MSA 6701 Project

Nick Wawee

11/12/2020

Loading and Inspecting

```
data(Boston)
df = Boston
for (col in colnames(df)){
  cat("Number of Missing values in ", col,": ", as.character(length(which(is.na(df[,col])))), "\n")
}
```

```
## Number of Missing values in  crim :  0
## Number of Missing values in  zn :  0
## Number of Missing values in  indus :  0
## Number of Missing values in  chas :  0
## Number of Missing values in  nox :  0
## Number of Missing values in  rm :  0
## Number of Missing values in  age :  0
## Number of Missing values in  dis :  0
## Number of Missing values in  rad :  0
## Number of Missing values in  tax :  0
## Number of Missing values in  ptratio :  0
## Number of Missing values in  black :  0
## Number of Missing values in  lstat :  0
## Number of Missing values in  medv :  0
```

```
str(df)
```

```
## ’data.frame':    506 obs. of  14 variables:
##  $ crim   : num  0.00632 0.02731 0.02729 0.03237 0.06905 ...
##  $ zn     : num  18 0 0 0 0 0 12.5 12.5 12.5 12.5 ...
##  $ indus  : num  2.31 7.07 7.07 2.18 2.18 2.18 7.87 7.87 7.87 7.87 ...
##  $ chas   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ nox    : num  0.538 0.469 0.469 0.458 0.458 0.458 0.524 0.524 0.524 0.524 ...
##  $ rm     : num  6.58 6.42 7.18 7 7.15 ...
##  $ age    : num  65.2 78.9 61.1 45.8 54.2 58.7 66.6 96.1 100 85.9 ...
##  $ dis    : num  4.09 4.97 4.97 6.06 6.06 ...
##  $ rad    : int  1 2 2 3 3 3 5 5 5 5 ...
##  $ tax    : num  296 242 242 222 222 222 311 311 311 311 ...
##  $ ptratio: num  15.3 17.8 17.8 18.7 18.7 18.7 15.2 15.2 15.2 15.2 ...
##  $ black  : num  397 397 393 395 397 ...
##  $ lstat  : num  4.98 9.14 4.03 2.94 5.33 ...
##  $ medv   : num  24 21.6 34.7 33.4 36.2 28.7 22.9 27.1 16.5 18.9 ...
```

```
df = as.data.frame(df)
head(df)
```

```
##       crim zn indus chas   nox    rm  age    dis rad tax ptratio  black lstat
## 1 0.00632 18  2.31    0 0.538 6.575 65.2 4.0900   1 296    15.3 396.90  4.98
## 2 0.02731  0  7.07    0 0.469 6.421 78.9 4.9671   2 242    17.8 396.90  9.14
## 3 0.02729  0  7.07    0 0.469 7.185 61.1 4.9671   2 242    17.8 392.83  4.03
## 4 0.03237  0  2.18    0 0.458 6.998 45.8 6.0622   3 222    18.7 394.63  2.94
## 5 0.06905  0  2.18    0 0.458 7.147 54.2 6.0622   3 222    18.7 396.90  5.33
## 6 0.02985  0  2.18    0 0.458 6.430 58.7 6.0622   3 222    18.7 394.12  5.21
##   medv
## 1 24.0
## 2 21.6
## 3 34.7
## 4 33.4
## 5 36.2
## 6 28.7
```

Model Fitting

```
mlr = lm(crim~., data =df)
summary(mlr)
```

```
##
## Call:
## lm(formula = crim ~ ., data = df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.924 -2.120 -0.353  1.019 75.051
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  17.033228   7.234903   2.354 0.018949 *
## zn            0.044855   0.018734   2.394 0.017025 *
## indus        -0.063855   0.083407  -0.766 0.444294
## chas         -0.749134   1.180147  -0.635 0.525867
## nox         -10.313535   5.275536  -1.955 0.051152 .
## rm            0.430131   0.612830   0.702 0.483089
## age           0.001452   0.017925   0.081 0.935488
## dis          -0.987176   0.281817  -3.503 0.000502 ***
## rad           0.588209   0.088049   6.680 6.46e-11 ***
## tax          -0.003780   0.005156  -0.733 0.463793
## ptratio      -0.271081   0.186450  -1.454 0.146611
## black        -0.007538   0.003673  -2.052 0.040702 *
## lstat         0.126211   0.075725   1.667 0.096208 .
## medv         -0.198887   0.060516  -3.287 0.001087 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 6.439 on 492 degrees of freedom
## Multiple R-squared:  0.454,  Adjusted R-squared:  0.4396
## F-statistic: 31.47 on 13 and 492 DF,  p-value: < 2.2e-16
```

