ISYE6501HW4

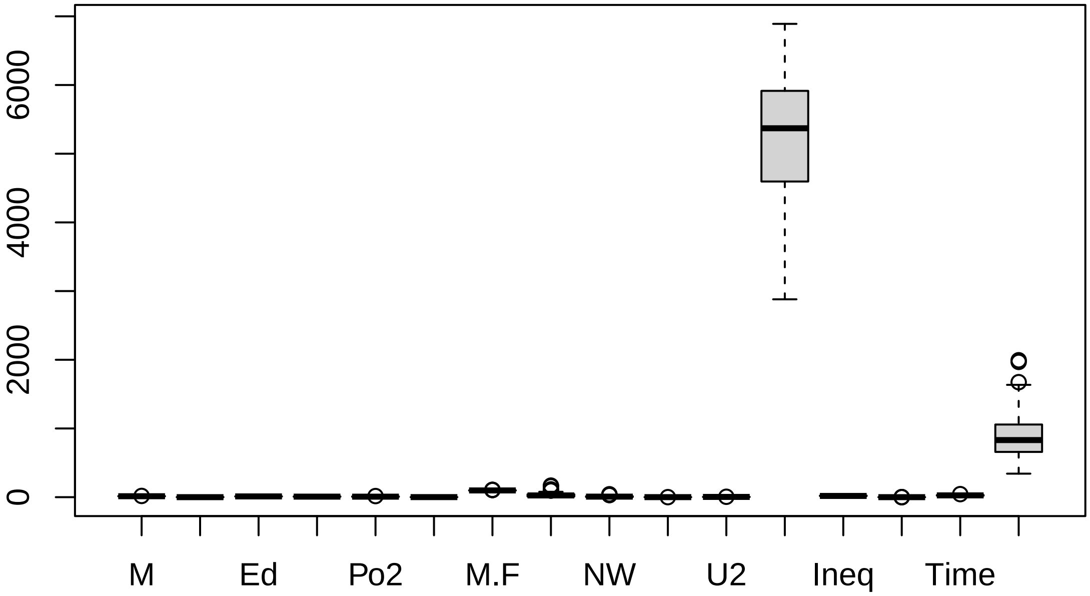
Yuanting Fan(904047984) Wenjia Hu(904057780) Sen Yang(904025995)

2024-09-17

Question 5.1

We implement boxplot to visualize features of 16 columns. For the last column (number of crimes per 100,000 people), the boxplot suggests that there are two possible outliers.

```
dataCrime <- data.frame(read.table("C:/Users/yuanting/Desktop/6501 hw4/data 5.1/uscrime
boxplot(dataCrime)
```



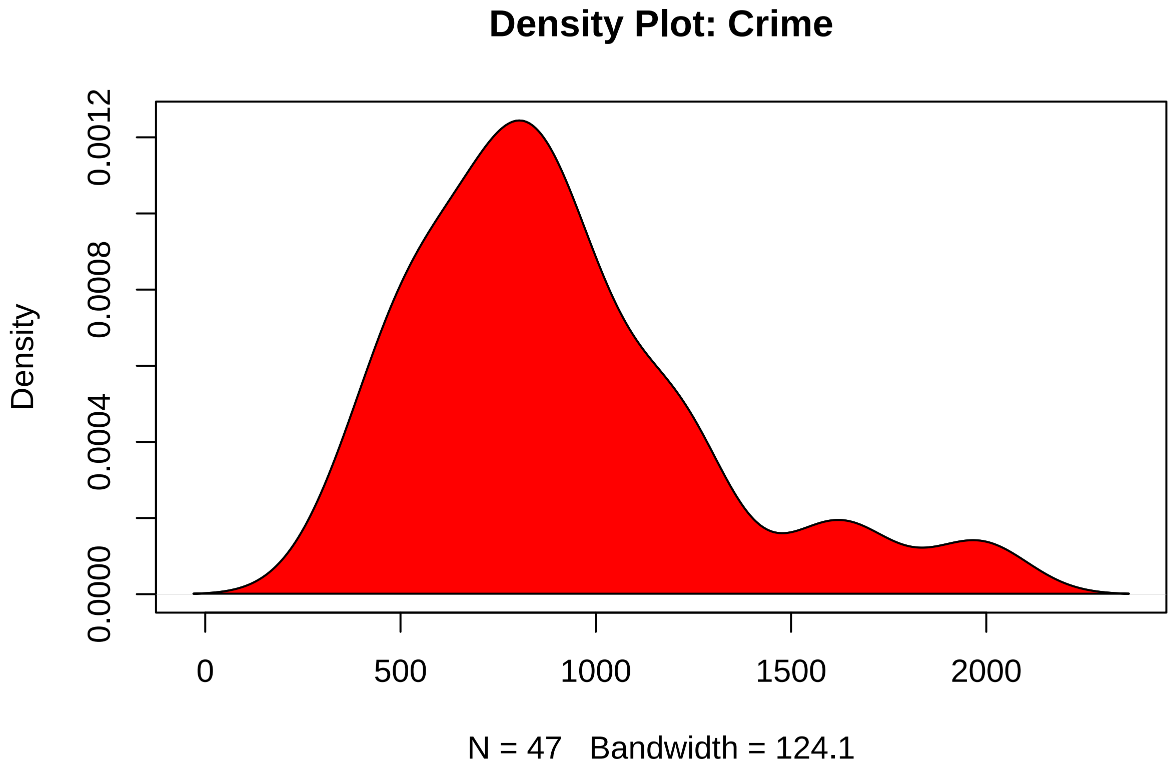
To check it in statistical methods, we run grubbs.test in R and the p-value equals 0.1577, not significant enough to reject the null hypothesis. Therefore, we can't say that highest value 1993 is an outlier even at 90% confidence interval.

```r
# install.packages("outliers", repos = "https://cran.r-project.org")
library(outliers)
grubbs.test(dataCrime[,16],two.sided = FALSE)
```

```
##
##  Grubbs test for one outlier
##
## data:  dataCrime[, 16]
## G = 2.81287, U = 0.82426, p-value = 0.07887
## alternative hypothesis: highest value 1993 is an outlier
```

