***A PROJECT ON***

# “Business Insights on Post-COVID Taxi System”

SUBMITTED IN

PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE COURSE OF

DIPLOMA IN BIG DATA ANALYTICS



*SUNBEAM INSTITUTE OF INFORMATION TECHNOLOGY*

“Sunbeam IT Park”, Ground Floor, Phase 2 of Rajiv Gandhi Infotech Park,

Hinjewadi, Pune-411057,

MH-INDIA

**SUBMITTED BY:**

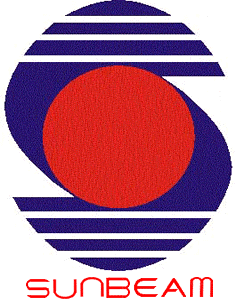
Onkar Eknath Shelar (56809)

**UNDER THE GUIDENCE OF:**

Mrs. Suvrunda Nangare

Faculty Member

Sunbeam Institute of Information Technology, PUNE.



**CERTIFICATE**

This is to certify that the project work under the title ‘Business Insights on Post-COVID Taxi System’ is done by Onkar Eknath Shelar in partial fulfillment of the requirement for award of Diploma in Big Data Analytics Course.

## Mrs. Suvrunda Nangare Mrs. Pradnya Dindorkar

**Project Guide Course Coordinator**

Date:

# ACKNOWLEDGEMENT

A project usually falls short of its expectation unless aided and guided by the right persons at the right time. We avail this opportunity to express our deep sense of gratitude towards Mr. Nitin Kudale (Center Coordinator, SIIT, Pune) and Mrs. Pradnya Dindorkar (Course Coordinator, SIIT ,Pune) and Project Guide Mrs. Suvrunda Nangare.

We are deeply indebted and grateful to them for their guidance, encouragement and deep concern for our project. Without their critical evaluation and suggestions at every stage of the project, this project could never have reached its present form.

Last but not the least we thank the entire faculty and the staff members of Sunbeam Institute of Information Technology, Pune for their support.

Onkar Eknath Shelar

DBDA Sept 2021 Batch,

SIIT Pune

.

**TABLE OF CONTENTS**

## Definition

* 1. Project Overview
  2. Problem Statements
  3. Metrics

## Analysis

* 1. Dataset
  2. Data Exploration and Visualizations
  3. Algorithms and Techniques

## Methodology

* 1. Data Preprocessing
  2. Analysis

1. **Results**
   1. Model Evaluation and Validation
   2. Conclusion
   3. Further Improvement

1. **Flow Diagram**

**Definition**

## Project Overview

## Making a prediction based on unknown data can be a deceptively difficult problem. What would be a simple calculation in hindsight, when you have the benefit of all the variables, often proves too be much more complicated when several variables cannot be known in advance. This project is a fantastic example of this problem and attempts to solve it with a mountain of data and a start-of-the-art algorithm.

## Specifically, this project creates a submission to the Diploma Project Assessment, using selection from the massive New York City (NYC) Taxi and Limousine Commission (TLC) Yellow Cab dataset that is that could be known when booking the ride, such as pickup and drop-off location but not unknowns like future traffic conditions or the route the driver is going to take. I am approaching this Project by putting myself in the shoes of an engineer with NYC TLC, and am attempting to create a production-ready model deployed.

## Problem Statement

## The goal is to create an almost production-ready model that predicts a taxi ride’s fare based only on the information any rider would be able to provide to the driver at the time of booking. Succes will entail:

## Building a model that could be deployed to a production environment with minimal extra work.

## Utilize the enormous millions rows of data sparring taxi rides from 2019 to 2021, with deep exploration of the data and cleaning/fixing erroneous entries as needed.

## Develop a process to transform the data just on historical records without accessing 3rd party services, and without using data that rider wouldn’t know when requesting the ride.

## Metrics

## We have used 4 metrics for evaluation purpose, which are r squared score, mean absolute error, mean squared error and root mean squared error.

## We evaluate the model mainly by judging the values of r squared score and root mean squared error.

## Analysis

## Dataset:

## For this topic there were multiple types of dataset available online, most of them were clean and hardly few were raw dataset.

