AN IMPROVED MULTI-LAYER CLUSTERING APPROACH FOR ENHANCING INFORMED TAXI DRIVING

Thesis submitted in partial fulfillment of the requirements for the award of degree of

Master of Engineering

in

Computer Science and Engineering

Submitted By

Niharika Verma

(Roll No. 801532036)

Under the supervision of:

Dr. Niyati Baliyan

Assistant Professor

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT

THAPAR UNIVERSITY

PATIALA – 147004

July 2017

Certificate

I hereby certify that the work which is being presented in the thesis titled, "**AN IMPROVED MULTI-LAYER CLUSTERING APPROACH FOR ENHANCING INFORMED TAXI DRIVING** ", in partial fulfillment of the requirements for the award of degree of Master of Engineering in Computer Science and Engineering submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of Dr. Niyati Baliyan and refers other researchers' work to the best of my knowledge which are duly listed in the reference section. The matter presented in the thesis has not been submitted for award of any other degree of this or any other University.

Date: 17/08/17

Niharika Verma
Candidate

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date: 17/08/17

Dr. Niyati Baliyan
Supervisor

Acknowledgements

The tenure from the beginning till the completion of my thesis has been a great learning paradigm, which has not only honed my research and problem-solving skills but has also widened my knowledge base. My thesis would not have been complete without the constant support and right guidance of my advisors, whom I would like to thank for.

First and foremost, I would like to offer my sincerest gratitude to my supervisor, **Dr. Niyati Baliyan** who has always motivated and supported me throughout my thesis. She has always shown me the right path to achieve my objectives with their vast knowledge and insurmountable patience. My supervisor has provided constant support and made all the resources available at my disposal.

I would also like to thank **Dr.Maninder Singh**, Head, Computer Science and Engineering Department, Thapar University for his support and cooperation. I acknowledge the efforts of the complete staff and faculty of Computer Science and Engineering Department, Thapar University to provide me with adequate facilities required for the completion of this thesis.

Finally, I would like to express my gratitude to my peers for stimulating discussions and family members for always inspiring me which kept me going.

Date: 17/08/17

Niharika Verma

Niharika Verma
801532036
ME-CSE

Abstract

A huge amount of spatio-temporal data is generated by millions of cabs in metro cities all around the world. This data, if analyzed correctly, can provide a better understanding of the taxi demand. With the increasing preference of customers to have a hassle-free experience, cabs are becoming the ultimate choice for all due to their point to point service. The inability of the taxi companies to meet ever increasing demand of cabs leads to high unavailability of cabs during peak hours and low usage during non peak hours. This taxi imbalance problem can be resolved by analyzing the spatial data and predicting the demand hotspots to identify areas with potential passengers.

Moreover, with the increasing demand of cabs around the city, the cabs need to be dispatched in such a way that the average waiting time of the cabs and the cancellation rate is reduced at the same time. This can be achieved by selecting appropriate places in the city for the cab bases to be setup. The knowledge about the base setup can help map the nearest cab of the taxi base to a pickup location. Hence, this thesis aims at filtering the data on various parameters such as day of the week, time of the day, nearest taxi base etc. to segregate the data for further useful insights.

In this thesis, a multi-layer clustering approach is implemented for hotspot detection and selection of taxi base setup location. Any new incoming request is first mapped to the nearest taxi base for allocation of cab and then it is identified in which hotspot region, the area falls. Using this approach, a region specific allocation of cabs is enhanced. Clustering techniques are used on the filtered dataset to provide the popularity of regions in the city at different timings. K-means, k-medians and CLARA clustering techniques are used and the results from these clustering techniques are compared. The clustering technique that provides the best results for spatial data is chosen for hotspot detection and taxi base setup based on the day and time. This date and time based hotspot detection helps the taxi companies to dispatch the cabs at locations predicted to have higher number of passengers in future resulting in better service for customers and increased revenue for the shareholders.

Table of Contents

Title **Page No.**

Abstract . . . . . **v**

Table of Contents . . . . . **vi**

List of Figures . . . . . **ix**

List of Tables . . . . . **x**

Chapter 1 Introduction **1**

1.1 Introduction to Cab Services 2

1.2 Taxi Imbalance Problem 3

1.3 Supervised and Unsupervised Learning 4

1.3.1 Supervised Learning 4

1.3.2 Unsupervised Learning 4

1.4 Data Collection and Pre-Processing 5

1.5 Outlier Detection 6

1.5.1 Turkey's Method of Outlier Detection 7

1.5.2 Outlier Removal 7

1.6 Spatial Clustering 8

1.6.1 Partitioning Methods 9

1.6.2 Hierarchical Methods 12

1.6.3 Density Based Methods 13

1.6.4 Grid Based Methods 16

1.7 Cluster Validation 17

1.7.1 Internal Quality Measures 18

1.7.2 External Quality Measures 19

1.8 Hotspot Detection 20

1.9 Base Location Setup 21

1.10 Thesis Organization 21

Chapter 2 Literature Review **23**

2.1 Related Work 23

Chapter 3 Problem Statement **31**
3.1 Research Motivation 31
3.2 Problem Statement 32
3.3 Research Objective 32

Chapter 4 Proposed Work **35**
4.1 About the Dataset 35
4.2 Proposed Model 37
4.3 Data Integration and Transformation 40
4.3.1 Data Integration 40
4.3.2 Data Transformation 40
4.4 Data Visualization 41
4.5 Model Creation 42
4.5.1 Data Grouping 42
4.5.2 Data Filtering 42
4.5.3 Outlier Detection 43
4.5.4 Outlier Removal 43
4.5.5 Cluster Number Selection 43
4.5.6 Clustering 44
4.5.7 Cluster Validation 44
4.5.8 Hotspot Detection 45
4.5.9 Base Address Allocation 45

Chapter 5 Simulation and Results **47**
5.1 Research Methodology 47
5.2 Data Visualization Results 49
5.2.1 Daily Trends 49
5.2.2 Hourly Trends 50
5.2.3 Weekly Trends 51
5.2.4 Base Specific Trends 52
5.2.5 Demand Based Data Splitting and Grouping 53
5.2.6 Morning Trip Trends 54
5.2.7 Evening Trip Trends 55
5.3 Outlier Detection and Removal 56
5.4 Clustering Results 57
5.5 Internal Cluster Validation 61
5.6 External Cluster Validation 64
5.7 Base Address Clustering Results 68

Chapter 6 Conclusion and Future Work **69**
6.1 Conclusion 69
6.2 Future Work 71

References **73**

List of Publications **77**

List of Figures

Figure No.	Title	Page No.
1.1	Data Mining	1
1.2	DBSCAN Clustering	14
1.3	OPTICS Clustering	15
1.4	STING Clustering	16
4.1	Original DataSet	35
4.2	Pickup Locations	36
4.3	Proposed Model for Hotspot Detection	37
4.4	Proposed Model for Base Setup Location	39
4.5	Transformed DataSet	40
5.1	Trips By Day	49
5.2	Trips By Hour	50
5.3	Trips By Weekday	51
5.4	Trips By Base	52
5.5	Hourly Trips by Weekday	53
5.6	Hourly Morning Trips by Weekday	54
5.7	Hourly Evening Trips by Weekday	55
5.8	Weekday Morning Outlier Detection and Removal	56
5.9	Weekday Evening Outlier Detection and Removal	56
5.10	Weekend Outlier Detection and Removal	57
5.11	Weekday Morning Clustering Algorithms	58
5.12	Weekday Evening Clustering Algorithms	59
5.13	Weekend Clustering Algorithms	60
5.14	Davies Bouldin Index	62
5.15	Calinhara Index	63
5.16	Sum of Squared Error	63
5.17	General Rand Index	66
5.18	Adjusted Rand Index	66
5.19	Fowlkes Index	67
5.20	Mirkin Metrics	67
5.21	Base Address Clustering	68

List of Tables

Table No.	Title	Page No.
2.1	Summary of Related Work on Taxi Trips Data	28
2.2	Summary of Related Work on Taxi Trajectory Data	29
5.1	Libraries and Functions Used Part 1	47
5.2	Libraries and Functions Used Part 2	48
5.3	Weekday Morning Cluster Quality Analysis	61
5.4	Weekday Evening Cluster Quality Analysis	61
5.5	Cluster Quality Analysis	61
5.6	Cluster Comparison for Weekday Morning Dataset	64
5.7	Cluster Comparison for Weekday Evening Dataset	65
5.8	Cluster Comparison for Weekend Dataset	65

Chapter 1

Introduction

Data mining refers to the technique of extracting meaningful information from raw data, which can be further used for data analysis and prediction. This technique acts as an important tool for large companies to analyze the huge amount of data generated each day, extract information out of the data and use them it for strategic planning and decision making. Better decision making is achieved by observing the past trends in the data and use for predicting future trends and behaviors. This strategic planning is important for higher business profits, minimizing loss of customers, set business targets, etc.

In order to extract knowledge from the data, a data mining process needs to be followed which is a stepwise process. The figure 1.1 describes the data mining steps in detail.

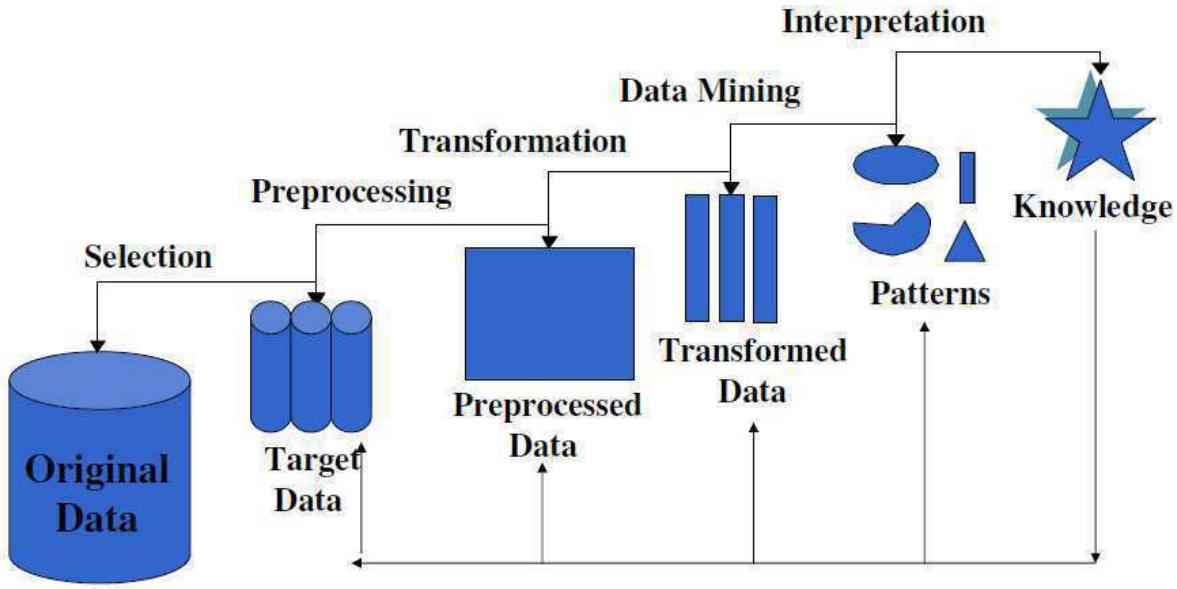


Figure 1.1: Data Mining

1. **Data Collection** : This process includes the collection of data from warehouses to have enough data that can be used for analysis. The analysis results then are able to provide useful insight into the data.

2. **Target Dataset** : Out of the data obtained in the previous step, a target dataset needs to be selected since not all the data obtained in Step 1 is of use. Subsets of

data can provide much accurate data to be used for knowledge extraction.

3. **Data Pre-Processing** : Data cleaning and transformation is necessary for all the techniques to work accurately on the target dataset. Hence, data preprocessing is a necessary process. It includes dimensionality reduction, noise removal, outlier detection, treating missing values, etc.

4. **Data Mining Techniques** : Various data mining techniques are available which can be applied to the target dataset. Algorithms like clustering, associations, classification, etc. can be selected depending to the domain of the target dataset.

5. **Interpretation** : The results obtained from Step 4 are then interpreted and patterns are visualized to find the interesting facts from the target dataset. Multiple iterations using different techniques can be performed on the data for more knowledge and patterns.

1.1 Introduction to Cab Services

Data mining can be used in various ways in the real world scenarios. With the rapid advancements in the field of IT, availability of wireless technology, mobile networks, GPS technology and many more, there was a need to devise ways which could help minimize the traffic congestion on roads in metro cities. Since, metro cities experience long duration and distance traffic jams and overloaded public transport, the passengers needed to have a hassle free experience with affordable prices. Also, people needed to travel long distances on day to day basis. Due to uneven landscape distribution of the city, both public and private vehicles were chosen for travelling. However, public transport alone could not meet the ever increasing demands of passengers, cab providers such as Ola and Uber emerged quickly in metro cities. Due to the ease of availability of cabs and point to point service, cab providers are generating huge revenues. With the benefit of affordable prices, more and more people opt for taking up cabs to work, party places, airport and many more.

Due to availability of all types of cabs ranging from hatchback to SUVs, all the sections of society are equally served. Cab providers aim to provide services to all irrespective of the section of the society to which the passenger belongs to, for better consumer experience. Only targeting the high class society cannot help the cab providers reach the expected profit and revenue. For people not willing to spend too much on cabs, shared rides are available with prices competitive with prices of public transport. This type of carpooling scheme also helps reduce the traffic congestion on the roads which these days is the major

concern in metro cities. Additionally, the safety, reliability and cost effectiveness of cabs make everyone travel in cabs without any concern, everything being made transparent and available on the application itself.

Travelling with GPS equipped cabs or using mobile GPS to detect the location of the cabs and keep track of the cab is one aspect which is most useful for all the passengers. Rather than contacting the driver for the location, the rider can easily track the cab. Many other functionalities such as recommender systems are employed in the software which can analyze the trips taken by the person in past during the same time interval. More and more analytics in this field can further help the drivers gain knowledge about the availability of passengers in the different areas of the city, detect shortest route to reach a location, avoid routes with high traffic congestion to avoid delay and other such functionalities. Still there is a lot of scope for research in this field which can improve the application in so many more ways to benefit both the companies and the passengers.

1.2 Taxi Imbalance Problem

Most of the cab providers, with little knowledge about the potential areas for pickups, deploy cabs randomly. Knowledge about such areas can help drivers enhance the driving directions. Most of the times a shortage of cabs is observed during the peak hours whereas the cabs roam around empty across the city in search of customers [1]. This problem, also known as the taxi imbalance problem [2] is a major issue which needs to be worked upon. This scenario leads to the loss of customers which in turn lowers the revenue generated from the customers. Hence, the knowledge of potential customers in different areas during different time of the day and day of the week becomes important for cab providers to deploy the cabs at high functional regions. This way cab providers can set up the taxi base near the regions which are observed to have high number of pickups. This approach can also help reduce the traffic congestion and provide driving directions to the drivers in search of customers.

To deal with the taxi imbalance problem, hotspot detection seems to be a likely approach to identify social functional regions [3] around the city. Many papers have aimed to tackle this taxi imbalance problem in different ways like hotspot detection, predicting taxi demand, recommending road clusters, analyzing taxi trajectories and many more. All the mentioned approaches aim primarily at minimizing the loss of customers during both the peak and non-peak hours. Also, reducing the taxi imbalance problem helps obtain larger customer satisfaction due to immediate availability of cabs and lesser wait

time. Many clustering approaches like multi-level clustering [4], density peaks clustering [5] have been followed to address this issue.

1.3 Supervised and Unsupervised Learning

In data related work especially machine learning, supervised and unsupervised learning play a very important role in prediction and analysis of various kinds of data for knowledge extraction. For using the past data for future prediction and forecasting, appropriate methods are needed to be used for better efficiency and obtaining robust results. Hence, depending on the type of dataset to be used for prediction, method needs to be devised for this purpose.

1.3.1 Supervised Learning

This machine learning method is used for dataset with labeled data. The data is split into two partitions namely - training data and testing data. Training data composed of training tuples from the original dataset. The datatset consists of a target variable to be predicted using the supervised learning methods. Hence, an input is provided which results in an output vector. During data training in supervised learning methods, a function is generated using different machine learning models for the training data which is further tested on the testing data. The testing data already consists of pre-labeled data, and based on the prediction from the model, it is analysed how correctly the model is able to correctly predict the labels of tuples.

Methods like classification and regression fall under supervised learning category. The datasets used for this technique consists of a target variable, which consists of different pre-defined classes. This data is used to train various models like SVM, neural network, decision tress, etc. to be used for further prediction of new and unseen data. A model which provides high accuracy on the testing data is only able to provide correct prediction results for the new data.

1.3.2 Unsupervised Learning

This type of learning mainly deals with unlabeled data. Since no labels are present corresponding to the tuples, hidden structures are extracted out of the data. There is no concept of separate training and testing data in this type of learning since the absence

of labels in the data does not lead to usage of target variable for finding the accuracy. Hence, instead of accuracy, this type of learning method uses different parameters such as distance, density, etc. for evaluation purpose. This also considers the statistical properties of the data and infers different results from them.

Methods like clustering can be categorized under the unsupervised learning since there are no labels used for clustering. The different parameters used for clustering are mainly distance, similarity and dissimilarity matrices, density functions and many more. Based on any of the given parameters, multiple partitions within the dataset are created which help visualize the diversity in the given dataset.

1.4 Data Collection and Pre-Processing

- **Historical Data Collection** : To identify the density regions for better availability of cabs, the historical data needs to be collected and analyzed efficiently. The historical data collected should be huge enough to present results that are in line with the actual results. The data collected can either be in a specific geographic location or spread across the landscape. It is preferred to have region specific data to provide better insight into the peak timings during both weekdays and weekends. Since the cab demand varies during different time of the day and day of the week, time specific and day specific analysis needs to be done in order to achieve better results which can be further used on real time situations. This type of data mainly contains all the pickup and drop off related information which can be used for trip time estimation, carpooling strategies, hotspot detection, informed driving, etc.

- **Data Integration** : There may be scenarios where the data obtained is raw and needs integration and cleansing before it can be used for analysis. The data obtained in raw format is not of much importance until meaningful patterns and knowledge is extracted from it. If city-specific data needs to be analyzed, then data from different regions of the city needs to be collected and a single file needs to be created which can be used for analysis. There may be cases where the amount of data becomes so large that it becomes impossible to analyze the data until the concepts of Big Data come into picture. This huge amount of data cannot be handled by ordinary systems and may fail to provide accurate results. Additionally, the data from different months needs to be integrated so as to take a range of possible scenarios for prediction. Taking up only months' data fails to provide diversity in the pattern of pickups and drop offs. Hence, larger the diversity of pickups in different regions

at different time intervals, higher is the possibility to obtain consistent and more generalized results.

- **Feature Selection** : After the historical data collection, there are cases where the number of features are so large that the irrelevant features might lead to diverted results. Hence, feature selection in data mining process becomes an important technique to obtain accurate results. A number of feature selection techniques are available which provide the most relevant features removing the irrelevant ones. Feature selection in this case is limited to the knowledge of pickup location. Hence, there need not be any feature selection technique applied on the data due to less number of features in the dataset already. The fields providing the knowledge of the pickup location and pickup time are only retained. All other fields are removed which provide little or no help in the analysis. However, removing irrelevant features leads to lesser running time of the algorithm and helps interpret the results in a better way.

- **Data Transformation** : Data transformation becomes an important step in data analysis of large scale data. The values of the features present in the raw data might not help provide any insight into the data. Hence, meaningful transformation needs to be done on various attributes and fields for analysis of the data. Converting data in numeric format, allocating rows to predefined classes, converting date and time into a consistent format are examples which are most commonly used at the time of data transformation. Transformation can have different objectives or goals such as to convert data into consistent format, reduce the dimensionality of the data, etc. Hence, the objective of the transformation needs to be defined clearly before using specific techniques.

1.5 Outlier Detection

Outliers are the data points which are highly deviated and different from other observations resulting in inefficient results. These data points do not follow the standard expected pattern. Outlier detection [6], [7] is a major technique used for different real world problems such as intrusion detection, system health monitoring, fraud detection in order to identify abnormal patterns observed in a given activity. These outliers are also treated as noise in the data which deviates from the final results. Both the local and global outliers are to be treated prior to any other process. Hence, outlier detection and removal is considered before any technique for knowledge discovery is implemented on

the data.

Following three type of outlier detection techniques can be applied on the given dataset:

- Unsupervised Anomaly Detection : This technique can be applied on the unlabelled data under the assumption that the dataset under consideration is normal.
- Supervised Anomaly Detection : This technique demands that the dataset under consideration should be labeled with values normal and abnormal for the classifier to be used for detection.
- Semi-Supervised Anomaly Detection : In this technique, a normal training set is considered for constructing a model and then testing the generated model.

1.5.1 Turkey's Method of Outlier Detection

This thesis primarily uses the Turkey's algorithm to identify and remove or treat the outliers present in the data in order to avoid inefficient results. This method exploits the inter-quartile range to detect outliers in the dataset. This method is highly insensitive to the statistical parameters such as mean and standard deviation, hence the dataset is not influenced by the extreme values present in the dataset.

The acceptable values only fall within the acceptable values of the upper threshold and the lower threshold values which can be calculated as follows:

$$Upperthreshold = Q3 + (Q3 - Q1) \times factor \tag{1.1}$$

$$Lowerthreshold = Q1 - (Q3 - Q1) \times factor \tag{1.2}$$

where Q1 and Q3 are the first and third quartile respectively and the standard value for factor is taken to be 1.5.

1.5.2 Outlier Removal

There are different ways of treating the outliers [8], [9] present in the data. The selection of the treatment depends on the effect and influence it has on the end results of the data. Hence, outliers can be treated in the following ways.

- **Imputation Using Mean** : The outliers detected as a result of the Turkey's algorithm are replaced with the mean of the entire dataset.
- **Imputation Using Median** : The outliers detected as a result of the Turkey's algorithm are replaced with the median of the dataset.
- **Imputation with NA** : All the outliers in the dataset can simply be ignored and substituted with NA such it does not have any effect on the dataset.

1.6 Spatial Clustering

Spatial clustering aims at grouping data objects into groups or clusters such that the objects within a cluster have high similarity among them whereas the objects belonging to different clusters are highly dissimilar in nature. Since the data is not labeled, this technique is classified as unsupervised learning with no separate training and testing dataset. It does not use any pre-defined classes for data grouping. A number of real world scenarios use spatial clustering such as detecting areas with similar land usage and distribution, and grouping areas with similar weather conditions. The type of clustering technique highly depends on various factors such as application goal, types of data attributes, dimensionality and amount of noise in data.

Clustering is used for grouping data to maximize the inter-cluster distances and minimize the intra-cluster distances. Various types of clustering algorithms [10] such as PAM (Partitioning Around Medoids), hierarchical clustering, density based clustering, grid based clustering, etc. can be used. Distance between various observation points is treated as a measure for allocation of points to clusters and to analyze the cluster for its goodness. A number of distance measures are available such as Euclidean distance, Manhattan distance, Mahalanobis distance, etc. which can be used depending upon the type of dataset with which the distance is to be associated. An observation point with minimum distance to a cluster center is assigned to that cluster. This process is iterated until no more reallocation is performed in the dataset.

