Final Project: Predictive Models for the World Bank Indicators

Zulay Attanasio, Grace Shin, Ayushi Tiwari

George Mason University

STAT 663

December 2, 2019

Abstract

This paper analyzes the selected eight World Development Indicators and explores supervised machine learning models to predict life expectancy. Initially, it performs exploratory data analysis to identify patterns and correlations among the different indicators. Once some correlation is observed, specific machine learning models such as Regression and Random Forest are employed, and results are analyzed to predict the life expectancy of people.

Keywords: WDI, life expectancy, predictive models

The Dataset

The dataset was obtained by combining some of the World Bank's (WB) collection of development indicators displayed on their website. These Indicators are a compilation of relevant, high-quality and internationally comparable statistics about global development and the fight against poverty[1](The World Bank, 2019). The WB Data Catalog containing the indicators can be accessed by using the World Bank application programming interface (API). For this project, we chose the following indicators: Population-total, Life Expectancy at Birth (years), Total Youth Literacy Rate (% of people ages 15-24), Gross National Income (per capita) in US$, the Ratio of Female to Male Labor Force Participation Rate (%) (modeled ILO estimate), Employment in Industry (% of total employment) (modeled ILO estimate), Infant Mortality Rate (per 1000 live births), and Gross Domestic Product[2] in US$.

Data Exploration, Preprocessing and Manipulation

To determine which supervised learning models, suit better to the indicators, the dataset underwent exploration, preprocessing, and manipulation. Missing values were omitted and the

[1] The database contains 1,600 time series indicators for 217 economies and more than 40 country groups, with data for many indicators going back more than 50 years.

[2] The indicators descriptions can be found in Appendix A.

dataset was cleaned from special characters. The summary statistics based on the groups of observations in the dataset were disaggregated as well. In addition, column names were changed for convenience and categorical variables were changed from character and integer types to factor type to better manipulate the variables. Subsets of data were created to plot different exploratory graphics, correlation matrices, and statistical summaries.

Exploratory Data Analysis and Graphics

This project aims to focus on the exploratory graphics, predictive models and analysis of results. The exploratory data analysis starts with comparing indicators with each other and observing whether they are correlated. Exploratory graphics between the indicators Life expectancy (LE) v.s. Gross Domestic Product (GDP) and Infant mortality (IM) v.s. GDP for 2015, are presented as follows:

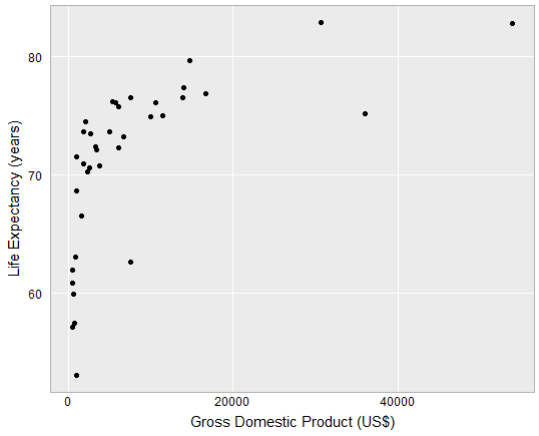


Figure 1. Life Expectancy vs. Gross Domestic Product: 2015

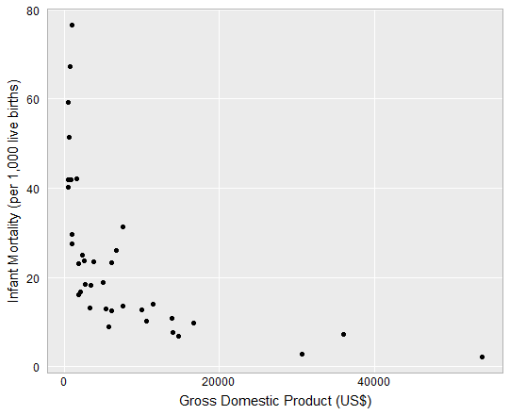


Figure 2. Infant Mortality v.s Gross Domestic Product: 2015

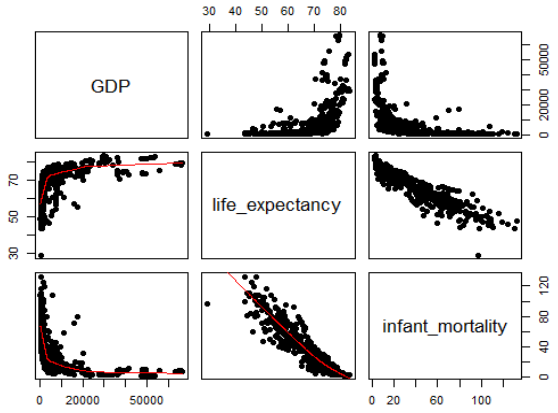


Figure 3. Scatterplot GDP, LE and IM:2015

Figures 1 and 2 displays the relationship between Life expectancy and Infant mortality v.s. GDP respectively. As observed, there is a moderate positive relationship between Life expectancy and GDP, which means that these variables are directly proportional. The better a country's GDP, it is assumed that the Life expectancy would increase. Likewise, Infant mortality and GDP present a negative relationship, meaning that these variables are inversely proportional. The correlation between the selected indicators would allow us to identify both the direction and the strength of any association.

Trends in Life Expectancy

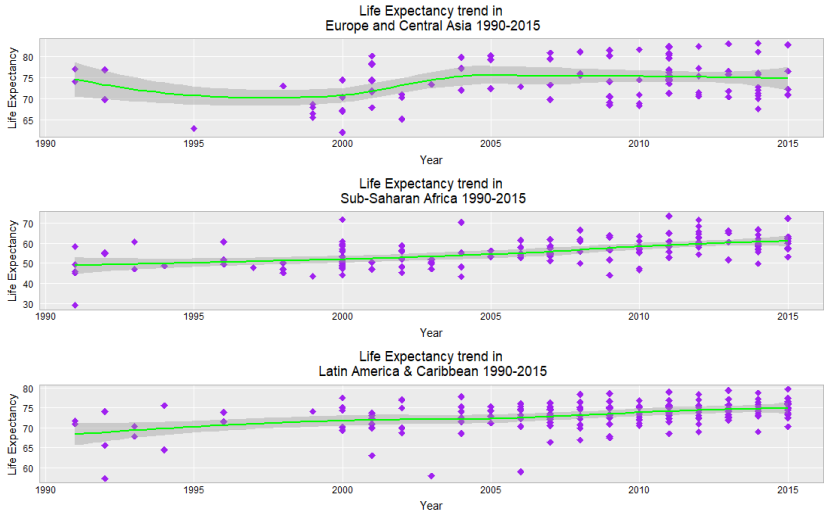


Figure 4. Trends in Life Expectancy

The above plot depicts the trends in life expectancy over a period of 25 years from 1990 - 2015. One can clearly observe a rising trend in life expectancy for major regions of the world.

Time Series Plot

The time series plot was also used in order to see the pattern of the data over the time period from 1990 to 2015 using the R package gridExtra for the 6 regions in the dataset. The regions are East Asia and Pacific, Europe and Central Asia, Latin American and Caribbean, Middle East and North Africa, South Asia, and Sub-Saharan Africa. To compare the regions side by side, the facet_grid command was used.

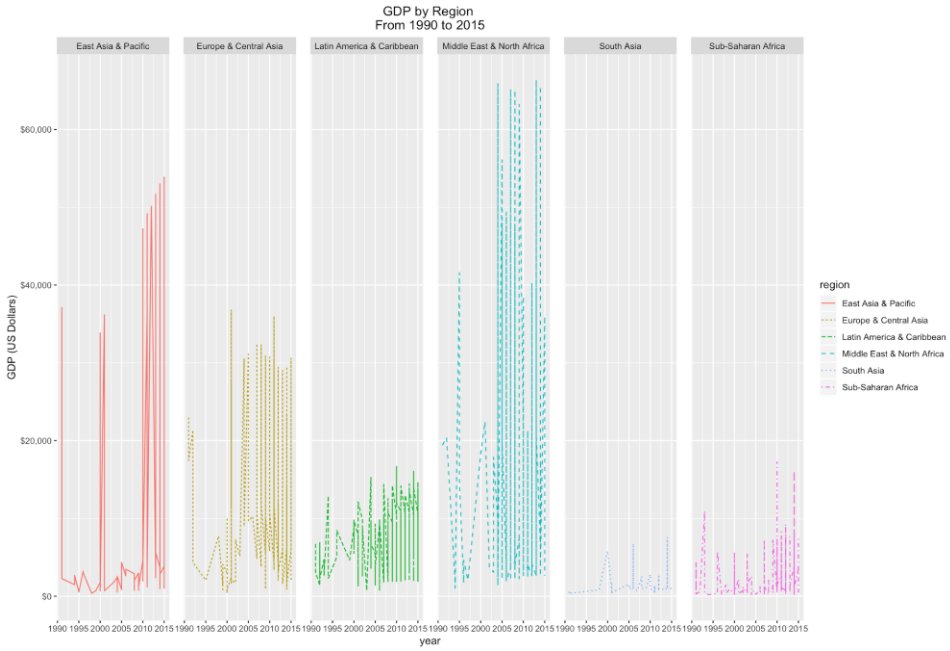


Figure 5. GDP by Regions in US Dollars from 1990 to 2015

The region with the lowest GDP are South Asia and Sub-Saharan Africa and the highest GDP were in the Middle East and North Africa and East Asia and Pacific regions. The GDP for the Middle East and North Africa region increased significantly after 2005.

