**ADBMS: -Bhavesh Patel 001680899**

**Final Project Report**



**Authors**

Bhavesh Patel

Date: 12 December 2016

Versions 2.0

Table of Contents

1. **Dataset**
2. **Problem Statement/Objective**
3. **Approach To Problem**
4. **Introduction**
5. **Findings and Research**
6. **Implementation**
7. **Issues and their Resolution**
8. **Conclusion/Results**
9. **Future Scope**

**Dataset**

Considering big-data, I wanted to do some analysis on a dataset which would be practically impossible to do using native BI tools like Tableau, Power BI etc.

For this I downloaded a NYC taxi data set which is in tens of GB’s to do some analysis on the Taxi riding patters in New York city. The data set has around 14 million rides per month.

The data set has 2 different sections with taxi trip data and corresponding taxi fare data for that particular trip. The trips and fares are csv files which are broken down into monthly data.

**http://www.nyc.gov/html/tlc/html/about/trip\_record\_data.shtml**

Some of the columns related to this dataset include as follows: -

**Tripdata.csv**

Medallion – Sort of unique taxi data

Hack license – License number

Vendor id – vendor of the taxi

Store and forward flag – Y or N

Pickup date time – Date of pickup

Dropoff date time – Date of dropoff

Passenger count – Count of passenger

Trip time in sec – Trip duration in sec

Trip Distance - Trip Distance in miles

Pickup Latitude – Pickup Latitude

Pickup Longitude - Pickup Longitude

Dropoff Latitude - Dropoff Latitude

Dropoff Longitude - Dropoff Longitude

**Faredata.csv**

Medallion – Sort of unique taxi data

Hack license – License number

Vendor id – vendor of the taxi

Pickup date time – Date of pickup

Payment type – Mode of payment

Fare amount – fare amount in dollars

Surcharge – Surcharge during trip

Mta tax – Tax

Tip amount – Total tip paid

Toll amount – Amount paid at tolls

Total Amount – Total amount including all the above.

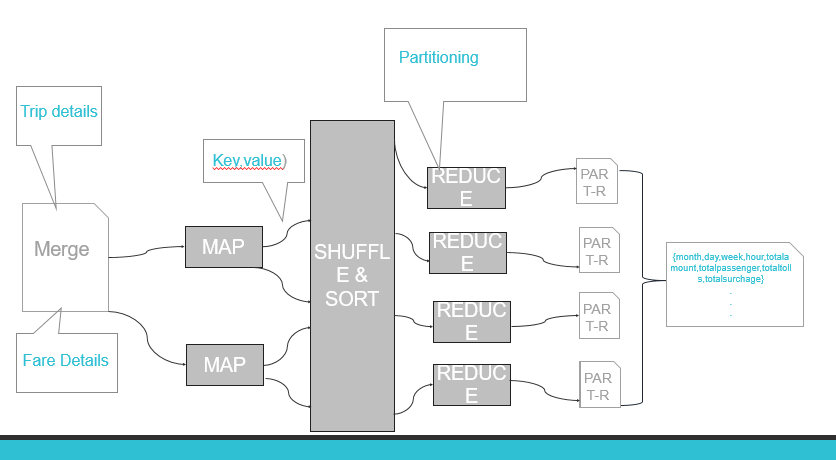
**Problem Statement/Objectives**

The main objective is to build an application that is used to analyze the NYC Yellow Taxi Dataset that can be predict prices and surcharges based on the various factors. Application should use map reduce algorithm to handle and analyze data that can be used to predict prices and surcharges on various categories and attributes. Application is capable of solving the following problems

* Which areas have most pickups and drop-offs of the 6 boroughs of NYC
* On which routes do passengers mostly tip their drivers
* Which areas high density of taxis
* Which Riders are most efficient in driving
* What revenue is generated on timely basis like month, day, week , hour etc.
* What are the ride patters of taxi drivers
* Surcharge Analysis
* Which routes are busiest, popular and where do passengers travel the most
* Which routes have maximum tolls
* Which routes have maximum surcharges

**Approach To Problem**

There were 2 different files one with ride history and the other with fare details. The first goal was to merge the two data based on the medallion and the pickup and date-time as a primary key. After merging the files, map reduce was done using java as well as pig along with partitioning and secondary sorting to get the output files from reducer. Further visualizations were done on these files using Power BI.



**Introduction to Application**

The important functionalities of the application include:

* Predict the price based on location, date, time and no. of passengers.
* Predict surcharge based on pickup and drop-offs
* The other Apache’s tool PIG is used to clean and analyze the data.
* Power BI is used to visualize the data and show various visualizations based on different results.

