Regression Project QMB-6304 Analytical Methods for Business

Srikrishna Krishnarao Srinivasan

March 31, 2019

Preprocessing

1. Load the file “6304 Regression Project Data.xlsx”.

It contains population and other factors of 437 counties in American Midewest.

```
rm(list=ls())
library(readxl)
library(knitr)
library(car)
```

```
## Warning: package 'car' was built under R version 3.5.3
```

```
## Loading required package: carData
```

```
setwd("/Analytics for Business/Rmarkdown")

pop.mw=read_excel("6304 Regression Project Data.xlsx", sheet="midwest",
skip=0)
colnames(pop.mw)=tolower(make.names(colnames(pop.mw)))
attach(pop.mw)
sprintf('Number of rows in dataframe %s is %s', 'pop.mw', nrow(pop.mw))
```

```
## [1] "Number of rows in dataframe pop.mw is 437"
```

```
#head(pop.mw)
```

2. Calculate new variables ‘popcollege’ and ‘popprof’ and add to dataframe. These contain population by county with college degree and professional job.

```
pop.mw$popcollege=pop.mw$poptotal*(pop.mw$percollege/100)
pop.mw$popprof=pop.mw$poptotal*(pop.mw$perprof/100)
#head(pop.mw)
pop.mw$popcollegelog=log(pop.mw$popcollege)
attach(pop.mw)
```

```
## The following objects are masked from pop.mw (pos = 3):
##
##     area, county, id, inmetro, perchildpoverty, percollege,
##     perelderlypoverty, perprof, popadult, popasian, popblack,
##     popchild, popdensity, poptotal, popwhite, state
```

3. Calculate a new variable for ratio of ‘popchild’ to ‘popadult’ and add to dataframe.

```
pop.mw$popca=pop.mw$popchild/pop.mw$popadult
#head(pop.mw)
```

4. Calculate a new variable for number of child living in poverty and add to dataframe.

```
pop.mw$popchpov=pop.mw$popchild*(pop.mw$perchildpoverty/100)
#head(pop.mw)
#pop.mw$popadpov=pop.mw$popadult*(pop.mw$perelderlypoverty/100)
```

5. Create two dataframes which contain only rural and metropolitan counties using ‘inmetro’ variable

```
pop.mw.r=subset(pop.mw, pop.mw$inmetro==0)
pop.mw.m=subset(pop.mw,pop.mw$inmetro==1)
if (nrow(pop.mw)==nrow(pop.mw.r)+nrow(pop.mw.m)) {
  print('Sum of Rural and Metro dataframe equals Total Population dataframe')
} else {
  print('Sum of Rural and Metro does NOT equal Total Population')
}

## [1] "Sum of Rural and Metro dataframe equals Total Population dataframe"

#pop.mw.r1=pop.mw[pop.mw$inmetro==0,]
#pop.mw.r2=subset(pop.mw, pop.mw$inmetro==0)
#identical(pop.mw.r1, pop.mw.r2)
#head(pop.mw.r1)
#nrow(pop.mw.r1)
```

6. Take a random sample of 60 counties from rural poverty data set

```
set.seed(26913083)
reduced.pop.mw.r=pop.mw.r[sample(1:nrow(pop.mw.r), 60, replace = FALSE),]
sprintf('Number of rows in rural population sample is %s',
nrow(reduced.pop.mw.r))

## [1] "Number of rows in rural population sample is 60"
```

7. Take a random sample of 30 counties from metro poverty data set

```
set.seed(26913083)
reduced.pop.mw.m=pop.mw.m[sample(1:nrow(pop.mw.m),30, replace = FALSE),]
sprintf('Number of rows in metro population sample is %s',
nrow(reduced.pop.mw.m))

## [1] "Number of rows in metro population sample is 30"
```

