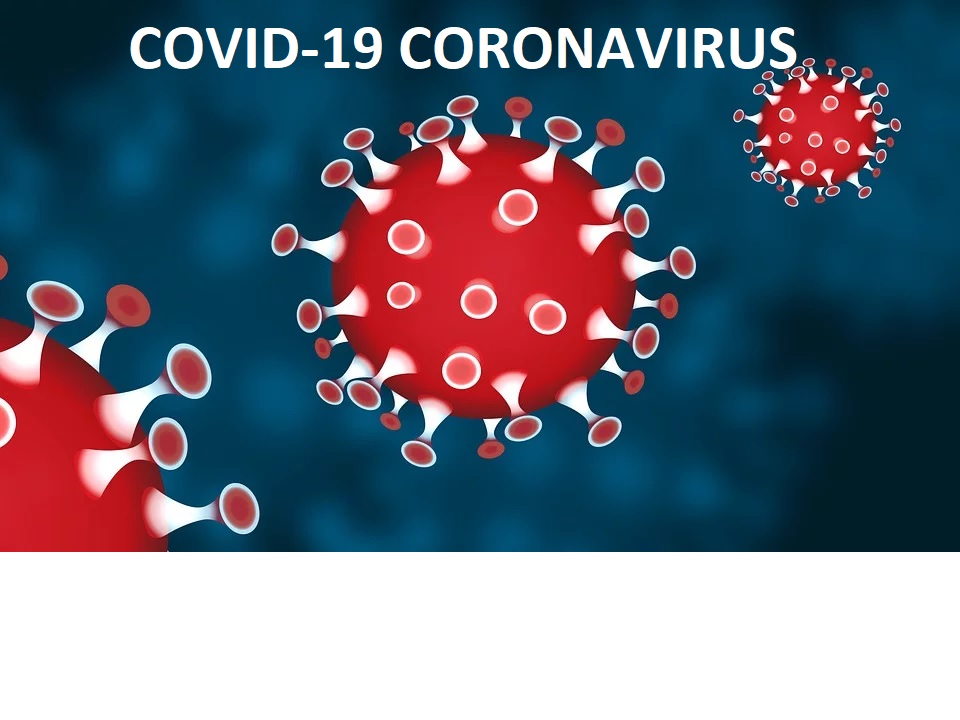
PROJECT REPORT ON COVID DATASET



Name: Nutan Hotwani

College: Maulana Abul Kalam Azad University of Technology, West Bengal

Degree: B.Tech-CSE (3rd yr)

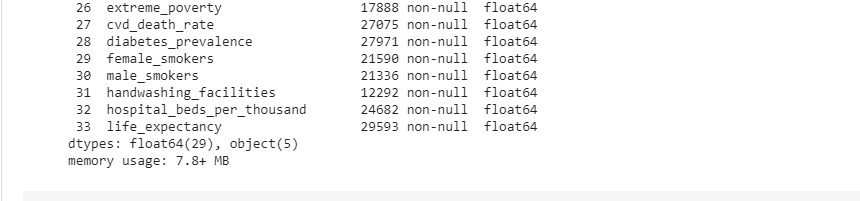
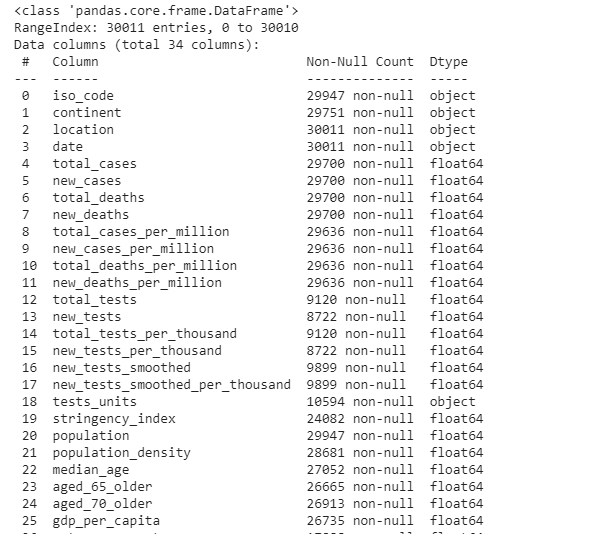
Subject: Machine Learning

