Econ 573 Project Mateo

Juan M. Alvarez

Data Visualization

```
Gas_raw <- read.csv("C:/Users/mateo/RawData.csv", na.strings = "?", stringsAsFactors = T )
View(Gas_raw)
```

We import our raw data in order to perfoma Data Analysis.

```
dim(Gas_raw)
```

```
## [1] 100  35
```

Our data comes from 2020 North Carolina Data at the county level. Since there is 100 counties in north carolina, we have 100 osbservations.

```
head(Gas_raw)
```

	County <fct>	GAS <int>	AIW <int>	POP05 <int>	WIS <int>	UPOP <int>	RPOP <int>	MHV <int>	BACH <int>
1	Alamance	94	4	0	7	126085	45330	160900	32545
2	Alexander	19	0	0	5	3978	32466	138900	4067
3	Alleghany	13	0	0	5	0	10888	150500	1967
4	Anson	30	0	0	5	4903	17152	104100	1701
5	Ashe	25	0	0	5	0	26577	158200	4622
6	Avery	18	0	0	6	0	17806	151800	2954

6 rows | 1-10 of 36 columns

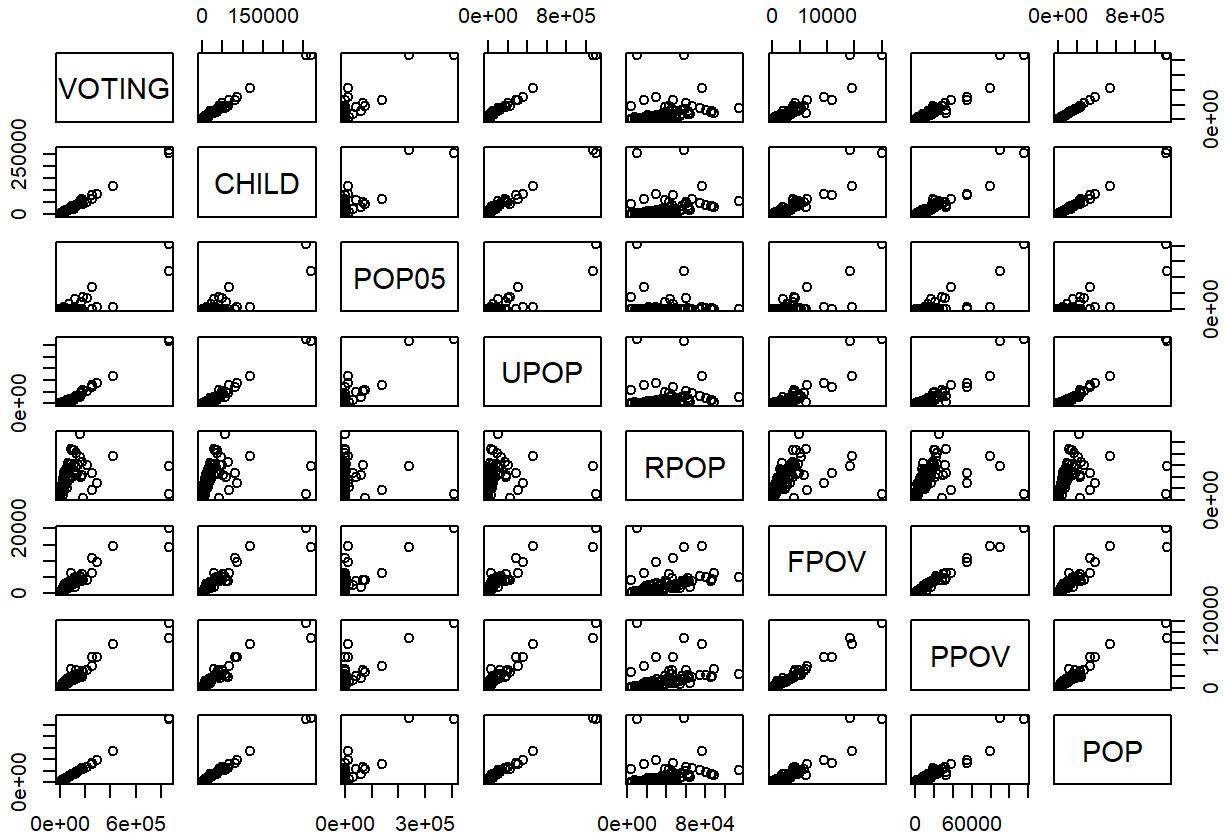
We can visualize the first 6 observations for our dataset

In our Data set, we have a lot of variables that measure similar things. For example, we have various measures that measure population based on different attributes such as Race, age groups, poverty levels, education, urban/rural, etc. We expect these variables to be heavily correlated to each other.

We also have various employnment measures such as the Average Annual Employment by Place of Work and teh Unemployment Rate. As well as various measures for Income such as Median Household Income, Annual Wages by Place of Work. We're interested to see if these variables capture different areas of gas station density at the county level in north Carolina.

For the purpose of our research, we don't expect all of these variables to play an important role in the regression of gas station as they all measure county-level population and follow the same trends. However, we're interest to see which of these measures is important.

```
pairs(
  ~ VOTING + CHILD + POP05 + UPOP + RPOP + FPOV + PPOV + POP, data = Gas_raw
)
```



We plot a few of these population measures and observe that indeed, a few of them seem to be correlated. Such as CHILD and UPOP, VOTING and UPOP, or the general measure POP with all of them.