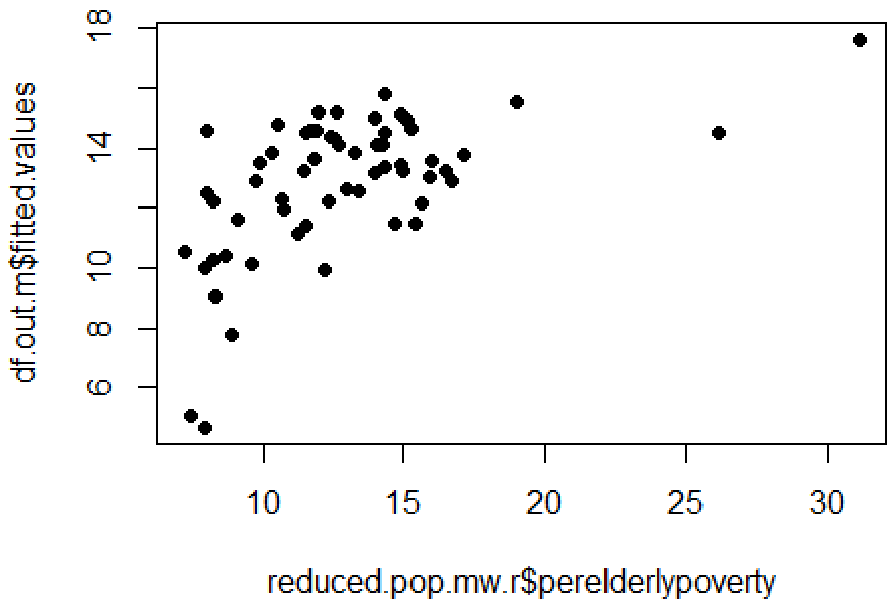
Analysis

1. Parameterize the best possible fit multiple regression model using ‘perelderlypoverty’ as dependent variable ‘some.rural.poverty’ dataframe. a). apply only main effects variables b). apply any or all numerical variables c). apply any data transformations to improve the fit d). show results of best fit model using summary(df.out) command e). describe the methodology used to arrive at selection of independent variables

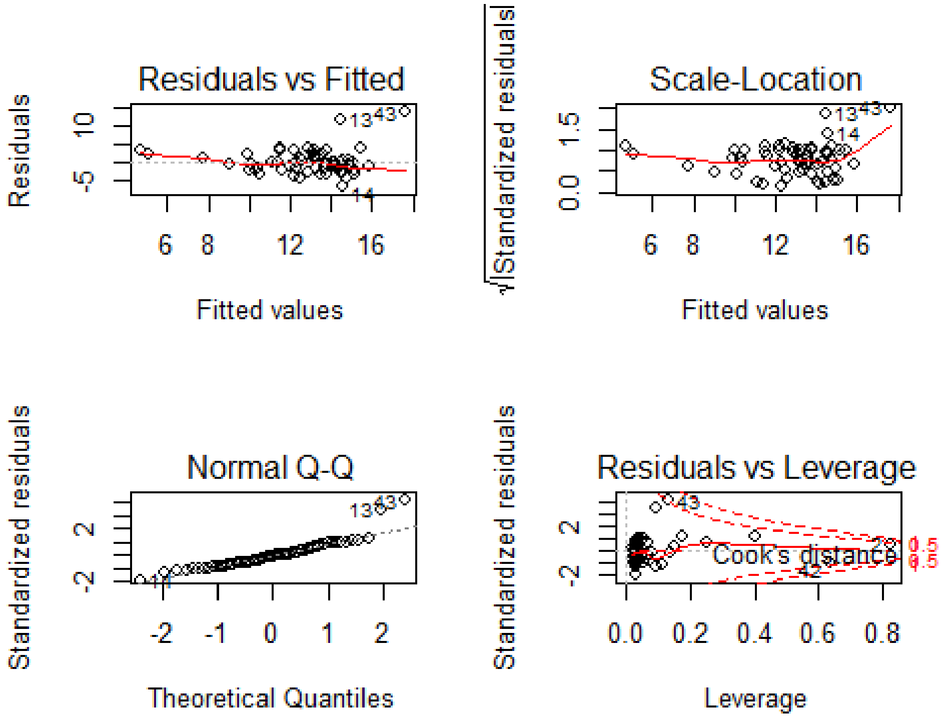
P-vales and R-squared values for lm(perelderlypoverty~independentvariables)

