University of Essex

Department of Mathematical Sciences

MA317-7-SP: Modelling Experimental Data

Group Project

Registration numbers:

1802655, 1907394, 2007519, 2003861, 2004352, 2007161

Group: 9

Date of submission: (26 March 2021)

Word count: 3205

Abstract

World Development Indicators (WDI) are drawn from a primary World Bank database and outline many factors relating to a country's population. These include features such as life expectancy, national income, and birth/mortality rate. However, this dataset is largely incomplete and contains many missing values. Within this report, we outline how to effectively impute onto these missing values to create a statistically robust dataset. We then investigate three different regression models in order to accurately predict life expectancy based on the other features within the WDI dataset, and then perform a prediction on a test set with the life expectancy label missing. Finally, we employ a one-way ANOVA test to analyse the different life expectancies of different continents.

1. Introduction

Life expectancy at birth indicates the average age of death for the population of a country (1). It is considered a good indicator of a country's' population health (2). Inequalities in life expectancies are found across countries and socio-economic factors such as health, education, employment, and healthcare spending have been linked to life expectancy (1)(3).

Health factors such as infant mortality and healthcare spending have been linked to lower life expectancies and employment levels also influence the life expectancy rate as those in employment have higher life expectancies (4)(5). It has been found that life expectancy increases as educational attainment increases, though as education levels increase the effect on life expectancy becomes less (6).

GDP per capita is reliable indicator to a country's health as the greater the wealth of a country the higher the life expectancy and the number of families living below the poverty line decreases the life expectancy of a country (4).

In this study we will predict the life expectancy at birth (our dependant variable) of 232 different countries using 20 features from data taken from the World Bank Indicators Database. We will implement the following models to perform our predictions: random forest regressor, multiple linear regressors, and support vector regressor. We will then evaluate the performance of each of these models and apply the most effective one to perform predictions on a test set.

2. Preliminary Analysis

2.1 Observation of the distribution

A value greater than +/-1 shows the distribution of a variable is highly skewed. A positive skew indicates many low values in the distribution while a negative skew indicates many high values. Appendix 1 shows 14 variables in the dataset fit these criteria.

The high kurtosis values for ‘Adjusted net national income’ and ‘Population, total’ indicate a high, sharp peak with heavy-tailed distribution. Appendix 3 shows the distributions as histograms for each variable and confirms these variables are skewed.

Performing a Shapiro Wilk test showed the p-value for ‘Expenditure on primary education’ is greater than 0.05 and the distribution is normally distributed. All other dependent variables have a p-value less than 0.05 showing the data for these variables are not normally distributed.

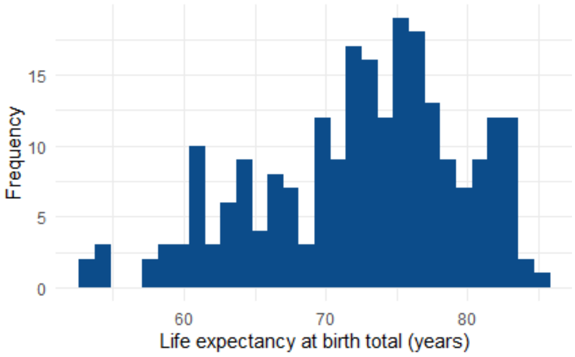


Figure 1. Histogram of the target variable, Life expectancy at birth (years)

Figure 1. shows the distribution of the target ‘Life expectancy at birth (years)’ which shows the data are not normally distributed. Performing a Shapiro Wilk test for normality gave a p-value of 0.0000103, confirming that the target is not normally distributed.

Outliers are present in 18 of the 20 variables (Appendix 3). Total population contains the highest number with 47 outliers. The variables ‘Access to electricity (% of population)’ and ‘Birth rate, crude (per 1,000 people)’ contain no outliers.

2.2 Regression test

Features	Correlation coefficient	p-value
Access to electricity (% of population)	0.80	4.81E-53
Adjusted net national income (annual % growth)	-0.08	0.3179

Adjusted net national income (constant 2010 US$)	0.40	7.05E-08
Children out of school (% of primary school age	-0.65	1.28E-20
Expenditure on primary education (% of government expenditure on education	-0.45	0.0317
Mortality rate, infant (per 1,000 live births)	-0.93	6.78E-100
Literacy rate, adult total (% of people ages 15 and above)	0.80	1.59E-25
Population growth (annual %)	-0.55	2.54E-19
Population, total	-0.04	0.5250
Primary completion rate, total (% of relevant age group)	-0.12	3.75E-20
Current health expenditure (% of GDP)	0.38	7.44E-09
Current health expenditure per capita, PPP (current international $)	0.85	1.84E-60
Unemployment, total (% of total labour force) (national estimate)	-0.09	0.2787
Mortality rate, adult, female (per 1,000 female adults)	-0.96	3.69E-106
Mortality rate, adult, male (per 1,000 male adults)	-0.91	2.72E-72
GDP growth (annual %)	-0.08	0.2229
GDP per capita, PPP (current international $)	0.85	2.59E-62
Birth rate, crude (per 1,000 people)	-0.88	5.92E-77
GNI per capita, PPP (current international $)	0.86	1.03E-63
Employment to population ratio, 15+, total (%)	-0.12	0.0696

Table 1. Correlation tests for feature variables against life expectancy

2.2.1 p-value

The p-value for four variables, 'Adjusted net national income (annual % growth)', 'Population, total', 'Unemployment, total (% of total labour force)', 'GDP growth (annual %)' and 'Employment to population ratio, 15+, total (%)', are greater than the statistically significant level of 0.05 and so are not considered statistically significant (Table 1). All other independent features have a p-value less than 0.05 and therefore can be considered statistically significant.

2.2.2 Correlation coefficient

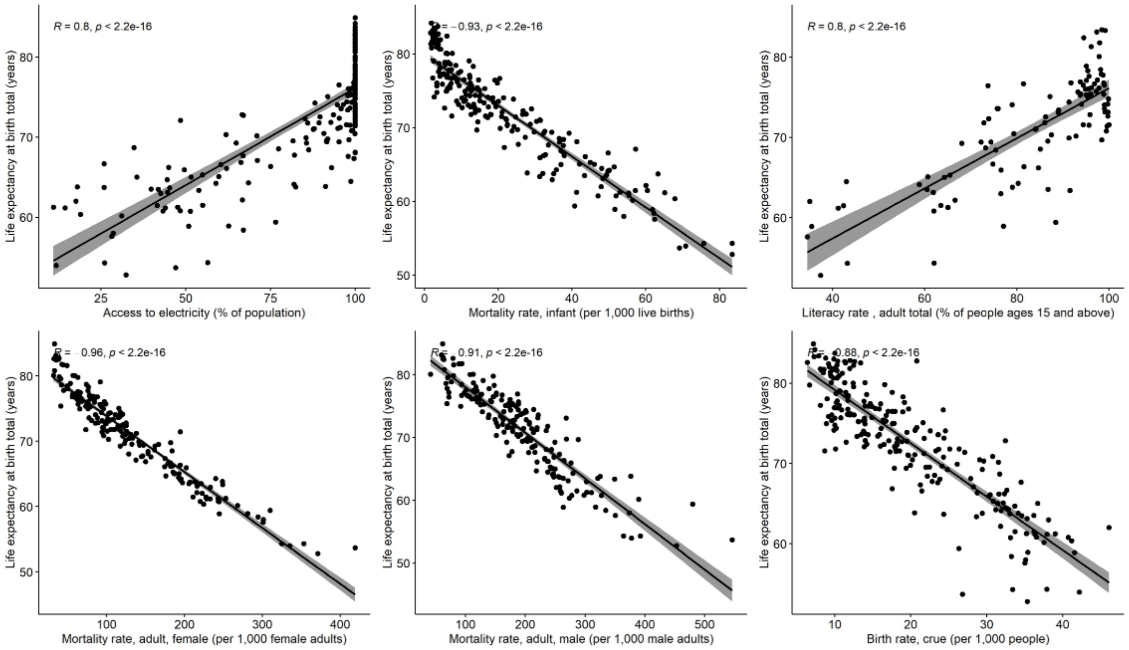


Figure 2. Scatter plots of variables with high correlation with the target variable (greater or equal to 0.8).

Six variables were found to have a high correlation with the target variable (Figure 2). ‘Access to electricity’ and ‘Literacy rate, adult total’ display a strong positive correlation with Life expectancy at birth indicating that as they increase, ‘Life expectancy at birth’ increases.

‘Mortality rate, infant’, ‘Mortality rate adult, Female’, ‘Male Mortality rate adult, male’ and ‘Birth rate’ display a strong negative correlation with ‘Life expectancy at birth’ indicating that as these rates increase, ‘Life expectancy at birth’ decreases.

2.3 Missing Values

Of the 20 variables in the dataset, 19 contain missing values. Five contain more than 25% missing values and the two variables ‘Expenditure on primary education’ and ‘Literacy rate adult total’ contain more than 50% missing values (Appendix 1 and 2). The target variable, ‘Life Expectancy at birth’ contains one missing value.

As the dataset contains such a large number of missing values, a complete case method of deleting rows with any null is not appropriate as that would result in the majority of the dataset being removed. This is because most of the rows have a missing value in one of their columns, there are only 7 rows with complete data in the entire dataset. As we cannot use the complete case method, we investigated other methods to replace the missing values.

Imputing the mean, mode, the median is an easy method to implement, but by imputing the same values in place of all the missing values significantly undermines the variance in the data. Instead, we will implement the MICE (Multivariate Imputation Via Chained Equations) package in R for imputation. This package creates multiple imputations (replacement values) for multivariate missing data (7).

We assume that the data is missing values at random, and that each column is dependent on a target variable. By using these assumptions, we can predict the missing values based on the corresponding target value.

The parameters used in MICE are 'method' and 'm'. 'Method' refers to the imputation method used for each column, for which we used a random forest. 'm' refers to the number of imputed data sets we will create which has a default value of 5. After creating these imputed datasets, we used the complete() function to select the most appropriate one for our response variable. The result is a statistically robust dataset that we can apply our models to.

