DS 6372 PROJECT 1

Authors: Ahmed Awadallah, Luke Stodgel, Leo Leal

Intro

In this project we will be exploring different factors that affect life expectancy and creating a variety of prediction models to predict life expectancy.

Data Description

The data set we are using contains information from the years 2000-2015 for 193 countries. It focuses on immunization factors, mortality factors, economic factors, social factors and other health related factors. The data set comes from the World Health Organization and contains 22 columns (20 predicting variables) and 2938 rows.

EDA

After viewing the scatterplot matrix (appendix Figures 1a, 1b, 1c), there is clearly collinearity between the predictor variables thinness 1.19 years and thinness 5.9 years, income composition of resources and schooling, infant mortality and under 5 deaths, diphtheria and polio. We can also see a clear relationship between life expectancy and adult mortality, income composition of resources and schooling.

One of the important variables in the model, the percent of 1 year olds immunized with a polio vaccine, has some peculiar behavior. A linear regression plot of its relationship with life expectancy would appear to suggest that as immunization increases so does life expectancy, this is up until 15% immunization where it then drops to levels lower than if there was no immunization at all. After this peculiarity life expectancy continues to grow with immunization as expected(appendix Figure 1d.). This behavior is interesting but upon deeper inspection of the data some troubling problems become apparent. There is a severe lack of data between 9% and around 30% immunization(appendix Figure 1f). Additionally the data points build up around 9% in a time independent fashion which is stark contrast to the data above 9% where we do see a time dependence on its distribution(appendix Figure 1g). This behavior is seen not just in Polio but also for Diphtheria and Hepatitis B, it seems to be a common trend among vaccinations(appendix Figures 1h, 1i). The previously mentioned lack of data between 9% and around 30% immunization is likely not an issue. The data does display that as time goes on there is an increase in immunizations most likely due to better access and distribution of these resources across the world so this behavior makes sense on its own(appendix Figure 1e). The lower immunization percentages should have less and less data through time. The real issue is that there is something unnaturally special about vaccination percentages below 9%.

The Status feature does display a clear difference between the Developing and Developed when checking the boxplot for the Life Expectancy (appendix Figure 1j). The histogram of the Life Expectancy (Figure 1k) does not show any evidence against normality and though there are less observations of the Developed Status than there are of the Developing Status, there is enough observation to make a Welch's t-test under the hypothesis that the mean Life Expectancy for both Status is equal to each other (Figure 1l). The results of the hypothesis test shows that there is overwhelming evidence that the mean Life Expectancy of the countries with a Developed Status is not equal to that of the countries with Developing Status. The test also shows that, with a 95% confidence, the Life Expectancy of the countries that have a Developed Status is on average 11.47 to 12.47 years longer than that of the countries with a Developing Status. This indicates that the country's status may be a feature of interest. Checking the correlation of the numeric features with the Life.expectancy feature will indicate other interesting features (Figure 1m and 1n).

Objective 1: Interpretable Linear Model

Restatement of Problem

For Objective 1 we are trying to create the best model using techniques learned in unit 1 and unit 2. We will try to reduce redundancy and bias by visually identifying and removing collinear predictor variables in a scatterplot matrix and by also using a VIF function. After that we will run our model through a LASSO, ols stepwise and forward selection algorithm to get their recommendations on which variables to include in the final model. We will use the model with the highest R^2 and lowest RMSE and test ASE. After we get the best model we will make sure the data passes all of the assumptions for linear regression and then will interpret the parameters in the model.

Mini EDA

First we will look at a scatter plot to see if we can examine the relationship each predictor has with the response and also to look for multicollinearity. Our response variable, Life Expectancy, is highlighted in the scatterplot matrices labeled 1a, 1b and 1c in the appendix.

Right off the bat we can see correlation between a few sets of predictor variables: thinness 1-19 years and thinness 5-9 years, income composition of resources and schooling, infant mortality and under 5 deaths, diphtheria and polio. We can also see a clear relationship between life expectancy and adult mortality, income composition of resources and schooling.

We will run a VIF function iteratively on the full model for confirmation and additional help with finding and removing redundancy. The VIF function might be able to see multicollinearity that isn't easy for us to see through viewing the scatterplot matrix. We will do this until we have no variable that exceeds a 10 VIF score. In order for the VIF function to work with our model initially we had to remove the Status predictor variable because the function found 100% collinearity between Status, Country and one or more other variables. The VIF function identified GDP, Schooling, infant deaths, under five deaths as being collinear so we will **remove** those columns from our model.

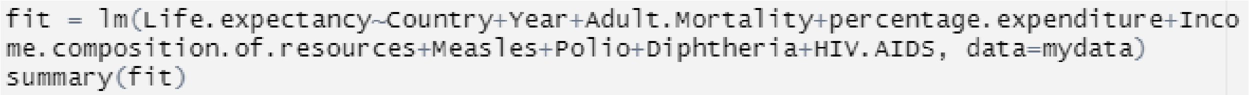
To address missing data, after going through and locating where the data was missing, I decided to only impute data in columns for countries that weren't completely empty. In other words, if we can take an average of data from that country and column to impute, then we will, otherwise, we will leave it blank. Imputing in this way brought the amount of missing data from 3.8% to 2.45% and also increased the R^2 and lowered the RMSE and test ASE of our final model. The imputation was done manually in Excel.

We will next run our model through a LASSO, ols stepwise, and ols forward selection process and compare their RMSEs, test ASEs and R^2s and proceed with using whichever has the better stats.

Variable Selection

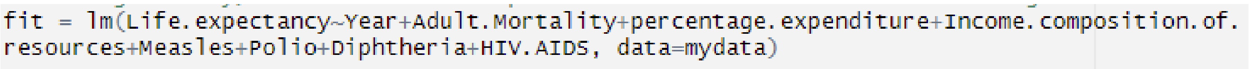
The variables selected by the ols forward and stepwise selection and lasso algorithms were very similar. The forward selection and LASSO algorithms actually only differed by one variable and after combining the two recommended models the strongest model was made.

This is our model thus far:



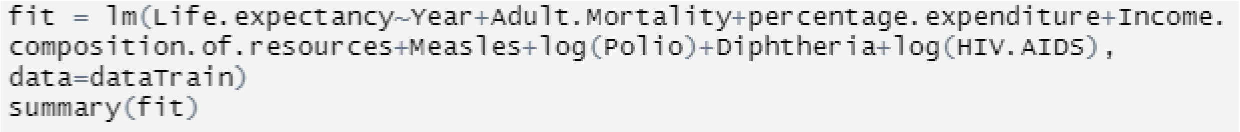
```
fit = lm(Life.expectancy~Country+Year+Adult.Mortality+percentage.expenditure+Inco
me.composition.of.resources+Measles+Polio+Diphtheria+HIV.AIDS, data=mydata)
summary(fit)
```

The summary statistics for this model showed an R^2 =0.96. I thought that was suspicious and when viewing the residual plots for the model, found that they broke the constant variance assumption. After removing the Country variable, the residual plots looked better, so now our model is this:



```
fit = lm(Life.expectancy~Year+Adult.Mortality+percentage.expenditure+Income.composition.of.
resources+Measles+Polio+Diphtheria+HIV.AIDS, data=mydata)
```

Additionally to improve our R^2, RMSE and test ASE values more, we logged the Polio and HIV/AIDS variables.
Our **final** model has an **R^2** of **0.8304**, an **RMSE** of **3.88**, and a **test ASE** of **15.29**. This is our final model:



```
fit = lm(Life.expectancy~Year+Adult.Mortality+percentage.expenditure+Income.
composition.of.resources+Measles+log(Polio)+Diphtheria+log(HIV.AIDS),
data=dataTrain)
summary(fit)
```

A plot of our predicted vs actual values from our test set can be viewed in the appendix and is labeled Figure 2a.

Checking Assumptions

The residual plots for our final model can be found in the appendix and are labeled Figure 3c. The plots are not perfect but they look like they pass the assumptions for linear regression. Also, there are no high leverage or high cook’s D values in our final model.

Parameter Interpretation

Please find the output of our summary(fit) and confint(fit) functions in the appendix labeled Figure 3a and 3b.

All of the variables present in our final model have a statistically significant effect on Life Expectancy.

Keeping all other variables constant, we are 95% confident that for every 1 unit increase in Year, Life expectancy will increase by 0.037 (p-value = 0.039). Our 95% confidence interval for this increase is [1.81, 7.28].
Keeping all other variables constant, we are 95% confident that for every 1 unit increase in Adult Mortality, Life expectancy will decrease by 0.015 (p-value = <2e-16). Our 95% confidence interval for this decrease is [-0.017, -0.013].
Keeping all other variables constant, we are 95% confident that for every 1 unit increase in Percentage expenditure, Life expectancy will increase by 0.000534 (p-value = <2e-16). Our 95% confidence interval for this increase is [4.50, 6.17].

Keeping all other variables constant, we are 95% confident that for every 1 unit increase in Income composition of resources, Life expectancy will increase by 13.4 (p-value = <2e-16). Our 95% confidence interval for this increase is [12.4, 14.3].
Keeping all other variables constant, we are 95% confident that for every 1 unit increase in Measles vaccinations, Life expectancy will decrease by 0.0000415 (p-value = 0.0000004). Our 95% confidence interval for this decrease is [-0.000058, -0.000025].
Keeping all other variables constant, we are 95% confident that a doubling of Polio vaccinations is associated with a 0.3345*log(2) = 0.232 unit increase in mean Life expectancy (p-value = 0.0398). Our 95% confidence interval for this increase is [0.016*log(2), 0.653*log(2)] = [0.011, 0.452].
Keeping all other variables constant, we are 95% confident that for every 1 unit increase in Diphtheria vaccinations, Life expectancy will increase by 0.3345 (p-value = 2.72e-13). Our 95% confidence interval for this increase is [0.025, 0.042].
Keeping all other variables constant, we are 95% confident that a doubling of the HIV/AIDS variable is associated with a 2.69*log(2) = 1.86 unit decrease in mean Life expectancy (p-value = <2e-16). Our 95% confidence interval for this decrease is [-2.83*log(2), -2.55*log(2)] = [-1.96, -1.77].

Objective 2: Complex Linear Model and Nonparametric Model

For this objective we needed to develop two more models. The first model is linear but allowed to be infinitely complex, meaning that the predictors' relationship to the target no longer needs to be understood or explained as long as the model performs better with the transformations. The second model needs to use a nonparametric approach such as KNN regression or binary tree regression.

Complex Linear Model

The approach for the complex linear model was to use features that had no NA values or had 5% or less of its data NA. The NA values were then removed from the dataset. After the removal process 1.7% of the datasets rows were removed(around 50 entries). The following were the countries affected by this process.

- Tuvalu: Island near Australia
- Timor-Leste: Island near Australia
- Sudan: North African Country
- South Sudan: Mid African Country
- San Marino: Small Country in Italy
- Saint Kitts and Nevis: Small Island North of South America
- Palau: Island near Australia
- Niue: Island near Australia
- Nauru: Island near Australia
- Montenegro: Country in the Balkans
- Monaco: Country in between France and Italy
- Marshall Islands: Island near Australia
- Dominica: Small Island North of South America
- Cook Islands: Island far east of Australia

Out of all these countries only Timor-Leste and Montenegro had data left. Interpretations of our model need to be tempered in the following three ways. First, all of these countries are now underrepresented in our model and using our model to make predictions for them is now inappropriate. Second, a lot of the affected countries are small islands off the coast of Australia so any use of our model to predict similar islands in the area would also be inappropriate. As for the rest of the world the impact on the population distributions by the NA removal process is largely non-existent.

