

### **Using Predictive Analytics to predict Surge Price Type**



### **INTRODUCTION**

With the upcoming cab aggregators and demand for mobility solutions, the past decade has seen immense growth in data collected from commercial vehicles with major contributors such as Uber and Ola to name a few.

There are loads of innovative data science and machine learning solutions being implemented using such data and that has led to tremendous business value for such organisations.





### **BUSINESS PROBLEM**

XXX Cab Private Limited company is a cab aggregator service company. Their customers can download their app on smartphones and book a cab from any where in the cities the company operates in. They, in turn search for cabs from various service providers and provide the best option to their client across available options. They have been in operation for little less than a year now. During this period, they have captured surge\_pricing\_type from the service providers.

The business aim is to build a predictive model, which could help them in predicting the surge\_pricing\_type pro-actively. This would in turn help them in matching the right cabs with the right customers quickly and efficiently.



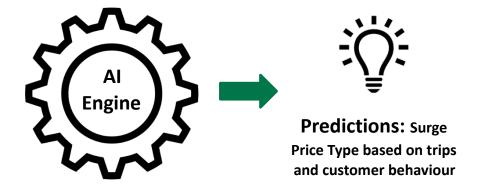
### **PROPOSED SOLUTION**

- Deploy AI engine to predict surge price type
  - Analogous to a typical classification problem
  - Use existing cab trips & customer data to train the Al model
  - Potential algorithms that can be explored:
    - Logistic Regression
    - Tree-based Classification (Decision Trees)
    - Bagging Algorithms (Random Forest)
    - Boosting Algorithms (XGBoost)

### Leveraging available input and target variables to make Surge Price **Predictions**

Input Variables	Definition
Trip_ID	ID for TRIP
Trip_Distance	The distance for the trip requested by the customer
Type_of_Cab	Category of the cab requested by the customer
Customer_Since_Months	Customer using cab services since n months; 0 month means current month
Life_Style_Index	Proprietary index created by XXX Cabs showing lifestyle of the customer based on their behavior
Confidence_Life_Style_Index	Category showing confidence on the index mentioned above
Destination_Type	XXX Cabs divides any destination in one of the 14 categories.
Customer_Rating	Average of life time ratings of the customer till date
Cancellation_Last_1Month	Number of trips cancelled by the customer in last 1 month
Var1, Var2 and Var3	Continuous variables masked by the company
Gender	Gender of the customer

Target Variable	Definition
Surge_Pricing_Type	Predictor variable can be of 3 types





- Automated solution for predicting surge price
- Accurate matching of right cabs with the right customers quickly and efficiently
- Avoiding loosing customers
- Mitigation of financial and business losses

### **Machine Learning Modelling Steps**

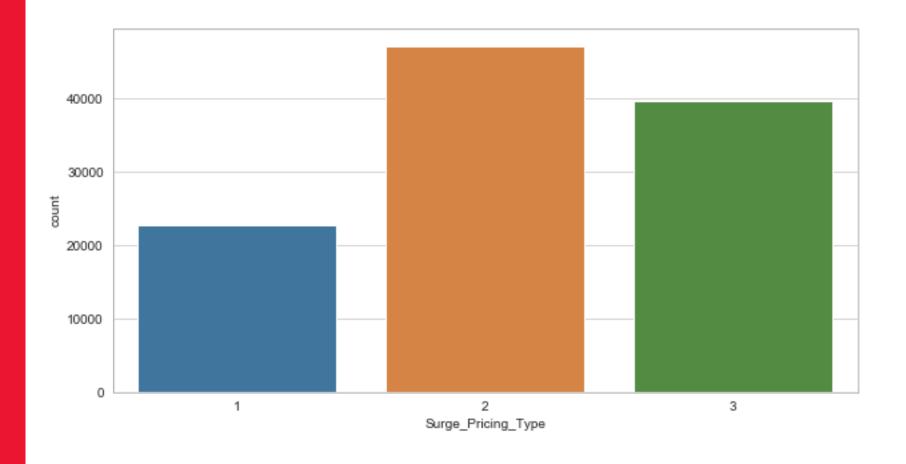
#### **Steps**

- Importing libraries and data
- Exploring data
- Checking for missing values
- Treating missing values by EDA
- Converting categorical columns to one-hot encoding
- Checking correlation & dropping highly correlated values
- Outlier detection & treatment
- Checking skewness of data
- Univariate, bivariate, multivariate analysis
- Scaling continuous values by Standard Scaler
- X and y assignment
- Train and test splitting (70-30)
- Training using LR, DT, RF, XGB
- Checking training accuracies
- Determining feature importance
- Prediction on Test data
- Evaluation metrics

