Local Policy Recommendations for Crime Reduction

Final Report

Alexa Bagnard, Joseph Gaustad, Kevin Hartman, Francis Leung
(W203 Wednesday 6:30pm Summer 2019)

8/7/2019

Abstract

This is our study on crime. Crime does not pay. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Contents

1 Introduction

1.1 Background

In this report, we seek to examine and discuss determinants of crime and offer recommend actionable policy recommendations for local politicians running for election at the county level in North Carolina. For our analysis, we draw on sample data collected from a study by Cornwell and Trumball, researchers from the University of Georgia and West Virginia University. Our sample data includes data on crime rates, arrests, sentences, demographics, local weekly wages, tax revenues and more drawn from local and federal government data sources. Although the age of the data may be a potential limitation of our study, we believe the insights we gather and policy recommendations remain appropriate for local campaigns today.

Our primary question that will drive our data exploration are to ask which variables affect crime rate the most.

1.2 The Variables

The crime_v2 dataset provided includes 25 variables of interest.

We include them below for reference by category of interest.

Data Dictionary

Category	Variable
Crime Rate	crmrte
Geographic	county, west, central
Demographic	urban, density, pctmin80, pctymle
Economic - Wage	wcon, wtuc, wtrd, wfir, wser, wmfg, wfed, wsta, wloc
Economic - Revenue	taxpc
Law Enforcment	polpc, prbarr, prbconv, mix
Judicial/Sentencing	prbpris, avgsen
Time Period	year

Table 1: Data Dictionary

The variables above operationalize the conditions we wish to explore and their affects on crime rate

Chiefly, these break down as follows.

- The Economic variables measures the county's economic activity and health (e.g. opportunity to pursue legal forms of income). These variables come in the form of available wages and tax revenue returned to the county.
- The Law enforcment variables measures the county's ability to utilize law enforcment policy to deter crime. Similarly, the Judicial variables also signify impact of deterence to crime.
- The Demographic variables measure the cultural variability that represent the social differences between each county, such as urban vs rural and minority populations.
- The Geographic elements are categorical. They represent they ways in which the population is segmented by geography.

2 Exploratory Data Analysis (EDA)

2.1 Data Prep and Exploration

We begin our analysis by loading the data set and performing basic checks and inspections.

```
dfCrime = read.csv("crime_v2.csv")
```

First, we will remove the missing rows from the dataset.

```
nrow(dfCrime)
```

```
[1] 97
```

```
dfCrime <-na.omit(dfCrime) # omit the NA rows
nrow(dfCrime)
```

```
[1] 91
```

Next, we will inspect the data to see if there are duplicate records

```
dfCrime[duplicated(dfCrime),]
```

```
   county year    crmrte   prbarr     prbconv  prbpris avgsen      polpc
89    193   87 0.0235277 0.266055 0.588859022 0.423423   5.86 0.00117887
     density    taxpc west central urban pctmin80     wcon     wtuc
89 0.8138298 28.51783    1       0     0  5.93109 285.8289 480.1948
       wtrd     wfir     wser   wmfg   wfed   wsta   wloc       mix
89 268.3836 365.0196 295.9352 295.63 468.26 337.88 348.74 0.1105016
      pctymle
89 0.07819394
```

A duplicate row exists. We'll remove it.

```
dfCrime <- dfCrime[!duplicated(dfCrime),] # remove the duplicated row
```

We also saw that pbconv was coded as a level. It is not a level but a ratio. We'll change that now.

```
dfCrime$prbconv<-as.numeric(levels(dfCrime$prbconv))[dfCrime$prbconv]
```

We also notice by comparision of pctymle and pctmin80 one of the variables is off by a factor of 100. We will divide pctmin80 by 100 so the two variables are in the same unit terms.

```
dfCrime$pctmin80<-dfCrime$pctmin80/100
```

County was expressed as a number. However, it is a categorical variable and we will convert it to a factor instead.

```
dfCrime$county<-as.factor(dfCrime$county)
```

Next we inspect the indicator variables to see if they were coded correctly.

```
dfCrime %>% group_by(west, central) %>% tally()
```

```
# A tibble: 4 x 3
# Groups:   west [2]
   west central     n
  <int>   <int> <int>
1     0       0    35
2     0       1    33
3     1       0    21
4     1       1     1
```

```
dfCrime %>%
filter(west ==1 & central ==1)
```

```
  county year    crmrte   prbarr prbconv  prbpris avgsen      polpc
1     71   87 0.0544061 0.243119 0.22959 0.379175  11.29 0.00207028
   density    taxpc west central urban pctmin80     wcon     wtuc     wtrd
1 4.834734 31.53658    1       1     0  0.13315 291.4508 595.3719 240.3673
      wfir     wser   wmfg   wfed   wsta   wloc       mix    pctymle
1 348.0254 295.2301 358.95 509.43 359.11 339.58 0.1018608 0.07939028
```

One county was either mis-coded (with west=1 and central=1), or it truly belongs to both regions. However, this is very unlikely as the intended technique is to widen the data and introduce indicator variables for each category. It is not likley the data was captured for both categories.

We will need further analysis on this datapoint as it relates to the rest of the data.

For now, we will encode a new region variable and place the datapoint in its own category.

```
#Map central and west to a region code, and create a new category for other
# Note that county 71 has both western and central codes
dfCrime$region <- case_when (
            (dfCrime$central ==0 & dfCrime$west ==0) ~ 0, #Eastern, Coastal, Other
            (dfCrime$central ==0 & dfCrime$west ==1) ~ 1, #Western
            (dfCrime$central ==1 & dfCrime$west ==0) ~ 2, #Central
            (dfCrime$central ==1 & dfCrime$west ==1) ~ 3 #Central-Western county?
        )
dfCrime$regcode =
            factor( dfCrime$region , levels = 0:3 , labels =
                    c( 'Coastal',
                       'West',
                       'Central',
                       'CW')
                   )
```

We will also introduce an indicator variable for counties located in the "other" region that are not west or central

```
dfCrime$other <- ifelse((dfCrime$central ==0 & dfCrime$west ==0), 1, 0)
```

And we'll add an indicator variable to serve as complement to the urban indicator variable and call this 'nonurban'

```
dfCrime$nonurban <- ifelse((dfCrime$urban==0), 1, 0)
```

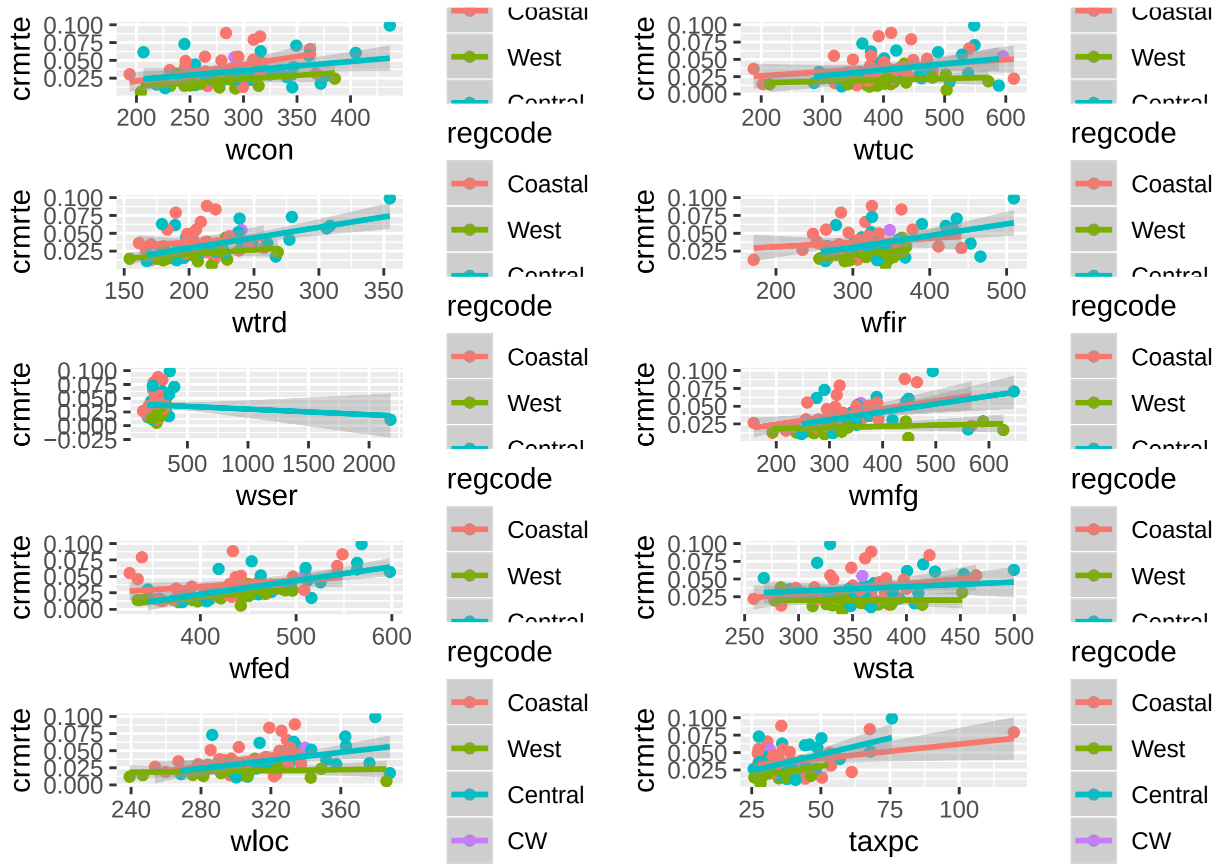
By way of the 1980 Census fact sheet, we discover the urban field is an encoding for SMSA (Standard Metropolitan Statistical Areas). https://www2.census.gov/prod2/decennial/documents/1980/1980censusofpopu8011uns_bw.pdf The value is one if the county is inside a metropolitan area. Otherwise, if the county is outisde a metropolitan area, the value is zero.

We create a metro factor variable to better describe this feature.

```
# create factor for SMSA (standard metropolitan statistical areas) with two levels
# (inside or outside)
#    https://www2.census.gov/prod2/decennial/documents/1980/1980censusofpopu8011uns_bw.pdf
dfCrime$metro =
            factor( dfCrime$urban , levels = 0:1 , labels =
                    c( 'Outside Metro',
                       'Inside Metro'
                      )
                   )
```

Next we will visualize each variable and its relationship to the variable crmrte through scatter plots

```
#Plot of the economic and tax related variables vs crmrte
q1<-ggplot(data = dfCrime, aes(x = wcon, y = crmrte, color = regcode)) +
      geom_point()+
  geom_smooth(method = "lm")
q2<-ggplot(data = dfCrime, aes(x = wtuc, y = crmrte, color = regcode)) +
      geom_point()+
  geom_smooth(method = "lm")
q3<-ggplot(data = dfCrime, aes(x = wtrd, y = crmrte, color = regcode)) +
      geom_point()+
  geom_smooth(method = "lm")
q4<-ggplot(data = dfCrime, aes(x = wfir, y = crmrte, color = regcode)) +
      geom_point()+
  geom_smooth(method = "lm")
q5<-ggplot(data = dfCrime, aes(x = wser, y = crmrte, color = regcode)) +
      geom_point()+
  geom_smooth(method = "lm")
q6<-ggplot(data = dfCrime, aes(x = wmfg, y = crmrte, color = regcode)) +
      geom_point()+
  geom_smooth(method = "lm")
q7<-ggplot(data = dfCrime, aes(x = wfed, y = crmrte, color = regcode)) +
      geom_point()+
  geom_smooth(method = "lm")
q8<-ggplot(data = dfCrime, aes(x = wsta, y = crmrte, color = regcode)) +
      geom_point()+
  geom_smooth(method = "lm")
q9<-ggplot(data = dfCrime, aes(x = wloc, y = crmrte, color = regcode)) +
      geom_point()+
  geom_smooth(method = "lm")
q10<-ggplot(data = dfCrime, aes(x = taxpc, y = crmrte, color = regcode)) +
      geom_point()+
  geom_smooth(method = "lm")
grid.arrange(q1, q2, q3, q4, q5, q6, q7, q8, q9, q10, ncol=2)
```



We observe a few data points of interest in the comparison above, notably, wser appears to have an extreme data point.

Other variables show outliers as well, but not as extreme. We will determine if any of these points have leverage or influence during model specification.

For now, lets dig deeper into one of the extreme outliers after our visual inspection.

```
dfCrime %>%
filter(wser > 2000) %>%
select(county, wser)
```

```
  county     wser
1    185 2177.068
```

This average service wage is much too high based on what we know about the 1980s and every other wage recorded in comparison. A review of the detailed population statistics describing mean wage per industry (table 231) confirms this. https://www2.census.gov/prod2/decennial/documents/1980/1980censusofpopu801352uns_bw.pdf

Outliers affect our ability to estimate statistics, resulting in overestimated or underestimated values. Outliers can be due to a number of different factors such as response errors and data entry errors. Outliers will introduce bias into our estimates and are addressed during the analysis phase. The mechanism for treatment include three approaches 1) trimming 2) winsorization or 3) imputation. Trimming will remove the rest of the values in the observation and is not an preferred treatment. Winsorazion relies on replacing outliers with the second largest or second smallest value excluding the outlier. Imputation methods can use the mean of a variable, or utilize regression models to predict the missing value. A number of packages are available in R that use the sample data to predict this value through regression. A full discussion on treatment methods can be found here: http://www.asasrms.org/Proceedings/y2004/files/Jsm2004-000559.pdf

We will use the Hmisc package which contains an impute function for treatment of this outlier

```
dfCrime$wser[which(dfCrime$county==185)]<-NA # set the value to NA so it will be imputed
```

```
impute_arg <- aregImpute(~ crmrte +  urban + central + west + other +
                          prbarr + prbconv + prbpris + avgsen + polpc +
                          density + taxpc + pctmin80 + wcon + wtuc +
                          wtrd + wfir + wser + wmfg + wfed + wsta + wloc +
                          mix + pctymle, data = dfCrime, match="weighted",
                          nk=3, B=10, n.impute = 100)
```

```
paste("R-squares for Predicting Non-Missing Values for Each Variable")
```

```
[1] "R-squares for Predicting Non-Missing Values for Each Variable"
```

```
impute_arg$rsq
```

```
     wser
0.8804895
```

```
paste("Distribution of Values for Each Imputation")
```

```
[1] "Distribution of Values for Each Imputation"
```

```
table(impute_arg$imputed$wser)
```

```

133.0430603 172.6280975 182.0196228 192.3076935 196.1453247 203.8864288
         10           1           2           1           1           1
204.3792114  206.281601 210.4414825 219.6342773 221.3903351 230.6580658
          1           2           1           1           1           1
239.2233429 243.4705658 251.4270172 253.6207123 274.1774597 292.7027283
          1           1           2           1          69           1
305.1542664 347.6608887
          1           1
```

We will reassign the value in our dataset to the mean from these trials.

```
dfCrime$wser[which(dfCrime$county==185)]<-mean(impute_arg$imputed$wser)
print("Newly Reassigned wser Value for County 185:")
```

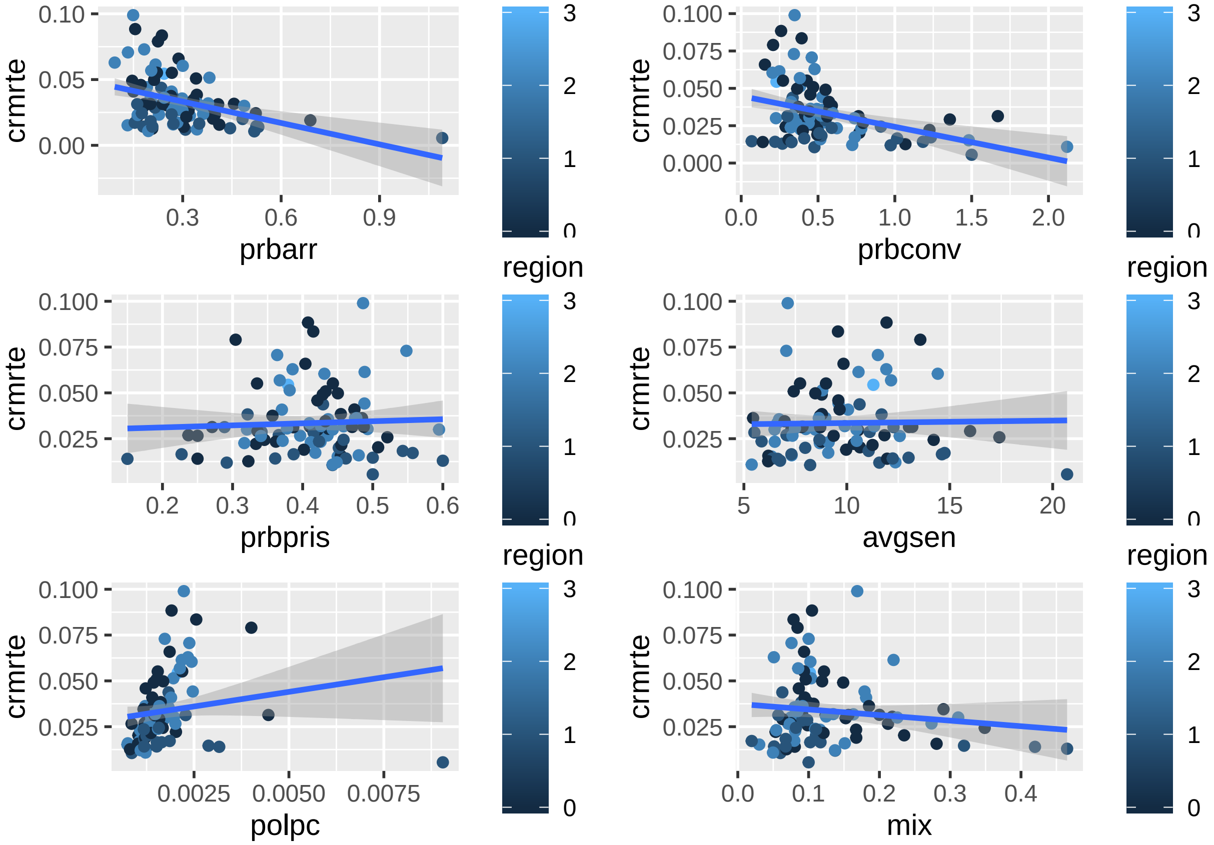
```
[1] "Newly Reassigned wser Value for County 185:"
```

```
dfCrime$wser[which(dfCrime$county==185)]
```

```
[1] 250.6144
```

Next, we will examine the criminal justice variables.

```
#Plot of the criminal justice and law enforcment related variables vs crmrte
q1<-ggplot(data = dfCrime, aes(x = prbarr, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q2<-ggplot(data = dfCrime, aes(x = prbconv, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q3<-ggplot(data = dfCrime, aes(x = prbpris, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q4<-ggplot(data = dfCrime, aes(x = avgsen, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q5<-ggplot(data = dfCrime, aes(x = polpc, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q6<-ggplot(data = dfCrime, aes(x = mix, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")

grid.arrange(q1, q2, q3, q4, q5, q6, ncol=2)
```

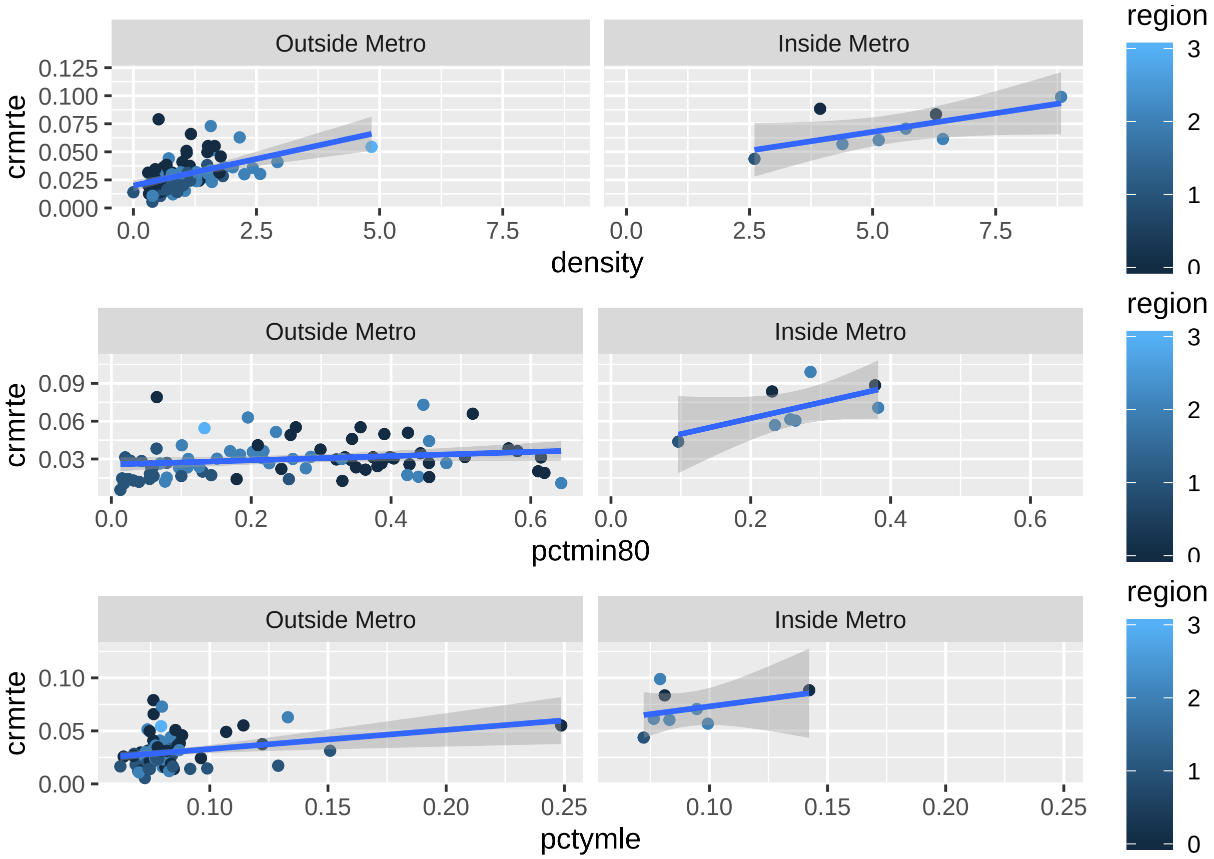


The criminal justice and law enforcement variables also show evidence of possible outliers, notably, pbarr and polpc appear to have extreme data points

We also see that prbarr and prbconv have values greater than 1. However, these are not true probabity numbers and are instead ratios used as a stand in for the true probability numbers.

There is a possibility of higher arrests per incident for an area. Meaning, the area has low incidents in general but when there were incidents there were also multiple arrests. The same case can be made for the convictions per arrest variable which we see is for a different region. In that county there may have been multiple charges brought per one arrest.

```
#plot of demographic information for counties Outside and Inside the metro areas
# population density, percent minority, percent young male

q1<-ggplot(data = dfCrime, aes(x = density, y = crmrte, color = region)) +
      geom_point() + facet_wrap(~ metro) +
  geom_smooth(method = "lm")
q2<-ggplot(data = dfCrime, aes(x = pctmin80, y = crmrte, color = region)) +
      geom_point() + facet_wrap(~ metro) +
  geom_smooth(method = "lm")
q3<-ggplot(data = dfCrime, aes(x = pctymle, y = crmrte, color = region)) +
      geom_point()+ facet_wrap(~ metro) +
  geom_smooth(method = "lm")

grid.arrange(q1, q2, q3, ncol=1)
```



Notably more outliers are observed in demographic information. Here, pctymle in one county outside of a metro area is nearly 25%. That seems quite high in normal statistical measures of the population, however, this can be explained as a county having a large college town population.

Finally, we can see our bright blue region 3 county and notice its population density. Its behavior is more similar to an inside metro area than outside. In addition to be coded for both western and central regions, it appears to be miscoded here as well.

