

BM1 Final Project

Yijia Jiang, Yifei Xu, Xinyi Zhou, Hengxuan Ma, Chao Gao

11/16/2021

Purpose

We will be analyzing data from the “County Demographic Information” (CDI) data set, which contains characteristics of 440 counties in the United States collected from 1990-1992. The primary objective of this investigation is to develop insight relevant to predicting the crime rate in counties.

Import the package we need

Data preprocessing

Transfer population variables to per capita variables

```
rm(list = ls())
cdi = read.csv("./data/cdi.csv") %>%
  mutate(crime_rate = crimes/pop,
         pcarea = area/pop,
         pcdocs = docs/pop,
         pcbeds = beds/pop,
         region = relevel(factor(region), ref = 3))
cdi_pc = cdi %>%
  dplyr::select(crime_rate, everything(), -id, -cty, -state, -area, -docs, -beds, -crimes, -totalinc)
summary(cdi_pc)
```

##	crime_rate	pop	pop18	pop65
##	Min. :0.004601	Min. : 100043	Min. :16.40	Min. : 3.000
##	1st Qu.:0.038102	1st Qu.: 139027	1st Qu.:26.20	1st Qu.: 9.875
##	Median :0.052429	Median : 217280	Median :28.10	Median :11.750
##	Mean :0.057286	Mean : 393011	Mean :28.57	Mean :12.170
##	3rd Qu.:0.072597	3rd Qu.: 436064	3rd Qu.:30.02	3rd Qu.:13.625
##	Max. :0.295987	Max. :8863164	Max. :49.70	Max. :33.800
##	hsgrad	bagrad	poverty	unemp
##	Min. :46.60	Min. : 8.10	Min. : 1.400	Min. : 2.200
##	1st Qu.:73.88	1st Qu.:15.28	1st Qu.: 5.300	1st Qu.: 5.100
##	Median :77.70	Median :19.70	Median : 7.900	Median : 6.200
##	Mean :77.56	Mean :21.08	Mean : 8.721	Mean : 6.597
##	3rd Qu.:82.40	3rd Qu.:25.32	3rd Qu.:10.900	3rd Qu.: 7.500
##	Max. :92.90	Max. :52.30	Max. :36.300	Max. :21.300

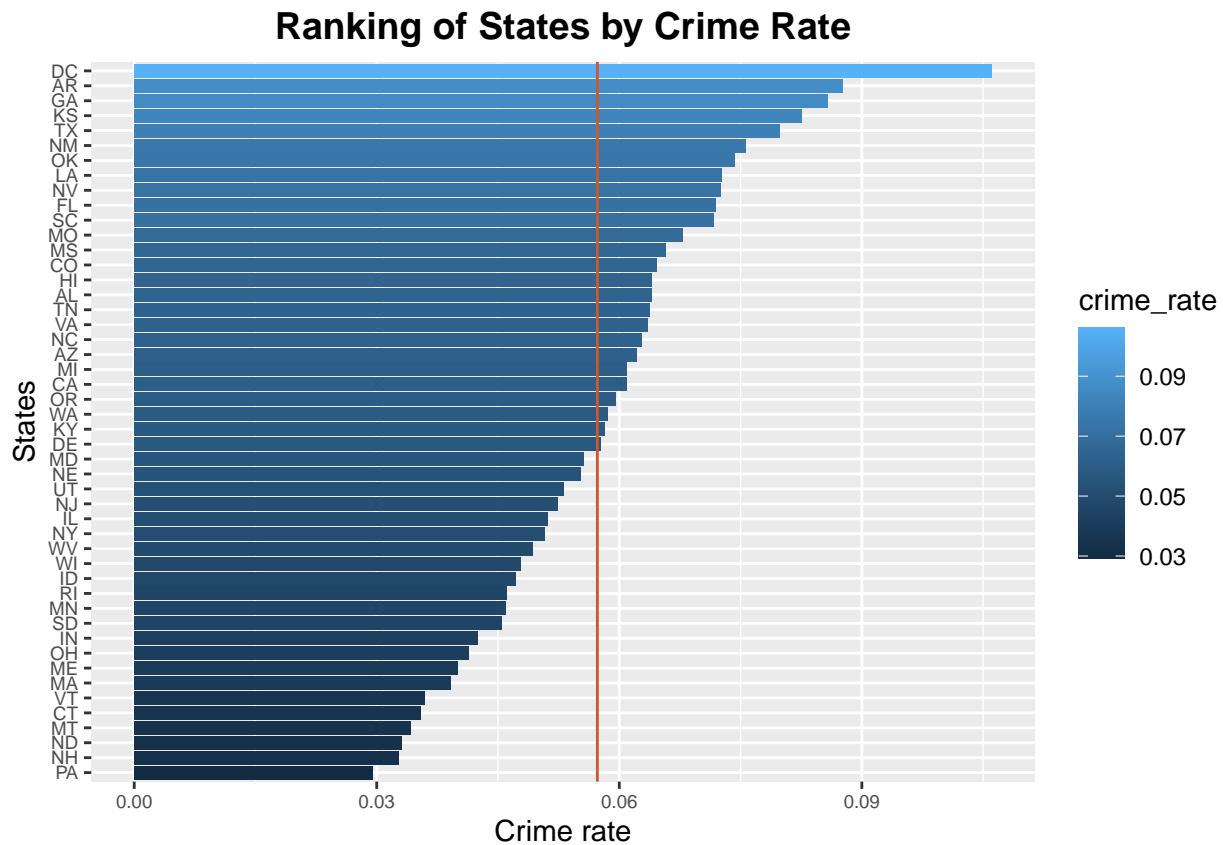
```
##      pcincome      region      pcarea      pcdocs
## Min.      : 8899      3:152      Min.      :3.086e-05      Min.      :0.0003559
## 1st Qu.:16118      1:103      1st Qu.:1.323e-03      1st Qu.:0.0012127
## Median :17759      2:108      Median :2.977e-03      Median :0.0017509
## Mean      :18561      4: 77      Mean      :4.760e-03      Mean      :0.0021230
## 3rd Qu.:20270              3rd Qu.:5.199e-03      3rd Qu.:0.0024915
## Max.      :37541              Max.      :7.542e-02      Max.      :0.0170377
##      pcbeds
## Min.      :0.0001649
## 1st Qu.:0.0021972
## Median :0.0033287
## Mean      :0.0036493
## 3rd Qu.:0.0045649
## Max.      :0.0196982
```

Exploratory Data Analysis

Rank of the crime rate by state

```
cdi_state <- cdi %>%
  group_by(state) %>%
  summarize(crime_rate = mean(crime_rate))
#cdi_state_rank <- cdi_state[order(-rank(cdi_state$crime_rate)),]

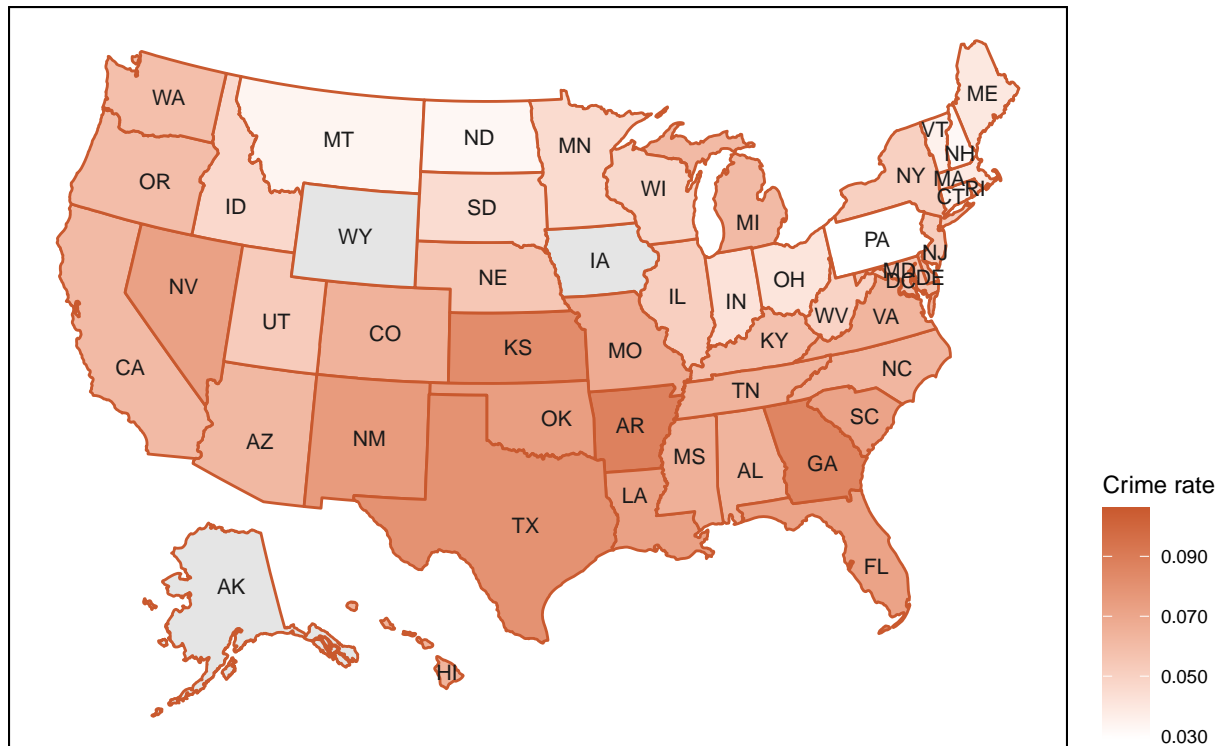
ggplot(cdi_state,aes(x=reorder(state,crime_rate),y=crime_rate,fill=crime_rate)) +
  geom_bar(stat = 'identity')+
  coord_flip() +
  theme_grey() +
  labs(title = 'Ranking of States by Crime Rate',
       y='Crime rate',x='States') +
  geom_hline(yintercept = mean(cdi$crime_rate),color = "#C9592E")+
  theme(plot.title = element_text(hjust = 0.5,size = 14, face = "bold"),
        axis.text=element_text(size=6.5))
```



