San Francisco Public Bike Sharing

Gaurav Shivhare

22, Feb 2019

Introduction

Bike Sharing have been explored as a very effective way of controlling traffic as well as pollution. The main challenge for the operators is to provide bikes to the user when the need it. Due to limited number of resources and space it is not possible to fulfil the requirement at all places. The probable reasons are the different pattern of usage of bikes based on the weekday, weather etc., which highly impact the scalability of the operator resources and the solution for it, is to reallocate the bikes where it is required the most. Hence, it is required to introduce algorithm to teach the operator where(station) the bikes are required and in this it had tried to find a solution for this problem based on understanding the pattern of usage and the variables impacting the number of trips. This work proposes different solution on the basis of weekday and weekend busy hours and station along with a prediction model which can give the real time requirement of bikes, after training on available data. This work follows the CRISP-DM method for providing the solution (Only First Iteration).

Business Requirements

Provide the optimized Placement/Reallocation schedule to *Bike Operator*.

Data Source : https://www.kaggle.com/benhamner/sf-bay-area-bike-share

Understanding the Data Provided

1. Station.csv - The list of the Stations with id and geographical location
2. Status.csv - Available Docks and Bikes by Date and Time
3. Trip.csv - Ride details of the users trip
4. Weather.csv - Information about the Weather by Date

Tableau Link for Deep Data Diving

https://public.tableau.com/profile/gaurav.shivhare#!/vizhome/SFBayAreaDU/BayAreaBikeSharingAnalysis