The linear model was built using the following process. First the data types were cleaned and NA values were removed. The data was then standardized(LASSO selection depends on unit size) and split into a training, validation and test set using a 80-10-10 split. Training and validation sets were used throughout the model building process while the test is only used at the very end for our final ASE metric. After this a simple linear model was built without any feature selection to observe base performance. Next LASSO was used to remove non essential features and another model was built with the new reduced feature set. Then complex features were added to the model. The addition of complex features used a graphical approach. Every feature relationship with Life Expectancy was analyzed for basic polynomial behaviors. These basic polynomial behaviors were then introduced into the model for the variable. For example if a feature has 2 vertices and was an odd function this would translate to the general behavior of a cubic polynomial. The model would then use feature + feature^2 + feature^3 for prediction instead of just feature. Once this complex model was trained additional features that were no longer significant for prediction were removed from the model. The summary statistics and residual plots for this model are displayed in figures 4a and 4b respectively. The performance metrics for the final model is found in figures 4c and 4d.

Nonparametric Model

A parametric model assumes the target variable has a linear relationship with the predictors. For this part of the project, we intend to use nonparametric regression models to make predictions on the "Life.expectancy" variable. One of the nonparametric models used in this study is the regression tree. The logic behind regression tree is simple. The model is built from the top down by first finding a factor that is important in determining the value of the target variable. The algorithm then finds a value in which to split that factor, creating 2 branches. The next important factor is chosen for each of the branches and the process repeats itself. The model ends with the leaves, which is where the algorithm judges there are no more important factors and makes a prediction on the value by taking the mean of the target variables that the model knows falls into that leaf.

The feature selection approach for the regression tree model was to select features that did not contain any missing values, transforming categorical features into dummy variables, and taking correlation between the predictors and the target variables into consideration. To ensure there would not be any missing values the first step was to select the features that had little missing values, in this case 34 or less, and removing the rows in which those values were missing. The next step taken was to remove the features that had 163 or more missing values in them. The other features removed from the data set were the Year and the Country features. The year feature was removed because the model is not intended to be used as a time series and any predictions made based on time would be made using values the model has not seen. The Country feature is removed because some of the countries are either underrepresented or not represented at all. The final dataset contains 13 features, 12 predictors and the target variable.

The model was built using the *tree* package in r. To build the model, the dataset was randomly split into 80% training data and 20% testing data. All the features were used for the model because the r function will determine which features are the best.

As we can see from the summary of the model, the tree function from the tree package in r chose 5 variables out of the 12 predictors that it deemed to be the best estimators of the Life Expectancy value (see Appendix Figure 5a). The plot of the tree shows the importance of those variables. Figure 5b shows the plot of the regression tree.

The plot shows in which order the factors were used and which value of those factors were used to split the branches (see Appendix Figure 5b). For example, the HIV.AIDS factor is the first node of the tree. The observations that have an HIV.AIDS value below 0.65 will move to the left side of it, while those that have an HIV.AIDS value >= 0.65 will move to the right. At the bottom end of the plot we can see the final nodes, or the leaves, which are the 9 possible predicted values for this model.

When we use the model to predict the Life.expectancy values of the test data, we get the RMSE of 3.853341 and MAE of 2.840269. Those values for RMSE and MAE can be considered good since the model chooses from 9 distinct values it came up with while training to be used as the predicted value.

Another nonparametric model taken into consideration for this study was the k-nearest neighbor (knn) regression model. The knn model uses a k number of neighbors to estimate the value of the response variable. It will accomplish that by calculating the distance between the predictors for the response we wish to attain and the predictors of the training data. Once the response variable of the training data is found, the algorithm will average them out to make the predicted value. The feature used for this model will be the same features deemed important in the decision tree model, with the addition of the Developed feature. The Developed feature is a dummy feature created to replace the Status feature since it was the only categorical variable in the dataset. The "Developed" feature has a numerical value of "1" for the observations where the "Status" is Developed, and a numerical value of "0" for the observations where the "Status" is Developing.

For the knn model, a range of 1 to 15 will be used in the training to see which value of k yields the best RMSE. A method known as cross validation will also be used to help in the training of the model. What cross validation will do is divide the training dataset into n subsets, in this case n = 5, and train the model with 4 of them and test it on the 5th then it retrains the model using a different subset to make the tests. All of the subsets are used in both the training and testing for each of the k values. The result of the knn model goes as follows:

The model results show that the training data set was equally divided into 5 subsets, with only one of them having 1847 observations, 1 less than the other 4 which had 1848 each (see Appendix Figure 5c). The results also show that the best value for k was chosen by the lowest RMSE during the training phase. The result suggests that the best model is one that uses 2 observation points from the training data set to make a prediction on any new data. Using k = 2 not only yielded the lowest RMSE, it also produced the highest Rsquared for the training dataset.

When utilizing the final model to make predictions on the test dataset, it produces an RMSE of approximately 2.33, and a MAE of approximately 1.56. Those better results than the ones produced by the regression tree model because it is not creating a set of results from which to pick, but instead it is calculating the results taking the average of the 2 closest neighbors according to the predictors. In other words, the regression tree model has a preset predicting variables while the regression tree makes predictions on a case by case basis, and for this dataset it seems to be a better method.

For the complex linear model the best **test ASE** was **17.59**, for the nonparametric model the best **test ASE** was **5.43**. Clearly the nonparametric model performed the best which isn't entirely surprising. The linear model requires the model to have transformations that accurately reflect the relationships in the data. Some relationships may be impossible to represent with a parametric restriction which will limit its performance. The KNN regression model doesn't need any tuning as it is fully capable of representing the complex relationships without any regard for parametric forms. This gives it the freedom to express relationships in the data in ways that the linear model is restricted from and thus giving it stronger performance.

Final Summary

For Objective 1 we created our final model after performing an EDA, comparing results from different variable selection algorithms and then manually testing different variables to see if any more improvement could be made to our model. In the end we found that combining the variables from the forward-selection and the LASSO recommended models created the best model. Our best model used eight out of the 20 potential predictor variables, namely: year, adult mortality, percentage expenditure, income composition of resources, measles, polio, diphtheria, hiv/aids. Our **final** model had an **AIC** of **13055.06**, **adj R^2** of **0.8299**, an **RMSE** of **3.88**, and a **test ASE** of **15.29**. The scope of inference for objective 1 is all of the countries within the dataset. No data was removed and some data was imputed in objective 1 only.

The final complex linear model had an **AIC** of **2703**, **adj R^2** of **0.811**, and **test ASE** of **17.59**. As stated earlier this model removed NA features so the scope of inference had to be slightly adjusted. Tuvalu, Timor-Leste, Sudan, South Sudan, San Marino, Saint Kitts, Palau, Niue, Nauru, Montenegro, Monaco, Marshall Islands, Dominica, and Cook Islands all had most if not all of their data remove and as a consequence our results can't be extended to these countries. Additionally a good portion of these countries are actually islands off the coast of Australia thus our results do not extend to other similar islands. Excluding these two cases the rest of the world lies within our scope of inference.

For the nonparametric models in Objective 2 we created 2 different models, a regression tree and a regressive k-nearest neighbor. Both models used a dataset that eliminated missing values by filtering out 50 rows of missing values and 7 features that had a high missing values count. The Year and Country features were also left out of the dataset because Year is not recommended to use outside of a time series model and the Country feature had countries that were either underrepresented or not represented at all. The regression tree algorithm developed a model that would use 5 of the 12 predicting variables to predict the values for the Life Expectancy. Those features were: HIV.AIDS, Adult.Mortality, thinness.5.9.years, thinness..1.19.years, and under.five.deaths. When used to make predictions on the test dataset, the model yielded **RMSE** of **3.85**, a **MAE** of **2.84**, and an **ASE** of **5.43**. The regressing knn model used the same predictors as the regression tree model as well as the Status feature transformed into a dummy feature named Developed that indicated whether the status was Developed or not. That model yielded **RMSE** of **2.795**, **Rsquared** of **0.915**, and **MAE** of **1.79**. With the test dataset, the knn produced an **RMSE** of **2.33**, an **MAE** of **1.56** and an **ASE** of **5.43**. Because of those values, the regressive knn is the best nonparametric model to be used with this dataset.

If there was more time there are a few things we would have liked to do. The first would be a thorough exploration of the data's validity. It's clear that the data has a lot of oddities and if the data is compromised then the models are compromised as well. Some examples are population values are all over the place, vaccination has an odd time independent behavior, certain countries have nonsensical values when compared to other countries, some countries only have 1 data point in a non random way.

Lastly, the knn model showed that the Developed (Status) variable was an important predictor for a non parametric model. It is interesting that the regression tree did not use it to make its predictions. It would be interesting to learn what method the regression tree model used to do feature selection to see if it actually used the best one or if there was not a better method to select the predictors for this dataset.

