OR-568: APPLIED PREDICTIVE ANALYTICS

FINAL PROJECT REPORT

Instructor: Dr. Ran Ji

LIFE EXPECTANCY PREDICTION

Team

Sai Chaitanya Sadasivuni (G01241462)

Keerthi Gollamudi (G01241047)

Saathvika Kommisetty (G01236616)

Sai Atchuth Reddy Syamala (G01240054)

Deepthi Tamma (G01241465)

Table of Contents

1. Abstract 3

2. Introduction 3

3. Data 3

3.1 Dataset Description 3

3.2 Dataset source 4

3.3 Dataset Background 5

3.4 Data Preprocessing 6

4. Feature selection 7

4.1 Correlation Analysis 7

4.2 Principal Component Analysis 9

5. Predictive models 9

5.1 Multiple linear Regression Models 10

5.2 LASSO Regression 11

5.3 Ridge Regression 12

5.4 Decision Trees 13

5.5 Random Forest Regression 15

5.6 Gradient Boosting Model 17

6. Problem statements 20

6.1 Problem statement 1 20

6.2 Problem statement 2 20

6.3 Problem statement 3 21

6.4 Problem statement 4 21

6.5 Problem statement 5 21

6.6 Problem statement 6 22

7. Conclusion 22

8. How to run our code? 22

9. References 23

1. Abstract

In all nations, life expectancy is a big concern for people's health. Life Expectation estimation can be beneficial for the countries to learn what factors influence the rate of life expectancy so that they can concentrate on developing the rate of life expectancy in those areas. The rate of life expectancy depends on the various kinds of diseases and other factors that influence the death rate of individuals in different demographics.

2. Introduction

In this project, incorporating demographic variables, income composition, and death rates (mortality), we propose to investigate the factors influencing life expectancy. We planned to perform statistical modelling, variable selection process, and few regression models in this project to learn which is the factor that most influences the rate of life expectancy. Different forms of attempts were made to pick variables. We also developed strategies such as the method of variable selection to assess which variables from the dataset can give the better estimate for life expectancy rate. To assess the skewness of the date, we also used a kurtosis test so that we could obtain the best results from the model. Then, with the help of various regression models, we were able to predict the variables that influence the life expectancy rate. We made use of only a few variables which we felt that they would act as the potential predictors and make meaningful predictions.

3. Data

3.1 Dataset Description

The size of the Life Expectancy data set is a Comma Separated File (CSV) which is 333kb with 22 columns or attributes and 2939 records or entries. This dataset is corresponding to the Life Expectancy where the data of health factors of the 193 countries were collected from the WHO repository website and the key economic data is collected from the United Nations

Website. This dataset consists of various factors such as Immunization factors, Economic factors, Social factors, Mortality factors, and other factors that would affect the rate of Life Expectancy.

3.2 Dataset source

As already stated, we retrieved the data from the Kaggle website, and the link is provided below

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

Below screenshot is a snapshot of the data from which we will be sorting the response and predictor variables.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
1	Country	Year	Status	Life expe	Adult Mo	infant de	Alcohol	percenta	Hepatitis	Measles	BMI	under-fiv	Polio	Total exp	Diphtheri	HIV/AIDS	GDP	Populatio	thinness	thinness	Income c	Schooling
2	Afghanistan	2015	Developing	65	263	62	0.01	71.279624	65	1154	19.1	83	6	8.16	65	0.1	584.25921	33736494	17.2	17.3	0.479	10.1
3	Afghanistan	2014	Developing	59.9	271	64	0.01	73.523582	62	492	18.6	86	58	8.18	62	0.1	612.69651	327582	17.5	17.5	0.476	10
4	Afghanistan	2013	Developing	59.9	268	66	0.01	73.219243	64	430	18.1	89	62	8.13	64	0.1	631.74498	31731688	17.7	17.7	0.47	9.9
5	Afghanistan	2012	Developing	59.5	272	69	0.01	78.184215	67	2787	17.6	93	67	8.52	67	0.1	669.959	3696958	17.9	18	0.463	9.8
6	Afghanistan	2011	Developing	59.2	275	71	0.01	7.0971087	68	3013	17.2	97	68	7.87	68	0.1	63.537231	2978599	18.2	18.2	0.454	9.5
7	Afghanistan	2010	Developing	58.8	279	74	0.01	79.679367	66	1989	16.7	102	66	9.2	66	0.1	553.32894	2883167	18.4	18.4	0.448	9.2
8	Afghanistan	2009	Developing	58.6	281	77	0.01	56.762217	63	2861	16.2	106	63	9.42	63	0.1	445.8933	284331	18.6	18.7	0.434	8.9
9	Afghanistan	2008	Developing	58.1	287	80	0.03	25.873925	64	1599	15.7	110	64	8.33	64	0.1	373.36112	2729431	18.8	18.9	0.433	8.7
10	Afghanistan	2007	Developing	57.5	295	82	0.02	10.910156	63	1141	15.2	113	63	6.73	63	0.1	369.8358	26616792	19	19.1	0.415	8.4
11	Afghanistan	2006	Developing	57.3	295	84	0.03	17.171518	64	1990	14.7	116	58	7.43	58	0.1	272.56377	2589345	19.2	19.3	0.405	8.1
12	Afghanistan	2005	Developing	57.3	291	85	0.02	1.3886477	66	1296	14.2	118	58	8.7	58	0.1	25.29413	257798	19.3	19.5	0.396	7.9
13	Afghanistan	2004	Developing	57	293	87	0.02	15.296066	67	466	13.8	120	5	8.79	5	0.1	219.14135	24118979	19.5	19.7	0.381	6.8
14	Afghanistan	2003	Developing	56.7	295	87	0.01	11.089053	65	798	13.4	122	41	8.82	41	0.1	198.72854	2364851	19.7	19.9	0.373	6.5
15	Afghanistan	2002	Developing	56.2	3	88	0.01	16.887351	64	2486	13	122	36	7.76	36	0.1	187.84595	21979923	19.9	2.2	0.341	6.2
16	Afghanistan	2001	Developing	55.3	316	88	0.01	10.574728	63	8762	12.6	122	35	7.8	33	0.1	117.49698	2966463	2.1	2.4	0.34	5.9
17	Afghanistan	2000	Developing	54.8	321	88	0.01	10.42496	62	6532	12.2	122	24	8.2	24	0.1	114.56	293756	2.3	2.5	0.338	5.5
18	Albania	2015	Developing	77.8	74	0	4.6	364.97523	99	0	58	0	99	6	99	0.1	3954.2278	28873	1.2	1.3	0.762	14.2
19	Albania	2014	Developing	77.5	8	0	4.51	428.74907	98	0	57.2	1	98	5.88	98	0.1	4575.7638	288914	1.2	1.3	0.761	14.2
20	Albania	2013	Developing	77.2	84	0	4.76	430.87698	99	0	56.5	1	99	5.66	99	0.1	4414.7231	289592	1.3	1.4	0.759	14.2
21	Albania	2012	Developing	76.9	86	0	5.14	412.44336	99	9	55.8	1	99	5.59	99	0.1	4247.6144	2941	1.3	1.4	0.752	14.2
22	Albania	2011	Developing	76.6	88	0	5.37	437.0621	99	28	55.1	1	99	5.71	99	0.1	4437.1787	295195	1.4	1.5	0.738	13.3
23	Albania	2010	Developing	76.2	91	1	5.28	41.822757	99	10	54.3	1	99	5.34	99	0.1	494.35883	291321	1.4	1.5	0.725	12.5
24	Albania	2009	Developing	76.1	91	1	5.79	348.05595	98	0	53.5	1	98	5.79	98	0.1	4114.1365	2927519	1.5	1.6	0.721	12.2
25	Albania	2008	Developing	75.3	1	1	5.61	36.622068	99	0	52.6	1	99	5.87	99	0.1	437.53965	2947314	1.6	1.6	0.713	12

