#M3 applied

#10(a)

In [64]:
```
library(ISLR)
summary(Weekly)
```

```
      Year           Lag1                Lag2                Lag3          
 Min.   :1990   Min.   :-18.1950   Min.   :-18.1950   Min.   :-18.195
0
 1st Qu.:1995   1st Qu.: -1.1540   1st Qu.: -1.1540   1st Qu.: -1.158
0
 Median :2000   Median :  0.2410   Median :  0.2410   Median :  0.241
0
 Mean   :2000   Mean   :  0.1506   Mean   :  0.1511   Mean   :  0.147
2
 3rd Qu.:2005   3rd Qu.:  1.4050   3rd Qu.:  1.4090   3rd Qu.:  1.409
0
 Max.   :2010   Max.   : 12.0260   Max.   : 12.0260   Max.   : 12.026
0
      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
```

In [65]:
```
summary(Smarket)
```

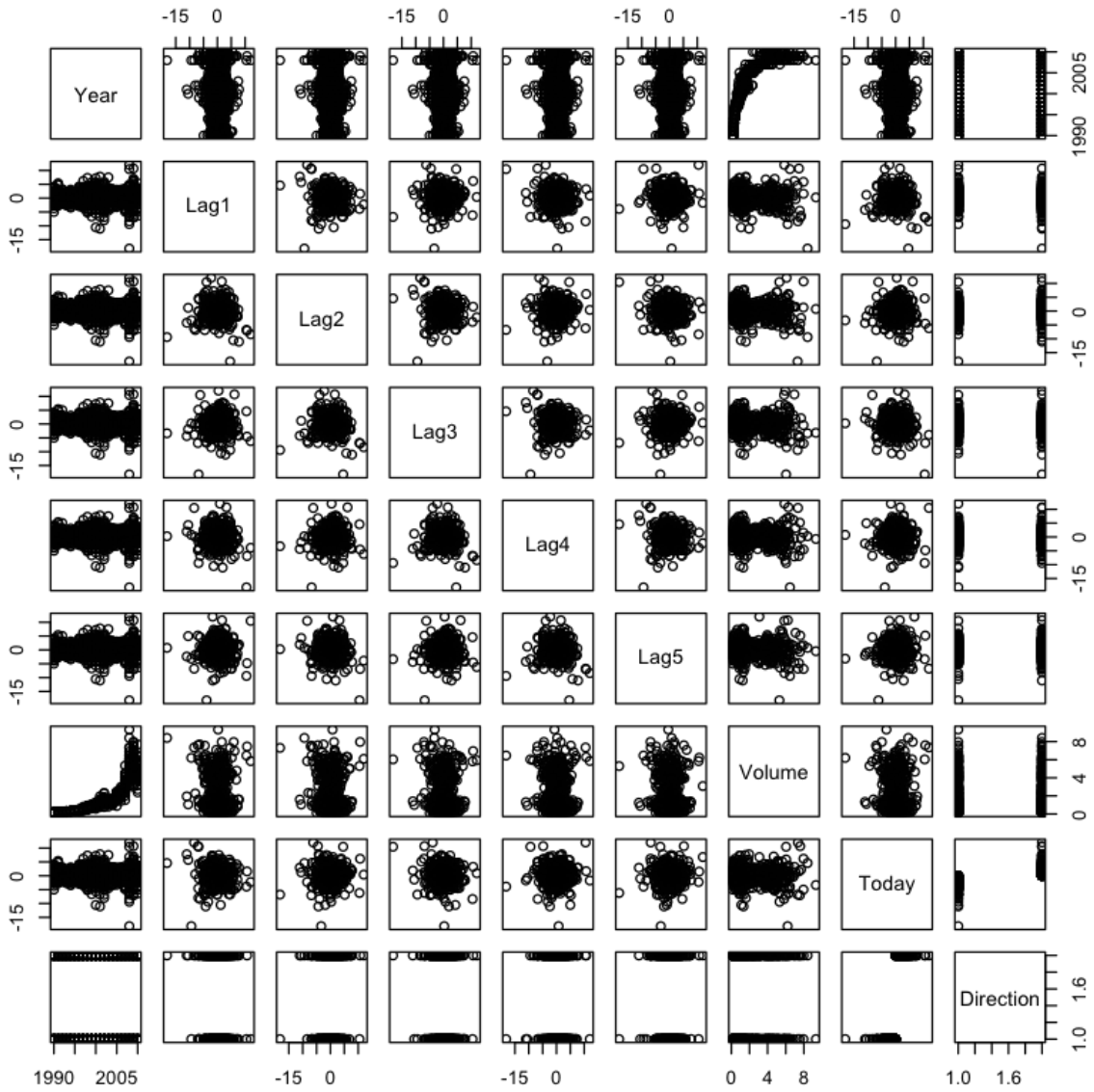
```
      Year           Lag1                Lag2                Lag3          
 Min.   :2001   Min.   :-4.922000   Min.   :-4.922000   Min.   :-4.92
2000
 1st Qu.:2002   1st Qu.:-0.639500   1st Qu.:-0.639500   1st Qu.:-0.64
0000
 Median :2003   Median : 0.039000   Median : 0.039000   Median : 0.03
8500
 Mean   :2003   Mean   : 0.003834   Mean   : 0.003919   Mean   : 0.00
1716
 3rd Qu.:2004   3rd Qu.: 0.596750   3rd Qu.: 0.596750   3rd Qu.: 0.59
6750
 Max.   :2005   Max.   : 5.733000   Max.   : 5.733000   Max.   : 5.73
3000
      Lag4                Lag5              Volume           Today          
 Min.   :-4.922000   Min.   :-4.92200   Min.   :0.3561   Min.   :-4.9
22000
 1st Qu.:-0.640000   1st Qu.:-0.64000   1st Qu.:1.2574   1st Qu.:-0.6
39500
 Median : 0.038500   Median : 0.03850   Median :1.4229   Median : 0.0
38500
 Mean   : 0.001636   Mean   : 0.00561   Mean   :1.4783   Mean   : 0.0
03138
 3rd Qu.: 0.596750   3rd Qu.: 0.59700   3rd Qu.:1.6417   3rd Qu.: 0.5
96750
 Max.   : 5.733000   Max.   : 5.73300   Max.   :3.1525   Max.   : 5.7
33000
 Direction
 Down:602
 Up  :648
```

