Analysis of the county data for alabama (UScensus20)

loading the libraries that we require primarily for the analysis in R. We need the sf package is required to deal with the data from the UScensus20 package as the data itself is a sf data frame and ggplot2 is used for visualizing the data. Psych is used for making correlation matrix as making a correlation matrix in base r is a little cumbersome

```
load("~/Downloads/alabamacounty20.rda")
library(sf)
```

```
## Linking to GEOS 3.10.2, GDAL 3.4.1, PROJ 8.2.1; sf_use_s2() is TRUE
```

```
library(ggplot2)
library(psych)
```

```
##
## Attaching package: 'psych'

## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```
library(UScensuscounty20)
```

alabamacounty20 contains the redistricting file data from the US census bureau. The data is available in a ready to use file format up on my github in a data package called "UScensuscounty20". Wrapper functions to access the data are available in the package "US census20".

```
data=alabamacounty20
#This lets me look at the structure of the data
str(data)
```

```
## Classes 'sf' and 'data.frame':   67 obs. of  400 variables:
##  $ GEOID20 : chr  "01001" "01003" "01005" "01007" ...
##  $ MTFCC20 : chr  "G4020" "G4020" "G4020" "G4020" ...
##  $ FILEID  : chr  "PLST" "PLST" "PLST" "PLST" ...
##  $ STUSAB  : chr  "AL" "AL" "AL" "AL" ...
##  $ SUMLEV  : chr  "050" "050" "050" "050" ...
##  $ GEOVAR  : chr  "00" "00" "00" "00" ...
##  $ GEOCOMP : chr  "00" "00" "00" "00" ...
##  $ CHARITER: chr  "000" "000" "000" "000" ...
##  $ CIFSN   : chr  "03" "03" "03" "03" ...
##  $ LOGRECNO: chr  "0000002" "0000003" "0000004" "0000005" ...
##  $ GEOID   : chr  "0500000US01001" "0500000US01003" "0500000US01005" "0500000US01007" ...
##  $ REGION  : chr  "3" "3" "3" "3" ...
```

```
##  $ DIVISION: chr  "6" "6" "6" "6" ...
##  $ STATE   : chr  "01" "01" "01" "01" ...
##  $ STATENS : chr  "01779775" "01779775" "01779775" "01779775" ...
##  $ COUNTY  : chr  "001" "003" "005" "007" ...
##  $ COUNTYCC: chr  "H1" "H1" "H1" "H1" ...
##  $ COUNTYNS: chr  "00161526" "00161527" "00161528" "00161529" ...
##  $ COUSUB  : chr  "" "" "" "" ...
##  $ COUSUBCC: chr  "" "" "" "" ...
##  $ COUSUBNS: chr  "" "" "" "" ...
##  $ SUBMCD  : chr  "" "" "" "" ...
##  $ SUBMCDCC: chr  "" "" "" "" ...
##  $ SUBMCDNS: chr  "" "" "" "" ...
##  $ ESTATE  : chr  "" "" "" "" ...
##  $ ESTATECC: chr  "" "" "" "" ...
##  $ ESTATENS: chr  "" "" "" "" ...
##  $ CONCIT  : chr  "" "" "" "" ...
##  $ CONCITCC: chr  "" "" "" "" ...
##  $ CONCITNS: chr  "" "" "" "" ...
##  $ PLACE   : chr  "" "" "" "" ...
##  $ PLACECC : chr  "" "" "" "" ...
##  $ PLACENS : chr  "" "" "" "" ...
##  $ TRACT   : chr  "" "" "" "" ...
##  $ BLKGRP  : chr  "" "" "" "" ...
##  $ BLOCK   : chr  "" "" "" "" ...
##  $ AIANHH  : chr  "" "" "" "" ...
##  $ AIHHTLI : chr  "" "" "" "" ...
##  $ AIANHHFP: chr  "" "" "" "" ...
##  $ AIANHHCC: chr  "" "" "" "" ...
##  $ AIANHHNS: chr  "" "" "" "" ...
##  $ AITS    : chr  "" "" "" "" ...
##  $ AITSFP  : chr  "" "" "" "" ...
##  $ AITSCC  : chr  "" "" "" "" ...
##  $ AITSNS  : chr  "" "" "" "" ...
##  $ TTRACT  : chr  "" "" "" "" ...
##  $ TBLKGRP : chr  "" "" "" "" ...
##  $ ANRC    : chr  "" "" "" "" ...
##  $ ANRCCC  : chr  "" "" "" "" ...
##  $ ANRCNS  : chr  "" "" "" "" ...
##  $ CBSA    : chr  "33860" "19300" "21640" "13820" ...
##  $ MEMI    : chr  "1" "1" "2" "1" ...
##  $ CSA     : chr  "388" "380" "999" "142" ...
##  $ METDIV  : chr  "99999" "99999" "99999" "99999" ...
##  $ NECTA   : chr  "" "" "" "" ...
##  $ NMEMI   : chr  "" "" "" "" ...
##  $ CNECTA  : chr  "" "" "" "" ...
##  $ NECTADIV: chr  "" "" "" "" ...
##  $ CBSAPCI : chr  "" "" "" "" ...
##  $ NECTAPCI: chr  "" "" "" "" ...
##  $ UA      : chr  "" "" "" "" ...
##  $ UATYPE  : chr  "" "" "" "" ...
##  $ UR      : chr  "" "" "" "" ...
##  $ CD116   : chr  "" "" "" "" ...
##  $ CD118   : chr  "" "" "" "" ...
##  $ CD119   : chr  "" "" "" "" ...
```

```
##  $ CD120   : chr  "" "" "" "" ...
##  $ CD121   : chr  "" "" "" "" ...
##  $ SLDU18  : chr  "" "" "" "" ...
##  $ SLDU22  : chr  "" "" "" "" ...
##  $ SLDU24  : chr  "" "" "" "" ...
##  $ SLDU26  : chr  "" "" "" "" ...
##  $ SLDU28  : chr  "" "" "" "" ...
##  $ SLDL18  : chr  "" "" "" "" ...
##  $ SLDL22  : chr  "" "" "" "" ...
##  $ SLDL24  : chr  "" "" "" "" ...
##  $ SLDL26  : chr  "" "" "" "" ...
##  $ SLDL28  : chr  "" "" "" "" ...
##  $ VTD     : chr  "" "" "" "" ...
##  $ VTDI    : chr  "" "" "" "" ...
##  $ ZCTA    : chr  "" "" "" "" ...
##  $ SDELM   : chr  "" "" "" "" ...
##  $ SDSEC   : chr  "" "" "" "" ...
##  $ SDUNI   : chr  "" "" "" "" ...
##  $ PUMA    : chr  "" "" "" "" ...
##  $ AREALAND: chr  "1539634184" "4117656199" "2292160149" "1612188717" ...
##  $ AREAWATR: chr  "25674812" "1132956041" "50523213" "9572303" ...
##  $ BASENAME: chr  "Autauga" "Baldwin" "Barbour" "Bibb" ...
##  $ NAME    : chr  "Autauga County" "Baldwin County" "Barbour County" "Bibb County" ...
##  $ FUNCSTAT: chr  "A" "A" "A" "A" ...
##  $ GCUNI   : chr  "" "" "" "" ...
##  $ POP100  : chr  "58805" "231767" "25223" "22293" ...
##  $ HU100   : chr  "24350" "124148" "11618" "9002" ...
##  $ INTPTLAT: chr  "+32.5322367" "+30.6592183" "+31.8702531" "+33.0158929" ...
##  $ INTPTLON: chr  "-086.6464395" "-087.7460666" "-085.4051035" "-087.1271475" ...
##  $ LSADC   : chr  "06" "06" "06" "06" ...
##  $ PARTFLAG: chr  "" "" "" "" ...
##  $ UGA     : chr  "" "" "" "" ...
##  $ P0010001: chr  "58805" "231767" "25223" "22293" ...
##   [list output truncated]
##  - attr(*, "sf_column")= chr "geometry"
##  - attr(*, "agr")= Factor w/ 3 levels "constant","aggregate",..: NA NA NA NA NA NA NA NA NA NA ...
##   ..- attr(*, "names")= chr [1:399] "GEOID20" "MTFCC20" "FILEID" "STUSAB" ...
```

```
alabama=data.frame("name"=data$BASENAME,"pop"=as.integer(data$P0010001),"institutional"=as.integer(data$
```

