```
## Loading required package: carData

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

##
## Please cite as:

##  Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary Statistics Tables.

##  R package version 5.2.2. https://CRAN.R-project.org/package=stargazer

##
## Attaching package: 'dplyr'

## The following object is masked from 'package:car':
##
##     recode

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

##
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':
##
##     smiths
```

```r
#read in the data
#data <- read.csv(file = 'H:/ROL/MIDS/W203 Stats/lab_3/crime_v2.csv') #Robert

data <- read.csv(file = '~/Desktop/W203/w203_lab3-master/crime_v2.csv') #Praveen
```

2. A Model Building Process

Exploratory Data Analaysis

We started by conducting exploratory data analysis. First, we read the original paper to get a better understanding of each variable. We defined the variables in the table below and grouped them into five groups in order to get a better handle on them.

```r
crime_count <- c(1:25)
data_variables <- c("county","year","crmrte","prbarr","prbconv","prbpris","avgsen","polpc","density","ta
data_description <- c("county identifier","1987","crimes committed per person","'probability' of arrest
data_group <- c("Control","","","Deterrent","Deterrent","Deterrent","Deterrent","Deterrent","Demographi
data_notes <- c("","","ratio of FBI index crimes to county population","ratio of arrests to offenses","
)
data_headers <- c("Variable", "Description", "Group", "Note")
data_table <- data.frame(data_variables, data_description, data_group, data_notes)
kable(data_table, col.names = data_headers, caption = "Descriptions and Groups of Variables")
```

Table 1: Descriptions and Groups of Variables

Variable	Description	Group	Note
county	county identifier	Control	
year	1987		
crmrte	crimes committed per person		ratio of FBI index crimes to county population
prbarr	‘probability’ of arrest	Deterrent	ratio of arrests to offenses
prbconv	‘probability’ of conviction	Deterrent	ratio of convictions to arrests
prbpris	‘probability’ of prison sentence	Deterrent	proportion of total convictions resulting in prison sentences
avgsen	avg. sentence, days	Deterrent	average sentence in days
polpc	police per capita	Deterrent	
density	people per sq. mile	Demographic	country population divided by county land area
taxpc	tax revenue per capita	Demographic	
west	=1 if in western N.C.	Region	dummy
central	=1 if in central N.C.	Region	dummy
urban	=1 if in SMSA	Urban	dummy
pctmin80	perc. minority, 1980	Demographic	proportion of country population that is minority or nonwhite
wcon	weekly wage, construction	Wages	average weekly wage in that sector
wtuc	wkly wge, trns, util, commun	Wages	average weekly wage in that sector
wtrd	wkly wge, whlesle, retail trade	Wages	average weekly wage in that sector
wfir	wkly wge, fin, ins, real est	Wages	average weekly wage in that sector
wser	wkly wge, service industry	Wages	average weekly wage in that sector
wmfg	wkly wge, manufacturing	Wages	average weekly wage in that sector
wfed	wkly wge, fed employees	Wages	average weekly wage in that sector
wsta	wkly wge, state employees	Wages	average weekly wage in that sector
wloc	wkly wge, local gov emps	Wages	average weekly wage in that sector
mix	offense mix: face-to-face/other	Demographic	ratio of face-to-face crimes (robbery, assault, rape) to non-face
pctymle	percent young male	Demographic	proportion of country population that is male between 15 and

To get a better sense of the data set the summary function was run.

```r
summary(data)
```

```
##      county           year        crmrte             prbarr       
##  Min.   :  1.0   Min.   :87   Min.   :0.005533   Min.   :0.09277  
##  1st Qu.: 52.0   1st Qu.:87   1st Qu.:0.020927   1st Qu.:0.20568  
##  Median :105.0   Median :87   Median :0.029986   Median :0.27095  
##  Mean   :101.6   Mean   :87   Mean   :0.033400   Mean   :0.29492  
##  3rd Qu.:152.0   3rd Qu.:87   3rd Qu.:0.039642   3rd Qu.:0.34438  
##  Max.   :197.0   Max.   :87   Max.   :0.098966   Max.   :1.09091  
##  NA's   :6       NA's   :6    NA's   :6          NA's   :6        
##         prbconv      prbpris           avgsen           polpc
##             : 5   Min.   :0.1500   Min.   : 5.380   Min.   :0.000746
##  0.588859022: 2   1st Qu.:0.3648   1st Qu.: 7.340   1st Qu.:0.001231
##  `          : 1   Median :0.4234   Median : 9.100   Median :0.001485
##  0.068376102: 1   Mean   :0.4108   Mean   : 9.647   Mean   :0.001702
##  0.140350997: 1   3rd Qu.:0.4568   3rd Qu.:11.420   3rd Qu.:0.001877
##  0.154451996: 1   Max.   :0.6000   Max.   :20.700   Max.   :0.009054
##  (Other)    :86   NA's   :6        NA's   :6        NA's   :6
##     density            taxpc             west           central
##  Min.   :0.00002   Min.   : 25.69   Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.54741   1st Qu.: 30.66   1st Qu.:0.0000   1st Qu.:0.0000
##  Median :0.96226   Median : 34.87   Median :0.0000   Median :0.0000
##  Mean   :1.42884   Mean   : 38.06   Mean   :0.2527   Mean   :0.3736
##  3rd Qu.:1.56824   3rd Qu.: 40.95   3rd Qu.:0.5000   3rd Qu.:1.0000
##  Max.   :8.82765   Max.   :119.76   Max.   :1.0000   Max.   :1.0000
##  NA's   :6         NA's   :6        NA's   :6        NA's   :6
##      urban            pctmin80           wcon            wtuc
##  Min.   :0.00000   Min.   : 1.284   Min.   :193.6   Min.   :187.6
##  1st Qu.:0.00000   1st Qu.: 9.845   1st Qu.:250.8   1st Qu.:374.6
##  Median :0.00000   Median :24.312   Median :281.4   Median :406.5
##  Mean   :0.08791   Mean   :25.495   Mean   :285.4   Mean   :411.7
##  3rd Qu.:0.00000   3rd Qu.:38.142   3rd Qu.:314.8   3rd Qu.:443.4
##  Max.   :1.00000   Max.   :64.348   Max.   :436.8   Max.   :613.2
##  NA's   :6         NA's   :6        NA's   :6       NA's   :6
##       wtrd            wfir            wser             wmfg
##  Min.   :154.2   Min.   :170.9   Min.   : 133.0   Min.   :157.4
##  1st Qu.:190.9   1st Qu.:286.5   1st Qu.: 229.7   1st Qu.:288.9
##  Median :203.0   Median :317.3   Median : 253.2   Median :320.2
##  Mean   :211.6   Mean   :322.1   Mean   : 275.6   Mean   :335.6
##  3rd Qu.:225.1   3rd Qu.:345.4   3rd Qu.: 280.5   3rd Qu.:359.6
##  Max.   :354.7   Max.   :509.5   Max.   :2177.1   Max.   :646.9
##  NA's   :6       NA's   :6       NA's   :6        NA's   :6
##       wfed            wsta            wloc            mix
##  Min.   :326.1   Min.   :258.3   Min.   :239.2   Min.   :0.01961
##  1st Qu.:400.2   1st Qu.:329.3   1st Qu.:297.3   1st Qu.:0.08074
##  Median :449.8   Median :357.7   Median :308.1   Median :0.10186
##  Mean   :442.9   Mean   :357.5   Mean   :312.7   Mean   :0.12884
##  3rd Qu.:478.0   3rd Qu.:382.6   3rd Qu.:329.2   3rd Qu.:0.15175
##  Max.   :598.0   Max.   :499.6   Max.   :388.1   Max.   :0.46512
##  NA's   :6       NA's   :6       NA's   :6       NA's   :6
##     pctymle
##  Min.   :0.06216
##  1st Qu.:0.07443
##  Median :0.07771
##  Mean   :0.08396
##  3rd Qu.:0.08350
##  Max.   :0.24871
##  NA's   :6
```

This function provides a high level view of each variable. Six rows have missing values for all variables. In addition, there is one duplicate row. Also the variable prbconv is loaded as a factor, so it needs to be converted to numeric. These issues are handled below to create the initial data set.

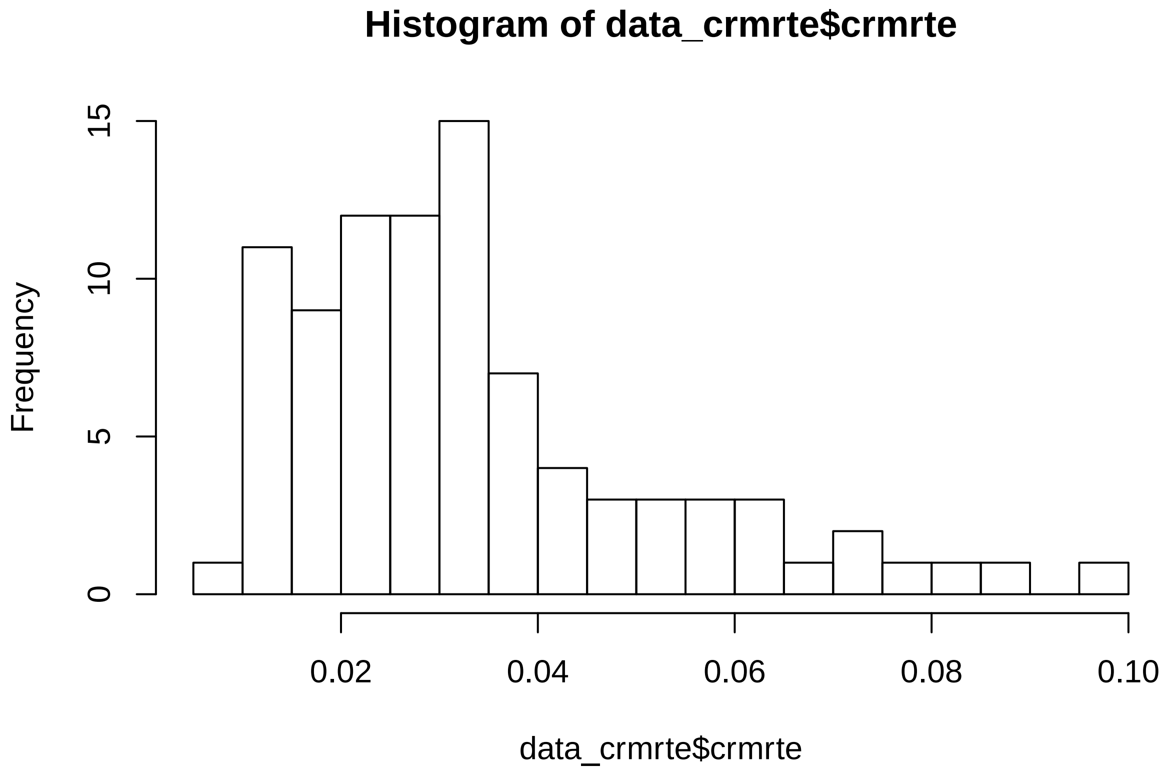
```
#eliminate N/A's
data_crmrte <- data[!is.na(data$crmrte),]

#remove duplicates
data_crmrte <- data_crmrte %>% distinct()

#make prbconv numeric
data_crmrte$prbconv <- as.numeric(as.character(data_crmrte$prbconv))
```

With 25 original variables in the data set the natural place to start is with the dependent variable, crmrte. To get a better sense of this variable, the distribution is graphed below.

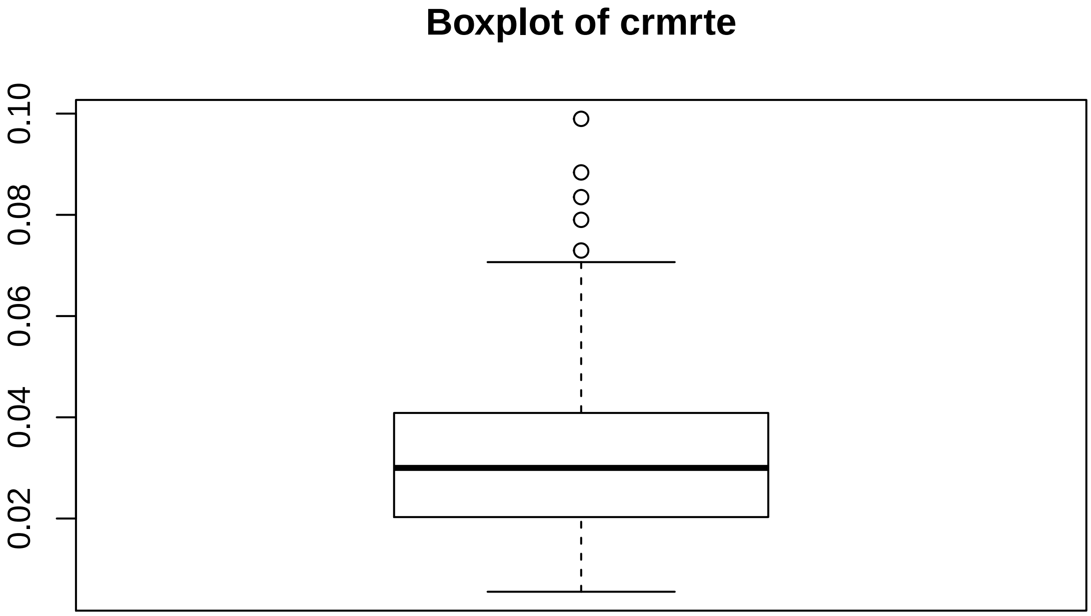
```
quantile(data_crmrte$crmrte, c(0, .01, .05, .10, .25, .50,  .75, .90, .95, .99, 1.0))
```

```
##         0%         1%         5%        10%        25%        50%        75% 
## 0.00553320 0.01006330 0.01235660 0.01418007 0.02060425 0.03000200 0.04024925 
##        90%        95%        99%       100% 
## 0.06054659 0.07191830 0.08954881 0.09896590
```

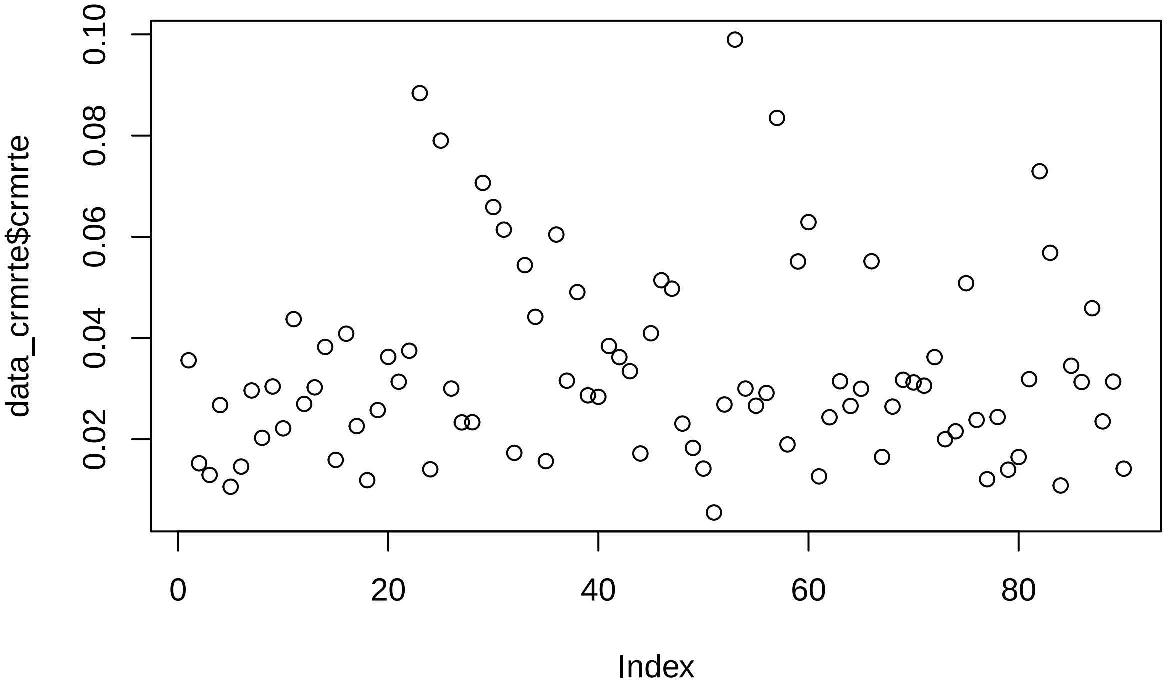
```
hist(data_crmrte$crmrte,breaks=20)
```



