# Rating prediction Model

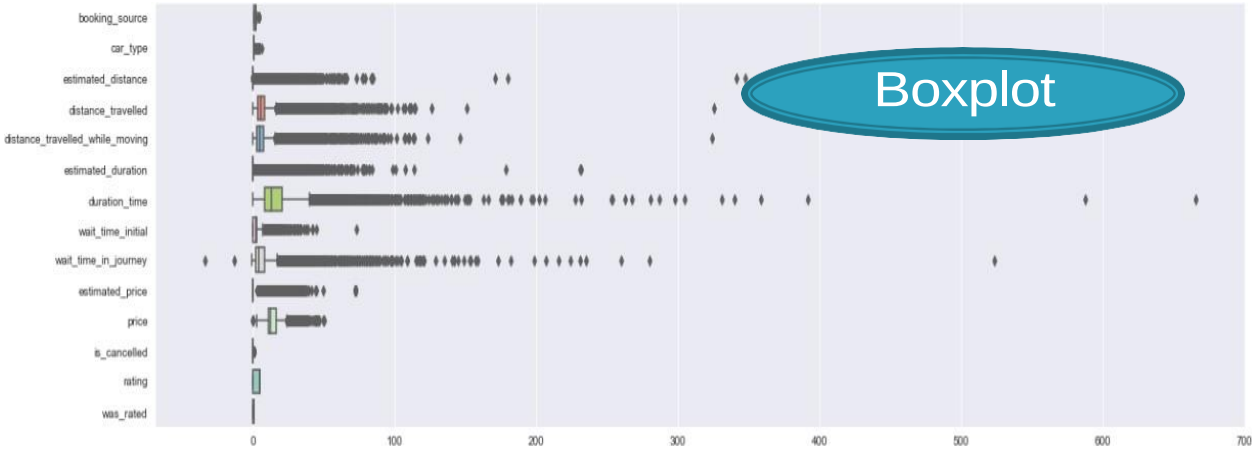
# Road Map

- Objective
- Exploratory Data Analysis
  - Univariate Analysis
  - Bivariate analysis
- Social Network Analysis
  - Customer Network
  - Network Central tendency : Graph
- Are People liking Speedy Rides ?
- Model Development
  - Feature Engineering
    - Feature generation
    - Feature transformation
      - Log/Linear Transformation based on chi square test
    - Feature selection
      - Sign Reversal
      - L1 Regularization (Ridge Regression)
  - Model development
    - Train/Test Split
    - ML algorithm selection (ensemble of LR,RF & GBM)
      - K-fold cross validation
    - Parameter optimization
      - GridsearchCV
    - Model fit
  - Model Validation
    - Decile analysis
    - Validation chart : ROC Curve
    - Confusion Matrix : Threshold Calculation
    - Variable importance
    - Recommendations
  - Scoring
    - Model Object Saving & Loading

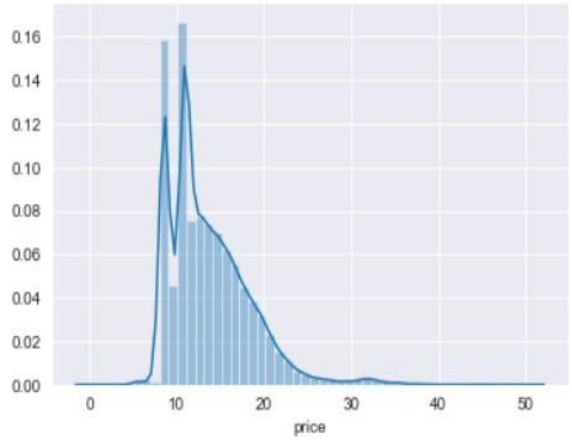
# Objective

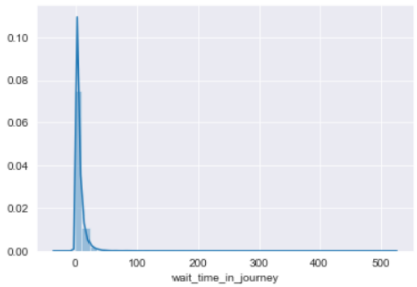
1. Develop a predictive model that will calculate the likelihood of rating the trip by the customer.
2. Need to find out if customer are liking the speedy rides.

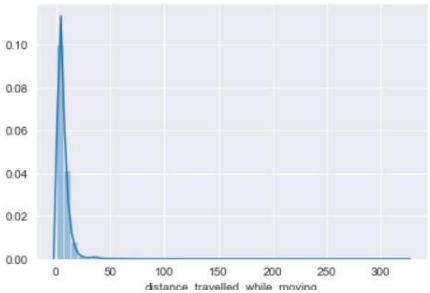
# Exploratory Data Analysis – Univariate Analysis



