HW5

Question 8.1

Describe a situation or problem from your job, everyday life, current events, etc., for which a linear regression model would be appropriate. List some (up to 5) predictors that you might use.

I am interested in renting Airbnb apartments in several metropolitan areas, these are the predictors I will use to estimate

- Distance of the listing the metropole (in Miles)
- Distance of the listing from the international airport (in Miles)
- Distance of the listing to shopping centers, gyms, downtown, restaurants
- Number of added amenities in the house (such as X-box/PlayStation, pool table, Backyard grill, complimentary breakfast, pool…)
- Average guest ratings from previous listings/total number of stays

I'd like to understand which attributes **Correlate** more to the ratings and be able to predict a reservation's rating based on my model.

Question 8.2

Using crime data from http://www.statsci.org/data/general/uscrime.txt (file uscrime.txt, description at http://www.statsci.org/data/general/uscrime.html), use regression (a useful R function is lm or glm) to predict the observed crime rate in a city with the following data:

M = 14.0
So = 0
Ed = 10.0
Po1 = 12.0
Po2 = 15.5
LF = 0.640
M.F = 94.0
Pop = 150
NW = 1.1
U1 = 0.120
U2 = 3.6
Wealth = 3200
Ineq = 20.1
Prob = 0.04
Time = 39.0

Show your model (factors used and their coefficients), the software output, and the quality of fit.

Note that because there are only 47 data points and 15 predictors, you'll probably notice some overfitting. We'll see ways of dealing with this sort of problem later in the course.

For this question, I first extracted the txt data to a table, then used a for loop to determine which K would be optimal for our cross validation linear regression. Then, I used the data points provided in this question to build a data frame and predict the crime rate based on that. My first prediction was 155, less than a half of our lowest crime rate (347). I thought about what could have caused it, and I concluded that it was probably because there was some overfitting due to the inclusion of insignificant factors. I chose to keep only factors with p-value<=0.1: M,Ed,Po1,U2,Ineq,Prob.
With this new updated data, the prediction was 1304, I plotted a qqnorm plot to confirm whether it was an outlier or not (seems like it was not the case).
I also plotted standardized residuals vs Crime to visualize if any of our standardized residuals exceeds 3 (this is an indication that there might be outliers in the data), none of the plots met this condition.
One observation I made: the Rsquared for updated crime data was less than Rsquared value for our original crime data, which makes sense, because we removed the insignificant factors (our initial ratio to data points was about 3:1). Another important thing was the big difference between the Rsquared for our cross validation linear regression model and our original data (0.803 vs 0.555), which is a good demonstration of the importance to do cross validation on our data.
I followed these exact same steps for using glm(), a more generalized function for regression, using the gaussian family. The Rsquared for the gaussian linear model cross validated with our updated data was almost equal to the Rquared of our Cross validated linear model with updated data. There was slight discrepancy between the Rsquareds of the original data (might be due to the insignificant factors?)
Note: I had to remove a lot of line codes to condense my report because it was quite lengthy.

```
install.packages("DAAG",repos = "http://cran.us.r-project.org")

##
## The downloaded binary packages are in
##  /var/folders/j3/_y2j_7ts0dnfx0t2rj704r940000gn/T//Rtmp3AwJTN/downloaded_p
ackages

library(DAAG)

## Loading required package: lattice

rm(list = ls())
set.seed(456)
crime_data <- read.table("uscrime.txt", stringsAsFactors = FALSE, header = TR
```

```
UE)
head(crime_data)

##      M So   Ed  Po1  Po2    LF   M.F Pop   NW    U1  U2 Wealth Ineq     Pr
ob
## 1 15.1  1  9.1  5.8  5.6 0.510  95.0  33 30.1 0.108 4.1   3940 26.1 0.0846
02
## 2 14.3  0 11.3 10.3  9.5 0.583 101.2  13 10.2 0.096 3.6   5570 19.4 0.0295
99
## 3 14.2  1  8.9  4.5  4.4 0.533  96.9  18 21.9 0.094 3.3   3180 25.0 0.0834
01
## 4 13.6  0 12.1 14.9 14.1 0.577  99.4 157  8.0 0.102 3.9   6730 16.7 0.0158
01
## 5 14.1  0 12.1 10.9 10.1 0.591  98.5  18  3.0 0.091 2.0   5780 17.4 0.0413
99
## 6 12.1  0 11.0 11.8 11.5 0.547  96.4  25  4.4 0.084 2.9   6890 12.6 0.0342
01
##      Time Crime
## 1 26.2011   791
## 2 25.2999  1635
## 3 24.3006   578
## 4 29.9012  1969
## 5 21.2998  1234
## 6 20.9995   682

#the lm function is used to fit a simple linear regression model using our cr
ime data
crime_model<-lm(Crime~.,data = crime_data)
summary(crime_model)

##
## Call:
## lm(formula = Crime ~ ., data = crime_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -395.74  -98.09   -6.69  112.99  512.67
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.984e+03  1.628e+03  -3.675 0.000893 ***
## M            8.783e+01  4.171e+01   2.106 0.043443 *
## So          -3.803e+00  1.488e+02  -0.026 0.979765
## Ed           1.883e+02  6.209e+01   3.033 0.004861 **
## Po1          1.928e+02  1.061e+02   1.817 0.078892 .
## Po2         -1.094e+02  1.175e+02  -0.931 0.358830
## LF          -6.638e+02  1.470e+03  -0.452 0.654654
## M.F          1.741e+01  2.035e+01   0.855 0.398995
## Pop         -7.330e-01  1.290e+00  -0.568 0.573845
## NW           4.204e+00  6.481e+00   0.649 0.521279
```

```
## U1          -5.827e+03  4.210e+03  -1.384 0.176238
## U2           1.678e+02  8.234e+01   2.038 0.050161 .
## Wealth       9.617e-02  1.037e-01   0.928 0.360754
## Ineq         7.067e+01  2.272e+01   3.111 0.003983 **
## Prob        -4.855e+03  2.272e+03  -2.137 0.040627 *
## Time        -3.479e+00  7.165e+00  -0.486 0.630708
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 209.1 on 31 degrees of freedom
## Multiple R-squared:  0.8031, Adjusted R-squared:  0.7078
## F-statistic: 8.429 on 15 and 31 DF,  p-value: 3.539e-07
```

```
ss=0
#Loop to determine the optimal K fold,computing the sum of squared errors ove
r a K fold of 1 to 20 (I start my loop wioth 2 to avoid generating NA values)
for (k in 2:20){
  cross_validation_model<-cv.lm(crime_data,crime_model,m=k)
  sum_squared_errors=sum((cross_validation_model$Crime-cross_validation_model
$cvpred)^2)
  ss[k]=sum_squared_errors
}
```

```
## Analysis of Variance Table
##
## Response: Crime
##           Df  Sum Sq Mean Sq F value  Pr(>F)
## M          1   55084   55084    1.26  0.2702
## So         1   15370   15370    0.35  0.5575
## Ed         1  905668  905668   20.72 7.7e-05 ***
## Po1        1 3076033 3076033   70.38 1.8e-09 ***
## Po2        1  153024  153024    3.50  0.0708 .
## LF         1   61134   61134    1.40  0.2459
## M.F        1  111000  111000    2.54  0.1212
## Pop        1   42649   42649    0.98  0.3309
## NW         1   14197   14197    0.32  0.5728
## U1         1    7065    7065    0.16  0.6904
## U2         1  269663  269663    6.17  0.0186 *
## Wealth     1   34748   34748    0.79  0.3795
## Ineq       1  547423  547423   12.52  0.0013 **
## Prob       1  222620  222620    5.09  0.0312 *
## Time       1   10304   10304    0.24  0.6307
## Residuals 31 1354946   43708
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## Warning in cv.lm(crime_data, crime_model, m = k):
##
##  As there is >1 explanatory variable, cross-validation
##  predicted values for a fold are not a linear function
```

```
##  of corresponding overall predicted values.  Lines that
##  are shown for the different folds are approximate
```

