Homework 3

Kris Hanus, Laura Glathar, Arkya Rakshit, Jace Crist, Brandon Dlugosz
University of Nebraska at Omaha

BSAD 8700 - Business Analytics
Due: February 2, 2015

ANSWER FOR 10:

(a)
```
tail(Weekly, 1)
```

```
##      Year  Lag1  Lag2  Lag3  Lag4   Lag5   Volume Today Direction
## 1089 2010 1.034 0.283 1.281 2.969 -0.861 2.707105 0.069        Up
```

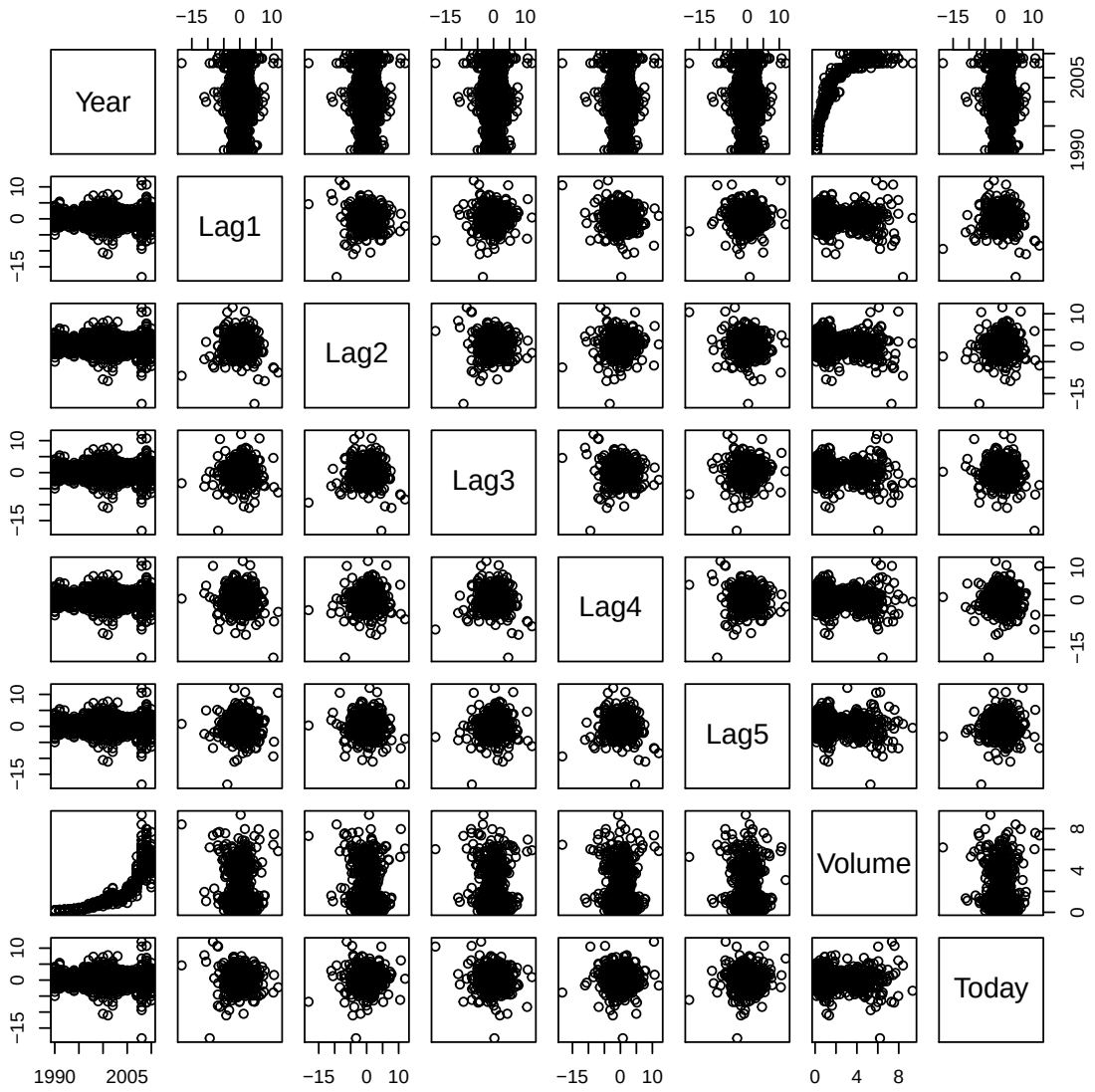
```
summary(Weekly)
```

```
##       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       
##  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
```

```
data1<-Weekly[,1:8]
attach(Weekly)
cor(data1)
```

```
##                Year         Lag1        Lag2        Lag3         Lag4
## Year    1.00000000 -0.032289274 -0.03339001 -0.03000649 -0.031127923
## Lag1   -0.03228927  1.000000000 -0.07485305  0.05863568 -0.071273876
## Lag2   -0.03339001 -0.074853051  1.00000000 -0.07572091  0.058381535
```

```
## Lag3   -0.03000649  0.058635682 -0.07572091  1.00000000 -0.075395865
## Lag4   -0.03112792 -0.071273876  0.05838153 -0.07539587  1.000000000
## Lag5   -0.03051910 -0.008183096 -0.07249948  0.06065717 -0.075675027
## Volume  0.84194162 -0.064951313 -0.08551314 -0.06928771 -0.061074617
## Today  -0.03245989 -0.075031842  0.05916672 -0.07124364 -0.007825873
##                Lag5      Volume        Today
## Year   -0.030519101  0.84194162 -0.032459894
## Lag1   -0.008183096 -0.06495131 -0.075031842
## Lag2   -0.072499482 -0.08551314  0.059166717
## Lag3    0.060657175 -0.06928771 -0.071243639
## Lag4   -0.075675027 -0.06107462 -0.007825873
## Lag5    1.000000000 -0.05851741  0.011012698
## Volume -0.058517414  1.00000000 -0.033077783
## Today   0.011012698 -0.03307778  1.000000000

pairs(data1)
```



There are a few interesting places of correlation. Primarily, with Volume and Year. Other wise, it is observable that each of the Lags are clustered, but it is difficult to observe other relationships.

