Lab 3

Code

Josiah McDonald & Simon Storey

This lab will contain five parts:
1. Introduction & Statement of Purpose
2. EDA
3. Model Specifications
4. Potential Omitted Variables
5. Results & Recommendations

1. Introduction & Statement of Purpose

The purpose of this lab is examine and understand the relationship between crimes committed per person and a number of other variables. Once the relationships are better understood we will use them to provide policy recommendations. The data set we use is taken from study by Cornwell and Trumball, but we only look at a single cross section of the data from 1987. The code below shows our process of reading in the data and cleaning up a few errors. While our initial EDA and model building will focus on understanding the entire picture, our recommendations will focus in on activities and polices a local government can control. We think this approach is best, because if we don't understand all factors, we won't be able to give coherent answers to what would work best.

Load our datafile and review its characteristics

Hide

```
crime = read.csv("./crime_v2.csv", stringsAsFactors=FALSE)
str(crime)
```

```
'data.frame':	97 obs. of  25 variables:
 $ county  : int  1 3 5 7 9 11 13 15 17 19 ...
 $ year    : int  87 87 87 87 87 87 87 87 87 87 ...
 $ crmrte  : num  0.0356 0.0153 0.013 0.0268 0.0106 ...
 $ prbarr  : num  0.298 0.132 0.444 0.365 0.518 ...
 $ prbconv : chr  "0.527595997" "1.481480002" "0.267856985" "0.525424004" ...
 $ prbpris : num  0.436 0.45 0.6 0.435 0.443 ...
 $ avgsen  : num  6.71 6.35 6.76 7.14 8.22 ...
 $ polpc   : num  0.001828 0.000746 0.001234 0.00153 0.00086 ...
 $ density : num  2.423 1.046 0.413 0.492 0.547 ...
 $ taxpc   : num  31 26.9 34.8 42.9 28.1 ...
 $ west    : int  0 0 1 0 1 1 0 0 0 0 ...
 $ central : int  1 1 0 1 0 0 0 0 0 0 ...
 $ urban   : int  0 0 0 0 0 0 0 0 0 0 ...
 $ pctmin80: num  20.22 7.92 3.16 47.92 1.8 ...
 $ wcon    : num  281 255 227 375 292 ...
 $ wtuc    : num  409 376 372 398 377 ...
 $ wtrd    : num  221 196 229 191 207 ...
 $ wfir    : num  453 259 306 281 289 ...
 $ wser    : num  274 192 210 257 215 ...
 $ wmfg    : num  335 300 238 282 291 ...
 $ wfed    : num  478 410 359 412 377 ...
 $ wsta    : num  292 363 332 328 367 ...
 $ wloc    : num  312 301 281 299 343 ...
 $ mix     : num  0.0802 0.0302 0.4651 0.2736 0.0601 ...
 $ pctymle : num  0.0779 0.0826 0.0721 0.0735 0.0707 ...
```

Clean up data

Lets removing NA's from the trailing part of the imported data.

Hide

```
crime = na.omit(crime)
```

Then lets set columns that should be factors and/or indicator variables

Hide

```
crime$county = factor(crime$county)
crime$year = factor(crime$year)
crime$west = factor(crime$west)
crime$central = factor(crime$central)
crime$urban = factor(crime$urban)
```

Lets clean up prbconv as it was converted to a chr by the import

Hide

```
crime$prbconv = as.numeric(crime$prbconv)
summary(crime$prbconv)
```

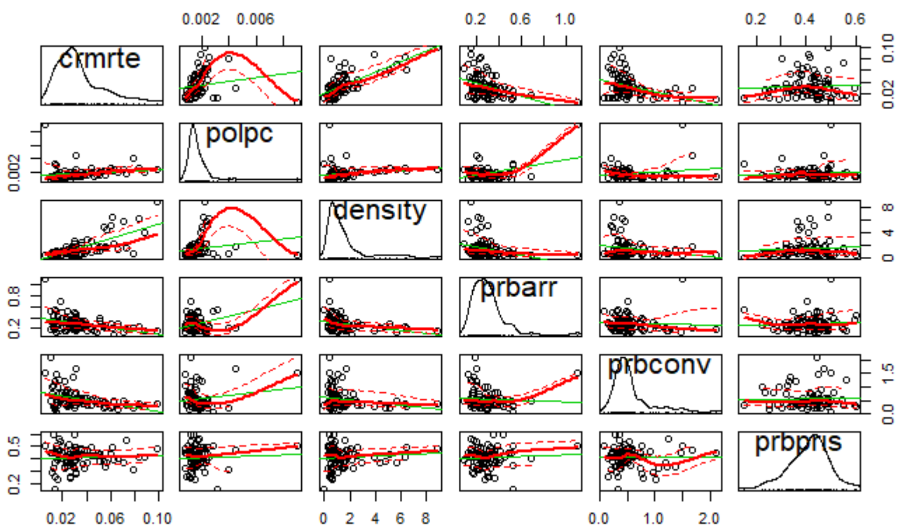
```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
0.06838 0.34541 0.45283 0.55128 0.58886 2.12121 
```

2. EDA

To start our EDA we first take a look at some scatter plot matrices to understand the relationship between crime and other variates.

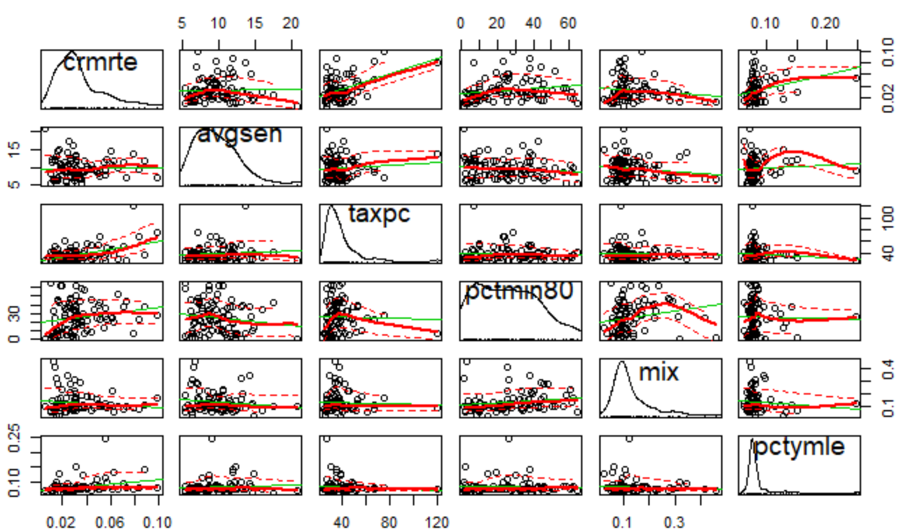
Hide

```
scatterplotMatrix(crime[ , c("crmrte", "polpc", "density", "prbarr", "prbconv","prbpris")])
```



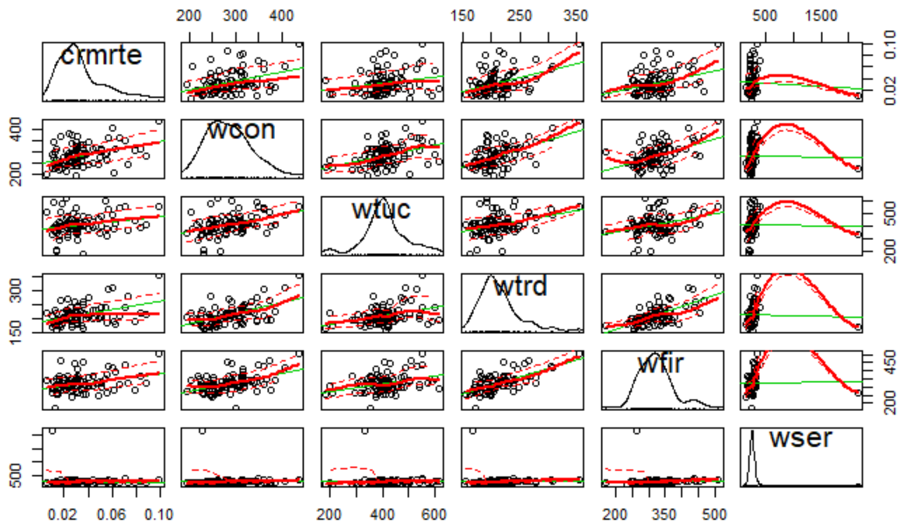
Hide

```
scatterplotMatrix(crime[ , c("crmrte", "avgsen", "taxpc", "pctmin80","mix", "pctymle" )])
```



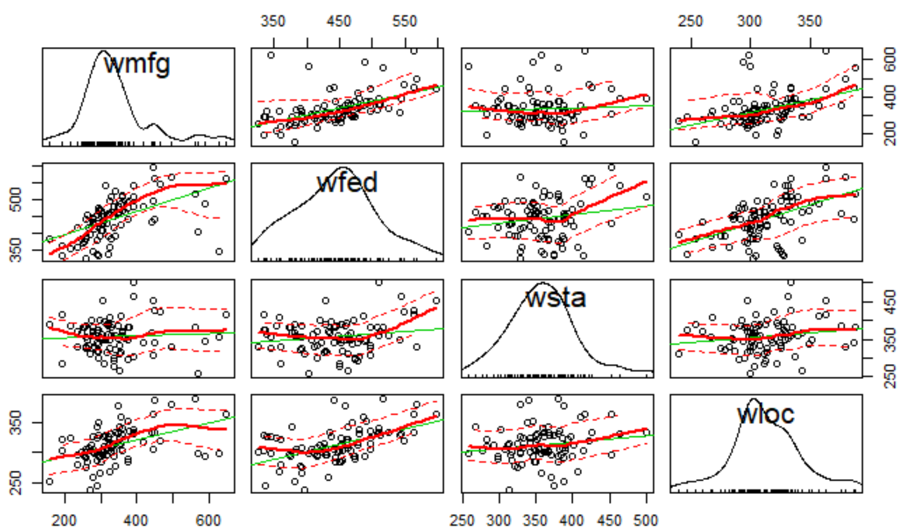
Hide

```
scatterplotMatrix(crime[ , c("crmrte", "wcon", "wtuc","wtrd","wfir","wser")])
```



Hide

```
scatterplotMatrix(crime[ , c("wmfg","wfed","wsta","wloc" )])
```



Review the following values as they all are subject to skewed distributions and find a suitable transform to try and create normality.

