl.cromatin, Normal.nucleoli, and Mitoses are 2.963964, 1.414414, 1.346847, 1.346847, 2.083333, 1.261261, and 1.065315, respectively. The means of these four groups (Cell.shape, Marg.adhesion, Bare.nuclei, Normal.nucleoli) are very close.

When the response variable Class=1 (representing malignant breast cancer), the group means for the predictor variables Cl.thickness, Cell.shape, Marg.adhesion, Bare.nuclei, Bl.cromatin, Normal.nucleoli, and Mitoses are 7.188285, 6.560669, 5.585774, 7.627615, 5.974895, 5.857741, and 2.543933, respectively. The means of these two groups (Cl.thickness, Bare.nuclei) are very close.

When the response variable Class=0, the group mean of all seven predictor variables is less than 5. When the response variable Class=1, the group mean of the first six predictor variables is greater than 5 and less than 10, whereas the group mean of the seventh predictor variable mitosis is greater than 2 and less

than 5. This is because normal cells can produce two daughter cells through mitosis, whereas only cancer cells can divide into three to four daughter cells. This also suggests that the response variable is positively correlated with the 7 predictor variables, which have high values, so it is likely that Class=1, i.e. malignant breast cancer.

The test error in the LDA model using 10-fold cross-validation is 0.04099 > 0.0322. (Test errors for the best subset of models in logistic regression) Therefore, the LDA model does not fit as well as the best subset model in logistic regression.

2.5.2 QDA Bayesian classifier

The most significant difference between QDA and LDA is the decision boundary, one is a curve and the other is a straight line. If the sample size n is large and the sample covariance matrix for the predictor variables in the K groups differ substantially, QDA is likely to perform better than LDA. In the MyCancer dataset, p=7, K=2, n=683, so we also need to consider the QDA model.

Assume that the conditional distributions for the predictor variables $\boldsymbol{X}$ are multivariate normal, with a group-specific mean vector, and different groups are allowed to have different covariance matrices, and multicollinearity, independence.

```
qda_fit = qda(Class ~ ., data=MyCancer_red)
```

The QDA model has the same prior probabilities as the LDA. When the response variable Class=0 or Class=1, the group members of QDA all have the same mean value as LDA. This again shows that the smaller the value of the seven predictor variables, the healthier the cell looks in terms of that attribute, i.e., the more likely the value of the response variable Class is to be 0. There is a positive correlation between the response variable and the predictor variables.

The test error in the QDA model using 10-fold cross-validation is 0.04831>0.04099(LDA model). As a result, in this dataset, the QDA model did not fit as well as the LDA model, which did not work as well as the best subset models in logistic regression.

3 Model comparison and Conclusion

According to Table 2, a total of four models were developed, and two cross-validation procedures were employed to calculate the test errors. From a fairness standpoint, we prefer to use k-fold cross-validation (especially 5-fold or 10-fold). K-fold is less volatile, has less bias, and does not overstate the test error compared to the validation set approach because all the data is used for training and testing. In contrast, the validation set technique employs data splitting, which can result in more or less favourable outcomes by chance. The best subset of the logistic regression models fit the best, based on the test error of the 10-fold cross-validation, correctly fitting 96.78% of the the data. Whereas the LDA model fits best based on the test error of the validation set method, we find that the test error is smaller than the training error at this stage, which is an unusual occurrence, presumably because the distributional difference between the training and test sets is greater. In conclusion, the best model in the MyCancer dataset is the best subset model of logistic regression (7 variables), and the most equitable method for comparing models is to measure the test error using k-fold cross-validation.

	Logistic regression		Discriminant analysis	
Comparison	Full model	Best Subsets (7-variables)	Linear discriminant analysis	Quadratic discriminant analysis
Train error	0.03074671	0.03074671	0.03953148	0.04538799
Test error (10-fold Cross-validation)	0.03367496	0.03221083	0.04099561	0.04831625
Test error (validation set methods)	0.04451039	0.04154303	0.03560831	0.04747774

