greatlearning

Learning for Life

Capstone Project

Prediction of Life Expectancy

Submitted by : Shilpa Pandey
Date : 4/5/2020

Acknowledgement

The success and final outcome of this project required a lot of guidance and assistance from many people and I am extremely privileged to have got this all along the completion of my project. All that I have done is only due to such supervision and assistance and I would not forget to thank them.

I owe my deep gratitude to our project guide Surya, who took keen interest on our project work and guided us all along, till the completion of our project work by providing all the necessary information for overcoming our challenges,

I respect and thank Mrs. Karuna for giving all support and guidance which made me complete the project duly. I am extremely thankful to her for mentoring the entire Business analytics concepts. Also. I would like to thank the great learning team for immense support throughout.

Abstract:

Life expectancy is the average number of additional years that a person of a particular age can expect to live. Life expectancy depends on various demographic, geographic and health parameters. In order to do this, we need to prepare a model which can work for regression data and before that we need to process data and check for outliers and missing values. The dataset should contain set of training and test data and after checking the performance of the model we need to select the best model and validate the model on the test data. Once we choose the model, we need to identify the factory which are significant for predicting life expectancy across the countries and also, we need to recommend the client what measures can be taken in order to improve life expectancy.

Conclusion from Data Analysis : this Data has lot of missing values, outliers and inconsistencies and hence data treatment is very must here before modeling the data.
Variables such as BMI which is not giving correct output should be removed. BMI should decrease for high life expectancy but it is vice versa.
Variable like population which is numerical should be important for predicting life expectancy but here it is not and this can be further checked.
Increase in alcohol consumption increases life expectancy which is quite strange and is collinear with number of years in school. This variable can be further analyzed.

The most significant variables for determining life expectancy are:

1. Income composition of resources: As more the income composition per capita across countries, the life expectancy is also increased.
2. Adult Mortality: If Adult Mortality is high then life expectancy will be less.
3. HIV.AIDS: If the number of HIV percentage is more the life expectancy will be less.
4. Schooling : studies shows that more the number of schooling years, more the life expectancy.
5. Infant deaths: In order to increase life expectancy , the health and immunity should be taken care of and infant death should be less.

6.3. Recommendations to business to improve life Expectancy

1. Countries need to work on improving the overall economy (GDP and income composition) of the country which will enhance the income and standard of living of the people and further will improve life expectancy.
2. Improve the health conditions by spending enough amount on healthcare, vaccination and safety against malnutrition.
3. Spreading awareness for diseases like HIV which has no cure and decreases life expectancy.
4. Work toward improving the literacy rate of the country. It has been observed that countries with high literacy rate has high life expectancy.

Table of Contents

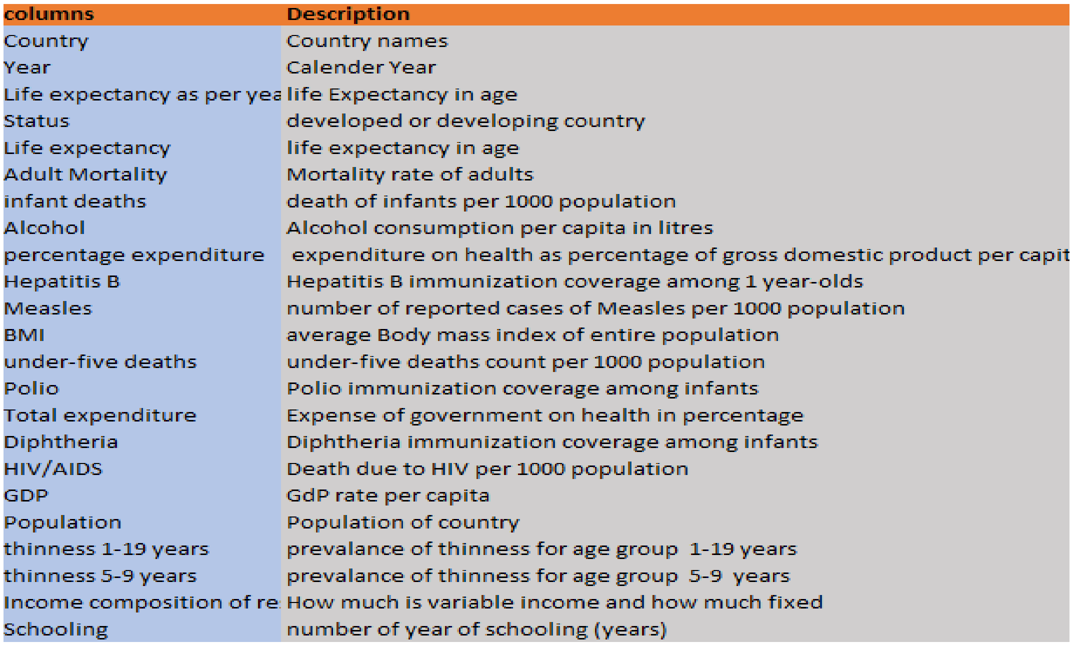
1. Introduction

1.1. Problem Statement

Life expectancy is the average number of additional years that a person of a particular age can expect to live. Life expectancy depends on various demographic, geographic and health parameters. Predicting life expectancy and the factors influencing life expectancy has relevance in several domains ranging from healthcare to insurance. It is also equally important to understand the factors that influence life expectancy so that these factors can be worked upon to enhance life expectancy of a person.

For this project, the data-set related to life expectancy, health factors for 193 countries has been collected from the same WHO data repository website and its corresponding economic data was collected from United Nation website. Among all categories of health-related factors only those critical factors were chosen which are more representative.

1.2. Data Dictionary



columns	Description
Country	Country names
Year	Calender Year
Life expectancy as per yea	life Expectancy in age
Status	developed or developing country
Life expectancy	life expectancy in age
Adult Mortality	Mortality rate of adults
infant deaths	death of infants per 1000 population
Alcohol	Alcohol consumption per capita in litres
percentage expenditure	expenditure on health as percentage of gross domestic product per capit
Hepatitis B	Hepatitis B immunization coverage among 1 year-olds
Measles	number of reported cases of Measles per 1000 population
BMI	average Body mass index of entire population
under-five deaths	under-five deaths count per 1000 population
Polio	Polio immunization coverage among infants
Total expenditure	Expense of government on health in percentage
Diphtheria	Diphtheria immunization coverage among infants
HIV/AIDS	Death due to HIV per 1000 population
GDP	GdP rate per capita
Population	Population of country
thinness 1-19 years	prevalance of thinness for age group 1-19 years
thinness 5-9 years	prevalance of thinness for age group 5-9 years
Income composition of re	How much is variable income and how much fixed
Schooling	number of year of schooling (years)

1.3. Objective and scope of the Project

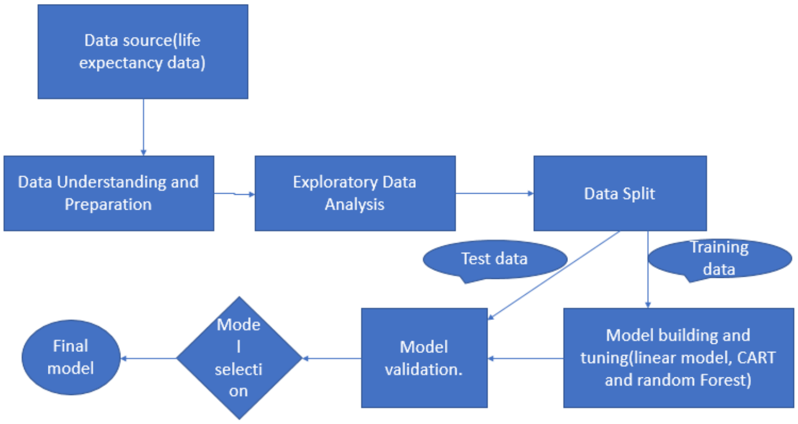
a) come up with a model to predict average Life expectancy of a person which depends on various demographic, geographic and health parameters

b) come up with at least 5 most important variables which is important in determining life expectancy

c) Find out factors/variables that can be worked upon to enhance life expectancy of a person.

1.4. Model Flow Chart

Here is the flow chart for regression Model:


Data source(life expectancy data)
Data Understanding and Preparation
Exploratory Data Analysis
Data Split
Test data
Training data
Model building and tuning(linear model, CART and random Forest)
Model validation.
Model selection
Final model

1.5. Tools and Techniques

The Analytics tools used are R studio, Tableau and MS excel

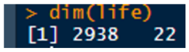
The Modelling techniques used are Linear Model, Random Forest model and decision tree model.

2. Data Understanding and Preparation

2.1. Data Description

Number of rows: **2938**

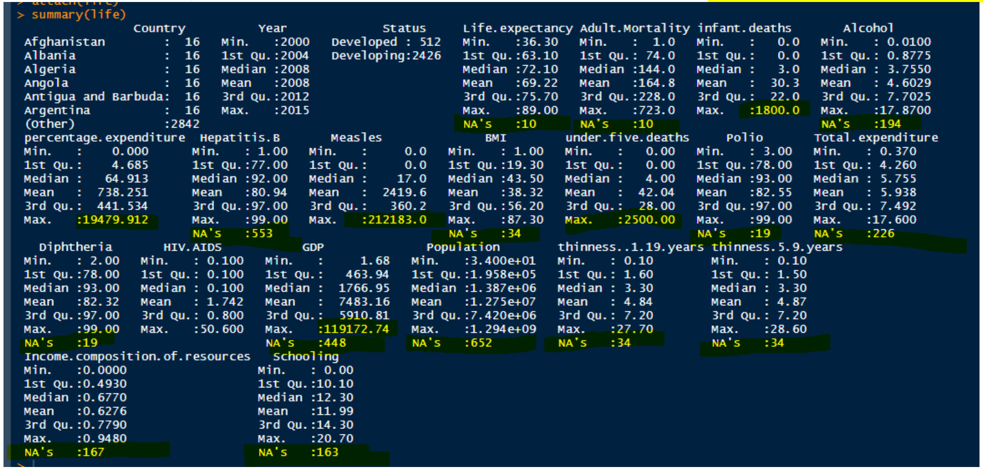
Number of columns: **22**



```
> dim(life)
[1] 2938   22
```

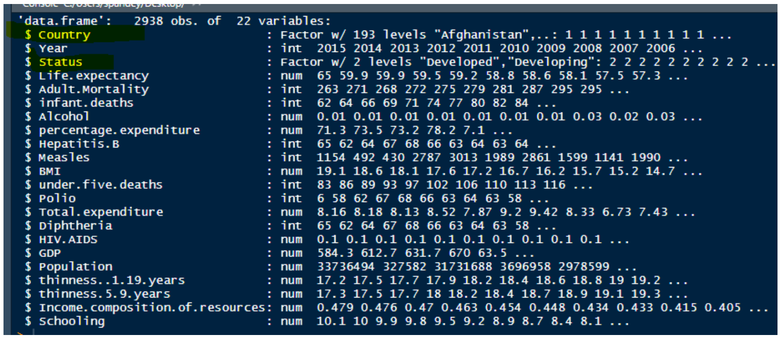
The summary command gives the description of the data in data set.
It gives the minimum value, max value , mean value, median value, 1st Quartile value and 3rd Quartile value of the data set. Also it gives the count of “na” values. It also helps in determining “outliers”.

From below many variables have “NA” values. Also variables like measeles, GDP,under five death has outliers.



```
> summary(life)
                  Country          Year            Status      Life.expectancy Adult.Mortality infant.deaths        Alcohol      
 Afghanistan        :  16   Min.   :2000   Developed : 512   Min.   :36.30   Min.   :  1.0   Min.   :   0.0   Min.   : 0.0100  
 Albania            :  16   1st Qu.:2004   Developing:2426   1st Qu.:63.10   1st Qu.: 74.0   1st Qu.:   0.0   1st Qu.: 0.8775  
 Algeria            :  16   Median :2008                     Median :72.10   Median :144.0   Median :   3.0   Median : 3.7550  
 Angola             :  16   Mean   :2008                     Mean   :69.22   Mean   :164.8   Mean   :  30.3   Mean   : 4.6029  
 Antigua and Barbuda:  16   3rd Qu.:2012                     3rd Qu.:75.70   3rd Qu.:228.0   3rd Qu.:  22.0   3rd Qu.: 7.7025  
 Argentina          :  16   Max.   :2015                     Max.   :89.00   Max.   :723.0   Max.   :1800.0   Max.   :17.8700  
 (Other)            :2842                                    NA's   :10      NA's   :10                       NA's   :194      
 percentage.expenditure  Hepatitis.B       Measles               BMI        under.five.deaths     Polio       Total.expenditure
 Min.   :    0.000      Min.   : 1.00   Min.   :     0.0   Min.   : 1.00   Min.   :   0.00   Min.   : 3.00   Min.   : 0.370   
 1st Qu.:    4.685      1st Qu.:77.00   1st Qu.:     0.0   1st Qu.:19.30   1st Qu.:   0.00   1st Qu.:78.00   1st Qu.: 4.260   
 Median :   64.913      Median :92.00   Median :    17.0   Median :43.50   Median :   4.00   Median :93.00   Median : 5.755   
 Mean   :  738.251      Mean   :80.94   Mean   :  2419.6   Mean   :38.32   Mean   :  42.04   Mean   :82.55   Mean   : 5.938   
 3rd Qu.:  441.534      3rd Qu.:97.00   3rd Qu.:   360.2   3rd Qu.:56.20   3rd Qu.:  28.00   3rd Qu.:97.00   3rd Qu.: 7.492   
 Max.   :19479.912      Max.   :99.00   Max.   :212183.0   Max.   :87.30   Max.   :2500.00   Max.   :99.00   Max.   :17.600   
                        NA's   :553                        NA's   :34                        NA's   :19      NA's   :226      
   Diphtheria       HIV.AIDS            GDP              Population        thinness..1.19.years thinness.5.9.years
 Min.   : 2.00   Min.   : 0.100   Min.   :     1.68   Min.   :3.400e+01   Min.   : 0.10        Min.   : 0.10     
 1st Qu.:78.00   1st Qu.: 0.100   1st Qu.:   463.94   1st Qu.:1.958e+05   1st Qu.: 1.60        1st Qu.: 1.50     
 Median :93.00   Median : 0.100   Median :  1766.95   Median :1.387e+06   Median : 3.30        Median : 3.30     
 Mean   :82.32   Mean   : 1.742   Mean   :  7483.16   Mean   :1.275e+07   Mean   : 4.84        Mean   : 4.87     
 3rd Qu.:97.00   3rd Qu.: 0.800   3rd Qu.:  5910.81   3rd Qu.:7.420e+06   3rd Qu.: 7.20        3rd Qu.: 7.20     
 Max.   :99.00   Max.   :50.600   Max.   :119172.74   Max.   :1.294e+09   Max.   :27.70        Max.   :28.60     
 NA's   :19                       NA's   :448         NA's   :652         NA's   :34           NA's   :34        
 Income.composition.of.resources   Schooling    
 Min.   :0.0000                  Min.   : 0.00  
 1st Qu.:0.4930                  1st Qu.:10.10  
 Median :0.6770                  Median :12.30  
 Mean   :0.6276                  Mean   :11.99  
 3rd Qu.:0.7790                  3rd Qu.:14.30  
 Max.   :0.9480                  Max.   :20.70  
 NA's   :167                     NA's   :163    
```

The dataset have almost all continuous numerical variables.
Only country and status are factor variables. However we will remove country from modeling as it can not be treated as factor as there are 193 Countries.



