Week 6 Assignment

David Russo

2/18/2017

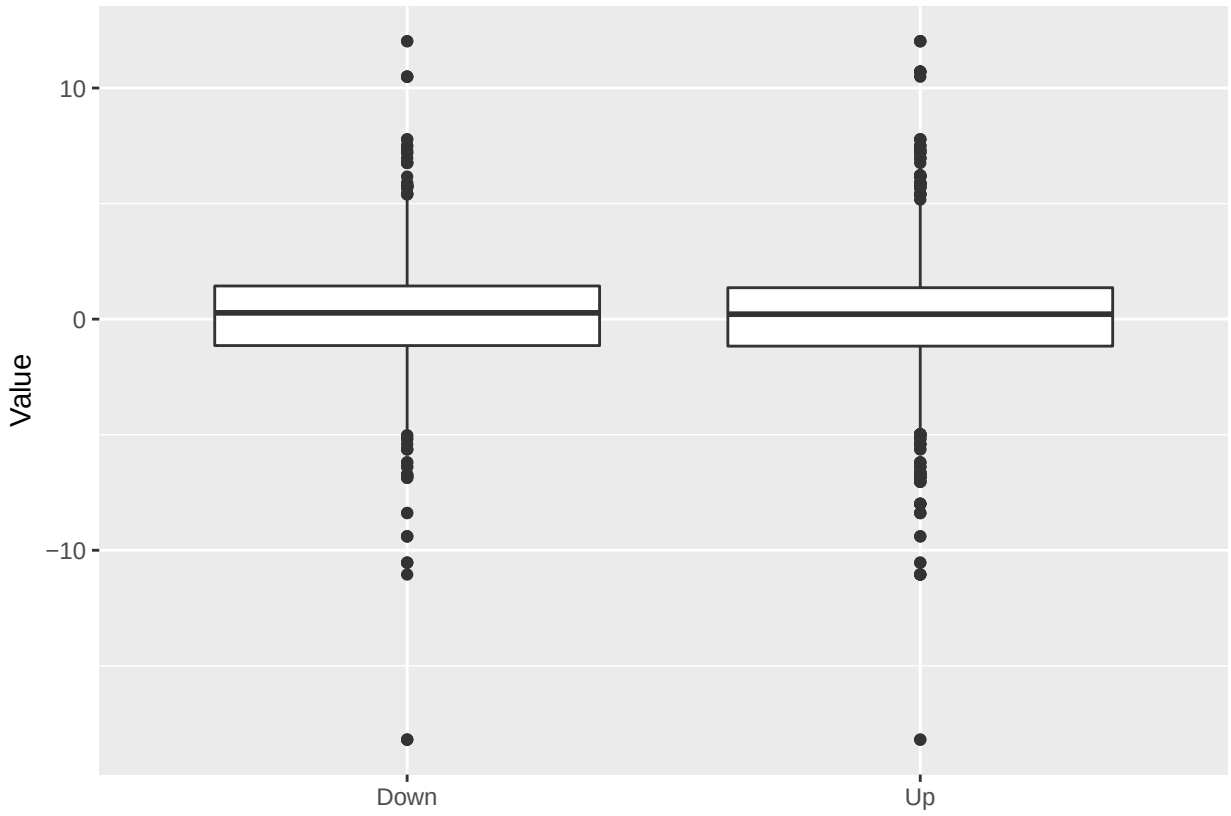
- 4.7 #10
- a.