### **Input Data Exploration and Understanding**

- Trip\_ID is not required in building up the predictive model and so can be dropped
- The dataset has no empty values for input variables Trip\_Distance, Destination\_Type, Customer\_Rating,
  Cancellation\_Last\_1Month, Var2, Var3, Gender
- There are no empty values in Target variable Surge\_Pricing\_Type
- We have some missing values in Type\_of\_Cab which is a categorical column with values A,B,C,D and E. For the missing values (nan), a new category 'F' can be created
- Customer\_Since\_Months are the customers using the cab service since n months. Nan values in this column can thus be replaced with 0, indicating that they are the newbies to this cab service
- Life\_Style\_Index and Confidence\_Life\_Style\_Index are the Proprietary indexes created by XXX Cabs showing lifestyle of the customer based on their behaviour. EDA can be performed to replace nan values
- Var1 is a continuous variable masked by the company. Again, need to perform EDA to replace the nan values

### Target Variable (Surge Price Type) Data Exploration



#### INSIGHTS

Since there is not much difference between the target variable classes, sampling isn't required.

### **Correlation Pair Plot**

Trip_Distance	1	0.13	0.47	-0.056	-0.0053	-0.05	0.2	0.23
Customer_Since_Months	0.13	1	0.16	-0.036	-0.0023	-0.015	0.056	0.11
Life_Style_Index	0.47	0.16	1	0.19	0.07	-0.086	0.21	0.3
Customer_Rating	-0.056	-0.036	0.19	1	0.0039	-0.012	-0.3	-0.23
Cancellation_Last_1Month	-0.0053	-0.0023	0.07	0.0039	1	0.011	0.096	0.13
Var1	-0.05	-0.015	-0.086	-0.012	0.011	1	-0.05	-0.065
Var2	0.2	0.056	0.21	-0.3	0.096	-0.05	1	0.68
Var3	0.23	0.11	0.3	-0.23	0.13	-0.065	0.68	1
	Trip_Distance	mer_Since_Months	Life_Style_Index	Oustomer_Rating	ation_Last_1Month	Var1	Var2	Var3

#### INSIGHTS

- 0.75

- 0.50

- 0.25

- 0.00

- -0.25

- -0.50

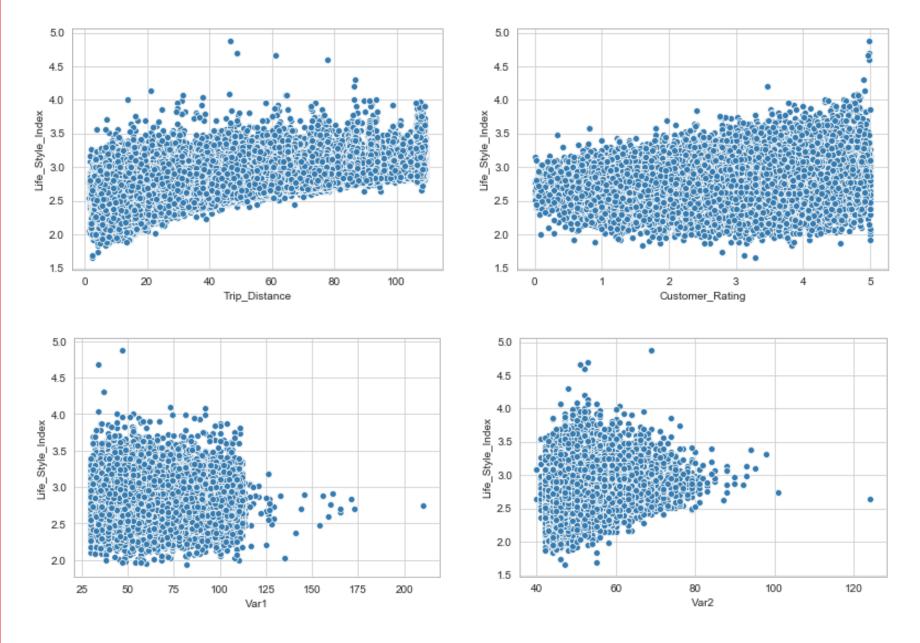
- -0.75

Correlation matrix values > 0.9 symbolises high correlation values between the independent features.

None of the variables are much correlated.

Var2 and Var3 are 68% correlated, no need of dropping any variable.