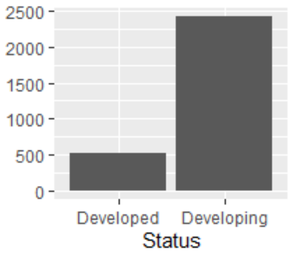
```
'data.frame':	2938 obs. of  22 variables:
 $ Country                        : Factor w/ 193 levels "Afghanistan",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Year                           : int  2015 2014 2013 2012 2011 2010 2009 2008 2007 2006 ...
 $ Status                         : Factor w/ 2 levels "Developed","Developing": 2 2 2 2 2 2 2 2 2 2 ...
 $ Life.expectancy                : num  65 59.9 59.9 59.5 59.2 58.8 58.6 58.1 57.5 57.3 ...
 $ Adult.Mortality                : int  263 271 268 272 275 279 281 287 295 295 ...
 $ infant.deaths                  : int  62 64 66 69 71 74 77 80 82 84 ...
 $ Alcohol                        : num  0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.03 0.02 0.03 ...
 $ percentage.expenditure         : num  71.3 73.5 73.2 78.2 7.1 ...
 $ Hepatitis.B                    : int  65 62 64 67 68 66 63 64 63 64 ...
 $ Measles                        : int  1154 492 430 2787 3013 1989 2861 1599 1141 1990 ...
 $ BMI                            : num  19.1 18.6 18.1 17.6 17.2 16.7 16.2 15.7 15.2 14.7 ...
 $ under.five.deaths              : int  83 86 89 93 97 102 106 110 113 116 ...
 $ Polio                          : int  6 58 62 67 68 66 63 64 63 58 ...
 $ Total.expenditure              : num  8.16 8.18 8.13 8.52 7.87 9.2 9.42 8.33 6.73 7.43 ...
 $ Diphtheria                     : int  65 62 64 67 68 66 63 64 63 58 ...
 $ HIV.AIDS                       : num  0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 ...
 $ GDP                            : num  584.3 612.7 631.7 670 63.5 ...
 $ Population                     : num  33736494 327582 31731688 3696958 2978599 ...
 $ thinness..1.19.years           : num  17.2 17.5 17.7 17.9 18.2 18.4 18.6 18.8 19 19.2 ...
 $ thinness.5.9.years             : num  17.3 17.5 17.7 18 18.2 18.4 18.7 18.9 19.1 19.3 ...
 $ Income.composition.of.resources: num  0.479 0.476 0.47 0.463 0.454 0.448 0.434 0.433 0.415 0.405 ...
 $ Schooling                      : num  10.1 10 9.9 9.8 9.5 9.2 8.9 8.7 8.4 8.1 ...
```

2.2. Data Preparation

2.2.1. Univariate Analysis: it explains the spread of the data

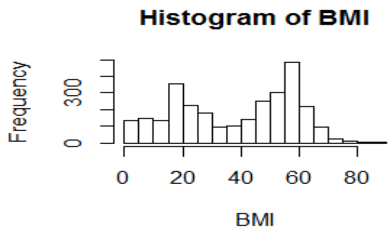
Univariate Analysis for Categorical variables

Number of developing countries are much higher than no. of developed countries

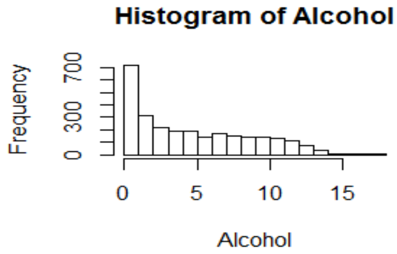


histogram for numerical variables

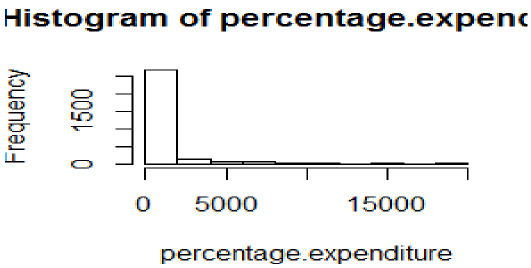
Normal distribution of variable BMI:

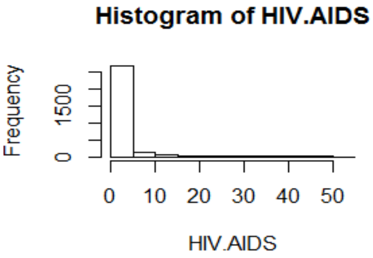


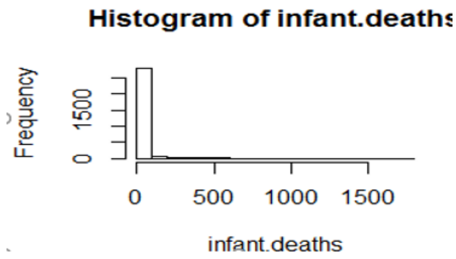
Distribution of variable Alcohol.



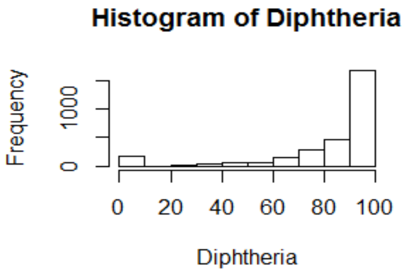
Percentage expenditures, HIV.AIDs and infant death distribution is right skewed.

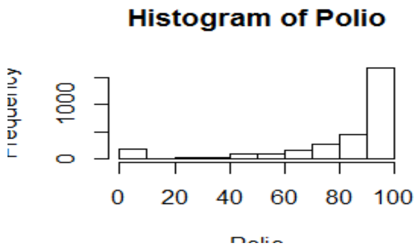


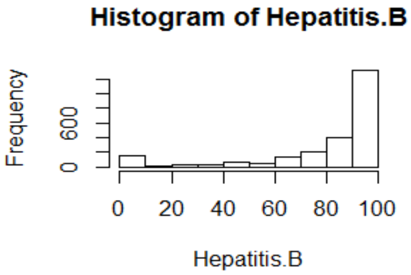




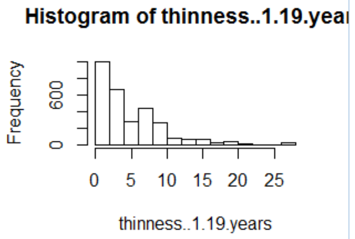
Diphtheria immunization,polio and hepatitis B distribution is left skewed.

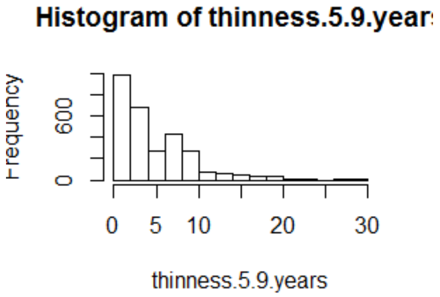


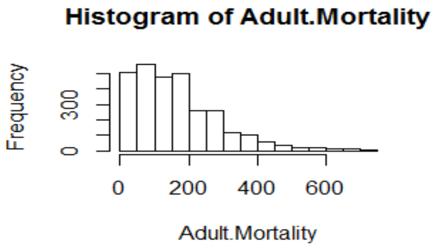




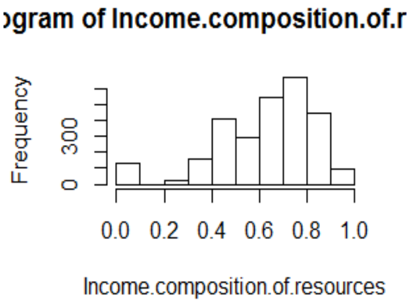
thinness..1.19.years, thinness.5.9.years and adult mortality is somewhat right skewed.

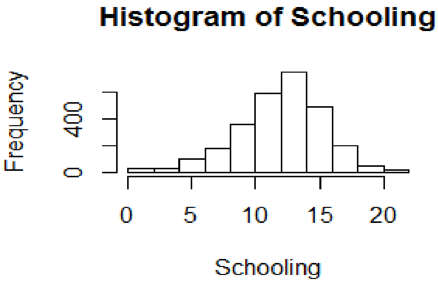


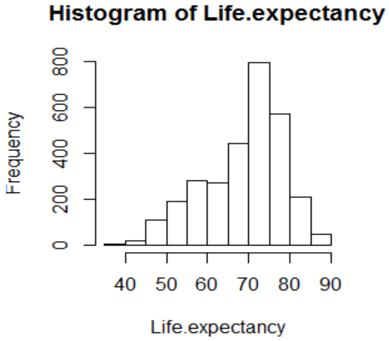




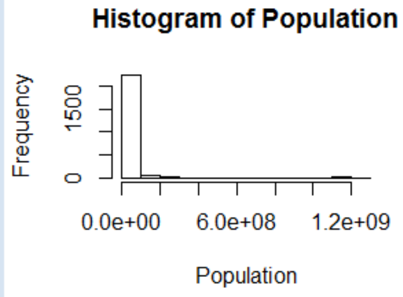
Income.composition.of.resources, Schooling and life expectancy has normal distribution.

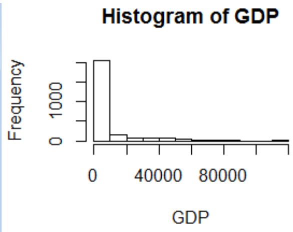


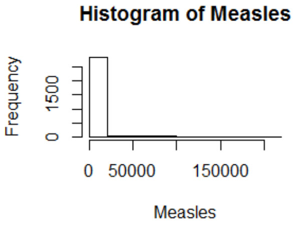




Population, measles and GDP has right skewed distribution



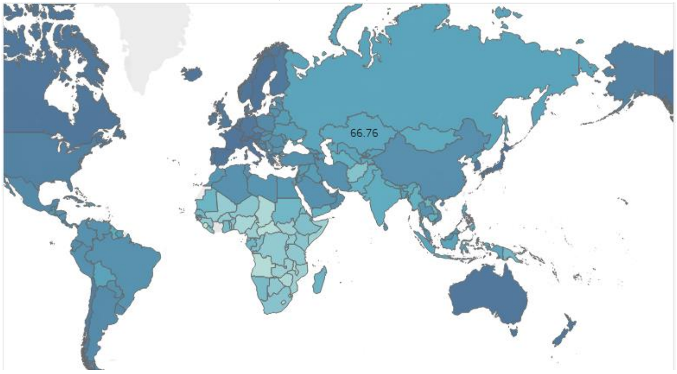


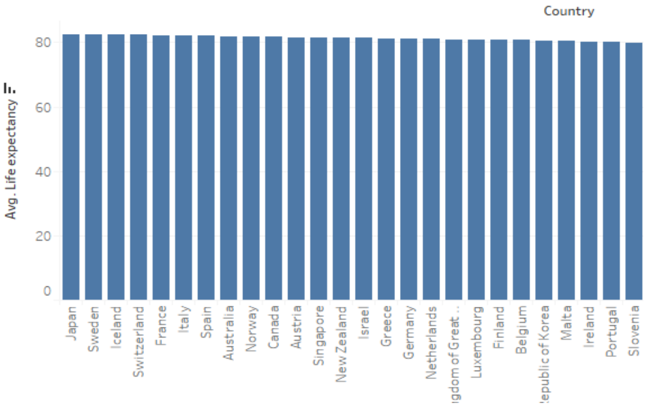


2.2.2. Bivariate Analysis: this analysis helps in finding how dependent variable behaves with respect to the target variable

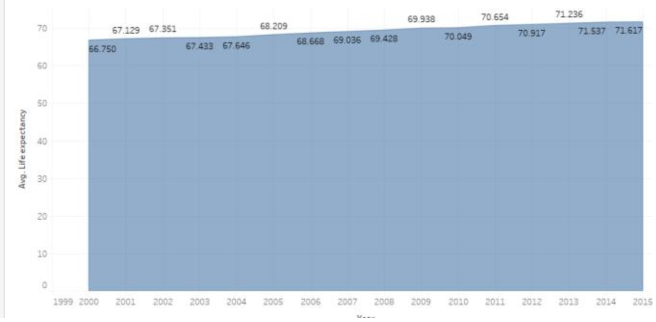
(using tableau and r functions)
Countries like **Japan , Sweden** has highest life expectancy of **82.54** years average life expectancy
Countries like **Sierra and Leone central African republic** with around **46 and 48** years average life expectancy

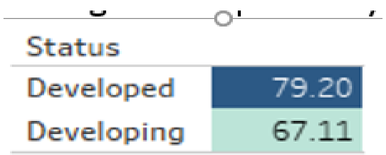
All the countries in dark blue have high life expectancy which are mostly developed countries. Most of the Asian countries have low life expectancy.





Average life expectancy for developing and developed countries:
Average Life expectancy for developed countries is 79.20 and for developing countries 67.11

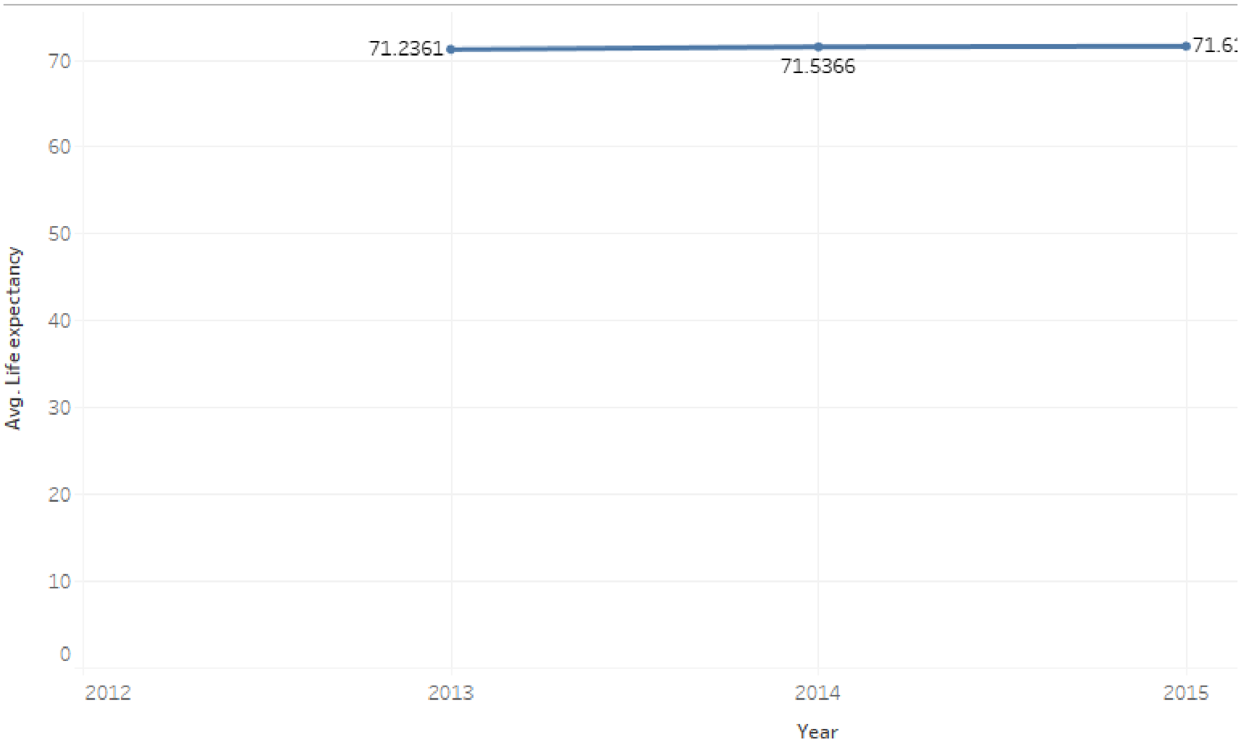




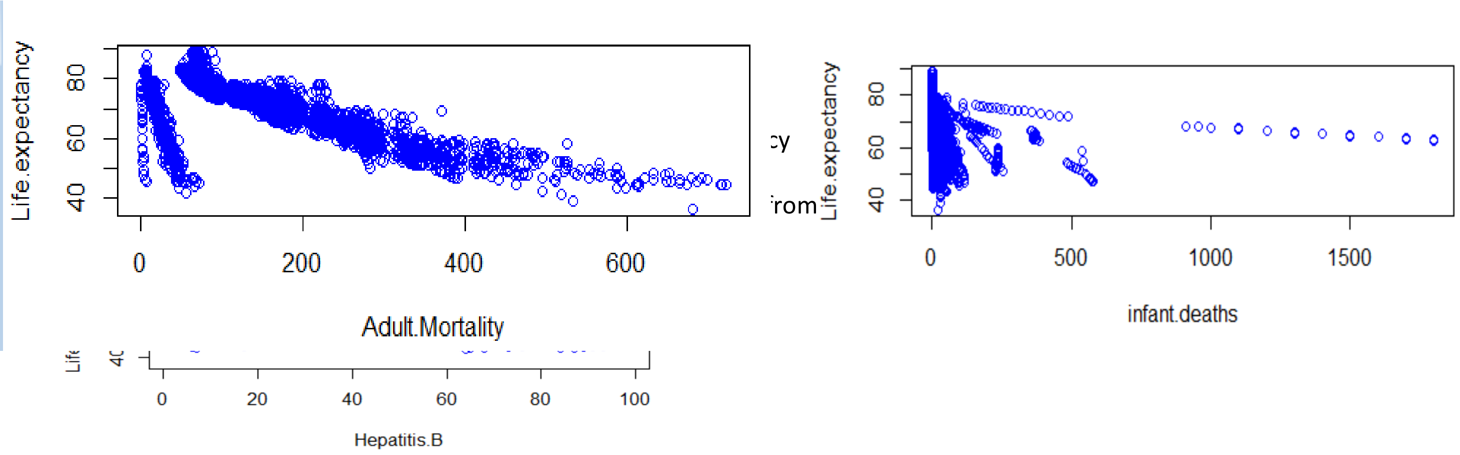
Status	
Developed	79.20
Developing	67.11

