**Analysis on Life Expectancy Rate for various countries**

**Group No**.: Group 02

**Student Names**: Shashikanth Dhanekula and Abhilash Nanja Ajjan

**Executive Summary:** Our goal from the study is to predict the Life Expectancy for the year 2016. Data is collected by the WHO (World Health Organization) repository. We have divided the data into Training set and Test set and applied Principal Component Analysis (PCA) to reduce the dimension. Multiple Linear Regression and Random Forest were the data mining techniques employed in this case study and we found that the Random Forest model is the best based on the accuracy measures.

**I. Background and Introduction:**

There have been a lot of studies in the past on factors affecting life expectancy considering demographic variables. It was found that the effect of immunization and human development index was not considered in the past. Also, some of the past research was done considering multiple linear regression based on data set of one year for all the countries. Hence, this gives motivation to resolve both the factors stated previously by formulating a regression model based on mixed effects model and multiple linear regression while considering data from a period of 2000 to 2015 for all the countries. Since the observations this dataset are based on different countries, it will be easier for a country to determine the predicting factor which is contributing to lower value of life expectancy. This will help in suggesting a country which area should be given importance in order to efficiently improve the life expectancy of its population.

The Global Health Observatory (GHO) data repository under World Health Organization (WHO) keeps track of the health status as well as many other related factors for all countries the data-sets are made available to the public for the purpose of health data analysis.

Sources:<https://www.kaggle.com/kumarajarshi/life-expectancy-who>

**Problem**:

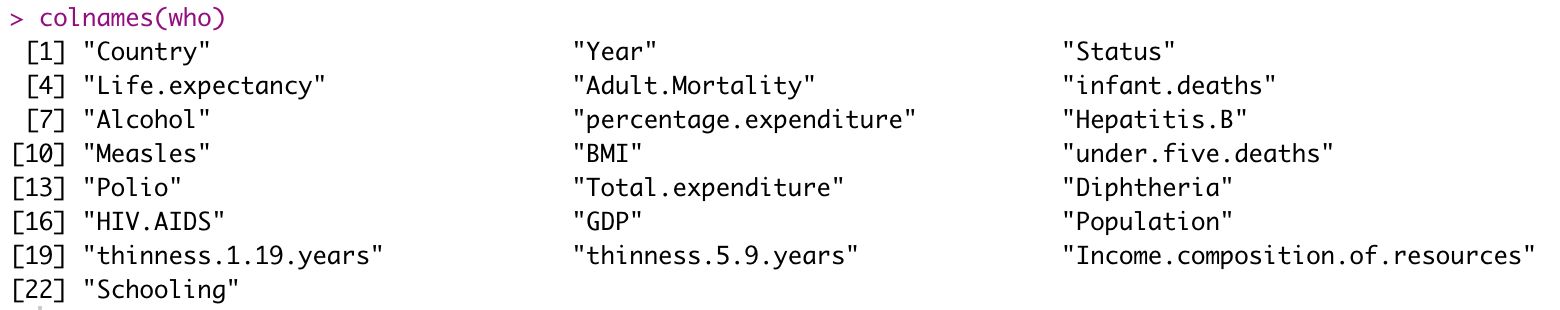
1. Does various predicting factors which has been chosen initially really affect the Life expectancy? What are the predicting variables affecting the life expectancy?
2. Does Life Expectancy have a positive or negative relationship with various factors affecting Life expectancy ?
3. To predict the Life Expectancy for all the countries across the globe in 2016.

**Objective:** Our goal is to develop the best model to predict the Life Expectancy for all the countries across the globe in 2016 using Multiple Linear Regression and Random Forest.

**Possible Solution**: Life Expectancy predicted for all the countries across the globe for the year 2016, to pick the factors that affect Life Expectancy the most and determine its effect (positive/negative).

# **II. Data Exploration and Visualization**

**Dataset:** The dataset is from kaggle (online repository) and contains 22 columns and 2938 rows. Below is the list of variables that we are dealing with where “Life.expectancy” is the response variable and the other 21 variables are predictor variables. There is a mix of both demographic variables and variables that account for the human development index.

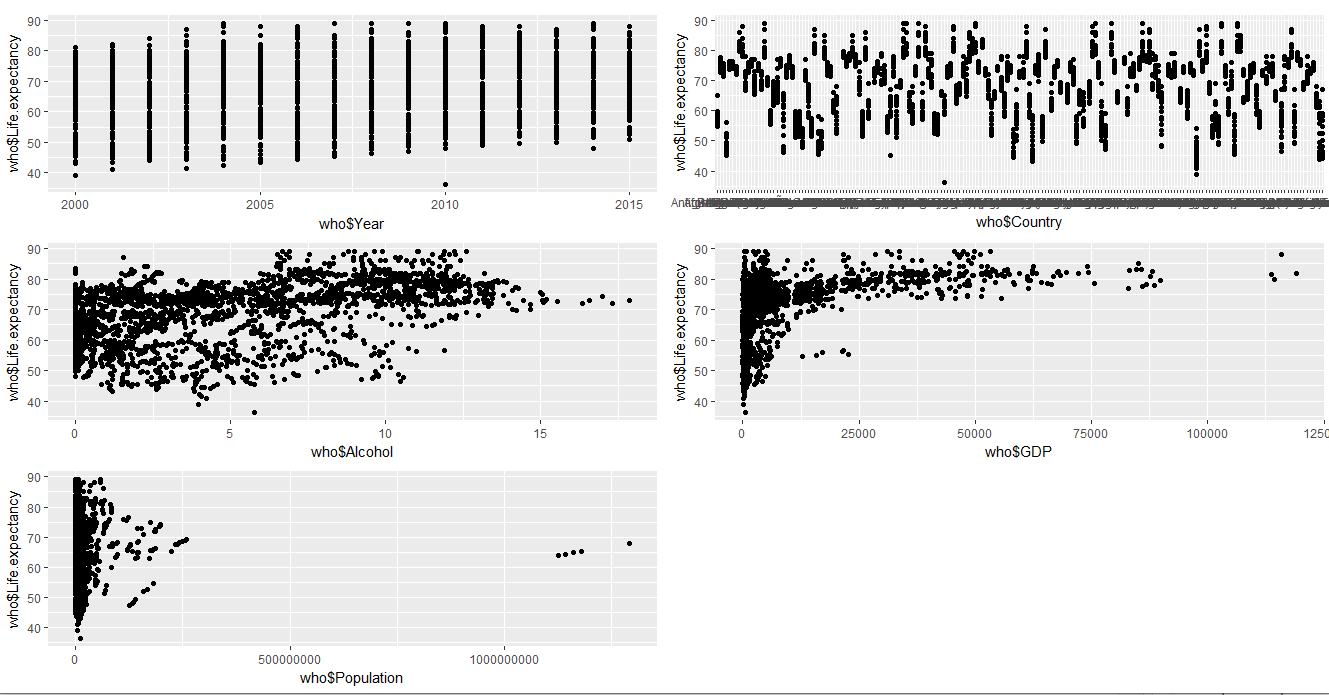


**Columns:**

1. **Country (factor) -** Name of the country
2. **Year (numeric) -** Year in which the corresponding data were collected
3. **Status (factor) -** Determines if the country is a “Developed” or “Developing” country
4. **Life expectancy (numeric) -** Life expectancy in terms of age
5. **Adult Mortality (numeric) -** Adult Mortality Rates of both sexes (probability of dying between 15 and 60 years per 1000 population)
6. **Infant Deaths (numeric) -** Number of Infant Deaths per 1000 population
7. **Alcohol (numeric) -** Alcohol, recorded per capita (15+) consumption (in litres of pure alcohol)
8. **Percentage expenditure (numeric) -** Expenditure on health as a percentage of Gross Domestic Product per capita(%)
9. **Hepatitis B (numeric) -** Hepatitis B (HepB) immunization coverage among 1-year-olds (%)
10. **Measles (numeric) -** number of reported measles cases per 1000 population
11. **BMI (numeric) -** Average Body Mass Index of entire population
12. **Under five deaths (numeric) -** Number of under-five deaths per 1000 population
13. **Polio (numeric) -** Number of under-five deaths per 1000 population
14. **Total expenditure (numeric) -** General government expenditure on health as a percentage of total government expenditure (%)
15. **Diphtheria (numeric) -** Diphtheria tetanus toxoid and pertussis (DTP3) immunization coverage among 1-year-olds (%)
16. **HIV/AIDS (numeric) -** Deaths per 1 000 live births HIV/AIDS (0-4 years)
17. **GDP (numeric) -** Gross Domestic Product per capita (in USD)
18. **Population (numeric) -** Population of the country
19. **Thinness 1-19 years (numeric) -** Prevalence of thinness among children and adolescents for Age 10 to 19 (% )
20. **Thinness 5-9 years (numeric) -** Prevalence of thinness among children for Age 5 to 9(%)
21. **Income composition of resources (numeric) -** Human Development Index in terms of income composition of resources (index ranging from 0 to 1)
22. **Schooling (numeric) -** Number of years of Schooling(years)

We have used various data exploration and visualization techniques to explore the data that includes scatter plots, imputing missing variables, correlating the factors, conversion of categorical variables to numerical variables.