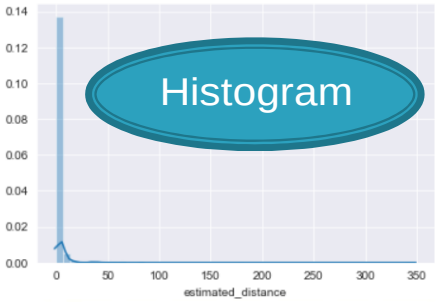

Boxplot

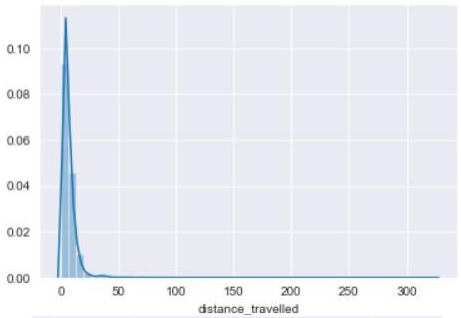


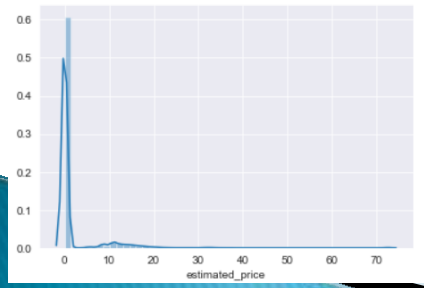



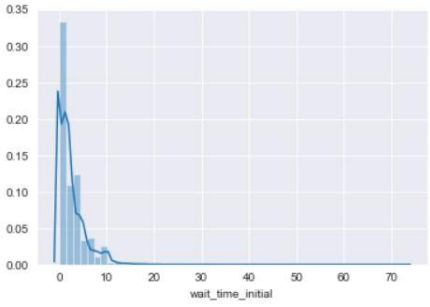



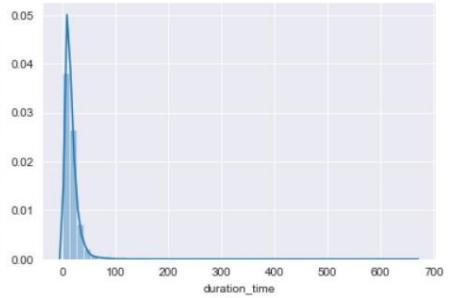

Histogram

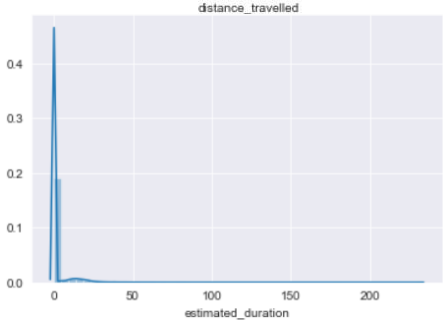


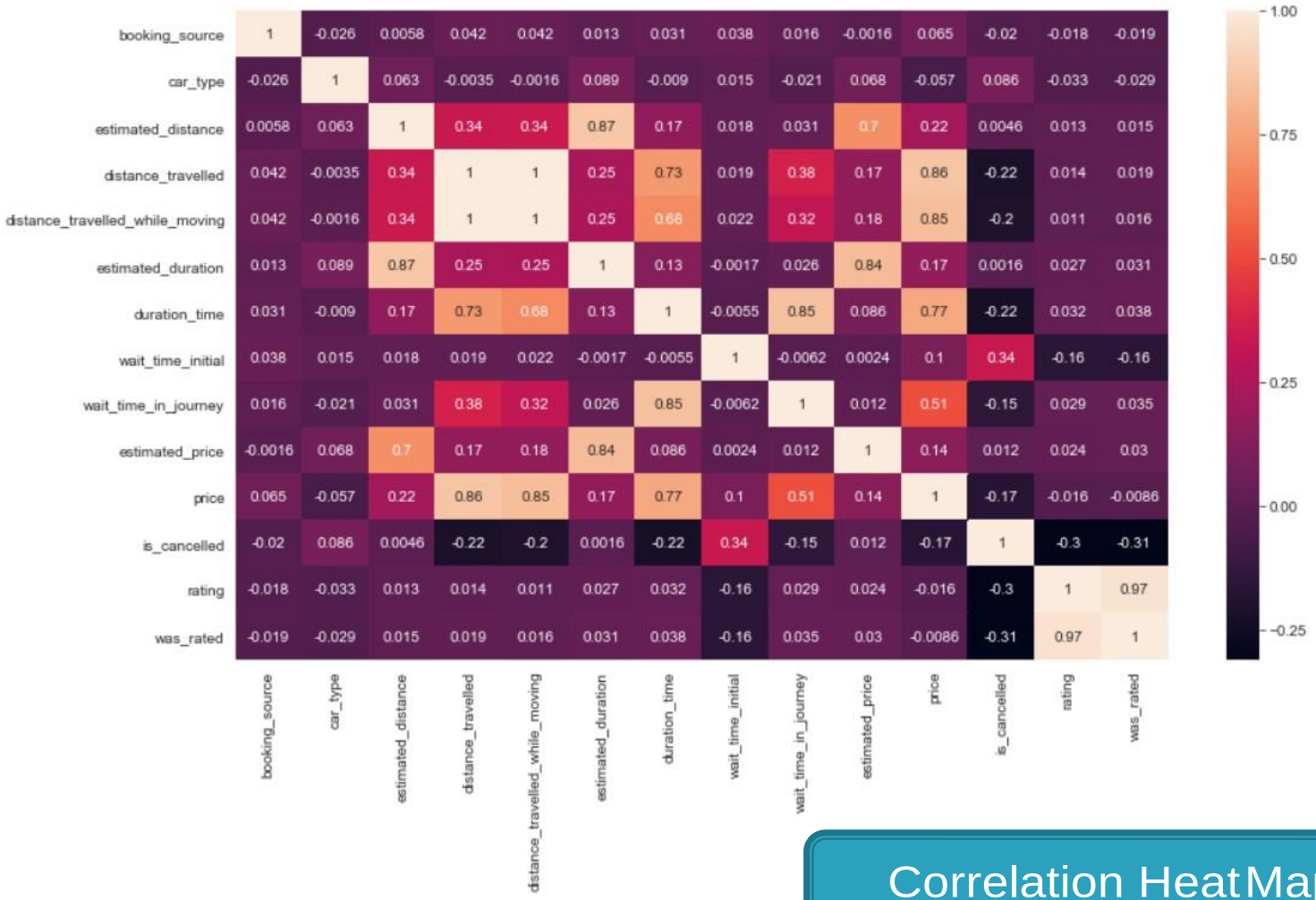



# Exploratory Data Analysis – Bivariate Analysis




Correlation HeatMap

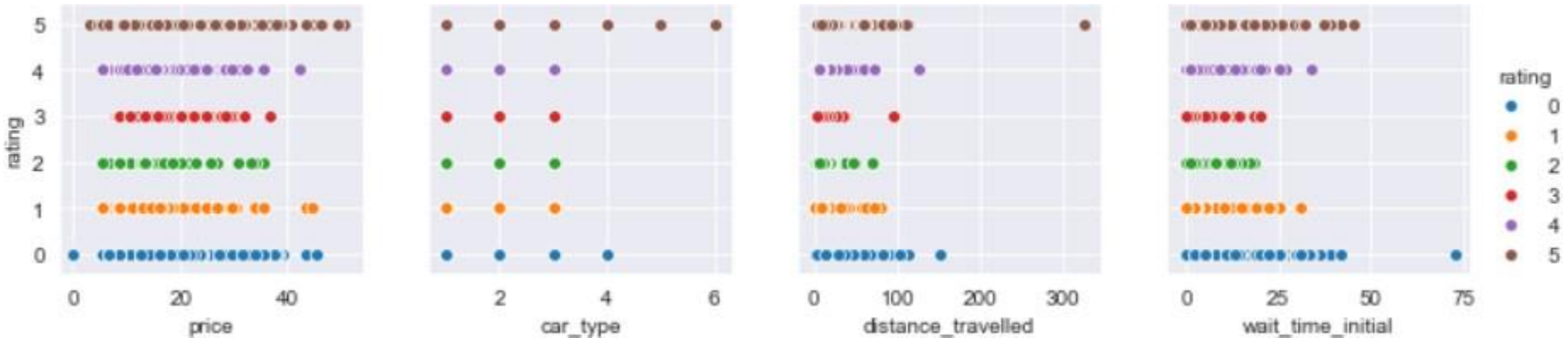
# Exploratory Data Analysis – Bivariate Analysis

## Rating Vs Cartype

| | Rating | | | | | |
|---|---|---|---|---|---|---|
| car_type | 0 | 1 | 2 | 3 | 4 | 5 |
| 1 | 15079 | 647 | 193 | 533 | 2887 | 28592 |
| 2 | 159 | 11 | 4 | 12 | 29 | 428 |
| 3 | 954 | 48 | 14 | 31 | 170 | 1268 |
| 4 | 14 | | | | | 8 |
| 5 | | | | | | 1 |
| 6 | | | | | | 1 |

| Car Type | Average of rating | Count of customer_id |