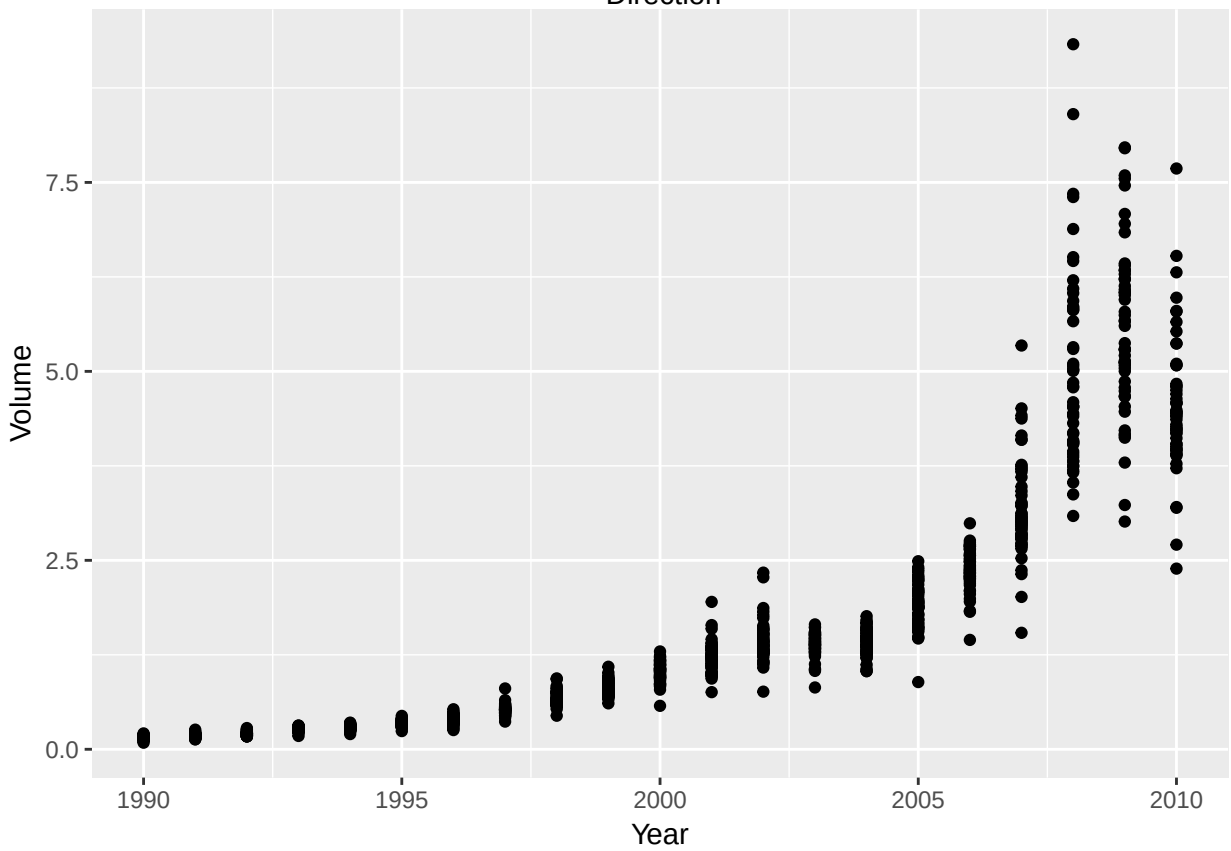
Direction	Count	Percent
Down	484	44%
Up	605	56%

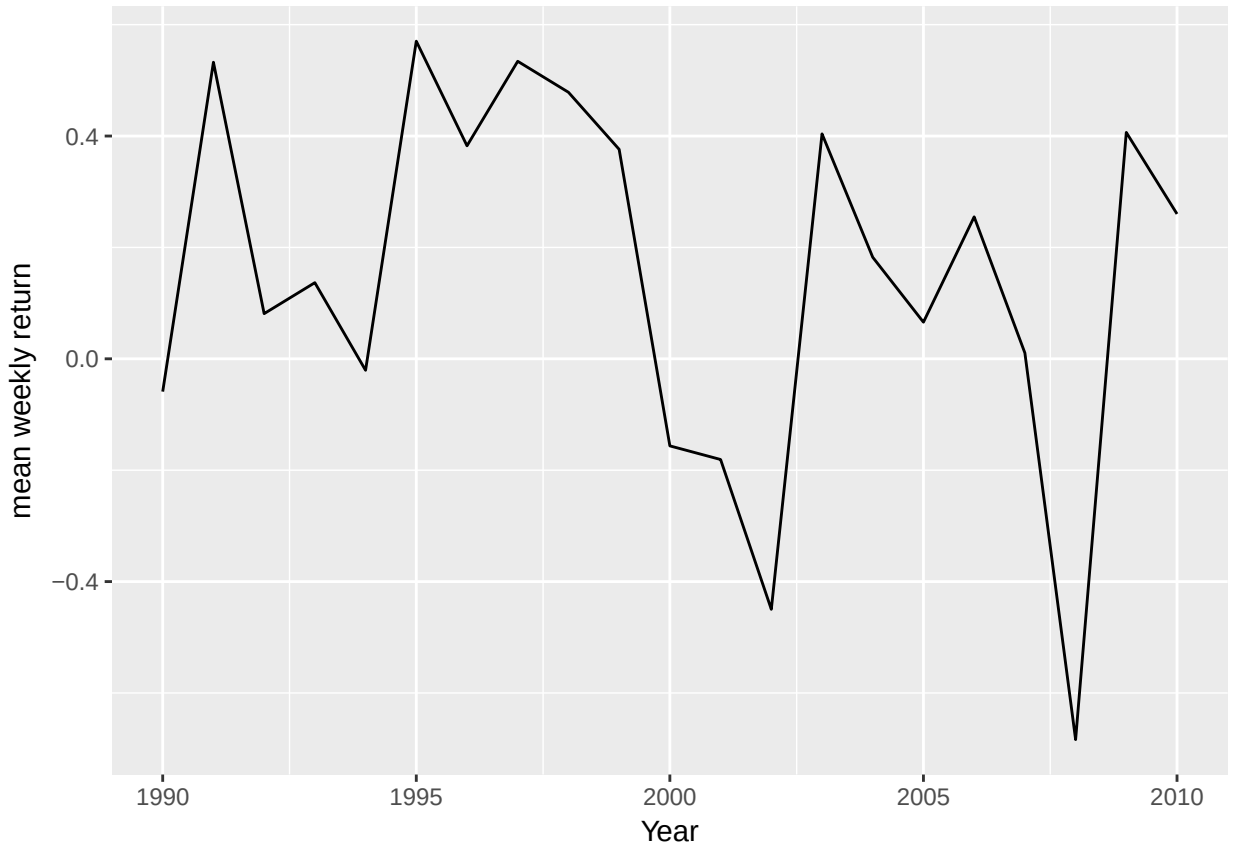
Direction	Mean Lag1	Mean Lag2	Mean Lag3	Mean Lag4	Mean Lag5	Mean Volume	Mean Today
Down	0.2823	-0.0404	0.2076	0.2000	0.1878	1.6085	-1.7466
Up	0.0452	0.3043	0.0989	0.1025	0.1015	1.5475	1.6671