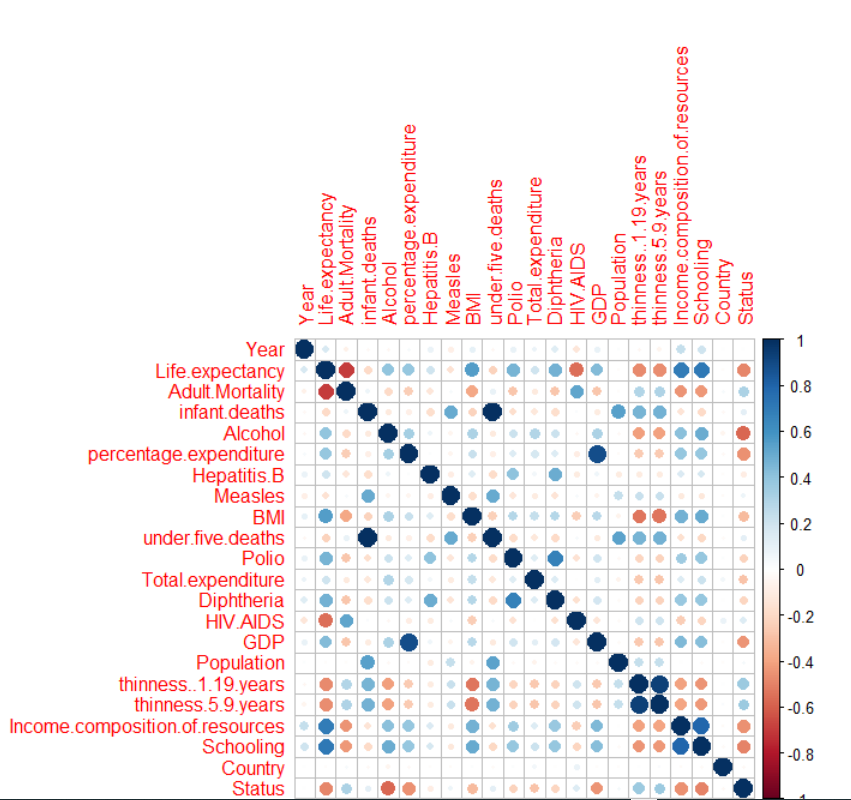
**Outlier Check:** We have observed that the dataset has 2 categorical variables and 20 numerical variables in the data set. In order to check the frequency of the data and remove the outliers, various plots with respect to output variable are shown below.



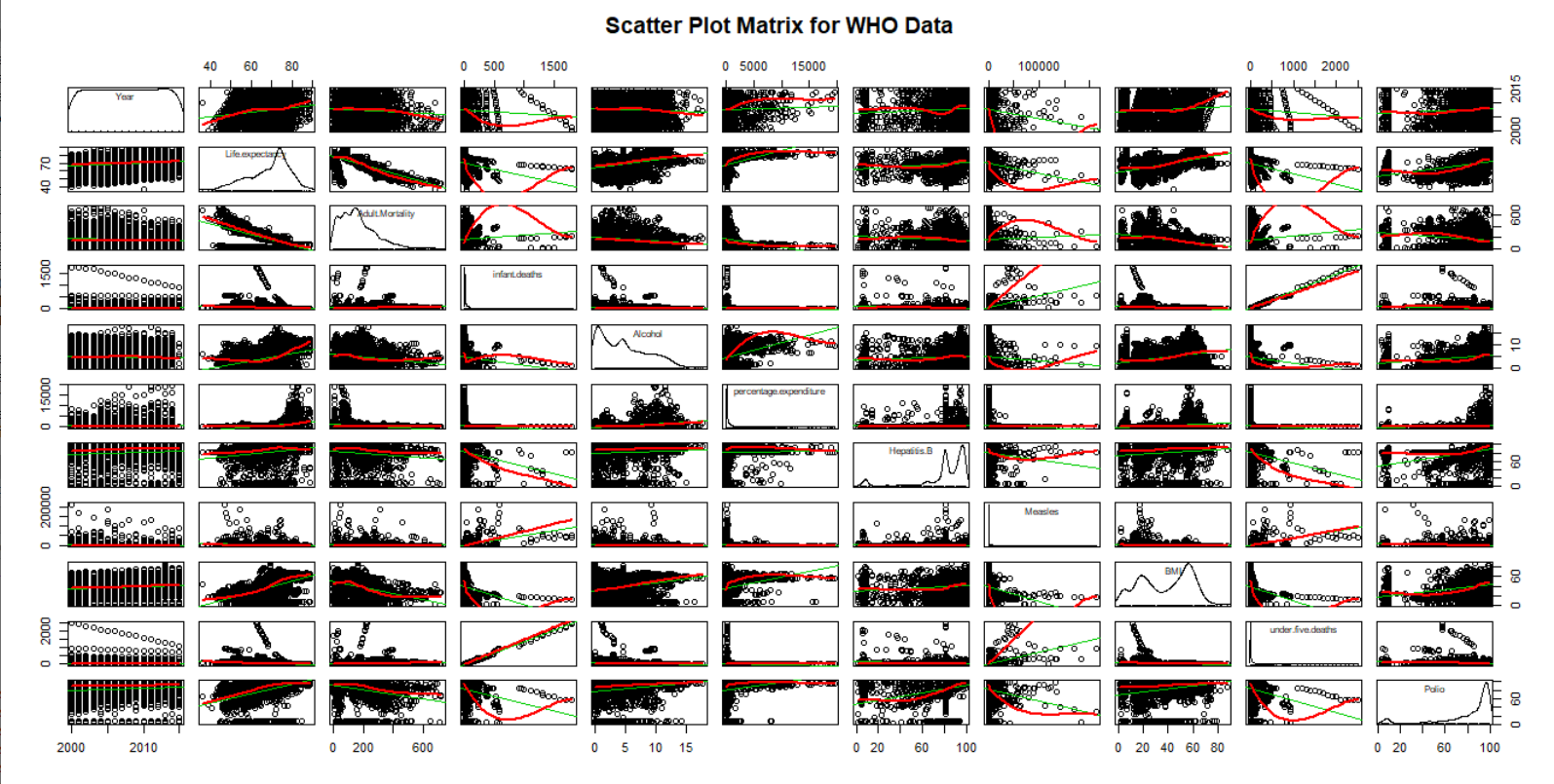
# **III. Data Preparation and Preprocessing**

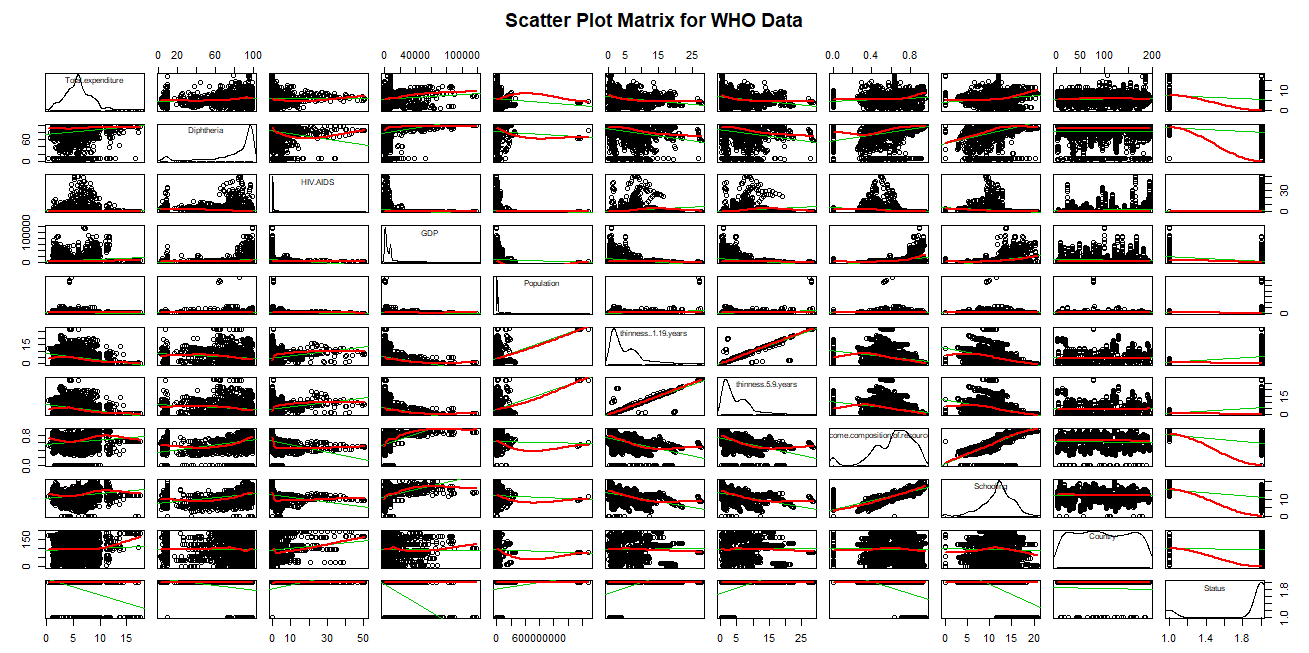
**Missing values:** From the data summary we found that we have quite a few missing values across different predictor variables in the data set. This was handled by imputing the corresponding mean values of the variables in the place of missing values.