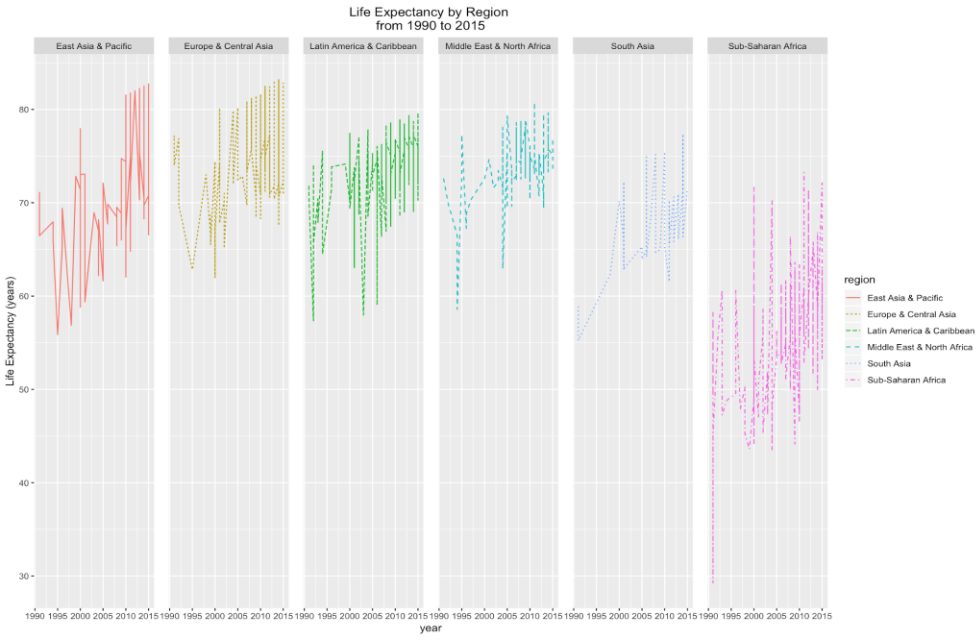


Figure 6. Life Expectancy Time Series Plot by Region from 1990 to 2015

The top 3 regions with the highest life expectancy are East Asia and Pacific, Europe and Central Asia, and Latin America and Caribbean. The region with the lowest life expectancy is Sub-Saharan Africa. One observation is that the Middle East and North Africa had the highest GDP but not the highest life expectancy for the years 1990 to 2015.

World Population in 2015

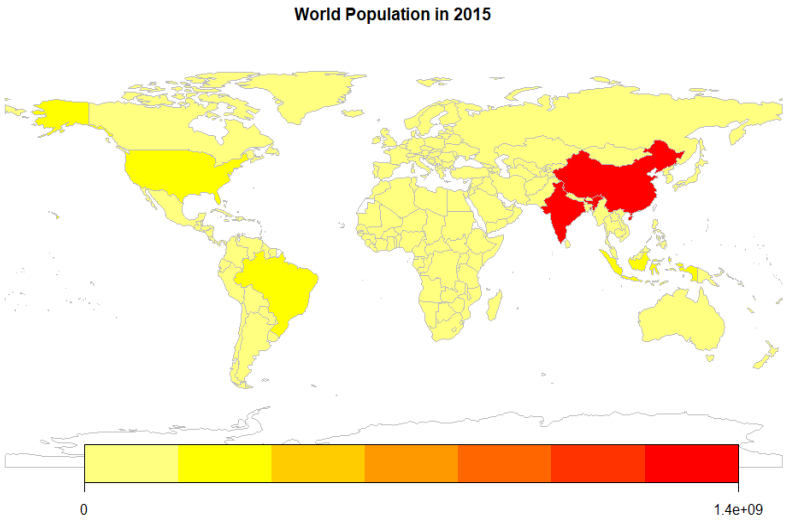


Figure 7. World Population: 2015

The above map was created using the rworldmap package from CRAN. The map highlights the highly populated countries of India and China and provides a rough estimation of population for the rest of the nations of the world.

Gross National Income (per capita) in 2015.

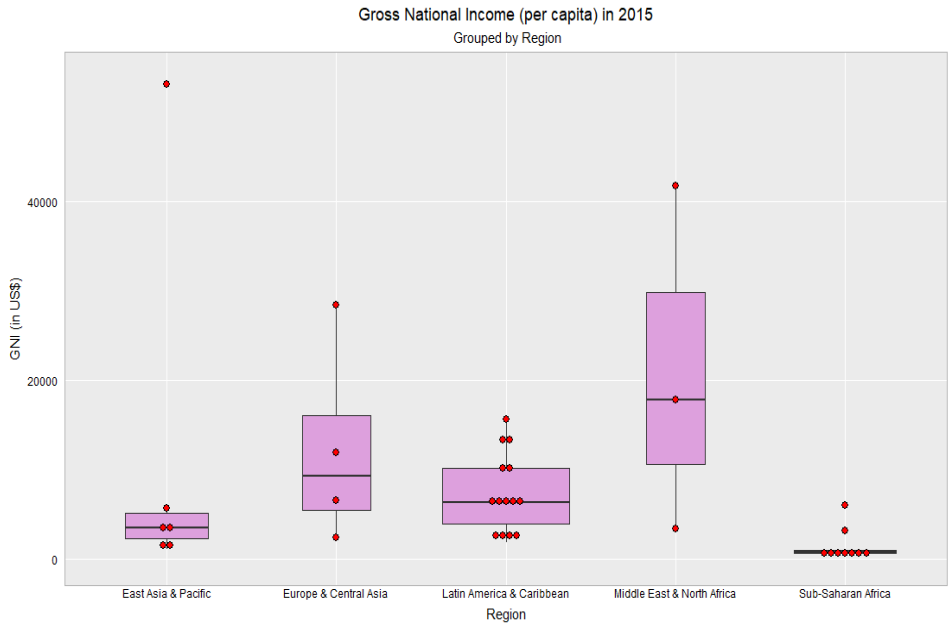


Figure 8. Gross National Income (per capita): 2015

The above box plot shows the distribution of Gross National Income across five major regions for the year 2015. Upon observation, one can see a clear outlier for East Asia and Pacific region. This outlier is Singapore, which is clearly explained by the flourishing economy of the nation. Two box plots are missing whiskers. This means that the lower quartile is equal to the minimum value and the upper quartile is equal to the maximum value in data. Sub-Saharan Africa has the least value for gross national income.

Distribution of Employment in the Industry - 2015

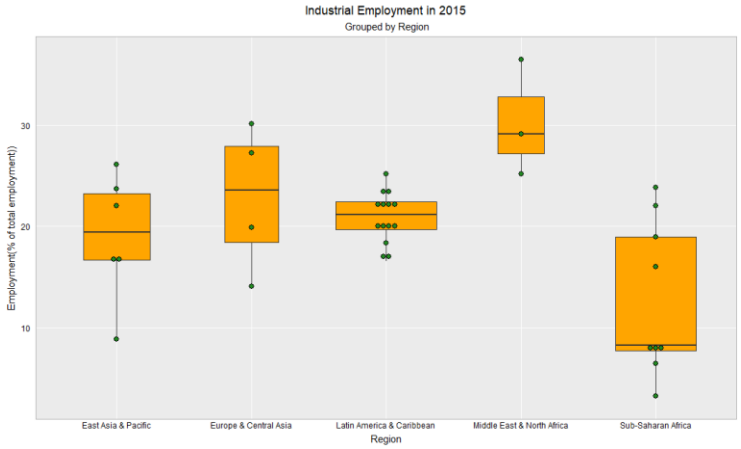


Figure 9. Industrial Employment: 2015

The above box plot is depicting the distribution of industrial employment in the industry for the year 2015. Sub-Saharan Africa has the highest variation in industrial employment rates followed by East Asia and Pacific. Middle East and North Africa has the highest rates of industrial employment. There are no observed outliers.

Gross National Income (per capita) as per Income Groups.

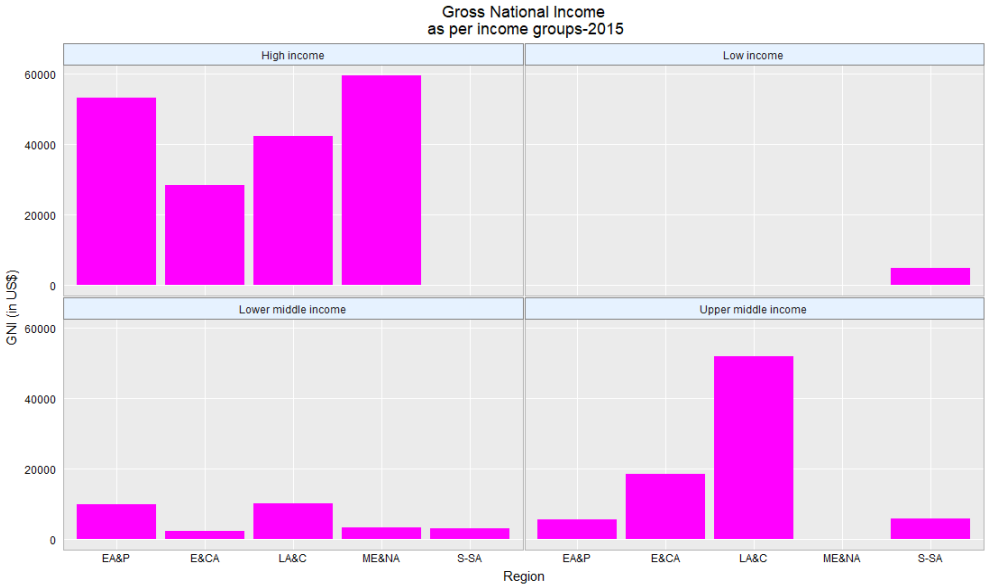


Figure 10. Gross National income (per capita): 2015

The above graph divides the different countries of the world into four income groups. The countries are placed into different income groups by World Bank on the basis of each country's Gross National Income, in US dollars, per capita. The majority of the countries in East Asia - Pacific and Middle East and East Africa fall into High Income groups, countries in Latin America and Caribbean in the Upper Middle-Income group, and countries in the Sub-Saharan Africa fall into the low-income group.

