Datta_Saurav_Lab3_Draft

Saurav Datta

3/24/2018

```r
#install.packages("sqldf")
# Sys.setenv(JAVA_HOME='/Library/Java/JavaVirtualMachines/jdk1.8.0_151.jdk/Contents/Home')
# install.packages("rJava")
# install.packages("RH2")
#install.packages("gridExtra")
```

```r
library(sqldf)
```

```
## Loading required package: gsubfn
```

```
## Warning: package 'gsubfn' was built under R version 3.4.4
```

```
## Loading required package: proto
```

```
## Loading required package: RSQLite
```

```r
# library(RH2)
library(ggplot2)
library(gridExtra)
library(stargazer)
```

```
##
## Please cite as:
```

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

```
##  R package version 5.2.1. https://CRAN.R-project.org/package=stargazer
```

```r
library(car)
```

```r
getwd()
```

```
## [1] "/Users/sdatta/Documents/1. Personal/MIDS/W203/Course material/Lab3"
```

```r
setwd("/Users/sdatta/Documents/1. Personal/MIDS/W203/Course material/Lab3")
```

```r
#db = dbConnect(SQLite(), dbname="lab3.sqllite")
#sqldf("attach 'lab3.sqllite' as new")
```

```r
#dbRemoveTable(db, "crime0")
```

```r
crime0=read.csv("crime_v2.csv",
                    header = TRUE
                  )
crime1=crime0
```

```r
sqldf("select * from crime1 limit 5")
```

```
##   county year    crmrte   prbarr     prbconv  prbpris avgsen      polpc
## 1      1   87 0.0356036 0.298270 0.527595997 0.436170   6.71 0.00182786
## 2      3   87 0.0152532 0.132029 1.481480002 0.450000   6.35 0.00074588
## 3      5   87 0.0129603 0.444444 0.267856985 0.600000   6.76 0.00123431
## 4      7   87 0.0267532 0.364760 0.525424004 0.435484   7.14 0.00152994
## 5      9   87 0.0106232 0.518219 0.476563007 0.442623   8.22 0.00086018
```

```
##     density    taxpc west central urban pctmin80     wcon     wtuc
## 1 2.4226327 30.99368    0       1     0 20.21870 281.4259 408.7245
## 2 1.0463320 26.89208    0       1     0  7.91632 255.1020 376.2542
## 3 0.4127659 34.81605    1       0     0  3.16053 226.9470 372.2084
## 4 0.4915572 42.94759    0       1     0 47.91610 375.2345 397.6901
## 5 0.5469484 28.05474    1       0     0  1.79619 292.3077 377.3126
##       wtrd     wfir     wser   wmfg   wfed   wsta   wloc        mix
## 1 221.2701 453.1722 274.1775 334.54 477.58 292.09 311.91 0.08016878
## 2 196.0101 258.5650 192.3077 300.38 409.83 362.96 301.47 0.03022670
## 3 229.3209 305.9441 209.6972 237.65 358.98 331.53 281.37 0.46511629
## 4 191.1720 281.0651 256.7214 281.80 412.15 328.27 299.03 0.27362204
## 5 206.8215 289.3125 215.1933 290.89 377.35 367.23 342.82 0.06008584
##      pctymle
## 1 0.07787097
## 2 0.08260694
## 3 0.07211538
## 4 0.07353726
## 5 0.07069755
```

Converting prbconv from factor to numeric

We see that column prbconv is factor datatype

```
crime1$prbconv_cast=as.numeric(as.matrix(crime1$prbconv))
```

```
## Warning: NAs introduced by coercion
```

```
crime_tmp=sqldf("SELECT * FROM crime1 WHERE NOT ( prbconv_cast<0 OR  prbarr<0 or prbpris<0)")
crime1=crime_tmp
sqldf("SELECT count(*) from crime1")
```

```
##   count(*)
## 1       91
```

Defining common function

```
f_check_null <- function(in_field_name ){
  sql=sprintf("SELECT COUNT(1) as COUNT_NULL_OR_NA FROM crime1 WHERE (%s IS \"NA\" or %s IS NULL)", in_
  sqldf(sql)
}
```

```
f_plot_one <- function(in_db_field_name,in_main_title ){

  title_log=paste("log of",in_main_title, sep = " ")

  par(mfrow=c(2,2))
  hist(in_db_field_name, main=in_main_title)
  hist(log(in_db_field_name), main=title_log)
  boxplot(in_db_field_name, main=in_main_title)

}
```

```
f_plot_two <- function(in_field_name1,in_xlabel,in_field_name2,in_y_label, in_main_title ){


  theme_update(plot.title = element_text(hjust = 0.5))
```

```
  p1<-ggplot(crime1, aes_string(in_field_name1,in_field_name2)) +
        geom_point() +
        geom_smooth(na.rm = FALSE, method = loess)
  p1 + ggtitle(in_main_title) +xlab(in_xlabel) + ylab(in_y_label)
}
```

```
f_plot_three  <- function(in_field_x,in_xlabel,in_field_y,in_y_label){

corr_val=round(cor(in_field_y, in_field_x),4)

main_title=paste(in_xlabel,'v/s',in_y_label, sep = ' ')
plot(in_field_x, in_field_y,
     main = main_title,
      sub=paste("Corr. coefficient:",corr_val),
      xlab=in_xlabel,
      ylab=in_y_label)
m = lm( in_field_y ~ in_field_x)
abline(m)

}
```

```
sqldf("select count(8) from crime1 where west=1  and central=1")
```

```
##   count(8)
## 1        1
```

We see that the same county is marked as both west and central.

Analyzing regions

```
crime_tmp = sqldf("SELECT *, CASE WHEN urban=1 THEN \'URBAN\'
                                ELSE \'RURAL\'
                          END AS 'urbanorrural'
                FROM crime1"
)

##END regionofcrime
```

```
crime1=crime_tmp
```

```
sqldf("SELECT urbanorrural as urbanorrural, count(8) as countofcrimes from crime1 GROUP BY urbanorrural
     )
```

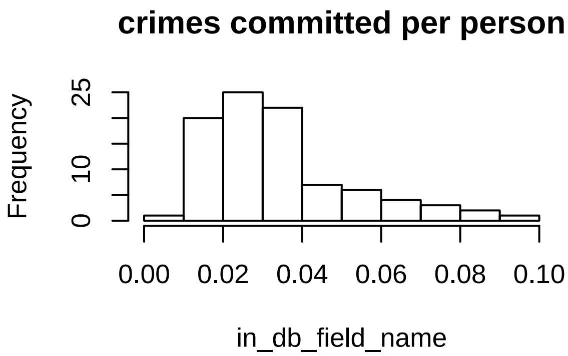
```
##   urbanorrural countofcrimes
## 1        RURAL            83
## 2        URBAN             8
```

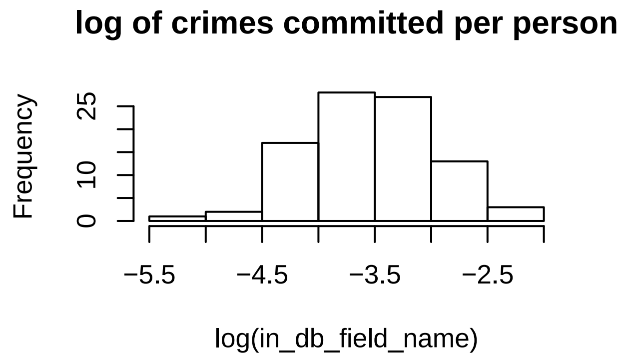
Analyzing crmrte

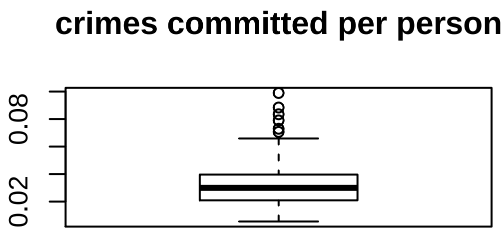
```
f_check_null("crmrte")
```

```
##   COUNT_NULL_OR_NA
## 1                0
```

```
f_plot_one(crime1$crmrte,"crimes committed per person")
crime1$logcrmrte=log(crime1$crmrte)
```







Analyzing the 6 records with missing crmrte values

```
sqldf("SELECT * FROM crime1 WHERE (crmrte IS \"NA\" or crmrte IS NULL) ")
```

```
##  [1] county        year          crmrte        prbarr        prbconv
##  [6] prbpris       avgsen        polpc         density       taxpc
## [11] west          central       urban         pctmin80      wcon
## [16] wtuc          wtrd          wfir          wser          wmfg
## [21] wfed          wsta          wloc          mix           pctymle
## [26] prbconv_cast urbanorrural logcrmrte
## <0 rows> (or 0-length row.names)
```

We see that all relevant columns of these 6 records are NA. So we can safely delete them

```
crime_tmp=sqldf( c("DELETE FROM crime1 WHERE crmrte IS NULL",
         "SELECT * FROM crime1"
         )
)
```

```
## Warning in rsqlite_fetch(res@ptr, n = n): Don't need to call dbFetch() for
## statements, only for queries
```

```
crime1=crime_tmp
sqldf("SELECT count(*) FROM crime1 ")
```

```
##   count(*)
## 1       91
```

Reanalyzing regions after deleting NAs

```
sqldf("SELECT urbanorrural as urbanorrural, count(8) as countofcrimes from crime1 GROUP BY urbanorrural
     )
```

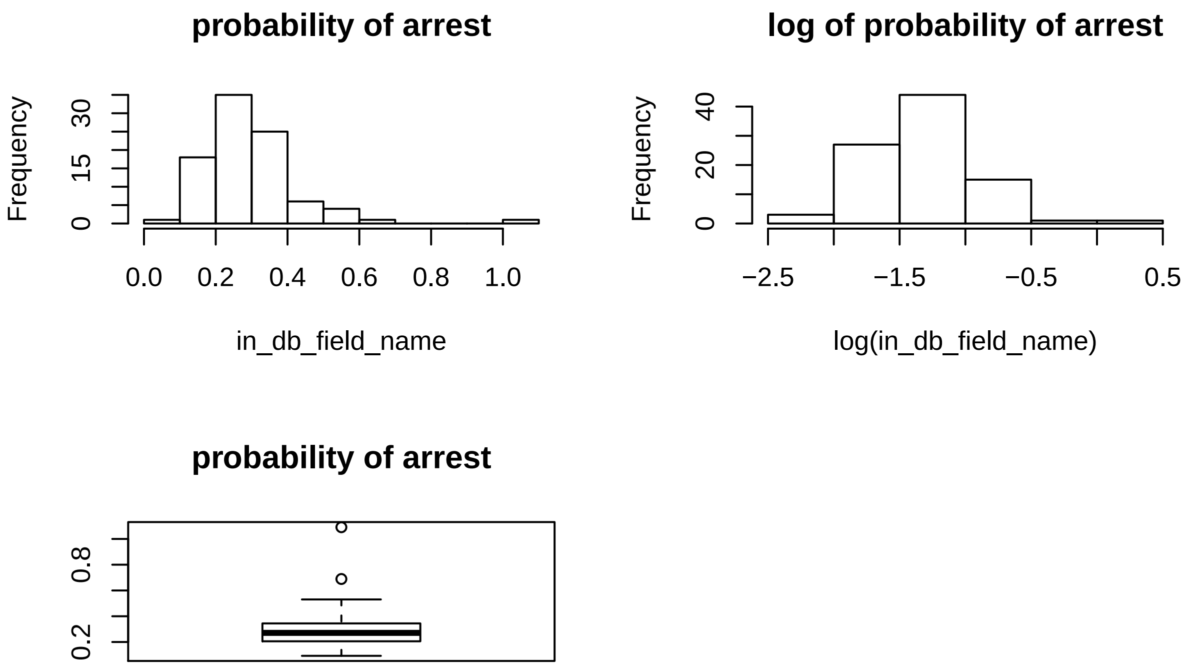
```
##   urbanorrural countofcrimes
## 1        RURAL            83
## 2        URBAN             8
```

Analyzing prbarr

```
f_check_null("prbarr")
```

```
##   COUNT_NULL_OR_NA
## 1                0
```

```
f_plot_one(crime1$prbarr,"probability of arrest")
```

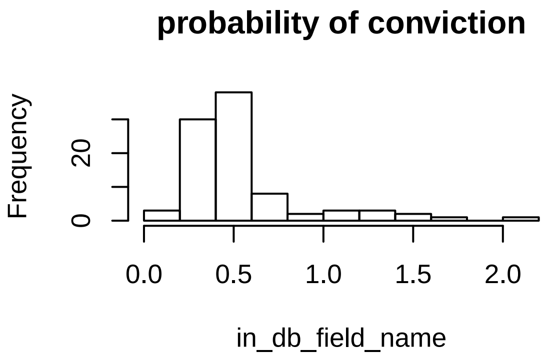


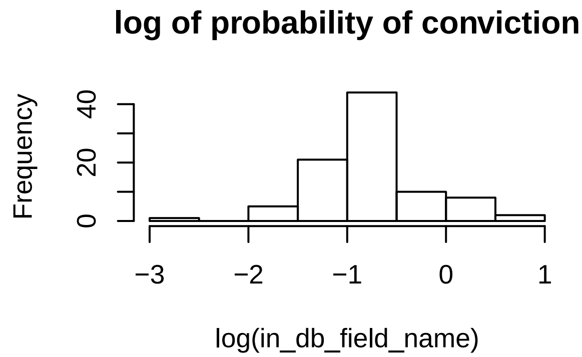
Analyzing prbconv_cast

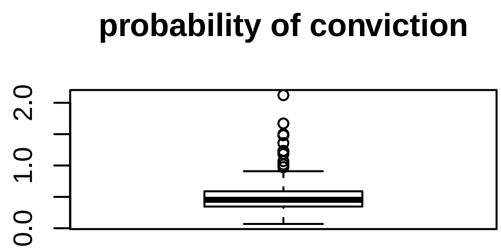
```
f_check_null("prbconv_cast")
```

```
##   COUNT_NULL_OR_NA
## 1                0
```

```
f_plot_one(crime1$prbconv_cast,"probability of conviction")
```





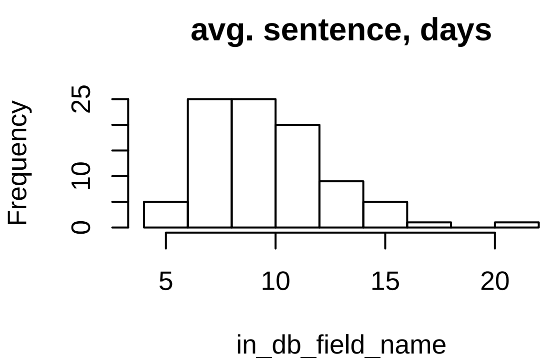


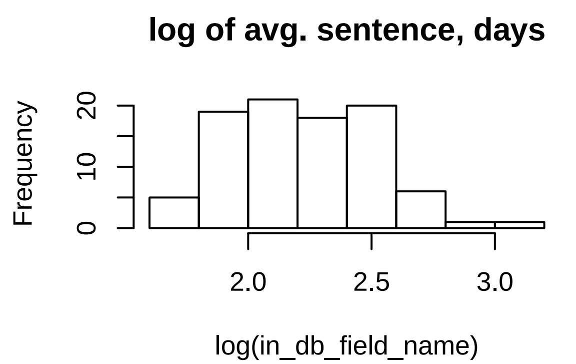
Analyzing avgsen

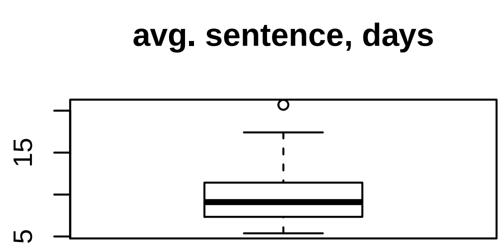
```
f_check_null("avgsen")
```

```
##   COUNT_NULL_OR_NA
## 1                0
```

```
f_plot_one(crime1$avgsen,"avg. sentence, days")
crime1$logavgsen =  log(crime1$avgsen)
```