US crime rate map by state

```
p<-plot_usmap(data = cdi_state, values = "crime_rate", color = "#C9592E", size = 0.5,
              labels = TRUE, label_color = "grey10") +
  scale_fill_continuous(low = "white", high = "#C9592E", na.value = "grey90",
                       name = "Crime rate", label = scales::comma) +
  theme(legend.position = "right") +
  theme(panel.background = element_rect(colour = "black")) +
  labs(title = "US Crime Rate Map") +
  theme(plot.title = element_text(hjust = 0.5, size = 14, face = "bold"))
p$layers[[2]]$aes_params$size <- 2.8
print(p)
```

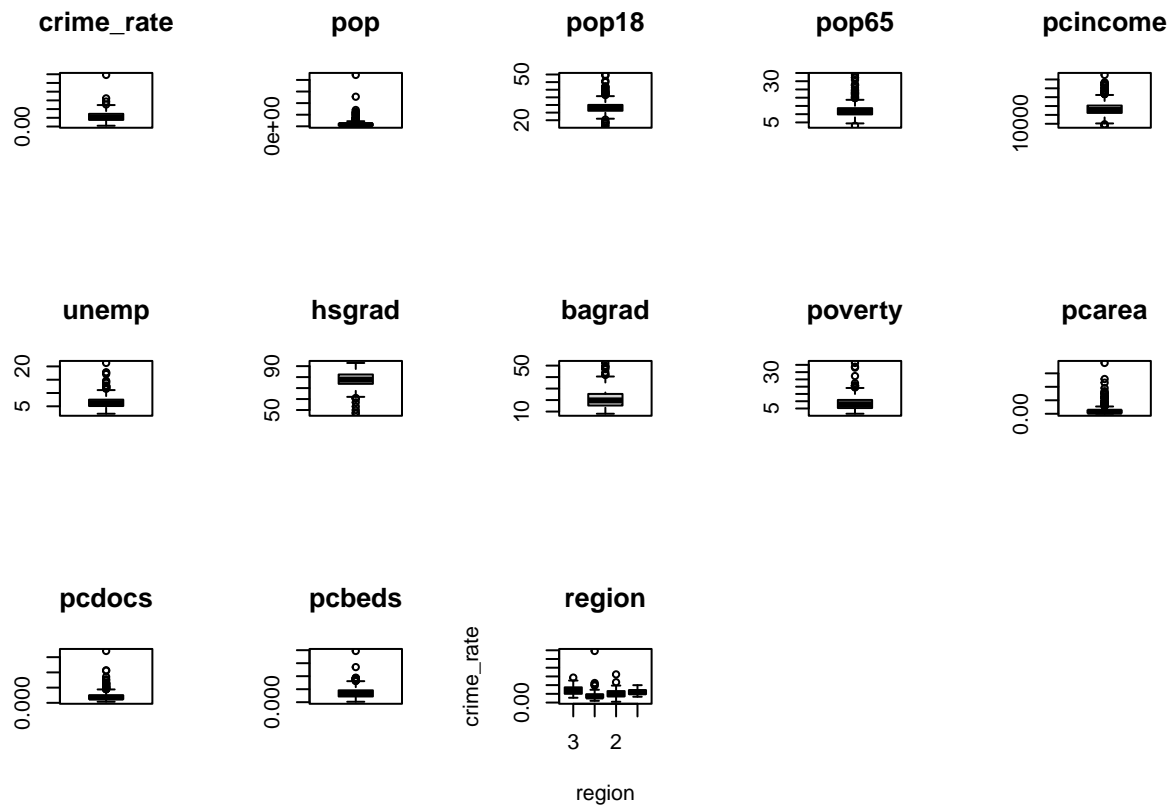
US Crime Rate Map



Boxplot for each variable

```
par(mfrow=c(3,5))
boxplot(cdi_pc$crime_rate, main = "crime_rate")
boxplot(cdi_pc$pop, main = "pop")
boxplot(cdi_pc$pop18, main = "pop18")
boxplot(cdi_pc$pop65, main = "pop65")
boxplot(cdi_pc$pcincome, main = "pcincome")
boxplot(cdi_pc$unemp, main = "unemp")
boxplot(cdi_pc$hsgrad, main = "hsgrad")
boxplot(cdi_pc$bagrad, main = "bagrad")
boxplot(cdi_pc$poverty, main = "poverty")
boxplot(cdi_pc$pcarea, main = "pcarea")
boxplot(cdi_pc$pcdocs, main = "pcdocs")
boxplot(cdi_pc$pcbeds, main = "pcbeds")

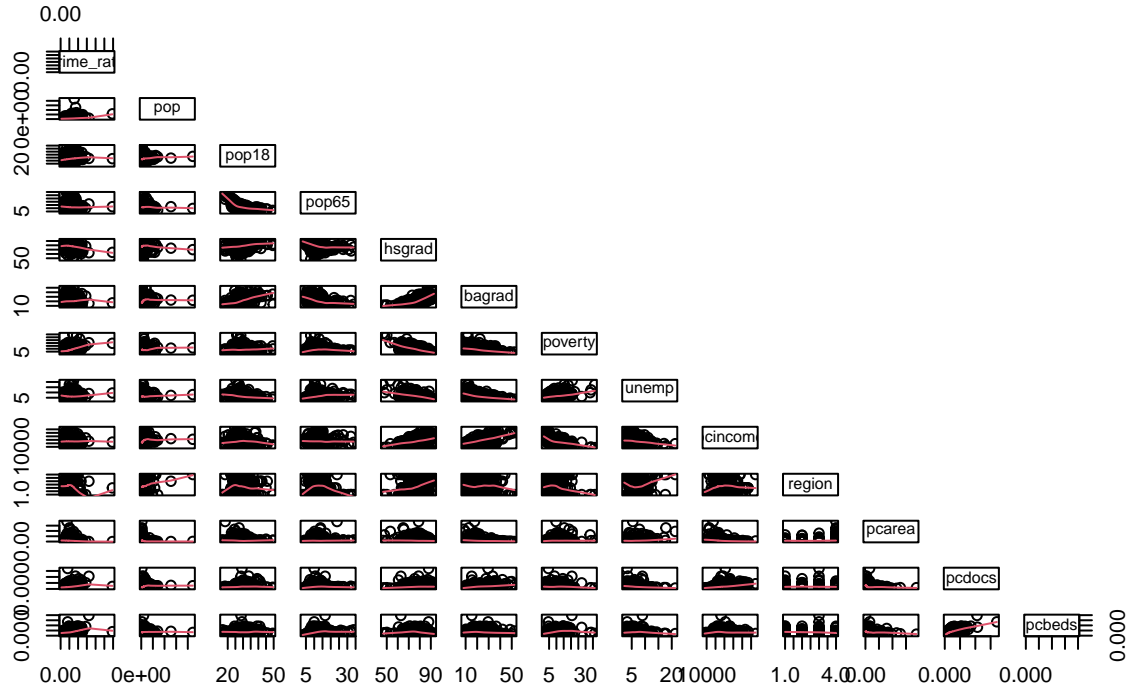
boxplot(crime_rate ~ region, data = cdi_pc, main = "region")
```



