**FINAL PROJECT REPORT**

**BY**

**Mayank Dubey**

**DEPARTMENT OF DATA ANALYTICS ENGINEERING**



**Introduction:**

According to **WHO**, suicide is the act of killing oneself. The major cause of suicide is mental disorder. Every year approximately 1 million people take their life.

In our project we will do age wise comparison of male and female. Countries where suicide rate is high. Countries where suicide rate is low. Year wise trend in suicide rate and population. We are querying the data to understand the underlying relationship of causes of suicide rate.

A screenshot of a cell phone

Description automatically generated

Figure – 1

The graph shows the suicide rate in different age group in the male and the female. We can depict from this as the age group increases the suicide rate also increases. The male has higher suicide rate than female.

A screenshot of a cell phone

Description automatically generated

Figure – 2

The graph illustrates, the top 9 countries where average suicide rate is high. The Russian Federation is highest followed by Hungary, Ukraine etc.

A screenshot of a cell phone

Description automatically generated

Figure – 3

The graph shows the countries where average suicide rate is lowest among 9 countries. Jamaica being the least.

A close up of a map

Description automatically generated

Figure – 4

The graph shows the year wise trend of suicide rate. With every passing year the suicide rate is decreasing.

A close up of a map

Description automatically generated

Figure – 5

The graph shows, the year wise trend in the population. With the year passing population increases.

A close up of a map

Description automatically generated

Figure – 6

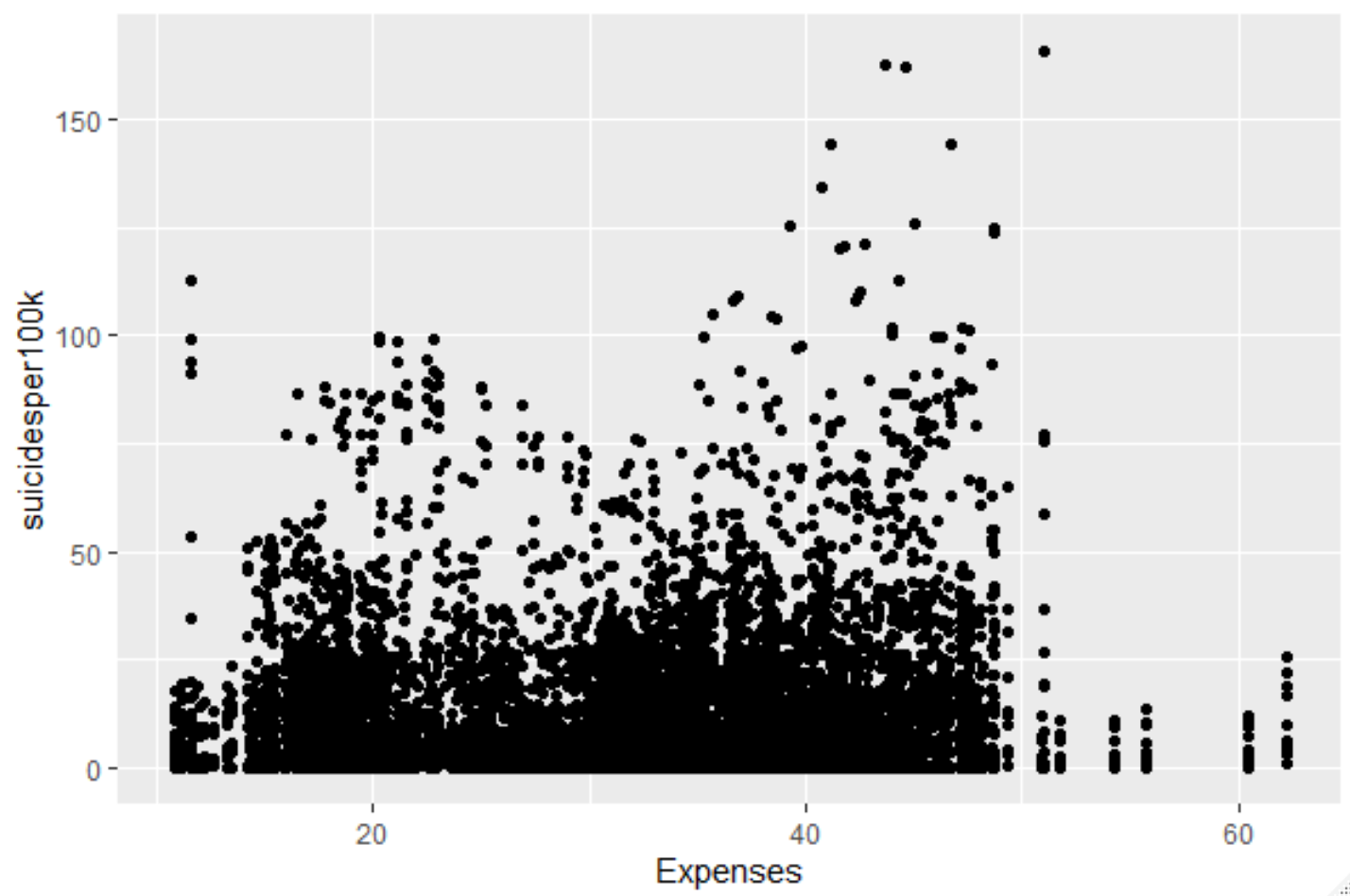
The unemployment rate till 2006 is decreasing, then after there is sudden increasing in unemployment rate.

**Data Preprocessing:**

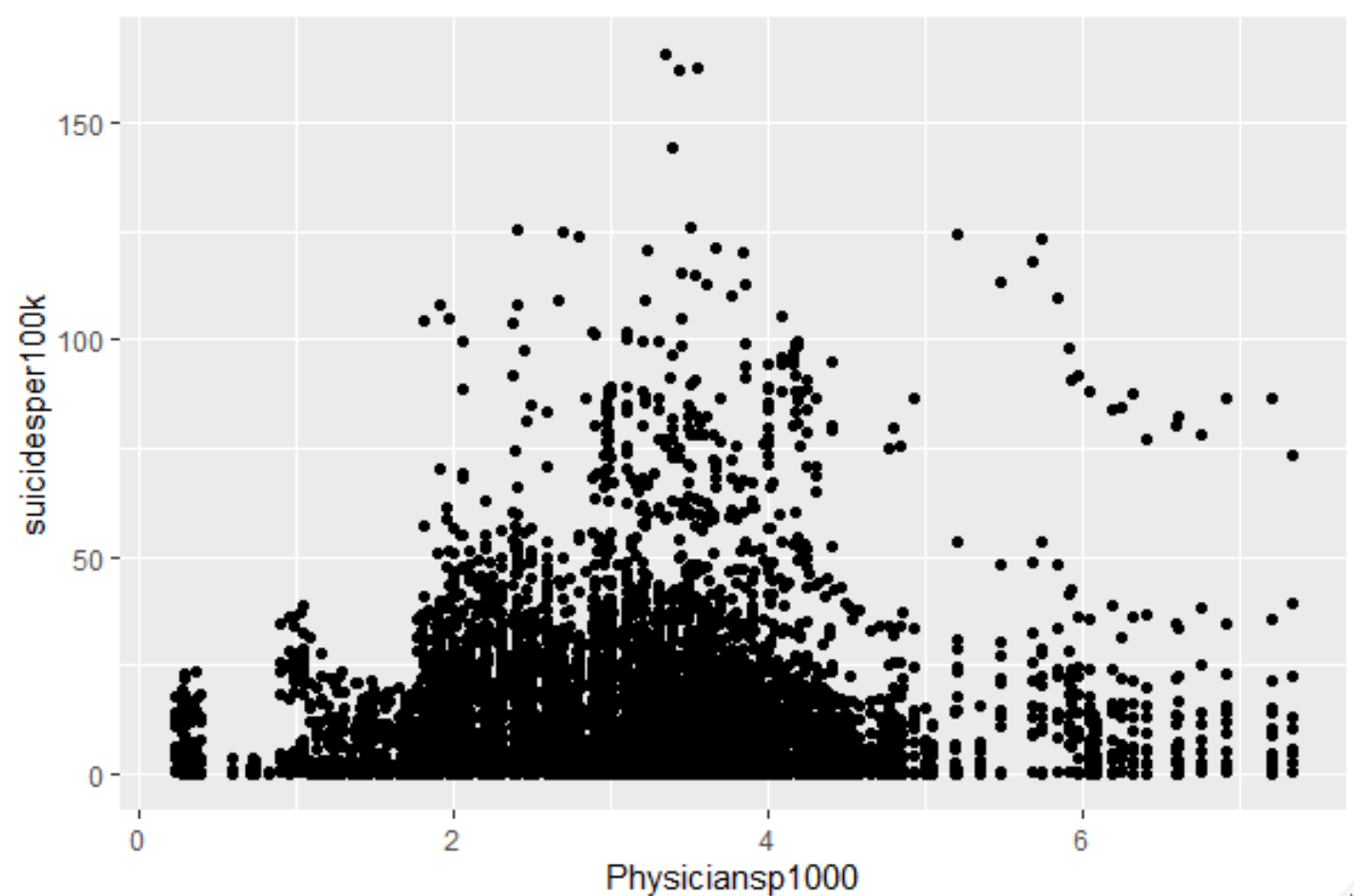
We took data from 1995-2013 as there are lots of data missing from rest of the years. It has 26 columns and have renamed those for better understanding and interpretation. We are using 16 columns of all the available features and removed the rest based on the below graph which shows the percentage of missing data column wise. Also imputed some missing data as they have some significance towards the prediction.