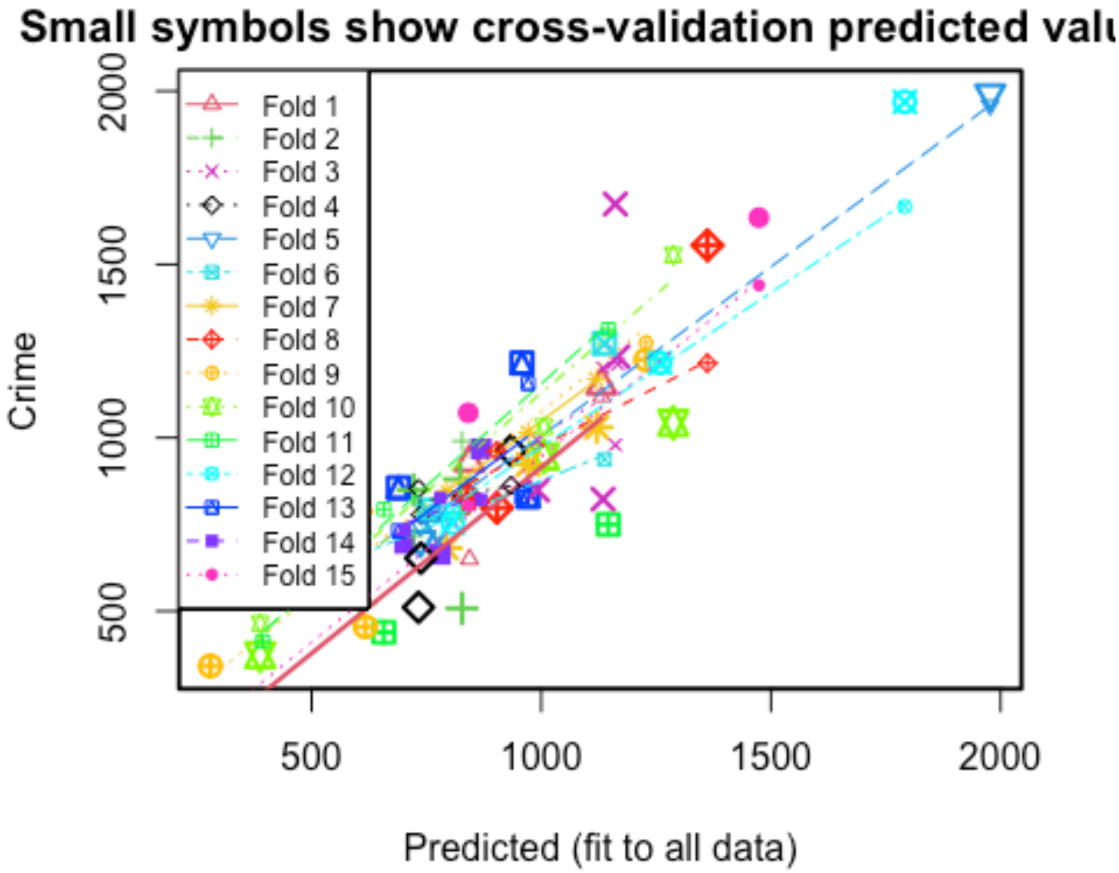
```r
ss=ss[2:20]
#adjusting since we looped from 2 to 20
Kfold_optimal=which.min(ss)+1
Kfold_optimal
```

```
## [1] 15
```

```r
#cross validation function with optimal K fold
cross_validation_model<-cv.lm(crime_data,crime_model,m=Kfold_optimal)
```

```
## Analysis of Variance Table
##
## Response: Crime
##           Df  Sum Sq Mean Sq F value  Pr(>F)
## M          1   55084   55084    1.26  0.2702
## So         1   15370   15370    0.35  0.5575
## Ed         1  905668  905668   20.72 7.7e-05 ***
## Po1        1 3076033 3076033   70.38 1.8e-09 ***
## Po2        1  153024  153024    3.50  0.0708 .
## LF         1   61134   61134    1.40  0.2459
## M.F        1  111000  111000    2.54  0.1212
## Pop        1   42649   42649    0.98  0.3309
## NW         1   14197   14197    0.32  0.5728
## U1         1    7065    7065    0.16  0.6904
## U2         1  269663  269663    6.17  0.0186 *
## Wealth     1   34748   34748    0.79  0.3795
## Ineq       1  547423  547423   12.52  0.0013 **
## Prob       1  222620  222620    5.09  0.0312 *
## Time       1   10304   10304    0.24  0.6307
## Residuals 31 1354946   43708
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## Warning in cv.lm(crime_data, crime_model, m = Kfold_optimal):
##
##  As there is >1 explanatory variable, cross-validation
##  predicted values for a fold are not a linear function
##  of corresponding overall predicted values.  Lines that
##  are shown for the different folds are approximate
```




```
##
## fold 1
## Observations in test set: 3
##                 3  18     40
## Predicted     322 844 1131.5
## cvpred        222 650 1119.1
## Crime         578 929 1151.0
## CV residual 356 279   31.9
##
## Sum of squares = 205534    Mean square = 68511    n = 3
##
## fold 2
## Observations in test set: 4
##                12   25    41   46
## Predicted     722  606 823.7  827
## cvpred        715  676 783.8  990
## Crime         849  523 880.0  508
## CV residual 134 -153  96.2 -482
##
## Sum of squares = 283151    Mean square = 70788    n = 4
##
## fold 3
## Observations in test set: 4
```

```
##                  5   11   43   47
## Predicted   1166.7 1161 1134  992
## cvpred      1212.2  979 1201  992
## Crime       1234.0 1674  823  849
## CV residual   21.8  695 -378 -143
##
## Sum of squares = 646774    Mean square = 161693    n = 4
##
## fold 4
## Observations in test set: 3
##               7   13   35
## Predicted   934  733  738
## cvpred      862  853  778
## Crime       963  511  653
## CV residual 101 -342 -125
##
## Sum of squares = 142910    Mean square = 47637    n = 3
##
## fold 5
## Observations in test set: 3
##                 10    21   26
## Predicted   736.5 774.9 1977
## cvpred      745.5 788.2 1959
## Crime       705.0 742.0 1993
## CV residual -40.5 -46.2   34
##
## Sum of squares = 4925    Mean square = 1642    n = 3
##
## fold 6
## Observations in test set: 3
##                  1   36    38
## Predicted   755.03 1138 562.7
## cvpred      783.13  938 612.6
## Crime       791.00 1272 566.0
## CV residual   7.87  334 -46.6
##
## Sum of squares = 114016    Mean square = 38005    n = 3
##
## fold 7
## Observations in test set: 3
##                 6     34   44
## Predicted    793  971.5 1121
## cvpred       847 1015.5 1169
## Crime        682  923.0 1030
## CV residual -165  -92.5 -139
##
## Sum of squares = 54925    Mean square = 18308    n = 3
##
## fold 8
## Observations in test set: 3
```

```
##                 8   15     39
## Predicted   1362  903 839.29
## cvpred      1215  962 819.89
## Crime       1555  798 826.00
## CV residual  340 -164   6.11
##
## Sum of squares = 142374    Mean square = 47458    n = 3
##
## fold 9
## Observations in test set: 3
##                  20  27   45
## Predicted   1227.8 279  617
## cvpred      1274.7 231  788
## Crime       1225.0 342  455
## CV residual  -49.7 111 -333
##
## Sum of squares = 125579    Mean square = 41860    n = 3
##
## fold 10
## Observations in test set: 3
##                  16   29    31
## Predicted   1005.7 1287 388.0
## cvpred      1031.9 1527 464.6
## Crime        946.0 1043 373.0
## CV residual  -85.9 -484 -91.6
##
## Sum of squares = 250477    Mean square = 83492    n = 3
##
## fold 11
## Observations in test set: 3
##              17   19   22
## Predicted   393 1146  657
## cvpred      412 1313  794
## Crime       539  750  439
## CV residual 127 -563 -355
##
## Sum of squares = 458964    Mean square = 152988    n = 3
##
## fold 12
## Observations in test set: 3
##                 4     28    32
## Predicted   1791 1258.5 807.8
## cvpred      1667 1227.1 791.4
## Crime       1969 1216.0 754.0
## CV residual  302  -11.1 -37.4
##
## Sum of squares = 92649    Mean square = 30883    n = 3
##
## fold 13
## Observations in test set: 3
```