Scatterplot Matrix

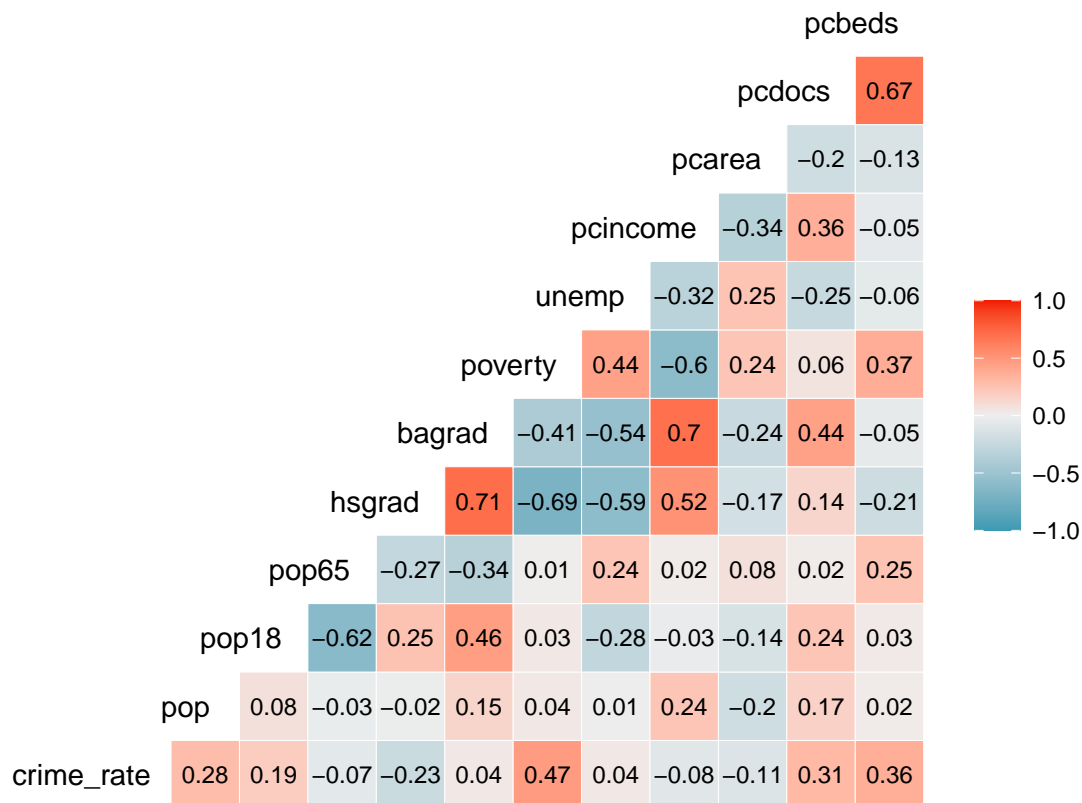
```
pairs(~crime_rate + ., data=cdi_pc, panel = panel.smooth, upper.panel = NULL, main = "Scatterplot Matrix")
```

Scatterplot Matrix



Correlation plot/ Heatmap

```
cdi_pc %>%
  dplyr::select(-region) %>%
  ggcorr(label = TRUE, hjust = 0.9, layout.exp = 2, label_size = 3, label_round = 2)
```



Modelling

Fit regression using all predictors

```
mult_fit = lm(crime_rate ~ ., data = cdi_pc)
summary(mult_fit)
```

```
##
## Call:
## lm(formula = crime_rate ~ ., data = cdi_pc)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.066728 -0.010908 -0.000201  0.009418  0.211437
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4.781e-02  3.043e-02  -1.571  0.11690
## pop          7.796e-09  1.773e-09   4.397 1.39e-05 ***
## pop18        1.111e-03  3.674e-04   3.024  0.00264 **
## pop65        1.204e-04  3.423e-04   0.352  0.72518
## hsgrad       2.926e-04  3.003e-04   0.974  0.33045
## bagrad      -4.633e-04  3.350e-04  -1.383  0.16737
## poverty      2.539e-03  4.273e-04   5.942 5.86e-09 ***
##
```

```
## unemp      4.462e-04  5.944e-04  0.751  0.45326
## pcincome   1.746e-06  5.308e-07  3.289  0.00109 **
## region1    -2.383e-02  2.963e-03 -8.042 8.87e-15 ***
## region2    -1.601e-02  2.894e-03 -5.533 5.51e-08 ***
## region4    -2.401e-03  3.482e-03 -0.689 0.49090
## pcarea     -4.886e-01  1.701e-01 -2.873 0.00427 **
## pcdocs     4.313e-01  1.134e+00  0.380 0.70392
## pcbeds     2.671e+00  8.893e-01  3.003 0.00283 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01991 on 425 degrees of freedom
## Multiple R-squared:  0.4862, Adjusted R-squared:  0.4693
## F-statistic: 28.73 on 14 and 425 DF,  p-value: < 2.2e-16
```

Backwards Elimination

```
mult_fit_back <- step(mult_fit, direction='backward')
```

```
## Start:  AIC=-3431.87
## crime_rate ~ pop + pop18 + pop65 + hsgrad + bagrad + poverty +
##      unemp + pcincome + region + pcarea + pcdocs + pcbeds
##
##           Df Sum of Sq    RSS    AIC
## - pop65    1 0.0000490 0.16850 -3433.7
## - pcdocs    1 0.0000573 0.16851 -3433.7
## - unemp     1 0.0002234 0.16867 -3433.3
## - hsgrad    1 0.0003763 0.16883 -3432.9
## - bagrad    1 0.0007582 0.16921 -3431.9
## <none>             0.16845 -3431.9
## - pcarea    1 0.0032712 0.17172 -3425.4
## - pcbeds    1 0.0035749 0.17203 -3424.6
## - pop18     1 0.0036248 0.17208 -3424.5
## - pcincome  1 0.0042869 0.17274 -3422.8
## - pop       1 0.0076647 0.17612 -3414.3
## - poverty   1 0.0139957 0.18245 -3398.8
## - region    3 0.0312633 0.19972 -3363.0
##
## Step:  AIC=-3433.74
## crime_rate ~ pop + pop18 + hsgrad + bagrad + poverty + unemp +
##      pcincome + region + pcarea + pcdocs + pcbeds
##
##           Df Sum of Sq    RSS    AIC
## - pcdocs    1 0.0000561 0.16856 -3435.6
## - unemp     1 0.0002518 0.16875 -3435.1
## - hsgrad    1 0.0003617 0.16886 -3434.8
## - bagrad    1 0.0007628 0.16926 -3433.8
## <none>             0.16850 -3433.7
## - pcarea    1 0.0032245 0.17172 -3427.4
## - pcbeds    1 0.0041665 0.17267 -3425.0
## - pop18     1 0.0042054 0.17271 -3424.9
## - pcincome  1 0.0042628 0.17276 -3424.7
```



```

## - pop      1 0.0077852 0.17629 -3415.9
## - poverty  1 0.0142181 0.18272 -3400.1
## - region   3 0.0312244 0.19973 -3364.9
##
## Step: AIC=-3435.59
## crime_rate ~ pop + pop18 + hsgrad + bagrad + poverty + unemp +
##      pcincome + region + pcarea + pcbeds
##
##           Df Sum of Sq    RSS    AIC
## - unemp      1 0.0002552 0.16881 -3436.9
## - hsgrad      1 0.0003389 0.16889 -3436.7
## - bagrad      1 0.0007073 0.16926 -3435.8
## <none>                0.16856 -3435.6
## - pcarea      1 0.0032310 0.17179 -3429.2
## - pop18       1 0.0043858 0.17294 -3426.3
## - pcincome    1 0.0045074 0.17306 -3426.0
## - pop         1 0.0078624 0.17642 -3417.5
## - pcbeds      1 0.0103601 0.17892 -3411.3
## - poverty     1 0.0141806 0.18274 -3402.1
## - region      3 0.0314610 0.20002 -3366.3
##
## Step: AIC=-3436.93
## crime_rate ~ pop + pop18 + hsgrad + bagrad + poverty + pcincome +
##      region + pcarea + pcbeds
##
##           Df Sum of Sq    RSS    AIC
## - hsgrad      1 0.000250 0.16906 -3438.3
## <none>                0.16881 -3436.9
## - bagrad      1 0.000935 0.16975 -3436.5
## - pcarea      1 0.003127 0.17194 -3430.9
## - pop18       1 0.004390 0.17320 -3427.6
## - pcincome    1 0.005080 0.17389 -3425.9
## - pop         1 0.007769 0.17658 -3419.1
## - pcbeds      1 0.010289 0.17910 -3412.9
## - poverty     1 0.016870 0.18568 -3397.0
## - region      3 0.032233 0.20104 -3366.0
##
## Step: AIC=-3438.28
## crime_rate ~ pop + pop18 + bagrad + poverty + pcincome + region +
##      pcarea + pcbeds
##
##           Df Sum of Sq    RSS    AIC
## - bagrad      1 0.000705 0.16977 -3438.4
## <none>                0.16906 -3438.3
## - pcarea      1 0.003317 0.17238 -3431.7
## - pop18       1 0.004209 0.17327 -3429.5
## - pcincome    1 0.004897 0.17396 -3427.7
## - pop         1 0.007570 0.17663 -3421.0
## - pcbeds      1 0.010555 0.17962 -3413.6
## - poverty     1 0.024140 0.19320 -3381.6
## - region      3 0.032219 0.20128 -3367.5
##
## Step: AIC=-3438.45
## crime_rate ~ pop + pop18 + poverty + pcincome + region + pcarea +