2.4 Collinearity

Following imputation, we can now investigate collinearity. Collinearity in the model increases the variance of the estimators, meaning the model becomes less adequate. In order to remove collinearity in the model, we can investigate the correlation between variables and their Variance Inflation Factor scores. We start by making a model with all the variables in the dataset and plotting the correlation of each variable as seen in figure 3.

We can see many of the variables are correlated with one another, which may incentivise us to remove some variables from our model to improve its effectiveness. To decide which variables to remove, we use the vif() function in R. This reports the Variance Inflation Factor scores of each variable. The description of this function can be found in the R documentation:

'The VIF of a predictor is a measure for how easily it is predicted from a linear regression using the other predictors. A general guideline is that a VIF larger than 5 or 10 is large, indicating that the model has problems estimating the coefficient. If the VIF is larger than $\frac{1}{1-R^2}$, where R^2

is the Multiple R-squared of the regression, then that predictor is more related to the other predictors than it is to the response.’ – (8).

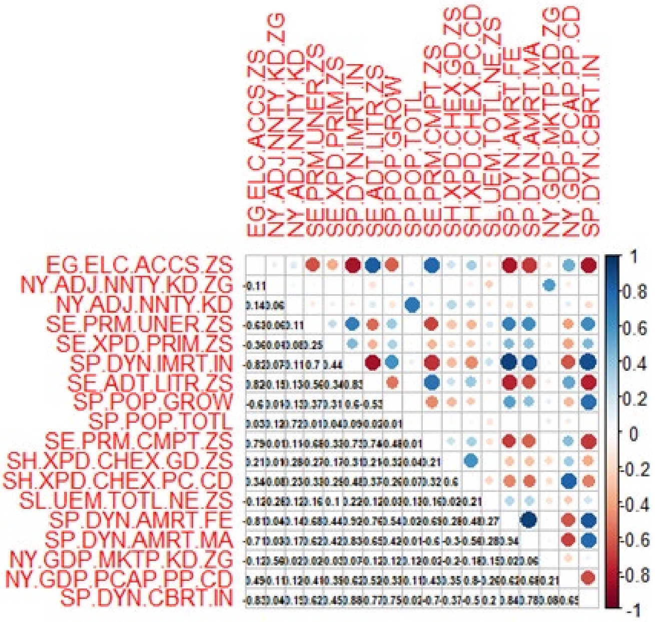


Figure 3. Correlation between variables

The result shows variables such as ‘Mortality rate, adult, female’, ‘Mortality rate, adult, male’, ‘Mortality rate, infant’, and ‘Birth rate, crude’ have a vif above 10, meaning these are the variables that we should investigate. The multiple R^2 value of the model with all variables is 0.9652, meaning $\frac{1}{1-R^2} = 28.736$. The vif of ‘Mortality rate, adult, female’ is close to this at 23.527, meaning it is possibly more correlated with other predictor variables than the response variable. We attempted to remove just this variable from the model first to see how it affected the other vif scores in case this one variable was the sole cause of collinearity within the model. The resulting “model2” has 7 significant variables, and a high adjusted R^2 score of 0.9591. However, checking the vif scores of the remaining variables shows ‘Birth rate, crude’ still has a score above 10 despite removing ‘Mortality rate, adult, female’. This means collinearity is still present within the model, so we create a new model “model3” with both these variables excluded. This new model has 8 significant variables, and an adjusted R^2 score of 0.9575. Checking vif scores again we can see ‘Mortality rate, infant’ still has a score of 9.155, so we experimented with removing this from the model and then comparing this model against model3 in an ANOVA test to check which is more appropriate. The resulting model4 has an adjusted R^2 score of 0.9369 and 11 significant variables which is a sizeable improvement from the others. In addition, when checking the vif scores of the remaining variables none report a score above 6 meaning collinearity has been removed from the data.

To confirm that model4 is a better model than model3 or model2, we completed an ANOVA test. Doing so for model2 and model3 revealed that we can reject the null hypothesis that is model2 with 99.9% certainty. However, performing two tests comparing model2 and model3 against model4 shows we can reject both with absolute certainty meaning model4 is the best model for this data. The result is a smaller subset of data with no collinearity within it, which will improve the performance of future models.

3. Model Analysis

In this section, we implement the following three machine learning models to predict life expectancy in 2018:

- Random forest regressor
- Multiple linear regressors
- Support vector regressor

We have chosen to use a random forest regression model because of its extremely high accuracy rate, and its ease of use. We have done a preliminary analysis of our data, and it is evident that there are outliers which this model can effectively handle.

Multiple linear regressors is another model proficient in identifying outliers or anomalies in a dataset. This model also works well in determining relevant relationships between predictor variables and the response variable. Thus, producing a relatively high accuracy rate.

Finally, we test the support vector regressor on our life expectancy dataset because SVR acknowledges non-linearity in the data and provides a proficient prediction model.

To test these models, we split our dataset into 70% train set, which we called 'data_train' and 30% test set, which we called 'data_test'. We fit these models individually on our train set and predicted life expectancy on the test set. We created a plot for each model for actual versus predicted life expectancy as displayed in appendix 5.

To evaluate the performance of each of our models, we use four evaluation metrics:

- Mean squared error (MSE)
- Mean absolute error (MAE)
- Root mean squared error (RMSE)

- R-squared value (R^2)

Mean squared error is the simplest and most common evaluation metric used to measure loss in regression models. This metrics is more sensitive to outliers as compared to the mean absolute error. It is, therefore, efficient in ensuring the trained model has no outlier predictions with huge errors.

The mean absolute error is the average of the absolute differences between the actual value and the model's predicted value. We chose to use this model for evaluation because it measures how far the predictions are from the actual output. Therefore, a bigger MAE will mean a more critical error. The random forest regression model had the smallest MAE score from all the models we tested.

While the root mean squared error measures the average magnitude of the error. This evaluation metrics is useful in our evaluation because it works great with large errors. For this metrics, a smaller RMSE equals a more accurate model.

With the R-squared value, the variance in the output variable is explained. A higher R-squared value means more variance explained by the dependent variables. It also attests to a better model. Our regression model gave an R-squared score of 0.951, meaning that the input variables explain 96% variance in the output variable.

The table in Table 2 shows the summary of these metrics that our models produce. From this, it is evident that the random forest regressor was the most effective at predicting life expectancy, and the support vector regressor did the worst at predicting life expectancy.

	Regression models		
Evaluation metrices	**Random Forest Regressor**	**Multiple linear Regressor**	**Support Vector Regressor**
Mean squared error	2.245	3.882	6.276
Mean absolute error	1.1553	1.422	1.669
Root mean squared error	1.498	1.9704	2.5052
R-squared value	0.951	0.916	0.816

Table 2: Summary of regression models performance

After implementing these models, we conclude the random forest regressor model is the most efficient at predicting life expectancy. This model is selected because of its out-of-bag error estimate, which means that cross-validation is estimated internally during the run, and the best output is produced. This model also gives us the most accurate prediction, and it is easily interpretable. Finally, this model clearly shows its prediction process with a **93.61%** variance explained, which measures how well the out-of-bag prediction explains the training set's target variance of the training set. The summary of the random forest regressor is shown in Table 3.

Prediction	**Actual**
Min. :56.26	**Min**. :52.80
1st Qu.:67.45	**1st Qu**.:67.77
Median:73.77	**Median:**72.73
Mean:72.09	**Mean:**71.74
3rd Qu.:77.41	**3rd Qu**.:77.12
Max. :82.50	**Max**. :83.35

Table 3: Summary of Random Forest regressor model Prediction Versus Actual

To utilise the predictive power of the trained and tested random forest regressor model, the pre-processing steps taken in building the model had to be applied to the LifeExpectancyData2 dataset.

On inspection the second dataset had missing values meaning the MICE package would have to be used to impute the values. However, due to the small number of observations within the second dataset it was unable to build the random forest model needed to derive the imputation values. In order to get the imputation to work we merged the LifeExpectancyData2 and the original LifeExpectancyData1 datasets which provided enough observations for the MICE package to impute the values. Once the values were imputed the set of observations from the second dataset were extracted and used with the chosen model to predict life expectancy. The results from the model's predictions can be found in the attached csv file [Results for predicting life expectancy]. Whilst we do not have ground truth labels to compare against the predictions, an initial look suggests they are within a reasonable range.

4. ANOVA

One-way ANOVA can be used to determine statistically significant differences between the means of certain groups – in this case, we investigate whether 1) Is ANOVA appropriate for this data set and 2) What conclusions can we gain by grouping by continent and carrying out one way ANOVA.

First, we group countries together based on their continent using the Country Code package. This creates six distinct continent groups within our data; Africa, Asia, Europe, North America, South America and Oceania. The ANOVA we employ tests for statistically significant differences between the means of all six groups. We define the null hypothesis as:

$$H_0 : \mu_{Africa} = \mu_{Asia} = \mu_{Europe} = \mu_{NorthAmerica} = \mu_{SouthAmerica} = \mu_{Oceania}$$

In the case where ANOVA returns a statistically significant result, we accept the alternative hypothesis that at least two group means are statistically significantly different.

4.1. Is it appropriate?

The one-way ANOVA primarily relies on three main assumptions:

1. Data is normally distributed
    a. A Shapiro test reveals that this assumption is not met. However, the one-way ANOVA can tolerate this assumption being broken
2. Homogeneity of variance
    a. A Bartlett's test for homogeneity of variances confirms this assumption is met (Bartlett's K-squared = 19.343, df = 5, p-value = 0.001659)
3. Observations are independent of each other
    a. The strictest requirement, met by nature of measurements being independent from one another

We conclude that a one-way ANOVA is appropriate in this case, but we could suffer from some type 1 error due to the violation of assumption 1 (rejecting the null hypothesis when we should have accepted), although this is greatly reduced as all groups are well populated.

4.2. Results

The result of the one-way ANOVA is displayed in appendix 4. We get a resulting P value that is statistically significant beyond even the 0.001 level, so conclude that there are significant differences between all the groups. However, this does not inform us which continents have the most significant differences (or differences at all), so we use a post hoc test known as Tukey Honest Significant Differences (HSD) to test for the pairwise comparisons. We report the difference between means and the P value adjusted for pairwise comparisons in table 4. Results are sorted by their significance level, with Africa showing the highest levels of significance from other groups. This is further evidenced by figure 5.