Please click on above and use presentation mode for clear view Sheet

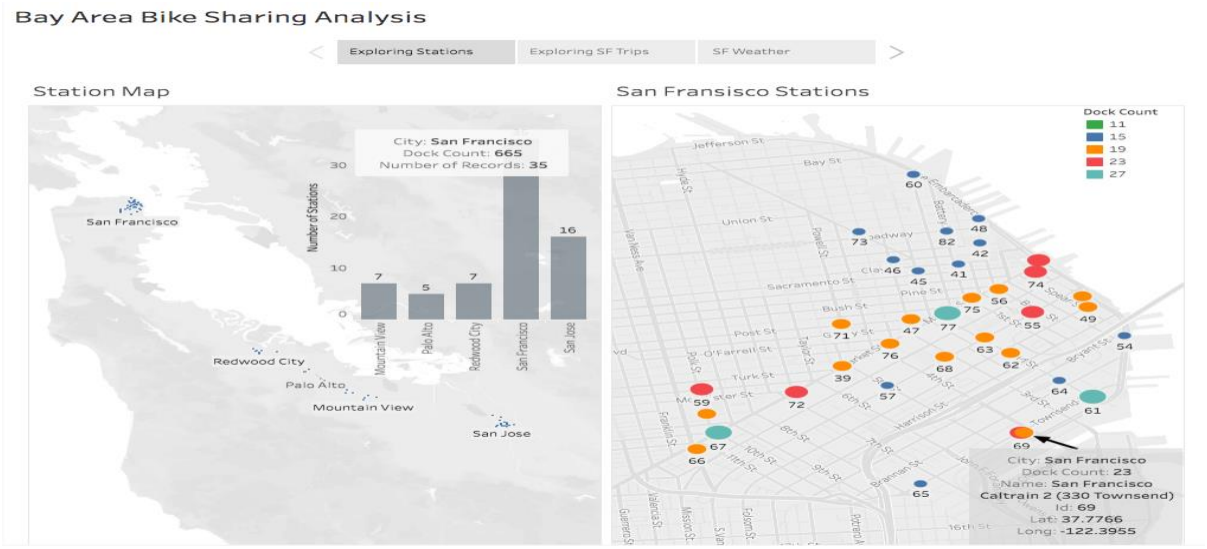


Figure 1 – Sheet 1 of Tableau Story

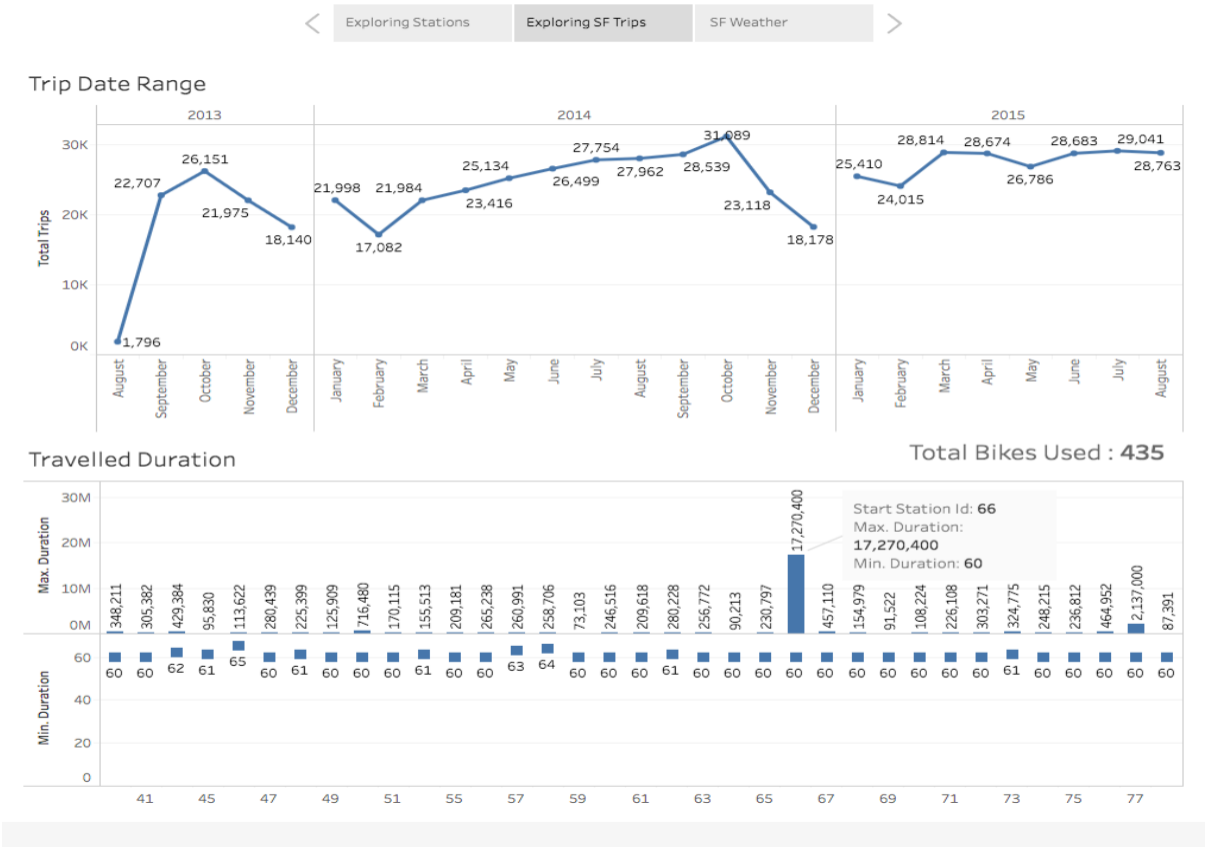
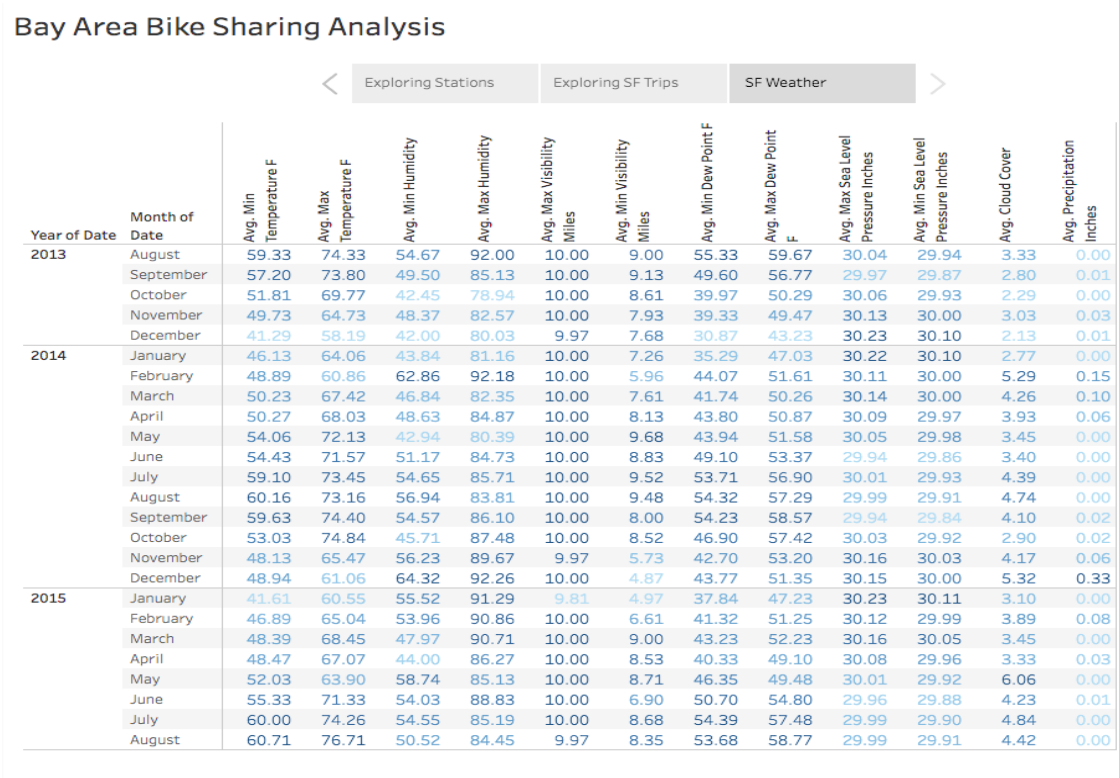