Appendix

Source R Code: https://github.com/AhmedAmrAwadallahK8/DS6372-Project1

Figure 1a.

```
pairs(mydata[,  c(2, 4:10)])
```

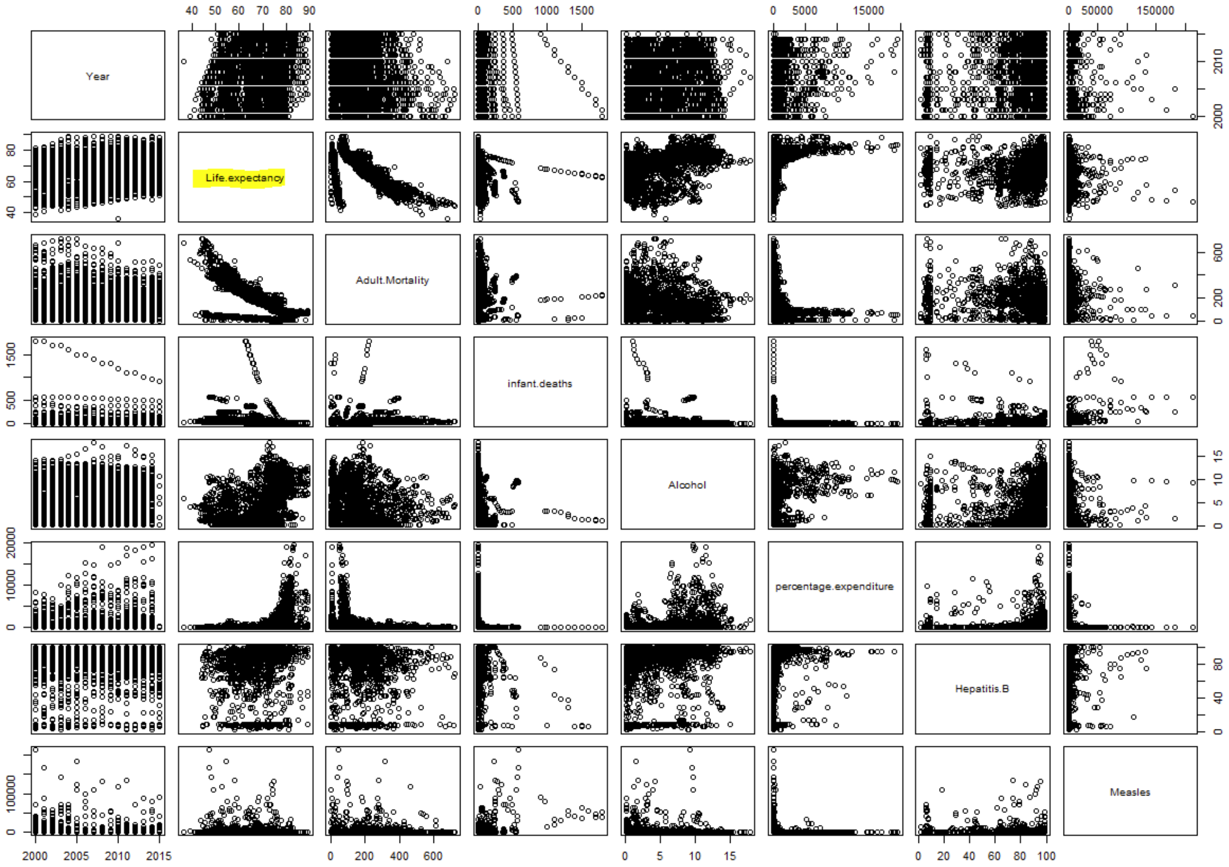


Figure 1b.

```
pairs(mydata[,  c(4, 11:16)])
```

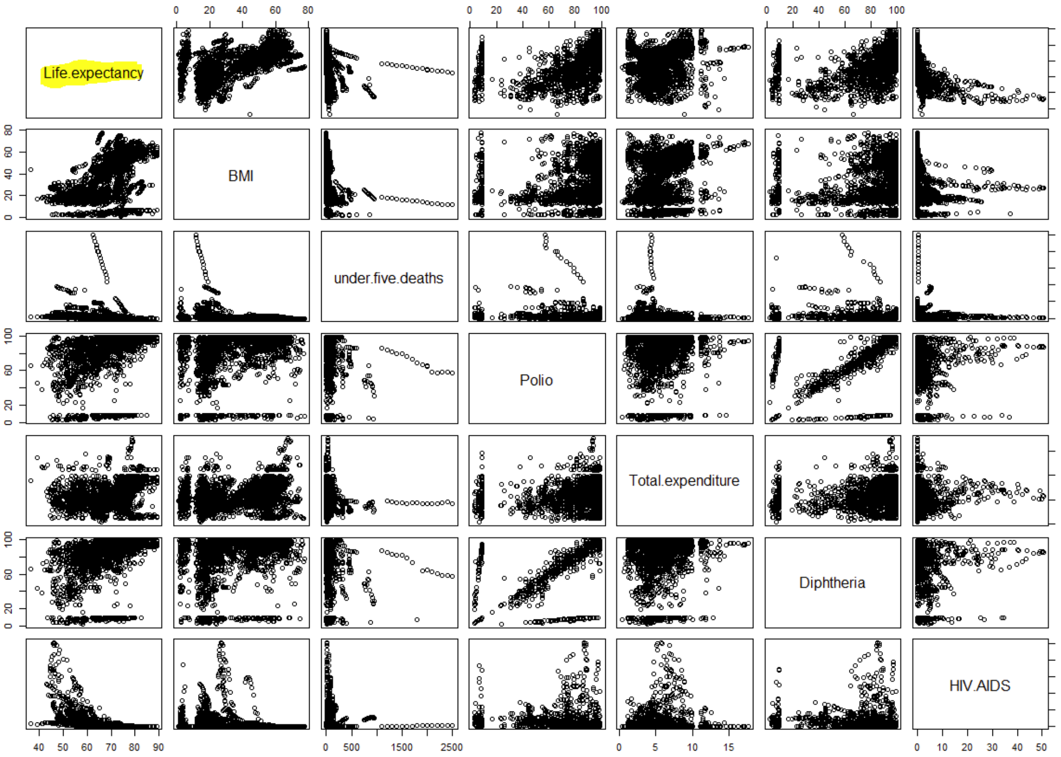


Figure 1c.

```
pairs(mydata[,  c(4, 17:22)])
```

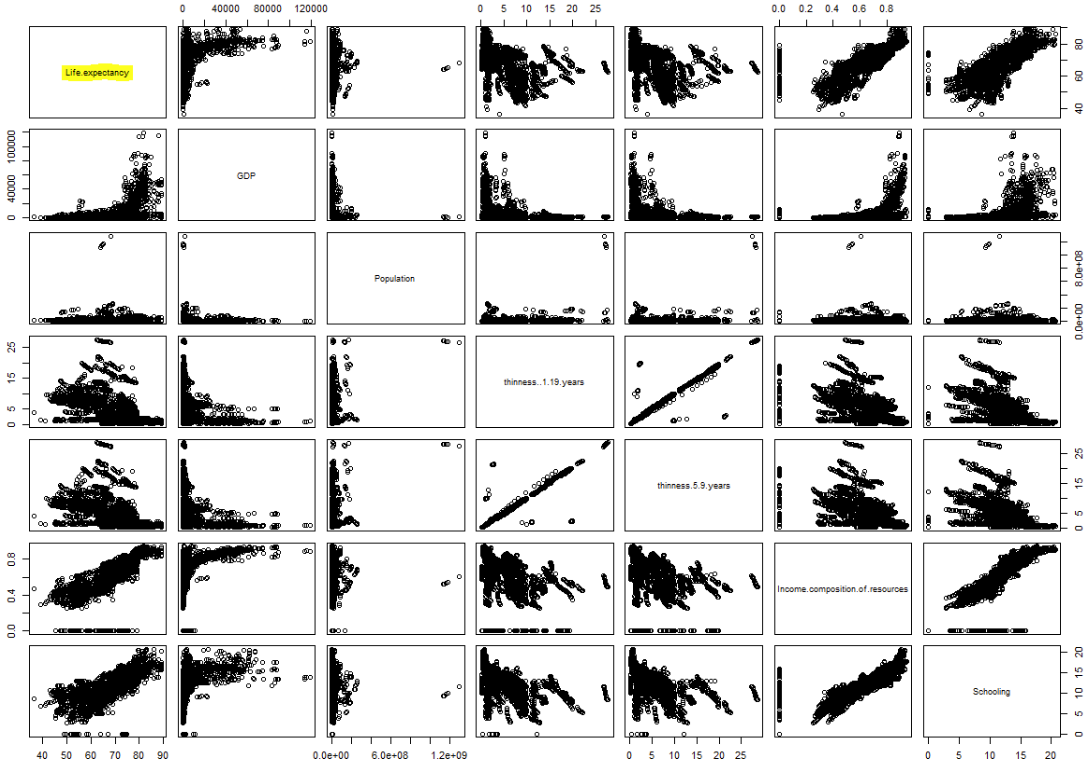


Figure 1d.

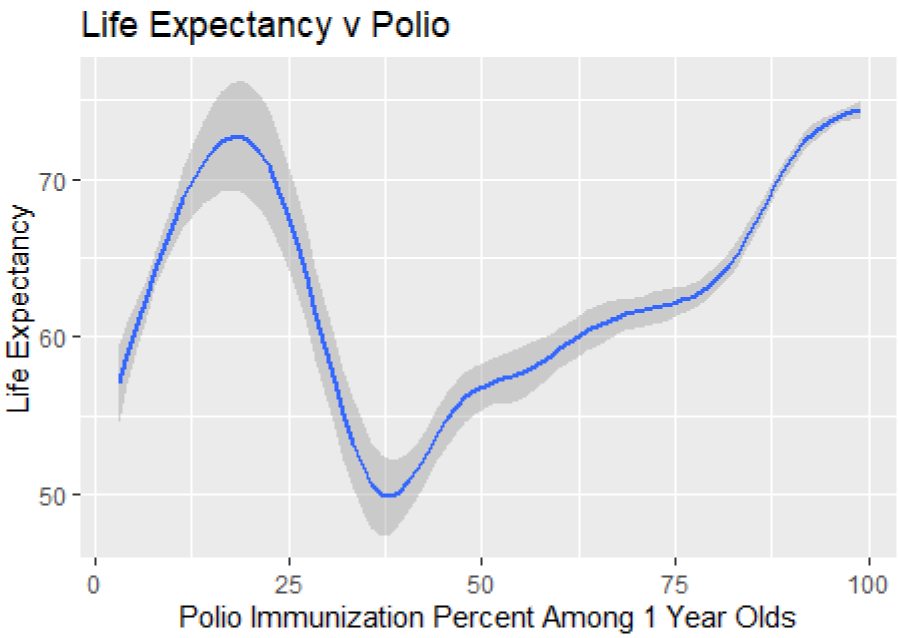


Figure 1e.

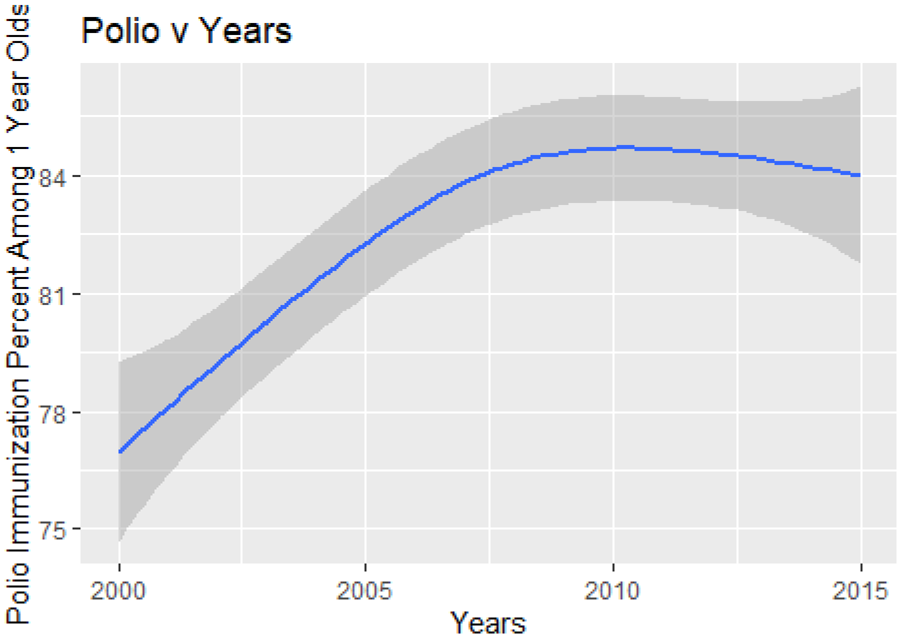


Figure 1f.

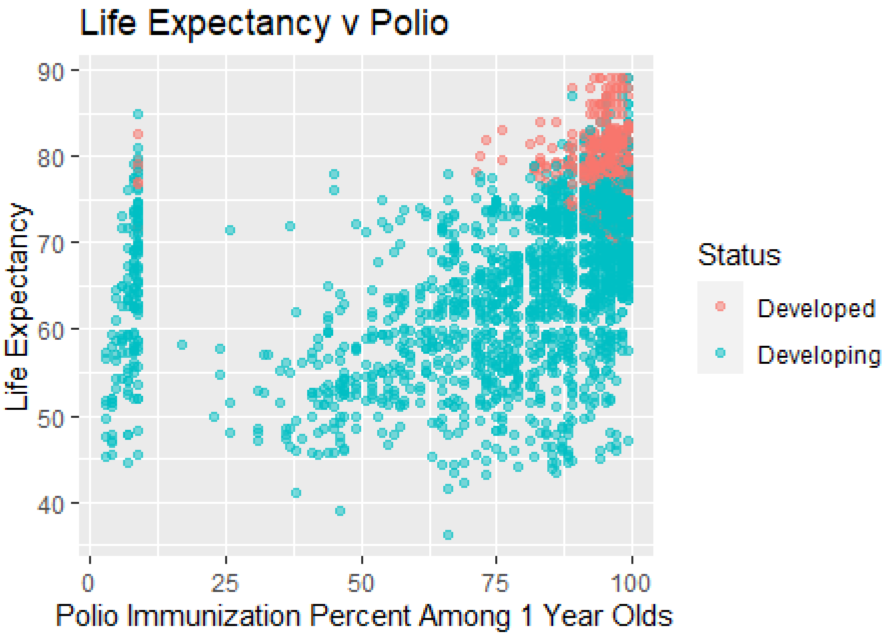


Figure 1g.