	Difference	P Adjusted
Asia-Africa*	10.13709	0
Europe-Africa*	15.00023	0
North America-Africa*	15.78456	0
Oceania-Africa*	10.00378	0
South America-Africa*	10.99605	0
Europe-Asia*	4.863137	0.000095
South America-Europe*	-4.00417	0.006235
Oceania-Europe*	-4.99645	0.023921
North America-Asia	5.647473	0.228431
Oceania-North America	-5.78079	0.310691
South America-North America	-4.78851	0.425343
South America-Asia	0.858962	0.970691
South America-Oceania	0.992276	0.990256
North America-Europe	0.784336	0.99962
Oceania-Asia	-0.13331	0.999999
***Statistically significant at the 0.05 level**		

Table 4: Difference between means and P value adjusted for pairwise comparison

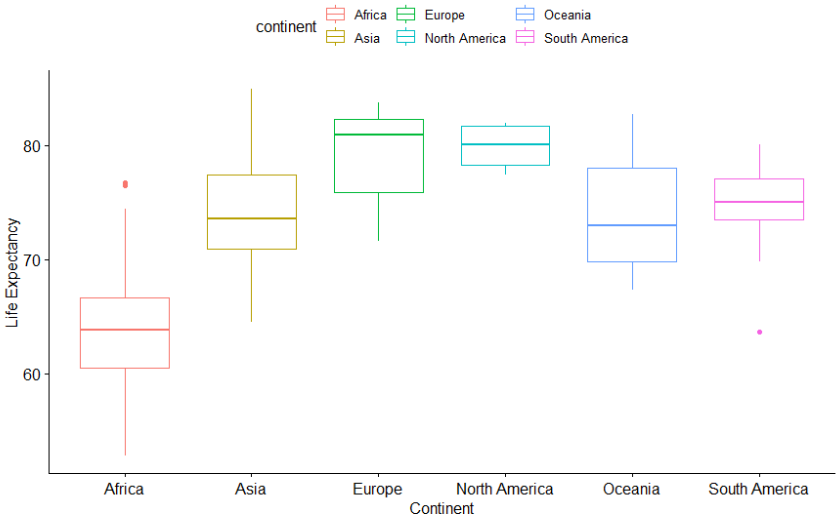


Figure 5: Box plots of life expectancy by continent

6. Conclusion

In conclusion, we have outlined effective methods to deal with large amounts of missing values and collinearity within the data. Following this we have implemented effective predictive techniques and have shown that a random forest regression model is the most appropriate model for this data. Finally, we have made what we believe are accurate predictions on a holdout test set and have performed an ANOVA test to determine statistically significant differences between the means of the life expectancy of different countries.

References

1. Life Expectancy - Our World in Data [Internet]. [cited 2021 Mar 25]. Available from: https://ourworldindata.org/life-expectancy#citation

2. Jen MH, Johnston R, Jones K, Harris R, Gandy A. International variations in life expectancy: A spatio-temporal analysis. Tijdschrift voor Economische en Sociale Geografie. 2010 Feb;101(1):73–90.

3. Wilkinson RG. Income distribution and life expectancy [Internet]. Vol. 304, British Medical Journal. BMJ Publishing Group; 1992 [cited 2021 Mar 25]. p. 165–8. Available from: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1881178/

4. Crémieux PY, Ouellette P, Pilon C. Health care spending as determinants of health outcomes. Health Economics. 1999 Nov;8(7):627–39.

5. Rogot E, Paul Sorlie MD, Johnson NJ. Life Expectancy by Employment Status, Income, and Education in the National Longitudinal Mortality Study [Internet]. ncbi.nlm.nih.gov. [cited 2021 Mar 25]. Available from: https://www.ncbi.nlm.nih.gov/pmc/articles/pmc1403677/

6. Wolfson MC. Health-adjusted life expectancy. Health reports / Statistics Canada, Canadian Centre for Health Information = Rapports sur la santé / Statistique Canada, Centre canadien d’information sur la santé [Internet]. 1996 [cited 2021 Mar 25];8(1). Available from: https://pubmed.ncbi.nlm.nih.gov/8844180/

7. MICE method R documentation. Available from: https://www.rdocumentation.org/packages/mice/versions/3.13.0/topics/mice

8. VIF function R documentation. Available from: https://www.rdocumentation.org/packages/regclass/versions/1.6/topics/VIF

Appendix

Group Contributions:

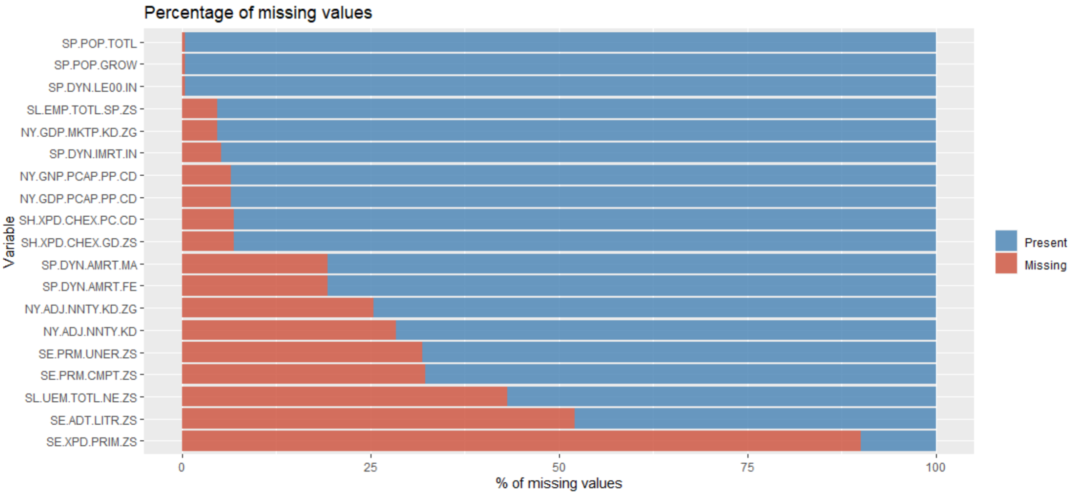
Group Member	Work completed
Carter Gibson	Q5, compiling all code into one document and testing
Laura Osede	Q4a/b
Jake Teague	Q3, compiling final report, abstract, conclusion
Gurleen Oberoi	Q2
Ryan O'Missenden	Q4c, PowerPoint formatting, compiling of final report
Stephanie Ruscillo	Q1, introduction

Appendix 1

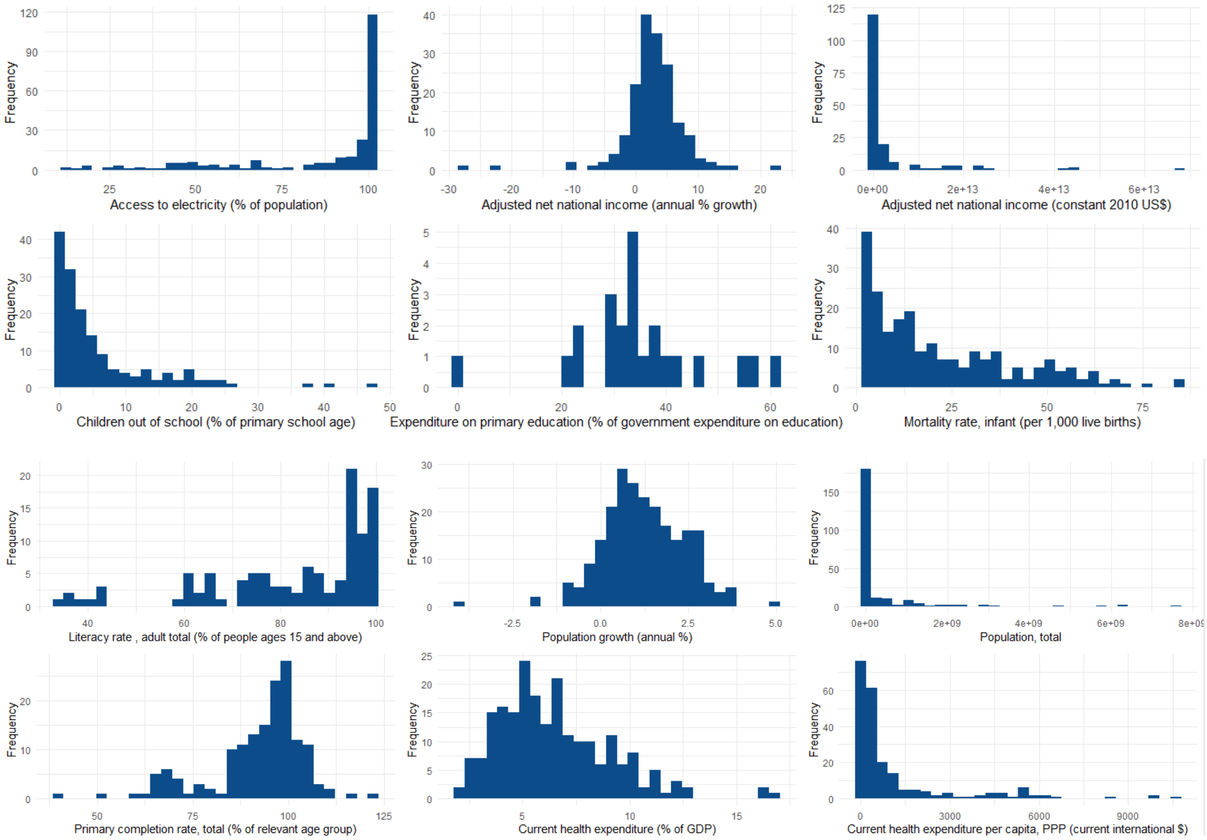
Variable	n	mean	sd	median	min	max	range	skew	kurtosis	Standard Error	Shapiro Wilk p-value	Na's	outliers
Access to electricity (% of population)	232	85.43	23.71	99.92	90.41	0.12	11.02	100.00	88.98	-1.53	2.20E-16	0	0
Adjusted net national income (annual % growth)	173	2.78	5.00	2.74	2.92	3.12	-28.10	22.08	50.18	-1.67	3.89E-13	59	12
Adjusted net national income (constant 2010 US$)	166	3.31E+12	8.99E+12	1.74E+11	4.01E+08	6.77E+13	6.77E+13	4.19	20.65	6.98E+11	2.20E-16	66	27
Children out of school (% of primary school age	158	6.03	8.12	2.84	0.00	47.35	47.35	2.29	6.31	0.65	3.77E-16	74	15
Expenditure on primary education (% of government expenditure on education	23	34.76	12.72	34.04	0.66	61.65	60.99	-0.11	0.90	2.65	0.1071	209	4
Mortality rate, infant (per 1,000 live births)	220	22.23	19.31	14.80	1.60	83.40	81.80	1.01	0.15	1.30	2.44E-12	12	2
Literacy rate, adult total (% of people ages 15 and above)	111	83.34	17.23	91.29	34.52	99.99	65.47	-1.23	0.73	1.64	9.07E-10	121	4
Population growth (annual %)	231	1.24	1.15	1.16	-4.05	4.92	8.97	-0.19	1.37	0.08	0.001976	1	2
Population, total	231	3.49E+08	1.03E+09	11485048	37910	7.592E+09	7.592E+09	4.6	23.58	67592264	2.20E-16	1	47
Primary completion rate, total (% of relevant age group)	157	92.12	12.62	95.79	40.56	123.00	82.44	-1.22	1.80	1.01	3.12E-09	75	17
Current health expenditure (% of GDP)	216	6.48	2.68	5.99	2.14	16.89	14.75	1.01	1.38	0.18	9.49E-08	16	3
Current health expenditure per capita, PPP (current international $)	216	1134.97	1924.86	350.07	18.51	10623.85	10605.34	2.59	6.97	130.97	2.20E-16	16	31
Unemployment, total (% of total labor force) (national estimate)	132	7.02	4.96	5.33	0.11	29.42	29.31	1.93	4.47	0.43	1.05E-11	100	12
Mortality rate, adult, female (per 1,000 female adults)	187	134.05	76.54	115.17	31.87	419.36	387.49	1.02	0.72	5.60	8.59E-09	45	3