**Correlation study:** To show the correlation between the variables we have created a correlation graph. Graph shows relationship between highly correlated values to Least correlated columns.

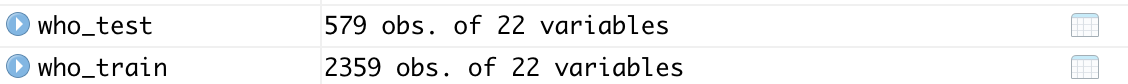


We also have created 2 scatter plot matrices where one shows the relation for first 11 columns and the second one shows the relation between the following columns.

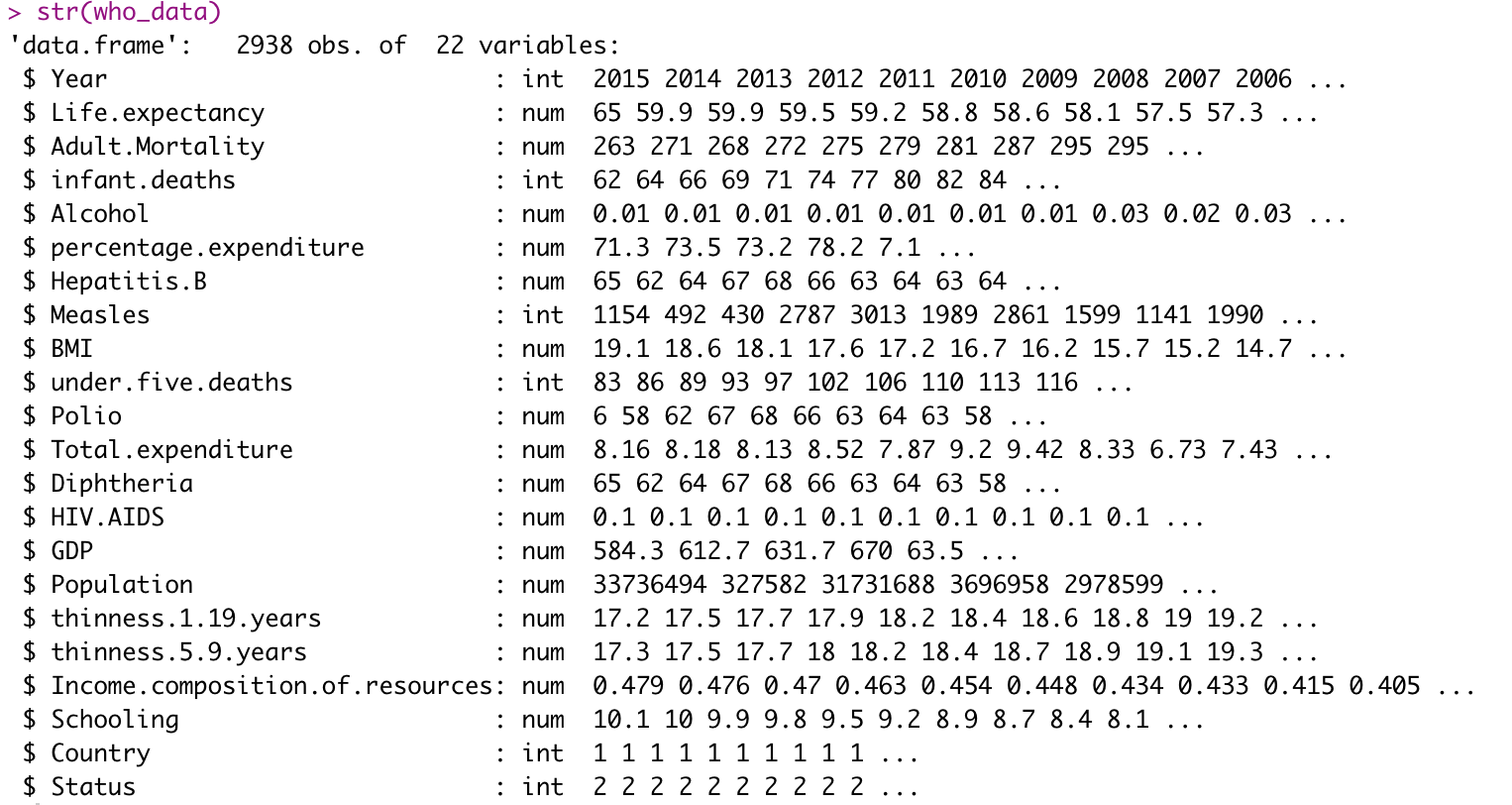




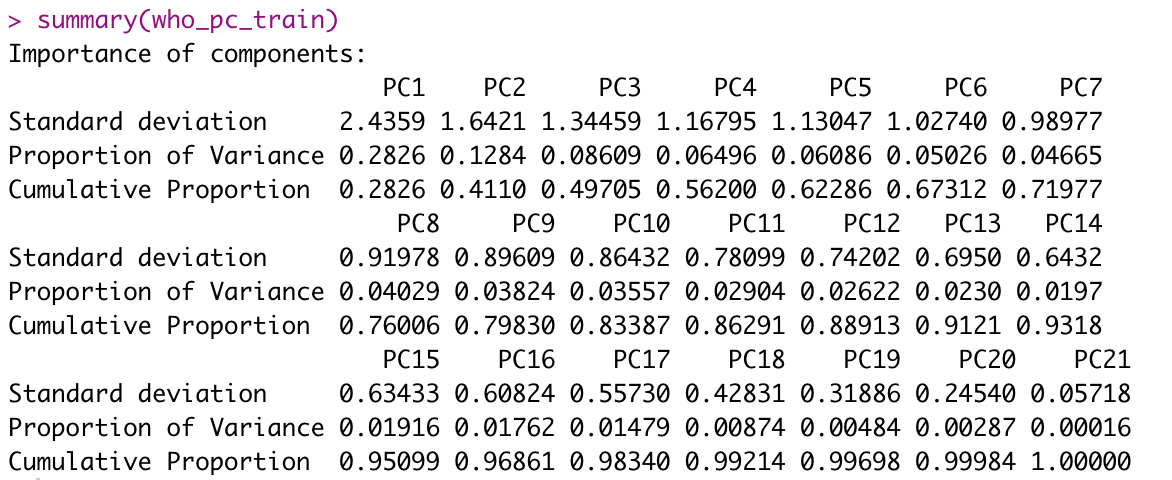
**Data partitioning:** The dataset was then partitioned as “80% - training” and “20% - testing” datasets which accounted to the below split up of records.