The following variables are subject to skew - crmrte - prbarr - polpc - density - taxpc - mix - pctymle

First lets examine the crime variable

Hide

```
summary(crime$crmrte)
```

```
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max. 
0.005533 0.020927 0.029986 0.033400 0.039642 0.098966 
```

Noting the large left skew - we review viable transforms for the crime rate variable

Hide

```
summary(crime$crmrte)
```

```
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max. 
0.005533 0.020927 0.029986 0.033400 0.039642 0.098966 
```

Hide

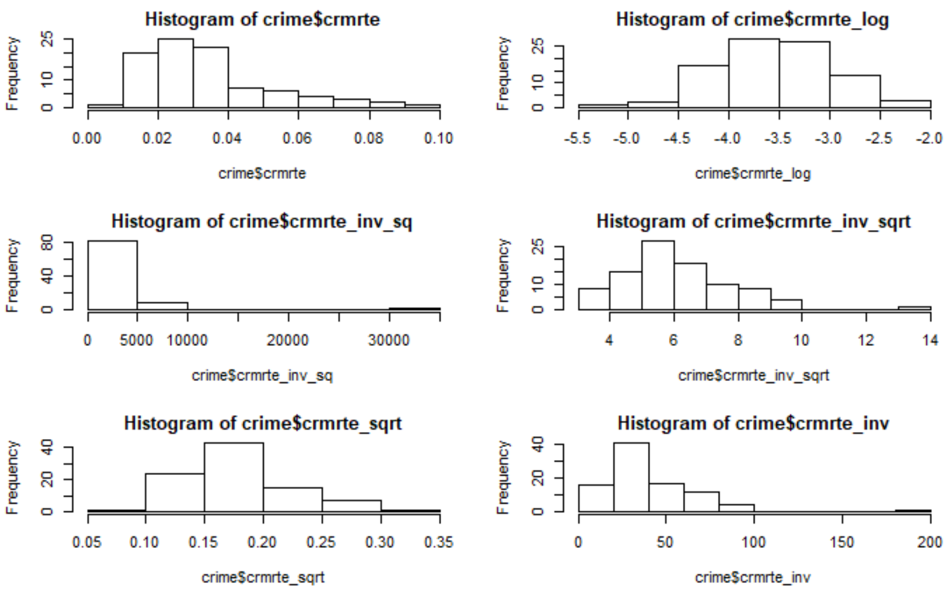
```
par(mfrow=c(3,2))
hist(crime$crmrte)
crime$crmrte_log = log(crime$crmrte)
crime$crmrte_inv_sq = 1/(crime$crmrte^2)
crime$crmrte_inv_sqrt = 1/(sqrt(crime$crmrte))
crime$crmrte_sqrt = sqrt(crime$crmrte)
crime$crmrte_inv = 1/crime$crmrte
hist(crime$crmrte_log)
```

Hide

```
hist(crime$crmrte_inv_sq)
hist(crime$crmrte_inv_sqrt)
```

Hide

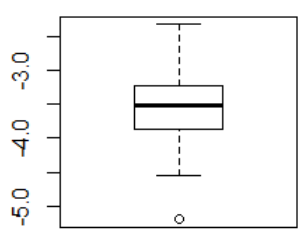
```
hist(crime$crmrte_sqrt)
hist(crime$crmrte_inv)
```



We opt to use a log transform for the crime rate variable. We also look at a boxplot and we notice 1 outlier (which could affect regressions at later juncture)

Hide

```
boxplot(crime$crmrte_log)
```



Next lets transform the taxpc and try a list of transformations (Normal,Lognormal etc). We will also consider a box-cox power transformation strategy.

Hide

```
summary(crime$taxpc)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  25.69   30.66   34.87   38.06   40.95  119.76 
```

Hide

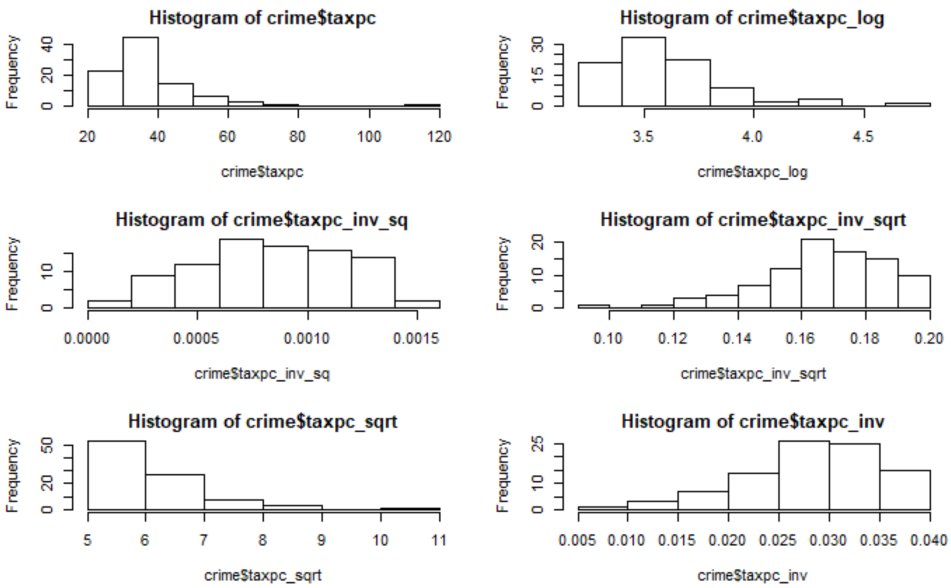
```
par(mfrow=c(3,2))
hist(crime$taxpc)
crime$taxpc_log = log(crime$taxpc)
crime$taxpc_inv_sq = 1/(crime$taxpc^2)
crime$taxpc_inv_sqrt = 1/(sqrt(crime$taxpc))
crime$taxpc_sqrt = sqrt(crime$taxpc)
crime$taxpc_inv = 1/crime$taxpc
hist(crime$taxpc_log)
```

Hide

```
hist(crime$taxpc_inv_sq)
hist(crime$taxpc_inv_sqrt)
```

Hide

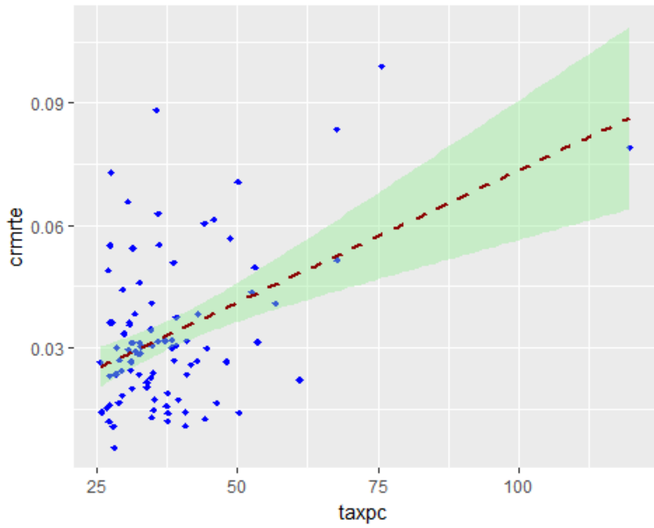
```
hist(crime$taxpc_sqrt)
hist(crime$taxpc_inv)
```



Comparing before and after the transformation

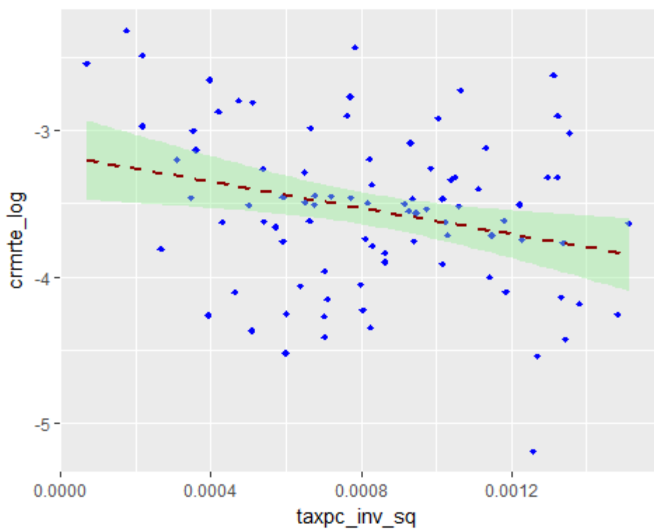
Hide

```
par(mfrow=c(2,1))
# Before
ggplot(crime, aes(x=taxpc, y=crmrte)) +
  geom_point(shape=18, color="blue")+
  geom_smooth(method=lm,  linetype="dashed",
             color="darkred", fill="lightgreen")
```



Hide

```
# After
ggplot(crime, aes(x=taxpc_inv_sq, y=crmrte_log)) +
  geom_point(shape=18, color="blue")+
  geom_smooth(method=lm,  linetype="dashed",
           color="darkred", fill="lightgreen")
```



From the results we can see the inverse square transform produces the best result for the taxpc variable. The plot shows our linear fit and the confidence levels with similiar distribution across the plot.

