

# Kevin's Sandbox

*Kevin Hanna*

*November 23, 2018*

```
library(knitr)
library(kableExtra)
library(car)

## Loading required package: carData

codebook <- read.csv('codebook.csv')
crime <- read.csv('crime_v2.csv')

# Convert columns to factors and logical.
crime$county <- as.factor(crime$county)
crime$year <- as.factor(crime$year)
crime$west <- as.logical(crime$west)
crime$central <- as.logical(crime$central)
crime$urban <- as.logical(crime$urban)

# Create a log of the dependent variable
crime$logcrmrte <- log(crime$crmrte)

# Delete the 6 empty observations at the end, including the row with the apostrophe.
# We can use complete.cases to do this as these 6 observations are the only incomplete observations.
crime = crime[complete.cases(crime), ]

# Fix prbconv which is a factor rather than numeric due to the apostrophe
# Convert from factor to numeric
crime$prbconv = as.numeric(as.character(crime$prbconv))

# county 193 is duplicated, remove one
crime = crime[!duplicated(crime), ]
```

## Preliminary Informations (not intended to be left in)

From the assignment:

- 1. What do you want to measure? Make sure you identify variables that will be relevant to the concerns of the political campaign.
- 2. What transformations should you apply to each variable? This is very important because transformations can reveal linearities in the data, make our results relevant, or help us meet model assumptions.
- 3. Are your choices supported by EDA? You will likely start with some general EDA to detect anomalies (missing values, top-coded variables, etc.). From then on, your EDA should be interspersed with your model building. Use visual tools to guide your decisions.
- 4. What covariates help you identify a causal effect? What covariates are problematic, either due to multicollinearity, or because they will absorb some of a causal effect you want to measure?

## Variables:

### 1. Target

- crmrte

### 2. Label

- county

### 3. Geographic:

- density (likely related to others, especially urban)
- west
- central
- urban

Correlation between logcrmrte and urban: 0.491 and with density 0.633.

Correlation between urban and density is 0.820

Correlation between logcrmrte and west is -0.414 west is also negatively correlated with density.

