FLIPR HIRING HACKATHON

ML ASSIGNMENT

1. Data Insights

Different label explanations in the given datasets

(array(['Bengaluru', 'Bhubaneshwar', 'Chandigarh', 'Chennai', 'Delhi'],

dtype=object), array([1661, 1627, 1658, 1606, 225]))

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

(array(['Female', 'Male'], dtype=object), array([3402, 3375]))

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

(array(['Mr', 'Mrs'], dtype=object), array([3375, 3402]))

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

(array(['NO', 'YES'], dtype=object), array([1517, 5260]))

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

(array([0., 1., 2.]), array([1690, 3389, 1698]))

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

(array(['Business', 'Cleaner', 'Clerk', 'Driver', 'Farmer', 'Legal',

'Manufacturing', 'Researcher', 'Sales'], dtype=object), array([713, 751, 721, 738, 786, 795, 769, 772, 732]))

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

(array(['Car', 'Public', 'Walk'], dtype=object), array([2210, 2229, 2338]))

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

(array(['Elevated', 'Normal', 'Stage-01', 'Stage-02'], dtype=object), array([1715, 1716, 1672, 1674]))

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

(array(['Coronary Heart Disease', 'Diabetes', 'Hypertension', 'None'],

dtype=object), array([1703, 1685, 1694, 1695]))

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

(array(['<100', '<200', '<300', '<400'], dtype=object), array([1674, 1672, 1715, 1716]))

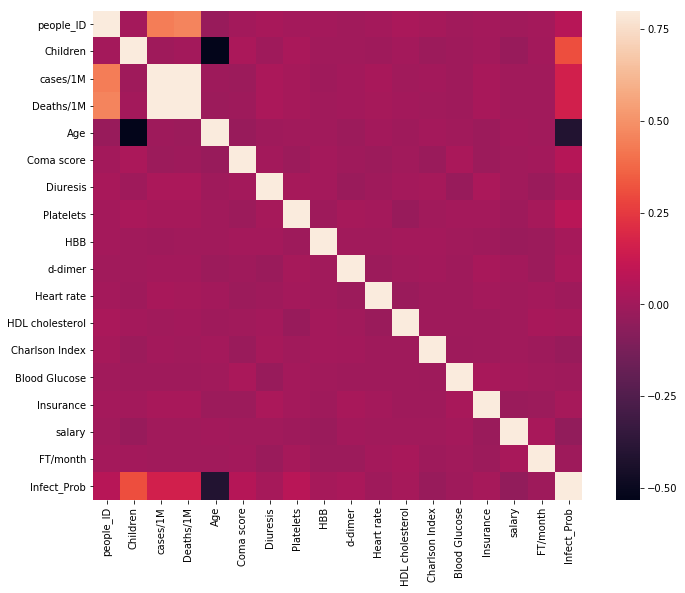
Cleaning and preprocessing

* All the labeled columns were first label encoded then they were one hot encoded

For better virtualisation of the features.

* All the numerical based values were normalised
* All nan columns were filled with appropriate values using imputer

Correlation among the variables



For cleaning and preprocessing refer to the notebook <https://github.com/rahulgarg28071998/Flipr_hiring_hackathon/blob/master/data_cleaning.ipynb>

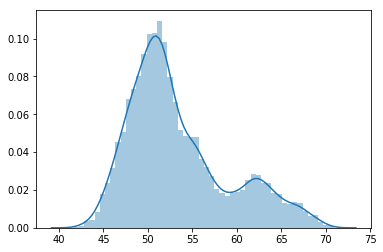
**Q1 Which model have you used for probability prediction? Explain your model.**

Many models were used to predict the infection probability.

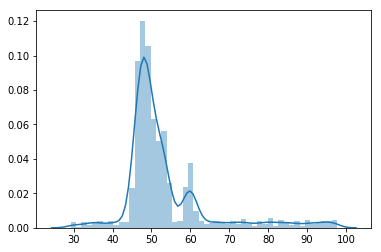
Regression models used to analyse the results are:

|  |  |
| --- | --- |
| **MODEL** | SCORE |
| LinearRegression | 0.29597580642812704 |
| DecisionTreeRegressor | 0.9999999999977941 |
| AdaBoostRegressor | 0.3147292384796998 |
| GradientBoostingRegressor | 0.4869782652915635 |
| MLPRegressor | 0.27780564821666687 |
| RandomForestRegressor | 0.8969930074559131 |