3.3 Dataset Background

Data Field	Data Type	Defining Data Field
Country	Categorical	Name of the countries
Year	Ordinal	Year in which the data is recorded (2000-2015)
Status	Categorical	Defines whether the country is developed or not (Developed, Developing)
Life Expectancy	Continuous	Value of life expectancy depending on age
Adult Mortality	Discrete	Number of Adult Deaths per 1000 population (Ages between 15 and 60)
infant deaths	Discrete	Number of Infant Deaths per 1000 population
Alcohol	Continuous	Consumption of Alcohol (in liters)

In this section, for all of the data present in the dataset, a detailed knowledge will be provided such as what are the variables present in the dataset, what does that variable in that dataset mean in real time and what is the data type of the variable in a data analyst/ scientist perspective.

***Source*:** This dataset is taken from Kaggle website and is owned by Google LLC.

***Privacy*:** This dataset is made available for public access for the purpose of health data analysis.

thinness 1-19 years	Continuous	Prevalence of thinness among children and adolescents for Age 10 to 19 (%)
thinness 5-9 years	Continuous	Prevalence of thinness among children for Age 5 to 9(%)
Income composition of resources	Continuous	Human Development Index in terms of income composition of resources (index ranging from 0 to 1)
Schooling	Continuous	Number of years of Schooling(years)

percentage expenditure	Continuous	Expenditure on Health as a percentage of GDP
Hepatitis B	Discrete	Immunization coverage of Hepatitis B as percentage among 1-year old
Measles	Discrete	Number of reported cases per 1000 population
BMI	Continuous	Average Body-Mass Index of entire population
under-five deaths	Discrete	Number of deaths under 5 years per 1000 population
Polio	Discrete	Immunization coverage of Polio as percentage among 1-year old
Total expenditure	Continuous	General Government Expenditure on Health as a percentage of total government expenditure (%)
Diphtheria	Discrete	Immunization coverage of Diphtheria as percentage among 1-year old
HIV/AIDS	Continuous	Deaths per 1000 live births (0-4 years)
GDP	Continuous	Gross Domestic Product per capita of the country in the specific year (in USD)
Population	Discrete	Population within the country in the specific year

3.4 Data Preprocessing

Data preprocessing is a very important step in the data mining process. The raw data, which is gathered may often be incomplete, noisy, and may contain missing values, outliers, and erroneous values and to be handled, the data must be preprocessed. The main goal of the data preprocessing is to minimize GIGO i.e., if the garbage input is minimized then the garbage output is also minimized. Out of 2938 records, it has been found that there are 2563 cells with missing data.

Firstly, missing values can pose problems in data analysis methods. As the number of missing values cells is high in number it's not a good practice to simply delete the missing records. Also deleting records creates a biased dataset. One of the efficient ways to handle the missing data we chose is by replacing the missing values with the mean of the rest values of their respective columns which is called as imputation. After data imputation, the updated file is generated and exported in a .csv format and is used for further processing in the feature selection procedures.