Predictive Models

Multiple Linear Regression using Cross Validation

Linear regression models the relationship between the magnitude of one variable and that of the second, quantifying the nature of the relationship. In this project, a Multiple linear regression using Cross-validation is selected to model and predict the Life expectancy. As an initial exploration, a correlation between independent variables (excluding the response variable) is displayed to observe every possible combination and detect which variables are highly correlated. This process allows getting a better response if any multicollinearity problem is present and our estimation is not stable. In other words, collinearity leads to overfitting the model. Thus, an alternative solution that selects the importance of variables should be performed—Ridge or LASSO regression will allow to shrink coefficients to prevent overfit.

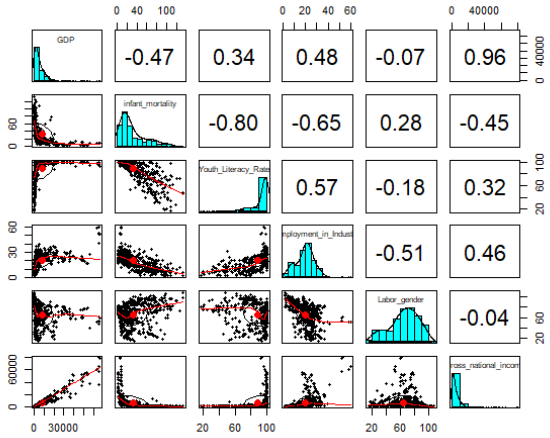


Figure 11. Correlation between independent variables.

As observed in Figure 11 there are some variables, Infant Mortality-Youth literacy and GDP-Gross national income, which present strong relationships. To perform the model, the dataset was *split into training to train our model and validation to test the model's prediction*. In addition, the variables infant mortality and population are excluded from this analysis. The results of the Multiple Linear Regression model using *Cross-validation* are as follows:

```
Call:
lm(formula = .outcome ~ ., data = dat)

Residuals:
    Min      1Q  Median      3Q     Max
-29.820  -2.018   1.063   2.972   9.464

Coefficients:
                          Estimate Std. Error t value Pr(>|t|)
(Intercept)             4.625e+01  1.693e+00  27.313  < 2e-16 ***
GDP                    -1.158e-04  9.159e-05  -1.264 0.206882
Youth_Literacy_Rate     2.751e-01  1.792e-02  15.347  < 2e-16 ***
Employment_in_Industry  1.173e-01  4.406e-02   2.662 0.008087 **
Labor_gender           -8.767e-02  1.369e-02  -6.404 4.23e-10 ***
Gross_national_income   3.384e-04  9.586e-05   3.530 0.000463 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.827 on 402 degrees of freedom
Multiple R-squared:  0.688,     Adjusted R-squared:  0.6841
F-statistic: 177.3 on 5 and 402 DF,  p-value: < 2.2e-16

> lm$results
  intercept     RMSE  Rsquared      MAE   RMSESD RsquaredSD     MAESD
1      TRUE 4.777486 0.6985261 3.511721 1.026892 0.09897678 0.5494867
```

```
Call:
lm(formula = .outcome ~ ., data = dat)

Coefficients:
           (Intercept)                    GDP     Youth_Literacy_Rate  Employment_in_Industry
             46.2495593             -0.0001158              0.2750562              0.1172597
          Labor_gender  Gross_national_income
            -0.0876731              0.0003384
```

In interpreting the results (only variables which present high importance according to the P-value —*** *represents importance of variables)*, it was found that for a given person in a specific country, life expectancy is predicted to increase by 0.0003384 years if the Gross National Income rate is increased by 1%, holding all other predictors fixed. Similarly, for a given person in a specific country, the life expectancy is predicted to decrease by 0.08767 years if the Labor Gender rate is increased by 1%, holding all other predictors fixed. In addition, it is expected that life expectancy increases by 0.2750 years if the variable Youth Literacy increases by 1%, holding all other predictors fixed.

Assessing the Model

In terms of assessing the model, the most important performance metric is the root mean square error (RMSE). This measures the overall accuracy of the model and is a basis for comparing it to other models. Lower values of RMSE indicates a better fit, which means that the predicted responses are very close to the true responses. For this case the RMSE is 4.77 Similarly, the coefficient of determination, known as the R-square, is useful in assessing how well the model fits the data and ranges between 0 to 1. In this case, R-squared has a value of 0.6985, indicating that the proposed model improves moderately prediction over the mean model. It is important to highlight that adding more variables to our model reduces RMSE and increases R-square, therefore other performance metrics such as BIC (Bayesian information criterion) which penalized-likelihood criteria could be used to help guide the model choice.

Diagnostics on the Linear Regression

After running the model, the diagnostics plots are generated to observe whether there is a violation of any of the residual assumptions.

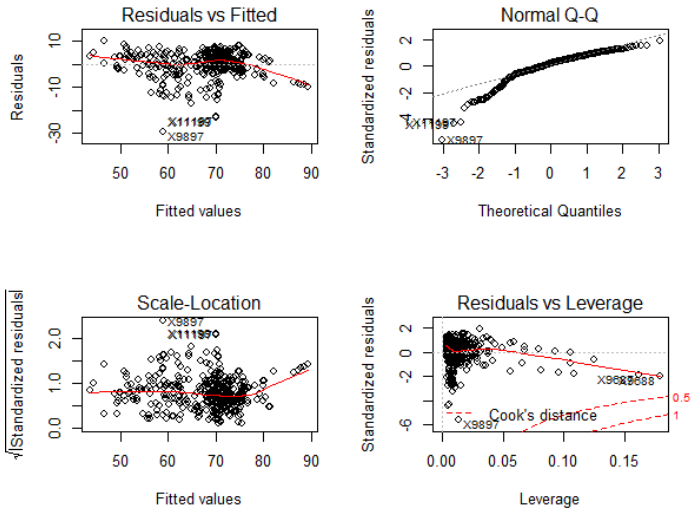


Figure 12. Diagnostic plot for the Multiple Linear Regression using Cross-Validation

Residual vs fitted: It shows a funnel pattern. The residual variance is lower for smaller values of predicted life expectancy, where variance increases as the predicted life expectancy increases. Therefore, the constant variance assumption (homoscedasticity) is violated.

Normal-(QQ) Plot: The plot looks fairly normal and the points seem to fall about a straight line. However, some of the points in the dataset do not lie in the straight line nearby to the tails. By using some transformation (Box-Cox) this pattern could be improved, allowing the data points to be normally distributed.

Ridge Regression

The Multiple linear regression model has a drawback. Adding more variables to the model incurred in a reduction of the RMSE and an increase in the R-square. Selecting subsets of predictors could be a solution. However, it is possible to fit a model containing all predictors and

shrink the coefficients towards zero. In addition, as observed in the correlation plot, there are near-linear relationships among some independent variables (collinearity). When multicollinearity occurs, least squares estimates are unbiased, but their variances are large so they may be far from the true value. Ridge Regression will allow to reduce the standard errors by adding a degree of bias to the regression estimates. The results of the Ridge Regression model are as follows:

```
glmnet

408 samples
  5 predictor

No pre-processing
Resampling: Cross-Validated (10 fold, repeated 5 times)
Summary of sample sizes: 367, 368, 366, 367, 367, 367, ...
Resampling results across tuning parameters:

  lambda    RMSE      Rsquared   MAE
  0.000100  4.797675  0.6965012  3.502860
  0.250075  4.797675  0.6965012  3.502860
  0.500050  4.797675  0.6965012  3.502860
  0.750025  4.803210  0.6960931  3.503368
  1.000000  4.815119  0.6953151  3.505859

Tuning parameter 'alpha' was held constant at a value of 0
RMSE was used to select the optimal model using the smallest value.
The final values used for the model were alpha = 0 and lambda = 0.50005.
```

The best value of the hyperparameter lambda is 0.50. As lambda increases the penalty increases as well, making coefficients to shrink. In other words, increasing lambda helps to reduce the size of the coefficients, especially the coefficients that are not contributing to the model.

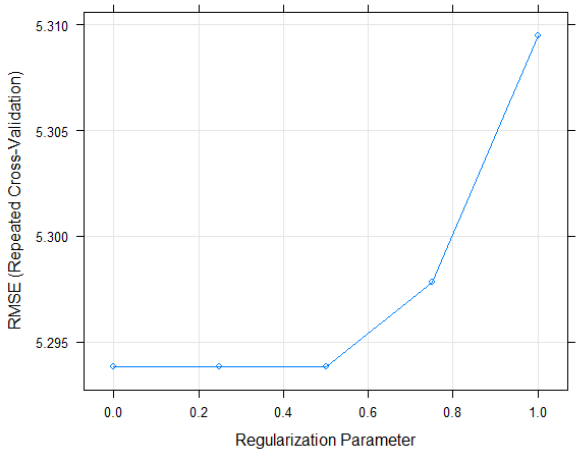


Figure 13. Regularization parameter lambda.