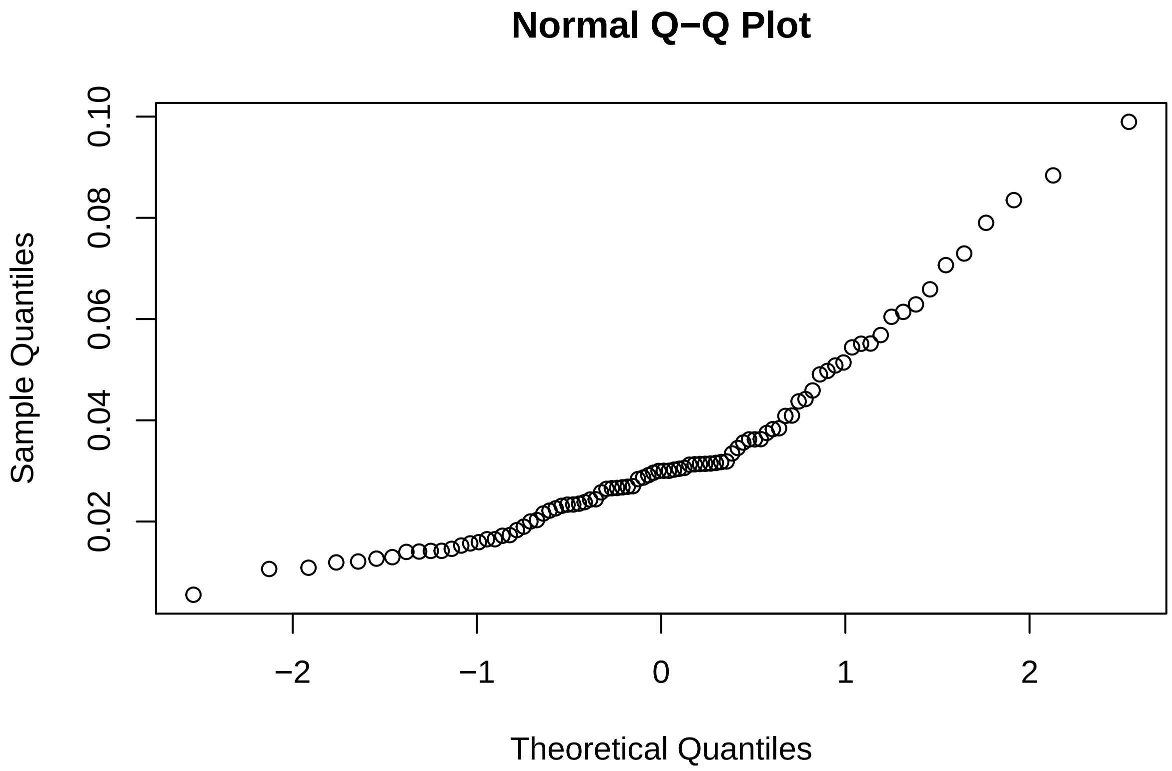
```
boxplot(data_crmrte$crmrte, main="Boxplot of crmrte")
```



```r
plot(data_crmrte$crmrte)
```



```
qqnorm(data_crmrte$crmrte)
```



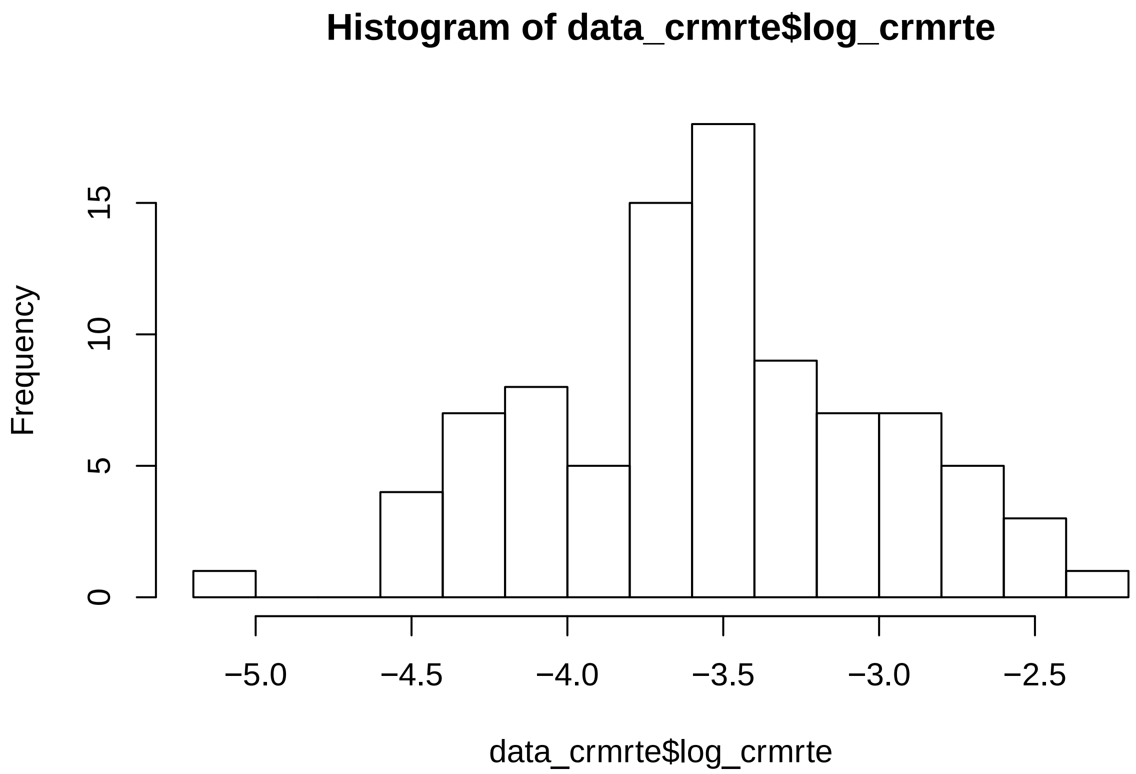
There are several outliers in the variable crmrte and the distribution is right skewed. We have ninety observations so perhaps we normality is not a top concern but this distribution is not perfectly normal. The largest outliers on the right side of the distribution are examined. Unfortunately, looking at these observations in a dataframe does not show any obvious patterns (e.g. they are all in the same region, they all have similar values of a variable like density, etc.)

```
data_crmrte[data_crmrte$crmrte > 0.065,]
```

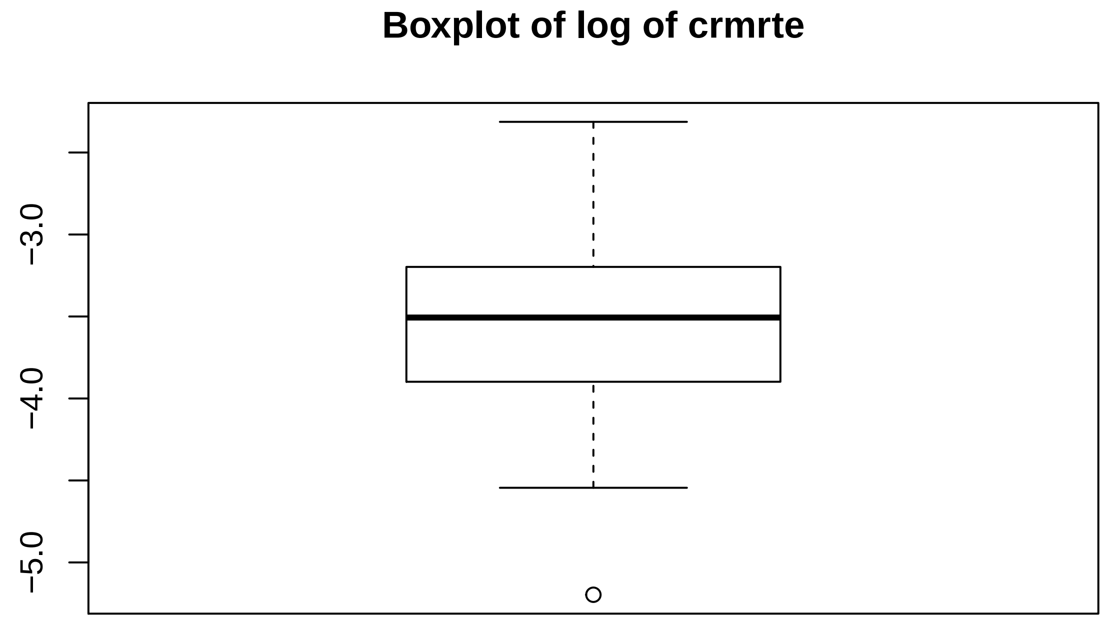
```
##    county year    crmrte   prbarr  prbconv  prbpris avgsen      polpc   density
## 23     51   87 0.0883849 0.155248 0.259833 0.407628  11.93 0.00190802 3.9345510
## 25     55   87 0.0790163 0.224628 0.207831 0.304348  13.57 0.00400962 0.5115089
## 29     63   87 0.0706599 0.133225 0.459216 0.363636  11.51 0.00237609 5.6744967
## 30     65   87 0.0658801 0.287330 0.154452 0.403922   9.84 0.00185739 1.1679842
## 53    119   87 0.0989659 0.149094 0.347800 0.486183   7.13 0.00223135 8.8276520
## 57    129   87 0.0834982 0.236601 0.393413 0.415158   9.57 0.00255849 6.2864866
## 82    181   87 0.0729479 0.182590 0.343023 0.548023   7.06 0.00172948 1.5702811
##        taxpc west central urban pctmin80     wcon     wtuc     wtrd     wfir
## 23  35.69936    0       0     1 37.77920 283.6695 412.4720 213.7524 324.8357
## 25 119.76145    0       0     0  6.49622 309.5238 445.2762 189.7436 284.5933
## 29  50.19918    0       1     1 38.22300 349.3267 548.9865 238.9154 435.1107
## 30  30.62824    0       0     0 51.69320 362.1527 540.1061 209.0579 316.2955
## 53  75.67243    0       1     1 28.54600 436.7666 548.3239 354.6761 509.4655
## 57  67.67963    0       0     1 23.04410 315.5760 392.0999 220.4530 363.2880
## 82  27.59179    0       1     0 44.62830 244.8362 365.4716 279.2273 325.0271
##        wser   wmfg   wfed   wsta   wloc        mix    pctymle
## 23 257.3344 441.72 433.94 367.34 333.71 0.10474319 0.14223780
## 25 221.3903 319.21 338.91 361.68 326.08 0.08437271 0.07613807
## 29 391.3081 646.85 563.77 415.51 362.58 0.07585382 0.09468981
## 30 216.4589 313.71 543.03 348.88 329.16 0.09364294 0.07622346
## 53 354.3007 494.30 568.40 329.22 379.77 0.16869897 0.07916495
## 57 292.7027 464.49 548.49 421.36 319.08 0.07871422 0.08109921
## 82 213.5822 290.69 453.53 317.23 286.45 0.10003893 0.07977433
```

For campaign purposes, we want to predict crime. We want our candidate to be able to say that he or she can reduce crime in order to win votes. What is the most effective way to convey that? Using crime rate as it appears in the data set is using the level of crime rate and would suggest the following statement as a campaign slogan - “I can reduce crime to this rate by doing x, y, and z”. Transforming crime rate into the log of crime rate allows for the statement “I can reduce crime by n% by doing x, y, and z.” We find the latter more powerful and meaningful to voters since voters have no idea about the level of crime rates. In addition, we will show that the transformation of crime rate imrpoves the normaility and distribution of the variable, which will often reduce skew in the errors as well.

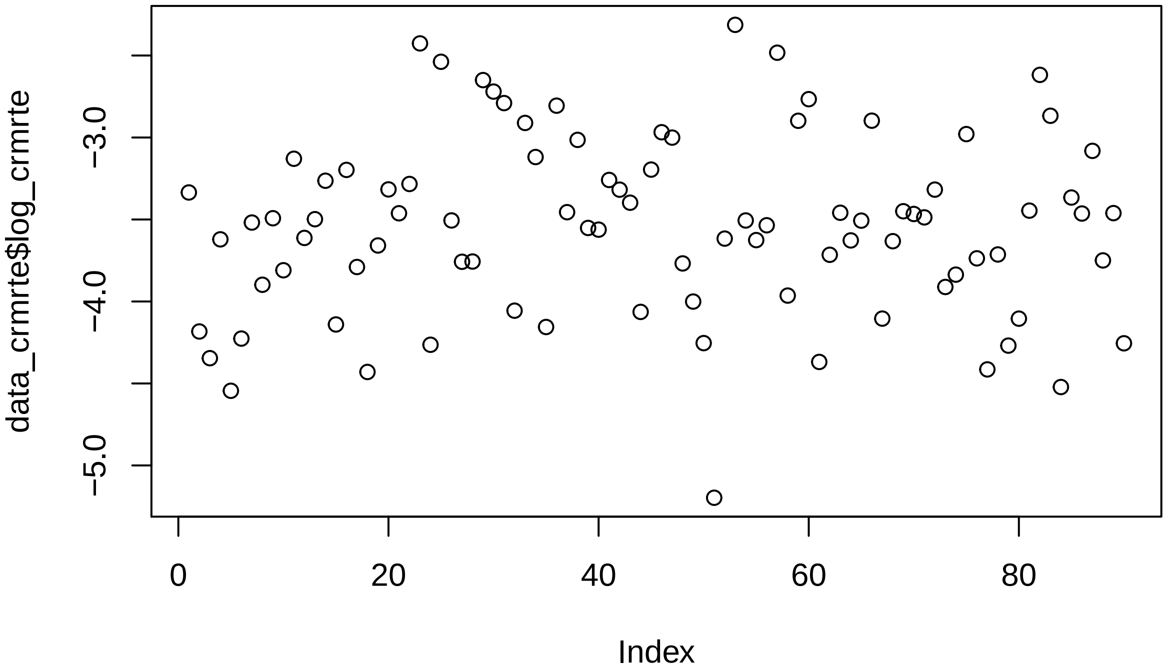
```
data_crmrte$log_crmrte <- log(data_crmrte$crmrte)
hist(data_crmrte$log_crmrte,breaks=20)
```



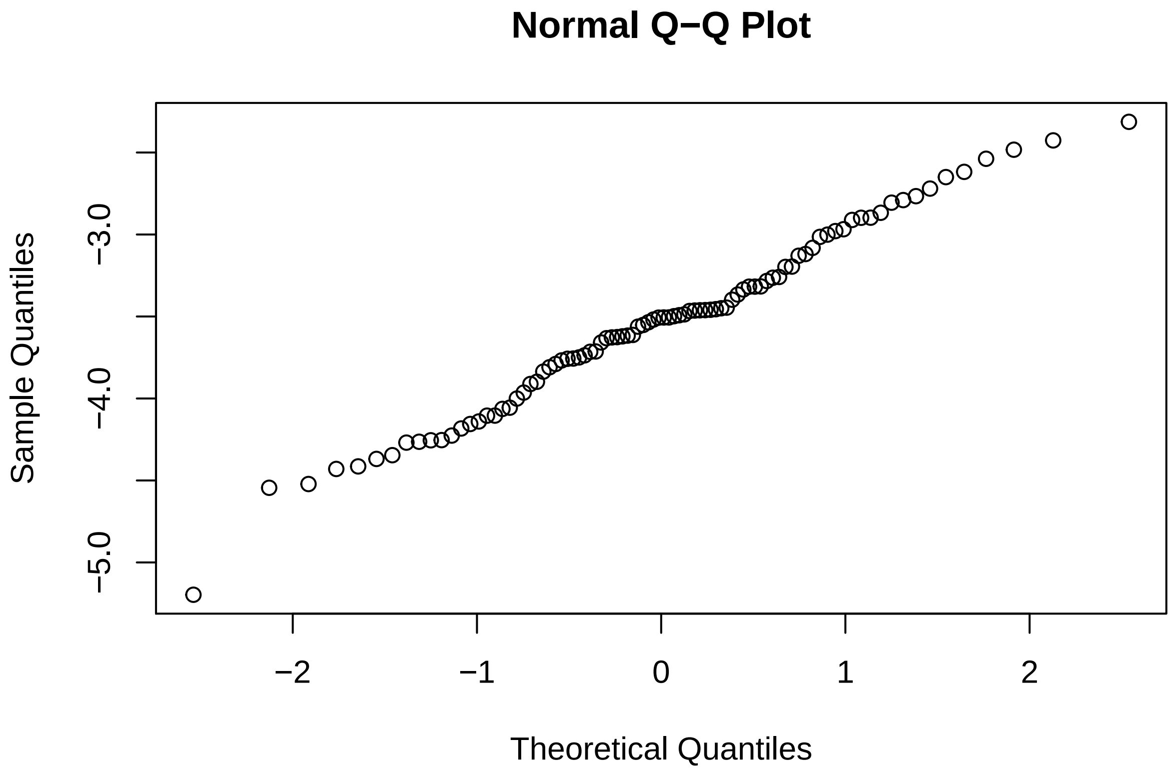
```
boxplot(data_crmrte$log_crmrte, main="Boxplot of log of crmrte")
```



```r
plot(data_crmrte$log_crmrte)
```



```r
qqnorm(data_crmrte$log_crmrte)
```



The histogram of the transformed crime rate is much more symmetrical and shows much less right skew. The box plot shows all of the outliers on the high end have been removed, though one outlier on the low end has been introduced. The scatter plot looks much more normal, and the Q-Q plot is much closer to normal with the data points hugging the 45 degree line much more closely. Given the stronger argument for the political campaign and the benefits to normality we have chosen to model the tranformation of crime rate as opposed to crime rate.

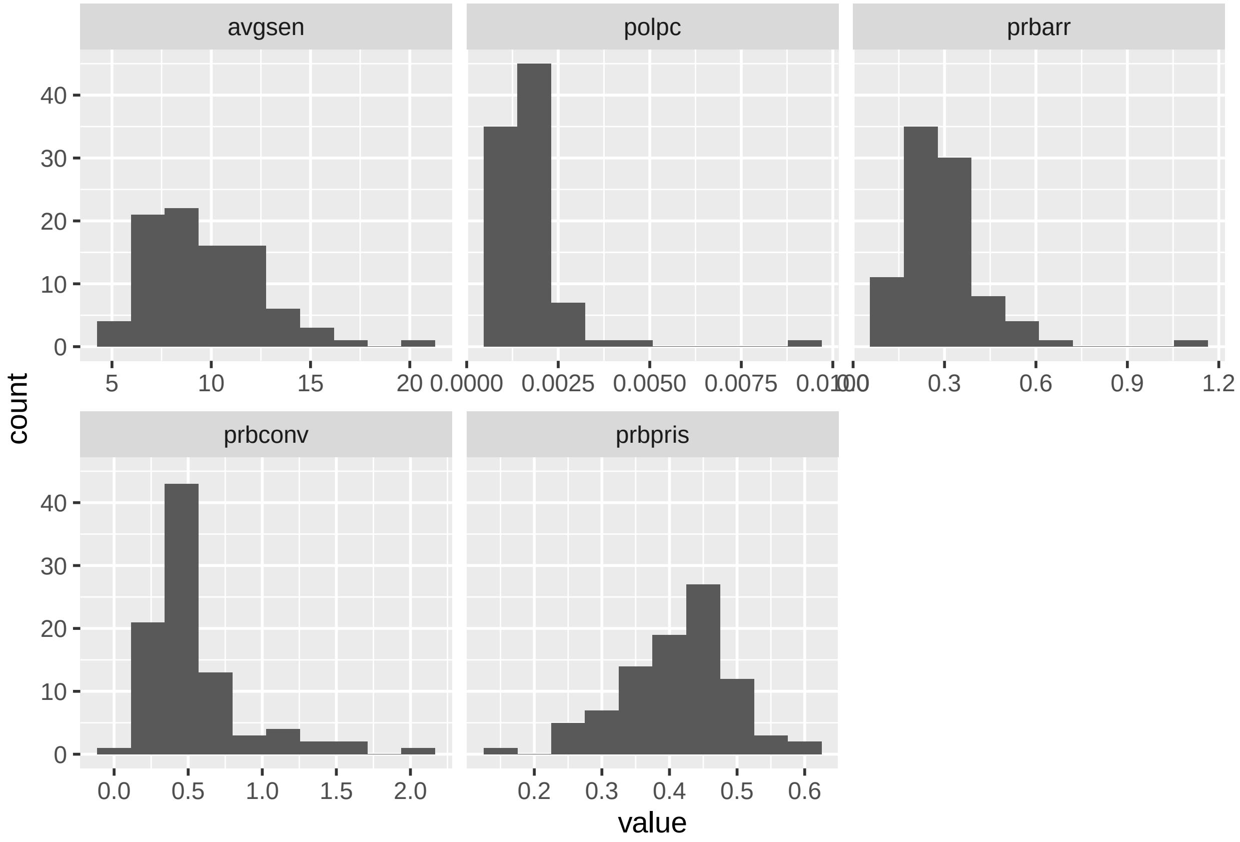
Groupings

In order to digest the data in the data set we decided to group the variables into five groups: deterrent, wages, demographic, region, and urban. We performed exploratory data analysis on all of these variables.