Off the record, the next step we performed in the data preprocessing is to detect the outliers as the presence of outliers may produce unstable results. We detected outliers by using graphical methods like Histogram, boxplot, and scatterplot and observed the distributions. Additionally, we also used skewness and kurtosis functions on each and every variable to check the quality of the variables.

4. Feature selection

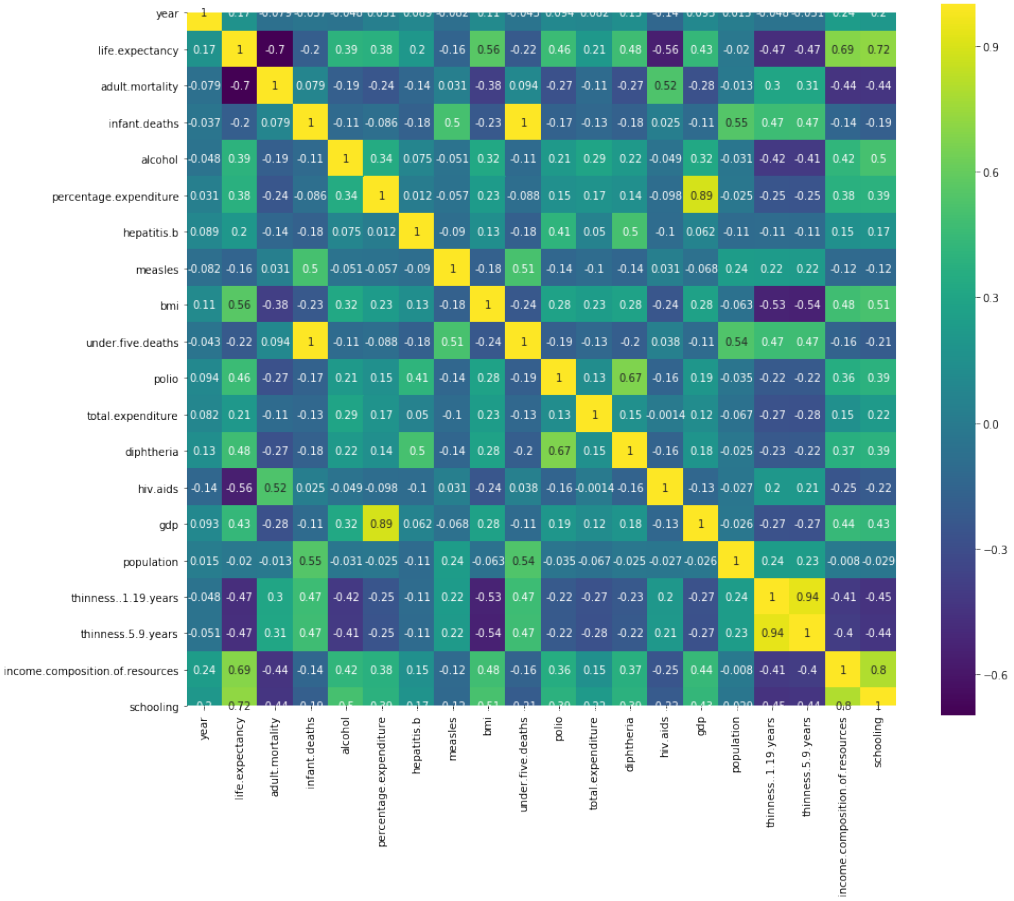
We had 21 variables excluding the target variable (life expectancy) and we can't use all the variables as few variables might be totally insignificant. To figure out that, we have performed the correlation analysis to check the level of correlation between the target variables and the remaining 21 predictors. We ignored the predictors that are not correlated and then performed the principal component analysis to find out the highly related predictor. The correlation analysis and PCA will be described in detail in the below paragraphs. (How can I avoid multicollinearity?, 2015)

4.1 Correlation Analysis

Correlation is nothing but a statistical measure which indicates to what extent two or more variables fluctuate together. In simple words how strongly or weakly the variables are related to the other variables in the dataset. From correlation plots we can observe three types of correlations, namely Positive correlation, Negative correlation and no correlation. In positive correlation, the target variable increases as the independent variable increases. In case of

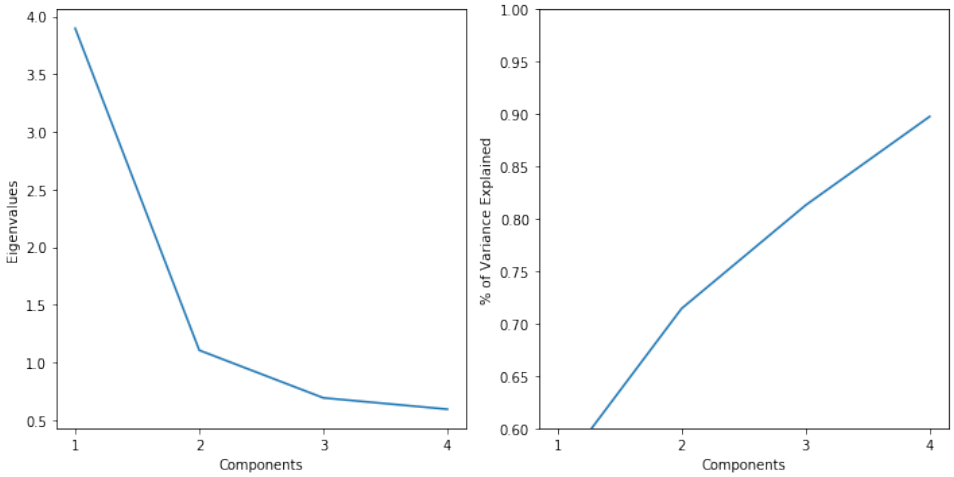
negative correlation, the target variable decreases as the independent variable increase. In No correlation, they do not pertain to any trend and scattered randomly.