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

```
##
## 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

coef(glm.fit)

## (Intercept)        Lag1        Lag2        Lag3        Lag4        Lag5
##  0.26686414 -0.04126894  0.05844168 -0.01606114 -0.02779021 -0.01447206
##      Volume
## -0.02274153
```

The only predictors which have significance are the intercept and Lag2. Lag2 is between 95% and 99% significant. The Intercept is 99% and 99.9%.

(c)
```
contrasts(Direction)

##      Up
## Down  0
## Up    1

glm.pred=rep("Down", 1089)
glm.probs=predict(glm.fit,type="response")
glm.probs[1:10]

##         1         2         3         4         5         6         7
## 0.6086249 0.6010314 0.5875699 0.4816416 0.6169013 0.5684190 0.5786097
##         8         9        10
## 0.5151972 0.5715200 0.5554287

glm.pred[glm.probs>0.5]<-"Up"
table(glm.pred,Direction)

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

```
557/(557+430)

## [1] 0.5643364

430/(557+430)

## [1] 0.4356636

48/(48+54)

## [1] 0.4705882
```

The confusion matrix shows that on days when logistic regression predicts an increase in the market, it has a 56.4% accuracy rate. The error rate is 43.6% for predicting Up and is actually Down. The error rate is 47.1% for predicting Down and is actually Up.

(d)
```
glm.fit2=glm(Direction~Lag2,data=Weekly, family = binomial)
summary(glm.fit2)

##
## Call:
## glm(formula = Direction ~ Lag2, family = binomial, data = Weekly)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.564  -1.267   1.008   1.086   1.386
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.21473    0.06123   3.507 0.000453 ***
## Lag2         0.06279    0.02636   2.382 0.017230 *
## ---
## 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: 1490.4  on 1087  degrees of freedom
## AIC: 1494.4
##
## Number of Fisher Scoring iterations: 4

coef(glm.fit2)

## (Intercept)        Lag2
##  0.21473151  0.06279058

contrasts(Direction)

##      Up
## Down  0
## Up    1
```

```
glm.pred2=rep("Down", 1089)
glm.probs2=predict(glm.fit2,type="response")
glm.probs2[1:10]

##         1         2         3         4         5         6         7
## 0.5777243 0.5661029 0.5492840 0.5132426 0.6071571 0.5644982 0.5716776
##         8         9        10
## 0.5321015 0.5659641 0.5541137

glm.pred2[glm.probs2>0.5]<-"Up"
table(glm.pred2,Direction)

##          Direction
## glm.pred2 Down  Up
##      Down   33  26
##      Up    451 579

579/(579+451)

## [1] 0.5621359

451/(579+451)

## [1] 0.4378641

26/(26+33)

## [1] 0.440678
```

The confusion matrix shows that on days when logistic regression predicts an increase in the market, it has a 56.2% accuracy rate. The error rate is 43.8% for predicting Up and is actually Down. The error rate is 44.1% for predicting Down and is actually Up.

ANSWER FOR 11:

(a)
```
#install.packages("RCurl")
library(RCurl)
## Loading required package:  bitops

dataset<-getURL(
    'https://raw.githubusercontent.com/Jwcrist/BusA/master/homeworks/Assignment%204/Auto.csv',
    ssl.verifypeer=0L, followlocation=1L)
dataset1<-read.csv(text=dataset)
head(dataset1,3)

