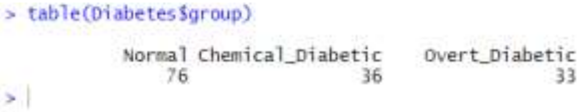
Kunal Mehta
50352960
kunalkam

Homework 4

Problem 1

a. An immediate observation about the Diabetes dataset to be used as the base for this problem is that it is a very small dataset, with even fewer observations than the Iris dataset. Simply observing the variable names and a few observations of the data does not really offer too much insight about the data. However, tabulating the different classes by their occurrence does reveal interesting information.

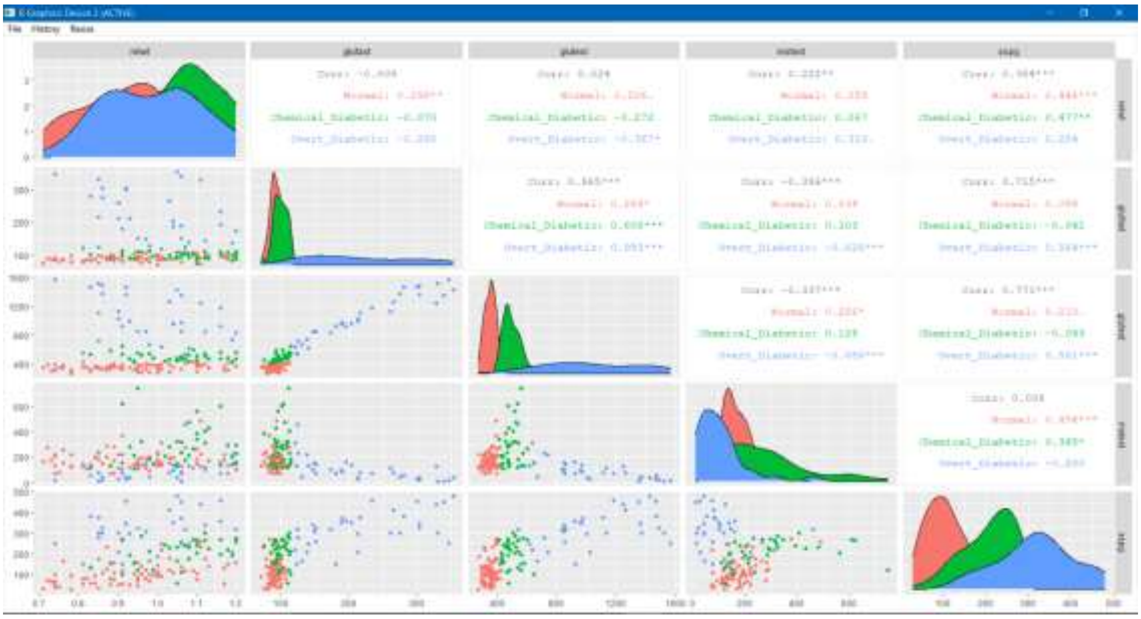


```
> table(Diabetes$group)

          Normal Chemical_Diabetic    Overt_Diabetic 
              76                36                33 
> |
```

An overwhelming majority of the group classifications belong to the class Normal, having more occurrences than the other two classes combined. Based on this table and as per the assumption made in the question statement, the probabilities for each class are:

Normal:	52.41%
Chemical_Diabetic:	24.83%
Overt_Diabetic:	22.76%

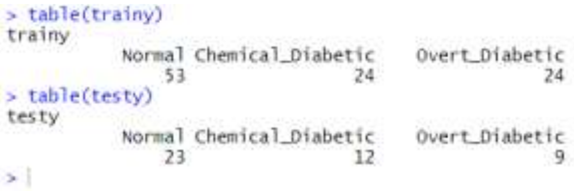
b. The next step was to produce a pairwise scatter plot of the first five variables, with the group variable being represented by different colors.



As can be seen from the plots and the values, the classes definitely do have different covariance matrices. For instance, the variance for class Overt_Diabetic for glufast and glutest is much higher than that of the other two classes.

The classes are not multi-variate normal either. For the same variables above, the Overt_Diabetic class cannot be said to have a normal distribution. At best, it is an approximate normal distribution that has a very high variance.

c. The next step was to divide the data into training and testing datasets, which I did by a 70-30 split. I also separated the Y values of the training and testing data and tabulated their distribution.