Correlation analysis is performed in python environment using Jupyter notebook. We generated heat map to visualize the level of correlation. Usually, Multicollinearity is a problem in regression analysis that occurs when two independent variables are highly correlated, e.g., r = 0.90, or higher. The relationship between the independent variables and the dependent variables is distorted by the very strong relationship between the independent variables, leading to the likelihood that our interpretation of relationships will be incorrect. So, after checking the correlation plot between the variables, few variables such as deaths under age 5, infant deaths, thinness 1-19 years, thinness 5-9 years, GDP, Percentage expenditure have been dropped as they distort our analysis by having correlation greater than 0.9 which is to be considered as a problem of Multicollinearity. To overcome the problem of multicollinearity and to make feature selection PCA is used.



4.2 Principal Component Analysis

Principal Component Analysis (PCA) is used to explain the variance-covariance structure of a set of variables through linear combinations. It is often used as a dimensionality-reduction technique. The main purpose of PCA is to find out the smaller number of numerical variables that contains most of the data.



From the above obtained “Scree Plot” we can observe that the first two principal components have eigenvalues greater than 1. These two components explain 71.42% proportion of variation in the data. The scree plot shows that the eigenvalues start to form a straight line after the second principal component. Inspecting the figure, we can depict that 71.42% is an adequate amount of variation explained in the data by first two principal components and there is an elbow after the second component.

5. Predictive models

Considering the variables which are correlated to the life expectancy variable, we have performed 6 predictive models on the dataset to know what factors (variables) influence the life expectancy the most. We made use of the models’ multiple linear regression, lasso regression, ridge regression, decision tree, random forest, gradient boosting method and the outcomes such as mean squared error and R squared value which help us in determining the

accuracy of the prediction each model will be individually reported when we move forward in the report.

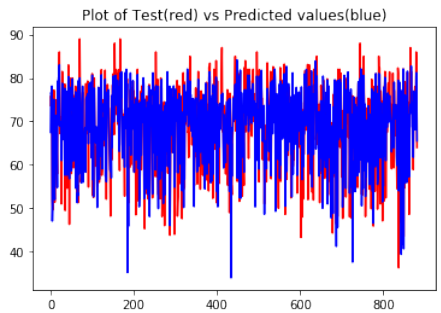
5.1 Multiple linear Regression Models

Multiple linear regression is a statistical method that uses several explanatory variables to predict the outcome of a response variable. The purpose of this method is to model the linear relationship between the independent variables and the target variable.

As we have many independent variables that may majorly affect the life expectancy rate in various countries, we have considered performing multiple linear regression. The dataset is divided into training and test sets in the ratio of 70% and 30% respectively. The model is trained using the training set and further used in predicting the values of the test set. The metrics which we have considered in determining the performance of the model are MSE and R2_score. (Kenton, 2020)

MSE=18.36% and R2_score=80.34% are obtained for this model.

```
In [94]: mse = mean_squared_error(y_test, y_head)
         mse

Out[94]: 18.363762743573748

In [95]: # r2 value:

In [96]: r2_degeri = r2_score(y_test, y_head)
         print("Test r2 score = ",r2_degeri)

         plt.plot(y_test_1,y_test,color="r")
         plt.plot(y_test_1,y_head,color="blue")
         plt.title("Plot of Test(red) vs Predicted values(blue)")
         plt.show()

         Test r2 score =  0.8034934655780901

In [97]: r2_degeri*100

Out[97]: 80.34934655780901
```



5.2 LASSO Regression

LASSO stands for *Least Absolute Shrinkage and Selection Operator*. It achieves L1 regularization, i.e., it adds a factor of sum of absolute value of coefficients in the optimization objective. Thus, below is the optimization it does:

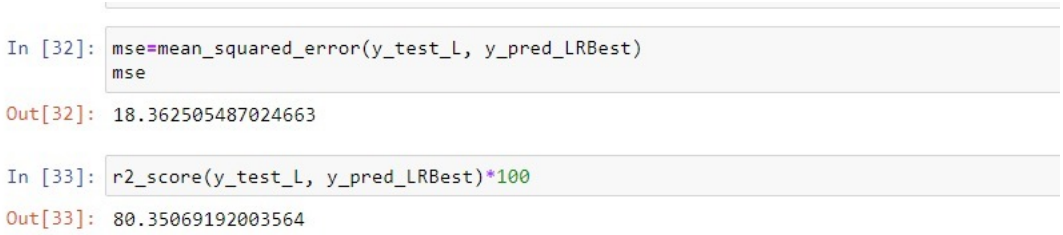
Objective = RSS + α * (sum of absolute value of coefficients)

From the above, RSS is the 'Residual Sum of Squares', the sum of the square of errors between the predicted and actual values in the training data set. Also, α (alpha) provides a trade-off between balancing RSS and magnitude of coefficients. α can take various values as follows:

1. α = 0: Same coefficients as simple linear regression
2. α = ∞: All coefficients zero (same logic as before)
3. 0 < α < ∞: coefficients between 0 and that of simple linear regression

The difference between ridge and lasso regression is that it tends to make coefficients to absolute zero as compared to Ridge which never sets the value of coefficient to absolute zero. The model can settle a significant number of the difficulties that we face with linear regression and can be a helpful device for fitting linear models. It's a superior way to dissect information and apprehend relationships in the information and try not to over-fit.