### **EDA** on Life\_Style\_Index and Missing value treatment

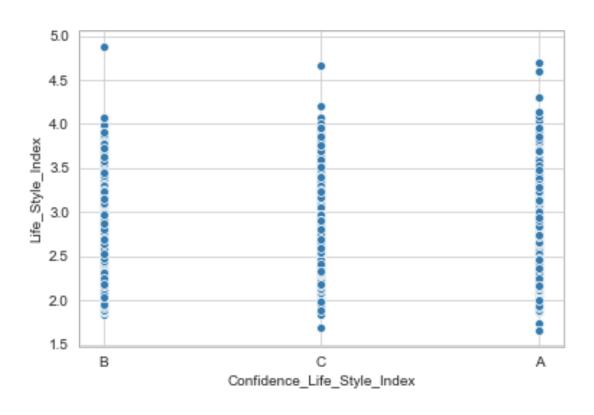


### INSIGHTS on Life\_Style\_index

From these scatter plots, it can be seen that most of the values of Life\_style\_index are distributed between 2 to 3.5

For simplicity, we can replace nan values with mean i.e. 2.8

### EDA on Confidence\_Life\_Style\_Index and Missing value treatment



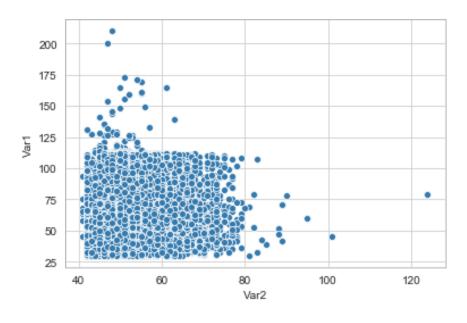
#### INSIGHTS on Confidence\_Life\_Style\_index

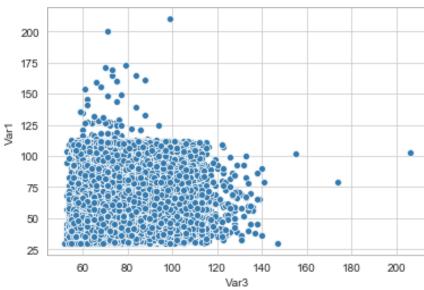
Confidence\_Life\_Style\_index is a category column showing customer confidence based on Life\_style\_index.

From the scatter plot, it can be seen that most of the values of Confidence\_Life\_Style\_index are randomly distributed with Life\_style\_index.

For simplicity, we can replace nan values randomly with A,B or C.

### **EDA on Var1 and Missing value treatment**



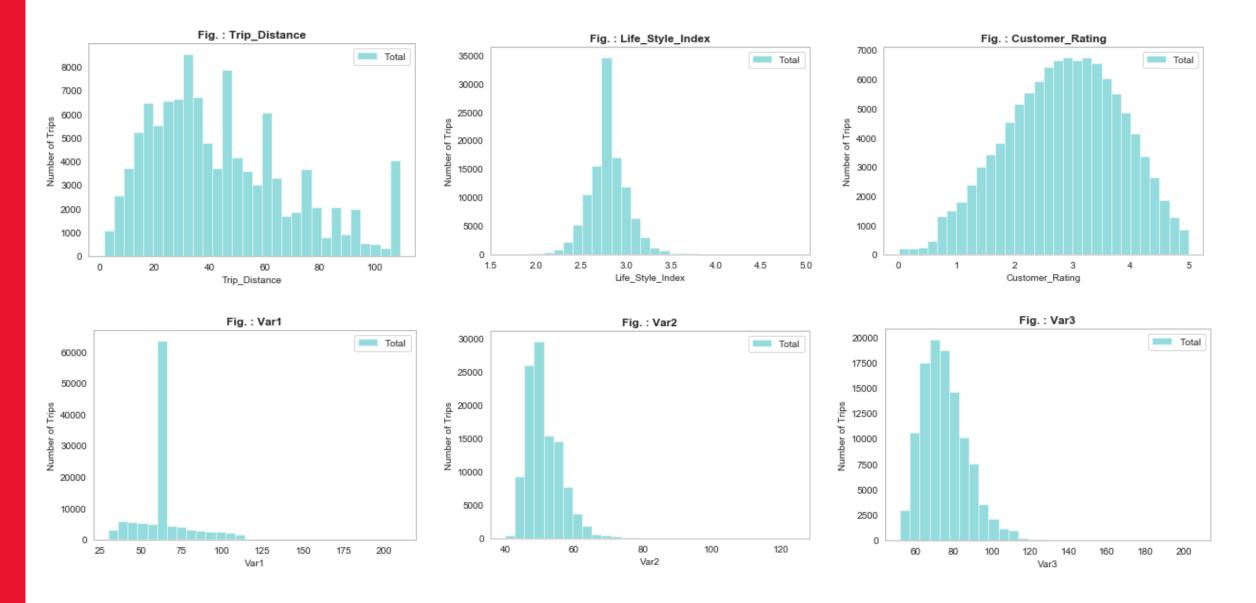


#### INSIGHTS on Var1

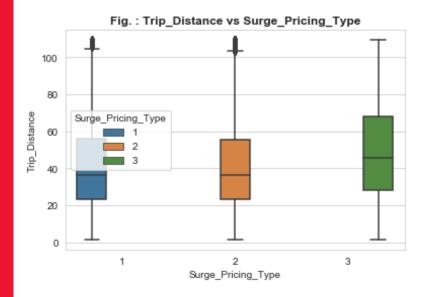
From these scatter plots, it can be seen that most of the values of Var1 are distributed with Var2 and Var3 between 25 to 112