I think density is an important variable (more so than urban). This would be logical as low income housing is often high-density.

```
# Geographic
#foo2 = lm(crmrte ~ urban + central + west + density, data = crime)
#foo2$coefficients
#vcov(foo2)

foo2log = lm(logcrmrte ~ urban + central + west + density, data = crime)
foo2log$coefficients

## (Intercept)  urbanTRUE centralTRUE  westTRUE  density
## -3.6949892 -0.2841904 -0.2604751 -0.5223082  0.2818198

#vcov(foo2log)

foo2rows = c("logcrmrte", "crmrte", "urban", "central", "west", "density")

round(cor(crime[foo2rows]), 3)

##          logcrmrte crmrte  urban central  west density
## logcrmrte      1.000  0.942  0.491   0.185 -0.414   0.633
## crmrte         0.942  1.000  0.615   0.166 -0.346   0.728
## urban          0.491  0.615  1.000   0.159 -0.087   0.820
## central        0.185  0.166  0.159   1.000 -0.390   0.358
## west          -0.414 -0.346 -0.087  -0.390  1.000  -0.136
## density        0.633  0.728  0.820   0.358 -0.136  1.000

scatterplotMatrix(crime[,foo2rows], diagonal = "histogram")

## Warning in applyDefaults(diagonal, defaults = list(method =
## "adaptiveDensity"), : unnamed diag arguments, will be ignored
```

```

## Warning in smoother(x[subs], y[subs], col = smoother.args$col[i], log.x =
## FALSE, : could not fit smooth

## Warning in smoother(x[subs], y[subs], col = smoother.args$col[i], log.x =
## FALSE, : could not fit smooth

## Warning in smoother(x[subs], y[subs], col = smoother.args$col[i], log.x =
## FALSE, : could not fit smooth

## Warning in smoother(x[subs], y[subs], col = smoother.args$col[i], log.x =
## FALSE, : could not fit smooth

## Warning in smoother(x[subs], y[subs], col = smoother.args$col[i], log.x =
## FALSE, : could not fit smooth

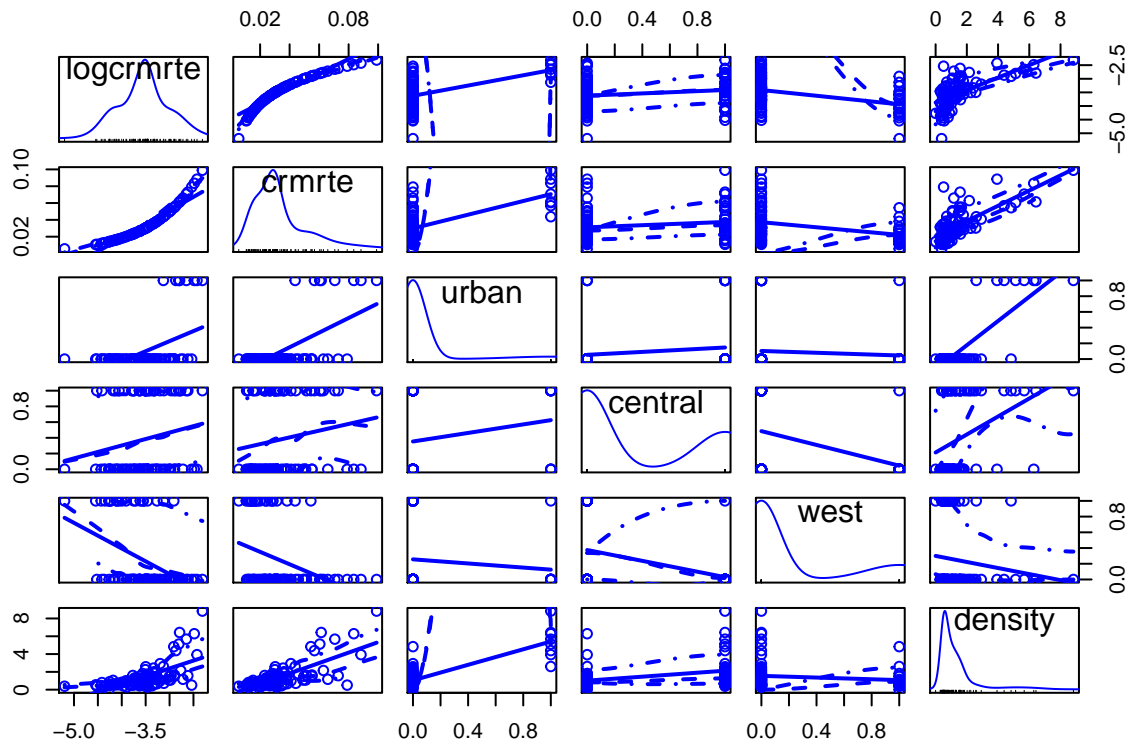
## Warning in smoother(x[subs], y[subs], col = smoother.args$col[i], log.x =
## FALSE, : could not fit smooth

## Warning in smoother(x[subs], y[subs], col = smoother.args$col[i], log.x =
## FALSE, : could not fit smooth

## Warning in smoother(x[subs], y[subs], col = smoother.args$col[i], log.x =
## FALSE, : could not fit smooth

## Warning in smoother(x[subs], y[subs], col = smoother.args$col[i], log.x =
## FALSE, : could not fit smooth