Table 2: Test errors under different models

Appendix 1

```
install.packages('mlbench')
library(mlbench)
data("BreastCancer")
dim(BreastCancer) #699rows 11colunms
head(BreastCancer,3) #print the first three rows
summary(BreastCancer)
?BreastCancer
class(BreastCancer[2,11]) #factor not numeric
is.factor(BreastCancer$Class) #TRUE

#Print 24th row of Breast Cancer data and note there is a NA in the Bare.nuclei column:
BreastCancer[24,]
#Test whether each element on the 24th row is a NA:
is.na(BreastCancer[24,])
#Count the number of NA in the whole BreastCancer dataset
table(is.na(BreastCancer))
#Returns the coordinates of the corresponding row of missing values
which(is.na(BreastCancer),arr.ind = TRUE)

#Delete rows containing NA
MyClean=na.omit(BreastCancer)
#We can then use the 'table' function to check whether the new data set still contains NA
table(is.na(MyClean)) #no number of TRUE

typeof(MyClean$Class) #integer
head(as.integer(MyClean$Class)) #value is 1 or 2
#Delete the first column of Sample code number and the 11th column of Class
#Add a new column for the response variable Class with a value between 0 and 1
MyCancer = data.frame(MyClean[,-c(1,11)] ,Class=as.integer(MyClean$Class)-1)
#Convert the first nine columns of predictor variables to numeric
MyCancer$Cl.thickness=as.numeric(MyCancer$Cl.thickness)
MyCancer$Cell.size=as.numeric(MyCancer$Cell.size)
MyCancer$Cell.shape=as.numeric(MyCancer$Cell.shape)
MyCancer$Marg.adhesion=as.numeric(MyCancer$Marg.adhesion)
MyCancer$Epith.c.size=as.numeric(MyCancer$Epith.c.size)
MyCancer$Bare.nuclei=as.numeric(MyCancer$Bare.nuclei)
MyCancer$Bl.cromatin=as.numeric(MyCancer$Bl.cromatin)
MyCancer$Normal.nucleoli=as.numeric(MyCancer$Normal.nucleoli)
MyCancer$Mitoses=as.numeric(MyCancer$Mitoses)
#Check that the nine cytological characteristics (predictor variables) are already numerical
str(MyCancer)
head(MyCancer,3)
dim(MyCancer)    #683 rows 10 columns
```

Appendix 2

```
table(MyCancer$Class) #0-benign:444,1-malignant:239
pairs(MyCancer[,1:9],col=MyCancer[,10]+1)
cor(MyCancer) #Find linear relationships

#Logistic model-full model
n=nrow(MyCancer)
p=ncol(MyCancer)-1
logreg_fit = glm(Class ~ ., data=MyCancer, family="binomial")
summary(logreg_fit)
summary(logreg_fit)$coef
summary(logreg_fit)$coef[,4]
phat = predict(logreg_fit, MyCancer, type="response")
yhat = as.numeric(ifelse(phat > 0.5, 1, 0))
1-mean(yhat==MyCancer$Class)      #train error:0.03074671

#k-fold: Set the seed (say, at 5) to make the analysis reproducible
set.seed(5)
## Sample the fold-assignment index
nfolds = 10
fold_index = sample(nfolds, n, replace=TRUE)
#A random sample of 683 numbers from 0-10 with release
## Print the first few fold-assignments
head(fold_index)

##10-fold test error function
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)
   for(fold in 1:nfolds) {
```

```
    fold_errors[fold] = fold_error_function(X, y, fold_ind==fold)
  }
  fold_sizes = numeric(nfolds)
  for(fold in 1:nfolds) fold_sizes[fold] = length(which(fold_ind==fold))
  test_error = weighted.mean(fold_errors, w=fold_sizes)
  return(test_error)
}

test_error = general_cv(MyCancer[,1:p], MyCancer[,p+1], fold_index, logistic_reg_fold_error)
#the full model test error(10-folds):0.03367496

##  Calculation of test errors for the full model using the validation set method
train_set=sample(c(TRUE,FALSE),nrow(MyCancer),replace=TRUE)
lr_train=glm(Class~.,data=MyCancer[train_set,],family='binomial')
summary(lr_train)
phat_test2=predict(lr_train,MyCancer[!train_set,],family='binomial')
yhat_test2=ifelse(phat_test2>0.5,1,0)
1-mean(yhat_test2==MyCancer$Class[!train_set])

##Best subset selection in logistic regression model
library(leaps)
library(bestglm)
?bestglm
bss_fit_AIC = bestglm(MyCancer, family=binomial, IC="AIC")
bss_fit_BIC = bestglm(MyCancer, family=binomial, IC="BIC")
bss_fit_AIC$Subsets
bss_fit_BIC$Subsets
## Identify best-fitting models
best_AIC=bss_fit_AIC$ModelReport$Bestk    #7
best_BIC=bss_fit_BIC$ModelReport$Bestk    #5

##k-fold verification in the best subset
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)
  for(fold in 1:nfolds) {
    tmp_fit = bestglm(Xy[fold_ind!=fold,], family=binomial, IC="AIC")
    best_models = as.matrix(tmp_fit$Subsets[,2:(1+p)])
    for(k in 1:(p+1)) {
      fold_errors[fold, k] = logistic_reg_fold_error(X[,best_models[k,]], y, fold_ind==fold)
```

```
    }
  }
  fold_sizes = numeric(nfolds)
  for(fold in 1:nfolds) fold_sizes[fold] = length(which(fold_ind==fold))
  test_errors = numeric(p+1)
  for(k in 1:(p+1)) {
    test_errors[k] = weighted.mean(fold_errors[,k], w=fold_sizes)
  }
  return(test_errors)
}
## Apply the cross-validation for best subset selection function
cv_errors = logistic_reg_bss_cv(MyCancer[,1:p], MyCancer[,p+1], fold_index)
#0.34992679 0.07027818 0.04685212 0.04099561 0.04245974 0.03660322 0.03367496
0.03221083 0.03367496 0.03367496
## Identify the number of predictors in the model which minimises test error
best_cv = which.min(cv_errors) – 1          #7
```