Linear Regression

```
# Loading our regularized data
Gas0 <- read.csv("C:/Users/mateo/nc_gas_scaled.csv", na.strings = "?", stringsAsFactors = T)
# Getting rid of the column with county names.
Gas <- subset(Gas0, select = -county)
```

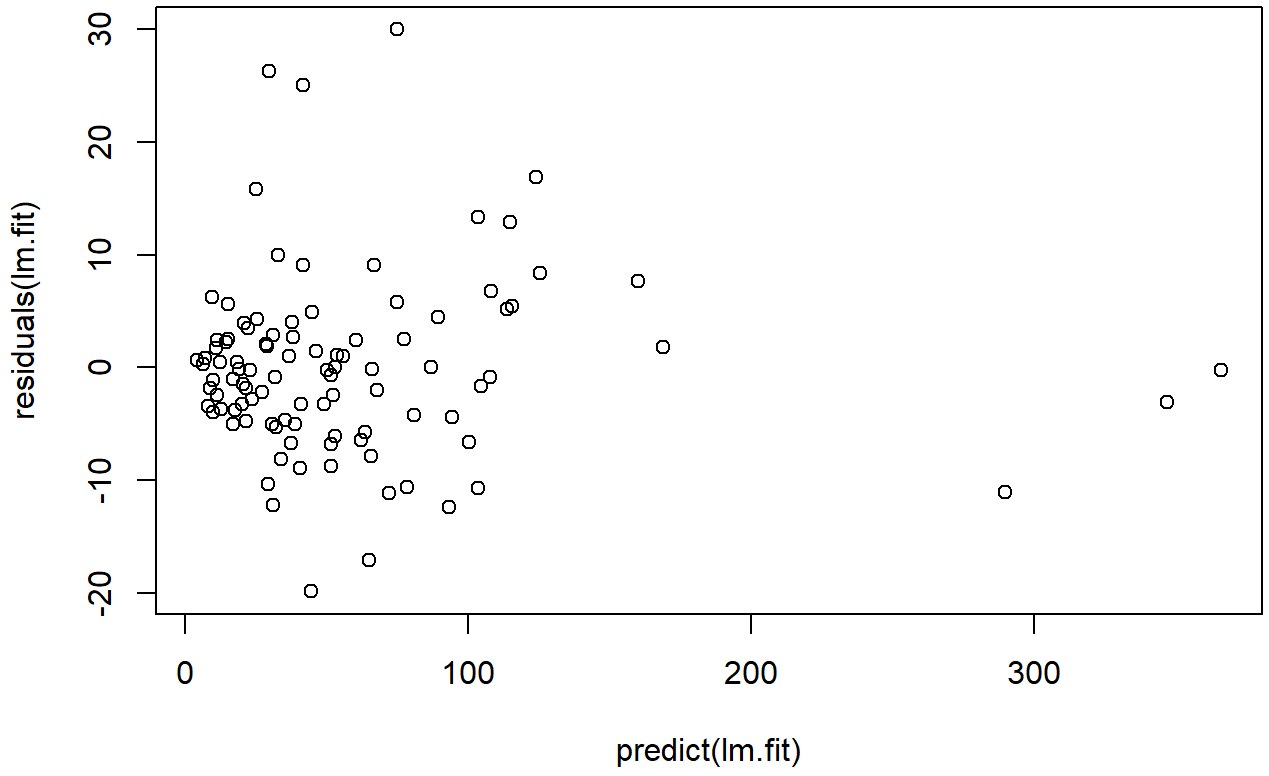
```
View(Gas)
```

```
# Performing the regression on the model with all the variables.
lm.fit <- lm(gas ~ ., data = Gas)
summary(lm.fit)
```