Stepwise Regression

```
sbi = stepwise(df, y = ’crim’, selection = "backward", select = ’AIC’)
sbi$variate
```

```
## [1] "intercept" "zn"        "nox"       "dis"       "rad"       "ptratio"
## [7] "black"     "lstat"     "medv"
```

```
mlr2 = lm(crim~ zn + nox + dis + rad + ptratio + black + lstat + medv, data =df )
summary(mlr2)
```

```
##
## Call:
## lm(formula = crim ~ zn + nox + dis + rad + ptratio + black +
##     lstat + medv, data = df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.860 -2.102 -0.363  0.895 75.702
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  19.683128   6.086010   3.234 0.001301 **
## zn            0.043293   0.017977   2.408 0.016394 *
## nox         -12.753708   4.760157  -2.679 0.007623 **
## dis          -0.918318   0.261932  -3.506 0.000496 ***
## rad           0.532617   0.049727  10.711  < 2e-16 ***
## ptratio      -0.310541   0.182941  -1.697 0.090229 .
## black        -0.007922   0.003615  -2.191 0.028897 *
## lstat         0.110173   0.069219   1.592 0.112097
## medv         -0.174207   0.053988  -3.227 0.001334 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 6.428 on 497 degrees of freedom
## Multiple R-squared:  0.4505, Adjusted R-squared:  0.4416
## F-statistic: 50.92 on 8 and 497 DF,  p-value: < 2.2e-16
```

Colinearity Check

```
vif(mlr2)
```

```
##       zn      nox      dis      rad  ptratio    black    lstat     medv
## 2.148871 3.719176 3.718604 2.291669 1.917428 1.331764 2.986626 3.013693
```

Correlation between regressors:

```
cor(df[,which(colnames(df)%in%sbi$variate)])
```

```
##                 zn        nox        dis        rad    ptratio      black
## zn       1.0000000 -0.5166037  0.6644082 -0.3119478 -0.3916785  0.1755203
## nox     -0.5166037  1.0000000 -0.7692301  0.6114406  0.1889327 -0.3800506
## dis      0.6644082 -0.7692301  1.0000000 -0.4945879 -0.2324705  0.2915117
## rad     -0.3119478  0.6114406 -0.4945879  1.0000000  0.4647412 -0.4444128
## ptratio -0.3916785  0.1889327 -0.2324705  0.4647412  1.0000000 -0.1773833
## black    0.1755203 -0.3800506  0.2915117 -0.4444128 -0.1773833  1.0000000
## lstat   -0.4129946  0.5908789 -0.4969958  0.4886763  0.3740443 -0.3660869
## medv     0.3604453 -0.4273208  0.2499287 -0.3816262 -0.5077867  0.3334608
##             lstat       medv
## zn      -0.4129946  0.3604453
## nox      0.5908789 -0.4273208
## dis     -0.4969958  0.2499287
## rad      0.4886763 -0.3816262
## ptratio  0.3740443 -0.5077867
## black   -0.3660869  0.3334608
## lstat    1.0000000 -0.7376627
## medv    -0.7376627  1.0000000
```