##   mpg cylinders displacement horsepower weight acceleration year origin
## 1  18         8          307        130   3504         12.0   70      1
## 2  15         8          350        165   3693         11.5   70      1
## 3  18         8          318        150   3436         11.0   70      1
##                        name mpg01
```

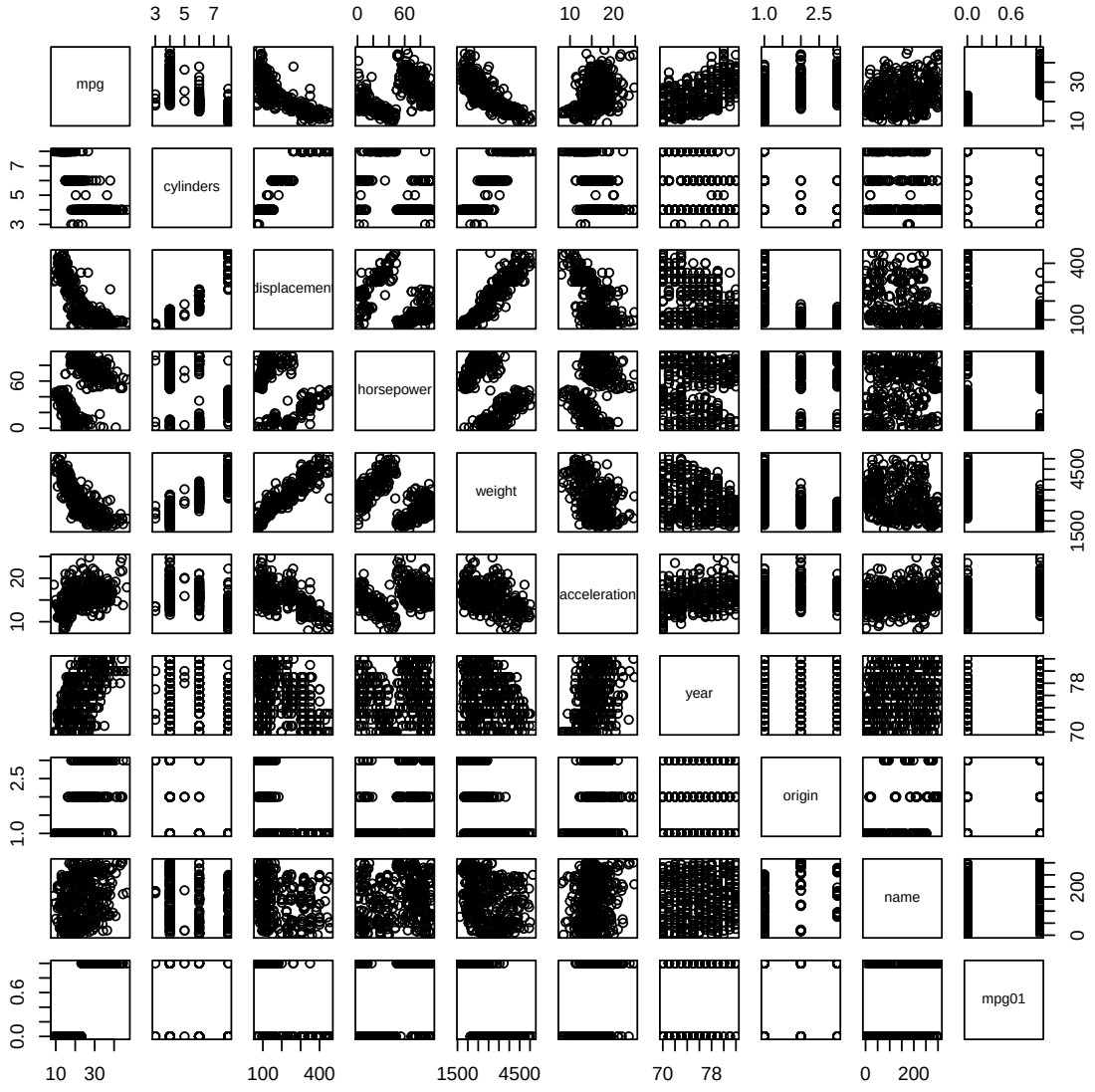
```
## 1 chevrolet chevelle malibu     0
## 2         buick skylark 320     0
## 3        plymouth satellite     0
```

As can be shown above we have created our the requested column.

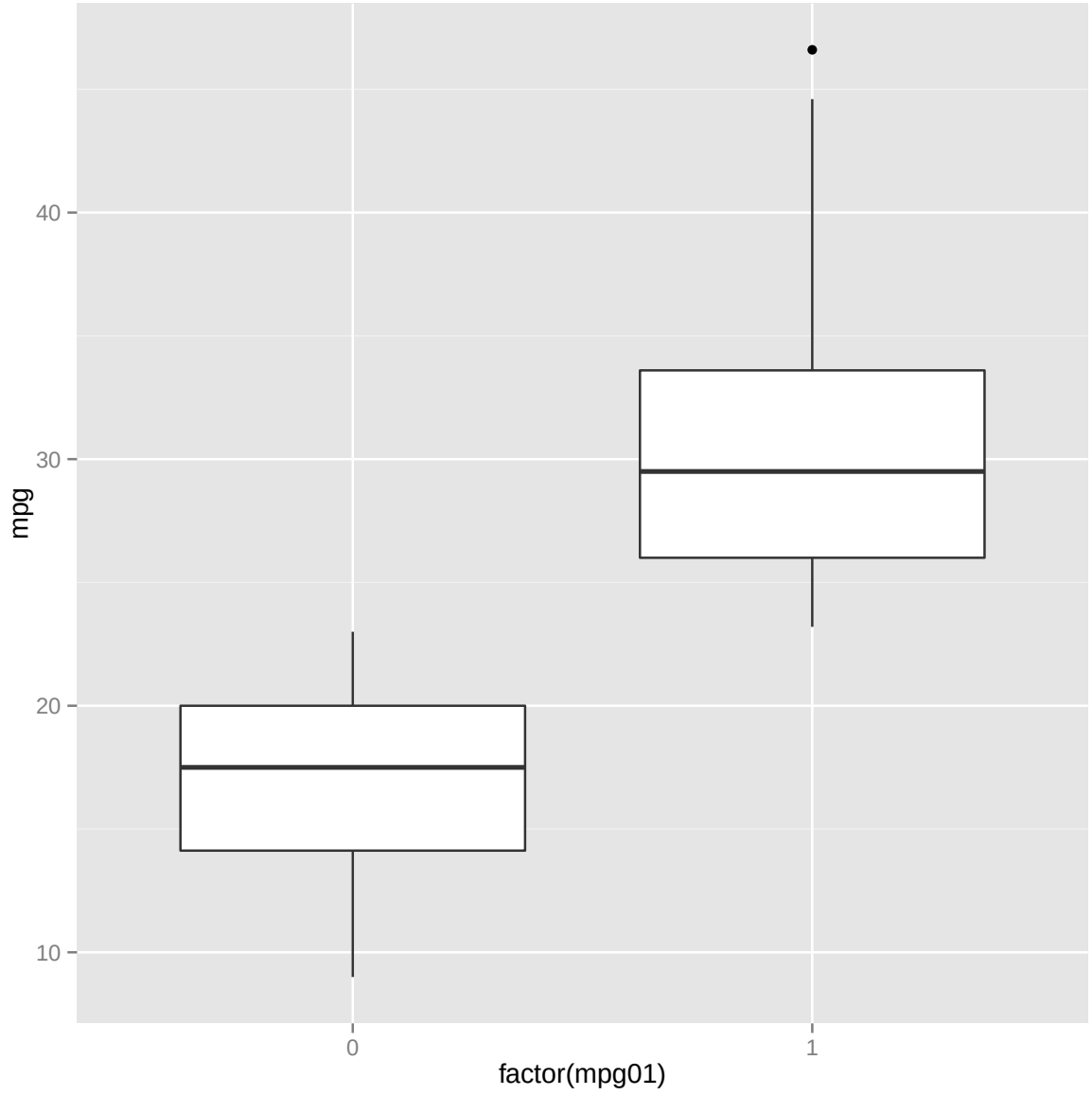
(b)
```
library(dplyr)
## Warning:  package 'dplyr' was built under R version 3.1.2
##
## Attaching package:  'dplyr'
##
## The following object is masked from 'package:MASS':
##
##    select
##
## The following object is masked from 'package:stats':
##
##    filter
##
## The following objects are masked from 'package:base':
##
##    intersect, setdiff, setequal, union