## I found the suitable data set on website NYC-Taxi and Limousine Commission (TLC). The URL is - https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page

## We have downloaded the dataset from January 2020 to July 2021. The dataset contains 18 variables. Following is the information of the dataset:

## 1. 'VendorID' – code indicating the TPER ( Taxicab and Livery Passenger er Enhancement Programs (TPEP ) provider that provided the record

## 2. 'tpep\_pickup\_datetime' – pick up date and time

## 3. 'tpep\_dropoff\_datetime' – drop off date and time

## 4. 'passenger\_count' – passenger count

## 5. 'trip\_distance' – elapsed trip distance in miles reported by the taximeter

## 6. 'RatecodeID' – final rate code effect at the end of the trip

## 1 – Standard Rate

## 2 – JFK

## 3 – Newark

## 4 – Nassau

## 5 – Negotiated fare

## 6 – group ride

## 7. 'store\_and\_fwd\_flag' – flag indicated whether the trip record was held in vehicle

## Y – store and forward trip

## N – not a store and forward trip

## 8. 'PULocationID' – TLC Taxi zone in which taximeter was engaged

## 9. 'DOLocationID' – TLC Taxi zone in whoch taximeter was disengaged

## 10. 'payment\_type' – numeric code signifying the payment mode

## 1 – credit card

## 2 – cash

## 3 – no charge

## 4 – dispute

## 5 – unknown

## 6 – voided trip

## 11. 'fare\_amount' – time and distance fare calculated by the meter

## 12. 'extra' – miscellaneous extras and surcharges.

## 1. Rush charge – 0.5 USD

## 2. Overnight charge – 1 USD

## 13. 'mta\_tax' – 0.50 MTA tax, automatically added

## 14. 'tip\_amount' – tip amount. field is automatically populated for credit card tips., cash tips are not included.

## 15. 'tolls\_amount' – total amount of all tolls paid in trip

## 16. 'improvement\_surcharge' – 0.30 imporvement surcharge assessed trips at the flag drop.

## 17. 'total\_amount' – total amount charged to passengers. Does not include cash tips.

## 18. 'congestion\_surcharge' – additional charge added to base charge during unusal events like strikes, bad winter, major port fires.

## Data Exploration and Visualization

## A quick summary of the dataset’s statistics reveals some questionable numbers. We also found out the descriptive statistics here which includes mean, median and mode. Following it I checked the null and nan and unique values of all the features.

## 

## From the statistics, we came to know few outliers. And thus, the data cleansing process started. Like negative distance, negative fare amount and so on. Such outliers were removed as they were logically wrong.

## Also, every instance which were outside the coordinates of NYC, were considered as outliers and removed from the dataset. In this way, all the outliers were removed and the data garbage was cleaned creating the dataset ready for feature engineering.

## In feature engineering, the required changes in features were carried out and thus altering the entire dataset. This is how we got the transformed dataset, which is ready for model building and model training and testing.

## To get the quick insights of the data we created few visualizations like distance vs time of the day to see the trip traveled and the impact on it based on the time slot of the day.

## 

## We also collected the insights of number of trips based on month, hour of the day, weekday, passenger count, and so on

## 

## We tried to find the behavior of fare against distance, against different zones of NYC.

## 

## Algorithms and Techniques

## Firstly, dataset was split into train and test here. With 80% od data in training data and 20% remaining data into testing data.

## As there were numerous columns, and important feature might be left out while considering the dependencies, we used the technique called as Principal Component Analysis. Which gave us feature affecting our required result the most.

## After getting the proper overview of the dependencies, and selecting the features for machine learning model, and based on the statistics of data, I came to result to use regression models. The three different machine learning algorithm used are :

## Linear regression - is one of the very basic forms of machine learning where we train a model to predict the behavior of your data based on some variables. In the case of linear regression as you can see the name suggests linear that means the two variables which are on the x-axis and y-axis should be linearly correlated.

## XGBoost - or extreme gradient boosting is one of the well-known [gradient](https://analyticsindiamag.com/gradient-descent-everything-you-need-to-know-with-implementation-in-python/) [boosting](https://analyticsindiamag.com/gradient-descent-everything-you-need-to-know-with-implementation-in-python/) techniques (ensemble) having enhanced performance and speed in tree- based (sequential decision trees) machine learning algorithms. XGBoost was created by Tianqi Chen and initially maintained by the Distributed (Deep) Machine Learning Community (DMLC) group. It is the most common algorithm used for applied machine learning in competitions and has gained popularity through winning solutions in structured and tabular data. It is open- source software. Earlier only [python and R packages](https://analyticsindiamag.com/python-vs-scala-for-apache-spark/) were built for XGBoost but now it has extended to Java, Scala, Julia and other languages as well.