	Year	Lag1	Lag2	Lag3	Lag4	Lag5	Volume	Today
Year	1.0000000	-0.0322893	-0.0333900	-0.0300065	-0.0311279	-0.0305191	0.8419416	-0.0324599
Lag1	-0.0322893	1.0000000	-0.0748531	0.0586357	-0.0712739	-0.0081831	-0.0649513	-0.0750318
Lag2	-0.0333900	-0.0748531	1.0000000	-0.0757209	0.0583815	-0.0724995	-0.0855131	0.0591667
Lag3	-0.0300065	0.0586357	-0.0757209	1.0000000	-0.0753959	0.0606572	-0.0692877	-0.0712436
Lag4	-0.0311279	-0.0712739	0.0583815	-0.0753959	1.0000000	-0.0756750	-0.0610746	-0.0078259
Lag5	-0.0305191	-0.0081831	-0.0724995	0.0606572	-0.0756750	1.0000000	-0.0585174	0.0110127
Volume	0.8419416	-0.0649513	-0.0855131	-0.0692877	-0.0610746	-0.0585174	1.0000000	-0.0330778
Today	-0.0324599	-0.0750318	0.0591667	-0.0712436	-0.0078259	0.0110127	-0.0330778	1.0000000

We see that for the market data, about 56% of the weeks had positive market performance while 44% of the weeks had negative market performance. The value of *today* does not appear to be highly correlated with any of the *lag* or *volume* covariates.


10
0
−10
Value
Down
Up
Direction


7.5
5.0
2.5
0.0
Volume
1990
1995
2000
2005
2010
Year



From the side-by-side boxplots, we see that markets finished up and down with relatively equal magnitudes. Furthermore, we can see that the number of trades has increased exponentially since 1990. Lastly, we can see that the average weekly return has varied from year to year, with several more down years between the years 2000 and 2010 vs 1990 and 2000.

- b.

```
##
## 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
```

Only the variable Lag2 appears to be a significant predictor of direction.

- c.

As can be seen from the confusion matrix below, the model tends to predict that the market will go up, as 91% of the predicted values were for the market going up. When the model predicts that the market will go up, it is correct 56% of the time. When the model predicts that the market will go down, it is right 53% of the time. Overall, the model has a 56% accuracy rate.The model is overly optimisitc, as the stock market only went up around 56% of the time.

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

```
##
## glm.b.preds Down  Up
##        Down   54  48
##        Up    430 557
```

- d.

As can be seen from the confusion matrix below, the overall accuracy on the test data is 62%.

```
##
## glm.d.preds Down Up
##        Down    9  5
##        Up     34 56
```

- e.

