Midterm Assignemnt, ME314 2019

#### Summer School 2019 midsession examination

# ME314 Introduction to Data Science and Machine Learning

## Suitable for all candidates

### Instructions to candidates

* Complete the assignment by adding your answers directly to the RMarkdown document, knitting the document, and submitting the HTML file to Moodle.
* Time allowed: due 19:00 on Wednesday, 7th August 2019.
* Submit the assignment via [Moodle](https://shortcourses.lse.ac.uk/course/view.php?id=158).

## Question 1:

This question should be answered using the Carseats data set, which is part of the **ISLR** package. This data contains simulated data set containing sales of child car seats at 400 different stores.

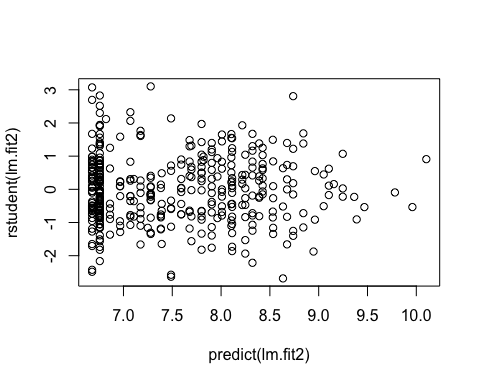
data("Carseats", package = "ISLR")  
  
#Regressiom Model 1 (Advertising & Price)  
lm.fit1 <- lm(Sales ~ Advertising + Price, data = Carseats)  
summary(lm.fit1)

##   
## Call:  
## lm(formula = Sales ~ Advertising + Price, data = Carseats)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -7.9011 -1.5470 -0.0223 1.5361 6.3748   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 13.003427 0.606850 21.428 < 2e-16 \*\*\*  
## Advertising 0.123107 0.018079 6.809 3.64e-11 \*\*\*  
## Price -0.054613 0.005078 -10.755 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.399 on 397 degrees of freedom  
## Multiple R-squared: 0.2819, Adjusted R-squared: 0.2782   
## F-statistic: 77.91 on 2 and 397 DF, p-value: < 2.2e-16

#Regressiom Model 2 (Advertising & Urban)  
lm.fit2 <- lm(Sales ~ Advertising + Urban + Advertising \* Urban , data = Carseats)  
summary(lm.fit2)

##   
## Call:  
## lm(formula = Sales ~ Advertising + Urban + Advertising \* Urban,   
## data = Carseats)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -7.2142 -1.9486 -0.0989 1.7583 8.3511   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 6.67852 0.34847 19.165 < 2e-16 \*\*\*  
## Advertising 0.14267 0.03897 3.661 0.000285 \*\*\*  
## UrbanYes 0.07914 0.41825 0.189 0.850013   
## Advertising:UrbanYes -0.03842 0.04586 -0.838 0.402639   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.726 on 396 degrees of freedom  
## Multiple R-squared: 0.07499, Adjusted R-squared: 0.06798   
## F-statistic: 10.7 on 3 and 396 DF, p-value: 8.891e-07

#Outlier Checker, I used this line of code from assignment 4. I used this to further explain which model is prefered.   
plot(predict(lm.fit2), rstudent(lm.fit2))



1. Fit a regression model predicting Sales using Advertising and Price as predictors. Interpret the coefficients, the , and the Residual standard error from the regression (by explaining each in a few statements).

**Interpretation of Coefficients** *The Price coefficient suggests a relationship between price and sales given the low p-value(2e-16) of the t-statistic.The price coefficient(-0.054613) states a negative relationship between Price and Sales meaning that as Price increases there will be a equal decrease in sales, equivalent to the value of the price coefficient (-0.054613)*

*The Advertising coefficient suggests a relationship between advertising and sales given the low p-value (3.64e-11) of the t-statistic. The advertsing coefficient(0.123107) states a postive relationship between Advertising and Sales meaning that as Advertising increases there will be a equal increase in sales, equivalent to the value of the advertising coefficient (0.123107).*