**Columns removed are:**

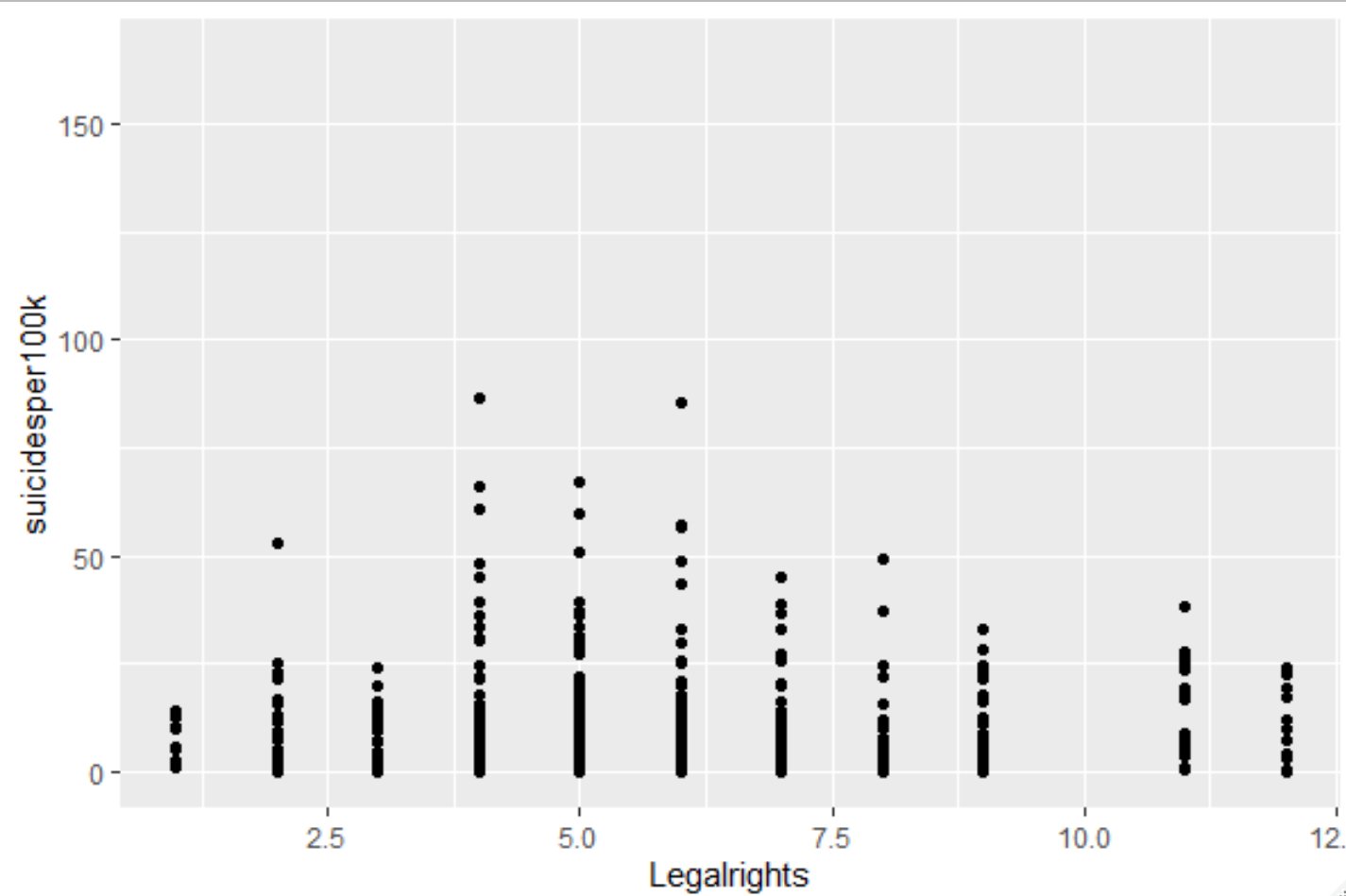
* Suicide\_no: We have chosen **Suicideper100k** as our response variable, so we have removed this column.
* Country-year: This column was not required as we already have taken **country** and **year** already separately.
* yearlyHDI: It has almost 65% missing values, which is way more and we should not impute this large amount of data to predict the values.
* suicide%: We have chosen **Suicideper100k** as our response variable, so we have removed this column as well like Suicide\_no.
* Expenses: Though expenses have only around 10% of missing values we removed it as we found that there is little correlation with our response variable i.e. **suicideper100k**.



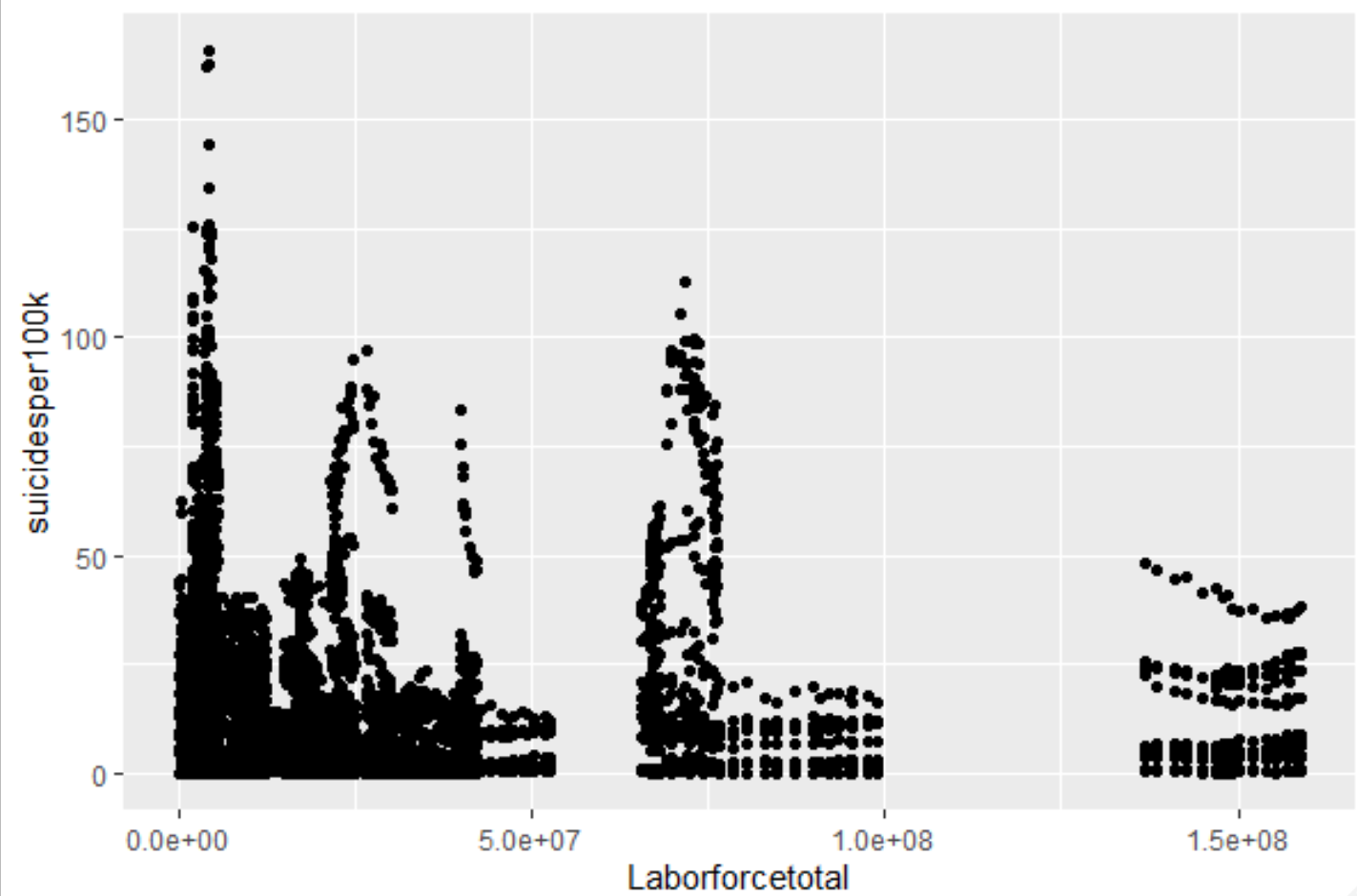
* Physiciansp1000: It has 15.5% of missing data and also have little correlation with our response variable i.e. **suicideper100k**.



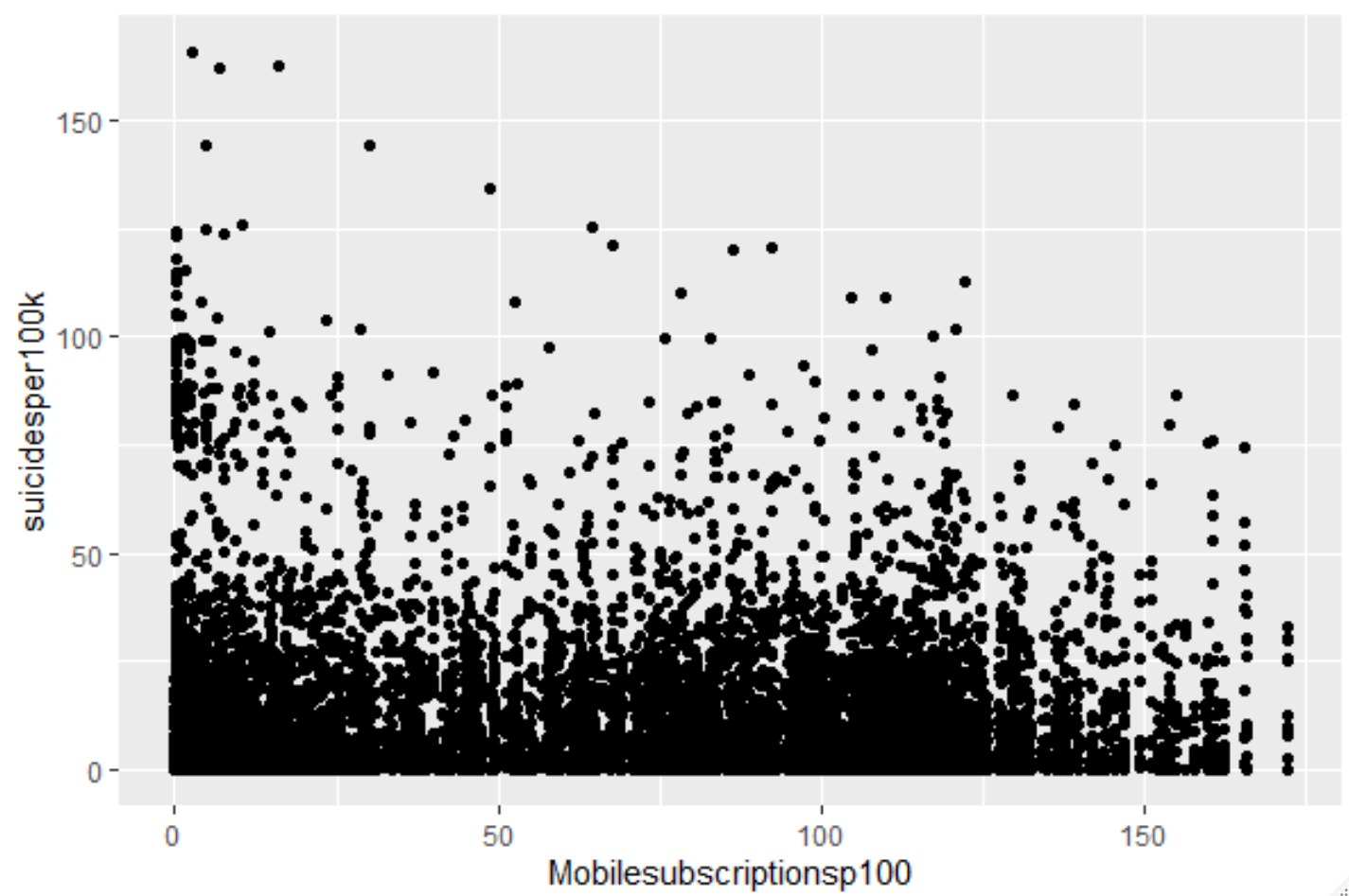
* Legalrights: It has almost 63% of missing data so have removed the column for better model development and also has no correlation with the response variable.