The average life expectancy is **71.67**

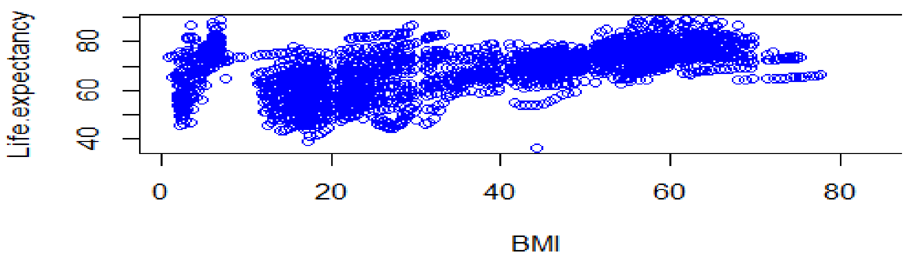
The average life expectancy from year 2013 -2015 :



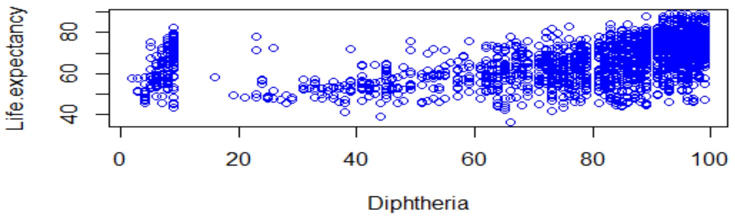
Adult. Mortality, infant death is inversely proportional to Life. Expectancy

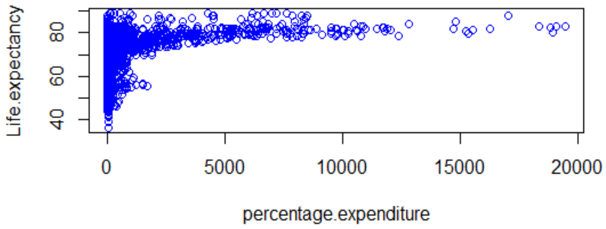


BMI is directly proportional to Life. Expectancy
Variable BMI is behaving strangely it should be decreasing with increase in life expectancy but it is behaving other wise and hence removing this variable from model analysis

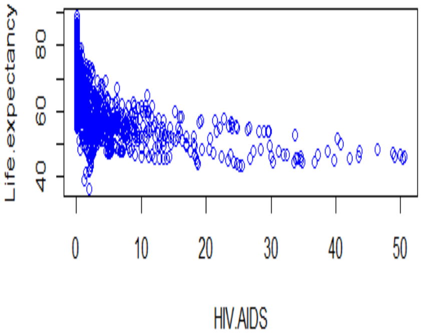


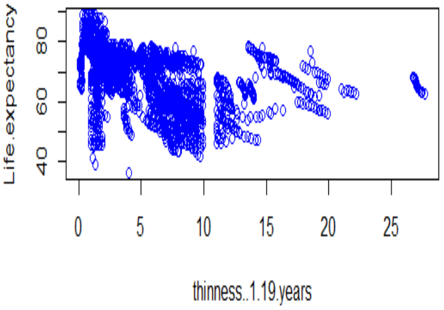
Diphtheria immunization and percentage expenditure is directly proportional to Life. Expectancy

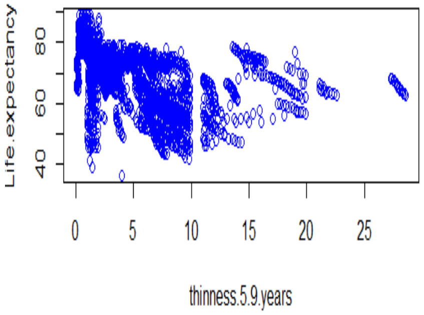




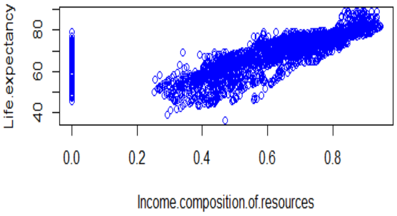
HIV.AIDS, thinness..1.19.years and thinness.5.9.years is inversely proportional to Life. Expectancy

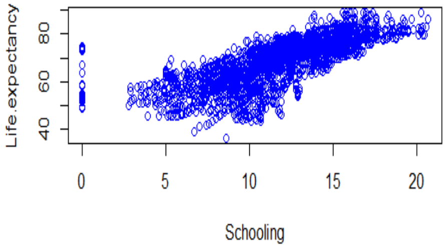


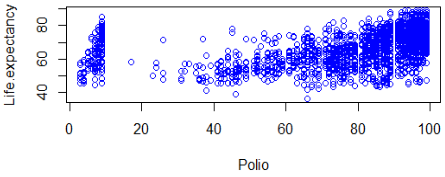




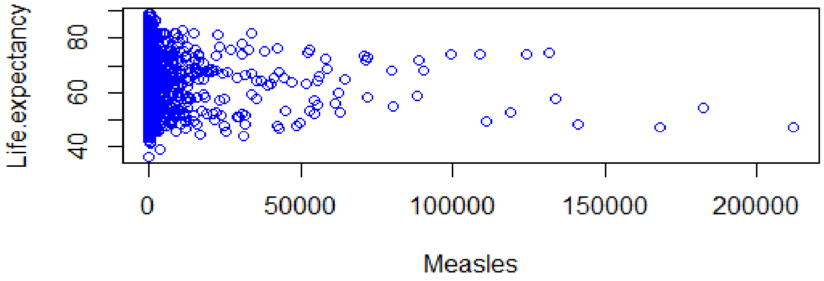
Income.composition.of.resources, Schooling and Polio immunization is directly proportional to Life. Expectancy



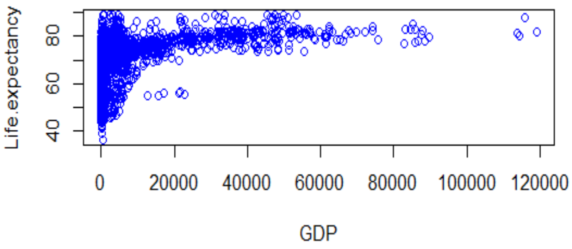


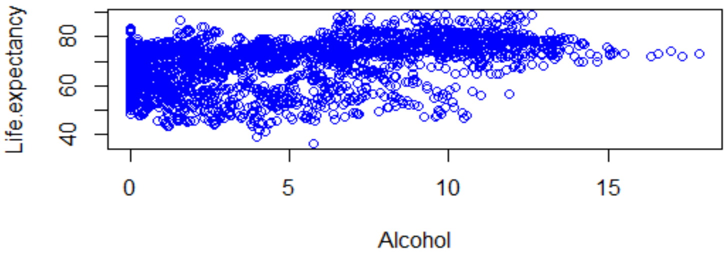


Measles and life expectancy spread is not very clear from graph, it is very scattered

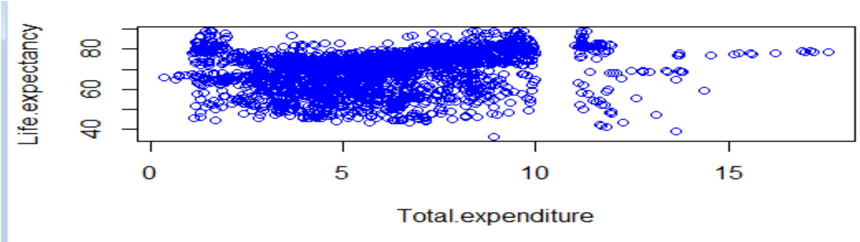


GDP and alcohol is directly proportional to Life. Expectancy

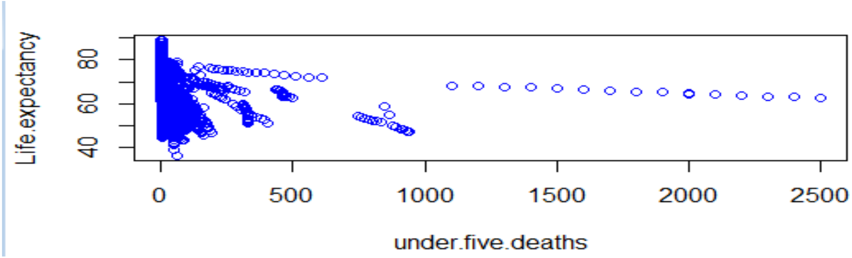




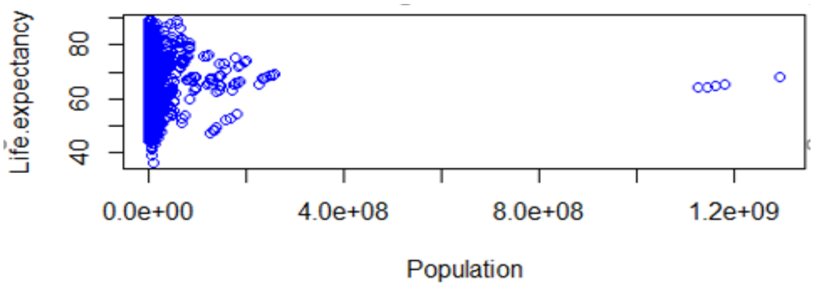
Total.expenditure and life expectancy spread is not very clear from graph, it is very scattered



Under.five.deaths is inversely proportional to life expectancy increases



Population is directly proportional to life expectancy which is strange. This variable spread is not very clear yet and need to be observed further.



2.2.3. Summary of Bivariate Analysis

1. Life expectancy is inversely proportion to below variables: if they will increase life expectancy will decrease.

a) Adult. Mortality
b) infant. Deaths
c) Population
d) HIV.AIDS
e)thinness..1.19.years
f) thinness.5.9.years
g)**under.five.deaths**

2. Life expectancy is Directly proportion to below variables: if they will be high then life expectancy will also increase.

a)Diphtheria immunization
b)Income.composition.of.resources
c)Schooling years
d)Polio immunization
e)GDP
f) percentage.expenditure
g)alcohol

3. Variable **BMI** is behaving strangely it should be decreasing with increase in life expectancy but it is behaving other wise and hence removing this variable from model analysis.
4. **Increase in alcohol consumption increases life expectancy** which is quite strange and is collinear with number of years in school. This variable can be further analyzed.

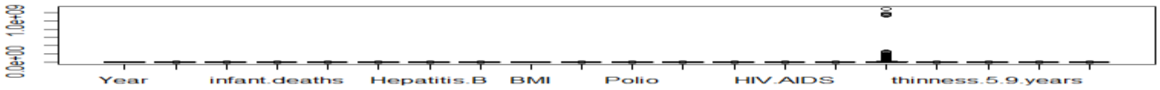
2.3. Data Quality

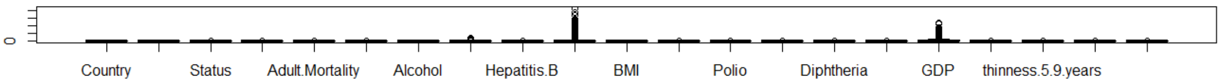
The data has lot of missing values , outlier and inconsistencies and the data need to be treated before applying the data for modelling.

3. Exploratory Data Analysis

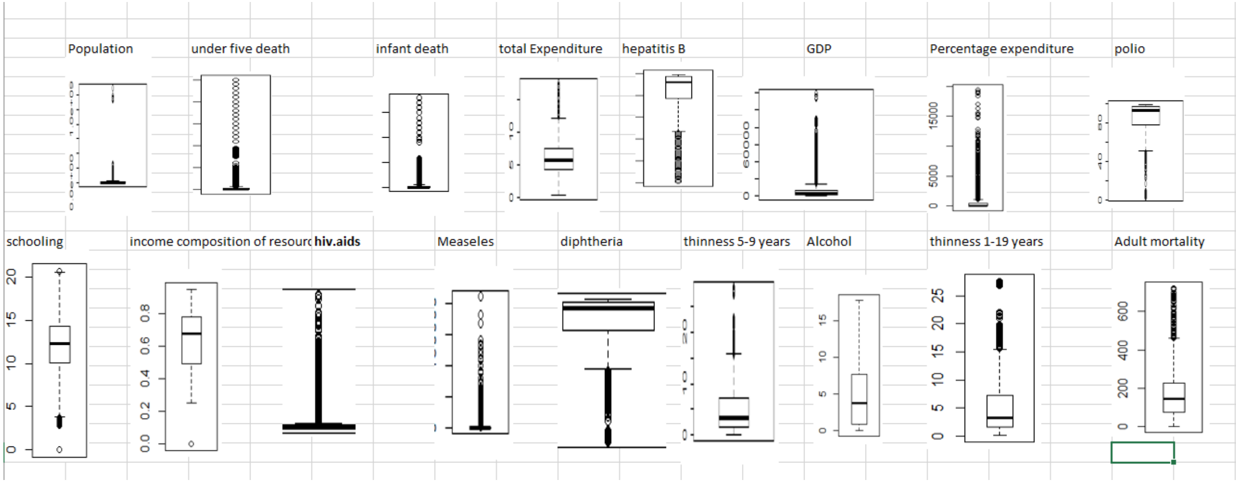
3.1. Box Plot and Outlier Treatment :(removing the column 1 -3 from outlier and missing value treatment as they have factor variable and no missing values and outlier)

Population has outlier and hence removing and checking other variables.

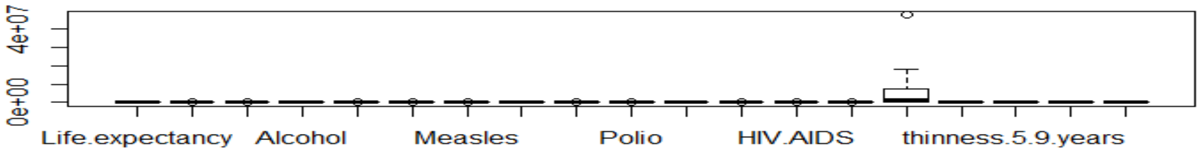




GDP and Measles have outliers but due to large number of variable it is not very clear and hence checking all the box plots individually:
Almost most of the variables have outliers like **infant death, under five death, GDP, Measles, percentage expenditure and HIV AIDs.**



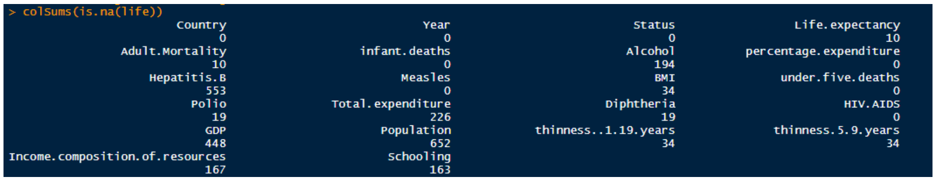
Treating Outliers data with 95 percent capping for higher value and 5 percent capping for lower value:



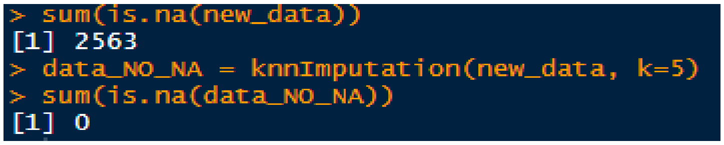
3.2. Missing value treatment

using the data after treating outliers: we will use **KNN imputation method** where it will use mean/mode value to replace missing values as per requirement.
This is based on a kNN algorithm. In this method, k neighbors are chosen based on some distance measure and their average is used as an imputation estimate. The method requires the selection of the number of nearest neighbors, and a distance metric.