dataset2<-dataset1 %>%
  select(mpg, cylinders, displacement, as.numeric(horsepower), weight, acceleration, year, origin, m
dataset2$horsepower <- gsub("?",NA,dataset2$horsepower, fixed = TRUE)
dataset2$horsepower <- as.numeric(dataset2$horsepower)
str(dataset2)

## 'data.frame': 397 obs. of  9 variables:
##  $ mpg         : num  18 15 18 16 17 15 14 14 14 15 ...
##  $ cylinders   : int  8 8 8 8 8 8 8 8 8 8 ...
##  $ displacement: num  307 350 318 304 302 429 454 440 455 390 ...
##  $ horsepower  : num  130 165 150 150 140 198 220 215 225 190 ...
##  $ weight      : int  3504 3693 3436 3433 3449 4341 4354 4312 4425 3850 ...
##  $ acceleration: num  12 11.5 11 12 10.5 10 9 8.5 10 8.5 ...
##  $ year        : int  70 70 70 70 70 70 70 70 70 70 ...
##  $ origin      : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ mpg01       : int  0 0 0 0 0 0 0 0 0 0 ...

cor(na.omit(dataset2))

##                     mpg  cylinders displacement horsepower     weight
## mpg           1.0000000 -0.7776175   -0.8051269 -0.7784268 -0.8322442
## cylinders    -0.7776175  1.0000000    0.9508233  0.8429834  0.8975273
## displacement -0.8051269  0.9508233    1.0000000  0.8972570  0.9329944
## horsepower   -0.7784268  0.8429834    0.8972570  1.0000000  0.8645377
## weight       -0.8322442  0.8975273    0.9329944  0.8645377  1.0000000
## acceleration  0.4233285 -0.5046834   -0.5438005 -0.6891955 -0.4168392
## year          0.5805410 -0.3456474   -0.3698552 -0.4163615 -0.3091199
## origin        0.5652088 -0.5689316   -0.6145351 -0.4551715 -0.5850054
## mpg01         0.8404525 -0.7382676   -0.7372797 -0.6492595 -0.7457191
```

