



Prediction of WHO Life Expectancy Data Using Multiple Linear Regression And Artificial Neural Network

¹Sambaran Saha, ²Sulagna Ray, ³Plabanjit Karmakar, ⁴Rupam Saha

¹Data Analyst, Registration Department, Brainware University, Kolkata, West-Bengal

²Subject Matter Expert, Department of Statistics, Evelyn Learning Systems, New Delhi, Delhi

³Senior Analyst, Ipsos MMA, Bangalore, Karnataka

⁴Assistant Teacher, Dept of Mathematics, M.C. Kejriwal Vidyapeeth, Howrah, West-Bengal

Abstract:

Life expectancy is the key for assessing the population health. A lot of researches have already been done based on the factors of life expectancy data considering demographic variables, income composition and mortality rates. Researches on multiple linear regression based on a data set of one year for all the countries are available.

Now we can build a predictive model based on this data using multiple linear regression from the data set of 2000-2015 of all countries by dividing the whole dataset into two parts namely training and testing set in ratio 80:20. After fitting the model in training data and checking the model performance on both dataset we found that MLR is a very good fit for prediction, but it violates normality and homoscedasticity assumptions. To overcome this problem, we fitted this data on ANN model in the same way like before which does not assume those assumptions. We found that ANN model performed even better than MLR. At last, we compared their performances and tried to find pros and cons of both models.

Keywords: Life Expectancy Multiple Linear Models, Artificial Neural Network Machine Learning, R, Coefficient of Determination, Prediction, MSE, MAPE, Accuracy.

I. INTRODUCTION

Life Expectancy is a statistical measure of the average time an organism is expected to live, based on the year of its birth, its current age and other demographic factors including gender. In other words, it is the statistical age that a person is expected to live until, based on actuarial data. Life expectancy at birth (LEB) is the most commonly used measure worldwide.

Assuming that a person maintains the minimum standard of hygiene, life expectancy may be the outcome of multiple factors such as Diet, Lifestyle, Heredity/genetic, Marital status, Local standard of healthcare and disease prevention, Access to clean water and fresh food supply, Community and social support, Absence of adverse environmental factors like pollution

To compute life expectancy, it's a bit difficult & uses lot of different data sources for each country. Life expectancy is calculated through life table or life table calculator. The life table incorporates data with age

specific death rates for the population in question which requires enumeration of data for the number of people & number of deaths at each age for that population. Most currently available life table or life expectancy calculators are based on a person's age, gender & race. According to actuarial science, life expectancy takes into account several individual-level as well as population-level factors to arrive at a figure.

The population of many rich developed countries has life expectancy around 80 years whereas the life expectancy of poor developing country has life expectancy around 50 to 60 years.

There are many uses for it in the financial world, including life insurance, pension planning, and social security benefits. The most important use of life expectancy is in clinical researches. Life expectancy is used in pricing and underwriting life insurance and insurance products like annuities, as well as in retirement and pension planning.

II. DATA DESCRIPTION

The dataset we used for our project was provided by the Kaggle.com website. It is a free open source platform to all. The data was collected from WHO and United Nations website with the help of Deeksha Russell and Duan Wang. This dataset has usability rating 8.2. The data consists of 2938 rows and 22 columns. As the dataset is from WHO, we assume there is no error. Dataset Link: <https://www.kaggle.com/kumarajarshi/life-expectancy-who>

Response Variable:

Y: Life expectancy – Life Expectancy in age

Regressors:

Country – Country

Year – Year

X1: Status – Developed or developing status

X2: Adult Mortality – Adult Mortality Rates of both sexes (probability of dying between 15 and 60 years per 1000 population)

X3: Infant deaths – Number of Infant Deaths per 1000 population

X4: Alcohol – Alcohol, recorded per capita (15+) consumption (in litres of pure alcohol)

X5: Percentage expenditure – Expenditure on health as a percentage of Gross Domestic Product per capita (%)

X6: Hepatitis B – Hepatitis B (HepB) immunization coverage among 1-year-olds (%)

X7: Measles – Number of reported Measles cases per 1000 population

X8: BMI – Average Body Mass Index of entire population

X9: Under-five deaths – Number of under-five deaths per 1000 population

X10: Polio – Polio (Pol3) immunization coverage among 1-year-olds (%)

X11: Total expenditure – General government expenditure on health as a percentage of total government expenditure (%)

X12: Diphtheria – Diphtheria tetanus toxoid and pertussis (DTP3) immunization coverage among 1-year-olds (%)

X13: HIV/AIDS – Deaths per 1 000 live births HIV/AIDS (0-4 years)

X14: GDP – Gross Domestic Product per capita (in USD)

X15: Population – Population of the country

X16: Thinness 1-19 years – Prevalence of thinness among children and adolescents for Age 10 to 19 (%)

X17: Thinness 5-9 years – Prevalence of thinness among children for Age 5 to 9 (%)

X18: Income composition of resources – Human Development Index in terms of income composition of resources (index ranging from 0 to 1)

X19: Schooling – Number of years of Schooling (years)

```
> str(data)
'data.frame': 2938 obs. of 22 variables:
 $ Country      : chr "Afghanistan" "Afghanistan" "Afghanistan" "Afghanistan" ...
 $ Year         : int 2015 2014 2013 2012 2011 2010 2009 2008 2007 2006 ...
 $ Status       : chr "Developing" "Developing" "Developing" "Developing" ...
 $ 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 0.01 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 ...
> |
```

III OBJECTIVE

The World Health Organization (WHO) keeps track of the health status and many others related factors for all countries, as well. This dataset for 193 countries has been collected from the WHO data. In this project we consider data from year 2000-2015 for 193 countries for analysis. As the dataset is from WHO there is no error in the data.

In the respective project we are going to find out the following factors:

- Whether the initially chosen predicting factors really affects the life expectancy or not & those predicting variables which actually affects the life expectancy.
- The type of correlation between life expectancy and other factors like smoking, exercise, lifestyles, alcohol consumptions etc.
- The impact of immunization coverage on life expectancy.

- Whether we can build a reliable predictivemodel to predict life expectancy in future or for any external dataset.

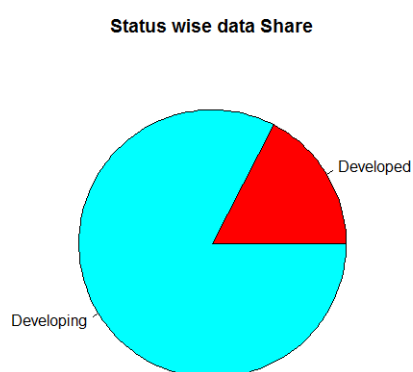
For doing the above we at first will build up a multiple linear regression model which is a supervised ML Model, based on Training data by dividing the whole data set in training set and test set randomly in ratio 80:20. Then we will check the multicollinearity, outliers, normality, and homoscedasticity. We will find out how well the model fits to the training data and how it predicts the test data by coefficient of determination along with MSE, MAPE and Accuracy. Then we will apply a Artificial Neural Network on the train data and check the model adequacy just like before. At last, we will compare both models and conclude our findings accordingly.

IV. THE R ENVIRONMENT

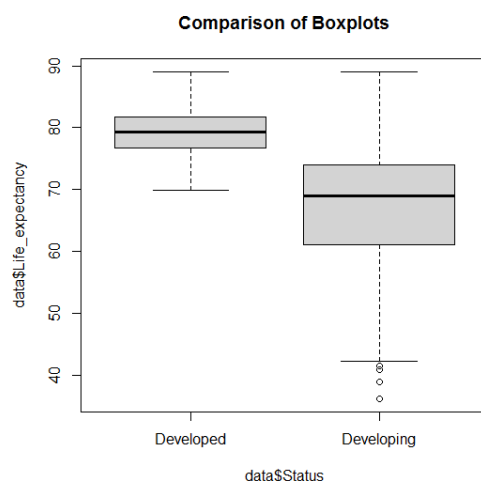
Here we are using R language for our analysis. R is a language and environment for statistical computing and graphics. R provides a wide variety of statistical (linear and nonlinear modeling, classical statistical tests, time- series analysis, classification, clustering ...) and graphical techniques, and is highly extensible.

R is one of the major languages for data science. It provides excellent visualization features, which is essential to explore the data before submitting it to any automated learning, as well as assessing the results of the learning algorithm. Many R packages for machine learning are available off the shelf and many modern methods in statistical learning are implemented in R as part of their development.