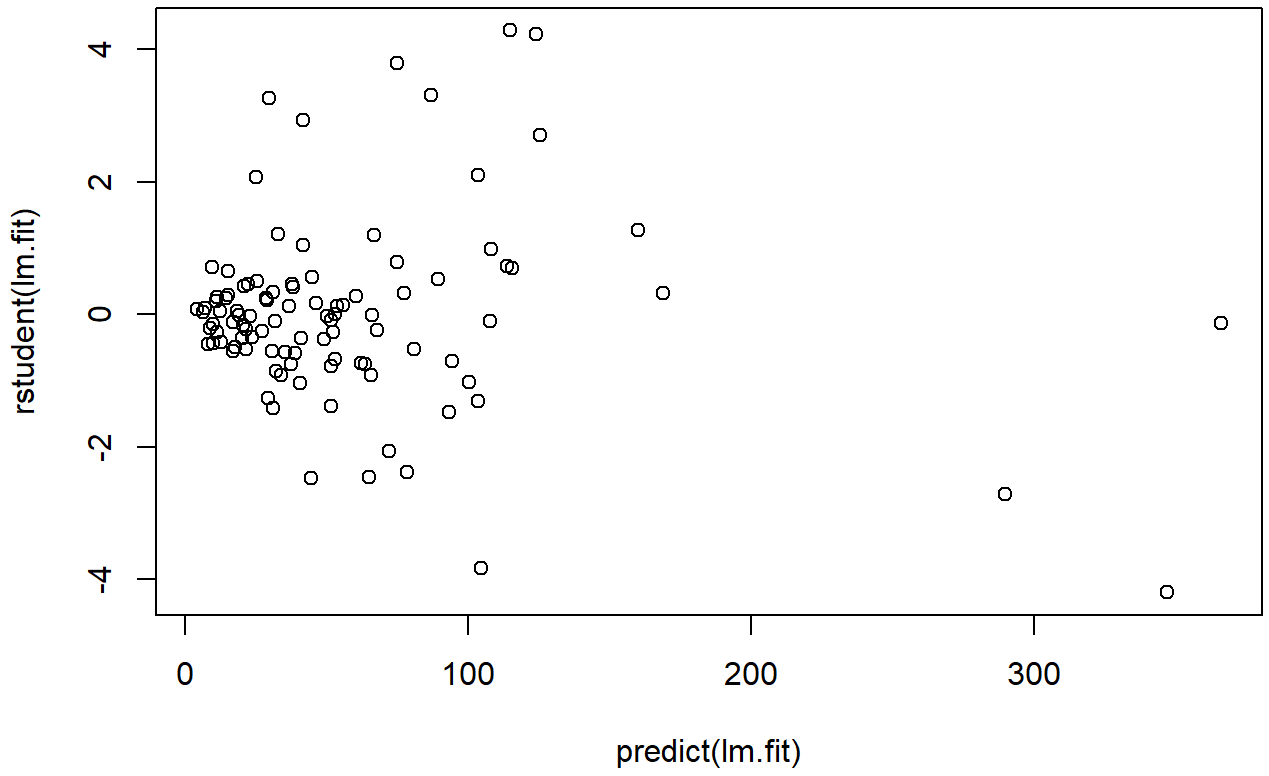
```
##
## Call:
## lm(formula = gas ~ ., data = Gas)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.8416  -4.4773  -0.2348   2.7590  29.9807
##
## Coefficients: (4 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  58.0400     0.9474  61.262  < 2e-16 ***
## aiw           2.7158     1.5000   1.811  0.07451 .
## pop05         4.8787     8.7005   0.561  0.57677
## wis          -0.1712     2.5309  -0.068  0.94625
## upop         65.9106    81.9717   0.804  0.42408
## rpop         11.6730    11.3547   1.028  0.30747
## mhv          -1.7081     3.2303  -0.529  0.59863
## bach        -40.6412    34.8072  -1.168  0.24693
## hs           -4.0282   403.0949  -0.010  0.99206
## x9gr          2.5650   403.6431   0.006  0.99495
## fpov         14.9589    14.1803   1.055  0.29510
## ppov         15.7997    17.2793   0.914  0.36366
## pop              NA         NA      NA       NA
## hisp        -29.4132    10.3618  -2.839  0.00593 **
## white       -27.5921    26.6857  -1.034  0.30471
## black       -14.1112    14.7768  -0.955  0.34288
## other            NA         NA      NA       NA
## voting      -30.4302    98.4659  -0.309  0.75821
## child            NA         NA      NA       NA
## mhi           3.5319     2.8892   1.222  0.22564
## awpw        -34.2697    30.6139  -1.119  0.26679
## aaepw        90.7672    35.4382   2.561  0.01259 *
## unemp        -0.8974     1.2462  -0.720  0.47383
## capin        -0.4730     3.0778  -0.154  0.87830
## crime        -5.1887     5.8696  -0.884  0.37973
## nomuns       -5.6478    12.3974  -0.456  0.65011
## munp         -1.9866     8.1027  -0.245  0.80703
## nmunp        -0.4395     3.0635  -0.143  0.88635
## shigh        11.8693    18.0174   0.659  0.51221
## noveh        -6.2639    16.7156  -0.375  0.70899
## commt        -0.5939     1.4204  -0.418  0.67715
## agmort        7.8572    34.6012   0.227  0.82102
## prpval           NA         NA      NA       NA
## retax        35.7601    28.7479   1.244  0.21768
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.474 on 70 degrees of freedom
## Multiple R-squared:  0.983,  Adjusted R-squared:  0.976
## F-statistic: 139.8 on 29 and 70 DF,  p-value: < 2.2e-16
```

pop, other, child, and prpval were dropped because they are linearly dependent on others. It seems that hips, aaepw, and aiw are the only statistically significant variables.

```
plot(predict(lm.fit), residuals(lm.fit))
```




```
plot(predict(lm.fit), rstudent(lm.fit))
```




```
# Identifying Problematic Variables; alias tells us which linear combinations from each other.

alias(lm(gas ~ ., data = Gas))
```

```
## Model :
## gas ~ aiw + pop05 + wis + upop + rpop + mhv + bach + hs + x9gr +
##     fpov + ppov + pop + hisp + white + black + other + voting +
##     child + mhi + awpw + aaepw + unemp + capin + crime + nomuns +
##     munp + nmunp + shigh + noveh + commt + agmort + prpval +
##     retax
##
## Complete :
##        (Intercept)   aiw           pop05         wis           upop
## pop                0             0             0             0   76391/78599
## other              0             0             0             0             0
## child              0             0             0             0    23037/5515
## prpval             0             0             0             0             0
##        rpop          mhv           bach          hs            x9gr
## pop         387/3062             0             0             0             0
## other              0             0             0             0             0
## child   77334/142367             0             0             0             0
## prpval             0             1             0             0             0
##        fpov          ppov          hisp          white         black
## pop                0             0             0             0             0
## other              0             0    3336/22097   10708/18069  45631/155876
## child              0             0             0             0             0
## prpval             0             0             0             0             0
##        voting        mhi           awpw          aaepw         unemp
## pop                0             0             0             0             0
## other              0             0             0             0             0
## child  -220756/66861             0             0             0             0
## prpval             0             0             0             0             0
##        capin         crime         nomuns        munp          nmunp
## pop                0             0             0             0             0
## other              0             0             0             0             0
## child              0             0             0             0             0
## prpval             0             0             0             0             0
##        shigh         noveh         commt         agmort        retax
## pop                0             0             0             0             0
## other              0             0             0             0             0
## child              0             0             0             0             0
## prpval             0             0             0             0             0
```

pop, child, other and prpval are causing aliasing, or multiple collinearity.