Life expectancy, adult mortality, diphtheria, Polio, thinnes..1.19.years, thinnes..5.9.years , BMI, Alcohol, hepatitis b, total expenditure, GDP, population and schooling need to be treated.

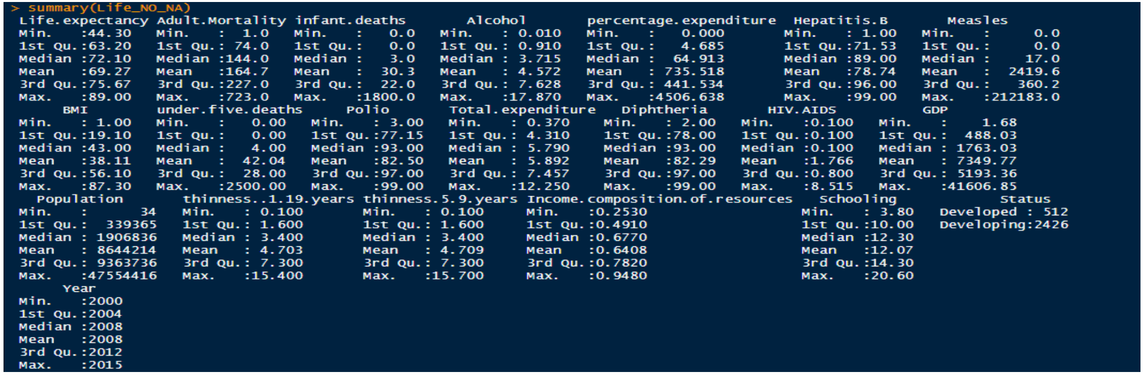


```
> colSums(is.na(life))
                        Country                            Year                          Status                 Life.expectancy
                              0                               0                               0                              10
                Adult.Mortality                   infant.deaths                         Alcohol          percentage.expenditure
                             10                               0                             194                               0
                    Hepatitis.B                         Measles                             BMI               under.five.deaths
                            553                               0                              34                               0
                          Polio               Total.expenditure                      Diphtheria                        HIV.AIDS
                             19                             226                              19                               0
                            GDP                      Population              thinness..1.19.years              thinness.5.9.years
                            448                             652                              34                              34
Income.composition.of.resources                       Schooling
                            167                             163
```



```
> sum(is.na(new_data))
[1] 2563
> data_NO_NA = knnImputation(new_data, k=5)
> sum(is.na(data_NO_NA))
[1] 0
```

Final Data after treating missing values and outliers: **No Missing value and outliers**



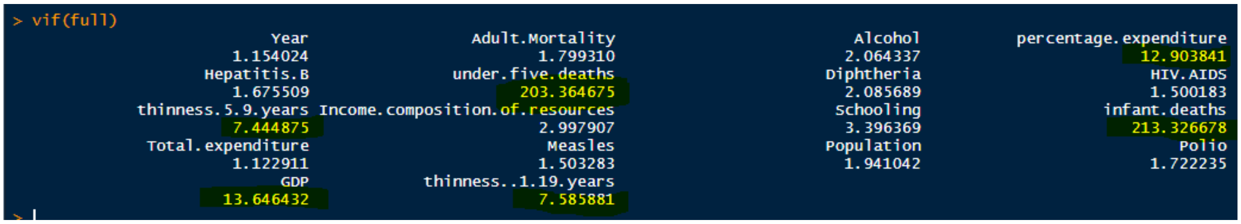
```
> summary(Life_NO_NA)
 Life.expectancy Adult.Mortality infant.deaths        Alcohol       percentage.expenditure  Hepatitis.B       Measles        
 Min.   :44.30   Min.   :  1.0   Min.   :   0.0   Min.   : 0.010   Min.   :   0.000       Min.   : 1.00   Min.   :     0.0  
 1st Qu.:63.20   1st Qu.: 74.0   1st Qu.:   0.0   1st Qu.: 0.910   1st Qu.:   4.685       1st Qu.:71.53   1st Qu.:     0.0  
 Median :72.10   Median :144.0   Median :   3.0   Median : 3.715   Median :  64.913       Median :89.00   Median :    17.0  
 Mean   :69.27   Mean   :164.7   Mean   :  30.3   Mean   : 4.572   Mean   : 735.518       Mean   :78.74   Mean   :  2419.6  
 3rd Qu.:75.67   3rd Qu.:227.0   3rd Qu.:  22.0   3rd Qu.: 7.628   3rd Qu.: 441.534       3rd Qu.:96.00   3rd Qu.:   360.2  
 Max.   :89.00   Max.   :723.0   Max.   :1800.0   Max.   :17.870   Max.   :4506.638       Max.   :99.00   Max.   :212183.0  
      BMI        under.five.deaths      Polio       Total.expenditure   Diphtheria        HIV.AIDS            GDP          
 Min.   : 1.00   Min.   :   0.00   Min.   : 3.00   Min.   : 0.370    Min.   : 2.00   Min.   :0.100   Min.   :    1.68  
 1st Qu.:19.10   1st Qu.:   0.00   1st Qu.:77.15   1st Qu.: 4.310    1st Qu.:78.00   1st Qu.:0.100   1st Qu.:  488.03  
 Median :43.00   Median :   4.00   Median :93.00   Median : 5.790    Median :93.00   Median :0.100   Median : 1763.03  
 Mean   :38.11   Mean   :  42.04   Mean   :82.50   Mean   : 5.892    Mean   :82.29   Mean   :1.766   Mean   : 7349.77  
 3rd Qu.:56.10   3rd Qu.:  28.00   3rd Qu.:97.00   3rd Qu.: 7.457    3rd Qu.:97.00   3rd Qu.:0.800   3rd Qu.: 5193.36  
 Max.   :87.30   Max.   :2500.00   Max.   :99.00   Max.   :12.250    Max.   :99.00   Max.   :8.515   Max.   :41606.85  
   Population        thinness..1.19.years thinness.5.9.years Income.composition.of.resources   Schooling            Status    
 Min.   :      34   Min.   : 0.100        Min.   : 0.100      Min.   :0.2530                   Min.   : 3.80   Developed : 512  
 1st Qu.:  339365   1st Qu.: 1.600        1st Qu.: 1.600      1st Qu.:0.4910                   1st Qu.:10.00   Developing:2426  
 Median : 1906836   Median : 3.400        Median : 3.400      Median :0.6770                   Median :12.30                    
 Mean   : 8644214   Mean   : 4.703        Mean   : 4.709      Mean   :0.6408                   Mean   :12.07                    
 3rd Qu.: 9363736   3rd Qu.: 7.300        3rd Qu.: 7.300      3rd Qu.:0.7820                   3rd Qu.:14.30                    
 Max.   :47554416   Max.   :15.400        Max.   :15.700      Max.   :0.9480                   Max.   :20.60                    
      Year     
 Min.   :2000  
 1st Qu.:2004  
 Median :2008  
 Mean   :2008  
 3rd Qu.:2012  
 Max.   :2015  
```

3.3. Removing multicollinear variables

Removing Factor variable like status, country as VIF is only applicable for numerical variables

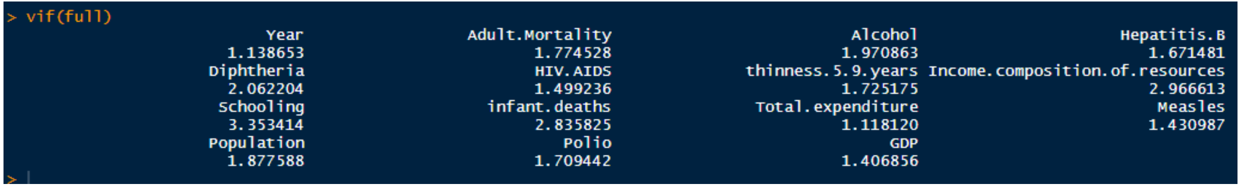
VIF > 5 should be removed but GDP is an important aspect for countries, also infant deaths and under five deaths are similar and thinness variables also look similar hence **removing below variables and checking VIF**:

5. Under.five.death.
6. Percentage Expenditure
7. Thinness.. 1.19 years



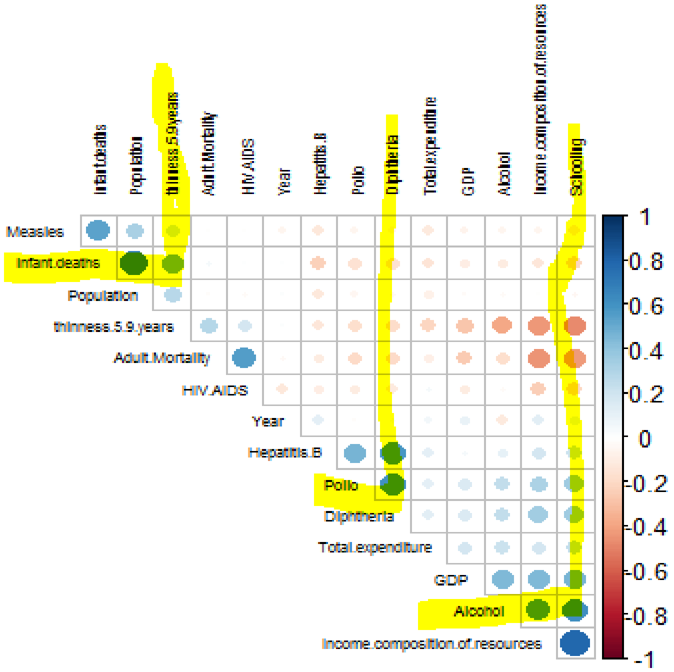
```
> vif(full)
                           Year                 Adult.Mortality                          Alcohol          percentage.expenditure 
                       1.154024                        1.799310                         2.064337                       12.903841 
                    Hepatitis.B               under.five.deaths                       Diphtheria                        HIV.AIDS 
                       1.675509                      203.364675                         2.085689                        1.500183 
             thinness.5.9.years Income.composition.of.resources                        Schooling                   infant.deaths 
                       7.444875                        2.997907                         3.396369                      213.326678 
              Total.expenditure                         Measles                       Population                           Polio 
                       1.122911                        1.503283                         1.941042                        1.722235 
                            GDP            thinness..1.19.years 
                      13.646432                        7.585881 
> |
```

After removing the above 3 variables **there is no multicollinearity**:



```
> vif(full)
                           Year                 Adult.Mortality                          Alcohol                     Hepatitis.B 
                       1.138653                        1.774528                         1.970863                        1.671481 
                     Diphtheria                        HIV.AIDS               thinness.5.9.years Income.composition.of.resources 
                       2.062204                        1.499236                         1.725175                        2.966613 
                      Schooling                   infant.deaths                Total.expenditure                         Measles 
                       3.353414                        2.835825                         1.118120                        1.430987 
                     Population                           Polio                              GDP 
                       1.877588                        1.709442                         1.406856 
> |
```

a. **Correlation Plot**



The correlation plot say the below variables are highly collinear:

1.Polio and Diphtheria (those are either immune to polio or Diphtheria can have high life expectancy)

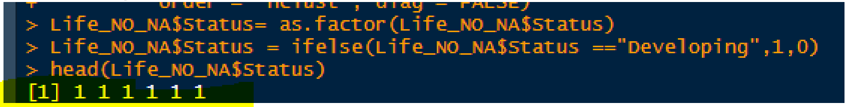
2. infant deaths and thinness 5.9 years.(if either infant deaths or thinness of 5-9 years are more life expectancy is less)

3. alcohol and schooling. (strangely if alcohol consumption is high or the number of schooling years are high then life expectancy is high) which does not make sense hence not considering this one.

3.5. Data Cleaning & Pre-processing

3.5.1 Data Transformation

Converting the categorical variable Status into factor variable and creating dummy variable where 1 is for developing country and 0 for developed country. This will help the status variable to be part of model building and data scaling as well.

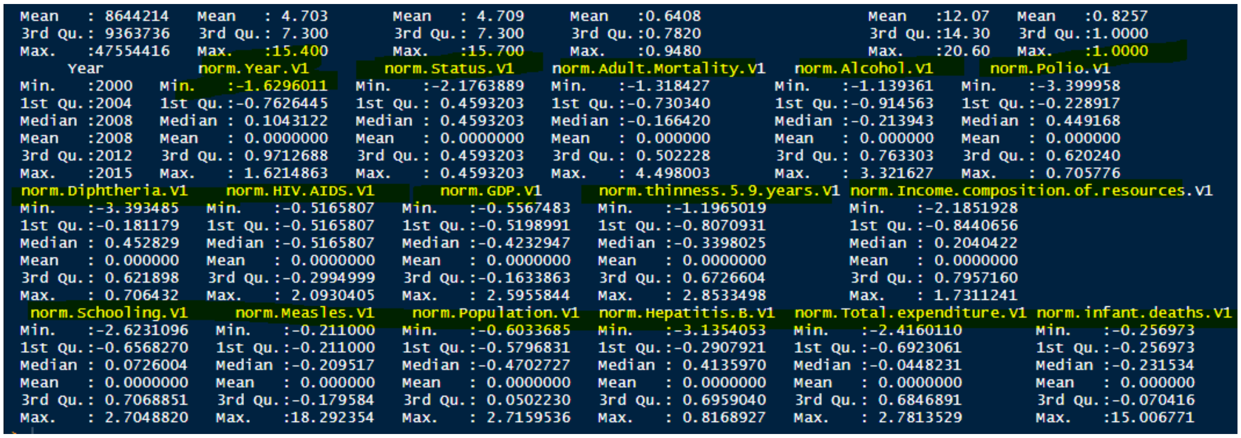


```
+         order = "hclust", diag = FALSE)
> Life_NO_NA$Status= as.factor(Life_NO_NA$Status)
> Life_NO_NA$Status = ifelse(Life_NO_NA$Status =="Developing",1,0)
> head(Life_NO_NA$Status)
[1] 1 1 1 1 1 1
```

3.5.2 Data scaling:

Used the scale method to scale the data such that **each column has mean 0 and variance 1.**