When we performed the modeling on data, with alpha=1, MSE =19.24, R2_score =79.41 and to get the best alpha value, tuned the parameter using the gridsearchCV class with a grid of values we have defined. Alpha=0.02 (best value), MSE=18.36, R2_score =80.35. (Predicting housing prices using advanced regression techniques) https://www.academia.edu/40754319/Predictive_Modelling_of_the_Housing_Prices_in_Melbourne



5.3 Ridge Regression

Ridge regression performs '**L2 regularization** ', i.e., it adds a factor of sum of squares of coefficients in the optimization objective. Thus,

Objective = RSS + α * (sum of square of coefficients)

From the above, RSS is the 'Residual Sum of Squares', the sum of the square of errors between the predicted and actual values in the training data set. Also, α (alpha) is the parameter which balances the amount of emphasis given to minimizing RSS vs minimizing sum of square of coefficients. α can take various values:

1. **α = 0:**
    - The objective becomes same as simple linear regression.
    - We'll get the same coefficients as simple linear regression.
2. **α = ∞:**
    - The coefficients will be zero. Because of infinite weightage on square of coefficients, anything less than zero will make the objective infinite.
3. **0 < α < ∞:**
    - The magnitude of α will decide the weightage given to different parts of objective.
    - The coefficients will be somewhere between 0 and ones for simple linear regression.

Ridge regression decreases the complexity of a model but does not reduce the number of variables since it never primes to a coefficient been zero relatively only minimizes it.

When we performed the modeling on data, with alpha=0.01, MSE =18.52, R2_score =78.14 and to get the best alpha value, tuned the parameter using the gridsearchCV class with a grid of values we have defined. Alpha=0.06 (best value), MSE=18.63, R2_score =78.00. (Bhattacharyya, 2018), (JAIN, 2016)

```
In [43]: mse1=mean_squared_error(y_test_R, y_pred_R_best)
         mse1

Out[43]: 18.632807618039568

In [44]: r2_score(y_test_R, y_pred_R_best)*100

Out[44]: 78.00940652517022
```

5.4 Decision Trees

Decision Tree model is the base for all the tree-based models. It can be used to visually represent decisions and decision making**.** This algorithm can be used for both regression and classification problems. It follows a set of if-else conditions for classifying and visualization data in a tree-based structure. In this dataset the response variable i.e., life expectancy is a continuous variable, so we have used regression tree analysis to represent and classify the information. We have divided the data into training (80%) and test (20%) sets. Also trained the model using training dataset. Once the model is trained, we have used the model to predict the test data.

Hyperparameter are nothing but the parameters which are used to control the learning process. Setting the optimal values for the hyper parameters will helps to improve the overall performance of the model**.** GridSearchCV is the process which tries all the combinations of the parameter values passed and evaluates the model for each combination using the Cross-Validation method. So, we have used GridSearchCV for hyper parameter tuning using the 5-fold cv. (Mujtaba, 2020)

Some of the hyper parameters we considered for regression tree are as follows:

min_samples_split: The minimum no. of sample required for a split.

max_depth: The length of the longest path from the tree root to a leaf.

max_leaf_nodes: The maximum number of leaf nodes to be considered.

The optimal values we got after performing hyper parameter tuning are as follows

min_samples_split = 10

max_depth = 10

max_leaf_nodes = 130

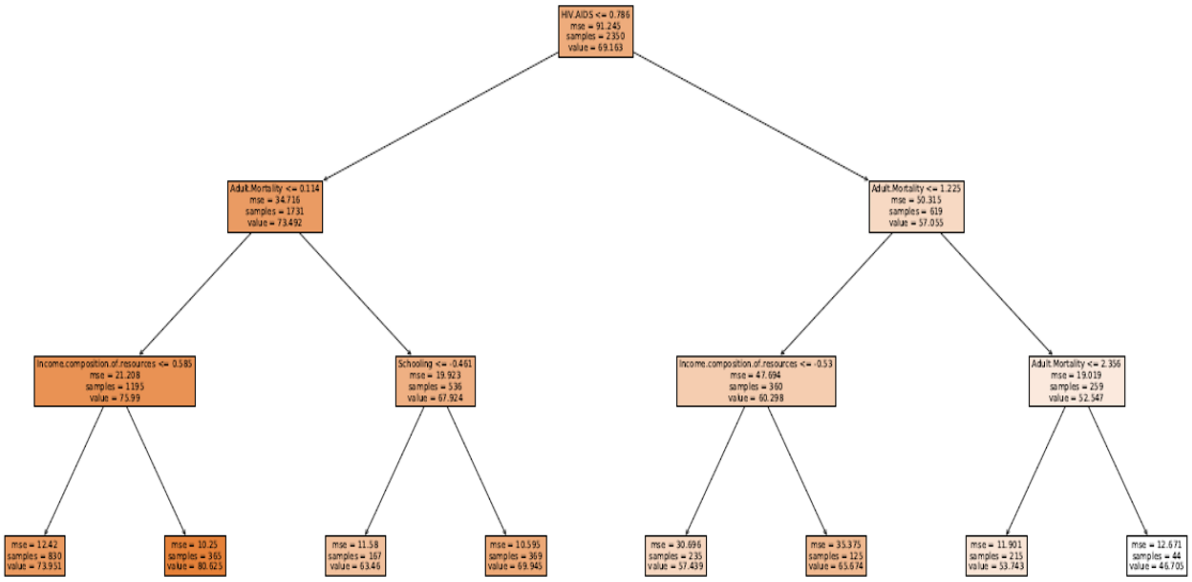
The below mentioned features are the ones which are sorted based on their importance (High – Low)