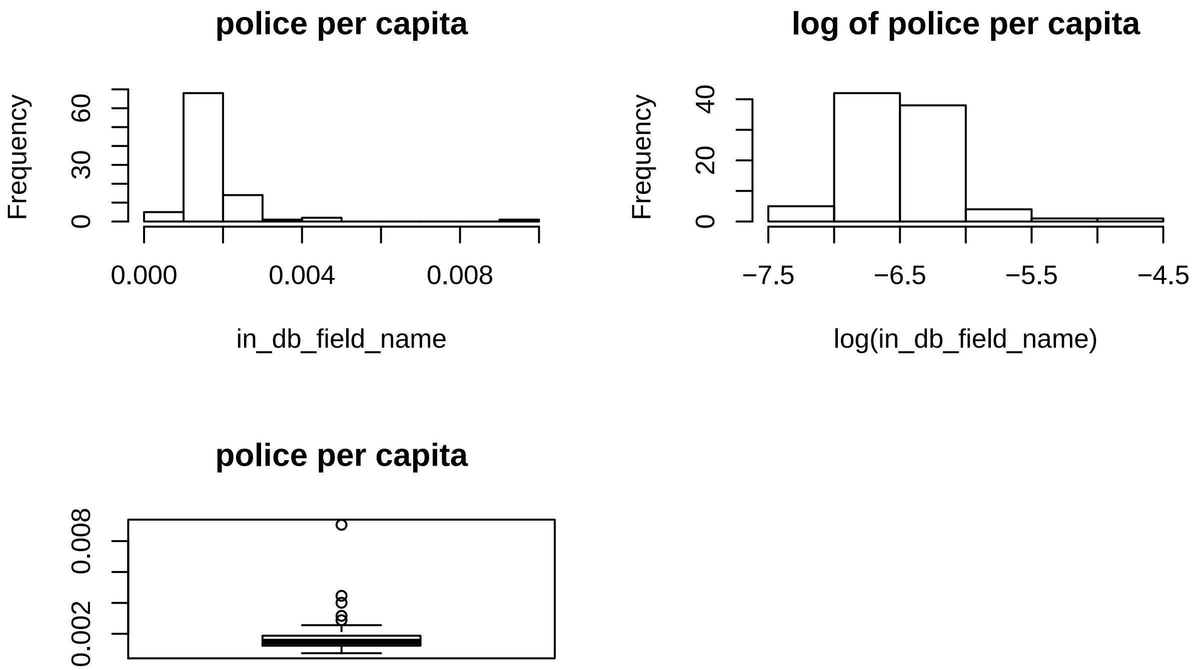


Analyzing polpc

```
f_check_null("polpc")
```

```
##   COUNT_NULL_OR_NA
## 1                0
```

```
f_plot_one(crime1$polpc,"police per capita")
crime1$logpolpc =  log(crime1$polpc)
```



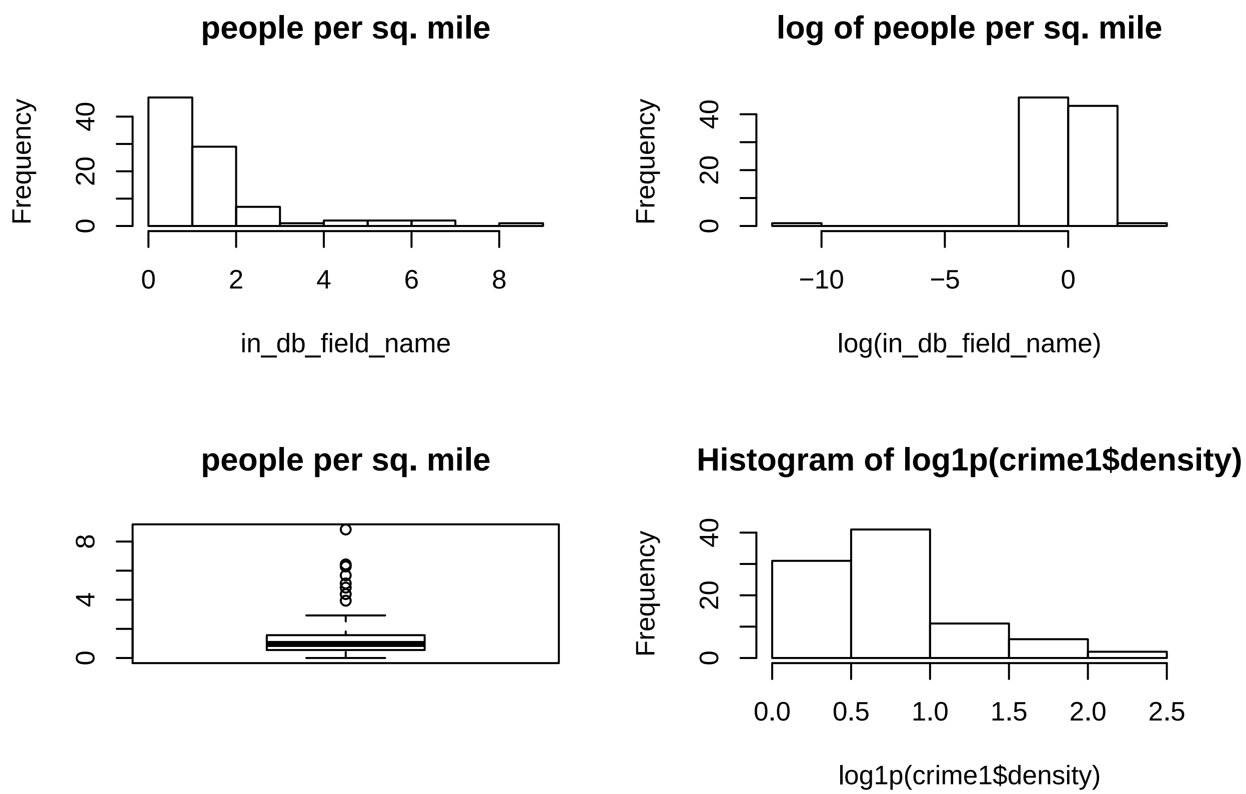
Analyzing density

We see that log1p of density is closer to normal distribution than either log or exp (tried it offline).

```
f_check_null("density")
```

```
##   COUNT_NULL_OR_NA
## 1                0
```

```
f_plot_one(crime1$density,"people per sq. mile")

hist(log1p(crime1$density))
```



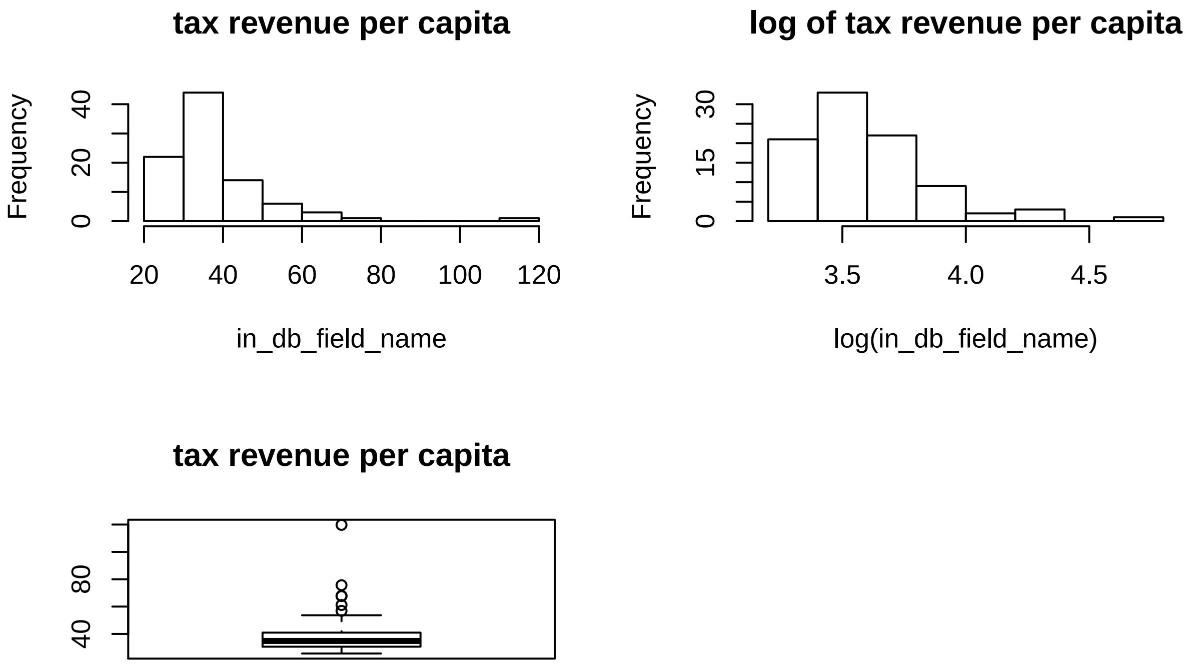

```
crime1$log1pdensity=log1p(crime1$density)
```

Analyzing taxpc

```
f_check_null("taxpc")
```

```
##   COUNT_NULL_OR_NA
## 1                0
```

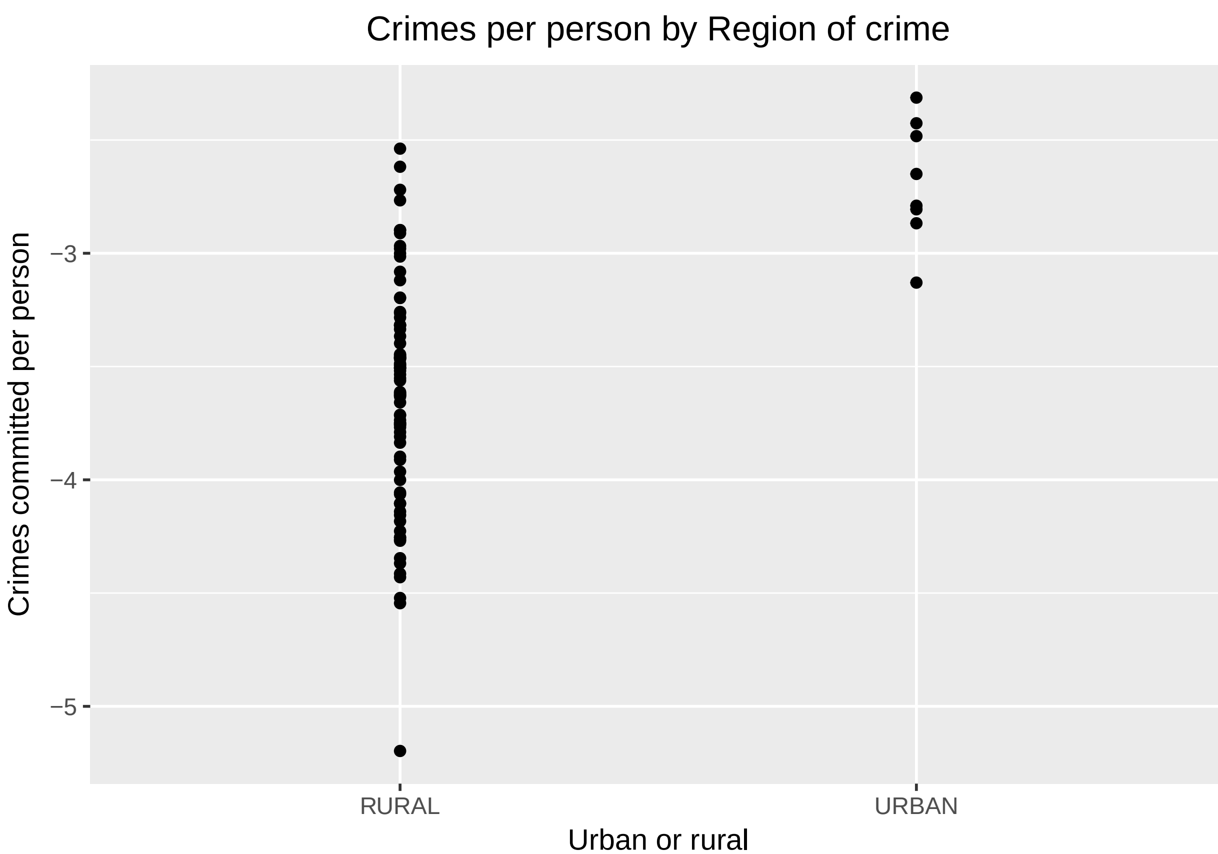
```
f_plot_one(crime1$taxpc,"tax revenue per capita")
```



Outlier of taxpc=120

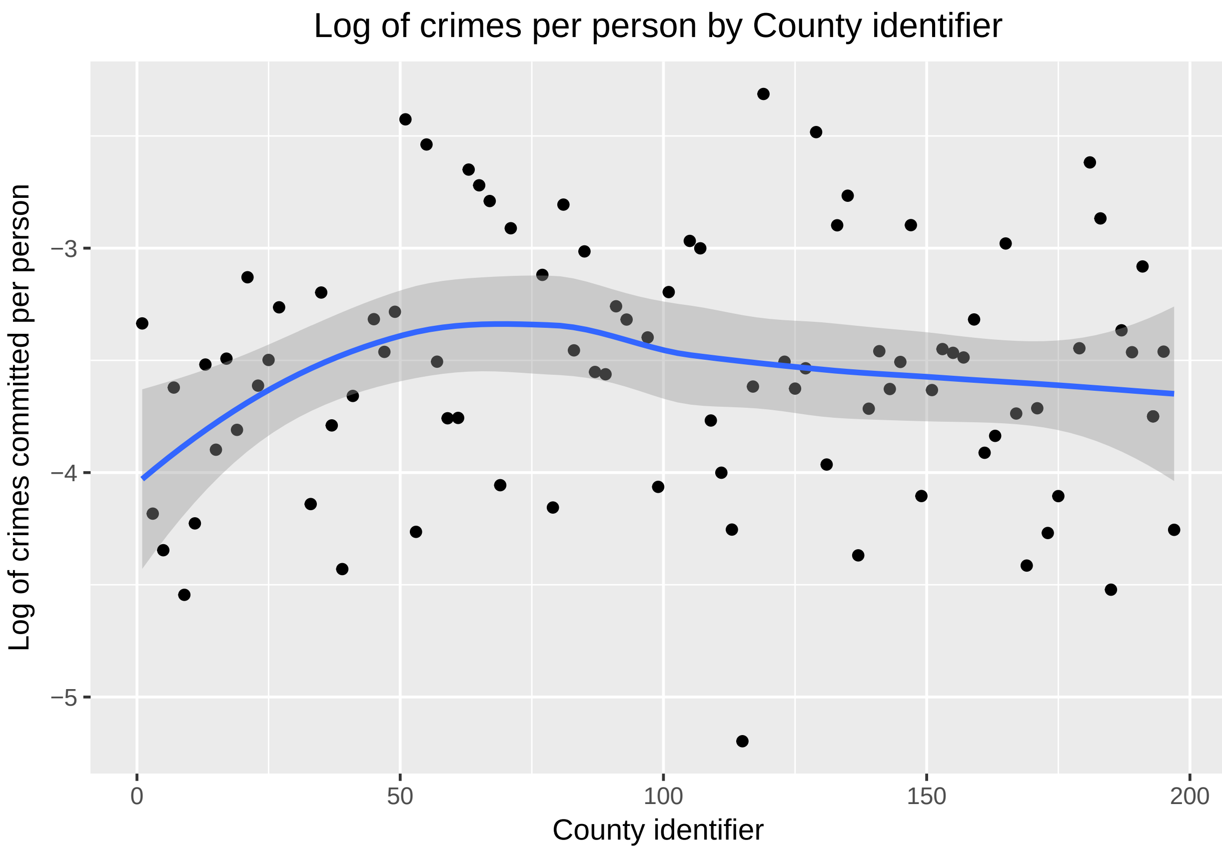
Analyzing Crimes committed per person by region

```
#
f_plot_two("urbanorrural","Urban or rural","logcrmrte","Crimes committed per person","Crimes per person
```



Analyzing Crimes committed per person by region

```
f_plot_two("county","County identifier","logcrmrte","Log of crimes committed per person","Log of crimes
```




```
sqldf("SELECT county,crmrte,logcrmrte FROM crime1 WHERE logcrmrte>=-2.5  ")
```

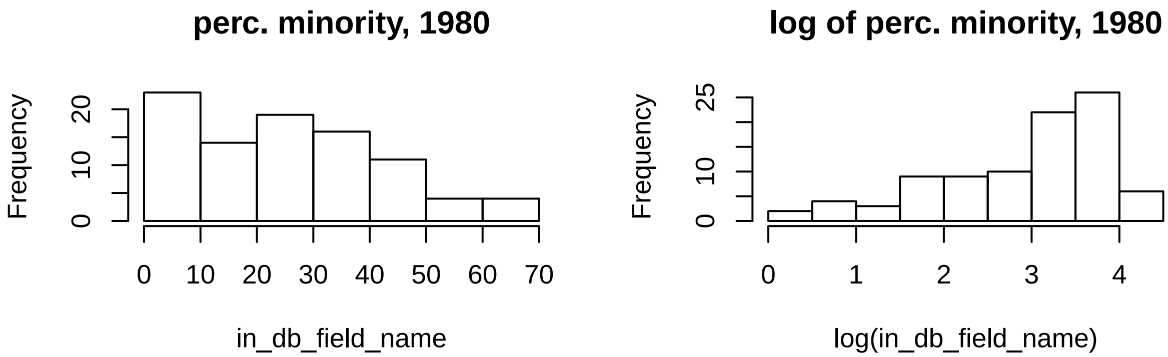
```
##   county    crmrte logcrmrte
## 1     51 0.0883849 -2.426054
## 2    119 0.0989659 -2.312980
## 3    129 0.0834982 -2.482930
```