======================================================================

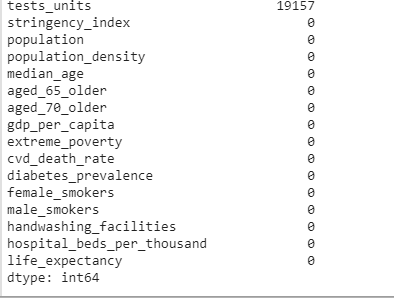
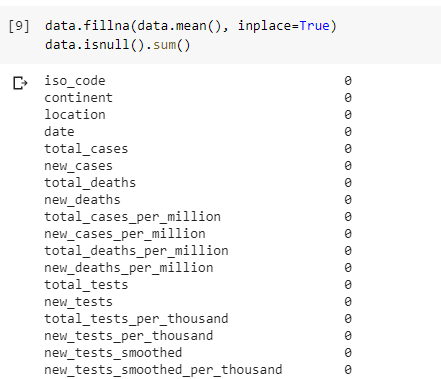
Dataset: <https://www.freelancer.com/users/l.php?url=https:%2F%2Fgithub.com%2Fowid%2Fcovid-19-data%2Ftree%2Fmaster%2Fpublic%2Fdata&sig=27bf4100ff8b2a2fd95f43dfe3a4775605447bdacb82b67c45555bed5ce221ff>

* Data Exploration:

The COVID dataset contains 34 columns each specifying the factors about the specified country and the general report on a daily basis. It includes common factors such as New\_cases\_per\_million, death\_cases\_per\_million, etc..

**The data was read using pandas in Python.** 

After exploring the data, it was found that several rows and columns contained NaN (Not a Number) values. Those values were to be eliminated to ensure smooth functioning of the model. Some rows such as the ones where countries were NaN were completely removed whereas others were mostly filled with the mean of the data in those columns.

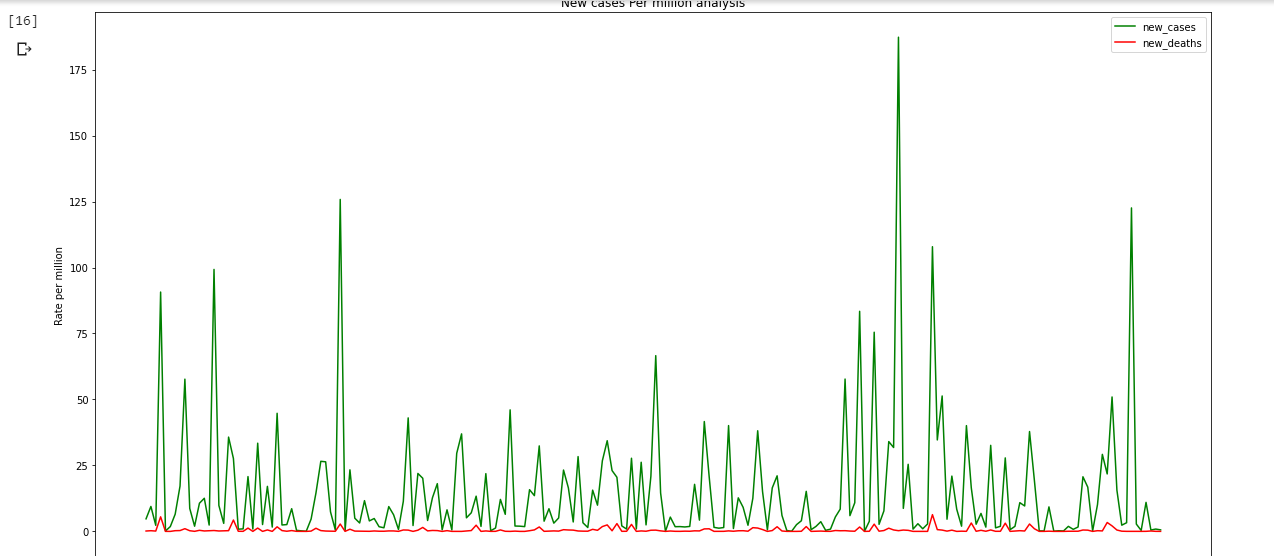
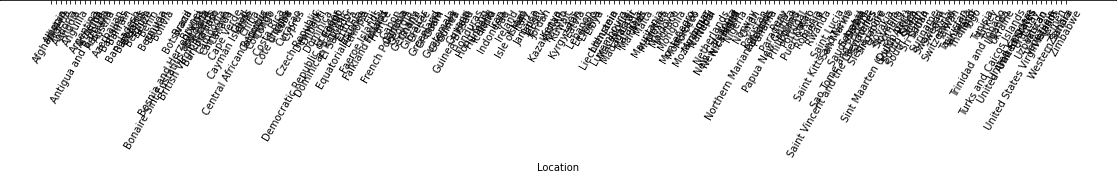
The ‘tests\_units’ contained the following elements, therefore could not be removed using the above methods:

array([nan, 'tests performed', 'units unclear', 'samples tested', 'people tested', 'units unclear (incl. non-PCR)'], dtype=object)

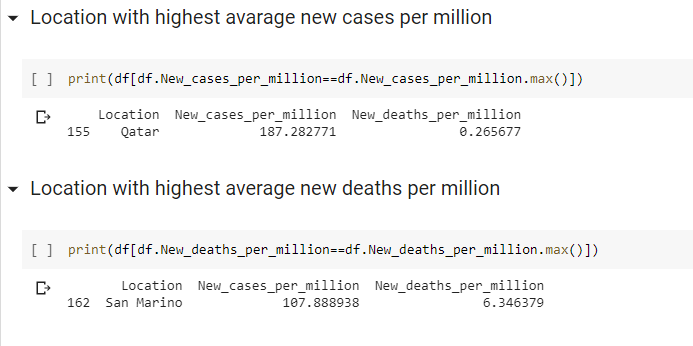
This column was later removed as it did not have much correlation with the dataset

* Data insights:

After exploring the data and cleaning it for smooth functioning, the insights and important features were observed. For instance, the maximum new\_cases\_increase per million and maximum new\_deaths per million was to be noted. The graphs were drawn and average increases per day of the countries were observed.

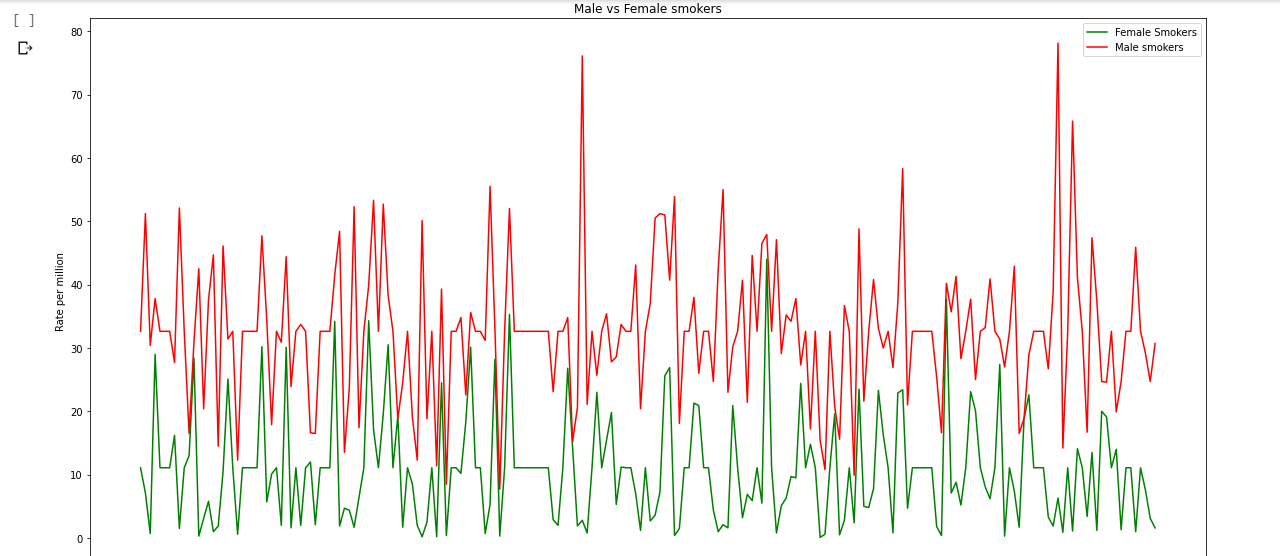
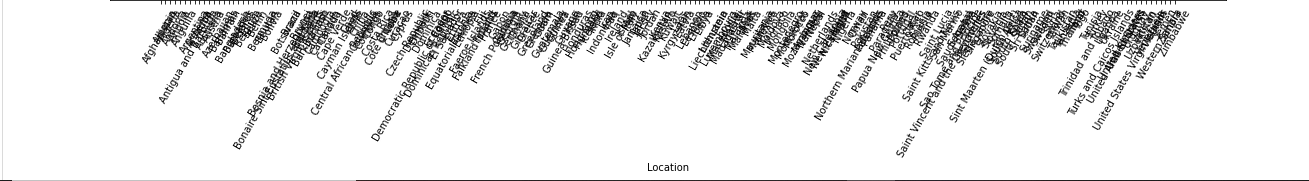