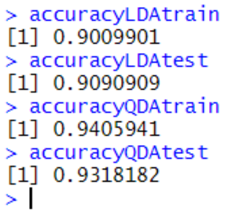


```
> table(trainy)
trainy
          Normal Chemical_Diabetic    Overt_Diabetic
              53                24                24
> table(testy)
testy
          Normal Chemical_Diabetic    Overt_Diabetic
              23                12                 9
> |
```

This feels like a satisfactory split in the sense that it more or less reflects the distribution in the overall total dataset, with Normal class having slightly more than the other two combined in both the training and testing data, and the Chemical and Overt classes having roughly the same number of observations.

d. The next part of the question involves applying LDA and QDA to the dataset and comparing their performance. This was relatively straightforward, and I recorded the training and testing accuracies of both methods in appropriately named variables. The results are:

	LDA	QDA
Training Accuracy	90.099%	94.059%
Testing Accuracy	90.909%	93.182%

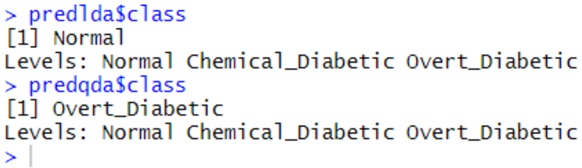


```
> accuracyLDAtrain
[1] 0.9009901
> accuracyLDAtest
[1] 0.9090909
> accuracyQDAtrain
[1] 0.9405941
> accuracyQDAtest
[1] 0.9318182
> |
```

As can be observed from the table above, QDA performs better than LDA in terms of accuracy. Since it could be that this was a coincidence due to the split of data, I used different seed values in the manner that I detailed in the last homework, but QDA always performed at least slightly better than LDA.

e. The last part of this problem detailed a specific case and which class each of the methods above classified this case as was to be mentioned. This was a straightforward problem, as I simply made a data frame with the same column names as the main Diabetes dataset and 1 single observation as the case data specified. Then it was simply a case of using the LDA and QDA fits to predict the class that this case would be classified as.

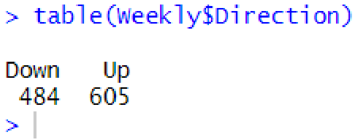
LDA Classification: Normal
QDA Classification: Overt_Diabetic



```
> predlda$class
[1] Normal
Levels: Normal Chemical_Diabetic Overt_Diabetic
> predqda$class
[1] Overt_Diabetic
Levels: Normal Chemical_Diabetic Overt_Diabetic
> 
```