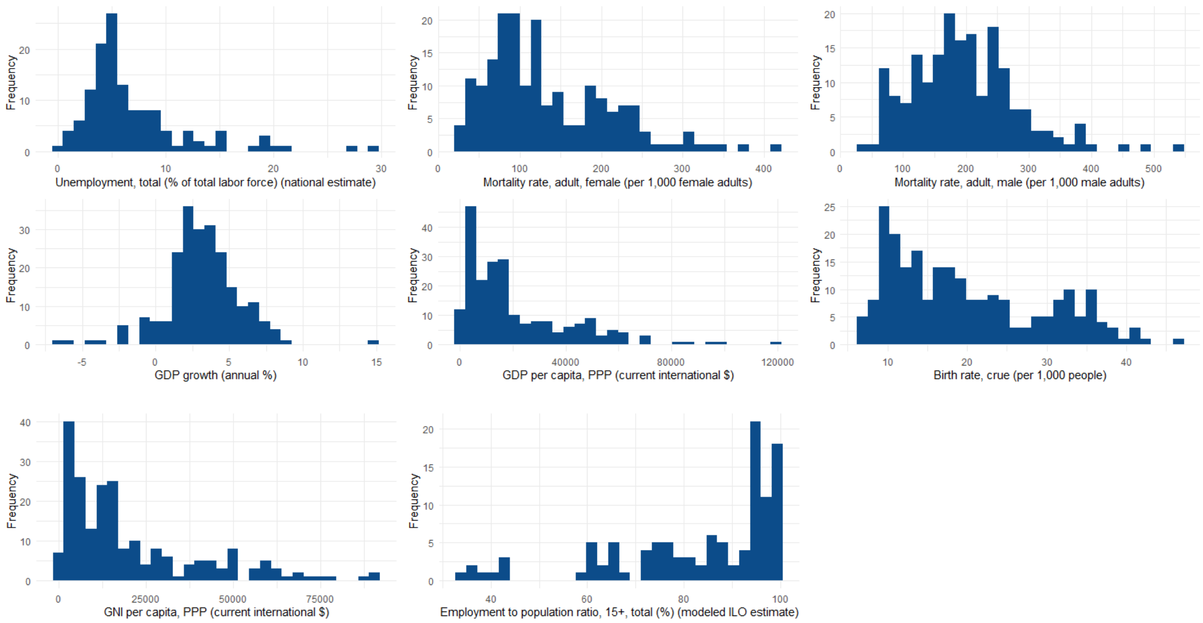
Mortality rate, adult, male (per 1,000 male adults)	187	197.23	85.23	188.41	41.50	545.68	504.18	0.81	1.31	6.23	4.53E-05	45	3
GDP growth (annual %)	221	3.19	2.55	3.12	-6.36	15.13	21.49	-0.14	3.12	0.17	1.74E-06	11	12
GDP per capita, PPP (current international $)	217	21259.20	21126.72	14208.07	780.0749	120325.92	119545.85	1.620793	2.890251	1434.175	5.80E-15	15	8
Birth rate, crude (per 1,000 people)	232	19.88	9.48	17.70	6.40	46.08	39.68	0.63	-0.76	0.62	9.32E-10	0	0
GNI per capita, PPP (current international $)	217	20616.75	20063.89	13960.00	780.00	91280.00	90500.00	1.37	1.23	1362.03	1.38E-14	15	9
Employment to population ratio, 15+, total (%)	221	58.03	10.45	58.42	33.17	86.61	53.44	0.09	0.017	0.70	0.01199	11	4