```



#### 4. Cost of doing crime:

##### Probabilities:

- prbconv
- prbpris
- prbarr

Both prbarr and prbconv are negatively correlated to logcrmte (-0.473 and -0.447 respectively). prbconv is less reliable (unless we can explain the  $> 1$  values.)

```
# Probabilities

#foo1 = lm(crmrte ~ prbarr + prbconv + prbpris, data = crime)
#foo1$coefficients
#vcov(foo1)

foo1log = lm(logcrmte ~ prbarr + prbconv + prbpris, data = crime)
foo1log$coefficients

## (Intercept)      prbarr      prbconv      prbpris
## -2.6846297 -1.9991732 -0.7364431  0.3380481

#vcov(foo1log)

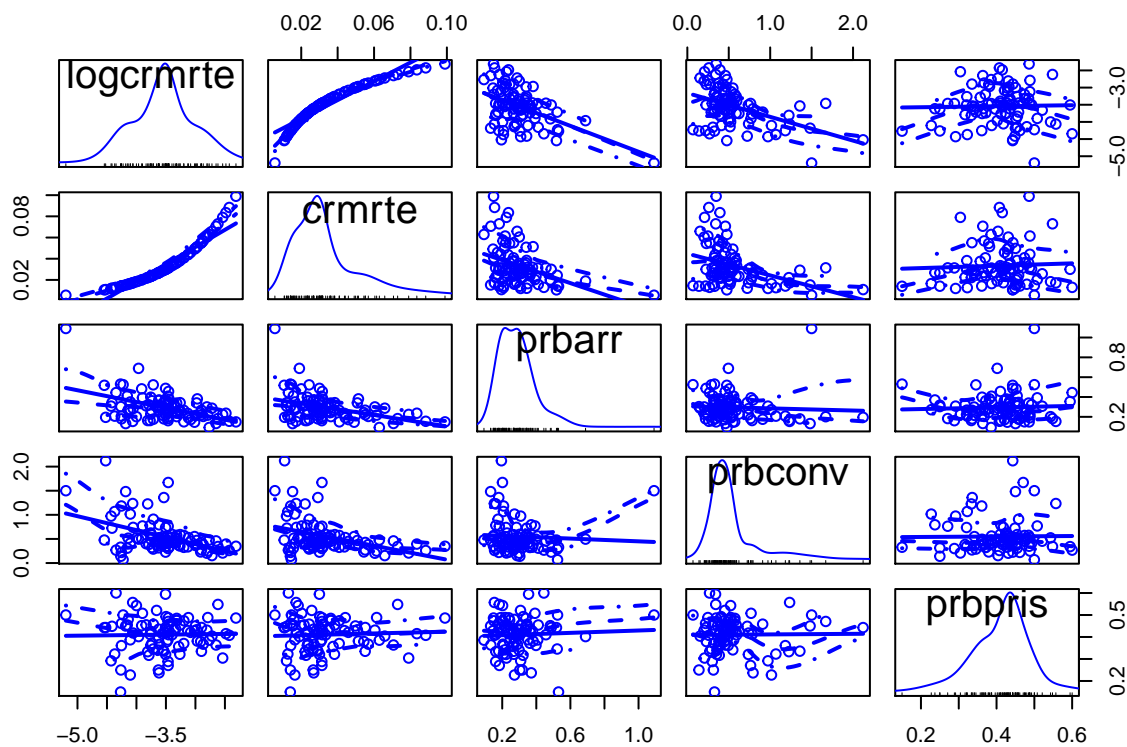
foo1rows = c("logcrmte", "crmte", "prbarr", "prbconv", "prbpris")

round(cor(crime[foo1rows]), 3)

##          logcrmte crmrte prbarr prbconv prbpris
## logcrmte      1.000  0.942 -0.473 -0.447  0.021
## crmrte        0.942  1.000 -0.395 -0.386  0.048
## prbarr        -0.473 -0.395  1.000 -0.056  0.046
## prbconv       -0.447 -0.386 -0.056  1.000  0.011
## prbpris        0.021  0.048  0.046  0.011  1.000

scatterplotMatrix(crime[,foo1rows], diagonal = "histogram")

## Warning in applyDefaults(diagonal, defaults = list(method =
## "adaptiveDensity"), : unnamed diag arguments, will be ignored
```



## Sentence and police

- avgscn
- polpc (likely related to prbconv)