Clustering techniques can be split into the following categories:

- **Partitioned Clustering** : This technique decomposes the observation points into disjoint sets. It uses an objective function to be used over multiple iterations until no more reassignment of observation points is done. The objective function is optimized over each iteration which might aim at providing a locally optimal or globally optimal solution. Locally optimal solutions have high intra-cluster similarity but

cannot promise low inter-cluster similarity whereas globally optimal solution aim at acheiving both.

- **Hierarchical Clustering** : This type of clustering either takes a large dataset and keeps decomposing it till the point where no more reassignment is performed or takes small clusters and keeps grouping them based on similarity. A hierarchical tree based clusters also known as a dendogram is obtained. The lowest level of the dendogram provides the required cluster solutions for the given dataset. To obtain solution with specified number of clusters, dendogram can also be cut at a certain required level to obtain specified number of clusters.

- **Density Based Clustering** : This clustering technique groups observation points based on the neighboring density of points. Cluster centers are randomly selected and based on the density of surrounding areas, observation points in that region are assigned to the closest cluster center. This clustering technique requires large memory for execution, hence, this technique under performs for large datasets and is unable to provide accurate results or no results at all.

- **Grid Based Clustering** : Spatial data mining is the primary application of this type of clustering technique. The spatial points are first quantized into several regions or specified number of cells and all the operations are then performed on this finite number of cells.

1.6.1 Partitioning Methods

For a specified number of clusters k, the dataset in these methods are partitioned into k distinct and disjoint groups termed as clusters. The assignment of observation point to the clusters is performed in such a way that the deviation of each object from the cluster is minimized. The deviation in most cases is computed using the similarity matrix between the objects. The primary aim is to have objects with high similarity within cluster and low similarity between any pair of clusters.

Three types of partitioning algorithms namely Expectation Maximization (EM), k-means and k-medoids algorithm come under this category. Different objective functions are followed in each of the method which is computed over multiple iterations till the cluster reassignment is over. The k-means and k-medoids algorithms form clusters based on the centroid/mean of the objects and most centrally positioned object respectively. EM works in a different manner than the other two as it finds the clusters using distribution of mean. Rather than using a single parameter for cluster assignment, this technique

uses probability based methods for assigning the objects to cluster which has highest probability of belongingness.

1.6.1.1 K-Means

K-means [MacQueen, 1967] is the most commonly used type of partitioning clustering technique. This clustering technique, as mentioned in the section 1.6.1 requires the number of clusters k, to be defined in advance. Post finding the optimal number of clusters to be formed, the mean of all the observation points is used to find the cluster centers. Similar to other PAM clustering, this method also optimizes the objective function given in 1.3 to form clusters using the Euclidean distance measure.

$$E = \sum_{i=1}^{c} \sum_{x \epsilon C_i} d(x, m_i) \tag{1.3}$$

Here, $d(x, m_i)$ depicts the Euclidean distance measure between the observation point x and m_i, where m_i represents the cluster center for C_i. Hence, the given objective function calculates the distance of each observation point from the center of the cluster and tries to minimize the value. Hence, this objective function is minimized in order to obtain the most optimal cluster for the given dataset. A set of cluster centers is chosen randomly in the beginning, then over each iteration, the observation points belonging to the nearest cluster center are assigned and the cluster center is re-evaluated at the end of each iteration to obtain the new cluster center of the newly formed cluster. This process is iterated till no more reassignment is performed.

This method proves to be highly efficient and scalable when dealing with large datasets since the complexity of this algorithm turns out to be $O(nkt)$ where n is the total number of observation points, k is the number of clusters to be formed and t is the number of iterations required for cluster formation. Being highly sensitive to noise and outliers, this method often terminates at locally optimal solution and fails to provide globally optimal solution.

1.6.1.2 K-Median

Unlike k-means, this clustering method uses the most centrally positioned observation point in the cluster rather than using the mean value of all the objects in the cluster, which makes this method less sensitive to noise and outliers.

Similar to k-means, this methods also requires the number of clusters to be formed prior to initialization and uses a distance measure for cluster formation. Hence, the cluster centers are again randomly selected and then updated over each iteration.

Many variants of this method such as PAM (Partitioning Around Medoids) are also implemented to provide better clustering results. In such methods, for each iteration, if the value of objective function decreases, then the cluster centers are replaced by other *(n-k)* objects in the dataset. This repeated computation makes it an expensive approach to be used with $O(k(n-k)^2)$ complexity for a single iteration. Hence, with such complexity, computation for large datasets becomes very expensive.

Hence, to deal with large datasets, CLARA (Clustering Large Applications) and CLARANS (Clustering Large Applications based on Randomized Search) are implemented which work on the sampling based techniques. In these techniques, rather than using the entire dataset, a small portion of the dataset is used as the representative of the whole dataset. Medoids from this sample dataset are then chosen to find the medoids using PAM and average dissimilarity is computed for the entire dataset. Any set of medoids with different sample data giving a lower dissimilarity score is then selected as the best medoids. Hence, in this technique, multiple samples are evaluated and the sample providing the best solution is selected for cluster formation.

Further, to improve the effectiveness, quality and scalability of CLARA, a variant of CLARA known as CLARANS was proposed since CLARA might not always fetch the best k medoids from the sample dataset. CLARANS finds a better solution by selecting the k centers and then randomly replacing them with another $(n-k)$ objects. It gives a computational complexity of $O(n^2)$ where n is the number of observation points.

1.6.1.3 Expectation Maximization

This method works on the probability model by representing each cluster with a probability distribution. In most cases, the Gaussian distribution is assumed to be followed by each cluster. A cluster is assumed to follow a d - dimensional Gaussian distribution with the mean μ_i and a $d \times d$ covariance matrix represented by M_i. Hence, the probability of an object placed at position x is represented by the probability function given in equation 1.4.

$$P(x|i) = \frac{1}{\sqrt{(2\pi)^2|M_i|}} e^{(\frac{1}{2}(x-\mu_i)^T M_i^- 1(x-\mu_i))} \tag{1.4}$$

The probability density function is calculated in the equation 1.5:

$$P(x) = \sum_{i=1}^{k} W_i P(x|i) \tag{1.5}$$

The objective function for expectation maximization is given in the equation 1.6:

$$E = \sum_{x \epsilon D} log(P(x)) \tag{1.6}$$

When the expectation maximization value between two consecutive iterations is negligible, then an efficient cluster formation is achieved and the algorithm can hence be terminated.

1.6.2 Hierarchical Methods

These methods hierarchically decompose the given dataset into a tree like structure also called as dendogram. This dendogram can be created using two different approaches agglomerative or bottom-up approach and divisive or top-down approach.

- **Agglomerative** - In this technique, initially, each data point is treated as a single cluster or a group with a single element. Based upon the selected distance measure, grouping is performed to merge the elements which are close to each other and computing the cluster center for the formed group. The algorithm terminates only when all the data points are merged to form a single cluster.
- **Divisive** - This technique works in opposite manner as that of agglomerative approach. All the data points initially are assumed to belong to a single cluster. Again, depending on the selected distance measure, the data points farthest from each other are split into two separate groups. This process is repeated till each observation point belongs to a separate cluster.

1.6.2.1 AGNES

AGNES or Agglomerative Nesting method follows a bottom-up approach which iteratively results in a single cluster at the end. The steps followed in this approach are irreversible which can lead to error-prone results. To terminate the algorithm, a predefined number of clusters can be chosen on which the algorithm terminates. Distance measures such as

Euclidean, Mahalanobis, Manhattan can be computed for the merging of points into a single cluster.

1.6.2.2 DIANA

DIANA or Divisive Analysis is a top-down approach which works on splitting the dataset into separate clusters. This works in a similar manner to that of AGNES except the approach followed by this technique. This algorithm can also be terminated by specifying the required number of clusters for the algorithm to stop. Different distance measures can be used to measure the similarity or dissimilarity between two given observation points based on which the splitting is performed. Similar to AGNES, this technique is also irreversible and the next iteration will only be performed on the newly generated clusters.

1.6.2.3 BIRCH

The BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies) of clustering primarily aims at obtaining agglomerative clustering results and then using the outputs for further refinement over multiple iterations. Multiple small sub-clusters, each defined by a clustering feature is formed from the dataset to be used for further clustering; hence the database scanning is performed only once. The clustering feature used is given equation 1.7:

$$CF = (N, \vec{LS}, SS) \tag{1.7}$$

Where N depicts the number of points in each sub-cluster, SS depicts the sum of squares of data points, LS is the linear sum of N points.

All the clustering features are stored in the CF tree dynamically as new objects are added to the data. This type of clustering results in good quality clusters with high scalability for large data. Since this technique largely works on radius or diameter for calculation, it is not able to provide efficient results for non-spherical clusters.

1.6.3 Density Based Methods

The density based clustering considers the notion of density as a depiction of data points closely fitting together and surrounded by low density regions. The density based clus-

tering techniques are highly insensitive to outliers, which further helps data cleansing by filtering out the noise or outliers. The major types of density based clustering are DBSCAN, OPTICS, DENCLUE which are further described in later sections.

1.6.3.1 DBSCAN

DBSCAN or Density Based Clustering of Applications with Noise, works with the parameter $MinPts$ and e, to find if the number of data points in the surrounding areas within a distance e is larger than $MinPts$. The algorithm converts areas with high density (areas with large number of data points) into well-defined clusters of arbitrary shape as given in figure 1.2. A combination of areas within the distance e (called as e-neighborhood) and consisting of $MinPts$ (called as core object) is termed as a new cluster. Multiple iterations are performed for converging the neighboring points satisfying the specified criterion.

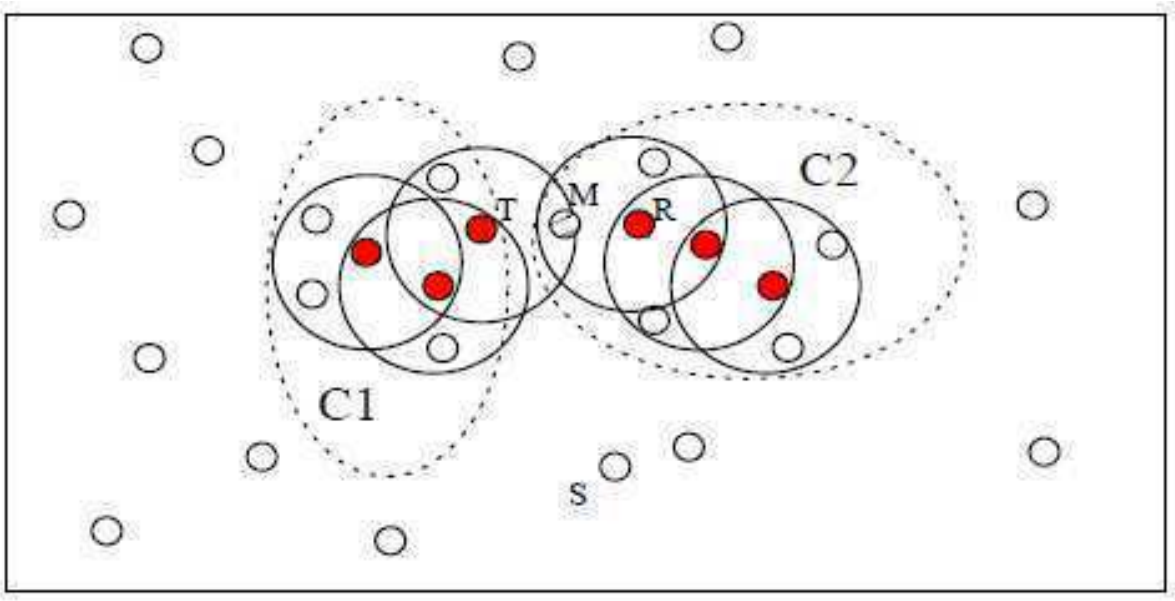


Figure 1.2: DBSCAN Clustering

The algorithm is terminated when no more core objects are discovered in the newly formed clusters and no data point is added to a cluster. The major drawback with this clustering technique is that it cannot be applied to large data sets since it works on a hit and trial method for selection of the required parameters. Hence, this clustering technique fails for large datasets.

1.6.3.2 OPTICS

OPTICS (Ordering Points to Identify Clustering Structure) unlike DBSCAN, does not work on selecting different values of the parameters e and MinPts using a hit and trial approach. This technique does not set the value of these parameters prior to using this approach, but orders the data points such that lower values of these parameters can result in lesser computational complexity. This algorithm also sets a threshold value of two different parameters core distance and reachability distance as given in figure 1.3.

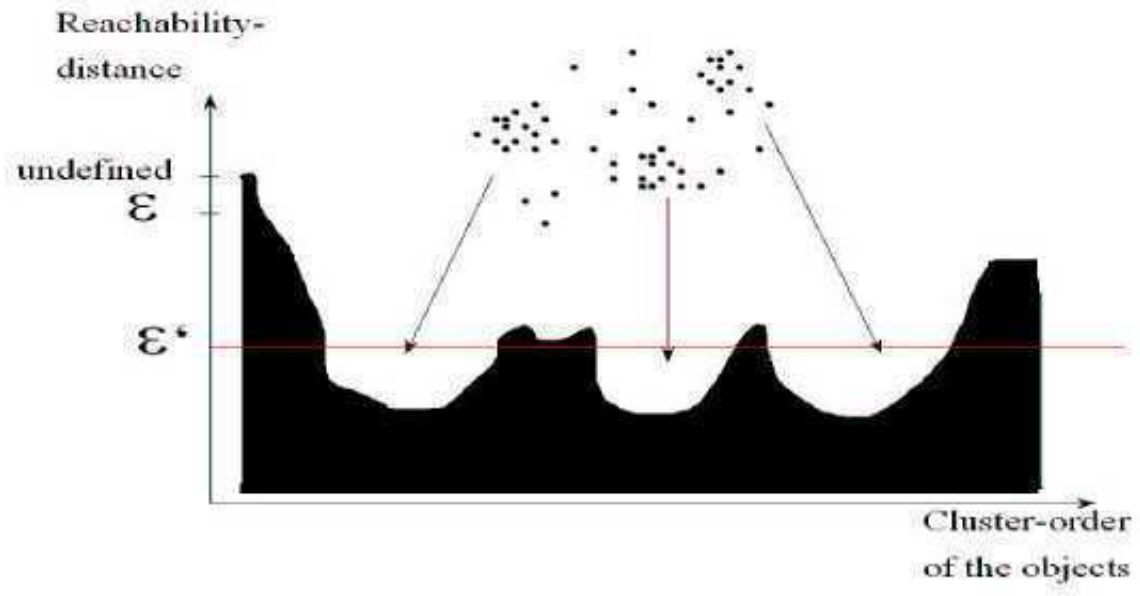


Figure 1.3: OPTICS Clustering

Hence, this algorithm is able to outperform the computational complexity in comparison to DBSCAN with a solution for handling noisy data and outliers in the dataset.

1.6.3.3 DENCLUE

The DENCLUE (Density Based Clustering) technique considers the density distribution functions for creating clusters. In this algorithm, an influence function which shows the impact of each data point on its neighbors is expressed as a mathematical function. The aggregation of the impact factors determined by each influence function provides the analytics of overall density distribution in the data space. The overall density function is further exploited to from clusters using the local maxima, also known as density attractors. This type of clustering can result in spherical, center-defined and arbitrary

cluster shapes. Hence, this clustering technique is highly efficient in forming good quality clusters.

1.6.4 Grid Based Methods

The grid based methods are able to remove the drawbacks of density based clustering techniques which becomes inefficient for datasets with high dimensions. These methods uses a grid data structure in which the space is broken down into tiny cells on which all the operations are performed. This methods result in high computation time independent of the number of objects in the dataset since it is only dependent on the number of cells in each dimension. The major types of grid based clustering methods are STING, WaveCluster and CLIQUE which are described in the below sections.

1.6.4.1 STING

STING or Statistical Information Grid is a clustering technique in which the spatial points are broken down into rectangular cells. At each iteration, the cells are further broken down to obtain a higher resolution to form a hierarchical structure. As the name suggests, the attributes and statistical information such as mean, maximum value, minimum value, etc. of each cell is stored which is further used for query processing.

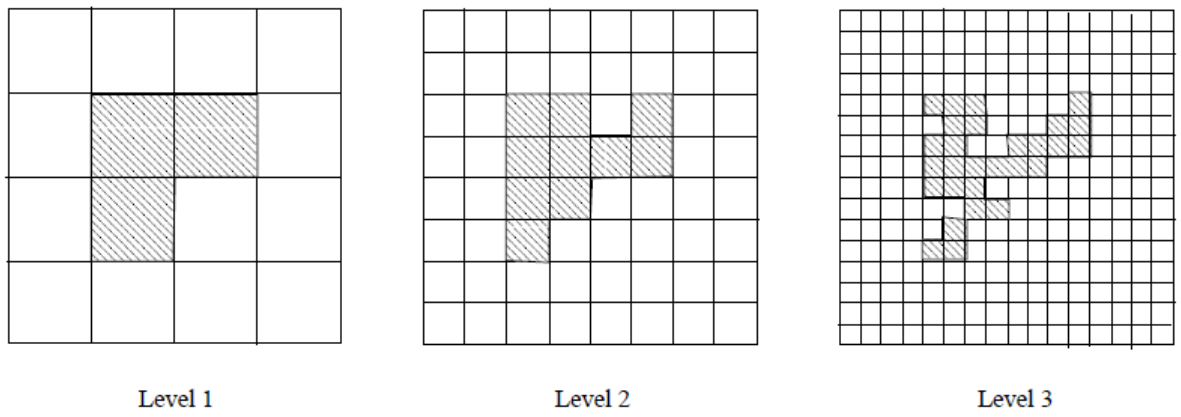


Figure 1.4: STING Clustering

The figure 1.4 shows three different levels of STING in which each higher level corresponds to higher number of cells in the spatial area. The level approach is used for computation of statistical parameters in which the parameter values at lower level are used for computation of statistical parameters at a higher level.

This method proves to be very efficient and provides a time complexity for generating clusters as $O(n)$, where n is the total number of objects whereas the query processing time taken is $O(g)$, where g is the number of grids/cells in the spatial area. This technique can also prove inefficient since it only considers vertical and horizontal boundaries and not diagonal boundaries.

1.6.4.2 WaveCluster

Clustering using wavelet transformation is one technique which summarizes the data in the spatial regions and then applies the wavelet transformation for finding dense regions in the transformed space. Hence, this technique serves both as a WaveCluster and a density based clustering method.

This technique generates different frequency sub-bands from the original signal. In terms of efficiency and quality of clustering, this technique outperforms BIRCH, CLARANS and DBSCAN and provides a computational complexity of $O(n)$.

1.6.4.3 CLIQUE

This algorithm also uses a combination of density based and grid based clustering. This technique is primarily used for clustering data in high dimensional space. The spatial area is first decomposed into non-overlapping rectangular cells. An input model parameter is provided to decide if a given cell is dense based on whether the number of data objects in the cells are greater than the input model parameter.

This technique also moves from a lower dimensional space to higher and uses dense unit obtained at lower levels to discover clustering solutions at higher levels. Multiple connected dense units are treated as a single cluster.

1.7 Cluster Validation

To identify whether the cluster formed as a result of a clustering algorithm is efficient or not, cluster validation techniques as in [3], [11] and [12] are required. A number of metrics are available which can be used to see whether the cluster formed is of good quality or not. These metrics include similarity matrices, inter-cluster distances, intra-cluster distances. The clustering algorithm providing the best value of these metrics is treated to be the best clustering algorithm for the given dataset, though different clustering techniques can

prove useful for different types of data. Internal cluster validation and external cluster validation measures aim at maximizing the intra-cluster distances and minimizing the inter-cluster distances respectively.

In this thesis, hotspot detection is done using clustering techniques, due to which quality of cluster plays an important role in providing accurate hotspots for informed driving. Since multiple clustering algorithms are implemented later in this thesis, there is a need to evaluate cluster indices to determine which clustering algorithm works best for a given type of dataset. These cluster indices can be split into two categories, internal quality measures and external quality measures discussed separately in sections 1.7.1 and 1.7.2 .

The major criterio used for cluster quality assessment are as follows:

- **Compactness** - This measure denotes how closely data points in the cluster fit together. This measure considers the variance in the cluster and lower variance is preferred for highly compact clusters.
- **Separation** - This measure evaluates the distance between different clusters. A good quality clustering results in large separation following the three approaches, single linkage, complete linkage and comparison of centroids.

1.7.1 Internal Quality Measures

These measures take into account the compactness of the clusters using different distance measures. The internal quality indices used for cluster assessment are given below.

1.7.1.1 Calinski Harabasz Index (CH Index)

This index also knows as *Variance Ratio Criterion (VRC)* takes the variance measure into account to assess the quality of a cluster. Since good clustering results in higher between cluster variance and lower within cluster variance, these standards must be followed. CH index can be calculated using the expression 1.8:

$$CHIndex = \frac{(n-k) \times \sum(diag(BCM))}{(k-1) \times \sum(diag(WCM))} \tag{1.8}$$

where k - Number of Clusters,
BCM - Between Cluster Means,
WCM - Within Cluster Covariance Matrix

Hence, according to the expression 1.8 , a higher VRC is required to obtain highly compact clusters which maximizes the between cluster mean and minimizes the within cluster means. This index is also used for selection of optimal number of clusters since the value of k with largest $CHindex$ is considered to be optimal.

This index provides best results if used with k-means with squared Euclidean distances and considers spherical clusters which are compact in the middle.

1.7.1.2 Davies Boludin Index (DB Index)

This index identifies clusters with high between cluster distances and are more compact in comparison to others. The average of all cluster similarity is taken for calculation of this index. The clustering solution with lowest DB index is considered to be the most efficient cluster.

1.7.1.3 Sum of Squared Error

The sum of squared error (SSE) is a measure of correctness in the results in comparison to the actual model. Smaller the value of SSE, better the data is fitting to the model. The value of SSE can be computed as given in equation 1.9:

$$SSE = \sum_{i=1}^{n} (y_i - f(x_i))^2 \tag{1.9}$$

1.7.2 External Quality Measures

The external quality assessment of clusters primarily uses the distance between clusters as the major criterion for evaluating the clustering results. The below given measures as per [13] are used for evaluation of clustering techniques used on the given dataset.

1.7.2.1 General Rand Index

This index measures the extent to which two given clusters are similar to each other. The value of this index ranges from 0 to 1. 0 indicates that the clusters are completely dissimilar to each other, whereas 1 indicates exactly same clustering solution obtained from different techniques applied on the same dataset.

1.7.2.2 Adjusted Rand Index

This measure is a corrected and improved form of General Rand Index where value of this index can also be negative unlike General Rand Index. If the value of Adjusted Rand Index is lesser than the expected value, then a negative value of this index is obtained.