independent.var	pval	rsqval	interpretation
Area	0.9415680	0.0000934	High p-value=low correlation between dependent and independent var; low R-squared value=reject independent var
Poptotal	0.0003953	0.1961083	Low p-value=high correlation, high R-squared value=select independent var
popdensity	0.0005055	0.1896698	Low p-value=high correlation, high R-squared value=select independent var
popwhite	0.0002315	0.2099925	Low p-value=high correlation, high R-squared value=select independent var
popblack	0.7869210	0.0012699	High p-value=low correlation, low R-squared value=reject independent var
popasian	0.0033916	0.1386572	Low p-value=high correlation, high R-squared value=select independent var
popcollege	0.0002047	0.2131568	Low p-value=high correlation, high R-squared value=select independent var
Popprof	0.0020075	0.1528975	Low p-value=high correlation, high R-squared value=select independent var
popadult	0.0003714	0.1977392	Low p-value=high correlation, high R-squared value=select independent var

```
df.out.m=lm(perelderlypoverty~popcollege+popwhite+popadult+poptotal,data = reduced.pop.mw.r)
summary(df.out.m)

##
## Call:
## lm(formula = perelderlypoverty ~ popcollege + popwhite + popadult +
##     poptotal, data = reduced.pop.mw.r)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.562 -2.084 -0.196  1.519 13.537
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 15.8845195  0.8572814  18.529  < 2e-16 ***
## popcollege  -0.0007958  0.0004227  -1.883  0.06502 .
## popwhite    -0.0010894  0.0003996  -2.726  0.00857 **
## popadult    -0.0008625  0.0004587  -1.880  0.06537 .
## poptotal     0.0016265  0.0005651   2.878  0.00569 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.514 on 55 degrees of freedom
## Multiple R-squared:  0.3225, Adjusted R-squared:  0.2732
## F-statistic: 6.545 on 4 and 55 DF,  p-value: 0.000221

plot(reduced.pop.mw.r$perelderlypoverty, df.out.m$fitted.values, pch=19)
```