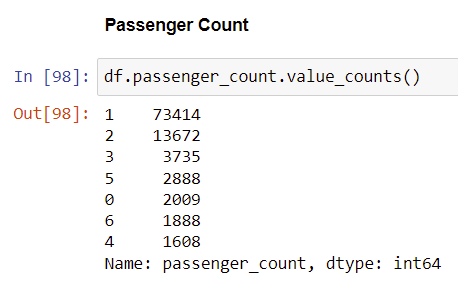
## Random Forest - is a Supervised Machine Learning Algorithm that is used widely in Classification and Regression problems. It builds decision trees on different samples and takes their majority vote for classification and average in case of regression.One of the most important features of the Random Forest Algorithm is that it can handle the data set containing continuous variables as in the case of regression and categorical variables as in the case of classification. It performs better results for classification problems

## Based on the evaluation metrics of all these algorithm, I decided to go for Random forest as the final algorithm. Which is capable of predicting multiple output by using multiple decision trees.

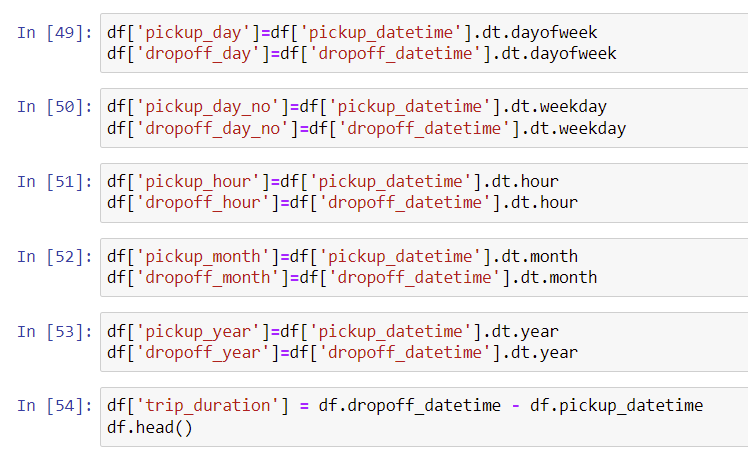
## METHODLOGY

## 3.1 Data Preprocessing

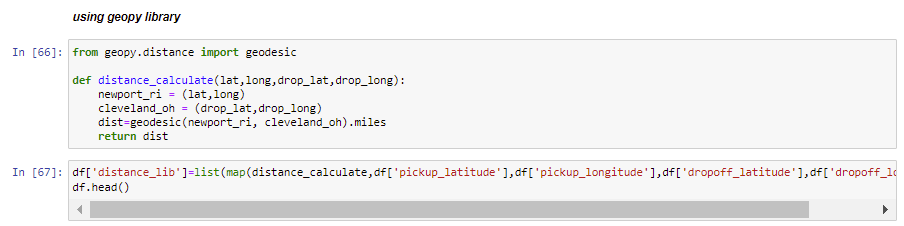
The first and foremost thing I did was to change the column names and made them readable. I checked the locations outside the NYC and removed them. Also, studied the passenger count data.

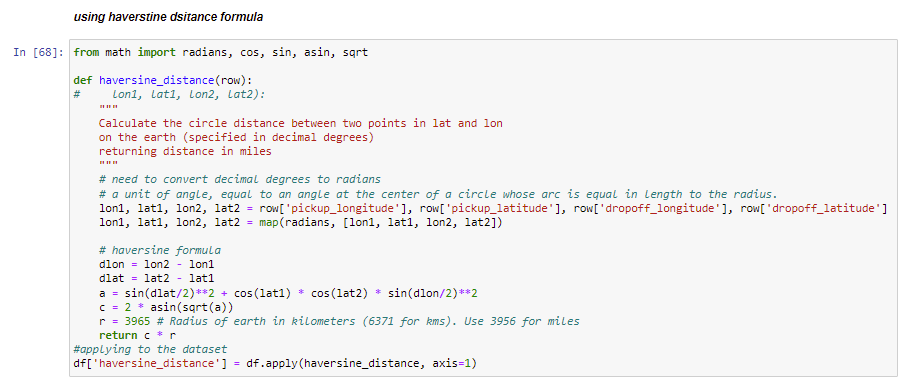


For time series analysis, I divided the time related feature into multiple sub columns, just to have an easy access to it’s different feature. This step opened up a whole lot of windows for me and to study more about the data.

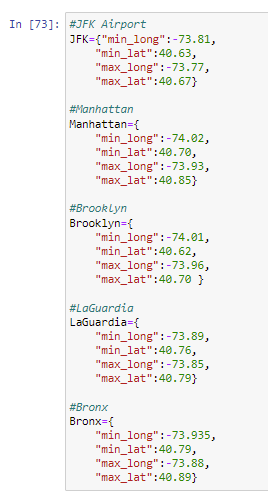


Also, to study more in-depth, I added the features like mid night trips, rush hour trip, airport trip. The major step in this whole preprocessing was to get the distance between two locations. It was done in two ways, one to use mathematical theorem giving us harvestine distance and other way to get distance using geopy library. In order to reduce the errors, I took the mean of these distances and used it further in the project.