1.7.2.3 Fowlkes and Mallows Index

This measure is used as an alternative to Rand Index. This evaluation metric also computes similarity between the clustering results. A higher value of this index indicates more similar clustering results. Since this metric considers the class labels prior to result evaluation, it also takes into account the four aspects, namely true positive, false positive, true negative and false negative.

1.7.2.4 Mirkin Metrics

For the computation of this metric, each partition is represented using the vector of Hamming distance between the clustering results. Null value is obtained for exactly similar clustering results whereas a positive value is obtained for dissimilar clusters. This metric can also be considered as alternative to Adjusted Rand Index.

1.8 Hotspot Detection

Different researchers have worked upon different ways to identify the social functional regions in a city. This hotspot detection can be done using taxi trajectory data, clustering techniques based on pickups, spatio-temporal data analysis, etc. Hotspot detection becomes an important step for analyzing areas which require higher cabs during a particular time interval. This time dependent knowledge about the hotspots can also help in reducing the taxi imbalance problem addressed in section 1.2. This can also provide driving directions to the drivers to aim for areas that experience higher number of pickups in the near future. This knowledge about the high density regions is important for business purposes and decision making for drivers.

1.9 Base Location Setup

In order to obtain better mapping of incoming request with the nearby hotspots and current location of the drivers, the taxi base should be setup near the areas which experience highest number of pickups at a given time interval. Knowledge about hotspots can further help implement a multi-layer clustering approach for enhancing location and time based service for better customer satisfaction.

1.10 Thesis Organization

The thesis is organized into 6 chapters. A brief outline is given below:

- **Chapter 1**: This chapter introduces the basic outline of the entire work structure of this thesis. An overview about all the aspects such as clustering, cluster validation techniques, cluster validity measures, hotspot detection etc. are provided in this chapter for a better understanding of the purpose of this thesis.
- **Chapter 2**: This chapter presents a survey of the literature work in the related area in order to identify the research gaps. These research gaps further help in formulation of the problem statement that is able to solve the real life problems faced in the related area.
- **Chapter 3**: Based on the research gaps identified in the section 2, a problem statement needs to be formulated which talks about the real life problem and how our research aims to resolve that problem. Primarily, the objective of this thesis is laid down to identify the purpose of the work done in this thesis.
- **Chapter 4**: Based on the problem statement formulated in chapter 3, this chapter deals with providing a methodology or a framework which can provide a solution to the above problem. A detailed workflow of the proposed solution is provided in this chapter to understand the novelty of the followed approach. Various techniques such as outlier detection, clustering, cluster validation metrics etc. are evaluated to provide the required results.
- **Chapter 5**: This chapter aims at providing a detailed knowledge about the technologies used for implementation of the proposed solution and results obtained from the implementation. These results are then analyzed to see which clustering algorithm works best for which kind of data. This analysis further helps to provide a

solution to the problem statement, i.e., to identify the hotspots and selection of base setup locations.

- **Chapter 6**: This chapter provides the conclusion of this thesis including results and observations. However, it includes drawbacks of the implemented work, if any alongwith the future work planned for further optimization of the results. This chapter gives a brief overview about the entire work done as a part of this thesis.

Chapter 2

Literature Review

2.1 Related Work

In paper [14], Kai Zhang et. al. propose a recommender system for driver based on demand. It makes use of historical taxi trajectories to predict passenger demand in various areas of the city. Spatio-temporal clustering [15] in combination with time series forecasting technique is used for this prediction. This approach also aims at predicting the demand and deploy the cabs at the predicted area with high demand in order to reduce the taxi imbalance problem. GPS trajectories of cabs are used to identify the passenger demand at location during that time interval. Hotspot identification is then performed to predict the hotness of demand at the obtained location during the obtained time interval. A combination of passenger demand and the hotspot are then recommended to the driver for the driver to start driving towards the location. This approach provides high efficiency in terms of recommendation with hit ratio of 79.6%.

The authors Kai Zhao et. al. in [16] devise a framework to predict taxi demand at high spatial resolution. Two real world datasets are used in this paper for analysis and prediction. This paper takes into account the human mobility to measure the demand uncertainty at a given location. Entropy and temporal correlation are two measures used to predict the demand uncertainty. Further, three predictive algorithms are implemented in this paper to see which technique proves best to provide most efficient results for the given dataset. Probability based Markov predictor, sequence based Lempel-Ziv-Welch predictor and machine learning based Neural Network predictor are the three approaches used in this paper.

The authors Neema Davis et. al. in [4] use a time series model to predict the taxi travel demand. A multi level clustering approach is used on the dataset obtained for the city of Bengaluru, India which helps resolve the problem of demand supply chain for cabs. To improve the efficiency of the technique, the demand over the neighboring cells/geohashes are aggregated to obtain the overall demand in that area. Hence, correlation between the adjacent geohashes are explored. The time series based prediction is then analyzed using

Mean Absolute Percentage Error (MAPE) as performance metrics. This approach results in 20% improvement in the results after incorporating the correlation information.

The authors Dongchang Liu et. al. [5] use a variant of Density Peaks Clustering (DPC) approach to predict the demand hotspots. The traditional DPC aims at generating clusters based on the density of observation points in an area and find the cluster centers of the generated clusters. Distances between all pairs of observation points needs to be calculated which is possible only for a small dataset. A dataset for large observation points result in a matrix which cannot be handled by ordinary systems. Hence, this paper aims at projecting all the density points on to a density image and apply the DPC variant on this density image. This approach results in much lesser execution time and lower memory consumption. Hence, the demand hotspots are detected using this DPC variant.

In this paper [17], the authors Luis et. al. introduce a framework for predicting the spatial distribution of taxi-passengers using streaming data. This paper also uses a time series based forecasting model to predict the demand. This technique helps in informed driving for the drivers to head towards demand hotspots. The GPS and event signals transmitted by the GPS equipped cabs were transformed into a histogram time series. Four types of models namely, Time Varying Poisson Model, Weighted Time Varying Poisson Model, ARIMA model and Sliding Window Ensemble Framework are used for prediction. The efficiency of this approach produces smart recommendations to the drivers.

In this paper, the authors Pengxiang et. al. [18] aim at discovering demand hotspots using taxi trajectory data. Trajectory clustering is the primary approach followed in this paper. A number of parameters such as number of clusters, cluster centers, cluster validation parameters etc. are selected. The proposed model is able to determine the above specified parameters. The dynamic nature of hotspots depending on the time and day are also explored using this approach. Since the trajectories are more dependent on the road networks available, network distance is considered as a more realistic approach rather than using Euclidean distance metric for evaluation and assignment of points to different clusters.

The paper [19] primarily aims at using taxi trajectory data to discover the social functional regions to depict human use of land in metro cities. The road network is first clustered in order to find the hot areas later. Real map is then used to identify the hot areas by splitting the graph based on area density. Social functional areas are then discovered in each hot area through classification.

The purpose of this paper [20] is to provide efficient clustering results after removal of

cluttering and occlusion due to overlapping points. Since overlapping and intersecting points can lead to significant loss and distortion of information, flow clusters considering both the origin and destination are taken into account rather than a single dimension. Flow clusters are created depending on the similarity of two flows from source to destination. Agglomerative approach is used as a base technique for flow clustering which considers a single parameter, number of nearest neighbors.

In this paper [21], the author has proposed a framework to identify road clusters using taxi trajectory data for highest passenger findings. The road network is first divided into much smaller road clusters. A feature set is then associated with each cluster to reflect the properties of the road cluster. ELM regression model is used for model training and testing after labelling the training clusters. A rank based approach is then followed to rank all the road clusters based on regression values. The top ranked clusters are then suggested to the drivers for driving to the location of the top clusters. This approach also aims at reducing the taxi imbalance problem by enhancing informed driving.

In this paper [22], a time dependent landmark graph is proposed to depict the dynamic nature of traffic patterns in the city. It also uses the drivers intelligence in the real world to enhance the driving directions from large taxi trajectories. A variance entropy based clustering approach is used to estimate time distribution between two locations at different time intervals. This way, the fastest route for a passenger is predicted depending on the pickup time and location. This type of approach is used to bypass the areas with traffic congestions and reduce the travel time by picking the most efficient route between two locations. The results show that in real scenarios, this approach on an average can save time up to 16% for a trip.

This paper [23] aims at providing driving routes to the drivers to maximize the profitability. Since driving without any information about potential customers can incur high loss to the companies, hence informed driving is the primary objective of this paper. The authors aim at proposing an objective function to evaluate the driving route taken by the driver before finding the passenger. Based on this evaluation, the driving routes are optimized and ranked. Highly ranked routes are then taken up by the drivers to find the passengers.

This paper [24] uses taxi data mining to be used for better public transportation system. Mining taxi dataset provides an insight into frequently visited places by passengers which can further help public transport planning accordingly. The leaks of coverage by public transport is identified using the taxi routes and patterns. The taxi data uses the GPS information, fare data information in both public and private transport to form clusters of regions and check the serviceability by public transport. Hence, this type of taxi data

mining can prove to be highly useful for better urban planning.

The authors Jin Liu et. al. [25] aim at using the cloud based taxi data for urban planning and smart city planning. Mainly, the GPS data of the taxi is used for mining the data and extract knowledge patterns out of this information. The SPARK based technique involves data preprocessing, trajectory pattern mining and trajectory clustering.

In this paper [26] , the passenger demands is inferred using Bayes formula based on the trip purposes. The uncertainty of individuals in a given region is measured using the data with high periodicity during the same time interval. The travel behaviors are inferred which help understand the mobility of humans in order to find the trip purposes. Also, the travel patterns of the past trips are also extracted using the taxi trajectory data.

The authors Feng Mao et. al.[27] in this paper use the GPS generated data from taxis and use it to explore travel patterns from the taxi trajectory data. Additionally, the approach uses the traffic grids to discover hotspots in order to understand the spatiotemporal distribution. Three major aspects of this approach are, clustering data based on pickup and drop off location, identifying threshold value for the clusters and visualizing analytics for better understanding of data distribution. This paper primarily focuses on identifying the residential locations using the mined patterns in the data and identifying job housing structures.

This paper [28] uses spatial and temporal data mining for developing recommendation system for GPS enabled taxis. Three traditional techniques, namely classification, clustering and regression are used for both passenger based and taxi based recommendation systems. The passenger based recommendation aims at providing waiting place recommendation to the passengers for early pickup whereas taxi based recommendation assumes different approaches such as hotspot recommendation, passenger finding facility, informed driving, etc. The above mentioned techniques are then used to explore the results of recommendations provided by different techniques.

This paper [29] uses historical trip data in order to extract similar trips in the past taken up by passengers to infer the taxi trip time. The unsupervised clustering technique is used based on the pickup location, drop off location and start time. This information is used to extract similar trips in the past to discover frequent sub-trajectories and time taken in the past trip. The primary measure used for performance testing is $RMSE$ and the results show that the $RMSE$ is reduced using the sub-trajectory clustering.

In this paper, Chengkun Liu et. al. [30] propose a framework for urban planning using location based service. Traffic congestion data is collected which is further used to infer the congestion duration and time based on the region of traffic congestion. Regions

experiencing high congestion are identified using DBSCAN clustering technique. Further, Ripley K function is used to define the global aggregation degrees. The interaction between the neighboring congested areas are also considered for better evaluation of clusters based on congestion.

In this paper [31] , PCA (Principal Component Analysis) and its variants are performed on the spatial data in order to discover spatial autocorrelation among objects. PCA associated with both the stationary and non-stationary objects are evaluated. This technique can be used for datasets containing spatial data to infer knowledge patterns out of this raw data. The results show that the computations proposed in this paper are a lot faster than the traditional methods for spatial data analysis.

The table given in 2.1 and **??** summarizes the objectives, techniques used and outcomes of the related work based on taxi trips data and taxi trajectory data separately.

Table 2.1: Summary of Related Work on Taxi Trips Data

Studies	Dataset	Objective	Techniques Used	Outcome
Kai Zhao et. al. [16]	NYC Yellow Cabs and Uber Taxi Trips Dataset	Taxi Demand Prediction	Markov, Lempel-Ziv-Welch, Neural Network Predictor	Neural Network performs better for areas with low predictabiliy and Markov Predictor performs better for areas with high predictability
Neema Davis et. al. [4]	Bengaluru Taxi Trips Dataset	Taxi Demand Prediction based on Human Mobility	Multi-Level clustering and Time Series Modelling on geohashes	Correlation based prediction improves results by 20%
Dongchang Liu et. al. [5]	Singapore Taxi Trips Dataset	Demand Hotspot Prediction	Density Peaks Clustering on Density Image	Lesser Execution time and low memory consumption
Luis et. al. [17]	Portigual Taxi Trips Dataset	Taxi Demand and Demand Hotspot Prediction	Time Varying Poisson Model, Weighted Time Varying Poisson Model, ARIMA model and Sliding Window Ensemble Framework	High prediction accuracy with error rate lower than 26%

Table 2.2: Summary of Related Work on Taxi Trajectory Data

Studies	Dataset	Objective	Techniques Used	Outcome
Kai Zhang et. al. [14]	Car INC. Taxi Trajectory Data	Hotspot Predction	Spatio-Temporal Clustering and Time Series Modelling	Hotspot Recommendation with hit ratio 79.6 %
Pengxiang et. al. [18]	Wuhan Taxi Trajectory Dataset	Hotspot Detection	Decision Graph based Trajectory Clustering	Detect similarity and difference between hotspots on different days
Wang et. al. [21]	Nanjing Taxi trajectory Dataset	Road Cluster Recommendation to Drivers	ELM regression model for rank based approach	High recommendation accuracy
Yuan at. al. [22]	Taxi Trajectory Dataset	Driving Direction Enhancement using Driver's Intelligence	Variance-Entropy-Based Clustering and Two-Stage routing algorithm	70% faster traffic-based route suggestion
Qu et. al. [23]	San Francisco Taxi Trajectory Dataset	Profit-based rommender syster for drivers	Net profit Objective Function Evaluation and graph based recommendation	Efficient Driving Routes with high profit
Chuah et. al. [24]	Taxi Trajectory Dataset	Bus Routes Design and Optimization	DBSCAN clustering and cluster filtering	Improved routes for public transport
Gong et. al. [26]	Taxi Trajectory Dataset	Infer Trip Purposes using Trajectory Information	Uncertainty evaluation using Bayes' Formula and Monte-Carlo simulation	Better Knowledge extraction and discovery of trip purposes
Feng Mao et. al.[27]	Shanghai Taxi Trajectory Dataset	Demand Hotspot Prediction in urban areas	Spatial Clustering	Highly Efficient hotspot detection for smart city management
Yuanhang Hu et. al. [28]	Taxi Trajectory Dataset	Trajectory Data based recommendation system	classification, clustering and regression	Better models with refined granularity

Chapter 3

Problem Statement

3.1 Research Motivation

Taxi presence around all the major cities in the world has now become a very important part of the living in all the households. More and more people prefer to take a cab to reach from one location to another. With increasing traffic on the roads, people have preferred to take cabs to work, party places airports, etc. to experience a comfortable journey and easy travelling. However, with greater comfort, there are many issues faced by customers, drivers and owners with need to be tackled for better customer service and to fulfill immediate customer requirements.

As mentioned in above section, all the metro cities face high traffic during the peak hours which leads to high taxi demand during those peak hours whereas taxis roam around in the city empty during other time intervals. This problem is the major issue which affects both the customer service as well as the taxi owners. Due to cabs not being able to fulfill the customer demands during the peak hours, cab owners experience high loss in the revenue generated from the customer service. Additionally, when a customer requests a pickup from a given location, the cab owners are sometimes not able to dispatch the cabs in the required areas with high demand or even if the cabs are dispatched, the estimated waiting time increases so much that the customer is forced to cancel the cab. This again affects the cab owners as it affects the income from the customers.

Since the drivers do not have any knowledge about the potential places for finding the customers, they drive in random locations in search of passengers. This kind of uninformed driving can have large scale implications on the owners if the cabs are not able to serve the customers every now and then. Hence, knowledge about the potential places for drivers is necessary so as to serve maximum requests for pickups at a given location.

Apart from the cabs being unavailable during the peak hours, the decision of setting up the taxi base at random locations in the city is yet another problem due to which the taxi imbalance problem arises. Taxi owners who have setup their taxi bases at locations which do not experience high demand and do not get sufficient requests from the customers.

Also, they are unable to fulfill the requests where the passenger is far from their current location. This leads to high cancellation rate for the cabs and such cabs experiencing very low pickup rate as compared to other cabs which have their taxi base setup at hotspots.

These three above mentioned problems affect the cabs service to a huge extent. If handled properly, it can lead to better resource utilization and customer satisfaction.

3.2 Problem Statement

In this thesis, major focus is laid on resolving the above mentioned problems in section 3.1. These three problems can be summarized as follows:

- Taxi Imbalance Problem
- Uninformed Driving
- Wrong Taxi Base Setup

3.3 Research Objective

The above mentioned problems are addressed in this thesis for better taxi service organization which can benefit both the customers and the owners. Uber taxi dataset for NYC is considered in this thesis to find ways for resolving such problems. Hence, this thesis aims at providing the following solutions:

- **Location Based Service (LBS)** - This type of service as described in [3] aims at using the location of both the customer and the driver as a parameter for dispatching the nearest cab at the given location. The taxi base setup plays major role in enhancing the location based service in order to dispatch the cabs from the base nearest to the pickup location of the customer.
- **Time Based Service (TBS)** - Time based service is important to be considered since the requests vary during different time intervals of the day and day of the week. Peak hours for the weekdays and weekends are different, hence the number of pickups at a given location is also dependent on the time of the day and day of the week.

This location and time based service allows the taxi drivers to identify the hotspots during the given time intervals and drive in the direction where maximum pickup requests can be attained. This enhances the informed driving for the drivers and lowers the waiting time as well as cancellation rate for the cabs.

Chapter 4

Proposed Work

4.1 About the Dataset

The dataset used in this thesis is obtained from Kaggle which gives an insight into more than 20 million trips taken up by Uber and other for-hire-vehicles (FHV) in New York City in year 2014. In cities like NYC, it becomes very important to analyze the data and form business rules for higher profits and also target customer satisfaction.

Data for taxi pickups around the city is accumulated for different months which include the following fields:

- Pickup Date and Time
- Pickup Location in terms of Longitude and Latitude
- Base Number which performed the pickup

	Date.Time	Lat	Lon	Base
1	4/1/2014 0:11:00	40.769	-73.9549	B02512
2	4/1/2014 0:17:00	40.7267	-74.0345	B02512
3	4/1/2014 0:21:00	40.7316	-73.9873	B02512
4	4/1/2014 0:28:00	40.7588	-73.9776	B02512
5	4/1/2014 0:33:00	40.7594	-73.9722	B02512
6	4/1/2014 0:33:00	40.7383	-74.0403	B02512

Figure 4.1: Original DataSet

The primary attribute of use in this dataset is spatial data in terms of longitude and latitude as in figure 4.1. This gives an insight into the overall land usage and distribution in terms of taxi pickup to analyze the areas with maximum probability of finding a customer in given time interval. Such large number of trips requires a lot of analysis in order to be able to draw useful patterns from raw data for better decision making. The figure 4.2 given below shows the pickup points in NYC based on the longitude and latitude. This shows the heart of the city to be fully utilized with maximum number of

pickups being observed in those regions, whereas the outskirts depict quite low number of pickups and less land usage.

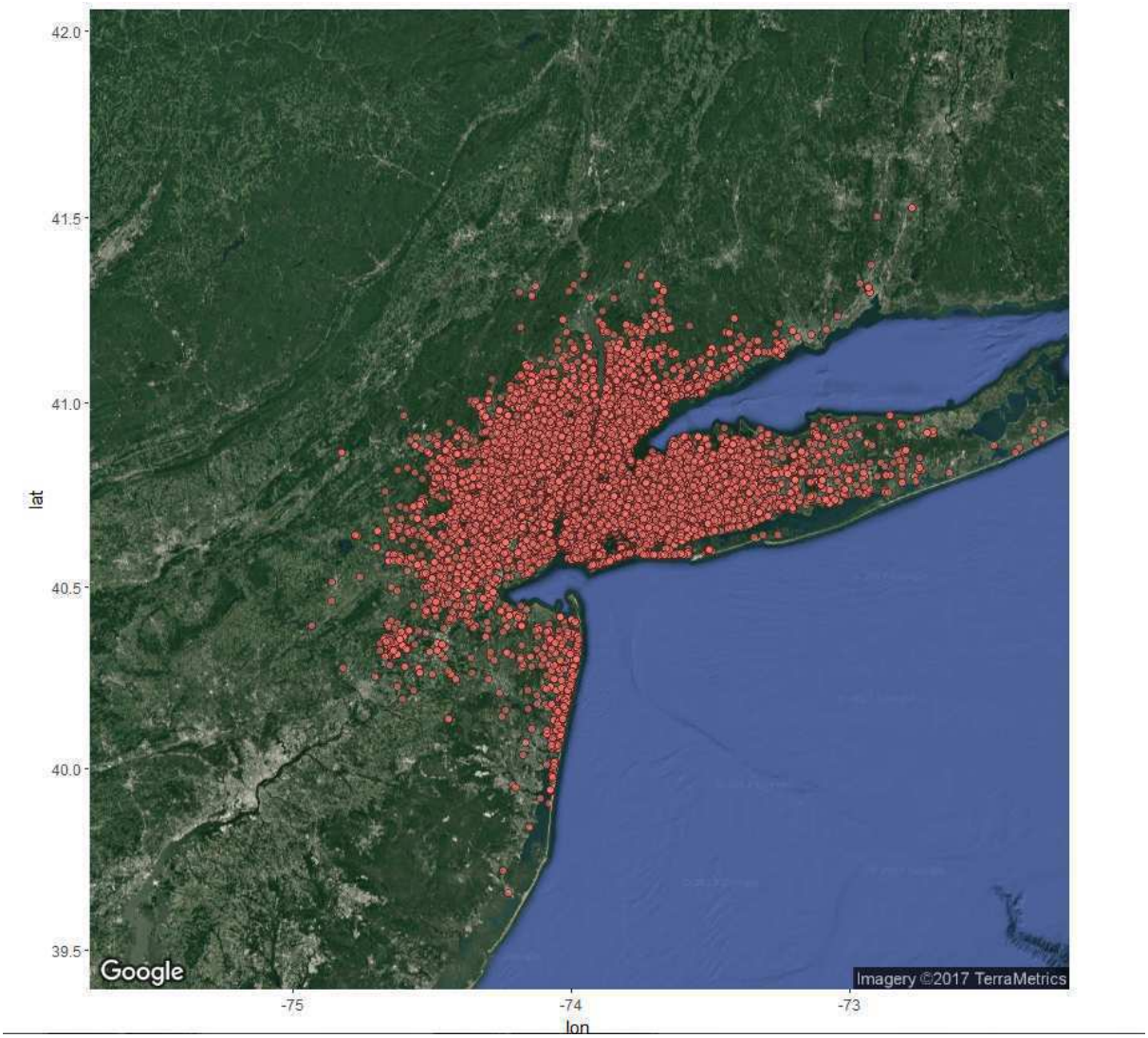


Figure 4.2: Pickup Locations

This gives rise to many questions such as:

- Identify areas which show maximum number of pickups during a given time interval
- Identify which bases are able to operate in most of the areas and which ones are region specific.
- Identify areas which need more taxi bases to be setup around them.
- Identify hotspots such as residential places, party places, office areas etc. depending on the time of pickup.

