Technical University of Munich
Department of Civil, Geo and Environmental Engineering
Chair of Transportation Systems Engineering

Master's Thesis

Using Floating Car Data (FCD) to Infer Mobility Patterns

Abdallah Abu-Aisha

Supervised by:
Univ.-Prof. Dr. Constantinos Antoniou
M.Sc. Ralph Harfouche

December 25, 2018

Abstract

With the widespread use of location sensing technologies such as GPS-enabled vehicles, huge volumes of vehicle trajectory data are increasingly generated. The growing availability of such data opens up new opportunities for performing more sophisticated and comprehensive spatial and temporal analyses for planning and management of transportation systems. One of the most useful types of analysis in this context is traffic data clustering, which can help in understanding and revealing valuable insights into urban mobility patterns and travel behavior.

In this thesis, a six-day dataset of floating car data (FCD) from Munich city is used to extract meaningful urban mobility patterns. Hierarchical clustering analysis is used first to spatially cluster the trips in each day based on the coordinates of their origin and destination points, such that each cluster contains the trips that travel from one specific origin zone to another destination zone. Next, an innovative tool, called Relative Deviation Area (RDA), is introduced to help in understanding travel behavior in the resulting clusters. RDA aims to find the relative area by which a given trajectory is deviating from a referential trajectory (typically the least-cost path). RDA is computed for each trip in each cluster on each day. This is followed by investigating the relationship between trip average speed (V) and RDA for each day using Kernel regression method. The resulting regression curves are found sensible and consistent throughout all days, which indicates a potential association between the two variables. In addition, the relationship is temporally investigated at peak and off-peak periods. V values at peak periods are found to be lower than those at off-peak periods for the same value of RDA. Another case is tested where only private cars are considered, excluding all other vehicle types like taxicabs and trucks. The results showed that RDA values in private cars case are higher than those in all vehicle types case.

The output illustrates the potential of using Big Data to infer mobility patterns and travel behavior. The developed RDA tool is expected to have several applications in different fields such as urban and transportation planning, transportation demand management, and traffic monitoring.

Keywords: Floating car data (FCD), Spatial Clustering, Origin-Destination (OD) points, Mobility Patterns, Travel Behavior, Relative Deviation Area (RDA), Kernel Regression, R Software, Trajectory Analysis.

Acknowledgements

I would first like to express my sincere gratitude to Prof. Dr. Constantinos Antoniou of the Chair of Transportation Systems Engineering at the Technical University of Munich, for offering me the topic and providing the data and for his constant supervision.

Special thanks to my supervisor M.Sc. Ralph Harfouche for his admirable support and guidance throughout my thesis. The door to his office was always open whenever I ran into a trouble spot or had a question about my research or writing. He consistently allowed this paper to be my own work, but steered me in the right the direction whenever he thought I needed it.

I would also like to thank the experts who were involved in this research project: Dr. Tao Ma and Dr. Christos Katrakazas. Without their passionate participation and input, this thesis could not have been successfully conducted.

Last but not least, I must express my very profound gratitude to my parents, my wife, and my family for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

Declaration

I hereby declare that this thesis is an outcome of my own efforts and has not been published anywhere else before and not used in any other examination. Also, to mention that the materials and methods used and quoted in this thesis has been properly referenced and acknowledged.

Munich, December 25, 2018

Abdallah Abu-Aisha

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ALCAMP	Algorithm for finding the Least-Cost Areal Mapping between Paths
CDR	Call Detail Records
DA	Deviation Area
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
DD	Decimal Degrees
EE	External to External
EI	External to Internal
Eps	Maximum radius of Neighborhood
FCD	Floating Car Data
GCS	Geographic Coordinate System
GPS	Global Positioning System
ICT	Information and Communication Technology
IE	Internal to External
II	Internal to Internal
ITS	Intelligent Transportation Systems
K	Number of Clusters
kNNdist	k-Nearest Neighbor Distance
LCS	Longest Common Subsequence
MinPts	Minimum Number of Points
OD	Origin-Destination
PAM	Partition Around Medoids
RDA	Relative Deviation Area
SNN	Shared Nearest Neighbors
SSE	Sum of Squared Errors
UTM	Universal Transverse Mercator
V	Trip Average Speed
WSS	Within-Cluster Sum of Squares

1. Introduction

1.1 Background and Problem Statement

How people move in cities, where they are likely to go from a given location, and what they do in various locations at different times form human mobility and activity patterns (Huang, Li, & Xu, 2016). Understanding mobility patterns and human travel behavior plays a key role in urban planning, traffic forecasting, public transport management and location-based mobile applications, among others (Lian, Li, Gu, Huang, & Zhang, 2018).

Understanding and discovering mobility patterns require reliable and high-quality traffic information. With the advent of ubiquitous sensing technology, massive amount of rich mobility data (e.g., human daily activities and vehicle trajectories) has become increasingly available from various sources such as cellular network data, geo-tagged social media data, and GPS floating car data (FCD) (Guo, Zhu, Jin, Gao, & Andris, 2012; Lian et al., 2018). These new data sources have the ability to improve data quality and accuracy and the potential to complement data collected using conventional methods.

Mobility data analysis is of great importance in revealing valuable insights and advancing our understanding of complex space-time dynamics in a variety of domains, especially in urban planning and transportation management (Guo & Zhu, 2014). However, there are great challenges for mobility data analysis due to the massive data volume and the complexity of dealing with the spatial and temporal dimensions (Kim & Mahmassani, 2015). One of the most popular big data techniques used to deal with such challenges in analyzing mobility data is spatial clustering, which identifies distinct groups of trips or trajectories based on their geographical characteristics, such that there is greater similarity within each group than between groups (Kumar et al., 2018). The resulting clusters can provide useful insights into traffic flow patterns, trip planning, predicting passenger demand, traffic monitoring, and location-based services.

For decades, in order to provide long-term guidance and short-term strategies for transportation planning and urban development, several studies have been conducted to identify, understand and predict human mobility patterns and travel behavior. For instance, some studies tried to investigate drivers' route choice and the factors influencing their decisions (Levinson & Shanjiang, 2013; Sun, Zhang, Zhang, Chen, & Peng, 2014). However, there were no studies found in the literature that investigated the area-based deviation between drivers' actual chosen route and the least-cost route between two zones. Therefore, this thesis tries to find a way to fill this gap by presenting a novel tool that computes the deviation area between any two trajectories sharing the same origin and destination zone, with the help of ALCAMP algorithm presented by Mueller, Perelman, & Veinott (2016). This tool can help leveraging trajectory data collected by passive collection methods, such as FCD and cell phone GPS data to detect mobility patterns, and understand drivers' travel behavior and route choice decision.

1.2 Objectives and Research Question

This thesis aims to leverage traffic data collected via FCD technique in inferring mobility patterns and investigating users' travel behavior. Since the raw data contains a large number of trips with multiple attributes, it is not feasible to extract meaningful well-structured patterns from it. Therefore, it is necessary to group trips that have similar origin and destination points into spatial clusters first.

For a better understanding of travel behavior and especially route choice, this thesis also seeks to develop a way to compute the area by which a vehicle traveling from one zone to another is deviating from the least-cost route between these two zones. In addition, the relationship between this deviation area and trip average speed is investigated in this thesis.

Given the aforementioned thesis objectives, the main research question is formulated to be how to understand travel behavior within a given set of spatial clusters of trips?

The research question to be answered introduces a wide spectrum of secondary research questions such as:

- How to preprocess and clean the raw data?
- Which similarity measure between trips shall be selected in clustering process?
- Which clustering method shall be used to group the trips?
- How many clusters shall the clustering produce? (What is the optimal number of clusters?)
- How to compute the deviation area between two trajectories in a given cluster?
- How to select the least-cost (referential) paths in each pair of zones?
- How to investigate the relationship between deviation area and average speed? And which regression method to use?

1.3 Expected Contributions

It is expected that this study and its findings will contribute both methodologically and practically as follows:

- This research presents a spatial clustering of trips in a trajectory dataset based on their start and end points, where each resulting cluster represents an origin-destination (OD) pair. These OD pairs can be utilized for different uses, such as estimating OD matrices.
- Criteria are developed in this thesis to help in determining the optimal number of spatial trip clusters, especially when the statistical methods fail to provide meaningful optimal number of clusters in this context.
- A new tool is introduced in this thesis that aims to determine the deviation area between two vehicle trajectories. This tool is expected to have several applications in different fields. For instance, this tool might help better understanding drivers' travel behavior (e.g., route choice). In addition, it can be employed as a part of the performance measures to determine the level of service of roads in a network.

1.4 Research Framework

This thesis consists of four main phases; Preliminary Study, Data Modeling, Data Analysis, and Conclusion. A framework is developed to proceed through the research systematically. Figure 1.1 illustrates the followed framework.

Phase 1: Preliminary Study phase comprises several steps including reviewing the literature and describing the existing problem that is needed to be addressed. This step generates the research questions which the research aims to answer, followed by the research goals and objectives. Simultaneously, Phase 1 contains raw data exploring and preprocessing in preparation for modeling and analysis phases.

Phase 2: Data Modeling phase consists of examining and testing different data clustering methods on a sample of the preprocessed data in order to select the method that gives the best results, spatially clustering dataset's trips based on the location of their origin and destination points using the selected clustering method via R software, and developing criteria to choose the optimal number of clusters.

Phase 3: Data Analysis phase aims to understand the travel behavior in the study area through applying an innovative analysis factor called Relative Deviation area (RDA) on the resulting clusters. This phase also contains investigating the relationship between this factor and some attributes of dataset's trips and especially trip average speed.

Phase 4: Conclusion phase synthesizes and discusses the research findings, describes the limitations of this research, refers to the research questions attempting to answer them, suggests potential future research, and indicates real-world insights and practical implications of this research.

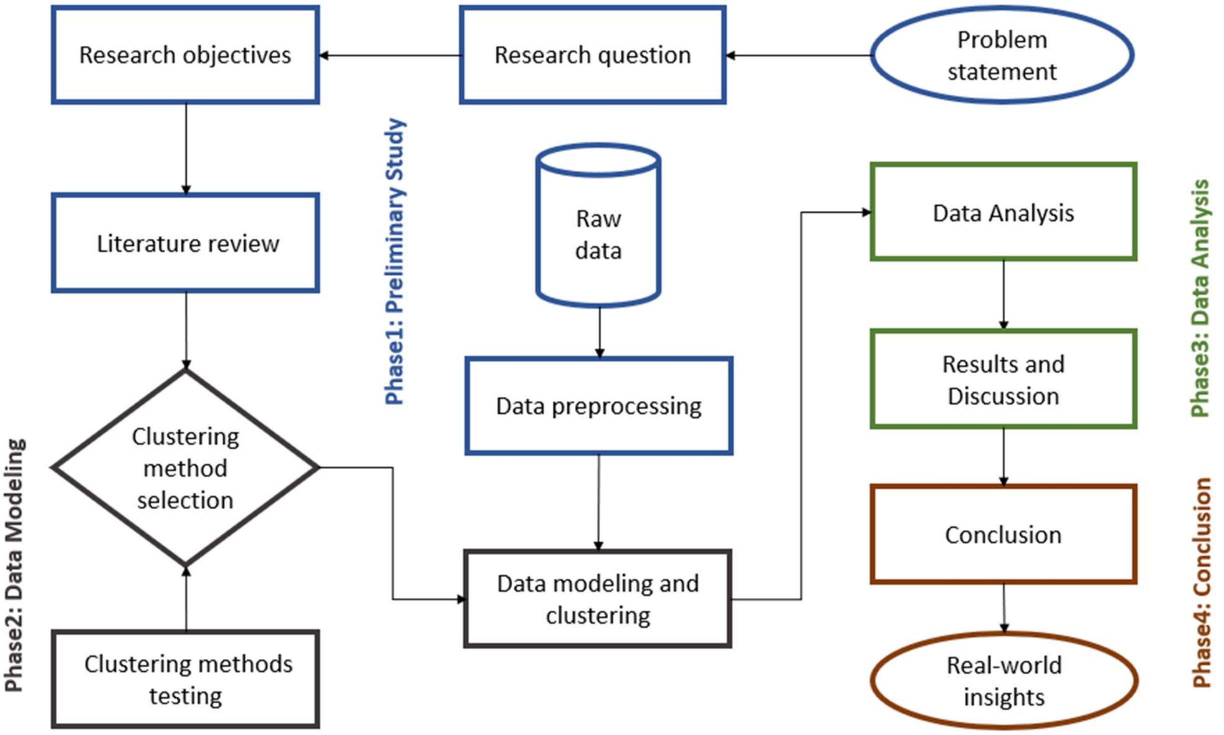


Figure 1.1: Research framework

1.5 Report Structure

The rest of the report is organized as follows. Chapter 2 presents a detailed overall literature review on FCD, spatial clustering applications in transportation, and travel behavior research. Chapter 3 describes the structure of the raw data and explains how it has been preprocessed and prepared to be used in the following chapters. Chapter 4 presents data modeling and analysis including clustering process and developing and applying RDA tool on the resulting clusters. Chapter 5 shows and discusses the results of the modeling and analysis step which include investigating some travel behavior relationships and characteristics. The study's key findings as well as suggestions for further research are concluded in chapter 6.

2. Literature Review

In this chapter, related work is reviewed and analyzed in order to acquire a deeper understanding of the problem in hand, and find the most appropriate methodologies to identify gaps in the literature and guide towards the development of methods that answer the research questions. This chapter is divided into three sections, section 2.1 describes FCD technique and its applications in transportation field. Section 2.2 covers spatial clustering of FCD, whether Origin-Destination (OD) points clustering or whole trajectory clustering, and the main approaches used in this domain. The related work on understanding and inferring urban travel behavior and mobility patterns using different analysis tools is summarized and covered in section 2.3.

2.1 FCD Concept and Applications

2.1.1 Background and Concept of FCD

Accurate and reliable real-time traffic information is the foundation of most Intelligent Transportation Systems (ITS) applications, and a crucial part of traffic management and control especially on urban roads. Traditionally, a variety of traffic data sources such as inductive loops, road tubes, video processing, radar, and bluetooth have been used to estimate traffic parameters such as link occupancy, average speed and corridor density. However, these methods have some limitations. For instance, an inductive-loop sensor can only provide traffic estimation in a certain road segment which is not an accurate representative for all road segments. Moreover, installation of inductive-loop sensor requires breaking up road surface. Finally, these sensors are relatively expensive (Burnos et al., 2007). Video image detection has a main disadvantage of reduced visibility where the detection rate depends on the ambient lighting and meteorological conditions (Leduc, 2008).

More recently, FCD has become another essential traffic data source and has an increasing usage due to its lower cost and higher coverage (Altintasi, Tuydes-Yaman, & Tuncay, 2017). FCD is mainly used to determine the status of traffic, average speed and travel time on roads based on the accumulated collection of vehicles' positions, instantaneous speed, time information, and direction of travel. The principle of FCD is to collect real-time traffic data by locating the vehicles via mobile phones or GPS over the entire road network, which means that every vehicle acts as a sensor for the road network (Ayala, Lin, Wolfson, Rishe, & Tanizaki, 2010).

FCD can be collected at a relatively low cost and can provides up-to-date and high-quality information in ITS (Huber, Lädke, & Ogger, 1999). In contrast to the traditional traffic data collection methods, FCD needs less maintenance and it doesn't require installation of any additional infrastructure or hardware on the road network.

Basically, there are two main types of FCD, namely GPS and cellular-based systems:

GPS-based FCD

Higher percentage of vehicles have been equipped with GPS system as it is becoming more and more useful and practical. GPS system utilizes the GPS receiver system, which is already attached to the vehicle, to gather information about it through transferring the data to the service provider (Wang, 2015). Therefore, the system can locate the exact location and movement of that specific vehicle and calculate the instantaneous speed for example. GPS probe data is widely used as a source of real-time traffic information by many service providers, but it suffers from the high equipment costs compared to floating cellular data (Leduc, 2008).

Cellular-based FCD

The mobile phone location data is regularly transmitted to the cellular network, and thus travel times and further data can be estimated over a series of road segments before being converted into useful information by traffic centers (Leduc, 2008). In contrast to GPS-based system, no special device/hardware is necessary as every mobile phone acts as a sensor. However, more complicated algorithms are required to extract high-quality information.

2.1.2 FCD Applications

FCD has some disadvantages including not providing direct information on traffic flow or density, infrequently sampling and occurring at irregular intervals, as well as involving potential privacy issues (Jones, Geng, Nikovski, & Hirata, 2013). However, FCD is an efficient emerging technology that collects accurate real-time traffic information covering an entire road network at a relatively low cost. These characteristics make this technology gain in popularity for the provision of data for traffic control and management systems.

There are several applications that can benefit from the implementation of FCD, especially in transportation field. Congestion monitoring, OD matrices estimation, incident management, traffic queue detection, and dynamic route guidance are some examples of the potential application areas.

There have been many conducted studies that have leveraged FCD technology in such areas. Many studies benefited from FCD in traffic control and management applications. For instance, Kerner et al. (2005) presented a method to estimate traffic state and detect incidents on road networks using FCD. The presented method can detect incidents with at least 20 minutes duration with a probability of 65% and a penetration rate of 1.5% FCD vehicles out of the whole number of vehicles. Brockfeld, Lorkowski, Mieth, & Wagner (2007) tested travel time data extracted from taxi-based FCD system, using about 500 taxis, with estimated travel time data obtained from a 4-day measurement campaign conducted along a main street in Nuremberg, Germany using license plate recognition technology. The study found that FCD system is particularly able to detect jammed situations, and the travel times calculated by the system deliver valuable data for mobility and traffic information systems. However, the authors suggested data fusion with locally continuous detecting sensors like inductive loops for applications like real-time traffic state detection and controlling traffic lights in real-time in order to improve the system performance that might be affected by the stochastic coverage by FCD.

Some studies applied FCD to integrate urban mobility patterns with land use applications. Liu, Biderman, & Ratti (2009) used multiple real time data sources including 5,000 floating car GPS data in South China for the real time evaluation of urban mobility dynamics. The research provided accurate and dynamic method to understand daily urban mobility patterns and explore the relationship between mobility and land use on the one hand, and between mobility and social-economic changes on the other.

Other studies focused on deriving mobility patterns and investigating travel behavior using FCD. For example, Ding, Jahnke, Wang, & Karja (2016) used FCD to analyze the spatio-temporal mobility patterns of transport hubs such as airports and railway stations. As a test dataset, the authors used one-week FCD in Shanghai, China to uncover the mobility patterns related to Hongqiao International Airport, attempting to understand passenger's travel behavior. The research results showed that there are obvious hourly and daily temporal mobility patterns as well as significant spatial hotspots related to transport hubs. In this thesis, FCD is used to infer travel behavior and spatial and temporal mobility patterns within the middle ring of Munich city. In contrast to Ding et al. (2016), the whole study area is divided into smaller pairs of OD zones, and then the travel behavior of passengers within each OD pair (cluster) is investigated.

2.2 Spatial Clustering

2.2.1 Background and Concept of Spatial Clustering

Clustering is the process of grouping a set of objects into groups or clusters in such a way that objects in the same group (cluster) are more similar to each other than to those in the other groups (clusters) (Ghuman, 2016). Cluster analysis has been studied in the field of machine learning as a kind of unsupervised learning since it learns from test data that has not yet been labeled, classified, or categorized. As a branch of statistics, clustering has also been studied extensively for many years. However, efforts to perform efficient clustering on large datasets only started in recent years due to the emergence of Big Data (Wang, 2016).

In spatial clustering, the objects to be grouped have certain dimensions and coordinates. Spatial clustering is an essential part of spatial data mining as it provides certain insights into the distribution of data and characteristics of the resulting spatial clusters (Neethu & Surendran, 2013).

Spatial clustering methods are mainly categorized into four methods: hierarchical, partitioning, density-based, and grid-based (Neethu & Surendran, 2013). All these categorizations are based on the specific criteria used in grouping similar objects. Many factors shall be considered before deciding on the clustering method to be used in a particular application. Such factors include application goal, desired clustering quality and speed, type and dimensionality of the data, and amount of noise in data (Wang, 2016).

The following is a brief definition of these clustering methods, noting that they are covered in more detail in chapter 4.

Partitioning Methods

In partitioning methods, the data is considered as a one big group which then will be divided into certain predefined numbers of clusters (K). Therefore, K centers are initially selected in a random way and each object is assigned to the nearest cluster center. Next, objects are relocated from one cluster to another such that the sum of squared errors is minimized (Maimon & Rokach, 2005).

Hierarchical Methods

These methods build nested clusters by recursively partitioning data objects in either a bottom-up (agglomerative) or top-down (divisive) form (Maimon & Rokach, 2005). The output of hierarchical clustering is a tree diagram called dendrogram, which illustrates the arrangement of the produced clusters.

Density-based Methods

Density-based clustering locates regions (neighborhoods) of high density that are separated from one another by regions of low density, where the density of a region is measured by the number of objects that belong to it. The key idea is that the density of a neighborhood with a given radius has to exceed some threshold (Moise & Pournaras, 2017).

Grid-based Methods

In grid-based methods, the space is divided into a finite number of cells first. Next, cells which contain more than a certain number of objects are considered as dense areas (Wang, 2016).

2.2.2 Spatial Clustering in Transportation

With the continuous progress in information and communication technologies in the past few decades, ITS have compiled massive amounts of data regarding the movement of people and goods (Lin, 2015). In addition to the traditional traffic data sources, new emerging sources and approaches such as FCD, social media, and crowdsourcing can be used to extract traffic information. To take advantage of all these data and to address the associated challenges, big data techniques are currently receiving increasing attention. Such techniques employ theories and tools from many fields such as statistics, machine learning, data mining, analytical models, and computer programming to solve the data analysis task (Lin, 2015). Therefore, it is vital to explore how big data techniques may be best employed for the analysis of transportation data.

Clustering, typically spatial clustering, is the most common big data technique used in transportation data analysis (Anand, Padmanabham, Govardhan, & Kulkarni, 2017). It has several applications in transportation engineering. Such applications include dynamic traffic forecasting, spatio-temporal mobility patterns detection, pavement management, accident analysis, transport and urban planning, traffic and congestion management, and hotspot recognition.

Several studies have applied spatial clustering in traffic management and control field. For instance, Liu & Ban (2013) used over 85 million taxicab GPS points (floating car data) collected in Wuhan, China to measure the degree of traffic congestion within the road network through spatio-temporal clustering of the low-speed GPS points. Zamani, Pourmand, & Saraee (2010) applied hierarchical clustering to aid in the development of traffic control by automatically generating traffic signal timing plans.

Other studies leveraged clustering algorithms in accidents analysis field. Dogru & Subasi (2014) tested different cluster analysis techniques to detect traffic accidents using vehicles' velocity and location values. Results of this study showed that Density-Based Spatial Clustering of Applications with Noise (DBSCAN) and hierarchical clustering methods successfully detected accidents with average accident detection rate of 100%.

Spatial clustering methods have been widely used in hotspot detection applications. For example, Yue, Zhuang, Li, & Mao (2009) employed taxi trajectory data collected in Wuhan, China to discover attractive areas in terms of frequency and density of passenger pick-up and drop-off events. The authors used hierarchical clustering method to spatio-temporally group the similar pick-up and drop-off points. According to Yue et al. (2009), hierarchical clustering was chosen over K-means (a common partitioning clustering method) since the latter requires a pre-knowledge of clusters number and the shape of all clusters is convex.

Another important application of clustering analysis in transportation engineering is detecting mobility patterns. Some studies derived mobility patterns by spatially clustering trips using their whole trajectory, while other studies clustered trips using only their start (origin) and end (destination) points. Most of the studies that performed clustering on whole trajectories had developed new clustering algorithms or modified existing algorithms as the traditional clustering methods are usually applicable for points only. Chen, Hu, Zhang, & Shi (2014) provided an improved DBSCAN clustering algorithm in which GPS trajectories are partitioned into a group of line segments that are used to find out individual clusters with similar track segments. The output of the algorithm is discovering the spatial distribution and temporal evolution characteristics of people's stay hot spots from their GPS trajectories data. Kumar et al. (2018) proposed a new clustering algorithm based on a novel Dijkstra-based dynamic time warping distance measure, which is suitable for extracting urban traffic patterns from large numbers of overlapping trajectories in dense road networks. Kim and Mahmassani (2015) introduced a trajectory clustering method to discover spatial and temporal travel patterns in a traffic network. The output of the proposed clustering approach is some spatially distinct traffic stream clusters, which together represent the major network traffic streams. The authors applied extended DBSCAN algorithm that uses Longest Common Subsequence (LCS) tool as a similarity measure among trajectories. LCS is an algorithm for finding the longest common subsequence of two sequences. It originates in the field of string matching, where two strings with different lengths are given to find a set of characters that appear left-to-right, not necessarily consecutively, in both strings (Bergroth, Hakonen, & Raita, 2000). LCS has been used as a similarity measure between two time series or trajectories in several studies. The similarity between two trajectories is usually expressed as the number of matched elements between two sequences as illustrated in Figure 2.1. There are several ways for determining whether two points in two sequences are matched or not. One way is to consider spatial proximity only and another way is to consider both spatial and temporal proximity together (Kim and Mahmassani, 2015).

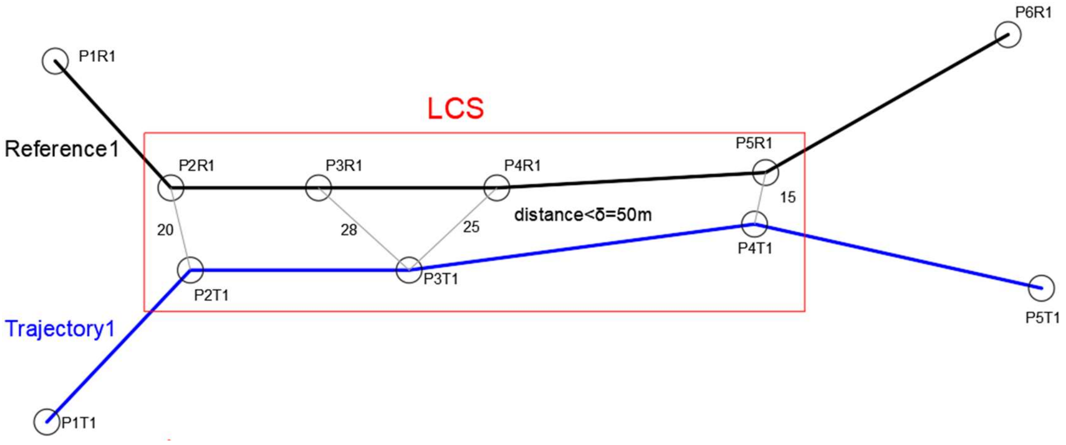


Figure 2.1: Concept of LCS tool in vehicle trajectories

On the other hand, several studies detected mobility patterns through spatial clustering of trips' origin and destination points. Guo et al. (2012) spatially clustered origin and destination points of taxi trajectories collected in Shenzhen, China to extract traffic flow measures, location patterns, and temporal structures embedded in the resulting clusters which represent the meaningful places. The authors applied hierarchical clustering method based on the Shared Nearest Neighbors (SNN) as a distance measure. The study excluded DBSCAN and K-means methods as the former is unable to identify clusters of different point densities, while the latter tends to assume a predetermined circular shape to find clusters. Ding, Meng, Yang, & Krisp (2018) propose a visual analytics approach for the exploration of spatiotemporal interaction patterns of massive OD data. The authors used hierarchical clustering of OD floating car data from Shanghai, China to extract OD hotspots and their spatial interaction patterns. In 2016, Kumar, Wu, Lu, Krishnaswamy, & Palaniswami applied DBSCAN algorithm to cluster OD pairs of large amount of passenger taxi rides collected in the city of Singapore to provide useful insight into mobility patterns and road network usage within the resulting clusters. Whereas, in this thesis, three of the most common clustering algorithms, i.e., K-means, hierarchical, and DBSCAN, are practically tested on a dataset sample, and then theoretically compared against each other in order to select the most appropriate method to cluster trips' OD points, such that each cluster presents a traffic flow from one specific zone to another within the middle ring of Munich city. Next, a novel analysis factor is computed for each resulting cluster attempting to infer passengers' travel behavior.

