UBER RELATED DATA ANALYSIS USING MACHINE LEARNING

Submitted in partial fulfilment of the requirements for the award of
Bachelor of Engineering Degree inComputer Science and Engineering

By

RISHI.S

(Reg.No. 37110642)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI-6000119, TAMIL NADU

APRIL 2021

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)
Accredited with ―A‖ grade by NAAC
Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai - 600119
www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Professional Training Report is the bonafide work of RISHI.S (37110642) who underwent the professional training in-Uber Related Data Analysis using Machine Learning‖ underour supervision from November 2020 to April2021.

Internal Guide

DR.B.ANKAYARKANNI,M.E, Ph.D

Head of the Department

Submitted for Viva Voce Examination held on ____________

InternalExaminer **ExternalExaminer**

DECLARATION

I, RISHI.S (37110642), hereby declare that the Professional Training Reporton -**UBER RELATED DATA ANALYSIS USING MACHINE LEARNING",**done by me under the guidance of **DR.B.ANKAYARKANNI ,M.E,Ph.D** atSathyabama Institute of Science and Technology, is submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in Computer Science.

DATE:

PLACE:CHENNAI **SIGNATURE OF THE CANDIDATE**

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.Sasikala M.E.,Ph.D.**, **Dean**, School of Computing **Dr.S.Vigneshwari M.E., Ph.D.** and **Dr.L.Lakshmanan M.E., Ph.D. ,** Heads of the Department of Computer Science and Engineering for providing us necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and a deep sense of gratitude to my Project Guide **Dr. B. Ankayarkanni M.E., Ph.D.,** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Departmentof Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

The project ―Uber Related Data Analysis using Machine Learning‖ explains the working of an Uber dataset, which contains data produced by Uber for New York City. Uber is defined as a P2P platform. The platform links you to drivers who can take you to your destination. The dataset includes primary data on Uber pickups with details including the date, time of the ride as well as longitude-latitude information, Using the information, the paper explains the use of the k-means clustering algorithm on the set of data and classify the various parts of New York City. Since the industry is booming and expected to grow shortly. Effective taxi dispatching will facilitate each driver and passenger to reduce the wait time to seek out one another. The model is employed to predict the demand on points of the city.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

ABBREVIATIONS	EXPANSION
ML	Machine Learning
ANN	Artificial Neural Network
GA	Genetic Algorithms
DOM	Date of Month

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

INTRODUCTION TO UBER DATA AND PREDICTIONS

Uber Technologies, Inc., commonly called Uber, is an American technology company. Its services include ride-hailing, food delivery (Uber Eats), package delivery, couriers, freight transportation, and, through a partnership with Lime, electric bicycle and motorized scooter rental. Uber Technologies may be a P2P network for sharing travel The Uber platform connects you with drivers who can take you to your destination or location. This dataset includes primary data on Uber collections with details that include the date, time of travel, yet as information on longitude and latitude in urban center and has operations in over 900 metropolitan areas worldwide. The prediction of the frequency of trips of knowledge is by implementing an element of k-means clustering algorithm,the quality algorithm describes the most variance within the group because the number of square distances Euclidean distances between the points and also the corresponding centroid.

Machine Learning Predictions

With the appearance of the computing device, uber data prediction has since moved into the technological realm. the foremost prominent technique involves the employment of artificial neural networks (ANNs) and Genetic Algorithms(GA). Scholars found bacterial chemotaxis optimization method may perform better than GA. ANNs is thought of as function approximators. the foremost common sort of ANN in use for stock exchange prediction is that the feed forward network utilizing the backward propagation of errors algorithm to update the network weights. These networks are commonly cited as Backpropagation networks. Another type of ANN that's more appropriate for uber data prediction is that the time recurrent neural network (RNN) or time delay neural network (TDNN). samples of RNN and TDNN For uber data prediction with ANNs, there are usually two approaches taken for forecasting different time horizons: independent and joint. The independent approach employs one ANN for every time horizon, for instance, 1-day, 2-day, or 5-day. The advantage of this approach is that network forecasting error for one horizon won't impact the error for one more horizon—since every time horizon is usually a novel problem. The joint approach, however, incorporates multiple time horizons together so they're determined simultaneously. during this approach, forecasting error for just once horizon may share its error therewith of another horizon, which might decrease performance. There also are more parameters required for a joint model, which increases the chance of overfitting.

INTRODUCTION TO MACHINE LEARNING

Machine learning (ML) is that the study of computer algorithms that improve automatically through experience. it's seen as a subset of computing. Machine learning algorithms build a mathematical model to support sample data, called "training data", so on form predictions or decisions without being explicitly programmed to undertake to do so. Machine learning algorithms are utilized in a very large sort of applications, like email filtering and computer vision, where it's difficult or infeasible to develop conventional algorithms to perform the needed tasks.

Overview

Machine learning involves computers discovering how they'll perform tasks without being explicitly programmed to try to to so. It involves computers learning from data provided so they perform certain tasks. for easy tasks assigned to computers, it's possible to program algorithms telling the machine the way to execute all steps required to unravel the matter at hand; on the computer's part, no learning is required. For more advanced tasks, it will be challenging for an individual's to manually create the needed algorithms. In practice, it can end up to be simpler to assist the machine develop its own algorithm, instead of having human programmers specify every needed step.

The discipline of machine learning employs various approaches to show computers to accomplish tasks where no fully satisfactory algorithm is on the market. In cases where vast numbers of potential answers exist, one approach is to label a number of the proper answers as valid. this may then be used as training data for the pc to boost the algorithm(s) it uses to work out correct answers. as an example, to coach a system for the task of digital character recognition, the MNIST dataset of handwritten digits has often been used

Machine Learning Approaches

Machine learning approaches are traditionally divided into three broad categories, counting on the character of the "signal" or "feedback" available to the training system:

• Supervised learning: the pc is presented with example inputs and their desired outputs, given by a "teacher", and so the goal is to be told a general rule that maps inputs to outputs.

• Unsupervised learning: No labels are given to the training algorithm, leaving it on its own to go looking out structure in its input. Unsupervised learning could be a goal in itself (discovering hidden patterns in data) or how towards an end (feature learning).

• Reinforcement learning: A worm interacts with a dynamic environment during which it must perform a specific goal (such as driving a vehicle or playing a game against an opponent). because it navigates its problem space, the program is provided feedback that's analogous to rewards, which it tries to maximise.

Other approaches are developed which don't fit neatly into this three-fold categorisation, and sometimes over one is used by the identical machine learning system. as an example topic modelling, dimensionality reduction or meta learning.

As of 2020, deep learning has become the dominant approach for ongoing arenas of machine learning.

Theory

A core objective of a learner is to generalize from its experience. Generalization during this context is that the ability of a learning machine to perform accurately on new, unseen examples/tasks after having experienced a learning data set. The training examples come from some generally unknown probability distribution (considered representative of the space of occurrences) and therefore the learner has got to build a general model about this space that permits it to supply sufficiently accurate predictions in new cases.

The computational analysis of machine learning algorithms and their performance could be a branch of theoretical engineering science referred to as computational learning theory. Because training sets are finite and therefore the future is uncertain, learning theory usually doesn't yield guarantees of the performance of algorithms. Instead, probabilistic bounds on the performance are quite common. The bias–variance decomposition is a method to quantify generalization error.

For the simplest performance within the context of generalization, the complexity of the hypothesis should match the complexity of the function underlying the information. If the hypothesis is a smaller amount complex than the function, then the model has under fitted the information. If the complexity of the model is increased in response, then the training error decreases. But if the hypothesis is simply too complex, then the model is subject to overfitting and generalization are going to be poorer.

In addition to performance bounds, learning theorists study the time complexity and feasibility of learning. In computational learning theory, a computation is taken into account feasible if it is tired polynomial time. There are two types of time complexity results. Positive results show that a specific class of functions is learned in polynomial time. Negative results show that certain classes can't be learned in polynomial time.

Models

ARTIFICIAL NEURAL NETWORKS

Artificial neural networks (ANNs), or connectionist systems, are computing systems vaguely inspired by the biological neural networks that constitute animal brains. Such systems "learn" to perform tasks by considering examples, generally without being programmed with any task-specific rules.

An ANN may be a model supported a group of connected units or nodes called "artificial neurons", which loosely model the neurons in a very biological brain. Each connection, just like the synapses during a biological brain, can transmit information, a "signal", from one artificial neuron to a different. a man-made neuron that receives a sign can process it and so signal additional artificial neurons connected to that. In common ANN implementations, the signal at a connection between artificial neurons could be a imaginary number, and therefore the output of every artificial neuron is computed by some non-linear function of the sum of its inputs. The connections between artificial neurons are called "edges". Artificial neurons and edges typically have a weight that adjusts as learning proceeds. the burden increases or decreases the strength of the signal at a connection. Artificial neurons may have a threshold specified the signal is barely sent if the mixture signal crosses that threshold. Typically, artificial neurons are aggregated into layers. Different layers may perform different sorts of transformations on their inputs. Signals travel from the primary layer (the input layer) to the last layer (the output layer), possibly after traversing the layers multiple times.

The original goal of the ANN approach was to resolve problems within the same way that somebody's brain would. However, over time, attention moved to performing specific tasks, resulting in deviations from biology. Artificial neural networks are used on a spread of tasks, including computer vision, speech recognition, MT, social network filtering, playing board and video games and diagnosing.

Deep learning consists of multiple hidden layers in a synthetic neural network. This approach tries to model the way the human brain processes light and sound into vision and hearing. Some successful applications of deep learning are computer vision and speech recognition.

REGRESSION ANALYSIS

Regression analysis encompasses an outsized style of statistical methods to estimate the link between input variables and their associated features. Its commonest form is statistical regression, where one line is drawn to best fit the given data in step with a mathematical criterion like ordinary statistical procedure. The latter is usually extended by regularization (mathematics) methods to mitigate overfitting and bias, as in ridge regression. When managing non-linear problems, go-to models include polynomial regression (for example, used for trendline fitting in Microsoft Excel), Logistic regression (often employed in statistical classification) or maybe kernel regression, which introduces non-linearity by taking advantage of the kernel trick to implicitly map input variables to higher dimensional space.

K-MEANS CLUSTERING

k-means clustering could be a method of vector quantization, originally from signal processing, that aims to partition n observations into k clusters within which each observation belongs to the cluster with the closest mean (cluster centres or cluster centroid), serving as a prototype of the cluster. This leads to a partitioning of the info space into Voronoi cells. k-means clustering minimizes within-cluster variances (squared Euclidean distances), but not regular Euclidean distances, which might be the harder Weber problem: the mean optimizes squared errors, whereas only the geometric median minimizes Euclidean distances. for example, better Euclidean solutions may be found using k-medians and k-medoids.

The problem is computationally difficult (NP-hard); however, efficient heuristic algorithms converge quickly to an area optimum. These are usually kind of like the expectation-maximization algorithm for mixtures of Gaussian distributions via an iterative refinement approach employed by both k-means and Gaussian mixture modelling. They both use cluster centers to model the data; however, k-means clustering tends to search out clusters of comparable spatial extent, while the expectation-maximization mechanism allows clusters to own different shapes.

The unsupervised k-means algorithm includes a loose relationship to the k-nearest neighbour classifier, a preferred supervised machine learning technique for classification that's often confused with k-means thanks to the name. Applying the 1-nearest neighbor classifier to the cluster centers obtained by k-means classifies new data into the present clusters. this can be referred to as nearest centroid classifier or Rocchio algorithm.