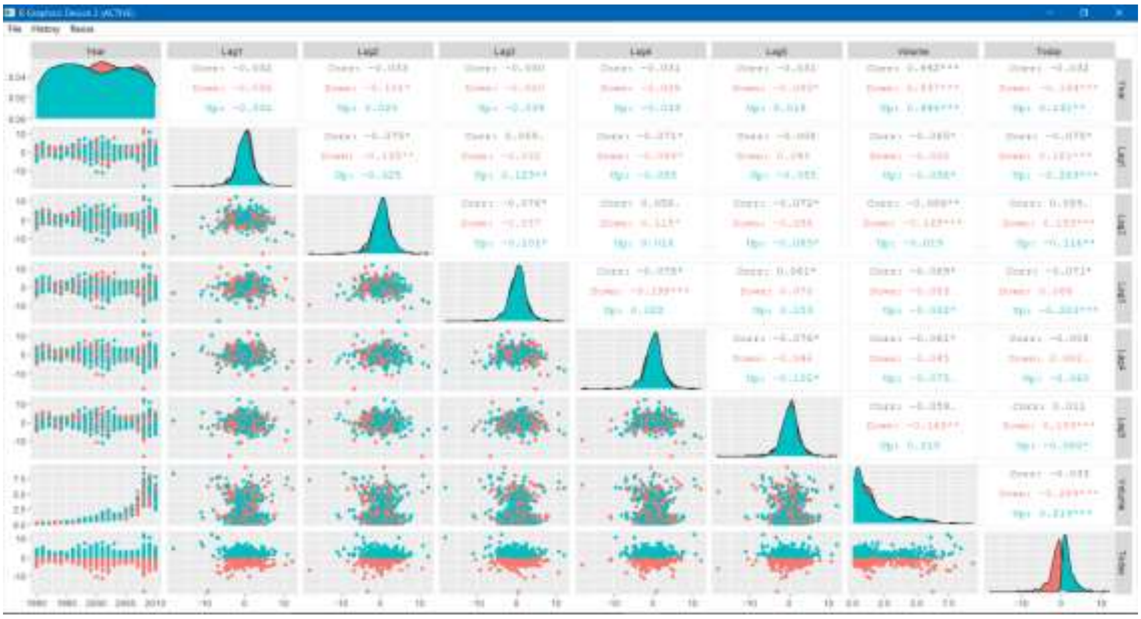
Problem 2

a. The Weekly dataset is a much larger dataset than the Diabetes one. Ordinary names and head functions on the dataset do not really reveal much about the dataset, nor does a summary. The numerical statistics about the data are informative, but not particularly useful, except for the detail about the Direction variable. 484 of 1089 observations record Down, while 605 record Up. I tabulated these results separately as shown below.

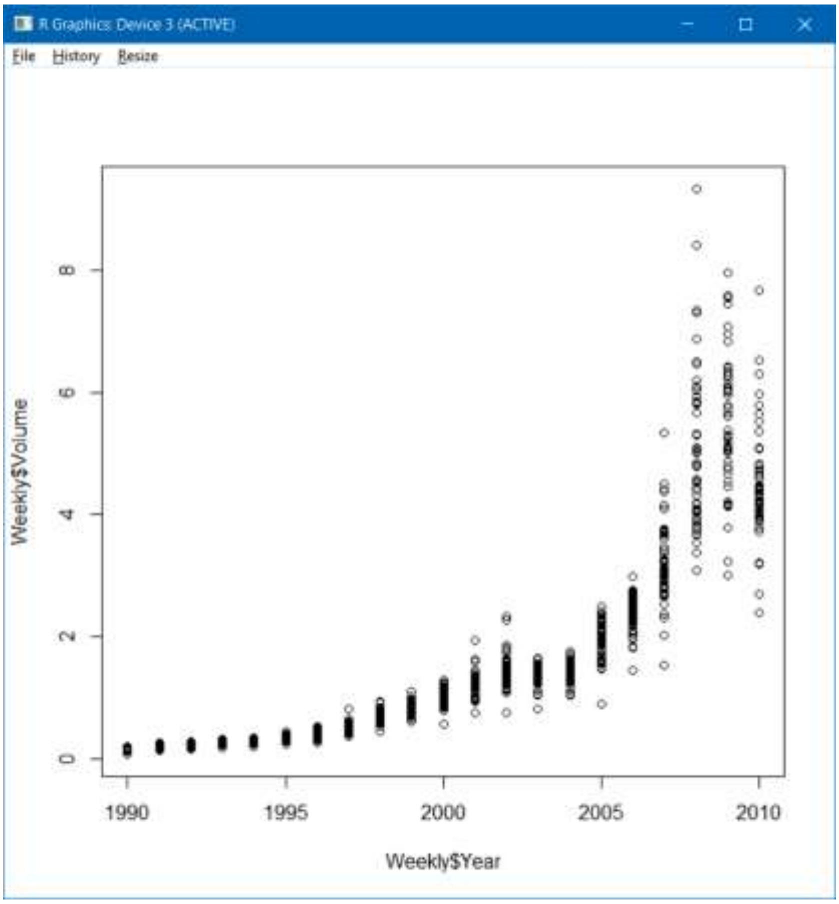


```
> table(Weekly$Direction)

Down   Up 
 484  605 
> 
```

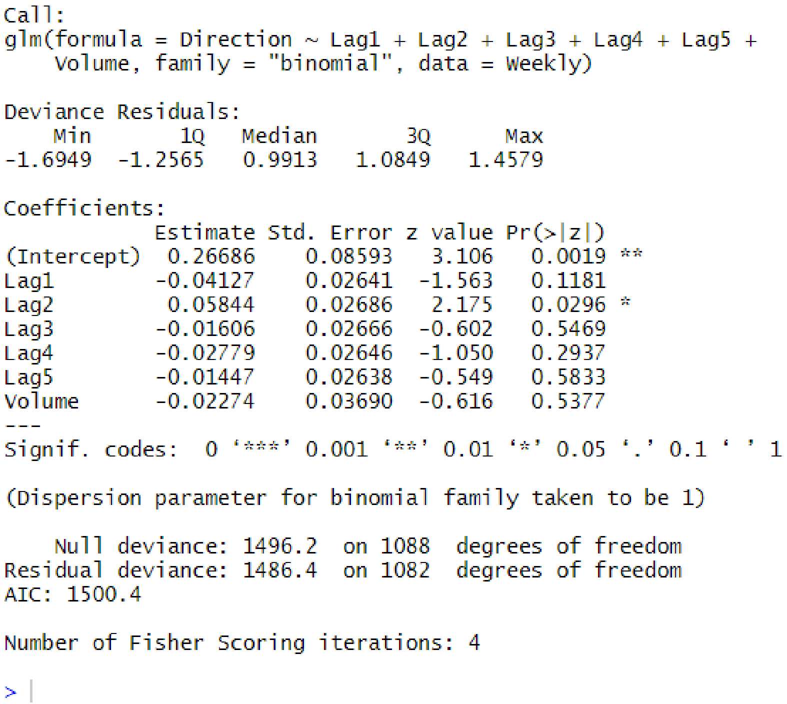
A pairwise scatterplot of all the variables with the Direction variable being depicted by two different colours shows that no variables except Year and Volume seem to have any tangible correlation, shown by their insignificant correlation values.