Next lets focus on population density using the same strategy as defined above.

Hide

```
summary(crime$density)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
0.00002 0.54741 0.96226 1.42884 1.56824 8.82765 
```

Hide

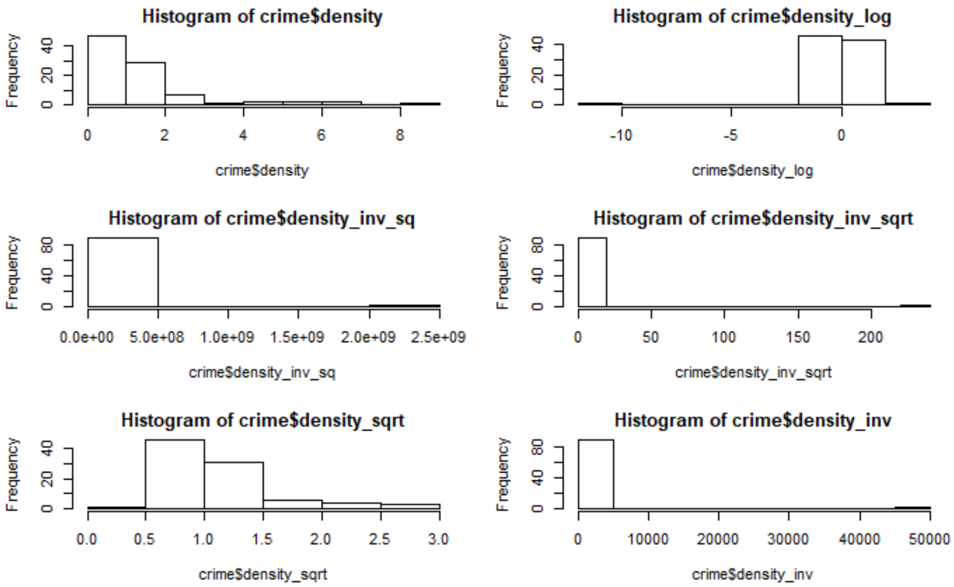
```
par(mfrow=c(3,2))
hist(crime$density)
crime$density_log = log(crime$density)
crime$density_inv_sq = 1/(crime$density^2)
crime$density_inv_sqrt = 1/(sqrt(crime$density))
crime$density_sqrt = sqrt(crime$density)
crime$density_inv = 1/crime$density
hist(crime$density_log)
```

Hide

```
hist(crime$density_inv_sq)
hist(crime$density_inv_sqrt)
```

Hide

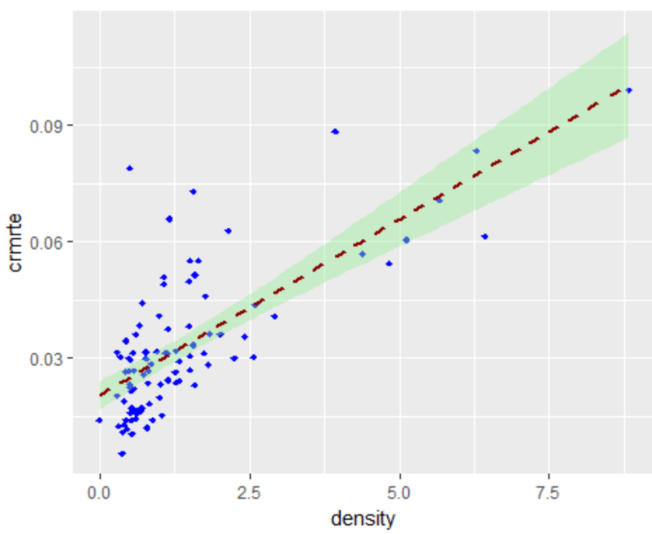
```
hist(crime$density_sqrt)
hist(crime$density_inv)
```



Comparing before and after the transformation

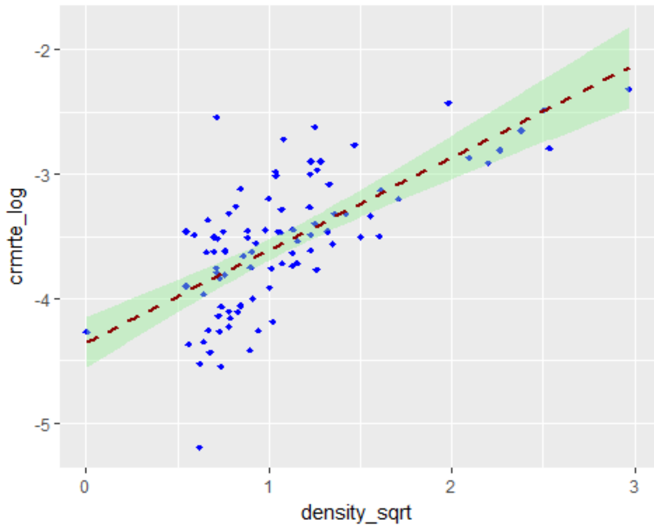
Hide

```
# Before
par(mfrow=c(2,1))
ggplot(crime, aes(x=density, y=crmrte)) +
  geom_point(shape=18, color="blue")+
  geom_smooth(method=lm,  linetype="dashed",
             color="darkred", fill="lightgreen")
```



Hide

```
# After
ggplot(crime, aes(x=density_sqrt, y=crmrte_log)) +
  geom_point(shape=18, color="blue")+
  geom_smooth(method=lm,  linetype="dashed",
             color="darkred", fill="lightgreen")
```



Here we can see a square root transform yields the best result for the density variable

Next, lets tackle the Police per capita.

Hide

```
summary(crime$polpc)
```

```
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max. 
0.0007459 0.0012308 0.0014853 0.0017022 0.0018768 0.0090543 
```

Hide

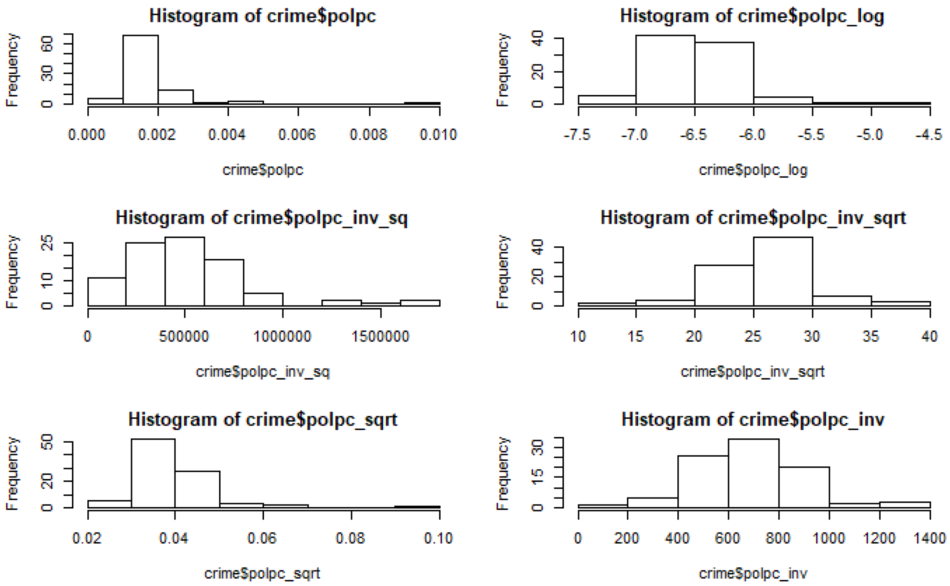
```
par(mfrow=c(3,2))
hist(crime$polpc)
crime$polpc_log = log(crime$polpc)
crime$polpc_inv_sq = 1/(crime$polpc^2)
crime$polpc_inv_sqrt = 1/(sqrt(crime$polpc))
crime$polpc_sqrt = sqrt(crime$polpc)
crime$polpc_inv = 1/crime$polpc
hist(crime$polpc_log)
```

Hide

```
hist(crime$polpc_inv_sq)
hist(crime$polpc_inv_sqrt)
```

Hide

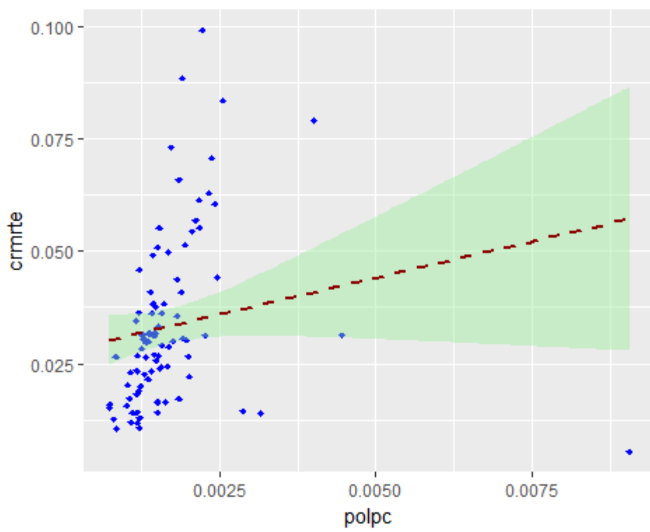
```
hist(crime$polpc_sqrt)
hist(crime$polpc_inv)
```



