6/29/2020

Regression Model for Covid-19 Infection Prediction

Table of Contents

Table of Figures

1. Introduction

Regression Models are used to predict any continues number or a quantity. The model for Covid-19 will be able to predict the number of new cases with the help of given features. There are many algorithms to be used in regression models for example: SVM Regressor, SGD Regressor etc. Every algorithm has their own pros and cons, for selecting an algorithm for a specific dataset needs to be validated before actual usage. There are many techniques for the validation of a model with specific dataset. We will be using Learning Curves and Model Complexity Graph techniques in the validation of a model. By the help of these two techniques we can find out the Underfitting and Overfitting problems of the model, which will help us to adjust the parameters of the model for avoiding those situations. Another technique for the optimization of the model is also been used which is known as "Grid Search". This technique will help us to find the best parameters for the optimization of model. After applying all these techniques, we can have an optimized model which can learn and predict in a better way.

2. Data Exploration

The dataset for this project has been taken from "Our World in Data (Covid-19 Cases) ". The dataset has almost 26,000 entries representing data about 33 features for Covid-19 cases. There are sperate cases for each country up to 25th June, 2020. For the purpose of this project only 4 features are selected as important features and other 29 features have been removed from the dataset. All features from the dataset are:

- ISO code.
- Continent.
- Location (Country Name).
- Date.
- Total cases.
- Total deaths.
- New deaths.
- Total cases per million.
- New cases per million.
- Total deaths per million.
- New deaths per million.
- Total tests.
- New tests.
- Total tests per thousand.
- New tests per thousand.
- New tests smoothed.
- New tests smoothed per thousand.
- Test units.
- Stringency index.

- Population.
- Population density.
- Median age.
- Aged 65 older.
- Aged 70 older.
- GDP per capita.
- Extreme poverty.
- Cvd death rate.
- Diabetes prevalence.
- Female smokers.
- Male smokers.
- Handwashing facilities.
- Hospital beds per thousand.
- Life expectancy.

2.1 Feature Selection

Feature selection is a very difficult step in which we have to analyze every feature and find out its importance with the corresponding model. Out of 33 features 4 features were selected to be used in the model. Selected Features are:

1. New tests.
2. Population.
3. Population density.
4. Median age.

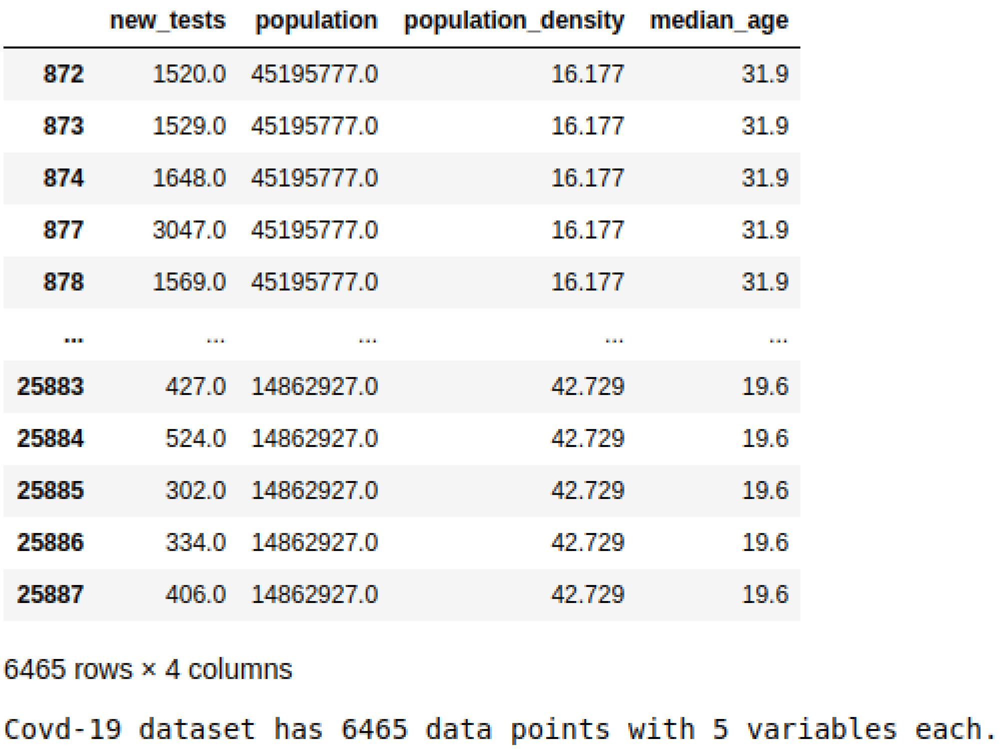
The other 29 features have many missing and invalid entries, some of them has up to 18k missing entries out of 26k which lead in the deletion of the feature. And some other features did not effect the output of the model which also lead in the deletion of the feature.