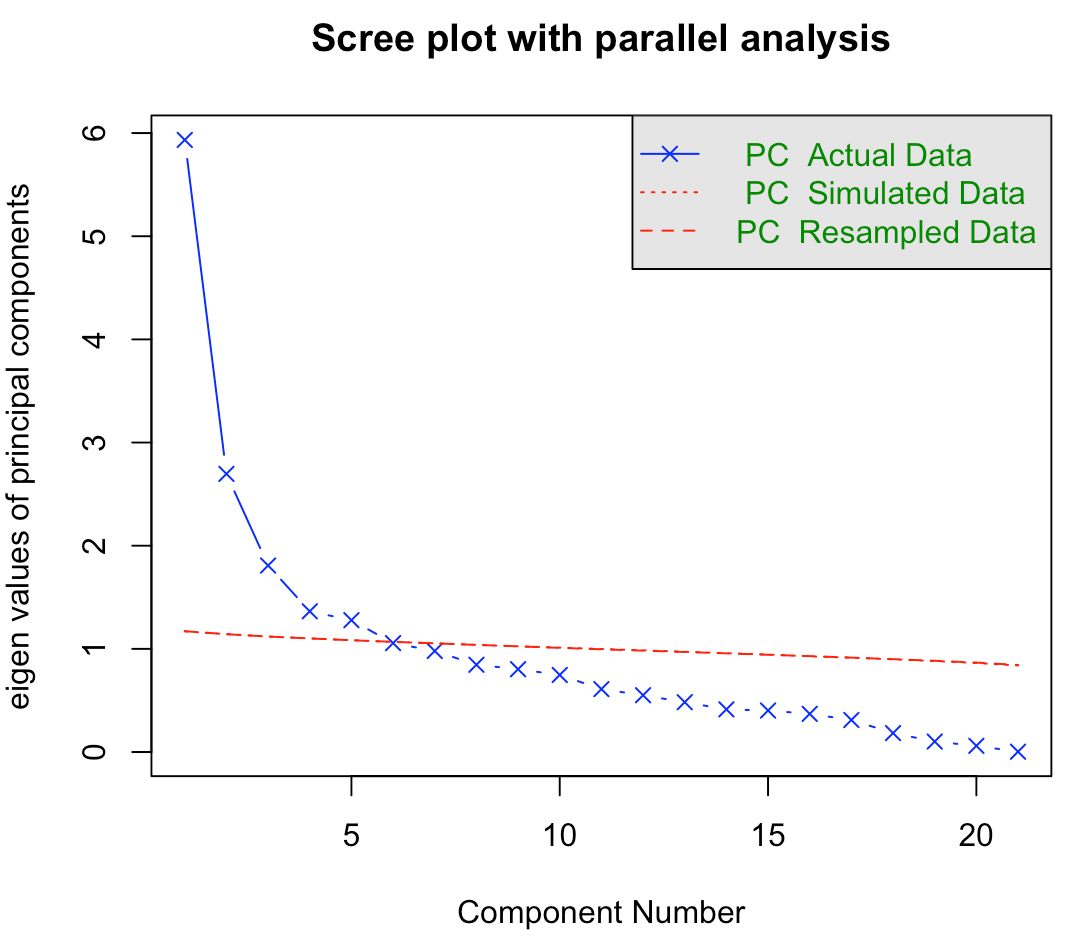
**Principal Component Analysis (PCA):** The dataset has 21 predictor variables and hence we have performed a Principal Component Analysis (PCA) to narrow them down to 5 variables (Principal Components) and those weights were predicted on the training and test data sets to get a new weighted dataset with just 5 variables/components. Since PCA can only be performed on numeric variables, the categorical variables in our dataset (Country and Status) were converted to numeric variables as seen below.



After doing a PCA using prcomp we get 21 Principal Components out of which the first 5 principal components (PC1 + PC2 + PC3 + PC4 + PC5) account for 63% of the total variation.



**Scree Plot with Parallel Analysis:** This was also supplemented by doing a Scree plot with parallel analysis test as seen below. The graph shows that we are good to select 5 factors (PC1 + PC2 + PC3 + PC4 + PC5) which are above the eigenvalue 1 line.

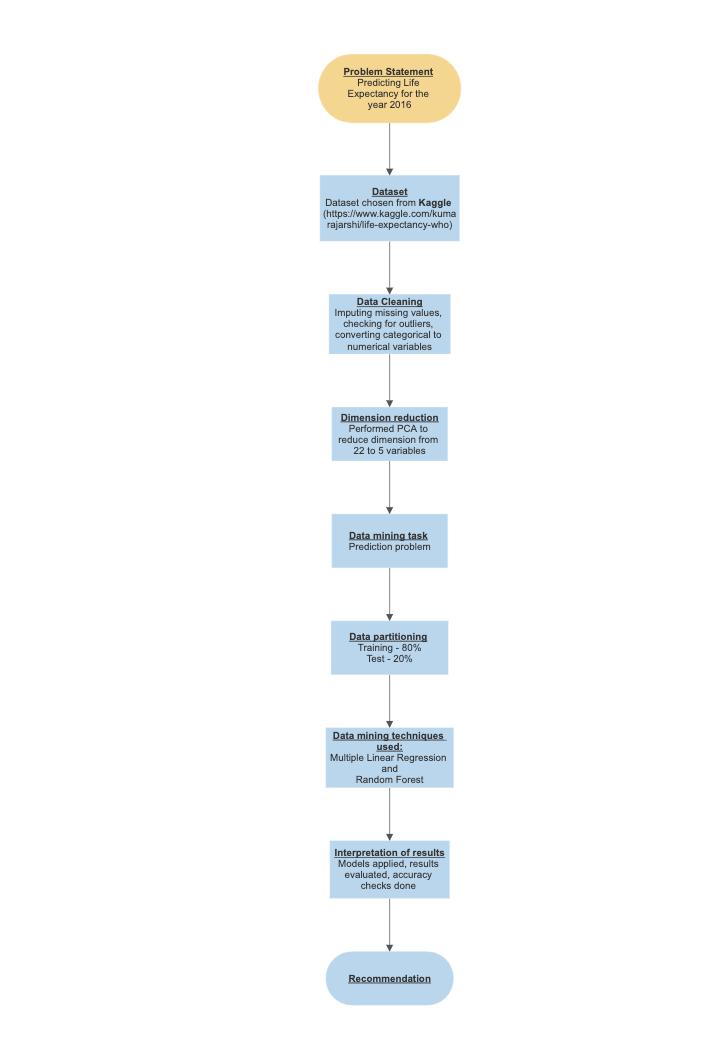


# 

PCA using “prcomp” function was done on both training and testing datasets. “prcomp” takes care of normalizing the data as we know that the data is of differing scales and there is a need for normalizing the data. The resultant component scores obtained were applied to their respective training and testing datasets using “predict” function to get the weighted values of the respective datasets.

# **IV. Data Mining Techniques and Implementation**

**Flow chart:** An overview of the entire process performed in this case study is shown in the below flow chart.

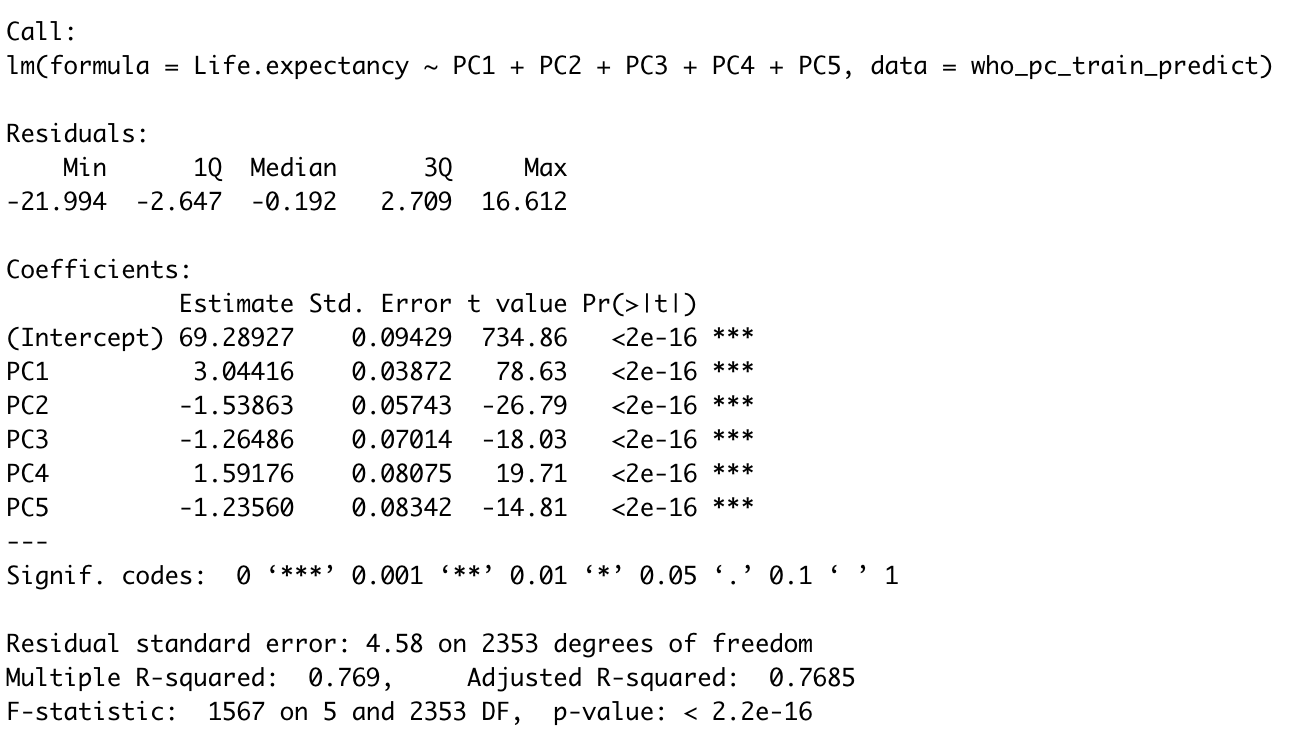


**Data Mining Techniques:** The problem was a prediction problem as we are predicting the life expectancy rates of the countries which was pretty straightforward. Hence, we decided to perform the following data mining techniques:

* Multiple linear regression
* Random forest

**Multiple Linear Regression:**

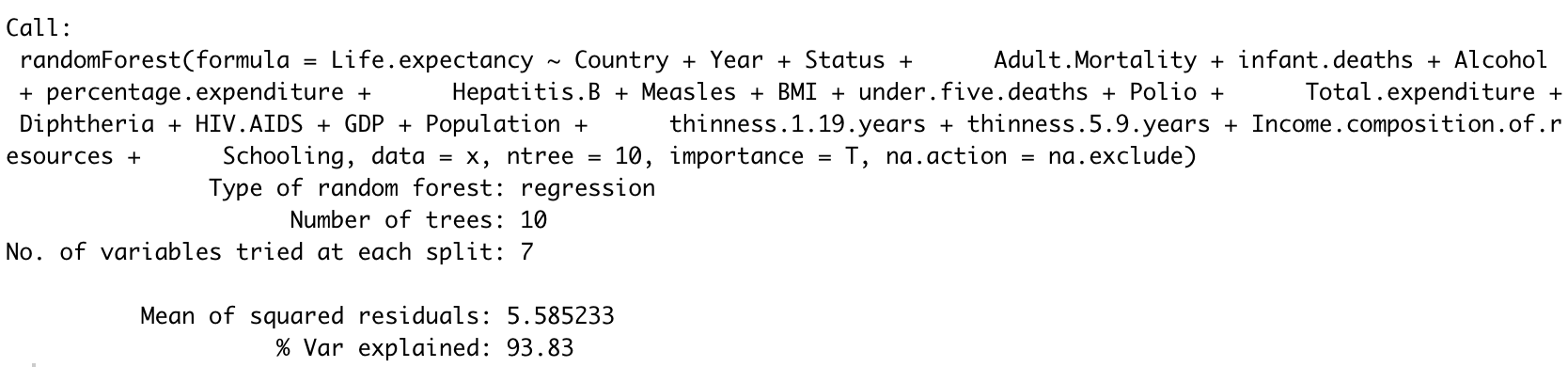
A multiple regression model was performed on the first 5 principal components as they accounted for most of the variability in the entire dataset and used to predict the values using training and test dataset.



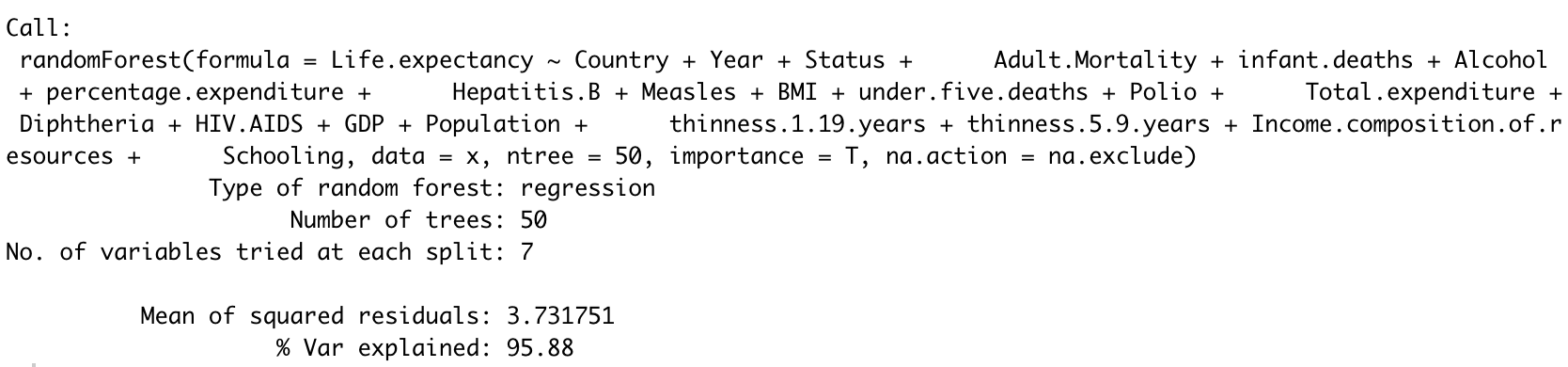
**Random Forest:**

Random forest model was run without any dimension reduction i.e all the predictor variables were considered. Different cases of random forest were performed based on different “number of trees” values in an attempt to fine tune the performance of the model.

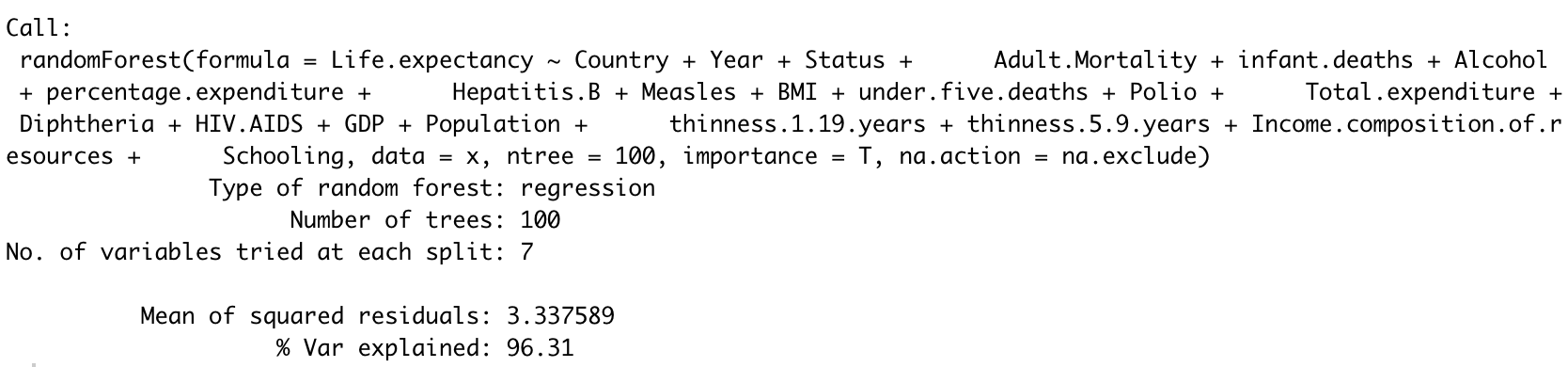
For n = 10,



For n = 50,



For n = 100,

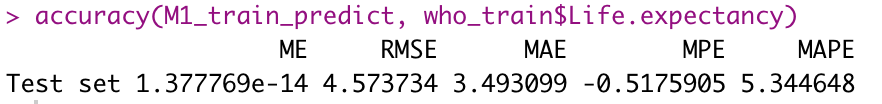


# **V. Performance Evaluation**

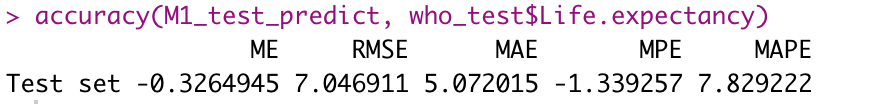
**Model 1: Multiple Linear Regression**

In order to check the performance of the model we used Prediction accuracy measures and applied it to both Training and Test data. The results for the multiple linear regression model are depicted below.