2.3 Travel Behavior

2.3.1 Travel Behavior Concept

Travel behavior can be defined as "the study of what people do over space, and how people use transportation" (CTI Reviews, 2016). Travel behavior tries to answer questions like: Why do people travel? What mode do they take and why? How many trips do they make? Where do they go? What route choices do they make and why? and other questions related to people's behavioral aspects of traveling and using transportation systems.

As technologies have progressed, emerging passively collected traffic information sources have been promising in helping transportation experts better understand people's movements through space and time (Rojas, Sadeghvaziri, & Jin, 2016). Traditional travel surveys have the disadvantages of low response rates, high respondent burden, and significant implementation costs (Wolf, Oliveira, & Thompson, 2003). Data from passive collection methods, such as FCD, GPS data, mobile network data, and cell phone GPS data, might be able to supplement or complement the traditional household travel surveys and overcome the existing issues (Rojas et al., 2016). These data present different opportunities to reflect aspects of people's travel behavior and mobility patterns.

Studying, understanding, and sometimes changing people's travel behavior have been a vital issue in transportation planning. Knowledge about travel behavior across transportation modes and demographics shall be considered to better predict future transportation requirements (SINTEF, n.d.). The ultimate objective of transportation planning is to design efficient and sustainable transportation infrastructures and services to meet the needs for accessibility and mobility (Montello & Goulias, 2018). Understanding human travel behavior, through applying different analytical tools and methods, is at the center of this design.

2.3.2 Travel Behavior Research and Applications

For decades, in order to provide long-term guidance and short-term strategies for transportation planning and urban development, many studies have been conducted to identify, understand and predict human travel behavior (Buliung & Kanaroglou, 2007; Yue, Lan, Yeh, & Li, 2014; French Barchers, & Zhang, 2015). Traditionally, data that support travel behavior research mainly came from travel surveys, which have numerous shortcomings that have restricted data collection and further obstructed travel behavior research progress to some extent (Mitchell, 2014; Liu, Li, Li, & Wu, 2015). In the era of Big Data, along with the development of Information and Communication Technology (ICT), various novel data sources have arisen to supplement or substitute for the traditional sources to support travel behavior research (Wang, He, & Leung, 2017). Examples include smartcard records data, floating car data, roadside sensor data, and mobile phone data.

There have been several applications of these data collected from the novel sources in the context of travel behavior research. Applications of OD matrix, travel patterns, choice model, and traffic monitoring are examples of such applications (Rojas et al., 2016; Wang et al., 2017).

Several studies were able to leverage the available data to produce different facets of OD tables. Some studies proposed a method that used Call Detail Records (CDR) data to infer trips and then estimate largescale OD matrices (Fang, Xue, & Qiu, 2014; Iqbal, Choudhury, Wang, & Gonzalez, 2014; Wang, Schrock, Broek, & Mulinazzi, 2013). These studies found that the data were most effective at the aggregate level. In an attempt to obtain more-detailed information, Rokib, Karim, Qiu, & Kim (2015) used similar CDR data in combination with Foursquare check-in data to reproduce OD matrices. Through the incorporation of time of day data, Li (2015) found that it was possible to create OD sample characteristics, mobile OD flow distributions, directional patterns, flow analysis for each OD pair, and spatial analysis.

Another common application encountered in the literature was travel patterns. One of the most fundamental applications of the emerging traffic data sources in travel behavior research is to detect stays (visits) and extract trips (Wang et al., 2017). Clustering methods such as distance-based methods are usually used to identify stays and trips from mobile phone data (Ye, Zheng, Chen, Feng, & Xie, 2009). After stays are detected, trips can be extracted, and travel patterns can be identified on different scales, such as the travel frequency and distance on the individual level, and OD matrices and travel flow distributions on the aggregated level (Zhang, Hong, Nasri, & Shen, 2012; Calabrese, Di Lorenzo, Liu, & Ratti, 2011). Besides the detection of stays and trips, activities and corresponding trip types have also been frequently explored. Most studies detect activities that frequently take place and last for a considerable time period, such as staying at home and working in a workplace, and thus the most common identified trip type is home-based-work trips (Colak, Alexander, Alvim, Mehndiratta, & Gonzalez, 2015).

Route choice has been the focus of several studies in choice model application (Rojas et al., 2016). Levinson & Shanjiang (2013) used GPS data to explore the application of route choice portfolios, which had the potential to solve the traffic assignment problem. The results indicated that the participants did not have a single dominant route. Another study considered the application of general route choice models based on real-world GPS data (Tawfik & Rakha, 2012). The study had three main findings. First, the observed route choice percentages varied from those derived through the use of stochastic user equilibrium expectations but were converging to specific values. Second, four types of heterogeneous driver learning and choice evolution pattern were identified. Third, driver and choice situation variables could predict the identified learning patterns.

Spissu, Meloni, & Sanjust (2011) successfully converted GPS data into routes to characterize route choice variability and compare the least-cost route to the actual route. The authors found that discretionary trips generally displayed greater intraindividual variability, while work and study trips displayed greater interindividual variability and deviation from the least-cost routes. One paper studied the factors that influence commuters' route choice and route switching based on real-world observations of travel behavior (Sun et al., 2014). Possible factors that may affect driver's route choice were then analyzed and regression methods were introduced, but the result indicated that a relationship between route choice and these factors was difficult to be established. However, travel distance, travel time and road preference were found to have comparable higher influence on drivers' route choice than other factors. In this thesis, a novel tool is introduced which might help in detecting mobility patterns, and understanding drivers' travel behavior and route choice decision. This tool computes the area by which a vehicle, travelling from one zone to another on a specific route, is deviating from the least-cost routes between these two zones, where the least-cost route can be the fastest route (i.e., with highest average speed), the shortest distance route, or any other desired criterion.

3. Data Collection and Preprocessing

3.1 Data Description

The technology of FCD is an emerging traffic information gathering method and an essential data source for most ITS. FCD is mainly used to determine the average traffic speed and travel time on roads based on the accumulated collection of vehicles' positions, instantaneous speed, time information, and direction of travel. In this technology, traveling cars act as moving sensors that are equipped with a location detecting device, such as a GPS unit, and a communication device, such as a cellular phone (Ayala et al., 2010).

The available FCD for this thesis was collected by INRIX, which is a leading software company that specializes in connected car services and transportation analytics. INRIX provides real-time traffic information, traffic forecasts, and travel times to government agencies, businesses, and individuals in more than 37 countries including the United States and most of Europe, for a better understanding of the movement of people, vehicles and goods (INRIX, 2018).

The FCD used in the thesis was collected in the city of Munich, Germany for 6 days from Sunday, February 25, 2018 to Friday, March 2, 2018. The raw data consists of two main data frames; Trips data frame and Waypoints data frame.

Trips data frame contains trips with metadata such as trip ID, start (end) GPS coordinates, start (end) date and time, provider ID, device ID, origin and destination zone, trip length, vehicle weight class, average speed, maximum speed, provider driving profile (consumer, taxi, etc.) and other fields. The data frame comprises 303,549 recorded trips within the city of Munich (whether starts, ends, starts and ends, or just passes through Munich city).

Waypoints data frame contains full GPS waypoint data, listed trip by trip. The data includes trip ID, waypoint sequence, waypoint GPS coordinates, capture date and time, zone name, road class, and raw speed. In total, there are 10,190,930 waypoints for these 303,549 trips. Therefore, each trip in the first data frame has several waypoints in the second data frame (around 34 waypoints/trip on average) which together form the trajectory of that trip.

The overviews of the general structure of Trips data frame and Waypoints data frame are described in Table 3.1 and Table 3.2 respectively, which give a listing of the most substantial available attributes contained in the FCD of Munich.

Table 3.1: Description of the data format of Trips data frame

Field name*	**Field value**	**Units/notes**
Trip ID	00a7da9f4b503f36fc937f386b11ca58	In a serial number of 32 digits
Start Date	2018-02-25T19:24:04.000Z	UTC ISO-8601 format (yyyy-mm-ddThh:mm:ss.sssZ)
Start Loc Lon	11.573	In decimal degrees; accurate to the 3rd decimal place
Start Loc Lat	48.178	In decimal degrees; accurate to the 3rd decimal place
Geospatial Type	EI	II for internal to internal, IE for internal to external, EI for external to internal, EE for external to external
Trip Mean Speed	31.5709577312315	In kilometer/hour (kph)
Trip Distance	7498.10246116749	In meters (m)

** According to INRIX*

Table 3.2: Description of the data format of Waypoints data frame

Field name*	**Field value**	**Units/notes**
Trip ID	00ecb452acfaa03053a3827d3418fa20	In a serial number of 32 digits
Waypoint Sequence	0	Waypoints order starting from 0 and incrementing by 1
Capture Date	2018-02-27T16:35:46.000Z	UTC ISO-8601 format (yyyy-mm-ddThh:mm:ss.sssZ)
Longitude	11.56688	In decimal degrees; accurate to the 5th decimal place
Latitude	48.18158	In decimal degrees; accurate to the 5th decimal place
Raw Speed	20	In kilometer/hour (kph)

** According to INRIX*

The following is the detailed description of each attribute in each data frame:

1) Trips data frame:
   - Trip ID: a trip's unique identifier.
   - Start Date: the trip's start date and time in UTC, ISO-8601 format.
   - End Date: the trip's end date and time in UTC, ISO-8601 format.
   - Start Loc Lon: the longitude coordinates of the trip's start point in decimal degrees.
   - Start Loc Lat: the latitude coordinates of the trip's start point in decimal degrees.
   - End Loc Lon: the longitude coordinates of the trip's end point in decimal degree.
   - End Loc Lat: the latitude coordinates of the trip's end point in decimal degrees.
   - Geospatial Type: describes the trip's geospatial intersection with the middle ring zone of Munich (II - Internal to Internal: trips that start and end within the middle ring zone; IE – Internal to External: trips that start within the middle ring zone and end outside of it; EI – External to Internal: trips that start outside of the middle ring zone and end within it; EE – External to External: trips that start and end outside of the middle ring zone, but were selected because of an intersection with it).
   - Trip Mean Speed: the average speed of the trip in kph. It is computed by dividing total trip length by total travel time.
   - Trip Distance: the total length of the trip in meters.

2) Waypoints data frame:
   - Trip ID: a trip's unique identifier.
   - Waypoint Sequence: the order of the waypoint within the trip starting with "0" and incrementing by one.
   - Capture Date: the capture date and time of the waypoint in UTC, ISO-8601 format.
   - Longitude: the decimal degree longitude coordinates of the waypoint.
   - Latitude: the decimal degree latitude coordinates of the waypoint.
   - Raw Speed: the instantaneous trip speed in kph.

Figure 3.1 shows the spatial distribution of the origin and destination points for all trips, where origin points are in blue, while destination points are in red. Most of the points are concentrated in Munich city. However, a considerable percentage of the points are located in other German cities or in the neighboring countries of Germany.

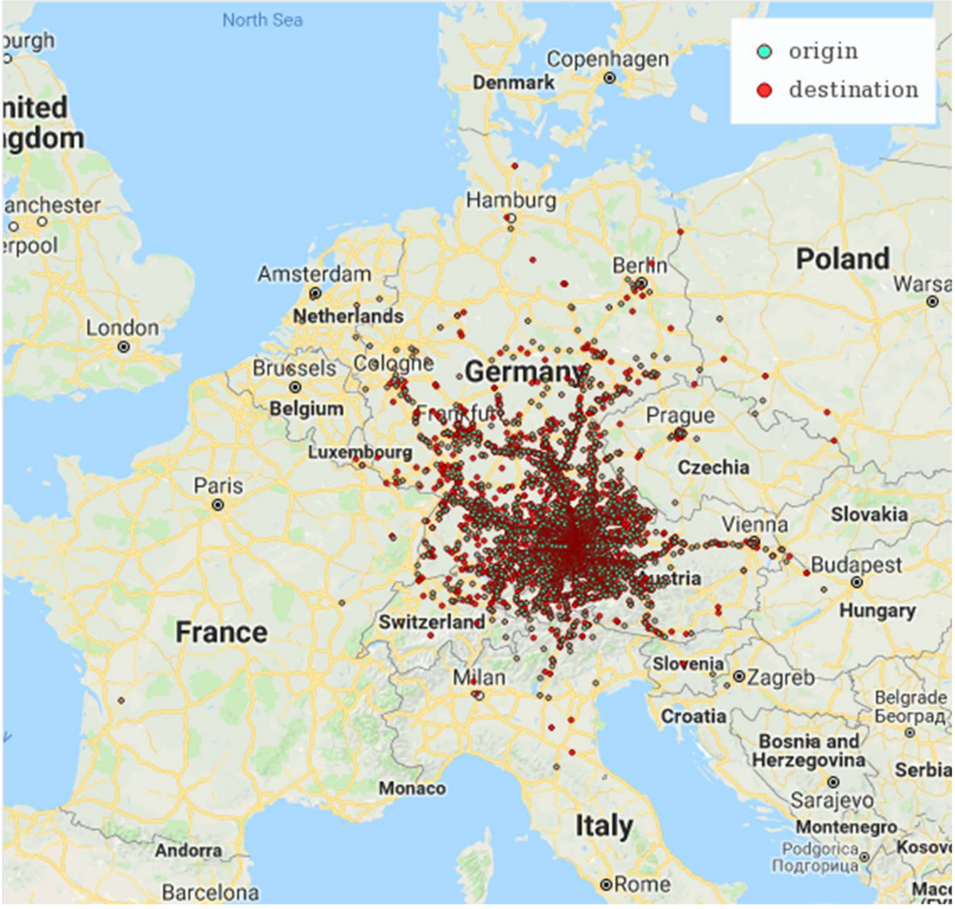


Figure 3.1: Spatial distribution of origin and destination points

3.2 Data Limitations:

Raw dataset has a few limitations that resulted in narrowing down the scope of the thesis. The main limitations are as follows:

I. Although the available dataset contains a considerable amount of the trips occurred in Munich city during the survey period, it does not represent all trips as it contains only the trips surveyed by INRIX.

II. Around 90% of all surveyed vehicles have only one recorded trip, as shown in Figure 3.2. As a result, the scope of the study is shifted from focusing on mobility patterns for individuals to focusing on mobility patterns on a trip-level.

III. As can be noticed from Figure 3.1, there are a lot of outlying trips that started or ended far away from Munich city. It was concluded, during data modeling process, that these trips have a direct negative impact on clustering results. As a result, only internal to internal (II) trips (i.e., trips that start and end within the middle ring of Munich) are considered in this thesis. There are 106,400 II trips which form almost 35% of all trips. Consequently, the middle ring zone is selected as the study area for this thesis, which is shown in Figure 3.3. Furthermore, this way mobility patterns and travel behavior within the middle ring zone can be further focused on, considering that this area is usually known to be suffering from serious traffic congestion problems.

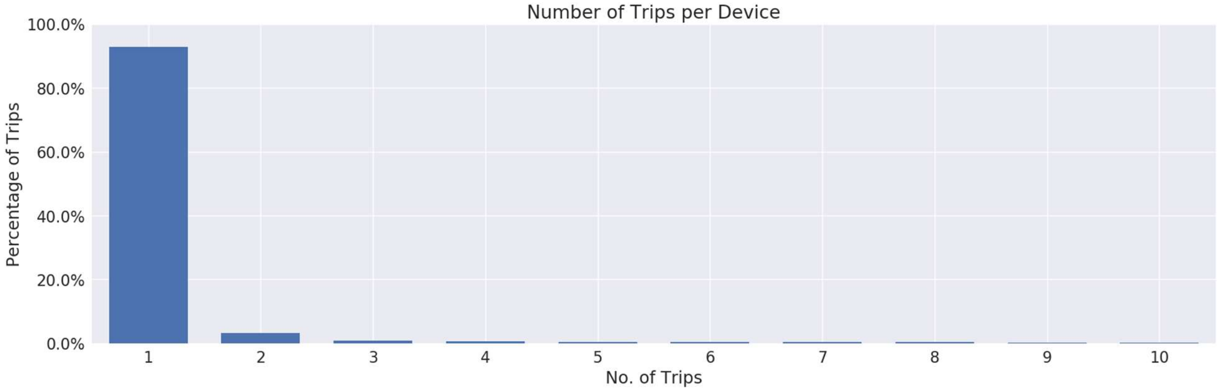


Figure 3.2: Number of surveyed trips per vehicle

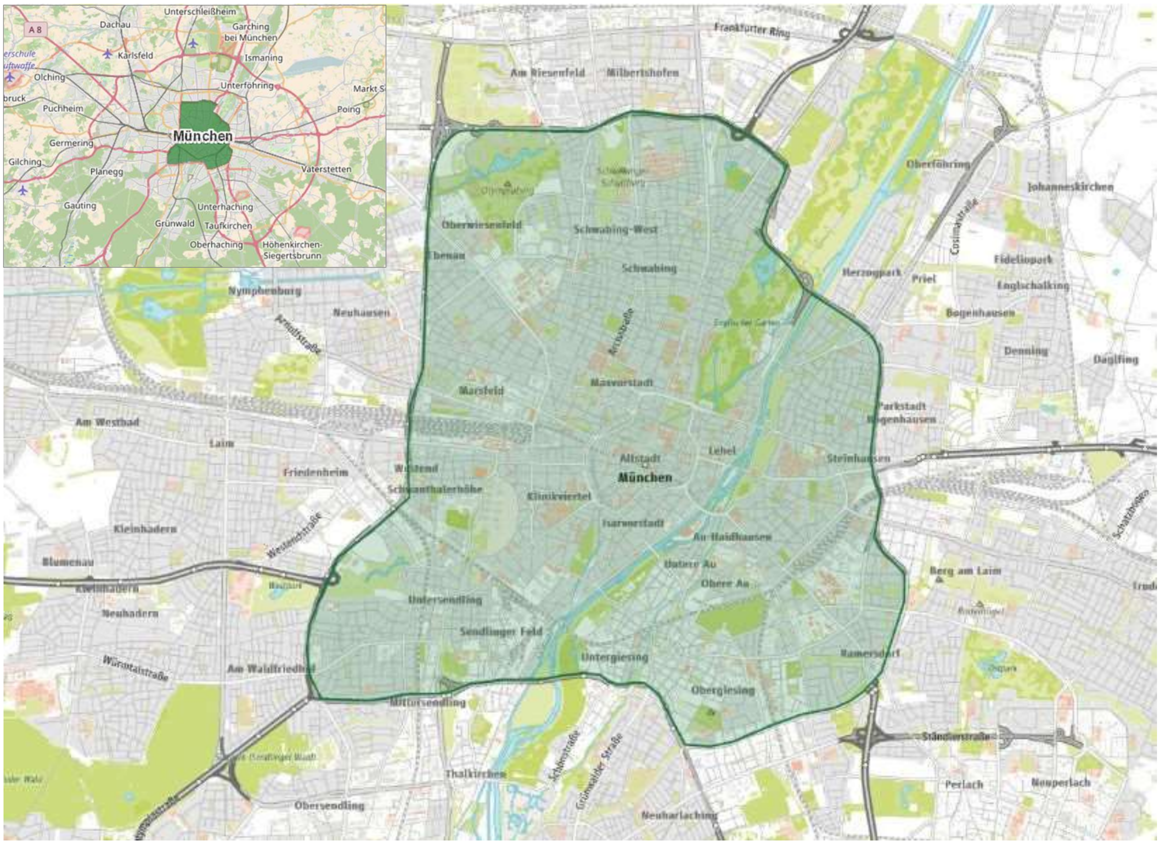


Figure 3.3: Study area

3.3 Data Preprocessing

Before using the available data and proceeding with data modeling and processing, there is a necessity to prepare and preprocess the raw data first, since real-world data is often incomplete, inconsistent, and/or lacking certain behaviors or trends, and is likely to contain some errors. Hence, data preprocessing is a proven technique to resolve such issues. Data preprocessing aims at improving the quality of raw data and, consequently, the quality of mining results, and preparing it for further analysis (Jambhorkar & Jondhale, 2015). Figure 3.4 shows the data preprocessing technique used in this thesis.

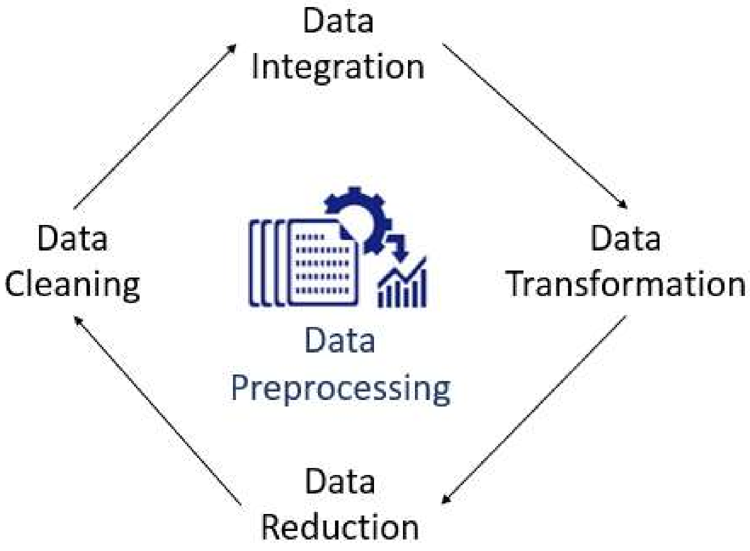


Figure 3.4: Data preprocessing technique used in the thesis

This technique consists of four steps; data cleaning, data integration, data transformation, and data reduction. Of course, these steps are not mutually exclusive, and they might be applied simultaneously. All data preprocessing techniques and steps in this thesis are applied using Microsoft Excel spreadsheet and R software.

In the following, each step is briefly described and what is done to raw data within each step is outlined:

1) Data cleaning: this step is applied to remove noise from raw data and to correct the inconsistences by filling in missing values, smoothing outlier values, and resolving blunders (Jambhorkar & Jondhale, 2015).

Raw data was checked and found to be plausible, almost clean, and well-organized in general. For instance, all values of all attributes in both data frames are checked and verified that they are not noisy and lie within a reasonable range; all start (end) dates and times are within the data collection timeframe (from February 25, 2018 to March 2, 2018), all longitude and latitude coordinates are in Germany or the neighboring countries, all speed values are positive, within the range from 0 to 200 kph, and follow the commonly observed skew distribution as illustrated in Figure 3.5, and the same applies for all other attributes.

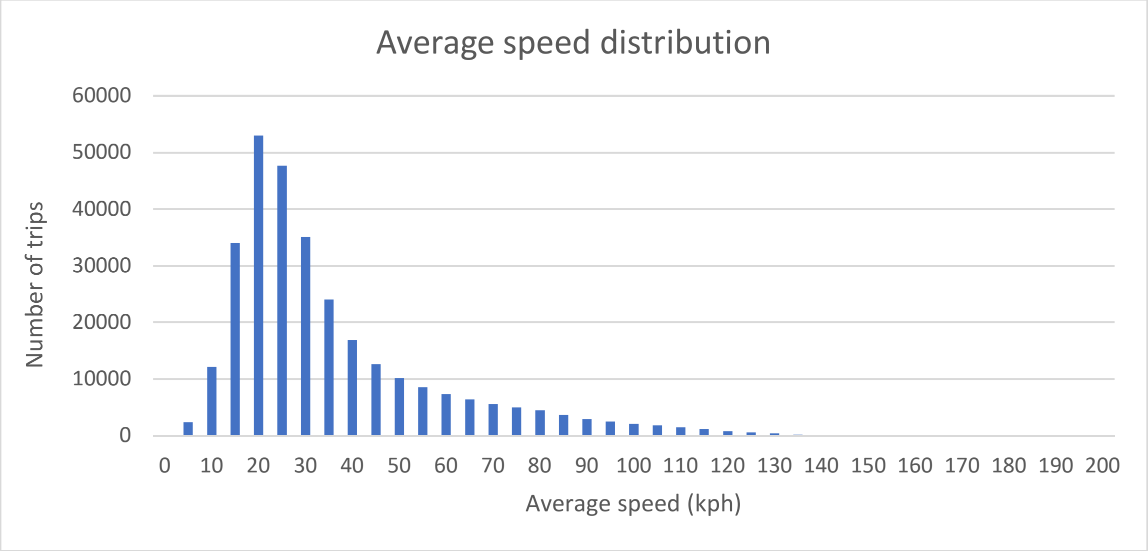


Figure 3.5: Average speed distribution of all trips

However, in Waypoints data frame, all attributes for consecutive 71,056 trips found to be completely missing. As all attributes are missing, it is not possible/feasible to fill in the missing values, and thus all attributes for these trips are totally ignored and deleted from both data frames (Trips data frame and Waypoints data frame). As a result, number of trips in the dataset is modified from 374,605 to 303,549 trips only, where the latter number is the one mentioned in section 3.1.

2) Data integration: this step is used to merge data from multiple sources into a coherent database. Data with different representations are put together, and conflicts within the data are resolved (Jambhorkar & Jondhale, 2015).