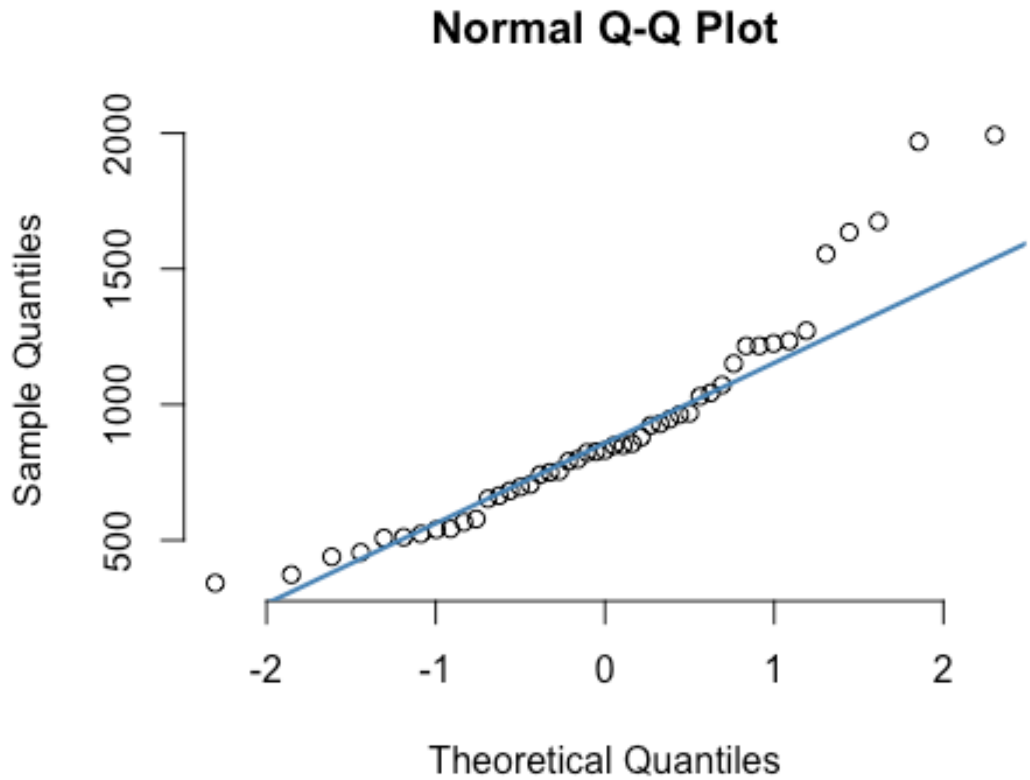
```
##                9   23   37
## Predicted    689  958  971
## cvpred       732  831 1157
## Crime        856 1216  831
## CV residual 124  385 -326
##
## Sum of squares = 269849    Mean square = 89950    n = 3
##
## fold 14
## Observations in test set: 3
##                14  24    30
## Predicted     780 869 702.7
## cvpred        827 822 732.9
## Crime         664 968 696.0
## CV residual -163 146 -36.9
##
## Sum of squares = 49234    Mean square = 16411    n = 3
##
## fold 15
## Observations in test set: 3
##                 2   33  42
## Predicted    1474  841 326
## cvpred       1440  805 209
## Crime        1635 1072 542
## CV residual  195  267 333
##
## Sum of squares = 220743    Mean square = 73581    n = 3
##
## Overall (Sum over all 3 folds)
##    ms
## 65151
```

```
#building my test data frame with the values given in Homework header
test_data_frame<-data.frame(M = 14.0,So = 0, Ed = 10.0, Po1 = 12.0, Po2 = 15.
5,LF = 0.640, M.F = 94.0, Pop = 150, NW = 1.1, U1 = 0.120, U2 = 3.6, Wealth =
3200, Ineq = 20.1, Prob = 0.040,Time = 39.0)
#Predict the crime rate for test data point
predicted_model <- predict(crime_model, test_data_frame,interval = 'confidenc
e')
predicted_model
```

```
##   fit   lwr  upr
## 1 155 -1310 1621
```