|---|---|---|
| 1 | 3.28 | 47931 |
| 2 | 3.59 | 643 |
| 3 | 2.89 | 2485 |
| 4 | 1.82 | 22 |
| 5 | 5.00 | 1 |
| 6 | 5.00 | 1 |
| Grand Total | 3.263120803 | 51083 |

## Pair plot



# Are Customers Liking Speedy Rides ?

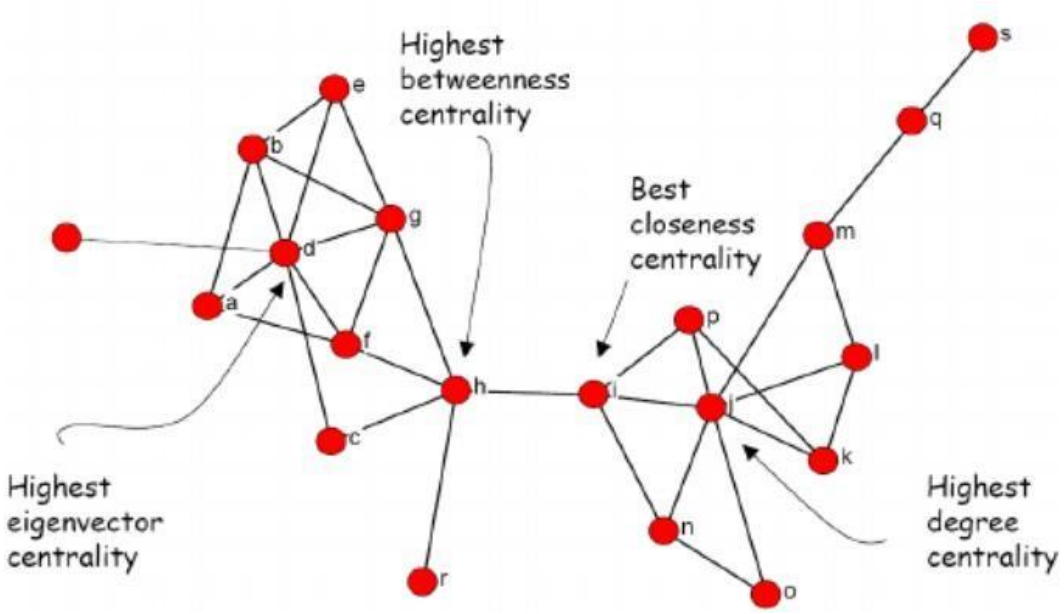
- Yes ! People are liking speedy rides.

| Rating | High_speed | Low_speed | Grand Total | PctOfTotal |
|---|---|---|---|---|
| 1 | 514 | 192 | 706 | 72.8% |
| 2 | 164 | 47 | 211 | 77.7% |
| 3 | 433 | 143 | 576 | 75.2% |
| 4 | 2429 | 657 | 3086 | 78.7% |
| 5 | 23672 | 6626 | 30298 | 78.1% |
| **Grand Total** | **27212** | **7665** | **34877** | |

- Speedy ride was calculated as percentage of moving distance of total distance
- High-speed & Low speed were calculated based on their Mean.
- And as per analysis people are giving rating 5 having around 79% speedy rides.