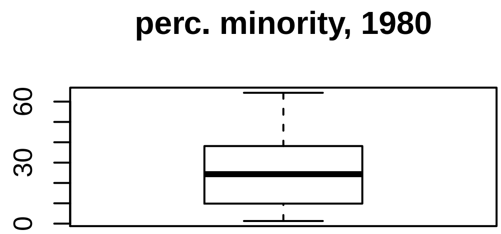
Analyzing percent minority

```
f_check_null("pctmin80")
```

```
##   COUNT_NULL_OR_NA
## 1                0
```

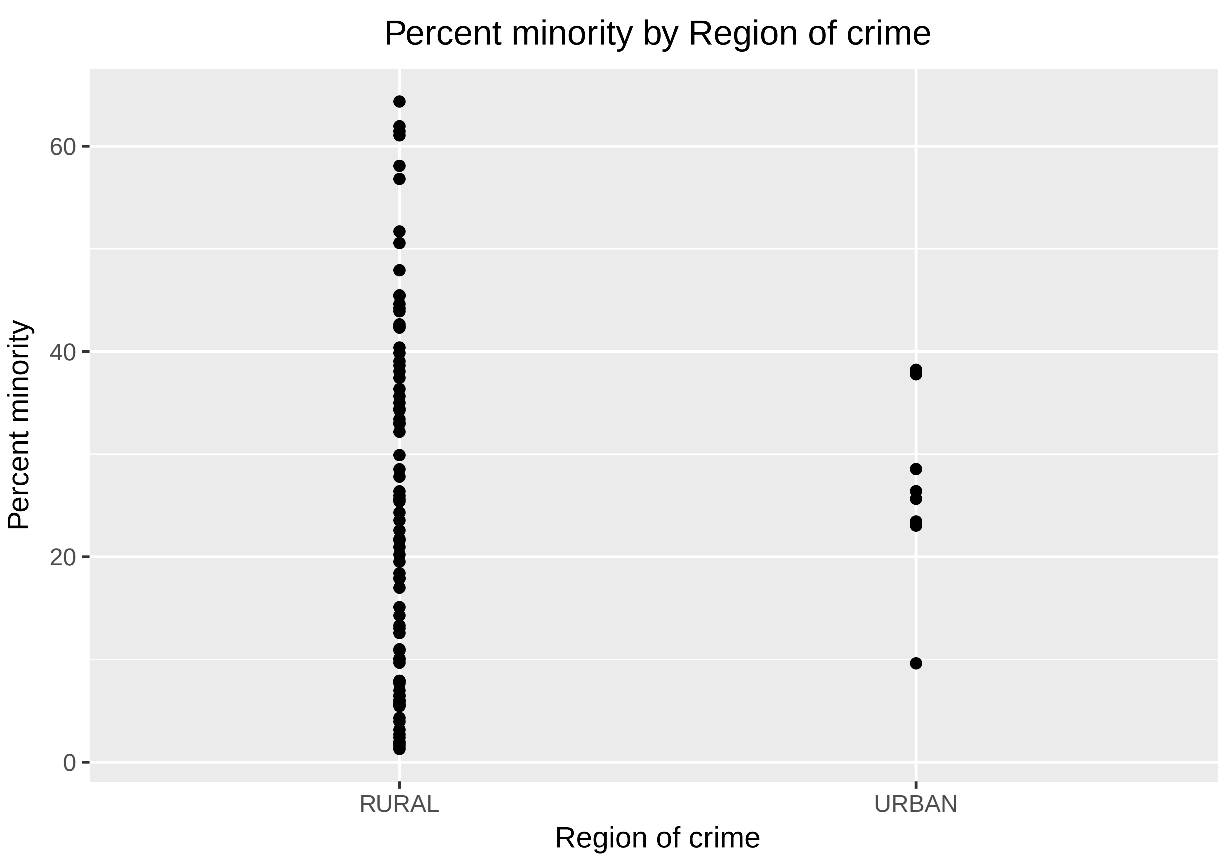
```
f_plot_one(crime1$pctmin80,"perc. minority, 1980")
```





Analyzing percent minority by region

```
f_plot_two("urbanorrural","Region of crime","pctmin80","Percent minority","Percent minority by Region o
```

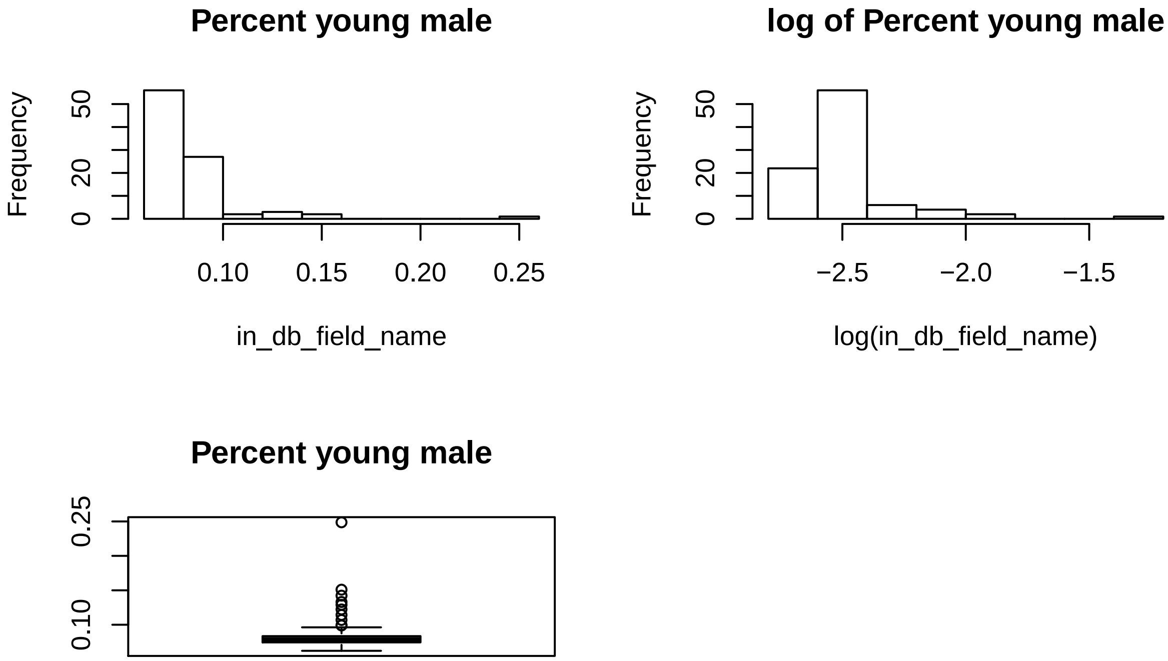


Analyzing pctymle

```
f_check_null("pctymle")
```

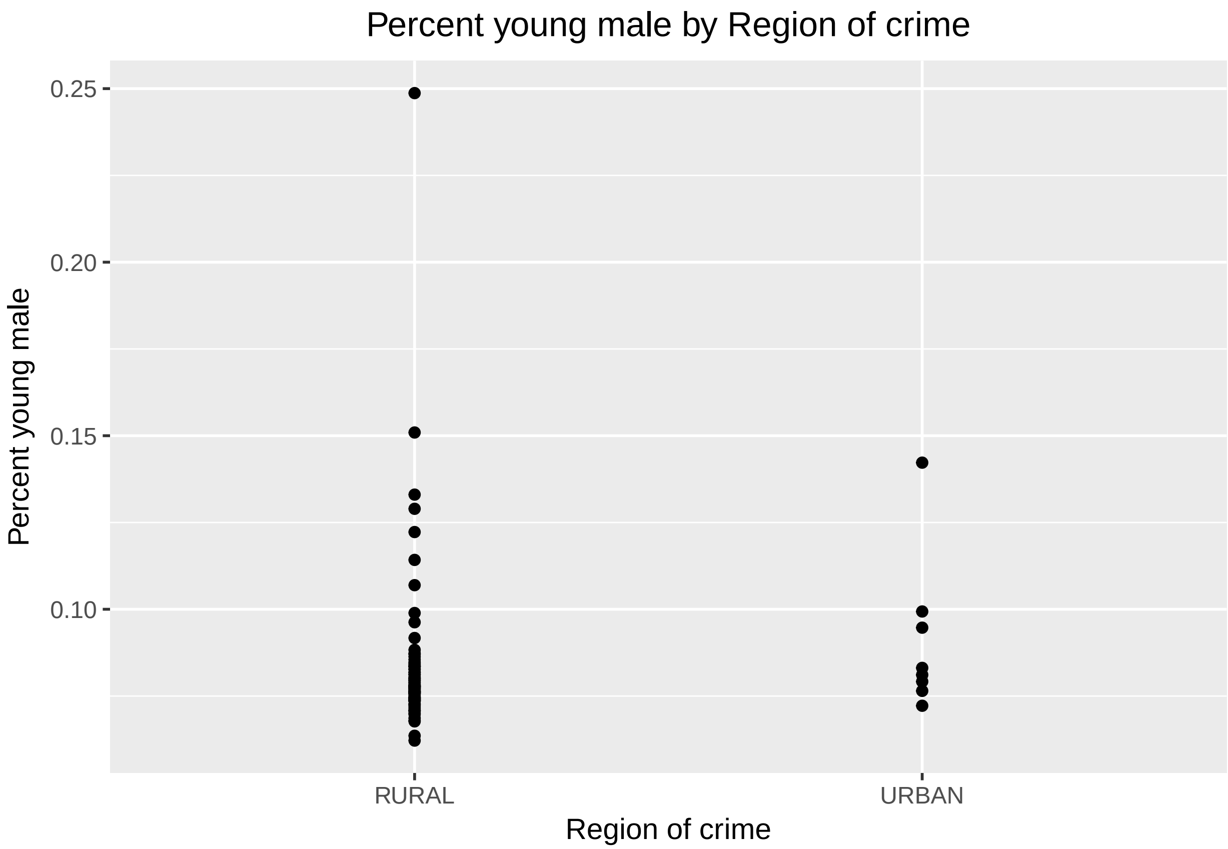
```
##   COUNT_NULL_OR_NA
## 1                0
```

```
f_plot_one(crime1$pctymle,"Percent young male")
```



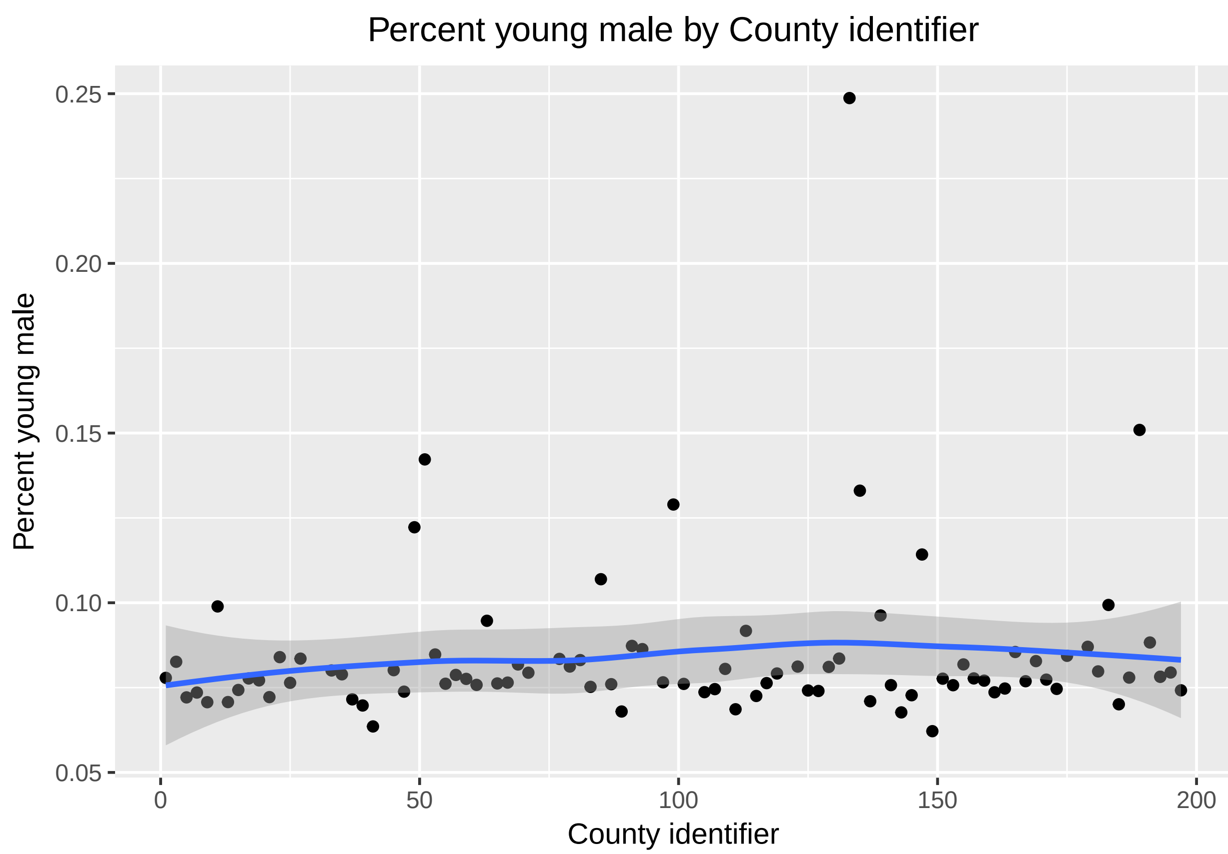
Analyzing pctymle by region

```
f_plot_two("urbanorrural","Region of crime","pctymle","Percent young male","Percent young male by Regio
```



We see that the UKNOWN region has the highest percent of young male

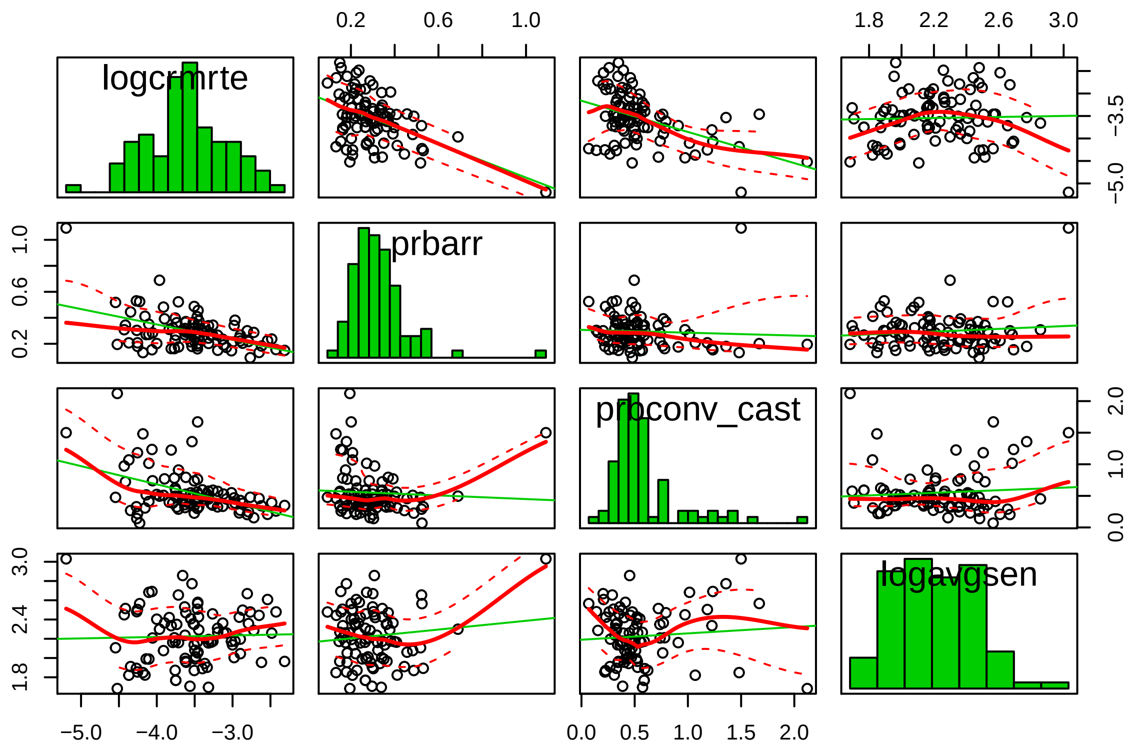
```
f_plot_two("county","County identifier","pctymle","Percent young male","Percent young male by County id
```