CHAPTER 2

LITERATURE REVIEW

2.1 RELATED WORKS

A state in which the results, k-means clustering is used to estimate the most likely collection points at a given time and to predict the best hotspots of nightlife learning trends from previous Uber pickups. This has been verified using the Lyft test set and is consistent with Yelp's best results.[1]

Poulsen, L.K In this document, they conducted a spatial analysis of the Green Cab and Uber races in the outer districts of New York City to determine the competitive position of the NYCTLC. We found that the demand for green taxis continues to increase, but the number of Uber trips in the same area is growing faster. However, the analysis showed that in Greece, in an area that is generally poor in general. They did not find any variation between the green taxi and Uber when variations were observed between weekdays and weekends. This research recommends that NYCTLC create a dashboard that analyzes and displays data in real time, as we believe this will increase its competitiveness compared to Uber. Uber is a recent taxi operator in New York and is constantly devouring the market share of the yellow and green taxis of the New York Taxi and Limousine Commission (NYCTLC)[2]

Faghih, S.S recommends a recent modeling approach in Manhattan, New York City, to capture the demand for electronic mail services, particularly the Uber application. Uber collection data is added to the Manhattan TAD level and at 15-minute time intervals. This aggregation allows the implementation of a modern approach to spatio-temporal modeling to obtain a spatial and temporal understanding of the demand. During a typical day, two spacetime models were developed using Uber collection data, the STAR and STAR and MSPE turns determine the output of the models. The results of the MSPE have shown that it is recommended to use the Lasso-Star system instead of the star design. A comparison between the demand for yellow and uber taxis in 2014 and 2015 in New York shows that the demand for uber has increased.[3]

Ghuha declared the grouping of the sequence observed using a small amount of memory and time. The data flow model is necessary for new kinds of applications that involve large data sets, such as web click flow analysis and multimedia data flow data model analysis where: data A sequence of points, the objective is Maintain a constant level. In the document, we will consider models in which the groups have a different point or "center". In the k-median question, the objective is to minimize the average distance from the data points to the nearest cluster centers.[4]

Ahmed, M., has shown that by using detailed data on taxis at the travel level and on the rental vehicle and data on complaints about the level of new complaints at the level of incidents, we study how Uber and Lyft enter damaged the quality of taxi services in New York City. The overall effect of the organizations based on the scenario and in particular of the riding administrations was enormous and widespread. One of these effects is the expansion of the rivalry between Uber and Lyft over the quality of taxi administration. They use a new set of complaint data to measure (the lack of) quality of service that we have never been analyzed before. Focus on the quality dimensions generated by most of the complaints we demonstrate. The increased competition for these shared travel services has had an intuitive impact on the behavior of taxi drivers[5]

Wallsten, S, stated that the results of New York and Chicago are consistent with the possibility that taxis react to the new challenge by improving quality. In New York, the rise of Uber is linked to the reduction of objections to travel to the city. They discuss the competitive effect of sharing taxis in the taxi industry using the complete data set of the New York City Taxi and Limousine Commission for more than one billion taxi trips in complaints and details of New York, New York and Chicago Google Trends on the success of Uber's largest shared travel service.[6]

Sotiropoulos, D.N, represented that this document addresses the problem of grouping, by using a new approach to genetic algorithms that is highly scalable in large volumes of textual details, developing a coding scheme based on centroids. We apply k means clustering algorithm in this document. Clustering is the unsupervised machine learning algorithm used to solve grouping problems based on similarities. This technique has aroused interest in a wide range of scientific fields, which address clustering methods, to solve complex classification problems.[7]

CHAPTER 3

AIM AND SCOPE OF PROJECT

AIM OF PROJECT

The aim of the project is to predict the pickup of the cab with respect to the location given by the user in the app from clusters which are main coordinated points in a city by applying data visualization on the basis of frequency of trips travelled with respect to the day in the month and applying data analysis using the algorithm of k-means clustering.

OBJECTIVES

The objective of the project is to plot a heat map for the prediction of frequency of trips across a collection of points(latitude,longitude)in a dataset which is data visualization and using these objectives apply the data in the algorithm adopted in the project.

SCOPE OF PROJECT

Based on the aim and the objectives in the project, the scope of the project is to apply an algorithm best suited to analyze the data imported from the dataset such that the algorithm can train the data and then proceed for data analysis.

.

CHAPTER 4

MODULE IMPLEMENTATION

MODULES

The modules used are based on the process of data acquisition,processing and analysis of a dataset.

The dataset includes primary data on Uber pick-ups with details including the date, time of the ride as well as longitude-latitude information of the city.

The dataset is imported and then analyzed and computized by analysis by clustering algorithms,(K-means clustering). The module description can be explained by importing the following modules from python.

%PYLAB INLINE

PyLab can be defined as a procedural interface to the Matplotlib object-oriented plotting libraryMatplotlib which is that the whole package; matplotlib.pyplot can be defined as a module in Matplotlib; and PyLab can be defined as a module that gets installed alongside Matplotlib.

PyLab can be defined as a convenience module that bulk imports matplotlib.pyplot (for plotting) and NumPy (for Mathematics and dealing with arrays) in a very single name space.

PANDAS

In computer programing, pandas can be defined as a software library written for the Python programing language for data manipulation and analysis. particularly, it offers data structures and operations for manipulating numerical tables and statistics and it's free software was released under the three-clause BSD license.

Pandas is principally used for data analysis. Pandas allows importing data from various file formats like comma-separated values, JSON, SQL, Microsoft Excel. Pandas allows various data manipulation operations like merging, reshaping, selecting, still as data cleaning, and data wrangling features.

SEABORN

Seaborn is a library for making statistical graphics in Python. It builds on top of matplotlib and integrates closely with pandas data structures.

Seaborn helps you explore and understand your data. Its plotting functions operate on dataframes and arrays containing whole datasets and internally perform the necessary semantic mapping and statistical aggregation to produce informative plots. Its dataset-oriented, declarative API lets you focus on what the different elements of your plots mean, rather than on the details of how to draw them.

It gives us the capability to create amplified data visuals. This helps us understand the data by displaying it in a visual context to unearth any hidden correlations between variables or trends that might not be obvious initially. Seaborn has a high-level interface as compared to the low level of Matplotlib.

Seaborn comes with a large number of high-level interfaces and customized themes that matplotlib lacks as it's not easy to figure out the settings that make plots attractive. Matplotlib functions don't work well with dataframes, whereas seaborn does.

Seaborn's integration with matplotlib allows you to use it across the many environments that matplotlib supports, inlcuding exploratory analysis in notebooks, real-time interaction in GUI applications, and archival output in a number of raster and vector formats.

While you can be productive using only seaborn functions, full customization of your graphics will require some knowledge of matplotlib's concepts and API. One aspect of the learning curve for new users of seaborn will be knowing when dropping down to the matplotlib layer is necessary to achieve a particular customization. On the other hand, users coming from matplotlib will find that much of their knowledge transfers.

KMEANS

The k means module is an unsupervised machine learning technique module accustomed to identify clusters of information objects in an exceeding dataset.

The module is employed effectively and goal of the algorithm is to seek out groups within the data, with the quantity of groups represented by the varaiable K. Data points are clustered which is supported by the feature similarity.

YELLOWBRICK

Yellowbrick can be defined as a suite of visual analysis and diagnostic tools designed to facilitate machine learning with scikit-learn. The library implements a replacement core API object, the Visualizer which is that's an scikit-learn estimator — an object that learns from data. almost like transformers or models, visualizers learn from data by creating a visible representation of the model selection workflow.

The Visualizer allow users to steer the model selection process, building intuitions around feature engineering, algorithm selections and hyperparameter tuning. for example, they'll help diagnose common problems surrounding model complexity and bias, heteroscedasticity, underfit and overtraining, or class balance issues. By applying visualizers to the model selection workflow, It allows you to steer predictive models toward more successful results, faster.

FOLIUM

Folium can be defined as a module and a tool for plotting maps, it's a strong library that helps you create several sorts of Leaflet maps.The Folium results are interactive as it makes the library very useful for dashboard building. By default,Folium creates a map in an exceedingly separate HTML file and plots the info obtained from the dataset

MODULE SOFTWARE/HARDWARE

Software Configuration:

- Anaconda Navigator
- Jupyter Notebook
- Google Chrome

Hardware Configuration:

- Windows 10,8,7
- Windows Server 2019 or Windows Server 2016
- Memory of atleast 4GB RAM
- Storage of Atleast 128 GB
- 1 GB Graphics Card

ARCHITECTURAL DESIGN

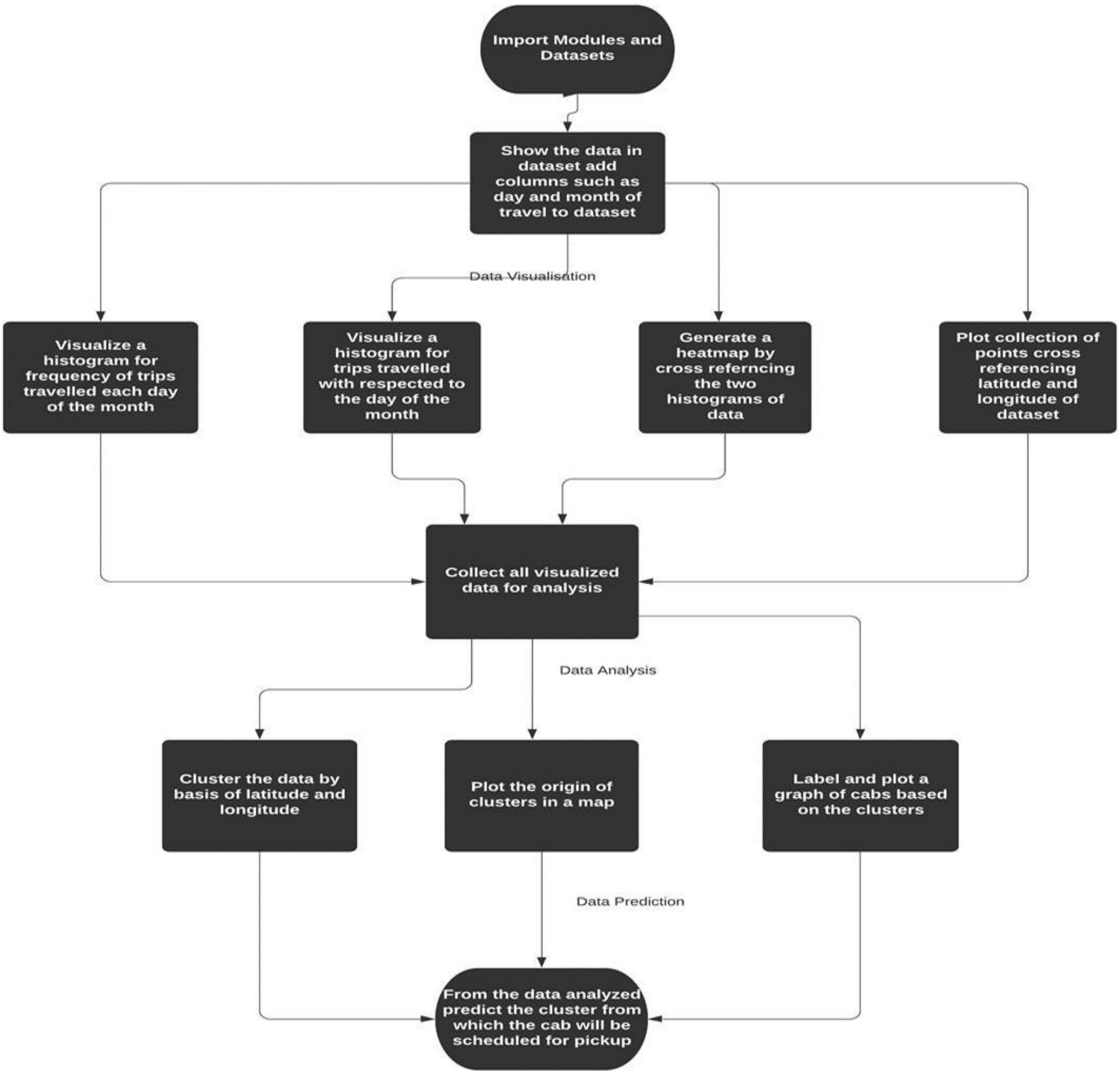

Import Modules and Datasets
Show the data in dataset add columns such as day and month of travel to dataset
Data Visualisation
Visualize a histogram for frequency of trips travelled each day of the month
Visualize a histogram for trips travelled with respected to the day of the month
Generate a heatmap by cross referncing the two histograms of data
Plot collection of points cross referencing latitude and longitude of dataset
Collect all visualized data for analysis
Data Analysis
Cluster the data by basis of latitude and longitude
Plot the origin of clusters in a map
Label and plot a graph of cabs based on the clusters
Data Prediction
From the data analyzed predict the cluster from which the cab will be scheduled for pickup

Fig 4.1: ARCHITECTURE DIAGRAM

MODULE IMPLEMENTATIONS

- The module is based on data acquisition, processing and analysis of a dataset.

- The dataset includes primary data on Uber pick-ups with details including the date, time of the ride as well as longitude-latitude information of the city .

- The dataset is imported and then analyzed and computized by analysis by clustering algorithms,(K-means clustering).