# SNA Centrality Measure

- Degree :
  - The number of edges connected to a node
  - Exposure to the network , opportunity to directly influence.

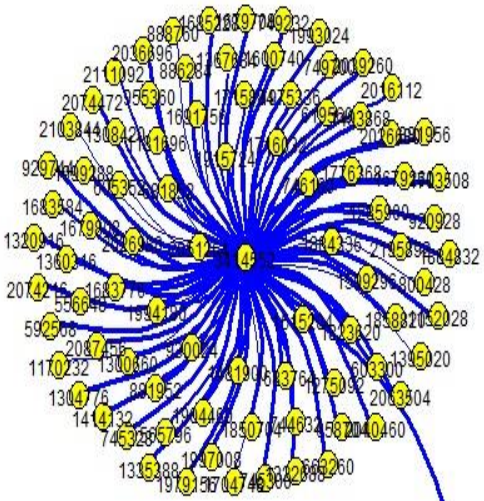



- Betweenness :
  - Extent to which a particular node lies on the shortest path between other nodes
  - Informal power, gate keeping , control flow of info.
- Closeness :
  - The average of the shortest distances to all other nodes in the graph. Estimates time to hear info, indirect influence, point of rapid diffusion
- Eigenvector :
  - A message of the extent to which a node is connected to influential other nodes. connected to influential nodes of high degree. ”not what you know but who you know

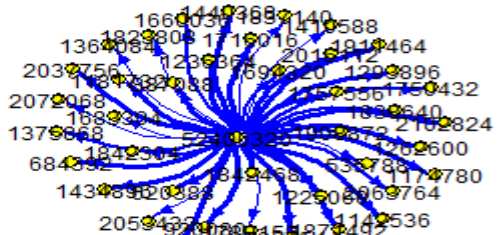
# SNA Outcome

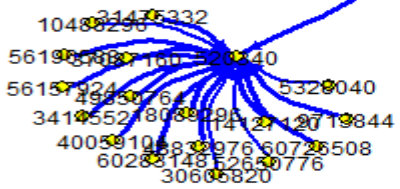
| customer_id | driver_id | rating | Degree | Closeness | Betweenness |
|---|---|---|---|---|---|
| 3414552 | 520340 | 5 | 86 | 53286.98 | 3274913 |
| 3414552 | 556648 | 2 | 86 | 53286.98 | 3274913 |
| 3414552 | 565796 | 4 | 86 | 53286.98 | 3274913 |
| 3414552 | 592568 | 5 | 86 | 53286.98 | 3274913 |
| 3414552 | 603300 | 5 | 86 | 53286.98 | 3274913 |
| 3414552 | 605352 | 1 | 86 | 53286.98 | 3274913 |