Figure 2 – Sheet 2 of Tableau Story



Year of Date	Month of Date	Avg. Min Temperature F	Avg. Max Temperature F	Avg. Min Humidity	Avg. Max Humidity	Avg. Max Visibility Miles	Avg. Min Visibility Miles	Avg. Min Dew Point F	Avg. Max Dew Point F	Avg. Max Sea Level Pressure Inches	Avg. Min Sea Level Pressure Inches	Avg. Cloud Cover	Avg. Precipitation Inches
2013	August	59.33	74.33	54.67	92.00	10.00	9.00	55.33	59.67	30.04	29.94	3.33	0.00
	September	57.20	73.80	49.50	85.13	10.00	9.13	49.60	56.77	29.97	29.87	2.80	0.01
	October	51.81	69.77	42.45	78.94	10.00	8.61	39.97	50.29	30.06	29.93	2.29	0.00
	November	49.73	64.73	48.37	82.57	10.00	7.93	39.33	49.47	30.13	30.00	3.03	0.03
	December	41.29	58.19	42.00	80.03	9.97	7.68	30.87	43.23	30.23	30.10	2.13	0.01
2014	January	46.13	64.06	43.84	81.16	10.00	7.26	35.29	47.03	30.22	30.10	2.77	0.00
	February	48.89	60.86	62.86	92.18	10.00	5.96	44.07	51.61	30.11	30.00	5.29	0.15
	March	50.23	67.42	46.84	82.35	10.00	7.61	41.74	50.26	30.14	30.00	4.26	0.10
	April	50.27	68.03	48.63	84.87	10.00	8.13	43.80	50.87	30.09	29.97	3.93	0.06
	May	54.06	72.13	42.94	80.39	10.00	9.68	43.94	51.58	30.05	29.98	3.45	0.00
	June	54.43	71.57	51.17	84.73	10.00	8.83	49.10	53.37	29.94	29.86	3.40	0.00
	July	59.10	73.45	54.65	85.71	10.00	9.52	53.71	56.90	30.01	29.93	4.39	0.00
	August	60.16	73.16	56.94	83.81	10.00	9.48	54.32	57.29	29.99	29.91	4.74	0.00
	September	59.63	74.40	54.57	86.10	10.00	8.00	54.23	58.57	29.94	29.84	4.10	0.02
	October	53.03	74.84	45.71	87.48	10.00	8.52	46.90	57.42	30.03	29.92	2.90	0.02
	November	48.13	65.47	56.23	89.67	9.97	5.73	42.70	53.20	30.16	30.03	4.17	0.06
	December	48.94	61.06	64.32	92.26	10.00	4.87	43.77	51.35	30.15	30.00	5.32	0.33
2015	January	41.61	60.55	55.52	91.29	9.81	4.97	37.84	47.23	30.23	30.11	3.10	0.00
	February	46.89	65.04	53.96	90.86	10.00	6.61	41.32	51.25	30.12	29.99	3.89	0.08
	March	48.39	68.45	47.97	90.71	10.00	9.00	43.23	52.23	30.16	30.05	3.45	0.00
	April	48.47	67.07	44.00	86.27	10.00	8.53	40.33	49.10	30.08	29.96	3.33	0.03
	May	52.03	63.90	58.74	85.13	10.00	8.71	46.35	49.48	30.01	29.92	6.06	0.00
	June	55.33	71.33	54.03	88.83	10.00	6.90	50.70	54.80	29.96	29.88	4.23	0.01
	July	60.00	74.26	54.55	85.19	10.00	8.68	54.39	57.48	29.99	29.90	4.84	0.00
	August	60.71	76.71	50.52	84.45	9.97	8.35	53.68	58.77	29.99	29.91	4.42	0.00