- Import the packages numpy and seaborn and import the dataset from Kaggle or github(uber.csv) which contains the dataset

- Import the data in jupyter notebook and using the data in dataset we calculate the frequency of trips in an given area in this dataset (New York City) from the latitudes and longitudes in the dataset

- Generate the heatmap and predict the pinpoint locations of travel of the cab

- Import the required packages required for analysis i.e sklearn,yellowbrick and folium

- Categorize the centroids on the basis of latitude and longitude

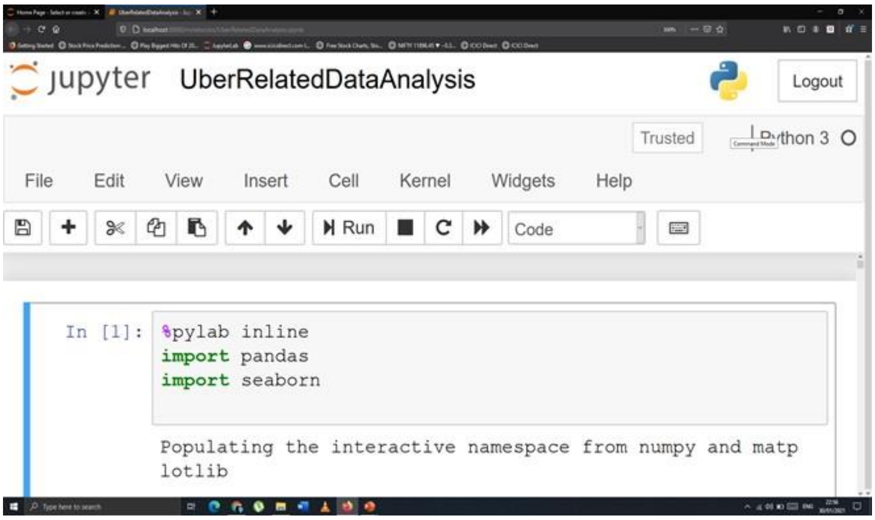
- Mark the centroid points on the map imported from the module folium

- Predict the cluster from where the cab is scheduled to pickup from the data.

CHAPTER 5

RESULTS AND DISCUSSIONS

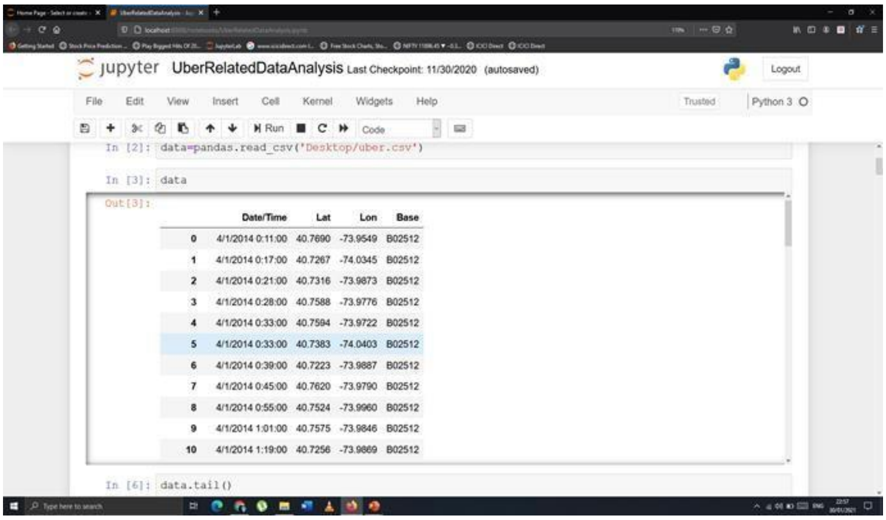
As a result of this system getting introduced in our data analysis, it will act like a mechanism to assign a cab to a customer to cover the gap of communication between the company and the customer. The company will be able to understand the customer better by deploying more cabs based on the hotspots plotted on the map using the algorithm.



```
In [1]: %pylab inline
        import pandas
        import seaborn

        Populating the interactive namespace from numpy and matp
        lotlib
```

FIG 5.1 IMPORTING MODULES

Figure 5.1 explains the importing of modules in the notebook after importing the modules we can import the dataset



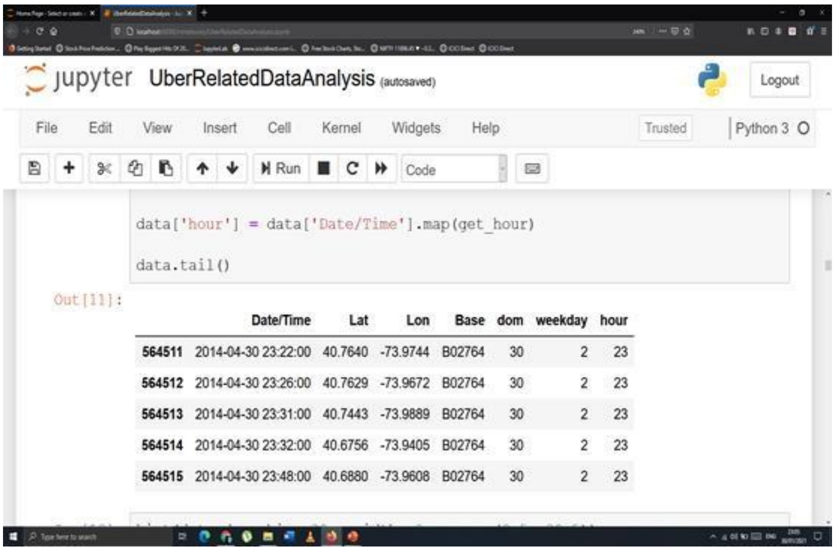
```
In [2]: data=pandas.read_csv('Desktop/uber.csv')

In [3]: data

Out[3]:
```

	Date/Time	Lat	Lon	Base
0	4/1/2014 0:11:00	40.7690	-73.9549	B02512
1	4/1/2014 0:17:00	40.7267	-74.0345	B02512
2	4/1/2014 0:21:00	40.7316	-73.9873	B02512
3	4/1/2014 0:28:00	40.7588	-73.9776	B02512
4	4/1/2014 0:33:00	40.7594	-73.9722	B02512
5	4/1/2014 0:33:00	40.7383	-74.0403	B02512
6	4/1/2014 0:39:00	40.7223	-73.9887	B02512
7	4/1/2014 0:45:00	40.7620	-73.9790	B02512
8	4/1/2014 0:55:00	40.7524	-73.9960	B02512
9	4/1/2014 1:01:00	40.7575	-73.9846	B02512
10	4/1/2014 1:19:00	40.7256	-73.9869	B02512

```
In [6]: data.tail()
```

FIG 5.2 IMPORTING DATASET

After importing the dataset,add more columns such as dom,weekday and hour



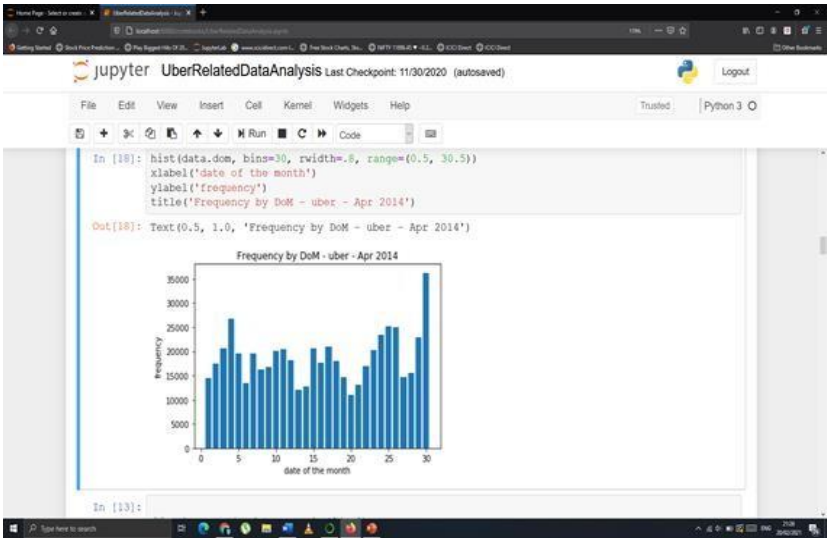
```
data['hour'] = data['Date/Time'].map(get_hour)

data.tail()
```

Out[11]:

	Date/Time	Lat	Lon	Base	dom	weekday	hour
564511	2014-04-30 23:22:00	40.7640	-73.9744	B02764	30	2	23
564512	2014-04-30 23:26:00	40.7629	-73.9672	B02764	30	2	23
564513	2014-04-30 23:31:00	40.7443	-73.9889	B02764	30	2	23
564514	2014-04-30 23:32:00	40.6756	-73.9405	B02764	30	2	23
564515	2014-04-30 23:48:00	40.6880	-73.9608	B02764	30	2	23

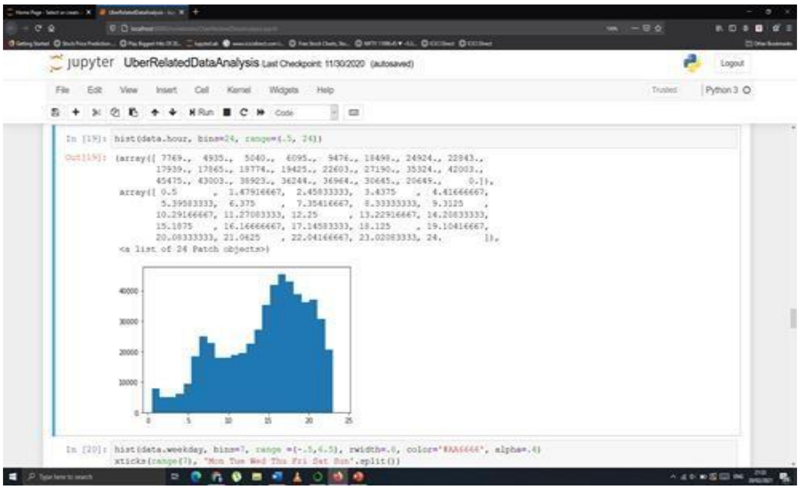
FIG 5.3 FINAL DATASET

After importing 5.3,let us visualize the data on the basis of frequency of cabs travelled in a day with respect to day,month.



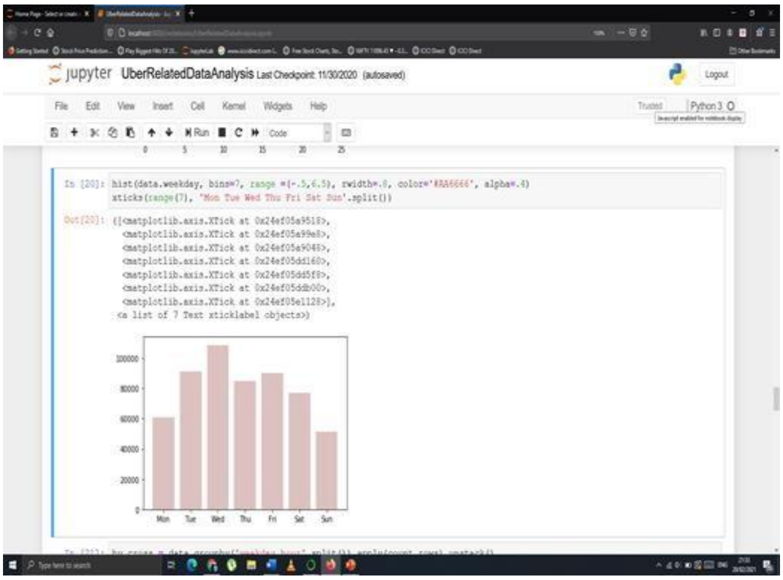
```
In [18]: hist(data.dom, bins=30, rwidth=.8, range=(0.5, 30.5))
         xlabel('date of the month')
         ylabel('frequency')
         title('Frequency by DoM - uber - Apr 2014')

Out[18]: Text(0.5, 1.0, 'Frequency by DoM - uber - Apr 2014')

In [13]:
```