**AIgorithms**

1. Top 10 Algorithm

2. Secondary Sort Algorithm

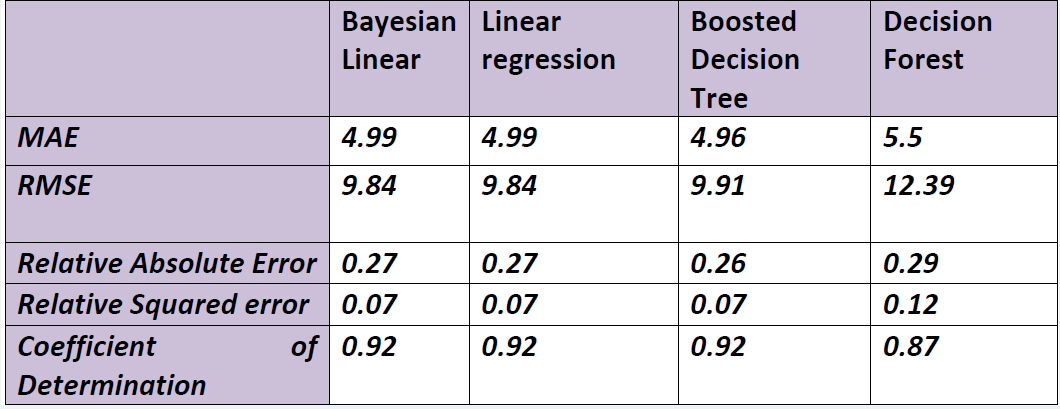
3. Order Inversion

4. Join Patterns

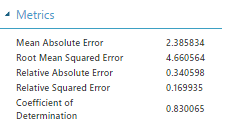
5. Filtering Patterns

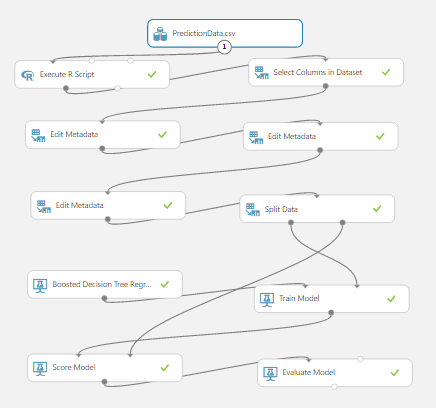
6. Boosted Decision tree regression Algorithm for price and surcharge prediction

* Bayesian
* **Boosted decision Tree**
* Neural Networks
* Linear Regression

****

**1.Boosted Decision Tree**





**Data Analysis Using PIG**

Pig Scripts: -

* Individual Analysis

timedata = LOAD '/project/dataset/TimeDataCombined' USING PigStorage(',') as (id:chararray,month:int,date:int,weekday:int,weekinyear:int,hour:int,surcharge:float,totalamount:float,passenger:int);

userdata = FILTER timedata BY (id=='00B7691D86D96AEBD21DD9E138F90840');

customgrouped = GROUP userdata BY (id,month,date) PARALLEL 1;

customaggregate = FOREACH customgrouped GENERATE FLATTEN(group) AS (id,month,date),

COUNT(userdata.id) AS totalrides,

org.apache.pig.builtin.ROUND\_TO((SUM(userdata.totalamount)),2) AS totalamount,

SUM(userdata.passenger) AS totalpassenger;

store customaggregate into '/project/dataset/IndividualAnalysisCombinedPig' using PigStorage(',','-schema');

* **Region Analysis**

data = LOAD '/project/dataset/faretripjoincombined' USING PigStorage(',') as (id:chararray,pickupdate:datetime,surcharge:float,tip:float,toll:float,totalamount:float,dropoffdate:datetime,

passenger:int,distance:float,pickuplon:float,pickuplat:float,dropofflon:float,dropofflat:float);

newdata = foreach data Generate pickuplon, pickuplat, dropofflon,dropofflat,distance,passenger,totalamount,surcharge,tip,toll;

customgrouped = GROUP newdata BY (pickuplon, pickuplat, dropofflon,dropofflat) PARALLEL 1;

customaggregate = FOREACH customgrouped

{

noofpassengerspayingtip = FILTER newdata BY (tip > 0);

GENERATE FLATTEN(group) AS (pickuplon, pickuplat, dropofflon,dropofflat),

COUNT(newdata) AS totalrides,

org.apache.pig.builtin.ROUND\_TO((AVG(newdata.distance)),2) AS averagedistance,

SUM(newdata.passenger) AS totalpassenger,

org.apache.pig.builtin.ROUND\_TO((AVG(newdata.totalamount)),2) AS averageamount,

org.apache.pig.builtin.ROUND\_TO((AVG(newdata.surcharge)),2) AS averagesurcharge,