I therefore plotted Year vs Volume to see in greater detail how the values correlate.



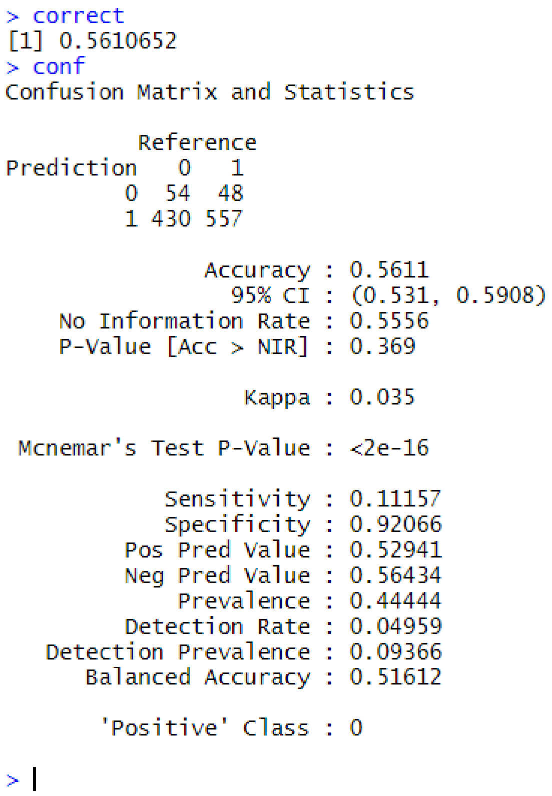
b. The next step was to perform logistic regression on all of the data with Direction being the response variable, and all the variables minus Year and Today as predictors. This process was simple enough, with the summary of this regression displayed below.



```
Call:
glm(formula = Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5 + 
    Volume, family = "binomial", data = Weekly)

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-1.6949  -1.2565   0.9913   1.0849   1.4579  

Coefficients:
            Estimate Std. Error z value Pr(>|z|)   
(Intercept)  0.26686    0.08593   3.106   0.0019 **
Lag1        -0.04127    0.02641  -1.563   0.1181   
Lag2         0.05844    0.02686   2.175   0.0296 * 
Lag3        -0.01606    0.02666  -0.602   0.5469   
Lag4        -0.02779    0.02646  -1.050   0.2937   
Lag5        -0.01447    0.02638  -0.549   0.5833   
Volume      -0.02274    0.03690  -0.616   0.5377   
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1496.2  on 1088  degrees of freedom
Residual deviance: 1486.4  on 1082  degrees of freedom
AIC: 1500.4

Number of Fisher Scoring iterations: 4

> |
```

From the summary, only Lag2 seems to be important enough to have an impact on predicting the Direction. The reason that Lag2 can be said to be statistically significant is its p-value.