Furthermore, we took a closer look at the data points of the last column.The density plot depicts taht distribution of crime data is not normally distributed: it is skewed toward the right.

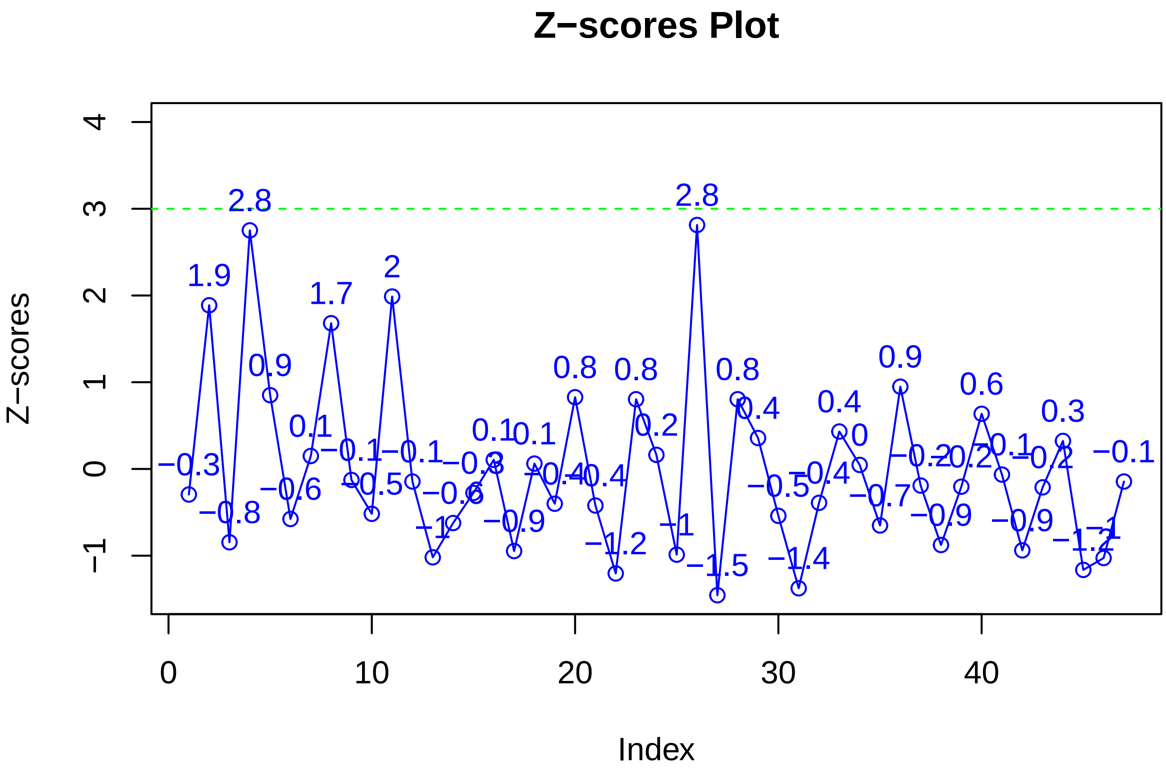
```r
plot(density(dataCrime$Crime), main="Density Plot: Crime")
polygon(density(dataCrime$Crime), col="red")
```