V. DATA INSIGHTS

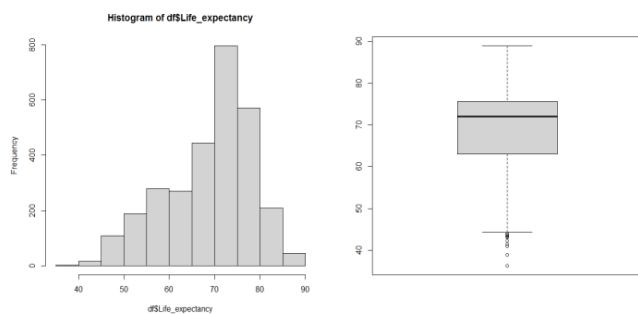
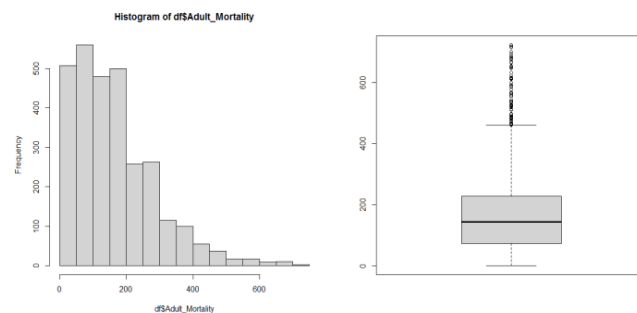
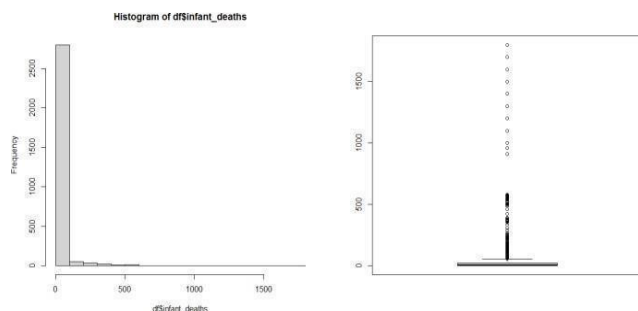
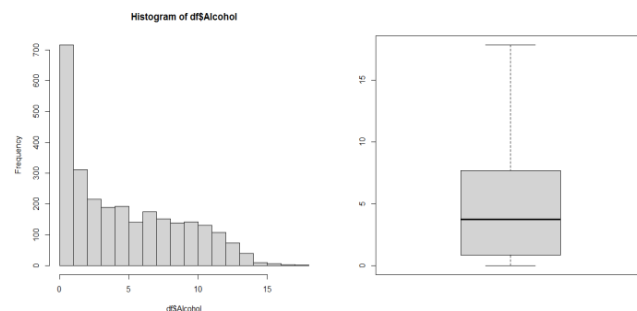
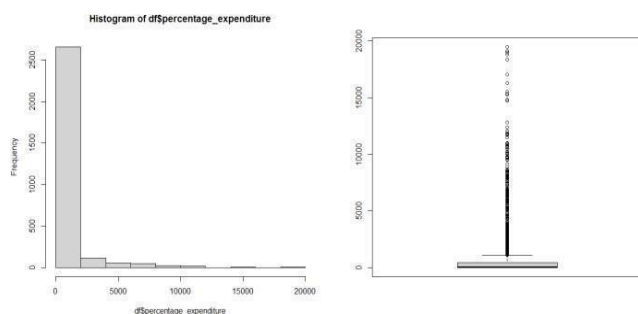
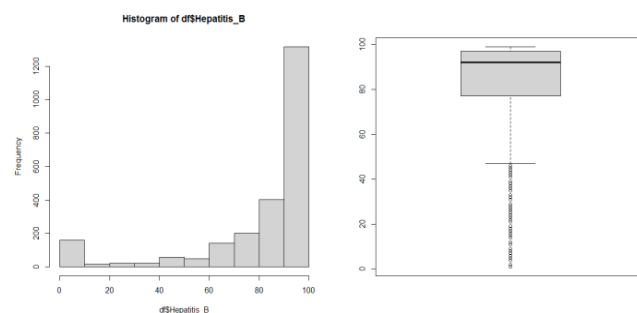
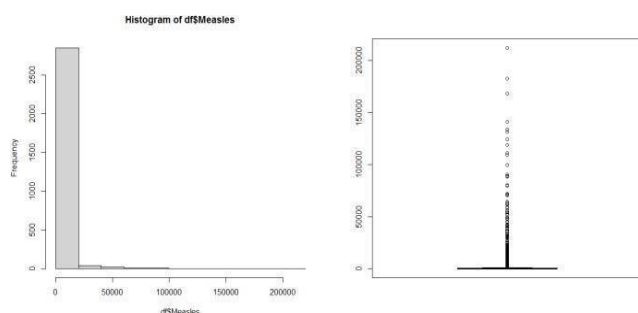
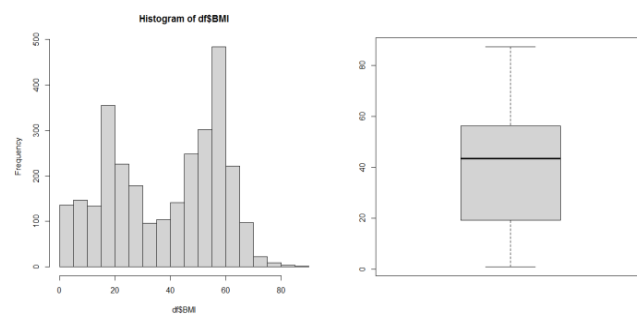
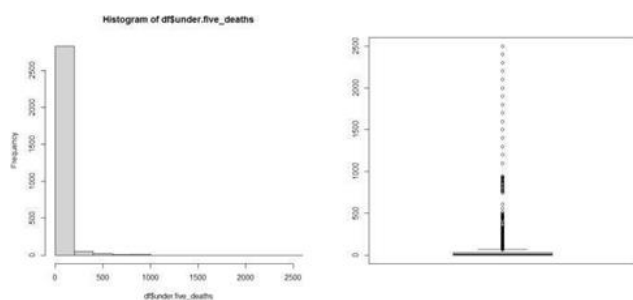
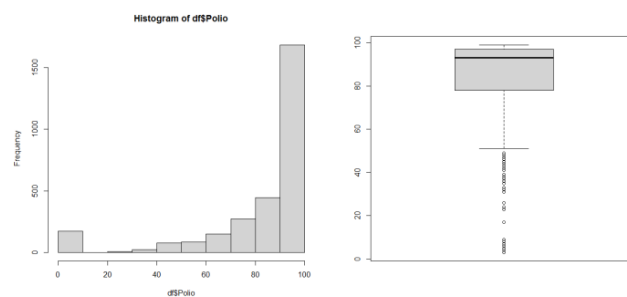