We will address the metro variable, and examine whether the region variable should be west, central or other instead of both central and west

```
dfCrime %>%
filter(west ==1 & central ==1) %>%
select(county, west, central, other, urban, region, regcode, metro)
```

```
  county west central other urban region regcode         metro
1     71    1       1     0     0      3      CW Outside Metro
```

```
dfCrime$west[which(dfCrime$county==71)]<-NA
dfCrime$central[which(dfCrime$county==71)]<-NA
dfCrime$other[which(dfCrime$county==71)]<-NA
dfCrime$urban[which(dfCrime$county==71)]<-NA
```

```
impute_arg <- aregImpute(~ crmrte +  urban + central + west +
                         prbarr + prbconv + prbpris + avgsen + polpc +
                         density + taxpc + pctmin80 + wcon + wtuc +
                         wtrd + wfir + wser + wmfg + wfed + wsta + wloc +
                         mix + pctymle, data = dfCrime, match="weighted",
                         nk=3, B=10, n.impute = 100)
```

```
paste("R-squares for Predicting Non-Missing Values for Each Variable")
```

```
[1] "R-squares for Predicting Non-Missing Values for Each Variable"
```

```
impute_arg$rsq
```

```
    urban   central      west
0.9759023 0.8769296 0.9824131
```

```
paste("Distribution of Values for Each Imputation")
```

```
[1] "Distribution of Values for Each Imputation"
```

```
table(impute_arg$imputed$central)
```

```

 0  1
33 67
```

```
paste("Distribution of Values for Each Imputation")
```

```
[1] "Distribution of Values for Each Imputation"
```

```
table(impute_arg$imputed$west)
```

```

 0  1
78 22
```

```
paste("Distribution of Values for Each Imputation")
```

```
[1] "Distribution of Values for Each Imputation"
```

```
table(impute_arg$imputed$urban)
```

```

 0  1
 9 91
```

The results confirm the county is urban. It is also highly probable that county 71 is not west and most likely associated with central. After correcting our data for urban and west, let's compare 'central' with 'other' to be certain we have the right region.

```
#We need a mode function, so lets define one. Source - public domain
Mode = function(x){
    ta = table(x)
    tam = max(ta)
    if (all(ta == tam))
         mod = NA
    else
         if(is.numeric(x))
    mod = as.numeric(names(ta)[ta == tam])
    else
         mod = names(ta)[ta == tam]
    return(mod)
}

dfCrime$urban[which(dfCrime$county==71)]<-Mode(impute_arg$imputed$urban)
print("Newly Reassigned urban Value for County 71:")
```

```
[1] "Newly Reassigned urban Value for County 71:"
```

```
dfCrime$urban[which(dfCrime$county==71)]
```

```
[1] 1
```

```
dfCrime$nonurban[which(dfCrime$county==71)]<-1-Mode(impute_arg$imputed$urban)
print("Newly Reassigned nonurban Value for County 71:")
```

```
[1] "Newly Reassigned nonurban Value for County 71:"
```

```
dfCrime$nonurban[which(dfCrime$county==71)]
```

```
[1] 0
```

```
dfCrime$west[which(dfCrime$county==71)]<-Mode(impute_arg$imputed$west)
print("Newly Reassigned west Value for County 71:")
```

```
[1] "Newly Reassigned west Value for County 71:"
```

```
dfCrime$west[which(dfCrime$county==71)]
```

```
[1] 0
```

```
impute_arg <- aregImpute(~ crmrte + central + other +
                         prbarr + prbconv + prbpris + avgsen + polpc +
                         density + taxpc + pctmin80 + wcon + wtuc +
                         wtrd + wfir + wser + wmfg + wfed + wsta + wloc +
                         mix + pctymle, data = dfCrime, match="weighted",
                         nk=3, B=10, n.impute = 100)
```

```
paste("R-squares for Predicting Non-Missing Values for Each Variable")
```

```
[1] "R-squares for Predicting Non-Missing Values for Each Variable"
```

```
impute_arg$rsq
```

```
  central     other
0.9147736 0.9050807
```

```
paste("Distribution of Values for Each Imputation")
```

```
[1] "Distribution of Values for Each Imputation"
```

```
table(impute_arg$imputed$other)
```

```

 0  1
91  9
```

```
paste("Distribution of Values for Each Imputation")
```

```
[1] "Distribution of Values for Each Imputation"
```

```
table(impute_arg$imputed$central)
```

```

 0  1
24 76
```

We also show a strong likelihood of the county not being other. The case for central is high. Since the county is not western and not other it must be in central by default, and the Hmisc algorithm bolsters that suggestion. We'll assign our new values.

```
dfCrime$other[which(dfCrime$county==71)]<-Mode(impute_arg$imputed$other)
dfCrime$other[which(dfCrime$county==71)]
```

```
[1] 0
```

```
dfCrime$central[which(dfCrime$county==71)]<-1-Mode(impute_arg$imputed$other)
dfCrime$central[which(dfCrime$county==71)]
```

```
[1] 1
```

Recode the categories for region and metro

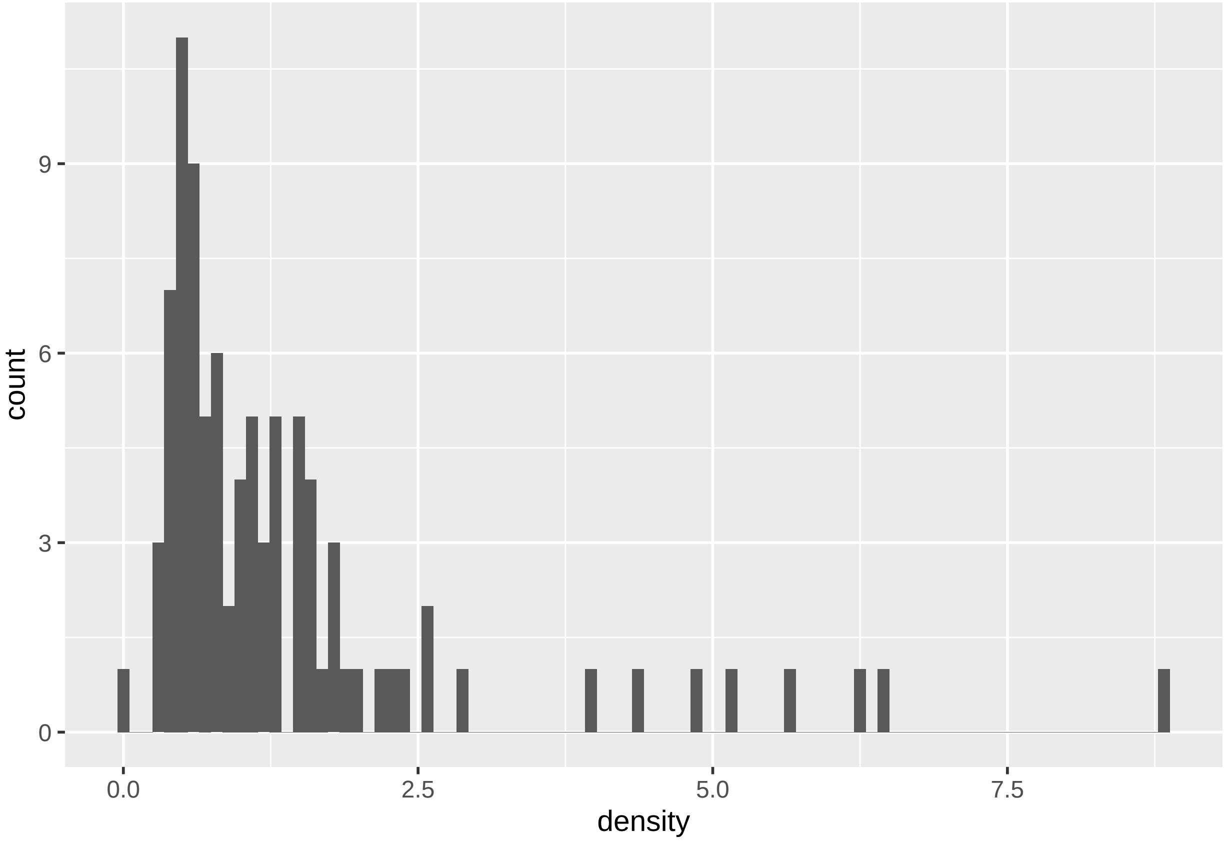
```
dfCrime$region <- case_when (
            (dfCrime$central ==0 & dfCrime$west ==0) ~ 0, #Eastern, Coastal, Other
            (dfCrime$central ==0 & dfCrime$west ==1) ~ 1, #Western
            (dfCrime$central ==1 & dfCrime$west ==0) ~ 2  #Central
        )
dfCrime$regcode =
            factor( dfCrime$region , levels = 0:2 , labels =
                    c( 'Coastal',
                       'West',
                       'Central' )
                   )

dfCrime$metro =
            factor( dfCrime$urban , levels = 0:1 , labels =
                    c( 'Outside Metro',
                       'Inside Metro'
                      )
                   )
```

```
dfCrime %>%
filter(county == 71) %>%
select(county, west, central, urban, region, regcode, metro)
```

```
  county west central urban region regcode        metro
1     71    0       1     1      2 Central Inside Metro
```

Let's review our density numbers again by looking in more detail at its distribution.

```
options(repr.plot.width=8, repr.plot.height=4)
ggplot(data = dfCrime, aes(x = density)) +
      geom_histogram(bins=90)
```



We note that one of the counties has an extremely low density. Near zero.

```
dfCrime %>%
filter(density < 0.01)
```

```
  county year    crmrte   prbarr  prbconv prbpris avgsen      polpc
1    173   87 0.0139937 0.530435 0.327869    0.15   6.64 0.00316379
      density     taxpc west central urban pctmin80     wcon     wtuc
1 2.03422e-05 37.72702    1       0     0 0.253914 231.696 213.6752
      wtrd     wfir     wser   wmfg   wfed   wsta   wloc       mix
1 175.1604 267.094 204.3792 193.01 334.44 414.68 304.32 0.4197531
     pctymle region regcode other nonurban         metro
1 0.07462687      1    West     0        1 Outside Metro
```

In review of the North Carolina county density data from 1985, the smallest population density in any county in North Carolina is 0.0952. This makes the density of 0.0000203422 (ie. average of ~2.0 people per 10,000 square miles) for county 173 statistically impossible. It is miscoded.

http://ncosbm.s3.amazonaws.com/s3fs-public/demog/dens7095.xls

(Note to team: We could use this table if we want to assign names to our counties by comparing the population densities. What is interesting is that the 6 rows of missing values we removed earlier can be found in the tail of this table. There was an arbitrary cut off after a certain density - lkely because the counties were not statistically significant. County 173 is not one of those counties, however, as our imputation process will demonstrate.)

```r
dfCrime$density[which(dfCrime$county==173)]<- NA
```

```r
#dfSubset <-  we will use the non-urban western counties
impute_arg <- aregImpute(~ crmrte +
                         prbarr + prbconv + prbpris + avgsen + polpc +
                         density + taxpc + pctmin80 + wcon + wtuc +
                         wtrd + wfir + wser + wmfg + wfed + wsta + wloc +
                         mix + pctymle, data = dfCrime %>% filter(urban==0 & west ==1),
                         match="weighted",  nk=3, B=10, n.impute = 30)
```

```r
paste("R-squares for Predicting Non-Missing Values for Each Variable")
```

```
[1] "R-squares for Predicting Non-Missing Values for Each Variable"
```

```r
impute_arg$rsq
```

```
density
      1
```

```r
paste("Distribution of Values for Each Imputation")
```

```
[1] "Distribution of Values for Each Imputation"
```

```r
table(impute_arg$imputed$density)
```

```

0.385809302  0.41276595 0.864864886 0.889880955 1.498938441 1.815508008
          1          20           2           1           2           4
```

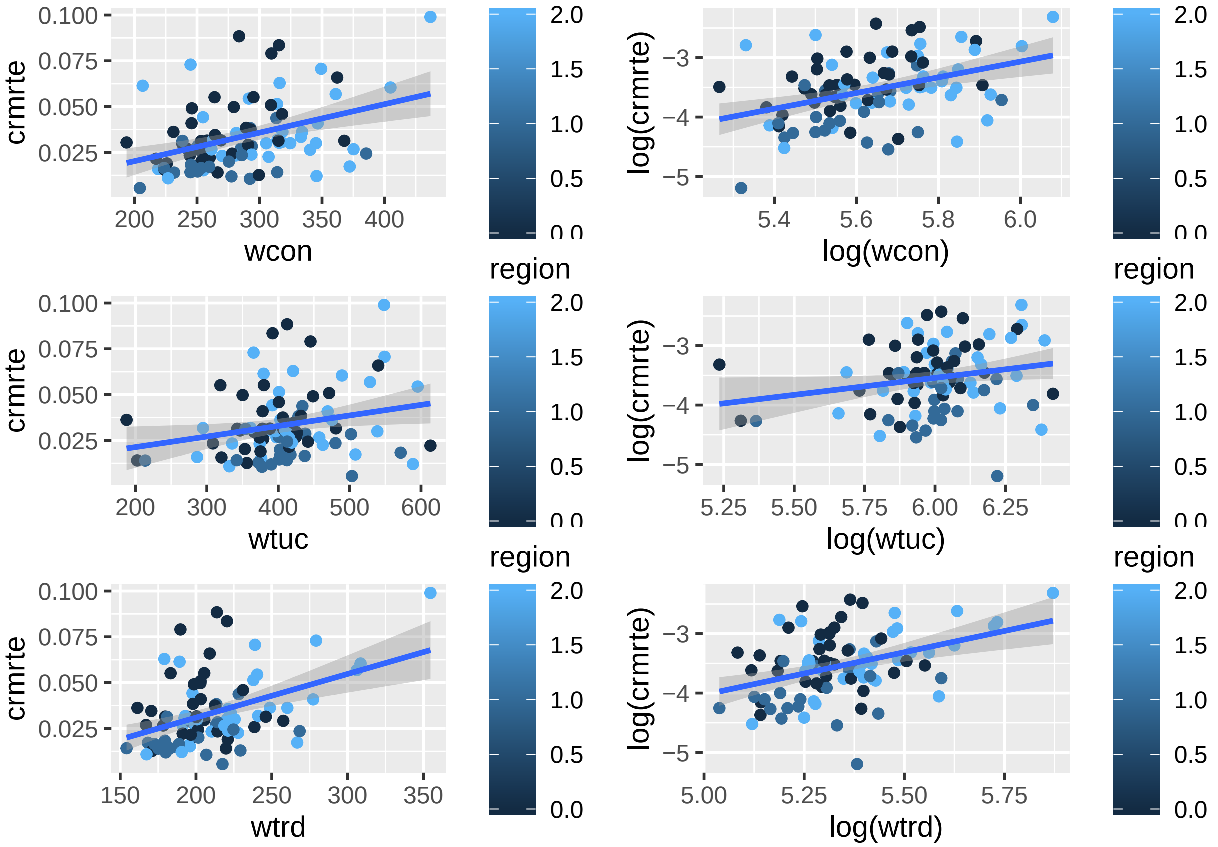
```r
dfCrime$density[which(dfCrime$county==173)]<-mean(impute_arg$imputed$density)
dfCrime$density[which(dfCrime$county==173)]
```

```
[1] 0.7173549
```

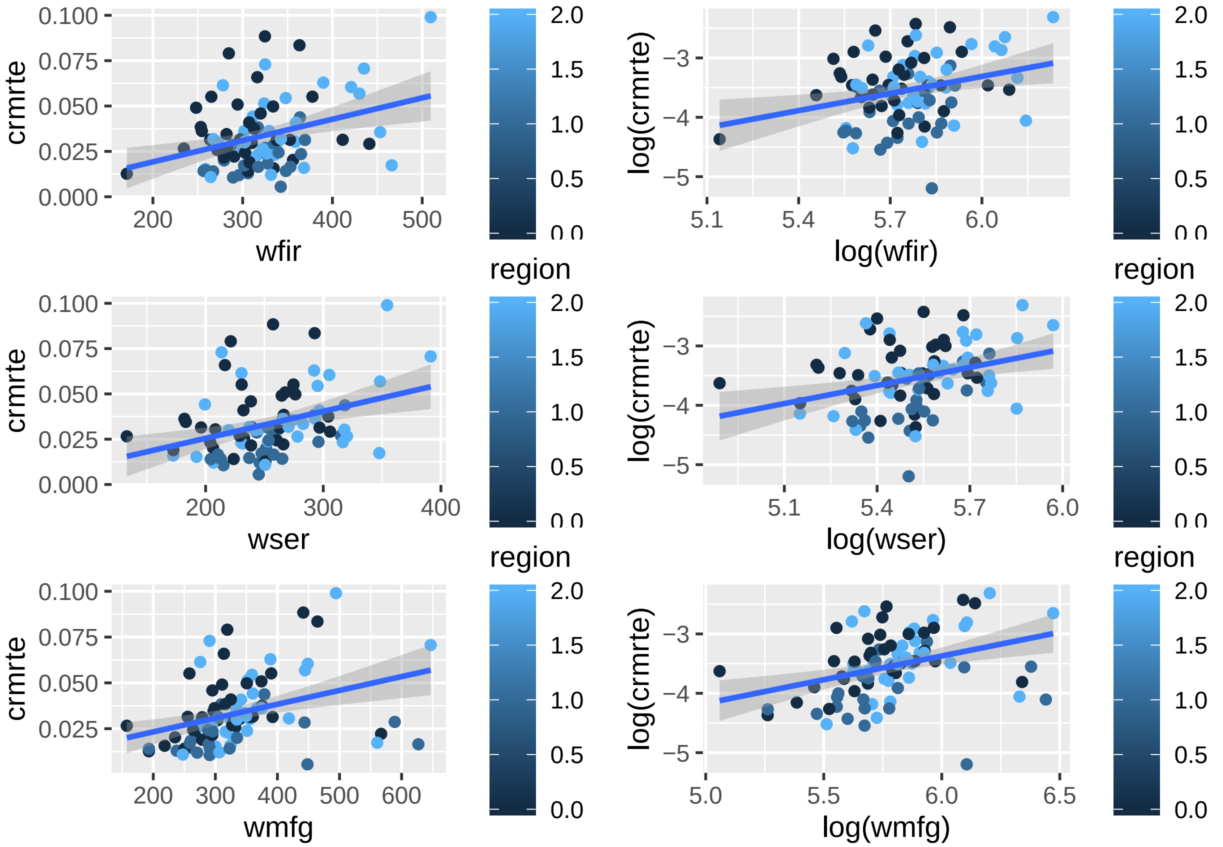
– MOVE TO END OF PUT IN EACH MODEL – Now, we will examine transforms for better linearity.

```r
#dfEconVars <- as.data.frame(cbind(dfCrime$wcon, dfCrime$wtuc, dfCrime$wtrd, dfCrime$wfir,
#                                  dfCrime$wser, dfCrime$wmfg, dfCrime$wfed, dfCrime$wsta,
#                                  dfCrime$wloc))
#names(dfEconVars) <- c('wcon', 'wtuc', 'wtrd', 'wfir', 'wser',
#                              'wmfg', 'wfed', 'wsta', 'wloc')
#
#ggplot(melt(dfEconVars),aes(x=value)) + geom_histogram(bins=30) + facet_wrap(~variable)

#The economic variables
q1<-ggplot(data = dfCrime, aes(x = wcon, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q1a<-ggplot(data = dfCrime, aes(x = log(wcon), y = log(crmrte), color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q2<-ggplot(data = dfCrime, aes(x = wtuc, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q2a<-ggplot(data = dfCrime, aes(x = log(wtuc), y = log(crmrte), color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q3<-ggplot(data = dfCrime, aes(x = wtrd, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
```

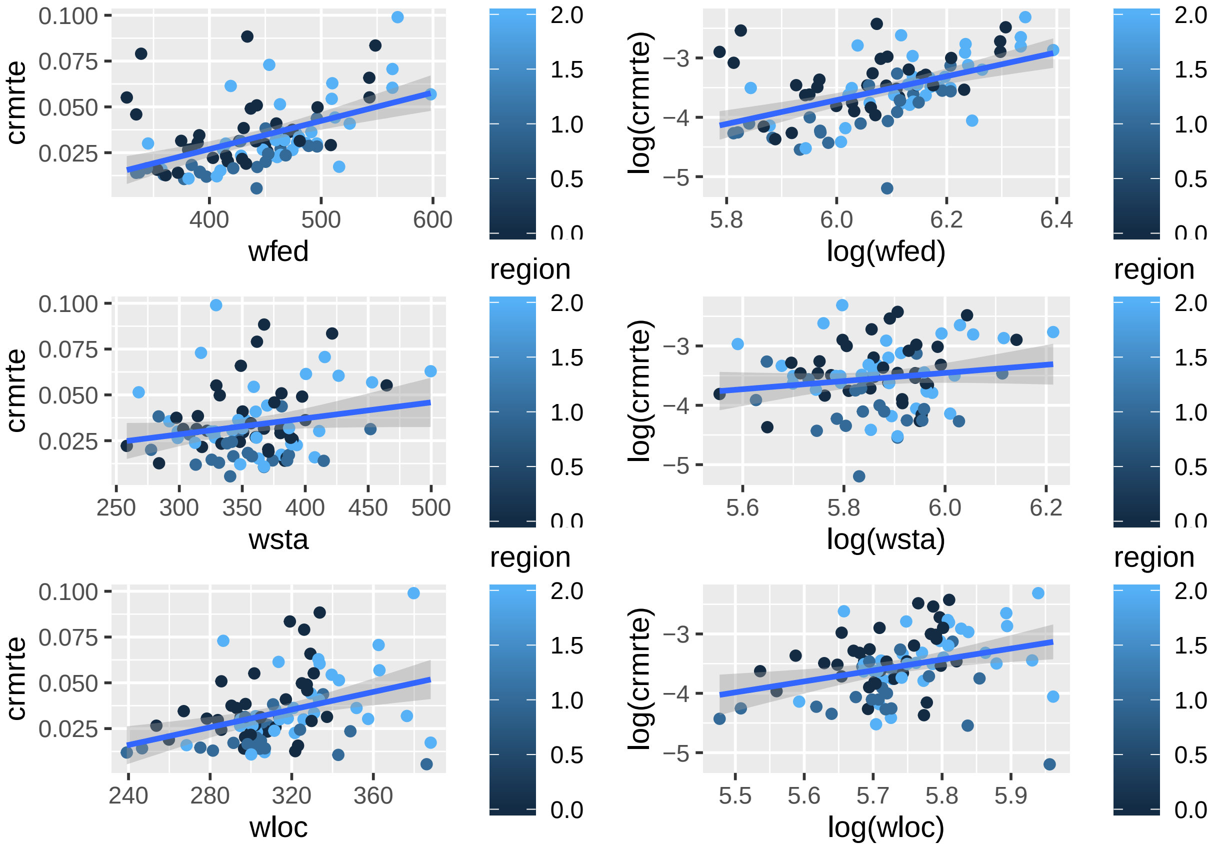
```
q3a<-ggplot(data = dfCrime, aes(x = log(wtrd), y = log(crmrte), color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q4<-ggplot(data = dfCrime, aes(x = wfir, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q4a<-ggplot(data = dfCrime, aes(x = log(wfir), y = log(crmrte), color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q5<-ggplot(data = dfCrime, aes(x = wser, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q5a<-ggplot(data = dfCrime, aes(x = log(wser), y = log(crmrte), color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q6<-ggplot(data = dfCrime, aes(x = wmfg, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q6a<-ggplot(data = dfCrime, aes(x = log(wmfg), y = log(crmrte), color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q7<-ggplot(data = dfCrime, aes(x = wfed, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q7a<-ggplot(data = dfCrime, aes(x = log(wfed), y = log(crmrte), color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q8<-ggplot(data = dfCrime, aes(x = wsta, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q8a<-ggplot(data = dfCrime, aes(x = log(wsta), y = log(crmrte), color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q9<-ggplot(data = dfCrime, aes(x = wloc, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q9a<-ggplot(data = dfCrime, aes(x = log(wloc), y = log(crmrte), color = region)) +
      geom_point()+
  geom_smooth(method = "lm")

options(repr.plot.width=8, repr.plot.height=16)
grid.arrange(q1, q1a, q2, q2a, q3, q3a, ncol=2)
```



```
grid.arrange(q4, q4a, q5, q5a, q6, q6a, ncol=2)
```



```r
grid.arrange(q7, q7a, q8, q8a, q9, q9a, ncol=2)
```

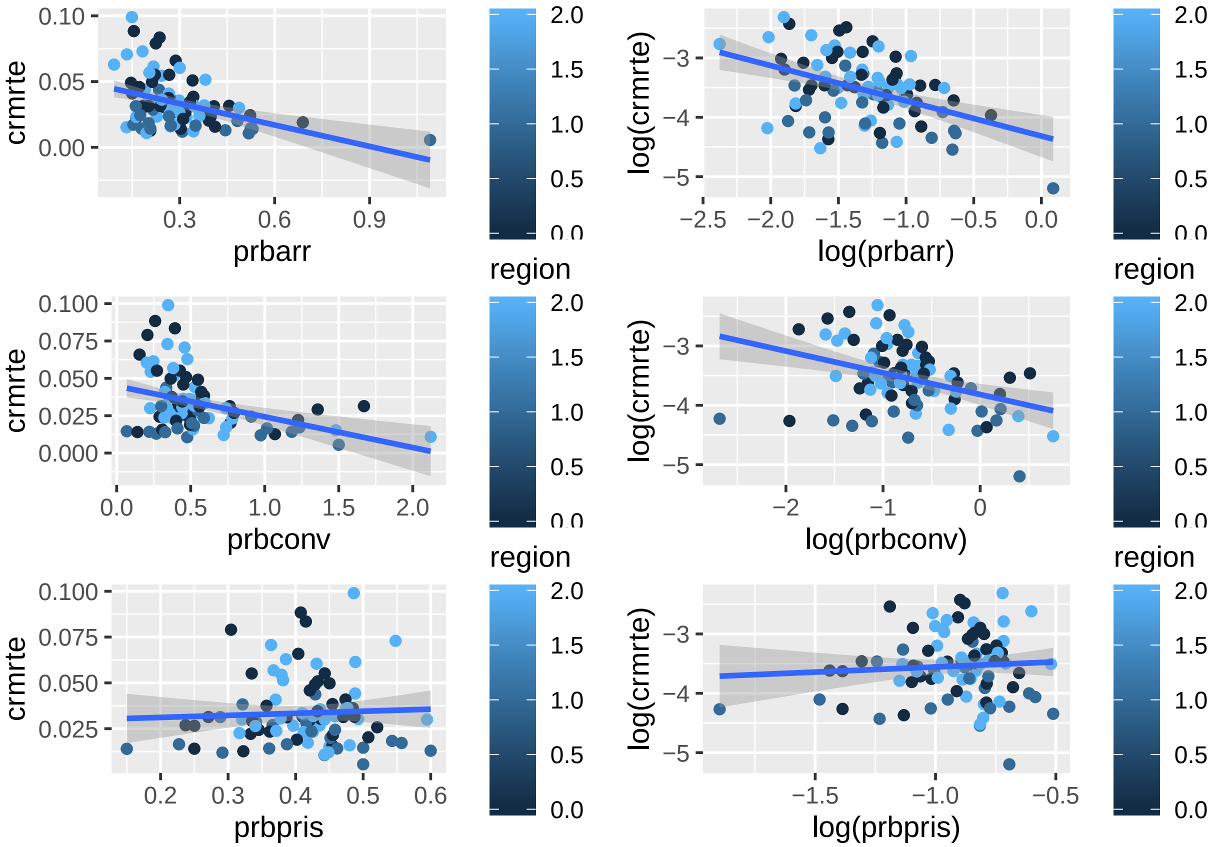


The transforms make the relationship more linearly distributed. We will transform these variables to their log equivalents.