The original data frame i.e. alabamacounty20 has nearly 400 variables and 67 observations. Although, not all of them are particularly useful for this project as i am only interested in the correlation between the population per county and the no of nursing facilities by the county. Furthermore, i also want to analyze which factors from the files 2 and 3 i.e. housing and facilities data have the most impact on county wise total population. This article is divided in three sections. The first section focuses on the visualization of the county data (population maps,age pyramid, choropleth maps and many others), the second section attempts to find if there is a relationship between the black population for the county and the number of institutional facilities and the third section is a multiple linear regression to find out the factors that affect the total population count the most for the state of Alabama

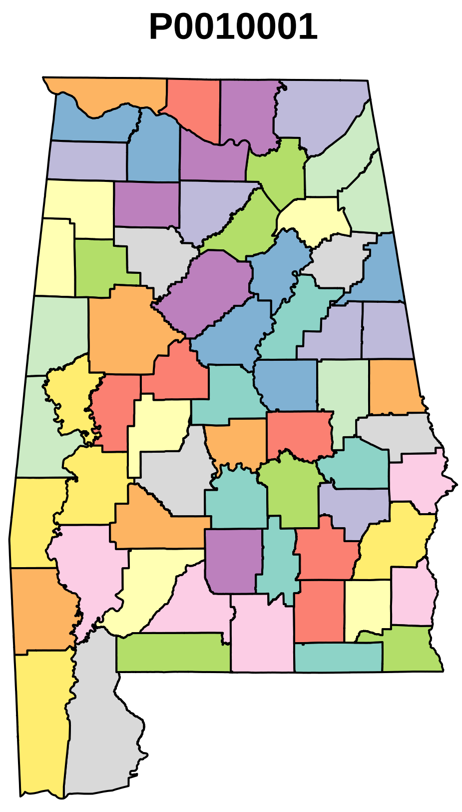
VISUALIZING THE CENSUS DATA

- Choropleth maps:

choropleth maps are a pictorial representation of the data and the cartographic boundaries. They typically follow a crs and require lat long coordinates.

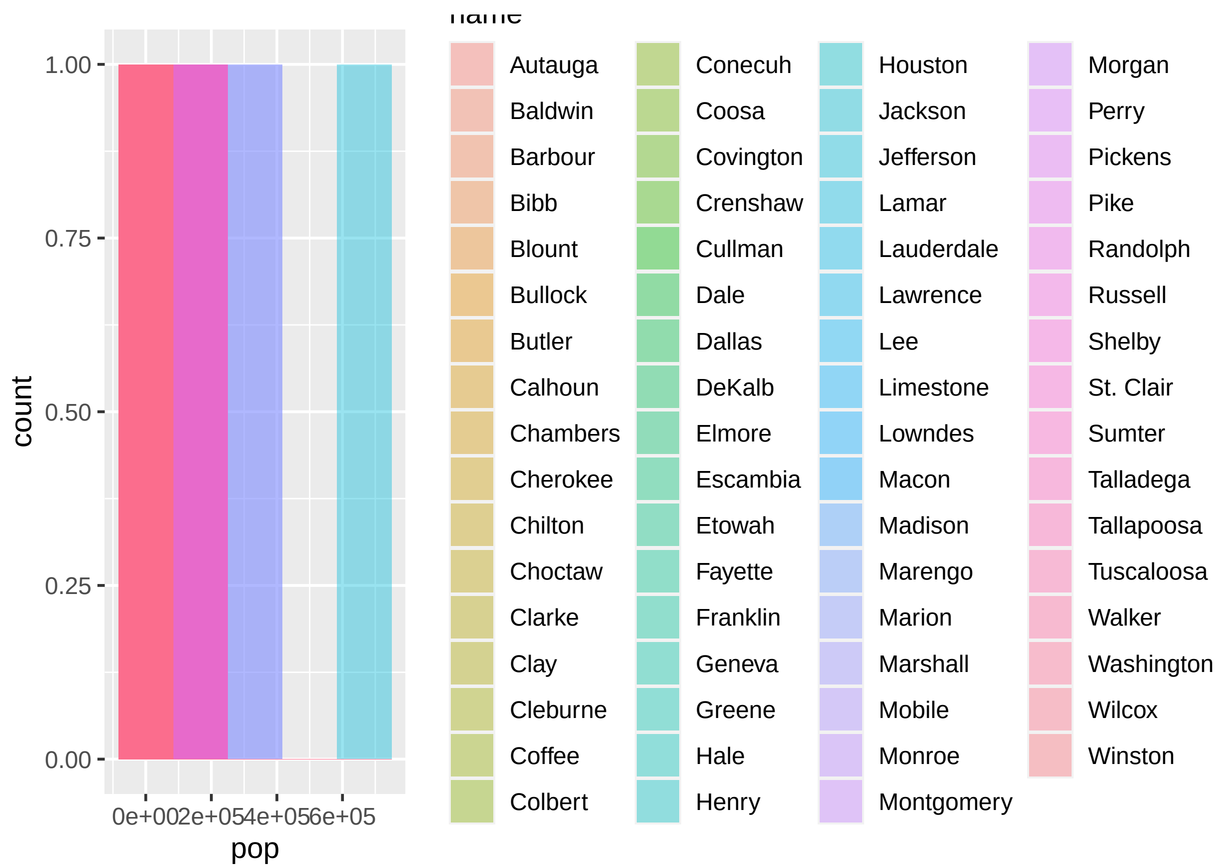
The UScensus20 package has shape file data joint with the data frame and hence it saves a lot of time and effort in writing the code and making it more GIS-friendly. The below graph is a choropleth map of the county wise total population for the state of Alabama:

```
plot(alabamacounty20['P0010001'])
```




```
#this is a very basic choropleth map, modifications can be done here or can use packages like ggplot2 o
```

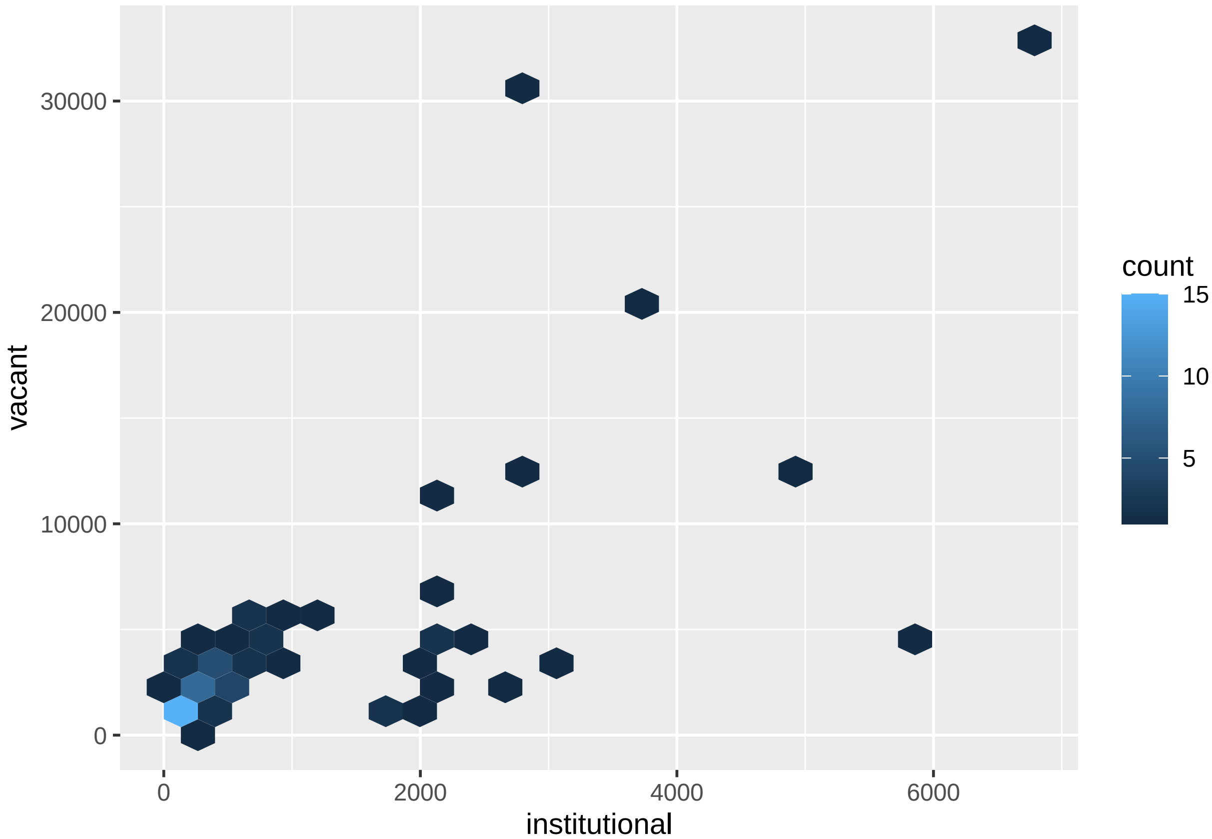
```
ggplot(alabama, aes(x = pop, fill = name)) +
  geom_histogram(position = "identity", alpha = 0.4,bins=5)
```




```
#wrote 5 so that it can be visible
```

