Lab 3

W203 Statistics for Data Science

Section 6 - Team 1 - Stone Jiang, Gabriela May Lagunes, Indrani Bose

Introduction

Initial Data Cleaning and Exploratory Analysis

For this project, the variable of interests (independent variable) is crime rate (crmrte). This is because being able to model crime rate can indicate policy makers which metrics should they focus on in order to improve the level of security of their states. Therefore, the goal of the developed models is to find the best possible causal predictors for crime rate.

Before choosing the best dependent variables for our models, the data was cleaned as follows. First, we omitted the last rows of the csv since they do not contain data. Second, we eliminated duplicated entries. Here there was just one county (193) which was repeated twice. Then, we verified the datatype of our variables. Here, the prbconv variable was the only non-numerical variable because it was stored as a factor due to omitted rows having non-numerical values. This was converted to numerical.

```
library(dplyr)
library(ggplot2)
library(tidyr)
library(caret)
library(MASS)
select <- dplyr::select # Unmask select from dplyr
library(stargazer)
library(tibble)
library(grid)
library(gridExtra)
library(usmap)
library(car)
library(sandwich)
library(lmtest)
library(reshape2)

setwd("~/Desktop/Stone/Berkeley_MIDS/Statistics/Labs/Lab_3")
full_data <- read.csv('crime_v2.csv')
data <- na.omit(full_data)

# Check for duplicated data and remove duplicates
sum(duplicated(data))
```

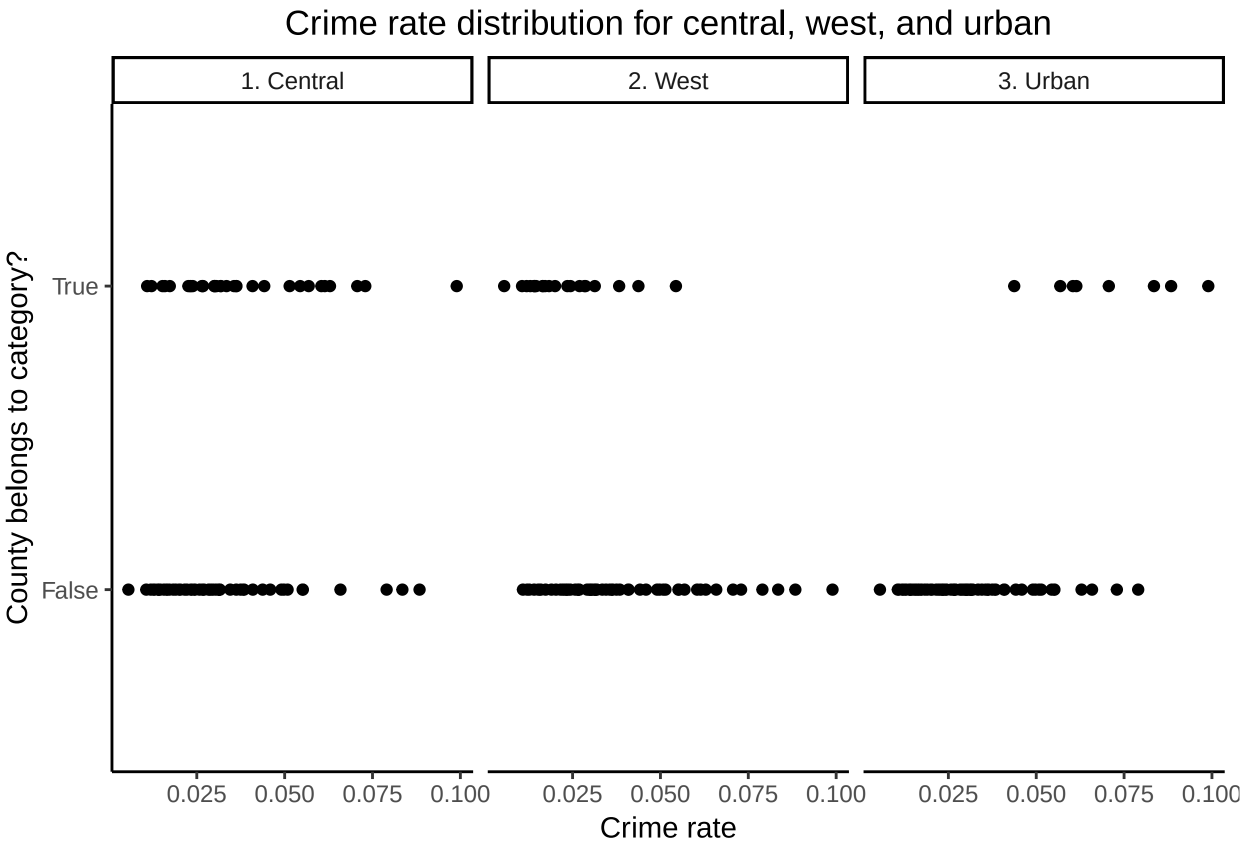
```
## [1] 1
```

```
data <- distinct(data, .keep_all=T)

# Convert prbconv factor in numeric
data$prbconv <- as.numeric(levels(data$prbconv))[data$prbconv]

# Check that all fields are numerical
for (field in names(data)) {
  stopifnot(class(data[,field]) %in% c("numeric", "integer"))
  }
```

After this initial data cleaning, we identified columns we believed could be potential casual predictors of crime rate. N.B. county and year were disregarded in our models because they are identifiers.

The variables west, central and urban are categorical, which have been one-hot encoded. These can be used directly in the regression. We first saw whether the distribution of crime rate is different depending on the location (west vs central) and whether the county was urban.

```
C <- data[, c('west', 'central', 'urban')]

df.categorical <- data[, c('crmrte', 'west', 'central', 'urban')]
colnames(df.categorical) <- c('crmrte', '2. West', '1. Central', '3. Urban')
dt_long <- gather(df.categorical, key, value, -crmrte)

ggplot(dt_long, aes(x = crmrte, y = factor(value))) +
  geom_point() +
  facet_grid(. ~ key) +
  theme_classic()+
  theme(plot.title = element_text(hjust = 0.5))+
  labs(title="Crime rate distribution for central, west, and urban",
       x='Crime rate',
       y = "County belongs to category?") +
  scale_y_discrete(breaks=c(0,1), labels=c("False","True"))
```



For Central versus not Central North Carolina, the crime rate distribution is relatively even. Counties in Western North Carolina appear to have less crime on average than those labeled as not Western. Counties labeled as Urban have more crime on average than those not. We see definitive summaries below for the urban and west variables below.

```
u <- data %>%
  group_by(urban) %>%
  summarise(mean_crime_rate = mean(crmrte)) %>%
  as.data.frame()
u$urban <- c('N', 'Y')
w <- data %>%
  group_by(west) %>%
  summarise(mean_crime_rate = mean(crmrte)) %>%
  as.data.frame()
w$west <- c('N', 'Y')
print(u)
```

```
##   urban mean_crime_rate
## 1     N      0.02990170
## 2     Y      0.07049427
```

```
print(w)
```

```
##   west mean_crime_rate
## 1    N      0.03720145
## 2    Y      0.02209975
```

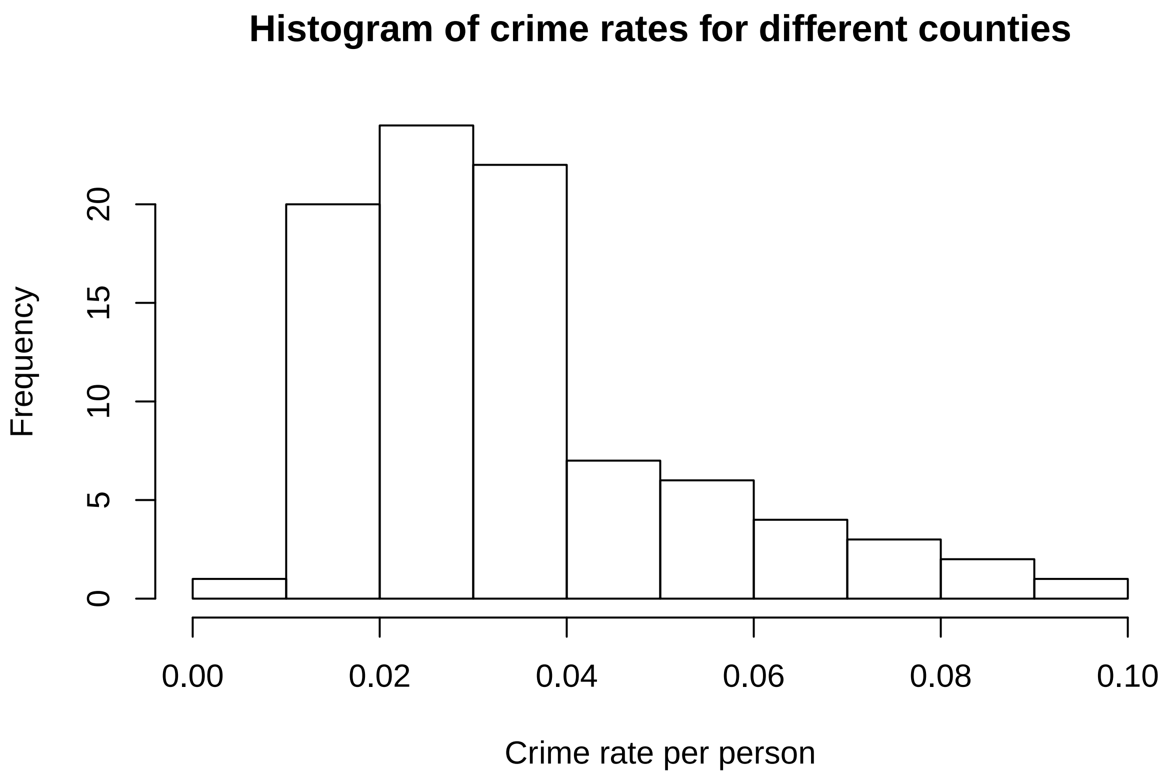
All other variables are numeric, and are also possible candidates for influencing crime rate.