```
##               acceleration       year     origin      mpg01
## mpg              0.4233285  0.5805410  0.5652088  0.8404525
## cylinders       -0.5046834 -0.3456474 -0.5689316 -0.7382676
## displacement    -0.5438005 -0.3698552 -0.6145351 -0.7372797
## horsepower      -0.6891955 -0.4163615 -0.4551715 -0.6492595
## weight          -0.4168392 -0.3091199 -0.5850054 -0.7457191
## acceleration     1.0000000  0.2903161  0.2127458  0.3230383
## year             0.2903161  1.0000000  0.1815277  0.4537035
## origin           0.2127458  0.1815277  1.0000000  0.5154393
## mpg01            0.3230383  0.4537035  0.5154393  1.0000000
```

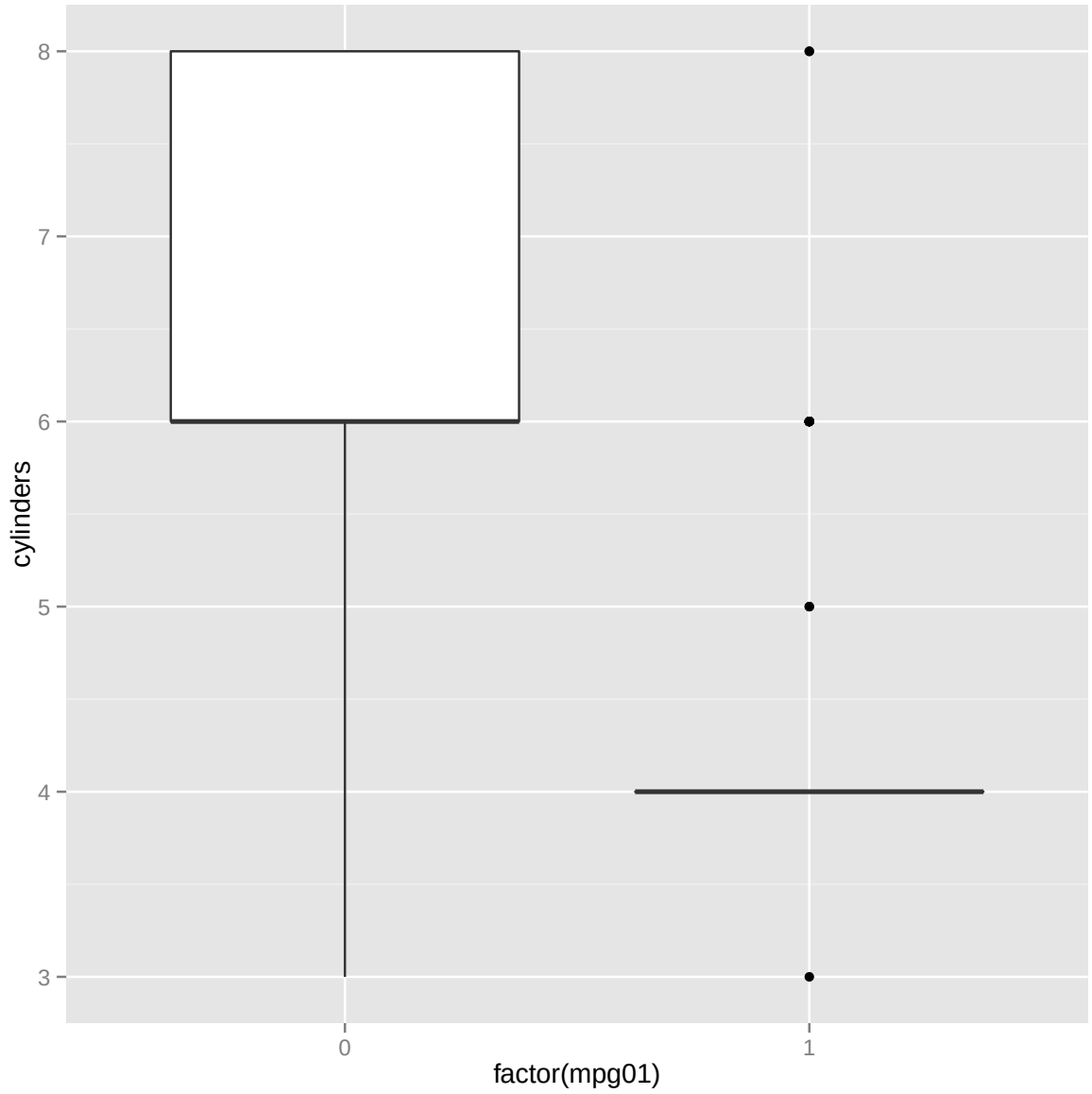
```
pairs(dataset1)
```



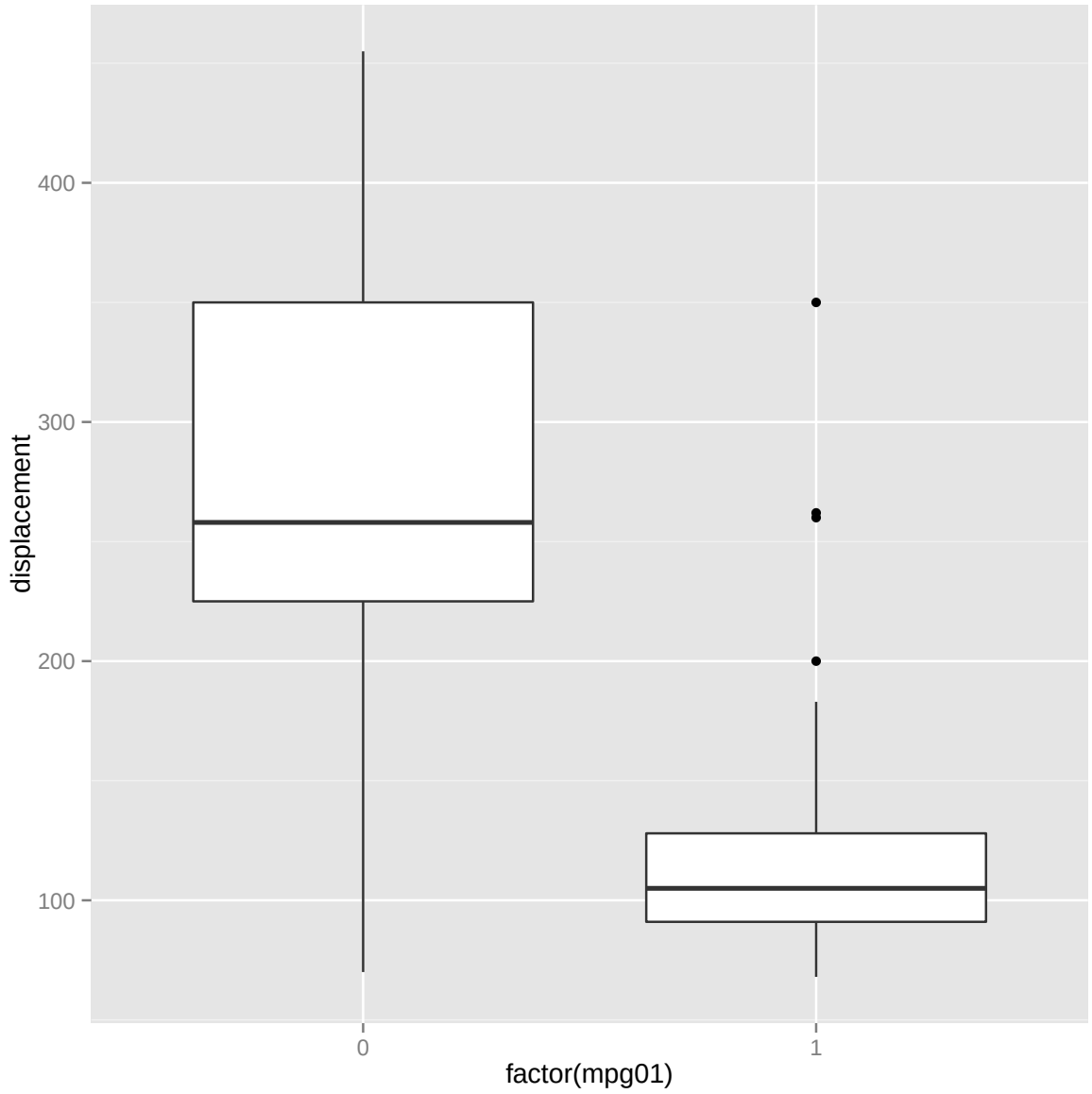
```
library(ggplot2)
ggplot(dataset1, aes(x=factor(mpg01), y=mpg))+geom_boxplot()
```



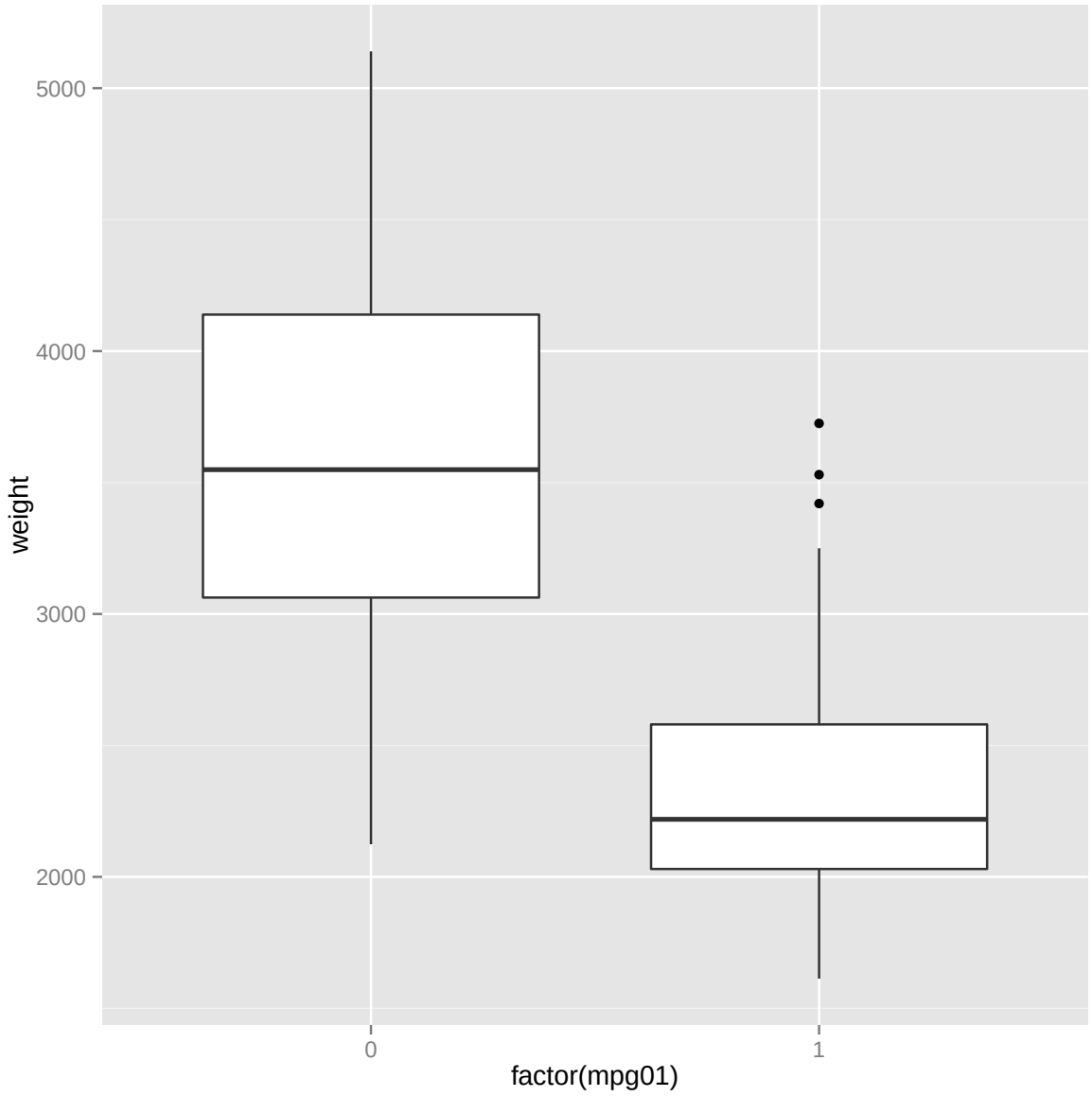