**It was shown that the country which showed maximum increase in new cases was Qatar**



**And maximum increase in deaths was San Marino**

**The variations in female and male smokers per country was also plotted and observed using graph**



There were total 201 unique countries. The plots above are made by taking the average of those countries.

* Overview of the Data

**A heat map is data analysis software that uses color the way a bar graph uses height and width: as a data visualization tool.**

**Feature selection** is one of the first and important steps while performing any machine learning task. A feature in case of a dataset simply means a column. When we get any dataset, not necessarily every column (feature) is going to have an impact on the output variable. If we add these irrelevant features in the model, it will just make the model worst.

We filter and take only the subset of the relevant features. The model is built after selecting the features. The filtering here is done using correlation matrix and it is most commonly done using **Pearson correlation**.

Here we will first plot the Pearson correlation heatmap and see the correlation of independent variables with the output variable ‘life\_expectancy’. We will only select features which has correlation of above 0.5 (taking absolute value) with the output variable.

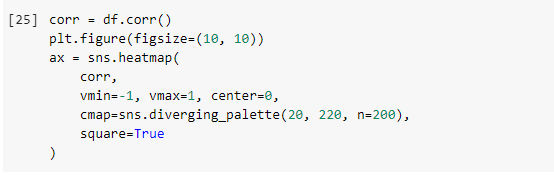
The correlation coefficient has values between -1 to 1

— A value closer to 0 implies weaker correlation (exact 0 implying no correlation)

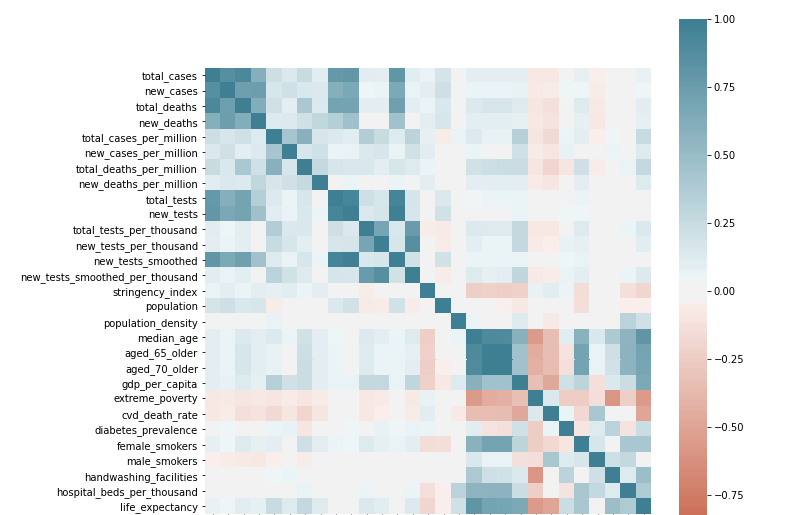
— A value closer to 1 implies stronger positive correlation

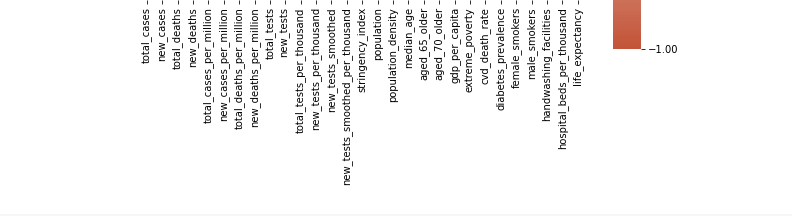
— A value closer to -1 implies stronger negative correlation

In the similar way, we can also make out the correlation between all the variables using the given technique.