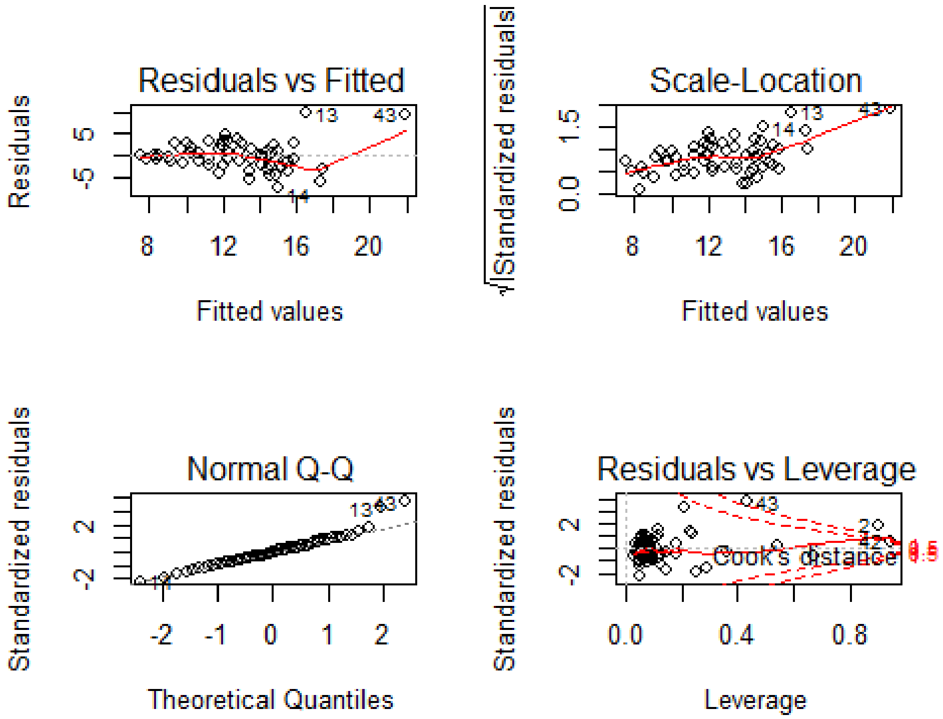


```
#abline(0,1,lwd=3, col='red')
#title('Actual perelderlypoverty vs fitted values')
layout(matrix(c(1,2,3,4), 2, 2))
plot(df.out.m)
```



```
df.out.msq=lm(perelderlypoverty~popcollege+I(popcollege^2)+popwhite+I(popwhit
e^2)+popadult+I(popadult^2)+poptotal+I(poptotal^2),data = reduced.pop.mw.r)
summary(df.out.msq)

##
## Call:
## lm(formula = perelderlypoverty ~ popcollege + I(popcollege^2) +
##     popwhite + I(popwhite^2) + popadult + I(popadult^2) + poptotal +
##     I(poptotal^2), data = reduced.pop.mw.r)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -7.009 -2.039  0.072  1.726  9.605
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      1.642e+01  1.365e+00  12.029  < 2e-16 ***
## popcollege      -3.435e-03  1.131e-03  -3.038  0.00375 **
## I(popcollege^2)  1.278e-07  5.320e-08   2.403  0.01994 *
## popwhite        -3.050e-03  9.251e-04  -3.297  0.00178 **
## I(popwhite^2)    1.802e-08  9.204e-09   1.958  0.05569 .
## popadult        -1.051e-03  2.113e-03  -0.497  0.62101
## I(popadult^2)    9.792e-09  3.682e-08   0.266  0.79136
## poptotal         3.998e-03  1.765e-03   2.265  0.02779 *
## I(poptotal^2)   -2.311e-08  2.072e-08  -1.116  0.26977
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 3.321 on 51 degrees of freedom
## Multiple R-squared:  0.439,  Adjusted R-squared:  0.351
## F-statistic: 4.989 on 8 and 51 DF,  p-value: 0.0001345

#abline(0,1, lwd=3, col='blue')
#title('Actual perelderlypoverty vs fitted values')

layout(matrix(c(1,2,3,4), 2, 2))
plot(df.out.msq)
```



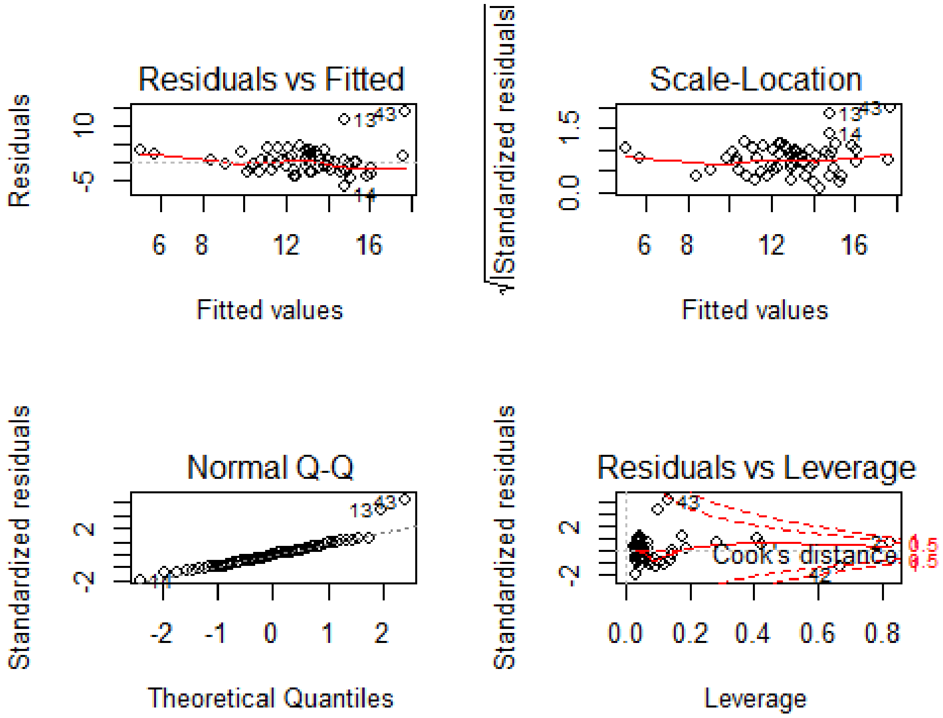
- a). apply only main effects variables: Main effect variables are chosen and applied as popcollege, popwhite, popadult and poptotal
- b). apply any or all numerical variables: Selected numerical variables as above were applied
- c). apply any data transformations to improve the fit: square terms of popcollege, popwhite, popadult and poptotal were used
- d). show results of best fit model using summary(df.out) command: summary is shown
- e). describe the methodology used to arrive at selection of independent variables: Correlation between perelderlypoverty vs each of the variables was verified/dislayed in a table above and retained only the highly correlating variables based on p-value and R-Squared value.
- Based on the Diagnositc plots displayed above, it is clear that there is no significant improvement due to usage of squared terms.