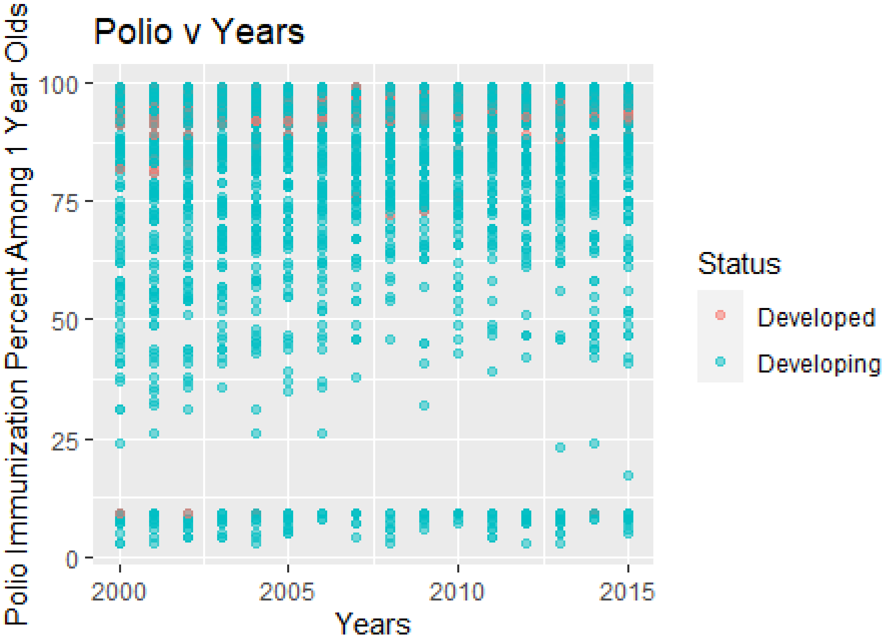


Figure 1h.

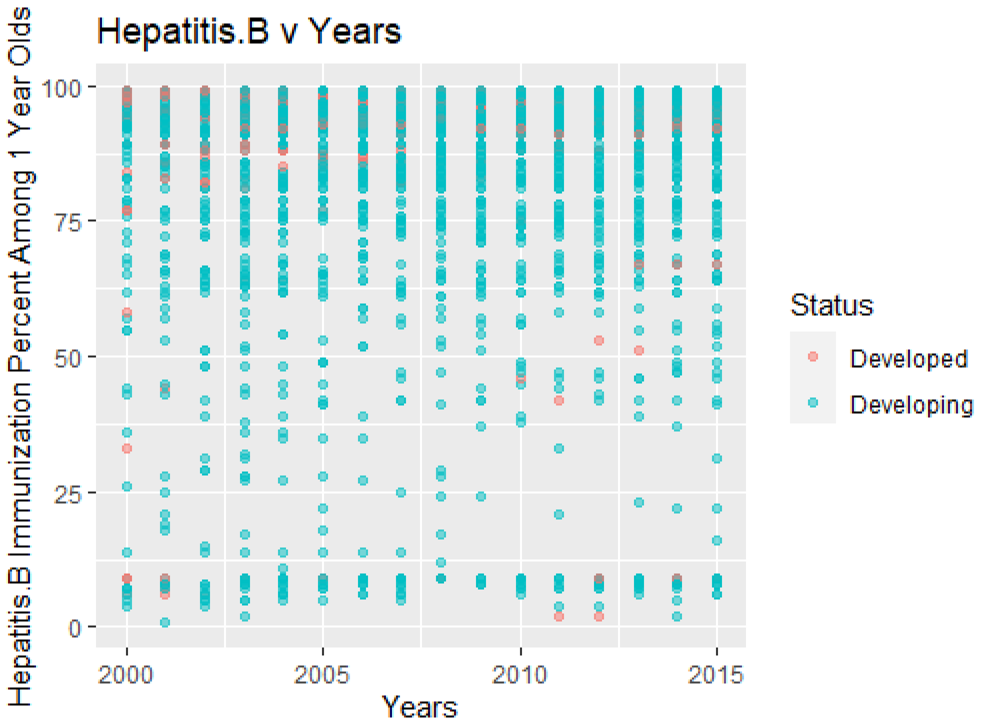


Figure 1i.

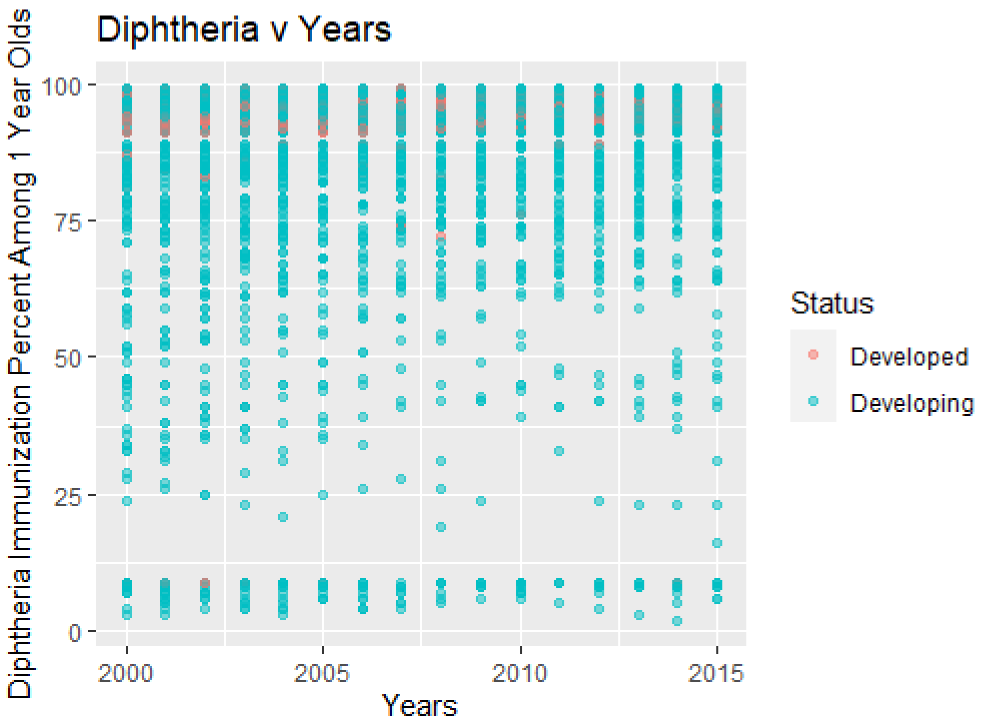


Figure 1j.

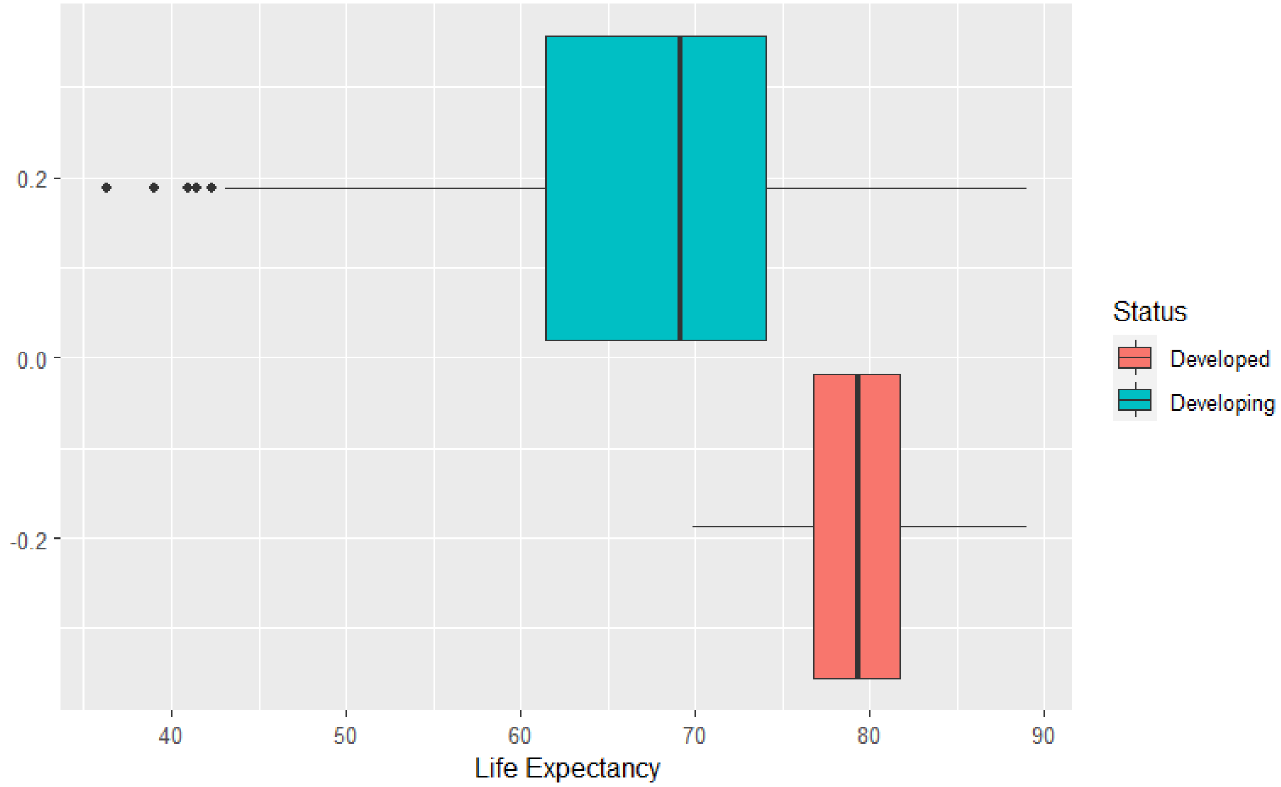


Figure 1k.