FIG 5.4 VISUALISATION OF X-Y GRAPH OF FREQUENCY OF CABS



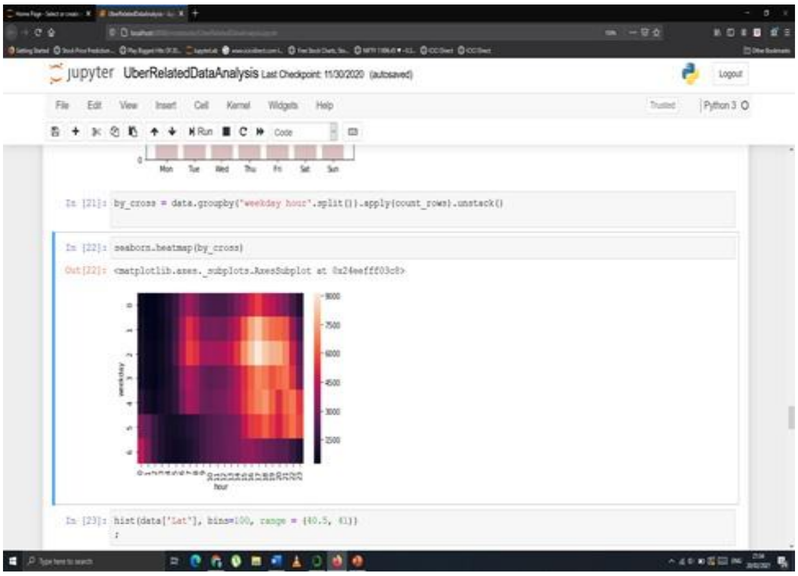
```
In [19]: hist(data.hour, bins=24, range=(.5, 24))

Out[19]: (array([ 7769.,  4935.,  5040.,  6095.,  9476., 18498., 24924., 22843.,
                 17939., 17865., 18774., 19425., 22603., 27190., 35324., 42003.,
                 45475., 43003., 38923., 36244., 36964., 30645., 20649.,     0.]),
          array([ 0.5       ,  1.47916667,  2.45833333,  3.4375    ,  4.41666667,
                  5.39583333,  6.375     ,  7.35416667,  8.33333333,  9.3125    ,
                 10.29166667, 11.27083333, 12.25      , 13.22916667, 14.20833333,
                 15.1875    , 16.16666667, 17.14583333, 18.125     , 19.10416667,
                 20.08333333, 21.0625    , 22.04166667, 23.02083333, 24.        ]),
          <a list of 24 Patch objects>)

In [20]: hist(data.weekday, bins=7, range =(-.5,6.5), rwidth=.8, color='#AA6666', alpha=.4)
         xticks(range(7), 'Mon Tue Wed Thu Fri Sat Sun'.split())
```

FIG 5.5 VISUALIZATION OF DATA BASED ON MONTH



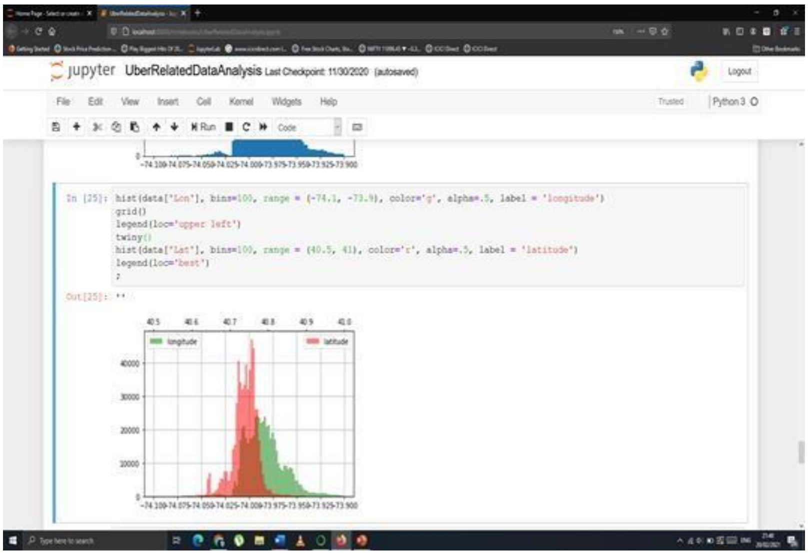
```
In [20]: hist(data.weekday, bins=7, range =(-.5,6.5), rwidth=.8, color='#AA6666', alpha=.4)
         xticks(range(7), 'Mon Tue Wed Thu Fri Sat Sun'.split())

Out[20]: ([<matplotlib.axis.XTick at 0x24ef05a9518>,
           <matplotlib.axis.XTick at 0x24ef05a99e8>,
           <matplotlib.axis.XTick at 0x24ef05a9048>,
           <matplotlib.axis.XTick at 0x24ef05dd160>,
           <matplotlib.axis.XTick at 0x24ef05dd5f8>,
           <matplotlib.axis.XTick at 0x24ef05ddb00>,
           <matplotlib.axis.XTick at 0x24ef05e1128>],
          <a list of 7 Text xticklabel objects>)
```

FIG 5.6 VISUALIZATION OF DATA BASED ON DAY



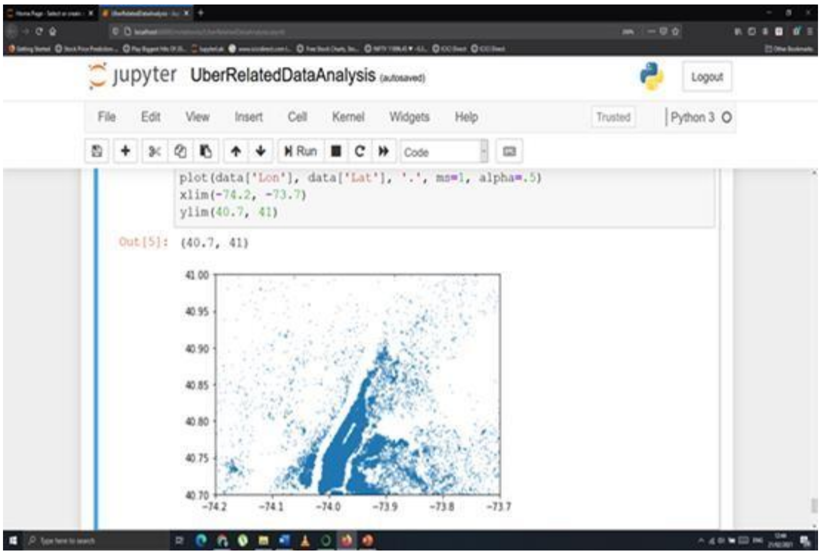
```
In [21]: by_cross = data.groupby('weekday hour'.split()).apply(count_rows).unstack()

In [22]: seaborn.heatmap(by_cross)

Out[22]: <matplotlib.axes._subplots.AxesSubplot at 0x24eefff03c8>

In [23]: hist(data['Lat'], bins=100, range = (40.5, 41))
         ;
```

FIG 5.7 PLOTTING HEAT MAP OF DATASET



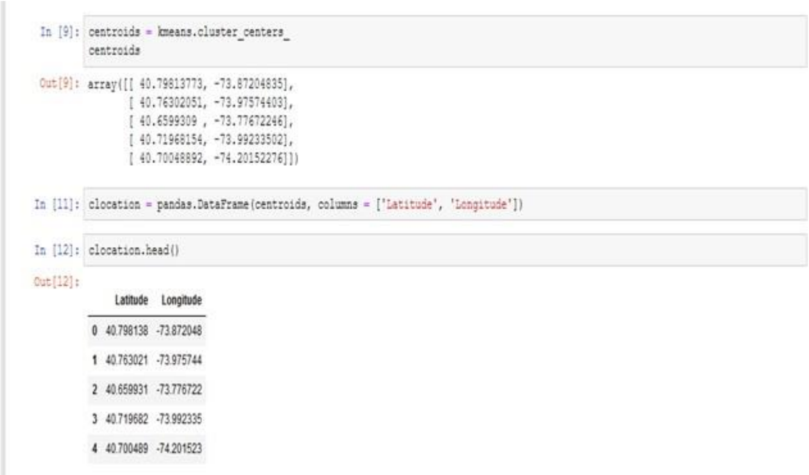
```
In [25]: hist(data['Lon'], bins=100, range = (-74.1, -73.9), color='g', alpha=.5, label = 'longitude')
         grid()
         legend(loc='upper left')
         twiny()
         hist(data['Lat'], bins=100, range = (40.5, 41), color='r', alpha=.5, label = 'latitude')
         legend(loc='best')
         ;

Out[25]: ''
```

FIG 5.8 CROSS REFERENCING LATITUDE AND LONGITUDE



```
plot(data['Lon'], data['Lat'], '.', ms=1, alpha=.5)
xlim(-74.2, -73.7)
ylim(40.7, 41)

Out[5]: (40.7, 41)
```

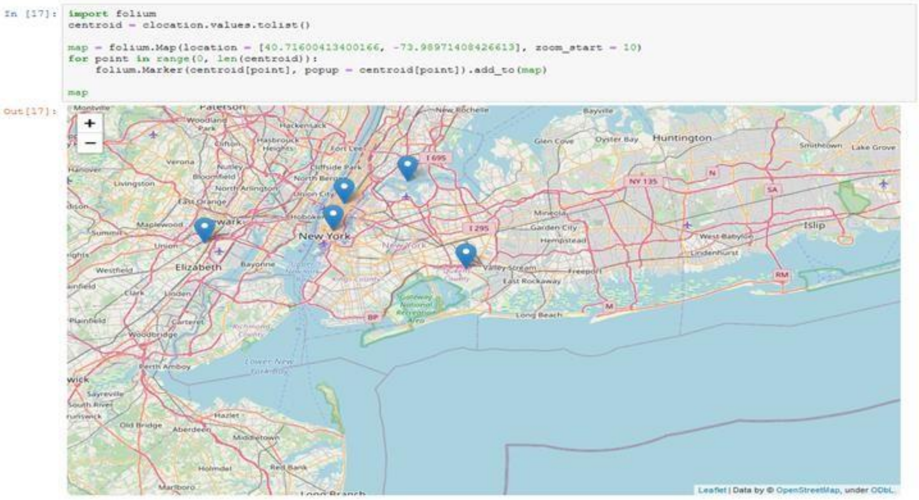
FIG 5.9 PLOTTING COLLECTION OF POINTS IN XY GRAPH



```
In [9]: centroids = kmeans.cluster_centers_
        centroids

Out[9]: array([[ 40.79813773, -73.87204835],
               [ 40.76302051, -73.97574403],
               [ 40.6599309 , -73.77672246],
               [ 40.71968154, -73.99233502],
               [ 40.70048892, -74.20152276]])

In [11]: clocation = pandas.DataFrame(centroids, columns = ['Latitude', 'Longitude'])

In [12]: clocation.head()

Out[12]:
```

	Latitude	Longitude
0	40.798138	-73.872048
1	40.763021	-73.975744
2	40.659931	-73.776722
3	40.719682	-73.992335
4	40.700489	-74.201523

FIG 5.10 GETTING THE DATA OF CENTROIDS FROM ALGORITHM

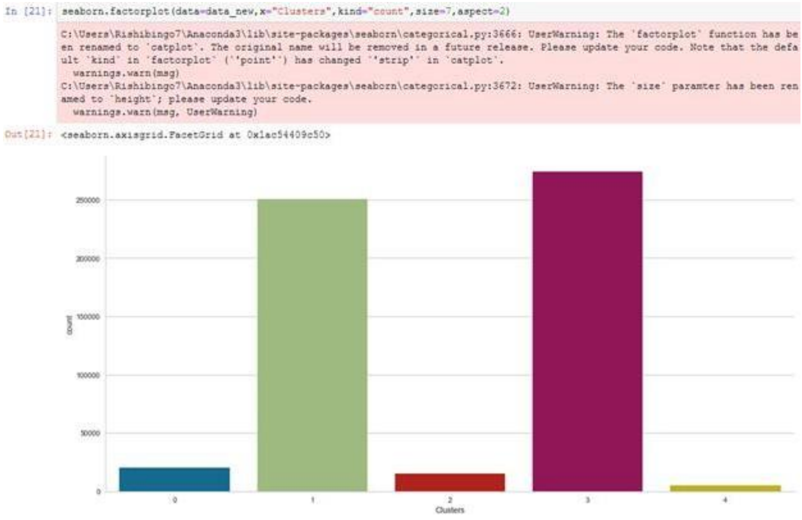


```
In [17]: import folium
         centroid = clocation.values.tolist()

         map = folium.Map(location = [40.71600413400166, -73.98971408426613], zoom_start = 10)
         for point in range(0, len(centroid)):
             folium.Marker(centroid[point], popup = centroid[point]).add_to(map)

         map
Out[17]:
```

FIG 5.11 PLOTTING THE CENTROIDS ON MAP

Based on figures 5.4 to 5.11 we are discussing how the data is visualized on the basis of day,month and plotting all the points of the dataset on a x-y graph referencing latitude and longitude after plotting the points we proceed for data analysis using the algorithm.