Also the z-score is not greater than 3. Statistically speaking, we conclude that there's no significant outlier in the crime data.

```r
# Calculate z-scores
z_scores <- scale(dataCrime$Crime)
```

```
plot(z_scores, type = "o", col = "blue", xlab = "Index", ylab = "Z-scores",
     main = "Z-scores Plot",ylim = c(min(z_scores), 4))
abline(h = 3, col = "green", lty = 2)
text(1:length(z_scores), z_scores, labels = round(z_scores, 1), pos = 3, col = "blue")
```



## Question 8.1

Situation: Predicting Employee Work Performance

We aim to develop a model to predict employee performance scores based on several measurable factors. This model will assist employers in making data-driven decisions regarding training, resource allocation, and employee development. Since the goal is to predict employee performance, a continuous variable that can be measured by metrics like sales revenue and KPI achievement rate, and the relationships between performance and the factors below are likely to be approximately linear or follow a predictable pattern, linear regression analysis would be appropriate.

Predictors

- **Workload**: The amount of work or number of tasks assigned to an employee. We usually measure workload by the number of tasks during a specific period. A higher workload might negatively influence performance due to stress or fatigue.
- **IQ (Intelligence Quotient)**: A measure of cognitive ability, generally assessed using

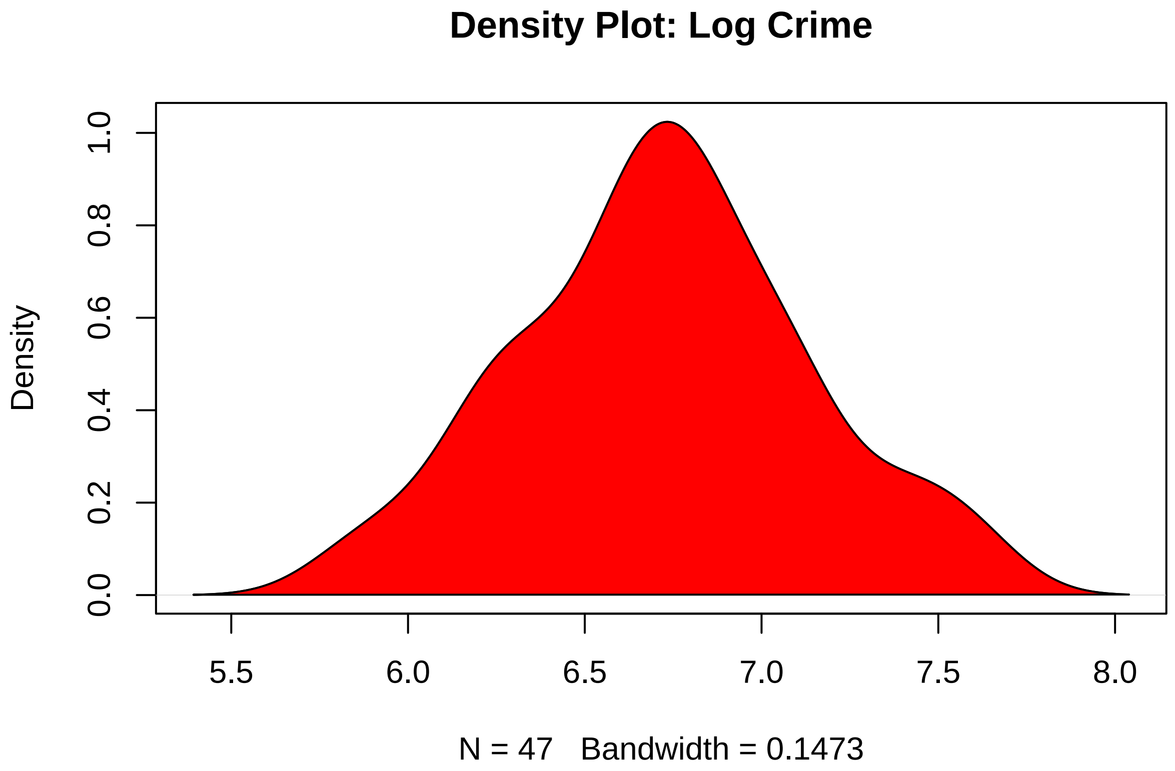
a questionnaire called the Binet Scale. Employees with higher IQs might perform better due to enhanced problem-solving and analytical skills.

- **Work Self-Efficacy**: A measure of how efficiently an employee completes tasks. We often use self-statement questionnaires, such as the Work Self-Efficacy Scale (WSES), to quantify work self-efficacy. Higher self-efficacy can lead to better overall performance.
- **Job Burnout**:The level of stress or exhaustion an employee experiences. The Maslach Burnout Inventory (MBI) is the most widely used tool for measuring job burnout. High levels of burnout can negatively impact performance.
- **Positive Feedback**: The frequency and quality of positive feedback received by the employee. We can track the number of positive feedback instances received over a period, including evaluations from colleagues, supervisors, or clients. More positive feedback can boost motivation and improve performance.