Then, we examined the crime rate variable on its own.

```
summary(data$crmrte)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## 0.005533 0.020604 0.030002 0.033510 0.040249 0.098966
```

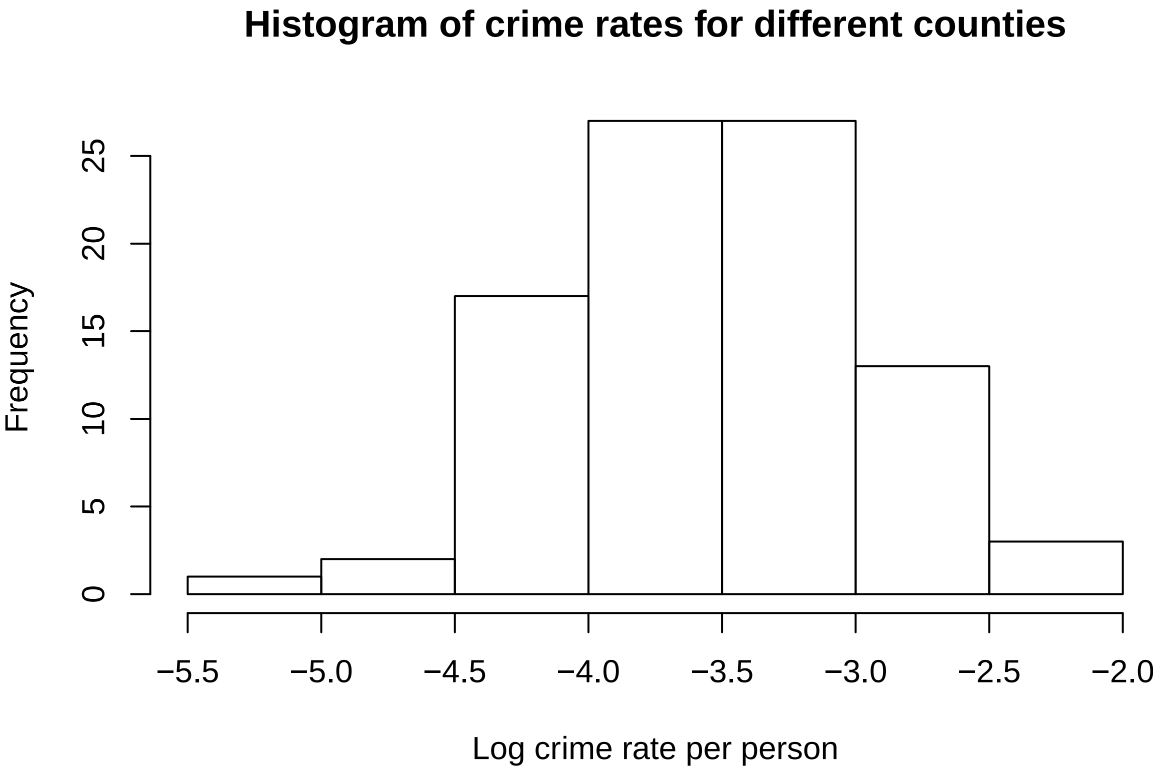
```
hist(data$crmrte,
     main='Histogram of crime rates for different counties',
     xlab = 'Crime rate per person')
```



We can see that crime rate is greater than 0 for all counties. We can also see from this plot that crime rate is highly skewed toward larger values. Since we are performing inference analysis, we would like to interpret our model coefficients as how changes in explanatory variables affecs changes in crime rate. However, since the baseline crime is different for different counties, it makes more sense to transform crime rate into the log of crime rate This changes the interpretation from absolute changes in crime rates to percent changes (at least for small changes in crime rate since the percent interpretation is only accurate for differentially small changes), which makes comparisons across counties more comparable. For example, a 0.01 change in crime rate for the lowest county (0.005) is a much larger change than for the largest county (0.099), but a 1% change is comparable. As a result, we perform inference on the log of crime rate.

We can visualize the transformed crime rate variable again.

```
data$crmrte_abs <- data$crmrte
data$crmrte <- log(data$crmrte)
hist(data$crmrte,
     main='Histogram of crime rates for different counties',
     xlab = 'Log crime rate per person')
```


Histogram of crime rates for different counties

Explain further

Model 1

Key Variables

For our initial model, we would like to focus on factors that intuition says should influence crime. We believe that there are four variables which represent deterrents to crime: probability variables of arrest, conviction, prison sentence, and the severity of punishment in average sentence days. We will call these variables the “fear factors;” namely, we believe the higher the chance someone believes they will be arrested, convicted, or sent to prison, the less likely they will commit a crime. Also, the more severely they believe the punishment to be (prison day sentences), the less likely they will commit a crime. Out of these four, we believe probability of arrest and probability of conviction will have the greatest effects. The reason is that a single arrest or convinction can permanently damage someone’s record. For most people who have never committed crimes before, just the idea of possibly getting in trouble with the police could be enough to deter them. In addition, there are many crimes that result in fines, community service, and other forms of punishment that does not involve prison. Many criminals are likely not thinking about possibility or severity of prison sentences because they might feel even if arrested they can talk their way out of it. For heavy repeat offenders, they will likely prison sentence into account, but possibility of arrest is still a heavy influencer. As a result, we believe arrest and convinction are the most relevant variables.

The wage variables can either deter or motivate individuals to commit a crime and were excluded from this base model. We believe that the more satisfied someone is with their income, the less likely they will commit a crime because they are more likely to attain their desires without having to pursue illegal routes. Along the same lines, unemployment is likely to lead to increased crime rates. Too high of a wage, especially in blue collar jobs, means me employees can be “priced out” some employees. As wage goes up, individuals paid that wage are expected to do more, lowering the amount of workforce necessary, leading to greater unemployment. As a result, we believe wage can go either way. For our base model, we will look at only what we consider traditional blue collar jobs: construction and manufacturing. We also take the log of these variables: this is common practice as we want to measure the effect of a percent changes in salary, and not absolute changes.

```
nonwage_variables <- c('prbarr', 'prbconv', 'prbpris', 'avgsen',
                       'polpc', 'density', 'taxpc',
                       'pctymle', 'pctmin80', 'mix',
                       'urban', 'central', 'west')

wage_variables <- c('wtrd', 'wfir', 'wser', 'wfed', 'wsta', 'wloc',
                    'wcon', 'wtuc', 'wmfg')

X_non_wage <- data[, names(data) %in% nonwage_variables]
X_wage <- lapply(data[, names(data) %in% wage_variables], log)

X_wage_transformed <- cbind(X_non_wage, X_wage)
```

Before performing EDA on the variables listed above, we note why we have chose to exclude the other variables in our base model:

We believe that density should be a positive predictor of crime. Highly dense population areas present more opportunities for crime. There also tends to be a larger wage and wealth gap in these areas, which increases the rate of crime as people will be tantalized to use illegal ways to get to the top. However, this may absorb too much of the causal effect.

We also believe that the police per capita is a key variable that would absorb too much of the model. Namely, more police are required for regions of greater crime, and so counties with more crime are more likely to have more police. In addition, the fact that there's more police could mean that more crime is detected and responded to, increasing the recorded number of criminal cases and perceived rate of crime. However, we also expect there to be a tipping point. If the density of police is extremely high, that likely acts as a major deterrent for criminals. As a result, police and crime rate are very intricately linked and want to avoid this for our base model.

Location could be important. Different geographic areas may be more prone to crime due to cultural and socioeconomic differences. However, we would like a model that can be generalized.

Male absorbs the causality because then it will only reflect that.

Tax could reflect how people vote; see paper

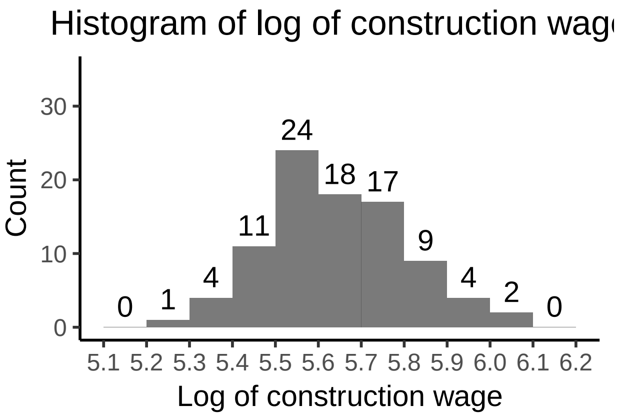
Demographics (minority, male) certainly increase prediction but even playing ground first (good moderator variable, but not mediator variable)

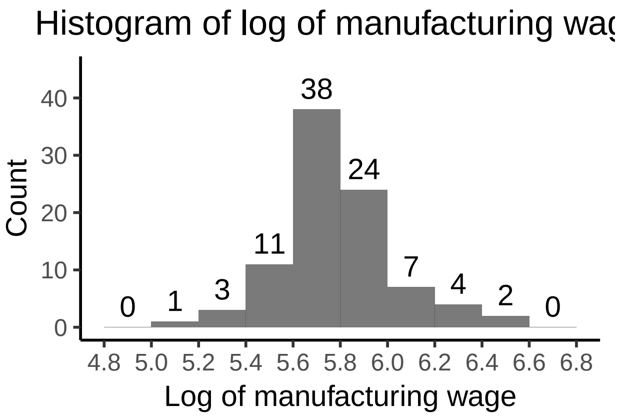
Mix shouldn't have too much effect.