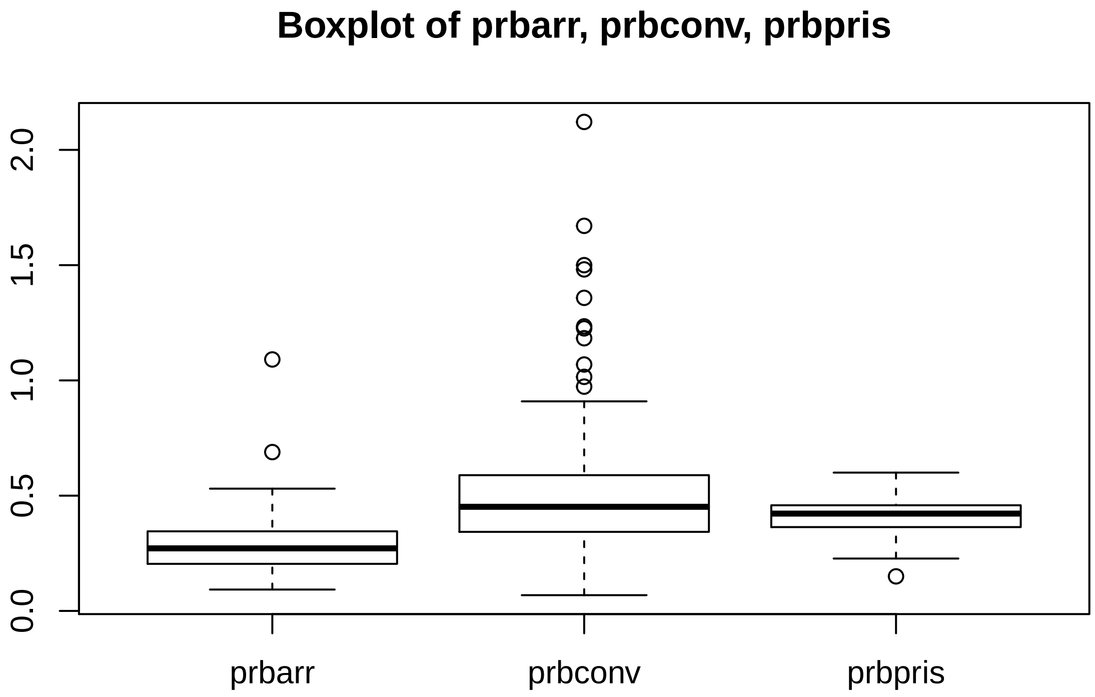
The group is deterrent data. As cited in the original paper, these variables were hypothesized to reduce crime rate through disincentivizing crime. Essentially, as the probability of getting caught increases, criminals' desire to commit crimes decreases.

Deterrent Data

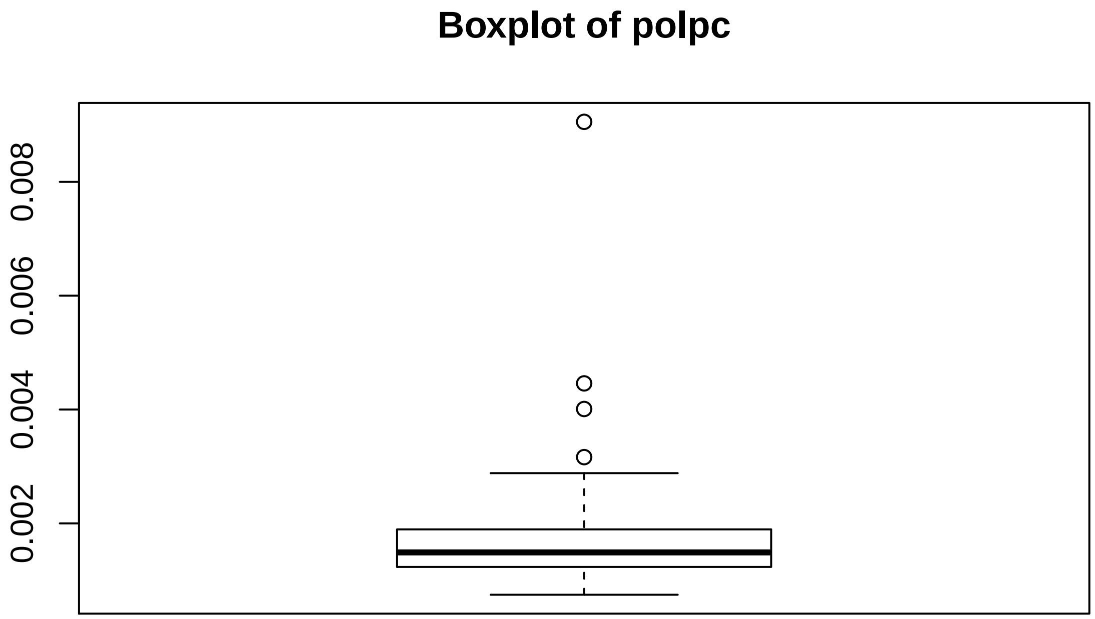
```
deterrent_data <- data_crmrte[,c('prbarr','prbconv','prbpris','crmrte',
                               'avgsen','polpc')]

ggplot(gather(deterrent_data[,c('prbarr','prbconv','prbpris',
                                           'avgsen','polpc')]), aes(value)) +
geom_histogram(bins = 10) + facet_wrap(~key, scales = 'free_x')
```



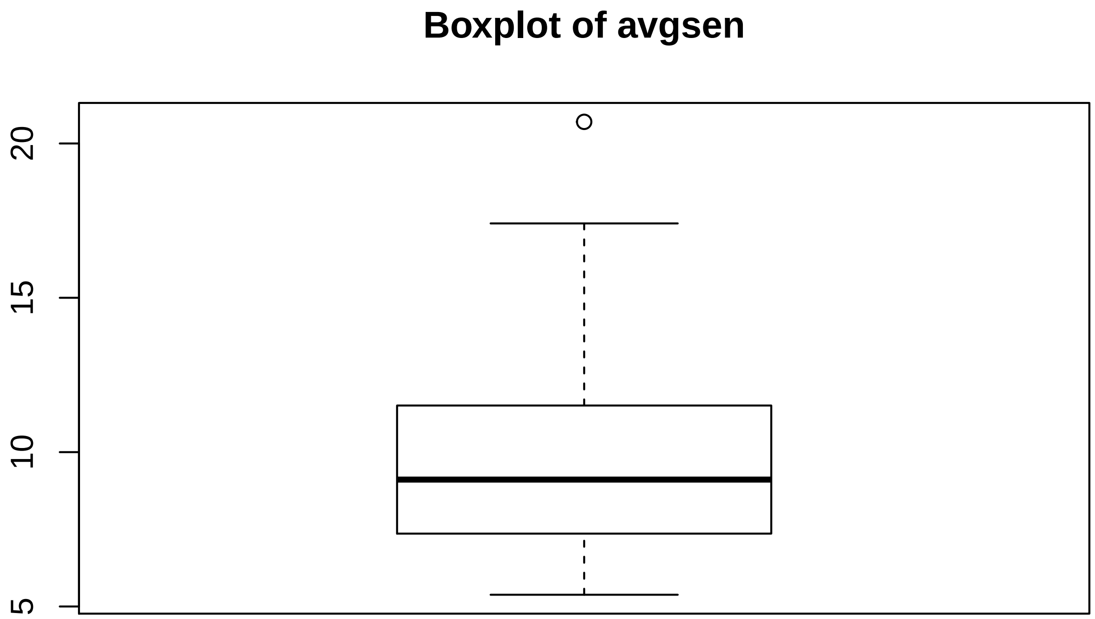
```
my_vars1 <- c("prbarr","prbconv","prbpris")
deterrent_data2 <- deterrent_data[my_vars1]
my_vars2 <- c("polpc")
deterrent_data3 <- deterrent_data[my_vars2]

boxplot(deterrent_data2, main="Boxplot of prbarr, prbconv, prbpris")
```



```
boxplot(deterrent_data3, main="Boxplot of polpc")
```



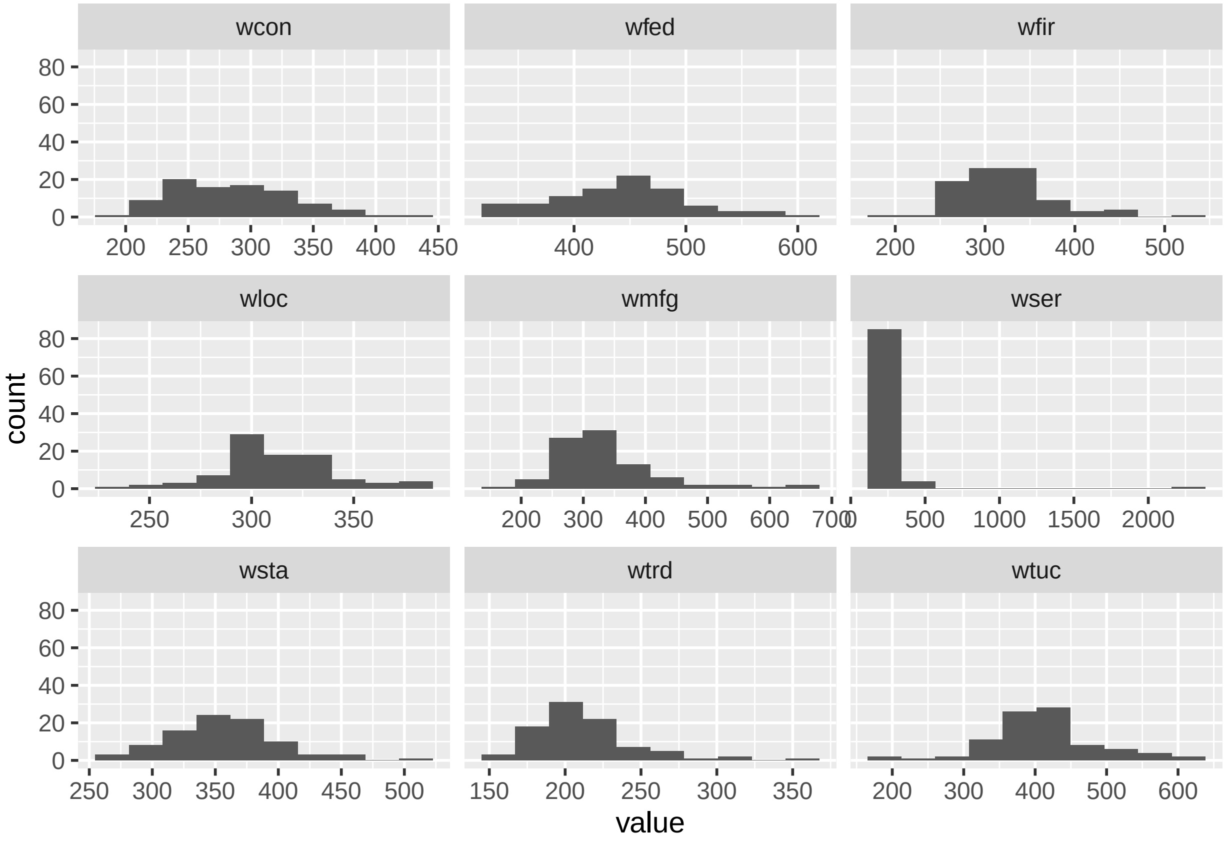
```r
boxplot(deterrent_data$avgsen, main="Boxplot of avgsen")
```



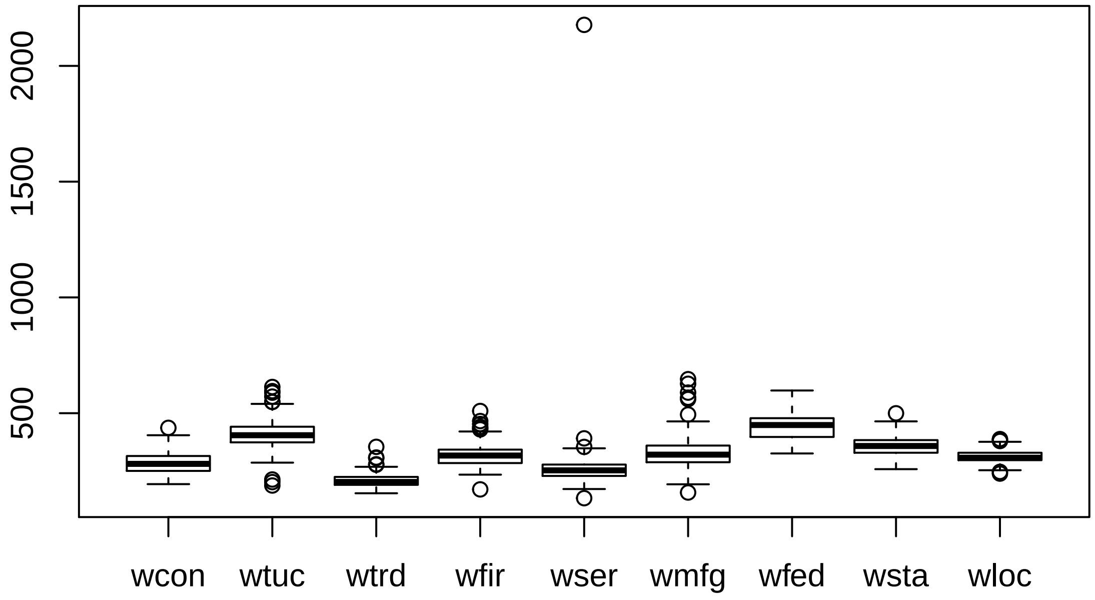
The first four histograms show right skew while prbpris shows left skew. The biggest outlier is observation 51. This observation has the lowest crime rate in the data set, obviously the highest polpc (police per capita), the highest avg sentence, the third highest prbconv, and the lowest pctmin80. This observation is likely to affect many of the regressions so it will need to be examined further. These variables are candidates to be transformed.