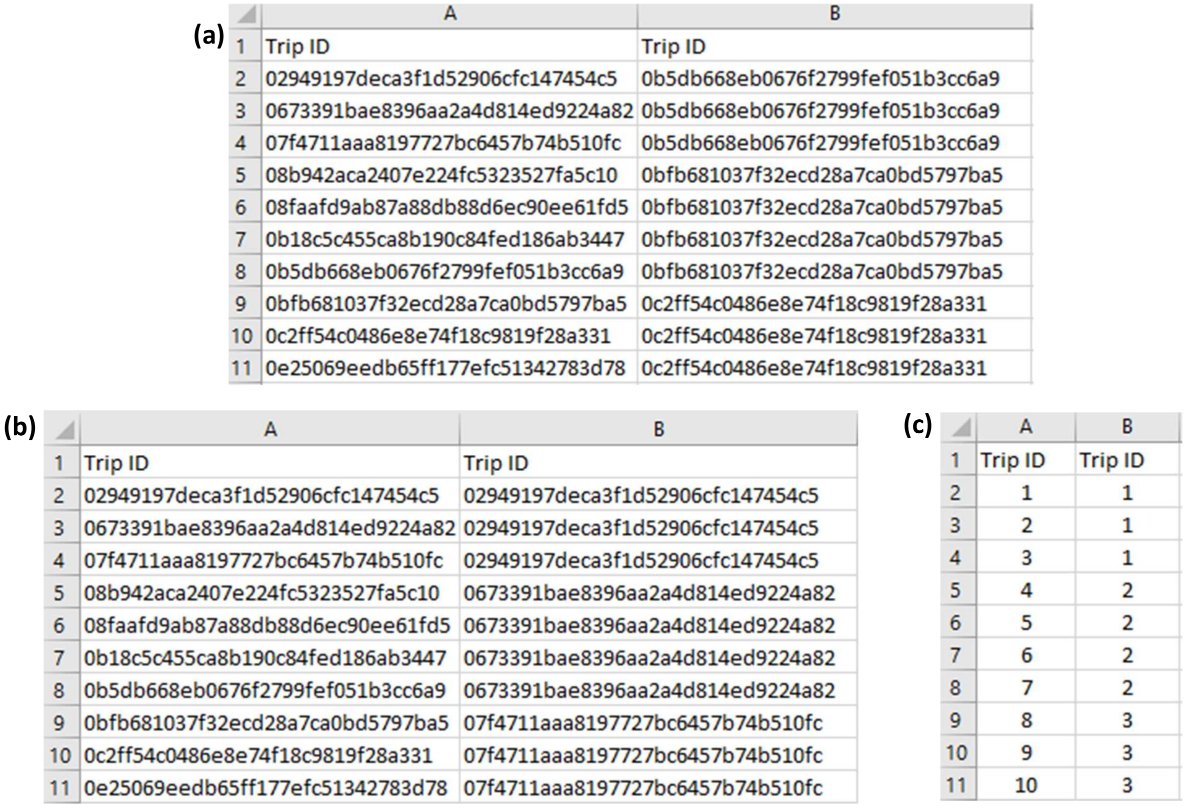
   In this thesis, Trips data frame and Waypoints data frame are merged into one coherent database by matching the common attribute "Trip ID" in both data frames.

3) Data transformation: in this step, data is converted from one form into another which is appropriate for mining process. Data transformation is applied to aggregate, generalize, or normalize data. For example, normalization is usually used to improve the efficiency of data mining algorithms involving distance measurements (Han, Kamber, & Pei, 2012).

   In this thesis, some attributes' values are transformed into forms appropriate for data modeling and processing. For instance, Trip ID values, as indicated in section 3.1, are in the form of serial numbers of 32 digits (Figure 3.6a), which is not practical at all for coding (especially for loops). Therefore, Trip ID values/rows in Waypoints data frame are sorted to match Trip ID values in Trips data frame (Figure 3.6b). Next, identical Trip ID values in both data frames are given the same number out of a group of consecutive numbers from 1 to 303,549 (total number of trips) (Figure 3.6c). As a result, Trip ID values in both data frames are in the form of consecutive numbers -which is easier to deal with in programming- instead of the form of arbitrary long serial numbers. Figure 3.6 shows a sample of Trip ID transformation.

   As for Start (End) Date attributes, first, Trips data frame and Waypoints data frame are divided into 6 parts for each data frame; one part for each day. Each part contains all rows for that specific day only. The reason behind this division is that each day will be modeled and analyzed separately to compare their results in the end. Next, dates in Start (End) Date values in each part are deleted, and time values are normalized/transformed form the 24-hr system (hh:mm:ss) (Figure 3.7a) to values between 0 and 1; where 0 is equivalent to 00:00 and 1 is equivalent to 24:00 (Figure 3.7b). Thus, it is easier to deal with them along with the other numeric attributes, especially in clustering process. Figure 3.7 shows a sample of Start Date transformation.

   In addition, Longitude and Latitude attributes in Waypoints data frame, and Start (End) Loc Lon and Start (End) Loc Lat attributes in Trips data frame, are transformed from Geographic Coordinate System (GCS) in Decimal Degrees (DD) (Figure 3.8a) to Universal Transverse Mercator (UTM) system in meters (m) (Figure 3.8b). Therefore, the Relative Deviation Area (RDA), which is a new tool introduced in this thesis which will be presented in detail in chapter 4, can be computed in metric units. Figure 3.8 shows a sample of coordinates transformation.



(a)

	A	B
1	Trip ID	Trip ID
2	02949197deca3f1d52906cfc147454c5	0b5db668eb0676f2799fef051b3cc6a9
3	0673391bae8396aa2a4d814ed9224a82	0b5db668eb0676f2799fef051b3cc6a9
4	07f4711aaa8197727bc6457b74b510fc	0b5db668eb0676f2799fef051b3cc6a9
5	08b942aca2407e224fc5323527fa5c10	0bfb681037f32ecd28a7ca0bd5797ba5
6	08faafd9ab87a88db88d6ec90ee61fd5	0bfb681037f32ecd28a7ca0bd5797ba5
7	0b18c5c455ca8b190c84fed186ab3447	0bfb681037f32ecd28a7ca0bd5797ba5
8	0b5db668eb0676f2799fef051b3cc6a9	0bfb681037f32ecd28a7ca0bd5797ba5
9	0bfb681037f32ecd28a7ca0bd5797ba5	0c2ff54c0486e8e74f18c9819f28a331
10	0c2ff54c0486e8e74f18c9819f28a331	0c2ff54c0486e8e74f18c9819f28a331
11	0e25069eedb65ff177efc51342783d78	0c2ff54c0486e8e74f18c9819f28a331

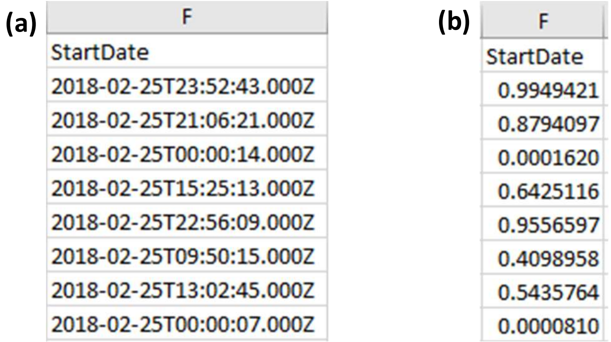
(b)

	A	B
1	Trip ID	Trip ID
2	02949197deca3f1d52906cfc147454c5	02949197deca3f1d52906cfc147454c5
3	0673391bae8396aa2a4d814ed9224a82	02949197deca3f1d52906cfc147454c5
4	07f4711aaa8197727bc6457b74b510fc	02949197deca3f1d52906cfc147454c5
5	08b942aca2407e224fc5323527fa5c10	0673391bae8396aa2a4d814ed9224a82
6	08faafd9ab87a88db88d6ec90ee61fd5	0673391bae8396aa2a4d814ed9224a82
7	0b18c5c455ca8b190c84fed186ab3447	0673391bae8396aa2a4d814ed9224a82
8	0b5db668eb0676f2799fef051b3cc6a9	0673391bae8396aa2a4d814ed9224a82
9	0bfb681037f32ecd28a7ca0bd5797ba5	07f4711aaa8197727bc6457b74b510fc
10	0c2ff54c0486e8e74f18c9819f28a331	07f4711aaa8197727bc6457b74b510fc
11	0e25069eedb65ff177efc51342783d78	07f4711aaa8197727bc6457b74b510fc

(c)

	A	B
1	Trip ID	Trip ID
2	1	1
3	2	1
4	3	1
5	4	2
6	5	2
7	6	2
8	7	2
9	8	3
10	9	3
11	10	3

Figure 3.6: Trip ID transformation sample, where Columns A and B represent Trip ID attributes in Trips data frame and Waypoints data frame respectively



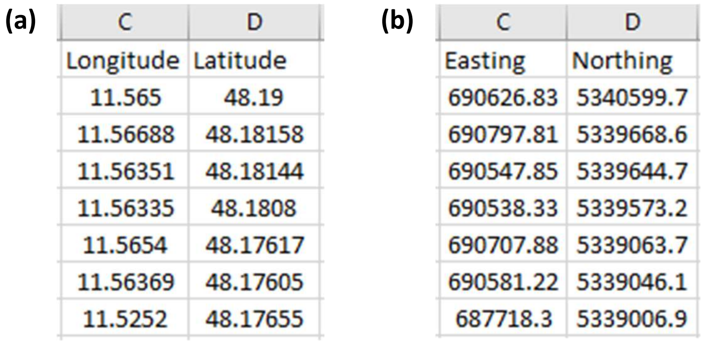
(a)

F
StartDate
2018-02-25T23:52:43.000Z
2018-02-25T21:06:21.000Z
2018-02-25T00:00:14.000Z
2018-02-25T15:25:13.000Z
2018-02-25T22:56:09.000Z
2018-02-25T09:50:15.000Z
2018-02-25T13:02:45.000Z
2018-02-25T00:00:07.000Z

(b)

F
StartDate
0.9949421
0.8794097
0.0001620
0.6425116
0.9556597
0.4098958
0.5435764
0.0000810

Figure 3.7: Start Date transformation sample



(a)

C	D
Longitude	Latitude
11.565	48.19
11.56688	48.18158
11.56351	48.18144
11.56335	48.1808
11.5654	48.17617
11.56369	48.17605
11.5252	48.17655

(b)

C	D
Easting	Northing
690626.83	5340599.7
690797.81	5339668.6
690547.85	5339644.7
690538.33	5339573.2
690707.88	5339063.7
690581.22	5339046.1
687718.3	5339006.9

Figure 3.8: Coordinates transformation sample from GCS to UTM

4) Data reduction: this step is applied to obtain reduced and smaller, in volume or number of attributes, representation of the raw data. Mining on reduced data should be faster and more efficient, yet produce almost the same analytical results (Han et al., 2012). Original dataset can be reduced by aggregating, eliminating redundant features, or clustering, for instance.

   In this thesis, raw data was reduced by spatially clustering trips based on the start and end points location (clustering is explained in more detail in chapter 4). In addition, the superfluous attributes are eliminated. The most relevant attributes used in this thesis are:

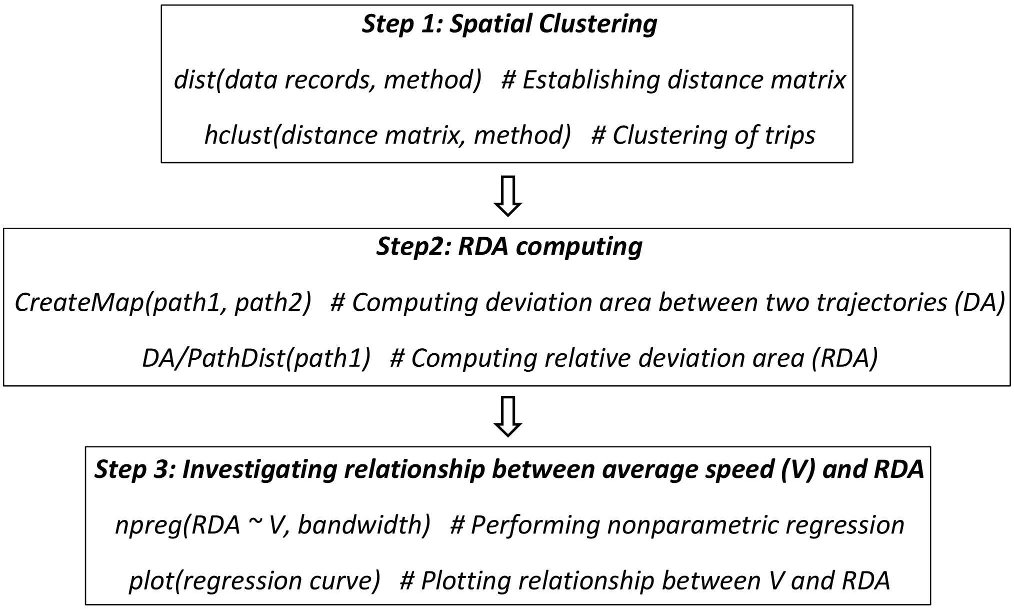
   - In Trips data frame: Trip ID, Start Date, Start Loc Lon, Start Loc Lat, End Loc Lon, End Loc Lat, Trip Mean Speed, Trip Distance, and Geospatial Type. Trip ID is used to distinguish the trips; Start Loc Lon, Start Loc Lat, End Loc Lon, and End Loc Lat are used to spatially cluster the trips based on their origins and destinations; Trip Mean Speed and Trip Distance are used to investigate the relationship between RDA and average speed; Start Date is used to divide data (based on date) into 6 parts (one for each day); and Geospatial Type is used to filter out trips with all geospatial types other than II trips (i.e., IE, EI, and EE trips).
   - In Waypoints data frame: Trip ID, Waypoint Sequence, Longitude, and Latitude. Trip ID is used to distinguish the trips; and Waypoint Sequence, Longitude, and Latitude are used to establish trips' trajectories in order to compute RDA.

   Therefore, all attributes, other than the mentioned above, are redundant, and thus, are removed from the raw dataset.

4. Data Modeling and Analysis

4.1 Approach Overview

In order to obtain meaningful output, data shall be modeled and analyzed effectively. This step is based on the preprocessed data resulted from chapter 3. Attempting to infer mobility patterns and understand travel behavior characteristics and relationships, first, data is clustered into groups based on the location of trips' start points and end points. Afterward, a novel analysis factor, called Relative Deviation Area (RDA), is computed for each trip within each cluster. At the end, the relationship between RDA and average trip speed is investigated for each day, attempting to understand people's travel behavior within the middle ring of Munich city. Figure 4.1 briefly presents the algorithm used in this thesis showing the main three data modeling steps mentioned above. These steps are covered in more detail in the following sections.



Step 1: Spatial Clustering

dist(data records, method) # Establishing distance matrix

hclust(distance matrix, method) # Clustering of trips

Step2: RDA computing

CreateMap(path1, path2) # Computing deviation area between two trajectories (DA)

DA/PathDist(path1) # Computing relative deviation area (RDA)

Step 3: Investigating relationship between average speed (V) and RDA

npreg(RDA ~ V, bandwidth) # Performing nonparametric regression

plot(regression curve) # Plotting relationship between V and RDA

Figure 4.1: Summary of the applied algorithm

4.2 Spatial Clustering of Trips

4.2.1 Clustering Methods Overview

Clustering is the task of grouping a set of objects in such a way that objects in the same group (cluster) are more similar to each other than to those in the other groups (clusters). In other words, a cluster is a closely-packed group of objects (Rohde, 2007). The main difference between classification and clustering is that in classification, the groups and number of groups are predefined (supervised learning), whereas in clustering, number of groups is unknown in advance, and therefore objects are grouped according to a selected similarity measure (unsupervised learning).

Clustering analysis is usually performed to find representatives for homogenous groups, to discover natural clusters and describe their unknown properties, to recognize useful and meaningful patterns, or to detect unusual data objects (outliers) (Ahuja, 2014).

In this thesis, trips within the study area are spatially grouped into clusters, where each cluster contains the trips travelling from one zone to another zone within the middle ring of Munich. The first question that arises while trying to do so is: Which clustering method to use?

There are several clustering methods and many clustering techniques. Clustering methods can be classified into five main approaches:

1) Density-based methods: these methods assume that points belong to each cluster are drawn from specific probability distribution, and therefore, the overall distribution is assumed to be a mixture of several distributions (Banfield & Raftery, 1993). Density-based methods produce clusters of arbitrary shape, which are not necessarily convex. The idea here is to continuously expand a given cluster as long as the density (number of objects) in the neighborhood -which has a given maximum radius- exceeds a predetermined threshold (Maimon & Rokach, 2005). Density-based methods are robust to outliers which are existing within low-density regions.

   The most popular density-based method is DBSCAN. Two parameters are to be predetermined by the user:

   - Eps: Maximum radius of the neighborhood.
   - MinPts: Minimum number of points in an Eps-neighborhood of a given point.

   The algorithm creates clusters by checking if the Eps-neighborhood of each object contains more than the MinPts (Ester, Kriegel, Sander, & Xu, 1996).

2) Partitioning methods: these methods relocate data objects from one cluster to another, starting from an initial partitioning, and then evaluate the different partitions by some

criterion, e.g., minimizing the Sum of Squared Errors (SSE) (Maimon & Rokach, 2005). Partitioning methods typically require predefining the desired number of clusters (K) by the user. These methods tend usually to create clusters with a spherical shape and similar size (Wang, 2015).

The most common partitioning algorithms are: K-means and K-medoids or PAM (Partition Around Medoids). In K-means, each cluster is represented by its center. While clusters in K-medoids are represented by one of the objects in each cluster (Zaiane, 1999).

3) Hierarchical methods: these methods build nested clusters by recursively partitioning data objects in either a top-down or bottom-up form (Maimon & Rokach, 2005). Hierarchical methods can be subdivided into two categories; agglomerative (more common) and divisive. Agglomerative hierarchical clustering, where each object initially represents a cluster of its own. Then, at each step, the two most similar clusters are merged, until at the top level all objects are joined into a single cluster (Hennig, Meila, Murtagh, & Rocci, 2015). Whereas, in divisive hierarchical clustering, all objects initially belong to one cluster. Thereafter the cluster is divided into sub-clusters, which are then successively divided into their own sub-clusters, until the desired level of clustering structure is obtained (Maimon & Rokach, 2005).

   The output of hierarchical clustering is a tree diagram called dendrogram, which illustrates the arrangement of the produced clusters. A clustering of the data objects is obtained by cutting the dendrogram at a specific desired similarity level.

4) Grid-based methods: these methods divide the space into a finite number of cells, forming a grid on which all the clustering operations are performed (Maimon & Rokach, 2005).

5) Model-based clustering methods: these methods try to optimize the fit between the database and some mathematical models. Model-based methods do not only create groups of objects, but they also find characteristic descriptions of the created groups (Maimon & Rokach, 2005). The most frequently used induction methods are decision trees and neural networks.

4.2.2 Clustering Methods Testing

In this thesis, the first three methods above are preliminary tested on a sample of the available data. Afterward, a comparison between the methods that work well in the preliminary test is conducted.

In the end, only one method is chosen to proceed with data modeling and analysis. All tests and analyses are performed using R software. The following is a description of each test of the three methods:

4.2.2.1 Density-Based Method

DBSCAN algorithm is tested on a sample (II trips sample) which is randomly taken out of II trips only. II trips sample contains 10,606 trips which accounts for 10% of all II trips. Figure 4.2 shows the spatial distribution of the trips' origin and destination points of II trips sample, before clustering, within the middle ring zone of Munich. The red dot in the figure indicates the location of Munich city center (Marienplatz).

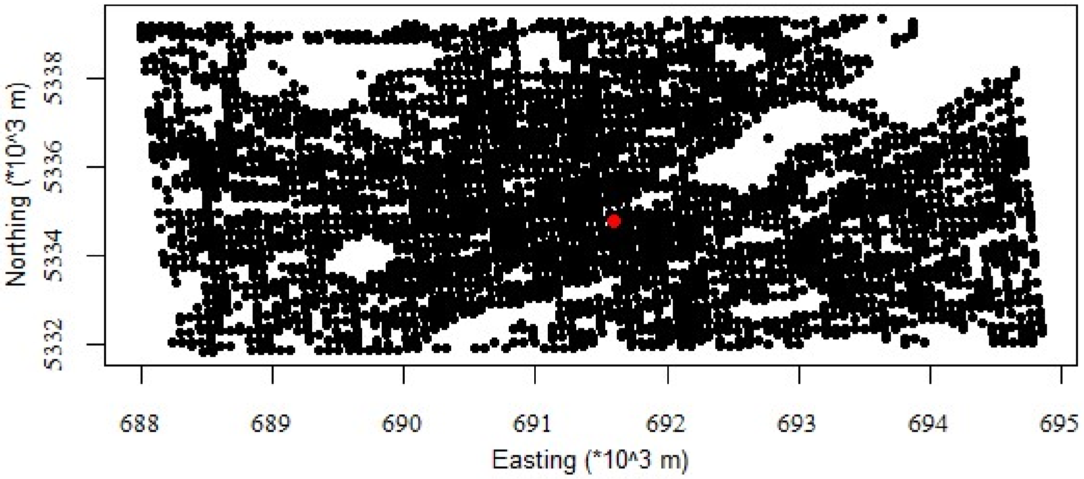


Figure 4.2: Spatial distribution of trips' origin and destination points of II trips sample

DBSCAN algorithm does not require to determine the desired number of clusters in advance. However, in order to cluster the origin and destination points of all trips sample, two parameters (i.e., Eps and MinPts) have to be determined first. According to "dbscan" R package, "MinPts is often set to be dimensionality of the data plus one or higher" (Hahsler & Piekenbrock, 2018, p. 3). In this case, there are 4 dimensions of the data to be clustered (i.e., Start Loc Lon, Start Loc Lat, End Loc Lon, and End Loc Lat). Therefore, MinPts should be 5 or higher. As for Eps, the curve knee in the k-Nearest Neighbor Distance (kNNdist) plot can be used to find suitable values for it (Hahsler & Piekenbrock, 2018). Plotting the kNNdist graph with MinPts equals 5, a value of 1,000 m can be assigned to Eps. Figure 4.3 shows the resulting kNNdist plot along with a straight line passing through the curve knee.

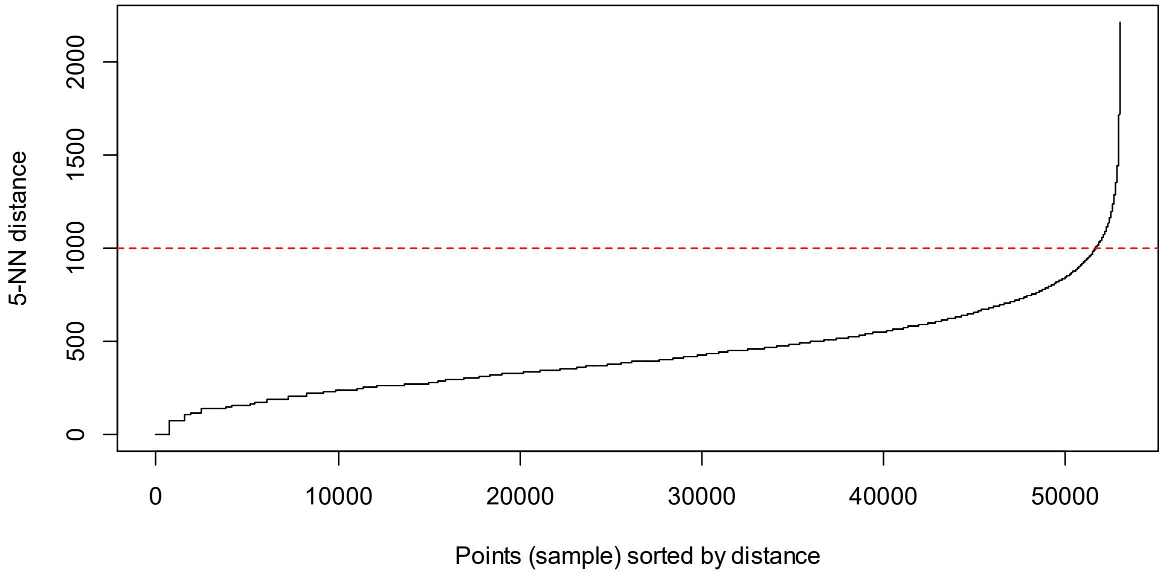


Figure 4.3: kNNdist plot to select Eps value

Therefore, MinPts parameter is set to 5 and Eps parameter is set to 1,000. Using these values, DBSCAN clustering algorithm is applied to II trips sample. The clustering resulted in 3 clusters only; one enormous cluster with 10,439 trips out of the 10,606 trips (around 98% of II trips sample size), in which the origin and destination zones cover the whole middle ring area, and another two very small clusters with 5 and 8 trips only. In addition, 154 trips are classified as outliers. Figure 4.4 and Figure 4.5 show the enormous cluster and one of the small clusters respectively, where each trip is represented by two points; the first one represents the trip start point (trip origin), and the other one represents the trip end point (trip destination). Origin points are in blue, while destination points are in red.

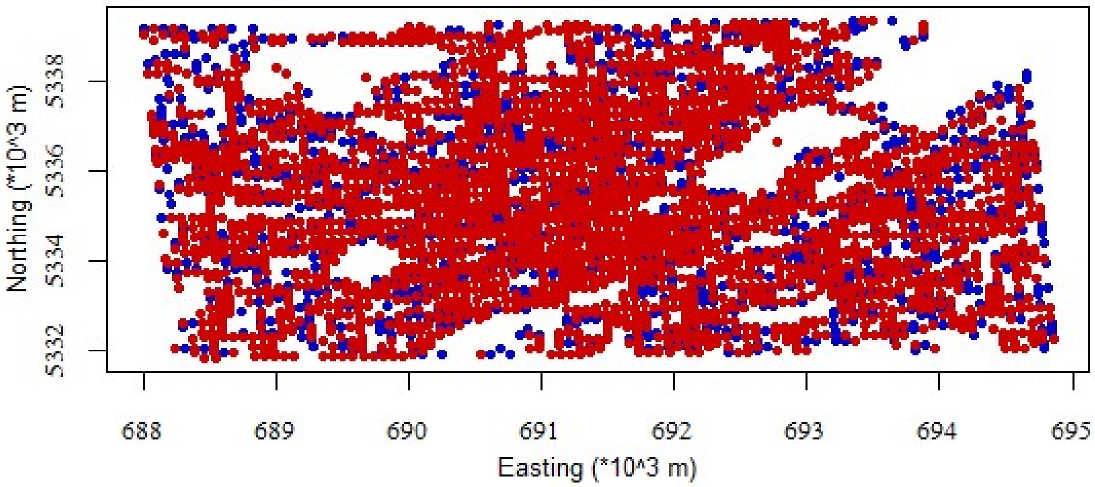


Figure 4.4: The resulting enormous cluster from DBSCAN clustering of II trips sample

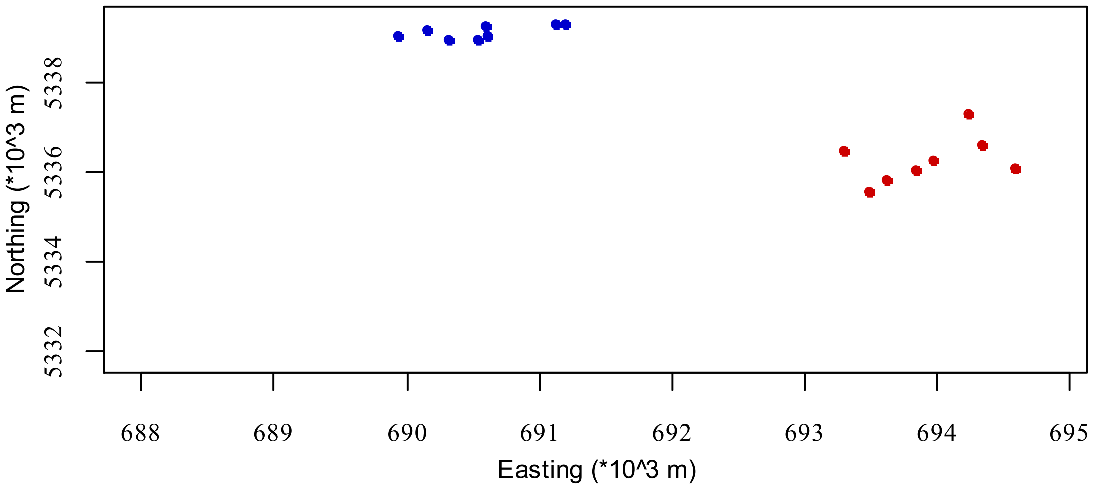


Figure 4.5: One of the small resulting clusters from DBSCAN clustering of II trips sample

From the results above, it can be concluded that it is neither feasible to compare the enormous cluster with the other small ones, nor possible to get meaningful mobility patterns out of them. Although different combinations of the two parameters are considered, the results do not differ a lot. The potential reason behind such results is that density-based clustering methods are somewhat not able to divide II area into different clusters, since the trips density is almost the same everywhere within this area. In other words, there is no big difference in trips' densities within II area, and therefore only 3 clusters are created. As a result, if dense areas—with large number of points—are desired to be partitioned into more clusters so that patterns at finer resolutions can be found, instead of grouping these points into a single cluster, then these methods are not the best choice in this case (Guo et al., 2012).