polpc has a huge correlation, it makes sense, but it's still so high we should be very cautious.

```
# Sentence and police

#foo3 = lm(crmrte ~ polpc + avgscn, data = crime)
#foo3$coefficients
#vcov(foo3)

foo3log = lm(logcrmte ~ polpc + avgscn, data = crime)
foo3log$coefficients

## (Intercept)      polpc      avgscn
## -3.45048112 25.08600936 -0.01383982

#vcov(foo3log)

foo3rows = c("logcrmte", "crmte", "polpc", "avgscn")

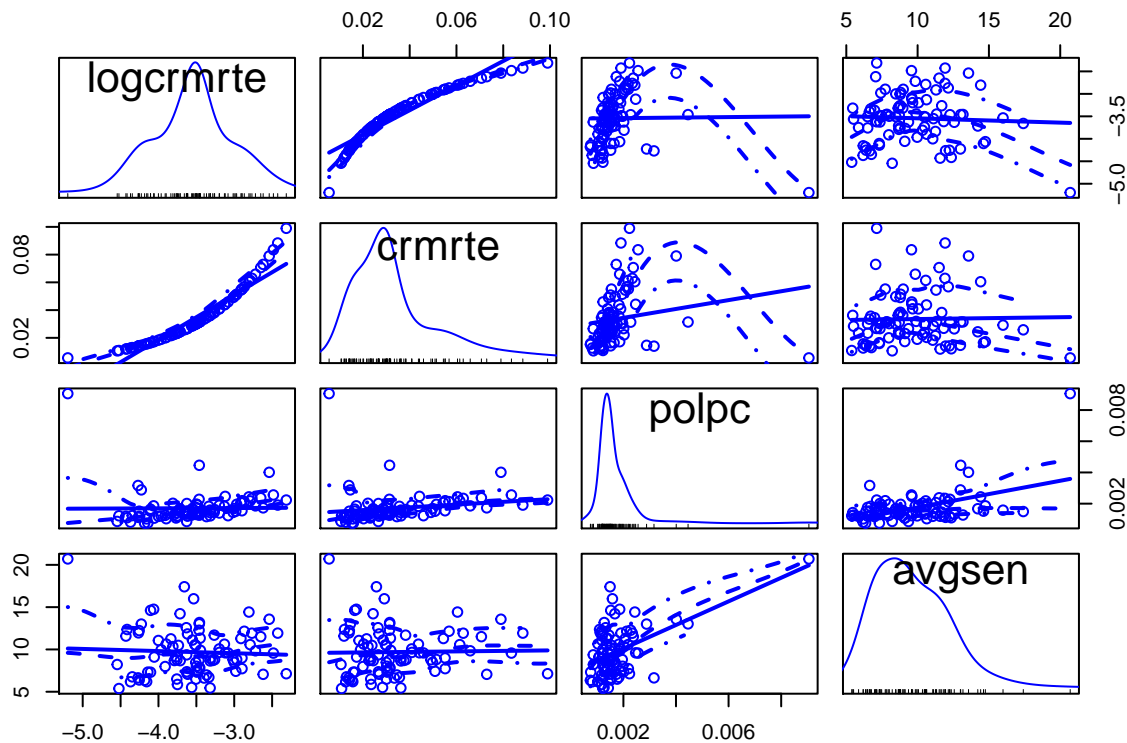
round(cor(crime[foo3rows]), 3)

##          logcrmte crmrte polpc avgscn
## logcrmte      1.000  0.942 0.010 -0.049
```

```
## crmrte      0.942  1.000  0.167  0.020
## polpc       0.010  0.167  1.000  0.488
## avgscen     -0.049  0.020  0.488  1.000
```

```
scatterplotMatrix(crime[,foo3rows], diagonal = "histogram")
```

```
## Warning in applyDefaults(diagonal, defaults = list(method =
## "adaptiveDensity"), : unnamed diag arguments, will be ignored
```