Wages Data

```
#create a dataframe of just the wage variables
wages_data <- data_crmrte[,c('wcon','wtuc','wtrd','wfir', 'wser',
                             'wmfg','wfed', 'wsta', 'wloc')]

#plot histograms of just the wage variables
ggplot(gather(wages_data), aes(value)) +
  geom_histogram(bins = 10) +
  facet_wrap(~key, scales = 'free_x')
```



```
#generate boxplots of just the wage variables
boxplot(wages_data)
```

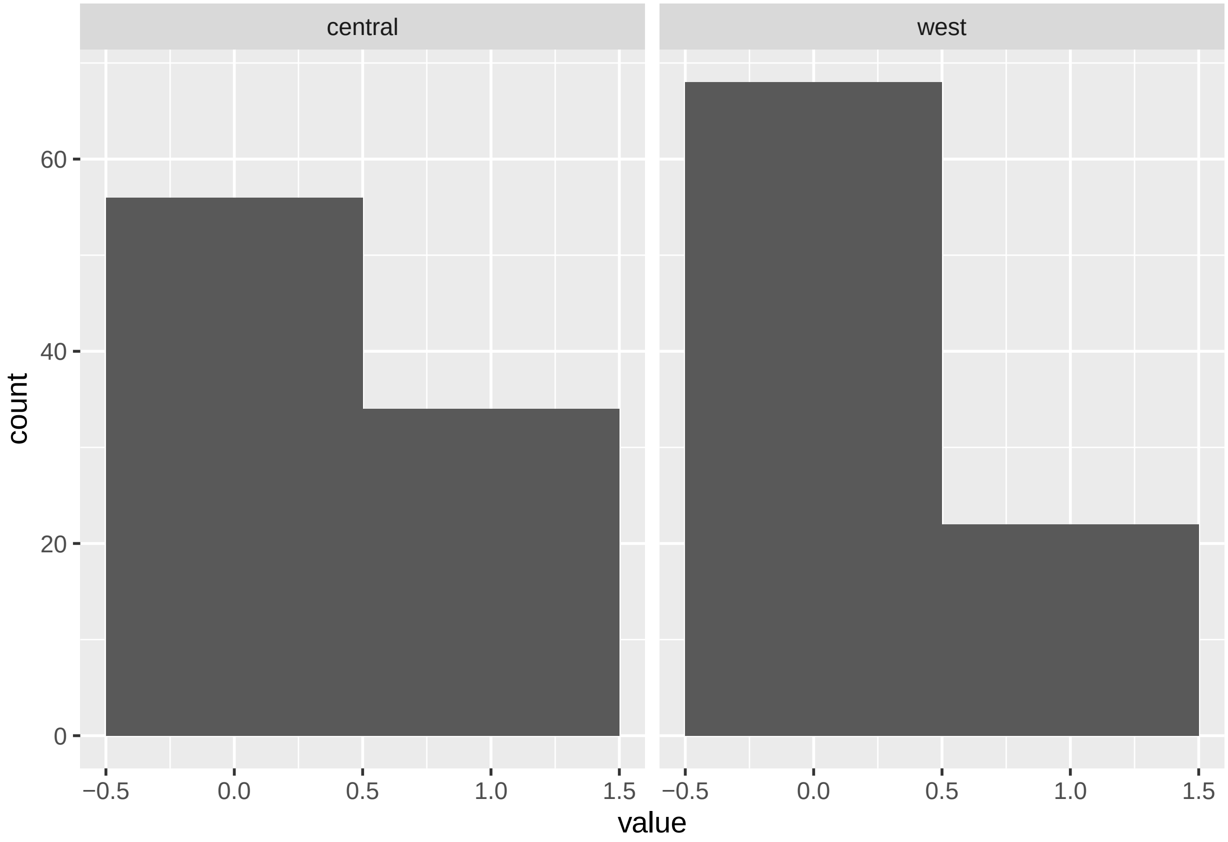


There is an obvious outlier in wser. This seems like a typo. The next highest average weekly wage in any sector is 646 versus the value of 2177.If you take the average of all weekly wages it has the highest average but only the 24th highest taxpc (though tax revenue is driven largely by sales and property taxes, many areas have local income tax as well. It is very possible that it is an error but we will revisit this later. For now, we create an additional variable that is the median of all wage variables for each observation. If it conveys as much information, it has the benefit of increasing our degress of freedom and removing the effect of the outlier.

```
data_crmrte$median_wage <- apply(data_crmrte[c("wcon", "wtuc", "wtrd",
                                               "wfir", "wser", "wmfg",
                                               "wfed", "wsta", "wloc")],
                                          1, FUN=median, na.rm=TRUE)
```

Region Data___

```
#create a dataframe of just the wage variables
dummies_data <- data_crmrte[,c('west','central')]

#plot histograms of just the dummy variables
ggplot(gather(dummies_data), aes(value)) +
  geom_histogram(bins = 2) +
  facet_wrap(~key)
```

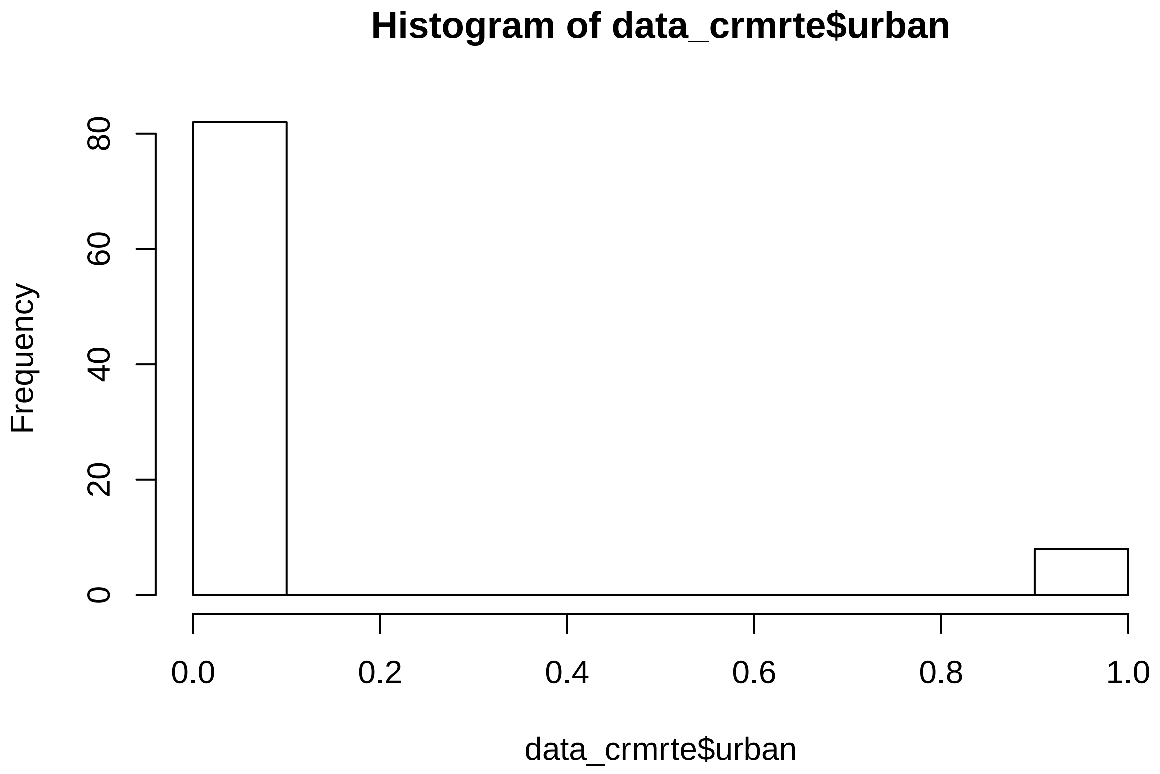


```
#just a quick check that there is no overlap
region_check <- data_crmrte[which(data_crmrte$west == 1 && data_crmrte$central == 1)]
```

The regions are broken up into central, west, and east. East is left out of the data set and it's effect as the final level of the indicator variable will move to the intercept.

Urban Data

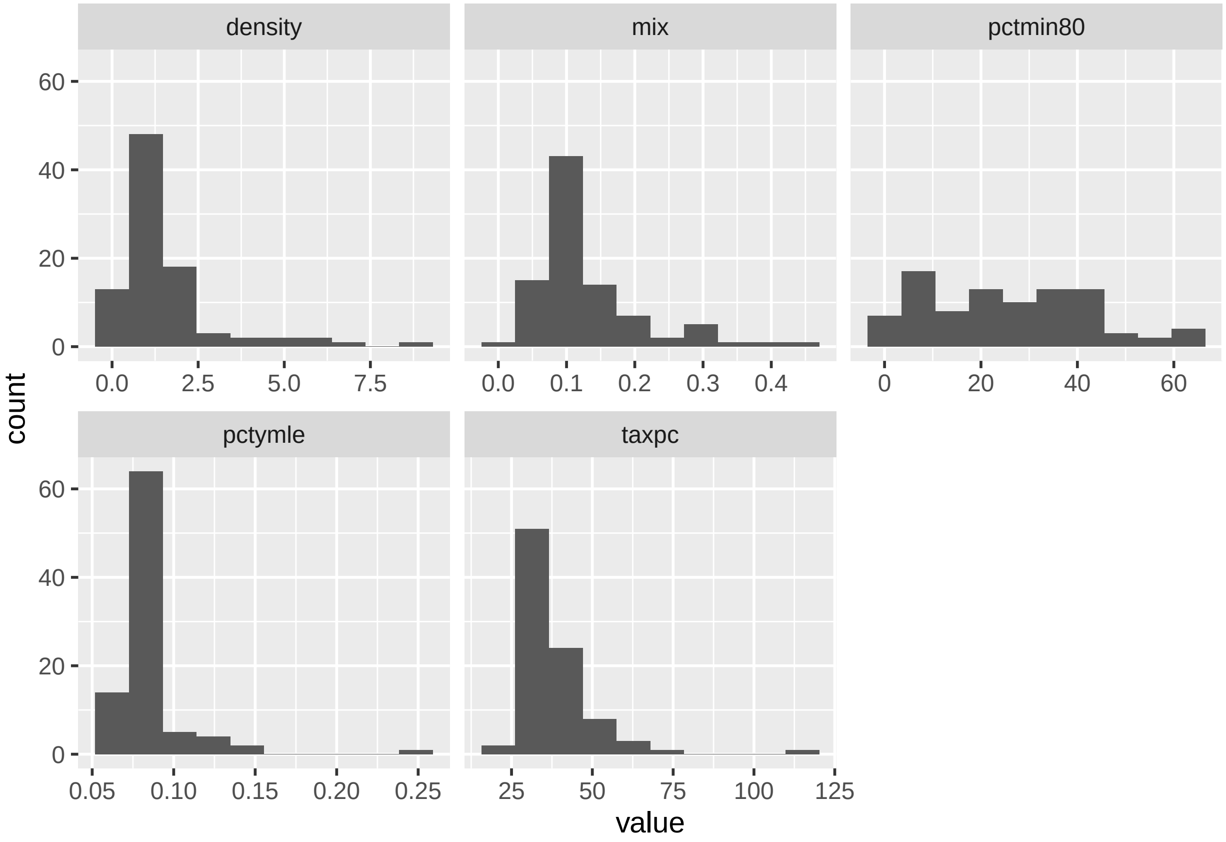
```
#plot histograms of just the wage variables
hist(data_crmrte$urban)
```



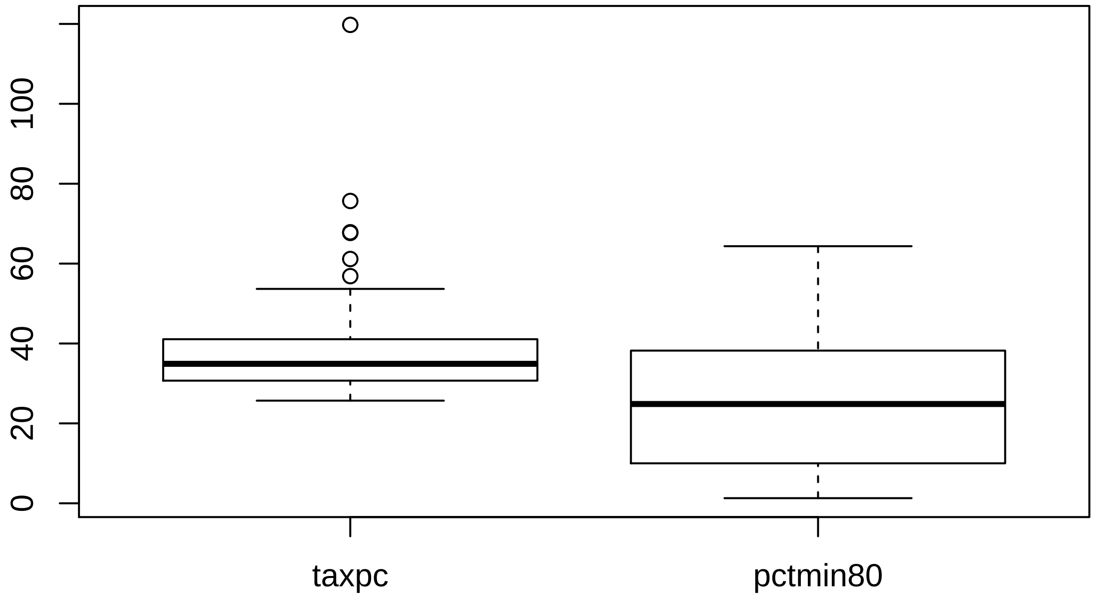
Urban did not fit into a great grouping so we left this variable on its own. A histogram shows that the state has relatively few urban counties, something to keep in mind when analyzing other variables such as density.

Demographic Data

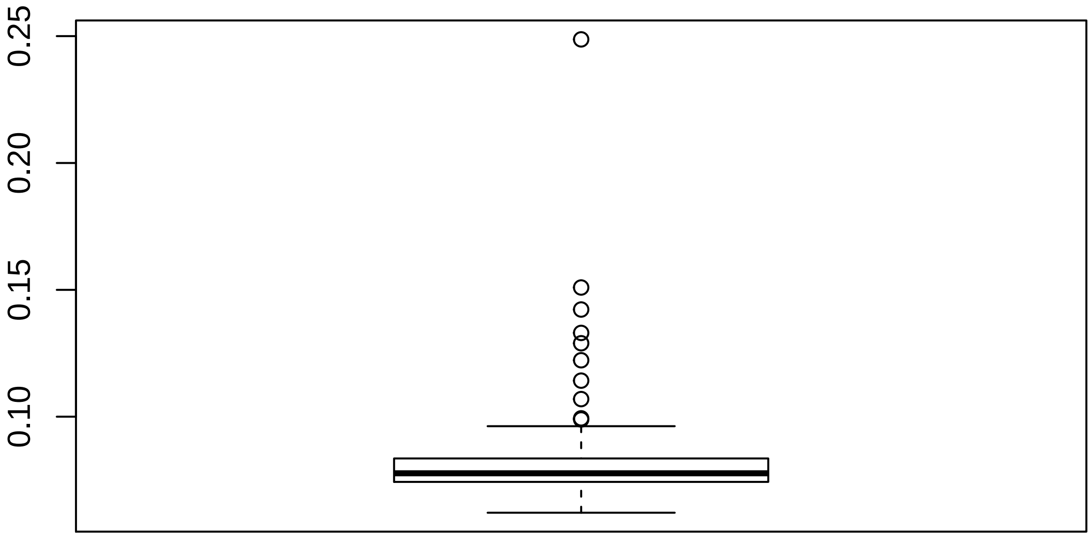
```r
#create a dataframe of just the demographic variables
demographic_data <- data_crmrte[,c('density', 'taxpc', 'pctmin80',
                                   'mix', 'pctymle')]

#plot histograms of just the demographic variables
ggplot(gather(demographic_data), aes(value)) +
  geom_histogram(bins = 10) +
  facet_wrap(~key, scales = 'free_x')
```



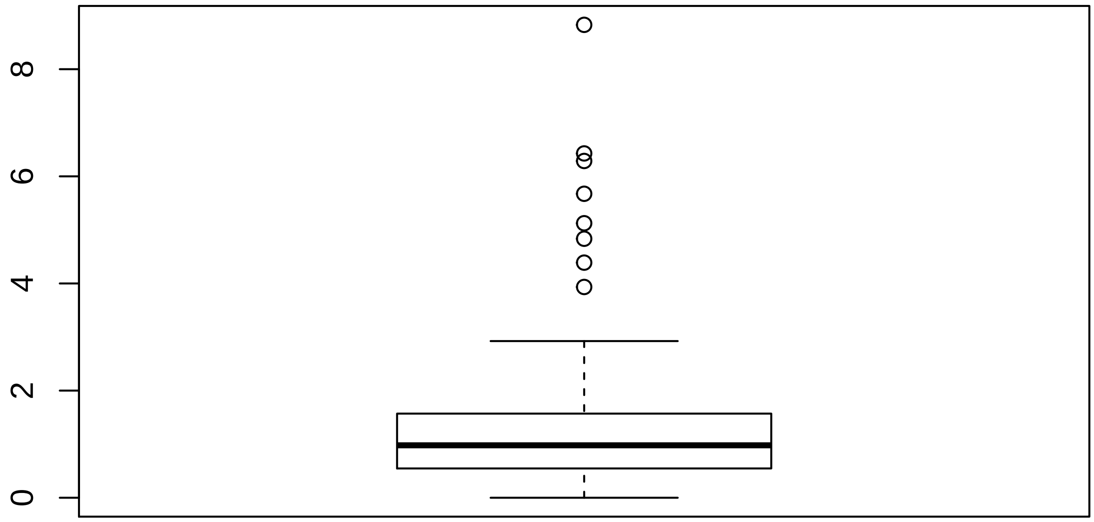
```
#Lots of skewed distributions above, particularly in pctymle and taxpc
#generate boxplots of just the demographic variables
demographic_data2 <- demographic_data[c("taxpc", "pctmin80")]
boxplot(demographic_data2)
```



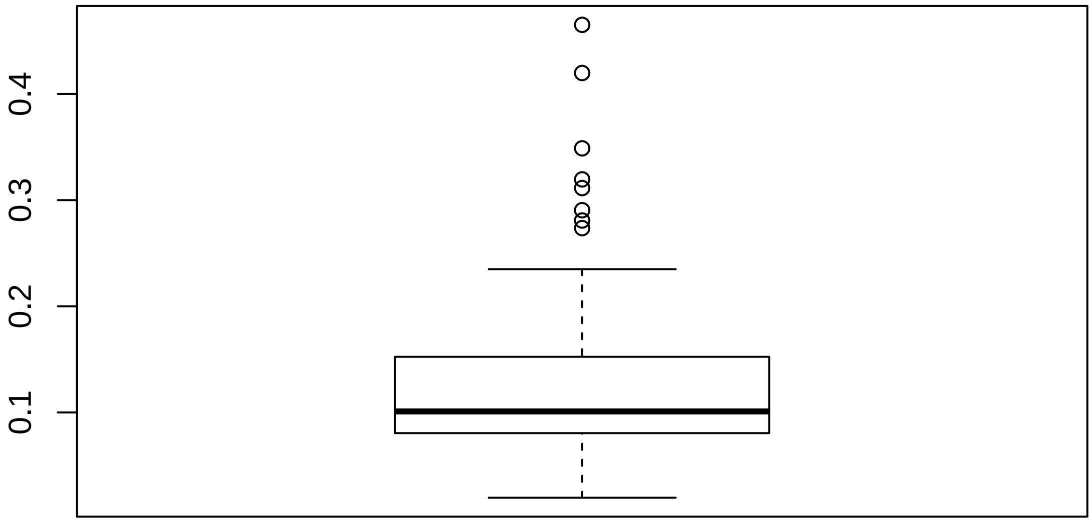
```
demographic_data3 <- demographic_data[c("pctymle")]
boxplot(demographic_data3)
```



```r
demographic_data4 <- demographic_data[c("density")]
boxplot(demographic_data4)
```