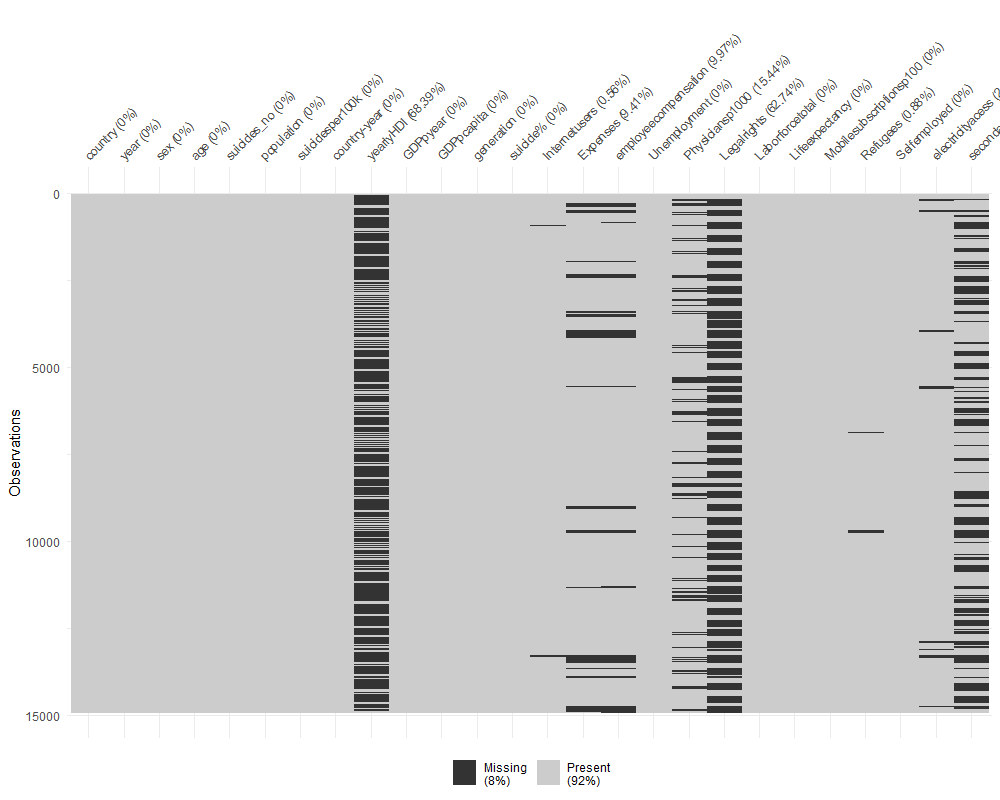
* Laborforcetotal: It has very less correlation (0.03) with suicideper100k.



* Mobilesubscriptionsp100: It has a correlation of -0.05 with suicideper100k.



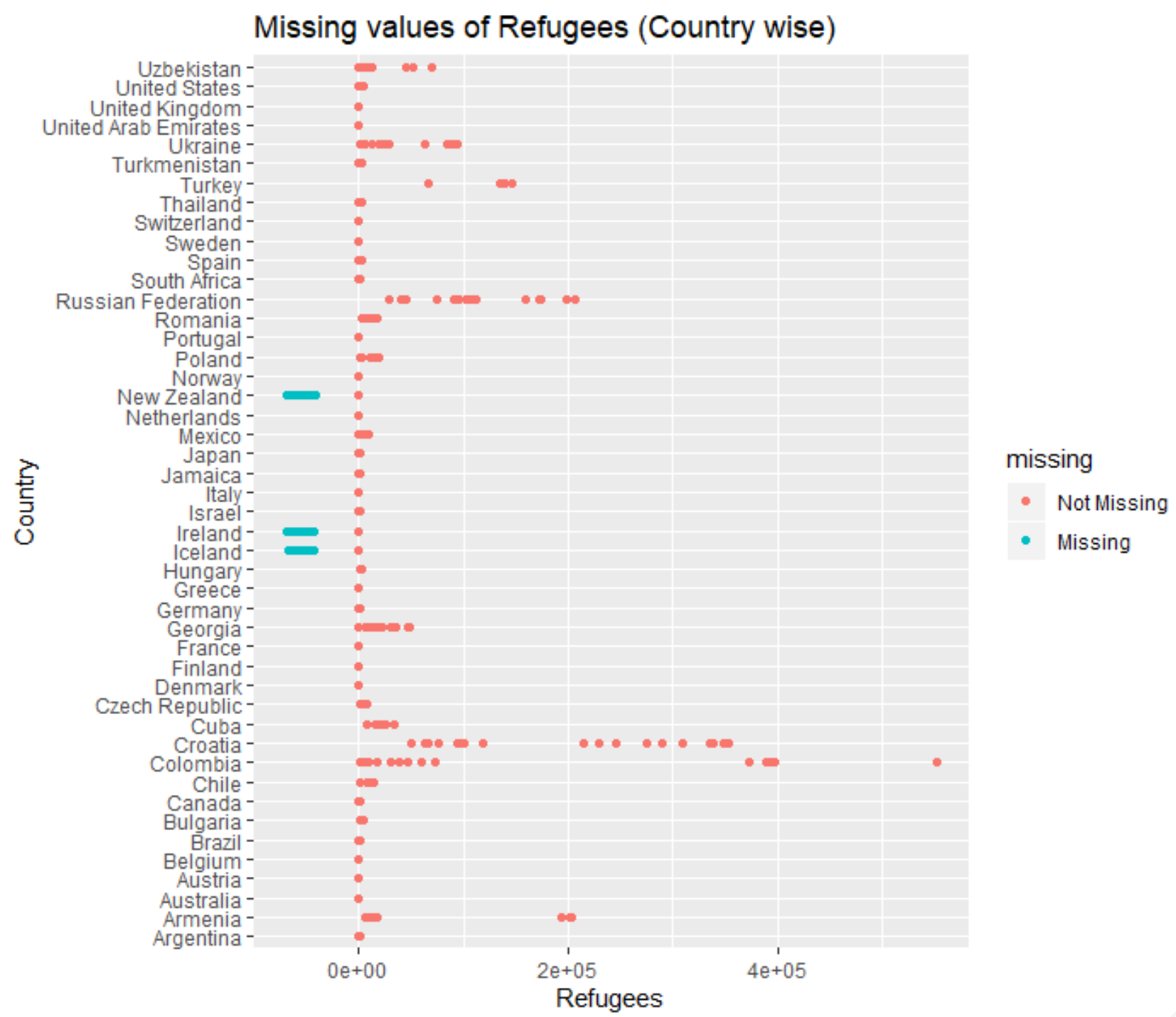
* Secondarycompletion: It has very large amount of missing data, so this had to be removed from the dataset.



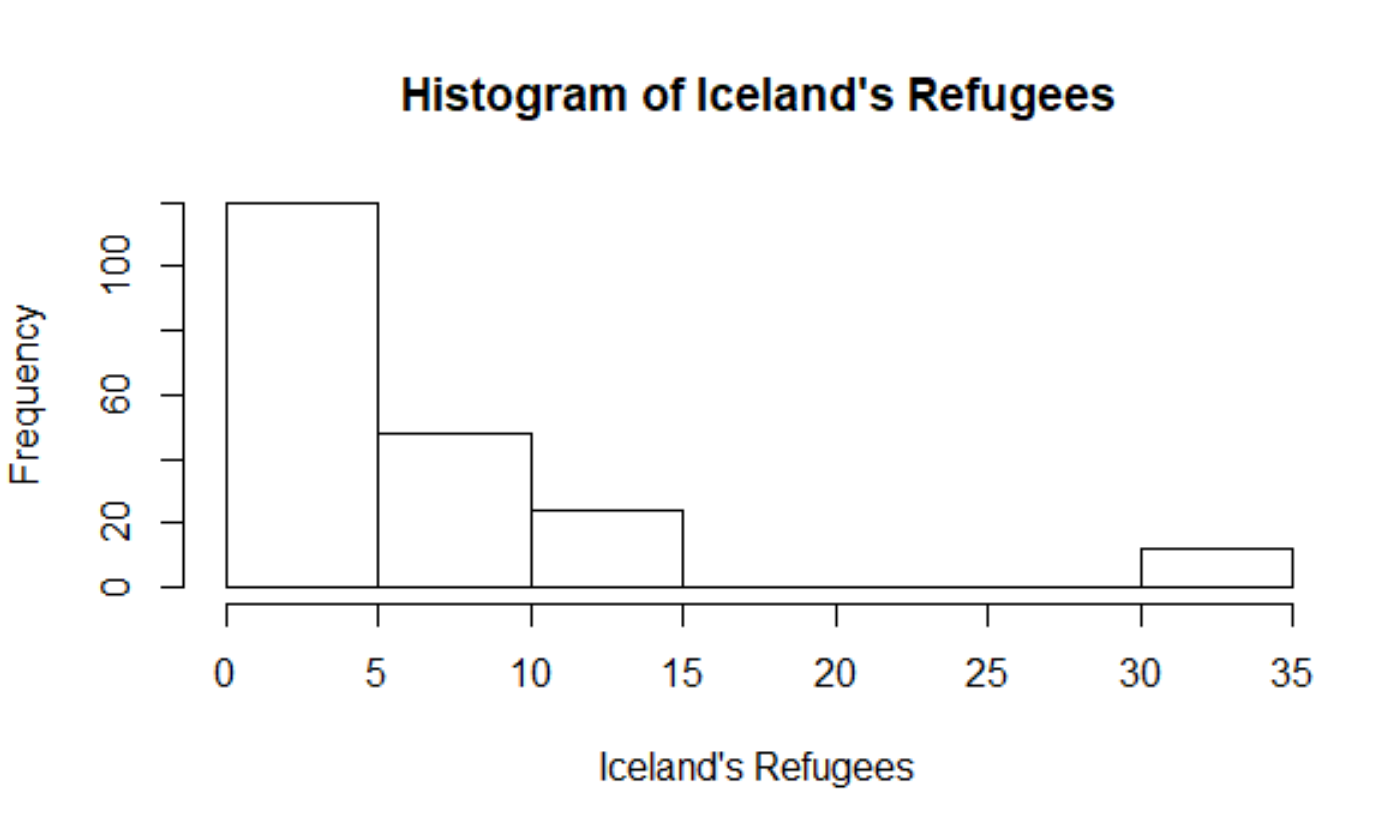
**This shows the columns which has missing values from the dataset.**