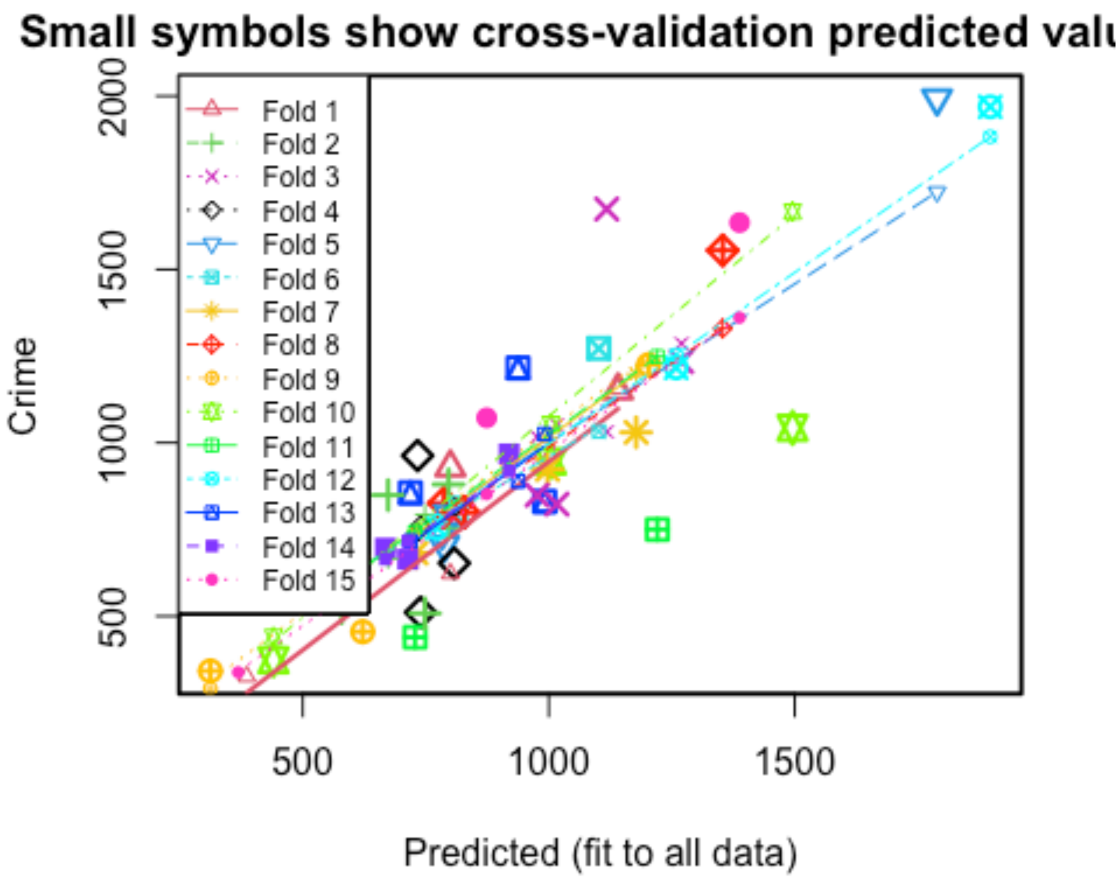
```
# The predicted crime value for our test data frame is less than half  than h
alf of the crime rate of the next-lowest city. None of the given values seem
out of range as well
#The issue might be that the full data frame includes a lot of factors that d
o not matter, so I adjusted and aonly chose factors with p-valuue<=0.1
```

```
crime_model_updated <- lm( Crime ~  M + Ed + Po1 + U2 + Ineq + Prob, data = c
rime_data)
summary(crime_model_updated)

##
## Call:
## lm(formula = Crime ~ M + Ed + Po1 + U2 + Ineq + Prob, data = crime_data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -470.7  -78.4  -19.7  133.1  556.2
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -5040.5      899.8   -5.60  1.7e-06 ***
## M              105.0       33.3    3.15   0.0031 **
## Ed             196.5       44.8    4.39  8.1e-05 ***
## Po1            115.0       13.8    8.36  2.6e-10 ***
## U2              89.4       40.9    2.18   0.0348 *
## Ineq            67.7       13.9    4.85  1.9e-05 ***
## Prob         -3801.8     1528.1   -2.49   0.0171 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 201 on 40 degrees of freedom
## Multiple R-squared:  0.766,  Adjusted R-squared:  0.731
## F-statistic: 21.8 on 6 and 40 DF,  p-value: 3.42e-11

#predict model based on our updated crime model
predicted_model_2 <-predict(crime_model_updated,test_data_frame,interval='con
fidence')
predicted_model_2



##    fit  lwr  upr
## 1 1304 1181 1428

#Our predicted value is 1304, plot a qq norm plot on the crime data to see if
the 1304 is an outlier.

qqnorm(crime_data$Crime,pch=1,frame=FALSE)
qqline(crime_data$Crime,col = "steelblue", lwd = 2)
```




```
#According to our plot, 1304 does not seem to be an outlier.

# Cross validation model updated without insignificant factors.

cross_validation_model_updated<-cv.lm(crime_data,crime_model_updated,m=Kfold_
optimal)

## Analysis of Variance Table
##
## Response: Crime
##            Df  Sum Sq Mean Sq F value  Pr(>F)
## M           1   55084   55084    1.37 0.24914
## Ed          1  725967  725967   18.02 0.00013 ***
## Po1         1 3173852 3173852   78.80 5.3e-11 ***
## U2          1  217386  217386    5.40 0.02534 *
## Ineq        1  848273  848273   21.06 4.3e-05 ***
## Prob        1  249308  249308    6.19 0.01711 *
## Residuals 40 1611057   40276
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Warning in cv.lm(crime_data, crime_model_updated, m = Kfold_optimal):
##
##  As there is >1 explanatory variable, cross-validation
```

```
##  predicted values for a fold are not a linear function
##  of corresponding overall predicted values.  Lines that
##  are shown for the different folds are approximate
```




```
##
## fold 1
## Observations in test set: 3
##                 3  18     40
## Predicted     386 800 1140.8
## cvpred        326 623 1155.9
## Crime         578 929 1151.0
## CV residual 252 306   -4.9
##
## Sum of squares = 157136    Mean square = 52379    n = 3
##
## fold 2
## Observations in test set: 4
##                12    25  41   46
## Predicted     673 579.1 796  748
## cvpred        666 602.7 754  793
## Crime         849 523.0 880  508
## CV residual 183 -79.7 126 -285
##
```

```
## Sum of squares = 137028    Mean square = 34257    n = 4
##
## fold 3
## Observations in test set: 4
##                  5   11   43   47
## Predicted   1269.8 1118 1017  976
## cvpred      1286.1 1032 1052 1017
## Crime       1234.0 1674  823  849
## CV residual  -52.1  642 -229 -168
##
## Sum of squares = 495961    Mean square = 123990    n = 4
##
## fold 4
## Observations in test set: 3
##               7   13   35
## Predicted   733  739  808
## cvpred      741  762  815
## Crime       963  511  653
## CV residual 222 -251 -162
##
## Sum of squares = 138461    Mean square = 46154    n = 3
##
## fold 5
## Observations in test set: 3
##                 10    21   26
## Predicted   787.3 783.3 1789
## cvpred      798.2 806.8 1723
## Crime       705.0 742.0 1993
## CV residual -93.2 -64.8  270
##
## Sum of squares = 85646    Mean square = 28549    n = 3
##
## fold 6
## Observations in test set: 3
##                 1   36      38
## Predicted   810.8 1102 544.373
## cvpred      826.8 1032 566.921
## Crime       791.0 1272 566.000
## CV residual -35.8  240  -0.921
##
## Sum of squares = 58652    Mean square = 19551    n = 3
##
## fold 7
## Observations in test set: 3
##                 6     34   44
## Predicted   730.3  997.5 1178
## cvpred      737.4 1013.2 1199
## Crime       682.0  923.0 1030
## CV residual -55.4  -90.2 -169
##
```

```
## Sum of squares = 39613    Mean square = 13204    n = 3
##
## fold 8
## Observations in test set: 3
##                8    15    39
## Predicted   1354 828.3 786.7
## cvpred      1330 821.5 778.3
## Crime       1555 798.0 826.0
## CV residual  225 -23.5  47.7
##
## Sum of squares = 53570    Mean square = 17857    n = 3
##
## fold 9
## Observations in test set: 3
##                 20    27   45
## Predicted   1203.0 312.2  622
## cvpred      1238.8 290.9  671
## Crime       1225.0 342.0  455
## CV residual  -13.8  51.1 -216
##
## Sum of squares = 49596    Mean square = 16532    n = 3
##
## fold 10
## Observations in test set: 3
##               16   29    31
## Predicted   1004 1495 440.4
## cvpred      1054 1667 439.9
## Crime        946 1043 373.0
## CV residual -108 -624 -66.9
##
## Sum of squares = 405029    Mean square = 135010    n = 3
##
## fold 11
## Observations in test set: 3
##                 17   19   22
## Predicted   527.37 1221  728
## cvpred      544.22 1249  743
## Crime       539.00  750  439
## CV residual  -5.22 -499 -304
##
## Sum of squares = 341825    Mean square = 113942    n = 3
##
## fold 12
## Observations in test set: 3
##                  4     28    32
## Predicted   1897.2 1259.0 773.7
## cvpred      1882.4 1254.1 774.3
## Crime       1969.0 1216.0 754.0
## CV residual   86.6  -38.1 -20.3
##
```

```
## Sum of squares = 9361    Mean square = 3120    n = 3
##
## fold 13
## Observations in test set: 3
##                9   23   37
## Predicted   719  938  992
## cvpred      717  889 1025
## Crime       856 1216  831
## CV residual 139  327 -194
##
## Sum of squares = 163755    Mean square = 54585    n = 3
##
## fold 14
## Observations in test set: 3
##                 14  24    30
## Predicted   713.6 919 668.0
## cvpred      716.1 919 664.4
## Crime       664.0 968 696.0
## CV residual -52.1  49  31.6
##
## Sum of squares = 6116    Mean square = 2039    n = 3
##
## fold 15
## Observations in test set: 3
##                 2   33  42
## Predicted   1388  874 369
## cvpred      1361  852 338
## Crime       1635 1072 542
## CV residual  274  220 204
##
## Sum of squares = 165322    Mean square = 55107    n = 3
##
## Overall (Sum over all 3 folds)
##    ms
## 49087
```

```
#library to compute r squared values of both linear regression models
install.packages("rsq",repos = "http://cran.us.r-project.org")
```

```
##
## The downloaded binary packages are in
##  /var/folders/j3/_y2j_7ts0dnfx0t2rj704r940000gn/T//RtmpMJhy4m/downloaded_p
ackages
```