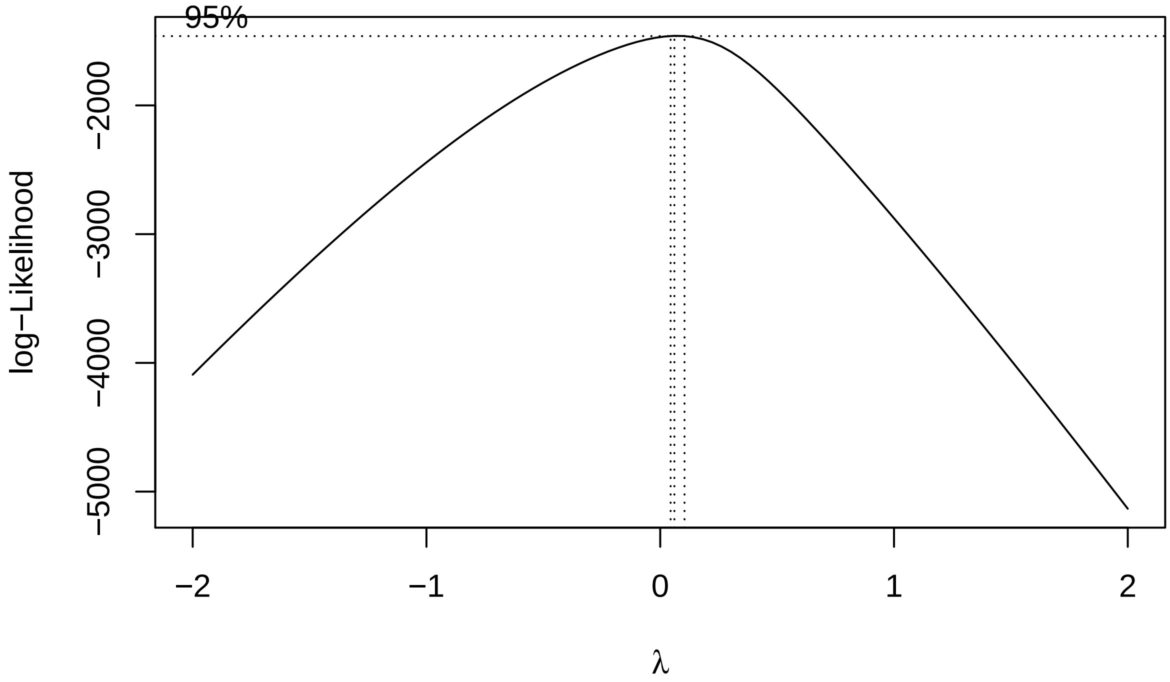
We see that the intercept is not statistically significant and it doesnt make any sense to have a negative crime rate, so we remove it.

Power Transformation

```
mlr4 = lm(crim~ 0 +  rad*dis + ptratio + black+medv+rad*nox , data = df)
summary(mlr4)
```

```
## 
## Call:
## lm(formula = crim ~ 0 + rad * dis + ptratio + black + medv + 
##     rad * nox, data = df)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -11.810  -1.238  -0.291   0.707  71.213 
## 
## Coefficients:
##          Estimate Std. Error t value Pr(>|t|)    
## rad      2.352948   0.323698   7.269 1.41e-12 ***
## dis      1.414262   0.227568   6.215 1.09e-09 ***
## ptratio -0.176981   0.117935  -1.501 0.134077    
## black   -0.005616   0.003354  -1.674 0.094688 .  
## medv    -0.161579   0.033120  -4.879 1.44e-06 ***
## nox      6.526916   3.313172   1.970 0.049393 *  
## rad:dis -0.349350   0.036138  -9.667  < 2e-16 ***
## rad:nox -1.597729   0.425654  -3.754 0.000195 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 6.047 on 498 degrees of freedom
## Multiple R-squared:  0.5859, Adjusted R-squared:  0.5792 
## F-statistic: 88.06 on 8 and 498 DF,  p-value: < 2.2e-16
```

```
bc = boxcox(mlr4, data = df)
```



```
lambda = bc$x[which.max(bc$y)]
lambda
```

```
## [1] 0.06060606
```