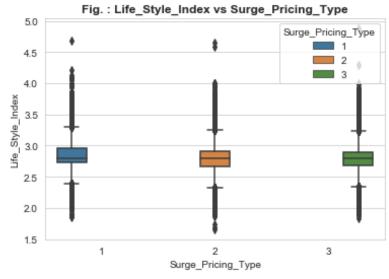
For simplicity, we can replace nan values with mean i.e. 64

### **EDA of Continuous Variables**



### **Bivariate Analysis using Box Plots**



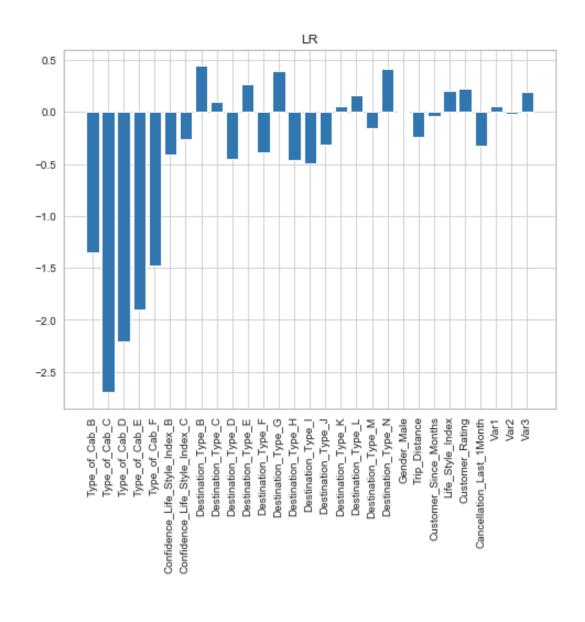


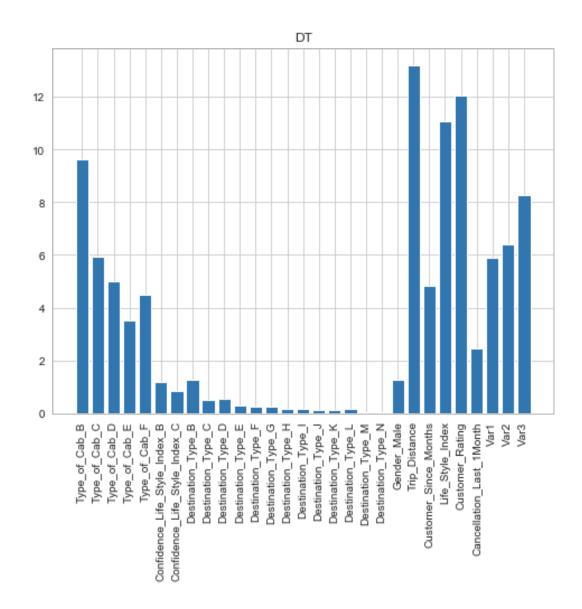


## **Correlation Pair Plot after One-Hot Encoding**

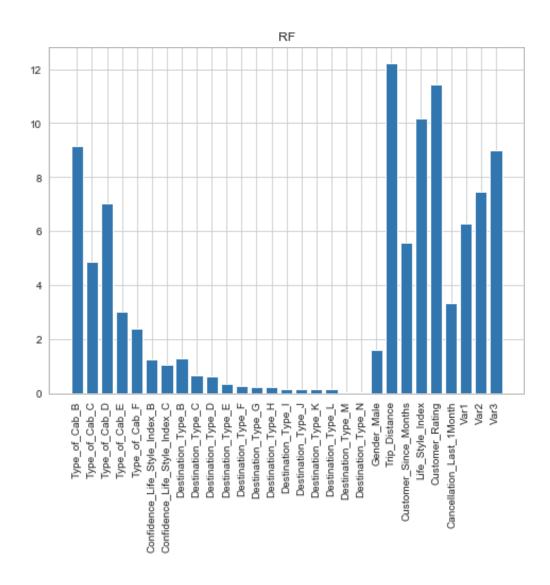
Trip_Distance	1	0.13	0.47	-0.056	-0.0053	-0.034	0.2	0.23	-0.057	-0.01	0.032	0.069	-0.0013	0.23	0.058	0.028	-0.012	-0.13	-0.07	0.012	-0.079	-0.065	-0.067	-0.072	-0.033	0.0076	-0.0082	-0.028	0.0036
Customer_Since_Months	0.13	1	0.16	-0.036	-0.0023	-0.0097	0.056	0.11	-0.0012	0.0053	0.0026	0.0011	-0.00076	0.0083	0.015	-0.014	0.026	-0.017	-0.027	-0.014	-0.023	-0.01	-0.036	-0.0033	-0.029	-0.01	0.0006	-0.021	0.0025
Life_Style_Index	0.47	0.16	1	0.19	0.07	-0.057	0.21	0.3	-0.0053	-0.031	-0.024	0.00022	0.00025	0.07	0.059	-0.0043	0.071	-0.047	-0.0066	0.022	-0.026	-0.0089	-0.018	-0.017	-0.0066	0.0046	0.009	-0.023	0.00074
Customer_Rating	-0.056	-0.036	0.19	1	0.0039	-0.0078	-0.3	-0.23	0.031	-0.032	-0.054	-0.035	-0.0015	-0.075	-0.0081	-0.021	0.087	0.043	0.068	0.056	0.058	0.014	0.023	0.013	0.029	0.033	0.0082	0.019	0.00094
Cancellation_Last_1Month	-0.0053	-0.0023	0.07	0.0039	1	0.0077	0.096	0.13	-0.058	0.025	0.055	0.048	0.0027	0.052	-0.0059	-0.041	0.05	0.01	0.019	0.03	0.0042	0.0033	0.0075	0.0022	0.013	0.037	0.0024	0.0054	0.0047
Var1	-0.034	-0.0097	-0.057	-0.0078	0.0077	1	-0.034	-0.043	0.0026	-0.0049	-0.0089	-0.0071	0.0088	0.01	-0.0087	0.0062	-0.01	0.0047	-0.00068	-0.0067	0.0029	-0.0024	0.00093	0.002	0.0026	0.00015	0.00055	-0.0041	-0.0002
Var2	0.2	0.056	0.21	-0.3	0.096	-0.034	1		-0.0072	-0.0049	0.0053	0.00051	-0.00026	0.044	0.011	0.03	-0.026	-0.057	-0.044	-0.033	-0.042	-0.016	-0.025	-0.021	-0.016	-0.015	-0.00027	-0.011	0.0028
Var3	0.23	0.11	0.3	-0.23	0.13	-0.043	0.68		-0.003	-0.012	-0.01	-0.0046	5.2e-05	0.066	0.0089	0.0021	0.016	-0.051	-0.036	-0.024	-0.032	-0.0072	-0.027	-0.016	-0.029	-0.0069	-0.0042	-0.015	0.0049
Type_of_Cab_B	-0.057	-0.0012	-0.0053	0.031	-0.058	0.0026	-0.0072	-0.003	1	-0.29	-0.23	-0.17	-0.24	-0.035	-0.022	0.052	-0.0033	-0.033	-0.0027	-0.015	-0.0017	-0.013	-0.011	-0.011	-0.0073	-0.0058	-0.0029	0.006	-0.00044