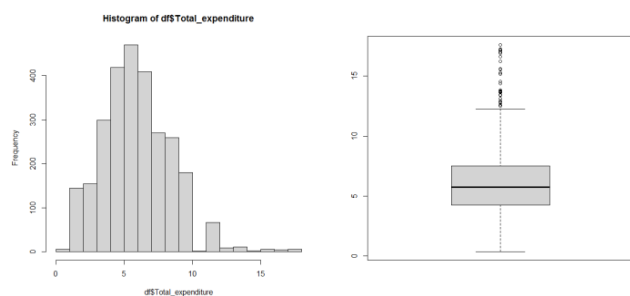
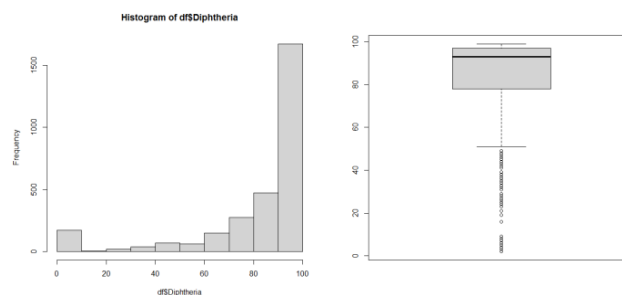
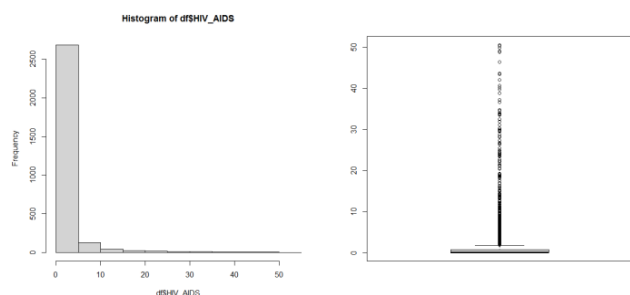
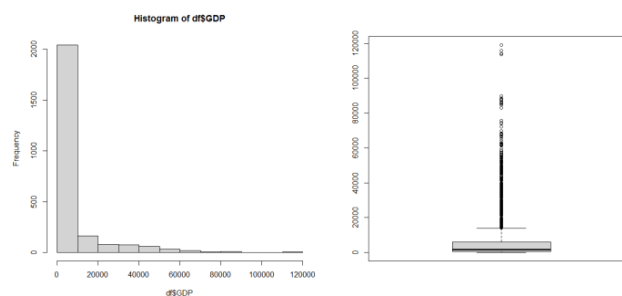
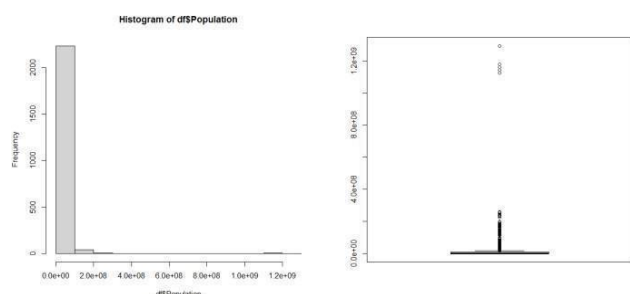
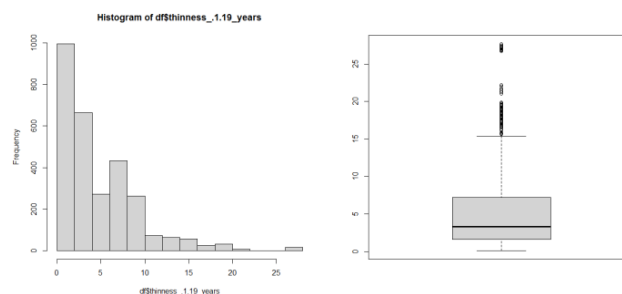
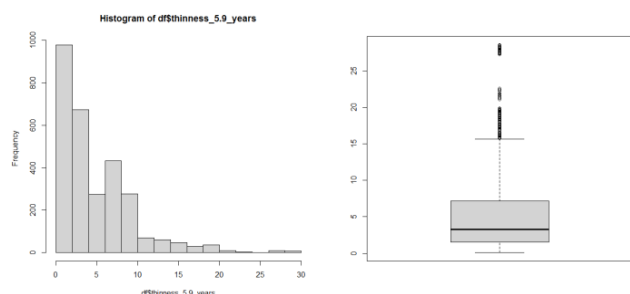
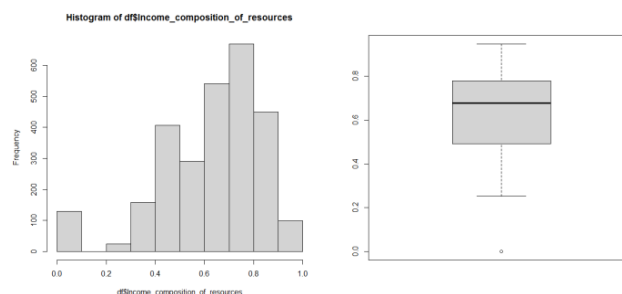
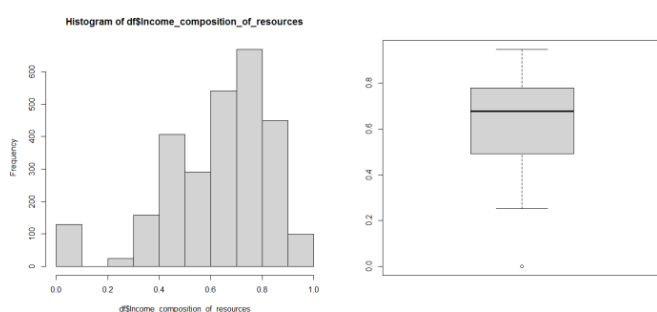
82.57318 % of total dataset comes from developing countries and 17.42682 % data comes from developed countries.



Note that avg. life expectancy in developed countries is around 80 years and spread is less compared to developing countries whereas in developing countries avg. life expectancy is around 70 years. There are some outliers in developing data points.

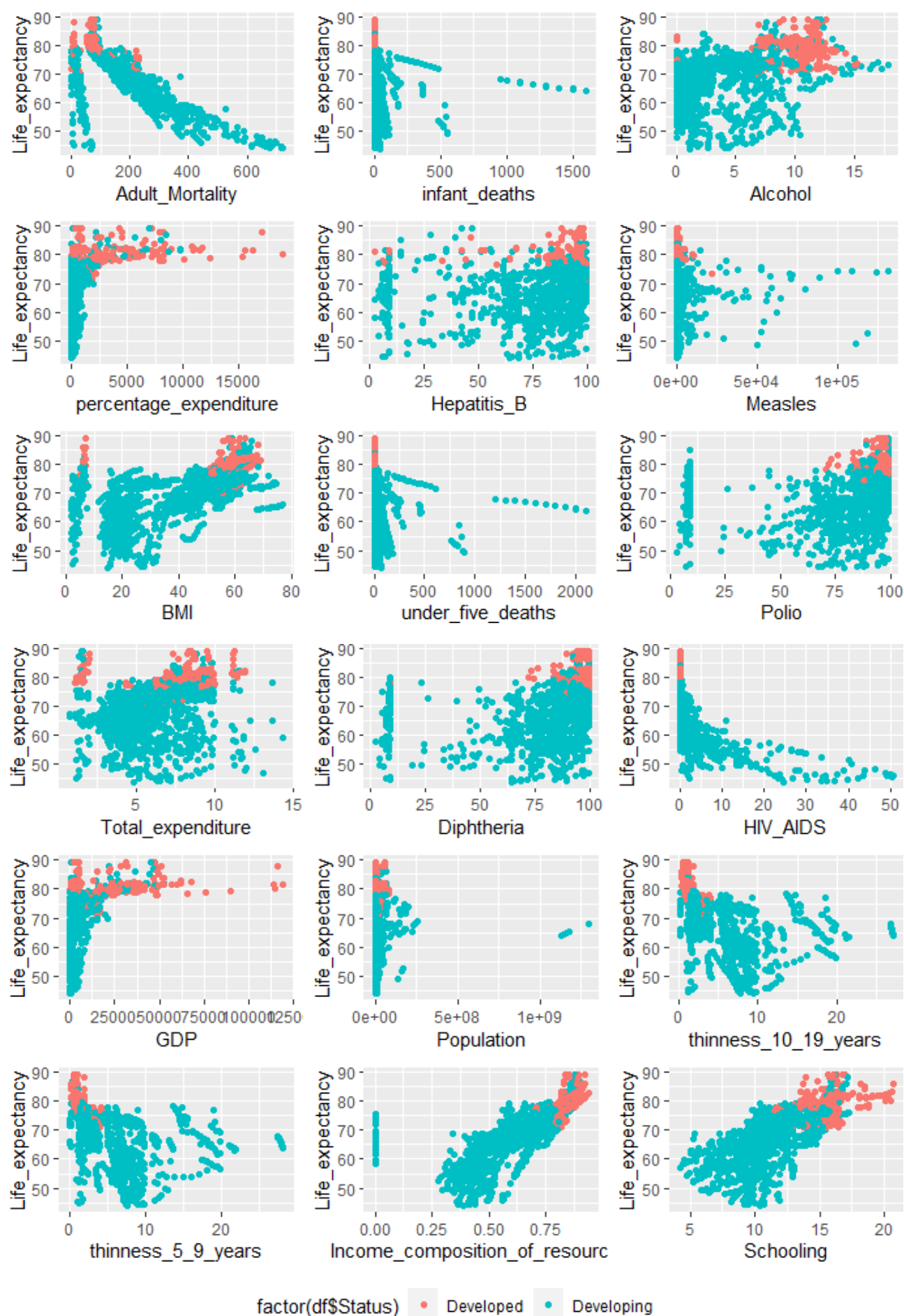
Now let us take a look into the data visualizations. Below we have histogram and boxplot of each variable:

Y: Life expectancy**X2: Adult Mortality****X3: Infant deaths****X4: Alcohol****X5: Percentage expenditure****X6: Hepatitis B****X7: Measles****X8: BMI****X9: Under-five deaths****X10: Polio**

X11: Total expenditure**X12: Diphtheria****X13: HIV/AIDS****X14: GDP****X15: Population****X16: Thinness 1-19 years****X17: Thinness 5-9 years****X18: Income composition of resources****X19: Schooling**

We can see most of the variables are highly skewed and contain too much outlier.

Now we will draw individual scatterplot to visualize the relationship of each regressor with the response variable. The blue points correspond to developing country and red points belong to developed country.