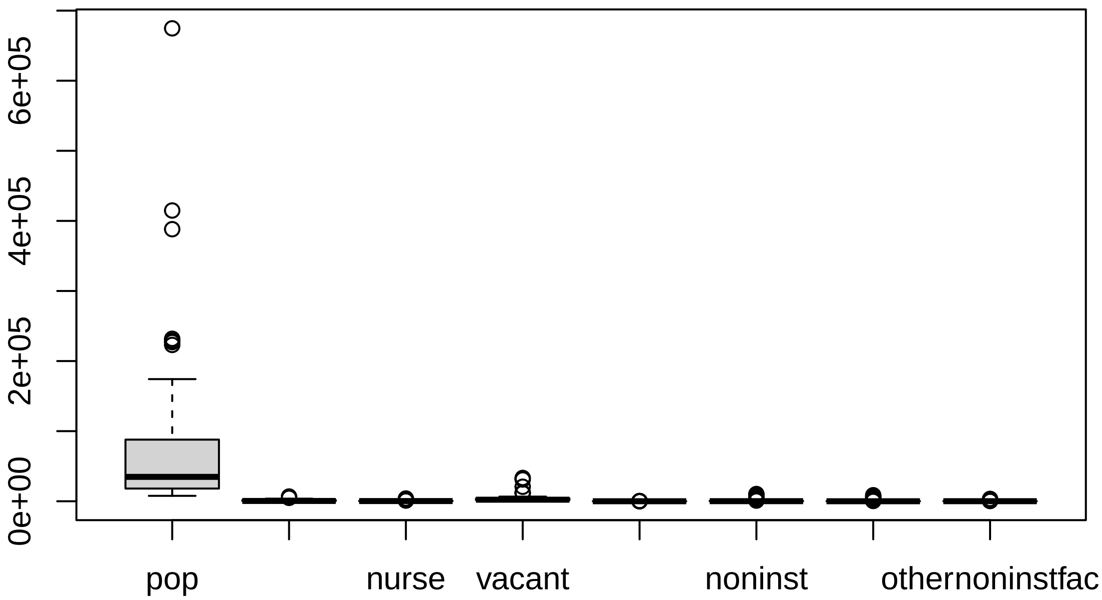
Hex plot between the instutional and nursing facilities:

```
d <- ggplot(alabama, aes(institutional, vacant))
d + geom_hex(bins=25)
```



Boxplot: Useful for finding the outliers and getting the general sense of the data. The outliers can be visualized using boxplot like this:

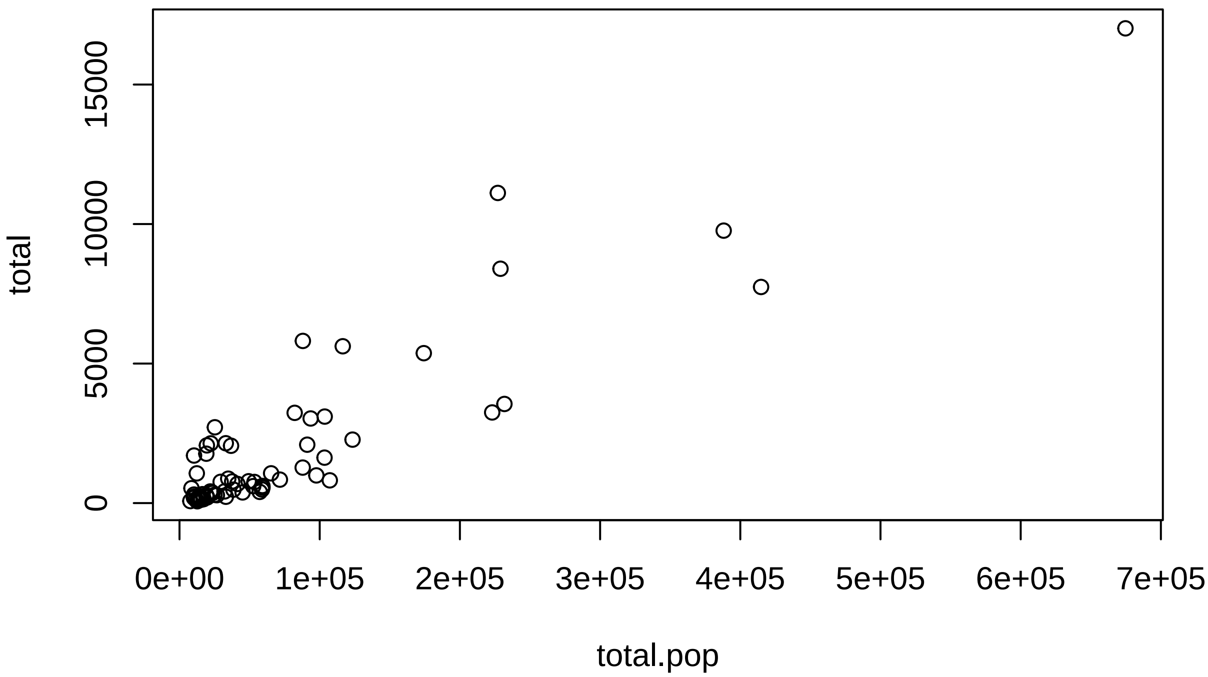
```
boxplot(x=alabama[,-1])$out
```




```
##  [1] 231767 674721 388153 414809 228954 223024 227036   5791   6717   4836
## [11]   1212   3884   1794   1769   1298   1022  31032  33228  12050  20691
## [21]  12070  11498     74    180    114    162    162    270     67     69
## [31]   3554  10297   2068   4656   1988   7011   4052   3563   1941   1196
## [41]    981   8949   3345   6952   1914   4535   1882   6305   2923   2647
## [51]    408   1915   1147    981    733   8576    482   3345    706   1129
## [61]    373
```

Scatterplot: Useful for finding out the relationships between the variables and to detect potential outliers

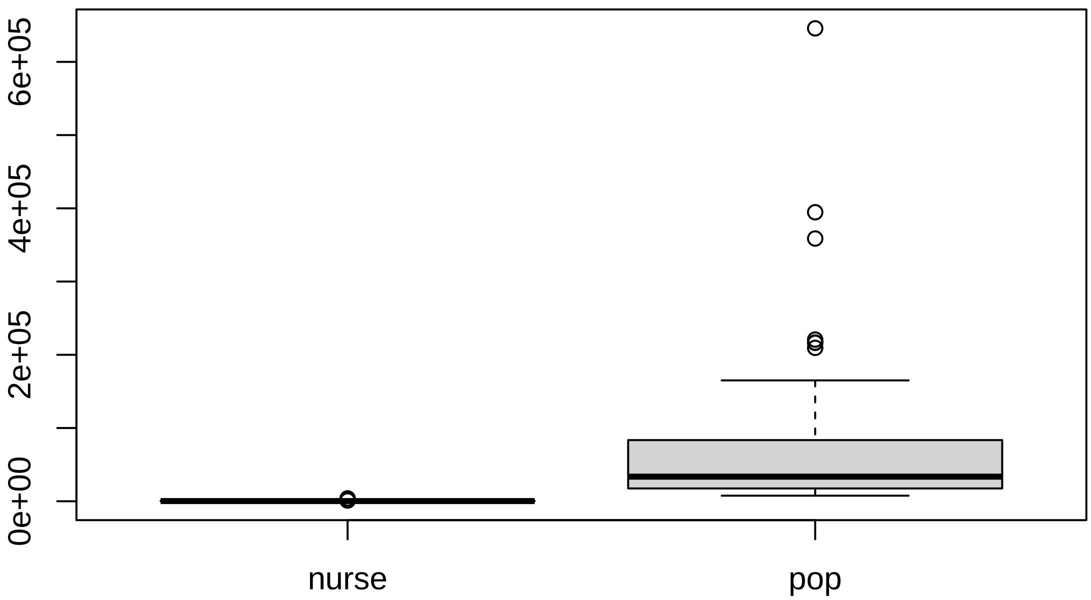
```
alabama_scatter=data.frame("total pop"=alabamacounty20$P0010001,"total"=alabamacounty20$P0050001)
plot(alabama_scatter)
```