So, we create a new data set without these variables.

```
Gas <- Gas[, !(names(Gas) %in% c("pop", "child", "other", "prpval"))]
```

We perform a new linear regression on the new using the new data set.

```
lm.fit2 <- lm(gas ~ ., data = Gas)
summary(lm.fit2)
```

```
##
## Call:
## lm(formula = gas ~ ., data = Gas)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.8416  -4.4773  -0.2348   2.7590  29.9807
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  58.0400     0.9474  61.262  < 2e-16 ***
## aiw           2.7158     1.5000   1.811  0.07451 .
## pop05         4.8787     8.7005   0.561  0.57677
## wis          -0.1712     2.5309  -0.068  0.94625
## upop         65.9106    81.9717   0.804  0.42408
## rpop         11.6730    11.3547   1.028  0.30747
## mhv          -1.7081     3.2303  -0.529  0.59863
## bach        -40.6412    34.8072  -1.168  0.24693
## hs           -4.0282   403.0949  -0.010  0.99206
## x9gr          2.5650   403.6431   0.006  0.99495
## fpov         14.9589    14.1803   1.055  0.29510
## ppov         15.7997    17.2793   0.914  0.36366
## hisp        -29.4132    10.3618  -2.839  0.00593 **
## white       -27.5921    26.6857  -1.034  0.30471
## black       -14.1112    14.7768  -0.955  0.34288
## voting      -30.4302    98.4659  -0.309  0.75821
## mhi           3.5319     2.8892   1.222  0.22564
## awpw        -34.2697    30.6139  -1.119  0.26679
## aaepw        90.7672    35.4382   2.561  0.01259 *
## unemp        -0.8974     1.2462  -0.720  0.47383
## capin        -0.4730     3.0778  -0.154  0.87830
## crime        -5.1887     5.8696  -0.884  0.37973
## nomuns       -5.6478    12.3974  -0.456  0.65011
## munp         -1.9866     8.1027  -0.245  0.80703
## nmunp        -0.4395     3.0635  -0.143  0.88635
## shigh        11.8693    18.0174   0.659  0.51221
## noveh        -6.2639    16.7156  -0.375  0.70899
## commt        -0.5939     1.4204  -0.418  0.67715
## agmort        7.8572    34.6012   0.227  0.82102
## retax        35.7601    28.7479   1.244  0.21768
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.474 on 70 degrees of freedom
## Multiple R-squared:  0.983,  Adjusted R-squared:  0.976
## F-statistic: 139.8 on 29 and 70 DF,  p-value: < 2.2e-16
```

By removing, pop, child, other, and ppval we see that linear regression on our model has a higher adjusted Rsquared compared to the previous model (0.9766 >0.976), althouhg its not a lot its an improvement. We see that the variable for the hispanic population retains its statistic importance, while aaepw increase, and aiw gains some statistoc significant at the 0.05 level.

```
library(car)
```

```
## Warning: package 'car' was built under R version 4.4.3
```

```
## Cargando paquete requerido: carData
```

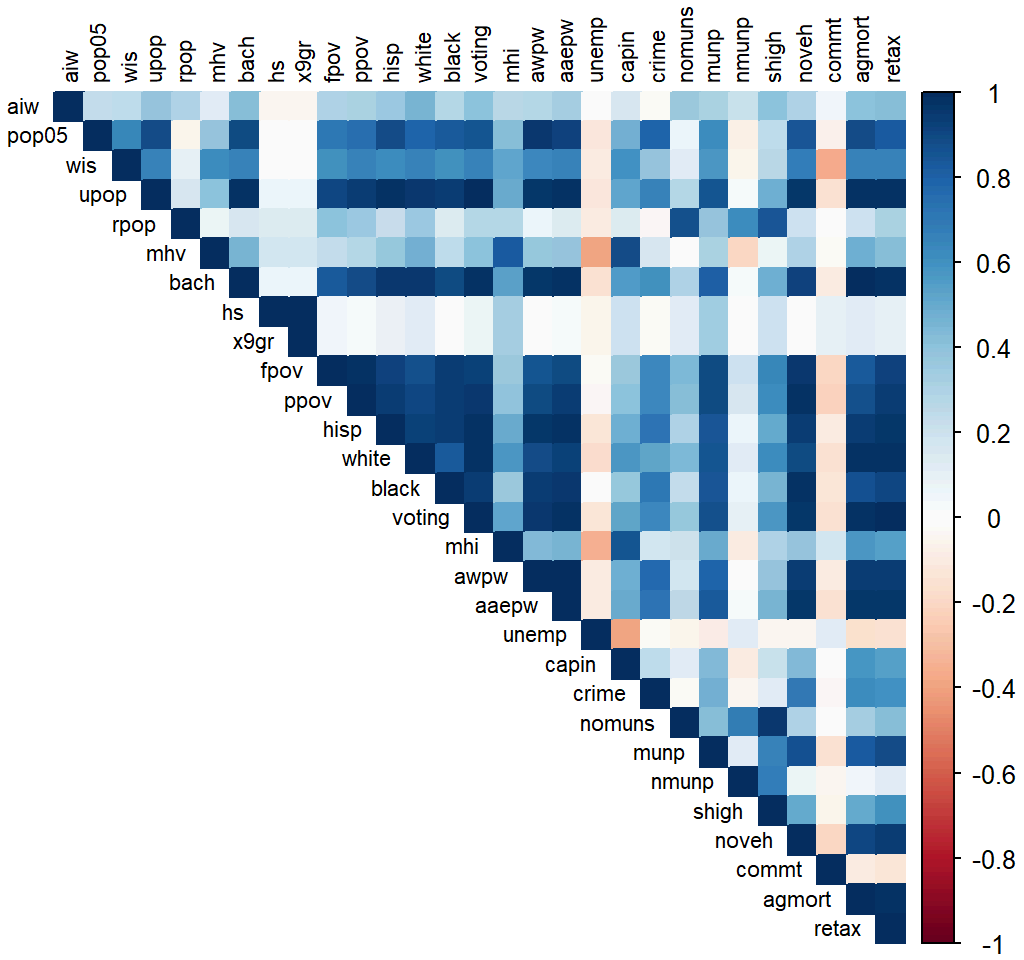
```
## Warning: package 'carData' was built under R version 4.4.3
```

```
# getting the predictors
predictors <- Gas[, !(names(Gas) %in% "gas")]
library(corrplot)
```

```
## Warning: package 'corrplot' was built under R version 4.4.3
```

```
## corrplot 0.95 loaded
```

```
# Computing the correlation matrix
cor_matrix <- cor(predictors, use = "pairwise.complete.obs")
corrplot(cor_matrix, method = "color", type = "upper", tl.cex = 0.7, tl.col = "black")
```