As observed in figure, as lambda increases the RMSE will increase as well. On the other hand, the Ridge regression model results are shown as follows:

	alpha	lambda	RMSE	Rsquared	MAE	RMSESD	RsquaredSD	MAESD
1	0	0.000100	4.797675	0.6965012	3.502860	1.023980	0.09844022	0.5501998
2	0	0.250075	4.797675	0.6965012	3.502860	1.023980	0.09844022	0.5501998
3	0	0.500050	4.797675	0.6965012	3.502860	1.023980	0.09844022	0.5501998
4	0	0.750025	4.803210	0.6960931	3.503368	1.023739	0.09835451	0.5517776
5	0	1.000000	4.815119	0.6953151	3.505859	1.022964	0.09827048	0.5544546

These results show that the RMSE is 4.7976 and R square is equal to 0.6965 when using a lambda equal to 0.50. These results will help when comparing the models performance. In addition, the variable importance plot displays the most important variables in the model representing which variables are relevant to predict a better model according to the Ridge regression. The variable Youth literacy, Employment in industry and Labor gender represent the most important variables when predicting Life expectancy in years. On the other hand, the variables GDP and Gross national income are the least important when predicting Life expectancy in years. This information can be used to re-run a Multiple Linear regression model taking only relevant variables.

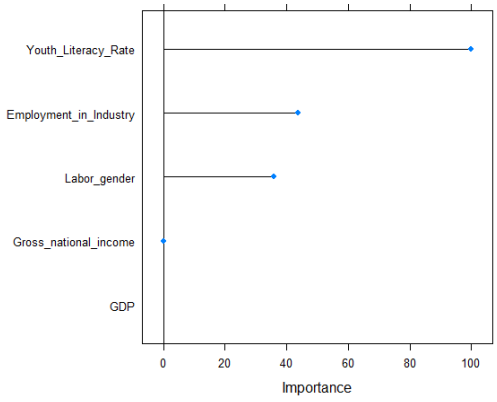


Figure 14. Variable importance Ridge regression

LASSO Regression

Different from ridge regression which includes all predictors in the final model, the LASSO regression model performs variable selection. As a result, models generated from the LASSO are generally much easier to interpret than those produced by ridge regression. LASSO regression was performed getting the following results:

```
glmnet

408 samples
  5 predictor

No pre-processing
Resampling: Cross-Validated (10 fold, repeated 5 times)
Summary of sample sizes: 367, 368, 366, 367, 367, 367, ...
Resampling results across tuning parameters:

  lambda    RMSE      Rsquared   MAE
  0.000100  4.786667  0.6973441  3.512888
  0.075075  4.777485  0.6983626  3.498588
  0.150050  4.778549  0.6982016  3.493729
  0.225025  4.782423  0.6979634  3.490471
  0.300000  4.789079  0.6976462  3.488882

Tuning parameter 'alpha' was held constant at a value of 1
RMSE was used to select the optimal model using the smallest value.
The final values used for the model were alpha = 1 and lambda = 0.075075.
```

These results show that the RMSE is 4.77 and R square is equal to 0.6983 when using a small lambda (0.0750). Figure # displays which lambda values improve the RMSE, highest values of lambda increase the RMSE.

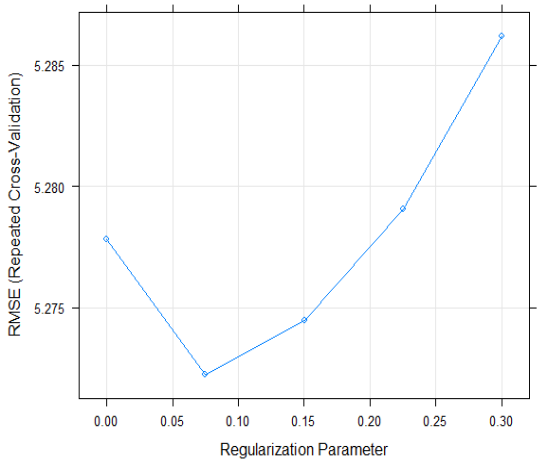


Figure 15. Best lambda

In addition, it was found that 60% of the variability is being explained only by four variables. The variable that grows rapidly next to the bottom corner has the least importance in the model compared to the variable at the top corner.

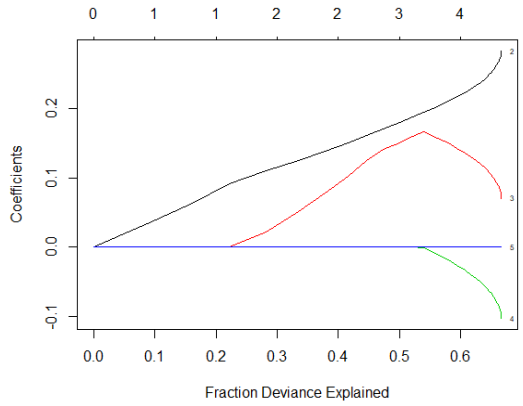


Figure 16. Coefficients explained

The variable importance plot displays the following results:

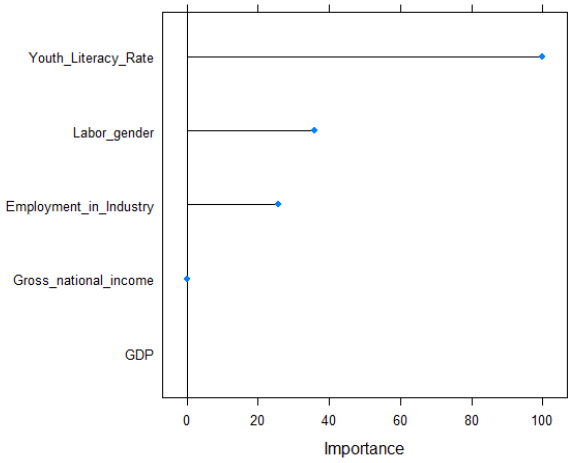


Figure 17. Variable importance Lasso

Similar to Variable importance in the Ridge regression model Figure 17 presents which variables are relevant to predict a better model according to the LASSO regression. The variable Youth literacy, Labor gender and Employment in industry represent the most important variables when

predicting Life expectancy in years. Observe that different from the results obtained in the Ridge regression model, the variable Labor gender is the second most important variable.

It is possible to conclude which model is the best by comparing the performance of the models developed previously using the performance metrics RMSE and R Square.

```
Call:
summary.resamples(object = res)

Models: LinearModel, Ridge, Lasso
Number of resamples: 50

MAE
                Min.  1st Qu.   Median     Mean  3rd Qu.     Max. NA's
LinearModel 2.505590 3.186090 3.456008 3.511721 3.771726 5.007938    0
Ridge       2.564342 3.182557 3.384062 3.502860 3.767238 4.876054    0
Lasso       2.530658 3.199341 3.437357 3.498588 3.766499 4.978213    0

RMSE
                Min.  1st Qu.   Median     Mean  3rd Qu.     Max. NA's
LinearModel 3.256848 4.155094 4.541735 4.777486 5.089675 7.509318    0
Ridge       3.365972 4.112413 4.524097 4.797675 5.163717 7.447526    0
Lasso       3.306208 4.097504 4.508901 4.777485 5.085996 7.522619    0

Rsquared
                Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
LinearModel 0.4394797 0.6280319 0.7187469 0.6985261 0.7831890 0.8381015    0
Ridge       0.4303975 0.6361774 0.7137894 0.6965012 0.7784887 0.8399028    0
Lasso       0.4370144 0.6302278 0.7195993 0.6983626 0.7822084 0.8376269    0
```

	Model		
Coefficients	lm	Ridge	Lasso
Intercept	46.249	47.300	46.382
GDP	-0.0001158	0.0000429	-
Youth literacy	0.2750	0.2489	0.2080
Employement in Industry	0.1172	0.1461	0.1142
Labor gender	-0.08767	-0.078	-0.08403
GNI	0.0003384	0.0001675	0.000215
RMSE	4.777	4.797	4.777
R Square	0.6985	0.6965	0.6983

Table 1: Model performance

According to Table 1, the models having the best performance are “lm” and “Lasso”. The Linear regression model presents a slightly higher R Squared and similar RMSE than Lasso regression. In addition, these models also present similar estimated coefficients. In this case, Linear

regression and Lasso models display similar performance, so, it is not clear which one performs better.

Random Forest

The Random Forest model also helped determine which variables are the most significant in predicting life expectancy. The 7 variables used for the model were GDP, life expectancy, youth literacy rate, employment in industry, ratio of the labor rate for females and males, total population, and the gross national income. The data was split into training and validation sets using life expectancy as the predictor variable. The model predicted that for 500 trees and 2 variables at each split the mean of the squared residuals is approximately 9.968, which has the square root of approximately 3.157.

```
> wdi7cols_Rf1

Call:
 randomForest(formula = life_expectancy ~ ., data = newWDI_7col,      importance = TRUE, proximity = TRUE)
               Type of random forest: regression
                     Number of trees: 500
No. of variables tried at each split: 2

          Mean of squared residuals: 9.967994
                    % Var explained: 87.61
> sqrt(9.967994)
[1] 3.157213
```

The variable importance was also calculated with the values below and the plot in Figure 18.

```
> importance(wdi7cols_Rf1)
                        %IncMSE IncNodePurity
GDP                    25.09219      7816.554
Youth_Literacy_Rate    23.99138      7671.548
Employment_in_Industry 15.91686      5468.631
Labor_gender           24.97913      2651.820
Pop_total              22.21699      1636.491
Gross_national_income  27.74374      8803.578
region                 45.58116     11794.141
>
```

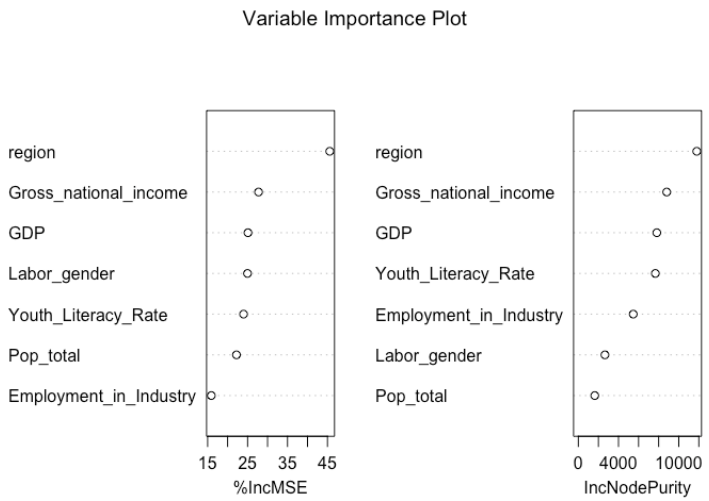


Figure 18. Variable Importance Plot

The variable importance plot predicts for the life expectancy, the most important variables are region, gross national income, GDP, and the labor gender. The plot of the model is in Figure 19, which illustrates that as the number of trees increases, the error rate decreases.

