Homework 1 Exercise 2 and 3

Ma Jingchun, 2020111235

This question should be answered using the Weekly data set, which is part of the ISLR2 package. It contains 1, 089 weekly returns for 21 years, from the beginning of 1990 to the end of 2010.

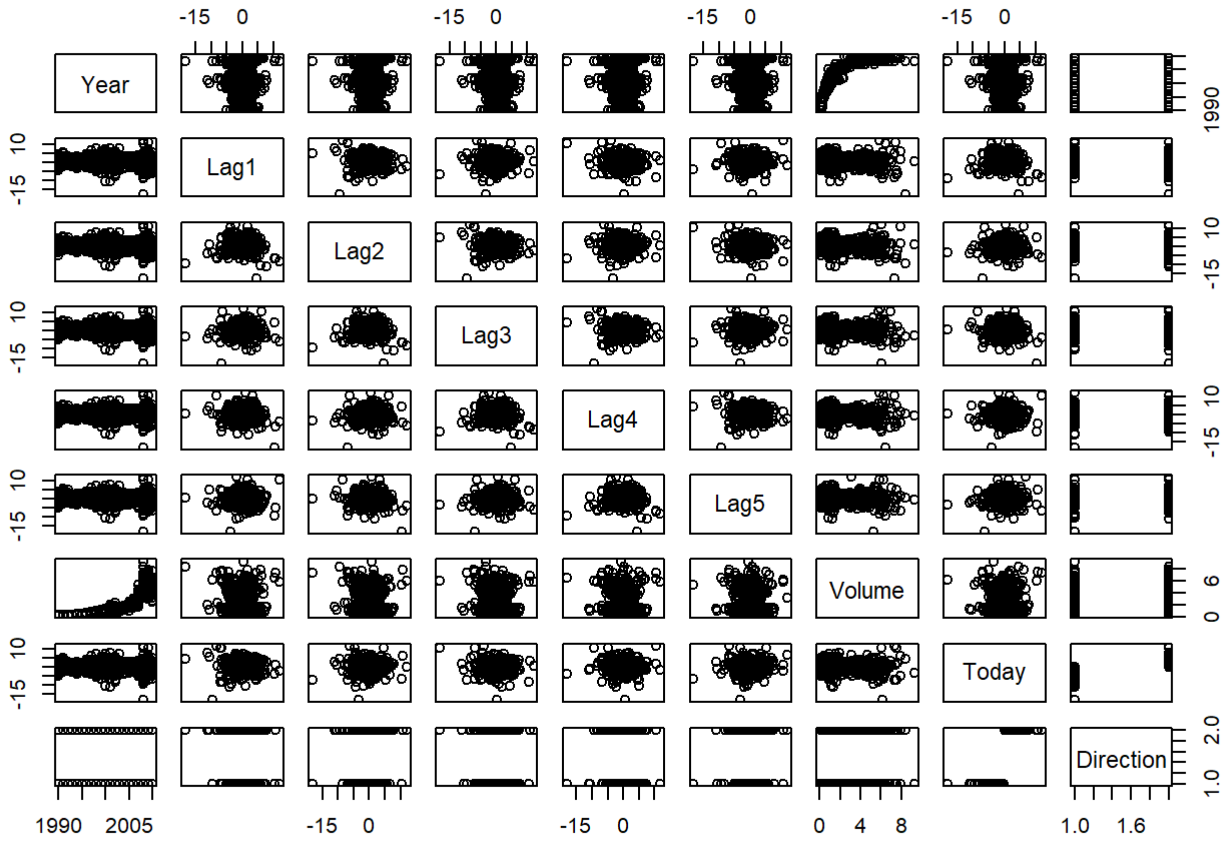
```
library(ISLR2)
library(MASS)
library(class)
library(e1071)
```

1. Produce some numerical and graphical summaries of the Weekly data. Do there appear to be any patterns?

```
summary(Weekly)
```

```
##       Year           Lag1               Lag2               Lag3
##  Min.   :1990   Min.   :-18.1950   Min.   :-18.1950   Min.   :-18.1950
##  1st Qu.:1995   1st Qu.: -1.1540   1st Qu.: -1.1540   1st Qu.: -1.1580
##  Median :2000   Median :  0.2410   Median :  0.2410   Median :  0.2410
##  Mean   :2000   Mean   :  0.1506   Mean   :  0.1511   Mean   :  0.1472
##  3rd Qu.:2005   3rd Qu.:  1.4050   3rd Qu.:  1.4090   3rd Qu.:  1.4090
##  Max.   :2010   Max.   : 12.0260   Max.   : 12.0260   Max.   : 12.0260
##       Lag4               Lag5              Volume            Today
##  Min.   :-18.1950   Min.   :-18.1950   Min.   :0.08747   Min.   :-18.1950
##  1st Qu.: -1.1580   1st Qu.: -1.1660   1st Qu.:0.33202   1st Qu.: -1.1540
##  Median :  0.2380   Median :  0.2340   Median :1.00268   Median :  0.2410
##  Mean   :  0.1458   Mean   :  0.1399   Mean   :1.57462   Mean   :  0.1499
##  3rd Qu.:  1.4090   3rd Qu.:  1.4050   3rd Qu.:2.05373   3rd Qu.:  1.4050
##  Max.   : 12.0260   Max.   : 12.0260   Max.   :9.32821   Max.   : 12.0260
##  Direction
##  Down:484
##  Up  :605
##
##
##
##
```

```
pairs(Weekly)
```



The correlation between the data is not strong. ‘Lags’ as well as ‘Today’ are very similar to each other.