In [66]:
```
names(Weekly)
dim(Weekly)
```

'Year' 'Lag1' 'Lag2' 'Lag3' 'Lag4' 'Lag5' 'Volume' 'Today' 'Direction'

1089 9

In [67]:
```
pairs(Weekly)
```



In [68]:
```
#Direction = Weekly$Direction
#Weekly$Direction = NULL
#Weekly$NumericDirection = as.numeric(Direction)
#Weekly$NumericDirection[Weekly$NumericDirection == 1] = -1
#Weekly$NumericDirection[Weekly$NumericDirection == 2] = +1
#Weekly.cor = cor(Weekly)
```

In [69]:
```
cor(Weekly[, -9])
#cor() creates matrix that contains all of the pairwise correlatic

#Year and Vol seems to have a relationship
```

	Year	Lag1	Lag2	Lag3	Lag4	Lag5	
Year	1.00000000	-0.032289274	-0.03339001	-0.03000649	-0.031127923	-0.030519101	0.84
Lag1	-0.03228927	1.000000000	-0.07485305	0.05863568	-0.071273876	-0.008183096	-0.06
Lag2	-0.03339001	-0.074853051	1.00000000	-0.07572091	0.058381535	-0.072499482	-0.08
Lag3	-0.03000649	0.058635682	-0.07572091	1.00000000	-0.075395865	0.060657175	-0.06
Lag4	-0.03112792	-0.071273876	0.05838153	-0.07539587	1.000000000	-0.075675027	-0.06
Lag5	-0.03051910	-0.008183096	-0.07249948	0.06065717	-0.075675027	1.000000000	-0.05
Volume	0.84194162	-0.064951313	-0.08551314	-0.06928771	-0.061074617	-0.058517414	1.00
Today	-0.03245989	-0.075031842	0.05916672	-0.07124364	-0.007825873	0.011012698	-0.03

#10(b)

In [70]:
```
#attach(Weekly)
#plot(Volume)
```

In [71]:
```
glm.fits=glm(Direction~Lag1+Lag2+Lag3+Lag4+Lag5+Volume,data=Weekly
coef(glm.fits)
summary(glm.fits)
#Lag 2 appears to have some statistical significance with a Pr(>|z
```

(Intercept)	0.266864141430795
Lag1	-0.0412689400271679
Lag2	0.0584416754635488
Lag3	-0.0160611438185425
Lag4	-0.0277902103879173
Lag5	-0.0144720643823032
Volume	-0.0227415314988368

```
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
```

#10(c)

In [72]:
```
dim(Weekly)
```

```
1089  9
```

In [73]:
```
summary(Weekly)
```

```
      Year           Lag1                Lag2                Lag3          
 Min.   :1990   Min.   :-18.1950   Min.   :-18.1950   Min.   :-18.195
0
 1st Qu.:1995   1st Qu.: -1.1540   1st Qu.: -1.1540   1st Qu.: -1.158
0
 Median :2000   Median :  0.2410   Median :  0.2410   Median :  0.241
0
 Mean   :2000   Mean   :  0.1506   Mean   :  0.1511   Mean   :  0.147
2
 3rd Qu.:2005   3rd Qu.:  1.4050   3rd Qu.:  1.4090   3rd Qu.:  1.409
0
 Max.   :2010   Max.   : 12.0260   Max.   : 12.0260   Max.   : 12.026
0
      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
```