DBSCAN clustering method is also tested on another sample (all trips sample) which is randomly taken out of all trips, and not only of II trips this time. All trips sample contains 37,460 trips which accounts for 10% of all trips. Figure 4.6 shows the spatial distribution of the trips' origin and destination points of all trips sample before clustering. The red dot in Figure 4.6 indicates the location of Munich city center (Marienplatz).

Following the same procedure before, MinPts and Eps parameters are determined. MinPts is set to 5, and Eps is set to 17,000. Next, DBSCAN algorithm is applied to all trips sample. The clustering resulted in 19 clusters; again, one enormous cluster contains around 98% of all trips of the sample, and another 18 very small clusters. Figure 4.7 shows the enormous resulting cluster. Origin points are in blue, while destination points are in red.

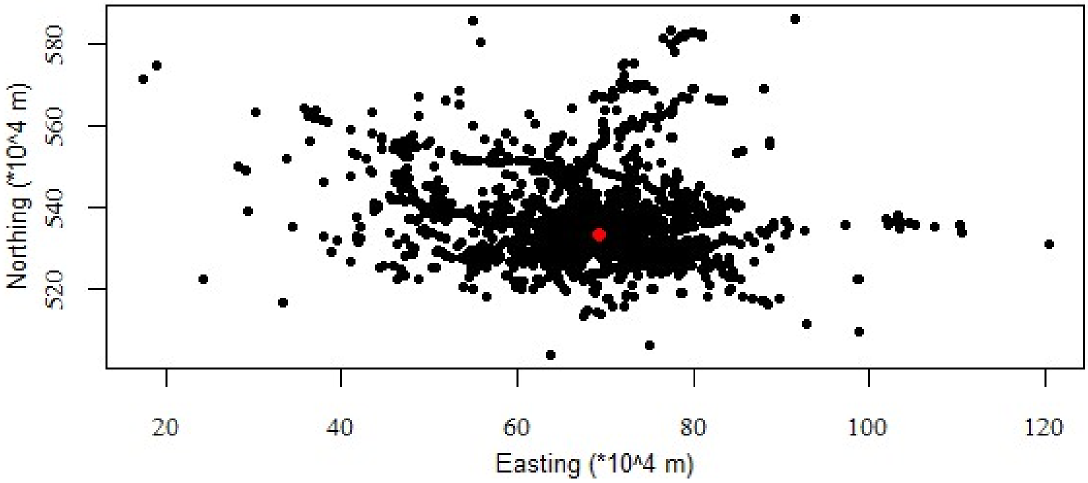


Figure 4.6: Spatial distribution of origin and destination points of all trips sample

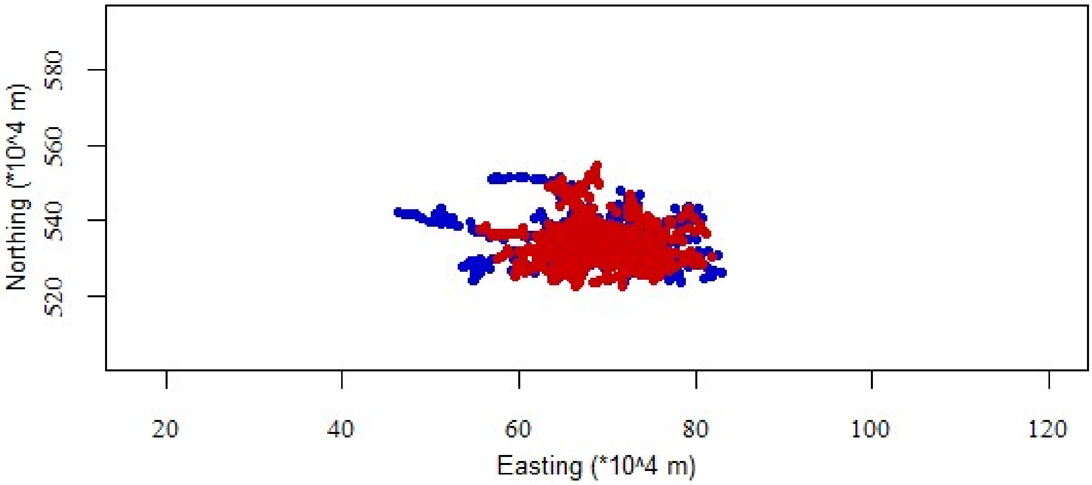


Figure 4.7: The resulting enormous cluster from DBSCAN clustering of all trips sample

Obviously, the results are not satisfactory. The potential reason behind such results is that density-based clustering methods are somewhat unable to cluster dataset with big differences in densities; setting of Eps and MinPts for identifying the neighborhood points will vary from cluster to cluster when the density varies, and thus, the parameters can't be chosen perfectly.

4.2.2.2 Partitioning Method

K-Means algorithm is applied to II trips sample. In K-Means clustering, the number of desired clusters has to be determined in advance. Selection of number of clusters is not an easy task, and depends on several factors. Some factors are related to statistical measures (e.g., elbow method and silhouette method), and others are related to application goals, characteristics of data, and meaningfulness of desired results. For this preliminary test, different numbers of clusters are considered to check whether this method is suitable for this thesis' dataset and gives meaningful results. In subsection 4.2.5, detailed criteria are developed to choose the optimal number of clusters for the clustering method that will eventually be selected.

For the preliminary test, 3 numbers of clusters are selected; 50, 100, and 200, and are tested on II trips sample. Figure 4.8 shows boxplot of the resulting clustering in terms of number of trips/cluster for the different number of clusters. Additionally, Figure 4.9 shows a sample of the resulting clusters for each case, where origin points are in blue and destination points are in red. Convex hulls are plotted to enclose the origin and destination area for each cluster.

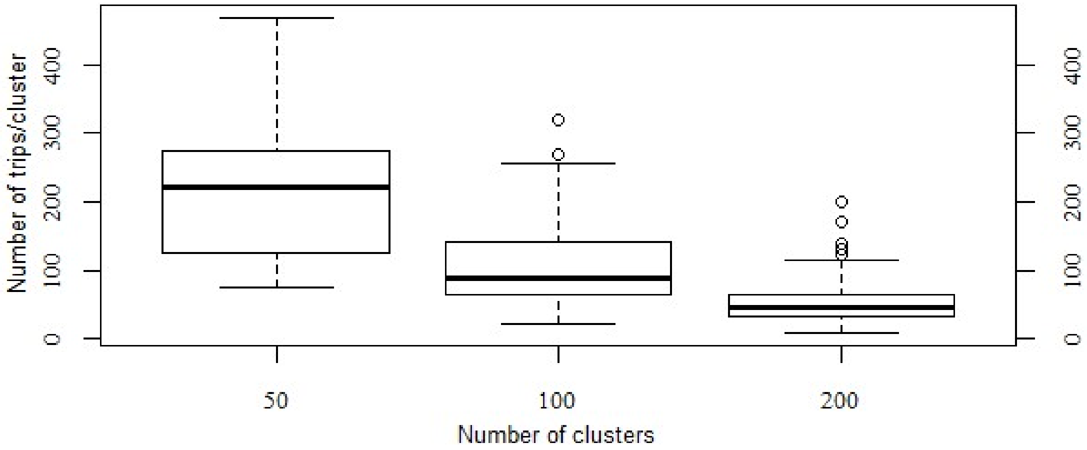


Figure 4.8: Boxplot of the resulting K-means clustering of II trips sample

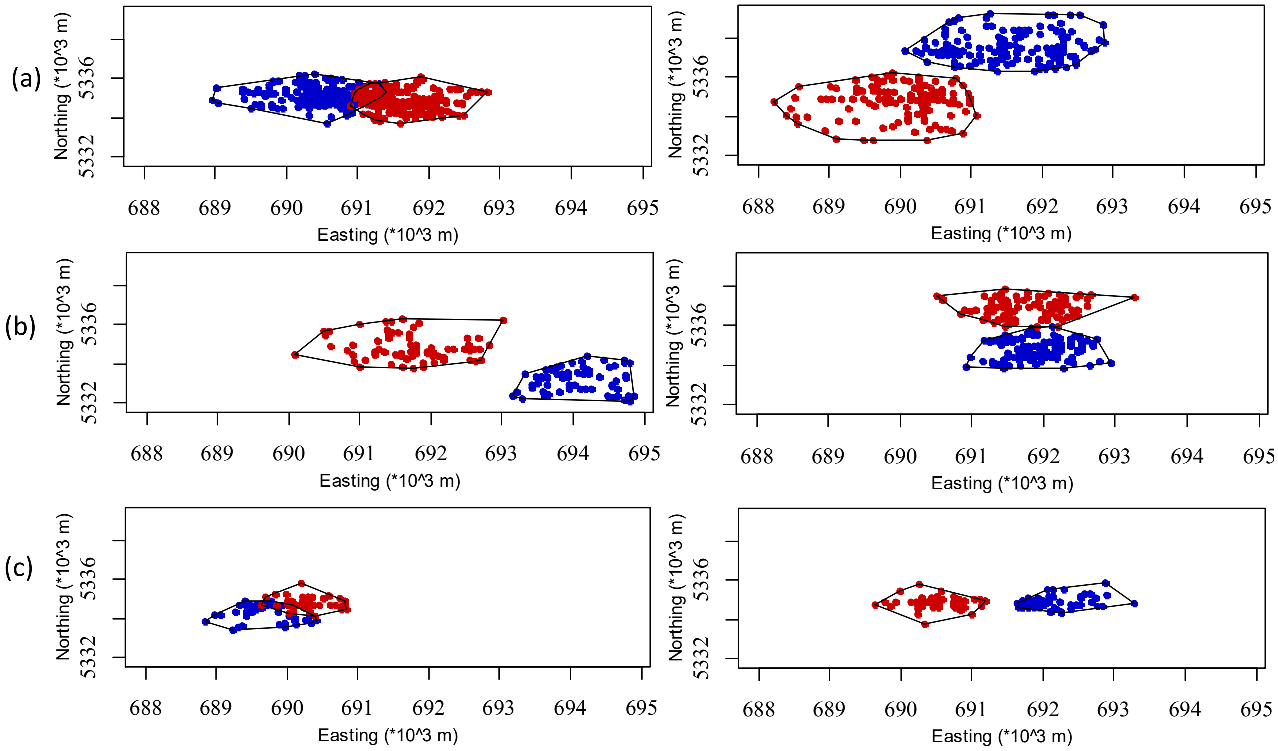


Figure 4.9: Sample of the resulting K-means clusters in the different cases: a) 50 clusters, b) 100 clusters and c) 200 clusters

From Figure 4.8, it can be concluded that there are no big differences in the number of trips/cluster within each case of the three cases, with only a few outliers, unlike the resulting clustering using DBSCAN method. Moreover, it can be noticed from Figure 4.9 that the clusters, in each case, are somewhat similar in size and spatially well-distributed throughout the whole study area. Thus, contrary to DBSCAN algorithm, K-means seems able to partition the points in the high-density areas into more clusters instead of grouping them into a single cluster.

4.2.2.3 Hierarchical method

Agglomerative hierarchical clustering method is tested on II trips sample. As in DBSCAN, there is no need to input the number of clusters before applying hierarchical clustering analysis. However, it is necessary to predetermine a measure of distance (similarity) and a linkage criterion to be used during the clustering process.

There are several distance metrics used in hierarchical clustering including Euclidean distance, Manhattan distance, Hamming distance, etc. The selection of the distance measure should be made based on the study domain and goals.

As the distance between the clusters is spatial and has to be computed based on the length of the straight line drawn from one cluster to another, Euclidean distance is the most common distance measure used in such situations, especially that the positions of origin and destination points are transformed from geographic coordinate system (longitude and latitude) to universal transverse mercator (in meters), and these points are located in a relatively small enclosed area. In other words, there is no need to use measures such as the great-circle distance to take the curvature of the earth into consideration.

After selecting a distance measure, it is necessary to choose a linkage criterion (i.e., from where to where the distance between two clusters should be computed) (Bock, n.d.). For example, it can be computed between the nearest two parts in these two clusters (minimum or single-linkage), the most distant two bits (maximum or complete-linkage), the centers of the two clusters (mean or average-linkage), or some other criterion. However, Ward’s method determines which observations to group based on reducing the sum of squared distances of each observation from the average observation in a cluster (Bock, n.d.). As with distance metrics, the choice of linkage criteria should be made based on the application’s domain and goals.

In the preliminary test, and for the above-mentioned reasons, Euclidean distance is used as a similarity measure in hierarchical clustering of II trips sample. Two of the most common linkage criteria are tested; Wards' linkage and single-linkage. As for the number of clusters in this test, the sample's trips are grouped into 150 clusters to check whether this method is suitable for this thesis' dataset. However, detailed criteria are introduced in subsection 4.2.5 to select the optimal number of clusters for the clustering method that will eventually be selected.

Single-linkage clustering of II trips sample resulted in one enormous cluster that contains 10,374 trips (accounts for 98% of II trips sample size), and covers the whole study area. The other clusters are very small with an average of 1.5 trips/cluster only. Therefore, the resulting clusters are not comparable and can't be analyzed to obtain meaningful results. Figure 4.10 shows the enormous cluster and one of the small clusters produced by single-linkage clustering.

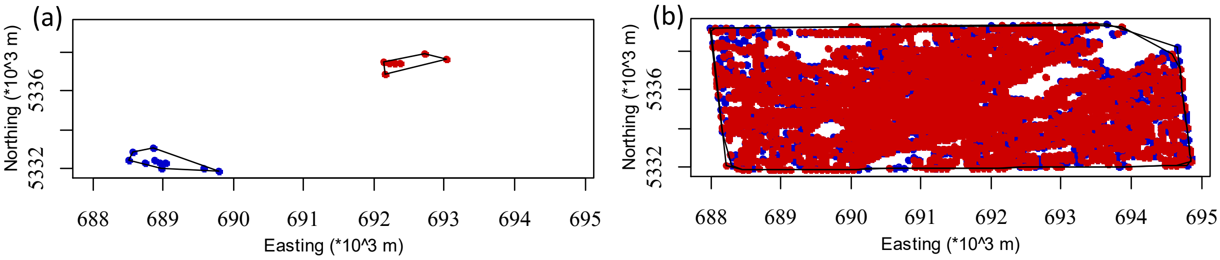


Figure 4.10: Sample of the resulting single-linkage clusters; a) one of the small clusters, b) the enormous cluster

Whereas, Ward's method clustering of II trips sample produced more similar clusters in terms of size and number of trips/cluster. The resulting clusters have an average of 71 trips/cluster and a median of 61 trips/cluster with a few outliers. Thus, the clusters are comparable and may give useful results. Figure 4.11 shows a sample of the resulting clusters created by Ward's hierarchical method. In addition, Figure 4.12 represents a boxplot of the clustering results in terms of number of trips/cluster.

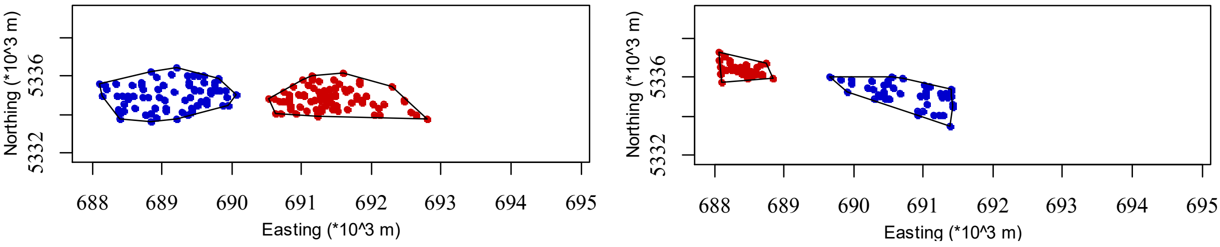


Figure 4.11: Sample of the produced clusters by Ward's hierarchical method

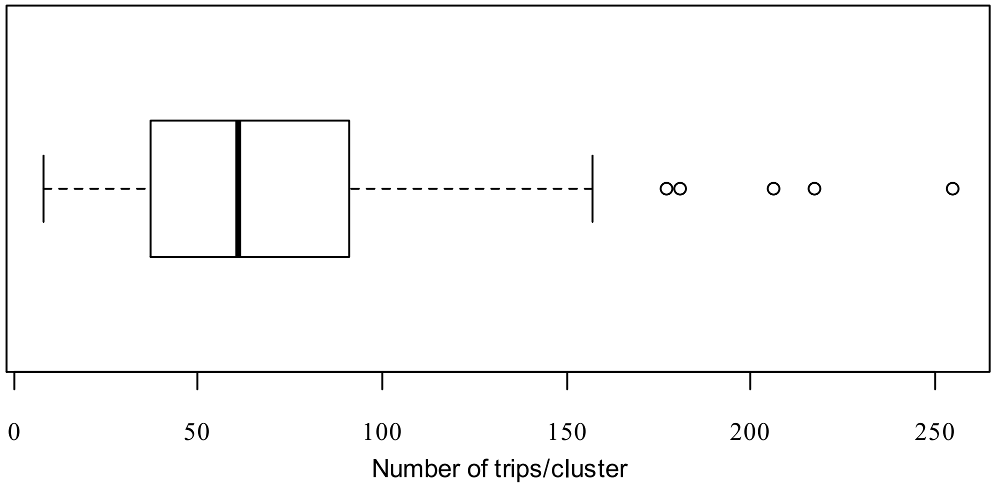


Figure 4.12: Boxplot of the resulting clustering of II trips sample using Ward's hierarchical method

In conclusion, the preliminary testing clearly shows that density-based clustering methods, like DBSCAN, are not suitable for this thesis as they are unable to divide II area into multiple clusters. Consequently, these method are excluded. However, partitioning methods, like K-means, and agglomerative hierarchical methods, such as Ward's method, successfully passed the preliminary test since they produce comparable, similar in size and number of trips, and well distributed clusters. In the next section, the strengths and weaknesses of the two methods are compared, and at the end, only one of them will be selected to proceed with data modeling and analysis.

4.2.3 Comparison between K-means and Hierarchical Clustering

As both of K-means method and hierarchical method produced similar good results in the preliminary test, which was conducted on II trips sample, both of them should practically and technically work well in clustering and analysis of the whole dataset. However, since only one method will be selected for the next steps, a comparison of both methods' theoretical advantages and disadvantages is conducted. Table 4.1 and Table 4.2 summarize the main advantages and disadvantages of K-means method and agglomerative hierarchical method respectively.

From Table 4.1 and Table 4.2, it can be concluded that K-means is simpler, faster, and needs lower computational costs than hierarchical clustering. However, hierarchical method doesn't require to determine number of clusters in advance, has more structured output, and shows more accuracy. Moreover, the disadvantages of K-means are much more critical than those of hierarchical method, and especially for the characteristics of the available dataset. For instance, K-means is somewhat unreliable and lack consistency since it randomly chooses the initial centers of clusters; thus, different results are produced on different runs. In addition, it is more sensitive to outliers and noisy data. On the contrary, most of the disadvantages of hierarchical clustering are not critical for the thesis' dataset. While hierarchical clustering is relatively slow when applied to big data, works poorly with mixed data types, and doesn't work with missing values in dataset, the available dataset is not very big considering that it will be divided into 6 parts (i.e., one part for each day) before applying clustering process, it consists of one data type only (numerical values), and it is free of any missing values. As a result, hierarchical clustering method is chosen over K-means method, and will be used in data modeling and analysis in this thesis.

Table 4.1: Advantages and disadvantages of K-means method

K-means Advantages	K-means Disadvantages
• Good for large number of variables and large datasets[a] • Suitable when clusters with a spherical shape and similar size are desired[b] • Fast; relatively low runtime complexity (linear)[a] • Simple and easy to setup and understand[a] • An instance can change cluster when the centroids are recomputed	• Requires a pre-knowledge of clusters number (K) which is difficult to predict in most cases[a] • Shape of all clusters is convex; tends to assume a predetermined circular shape to find clusters[c] • Cannot handle non-globular data of different sizes and densities[d] • Very sensitive to outliers; Unable to handle noisy data[d] • Restricted to data which has the notion of a center (centroid)[d] • Initial seeds and order of data have a strong impact on the final results[e] • Randomly choosing of initial clusters' centers • Different clustering results on different runs; results may not be repeatable and lack consistency[f] • Produces a single partitioning[g] • Uniform effect: often produces clusters with relatively uniform size even if the input data have different cluster size[h] • Empty clusters can be obtained if no points are allocated to a cluster during assignment step[i]

[a]Kaushik and Mathur (2014). [b]Fahim, Saake, Salem, Torkey, and Ramadan (2008). [c] Guo et al. (2012). [d]Sonagara and Badheka (2014). [e]Santini (2016). [f]Seif (2018). [g]Hu, Qian, Pei, Jin, and Zhu (2015). [h]Kumar, Rao, Govardhan, and Sandhya (2015). [i]Tan, Steinbach, and Kumar (2005).

Table 4.2: Advantages and disadvantages of hierarchical clustering method

Hierarchical Clustering Advantages	Hierarchical Clustering Disadvantages
• Can give different partitionings depending on the level of resolution; Any desired number of clusters can be obtained by cutting the dendrogram at the proper level[a] • Shows more quality (accuracy) as compared to K-means algorithm[b] • Does not require to specify the number of clusters in advance[b] • Easy to understand; Conceptually simple and theoretical properties are well understood[b] • It has a logical structure; well-structured output "dendrogram" • Gives deep insight of each step of merging different clusters • Very good for clustering of small datasets[b] • Any valid measure of distance can be used[c]	• Relatively slow (has to make several merge/split decisions) and thus relatively high run time complexity (quadratic)[a] • Doesn't work well on big data; as the number of records increases, the performance decreases[d] • Initial seeds might have impacts on final results • Sometimes it's difficult to identify the correct number of clusters from the dendrogram • Works poorly with mixed data types[e] • It is not possible to undo the previous step; once the instances are assigned to a cluster, they can no longer be moved around[f] • Does not work with values are missing in the data[e]

[a]Rai (2011). [b]Kaushik and Mathur (2014). [c]Chatterjee (2012). [d]Kaur and Kaur (2013). [e]Bock (n.d.). [f]Santini (2016).

4.2.4 Clustering Process

As indicated in the previous subsection, hierarchical clustering method is selected to cluster the trips of the dataset. This method is used to spatially group the trips, that were surveyed within Munich middle ring area, into clusters based on the location of the start and end points of these trips. Therefore, trips that have similar origin and destination points are put together in one cluster and so on. The clustering is done for each day separately.

In hierarchical clustering, a distance metric and a linkage criterion shall be selected before running the algorithm. The distance metric is important to establish the distance (dissimilarity) matrix, which is an n*n table that shows the distance between each pair of the observations in the dataset, where n is the total number of observations in the dataset. Euclidean distance is selected as a distance metric for the reasons mentioned in the previous section. The linkage criterion determines how the selected distance metric will be used to measure the distance between clusters to be merged, i.e., from which point in one cluster to which point in the other cluster. Ward's method is selected as a linkage criterion over single-linkage based on the results of the conducted preliminary test. In addition, according to Bock (n.d.), Ward's method is the sensible default where there are no clear theoretical justifications for choice of linkage criteria. Ward's method groups observations based on reducing the sum of squared distances of each observation from the average observation in a cluster. This is often appropriate as this concept of distance matches the standard assumptions of how to compute differences between groups in statistics (Bock, n.d.).

Consequently, agglomerative hierarchical clustering based on Euclidean distance metric and Ward's method linkage criterion is applied, using R software, to II dataset to group the trips based on the similarity in their origin and destination points. As mentioned earlier, II trips dataset is divided into six parts; one part for each day. Thus, clustering process is conducted for each day separately. At this step, a hierarchy (dendrogram) is produced as the output of the clustering process, which shows the full structure of how trips are clustered. Figure 4.13 shows only the very upper part of the dendrogram of one day since it is not possible to display the whole structure here due to page size limit. In the dendrogram, the y-axis (Height) is simply the value of the distance metric between clusters. For example, if two clusters merged at a height x, it means that the distance between those clusters is x.

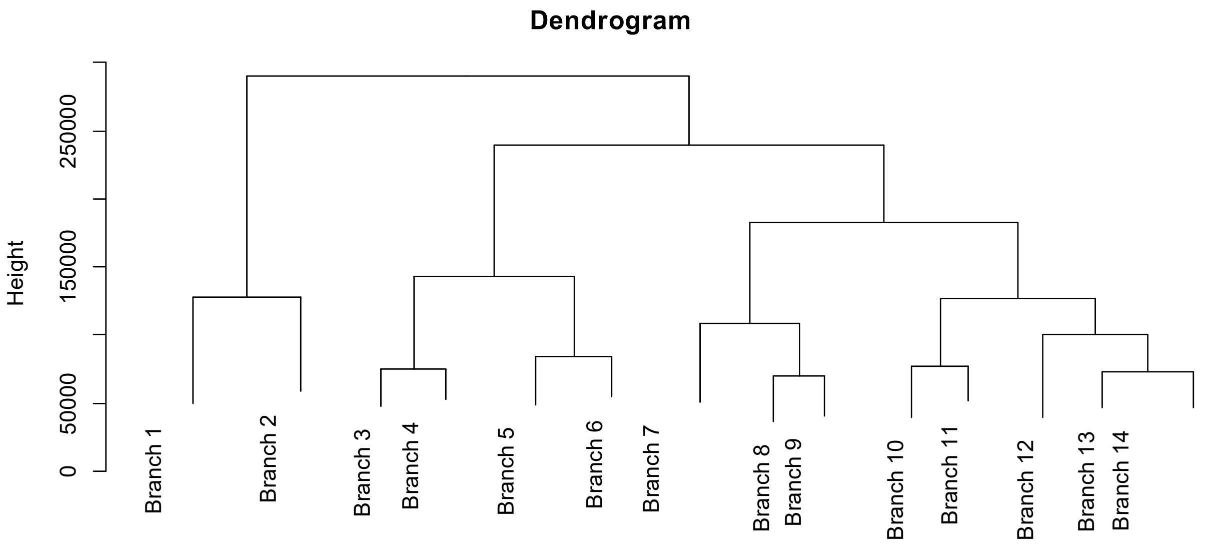


Figure 4.13: Upper part of the resulting dendrogram for Monday

4.2.5 Selecting Number of Clusters

Next step is to cut the dendrogram at a specific level to get the desired number of clusters. However, determining the desired number of clusters in not an easy task, and depends on several factors.