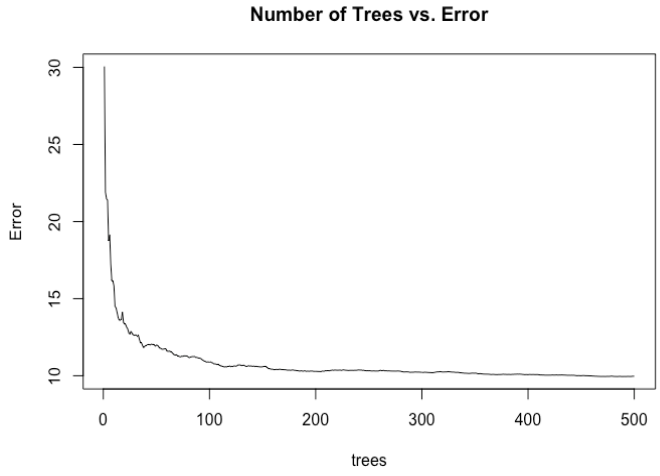


Figure 19. Number of trees vs. Error Rate

According to the plot, the model shows that the error rate decreases when the number of trees is around 100. The error rate decreases after 100 trees and stabilizes between 300 to 400 trees. Cross validation methods with the training and testing data were also used with bagging methods

to determine how the mean squared error, or the MSE, changes with different number of trees and variables. The results of the cross validation with the train and test cases are below:

```
> #cross validation with train and test cases from ISLR
> set.seed(1)
> wdi7cols_train = sample(1:nrow(newWDI_7col), nrow(newWDI_7col)/2)
> newWDI_7colTrain <-
+   randomForest(life_expectancy ~.,data=newWDI_7col,subset=wdi7cols_train,
+                importance=TRUE)
> yhat = predict(newWDI_7colTrain,newdata=newWDI_7col[-wdi7cols_train,])
> mse <- mean(newWDI_7col$life_expectancy[-wdi7cols_train]- yhat)^2
> mse
[1] 0.0318196
> #reference: https://uc-r.github.io/random_forests & https://www.tutorialspoint.com/r/r_random_forest.htm
> #number of trees with lowest MSE
> which.min(wdi7cols_Rf1$mse)
[1] 488
> #RMSE of optimal random forest
> sqrt(wdi7cols_Rf1$mse[which.min(wdi7cols_Rf1$mse)])
[1] 3.153816
>
```

The mean squared error for the training data is calculated to be approximately 0.0318, which is found when there are 488 trees and the root mean squared error is approximately 3.154.

The bagging model was used to calculate values of the MSE for different number of splits at each variable, mtry, and the number of trees, ntree. The results are shown below and in Table 2.

```
> bag1.wdi7cols <- randomForest(life_expectancy ~., data=newWDI_7col,
+                               mtry = 7, ntree = 25)
> bag1.wdi7cols

Call:
 randomForest(formula = life_expectancy ~ ., data = newWDI_7col,      mtry = 7, ntree = 25)
               Type of random forest: regression
                     Number of trees: 25
No. of variables tried at each split: 7

          Mean of squared residuals: 10.91778
                    % Var explained: 86.43
> bag2.wdi7cols <- randomForest(life_expectancy ~., data=newWDI_7col,
+                               mtry = 7, ntree = 100)
> bag2.wdi7cols

Call:
 randomForest(formula = life_expectancy ~ ., data = newWDI_7col,      mtry = 7, ntree = 100)
               Type of random forest: regression
                     Number of trees: 100
No. of variables tried at each split: 7

          Mean of squared residuals: 10.07358
                    % Var explained: 87.47
>
```

Number of Trees	Number of Variables	MSE	Square Root of MSE	Mean of Train & Test Set
25	7	11.12457	3.335351556	2.130926
100	7	10.45783	3.23385683	2.214761
200	7	10.17041	3.189108026	2.181051
300	7	10.23651	3.199454641	2.137391
400	7	9.990693	3.160805752	2.119512
500	7	9.833014	3.135763703	2.118915

Table 2. Number of Trees, Variables, and MSE for Random Forest Model

As the number of trees increased, the MSE decreased for the 7 variables. At 25 trees, the MSE was approximately 11.125 and at 500 trees the MSE was 9.833. The MSE did not change significantly between 400 and 500 trees for the 7 variables. At 400 trees the MSE is approximately 9.991 and at 500 trees is approximately 9.833.

Conclusion

The project used a variety of exploratory graphics to assess the different indicators of World Bank for correlation, prediction and pattern detection. Upon visualization, certain patterns were observed in data and correlation among variables was identified. Supervised machine learning models such as Regression (Multiple Linear, Ridge and Lasso) and Random Forest were employed, and results were analyzed to predict the life expectancy of people in different countries of the world. For the Linear Regression model, Life expectancy was predicted to increase by 0.0003384 years if the Gross National income rate is increased by 1%, holding all other predictors fixed. In addition, it is expected that Life expectancy increase by 0.2750 years if the variable Youth literacy increase by 1%, holding all other predictors fixed. In case of Ridge Regression, the variable Youth literacy, Employment in industry and Labor gender represent the most important variables when predicting Life expectancy in years. On the other hand, the variables GDP and Gross national income are the least important. In case of LASSO Regression, as per the variable importance plot, the variable Youth literacy, Labor gender and Employment in

Industry represent the most important variables when predicting the life expectancy in years. For the Random Forest model, the important variables to predict life expectancy were Region, Gross National Income, GDP, and Labor Gender.

Future Work

This project is a data analysis general overview, much work could be done. Data exploration can be improved, and transformation to normality using tools such as Box-cox transformation can be performed, which would improve models' performance metrics. Analysis of the variables can also improve the data in terms of skewness and outliers, possibly leading to an improvement in the models. In addition, other indicators can be added to generate more predictors. For instance, the variables income and population could be highly correlated to Life expectancy in years. In this case, LASSO regression could be more appropriate. A cluster analysis could be done to subset the data and build a model for each subset.

References

James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani, An Introduction to Statistical Learning with Applications in R, Springer, 2013.

ColorBrewer 2.0. (n.d.). Retrieved from http://colorbrewer2.org/#type=sequential&scheme=RdPu&n=3.

De Cock, D. (2011). Ames, Iowa: Alternative to the Boston housing data as an end of semester regression project. Journal of Statistics Education, 19(3).

Fox, J. & Weisberg, S. (2011). An {R} Companion to Applied Regression, Second Edition. Thousand Oaks CA: Sage. URL:http://socserv.socsci.mcmaster.ca/jfox/Books/Companion

Grothendieck, G. (2017). sqldf: Manipulate R Data Frames Using SQL. R package version 0.4-11. https://CRAN.R-project.org/package=sqldf

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning (Vol. 112, p. 18). New York: springer.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2017). ISLR: Data for an Introduction to Statistical Learning with Applications in R. R package version 1.2. https://CRAN.R-project.org/package=ISLR

Lander, Jared P. (2017). R For Everyone: Advanced Analytics. Boston: Addison-Wesley

Liaw, A. & Wiener, M. (2002). Classification and Regression by randomForest. RNews 2(3),

18--22.

World Development Indicators. (n.d.). Retrieved from http://datatopics.worldbank.org/world-development-indicators/.

APPENDIX A

Indicators Description:

Life expectancy at birth (years) shows the number of years a newborn infant would live if prevailing patterns of mortality at the time of its birth were to stay the same throughout its life.

Youth literacy rate is the percentage of people ages 15-24 who can both read and write with understanding a short simple statement about their everyday life.

GDP per capita is gross domestic product divided by midyear population. GDP is the sum of gross value added by all resident producers in the economy plus any product taxes and minus any subsidies not included in the value of the products. It is calculated without making deductions for depreciation of fabricated assets or for depletion and degradation of natural resources. Data are in constant 2010 U.S. dollars.

Total population is based on the de facto definition of population, which counts all residents regardless of legal status or citizenship. The values shown are midyear estimates.

Employment Industry. Employment is defined as persons of working age who were engaged in any activity to produce goods or provide services for pay or profit, whether at work during the reference period or not at work due to temporary absence from a job, or to working-time arrangements. The industry sector consists of mining and quarrying, manufacturing, construction, and public utilities (electricity, gas, and water), in accordance with divisions 2-5 (ISIC 2) or categories C-F (ISIC 3) or categories B-F (ISIC 4).

Infant mortality. Completeness of infant death reporting is the number of infant deaths reported by national statistics authorities to the United Nations Statistics Division's Demographic Yearbook divided by the number of infant deaths estimated by the United Nations Population Division.