4.2 Proposed Model

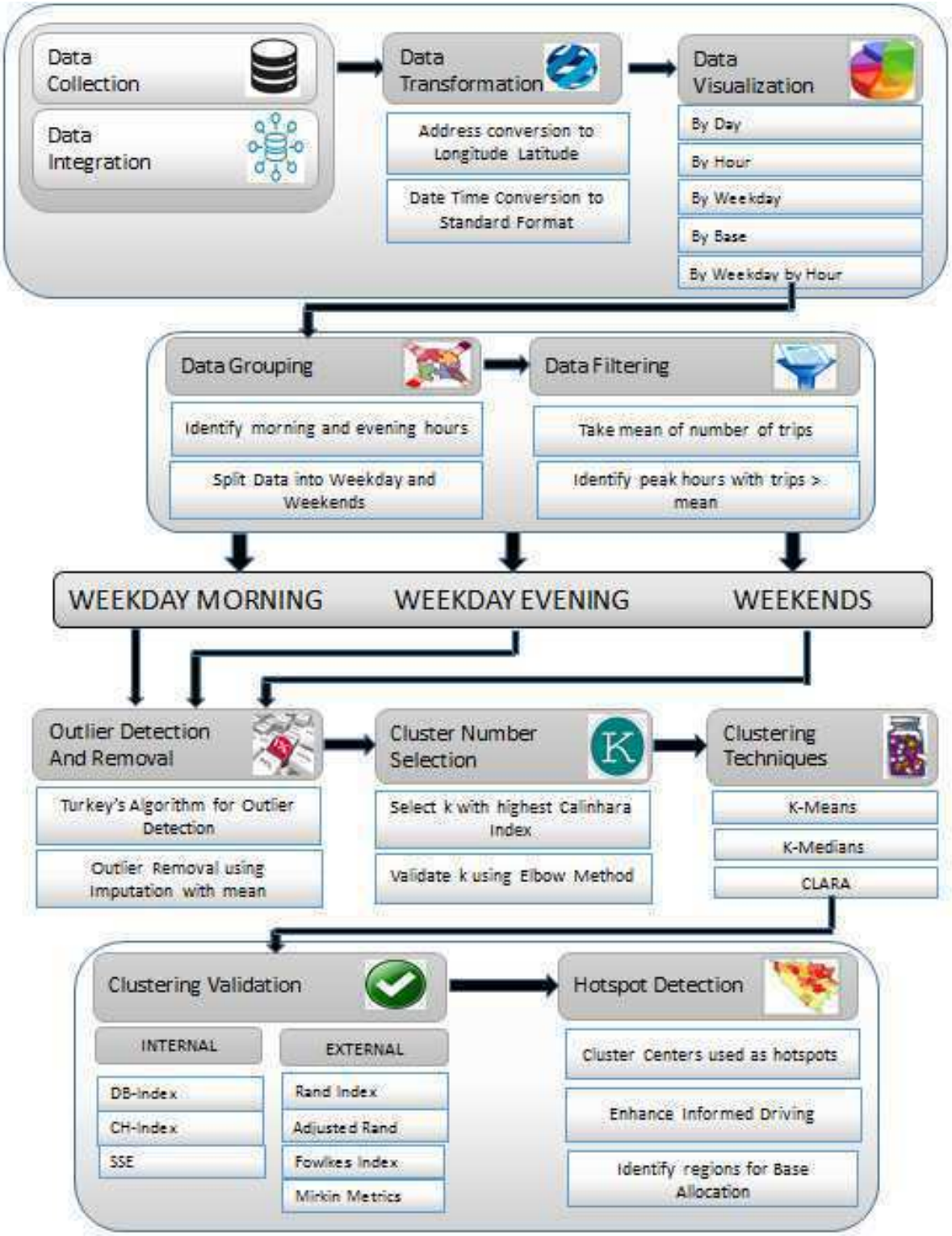


Figure 4.3: Proposed Model for Hotspot Detection

Pseudocode for Proposed Model

Input:

DataSet(uberApr, uberMay, uberJun, uberJul, uberAug, uberSep)
Where uberApr, uberMay, uberJun, uberJul, uberAug, uberSep are csv files for month of April, May, June, July, Aug, Sep

Address Conversion to Longitude Latitude:

1. Integrate by rbind(uberApr, uberMay, uberJun, uberJul, uberAug, uberSep)
2. Convert address field to character type using as.character() function
3. Extract longitude and latitude from address using geocode() function

DataSet Transformation:

4. Convert Date-Time to obtain consistent format using mdyhms() function
5. Extract day,month, year, weekday, hour, minute, second from date time field

Generating Day-Specific DataSets:

6. morning − Hour >= 6 A.M and <= 9 A.M and Weekday in (Mon,Tues,Wed, Thurs,Fri)
7. evening − Hour >= 4 P.M. and <= 8 P.M and Weekday in (Mon,Tues,Wed, Thurs,Fri)
8. weekEnd − Hour ! = 4, 5, 6, 7, 8, 9, 10, 11 A.M and Weekday in (Sat,Sun)

Outlier Detection and Removal:

9. distance − mahalanobis(dataFrame, mean, covariance)
10. outlier − boxplot.stat(distance)
11. distance − ifelse(distance in outlier, meanBeforeRemoval, distance)

Cluster Number Selection and Clustering:

12. **for** each i in range 8 to 20
13. CHindex − calinhara(dataFrame, cluster)
14. k − value of i with highest CHindex

Cluster Validation:

15. result − Clustering with highest CH-index, lowest DB-index and SSE

The proposed model given in figure 4.3 shows the entire stepwise procedure to be followed from data collection to hotspot detection. A detailed description of each step is given in section 4.5.

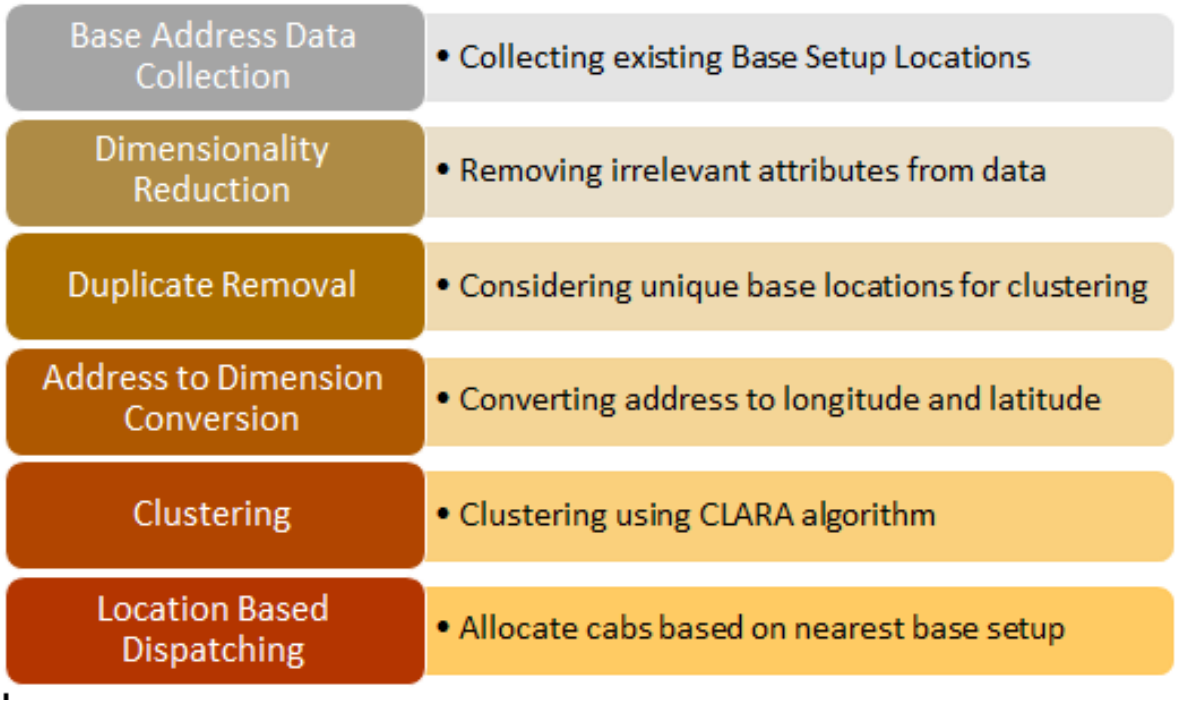


Figure 4.4: Proposed Model for Base Setup Location

The proposed model for base setup location given in figure 4.4 above is treated as the second layer of clustering in order to dispatch the nearest cabs to the incoming request in order to reduce the waiting time and cancellation rate. A multi-layer clustering approach is used with overlapping results obtained from the proposed model of hotspot detection and base setup locations. The methodology used in both the cases is more or less similar.

4.3 Data Integration and Transformation

Before the data is directly used, data integration and transformation are two major techniques used in order to make the dataset ready.

4.3.1 Data Integration

The data obtained from Kaggle consists of multiple csv files, each for months April to September, 2014. This data is present in raw format which cannot be directly used for knowledge extraction and finding patterns and trends in the data. Hence, this data needs to be integrated into a single csv file to be given as input in RStudio.

4.3.2 Data Transformation

In this dataset, there are fields which cannot be directly used, hence need some transformation to give insight into the data. Date and time provided in figure 4.1 are present in raw format in a single column which needs to be transformed. This data and time field is hence split [32] into multiple fields such as day, month, year, weekday, hour, minute, second. This transformation helps the field specific data to be abstracted out of the data with millions of records. Further analysis is done in later sections using this data since date and time is one major aspect of analyzing the pickups around the city.

	X	Date.Time	Lat	Lon	Base	Day	Month	Year	Weekday	Hour	Minute	Second
1	1	2014-04-01 00:11:00	40.769	-73.9549	B02512	1	Apr	2014	Tues	0	11	0
2	2	2014-04-01 00:17:00	40.7267	-74.0345	B02512	1	Apr	2014	Tues	0	17	0
3	3	2014-04-01 00:21:00	40.7316	-73.9873	B02512	1	Apr	2014	Tues	0	21	0
4	4	2014-04-01 00:28:00	40.7588	-73.9776	B02512	1	Apr	2014	Tues	0	28	0
5	5	2014-04-01 00:33:00	40.7594	-73.9722	B02512	1	Apr	2014	Tues	0	33	0
6	6	2014-04-01 00:33:00	40.7383	-74.0403	B02512	1	Apr	2014	Tues	0	33	0

Figure 4.5: Transformed DataSet

4.4 Data Visualization

Since the amount of pickups in a particular area is a clear indicator of the popularity of the place in comparison to the neighboring regions, it is important to filter data based on certain factors to achieve time or day specific popularity of a region. Initially, data visualization is performed to see monthly, hourly or weekly trends observed in the dataset. This visualization is important to select the parameters which highly affect the number of pickups in a particular region which are further used in hotspot analysis and detection.

The data visualization consists of the following components:

1. **Daily Trends** : The daily trends capture the total number of pickups in all the regions to see which set of days experience highest number of pickups. The pickups are shown in form of percentage of total trips taken on the specific days of the month.

2. **Hourly Trends** : This trend considers total number of pickups observed in each hour, each day of the month. The graph generated using the hourly data shows the total percentage of overall trips observed during a given hour.

3. **Weekly Trends** : The weekly trend shows the number of pickups observed on each day of the week and analyze which day is expected to experience the maximum number of pickups.

4. **Base Specific Trends** : The dataset obtained consists different taxi bases majorly operating in the city. Hence base specific pickups are analyzed to check the performance of taxi bases on different days.

5. **Demand Based Data Splitting and Grouping** : Using the above given data visualization and obtained graphs, data splitting and grouping is done in order to segregate the data for hotspot analysis.

6. **Morning Trip Trends** : The data given above is split into two categories, morning trips and evening trips. The hour range from 0 to 11 is taken for the morning trips for analysis and finding patterns in the data.

7. **Evening Trip Trends** : The second category considers evening trips with hourly range between 12 PM to 12 AM. The graphs are analyzed to find the hours undergoing maximum number of pickups on a given day.

The selection of peak timings further depends on the number of trips taken in each

category during a given hour. Since some statistical measure is required to select the hours experiencing highest number of pickups, the mean of trips is taken and hours with trips greater than the mean are considered as peak values in all the three categories.

4.5 Model Creation

4.5.1 Data Grouping

From the entire data obtained, the data needs to be grouped using [33] from better knowledge extraction based on the morning and evening hours and weekdays and weekends. This grouping provides more insight into the data to identify high pickups.

First the data is split into following three categories:

- Weekday Morning Data
- Weekday Evening Data
- Weekend Data

This data grouping is done in order to segregate the data and identify hotspots on weekdays and weekends since hotspots will be different in both the cases.

4.5.2 Data Filtering

The data grouped in the above section needs to be filtered to identify the peak hours which observe the maximum number of pickups. For this purpose, the mean of the number of trips is taken for each hour and the hours with the number of pickups higher than the mean is only selected to be potential candidates to be termed as hotspots.

The peak hours hence selected for each of the three categories are as follows:

- Weekday Morning Hours 6AM to 9AM
- Weekday Evening Hours 4PM to 7PM
- Weekends A rising trend is observed in the evening, hence all hours except 9AM to 4PM are excluded from the dataset under consideration.

This grouped and filtered data is then used as an input for the next steps on which the evaluation of hotspots is done.

4.5.3 Outlier Detection

The data obtained in the above step consists of noise and outliers which highly affect the results since they lead to diverted results. The presence of noise and outliers can lead to results which are not in-line with the actual scenarios observed in providing the cab services.

The outliers in this dataset are removed using the Turkey's algorithm which considers the boxplot statistics to identify the outliers and remove them. This method also considers the interquartile range for outlier detection and removal. All the data points outside the 1st and 3rd quartile are treated or removed.

4.5.4 Outlier Removal

The outliers detected using the Turkey's algorithm need to be treated such that it does not affect the final results. This provides more accurate results. The interquartile range is used to detect and remove the outliers. The outlier removal can be done using different methods such as imputation with mean, median or NA. In this scenario, imputation with the mean is used

For outlier removal, imputation [34] is one method that is used to treat the outliers. In this scenario, imputation of outliers is done with mean rather than removing those observations from the dataset since it provides more accurate results rather than removing them completely from the list to obtain all the centrally located pickup locations.

4.5.5 Cluster Number Selection

Since clustering is the primary hotspot detection technique used on this dataset, the number of clusters needs to be defined [35] in order to use partitioned clustering on the selected tuples. For partitioned clustering, the value of k needs to be pre-defined, hence, the Calinhara index is used for k selection. The Calinhara index is evaluated for different values of k and the value of k with the highest value of Calinhara index is selected as the optimal value.

This value of k is validated using Elbow method which aims as reducing the within cluster sum of squares. If the value of k selected from the Calinhara index results in an elbow in the graph for elbow method, then that value is selected.

4.5.6 Clustering

Partitioned Clustering algorithms [3] are applied on the dataset with treated outliers. Since the data is too large, the density based algorithms cannot be used for clustering and hotspot detection. The different types of partitioned clustering applied on the given dataset are as follows:

- K-Means
- K-Medians
- CLARA

4.5.7 Cluster Validation

To assess the quality of the clusters obtained as a result of a given clustering algorithm, both internal and external cluster quality measures are evaluated for comparison of clustering results for different dataset categories.

- **Internal Quality Measures** - The internal quality index aim at creating compact and dense clusters with low intra-cluster distances. The indexes considered are as follows:
  - DB-Index
  - CH-Index
  - SSE
- **External Quality Measures** - The external quality indices primarily aim at creation of highly separated clusters or clusters with high inter-cluster distances. The measures considered are mentioned below:
  - Rand Index
  - Adjusted Rand Index
  - Fowlkes Index
  - Mirkin Metrics

4.5.8 Hotspot Detection

The clusters obtained as a result of applying different clustering algorithms on given three datasets are considered for hotspot detection. The clustering algorithm providing the best results as per the metrics given above is considered and the cluster centers are then treated as the hotspots.

4.5.9 Base Address Allocation

Based on the incoming pickup request, cabs can be dispatched easily if the areas are partitioned according to the bases setup already by the cab owners. The new taxi bases can be setup depending on the area observing high demand during peak hours. Also, the demand based taxi base setup helps the cab bases undergoing minimum pickups around the city. This clustering is performed using the CLARA clustering.

This way, a multi-layer clustering approach is proposed where results from both the hotspot detection and taxi base clustering are used to allocate cabs around city during the peak hours such that no cabs roam around empty especially in the bases which are not able to serve sufficient number of requests.

Chapter 5

Simulation and Results

5.1 Research Methodology

The primary technology used for data analytics and visualization used in this thesis is performed in the language R which is used for statistical computations and visualization using graphs.

The platform used for implementation of this language is RStudio which is an open source development environment uses for statistical analysis of data. Various libraries depending on the type of application to be analyzed are needed to be installed and imported in RStudio in order to use it for implementation purpose. The table 5.1 and 5.2 given below provides the R libraries and functions used for implementation of the proposed model.

Table 5.1: Libraries and Functions Used Part 1

Library	Function	Description
ggmap	geocode(location)	This function is used to covert address to longitude and latitude.
	get_map(location)	This function is used to get the earth map to plot the pickup points on the geographic location.
dplyr	group_by	This function is used to group data based on a given variable and summarize the results.
Gmedian	KGmedian(data)	This function is used to evaluate the median to be used for k-median clustering.
MixSim	RandIndex(C1, C2)	This function is used to evaluate the general and adjuster rand index for cluster solutions.
ggplot2	aes(x,y)	This function is used to map the variable to visual properties in the plot.
	geom_bar(data)	This function is used to plot bar graphs on the given data.
	ggplot(data)	This function is used to initialize a ggplot object with aesthetics for plotting a data frame.
	ggtitle(value)	This function is used to display the chart title on the plot.

Table 5.2: Libraries and Functions Used Part 2

Library	Function	Description
In-built	rbind(D1, D2)	This function is used to bind the rows of two different data frames.
	cbind(D1, D2)	This function is used to bind the columns of two different data frames to form one single dataframe.
	cov(data)	This function is used to find the covariance of the given dataframe.
	mahalanobis(data)	This function is used to find the mahalanobis distance using the given observations.
	boxplot(data)	This function is used to obtain a boxplot using the given data.
	hist(data)	This function is used to plot a histogram of the given data.
	boxplot.stats(data)	This function is used to identify the outliers in the given dataset.
	kmeans(data)	This function is used to cluster data based on the k-means clustering algorithm.
lubridate	mdy_hms(date)	This function is used to convert the given date in the mdy hms format to make it consistent.
	day(date)	This function is used to extract day of the month from the transformed date.
	month(date)	This function is used to extract month of the year from the transformed date.
	year(date)	This function is used to extract year from the transformed date.
	wday(date)	This function is used to extract day of the week from the transformed date.
	hour(date)	This function is used to extract hour of the day from the transformed date.
	minute(date)	This function is used to extract minute of the hour from the transformed date.
	second(date)	This function is used to extract second from the transformed date.
ClusterSim	index.DB(cluster)	This function is used to evaluate the Davies Bouldin index for a clustering solution.
	calinhara(cluster)	This function is used to calculate the calinhara index for a computed clustering result.
	clara(dataFrame)	This function is used to cluster data using the CLARA clustering algorithm.

5.2 Data Visualization Results

5.2.1 Daily Trends

The figure 5.1 shows the variance seen for analysis. Since the day 31 is not present in all the months, this date is excluded from overall analysis of the data as it provides inconsistent results. Hence, from the graph it can be seen that the last day of the month is expected to experience the highest number of pickups considering the last day as the pay day for all the working class. Although there is not much of a difference that can be seen on other days to filter out the days for which maximum pickups are observed.

Hence, more permutations and combinations for date and time will be performed in the later sections to obtain significant results which show peak in the graph.

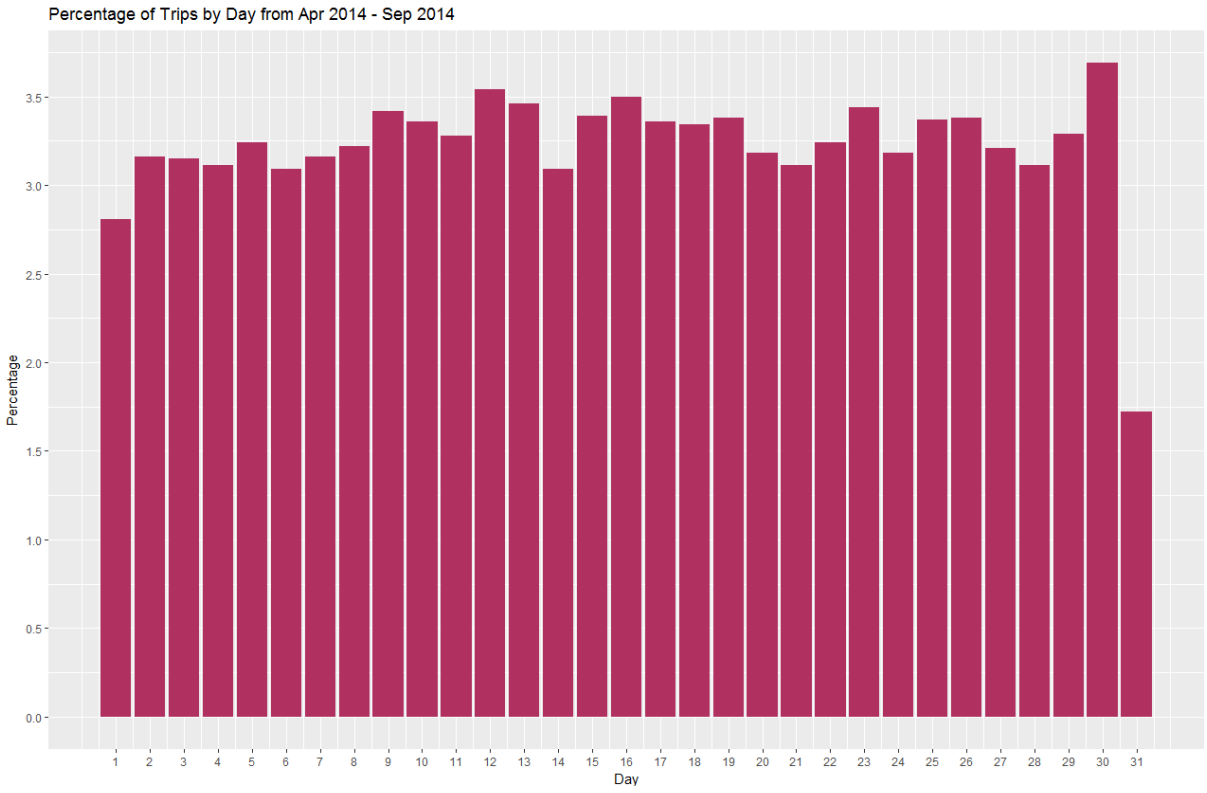


Figure 5.1: Trips By Day

5.2.2 Hourly Trends

It can be inferred from the graph in figure 5.2 that a peak is observed mostly during the morning and evening hours. Hence, considering the office timings and standard working hours in mind, these peaks in the graphs are justified. Since this graph considers all the days, the weekday specific data also needs to be considered to filter out the timings of highest demand.