2.2 Data Filtration

The dataset has almost 18% missing values. For handling those missing values there are many techniques used in Machine Learning, but there are lots of side effects when we use them. Three techniques are used for dealing with missing values and one is selected for the filtration of dataset. Techniques which have been used are:

- Delete the entire entry if there is one missing feature in it.
- Fill up the missing values of feature by calculating the mean/median of the values.
- Filling up the missing values by the help of Sklearn Imuter.

When we try to fill up the missing values from any one of the two techniques mention above, the entire entry ended up being an outlier to the whole dataset and also it effected the accuracy rate of the model. Deletion of entries tend to be the best policy used in this dataset because we removed all outliers and random values used to fill up the entry. After the removal of all those missing value entries, the remaining entries were 6,465. Out of 26,000 entries 6,465 were used in the training and testing of the model.



	new_tests	population	population_density	median_age
872	1520.0	45195777.0	16.177	31.9
873	1529.0	45195777.0	16.177	31.9
874	1648.0	45195777.0	16.177	31.9
877	3047.0	45195777.0	16.177	31.9
878	1569.0	45195777.0	16.177	31.9
...	...	...	...	...
25883	427.0	14862927.0	42.729	19.6
25884	524.0	14862927.0	42.729	19.6
25885	302.0	14862927.0	42.729	19.6
25886	334.0	14862927.0	42.729	19.6
25887	406.0	14862927.0	42.729	19.6

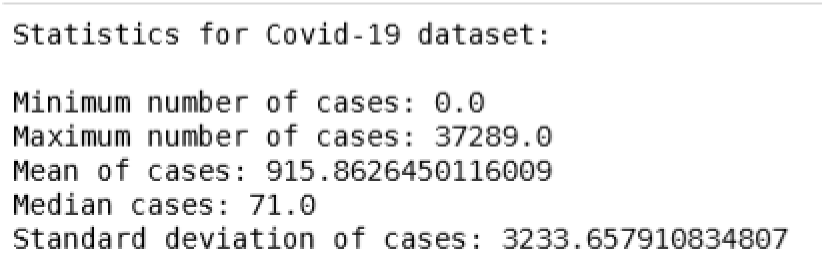
6465 rows × 4 columns

```
Covd-19 dataset has 6465 data points with 5 variables each.
```

Figure 1

3. Statistical Analysis

Statistical analysis has been done on the label (output) of the model which is “new cases”. In statistical analysis we have to computed minimum number of cases, maximum number of cases, mean of cases, median of cases and standard deviation of cases. Those statistical analysis helps us in the understanding of the data. The results for the statistical analysis are given below



```
Statistics for Covid-19 dataset:

Minimum number of cases: 0.0
Maximum number of cases: 37289.0
Mean of cases: 915.8626450116009
Median cases: 71.0
Standard deviation of cases: 3233.657910834807
```

Figure 2

4. Performance Metric

Performance metric functions are used to determine the performance of the model, we will be using R^2 score as our performance metric function. R^2 is known as **coefficient of determination,** it is a useful statistic in regression analysis, as it often describes how "good" or "bad" a model is at making predictions. The values of R^2 ranges from 0 to 1, which captures the percentage of squared correlation between the predicted and actual values of the target variable. A model with R^2 score closer to 0 is bad at making prediction and a model with R^2 score closer than 1 is good at making predictions, sometimes R^2 score will be below 0 in negative number then it is clear that we are not using the right model for the dataset.

5. Analyzing Model Performance

For the analysis of model performance, we will be using two techniques **Learning Curves** and **Model Complexity Graph.** Both these techniques will give us much information for the validation of the model and by the help of which we will decide which model to use or not.