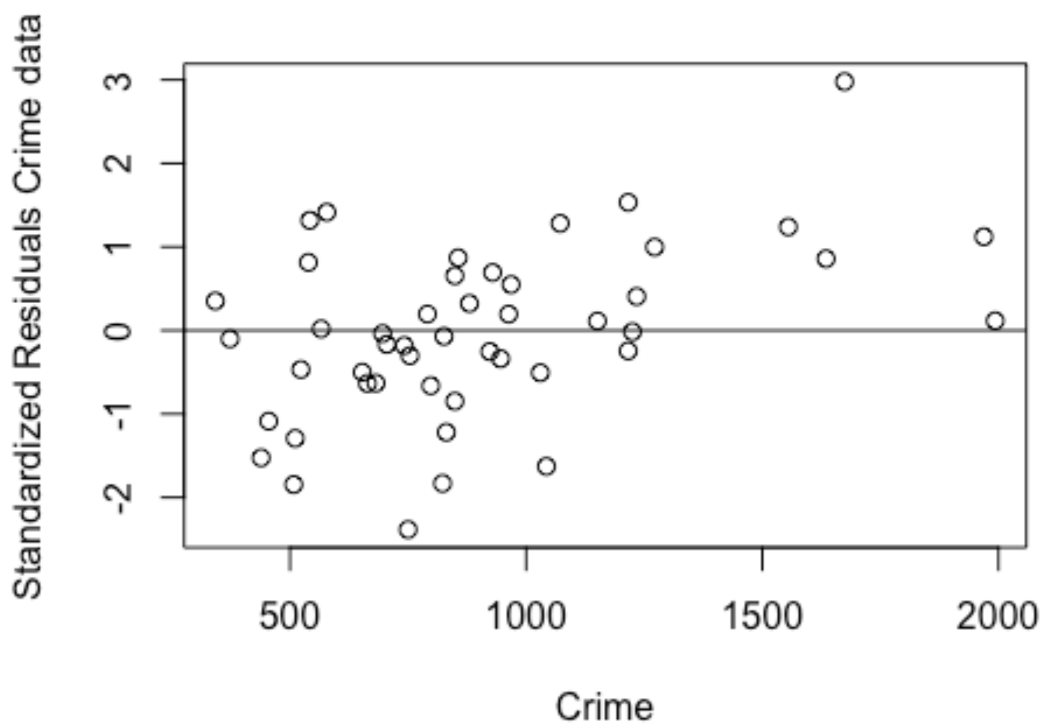
```
library(rsq)
rsq_model<-rsq(crime_model,adj=FALSE, type = 'sse')
rsq_model
```

```
## [1] 0.803
```