**Interpretation of Residual Standard Error** *The residual standard error can be decribed as the measure of the quality of any linear regression fit. The value of the residual standard error is basically the average amount that the response will deviate from the true regression line. Degrees of freedom are indicated by the number of data points that went into the estimation of the paramenters used after taking into account these retrictions. In this model, we had 400 data points and 3 parameters.*

*For this model (lm.fit1), the residual standard error comes out to 2.402 on 397 degrees of freedom. Meaning that, the average distance of the data points from the fitted line is about 2.4%.*

**Interpretation of R Squared** *The R-Squared value of a regression model can be described as the amount of variance that can be explained with this particualur model. The Adjusted R-Squared value is considered to be more dependable calculation because it takes into account the number of varaibles added to the whole model. The Multiple R-Squared value does not take this factor into account hence that the more variables added to the model, the value of R-Squared will be higher regardless because of the increased amount of different variables.*

*In reguards to this model (lm.fit1), this discrepency can become an problem when looking at regression models with large amounts of different variables. The Adjusted R-Value in this model is 0.2782 which indicates that 27.82% of the variance can be explained through this regression model(lm.fit1).*

2.Fit a second model by adding Urban as an interactive variable with Advertising. Interpret the two new coefficients produced by adding this interaction to the Advertising variable that was already present from the first question, in a few statements.

**Interpretation of Coefficients** *The UrbanYes coefficient suggests a no relationship between UrbanYes and sales given the high p-value(0.850013) of the t-statistic.*

*The new interactive coefficent (Advertising:UrbanYes) suggests a relationship between these two varibles together with Sales given the p-value(0.402639) of the t-statistic. The Advertising:UrbanYes coefficent (-0.03842) indicates a negative relationship between Advertising:UrbanYes and Sales.*

**Interpretation of Residual Standard Error** *For this model (lm.fit2), the residual standard error comes out to 2.726 on 396 degrees of freedom. Meaning that, the average distance of the data points from the fitted line is about 2.7%.*

**Interpretation of R Squared** *In reguards to this model (lm.fit2), this discrepency can become an problem when looking at regression models with large amounts of different variables. The Adjusted R-Value in this model is 0.06798 which indicates that 6.798% of the variance can be explained through this regression model(lm.fit2).*

1. Which of these two models is preferable, and why? **Preferable Model** *After checking for outliers, there was 0 indication on any outlier values. So based on the Residual of Standard Error and R Squared values, the lm.fit1 model would be preferable compared to the lm.fit2 model.Both models fit the data similarly, with linear regression from lm.fit1 fitting the data slightly better due to a slightly lower residual of standard of error. From the summaries it was gathered that in the first model the RSE (Residual of Standard Error) was 2.402 on 396 degrees of freedom and the RSE on the second model was 2.726 on 396 degrees of freedom. Addtionally, the R Squared values varied as well. The first model (lm.fit1) had a adjusted r-squared value of 0.2782 which indicates that 27.82% of the variance can be explained through this regression model(lm.fit1), and the second model (lm.fit2) had a adjusted r-squared value 0.06798 which indicates that 6.798% of the variance can be explained through this regression model(lm.fit2).*

## Question 2:

You will need to load the core library for the course textbook and any other libraries you find suitable to answer the question:

data("Weekly", package = "ISLR")  
library("MASS")  
library("class")  
library("caret")

## Loading required package: lattice

## Loading required package: ggplot2

#Question 1 - Exploratory Data Analysis  
names(Weekly) #to see whats on the data frame

## [1] "Year" "Lag1" "Lag2" "Lag3" "Lag4" "Lag5"   
## [7] "Volume" "Today" "Direction"

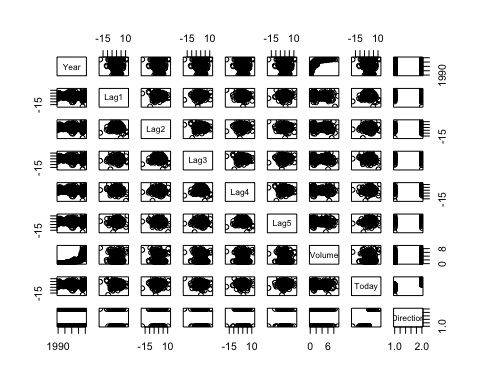
head(Weekly) #to see the first couple of rows