For this purpose, the data needs to be split into weekdays and weekends to obtain the hotspots during the weekdays and during weekends considering that hotspots will definitely differ during working and non-working days. The split data will be able to better depict the hourly demand at different places.

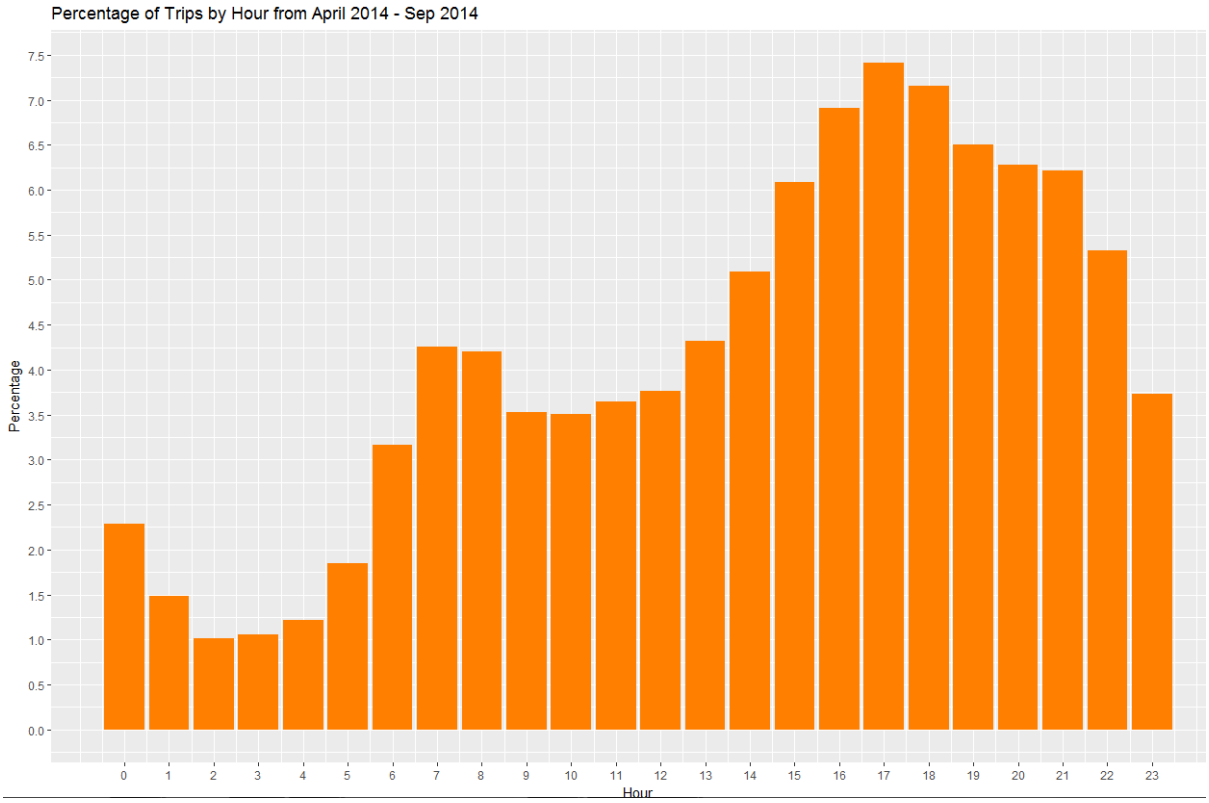


Figure 5.2: Trips By Hour

5.2.3 Weekly Trends

As seen in figure 5.2 , peaks are observed during the morning and evening hours. Considering this situation, the weekly data also needs to be included to see the trends during the weekdays and weekends as mentioned above.

The graph 5.3 given below shows maximum number of pickups observed on Sunday and Thursday. This might not give accurate and quantified results for hotspot detection and prediction. Hence, further a combination of hourly demand and weekly demand is considered in later section.

The hotspots for all the weekdays and weekends will be much similar with very low uncertainty. The hotspots during weekdays will mostly consist of office places and residential places whereas the hotspot during weekdays will mostly consist of party places and shopping places. This segregation of data into weekday and weekend will provide better statistics as per the number of pickups.

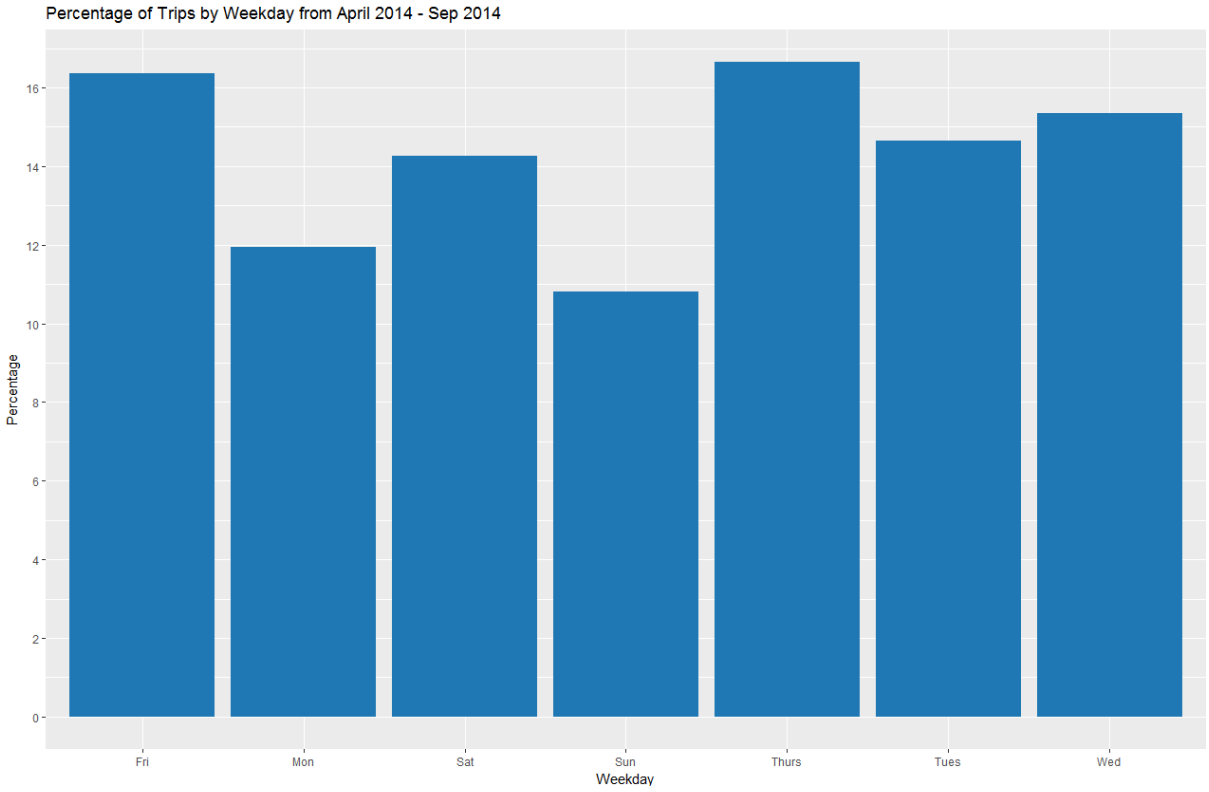


Figure 5.3: Trips By Weekday

5.2.4 Base Specific Trends

The data obtained from Kaggle considers five major base companies operating in NYC. Hence, the total trips taken by different bases are depicted in the graph 5.4 given below. The major operating bases are given below:

- B02512
- B02598
- B02617
- B02682
- B02764

It can be clearly seen from the graph that the number of trips taken by bases B02598, B02617, B02682 is considerably high. Hence, this statistics can be used for the other bases B02512 and B02764 to set up the taxi bases near the areas where maximum pickups are performed using hotspot detection.

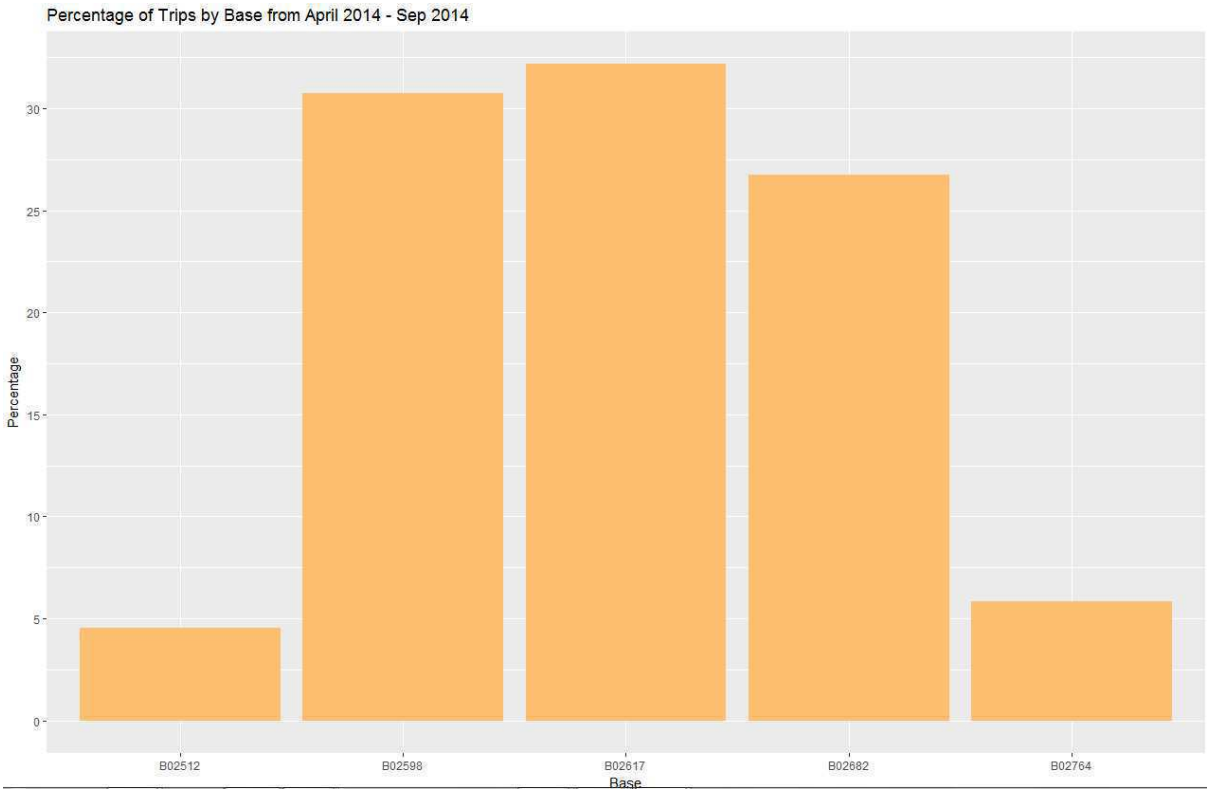


Figure 5.4: Trips By Base

5.2.5 Demand Based Data Splitting and Grouping

From the figures 5.1 to 5.4, it can be seen that the data needs to be grouped taking different permutations and combinations for better insight into the data. More significant results were obtained in the graphs where a large difference in number of trips was observed. Hence, grouping is performed first according to the Weekday and then according to the hour. This allows the hourly comparison of number of trips during different weekdays. The graph 5.5 given below shows the hourly data for both weekdays and weekends.

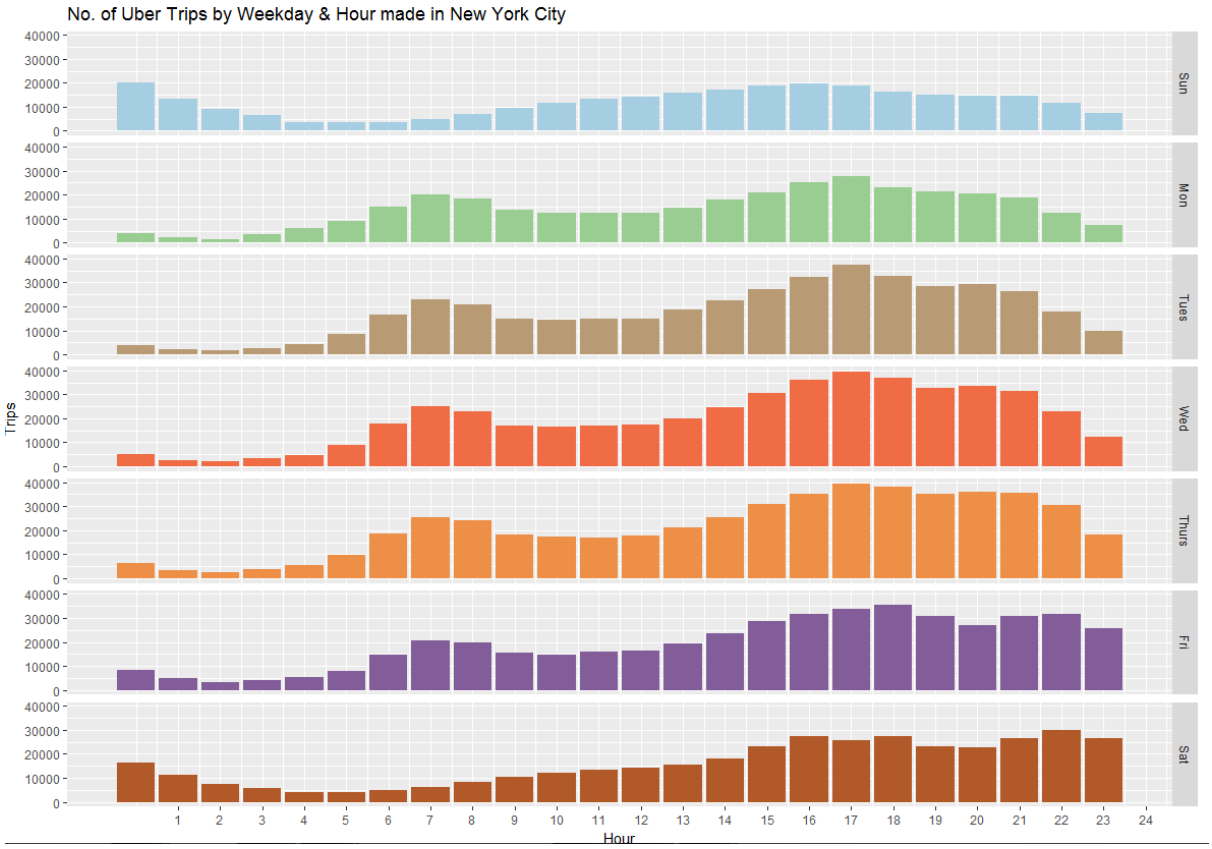


Figure 5.5: Hourly Trips by Weekday

The graph 5.5 shows hourly trends show peak during morning and evening hours which is further dependent on the weekday. Hence, the data is split into three broad categories as follows:

- Weekday Morning Trips
- Weekday Evening Trips
- Weekends

5.2.6 Morning Trip Trends

The graph given below shows hourly trips taken during different days of the week. It can be inferred from the graph given below that during weekdays, maximum number of trips are taken at hour 6 AM, 7 AM and 8 AM whereas for weekends, maximum trips are taken during hours 11PM, 12AM and 1AM.

According to the peak timing selection criterion, the mean of trips is taken and hourly range between 6 AM to 9 AM for weekdays and 12 AM to 3 AM for weekends is considered.

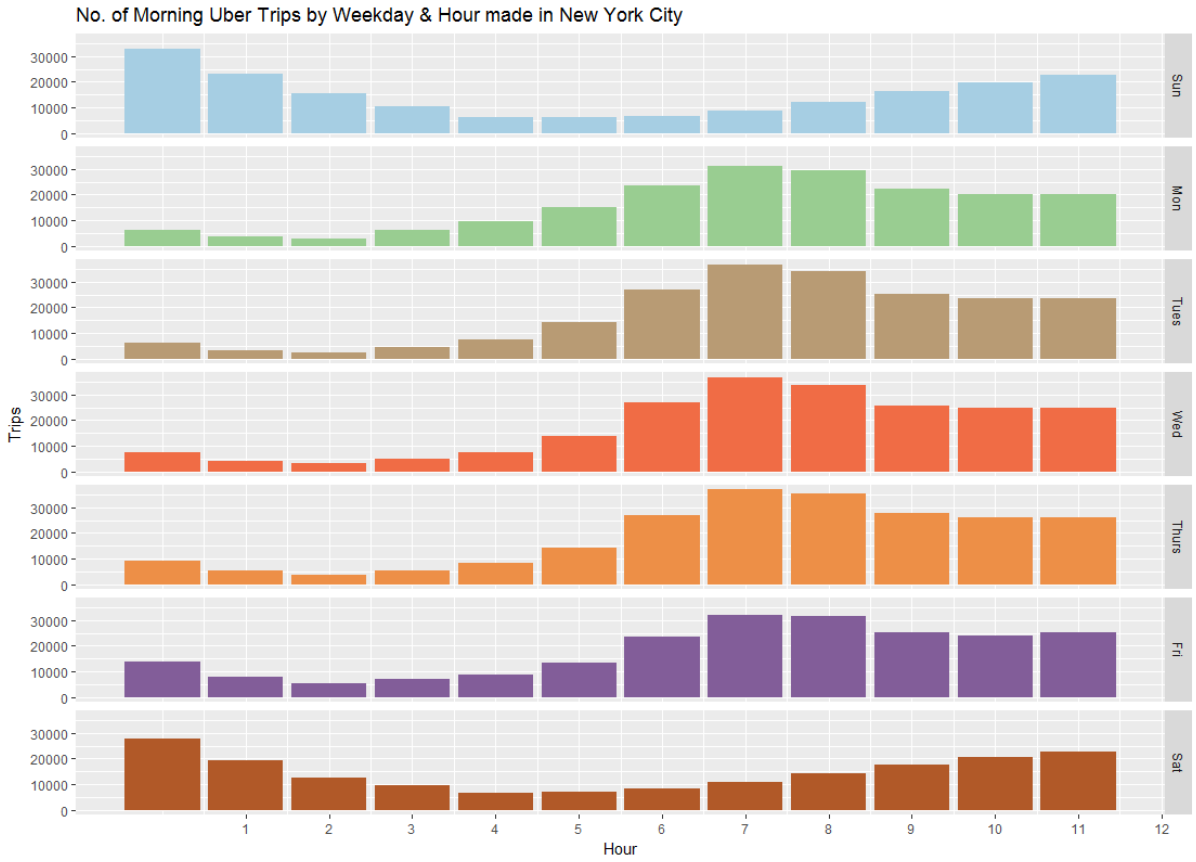


Figure 5.6: Hourly Morning Trips by Weekday

5.2.7 Evening Trip Trends

It can be seen from the graphs 5.7 given below that during weekdays, maximum trips can be observed from 4 PM to 7 PM whereas no significant timings are observed during weekends. During weekends, the number of trips are almost similar with no major timings acting as peak timings. Hence, no inference can be drawn solely considering the weekend and morning trips are also needed to be considered for analysis of peak timings during weekends.

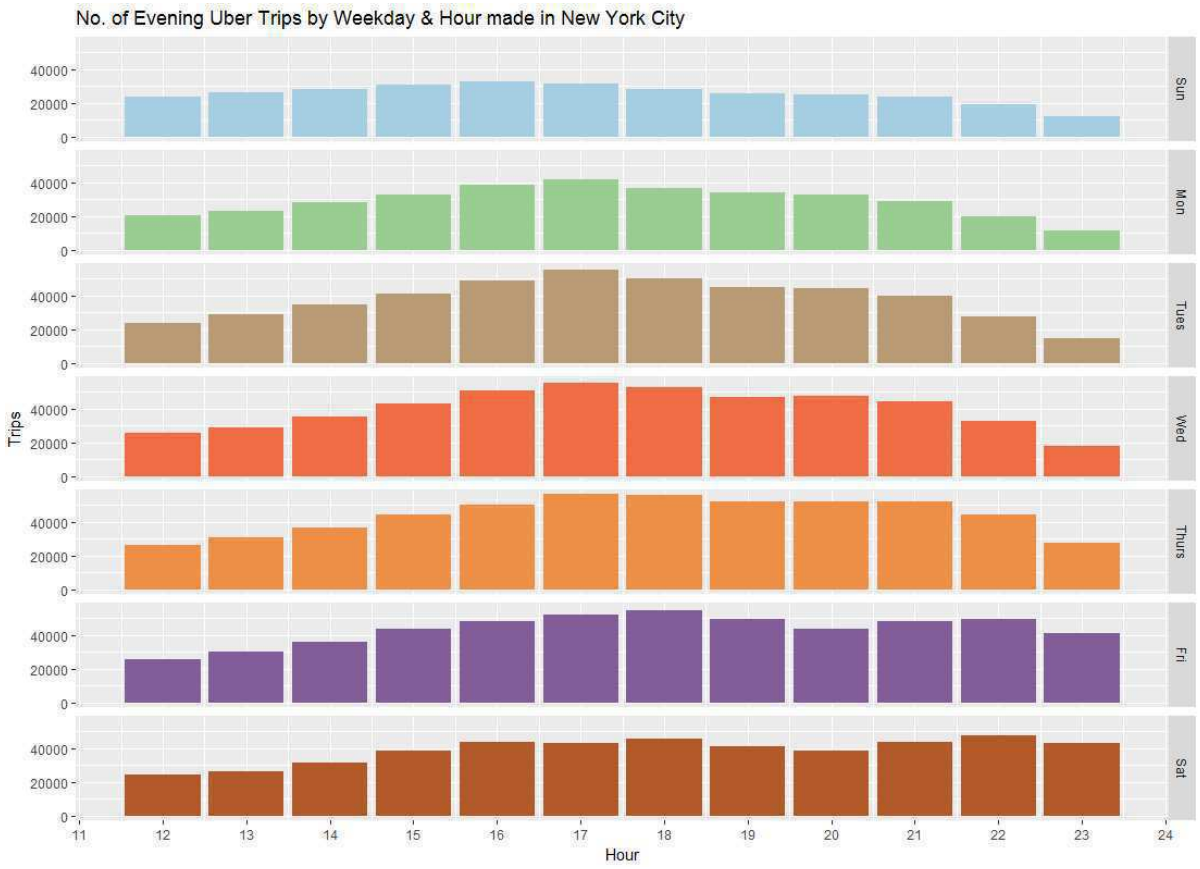


Figure 5.7: Hourly Evening Trips by Weekday

5.3 Outlier Detection and Removal

The figures 5.8 to 5.10 given below show the boxplot and histogram of all the three datasets before and after removal/treatment of the outliers. It can be observed that a spread-out boxplot is obtained after treating the outliers.