## 5. Economics

- taxpc
- wcon
- wtuc
- wtrd
- wfir
- wser
- wmfg
- wfed
- wsta
- wloc

There's a lot to take in, however the negative relationship to wser (wage service worker) is initially the most interesting.

```
# Economics
#foo4 = lm(crmrte ~ taxpc + wcon + wtuc + wtrd + wfir + wser + wmfg + wfed + wsta + wloc, data = crime)
```

```
#foo4$coefficients
#vcov(foo4)
```

```
foo4log = lm(logcrmrte ~ taxpc + wcon + wtuc + wtrd + wfir + wser + wmfg + wfed + wsta + wloc, data = crime)
foo4log$coefficients
```

```
##      (Intercept)      taxpc      wcon      wtuc      wtrd
## -6.2657436983  0.0139749059  0.0015560093 -0.0003859724  0.0017671261
##           wfir      wser      wmfg      wfed      wsta
## -0.0018814706 -0.0003771697  0.0001090428  0.0046852095  0.0017895087
##           wloc
## -0.0016300505
```

```
#vcov(foo4log)
```

```
foo4rows = c("logcrmrte", "crmrte", "taxpc", "wcon", "wtuc", "wtrd", "wfir", "wser", "wmfg", "wfed", "wsta", "wloc")
round(cor(crime[foo4rows]), 2)
```

```
##      logcrmrte  crmrte  taxpc  wcon  wtuc  wtrd  wfir  wser  wmfg  wfed
## logcrmrte      1.00   0.94  0.36  0.39  0.20  0.39  0.29 -0.11  0.31  0.52
## crmrte         0.94   1.00  0.45  0.39  0.24  0.43  0.34 -0.05  0.35  0.49
## taxpc          0.36   0.45  1.00  0.26  0.17  0.18  0.13  0.08  0.26  0.06
## wcon           0.39   0.39  0.26  1.00  0.41  0.56  0.49 -0.01  0.35  0.51
## wtuc           0.20   0.24  0.17  0.41  1.00  0.35  0.33 -0.02  0.47  0.40
## wtrd           0.39   0.43  0.18  0.56  0.35  1.00  0.67 -0.02  0.37  0.64
## wfir           0.29   0.34  0.13  0.49  0.33  0.67  1.00  0.01  0.50  0.62
## wser          -0.11  -0.05  0.08 -0.01 -0.02 -0.02  0.01  1.00  0.01  0.02
## wmfg           0.31   