Some factors are related to statistical measures and others are related to application goals, characteristics of data, and meaningfulness of the desired results.

4.2.5.1 Statistical Methods

Three statistical methods are applied in this thesis on II trips sample attempting to determine the optimal number of clusters. Two of them are direct methods (elbow and silhouette) which consist of optimizing a criterion, such as the Within-Cluster Sum of Squares (WSS), while the third one (gap statistic) is a statistical testing method which consists of comparing evidence against null hypothesis (Kassambara, 2017).

Elbow method looks at the total WSS as a function of the number of clusters. One should choose a number of clusters so that adding another cluster doesn't improve much better the total WSS. The location of a bend (knee) in the resulting plot is generally considered as an indicator of the appropriate number of clusters (Kassambara, 2017). Figure 4.14 shows the resulting plot from applying elbow method on II trips sample. The dashed line passes through the plot's knee which indicates the recommended number of clusters that equals 4 clusters in this case.

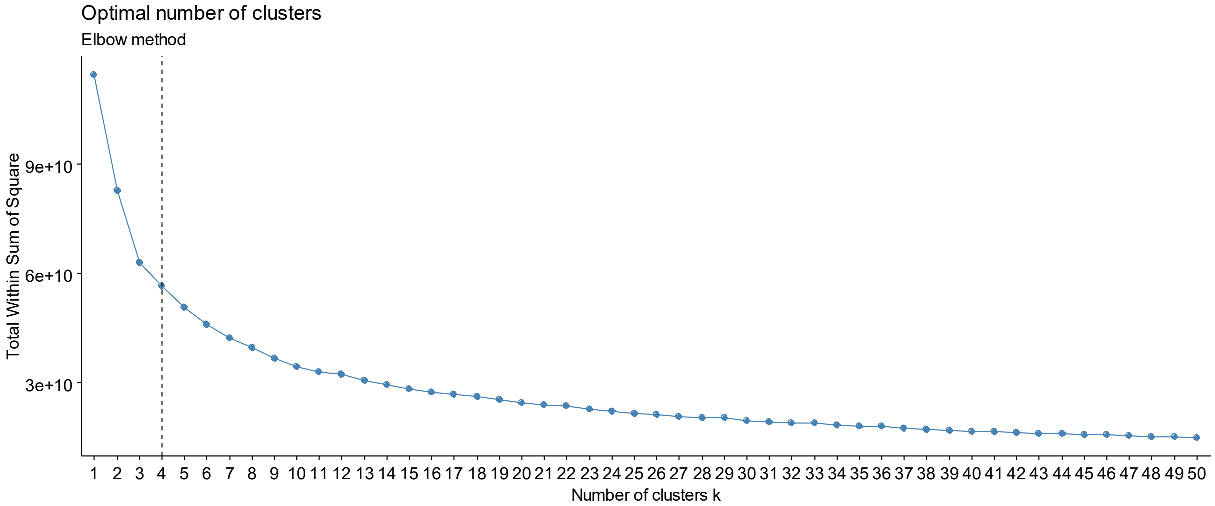


Figure 4.14: The resulting elbow method's plot

Average silhouette method measures the quality of a clustering by determining how well each object lies within its cluster. A high average silhouette width indicates a good clustering. Therefore, the location of the maximum in the resulting plot is considered as the appropriate number of clusters (Kassambara, 2017). Figure 4.15 shows the resulting plot from applying average silhouette method on II trips sample. The dashed line indicates the location of the maximum value (3) that represents the recommended number of clusters.

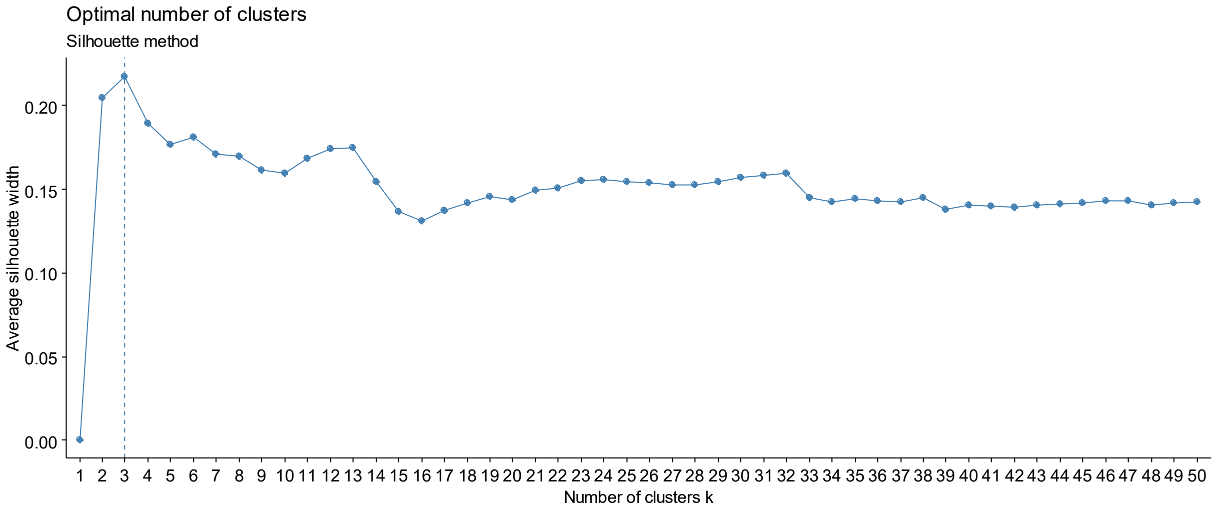


Figure 4.15: The plot resulting from average silhouette method

The gap statistic method has been published by Tibshirani, Walther, & Hastie (2001). This method compares the total within intra-cluster variation for different values of number of clusters with their expected values under null reference distribution of the data. The optimal number of clusters is the value that maximizes the gap statistic (i.e., that yields the largest gap statistic). This means that the clustering structure is far away from the random uniform distribution of points. Figure 4.16 shows the resulting plot from applying gap statistic method on II trips sample. The dashed line indicates the value that gives the maximum gap statistic which equals 9 clusters in this case.

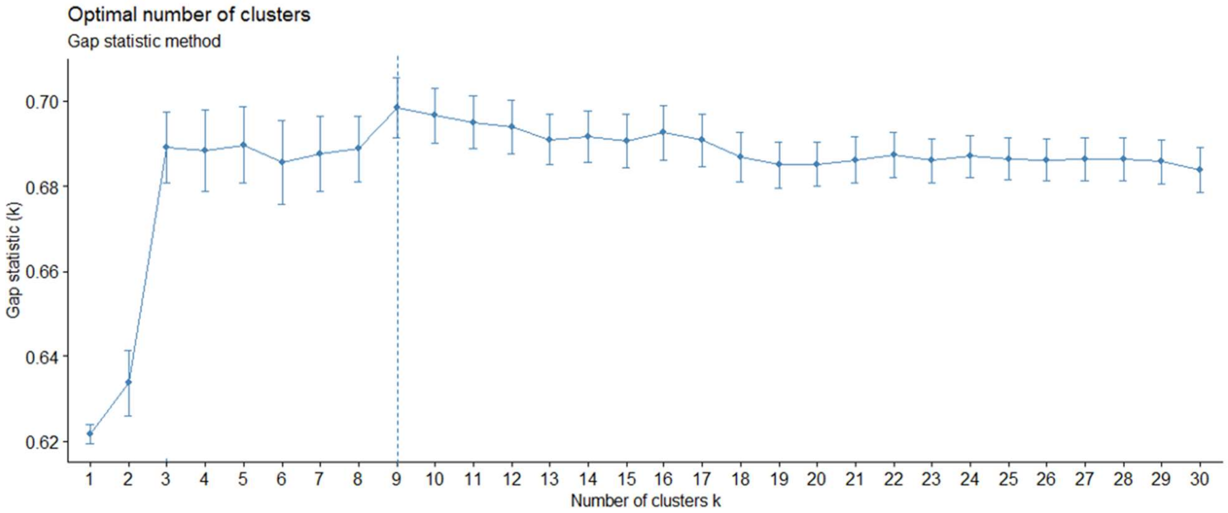


Figure 4.16: The plot resulting from gap statistic method

In summary, elbow, average silhouette, and gap statistic methods suggest 4, 3, and 9 clusters solution respectively. However, these values are too low and not satisfactory in the context of this thesis, where it is desired to preserve the data resolution by constructing as many OD pairs as possible (while satisfying the minimum cluster size constraints) in the study area to analyze the travel behavior and mobility patterns within these pairs.

4.2.5.2 Developed Criteria to Determine Optimal Number of Clusters

As the statistical methods couldn't provide satisfactory values for optimal number of clusters, special criteria are developed in this thesis to decide on number of clusters.

A cluster in this thesis is defined as a group of trips that travel from one specific zone (origin zone) to another (destination zone) within the study area. It is found that the size of the origin and destination zones for the resulting clusters differs from one number of clusters to another; the higher the number of clusters is, the smaller the origin and destination zones are. This is the starting point in the developed criteria where a meaningful size of zones is desired to be determined. In order to define this meaningful size of zones, a specific existing zoning system of the middle ring area in Munich city shall be selected and referred to. Parking management plan in the state capital of Munich (in German: Parkraummanagement in der Landeshauptstadt München) set by Munich city planning unit in 2013 (Figure 4.17), is used as a referential zoning system in this thesis. The plan is selected since parking zones are somehow related to trips' origin and destination zones. That is to say, most of trips travel within the study area were parked at one of the parking zones before starting the trip and will terminate the trip by parking in another one of these parking zones.

Size of a zone is defined in this thesis as the length of the longest x (easting) or y (northing) dimension in that zone. Figure 4.18 illustrates the concept used in determining the size of each zone. For example, the longest x dimension in Figure 4.18 is 2.20 km, and the longest y dimension is 1.03 km; thus, the size of this zone is the maximum of the two values which is 2.20 km in this case.

The size of parking zones in the parking management plan is found to be ranging from 0.55 to 1.52 km with an average of 1.02 km. Thus, the average size of origin and destination zones in the resulting clusters is desired to be as close as possible to 1.02 km.

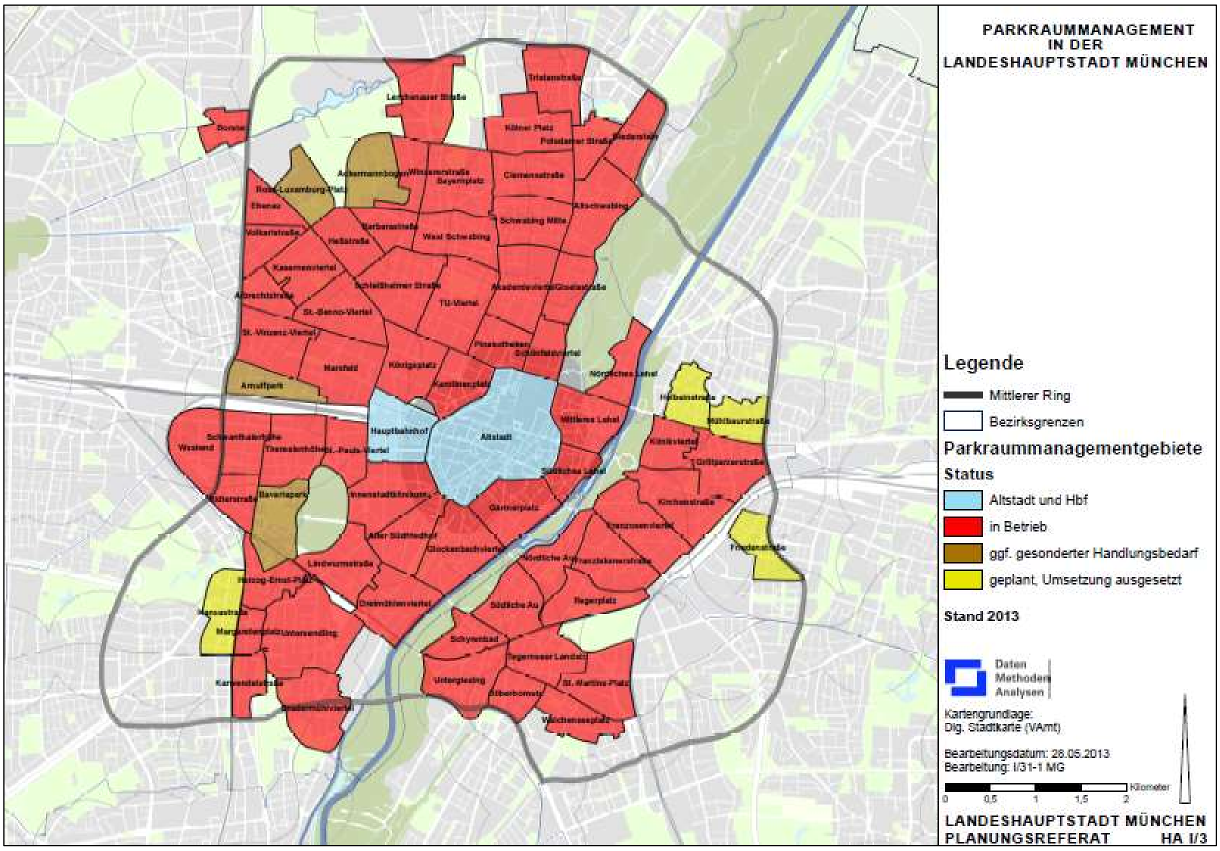


Figure 4.17: Parking management plan in the state capital of Munich, 2013. Source: Muenchen.de

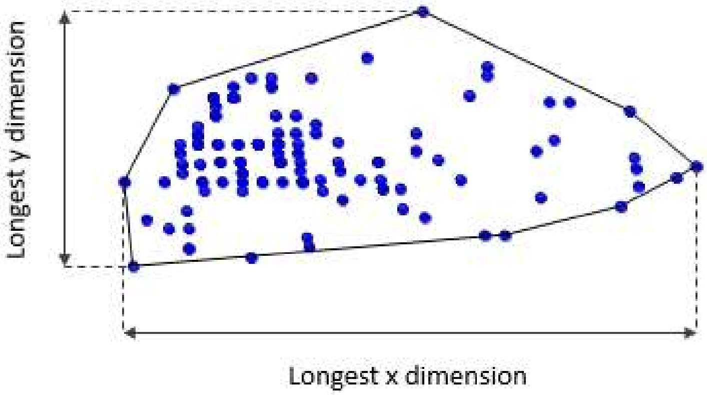


Figure 4.18: The concept used in determining the size of each zone

In addition to the first criterion (average size of origin and destination zones), other criteria are taken into consideration in determining the optimal number of clusters. Such criteria are mainly considered in order to assure the meaningfulness and usefulness of the resulting clusters, especially in the analysis part. For instance, it is important that the number of trips/cluster shall be not less than a specific threshold, since the lower the number of trips/cluster is, the less the representativeness of the sample (cluster) is. Another criterion taken into account is regarding the overlapping between the origin zone and the destination zone in a given cluster. No overlap of these zones in each cluster is desired, such that each cluster shall have only one major traffic flowline from one origin zone to another fully separated destination zone. Therefore, the clusters with an overlap between their origin and destination zones will not be useful in the analysis, and they will be excluded as a result. This is an essential requirement for computing the analysis factor that will be explained in more detail in next chapter. Examples of the overlap cases are shown in Figure 4.19.

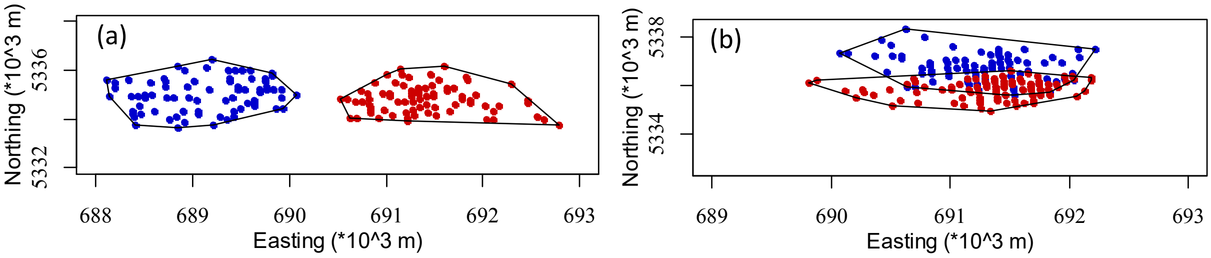


Figure 4.19: Examples of the no overlap case (a) and overlap case (b)

It is found that the percentage of the clusters with an overlap decreases as the number of clusters increases. Hence, in order to exclude as few clusters as possible, a specific threshold for the percentage of clusters with no overlap is set.

To summarize, the following are the criteria developed in this thesis to define the optimal number of clusters on each day:

1) Average size of all origin and destination zones of all clusters shall be as close as possible to the average size of the parking zones in the referential zoning system which equals 1.02 km.
2) Average number of trips/cluster shall be not less than a threshold of 30 trips/cluster. This threshold is selected as a rule of thumb for an adequate sample size according to Hogg & Tanis (2005).
3) Percentage of clusters with no overlap between their origin and destination zones shall be not less than 60% of all resulting clusters for each day.

The applied clustering algorithm in this thesis, taking into consideration the above-mentioned criteria, is shown in Figure 4.20.

Input: *Trips data frame includes Start Easting, Start Northing, End Easting, and End Northing attributes (Coordinates of origin and destination points of all trips)*

Output: *Clusters $C_1, C_2, \dots, C_k$ (Clusters of trips that share same origin and destination zones)*

Procedure:

```
Trips.dist<- dist(Trips, method= Euclidean)   # Establishing distance matrix using Euclidean distance
Trips.clustering<- hclust(Trips.dist, method= Ward)   # Spatially clustering of trips using hierarchical
                        clustering algorithm based on Ward's linkage method
Trips.clusters<- cutree(Trips.clustering, n)   # Cutting the resulting hierarchy into n clusters
For each cluster $C_i$ in Trips.clusters{
    Compute the size of origin and destination zone}
Determine average size of all origin and destination zones in all clusters (1st Criterion)
mean(table(Trips.clusters))   # Determining average number of trips/cluster (2nd Criterion)
For each cluster $C_i$ in Trips.clusters{
    If gIntersects($C_i$ Origin, $C_i$ Destination) == False{
        r = r + 1}}   # Defining number of clusters with no overlap between origin and destination zones
%no.overlap<- (r/n)*100   # Determining percentage of clusters with no overlap (3rd Criterion)
Trial and error of different numbers of clusters until finding the optimal number (k) based on the
  three criteria above
Trips.clusters<- cutree(Trips.clustering, k)   # Cutting the resulting hierarchy into k clusters
```

Figure 4.20: The applied spatial clustering algorithm

4.2.6 Clustering Results

Several iterations to specify the optimal number of clusters for each day are conducted until a trade-off among these criteria is reached. Table 4.3 shows the summary of the selected number of clusters for each day based on the developed criteria.

Table 4.3: Summary of selected number of clusters for each day

Day	Number of clusters	Average size of all zones (km)	Average number of trips/cluster	Number of clusters with no overlap	Percentage of clusters with no overlap
Sunday	250	1.62	51	169	68%
Monday	300	1.61	61	194	65%
Tuesday	300	1.64	68	191	64%
Wednesday	300	1.60	60	196	65%
Thursday	300	1.57	62	200	67%
Friday	300	1.60	62	197	65%

Figure 4.21 shows a sample of the resulting clusters for Monday. Every cluster has two convex polygons in the same color, one of them represents the cluster's origin zone, while the other represents the cluster's destination zone. Figure 4.22 shows another sample of the resulting clusters for Monday, where each cluster is represented by one square and one circle in the same color; the

square represents the centroid of the cluster's origin zone and the circle represents the centroid of the cluster's destination zone. An arrow is drawn from each cluster's origin zone centroid to its destination zone centroid indicating the traffic flow direction in that cluster.

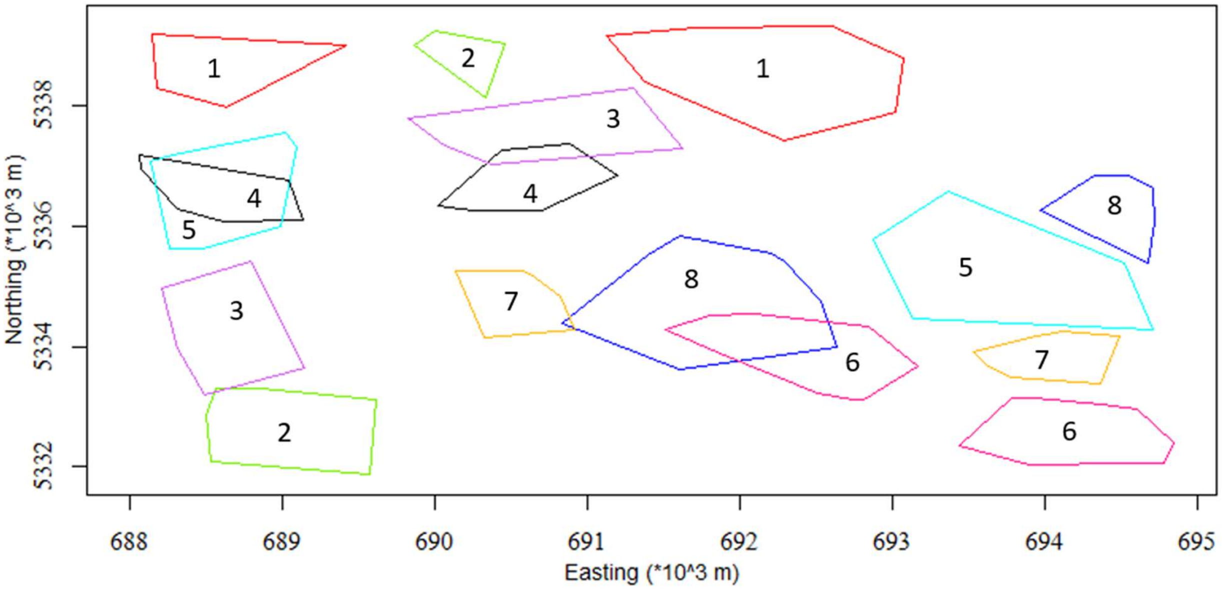


Figure 4.21: Sample of the resulting clusters represented by their origin and destination zones

.

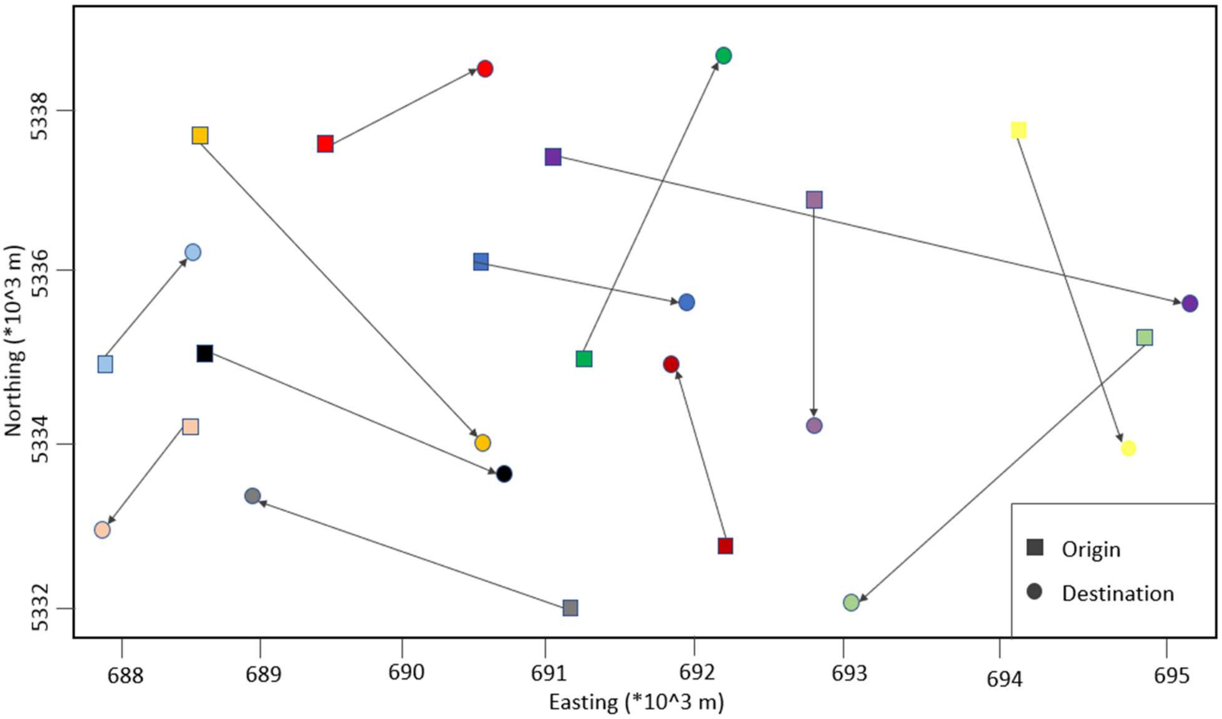


Figure 4.22: Sample of the resulting clusters represented by their origin and destination centroids

4.3 Relative Deviation Area (RDA)

In the previous section, spatial clusters of II trips for each day are created using hierarchical clustering method based on the location of trips' origin and destination points. While at this section, a new analysis factor is introduced and computed for the produced clusters. This factor will be used in the analysis part to understand the travel behavior within the study area.