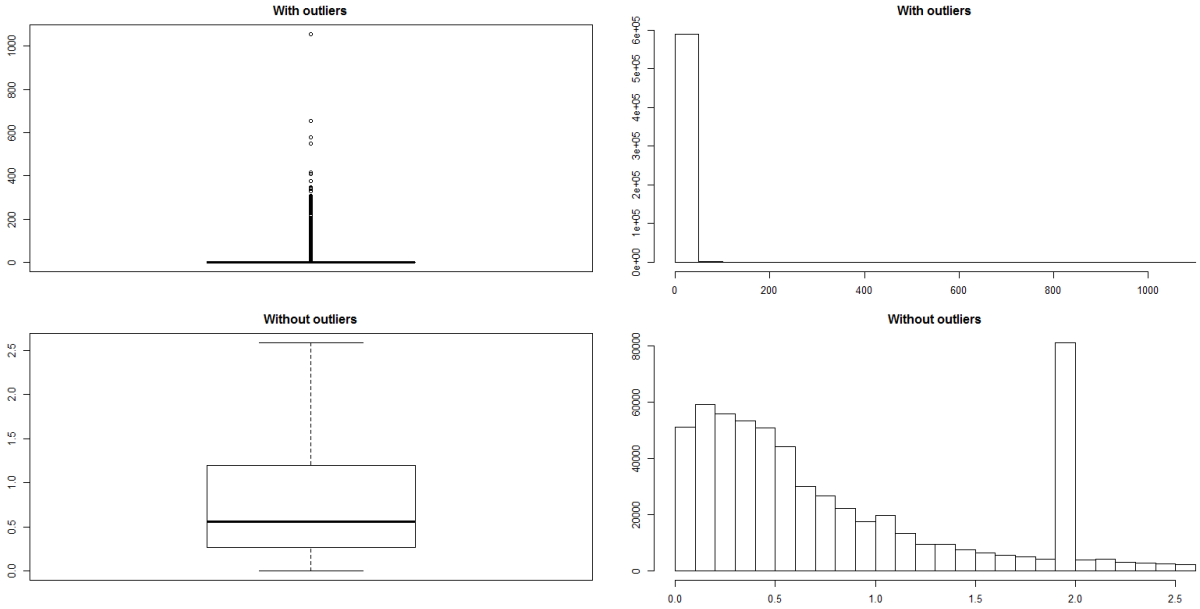


Figure 5.8: Weekday Morning Outlier Detection and Removal

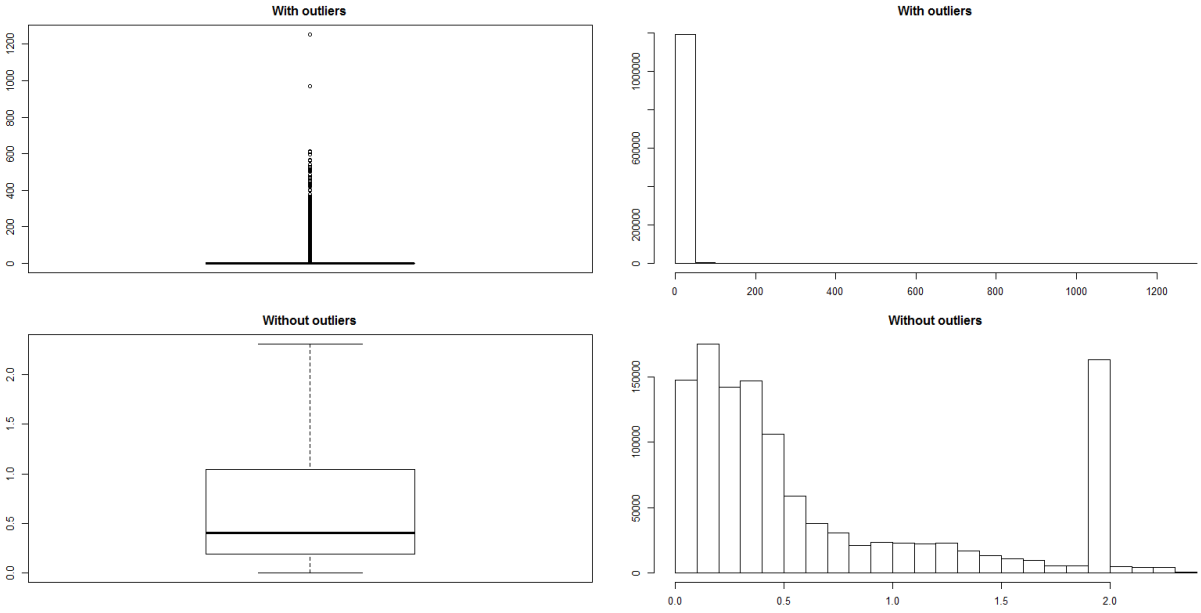


Figure 5.9: Weekday Evening Outlier Detection and Removal

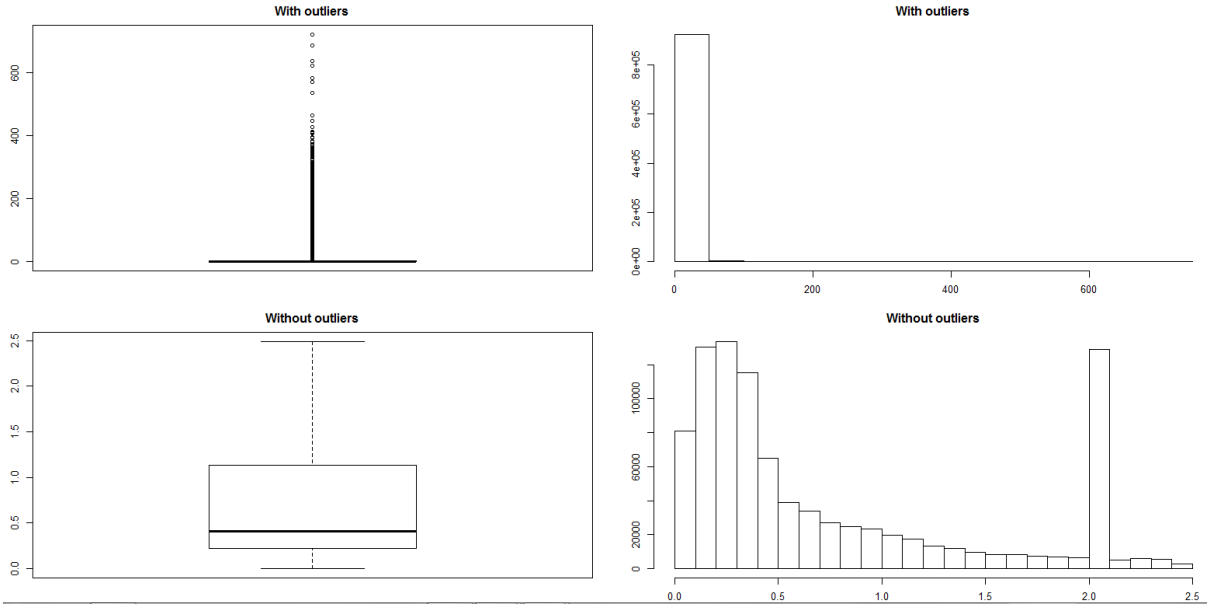
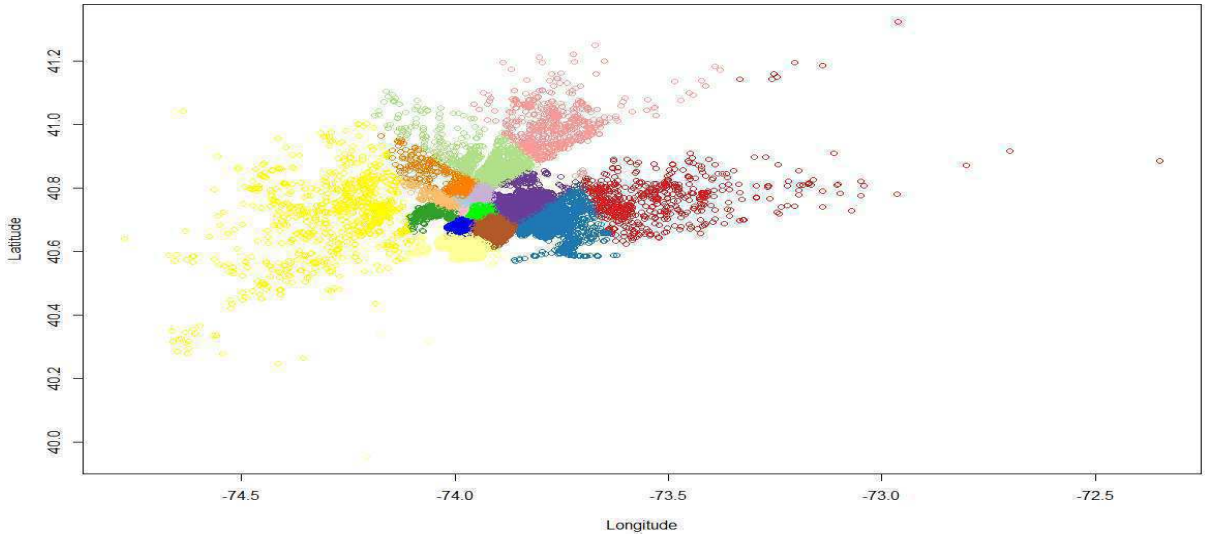


Figure 5.10: Weekend Outlier Detection and Removal

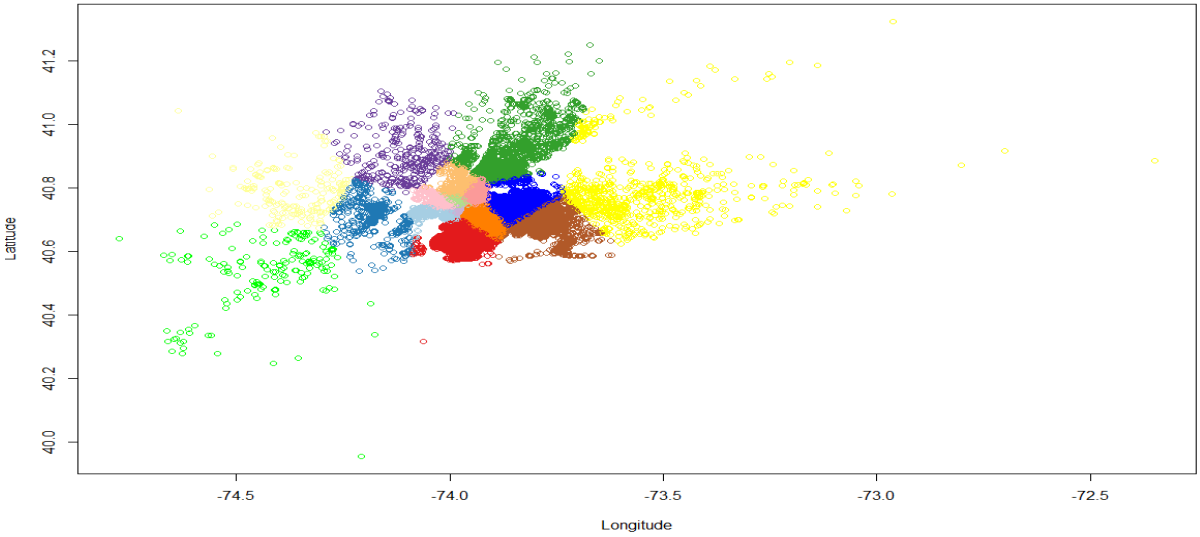
5.4 Clustering Results

On each dataset created after grouping and filtering of original dataset, all the three clustering algorithm k-means, k-median and CLARA ae implemented one by one. Based on the results obtained for the internal and external quality indices, the best clustering algorithm is chosen for hotspot detection. The cluster centeres are treated as the hotspot for that area. Based on the type of dataset under consideration, the hotspot category is detected according to the time of the day and day of the week.

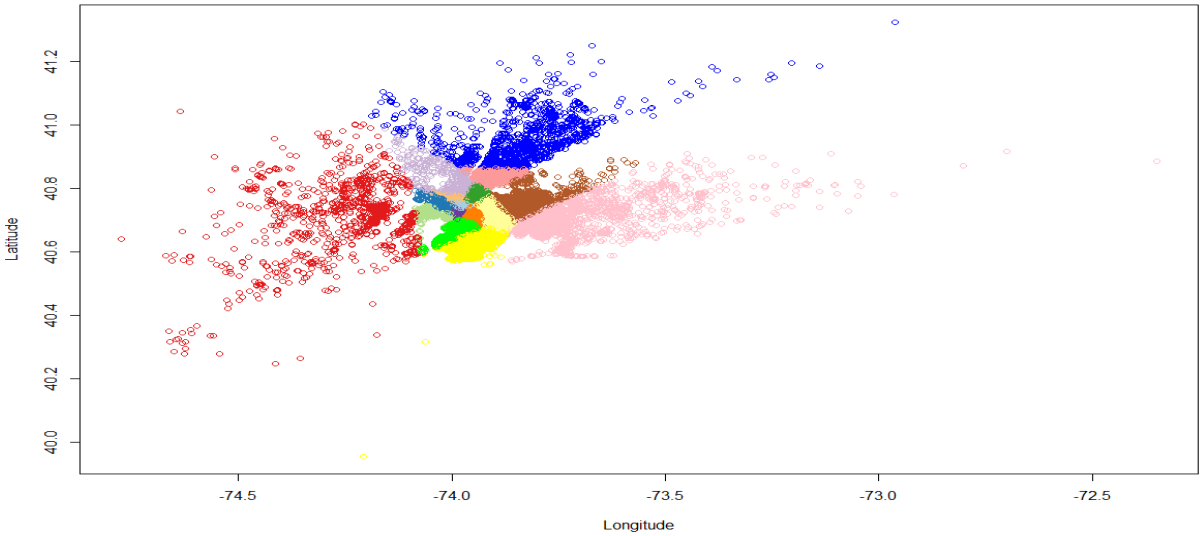
The figure 5.11 shows the clusters formed as a result of all the above mentioned three algorithms. The number of clusters to be formed is fixed at $k = 16$ for all the clustering algorithms. Hence, 16 clusters are formed in each of the figure from 5.11 to 5.13.



(a) Weekday Morning K-Means

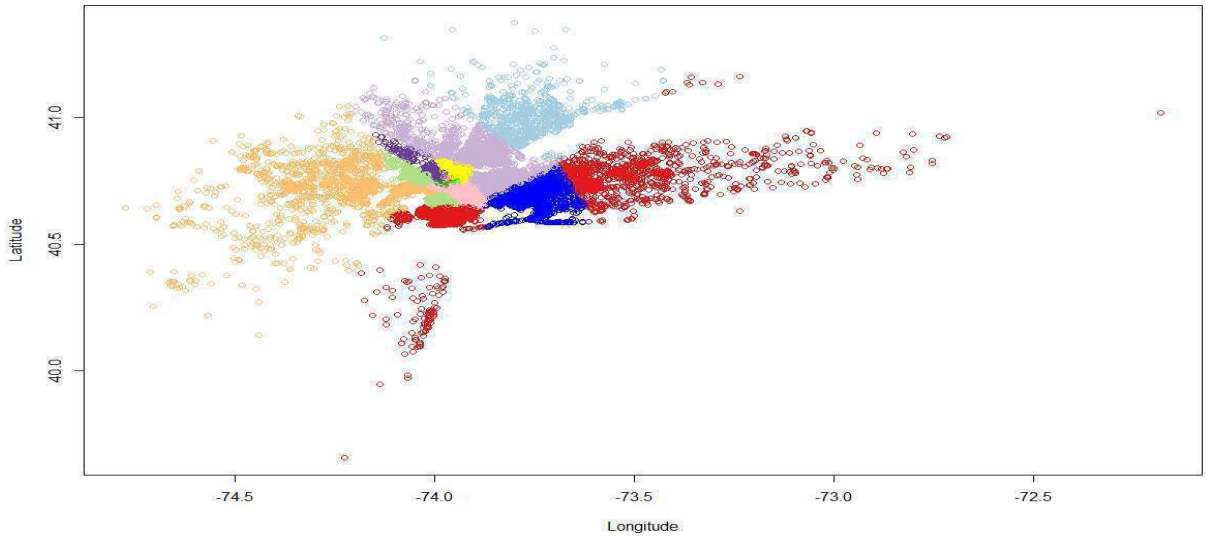


(b) Weekday Morning K-Medians

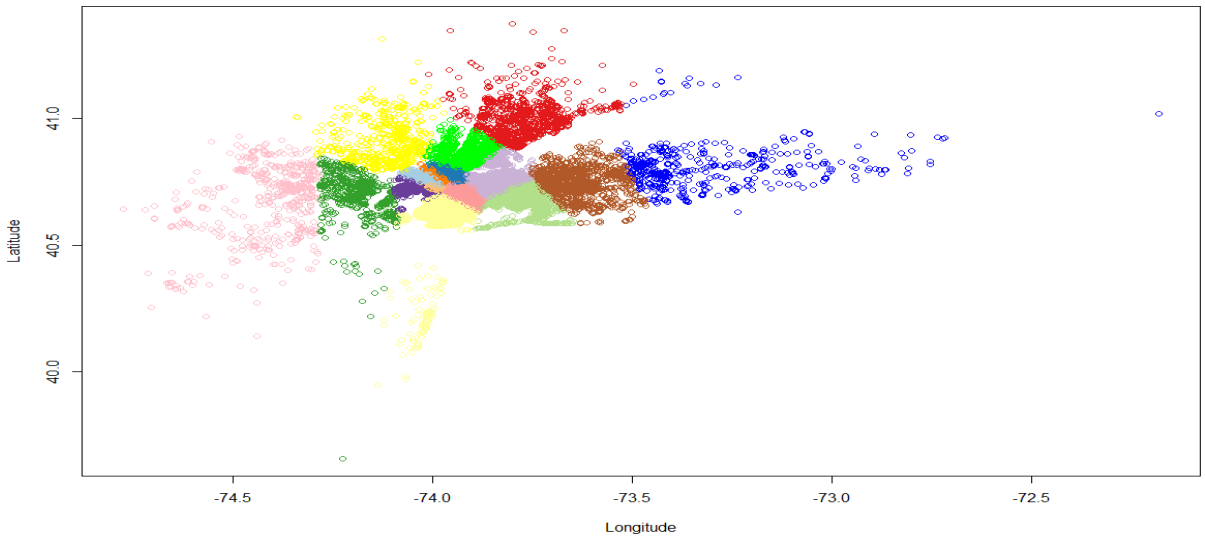


(c) Weekday Morning CLARA

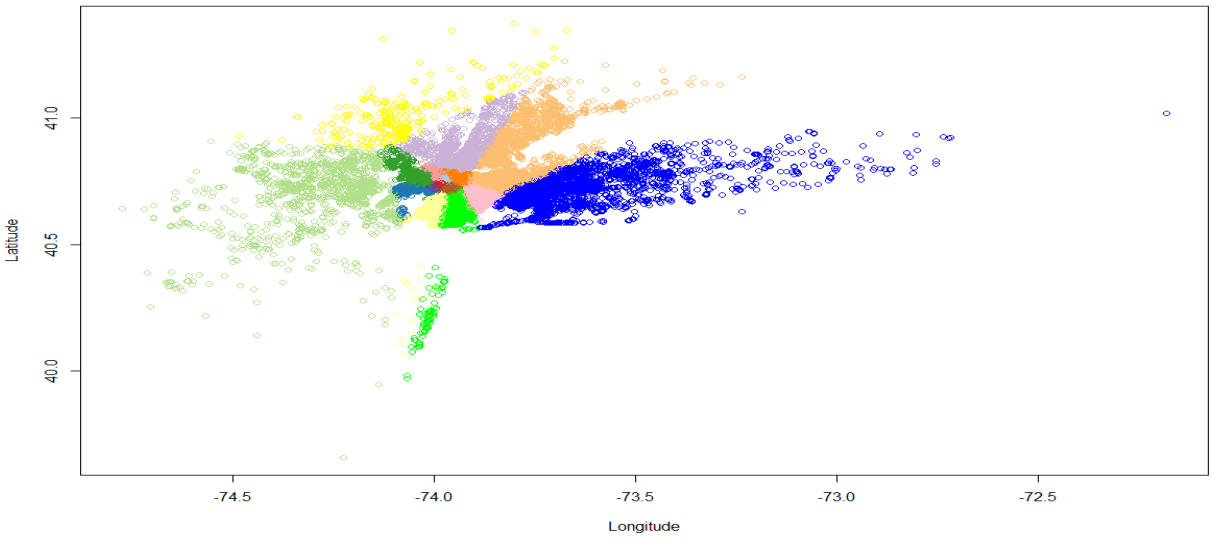
Figure 5.11: Weekday Morning Clustering Algorithms