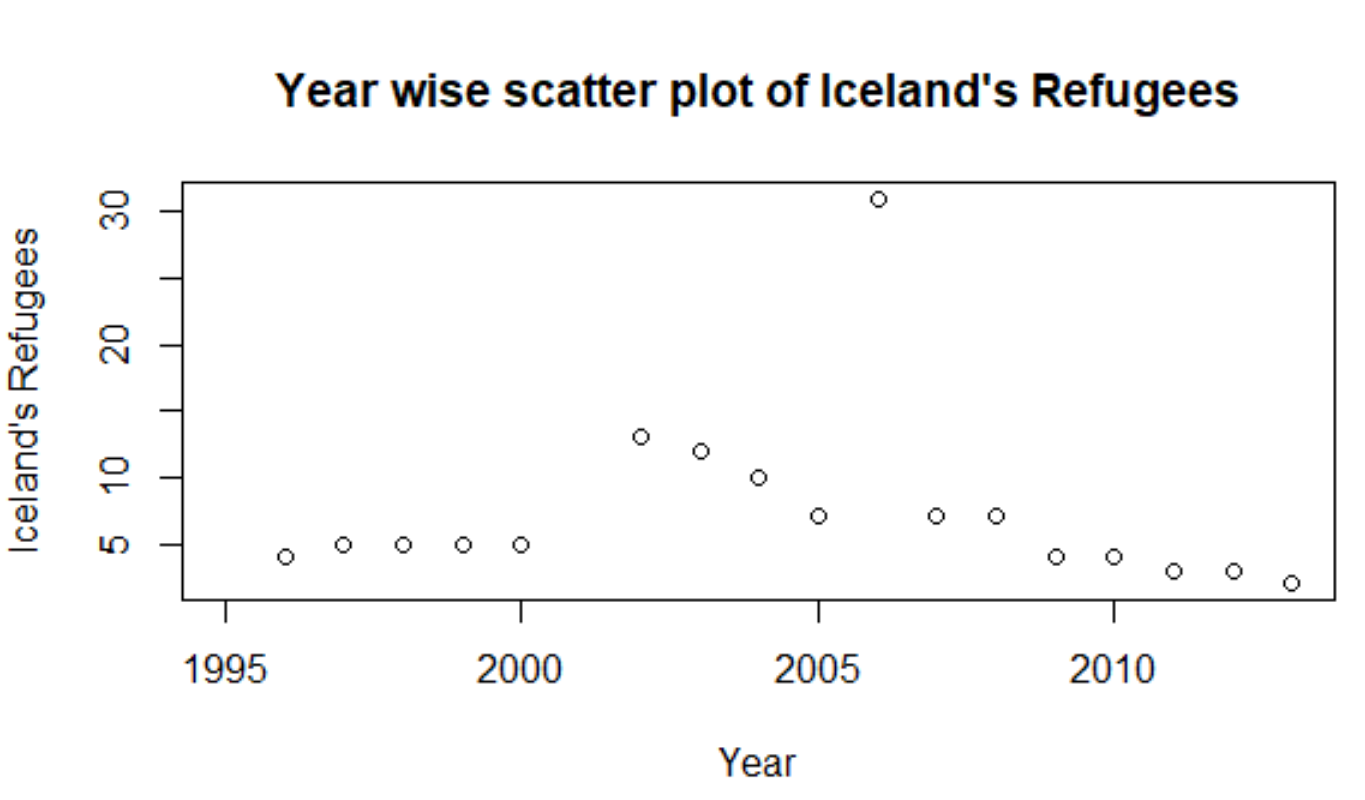
**Data analysis and Imputation of columns:**

* **Refugees:** There are some missing data in 3 countries namely Ireland, Iceland and New Zealand. So instead of imputing the mean or median, we choose to analyze the data further based on the respective countries data and then imputing them with some appropriate values.

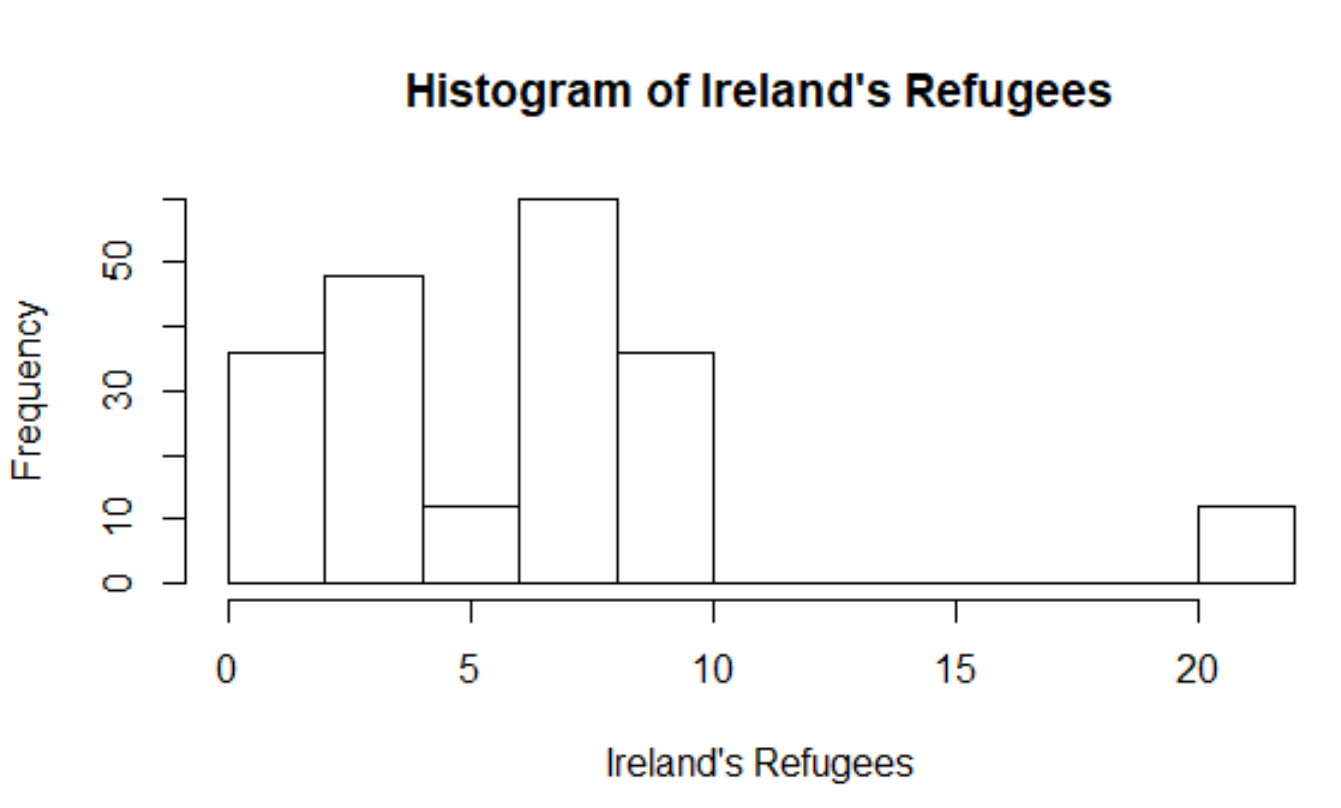


* **Analysis of Iceland’s refugee data:** The data is missing for the years 1995 and 2001. The below histogram shows that the data is right skewed, and median would be good to use as an imputation values for Iceland’s refugees.