VI. DATA PREPROCESSING

For machine learning algorithms to work it is necessary to convert the raw data into a clean data set and dataset must be converted to numeric data. You have to encode all the categorical labels to column vectors with binary values. Missing values or NaNs in the dataset are an annoying problem. You have to either drop the missing rows or fill them up with mean or interpolated values. Both the training data and test data must have same dimensions for the model.

Dropping Columns which are not useful

At first step we will drop some of the columns which may not contribute much to our machine learning model such as Country, Year.

Renaming Columns

For ease of our study and modeling we have renamed a few columns such as: "Life.expectancy" to "Life_expectancy"; or "thinness..1.19.years" to "thinness_10_19_years" etc.

Dealing with Missing values

```
> sapply(data, function(x) sum(is.na(x))/length(x)*100)
      Life_expectancy      Status
      0.3403676      0.0000000
      Adult_Mortality      infant_deaths
      0.3403676      0.0000000
      Alcohol      percentage_expenditure
      6.6031314      0.0000000
      Hepatitis_B      Measles
      18.8223281      0.0000000
      BMI      under_five_deaths
      1.1572498      0.0000000
      Polio      Total_expenditure
      0.6466984      7.6923077
      Diphtheria      HIV_AIDS
      0.6466984      0.0000000
      GDP      Population
      15.2484683      22.1919673
      thinness_10_19_years      thinness_5_9_years
      1.1572498      1.1572498
      Income_composition_of_resources      Schooling
      5.6841389      5.5479918
```

We can see that a lot of values are missing in this data. If we delete those rows completely, we might lose a large amount of data. So instead of removing the null values completely, we have used data imputation technique. We have replaced the missing cells by interpolation technique. At last, we have removed remaining null values.

VII. METHOD OF ANALYSIS

A. MULTIPLE LINEAR REGRESSION

Here our multiple linear model is $Y = X\beta + \epsilon$

Where,

Y = Vector of responses

X = Design matrix (Vector of Regressors)

β = Vector of unknown parameters

ϵ = Error of linear model

Assumptions:

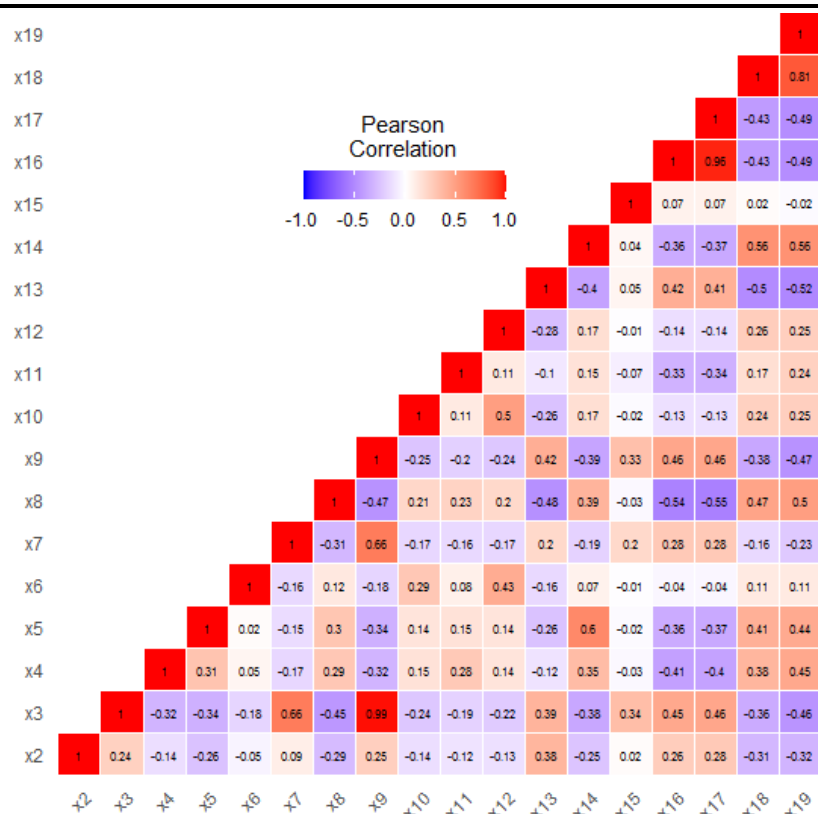
- Errors are i.i.d. distributed with $E(\epsilon) = 0$ & $\text{var}(\epsilon) = \sigma^2 I_n$.
- Explanatory variables are independent i.e. X is of full column rank.

Data Transformation:

From the Histogram and boxplot of the Regressors it is evident that most of them are highly skewed. Skewed data is cumbersome and common. Different features in the data set may have values in different ranges. Data transformation comes to our aid in such situations. Here we have used log transformation and cube root transformation for skewed data.

Heatmap & Correlation Matrix:

A good way to quickly check correlations among columns is by visualizing the correlation matrix as a heatmap.



We see there is high correlation between x18-x19, x16-x17 & x3-x9. So, we have removed x19, x16 & x9 from the data frame.

Detection & Removal of Outlier:

In statistics, an outlier is defined as an observation which stands far away from the most of other observations. Often an outlier is present due to the measurements error. Therefore, one of the most important tasks in data analysis is to identify and (if is necessary) to remove the outliers. Outliers can occur by chance in any distribution, but they often indicate either measurement error or that the population has a heavy-tailed distribution. In the former case one wishes to discard them or use statistics that are robust to outliers, while in the latter case they indicate that the distribution has high skewness and that one should be very cautious in using tools or intuitions that assume a normal distribution. There is no rigid mathematical definition of what constitutes an outlier; determining whether or not an observation is an outlier is ultimately a subjective exercise. There are various methods of outlier detection.

Cook's Distance:

D_i is used in Regression Analysis to find the influential outliers in a set of predictor variables. In other words, it's a way to identify points that negatively affect your regression model. The measurement is a combination of each observation's leverage and residual values; the higher the leverage and residuals, the higher the Cook's distance.

Technically, Cook's D_i is calculated by removing the i th data point from the model and recalculating the regression. It summarizes how much all the values in the regression model change when the i th observation is removed. The formula for Cook's distance is:

$$D_i = \frac{\sum_{j=1}^n (\hat{Y}_j - \hat{Y}_{j(i)})^2}{(p+1)\hat{\sigma}^2}$$

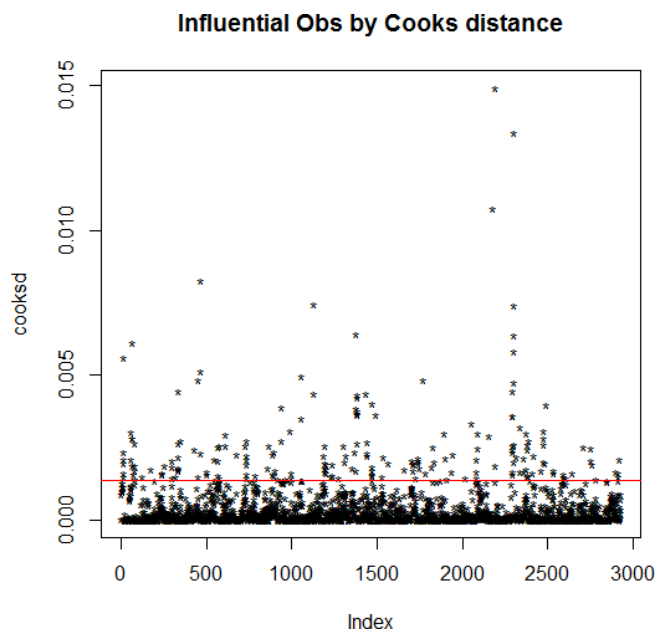
Where, $\hat{Y}_{j(i)}$ is the fitted response value obtained when excluding i .

Decision Rule:

If (cook's distance) $D_i > 4/n$

Then we conclude that point is outlier.

Where, n = no of rows in dataset.



The points above the red line are considered to be outliers.

After this we have divided the whole dataset in training and test set randomly in the ratio 80:20. Then we have deployed MLR on the training dataset.

Multicollinearity:

Variance Inflation Factor: Consider the following linear model with k independent variables:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p + \varepsilon.$$

$$VIF = 1/(1-R_j^2), \text{ for all } j = 1(1)p$$

Where, R_j^2 is the coefficient of determination when X_j is regressed on the remaining regressor. High value of R_j^2 suggests that X_j is well explained by the remaining set of regressor.