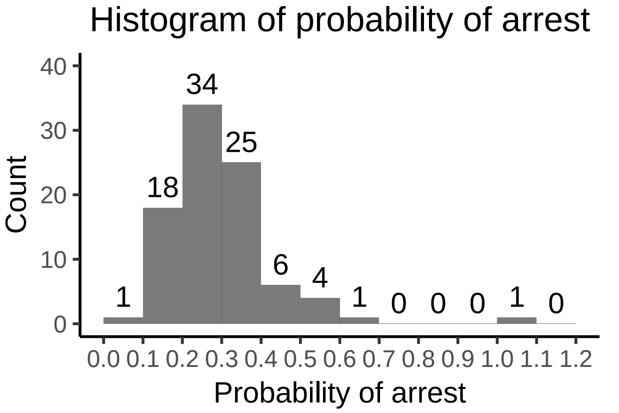
Explain above further

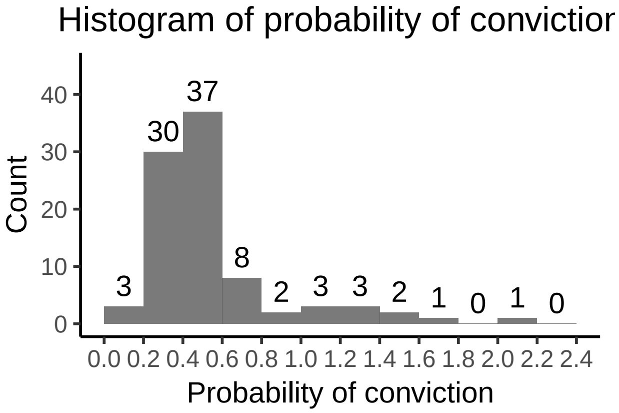
For our 4 variables, we will perform EDA to ensure that we have reasonable data. We will plot a grid of histograms and look at the distribution of the explanatory variables.

```
hist.wcon <- ggplot(data = X_wage_transformed, aes(x=wcon)) +
  geom_histogram(alpha=0.8, breaks=seq(5.1, 6.2, by=0.1)) +
  labs(title='Histogram of log of construction wage',
       x='Log of construction wage',
       y = "Count") +
  theme_classic() +
  ylim(0,35)+
  theme(plot.title = element_text(hjust = 0.5))+
  scale_x_continuous(breaks=seq(5.1, 6.2, by=0.1)) +
  stat_bin(aes(y=..count.., label=(..count..)),
           geom="text",
           vjust=-.5,
           breaks=seq(5.1, 6.2, by=0.1))

hist.wmfg <- ggplot(data = X_wage_transformed, aes(x=wmfg)) +
  geom_histogram(alpha=0.8, breaks=seq(4.8, 6.8, by=0.2)) +
  labs(title='Histogram of log of manufacturing wage',
       x='Log of manufacturing wage',
       y = "Count") +
  theme_classic() +
  ylim(0,45)+
  theme(plot.title = element_text(hjust = 0.5))+
  scale_x_continuous(breaks=seq(4.8, 6.8, by=0.2)) +
  stat_bin(aes(y=..count.., label=(..count..)),
           geom="text",
           vjust=-.5,
           breaks=seq(4.8, 6.8, by=0.2))
hist.prbarr <- ggplot(data = X_wage_transformed, aes(x=prbarr)) +
  geom_histogram(alpha = 0.8, breaks=seq(0,1.2,0.1)) +
  labs(title = "Histogram of probability of arrest",
       x = "Probability of arrest",
       y = "Count") +
  theme_classic() +
  ylim(0,40)+
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_x_continuous(breaks=seq(0,1.2,0.1)) +
  stat_bin(aes(y=..count.., label=(..count..)),
           geom="text",
           vjust=-.5,
           breaks=seq(0, 1.2, by=0.1))
hist.prbconv <- ggplot(data = X_wage_transformed, aes(x=prbconv)) +
  geom_histogram(alpha = 0.8, breaks=seq(0,2.4,0.2)) +
  labs(title = "Histogram of probability of conviction",
       x = "Probability of conviction",
       y = "Count") +
  theme_classic() +
  ylim(0,45)+
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_x_continuous(breaks=seq(0,2.4,0.2)) +
  stat_bin(aes(y=..count.., label=(..count..)),
           geom="text",
           vjust=-.5,
           breaks=seq(0, 2.4, by=0.2))
grid.arrange(hist.wcon, hist.wmfg,
             hist.prbarr, hist.prbconv,
             nrow=2, ncol=2)
```









We see that the log of the two wage variables have no outliers. Both are somewhat upward skewed in that there are more data points above the mode of each, but overall, the distribution looks fairly symmetric. Both of the probability variables are upward skewed.

For the probability of arrest, defined as the number of arrests to offenses, the general trend is a skew to the right, and there is one data point that lies above 1. One possible explanation for this is that we are looking at a cross-sectional data pooled from a multi-year study. For example, if data collection started in June of one year, and people committed an offence in January of that year but not arrested until after June, that person could appear in this data set as having been arrested but not committing an offense. As a result, even though we have one outlier, we will leave in this data point in our regression analysis.

For the probability of convinction, defined as ratio of conviction to arrests, there are many more data points skewed to the right. This variable is confounded by the fact that one does not necessarily need to be arrested to be convicted of a crime. The most common form of this is a citation, which are issued in place of arrests for smaller crimes. As a result, we believe it is perfectly reasonable for this variable to exceed 1.

Coefficient interpretation

For our first model, the coefficient in prbarr represents the effect of probability of arrest on crime rate. Specifically, keeping all other variables constant, per unit increase in the probability of arrest leads to a certain percent change in crime rate. Since this variable represents the "fear factor" we presented above, we would hope that this change is negative. An analogous interpretation can be said for probability of conviction. With these variables, we want to measure how a perceived probability of getting in trouble with the legal system deters crime.

For our first model, the coefficients on the wage variables represents how a small percent change in average wage in that blue collar industry relates to a small percent change in crime rate. This variable can really go both ways: a higher wage could mean that potential criminal are satisfied with their income and would pursue alternative methods for achieving their goals. Alternatively, higher wage could mean less jobs for potential criminals. We now build our regression model:

```
#first model
model_1 <- lm(data$crmrte ~ prbarr + prbconv + wcon + wmfg, data = X_wage_transformed)
print(model_1)
```

```
##
## Call:
## lm(formula = data$crmrte ~ prbarr + prbconv + wcon + wmfg, data = X_wage_transformed)
##
## Coefficients:
## (Intercept)       prbarr      prbconv         wcon         wmfg
##     -8.4338      -1.6815      -0.7070       0.4696       0.5408
```

Finish this; I'm fairly sure but not entirely that for large coefficients whether we can interpret at percentage changes; For example, -1.68, does that mean a 0.01% change in prbarr leads to about a 1.68% decrease in crime rate?

For this model, we see that the coefficients on the "fear" variables are both negative, meaning that these factors negative influence crime rate. Since the coefficients are all rather large, we cannot directly interpret this in terms of a percent change. We can instead note that increasing either of independent variables by a single unit is a significant amount. For example, increasing the probability of arrest from 0.5 to 1.5 means a 3x increase in probability of arrest, which would require drastic changes and efforts on the part of law enforcement. As a result, we will interpret the coefficients in terms of a 0.1, or 10% increases.

Namely, keeping all other explanatory variables constant, we see that a 0.1 unit increase in the probability of arrest, or a 10% change, leads to a 0.17 units of decrease in the the log of crime rate (or very roughly 17% decrease in crime rate). Similarity, a 0.1 unit increase in the probability of arrest leads to a 0.07 units of decrease in the log of crime rate, or about a 7% decrease in crime rate.

For the wage variables, the pattern is in the opposite direction. Keeping all other explanatory variables constant, for each 0.1 log unit of increase in wage of construction, we see approximately a 0.047 log units of increase in crime. Similarity, for each 0.1 log unit of increase in wage of manufacturing jobs leads to 0.054 log units of increase in crime.

At this point, we will evaluate the Classical Linear Model Assumptions, and perform hypothesis testing to see whether each of our coefficients are statistically significant.

CLM 1: Linear in parameters

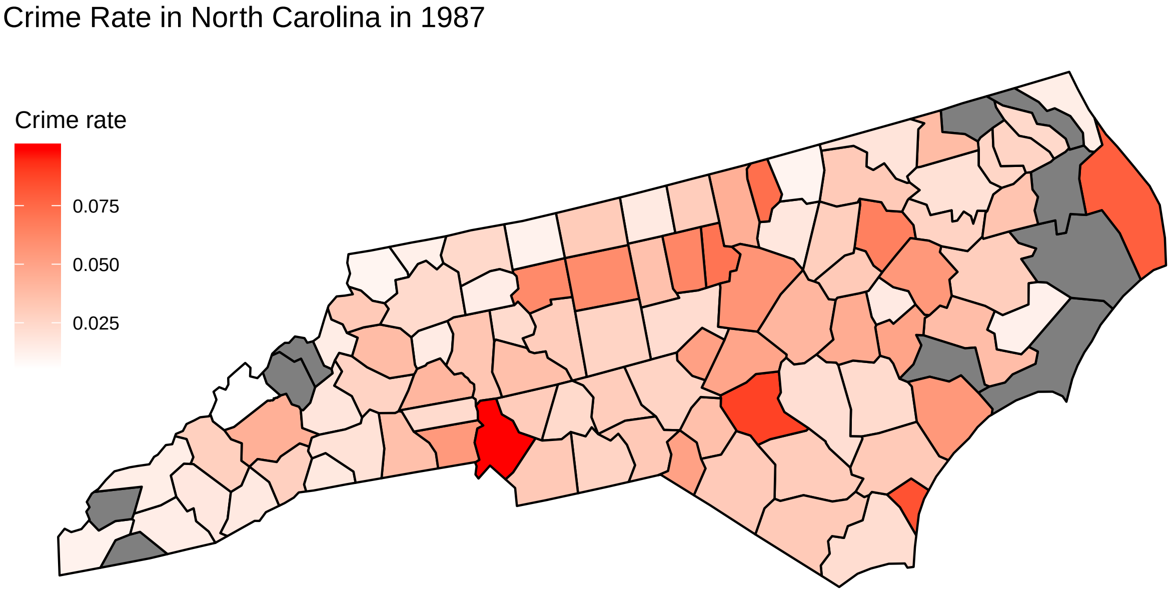
Nothing to assess here. We define the model with an error term such that the parameters are linear (and assume this model is the population model and estimate its parameters). The independent variables can be transformed in any way, including taking logs as we have done.

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_k x_k + u$$

CLM 2: Random Sampling

This is a rare case where we actually have a majority of the population at hand. We are interested in crime rate in the state of North Carolina, which has 100 counties. We have data for 90 of these counties. We can generate a fun visualization to see where which counties were eliminated to see if there was systematic geographic bias. This is done with the plot_usmap package:

```r
data$fips <- 37*1000 + data$county
plt_data <- select(data, fips, crmrte_abs)

plot_usmap('counties', include = 'NC', data = plt_data, values='crmrte_abs')+
  scale_fill_continuous(low='white', high='red') +
  labs(title = "Crime Rate in North Carolina in 1987", fill="Crime rate")+
  theme(legend.position = c(0,0.4))
```



We see that the 10 counties without data (in black) are somewhat clustered along the eastern and western/north western boarders of NC. This can certainly skew our analysis to that of central NC. But since we have data points for even clustered geographic regions where data is missing, we should be able to draw fairly reasonable conclusions about crime in the state as a whole.

Within each county, which we can view as our available population, we have no reason to believe that the sampling of random, or even in some cases a concensus. For example, it is not hard to imagine that the crime rate per capita could be calculated from police records as a consensus. Our police per capita, data from the FBI, is also likely a consensus. Wage variables are likely a sample of employees, at least from available data reported to the IRS. We have no reason to believe that this sample was biased in any way. Overall, given the limited information, we have little reason to drastic doubt an IID sample within our available population of 90 states.