Question 8.2

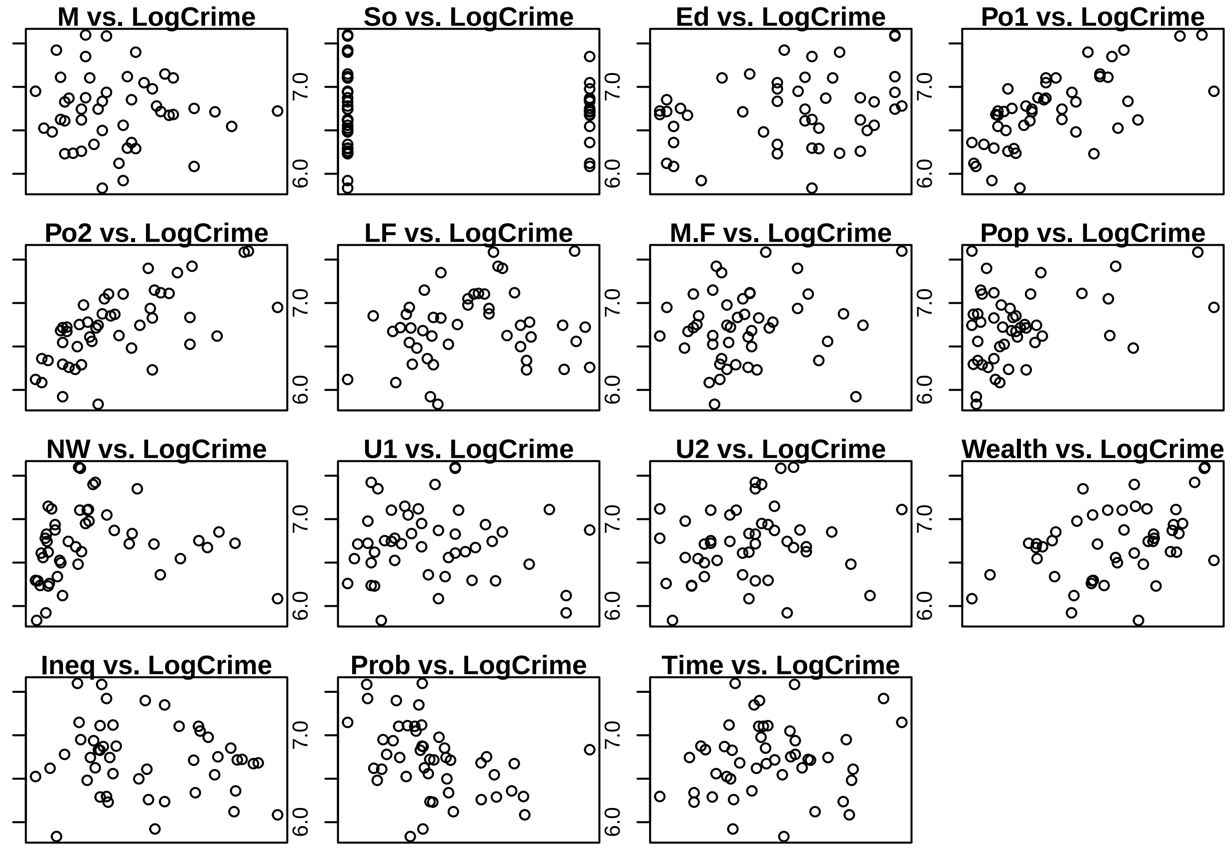
As shown above, the distribution of crime data is not normal.To run linear regression model, Log transformation is a common method to normalized the dependent variable. We can see that the Log(Crime) distribution has better normality than the original dependent variable (Crime)

```
plot(density(log(dataCrime$Crime)), main="Density Plot: Log Crime")
polygon(density(log(dataCrime$Crime)), col="red")
```



To visualize the potential relationship between each predictor and the response, we plotted scatter plots of the predictor variables against the log transformed response variable.No significant conclusion can be made at this stage.

```r
predictors <- dataCrime[, 1:15]
par(mfrow = c(4, 4), mar = c(1, 1, 1, 1))
for (predictor in names(predictors)) {
  plot(dataCrime[[predictor]], log(dataCrime$Crime),xaxt = 'n',
       main = paste(predictor, "vs. LogCrime"))
}
```