```

```
##      pcbeds
##
##           Df Sum of Sq      RSS       AIC
## <none>                0.16977 -3438.4
## - pcarea      1  0.003619 0.17339 -3431.2
## - pop18       1  0.004100 0.17387 -3429.9
## - pcincome    1  0.005499 0.17527 -3426.4
## - pop        1  0.008260 0.17803 -3419.5
## - pcbeds     1  0.010408 0.18018 -3414.3
## - poverty    1  0.024541 0.19431 -3381.0
## - region     3  0.032233 0.20200 -3368.0
```

```
mult_fit_back
```

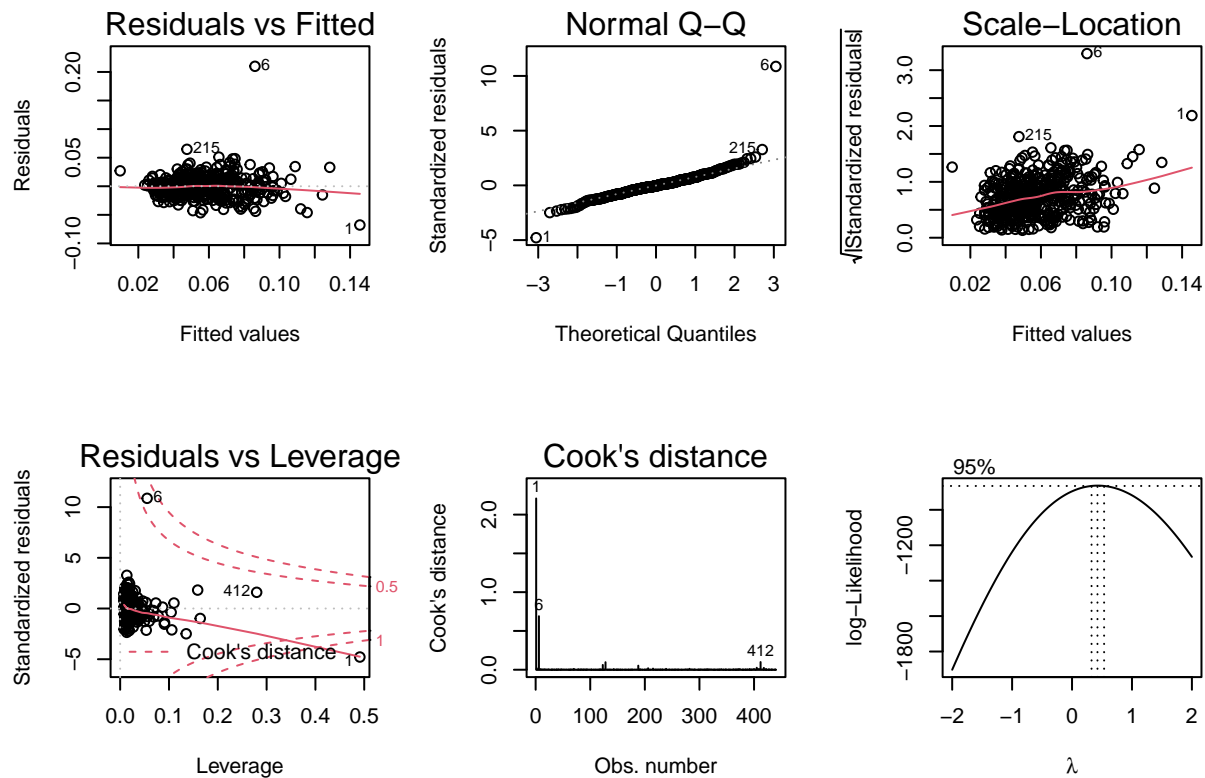
```
##
## Call:
## lm(formula = crime_rate ~ pop + pop18 + poverty + pcincome +
##      region + pcarea + pcbeds, data = cdi_pc)
##
## Coefficients:
## (Intercept)      pop      pop18      poverty      pcincome      region1
## -1.034e-02  7.962e-09  7.451e-04  2.423e-03  1.236e-06 -2.209e-02
##      region2      region4      pcarea      pcbeds
## -1.442e-02 -1.124e-03 -5.063e-01  2.868e+00
```

```
crime_rate ~ pop + pop18 + poverty + pcincome + region + pcarea + pcbeds
```

Model Diagnostics

Create Residuals vs Fitted plot & Normal Q-Q plot & Scale-Location plot & Residuals vs Leverage plot to detect the normality of residuals and outliers

```
par(mfrow=c(2,3))
plot(mult_fit_back)
plot(mult_fit_back, which = 4)
bc = boxcox(mult_fit_back)
```



Diagnose the model without outliers

```
# remove influential points
cdi_pc_out = cdi_pc[-c(1,6,412),]

# fit model with and without influential points
mult_fit_back_without = lm(crime_rate ~ pop + pop18 + poverty + pcincome + region + pcarea + pcbeds, data = cdi_pc_out)

summary(mult_fit_back)
```