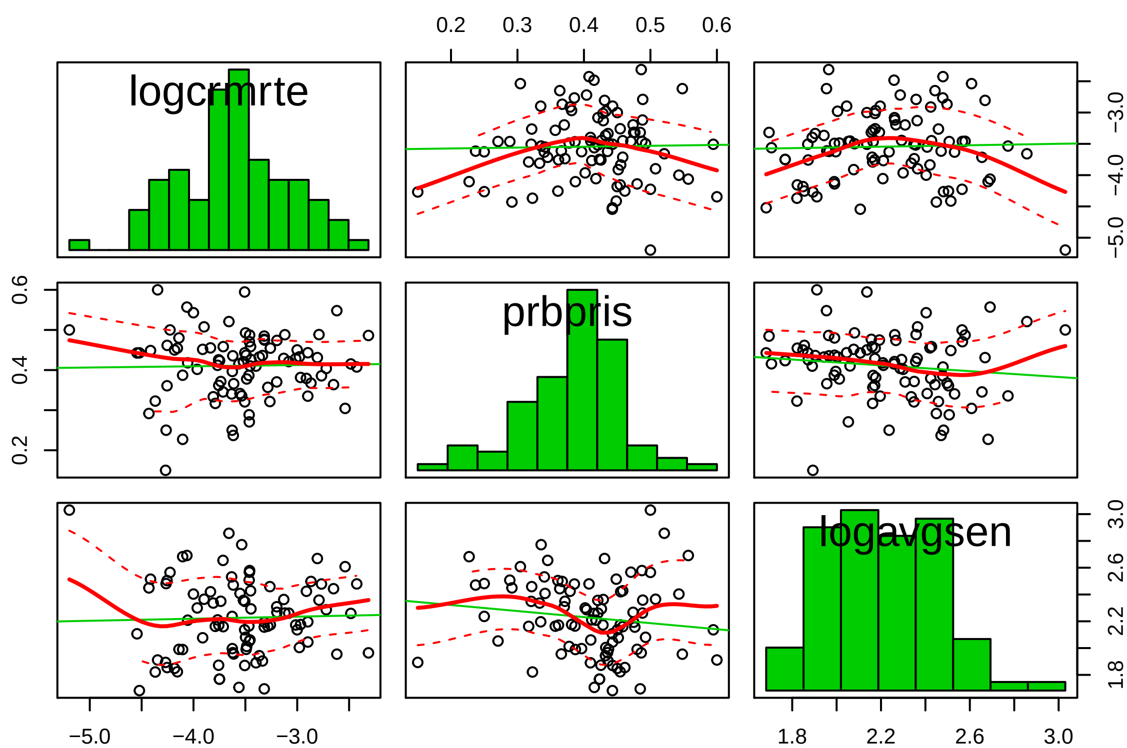

```r
sqldf("SELECT county,pctymle from crime1 WHERE pctymle>=0.20")
```

```
##   county   pctymle
## 1    133 0.2487116
```

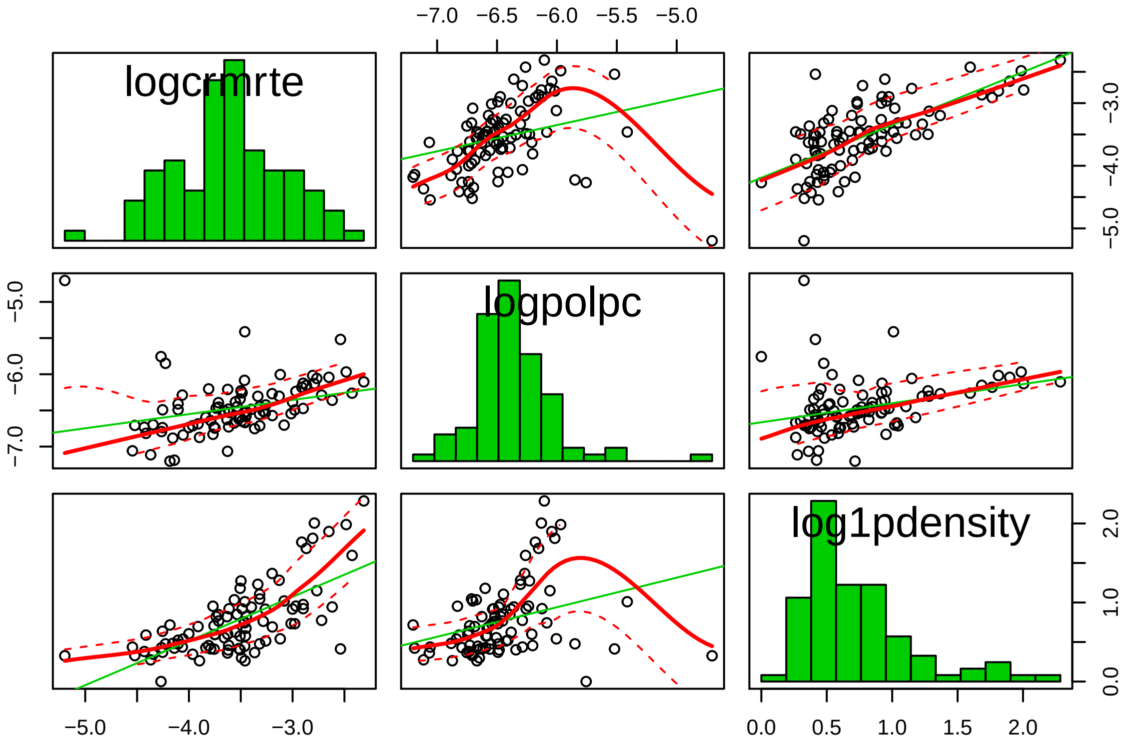
```r
scatterplotMatrix(crime1[,c("logcrmrte","prbarr",  "prbconv_cast", "logavgsen")], diagonal = "histogram
```



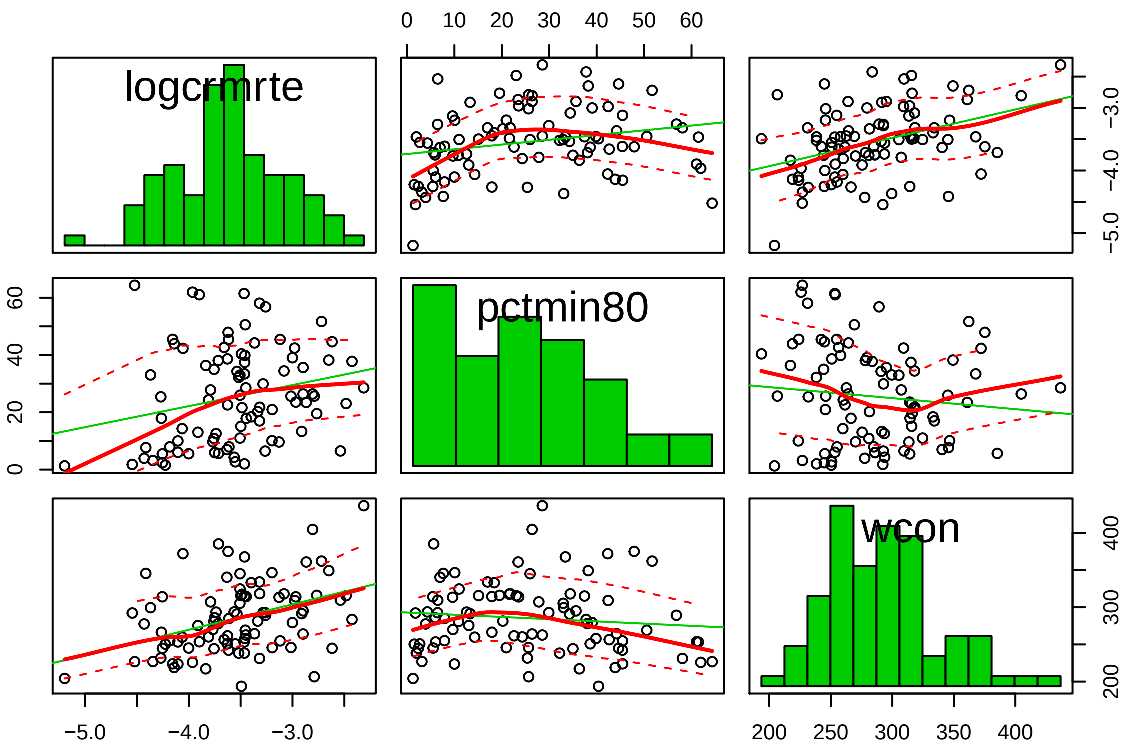

```r
scatterplotMatrix(crime1[,c("logcrmrte","prbpris",  "logavgsen")], diagonal = "histogram")
```



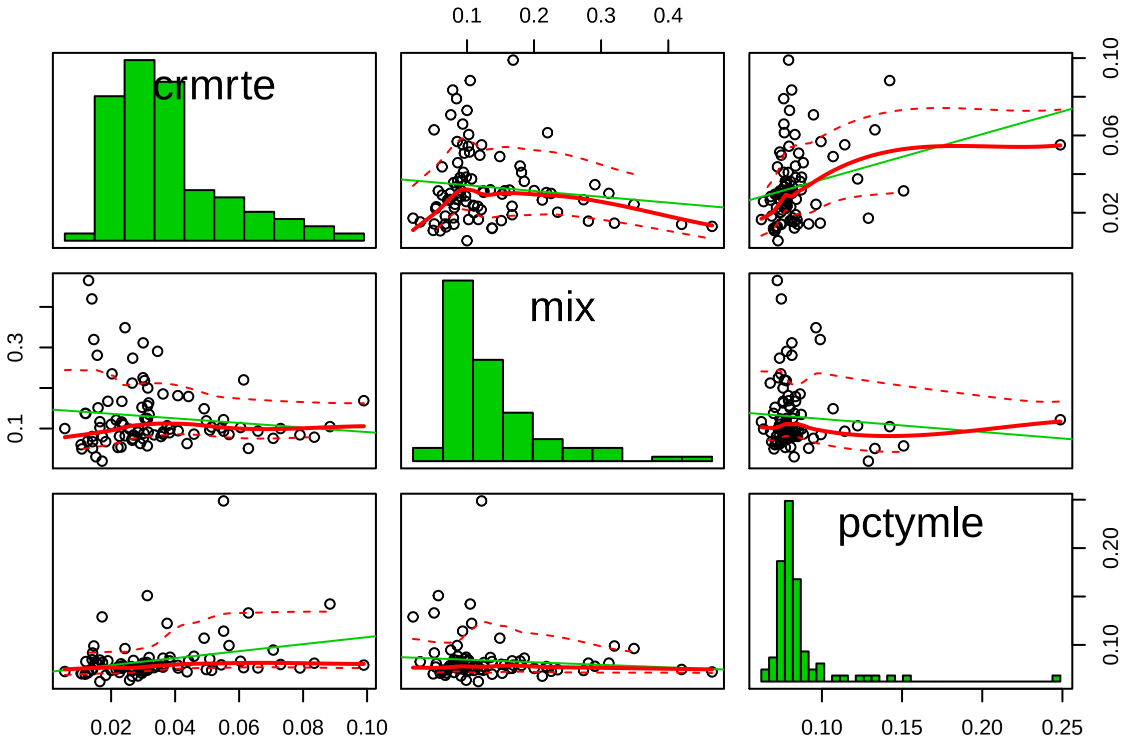

```r
scatterplotMatrix(crime1[,c("logcrmrte","logpolpc",  "log1pdensity")], diagonal = "histogram")
```



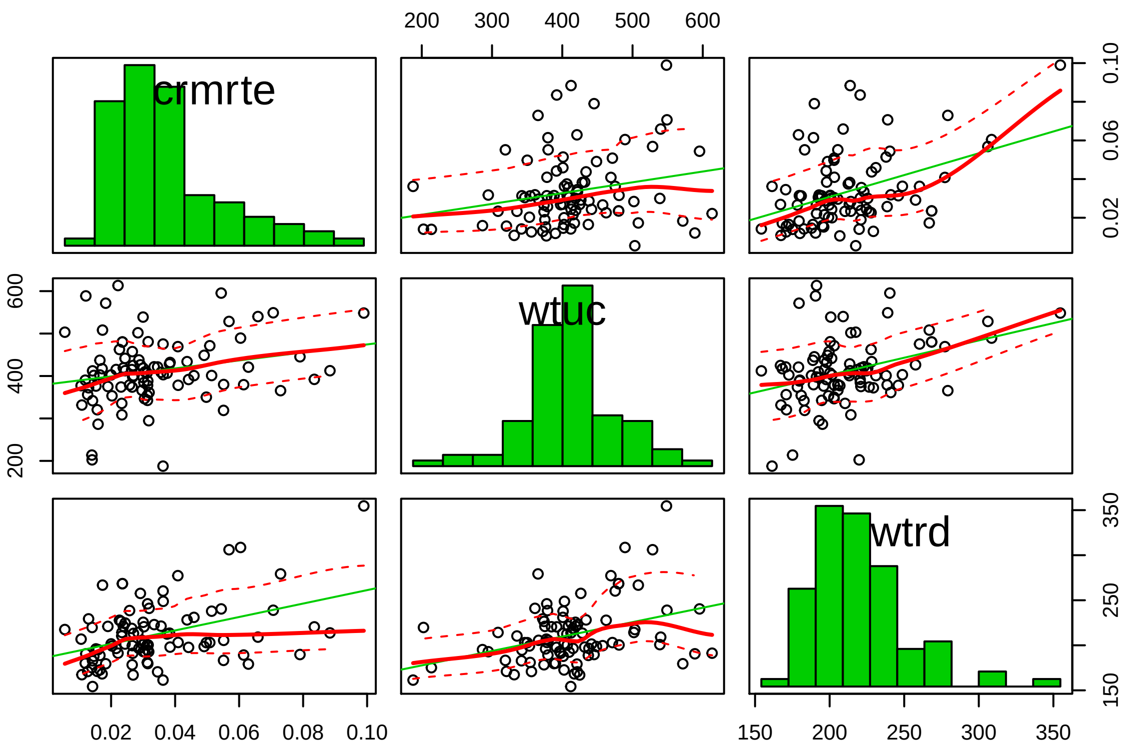

```
scatterplotMatrix(crime1[,c("logcrmrte","pctmin80",  "wcon")], diagonal = "histogram")
```



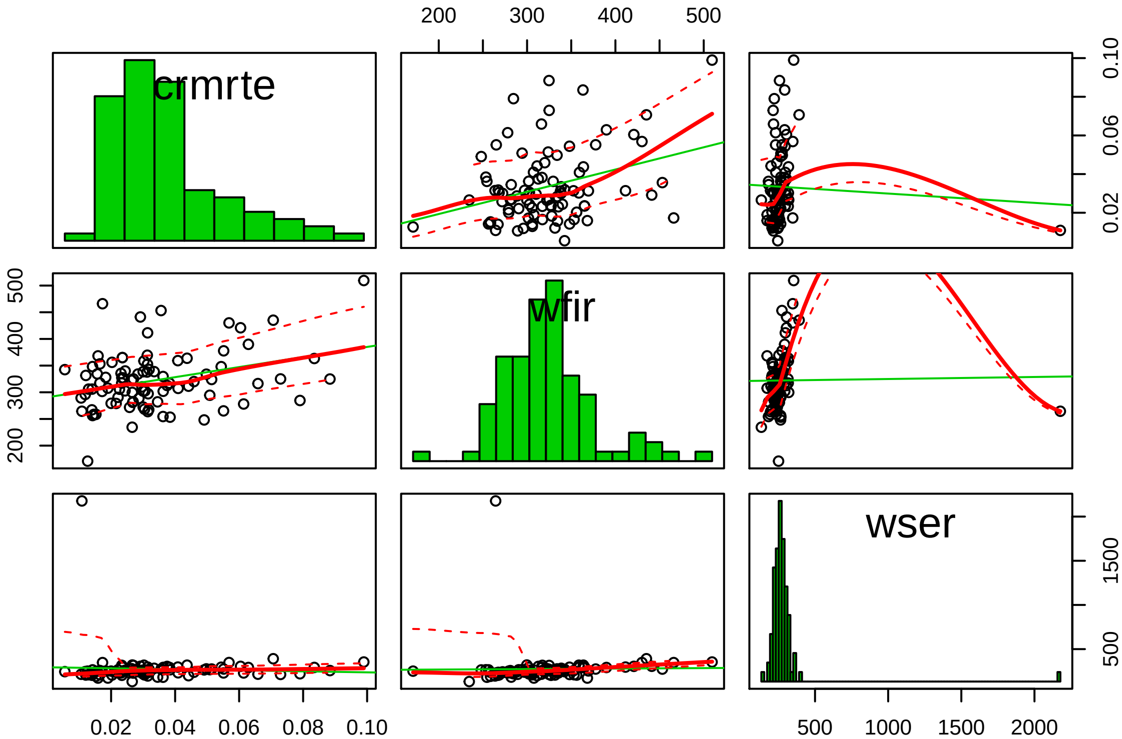

```
scatterplotMatrix(crime1[,c("crmrte","mix","pctymle")], diagonal = "histogram")
```