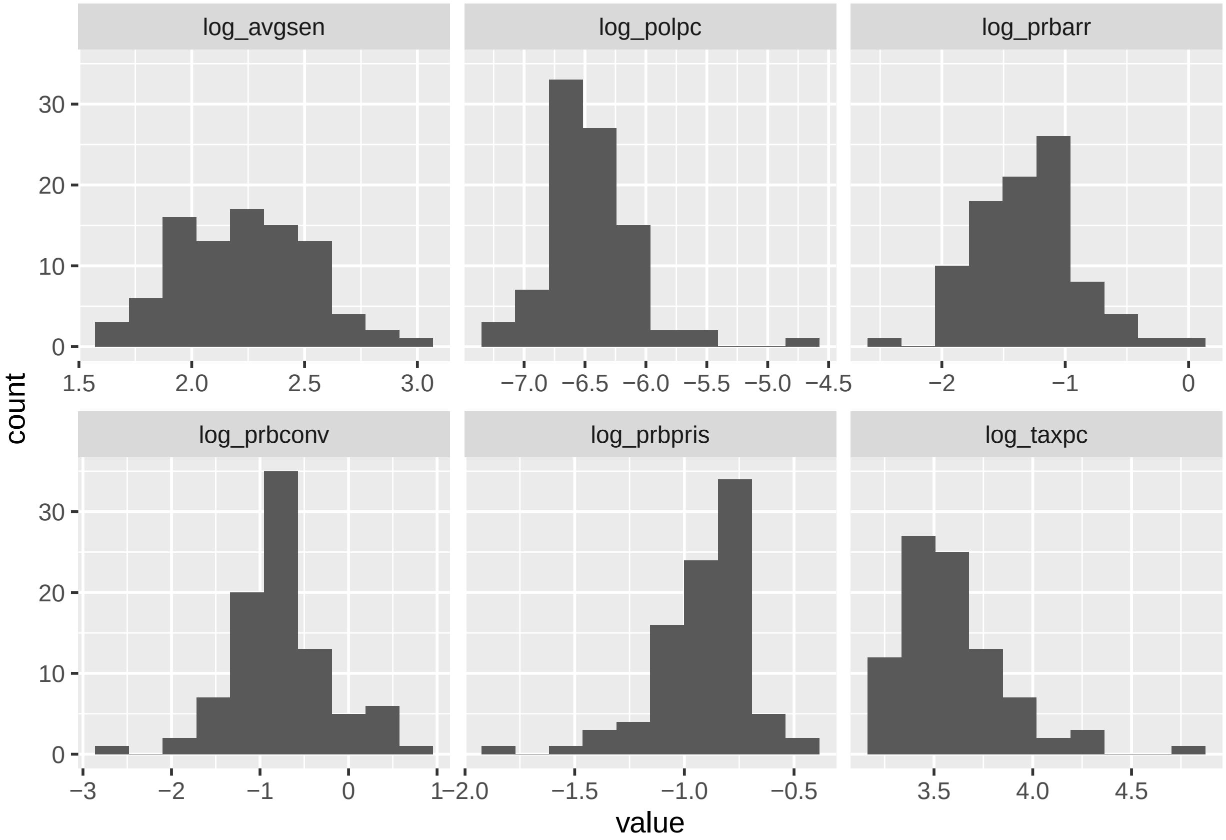
```
demographic_data5 <- demographic_data[c("mix")]
boxplot(demographic_data5)
```



Once again we see a lot of right skewed distributions in the historgams and in the box plots.

After exploring all of the variables we decided to tranform the other variables that are potentially under a politican's control - the deterrent variables. This gives us our final data set andso we can start running regressions.

```
data_crmrte$prbconv <- as.numeric(as.character(data_crmrte$prbconv))
data_crmrte$log_prbarr <- log(data_crmrte$prbarr)
data_crmrte$log_prbconv <- log(data_crmrte$prbconv)
data_crmrte$log_prbpris <- log(data_crmrte$prbpris)
data_crmrte$log_avgsen <- log(data_crmrte$avgsen)
data_crmrte$log_polpc <- log(data_crmrte$polpc)
data_crmrte$log_taxpc <- log(data_crmrte$taxpc)


#plot histograms of just the demographic variables
ggplot(gather(data_crmrte[,c('log_prbarr', 'log_prbconv', 'log_prbpris', 'log_avgsen', 'log_polpc', 'log
  geom_histogram(bins = 10) +
  facet_wrap(~key, scales = 'free_x')
```



Though the distribution of the variables still exhibits skew, the skew does seem to be reduced.

Log Tranformed Dependent Variable Comparison

In order to settle on the final data set we compare an all-in log-log model with an all-in log-linear to see which dependent variables are more suitable.

```
###### Initial Models #####
all_in_model <- lm(crmrte ~ prbarr + prbconv + prbpris
                   + avgsen + polpc + density
                   + taxpc + west + central + urban + pctmin80 + wcon
                   + wtuc + wtrd + wfir + wser + wmfg
                   + wfed + wsta + wloc
                   + mix + pctymle,
                   data = data_crmrte)
se.all_in_model = sqrt(diag(vcovHC(all_in_model)))
coeftest(all_in_model, vcov = vcovHC)
```

```
##
## t test of coefficients:
##
##                 Estimate  Std. Error t value  Pr(>|t|)
## (Intercept)  1.3853e-02  3.0755e-02  0.4504 0.6538622
## prbarr      -5.1466e-02  1.5689e-02 -3.2805 0.0016467 **
## prbconv     -1.8633e-02  6.5853e-03 -2.8295 0.0061464 **
## prbpris      3.1727e-03  1.3586e-02  0.2335 0.8160642
## avgsen      -3.9858e-04  5.5361e-04 -0.7200 0.4740570
## polpc        6.9679e+00  2.9536e+00  2.3591 0.0212406 *
## density      5.3314e-03  1.4895e-03  3.5793 0.0006464 ***
## taxpc        1.6240e-04  2.8408e-04  0.5717 0.5694537
## west        -2.5652e-03  4.4698e-03 -0.5739 0.5679579
## central     -4.2416e-03  3.7423e-03 -1.1334 0.2610725
## urban       -9.6498e-05  8.2752e-03 -0.0117 0.9907307
## pctmin80     3.2542e-04  1.3849e-04  2.3497 0.0217429 *
## wcon         2.3025e-05  3.2876e-05  0.7004 0.4861334
## wtuc         6.1914e-06  1.9862e-05  0.3117 0.7562178
## wtrd         2.8767e-05  8.7294e-05  0.3295 0.7427756
## wfir        -3.5455e-05  3.5699e-05 -0.9932 0.3242068
## wser        -1.7158e-06  9.9447e-05 -0.0173 0.9862856
## wmfg        -8.9675e-06  1.7469e-05 -0.5133 0.6094087
## wfed         2.9075e-05  3.7780e-05  0.7696 0.4442480
## wsta        -2.2302e-05  3.6828e-05 -0.6056 0.5468431
## wloc         1.4456e-05  8.5367e-05  0.1693 0.8660410
## mix         -1.8693e-02  2.2922e-02 -0.8155 0.4176761
## pctymle      1.0125e-01  4.7826e-02  2.1170 0.0379748 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
AIC(all_in_model, k=2)
```

```
## [1] -585.5858
```

```r
all_in_model_log_level <- lm(log_crmrte ~ prbarr + prbconv + prbpris
                             + avgsen + polpc + density
                             + taxpc + west + central + urban
                             + pctmin80 + wcon
                             + wtuc + wtrd + wfir + wser + wmfg
                             + wfed + wsta + wloc
                             + mix + pctymle,
                             data = data_crmrte)
se.all_in_model_log_level = sqrt(diag(vcovHC(all_in_model_log_level)))
coeftest(all_in_model_log_level, vcov = vcovHC)
```

```
##
## t test of coefficients:
##
##                Estimate  Std. Error t value  Pr(>|t|)
## (Intercept) -4.0261e+00  8.4822e-01 -4.7466 1.128e-05 ***
## prbarr      -1.8891e+00  3.7955e-01 -4.9773 4.770e-06 ***
## prbconv     -6.5603e-01  1.7443e-01 -3.7611 0.0003579 ***
## prbpris     -9.3077e-02  3.9921e-01 -0.2332 0.8163542
## avgsen      -7.8769e-03  1.6125e-02 -0.4885 0.6267962
## polpc        1.5484e+02  8.6523e+01  1.7895 0.0780510 .
## density      1.1653e-01  5.4037e-02  2.1566 0.0346326 *
## taxpc        3.3224e-03  7.2890e-03  0.4558 0.6500012
## west        -1.1492e-01  1.2509e-01 -0.9187 0.3615403
## central     -1.0078e-01  9.2053e-02 -1.0948 0.2775232
## urban       -1.6923e-01  2.2872e-01 -0.7399 0.4619535
## pctmin80     9.9770e-03  3.0480e-03  3.2733 0.0016833 **
## wcon         4.6001e-04  8.3564e-04  0.5505 0.5838140
## wtuc         1.0174e-04  6.0187e-04  0.1690 0.8662750
## wtrd         2.5964e-04  1.7638e-03  0.1472 0.8834136
## wfir        -1.1015e-03  1.1960e-03 -0.9210 0.3603557
## wser        -1.3142e-04  1.5060e-03 -0.0873 0.9307193
## wmfg        -2.0528e-04  5.1630e-04 -0.3976 0.6921878
## wfed         2.3405e-03  1.0820e-03  2.1632 0.0340968 *
## wsta        -1.1357e-03  8.9769e-04 -1.2651 0.2102213
## wloc         5.8983e-04  2.4003e-03  0.2457 0.8066400
## mix         -2.3924e-01  6.2632e-01 -0.3820 0.7036869
## pctymle      2.7706e+00  1.4330e+00  1.9334 0.0574191 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
AIC(all_in_model_log_level, k=2)
```

```
## [1] 21.354
```

```
all_in_model_log_log <- lm(log_crmrte ~ log_prbarr + log_prbconv
                            + log_prbpris + log_avgsen + log_polpc
                            + density+ log_taxpc + west + central
                            + urban + pctmin80 + wcon
                            + wtuc + wtrd + wfir
                            + wser + wmfg + wfed + wsta + wloc
                            + mix + pctymle,
                            data = data_crmrte)
se.all_in_model_log_log = sqrt(diag(vcovHC(all_in_model_log_log)))
coeftest(all_in_model_log_log, vcov = vcovHC)
```

```
##
## t test of coefficients:
##
##                Estimate  Std. Error t value Pr(>|t|)
## (Intercept) -3.36882669  2.97497990 -1.1324 0.261508
## log_prbarr  -0.52143620  0.16459898 -3.1679 0.002313 **
## log_prbconv -0.33101341  0.15365522 -2.1543 0.034820 *
## log_prbpris -0.06569465  0.19741379 -0.3328 0.740342
## log_avgsen  -0.19652151  0.18205821 -1.0794 0.284261
## log_polpc    0.29132794  0.27176129  1.0720 0.287567
## density      0.12320127  0.06040422  2.0396 0.045335 *
## log_taxpc    0.06158051  0.30979897  0.1988 0.843040
## west        -0.18453792  0.16353910 -1.1284 0.263174
## central     -0.10789292  0.09991865 -1.0798 0.284100
## urban       -0.14767055  0.26670745 -0.5537 0.581641
## pctmin80     0.00956927  0.00358175  2.6717 0.009466 **
## wcon         0.00078953  0.00090745  0.8701 0.387376
## wtuc         0.00010106  0.00075559  0.1337 0.894001
## wtrd         0.00029022  0.00177967  0.1631 0.870952
## wfir        -0.00108230  0.00125937 -0.8594 0.393186
## wser        -0.00042887  0.00096365 -0.4451 0.657718
## wmfg        -0.00014147  0.00061356 -0.2306 0.818343
## wfed         0.00224918  0.00136611  1.6464 0.104363
## wsta        -0.00102039  0.00106131 -0.9614 0.339787
## wloc         0.00017815  0.00261968  0.0680 0.945986
## mix         -0.44834658  0.77846459 -0.5759 0.566587
## pctymle      2.00755501  2.60186976  0.7716 0.443075
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
AIC(all_in_model_log_log, k=2)
```

```
## [1] 44.17803
```

```
#Not comparing r-squared, just looking at significant variables
stargazer(all_in_model, all_in_model_log_level,
          all_in_model_log_log,
          type = "text", omit.stat = "f",
          se = list(se.all_in_model, se.all_in_model_log_level,
                    se.all_in_model_log_log),
          star.cutoffs = c(0.05, 0.01, 0.001))
```

```
##
## =============================================================
##                                     Dependent variable:
##                               -------------------------------
##                                 crmrte        log_crmrte
##                                  (1)        (2)        (3)
## -------------------------------------------------------------
## prbarr                         -0.051**  -1.889***
##                                (0.016)    (0.380)
##
## prbconv                        -0.019**  -0.656***
##                                (0.007)    (0.174)
##
## prbpris                         0.003      -0.093
##                                (0.014)    (0.399)
##
## avgsen                         -0.0004     -0.008
##                                (0.001)    (0.016)
##
## polpc                           6.968*    154.835
##                                (2.954)    (86.523)
##
## log_prbarr                                          -0.521**
##                                                      (0.165)
##
## log_prbconv                                         -0.331*
##                                                      (0.154)
##
## log_prbpris                                         -0.066
##                                                      (0.197)
##
## log_avgsen                                          -0.197
##                                                      (0.182)
##
```

```
## log_polpc                                            0.291
##                                                     (0.272)
##
## density                        0.005***    0.117*    0.123*
##                                (0.001)    (0.054)    (0.060)
##
## taxpc                           0.0002     0.003
##                                (0.0003)   (0.007)
##
## log_taxpc                                            0.062
##                                                     (0.310)
##
## west                            -0.003     -0.115    -0.185
##                                (0.004)    (0.125)    (0.164)
##
## central                         -0.004     -0.101    -0.108
##                                (0.004)    (0.092)    (0.100)
##
## urban                          -0.0001     -0.169    -0.148
##                                (0.008)    (0.229)    (0.267)
##
## pctmin80                       0.0003*    0.010**    0.010**
##                                (0.0001)   (0.003)    (0.004)
##
## wcon                           0.00002     0.0005     0.001
##                               (0.00003)   (0.001)    (0.001)
##
## wtuc                           0.00001     0.0001    0.0001
##                               (0.00002)   (0.001)    (0.001)
##
## wtrd                           0.00003     0.0003    0.0003
##                                (0.0001)   (0.002)    (0.002)
##
## wfir                           -0.00004    -0.001    -0.001
##                               (0.00004)   (0.001)    (0.001)
##
## wser                           -0.00000   -0.0001    -0.0004
##                                (0.0001)   (0.002)    (0.001)
##
## wmfg                           -0.00001   -0.0002    -0.0001
##                               (0.00002)   (0.001)    (0.001)
##
## wfed                           0.00003     0.002*     0.002
##                               (0.00004)   (0.001)    (0.001)
##
## wsta                           -0.00002    -0.001    -0.001
##                               (0.00004)   (0.001)    (0.001)
##
## wloc                           0.00001     0.001     0.0002
##                                (0.0001)   (0.002)    (0.003)
##
## mix                             -0.019     -0.239    -0.448
##                                (0.023)    (0.626)    (0.778)
##
```

```
## pctymle                         0.101*     2.771      2.008
##                                (0.048)    (1.433)    (2.602)
##
## Constant                        0.014    -4.026***   -3.369
##                                (0.031)    (0.848)    (2.975)
##
## -------------------------------------------------------------
## Observations                      90         90         90
## R2                              0.855      0.854      0.812
## Adjusted R2                     0.807      0.806      0.750
## Residual Std. Error (df = 67)   0.008      0.242      0.275
## =============================================================
## Note:                           *p<0.05; **p<0.01; ***p<0.001
```

```
# #r-squared comparison of final two models
# yhat_level_level <- predict(all_in_model)
#
# #get the coefficients
# for (b in coef(all_in_model_log_level))
# {
#   beta_log_level <- c(beta_log_level, b)
# }
# #calculate the predictions
# for (b in coef(all_in_model_log_level))
# {
#   beta_log_level <- c(beta_log_level, b)
# }
#
# data_crmrte$log_level_yhat <- exp(--3.36882669
#                                    -0.5214362*data_crmrte$log_prbarr
#                                    -0.33101341*data_crmrte$log_prbconv
#                                    -0.06569465*data_crmrte$log_prbpris
#                                    -0.19652151*data_crmrte$log_avgsen
#                                    +0.29132794*data_crmrte$log_polpc
#                                    +0.12320127*data_crmrte$density
#                                    +0.06158051*data_crmrte$log_taxpc
#                                    -0.18453792*data_crmrte$west
#                                    -0.10789292*data_crmrte$central
#                                    -0.14767055*data_crmrte$urban
#                                    +0.00956927*data_crmrte$pctmin80
#                                    +0.00078953*data_crmrte$wcon
#                                    +0.00010106*data_crmrte$wtuc
#                                    +0.00029022*data_crmrte$wtrd
#                                    -0.0010823*data_crmrte$wfir
#                                    -0.00042887*data_crmrte$wser
#                                    -0.00014147*data_crmrte$wmfg
#                                    +0.00224918*data_crmrte$wfed
#                                    -0.00102039*data_crmrte$wsta
#                                    +0.00017815*data_crmrte$wloc
#                                    -0.44834658*data_crmrte$mix
#                                    +2.00755501*data_crmrte$pctymle
#                                    )
# r_squared_level_level <- cor(data_crmrte$crmrte, yhat_level_level)
# r_squared_log_level <- cor(data_crmrte$crmrte, data_crmrte$log_level_yhat)
# (r_squared_level_level)
# (r_squared_log_level)
```