We see a lot of heavily correlated variables in our original model. We have a lot problematic variables and there is celarly a collinearity problem in the data. Howerver, we are not abel to get the full scope of the problem by simply inspecting the correlation matrix. There is most likely multicollinearity present in teh data.

We compute the variance infation factor for all variables to assess multicollinearity.

```
vif_values <- vif(lm.fit2)
sorted_vif <- sort(vif_values, decreasing = TRUE)
print(sorted_vif)
```

```
##         x9gr           hs       voting         upop        aaepw         bach
## 1.797039e+05 1.792161e+05 1.069384e+04 7.411239e+03 1.385176e+03 1.336288e+03
##       agmort         awpw        retax        white        shigh         ppov
## 1.320516e+03 1.033713e+03 9.115390e+02 7.854507e+02 3.580517e+02 3.293187e+02
##        noveh        black         fpov       nomuns         rpop         hisp
## 3.081810e+02 2.408368e+02 2.217862e+02 1.695205e+02 1.422042e+02 1.184227e+02
##        pop05         munp        crime          mhv        capin        nmunp
## 8.349367e+01 7.241329e+01 3.800012e+01 1.150917e+01 1.044804e+01 1.035152e+01
##          mhi          wis          aiw        commt        unemp
## 9.207053e+00 7.064934e+00 2.481827e+00 2.225317e+00 1.712858e+00
```

We observe a serious multicollinearity problem, as we a large amount of variables with values above 100, with some even reaching over 150,000.

We can further remove these variable from our mode and perform another linear regression.

```
# Start by removing highest-VIF variable (hs and x9gr)
Gas <- Gas[, !(names(Gas) %in% c("pop", "child", "other", "prpval", "hs", "x9gr"))]
names(Gas)
```

```
##  [1] "gas"    "aiw"    "pop05"  "wis"    "upop"   "rpop"   "mhv"    "bach"
##  [9] "fpov"   "ppov"   "hisp"   "white"  "black"  "voting" "mhi"    "awpw"
## [17] "aaepw"  "unemp"  "capin"  "crime"  "nomuns" "munp"   "nmunp"  "shigh"
## [25] "noveh"  "commt"  "agmort" "retax"
```

```
# Refit and recheck
lm.fit3 <- lm(gas ~ ., data = Gas)
summary(lm.fit3)
```

```
##
## Call:
## lm(formula = gas ~ ., data = Gas)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.6344  -4.1624  -0.6478   3.4353  30.3416
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  58.0400     0.9365  61.975  < 2e-16 ***
## aiw           2.9506     1.4267   2.068  0.04222 *
## pop05         5.2157     7.7899   0.670  0.50529
## wis          -0.1209     2.5000  -0.048  0.96156
## upop         53.2943    78.2491   0.681  0.49800
## rpop          9.8779    10.7960   0.915  0.36327
## mhv          -1.4814     3.1214  -0.475  0.63652
## bach        -32.3451    30.0922  -1.075  0.28602
## fpov         17.3808    13.3419   1.303  0.19682
## ppov         14.3211    16.8849   0.848  0.39916
## hisp        -31.2019     9.7776  -3.191  0.00210 **
## white       -25.6278    26.1731  -0.979  0.33078
## black       -12.3429    14.3068  -0.863  0.39115
## voting      -25.2068    96.9125  -0.260  0.79553
## mhi           3.6213     2.8478   1.272  0.20760
## awpw        -39.2214    27.2346  -1.440  0.15416
## aaepw       100.4196    29.3478   3.422  0.00103 **
## unemp        -0.9606     1.2248  -0.784  0.43541
## capin        -0.5523     2.9952  -0.184  0.85423
## crime        -4.1843     5.3423  -0.783  0.43605
## nomuns       -5.9397    12.1710  -0.488  0.62702
## munp         -4.5352     6.1337  -0.739  0.46208
## nmunp        -0.6178     2.9883  -0.207  0.83680
## shigh        12.4876    17.6742   0.707  0.48213
## noveh        -9.9526    15.0732  -0.660  0.51118
## commt        -0.6625     1.3873  -0.478  0.63439
## agmort       -4.9227    26.1768  -0.188  0.85136
## retax        44.8512    23.6087   1.900  0.06147 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.365 on 72 degrees of freedom
## Multiple R-squared:  0.9829, Adjusted R-squared:  0.9766
## F-statistic: 153.7 on 27 and 72 DF,  p-value: < 2.2e-16
```

we don’t get an improvement from our previous model.

```
vif_values2 <- vif(lm.fit3)
sorted_vif2 <- sort(vif_values2, decreasing = TRUE)
print(sorted_vif2)
```

```
##       voting         upop         bach        aaepw         awpw       agmort
## 10601.713761  6911.540692  1022.170014   972.227247   837.257746   773.477624
##        white        retax        shigh         ppov        noveh        black
##   773.264377   629.161101   352.610617   321.819511   256.464849   231.048923
##         fpov       nomuns         rpop         hisp        pop05         munp
##   200.932994   167.211787   131.566206   107.914022    68.497564    42.468382
##        crime          mhv        capin        nmunp          mhi          wis
##    32.215823    10.998087    10.126609    10.080257     9.154501     7.055160
##          aiw        commt        unemp
##     2.297717     2.172336     1.693212
```