Type_of_Cab_C	-0.01	0.0053	-0.031	-0.032	0.025	-0.0049	-0.0049	-0.012	-0.29	1	-0.21	-0.16	-0.22	0.017	0.0081	-0.009	-0.0014	-0.01	-0.016	-0.012	-0.0047	-0.0046	-0.0054	-0.0036	-0.0095	-0.0034	-0.0024	-0.00074	-0.00042
Type_of_Cab_D	0.032	0.0026	-0.024	-0.054	0.055	-0.0089	0.0053	-0.01	-0.23	-0.21	1	-0.13	-0.17	0.023	0.023	-0.056	-0.0057	0.032	-0.0034	0.014	-0.0067	0.013	0.013	0.0092	0.004	-0.0012	-0.00064	-0.008	0.00085
Type_of_Cab_E	0.069	0.0011	0.00022	-0.035	0.048	-0.0071	0.00051	-0.0046	-0.17	-0.16	-0.13		-0.13	0.044	0.014	-0.047	-0.0045	0.034	-0.0031	0.022	-0.013	0.018	0.014	0.012	0.0048	0.0063	0.003	-0.0044	0.00066
Type_of_Cab_F	-0.0013	-0.00076	0.00025	-0.0015	0.0027	0.0088	-0.00026	5.2e-05	-0.24	-0.22	-0.17	-0.13	1	-0.0015	0.00074	-0.0049	-0.0023	-0.001	-0.0032	-0.0019	0.0024	-0.0014	-0.0042	0.0001	0.0058	0.0017	-0.00032	-0.0041	-0.00058
Confidence_Life_Style_Index_B	0.23	0.0083	0.07	-0.075	0.052	0.01	0.044	0.066	-0.035	0.017	0.023	0.044	-0.0015	1		-0.021	0.0087	-0.02	-0.013	0.033	-0.021	-0.01	-0.014	-0.008	-0.0027	0.003	-0.0029	-0.01	0.0023
Confidence_Life_Style_Index_C	0.058	0.015	0.059	-0.0081	-0.0059	-0.0087	0.011	0.0089	-0.022	0.0081	0.023	0.014	0.00074	-0.57	1	0.0013	-0.0049	-0.0094	-0.01	-0.0037	-0.0053	-0.005	-0.0037	-0.0065	-0.01	0.0025	0.0061	-0.011	0.0039
Destination_Type_B	0.028	-0.014	-0.0043	-0.021	-0.041	0.0062	0.03	0.0021	0.052	-0.009	-0.056	-0.047	-0.0049	-0.021	0.0013	1	-0.13	-0.12	-0.079	-0.066	-0.058	-0.053	-0.043	-0.039	-0.039	-0.038	-0.014	-0.015	-0.0019
Destination_Type_C	-0.012	0.026	0.071	0.087	0.05	-0.01	-0.026	0.016	-0.0033	-0.0014	-0.0057	-0.0045	-0.0023	0.0087	-0.0049	-0.13	1	-0.057	-0.036	-0.03	-0.026	-0.024	-0.02	-0.018	-0.018	-0.017	-0.0064	-0.007	0.0047
Destination_Type_D	-0.13	-0.017	-0.047	0.043	0.01	0.0047	-0.057	-0.051	-0.033	-0.01	0.032	0.034	-0.001	-0.02	-0.0094	-0.12	-0.057	1	-0.034	-0.028	-0.025	-0.022	-0.018	-0.017	-0.017	-0.016	-0.006	-0.0065	0.0016
Destination_Type_E	-0.07	-0.027	-0.0066	0.068	0.019	-0.00068	-0.044	-0.036	-0.0027	-0.016	-0.0034	-0.0031	-0.0032	-0.013	-0.01	-0.079	-0.036	-0.034	1	-0.018	-0.016	-0.014	-0.012	-0.011	-0.011	-0.01	-0.0038	-0.0042	0.0023
Destination_Type_F	0.012	-0.014	0.022	0.056	0.03	-0.0067	-0.033	-0.024	-0.015	-0.012	0.014	0.022	-0.0019	0.033	-0.0037	-0.066	-0.03	-0.028	-0.018	1	-0.013	-0.012	-0.0097	-0.009	-0.0089	-0.0087	-0.0032	-0.0035	-0.00047