Model 1: Simple Model

In order to create a simple model we decided to build using a bottom up approach. We looked at a correlation matrix

```
#Anyone know how to print this better?
cor(data_crmrte$log_crmrte,data_crmrte)
```

```
## Warning in cor(data_crmrte$log_crmrte, data_crmrte): the standard deviation is
## zero
```

```
##          county year    crmrte     prbarr    prbconv    prbpris      avgsen
## [1,] 0.02376789   NA 0.9415465 -0.4727669 -0.4468136 0.02147024 -0.04936931
##          polpc   density     taxpc       west   central     urban  pctmin80
## [1,] 0.0104058 0.6330234 0.3583234 -0.4143996 0.1847192 0.4914645 0.2329182
##           wcon      wtuc      wtrd      wfir      wser      wmfg      wfed
## [1,] 0.3937149 0.2014649 0.3937924 0.2932426 -0.113128 0.3075373 0.5233058
##           wsta      wloc        mix   pctymle log_crmrte median_wage log_prbarr
## [1,] 0.1697021 0.2885668 -0.1247344 0.2781547          1    0.454422 -0.4357539
##      log_prbconv log_prbpris log_avgsen log_polpc log_taxpc
## [1,]  -0.3724961  0.06960729 0.02341717 0.2845396 0.3398432
```

/ In the above correlation matrix, focusing on the correlations between the log_crmrte and all other variables, denisty has the highest correlation. This variable makes intuitive sense. As a single variable it might encompass a lot of other factors. Lower income people with more incentive to commit crimes tend to live in more highly populated areas. Below is the simple regression.

```
simple_regression_model <- lm(log_crmrte ~ density, data = data_crmrte)
se.simple_regression_model = sqrt(diag(vcovHC(simple_regression_model)))
coeftest(simple_regression_model, vcov = vcovHC)
```

```
##
## t test of coefficients:
##
##              Estimate Std. Error  t value Pr(>|t|)
## (Intercept) -3.869488   0.068563 -56.4366  < 2e-16 ***
## density      0.228298   0.030439   7.5003  4.8e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

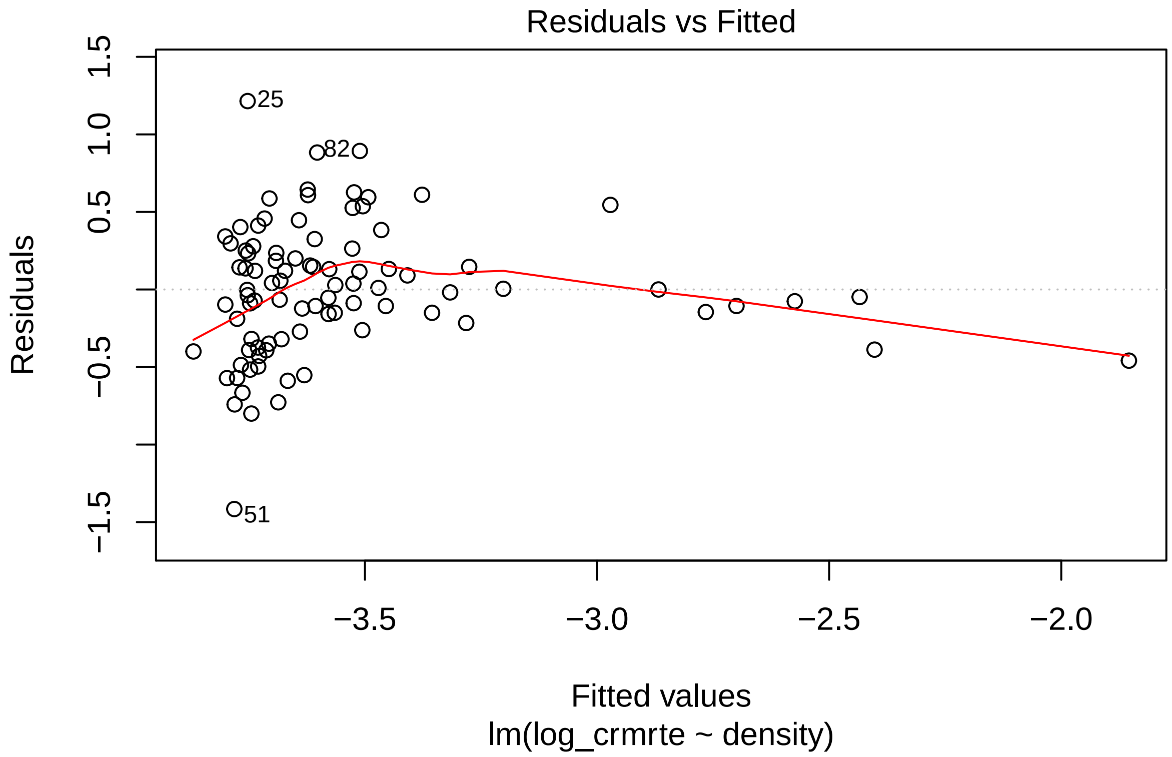
```
AIC(simple_regression_model, k=2)
```

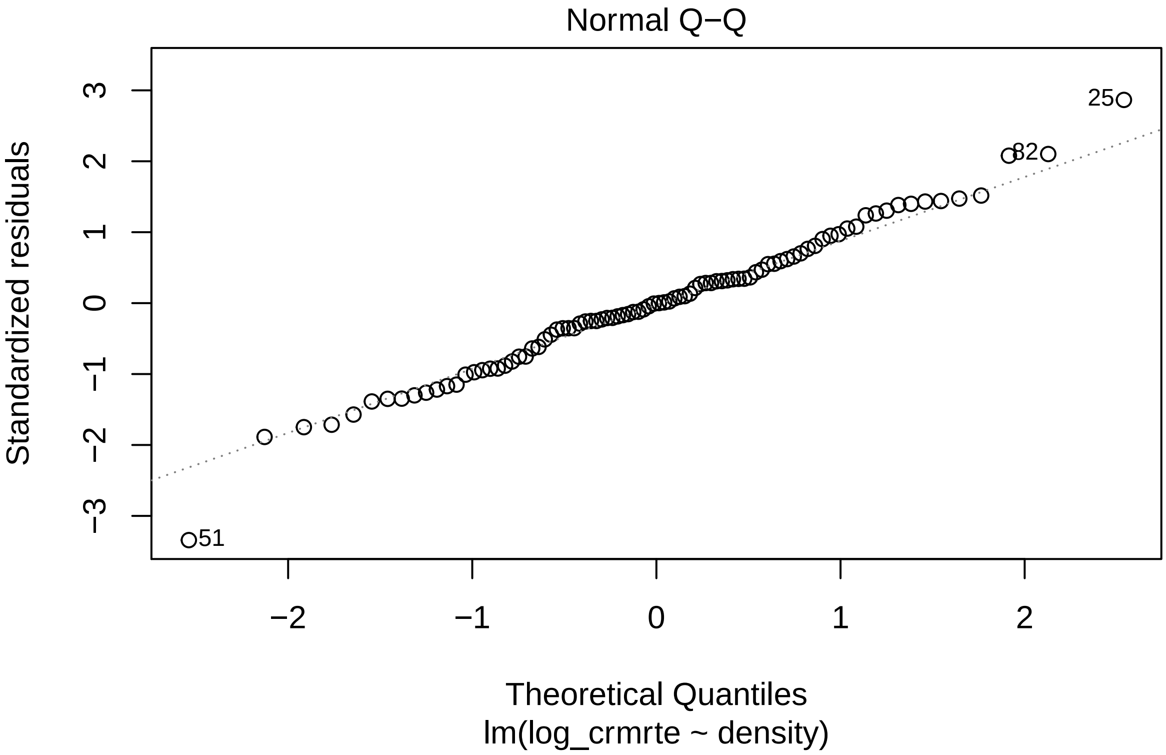
```
## [1] 106.2991
```

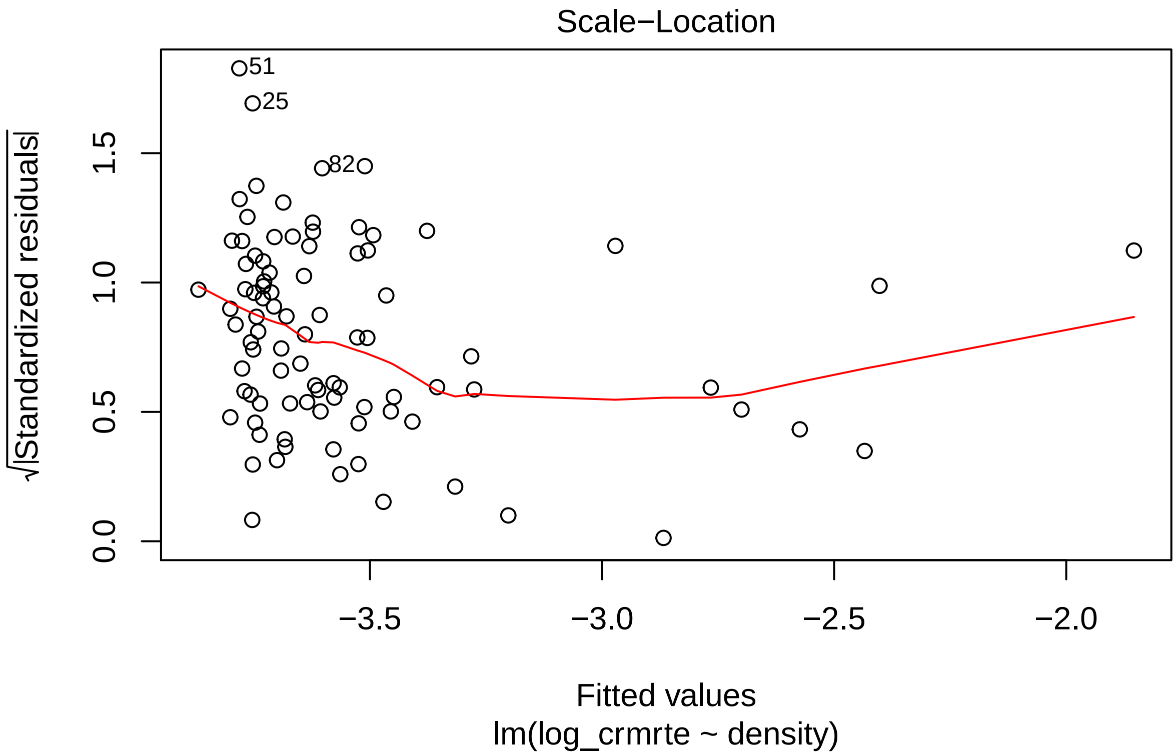
```
stargazer(simple_regression_model,
          type = "text", omit.stat = "f",
          se = list(se.simple_regression_model),
          star.cutoffs = c(0.05, 0.01, 0.001))
```

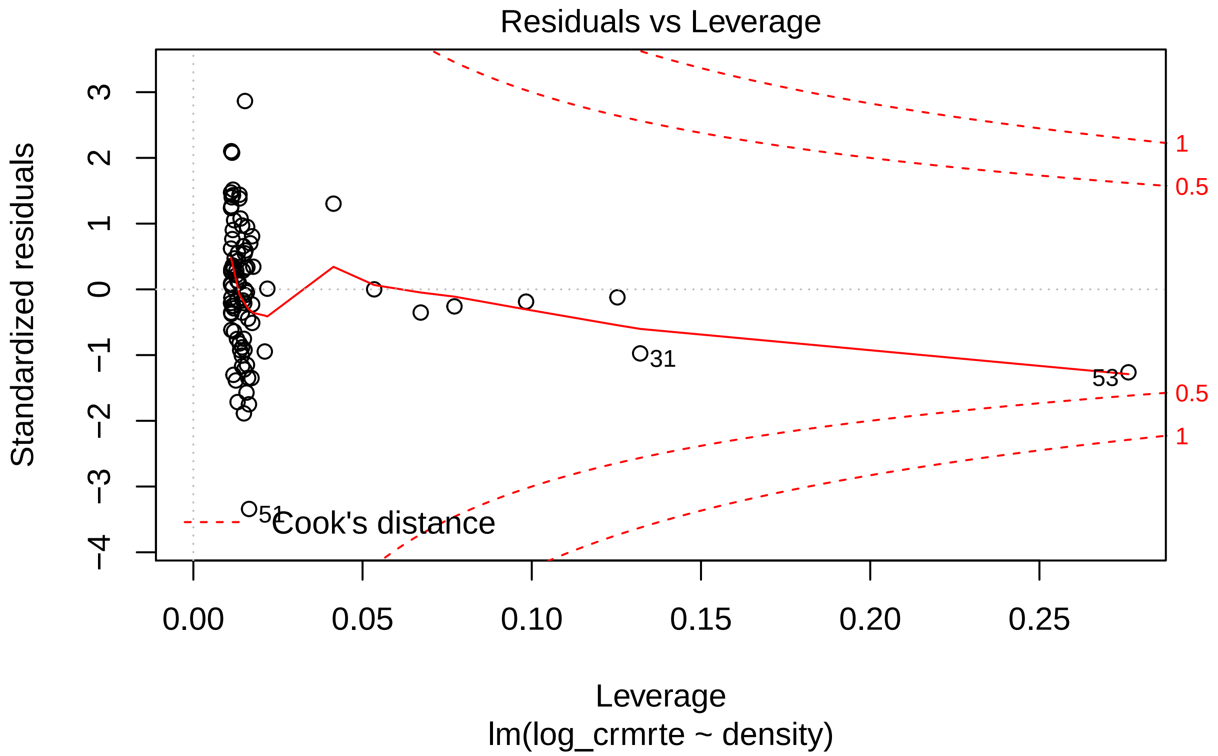
```
##
## ===================================================
##                            Dependent variable:
##                        ----------------------------
##                                 log_crmrte
## ---------------------------------------------------
## density                         0.228***
##                                  (0.030)
##
## Constant                        -3.869***
##                                  (0.069)
##
## ---------------------------------------------------
## Observations                        90
## R2                                0.401
## Adjusted R2                       0.394
## Residual Std. Error           0.427 (df = 88)
## ===================================================
## Note:                  *p<0.05; **p<0.01; ***p<0.001
```

```r
plot(simple_regression_model)
```




Normal Q−Q
Standardized residuals
3
2
1
0
−1
−2
−3
25
82
51
−2
−1
0
1
2
Theoretical Quantiles
lm(log_crmrte ~ density)


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
51
25
82
−3.5
−3.0
−2.5
−2.0
Fitted values
lm(log_crmrte ~ density)



The variable density explains 40.1% of the variation in the log of crime rate. As density increases by 1 unit (as the county population divided by the county land area increases by 1%) crime increases by 22%. The residuals vs. fitted plot indicates that the zero conditional mean assumpotion is violated. The Q-Q plot shows that the residuals are normally distributed, and the residuals vs leverage plot shows that there are no influential outliers.

Model 3: Kitchen Sink Model

Still, we can do better in predicting the log crime rate than simply using one variable. We know examine a "kitchen sink" model. This model includes all of the variables in the data set except county (which has too many values to be a useful indicator variable) and year, which is a constant (1987). Below are the results.

```
all_in_model_log_level <- lm(log_crmrte ~ prbarr + prbconv + prbpris
                             + avgsen + polpc + density
                             + taxpc + west + central + urban
                             + pctmin80 + wcon
                             + wtuc + wtrd + wfir + wser + wmfg
                             + wfed + wsta + wloc
                             + mix + pctymle,
                             data = data_crmrte)
se.all_in_model_log_level = sqrt(diag(vcovHC(all_in_model_log_level)))
coeftest(all_in_model_log_level, vcov = vcovHC)
```

```
##
## t test of coefficients:
##
```

```
##                Estimate  Std. Error t value  Pr(>|t|)
## (Intercept) -4.0261e+00  8.4822e-01 -4.7466 1.128e-05 ***
## prbarr      -1.8891e+00  3.7955e-01 -4.9773 4.770e-06 ***
## prbconv     -6.5603e-01  1.7443e-01 -3.7611 0.0003579 ***
## prbpris     -9.3077e-02  3.9921e-01 -0.2332 0.8163542
## avgsen      -7.8769e-03  1.6125e-02 -0.4885 0.6267962
## polpc        1.5484e+02  8.6523e+01  1.7895 0.0780510 .
## density      1.1653e-01  5.4037e-02  2.1566 0.0346326 *
## taxpc        3.3224e-03  7.2890e-03  0.4558 0.6500012
## west        -1.1492e-01  1.2509e-01 -0.9187 0.3615403
## central     -1.0078e-01  9.2053e-02 -1.0948 0.2775232
## urban       -1.6923e-01  2.2872e-01 -0.7399 0.4619535
## pctmin80     9.9770e-03  3.0480e-03  3.2733 0.0016833 **
## wcon         4.6001e-04  8.3564e-04  0.5505 0.5838140
## wtuc         1.0174e-04  6.0187e-04  0.1690 0.8662750
## wtrd         2.5964e-04  1.7638e-03  0.1472 0.8834136
## wfir        -1.1015e-03  1.1960e-03 -0.9210 0.3603557
## wser        -1.3142e-04  1.5060e-03 -0.0873 0.9307193
## wmfg        -2.0528e-04  5.1630e-04 -0.3976 0.6921878
## wfed         2.3405e-03  1.0820e-03  2.1632 0.0340968 *
## wsta        -1.1357e-03  8.9769e-04 -1.2651 0.2102213
## wloc         5.8983e-04  2.4003e-03  0.2457 0.8066400
## mix         -2.3924e-01  6.2632e-01 -0.3820 0.7036869
## pctymle      2.7706e+00  1.4330e+00  1.9334 0.0574191 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
AIC(all_in_model_log_level, k=2)
```

```
## [1] 21.354
```

```r
stargazer(simple_regression_model, all_in_model_log_level,
          type = "text", omit.stat = "f",
          se = list(se.simple_regression_model, se.all_in_model_log_level),
          star.cutoffs = c(0.05, 0.01, 0.001))
```

```
## 
## ===================================================
##                            Dependent variable:     
##                      -------------------------------
##                                log_crmrte          
##                            (1)             (2)     
## ---------------------------------------------------
## prbarr                                  -1.889***  
##                                          (0.380)   
##                                                    
## prbconv                                 -0.656***  
##                                          (0.174)   
##                                                    
## prbpris                                  -0.093    
##                                          (0.399)   
##                                                    
## avgsen                                   -0.008    
##                                          (0.016)
##
## polpc                                    154.835
##                                         (86.523)
##
## density                 0.228***         0.117*
##                          (0.030)         (0.054)
##
## taxpc                                     0.003
##                                          (0.007)
##
## west                                     -0.115
##                                          (0.125)
##
## central                                  -0.101
##                                          (0.092)
##
## urban                                    -0.169
##                                          (0.229)
##
## pctmin80                                 0.010**
##                                          (0.003)
##
## wcon                                     0.0005
##                                          (0.001)
##
## wtuc                                     0.0001
##                                          (0.001)
##
## wtrd                                     0.0003
##                                          (0.002)
##
## wfir                                     -0.001
##                                          (0.001)
##
## wser                                     -0.0001
##                                          (0.002)
##
## wmfg                                     -0.0002
##                                          (0.001)
##
## wfed                                     0.002*
##                                          (0.001)
##
## wsta                                     -0.001
##                                          (0.001)
##
## wloc                                      0.001
##                                          (0.002)
##
## mix                                      -0.239
##                                          (0.626)
##
## pctymle                                   2.771
##                                         (1.433)
##
## Constant               -3.869***       -4.026***
##                         (0.069)         (0.848)
##
## ---------------------------------------------------
## Observations              90              90
## R2                       0.401           0.854
## Adjusted R2              0.394           0.806
## Residual Std. Error 0.427 (df = 88) 0.242 (df = 67)
## ===================================================
## Note:                 *p<0.05; **p<0.01; ***p<0.001
```

Unsuprisingly, the r-squared of the "kitchen sink" model is substantially higher (85.4% vs. 40.1%). More importantly, the adjusted r-squared which accounts for the number of variables in the models, is also higher (80.6% vs 39.4%). Interestingly, density is no longer the variable with the highest statistical significance. The coefficients show the effect after all the other variables have been controlled for (partialled out). In the "kitchen sink" model prbarr and prbconv both have the lowest p-values.

Model 2: Balanced Model

We took two approaches to building the balanced model. We used a bottom up approach that relied on both the correlation matrix and stepwide regression. We also used a top down approach that started with the "kitchen sink" model and excluded variables. Both methods are discussed below. Both approaches relied on our categories of variables to simplify the process.