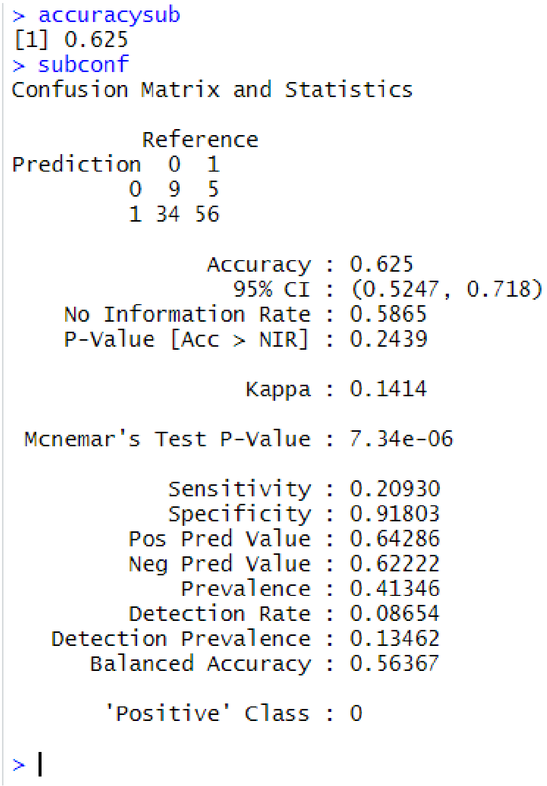
c. The next step was to predict the direction based on the logistic regression fit, and then compute the confusion matrix and overall fraction of correct predictions. This was a straightforward task with the following results.



```
> correct
[1] 0.5610652
> conf
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0  54  48
         1 430 557

               Accuracy : 0.5611
                 95% CI : (0.531, 0.5908)
    No Information Rate : 0.5556
    P-Value [Acc > NIR] : 0.369

                  Kappa : 0.035

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.11157
            Specificity : 0.92066
         Pos Pred Value : 0.52941
         Neg Pred Value : 0.56434
             Prevalence : 0.44444
         Detection Rate : 0.04959
   Detection Prevalence : 0.09366
      Balanced Accuracy : 0.51612

       'Positive' Class : 0

> |
```

The number of correct predictions was 611, with an accuracy percentage of approximately 56.11%. The 0-0 and 1-1 elements of the confusion matrix indicate correct predictions, while the 0-1 and 1-0 elements indicate a difference in the prediction and the true value. The confusion matrix indicates that an overwhelming majority of the predictions made indicate Up, with just 102 of 1089 being predictions for Down, about 9.37%. True, this gives a high number of accurate 1-1 cases, 51.15% of the total, but a similarly high number of wrong 1-0 cases, 430 to be precise, 39.48% of the total which are nothing but false positives. Also, for a total of 484 true values which are Down, logistic regression only predicts 54 of them accurately, meaning that the true negative rate is an abysmal 11.16%.

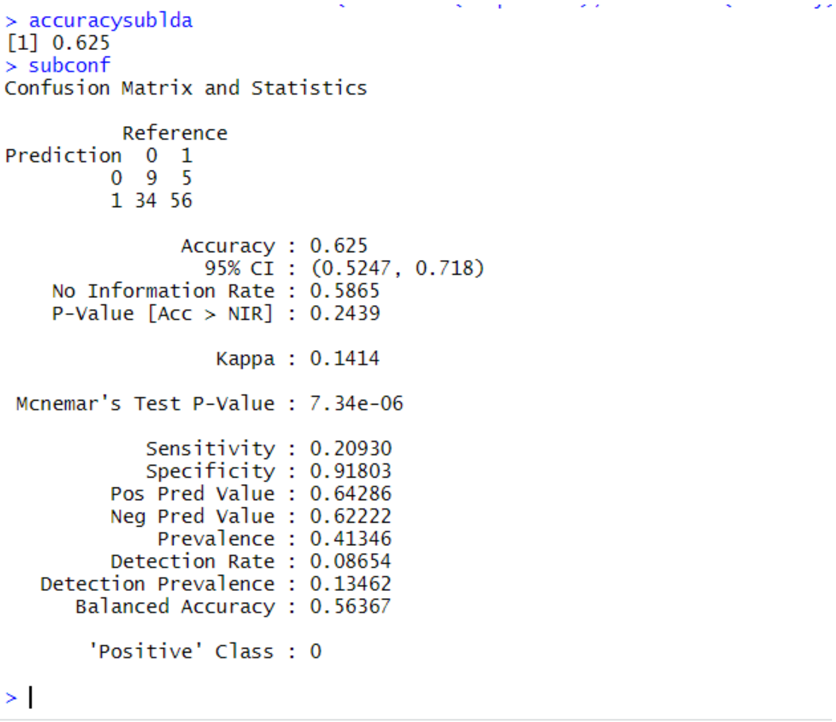
d. The next step was to divide the data into training and testing data on the basis of the date period, with observations in the time range of 1990-2008 being the training data and the rest being testing data. This was a straightforward way to subset the data. A logistic regression operation was to be performed on these datasets, with just the Lag2 variable as a predictor. The following results were obtained.



```
> accuracysub
[1] 0.625
> subconf
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0  9  5
         1 34 56

               Accuracy : 0.625
                 95% CI : (0.5247, 0.718)
    No Information Rate : 0.5865
    P-Value [Acc > NIR] : 0.2439

                  Kappa : 0.1414

 Mcnemar's Test P-Value : 7.34e-06

            Sensitivity : 0.20930
            Specificity : 0.91803
         Pos Pred Value : 0.64286
         Neg Pred Value : 0.62222
             Prevalence : 0.41346
         Detection Rate : 0.08654
   Detection Prevalence : 0.13462
      Balanced Accuracy : 0.56367

       'Positive' Class : 0

> |
```