Destination_Type_G	-0.079	-0.023	-0.026	0.058	0.0042	0.0029	-0.042	-0.032	-0.0017	-0.0047	-0.0067	-0.013	0.0024	-0.021	-0.0053	-0.058	-0.026	-0.025	-0.016	-0.013	1	-0.01	-0.0085	-0.0079	-0.0078	-0.0076	-0.0028	-0.0031	0.00092
Destination_Type_H	-0.065	-0.01	-0.0089	0.014	0.0033	-0.0024	-0.016	-0.0072	-0.013	-0.0046	0.013	0.018	-0.0014	-0.01	-0.005	-0.053	-0.024	-0.022	-0.014	-0.012	-0.01	1	-0.0078	-0.0072	-0.0071	-0.0069	-0.0025	-0.0028	-0.0025
Destination_Type_I	-0.067	-0.036	-0.018	0.023	0.0075	0.00093	-0.025	-0.027	-0.011	-0.0054	0.013	0.014	-0.0042	-0.014	-0.0037	-0.043	-0.02	-0.018	-0.012	-0.0097	-0.0085	-0.0078	1	-0.0058	-0.0058	-0.0056	-0.0021	-0.0023	-0.0061
Destination_Type_J	-0.072	-0.0033	-0.017	0.013	0.0022	0.002	-0.021	-0.016	-0.011	-0.0036	0.0092	0.012	0.0001	-0.008	-0.0065	-0.039	-0.018	-0.017	-0.011	-0.009	-0.0079	-0.0072	-0.0058	1	-0.0053	-0.0052	-0.0019	-0.0021	-0.0017
Destination_Type_K	-0.033	-0.029	-0.0066	0.029	0.013	0.0026	-0.016	-0.029	-0.0073	-0.0095	0.004	0.0048	0.0058	-0.0027	-0.01	-0.039	-0.018	-0.017	-0.011	-0.0089	-0.0078	-0.0071	-0.0058	-0.0053	1	-0.0051	-0.0019	-0.0021	-0.0025
Destination_Type_L	0.0076	-0.01	0.0046	0.033	0.037	0.00015	-0.015	-0.0069	-0.0058	-0.0034	-0.0012	0.0063	0.0017	0.003	0.0025	-0.038	-0.017	-0.016	-0.01	-0.0087	-0.0076	-0.0069	-0.0056	-0.0052	-0.0051	1	-0.0018	-0.002	-0.0012
Destination_Type_M	-0.0082	0.0006	0.009	0.0082	0.0024	0.00055	-0.00027	-0.0042	-0.0029	-0.0024	-0.00064	0.003	-0.00032	-0.0029	0.0061	-0.014	-0.0064	-0.006	-0.0038	-0.0032	-0.0028	-0.0025	-0.0021	-0.0019	-0.0019	-0.0018	1	-0.00074	-0.0014
Destination_Type_N	-0.028	-0.021	-0.023	0.019	0.0054	-0.0041	-0.011	-0.015	0.006	-0.00074	-0.008	-0.0044	-0.0041	-0.01	-0.011	-0.015	-0.007	-0.0065	-0.0042	-0.0035	-0.0031	-0.0028	-0.0023	-0.0021	-0.0021	-0.002	-0.00074	1	-0.0018
Gender_Male	0.0036	0.0025	0.00074	0.00094	0.0047	-0.0002	0.0028	0.0049	-0.00044	-0.00042	0.00085	0.00066	-0.00058	0.0023	0.0039	-0.0019	0.0047	0.0016	0.0023	-0.00047	0.00092	-0.0025	-0.0061	-0.0017	-0.0025	-0.0012	-0.0014	-0.0018	1
	8	2	X	2	é	Ē	22	13	ω,	0	0	w <sub>j</sub>	4	89	0	ω,	0	ο,	ш	ш,	0	Ξ,	-	2	¥	-5	≥	z,	alo