Below is the snap shot of scaled data.



```
 Mean   : 8644214   Mean   : 4.703      Mean   : 4.709    Mean   :0.6408              Mean   :12.07   Mean   :0.8257
 3rd Qu.: 9363736   3rd Qu.: 7.300      3rd Qu.: 7.300    3rd Qu.:0.7820              3rd Qu.:14.30   3rd Qu.:1.0000
 Max.   :47554416   Max.   :15.400      Max.   :15.700    Max.   :0.9480              Max.   :20.60   Max.   :1.0000
      Year            norm.Year.V1         norm.Status.V1    norm.Adult.Mortality.V1    norm.Alcohol.V1      norm.Polio.V1
 Min.   :2000    Min.   :-1.6296011   Min.   :-2.1763889   Min.   :-1.318427        Min.   :-1.139361   Min.   :-3.399958
 1st Qu.:2004    1st Qu.:-0.7626445   1st Qu.: 0.4593203   1st Qu.:-0.730340        1st Qu.:-0.914563   1st Qu.:-0.228917
 Median :2008    Median : 0.1043122   Median : 0.4593203   Median :-0.166420        Median :-0.213943   Median : 0.449168
 Mean   :2008    Mean   : 0.0000000   Mean   : 0.0000000   Mean   : 0.000000        Mean   : 0.000000   Mean   : 0.000000
 3rd Qu.:2012    3rd Qu.: 0.9712688   3rd Qu.: 0.4593203   3rd Qu.: 0.502228        3rd Qu.: 0.763303   3rd Qu.: 0.620240
 Max.   :2015    Max.   : 1.6214863   Max.   : 0.4593203   Max.   : 4.498003        Max.   : 3.321627   Max.   : 0.705776
 norm.Diphtheria.V1    norm.HIV.AIDS.V1       norm.GDP.V1        norm.thinness.5.9.years.V1 norm.Income.composition.of.resources.V1
 Min.   :-3.393485   Min.   :-0.5165807   Min.   :-0.5567483   Min.   :-1.1965019         Min.   :-2.1851928
 1st Qu.:-0.181179   1st Qu.:-0.5165807   1st Qu.:-0.5198991   1st Qu.:-0.8070931         1st Qu.:-0.8440656
 Median : 0.452829   Median :-0.5165807   Median :-0.4232947   Median :-0.3398025         Median : 0.2040422
 Mean   : 0.000000   Mean   : 0.0000000   Mean   : 0.0000000   Mean   : 0.0000000         Mean   : 0.0000000
 3rd Qu.: 0.621898   3rd Qu.:-0.2994999   3rd Qu.:-0.1633863   3rd Qu.: 0.6726604         3rd Qu.: 0.7957160
 Max.   : 0.706432   Max.   : 2.0930405   Max.   : 2.5955844   Max.   : 2.8533498         Max.   : 1.7311241
  norm.Schooling.V1     norm.Measles.V1     norm.Population.V1  norm.Hepatitis.B.V1  norm.Total.expenditure.V1 norm.infant.deaths.V1
 Min.   :-2.6231096   Min.   :-0.211000   Min.   :-0.6033685   Min.   :-3.1354053   Min.   :-2.4160110        Min.   :-0.256973
 1st Qu.:-0.6568270   1st Qu.:-0.211000   1st Qu.:-0.5796831   1st Qu.:-0.2907921   1st Qu.:-0.6923061        1st Qu.:-0.256973
 Median : 0.0726004   Median :-0.209517   Median :-0.4702727   Median : 0.4135970   Median :-0.0448231        Median :-0.231534
 Mean   : 0.0000000   Mean   : 0.000000   Mean   : 0.0000000   Mean   : 0.0000000   Mean   : 0.0000000        Mean   : 0.000000
 3rd Qu.: 0.7068851   3rd Qu.:-0.179584   3rd Qu.: 0.0502230   3rd Qu.: 0.6959040   3rd Qu.: 0.6846891        3rd Qu.:-0.070416
 Max.   : 2.7048820   Max.   :18.292354   Max.   : 2.7159536   Max.   : 0.8168927   Max.   : 2.7813529        Max.   :15.006771
```

3.5.3. Summary of number of variables removed

We have removed below variables from our data for analysis which is either non-significant, have no statistical information, highly collinear variables :

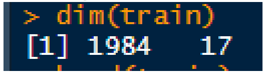
1. **Country :** there are 193 countries and hence it has no statistical importance
2. **BMI :** Variable BMI not showing correct data. It is increasing with life expectancy but it should decrease.
3. **Under.five.death. :** multicollinear as shown in VIF result
4. **Percentage Expenditure:** multicollinear as shown in VIF result
5. **Thinness.. 1.19 years:** multicollinear as shown in VIF result
6. **Polio:** based on correlation plot
7. **Thinness 5.9 years:** based on correlation plot.

4. Model building and tuning

4.1. Data Slicing : after treating data with missing values, outliers and removing unwanted variables and scaling data we will divide the data in to training and test data set:

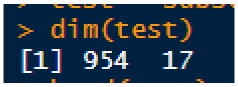
Training Data is 70 percent and test data is 30 percent:

Training data has **1987 rows and 17 columns**



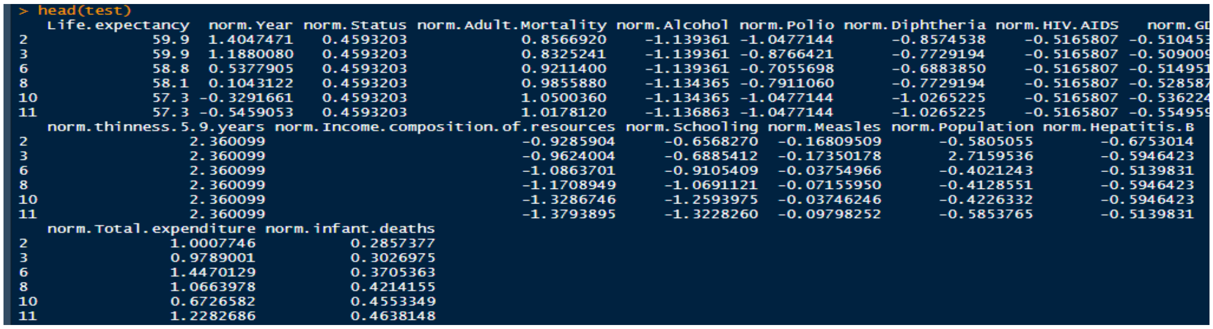
```
> dim(train)
[1] 1984   17
```

Test Data: has **954 rows and 17 columns**



```
> dim(test)
[1] 954  17
```

All the variables for test data:



```
> head(test)
   Life.expectancy  norm.Year norm.Status norm.Adult.Mortality norm.Alcohol norm.Polio norm.Diphtheria norm.HIV.AIDS    norm.GD
2             59.9  1.4047471   0.4593203            0.8566920    -1.139361 -1.0477144      -0.8574538    -0.5165807 -0.51045
3             59.9  1.1880080   0.4593203            0.8325241    -1.139361 -0.8766421      -0.7729194    -0.5165807 -0.50900
6             58.8  0.5377905   0.4593203            0.9211400    -1.139361 -0.7055698      -0.6883850    -0.5165807 -0.51495
8             58.1  0.1043122   0.4593203            0.9855880    -1.134365 -0.7911060      -0.7729194    -0.5165807 -0.52858
10            57.3 -0.3291661   0.4593203            1.0500360    -1.134365 -1.0477144      -1.0265225    -0.5165807 -0.53622
11            57.3 -0.5459053   0.4593203            1.0178120    -1.136863 -1.0477144      -1.0265225    -0.5165807 -0.55495
   norm.thinness.5.9.years norm.Income.composition.of.resources norm.Schooling norm.Measles norm.Population norm.Hepatitis.B
2                 2.360099                           -0.9285904     -0.6568270  -0.16809509      -0.5805055       -0.6753014
3                 2.360099                           -0.9624004     -0.6885412  -0.17350178       2.7159536       -0.5946423
6                 2.360099                           -1.0863701     -0.9105409  -0.03754966      -0.4021243       -0.5139831
8                 2.360099                           -1.1708949     -1.0691121  -0.07155950      -0.4128551       -0.5946423
10                2.360099                           -1.3286746     -1.2593975  -0.03746246      -0.4226332       -0.5946423
11                2.360099                           -1.3793895     -1.3228260  -0.09798252      -0.5853765       -0.5139831
   norm.Total.expenditure norm.infant.deaths
2               1.0007746          0.2857377
3               0.9789001          0.3026975
6               1.4470129          0.3705363
8               1.0663978          0.4214155
10              0.6726582          0.4553349
11              1.2282686          0.4638148
```

4.1.2 Reasons for choosing the Models for the Analysis

1. **Linear Model:**

   a)since this is **regression data** and hence we can use this model

b) It will be easier to **interpret** or modify this model.
c)It will help identify the **significant variables** for prediction
d) most of the independent variables are **numerical variables**

2. **Decision tree**:
   a) This model can be **tuned** for performance.
   b) Trees are **easily explainable**.
   c) This model is used for **regression data**.
   d) It can also help identify **important variables.**
3. **Random Forest**
   a)Random forest can be used for **regression data.**
   b) this model can be **tuned.**
   c) It uses **ensemble technique like bagging.**
   d) it helps identify the most **important variables with numerical weightage.**

4.2. Linear Regression

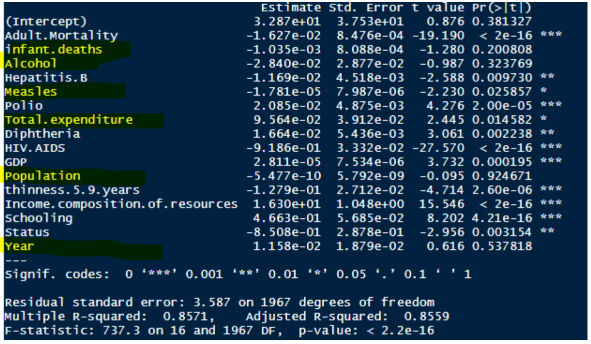
Multiple Linear Regression: this is an extension of the simple linear regression model in which the number of independent variables will be more than one.

$$Y= \beta 0+\beta 1X1+\beta 2X2$$

- **Multiple R Squared**: It tells you how strong the linear relationship **is**. For example, a value of 1 means a perfect positive relationship and a value of zero means no relationship at all.
- **Adjusted R Squared:** it compensates the increase in R square with more number of variables added and The **adjusted R-squared** increases only if the new variable improves the model.
- **Beta Coefficients :** β0 is the intercept ,β1 and β1 are slopes w.r.t variables x1 and x2.
- **P values:** how statistically significant each of our estimates for the variables

4.2.1. Model Summary

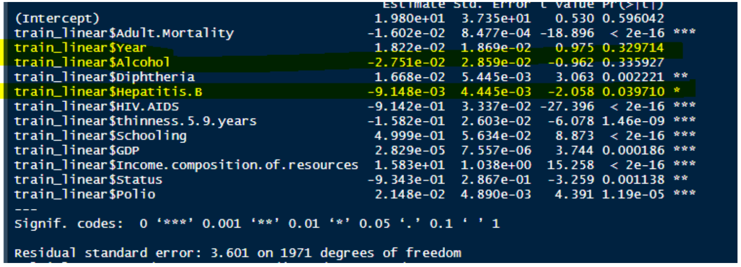
Removing Insignificant Variables:



```
                              Estimate Std. Error t value Pr(>|t|)
(Intercept)                  3.287e+01  3.753e+01   0.876 0.381327
Adult.Mortality             -1.627e-02  8.476e-04 -19.190  < 2e-16 ***
infant.deaths               -1.035e-03  8.088e-04  -1.280 0.200808
Alcohol                     -2.840e-02  2.877e-02  -0.987 0.323769
Hepatitis.B                 -1.169e-02  4.518e-03  -2.588 0.009730 **
Measles                     -1.781e-05  7.987e-06  -2.230 0.025857 *
Polio                        2.085e-02  4.875e-03   4.276 2.00e-05 ***
Total.expenditure            9.564e-02  3.912e-02   2.445 0.014582 *
Diphtheria                   1.664e-02  5.436e-03   3.061 0.002238 **
HIV.AIDS                    -9.186e-01  3.332e-02 -27.570  < 2e-16 ***
GDP                          2.811e-05  7.534e-06   3.732 0.000195 ***
Population                  -5.477e-10  5.792e-09  -0.095 0.924671
thinness.5.9.years          -1.279e-01  2.712e-02  -4.714 2.60e-06 ***
Income.composition.of.resources  1.630e+01  1.048e+00  15.546  < 2e-16 ***
Schooling                    4.663e-01  5.685e-02   8.202 4.21e-16 ***
Status                      -8.508e-01  2.878e-01  -2.956 0.003154 **
Year                         1.158e-02  1.879e-02   0.616 0.537818
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.587 on 1967 degrees of freedom
Multiple R-squared:  0.8571,	Adjusted R-squared:  0.8559 
F-statistic: 737.3 on 16 and 1967 DF,  p-value: < 2.2e-16
```

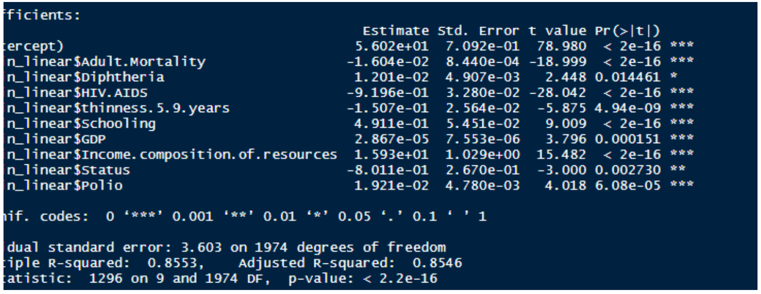
The variables highlighted above shows insignificant or less significant**(with no stars signs).** Creating model removing measles, population, infant deaths, total expenditure and checked but then also the next model has insignificant variables like alcohol, year and hepatitis B:



```
                                           Estimate Std. Error t value Pr(>|t|)
(Intercept)                               1.980e+01  3.735e+01   0.530 0.596042
train_linear$Adult.Mortality             -1.602e-02  8.477e-04 -18.896  < 2e-16 ***
train_linear$Year                         1.822e-02  1.869e-02   0.975 0.329714
train_linear$Alcohol                     -2.751e-02  2.859e-02  -0.962 0.335927
train_linear$Diphtheria                   1.668e-02  5.445e-03   3.063 0.002221 **
train_linear$Hepatitis.B                 -9.148e-03  4.445e-03  -2.058 0.039710 *
train_linear$HIV.AIDS                    -9.142e-01  3.337e-02 -27.396  < 2e-16 ***
train_linear$thinness.5.9.years          -1.582e-01  2.603e-02  -6.078 1.46e-09 ***
train_linear$Schooling                    4.999e-01  5.634e-02   8.873  < 2e-16 ***
train_linear$GDP                          2.829e-05  7.557e-06   3.744 0.000186 ***
train_linear$Income.composition.of.resources  1.583e+01  1.038e+00  15.258  < 2e-16 ***
train_linear$Status                      -9.343e-01  2.867e-01  -3.259 0.001138 **
train_linear$Polio                        2.148e-02  4.890e-03   4.391 1.19e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.601 on 1971 degrees of freedom
```

final model with all significant variables
After removing the above variable the final model with all significant variable is below:



```
fficients:
                                          Estimate Std. Error t value Pr(>|t|)    
ercept)                                  5.602e+01  7.092e-01  78.980  < 2e-16 ***
n_linear$Adult.Mortality                -1.604e-02  8.440e-04 -18.999  < 2e-16 ***
n_linear$Diphtheria                      1.201e-02  4.907e-03   2.448 0.014461 *  
n_linear$HIV.AIDS                       -9.196e-01  3.280e-02 -28.042  < 2e-16 ***
n_linear$thinness.5.9.years             -1.507e-01  2.564e-02  -5.875 4.94e-09 ***
n_linear$Schooling                       4.911e-01  5.451e-02   9.009  < 2e-16 ***
n_linear$GDP                             2.867e-05  7.553e-06   3.796 0.000151 ***
n_linear$Income.composition.of.resources 1.593e+01  1.029e+00  15.482  < 2e-16 ***
n_linear$Status                         -8.011e-01  2.670e-01  -3.000 0.002730 ** 
n_linear$Polio                           1.921e-02  4.780e-03   4.018 6.08e-05 ***

if. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

dual standard error: 3.603 on 1974 degrees of freedom
iple R-squared:  0.8553,	Adjusted R-squared:  0.8546 
atistic:  1296 on 9 and 1974 DF,  p-value: < 2.2e-16
```

4.2.2. Interpretation :

This Models has a Rsquare value of 85.5 which is good which means the target variable has a strong linear relation with the variables shown, also adjusted R squared is equivalent to R square which means no variable is insignificant here.

Beta Coefficients:
The estimate value tells the magnitude at which the target variable will be impacted with change in x.
For Eg. : with 1 unit increase in Adult Mortality, life expectancy will increase by 5.602e+01 unit.

P value: lesser the p value, more significant the variable for the linear relationship.
4.2.3. Evaluating model performance
To improve the performance of the model we need to remove the insignificant variables.

1. Residuals: residuals have a mean of zero therefore the median should be very close to zero.

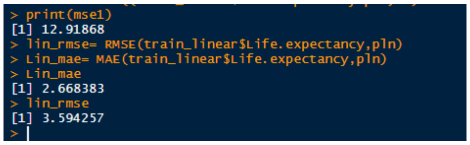
2. Coefficients: are the beta coefficients and their significance: a. Standard Error: defines the accuracy of beta coefficients. We use this value to predict the lower and upper limits at 95% confidence intervals.

b**. T-test value** (H0 = all coefficients equal to zero): if the value is greater than 1.963 (at

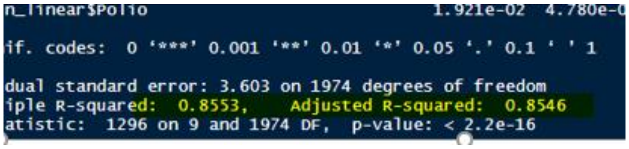
c. P-value: lower that 0.05 means the null hypothesis (all coefficients equal to zero) is rejected and the relationship is significant.