SIMPLE LINEAR REGRESSION : I will be using the linear regression to find out if the county wise population of people of black race has any effect on the county wise institutionalized population

```
#creating the dataframe of two dataframes
alabama_reg_sim=data.frame("nurse"=as.integer(alabamacounty20$P0050005), "pop"=as.integer(alabamacounty2
```

```
#detecting the outliers
boxplot(alabama_reg_sim)$out
```




```
##  [1]    1212    3884    1794    1769    1298    1022 216743 645772 358764 394678
## [11] 220759 209440 216301
```

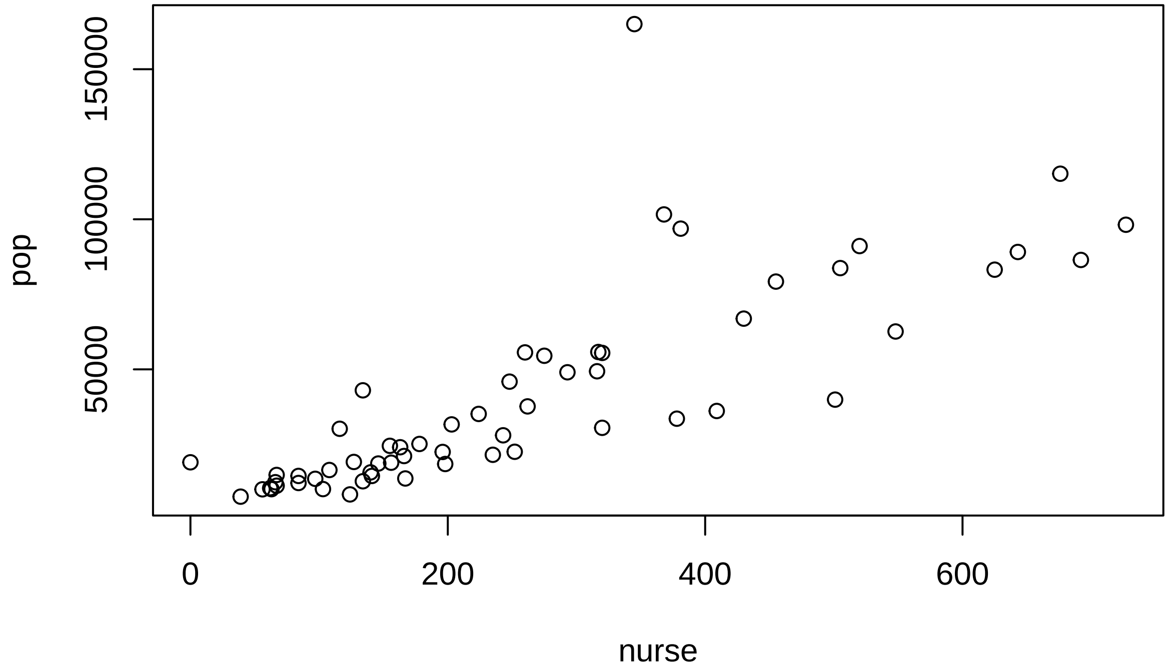
Removing the outliers:

```
outliers <- function(x) {
 #defines the quartiles and the interquantile ranges and detecting the data points which exceed the upp
  Q1 <- quantile(x, probs=.25)
  Q3 <- quantile(x, probs=.75)
  iqr = Q3-Q1

 upper_limit = Q3 + (iqr*1.5)
 lower_limit = Q1 - (iqr*1.5)

 x > upper_limit | x < lower_limit
}
#this function basically takes the data and the columns that you want to remove the outliers from and t
remove_outliers <- function(df, cols = names(df)) {
  for (col in cols) {
    df <- df[!outliers(df[[col]]),]
  }
  df
}
#this new data frame has no outliers hence the accuracy of the fit will not be affected
alabama_reg_sim=remove_outliers(alabama_reg_sim, c('pop','nurse'))
```

```
plot(alabama_reg_sim)
```



From the current plot, we can see that there is a relationship between the total population per county and the nursing facilities per county. i.e. with increasing in population count, the nursing facilities increase too.

The relationship can be modeled as:

population=constant coefficient * nursing facilities. Obviously, the depending upon the value of that constant coefficient, the "goodness" of the fit will differ. This coefficient can help us determine that given that nursing facilities for a county in Alabama, can we predict the total population of the county. How good our coefficient is at depicting the data can be evaluated by several error. MSE is a popular error metric. It can be calculated using:

```
mean_square_error=function(coef){
  #the relationship between two variables
  pop_est= coef*alabama_reg_sim$nurse
  #finding the error
  err=pop_est-alabama_reg_sim$pop
  #the returned will be the mean squared error
  return(mean(err*err))
}
```

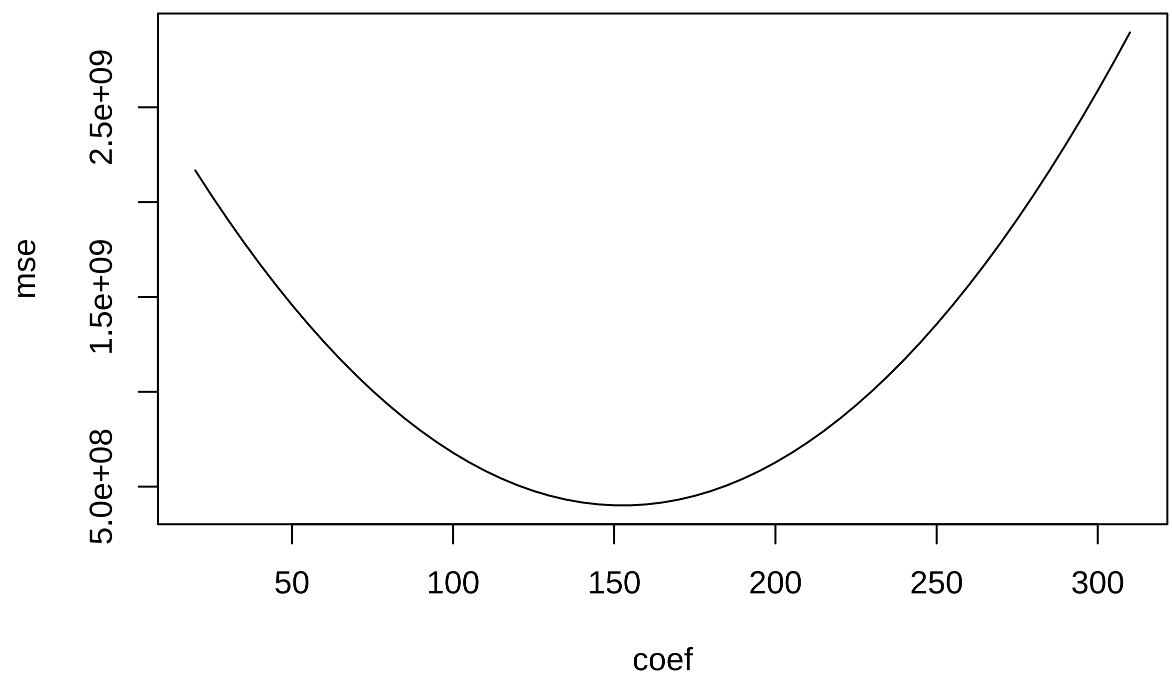
Finding out the coefficient for which error is minimum. This is a bit of trial and error but i estimated the range of efficiency by randomly imputing the coefficient values in the mean squared error function which gave me a range of 100 to 300. Will be calculating the mse for all of them and then plotting them to find the minimum

```
mean_square_error(400)
```

```
## [1] 6559598651
```

```
coef=seq(20,310,5)
alabama_copy=alabama_reg_sim
abc=data.frame(coef)
abc$mse=sapply(abc$coef,mean_square_error)
```

```
plot(abc,type="l")
```