Comparing before and after the transformation

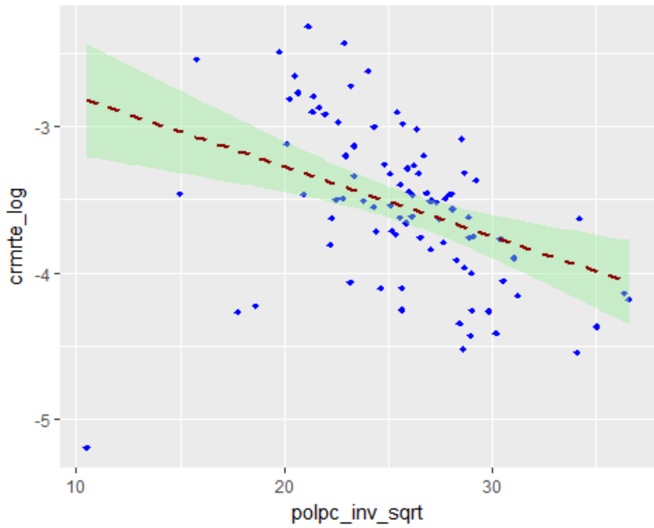
Hide

```
# Before
par(mfrow=c(2,1))
ggplot(crime, aes(x=polpc, y=crmrte)) +
  geom_point(shape=18, color="blue")+
  geom_smooth(method=lm,  linetype="dashed",
             color="darkred", fill="lightgreen")
```



Hide

```
# After
ggplot(crime, aes(x=polpc_inv_sqrt, y=crmrte_log)) +
  geom_point(shape=18, color="blue")+
  geom_smooth(method=lm,  linetype="dashed",
             color="darkred", fill="lightgreen")
```



Here we can see an inverse square root provides the best result for the polpc variable.

Next, lets tackle prbarr

Hide

```
summary(crime$prbarr)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
0.09277 0.20568 0.27095 0.29492 0.34438 1.09091 
```

Hide

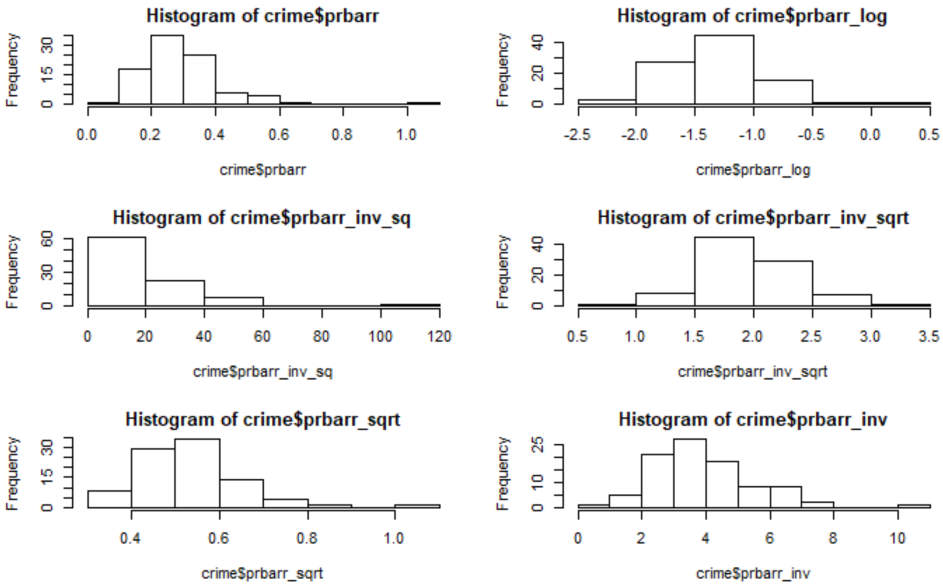
```
par(mfrow=c(3,2))
hist(crime$prbarr)
crime$prbarr_log = log(crime$prbarr)
crime$prbarr_inv_sq = 1/(crime$prbarr^2)
crime$prbarr_inv_sqrt = 1/(sqrt(crime$prbarr))
crime$prbarr_sqrt = sqrt(crime$prbarr)
crime$prbarr_inv = 1/crime$prbarr
hist(crime$prbarr_log)
```

Hide

```
hist(crime$prbarr_inv_sq)
hist(crime$prbarr_inv_sqrt)
```

Hide

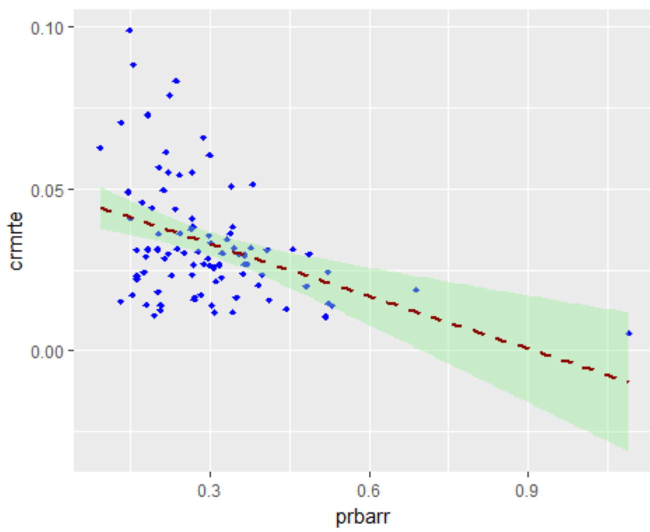
```
hist(crime$prbarr_sqrt)
hist(crime$prbarr_inv)
```



Comparing before and after the transformation

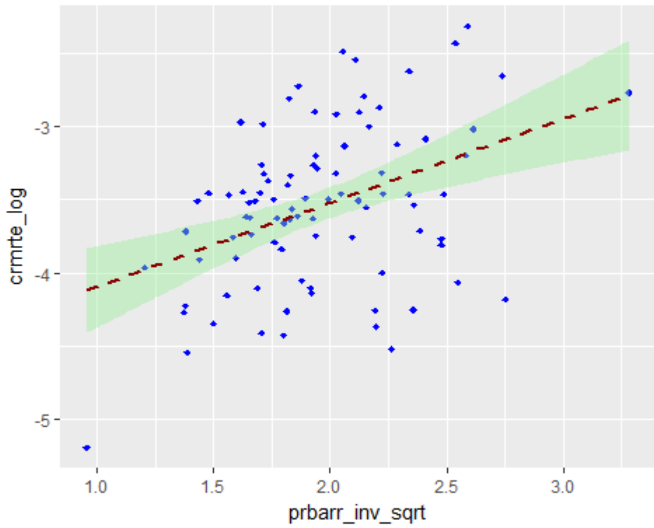
Hide

```
# Before
par(mfrow=c(2,1))
ggplot(crime, aes(x=prbarr, y=crmrte)) +
  geom_point(shape=18, color="blue")+
  geom_smooth(method=lm,  linetype="dashed",
             color="darkred", fill="lightgreen")
```



Hide

```
# After
ggplot(crime, aes(x=prbarr_inv_sqrt, y=crmrte_log)) +
  geom_point(shape=18, color="blue")+
  geom_smooth(method=lm,  linetype="dashed",
             color="darkred", fill="lightgreen")
```



Here we can see an inverse square root provides the best result.

Lets now tackle the mix variable

Hide

```
summary(crime$mix)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
0.01961 0.08073 0.10186 0.12884 0.15175 0.46512 
```

Hide

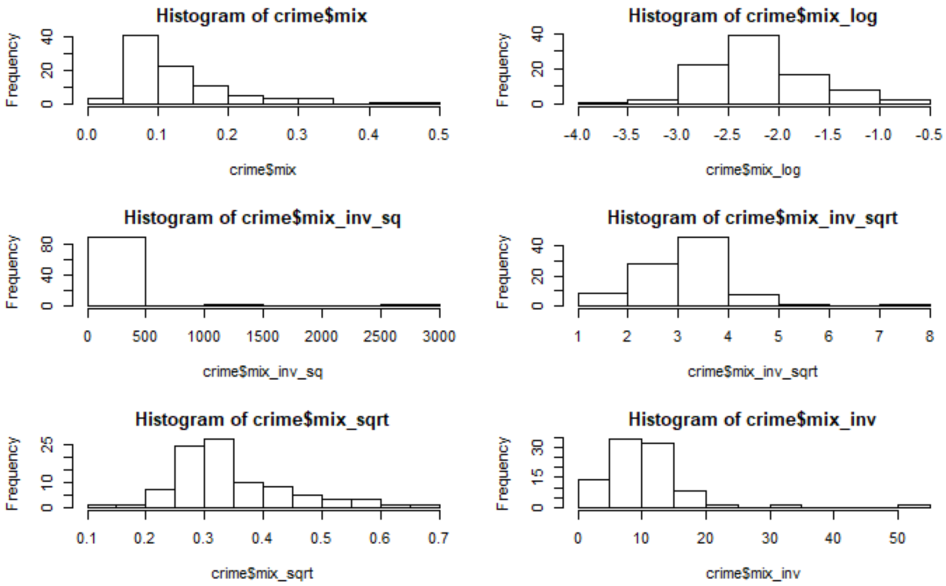
```
par(mfrow=c(3,2))
hist(crime$mix)
crime$mix_log = log(crime$mix)
crime$mix_inv_sq = 1/(crime$mix^2)
crime$mix_inv_sqrt = 1/(sqrt(crime$mix))
crime$mix_sqrt = sqrt(crime$mix)
crime$mix_inv = 1/crime$mix
hist(crime$mix_log)
```