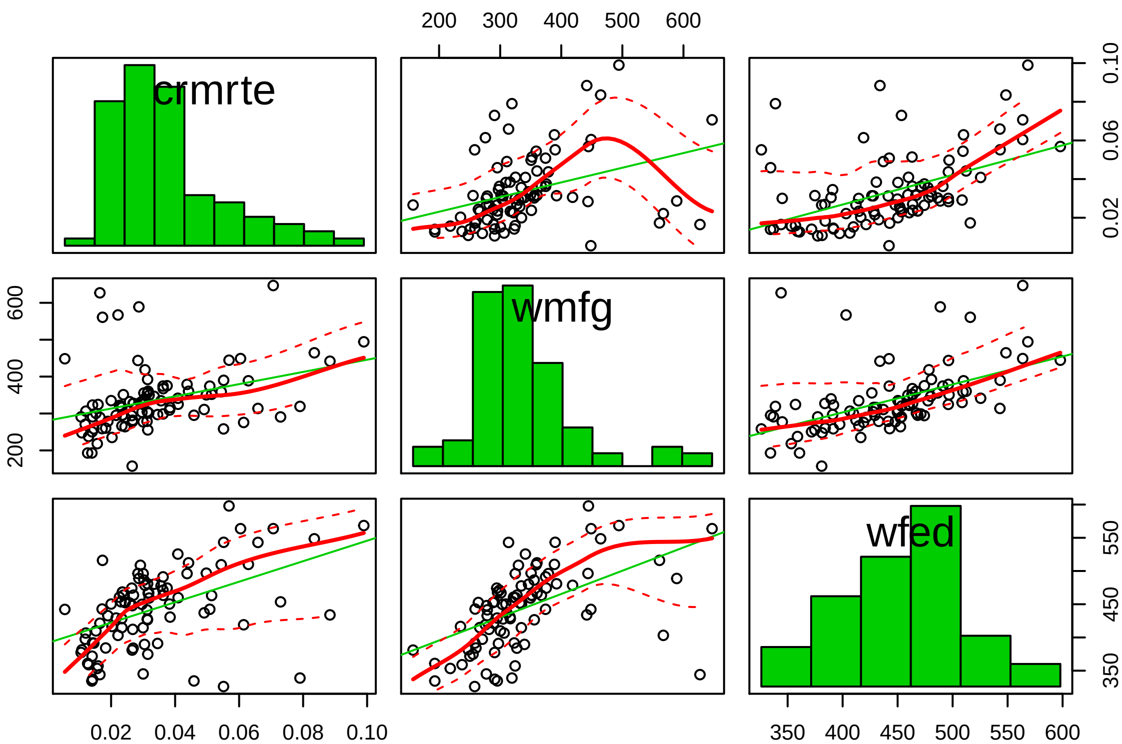

```
scatterplotMatrix(crime1[,c("crmrte","wtuc",  "wtrd")], diagonal = "histogram")
```



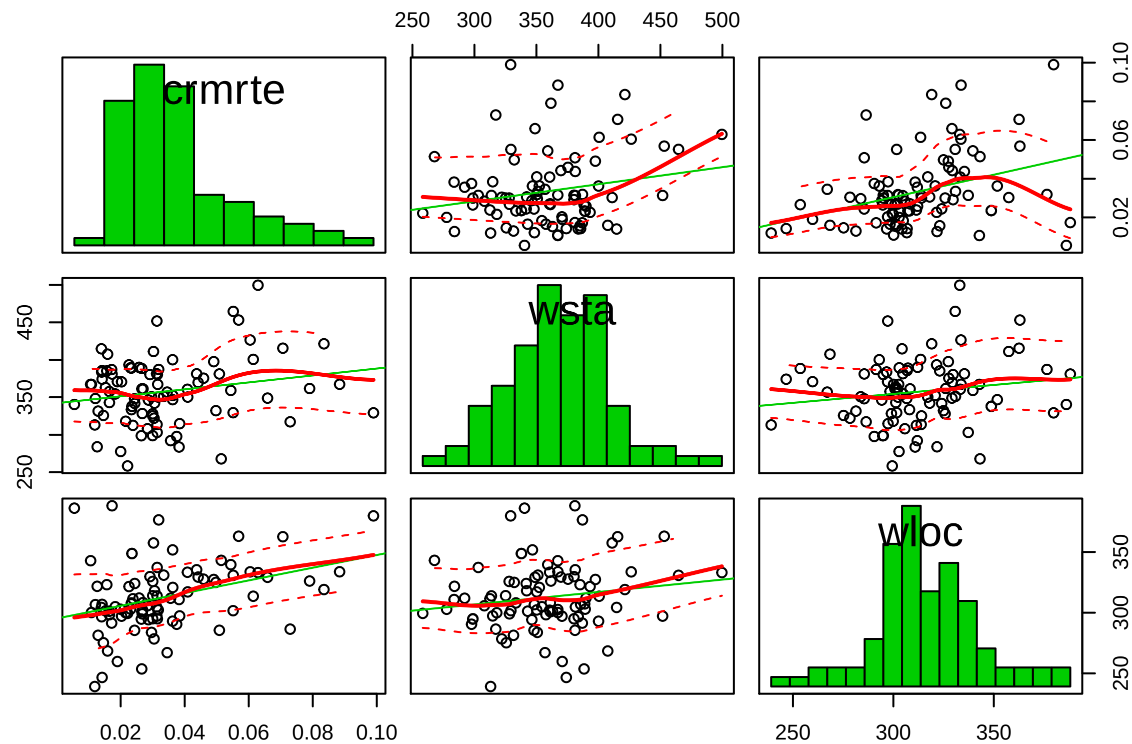

```
scatterplotMatrix(crime1[,c("crmrte","wfir",  "wser")], diagonal = "histogram")
```



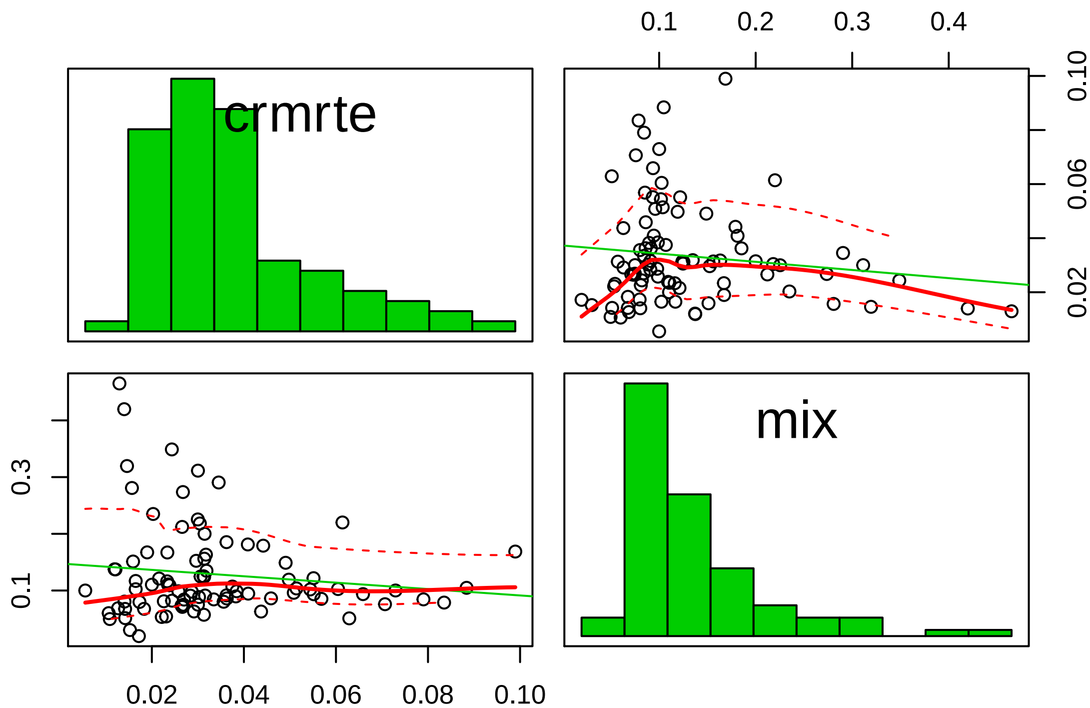

```
scatterplotMatrix(crime1[,c("crmrte","wmfg",  "wfed")], diagonal = "histogram")
```



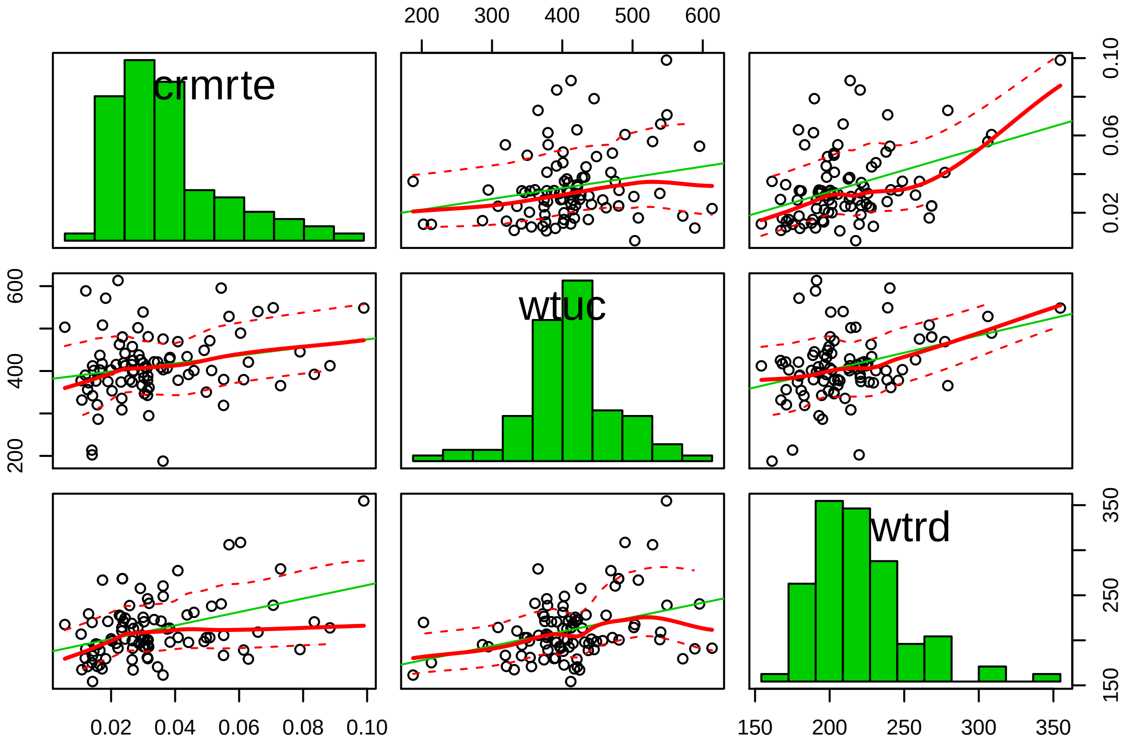

```
scatterplotMatrix(crime1[,c("crmrte","wsta",  "wloc")], diagonal = "histogram")
```



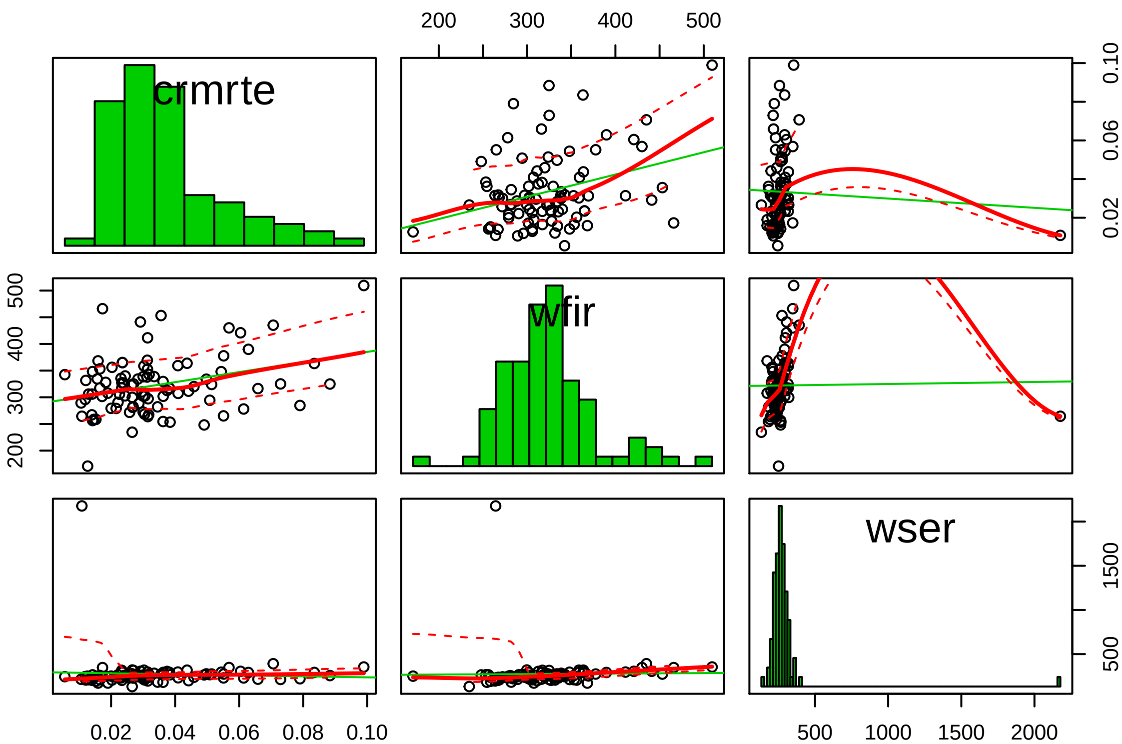

```
scatterplotMatrix(crime1[,c("crmrte","mix")], diagonal = "histogram")
```




```
scatterplotMatrix(crime1[,c("crmrte","wtuc",  "wtrd")], diagonal = "histogram")
```



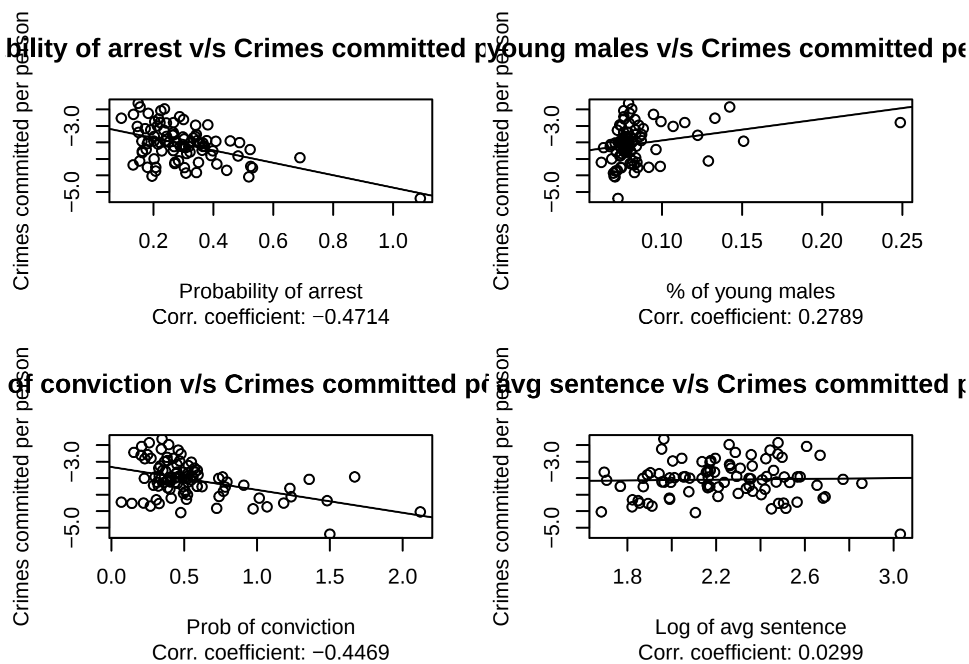

```
scatterplotMatrix(crime1[,c("crmrte","wfir",  "wser")], diagonal = "histogram")
```



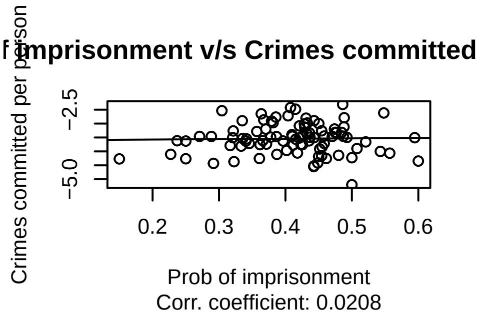

```
# corr_val=round(cor(crime1$logcrmrte, crime1$prbarr),4)
#
# main_title=paste(in_xlabel,'v/s',in_y_label, sep = ' ')
# plot(crime1$prbarr, crime1$logcrmrte,
#      main = main_title,
```