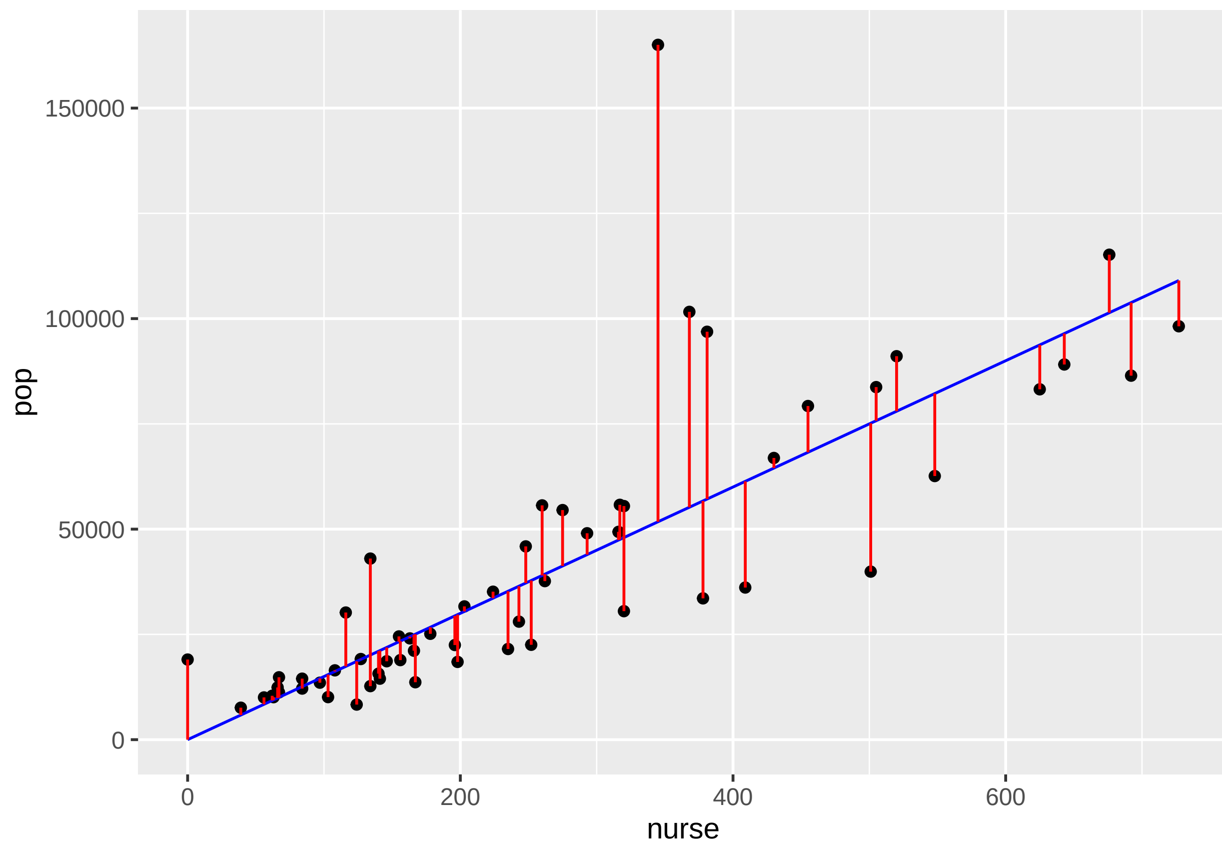


From the graph, we can see that the best estimate would be around when the value of the coefficient is 150. so we will be using that to fit our data. This is what it looks like:

```
alabama_reg_sim$pop_est=150*alabama_reg_sim$nurse
```

This plot is kind of the best estimate that i could get for the data.

```
alabama_reg_sim%>%
  ggplot()+
  geom_point(aes(y=pop,x=nurse),color="black")+
  geom_line(aes(y=pop_est,x=nurse),color="blue")+
  geom_segment(aes(x=nurse,xend=nurse,y=pop,yend=pop_est),color="red")
```



The same process can also be done by base R. I used lm here but there are several other methods which can be used as well By base R: This uses calculus to find the mse instead of the brute force approach used here and a lot of the work is done under the hood but it can be tricky for a peculiar variables

```
sim_model=lm(formula = pop~nurse,data=alabama_reg_sim)
summary(sim_model)
```

```
##
## Call:
## lm(formula = pop ~ nurse, data = alabama_reg_sim)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -35614  -9344  -2210   4862 112066
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3050.73    4594.75   0.664    0.509
## nurse         144.66      14.49   9.984 3.99e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 20290 on 57 degrees of freedom
## Multiple R-squared:  0.6362, Adjusted R-squared:  0.6298
## F-statistic: 99.68 on 1 and 57 DF,  p-value: 3.985e-14
```

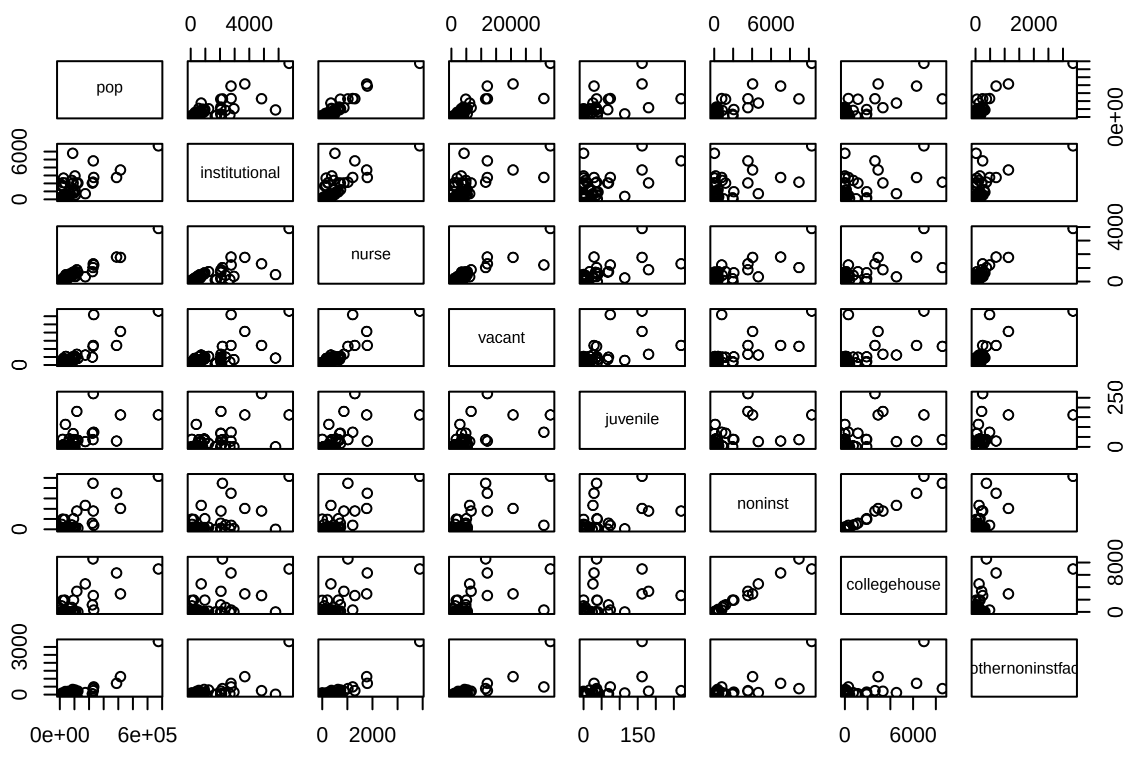
From here we can see that the adjusted r squared for this fit is around 0.6298. R squared is a popular performance metric which basically tells you how much variaability of the data is explained by the model. As a general rule of thumb, the better the value of rsquared, the better your model is. In our case, the value is around 0.63. This can be done even better by using multiple regression

PERFORMING MULTIPLE REGRESSION: I am using multiple regression to hopefully find the variables that can explain and predict the trends of total population per county in alabama. Given a set of around 400 features, i carefully selected around 10 of them. This step is largely based off of domain knowledge. However, not all of these variables will be used for final analysis.

```
#creating a dataframe with the important variables from file 2 and file 3
alabama=data.frame("name"=data$BASENAME,"pop"=as.integer(data$P0010001),"institutional"=as.integer(data$
```

The reason R works well for data analysis is because of its vectorization and graphical capabilities. Plotting multivariate scatter graph is as easy as just using the plot function in R. this makes EDA and visualization awfully easy using R. the [] are used for sub-setting rows and columns in R. the below code indicates that we want all the rows and columns except for the first column which happens to be our name column

```
plot(alabama[,-1])
```



This plot lets us get a general idea and a feel of the data. A lot of features here are not useful and performing feature selection will not only make sense because things would be a lot less computationally intensive but also it can improve the accuracy of our analysis. I am using variance as a feature selection metric to choose a subset of the features. I will be dropping the features with low variability as they will not contribute much to the analysis. The variance of the column can be calculated using the var() function in base R.

```r
#using the same function as i did in uni variate regression to detect and remove the outliers from an e
outliers <- function(x) {

  Q1 <- quantile(x, probs=.25)
  Q3 <- quantile(x, probs=.75)
  iqr = Q3-Q1

 upper_limit = Q3 + (iqr*1.5)
 lower_limit = Q1 - (iqr*1.5)

 x > upper_limit | x < lower_limit
}

remove_outliers <- function(df, cols = names(df)) {
  for (col in cols) {
    df <- df[!outliers(df[[col]]),]
  }
  df
}
alabama_no_outliers=remove_outliers(alabama, c('pop', 'institutional', 'nurse','vacant','juvenile','non
```