COUNT(noofpassengerspayingtip) AS noofpassengerspayingtip,

org.apache.pig.builtin.ROUND\_TO((AVG(newdata.toll)),2) AS averagetoll;

};

top20basedontotalrides = ORDER customaggregate BY totalrides DESC;

top20basedontotalrides = LIMIT top20basedontotalrides 20;

top20longestdistance = ORDER customaggregate BY averagedistance DESC;

top20longestdistance = LIMIT top20longestdistance 20;

top20basedontotalpassengers = ORDER customaggregate BY totalpassenger DESC;

top20basedontotalpassengers = LIMIT top20basedontotalpassengers 20;

top20basedonamount = ORDER customaggregate BY averageamount DESC;

top20basedonamount = LIMIT top20basedonamount 20;

top20highestsurcharge = ORDER customaggregate BY averagesurcharge DESC;

top20highestsurcharge = LIMIT top20highestsurcharge 20;

top20tiparea = ORDER customaggregate BY noofpassengerspayingtip DESC;

top20tiparea = LIMIT top20tiparea 20;

top20highesttollroute = ORDER customaggregate BY averagetoll DESC;

top20highesttollroute = LIMIT top20highesttollroute 20;

store top20basedontotalrides into '/project/dataset/RegionAnalysis/top20basedontotalrides' using PigStorage(',','-schema');

store top20longestdistance into '/project/dataset/RegionAnalysis/top20longestdistance' using PigStorage(',','-schema');

store top20basedontotalpassengers into '/project/dataset/RegionAnalysis/top20basedontotalpassengers' using PigStorage(',','-schema');

store top20basedonamount into '/project/dataset/RegionAnalysis/top20basedonamount' using PigStorage(',','-schema');

store top20highestsurcharge into '/project/dataset/RegionAnalysis/top20highestsurcharge' using PigStorage(',','-schema');

store top20tiparea into '/project/dataset/RegionAnalysis/top20tiparea' using PigStorage(',','-schema');

store top20highesttollroute into '/project/dataset/RegionAnalysis/top20highesttollroute' using PigStorage(',','-schema');

* **Top1000Dropoff**

latlondata = LOAD '/project/dataset/DropoffCombined' USING PigStorage(',') as (latitude:float,longitude:float,total:int);

latlondata = ORDER latlondata BY total DESC;

Top1000Dropoff = LIMIT latlondata 1000;

store Top1000Dropoff into '/project/dataset/Top1000DropoffCombined' using PigStorage(',','-schema');

* **Top1000Pickup**

latlondata = LOAD '/project/dataset/PickupCombined' USING PigStorage(',') as (latitude:float,longitude:float,total:int);

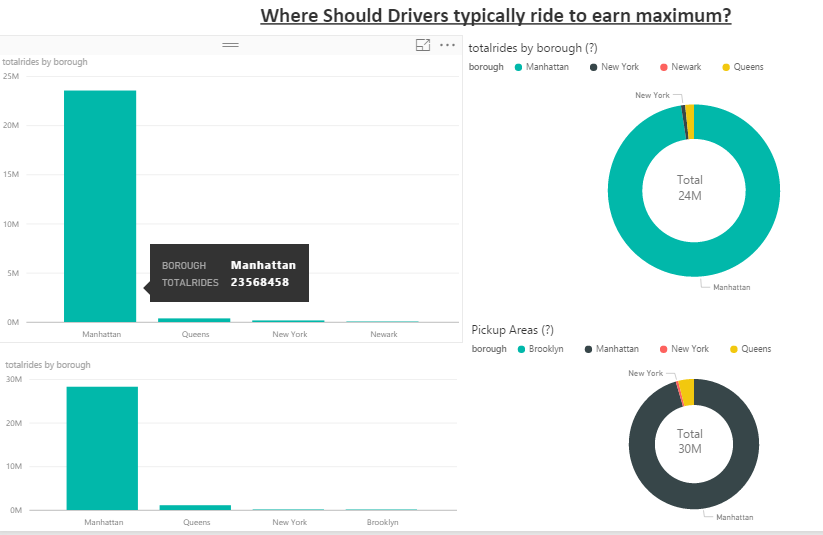
latlondata = ORDER latlondata BY total DESC;