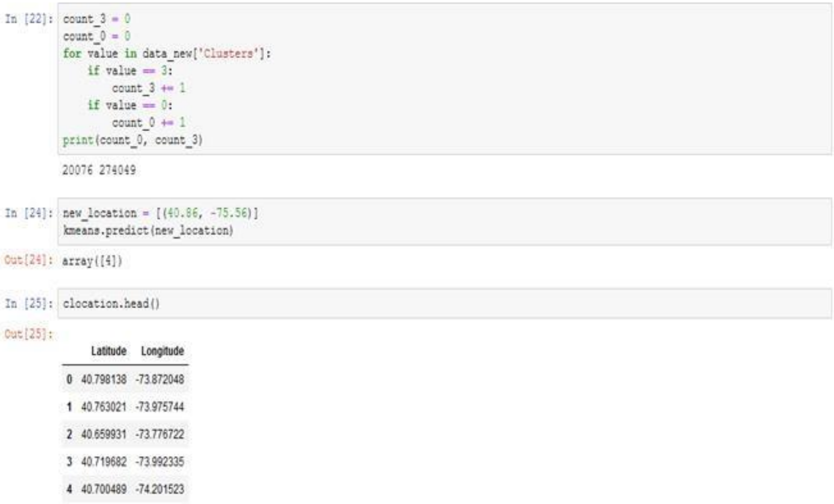
Figure 5.10 shows taking the data from the dataset and using the algorithm to calculate the centroids using k-means algorithm and plotting the points on the map.



```
In [21]: seaborn.factorplot(data=data_new,x="Clusters",kind="count",size=7,aspect=2)

         C:\Users\Rishibingo7\Anaconda3\lib\site-packages\seaborn\categorical.py:3666: UserWarning: The `factorplot` function has be
         en renamed to `catplot`. The original name will be removed in a future release. Please update your code. Note that the defa
         ult `kind` in `factorplot` (`'point'`) has changed `'strip'` in `catplot`.
           warnings.warn(msg)
         C:\Users\Rishibingo7\Anaconda3\lib\site-packages\seaborn\categorical.py:3672: UserWarning: The `size` paramter has been ren
         amed to `height`; please update your code.
           warnings.warn(msg, UserWarning)
Out[21]: <seaborn.axisgrid.FacetGrid at 0x1ac54409c50>
```

FIG 5.12TOTAL NUMBER OF TRIPS BASED ON CLUSTERS

Based on the data collected by the means of klabels by labelling the data in the dataset in the form of clusters. Fig 5.11 explains the total number of trips on the basis of clusters.



```
In [22]: count_3 = 0
         count_0 = 0
         for value in data_new['Clusters']:
             if value == 3:
                 count_3 += 1
             if value == 0:
                 count_0 += 1
         print(count_0, count_3)

         20076 274049
```

```
In [24]: new_location = [(40.86, -75.56)]
         kmeans.predict(new_location)

Out[24]: array([4])
```

```
In [25]: clocation.head()

Out[25]:
```

	Latitude	Longitude
0	40.798138	-73.872048
1	40.763021	-73.975744
2	40.659931	-73.776722
3	40.719682	-73.992335
4	40.700489	-74.201523

FIG 5.13 PREDICTION OF PICKUP OF CAB BASED ON THE CLUSTER

Based on the figures 5.1 to 5.12 we can discuss that using data analysis the program can predict the pickup location of the cab based on clusters analyzed and applied by k-means clustering to schedule the cab for pickup.

CHAPTER 5

CONCLUSION AND FUTURE WORK

CONCLUSION

This program will make the system of deploying more cabs to the required location and makes it flexible for users. The users have no need to worry about the location as the program will help in scheduling a cab for pickup nearest to the location. The program shows the concepts of machine learning such as data visualization and data analysis which makes the program and efficient for future work

FUTURE WORK

In future, system will provide the location of pickup to the users. Users can send their location to the app, and the program used in the project will predict the nearest location to the user and assign a cab to the user. The program and data elements in the program developed must be tested by Uber such that it can be used as an operational environment. It will make the program of predicting the trips using data analysis more flexible and efficient for users.

.

REFERENCES

[1]. Poulsen, L.K., Dekkers, D., Wagenaar, N., Snijders, W., Lewinsky, B., Mukkamala, R.R. and Vatrapu, R., 2016, June. Green Cabs vs. Uber in New York City. In 2016 IEEE International Congress on Big Data (BigData Congress) (pp. 222-229). IEEE.

[2]. Verma, N. and Baliyan, N., 2017, July. PAM clusteringbased taxi hotspot detection for informed driving. In 2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (pp. 1-7). IEEE.

[3]. Sotiropoulos, D.N., Pournarakis, D.E. and Giaglis, G.M., 2016, July. A genetic algorithm approach for topic clustering: A centroid-based encoding scheme. In 2016 7th International Conference on Information, Intelligence, Systems & Applications (IISA) (pp. 1-8). IEEE.

[4]. Guha, S. and Mishra, N., 2016. Clustering data streams. In Data stream management (pp.169-187). Springer, Berlin, Heidelberg.

[5]. Shah, D., Kumaran, A., Sen, R. and Kumaraguru, P., 2019, May. Travel Time EstimationAccuracy in Developing Regions: An Empirical Case Study with Uber Data in Delhi-NCR∗.In Companion Proceedings of The 2019 World Wide Web Conference (pp. 130-136). ACM.

[6]. Ahmed, M., Johnson, E.B. and Kim, B.C., 2018. The Impact of Uber and Lyft on TaxiService Quality Evidence from New York City. Available at SSRN 3267082.

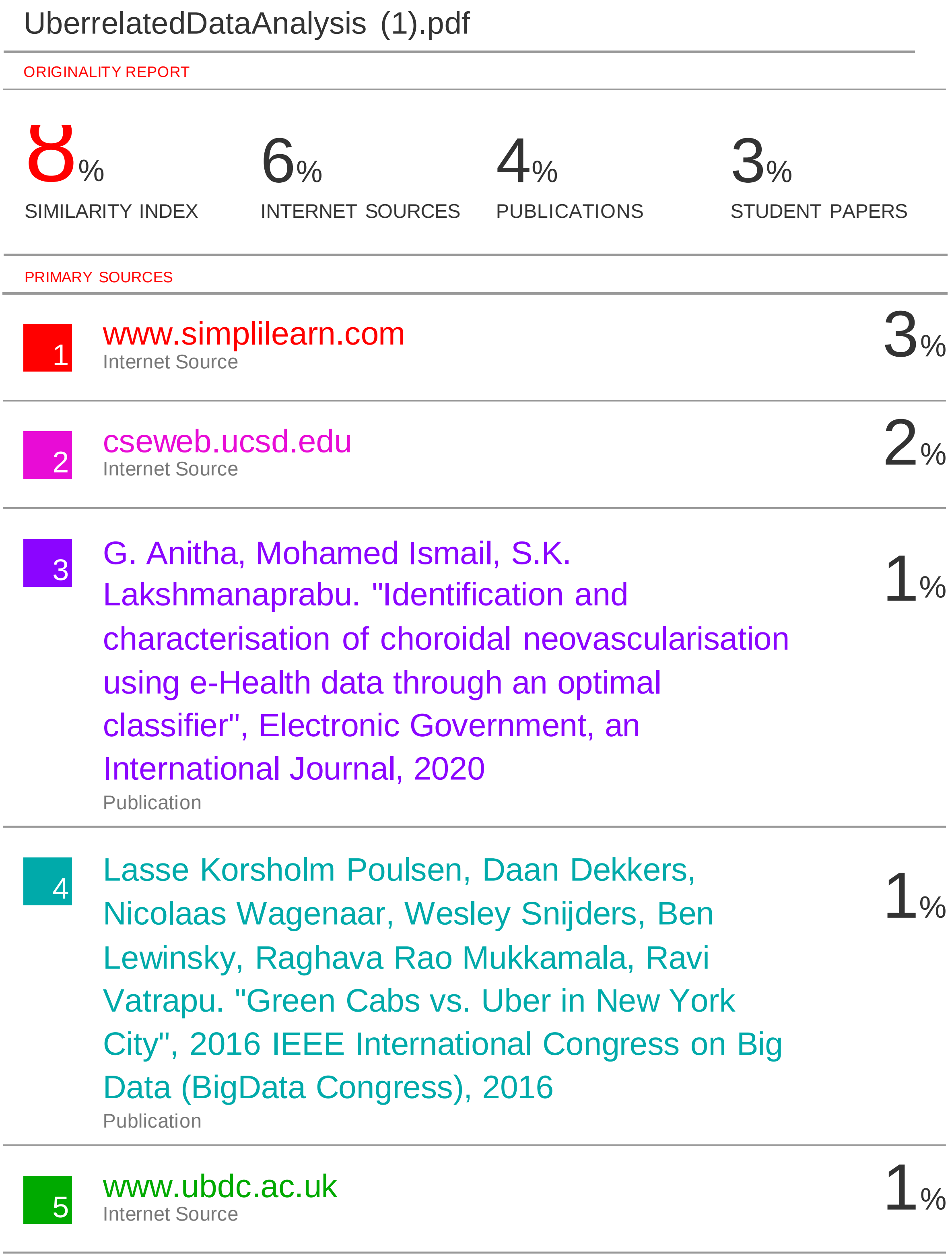
[7]. Wallsten, S., 2015. The competitive effects of the sharing economy: how is Uber changing taxis. Technology Policy Institute, 22, pp.1-21.

[8]. Verma, N. and Baliyan, N., 2017, July. PAM clustering based taxi hotspot detection forinformed driving. In 2017 8th International Conference on Computing, Communication andNetworking Technologies (ICCCNT) (pp. 1-7). IEEE.

[9]. Kumar, A., Surana, J., Kapoor, M. and Nahar, P.A., CSE 255 Assignment II Perfecting Passenger Pickups: An Uber Case Study.

APPENDIX

A) PLAGIARISM REPORT



UberrelatedDataAnalysis (1).pdf

ORIGINALITY REPORT

8%	6%	4%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.simplilearn.com Internet Source	3%
2	cseweb.ucsd.edu Internet Source	2%
3	G. Anitha, Mohamed Ismail, S.K. Lakshmanaprabu. "Identification and characterisation of choroidal neovascularisation using e-Health data through an optimal classifier", Electronic Government, an International Journal, 2020 Publication	1%
4	Lasse Korsholm Poulsen, Daan Dekkers, Nicolaas Wagenaar, Wesley Snijders, Ben Lewinsky, Raghava Rao Mukkamala, Ravi Vatrapu. "Green Cabs vs. Uber in New York City", 2016 IEEE International Congress on Big Data (BigData Congress), 2016 Publication	1%
5	www.ubdc.ac.uk Internet Source	1%

B) BASE PAPER

Uber Related Data Analysis using Machine Learning

Rishi Srinivas [1]

[1]UG Student, Department of CSE, Sathyabama Institute of Science and Technology,Chennai.India.
[1]mrchandrika2000@gmail.com,

B. **Ankayarkanni**[2]

[2] Associate Professor,Department of CSE, Sathyabama Institute of Science and Technology,India.
[2]ankayarkanni.s@gmail.com,

***Abstract*-The paper explains the working of an Uber dataset, which contains data produced by Uber for New York City. Uber is defined as a P2P platform. The platform links you to drivers who can take you to your destination. The dataset includes primary data on Uber pickups with details including the date, time of the ride as well as longitude-latitude information , Using the information, the paper explains the use of the k-means clustering algorithm on the set of data and classify the various parts of New York City. Since the industry is booming and expected to grow shortly. Effective taxi dispatching will facilitate each driver and passenger to reduce the wait time to seek out one another. The model is employed to predict the demand on points of the city.**

Keywords–Artificial Neural Network, Genetic Algorithms, K-means Clustering, Recurrent Neural Network, Time delay Neural Network, Convolutional Neural Network.

I.INTRODUCTION

The Uber platform connects you with drivers who can take you to your destination or location. This dataset includes primary data on Uber collections with details that include the date, time of travel, as well as information on longitude and latitude in San Francisco and has operations in over 900 metropolitan areas worldwide. The prediction of the frequency of trips of data is by implementing a part of k-means clustering algorithm

The standard algorithm describes the maximum variance within the group as the number of square distances Euclidean distances between the points and the corresponding centroid.The use of the digital computer has since moved to technology where the program involves the use of neural networks ,Examples of RNN (Recurrent Neural Network) and TDNN (Time delay Neural Network)for importing data from uber dataset which takes the data for forecasting on a time horizon.