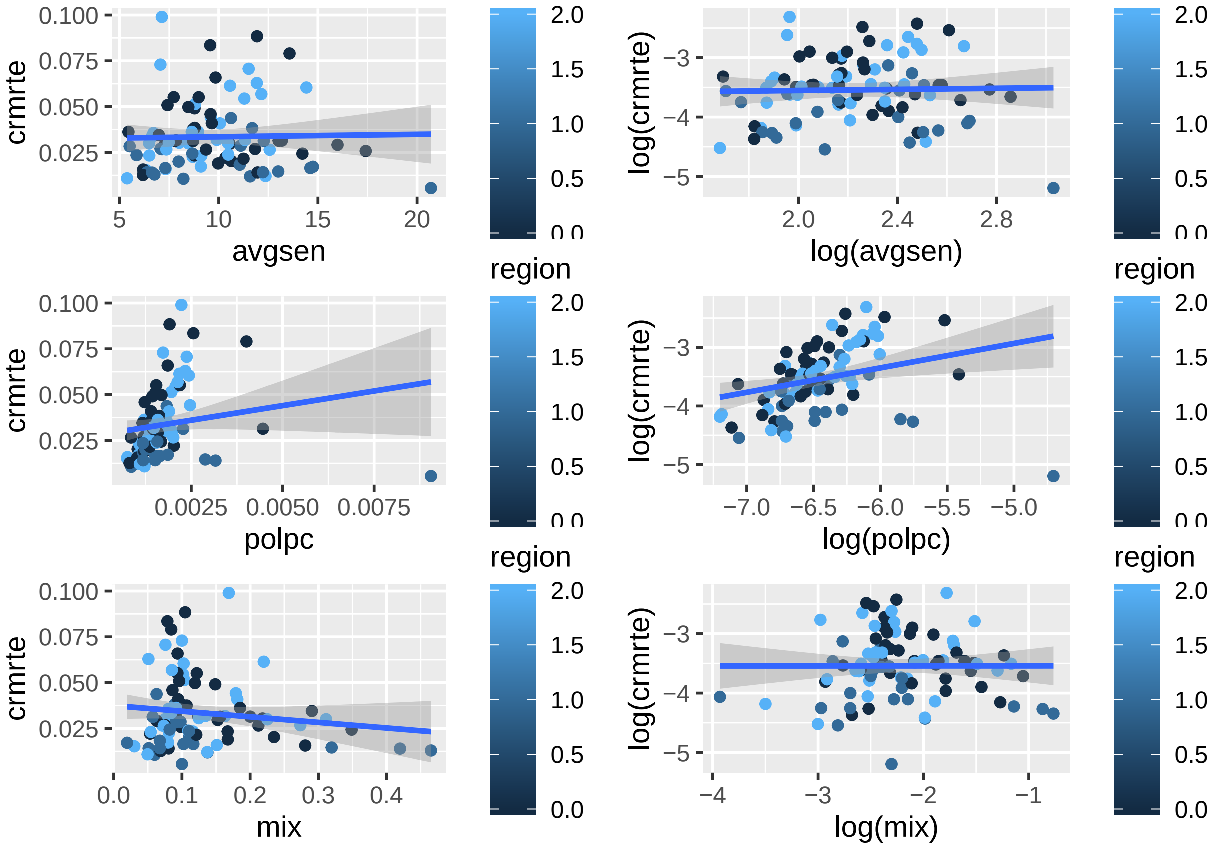
```
dfCrime$logwcon<-log(dfCrime$wcon)
dfCrime$logwtuc<-log(dfCrime$wtuc)
dfCrime$logwtrd<-log(dfCrime$wtrd)
dfCrime$logwfir<-log(dfCrime$wfir)
dfCrime$logwser<-log(dfCrime$wser)
dfCrime$logwmfg<-log(dfCrime$wmfg)
dfCrime$logwfed<-log(dfCrime$wfed)
dfCrime$logwsta<-log(dfCrime$wsta)
dfCrime$logwloc<-log(dfCrime$wloc)
```

We move to the justice an law enforcement variables. With these variables being mostly < 1 we'll also take the log for comparison.

```
#Plot of the criminal justice and law enforcment related variables vs crmrte
q1<-ggplot(data = dfCrime, aes(x = prbarr, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q1a<-ggplot(data = dfCrime, aes(x = log(prbarr), y = log(crmrte), color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q2<-ggplot(data = dfCrime, aes(x = prbconv, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q2a<-ggplot(data = dfCrime, aes(x = log(prbconv), y = log(crmrte), color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q3<-ggplot(data = dfCrime, aes(x = prbpris, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q3a<-ggplot(data = dfCrime, aes(x = log(prbpris), y = log(crmrte), color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q4<-ggplot(data = dfCrime, aes(x = avgsen, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q4a<-ggplot(data = dfCrime, aes(x = log(avgsen), y = log(crmrte), color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q5<-ggplot(data = dfCrime, aes(x = polpc, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q5a<-ggplot(data = dfCrime, aes(x = log(polpc), y = log(crmrte), color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q6<-ggplot(data = dfCrime, aes(x = mix, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q6a<-ggplot(data = dfCrime, aes(x = log(mix), y = log(crmrte), color = region)) +
      geom_point()+
  geom_smooth(method = "lm")

grid.arrange(q1, q1a, q2, q2a, q3, q3a, ncol=2)
```



```
grid.arrange(q4, q4a, q5, q5a, q6, q6a, ncol=2)
```



The log transformation for these variables makes the relationship more linear. We will transform these variables to their log equivalents.

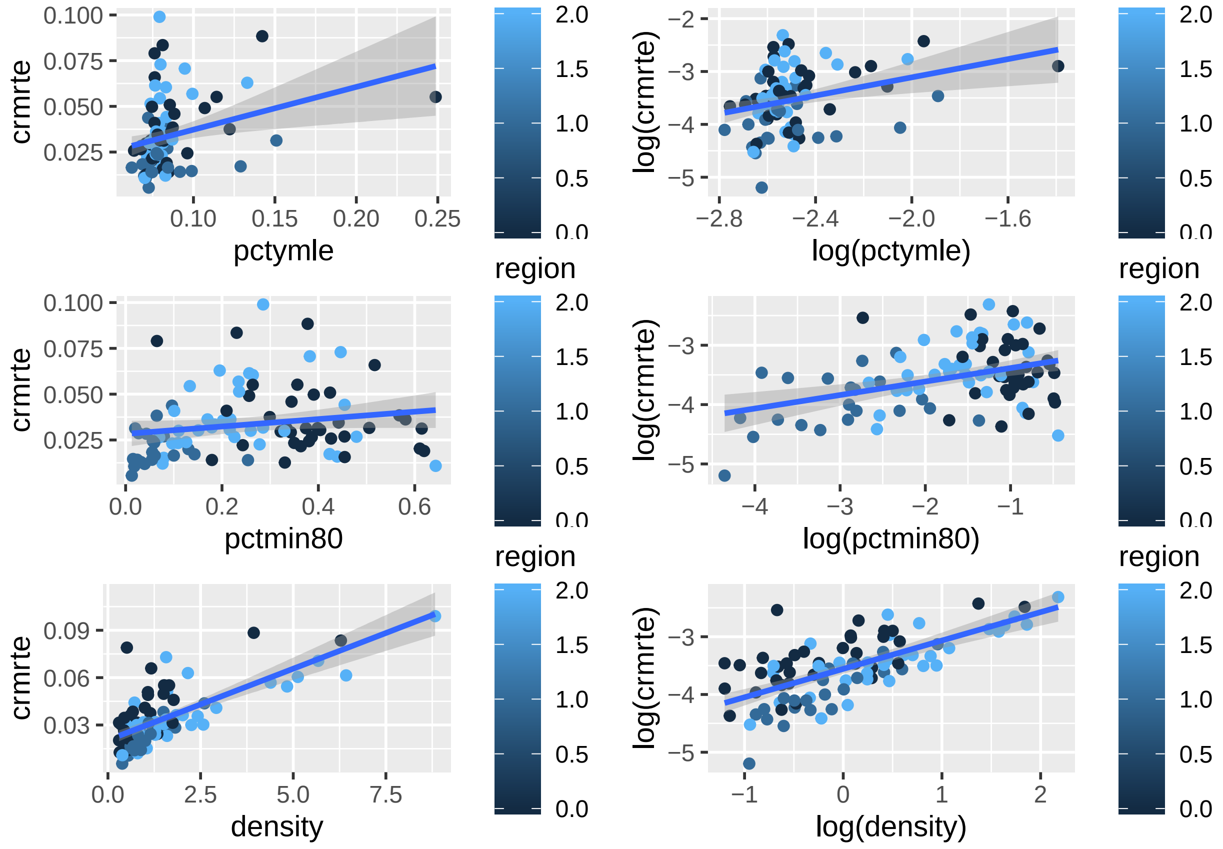
We also note that of the six variables, only prbarr, prbconv and polpc show univariate correlation with crime. We believe these will be better candidates for our model selection. Further, we see mix has no correlation with crmrate and may be its own outcome variable.

```
dfCrime$logprbarr <- log(dfCrime$prbarr)
dfCrime$logprbconv <- log(dfCrime$prbconv)
dfCrime$logprbpris <- log(dfCrime$prbpris)
dfCrime$logavgsen <- log(dfCrime$avgsen)
dfCrime$logpolpc <- log(dfCrime$polpc)
dfCrime$logmix <- log(dfCrime$mix)
```

Next we take a look at the demographic variables and their log alternatives

```
q1<-ggplot(data = dfCrime, aes(x = pctymle, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q1a<-ggplot(data = dfCrime, aes(x = log(pctymle), y = log(crmrte), color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q2<-ggplot(data = dfCrime, aes(x = pctmin80, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q2a<-ggplot(data = dfCrime, aes(x = log(pctmin80), y = log(crmrte), color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
```

```
q3<-ggplot(data = dfCrime, aes(x = density, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q3a<-ggplot(data = dfCrime, aes(x = log(density), y = log(crmrte), color = region)) +
      geom_point()+
  geom_smooth(method = "lm")


grid.arrange(q1, q1a, q2, q2a, q3, q3a, ncol=2)
```

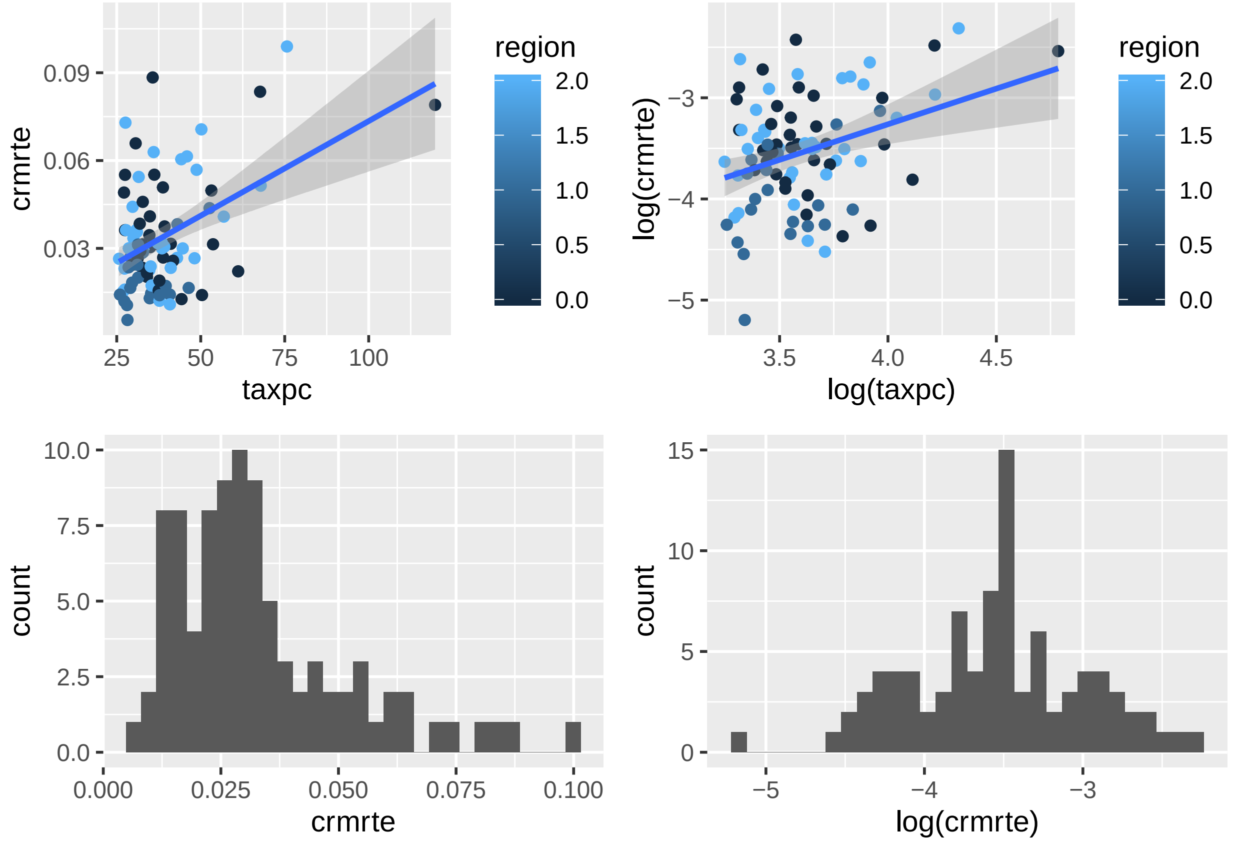


Again we see improvements after transformation. We will include transforms of these variables as well.

```
dfCrime$logdensity <- log(dfCrime$density)
dfCrime$logpctmin80 <- log(dfCrime$pctmin80)
dfCrime$logpctymle <- log(dfCrime$pctymle)
```

Finally, we'll take a look at taxpc and a histogram of the crmrte variable itself.

```
q1<-ggplot(data = dfCrime, aes(x = taxpc, y = crmrte, color = region)) +
      geom_point()+
  geom_smooth(method = "lm")
q1a<-ggplot(data = dfCrime, aes(x = log(taxpc), y = log(crmrte), color = region)) +
      geom_point()+
  geom_smooth(method = "lm")

q2<-ggplot(data = dfCrime, aes(x = crmrte)) +
      geom_histogram(bins=30)
q2a<-ggplot(data = dfCrime, aes(x = log(crmrte))) +
      geom_histogram(bins=30)

grid.arrange(q1, q1a, q2, q2a, ncol=2)
```

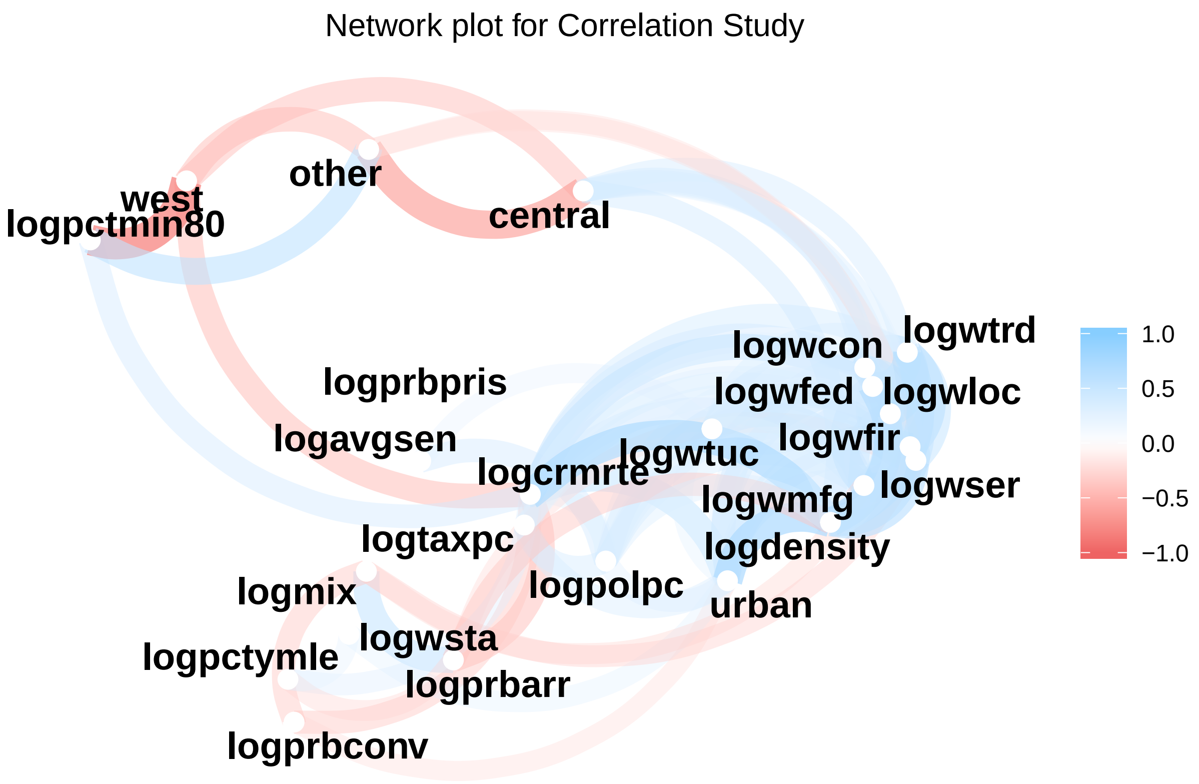


The crmrte and taxpc variables also show improvement after transformation. We'll add those to our dataframe.

```
dfCrime$logcrmrte = log(dfCrime$crmrte)
dfCrime$logtaxpc = log(dfCrime$taxpc)
```

With our variables transformed, we now turn to discussion on collinearity and multicollinearity in our data set. To facilitate the discussion we'll draw reference to a network plot.

```
options(repr.plot.width=8, repr.plot.height=8)
myData<-dfCrime
myData<-myData[, c("logcrmrte", "west", "central", "other", "urban", "logprbarr", "logprbconv", "logprbp
           "logpctmin80", "logwcon", "logwtuc", "logwtrd", "logwfir", "logwser", "logwmfg", "logwfed",
           "logmix", "logpctymle", "logdensity")]
plot<-myData %>% correlate() %>% network_plot(min_cor=.25)


Correlation method: 'pearson'
Missing treated using: 'pairwise.complete.obs'

grid.arrange(arrangeGrob(plot, bottom = 'Correlations Among Variables'),
             top = "Network plot for Correlation Study", ncol=1)
```



Correlations Among Variables

First, we note the general proximity of variables with one another. Variables that are clustered together have stronger affinities and degrees of collinearity. In fact, the cluster of the wage variables are an indication of multicollinearity. Only state wages fall outside this group. The telecome and utlity wage variable, while still near the cluster, show a little less collinearity. If we choose to operationalize the wage variables we must pick an appropriate strategy to minimize their multicollinearity impact. We also see the wage variables are positively correlated with our crime outcome variable. Density also positively correlates with wage and the crime rate variable. Urban also correlates with wage, but suprisingly the correlation between crime and urban is not as high.

Next, we notice the Law enforcement and Judicial variables are clustered and have a negative correlation with our outcome variable on crime. We also see they tend to be negatively correlated amongst one another. For example, probability of conviction is slightly negatively corrrelated with the probability of arrest, and both are negatively correlated with our outcome variable. We may wish to combine their impacts. We also see that police per capita and tax per capita are positively correlated with another. This makes sense as the more revenues collected the higher the ability to pay for community services such as law enforcement and protection. Both are also positively correlated with our outcome variable on crime. We also notice that percent young male has a positive correlation with crime rate. A possible explanation for this is that more crimes are committed by younger men as a whole. We also note that counties with higher state wages are correlated with higher percentages of young males, and these two variables are clustered together.

The mix variable is an odd one. It is positively correlated with probability of arrests, negatively correlated with probability of convictions, and negatively correlated with service wages and manufacturing wages. It also has a slight positive correlation with the state wage and seems to be clustered with it.

Last, we turn to our region variables and notice the high negative correlation of the minority variable with the western region. We also notice a high positive corrlation of minorities with the 'other' (eastern) region. This variable also correlates positively with crime rate, although the two are not clustered. We especially note that west is negatively correlated with crime rate. There appears to be a lessor propensity for crime in this region. We will examine this phenomenom further. Also, for a futher examination of correlation plots for each of the regions please see the network diagrams in the appendix.

2.2 Additional Variables to Operationalize

As a final point of discussion we will identify variables we wish to operationalize for use in our models. We will include a variable that expresses the economic condition of the county and a variable that expresses criminal justice effectiveness.

The first variable on the economic condition will include the sum of all average weekly wages from the 1980 census information. Since we do not know how many were employed at that wage we use this summary the best available proxy. Summing the wages into one variable will also remove their multicollinearity effects.

```
dfCrime$allWages<-dfCrime$wcon + dfCrime$wtuc + dfCrime$wtrd + dfCrime$wfir +
    dfCrime$wser + dfCrime$wmfg + dfCrime$wfed + dfCrime$wsta + dfCrime$wloc
```

As a second variable, we are interested in understanding the effectiveness of the criminal justice system as a crime deterrent. Our proxy will be the number of convictions per incident.

This is operationalized by taking the probability of arrests, pbrarr (which is defined as arrests per incident) and multiplying by the probability of convictions, pbrconv (which is defined as convictions per arrest). The new variable is defined below.

```
dfCrime$crimJustEff<-dfCrime$prbarr * dfCrime$prbconv
```

We will also create a logarithmic transformation of this variable based on our histogram analysis from before.

