1. Briefly describe the conceptual approach you chose! What are the trade-offs?

The algorithm chosen is RandomForestClassifier. As part of the exploratory data analysis, all the missing data is replaced with the median of the data in that particular column. When the data is plotted against the norm, it is found that the values in all the columns are not normally distributed. Since the data is skewed, median is used to fill the missing data. No columns are removed as all the column distributions are varying without any correlations. And also, the number of features are not huge, a decision is taken to not remove any data.

The training data is divided into a training set and a validation set. The validation set has 25000 records and the rest is the training set. The same median values which are used to fill the missing values in columns are used in the test set as well for the respective columns.

Since the data is varying, with random forest it becomes difficult to come up with an ideal model as the data is highly segmented. However, it is good for a small to medium set of data to come up with a recently good model.

Tried PCA to reduce some dimensions but it was counter-productive as the training duration increased significantly without any improvement in the performance.

1. What's the model performance? What is the complexity? Where are the bottlenecks?

The metrics taken for measuring the performance of the model are

1. Runtime or Duration - 1min 28s
2. R Square –

0.9987635726795097(training set)

0.96436(validation set)

1. OOB Score - 0.9635464098073555
2. F1score - 0.95 (validation set)

The complexity of a randomforest classifier is O(v \* n log(n)), where n is the number of records (5,71,000) and v is the number of features (57)

If the data is highly segmented, it is difficult to come up with a good model. It takes time to tune the hyper parameters like the number of estimators, minimum sample leaves, max features etc. This model can only be used for average amount of data to the tune of 0.5 to 1 million.

1. If you had more time, what improvements would you make, and in what order of priority?

More data analysis might be required to come up with a robust model although initial analysis revealed that most of the data is skewed. More feature dependencies need to be explored to identify the features that affect the prediction.

Trying out alternate algorithms like Gradient Boost classifier, Logistic regression and a deep neural network. Each algorithm has its own hyper parameters which need to be tuned to output a good model. We can also try clustering algorithms like KMeans and DBScan and measure their efficiency after segregating the data into 2 clusters.