Figure 3 – Sheet 3 of Tableau Story

1 : Understanding Stations Sheet [Figure 1]

2 : Understanding Trips Sheet [Figure 2]

3 : Weather Overview [Figure 3]

Inferences:

1. Maximum Docks are available in San Francisco.
2. Cities are quite away from each other, intercity analysis is not useful hence only San Francisco is used.
3. Totals Docks available in San Francisco are 665 while bikes are 435 [Sheet 2] hence ratio between these is 435/665 = 0.65, which means 65% of all docks have bike and 35% are free.
4. We have trip data from Aug 2013 to Aug 2015 i.e. 2 years.
5. Minimum trip time is 60 Seconds while Maximum reaches up to more than a year which clearly shows there are outliers which are needed to be eliminated
6. We have multiple constraints for Weather(like humidity, visibility and temp.) daily stats for all time frame. Note : Status file is ignored due to big size and it is clearly visible it have availability data which is very important can be used for real time bike shifting.

Approach

1. Business --- Based on Compromised Business (Practical)
2. Technical --- Based on Predicting the number of bikes required on real time basis.

Note: Reallocation highly depends on the frequency of truck operator can afford and capacity of the vehicle used.

Approach

1. 7 AM - 9 AM are generally peak hours for routing toward the business areas while 4 PM - 6PM towards the residential area. So Technically we can identify the busy stations during these peak times, so it can be said the stations have starting point in morning hours are residential or connected (Station, Intercity Bus Stop) areas, and similarly the starting points in evening hours are business areas. Hence, these stations can be provided with the efficient number of bikes from the non-busy stations at difference windows by the operators. Identifying these areas will help in reallocation of the bikes for weekdays.

2. On Weekends identify the busy stations and hour for bike allocation.

3. Identify, how weather condition change the usage pattern, can be used to increase/decrease number of bikes accordingly.

4. Ride distances and travels can be used to identify the scope of a new station.

5. Create a model to regress the number of bikes needed at a station at a time by using all constraints.

EDA

Trips

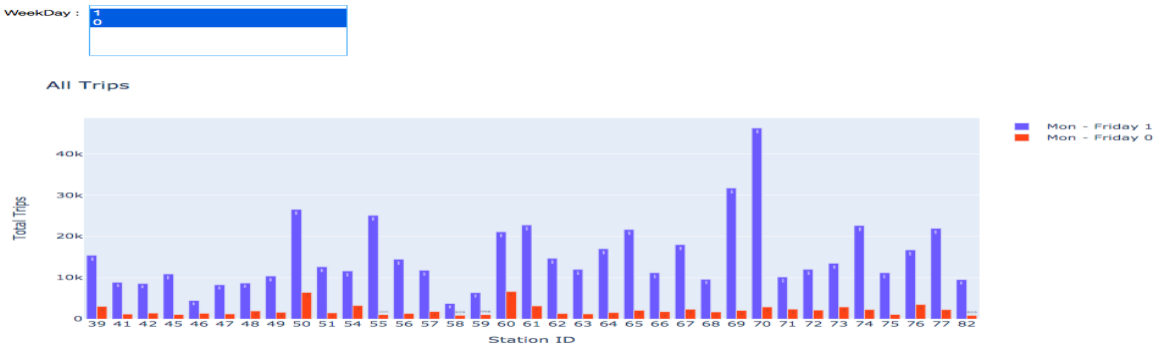


Figure 4 -Bikes trips during Weekdays (Blue) and Weekends (Red)

After this analysis we can say Bikes are quiet often used in Weekdays [Figure 4].