Training data accuracy,

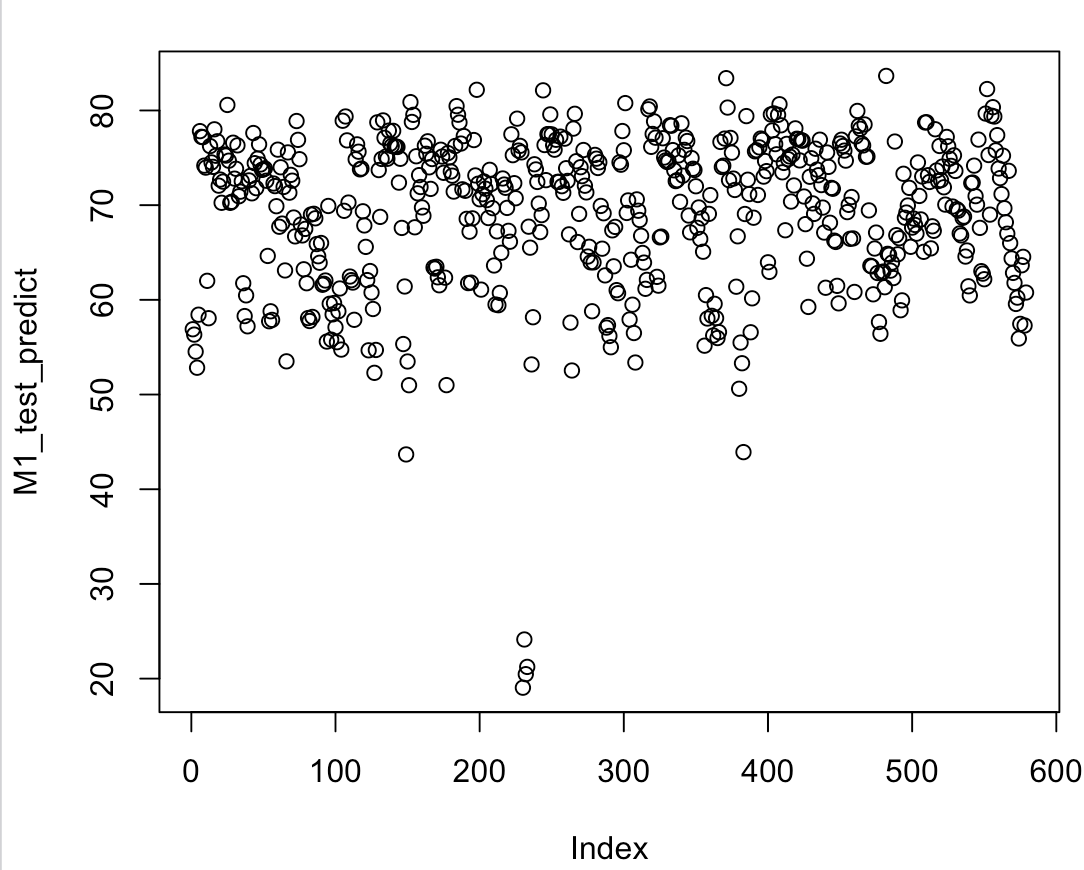
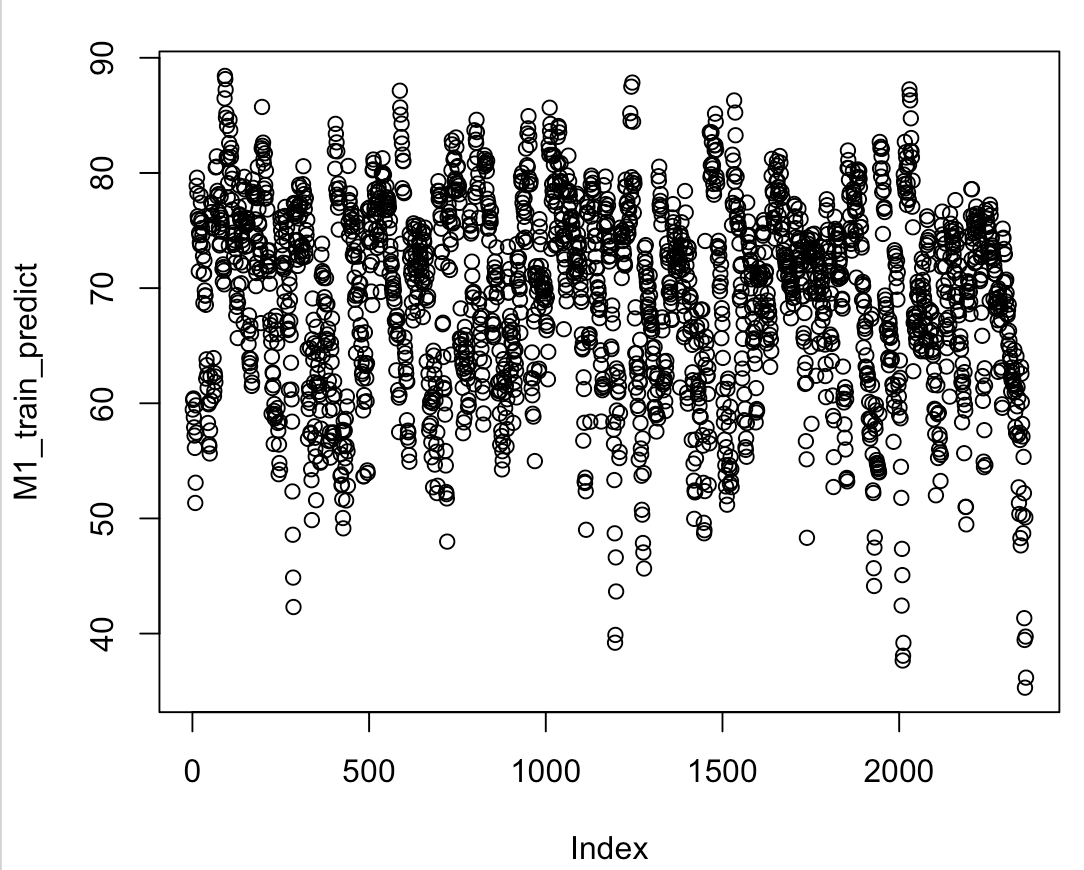


Test data accuracy,

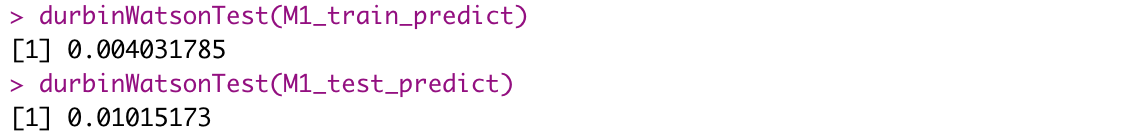


A few diagnostics were run on both the training and testing data predicted values.