As can be seen from the confusion matrix below, the predictions for linear discriminant analysis mirror those of logistic regression. The overall accuracy rate is again 62%.

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

- f.

As can be seen from the confusion matrix below, the predictions for quadratic discriminant analysis are always "Up". The overall accuracy is 59%. It is possible that a cut-off other than 0.50 will yield improved accuracy.

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

- g.

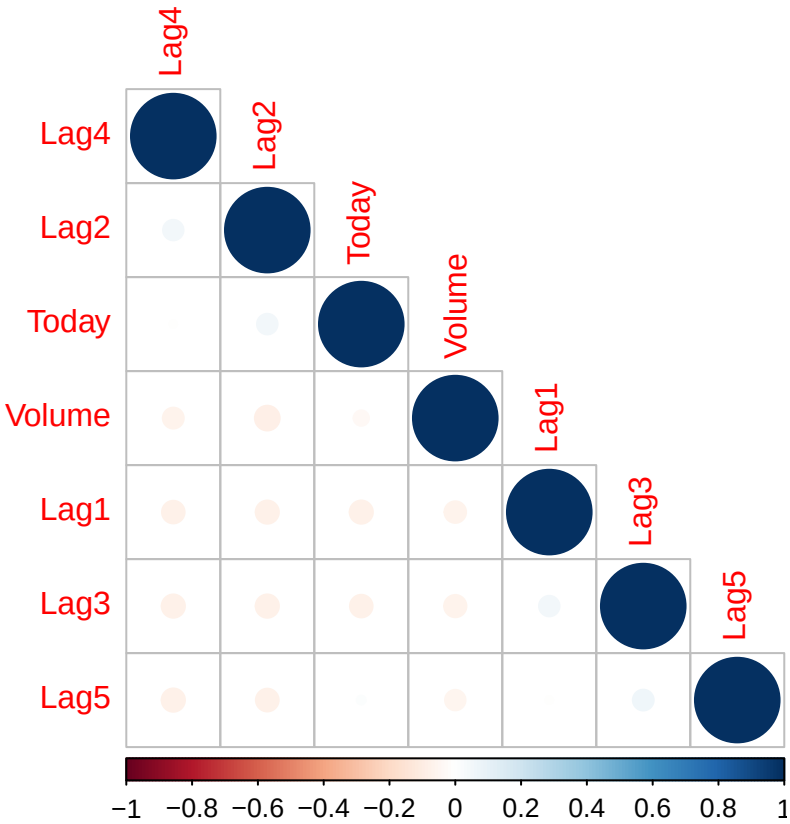
As can be seen from the confusion matrix below, KNN with K = 1 has an overall accuracy of 50%.

```
##
```