Hide

```
hist(crime$mix_inv_sq)
hist(crime$mix_inv_sqrt)
```

Hide

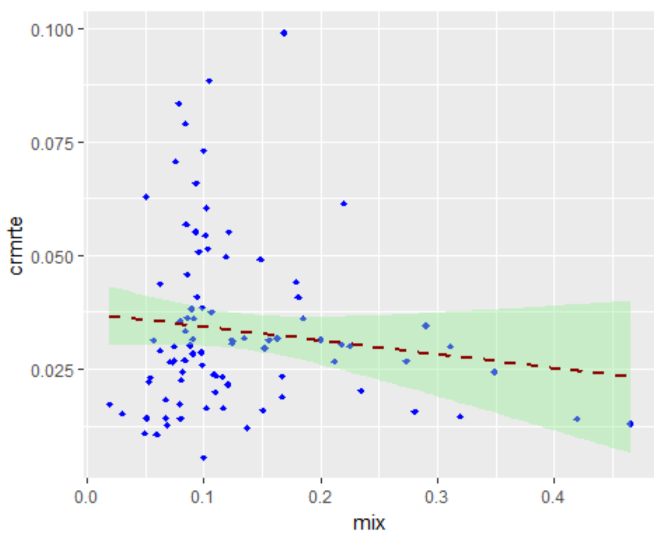
```
hist(crime$mix_sqrt)
hist(crime$mix_inv)
```



Comparing before and after the transformation

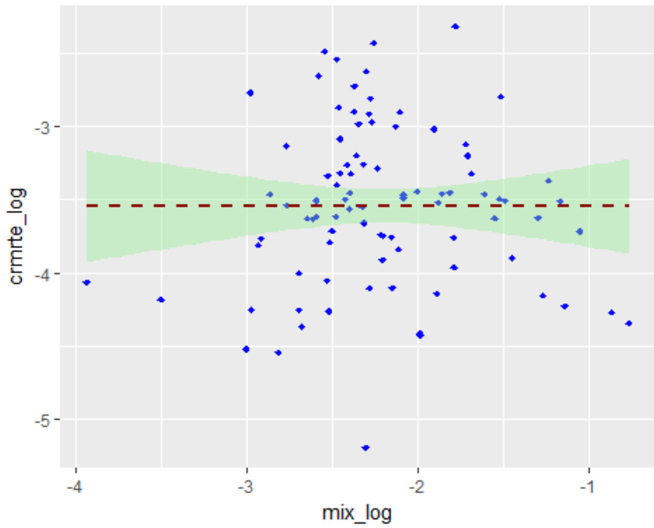
Hide

```
# Before
par(mfrow=c(2,1))
ggplot(crime, aes(x=mix, y=crmrte)) +
  geom_point(shape=18, color="blue")+
  geom_smooth(method=lm,  linetype="dashed",
           color="darkred", fill="lightgreen")
```



Hide

```
# After
ggplot(crime, aes(x=mix_log, y=crmrte_log)) +
  geom_point(shape=18, color="blue")+
  geom_smooth(method=lm,  linetype="dashed",
           color="darkred", fill="lightgreen")
```



The log transform provides the best result

Finally lets tackle the pctymle variable

Hide

```
summary(crime$pctymle)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
0.06216 0.07443 0.07771 0.08396 0.08350 0.24871 
```

Hide

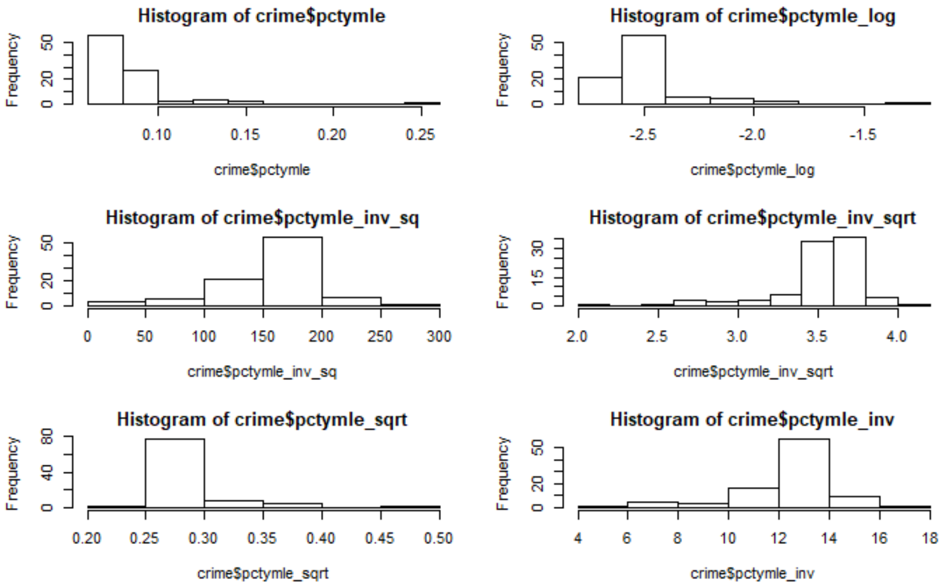
```
par(mfrow=c(3,2))
hist(crime$pctymle)
crime$pctymle_log = log(crime$pctymle)
crime$pctymle_inv_sq = 1/(crime$pctymle^2)
crime$pctymle_inv_sqrt = 1/(sqrt(crime$pctymle))
crime$pctymle_sqrt = sqrt(crime$pctymle)
crime$pctymle_inv = 1/crime$pctymle
hist(crime$pctymle_log)
```

Hide

```
hist(crime$pctymle_inv_sq)
hist(crime$pctymle_inv_sqrt)
```

Hide

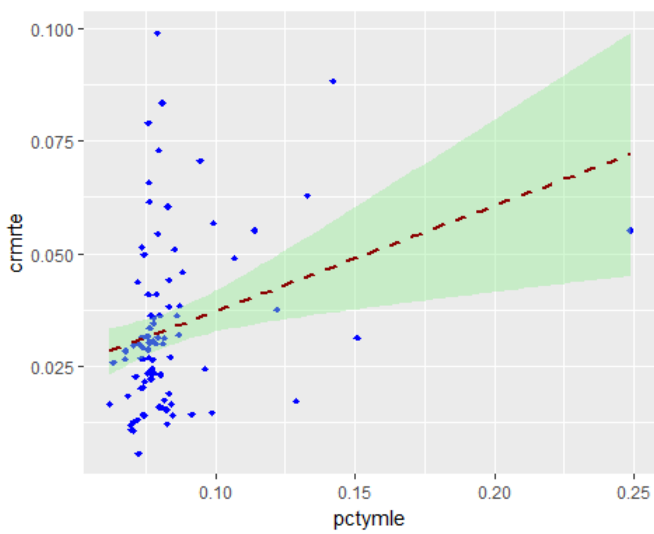
```
hist(crime$pctymle_sqrt)
hist(crime$pctymle_inv)
```



Comparing before and after the transformation

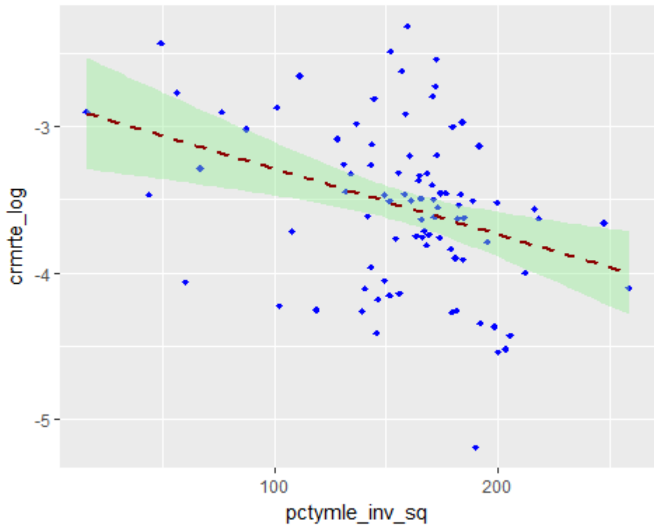
Hide

```
# Before
par(mfrow=c(2,1))
ggplot(crime, aes(x=pctymle, y=crmrte)) +
  geom_point(shape=18, color="blue")+
  geom_smooth(method=lm,  linetype="dashed",
           color="darkred", fill="lightgreen")
```



Hide

```
# After
ggplot(crime, aes(x=pctymle_inv_sq, y=crmrte_log)) +
  geom_point(shape=18, color="blue")+
  geom_smooth(method=lm,  linetype="dashed",
           color="darkred", fill="lightgreen")
```



Quickly reviewing our analysis so far, we have determined the following variable and transforms are applicable to handle any skew found in the data set provided.

We also note one or possibly more data points that are influencing the regression lines and these will need further investigation.

So, to conclude we have identified the following variable transforms

crime$crmrte_logcrime$taxpc_inv_sq crime$polpc_inv_sqrtcrime$density_sqrt crime$prbarr_inv_sqrtcrime$mix_log crime$pctymle_inv_sq

Next lets look at our wages and possible transformations.