Since the outliers are now removed, let us look at the variance

```r
sapply(alabama_no_outliers[,-1],var)
```

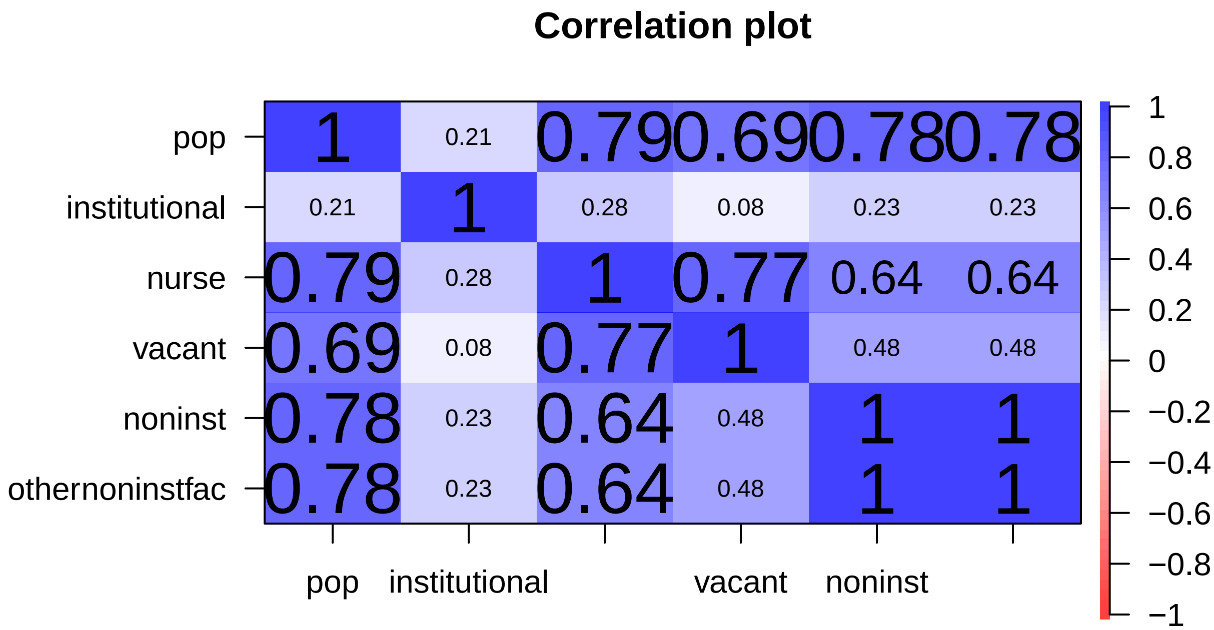
```
##             pop   institutional           nurse          vacant        juvenile
##    2.216904e+08    1.758800e+05    1.286829e+04    1.134264e+06    0.000000e+00
##         noninst    collegehouse othernoninstfac
##    1.737513e+03    0.000000e+00    1.737513e+03
```

We will be dropping the columns which have low variance and hence little impact on the data

```r
alabama_no_outliers=alabama_no_outliers[,-c(6,8)]
```

There are so many libraries in R which are helpful for making a correlation matrix but my personal favorite is psych. This color codes the correlation between variables and it makes it really easy to find what we are looking for just at a glance

```r
library(psych)
cor.plot(alabama_no_outliers[,-1])
```



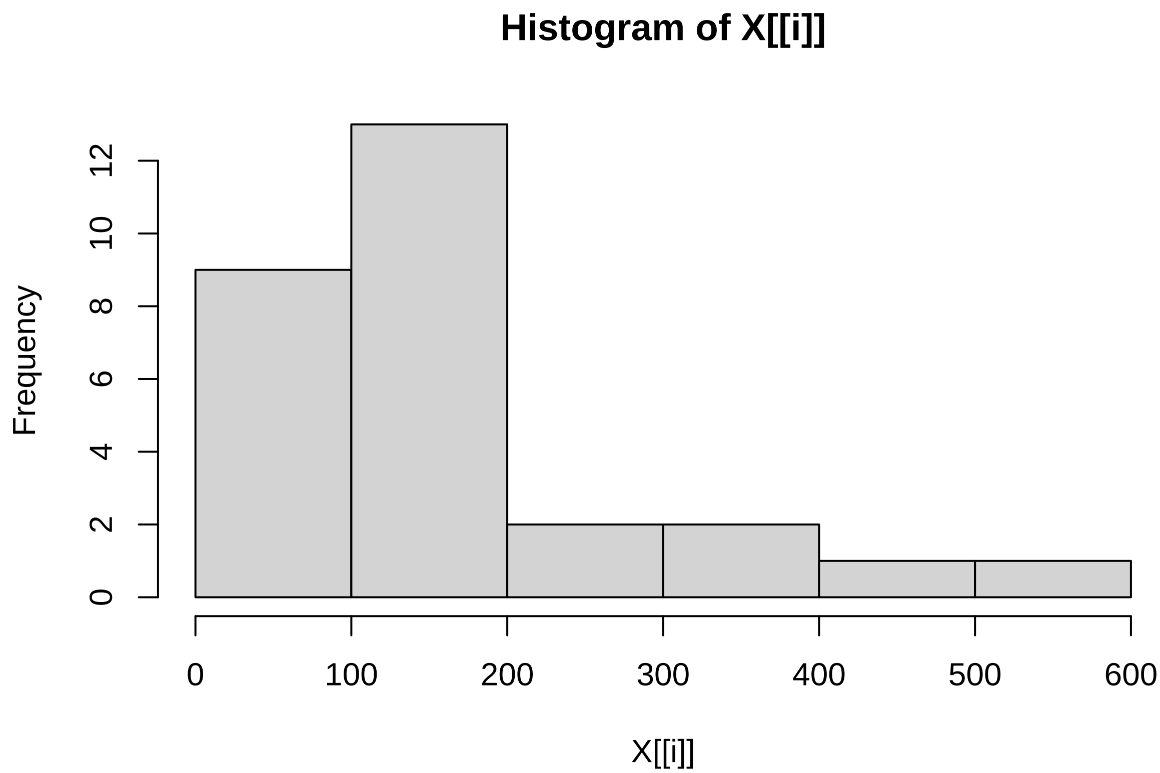
From the correlation matrix, it is clear that institutional variable doesn't account much for the data and is not correlated. Hence, we will be dropping institutional and other non institutional fac variables from our data frame

```
alabama_no_outliers=alabama_no_outliers[-c(1,3,7)]
```

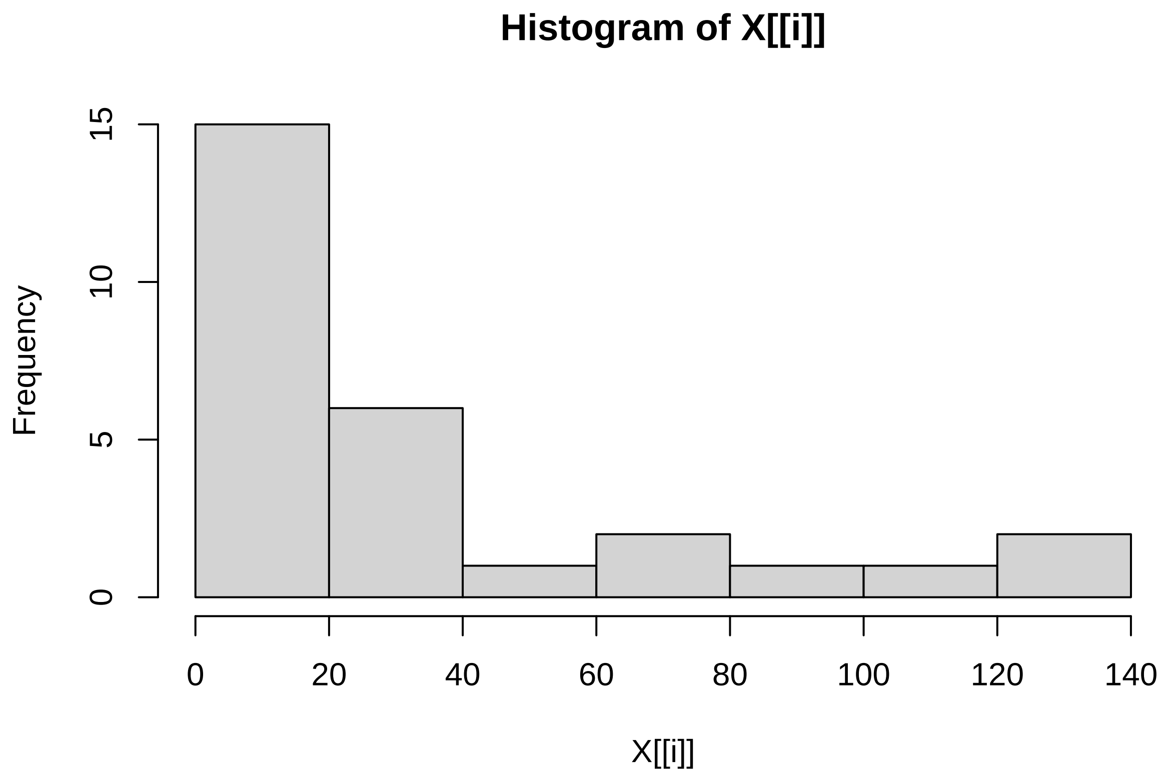
Visualizing the data to make sure that the data is ready for regression

```
sapply(alabama_no_outliers,hist)
```