```
base_forward = lm(log_crmrte ~ density,
                             data = data_crmrte)
forward_step = step(base_forward, scope = formula(all_in_model_log_level), direction = "forward")
```

```
## Start:  AIC=-151.11
## log_crmrte ~ density
##
##            Df Sum of Sq    RSS     AIC
## + west      1   2.94477 13.116 -167.34
## + prbconv   1   2.59934 13.461 -165.00
## + prbarr    1   2.33206 13.729 -163.23
## + pctmin80  1   2.11480 13.946 -161.82
## + pctymle   1   1.14375 14.917 -155.76
## + wfed      1   0.92973 15.131 -154.48
## + taxpc     1   0.72379 15.337 -153.26
## + wser      1   0.53323 15.527 -152.15
## + wcon      1   0.38803 15.672 -151.31
## <none>                  16.061 -151.11
## + avgsen    1   0.24653 15.814 -150.50
## + polpc     1   0.22427 15.836 -150.38
## + wfir      1   0.10688 15.954 -149.71
## + urban     1   0.06260 15.998 -149.46
## + central   1   0.05318 16.007 -149.41
## + mix       1   0.03960 16.021 -149.33
## + wmfg      1   0.02854 16.032 -149.27
## + wsta      1   0.02520 16.035 -149.25
## + prbpris   1   0.02040 16.040 -149.22
## + wtrd      1   0.01224 16.048 -149.18
## + wtuc      1   0.00322 16.057 -149.13
## + wloc      1   0.00022 16.060 -149.11
##
## Step:  AIC=-167.34
## log_crmrte ~ density + west
##
##            Df Sum of Sq    RSS     AIC
## + prbconv   1   2.50648 10.609 -184.43
## + prbarr    1   1.68746 11.428 -177.73
## + pctymle   1   1.05265 12.063 -172.87
## + central   1   0.86332 12.252 -171.47
## + wser      1   0.68045 12.435 -170.13
## + wfed      1   0.57636 12.539 -169.38
## + taxpc     1   0.36020 12.756 -167.85
## <none>                  13.116 -167.34
## + pctmin80  1   0.19632 12.919 -166.70
## + wcon      1   0.14814 12.968 -166.36
## + avgsen    1   0.08265 13.033 -165.91
## + wmfg      1   0.07694 13.039 -165.87
## + wfir      1   0.06497 13.051 -165.79
## + mix       1   0.06018 13.056 -165.75
## + prbpris   1   0.04089 13.075 -165.62
## + wtuc      1   0.03304 13.083 -165.57
## + urban     1   0.03096 13.085 -165.55
## + polpc     1   0.02985 13.086 -165.54
## + wloc      1   0.01397 13.102 -165.44
## + wsta      1   0.00583 13.110 -165.38
## + wtrd      1   0.00529 13.111 -165.38
##
## Step:  AIC=-184.43
## log_crmrte ~ density + west + prbconv
##
##            Df Sum of Sq     RSS     AIC
## + prbarr    1   2.34643  8.2629 -204.92
## + wfed      1   0.83512  9.7742 -189.81
## + central   1   0.73021  9.8791 -188.84
## + mix       1   0.71874  9.8906 -188.74
## + pctymle   1   0.65915  9.9502 -188.20
## + pctmin80  1   0.32029 10.2890 -185.19
## + taxpc     1   0.26004 10.3493 -184.66
## + wmfg      1   0.24268 10.3666 -184.51
## <none>                  10.6093 -184.43
## + wcon      1   0.13057 10.4787 -183.54
## + wtuc      1   0.08905 10.5203 -183.19
## + urban     1   0.04370 10.5656 -182.80
## + polpc     1   0.03042 10.5789 -182.69
## + wloc      1   0.02897 10.5803 -182.67
## + prbpris   1   0.02395 10.5854 -182.63
## + wser      1   0.00516 10.6042 -182.47
## + wtrd      1   0.00479 10.6045 -182.47
## + wsta      1   0.00284 10.6065 -182.45
## + wfir      1   0.00233 10.6070 -182.45
## + avgsen    1   0.00006 10.6093 -182.43
```

```
##
## Step:  AIC=-204.92
## log_crmrte ~ density + west + prbconv + prbarr
##
##            Df Sum of Sq    RSS     AIC
## + polpc     1   1.41147 6.8514 -219.78
## + wfed      1   0.81108 7.4518 -212.22
## + central   1   0.76023 7.5026 -211.61
## + pctmin80  1   0.70778 7.5551 -210.98
## + pctymle   1   0.30509 7.9578 -206.31
## + wmfg      1   0.22113 8.0418 -205.37
## + taxpc     1   0.21216 8.0507 -205.26
## + wloc      1   0.19827 8.0646 -205.11
## <none>                  8.2629 -204.92
## + avgsen    1   0.12246 8.1404 -204.27
## + wtuc      1   0.11422 8.1487 -204.18
## + mix       1   0.08199 8.1809 -203.82
## + wsta      1   0.04978 8.2131 -203.47
## + wcon      1   0.03148 8.2314 -203.27
## + wser      1   0.02647 8.2364 -203.21
## + wtrd      1   0.01465 8.2482 -203.08
## + urban     1   0.01199 8.2509 -203.05
## + wfir      1   0.00439 8.2585 -202.97
## + prbpris   1   0.00072 8.2622 -202.93
##
## Step:  AIC=-219.78
## log_crmrte ~ density + west + prbconv + prbarr + polpc
##
##            Df Sum of Sq    RSS     AIC
## + pctmin80  1   1.23128 5.6201 -235.61
## + central   1   0.63049 6.2209 -226.47
## + wfed      1   0.59242 6.2590 -225.92
## <none>                  6.8514 -219.78
## + wsta      1   0.13668 6.7147 -219.59
## + pctymle   1   0.10414 6.7473 -219.16
## + wtuc      1   0.05144 6.8000 -218.46
## + wcon      1   0.04219 6.8092 -218.34
## + wmfg      1   0.03521 6.8162 -218.25
## + mix       1   0.03490 6.8165 -218.24
## + avgsen    1   0.01955 6.8319 -218.04
## + wtrd      1   0.01788 6.8335 -218.02
## + urban     1   0.01590 6.8355 -217.99
## + wloc      1   0.00690 6.8445 -217.87
## + wfir      1   0.00240 6.8490 -217.81
## + prbpris   1   0.00027 6.8511 -217.79
## + taxpc     1   0.00020 6.8512 -217.78
## + wser      1   0.00014 6.8513 -217.78
##
## Step:  AIC=-235.61
## log_crmrte ~ density + west + prbconv + prbarr + polpc + pctmin80
##
##           Df Sum of Sq    RSS     AIC
## + wfed     1   0.53345 5.0867 -242.59
## + wsta     1   0.31439 5.3057 -238.79
## + wcon     1   0.22014 5.4000 -237.21
## + mix      1   0.19495 5.4252 -236.79
## + urban    1   0.14253 5.4776 -235.92
## + central  1   0.13315 5.4870 -235.77
## <none>                 5.6201 -235.61
## + wtuc     1   0.12224 5.4979 -235.59
## + wtrd     1   0.09692 5.5232 -235.18
## + pctymle  1   0.07319 5.5469 -234.79
## + wloc     1   0.06955 5.5506 -234.73
## + wser     1   0.05762 5.5625 -234.54
## + wmfg     1   0.04954 5.5706 -234.41
## + prbpris  1   0.01786 5.6023 -233.90
## + taxpc    1   0.00591 5.6142 -233.71
## + avgsen   1   0.00101 5.6191 -233.63
## + wfir     1   0.00073 5.6194 -233.62
##
## Step:  AIC=-242.59
## log_crmrte ~ density + west + prbconv + prbarr + polpc + pctmin80 +
##     wfed
##
##           Df Sum of Sq    RSS     AIC
## + wsta     1   0.36526 4.7214 -247.29
## + central  1   0.26095 4.8257 -245.33
## + pctymle  1   0.17155 4.9151 -243.67
## + wfir     1   0.15143 4.9353 -243.31
## <none>                 5.0867 -242.59
## + urban    1   0.08107 5.0056 -242.03
## + taxpc    1   0.06932 5.0174 -241.82
## + wcon     1   0.05252 5.0342 -241.52
## + mix      1   0.05158 5.0351 -241.50
## + wser     1   0.03779 5.0489 -241.26
## + prbpris  1   0.02681 5.0599 -241.06
## + wtuc     1   0.02287 5.0638 -240.99
## + avgsen   1   0.01320 5.0735 -240.82
## + wmfg     1   0.00108 5.0856 -240.61
## + wtrd     1   0.00084 5.0858 -240.60
## + wloc     1   0.00053 5.0862 -240.60
##
## Step:  AIC=-247.29
## log_crmrte ~ density + west + prbconv + prbarr + polpc + pctmin80 +
##     wfed + wsta
##
##           Df Sum of Sq    RSS     AIC
## + pctymle  1  0.269426 4.4520 -250.58
## + central  1  0.222728 4.4987 -249.64
## <none>                 4.7214 -247.29
## + wfir     1  0.088681 4.6327 -247.00
## + mix      1  0.068999 4.6524 -246.62
## + prbpris  1  0.040292 4.6811 -246.06
## + urban    1  0.026383 4.6950 -245.80
## + wser     1  0.025477 4.6960 -245.78
## + taxpc    1  0.024825 4.6966 -245.77
## + wtrd     1  0.019294 4.7021 -245.66
## + wcon     1  0.018232 4.7032 -245.64
## + wmfg     1  0.008071 4.7134 -245.45
## + wloc     1  0.006051 4.7154 -245.41
## + avgsen   1  0.000694 4.7207 -245.31
## + wtuc     1  0.000097 4.7213 -245.29
##
## Step:  AIC=-250.58
## log_crmrte ~ density + west + prbconv + prbarr + polpc + pctmin80 +
##     wfed + wsta + pctymle
##
##           Df Sum of Sq    RSS     AIC
## + central  1  0.160121 4.2919 -251.88
## + taxpc    1  0.106562 4.3454 -250.76
## <none>                 4.4520 -250.58
## + wfir     1  0.090730 4.3613 -250.43
## + mix      1  0.037862 4.4141 -249.35
## + prbpris  1  0.027609 4.4244 -249.14
## + wser     1  0.026573 4.4254 -249.12
## + wcon     1  0.020851 4.4312 -249.00
## + urban    1  0.016567 4.4354 -248.92
## + wmfg     1  0.009857 4.4421 -248.78
## + wloc     1  0.007999 4.4440 -248.74
## + wtrd     1  0.006269 4.4457 -248.71
## + avgsen   1  0.004132 4.4479 -248.66
## + wtuc     1  0.000726 4.4513 -248.60
##
## Step:  AIC=-251.88
## log_crmrte ~ density + west + prbconv + prbarr + polpc + pctmin80 +
##     wfed + wsta + pctymle + central
##
##           Df Sum of Sq    RSS     AIC
## <none>                 4.2919 -251.88
## + wfir     1  0.074114 4.2178 -251.44
## + taxpc    1  0.065113 4.2268 -251.25
## + urban    1  0.043628 4.2483 -250.80
## + wcon     1  0.039905 4.2520 -250.72
## + avgsen   1  0.030713 4.2612 -250.52
## + mix      1  0.024668 4.2672 -250.40
## + wloc     1  0.019087 4.2728 -250.28
## + wmfg     1  0.010493 4.2814 -250.10
## + prbpris  1  0.007598 4.2843 -250.04
## + wtuc     1  0.003155 4.2887 -249.94
## + wtrd     1  0.002043 4.2898 -249.92
## + wser     1  0.001662 4.2902 -249.91
```

With the top down approach, we started with model 3 and looked to exclude variables that weren't as predictive. We ran hypothesis testing on all five groups, one group at a time.

```
#deterrent
linearHypothesis(all_in_model_log_level,
                 c("prbarr = 0", "prbconv = 0", "prbpris = 0",
                   "avgsen = 0", "polpc = 0"),
                 vcov = vcovHC)
```

```
## Linear hypothesis test
```

```
##
## Hypothesis:
## prbarr = 0
## prbconv = 0
## prbpris = 0
## avgsen = 0
## polpc = 0
##
## Model 1: restricted model
## Model 2: log_crmrte ~ prbarr + prbconv + prbpris + avgsen + polpc + density +
##     taxpc + west + central + urban + pctmin80 + wcon + wtuc +
##     wtrd + wfir + wser + wmfg + wfed + wsta + wloc + mix + pctymle
##
## Note: Coefficient covariance matrix supplied.
##
##   Res.Df Df      F    Pr(>F)
## 1     72
## 2     67  5 6.0582 0.0001101 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#wage
linearHypothesis(all_in_model_log_level,
                 c("wcon = 0", "wtuc = 0", "wtrd = 0",
                   "wfir = 0", "wser = 0", "wmfg = 0",
                   "wfed = 0", "wsta = 0", "wloc = 0"),
                 vcov = vcovHC)
```

```
## Linear hypothesis test
##
## Hypothesis:
## wcon = 0
## wtuc = 0
## wtrd = 0
## wfir = 0
## wser = 0
## wmfg = 0
## wfed = 0
## wsta = 0
## wloc = 0
##
## Model 1: restricted model
## Model 2: log_crmrte ~ prbarr + prbconv + prbpris + avgsen + polpc + density +
##     taxpc + west + central + urban + pctmin80 + wcon + wtuc +
##     wtrd + wfir + wser + wmfg + wfed + wsta + wloc + mix + pctymle
##
## Note: Coefficient covariance matrix supplied.
##
##   Res.Df Df     F Pr(>F)
## 1     76
## 2     67  9 1.372 0.2185
```

```
#region
linearHypothesis(all_in_model_log_level,
                 c("west = 0", "central = 0"),
                 vcov = vcovHC)
```

```
## Linear hypothesis test
##
## Hypothesis:
## west = 0
## central = 0
##
## Model 1: restricted model
## Model 2: log_crmrte ~ prbarr + prbconv + prbpris + avgsen + polpc + density +
##     taxpc + west + central + urban + pctmin80 + wcon + wtuc +
##     wtrd + wfir + wser + wmfg + wfed + wsta + wloc + mix + pctymle
##
## Note: Coefficient covariance matrix supplied.
##
##   Res.Df Df     F Pr(>F)
## 1     69
## 2     67  2 0.623 0.5394
```

```
#urban
linearHypothesis(all_in_model_log_level,
                 c("urban = 0"),
                 vcov = vcovHC)
```

```
## Linear hypothesis test
##
## Hypothesis:
## urban = 0
##
## Model 1: restricted model
## Model 2: log_crmrte ~ prbarr + prbconv + prbpris + avgsen + polpc + density +
##     taxpc + west + central + urban + pctmin80 + wcon + wtuc +
##     wtrd + wfir + wser + wmfg + wfed + wsta + wloc + mix + pctymle
##
## Note: Coefficient covariance matrix supplied.
##
##   Res.Df Df      F Pr(>F)
## 1     68
## 2     67  1 0.5474  0.462
```

```
#demographic
linearHypothesis(all_in_model_log_level,
                 c("density = 0", "taxpc = 0", "pctmin80 = 0",
                   "mix = 0", "pctymle = 0"),
                 vcov = vcovHC)
```

```
## Linear hypothesis test
##
## Hypothesis:
```

```
## density = 0
## taxpc = 0
## pctmin80 = 0
## mix = 0
## pctymle = 0
##
## Model 1: restricted model
## Model 2: log_crmrte ~ prbarr + prbconv + prbpris + avgsen + polpc + density +
##     taxpc + west + central + urban + pctmin80 + wcon + wtuc +
##     wtrd + wfir + wser + wmfg + wfed + wsta + wloc + mix + pctymle
##
## Note: Coefficient covariance matrix supplied.
##
##   Res.Df Df      F   Pr(>F)
## 1     72
## 2     67  5 3.9627 0.003298 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