$VIF > 2$ is considered large & it suggests the presence of multicollinearity.

```
> vif(fit)
      x1      x2      x3      x4      x5      x6      x7      x8
1.758423 1.283388 2.653781 1.560637 1.820716 1.350836 1.773769 1.777512
      x10     x11     x12     x13     x14     x15     x17     x18
1.348084 1.258930 1.579042 1.859377 2.112175 1.205035 1.982151 2.234704
```

Variable Selection:

Forward, Backward & Stepwise Regression: The stepwise regression (or stepwise selection) consists of iteratively adding and removing predictors, in the predictive model, in order to find the subset of variables in the data set resulting in the best performing model that is a model that lowers prediction error.

There are three strategies of stepwiseregression (James et al. 2014, P. Bruce and Bruce (2017)):

Forward selection, which starts with no predictors in the model, iteratively adds the most contributive predictors, and stops when the improvement is no longer statistically significant.

Backward selection (or backward elimination), which starts with all predictors in the model (full model), iteratively removes the least contributive predictors, and stops when you have a model where all predictors are statistically significant.

Stepwise selection (or sequential replacement), which is a combination of forward and backward selections. You start with no predictors, then sequentially add the most contributive predictors (like forward selection). After adding each new variable, remove any variables that no longer provide an improvement in the model fit (like backwardselection). After stepwise selection, we will apply Mellow Cp Statistic, AIC and adjusted R2 criterion for variable selection.

Mellow Cp statistic: In Mellow Cp statistic, we check the value of $E(p)$ for different p (no. of regressor). A common method is to perform all possible regressions, then use Mallow's Cp to compare the results. The smaller Cp value is better as it indicate smaller amounts of unexplained error. We select the model of those p regressor for which $E(p) \cong p$.

AIC: The Akaike Information Criterion, or AIC for short, is a method for scoring and selecting a model. To use AIC for model selection, we simply choose the model giving smallest AIC over the set of models considered.

Output in Forward Selection:

```

> fit <- lm(y ~x1 + x2 + x4 + x5 + x6 + x7 + x8 + x10 +
+          x11 + x12 + x13 + x14 + x15 + x17,data=mlr_train)
> step_model_forward = stepAIC(fit ,direction="forward",trace=1)
Start:  AIC=5708.14
y ~ x1 + x2 + x4 + x5 + x6 + x7 + x8 + x10 + x11 + x12 + x13 +
    x14 + x15 + x17

> step_model_forward

Call:
lm(formula = y ~ x1 + x2 + x4 + x5 + x6 + x7 + x8 + x10 + x11 +
    x12 + x13 + x14 + x15 + x17, data = mlr_train)

Coefficients:
(Intercept)  x1Developing      x2          x4          x5
  57.85164    -1.51952    -0.93520    1.06096    0.19285
          x6          x7          x8          x10          x11
 -0.02288   -0.07476    0.02849    0.66452    0.13929
          x12          x13          x14          x15          x17
  1.00246   -3.43951    0.35661   -0.01035   -0.73650

> best_subset_forward <-ols_step_best_subset(step_model_forward)
> best_subset_forward

Best Subsets Regression
-----
Model Index    Predictors
-----
1             x13
2             x5 x13
3             x4 x5 x13
4             x4 x5 x13 x17
5             x2 x4 x5 x13 x17
6             x2 x4 x5 x12 x13 x17
7             x2 x4 x5 x7 x12 x13 x17
8             x2 x4 x5 x7 x12 x13 x14 x17
9             x1 x2 x4 x5 x7 x12 x13 x14 x17
10            x1 x2 x4 x5 x7 x8 x12 x13 x14 x17
11            x1 x2 x4 x5 x7 x8 x10 x12 x13 x14 x17
12            x1 x2 x4 x5 x7 x8 x10 x11 x12 x13 x14 x17
13            x1 x2 x4 x5 x7 x8 x10 x11 x12 x13 x14 x15 x17
14            x1 x2 x4 x5 x6 x7 x8 x10 x11 x12 x13 x14 x15 x17
-----
Subsets Regr

-----
Model    R-Square    Adj.    Pred    C(p)    AIC
R-Square R-Square R-Square
-----
1         0.6948     0.6947   0.6943   1978.1067 13300.3883
2         0.7628     0.7626   0.7623   1053.4563 12751.8446
3         0.7930     0.7927   0.7923   644.3261 12456.6826
4         0.8063     0.8059   0.8054   465.3964 12313.8917
5         0.8170     0.8166   0.8159   321.5698 12191.7727
6         0.8248     0.8243   0.8236   217.5346 12098.8736
7         0.8296     0.8291   0.8283   153.6494 12039.7598
8         0.8333     0.8327   0.8318   106.0126 11994.5607
9         0.8361     0.8354   0.8346   69.6374 11959.3575
10        0.8385     0.8378   0.8369   38.2563 11928.4721
11        0.8398     0.8390   0.838   22.7116 11912.9872
12        0.8408     0.8399   0.8389   11.1269 11901.3520
13        0.8408     0.8399   0.8388   13.0223 11903.2467
14        0.8408     0.8398   0.8386   15.0000 11905.2242
-----

AIC: Akaike Information Criteria
SBIC: Sawa's Bayesian Information Criteria
SBC: Schwarz Bayesian Criteria
MSEP: Estimated error of prediction, assuming multivariate normality
FPE: Final Prediction Error
HSP: Hocking's Sp
APC: Amemiva Prediction Criteria

```

Here, the value of $E(C_p) = 13$ corresponding the $p(13)$, So, we select 13th subset model. Also, the value of adj. R2 correspond to 14th subset model is 0.8399 which is highest among all the subset models& AIC is minimum.

Output in Backward Selection:

```
> step_model_backward = stepAIC(fit ,direction="backward",trace=1)
Start: AIC=5708.14
```

```
y ~ x1 + x2 + x4 + x5 + x6 + x7 + x8 + x10 + x11 + x12 + x13 +
x14 + x15 + x17
```

	Df	Sum of Sq	RSS	AIC
- x6	1	0	29423	5706.2
- x15	1	1	29424	5706.2
<none>			29423	5708.1
- x11	1	183	29606	5719.7
- x10	1	226	29649	5722.9
- x8	1	400	29822	5735.6
- x1	1	437	29860	5738.3
- x12	1	449	29872	5739.2
- x14	1	465	29888	5740.4
- x7	1	563	29986	5747.5
- x17	1	652	30075	5754.0
- x4	1	688	30111	5756.6
- x5	1	1176	30599	5791.7
- x2	1	1605	31027	5822.1
- x13	1	39445	68868	7562.6

```
Step: AIC=5706.16
```

```
y ~ x1 + x2 + x4 + x5 + x7 + x8 + x10 + x11 + x12 + x13 + x14 +
x15 + x17
```

	Df	Sum of Sq	RSS	AIC
- x15	1	1	29425	5704.3
<none>			29423	5706.2
- x11	1	183	29606	5717.7
- x10	1	226	29649	5720.9
- x8	1	399	29823	5733.6
- x1	1	437	29860	5736.3
- x14	1	465	29888	5738.4
- x12	1	524	29947	5742.7
- x7	1	564	29987	5745.6
- x17	1	659	30083	5752.5
- x4	1	688	30111	5754.6
- x5	1	1180	30603	5790.0
- x2	1	1606	31029	5820.2
- x13	1	39487	68910	7561.9

```
Step: AIC=5704.27
```

```
y ~ x1 + x2 + x4 + x5 + x7 + x8 + x10 + x11 + x12 + x13 + x14 +
x17
```

```
Step: AIC=5704.27
```

```
y ~ x1 + x2 + x4 + x5 + x7 + x8 + x10 + x11 + x12 + x13 + x14 +
x17
```

	Df	Sum of Sq	RSS	AIC
<none>			29425	5704.3
- x11	1	184	29609	5715.9
- x10	1	226	29651	5719.0
- x8	1	398	29823	5731.6
- x1	1	436	29860	5734.4
- x14	1	464	29888	5736.4
- x12	1	523	29947	5740.7
- x7	1	598	30023	5746.2
- x17	1	660	30085	5750.7
- x4	1	692	30116	5753.0
- x5	1	1190	30614	5788.8
- x2	1	1606	31031	5818.3
- x13	1	39693	69117	7566.5

```
> step_model_backward
```

```
Call:
```

```
lm(formula = y ~ x1 + x2 + x4 + x5 + x7 + x8 + x10 + x11 + x12 +
x13 + x14 + x17, data = mlr_train)
```

```
Coefficients:
```

(Intercept)	x1Developing	x2	x4	x5
57.68485	-1.51624	-0.93552	1.06293	0.19341
x7	x8	x10	x11	x12
-0.07536	0.02837	0.66265	0.13963	0.99013
x13	x14	x17		
-3.44054	0.35469	-0.73823		

```
> best_subset_backward <-ols_step_best_subset(step_model_backward)
```

```
> best_subset_backward
```

```
Best Subsets Regression
```