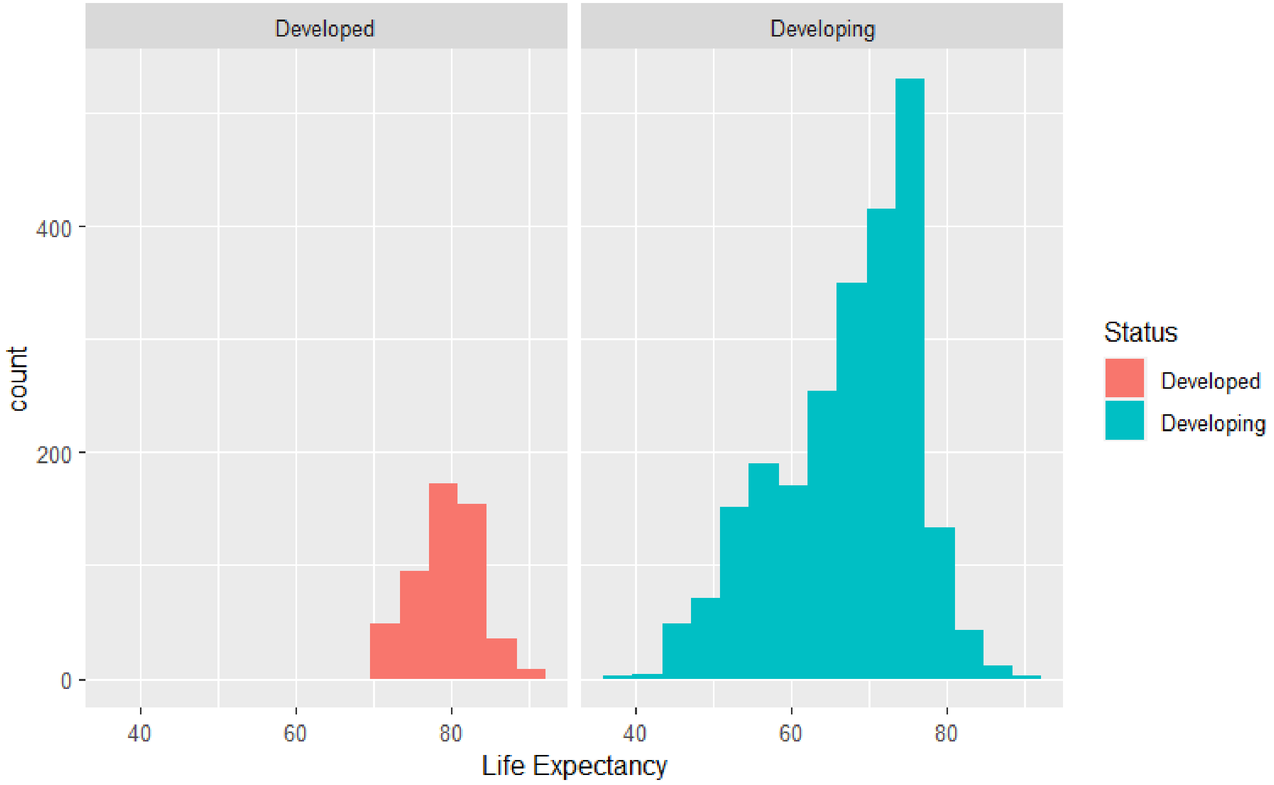
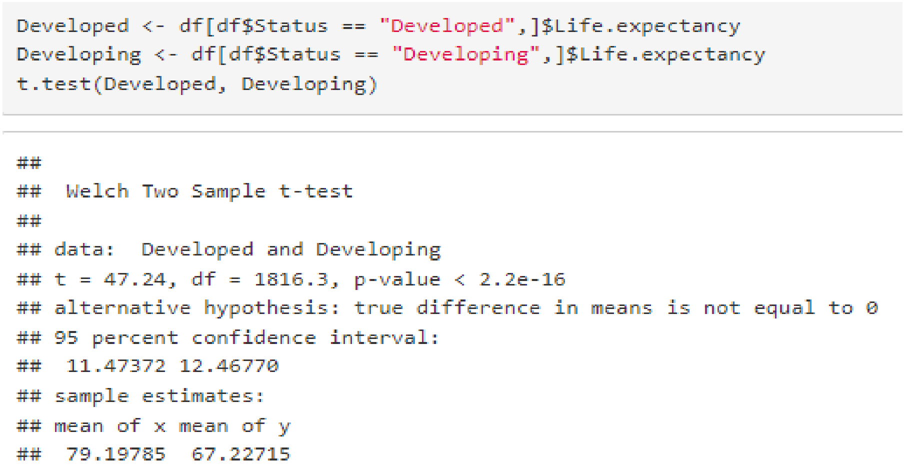


Figure 1l.



```
Developed <- df[df$Status == "Developed",]$Life.expectancy
Developing <- df[df$Status == "Developing",]$Life.expectancy
t.test(Developed, Developing)
```

```
##
##  Welch Two Sample t-test
##
## data:  Developed and Developing
## t = 47.24, df = 1816.3, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  11.47372 12.46770
## sample estimates:
## mean of x mean of y
##  79.19785  67.22715
```

Figure 1m.

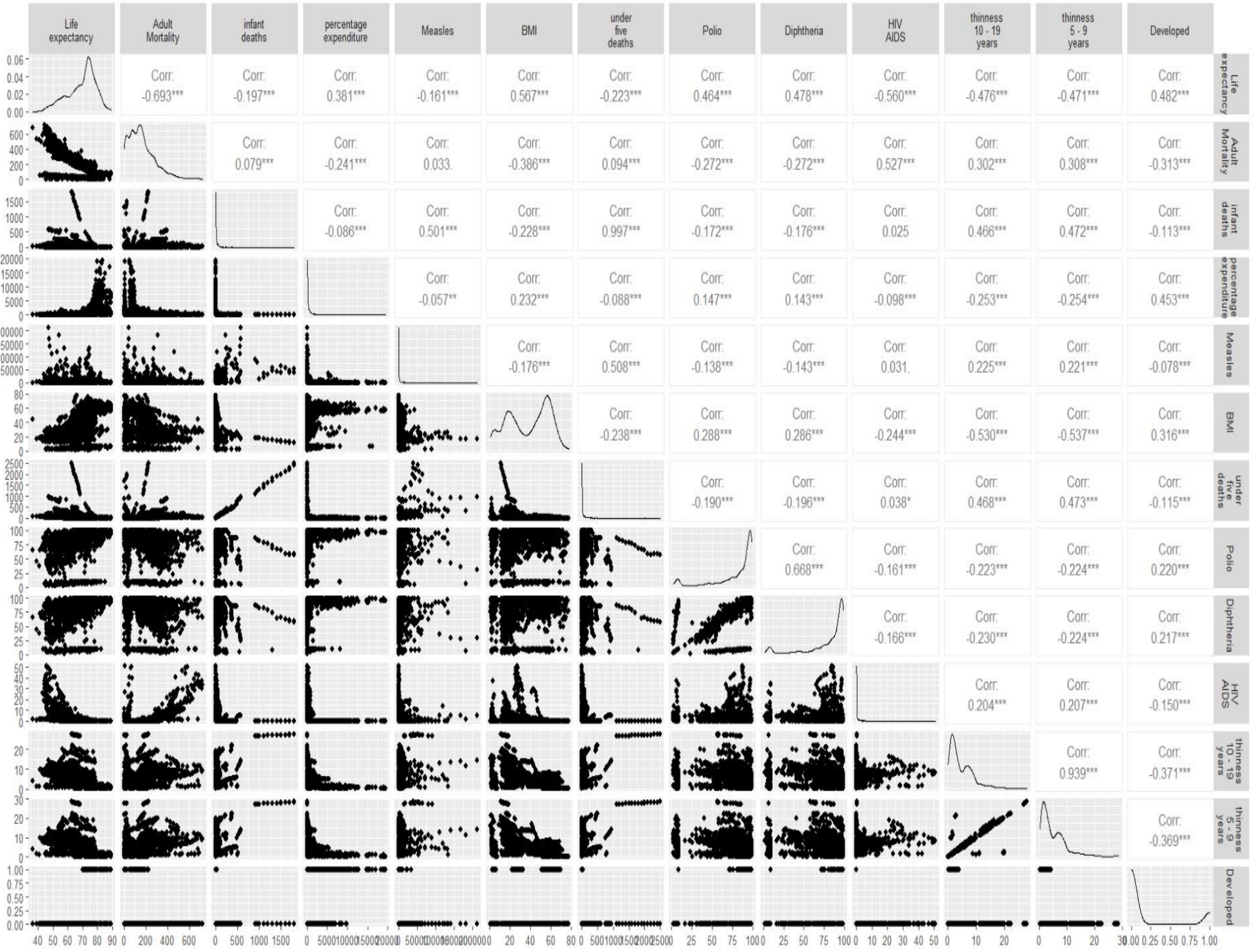
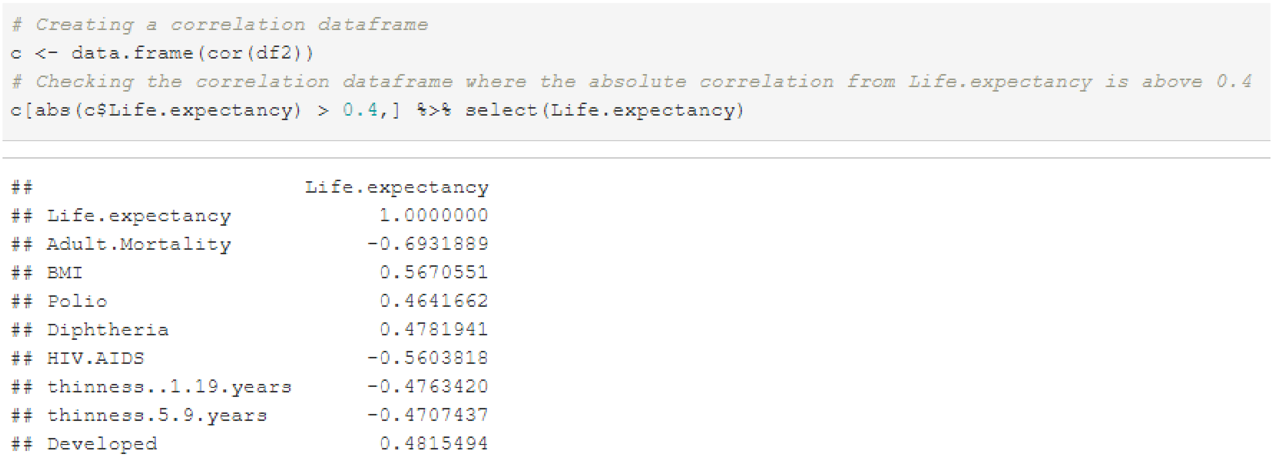


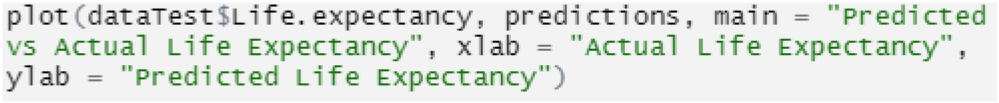
Figure 1n.



```
# Creating a correlation dataframe
c <- data.frame(cor(df2))
# Checking the correlation dataframe where the absolute correlation from Life.expectancy is above 0.4
c[abs(c$Life.expectancy) > 0.4,] %>% select(Life.expectancy)
```

```
##                          Life.expectancy
## Life.expectancy                1.0000000
## Adult.Mortality               -0.6931889
## BMI                            0.5670551
## Polio                          0.4641662
## Diphtheria                     0.4781941
## HIV.AIDS                      -0.5603818
## thinness..1.19.years          -0.4763420
## thinness.5.9.years            -0.4707437
## Developed                      0.4815494
```

Figure 2a.



```
plot(dataTest$Life.expectancy, predictions, main = "Predicted
vs Actual Life Expectancy", xlab = "Actual Life Expectancy",
ylab = "Predicted Life Expectancy")
```