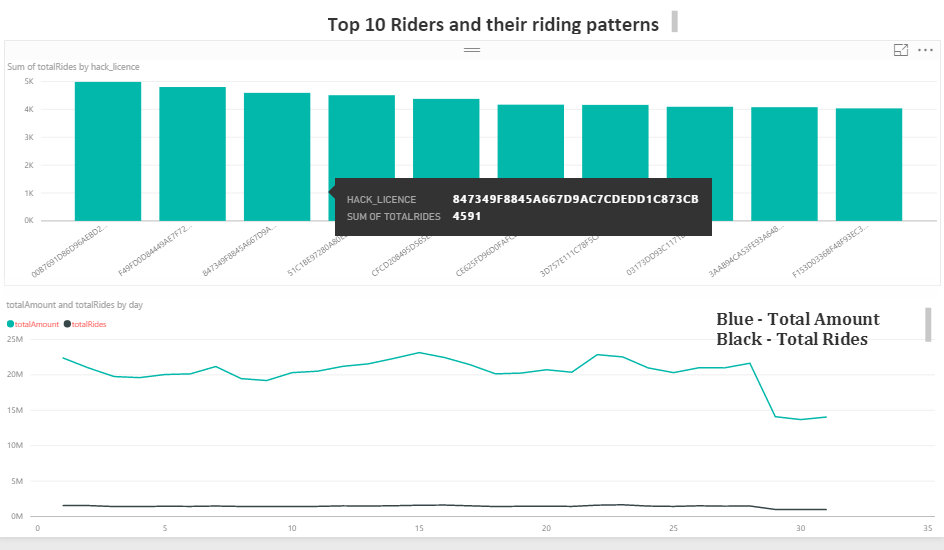
Top1000Pickup = LIMIT latlondata 1000;

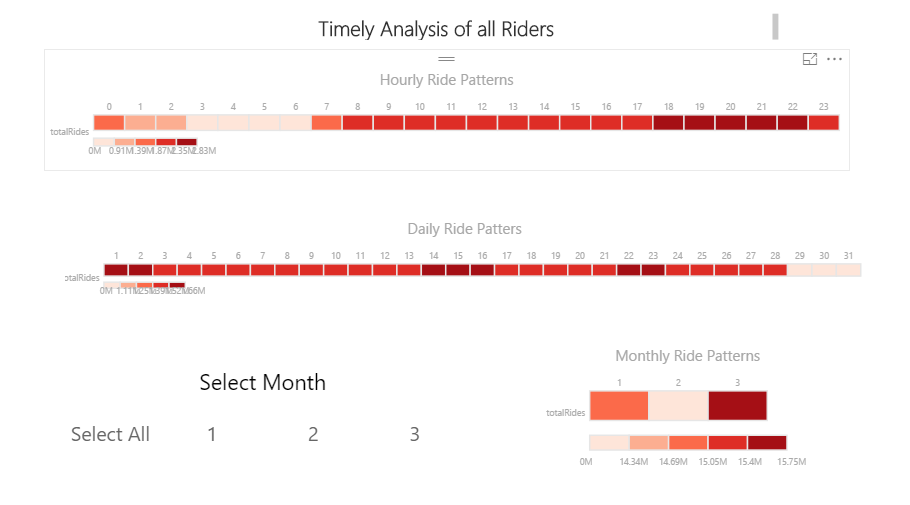
store Top1000Pickup into '/project/dataset/Top1000PickupCombined' using PigStorage(',','-schema');