CLM 3: No perfect multi-collinearity

First, multi-collinearity is guaranteed when we have more features than samples, which is not the case here. Second, multi-collinearity can occur when one variable is a perfect linear combination of another set of variables. In that case, the one of those variables are regressed on the remaining of the group, the R^2 will be 1. R would have warned us if this were the case that we had perfect multi-collinearity, so in this case we have fulfilled this requirement. We can this using the VIF for each coefficient to evaluate whether some degree of multicollinearity should be of worry:

```r
vif(model_1)
```

```
##   prbarr  prbconv     wcon     wmfg
## 1.090235 1.026432 1.244954 1.164963
```

We see that all VIF factors are significant below 4, which means we do not have significant multi-collinearity to worry about.

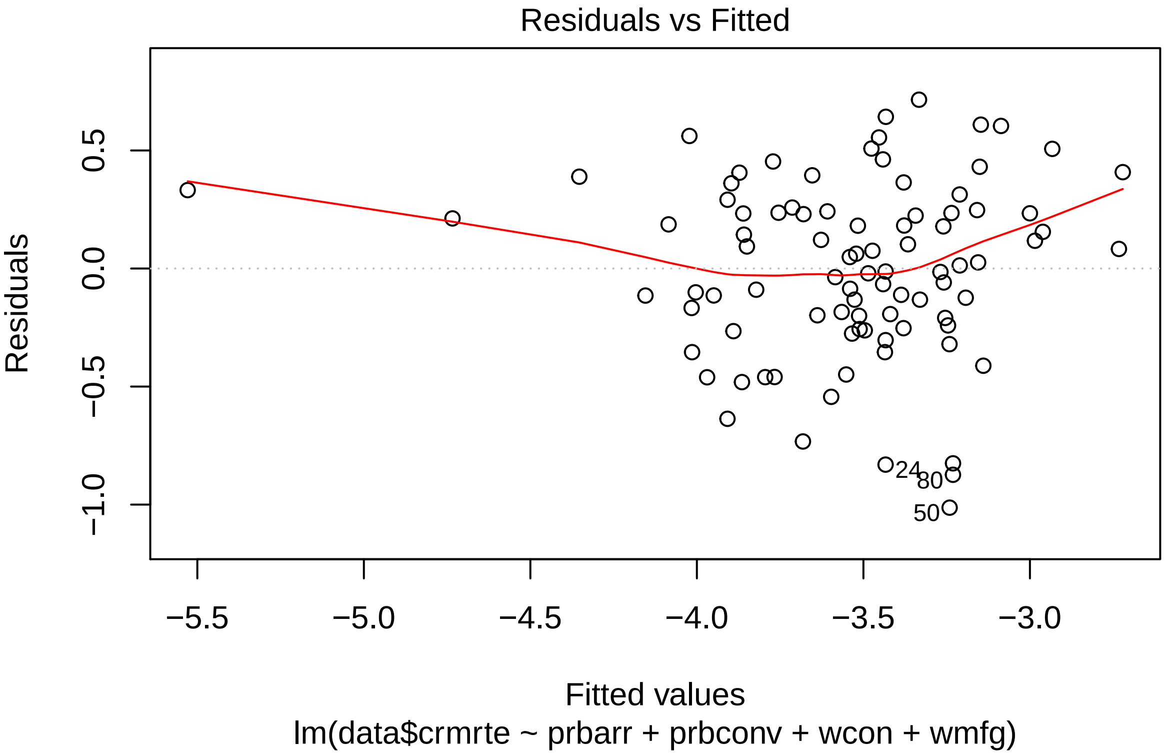
CLM 4: Zero Conditional Mean

Zero conditional mean states that the expected value of the error term is 0 for all values of the independent variables x_i's:

$$E(u|x_1, x_2, ..., x_k) = 0$$

Under zero conditional mean, we expect that the residuals on the residuals versus fitted value plot to have an expected value of 0 across the board. To check this, we plot this:

```r
plot(model_1, which = 1)
```



Based on this plot, we see that unfortunately, we see that the line adopts a U shape. However, the curvature is a result of very few data points on the extreme ends of the fitted values. In the middle where the bulk of our data is, from -4 to just before -3, the line seems flat and centered around 0. However, above 3, the 6 data points are all above 0. The conclusion is that our model most likely does not satisfy CLM 4. We will need to adjust our model by adding more parameters in order to capture more of the variation in crime rate due to omitted variables.

CLM 5: Homoskedasticity

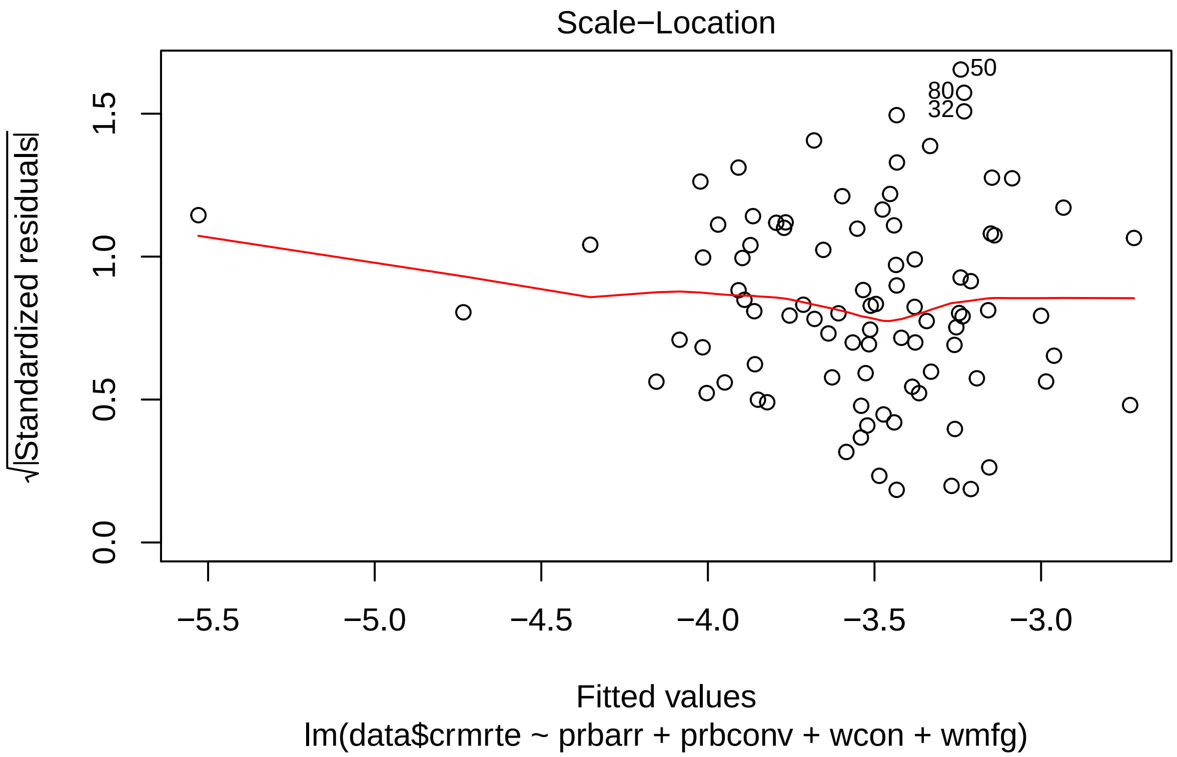
Homoskedasticity assumption is that the variance of the error terms are constant for any combination of x_i values:

$$Var(u|x_1, x_2, ..., x_k) = \sigma^2$$

Examining the fitted values versus residuals plot above, while the spread (larger the spread the greater the estimated variance) appears to be slightly larger around fitted values of around 3.75 (around -1 to 0.5) than around 4 (around -0.5 to 0.5), overall there aren't major observable patterns in differences in variance as a function of x.

We can also check the scale-location plot. If homoskedasticity were achieved, we would expect a horizontal line across this plot:

```r
plot(model_1, which=3)
```



We see that this line is roughly horizontal from -5 to -3. The only major curvature is the single data point around -5.5. However, this is likely due to small sample size for that particular fitted values. Discrepancies such as that observed are much more likely when the sample size is small. This indicates that we most likely have close to homoskedasticity.

One way to test for homoskedasticity is the Breusch-Pagan Test. The null hypothesis of the test states that we have homoskedasticity. We will test at a standard significance level of 0.05.

$$H_0 : \text{Homoskedasticity}$$
$$H_a : \text{Heteroskedasticity}$$

```r
bptest(model_1)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  model_1
## BP = 4.0136, df = 4, p-value = 0.4042
```

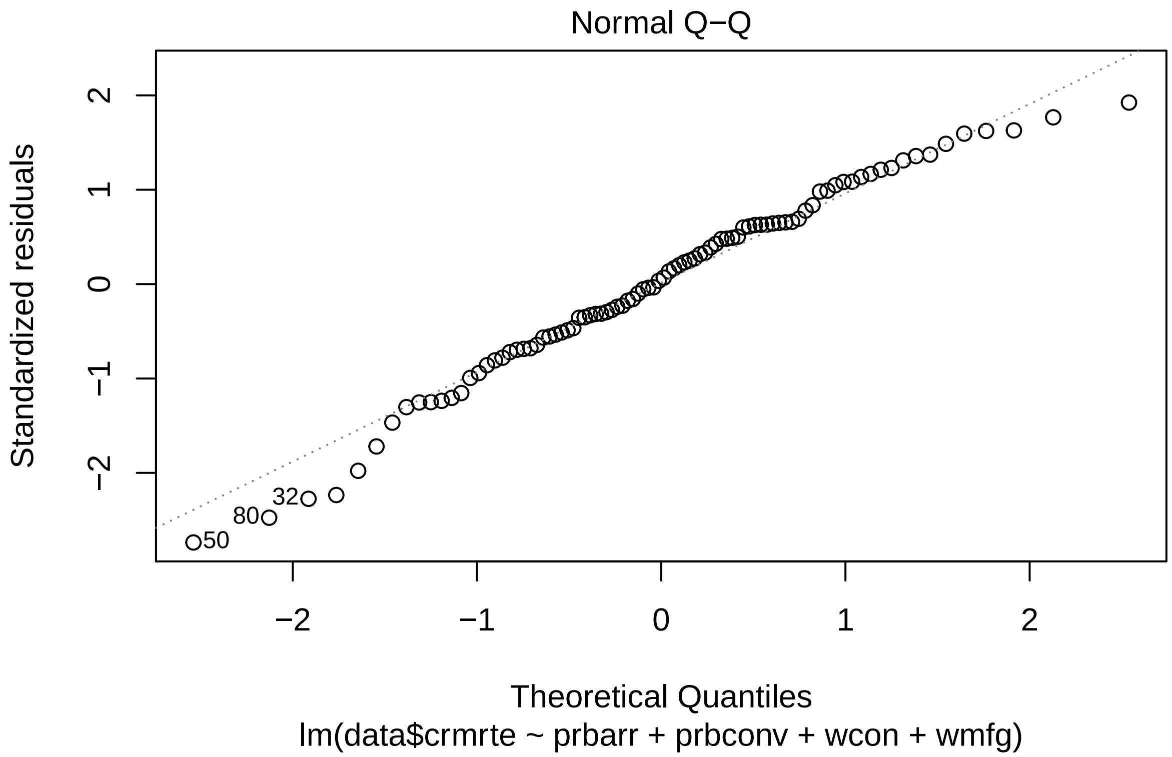
Since the p-value >> 0.05, we fail to reject the null hypothesis that we have homoskedasticity.