Appendix 2

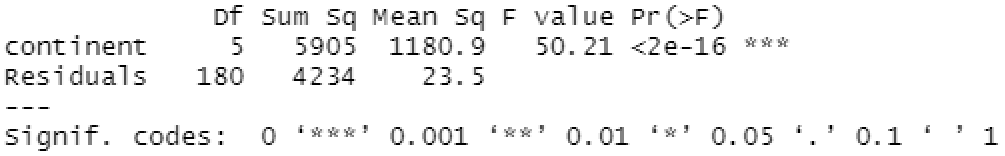


Appendix 3



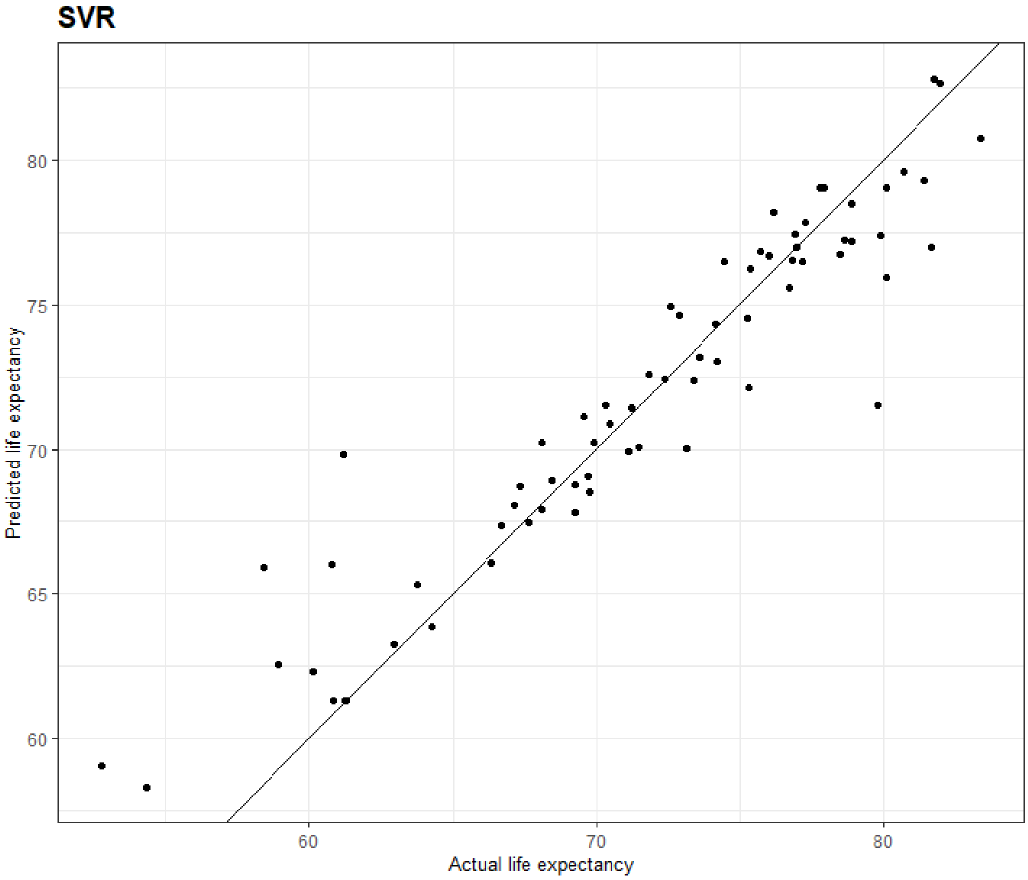


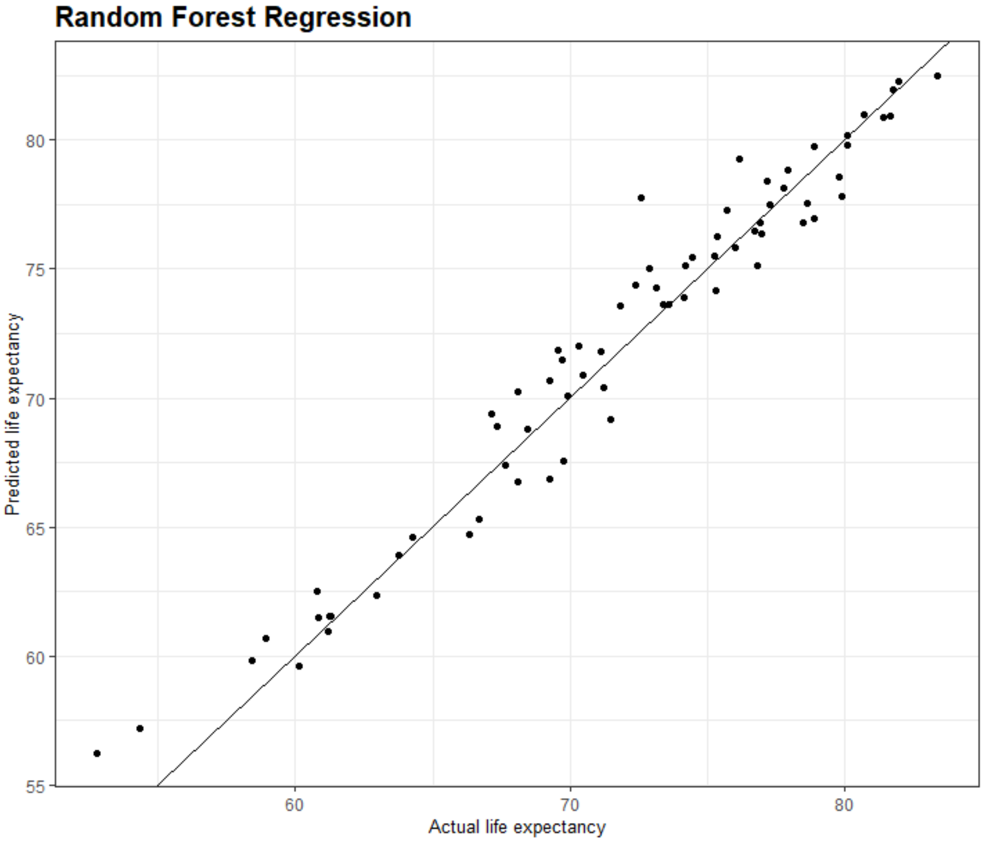
Appendix 4

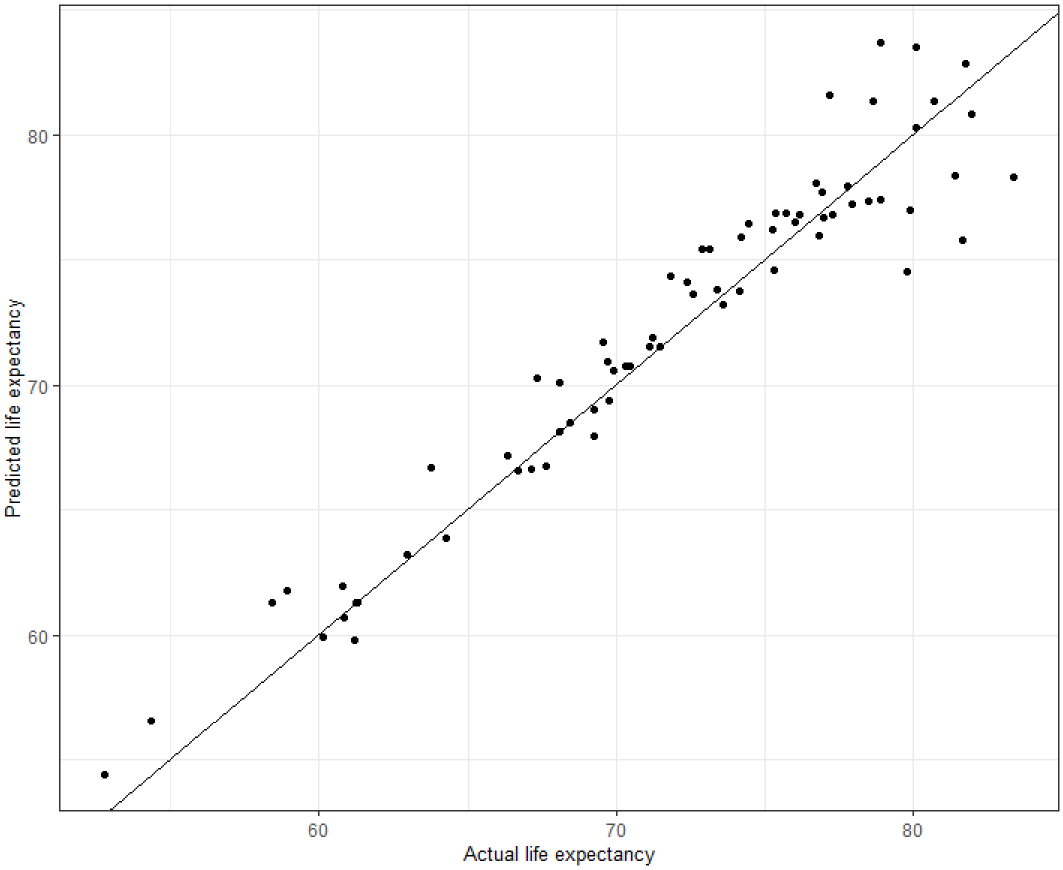


```
              Df Sum Sq Mean Sq F value Pr(>F)    
continent      5   5905  1180.9   50.21 <2e-16 ***
Residuals    180   4234    23.5                   
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Appendix 5: Plots showing actual versus predicted for different machine learning algorithms.







Multiple Linear Regression

```
######################### LOADING LIBRARIES #########################

library(tidyverse)
library(mice)
library(VIM)
library("faraway")
library(mice)
library(corrplot)
library(dplyr)    # for data wrangling
library(ggplot2)  # for awesome graphics
library(fields)
library(ranger)   # a c++ implementation of random forest
library(h2o)      # a java-based implementation of random forest
library(randomForest) # basic implementation
library(caret)        # an aggregator package for performing many machine learning models
library(rsample)      # data splitting
library(tidyverse)
library(tidymodels)
library(e1071)
library(corrplot)
library(caTools)
#library(extrafont)
library(Metrics)
library('writexl')
library('dplyr')
library('readr')
library('olsrr')
library('leaps')
library('ggplot2')
library('ggpubr') #
library('gridExtra') #create grid of graphs
library('data.table')
library("Hmisc") #correlation matrix and reshaping
library("xlsx")

######################### LOADING DATA #########################

data <- read.csv(file = 'LifeExpectancyData1.csv', stringsAsFactors = TRUE)
descriptive_cols = subset(data, select = c(Country.Code, Country.Name))
data = subset(data, select = -c(Country.Code, Country.Name))

######################### EDA (Q1) #########################

dim(data)

str(data)

head(data)

colSums(sapply(data, is.na))

summary(data)
sapply(data, mean, na.rm=TRUE)
sapply(data, median, na.rm=TRUE)
sapply(data, sd, na.rm=TRUE)

length(data$SP.DYN.AMRT.FE)

## Missing Values
missing.values <- data %>%
  gather(key = "key", value = "val") %>%
  mutate(isna = is.na(val)) %>%
  group_by(key) %>%
  mutate(total = n()) %>%
  group_by(key, total, isna) %>%
  summarise(num.isna = n()) %>%
  mutate(pct = num.isna / total * 100)

levels <-
```

```
  (missing.values  %>% filter(isna == T) %>% arrange(desc(pct)))$key

percentage.plot <- missing.values %>%
  ggplot() +
  geom_bar(aes(x = reorder(key, desc(pct)),
               y = pct, fill=isna),
           stat = 'identity', alpha=0.8) +
  scale_x_discrete(limits = levels) +
  scale_fill_manual(name = "",
                    values = c('steelblue', 'tomato3'), labels = c("Present", "Missing")) +
  coord_flip() +
  labs(title = "Percentage of missing values", x =
         'Variable', y = "% of missing values")

percentage.plot

row.plot <- data %>%
  mutate(id = row_number()) %>%
  gather(-id, key = "key", value = "val") %>%
  mutate(isna = is.na(val)) %>%
  ggplot(aes(key, id, fill = isna)) +
  geom_raster(alpha=0.8) +
  scale_fill_manual(name = "",
                    values = c('steelblue', 'tomato3'),
                    labels = c("Present", "Missing")) +
  scale_x_discrete(limits = levels) +
  labs(x = "Variable",
       y = "Row Number", title = "Missing values in rows") +
  coord_flip()

row.plot

sapply(data, mean)
sapply(data, median)

## Correlation

correlation <- rcorr(as.matrix(data))

# Reshape matrix
f_matrix <- function(cormat, pmat) {
  ut <- upper.tri(cormat)
  data.frame(
    row = rownames(cormat)[row(cormat)[ut]],
    column = rownames(cormat)[col(cormat)[ut]],
    cor  =(cormat)[ut],
    p = pmat[ut]
  )
}
f <- f_matrix(correlation$r, correlation$P)
write_xlsx(as.data.frame(f), path = "correlation matrix.xlsx")

## Shapiro Test for Normality

shapiro.test(data$EG.ELC.ACCS.ZS)
shapiro.test(data$NY.ADJ.NNTY.KD.ZG)
shapiro.test(data$NY.ADJ.NNTY.KD)
shapiro.test(data$SE.PRM.UNER.ZS)
shapiro.test(data$SE.XPD.PRIM.ZS)
shapiro.test(data$SP.DYN.IMRT.IN)
shapiro.test(data$SE.ADT.LITR.ZS)
shapiro.test(data$SP.POP.GROW)
shapiro.test(data$SP.POP.TOTL)
shapiro.test(data$SE.PRM.CMPT.ZS)
shapiro.test(data$SH.XPD.CHEX.GD.ZS)
shapiro.test(data$SH.XPD.CHEX.PC.CD)
shapiro.test(data$SL.UEM.TOTL.NE.ZS)
shapiro.test(data$SP.DYN.AMRT.FE)
shapiro.test(data$SP.DYN.AMRT.MA)
shapiro.test(data$NY.GDP.MKTP.KD.ZG)
shapiro.test(data$NY.GDP.PCAP.PP.CD)
shapiro.test(data$SP.DYN.CBRT.IN)
shapiro.test(data$NY.GNP.PCAP.PP.CD)
shapiro.test(data$SL.EMP.TOTL.SP.ZS)

describe <- describe(cbind(
  data$EG.ELC.ACCS.ZS,
  data$NY.ADJ.NNTY.KD.ZG,
  data$NY.ADJ.NNTY.KD,
  data$SE.PRM.UNER.ZS,
  data$SE.XPD.PRIM.ZS,
  data$SP.DYN.IMRT.IN,
  data$SE.ADT.LITR.ZS,
  data$SP.POP.GROW,
  data$SP.POP.TOTL,
  data$SE.PRM.CMPT.ZS,
  data$SH.XPD.CHEX.GD.ZS,
  data$SH.XPD.CHEX.PC.CD,
  data$SL.UEM.TOTL.NE.ZS,
  data$SP.DYN.AMRT.FE,
  data$SP.DYN.AMRT.MA,
  data$NY.GDP.MKTP.KD.ZG,
  data$NY.GDP.PCAP.PP.CD,
  data$SP.DYN.CBRT.IN,
  data$NY.GNP.PCAP.PP.CD,
  data$SL.EMP.TOTL.SP.ZS))
write_xlsx(as.data.frame(describe), path = "/Users/User/describe.xlsx")