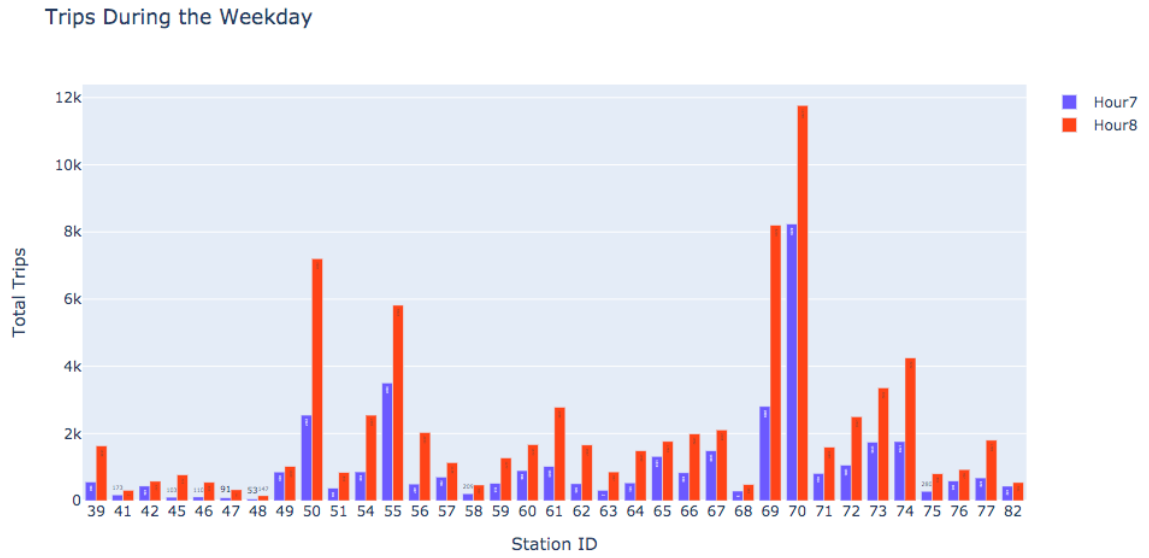


Figure 5 – Bikes usage in Weekdays at Stations during morning peak hours.

In Weekdays, 69 and 70 that's stations near train stations are most busy in morning rush hours hence bikes from nearby stations can be placed on these stations, while evening rush is quite distributed but this approach can be used for all stations at all time.