0.35  0.26  0.35  0.47  0.37  0.50  0.01  1.00  0.52
## wfed           0.52   0.49  0.06  0.51  0.40  0.64  0.62  0.02  0.52  1.00
## wsta           0.17   0.20 -0.03 -0.02 -0.15  0.01  0.24  0.04  0.05  0.19
## wloc           0.29   0.36  0.22  0.52  0.33  0.58  0.55  0.08  0.45  0.52
##           wsta  wloc
## logcrmrte  0.17  0.29
## crmrte     0.20  0.36
## taxpc      -0.03  0.22
## wcon       -0.02  0.52
## wtuc       -0.15  0.33
## wtrd        0.01  0.58
## wfir        0.24  0.55
## wser        0.04  0.08
## wmfg        0.05  0.45
## wfed        0.19  0.52
## wsta        1.00  0.16
## wloc        0.16  1.00
```

```
#scatterplotMatrix(crime[,foo4rows], diagonal = "histogram")
```

## 6. Demographics

- pctmin80
- pctymle

pctmin80 is positively correlated and mix is negatively correlated. This is counter intuitive to me. Requires further

```

# Demographics
#foo5 = lm(crmrte ~ pctmin80 + mix + pctymle, data = crime)
#foo5$coefficients
#vcov(foo5)

foo5log = lm(logcrmrte ~pctmin80 + mix + pctymle, data = crime)
foo5log$coefficients

## (Intercept)      pctmin80          mix      pctymle
## -4.160995761  0.008693096 -1.032861486  6.296019200

#vcov(foo5log)

foo5rows = c("logcrmrte", "crmrte", "pctmin80", "pctymle")

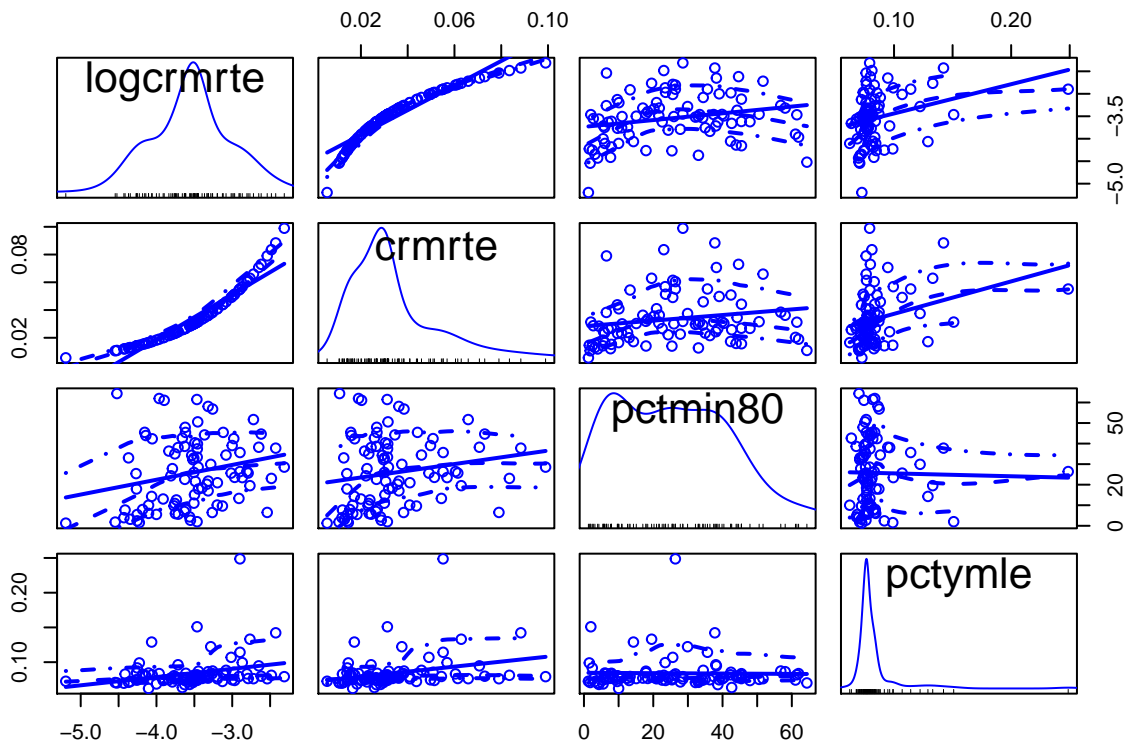
round(cor(crime[foo5rows]), 3)

##          logcrmrte crmrte pctmin80 pctymle
## logcrmrte      1.000  0.942    0.233  0.278
## crmrte         0.942  1.000    0.182  0.290
## pctmin80       0.233  0.182    1.000 -0.019
## pctymle        0.278  0.290   -0.019  1.000

scatterplotMatrix(crime[,foo5rows], diagonal = "histogram")

## Warning in applyDefaults(diagonal, defaults = list(method =
## "adaptiveDensity"), : unnamed diag arguments, will be ignored