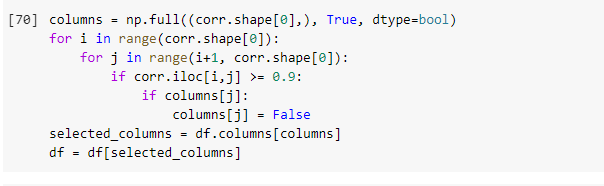
**Heat Map:**

****

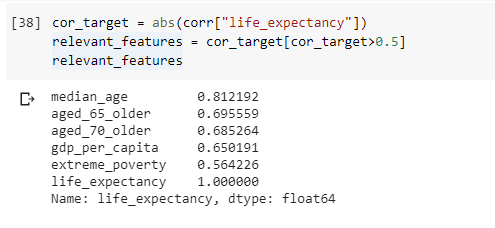
****

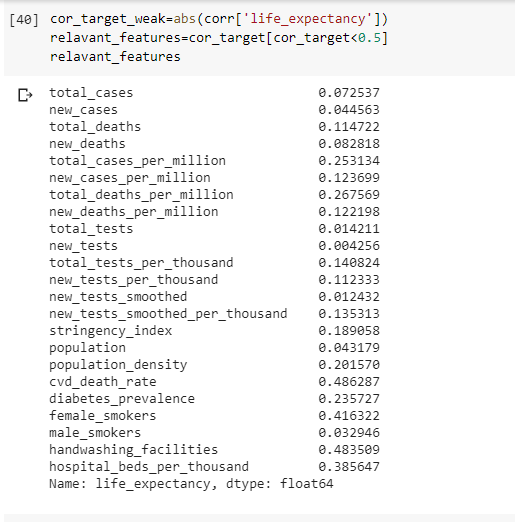
**As we see from the heatmap, the relation between all the elements can be established. The colour bar beside shows the variation in the features. A deep blue colour indicates positive correlation indicating the positive dependence of the features on each other. A deep red colour indicates a negative correlation indicating the negative dependence of the features on each other. As a data scientist, we should be concerned on improvising on the life expectancy of the countries. Keeping this in mind, we detect the dependence of the features on the ‘life\_expectancy’ feature.**