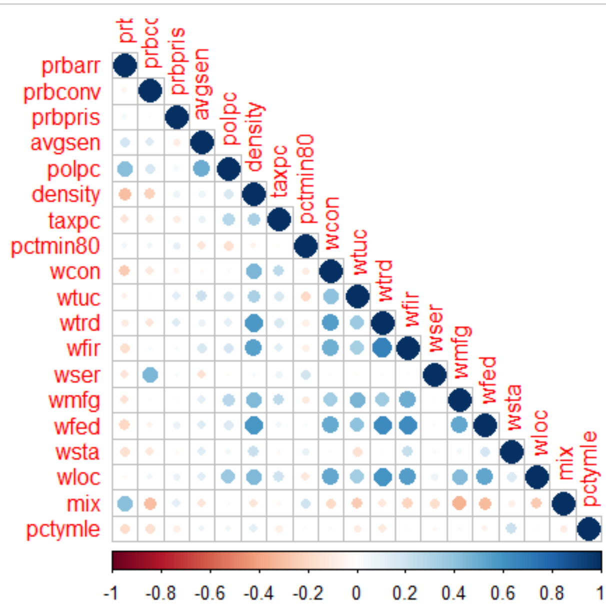
Hide

```
crime$avg_wage = (crime$wcon + crime$wtuc + crime$wtrd + crime$wfir + crime$wser +
                    crime$wmfg + crime$wfed + crime$wsta + crime$wloc)/9
crime$low_wage = ifelse(crime$wcon < quantile(crime$wcon, .1) | crime$wtrd < quantile(crime$wtrd, .1)
                          | crime$wfir < quantile(crime$wfir, .1) | crime$wser < quantile(crime$wser, .1)
                          | crime$wmfg < quantile(crime$wmfg, .1) | crime$wfed < quantile(crime$wfed, .1)
                          | crime$wsta < quantile(crime$wsta, .1) | crime$wloc < quantile(crime$wloc, .1),1,0)
crime$low_wage2 = ifelse(crime$wcon < quantile(crime$wcon, .4) & crime$wtrd < quantile(crime$wtrd, .4)
                          & crime$wfir < quantile(crime$wfir, .4) & crime$wser < quantile(crime$wser, .4)
                          & crime$wmfg < quantile(crime$wmfg, .4) & crime$wfed < quantile(crime$wfed, .4)
                          & crime$wsta < quantile(crime$wsta, .4) & crime$wloc < quantile(crime$wloc, .4) ,1,0)
```

Lets also review the correlation between the variables

Hide

```
c = cor(crime[ ,c("prbarr","prbconv","prbpris","avgsen","polpc","density","taxpc","pctmin80","wcon","wtuc","wtrd","wfir","ws
er","wmfg","wfed","wsta","wloc","mix","pctymle")])
corrplot(c,type='lower')
```



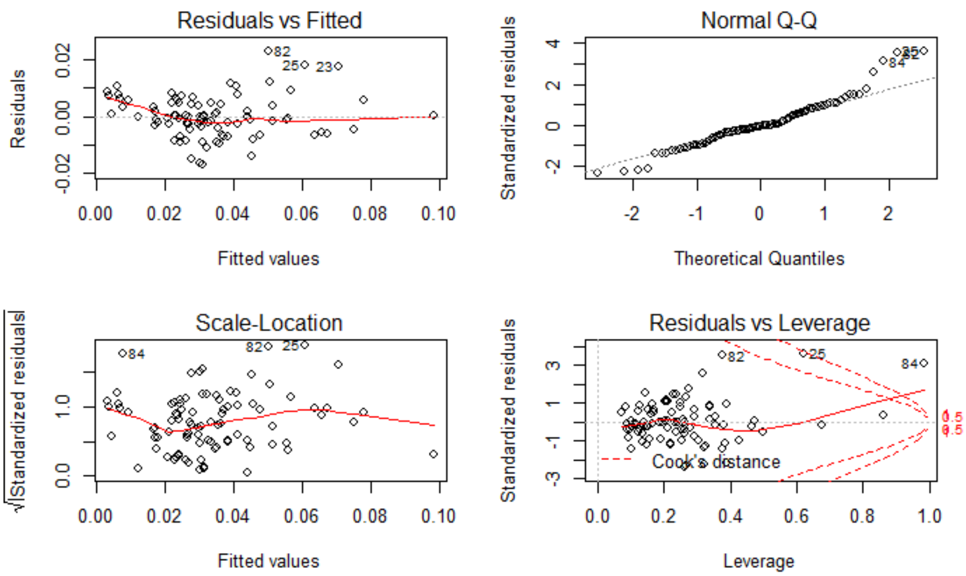
3. Model Specifications

To fit our models we first started with a model that included all variables to see, when controlling for everything, what factors still seemed to matter. This can be found in “model_all”. Additionally we used the ols_step_forward_p function to find the model of best fit, this is displayed as “model2”. Finally we wanted to find the most parsimonious model that only includes the essential factors. This can be found in “model3”. After creating our models, we test if assumptions for zero conditional mean and homoscedasticity hold for our models. We find for all three models that we can’t say our model violates the zero conditional mean assumption, but errors for all three are heteroscedastic. For our results printout in section 5, we use heteroscedastic robust results.

Model all

Hide

```
model_all = 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 = crime)
par(mfrow=c(2,2))
plot(model_all)
```



Hide

```
# we don't seem to violate zero conditional mean
shapiro.test(model_all$residuals)
```

```

    Shapiro-Wilk normality test

data:  model_all$residuals
W = 0.97991, p-value = 0.1727
```

Hide

```
# we do have heteroskdacity
bptest(model_all)
```

```

    studentized Breusch-Pagan test

data:  model_all
BP = 35.088, df = 22, p-value = 0.03793
```

Hide

```
summary(model_all)
```

```
Call:
lm(formula = crmrte ~ prbarr + prbconv + prbpris + avgsen + polpc +
    density + taxpc + west + central + urban + pctmin80 + wcon +
    wtuc + wtrd + wfir + wser + wmfg + wfed + wsta + wloc + mix +
    pctymle, data = crime)

Residuals:
      Min        1Q    Median        3Q       Max
-0.016815 -0.003860 -0.000455  0.004558  0.022847

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.369e-02  1.937e-02   0.707 0.482068
prbarr      -5.150e-02  9.779e-03  -5.266 1.54e-06 ***
prbconv     -1.863e-02  3.740e-03  -4.980 4.60e-06 ***
prbpris      3.127e-03  1.187e-02   0.263 0.793017
avgsen      -4.045e-04  4.073e-04  -0.993 0.324176
polpc        6.966e+00  1.529e+00   4.555 2.23e-05 ***
density      5.317e-03  1.338e-03   3.973 0.000174 ***
taxpc        1.624e-04  9.513e-05   1.707 0.092385 .
west1       -2.550e-03  3.796e-03  -0.672 0.504001
central1    -4.257e-03  2.749e-03  -1.549 0.126043
urban1      -6.068e-05  6.090e-03  -0.010 0.992079
pctmin80     3.251e-04  9.189e-05   3.538 0.000732 ***
wcon         2.274e-05  2.738e-05   0.831 0.408989
wtuc         6.350e-06  1.494e-05   0.425 0.672203
wtrd         2.938e-05  4.511e-05   0.651 0.516962
wfir        -3.543e-05  2.694e-05  -1.315 0.192951
wser        -1.718e-06  5.635e-06  -0.305 0.761458
wmfg        -9.109e-06  1.406e-05  -0.648 0.519140
wfed         2.916e-05  2.537e-05   1.150 0.254320
wsta        -2.229e-05  2.577e-05  -0.865 0.390003
wloc         1.492e-05  4.782e-05   0.312 0.755985
mix         -1.867e-02  1.457e-02  -1.281 0.204513
pctymle      1.014e-01  4.498e-02   2.255 0.027370 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.008241 on 68 degrees of freedom
Multiple R-squared:  0.855,	Adjusted R-squared:  0.8081
F-statistic: 18.23 on 22 and 68 DF,  p-value: < 2.2e-16
```

Assess the residuals

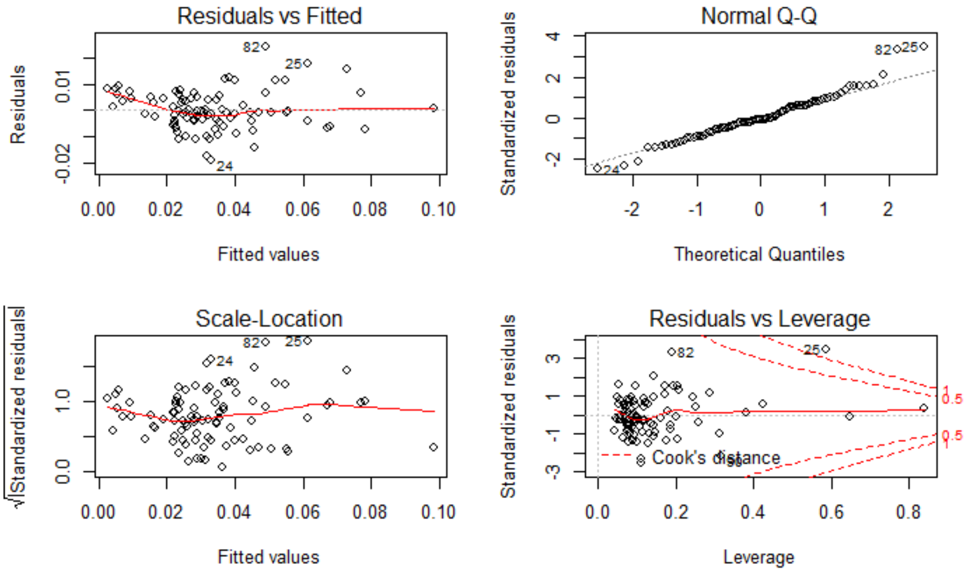
Hide

```
par(mfrow=c(1,1))
hist(model$residuals)
```