4.3.1 RDA concept

Relative Deviation Area (RDA) is a novel tool introduced in this thesis that aims at computing the relative area by which a vehicle, traveling from one zone (A) to another (B), is deviating from some referential path (e.g., the fastest path) from zone A to zone B (Equation 1). Where the relative area is defined as the area between a trip's trajectory and the trajectory of the referential trip divided by the length of the section of the trip's trajectory that extends between the nearest pair (out of the four possible pairs) of OD points of the two trajectories.

$$RDA = \frac{DA}{L} \tag{1}$$

Where; RDA is the relative deviation area (area unit/length unit)

DA (deviation area) is the area by which a trajectory is deviating from another referential trajectory (area unit)

L is the length of the trajectory's section that extends between nearest pair of OD points of the two trajectories (length unit)

In this thesis, the fastest paths in each cluster are selected as the referential paths in computing RDA. The fastest trip is defined here as the trip that has the highest average speed among all the other trips traveling from one zone to another.

Figure 4.23 illustrates the concept of RDA, where OP1 and OR1 are the origin points of path 1 and reference path 1 respectively, and DP1 and DR1 are the destination points of path 1 and reference path 1 respectively. The shaded area represents the deviation area of trajectory 1 from the reference path 1. The red lines indicate the location of the nearest pair of OD points.

The idea of this tool is generated from attempting to understand and answer questions related to people travel behavior. Examples of such questions include: Why do people choose specific routes over others? Do most people travel along the fastest routes? Are people willing to take longer routes in order to avoid the traffic congestion on the shorter routes? Is there a relationship between trips average speed and the spatial distribution of trips? and other similar questions.

Moreover, this tool can be used as a part of the performance measures for a given road network. It also can be used in transport planning applications such as traffic assignment (route choice) among different OD pairs within a specific road network. In addition, it can form a basis for more detailed analytical framework.

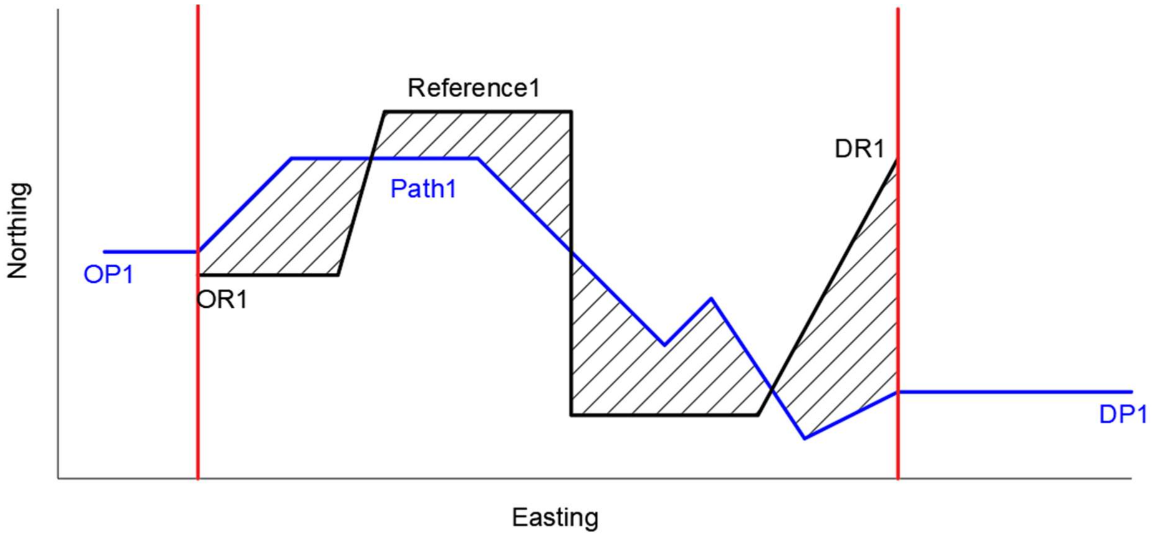


Figure 4.23: Concept of RDA tool

4.3.2 RDA Computation

Next step is finding how to compute RDA between any two trajectories. A trip trajectory, denoted by T, is a time-ordered sequence or time-series of 3 tuples (x, y, t) representing the x and y (Easting and Northing) coordinates of a trip at time t. Let $T_i = (p^i_1, p^i_{2,\ldots,} p^i_N)$ denote the trajectory of trip i, where $p^i_n = (x, y, t)^i_n$ is the n^th^ point of the sequence (n = 1,…, N). Given two trip trajectories $T_i = (p^i_1, p^i_{2,\ldots,} p^i_N)$ and $T_j = (p^j_1, p^j_{2,\ldots,} p^j_M)$ with size N and M, respectively, the trajectory data addressed in this thesis have the following characteristics:

- Different lengths: trajectories may have different lengths in terms of the number of waypoints within each trajectory (e.g., N and M can be different for $T_i = (p^i_1, p^i_{2,\ldots,} p^i_N)$ and $T_j = (p^j_1, p^j_{2,\ldots,} p^j_M)$).
- Different and uneven sampling rates: data sampling rates are not necessarily identical across trajectories (e.g., $p^i_1, p^i_{2,\ldots,} p^i_N$ are recorded every 30 seconds while $p^j_1, p^j_{2,\ldots,} p^j_M$ are recorded every 60 seconds). Moreover, the time interval between two consecutive waypoints within

the same trajectory can also vary (e.g., 30 seconds between p^i_1 and p^i_2 while 45 seconds between p^i_2 and p^i_3).

- Two or more y values for the same x value and vice versa: unlike trajectories of free moving objects in Euclidean space such as animals and hurricanes, vehicle trajectories are constrained by the underlying road network (Kim & Mahmassani, 2015). Therefore, two or more waypoints within the same trajectory can have the same x (Easting) or y (Northing) value, like in U-turn movements (e.g., p^i_1 = (691257, 5332638, 0) and p^i_2 = (691257, 5332660, 30)).

Taking the above trajectory data characteristics into consideration, using the traditional area-computing methods to find the area between any two trajectories (T_i and T_j) is not applicable. For instance, let y (Northing) values denote a mathematical function of x (Easting) values in both trajectories T_i and T_j, such that T_i has function $f_i(x)$ and T_j has function $f_j(x)$. Trying to find the area between the curves of the two functions by integration will not work due to the third consideration above where the same value of x can have two or more y values, and thus estimating the functions $f_i(x)$ and $f_j(x)$ is not possible as any function in general is defined as a set of ordered pairs in which each x element has only one y element associated with it. Moreover, using the area under curve methods like Simpson's rule or trapezoidal rule to calculate area between T_i and T_j is also not possible, as these methods require that both functions (trajectories) have the same number of elements (points) and even the same x values. This requirement conflicts with the first characteristic of the available trajectory dataset where trajectories may have different lengths in terms of the number of waypoints.

Consequently, an optimization approach for robustly measuring the area-based deviation between two paths, called ALCAMP (Algorithm for finding the Least-Cost Areal Mapping between Paths) which was introduced by Mueller et al. (2016), is used in this thesis to compute the deviation area between two trajectories. ALCAMP measures the deviation between two paths and produces a mapping between corresponding points on the two paths, that produces the least total cost, where cost is the area between corresponding points and segments. The method is robust to a number of aspects in real path data, such as crossovers, self-intersections, differences in path segmentation, and partial or incomplete paths. Figure 4.24 shows example mappings produced by this algorithm, where the red and dark gray polylines in each plot represent the two paths, and the light gray area represents the computed deviation area between the two paths.

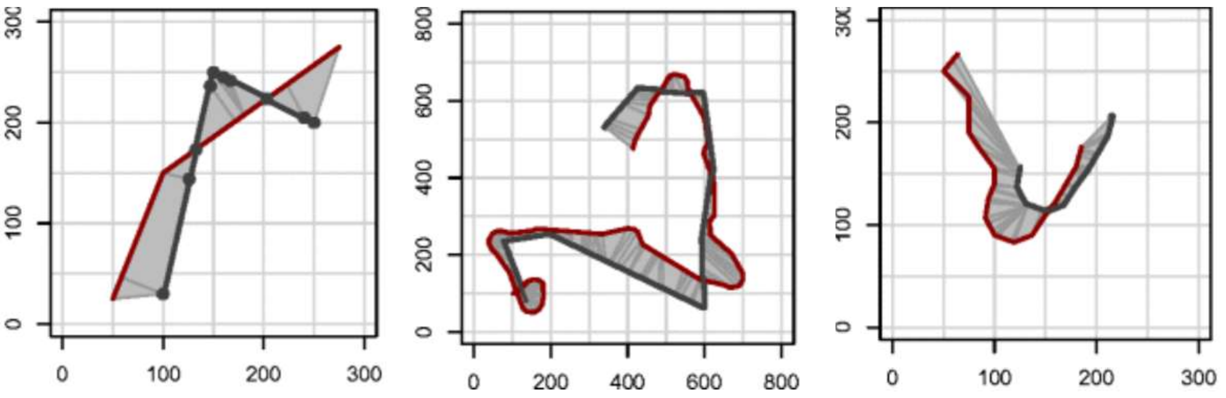


Figure 4.24: Example mappings produced by ALCAMP. Source: Mueller et al. (2016)

Nevertheless, some modifications shall be implemented to the algorithm before applying it to dataset's trajectories. These modifications are related to some considerations linked to the characteristics and application of RDA tool. The following is a description of these considerations and their consequent modifications:

- RDA limits: the two trajectories, between which the deviation area shall be calculated, might not have very close origin and destination points although they belong to the same cluster, as the average size of the origin and destination zones for each cluster is around 1.60 km. Therefore, limits for the deviation area are set to terminate the trajectories so that the unnecessary extensions of one or both trajectories will not be considered in the computation of the deviation area. These limits are defined as the nearest (internal) OD pair of the two trajectories. Figure 4.25 and Figure 4.26 show an example of the deviation area with and without the limits respectively, where the red lines in Figure 4.25 represent the RDA limits between the nearest OD pair (i.e., OR1, DR1), and the shaded area in both figures represents the deviation area between path 1 and reference 1.
- Direction of travel: this point is actually related to the first point, where a special attention shall be paid to the travel direction of the trajectories before setting the RDA limits. Direction of traffic flow differs from one cluster to another (e.g., from north to south in one cluster and from east to west in another, etc.). Therefore, all these cases of travel direction shall be checked in order to properly define the RDA limits for all trajectories in each cluster.
- Trajectories' detours: in some cases, a trip's trajectory may extend beyond the origin and/or destination point of the trip. Thus, this extension, that is mainly resulted from road network constraints and detours (e.g., U-turn movements or sharp curves), will be terminated by the

RDA limits and excluded from calculated deviation area if no improvements done to consider it (Figure 4.27). In the algorithm used to compute RDA, particular attention is given to such detours so that the consequent extensions are taken into consideration in the calculated deviation area (Figure 4.28). In Figure 4.27, it is clear that the deviation area is incomplete due to ignoring the sharp curve at the end of reference 1 trajectory.

- Path simplifying: For paths with relatively few points (fewer than 100), the completion time of mapping and computing deviation area using ALCAMP algorithm is tolerable, but when mapping one or more complex paths containing hundreds of points, the time to complete the mapping can take much longer (Mueller, Perelman, & Veinott, 2016). Although the average number of waypoints per trip in the dataset is 34, there are some trips that have much higher number of waypoints up to 1700. Therefore, A robust algorithm that has been proposed by Latecki & Lakamper (2000) is used to improve the efficiency and reduce the computational costs. This algorithm can simplify paths by removing redundant points and merging segments that lie on the same line, or that do not contribute significantly to the shape of the path. Hence, any individual path can be reduced to a smaller number of critical points using this algorithm. Figure 4.29 shows an example of a path that originally has 73 waypoints. However, in Figure 4.30, the number of waypoints is decreased to only 26 waypoints after applying the path simplifying algorithm. This helps in improving the efficiency without affecting much the shape of the path. In both figures, the blue point indicates the start (origin) point of the path, while the red one indicates the end (destination) point of the path.

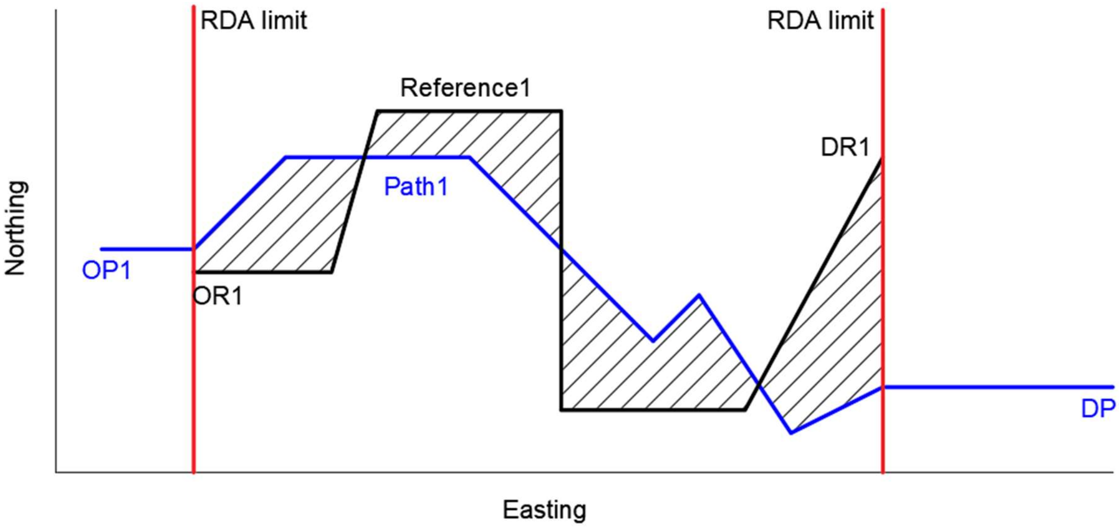


Figure 4.25: Example of deviation area considering the limits between nearest OD pair

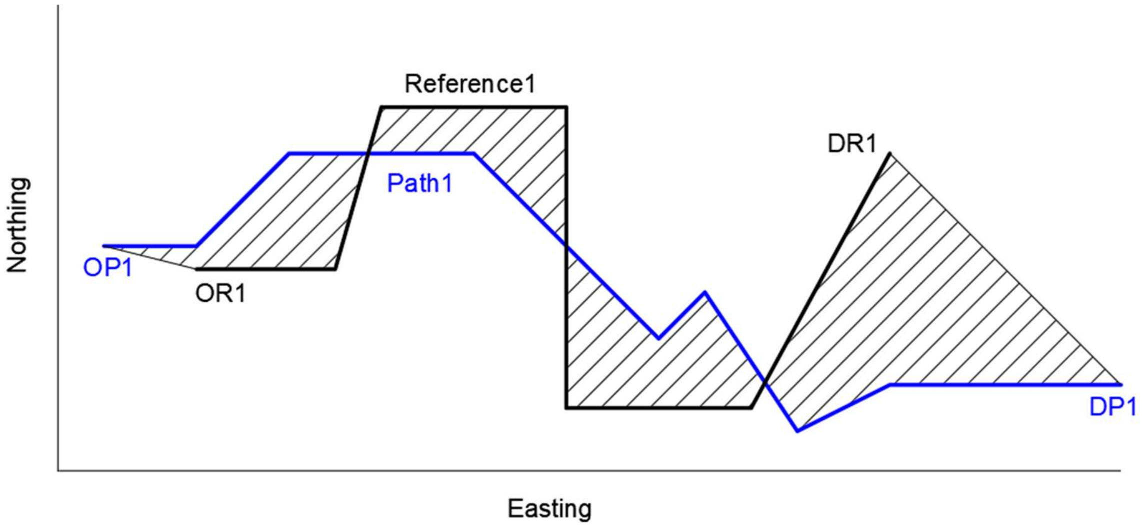


Figure 4.26: Example of deviation area without considering the limits between nearest OD pair

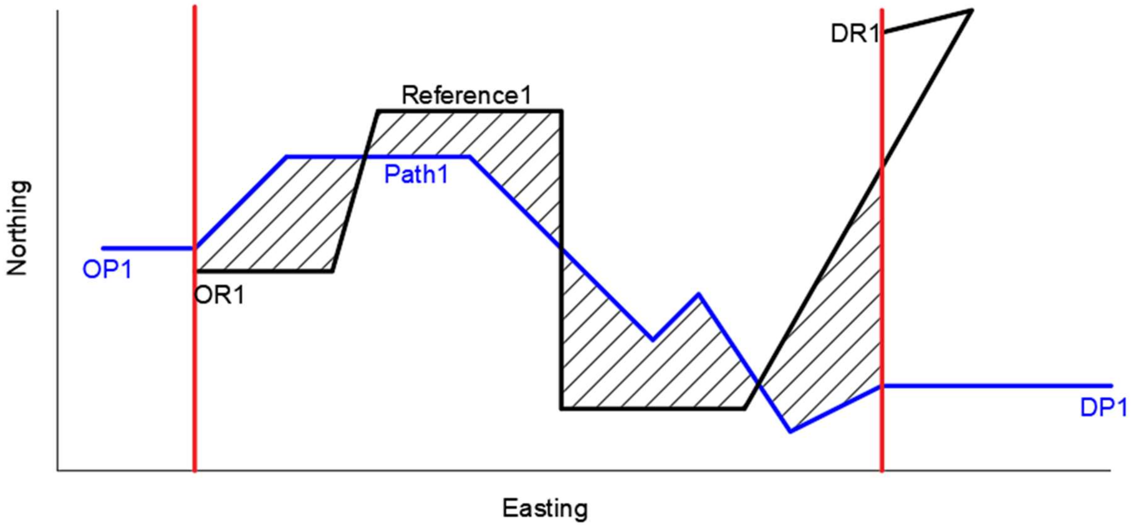


Figure 4.27: Example of deviation area without considering trajectories' detours

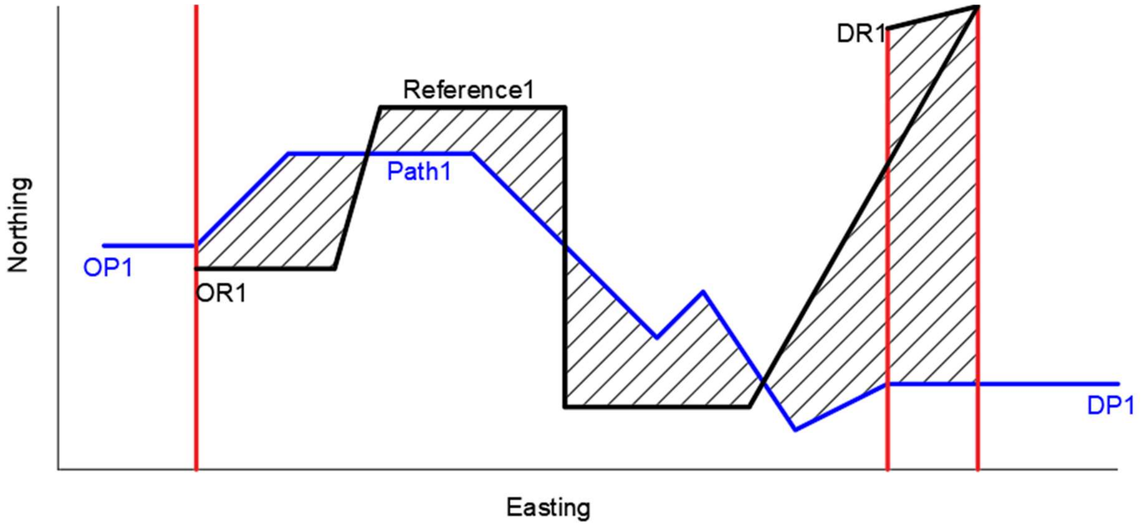


Figure 4.28: Example of deviation area considering trajectories' detours

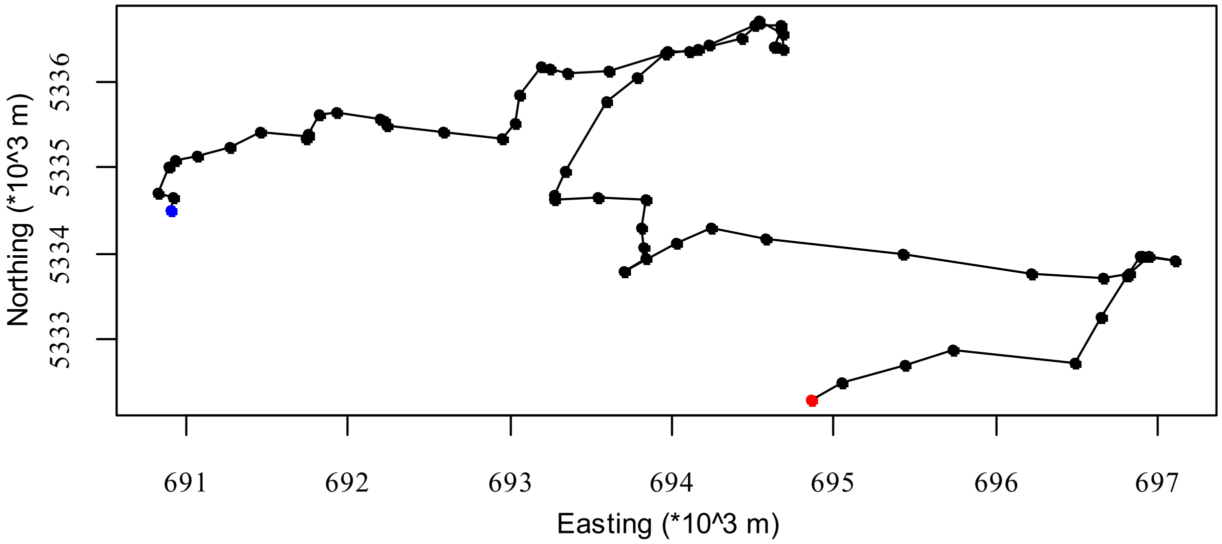


Figure 4.29: Example of a path before applying path simplifying algorithm

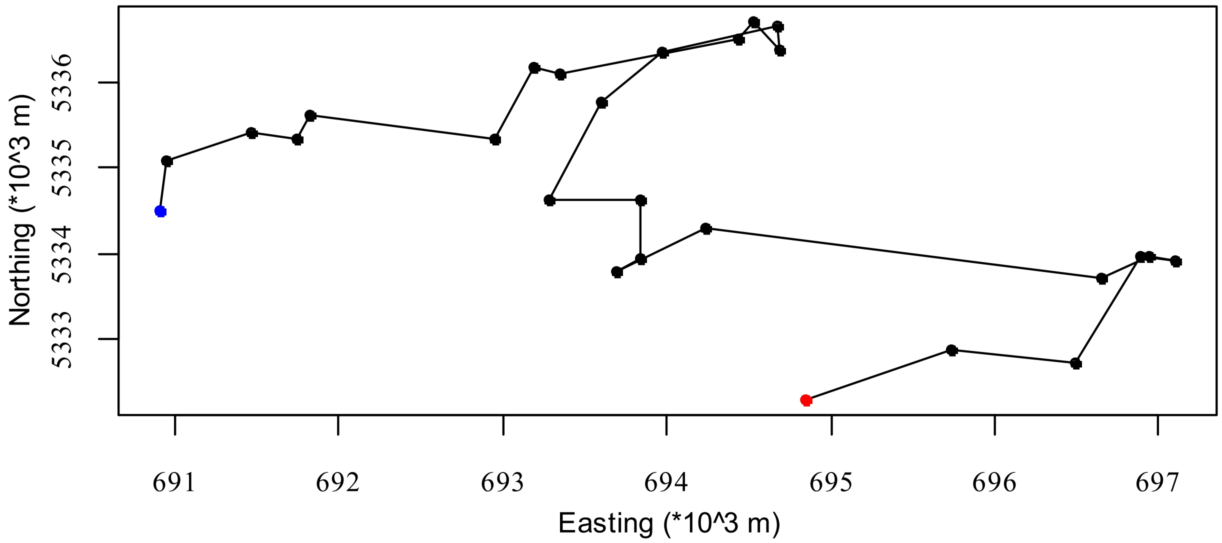


Figure 4.30: Example of a path after applying path simplifying algorithm

4.3.3 RDA Application to Dataset

All points mentioned above are taken into consideration in RDA tool. In this thesis, RDA is computed for all clusters on all days. First, fastest seven trips in each cluster are selected as the referential trips (references) for that cluster, where fastest trips are defined as the trips that have the highest average speed among all other trips within the same cluster. Average speed, in turn, is the total distance traveled by a vehicle divided by the elapsed time to cover that distance. Figure 4.31 shows the trajectory of the seven references of one of the resulting clusters, where each reference trajectory is shown in a unique color. Origin points are in blue, while destination points are in red.

The fastest trips are also desired to be those that have relatively short travel time, as some of the fastest trips might have high average speed but take very long routes and thus high travel distance and time. For example, two vehicles travel from zone A to zone B; the first takes a 5-km route with a travel time of 10 minutes, while the other vehicle takes a 15-km route with travel time of 30 minutes, both vehicles have the same average speed of 30 kph; however, the first case is preferable as the travel time is shorter. For this reason, it was checked that most of the fast trips have relatively short travel time. The scatter plot in Figure 4.32 shows an example of the relationship between average speed and travel time for all trips in one of the resulting cluster. It can be noticed that the trips with highest average speed have short travel time compared to the other trips.