we still have exteme multicollinear variables. We remove the ones with values over 1000.

```
Gas <- Gas[, !(names(Gas) %in% c("pop", "child", "other", "prpval", "hs", "x9gr", "voting", "upo
p", "bach"))]
names(Gas)
```

```
##  [1] "gas"    "aiw"    "pop05"  "wis"    "rpop"   "mhv"    "fpov"   "ppov"
##  [9] "hisp"   "white"  "black"  "mhi"    "awpw"   "aaepw"  "unemp"  "capin"
## [17] "crime"  "nomuns" "munp"   "nmunp"  "shigh"  "noveh"  "commt"  "agmort"
## [25] "retax"
```

```
lm.fit4 <- lm(gas ~ ., data = Gas)
summary(lm.fit4)
```

```
##
## Call:
## lm(formula = gas ~ ., data = Gas)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -20.6411  -4.0386  -0.3786   3.2950  30.2882
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  58.04000    0.92658  62.639  < 2e-16 ***
## aiw           2.67982    1.29639   2.067 0.042172 *
## pop05         3.64531    6.90314   0.528 0.599012
## wis           0.07178    2.35592   0.030 0.975774
## rpop          2.71514    3.16899   0.857 0.394294
## mhv          -2.06095    2.94427  -0.700 0.486102
## fpov         23.44221   11.26594   2.081 0.040866 *
## ppov          9.99353   15.70229   0.636 0.526429
## hisp        -27.59541    7.58115  -3.640 0.000499 ***
## white       -14.90295   13.67706  -1.090 0.279364
## black        -9.05541    7.97960  -1.135 0.260065
## mhi           4.57722    2.68649   1.704 0.092560 .
## awpw        -47.95101   25.78580  -1.860 0.066865 .
## aaepw       100.92870   27.84271   3.625 0.000524 ***
## unemp        -0.74493    1.19672  -0.622 0.535517
## capin        -0.68366    2.84811  -0.240 0.810953
## crime         0.08190    2.66240   0.031 0.975541
## nomuns       -7.59038   11.93134  -0.636 0.526601
## munp         -3.89078    5.98725  -0.650 0.517778
## nmunp        -1.22126    2.89374  -0.422 0.674206
## shigh        15.77012   17.04573   0.925 0.357847
## noveh       -10.26711   14.53400  -0.706 0.482116
## commt        -0.82657    1.34176  -0.616 0.539735
## agmort      -21.58405   15.13234  -1.426 0.157916
## retax        45.43167   22.23218   2.044 0.044515 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.266 on 75 degrees of freedom
## Multiple R-squared:  0.9826, Adjusted R-squared:  0.977
## F-statistic: 176.6 on 24 and 75 DF,  p-value: < 2.2e-16
```

We get a minor improvement with adjust rsquared, the same variables remaing significant.

```
library(car)
vif_values3 <- vif(lm.fit4)
sorted_vif3 <- sort(vif_values3, decreasing = TRUE)
print(sorted_vif3)
```

```
##      aaepw       awpw      retax      shigh       ppov     agmort      noveh
## 893.910335 766.711638 569.947907 335.044174 284.313175 264.048197 243.579865
##      white     nomuns       fpov      black       hisp      pop05       munp
## 215.703076 164.153170 146.354302  73.423252  66.273744  54.949533  41.335692
##       rpop        mhv      nmunp      capin        mhi      crime        wis
##  11.580144   9.996033   9.655815   9.353692   8.322233   8.173659   6.400192
##      commt        aiw      unemp
##   2.075955   1.937956   1.651410
```

There are still some variables with very high IVF values, however hisp and aaepw appear to be statistifclaly significant such taht the explain a lot of the variance in the regression, so we'll keep them. We remove all variables with over 50 VIF that are not statistifcaly significant.

```
Gas <- Gas[, !(names(Gas) %in% c("pop", "child", "other", "prpval", "hs", "x9gr", "voting", "upo
p", "bach", "awpw", "retax", "shigh", "ppov", "agmort", "noveh", "white", "nomuns", "fpov", "bla
ck"))]
names(Gas)
```

```
##  [1] "gas"   "aiw"   "pop05" "wis"   "rpop"  "mhv"   "hisp"  "mhi"   "aaepw"
## [10] "unemp" "capin" "crime" "munp"  "nmunp" "commt"
```

```
lm.fit5 <- lm(gas ~ ., data = Gas)
summary(lm.fit5)
```

```
##
## Call:
## lm(formula = gas ~ ., data = Gas)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.693  -6.533  -2.319   5.148  35.941
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  58.0400     1.1747  49.408  < 2e-16 ***
## aiw           1.7155     1.4841   1.156  0.25096
## pop05       -26.7563     5.0975  -5.249 1.11e-06 ***
## wis           7.3210     2.3174   3.159  0.00219 **
## rpop         13.2547     2.1048   6.297 1.28e-08 ***
## mhv          -6.0735     3.0668  -1.980  0.05090 .
## hisp        -14.4570     7.1026  -2.035  0.04492 *
## mhi           1.1314     3.1864   0.355  0.72341
## aaepw        82.0787     8.6084   9.535 4.47e-15 ***
## unemp        -0.5164     1.4353  -0.360  0.71989
## capin         0.9802     3.2134   0.305  0.76108
## crime         0.8853     2.4237   0.365  0.71581
## munp          6.1173     3.2055   1.908  0.05972 .
## nmunp         2.0205     1.7287   1.169  0.24574
## commt        -0.5803     1.6004  -0.363  0.71782
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.75 on 85 degrees of freedom
## Multiple R-squared:  0.9683, Adjusted R-squared:  0.9631
## F-statistic: 185.6 on 14 and 85 DF,  p-value: < 2.2e-16
```