After the log transformation, we build our regression model with all predictors included.

```
model_lm_1 <- lm(log(Crime) ~.,data =dataCrime)
summary(model_lm_1)
```

```
##
## Call:
## lm(formula = log(Crime) ~ ., data = dataCrime)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.41697 -0.10961  0.01903  0.10971  0.47322
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.4345908  1.7884057   0.243  0.80960
## M            0.1160700  0.0458149   2.533  0.01657 *
## So           0.0917576  0.1633799   0.562  0.57841
## Ed           0.2147289  0.0681926   3.149  0.00361 **
## Po1          0.1862712  0.1165418   1.598  0.12012
## Po2         -0.1077276  0.1290273  -0.835  0.41015
## LF           0.1874034  1.6142245   0.116  0.90833
## M.F         -0.0061943  0.0223549  -0.277  0.78355
## Pop         -0.0009638  0.0014163  -0.681  0.50124
## NW           0.0047976  0.0071181   0.674  0.50530
## U1          -4.3033459  4.6242214  -0.931  0.35925
## U2           0.1717980  0.0904308   1.900  0.06680 .
## Wealth       0.0001737  0.0001139   1.526  0.13715
## Ineq         0.0808730  0.0249499   3.241  0.00284 **
## Prob        -6.0950555  2.4957820  -2.442  0.02050 *
## Time        -0.0080381  0.0078697  -1.021  0.31497
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2296 on 31 degrees of freedom
## Multiple R-squared:  0.7897, Adjusted R-squared:  0.688
## F-statistic: 7.761 on 15 and 31 DF,  p-value: 8.862e-07
```

The result of the linear regression model shows that predictors M (percentage of males aged 14–24 in total state population), Ed(mean years of schooling of the population aged 25 years or over), U2(unemployment rate of urban males 35–39), Ineq(income inequality), and Prob(probability of imprisonment) are statistically significant and are possible to have a meaningful impact on the response.The Multiple R-squared equals 0.7897.

However, further analysis reveals potential multicollinearity among the variables

```
cor(dataCrime, method = "pearson")
```

```
##                  M          So          Ed         Po1         Po2         LF
## M       1.00000000  0.58435534 -0.53023964 -0.50573690 -0.51317336 -0.1609488
## So      0.58435534  1.00000000 -0.70274132 -0.37263633 -0.37616753 -0.5054695
## Ed     -0.53023964 -0.70274132  1.00000000  0.48295213  0.49940958  0.5611780
## Po1    -0.50573690 -0.37263633  0.48295213  1.00000000  0.99358648  0.1214932
## Po2    -0.51317336 -0.37616753  0.49940958  0.99358648  1.00000000  0.1063496
## LF     -0.16094882 -0.50546948  0.56117795  0.12149320  0.10634960  1.0000000
## M.F    -0.02867993 -0.31473291  0.43691492  0.03376027  0.02284250  0.5135588
## Pop    -0.28063762 -0.04991832 -0.01722740  0.52628358  0.51378940 -0.1236722
## NW      0.59319826  0.76710262 -0.66488190 -0.21370878 -0.21876821 -0.3412144
## U1     -0.22438060 -0.17241931  0.01810345 -0.04369761 -0.05171199 -0.2293997
## U2     -0.24484339  0.07169289 -0.21568155  0.18509304  0.16922422 -0.4207625
## Wealth -0.67005506 -0.63694543  0.73599704  0.78722528  0.79426205  0.2946323
## Ineq    0.63921138  0.73718106 -0.76865789 -0.63050025 -0.64815183 -0.2698865
## Prob    0.36111641  0.53086199 -0.38992286 -0.47324704 -0.47302729 -0.2500861
## Time    0.11451072  0.06681283 -0.25397355  0.10335774  0.07562665 -0.1236404
## Crime  -0.08947240 -0.09063696  0.32283487  0.68760446  0.66671414  0.1888663
##                M.F         Pop          NW           U1          U2
## M      -0.02867993 -0.28063762  0.59319826 -0.224380599 -0.24484339
## So     -0.31473291 -0.04991832  0.76710262 -0.172419305  0.07169289
## Ed      0.43691492 -0.01722740 -0.66488190  0.018103454 -0.21568155
## Po1     0.03376027  0.52628358 -0.21370878 -0.043697608  0.18509304
## Po2     0.02284250  0.51378940 -0.21876821 -0.051711989  0.16922422
## LF      0.51355879 -0.12367222 -0.34121444 -0.229399684 -0.42076249
## M.F     1.00000000 -0.41062750 -0.32730454  0.351891900 -0.01869169
## Pop    -0.41062750  1.00000000  0.09515301 -0.038119948  0.27042159
## NW     -0.32730454  0.09515301  1.00000000 -0.156450020  0.08090829
## U1      0.35189190 -0.03811995 -0.15645002  1.000000000  0.74592482
## U2     -0.01869169  0.27042159  0.08090829  0.745924815  1.00000000
## Wealth  0.17960864  0.30826271 -0.59010707  0.044857202  0.09207166
## Ineq   -0.16708869 -0.12629357  0.67731286 -0.063832178  0.01567818
## Prob   -0.05085826 -0.34728906  0.42805915 -0.007469032 -0.06159247
## Time   -0.42769738  0.46421046  0.23039841 -0.169852838  0.10135833
## Crime   0.21391426  0.33747406  0.03259884 -0.050477918  0.17732065
##               Wealth        Ineq         Prob          Time       Crime
## M      -0.6700550558  0.63921138  0.361116408  0.1145107190 -0.08947240
## So     -0.6369454328  0.73718106  0.530861993  0.0668128312 -0.09063696
## Ed      0.7359970363 -0.76865789 -0.389922862 -0.2539735471  0.32283487
## Po1     0.7872252807 -0.63050025 -0.473247036  0.1033577449  0.68760446
## Po2     0.7942620503 -0.64815183 -0.473027293  0.0756266536  0.66671414
## LF      0.2946323090 -0.26988646 -0.250086098 -0.1236404364  0.18886635
## M.F     0.1796086363 -0.16708869 -0.050858258 -0.4276973791  0.21391426
## Pop     0.3082627091 -0.12629357 -0.347289063  0.4642104596  0.33747406
## NW     -0.5901070652  0.67731286  0.428059153  0.2303984071  0.03259884
## U1      0.0448572017 -0.06383218 -0.007469032 -0.1698528383 -0.05047792
## U2      0.0920716601  0.01567818 -0.061592474  0.1013583270  0.17732065
## Wealth  1.0000000000 -0.88399728 -0.555334708  0.0006485587  0.44131995
## Ineq   -0.8839972758  1.00000000  0.465321920  0.1018228182 -0.17902373
## Prob   -0.5553347075  0.46532192  1.000000000 -0.4362462614 -0.42742219
## Time    0.0006485587  0.10182282 -0.436246261  1.0000000000  0.14986606
## Crime   0.4413199490 -0.17902373 -0.427422188  0.1498660617  1.00000000
```