k-fold validation in the best subset to calculate the test error

```
pstar=7
#check which predictors are in the 7-predictor model
bss_fit_AIC$Subsets[pstar+1,]
## Construct a reduced data set containing only the 7 selected predictors
bss_fit_AIC$Subsets[pstar+1, 2:(p+1)]
indices = which(bss_fit_AIC$Subsets[pstar+1, 2:(p+1)]==TRUE)
MyCancer_red=MyCancer[,c(indices,p+1)]
# Obtain regression coefficients for this best subset model
logreg1_fit = glm(Class ~ ., data=MyCancer_red, family="binomial")
summary(logreg1_fit)
test_error_red = general_cv(MyCancer_red[,1:pstar], MyCancer_red[,pstar+1], fold_index,
logistic_reg_fold_error)
test_error_red #0.03221083
```

 # Calculating training error in the best subset

```
logreg_fit_train=glm(Class ~ .,data=MyCancer_red,family='binomial')
phat_log_train=predict(logreg_fit_train,MyCancer_red,type='response')
yhat_log_train=ifelse(phat_log_train>0.5,1,0)
1-mean(yhat_log_train==MyCancer_red$Class)    #  0.03074671
```

Test errors for the best subset calculated using the validation set method

```
summary(glm(Class~Cl.thickness+Cell.shape+Marg.adhesion+Bare.nuclei+Bl.cromatin+Normal.nu
cleoli+Mitoses,family='binomial',data=MyCancer))
lr_red_train=glm(Class~Cl.thickness+Cell.shape+Marg.adhesion+Bare.nuclei+Bl.cromatin+Normal
.nucleoli+Mitoses,data=MyCancer[train_set,],family='binomial')
head(lr_red_train)
```

```
phat_red_test=predict(lr_red_train,MyCancer[!train_set,],type='response')
yhat_red_test=ifelse(phat_red_test>0.5,1,0)
1-mean(yhat_red_test==MyCancer$Class[!train_set])
#test error of 7-variables subset model:  0.4836795
```

#LDA model with 7-vriables

```
install.packages('dplyr')
library(MASS)
## MyCancer_red=MyCancer[,c(indices,p+1)]
lda_fit = lda(Class ~ ., data=MyCancer_red)
lda_predict = predict(lda_fit, MyCancer_red)
lda_fit$prior
lda_fit$means
lda_fit$scaling
yhat_lda = lda_predict$class
confusion_lda = table(Observed=MyCancer_red$Class, Predicted=yhat_lda)
1-sum(diag(confusion_lda))/sum(confusion_lda)
1 - mean(MyCancer_red$Class == yhat_lda)
```

#training error: 0.03953148

Use 10-fold to verify test error in the LDA model (7-variable)

```
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)
}
test_error_lda = general_cv(MyCancer_red[,1:pstar], MyCancer_red[,pstar+1], fold_index,
lda_fold_error)
```

#test error：0.04099561（10-fold cross-validation）

7-variable lda model validation test error with **validation set**

```
lda_train=lda(Class~Cl.thickness+Cell.shape+Marg.adhesion+Bare.nuclei+Bl.cromatin+Normal.nu
cleoli+Mitoses,data=MyCancer_red[train_set,])
lda_test=predict(lda_train,MyCancer_red[!train_set,])
yhat_test_lda=lda_test$class
1-mean(yhat_test_lda==MyCancer_red$Class[!train_set])
```

```
#QDA model with 7-variables
qda_fit = qda(Class ~ ., data=MyCancer_red)
summary(qda_fit)
qda_predict = predict(qda_fit, MyCancer_red)
```

```
yhat_qda = qda_predict$class
confusion_qda = table(Observed=MyCancer_red$Class, Predicted=yhat_qda)
1-sum(diag(confusion_qda))/sum(confusion_qda)
1-mean(yhat_qda==MyCancer_red$Class)    #training error: 0.04538799

#10-fold test error for QDA model
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)
}
test_error_qda = general_cv(MyCancer_red[,1:pstar], MyCancer_red[,pstar+1], fold_index,
qda_fold_error)

#validation set method test error for QDA model
qda_train=qda(Class~Cl.thickness+Cell.shape+Marg.adhesion+Bare.nuclei+Bl.cromatin+Normal.n
ucleoli+Mitoses, data=MyCancer_red[train_set,])
qda_test=predict(qda_train,MyCancer_red[!train_set,])
yhat_test_qda=qda_test$class
1-mean(yhat_test_qda==MyCancer_red$Class[!train_set])
```