Model Index	Predictors
1	x13
2	x5 x13
3	x4 x5 x13
4	x4 x5 x13 x17
5	x2 x4 x5 x13 x17
6	x2 x4 x5 x12 x13 x17
7	x2 x4 x5 x7 x12 x13 x17
8	x2 x4 x5 x7 x12 x13 x14 x17
9	x1 x2 x4 x5 x7 x12 x13 x14 x17
10	x1 x2 x4 x5 x7 x8 x12 x13 x14 x17
11	x1 x2 x4 x5 x7 x8 x10 x12 x13 x14 x17
12	x1 x2 x4 x5 x7 x8 x10 x11 x12 x13 x14 x17

```
Subsets Regre
```

Model	R-Square	Adj. R-Square	Pred R-Square	C(p)	AIC
1	0.6948	0.6947	0.6943	1981.6981	13300.3883
2	0.7628	0.7626	0.7623	1056.2471	12751.8446
3	0.7930	0.7927	0.7923	646.7618	12456.6826
4	0.8063	0.8059	0.8054	467.6757	12313.8917
5	0.8170	0.8166	0.8159	323.7232	12191.7727
6	0.8248	0.8243	0.8236	219.5964	12098.8736
7	0.8296	0.8291	0.8283	155.6542	12039.7598
8	0.8333	0.8327	0.8318	107.9746	11994.5607
9	0.8361	0.8354	0.8346	71.5662	11959.3575
10	0.8385	0.8378	0.8369	40.1563	11928.4721
11	0.8398	0.8390	0.838	24.5964	11912.9872
12	0.8408	0.8399	0.8389	13.0000	11901.3520

```
AIC: Akaike Information Criteria
```

```
SBIC: Sawa's Bayesian Information Criteria
```

```
SBC: Schwarz Bayesian Criteria
```

```
MSEP: Estimated error of prediction, assuming multivariate normality
```

```
FPE: Final Prediction Error
```

```
HSP: Hocking's Sp
```

```
APC: Amemiya Prediction Criteria
```

Here, the value of $E(C_p) = 13$ corresponding the $p(12)$, So, we select 12th subset model. Also, the value of adj. R2 correspond to 14th subset model is 0.8399 which is highest among all the subset models & AIC is minimum

Output in Stepwise Selection:

```
> step_model_both = stepAIC(fit ,direction="both",trace=1)
Start: AIC=5708.14
y ~ x1 + x2 + x4 + x5 + x6 + x7 + x8 + x10 + x11 + x12 + x13 +
x14 + x15 + x17
```

	Df	Sum of Sq	RSS	AIC
- x6	1	0	29423	5706.2
- x15	1	1	29424	5706.2
<none>			29423	5708.1
- x11	1	183	29606	5719.7
- x10	1	226	29649	5722.9
- x8	1	400	29822	5735.6
- x1	1	437	29860	5738.3
- x12	1	449	29872	5739.2
- x14	1	465	29888	5740.4
- x7	1	563	29986	5747.5
- x17	1	652	30075	5754.0
- x4	1	688	30111	5756.6
- x5	1	1176	30599	5791.7
- x2	1	1605	31027	5822.1
- x13	1	39445	68868	7562.6

Step: AIC=5706.16

```
y ~ x1 + x2 + x4 + x5 + x7 + x8 + x10 + x11 + x12 + x13 + x14 +
x15 + x17
```

	Df	Sum of Sq	RSS	AIC
- x15	1	1	29425	5704.3
<none>			29423	5706.2
+ x6	1	0	29423	5708.1
- x11	1	183	29606	5717.7
- x10	1	226	29649	5720.9
- x8	1	399	29823	5733.6
- x1	1	437	29860	5736.3
- x14	1	465	29888	5738.4
- x12	1	524	29947	5742.7
- x7	1	564	29987	5745.6
- x17	1	659	30083	5752.5
- x4	1	688	30111	5754.6
- x5	1	1180	30603	5790.0
- x2	1	1606	31029	5820.2
- x13	1	39487	68910	7561.9

Step: AIC=5704.27

```
y ~ x1 + x2 + x4 + x5 + x7 + x8 + x10 + x11 + x12 + x13 + x14 +
x17
```

	Df	Sum of Sq	RSS	AIC
<none>			29425	5704.3
+ x15	1	1	29423	5706.2
+ x6	1	0	29424	5706.2
- x11	1	184	29609	5715.9
- x10	1	226	29651	5719.0
- x8	1	398	29823	5731.6
- x1	1	436	29860	5734.4
- x14	1	464	29888	5736.4
- x12	1	523	29947	5740.7
- x7	1	598	30023	5746.2
- x17	1	660	30085	5750.7
- x4	1	692	30116	5753.0
- x5	1	1190	30614	5788.8
- x2	1	1606	31031	5818.3
- x13	1	39693	69117	7566.5

```
> step_model_both
```

```
Call:
lm(formula = y ~ x1 + x2 + x4 + x5 + x7 + x8 + x10 + x11 + x12 +
x13 + x14 + x17, data = mlr_train)
```

```
Coefficients:
(Intercept) x1Developing x2 x4 x5
57.68485 -1.51624 -0.93552 1.06293 0.19341
x7 x8 x10 x11 x12
-0.07536 0.02837 0.66265 0.13963 0.99013
x13 x14 x17
-3.44054 0.35469 -0.73823
```

```
> best_subset_both <-ols_step_best_subset(step_model_both)
```

```
> best_subset_both
```

Best Subsets Regression

Model Index	Predictors
1	x13
2	x5 x13
3	x4 x5 x13
4	x4 x5 x13 x17
5	x2 x4 x5 x13 x17
6	x2 x4 x5 x12 x13 x17
7	x2 x4 x5 x7 x12 x13 x17
8	x2 x4 x5 x7 x12 x13 x14 x17
9	x1 x2 x4 x5 x7 x12 x13 x14 x17
10	x1 x2 x4 x5 x7 x8 x12 x13 x14 x17
11	x1 x2 x4 x5 x7 x8 x10 x12 x13 x14 x17
12	x1 x2 x4 x5 x7 x8 x10 x11 x12 x13 x14 x17

Subsets Regre

Model	R-Square	Adj. R-Square	Pred R-Square	C(p)	AIC
1	0.6948	0.6947	0.6943	1981.6981	13300.3883
2	0.7628	0.7626	0.7623	1056.2471	12751.8446
3	0.7930	0.7927	0.7923	646.7618	12456.6826
4	0.8063	0.8059	0.8054	467.6757	12313.8917
5	0.8170	0.8166	0.8159	323.7232	12191.7727
6	0.8248	0.8243	0.8236	219.5964	12098.8736
7	0.8296	0.8291	0.8283	155.6542	12039.7598
8	0.8333	0.8327	0.8318	107.9746	11994.5607
9	0.8361	0.8354	0.8346	71.5662	11959.3575
10	0.8385	0.8378	0.8369	40.1563	11928.4721
11	0.8398	0.8390	0.838	24.5964	11912.9872
12	0.8408	0.8399	0.8389	13.0000	11901.3520

AIC: Akaike Information Criteria

SBIC: Sawa's Bayesian Information Criteria

SBC: Schwarz Bayesian Criteria

MSEP: Estimated error of prediction, assuming multivariate normality

FPE: Final Prediction Error

HSP: Hocking's Sp

APC: Amemiya Prediction Criteria

Here, the value of E(Cp) =13 corresponding the p(12),So, we select 12th subset model.

Also, the value of adj. R2 correspond to 14th subset model is 0.8399 which is highest among all the subset models & AIC is minimum.

Final Model:

After, variable selection we lead to the final model which is:

$$y = (\beta_1 x_1 + \beta_2 x_2 + \beta_4 x_4 + \beta_5 x_5 + \beta_7 x_7 + \beta_8 x_8 + \beta_{10} x_{10} + \beta_{11} x_{11} + \beta_{12} x_{12} + \beta_{13} x_{13} + \beta_{14} x_{14} + \beta_{17} x_{17})$$

```
> fit1 <- lm(y ~ x1 + x2 + x4 + x5 + x7 + x8 + x10 +
+           x11 + x12 + x13 + x14 + x17, data=mlr_train)
> summary(fit1)

Call:
lm(formula = y ~ x1 + x2 + x4 + x5 + x7 + x8 + x10 + x11 + x12 +
    x13 + x14 + x17, data = mlr_train)

Residuals:
    Min       1Q   Median       3Q      Max
-12.986  -2.226  -0.027   2.326  11.484

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  57.684851   1.063049   54.264 < 2e-16 ***
x1Developing -1.516242   0.267458  -5.669 1.63e-08 ***
x2           -0.935517   0.085957 -10.884 < 2e-16 ***
x4            1.062930   0.148813   7.143 1.24e-12 ***
x5            0.193408   0.020648   9.367 < 2e-16 ***
x7           -0.075356   0.011346  -6.642 3.91e-11 ***
x8            0.028366   0.005236   5.418 6.70e-08 ***
x10           0.662648   0.162202   4.085 4.56e-05 ***
x11           0.139625   0.037866   3.687 0.000232 ***
x12           0.990132   0.159480   6.209 6.39e-10 ***
x13          -3.440537   0.063591 -54.104 < 2e-16 ***
x14           0.354689   0.060643   5.849 5.70e-09 ***
x17          -0.738233   0.105801  -6.978 3.97e-12 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.682 on 2170 degrees of freedom
Multiple R-squared:  0.8408,    Adjusted R-squared:  0.8399
F-statistic: 955.2 on 12 and 2170 DF, p-value: < 2.2e-16
```

Test of Normality:**Shapiro Wilk Test**

The Shapiro – Wilk test is a test of normality of a random sample.