- Based on the same Diagnostic plot, it is clear that the fit is linear and normal (follows L and N in LINE) and not Equivalent distributon of points above and below the zero slope mean line
- Multiple-R-squared is approximately 32% which is not very good, but still reliable for a fit
- p-value of popcollege and popadult are more than 5%, the Beta coefficents for these cannot be accepted
- There is no heteroscadasticity

```
df.out.m.log=lm(perelderlypoverty~popcollege+popcollegelog+popwhite+popadult+
poptotal,data = reduced.pop.mw.r)
summary(df.out.m.log)
```

```
##
## Call:
## lm(formula = perelderlypoverty ~ popcollege + popcollegelog +
##     popwhite + popadult + poptotal, data = reduced.pop.mw.r)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.7532 -2.2939 -0.1014  1.3973 13.4679
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   25.1407537  9.4962956   2.647   0.0106 *
## popcollege    -0.0007042  0.0004331  -1.626   0.1098
## popcollegelog -1.3162181  1.3448364  -0.979   0.3321
## popwhite      -0.0009293  0.0004319  -2.151   0.0359 *
## popadult      -0.0008613  0.0004589  -1.877   0.0659 .
## poptotal       0.0015065  0.0005785   2.604   0.0119 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.516 on 54 degrees of freedom
## Multiple R-squared:  0.3343, Adjusted R-squared:  0.2727
## F-statistic: 5.423 on 5 and 54 DF,  p-value: 0.0004102
```

```
layout(matrix(c(1,2,3,4),2,2))
plot(df.out.m.log)
```



1. Using the Diagnostic 4x4 plots above, following are observed

- a). By using log of popcollege, the p-values are highly acceptable and Multiple R-Squared remains at 33%
- b). Linearity is maintained, Normality is also maintained
- c). This model with log term is even more preferable than the model with square terms
- d). Equality is not present in the model

```
## [1] ".........."
```

2. Best fit model's LINE assumptions are expalined above with the help of 4x4 Diagnostic plot.

```
## [1] ".........."
```