The ultimate aim of the project is to predict the pickup of the cab on the basis of clusters defined by the k-means clustering algorithm. This algorithm is used to divide the dataset into k-groups. where k is defined as the number of groups provided by the user. The standard algorithm describes the maximum variance within the group as the number of square distances Euclidean distances between the points and the corresponding centroid.

The important packages used in the project are pandas,numpy,seaborn,kmeans,yellowbrick and folium.

II. LITERATURE SURVEY

Past few years have seen tremendous growth in uber related data analysis using machine learning. People are coming up with various methods to analyze uber related data such as A state in which the results, k-means clustering is used to estimate the most likely collection points at a given time and to predict the best hotspots of nightlife learning trends from previous Uber pickups. The center of the taxi service decides on the space of area to be targeted for the pickup of passengers.

This can be justified by explaining that machine learning is the core of Uber and how it has impacted on tremendous growth

- Bridging the supply demand gap
- Reduction in ETA
- Route Optimization

Poulsen, L.K In this document applied an experiment of spatial analysis of Green cab and Uber to hotspots of New York to determine the competitive position of the NYCTLC. The resulted research showed that as demand of green cabs on the hotspots grew,the demand of Uber taxis on the hotspots also growed.

This research recommends that NYCTLC creates a dashboard that analyzes and displays data in real time, as we believe this will increase its competitiveness compared to Uber. Uber is a recent taxi operator in New York and is constantly devouring the market share of the yellow and green taxis of the New York Taxi and Limousine Commission (NYCTLC).The NYCTLC is an agency of the New York City Government which licenses and regulates taxis and vehicle for hire industries and also app based companies. The commission was founded on March 2 ,1971 and their headquarters are based in New York.[1].

Faghih, S.S recommends a recent modeling approach in Manhattan, New York City, to capture the demand for electronic mail services, particularly the Uber application. Uber collection data is added to the Manhattan TAD level and at 15-minute time intervals. This aggregation allows the implementation of a modern approach to spatio-temporal modeling to obtain a spatial and temporal understanding of the demand. During a typical day, two spacetime models were developed using Uber collection data, the STAR and STAR and MSPE turns determine the output of the models. The results of the MSPE have shown that it is recommended to use the Lasso-Star system instead of the star design. A comparison between the demand for yellow and uber taxis in 2014 and 2015 in New York shows that the demand for uber has increased[2].

Ghuhaexplained the grouping of the sequences calculated and observed by using a small amount of memory and time was necessary for applications that needed to develop a data flow model to involve large data sets and consider categorizing the data in the form of clusters[3].

Ahmed, M., has shown that by using detailed data on taxis at the travel level and on the rental vehicle and data on complaints about the level of new complaints at the level of incidents, we study how Uber and Lyft enter damaged the quality of taxi services in New York City. The overall effect of the organizations based on the scenario and in particular of the riding administrations was enormous and widespread. One of these effects is the expansion of the rivalry between Uber and Lyft over the quality of taxi administration. They use a new set of complaint data to measure (the lack of) quality of service that we have never been analyzed before. Focus on the quality dimensions generated by most of the complaints we demonstrate. The increased competition for these shared travel services has had an intuitive impact on the behavior of taxi drivers[4].

Wallsten, S, stated that the results of New York and Chicago are consistent with the possibility that taxis react to the new challenge by improving quality. In New York, the rise of Uber is linked to the reduction of objections to travel to the city. They discuss the competitive effect of sharing taxis in the taxi industry using the complete data set of the New York City Taxi and Limousine Commission for more than one billion taxi trips in complaints and details of New York, New York and Chicago Google Trends on the success of Uber's largest shared travel service.[5].

Sotiropoulos, D.N, represented that this document addresses the problem of grouping, by using a new approach to genetic algorithms that is highly scalable in large volumes of textual details, developing a coding scheme based on centroids. We apply k means clustering algorithm in this document. Clustering is the unsupervised machine learning algorithm used to solve grouping problems based on similarities. This technique has aroused interest in a wide range of scientific fields, which address clustering methods, to solve complex classification problems.[6].

Faghih, S.S said that the demand for electronic mail services is growing rapidly, particularly in large cities. Uber is the first and most famous email company in the United States and New York City. A comparison between the demand for yellow and Uber taxis in New York in 2014 and 2015 shows that the demand for Uber has increased. To study the forecast performance of the models, you choose to choose data for a typical day. Our goal in this document is to describe how these models can be used for forecasting Uber demand. The Uber data contains information about the position and time of the pick-ups and returns of each trip during a day. According to the available data, the Uber historical data of April 2014[7].Kumar, states that, k-means clustering is used to estimate the most likely collection points at a given time and to predict the best hotspots of nightlife learning trends from previous Uber pickups [8,9].

L.Liu, C.Andris, and C.Ratti planned for a strategy to disclose cabdrivers working patterns by inspecting their unbroken anatomy track[10].R-H Hwand focuses on GPS and the locality to pick up passengers , A venue to venue plot model referred to as an OFF-ON model [11].

III. PROPOSED METHOD

Based on the problems of forecasting errors and risk of overfitting due to large datasets. The data analyzed and sent to the company is resulted as inefficient and ineffective. Thus to overcome the problem we are going to predict the pickup of cab from a coordinated cluster of points predicted by using applied k-means clustering algorithm.

The k-means clustering algorithm adopted will effectively dispatch taxis to the cluster.This facilitates each driver and passenger to attenuate the wait-time to search out one another. Drivers don’t have enough info concerning wherever passengers and different taxis area unit and shall move.Therefore, a cab center will organize the taxicab fleet and with efficiency give out consistent request to the whole town.

The system uses the latitude and longitude of the cab scheduled and also the day of the travel and the month.An unsupervised learning model is trained with this dataset and the model is employed to predict the pickup of the cab on the cluster.The proposed method for the project is explained on 7 steps.

A. System Architecture

B. Raw Data

C. Data Importing

D. Data Visualization

E. Testing Data

F. Predicted Scheduling Of Cab Using Algorithm

G. Algorithm

A. System Architecture

The system architecture for the given module is as follows:

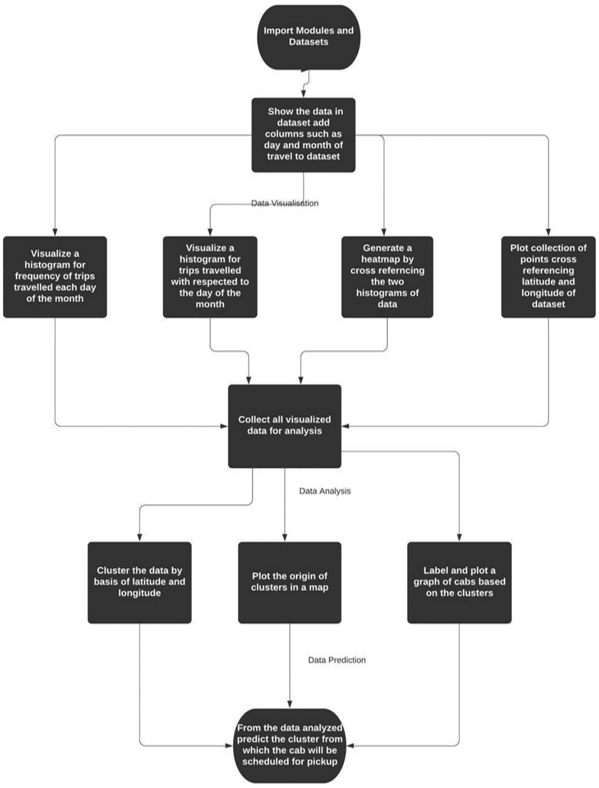
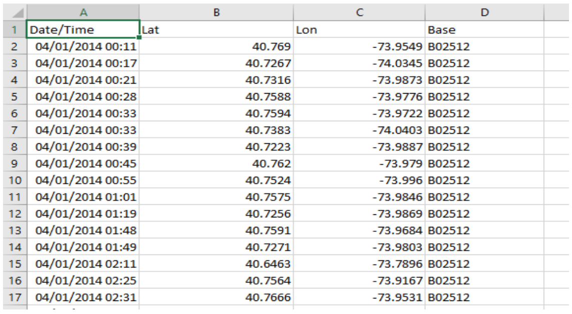

Import Modules and Datasets
Show the data in dataset add columns such as day and month of travel to dataset
Data Visualisation
Visualize a histogram for frequency of trips travelled each day of the month
Visualize a histogram for trips travelled with respected to the day of the month
Generate a heatmap by cross referncing the two histograms of data
Plot collection of points cross referencing latitude and longitude of dataset
Collect all visualized data for analysis
Data Analysis
Cluster the data by basis of latitude and longitude
Plot the origin of clusters in a map
Label and plot a graph of cabs based on the clusters
Data Prediction
From the data analyzed predict the cluster from which the cab will be scheduled for pickup

Fig 1. System Architecture

B. Raw Data(Dataset)

The definition of raw data comes from the concept of data not processed and is obtained from the dataset or is sometimes made by the end product of data processing. The steps required in raw data are extraction, organization and sometimes analysis.

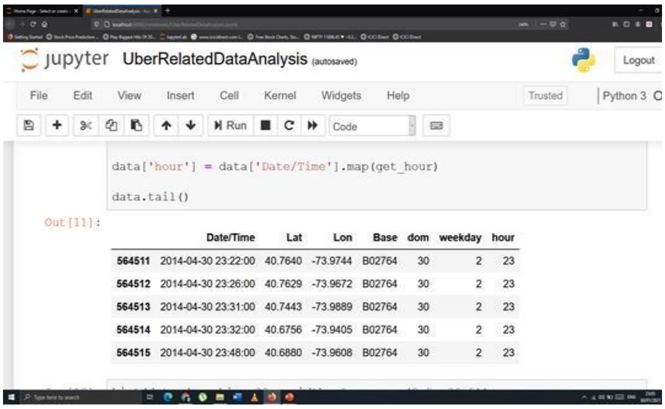


	A	B	C	D
1	Date/Time	Lat	Lon	Base
2	04/01/2014 00:11	40.769	-73.9549	B02512
3	04/01/2014 00:17	40.7267	-74.0345	B02512
4	04/01/2014 00:21	40.7316	-73.9873	B02512
5	04/01/2014 00:28	40.7588	-73.9776	B02512
6	04/01/2014 00:33	40.7594	-73.9722	B02512
7	04/01/2014 00:33	40.7383	-74.0403	B02512
8	04/01/2014 00:39	40.7223	-73.9887	B02512
9	04/01/2014 00:45	40.762	-73.979	B02512
10	04/01/2014 00:55	40.7524	-73.996	B02512
11	04/01/2014 01:01	40.7575	-73.9846	B02512
12	04/01/2014 01:19	40.7256	-73.9869	B02512
13	04/01/2014 01:48	40.7591	-73.9684	B02512
14	04/01/2014 01:49	40.7271	-73.9803	B02512
15	04/01/2014 02:11	40.6463	-73.7896	B02512
16	04/01/2014 02:25	40.7564	-73.9167	B02512
17	04/01/2014 02:31	40.7666	-73.9531	B02512

Fig 2. Raw dataset (csv file)

C. Data Importing

A huge amount of trip data will be collected fromUberfor training and testingdata. From the collected dataset the latitude and latitude will be clustered and classified based on the frequency of trips travelled by the cab during the day. When these criteria are considered, and data preprocess will be done on these datasets.



```
data['hour'] = data['Date/Time'].map(get_hour)

data.tail()
```

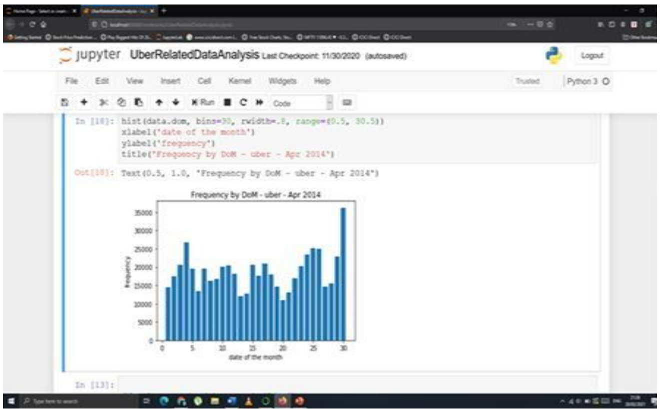
Out[11]:

	Date/Time	Lat	Lon	Base	dom	weekday	hour
564511	2014-04-30 23:22:00	40.7640	-73.9744	B02764	30	2	23
564512	2014-04-30 23:26:00	40.7629	-73.9672	B02764	30	2	23
564513	2014-04-30 23:31:00	40.7443	-73.9889	B02764	30	2	23
564514	2014-04-30 23:32:00	40.6756	-73.9405	B02764	30	2	23
564515	2014-04-30 23:48:00	40.6880	-73.9608	B02764	30	2	23

Fig 3. Processing the dataset

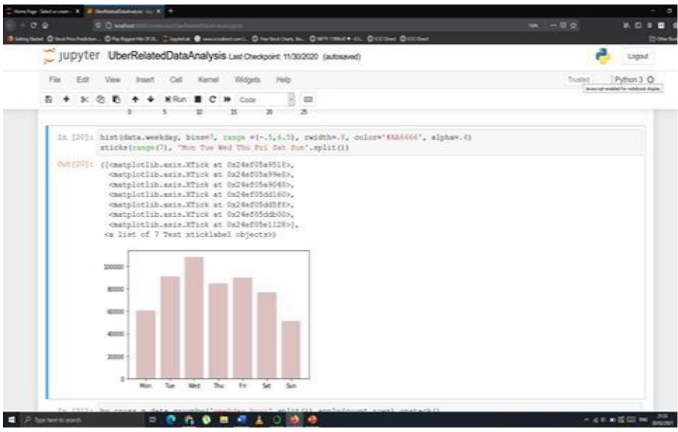
D. Data Visualization

Data visualization is defined as to evaluate the performance of a model by using graphs and metrics that calculate performance.Data visualization can be mainly used to categorize the data into new levels such that the algorithm used can be generalized to an observation of each output variable derived by an observed input variable.