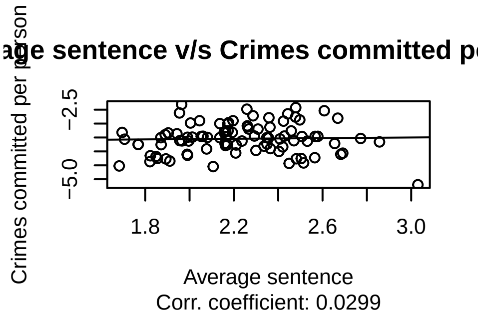
```r
#        sub=paste("Corr. coefficient:",corr_val),
#        xlab=in_xlabel,
#        ylab=in_y_label)
# m = lm(in_field_x ~ in_field_y)
# abline(m)
```

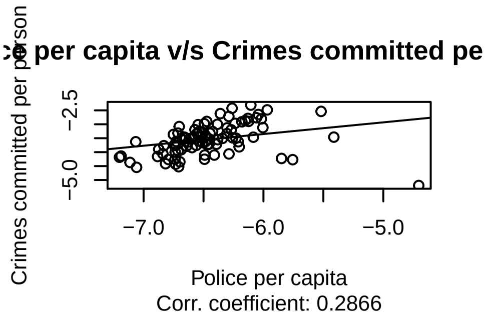
```r
par(mfrow = c(4, 2))

f_plot_three(crime1$prbarr, "Probability of arrest",crime1$logcrmrte,"Crimes committed per person" )

f_plot_three(crime1$pctymle, "% of young males",crime1$logcrmrte,"Crimes committed per person" )

f_plot_three(crime1$prbconv_cast, "Prob of conviction",crime1$logcrmrte,"Crimes committed per person" )

f_plot_three(crime1$logavgsen, "Log of avg sentence",crime1$logcrmrte,"Crimes committed per person" )
```



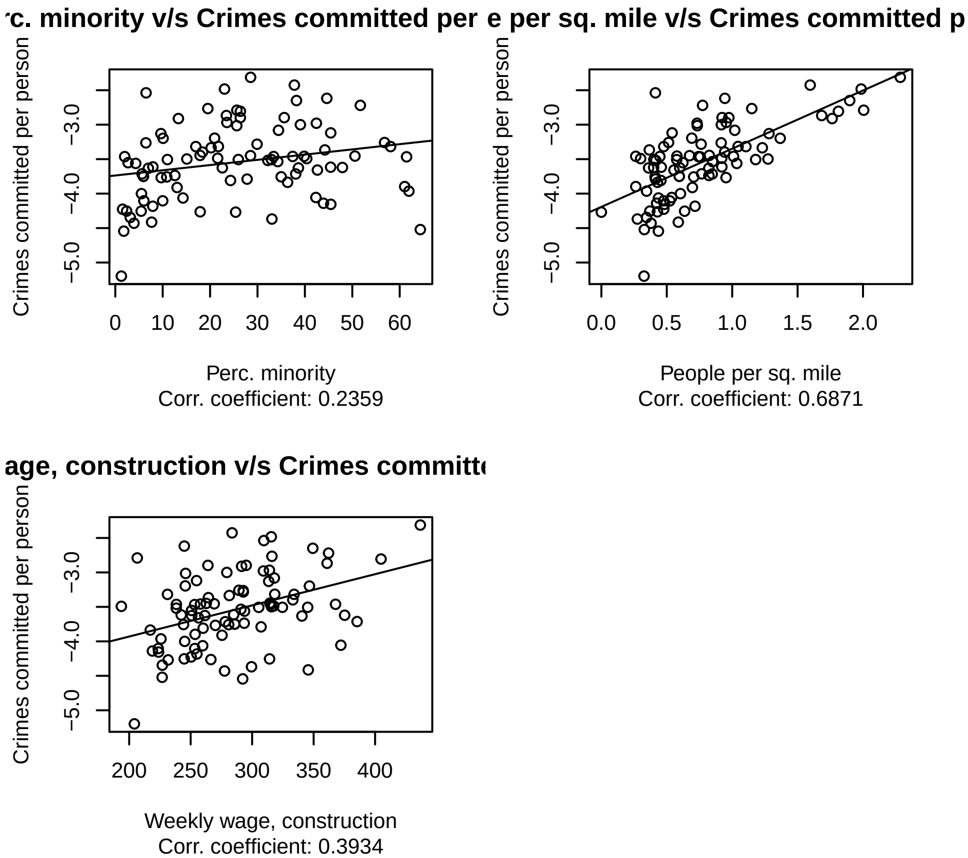

```r
par(mfrow = c(3, 2))

f_plot_three(crime1$prbpris, "Prob of imprisonment",crime1$logcrmrte,"Crimes committed per person" )

f_plot_three(crime1$logavgsen, "Average sentence",crime1$logcrmrte,"Crimes committed per person" )

f_plot_three(crime1$logpolpc, "Police per capita",crime1$logcrmrte,"Crimes committed per person" )
```



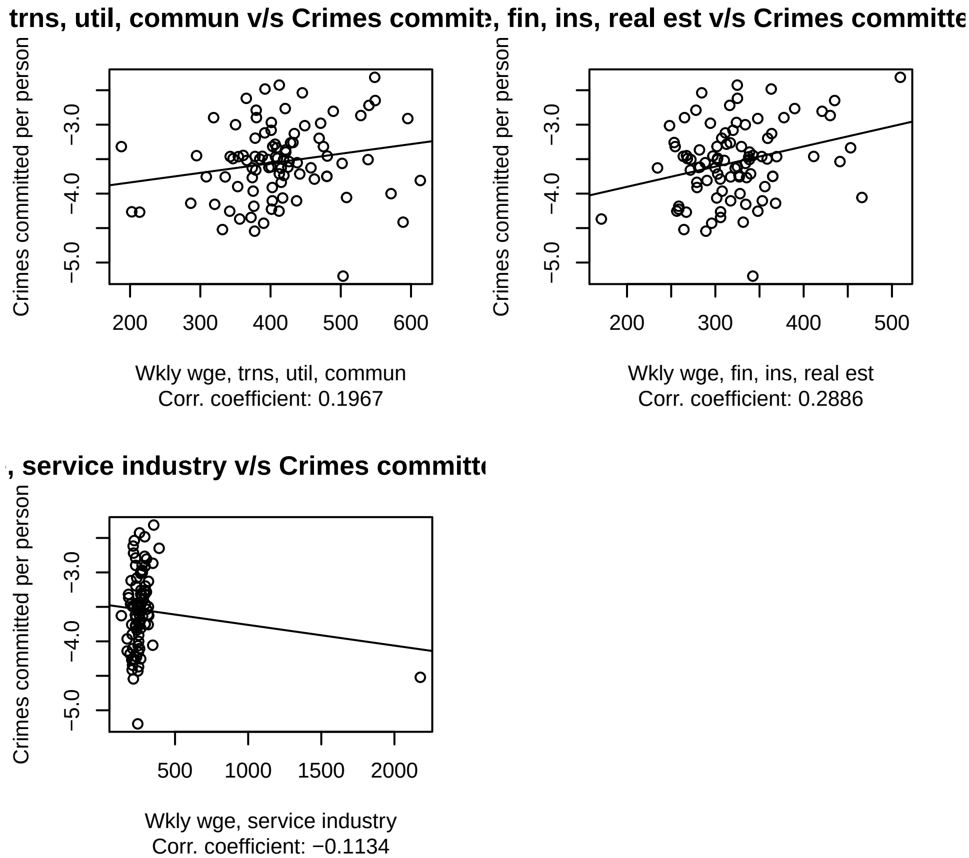




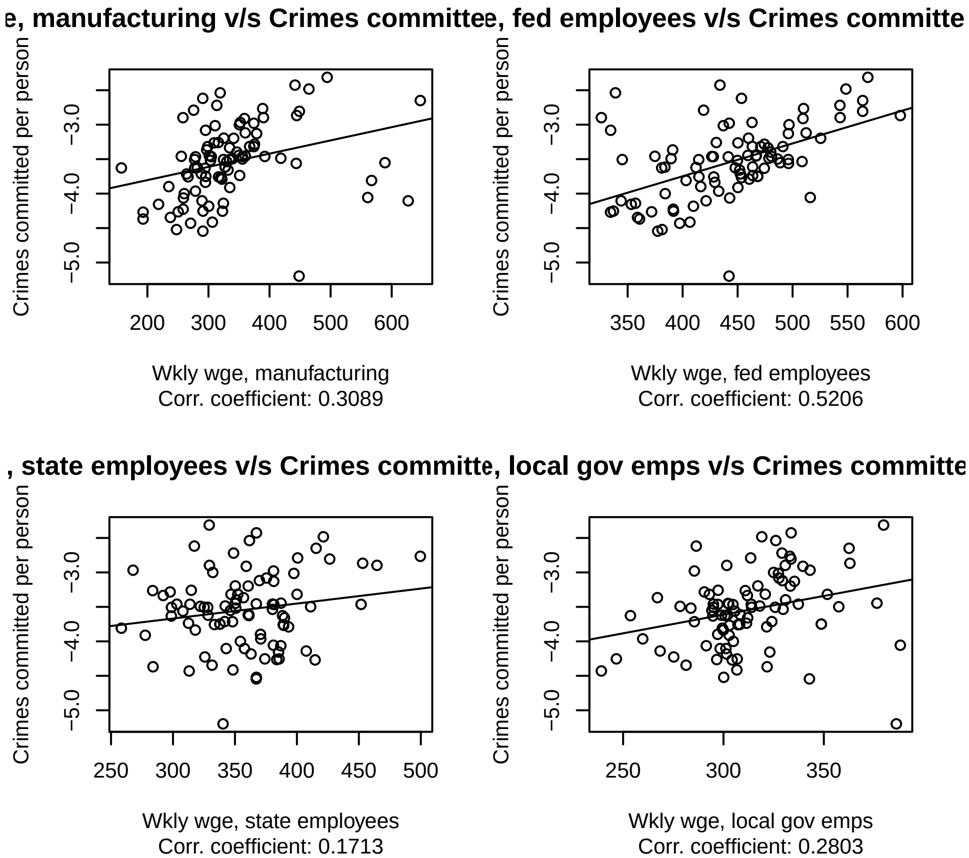

```
par(mfrow = c(3, 2))

f_plot_three(crime1$pctmin80, "Perc. minority",crime1$logcrmrte,"Crimes committed per person" )

f_plot_three(crime1$log1pdensity, "People per sq. mile",crime1$logcrmrte,"Crimes committed per person" )

f_plot_three(crime1$wcon, "Weekly wage, construction",crime1$logcrmrte,"Crimes committed per person" )
```




```r
par(mfrow = c(3, 2))

f_plot_three(crime1$wtuc, "Wkly wge, trns, util, commun",crime1$logcrmrte,"Crimes committed per person"

f_plot_three(crime1$wfir, "Wkly wge, fin, ins, real est",crime1$logcrmrte,"Crimes committed per person"

f_plot_three(crime1$wser, "Wkly wge, service industry",crime1$logcrmrte,"Crimes committed per person" )
```




```r
par(mfrow = c(3, 2))

f_plot_three(crime1$wmfg, "Wkly wge, manufacturing",crime1$logcrmrte,"Crimes committed per person" )

f_plot_three(crime1$wfed, "Wkly wge, fed employees",crime1$logcrmrte,"Crimes committed per person" )

f_plot_three(crime1$wsta, "Wkly wge, state employees",crime1$logcrmrte,"Crimes committed per person" )

f_plot_three(crime1$wloc, "Wkly wge, local gov emps",crime1$logcrmrte,"Crimes committed per person" )
```



Strong positive correlation:

crmrte v/s polpc crmrte v/s density crmrte v/s wcon crmrte v/s wser crmrte v/s wmfg crmrte v/s wfed crmrte v/s wloc

Weak positive correlation:

crmrte v/s pctymle crmrte v/s percent of minority crmrte v/s wkly wge, fin, ins, real est crmrte v/s wtuc

Strong negative correlation:

crmrte v/s prbarr

Weak negative correlation:

crmrte v/s prbconv

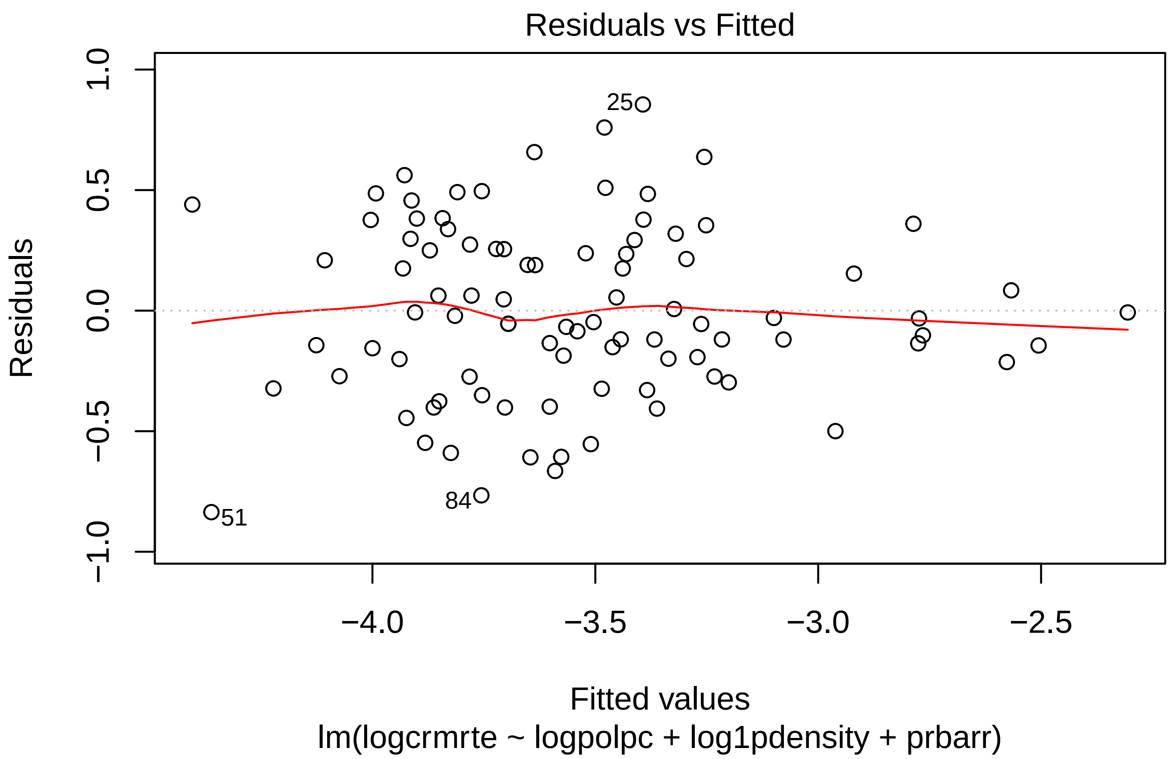
Model 1: Crime related variables

log(crime rate) = log(beta0 + beta1*polpc* + *beta2*density)

```
logcrmrte.lm1 = lm( logcrmrte ~ logpolpc + log1pdensity + prbarr , data=crime1)
logcrmrte.lm1
```

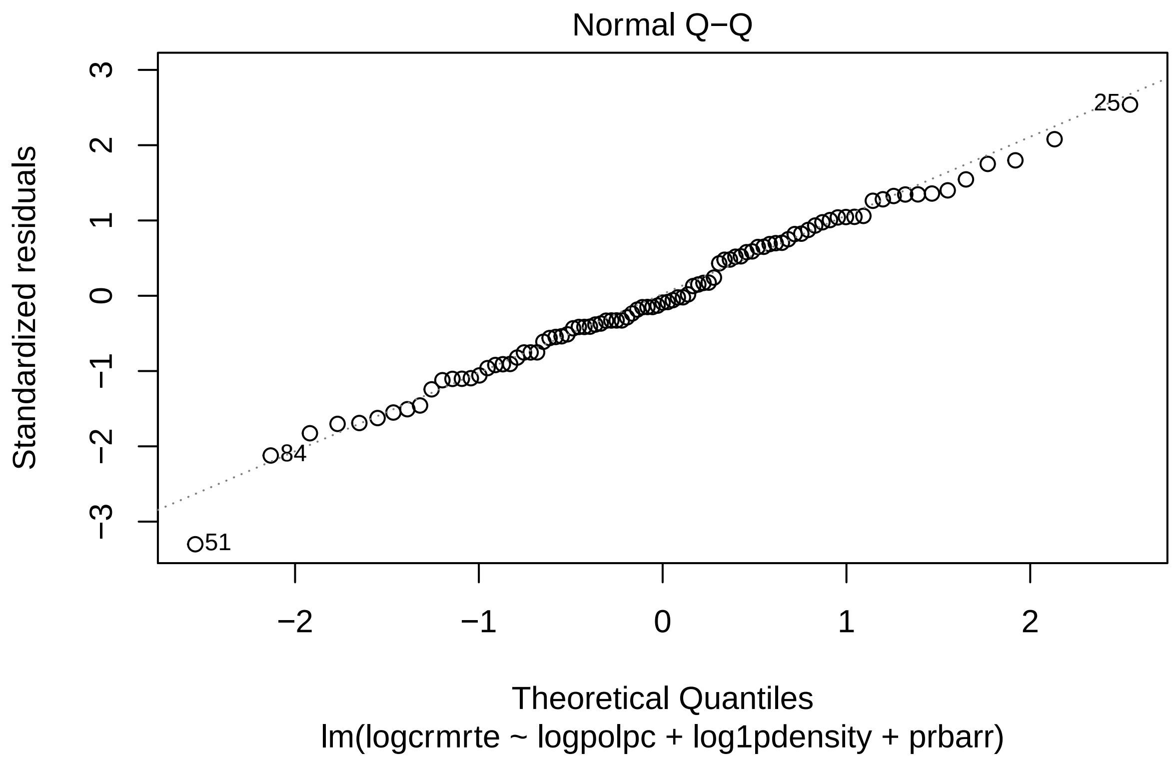
```
##
## Call:
## lm(formula = logcrmrte ~ logpolpc + log1pdensity + prbarr, data = crime1)
##
## Coefficients:
##  (Intercept)      logpolpc  log1pdensity        prbarr
##      -1.7284        0.2938        0.6193       -1.3316
```

```r
plot(logcrmrte.lm1, which =1)
```