5.1 Model Complexity Graph

The model complexity graph validates the model with training and validation set of the data. This graph will generate two complexity curves – one for training and one for validation of the model. The shaded region of the curves denotes the uncertainty in those curves and the score of the model is measured by R^2 performance metric. From the graphs below we can clearly see that the best algorithm to use is **Decision Tree Regressor**. I have also tried another algorithm which is **Stochastic Gradient Descent** but that one given the same result as **Logistic Regressor**. Model Complexity graphs for different models are given below:

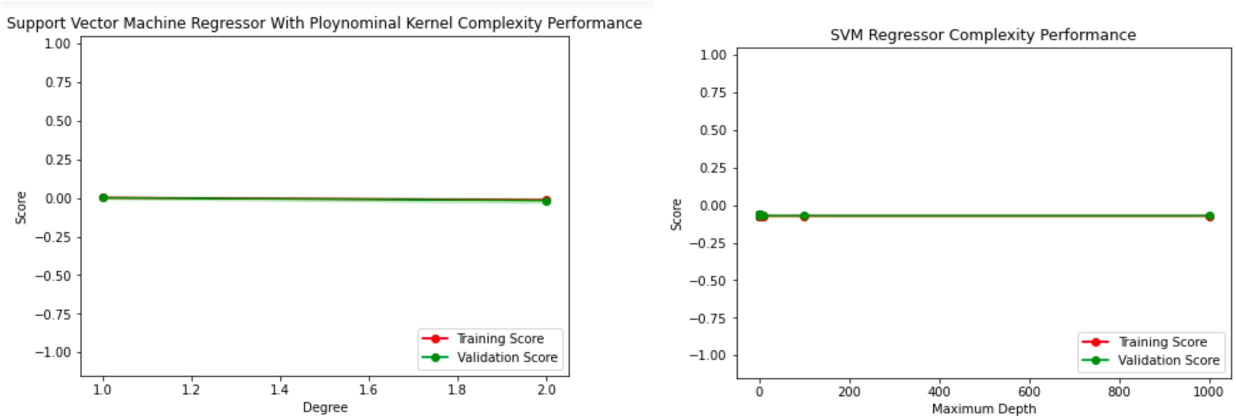


Figure 3

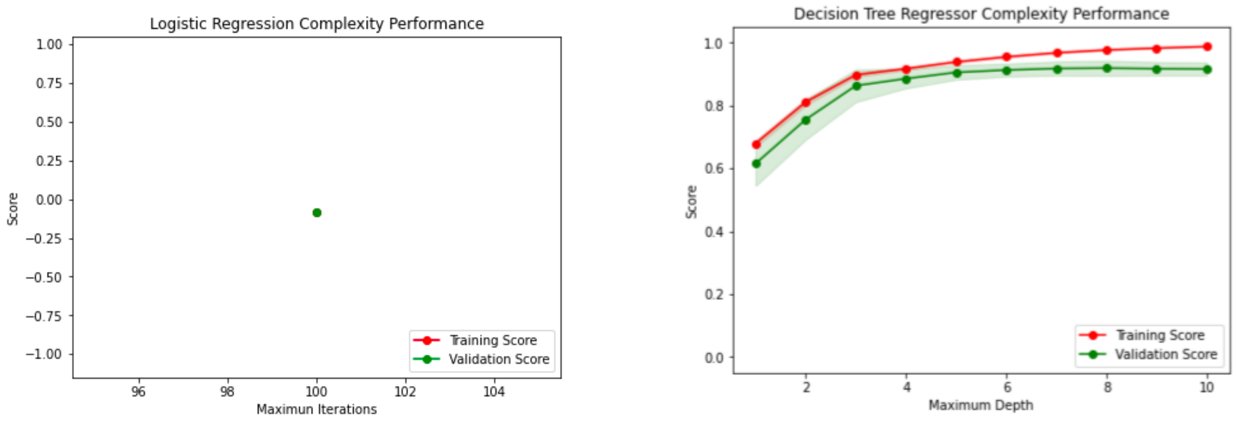


Figure 4

5.2 Learning Curves

For the selected model **Decision Tree Regressor** now we will apply learning curves technique to measure the different aspects of the model with the given dataset. The learning curves will generate multiple graphs with different maximum depths for the Decision Tree Regressor model. The shaded region of the curves denotes the uncertainty in those curves and the score of the model is measured by R^2 performance metric.