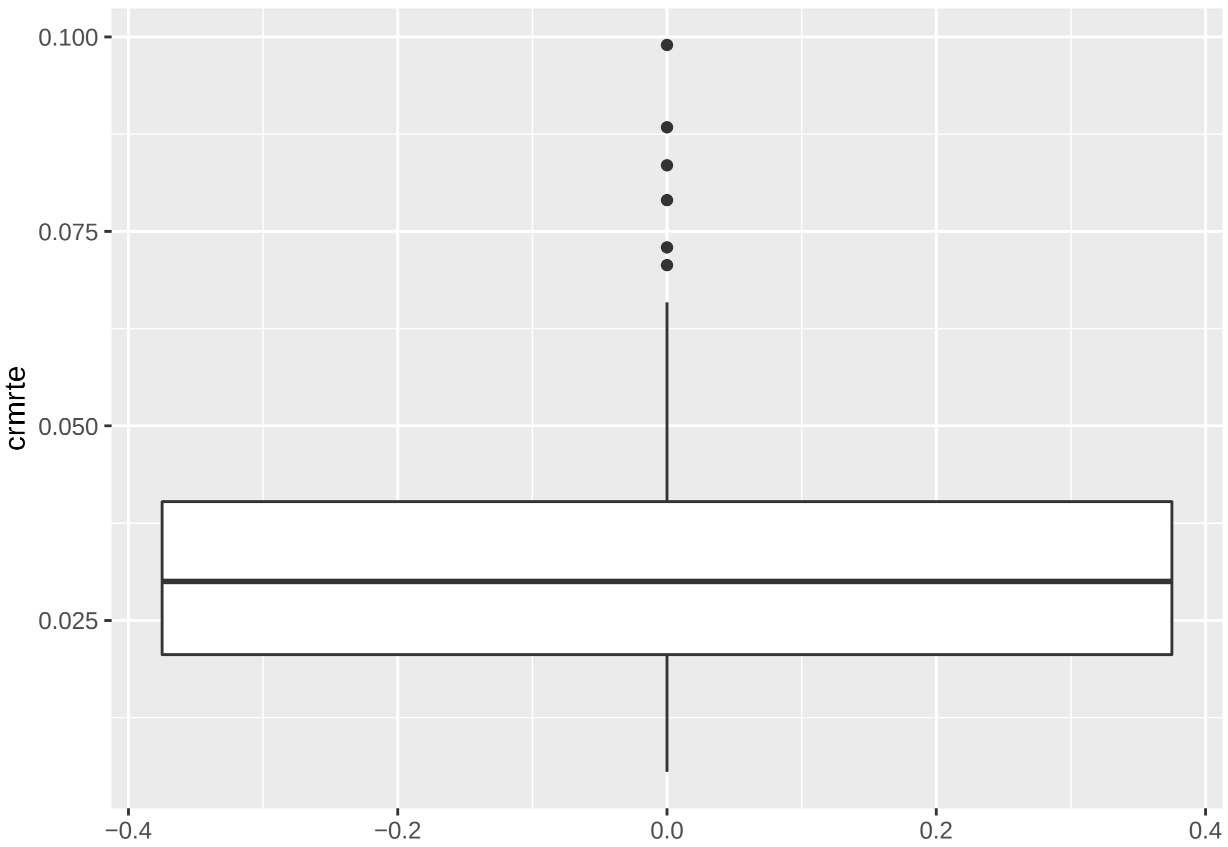
```
dfCrime$logcrimJustEff<-log(dfCrime$crimJustEff)
```

2.3 Summary and Results

Our outcome variable is the *crime rate* (“crmrte”), which is defined as the crimes committed per person in a specific county during 1987. The crime rate of the 90 counties in our sample dataset range between 0.0055 - 0.0990, with a mean of 0.0335.

From the boxplot below, most of the counties have a crime rate between 0.0055 and 0.0700, with 5 outliers having a crime rate > 0.0700.

```
options(repr.plot.width=3, repr.plot.height=4)
ggplot(data = dfCrime, aes(y = crmrte)) +
      geom_boxplot()
```



While mix (the type of crime committed) is also potentially an outcome variable, our research focuses on providing policy recommendations to reduce crime in general and not a specific type of crime. Mix is also not a linear outcome and hence difficult to measure.

We propose 3 multiple linear regression models

- First Model: Has only the explanatory variables of key interest and no other covariates.
- Second Model: Includes the explanatory variables and covariates that increase the accuracy of our results without substantial bias.
- Third Model: An expansion of the second model with most covariates, designed to demonstrate the robustness of our results to model specification.

As we proceed with each model, we verify the CLM assumptions of OLS are addressed below:

- **MLR1** Linear in parameters: The models have had its data transformed as described above to allow a linear fit of the model.
- **MLR2** Random Sampling: The data is collected from a data set with rolled up data for each county. It is not randomly sampled by area or population.
- **MLR3** to be discussed on a model by model basis.
- **MLR4** to be discussed on a model by model basis.
- **MLR5’** to be discussed on a model by model basis.
- **MLR6’** to be discussed on a model by model basis.

By satisfying these assumptions, we can expect our coefficients will be approaching the true parameter values in probability.

3 Model Analysis

3.1 Model 1

3.1.1 Introduction

Our base hypothesis is that crime can be fundamentally explained by two factors: the effectiveness of the criminal justice system and the economic conditions.

Criminal Justice Effectiveness is self defined : To be able to track crimes, they must be reported to police, who can then make arrests and the legal system provides judgement (convictions/sentencing).
Criminal justice also has a relationship to crime as a deterrent, as the probability of getting caught, convicted, sentenced could potentially deter crime.

We operationalize criminal justice effectiveness as (probability of Convictions * Crimes committed). We define this as: prbconv * prbarr = conv/arrest * arrest/crime = convictions/crime. Without more granular data, this provides a single parsimonious metric that helps understand how well the law enforcement and criminal justice system works.

We theorize that the second major cause of crime are bad economic conditions. When there are worse economic conditions, crime can be increased due to:

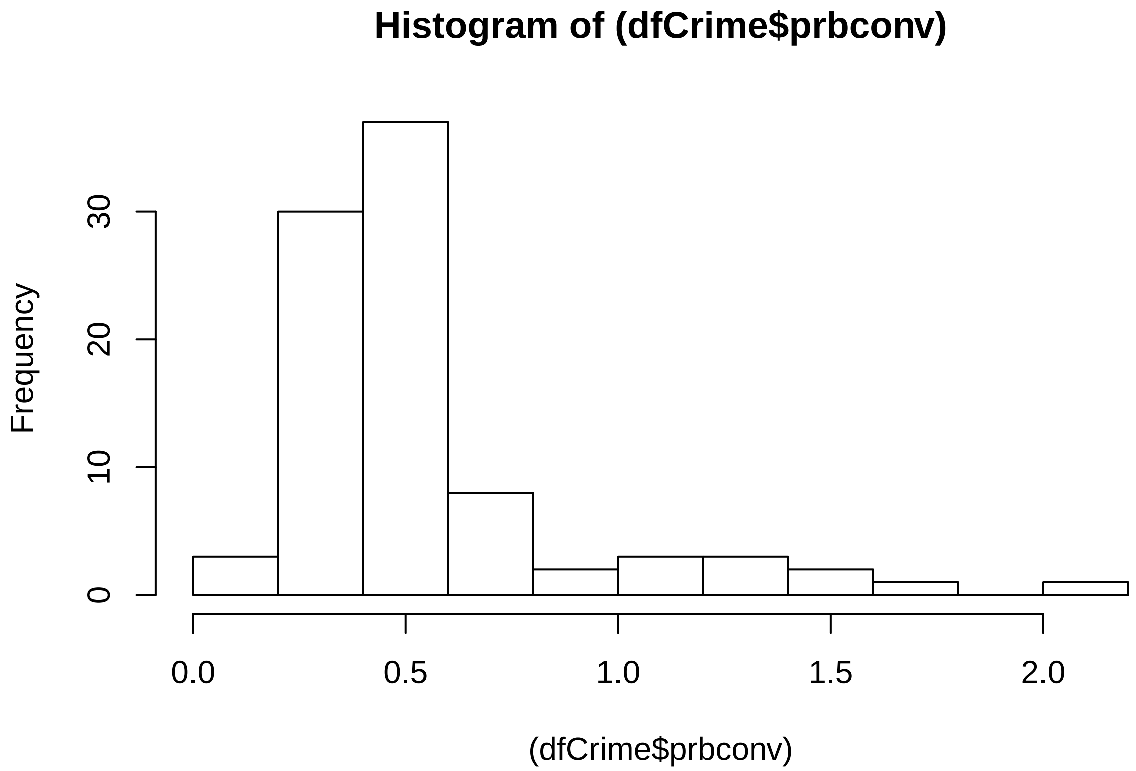
- Lack of means: People forced into crimes because they need to make ends meet
- Lack of occupation: People commit crimes because they are not busy at work
- Lack of opportunity: High discount rate for future due to no long-term opportunity, incentive to take the risk and commit crimes hoping for big payoff.

We operationalize economic conditions by looking at wages. For this model, we define this as the unweighted average weekly pay from each sector provided in the data set. We think this is best proxy from our data because it answers all of the above (higher wages leads to better means and better opportunities). From our EDA we also confirm that in general these sums are not skewed by having 1 really high paying sector in each county as we see a strong relationship between avg quartile across all job types and total sum. This can be seen in the chart below.

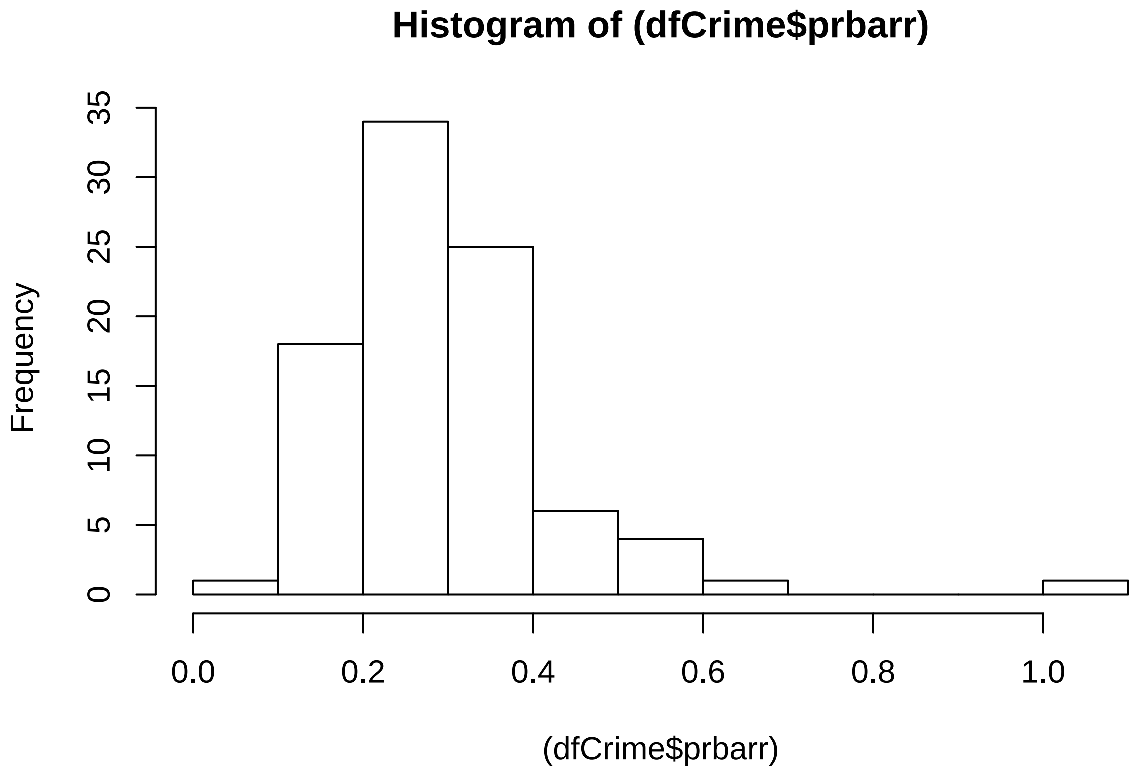
3.1.2 Model 1 EDA

Data Transformations

```
options(repr.plot.width=4, repr.plot.height=4)
hist((dfCrime$prbconv))
```



```
hist((dfCrime$prbarr))
```



The distribution of both probability of conviction and probability of arrest are peculiar and non-normal. It could be argued that both of these variables should be bound between 0 and 1. However, "probability" of conviction is proxied by a ratio of convictions to arrests. It is in fact common that defendents are charged with multiple crimes and convicted, but were only arrested once.

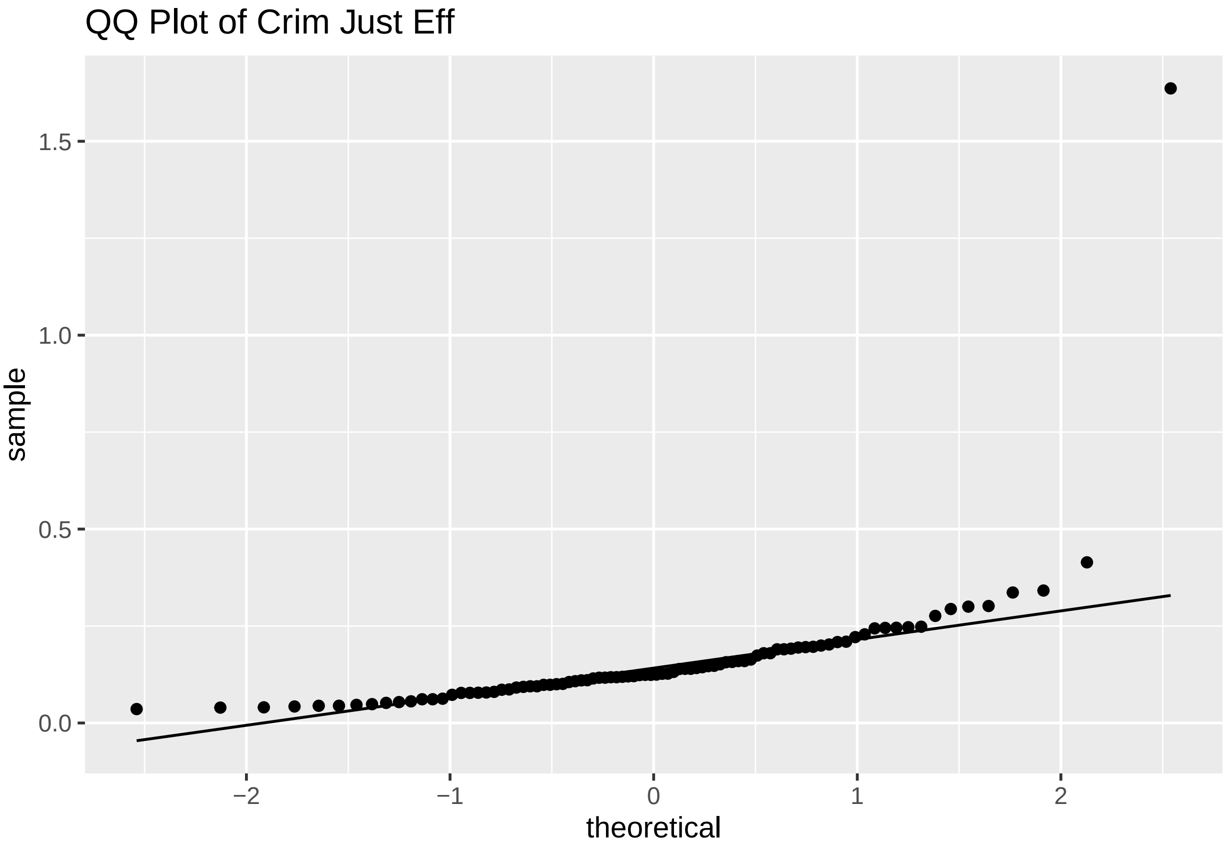
For "probability" of arrest, it could be possible there are multiple arrests for a single offense. However, the single data point that is greater than one, is >3 standard deviations away from the distribution. Since this value falls so far out of distribution it will have high leverage on our model and will be preemptively imputed as the data supplied is likely in error and is not representative of the bulk of North Carolina counties.

For parsimony, we can simply the probability of arrest and probability of conviction by multiplying to effectively get the ratio of convictions to offenses. The normality of this factor can be improved by taking a log transform. QQ plots help to visualize how normality improves for the inner quartiles.

```r
# how many standard deviations away the outlier lies
(dfCrime[51,]$prbarr - mean(dfCrime$prbarr))/sd(dfCrime$prbarr)
```

```
[1] 5.779438
```

```r
#hist(log(dfCrime$crimJustEff))
ggplot(data=dfCrime, aes(sample= crimJustEff)) + stat_qq() + stat_qq_line() +
  ggtitle("QQ Plot of Crim Just Eff")
```



```
dfCrime[dfCrime$crimJustEff > 1,] # find outlier
```

```
   county year    crmrte   prbarr prbconv prbpris avgsen      polpc
51    115   87 0.0055332 1.09091     1.5     0.5   20.7 0.00905433
     density   taxpc west central urban  pctmin80     wcon     wtuc
51 0.3858093 28.1931    1       0     0 0.0128365 204.2206 503.2351
       wtrd     wfir     wser   wmfg  wfed    wsta   wloc mix    pctymle
51 217.4908 342.4658 245.2061 448.42 442.2 340.39 386.12 0.1 0.07253495
   region regcode other nonurban         metro  logwcon  logwtuc  logwtrd
51      1    West     0        1 Outside Metro 5.319201 6.221057 5.382157
    logwfir  logwser logwmfg  logwfed  logwsta  logwloc  logprbarr
51 5.836172 5.502099 6.10573 6.091762 5.830092 5.956148 0.08701217
   logprbconv logprbpris logavgsen  logpolpc    logmix logdensity
51  0.4054651 -0.6931472  3.030134 -4.704512 -2.302585 -0.9524121
   logpctmin80 logpctymle logcrmrte logtaxpc allWages crimJustEff
51   -4.355463  -2.623687 -5.196989 3.339077 3129.748    1.636365
   logcrimJustEff
51      0.4924773
```

We will use the imputation method to replace the large prbarr value and remove the outlier effect, while also retaining the rest of the variables in the county.

—delete—- We also see that polpc is .009. We noticed this outlier during our EDA analysis. Based on the records describing the US population on police officers per capita, the highest police per capita on record is .007 in Atlantic City, NJ. https://www.governing.com/gov-data/safety-justice/police-officers-per-capita-rates-employment-for-city-departments.html This datapoint is also in error and we will impute it's replacement. —/delete—

```
dfCrime$prbarr[which(dfCrime$county==115)]<-NA # set the value to NA so it will be imputed
#dfCrime$prbconv[which(dfCrime$county==115)]<-NA # set the value to NA so it will be imputed
#dfCrime$polpc[which(dfCrime$county==115)]<-NA # set the value to NA so it will be imputed delete
```

```
impute_arg <- aregImpute(~ crmrte +  urban + central + west + other +
                         prbarr + prbconv + prbpris + avgsen + polpc +
                         density + taxpc + pctmin80 + wcon + wtuc +
                         wtrd + wfir + wser + wmfg + wfed + wsta + wloc +
                         mix + pctymle, data = dfCrime, match="weighted",
                         nk=3, B=10, n.impute = 100)
```

```
paste("R-squares for Predicting Non-Missing Values for Each Variable")
```

```
[1] "R-squares for Predicting Non-Missing Values for Each Variable"
```

```
impute_arg$rsq
```

```
   prbarr 
0.9196967 
```

```
paste("Distribution of Values for Each Imputation")
```

```
[1] "Distribution of Values for Each Imputation"
```

```
table(impute_arg$imputed$prbarr)
```

```

0.092770003 0.132028997 0.149936005 0.161381006 0.182589993 0.221542001 
          1           3           1           1           1           1 
0.238636002 0.243119001 0.251599014  0.26602599 0.266054988 0.266959995 
          1           1           1           1           2           1 
0.269042999 0.271966994 0.283504993 0.289121002 0.298269987 0.310986996 
          2           1           2           1          56           1 
0.338901997 0.350347996  0.35473299 0.362269998 0.392111003 0.444444001 
          1           1           1           2           1           1 
0.456393987 0.482425004 0.487430006 0.518218994 0.522696018 0.524663985 
          1           1           1           1           3           2 
0.530435026 0.689023972 
          2           4 
```

```
paste("Distribution of Values for Each Imputation")
```

```
[1] "Distribution of Values for Each Imputation"
```

```
table(impute_arg$imputed$prbconv)
```

```
< table of extent 0 >
```

```
paste("Distribution of Values for Each Imputation")
```

```
[1] "Distribution of Values for Each Imputation"
```

```
table(impute_arg$imputed$polpc)
```

```
< table of extent 0 >
```

We will reassign the value in our dataset to the mean from these trials.

```
dfCrime$prbarr[which(dfCrime$county==115)]<-mean(impute_arg$imputed$prbarr)
dfCrime$prbarr[which(dfCrime$county==115)]
```

```
[1] 0.3266697
```

```
dfCrime$prbconv[which(dfCrime$county==115)]<-mean(impute_arg$imputed$prbconv)
dfCrime$prbconv[which(dfCrime$county==115)]
```

```
[1] NA
```

```
dfCrime$polpc[which(dfCrime$county==115)]<-mean(impute_arg$imputed$polpc)
dfCrime$polpc[which(dfCrime$county==115)]
```

```
[1] NA
```

```
dfCrime$logprbarr[which(dfCrime$county==115)]<-log(dfCrime$prbarr[which(dfCrime$county==115)])
dfCrime$logprbarr[which(dfCrime$county==115)]
```

```
[1] -1.118806
```

```
dfCrime$logprbconv[which(dfCrime$county==115)]<-log(dfCrime$prbconv[which(dfCrime$county==115)])
dfCrime$logprbconv[which(dfCrime$county==115)]
```

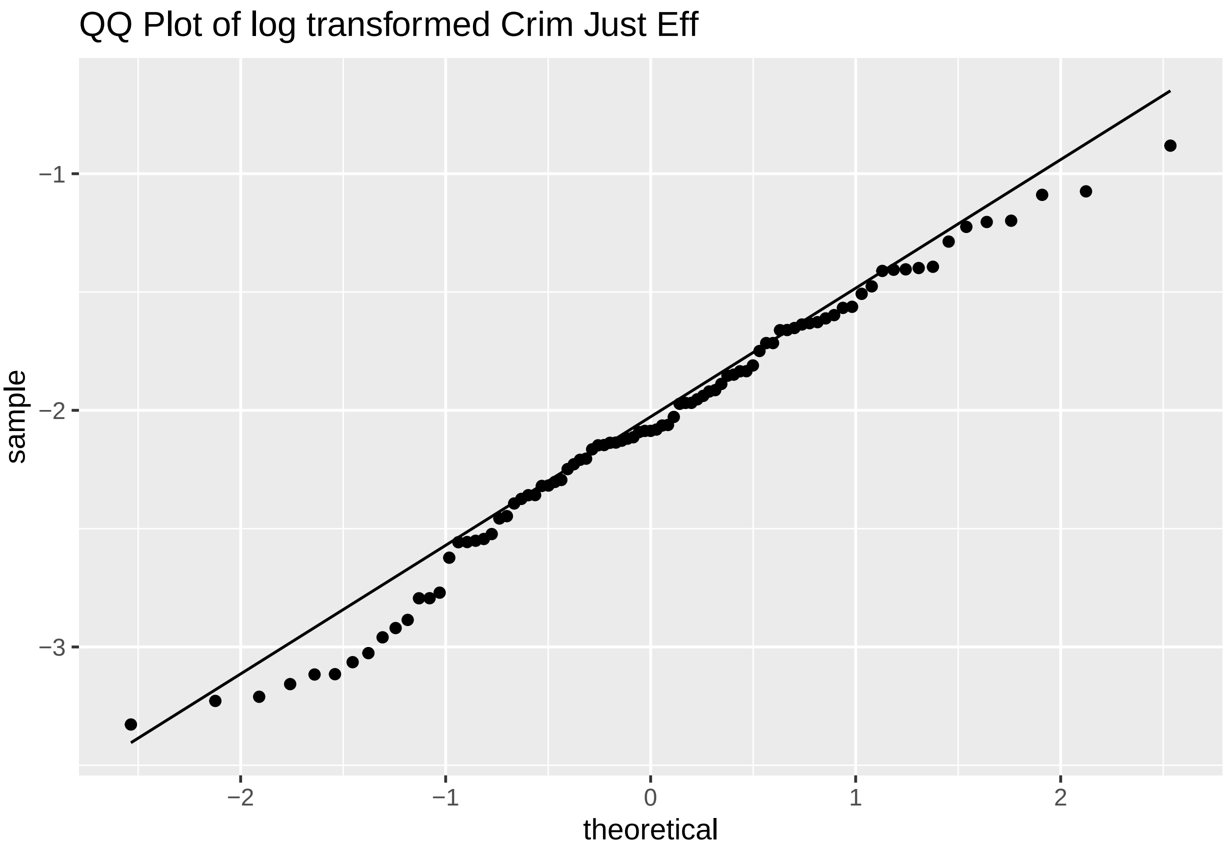
```
[1] NA
```

```
dfCrime$logpolpc[which(dfCrime$county==115)]<-log(dfCrime$polpc[which(dfCrime$county==115)])
dfCrime$logpolpc[which(dfCrime$county==115)]
```

```
[1] NA
```

```
dfCrime$crimJustEff<-dfCrime$prbarr * dfCrime$prbconv
dfCrime$logcrimJustEff<-log(dfCrime$crimJustEff)
```