Network with Betweenness



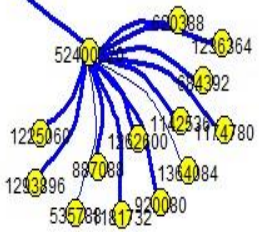
Sub Network with Betweenness





520340

Top 20 customer having high betweenness



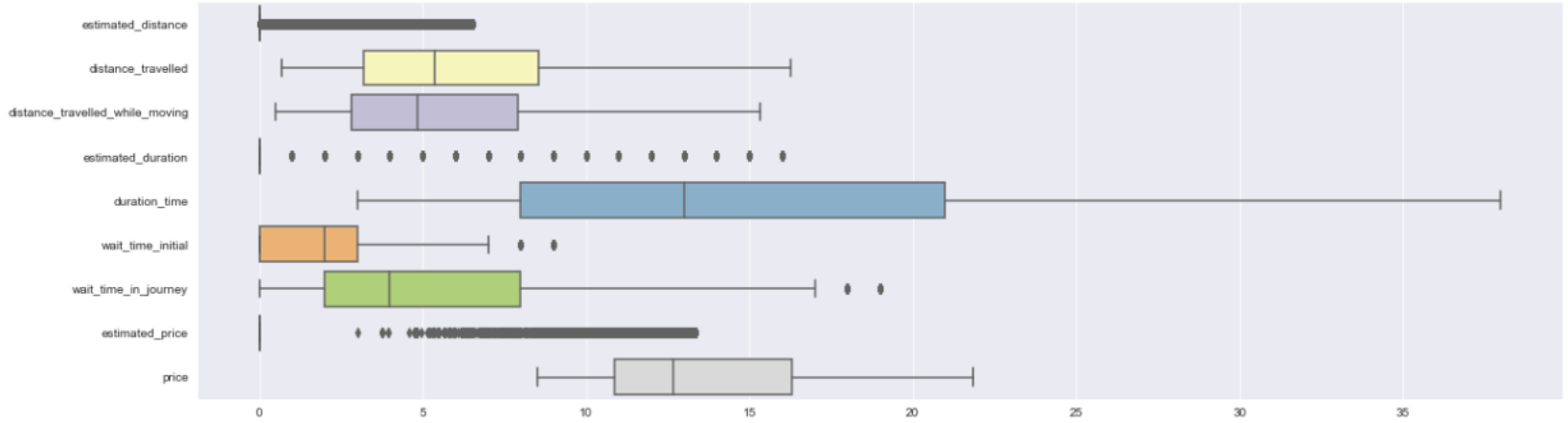
# Model Development

- Feature Engineering
  - Outlier Treatment
  - Derived Variables
  - Transformation
  - Feature Reduction
    - Correlation Analysis
    - Sign check
    - L1 regularization (ridge regression)
    - P–value

# Feature Engineering –Outlier Treatment

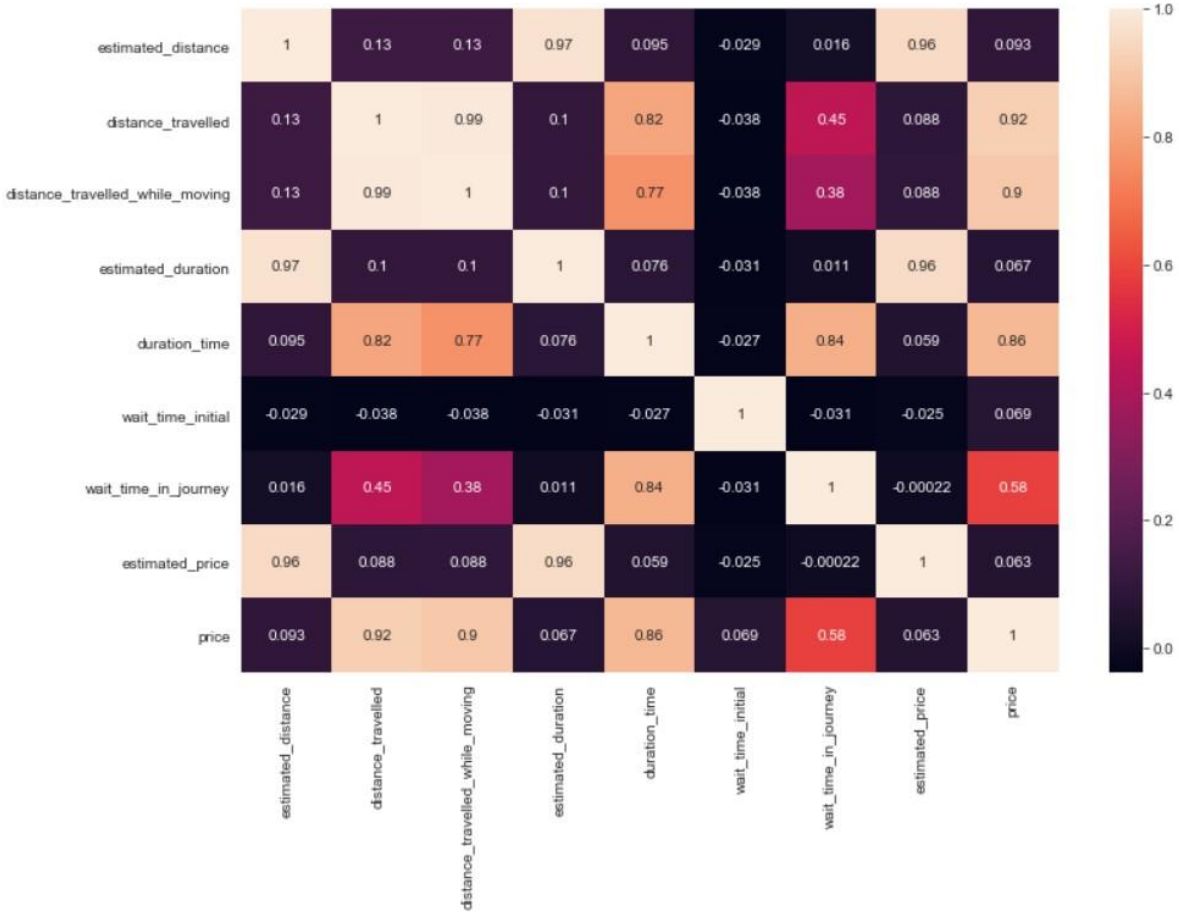
- Outlier Treatment was performed using 95$^{th}$ percentile ceiling & 5$^{th}$ percentile flooring method
- variables treated for outlier.
- Below is the boxplot after the outlier treatment

| |
|---|
| estimated_distance |
| distance_travelled |
| distance_travelled_while_moving |
| estimated_duration |
| duration_time |
| wait_time_initial |
| wait_time_in_journey |
| estimated_price |
| price |