In any case, it is good practice to almost always use heteroskedastic robust errors, especially since we have some doubt from the residuals versus fitted values plot.

CLM 6: Normality

CLM 6 assumes that population error is independent of the explanatory variables x1 through xk, and that the error term is normally distributed with mean 0 and constant variance. We can check this with the qqplot of the fitted values versus residuals plot.

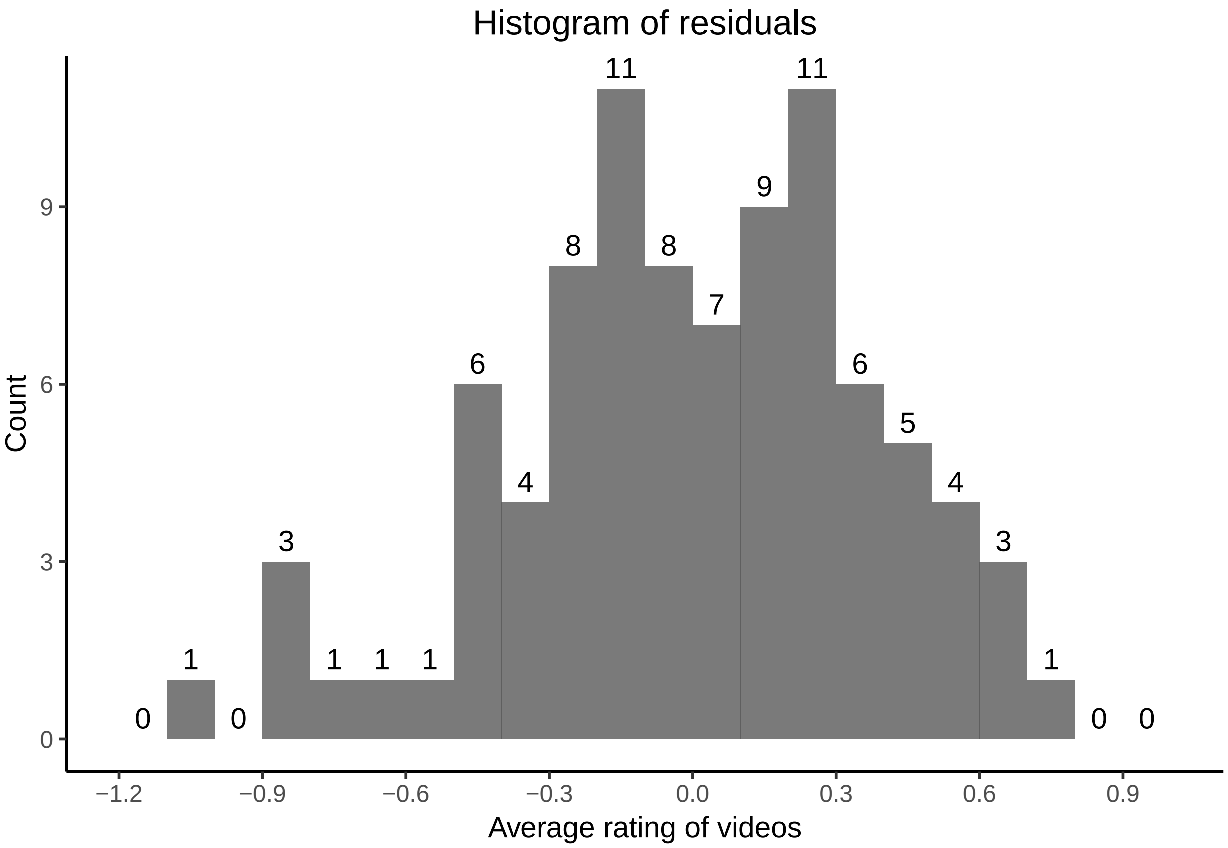
```
plot(model_1,which=2)
```



Not even counting the exception of extreme values, the data points waivering back and forth, which could indicate a kurtosis problem. Also, most differ from where we would like them to be on the line, so this indicates we most likely do not have normality of errors.

We can visualize the residuals in a histogram:

```
bins <- seq(-1.2,1,0.1)
ggplot(data = as.data.frame(model_1$fitted.values), aes(x=model_1$residuals))+
  geom_histogram(alpha=0.8, breaks=bins)+
  labs(title='Histogram of residuals',
       x='Average rating of videos',
       y = "Count") +
  theme_classic() +
  #ylim(0,2200)+
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_x_continuous(breaks=seq(-1.2, 1, 0.3)) +
  stat_bin(aes(y=..count.., label=(..count..)),
           geom="text",
           vjust=-.5,
           breaks=bins
           )
```

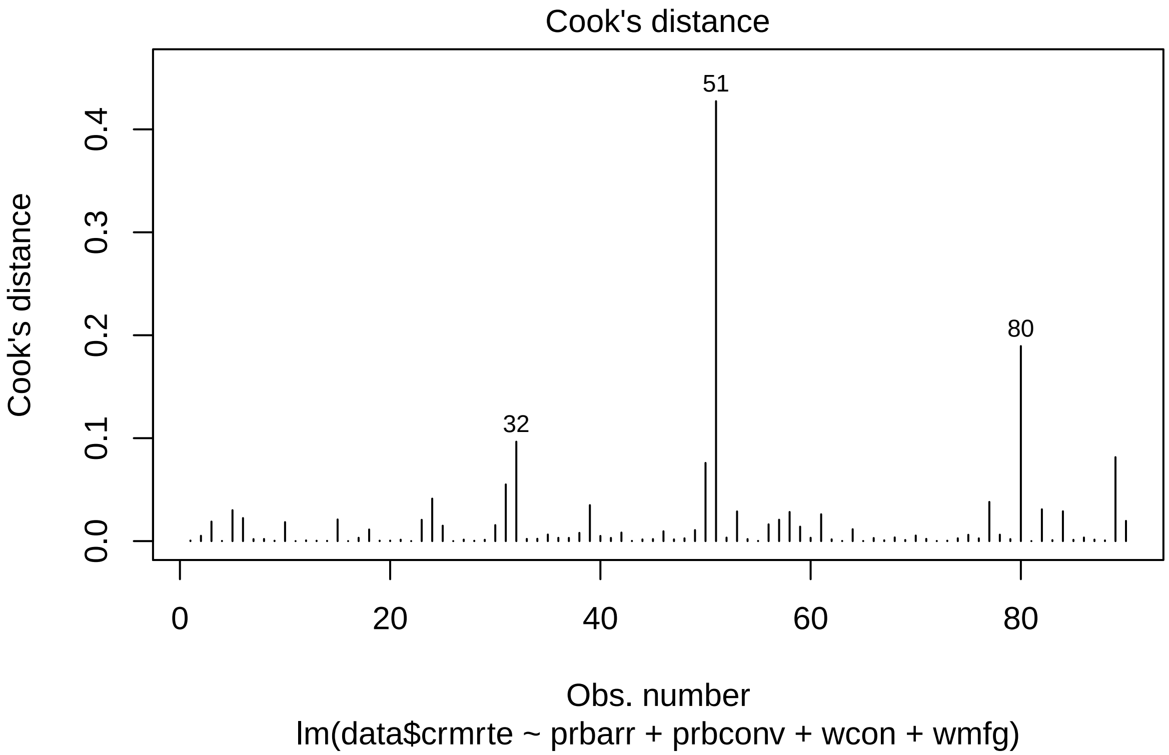


Based on the histogram, the data does not appear very normal. In fact, it is somewhat bimodal around -0.4, and 0.2.

In any case, since our sample size 90 is much greater than 30, asymptotics also kicks in, ensuring that the sampling distribution of our coefficients are approximately normal. This will be important in statistical testing.

Finally, we would like to check and see if there are any outliers in our model that might have significant influence:

```r
plot(model_1, which=4)
```



We see that data point 51 could be problematic. While its Cook's distance is stil below 0.5, which is typically considered to be large, it does deviate significantly from the average. We will examine this data point further in our future models.

We will now perform statistical testing to see whether the four coefficients we included are statistically significant. To do this, we derive heteroskedastic errors from the vcovHC function from the sandwich package. This function produces a covariance matrix, and the standard errors are the square root of the diagnal.

```
se.model_1<-sqrt(diag(vcovHC(model_1)))
se.model_1
```

```
## (Intercept)      prbarr     prbconv        wcon        wmfg 
##   1.9047222   0.4570443   0.1447330   0.3162314   0.2486414
```

We see that prbarr and wcon have the largest standard errors, so we are least certain about their estimates from the model.