Gross national income per capita based on purchasing power parity (PPP). PPP GNI is gross national income (GNI) converted to international dollars using purchasing power parity rates. the **Ratio of Female to Male Labor Force Participation Rate** is calculated by dividing female labor force participation rate by male labor force participation rate and multiplying by 100.

APPENDIX B

Downloading and Cleaning Data from WDI Database

```
library(WDI)
#downloading data
WDI_dataset <- WDI(indicator = c("NY.GDP.PCAP.KD", "SP.DYN.LE00.IN",
"SP.DYN.IMRT.IN", "SE.ADT.1524.LT.ZS","SL.IND.EMPL.ZS","SL.TLF.CACT.FM.ZS","
SP.POP.TOTL","NY.GNP.PCAP.CD"), start = 1960, end = 2015, extra = TRUE)
View(WDI_dataset)

#renaming columns
WDI_dataset <- subset(WDI_dataset, region != "Aggregates")
names(WDI_dataset)[which(names(WDI_dataset) == "SP.POP.TOTL")] <- "Pop_total"
names(WDI_dataset)[which(names(WDI_dataset) == "NY.GDP.PCAP.KD")] <- "GDP"
names(WDI_dataset)[which(names(WDI_dataset) == "SP.DYN.LE00.IN")] <- "life_expectancy"
names(WDI_dataset)[which(names(WDI_dataset) == "SP.DYN.IMRT.IN")] <-
"infant_mortality"
names(WDI_dataset)[which(names(WDI_dataset) == "SE.ADT.1524.LT.ZS")] <-
"Youth_Literacy_Rate"
names(WDI_dataset)[which(names(WDI_dataset) == "SL.IND.EMPL.ZS")] <-
"Employment_in_Industry"
names(WDI_dataset)[which(names(WDI_dataset) == "SL.TLF.CACT.FM.ZS")] <-
"Labor_gender"
names(WDI_dataset)[which(names(WDI_dataset) == "NY.GNP.PCAP.CD")] <-
"Gross_national_income"
WDI_dataset<-na.omit(WDI_dataset)

#data summary
dim(WDI_dataset)
str(WDI_dataset)
summary(WDI_dataset)
```

Sample of WDI Database Data Used:

country	year	GDP	life_expectancy	infant_mortality	Youth_Literacy_Rate	Employment_in_Industry	Labor_gender	SP.POP.TOTL	Gross_national_income	iso3c
Indonesia	2015	3824.2749	70.76800	23.5	99.67007	22.038	61.56150	258383256	3430	IDN
Brazil	2015	11431.1545	74.99400	14.0	98.96375	22.159	70.77245	204471769	10160	BRA
Bangladesh	2015	1002.3889	71.51400	29.6	87.88877	19.929	40.29414	156256276	1220	BGD
Mexico	2015	10037.2015	74.90400	12.7	98.94471	25.160	55.35349	121858258	10170	MEX
Philippines	2015	2605.4936	70.64400	23.7	99.08260	16.635	64.56927	102113212	3510	PHL
Turkey	2015	13853.0971	76.53200	10.9	99.49439	27.226	43.89390	78529409	11960	TUR
Thailand	2015	5741.3397	76.09100	9.0	98.14663	23.681	78.39019	68714511	5710	THA
South Africa	2015	7556.7659	62.64900	31.4	98.95578	23.828	77.22206	55386367	6050	ZAF
Tanzania	2015	871.9984	63.11100	41.9	85.75514	6.484	90.97802	51482633	980	TZA
Colombia	2015	7572.3655	76.53100	13.5	98.53473	19.825	71.34738	47520667	7330	COL
Spain	2015	30595.1568	82.83171	2.7	99.65568	19.904	80.85405	46444832	28460	ESP
Argentina	2015	10568.1578	76.06800	10.2	99.55970	23.567	65.20020	43131966	12600	ARG
Uzbekistan	2015	1831.3229	70.92800	23.0	100.00000	30.116	68.89032	31298900	2440	UZB
Peru	2015	6114.4300	75.79200	12.5	99.00954	16.570	80.81759	30470734	6340	PER
Mozambique	2015	529.0911	57.20600	59.3	70.52510	7.746	97.26367	27042002	600	MOZ
Chile	2015	14722.3663	79.64600	6.7	99.35394	23.284	66.95215	17969353	14140	CHL
Mali	2015	727.4611	57.50900	67.2	49.36653	8.261	75.45399	17438778	790	MLI

Countries:

```
> unique(WDI_dataset$country)
  [1] "United Arab Emirates"     "Afghanistan"              "Albania"
  [4] "Armenia"                  "Angola"                   "Argentina"
  [7] "Azerbaijan"               "Bosnia and Herzegovina"   "Barbados"
 [10] "Bangladesh"               "Burkina Faso"             "Bulgaria"
 [13] "Bahrain"                  "Burundi"                  "Benin"
 [16] "Brunei Darussalam"        "Bolivia"                  "Brazil"
 [19] "Bhutan"                   "Botswana"                 "Belarus"
 [22] "Belize"                   "Congo, Dem. Rep."         "Central African Republic"
 [25] "Congo, Rep."              "Cote d'Ivoire"            "Chile"
 [28] "Cameroon"                 "China"                    "Colombia"
 [31] "Costa Rica"               "Cuba"                     "Cabo Verde"
 [34] "Cyprus"                   "Dominican Republic"       "Algeria"
 [37] "Ecuador"                  "Estonia"                  "Egypt, Arab Rep."
 [40] "Eritrea"                  "Spain"                    "Ethiopia"
 [43] "Fiji"                     "Gabon"                    "Georgia"
 [91] "Malaysia"                 "Mozambique"               "Niger"
 [94] "Nigeria"                  "Nicaragua"                "Nepal"
 [97] "Oman"                     "Panama"                   "Peru"
[100] "Papua New Guinea"         "Philippines"              "Pakistan"
[103] "Poland"                   "West Bank and Gaza"       "Portugal"
[106] "Paraguay"                 "Qatar"                    "Romania"
[109] "Serbia"                   "Russian Federation"       "Rwanda"
[112] "Saudi Arabia"             "Sudan"                    "Singapore"
[115] "Slovenia"                 "Sierra Leone"             "Senegal"
[118] "Suriname"                 "Sao Tome and Principe"    "El Salvador"
[121] "Eswatini"                 "Chad"                     "Togo"
```

Regions:

```
> unique(WDI_dataset$region)
[1] Middle East & North Africa South Asia                  Europe & Central Asia
[4] Sub-Saharan Africa         Latin America & Caribbean  East Asia & Pacific
8 Levels: Aggregates East Asia & Pacific Europe & Central Asia ... Sub-Saharan Africa
>
```