H0: Sample comes from normally distributed population

HA: Sample does not come from normal population

Test Statistic:

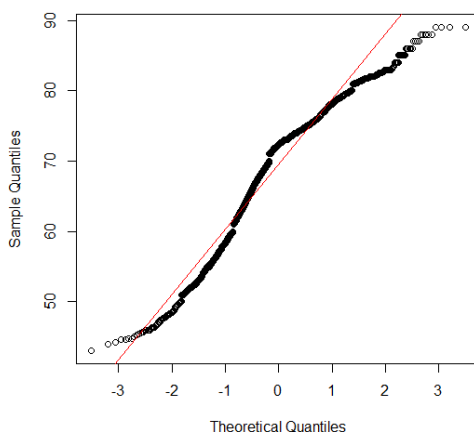
$$W = \frac{\left(\sum_{i=1}^n a_i x_{(i)}\right)^2}{\sum_{i=1}^n (x_i - \bar{x})^2}$$

Where, x_i 's are the ordered random sample values. a_i 's are the constant generated from the covariance, variances and means of the sample (size n) from a normally distributed sample.

Decision Rule:

We reject H_0 if $W < W_\alpha$, at α % level of significance otherwise accepts H_0

Normal Q-Q Plot



```
> qqnorm(mlr_train$y)
> qqline(mlr_train$y, col="red")
> shapiro.test(mlr_train$y)
```

Shapiro-Wilk normality test

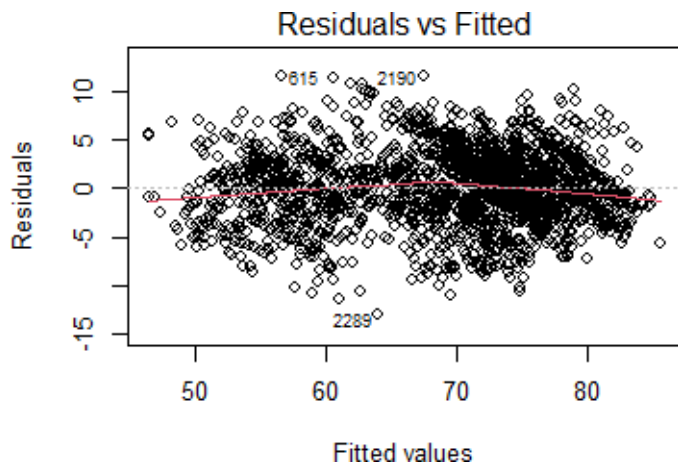
```
data:  mlr_train$y
W = 0.94986, p-value < 2.2e-16
```

Since, $p\text{-value} < 0.05$ thus, we reject the null hypothesis at 5% level of significance. Hence, data is non-normal.

Homoscedasticity:

H0: the residual is homoscedastic

H1: the residual is heteroscedastic

Residual Plot:

```
> #Breusch-Pagan test
> bptest(fit1)
```

```
studentized Breusch-Pagan test
```

```
data: fit1
BP = 59.712, df = 12, p-value = 2.548e-08
```

Since, p-value < 0.01 thus, we reject the null hypothesis at 1% level of significance. Hence, Training data is **Heteroskedastic**.

Auto Correlation:

H0: there is no serial correlation.

H1: there is serial correlation.

```
> dwt(fit1)
lag Autocorrelation D-W Statistic p-value
1 -0.01266898 2.021858 0.552
Alternative hypothesis: rho != 0
```

Since, p-value > 0.01 thus, we fail to reject the null hypothesis at 1% level of significance. Hence, there is no serial autocorrelation.

Model Performance:

It is important to appropriately estimate the prediction error of a model, since (a) it provides insight into its accuracy; (b) it allows comparison of different models, and (c) it is used to define warning thresholds. In order to facilitate the analysis of final calculation results, different performance evaluation functions are adopted in this paper, that is, mean square error (MSE), maximum absolute percentage error (MAPE).

The mean absolute percentage error (MAPE) is a statistical measure of how accurate a forecast system is. It measures this accuracy as a percentage, and can be calculated as the average absolute percent error for each time period minus actual values divided by actual values. Where A_t is the actual value and F_t is the forecast value, this is given by:

$$M = \frac{1}{n} \sum_{t=1}^n \left| \frac{A_t - F_t}{A_t} \right|$$

The mean squared error (MSE) tells you how close a regression line is to a set of points. It does this by taking the distances from the points to the regression line (these distances are the “errors”) and squaring them. The squaring is necessary to remove any negative signs. It also gives more weight to larger differences. It’s called the mean squared error as you’re finding the average of a set of errors.

```
> MSE_train
[1] 13.47896
> MSE_test
[1] 14.57727
> train_Mape
[1] 4.304494
> test_Mape
[1] 4.458779
> accuracy_train
[1] 99.68379
> accuracy_test
[1] 99.94273
```

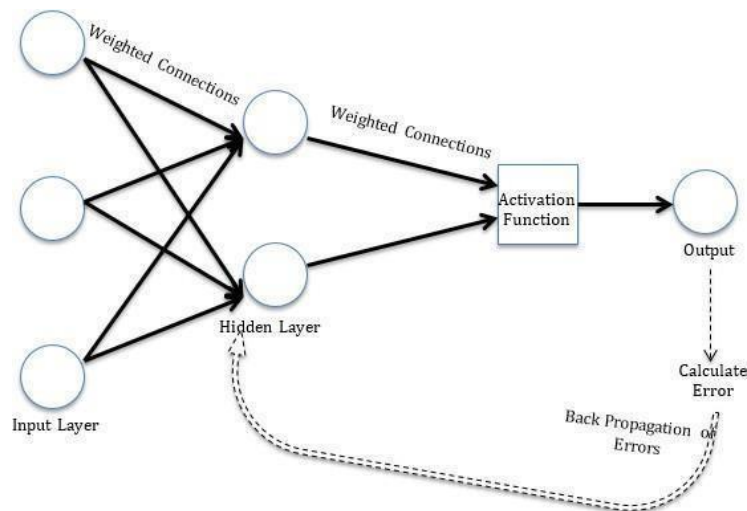
Multiple Linear regression Equation:

$$\begin{aligned} (\text{Life_expectancy}) = & 57.684851 + (-1.516242)*(\text{Status}) + (-0.935517)*(\text{Adult_Mortality}) + (1.062930)*(\text{Alcohol}) + \\ & (0.193408)*(\text{percentage_expenditure}) + (-0.075356)*(\text{Measles}) + (0.028366)*(\text{BMI}) + (0.662648)*(\text{Polio}) + \\ & (0.139625)*(\text{Total_expenditure}) + (0.990132)*(\text{Diphtheria}) + (-3.440537)*(\text{HIV_AIDS}) + (0.354689)*(\text{GDP}) + \\ & (-0.738233)*(\text{thinness_5_9_years}) \end{aligned}$$

Even though MLR performs extremely well in this data for model Building and Prediction, but it violates homoscedasticity and normality assumptions. This encourages us to go for another algorithm that does not take into consideration normality.

B. ARTIFICIAL NEURAL NETWORK

It has the ability to learn by examples. ANN is an information processing model inspired by the biological neuron system. It is composed of a large number of highly interconnected processing elements known as the neuron to solve problems. It follows the non-linear path and process information in parallel throughout the nodes. A neural network is a complex adaptive system. Adaptive means it has the ability to change its internal structure by adjusting weights of inputs.