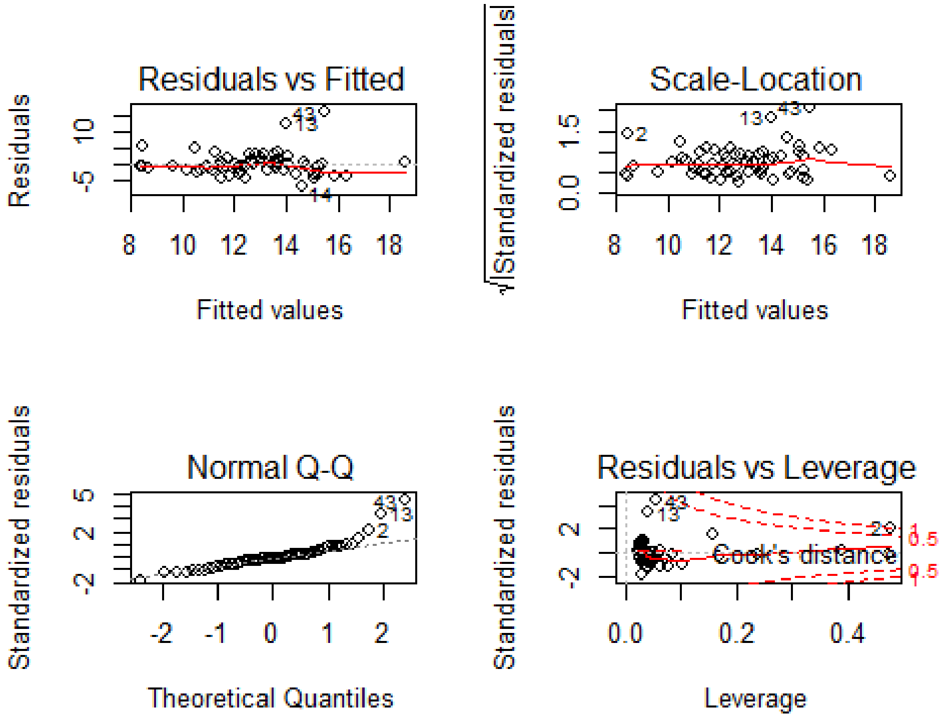
3. Multi-collinearity test is conducted by the vif(regression model command)

```
vif(df.out.m.log)

##    popcollege popcollegelog      popwhite      popadult      poptotal
##     14.369245      6.246412    334.209110    167.018859    670.577123
```

- a). As per the vif() command test for the best fit model following are observed:
  - i). popcollege has a value of 14
  - ii). popcollegelog has a value of 6
  - iii). popwhite has a value of 334
  - iv). popadult has a value of 167 and
  - v). poptotal has a value of 670

- b). a value of 2 to 5 is ok and it will establish low multi-collinearity
  - i). accordingly popcollegelog is not influenced by other variables
  - ii). all the other variables have very high multi-collinearity
  - iii). to address this, the variables with high values (popwhite and poptotal) can be dropped to check further

```
df.out.m.log1=lm(perelderlypoverty~popcollege+popcollegelog+popadult,data =
reduced.pop.mw.r)
summary(df.out.m.log1)

##
## Call:
## lm(formula = perelderlypoverty ~ popcollege + popcollegelog +
##     popadult, data = reduced.pop.mw.r)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.5692 -1.9905 -0.5635  0.9120 15.6852
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.003e+01  9.123e+00   3.291  0.00173 **
## popcollege    -1.483e-04  3.410e-04  -0.435  0.66531
## popcollegelog -2.074e+00  1.274e+00  -1.628  0.10919
## popadult       2.085e-05  1.045e-04   0.199  0.84264
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.663 on 56 degrees of freedom
## Multiple R-squared:  0.2507, Adjusted R-squared:  0.2105
## F-statistic: 6.245 on 3 and 56 DF,  p-value: 0.0009838

layout(matrix(c(1,2,3,4),2,2))
plot(df.out.m.log1)
```

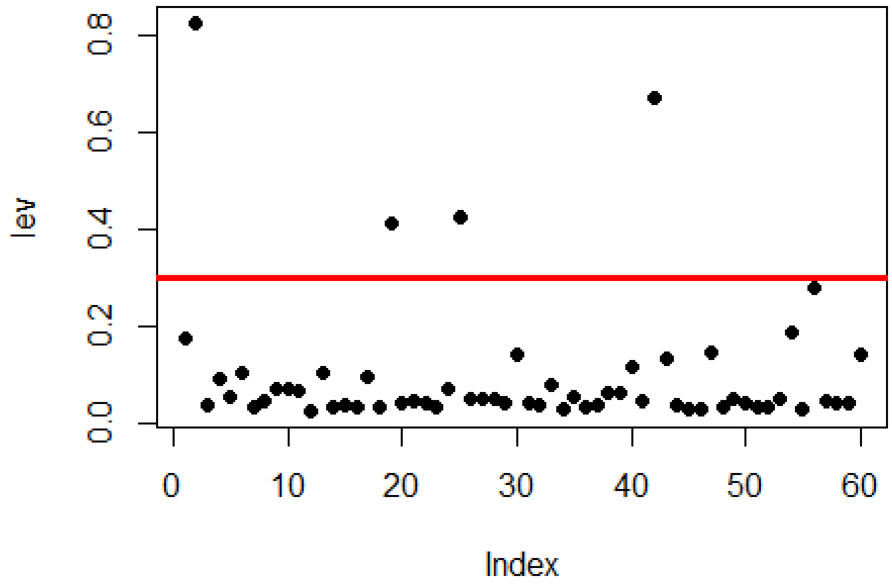


```
vif(df.out.m.log1)

##    popcollege popcollegelog      popadult
##      8.208856      5.166942      7.983590
```