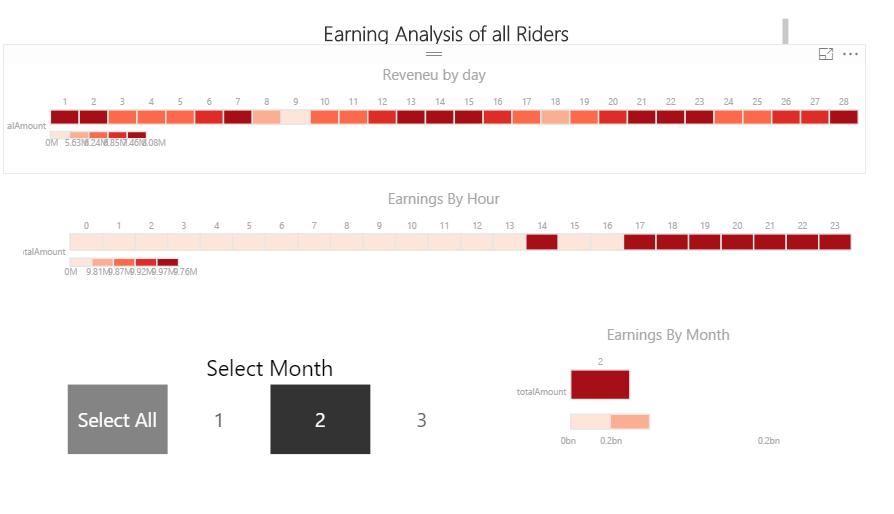
**Visualizations Using Power BI**

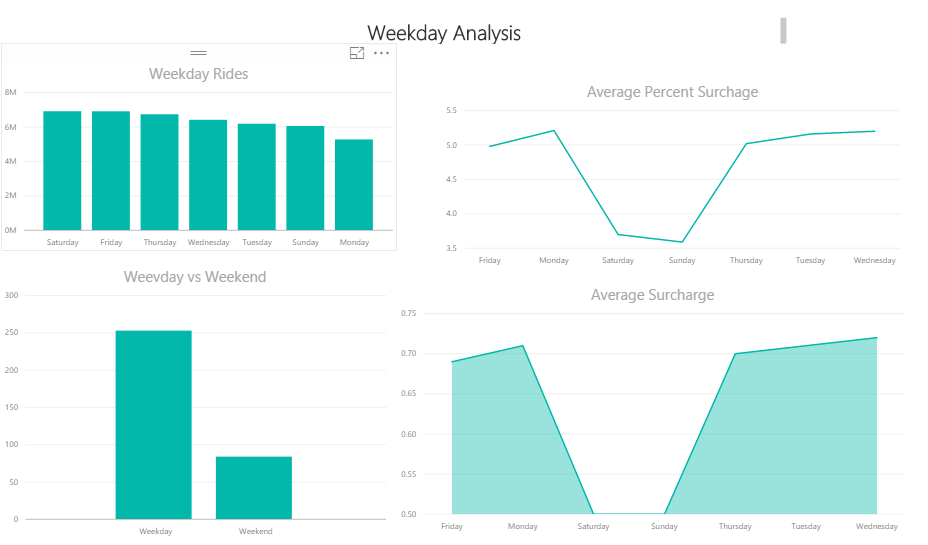
We have used map reduce jobs output’s that combines the outputs of the NYC dataset for analyzing the ride patters and region analysis