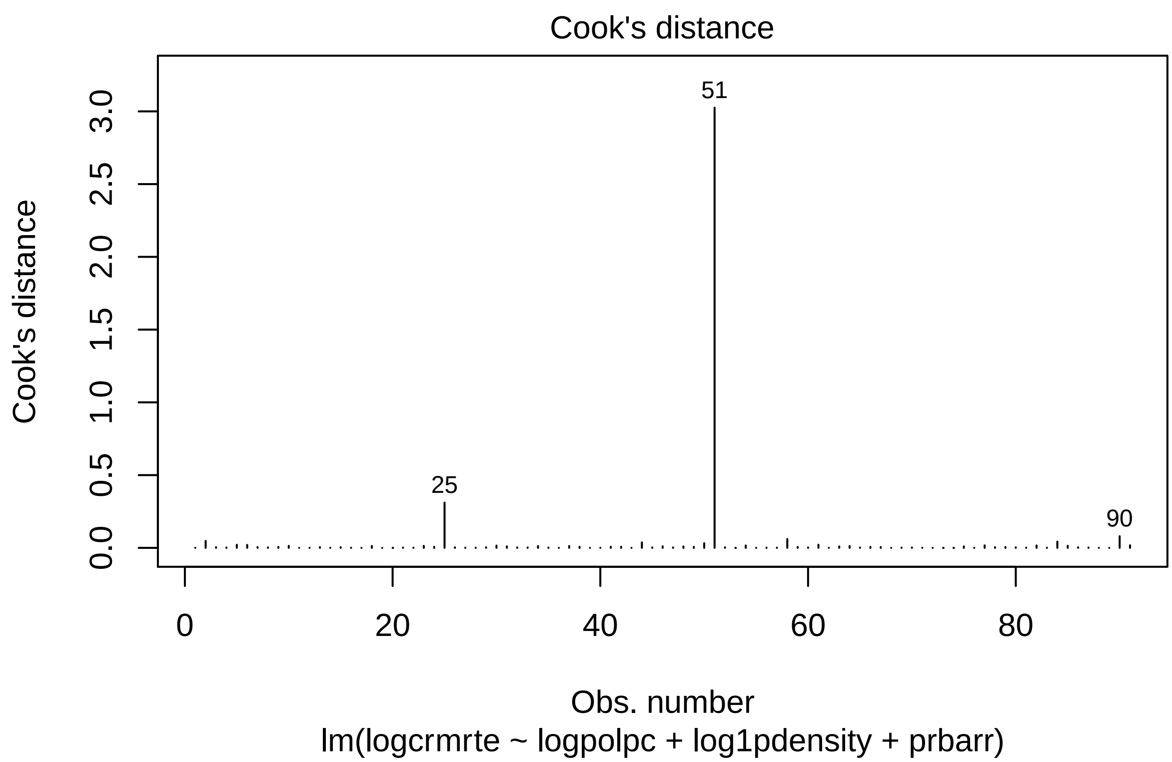
The regression lines is close to 0 residual. This indicates there is a linear relationship among logcrmrte with logpolpc + log1pdensity.

```r
plot(logcrmrte.lm1, which =2)
```



The Normal QQ plot is roughly on a straight line. This indicates that our data has been sourced from a normal distribution

```
plot(logcrmrte.lm1, which =4)
```



None of the points have Cook's distance greater than 1. However, Points 5, 23 and 72 have higher leverage than all other points. Let us observe the points:

```r
crime1[5,]
```

```
##   county year    crmrte   prbarr     prbconv  prbpris avgsen      polpc
## 5      9   87 0.0106232 0.518219 0.476563007 0.442623   8.22 0.00086018
##     density    taxpc west central urban pctmin80     wcon     wtuc
## 5 0.5469484 28.05474    1       0     0  1.79619 292.3077 377.3126
##       wtrd     wfir     wser    wmfg    wfed    wsta    wloc        mix
## 5 206.8215 289.3125 215.1933 290.89 377.35 367.23 342.82 0.06008584
##      pctymle prbconv_cast urbanorrural logcrmrte logavgsen  logpolpc
## 5 0.07069755     0.476563        RURAL -4.544715   2.10657 -7.058369
##   log1pdensity
## 5    0.4362842
```

```r
crime1[23,]
```

```
##    county year    crmrte   prbarr     prbconv  prbpris avgsen      polpc
## 23     51   87 0.0883849 0.155248 0.259833008 0.407628  11.93 0.00190802
##     density    taxpc west central urban pctmin80     wcon    wtuc     wtrd
## 23 3.934551 35.69936    0       0     1  37.7792 283.6695 412.472 213.7524
##        wfir     wser    wmfg   wfed    wsta    wloc       mix   pctymle
## 23 324.8357 257.3344 441.72 433.94 367.34 333.71 0.1047432 0.1422378
##    prbconv_cast urbanorrural logcrmrte logavgsen  logpolpc log1pdensity
## 23     0.259833        URBAN -2.426054  2.479056 -6.261689     1.596262
```

```r
crime1[72,]
```

```
##    county year   crmrte  prbarr     prbconv  prbpris avgsen      polpc
## 72    159   87 0.036233 0.24359 0.492940009 0.476563   8.64 0.00158619
##     density    taxpc west central urban pctmin80     wcon     wtuc    wtrd
## 72 2.019268 27.76489    0       1     0  16.9913 334.1035 475.3228 260.271
##        wfir     wser    wmfg   wfed    wsta    wloc        mix    pctymle
## 72 329.5464 265.4315 374.41 491.16 346.81 351.74 0.09146758 0.07705218
##    prbconv_cast urbanorrural logcrmrte logavgsen logpolpc log1pdensity
## 72      0.49294        RURAL -3.317785  2.156403 -6.44642     1.105014
```

Checking for multicollinearity

```r
## Reference: https://www.r-bloggers.com/collinearity-and-stepwise-vif-selection/
## https://datascienceplus.com/multicollinearity-in-r/
vif(logcrmrte.lm1)
```

```
##     logpolpc log1pdensity       prbarr
##     1.288271     1.409444     1.333396
```

Since vif output for both th variable sis less than 10, there does not exist a multicollinearity between the variables.

Model 2: Wage variable wcon

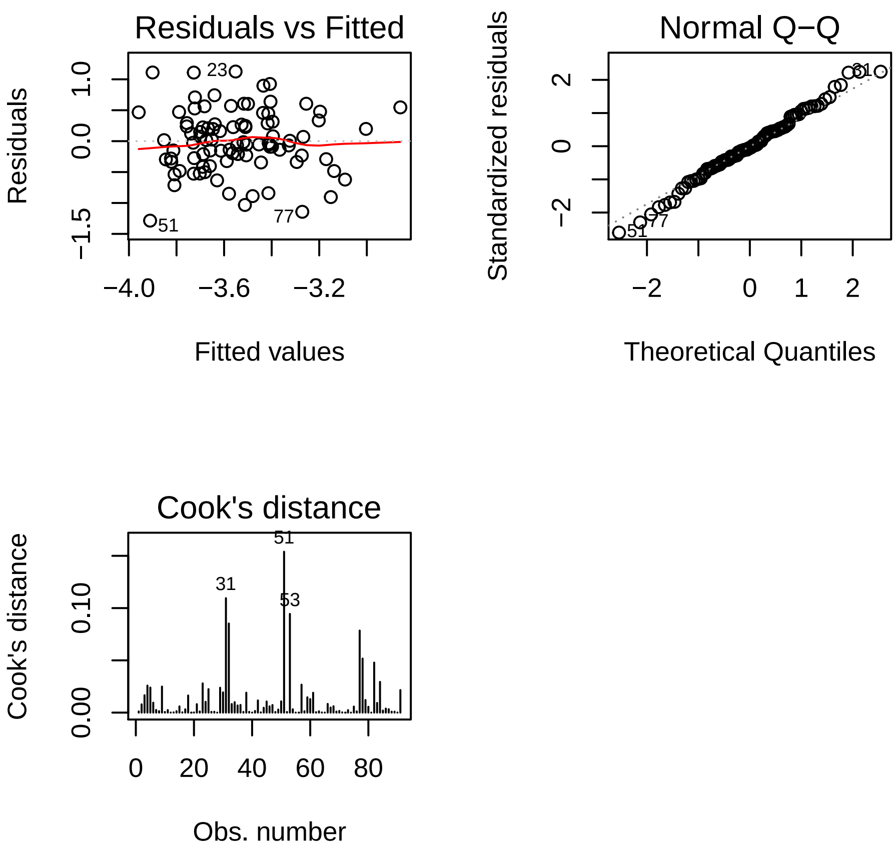
log(crime rate) = log(beta0 + beta1*wcon + error)

```r
logcrmrte.lm2 = lm( logcrmrte ~ wcon, data=crime1)
logcrmrte.lm2
```

```
##
## Call:
```

```
## lm(formula = logcrmrte ~ wcon, data = crime1)
##
## Coefficients:
## (Intercept)         wcon
##   -4.834926     0.004524
```

```r
par(mfrow =c(2,2))
plot(logcrmrte.lm2,which=c(1,2,4))
par(mfrow =c(1,1))
```



The regression lines is close to 0 residual. This indicates there is a linear relationship among logcrmrte with wcon. The Normal QQ plot is roughly on a straight line. This indicates that our data has been sourced from a normal distribution None of the points have Cook's distance greater than 1.

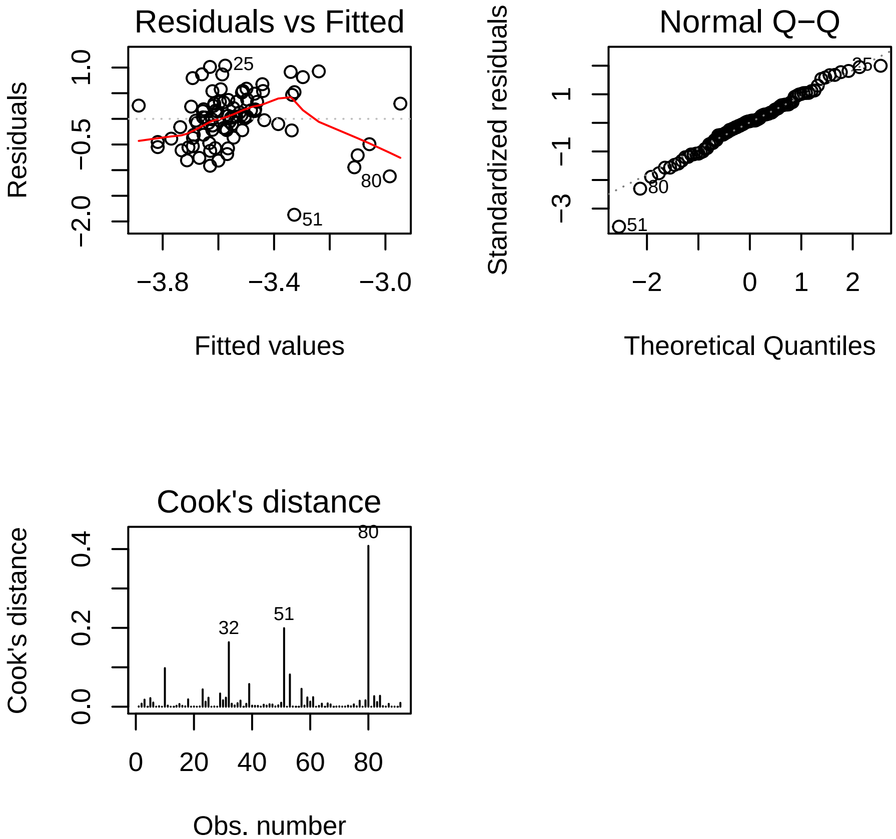
Model 2: Wage variable wmfg

log(crime rate) = log(beta0 + beta1*wmfg + error)

```r
logcrmrte.lm3 = lm( logcrmrte ~ wmfg, data=crime1)
logcrmrte.lm3
```

```
##
## Call:
## lm(formula = logcrmrte ~ wmfg, data = crime1)
##
## Coefficients:
## (Intercept)         wmfg
##    -4.18842      0.00192
```

```
par(mfrow =c(2,2))
plot(logcrmrte.lm3,which=c(1,2,4))
par(mfrow =c(1,1))
```



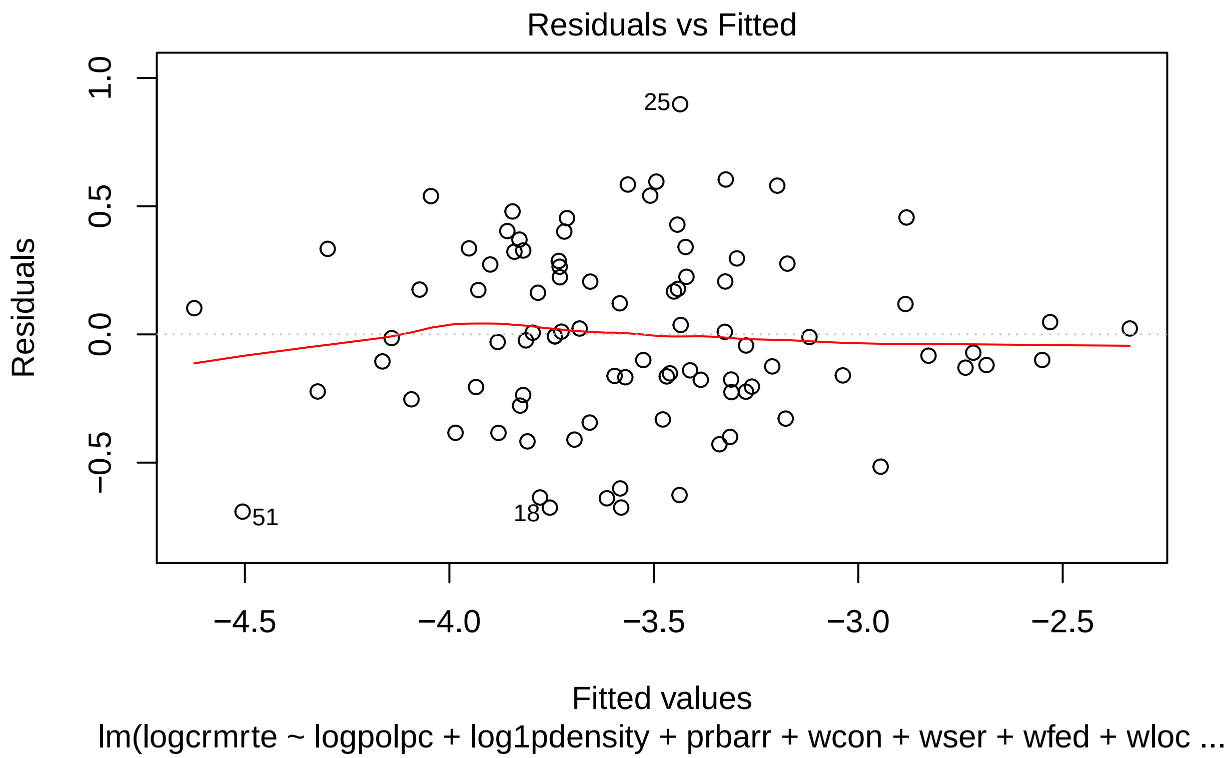
The regression lines is varying from residual 0. This indicates there is no linear relationship between logcrmrte and wmfg. The Normal QQ plot is roughly on a straight line. This indicates that our data has been sourced from a normal distribution None of the points have Cook's distance greater than 1.