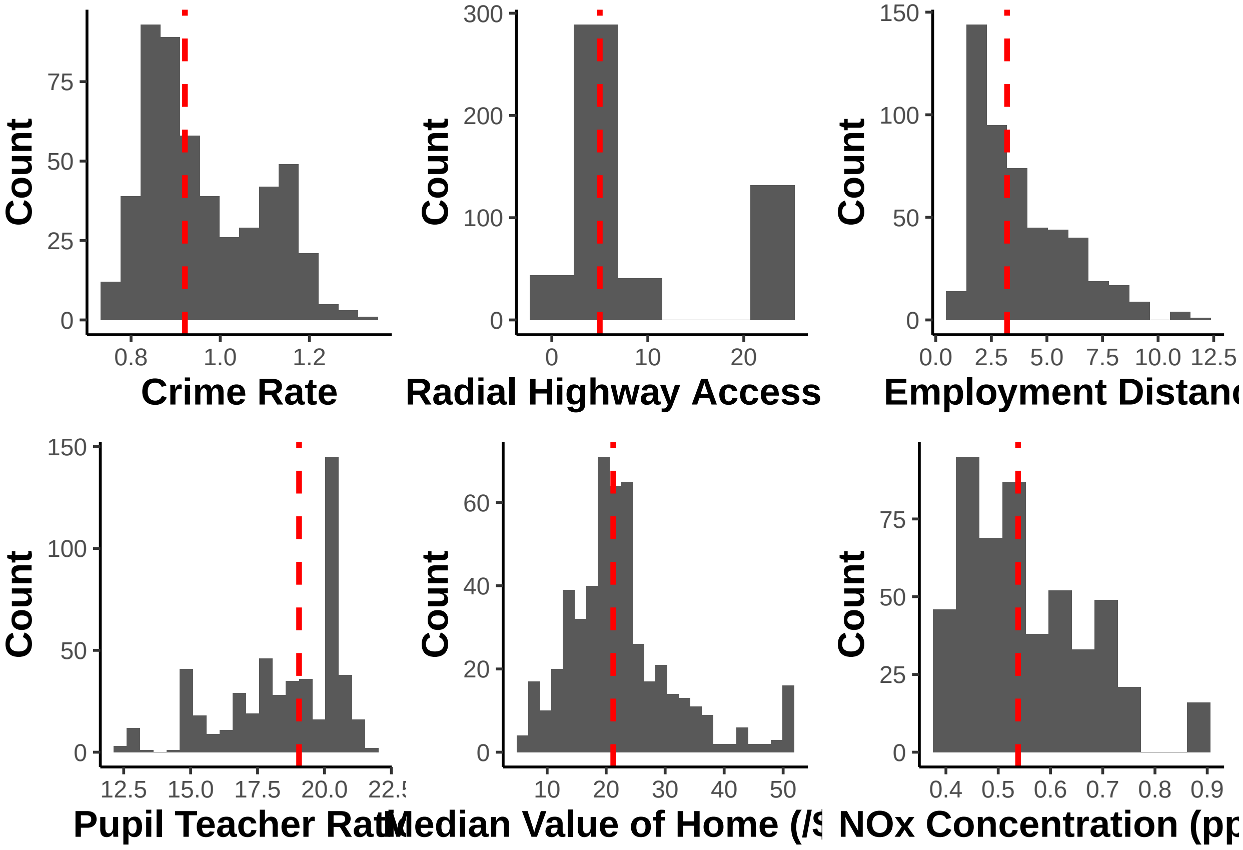
```
dfnew = df
dfnew$crim = df$crim^lambda
```

```
mlr5 = lm(crim~   0 +rad*dis + ptratio + medv + nox, data = dfnew)
summary(mlr5)
```

```
##
## Call:
## lm(formula = crim ~ 0 + rad * dis + ptratio + medv + nox, data = dfnew)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.158766 -0.037238  0.002491  0.043698  0.177525
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## rad      0.0098229  0.0008806  11.154  < 2e-16 ***
## dis      0.0169361  0.0022508   7.525 2.48e-13 ***
## ptratio  0.0183480  0.0009885  18.561  < 2e-16 ***
## medv     0.0024981  0.0002924   8.543  < 2e-16 ***
## nox      0.8045297  0.0277055  29.039  < 2e-16 ***
## rad:dis -0.0014372  0.0003429  -4.191 3.28e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06163 on 500 degrees of freedom
## Multiple R-squared:  0.996,  Adjusted R-squared:  0.996
## F-statistic: 2.084e+04 on 6 and 500 DF,  p-value: < 2.2e-16
```