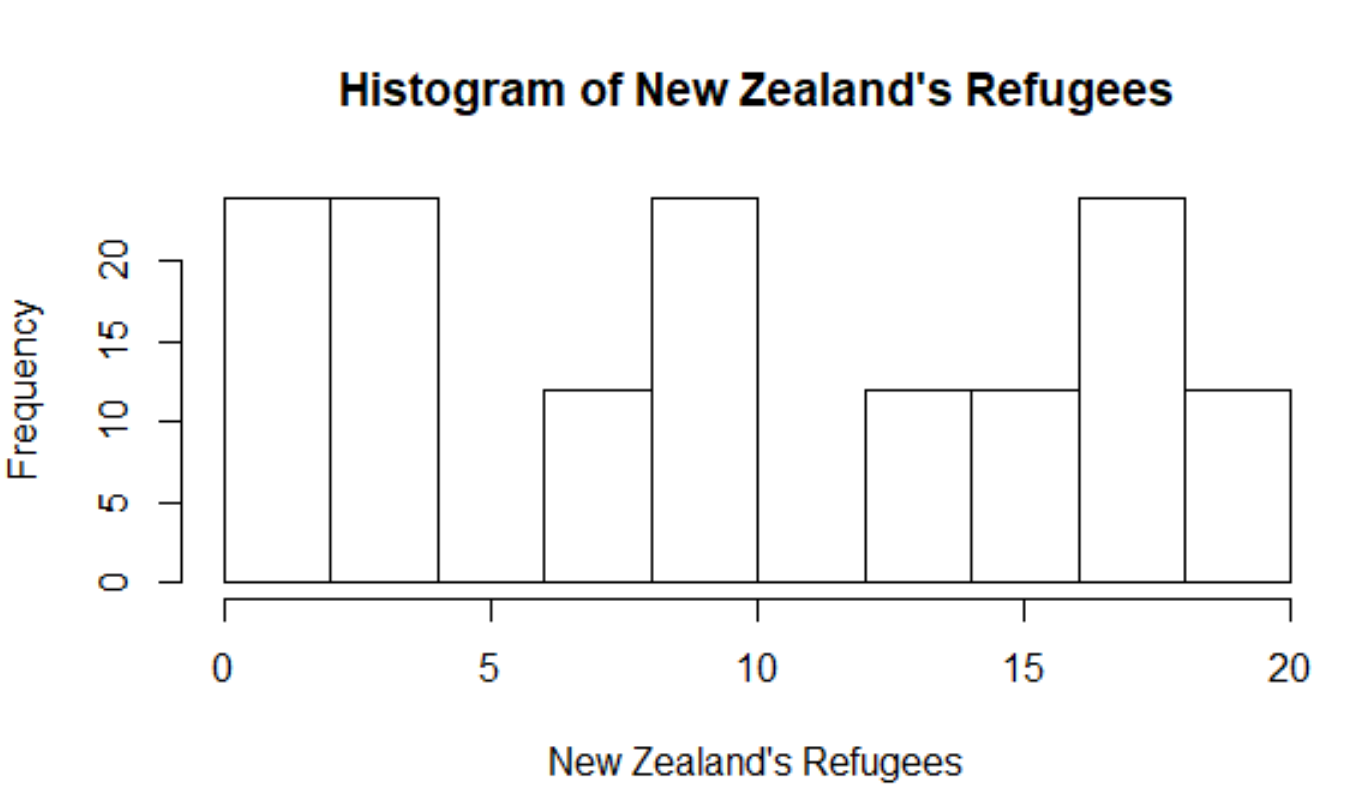


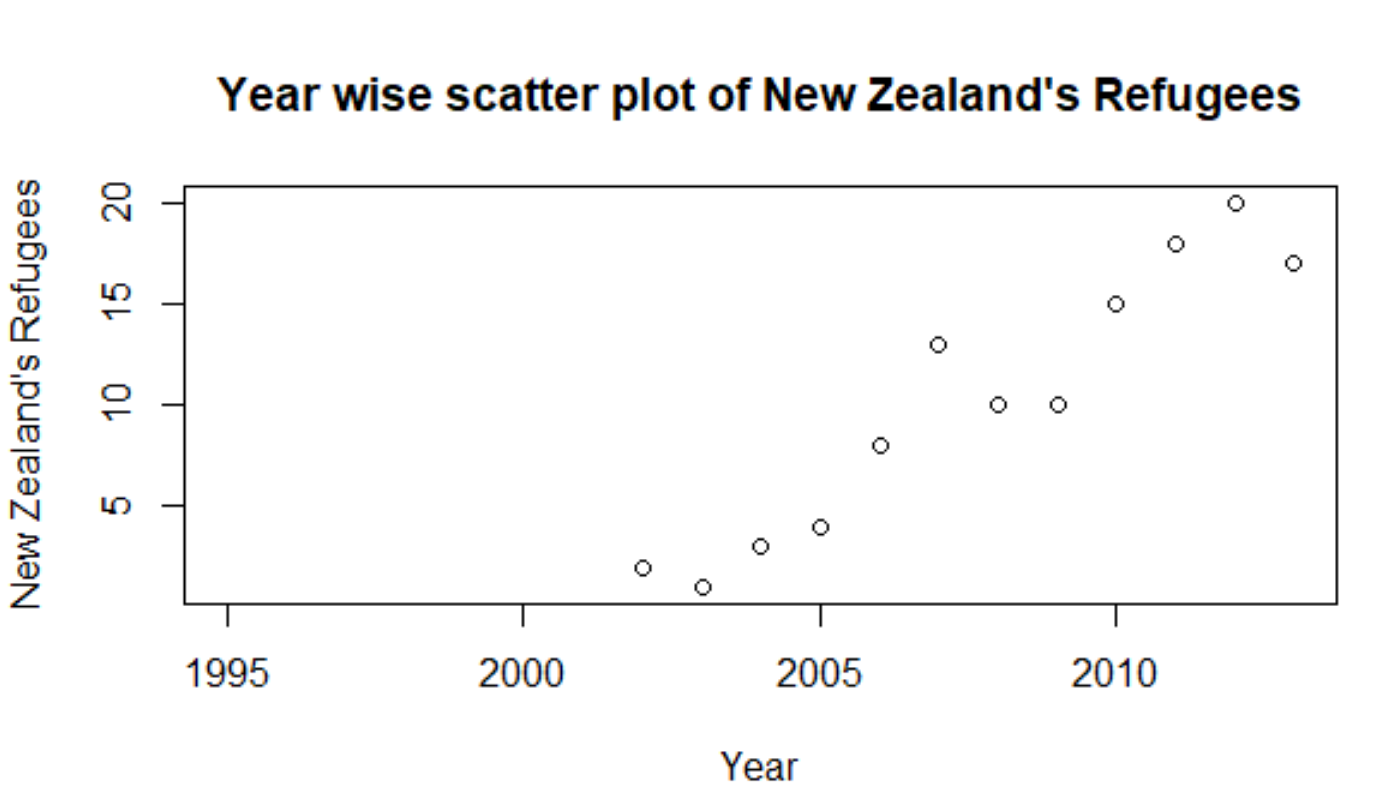
* **Analysis of Ireland’s refugee data:** Here the refugee data is missing for the years 1998 and 1999. We have imputed the data based on the previous year’s data i.e. 2.



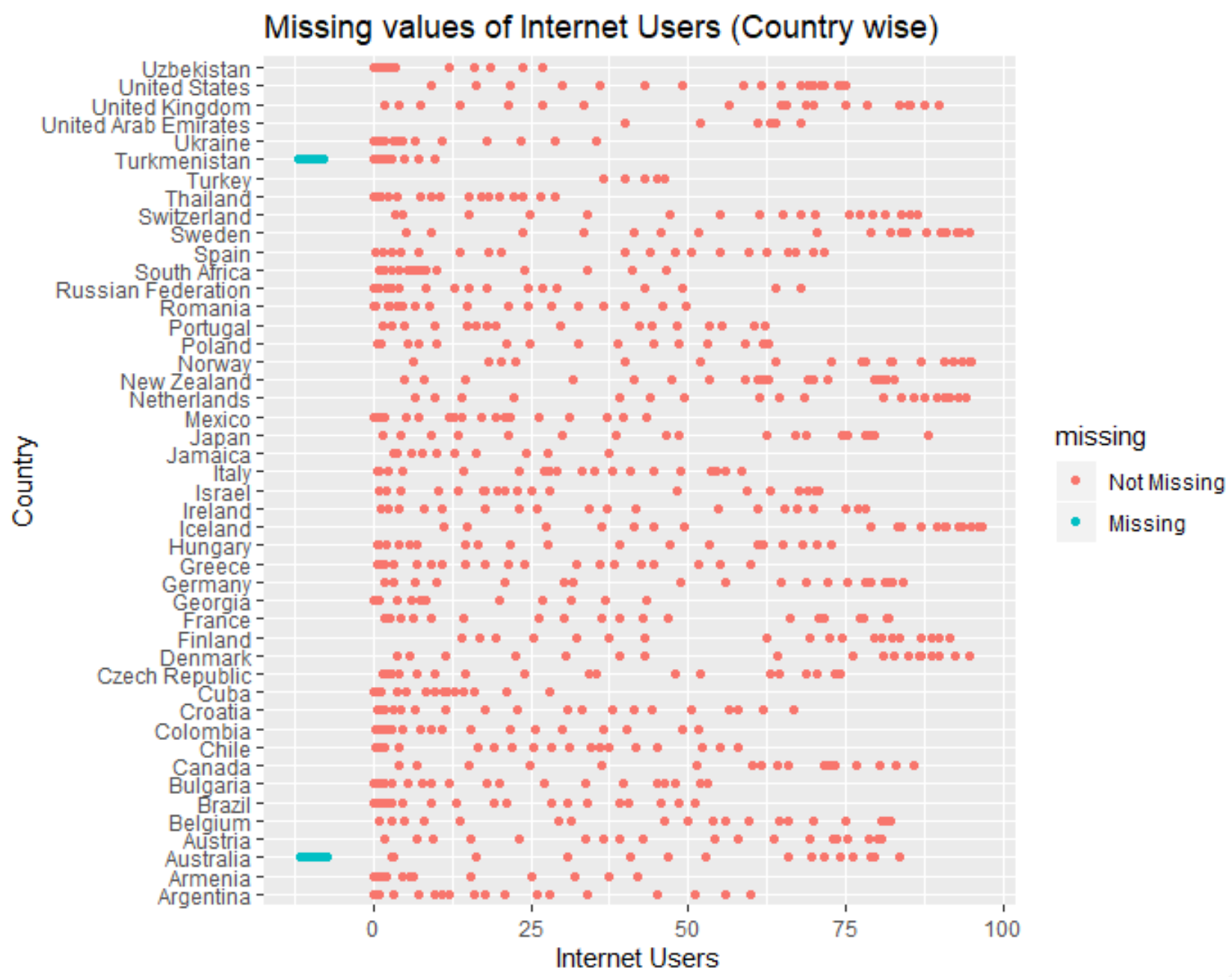


* **Analysis of New Zealand’s refugee data:** Much of data are missing here for previous years i.e. from 1995-2001. Imputed the data analyzing previous years data. We have imputed the data based on the close by year’s data i.e. 1.