```
ggplot(dataset1, aes(x=factor(mpg01), y=cylinders))+geom_boxplot()
```



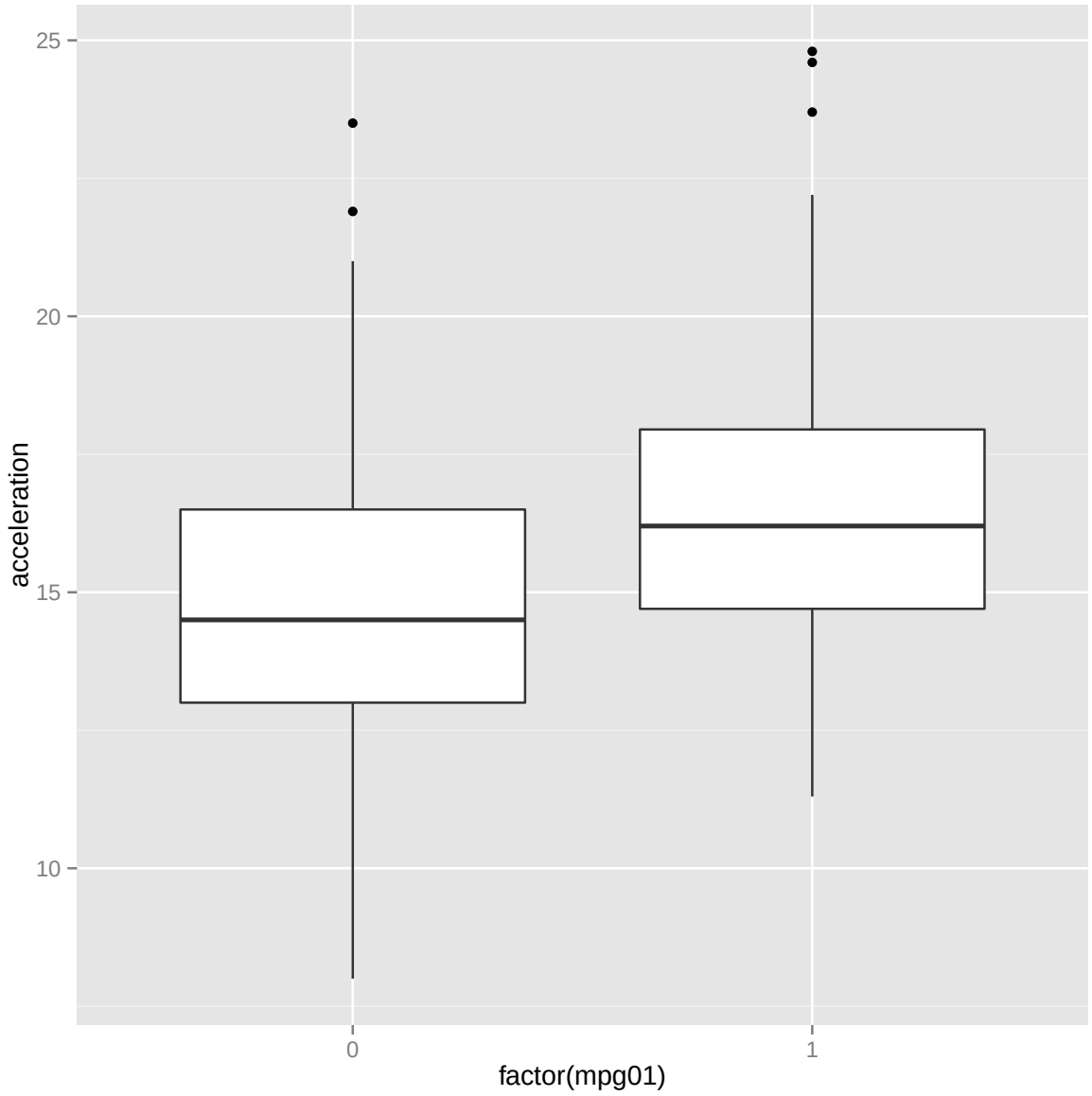