APPENDIX C

```
# 0. Set-Up:Downloading packages needed
install.packages('WDI')
#install.packages('ggplot2') # install if you do not have it
#install.packages('tidyverse')
#install.packages('dplyr')
install.packages('GGally')
install.packages("plotly")
install.packages('rworldmap')
library(WDI)
library(wbstats)
library(ggplot2)
library(dplyr)
library(tidyverse)
library(GGally)
```

```
library(corrplot)
library(ggcorrplot)
library("PerformanceAnalytics")
library(rworldmap)
library(RColorBrewer)
library(gridExtra)
library(randomForest)
hw <- theme_gray()+ theme(
  plot.title=element_text(hjust=0.5),
  plot.subtitle=element_text(hjust=0.5),
  plot.caption=element_text(hjust=-.5),
  #  strip.text.y = element_blank(),
  strip.background=element_rect(fill=rgb(.9,.95,1),
                                colour=gray(.5), size=.2),
  panel.border=element_rect(fill=FALSE,colour=gray(.70)),
  panel.grid.minor.y = element_blank(),
  panel.grid.minor.x = element_blank(),
  panel.spacing.x = unit(0.050,"cm"),
  panel.spacing.y = unit(0.050,"cm"),
  # axis.ticks.y= element_blank()
  axis.ticks=element_blank(),
  axis.text=element_text(colour="black"),
  axis.text.y=element_text(margin=margin(0,3,0,3)),
  axis.text.x=element_text(margin=margin(-1,0,3,0))
)
# Visualizations
plot_1<-ggplot(subset(WDI_dataset, year == 2015), aes(x = GDP, y = infant_mortality)) +
geom_point() + hw +
  labs(x ='Gross Domestic Product (GDP)', y='Infant Mortality',title = 'Infant Mortality v.s Gross
Domestic Product: 2015')
plot_2<-ggplot(subset(WDI_dataset, year == 2015), aes(y = GDP, x = life_expectancy)) +
geom_point() + hw +
  labs(y ='Gross Domestic Product (GDP)', x='Life Expectancy',title = 'Life Expectancy v.s Gross
Domestic Product: 2015')
plot_3<-ggplot(WDI_dataset, aes(y = Employment_in_Industry, x = Youth_Literacy_Rate)) +
geom_point() + hw +
  labs(y ='Employment in Industry', x='Youth Literacy Rate',title = 'Does literacy rate affect
employment in industry')
pairs(WDI_dataset[4:6], col="black",lower.panel = panel.smooth,pch=19, main="Scatterplot
World Bank")
#ggpairs(WDI_dataset,mapping=aes(col=Type),axisLabels="internal")# not necessary
# Creating Correlation matrix
my_data<-WDI_dataset[,c(4,5,6,7,8,9)]
head(my_data,6)
res<-cor(my_data)
round(res,2)
```

```
cor(my_data, use = "complete.obs") #handle missing values

corrplot(res, type = "upper", order = "hclust",
         tl.col = "black", tl.cex=1,tl.srt = 45)+hw
# Create correlogram
ggcorrplot(res, hc.order = TRUE,
         type = "lower",
         lab = TRUE,
         lab_size = 2,
         method="circle",
         colors = c("tomato2", "white", "springgreen3"),
         title="Correlogram of WDI Indicators")+hw
#The function chart.Correlation() in the package [PerformanceAnalytics], can be used to display
a chart of a correlation matrix.
my_data <- WDI_dataset[,c(4,6,7,8,9,5)]
chart.Correlation(my_data, histogram=TRUE, pch=19) +hw
# The above graph provides the following information:
#Correlation coefficient (r) - The strength of the relationship.
#p-value - The significance of the relationship. Significance codes 0 '***' 0.001 '**' 0.01 '*' 0.05
'.' 0.1 ' ' 1
#Histogram with kernel density estimation and rug plot.
#Scatter plot with fitted line.
#create a heatmap
col<- colorRampPalette(c("red", "white", "blue"))(10)
#png("S10.png", width=1000, height=1000)
heatmap(x = res, col = col, symm = TRUE)+hw
dev.off()
#Create world map for WDI indicators with rworldmap
indicator<-"SP.POP.TOTL"
dFpopulation <- WDI( indicator=indicator,
                     start=2015,
                     end=2015 )
sPDFpopulation <-
  joinCountryData2Map( dFpopulation,
                     nameJoinColumn= "iso2c",
                     joinCode='ISO2')
#numCats <- 5
#colourPalette <- rev(brewer.pal(numCats, "PuBuGn"))
mapCountryData( sPDFpopulation,
                     nameColumnToPlot=indicator,
                     mapTitle="World Population in 2015", catMethod = "pretty",
                     colourPalette = "heat")+hw
mapCountryData( sPDFpopulation,
                     nameColumnToPlot=indicator,
                     mapTitle="Eurasia Population in 2005", catMethod = "pretty",
                     mapRegion= "Eurasia",colourPalette = "heat")
```

```
#Gross National Income\n as per income groups-2015
ggplot(subset(WDI_dataset, year == 2015 & region!="South Asia",
        select=c(region,Gross_national_income,income)),
  aes(x=region, y=Gross_national_income))+
  geom_bar(stat='identity', fill="magenta")+
  scale_x_discrete(labels = abbreviate)+
  facet_wrap(~income)+hw+
  ggtitle("Gross National Income\n as per income groups-2015") +
  xlab("Region") +
  ylab("GNI (in US$)")
# time series plot
timeseries1<-ggplot(subset(WDI_dataset, region == "Europe & Central Asia"),
                    aes(year, life_expectancy)) +
  geom_point(na.rm=TRUE,color="purple", size=3, pch=18)+hw+
  ggtitle("Life Expectancy trend in\n Europe and Central Asia 1990-2015") +
  xlab("Year") +
  ylab("Life Expectancy")
timeseries1
timeseries1_trend <- timeseries1 + stat_smooth(colour="green")
timeseries1_trend
timeseries2<-ggplot(subset(WDI_dataset, region == "Sub-Saharan Africa"),
                    aes(year, life_expectancy)) +
  geom_point(na.rm=TRUE,color="purple", size=3, pch=18)+hw+
  ggtitle("Life Expectancy trend in\n Sub-Saharan Africa 1990-2015") +
  xlab("Year") +
  ylab("Life Expectancy")
timeseries2
timeseries2_trend <- timeseries2 + stat_smooth(colour="green")
timeseries2_trend
timeseries3<-ggplot(subset(WDI_dataset, region == "Latin America & Caribbean"),
                    aes(year, life_expectancy)) +
  geom_point(na.rm=TRUE,color="purple", size=3, pch=18)+hw+
  ggtitle("Life Expectancy trend in\n Latin America & Caribbean 1990-2015") +
  xlab("Year") +
  ylab("Life Expectancy")
timeseries3
timeseries3_trend <- timeseries3 + stat_smooth(colour="green")
timeseries3_trend
grid.arrange(timeseries1_trend, timeseries2_trend, timeseries3_trend,ncol=1)
# Create Box plot for GNI
g <- ggplot(subset(WDI_dataset, year == 2015 & region!="South Asia"),
        aes(region, Gross_national_income))
g + geom_boxplot(varwidth=T, fill="plum") + geom_dotplot(binaxis='y',
                                          stackdir='center',
                                          dotsize = .5,
                                          fill="red") +
```

```
  labs(title="Gross National Income (per capita) in 2015",
          subtitle=" Grouped by Region",
          caption="Source: mpg",
          x="Region",
          y="GNI (in US$)")+hw
#Box plot for life expectancy
q <- ggplot(subset(WDI_dataset, year == 2015 & region!="South Asia"),
          aes(region, life_expectancy))
q + geom_boxplot(varwidth=T, fill="green") + geom_dotplot(binaxis='y',
                                          stackdir='center',
                                          dotsize = .5,
                                          fill="black")+
  labs(title="Life Expectancy in 2015",
          subtitle=" Grouped by Region",
          x="Region",
          y="Life Expectancy(in years))")+hw
# Box plot for Employment in industry
r <- ggplot(subset(WDI_dataset, year == 2015 & region!="South Asia"),
          aes(region, Employment_in_Industry))
r + geom_boxplot(varwidth=T, fill="orange") + geom_dotplot(binaxis='y',
                                          stackdir='center',
                                          dotsize = .5,
                                          fill="forest green")+
  labs(title="Industrial Employment in 2015",
          subtitle=" Grouped by Region",
          x="Region",
          y="Employment(% of total employment))")+hw
```

Sample code for time series plots:

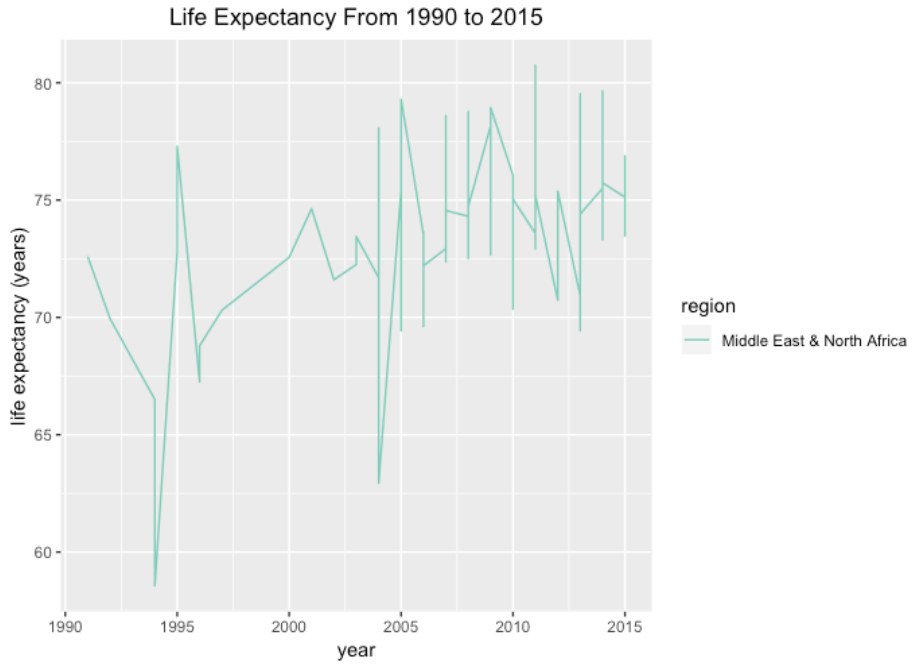
```
timeseriesplt <- ggplot(WDI_dataset, aes(year, life_expectancy, color = region, linetype =
region)) + geom_line() + scale_y_continuous(name="Life Expectancy (years)") + ggtitle('Life
Expectancy by Region \nfrom 1990 to 2015')+ theme(plot.title = element_text(hjust = 0.5))

timeseriesplt2 <- ggplot(WDI_dataset, aes(year, GDP, color = region, linetype = region)) +
geom_line() + scale_y_continuous(name = 'GDP (US Dollars)',label = dollar) + ggtitle('GDP by
Region \nFrom 1990 to 2015')+ theme(plot.title = element_text(hjust = 0.5))
```

Sample of subsetting by regions and using color brewer to change the color lines:

```
df.wdidataset <- data.frame(WDI_dataset)
MENA <- subset(df.wdidataset, df.wdidataset$region == 'Middle East & North Africa')
SA <- subset(df.wdidataset, df.wdidataset$region == 'South Asia')
ECA <- subset(df.wdidataset, df.wdidataset$region == 'Europe & Central Asia')
SubSA <- subset(df.wdidataset, df.wdidataset$region == 'Sub-Saharan Africa')
LaAMCa <- subset(df.wdidataset, df.wdidataset$region == 'Latin America & Caribbean')
EAPac <- subset(df.wdidataset, df.wdidataset$region == 'East Asia & Pacific')
```