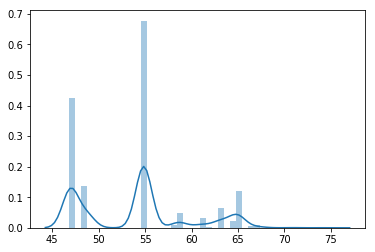
**Linear Regression fit curve:**



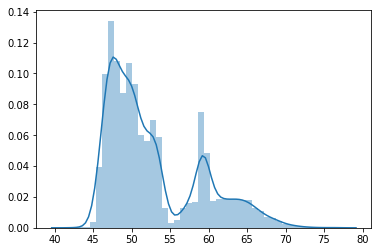
**DecisionTreeRegressor fitness curve:**



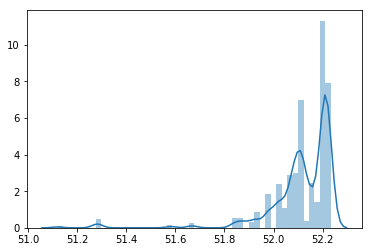
**AdaBoostRegressor** **fitness curve:**



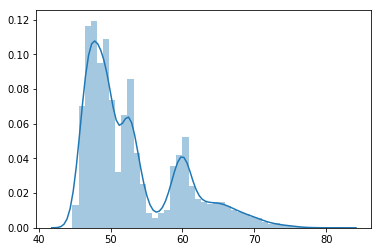
**GradientBoostingRegressor** **fitness curve:**



**MLPRegressor** **fitness curve:**

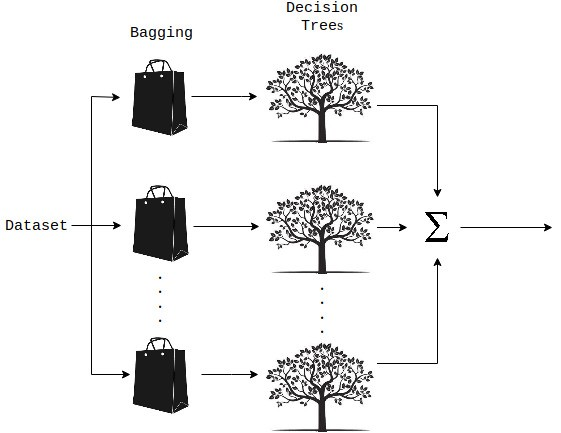


**RandomForestRegressor** **fitness curve:**



**Out of all the models RandomForestRegressor gave promising results.**

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. What is bagging you may ask? Bagging, in the Random Forest method, involves training each decision tree on a different data sample where sampling is done with replacement.



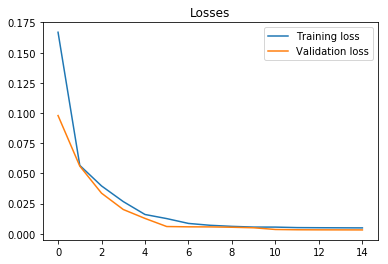
[**https://medium.com/datadriveninvestor/random-forest-regression-9871bc9a25eb**](https://medium.com/datadriveninvestor/random-forest-regression-9871bc9a25eb)

**Q2 .Which model have you used for Diuresis Time series prediction? Explain your model.**

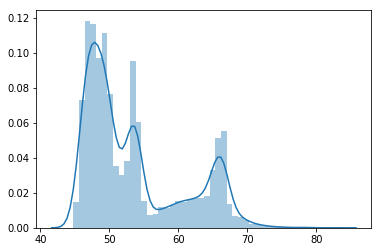
**Model used LSTM for time series prediction:**

The Long Short Term Memory neural network is a type of a Recurrent Neural Network (RNN). RNNs use previous time events to inform the later ones. For example, to classify what kind of event is happening in a movie, the model needs to use information about previous events. RNNs work well if the problem requires only recent information to perform the present task. If the problem requires long term dependencies, RNN would struggle to model it. The LSTM was designed to learn long term dependencies. It remembers the information for long periods.

<https://towardsdatascience.com/lstm-for-time-series-prediction-de8aeb26f2ca>



**RandomForestRegressor** **fitness curve on 27 march 2020:**



**DecisionTreeRegressor** **fitness curve on 27 march 2020:**