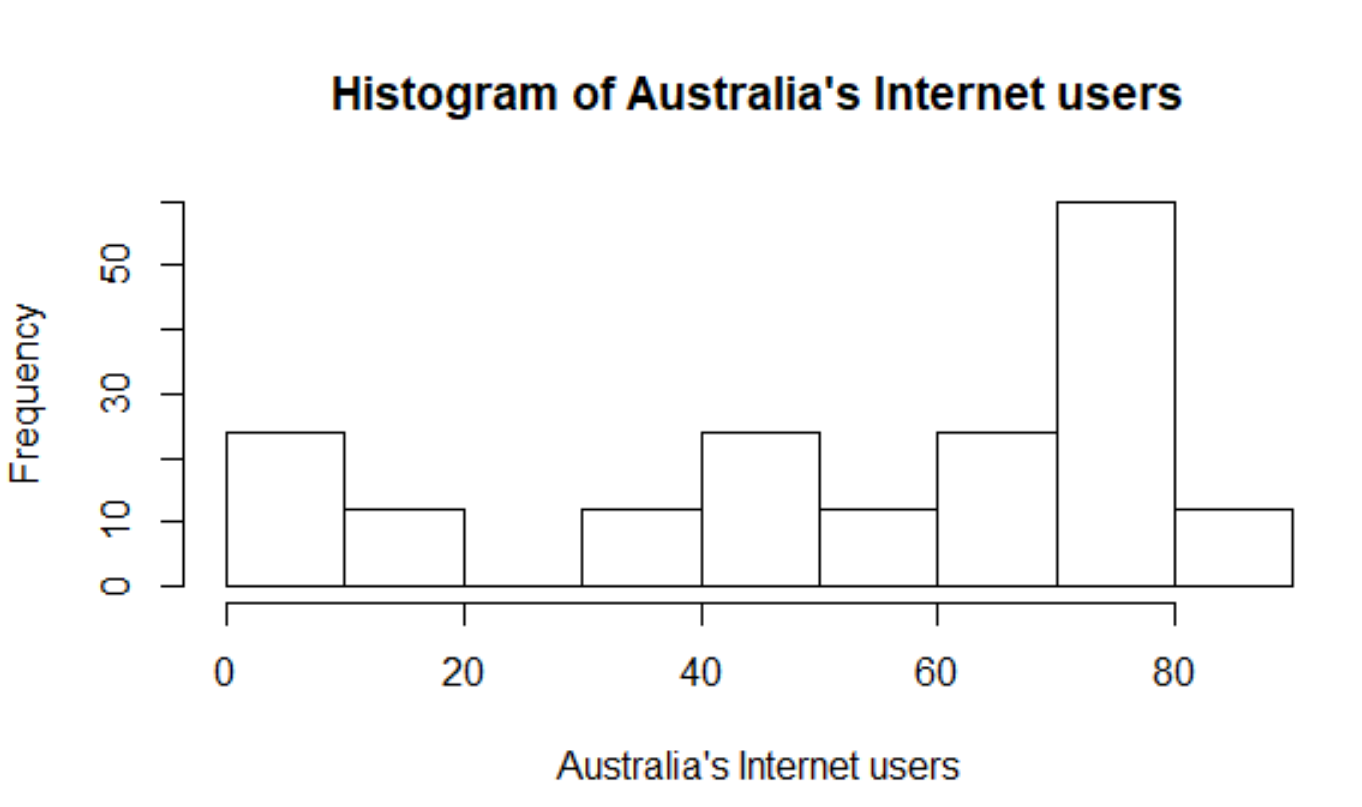


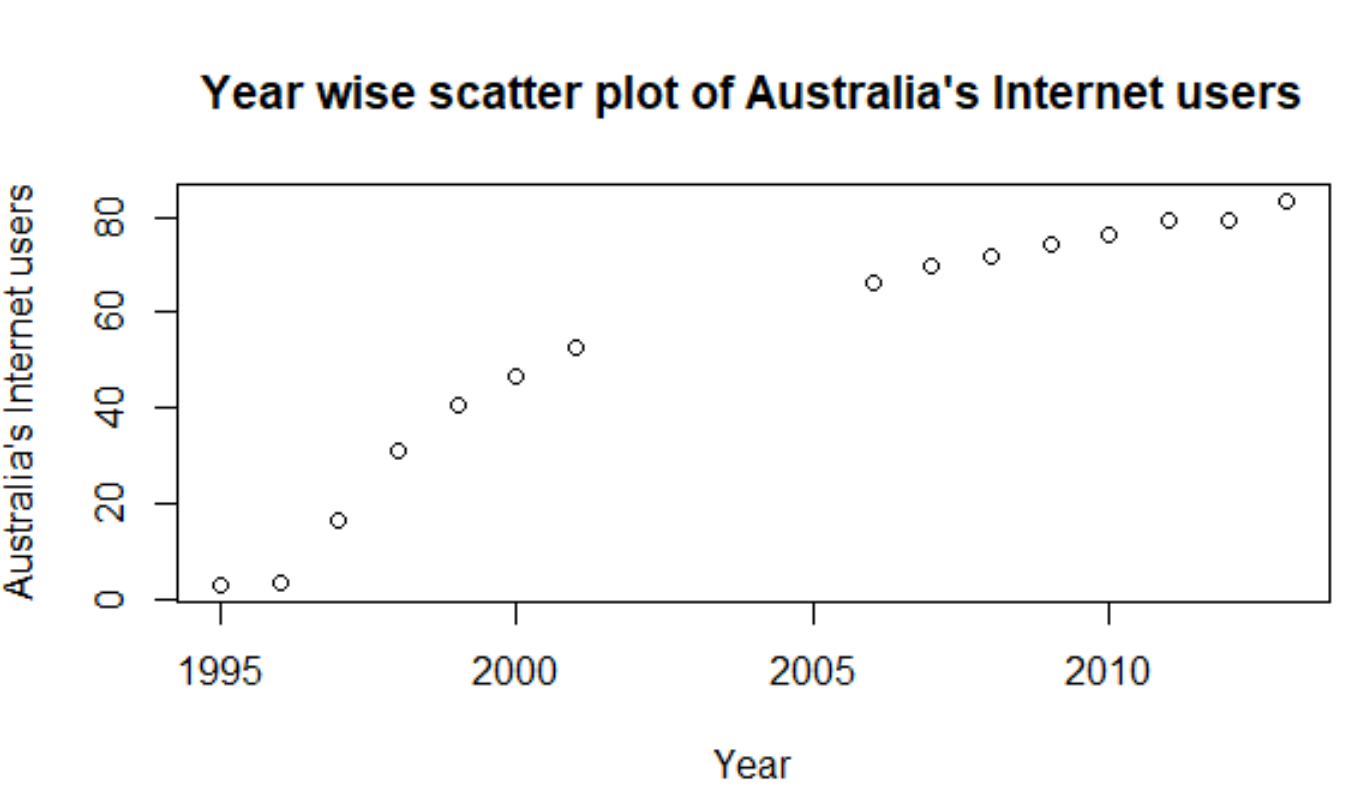


* **Internet User:** From the below plot we could see only Australia and Turkmenistan has missing data. So instead of imputing the mean or median, we choose to analyze the data further based on the respective countries data and then imputing them with some appropriate values.

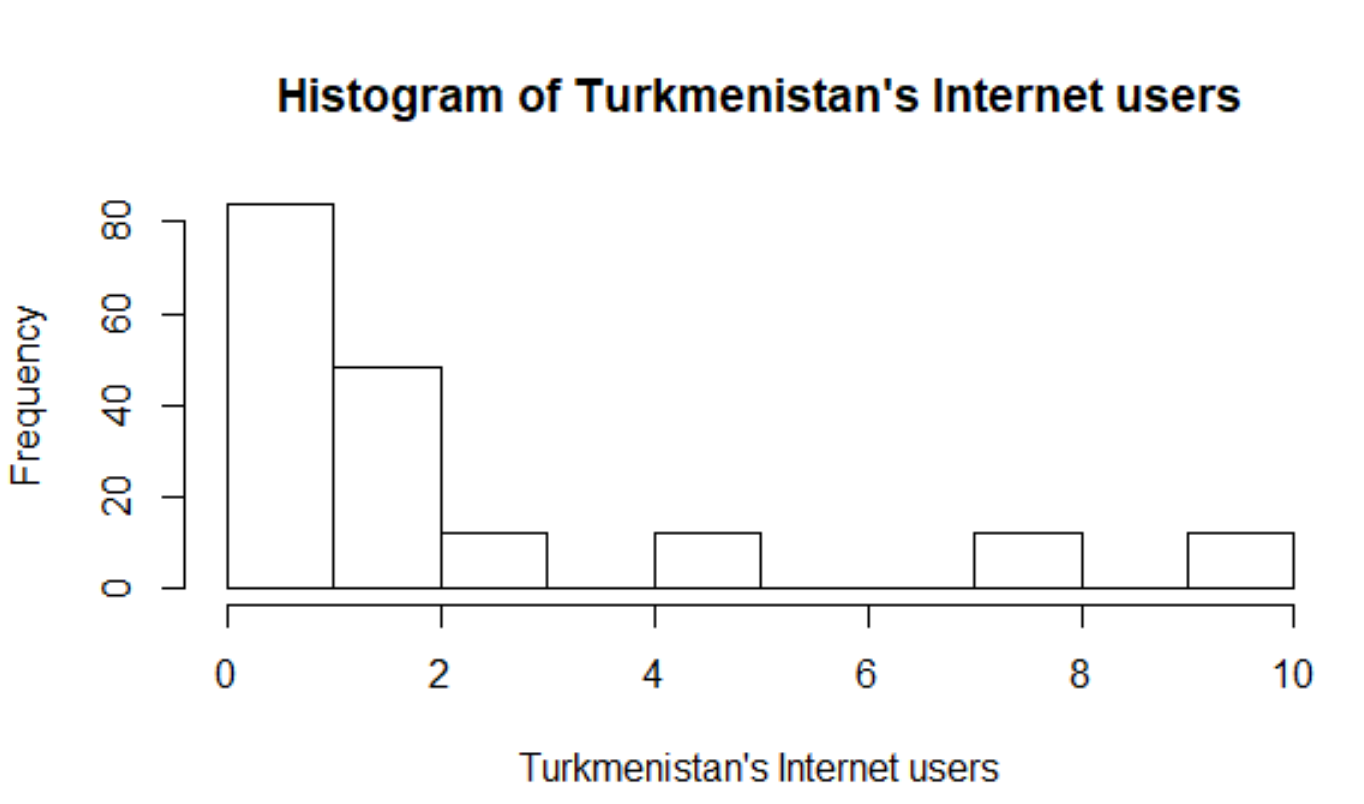


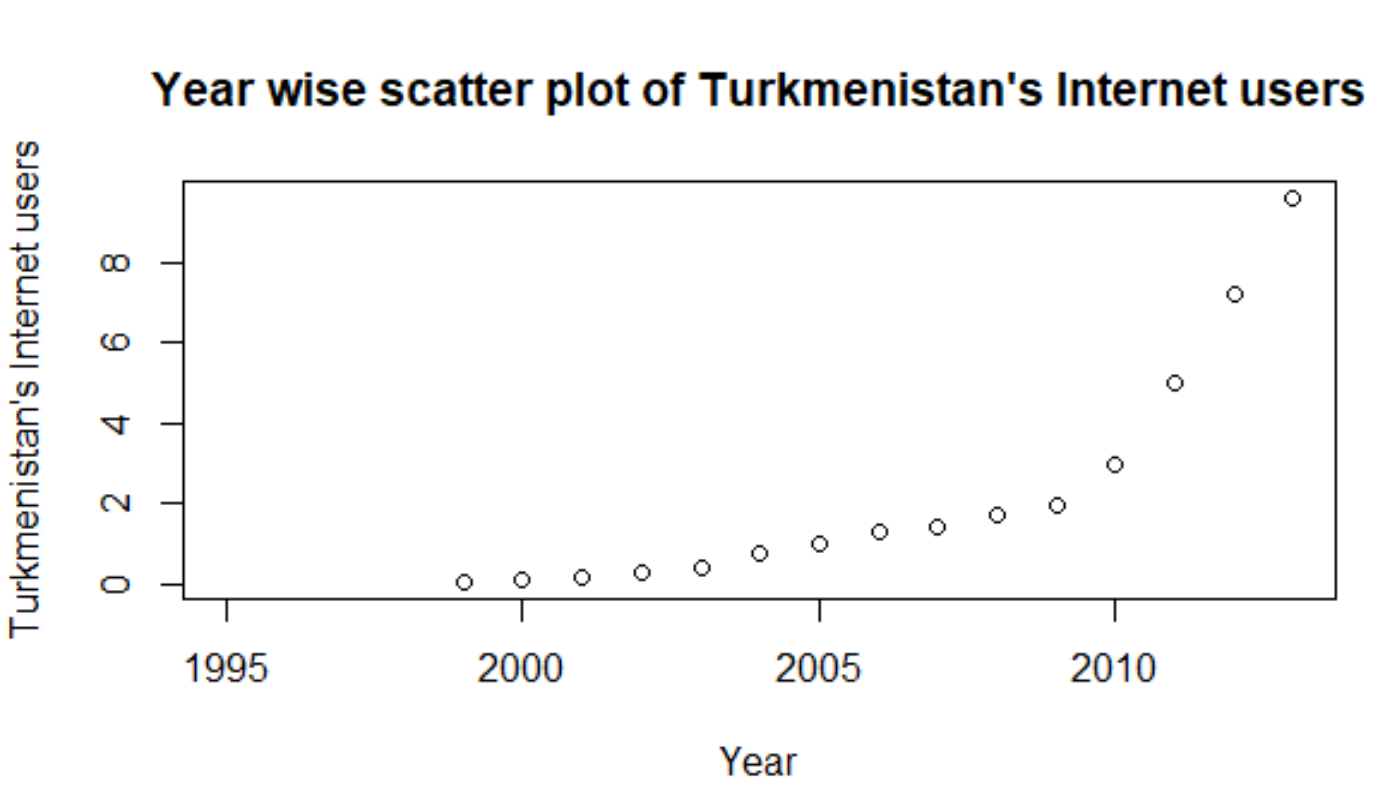
* **Analysis of Australia’s internet user data:** The data here is missing form 2002 -2005 which can be seen from the scatter plot below. The histogram plotted show that it is left skewed, due to this we have done the imputation with median.