We see that rpop, black, munp and crime gain statistical signficance; however, our Adjuste R-squared greatly decreased.

```
vif_values5 <- vif(lm.fit5)
sorted_vif5 <- sort(vif_values5, decreasing = TRUE)
print(sorted_vif5)
```

```
##     aaepw      hisp     pop05     capin      munp       mhi       mhv     crime
## 53.164668 36.192016 18.641990  7.408171  7.371593  7.284018  6.747806  4.214389
##       wis      rpop     nmunp     commt       aiw     unemp
##  3.852942  3.178455  2.143879  1.837599  1.580173  1.478059
```

The VIF values for the variables that are left was greatly decreases. There is still mdoerate collinearity (>5) present in the model, we proceed to remove those that aren't statistically significant to compare.

```
Gas <- Gas[, !(names(Gas) %in% c("pop", "child", "other", "prpval", "hs", "x9gr", "voting", "upo
p", "bach", "aaepw", "awpw", "retax", "shigh", "ppov", "agmort", "noveh", "white", "nomuns", "fp
ov", "black", "capin", "munp", "mhi", "mhv"))]
names(Gas)
```

```
##  [1] "gas"   "aiw"   "pop05" "wis"   "rpop"  "hisp"  "unemp" "crime" "nmunp"
## [10] "commt"
```

and perform linear regression on the new model.

```
lm.fit6 <- lm(gas ~ ., data = Gas)
summary(lm.fit6)
```

```
##
## Call:
## lm(formula = gas ~ ., data = Gas)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -63.101 -10.498  -0.980   9.137  75.687
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  58.0400     1.9717  29.437  < 2e-16 ***
## aiw          -0.5476     2.4350  -0.225 0.822575
## pop05        -7.7727     6.6187  -1.174 0.243347
## wis           5.7788     3.1659   1.825 0.071265 .
## rpop         11.6654     3.3322   3.501 0.000724 ***
## hisp         60.5176     5.7386  10.546  < 2e-16 ***
## unemp         3.7447     2.1296   1.758 0.082078 .
## crime        -6.8142     3.8431  -1.773 0.079596 .
## nmunp         2.5746     2.7365   0.941 0.349296
## commt        -5.7766     2.2973  -2.514 0.013697 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19.72 on 90 degrees of freedom
## Multiple R-squared:  0.9055, Adjusted R-squared:  0.8961
## F-statistic: 95.84 on 9 and 90 DF,  p-value: < 2.2e-16
```

munp, rpop, and hisp retain theri statistical signficance, wis and capin gain some, and crime loses its.

```
vif_values6 <- vif(lm.fit6)
sorted_vif6 <- sort(vif_values6, decreasing = TRUE)
print(sorted_vif6)
```

```
##     pop05      hisp     crime      rpop       wis     nmunp       aiw     commt
## 11.156196  8.386587  3.761301  2.827635  2.552437  1.906993  1.509955  1.344070
##     unemp
##  1.154948
```

Pop05 loses it's statistical significance and we can see has a problematic VIF value, so we remove it.

```
Gas <- Gas[, !(names(Gas) %in% c("pop", "child", "other", "prpval", "hs", "x9gr", "voting", "upo
p", "bach", "aaepw", "awpw", "retax", "shigh", "ppov", "agmort", "noveh", "white", "nomuns", "fp
ov", "black", "capin", "munp", "mhi", "mhv", "pop05"))]
names(Gas)
```

```
## [1] "gas"   "aiw"   "wis"   "rpop"  "hisp"  "unemp" "crime" "nmunp" "commt"
```

```
lm.fit7 <- lm(gas ~ ., data = Gas)
summary(lm.fit7)
```

```
##
## Call:
## lm(formula = gas ~ ., data = Gas)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -61.704 -10.537  -1.007   8.544  81.688
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  58.0400     1.9758  29.376  < 2e-16 ***
## aiw          -0.8755     2.4240  -0.361  0.71880
## wis           4.1392     2.8473   1.454  0.14946
## rpop         13.8612     2.7638   5.015  2.6e-06 ***
## hisp         55.8780     4.1708  13.397  < 2e-16 ***
## unemp         4.2186     2.0953   2.013  0.04704 *
## crime        -8.9912     3.3735  -2.665  0.00910 **
## nmunp         1.8676     2.6750   0.698  0.48684
## commt        -6.4338     2.2328  -2.882  0.00494 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19.76 on 91 degrees of freedom
## Multiple R-squared:  0.9041, Adjusted R-squared:  0.8956
## F-statistic: 107.2 on 8 and 91 DF,  p-value: < 2.2e-16
```

```
vif_values7 <- vif(lm.fit7)
sorted_vif7 <- sort(vif_values7, decreasing = TRUE)
print(sorted_vif7)
```

```
##     hisp    crime      wis     rpop    nmunp      aiw    commt    unemp
## 4.411743 2.886149 2.056005 1.937270 1.814698 1.490103 1.264328 1.113472
```

Although we end up with a model thas has a lower Adjusted R-Squared, we now have fixed the issue of having highly correlated variabels. Most of the variables that are left in the model seem to be statistically significant. Out of the 35 predictors we started with, we reduced the model to only having 8 based on their IVF to fixe the multicollinearity present in our model. However, we can try other variable selection methods to compare the values.

```
library(boot)
```

```
## Warning: package 'boot' was built under R version 4.4.3
```

```
##
## Adjuntando el paquete: 'boot'
```

```
## The following object is masked from 'package:car':
##
##     logit
```