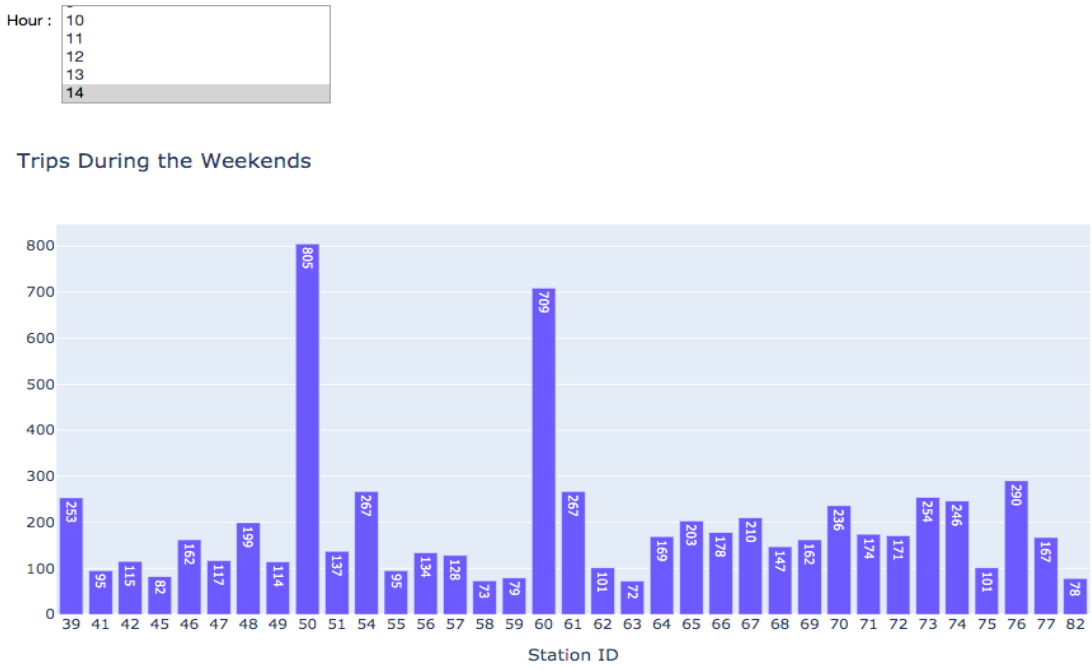


Figure 6 - Bikes usage in Weekends at Stations during morning peak hours.

In Weekends, stations 50 and 60 are busy at noon time (14 and 16 hours) as they are close to bridge and bay hence bikes can be allocated at these sites.

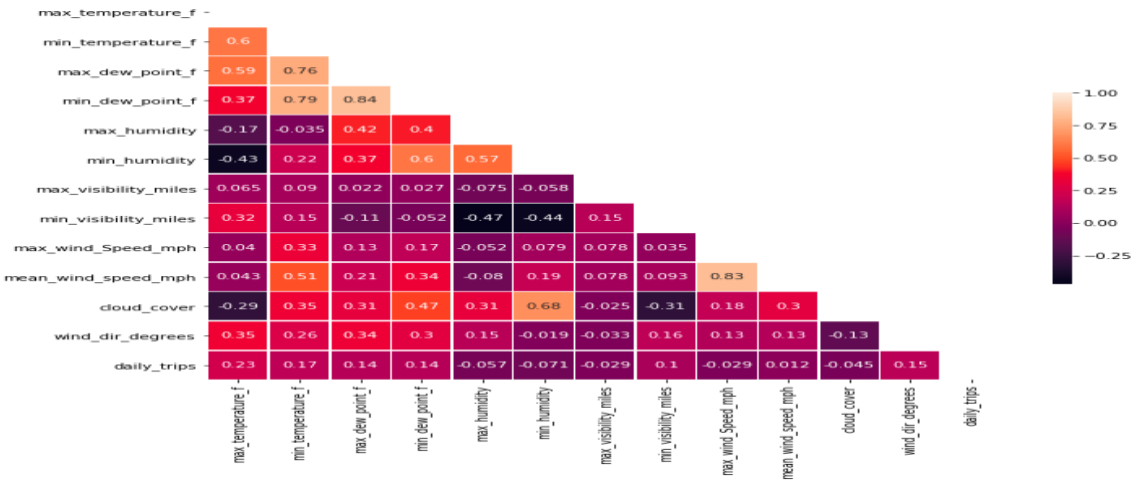


Figure 7 – Pearson Correlation between the weather variable and daily trips

It is visible from the above the Temperature have the highest statistical impact on Trips,

Moreover, from the tableau sheet, Exploring SF Trips [Figure 2] it was visible that the Feb and Dec are the least favoured day for bike and also have minimal temperature [Figure 3] of whole year, similarly Oct is most favoured and have maximum temperature of whole year.

Hence it can be said as the operator can be relaxed in minimum temperature months and highly active in maximum temperature months.

Prediction Model and Evaluation

For the dynamic reallocation, a prediction model can be used to predict number of start trip, end trip, bikes available and Dock available. Hence which can be used for reallocation by

```
Bikes_Required = Start_Trip - Bike Available - End Trip
```

and

```
if Bikes_Required > Docks_Available
          then Allocate_Bikes = Docks_Available - End_Trip
else
          Bike_Required
```