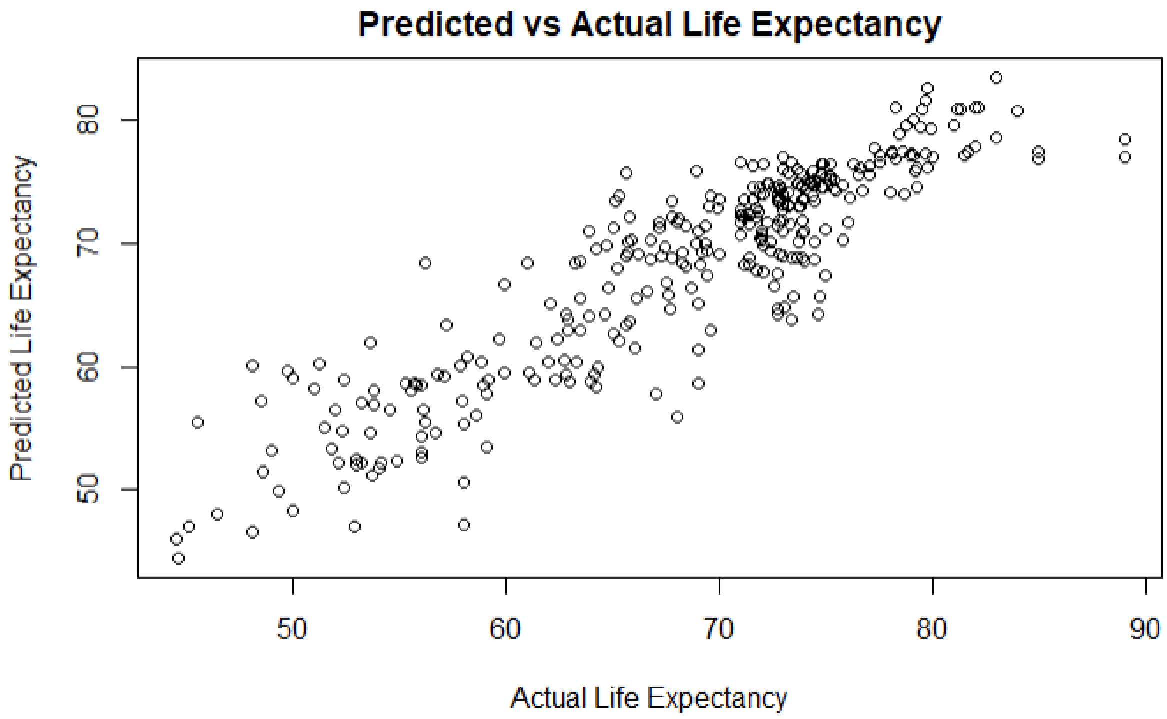
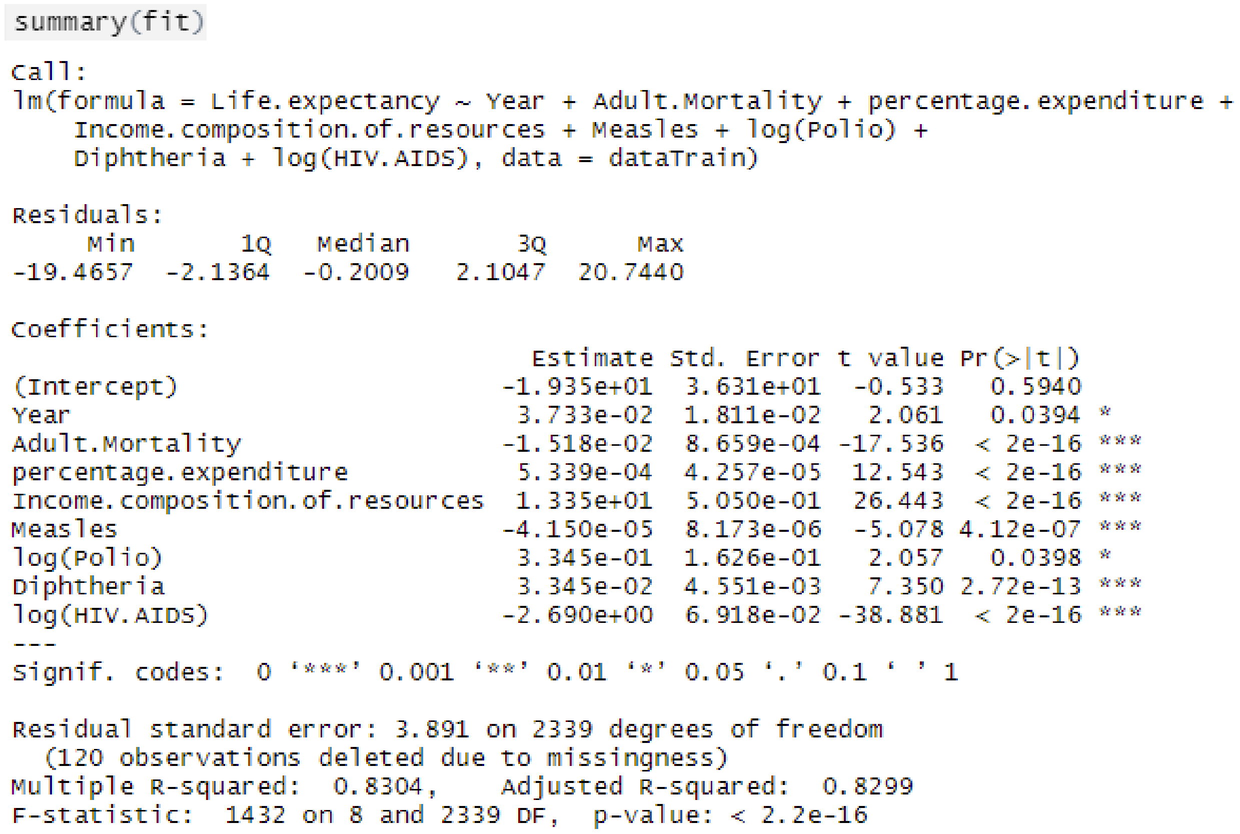


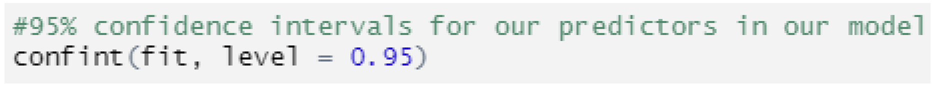
Figure 3a.



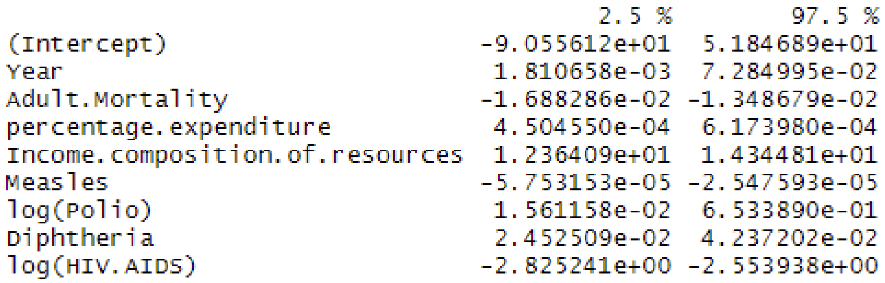
```
summary(fit)
```

```
Call:
lm(formula = Life.expectancy ~ Year + Adult.Mortality + percentage.expenditure +
    Income.composition.of.resources + Measles + log(Polio) +
    Diphtheria + log(HIV.AIDS), data = dataTrain)

Residuals:
     Min       1Q   Median       3Q      Max
-19.4657  -2.1364  -0.2009   2.1047  20.7440

Coefficients:
                                  Estimate Std. Error t value Pr(>|t|)
(Intercept)                     -1.935e+01  3.631e+01  -0.533   0.5940
Year                             3.733e-02  1.811e-02   2.061   0.0394 *
Adult.Mortality                 -1.518e-02  8.659e-04 -17.536  < 2e-16 ***
percentage.expenditure           5.339e-04  4.257e-05  12.543  < 2e-16 ***
Income.composition.of.resources  1.335e+01  5.050e-01  26.443  < 2e-16 ***
Measles                         -4.150e-05  8.173e-06  -5.078 4.12e-07 ***
log(Polio)                       3.345e-01  1.626e-01   2.057   0.0398 *
Diphtheria                       3.345e-02  4.551e-03   7.350 2.72e-13 ***
log(HIV.AIDS)                   -2.690e+00  6.918e-02 -38.881  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.891 on 2339 degrees of freedom
  (120 observations deleted due to missingness)
Multiple R-squared:  0.8304,    Adjusted R-squared:  0.8299
F-statistic:  1432 on 8 and 2339 DF,  p-value: < 2.2e-16
```

Figure 3b.

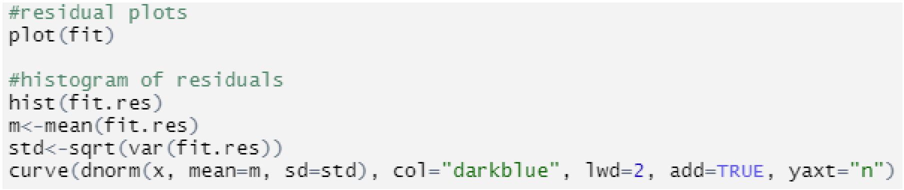


```
#95% confidence intervals for our predictors in our model
confint(fit, level = 0.95)
```

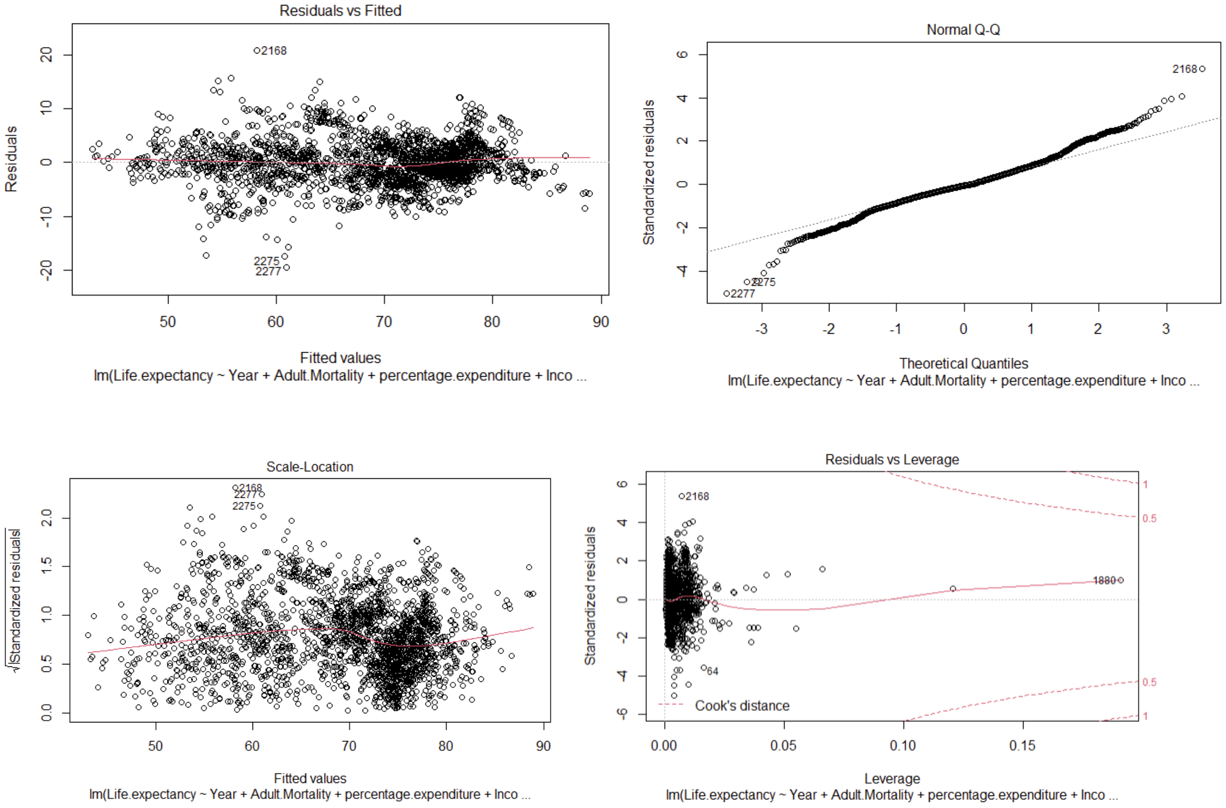


```
                                      2.5 %        97.5 %
(Intercept)                   -9.055612e+01  5.184689e+01
Year                           1.810658e-03  7.284995e-02
Adult.Mortality               -1.688286e-02 -1.348679e-02
percentage.expenditure         4.504550e-04  6.173980e-04
Income.composition.of.resources 1.236409e+01  1.434481e+01
Measles                       -5.753153e-05 -2.547593e-05
log(Polio)                     1.561158e-02  6.533890e-01
Diphtheria                     2.452509e-02  4.237202e-02
log(HIV.AIDS)                 -2.825241e+00 -2.553938e+00
```