/ The hypothesis tests below show that of the five groups the only groups that are jointly significant are the deterrent data and the demographic data. These tests measure whether removing all the variables within a group reduces the r-squared by s attsitically signficant amount. We will re-run the models and compare.

```
balanced_model_top_1 <- lm(log_crmrte ~ prbarr + prbconv + prbpris
                             + avgsen + polpc + density
                             + taxpc + pctmin80 + mix + pctymle,
                             data = data_crmrte)
se.balanced_model_top_1 = sqrt(diag(vcovHC(balanced_model_top_1)))
coeftest(balanced_model_top_1, vcov = vcovHC)
```

```
##
## t test of coefficients:
##
##                Estimate  Std. Error t value  Pr(>|t|)
## (Intercept)  -3.3522918   0.3558217 -9.4213 1.467e-14 ***
## prbarr       -1.9627484   0.4014808 -4.8888 5.228e-06 ***
## prbconv      -0.7672158   0.1366862 -5.6130 2.846e-07 ***
## prbpris      -0.0764993   0.4732818 -0.1616  0.872005
## avgsen       -0.0044749   0.0140406 -0.3187  0.750789
## polpc       176.1347220  82.5884550  2.1327  0.036056 *
## density       0.1135225   0.0351279  3.2317  0.001796 **
## taxpc         0.0020988   0.0055753  0.3764  0.707593
## pctmin80      0.0125062   0.0016215  7.7128 3.155e-11 ***
## mix          -0.7304967   0.5396416 -1.3537  0.179702
## pctymle       1.3832565   1.6211791  0.8532  0.396105
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
AIC(balanced_model_top_1, k=2)
```

```
## [1] 27.4514
```

```r
stargazer(simple_regression_model, all_in_model_log_level, balanced_model_top_1,
          type = "text", omit.stat = "f",
          se = list(se.simple_regression_model, se.all_in_model_log_level, se.balanced_model_top_1),
          star.cutoffs = c(0.05, 0.01, 0.001))
```

```
##
## ===================================================================
##                                    Dependent variable:
##                      -----------------------------------------------
##                                        log_crmrte
##                           (1)             (2)             (3)
## -------------------------------------------------------------------
## prbarr                                 -1.889***       -1.963***
##                                         (0.380)         (0.401)
##
## prbconv                                -0.656***       -0.767***
##                                         (0.174)         (0.137)
##
## prbpris                                 -0.093          -0.076
##                                         (0.399)         (0.473)
##
## avgsen                                  -0.008          -0.004
##                                         (0.016)         (0.014)
##
## polpc                                   154.835        176.135*
##                                        (86.523)        (82.588)
##
## density                 0.228***         0.117*          0.114**
##                          (0.030)        (0.054)         (0.035)
##
## taxpc                                    0.003           0.002
##                                         (0.007)         (0.006)
##
## west                                    -0.115
##                                         (0.125)
##
## central                                 -0.101
##                                         (0.092)
##
## urban                                   -0.169
##                                         (0.229)
##
## pctmin80                                0.010**        0.013***
##                                         (0.003)         (0.002)
##
## wcon                                    0.0005
##                                         (0.001)
##
## wtuc                                    0.0001
##                                         (0.001)
##
## wtrd                                    0.0003
##                                         (0.002)
```

```
##
## wfir                                    -0.001
##                                         (0.001)
##
## wser                                    -0.0001
##                                         (0.002)
##
## wmfg                                    -0.0002
##                                         (0.001)
##
## wfed                                    0.002*
##                                         (0.001)
##
## wsta                                    -0.001
##                                         (0.001)
##
## wloc                                     0.001
##                                         (0.002)
##
## mix                                     -0.239          -0.730
##                                         (0.626)         (0.540)
##
## pctymle                                  2.771           1.383
##                                         (1.433)         (1.621)
##
## Constant               -3.869***       -4.026***       -3.352***
##                         (0.069)         (0.848)         (0.356)
##
## --------------------------------------------------------------------
## Observations              90              90              90
## R2                       0.401           0.854           0.796
## Adjusted R2              0.394           0.806           0.770
## Residual Std. Error 0.427 (df = 88) 0.242 (df = 67) 0.263 (df = 79)
## ====================================================================
## Note:                                 *p<0.05; **p<0.01; ***p<0.001
```

Our adjusted r-squared has only fallen from 80.6% to 77.6% but we have dropped 12 variables. This is a much more parisimous model. In order to double check wages, we decided to try to one more model that included just the median wage from all industries. The fundamental concept behind this is that the median could capture all opportunity for potential criminals, and it has the benefit of not being affected by the outlier in wser. Unfortunately, though it was much better, it was still not predictive.

```r
balanced_model_top_2 <- lm(log_crmrte ~ prbarr + prbconv + prbpris
                             + avgsen + polpc + density
                             + taxpc + pctmin80 + mix + pctymle
                             + median_wage,
                             data = data_crmrte)
se.balanced_model_top_2 = sqrt(diag(vcovHC(balanced_model_top_2)))
coeftest(balanced_model_top_2, vcov = vcovHC)
```

```
##
## t test of coefficients:
##
```

```
##                Estimate  Std. Error t value  Pr(>|t|)    
## (Intercept)  -3.7018614   0.6241512 -5.9310 7.799e-08 ***
## prbarr       -1.9379154   0.4176289 -4.6403 1.381e-05 ***
## prbconv      -0.7650603   0.1360985 -5.6214 2.826e-07 ***
## prbpris      -0.1114352   0.4509889 -0.2471  0.805487    
## avgsen       -0.0046181   0.0137443 -0.3360  0.737770    
## polpc       167.9127417  85.2311288  1.9701  0.052378 .  
## density       0.0977596   0.0343625  2.8450  0.005671 ** 
## taxpc         0.0020705   0.0056750  0.3648  0.716214    
## pctmin80      0.0123893   0.0016202  7.6467 4.534e-11 ***
## mix          -0.5896970   0.5806401 -1.0156  0.312961    
## pctymle       1.5670818   1.9109317  0.8201  0.414680    
## median_wage   0.0011536   0.0014133  0.8162  0.416848    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
AIC(balanced_model_top_2, k=2)
```

```
## [1] 28.07462
```

```r
stargazer(simple_regression_model, all_in_model_log_level, balanced_model_top_1, balanced_model_top_2,
          type = "text", omit.stat = "f",
          se = list(se.simple_regression_model, se.all_in_model_log_level,
                    se.balanced_model_top_1, se.balanced_model_top_2),
          star.cutoffs = c(0.05, 0.01, 0.001))
```

```
## 
## ===================================================================================
##                                            Dependent variable:                     
##                      ---------------------------------------------------------------
##                                                log_crmrte                          
##                            (1)             (2)             (3)             (4)     
## -----------------------------------------------------------------------------------
## prbarr                                  -1.889***       -1.963***       -1.938***  
##                                          (0.380)         (0.401)         (0.418)   
##                                                                                    
## prbconv                                 -0.656***       -0.767***       -0.765***  
##                                          (0.174)         (0.137)         (0.136)   
##                                                                                    
## prbpris                                  -0.093          -0.076          -0.111    
##                                          (0.399)         (0.473)         (0.451)   
##                                                                                    
## avgsen                                   -0.008          -0.004          -0.005    
##                                          (0.016)         (0.014)         (0.014)   
##                                                                                    
## polpc                                    154.835        176.135*        167.913*   
##                                         (86.523)        (82.588)        (85.231)   
##                                                                                    
## density                 0.228***         0.117*          0.114**         0.098**   
##                          (0.030)         (0.054)         (0.035)         (0.034)   
##                                                                                    
## taxpc                                     0.003           0.002           0.002    
##                                          (0.007)         (0.006)         (0.006)   
```

```
##
## west                                    -0.115
##                                         (0.125)
##
## central                                 -0.101
##                                         (0.092)
##
## urban                                   -0.169
##                                         (0.229)
##
## pctmin80                                0.010**         0.013***        0.012***
##                                         (0.003)          (0.002)         (0.002)
##
## wcon                                    0.0005
##                                         (0.001)
##
## wtuc                                    0.0001
##                                         (0.001)
##
## wtrd                                    0.0003
##                                         (0.002)
##
## wfir                                    -0.001
##                                         (0.001)
##
## wser                                    -0.0001
##                                         (0.002)
##
## wmfg                                    -0.0002
##                                         (0.001)
##
## wfed                                    0.002*
##                                         (0.001)
##
## wsta                                    -0.001
##                                         (0.001)
##
## wloc                                     0.001
##                                         (0.002)
##
## mix                                     -0.239          -0.730          -0.590
##                                         (0.626)          (0.540)         (0.581)
##
## pctymle                                  2.771           1.383           1.567
##                                         (1.433)          (1.621)         (1.911)
##
## median_wage                                                                0.001
##                                                                           (0.001)
##
## Constant               -3.869***       -4.026***       -3.352***       -3.702***
##                         (0.069)         (0.848)         (0.356)         (0.624)
##
## -----------------------------------------------------------------------------------
## Observations              90              90              90              90
## R2                       0.401           0.854           0.796           0.799
## Adjusted R2              0.394           0.806           0.770           0.770
## Residual Std. Error 0.427 (df = 88) 0.242 (df = 67) 0.263 (df = 79) 0.263 (df = 78)
## ===================================================================================
## Note:                                                 *p<0.05; **p<0.01; ***p<0.001
```

Three of the five groups have been eliminated, with only the deterrent and demographic groups remaining. We will use step wise regression to evaluate.

```
base_backward = lm(log_crmrte ~ prbarr + prbconv + prbpris
                             + avgsen + polpc + density
                             + taxpc + pctmin80 + mix + pctymle,
                             data = data_crmrte)
backward_step = step(base_backward, scope = formula(base_backward), direction = "backward")
```

```
## Start:  AIC=-229.96
## log_crmrte ~ prbarr + prbconv + prbpris + avgsen + polpc + density +
##     taxpc + pctmin80 + mix + pctyml e
##
##            Df Sum of Sq    RSS     AIC
## - prbpris   1    0.0031 5.4786 -231.91
## - avgsen    1    0.0103 5.4858 -231.79
## - taxpc     1    0.0474 5.5229 -231.18
## - pctymle   1    0.0771 5.5526 -230.70
## <none>                  5.4755 -229.96
## - mix       1    0.2159 5.6914 -228.48
## - polpc     1    1.1452 6.6207 -214.87
## - density   1    1.8044 7.2799 -206.32
## - prbarr    1    2.9989 8.4744 -192.65
## - pctmin80  1    3.4894 8.9649 -187.59
## - prbconv   1    4.2940 9.7695 -179.85
##
## Step:  AIC=-231.91
## log_crmrte ~ prbarr + prbconv + avgsen + polpc + density + taxpc +
##     pctmin80 + mix + pctymle
##
##            Df Sum of Sq    RSS     AIC
## - avgsen    1    0.0091 5.4877 -233.76
## - taxpc     1    0.0529 5.5316 -233.04
## - pctymle   1    0.0815 5.5601 -232.58
## <none>                  5.4786 -231.91
## - mix       1    0.2221 5.7007 -230.33
## - polpc     1    1.1489 6.6275 -216.77
## - density   1    1.8111 7.2897 -208.20
## - prbarr    1    2.9964 8.4750 -194.64
## - pctmin80  1    3.5028 8.9814 -189.42
## - prbconv   1    4.2956 9.7742 -181.81
##
## Step:  AIC=-233.76
## log_crmrte ~ prbarr + prbconv + polpc + density + taxpc + pctmin80 +
##     mix + pctymle
```

```
##
##            Df Sum of Sq    RSS     AIC
## - taxpc     1    0.0545 5.5422 -234.87
## - pctymle   1    0.0789 5.5667 -234.47
## <none>                  5.4877 -233.76
## - mix       1    0.2139 5.7016 -232.32
## - polpc     1    1.2273 6.7150 -217.59
## - density   1    1.8088 7.2965 -210.12
## - prbarr    1    3.0183 8.5060 -196.31
## - pctmin80  1    3.5470 9.0348 -190.89
## - prbconv   1    4.3218 9.8095 -183.48
##
## Step:  AIC=-234.87
## log_crmrte ~ prbarr + prbconv + polpc + density + pctmin80 +
##     mix + pctymle
##
##            Df Sum of Sq     RSS     AIC
## - pctymle   1    0.0525  5.5947 -236.02
## <none>                   5.5422 -234.87
## - mix       1    0.2148  5.7571 -233.44
## - polpc     1    1.7384  7.2806 -212.31
## - density   1    1.8905  7.4328 -210.45
## - prbarr    1    3.5982  9.1404 -191.84
## - pctmin80  1    3.6798  9.2220 -191.04
## - prbconv   1    4.8508 10.3931 -180.28
##
## Step:  AIC=-236.02
## log_crmrte ~ prbarr + prbconv + polpc + density + pctmin80 +
##     mix
##
##            Df Sum of Sq     RSS     AIC
## <none>                   5.5947 -236.02
## - mix       1    0.2319  5.8267 -234.36
## - density   1    1.8553  7.4500 -212.24
## - polpc     1    1.9402  7.5349 -211.22
## - pctmin80  1    3.7524  9.3471 -191.83
## - prbarr    1    3.9930  9.5877 -189.54
## - prbconv   1    5.3975 10.9922 -177.24
```

```
balanced_model_top_3 <- lm(log_crmrte ~ mix + density
                           + polpc + pctmin80
                           + prbarr + prbconv,
                             data = data_crmrte)
se.balanced_model_top_3 = sqrt(diag(vcovHC(balanced_model_top_3)))
coeftest(balanced_model_top_2, vcov = vcovHC)
```

```
##
## t test of coefficients:
##
##                 Estimate  Std. Error t value  Pr(>|t|)
## (Intercept)  -3.7018614   0.6241512 -5.9310 7.799e-08 ***
## prbarr       -1.9379154   0.4176289 -4.6403 1.381e-05 ***
## prbconv      -0.7650603   0.1360985 -5.6214 2.826e-07 ***
## prbpris      -0.1114352   0.4509889 -0.2471  0.805487
## avgsen       -0.0046181   0.0137443 -0.3360  0.737770
## polpc       167.9127417  85.2311288  1.9701  0.052378 .
## density       0.0977596   0.0343625  2.8450  0.005671 **
## taxpc         0.0020705   0.0056750  0.3648  0.716214
## pctmin80      0.0123893   0.0016202  7.6467 4.534e-11 ***
## mix          -0.5896970   0.5806401 -1.0156  0.312961
## pctymle       1.5670818   1.9109317  0.8201  0.414680
## median_wage   0.0011536   0.0014133  0.8162  0.416848
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
AIC(balanced_model_top_2, k=2)
```

```
## [1] 28.07462
```

```
stargazer(simple_regression_model, all_in_model_log_level, balanced_model_top_1, balanced_model_top_3,
          type = "text", omit.stat = "f",
          se = list(se.simple_regression_model, se.all_in_model_log_level,
                    se.balanced_model_top_1, se.balanced_model_top_3),
          star.cutoffs = c(0.05, 0.01, 0.001))
```

```
##
## ===================================================================================
##                                            Dependent variable:
##                      ---------------------------------------------------------------
##                                                log_crmrte
##                           (1)             (2)             (3)             (4)
## -----------------------------------------------------------------------------------
## prbarr                                 -1.889***       -1.963***       -2.100***
##                                         (0.380)         (0.401)         (0.436)
##
## prbconv                                -0.656***       -0.767***       -0.809***
##                                         (0.174)         (0.137)         (0.126)
##
## prbpris                                 -0.093          -0.076
##                                         (0.399)         (0.473)
##
## avgsen                                  -0.008          -0.004
##                                         (0.016)         (0.014)
##
## polpc                                   154.835        176.135*        190.549**
##                                        (86.523)        (82.588)        (71.937)
##
## density                  0.228***         0.117*          0.114**        0.113***
##                          (0.030)         (0.054)         (0.035)         (0.027)
##
## taxpc                                    0.003           0.002
##                                         (0.007)         (0.006)
##
## west                                    -0.115
##                                         (0.125)
##
## central                                 -0.101
##                                         (0.092)
##
## urban                                   -0.169
##                                         (0.229)
##
## pctmin80                                0.010**         0.013***        0.013***
##                                         (0.003)          (0.002)         (0.001)
##
## wcon                                    0.0005
##                                         (0.001)
##
## wtuc                                    0.0001
##                                         (0.001)
##
## wtrd                                    0.0003
##                                         (0.002)
##
## wfir                                    -0.001
##                                         (0.001)
##
## wser                                    -0.0001
##                                         (0.002)
##
## wmfg                                    -0.0002
##                                         (0.001)
##
## wfed                                    0.002*
##                                         (0.001)
##
## wsta                                    -0.001
##                                         (0.001)
##
## wloc                                     0.001
##                                         (0.002)
##
## mix                                     -0.239          -0.730          -0.745
##                                         (0.626)          (0.540)         (0.474)
##
## pctymle                                  2.771           1.383
##                                         (1.433)          (1.621)
##
## Constant                -3.869***      -4.026***       -3.352***       -3.197***
##                          (0.069)         (0.848)          (0.356)         (0.237)
##
## -----------------------------------------------------------------------------------
## Observations               90              90              90              90
## R2                        0.401           0.854           0.796           0.791
## Adjusted R2               0.394           0.806           0.770           0.776
## Residual Std. Error 0.427 (df = 88) 0.242 (df = 67) 0.263 (df = 79) 0.260 (df = 83)
## ===================================================================================
## Note:                                                 *p<0.05; **p<0.01; ***p<0.001
```

The difference between the backward and forward model is that the backward model chooses variables for exclusion based on comparing significance while the forward model looks for significance in inclusion. We also used the f-tests (hypothesis tests) to give the backward stepwise regression a head start.

The backward stepwise regression yielded a more reasonable model so that is the model we are choosing for our balanced model. This model strikes a nice balance between parsimony and explanatory power. The variables included are prbarr, prbconv, polpc, density, pctmin80, and mix. Six out of the original 24 independent variables are included. The adjust r-squared is only 3% lower (77.6% vs. 80.6%). It includes a blend of actionable items for the campaign in the deterrent data as well as demographic variables that perhaps can focus the campaign's efforts.