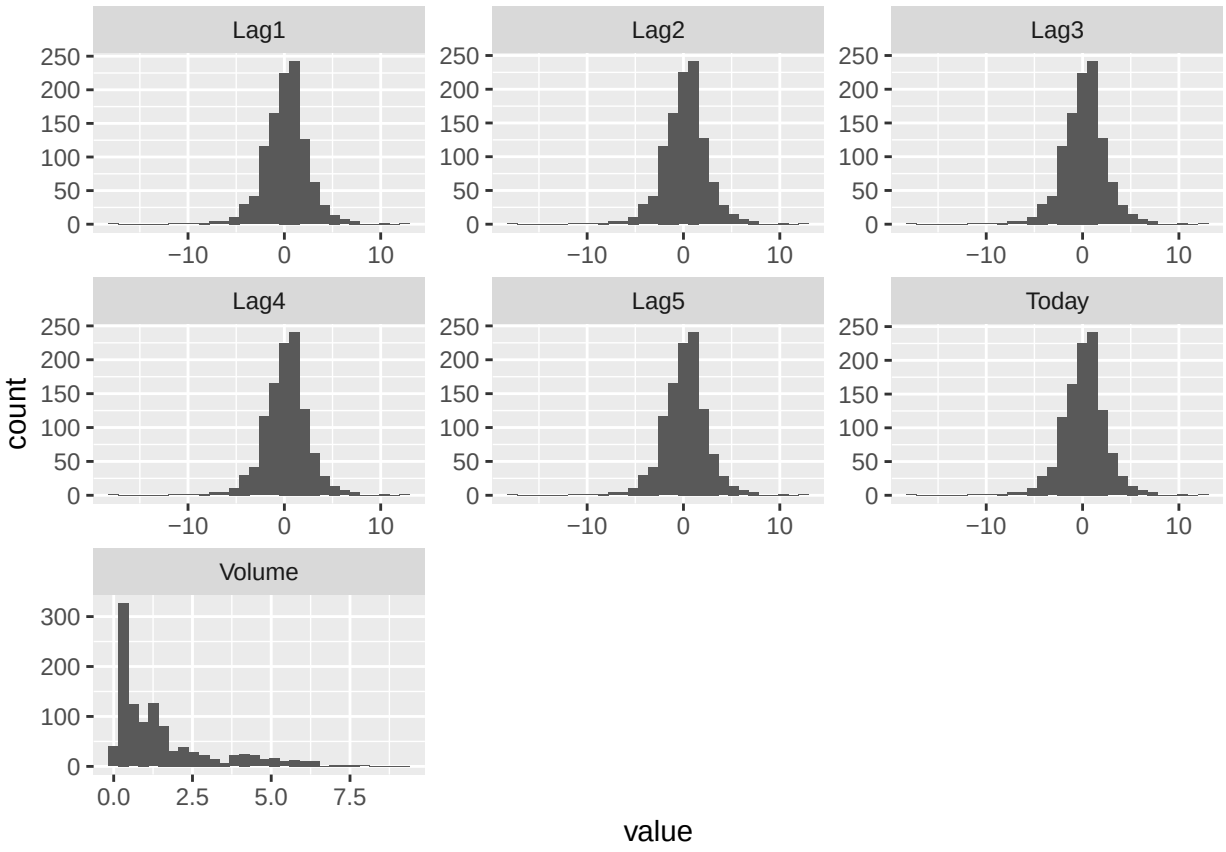
```
## knn.pred Down Up
##      Down   21 30
##      Up     22 31
```

- h. Of these methods, both logistic regression and LDA perform the best with respect to their test set error rates.
- i. As can be seen from the correlation matrix plot below, there aren't many candidates for interactions. Volume and Lag2 have a correlation of -0.086 while Lag2 and Lag3 have a correlation of -0.076. Judging by the histograms, the only variable that could be improved by a transformation is Volume.