The correlation analysis returns high correlation between some of the variables which can lead to multicollinearity, making it difficult to determine the contribution of each variable. For instance, there is a correlation as high as 0.99 between Po1 and Po2. The variable wealth also has a relatively high correlation with some other variables. These results prompt us to perform subset selection on the variables to identify the best possible predictor combinations.

Here, we use some of the codes from the crime data source website to help select subsets.We use leap() function with nbest=2 to get the best two one-variable models,best two 2-variable models, all the way up to best two 15-variable models.

```
library(leaps)
leaps.crime <- leaps(dataCrime[, 1:15], dataCrime$Crime, nbest = 2)
leaps.tab <- data.frame(p = leaps.crime$size, Cp = leaps.crime$Cp)
```

For the Mallows' Cp values from the leaps(), we aim for a small Cp value.while still close to P so that the model can strike a good balance between model fitting and complexity. According to the result, p=7 Cp=3.8596, p=8 Cp =4.4889, and p = 9, Cp = 4.2449 could be good choices.

```
print(leaps.tab)
```

```
##     p        Cp
## 1   2 39.996975
## 2   2 44.451000
## 3   3 25.070558
## 4   3 27.886466
## 5   4 13.639362
## 6   4 16.670956
## 7   5 10.161988
## 8   5 10.263192
## 9   6  6.257739
## 10  6  7.563549
## 11  7  3.859603
## 12  7  6.278932
## 13  8  4.488920
## 14  8  4.605138
## 15  9  4.244947
## 16  9  5.090015
## 17 10  5.638805
## 18 10  5.862945
## 19 11  7.127562
## 20 11  7.335447
## 21 12  8.745335
## 22 12  8.968262
## 23 13 10.478229
## 24 13 10.580450
## 25 14 12.237239
## 26 14 12.249527
## 27 15 14.000654
## 28 15 14.204002
## 29 16 16.000000
```