In order to look at the statistical significance of our statistics, we can perform a t-test using the robust standard errors. Since we have large sample size, the sampling distribution of our statistic is approximately normal, so our statistic is distributed as a t-distribution:

$$\frac{\hat{\beta_j} - \beta_j}{se(\hat{\beta_j})} \sim t_{n-k-1}$$

For all the betas, we will use a 2-sided test as significance level 0.05

$$H_0 : \beta_j = 0$$
$$H_a : \beta_j \neq 0$$

To perform the test for all of the variables, we use the coeftest package, specifying the degrees of freedom as sample size - 4 (number parameters except beta_0) - 1, and the heteroskedasticity-consistent estimation of the covariance matrix.

```
model_1.tests<-coeftest(model_1, vcov = vcovHC, df=dim(X_wage_transformed)[1] - 4 - 1)
model_1.tests
```

```
##
## t test of coefficients:
##
##              Estimate Std. Error t value  Pr(>|t|)
## (Intercept) -8.43382    1.90472 -4.4278 2.813e-05 ***
## prbarr      -1.68150    0.45704 -3.6791 0.0004097 ***
## prbconv     -0.70698    0.14473 -4.8847 4.815e-06 ***
## wcon         0.46959    0.31623  1.4849 0.1412566
## wmfg         0.54077    0.24864  2.1749 0.0324169 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Based on the statistical test, we see that both the probability of arrest and probability of conviction are both highly significant variables, while the wage of construction is not statistically significant. The wage of manufacturing is statistically significant however.

FINISH THIS INTERPRETATION

Model 2

model 2 <- lm(data$crmrte ~ prbarr + prbconv + polpc + taxpc + pctmin80 + pctymle + wfed + wsta, data = X_wage_transformed)

For model 2, we wanted to add in other covariates meant to increase accuracy of prediction. To do this, we wanted to first get a sense of crmrte correlation with all numeric variables. We also parse our data into X (numeric variables) and y (crmrte)

```
y <- data$crmrte
X <- data[,!names(data) %in% c('county',
                                'year',
                                'crmrte',
                                'west',
                                'central',
                                'urban',
                                'crmrte_abs')]



#correlate all variables and store in new dataframe
cor_df <- data.frame(variable = character(),
                     crmrte_cor = numeric())
for (x in names(X)) {
  crmrte_cor <- cor(y, data[,x])
  corr <- as.data.frame(crmrte_cor,
                        col.names = c('crmrte_cor')) %>%
                        add_column(variable = x, .before = 1)
  cor_df <- rbind(cor_df, corr)
}
```

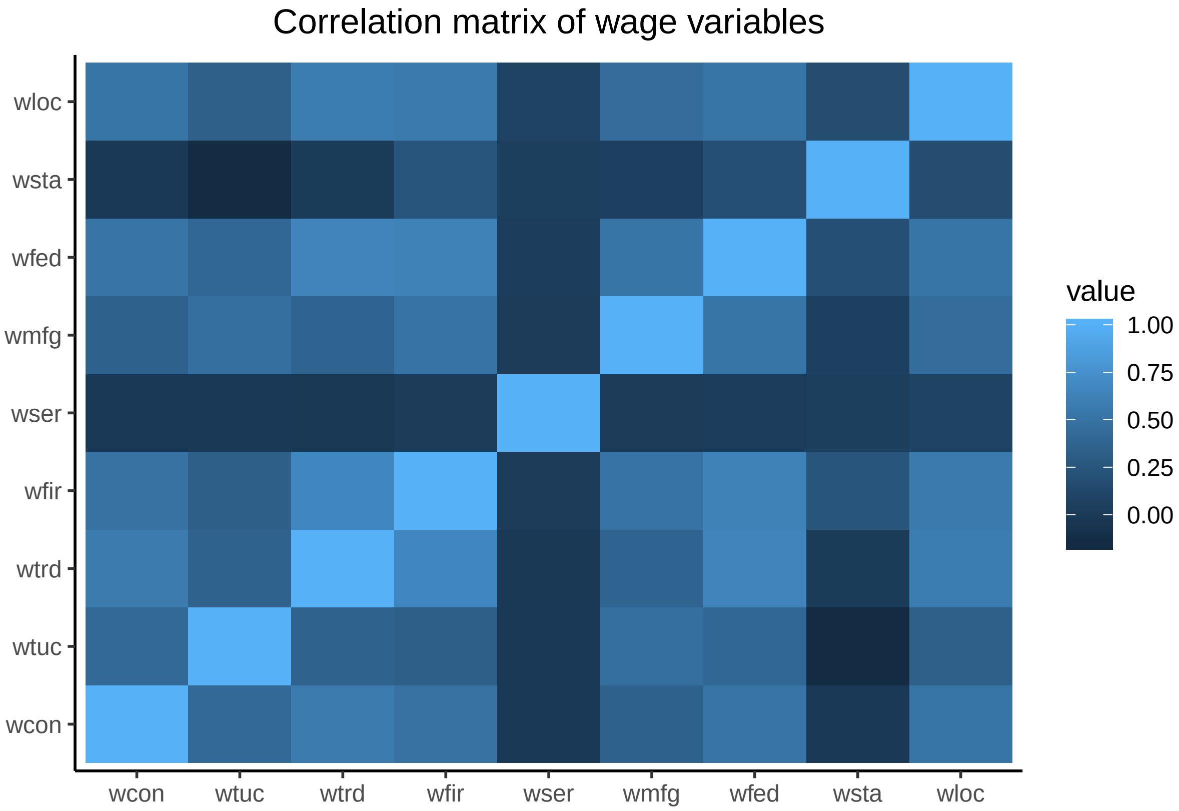
```r
cor_df <- arrange(cor_df, desc(crmrte_cor))
print(cor_df)
```

```
##    variable  crmrte_cor
## 1   density  0.63302339
## 2      wfed  0.52330585
## 3      wtrd  0.39379240
## 4      wcon  0.39371486
## 5     taxpc  0.35832339
## 6      wmfg  0.30753731
## 7      wfir  0.29324265
## 8      wloc  0.28856678
## 9   pctymle  0.27815466
## 10 pctmin80  0.23291821
## 11     wtuc  0.20146493
## 12     wsta  0.16970208
## 13     fips  0.02376789
## 14  prbpris  0.02147024
## 15    polpc  0.01040580
## 16   avgsen -0.04936931
## 17     wser -0.11312801
## 18      mix -0.12473445
## 19  prbconv -0.44681361
## 20   prbarr -0.47276691
```

It should be no surprise that density is the best single positive predictor of crime rate. As stated before, highly dense population areas present more opportunities for crime, and also have larger wealth gaps. In fact, since we want to predict crime, density in some ways may be viewed as an output variable. Crime is in terms of per person, and a person ability to commit crime, even perhaps unknowingly, increases as the density of population increases, simply due to more opportunities. For example, imagine the thought experiment where we randomly sample some group of people from the entire population of NC state. Then, we randomly assign each person to live in a rural area or a densely populated area. We believe that every time, the group assigned to the densely populated area will commit more crime on average, simply because each person has more opportunity to do this. As a result, this variable may absorb some of the "causal effects" of other variables, and we would like to exclude this from our regression models. Instead, we will save this variable for model 3 in order to check the robustness of our model 2 (the golden child of this report). Furthermore, urban is a similar categorical variable that is directly related to density, and will serve the same purpose in model 3.

We will now take a closer look at wage variables:

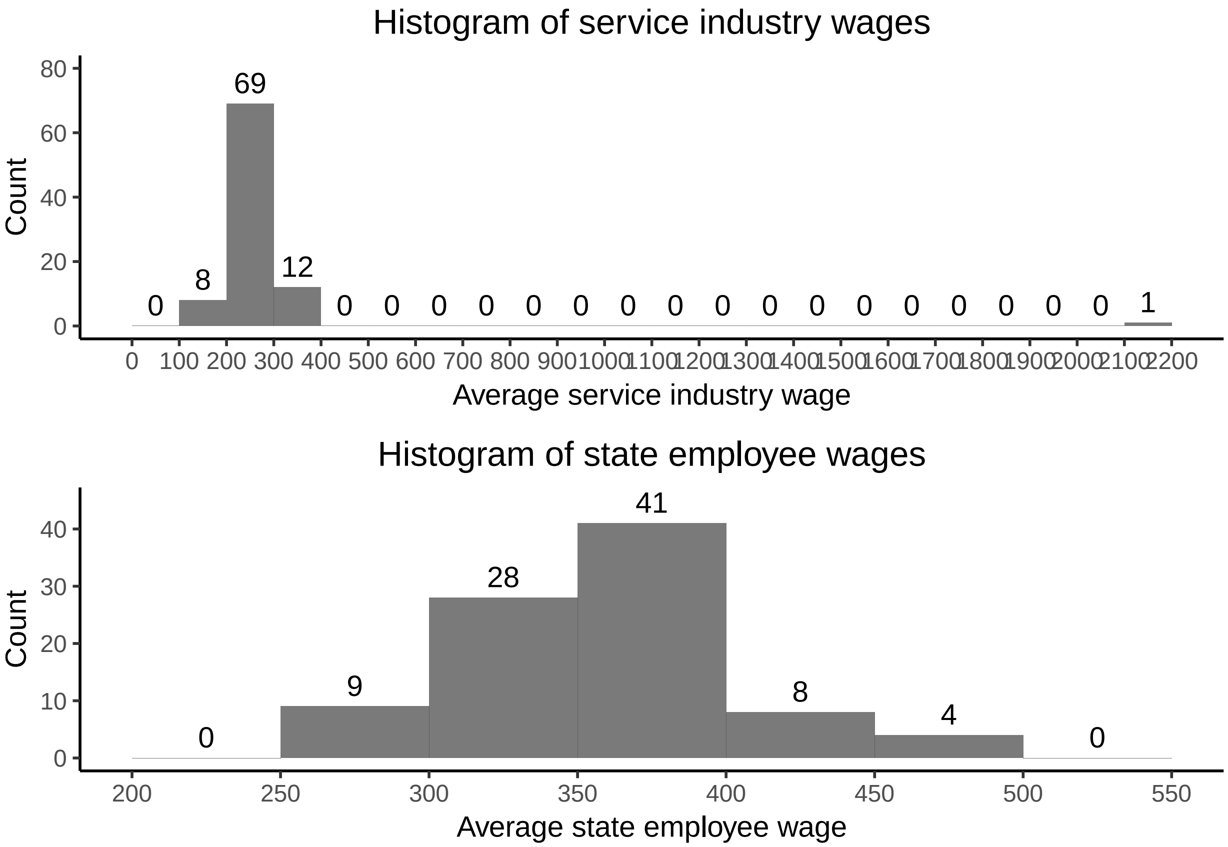
```r
nonwage_variables <- c('prbarr', 'prbconv', 'prbpris', 'avgsen',
                       'polpc', 'density', 'taxpc',
                       'pctymle', 'pctmin80', 'mix',
                       'urban', 'central', 'west')

wage_variables <- c('wtrd', 'wfir', 'wser', 'wfed', 'wsta', 'wloc',
                    'wcon', 'wtuc', 'wmfg')

X_non_wage <- data[, names(data) %in% nonwage_variables]
X_wage <- data[, names(data) %in% wage_variables]

heatmap.data <- melt(cor(X_wage))
ggplot(data = heatmap.data, aes(x=Var1, y=Var2, fill=value)) +
  geom_tile()+
  labs(title='Correlation matrix of wage variables',
       x='',
       y = "") +
  theme_classic() +
  theme(plot.title = element_text(hjust = 0.5))
```



Interestingly, wser and wsta are both relatively dark compared to the rest of the data set. If the data was accurate, then that's a good indication of strong independent predictors within the wage category. wfed is the largest single univariate predictor from the correlation table. Both wfed and wsta are government (federal and state) jobs with the potential to influence social and political change.

We start by performing some EDA to ensure that these variables are reasonable:

```
hist.wser <- ggplot(data = data, aes(x=wser)) +
  geom_histogram(alpha = 0.8, breaks=seq(0,2200,100)) +
  labs(title = "Histogram of service industry wages",
       x = "Average service industry wage",
       y = "Count") +
  theme_classic() +
  ylim(0,80)+
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_x_continuous(breaks=seq(0,2200,100)) +
  stat_bin(aes(y=..count.., label=(..count..)),
           geom="text",
           vjust=-.5,
           breaks=seq(0,2200,100))
hist.wsta <- ggplot(data = data, aes(x=wsta)) +
  geom_histogram(alpha = 0.8, breaks=seq(200,550,50)) +
  labs(title = "Histogram of state employee wages",
       x = "Average state employee wage",
       y = "Count") +
  theme_classic() +
  ylim(0,45)+
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_x_continuous(breaks=seq(200,550,50)) +
  stat_bin(aes(y=..count.., label=(..count..)),
           geom="text",
           vjust=-.5,
           breaks=seq(200,550,50))
grid.arrange(hist.wser, hist.wsta,
             nrow=2, ncol=1)
```

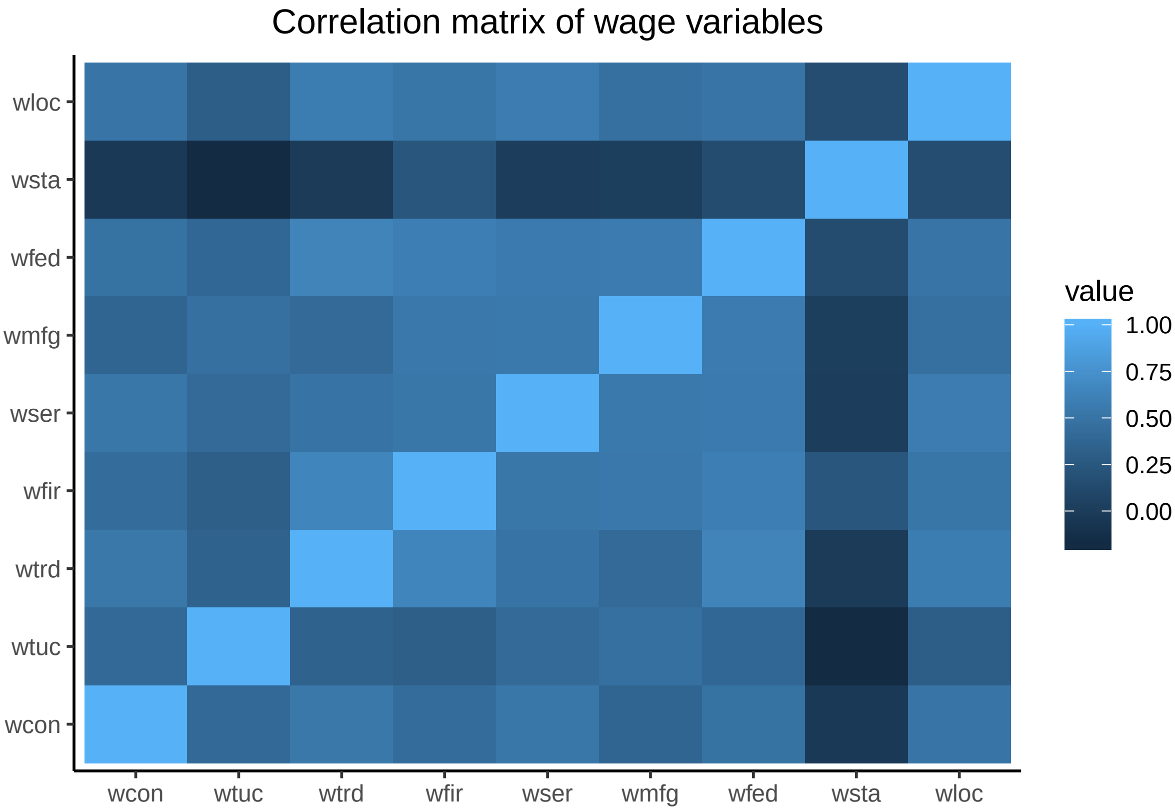


The state wage is somewhat right skewed, likely because high ranking officials make more than most average employees. However, there is not any red flags. However, we see that there's a huge outlier in service industry wages, more than 10 fold. Very likely this is an error in which the decimal was shifted by 1. The county is Warren County, which is not known to have such high service industry wages. Even if the data point is accurate, it may be significantly skewed for example due to non-random sampling of CEOs of service industry. In this case, this data point would not represent our target population, which is all employees in the service industry. We choose to fix this point by imputing the value to the average across the state.

need to justify above a little bit better

```
data$wser <- ifelse(data$wser>1000, mean(data$wser), data$wser)
X_wage_transformed$wser <- log(data$wser)

X_wage <- X_wage_transformed[, names(X_wage_transformed) %in% wage_variables]
```

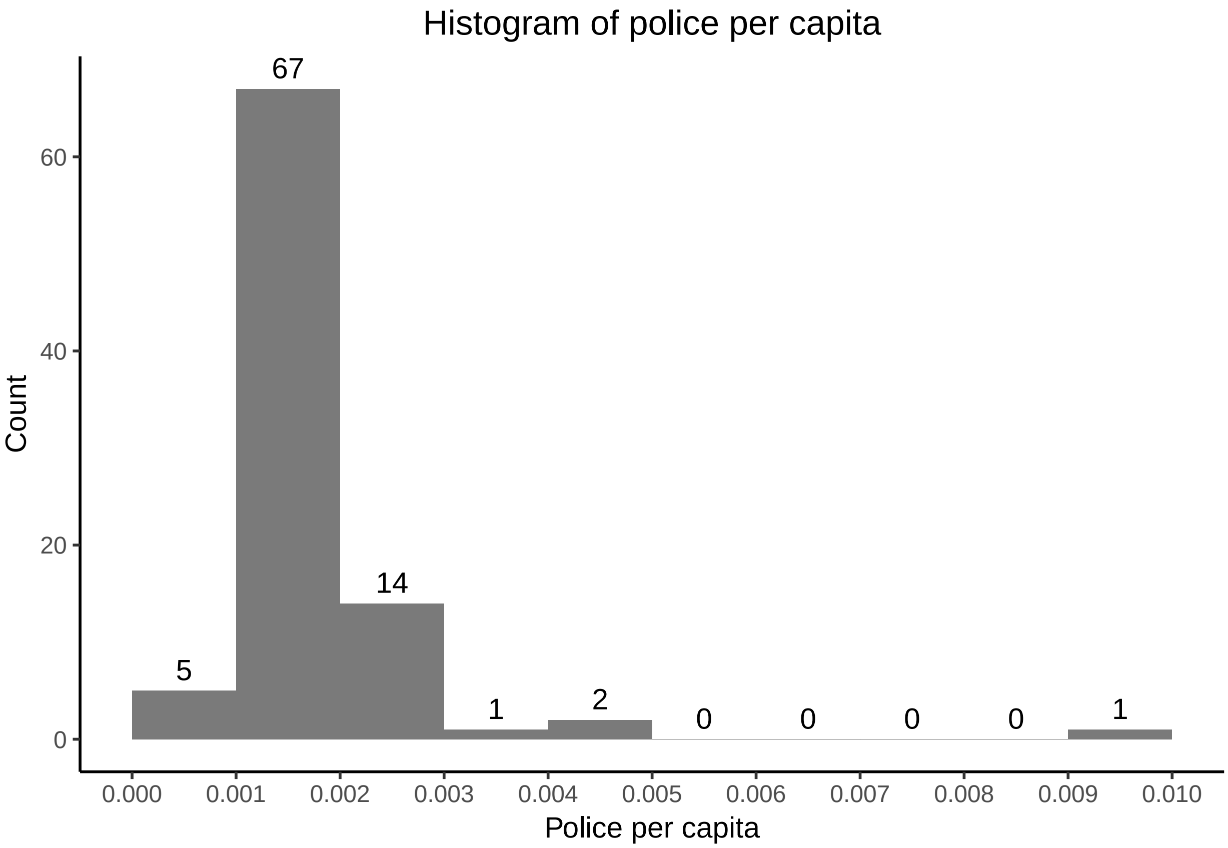
```
heatmap.data <- melt(cor(X_wage))
ggplot(data = heatmap.data, aes(x=Var1, y=Var2, fill=value)) +
  geom_tile()+
  labs(title='Correlation matrix of wage variables',
       x='',
       y = "") +
  theme_classic() +
  theme(plot.title = element_text(hjust = 0.5))
```



Unfortunately, with the imputatation of the incorrect data point, we see that wser is now well-correlated with other wage variables. However, wsta still has weak correlation with all other wage variables and represents a good independent predictor in the wage category.

Next, we state above that we would like to include police because more police could result in stronger detection of crime. #explain a bit more

```
breaks = seq(0,0.01,0.001)
ggplot(data = data, aes(x=polpc)) +
  geom_histogram(alpha = 0.8, breaks=breaks) +
  labs(title = "Histogram of police per capita",
       x = "Police per capita",
       y = "Count") +
  theme_classic() +
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_x_continuous(breaks=breaks) +
  stat_bin(aes(y=..count.., label=(..count..)),
           geom="text",
           vjust=-.5,
           breaks=breaks)
```



Again, we see a significant outlier. However, we believe this could be real for the following reason:

```r
report <- rbind(data[51, c('crmrte_abs', 'prbarr', 'prbconv', 'polpc')],
                as.data.frame(lapply(select(data, crmrte_abs, prbarr, prbconv, polpc), mean)))
rownames(report) <- c('outlier county', 'state average')
stargazer(report,type='text', summary = FALSE)
```