## Year Lag1 Lag2 Lag3 Lag4 Lag5 Volume Today Direction  
## 1 1990 0.816 1.572 -3.936 -0.229 -3.484 0.1549760 -0.270 Down  
## 2 1990 -0.270 0.816 1.572 -3.936 -0.229 0.1485740 -2.576 Down  
## 3 1990 -2.576 -0.270 0.816 1.572 -3.936 0.1598375 3.514 Up  
## 4 1990 3.514 -2.576 -0.270 0.816 1.572 0.1616300 0.712 Up  
## 5 1990 0.712 3.514 -2.576 -0.270 0.816 0.1537280 1.178 Up  
## 6 1990 1.178 0.712 3.514 -2.576 -0.270 0.1544440 -1.372 Down

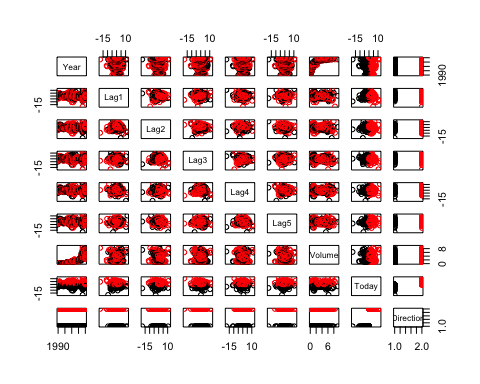
summary(Weekly) #a full summary of all the data

## 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   
## Min. :-18.1950 Min. :-18.1950 Min. :0.08747   
## 1st Qu.: -1.1580 1st Qu.: -1.1660 1st Qu.:0.33202   
## Median : 0.2380 Median : 0.2340 Median :1.00268   
## Mean : 0.1458 Mean : 0.1399 Mean :1.57462   
## 3rd Qu.: 1.4090 3rd Qu.: 1.4050 3rd Qu.:2.05373   
## Max. : 12.0260 Max. : 12.0260 Max. :9.32821   
## Today Direction   
## Min. :-18.1950 Down:484   
## 1st Qu.: -1.1540 Up :605   
## Median : 0.2410   
## Mean : 0.1499   
## 3rd Qu.: 1.4050   
## Max. : 12.0260

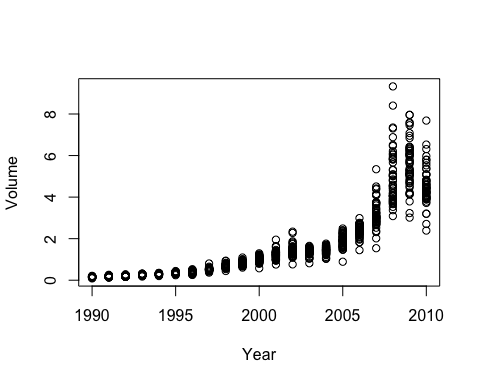
pairs(Weekly) #produces a scatterplot matrix data