## Scatter Plots
v1 <- ggscatter(data, x = "EG.ELC.ACCS.ZS", y = "SP.DYN.LE00.IN",
          add = "reg.line", conf.int = TRUE,
          cor.coef = TRUE, cor.method = "pearson",
          xlab = "Access to electricity (% of population)", ylab = "Life expectancy at birth total (years)")
v2 <- ggscatter(data, x = "NY.ADJ.NNTY.KD.ZG", y = "SP.DYN.LE00.IN",
          add = "reg.line", conf.int = TRUE,
          cor.coef = TRUE, cor.method = "pearson",
          xlab = "Adjusted net national income (annual % growth)", ylab = "Life expectancy at birth total
(years)")
v3 <- ggscatter(data, x = "NY.ADJ.NNTY.KD", y = "SP.DYN.LE00.IN",
          add = "reg.line", conf.int = TRUE,
          cor.coef = TRUE, cor.method = "pearson",
          xlab = "Adjusted net national income (constant 2010 US$)", ylab = "Life expectancy at birth total
(years)")
v4 <- ggscatter(data, x = "SE.PRM.UNER.ZS", y = "SP.DYN.LE00.IN",
          add = "reg.line", conf.int = TRUE,
          cor.coef = TRUE, cor.method = "pearson",
          xlab = "Children out of school (% of primary school age)", ylab = "Life expectancy at birth total
(years)")
v5 <- ggscatter(data, x = "SE.XPD.PRIM.ZS", y = "SP.DYN.LE00.IN",
          add = "reg.line", conf.int = TRUE,
          cor.coef = TRUE, cor.method = "pearson",
          xlab = "Expenditure on primary education (% of government expenditure on education)", ylab = "Life
expectancy at birth total (years)")
v6 <- ggscatter(data, x = "SP.DYN.IMRT.IN", y = "SP.DYN.LE00.IN",
          add = "reg.line", conf.int = TRUE,
          cor.coef = TRUE, cor.method = "pearson",
          xlab = "Mortality rate, infant (per 1,000 live births)", ylab = "Life expectancy at birth total (years)")
v7 <- ggscatter(data, x = "SE.ADT.LITR.ZS", y = "SP.DYN.LE00.IN",
          add = "reg.line", conf.int = TRUE,
          cor.coef = TRUE, cor.method = "pearson",
          xlab = "Literacy rate , adult total (% of people ages 15 and above)", ylab = "Life expectancy at birth
total (years)")
v8 <- ggscatter(data, x = "SP.POP.GROW", y = "SP.DYN.LE00.IN",
          add = "reg.line", conf.int = TRUE,
          cor.coef = TRUE, cor.method = "pearson",
          xlab = "Population growth (annual %)", ylab = "Life expectancy at birth total (years)")
v9 <- ggscatter(data, x = "SP.POP.TOTL", y = "SP.DYN.LE00.IN",
          add = "reg.line", conf.int = TRUE,
           cor.coef = TRUE, cor.method = "pearson",
           xlab = "Population, total", ylab = "Life expectancy at birth total (years)")
v10 <- ggscatter(data, x = "SE.PRM.CMPT.ZS", y = "SP.DYN.LE00.IN",
            add = "reg.line", conf.int = TRUE,
            cor.coef = TRUE, cor.method = "pearson",
            xlab = "Primary completion rate, total (% of relevant age group)", ylab = "Life expectancy at birth
total (years)")
v11 <- ggscatter(data, x = "SH.XPD.CHEX.GD.ZS", y = "SP.DYN.LE00.IN",
            add = "reg.line", conf.int = TRUE,
            cor.coef = TRUE, cor.method = "pearson",
            xlab = "Current health expenditure (% of GDP)", ylab = "Life expectancy at birth total (years)")
v12 <- ggscatter(data, x = "SH.XPD.CHEX.PC.CD", y = "SP.DYN.LE00.IN",
            add = "reg.line", conf.int = TRUE,
            cor.coef = TRUE, cor.method = "pearson",
            xlab = "Current health expenditure per capita, PPP (current international $)", ylab = "Life expectancy
at birth total (years)")
v13 <- ggscatter(data, x = "SL.UEM.TOTL.NE.ZS", y = "SP.DYN.LE00.IN",
            add = "reg.line", conf.int = TRUE,
            cor.coef = TRUE, cor.method = "pearson",
            xlab = "Unemployment, total (% of total labor force) (national estimate)", ylab = "Life expectancy at
birth total (years)")
v14 <- ggscatter(data, x = "SP.DYN.AMRT.FE", y = "SP.DYN.LE00.IN",
            add = "reg.line", conf.int = TRUE,
            cor.coef = TRUE, cor.method = "pearson",
            xlab = "Mortality rate, adult, female (per 1,000 female adults)", ylab = "Life expectancy at birth total
(years)")
v15 <- ggscatter(data, x = "SP.DYN.AMRT.MA", y = "SP.DYN.LE00.IN",
            add = "reg.line", conf.int = TRUE,
            cor.coef = TRUE, cor.method = "pearson",
            xlab = "Mortality rate, adult, male (per 1,000 male adults)", ylab = "Life expectancy at birth total
(years)")
v16 <- ggscatter(data, x = "NY.GDP.MKTP.KD.ZG", y = "SP.DYN.LE00.IN",
            add = "reg.line", conf.int = TRUE,
            cor.coef = TRUE, cor.method = "pearson",
            xlab = "GDP growth (annual %)", ylab = "Life expectancy at birth total (years)")
v17 <- ggscatter(data, x = "NY.GDP.PCAP.PP.CD", y = "SP.DYN.LE00.IN",
            add = "reg.line", conf.int = TRUE,
            cor.coef = TRUE, cor.method = "pearson",
            xlab = "GDP per capita, PPP (current international $)", ylab = "Life expectancy at birth total (years)")
v18 <- ggscatter(data, x = "SP.DYN.CBRT.IN", y = "SP.DYN.LE00.IN",
            add = "reg.line", conf.int = TRUE,
            cor.coef = TRUE, cor.method = "pearson",
            xlab = "Birth rate, crue (per 1,000 people)", ylab = "Life expectancy at birth total (years)")
v19 <- ggscatter(data, x = "NY.GNP.PCAP.PP.CD", y = "SP.DYN.LE00.IN",
            add = "reg.line", conf.int = TRUE,
            cor.coef = TRUE, cor.method = "pearson",
            xlab = "GNI per capita, PPP (current international $)", ylab = "Life expectancy at birth total (years)")
v20 <- ggscatter(data, x = "SL.EMP.TOTL.SP.ZS", y = "SP.DYN.LE00.IN",
            add = "reg.line", conf.int = TRUE,
            cor.coef = TRUE, cor.method = "pearson",
            xlab = "Employment to population ratio, 15+, total (%) (modeled ILO estimate)", ylab = "Life
expectancy at birth total (years)")
vlog12 <- ggscatter(data, x = "log_SH.XPD.CHEX.PC.CD", y = "SP.DYN.LE00.IN",
              add = "reg.line", conf.int = TRUE,
              cor.coef = TRUE, cor.method = "pearson",
              xlab = "Current health expenditure per capita, PPP (current international $)", ylab = "Life
expectancy at birth total (years)")
vlog17 <- ggscatter(data, x = "log_NY.GDP.PCAP.PP.CD", y = "SP.DYN.LE00.IN",
              add = "reg.line", conf.int = TRUE,
              cor.coef = TRUE, cor.method = "pearson",
              xlab = "GDP per capita, PPP (current international $)", ylab = "Life expectancy at birth total
(years)")
vlog19 <- ggscatter(data, x = "log_NY.GNP.PCAP.PP.CD", y = "SP.DYN.LE00.IN",
              add = "reg.line", conf.int = TRUE,
              cor.coef = TRUE, cor.method = "pearson",
              xlab = "GNI per capita, PPP (current international $)", ylab = "Life expectancy at birth total
(years)")
vlog3 <- ggscatter(data, x = "log_NY.ADJ.NNTY.KD", y = "SP.DYN.LE00.IN",
             add = "reg.line", conf.int = TRUE,
             cor.coef = TRUE, cor.method = "pearson",
             xlab = "Adjusted net national income (constant 2010 US$)", ylab = "Life expectancy at birth total
(years)")


#arrange scatterplots with high correlation and save
high_corr <- grid.arrange(v1, v6, v7, v14, v15, v18, ncol = 3)
ggsave(file="MA317_scatter_high_corr.jpg", high_corr)