# FeatureEngineering – Correlation Analysis

- Correlation Graph after outlier treatment



# Feature Engineering – Derived Feature

- Feature Generation
  - Below are the new derived features
    1. time_hour: booking time broken into hour of booking
    2. dayofmonth: booing day was extracted from creation day.
    3. trafic_distance : difference between distance travelled & distance_travelled_while_moving
    4. speedpct : distance_travelled_while_moving/ distance travelled
    5. Totwaittm : wait_time_initial/ wait_time_in_journey

```
### time_hour = hour of (creation date)
train['time_hour'] = train.creation_date.dt.hour
test['time_hour'] = test.creation_date.dt.hour

## dayofmonth = dayofmonth of creation date
train['dayofmonth'] = train.creation_date.dt.day
test['dayofmonth'] = test.creation_date.dt.day

### trafic_distance = (distance_travelled - distance_travelled_while_moving)
train['trafic_distance'] = train["distance_travelled"] - train["distance_travelled_while_moving"]
test['trafic_distance'] = test["distance_travelled"] - test["distance_travelled_while_moving"]

#### speedpct = distance_travelled_while_moving/distance_travelled
train['speedpct'] = train["distance_travelled_while_moving"]/train["distance_travelled"]
test['speedpct'] = train["distance_travelled_while_moving"]/test["distance_travelled"]

#### totwaittm = wait_time_initial  + wait_time_in_journey
train['totwaittm'] = train["wait_time_initial"] + train["wait_time_in_journey"]
test['totwaittm'] = test["wait_time_initial"] + test["wait_time_in_journey"]
```

# Feature Engineering – Transformation & Reduction

- Feature Transformation
  - Log / Linear Transformation done based on best chi-square value.
- Feature Reduction
  - Based on Sign check
  - L1 regularization (Ridge regression)
  - P value
- Final Feature Selected

| Variables |
|---|
| car_type |
| booking_source |
| dayofmonth |
| estimated_distance |
| time_hour |
| price |
| wait_time_initial |
| distance_travelled |

# Algorithm Selection

- ML Parameter optimization was done using GridSearchCV method.
- Algorithm is selected based on model accuracy using K-fold cross validation.
- Random Forest algorithm is selected as it is giving highest accuracy.

```
Model # Random Forest
Mean accuracy score is 0.7240058067247418

Model # Logistic Regression
Mean accuracy score is 0.6816097220745957
```

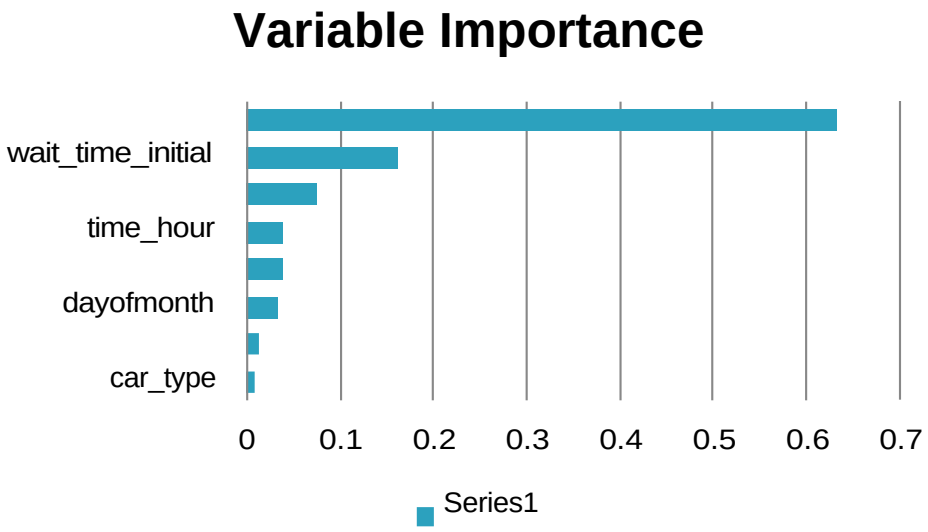
```
paramgrid = {'max_depth': list(range(1, 10, 2)), 'n_estimators': list(range(1, 200, 30))}

grid_search=GridSearchCV(RandomForestClassifier(random_state=1),paramgrid)

x_train_cv, x_test_cv, y_train_cv, y_test_cv = train_test_split(x,y, test_size =0.3, random_state=1)
# Fit the grid search model
grid_search.fit(x_train_cv,y_train_cv)
# Estimating the optimized value
grid_search.best_estimator_
```

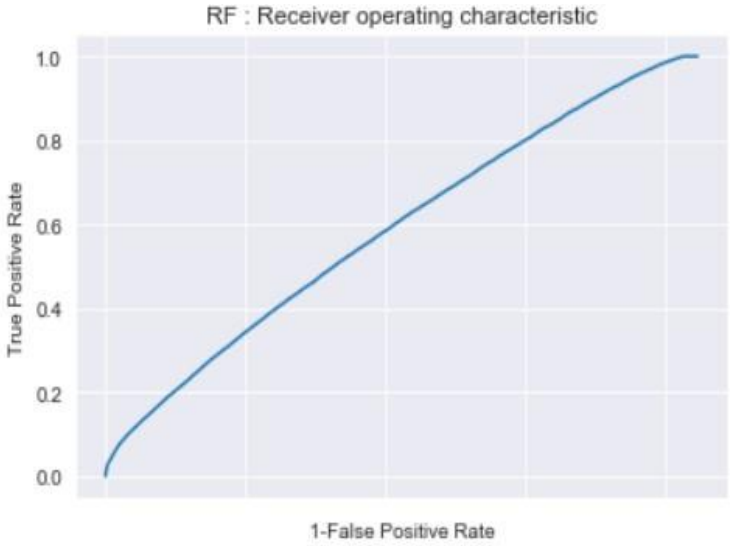
```
RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
            max_depth=3, max_features='auto', max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=1, min_samples_split=2,
            min_weight_fraction_leaf=0.0, n_estimators=51, n_jobs=1,
            oob_score=False, random_state=1, verbose=0, warm_start=False)
```

# Random Forest Outcome – 1




- As per ML model distance travelled ,wait time & price are the measure factor for giving rating to the trip.