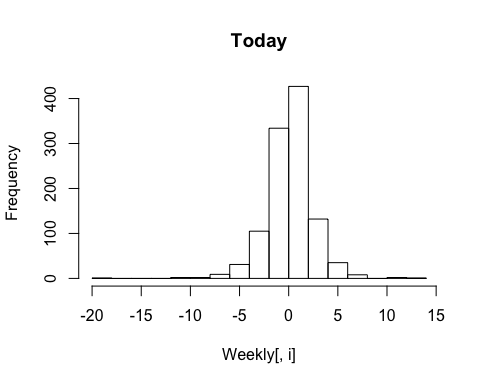
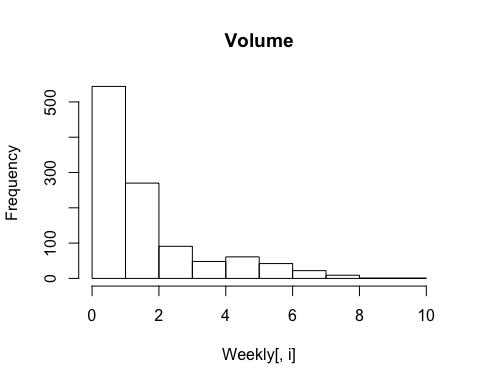
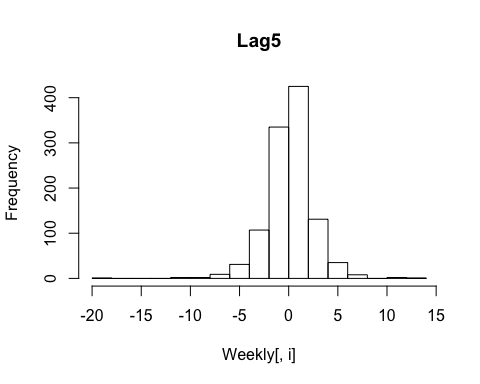
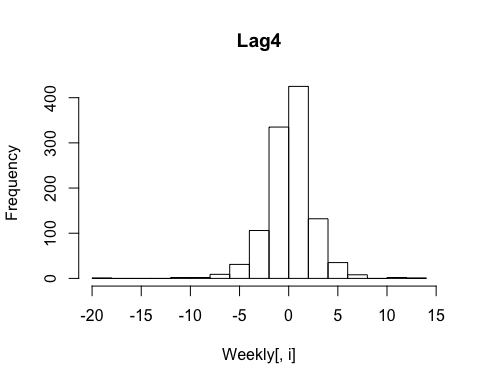
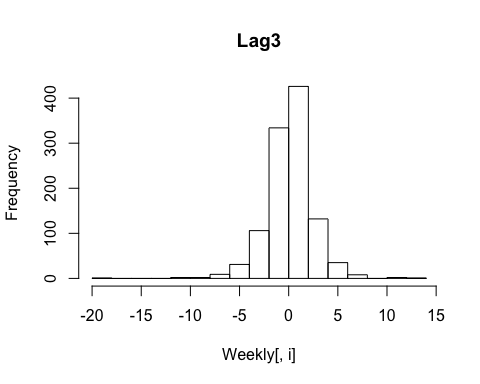
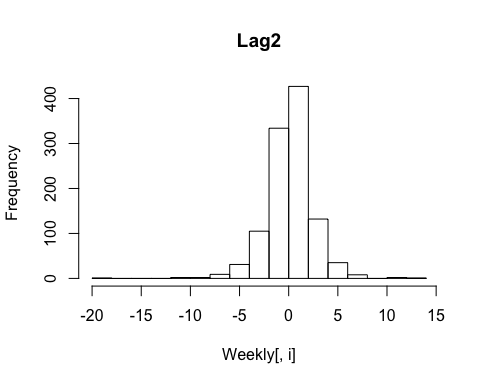
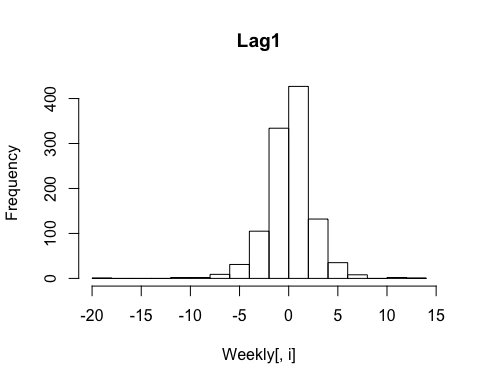
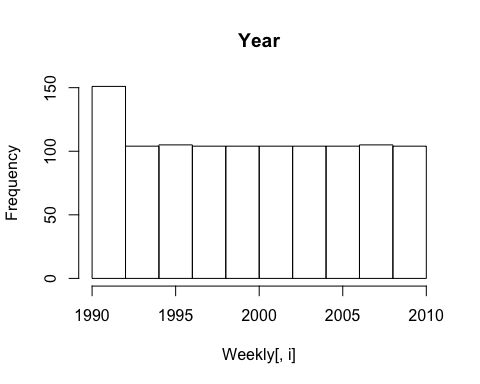
pairs(Weekly, col = Weekly$Direction) #produces a scatterplot matrix with the Direction variable