## Histograms create histograms
hy <- ggplot(data, aes(SP.DYN.LE00.IN)) + geom_histogram() + xlab("Life expectancy at birth total (years)") +
ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()
h1 <- ggplot(data, aes(EG.ELC.ACCS.ZS)) + geom_histogram() + xlab ("Access to electricity (%
of population)") + ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()
h2 <- ggplot(data, aes(NY.ADJ.NNTY.KD.ZG)) + geom_histogram() + xlab ("Adjusted net national income
(annual % growth)")+ ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()
h3 <- ggplot(data, aes(NY.ADJ.NNTY.KD)) + geom_histogram() + xlab ("Adjusted net national income
(constant 2010 US$)")+ ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()
h4 <- ggplot(data, aes(SE.PRM.UNER.ZS)) + geom_histogram() + xlab ("Children out of school (% of primary
school age)")+ ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()
h5 <- ggplot(data, aes(SE.XPD.PRIM.ZS)) + geom_histogram() + xlab ("Expenditure on primary education (%
of government expenditure on education)")+ ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()
h6 <- ggplot(data, aes(SP.DYN.IMRT.IN)) + geom_histogram() + xlab("Mortality rate, infant (per 1,000 live
births)")+ ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()
h7 <- ggplot(data, aes(SE.ADT.LITR.ZS)) + geom_histogram() + xlab("Literacy rate , adult total (% of people
ages 15 and above)")+ ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()
h8 <- ggplot(data, aes(SP.POP.GROW)) + geom_histogram() + xlab("Population growth (annual %)")+
ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()
h9 <- ggplot(data, aes(SP.POP.TOTL)) + geom_histogram() + xlab("Population, total")+ ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()
h10 <- ggplot(data, aes(SE.PRM.CMPT.ZS)) + geom_histogram() + xlab("Primary completion rate, total (%
of relevant age group)")+ ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()
h11 <- ggplot(data, aes(SH.XPD.CHEX.GD.ZS)) + geom_histogram() + xlab("Current health expenditure (%
of GDP)")+ ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()
h12 <- ggplot(data, aes(SH.XPD.CHEX.PC.CD)) + geom_histogram() + xlab("Current health expenditure per
capita, PPP (current international $)")+ ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()
h13 <- ggplot(data, aes(SL.UEM.TOTL.NE.ZS)) + geom_histogram() + xlab("Unemployment, total (% of total
labor force) (national estimate)")+ ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()
h14 <- ggplot(data, aes(SP.DYN.AMRT.FE)) + geom_histogram() + xlab("Mortality rate, adult, female (per
1,000 female adults)")+ ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()
h15 <- ggplot(data, aes(SP.DYN.AMRT.MA)) + geom_histogram() + xlab("Mortality rate, adult, male (per
1,000 male adults)")+ ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()
h16 <- ggplot(data, aes(NY.GDP.MKTP.KD.ZG)) + geom_histogram() + xlab("GDP growth (annual %)")+
ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()
h17 <- ggplot(data, aes(NY.GDP.PCAP.PP.CD)) + geom_histogram()+ xlab("GDP per capita, PPP (current
international $)")+ ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()
h18 <- ggplot(data, aes(SP.DYN.CBRT.IN)) + geom_histogram()+ xlab("Birth rate, crue (per 1,000 people)")+
ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()
h19 <- ggplot(data, aes(NY.GNP.PCAP.PP.CD)) + geom_histogram()+ xlab("GNI per capita, PPP (current
international $)")+ ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()
h20 <- ggplot(data, aes(SE.ADT.LITR.ZS)) + geom_histogram() + xlab("Employment to population ratio, 15+,
total (%) (modeled ILO estimate")+ ylab("Frequency") +
  geom_histogram(fill="#0c4c8a") + theme_minimal()

hy
g <-grid.arrange(h1, h2, h3, h4, h5, h6,ncol=3)
g <-grid.arrange(h7, h8, h9, h10, h11, h12, ncol=3)
g <-grid.arrange(h13, h14, h15, h16, h17, h18, ncol=3)
g <-grid.arrange(h19, h20, ncol=3)
ggsave(file="MA317_histograms.jpg", g)



## Outliers
out1 <- boxplot.stats(data$EG.ELC.ACCS.ZSy)$out
out_ind1 <- which(data$EG.ELC.ACCS.ZS %in% c(out1))
length(out2)
out2 <- boxplot.stats(data$NY.ADJ.NNTY.KD.ZG)$out
out_ind2 <- which(data$NY.ADJ.NNTY.KD.ZG %in% c(out2))
length(out2)
out3 <- boxplot.stats(data$NY.ADJ.NNTY.KD)$out
out_ind3 <- which(data$NY.ADJ.NNTY.KD %in% c(out3))
length(out3)
out4 <- boxplot.stats(data$SE.PRM.UNER.ZS)$out
out_ind4 <- which(data$SE.PRM.UNER.ZS %in% c(out4))
length(out4)
out5 <- boxplot.stats(data$SE.XPD.PRIM.ZS)$out
out_ind5 <- which(data$SE.XPD.PRIM.ZS %in% c(out5))
length(out5)
out6 <- boxplot.stats(data$SP.DYN.IMRT.IN)$out
out_ind6 <- which(data$SP.DYN.IMRT.IN %in% c(out6))
length(out6)
out7 <- boxplot.stats(data$SE.ADT.LITR.ZS)$out
out_ind7 <- which(data$EG.ELC.ACCS.ZS %in% c(out7))
length(out7)
out8 <- boxplot.stats(data$SP.POP.GROW)$out
out_ind8 <- which(data$SP.POP.GROW %in% c(out8))
length(out8)
out9 <- boxplot.stats(data$SP.POP.TOTL)$out
out_ind9 <- which(data$SP.POP.TOTL %in% c(out9))
length(out9)
out10 <- boxplot.stats(data$SE.PRM.CMPT.ZS)$out
out_ind10 <- which(data$SE.PRM.CMPT.ZS%in% c(out10))
length(out10)
out11 <- boxplot.stats(data$SH.XPD.CHEX.GD.ZS)$out
out_ind11 <- which(data$SH.XPD.CHEX.GD.ZS %in% c(out11))
length(out11)
out12 <- boxplot.stats(data$SH.XPD.CHEX.PC.CD)$out
out_ind12 <- which(data$SH.XPD.CHEX.PC.CD %in% c(out12))
length(out12)
out13 <- boxplot.stats(data$SL.UEM.TOTL.NE.ZS)$out
out_ind13 <- which(data$SL.UEM.TOTL.NE.ZS %in% c(out13))
length(out13)
out14 <- boxplot.stats(data$SP.DYN.AMRT.FE)$out
out_ind14 <- which(data$SP.DYN.AMRT.FE %in% c(out14))
length(out14)
out15 <- boxplot.stats(data$SP.DYN.AMRT.MA)$out
out_ind15 <- which(data$SP.DYN.AMRT.MA %in% c(out15))
length(out15)
out16 <- boxplot.stats(data$NY.GDP.MKTP.KD.ZG)$out
out_ind16 <- which(data$NY.GDP.MKTP.KD.ZG %in% c(out16))
length(out16)
out17 <- boxplot.stats(data$NY.GDP.PCAP.PP.CD)$out
out_ind17 <- which(data$NY.GDP.PCAP.PP.CD %in% c(out17))
length(out17)
out18 <- boxplot.stats(data$SP.DYN.CBRT.IN)$out
out_ind18 <- which(data$SP.DYN.CBRT.IN %in% c(out18))
length(out18)
```

```
out19 <- boxplot.stats(data$NY.GNP.PCAP.PP.CD)$out
out_ind19 <- which(data$NY.GNP.PCAP.PP.CD %in% c(out19))
length(out19)
out20 <- boxplot.stats(data$SE.ADT.LITR.ZS)$out
out_ind20 <- which(data$SE.ADT.LITR.ZS %in% c(out20))
length(out20)

######################### PREPROCESSING (Q2) #########################

### Missing Values

#number of rows with complete data
sum(complete.cases(data))

#imputing missing values
imputations = mice(data,m = 10,method = "rf",seed = 22)

#summary of imputed values
summary(imputations)

complete_dataset = complete(imputations)

######################### COLLINEARITY ANALYSIS (Q3) #########################

# Model 1 -> All predictor variables
model1 <-lm(SP.DYN.LE00.IN ~ ., data=complete_dataset)
summary(model1)

# Create data.frame of all predictors
X<-subset(complete_dataset, select = -c(SP.DYN.LE00.IN))

# Correlation plot of all variables
seatpos.corr<-cor(X)
corrplot.mixed(seatpos.corr, lower.col = "black", number.cex = .5, tl.pos = "lt")

# Variation Inflation Factors of predictors
vif(X)

# Model 2 -> SP.DYN.AMRT.FE excluded
model2 <- lm(SP.DYN.LE00.IN ~ EG.ELC.ACCS.ZS + NY.ADJ.NNTY.KD.ZG + NY.ADJ.NNTY.KD +
SE.PRM.UNER.ZS + SE.XPD.PRIM.ZS + SP.DYN.IMRT.IN + SE.ADT.LITR.ZS + SP.POP.GROW +
SP.POP.TOTL + SE.PRM.CMPT.ZS + SH.XPD.CHEX.GD.ZS + SH.XPD.CHEX.PC.CD +
SL.UEM.TOTL.NE.ZS + SP.DYN.AMRT.MA + NY.GDP.MKTP.KD.ZG + NY.GDP.PCAP.PP.CD +
SP.DYN.CBRT.IN, data=complete_dataset)
summary(model2)

model_2_data <- subset(complete_dataset, select = -c(SP.DYN.AMRT.FE))
X2<-model_2_data
vif(X2)

# Model 3 -> SP.DYN.AMRT.FE and SP.DYN.CBRT.IN excluded
model3 <- lm(SP.DYN.LE00.IN ~ EG.ELC.ACCS.ZS + NY.ADJ.NNTY.KD.ZG + NY.ADJ.NNTY.KD +
SE.PRM.UNER.ZS + SE.XPD.PRIM.ZS + SP.DYN.IMRT.IN + SE.ADT.LITR.ZS + SP.POP.GROW +
SP.POP.TOTL + SE.PRM.CMPT.ZS + SH.XPD.CHEX.GD.ZS + SH.XPD.CHEX.PC.CD +
SL.UEM.TOTL.NE.ZS + SP.DYN.AMRT.MA + NY.GDP.MKTP.KD.ZG + NY.GDP.PCAP.PP.CD,
data=complete_dataset)
summary(model3)

model_3_data <- subset(complete_dataset, select = -c(SP.DYN.AMRT.FE, SP.DYN.CBRT.IN))
X3<-model_3_data
vif(X3)