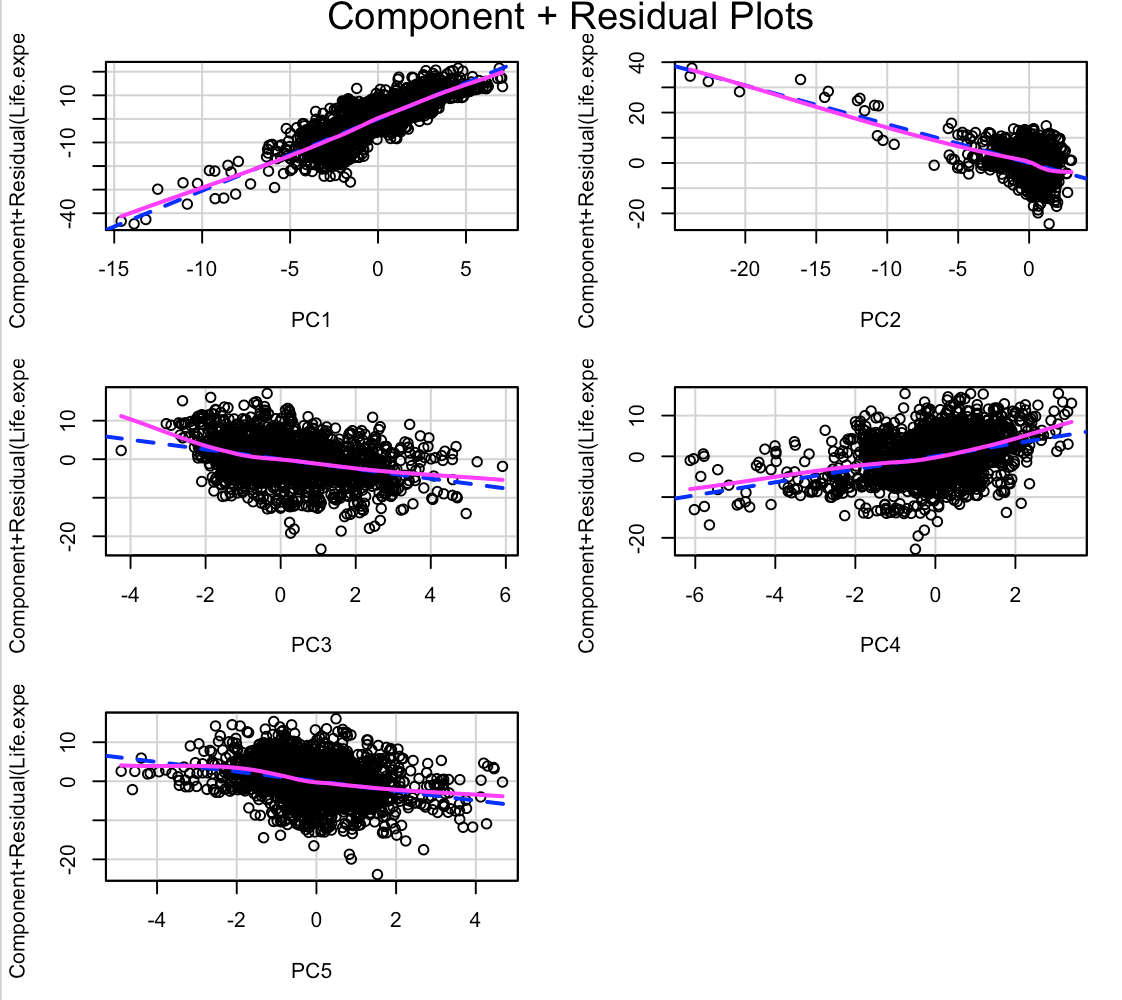
**Diagnostic 1: Predicted value plots**



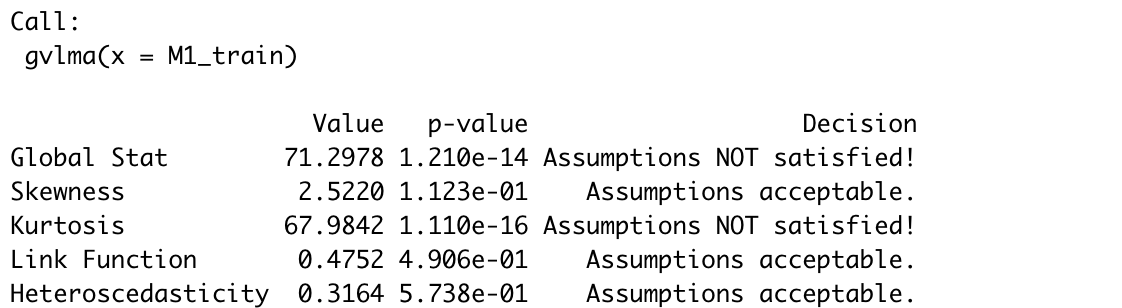
**Diagnostic 2: DurbinWatson Tests**



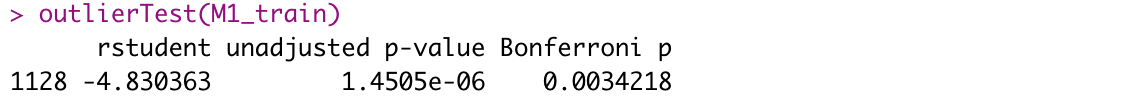
**Diagnostic 3: CR Plots:**

****

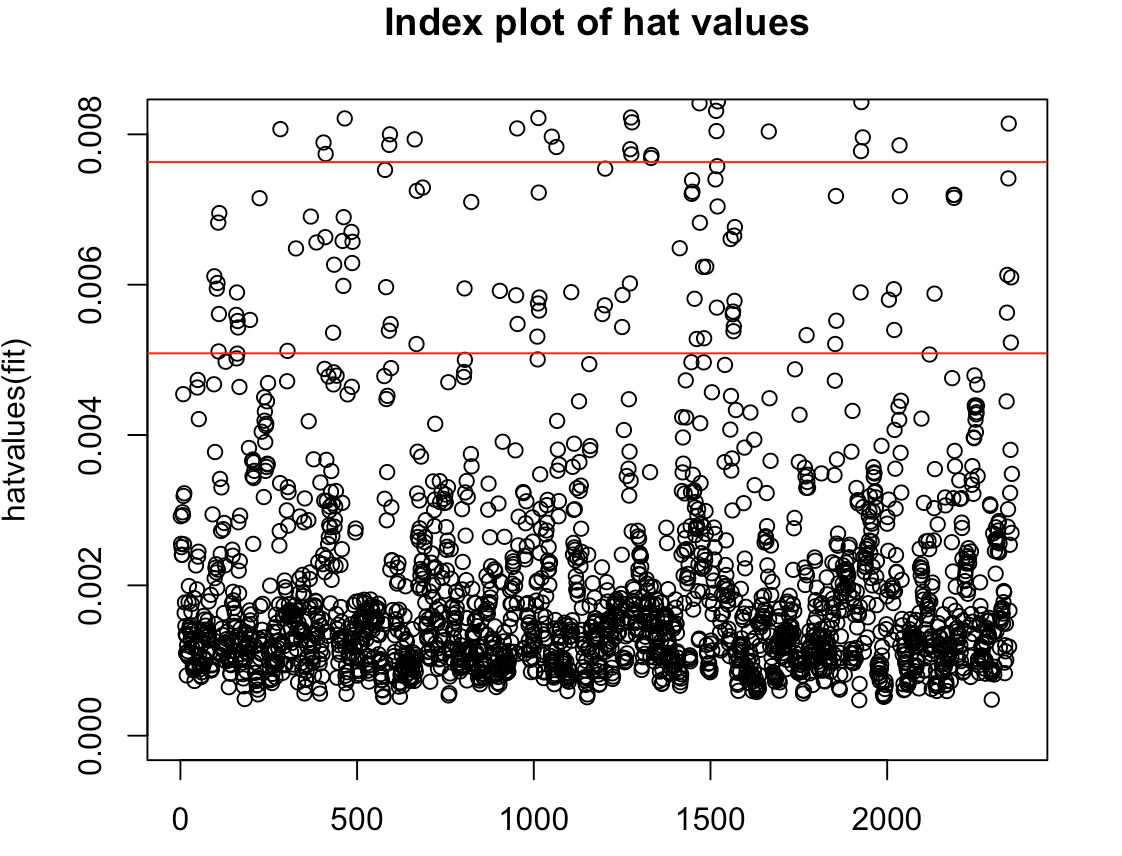
**Diagnostic 4: Global Validation of Linear Model Assumptions**

****

**Diagnostic 5: Outlier test**

****

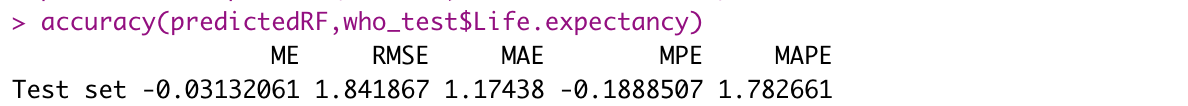
**Diagnostic 6: High leverage points**

****

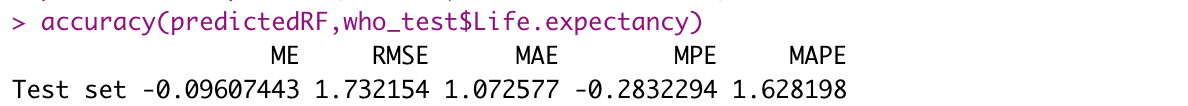
**Model 2: Random Forest**

We have also created a Random forest model using the training data set. In order to check the performance of the model we used Prediction accuracy measures and applied it to both Training and Test data. The results for the Random Forest model for different cases (n=10, n=50, n=100) are depicted below.

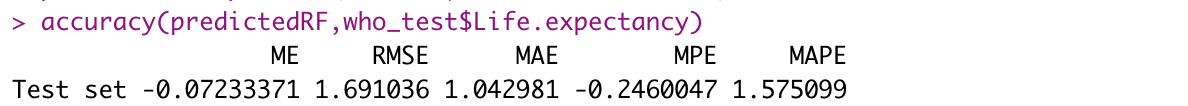
For n = 10,



For n = 50,



For n = 100,



The different “number of trees” values were considered in order to fine tune the model and it looks like the accuracy hits maximum when n = 100.

# **VI. Discussion and Recommendation**

Based on the prediction accuracy measures for both the models (Multiple Linear Regression and Random Forest), it is observed that Random Forest (with n = 100) has good accuracy with an RMSE (Root Mean Squared Error) value of 1.69. Hence, we decided to proceed with the Random FOrest model with number of trees as 100 for the dataset to predict Life Expectancy rate for all the countries for the year 2016.

# **VII. Summary**

We wanted to predict the Life Expectancy for all the countries for the year 2016. So first we imputed the missing values in the variables and divided the data into Training and Test data set (80:20). Then using PCA correlation analysis we have reduced the number of variables to 5 from 22.

A Multiple regression model is created and accuracy the model is validated using Predicted accuracy measures. In order to find the best suitable model we have also created the Random forest model and accuracy measures for both are compared from which we found that Random Forest model is more accurate to our data than Multiple regression model. Hence the predicted values of Life Expectancy for the year 2016 are obtained by using Random Forest Model.

# **Appendix: R Code for use case study**

---

title: "Analysis on Life Expectancy Rate for various countries"

output:

pdf\_document: default

html\_notebook: default

editor\_options:

chunk\_output\_type: console

---

# Loading the libraries

```{r}

library(dplyr)

library(dbplyr)

library(ggplot2)

library(corrplot)

library(scatterplot3d)

library(plotrix)

library(gridExtra)

library(psych)

library(lubridate)

library(BBmisc)

library(forecast)

library(randomForest)

library(ROCR)

library(MASS)

library(distr)

library(leaps)

library(bootstrap)

library(gvlma)

library(caret)

library(car)

library(lmtest)

```

# Loading the dataset

```{r}

who <- read.csv("C:/Users/shash/Desktop/Data mining/life expectancy project/Life Expectancy Data.csv")

head(who)

summary(who)

colnames(who)