- a). In the model after dropping popadult and poptotal, following are observed
  - i). Linearity is maintained as before
  - ii). Normality is maintained, same as before
  - iii). Equality is not very good, same as before
  - iv). Multicollinearity has decreased drastically as below
  - popcollege decreased from 14 to 8
  - popadult decreased from 167 to 7
  - popcollegelog decreased from 6 to 5
- b). Though 5 to 8 are lower values, the p-values increased drastically and Multiple R-squared values dropped. Therefore, compared to this model df.out.m.log1, the previous model df.out.m.log is better.

4. Determine outsized leverage and the county name and state

```
lev = hat(model.matrix(df.out.m.log))
plot(lev, pch=19)
abline(3*mean(lev), 0, col="red", lwd=3)
```

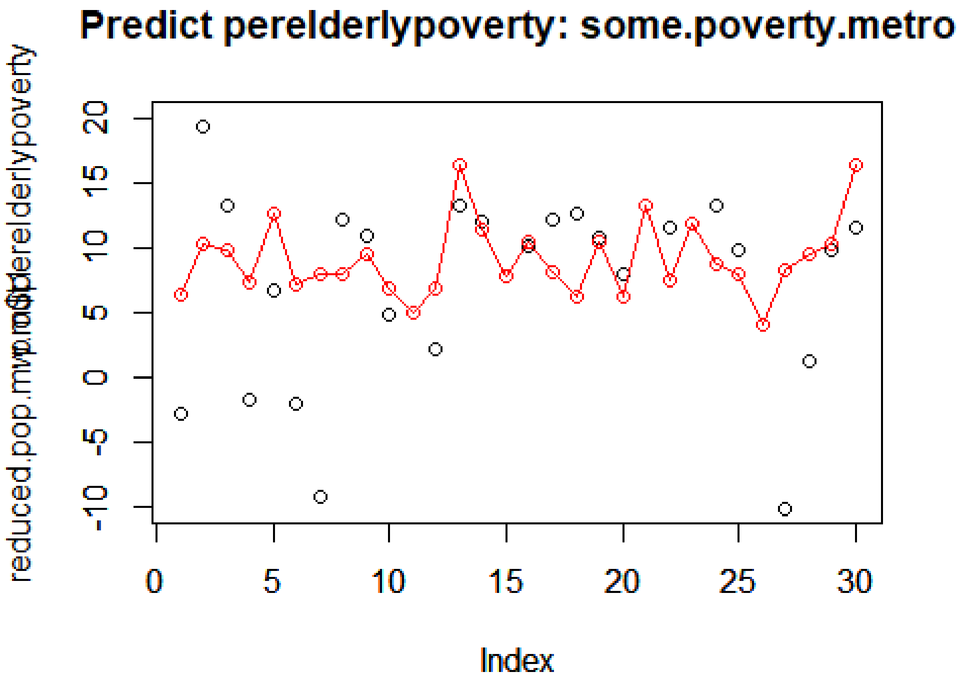


```
values = reduced.pop.mw.r[lev>(3*mean(lev)), ]
values

## # A tibble: 4 x 21
##      id county state  area poptotal popdensity popwhite popblack popasian
##   <dbl> <chr>  <chr> <dbl>    <dbl>      <dbl>    <dbl>    <dbl>    <dbl>
## 1   231 ISABE~ MI     574.    54624      95.1     52212      635      456
## 2   309 HANCO~ OH     531.    65536     123.      63572      591      401
## 3   236 KEWEE~ MI     541.     1701       3.14     1688        1        6
## 4   148 LA PO~ IN     598.   107066     179.      96286     9580      431
## # ... with 12 more variables: popadult <dbl>, popchild <dbl>,
## #   percollege <dbl>, perprof <dbl>, perchildpoverty <dbl>,
## #   perelderlypoverty <dbl>, inmetro <dbl>, popcollege <dbl>,
## #   popprof <dbl>, popcollegelog <dbl>, popca <dbl>, popchpov <dbl>

sprintf('The outsized leverage county names and states are %s, %s, %s, %s,
%s, %s, %s, %s', values[1,2], values[1,3], values[2,2], values[2,3],
values[3,2], values[3,3], values[4,2], values[4,3])

## [1] "The outsized leverage county names and states are ISABELLA, MI,
HANCOCK, OH, KEWEENAW, MI, LA PORTE, IN"
```