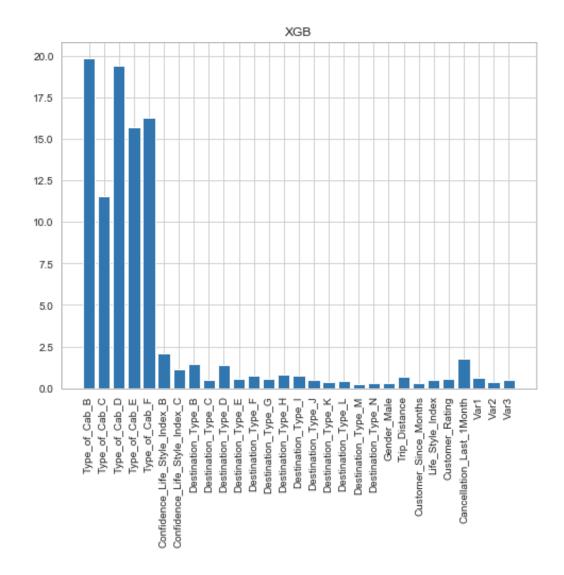
### Feature Significance Graphs on Logistic Regression and Decision Trees



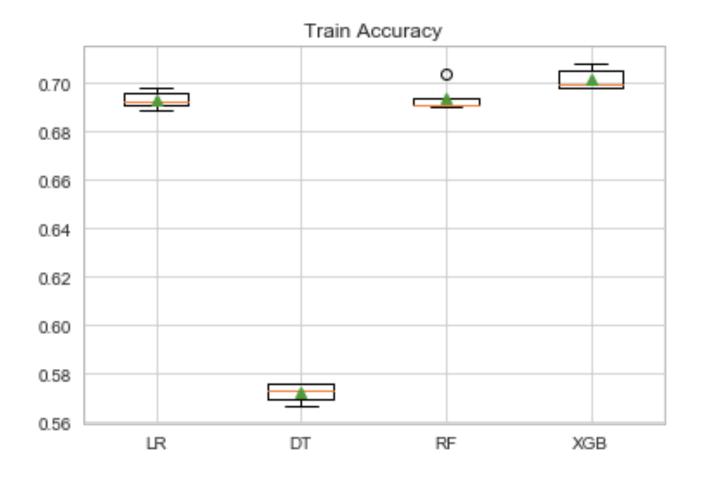


### **Feature Significance Graphs on Random Forest and XGBoost**





### **Algorithm Comparison Graphs**



#### INSIGHTS

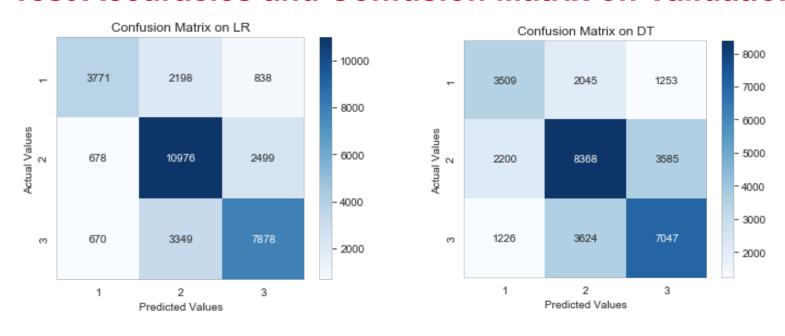
Stratified K-fold Cross Validation is used to avoid changes in train accuracies each time we train the model and also enable some proportion of each target class in train and validation sets.