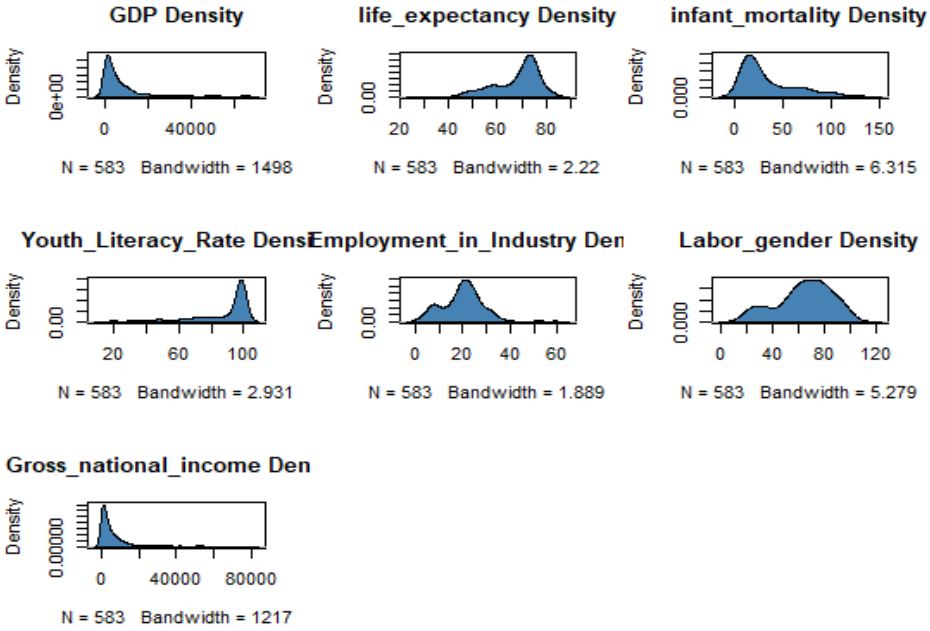
```
ts_menalexp <- ggplot(MENA, aes(year, life_expectancy, color = region, linetype = region)) +
  geom_line(color = '#7fcdbb') + scale_y_continuous(name="life expectancy (years)") +
ggtitle('Life Expectancy From 1990 to 2015')+ theme(plot.title = element_text(hjust = 0.5))
```




```
# === Visualization 2-- Or create multiple plots with a loop ===
colnames(WDI_dataset)
y<-WDI_dataset[,c(4,5,6,7,8,9,11)]
par(mfrow = c(3, 3))
for (i in 1:ncol(y)) {
  boxplot(y[ ,i], xlab = names(y[i]), horizontal=T,main = paste(names(y[i]), "BoxP"), col="steelblue")
}+hw
```

```
#=== Visualization 3-- Kernel Density Plot ===

par(mfrow = c(3, 3))
for (i in 1:ncol(y)) {
  d<- density(y[,i], na.rm = TRUE)
  plot(d, main = paste(names(y[i]), "Density"))
  polygon(d, col="steelblue")}+hw
```




```
#==== Multiple Linear Regression Model using Cross Validation ====
str(WDI_dataset)
dim(WDI_dataset)

# Looking for correlation between independent variables
pairs.panels(WDI_dataset[c(-1,-2,-3,-10,-12,-13,-14,-15,-16,-17,-18)], cex= 1.25)

# Data Partition
set.seed(12345) # Set Seed to reproduce the same sample in future also
# Now Selecting 70% of data as sample from total 'n' rows of the data
sample_WB <- sample.int(n = nrow(WDI_dataset), size = floor(.70*nrow(WDI_dataset)), replace = F)
train_WB<- WDI_dataset[sample_WB, ] # data into training
test_WB  <- WDI_dataset[-sample_WB, ]# data into Validation
dim(train_WB)

# Custom Control parameters
custom<-trainControl(method ="repeatedcv",
                     number =10,
                     repeats = 5,
                     verboseIter = T)
```

```
# linear model
set.seed(12345)
lm <- train(life_expectancy ~ GDP + Youth_Literacy_Rate +
              Employment_in_Industry + Labor_gender+ Gross_national_income,
            train_WB,
            method = 'lm',
            trControl = custom) #model

# Results
lm
summary(lm)
lm$results
par(mfrow = c(2, 2))
plot(lm$finalModel)

# Summary of the regression
BIC(lm$finalModel)
sm_1= summary(lm$finalModel)
sm_1$r.squared
mean((test_WB$life_expectancy -predict.lm(lm$finalModel, test_WB))^2) # MSE in validation test
P_WB<-predict.lm(lm$finalModel, test_WB)
length(P_WB)
head(P_WB)

# Ridge Regression
set.seed(12345)
ridge <- train(life_expectancy ~ GDP+ Youth_Literacy_Rate + Employment_in_Industry +
                 Labor_gender+ Gross_national_income,train_WB,
               method = 'glmnet',
               tuneGrid =expand.grid(alpha=0,
                                     lambda =seq(0.0001,1,length=5)),
               trControl = custom) #model
# Results
ridge
plot(ridge)
plot(varImp(ridge,scale=T))
ridge$results

# LASSO Regression
set.seed(12345)
lassso<-train(life_expectancy ~ GDP  + Youth_Literacy_Rate +
                Employment_in_Industry + Labor_gender+ Gross_national_income,
              train_WB,
              method = 'glmnet',
              tuneGrid =expand.grid(alpha=1,
                                    lambda =seq(0.0001,0.3,length=5)),
              trControl = custom) #model
# Results
plot(lassso)
```

```
lassso
plot(lassso$finalModel,xvar='dev', label=T)
plot(varImp(lassso,scale=T))

# Comparing models
summary(lm$finalModel)
lm$finalModel
#lassso
lassso$bestTune
best<-lassso$finalModel
coef(best, s =lassso$bestTune$lambda)
#ridge
ridge$bestTune
best<-ridge$finalModel
coef(best, s =ridge$bestTune$lambda)

#random forest starts here:

set.seed(137)
wdi7cols_Rf0 <- randomForest(x=newWDI_7col[,-2],y=newWDI_7col[,2],
                    data=newWDI_7col)
wdi7cols_Rf0 #model summary

wdi7cols_Rf1 <- randomForest(life_expectancy ~., data=newWDI_7col,
                   importance=TRUE, proximity=TRUE )
wdi7cols_Rf1

importance(wdi7cols_Rf1)
varImpPlot(wdi7cols_Rf1, main = "Variable Importance Plot")
plot(wdi7cols_Rf1, main = 'Number of Trees vs. Error')

#cross validation with train and test cases from ISLR
set.seed(1)
wdi7cols_train = sample(1:nrow(newWDI_7col), nrow(newWDI_7col)/2)
newWDI_7colTrain <-
  randomForest(life_expectancy ~.,data=newWDI_7col,subset=wdi7cols_train,
            importance=TRUE)
yhat = predict(newWDI_7colTrain,newdata=newWDI_7col[-wdi7cols_train,])
mse <- mean(newWDI_7col$life_expectancy[-wdi7cols_train]- yhat)^2
mse

#reference: https://uc-r.github.io/random_forests &
https://www.tutorialspoint.com/r/r_random_forest.htm
#number of trees with lowest MSE
which.min(wdi7cols_Rf1$mse)
#RMSE of optimal random forest
sqrt(wdi7cols_Rf1$mse[which.min(wdi7cols_Rf1$mse)])
```

```
#test data
newwdi7col.test=newWDI_7col[-wdi7cols_train,"life_expectancy"]

#bagging:

#with all 7 variables:
#25 trees
bag1.wdi7cols <- randomForest(life_expectancy ~., data=newWDI_7col,
                      mtry = 7, ntree = 25)
bag1.wdi7cols

yhat.bag1 = predict(bag1.wdi7cols,newWDI_7col[-wdi7cols_train,])
mean((yhat.bag1-newwdi7col.test)^2)

plot(yhat.bag1, newwdi7col.test,las=1,)
abline(0,1)
title(main=paste("WDI World Bank Data",
              "Life Expectancy"),sep="\n")

#100 trees
bag2.wdi7cols <- randomForest(life_expectancy ~., data=newWDI_7col,
                       mtry = 7, ntree = 100)
bag2.wdi7cols

yhat.bag2 = predict(bag2.wdi7cols,newWDI_7col[-wdi7cols_train,])
mean((yhat.bag2-newwdi7col.test)^2)

#200 trees
bag3.wdi7cols <- randomForest(life_expectancy ~., data=newWDI_7col,
                       mtry = 7, ntree = 200)
bag3.wdi7cols

yhat.bag3 = predict(bag3.wdi7cols,newWDI_7col[-wdi7cols_train,])
mean((yhat.bag3-newwdi7col.test)^2)

#500 trees
bag3.wdi7cols <- randomForest(life_expectancy ~., data=newWDI_7col,
                       mtry = 7, ntree = 500)
bag3.wdi7cols

#test set
yhat.bag3 = predict(bag3.wdi7cols,newWDI_7col[-wdi7cols_train,])
mean((yhat.bag3-newwdi7col.test)^2)


#500 trees with 5 variables
```

```
bag4.wdi7cols <- randomForest(life_expectancy ~., data=newWDI_7col,
                        mtry = 5, ntree = 500)
bag4.wdi7cols

#testing & training set
yhat.bag4 = predict(bag4.wdi7cols,newWDI_7col[-wdi7cols_train,])
mean((yhat.bag4-newwdi7col.test)^2)

#500 trees with 2 variables
bag5.wdi7cols <- randomForest(life_expectancy ~., data=newWDI_7col,
                        mtry = 2, ntree = 500)
bag5.wdi7cols

#test set
yhat.bag5 = predict(bag5.wdi7cols,newWDI_7col[-wdi7cols_train,])
mean((yhat.bag5-newwdi7col.test)^2)

#300 trees with 7 variables
bag6.wdi7cols <- randomForest(life_expectancy ~., data=newWDI_7col,
                        mtry = 7, ntree = 300)
bag6.wdi7cols

#test set
yhat.bag6 = predict(bag6.wdi7cols,newWDI_7col[-wdi7cols_train,])
mean((yhat.bag6-newwdi7col.test)^2)

#400 trees with 7 variables
bag7.wdi7cols <- randomForest(life_expectancy ~., data=newWDI_7col,
                        mtry = 7, ntree = 400)
bag7.wdi7cols

#test set
yhat.bag7 = predict(bag7.wdi7cols,newWDI_7col[-wdi7cols_train,])
mean((yhat.bag7-newwdi7col.test)^2)
```