```





## 7. Crime types

The higher the ratio of face-to-face crimes ends up with fewer crimes. I suspect this is the result of a small police force that doesn't have as much time to go after less significant crimes, so I added that variable in too. They're not strongly correlated.

```
# Crime Types
foo6log = lm(logcrmrte ~ mix + polpc, data = crime)
foo6log$coefficients

## (Intercept)          mix          polpc
## -3.4461127 -0.8393071  7.4322745

#vcov(foo5log)

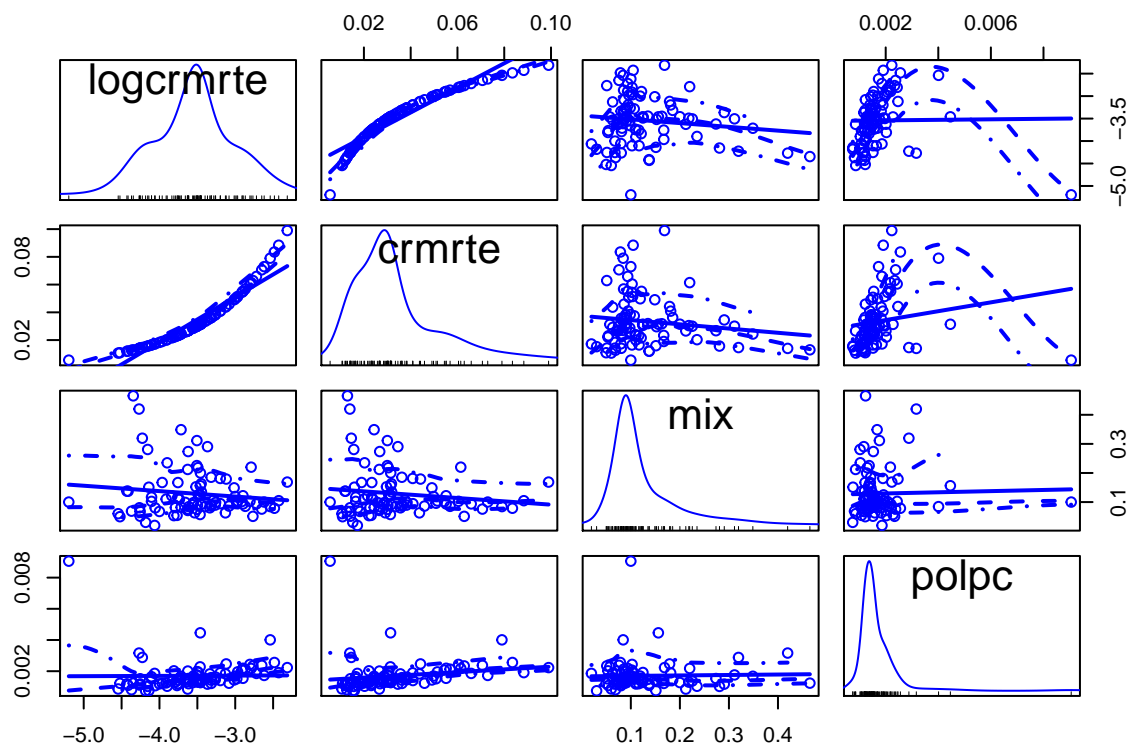
foo6rows = c("logcrmrte", "crmrte", "mix", "polpc")

round(cor(crime[foo6rows]), 3)

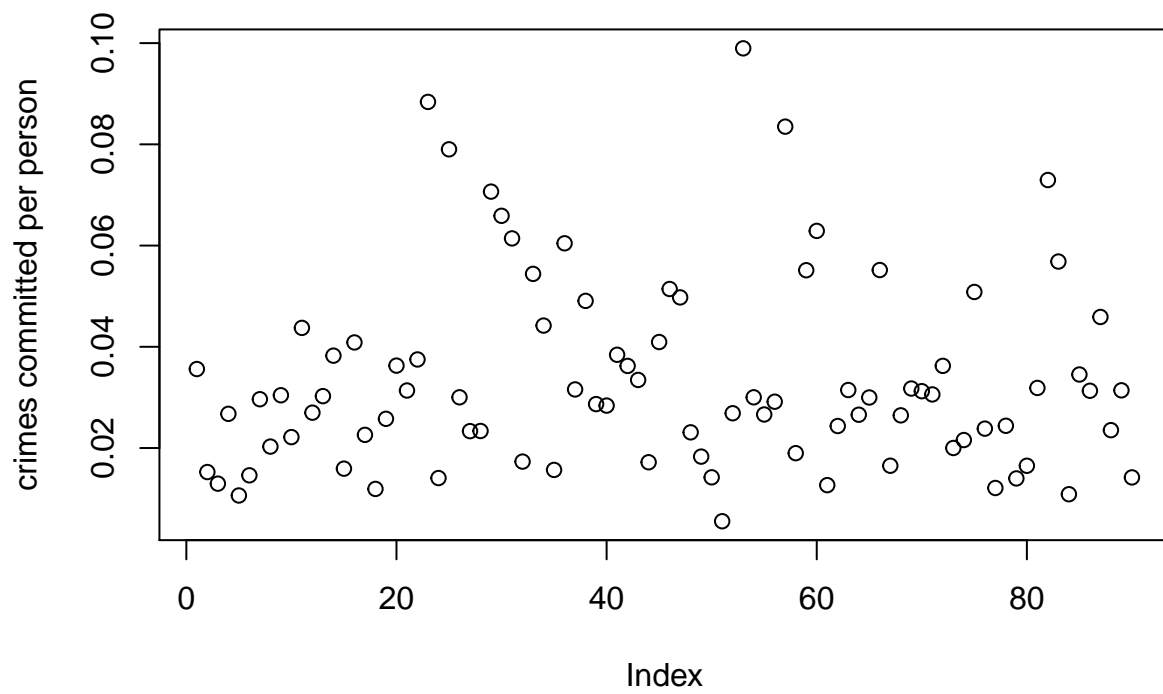
##          logcrmrte crmrte      mix polpc
## logcrmrte      1.000  0.942 -0.125 0.010
## crmrte         0.942  1.000 -0.132 0.167
## mix            -0.125 -0.132  1.000 0.024
## polpc          0.010  0.167  0.024 1.000

scatterplotMatrix(crime[,foo6rows], diagonal = "histogram")

## Warning in applyDefaults(diagonal, defaults = list(method =
## "adaptiveDensity"), : unnamed diag arguments, will be ignored
```