To get the insights geographical location wise, I mapped the four zones in NYC and tried to get insights from each part of NYC individually, giving us more business insights.



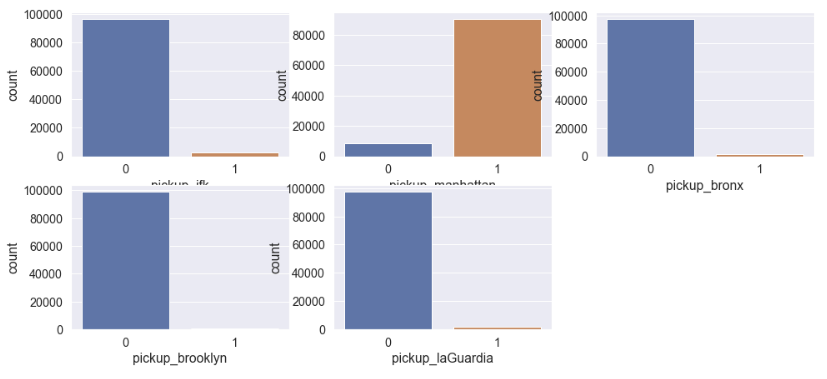
**3.2 Analysis**

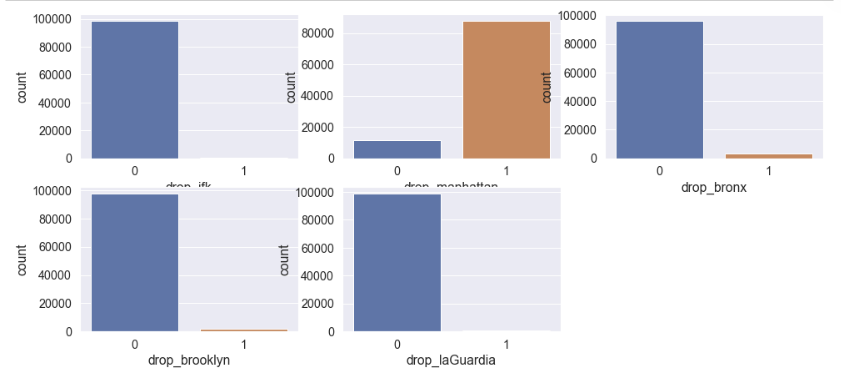
Once the data was process we had cleaned data in our hands and to make this data into useful information I attempted to do analysis. I analysed data into two steps:

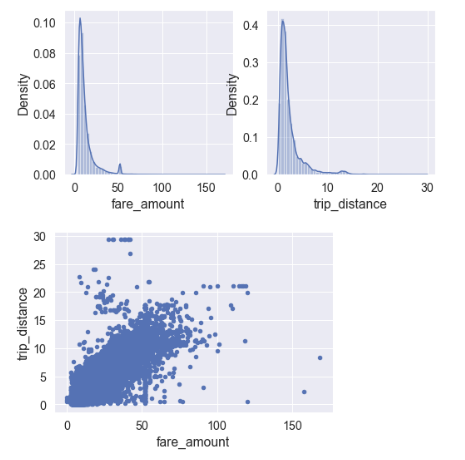
* + - Univariate analysis
    - Bivariate Analysis

Univariate analysis gave me all the insights from the individual features which deepen my understanding about the data and made it easy to understand there nature.

Bivariate analysis helped me to deepen my understanding about the dependency of fare amount on other features.







**RESULTS**

**4.1 Model Evaluation and Validation**

Based on the evaluation metrics, the model was selected. I have also implemented the the customized tuning of hypermeter which helped me in achieving enhanced accurate results.

Model was validated by attempting to predict the Total amount, fare amount, and tip amount from the training dataset and comparing the predicted value to the actual value, which showed the accuracy of our prediction, and thus applying it on testing dataset.

**4.2 Conclusion:**

Random Forest have the highest accuracy among all the algorithms used. So we prefer the random forest as the final algorithm for the prediction.

**4.3 Further Improvement:**

Although all the features which were required had already been included in prediction, but one can also add the functionality in machine Learning model, where it suggest the type of ride based on the economical fare. For this one can even implement recommendation system and integrating it with these regression models and creating a pipeline for it.

**5.0 Flow Diagram**