# Model 4 ->SP.DYN.AMRT.FE, SP.DYN.CBRT.IN and SP.DYN.IMRT.IN excluded
model4 <- lm(SP.DYN.LE00.IN ~ EG.ELC.ACCS.ZS + NY.ADJ.NNTY.KD.ZG + NY.ADJ.NNTY.KD +
SE.PRM.UNER.ZS + SE.XPD.PRIM.ZS + SE.ADT.LITR.ZS + SP.POP.GROW + SP.POP.TOTL +
SE.PRM.CMPT.ZS + SH.XPD.CHEX.GD.ZS + SH.XPD.CHEX.PC.CD + SL.UEM.TOTL.NE.ZS +
SP.DYN.AMRT.MA + NY.GDP.MKTP.KD.ZG + NY.GDP.PCAP.PP.CD, data=complete_dataset)
summary(model4)
```

```
model_4_data <- subset (complete_dataset, select = -c(SP.DYN.AMRT.FE, SP.DYN.CBRT.IN,
SP.DYN.IMRT.IN))
X4<-model_4_data
vif(X4)

# Anova tests
anova(model2,model3)
anova(model3, model4)
anova(model2, model4)


######################## MODELLING (Q4a) ########################

## Train Test Split

names(complete_dataset)[1] <- "Target"
set.seed(123)
split = sample.split(complete_dataset$Target , SplitRatio = .9)
data_train = subset(complete_dataset, split == TRUE)
data_test = subset(complete_dataset, split == FALSE)

## Random Forest Model

# Training
regressor_rf <- randomForest(Target~ .,
                            data = data_train,
                            tree = 300,
                            mtry = 8,
                            proximity = TRUE,
                            importance = TRUE)

print(regressor_rf)

# Testing
y_pred_rf = predict(regressor_rf, newdata = data_test)

Pred_Actual_rf <- as.data.frame(cbind(Prediction = y_pred_rf, Actual = data_test$Target))

print(Pred_Actual_rf)
summary(Pred_Actual_rf)

# Plot
gg.rf <- ggplot(Pred_Actual_rf, aes(Actual, Prediction )) +
  geom_point() + theme_bw() + geom_abline() +
  labs(title = "Random Forest Regression", x = "Actual life expectancy",
       y = "Predicted life expectancy") +
  theme(plot.title = element_text(family = "Lucida Sans", face = "bold", size = (15)),
        axis.title = element_text(family = "Lucida Sans", size = (10)))
gg.rf

## Multiple Linear Regression

# Training
regressor_lm = lm(formula = Target ~ .,
                 data = data_train)
print(regressor_lm)

summary(regressor_lm)

# Testing
y_pred_lm = predict(regressor_lm, newdata = data_test)


Pred_Actual_lm <- as.data.frame(cbind(Prediction_lm = y_pred_lm, Actual_lm = data_test$Target))

# Plot
gg.lm <- ggplot(Pred_Actual_lm, aes(Actual_lm, Prediction_lm )) +
  geom_point() + theme_bw() + geom_abline() +
  labs(title = "Multiple Linear Regression", x = "Actual life expectancy",
      y = "Predicted life expectancy") +
  theme(plot.title = element_text(family = "Lucida Sans", face = "bold", size = (15)),
        axis.title = element_text(family = "Lucida Sans", size = (10)))
gg.lm

## Support Vector Regressor (SVR)

# Training
regressor_svr = svm(formula = Target ~ .,
                    data = data_train,
                    type = 'eps-regression',
                    kernel = 'radial')

summary(regressor_svr)


# Test
y_pred_svr = predict(regressor_svr,  newdata = data_test)

Pred_Actual_svr <- as.data.frame(cbind(Prediction_svr = y_pred_svr, Actual_svr = data_test$Target))


# Plot
gg.svr <- ggplot(Pred_Actual_svr, aes(Actual_svr, Prediction_svr )) +
  geom_point() + theme_bw() + geom_abline() +
  labs(title = "SVR", x = "Actual life expectancy",
      y = "Predicted life expectancy") +
  theme(plot.title = element_text(family = "Lucida Sans", face = "bold", size = (15)),
        axis.title = element_text(family = "Lucida Sans", size = (10)))
gg.svr


######################## EVALUATION (Q4b) ########################

## Mean Square Error of all Models
MSE.rf <- sum((data_test$Target - y_pred_rf)^2)/nrow(data_test)
MSE.lm <- sum((data_test$Target - y_pred_lm)^2)/nrow(data_test)
MSE.svr <- sum((data_test$Target - y_pred_svr)^2)/nrow(data_test)

print(paste("Mean Squared Error (Random Forest regressor):", MSE.rf))
print(paste("Mean Squared Error (Multiple Linear Regression):", MSE.lm))
print(paste("Mean Squared Error (Support vector regressor):", MSE.svr))

## Mean Absolute Error of all models
MAE.rf <- mae(data_test$Target, y_pred_rf)
MAE.lm <- mae(data_test$Target, y_pred_lm)
MAE.svr <- mae(data_test$Target, y_pred_svr)

print(paste("Mean Absolute Error (Random Forest regressor):", MAE.rf))
print(paste("Mean Absolute Error (Multiple Linear Regression):", MAE.lm))
print(paste("Mean Absolute Error (Support vector regressor):", MAE.svr))

## Root Mean square error of all models
RMSE.rf <- rmse(data_test$Target, y_pred_rf)
RMSE.lm <- rmse(data_test$Target, y_pred_lm)
RMSE.svr <- rmse(data_test$Target, y_pred_svr)

print(paste("Root Mean Squared Error (Random Forest regressor):", RMSE.rf))
print(paste("Root Mean Squared Error (Multiple Linear Regression):", RMSE.lm))
print(paste("Root Mean Squared Error (Support vector regressor):", RMSE.svr))

## R-square error of all models
R2.rf <- R2(data_test$Target, y_pred_rf, form = "traditional")
R2.lm <- R2(data_test$Target, y_pred_lm, form = "traditional")
R2.svr <- R2(data_test$Target, y_pred_svr, form = "traditional")

print(paste("R-square error  (Random Forest regressor):", R2.rf))
print(paste("R-square error  (Multiple Linear Regression):", R2.lm))
print(paste("R-square error  (Support vector regressor):", R2.svr))
```

```
######################## PREDICTIONS (Q4C) ########################

library("readxl")
holdout_data <- read_excel('LifeExpectancyData2.xlsx')

# Switch names to match
colnames(holdout_data)[1] <- "Country.Name"
colnames(holdout_data)[2] <- "Country.Code"

# Store dropped columns
dropped_cols <- subset(holdout_data, select = c(Country.Name, Country.Code))

# Drop columns for now
holdout_data = subset(holdout_data, select = -c(Country.Name, Country.Code))

# Drop target colum
data_drop <- subset(complete_dataset, select = -Target)

# Add the holdout data to the original dataset
data_merged <- rbind(data_drop, holdout_data)

# Convert fields to numeric
data_merged <- sapply(data_merged, as.numeric)

# Impute the missing values using MICE
impute = mice(data_merged,m = 10,method = "rf",seed = 22)

# Provide a summary of the imputed values
summary(impute)

# Provides the merged dataset with the imputed values
holdout_data = complete(impute)

# Selects only the original holdout dataset observations
holdout_data = holdout_data[233:243,]

# Uses the SVR model to predict on the holdout dataset
svr_pred = predict(regressor_svr,  newdata = holdout_data)
# Used the Random Forest model to predict on the holdout dataset
rf_pred = predict(regressor_rf, newdata = holdout_data)
# Used the multiple linear regression model to predict on the holdout dataset
lm_pred = predict(regressor_lm, newdata = holdout_data)

# Adds the predictions from all of the models on to the holdout dataset as columns
holdout_data$SVR_Life_Expectancy_preds = svr_pred
holdout_data$LM_Life_Expectancy_preds = lm_pred
holdout_data$RF_Life_Expectancy_preds = rf_pred


holdout_data <- cbind(dropped_cols, holdout_data)


# Exports the holdout dataset with predictions into an xlsx file
write.xlsx(
  holdout_data,
  file = 'holdout_predictions.xlsx' ,
  sheetName = 'Sheet1',
  col.names = TRUE,
  row.names = FALSE
)

######################## ANOVA (Q5) ########################
library(countrycode)

complete_dataset <- cbind(descriptive_cols, complete_dataset)

complete_dataset <- complete_dataset[1:186,]
```

```
# Add continent and subcontinent column
complete_dataset$continent <- countrycode(sourcevar = complete_dataset[, "Country.Name"],
                            origin = "country.name",
                            destination = "continent")

complete_dataset$subcont <- countrycode(sourcevar = complete_dataset[, "Country.Name"],
                          origin = "country.name",
                          destination = "region")

complete_dataset <- within(complete_dataset, continent[subcont == 'Latin America & Caribbean'] <- 'South
America')
complete_dataset <- within(complete_dataset, continent[subcont == 'North America'] <- 'North America')
complete_dataset <- within(complete_dataset, continent[Country.Name == 'Greenland'] <- 'North America')
complete_dataset <- within(complete_dataset, continent[Country.Name == 'Kosovo'] <- 'Europe')


complete_dataset$continent <- as.factor(complete_dataset$continent)

str(complete_dataset)

anova_data <- complete_dataset[1:186,]

anova_data <- subset(complete_dataset, select = -c(Country.Code, Country.Name))

levels(anova_data$continent)

group_by(anova_data, continent) %>%
  summarise(
    count = n(),
    mean = mean(Target, na.rm = TRUE),
    sd = sd(Target, na.rm = TRUE)
  )

library("ggpubr")
ggboxplot(anova_data, x = "continent", y = "Target",
          color = "continent",
          order = c("Africa", "Asia", "Europe", "North America", "Oceania", "South America"),
          ylab = "continent", xlab = "Life Expectancy")



# Compute the analysis of variance
res.aov <- aov(Target ~ continent, data = complete_dataset)
# Summary of the analysis
summary(res.aov)


TukeyHSD(res.aov)
```