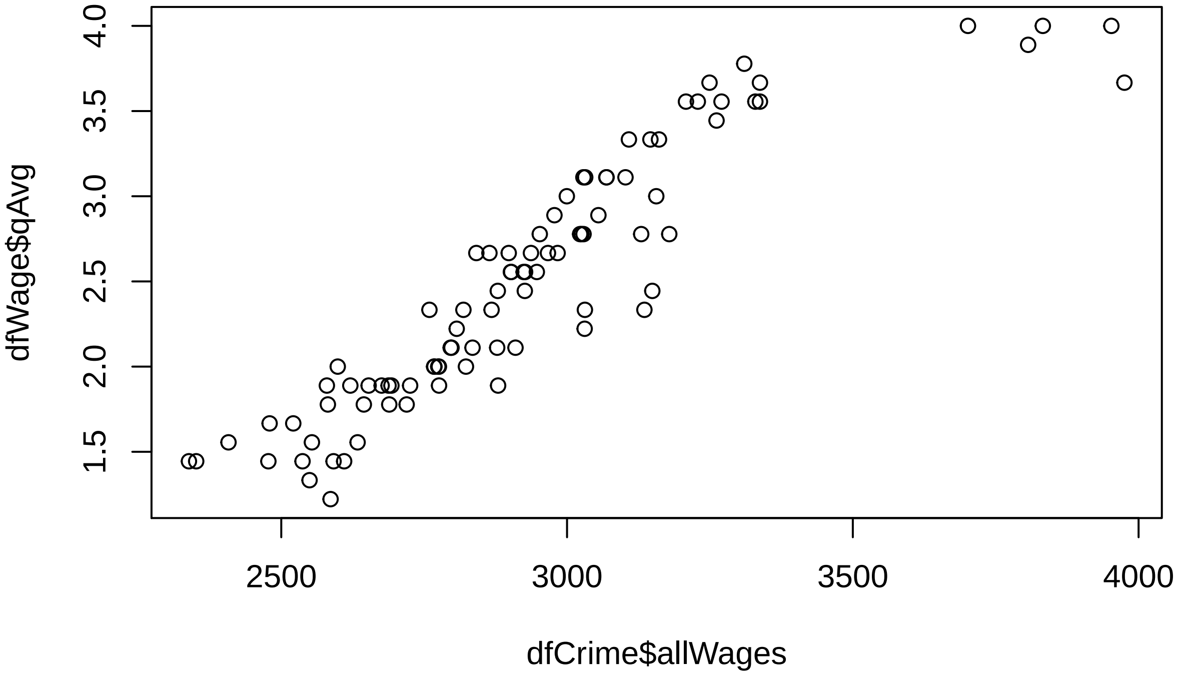
```
ggplot(data=dfCrime, aes(sample= logcrimJustEff)) + stat_qq() + stat_qq_line() +
ggtitle("QQ Plot of log transformed Crim Just Eff")
```



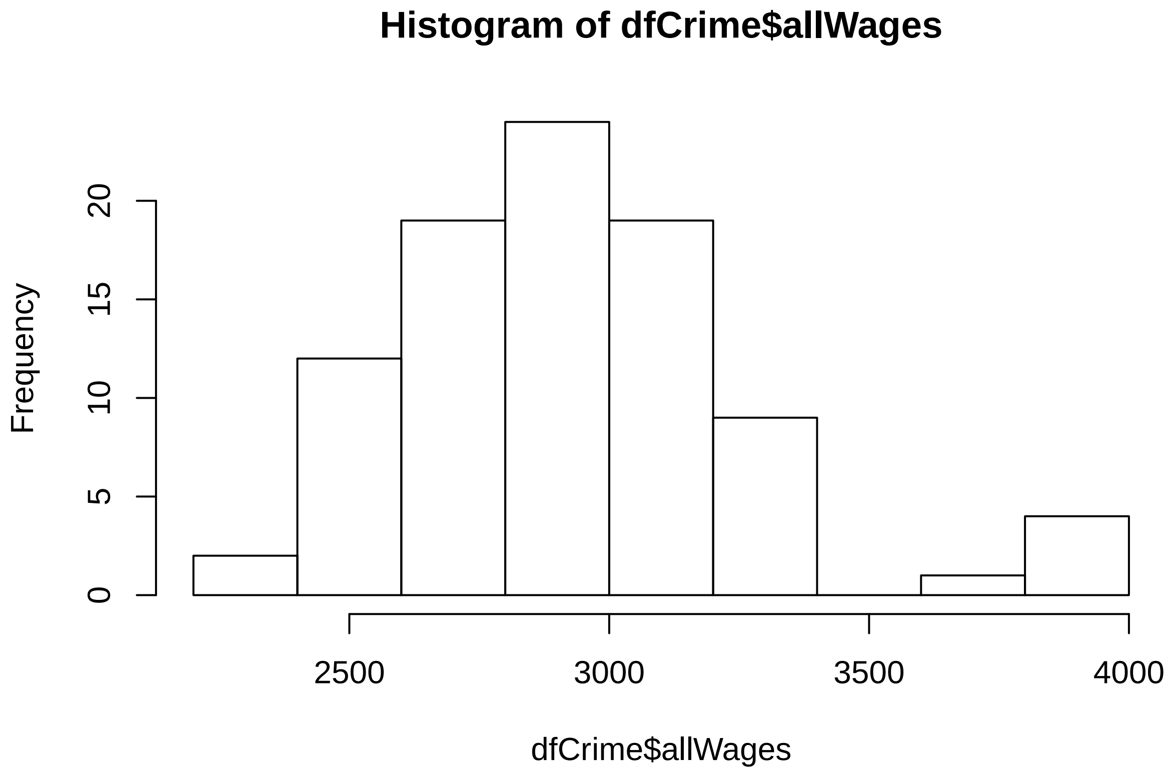
```
## Can show histogram/qqplot side by side in RMD.
```

For unweighted average of the wages for each sector, we can see issues.

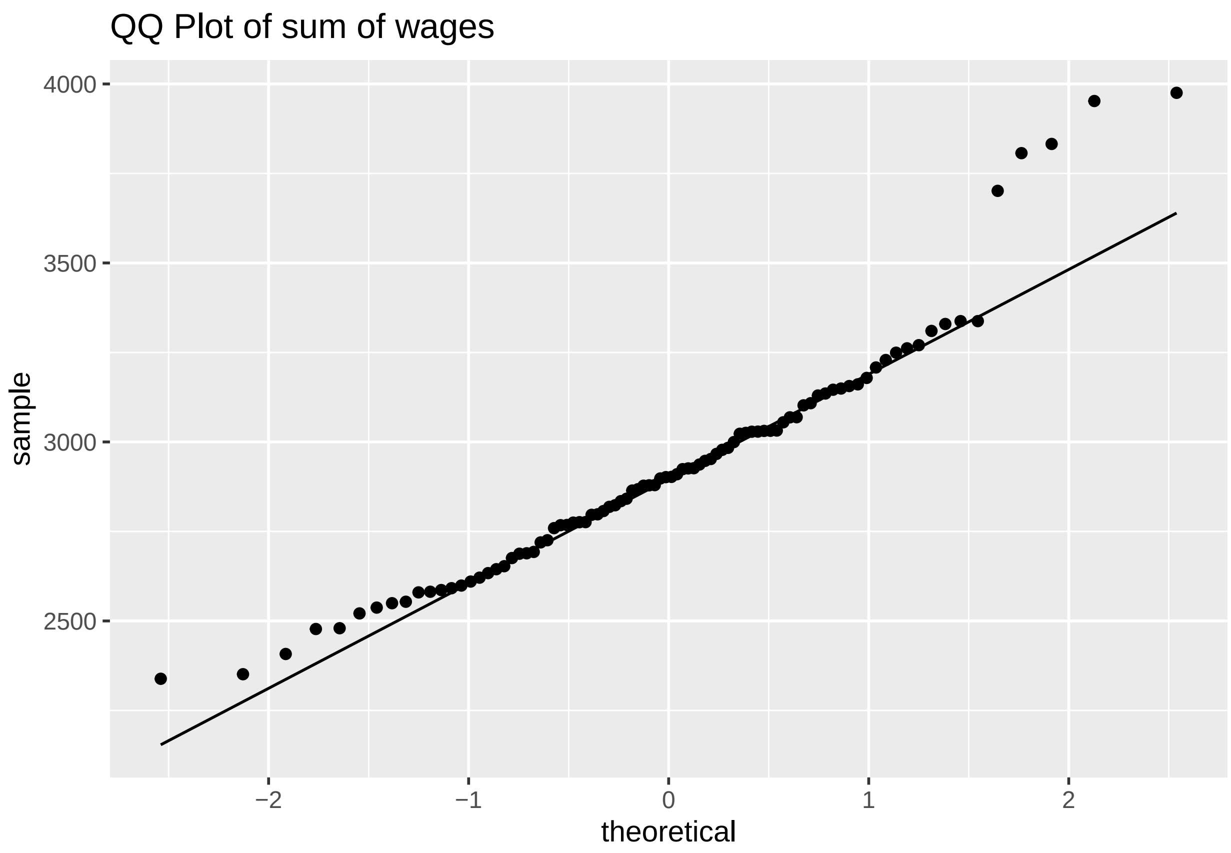
```
# # Quantiles for all jobs
dfWage<-mutate(dfCrime,qCon=ntile(dfCrime$wcon,4))
dfWage<-mutate(dfWage,qTuc=ntile(dfCrime$wtuc,4))
dfWage<-mutate(dfWage,qTrd=ntile(dfCrime$wtrd,4))
dfWage<-mutate(dfWage,qFir=ntile(dfCrime$wfir,4))
dfWage<-mutate(dfWage,qSer=ntile(dfCrime$wser,4))
dfWage<-mutate(dfWage,qMfg=ntile(dfCrime$wmfg,4))
dfWage<-mutate(dfWage,qFed=ntile(dfCrime$wfed,4))
dfWage<-mutate(dfWage,qSta=ntile(dfCrime$wsta,4))
dfWage<-mutate(dfWage,qLoc=ntile(dfCrime$wloc,4))
## Average quantile
dfWage$qAvg= (dfWage$qCon+dfWage$qTuc+dfWage$qTrd+dfWage$qFir+dfWage$qSer+dfWage$qMfg+
                dfWage$qFed+dfWage$qSta+dfWage$qLoc)/9
plot(dfCrime$allWages,dfWage$qAvg)
```



```
hist(dfCrime$allWages)
```



```
ggplot(data=dfCrime, aes(sample= allWages)) + stat_qq() + stat_qq_line() +
  ggtitle("QQ Plot of sum of wages")
```



3.1.3 Model 1 Linear Model

```
dfCrime$unweighted_avg_wage <- dfCrime$allWages/9
mod1 <- lm(dfCrime$logcrmrte ~ dfCrime$unweighted_avg_wage + dfCrime$logcrimJustEff)
coeftest(mod1, vcov=vcovHC)
```

```

t test of coefficients:

                              Estimate Std. Error  t value  Pr(>|t|)    
(Intercept)                 -6.3002758  0.3824297 -16.4743 < 2.2e-16 ***
dfCrime$unweighted_avg_wage  0.0057448  0.0016156   3.5560 0.0006151 ***
dfCrime$logcrimJustEff      -0.4344602  0.1221824  -3.5558 0.0006153 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
vif(mod1)
```

```
dfCrime$unweighted_avg_wage      dfCrime$logcrimJustEff 
                   1.062147                    1.062147 
```

```
summary(mod1)$adj.r.square
```

```
[1] 0.4561573
```

```
shapiro.test(mod1$residuals)
```

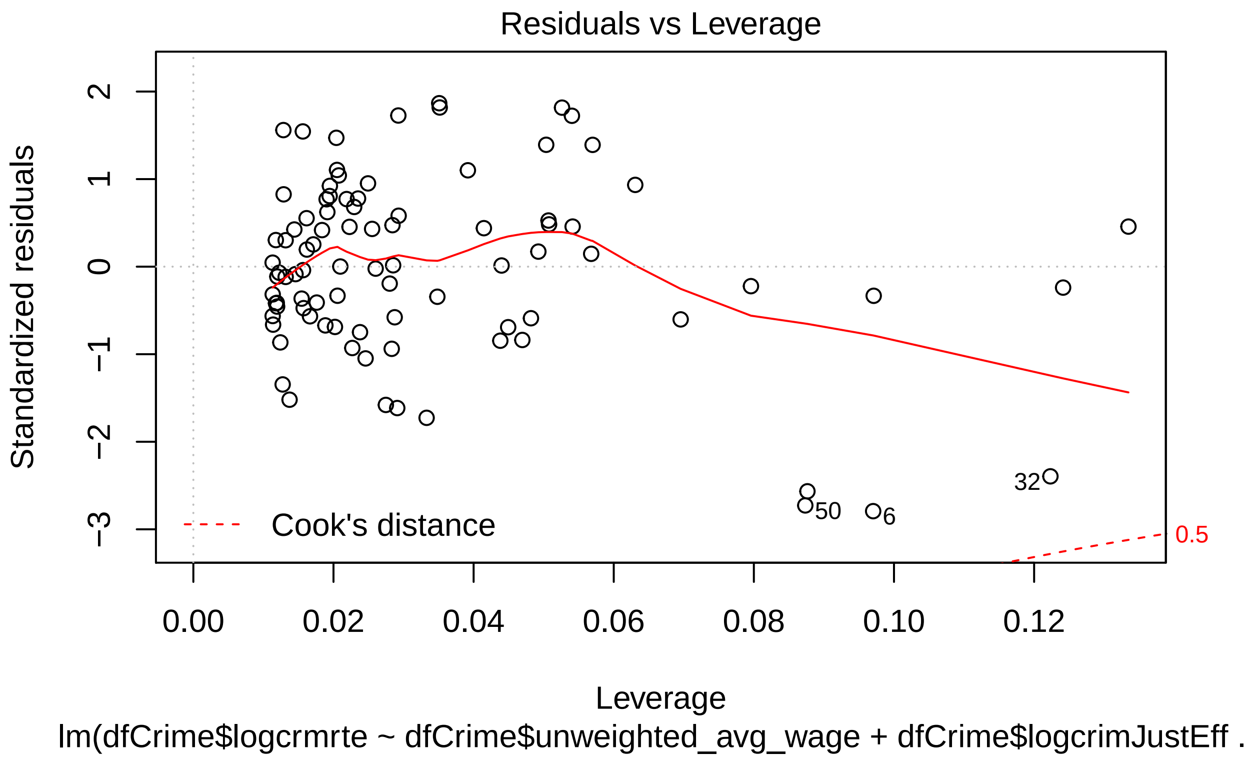
```

    Shapiro-Wilk normality test
```

```
data:  mod1$residuals
W = 0.97377, p-value = 0.0682
```

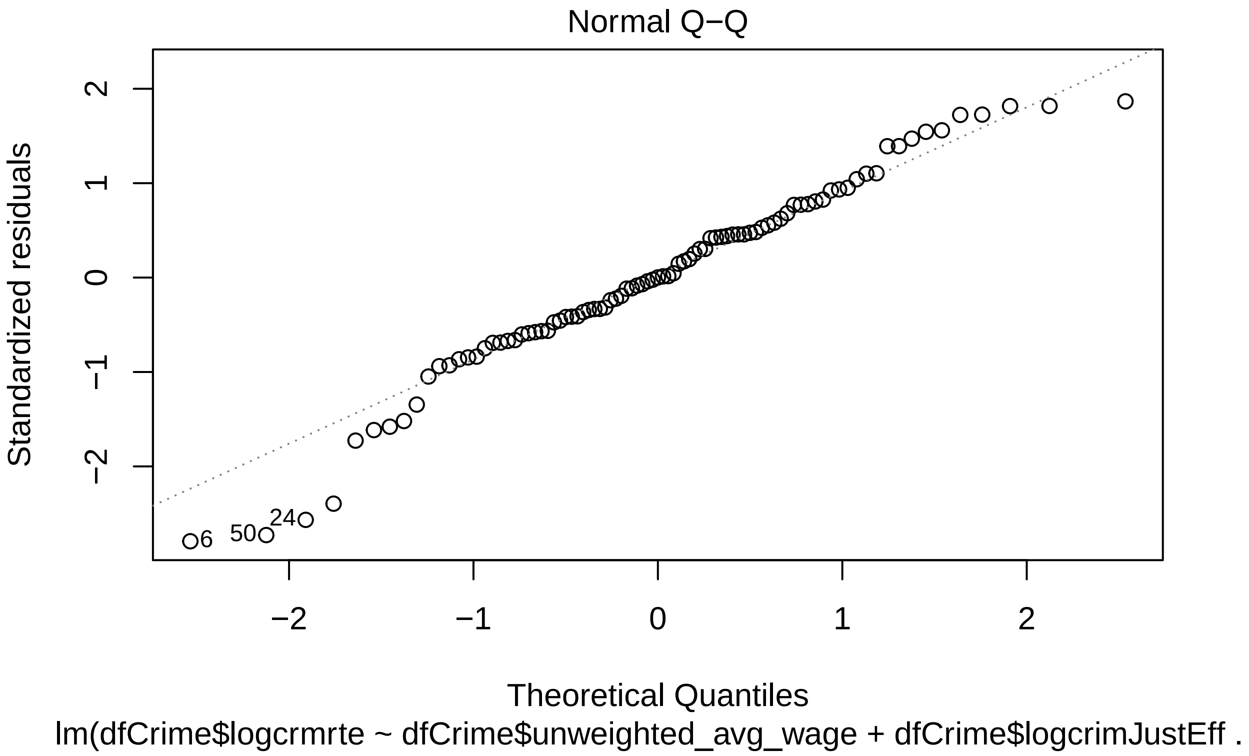
The model gives estimates and standard errors that are heteroskedastic consistent. The coefficient of unweighted_avg_wage is calculated to have a coefficient of .005. This means that an increase of $100 in weekly wages is correlated with an increase of .5% in crime rate. Generally increased wages are not associated with increased crime. This suggests that wages are correlated with a stronger omitted variable that affects crime.

Similarly, criminal justice effectiveness (convictions/crime) is given a coefficient of -0.489 which suggests that an increase 1% increase in convictions per crime is will decrease crime by nearly .5%. This suggests that we have found a are strong correlation and perhaps a good influence on crime rate in a county.

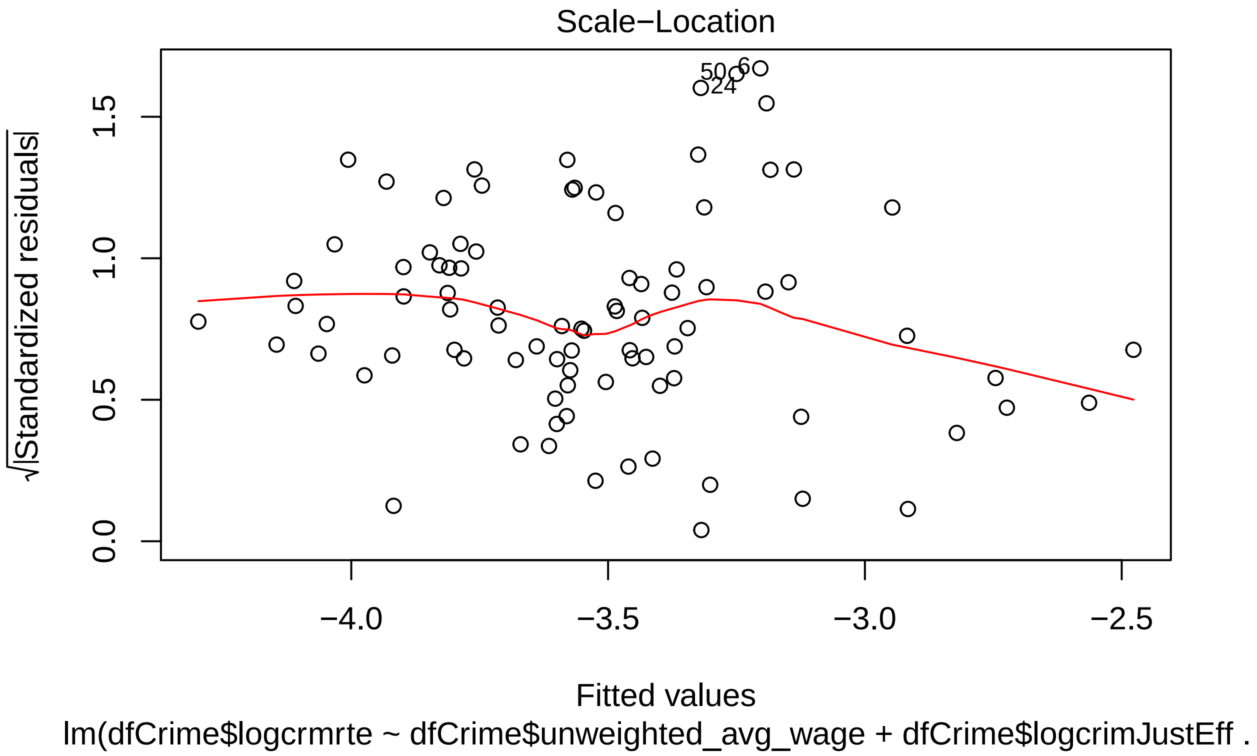
```
plot(mod1, which=5)
```



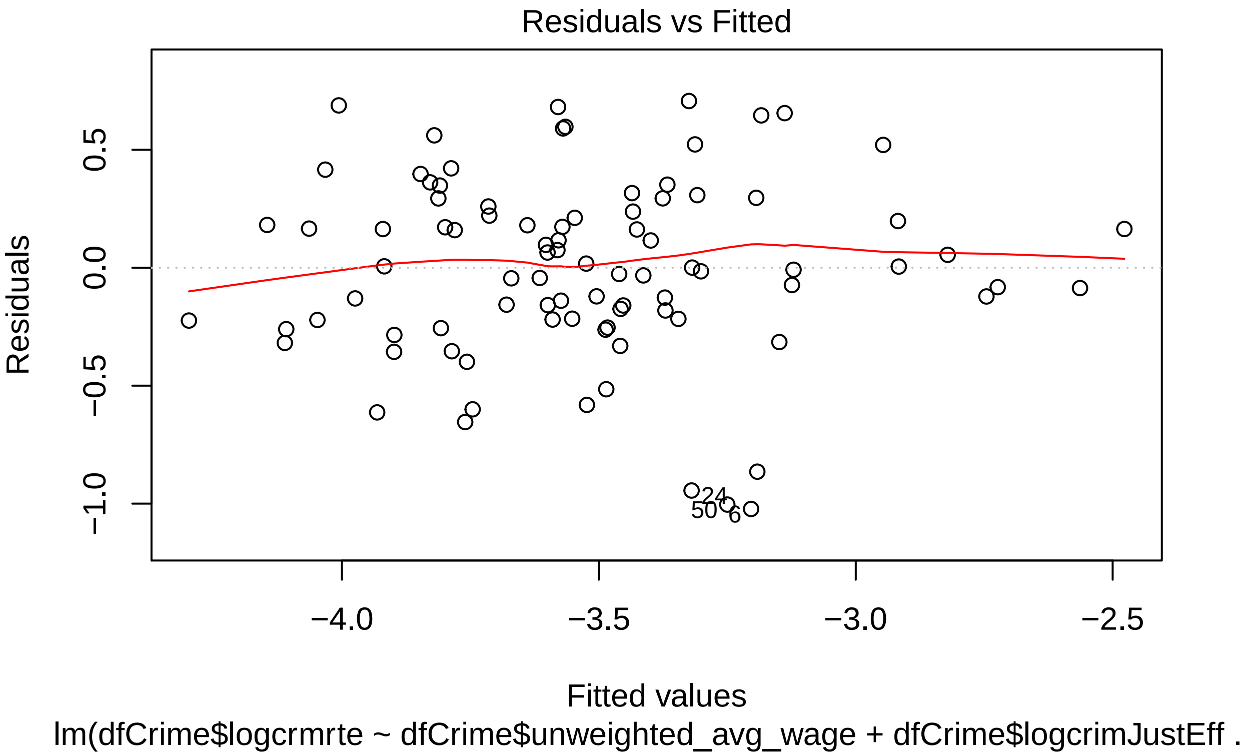
```
plot(mod1, which=2)
```



```
plot(mod1, which=3)
```



```
plot(mod1, which=1)
```



The model shows a moderate good fit, with an adjusted R square of 0.46. This can be interpreted as, the model explains 46% of the variation in crime. Next the model is plotted in a Residuals vs Leverage plot. This plot shows that all the points have a cook's distance of less than 0.5. There are no points that have enough leverage and residual than when deleted greatly alter the model coefficients.

The root of standardized residuals all fall within about 1.6. This is very good, as we can expect 95% of the points to fall within 3 standardized residuals of each other. $(\sqrt{(3)} \approx 1.73)$

Finally, the residuals vs fitted plot shows a well centered and mostly nromal distribution about 0. There are no major trends or variation changes across the fitted values. This suggests that major uncorrelated variables have not been left out of the model. We will discuss the possible ommited variable biases further, in the next sections.