(a) Weekday Evening K-Means

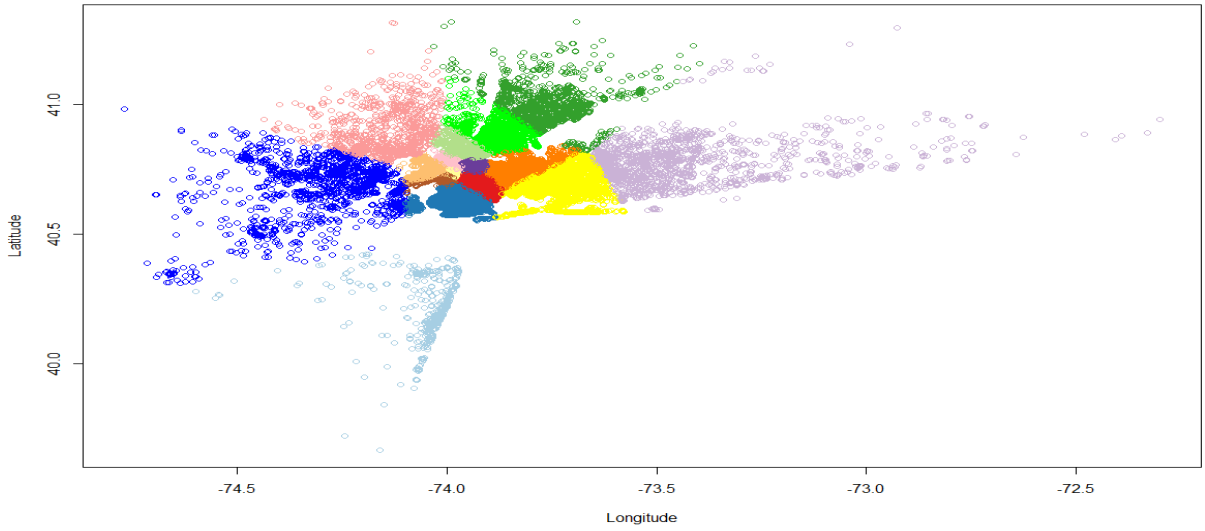


(b) Weekday Evening K-Medians

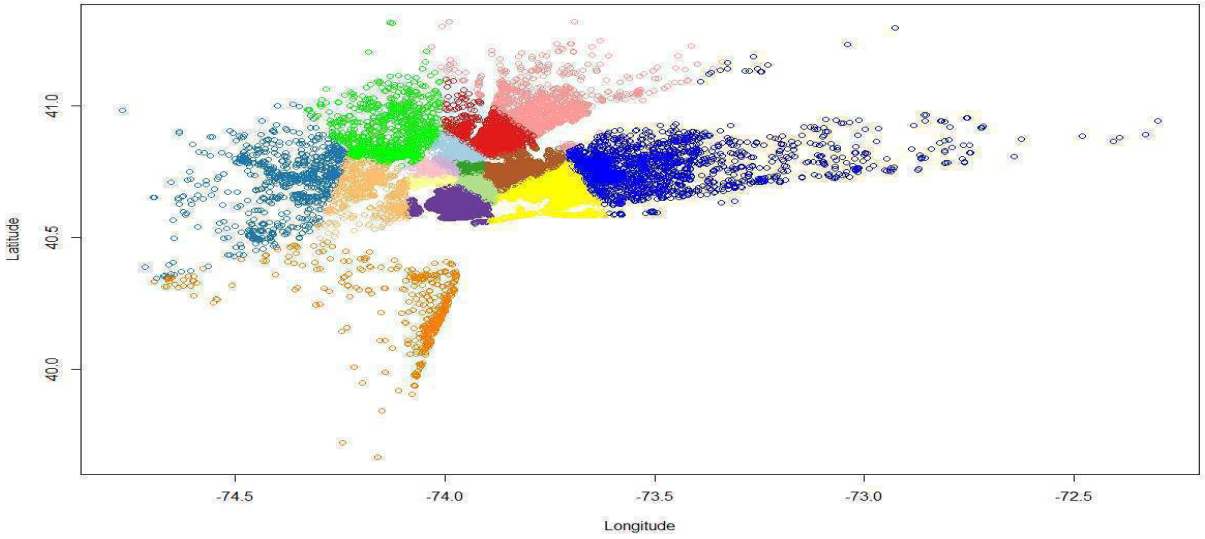


(c) Weekday Evening CLARA

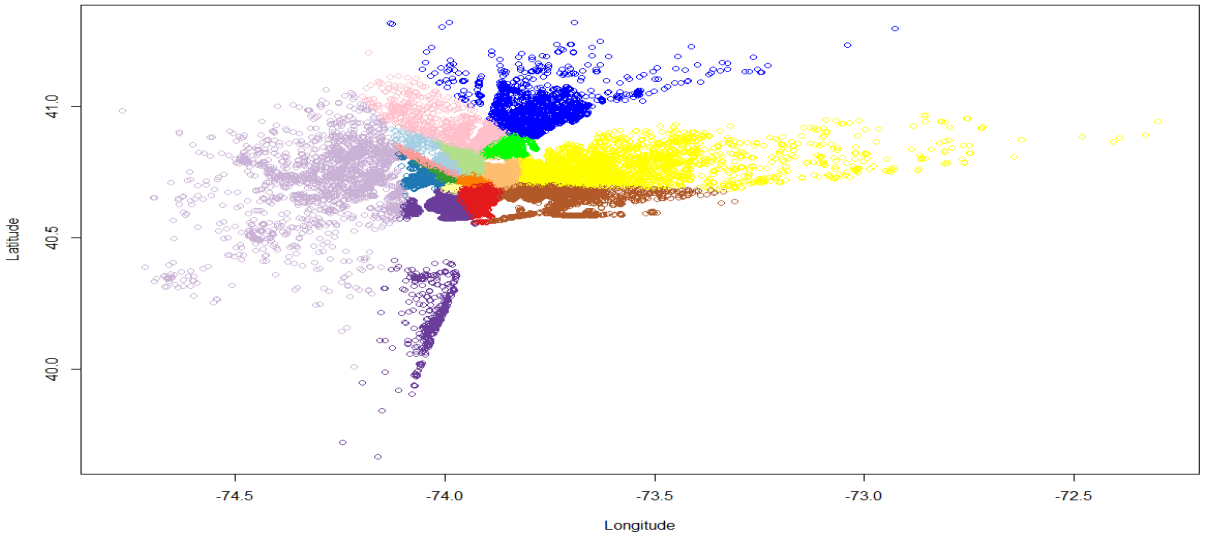
Figure 5.12: Weekday Evening Clustering Algorithms



(a) Weekend K-Means



(b) Weekend K-Medians



(c) Weekend CLARA

Figure 5.13: Weekend Clustering Algorithms

5.5 Internal Cluster Validation

From Chapter 1, it can be seen that different values of cluster parameters can result in different inferences drawn from the results. The table and graphs given below depict the internal cluster validation results to provide the best clustering algorithm for a given dataset.

From Table 5.3, it can be inferred that a lower DB-Index, higher CH-Index and lower SSE is achieved using the k-means algorithm. For this dataset, k-means clustering algorithm outperforms the other two clustering algorithms. Hence, the results obtained from k-means is taken to be an optimal value.

Table 5.3: Weekday Morning Cluster Quality Analysis

Scenario	Index	K-Means	CLARA	K-Median
Morning	DB-Index	0.8736371	1.092541	0.9470014
	CH-Index	380160.2	259014.1	330211.9
	SSE	221.6608	223.3189	238.3982

Table 5.4 shows the internal cluster quality indices for the weekday evening dataset. For this dataset, the k-median clustering algorithm is providing the best results with lowest DB-Index and SSE with highest CH-Index.

Table 5.4: Weekday Evening Cluster Quality Analysis

Scenario	Index	K-Means	CLARA	K-Median
Evening	DB-Index	0.9831674	1.165137	0.9174828
	CH-Index	769434.6	424904.4	795167.1
	SSE	441.4857	425.1924	420.1925

The weekends dataset also works best for the k-median clustering algorithm according to the results provided in table 5.5. Hence, hotspot detection can be done using the k-median clustering algorithm

Table 5.5: Cluster Quality Analysis

Scenario	Index	K-Means	CLARA	K-Median
Weekend	DB-Index	0.9286918	1.322452	0.919132
	CH-Index	513379.2	337399.1	527054.3
	SSE	545.5926	586.3831	532.8188

Figures 5.14, 5.15 and 5.16 provide the graphical results for all the three dataset and the comparison of clustering results obtained as a result of implementation on weekday morning, weekday evening and weekends dataset. These figures provide a statistical analysis of all the results to clearly extract the best clustering algorithm for a given dataset.

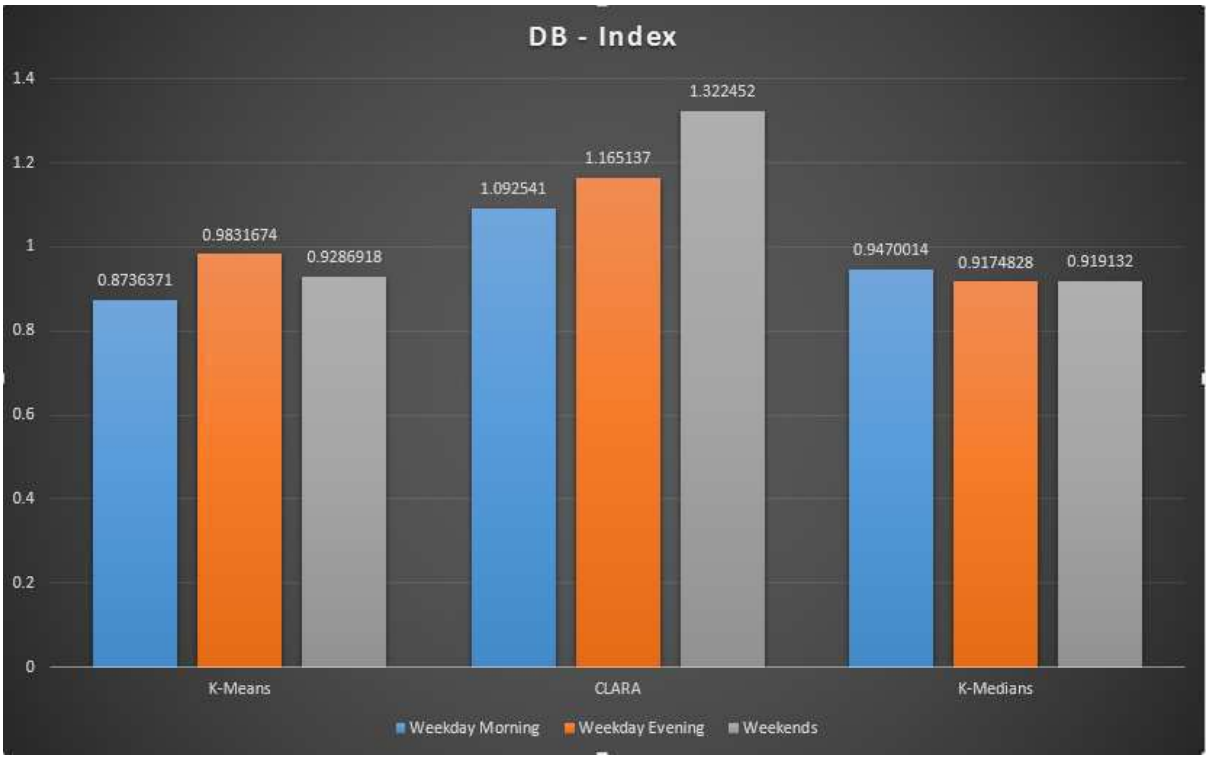


Figure 5.14: Davies Bouldin Index

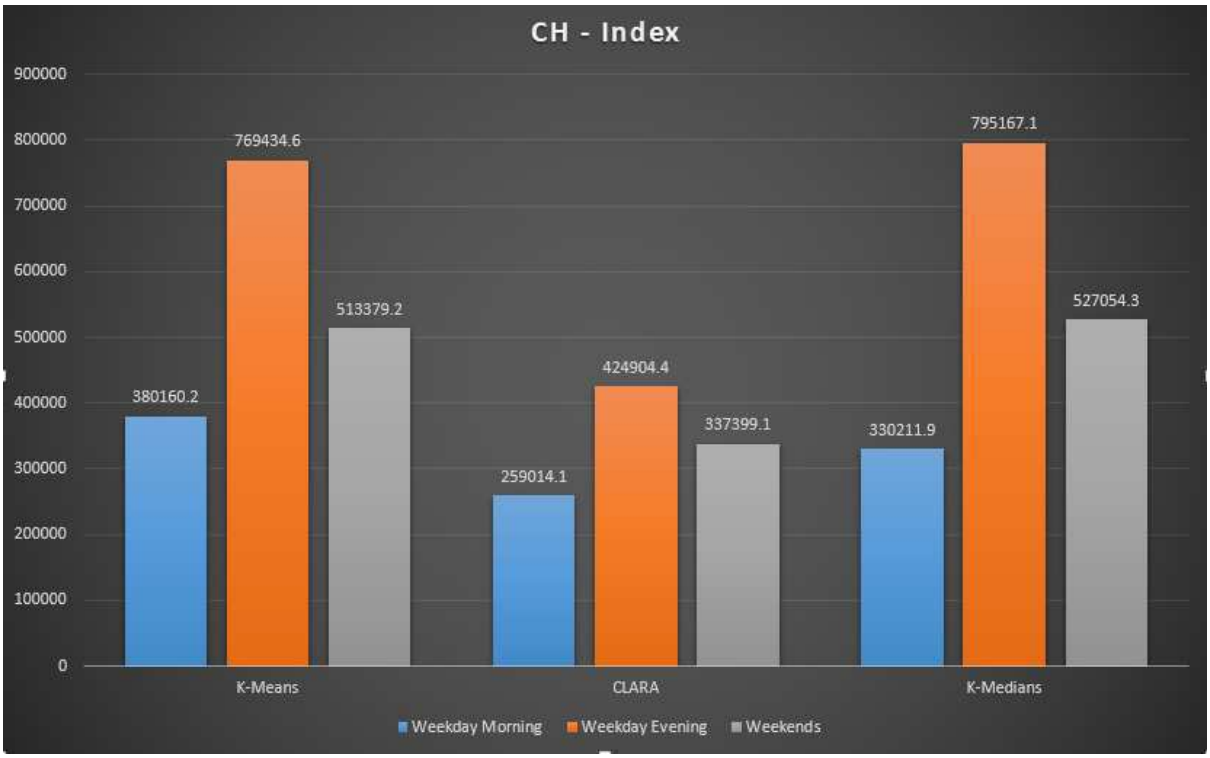


Figure 5.15: Calinhara Index

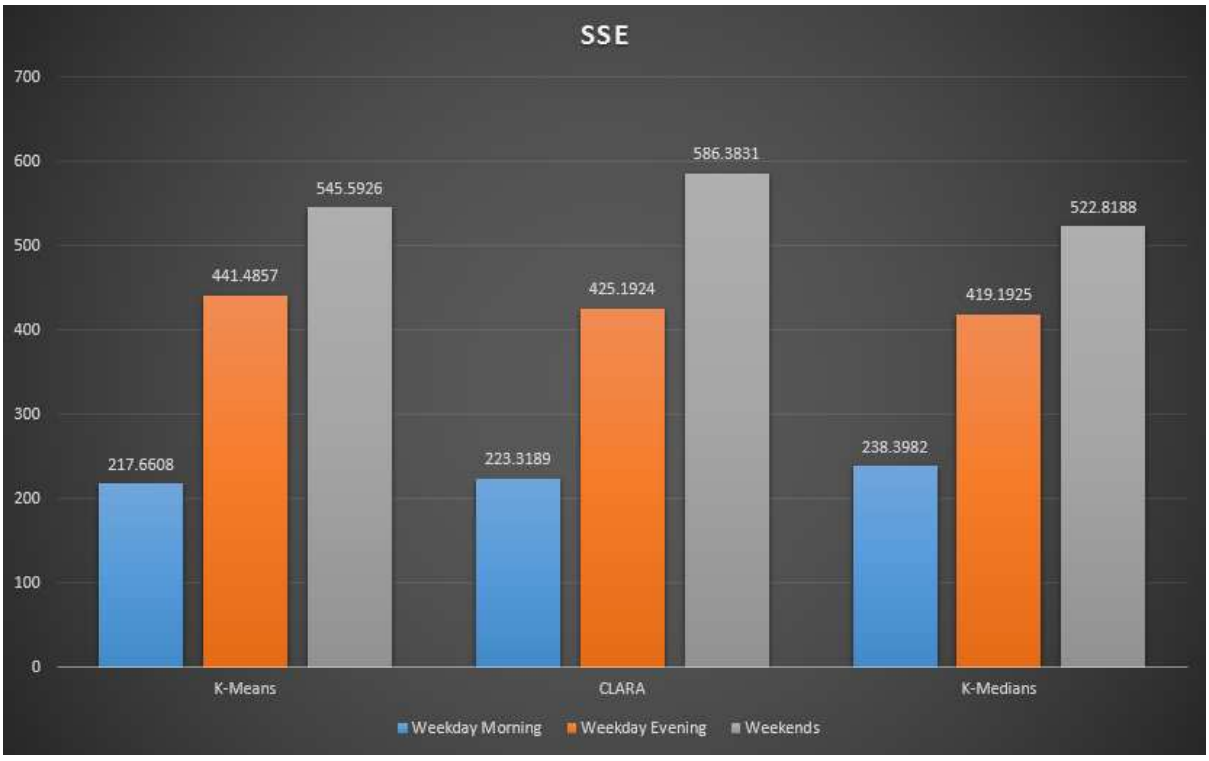


Figure 5.16: Sum of Squared Error

5.6 External Cluster Validation

Table 5.6 to 5.8 shows the cluster comparison for the weekday morning, weekday evening and weekends dataset repsectively. The values for external validation parameters such as Rand Index, Adjusted Rand Index, Fowlkes and Mallows Index and Mirkin Metrics is provided in the table 5.6 to 5.8. A graphical representation of these parameters are provided below in figures 5.17 to 5.20.

The inferences drawn from tables 5.6 to 5.8 can be done on the following criteria:

- Rand Index - A higher value of this index results in similar clustering with a range from 0 to 1. A value of 0 represents completely different clustering whereas 1 represent identical clustering.
- Adjusted Rand Index - This index is used in a similar way as General Rand Index expect that it can result in negative values as well.
- Fowlkes and Mallows Index - A higher value of this index results in high similarity between the clusters and benchmarks used for classification.
- Mirkin Metrics - A positive value is obtained for dissimilar cluster solutions whereas a null value is obtained for identical solutions.

Table 5.6, it can be seen that highly similar cluster solutions are obtained after applying k-means and k-medians clustering algorithms to the dataset with highest values of all the measures obtained in k-means and k-median clustering.

Table 5.6: Cluster Comparison for Weekday Morning Dataset

Scenario	Algorithm1	Algorithm2	R	AR	F	M
Morning	K-means	K-median	0.9341579	0.6433677	0.6800715	23016971604
	K-means	CLARA	0.9106718	0.5771093	0.6332463	31227219274
	K-medians	CLARA	0.9139329	0.5939077	0.6480588	30087200170

Table 5.7 shows that highest similarity is obtained with k-means and CLARA when weekday evening clustering algorithms are considered. Highest value of R, AR, F and lowest value of M is obtained in mentioned algorithms.

Table 5.7: Cluster Comparison for Weekday Evening Dataset

Scenario	Algorithm1	Algorithm2	R	AR	F	M
Evening	K-means	K-median	0.9224098	0.652782	0.7037514	111257527060
	K-means	CLARA	0.9401346	0.775602	0.8132372	85841690880
	K-medians	CLARA	0.8925736	0.556311	0.6306356	154039948340

Table 5.8 shows that in the weekends dataset also, most similar clustering results are obtained using the clustering algorithms k-means and CLARA as in table 5.7 with the desired value of different parameters.

Table 5.8: Cluster Comparison for Weekend Dataset

Scenario	Algorithm1	Algorithm2	R	AR	F	M
Weekend	K-means	K-median	0.8795825	0.4730742	0.5429671	103765418144
	K-means	CLARA	0.9652096	0.8425825	0.8626144	29979411736
	K-medians	CLARA	0.8799788	0.4818384	0.5513816	103423947116

If two clustering results in highly similar clustering, then any of the algorithm can be used interchangeably without affecting the clustering results obtained using each of the algorithms. The figures 5.17 to 5.20 given below show the graphical representation of the external validation metrics for each of the three datasets to provide visualization of the results.