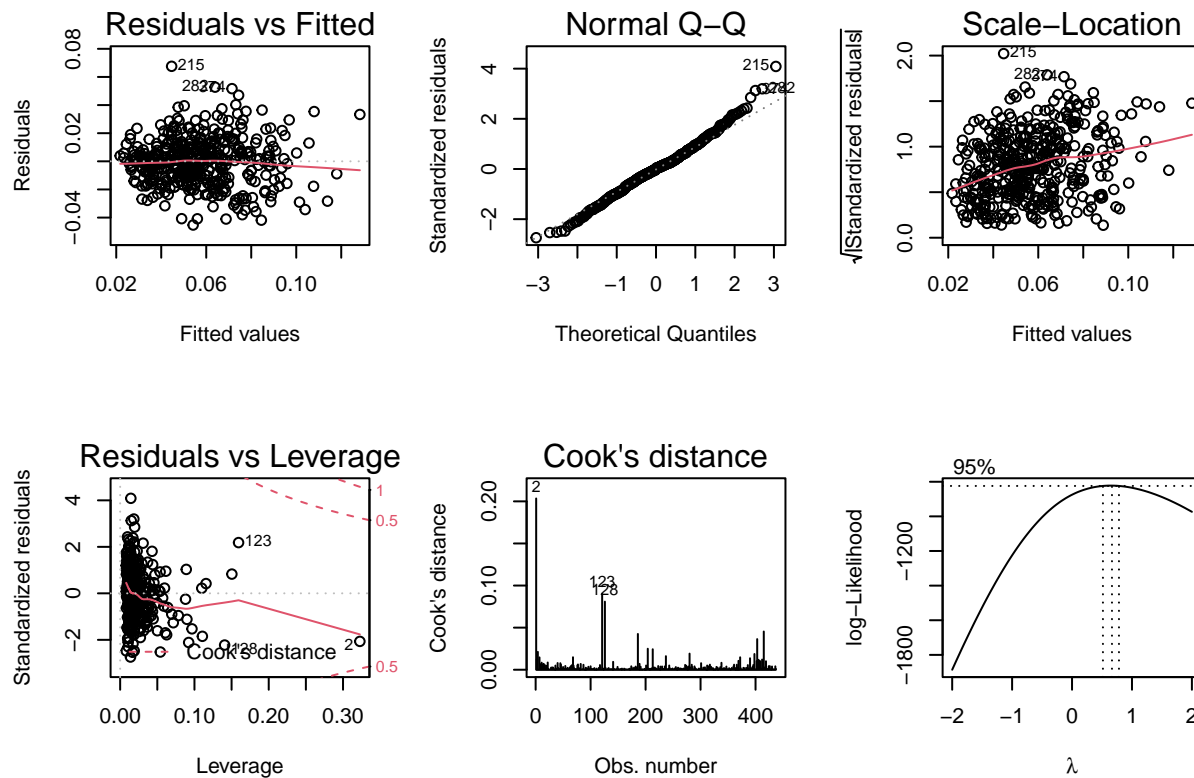
```
##
## Call:
## lm(formula = crime_rate ~ pop + pop18 + poverty + pcincome +
##     region + pcarea + pcbeds, data = cdi_pc)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.067811 -0.010690 -0.000725  0.010118  0.209881
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.034e-02  1.060e-02  -0.976  0.329595
## pop          7.962e-09  1.741e-09   4.574  6.26e-06 ***
## pop18        7.451e-04  2.312e-04   3.223  0.001367 **
## poverty      2.423e-03  3.073e-04   7.884  2.63e-14 ***
## pcincome     1.236e-06  3.312e-07   3.732  0.000216 ***
```

```
## region1      -2.209e-02  2.730e-03  -8.093  6.01e-15 ***
## region2      -1.442e-02  2.616e-03  -5.511  6.15e-08 ***
## region4      -1.124e-03  3.188e-03  -0.352  0.724665
## pcarea       -5.063e-01  1.672e-01  -3.027  0.002615 **
## pcbeds       2.868e+00  5.586e-01   5.134  4.30e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01987 on 430 degrees of freedom
## Multiple R-squared:  0.4822, Adjusted R-squared:  0.4713
## F-statistic: 44.49 on 9 and 430 DF,  p-value: < 2.2e-16
```

```
summary(mult_fit_back_without)
```

```
##
## Call:
## lm(formula = crime_rate ~ pop + pop18 + poverty + pcincome +
##      region + pcarea + pcbeds, data = cdi_pc_out)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.045291 -0.010145  0.000322  0.009630  0.067563
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.857e-03  8.932e-03  -0.432  0.666122
## pop          1.068e-08  2.077e-09   5.144  4.11e-07 ***
## pop18        7.779e-04  1.940e-04   4.009  7.20e-05 ***
## poverty      1.907e-03  2.635e-04   7.237  2.14e-12 ***
## pcincome     9.829e-07  2.839e-07   3.463  0.000589 ***
## region1     -2.542e-02  2.306e-03 -11.019 < 2e-16 ***
## region2     -1.569e-02  2.197e-03  -7.140  4.04e-12 ***
## region4     -9.152e-04  2.711e-03  -0.338  0.735789
## pcarea      -5.121e-01  1.642e-01  -3.119  0.001935 **
## pcbeds      3.262e+00  4.696e-01   6.946  1.41e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01665 on 427 degrees of freedom
## Multiple R-squared:  0.5614, Adjusted R-squared:  0.5521
## F-statistic: 60.73 on 9 and 427 DF,  p-value: < 2.2e-16
```

```
# diagnose the model without outliers
par(mfrow=c(2,3))
plot(mult_fit_back_without)
plot(mult_fit_back_without, which = 4)
bc_without = boxcox(mult_fit_back_without)
```



Box-cox transformation

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

## [1] 0.6666667

lambda

## [1] 0.6666667

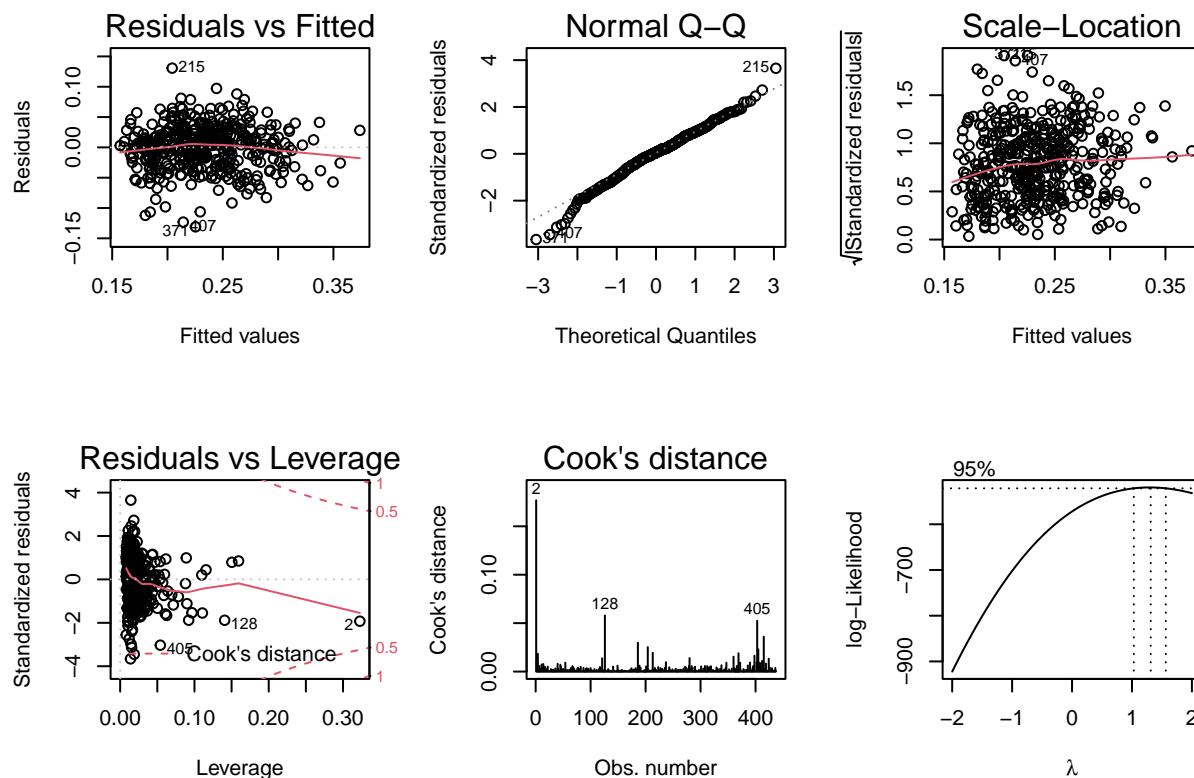
mult_fit_back_without_trans = lm(crime_rate^0.5 ~ pop + pop18 + poverty + pcincome +
                                region + pcarea + pcbeds, data = cdi_pc_out)

summary(mult_fit_back_without_trans)

##
## Call:
## lm(formula = crime_rate^0.5 ~ pop + pop18 + poverty + pcincome +
##     region + pcarea + pcbeds, data = cdi_pc_out)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.131052 -0.020616  0.001528  0.023104  0.130713
##
## Coefficients:
```

```
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.087e-01  1.934e-02   5.621 3.43e-08 ***
## pop          2.226e-08  4.496e-09   4.951 1.07e-06 ***
## pop18        1.608e-03  4.201e-04   3.827 0.000149 ***
## poverty      3.798e-03  5.705e-04   6.658 8.57e-11 ***
## pcincome     2.093e-06  6.146e-07   3.405 0.000724 ***
## region1     -5.709e-02  4.994e-03 -11.432 < 2e-16 ***
## region2     -3.463e-02  4.756e-03  -7.282 1.59e-12 ***
## region4      2.682e-04  5.868e-03   0.046 0.963566
## pcarea      -9.919e-01  3.554e-01  -2.791 0.005490 **
## pcbeds       6.543e+00  1.017e+00   6.435 3.31e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03605 on 427 degrees of freedom
## Multiple R-squared:  0.547, Adjusted R-squared:  0.5374
## F-statistic: 57.28 on 9 and 427 DF,  p-value: < 2.2e-16
```