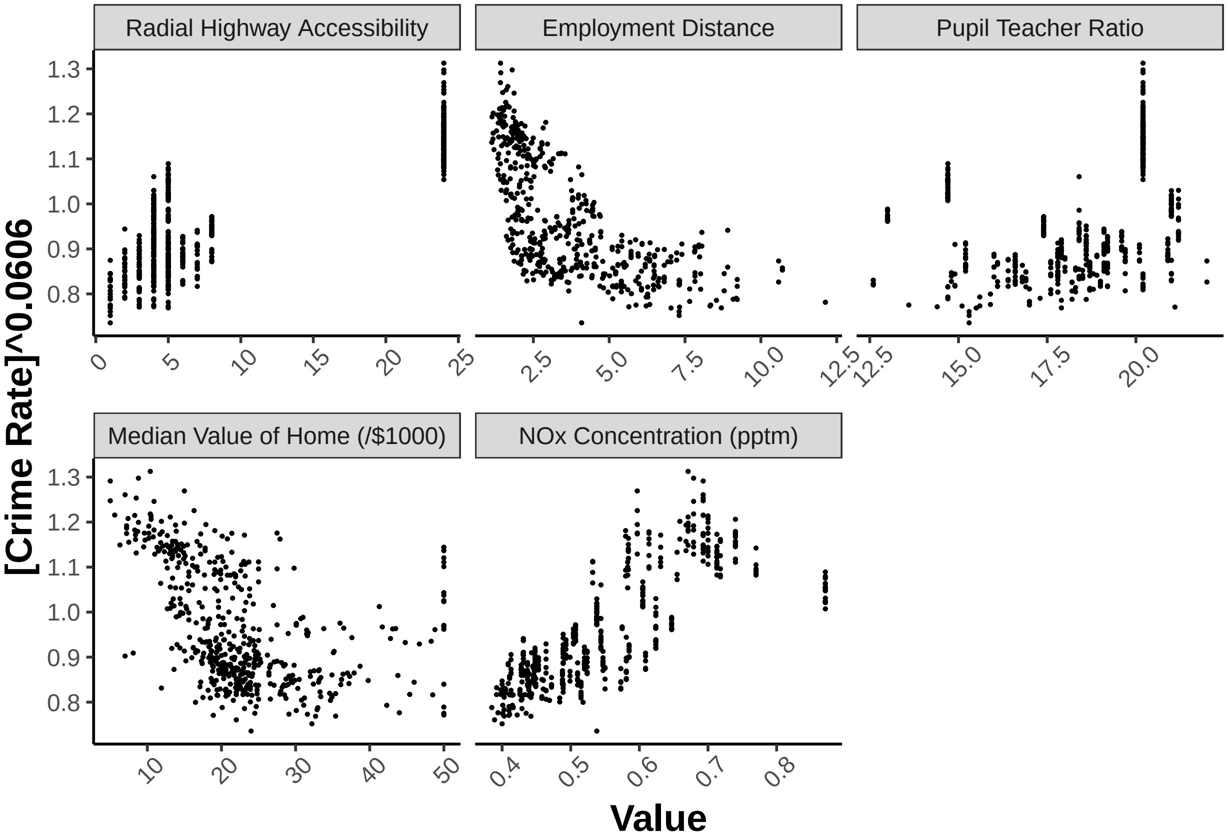
Distribution of Variables

```
dfnew2 = dfnew[,c(’crim’,’rad’,’dis’, ’ptratio’, ’medv’, ’nox’)]
colnames(dfnew2) = c(’Crime Rate’, ’Radial Highway Accessibility’, ’Employment Distance’, ’Pupil Teache

mlr6 = lm(‘Crime Rate‘ ~ 0 +‘Radial Highway Accessibility‘+‘Employment Distance‘ + ‘Pupil Teacher Ratio


outp = "../plots/"
distplot = plotdists(dfnew2,outp, brtype = ’FD’)
distplot
```