Figure 3c.



```
#residual plots
plot(fit)

#histogram of residuals
hist(fit.res)
m<-mean(fit.res)
std<-sqrt(var(fit.res))
curve(dnorm(x, mean=m, sd=std), col="darkblue", lwd=2, add=TRUE, yaxt="n")
```



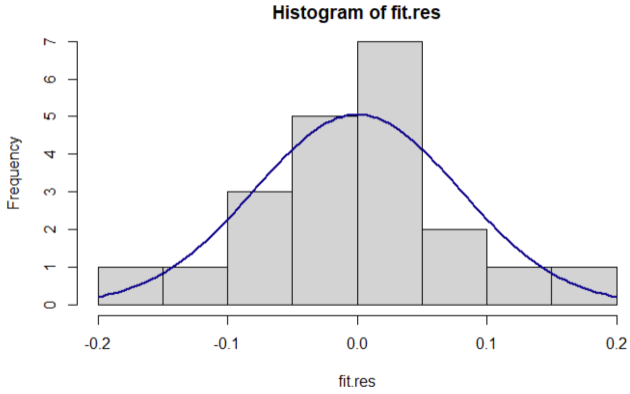
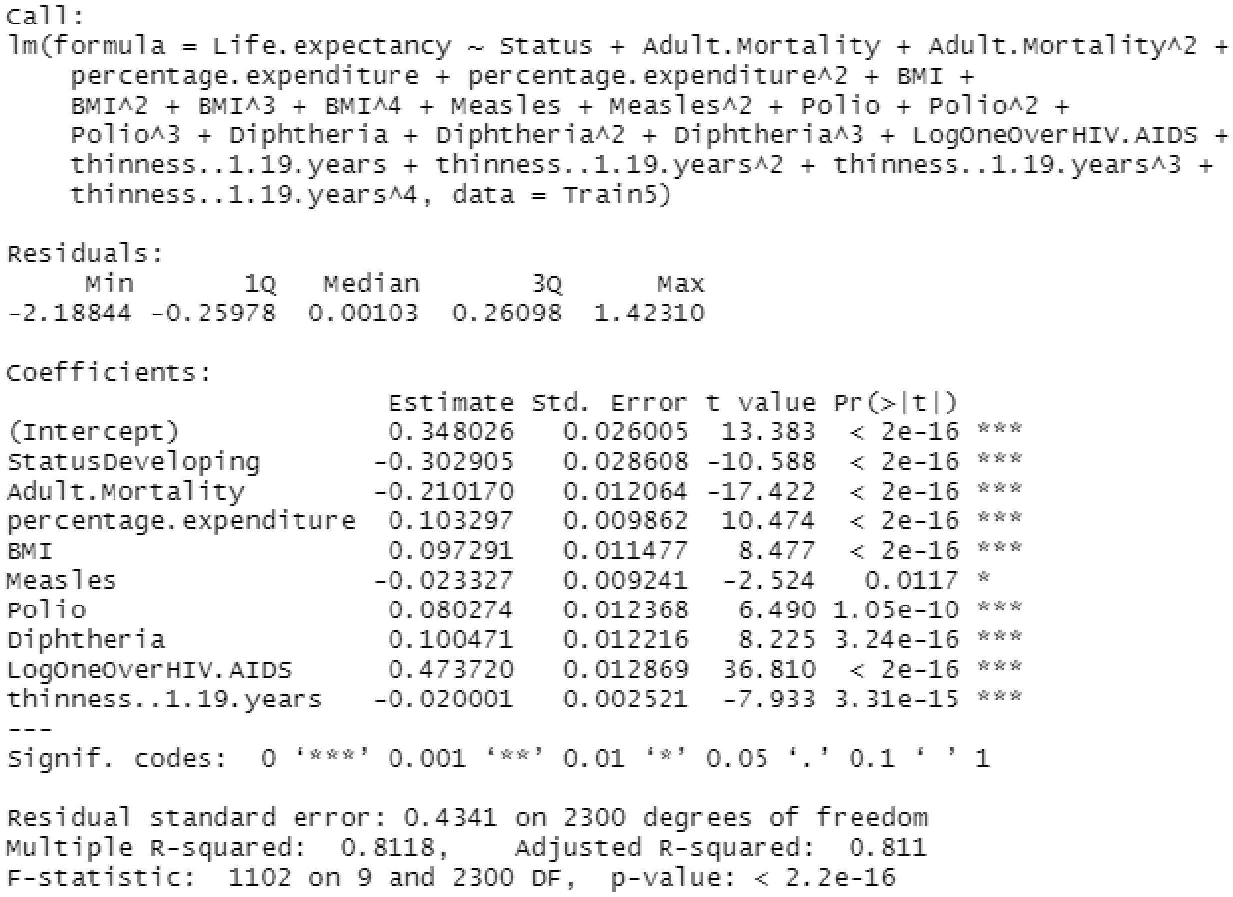
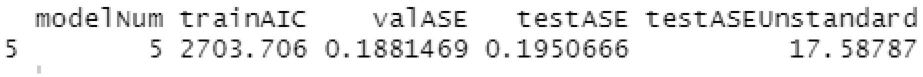


Figure 4a.



```
Call:
lm(formula = Life.expectancy ~ Status + Adult.Mortality + Adult.Mortality^2 + 
    percentage.expenditure + percentage.expenditure^2 + BMI + 
    BMI^2 + BMI^3 + BMI^4 + Measles + Measles^2 + Polio + Polio^2 + 
    Polio^3 + Diphtheria + Diphtheria^2 + Diphtheria^3 + LogOneOverHIV.AIDS + 
    thinness..1.19.years + thinness..1.19.years^2 + thinness..1.19.years^3 + 
    thinness..1.19.years^4, data = Train5)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.18844 -0.25978  0.00103  0.26098  1.42310 

Coefficients:
                        Estimate Std. Error t value Pr(>|t|)    
(Intercept)             0.348026   0.026005  13.383  < 2e-16 ***
StatusDeveloping       -0.302905   0.028608 -10.588  < 2e-16 ***
Adult.Mortality        -0.210170   0.012064 -17.422  < 2e-16 ***
percentage.expenditure  0.103297   0.009862  10.474  < 2e-16 ***
BMI                     0.097291   0.011477   8.477  < 2e-16 ***
Measles                -0.023327   0.009241  -2.524   0.0117 *  
Polio                   0.080274   0.012368   6.490 1.05e-10 ***
Diphtheria              0.100471   0.012216   8.225 3.24e-16 ***
LogOneOverHIV.AIDS      0.473720   0.012869  36.810  < 2e-16 ***
thinness..1.19.years   -0.020001   0.002521  -7.933 3.31e-15 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.4341 on 2300 degrees of freedom
Multiple R-squared:  0.8118,	Adjusted R-squared:  0.811 
F-statistic:  1102 on 9 and 2300 DF,  p-value: < 2.2e-16
```

Figure 4b.



```
  modelNum trainAIC    valASE   testASE testASEUnstandard
5        5 2703.706 0.1881469 0.1950666          17.58787
```