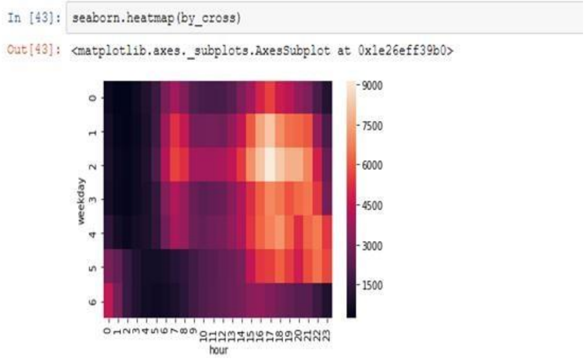
```
In [18]: hist(data.dom, bins=30, rwidth=.8, range=(0.5, 30.5))
         xlabel('date of the month')
         ylabel('frequency')
         title('Frequency by DoM - uber - Apr 2014')

Out[18]: Text(0.5, 1.0, 'Frequency by DoM - uber - Apr 2014')
```

Fig 4. Data Visualization of the data based on graph where graph has data of total trips travelled during the month .



```
In [20]: hist(data.weekday, bins=7, range =(-.5,6.5), rwidth=.8, color='#AA6666', alpha=.4)
         xticks(range(7), 'Mon Tue Wed Thu Fri Sat Sun'.split())

Out[20]: ([<matplotlib.axis.XTick at 0x24ef05a9518>,
           <matplotlib.axis.XTick at 0x24ef05a99e8>,
           <matplotlib.axis.XTick at 0x24ef05a9048>,
           <matplotlib.axis.XTick at 0x24ef05dd160>,
           <matplotlib.axis.XTick at 0x24ef05dd5f8>,
           <matplotlib.axis.XTick at 0x24ef05ddb00>,
           <matplotlib.axis.XTick at 0x24ef05e1128>],
          <a list of 7 Text xticklabel objects>)
```

Fig 5.Visualizing data on graph on frequency of trips travelled during the day.



```
In [43]: seaborn.heatmap(by_cross)

Out[43]: <matplotlib.axes._subplots.AxesSubplot at 0x1e26eff39b0>
```

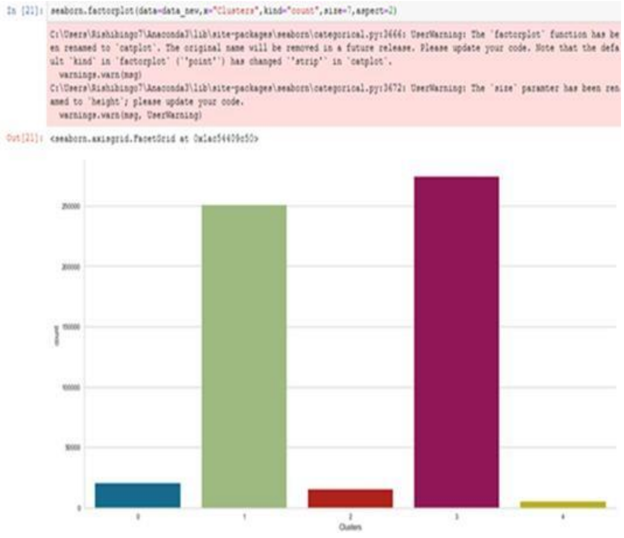
Fig 6 Data Visualization of Heatmap of frequency of cabs travelled during the hour.

E. Testing data

The main step after visualizing data in an algorithm is to test the data,the test set can be defined as a set of observations which is used to evaluate the performance of a model by using performance metrics.The program that uses the test set must be able to generalize and effectively perform with the dataset to yield the predicted data accurately such that the program is effective in nature. Moreover when the program memorizes the dataset it is termed overfitting hence to balance overfitting we use regularization which is applied to the model to reduce it.

F. Predicted Scheduling of Cab using Algorithm

The scheduling of the cab can be predicted on the basis of the location given by the user and the proposed method finds the nearest hotspot which is defined as a cluster of points analyzed by k-means clustering and gives info to the cab on the hotspot nearest to the location of the user and is booked to pickup the user.



```
In [21]: seaborn.factorplot(data=data_new,x="Clusters",kind="count",size=7,aspect=2)

C:\Users\Rishibingo7\Anaconda3\lib\site-packages\seaborn\categorical.py:3666: UserWarning: The `factorplot` function has be
en renamed to `catplot`. The original name will be removed in a future release. Please update your code. Note that the defa
ult `kind` in `factorplot` (`'point'`) has changed `'strip'` in `catplot`.
  warnings.warn(msg)
C:\Users\Rishibingo7\Anaconda3\lib\site-packages\seaborn\categorical.py:3672: UserWarning: The `size` paramter has been ren
amed to `height`; please update your code.
  warnings.warn(msg, UserWarning)

Out[21]: <seaborn.axisgrid.FacetGrid at 0x1ac54409c50>
```

Fig.7.Visualization of total trips travelled by cabs on a day on the basis of clusters.

G. Algorithm

The algorithm used is involved on the concept of k-means clustering algorithm. which belongs to unsupervised learning. There is no labeled data for this clustering, unlike in supervised learning. K-Means performs division of objects into clusters that share similarities and are dissimilar to the objects belonging to another cluster.

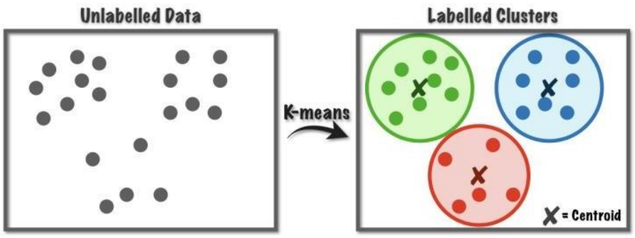


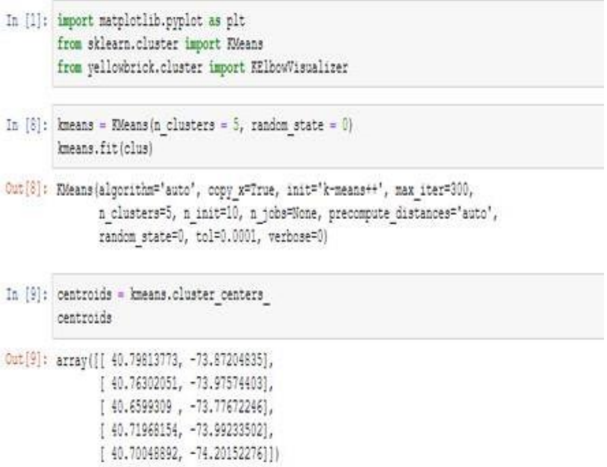
Fig.8. K means clustering

The clustering algorithm is categorized into three steps:

- Take mean value
- Find nearest number of mean and put in cluster
- Repeat the mean value and number of means in the cluster till we get same mean

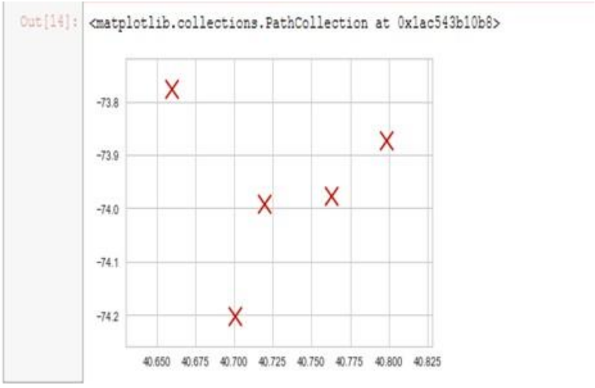
Clustering is the method of grouping objects into groups based on similarities. This algorithm is used to divide a given data set into k groupsHere, k represents the number of groups and must be provided by the user. The idea behind the grouping of k-means is to identify the clusters in such a way to reduce total variation within the cluster. The standard algorithm describes the maximum variance within the group as the number of square distances Euclidean distances between the points and the corresponding centroid. The grouping can be classified into two groups.

Hard grouping and soft grouping. Through a hard grouping, each object or data point belongs to a cluster. For example, all locations clustered in the dataset belong to a district.In the soft grouping, a data point can belong to more than one group with a certain probability or probability value. In connectivity-based clustering, the main idea behind this cluster is that the data points closest to the data space are more related than those of the data point further back. Groups are created by linking data points based on their length. Grouping based on centroids, in this type of grouping, the groups are represented by a central vector or a centroid. This centroid may not necessarily be a member of the data set. This is an iterative grouping algorithm in which the notion of similarity derives from the proximity of the data point to the center of the cluster.



```
In [1]: import matplotlib.pyplot as plt
        from sklearn.cluster import KMeans
        from yellowbrick.cluster import KElbowVisualizer

In [8]: kmeans = KMeans(n_clusters = 5, random_state = 0)
        kmeans.fit(clus)

Out[8]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
               n_clusters=5, n_init=10, n_jobs=None, precompute_distances='auto',
               random_state=0, tol=0.0001, verbose=0)

In [9]: centroids = kmeans.cluster_centers_
        centroids

Out[9]: array([[ 40.79813773, -73.87204835],
               [ 40.76302051, -73.97574403],
               [ 40.6599309 , -73.77672246],
               [ 40.71968154, -73.99233502],
               [ 40.70048892, -74.20152276]])
```

Fig.9 Importing k-means module and calculating the centroids of the dataset



```
Out[14]: <matplotlib.collections.PathCollection at 0x1ac543b10b8>
```

Fig 10. Plotting the centroids on x-y graph of latitude and longitude.

IV. RESULTS AND DISCUSSION

The program predicts the pickup location of the cab based on the centroids plotted using applied by k-means clustering for appropriate cab scheduled for pickup.The results discussed are based on the following figures below

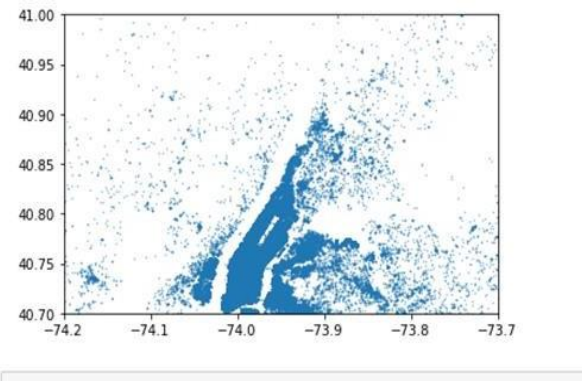


Fig.11 Plotting the collection of points through which cab has travelled during the course of the month.