Model 1 CLM Assumptions: [To be finalized] * **MLR1** Linear in paramters: The model has had its data transformed as described above to allow a linear fit of the model. * **MLR2** Random Sampling: The data is collected from a data set with rolled up data for each county. It is not randomly sampled by area or population. * **MLR3** No perfect multicollinearity: None of the variables chosen for the model are constant or perfectly collinear as the economy and criminal justice effectiveness are independent. * **MLR4'** The expectation of u and and covariance of each regressor with u are ~0. This shows that our model's regressors are exogenous with the error.
* **MLR4** The zero conditional mean assumption is well supported when viewing the Residuals vs fitted plot. The split fit is nearly flat and centered at 0. * **MLR5** There does appear to be heteroskedacity in the 'lips' appearance of the Residuals vs fitted plot. This is acknowledged and can be accounted for by using the heteroskedastic robust standard errors. This is seen in the coeftest. * **MLR6** The final assumption of linear regression is that the errors are normall distributed. This appears to hold for the bulk of the residuals with some skewness in the tails. This is shown in the significant return on the shapiro test. The model should not be used when predicting crime rate for counties with extreme criminal justice effectiveness or wages.

To summarize the value of model 1 we found a strong predictor in the form of criminal justice effectiveness while wages are not good predictors.

```
#cov(resid(mod1), dfCrime$allWages)
#cov(resid(mod1), log(dfCrime$crimJustEff))
mean(resid(mod1))
```

```
[1] -5.23828e-18
```

3.2 Model 2

3.2.1 Introduction

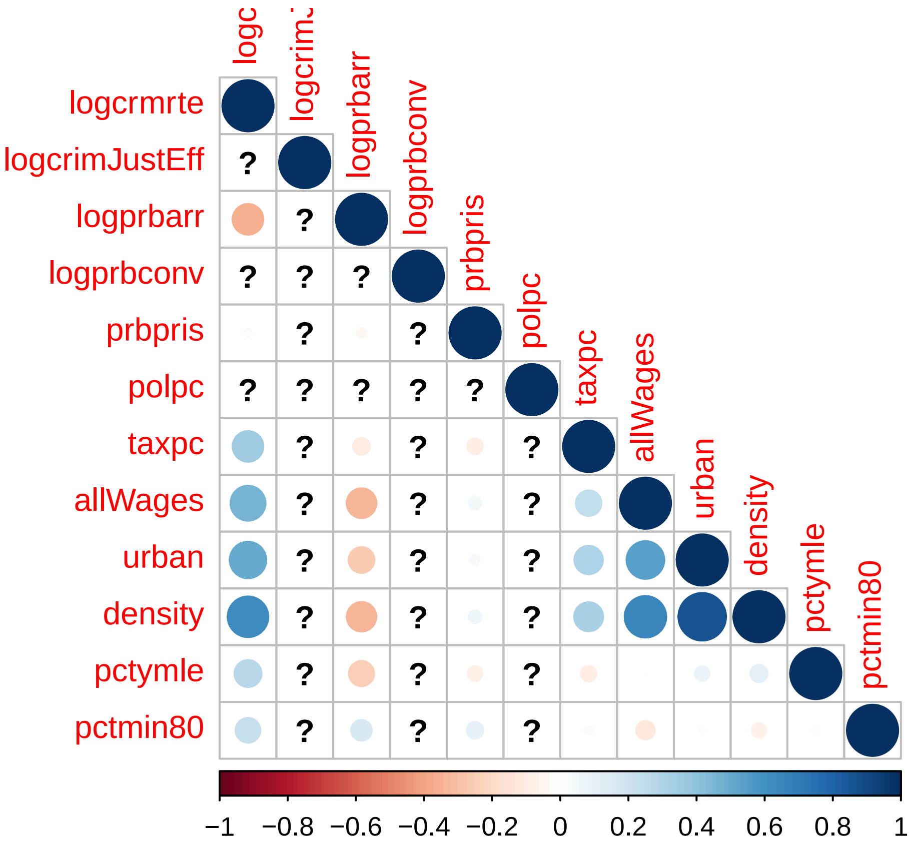
In this model, we introduce the additional covariates of population per square mile (density), tax per capita (taxpc) and police per capita (polpc) to increase the accuracy of our regression. We are including these additional variables to our second model, as they add accuracy to the explanatory variables used in our first model:

1. The **Density** of an area can have significant impacts on:
   - **Criminal Justice Effectiveness**: with more people in a given area, crime frequency increases (+ bias direction). However, more people means there are more potential witnesses, making it easier to catch criminals (- bias direction).
   - **Economic Opportunity (ie. AllWages)**: in high density areas, there is an increase in demand for support services such as food, retail, utilities, etc. As a result, there is a high demand for service jobs, which increases the economic opportunities within the area (+ bias direction). However, more people in a given area, there is a closer proximity to drugs, alcohol and gang violence - all of which are inhimitors to better economic outcomes.
2. The **Police Per Capita** in a county can be influential on the Criminal Justice Effectiveness. With more police in a given area, one would think that crime rates would decrease, however our correlation plot below tells a different story. Including this variable in our analysis will give us more insight into the variables used in model 1.
3. The **Tax Per Capita** can have a direct impact on the Police Per Capita. A higher tax per capita, means that the county has more tax dollars to spend on protection services (ie. increasing the number of police in the county).

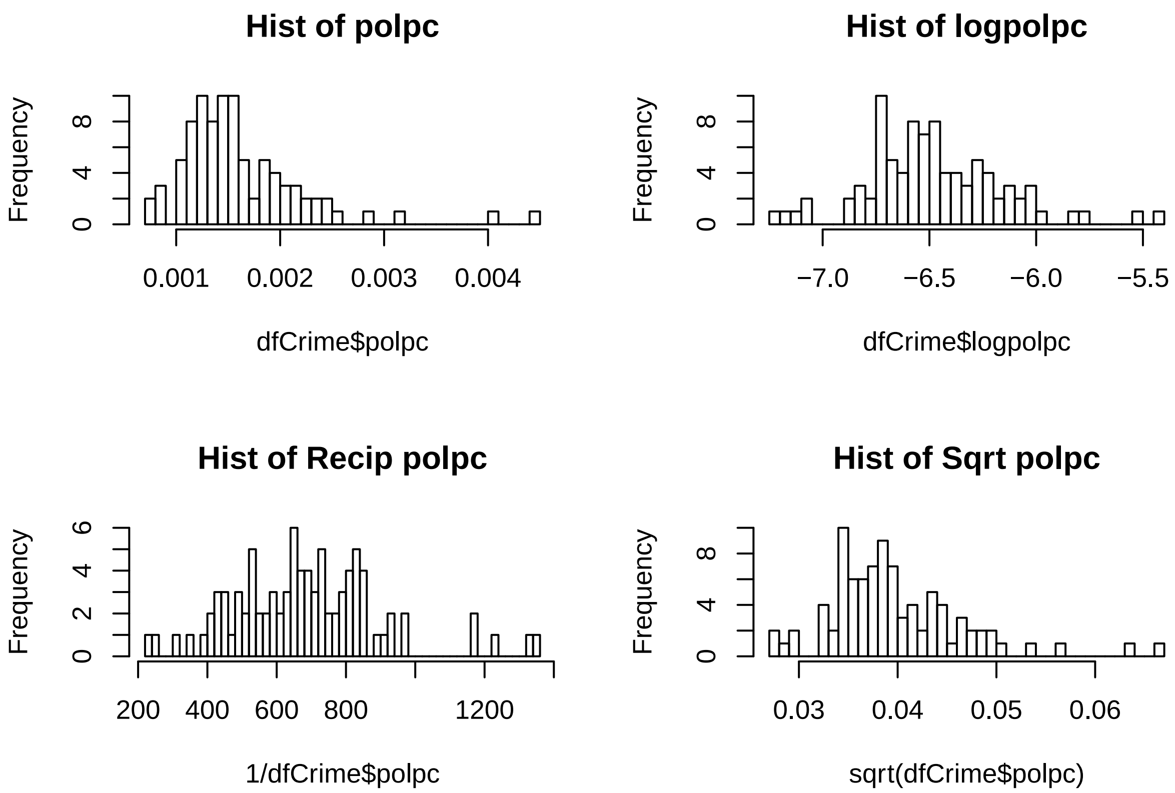
$$log(crmrate) = \beta_0 + \beta_1 crimjusteff + \beta_2 log(polpc) + \beta_3 density + \beta_4 allWages + \beta_5 taxpc + u$$

3.2.2 Model 2 EDA and Data Transformations

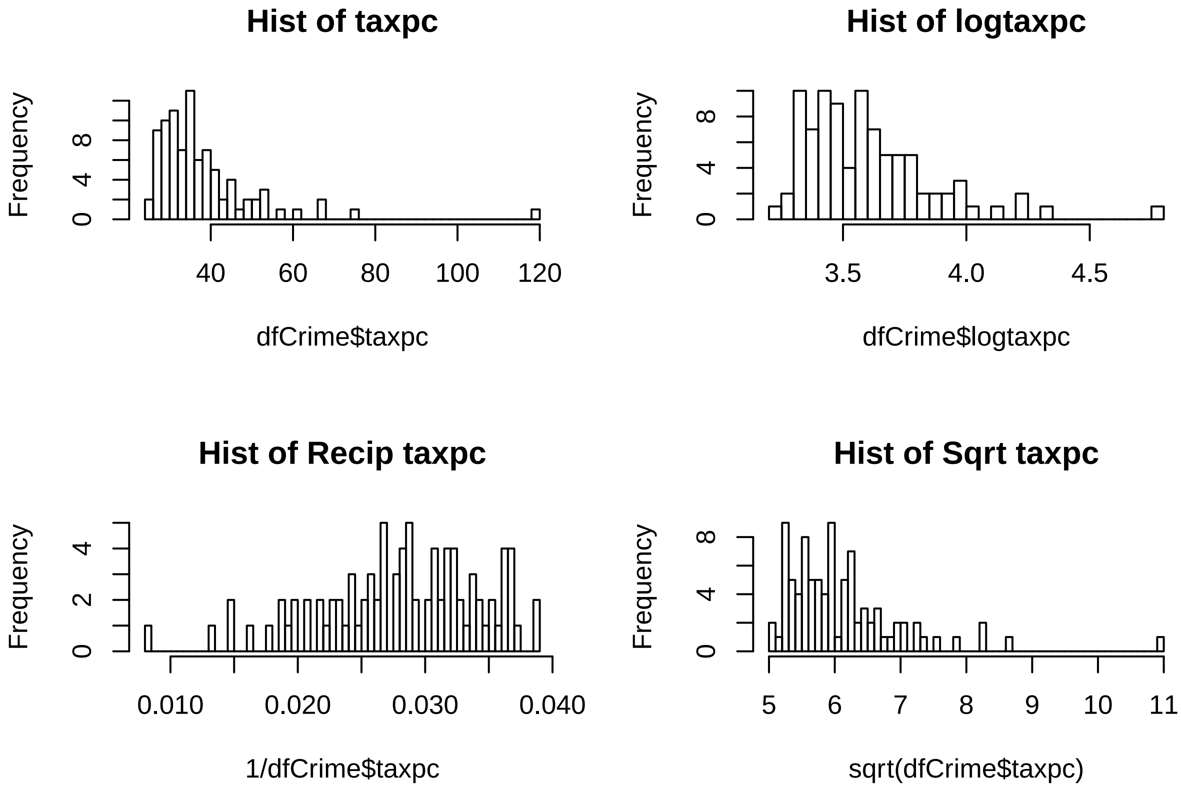
```
corrplot(cor(dfCrime[,c("logcrmrte", "logcrimJustEff", "logprbarr", "logprbconv",
                        "prbpris", "polpc", "taxpc", "allWages", "urban", "density",
                        "pctymle", "pctmin80")]),method='circle', type = 'lower')
```



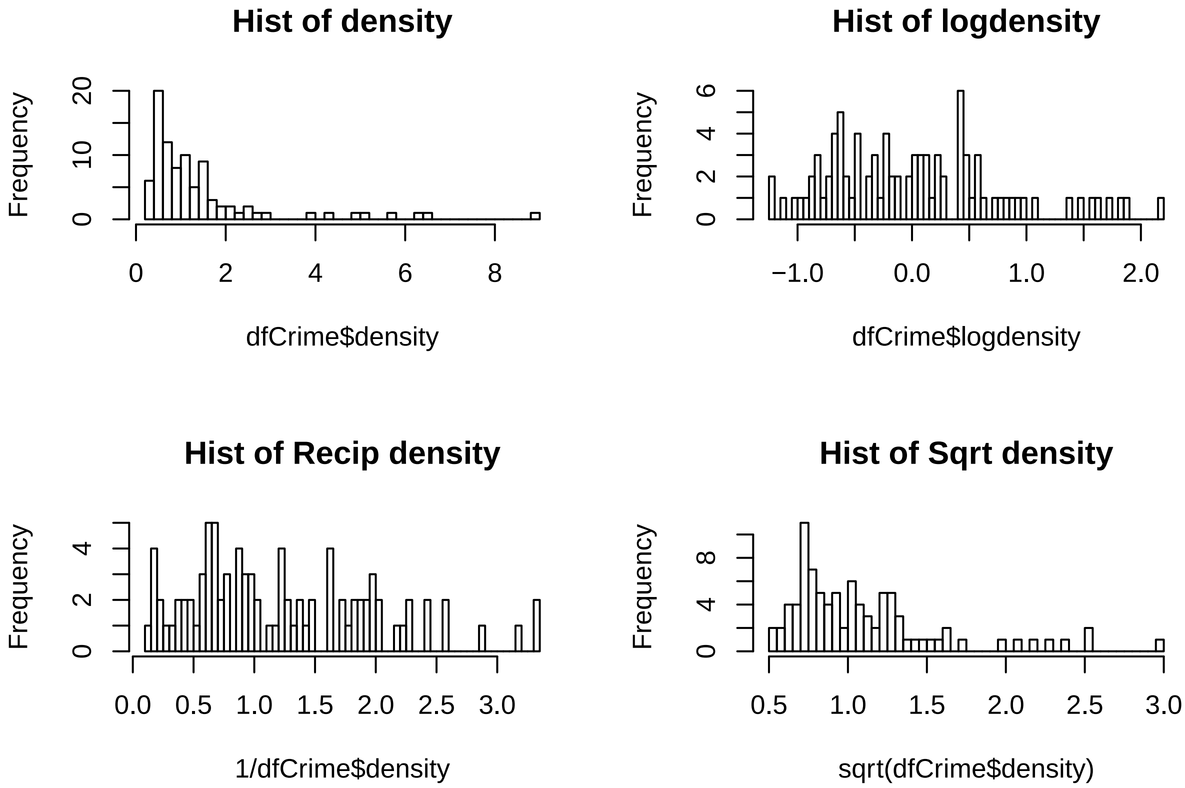
```
# polpc transformation analysis
par(mfrow = c(2,2))
hist(dfCrime$polpc, main="Hist of polpc", breaks=50)
hist(dfCrime$logpolpc, main="Hist of logpolpc", breaks=50)
hist(1/dfCrime$polpc, main="Hist of Recip polpc", breaks=50)
hist(sqrt(dfCrime$polpc), main="Hist of Sqrt polpc", breaks=50)
```



```
# taxpc transformation analysis
par(mfrow = c(2,2))
hist(dfCrime$taxpc, main="Hist of taxpc", breaks=50)
hist(dfCrime$logtaxpc, main="Hist of logtaxpc",breaks=50)
hist(1/dfCrime$taxpc, main="Hist of Recip taxpc", breaks=50)
hist(sqrt(dfCrime$taxpc), main="Hist of Sqrt taxpc", breaks=50)
```



```
# density transformation analysis
par(mfrow = c(2,2))
hist(dfCrime$density, main="Hist of density", breaks=50)
hist(dfCrime$logdensity, main="Hist of logdensity",breaks=50)
hist(1/dfCrime$density, main="Hist of Recip density", breaks=50)
hist(sqrt(dfCrime$density), main="Hist of Sqrt density", breaks=50)
```



```
# par(mfrow = c(2,2))
# plot(dfCrime$logcrimJustEff, dfCrime$polpc, main = 'polpc vs logcrimJustEff', xlab='logcrimJustEff', y
# plot(dfCrime$logcrimJustEff, dfCrime$logpolpc, main = 'logpolpc vs logcrimJustEff', xlab='logcrimJust
# plot(dfCrime$logcrimJustEff, dfCrime$taxpc, main = 'taxpc vs logcrimJustEff', xlab='logcrimJustEff', y
# plot(dfCrime$logcrimJustEff, dfCrime$logtaxpc, main = 'logtaxpc vs logcrimJustEff', xlab='logcrimJust
```

– AXLB - WIP

In the histograms above, we see that the both polpc and taxpc exhibit right skew. Taking the natural log of polpc brings the distribution closer to normal. However, the *log* of taxpc and density makes the distributions even more skewed.

As a result, we will use the *log* of polpc (logpolpc) in our second model and will not transform the taxpc and density variables.

3.2.3 Model 2 Linear Model

```
model2 <- lm(logcrmrte ~ logcrimJustEff + logpolpc + log(allWages) + logtaxpc + sqrt(dfCrime$density), d
model2
```

```

Call:
lm(formula = logcrmrte ~ logcrimJustEff + logpolpc + log(allWages) +
    logtaxpc + sqrt(dfCrime$density), data = dfCrime)

Coefficients:
          (Intercept)         logcrimJustEff               logpolpc
             -6.47921               -0.25967                0.33279
```

```
     log(allWages)              logtaxpc  sqrt(dfCrime$density)
           0.50075               0.03552                0.40521
```

– cooks distance analysis –

Model 2 CLM Assumptions:

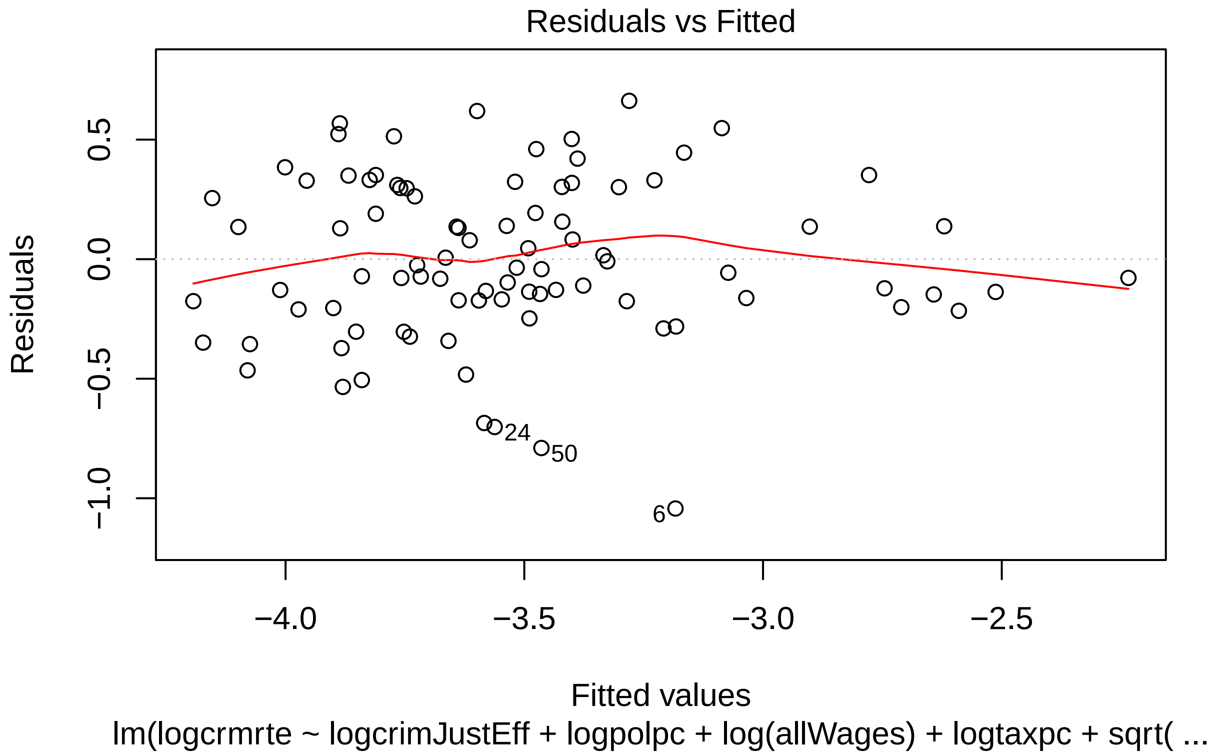
- **MLR1** Discussed above.
- **MLR2** Discussed above.
- **MLR3: Non-perfect Collinearity** We will use the VIF function to provide evidence that our variables in model2 are not perfectly multicollinear. As we can see from the VIF results, below, all of the variables' values are less than five, which allows us to conclude model2 is free from multicollinearity.

```
vif(model2)
```

```
##          logcrimJustEff              logpolpc         log(allWages)
##                1.378627              1.663614              2.006571
##                logtaxpc sqrt(dfCrime$density)
##                1.358061              2.385209
```

- **MLR4: Zero Conditional Mean** The residual vs. fitted chart, below, gives us evidence that we meet the zero conditional mean assumption as the majority of the residual means lie close to zero. The exceptions to this trend, lie on the right side of the chart where there are fewer data points (evidence for heteroscedasticity - see MLR5, below).

```
plot(model2, which=1)
```



- **MLR5: Homoscedasticity** The above Residuals vs Fitted graph provides evidence of heteroscedasticity as right side of the chart have fewer datepoints. To provide further evidence of heteroscedasticity, we will use the White test to generate coefficients with robust with vcovHC

non-significan coeffs (pr(>t)) > .05...make a model to remove all the non-significant and then analyze using the f-test.

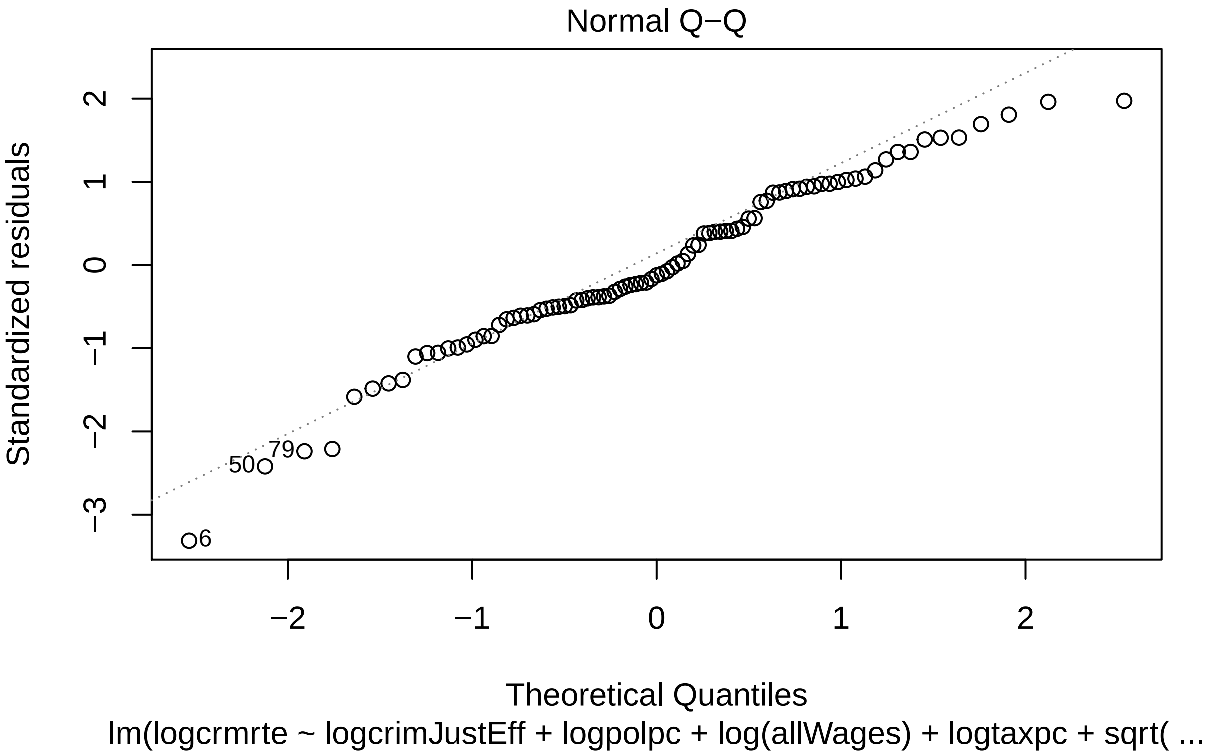
– AXLB - WIP

```
coeftest(model2, vcov=vcovHC)
```

```

t test of coefficients:

                        Estimate Std. Error t value Pr(>|t|)
(Intercept)            -6.479213   5.863983 -1.1049 0.272388
logcrimJustEff         -0.259675   0.125841 -2.0635 0.042189 *
logpolpc                0.332786   0.233732  1.4238 0.158255
log(allWages)           0.500752   0.596805  0.8391 0.403849
logtaxpc                0.035525   0.231771  0.1533 0.878553
sqrt(dfCrime$density)  0.405213   0.133994  3.0241 0.003318 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- **MLR6: Normal Distribution of Errors** The Normal Q-Q plot, below, provides evidence that our residuals follow a normal distribution. While there are some data points on the left and right side of the graph that stray from the diagonal line, since our data set has over 30 datapoints, per the CLT, we can assume residuals have a normal distribution.