Model 3: All variables with strong positive or negative correlation

```
logcrmrte.lm4 = lm( logcrmrte ~ logpolpc + log1pdensity +prbarr + wcon + wser  + wfed + wloc, data=crim
logcrmrte.lm4
```

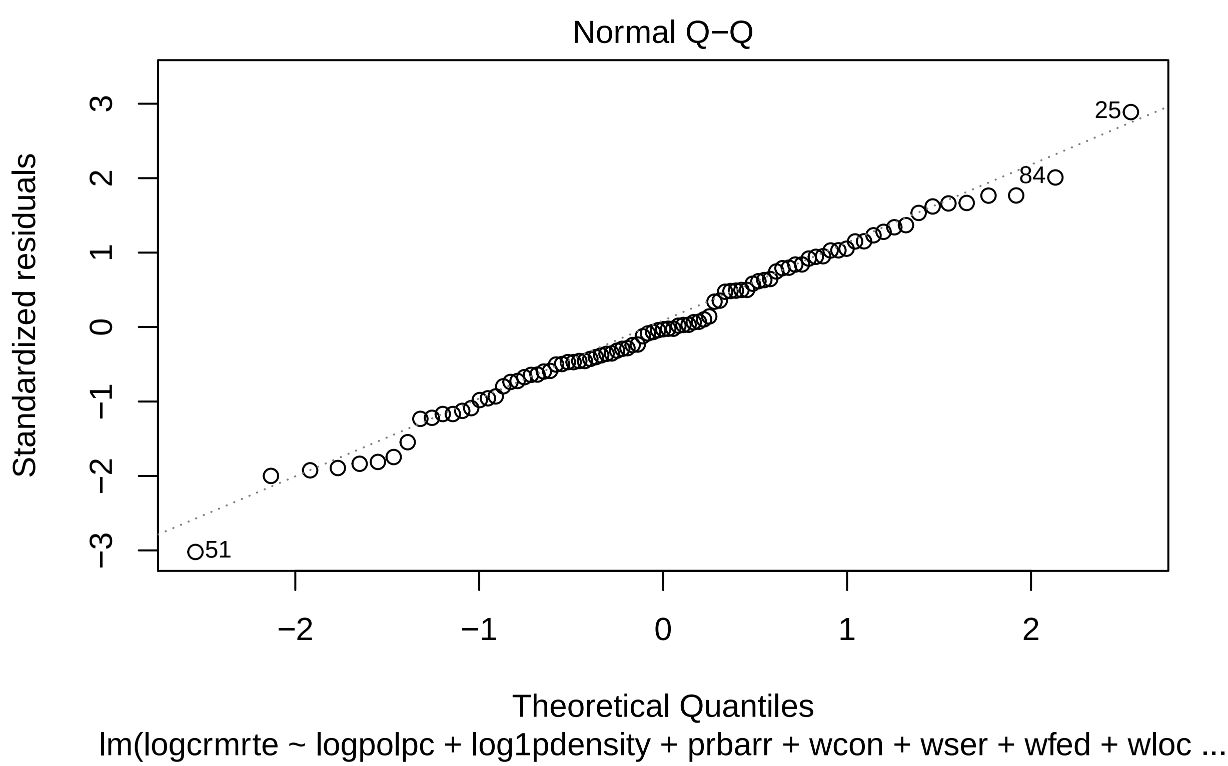
```
##
## Call:
## lm(formula = logcrmrte ~ logpolpc + log1pdensity + prbarr + wcon +
##     wser + wfed + wloc, data = crime1)
##
## Coefficients:
##  (Intercept)      logpolpc  log1pdensity        prbarr          wcon
##   -1.4051226     0.3205103     0.5438489    -1.3763408     0.0006770
##         wser          wfed          wloc
##   -0.0004309     0.0010619    -0.0019969
```

```
plot(logcrmrte.lm4, which =1)
```



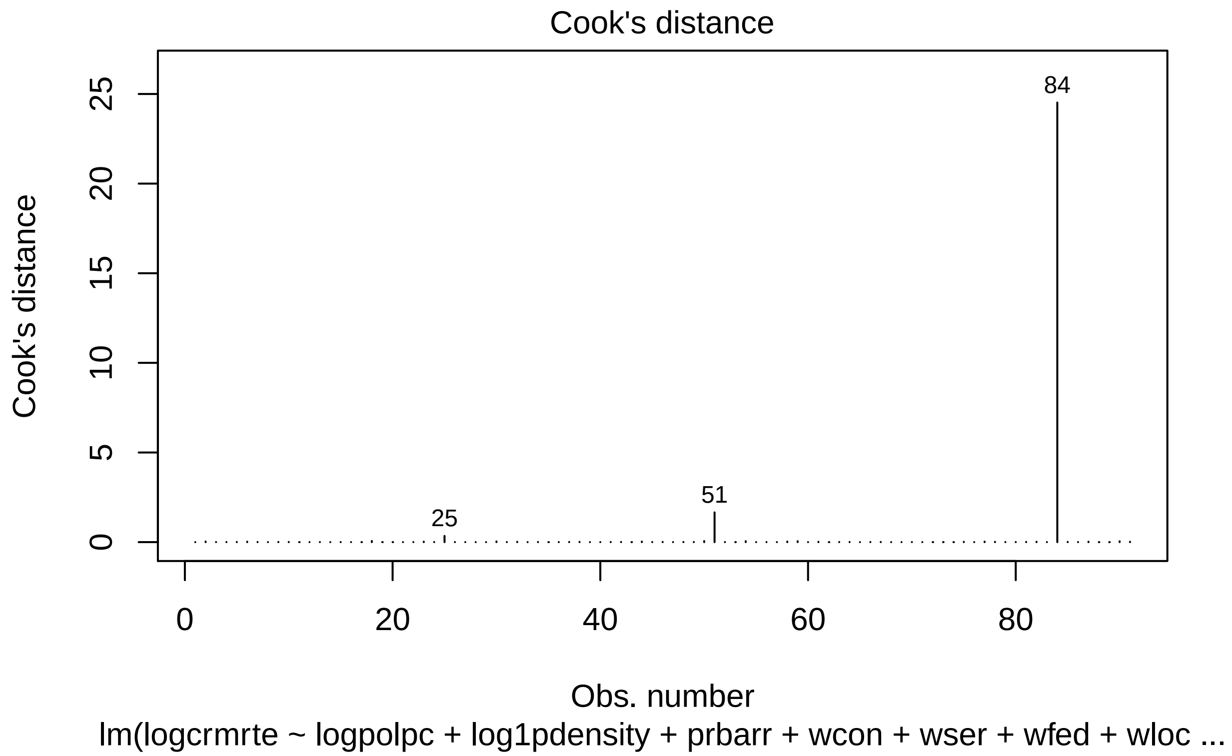
The regression lines is close to 0 residual. This indicates there is a linear relationship between logcrmrte and (logpolpc + log1pdensity +prbarr + wcon + wser + wfed + wloc)

```
plot(logcrmrte.lm4, which =2)
```



The Normal QQ plot is roughly on a straight line. This indicates that our data has been sourced from a normal distribution

```r
plot(logcrmrte.lm4, which =4)
```



None of the points have Cook's distance greater than 1. However, Points 5, 23 and 72 have higher leverage than all other points.

```r
vif(logcrmrte.lm4)
```

```
##     logpolpc log1pdensity       prbarr         wcon         wser
##     1.393731     2.250659     1.396847     1.665406     1.036813
##         wfed         wloc
##     2.050598     1.781272
```

There is no multicollinearity since none of the vif values are more than 10.

Comparing the models

```r
summary(logcrmrte.lm1)$r.squared
```

```
## [1] 0.5612806
```

```r
summary(logcrmrte.lm2)$r.squared
```

```
## [1] 0.1547315
```

```r
summary(logcrmrte.lm3)$r.squared
```

```
## [1] 0.0953937
```

```r
summary(logcrmrte.lm4)$r.squared
```

```
## [1] 0.6027055
```

Based on the r-squared output logcrmrte.lm1 or logcrmrte.lm3 is preferred.

```
## The Akaike information criterion (AIC) is an estimator of the relative quality of statistical models
AIC(logcrmrte.lm1, logcrmrte.lm2, logcrmrte.lm3, logcrmrte.lm4)
```

```
##               df       AIC
## logcrmrte.lm1  5  82.17264
## logcrmrte.lm2  3 137.84992
## logcrmrte.lm3  3 144.02387
## logcrmrte.lm4  9  81.14706
```

Based on the AIC output logcrmrte.lm1 or logcrmrte.lm3 is preferred.

```
stargazer(logcrmrte.lm1, logcrmrte.lm2,logcrmrte.lm3, type = "latex",
          report = "vc", # Don't report errors, since we haven't covered them
          title = "Linear Models Predicting Crime rate per persone",
          keep.stat = c("rsq", "n"),
          omit.table.layout = "n") # Omit more output related to errors
```

% Table created by stargazer v.5.2 by Marek Hlavac, Harvard University. E-mail: hlavac at fas.harvard.edu
% Date and time: Fri, Mar 30, 2018 - 20:15:39

Table 1: Linear Models Predicting Crime rate per persone

	Dependent variable:		
	logcrmrte		
	(1)	(2)	(3)
logpolpc	0.294		
log1pdensity	0.619		
prbarr	−1.332		
wcon		0.005	
wmfg			0.002
Constant	−1.728	−4.835	−4.188
Observations	91	91	91
R^2	0.561	0.155	0.095

```
crime_west=sqldf("SELECT * FROM crime1 WHERE west=1")

sqldf("SELECT COUNT(*) AS count_crime_west FROM crime_west")
```

```
##   count_crime_west
## 1               23
```

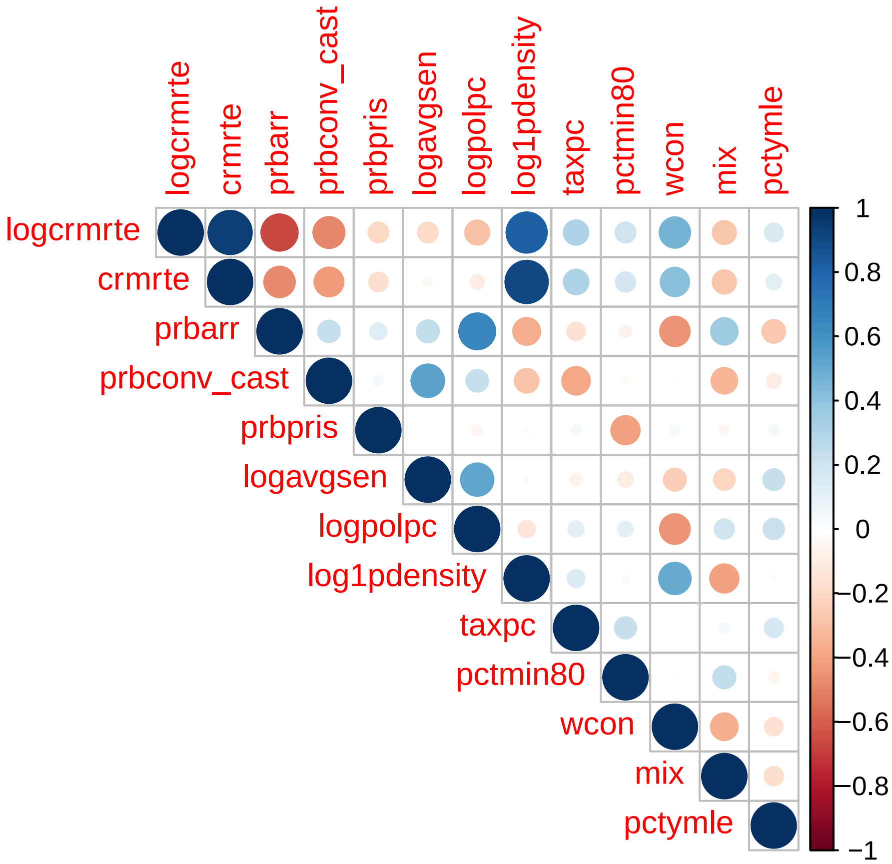
```
crime_central=sqldf("SELECT * FROM crime1 WHERE central=1")

sqldf("SELECT COUNT(*) AS count_crime_central FROM crime_central")
```

```
##   count_crime_central
## 1                  34
```

```r
library(corrplot)
```

```
## corrplot 0.84 loaded
```

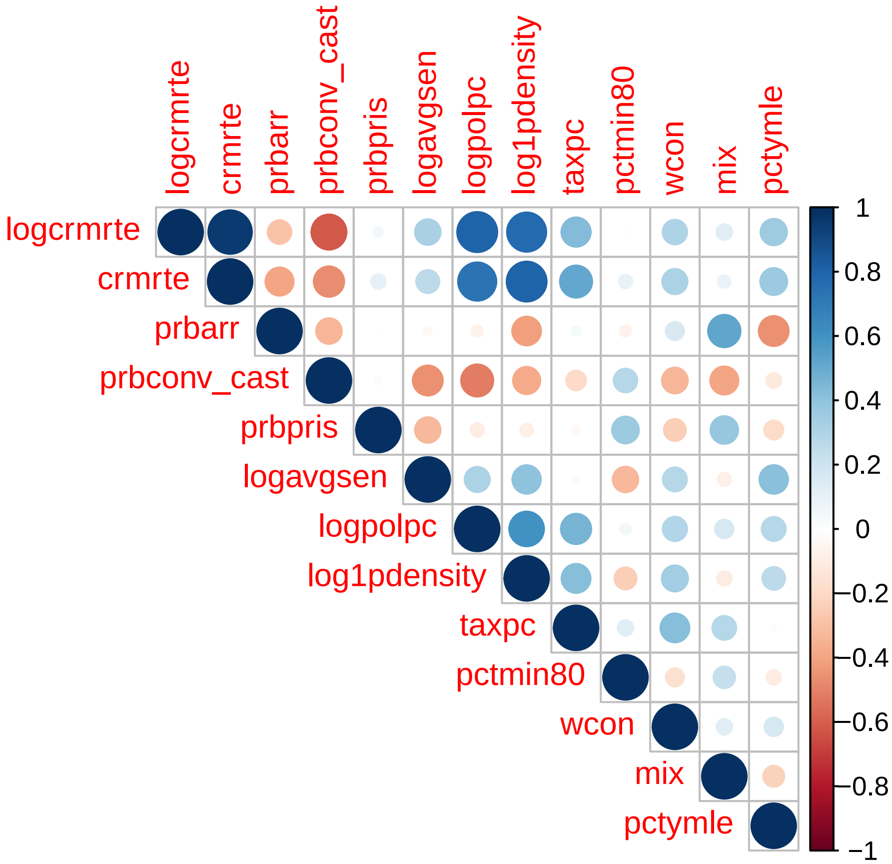
```r
corrplot(cor(crime_west[,
                        c("logcrmrte","crmrte", "prbarr", "prbconv_cast", "prbpris", "logavgsen",
                          "logpolpc", "log1pdensity", "taxpc", "pctmin80", "wcon",
                          "mix", "pctymle")]), type = "upper")
```




```r
logcrmrte.west.lm5 = lm( logcrmrte ~ logpolpc + log1pdensity , data=crime_west)
logcrmrte.west.lm5
```

```
##
## Call:
## lm(formula = logcrmrte ~ logpolpc + log1pdensity, data = crime_west)
##
## Coefficients:
##  (Intercept)      logpolpc  log1pdensity
##      -5.9189       -0.1974        1.1140
```

```r
corrplot(cor(crime_central[,
                        c("logcrmrte", "crmrte","prbarr", "prbconv_cast", "prbpris", "logavgsen",
                          "logpolpc", "log1pdensity", "taxpc", "pctmin80", "wcon",
                          "mix", "pctymle")]), type = "upper")
```




```
logcrmrte.central.lm6 = lm( logcrmrte ~ logpolpc + log1pdensity + prbconv_cast , data=crime_central)
logcrmrte.central.lm6
```

```
##
## Call:
## lm(formula = logcrmrte ~ logpolpc + log1pdensity + prbconv_cast,
##     data = crime_central)
##
## Coefficients:
##  (Intercept)      logpolpc  log1pdensity  prbconv_cast
##       0.8912        0.7057        0.4325       -0.3425
```

```
summary(logcrmrte.lm1)$r.squared
```

```
## [1] 0.5612806
```

```
summary(logcrmrte.lm2)$r.squared
```

```
## [1] 0.1547315
```

```
summary(logcrmrte.lm3)$r.squared
```

```
## [1] 0.0953937
```

```
summary(logcrmrte.lm4)$r.squared
```

```
## [1] 0.6027055
```

```
summary(logcrmrte.west.lm5)$r.squared
```

```
## [1] 0.703168
```

```r
summary(logcrmrte.central.lm6)$r.squared
```

```
## [1] 0.8194858
```