Figure 4c

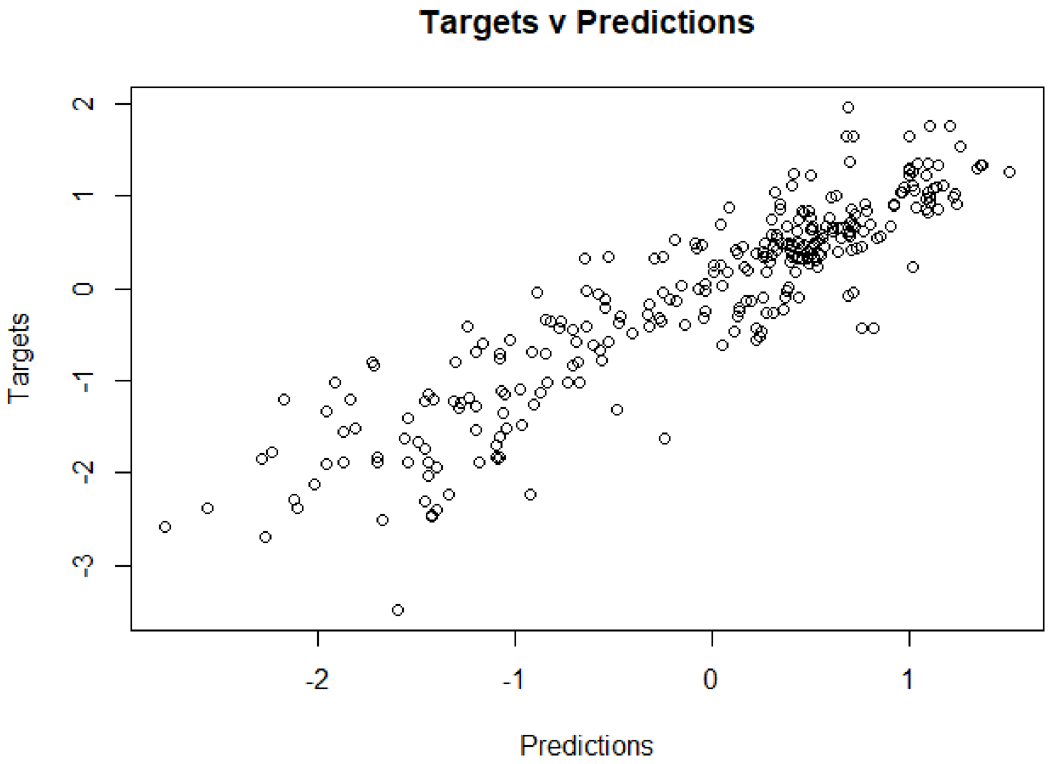


Figure 4d

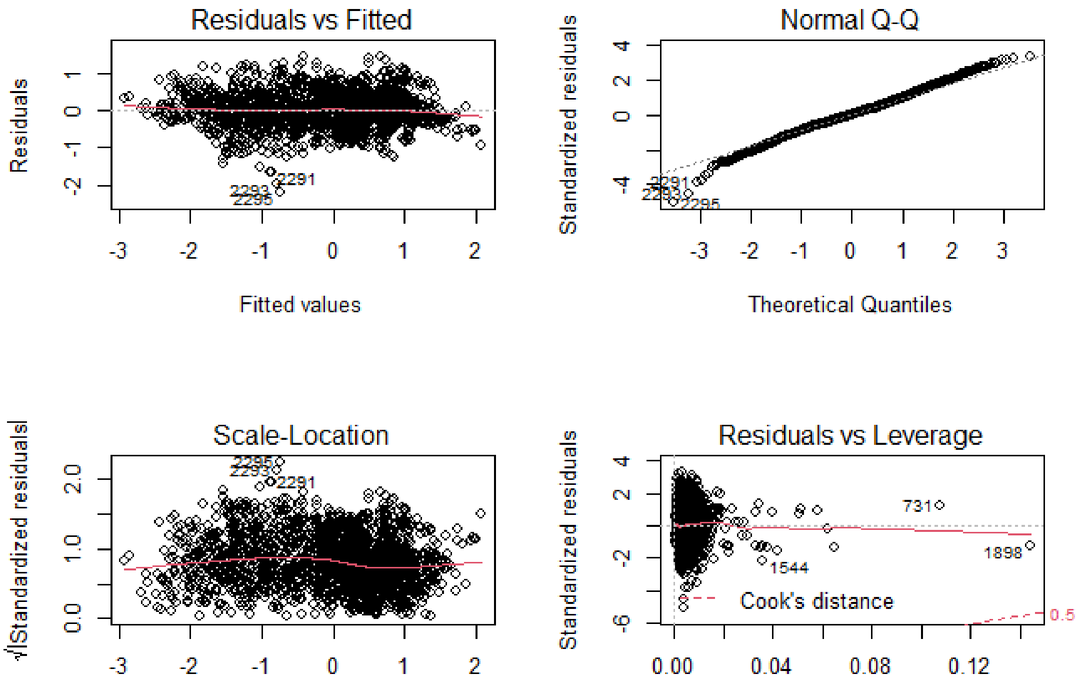
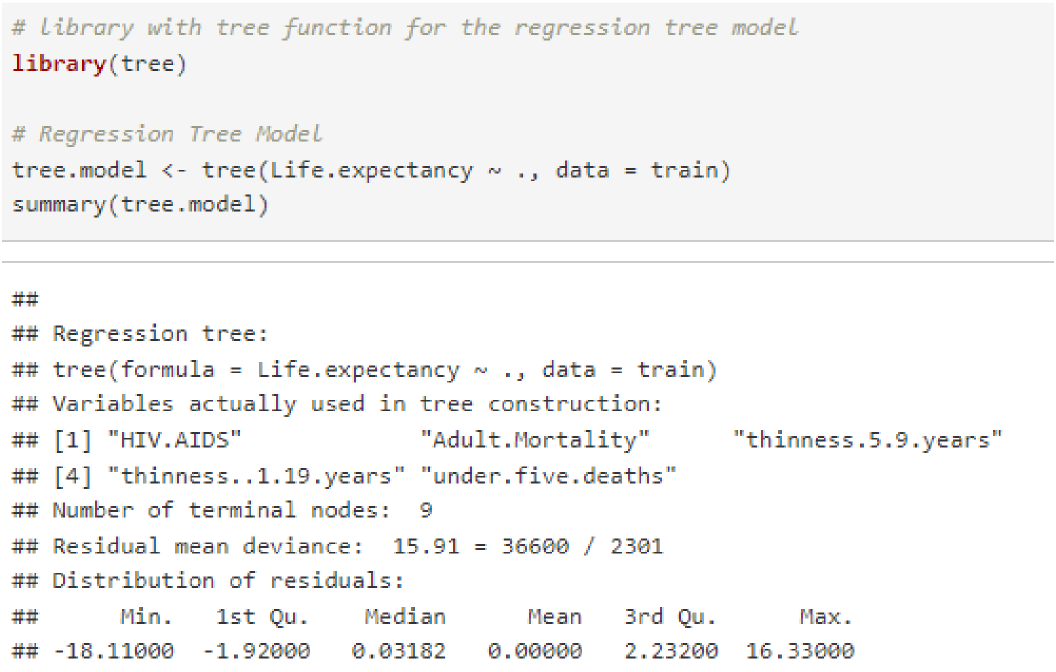


Figure 5a



```
# library with tree function for the regression tree model
library(tree)

# Regression Tree Model
tree.model <- tree(Life.expectancy ~ ., data = train)
summary(tree.model)
```

```
##
## Regression tree:
## tree(formula = Life.expectancy ~ ., data = train)
## Variables actually used in tree construction:
## [1] "HIV.AIDS"             "Adult.Mortality"      "thinness.5.9.years"
## [4] "thinness..1.19.years" "under.five.deaths"
## Number of terminal nodes:  9
## Residual mean deviance:  15.91 = 36600 / 2301
## Distribution of residuals:
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -18.11000  -1.92000   0.03182   0.00000   2.23200  16.33000
```

Figure 5b

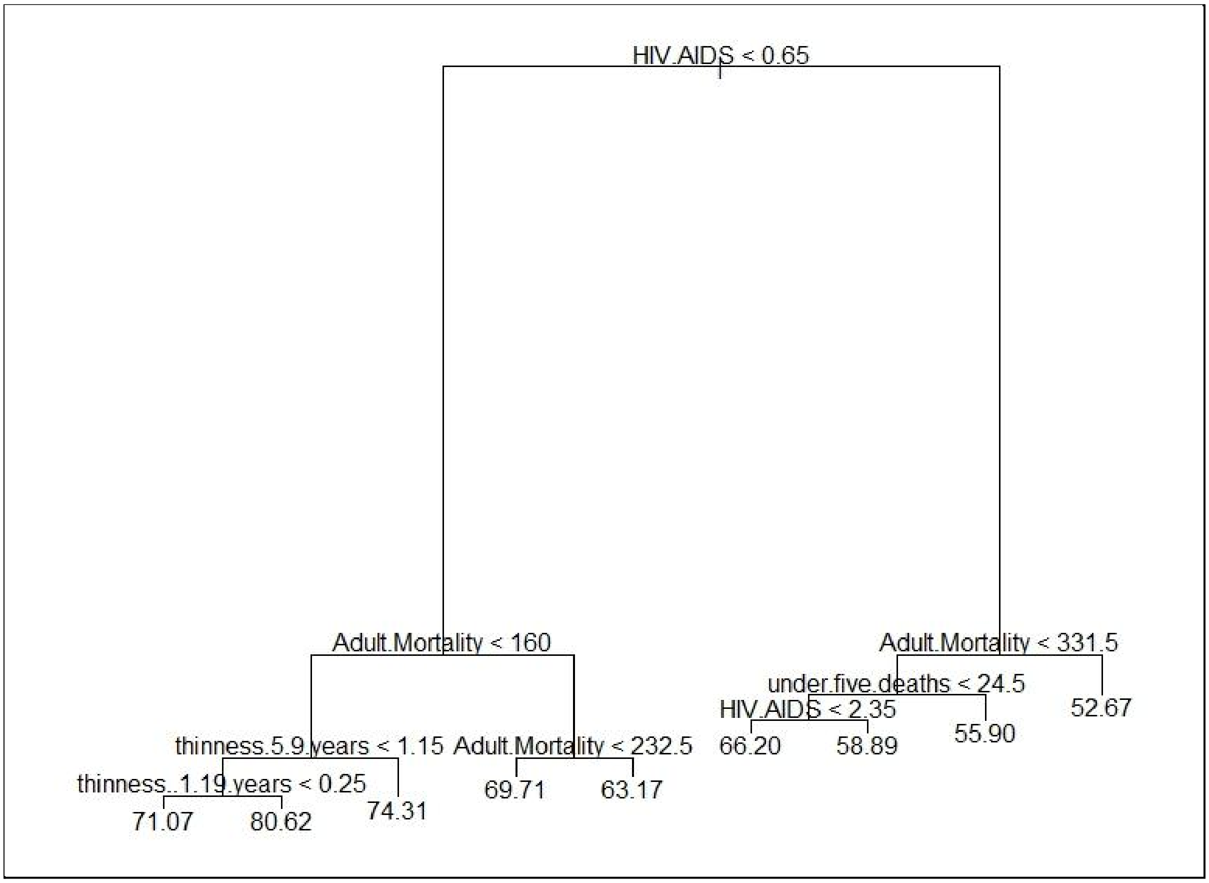
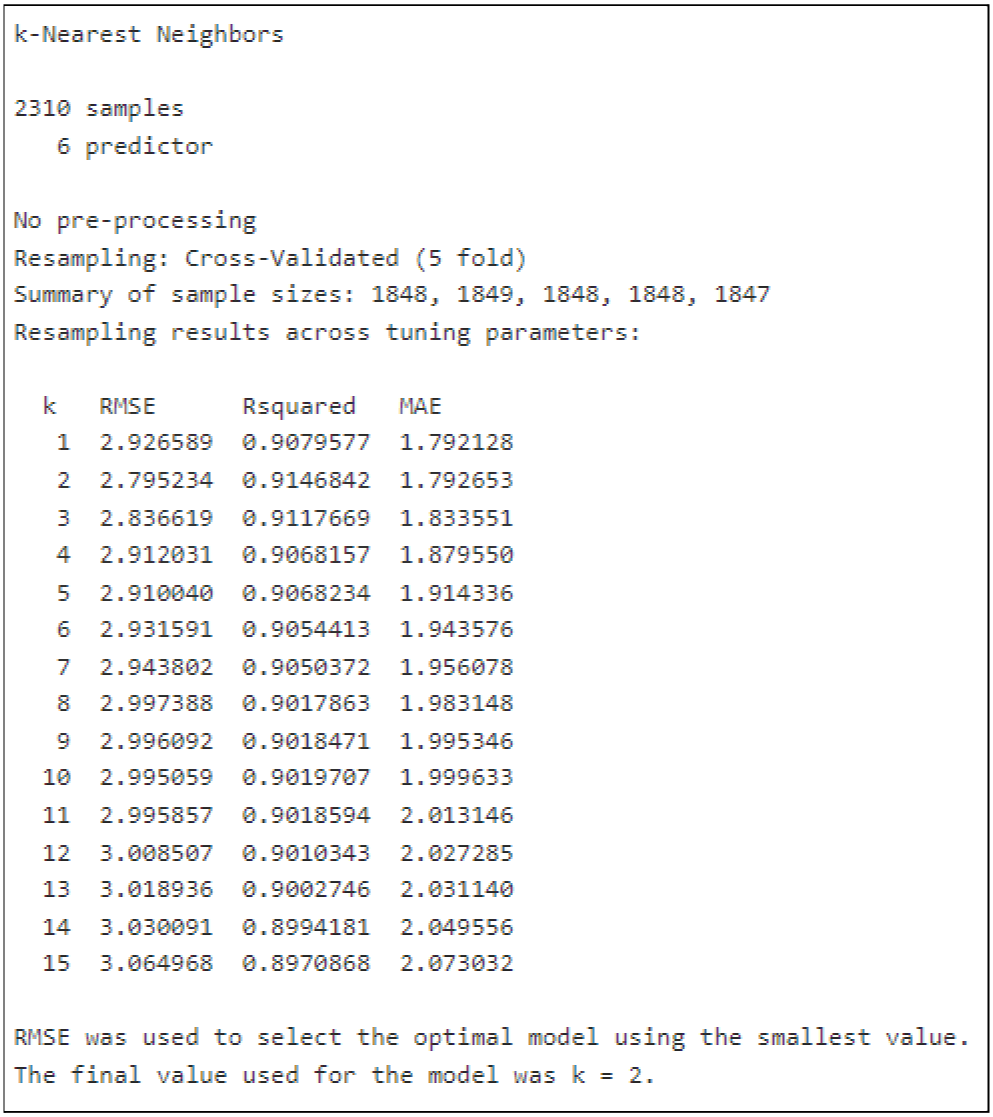


Figure 5c



```
k-Nearest Neighbors

2310 samples
   6 predictor

No pre-processing
Resampling: Cross-Validated (5 fold)
Summary of sample sizes: 1848, 1849, 1848, 1848, 1847
Resampling results across tuning parameters:

  k   RMSE      Rsquared   MAE     
   1  2.926589  0.9079577  1.792128
   2  2.795234  0.9146842  1.792653
   3  2.836619  0.9117669  1.833551
   4  2.912031  0.9068157  1.879550
   5  2.910040  0.9068234  1.914336
   6  2.931591  0.9054413  1.943576
   7  2.943802  0.9050372  1.956078
   8  2.997388  0.9017863  1.983148
   9  2.996092  0.9018471  1.995346
  10  2.995059  0.9019707  1.999633
  11  2.995857  0.9018594  2.013146
  12  3.008507  0.9010343  2.027285
  13  3.018936  0.9002746  2.031140
  14  3.030091  0.8994181  2.049556
  15  3.064968  0.8970868  2.073032

RMSE was used to select the optimal model using the smallest value.
The final value used for the model was k = 2.
```