**Selection of the important features:**

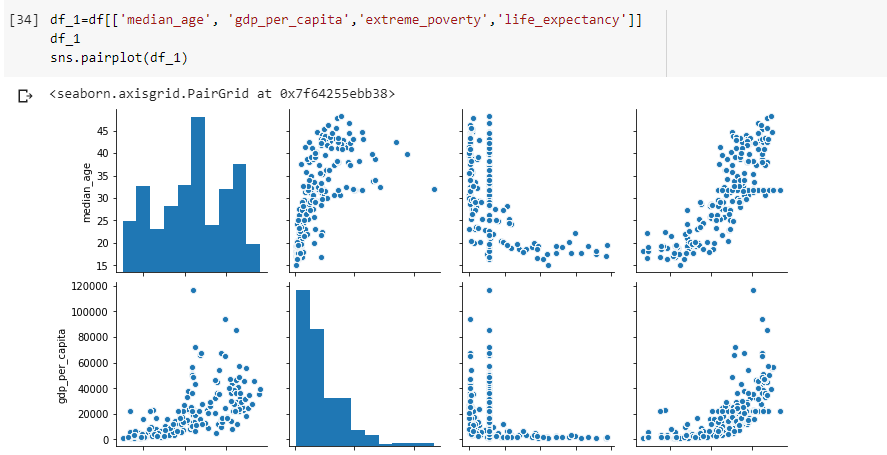
****

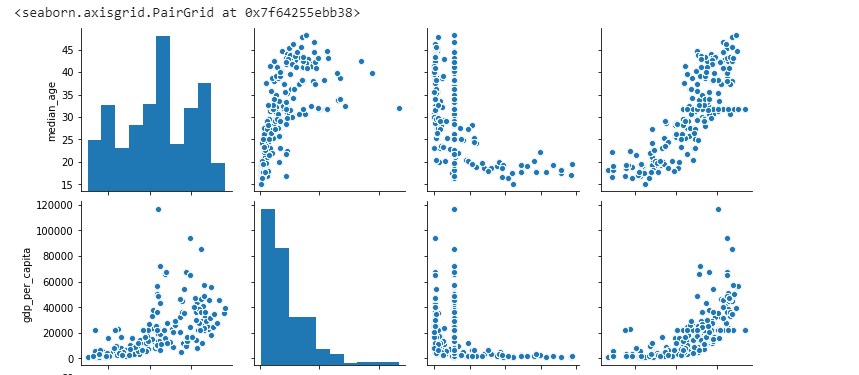
**Most relevant features to ‘life\_expectancy’:**

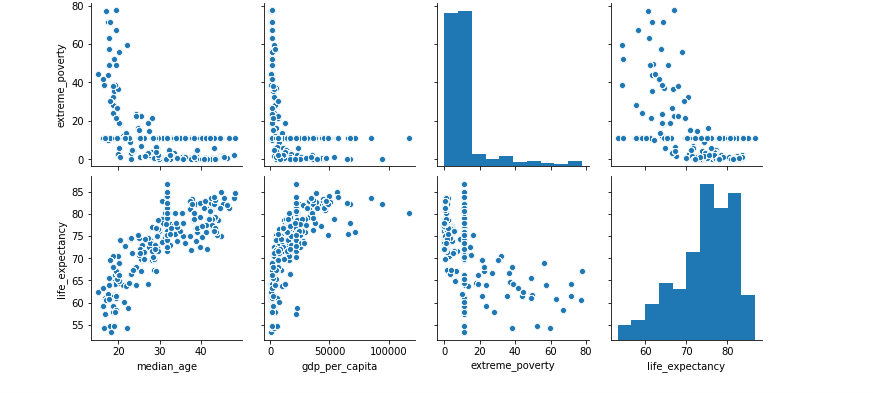


**Least relevant features to ‘life\_expectancy’:**

**We can plot using the pairplot feature in seaborn and detect the relation between all the relavant features**







**Thus, out of the 34 columns, these are the few features which are responsible in increasing the life expectancy of the people. We will now build a model on these features and can predict the life expectancy with maximum accuracy**

* How can we merge the dataset and create heterogeneous data

**“Merging” two datasets is the process of bringing two datasets together into one, and aligning the rows from each based on common attributes or columns.**

We can merge dataframes in Python using Pandas pd.merge method. Merging is basically required in Machine Learning when to find out a particular prediction, we have to take features from different datasets. In such a case, we can merge the different datasets and create a new dataset where we will have all the essential features for our model.

Syntax:

pd.merge(left\_df, right\_df, on='column\_name', how='inner')

Pandas uses “inner” merge by default. This keeps only the common values in both the left and right dataframes for the merged data.

Although the “inner” merge is used by Pandas by default, the parameter inner is specified above to be explicit.

The “outer” merge combines all the rows with right and left dataframes with NaN when there are no matched values in the rows.

* Innovative machine learning model we can create using this dataset

We could build a Regressor model in Machine Learning that would predict the life\_expectancy rate of the different countries.

1. **Using Random Forest Regressor**:

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 and Aggregation, commonly known as bagging. The basic idea behind this is to combine multiple decision trees in determining the final output rather than relying on individual decision trees.  
Random Forest has multiple decision trees as base learning models. We randomly perform row sampling and feature sampling from the dataset forming sample datasets for every model. This part is called Bootstrap.