3. **Residual Standard Error:** is the average variation of points around the fitted regression line. This matrix gives the overall quality of the model, lower the RSE better the quality. We can calculate the percentage error for a given model to assess if the value is acceptable or not.
4. **F-Static:** Gives the overall significance of the model, but in simple linear model this is redundant.
5. **R-squared & Adj. R-squared**: The value ranges from 0 to 1 and it represents the proportion of variation in the data that is explained by the model. The adjusted R-squared is adjusted for degrees of freedom. Higher the value is better.

The key performance measures like MSE, MAE and RMSE are shown below:



```
> print(mse1)
[1] 12.91868
> lin_rmse= RMSE(train_linear$Life.expectancy,pln)
> Lin_mae= MAE(train_linear$Life.expectancy,pln)
> Lin_mae
[1] 2.668383
> lin_rmse
[1] 3.594257
> |
```



```
n_linear$Polio                           1.921e-02  4.780e-0
if. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

dual standard error: 3.603 on 1974 degrees of freedom
iple R-squared:  0.8553,	Adjusted R-squared:  0.8546 
atistic:  1296 on 9 and 1974 DF,  p-value: < 2.2e-16
```

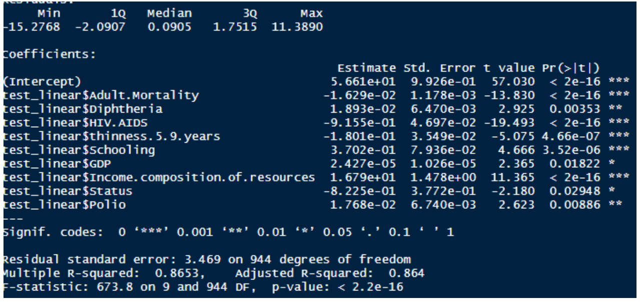
4.2.4 Linear Regression – significant variables

The most significant variable as per linear model for determining life expectancy are as below:
1. Adult Mortality
2. Income composition of resources

3. Schooling
4. GDP
5. Polio
6. Diphtheria
7. thinness 5-9 years

4.2.5 Predict model using test data

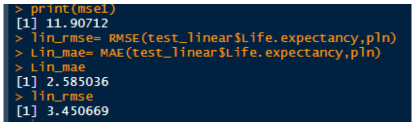
Strangely test data performing little better as compared to training data:



```
    Min       1Q   Median       3Q      Max
-15.2768  -2.0907   0.0905   1.7515  11.3890

Coefficients:
                                          Estimate Std. Error t value Pr(>|t|)    
(Intercept)                              5.661e+01  9.926e-01  57.030  < 2e-16 ***
test_linear$Adult.Mortality             -1.629e-02  1.178e-03 -13.830  < 2e-16 ***
test_linear$Diphtheria                   1.893e-02  6.470e-03   2.925  0.00353 ** 
test_linear$HIV.AIDS                    -9.155e-01  4.697e-02 -19.493  < 2e-16 ***
test_linear$thinness.5.9.years          -1.801e-01  3.549e-02  -5.075 4.66e-07 ***
test_linear$Schooling                    3.702e-01  7.936e-02   4.666 3.52e-06 ***
test_linear$GDP                          2.427e-05  1.026e-05   2.365  0.01822 *  
test_linear$Income.composition.of.resources  1.679e+01  1.478e+00  11.365  < 2e-16 ***
test_linear$Status                      -8.225e-01  3.772e-01  -2.180  0.02948 *  
test_linear$Polio                        1.768e-02  6.740e-03   2.623  0.00886 ** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.469 on 944 degrees of freedom
Multiple R-squared:  0.8653,	Adjusted R-squared:  0.864 
F-statistic: 673.8 on 9 and 944 DF,  p-value: < 2.2e-16
```



```
> print(mse1)
[1] 11.90712
> lin_rmse= RMSE(test_linear$Life.expectancy,pln)
> Lin_mae= MAE(test_linear$Life.expectancy,pln)
> Lin_mae
[1] 2.585036
> lin_rmse
[1] 3.450669
```

4.3. Decision Tree /CART

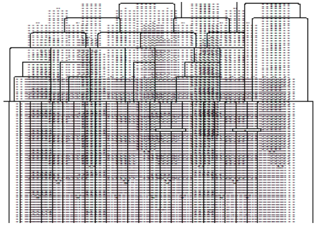
Decision Trees are a type of Supervised Machine Learning where the data is continuously split according to a certain parameter.

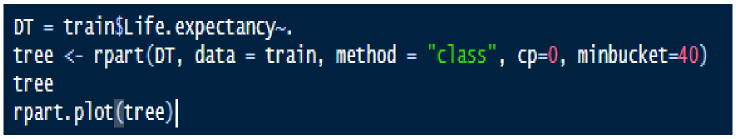
Decision tree is a graph to represent choices and their results in form of a **tree**

Parameter	Value	Significance
minsplit	1000	If the node will have at least 1000 observations then only it will split.
minbucket	100	The terminal nodes should have at least 100 observations.
cp (Complexity Parameter)	0	Allowing the full tree to be grown.
xval(Cross Validation)	10	It will cross validate 10 times.

4.3.1. Model Summary

Due to many variables tree is very complex and difficult to understand, complexity parameter is set to 0 means very complex tree.

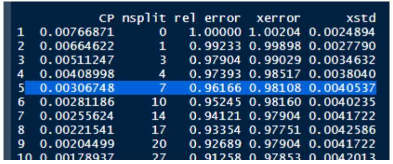




```
DT = train$Life.expectancy~.
tree <- rpart(DT, data = train, method = "class", cp=0, minbucket=40)
tree
rpart.plot(tree)
```

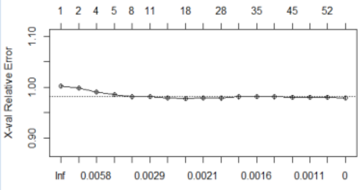
4.3.2. Interpretation:

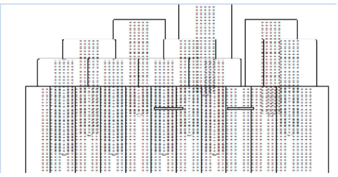
The error will keep decreasing but if we check xerror which is cross validation sample error, it will not decrease after a point and we need to prune tree with that complexity parameter.



```
           CP nsplit rel error  xerror      xstd
1  0.00766871      0   1.00000 1.00204 0.0024894
2  0.00664622      1   0.99233 0.99898 0.0027790
3  0.00511247      3   0.97904 0.99029 0.0034632
4  0.00408998      4   0.97393 0.98517 0.0038040
5  0.00306748      7   0.96166 0.98108 0.0040537
6  0.00281186     10   0.95245 0.98160 0.0040235
7  0.00255624     14   0.94121 0.97904 0.0041722
8  0.00221541     17   0.93354 0.97751 0.0042586
9  0.00204499     20   0.92689 0.97904 0.0041722
10 0.00178937     27   0.91258 0.97853 0.0042013
```

After 0.0029 as per graph the tree has stopped decreasing error : Even after that tree is very complex.



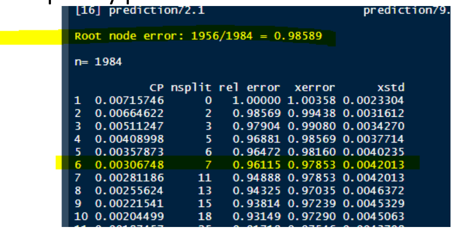


4.3.3. Effort to improve model performance by Pruning

Prunning:

- You can start with a large tree and can come down to small tree. Pruning is done to avoid over fitting of data. Here we are using complexity parameter(alpha) to prune the tree.
- Every time you add a branch make sure the error you decrease is more than alpha.
  More complex a tree is more better it is.

The model tuning is important and that can be done with choosing the xerror value and then using that in complexity parameter selection:



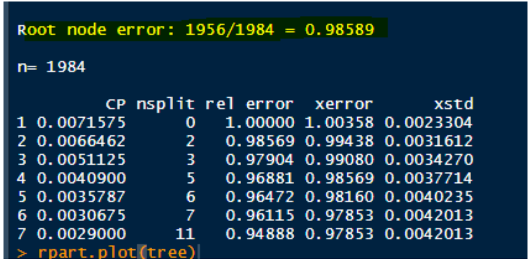
```
[16] prediction72.1                              prediction79.

Root node error: 1956/1984 = 0.98589

n= 1984

           CP nsplit rel error  xerror      xstd
1  0.00715746      0   1.00000 1.00358 0.0023304
2  0.00664622      2   0.98569 0.99438 0.0031612
3  0.00511247      3   0.97904 0.99080 0.0034270
4  0.00408998      5   0.96881 0.98569 0.0037714
5  0.00357873      6   0.96472 0.98160 0.0040235
6  0.00306748      7   0.96115 0.97853 0.0042013
7  0.00281186     11   0.94888 0.97853 0.0042013
8  0.00255624     13   0.94325 0.97035 0.0046372
9  0.00221541     15   0.93814 0.97239 0.0045329
10 0.00204499     18   0.93149 0.97290 0.0045063
```

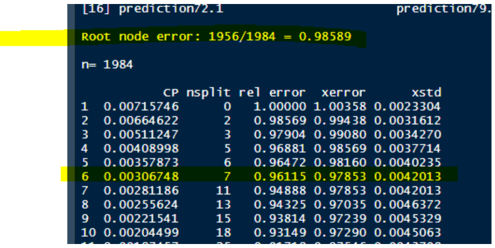
After pruning the model: after pruning the tree size also the error has not been reduced. Its is 98.5 percent



```
Root node error: 1956/1984 = 0.98589

n= 1984

         CP nsplit rel error  xerror      xstd
1 0.0071575      0   1.00000 1.00358 0.0023304
2 0.0066462      2   0.98569 0.99438 0.0031612
3 0.0051125      3   0.97904 0.99080 0.0034270
4 0.0040900      5   0.96881 0.98569 0.0037714
5 0.0035787      6   0.96472 0.98160 0.0040235
6 0.0030675      7   0.96115 0.97853 0.0042013
7 0.0029000     11   0.94888 0.97853 0.0042013
> rpart.plot(tree)
```



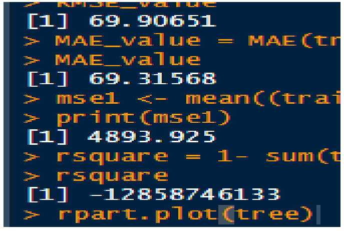
```
[16] prediction72.1                              prediction79.

Root node error: 1956/1984 = 0.98589

n= 1984

           CP nsplit rel error  xerror      xstd
1  0.00715746      0   1.00000 1.00358 0.0023304
2  0.00664622      2   0.98569 0.99438 0.0031612
3  0.00511247      3   0.97904 0.99080 0.0034270
4  0.00408998      5   0.96881 0.98569 0.0037714
5  0.00357873      6   0.96472 0.98160 0.0040235
6  0.00306748      7   0.96115 0.97853 0.0042013
7  0.00281186     11   0.94888 0.97853 0.0042013
8  0.00255624     13   0.94325 0.97035 0.0046372
9  0.00221541     15   0.93814 0.97239 0.0045329
10 0.00204499     18   0.93149 0.97290 0.0045063
```

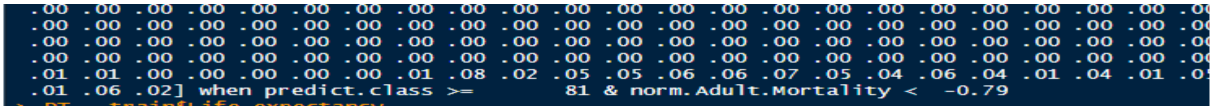
Key performance measures like MAE, MSE and RMSE showing very strange value



```
> RMSE_value
[1] 69.90651
> MAE_value = MAE(tr
> MAE_value
[1] 69.31568
> mse1 <- mean((trai
> print(mse1)
[1] 4893.925
> rsquare = 1- sum(t
> rsquare
[1] -12858746133
> rpart.plot(tree)
```

4.3.4 Decision Tree– significant variables

Using rpart.rules : the most important variable is **adult Mortality** for determining life expectancy

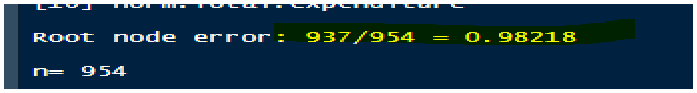


```
 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .0
 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .0
 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .0
 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .00 .0
 .01 .01 .00 .00 .00 .00 .00 .01 .08 .02 .05 .05 .06 .06 .07 .05 .04 .06 .04 .01 .04 .01 .0
 .01 .06 .02] when predict.class >=       81 & norm.Adult.Mortality <  -0.79
```

4.3.5 Predict model using test data

Error and other key measures are similar in test data: and even bad in test data

Error :

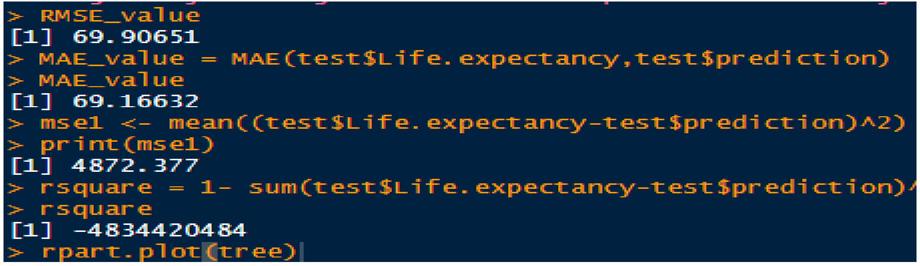


```
[10] norm.Total.expenditure

Root node error: 937/954 = 0.98218

n= 954
```

Other Key Performance measures for regressions



```
> RMSE_value
[1] 69.90651
> MAE_value = MAE(test$Life.expectancy,test$prediction)
> MAE_value
[1] 69.16632
> mse1 <- mean((test$Life.expectancy-test$prediction)^2)
> print(mse1)
[1] 4872.377
> rsquare = 1- sum(test$Life.expectancy-test$prediction)^
> rsquare
[1] -4834420484
> rpart.plot(tree)
```

4.4. Random Forest

A **decision tree** is built on an entire dataset, using all the features/variables of interest, whereas a **random forest** randomly selects observations/rows and specific features/variables to build multiple **decision trees** from and then averages the results

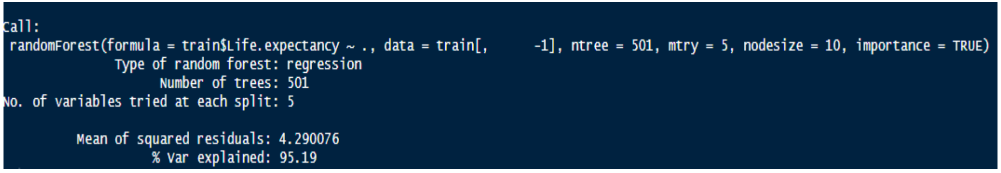
for prediction we could use the **mean for regression trees** and **mode for classification trees**.

Pros Of Random Forest:

1. Decision trees are very sensitive to even small changes in the data - usually called unstable. Random Forest is more **robust** .
2. While individual trees are tend to over-fit training data, averaging corrects this.
3. The general procedure of using multiple models to obtain better predictive performance is called **ensemble learning**(Bootstrap aggregating) also called bagging: Generate new training subsets of the original, each of the same size (usually the size of the data) by sampling with replacement.