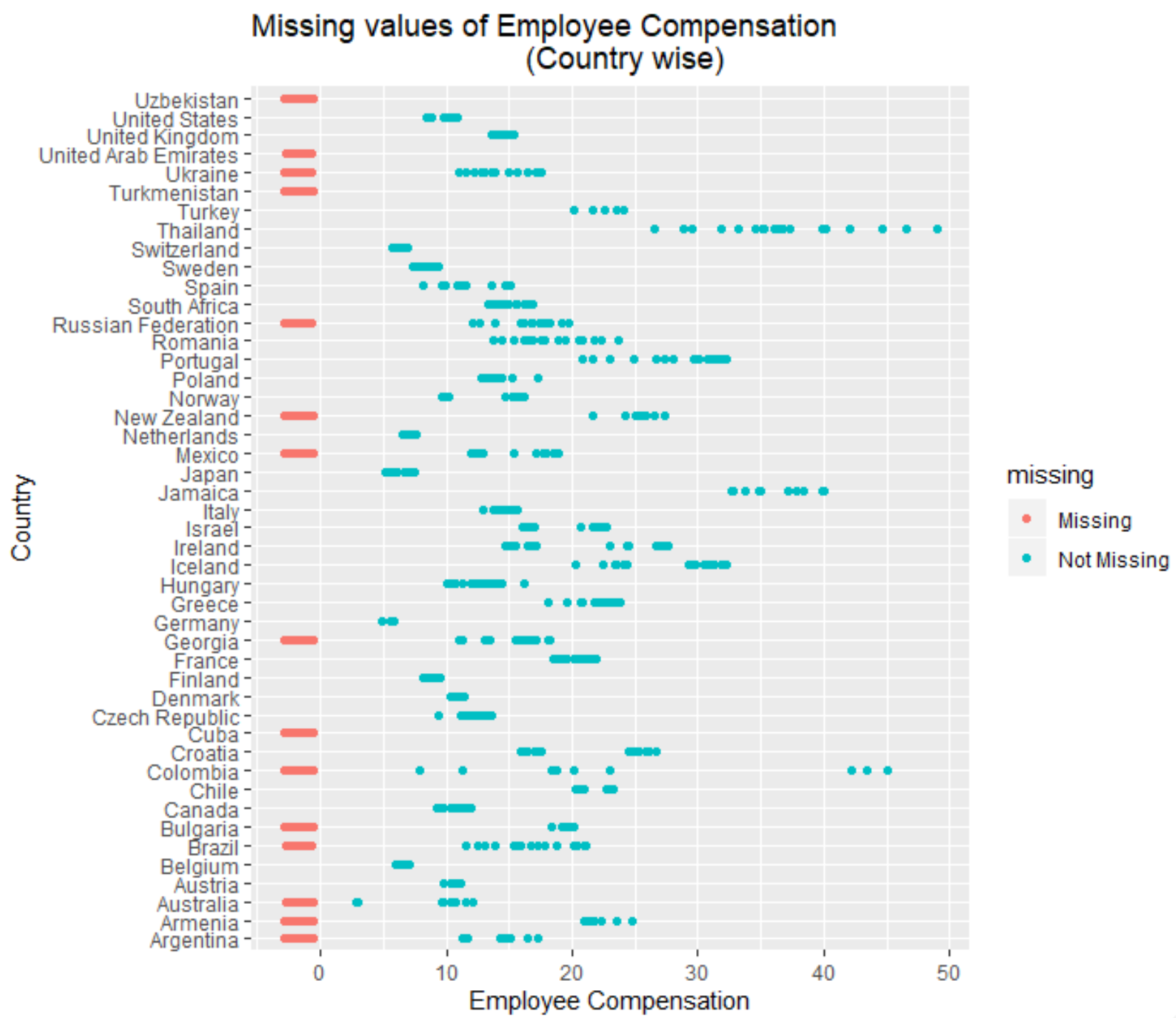


* **Analysis of Turkmenistan’s internet user data:** For Turkmenistan, internet user data is missing from 1995-1998. We have imputed with 0 as data after it also has 0 for some years and internet was not very popular as of today.



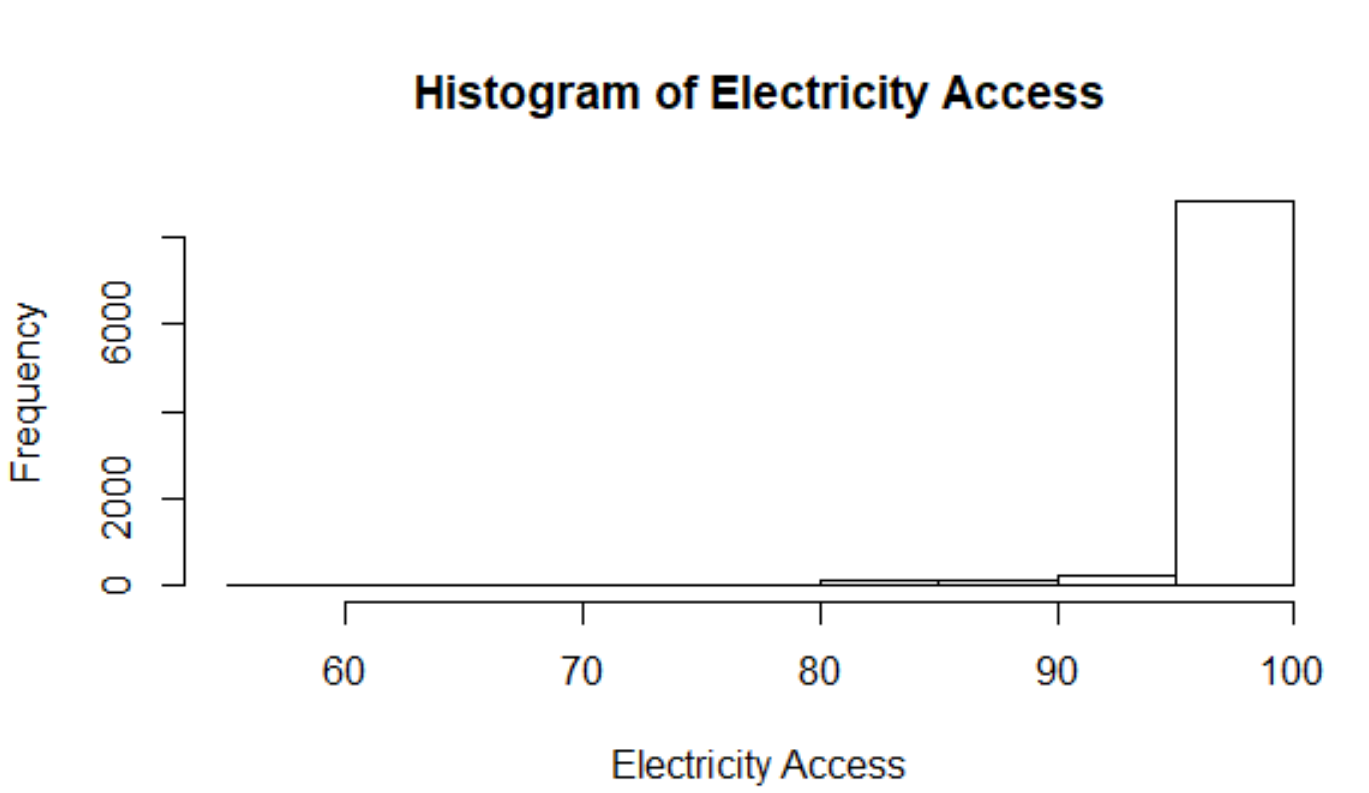
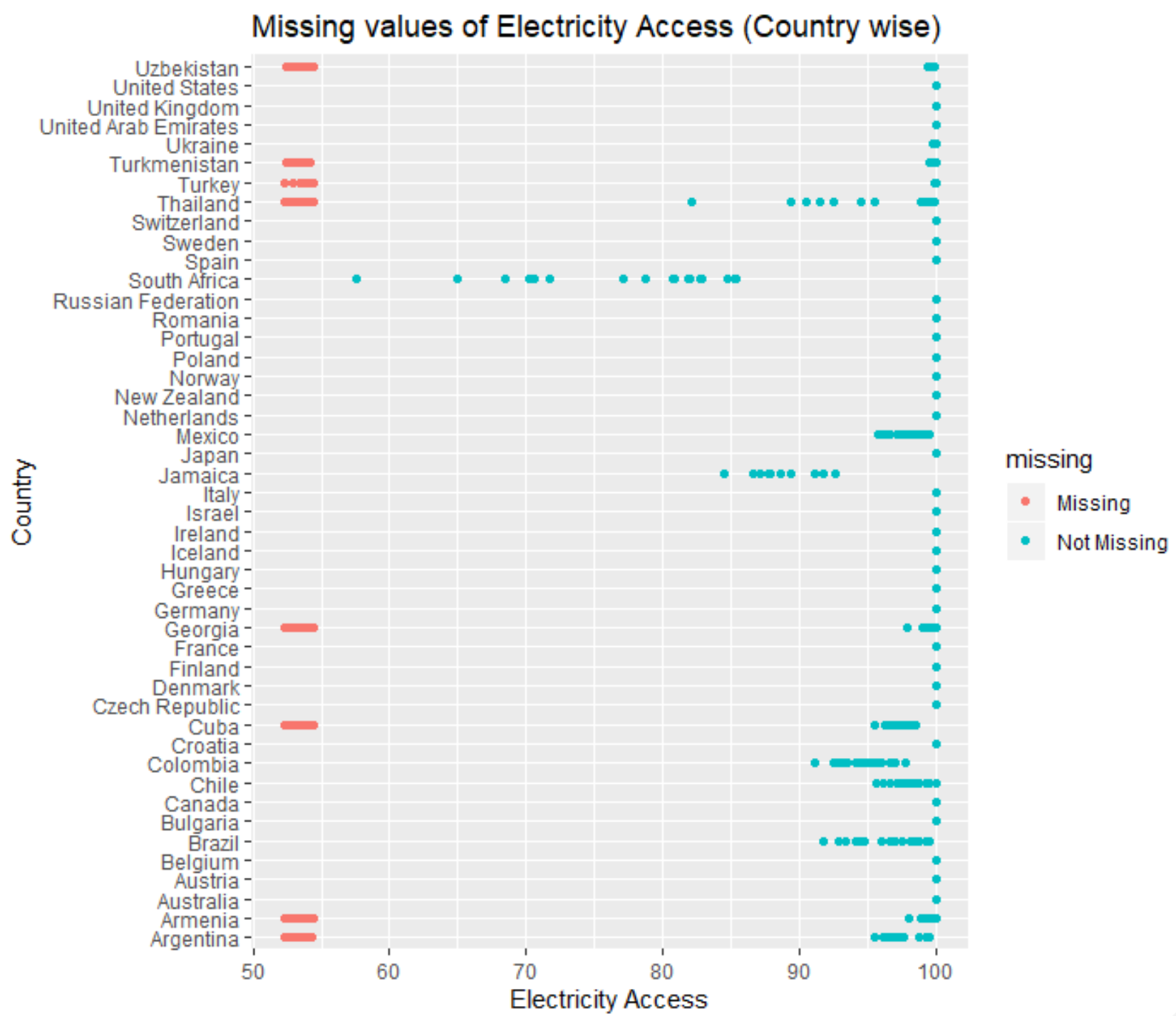