2. Use the full data set to perform a logistic regression with Direction as the response and the five lag variables plus Volume as predictors. Use the summary function to print the results. Do any of the predictors appear to be statistically significant? If so, which ones?

```
glm.fit1 = glm(Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5 + Volume, data = Weekly, family = binomial)
summary(glm.fit1)
```

```
##
## 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
```

Lag2 is most significant. Because Pr(>|z|) < 0.05, so Lag1, Lag2 and Lag4 are statistically significant compared to others.

3. Compute the confusion matrix and overall fraction of correct predictions. Explain what the confusion matrix is telling you about the types of mistakes made by logistic regression.

```
glm.probs = predict(glm.fit1, type='response')
glm.pred = rep("Down", nrow(Weekly))
glm.pred[glm.probs > .5] = "Up"
table(glm.pred, Weekly$Direction)
```

```
##
## glm.pred Down  Up
##     Down   54  48
##     Up    430 557
```

```
mean(glm.pred == Weekly$Direction)
```

```
## [1] 0.5610652
```

```
430 / (54 + 430)
```

```
## [1] 0.8884298
```

```
48 / (48 + 557)
```

```
## [1] 0.07933884
```

overall fraction of correct predictions = (54 + 557) / (54 + 48 + 430 + 557) = 56.1%

false positive rate = 430 / (54 + 430) = 88.8%

false negative rate = 48 / (48 + 557) = 7.9%

So the error should be type 2 error

4. Now fit the logistic regression model using a training data period from 1990 to 2008, with Lag2 as the only predictor. Compute the confusion matrix and the overall fraction of correct predictions for the held out data (that is, the data from 2009 and 2010).

```
attach(Weekly)
train = (Year < 2009)
Weekly.train = Weekly[train,]
Weekly.test = Weekly[!train,]
Direction.test = Weekly.test$Direction
glm.fit2 = glm(Direction~Lag2, data=Weekly, family=binomial, subset=train)
glm.probs2 = predict(glm.fit2, Weekly.test, type=’response’)
glm.pred2 = rep(’Up’,nrow(Weekly.test))
glm.pred2[glm.probs2<.5] = ’Down’
table(glm.pred2, Direction.test)
```

```
##          Direction.test
## glm.pred2 Down Up
##      Down    9  5
##      Up     34 56
```

```
mean(glm.pred2 == Direction.test)
```

```
## [1] 0.625
```

overall fraction of correct predictions = (9 + 56) / (9 + 56 + 5 + 34) = 62.5%

5. Repeat 4. using LDA.

```
lda.fit = lda(Direction~Lag2, data=Weekly, subset=train)
lda.pred = predict(lda.fit, Weekly.test)
lda.class = lda.pred$class
table(lda.class, Direction.test)
```

```
##          Direction.test
## lda.class Down Up
##      Down    9  5
##      Up     34 56
```

```
mean(lda.class == Direction.test)
```

```
## [1] 0.625
```

overall fraction of correct predictions = (9 + 56) / (9 + 56 + 5 + 34) = 62.5%

6. Repeat 4. using QDA.

```
qda.fit = qda(Direction~Lag2, data=Weekly, subset=train)
qda.pred = predict(qda.fit, Weekly.test)
qda.class = qda.pred$class
table(qda.class, Direction.test)
```

```
##          Direction.test
## qda.class Down Up
##      Down    0  0
##      Up     43 61
```

```
mean(qda.class == Direction.test)
```

```
## [1] 0.5865385
```

overall fraction of correct predictions = 61 / (43 + 61) = 58.6%

7. Repeat 4. using KNN with K = 1. You can also experiment with values for K in the KNN classifier. (Hint: Use knn() in the class package.)

```
train.X = as.matrix(Weekly$Lag2[train])
test.X = as.matrix(Weekly$Lag2[!train])
Direction.train = Weekly$Direction[train]
set.seed(1)
knn.pred = knn(train.X, test.X, Direction.train, k=1)
table(knn.pred, Direction.test)
```

```
##         Direction.test
## knn.pred Down Up
##     Down   21 30
##     Up     22 31
```

```
mean(knn.pred == Direction.test)
```

```
## [1] 0.5
```

overall fraction of correct predictions = (21 + 31) / (21 + 31 + 30 + 22) = 50%

8. Repeat 4. using naive Bayes.

```
nb.fit <- naiveBayes(Direction~Lag2, data=Weekly, subset=train)
nb.class = predict(nb.fit, Weekly.test)
table(nb.class, Direction.test)
```

```
##         Direction.test
## nb.class Down Up
##     Down    0  0
##     Up     43 61
```

```
mean(nb.class == Direction.test)
```

```
## [1] 0.5865385
```

overall fraction of correct predictions = 61 / (43 + 61) = 58.6%

9. Which of these methods appears to provide the best results on this data?

Logistic regression and linear discriminant analysis give better predictions than others