```
plot(model2, which=2)
```



```
# hist(model2$residuals)
# shapiro.test(model2$residuals)
#null hypothesis: residuals drawn from population with a normal distribution.
#small p-value tells you if you can reject the null hypothesis.
#this test depends on sample size, it does not take very much deviation from normality for
#us to get a statistically significant result
```

```
summary(model2)
```

```
##
## Call:
## lm(formula = logcrmrte ~ logcrimJustEff + logpolpc + log(allWages) +
##     logtaxpc + sqrt(dfCrime$density), data = dfCrime)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.04290 -0.17570 -0.04186  0.29765  0.66230
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           -6.47921    4.09171  -1.583 0.117112
## logcrimJustEff        -0.25967    0.07743  -3.354 0.001203 **
## logpolpc               0.33279    0.14594   2.280 0.025150 *
## log(allWages)          0.50075    0.48309   1.037 0.302949
## logtaxpc               0.03552    0.16220   0.219 0.827177
## sqrt(dfCrime$density)  0.40521    0.11631   3.484 0.000792 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.346 on 83 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.5864, Adjusted R-squared:  0.5615
## F-statistic: 23.53 on 5 and 83 DF,  p-value: 1.189e-14
```

The Adjusted R-squared variable penalizes for additional variables, which means there is a chance that this value will decrease if the added variables do not contribute to the model. By comparing the Adjusted R-squared value between our first and second models, we see that log(polpc), taxpc and density help describe log(crmrate). Our second model has an Adjusted R-squared value of 0.5004, which means 50.04% of the variation in the log_{10} of crime rate is explained by the explanatory variables used in this model. This is a significant increase compared to our first model, that has an Adjusted R-squared value of 0.4520.

In addition, the F-statistic is 16.62 with a statistically significant p-value of < 6.263e-11. As a result, we reject the null hypothesis that none of the independent variables help to describe log(crmrate).

Coefficient Analysis (assuming ceterus paribus): - logcrimJustEff: -0.1607. This suggests that for a 1% increase in criminal justice efficiency, there is a 0.1607% decrease in crime rate. - logpolpc: 0.3701. This suggests that for a 1% increase in police per capita, there is a 0.3701% increase in crime rate. - allWages: 0.00006692. This suggests that for a 1% increase in total average weekly wage, there is a 0.0067% increase in crime rate. - taxpc: -0.001632. This suggests that for a 1% increase in tax per capita, there is a 0.1632% decrease in crime rate. - density: 0.06259. This suggests that for a 1% increase in density, there is a 6.259% increase in crime rate.

3.2.4 Conclusion : Are the conclusions they draw based on this evaluation appropriate? Did the team interpret the results in terms of their research question?

Compared to model 1, the adjusted R^2 of model 2 is only marginally higher. This suggests that we should continue our analysis by focusing on the join significance of the variables added in model 2.

3.3 Model 3

3.3.1 Discussion of Variables

Despite the improvements in the accuracy of model 2 over model 1, we are still only explaining about 55% of the variation in our data. As a result, we propose to also analyse the topic of demographics which could have an effect on both of our key explanatory variables.

One key component of demographics is the race of the county inhabitants and how they are perceived and treated by others, especially for minorities in the population. For example, systemic racism could have an important effect on: * Criminal Justice Effectiveness: If police, lawyers and judges are racially biased, this could lead to more arrests and more convictions regardless of the strength of the legal case and the evidence. As a result, we hypothesize the crime rate would increase. * Economic Opportunity: Racism could prohibit members of the minority from having access to education, jobs and higher wages. Racism could also limit access to healthcare and social programmes which has a negative effect on economic opportunity.

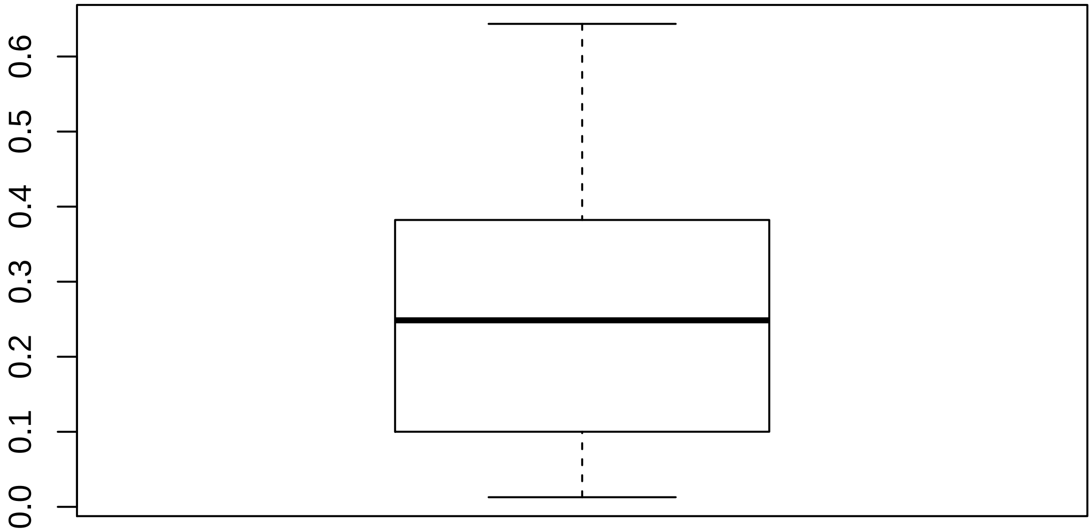
However, since we cannot directly measure racism, we have to operationalize this covariate by examining its effect in the real world. We propose to use the variable pctmin80, which represents the percentage of minorities in the population of the county. This is also a continous parameter and so given a higher the percentage of minorities, we should expect to see a greater effect.

From the summary and boxplot below, we can see that the percentage of minorities ranges from 0.0154 - 0.6435, with a mean of 0.2621. We note that there are no major outliers. We will apply the natural log to the variable pctmin80 to 1) make it easier for us to interpret the coefficient in our linear model and 2) to better expose the linear relationship in the model.

```
summary(dfCrime$pctmin80)
```

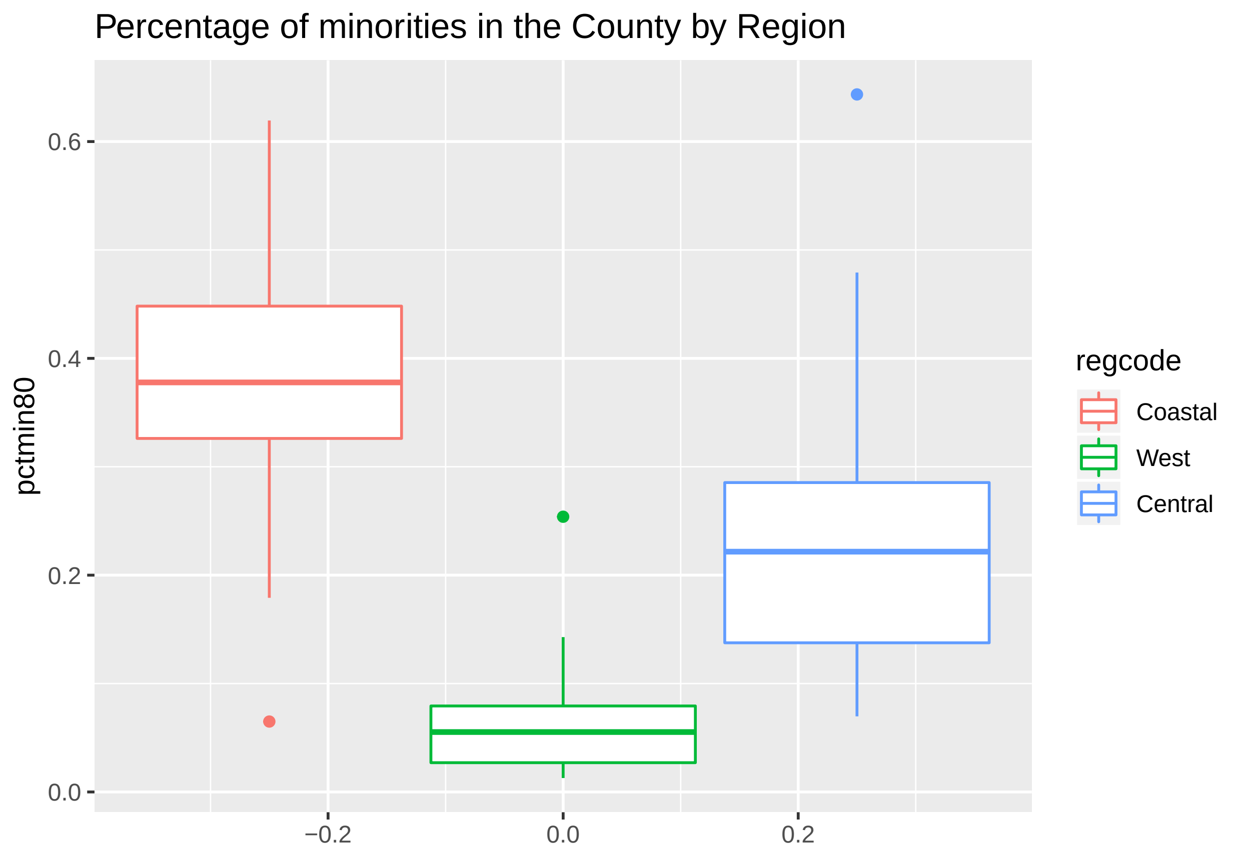
```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
0.01284 0.10024 0.24852 0.25713 0.38183 0.64348 
```

```
boxplot(dfCrime$pctmin80)
```



In addition, we hypothesize that the location of the county can have an impact on demographcs. We can see that counties in the West have a significantly lower percentage of minorities than the other two regions. As a result, we will also test for region in our model.

```
ggplot(data = dfCrime, aes(y = pctmin80, color=regcode)) +
      geom_boxplot() + ggtitle("Percentage of minorities in the County by Region")
```



UPDATE AFTER ALEXA Given that our analysis of model 2 showed that taxpc was not statistically significant, we will not consider it in our model.

We have also chosen not to include other variables from our dataset in our model: * Urban: We believe the variable "density" better explains the same effects as "urban", while also being a linear parameter. In addition, there may be data points that failed to meet the cutoff for being defined as urban, but may still see the same effects as being urban and hence may distort our analysis. * Age and Gender: While age and gender are important demographic variables, the only variable in our dataset is pctymle which provides the percentage of young males in the population. However, given that this variable encompasses both male and young, we may not be able to discern if age or gender has the larger effect (if any at all). * Judgement: We chose not to include the varibles concerning the probability of a prison sentence as well as the average sentence as we believe it is unlikely that potential criminals would have good access to this information. In addition, local county officials have limited influence over the decisions of the judiciary system, as they are separate branches of government.

Our equation for model 3 is as follows:

$$log(crmrate) = \beta_0 + \beta_1 log(crimjusteff) + \beta_2 log(polpc) + \beta_3 density + \beta_4 log(allWages) + \beta_5 logpctmin80 + \beta_6 west + \beta_7 central + u$$

3.3.2 Model 3 Linear Model

```
dfCrime$logunweightedavg<-log(dfCrime$unweighted_avg_wage)
model3_initial<-lm(logcrmrte ~ logcrimJustEff + logpolpc + logunweightedavg +  density + logpctmin80 + w
summary(model3_initial)
```

```

Call:
lm(formula = logcrmrte ~ logcrimJustEff + logpolpc + logunweightedavg + 
    density + logpctmin80 + west + central, data = dfCrime)
```

```
Residuals:
     Min       1Q   Median       3Q      Max
-0.88805 -0.14672  0.05094  0.14252  0.59964

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)      -8.25258    2.41603  -3.416 0.000997 ***
logcrimJustEff   -0.34825    0.06551  -5.316 9.17e-07 ***
logpolpc          0.33920    0.10682   3.175 0.002117 **
logunweightedavg  1.11492    0.37410   2.980 0.003801 **
density           0.08362    0.02842   2.943 0.004241 **
logpctmin80       0.14324    0.05640   2.539 0.013015 *
west             -0.26442    0.13219  -2.000 0.048807 *
central          -0.17853    0.07786  -2.293 0.024448 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2749 on 81 degrees of freedom
  (1 observation deleted due to missingness)
Multiple R-squared:  0.7452,    Adjusted R-squared:  0.7231
F-statistic: 33.84 on 7 and 81 DF,  p-value: < 2.2e-16
```

We note from the two F-tests below that west and central were not statistically significant to the model, but the inclusion of logpctmin80 is highly significant. An analysis of VIF(model3) also indicates that logpctmin80 and west have higher variance inflation factors as compared to the others. It appears then that logpctmin80 better explains the difference in demographics than west or central, and we thus remove the latter 2 variables from our model.

```
linearHypothesis(model3_initial,c("west=0","central=0"), vcov=vcovHC)
```

```
## Linear hypothesis test
##
## Hypothesis:
## west = 0
## central = 0
##
## Model 1: restricted model
## Model 2: logcrmrte ~ logcrimJustEff + logpolpc + logunweightedavg + density +
##     logpctmin80 + west + central
##
## Note: Coefficient covariance matrix supplied.
##
##   Res.Df Df      F Pr(>F)
## 1     83
## 2     81  2 1.8163 0.1692
```

```
linearHypothesis(model3_initial,c("west=0","central=0","logpctmin80=0"), vcov=vcovHC)
```

```
## Linear hypothesis test
##
## Hypothesis:
## west = 0
## central = 0
## logpctmin80 = 0
##
```

```
## Model 1: restricted model
## Model 2: logcrmrte ~ logcrimJustEff + logpolpc + logunweightedavg + density +
##     logpctmin80 + west + central
##
## Note: Coefficient covariance matrix supplied.
##
##   Res.Df Df      F    Pr(>F)
## 1     84
## 2     81  3 14.624 1.069e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
vif(model3_initial)
```

```
##   logcrimJustEff         logpolpc logunweightedavg          density
##         1.563381         1.412016         1.906089         2.172243
##      logpctmin80             west          central
##         3.144130         3.583877         1.685033
```

Our revised equation for model 3 is as follows:

$log(crmrate) = \beta_0 + \beta_1 log(crimjusteff) + \beta_2 log(polpc) + \beta_3 density + \beta_4 log(allWages) + \beta_5 logpctmin80 + u$

```
model3<-lm(logcrmrte ~ logcrimJustEff + logpolpc + logunweightedavg +  density + logpctmin80, data = dfC
model3
```

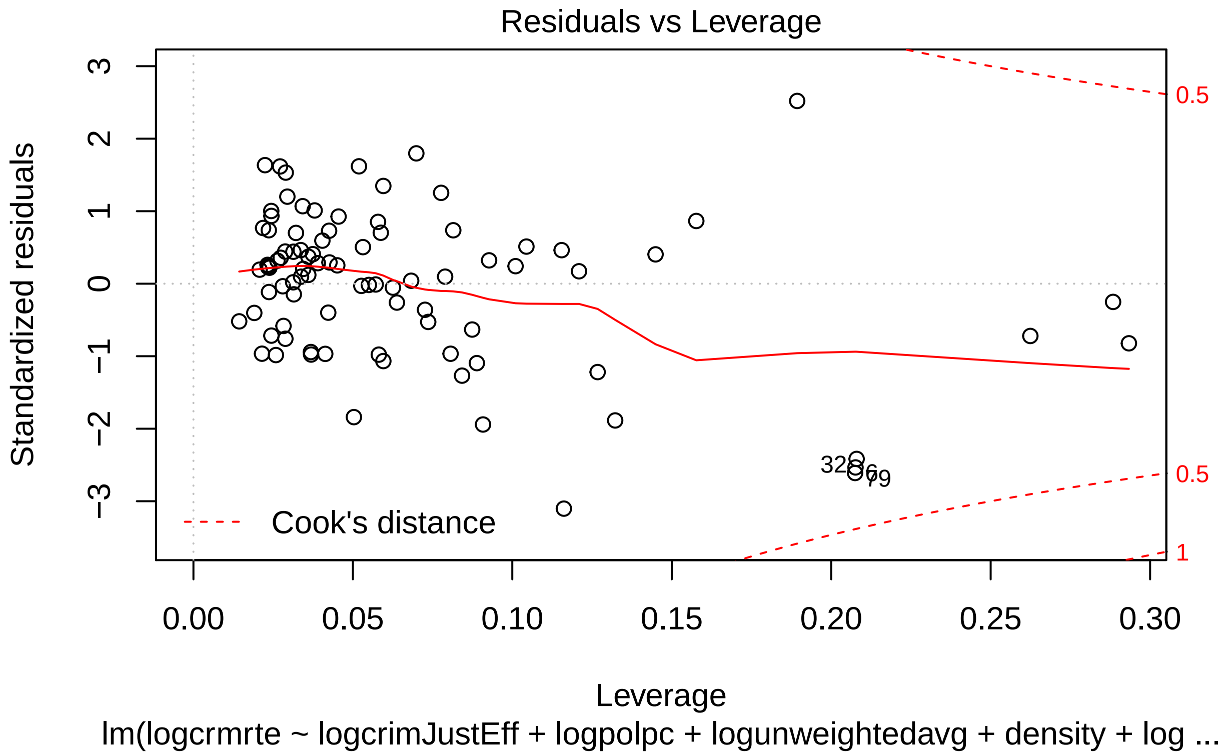
```

Call:
lm(formula = logcrmrte ~ logcrimJustEff + logpolpc + logunweightedavg +
    density + logpctmin80, data = dfCrime)

Coefficients:
     (Intercept)    logcrimJustEff          logpolpc  logunweightedavg
        -7.62186          -0.38557           0.33980           0.99872
         density       logpctmin80
         0.07288           0.23065
```

```
summary(model3)$adj.r.square
```

```
[1] 0.7092636
```

From the Residuals vs Leverage plot below, we also note that there are no major outliers that have significant influence on our model (no points have a cook's distance > 0.5).

```
plot(model3,which=5)
```



Model 3 CLM Assumptions:

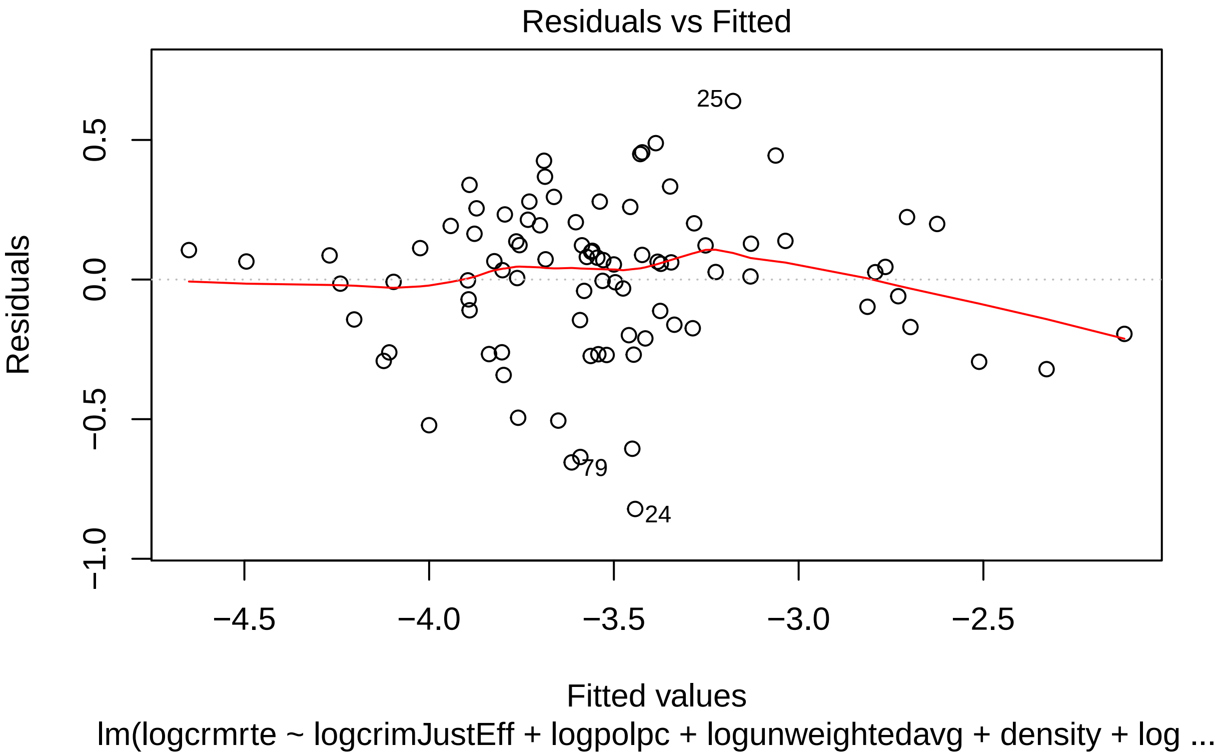
- **MLR1 and 2**: Discussed earlier.
- **MLR3** No perfect multicollinearity: We demonstrate that our independent variables are not perfectly multicolinear using the VIF function, and note that all of our variance inflation factors are less than 5.

```
vif(model3)
```

```
##   logcrimJustEff         logpolpc logunweightedavg          density
##         1.434962         1.370508         1.790625         2.104673
##      logpctmin80
##         1.069661
```

- **MLR4'** Zero Conditional Mean: From the residual vs. fitted chart below, we see that the mean of the residuals mostly lie along 0, except towards the left side of our chart where there are fewer data points. We can reasonably conclude that we satisfy MLR4.

```
plot(model3, which = 1)
```



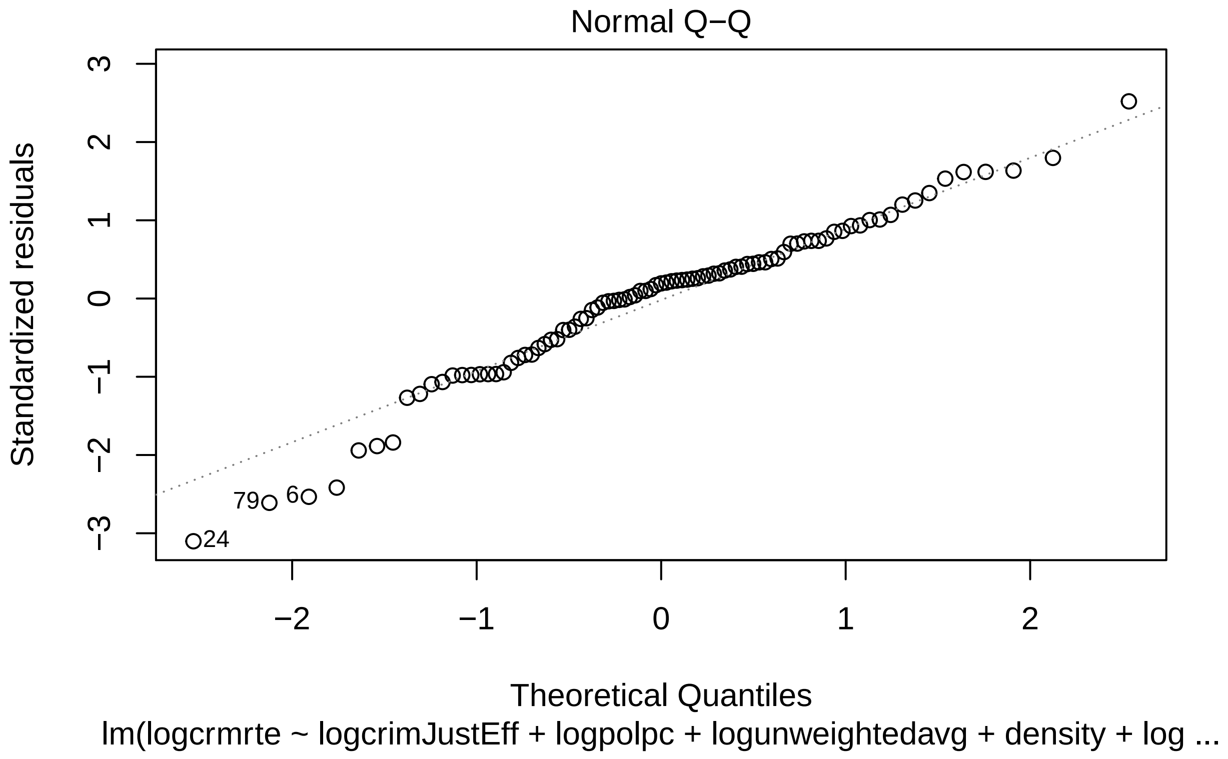
- **MLR5’** Spherical errors: We note from the residuals vs fitted chart above that we have some evidence of heteroscedasticity, since there are less datapoints on both the left and right of the chart. As a result, we use the vcovHC method to estimate a robust variance-covariance matrix using White and Huber’s method and generate coefficients that are robust to heteroscedasticity.