* **Employee compensation:** The Employee compensation data of several countries is missing from the dataset. As the distribution of the data is right skewed, we have imputed the missing values with median of rest of the data from very column.





* **Electricity Access:** This data is missing from 8 different countries and upon plotting the data distribution plot, it is left skewed. So, we have imputed the missing values with median.



**Model Development:**

The goal of the model is to predict risk of suicides in different countries, based on social and economic features.

**Initial steps done for fitting the model:**

**One Hot Encoding:**

**Why is one hot encoding required**:

Categorical data are variables that contain label values rather than numeric values.

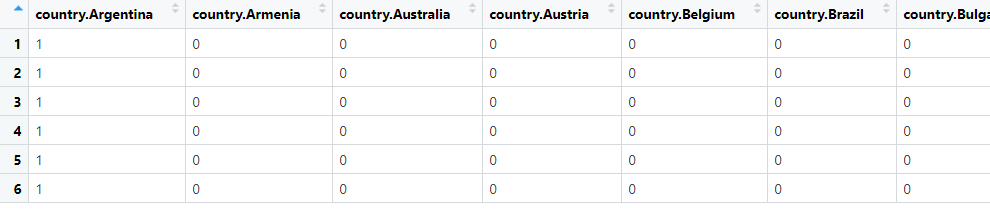
For categorical variables, which have natural ordering between categories, one hot encoding is required.

Else, it results in poor performance of the model.

What one hot encoding does is, it takes a column which has categorical data, which has been label encoded and then splits the column into multiple columns. The numbers are replaced by 1s and 0s, depending on which column has what value. In our example, we’ll get four new columns, one for each categorical variable.

For example, in the project dataset –

For rows which have first column value as “Argentina”, the “Argentina” column will have value 1 and the remaining columns will have value ‘0’.

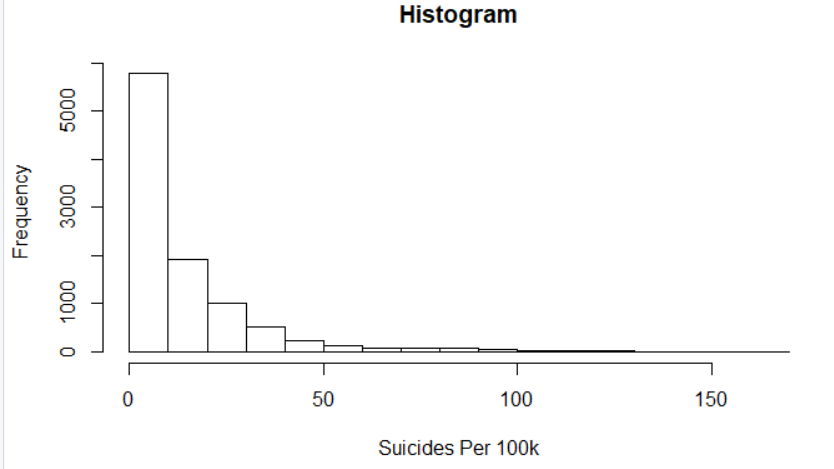


**Classifying Decision variable for using in the model**:

We are using column “SuicidesPer100k” as response variable in the model.

Since our goal is to predict risk of suicide based on predictor variables, we are categorizing the predictors variable to binary.

Plotting histogram to identify the cut-off value for risk variable.



As the histogram is right skewed, we are considering median value as cut-off for Risk variable.

If value of Suicidesper100k is less than median, Risk variable will be “0”, else “1”.

**Model Fitting:**

We are using two classification algorithms to predict and explore the causes of suicide in countries.

Model is using below columns as predictors.

Country, year, sex, age, population, GDPpyear, GDPpcapita, generation, employeecompensation, Unemployment, Lifeexpectancy, Refugees, Selfemployed, Internetusers, suicidesper100k, electricityacess

**Random Forest**:

A Random Forest is an ensemble technique capable of performing both regression and classification tasks with the use of multiple decision trees and a technique called Bootstrap Aggregation, commonly known as bagging.

Random Forests can be used for feature selection because if you fit the algorithm with features that are not useful, the algorithm simply won't use them to split on the data.

It's possible to extract the 'best' features (which could be the total number of times a feature was used to split on the data, or the mean decrease in impurity etc).

Here, Risk is the decision/response variable and all other columns from the training data are used as predictors.

**Evaluating the accuracy of model on Test data**:

There are a number of metrics to evaluate the accuracy of the classifier, listing each one of them below-

* **Accuracy -** Based on the prediction on test data, accuracy of the model is 95%, which means classifier is accurate 95% of the times. But in case of classification problem, having only one classification accuracy may not convey the whole picture. Hence we will be using confusion matrix to summarize the performance of classification algorithm.
* **Confusion Matrix** –

A confusion matrix provides a summary of the predictive results in a classification problem. Correct and incorrect predictions are summarized in a table with their values and broken down by each class.

**True positives (TP)**: the cases where classifier predicted risk of suicides and the records were actually “risk”. We have total 1434 true positives out of 2956 records.

**True negatives (TN)**: the cases for which the classifier predicted ‘no risk’ and the records were actually “no risk”.

**False positives (FP)**: the cases for which the classifier predicted ‘no risk’ but the records were actually “risk”.

**False negatives (FN)**: the cases for which the classifier predicted ‘no risk’ but the records were actually ‘risk’.

**Precision, Recall and F1-score**

Using the confusion matrix, we can compute accuracy metrics defined below.

**Precision**:

Precision is defined as the ratio of the total number of correctly classified positive classes divided by the total number of predicted positive classes. Or, out of all the predictive positive classes, how much we predicted correctly. Precision should be high.

In this case precision is 0.96, which means if the model predicts risk, how often it will be correct.

**Recall**:

Recall is defined as the ratio of the total number of correctly classified positive classes divide by the total number of positive classes. Or, out of all the positive classes, how much we have predicted correctly. Recall should be high.

In this case recall is 0.95, which shows that when data is actually risk, how often the model is predicting it as risk.

**F1-Score:**

It is difficult to compare two models with different Precision and Recall. So to make them comparable, we use F-Score. It is the Harmonic Mean of Precision and Recall. As compared to Arithmetic Mean, Harmonic Mean punishes the extreme values more.

**F-score = (2\*Recall\*Precision)/(Recall+Presision)**

**F-score is 0.95, since the F-score is high we can say that model is predicting correct cases most of the times.**

**Logistic Regression**:

Logistic regression was executed with risk as response variable, and other features in the training data as predictor variables.

Based on the prediction done on test dataset, below is the model accuracy.

* **Accuracy -**

Based on the prediction on test data, accuracy of the model is 93%, which means classifier is accurate 93% of the times.

* **Confusion Matrix** –

A confusion matrix provides a summary of the predictive results in a classification problem. Correct and incorrect predictions are summarized in a table with their values and broken down by each class.

**True positives (TP)**: the cases where algorithm predicted risk of suicides and the records were actually “risk”. We have total 1384 true positives out of 2956 records.

**True negatives (TN)**: the cases for which the classifier predicted ‘no risk’ and the records were actually “no risk”. Number of records – 1374.

**False positives (FP)**: the cases for which the classifier predicted ‘no risk’ but the records were actually “risk”. Number of records – 99.

**False negatives (FN)**: the cases for which the classifier predicted ‘no risk’ but the records were actually ‘risk’. Number of records – 99.

**Precision, Recall and F1-score**

Using the confusion matrix, we can compute accuracy metrics defined below.

**Precision**:

Precision from logistic regression is 0.93, which means if model predicts risk, 0.93 times it will be correct.

**Recall**:

Recall is defined as the ratio of the total number of correctly classified positive classes divide by the total number of positive classes.

In this case recall is 0.93, which shows that when data is actually risk, how often the model is predicting it as risk.

**F1-Score:**

**F-score = (2\*Recall\*Precision)/(Recall+Presision)**

F-score is 0.93, since the F-score is high we can say that model is predicting correct cases most of the times.

Since, F1-score is close to 1, we can say that model is predicting positive and negative cases accurately.

**Comparing Accuracy of Classfication algorithms**:

Based on the predicted output excuted on test dataset, we can confirm that

Random forest algorithm is more accurate in predicting risk of suicide compared to

Logistics regression.

The difference between the two is not very huge to discard one for the other.

**References:**

**Dataset source link –**

1. World Development Indicators. (2019, October 28). Retrieved from <https://datacatalog.worldbank.org/dataset/world-development-indicators>.

**Reference material to understand the dataset –**

1. Sharma, Prateek. “Decoding the Confusion Matrix.” Medium. Towards Data Science, August 9, 2019. <https://towardsdatascience.com/decoding-the-confusion-matrix-bb4801decbb.>
2. Evaluation Metrics for Machine Learning - Accuracy, Precision, Recall, and F1 Defined. (n.d.). Retrieved from <https://skymind.ai/wiki/accuracy-precision-recall-f1.>
3. Suicide. (n.d.). Retrieved from <https://www.who.int/news-room/fact-sheets/detail/suicide.>

**Appendix: -**

**library(naniar)**

**library(onehot)**

**library(randomForest)**

**library(caret)**

**library(e1071)**

**library(corrplot)**

**library(tidyverse)**

**library(ggplot2)**

**library(dplyr)**

**library(corrgram)**

**library(glmnet)**

**library(naniar)**