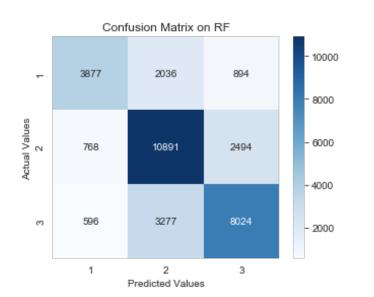
Train Accuracies with different models are -

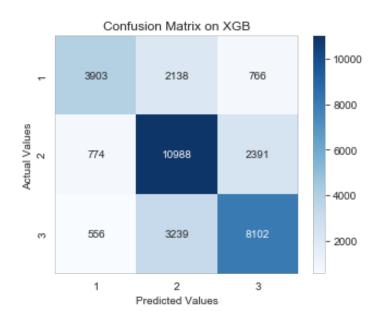
LR: 69.34% DT: 57.23% RF: 69.42% XGB: 70.19%

Highest train accuracies are achieved with XGB so we can select XGB as our base model and further try enhancing the accuracies by hyper-parameter optimisation of XGB.

### **Test Accuracies and Confusion Matrix on Validation Dataset**







The Evaluation Matrix that we have chosen are -

- Confusion Matrix
- Precision
- Recall
- F1-score

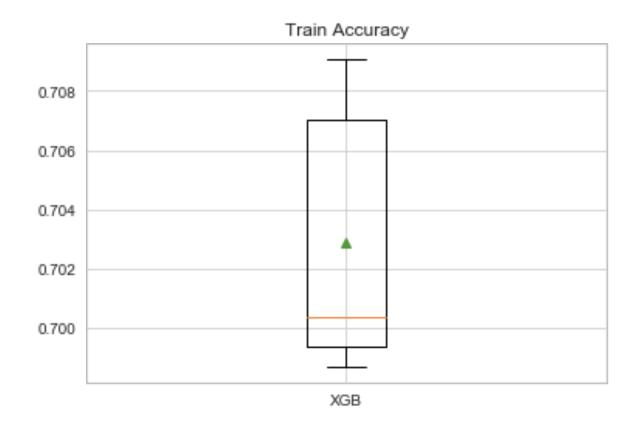
Accuracy is used when there are nearly equal number of samples belonging to each class.

But here, in our case, class 2 of the target variable holds the majority and there are not equal sample distribution in 3 classes, so we use the above mentioned metrics to validate our predictions.

Test Accuracies with different models are -

LR: 68.86% DT: 57.60% RF: 69.37% XGB: 69.98%

### **Train Accuracy with XGBoost**

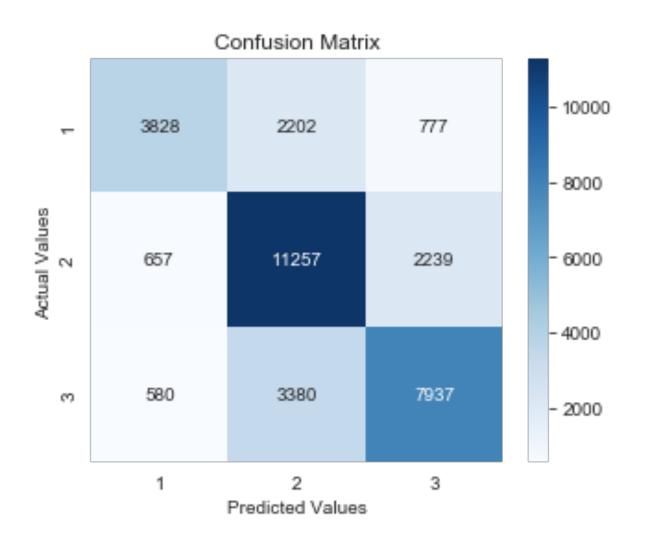


Optimising the hyper parameters, we are able to enhance the train accuracy of XGB to 70.29%.

We have selected XGB as our base model due to the following reasons -

- Gave highest train-test accuracy on our data compared to others
- Ensemble boosting algorithm with base models (DT) added sequentially to reduce the residuals to a great extent
- Fast algorithm

### **Confusion Matrix on XGBoost**



New Test Accuracy of XGB after hyper parameter optimisation -

XGB: 70.07%

# THANK YOU