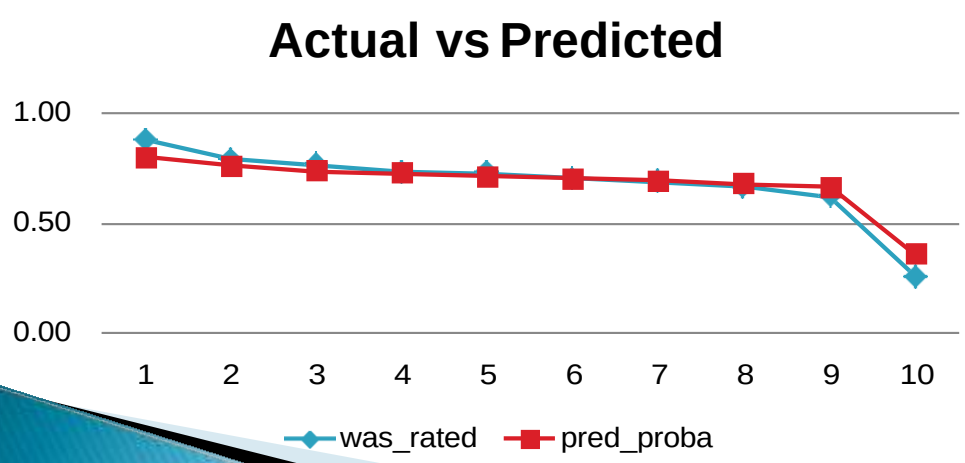
```
RF Accuracy Score 0.7274239962414111
######### RF AUC  #################
Area under the ROC curve RF : 0.677126
[0.7037486]
######### RF Confusion_Matrix  ################
[[ 9952  6254]
 [13461 21416]]
```

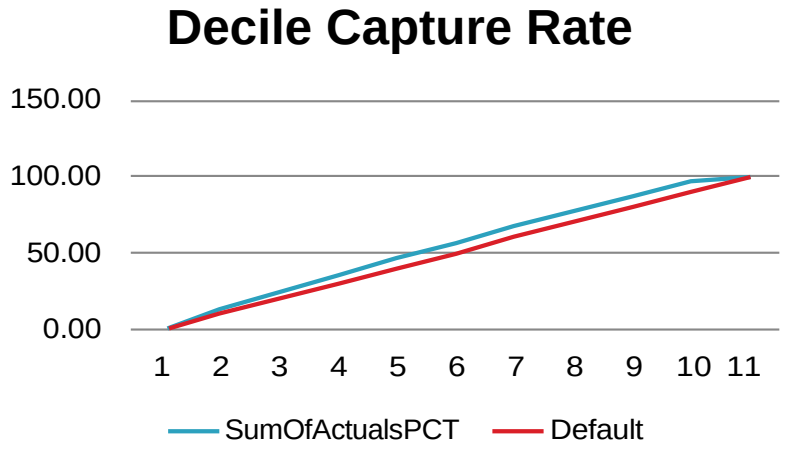


# Random Forest Outcome –2

| Decile | was_rated | pred_proba | SumOfActuals | TotalCount | SumOfActualsCUM | SumOfActualsPCT | Default |
|---|---|---|---|---|---|---|---|
| 0 | 0.88 | 0.80 | 4492 | 5109 | 4492 | 0.00 | 0.00 |
| 1 | 0.79 | 0.76 | 4039 | 5108 | 8531 | 12.88 | 10.00 |
| 2 | 0.76 | 0.74 | 3907 | 5108 | 12438 | 24.46 | 20.00 |
| 3 | 0.73 | 0.73 | 3740 | 5109 | 16178 | 35.66 | 30.00 |
| 4 | 0.73 | 0.71 | 3714 | 5108 | 19892 | 46.39 | 40.00 |
| 5 | 0.71 | 0.70 | 3602 | 5108 | 23494 | 57.03 | 50.00 |
| 6 | 0.69 | 0.69 | 3523 | 5109 | 27017 | 67.36 | 60.00 |
| 7 | 0.66 | 0.68 | 3395 | 5108 | 30412 | 77.46 | 70.00 |
| 8 | 0.62 | 0.66 | 3164 | 5108 | 33576 | 87.20 | 80.00 |
| 9 | 0.25 | 0.36 | 1301 | 5108 | 34877 | 96.27 | 90.00 |
| | | | | | | 100.00 | 100.00 |

Capture rate in the third decile



# Logistic Regression outcome –1

Logit Regression Results

| | | | |
|---|---|---|---|
| Dep. Variable: | was_rated | No. Observations: | 38312 |
| Model: | Logit | Df Residuals: | 38303 |
| Method: | MLE | Df Model: | 8 |
| Date: | Tue, 25 Sep 2018 | Pseudo R-squ.: | 0.07879 |
| Time: | 17:55:28 | Log-Likelihood: | -22052. |
| converged: | True | LL-Null: | -23938. |
| | | LLR p-value: | 0.000 |

| | coef | std err | z | P>\|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| intercept | 2.2404 | 0.080 | 28.069 | 0.000 | 2.084 | 2.397 |
| distance_travelled | 0.3544 | 0.010 | 34.708 | 0.000 | 0.334 | 0.374 |
| estimated_distance | 0.0344 | 0.004 | 9.579 | 0.000 | 0.027 | 0.041 |
| price | -0.0797 | 0.003 | -24.359 | 0.000 | -0.086 | -0.073 |
| wait_time_initial | -0.0494 | 0.005 | -10.384 | 0.000 | -0.059 | -0.040 |
| car_type | -0.1066 | 0.027 | -4.008 | 0.000 | -0.159 | -0.054 |
| time_hour | -0.0108 | 0.002 | -4.538 | 0.000 | -0.015 | -0.006 |
| booking_source | -0.0783 | 0.023 | -3.415 | 0.001 | -0.123 | -0.033 |
| dayofmonth | -0.0039 | 0.001 | -3.028 | 0.002 | -0.006 | -0.001 |