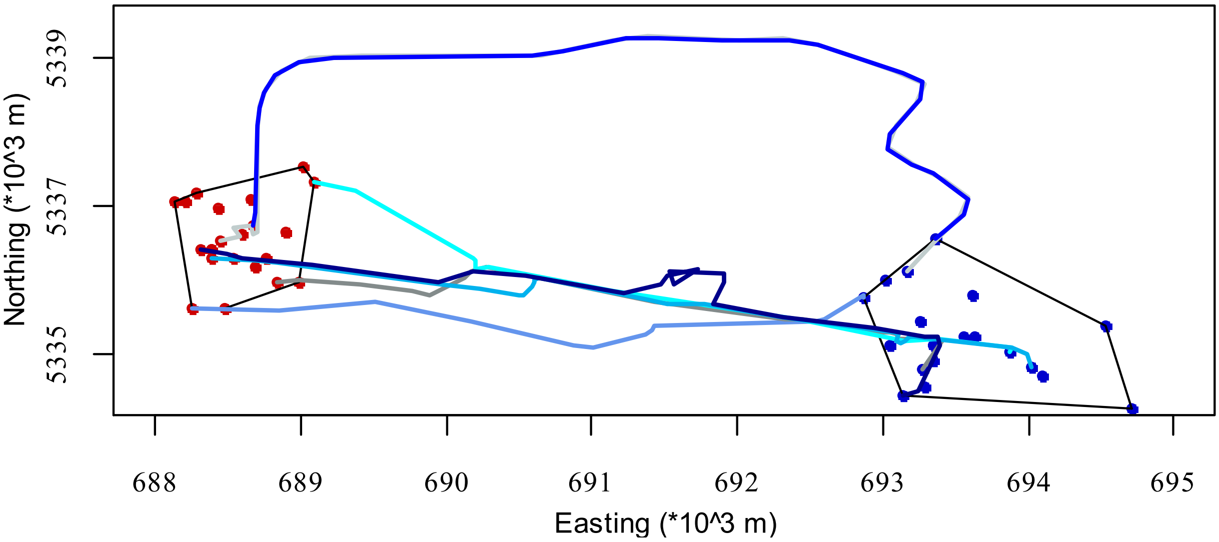


Figure 4.31: Trajectory of the seven references in one of the resulting clusters

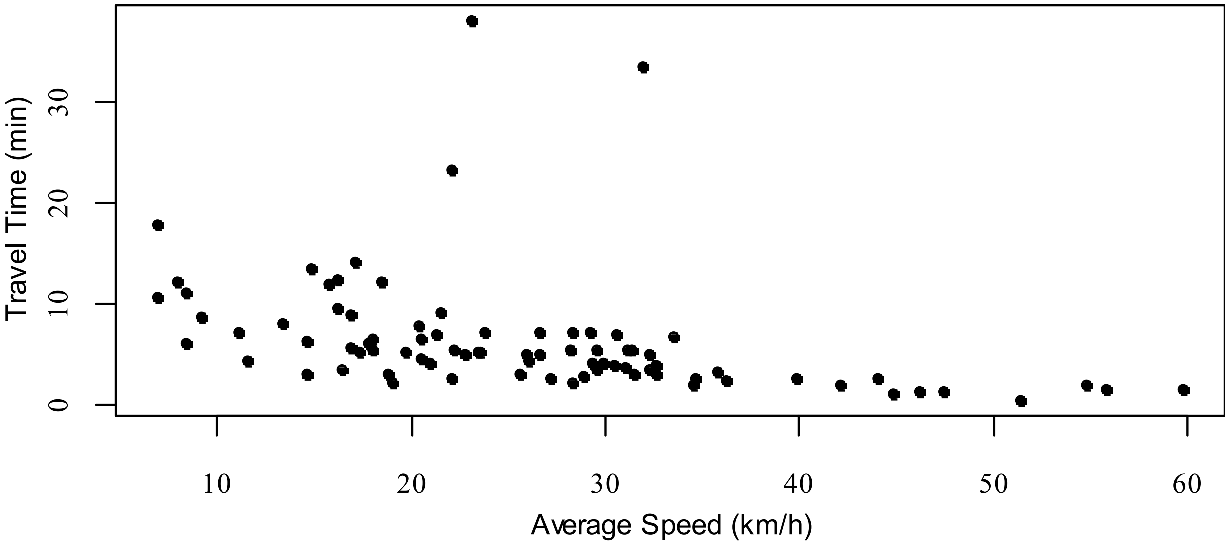


Figure 4.32: Average speed vs. travel time for all trips of one cluster

Two reasons behind considering several (seven) references for each cluster; first, to guarantee that the references are spatially distributed within the cluster, and thus at least one reference is suitable (similar) for each trip. Second, to assure that the high-speed references with relatively long travel time (if any) do not affect the RDA final results, as they produce high DA and RDA values and thus will not be selected as the minimum RDA in this case.

Next, DA is calculated seven times for each trip in each cluster (one time for each reference). In other words, DA is computed between each trip and each reference that belong to the same cluster. Then, RDA is also computed seven times for each trip by dividing each resulting DA from the previous step by the respective efficient trajectory length of that trip, where the efficient length is the length of the trajectory's part used in computing DA (i.e., neglecting the unnecessary extensions). Finally, only the minimum value out of the seven RDA values resulting from the third step is selected as the approved RDA for that trip. This is because the reference that gives the minimum RDA of a trip is the most similar one to that trip. Figure 4.33 shows the used algorithm to calculate RDA.

4.4 Relationship between RDA and Average Speed

In order to understand passenger's travel behavior, DA and RDA are computed for each trip in each cluster on each day according to the steps explained in section 4.3. Next, the relationships between trip average speed (V) and RDA, and between the average speed difference between respective reference and trip (ΔV) and RDA are investigated. These relationships are examined for each day separately and for all days together. In addition, the relations at peak and off-peak periods for one weekday and one weekend day are compared. Another case considering private cars only , i.e., excluding taxicab services, is tested as well.

Since the number of clusters for each day is high (i.e., 191 cluster/day on average), it is not possible to separately investigate each cluster or each zone in the study area in the context of this thesis. Therefore, all resulting RDA and average speed values from all clusters are aggregated for each day. Figure 4.34 and Figure 4.35 show the histogram distributions of V and RDA for one weekday and one weekend day respectively. It can be noticed that V follows the Gaussian (normal) distribution, while RDA follows the exponential distribution. In addition, V values are slightly higher on weekends than those on weekdays, which can be explained that peak hours are concentrated in relatively short time intervals (i.e., work and school start and end times) on weekdays compared to weekends where peak hours are more distributed throughout the whole day. As for RDA values, there is no big difference between weekdays and weekends.

Input: Clusters $C_1, C_2, \ldots, C_k$ and waypoints (trajectory) of each trip

Output: RDA of each trip in each cluster

Procedure:

```
for each cluster $C_i$ in Trips.clusters{
  sort(trips in $C_i$, V, decreasing)   # Sorting trips in descending order based on their average speed
  ref(i)<- the fastest seven trips in $C_i$}   # Defining the referential trips for each cluster
for each cluster $C_i$ in Trips.clusters{
  for each trip $t_j$ in $C_i${
    path1<- waypoints of $t_j$ in $C_i$   # Defining trajectory of $t_j$ in $C_i$
    for each reference $ref_x$ in $C_i${
      path2<- waypoints of $ref_x$ in $C_i$   # Defining trajectory of $ref_x$ in $C_i$
      if $O_1$<$D_1$ && $O_2$<$D_2${   # Checking the direction of travel for both paths by comparing the
      coordinates of their origin and destination points
        if $O_1$<$O_2$ && $D_1$>$D_2${   # Checking which OD pair is the nearest one
          path1.1<- path1 excluding the unnecessary extensions outside the nearest OD pair
          path2.1<- path2 excluding the unnecessary extensions outside the nearest OD pair
          path1.1<- SimplifyPath(path1.1)   # Simplifying path1.1 by removing redundant waypoints
          path2.1<- SimplifyPath(path1.1)   # Simplifying path2.1 by removing redundant waypoints
          DA($C_i$, $t_j$, $ref_x$)<- Createmap(path1.1, path2.1)   # Computing deviation area between $t_j$ and
         $ref_x$ in $C_i$
          RDA($C_i$, $t_j$, $ref_x$)<- DA($C_i$, $t_j$, $ref_x$)/ PathDist(path1.1)   # Computing relative deviation area
         between $t_j$ and $ref_x$ in $C_i$ by dividing deviation area by the length of path1.1
        } else if $O_2$>$O_1$ && $D_1$<$D_2${
        …  # Checking all other nearest OD possibilities
      } else if $O_1$<$D_1$ && $O_2$>$D_2${
      …}}}}} # Checking all other travel direction possibilities for both paths
for each cluster $C_i$ in Trips.clusters{
  for each trip $t_j$ in $C_i${
    RDA($C_i$, $t_j$)<- min(all resulting RDA($C_i$, $t_j$, $ref_x$))}}   # Determining final RDA value for $t_j$ in $C_i$ which
    is the minimum RDA out of the seven calculated values for $t_j$ with the seven references in $C_i$
```

Figure 4.33: The applied algorithm for computing RDA

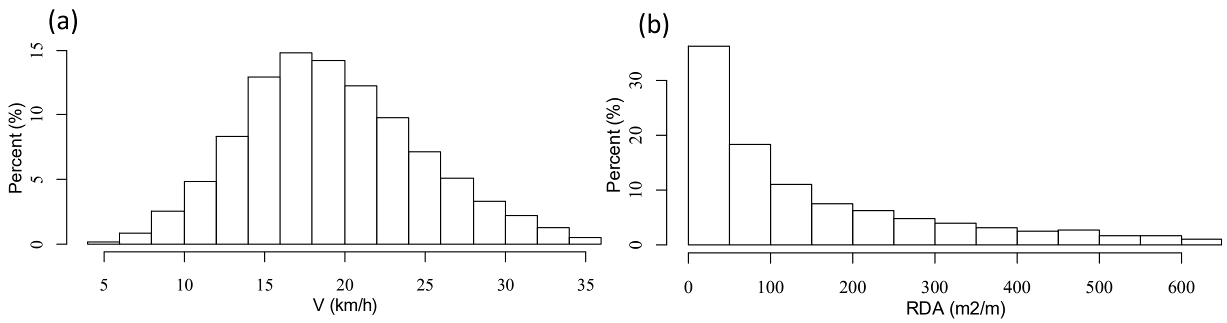


Figure 4.34: Histogram distibutions of a) V and b) RDA for a weekday (Monday)

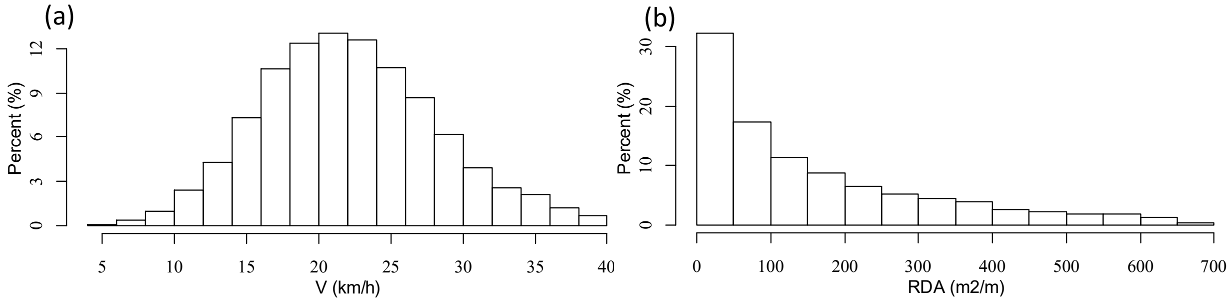


Figure 4.35: Histogram distributions of a) V and b) RDA for a weekend day (Sunday)

Figure 4.36 and Figure 4.37 show the scatter plot of the relationship between V and RDA and between ΔV and RDA for all trips in one day respectively. It can be concluded from both figures that there are some outliers which are very distant from other observations. These outliers shall be detected and handled. Two techniques are usually used to detect outliers; boxplot and bagplot. In boxplot, the outliers in each variable are detected separately. Whereas, bagplot is a technique proposed by Rousseeuw, Ruts, & Tukey (1999) as a generalization of the boxplot to bivariate data. It aims to visualize the location, spread, skewness and outliers of the data set. In bagplot, the outliers in both variables are detected simultaneously. Figure 4.38 shows an example of outliers detection using bagplot method, where the red stars represent the outliers.

In this thesis, bagplot method is used to detect outliers. It was observed that the outliers form less than 5% of the dataset only and are very far from other normal observations especially regarding RDA values, i.e., up to 500 times the values of non-outliers. This can be caused by measurement errors, abnormal driving behavior (e.g., taking a very circuitous route, driving around, or trip-chaining especially by taxi drivers) such as in Figure 4.39, or other unknown reasons. Moreover, a considerable percentage of the outliers have average trip speed higher than 50 kph exceeding the speed limit within the study area. For these reasons, the outliers are excluded and not considered.

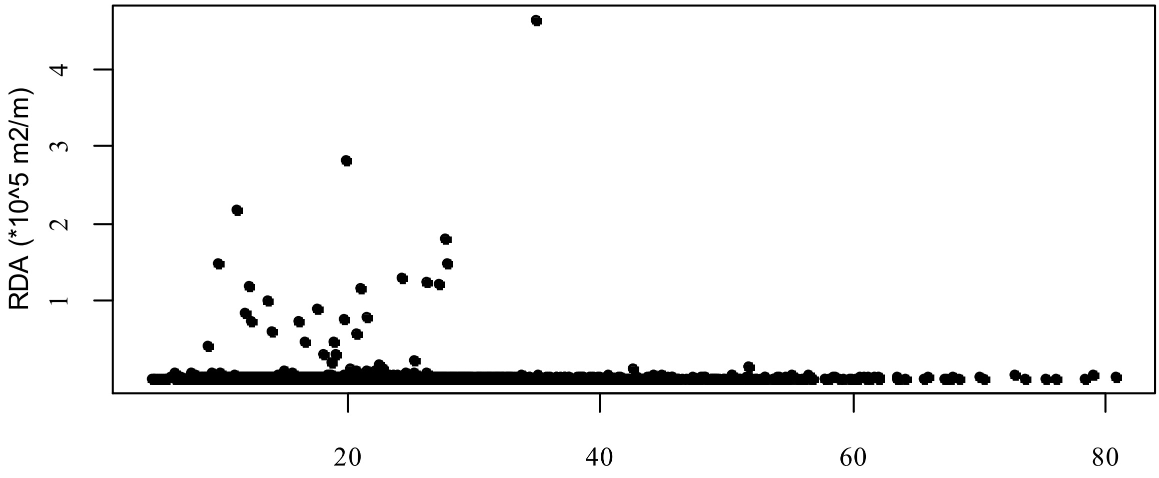


Figure 4.36: Scatter plot of V and RDA for Monday

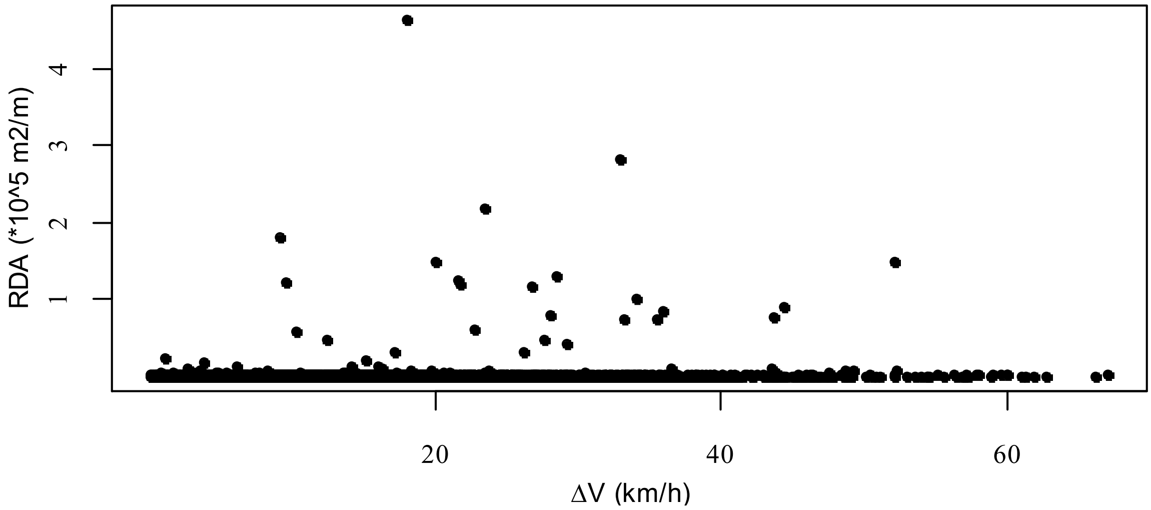


Figure 4.37: Scatter plot of ΔV and RDA for Monday

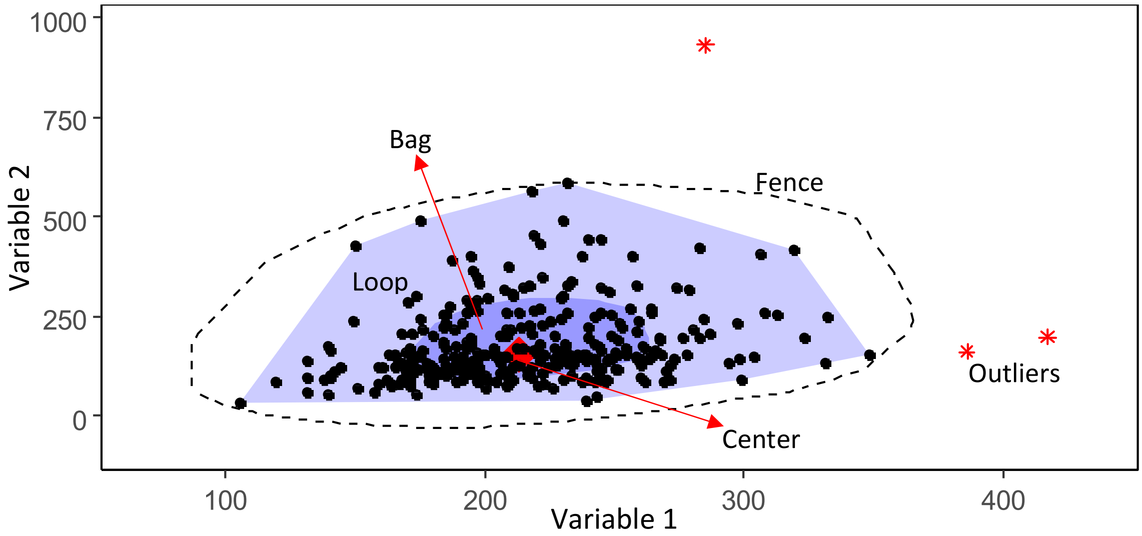


Figure 4.38: Example of outliers detection using bagplot method

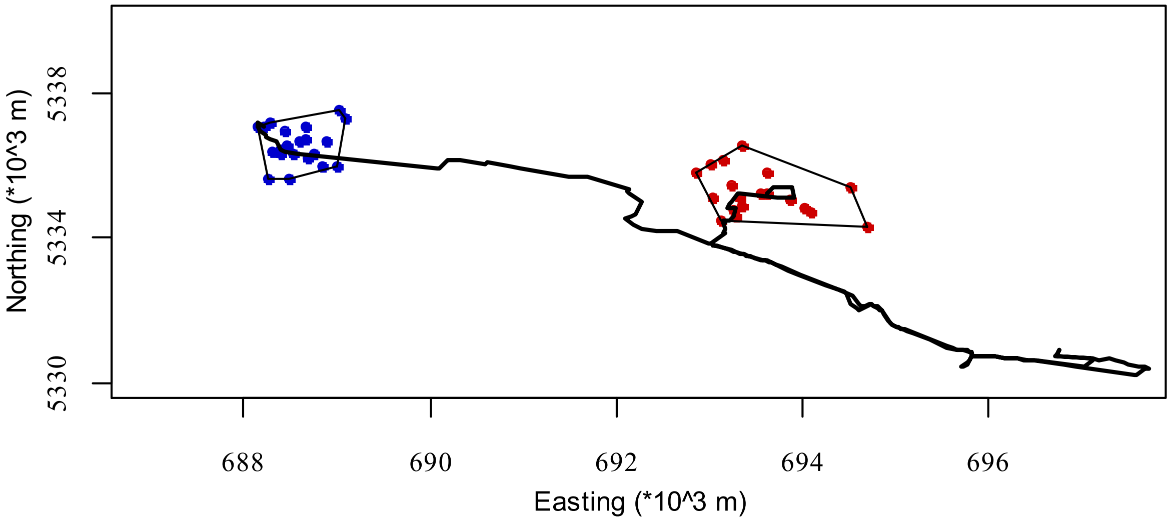


Figure 4.39: An example of the noticed abnormal driving behavior, where the black line represents the vehicle's trajectory

As mentioned earlier, two relationships are investigated in this thesis; relationship between V and RDA, and relationship between ΔV and RDA. Figure 4.40 and Figure 4.41 show an example of the resulting scatter plots of the relationship between V and RDA, and between ΔV and RDA for all trips in one day after excluding the outliers respectively.

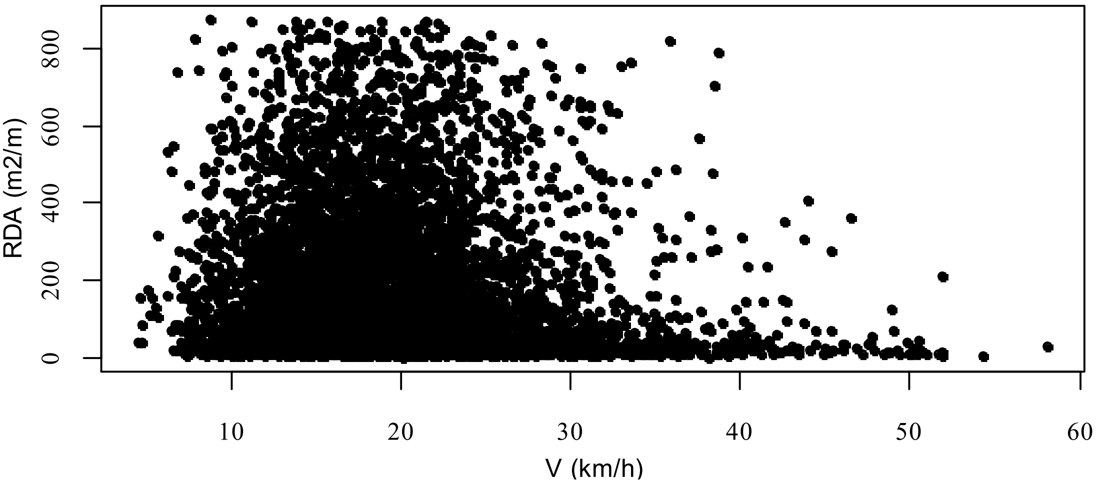


Figure 4.40: Scatter plot of V and RDA for Monday after excluding outliers

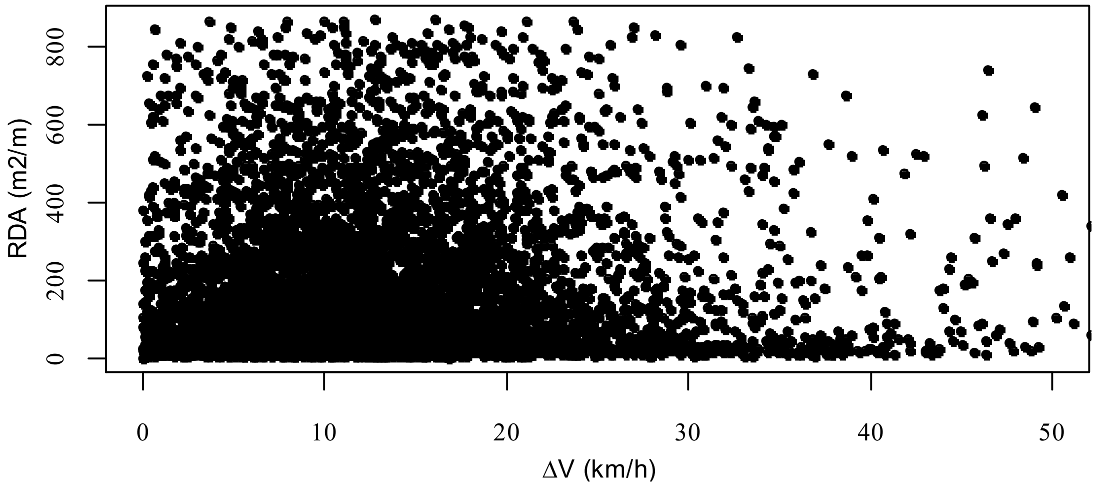


Figure 4.41: Scatter plot of ΔV and RDA for Monday after excluding outliers

In order to understand these relationships, nonparametric regression analysis is applied. Nonparametric regression analysis is usually used when the relationship is nonlinear, and the parametric form and the parameters are unknown (Tibshirani & Wasserman, 2015). In nonparametric regression, the shape of the functional relationship between the response (dependent) and the explanatory (independent) variables are not predetermined, and thus the shape of the function is determined from the data (Tibshirani & Wasserman, 2015; Mahmoud, 2014). In this thesis, Kernel regression, which is one of the most common nonparametric regression methods, is applied. Kernel regression is proposed to estimate the conditional expectation of a random variable. In nonparametric regression, the conditional expectation of a variable Y given a variable X is written as E(Y|X) = m(X), where the unknown function m is approximated by a locally weighted average, using a kernel as the weighting function such that closer points are given higher weights (Huang & Sun, 2013).

Kernel regression is applied using “np” R package developed by Hayfield & Racine (2008). It is used to investigate the aforementioned relationships for each day and for all days together as well. The applied algorithm to perform the regression and plot the relationships is shown in Figure 4.42.

```
Input: RDA and V values for each trip $t_j$ in each cluster $C_i$
Output: Relationship plots showing the regression curve
Procedure:
1 all.RDA<- do.call(rbind, RDA)   # Aggregating RDA values of all trips in all clusters for each day into
   one object
2 V.RDA<- cbind(V, all.RDA)   # Defining a matrix contains all V and RDA values for each day
3 for each cluster $C_i$ in Trips.clusters{
4    for each trip $t_j$ in $C_i${
5       $\Delta V(C_i, t_j)$<- $V_{ref.min} – V(C_i, t_j)$}}   # Computing average speed difference between each trip and the
                                                   reference that gives minimum RDA for that trip
6 ΔV.RDA<- cbind(ΔV, all.RDA)   # Defining a matrix contains all ΔV and RDA values for each day
7 compBagplot(V.RDA)   # Detecting and excluding outliers from V.RDA matrix using bagplot
8 bws<- npregbw(V.RDA)   # Determining V.RDA bandwidth value to be used in Kernel regression
9 reg.V.RDA<- npreg(V.RDA, bws)   # Performing Kernel regression between V and RDA using
                                                    computed bandwidth
10 plot(reg.V.RDA, plot.errors.method="asymptotic", plot.errors.style="band")   # Plotting resulting
    Kernel regression curve between V and RDA showing standard error bands
11 Repeat steps from line 7 to line 10 for relationship between ΔV and RDA
```

Figure 4.42: The applied algorithm to perform Kernel regression and plot the relationships

5. Results and Discussion

In this chapter, the results of Kernel regression applied to examine the relationships between V and RDA, and between ΔV and RDA for different cases are presented and discussed.

Figure 5.1 and Figure 5.3 show the scatter plot of the relationships between V and RDA, and between ΔV and RDA for all days respectively, where the blue line indicates the estimated relation curve using Kernel regression method. Figure 5.2 and Figure 5.4 show the zoomed-in plot of the regression curve for the relations between V and RDA, and between ΔV and RDA for each day separately respectively, where the solid line represents the regression curve and the two dotted lines on either side indicate the standard error.

From Figure 5.1 and Figure 5.2, it can be concluded that the regression curves of the relationship between V and RDA for each day are broadly similar. RDA value in each day slightly increases at the beginning of the curve until reaching a peak point when V value ranges between 10 and 15 kph. Next, RDA value starts decreasing slightly at first before considerably dropping, in general, after V value approaches 25-30 kph. Therefore, the curve can be divided into two regimes; when V is lower than 10-15 kph (before peak point) and when V is higher than 10-15 kph (after peak point). In the first regime, V increases when RDA increases. In other words, when vehicles take longer routes and deviate more from the fastest routes, they can achieve higher trip average speed. This may refer to the situation when the fastest routes are congested, and thus deviating from them will achieve a similar or a bit higher average speed. In the second regime, V decreases when RDA increases. This means that the higher the deviation area between the route that a vehicle takes and the respective fastest route, the lower the trip average speed the vehicle can achieve. This may refer to the situation when the fastest routes are not congested, and thus deviating from them will achieve a lower average speed. As a result, these figures might indicate a potential relationship between V and RDA.