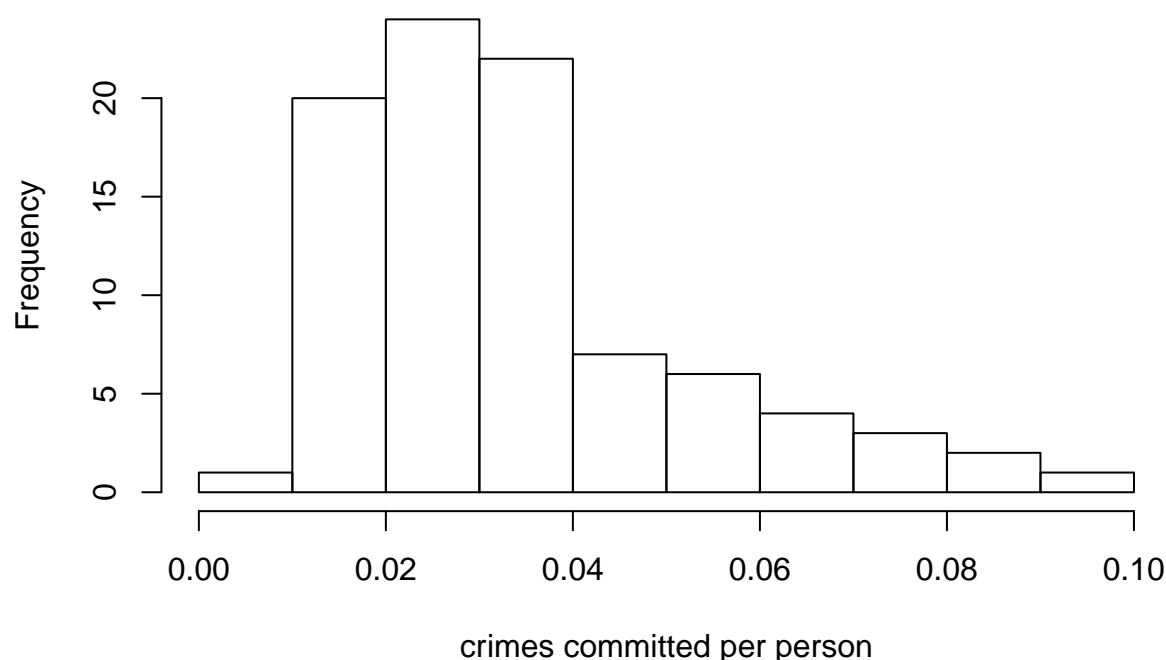


```
plot(crime$crmte, ylab = 'crimes committed per person')
```



```
hist(crime$crmrte, xlab = 'crimes committed per person', main = 'Histogram of crimes committed per person')
```

## Histogram of crimes committed per person



```
modell1 <- lm(crmrte ~ prbarr + polpc + density, data = crime)
(modell1$coefficients)
```

```
## (Intercept)      prbarr      polpc      density
## 0.028231799 -0.040443974 3.738309135 0.007546142
```

## Steps for evaluating variables

Leverage (and Influence if required)

Goodness-of-Fit : AIC

Omitted variable bias

MSE

$E[\text{theta hat}] = \text{theta}$

```
crime$urban + crime$west + crime$central
```

```
## [1] 1 1 1 1 1 1 0 0 0 0 2 1 1 1 1 1 1 0 1 0 0 1 0 0 1 1 0 2 0 2 1 2 1 0
## [36] 2 0 0 1 1 0 0 1 1 0 1 0 1 1 1 1 0 2 1 1 0 1 0 0 1 0 0 0 0 1 0 1 1 1 0
## [71] 1 1 1 0 0 1 1 1 1 1 1 2 1 0 1 0 1 0 1
```

```
crime$urban + crime$west
```

```
## [1] 0 0 1 0 1 1 0 0 0 0 2 1 0 1 0 0 0 1 0 0 0 0 1 0 0 0 0 0 1 0 1 0 1 0 0
## [36] 1 0 0 1 1 0 0 0 1 0 0 0 0 1 1 1 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0
## [71] 0 0 1 0 0 0 0 1 1 1 0 0 1 0 0 1 0 1 0 1
```

```
crime$urban + crime$central
```

```
## [1] 1 1 0 1 0 0 0 0 0 0 1 0 1 0 1 1 1 0 0 1 0 0 1 0 0 1 1 0 2 0 2 1 1 1 0
## [36] 2 0 0 0 0 0 0 1 0 0 1 0 1 0 0 0 0 2 1 1 0 1 0 0 1 0 0 0 0 1 0 0 1 1 0
## [71] 1 1 0 0 0 1 1 0 0 0 1 1 2 1 0 0 0 0 0 0
```

```
crime$west + crime$central
```

```
## [1] 1 1 1 1 1 1 0 0 0 0 1 1 1 1 1 1 1 0 1 0 0 0 0 0 1 1 0 1 0 1 1 2 1 0
## [36] 1 0 0 1 1 0 0 1 1 0 1 0 1 1 1 1 0 1 1 1 0 0 0 0 1 0 0 0 0 1 0 1 1 1 0
## [71] 1 1 1 0 0 1 1 1 1 1 1 1 1 1 0 1 0 1 0 1
```