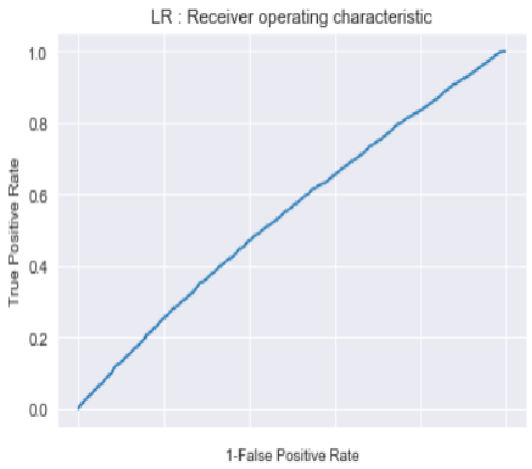
- As per LR model distance travelled is one of the measure factor for rating the trip.

```
######### AUC  ################
Area under the ROC curve : 0.620585
```

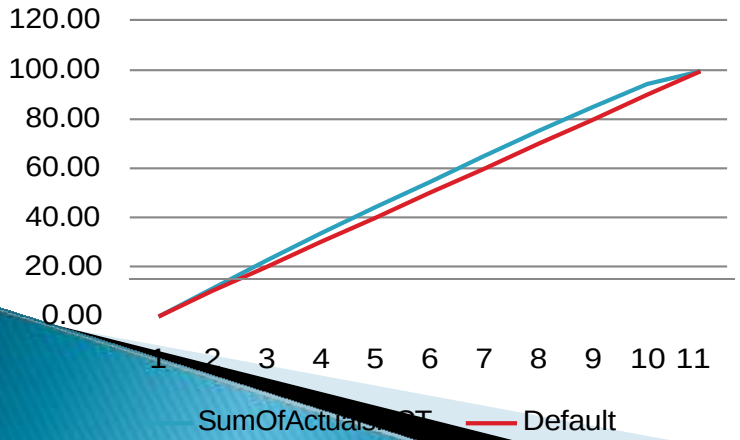
```
######### Confusion_Matrix
[[2341 1708]
 [3682 5040]]
```

# Logistic Regression outcome –2

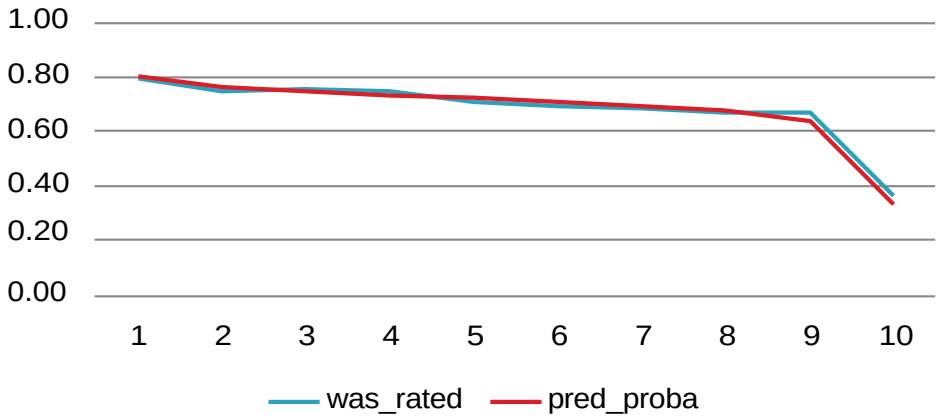
| Decile | was_rated | pred_proba | SumOfActuals | TotalCount | SumOfActualsCUM | SumOfActualsPCT | Default |
|---|---|---|---|---|---|---|---|
| 0 | 0.79 | 0.80 | 1014 | 1278 | 1014 | 0.00 | 0.00 |
| 1 | 0.75 | 0.76 | 956 | 1277 | 1970 | 11.63 | 10.00 |
| 2 | 0.75 | 0.75 | 963 | 1277 | 2933 | 22.59 | 20.00 |
| 3 | 0.74 | 0.73 | 949 | 1277 | 3882 | 33.63 | 30.00 |
| 4 | 0.71 | 0.72 | 903 | 1277 | 4785 | 44.51 | 40.00 |
| 5 | 0.69 | 0.71 | 885 | 1277 | 5670 | 54.86 | 50.00 |
| 6 | 0.69 | 0.69 | 877 | 1277 | 6547 | 65.01 | 60.00 |
| 7 | 0.67 | 0.67 | 850 | 1277 | 7397 | 75.06 | 70.00 |
| 8 | 0.67 | 0.64 | 857 | 1277 | 8254 | 84.81 | 80.00 |
| 9 | 0.37 | 0.33 | 468 | 1277 | 8722 | 94.63 | 90.00 |
| | | | | | | 100.00 | 100.00 |




## Decile capture Rate




## Actual vs Predicted



Thanks