```
ggplot(dataset1, aes(x=factor(mpg01), y=displacement))+geom_boxplot()
```



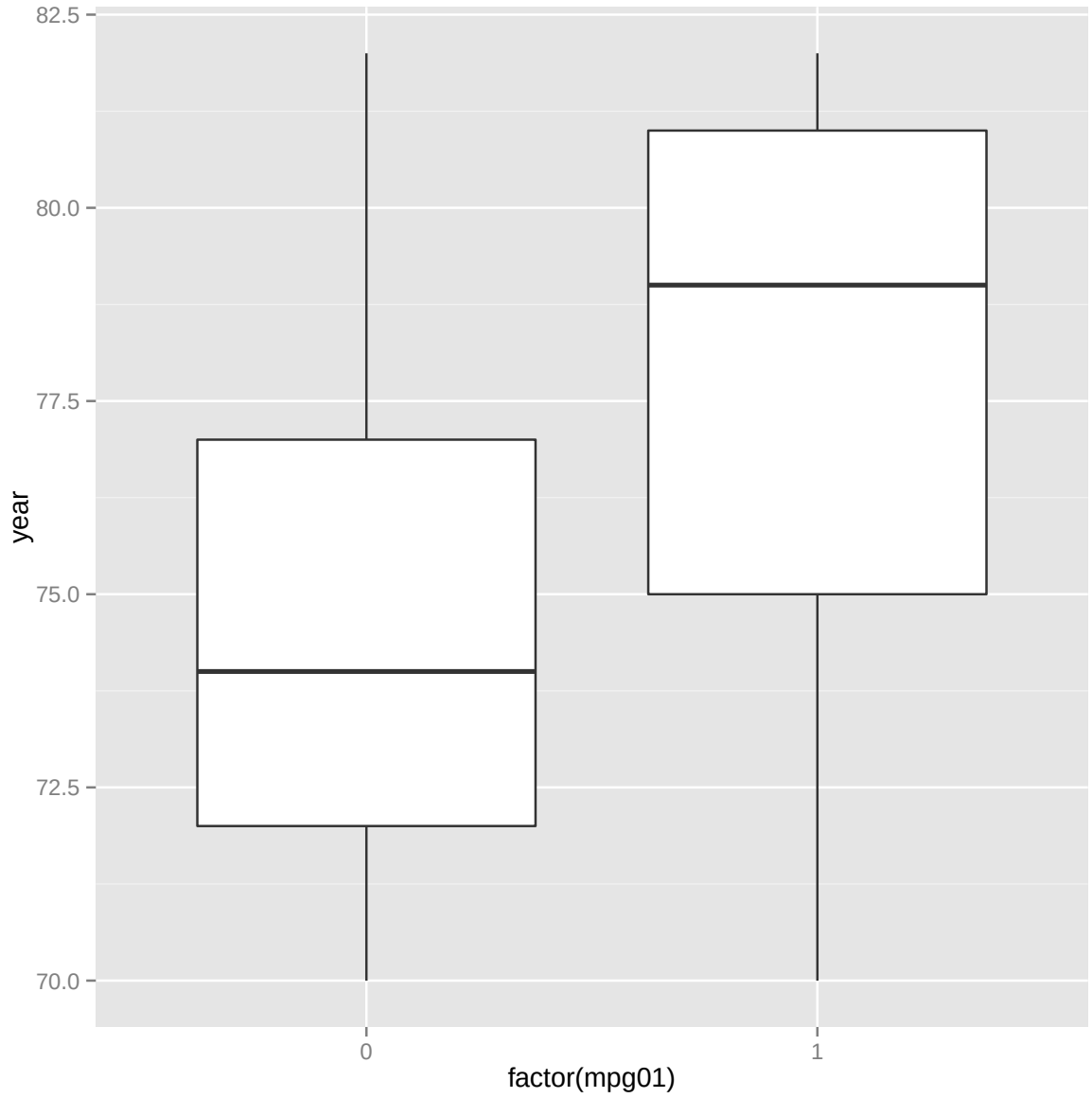