Scatter Plot of All Variables

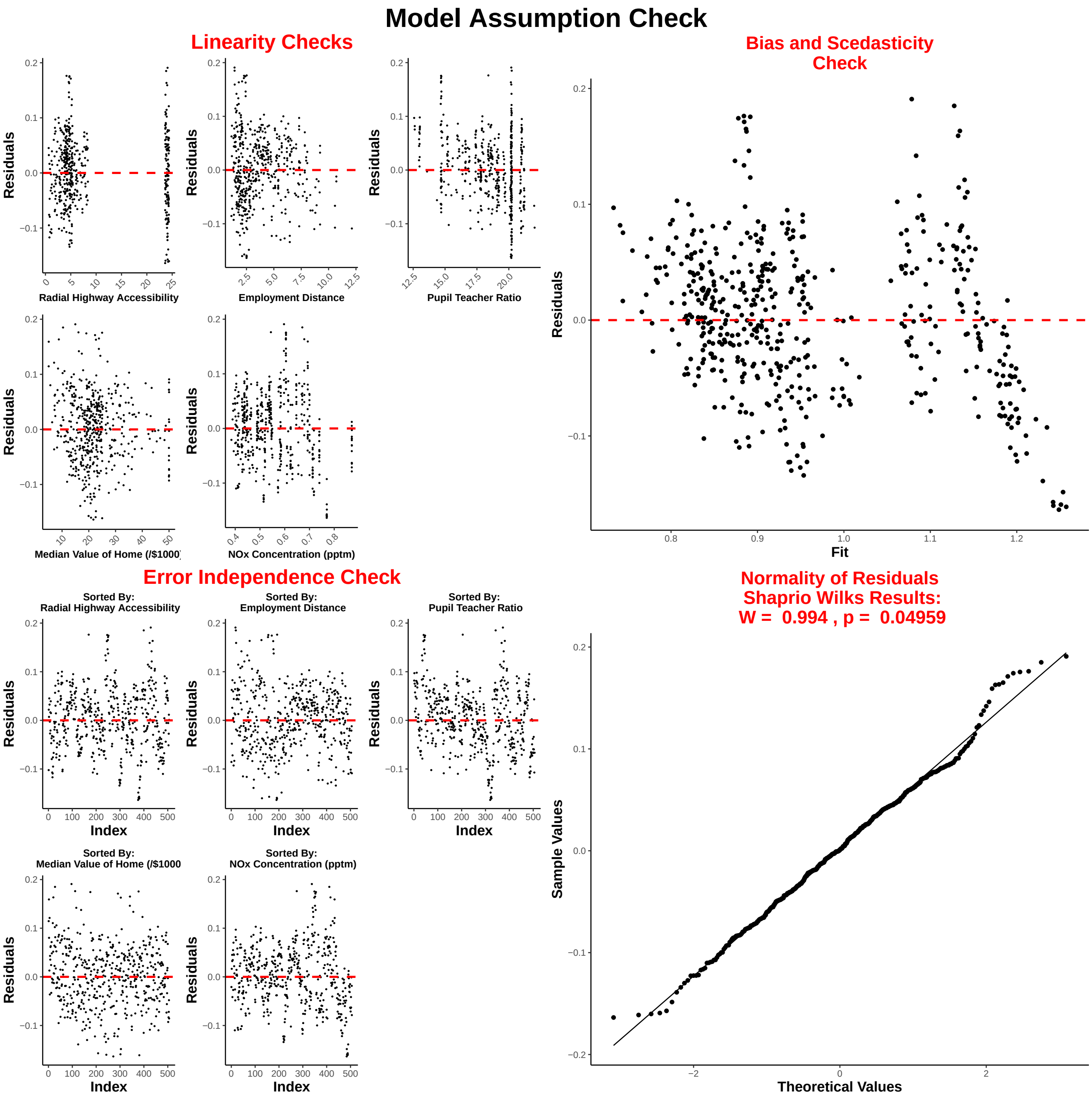
```
df.m = melt(dfnew2,id.vars = ’Crime Rate’)
colnames(df.m) = c(’Crime Rate’, ’Variable’, ’Value’)
#df.m$Variable = apply(strsplit2(df.m$Variable, split = "_")[,1:5], 1, paste, collapse = " ")
ggplot(data=df.m, aes(x=Value, y = ‘Crime Rate‘))+
  geom_point(size = 0.25)+plot_opts+facet_wrap(~Variable, scales = ’free_x’)+
  ylab(paste(’[Crime Rate]^’, as.character(round(lambda,4)), sep = ""))+theme(axis.text.x=element_text(
```



```
ggsave(filename = paste(outp,'scatter.png',sep=""), dpi = 600, width = 1.5*5, height = 5, units = 'in')
```