```
coeftest(model3, vcov=vcovHC)
```

```

t test of coefficients:

                  Estimate Std. Error t value  Pr(>|t|)    
(Intercept)      -7.621859   4.485588 -1.6992 0.0930291 .  
logcrimJustEff   -0.385566   0.094619 -4.0749 0.0001048 ***
logpolpc          0.339799   0.195159  1.7411 0.0853644 .  
logunweightedavg  0.998724   0.635210  1.5723 0.1196913    
density           0.072882   0.032747  2.2256 0.0287516 *  
logpctmin80       0.230646   0.038722  5.9564 5.988e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- **MLR6’** Normality of errors: From the qqplot below, we see that the residuals in our model follow a fairly normal distribution. In addition, since we have a large sample size of 90 datapoints, we can rely on a version of the central limit theorem to assume normally distributed errors.

```
plot(model3,which=2)
```



By satisfying these assumptions, we can expect that our coefficients are approaching the true parameter values in probability.

3.3.3 Analysis

The model shows a good fit, with an adjusted R-squared of 0.72, meaning that the model explains 72% of the variation in crime.

After accounting for coefficients that are robust to heteroscedasticity, we note only three them have statistical significance at the 95% level or better. These are criminal justice efficiency, minority percentages and density.

Interpretation of coefficients (Assuming ceterus paribus):

Positive coefficients: * Police presence: If we increase police per capita by 1 percent, we expect the crime rate to increase by 0.28%. * AllWages: If we increase wages by 1 percent, we expect the crime rate to increase by 0.94% * Density: If we increase density by 1 person per square mile, we expect the crime rate to increase by 8% * Percentage of minorities: If the percentage of minorities increase by 1%, we expect the crime rate to increase by 0.24%

Negative coefficients: * Criminal justice efficiency: If we increase the criminal justice efficiency by 1%, we expect the crime rate to decrease by 0.42%.

3.3.4 Results:

3.4 Comparison of Regression Models

***Can anyone figure out why logcrimJustEff is on 2 lines?**

```
stargazer(mod1,model2,model3,type="text")
```

```
========================================================================================
                                            Dependent variable:
                    --------------------------------------------------------------------
                          logcrmrte                          logcrmrte
                             (1)                    (2)                    (3)
----------------------------------------------------------------------------------------
unweighted_avg_wage        0.006***
                           (0.001)

logcrimJustEff            -0.434***
                           (0.076)

logcrimJustEff                                   -0.260***              -0.386***
                                                  (0.077)                (0.064)

logpolpc                                          0.333**                0.340***
                                                  (0.146)                (0.108)

log(allWages)                                      0.501
                                                  (0.483)

logtaxpc                                           0.036
                                                  (0.162)

density)                                          0.405***
                                                  (0.116)

logunweightedavg                                                         0.999***
                                                                         (0.372)

density                                                                  0.073**
                                                                         (0.029)

logpctmin80                                                              0.231***
                                                                         (0.034)

Constant                  -6.300***                -6.479               -7.622***
                           (0.374)                (4.092)                (2.394)

----------------------------------------------------------------------------------------
Observations                  89                     89                     89
R2                          0.469                  0.586                  0.726
Adjusted R2                 0.456                  0.561                  0.709
Residual Std. Error    0.385 (df = 86)        0.346 (df = 83)        0.282 (df = 83)
F Statistic         37.906*** (df = 2; 86) 23.532*** (df = 5; 83) 43.936*** (df = 5; 83)
========================================================================================
Note:                                                        *p<0.1; **p<0.05; ***p<0.01
```

Comparing the 3 models, we see that our adjusted R2 value has steadily increased from 0.456-0.732 as we introduce more covariates which indicates that we were able to explain more variation in our model not purely by increasing the number of indepedent variables.

At the same time, our standard errors have decreased **insert more commentary on standard errors**.

We see that by expanding our definitions of criminal justice efficiency and economic opportunity between model 1 and model 3 lowered the coefficients for logcrimJustEff and allWages. This is most likely because that we were able to better explain the effects with our newer variables.

Comment on practical significance after week 12

4 Conclusion

4.1 Policy Recommendations

Given that across all 3 models, we show that both criminal justice efficiency and tax revenues per capita have negative correlations to crime rate, we propose the policy recommendations below to address these issues. In addition, since minority percentages and density were found to be highly significant in the model 3, we believe our recommendations will be of particularly help to those running for political office in counties with a high percentage of minorities or dense urban populations.

1. Since increasing both criminal justice and tax revenues are negatively correlated, we propose providing more funding for the local justice system.
2. While increasing taxes on constituents may be difficult politically and may cost candidates the ballot, candidates can instead try to attract investment to bring more jobs with higher wages so you can increase revenues.
3. Candidates can also propose to levy taxes on things that could lead to crimes or violence such as alcohol and weapons.
4. Given the significance and relatively large coefficient size of percentage minority, candidates should enroll local law enforcement into bias training.

4.2 Ommitted Variables

Expected correlation between omitted and included variables

Omitted Variable	Crime Rate (B_k)	Criminal Justice Effectiveness	Economic Conditions
Education	-	unknown	+
Social Services	-	unknown	unknown
Unemployment	+	unknown	-
Gang Activity	+	-	-

The 4 major identified ommited variables are shown above.

- Education is an important variable because of demographic insights it provides. First, adults with higher education are less likely to participate in Crime and are more likely to have better economic opportunity. Second, a strong school system is also likely correlated with less youth crime. Because of these expected correlations we are likely overestimating the economic conditions coefficient estimate.
- Available Social Services could also lower crime. Citizens with strong social services support have more options to get help when they lack means for purchasing basic life needs. However this is more difficult to predict, as some social service projects, like homeless shelters, could lead to more criminal activity.
- Unemployment is used as an important indicator of economic health and opportunity. This is would be highly correlated to economic conditions variables like sum of wages. This indicator variable if added to the model would decrease the magnitude of the sum of wage means coefficient estimate.
- Gang or Organized Crime is special case of crime that contains unique causes. It is expected that

it would be negatively correlated with criminal justice effectiveness as large social pressures prevent witnesses from supporting prosecution. Gang crime is also negatively correlated with economic conditions. From these assumed correlations, we can say that criminal justice effectiveness and economic conditions are both underestimated compared to including gang activity operationalized variable in the model.

4.3 Research Recommendations

We have shown in this report 3 different models that seek to explain and model changes in the crime rate in North Carolina in 1980. We start with the fundamental premise that crime is caused by both criminal justice efficiency and economic conditions, and further develop our definition of these two key explanatory variables which each new model.

In Model 3, we were able to explain up to 73% of the variation in our data, and found statistical significance at the 95% level or better for each of our covariates. Of these, we believe that increasing the efficiency of the criminal justice system and tax revenues were the most important, particularly for counties with high density and minority populations. However, our findings should be noted with caution as we were unable to study the effect of several ommitted variables including education, availability of social services, unemployment rates and the presence of organized crime. Had we been able to collect data on these variables and apply them in our model, we believe we could increase accuracy without bias.

5 Appendix

```
options(repr.plot.width=8, repr.plot.height=4)
#myData<-myData[, c("crmrte", "prbarr", "prbconv", "prbpris", "avgsen", "polpc", "density", "taxpc",
#           "pctmin80", "wcon", "wtuc", "wtrd", "wfir", "wser", "wmfg", "wfed", "wsta", "wloc",
#           "mix", "pctymle")]
myData<-dfCrime %>% filter(other==1)
myData<-myData[, c("logcrmrte", "logprbarr", "logprbconv", "logprbpris", "logavgsen", "logpolpc", "logta
           "logpctmin80", "logwcon", "logwtuc", "logwtrd", "logwfir", "logwser", "logwmfg", "logwfed",
           "logmix", "logpctymle")]
r0 <- myData %>% correlate() %>% network_plot(min_cor=.25)
```

```
Correlation method: 'pearson'
Missing treated using: 'pairwise.complete.obs'
```

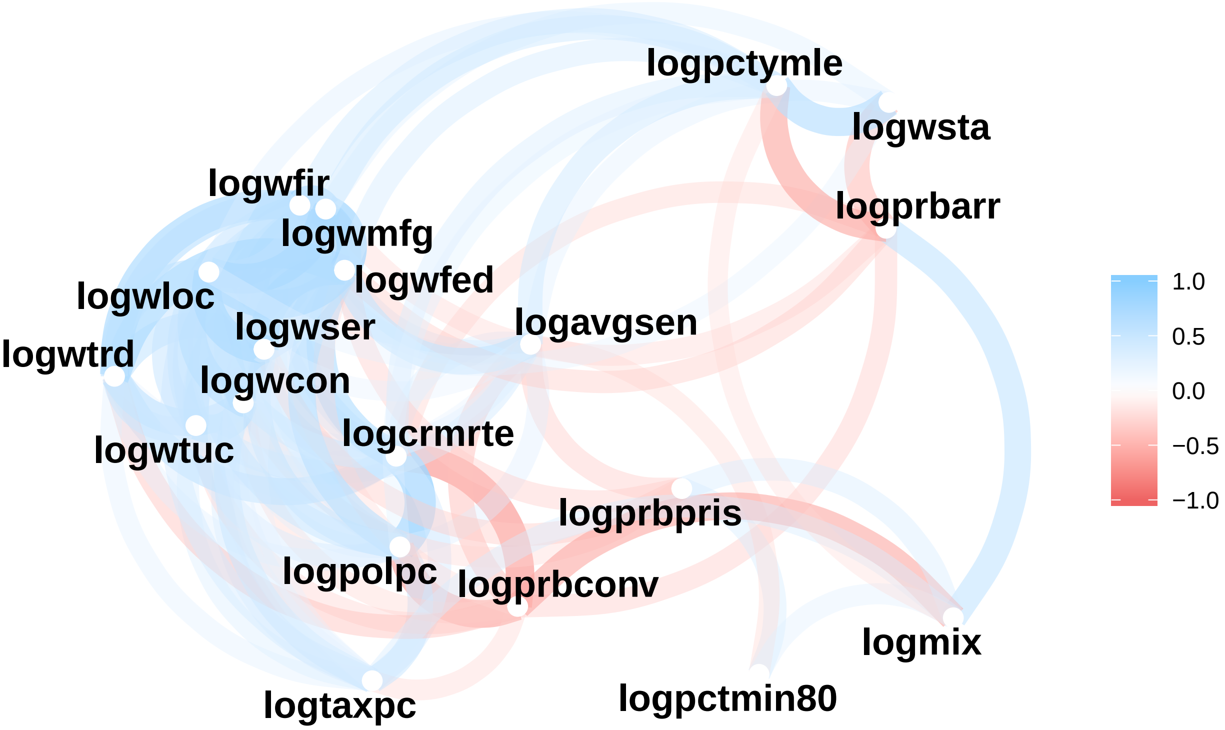
```
myData<-dfCrime %>% filter(central==1)
myData<-myData[, c("logcrmrte", "logprbarr", "logprbconv", "logprbpris", "logavgsen", "logpolpc", "logta
           "logpctmin80", "logwcon", "logwtuc", "logwtrd", "logwfir", "logwser", "logwmfg", "logwfed",
           "logmix", "logpctymle")]
r1 <- myData %>% correlate() %>% network_plot(min_cor=.25)
```

```
Correlation method: 'pearson'
Missing treated using: 'pairwise.complete.obs'
```

```
myData<-dfCrime %>% filter(west==1)
myData<-myData[, c("logcrmrte", "logprbarr", "logprbconv", "logprbpris", "logavgsen", "logpolpc", "logta
           "logpctmin80", "logwcon", "logwtuc", "logwtrd", "logwfir", "logwser", "logwmfg", "logwfed",
           "logmix", "logpctymle")]
r2 <- myData %>% correlate() %>% network_plot(min_cor=.25)
```

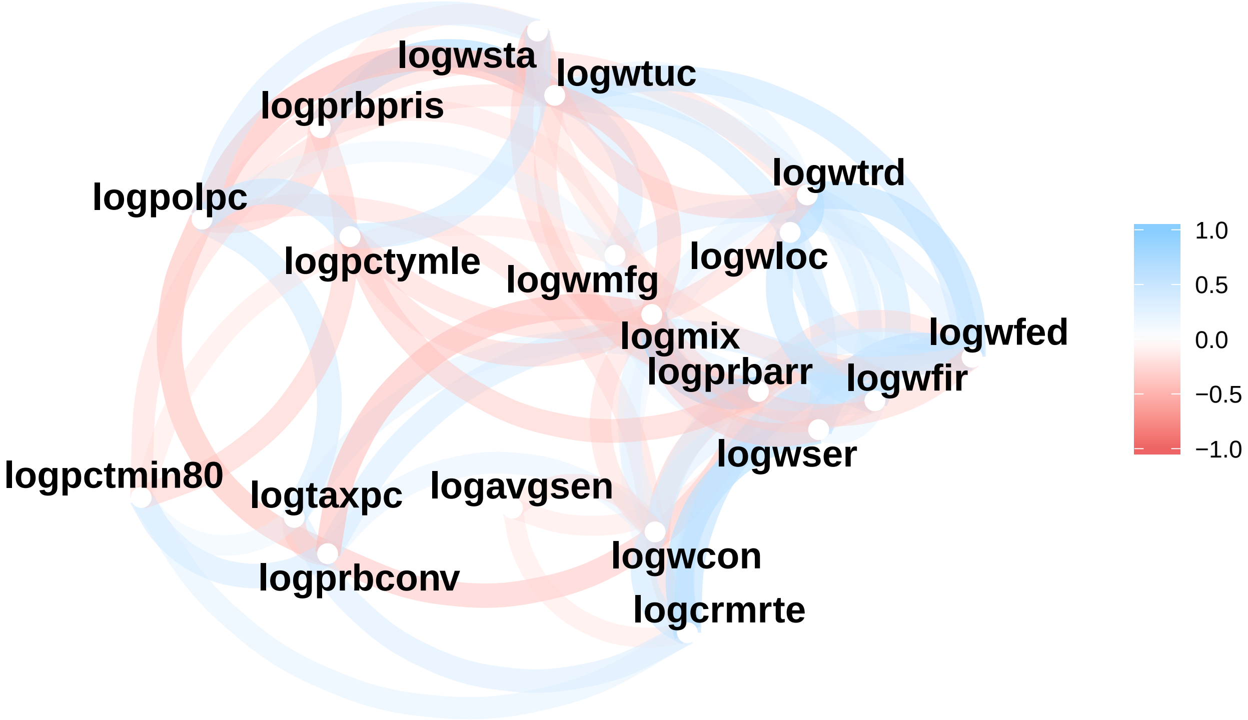
```
Correlation method: 'pearson'
Missing treated using: 'pairwise.complete.obs'
```

```
grid.arrange(arrangeGrob(r1, bottom = 'Central Region Correlation Plot'), ncol=1)
```



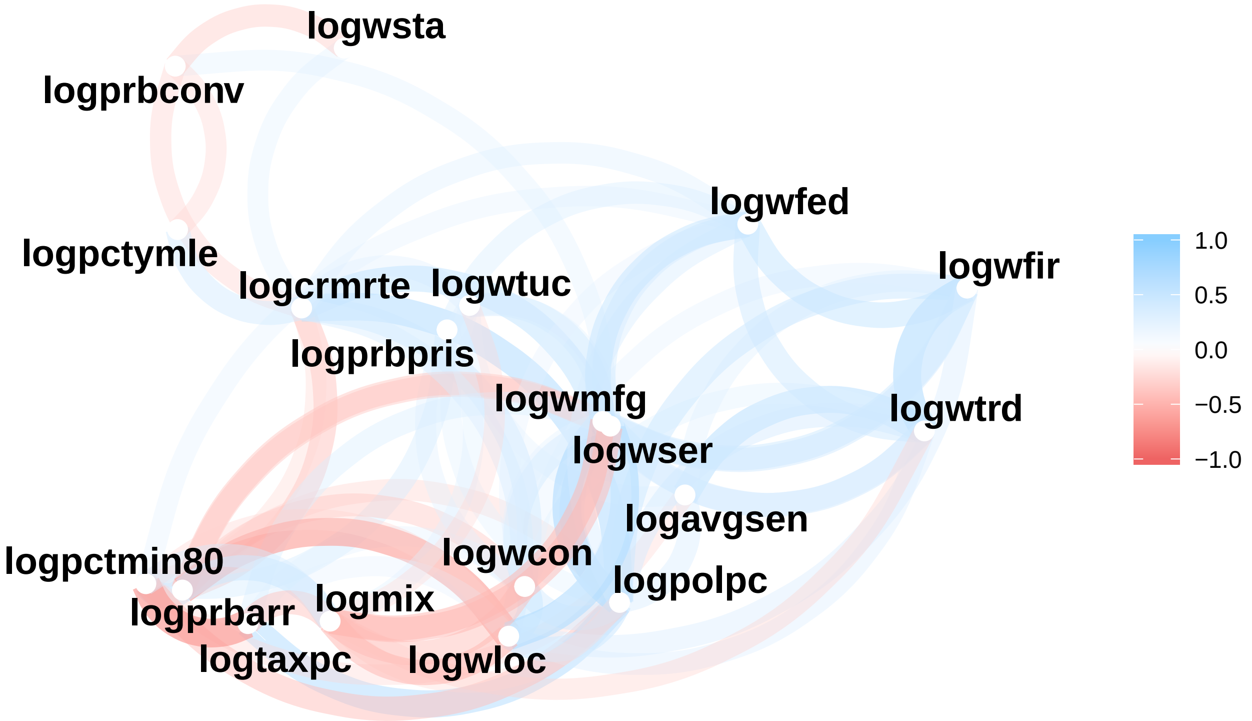
Central Region Correlation Plot

```
grid.arrange(arrangeGrob(r2, bottom = 'Western Region Correlation Plot'), ncol=1)
```



Western Region Correlation Plot

```
grid.arrange(arrangeGrob(r0, bottom = 'Other Region Correlation Plot'), ncol=1)
```



Other Region Correlation Plot

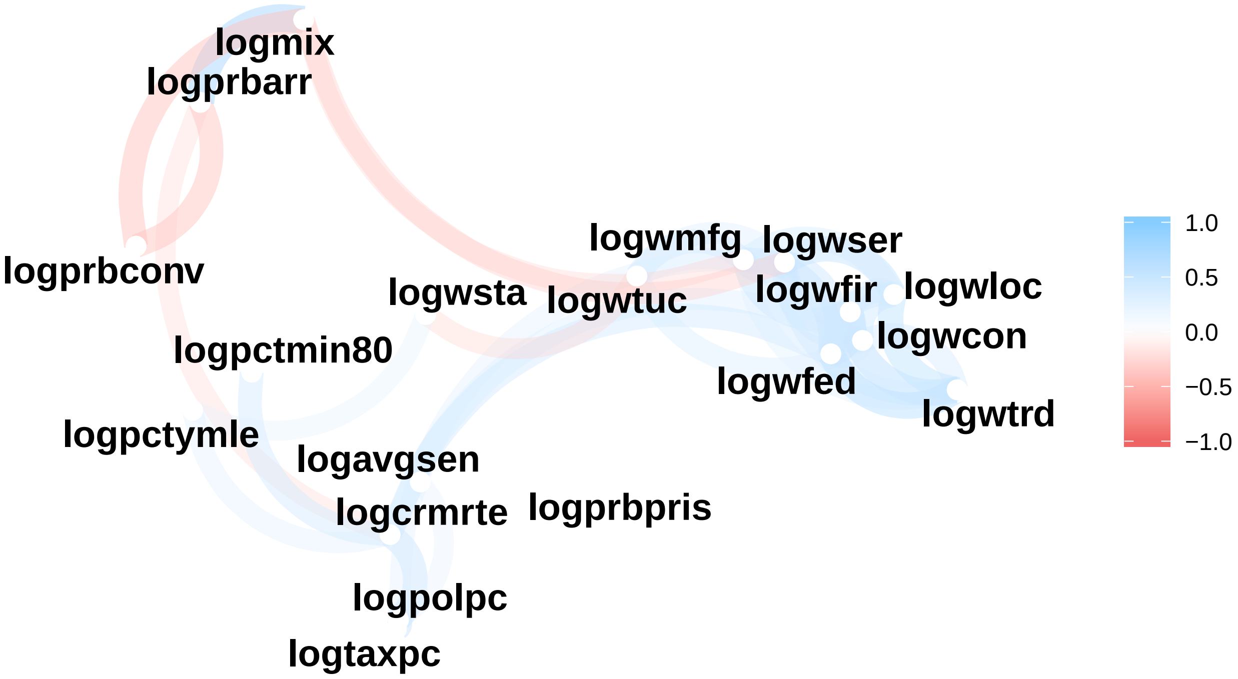
```
myData<-dfCrime %>% filter(urban==0)
myData<-myData[, c("logcrmrte", "logprbarr", "logprbconv", "logprbpris", "logavgsen", "logpolpc", "logta
           "logpctmin80", "logwcon", "logwtuc", "logwtrd", "logwfir", "logwser", "logwmfg", "logwfed",
           "logmix", "logpctymle")]
r0 <- myData %>% correlate() %>% network_plot(min_cor=.25)
```

```

Correlation method: 'pearson'
Missing treated using: 'pairwise.complete.obs'
```

```
myData<-dfCrime %>% filter(urban==1)
myData<-myData[, c("logcrmrte", "logprbarr", "logprbconv", "logprbpris", "logavgsen", "logpolpc",  "log
           "logpctmin80", "logwcon", "logwtuc", "logwtrd", "logwfir", "logwser", "logwmfg", "logwfed",
           "logmix", "logpctymle")]
r1 <- myData %>% correlate() %>% network_plot(min_cor=.25)
```

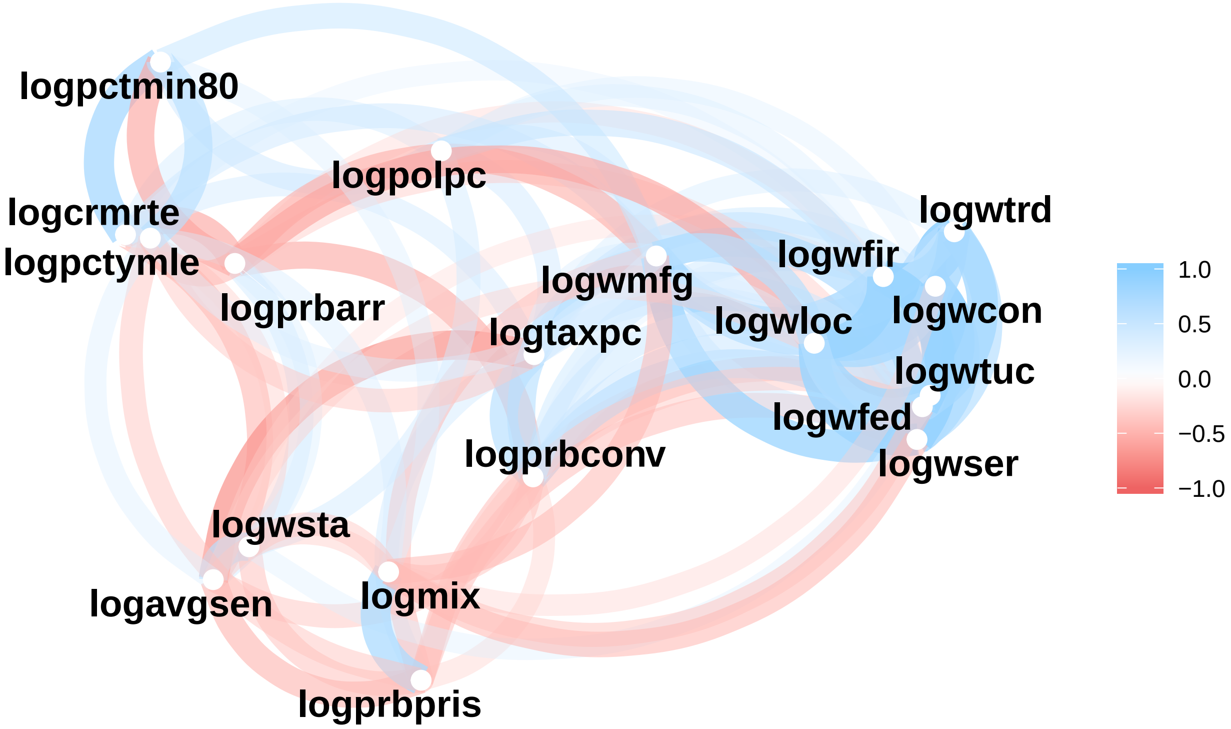
```

Correlation method: 'pearson'
Missing treated using: 'pairwise.complete.obs'
```

```
grid.arrange(arrangeGrob(r0, bottom = 'Non-Urban Correlation Plot'), ncol=1)
```



Non−Urban Correlation Plot

```
grid.arrange(arrangeGrob(r1, bottom = 'Urban Correlation Plot'), ncol=1)
```



Urban Correlation Plot