```
ggplot(dataset1, aes(x=factor(mpg01), y=weight))+geom_boxplot()
```




```
ggplot(dataset1, aes(x=factor(mpg01), y=acceleration))+geom_boxplot()
```




```
ggplot(dataset1, aes(x=factor(mpg01), y=year))+geom_boxplot()
```



There is too many layers to observe anything of significance in horsepower. Acceleration and year may have be significantly different from eachother. All the other plots show that there is probable differences between our binary factors.

(c)
```
tail(dataset1)

##     mpg cylinders displacement horsepower weight acceleration year origin
## 392  27         4          151         90   2950         17.3   82      1
## 393  27         4          140         86   2790         15.6   82      1
## 394  44         4           97         52   2130         24.6   82      2
## 395  32         4          135         84   2295         11.6   82      1
## 396  28         4          120         79   2625         18.6   82      1
## 397  31         4          119         82   2720         19.4   82      1
##                 name mpg01
## 392 chevrolet camaro     1
```

```
## 393  ford mustang gl     1
## 394         vw pickup     1
## 395    dodge rampage     1
## 396      ford ranger     1
## 397       chevy s-10     1

trainingdata<-dataset1[1:318,]
testdata<-dataset1[79,]
```

A 80% training data set was choosen to be used inorder to prove against our test data.

(d) Do Not Do

(e) Do Not Do

(f)
```
attach(dataset2)
## The following object is masked from package:ggplot2:
##
##     mpg

glm.fit3=lm(mpg01~mpg+cylinders+horsepower,data=dataset2, family = binomial)
summary(glm.fit3)

##
## Call:
## lm(formula = mpg01 ~ mpg + cylinders + horsepower, data = dataset2,
##     family = binomial)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.6527 -0.1661  0.0257  0.1789  0.8012
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.382820   0.129915  -2.947  0.00341 **
## mpg          0.047609   0.002848  16.715  < 2e-16 ***
## cylinders   -0.102064   0.015208  -6.711 6.86e-11 ***
## horsepower   0.002892   0.000675   4.284 2.32e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2575 on 388 degrees of freedom
##   (5 observations deleted due to missingness)
## Multiple R-squared:  0.737,Adjusted R-squared:  0.7349
## F-statistic: 362.4 on 3 and 388 DF,  p-value: < 2.2e-16

coef(glm.fit3)

##  (Intercept)          mpg    cylinders   horsepower
## -0.382820390  0.047608751 -0.102064129  0.002891938

glm.pred3=rep("0", 397)
glm.probs3=predict(glm.fit3,type="response")
glm.probs3[1:10]
```

```
##            1            2            3            4            5
##  0.033576028 -0.008032393  0.091414788 -0.003802713  0.014886658
##            6            7            8            9           10
##  0.087401562  0.103415448  0.088955758  0.117875138  0.064266058

glm.pred3[glm.probs3>0.5]<-"1"
table(glm.pred3,mpg01)

##          mpg01
## glm.pred3   0   1
##         0 158  43
##         1  48 148

(158+148)/397

## [1] 0.7707809

43/(158+43)

## [1] 0.2139303

48/(148+48)

## [1] 0.244898
```

Our model predicted with a 77.1% accuracy. The probability that our model incorrectly predicts below median and is above median is 21.4%. The probability that our model incorrectly predicts above the median and is below the median is 24.5%.