Model 2

Hide

```
model2 = lm(crmrte ~ prbarr + prbconv + polpc + density + taxpc + pctmin80 + west + central + wcon + wsta + taxpc +
              mix + pctymle, data = crime)
par(mfrow=c(2,2))
plot(model2)
```



Hide

```
# we don't seem to violate zero conditional mean
shapiro.test(model2$residuals)
```

```

    Shapiro-Wilk normality test

data:  model2$residuals
W = 0.98744, p-value = 0.5369
```

Hide

```
# we do have heteroskdacity
bptest(model2)
```

```

    studentized Breusch-Pagan test

data:  model2
BP = 22.501, df = 12, p-value = 0.03227
```

Hide

```
summary(model2)
```

```
Call:
lm(formula = crmrte ~ prbarr + prbconv + polpc + density + taxpc + 
    pctmin80 + west + central + wcon + wsta + taxpc + mix + pctymle, 
    data = crime)

Residuals:
       Min         1Q     Median         3Q        Max 
-0.0190321 -0.0042371 -0.0007563  0.0046459  0.0241156 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.942e-02  1.265e-02   2.325   0.0227 *  
prbarr      -5.290e-02  9.193e-03  -5.754 1.63e-07 ***
prbconv     -2.090e-02  3.009e-03  -6.944 1.01e-09 ***
polpc        6.952e+00  1.241e+00   5.601 3.07e-07 ***
density      5.521e-03  7.495e-04   7.366 1.57e-10 ***
taxpc        1.162e-04  8.078e-05   1.439   0.1542    
pctmin80     3.426e-04  7.931e-05   4.319 4.56e-05 ***
west1       -2.872e-03  3.306e-03  -0.869   0.3876    
central1    -3.697e-03  2.356e-03  -1.569   0.1207    
wcon         3.088e-05  2.166e-05   1.426   0.1580    
wsta        -3.483e-05  2.138e-05  -1.629   0.1073    
mix         -2.078e-02  1.254e-02  -1.657   0.1015    
pctymle      7.920e-02  4.065e-02   1.948   0.0550 .  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.007964 on 78 degrees of freedom
Multiple R-squared:  0.8447,	Adjusted R-squared:  0.8208 
F-statistic: 35.35 on 12 and 78 DF,  p-value: < 2.2e-16
```

Assess the residuals

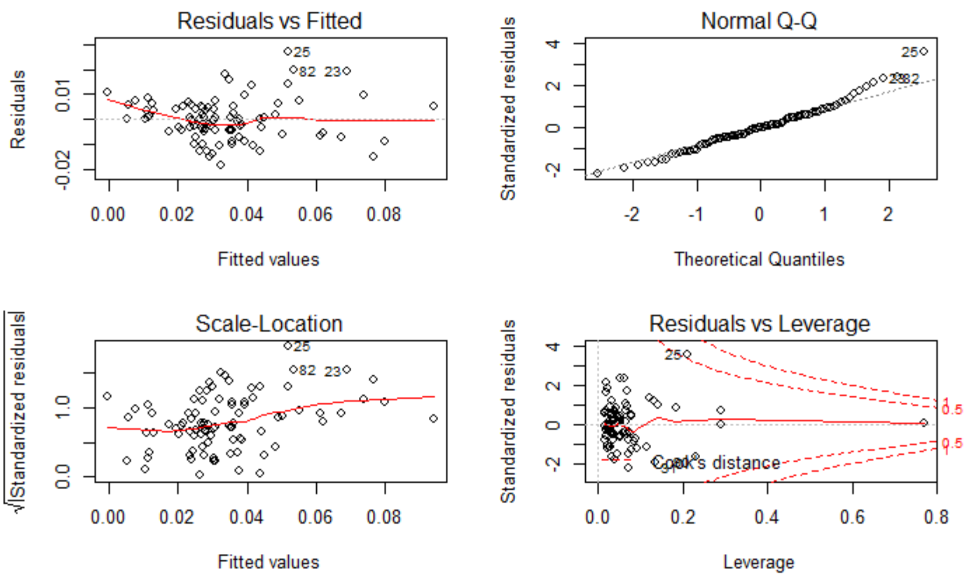
Hide

```
par(mfrow=c(1,1))
hist(model2$residuals)
```

Model 3

Hide

```
model3 = lm(crmrte ~ prbarr +
              prbconv +
              polpc +
              density +
              pctmin80,
            data = crime)
par(mfrow=c(2,2))
plot(model3)
```



Hide

```
# we don't seem to violate zero conditional mean
shapiro.test(model3$residuals)
```

```

	Shapiro-Wilk normality test

data:  model3$residuals
W = 0.98037, p-value = 0.1861
```

Hide

```
# we do have heteroskdacity
bptest(model3)
```

```

	studentized Breusch-Pagan test

data:  model3
BP = 22.944, df = 5, p-value = 0.000346
```

Hide

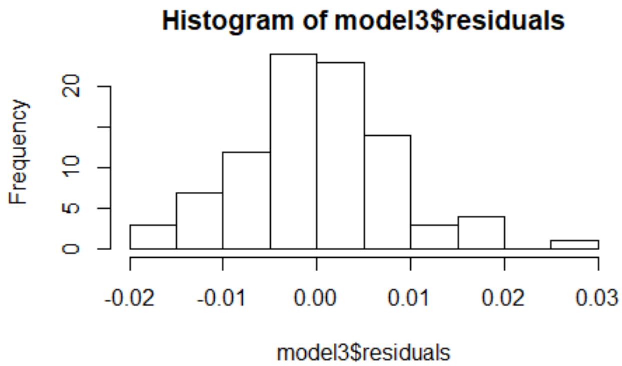
```
summary(model3)
```

```

Call:
lm(formula = crmrte ~ prbarr + prbconv + polpc + density + pctmin80,
    data = crime)

Residuals:
       Min         1Q     Median         3Q        Max
-0.0184202 -0.0046581 -0.0000045  0.0048587  0.0269763

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.357e-02  3.496e-03   9.602 3.26e-15 ***
prbarr      -6.594e-02  8.441e-03  -7.812 1.35e-11 ***
prbconv     -2.160e-02  2.827e-03  -7.643 2.95e-11 ***
polpc        8.111e+00  1.164e+00   6.971 6.34e-10 ***
density      5.551e-03  7.092e-04   7.828 1.25e-11 ***
pctmin80     3.705e-04  5.448e-05   6.800 1.37e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.008533 on 85 degrees of freedom
Multiple R-squared:  0.8056,	Adjusted R-squared:  0.7942
F-statistic: 70.47 on 5 and 85 DF,  p-value: < 2.2e-16
```

Assess the residuals

Hide

```
par(mfrow=c(1,1))
hist(model3$residuals)
```



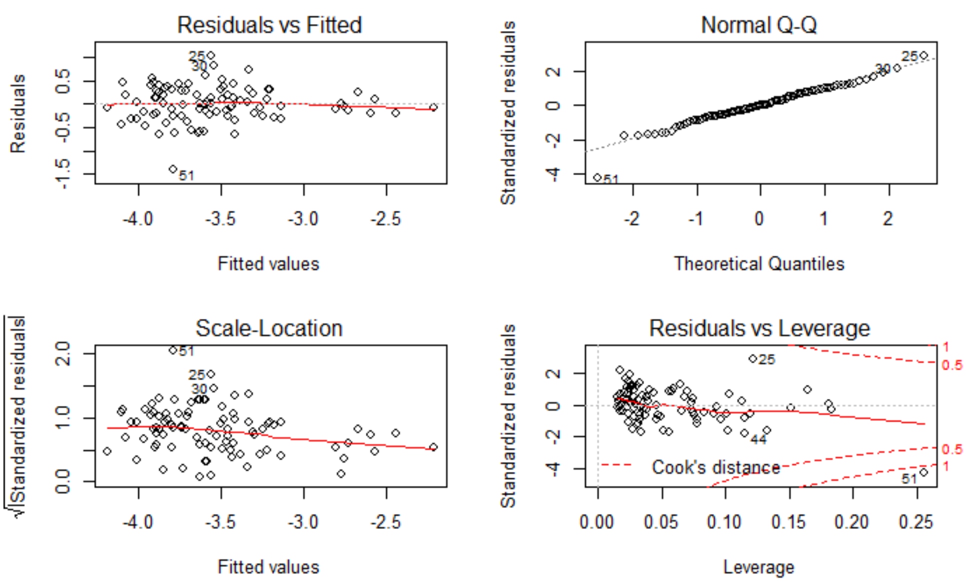
Model 3 (working variant of model 3 using transforms)

Simon's variant, reducing the number of variables and using variables that have been transformed. This will be merged with model 3 in the final paper.

crime$crmrte_{l}ogcrime$taxpc_inv_sq crime$polpc_{i}nv_{s}qrtcrime$density_sqrt crime$prbarr_{i}nv_{s}qrtcrime$mix_log crime$pctymle_inv_sq

Hide

```
model3b = lm(crmrte_log ~ density_sqrt + polpc_inv_sqrt + pctymle_inv_sq + prbarr_inv_sqrt, data = crime)
par(mfrow=c(2,2))
plot(model3b)
```



Hide

```
# we don't seem to violate zero conditional mean
shapiro.test(model3b$residuals)
```

```

	Shapiro-Wilk normality test

data:  model3b$residuals
W = 0.98052, p-value = 0.1907
```