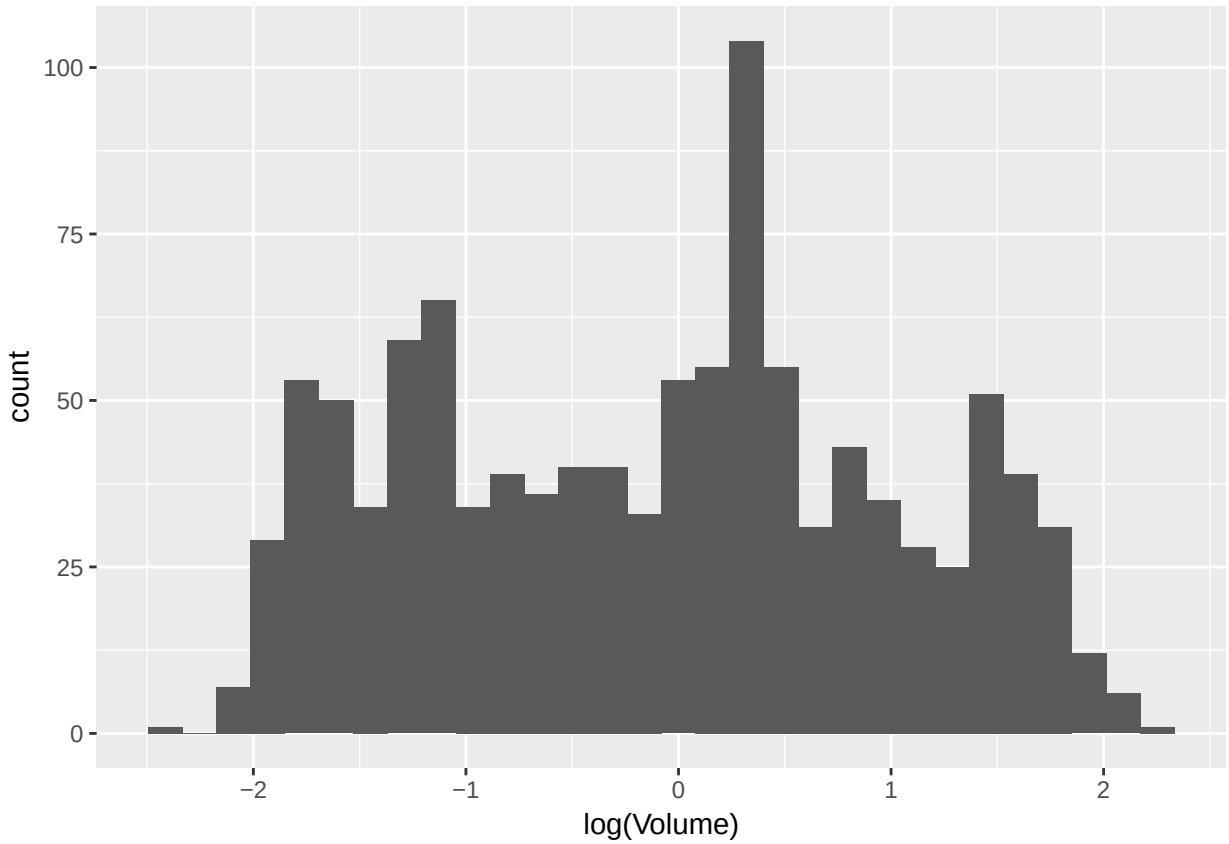
```
##                Lag1        Lag2        Lag3         Lag4         Lag5
## Lag1    1.000000000 -0.07485305  0.05863568 -0.071273876 -0.008183096
## Lag2   -0.074853051  1.00000000 -0.07572091  0.058381535 -0.072499482
## Lag3    0.058635682 -0.07572091  1.00000000 -0.075395865  0.060657175
## Lag4   -0.071273876  0.05838153 -0.07539587  1.000000000 -0.075675027
## Lag5   -0.008183096 -0.07249948  0.06065717 -0.075675027  1.000000000
## Volume -0.064951313 -0.08551314 -0.06928771 -0.061074617 -0.058517414
## Today  -0.075031842  0.05916672 -0.07124364 -0.007825873  0.011012698
##              Volume        Today
## Lag1   -0.06495131 -0.075031842
## Lag2   -0.08551314  0.059166717
## Lag3   -0.06928771 -0.071243639
## Lag4   -0.06107462 -0.007825873
## Lag5   -0.05851741  0.011012698
## Volume  1.00000000 -0.033077783
## Today  -0.03307778  1.000000000
```




```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



To determine which variables to investigate, we will fit a logistic regression on all variables with an interaction between Volume and Lag2, an interaction between Lag2 and Lag3, and a log-transformed Volume variable. As can be seen from the results below, Lag1 is significant at the $\alpha = 0.05$ level. Lag2 and log(Volume) are not significant at the $\alpha - 0.05$ level, but are borderline significant. Not surprisingly, the interactions are not significant as there was not much evidence of correlation amongst the variables.

```
##
## Call:
## glm(formula = Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5 +
##     log(Volume) + Lag2 * log(Volume) + Lag2 * Lag3, family = binomial,
##     data = Weekly_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8465  -1.2518   0.9877   1.0823   1.5563
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)       0.202546   0.068294   2.966  0.00302 **
## Lag1             -0.060516   0.029164  -2.075  0.03799 *
## Lag2              0.049432   0.031203   1.584  0.11315
## Lag3             -0.013993   0.029330  -0.477  0.63330
## Lag4             -0.030818   0.029361  -1.050  0.29389
## Lag5             -0.036623   0.029468  -1.243  0.21394
## log(Volume)      -0.099891   0.066059  -1.512  0.13050
## Lag2:log(Volume)  0.023994   0.029614   0.810  0.41781
## Lag2:Lag3         0.004577   0.008281   0.553  0.58045
```