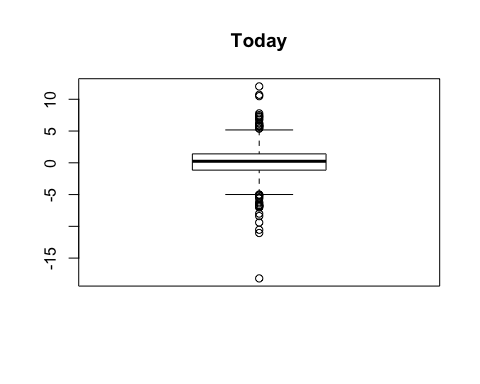
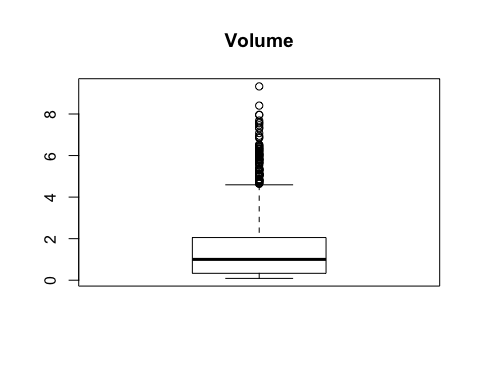
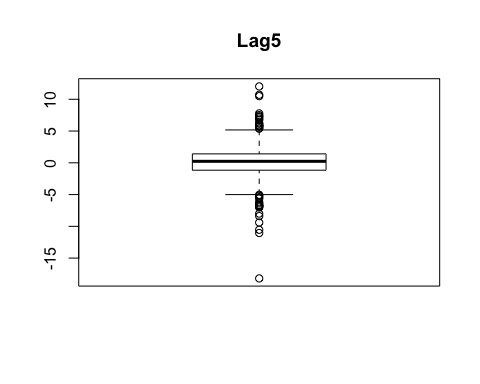
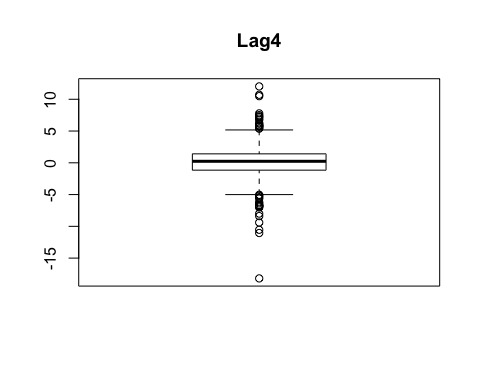
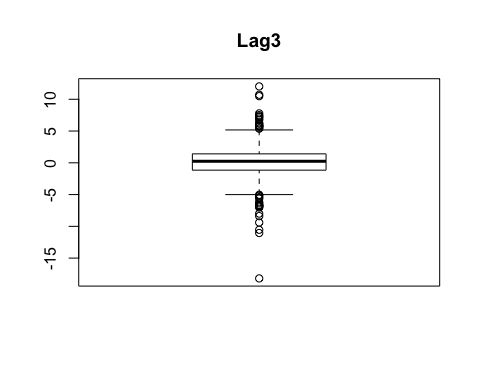
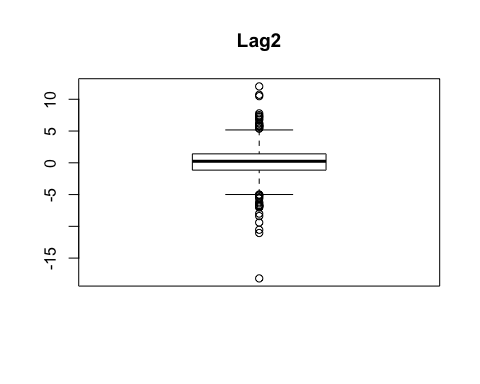
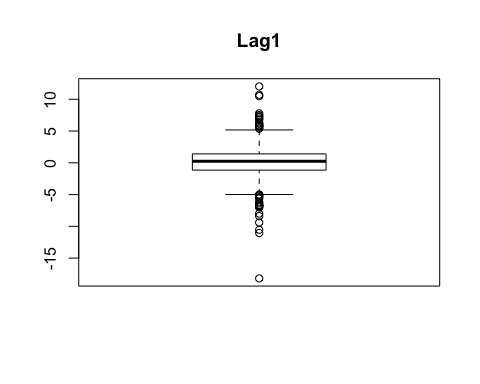
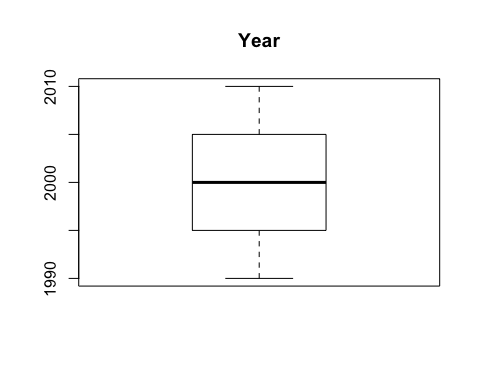
plot(Weekly$Year, Weekly$Volume,   
 xlab = "Year", ylab = "Volume") #produces side-by-side boxplots of Year and Volume