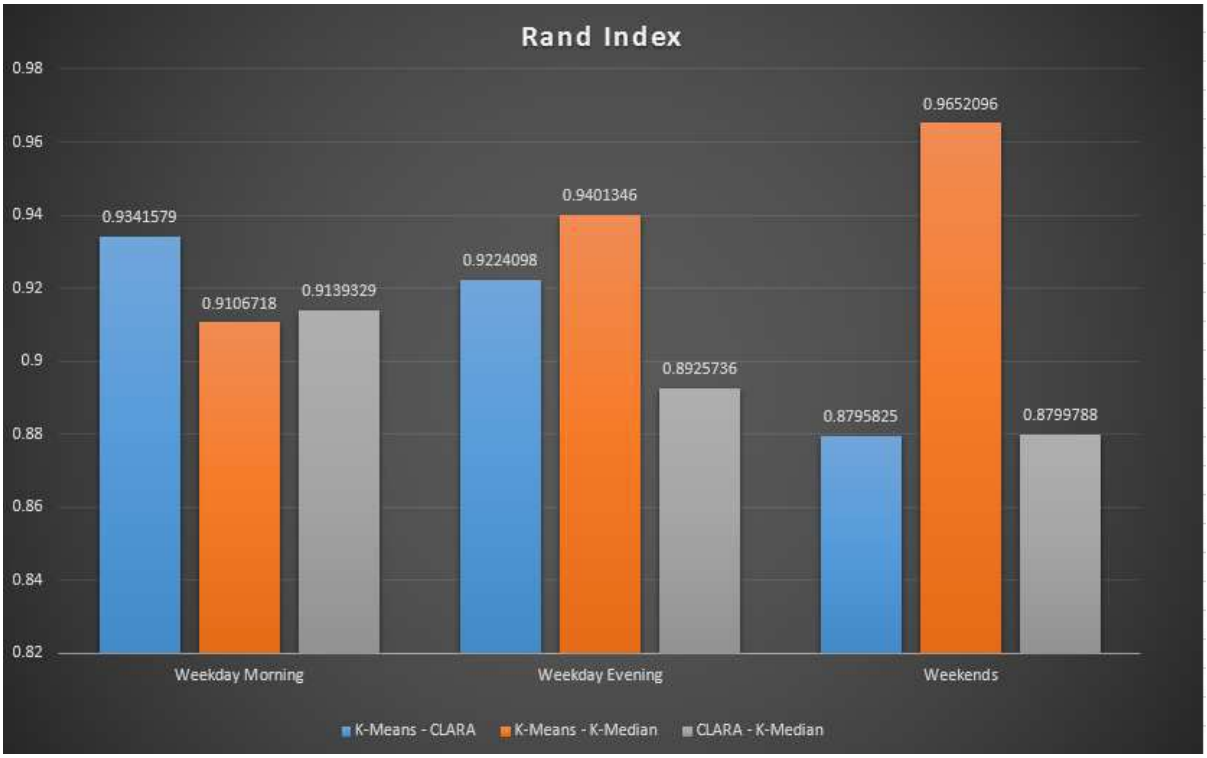


Figure 5.17: General Rand Index

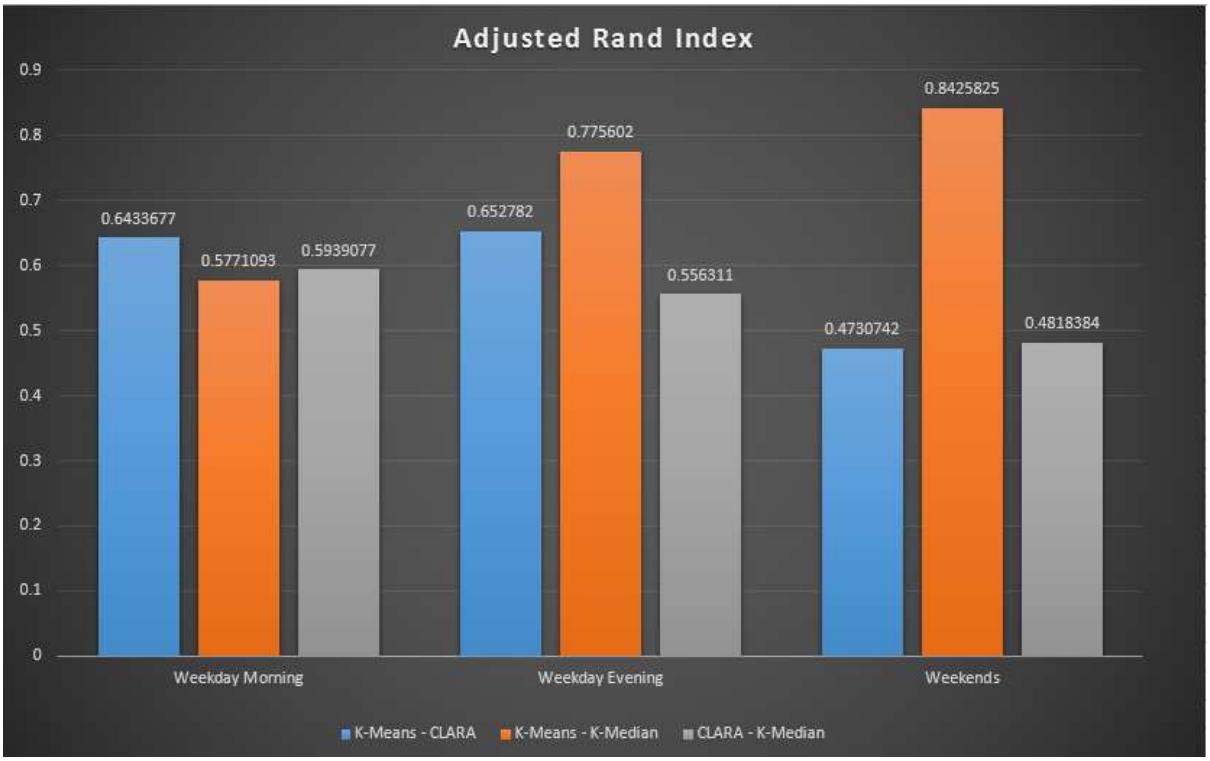


Figure 5.18: Adjusted Rand Index

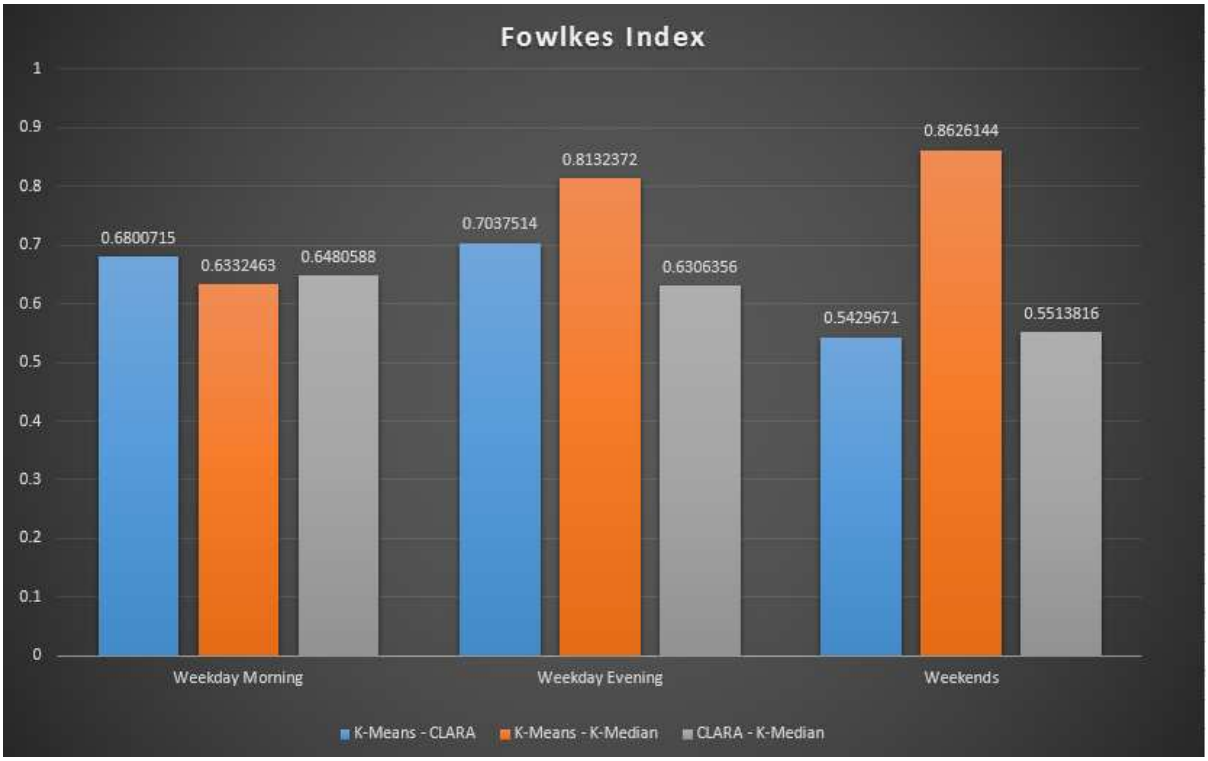


Figure 5.19: Fowlkes Index

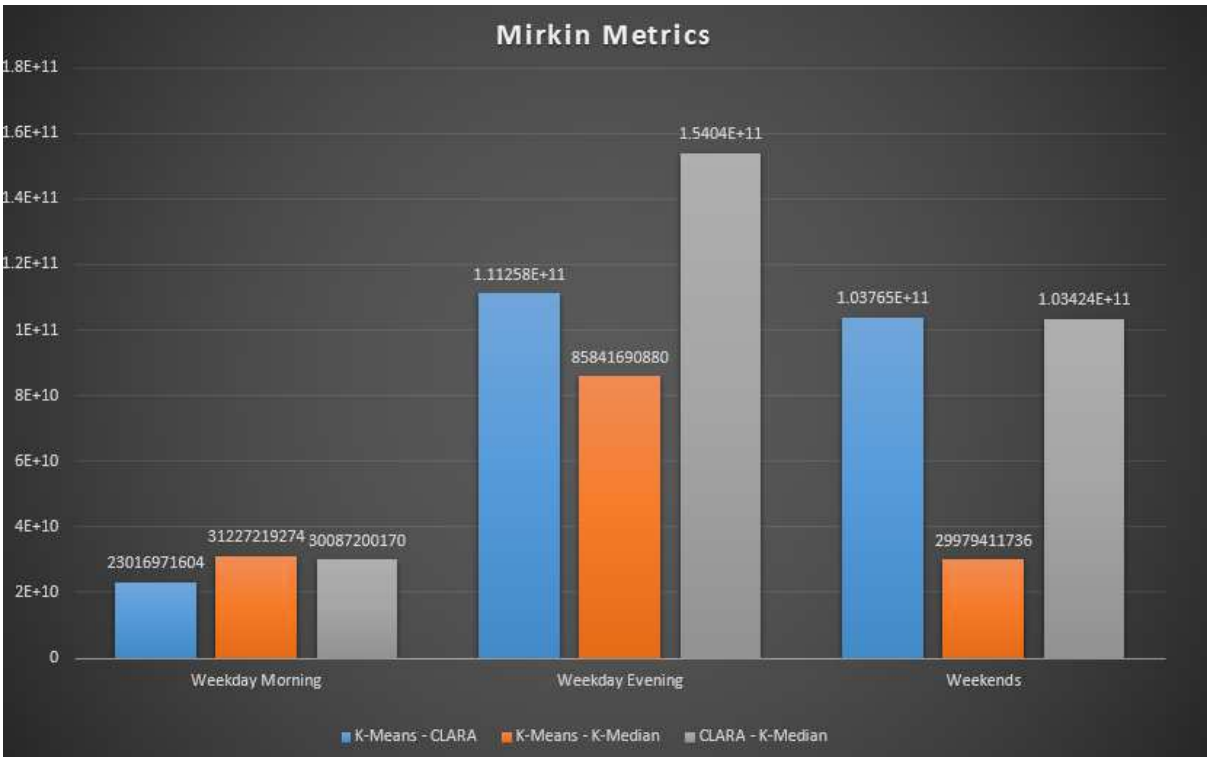


Figure 5.20: Mirkin Metrics

5.7 Base Address Clustering Results

For a multi-layer clustering approach to be implemented, the knowledge about the existing taxi base locations needs to be utilized for efficient base setup in order to serve high demand during the peak hours. The figure 5.21 shows the existing taxi base locations clustered into 10 partitions using CLARA clustering algorithm, with each partition allocated to a different hotspot area.

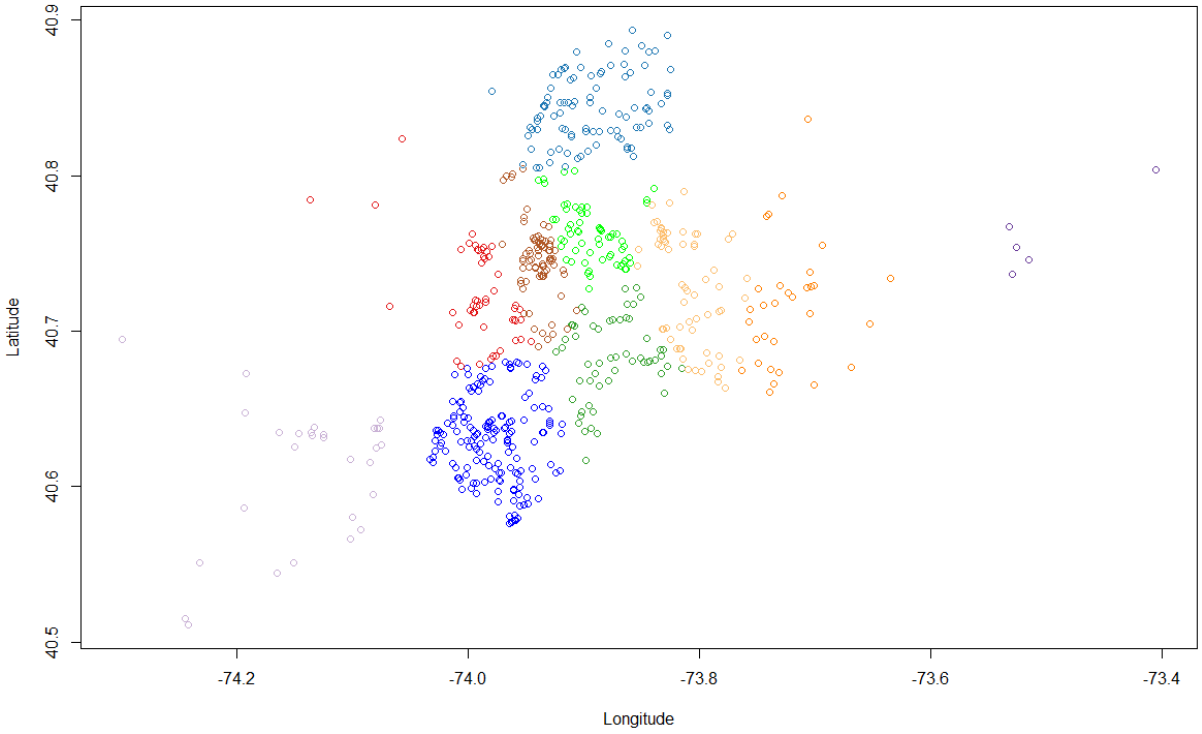


Figure 5.21: Base Address Clustering

The overlapping clustering results obtained from the relevant hotspot detection and base setup location is treated as a multi-layer clustering technique in order to achieve ease of mapping the incoming pickup from a customer.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

With more and more people opting for cabs in metro cities, fulfilling the every-increasing demand has become a challenge for both the public and private sectors. The public transport alone is not able to serve the demand and customer satisfaction due to over-crowded public transport during peak hours which are not at all preferrable by people. Hence, private transport and cabs have established themselves in order to provide the necessary comfort and easy travelling. This thesis aims at reducing the problems faced by both the customers and the cab owners in metro-cities and provide even more convenient travelling.

In this thesis, the primary taxi imbalance problem is tackled using the location based service and time based service by the cab owner to the customers. Clustering is the major technique used to resolve this problem. Uber dataset for NYC is gathered for analysis and taxi imbalance problem reduction. Since time and day play a major role in extracting knowledge from the data and draw inferences from it, data visualization and analysis is also performed on this data for further insight into the data.

Hence, the original dataset is first split into three categories weekday morning, weekday evening and weekend. The hotspots detected in each of these categories is treater differently. For example, weekday morning incoming requests generally will be made from residential areas and there will be very low variability in the timings since the standard work timings are followed. The requests observed during weekday evenings will mostly be made from office places or other similar work places whereas the weekends will observe pickups from residential places in the morning and party places during evening and night. This segregation of data into three categories allows time and day specific patterns to be extracted out of all the datasets for further analysis.

Every dataset consists of outliers and noise which can result in high deviation from the actual results and provide incorrect or inconsistent results, hence outlier detection and removal is performed on the dataset in order to cleanse the dataset and make it

cluster-ready. Different types of clustering techniques such as k-means, k-median and CLARA is implemented on each of these three dataset in order to see which clustering algorithm works best for one kind of data. Since partitioned clustering require the number of clusters to be specified prior to the algorithm execution, calinhara index and elbow method are used to identify the optimal clusters to be formed in each dataset. Once the optimal value is obtained, then the cluster partitions can be created using different clustering algorithms. The cluster centers hence obtained as a result of different clustering algorithms are treated as hotspots.

In order to asses the goodness and quality of clusters formed, different internal and external quality measures are used for analysis. The internal quality measures such as Davies Bouldin Index, Calinhara Index and SSE are used whereas external quality measures such as Rand Index, Fowlkes and Mallows Index and Mirking Metrics are used for evaluation.

It has been observed that k-means works best for the weekday morning dataset whereas k-median provides best results in case of weekday evening and weekend dataset. Hence, using these clustering techniques on the given dataset can provide accurate results for detecting the hotspots around the city.

Also, in order to further reduce the taxi imbalance problem, proper base locations are identified in this thesis in order to reduce the average waiting time of the customer and reduce the cancellation rate. In order to achieve this, the existing locations of the taxi bases are clustered in order to see the geographic location of the taxi base and allocate the incoming request to the nearest taxi base. An overlapping clustering of hotspot detection and base setup location results in appropriate base to be allocated the request. This approach also help the the taxi bases to take higher trips which are observing a lower number of pickups as compared to the other ones.

Hence, this multi-layer clustering approach is able to resolve the taxi imbalance problem to a great extent while giving a chance to other taxi bases to flourish as much as the other ones. A real-life problem is tackled in this thesis which benefits both the service providers as well as service consumers.

6.2 Future Work

There are thousands of trips taken at different locations in a city each day, the data once accumulated becomes huge which becomes difficult to work upon ana analyze. Also, since the amount of data under consideration is huge, not all the clustering algorithms can be implemented on such huge data. In order to tackle such large data, big data approaches need to be introduced for analysis. None of the density based clustering algorithms can be implemented on this huge data without using big data techniques. Big data, hence becomes an important feature which is necessary for further improved results as the data under consideration keeps expanding.

References

[1] Santi Phithakkitnukoon, Marco Veloso, Carlos Bento, Assaf Biderman, and Carlo Ratti. Taxi-aware map: Identifying and predicting vacant taxis in the city. In *AmI*, pages 86–95. Springer, 2010.

[2] Der-Horng Lee and Xian Wu. Dispatching strategies for the taxi-customer searching problem in the booking taxi service. In *In: Proceedings of the Transportation Research Board 92nd Annual Meeting*, 2013.

[3] H.A. Abbass. The self-adaptive pareto differential evolution algorithm. In *IEEE Congress on Evolutionary Computation (CEC)*, volume 1, pages 831–836, 2002.

[4] Neema Davis, Gaurav Raina, and Krishna Jagannathan. A multi-level clustering approach for forecasting taxi travel demand. In *Intelligent Transportation Systems (ITSC), 2016 IEEE 19th International Conference on*, pages 223–228. IEEE, 2016.

[5] Dongchang Liu, Shih-Fen Cheng, and Yiping Yang. Density peaks clustering approach for discovering demand hot spots in city-scale taxi fleet dataset. In *Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on*, pages 1831–1836. IEEE, 2015.

[6] C-T Lu, Dechang Chen, and Yufeng Kou. Algorithms for spatial outlier detection. In *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*, pages 597–600. IEEE, 2003.

[7] Victoria Hodge and Jim Austin. A survey of outlier detection methodologies. *Artificial intelligence review*, 22(2):85–126, 2004.

[8] Ville Hautamäki, Svetlana Cherednichenko, Ismo Kärkkäinen, Tomi Kinnunen, and Pasi Fränti. Improving k-means by outlier removal. *Image Analysis*, pages 219–227, 2005.

[9] Carl Olsson, Anders Eriksson, and Richard Hartley. Outlier removal using duality. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1450–1457. IEEE, 2010.

[10] H Miller and J Han. Spatial clustering methods in data mining: a survey. *Geographic data mining and knowledge discovery, Taylor and Francis*, 2001.

[11] Maria Halkidi, Yannis Batistakis, and Michalis Vazirgiannis. On clustering validation techniques. *Journal of intelligent information systems*, 17(2):107–145, 2001.

[12] Ferenc Kovács, Csaba Legány, and Attila Babos. Cluster validity measurement techniques. In *6th International symposium of hungarian researchers on computational intelligence*, 2005.

[13] Ricardo JGB Campello. A fuzzy extension of the rand index and other related indexes for clustering and classification assessment. *Pattern Recognition Letters*, 28(7):833–841, 2007.

[14] Kai Zhang, Zhiyong Feng, Shizhan Chen, Keman Huang, and Guiling Wang. A framework for passengers demand prediction and recommendation. In *Services Computing (SCC), 2016 IEEE International Conference on*, pages 340–347. IEEE, 2016.

[15] Slava Kisilevich, Florian Mansmann, Mirco Nanni, and Salvatore Rinzivillo. Spatio-temporal clustering. *Data mining and knowledge discovery handbook*, pages 855–874, 2010.

[16] Kai Zhao, Denis Khryashchev, Juliana Freire, Cláudio Silva, and Huy Vo. Predicting taxi demand at high spatial resolution: Approaching the limit of predictability. In *IEEE International Conference on BigData*, 2016.

[17] Luis Moreira-Matias, Joao Gama, Michel Ferreira, Joao Mendes-Moreira, and Luis Damas. Predicting taxi–passenger demand using streaming data. *IEEE Transactions on Intelligent Transportation Systems*, 14(3):1393–1402, 2013.

[18] Pengxiang Zhao, Kun Qin, Xinyue Ye, Yulong Wang, and Yixiang Chen. A trajectory clustering approach based on decision graph and data field for detecting hotspots. *International Journal of Geographical Information Science*, 31(6):1101–1127, 2017.

[19] Ke Fan, Daqiang Zhang, Yunsheng Wang, and Shengjie Zhao. Discovering urban social functional regions using taxi trajectories. In *Ubiquitous Intelligence and Computing and 2015 IEEE 12th Intl Conf on Autonomic and Trusted Computing and 2015 IEEE 15th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UIC-ATC-ScalCom), 2015 IEEE 12th Intl Conf on*, pages 356–359. IEEE, 2015.

[20] Xi Zhu and Diansheng Guo. Mapping large spatial flow data with hierarchical clustering. *Transactions in GIS*, 18(3):421–435, 2014.

[21] Ran Wang, Chi-Yin Chow, Yan Lyu, Victor Lee, Sam Kwong, Yanhua Li, and Jia Zeng. Taxirec: recommending road clusters to taxi drivers using ranking-based extreme learning machines. In *Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems*, page 53. ACM, 2015.

[22] Jing Yuan, Yu Zheng, Xing Xie, and Guangzhong Sun. T-drive: Enhancing driving directions with taxi drivers' intelligence. *IEEE Transactions on Knowledge and Data Engineering*, 25(1):220–232, 2013.

[23] Meng Qu, Hengshu Zhu, Junming Liu, Guannan Liu, and Hui Xiong. A cost-effective recommender system for taxi drivers. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 45–54. ACM, 2014.

[24] Seong Ping Chuah, Huayu Wu, Yu Lu, Liang Yu, and Stephane Bressan. Bus routes design and optimization via taxi data analytics. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 2417–2420. ACM, 2016.

[25] Jin Liu, Xiao Yu, Zheng Xu, Kim-Kwang Raymond Choo, Liang Hong, and Xiaohui Cui. A cloud-based taxi trace mining framework for smart city. *Software: Practice and Experience*, 2016.

[26] Li Gong, Xi Liu, Lun Wu, and Yu Liu. Inferring trip purposes and uncovering travel patterns from taxi trajectory data. *Cartography and Geographic Information Science*, 43(2):103–114, 2016.

[27] Feng Mao, Minhe Ji, and Ting Liu. Mining spatiotemporal patterns of urban dwellers from taxi trajectory data. *Frontiers of Earth Science*, 10(2):205–221, 2016.

[28] Yuanhang Hu, Yujiu Yang, and Biqing Huang. A comprehensive survey of recommendation system based on taxi gps trajectory. In *Service Science (ICSS), 2015 International Conference on*, pages 99–105. IEEE, 2015.

[29] Aakash Deep Singh, Wei Wu, Shili Xiang, and Shonali Krishnaswamy. Taxi trip time prediction using similar trips and road network data. In *Big Data (Big Data), 2015 IEEE International Conference on*, pages 2892–2894. IEEE, 2015.

[30] Chengkun Liu, Kun Qin, and Chaogui Kang. Exploring time-dependent traffic congestion patterns from taxi trajectory data. In *Spatial Data Mining and Geographical Knowledge Services (ICSDM), 2015 2nd IEEE International Conference on*, pages 39–44. IEEE, 2015.

[31] Urška Demšar, Paul Harris, Chris Brunsdon, A Stewart Fotheringham, and Sean McLoone. Principal component analysis on spatial data: an overview. *Annals of the Association of American Geographers*, 103(1):106–128, 2013.

[32] Garrett Grolemund, Hadley Wickham, et al. Dates and times made easy with lubridate. *Journal of Statistical Software*, 40(3):1–25, 2011.

[33] Hadley Wickham and Romain Francois. dplyr: A grammar of data manipulation. *R package version 0.4*, 1:20, 2015.

[34] Jean Mundahl Engels and Paula Diehr. Imputation of missing longitudinal data: a comparison of methods. *Journal of clinical epidemiology*, 56(10):968–976, 2003.

[35] Xuelei Hu and Lei Xu. A comparative study of several cluster number selection criteria. In *International Conference on Intelligent Data Engineering and Automated Learning*, pages 195–202. Springer, 2003.

List of Publications

International Conference

1. Verma, N., and Baliyan, N. "PAM Clustering Based Taxi Hotspot Detection For Informed Driving" presented at IEEE 8th International Conference on Computing, Communication and Networking Technologies at IIT Delhi, 2017 [In Press]

Journal

1. Verma, N. and Baliyan, N. "An Improved KDD Process on Uber Taxi Dataset" to be communicated.

Video Link

The link for the video presentation of the dissertation is given below:

https://www.youtube.com/watch?v=clYYlXZYyek&t=7s

M.E.Thesis_Niharika

ORIGINALITY REPORT

%3	%2	%2	%1
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.docstoc.com Internet Source	%1
2	www.toxicology.org Internet Source	<%1
3	www.cs.uiuc.edu Internet Source	<%1
4	Zhou, Chunjie, Pengfei Dai, Fusheng Wang, and Zhenxing Zhang. "Predicting the passenger demand on bus services for mobile users", Pervasive and Mobile Computing, 2016. Publication	<%1
5	Han, Jiawei, Micheline Kamber, and Jian Pei. "Cluster Analysis", Data Mining, 2012. Publication	<%1
6	www.iaeng.org Internet Source	<%1
7	Submitted to The University of Manchester Student Paper	<%1
	Submitted to VIT University	