```
# Diagnose the model by square root transformation
par(mfrow = c(2,3))
plot(mult_fit_back_without_trans)
plot(mult_fit_back_without_trans, which = 4)
bc_without_trans = boxcox(mult_fit_back_without_trans)
```



Compare the Adjusted R²

```
rbind(mult_fit_back %>% broom::glance() %>% mutate(model_type = "mult_fit_back"),
      mult_fit_back_without %>% broom::glance() %>% mutate(model_type = "mult_fit_back_without"),
      mult_fit_back_without_trans %>% broom::glance() %>% mutate(model_type = "mult_fit_back_without_trans"),
      dplyr::select(model_type, everything()))
```

```
## # A tibble: 3 x 13
##   model_type      r.squared adj.r.squared  sigma statistic  p.value    df logLik
##   <chr>          <dbl>      <dbl>  <dbl>    <dbl>    <dbl> <dbl> <dbl>
## 1 mult_fit_back    0.482        0.471 0.0199    44.5 3.64e-56     9 1105.
## 2 mult_fit_back_~  0.561        0.552 0.0167    60.7 6.54e-71     9 1175.
## 3 mult_fit_back_~  0.547        0.537 0.0360    57.3 6.01e-68     9  837.
## # ... with 5 more variables: AIC <dbl>, BIC <dbl>, deviance <dbl>,
## #   df.residual <int>, nobs <int>
```

Assessing Multicollinearity

```
# Calculate the variance inflation factor (VIF)
check_collinearity(mult_fit_back_without)
```

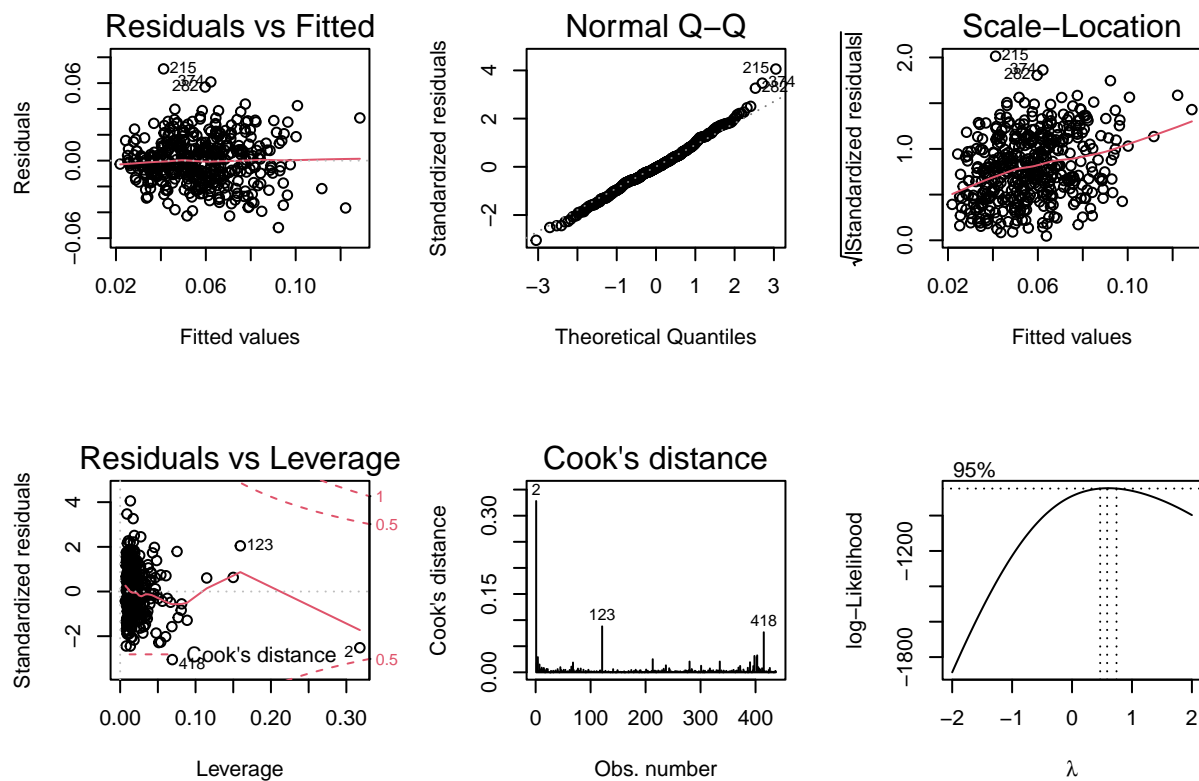
```
## # Check for Multicollinearity
##
## Low Correlation
##
##      Term  VIF Increased SE Tolerance
##      pop  1.00      1.00      1.00
##      pop18 4.82      2.20      0.21
##      pcincome 1.00      1.00      1.00
##      region 2.08      1.44      0.48
##      pcarea 1.96      1.40      0.51
##      pcbeds 3.92      1.98      0.26
##
## Moderate Correlation
##
##      Term  VIF Increased SE Tolerance
##      poverty 9.39      3.06      0.11
```

```
# Remove the variable with high VIF
mult_fit_back_without_vif = lm(crime_rate ~ pop + pop18 + pcincome + region + pcarea + pcbeds, data = cdi_pc_out)
summary(mult_fit_back_without_vif)
```

```
##
## Call:
## lm(formula = crime_rate ~ pop + pop18 + pcincome + region + pcarea +
##      pcbeds, data = cdi_pc_out)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -0.051836 -0.010389 -0.000806 0.010247 0.070996
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.984e-02  8.067e-03   3.699 0.000245 ***
## pop          1.470e-08  2.118e-09   6.943 1.44e-11 ***
## pop18        7.500e-04  2.053e-04   3.653 0.000291 ***
## pcincome     -2.241e-07  2.431e-07  -0.922 0.357136
## region1      -2.898e-02  2.385e-03 -12.154 < 2e-16 ***
## region2      -2.015e-02  2.231e-03  -9.030 < 2e-16 ***
## region4      -2.505e-03  2.859e-03  -0.876 0.381423
## pcarea       -2.990e-01  1.709e-01  -1.750 0.080901 .
## pcbeds        4.854e+00  4.390e-01  11.057 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01762 on 428 degrees of freedom
## Multiple R-squared:  0.5076, Adjusted R-squared:  0.4984
## F-statistic: 55.15 on 8 and 428 DF,  p-value: < 2.2e-16
```

```
# Diagnose the model removing poverty term
par(mfrow = c(2,3))
plot(mult_fit_back_without_vif)
plot(mult_fit_back_without_vif, which = 4)
bc_without_trans = boxcox(mult_fit_back_without_vif)
```



Add the interaction terms

```
mult_fit_back_without_int = lm(crime_rate ~ pop + pop18 + pcincome + region + pcarea + pcbeds + poverty  
fit1 =step(mult_fit_back_without_int,direction="both")
```