- HIV.AIDS
- Adult.Mortality
- Income.composition.of.resources
- Schooling
- BMI
- under.five.deaths
- Alcohol
- thinness.5.9.years
- Total.expenditure
- infant.deaths
- Hepatitis.B
- thinness..1.19.years
- percentage.expenditure
- Diphtheria
- Measles
- GDP
- Population
- Polio

Before tuning the Hyperparameters, r squared value which we got on validation set is 92.432. After tuning the hyperparameters, we have got MSE as 6.54 and r squared value is 92.47 on the validation dataset.

```
In [59]: mse
Out[59]: 6.535479415472587

In [61]: r2_score(y_test, y_pred_dt_best)*100
Out[61]: 92.46793654027331
```

The regression tree with max_depth = 3



The topmost node is the root node. Internal nodes are considered as decision nodes. The final subgroups at the bottom of the tree are called the *terminal nodes* or *leaves*.

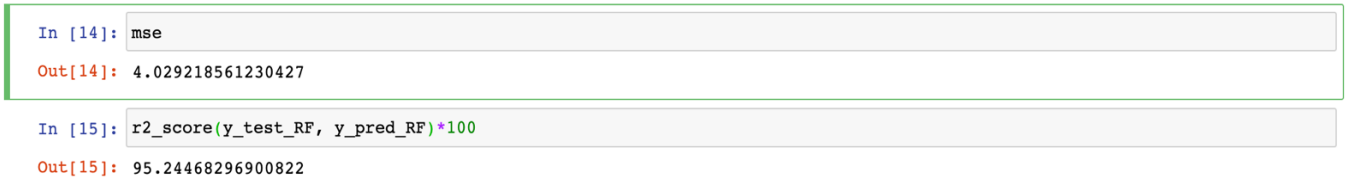
HIV Aids which is a root node here is the most important feature in determining life expectancy. At each internal node, it checks whether the associated condition is met or not and goes to the left child if the answer is yes, to the right child if the answer is No. So, the value 73.492 which is at the 2nd level left corner is the predicted life expectancy if the conditions of both HIV aids and adult mortality returns yes. As the value of max_depth increases, there will be an increase in the depth of the decision tree as well.

Each node has 3 sets of values i.e., MSE, Number of sample nodes, and the predicted life expectancy value.

5.5 Random Forest Regression

Random forest is a Supervised Learning ensemble algorithm for regression which takes a subset of observations and variables to build many decision trees in parallel and amalgamate them together to get a more accurate and stable prediction. The number of features that can be split on at each node is limited to some percentage of the total (which is known as the **hyperparameter**).

This makes sure that the ensemble model **doesn't totally rely on any particular single variable solely** and makes **fair use of all the variables.** The parameters in this model are either increase the predictive power of the model or make it simpler to train the model. The parameters which are chosen to build the random forest are max_depth, n_estimators, max_features, min_samples_split, min_samples_leaf, bootstrap, n_jobs and number of cross-validation folds. First, we broke the dataset into train and test data in the ratio of 70:30 respectively and then took the default values for the hyper-parameters and constructed a decision tree where the MSE value is 4.0292 and the r-squared value is 95.2446.



Picking the best hyperparameters is very important because it plays an equivalent role in developing a model and this in-turn helps in increasing the R-squared value, reducing the value of mean squared error. We have used the GridserachCV method, which is used to select the hyperparameters, the values are given as parameters to the method so that the best of all combinations are given and train the algorithm. This is measured using the cross-validation technique and the best score for the parameters is generated.

The value for max_depth is 30 which means that the maximum number of levels in each decision tree. The value of n_estimators is 30 which means that we are considering the number of trees in the forest before taking the predictions. Max_features is the max number of features considered for splitting a node (“sqrt”, “auto”, “log2”) where we have considered sqrt for our model. The value of min_samples_split is set to 2 which means that the min number of data points placed in a node before the node is split. The value of min_samples_leaf is set to 1 which means that min number of data points allowed in a leaf node. After assigning these values to the hyperparameters, we see that there is increase in the r-squared value which is 95.19 and the mean square error is 4.07.

```
In [19]: mse
Out[19]: 4.071936841647073

In [20]: r2_score(y_test_RF, y_pred_RF_best)*100
Out[20]: 95.19426650156846
```

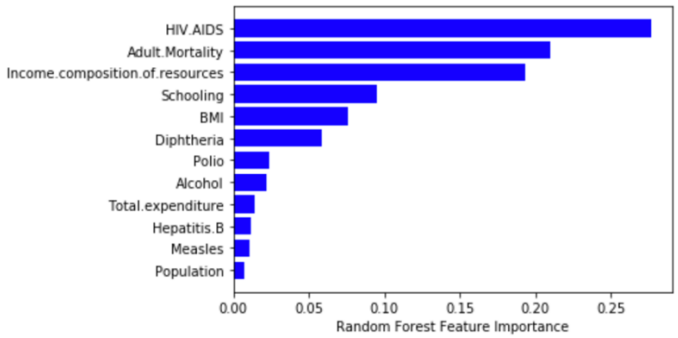
Feature importance is an approach of ranking the input features based on how important they are for predicting the target variable. It helps in having an effective way of having an understanding of the data and the model. This also plays an important role in figuring out the effectiveness of the predictive model. Here is the table, which shows the most important variables for the Random Forest model.

```
In [16]: feature_imp = pd.DataFrame(rf_regf_best.feature_importances_, index=X_train_RF.columns,
                                   columns=['importance']).sort_values('importance', ascending=False)
         feature_imp
```