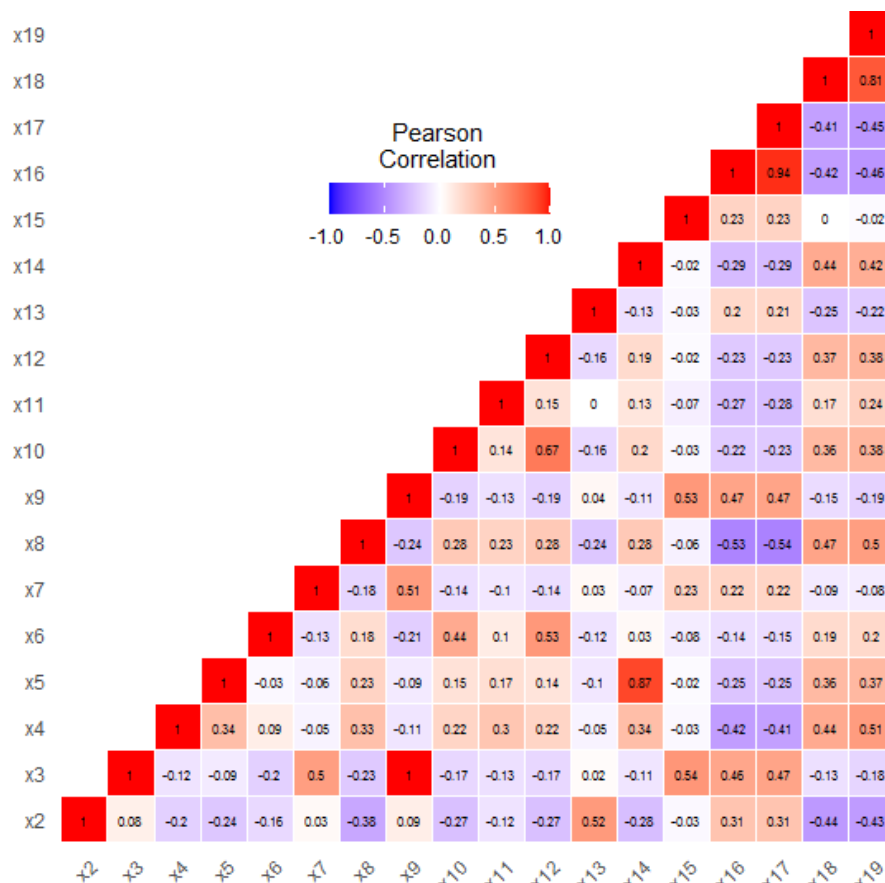
A Neural Network Consists of:

- Input layers: Layers that take inputs based on existing data. Here we have 7 input variables.
- Hidden layers: Layers that use back propagation to optimize the weights of the input variables in order to improve the predictive power of the model.
- Output layers: Output of predictions based on the data from the input and hidden layers.

Before fitting a neural network, some preparation needs to be done. Neural networks are not that easy to train and tune. As a *first step*, we are going to address data preprocessing. In this case also we removed unnecessary columns, renamed them, and imputed missing values by interpolation.

Data Normalization:

One of the most important procedures when forming a neural network is data normalization. This involves adjusting the data to a common scale so as to accurately compare predicted and actual values. Failure to normalize the data will typically result in the prediction value remaining the same across all observations, regardless of the input values. Most of the times the algorithm will not converge before the number of maximum iterations allowed. You can choose different methods to scale the data (z-normalization, min-max scale, etc...). I chose to use the min-max method and scale the data in the interval [0, 1].

Heatmap & Correlation Matrix:

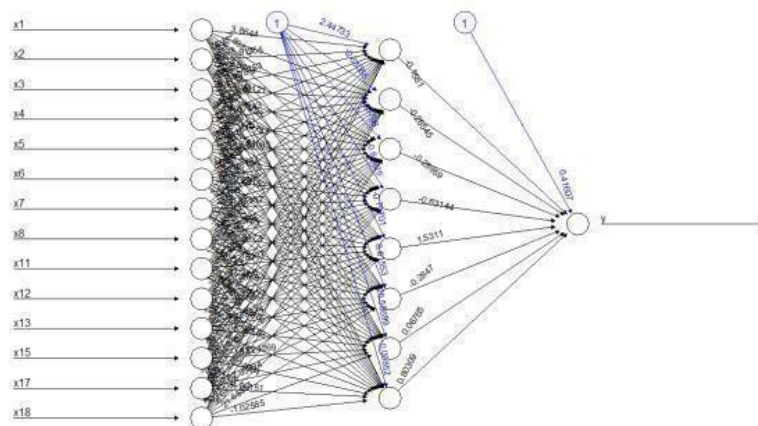
We see there is high correlation between x18-x19, x16-x17, x5-x14, x10-x12 & x3-x9. So we have removed x19, x16, x10, x14 & x9 from the data frame to avoid over fitting.

After this we have divided the whole dataset in training and test set randomly in the ratio 80:20. Then we have deployed MLR on the training dataset.

Parameter Setting:

There is no fixed rule as to how many layers and neurons to use although there are several more or less accepted rules of thumb. Usually, if at all necessary, one hidden layer is enough for a vast number of applications. As far as the number of neurons is concerned, it should be between the input layer size and the output layer size, usually 2/3 of the input size. Here we have 1 response and 14 regressors so we used 8 neurons in the only hidden layer. The configuration of our neural network is (14:8:1)

This is the graphical representation of the model with the weights on each connection:



Predicting Using The Neural Network:

Now we can try to predict the values for the both train & test set and calculate the MSE, MAPE & Accuracy. Remember that the net will output a normalized prediction, so we need to scale it back in order to make a meaningful comparison (or just a simple prediction).

```
> MSE.nn1_train
[1] 4.684468
> MSE.nn1_test
[1] 7.257011
> train_Mape
[1] 2.323643
> test_Mape
[1] 2.882731
> accuracy_train
[1] 99.88873
> accuracy_test
[1] 99.78785
```

VIII. PERFORMACE OF THE ALGORITHMS

To compare the performance of these two algorithms we will use MSE, MAPE & accuracy of the both models on both training data and test data.

Statistical performance of the MLR and ANN:

	MLR		ANN	
	Training	Testing	Training	Testing
MSE	13.47896	14.57727	4.684468	7.257011
MAPE	4.304494	4.458779	2.323643	2.882731
Accuracy	99.68379	99.94273	99.88873	99.78785

IX. CONCLUSION

In both the models we can see that all the Regressors are not contributing in predicting Life Expectancy. We had to drop few Regressors due to multicollinearity.

MLR model indicates that Alcohol, BMI, Polio, Total expenditure, Diphtheria, GDP and percentage expenditure have positive impact on Life Expectancy. On the other hand, Measles, HIV_AIDS and thinness_5_9_years have negative impact on Life Expectancy. It is evident that immunization coverage improves life expectancy.

Both the models have performed exceptionally in predicting Life Expectancy. But neural networks usually outperform linear regression as they deal with non-linearity automatically with large number of inputs, whereas in linear regression you need to mention explicitly. Here also MAPE and MSE suggest that neural network has performed better than MLR in Prediction Life Expectancy.

FUTURE WORK:

- There are many scopes in future with this data. We can run cross validation models to check if we can improve results.
- We can include the year column to do time series analysis, but for that we need more data.
- We can deploy other algorithms such as k means or decision tree according to our requirements.
- We can divide the whole dataset into two parts. One for developed country, another for developing country, and then we can again run MLR on that.

REFERENCES:

- <https://www.quora.com/Why-is-life-expectancy-an-important-indicator-of-development>
- <https://www.investopedia.com/terms/t/t-test.asp>
- <https://www.statisticshowto.com/forward-selection/>
- <https://www.statisticshowto.com/mean-squared-error/>
- <https://towardsdatascience.com/transforming-skewed-data-73da4c2d0d16>
- <https://www.kaggle.com/kumarajarshi/life-expectancy-who/kernels>
- <https://machinelearningmastery.com/probabilistic-model-selection-measures/>
- <https://www.datacamp.com/community/tutorials/neural-network-models-r>
- <https://www.analyticsvidhya.com/blog/2017/09/creating-visualizing-neural-network-in-r/>
- <https://www.sciencedirect.com/topics/computer-science/backward-elimination>
- <https://datascienceplus.com/fitting-neural-network-in-r/>
- <https://datascienceplus.com/neuralnet-train-and-test-neural-networks-using-r/>
- <https://stats.stackexchange.com/questions/364942/checking-for-multicollinearity-in-selection-of-f-variables-for-regression-model>
- <https://analyticsindiamag.com/a-beginners-guide-to-build-and-visualize-ann-in-r-with-code/>
- <https://www.hindawi.com/journals/mpe/2019/7620948/>
- <https://medium.com/@TheDataGyan/day-8-data-transformation-skewness-normalization-and-much-more-4c144d370e55>