```

# Replacing missing values with mean values

```{r}

who$Life.expectancy[is.na(who$Life.expectancy)] <- mean(who$Life.expectancy, na.rm = TRUE)

summary(who$Life.expectancy)

who$Adult.Mortality[is.na(who$Adult.Mortality)] <- mean(who$Adult.Mortality, na.rm = TRUE)

summary(who$Adult.Mortality)

who$Alcohol[is.na(who$Alcohol)] <- mean(who$Alcohol, na.rm = TRUE)

summary(who$Alcohol)

who$Hepatitis.B[is.na(who$Hepatitis.B)] <- mean(who$Hepatitis.B, na.rm = TRUE)

summary(who$Hepatitis.B)

who$BMI[is.na(who$BMI)] <- mean(who$BMI, na.rm = TRUE)

summary(who$BMI)

who$Polio[is.na(who$Polio)] <- mean(who$Polio, na.rm = TRUE)

summary(who$Polio)

who$Total.expenditure[is.na(who$Total.expenditure)] <- mean(who$Total.expenditure, na.rm = TRUE)

summary(who$Total.expenditure)

who$Diphtheria[is.na(who$Diphtheria)] <- mean(who$Diphtheria, na.rm = TRUE)

summary(who$Diphtheria)

who$GDP[is.na(who$GDP)] <- mean(who$GDP, na.rm = TRUE)

summary(who$GDP)

who$Population[is.na(who$Population)] <- mean(who$Population, na.rm = TRUE)

summary(who$Population)

who$thinness.1.19.years[is.na(who$thinness.1.19.years)] <- mean(who$thinness.1.19.years, na.rm = TRUE)

summary(who$thinness.1.19.years)

who$thinness.5.9.years[is.na(who$thinness.5.9.years)] <- mean(who$thinness.5.9.years, na.rm = TRUE)

summary(who$thinness.5.9.years)

who$Income.composition.of.resources[is.na(who$Income.composition.of.resources)] <- mean(who$Income.composition.of.resources, na.rm = TRUE)

summary(who$Income.composition.of.resources)

who$Schooling[is.na(who$Schooling)] <- mean(who$Schooling, na.rm = TRUE)

summary(who$Schooling)

# Summary of the imputed data

summary(who)

```

# Plotting the graphs to remove the outliers from the data

```{r}

plot1 <- ggplot(data = who, aes(x = who$Year ,y=who$Life.expectancy))+geom\_point()

plot2 <- ggplot(data = who, aes(x = who$Country ,y=who$Life.expectancy))+geom\_point()

plot3 <- ggplot(data = who, aes(x = who$Alcohol ,y=who$Life.expectancy))+geom\_point()

plot4 <- ggplot(data = who, aes(x = who$GDP ,y=who$Life.expectancy))+geom\_point()

plot5 <- ggplot(data = who, aes(x = who$Population ,y=who$Life.expectancy))+geom\_point()

grid.arrange(plot1,plot2,plot3,plot4,plot5, ncol=2)

```

# Converting categorical to numerical variables

```{r}

str(who)

who\_1<-sapply(who,is.factor)

who\_2<-sapply(who[,who\_1],unclass)

who\_data<-cbind(who[,!who\_1],who\_2)

who\_data

str(who\_data)

```

# Plot scatter plot matrix

```{r}

who\_data\_scatter1 <- who\_data[,(1:11)]

scatterplotMatrix(who\_data\_scatter1, spread = FALSE, lty.smooth = 2,

main = "Scatter Plot Matrix for WHO Data")

who\_data\_scatter2 <- who\_data[,(12:22)]

scatterplotMatrix(who\_data\_scatter2, spread = FALSE, lty.smooth = 2,

main = "Scatter Plot Matrix for WHO Data")

```

# Correlation plot

```{r}

who\_cor <- cor(who\_data)

who\_cor

corrplot(who\_cor)

```

# Data partition

```{r}

set.seed(123)

ind <- sample(2, nrow(who\_data),

replace = TRUE,

prob = c(0.8, 0.2))

who\_train <- who\_data[ind == 1,]

who\_test <- who\_data[ind == 2,]

who\_train

who\_test

```

# PCA - prcomp

```{r}

who\_pc\_train <- prcomp(who\_train[,-2],

center = TRUE,

scale. = TRUE)

attributes(who\_pc\_train)

who\_pc\_train$center

who\_pc\_train$scale

who\_pc\_train

summary(who\_pc\_train)

who\_pc\_train\_predict <- predict(who\_pc\_train, who\_train)

who\_pc\_train\_predict <- data.frame(who\_pc\_train\_predict, who\_train[2])

who\_pc\_train\_predict

who\_pc\_test <- prcomp(who\_test[,-2],

center = TRUE,

scale. = TRUE)

attributes(who\_pc\_test)

who\_pc\_test$center

who\_pc\_test$scale

who\_pc\_test

summary(who\_pc\_test)

who\_pc\_test\_predict <- predict(who\_pc\_test, who\_test)

who\_pc\_test\_predict <- data.frame(who\_pc\_test\_predict, who\_test[2])

who\_pc\_test\_predict

```

# Multiple Linear Regression and diagnostics

```{r}

M1\_train <- lm(Life.expectancy ~ PC1 + PC2 + PC3 + PC4 + PC5,

data = who\_pc\_train\_predict)

M1\_train

summary(M1\_train)

# Model prediction - training

M1\_train\_predict <- predict(M1\_train, who\_pc\_train\_predict)

M1\_train\_predict

M1\_train\_predict\_table <- table(M1\_train\_predict, who\_pc\_train\_predict$Life.expectancy)

M1\_train\_predict\_table

# Model prediction - testing

M1\_test\_predict <- predict(M1\_train, who\_pc\_test\_predict)

M1\_test\_predict\_table <- table(M1\_test\_predict, who\_pc\_test\_predict$Life.expectancy)

M1\_test\_predict\_table

# Training Accuracy

some.residuals <- who\_train$Life.expectancy - M1\_train\_predict

some.residuals

data.frame(M1\_train\_predict, who\_train$Life.expectancy, some.residuals)

accuracy(M1\_train\_predict, who\_train$Life.expectancy)

# Testing Accuracy

some.residuals <- who\_test$Life.expectancy - M1\_test\_predict

some.residuals

data.frame(M1\_test\_predict, who\_test$Life.expectancy, some.residuals)

accuracy(M1\_test\_predict, who\_test$Life.expectancy)

# Diagnostics

par(mfrow = c(2,2))

plot(M1\_train\_predict)

plot(M1\_test\_predict)

durbinWatsonTest(M1\_train\_predict)

durbinWatsonTest(M1\_test\_predict)

crPlots(M1\_train)

gvlma(M1\_train)

outlierTest(M1\_train)

hatplot <- function(fit){

p = length(coefficients(fit))

n = length(fitted(fit))

plot(hatvalues(fit), ylim = c(0,3.2)\*p/n, main = "Index plot of hat values")

abline(h = c(2,3)\*p/n, col = "red", Ity = 2)

identify(1:n, hatvalues(fit), names(hatvalues(fit)))

}

hatplot(M1\_train)

```

# Random forest and diagnostics

```{r}

# For ntree = 10

randomForestAlgo <- function(x){

rf <- randomForest(Life.expectancy ~ Country +

Year + Status + Adult.Mortality + infant.deaths + Alcohol +

percentage.expenditure + Hepatitis.B + Measles + BMI +

under.five.deaths + Polio + Total.expenditure + Diphtheria +

HIV.AIDS + GDP + Population + thinness.1.19.years + thinness.5.9.years +

Income.composition.of.resources + Schooling , ntree=10, na.action =

na.exclude, data=x, importance=T)

return(rf)

}

rfModel = randomForestAlgo(who\_train)

print(rfModel)

predictedRF<-predict(rfModel,newdata = who\_test)

# Accuracy for ntree = 10

accuracy(predictedRF,who\_test$Life.expectancy)

# For ntree = 50

randomForestAlgo <- function(x){

rf <- randomForest(Life.expectancy ~ Country +

Year + Status + Adult.Mortality + infant.deaths + Alcohol +

percentage.expenditure + Hepatitis.B + Measles + BMI +

under.five.deaths + Polio + Total.expenditure + Diphtheria +

HIV.AIDS + GDP + Population + thinness.1.19.years + thinness.5.9.years +

Income.composition.of.resources + Schooling , ntree=50, na.action =

na.exclude, data=x, importance=T)

return(rf)

}

rfModel = randomForestAlgo(who\_train)

print(rfModel)

predictedRF<-predict(rfModel,newdata = who\_test)

# Accuracy for ntree = 50

accuracy(predictedRF,who\_test$Life.expectancy)

# For ntree = 100

randomForestAlgo <- function(x){

rf <- randomForest(Life.expectancy ~ Country +

Year + Status + Adult.Mortality + infant.deaths + Alcohol +

percentage.expenditure + Hepatitis.B + Measles + BMI +

under.five.deaths + Polio + Total.expenditure + Diphtheria +

HIV.AIDS + GDP + Population + thinness.1.19.years + thinness.5.9.years +

Income.composition.of.resources + Schooling , ntree=100, na.action =

na.exclude, data=x, importance=T)

return(rf)

}

rfModel = randomForestAlgo(who\_train)

print(rfModel)

predictedRF<-predict(rfModel,newdata = who\_test)

# Accuracy for ntree = 100

accuracy(predictedRF,who\_test$Life.expectancy)

```