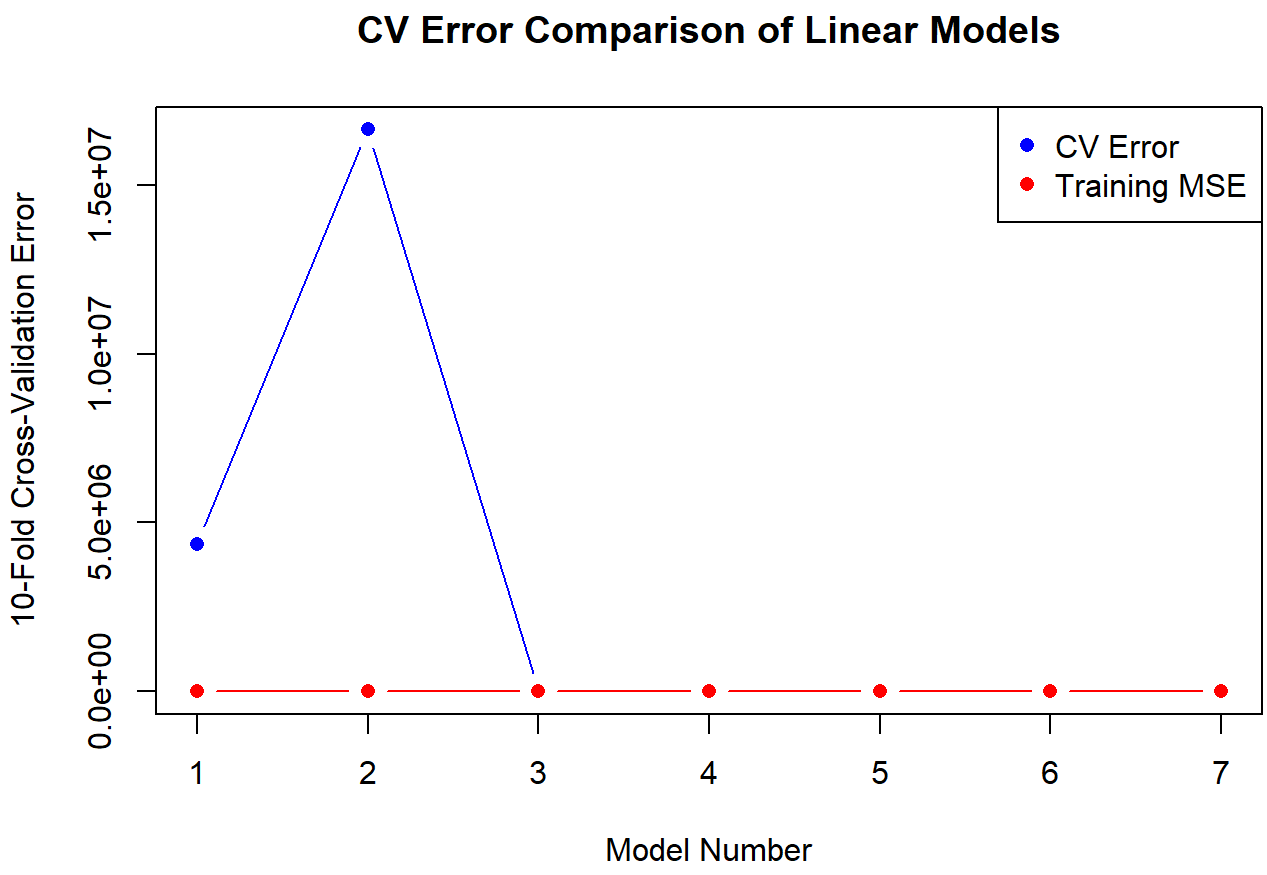
```
models <- list(lm.fit, lm.fit2, lm.fit3, lm.fit4, lm.fit5, lm.fit6, lm.fit7)

cv_errors <- sapply(models, function(model) {
  model_data <- model.frame(model)  # exact data used in that model
  cv.glm(model_data, glm(formula(model), data = model_data), K=10)$delta[1]
})

cv_errors
```

```
## [1] 4.363434e+06 1.664326e+07 1.594950e+03 5.290124e+02 2.753236e+03
## [6] 1.827180e+03 2.221505e+03
```

```
train_errors <- sapply(models, function(model) {
  mean(model$residuals^2)
})

# Create a comparison table
results <- data.frame(
  Model = paste0("lm.fit", c("", 2:7)),
  CV_Error = cv_errors,
  Train_MSE = train_errors
)

print(results)
```

```
##     Model     CV_Error Train_MSE
## 1  lm.fit 4.363434e+06  62.83054
## 2 lm.fit2 1.664326e+07  62.83054
## 3 lm.fit3 1.594950e+03  63.14683
## 4 lm.fit4 5.290124e+02  64.39105
## 5 lm.fit5 2.753236e+03 117.29397
## 6 lm.fit6 1.827180e+03 349.86798
## 7 lm.fit7 2.221505e+03 355.22926
```

```
plot(1:7, cv_errors, type="b", col="blue", pch=16,
     xlab="Model Number", ylab="10-Fold Cross-Validation Error",
     main="CV Error Comparison of Linear Models")

lines(1:7, train_errors, type="b", col="red", pch=16)
legend("topright", legend=c("CV Error", "Training MSE"), col=c("blue", "red"), pch=16)
```



While

lm.fit initially achieved the highest in-sample adjusted R-squared, its cross-validation performance was extremely poor due to severe multicollinearity. As variables with high VIF values were iteratively removed, model performance improved drastically. The fourth iteration (lm.fit4) achieved the lowest 10-fold cross-validated error, suggesting that it balances explanatory power and model stability most effectively. Further reductions in variables beyond this point slightly increased prediction error, indicating potential underfitting.