Histogram of X[[i]]
Frequency
0
5
10
15
0
20000
40000
60000
80000
X[[i]]


Histogram of X[[i]]
Frequency
0
2
4
6
8
10
12
0
100
200
300
400
500
600
X[[i]]


Histogram of X[[i]]
Frequency
0
5
10
15
0
1000
2000
3000
4000
5000
6000
X[[i]]




```
##          pop       nurse     vacant    noninst
## breaks   numeric,9 numeric,7 numeric,7 numeric,8
## counts   integer,8 integer,6 integer,6 integer,7
## density  numeric,8 numeric,6 numeric,6 numeric,7
## mids     numeric,8 numeric,6 numeric,6 numeric,7
## xname    "X[[i]]"  "X[[i]]"  "X[[i]]"  "X[[i]]"
## equidist TRUE      TRUE      TRUE      TRUE
```

Performing multiple linear regression in R: After performing linear regression, i will now include the data for the files 2 and 3 to see if the other variables can explain the trend of the total population better than the nursing facilities did. Instead of trying the brute force approach, will simply be calculating the regression using the lm() function in base R as doing that approach here will be quite computationally intensive

```
model <- lm(pop~ nurse + vacant + noninst, data = alabama_no_outliers)
options(scipen=4)
summary(model)
```

```
##
## Call:
## lm(formula = pop ~ nurse + vacant + noninst, data = alabama_no_outliers)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13165.2  -3300.2   -660.8   2693.7  18037.4
##
```

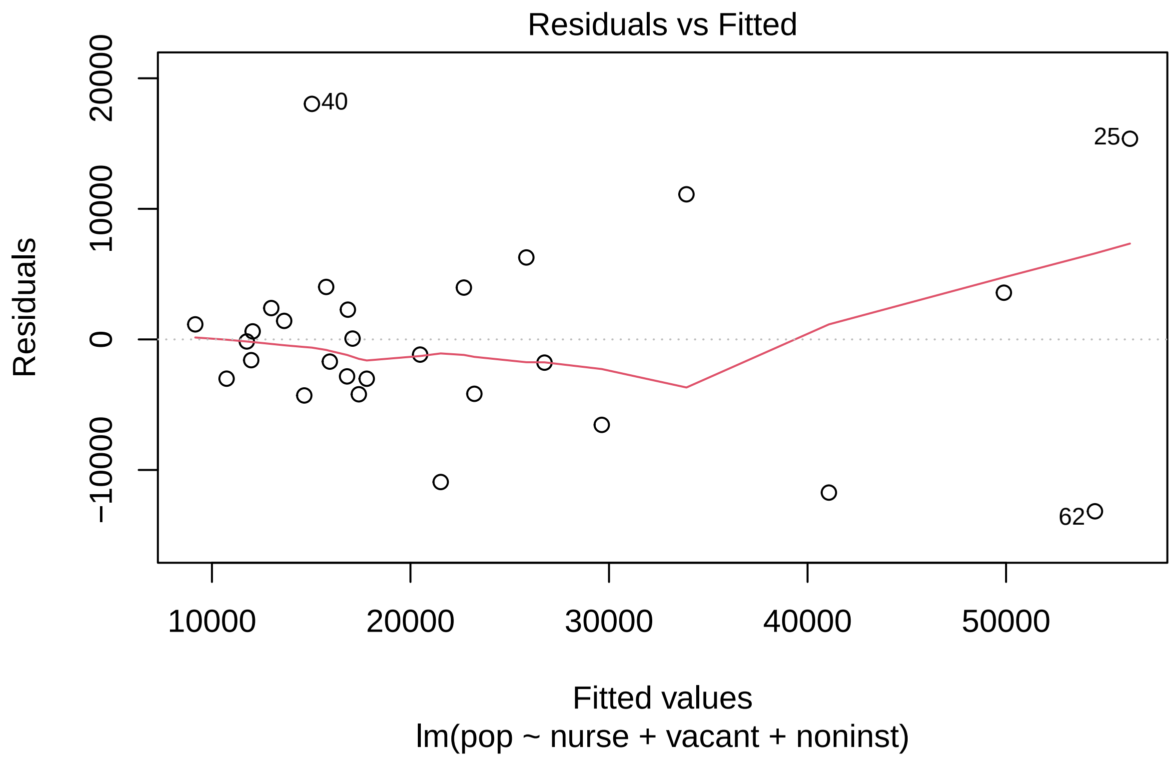
```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4932.738   2768.742   1.782 0.087478 .
## nurse         42.974     22.702   1.893 0.070482 .
## vacant         2.996      2.126   1.410 0.171451
## noninst      168.091     44.804   3.752 0.000984 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7490 on 24 degrees of freedom
## Multiple R-squared:  0.775,  Adjusted R-squared:  0.7469
## F-statistic: 27.56 on 3 and 24 DF,  p-value: 0.00000006025
```

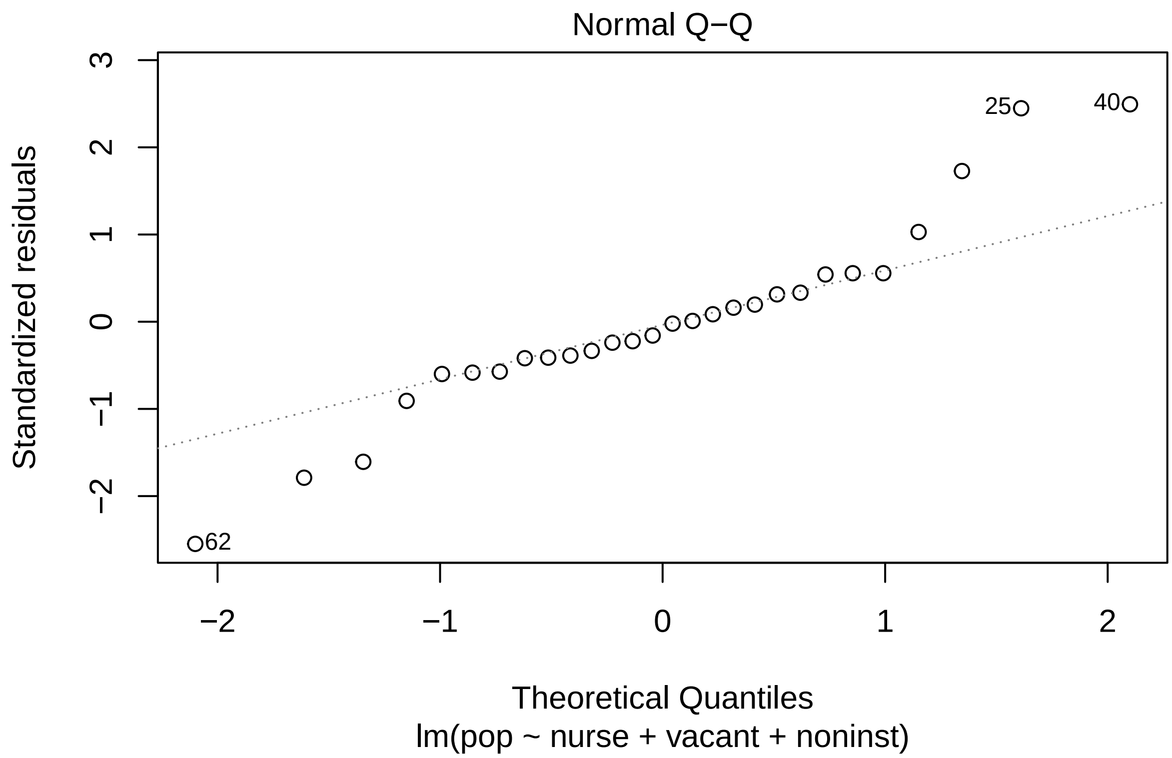
```
summary(model)$coefficient
```

```
##                Estimate  Std. Error  t value     Pr(>|t|)
## (Intercept) 4932.738027 2768.741700 1.781581 0.0874776131
## nurse         42.973647   22.701723 1.892969 0.0704822745
## vacant         2.996441    2.125562 1.409717 0.1714511189
## noninst      168.090921   44.803712 3.751719 0.0009842925
```