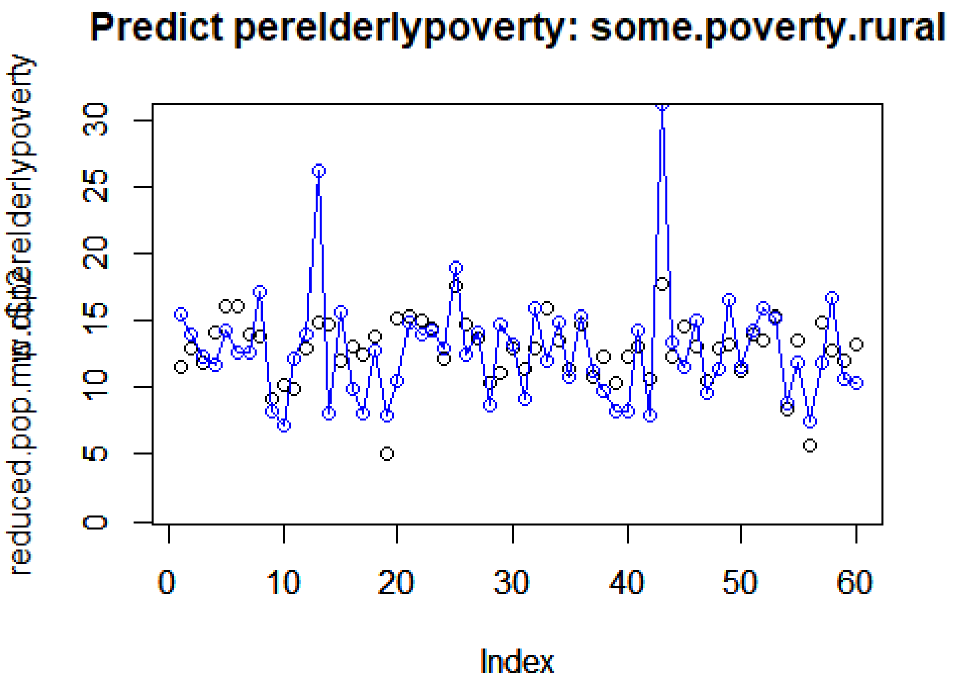
5. Predict perelderlypoverty for some.metro.poverty. State if the fit is better or worse compared to some.rural.poverty

```
p.out1 = predict(df.out.m.log, newdata=reduced.pop.mw.m)
plot(p.out1, ylim=c(-10,20))
par(new=TRUE)
```

```r
plot(reduced.pop.mw.m$perelderlypoverty, type='o', col='red', ylim=c(-10,20))
title('Predict perelderlypoverty: some.poverty.metro')
```



```r
p.out2 = predict(df.out.m.log, newdata=reduced.pop.mw.r)
plot(p.out2, ylim=c(1,30))
par(new=TRUE)
plot(reduced.pop.mw.r$perelderlypoverty, type='o', col='blue', ylim=c(1,30))
title('Predict perelderlypoverty: some.poverty.rural')
```



- a). Based on the above graphs and the predict function used on both the some.poverty.metro and some.poverty.rural data, following are observed
  - i). The prediction of values is closely matching the actual values for some.poverty.rural
  - ii). The prediction of values is only 50% reliable when compared to actual values for some.poverty.metro