```
## ---
## 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: 1342.0  on 976  degrees of freedom
## AIC: 1360
##
## Number of Fisher Scoring iterations: 4
```

Each of logistic regression, LDA, and QDA will be tested with the following models:

1) $Direction\ Lag1$
2) $Direction\ Lag2$
3) $Direction\ Lag1 + Lag2$
4) $Direction\ Lag1 + Lag2 + log(Volume)$
5) $Direction\ Lag1 + Lag2 + log(Volume) + Lag2 * Log(Volume)$
6) $Direction\ Lag1 + Lag2 + log(Volume) + Lag2 * Lag3$
7) $Direction\ Lag1 + Lag2 + log(Volume) + Lag2 * Log(Volume) + Lag2 * Lag3$

The training set containing years 1990 through 2008 will be used to train each model, and the test set containing the years 2009 and 2010 will be used to evaluate the models. In total, 21 models will be trained and tested: 7 variable specifications and 3 model types. Similarly, 11 models were fit for KNN with K ranging from 1 to 100. The final results of the 32 models are sorted by accuracy below. The best performing models are logistic regression and LDA, using Lag2, and a combination of Lag1, Lag2, log(Volume), and the Lag2*Lag3 interaction. The top 4 models all had an accuracy of 62.50%. Their confusion matrices can be found below.

method	model	accuracy
logistic regression	Direction ~ Lag2	0.6250
logistic regression	Direction ~ Lag1 + Lag2 + log(Volume) + Lag2*Lag3	0.6250
lda	Direction ~ Lag2	0.6250
lda	Direction ~ Lag1 + Lag2 + log(Volume) + Lag2*Lag3	0.6250
logistic regression	Direction ~ Lag1 + Lag2 + log(Volume)	0.6058
lda	Direction ~ Lag1 + Lag2 + log(Volume)	0.6058
KNN	k = 80	0.6058
logistic regression	Direction ~ Lag1 + Lag2 + log(Volume) + Lag2*log(Volume)	0.5865
logistic regression	Direction ~ Lag1 + Lag2 + log(Volume) + Lag2*log(Volume)* + *Lag2*Lag3	0.5865
lda	Direction ~ Lag1 + Lag2 + log(Volume) + Lag2*log(Volume)	0.5865
lda	Direction ~ Lag1 + Lag2 + log(Volume) + Lag2*log(Volume)* + *Lag2*Lag3	0.5865
qda	Direction ~ Lag1	0.5865
qda	Direction ~ Lag2	0.5865
KNN	k = 50	0.5865
KNN	k = 90	0.5865
logistic regression	Direction ~ Lag1 + Lag2	0.5769
lda	Direction ~ Lag1 + Lag2	0.5769
KNN	k = 10	0.5769
KNN	k = 60	0.5769
logistic regression	Direction ~ Lag1	0.5673
lda	Direction ~ Lag1	0.5673
qda	Direction ~ Lag1 + Lag2 + log(Volume) + Lag2*Lag3	0.5673

method	model	accuracy
qda	Direction ~ Lag1 + Lag2 + log(Volume) + Lag2*log(Volume)* + *Lag2*Lag3	0.5673
KNN	k = 20	0.5673
KNN	k = 70	0.5673
qda	Direction ~ Lag1 + Lag2	0.5577
qda	Direction ~ Lag1 + Lag2 + log(Volume) + Lag2*log(Volume)	0.5577
KNN	k = 40	0.5577
KNN	k = 100	0.5577
KNN	k = 30	0.5385
qda	Direction ~ Lag1 + Lag2 + log(Volume)	0.5096
KNN	k = 1	0.5000

```
# Logistic Regression Direction ~ Lag2
table(lr.1.preds, Weekly_test$Direction)
```

```
##
## lr.1.preds Down Up
##       Down    9  5
##       Up     34 56
```

```
# Logistic Regression Direction ~ Lag1 + Lag2 + log(Volume) + Lag2*Lag3
table(lr.2.preds, Weekly_test$Direction)
```

```
##
## lr.2.preds Down Up
##       Down   22 18
##       Up     21 43
```

```
# LDA Direction ~ Lag2
table(lda.1.preds, Weekly_test$Direction)
```

```
##
## lda.1.preds Down Up
##        Down    9  5
##        Up     34 56
```

```
# LDA Direction ~ Lag1 + Lag2 + log(Volume) + Lag2*Lag3
table(lda.2.preds, Weekly_test$Direction)
```