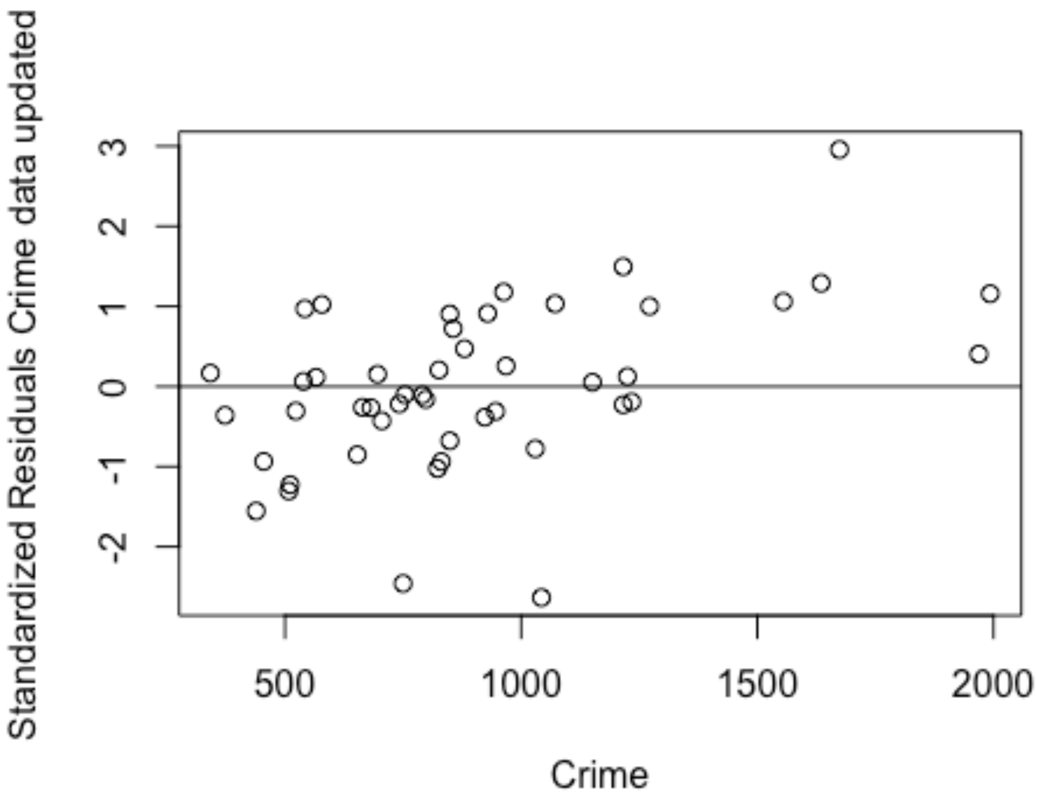
```
rsq_model_updated<-rsq(crime_model_updated,adj=FALSE,type='sse')
rsq_model_updated

## [1] 0.766

#Our rsq for the updated model is lower that our generic model, it shows that
including the insignificant factors overfits compared to when they are remove
d.

#Calculating standardized residuals
std_res<-rstandard(crime_model)
std_res

##       1       2       3       4       5       6       7       8       9
10
##  0.1953  0.8584  1.4148  1.1218  0.4060 -0.6314  0.1945  1.2387  0.8692 -0
.1672
##      11      12      13      14      15      16      17      18      19
20
##  2.9782  0.6546 -1.2922 -0.6346 -0.6626 -0.3387  0.8139  0.6933 -2.3846 -0
.0162
##      21      22      23      24      25      26      27      28      29
30
## -0.1813 -1.5254  1.5330  0.5491 -0.4679  0.1155  0.3517 -0.2455 -1.6275 -0
.0376
##      31      32      33      34      35      36      37      38      39
40
## -0.1014 -0.3031  1.2814 -0.2533 -0.5001  0.9973 -1.2196  0.0184 -0.0711  0
.1120
##      41      42      43      44      45      46      47
##  0.3225  1.3172 -1.8317 -0.5063 -1.0868 -1.8451 -0.8477

std_res_updated<-rstandard(crime_model_updated)
std_res_updated

##       1       2       3       4       5       6       7       8       9
10
## -0.1056  1.2893  1.0241  0.4038 -0.1923 -0.2653  1.1788  1.0599  0.7225 -0
.4290
##      11      12      13      14      15      16      17      18      19
20
##  2.9572  0.9044 -1.2256 -0.2601 -0.1607 -0.3109  0.0629  0.9124 -2.4594  0
.1239
##      21      22      23      24      25      26      27      28      29
30
## -0.2182 -1.5544  1.4959  0.2561 -0.3065  1.1586  0.1675 -0.2280 -2.6350  0
.1526
##      31      32      33      34      35      36      37      38      39
40
## -0.3594 -0.1021  1.0328 -0.3832 -0.8510  1.0010 -0.9371  0.1149  0.2057  0
.0528
```

```
##       41       42       43       44       45       46       47
##   0.4716   0.9714  -1.0236  -0.7770  -0.9351  -1.3048  -0.6771
```

```
#Column bind standard residuals back to original data frame
d1<-cbind(crime_data,std_res)
#Crime data updated without insignificant factors
crime_data_updated<-crime_data[,c(1,3,4,11,13,14,16)]
d2<-cbind(crime_data_updated,std_res_updated)
#Sort standard residuals descending
d1[order(-std_res),]

d2[order(-std_res_updated),]

#Plot predictor variable (crime) vs standardized residuals.

plot(d1$Crime,std_res,ylab='Standardized Residuals Crime data', xlab='Crime')
abline(0,0)
```




```
plot(d2$Crime,std_res_updated,ylab='Standardized Residuals Crime data updated
', xlab='Crime')
abline(0,0)
```