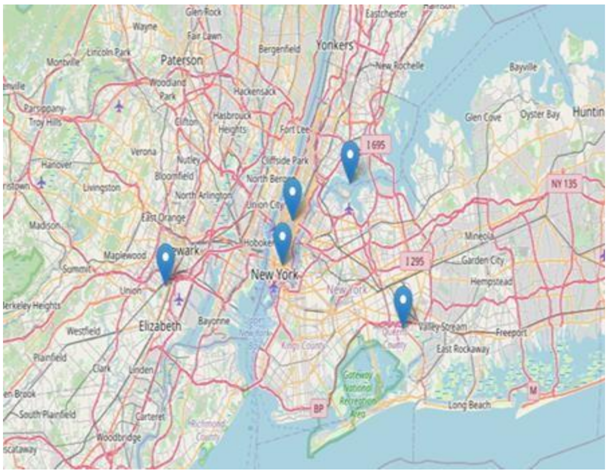
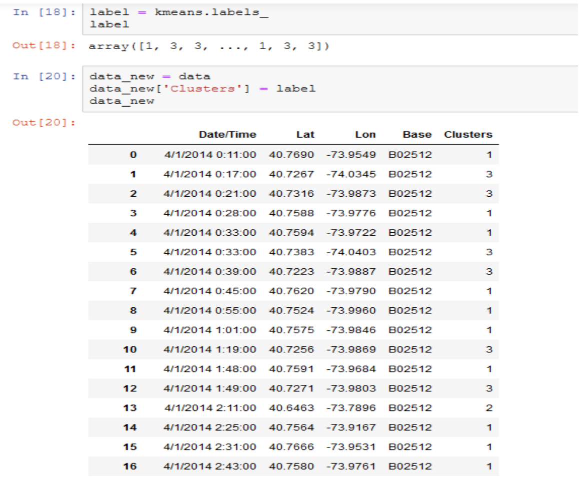


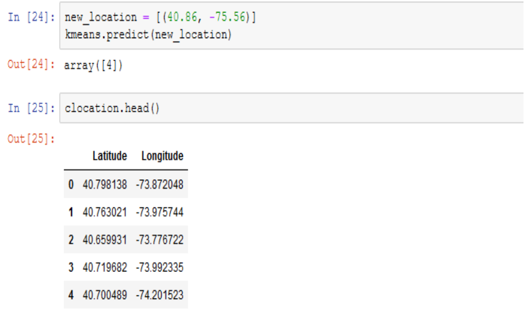
Fig. 12. Plotting the centroids calculated by k-means on the map of New York City imported by Folium.



```
In [18]: label = kmeans.labels_
         label

Out[18]: array([1, 3, 3, ..., 1, 3, 3])

In [20]: data_new = data
         data_new['Clusters'] = label
         data_new

Out[20]:
```

	Date/Time	Lat	Lon	Base	Clusters
0	4/1/2014 0:11:00	40.7690	-73.9549	B02512	1
1	4/1/2014 0:17:00	40.7267	-74.0345	B02512	3
2	4/1/2014 0:21:00	40.7316	-73.9873	B02512	3
3	4/1/2014 0:28:00	40.7588	-73.9776	B02512	1
4	4/1/2014 0:33:00	40.7594	-73.9722	B02512	1
5	4/1/2014 0:33:00	40.7383	-74.0403	B02512	3
6	4/1/2014 0:39:00	40.7223	-73.9887	B02512	3
7	4/1/2014 0:45:00	40.7620	-73.9790	B02512	1
8	4/1/2014 0:55:00	40.7524	-73.9960	B02512	1
9	4/1/2014 1:01:00	40.7575	-73.9846	B02512	1
10	4/1/2014 1:19:00	40.7256	-73.9869	B02512	3
11	4/1/2014 1:48:00	40.7591	-73.9684	B02512	1
12	4/1/2014 1:49:00	40.7271	-73.9803	B02512	3
13	4/1/2014 2:11:00	40.6463	-73.7896	B02512	2
14	4/1/2014 2:25:00	40.7564	-73.9167	B02512	1
15	4/1/2014 2:31:00	40.7666	-73.9531	B02512	1
16	4/1/2014 2:43:00	40.7580	-73.9761	B02512	1

Fig.13. Labelling the dataset on the basis of clusters



```
In [24]: new_location = [(40.86, -75.56)]
         kmeans.predict(new_location)

Out[24]: array([4])

In [25]: clocation.head()

Out[25]:
```

	Latitude	Longitude
0	40.798138	-73.872048
1	40.763021	-73.975744
2	40.659931	-73.776722
3	40.719682	-73.992335
4	40.700489	-74.201523

Fig. 14. Predicting the pickup of the cab from the particular cluster and showing the cluster coordinates.

V. CONCLUSION AND FUTURE WORK

The conclusion of the project is to project a basic outline of trips travelled with respect to latitude and longitude of locations and pinpoint the locations travelled with respect to the frequency of trips travelled by a uber cab during the day and also based on the cross analyzing of the dataset based on the latitude and longitude of the point travelled by the cab which is then analyzed by deploying k-means clustering which classifies the locations on the basis of centroids and then orders the frequency of trips based on labels or clusters. By the location given by the user,the algorithm predicts the cluster nearest to the location so that cab can be assigned to the user for pickup.

The merits of the project is that it explains the functioning of how cabs are assigned to passengers based on an unsupervised algorithm and also explains the key concepts of machine learning.The limitations of the project are that the algorithm deployed may be inefficient for huge data for over 10 years.

The future work suggests that the system will provide the location to the user. The algorithm then records the time,latitude ,longitude of the trip and assigns it to a cluster nearest to the passenger location where a cab is scheduled for pickup. We can also predict the passenger count on each district to deploy more cabs to the clustered coordinates using convolutional neural networks (CNN)

REFERENCES

[1] Poulsen, L.K., Dekkers, D., Wagenaar, N., Snijders, W., Lewinsky, B., Mukkamala, R.R.andVatrapu, R., 2016, June. Green Cabs vs. Uber in New York City. In 2016 IEEEInternational Congress on Big Data (BigData Congress) (pp. 222-229). IEEE.

[2] Faghih, S.S., Safikhani, A., Moghimi, B. and Kamga, C., 2017. Predicting Short-Term UberDemand Using Spatio-Temporal Modeling: A New York City Case Study. arXiv preprintarXiv:1712.02001.

[3] Guha, S. and Mishra, N., 2016. Clustering data streams. In Data stream management (pp.169-187). Springer, Berlin, Heidelberg.

[4] Ahmed, M., Johnson, E.B. and Kim, B.C., 2018. The Impact of Uber and Lyft on TaxiService Quality Evidence from New York City. Available at SSRN 3267082.

[5] Wallsten, S., 2015. The competitive effects of the sharing economy: how is Uber changingtaxis. Technology Policy Institute, 22, pp.1-21.

[6] Sotiropoulos, D.N., Pournarakis, D.E. and Giaglis, G.M., 2016, July. A genetic algorithmapproach for topic clustering: A centroid-based encoding scheme. In 2016 7th InternationalConference on Information, Intelligence, Systems & Applications (IISA) (pp. 1-8). IEEE

[7] Faghih, S.S., Safikhani, A., Moghimi, B. and Kamga, C., 2019. Predicting Short-TermUber Demand in New York City Using Spatiotemporal Modeling. Journal of Computing inCivil Engineering, 33(3), p.05019002.

[8] Shah, D., Kumaran, A., Sen, R. and Kumaraguru, P., 2019, May. Travel Time EstimationAccuracy in Developing Regions: An Empirical Case Study with Uber Data in Delhi-NCR∗.In Companion Proceedings of The 2019 World Wide Web Conference (pp. 130-136). ACM.

[9] Kumar, A., Surana, J., Kapoor, M. and Nahar, P.A., CSE 255 Assignment II PerfectingPassenger Pickups: An Uber Case Study.

[10] L.Liu, C.Andris, and C.Ratti , “Uncovering cabdrivers behaviour patterns from their digital traces”,Compu. Environ.UrbanSyst.,vol.34,no.6,pp.541-548,2010

[11] R.H.Hwang,Y.L.Hsueh , and Y.T.Chen,”An effective taxi recommender system model on a spatio-temporal factor analysis model,”Inf.Sci.,vol.314,pp.28-40,2015.

[12] Vigneshwari, S., and M. Aramudhan. "Web information extraction on multiple ontologies based on concept relationships upon training the user profiles." In Artificial Intelligence and Evolutionary Algorithms in Engineering Systems, pp. 1-8. Springer, New Delhi, 2015.

[13] L. Rayle, D. Dai, N. Chan, R. Cervero, and S. Shaheen, “Just a better taxi? a survey-based comparison of taxis, transit, and ridesourcing services in san francisco,” Transport Policy, vol. 45, 01 2016.

[14] O. Flores and L. Rayle, “How cities use regulation for innovation: the case of uber, lyft and sidecar in san francisco,” Transportation research procedia, vol. 25, pp. 3756–3768, 2017.

[15] H. A. Chaudhari, J. W. Byers, and E. Terzi, “Putting data in the driver’s seat: Optimizing earnings for on-demand ride-hailing,” in Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining. ACM, 2018, pp. 90–98.

C) SOURCE CODE

```
%pylab inline
import pandas
import seaborn
data=pandas.read_csv('Desktop/uber.csv')
data
data.tail()
data['Date/Time'] = data['Date/Time'].map(pandas.to_datetime)
def get_dom(dt):
return dt.day
data['dom'] = data['Date/Time'].map(get_dom)
data.tail()
def get_weekday(dt):
 return dt.weekday()
data['weekday'] = data['Date/Time'].map(get_weekday)
def get_hour(dt):
 return dt.hour
data['hour'] = data['Date/Time'].map(get_hour)
data.tail()
hist(data.dom, bins=30, rwidth=.8, range=(0.5, 30.5))
xlabel('date of the month')
ylabel('frequency')
title('Frequency by DoM - uber - Apr 2014')
```

```
def count_rows(rows):

return len(rows)
by_date = data.groupby('dom').apply(count_rows)
by_date
bar(range(1, 31), by_date)
by_date_sorted = by_date.sort_values()
by_date_sorted
bar(range(1, 31), by_date_sorted)
xticks(range(1,31), by_date_sorted.index)
xlabel('date of the month')
ylabel('frequency')
title('Frequency by DoM - uber - Apr 2014');
hist(data.hour, bins=24, range=(.5, 24))
hist(data.weekday, bins=7, range =(-.5,6.5), rwidth=.8, color='#AA6666', alpha=.4)
xticks(range(7), 'Mon Tue Wed Thu Fri Sat Sun'.split())
by_cross = data.groupby('weekday hour'.split()).apply(count_rows).unstack()
seaborn.heatmap (by_cross)
hist(data['Lat'], bins=100, range = (40.5, 41));
hist(data['Lon'], bins=100, range = (-74.1, -73.9));
hist(data['Lon'], bins=100, range = (-74.1, -73.9), color='g', alpha=.5, label = 'longitude')
grid()
legend(loc='upper left')
twiny()
hist(data['Lat'], bins=100, range = (40.5, 41), color='r', alpha=.5, label = 'latitude')
legend(loc='best');
```

```
figure(figsize=(20, 20))

plot(data['Lon'], data['Lat'], '.', ms=1, alpha=.5)

xlim(-74.2, -73.7) ylim(40.7, 41)

clus = data[['Lat','Lon']]

clus.dtypes

pip install sklearn

pip install yellowbrick

import matplotlib.pyplot as plt

from sklearn.cluster import KMeans

from yellowbrick.cluster import KElbowVisualizer

kmeans = KMeans(n_clusters = 5, random_state = 0)

kmeans.fit(clus)

centroids = kmeans.cluster_centers_

centroids

clocation = pandas.DataFrame(centroids, columns = ['Latitude', 'Longitude'])

clocation.head()

plt.scatter(clocation['Latitude'], clocation['Longitude'], marker = "x", color = 'R', s = 200)

pip install folium

import folium

centroid = clocation.values.tolist()

map = folium.Map(location = [40.71600413400166, -73.98971408426613], zoom_start = 10)

for point in range(0, len(centroid)):

folium.Marker(centroid[point], popup = centroid[point]).add_to(map)

map

label = kmeans.labels_

label

data_new = data

data_new['Clusters'] = label

data_new
```

```
seaborn.factorplot(data=data_new,x="Clusters",kind="count",size=7,aspect=2)
count_3 = 0
count_0 = 0
for value in data_new['Clusters']:
    if value == 3:
        count_3 += 1
    if value == 0:
        count_0 += 1
print(count_0, count_3)
new_location = [(40.86, -75.56)]
kmeans.predict(new_location)
clocation.head()
```