It shows which predictors are chosen for p=7 Cp=3.8596, p=8 Cp =4.4889, and p = 9, Cp = 4.2449

```
leaps.crime$which
```

```
##         1     2     3     4     5     6     7     8     9     A     B     C
## 1  FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## 1  FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## 2  FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## 2   TRUE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## 3  FALSE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## 3  FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
## 4   TRUE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## 4  FALSE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## 5   TRUE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## 5   TRUE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE
## 6   TRUE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE
## 6   TRUE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
## 7   TRUE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE
## 7   TRUE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE
## 8   TRUE FALSE  TRUE  TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE FALSE
## 8   TRUE FALSE  TRUE  TRUE FALSE FALSE FALSE  TRUE FALSE  TRUE  TRUE FALSE
## 9   TRUE FALSE  TRUE  TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE
## 9   TRUE FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE FALSE
## 10  TRUE FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE
## 10  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE
## 11  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE
## 11  TRUE FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
## 12  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
## 12  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE
## 13  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
## 13  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
## 14  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
## 14  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
## 15  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##         D     E     F
## 1  FALSE FALSE FALSE
## 1  FALSE FALSE FALSE
## 2   TRUE FALSE FALSE
## 2  FALSE FALSE FALSE
## 3   TRUE FALSE FALSE
## 3   TRUE FALSE FALSE
## 4   TRUE FALSE FALSE
## 4   TRUE  TRUE FALSE
## 5   TRUE  TRUE FALSE
## 5   TRUE FALSE FALSE
## 6   TRUE  TRUE FALSE
## 6   TRUE  TRUE FALSE
## 7   TRUE  TRUE FALSE
## 7   TRUE  TRUE FALSE
## 8   TRUE  TRUE FALSE
## 8   TRUE  TRUE FALSE
## 9   TRUE  TRUE FALSE
## 9   TRUE  TRUE FALSE
## 10  TRUE  TRUE FALSE
## 10  TRUE  TRUE FALSE
## 11  TRUE  TRUE FALSE
## 11  TRUE  TRUE FALSE
## 12  TRUE  TRUE FALSE
## 12  TRUE  TRUE FALSE
## 13  TRUE  TRUE FALSE
## 13  TRUE  TRUE  TRUE
## 14  TRUE  TRUE  TRUE
## 14  TRUE  TRUE  TRUE
## 15  TRUE  TRUE  TRUE
```

Then we use the selected independent variables to build new models. The result shows that the model with 7 predictors performs better as all 7 predictors are statistically significant or slightly higher than the 0.05 threshold.

```r
model_lm_2 <- lm(log(Crime) ~ M+Ed+Po1+U2+Wealth+Ineq+Prob,data =dataCrime)
model_lm_3 <- lm(log(Crime) ~ M+Ed+Po1+M.F+U1+U2+Ineq+Prob,data =dataCrime)
model_lm_4 <- lm(log(Crime) ~ M+Ed+Po1+M.F+U1+U2+Wealth+Ineq+Prob,data =dataCrime)
summary(model_lm_2)
```

```
##
## Call:
## lm(formula = log(Crime) ~ M + Ed + Po1 + U2 + Wealth + Ineq +
##     Prob, data = dataCrime)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.41172 -0.09896  0.01272  0.15639  0.40938
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.8178767  1.1689986  -0.700 0.488305
## M            0.1309364  0.0371378   3.526 0.001097 **
## Ed           0.1863878  0.0508344   3.667 0.000731 ***
## Po1          0.0993984  0.0185764   5.351 4.12e-06 ***
## U2           0.0856809  0.0452397   1.894 0.065669 .
## Wealth       0.0001855  0.0001022   1.815 0.077167 .
## Ineq         0.0931915  0.0192316   4.846 2.04e-05 ***
## Prob        -3.3773533  1.7256123  -1.957 0.057510 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2204 on 39 degrees of freedom
## Multiple R-squared:  0.7562, Adjusted R-squared:  0.7124
## F-statistic: 17.28 on 7 and 39 DF,  p-value: 3.619e-10
```

```r
summary(model_lm_3)
```

```
##
## Call:
## lm(formula = log(Crime) ~ M + Ed + Po1 + M.F + U1 + U2 + Ineq +
##     Prob, data = dataCrime)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.46379 -0.13284  0.02277  0.13958  0.42131
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.36082    1.36931  -0.264 0.793587
## M            0.11594    0.03840   3.020 0.004505 **
## Ed           0.22610    0.06047   3.739 0.000607 ***
## Po1          0.10381    0.01779   5.834 9.62e-07 ***
## M.F          0.00995    0.01559   0.638 0.527177
## U1          -6.49946    3.82761  -1.698 0.097675 .
## U2           0.21315    0.08309   2.565 0.014376 *
## Ineq         0.06783    0.01600   4.239 0.000138 ***
## Prob        -4.02555    1.70864  -2.356 0.023734 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2241 on 38 degrees of freedom
## Multiple R-squared:  0.7544, Adjusted R-squared:  0.7027
## F-statistic: 14.59 on 8 and 38 DF,  p-value: 1.806e-09
```

```r
summary(model_lm_4)
```

```
##
## Call:
## lm(formula = log(Crime) ~ M + Ed + Po1 + M.F + U1 + U2 + Wealth +
##     Ineq + Prob, data = dataCrime)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.40175 -0.14041  0.02824  0.15134  0.36390
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.0889467  1.4305191  -0.761 0.451347
## M            0.1277930  0.0385714   3.313 0.002070 **
## Ed           0.2078524  0.0606889   3.425 0.001519 **
## Po1          0.0895982  0.0198722   4.509 6.36e-05 ***
## M.F          0.0061592  0.0155381   0.396 0.694092
## U1          -5.3057026  3.8468054  -1.379 0.176098
## U2           0.1852249  0.0837853   2.211 0.033318 *
## Wealth       0.0001581  0.0001047   1.510 0.139508
## Ineq         0.0869787  0.0202089   4.304 0.000118 ***
## Prob        -3.4103712  1.7292329  -1.972 0.056098 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2205 on 37 degrees of freedom
## Multiple R-squared:  0.7686, Adjusted R-squared:  0.7124
## F-statistic: 13.66 on 9 and 37 DF,  p-value: 2.544e-09
```