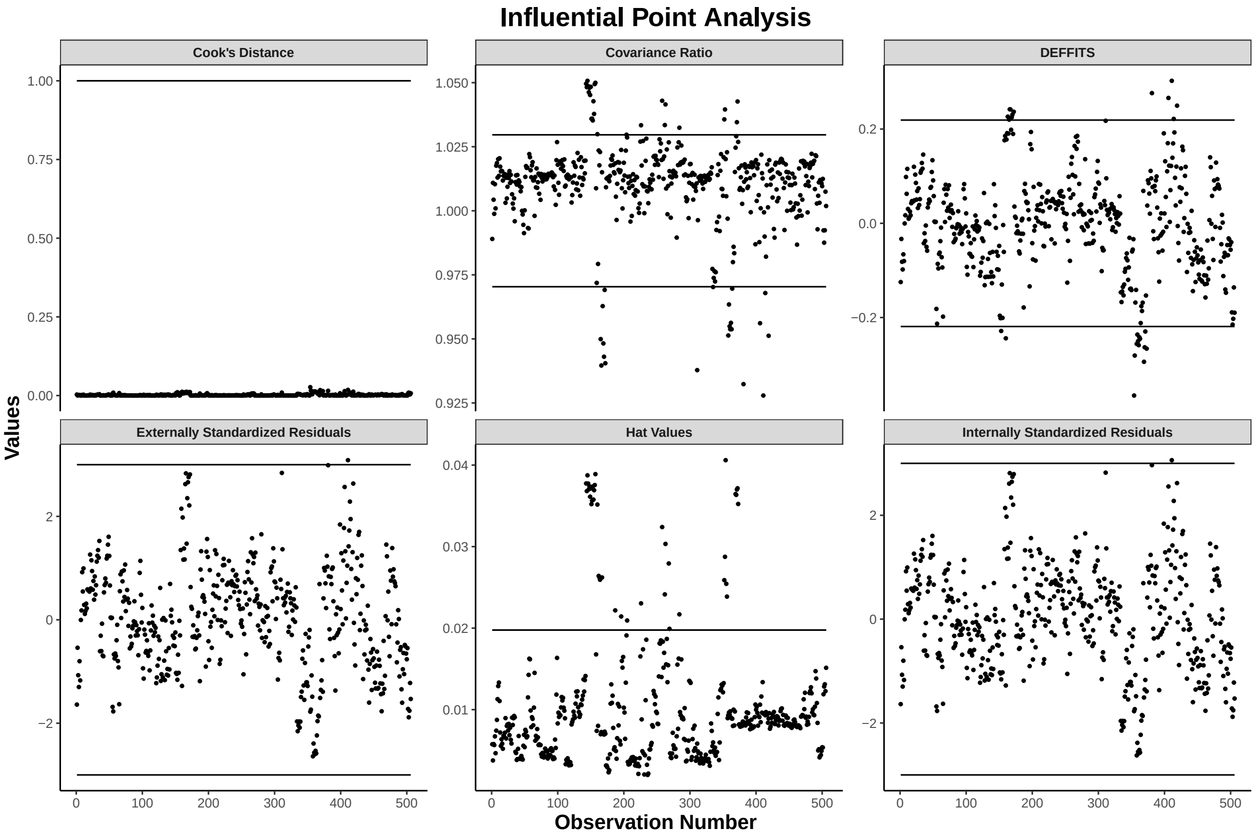
Assumption Check

```
pf = Assumption_Check(mlr6, outp)
pf
```