Out[16]:

	importance
HIV.AIDS	0.277042
Adult.Mortality	0.209420
Income.composition.of.resources	0.193605
Schooling	0.095054
BMI	0.076143
Diphtheria	0.058470
Polio	0.023882
Alcohol	0.022479
Total.expenditure	0.014617
Hepatitis.B	0.011689
Measles	0.010364
Population	0.007236

The plot below shows the sorted version of the important features in the order of high to low.



(Koehrsen, 2018)

5.6 Gradient Boosting Model

Gradient Boosting or XGBoost is a powerful approach for building supervised regression models method which builds many decision trees sequentially. It combines a set of weak learners and delivers improved prediction accuracy. It contains loss function and a

regularization term. It tells about the difference between actual values and predicted values, i.e., how far the model results are from the real. Though GBM is fairly robust at higher number of trees but it can still be overfit at a point. Hence, this should be tuned using CV for a particular learning rate.

The parameters which are taken to build the model can be classified as

- **Tree-Specific Parameters:** These affect each individual tree in the model. max_depth, max_features, min_samples_split, min_samples_leaf.
- **Boosting Parameters which** affect the boosting operation in the model which are learning rate and number of estimators

Learning rate determines the impact of each tree on the final outcome.

With the default parameters, the model's mean square error is 5.1449 and the r-squared value is 94.4944.

```
In [144]: mse=mean_squared_error(y_test_L, reg_predict)
          mse

Out[144]: 5.14498207149252

In [145]: r2_score(y_test_L, reg_predict)*100

Out[145]: 94.49446929450419
```

Tree specific parameters are same as that of building the random forest model and the values that are set to the hyperparameters are as follows:

learning_rate: 0.15

max_depth: 7

max_features: 'auto'

min_samples_leaf: 5

min_samples_split: 2

n_estimators: 30

After the parameter tuning there is a decrease in the value of mean square error and also increase in the value of r squared value. The MSE for Gradient boosting model is 3.6530 and the r-squared value is 96.0909.

```
In [152]: gbm_bestfit= gbm_best.fit(X_train_L,y_train_L)
          y_pred_gbm = gbm_bestfit.predict(X_test_L)
          mse=mean_squared_error(y_test_L, y_pred_gbm)
          mse
Out[152]: 3.6530683676056674

In [153]: r2_score(y_test_L, y_pred_gbm)*100
Out[153]: 96.09093291528339
```

Compared to other regressor models, this gave best metrics and the least mean squared error and the highest r-squared value.

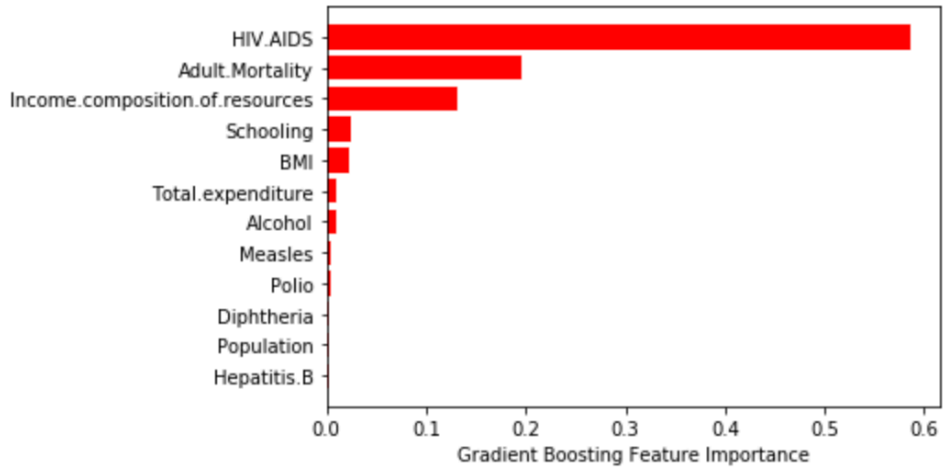
The following is the table that shows the features based on the importance that are used for the Gradient Boosting model. And below is the corresponding plot for the feature importance.

```
In [154]: feature_imp = pd.DataFrame(gbm_bestfit.feature_importances_, index=X_train_L.columns,
                                     columns=['importance']).sort_values('importance', ascending=False)
          feature_imp
```

Out[154]:

	importance
HIV.AIDS	0.587077
Adult.Mortality	0.196620
Income.composition.of.resources	0.132314
Schooling	0.025458
BMI	0.022191
Total.expenditure	0.009354
Alcohol	0.009335
Measles	0.005191
Polio	0.004641
Diphtheria	0.002797
Population	0.002572
Hepatitis.B	0.002450

The plot shows that features which are in the ascending order.



(JAIN, Complete Guide to Parameter Tuning in XGBoost with codes in Python, 2016)

6. Problem statements

We have added to our abstract a few problem statements which are nothing but the inspirations we had to take up the project and address the unanswered questions. The answers provide a summary on why we have performed the models and what conclusions have we drawn from the outcomes. The following are the questions we had along with the answers. (Life Expectancy (WHO))

6.1 Problem statement 1

Does the life expectancy affected by different forecasting variables that were originally chosen? What are the predictive factors that are currently impacting life expectancy?