4.4.1. Model Summary

- **mtry:** Number of variables available for splitting at each tree node
- **ntree:** Number of trees to grow
- **nodesize** = minimum number of node size (small number will make larger tree and large number will make small tree)
- **importance Variable** : give importance of each variable with weights



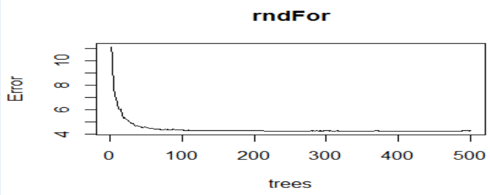
```
Call:
 randomForest(formula = train$Life.expectancy ~ ., data = train[,      -1], ntree = 501, mtry = 5, nodesize = 10, importance = TRUE)
               Type of random forest: regression
                     Number of trees: 501
No. of variables tried at each split: 5

          Mean of squared residuals: 4.290076
                    % Var explained: 95.19
```

4.4.2. Interpretation and tuning

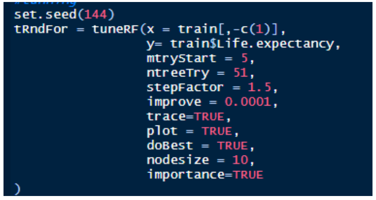
Model tuning is very important for the better performance.

Plot the tree : this tree shows the error rate is constant when number of tree is around 51 (between 0-100)



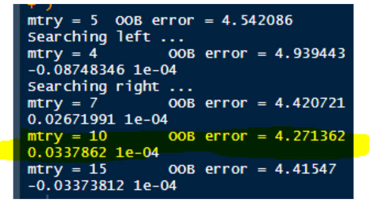
4.4.3. Effort to improve model performance by tuning

Tunning the model with Mtree =51(as discussed), mtree = 5 as start which will increase 1.5 time everytime.

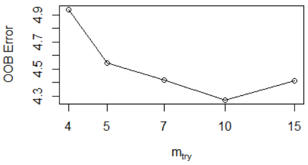


```
set.seed(144)
tRndFor = tuneRF(x = train[,-c(1)],
                 y= train$Life.expectancy,
                 mtryStart = 5,
                 ntreeTry = 51,
                 stepFactor = 1.5,
                 improve = 0.0001,
                 trace=TRUE,
                 plot = TRUE,
                 doBest = TRUE,
                 nodesize = 10,
                 importance=TRUE
)
```

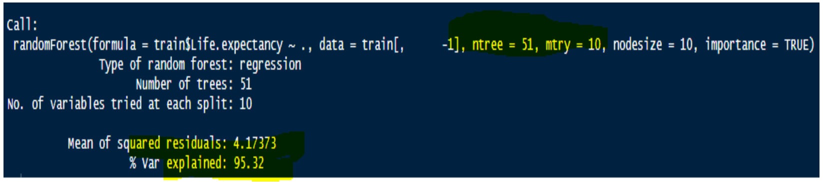
mtree = 10 as OOB is least for that



```
mtry = 5  OOB error = 4.542086
Searching left ...
mtry = 4        OOB error = 4.939443
-0.08748346 1e-04
Searching right ...
mtry = 7        OOB error = 4.420721
0.02671991 1e-04
mtry = 10       OOB error = 4.271362
0.0337862 1e-04
mtry = 15       OOB error = 4.41547
-0.03373812 1e-04
```



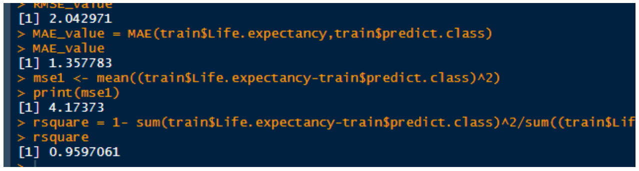
Tuned random forest with Mtree =51(as discussed), mtree = 10



```
Call:
 randomForest(formula = train$Life.expectancy ~ ., data = train[,      -1], ntree = 51, mtry = 10, nodesize = 10, importance = TRUE)
               Type of random forest: regression
                     Number of trees: 51
No. of variables tried at each split: 10

          Mean of squared residuals: 4.17373
                    % Var explained: 95.32
```

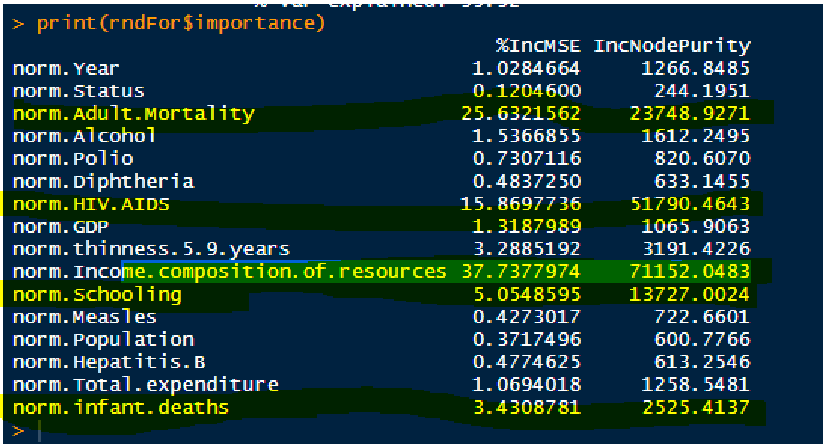
Key performance measures like MAE, MSE and RMSE



```
> RMSE_value
[1] 2.042971
> MAE_value = MAE(train$Life.expectancy,train$predict.class)
> MAE_value
[1] 1.357783
> mse1 <- mean((train$Life.expectancy-train$predict.class)^2)
> print(mse1)
[1] 4.17373
> rsquare = 1- sum(train$Life.expectancy-train$predict.class)^2/sum((train$Lif
> rsquare
[1] 0.9597061
```

4.4.4 Random Forest – significant variables

Important variables using importance function: greater the number on IncNodePurity, more significant the dependent variables are in predicting life expectancy.



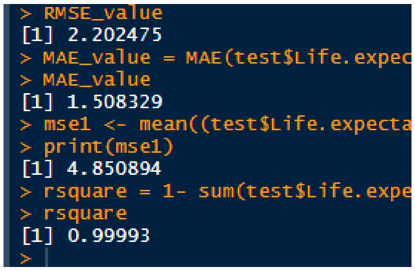
```
> print(rndFor$importance)
                                    %IncMSE IncNodePurity
norm.Year                         1.0284664     1266.8485
norm.Status                       0.1204600      244.1951
norm.Adult.Mortality             25.6321562    23748.9271
norm.Alcohol                      1.5366855     1612.2495
norm.Polio                        0.7307116      820.6070
norm.Diphtheria                   0.4837250      633.1455
norm.HIV.AIDS                    15.8697736    51790.4643
norm.GDP                          1.3187989     1065.9063
norm.thinness.5.9.years           3.2885192     3191.4226
norm.Income.composition.of.resources 37.7377974    71152.0483
norm.Schooling                    5.0548595    13727.0024
norm.Measles                      0.4273017      722.6601
norm.Population                   0.3717496      600.7766
norm.Hepatitis.B                  0.4774625      613.2546
norm.Total.expenditure            1.0694018     1258.5481
norm.infant.deaths                3.4308781     2525.4137
>
```

1.Income composition of resources

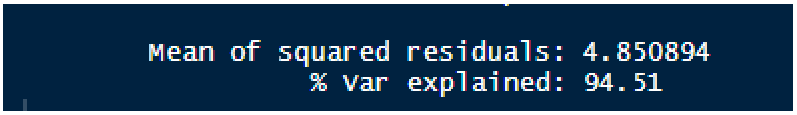
8. Adult Mortality
9. HIV.AIDS
10. Schooling
11. Infant.deaths

4.4.5 Predict model using test data:

The error is less also r square value is 99.9 percent which is near to perfect



```
> RMSE_value
[1] 2.202475
> MAE_value = MAE(test$Life.expec
> MAE_value
[1] 1.508329
> mse1 <- mean((test$Life.expecta
> print(mse1)
[1] 4.850894
> rsquare = 1- sum(test$Life.expe
> rsquare
[1] 0.99993
>
```



```
          Mean of squared residuals: 4.850894
                    % Var explained: 94.51
```

The important variables are same for test data.

5. Model validation

It is important to validate the model on training and test data and check them on accuracy and other performance measures to find the best model.

5.1 Key Performance Measures:

This is regression data and hence we are choosing below performance measure for our model selection:

1. **Mean Absolute Error (MAE):**The Mean Absolute Error measures the average of the absolute difference between each ground truth and the predictions

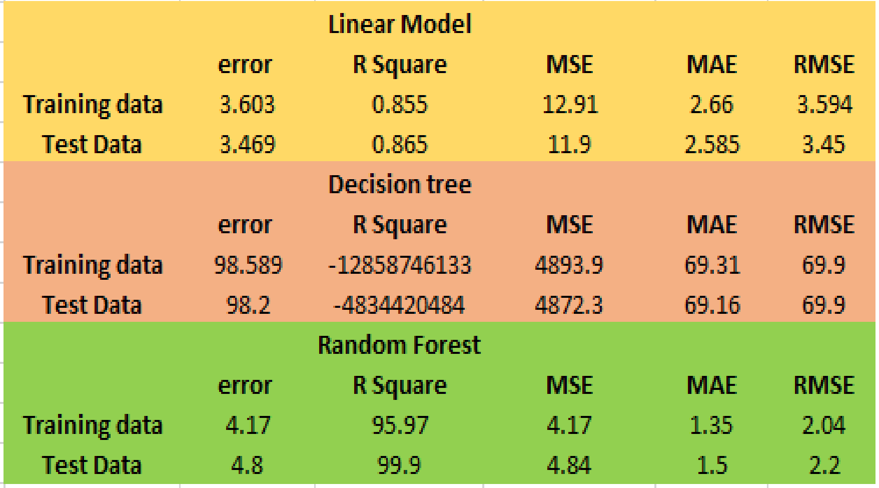
$$MAE = \frac{1}{N}\sum|y_i - \hat{y}_i|$$

2. **Root Mean Squared Error (RMSE):**The Root Mean Squared Error measures the square root of the average of the squared difference between the predictions and the ground truth.

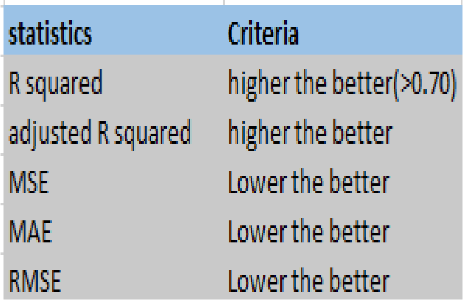
$$RMSE = \sqrt{\frac{1}{N}\sum(y_i - \hat{y}_i)^2}$$

3. **coefficient of determination**, **denoted R2**: is the proportion of the variance in the dependent variable that is predictable from the independent variable(s).
4. **Mean squared error** (**MSE**) measures the average of the squares of the errors—that is, the average squared difference between the estimated values and the actual value

5.2 Model Comparison



		Linear Model			
	error	R Square	MSE	MAE	RMSE
Training data	3.603	0.855	12.91	2.66	3.594
Test Data	3.469	0.865	11.9	2.585	3.45
		Decision tree			
	error	R Square	MSE	MAE	RMSE
Training data	98.589	-12858746133	4893.9	69.31	69.9
Test Data	98.2	-4834420484	4872.3	69.16	69.9
		Random Forest			
	error	R Square	MSE	MAE	RMSE
Training data	4.17	95.97	4.17	1.35	2.04
Test Data	4.8	99.9	4.84	1.5	2.2



statistics	Criteria
R squared	higher the better(>0.70)
adjusted R squared	higher the better
MSE	Lower the better
MAE	Lower the better
RMSE	Lower the better

Based on the performance measures we see that the **decision tree model is not performing well** due to the more no. of dependent variables and hence the model fails in getting important insight and hence comparing linear model and Random Forest Model:

1. R square is high for both random forest(more) and linear model I(should be high))
2. MSE is lesser in random forest as compared to linear model (should be less)
3. RMSE is lesser in random forest as compared to linear model(should be less)
4. Error is one percent less I linear model as compared to random forest(should be less)
5. MAE is lesser in random forest as compared to linear model(should be less)

5.3. Selection of Best Model :

Random Forest is the best model for below reasons:

1. **Based on above key performance measure random forest is the best among all.**
2. Also it is **performing well in both training and test data**.
3. Random forest **uses ensemble technique like bagging** and hence the model is **Robust.**

6. Recommendations and conclusions

6.1. insights and Conclusions

1. Data has lot of missing values, outliers and inconsistencies and hence data treatment is very must here before modeling the data.
2. The Life expectancy in developed countries are higher as compared to developing countries
3. Since it has large number of dependent variables and hence regression technique like decision tree will not help here much
4. Variables such as **BMI** which is not giving correct output should be removed. BMI should decrease for high life expectancy but it is vice versa.
5. The collinear variables should be removed for proper analysis else the accuracy will not be proper.
6. Variable like **population** which is numerical should be important for predicting life expectancy but here it is not and this can be further checked.
7. **Increase in alcohol consumption increases life expectancy** which is quite strange and is collinear with number of years in school. This variable can be further analyzed.

6.2. Selection of five most important variables:

6. **Income composition of resources**: As more the income composition per capita across countries, the life expectancy is also increased.
7. **Adult Mortality:** If Adult Mortality is high then life expectancy will be less.

8. **HIV.AIDS:** If the number of HIV percentage is more the life expectancy will be less.
9. **Schooling :** studies shows that more the number of schooling years, more the life expectancy.
10. **Infant deaths:** In order to increase life expectancy , the health and immunity should be taken care of and infant death should be less.

6.3. Recommendations to business to improve life Expectancy

The reason for low life expectancy is mainly due to poor health conditions, lack of vaccination for deadly diseases, poor standard of living and low income and lack of education.

In order to improve the life expectancy following measures should be taken by countries:

5. Help improving the overall economy (GDP and income composition) of the country which will enhance the income and standard of living of the people and further will improve life expectancy.
6. Improve the health conditions by spending enough amount on healthcare, vaccination and safety against malnutrition.
7. Spreading awareness for diseases like HIV which has no cure and decreases life expectancy.
8. Work toward improving the literacy rate of the country. It has been observed that countries with high literacy rate has high life expectancy.

7.Bibliography and References

The R code is attached in r File along with the attachment and below is the r code used:

7.1. Abbreviations Used:

VIF, MAE, RMSE, MSE, R square, kNN imputation.