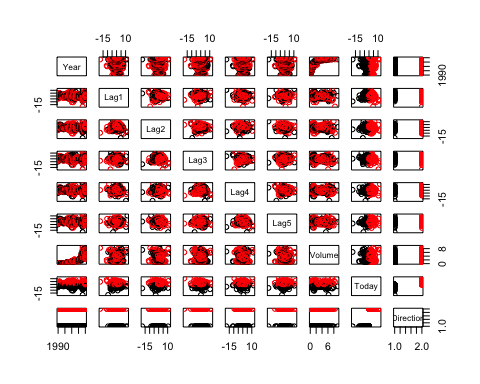
#par(mfrow = c(1,8)) #This line of code was creating a error  
##xlab = names(Weekly)[i]) -- I tried to do the xlab part to label the x-axis but it was not working  
for (i in 1:8) { #creates histograms  
 hist(Weekly[,i], main = names(Weekly)[i])   
}



for (i in 1:8) { #creates boxplots  
 boxplot(Weekly[,i], main = names(Weekly)[i])   
}



pairs(Weekly, col = Weekly$Direction)



#Question 2 - Logistic Regression  
#not splitted data  
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

#Training Data Set  
training = (Weekly$Year < 2009)  
glm.fitTrainingData <- glm(Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5 + Volume, data = Weekly, family = binomial, subset = training)  
summary(glm.fitTrainingData)

##   
## Call:  
## glm(formula = Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5 +   
## Volume, family = binomial, data = Weekly, subset = training)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.7186 -1.2498 0.9823 1.0841 1.4911   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 0.33258 0.09421 3.530 0.000415 \*\*\*  
## Lag1 -0.06231 0.02935 -2.123 0.033762 \*   
## Lag2 0.04468 0.02982 1.499 0.134002   
## Lag3 -0.01546 0.02948 -0.524 0.599933   
## Lag4 -0.03111 0.02924 -1.064 0.287241   
## Lag5 -0.03775 0.02924 -1.291 0.196774   
## Volume -0.08972 0.05410 -1.658 0.097240 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 1354.7 on 984 degrees of freedom  
## Residual deviance: 1342.3 on 978 degrees of freedom  
## AIC: 1356.3  
##   
## Number of Fisher Scoring iterations: 4

#Test Data Set  
testing = (Weekly$Year > 2008)  
glm.fitTestingData <- glm(Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5 + Volume, data = Weekly, family = binomial, subset = testing)  
summary(glm.fitTestingData)