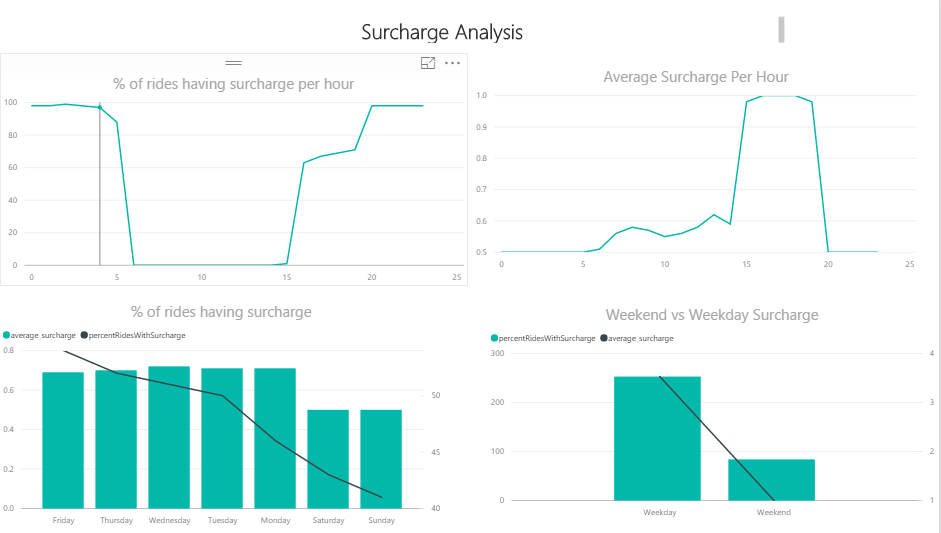


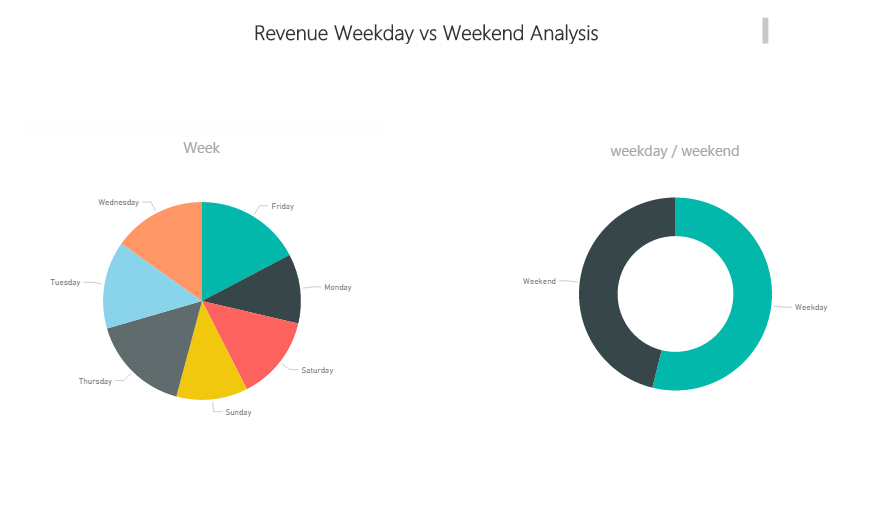


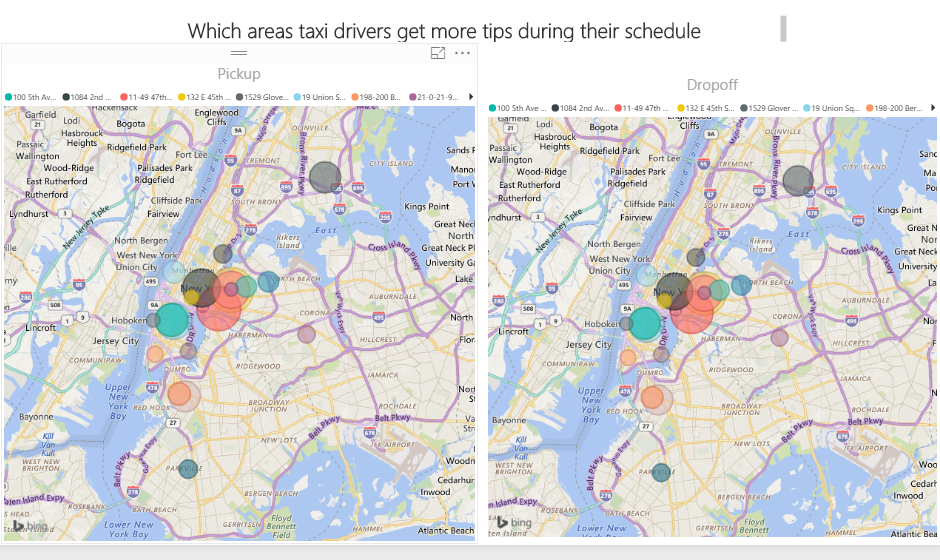


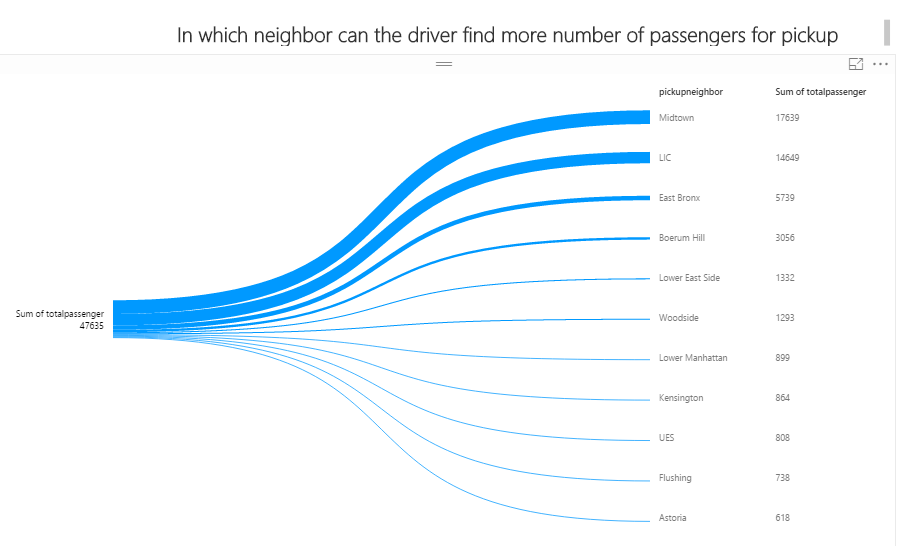


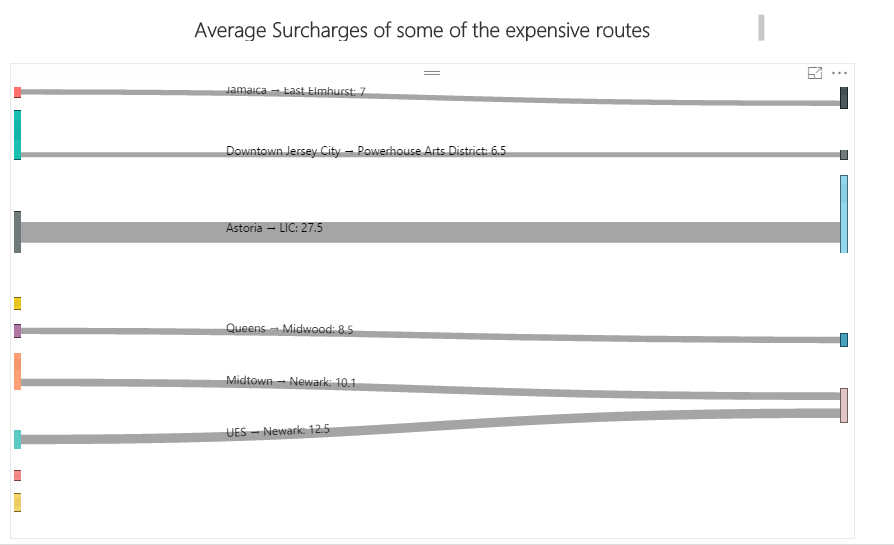


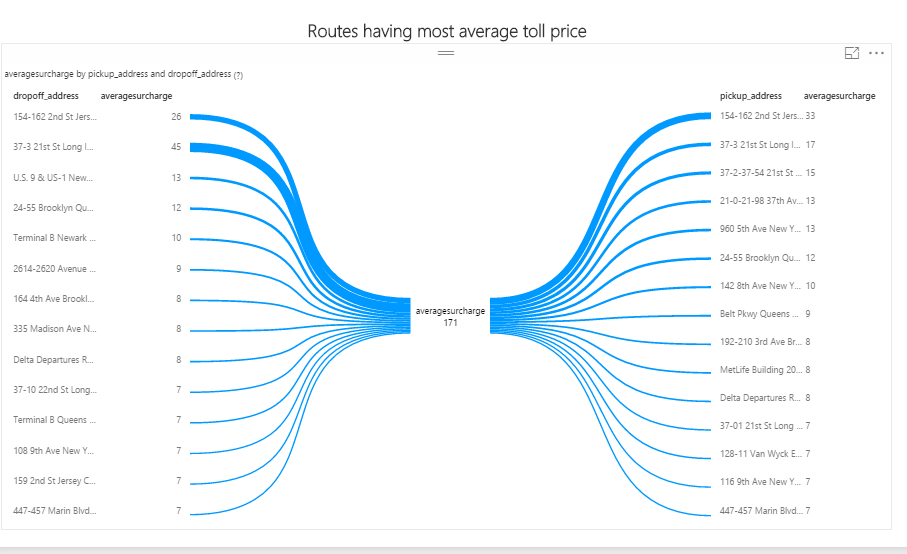




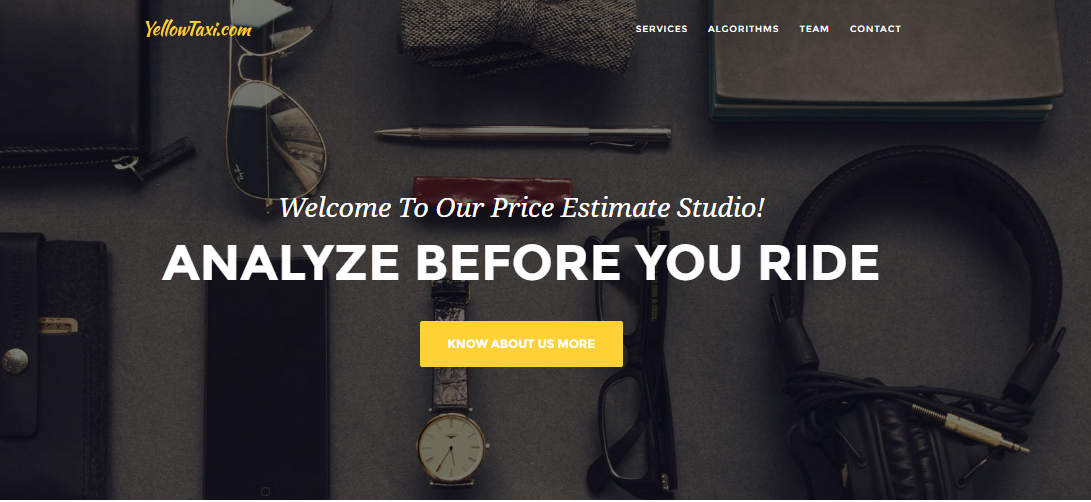


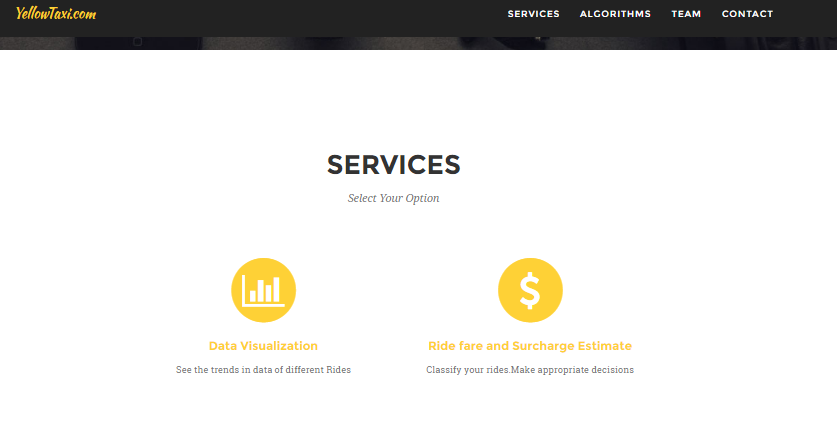


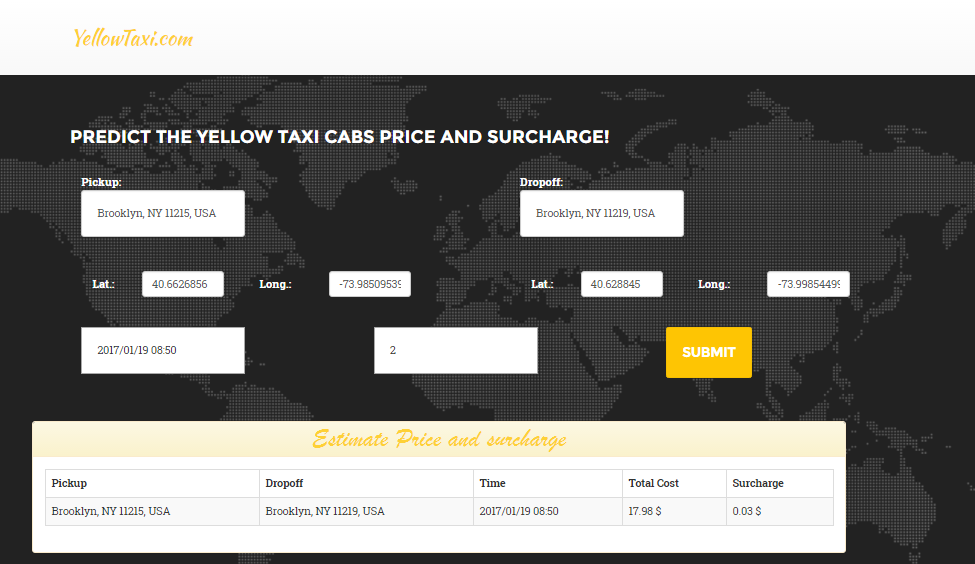




**Application Screenshots**







**Issues and their Resolution**

* Faced issues with the google maps API as it provides location details for 2500 hits per day only and I had around millions of data.
* Processing a huge dataset of around 20 GB took a long time to process for which aws was used for fully distributed mode.

**Conclusion/Results**

All the analysis decided was implemented under this project using map reduce, pig and HBase. For virtualization Power BI was used and the predictions were done in Microsoft Azure and was hosted as a web service to deploy an application on AWS.

All the problem statements were successfully implemented using various technologies and algorithms. A successful prediction model was done using machine learning and end to end analysis were done using pig and map reduce to solve all the business problems.

**Future Scope**

* Future scope includes showing the visualizations more on the google maps ad creating live interactions between the user and the application.
* Showing real time traffic and delay predictions of the journey time in the Taxi.

**References**

* [**https://github.com/marcogx/taxi-analysis**](https://github.com/marcogx/taxi-analysis)
* [**http://www.nyc.gov/html/tlc/html/about/trip\_record\_data.shtml**](http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml)
* [**http://www.andresmh.com/nyctaxitrips/**](http://www.andresmh.com/nyctaxitrips/)
* **https://www.ocf.berkeley.edu/~dlevitt/2015/12/13/final-project-nyc-taxi-and-uber-data/**