The accuracy of logistic regression with this datasets and just Lag2 as a predictor is 62.5%. Not too many surprises here as well as a majority of predictions are for Up, with just 14 of 104 being for Down.

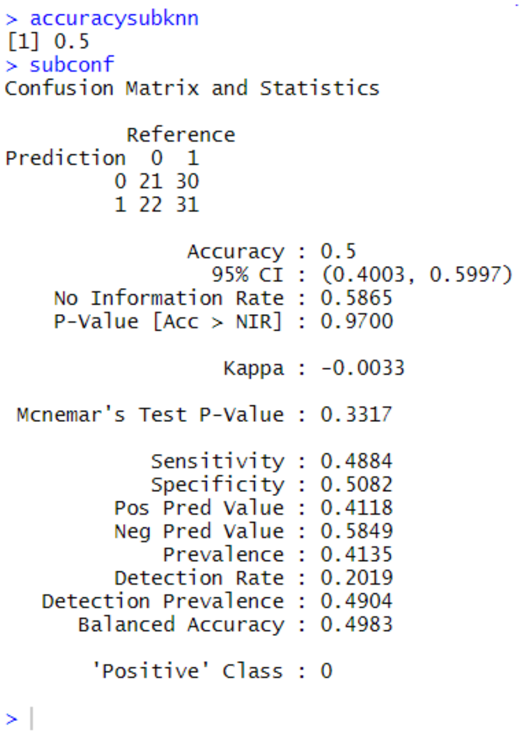
e. The next step was to perform the same process using Linear Discriminant Analysis. The results of that process are shown below.



```
> accuracysublda
[1] 0.625
> subconf
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0  9  5
         1 34 56

               Accuracy : 0.625
                 95% CI : (0.5247, 0.718)
    No Information Rate : 0.5865
    P-Value [Acc > NIR] : 0.2439

                  Kappa : 0.1414

 Mcnemar's Test P-Value : 7.34e-06

            Sensitivity : 0.20930
            Specificity : 0.91803
         Pos Pred Value : 0.64286
         Neg Pred Value : 0.62222
             Prevalence : 0.41346
         Detection Rate : 0.08654
   Detection Prevalence : 0.13462
      Balanced Accuracy : 0.56367

       'Positive' Class : 0

> |
```

The results are identical to those of logistic regression. Both methods have an accuracy of 62.5%

f. The next step was to perform k-Nearest Neighbors algorithm on the same data with k=1. For this, the data needed to be converted to matrix form, for which I separated the Lag2 variable for both the train and test data. The results for this method are shown below.



```
> accuracysubknn
[1] 0.5
> subconf
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 21 30
         1 22 31

               Accuracy : 0.5
                 95% CI : (0.4003, 0.5997)
    No Information Rate : 0.5865
    P-Value [Acc > NIR] : 0.9700

                  Kappa : -0.0033

 Mcnemar's Test P-Value : 0.3317

            Sensitivity : 0.4884
            Specificity : 0.5082
         Pos Pred Value : 0.4118
         Neg Pred Value : 0.5849
             Prevalence : 0.4135
         Detection Rate : 0.2019
   Detection Prevalence : 0.4904
      Balanced Accuracy : 0.4983

       'Positive' Class : 0

> |
```

g. Of all these methods the most accurate are logistic regression and linear discriminant analysis. Maybe if we did still have the h part of the original assignment, we’d be able to isolate a particular case with one of the methods with a particular set of predictors as having the best results, but as of now with the information we have, both linear discriminant analysis and logistic regression are apparently the best methods.