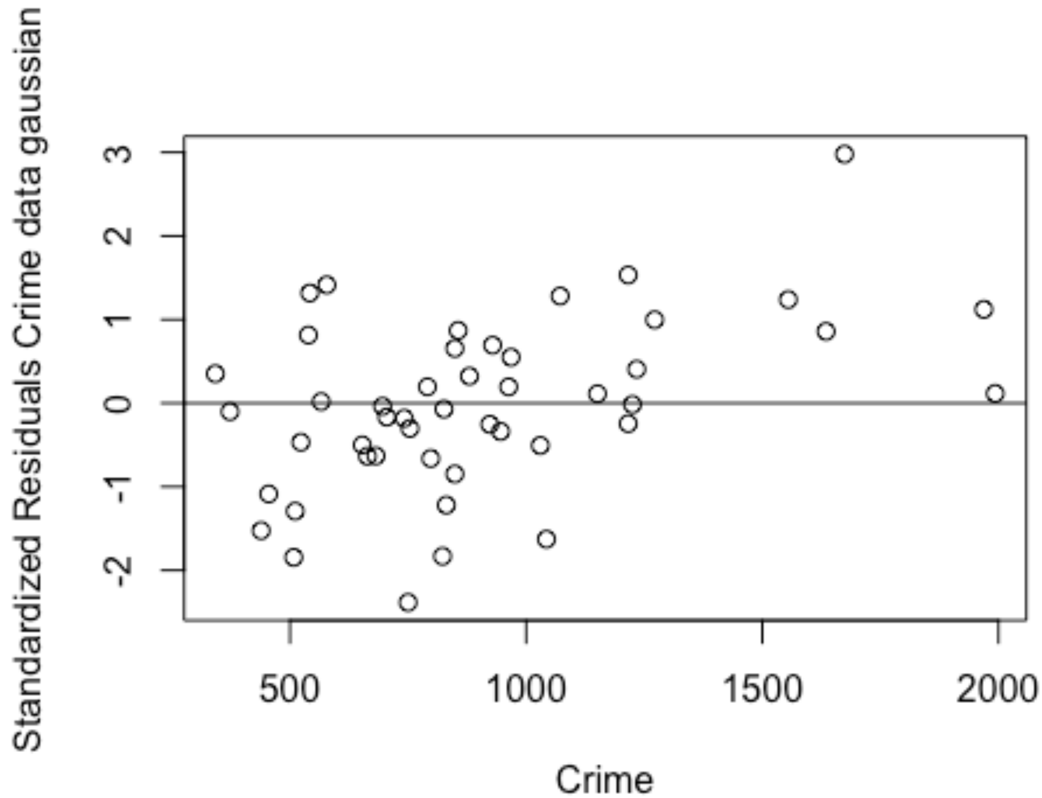



```
#We see that none of our standardized residuals absolute value does not excee
d 3, so it seems like none of the observations appear to be an outlier.

#total sum of squares between data and mean values
SStotal<- sum((crime_data$Crime - mean(crime_data$Crime))^2)
# For both our cross validation linear models sum of squared errors= the mean
squared error* number of data points
Sres<- attr(cross_validation_model,"ms")*nrow(crime_data)
Sres2<-attr(cross_validation_model_updated,"ms")*nrow(crime_data)
#  R-squared = 1 - SSEresiduals/SSEtotal
rsq_cv<-1-Sres/SStotal
rsq_cv

## [1] 0.555

rsq_cv2<-1-Sres2/SStotal
rsq_cv2

## [1] 0.665

#rsq for our updated cross validation model is larger than the rsq for our cr
original cross validation.
```