```
## Start: AIC=-3608.07  
## crime_rate ~ pop + pop18 + pcincome + region + pcarea + pcbeds +  
## poverty + pcincome * poverty  
##  
##           Df Sum of Sq    RSS    AIC  
## <none>                0.10787 -3608.1  
## - pcarea             1  0.001419 0.10929 -3604.4  
## - pcbeds             1  0.002325 0.11019 -3600.7  
## - pop               1  0.003705 0.11157 -3595.3  
## - pop18             1  0.003926 0.11179 -3594.4  
## - pcincome:poverty  1  0.010513 0.11838 -3569.4  
## - region            3  0.034788 0.14265 -3491.9
```

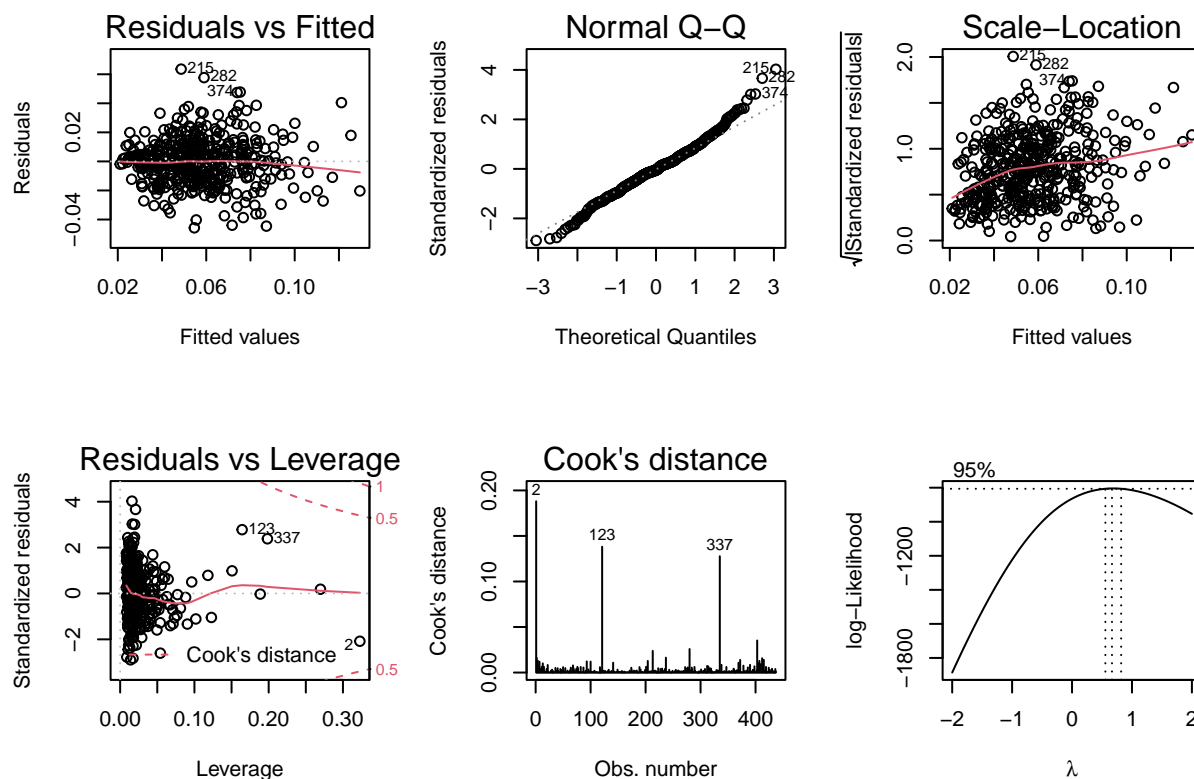
```
summary(mult_fit_back_without_int)
```

```
##  
## Call:  
## lm(formula = crime_rate ~ pop + pop18 + pcincome + region + pcarea +  
## pcbeds + poverty + pcincome * poverty, data = cdi_pc_out)  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max   
## -0.045731 -0.009495 -0.000758  0.008857  0.063537   
##  
## Coefficients:  
##              Estimate Std. Error t value Pr(>|t|)      
## (Intercept)   9.987e-03  8.802e-03   1.135  0.25717      
## pop           7.785e-09  2.035e-09   3.825  0.00015 ***  
## pop18         7.308e-04  1.856e-04   3.938 9.61e-05 ***  
## pcincome      8.776e-08  3.048e-07   0.288  0.77353      
## region1      -2.427e-02  2.211e-03 -10.977 < 2e-16 ***  
## region2      -1.504e-02  2.102e-03  -7.157 3.63e-12 ***  
## region4      -3.649e-03  2.625e-03  -1.390  0.16523      
## pcarea       -3.749e-01  1.583e-01  -2.368  0.01834 *     
## pcbeds        1.575e+00  5.196e-01   3.030  0.00259 **    
## poverty      -1.525e-03  5.891e-04  -2.589  0.00997 **    
## pcincome:poverty 2.707e-07  4.201e-08   6.444 3.16e-10 ***  
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
##  
## Residual standard error: 0.01591 on 426 degrees of freedom  
## Multiple R-squared:  0.6003, Adjusted R-squared:  0.591  
## F-statistic: 63.99 on 10 and 426 DF, p-value: < 2.2e-16
```

```
anova(mult_fit_back_without,mult_fit_back_without_int)
```

```
## Analysis of Variance Table
##
## Model 1: crime_rate ~ pop + pop18 + poverty + pcincome + region + pcarea +
##   pcbeds
## Model 2: crime_rate ~ pop + pop18 + pcincome + region + pcarea + pcbeds +
##   poverty + pcincome * poverty
##   Res.Df    RSS Df Sum of Sq    F    Pr(>F)
## 1      427 0.11838
## 2      426 0.10787   1  0.010513 41.52 3.16e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Diagnose the model with interaction term
par(mfrow = c(2,3))
plot(mult_fit_back_without_int)
plot(mult_fit_back_without_int, which = 4)
bc_without_trans = boxcox(mult_fit_back_without_int)
```



Compare the Adjusted R^2 again

```
rbind(mult_fit_back %>% broom::glance() %>% mutate(model_type = "mult_fit_back"),
      mult_fit_back_without %>% broom::glance() %>% mutate(model_type = "mult_fit_back_without"),
      mult_fit_back_without_trans %>% broom::glance() %>% mutate(model_type = "mult_fit_back_without_trans"))
```

```

mult_fit_back_without_vif %>% broom::glance() %>% mutate(model_type = "mult_fit_back_without_vif")
mult_fit_back_without_int %>% broom::glance() %>% mutate(model_type = "mult_fit_back_without_int")
dplyr::select(model_type, everything())

```

```

## # A tibble: 5 x 13
##   model_type      r.squared adj.r.squared  sigma statistic  p.value    df logLik
##   <chr>          <dbl>      <dbl>  <dbl>    <dbl>    <dbl> <dbl> <dbl>
## 1 mult_fit_back    0.482        0.471 0.0199    44.5 3.64e-56     9 1105.
## 2 mult_fit_back_~  0.561        0.552 0.0167    60.7 6.54e-71     9 1175.
## 3 mult_fit_back_~  0.547        0.537 0.0360    57.3 6.01e-68     9  837.
## 4 mult_fit_back_~  0.508        0.498 0.0176    55.1 3.21e-61     8 1149.
## 5 mult_fit_back_~  0.600        0.591 0.0159    64.0 1.71e-78    10 1195.
## # ... with 5 more variables: AIC <dbl>, BIC <dbl>, deviance <dbl>,
## #   df.residual <int>, nobs <int>

```

Model Validation

Compute RMSE and adjusted R^2 by cross-validation