##   
## Call:  
## glm(formula = Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5 +   
## Volume, family = binomial, data = Weekly, subset = testing)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.7110 -1.2719 0.8643 1.0041 1.4546   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) 0.47502 0.95259 0.499 0.618  
## Lag1 0.03944 0.06781 0.582 0.561  
## Lag2 0.09575 0.06916 1.384 0.166  
## Lag3 -0.03850 0.06743 -0.571 0.568  
## Lag4 -0.04248 0.06699 -0.634 0.526  
## Lag5 0.08877 0.06939 1.279 0.201  
## Volume -0.03158 0.18179 -0.174 0.862  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 141.04 on 103 degrees of freedom  
## Residual deviance: 136.54 on 97 degrees of freedom  
## AIC: 150.54  
##   
## Number of Fisher Scoring iterations: 4

#Question 3 - Confusion Matrix   
DirectionTest1 = Weekly$Direction[!training]  
WeeklyTest1 = Weekly[!training,]  
glm.fit2 = glm(Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5 + Volume, data = Weekly, family = binomial, subset = testing)  
glm.probs = predict(glm.fit2,WeeklyTest1,type="response")  
dim(WeeklyTest1)

## [1] 104 9

glm.prediction = rep("Down", 104)  
glm.prediction[glm.probs >.5] = "Up"  
table(glm.prediction,DirectionTest1)

## DirectionTest1  
## glm.prediction Down Up  
## Down 9 8  
## Up 34 53

#Accuracy -- Confusion Matrix  
sum(diag(table(glm.prediction,DirectionTest1)))/sum(table(glm.prediction,DirectionTest1))

## [1] 0.5961538

#Recall -- Confusion Matrix  
recall\_calc = recall(table(glm.prediction,DirectionTest1))  
recall\_calc

## [1] 0.2093023

#Precision -- Confusion Matrix  
precision\_calc = precision(table(glm.prediction,DirectionTest1))  
precision\_calc

## [1] 0.5294118

#F1 -- Confusion Matrix  
F1 <- (2 \* precision\_calc \* recall\_calc) / (precision\_calc + recall\_calc)

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

1. Perform exploratory data analysis of the Weekly data (produce some numerical and graphical summaries). Discuss any patterns that emerge.

*As expected, the correlations between the lag1 - lag5 coeffeicents and Today return values are near to zero. Basically, there appears to be little to zero correlation between Today’s return values and previous return values over time.*

*The only predominant correlation is between Year and Volume. By plotting the data it is evident that Volume is increasing over time. Hence, the average number of shares traded daily increased over time*

1. Fit a logistic regression with Direction as the response and different combinations of lag variables plus Volume as predictors. Use the period from 1990 to 2008 as your training set and 2009-2010 as your test set. Produce a summary of results.

**Do any of the predictors appear to be statistically significant in your training set? If so, which ones?** *The predictor Lag1 seems to be statistically significant due to its p-value of 0.033762*

1. From your test set, compute the confusion matrix, and calculate accuracy, precision, recall and F1.

* Explain what the confusion matrix is telling you about the types of mistakes made by logistic regression, and what can you learn from additional measures of fit like accuracy, precision, recall, and F1. *The confusion matrix is telling us about the performance of a classifcaiton model on a test data set which we created of whihch the true values are known.This matrix calculates a whole cross-tablulation of observed and predicted varibles.*

*I also calculated the accuracy, precision, recall, and the F1. From the accuracy calculation (0.596158), we learn how often the classifier is correct. From the precsion calculation, we know that it is a good measure to determine, when the costs of False Positive is high. From the recall calcualtion we are able to have the ability to find all the relevant cases within the set of data. Lastly, F1 is the function of precision and recall meaning it is a good measure to use if there is a need to seek a balance between the Precision and Recall calcualtions and there is an uneven class distribution within the data (large number of actual negatives).*

1. (Extra credit) Experiment with alternative classification methods. Present the results of your experiments reporting method, associated confusion matrix, and measures of fit on the test set like accuracy, precision, recall, and F1.