Note : `Bikes_Required = Start Trips on Station`

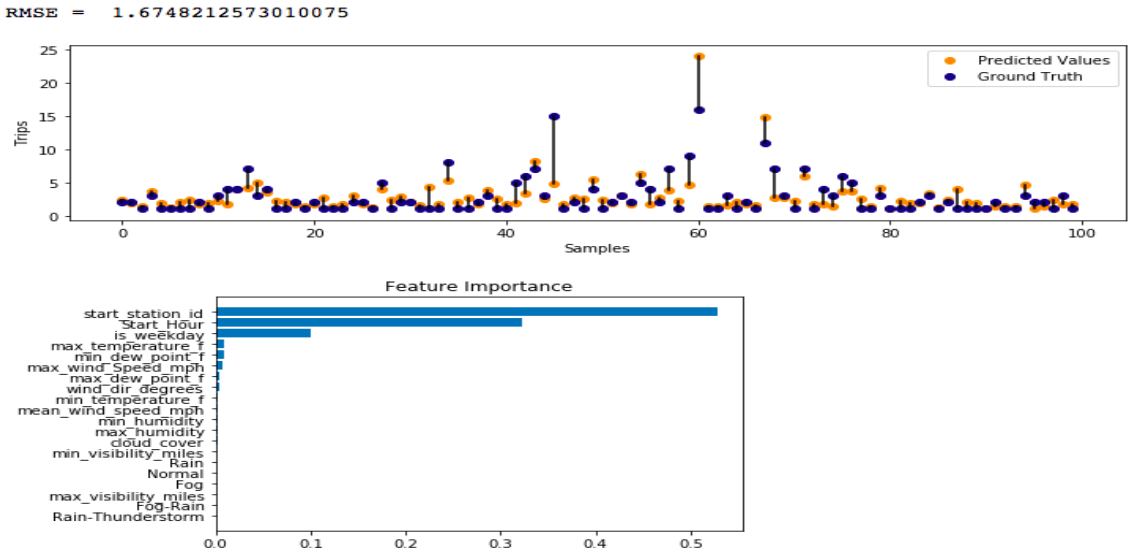


Figure 8 -Evaluation and Feature Importance of Random Forest model

Results looks good after model evaluation but there are still scope.

Inference

1. As per current model, StationId, week day, Start day, temperature highly important for prediction of trips.

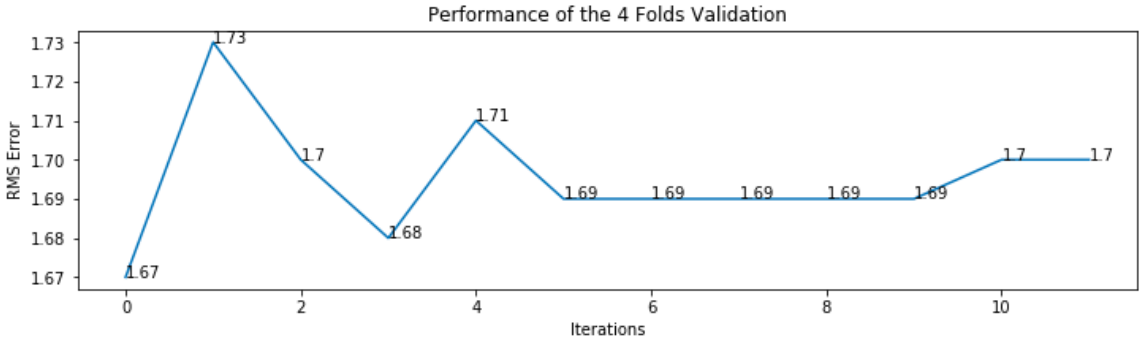


Figure 9 – Results of 4 X 3 Fold Validation of above model

2. Cross Validation gives results where SD is very low which indicate model is not overfitted.

Other models can be used for efficiency like XG Boost or ANN, and can be optimized using different approaches.

Model for other metrices can be created using same approach and finally a way optimized way can be approached to the Operator on real time.

Scope and Limitation

This study is solely focused on data analysis and providing solution other model can be used as discussed earlier with tuning and optimization, statistical study still have hope on feature selection, and accuracy of the model can be improved by different techniques like PCA etc. which is not covered in this work.

Ride-distance and duration is not covered which can be very useful for feature development.

Furthermore, An effective way of reallocation is provided in **A Solution For Reallocating Public Bike Among Bike Stations by Jinfeng Li from IBM labs,** this can be used for daily operation. Moreover, status file is being skipped but have good use for prediction for other metrices like dock available and bike available, which is not covered in this work due to time constraint.

Tools and Language

We are able to find solutions for operator based on the daily patterns, the choice of programming language is python because it consists of library providing the state-of-the-art techniques to find the solutions for data science problems. Tableau is used for data understanding due to its effective way of providing visual solution for initial exploration. Plotly is used for EDA of trip as combining it python widgets proving dynamic solution within the notebook, dynamic visualization can be generated using matplotlib and seaborne providing best visualization with least code.