```

set.seed(1234)
cv_df =
  crossv_kfold(cdi_pc_out, k = 10) %>%
  mutate(
    train = map(train, as_tibble),
    test = map(test, as_tibble)) %>%
  mutate(
    mult_fit_back_without = map(train, ~lm(crime_rate ~ pop + pop18 + poverty +
      pcincome + region + pcarea + pcbeds, data = .x)),
    mult_fit_back_without_trans = map(train, ~lm(sqrt(crime_rate) ~ pop + pop18 + poverty +
      pcincome + region + pcarea + pcbeds, data = .x)),
    mult_fit_back_without_vif = map(train, ~lm(crime_rate ~ pop + pop18 +
      pcincome + region + pcarea + pcbeds, data = .x)),
    mult_fit_back_without_int = map(train, ~lm(crime_rate ~ pop + pop18 + pcincome + region + pcarea +
      pcbeds, data = .x))) %>%
  mutate(
    rmse_model2 = map2_dbl(mult_fit_back_without, test, ~rmse(model = .x, data = .y)),
    rmse_model3 = map2_dbl(mult_fit_back_without_trans, test, ~rmse(model = .x, data = .y)),
    rmse_model4 = map2_dbl(mult_fit_back_without_vif, test, ~rmse(model = .x, data = .y)),
    rmse_model5 = map2_dbl(mult_fit_back_without_int, test, ~rmse(model = .x, data = .y))) %>%
  mutate(
    res_model2 = map(mult_fit_back_without, broom::glance %>% as.data.frame),
    res_model3 = map(mult_fit_back_without_trans, broom::glance %>% as.data.frame),
    res_model4 = map(mult_fit_back_without_vif, broom::glance %>% as.data.frame),
    res_model5 = map(mult_fit_back_without_int, broom::glance %>% as.data.frame)) %>%
  unnest(res_model2, res_model3, res_model4, res_model5) %>%
  dplyr::select(rmse_model2, rmse_model3, rmse_model4, rmse_model5,
    value.adj.r.squared, value.adj.r.squared1, value.adj.r.squared2, value.adj.r.squared3) %>%
  rename(adjR_model2 = value.adj.r.squared,
    adjR_model3 = value.adj.r.squared1,
    adjR_model4 = value.adj.r.squared2,
    adjR_model5 = value.adj.r.squared3)

```

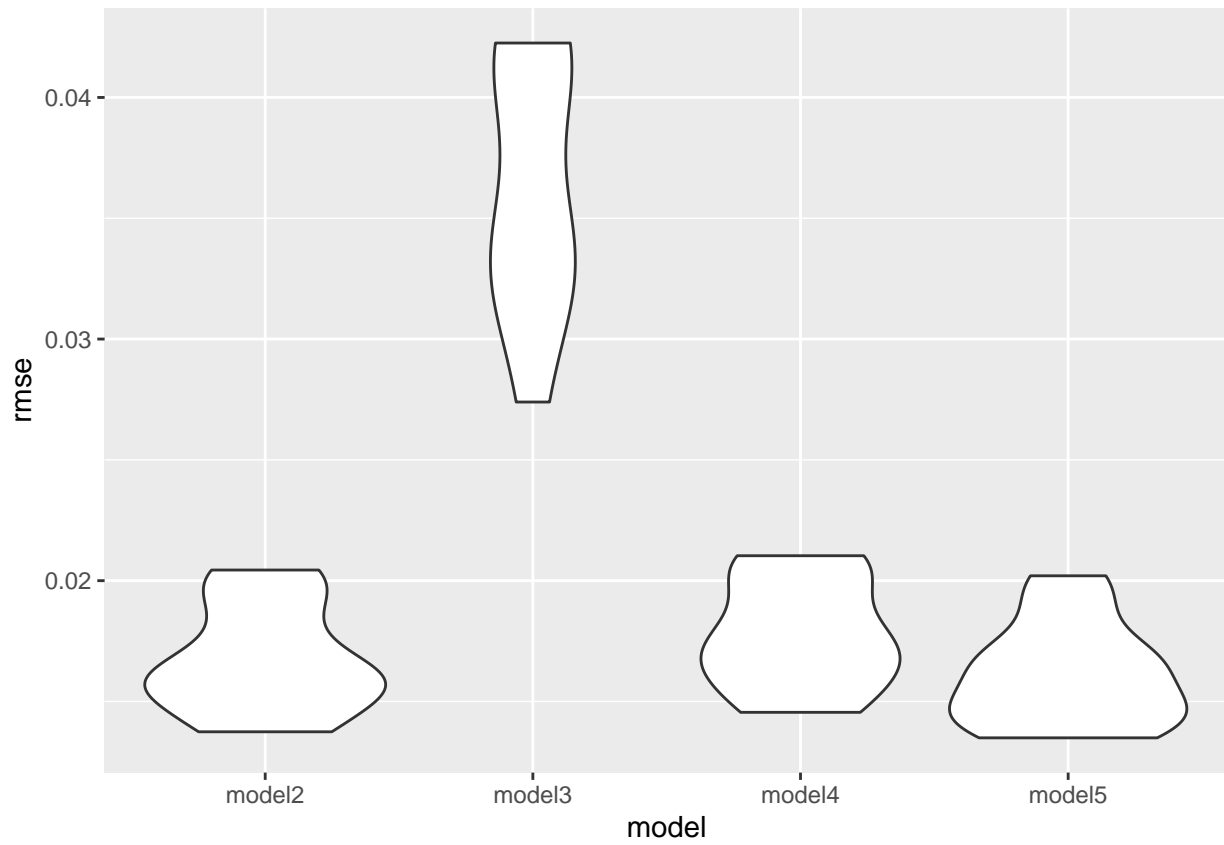
```
cv_df %>%
  summarise_each(funs(mean( .,na.rm = TRUE))) %>%
  t()
```

```
##           [,1]
## rmse_model2 0.01673048
## rmse_model3 0.03621843
## rmse_model4 0.01777271
## rmse_model5 0.01608495
## adjR_model2 0.55207433
## adjR_model3 0.53745725
## adjR_model4 0.49859335
## adjR_model5 0.59113867
```

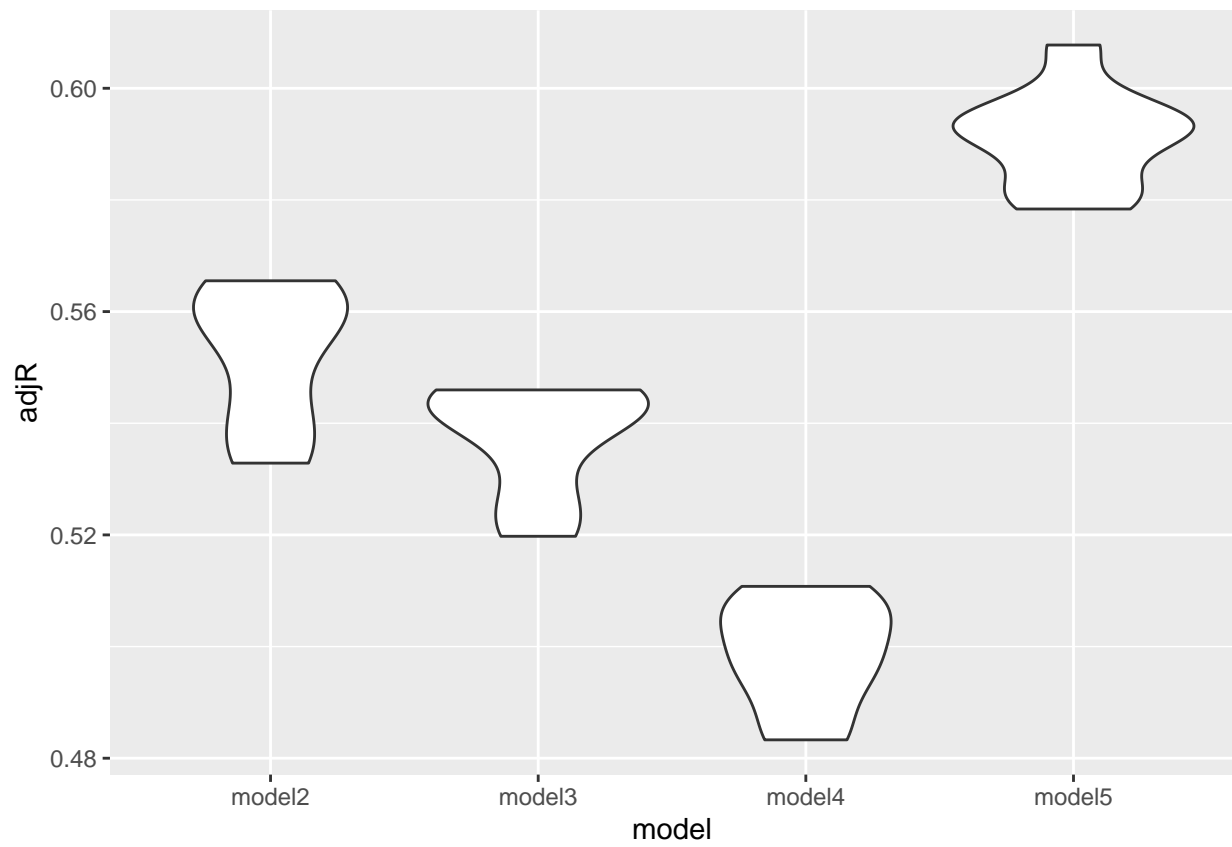
Plot the violin plot

```
unnest_cd_df = cv_df %>%
  pivot_longer(rmse_model2:adjR_model5,
    names_pattern = "(.*)"(.*)$",
    names_to = c("limit", "model")) %>%
  mutate(limit=ifelse(limit=="", "value", limit)) %>%
  pivot_wider(id_cols = model,
    names_from = limit,
    values_from = value,
    names_repair = "check_unique") %>%
  unnest(c(rmse_, adjR_)) %>%
  rename(rmse = rmse_,
    adjR = adjR_)

unnest_cd_df %>% ggplot(aes(x = model, y = rmse)) + geom_violin()
```



```
unnest_cd_df %>% ggplot(aes(x = model, y = adjR)) + geom_violin()
```



Thanks for reading!