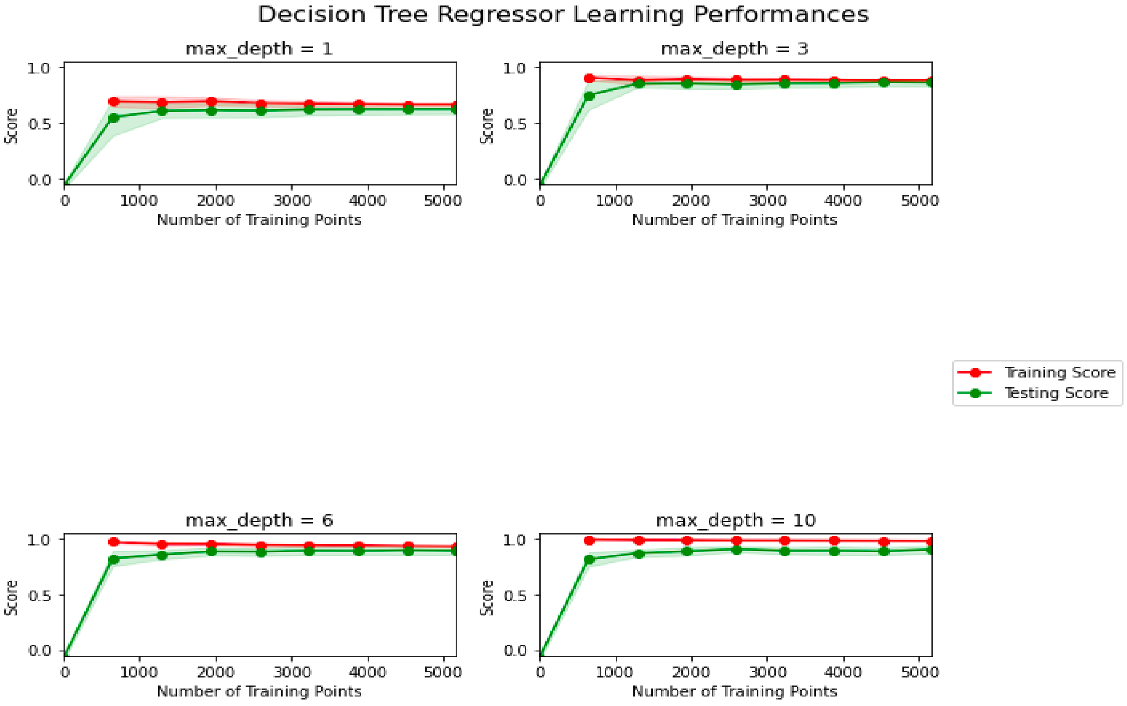


Figure 5

6. Model Evaluation

6.1 Model Training

The model will be trained on the selected algorithm **Decision Tree Regressor.** But before that we will ensure that we are using the optimized model for training which will be done using **Grid Search.** Grid search will help us in selecting the best parameter values for the maximum depth of a tree. Grid search works cross validation to ensure the best output results. After applying the Grid search, it has given us an **optimized maximum depth value of 10,** and also it has provided us an optimized regression model.

6.2 Model Testing

With the help of trained model now we will test the model with testing dataset. R^2 score of predicted data s **0.857** which is closer to 1. The result after testing the model is given below:

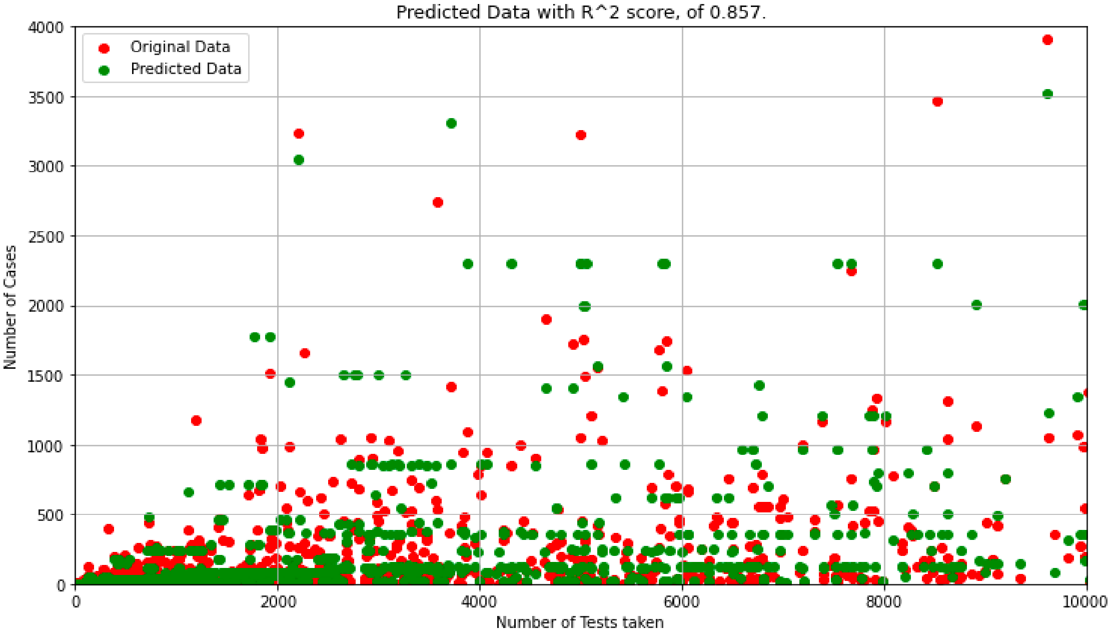
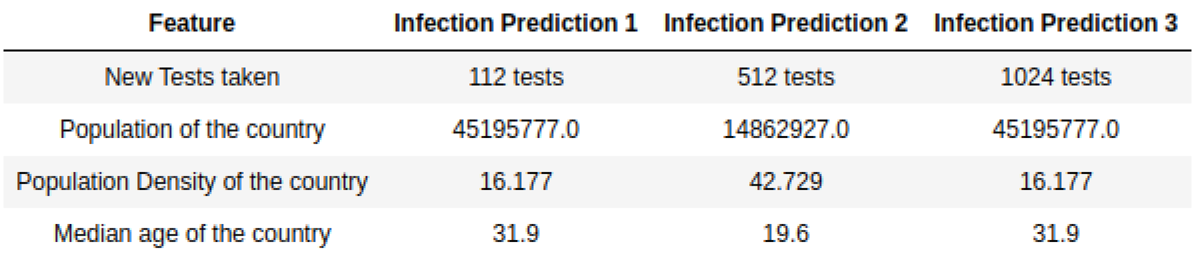


Figure 6

6.3 Predicting cases with custom data

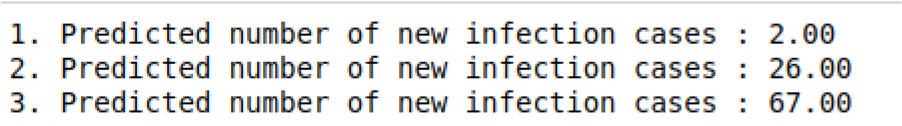
We can also test this model with our own data. New data for 3 predictions has been filled with different values for input features. The new input data is given below



Feature	Infection Prediction 1	Infection Prediction 2	Infection Prediction 3
New Tests taken	112 tests	512 tests	1024 tests
Population of the country	45195777.0	14862927.0	45195777.0
Population Density of the country	16.177	42.729	16.177
Median age of the country	31.9	19.6	31.9

Figure 7

The results with this given dataset is given below:



```
1. Predicted number of new infection cases : 2.00
2. Predicted number of new infection cases : 26.00
3. Predicted number of new infection cases : 67.00
```

Figure 8

7. Model Visualization

A decision tree is generated by Decision Tree Regressor which helps the algorithm in making the decision about the incoming data. The depth of decision tree generated by our given dataset is 10, and that's why the generated tree is very large to fit in this document. A picture of small subtree from that decision tree is given below

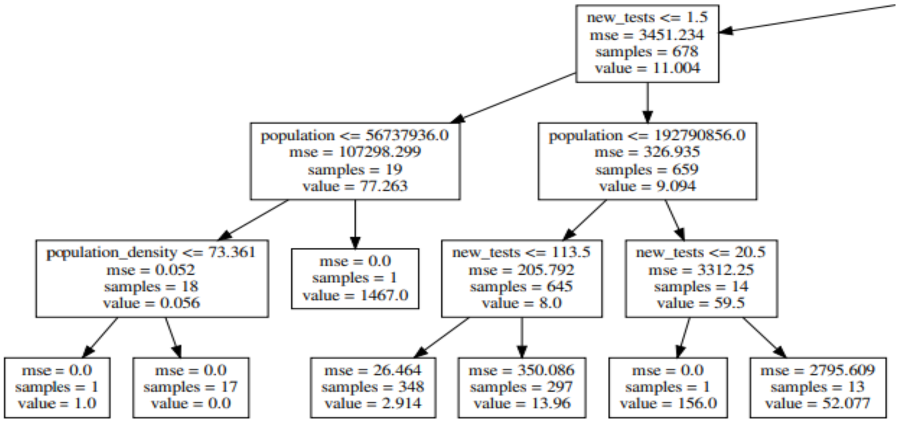


Figure 9