We need to approach the Random Forest regression technique like any other machine learning technique

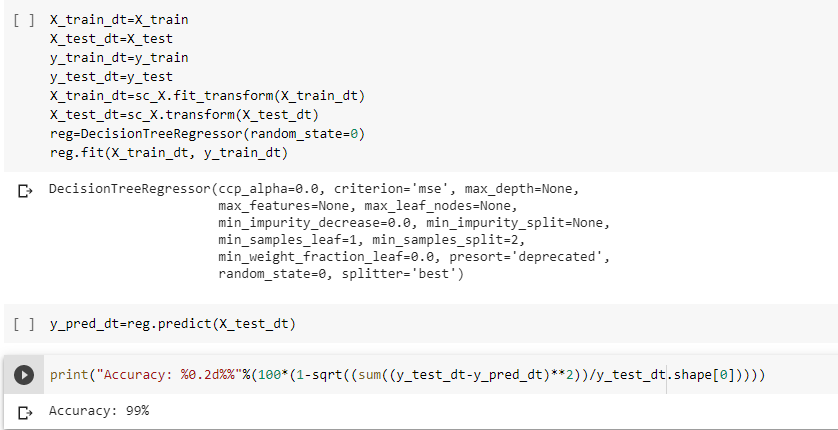
* Design a specific question or data and get the source to determine the required data.
* Make sure the data is in an accessible format else convert it to the required format.
* Specify all noticeable anomalies and missing data points that may be required to achieve the required data.
* Create a machine learning model
* Set the baseline model that you want to achieve
* Train the data machine learning model.
* Provide an insight into the model with test data
* Now compare the performance metrics of both the test data and the predicted data from the model.
* If it doesn’t satisfy your expectations, you can try improving your model accordingly or dating your data or use another data modeling technique.
* At this stage you interpret the data you have gained and report accordingly.

**It gave us an accuracy of 99% which excellent!**



1. **Using Decision Tree Regressor**:

Decision tree regression observes features of an object and trains a model in the structure of a tree to predict data in the future to produce meaningful continuous output. Continuous output means that the output/result is not discrete, i.e., it is not represented just by a discrete, known set of numbers or values.

****

**It also gave us an accuracy of 99%!**

**Difference between decision tree regressors and random forest regressors:**

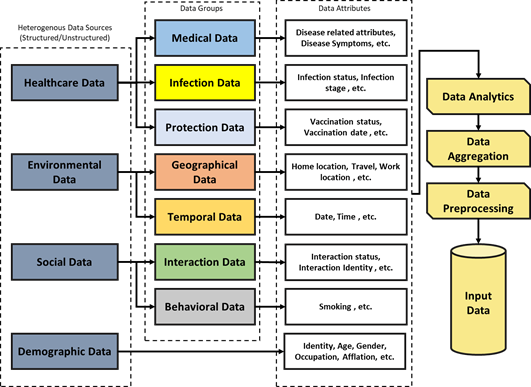
A decision tree is built on an entire dataset, using all the features/variables of interest, whereas a random forest randomly selects observations/rows and specific features/variables to build multiple decision trees from and then averages the results. After a large number of trees are built using this method, each tree "votes" or chooses the class, and the class receiving the most votes by a simple majority is the "winner" or predicted class. There are of course some more detailed differences, but this is the main conceptual difference.

When using a decision tree model on a given training dataset the accuracy keeps improving with more and more splits. You can easily overfit the data and doesn't know when you have crossed the line unless you are using cross validation (on training data set). The advantage of a simple decision tree is model is easy to interpret, you know what variable and what value of that variable is used to split the data and predict outcome.

A random forest is like a black box and works as mentioned in above answer. It's a forest you can build and control. You can specify the number of trees you want in your forest (n\_estimators) and also you can specify max num of features to be used in each tree. But you cannot control the randomness, you cannot control which feature is part of which tree in the forest, you cannot control which data point is part of which tree. Accuracy keeps increasing as you increase the number of trees, but becomes constant at certain point. Unlike decision tree, it won't create highly biased model and reduces the variance.

* Summary

Thus, having studied the model correctly and extracted out the features, we can conclude that the ‘life\_expectancy’ of the countries depend on only few of the above mentioned features. As shown in the heap map, most of the features were white in colour, meaning neutral to the task and could not contribute in it. Some also gave a negative response (markings in red).



Out of the given features as shown in the above picture**, our dataset contained demographic data, behavioural data and few more. It however, failed to incorporate medical facilities which could be an important feature for predicting the life\_expectancy of the people.** Geographical and temporal data were seemed meaningless after exploring the data. Hence, if possible, we could arrange medical data of the people and could smoothly predict the recovery rate of the people.