We can also use AIC to check the fit of these models.Since the dataset is quite small, the corrected AIC would be a better choice to avoid overfitting.

```
library(AICcmodavg)
aic_c_value_1 <- AICc(model_lm_1)
aic_c_value_2 <- AICc(model_lm_2)
aic_c_value_3 <- AICc(model_lm_3)
aic_c_value_4 <- AICc(model_lm_4)
print(aic_c_value_1)
```

```
## [1] 30.61853
```

```
print(aic_c_value_2)
```

```
## [1] 5.338573
```

```
print(aic_c_value_3)
```

```
## [1] 8.927205
```

```
print(aic_c_value_4)
```

```
## [1] 9.547938
```

For AIC values, we prefer the models with a smaller AIC value. Compared with the original model with all the 15 predictors, the new models all show better performance in terms of AIC, indicating a better balance between model fitting and complexity.As the AIC for the model_lm_2 is the smallest, it seems to be a good choice.

For model_lm_2, the p-values for U2, Wealth, and Prob are 0.065, 0.077, 0.057, showing that they are marginally insignificant.A closer look would find that the predictor wealth shows a high correlation of -0.88 with Ineq, which may explain why its p value is not highly significant. Also, the coefficient for wealth is only 0.0001855, close to zero and indicating a small effect from this predictor on the final outcome.

For U2 and Prob, they do not show a high correlation with the other chosen predictors.Moreover, their estimated coefficients are 0.0856809 and -3.3773533, indicating they would have meaningful impact on the final model.

In practice, we believe that U2(unemployment rate of urban males 35–39) and Prob(ratio of number of commitments to number of offenses) would be likely to influence the crime rate in a region. Therefore we decide to remove wealth from our model while keeping U2 and Prob.

```r
model_lm_5 <- lm(log(Crime) ~ M+Ed+Po1+U2+Ineq+Prob,data =dataCrime)
summary(model_lm_5)
```

```
##
## Call:
## lm(formula = log(Crime) ~ M + Ed + Po1 + U2 + Ineq + Prob, data = dataCrime)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.47574 -0.12217  0.01661  0.14716  0.49322
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.31516    1.01640   0.310 0.758110
## M            0.11927    0.03761   3.171 0.002914 **
## Ed           0.20987    0.05055   4.152 0.000168 ***
## Po1          0.11902    0.01554   7.661 2.29e-09 ***
## U2           0.09523    0.04620   2.061 0.045835 *
## Ineq         0.07206    0.01574   4.578 4.50e-05 ***
## Prob        -4.10406    1.72603  -2.378 0.022287 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2267 on 40 degrees of freedom
## Multiple R-squared:  0.7356, Adjusted R-squared:  0.6959
## F-statistic: 18.54 on 6 and 40 DF,  p-value: 3.618e-10
```

The AIC for the new model is slightly higher than the model_lm_2, but still a relatively small value compared with the other model choices,indicating that this is a model performing a good balance between model complexity(less predictors) and fitting.

```r
aic_c_value_5 <- AICc(model_lm_5)
print(aic_c_value_5)
```

```
## [1] 6.075739
```

In conclusion, our final regression model is log(Crime) = 0.11927*M+0.20987*Ed +0.11902*Po1+0.09523*U2+0.07206*Ineq-4.10406*Prob

For the new 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

Based on the new data, the prediction for the observed crime rate in the city is 919.4647.

```
log_Crime <- 0.11927*14.0+0.20987*10.0+0.11902*12.0+0.09523*3.6+0.07206*20.1-4.10406*0.0
Crime_newdata <- exp(log_Crime)
print(Crime_newdata)
```

```
## [1] 919.4647
```

One last matter we'd like to point out is we double check whether the highest value in crime data affect the regression model or whether is it wrong to keep it in regression process. Via Cooks'distance stats, the answer is NO: the highest value does not affect the regression model thus it's proper to keep it in regression.

```
cooks_d <- cooks.distance(model_lm_5)
# Print Cook's distance
influential_points <- which(cooks_d > 1)
influential_data <- dataCrime[influential_points, ]
# Print the filtered data frame
print(influential_data)
```

```
##  [1] M      So     Ed     Po1    Po2    LF     M.F    Pop    NW     U1
## [11] U2     Wealth Ineq   Prob   Time   Crime
## <0  > ( 0-  row.names)
```