Hide

```
# we do have heteroskdacity ?
bptest(model3b)
```

```

    studentized Breusch-Pagan test

data:  model3b
BP = 23.595, df = 4, p-value = 9.627e-05
```

Hide

```
summary(model3b)
```

```

Call:
lm(formula = crmrte_log ~ density_sqrt + polpc_inv_sqrt + pctymle_inv_sq +
    prbarr_inv_sqrt, data = crime)

Residuals:
     Min       1Q   Median       3Q      Max
-1.40398 -0.22795 -0.00429  0.26725  1.01989

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)     -3.868990   0.388457  -9.960 5.49e-16 ***
density_sqrt     0.585515   0.094157   6.219 1.75e-08 ***
polpc_inv_sqrt  -0.019518   0.010204  -1.913   0.0591 .
pctymle_inv_sq  -0.001552   0.001062  -1.462   0.1474
prbarr_inv_sqrt  0.221942   0.118649   1.871   0.0648 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3834 on 86 degrees of freedom
Multiple R-squared:  0.5291,    Adjusted R-squared:  0.5072
F-statistic: 24.16 on 4 and 86 DF,  p-value: 2.049e-13
```

Assess the residuals

Hide

```
par(mfrow=c(1,1))
hist(model3b$residuals)
```

4. Potential Omitted Variables

Before discussing our results, it’s important that discuss what omitted variables might be skewing our results. Given that crime is such a complex issue, it’s certain that we are missing quite a few key factors, but we will try to discuss the ones we found most important and their likely effects on our estimates.

Crime Type

In our estimation, this is likely the largest omitted variable. For some crimes, such as shoplifting the probability of arrest/conviction would likely have a huge impact on how likely an individual is to commit the crime. On the other hand, for crimes of passion like Voluntary manslaughter, an individual isn’t thinking about the results of their actions and their decision likely wouldn’t be as impacted by the chance of getting caught. Given that we don’t know the distribution of crime type for our data, it’s very hard to guess what sorts of impacts it would have on our estimates. ### Societal Values
While this one is a bit tough to nail down, and would be very hard to measure, but it’s clear that an individual’s personal moral compass would impact their choice to commit a crime. Given that this is so hard to measure, it’s unclear what variables are likely to be correlated with social norms that encourage/discourage crime, so we can guess as to the impact on our estimates. ### Poverty
In the United States minoritites have higher poverty rates than those of the majority. Given that poverty is also correlated with crime, we know that

our percent minority coefficient is likely overstating the impact of minorities on crime. We also know that impoverished communities also have a higher police presence. Given that heavy police presence is also correlated with crime, we know that our police per capita coefficient is overstating the impact of police on crime. ### Unemployment Rate
The effects of this omitted variable should be very similar to poverty. In the US minorities often have the highest unemployment rates (and higher police presence) and so our coefficients for minorities and police are likely overstated. ### Gang Presence/ubiquity
Normally seen in dense urban areas, so our density coefficient is likely overstated.
Drug Use
% of population living in government housing

5. Results & Recommendations

Hide

```
(se.model_all = sqrt(diag(vcovHC(model_all))))
```

```
 (Intercept)       prbarr      prbconv      prbpris       avgsen        polpc      density        taxpc        west1     cen
tral1
3.090444e-02 1.558134e-02 6.565246e-03 1.354780e-02 5.338566e-04 2.943982e+00 1.461619e-03 2.835321e-04 4.406283e-03 3.76417
6e-03
      urban1     pctmin80         wcon         wtuc         wtrd         wfir         wser         wmfg         wfed         
wsta
8.198884e-03 1.387495e-04 3.193561e-05 1.978770e-05 8.440095e-05 3.566720e-05 9.941475e-05 1.740497e-05 3.772863e-05 3.67772
1e-05
        wloc          mix      pctymle
8.553835e-05 2.279019e-02 4.763417e-02
```

Hide

```
(se.model2 = sqrt(diag(vcovHC(model2))))
```

```
 (Intercept)       prbarr      prbconv        polpc      density        taxpc     pctmin80        west1     central1         
wcon
1.821522e-02 1.314918e-02 4.835163e-03 1.963094e+00 1.454442e-03 2.525982e-04 1.092621e-04 3.656811e-03 3.101362e-03 2.58936
2e-05
        wsta          mix      pctymle
3.034674e-05 1.555509e-02 3.258885e-02
```

Hide

```
(se.model3 = sqrt(diag(vcovHC(model3))))
```

```
 (Intercept)       prbarr      prbconv        polpc      density     pctmin80
4.523438e-03 1.343829e-02 3.986377e-03 2.186813e+00 1.145439e-03 5.297248e-05
```

Hide

```
(se.model2b = sqrt(diag(vcovHC(model2b))))
```

```
   (Intercept)   density_sqrt polpc_inv_sqrt        pctymle
     0.8961692      0.1270408      0.0260127      1.4891938
```

Hide

```
# Display our models
stargazer(model_all, model2, model3, model2b, type = "text", omit.stat = "f",
          se = list(se.model_all, se.model2, se.model3,se.model2b),
          star.cutoffs = c(0.05, 0.01, 0.001))
```

```
number of rows of result is not a multiple of vector length (arg 2)number of rows of result is not a multiple of vector leng
th (arg 2)number of rows of result is not a multiple of vector length (arg 2)
```

	Dependent variable:			
	crmrte			crmrte_log
	(1)	(2)	(3)	(4)
prbarr	-0.051***	-0.053***	-0.066***	
	(0.016)	(0.013)	(0.013)	
prbconv	-0.019**	-0.021***	-0.022***	
	(0.007)	(0.005)	(0.004)	
prbpris	0.003			
	(0.014)			
avgsen	-0.0004			
	(0.001)			
polpc	6.966*	6.952***	8.111***	
	(2.944)	(1.963)	(2.187)	
density	0.005***	0.006***	0.006***	
	(0.001)	(0.001)	(0.001)	
taxpc	0.0002	0.0001		
	(0.0003)	(0.0003)		
west1	-0.003	-0.003		
	(0.004)	(0.004)		
central1	-0.004	-0.004		
	(0.004)	(0.003)		
urban1	-0.0001			
	(0.008)			
pctmin80	0.0003*	0.0003**	0.0004***	
	(0.0001)	(0.0001)	(0.0001)	
wcon	0.00002	0.00003		
	(0.00003)	(0.00003)		
wtuc	0.00001			
	(0.00002)			
wtrd	0.00003			
	(0.0001)			
wfir	-0.00004			
	(0.00004)			
wser	-0.00000			
	(0.0001)			
wmfg	-0.00001			
	(0.00002)			
wfed	0.00003			
	(0.00004)			
wsta	-0.00002	-0.00003		
	(0.00004)	(0.00003)		
wloc	0.00001			
	(0.0001)			
mix	-0.019	-0.021		
	(0.023)	(0.016)		
density_sqrt				0.662***
				(0.127)
polpc_inv_sqrt				-0.017
				(0.026)
pctymle	0.101*	0.079*		3.605*

```
                        (0.048)         (0.033)                         (1.489)

Constant                 0.014           0.029         0.034***        -4.117***
                        (0.031)         (0.018)         (0.005)         (0.896)

-----------------------------------------------------------------------------------
Observations              91              91              91              91
R2                       0.855           0.845           0.806           0.509
Adjusted R2              0.808           0.821           0.794           0.492
Residual Std. Error 0.008 (df = 68) 0.008 (df = 78) 0.009 (df = 85) 0.389 (df = 87)
===================================================================================
Note:                                                 *p<0.05; **p<0.01; ***p<0.001
```

From our results it's clear model 3 does a great job explaining the variance per captia crime, while only using a few variables. The main problem is that the coefficients for two variables are much lower in the other models. We will discuss our results by walking through each variable and providing policy recommendations.

First we look at the vairable that lose significance when controlling for other factors

Police Per Capita

This is one of the most interesting results. Our most simplistic model has the highest coefficient at 7.7 while the other two have a lower coefficient of ~7. Given how many factors from our omitted variable discussion are correlated with both crime and police presence we are confident that this coefficient is far to high. The fact that all else equal, more police would lead to more crime makes no sense, while the idea that there will be more police in locations with more crime is intuitive. In reality, we should expect that coefficient for Police Per Capita should be negative, and until we start to see this result we know that omitted variables are skewing our results. #### Percent Minority Without even looking at our results, given our OVB discussion, we should expect that simple models will overstate the impact of minorities. This is exactly what we see – model 3 has the highest coefficient. While it's likely racist and immoral to advocate for policies that decrease the percent of minorities in a community, it also wouldn't be an effective policy, because the impact is so small (coefficient of 0.0003), and this estimate is still likely overstated because of OVB.

Next we look at the vairable that remain significant when controlling for other factors

'Probability' of Arrest & 'Probability' of Conviction

These two factors provide the greatest policy recommendation. Given that even when controlling for all factors in our data set, these two factors have significant coefficients at ~0.05 and ~0.02. Given this significant impact, our policy recommendation is to invest in policies that can increase the rate of arrest and conviction. This could range from more and better camera systems in risk areas, to laws that provide immunity and protection to individuals who speak to police/testify about crimes. While this seems like a silver bullet, we should not the crime type was an important omitted variable. We can expect that these policies will have a large impact on some types of crime, but less on others.
Population Density
While all three models show some sort of relationship between density and crime, this is likely due to some sort of omitted variable(s) (such a gang presence). Additionally policies aimed at decreasing population density are unlikely to be ineffective, and are very likely to have unseen negative consequences.