```
##
## ==============================================
##                crmrte_abs prbarr prbconv polpc
## ----------------------------------------------
## outlier county   0.006    1.091   1.500  0.009
## state average    0.034    0.295   0.551  0.002
## ----------------------------------------------
```

The probability of arrest and is more than 3x state average, as is the probability of conviction. The police force is 4.5 times the state average. More police in a county means more crime gets detected, and more police means that more crime gets responded to in a timely fashion. Interestingly, the rate of crime is about 5-6x less than state average. This is actually consistent with the analysis above. The capacity of the police force to respond to crime likely exceeds the rate of crime. As a result, people are less likely to commit crime because they know they will be caught, and at the same time any crime that does get committed is likely dealt with leading to arrests and convictions. All in all, this is to say that we actually believe this is a legitimate data point, and as a result will keep it in our regression analysis.

We also believe percent minority will be a good independent predictor of crime. The unfortunately truth is that especially in rural regions, minorities are more likely to targets for the police. Their behavior tends to be scrutinized more, and sometimes even activities considered non-criminal could be considered criminal for minorities. In addition, due to their socialeconomic position, minorities statistically are poorer, which could be an exogenous variable correlated with crime. In our case, young male won't be considered a priority because the nature of the crime is not specified. Young male can't commit some of the white collar crimes for example. If the data was only for example, petty theft, perhaps young male would be a good predictor.

Explain this more just so we have something to add to our model later

Due to the success of the fear variables above, we also believe that probability and length of prison sentence are both potentially important.

Finally, while we see that geography is important in some cases, we are uncertain whether it will be a strong predictor.

R^2 for effect size

Conclude that at this point we believe variables that should be added back (wsta, wser, police, minorties); see if forward selection with AIC optimization supports this. General strategy: Optimize to adjusted R^2 and AIC, and once we have the full model confirm at the very end that all of our coefficients are significant.

At this point, we've outlined a few variables both from model 1 and EDA that we think would be fruitful to include in our model 2. In order to further narrow down the variable selection further, we begin with a global AIC optimization using a combination of forward and backward selection, and then apply knowledge from the EDA above to further adjust our model.

In order to perform automatic feature selection using the AIC criteria, we will use the MASS package:

```
names_not_include <- c('density', 'urban')

y <- data$crmrte
X_stepwise <- X_wage_transformed[, !(names(X_wage_transformed) %in% names_not_include)]

model_upper <- lm(y ~ ., data=X_stepwise)
model_lower <- lm(y ~ 1, data=X_stepwise)

AIC.mixed <- stepAIC(model_upper,
                     trace = FALSE,
                     direction = 'both',
                     scope = list(upper=model_upper,lower=model_lower))
AIC.mixed
```

```
##
## Call:
## lm(formula = y ~ prbarr + prbconv + polpc + taxpc + pctmin80 +
##     pctymle + wfed + wsta, data = X_stepwise)
##
## Coefficients:
## (Intercept)       prbarr      prbconv        polpc        taxpc
##   -8.971281    -2.170226    -0.785200   161.991529     0.006035
##    pctmin80      pctymle         wfed         wsta
##    0.011242     3.568045     1.374461    -0.503116
```

```
model_2 <- AIC.mixed
```

First, we see that the insignificant variables in our first model did not show up in this AIC optimized model, which is a good sign as we likely would not want to include wages of blue collar jobs in our second model. The AIC mixed model also suggests that a lot of the same variables are from our EDA above, with addition of two variables we did not expect to include, which are percent young male, and taxpc, or tax revenue per capita, and the exclusion of probability of conviction and length of prison sentence which we thought would be important. #talk a bit more above.

CLM assumptions look good; zero conditional mean satisfied; homoskedasticity and normality no but that's ok

```
coeftest(AIC.mixed, vcov. = vcovHC, df=dim(X_wage_transformed)[1] - 8 - 1)
```

```
##
## t test of coefficients:
##
##                Estimate  Std. Error t value  Pr(>|t|)
## (Intercept)  -8.9712809   2.3524359 -3.8136 0.0002662 ***
## prbarr       -2.1702256   0.3073045 -7.0621 5.115e-10 ***
## prbconv      -0.7852002   0.0990757 -7.9253 1.054e-11 ***
## polpc       161.9915286  42.8434794  3.7810 0.0002976 ***
## taxpc         0.0060345   0.0039908  1.5121 0.1343975
## pctmin80      0.0112424   0.0014907  7.5416 5.968e-11 ***
## pctymle       3.5680450   2.5768716  1.3846 0.1699643
## wfed          1.3744611   0.3245105  4.2355 5.987e-05 ***
## wsta         -0.5031155   0.2457556 -2.0472 0.0438783 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Beautiful! Everything is highly significant except our theory above about young male and tax which we did not even consider. Since these variables are were not in our original hypothesis, let's check their significance. We don't expect much co-linearity between these two variables, but we can test whether they are jointly significant.

explain a bit more

```
linearHypothesis(AIC.mixed, c("taxpc = 0", "pctymle = 0"), vcov = vcovHC)
```

```
## Linear hypothesis test
##
## Hypothesis:
## taxpc = 0
## pctymle = 0
##
## Model 1: restricted model
## Model 2: y ~ prbarr + prbconv + polpc + taxpc + pctmin80 + pctymle + wfed +
##     wsta
##
## Note: Coefficient covariance matrix supplied.
##
##   Res.Df Df      F Pr(>F)
## 1     83
## 2     81  2 1.9224 0.1529
```

We will remove these variables from our model.

MODEL 2 removing these variables.

ADD exclusion of probability of conviction and length of prison sentence which we thought would be important and see how it affects the model. See that they are not. Guess multicollinear so check matrix. See if significant if we remove arrest/conviction. Yes but worse Rˆ2 so keep arrest/convict

Now do a "RESET" test to see if non-linear combinations of these variables might help further improve the model

TEST if the effect of state and federal wages on crime is the same.

TEST if probability of arrest versus conviction is the same. Which fear is more important, to convict or to just give more arrests?

```r
#linearHypothesis(model2, "prbarr = prbconv", vcov = vcovHC)
```

Model 3

Adding back density no longer influences the adjusted r squared value! The model_2 is already very robust. 23 predictors increase the adjusted r squared by very little compared to using just 8.

Regress on everything including density. Every variable can be argued in some way.

```r
model_3 <- lm(data$crmrte ~ ., data = X_wage_transformed)
summary(model_3)$adj.r.squared
```

```
## [1] 0.8157934
```

```r
AIC(model_3)
```

```
## [1] 16.57352
```

```r
model_2.coef <- model_2$coefficients
model_3.coef2 <- model_3$coefficients[names(model_3$coefficients)  %in% names(model_2$coefficients)]
report <- cbind(model_2.coef, model_3.coef2)
colnames(report) <- c('model_2 coefficients', 'model_3_coefficients')
stargazer(report, summary=FALSE, header=FALSE, type='text')
```

```
##
## =======================================================
##             model_2 coefficients model_3_coefficients
## -------------------------------------------------------
## (Intercept)        -8.971               -8.274
## prbarr             -2.170               -1.915
## prbconv            -0.785               -0.687
## polpc             161.992              156.394
## taxpc               0.006                0.003
## pctmin80            0.011                0.009
## pctymle             3.568                3.166
## wfed                1.374                1.027
## wsta               -0.503               -0.395
## -------------------------------------------------------
```

AIC worse; adjusted rˆ2 only slighly better; coefficients barely change

other useful discussions

Other variables include the probability variables of arrest, conviction, prison sentence, as well as the severity of punishment in average sentence days. We will call these variables the “fear factors;” namely, we believe the higher the chance someone believes they will be arrested, convicted, or sent to prison, the less likely they will commit a crime. Also, the more severely they believe the punishment to be (prison day sentences), the less likely they will commit a crime.

We need to include more reasons why from the original research paper

First, we look at our data for probability variables, and average sentence days.

```
fear_factors <- c('prbarr', 'prbconv', 'prbpris', 'avgsen')
stargazer(data[, fear_factors], type='text',
          summary.stat = c("min", "p25", "median", "p75", "max", "median", "sd"))
```

```
##
## ===============================================================
## Statistic  Min  Pctl(25) Median Pctl(75)  Max   Median St. Dev.
## ---------------------------------------------------------------
## prbarr    0.093  0.205   0.271   0.345   1.091  0.271   0.138
## prbconv   0.068  0.344   0.452   0.585   2.121  0.452   0.354
## prbpris   0.150  0.364   0.422   0.458   0.600  0.422   0.081
## avgsen    5.380  7.375   9.110   11.465  20.700 9.110   2.834
## ---------------------------------------------------------------
```

Interestingly, we see that the probability of arrest and conviction variables can both exceed 1. Upon doing some external research, we believe this is ok.

PREVIOUS VERSION

need to perform automated residual detection

In order to detect outliers in our data that may greatly influence our prediction model, we will perform pairwise correlation for all numerical variables with the crimerate, and determine the Cook’s distance for all data points in the regression. Any variable for which there contains one more data points with a Cook’s distance greater than or equal to 1 are typically considered influential outliers. These will all be examined closely as part of our EDA.

```
cooks.outliers <- data.frame(cor_variable = character(),
                  distance = c(),
                  county = c())

for (var in names(X)) {
  mod <- lm(y ~ X[, var])
  cooks.distances <- cooks.distance(mod)
  influential_outliers <- names(cooks.distances)[cooks.distances >= 1]

  if (length(influential_outliers > 1)) {
  cooks.outliers <- rbind(cooks.outliers, data.frame(cor_variable = var,
                                    distance = cooks.distances[influential_outliers],
                                    county = influential_outliers))
  }
}

print(cooks.outliers)
```

```
##     cor_variable   distance county
## 51        avgsen   1.064429     51
## 511        polpc  21.573927     51
## 84          wser 137.563613     84
```

explain cook's distance a bit more

We can see that there are 3 influential outliers, that show up in separate correlations.

deal with each of these outliers

With these variables, to find the best optimal set, we first perform backward stepwise model selection with AIC as the criteria to find one possible solution.

EXPLAIN STEPWISE REGRESSION FURTHER.

EDA on each of these; some of the probabilities are greater than 1; explain that these actually aren't a probability. LOOK FOR OUTLIERS.

```
X$prbconv_offense <- X$prbarr * X$prbconv
X$prbpris_offense <- X$prbconv_offense * X$prbpris
X <- X[-2:-3] #remove original prbarr, prbpris variable in factor of transformed

train_X <- cbind(X, C)
nonwage_variables <- c('prbarr', 'prbconv', 'prbpris', 'avgsen',
                       'polpc',
                       'pctymle', 'pctmin80')

wage_variables <- c('wtrd', 'wfir', 'wser', 'wfed', 'wsta', 'wloc',
                    'wcon', 'wtuc', 'wmfg')

X_non_wage <- data[, names(data) %in% nonwage_variables]
X_wage <- lapply(data[, names(data) %in% wage_variables], log)

X_stepwise_set1 <- cbind(X_non_wage, X_wage)

model_upper <- lm(y ~ ., data=train_X)
model_lower <- lm(y ~ 1, data=train_X)

AIC.min_model <- stepAIC(model_upper,
                         trace = FALSE,
                         direction = 'backward',
                         scope = list(upper=model_upper,lower=model_lower))
summary(AIC.min_model)$r.squared
```

```
## [1] 0.8140556
```