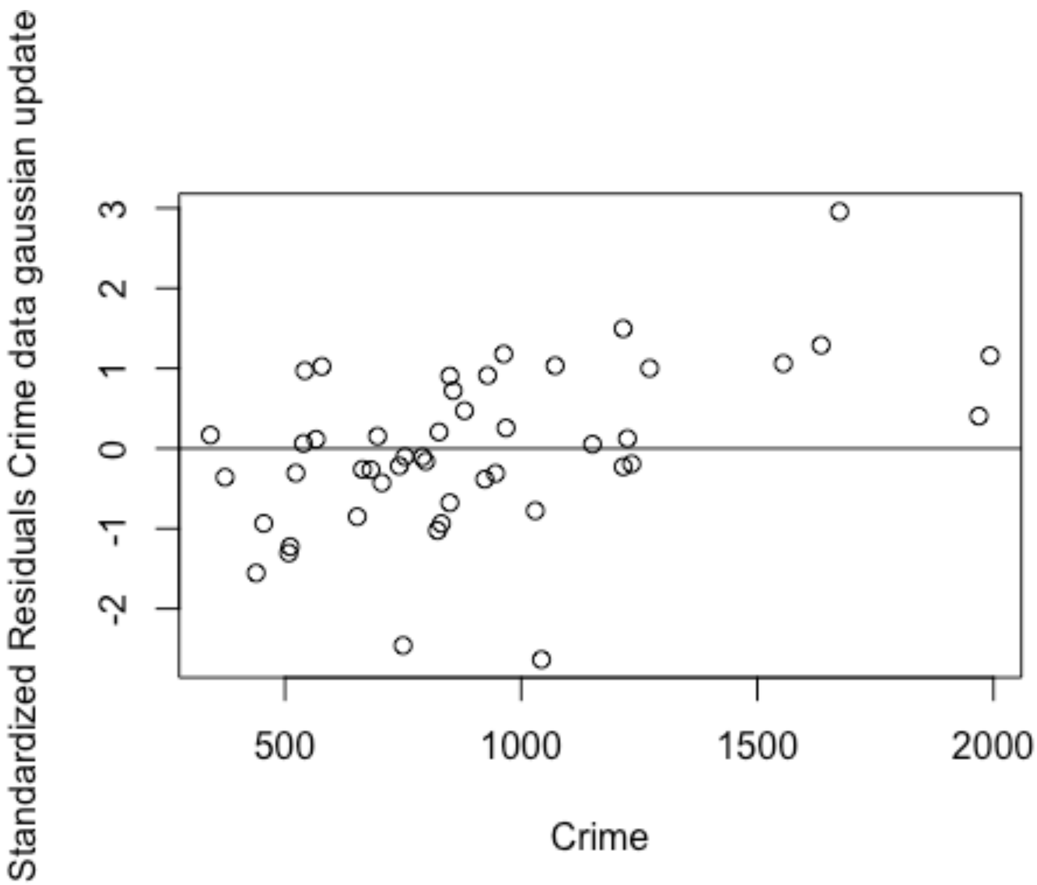
```r
#Using glm
#We need the boot library for gaussian cross validation
library(boot)

##
## Attaching package: 'boot'

## The following object is masked from 'package:lattice':
##
##     melanoma

gaussian_crime <- glm(Crime ~ ., data=crime_data, family="gaussian")
summary(gaussian_crime)

##
## Call:
## glm(formula = Crime ~ ., family = "gaussian", data = crime_data)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -395.7   -98.1    -6.7   113.0   512.7
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.98e+03   1.63e+03   -3.68  0.00089 ***
## M            8.78e+01   4.17e+01    2.11  0.04344 *
## So          -3.80e+00   1.49e+02   -0.03  0.97977
## Ed           1.88e+02   6.21e+01    3.03  0.00486 **
## Po1          1.93e+02   1.06e+02    1.82  0.07889 .
## Po2         -1.09e+02   1.17e+02   -0.93  0.35883
## LF          -6.64e+02   1.47e+03   -0.45  0.65465
## M.F          1.74e+01   2.04e+01    0.86  0.39900
## Pop         -7.33e-01   1.29e+00   -0.57  0.57385
## NW           4.20e+00   6.48e+00    0.65  0.52128
## U1          -5.83e+03   4.21e+03   -1.38  0.17624
## U2           1.68e+02   8.23e+01    2.04  0.05016 .
## Wealth       9.62e-02   1.04e-01    0.93  0.36075
## Ineq         7.07e+01   2.27e+01    3.11  0.00398 **
## Prob        -4.86e+03   2.27e+03   -2.14  0.04063 *
## Time        -3.48e+00   7.17e+00   -0.49  0.63071
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 43708)
##
##     Null deviance: 6880928  on 46  degrees of freedom
## Residual deviance: 1354946  on 31  degrees of freedom
## AIC: 650
##
## Number of Fisher Scoring iterations: 2
```

```
gaussian_crime_updated <- glm(Crime ~ M + Ed + Po1 + U2 + Ineq + Prob , data=
crime_data, family="gaussian")
summary(gaussian_crime_updated)

##
## Call:
## glm(formula = Crime ~ M + Ed + Po1 + U2 + Ineq + Prob, family = "gaussian"
,
##     data = crime_data)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -470.7   -78.4   -19.7   133.1   556.2
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -5040.5      899.8   -5.60  1.7e-06 ***
## M              105.0       33.3    3.15   0.0031 **
## Ed             196.5       44.8    4.39  8.1e-05 ***
## Po1            115.0       13.8    8.36  2.6e-10 ***
## U2              89.4       40.9    2.18   0.0348 *
## Ineq            67.7       13.9    4.85  1.9e-05 ***
## Prob         -3801.8     1528.1   -2.49   0.0171 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 40276)
##
##     Null deviance: 6880928  on 46  degrees of freedom
## Residual deviance: 1611057  on 40  degrees of freedom
## AIC: 640.2
##
## Number of Fisher Scoring iterations: 2

gaussian_model <- cv.glm(crime_data,gaussian_crime,K=Kfold_optimal) #using th
e same optimal K value

## Warning in cv.glm(crime_data, gaussian_crime, K = Kfold_optimal): 'K' has
been
## set to 16.000000

gaussian_model_updated<-cv.glm(crime_data,gaussian_crime_updated,K=Kfold_opti
mal)

## Warning in cv.glm(crime_data, gaussian_crime_updated, K = Kfold_optimal):
'K'
## has been set to 16.000000

# mean squared error is gaussian_model$delta[1]
```

```
rsq_glm<-1 - gaussian_model$delta[1]*nrow(crime_data)/SStotal
rsq_glm

## [1] 0.424

#With our gaussian model the rsq of our original data is less than the rsq of
our cross validation linear regression model (there is 23%error)

rsq_glm_updated<-1 - gaussian_model_updated$delta[1]*nrow(crime_data)/SStotal
rsq_glm_updated

## [1] 0.658

#With our gaussian model the rsq of our updated data is about the same of our
cross validation linear regression model for our updated model (there is 1% e
rror)

std_res<-rstandard(gaussian_crime)

std_res_updated<-rstandard(gaussian_crime_updated)
#Column bind standard residuals back to original data frame
d1<-cbind(crime_data,std_res)
d2<-cbind(crime_data_updated,std_res_updated)
#Sort standard residuals descending
d1[order(-std_res),]

d2[order(-std_res_updated),]

plot(d1$Crime,std_res,ylab='Standardized Residuals Crime data gaussian', xlab
='Crime')
abline(0,0)
```




```
plot(d2$Crime,std_res_updated,ylab='Standardized Residuals Crime data gaussia
n updated', xlab='Crime')
abline(0,0)
```


Standardized Residuals Crime data gaussian update
3
2
1
0
-1
-2
500
1000
1500
2000
Crime