In [74]:
```
dim(Weekly$Direction)
```

```
NULL
```

In [75]:
```
glm.probs=predict(glm.fits,type="response")
glm.pred=ifelse(glm.probs>.5,"Up","Down")
table(glm.pred,Weekly$Direction)

```

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

10(d)

In [76]:
```
#Logistic regression using only Lag2 as the predictor
#(since it is the most significant predictor)

train.year <- Weekly$Year %in% (1990:2008)
train = Weekly[train.year,]
test = Weekly[!train.year,]
fit2 = glm(Direction~Lag2, data=train, family=binomial)
fit2.prob = predict(fit2, test, type="response")
fit2.pred <- ifelse(fit2.prob > 0.5, "Up", "Down")
table(fit2.pred, test$Direction)

```

```
fit2.pred Down Up
     Down    9  5
     Up     34 56
```

In [77]:
```
mean(fit2.pred == test$Direction)
```

0.625

#10(e) LDA

In [78]:
```

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

```
fit.lda.pred Down Up
        Down    9  5
        Up     34 56
```

In [79]:
```
mean(fit.lda.pred == test$Direction)
#accuracy is 62%
```

0.625

#10(f) QDA

In [80]:
```
fit.qda = qda(Direction~Lag2, data=train)
fit.qda.pred = predict(fit.qda, test)$class
table(fit.qda.pred, test$Direction)
mean(fit.qda.pred == test$Direction)
```

```
fit.qda.pred Down Up
        Down    0  0
        Up     43 61
```

0.586538461538462

In [81]:
```
#accuracy 58%
```

#10(g)

In [82]:
```
#require(class)
library(class)
set.seed(1)
train.X = as.matrix(train$Lag2)
test.X = as.matrix(test$Lag2)
knn.pred = knn(train.X, test.X, train$Direction, k=1)
table(knn.pred, test$Direction)
mean(knn.pred == test$Direction)
```

```
knn.pred Down Up
    Down   21 30
    Up     22 31
```

0.5

In [83]:
```
#accuracy 0.5
```

In [84]:
```
### # 10(h)
#Logistic Regression and LDA produced the best results
```

#10(i)

In [85]:
```
knn.pred = knn(train.X, test.X, train$Direction, k=5)
table(knn.pred, test$Direction)
mean(knn.pred == test$Direction)
knn.pred = knn(train.X, test.X, train$Direction, k=15)
table(knn.pred, test$Direction)
mean(knn.pred == test$Direction)
knn.pred = knn(train.X, test.X, train$Direction, k=30)
table(knn.pred, test$Direction)
mean(knn.pred == test$Direction)
knn.pred = knn(train.X, test.X, train$Direction, k=50)
table(knn.pred, test$Direction)
mean(knn.pred == test$Direction)
```

```
knn.pred Down Up
    Down   15 20
    Up     28 41
```

0.538461538461538

```
knn.pred Down Up
    Down   20 20
    Up     23 41
```

0.586538461538462

```
knn.pred Down Up
    Down   19 23
    Up     24 38
```

0.548076923076923

```
knn.pred Down Up
    Down   20 22
    Up     23 39
```

0.567307692307692

In [86]:
```
#higher value of K gives best results, predictor-lag2
```

#13

In [87]:
```
library(MASS)
summary(Boston)
```