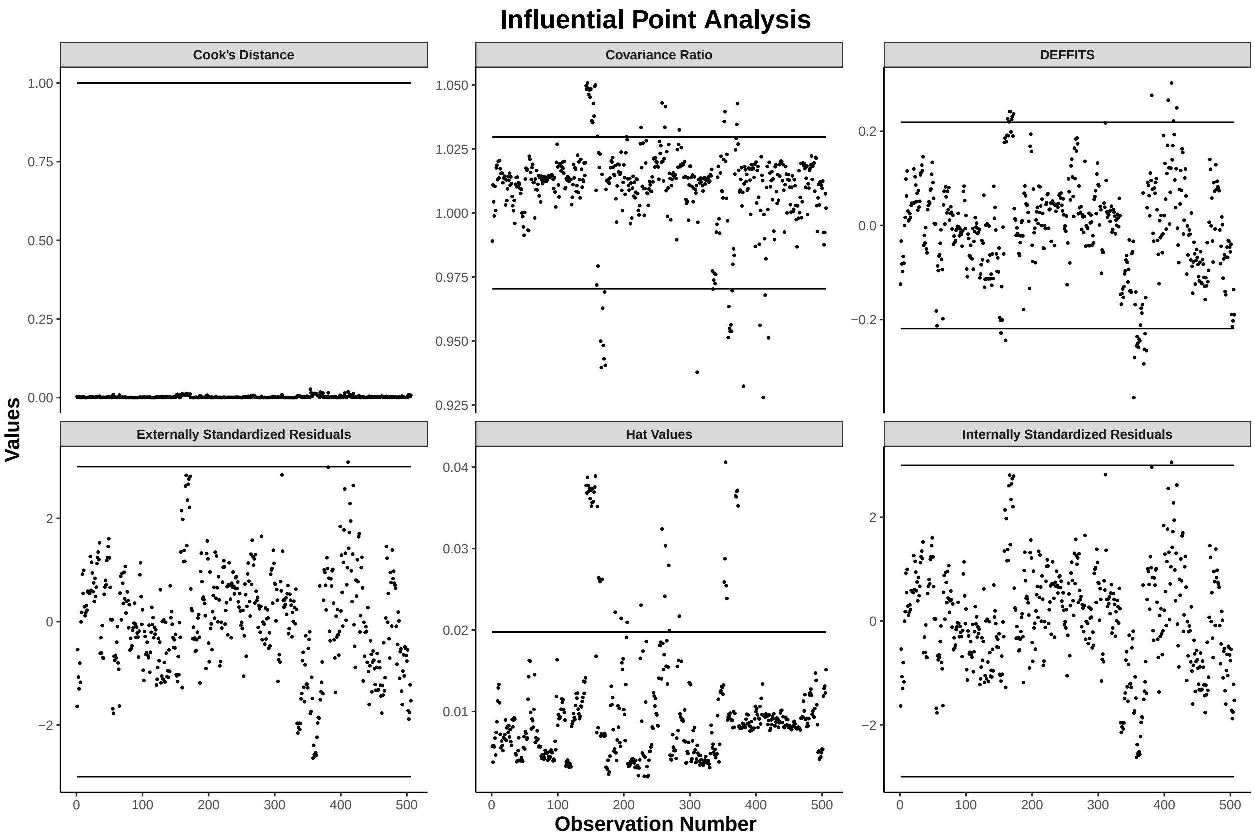
Influential Analysis

```
ret_df = infl_analysis(mlr6)
ret_df = cbind(ret_df, dfnew2)
p = ggplot(data= ret_df, aes(x= Row_Num, y = Values))+
  geom_point(size = .8)+
  facet_wrap(~Type, scales = "free_y")+plot_opts+geom_line(aes(y=Bound1))+geom_line(aes(y=Bound2))+
  #geom_label_repel(aes(label=Label), size = 4)+
  labs(title = ’Influential Point Analysis’, x = ’Observation Number’)+ theme(strip.text = element_text
p
```



```
ggsave(filename = ’../plots/infl.png’, plot = p, width = 1.5*10, height = 10, units = ’in’, limitsize =
```

```
p = ggplot(data= ret_df, aes(x= Row_Num, y = Values))+
  geom_point(size = .5)+
  facet_wrap(~Type, scales = "free_y")+plot_opts+geom_line(aes(y=Bound1))+geom_line(aes(y=Bound2))+
  #geom_label_repel(aes(label=Label), size = 4)+
  labs(title = ’Influential Point Analysis’, x = ’Observation Number’)+ theme(strip.text = element_text
p
```



```
ggsave(filename = ’../plots/infl.png’, plot = p, width = 1.5*6, height = 6, units = ’in’, limitsize = F
```

Hypothesis Testing

Test for significance of regression: H0: All regression coefficients are equal to zero. H1: At least one regression coefficient is not equal to 0.

```
linearHypothesis(mlr5, c(’rad = 0’, ’dis = 0’, ’ptratio = 0’, ’medv = 0’, ’nox = 0’))
```

```
## Linear hypothesis test
##
## Hypothesis:
## rad = 0
## dis = 0
## ptratio = 0
## medv = 0
## nox = 0
##
## Model 1: restricted model
## Model 2: crim ~ 0 + rad * dis + ptratio + medv + nox
##
##   Res.Df     RSS Df Sum of Sq      F    Pr(>F)
## 1    505 127.341
## 2    500   1.899  5    125.44 6604.9 < 2.2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

Confidence Intervals

```
rmse = sqrt(mean(mlr6$residuals^2))
knitr::kable(confint(mlr6))
```

	2.5 %	97.5 %
Radial Highway Accessibility	0.0057384	0.0073894
Employment Distance	0.0071787	0.0136930
Pupil Teacher Ratio	0.0167573	0.0206891
Median Value of Home (/$1000)	0.0019411	0.0031086
NOx Concentration (pptm)	0.7661554	0.8756984

Other Metrics

```
rmse = sqrt(mean(mlr6$residuals^2))
rmse
```

```
## [1] 0.0623321
```

```
AIC(mlr6)
```

```
## [1] -1360.616
```

```
BIC(mlr6)
```

```
## [1] -1335.257
```