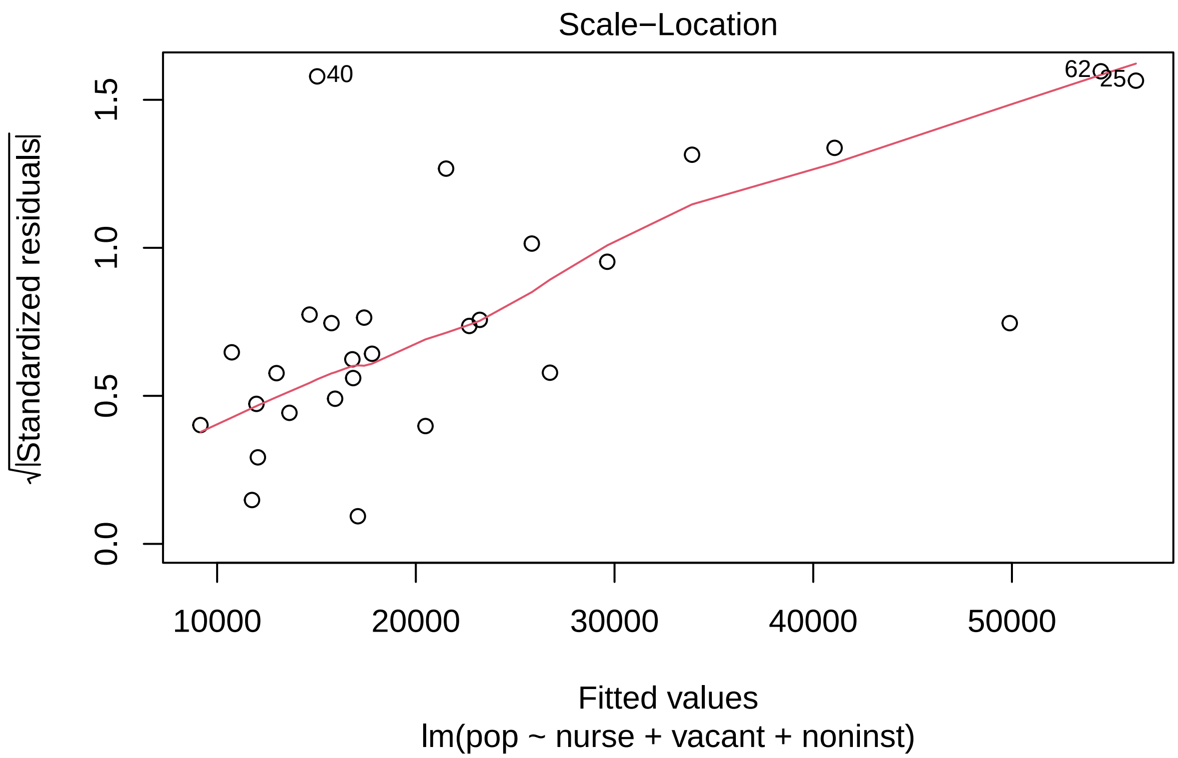
Upon looking at the rsquared, it is instantly clear that this model job explains the variance in the target variable a lot better than our previous uni variate model. This means that the nursing facilities, no of vacant houses and non institutionalized population does a better job of predicting the trends of the total population than the nursing facilities alone. Plotting the model summary can give us a quite a bit of further insight on the data and what further steps will taken. My approach, however will be limited to this as my sample size is quite small and some randomness is expected from less than 30 observations

```
plot(model)
```

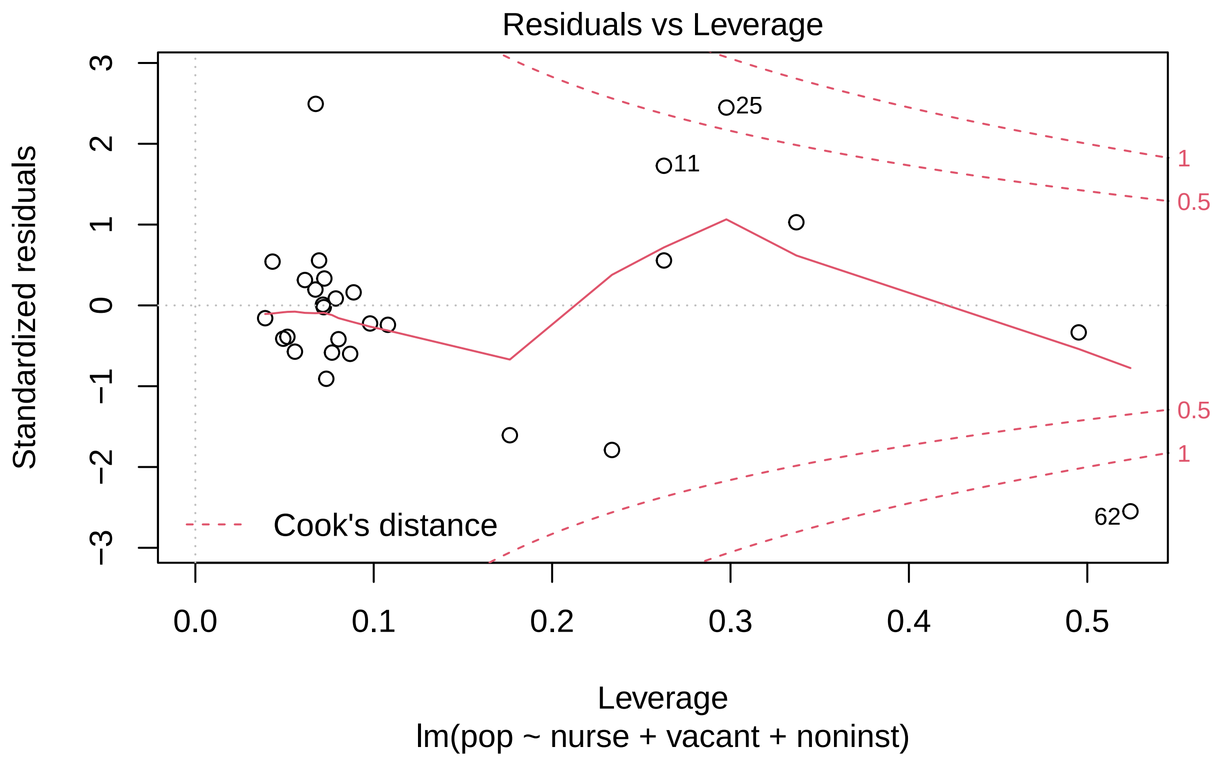

Residuals vs Fitted
Residuals
20000
10000
0
−10000
40
25
62
10000
20000
30000
40000
50000
Fitted values
lm(pop ~ nurse + vacant + noninst)



Normal Q−Q
Standardized residuals
3
2
1
0
−1
−2
25
40
62
−2
−1
0
1
2
Theoretical Quantiles
lm(pop ~ nurse + vacant + noninst)



Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
10000
20000
30000
40000
50000
Fitted values
lm(pop ~ nurse + vacant + noninst)
40
62
25




```r
#getting the pvalues
round(summary(model)$coef, 3)
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4932.738   2768.742   1.782    0.087
## nurse         42.974     22.702   1.893    0.070
## vacant         2.996      2.126   1.410    0.171
## noninst      168.091     44.804   3.752    0.001
```

Interpreting these plots:

The fitted values vs residuals plot indicates that the variance is not constant and usually when we look at the residuals vs fitted plot, the underlying assumption is that the variance is constant for residuals when plotted against fitted values i.e the data is heteroscedastic. which means our standard errors are biased and there is a higher chance that our model might not perform decent.

the q-q plot is light tailed which means that more data is gathered around the extremities than at the center. This often happens in real life data and some of the solutions to work around this would be to transform the data (e.g. using box-cox normality plot to transform the data)

```r
#Calculating the RSE
sigma(model)/mean(alabama_no_outliers$pop)
```

```
## [1] 0.3301087
```