```
      crim                zn              indus            chas        
 Min.   : 0.00632   Min.   :  0.00   Min.   : 0.46   Min.   :0.00000  
 1st Qu.: 0.08204   1st Qu.:  0.00   1st Qu.: 5.19   1st Qu.:0.00000  
 Median : 0.25651   Median :  0.00   Median : 9.69   Median :0.00000  
 Mean   : 3.61352   Mean   : 11.36   Mean   :11.14   Mean   :0.06917  
 3rd Qu.: 3.67708   3rd Qu.: 12.50   3rd Qu.:18.10   3rd Qu.:0.00000  
 Max.   :88.97620   Max.   :100.00   Max.   :27.74   Max.   :1.00000  
      nox               rm              age              dis         
 Min.   :0.3850   Min.   :3.561   Min.   :  2.90   Min.   : 1.130  
 1st Qu.:0.4490   1st Qu.:5.886   1st Qu.: 45.02   1st Qu.: 2.100  
 Median :0.5380   Median :6.208   Median : 77.50   Median : 3.207  
 Mean   :0.5547   Mean   :6.285   Mean   : 68.57   Mean   : 3.795  
 3rd Qu.:0.6240   3rd Qu.:6.623   3rd Qu.: 94.08   3rd Qu.: 5.188  
 Max.   :0.8710   Max.   :8.780   Max.   :100.00   Max.   :12.127  
      rad              tax            ptratio          black       
 Min.   : 1.000   Min.   :187.0   Min.   :12.60   Min.   :  0.32  
 1st Qu.: 4.000   1st Qu.:279.0   1st Qu.:17.40   1st Qu.:375.38  
 Median : 5.000   Median :330.0   Median :19.05   Median :391.44  
 Mean   : 9.549   Mean   :408.2   Mean   :18.46   Mean   :356.67  
 3rd Qu.:24.000   3rd Qu.:666.0   3rd Qu.:20.20   3rd Qu.:396.23  
 Max.   :24.000   Max.   :711.0   Max.   :22.00   Max.   :396.90  
     lstat            medv      
 Min.   : 1.73   Min.   : 5.00  
 1st Qu.: 6.95   1st Qu.:17.02  
 Median :11.36   Median :21.20  
 Mean   :12.65   Mean   :22.53  
 3rd Qu.:16.95   3rd Qu.:25.00  
 Max.   :37.97   Max.   :50.00  
```

In [88]:
```
attach(Boston)
crime01 = rep(0, length(crim))
crime01[crim > median(crim)] = 1
Boston = data.frame(Boston, crime01)
```

In [89]:
```
train = 1:(dim(Boston)[1]/2)
test = (dim(Boston)[1]/2 + 1):dim(Boston)[1]
Boston.train = Boston[train, ]
Boston.test = Boston[test, ]
crime01.test = crime01[test]
```

In [90]:
```
# logistic regression
glm.fit = glm(crime01 ~ . - crime01 - crim, data = Boston, family
    subset = train)
```

```
Warning message:
“glm.fit: fitted probabilities numerically 0 or 1 occurred”
```

In [91]:
```
glm.probs = predict(glm.fit, Boston.test, type = "response")
glm.pred = rep(0, length(glm.probs))
glm.pred[glm.probs > 0.5] = 1
mean(glm.pred != crime01.test)

#18.2% test error rate(appx)
```

```
0.181818181818182
```

In [92]:
```
glm.fit = glm(crime01 ~ . - crime01 - crim - chas - tax, data = Bo
    subset = train)
```

```
Warning message:
“glm.fit: fitted probabilities numerically 0 or 1 occurred”
```

In [93]:
```
glm.probs = predict(glm.fit, Boston.test, type = "response")
glm.pred = rep(0, length(glm.probs))
glm.pred[glm.probs > 0.5] = 1
mean(glm.pred != crime01.test)

#18.6% test error rate
```

```
0.185770750988142
```

In [94]:
```
# LDA
lda.fit = lda(crime01 ~ . - crime01 - crim, data = Boston, subset
lda.pred = predict(lda.fit, Boston.test)
mean(lda.pred$class != crime01.test)

#13.4% test error rate
```

```
0.134387351778656
```

In [95]:
```
lda.fit = lda(crime01 ~ . - crime01 - crim - chas - tax, data = Bo
lda.pred = predict(lda.fit, Boston.test)
mean(lda.pred$class != crime01.test)

#12.3%test error rate
```

```
0.122529644268775
```

In [96]:
```
lda.fit = lda(crime01 ~ . - crime01 - crim - chas - tax - lstat -
    data = Boston, subset = train)
lda.pred = predict(lda.fit, Boston.test)
mean(lda.pred$class != crime01.test)

#11.9% test error rate
```

0.118577075098814

In [97]:
```
# KNN
library(class)
train.X = cbind(zn, indus, chas, nox, rm, age, dis, rad, tax, ptra
    lstat, medv)[train, ]
test.X = cbind(zn, indus, chas, nox, rm, age, dis, rad, tax, ptrat
    lstat, medv)[test, ]
train.crime01 = crime01[train]
set.seed(1)
# KNN(k=1)
knn.pred = knn(train.X, test.X, train.crime01, k = 1)
mean(knn.pred != crime01.test)

#45.8% test error rate
```

0.458498023715415

In [98]:
```
# KNN(k=10)
knn.pred = knn(train.X, test.X, train.crime01, k = 10)
mean(knn.pred != crime01.test)

#11.1% test error rate
```

0.118577075098814

In [99]:
```
# KNN(k=100)
knn.pred = knn(train.X, test.X, train.crime01, k = 100)
mean(knn.pred != crime01.test)

#49% test error rate
```

0.490118577075099

In [100]:

```
# KNN(k=10) with subset of variables
train.X = cbind(zn, nox, rm, dis, rad, ptratio, black, medv)[train
test.X = cbind(zn, nox, rm, dis, rad, ptratio, black, medv)[test,
knn.pred = knn(train.X, test.X, train.crime01, k = 10)
mean(knn.pred != crime01.test)

#27.8% test error rate
```

```
0.272727272727273
```