```
##
## lda.2.preds Down Up
##        Down   22 18
##        Up     21 43
```

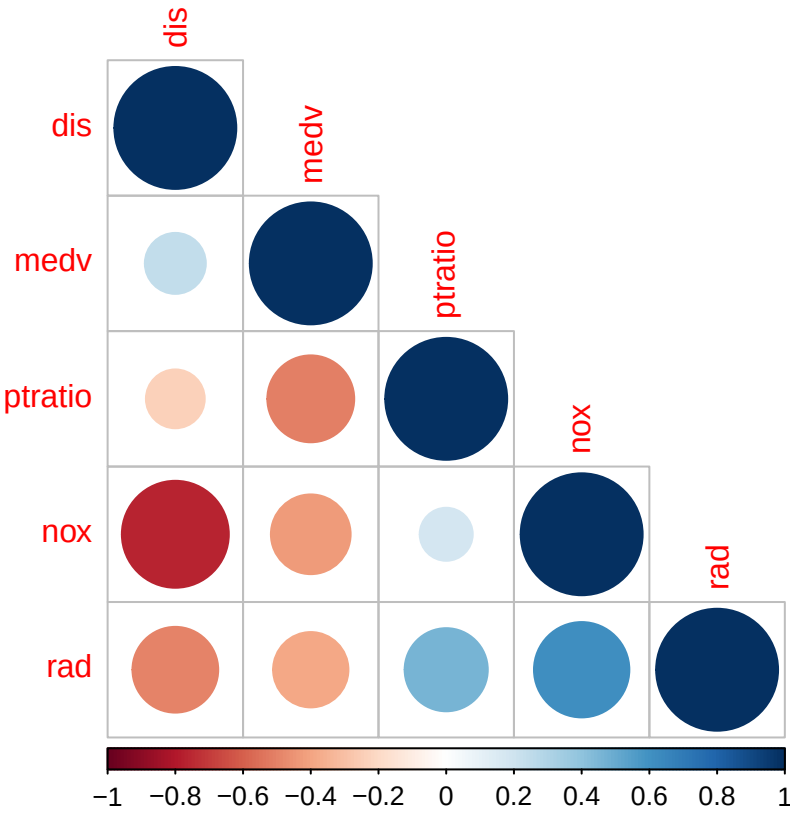
- 4.7 #13 The median of the crime variable is 0.25651. We begin by creating a variable that equals "above" if crime is above the median and "below" if crime is below the median. We also create training and testing data sets.

```
# load Boston data
data(Boston)

# Create Above Median Crime rate variable called amc
Boston$amc <- as.factor(ifelse(Boston$crim >= median(Boston$crim), "above", "below"))

# Remove the crime variable to avoid non convergence of glm
Boston$crim <- NULL
```

```
# Create training and testing data by reserving 75% of the Boston data for training
# and the remaining 25% for testing
set.seed(1)
train_index <- sample(c(TRUE, FALSE), nrow(Boston), prob = c(0.75, 0.25), replace = TRUE)
Boston_train <- Boston[train_index, ]
Boston_test <- Boston[!train_index, ]
```

To perform the initial variable selection, we will perform a logistic regression on the training data to identify candidate variables.The following variables are significant at the $\alpha = 0.02$ level: *nox*, *dis*, *rad*, *ptratio*, and *medv*.

We now examine potential interactions between *nox*, *dis*, *rad*, *ptratio*, and *medv*, as well as possible transformations. Judging by the correlation plot, nox and dis as well as rad and nox have the highest correlations.



For the logistic and LDA models, we will fit the following models:

1) $amc\ nox + dis + rad + ptratio + medv$
2) $amc\ nox + dis + rad + ptratio + medv + nox * dis$
3) $amc\ nox + dis + rad + ptratio + medv + nox * rad$
4) $amc\ nox + dis + rad + ptratio + medv + nox * dis + nox * rad$

KNN was fit with *nox*, *dis*, *rad*, *ptratio*, and *medv*.

As can be seen from the table below, KNN with k = 5 had the highest accuracy on the test data.

method	model	accuracy
KNN	k = 5	0.9016

method	model	accuracy
KNN	k = 1	0.8934
logistic regression	amc ~ nox + dis + rad + ptratio + medv + nox*dis	0.8852
logistic regression	amc ~ nox + dis + rad + ptratio + medv + nox*dis* + *nox*rad	0.8852
KNN	k = 3	0.8852
KNN	k = 7	0.8770
logistic regression	amc ~ nox + dis + rad + ptratio + medv + nox*rad	0.8689
lda	amc ~ nox + dis + rad + ptratio + medv + nox*dis* + *nox*rad	0.8672
logistic regression	amc ~ nox + dis + rad + ptratio + medv	0.8607
lda	amc ~ nox + dis + rad + ptratio + medv + nox*rad	0.8568
lda	amc ~ nox + dis + rad + ptratio + medv + nox*dis	0.8542
KNN	k = 9	0.8443
KNN	k = 11	0.8443
KNN	k = 29	0.8443
KNN	k = 51	0.8443
KNN	k = 53	0.8443
KNN	k = 55	0.8443
KNN	k = 57	0.8443
KNN	k = 59	0.8443
KNN	k = 61	0.8443