7.2. R code used: R code given below

```
library(ROCR)

library(corrplot)

library(car)

library(class)

library(e1071)

library(ggplot2)

library(MASS)

library(nnet)

library(plyr)

library(scatterplot3d)

library(SDMTools)

library(dplyr)

library(ElemStatLearn)

library(rpart)

library(rpart.plot)

library(randomForest)

library(neuralnet)

library(caTools)
library(rpart)

library(rpart.plot)

library(RColorBrewer)

library(data.table)

library(SDMTools)

library(pROC)

library(Hmisc)

library(caret)



setwd("C:/Users/spandey/Desktop")

getwd()

life = read.csv("Life_expectancy.csv", header = TRUE)

summary(life)

#attach(life)



#histogram for numerical variables

hist(Year, data= life)

hist(Life.expectancy, data= life)

hist(Adult.Mortality, data= life)

hist(infant.deaths, data= life)

hist(Alcohol, data= life)

hist(percentage.expenditure, data= life)

hist(Hepatitis.B, data= life)

hist(under.five.deaths, data= life)

hist(BMI, data= life)

hist(Diphtheria, data= life)

hist(HIV.AIDS, data= life)

hist(thinness..1.19.years, data= life)

hist(thinness.5.9.years, data= life)

hist(Income.composition.of.resources, data= life)

hist(Schooling, data= life)
hist(Polio, data= life)
hist(Measles, data= life)
hist(GDP, data= life)
hist(Total.expenditure, data= life)
hist(Population, data= life)
qplot(Country, data= life)

#histogram for categorical variables
qplot(Status, data= life)
#bar plot for categorical variables
qplot(Country, data= life, geom = "bar")
qplot(Status, data= life, geom = "bar")

## bivariate analysis
plot(Adult.Mortality,Life.expectancy, data= life,col="blue")
plot(infant.deaths,Life.expectancy, data = life,col="blue")
plot(Alcohol,Life.expectancy, data= life,col="blue")
plot(percentage.expenditure,Life.expectancy, data= life,col="blue")
plot(Hepatitis.B,Life.expectancy, data= life,col="blue")
plot(under.five.deaths,Life.expectancy, data= life,col="blue")
plot(BMI,Life.expectancy, data= life,col="blue")
plot(Diphtheria,Life.expectancy, data= life,col="blue")
plot(HIV.AIDS,Life.expectancy, data= life,col="blue")
plot(thinness..1.19.years,Life.expectancy, data= life,col="blue")
plot(thinness.5.9.years,Life.expectancy, data= life,col="blue")
plot(Income.composition.of.resources,Life.expectancy, data= life,col="blue")
plot(Schooling,Life.expectancy, data= life,col="blue")
plot(Polio,Life.expectancy , data= life,col="blue")
plot(Measles,Life.expectancy,  data= life,col="blue" )
plot(GDP,Life.expectancy , data= life,col="blue")
plot(Total.expenditure,Life.expectancy , data= life,col="blue")
plot(Population,Life.expectancy, data= life,col="blue" )
```

```r
#boxplot
names(life)
boxplot(life)
boxplot(life[,-c(18)])
boxplot(Measles)
boxplot(infant.deaths)
boxplot(Life.expectancy)
boxplot(Adult.Mortality )
boxplot(Alcohol )
boxplot(percentage.expenditure)
boxplot(Hepatitis.B)
boxplot(under.five.deaths)
boxplot(Diphtheria)
boxplot(HIV.AIDS)
boxplot(thinness..1.19.years)
boxplot(thinness.5.9.years)
boxplot(Income.composition.of.resources)
boxplot(Schooling)
boxplot(Polio)
boxplot(GDP)
boxplot(Total.expenditure)
boxplot(Population)
# outlier Treatment:

#As we can find outliers in numerical variables only so we create 2 subsets of dataset with only numerical variables and categorical variables respectively.

#creating subset of dataset

# continuous attributes
names(life)
```

```
cat(" The column names which are numeric in nature are :",names(life)[which(sapply(life, is.numeric))])

# discrete attributes
cat("\n The column names which are categorical in nature are :",names(life)[which(sapply(life,is.factor))])

#creating datasets of only factor variable and only numeric variable for EDA
life_Num = life[,c(4:22)]
life_Fact = life[,c(1,3)]
cat("\n Number of columns in subset containing numerical variables :", ncol(life_Num))
cat("\n Number of columns in subset containing categorical variables :", ncol(life_Fact))

# Boxplot to check outliers
boxplot(life_Num)

#We can see that outliers are present in the data.
## Finding List of Outliers

list("OutLiers")
OutLiers <- life_Num
for (i in c(1:19)) {

  Box_Plot <- boxplot(life_Num[,i],plot = F)$out
  OutLiers[,i] <- NA

  if (length(Box_Plot)>0) {
    OutLiers[(1:length(Box_Plot)),i] <- Box_Plot
  }
}

OutLiers <- OutLiers[(1:19),]
OutLiers
```

```
#The above table shows the list of outliers in the dataset. We can see that all numeric variables contain outliers i.e. "AccountWeeks","DataUsage", "DayMins","DayCalls","MonthlyCharge","OverageFee" and "RoamMins".

#We assume that the data is authentic and all the values are practical in this case.

#We can opt to keep outliers in the data as it will not affect the analysis.

#We can also move the outliers to the nearest 1st quartile or the 3rd quartile.

#But it will add bias to the modelling.


# Treating Outliers to the 1st or 3rd Quartile


#Since an outlier is considered so if it is below the first quartile -1.5?IQR or above third quartile + 1.5?IQR. So, making a custom function accordingly.

capOutlier <- function(x){

  qnt <- quantile(x, probs=c(.25, .75), na.rm = T)

  caps <- quantile(x, probs=c(.05, .95), na.rm = T)

  H <- 1.5 * IQR(x, na.rm = T)

  x[x < (qnt[1] - H)] <- caps[1]

  x[x > (qnt[2] + H)] <- caps[2]

  return(x)

}

#Way to use this custom function will be:

#df$colName=capOutlier(df$colName)


life_Otlr= life[,]


#Using the above custom function to treat the outliers now in each of the columns:


for (i in names(life_Otlr))

  #for (i in  colnames_list)

{

  if (sapply(life_Otlr[,i], class) == "numeric")

  {

    life_Otlr[i] <- capOutlier(life_Otlr[[i]])

    cat("\n Outliers treated in numeric column  : ", i)
```

```
 }
 else
 {
   cat("\n Outlier treatment not applicable for Non-numeric column  : ", i)
 }
}

#boxplot of treated dataset
life_Otlr_Num = life_Otlr[,c(4:22)]
boxplot(life_Otlr_Num)



#We can see that outliers have been removed in the dataset.
#But removing outliers from the data will not be a good option as it will add bias to the data.
#Also data is assumed to be authentic so we will not remove outliers from the data.

#We will work on Treated_Data instead of Treated_Otlr_Data

#treat missing values and negative values in data set
colSums(is.na(life_Otlr_Num))
new_data= life_Otlr_Num
library(DMwR)
sum(is.na(life_Otlr_Num))
Life_NO_NA = knnImputation(life_Otlr_Num, k=5)
Life_NO_NA$Status= life$Status
Life_NO_NA$Year= life$Year
sum(is.na(Life_NO_NA))
summary(Life_NO_NA)
attach(Life_NO_NA)
```

```
#check multicollinearity using vif factor (take full datset)
library(car)
names(Life_NO_NA)
linear1= Life.expectancy ~ Year+Adult.Mortality+Alcohol+percentage.expenditure+Hepatitis.B+
  under.five.deaths+Diphtheria+HIV.AIDS+thinness.5.9.years+Income.composition.of.resources+
  Schooling+infant.deaths+Total.expenditure+Measles+Population+Polio+GDP+thinness..1.19.years
full = lm(linear1, data = life)
summary(full)
vif(full)

# removing multicollinear variables
linear2= Life.expectancy ~ Year+Adult.Mortality+Alcohol+Hepatitis.B+
  Diphtheria+HIV.AIDS+thinness.5.9.years+Income.composition.of.resources+
  Schooling+infant.deaths+Total.expenditure+Measles+Population+Polio+GDP
full = lm(linear2, data = life)
summary(full)
vif(full)



#correlation plot

library(corrplot)
names(life)
life1 = na.omit(life)
summary(life1)
scatter1 = cor(life1[,c(-1,-3,-4,-8,-11,-12,-19)],method = c("pearson","kendall","spearman"))
corrplot(scatter1,type = "upper", tl.pos = "td",
         method = "circle", tl.cex = 0.5, tl.col = 'black',
         order = "hclust", diag = FALSE)


# Data Transformation:
```

```
Life_NO_NA$Status= as.factor(Life_NO_NA$Status)

Life_NO_NA$Status = ifelse(Life_NO_NA$Status =="Developing",1,0)

head(Life_NO_NA$Status)

# But before that, we will normalize


#if you don't scale, than the betas/coefficient values are not meaningful.

names(Life_NO_NA)

Life_NO_NA$norm.Year<-scale(Year)

Life_NO_NA$norm.Status<-scale(Life_NO_NA$Status)

Life_NO_NA$norm.Adult.Mortality<-scale(Life_NO_NA$Adult.Mortality)

Life_NO_NA$norm.Alcohol<-scale(Life_NO_NA$Alcohol)

Life_NO_NA$norm.Polio<-scale(Life_NO_NA$Polio)

Life_NO_NA$norm.Diphtheria<-scale(Life_NO_NA$Diphtheria)

Life_NO_NA$norm.HIV.AIDS<-scale(Life_NO_NA$HIV.AIDS)

Life_NO_NA$norm.GDP<-scale(Life_NO_NA$GDP)

Life_NO_NA$norm.thinness.5.9.years<-scale(Life_NO_NA$thinness.5.9.years)

Life_NO_NA$norm.Income.composition.of.resources<-scale(Life_NO_NA$Income.composition.of.resources)

Life_NO_NA$norm.Schooling<-scale(Life_NO_NA$Schooling)


Life_NO_NA$norm.Measles<-scale(Life_NO_NA$Measles)

Life_NO_NA$norm.Population<-scale(Life_NO_NA$Population)

Life_NO_NA$norm.Hepatitis.B<-scale(Life_NO_NA$Hepatitis.B)

Life_NO_NA$norm.Total.expenditure<-scale(Life_NO_NA$Total.expenditure)

Life_NO_NA$norm.infant.deaths<-scale(Life_NO_NA$infant.deaths)

#Life_NO_NA$norm.Life.expectancy<-scale(Life_NO_NA$Life.expectancy)

summary(Life_NO_NA)

names(Life_NO_NA)


#dividing data into training and test ( keeping target varible and scaled data and status variable)

set.seed(144)

names(Life_NO_NA)

spl = sample.split(Life_NO_NA, SplitRatio = 0.7)

train = subset(Life_NO_NA[,-c(2:21)], spl== T)
```

```
dim(train)
head(train)
test = subset(Life_NO_NA[,-c(2:21)], spl== F)
dim(test)
head(test)

###################applying models and measuring performance###############3

#########  Multiple Linear model  #################

#Division of training and test data for linear model (non scaled)
set.seed(144)
names(Life_NO_NA)
spl = sample.split(Life_NO_NA, SplitRatio = 0.7)
train_linear = subset(Life_NO_NA[,-c(5,8,9,16,22:37)], spl== T)
dim(train_linear)
head(train_linear)
test_linear = subset(Life_NO_NA[,-c(5,8,9,16,22:37)], spl== F)
dim(test_linear)
head(test_linear)
# removing multicollinear variables
linear2= Life.expectancy~.
full = lm(linear2, data = train_linear)
summary(full)
vif(full)

# removing multicollinear variables
linear3=Life.expectancy ~ train_linear$Adult.Mortality + train_linear$Year+train_linear$Alcohol+train_linear$Diphtheria+train_linear$Hepatitis.B+train_linear$HIV.AIDS+train_linear$thinness.5.9.years+train_linear$Schooling+train_linear$GDP+train_linear$Income.composition.of.resources+train_linear$Status+ train_linear$Polio

full = lm(linear3, data = train_linear)
summary(full)
```

```
vif(full)


# removing insignificant  variables anf final model

linear4=Life.expectancy ~
train_linear$Adult.Mortality+train_linear$Diphtheria+train_linear$HIV.AIDS+train_linear$thinness.5.9.years+train_linear$Scho
oling+train_linear$GDP+train_linear$Income.composition.of.resources+train_linear$Status+train_linear$Polio


full = lm(linear4, data = train_linear)

summary(full)

plot(linear4)

pln = predict(full, train_linear)

pln

mse1 <- mean((train_linear$Life.expectancy-pln)^2)

print(mse1)

lin_rmse= RMSE(train_linear$Life.expectancy,pln)

Lin_mae= MAE(train_linear$Life.expectancy,pln)

Lin_mae

lin_rmse


#validation on test data


linear4=Life.expectancy ~
test_linear$Adult.Mortality+test_linear$Diphtheria+test_linear$HIV.AIDS+test_linear$thinness.5.9.years+test_linear$Schooling
+test_linear$GDP+test_linear$Income.composition.of.resources+test_linear$Status+test_linear$Polio


full = lm(linear4, data = test_linear)

summary(full)

plot(linear4)

pln = predict(full, test_linear)

pln

mse1 <- mean((test_linear$Life.expectancy-pln)^2)

print(mse1)

lin_rmse= RMSE(test_linear$Life.expectancy,pln)

Lin_mae= MAE(test_linear$Life.expectancy,pln)
```

```
Lin_mae

lin_rmse



#########   Random Forest  ##############


library(randomForest)

library(neuralnet)

set.seed(144)

head(train)

rndFor = randomForest(train$Life.expectancy~., data = train[,-1],

              ntree=501, mtry = 5, nodesize = 10,

              importance=TRUE)

rndFor

print(rndFor)

print(rndFor$importance)

plot(rndFor)

importance(rndFor)

#tunning

set.seed(144)

tRndFor = tuneRF(x = train[,-c(1)],

          y= train$Life.expectancy,

          mtryStart = 5,

          ntreeTry = 51,

          stepFactor = 1.5,

          improve = 0.0001,

          trace=TRUE,

          plot = TRUE,

          doBest = TRUE,

          nodesize = 10,

          importance=TRUE

)

tRndFor
```

```
# tuned random forest
set.seed(144)
head(train)
rndFor = randomForest(train$Life.expectancy~., data = train[,-1],
                    ntree=51, mtry = 10, nodesize = 10,
                    importance=TRUE)
rndFor
print(rndFor)
print(rndFor$importance)
plot(rndFor)
importance(rndFor)

#Lets make predictions on the training data and measure the prediction error rate.
dim(train)
train$predict.class = predict(rndFor, data=train, type="class")


#accuracy:
RMSE_value = RMSE(train$Life.expectancy,train$predict.class)
RMSE_value
MAE_value = MAE(train$Life.expectancy,train$predict.class)
MAE_value
mse1 <- mean((train$Life.expectancy-train$predict.class)^2)
print(mse1)
rsquare = 1- sum(train$Life.expectancy-train$predict.class)^2/sum((train$Life.expectancy-mean(train$Life.expectancy))^2)
rsquare

#Validation on test data
# tuned random forest
set.seed(144)
head(test)
rndFor = randomForest(test$Life.expectancy~., data = test[,-1],
                    ntree=51, mtry = 10, nodesize = 10,
                    importance=TRUE)
rndFor
plot(rndFor)
importance(rndFor)

#Lets make predictions on the testing data and measure the prediction error rate.
dim(test)
test$predict.class = predict(rndFor, data=test, type="class")



#accuracy:
RMSE_value = RMSE(test$Life.expectancy,test$predict.class)
RMSE_value
MAE_value = MAE(test$Life.expectancy,test$predict.class)
MAE_value
mse1 <- mean((test$Life.expectancy-test$predict.class)^2)
print(mse1)
rsquare = 1- sum(test$Life.expectancy-test$predict.class)^2/sum((test$Life.expectancy-mean(test$Life.expectancy))^2)
rsquare

#########  Decision tree #################
library(NbClust)
library(rpart)
library(rpart.plot)



DT = train$Life.expectancy~.
tree <- rpart(DT, data = train, method = "class", cp=0, minbucket=20)
tree
rpart.plot(tree)

#The cost complexity table can be obtained using the printcp or plotcp functions
```

```
printcp(tree)


plotcp(tree)


#The unncessarily complex tree above can be pruned using a cost complexity threshold. Using a complexity threshold of 0.015 gives us a much simpler tree.

ptree = prune(tree, cp=  0.0029,"CP")

printcp(ptree)

ptree

rpart.rules(ptree)

rpart.plot(ptree)


#as per our model, we are finding  the prediction of our Y variable

train$prediction = predict(ptree, data=train, type="prob")

train$prediction

head(train)


######### performance measures #######


RMSE_value = RMSE(Life.expectancy,train$prediction)

RMSE_value

MAE_value = MAE(train$Life.expectancy,train$prediction)

MAE_value

mse1 <- mean((train$Life.expectancy-train$prediction)^2)

print(mse1)

rsquare = 1- sum(train$Life.expectancy-train$prediction)^2/sum((train$Life.expectancy-mean(train$Life.expectancy))^2)

rsquare


#############validation on test data###########################

DT = test$Life.expectancy~.

tree_test <- rpart(DT, data = test, method = "class", cp=0.0029, minbucket=20)

tree_test

printcp(tree_test)
```

```
rpart.plot(tree_test)
test$prediction = predict(tree_test, data=test, type="prob")
test$prediction
head(test)

######### performance measures #######

RMSE_value = RMSE(Life.expectancy,test$prediction)
RMSE_value
MAE_value = MAE(test$Life.expectancy,test$prediction)
MAE_value
mse1 <- mean((test$Life.expectancy-test$prediction)^2)
print(mse1)
rsquare = 1- sum(test$Life.expectancy-test$prediction)^2/sum((test$Life.expectancy-mean(test$Life.expectancy))^2)
rsquare
```