Yes, definitely. There are 21 variables in the dataset apart from the target variable. Through correlation analysis, we have filtered out the variables that are less significant on the models, and that impacted achieving accuracy for the target variable. The predictive factors that are currently impacting life expectancy as per our analysis are Hepatitis B, Polio, Diphtheria, Measles, BMI, Total expenditure, Income composition, Adult mortality, HIV/AIDS, Developing status of the individual countries, Schooling, Year, Alcohol, Population.

6.2 Problem statement 2

Are the child and adult mortality rates influencing life expectancy? If yes, how?

Through the correlation analysis, it has been observed that the child mortality hasn't influenced much in predicting the life expectancy, but the principal component analysis has proved that the adult mortality rates show the highest influence on life expectancy.

We support our statement by checking the Eigenvalues that are greater than 1 and looking at the elbow in the scree plot. The PCA has returned us that only 2 principal components which explain 71.42% of the variance in our data.

6.3 Problem statement 3

What kind of correlation does life expectancy have with the predictors such as eating habits, lifestyle, exercise, smoking, drinking alcohol etc.

In this process of analysis, we have double-checked the correlation values with the scatter plots and concluded that most of the predictors have a high correlation.

Alcohol – 0.39 (low correlation)

BMI – 0.56 (high correlation)

Income composition – 0.69 (high correlation)

Schooling – 0.72 (high correlation)

6.4 Problem statement 4

Does education (schooling) of an individual impact the lifespan of humans in any way?

As already discussed in the previous slide, the correlation coefficient of the variable schooling on the life expectancy is 0.72 which means that both the variables are almost in a perfect positive correlation and move in the same direction together. We conclude that education has an impact on the target variable.

6.5 Problem statement 5

Is there a positive or negative correlation between life expectancy and consumption of alcohol?

The alcohol consumption has a positive correlation coefficient with the target variable, but it is a low correlation as the values are close to 0 which means that there is a weak or no linear relationship.

6.6 Problem statement 6

What are the effects of immunization coverage life expectancy?

The variables polio and diphtheria fall under the immunizations along with measles and Hepatitis-B where the correlation coefficient for the attributes polio and diphtheria have a better effect on the life expectancy. They both have a positive correlation around 0.4.

7. Conclusion

After performing all the models, it is found out that the model that gives us the best accuracy in predicting the life expectancy with the factors influencing it is **Gradient boosting method** with an accuracy (R2_score) of **96.1%** and the lowest error (mean squared error) of **3.65%**. For better understanding purpose, the scores for all the models are consolidated as follows

Model name	MSE	R2 Score
Multiple linear regression	18.36%	80.35%
Lasso regression	18.36%	80.35%
Ridge regression	18.63%	78.01%
Decision tree	6.53%	92.46%
Random forest	4.07%	95.19%
Gradient boosting method	3.65%	96.09%

8. How to run our code?

Step 1: Open "**OR568_final project_data cleaning (Team 11).R**" file in R studio

Step 2: Load the "**OR568_final project_R (Team 11).csv**" using the R command in line 2.

Step 3: Run the entire code (**Do not forget to run line 20**).

Step 4: A .csv file "**lifeexpectancy.csv**" will be generated after running line 20 in R file.

Step 5: Open the file "**OR568_final project python(Team 11).ipynb**" in Jupyter notebook and load the .csv generated in step 5.

Step 6: Proceed till last step in the above file to find out the best model.

9. References

Bhattacharyya, S. (2018, September 26). *Ridge and Lasso Regression: L1 and L2 Regularization.* Retrieved from towards datascience: https://towardsdatascience.com/ridge-and-lasso-regression-a-complete-guide-with-python-scikit-learn-e20e34bcbf0b

How can I avoid multicollinearity? (2015, February 12). Retrieved from ResearchGate: https://www.researchgate.net/post/How_can_I_avoid_multicollinearity

JAIN, A. (2016, January 28). *A Complete Tutorial on Ridge and Lasso Regression in Python.* Retrieved from analyticsvidhya: https://www.analyticsvidhya.com/blog/2016/01/ridge-lasso-regression-python-complete-tutorial/#three

JAIN, A. (2016, March 1). *Complete Guide to Parameter Tuning in XGBoost with codes in Python.* Retrieved from analyticsvidhya: https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python/

Kenton, W. (2020, September 21). *Multiple Linear Regression (MLR)*. Retrieved from Investopedia: https://www.investopedia.com/terms/m/mlr.asp

Koehrsen, W. (2018, January 10). *Hyperparameter Tuning the Random Forest in Python.* Retrieved from towardsdatascience: https://towardsdatascience.com/hyperparameter-tuning-the-random-forest-in-python-using-scikit-learn-28d2aa77dd74

Life Expectancy (WHO). (n.d.). Retrieved from kaggle: https://www.kaggle.com/kumarajarshi/life-expectancy-who

Mujtaba, H. (2020, September 29). *Hyperparameter Tuning with GridSearchCV.* Retrieved from mygreatlearning: https://www.mygreatlearning.com/blog/gridsearchcv/

Predicting housing prices using advanced regression techniques. (n.d.). Retrieved from Academia: https://www.academia.edu/39014594/Predicting_housing_prices_using_advanced_regression_techniques