The regression curves for each day which represent the relationship between ΔV and RDA are also somewhat similar. This can be noticed in Figure 5.3 and Figure 5.4 where ΔV values steadily increase as RDA values increase. This relation is sensible as it means that the higher the deviation area between a trip's path and the respective fastest path, the higher the average speed difference between them, and thus the lower the average speed of that trip. Consequently, these figures might denote a potential relationship between ΔV and RDA.

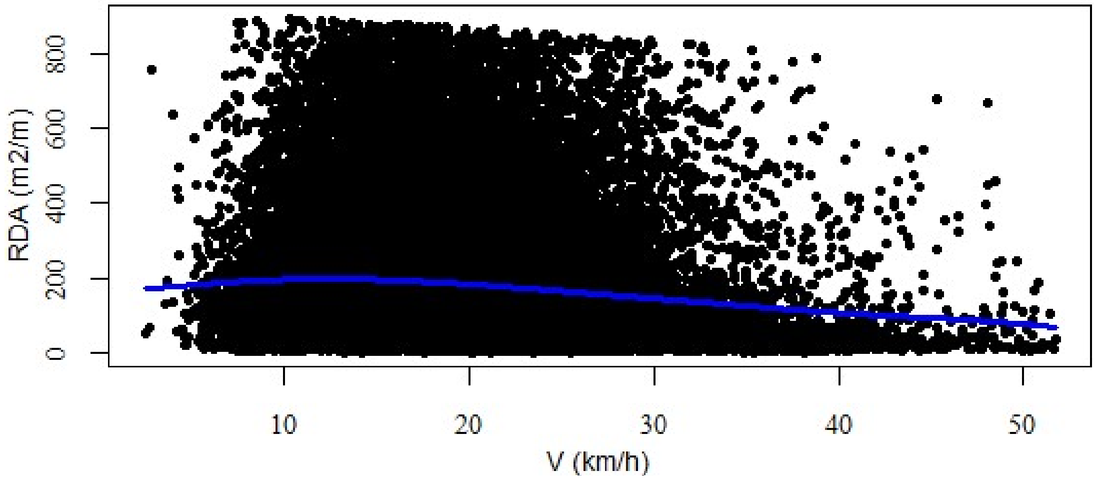


Figure 5.1: Estimated curve of the relationship between V and RDA for all days

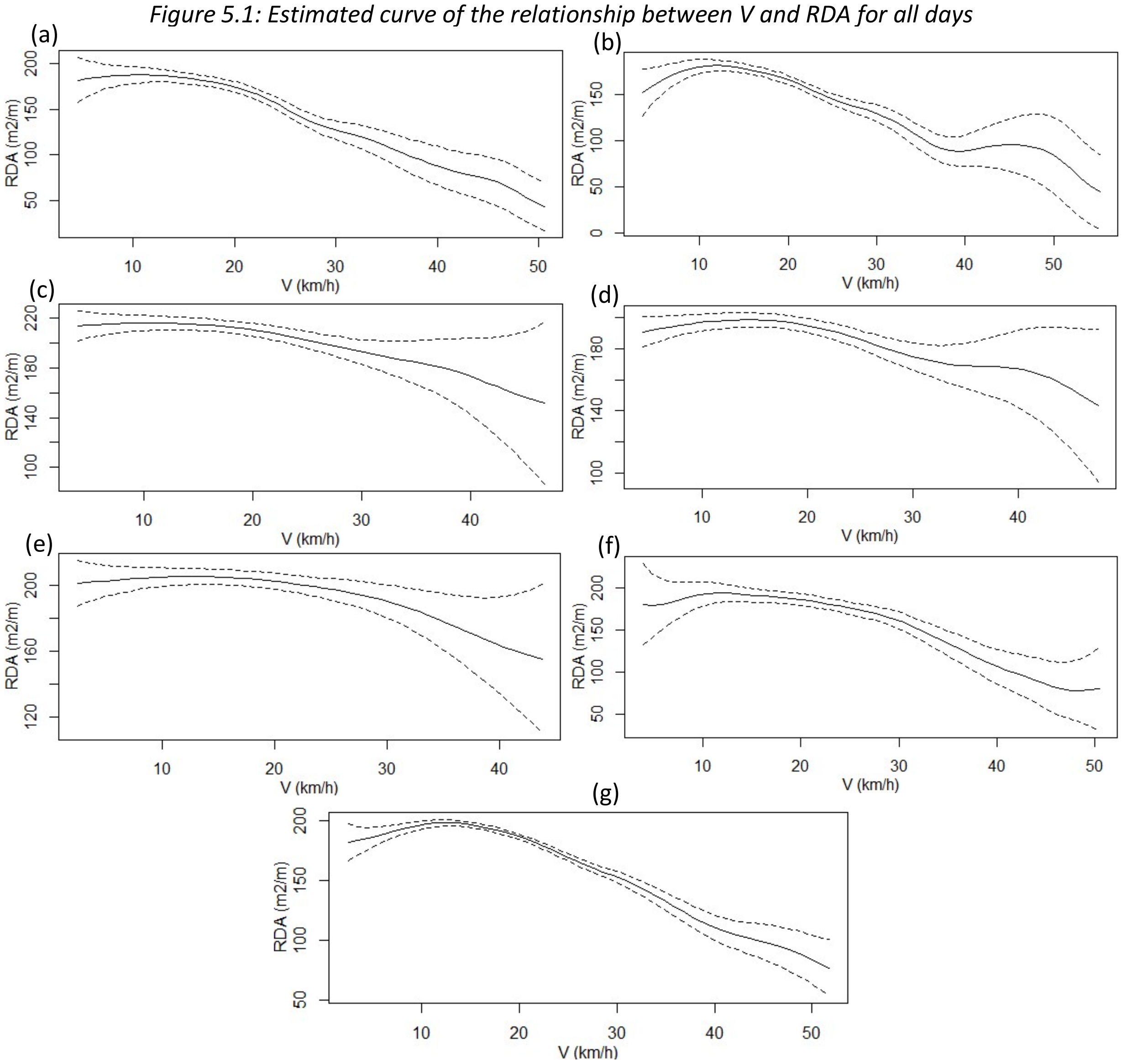


Figure 5.2: Resulting regression curves of the relation between V and RDA for a)Monday, b)Tuesday, c)Wednesday, d)Thursday, e)Friday, f)Sunday, and g)All days

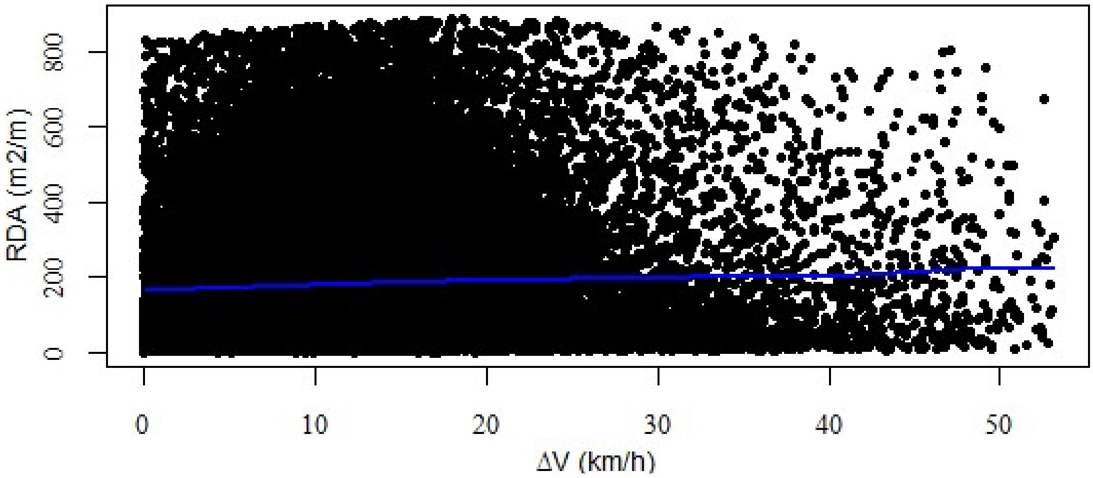


Figure 5.3: Estimated curve of the relationship between ΔV and RDA for all days

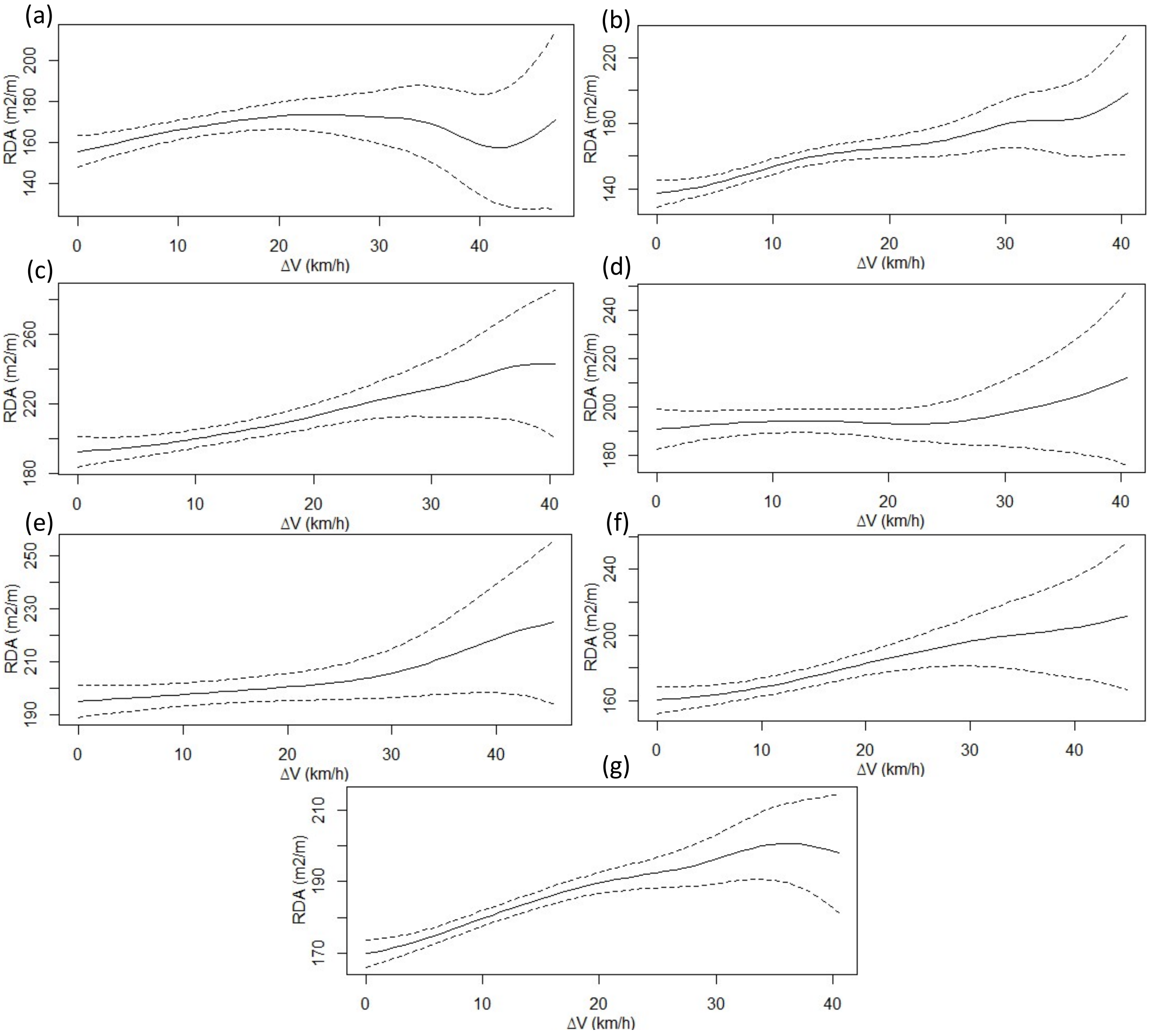


Figure 5.4: Resulting regression curves of the relation between ΔV and RDA for a)Monday, b)Tuesday, c)Wednesday, d)Thursday, e)Friday, f)Sunday, and g)All days

In addition, the relationship between V and RDA is temporally investigated at peak and off-peak periods of one weekday (Wednesday) and one weekend day (Sunday). AM and PM peak periods are determined for each day first. Next, V and RDA values during AM and PM peak periods are combined together as the values for peak periods. Off-peak periods are defined as all day periods other than peak periods. AM peak periods are determined to be 8:00-10:00 and 10:00-12:00 for Wednesday and Sunday respectively, whereas PM peak periods are determined to be 17:00-19:00 and 16:00-18:00 for Wednesday and Sunday respectively. Figure 5.5 and Figure 5.6 show the resulting regression curves for the peak and off-peak periods of Wednesday and Sunday respectively.

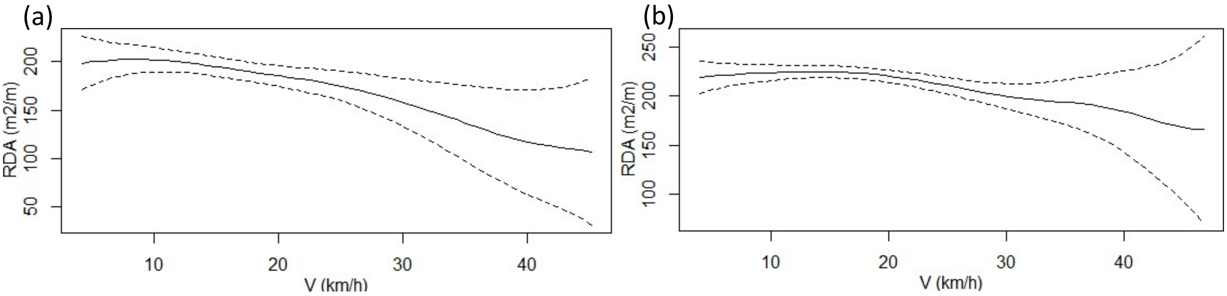


Figure 5.5: Relation between V and RDA at a) peak periods and b) off-peak periods for Wednesday

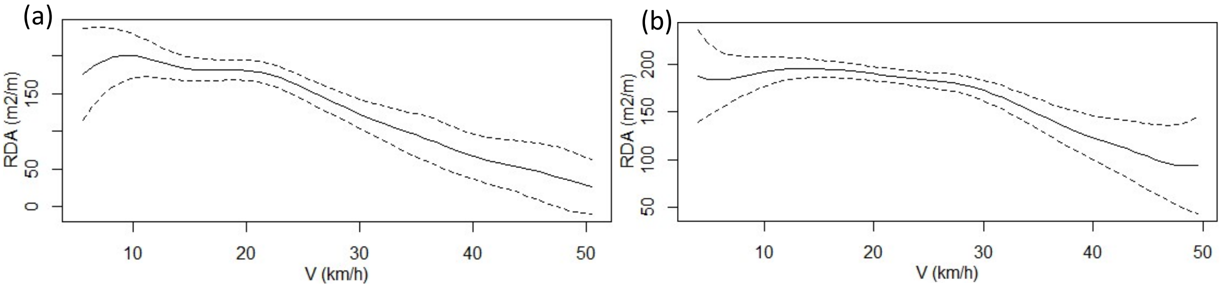


Figure 5.6: Relation between V and RDA at a) peak periods and b) off-peak periods for Sunday

From both figures, it can be concluded that there is no big difference in the regression curve between weekday and weekend day. However, RDA values are generally a bit higher on Wednesday than on Sunday. As for peak and off-peak periods, V values at peak periods are lower than those at off-peak periods for the same value of RDA on both days. For instance, at RDA = 100 m^2/m on Sunday, V = 35 kph at peak periods while V = 45 kph at off-peak periods. One potential reason for this is because even the routes that drivers take trying to aviod the congestion on the fastest routes (references) at peak hours are also congested in varying degrees, and thus thier average speed will not be much better. Another potential explaination might be that some of the drivers who prefer to deviate from the fastest routes during peak periods to avoid congestion on these routes, still do the same during off-peak periods too.

Another case is tested in this thesis where only private passenger cars are considered. Thus, all other types including taxicabs, trucks, and local delivery fleets are excluded. The regression curve of the relationship between V and RDA for this special case on one weekday (Wednesday) is shown in Figure 5.7.

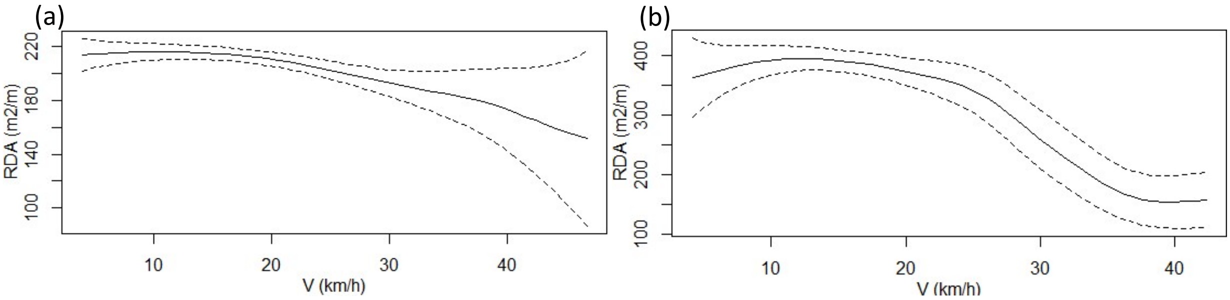


Figure 5.7: Relationship between V and RDA considering a) all vehicle types and b) private cars only on Wednesday

The trend of the regression curves in each case are roughly similar where RDA values slightly increase at first, reach a peak point when V = 10-15 kph, and then steadily decrease. However, RDA values decrease more steeply in private cars case. In addition, RDA values in private cars case are much higher than those in all vehicle types case almost at every V value. One of the reasons behind this might be due to the fact that taxi and local delivery service drivers have better knowledge of the road network and thus are more aware of the fastest routes from one zone to another, which resulting in lower RDA values compared to private cars' drivers.

6. Conclusion

This chapter presents a summary of the research findings and the approach followed in this thesis to obtain them. It also describes the research limitations, provides some recommendations for future work, and indicates potential practical implications of this research.

6.1 Summary

With the widespread use of location sensing technologies such as GPS-enabled vehicles, huge volumes of vehicle trajectory data are increasingly generated. Compared to traditional traffic data obtained from conventional data collection methods like fixed loop detectors, vehicle trajectories provide much richer information. The increasing availability of such data opens up new opportunities for performing more sophisticated and comprehensive spatial and temporal analyses for planning and management of transportation systems. One of the most useful types of analysis in this context is traffic data clustering, which can help in understanding and revealing valuable insights into urban mobility patterns and travel behavior.

In this thesis, a six-day dataset of floating car data from Munich city is clustered to extract meaningful urban mobility patterns. The trajectory dataset contains information like waypoints' GPS coordinates and speed for approximately 100,000 trips. K-means, hierarchical, and DBSCAN are the clustering methods that have been tested on a 10% sample of the dataset's trips. In addition, a theoretical comparison among them is conducted. The comparison and test results showed that hierarchical clustering method performs the best out of the three methods in this context. Therefore, hierarchical clustering analysis, based on Euclidean distance and Ward's linkage method, is used to spatially cluster the trips in each day according to the coordinates of their origin and destination points, such that trips that have similar origin and destination points are put together in one cluster. The number of clusters for each day is defined using special criteria that are developed to determine the optimal number of clusters, taking into consideration clusters' dimensions, number of trips per cluster, and the percentage of clusters with no overlap between their origin and destination zones. The output of this clustering process is clusters of trips, where each cluster represents an OD pair.

To explore mobility patterns and drivers' travel behavior within the resulting clusters, a new tool is presented in this thesis. Relative Deviation Area (RDA) is a tool that aims to find the deviation between two trajectories that share the same origin and destination zone. Ideally, one of the trajectories presents the least-cost route (referential path) between the two zones. For instance, the trajectory that has the highest average speed while traveling from one zone to another is considered as the least-cost route between the two zones in this thesis. RDA computes the relative area by which a vehicle traveling from one zone to another is deviating from the least-cost route, while relative area means that the resulting deviation area is divided by the length of the vehicle's trajectory section that extends between the nearest pair of OD points of the two trajectories. In this research, seven trips that have the highest average speed in each cluster are defined as the referential paths for that cluster. Next, RDA is computed seven times (one time with each referential path) for each trip in each cluster, such that the minimum value out of the seven values is considered as the final RDA value for that trip.

Subsequently, all resulting RDA and V values from all clusters are aggregated together in each day, as it was not possible to separately investigate each cluster or each zone in the context of this thesis. The relationships between V and RDA and between ΔV and RDA are investigated for each day attempting to understand drivers' travel behavior, where ΔV is the average speed difference between a given path and the referential path that gives the minimum RDA value. Before applying nonparametric regression to understand these relationships, outliers are detected and excluded using bagplot method. Afterward, Kernel regression method is applied to investigate the aforementioned relationships for each day and for all days together as well. It was found that the resulting regression curves in both relationships are almost consistent throughout all weekdays and weekend day as well. In the relation between V and RDA, it was found that V, generally, decreases as RDA increases. In other words, it was noticed that the higher the deviation area between the route that a vehicle takes and the respective fastest route between two zones, the lower the trip average speed the vehicle can achieve. As for the other relation between ΔV and RDA, it was found that ΔV increases as RDA increases. This means that the higher the deviation area between a trip's path and the respective fastest path, the higher the average speed difference between them, and thus the lower the average speed of that trip. The resulting regression curves in both cases are found to be sensible and consistent; therefore, this might indicate a potential association between deviation area and trip average speed. However, this relationship should be deeper investigated and validated by applying the presented methodology on different datasets in different locations for example.

In addition, the relationship between V and RDA is temporally investigated at peak and off-peak periods of one weekday and one weekend day. The regression curves suggest no big difference between weekday and weekend day, where RDA values are only a bit higher on weekday than on weekend day. With regard to peak and off-peak periods, V values at peak periods are found to be lower than those at off-peak periods for the same value of RDA on both days. Another case is tested where only private cars are considered on one weekday, excluding all other vehicle types like taxicabs and trucks. The results showed that RDA values in private cars case are much higher and decrease more steeply than for those in all vehicle types case.

6.2 Limitations and Recommendations

Some of the limitations of the work carried out in this thesis, mainly due to time and computational cost constraints, and some recommendations concluded from these limitations are listed below. In future research, an attempt should be made to address these limitations in order to improve the reliability and validity of the predicted results.

- The resulting clusters that have an overlap between their own origin and destination zones are excluded, as it is required to have only one major traffic flowline for each cluster from one origin zone to another fully separated destination zone. It would be ideal if it was possible to control clustering process such that only clusters that have fully separated origin and destination zones are created. In this way, no clusters have to be excluded and no trips are overlooked.
- In RDA computation, paths are simplified by removing redundant points that lie on the same line, or that do not contribute significantly to the shape of the path. Although this step noticeably improved the efficiency and reduced the computational costs, the results would be a bit more accurate if all points of each path were considered.
- Waypoints in the raw vehicle trajectory data are recorded unevenly and at relatively long-time intervals which range from 1 second to 3 minutes with a median of 1 minute. It would be better for further studies to use trajectory datasets with higher waypoints frequency such that the distance between two consecutive points can be reasonably and safely approximated by a straight line connecting the two points. This would help producing more reliable RDA values.
- As the number of the resulting clusters in each day is high, and due to thesis time limit, all resulting RDA and V values from all clusters are aggregated together for each day prior to studying travel behavior and investigating different relationships. For further studies, it will be better and more comprehensive to investigate travel behavior on a cluster-level.

- In this thesis, trips are spatially clustered based on the coordinates of their origin and destination points. However, for further studies, it might be better to spatiotemporally cluster trips, especially if it desired to extract and analyze temporal mobility patterns.
- GPS coordinates for trips' start and end points in the raw trajectory data are accurate to the 3rd decimal place only. Since the trips are clustered based on these points, the clustering results would be more accurate if the precision of their coordinates was higher, although the current precision didn't affect much the clustering results.
- This thesis didn't take into consideration the semantic meanings of the resulting clusters as this was not very relevant in the context of the thesis. Nevertheless, it would be ideal to find out where exactly the resulting clusters are and what kind of semantic category they belong to by projecting the resulting clusters on the map.
- Outlying trips that produce very extreme RDA values are detected and excluded in this thesis, but it is not guaranteed that all trips with abnormal travel behavior are detected. However, the best solution will be to find a way to handle these outliers instead of excluding them.
- Three clustering methods are tested in this thesis, and consequently hierarchical clustering was used at the end. For further studies, other clustering techniques might be tested and applied in this context as they perhaps will produce better clustering results.

List of References

Ahuja, S. (2014). Clustering in expert system. In: Marwaha, S., Malhora, P.K., Goyal, R.C., Arora, A., Singh, P., editors. Development of expert system in agriculture [E-book]. New Delhi:Indian Agricultural Statistics Research Institute (ICAR). Retrieved from: http://www.iasri.res.in/ebook/expertsystem/Home.htm

Altintasi, O., Tuydes-Yaman, H., & Tuncay, K. (2017). Detection of urban traffic patterns from Floating Car Data (FCD). Transportation Research Procedia, 22. 382-391. Retrieved from: https://www.sciencedirect.com/science/article/pii/S235214651730193X

Anand, S., Padmanabham, P., Govardhan, A., & Kulkarni, R.H. (2017). An Extensive Review on Data Mining Methods and Clustering Models for Intelligent Transportation System. Journal of Intelligent Systems, 27(2), 263-273.

Ayala, D., Lin, J., Wolfson, O., Rishe, N., & Tanizaki, M. (2010). Communication reduction for floating car data-based traffic information systems. In 2nd International Conference on Advanced Geographic Information Systems, Applications, and Services, GEOProcessing 2010 44-51. Retrieved from: https://doi.org/10.1109/GEOProcessing.2010.14

Banfield, J. & Raftery, A. (1993). Model-Based Gaussian and Non-Gaussian Clustering. Biometrics, 49(3), 803-821.

Bergroth, L., Hakonen, H., & Raita, T. (2000). A survey of longest common subsequence algorithms, in: Seventh International Symposium on String Processing and Information Retrieval, 2000. SPIRE 2000. Proceedings. 39–48.

Bock, T. (n.d.). What is Hierarchical Clustering?. Retrieved from: https://www.displayr.com/what-is-hierarchical-clustering/

Brockfeld, E. ,Wagner, P., Lorkowski, S., Mieth, P. (2007). Benefits and limits of recent floating car data technology—An evaluation study. In: Proc. 11th World Conf. Transp. Res., Berkeley, CA, USA. Retrieved from: https://elib.dlr.de/49618/1/Brockfeld_WCTR2007.pdf

Buliung, R. N. & Kanaroglou, P. S. (2007). Activity–travel behaviour research: conceptual issues, state of the art, and emerging perspectives on behavioural analysis and simulation modeling. Transport Reviews, 27, 151-187.

Burnos, P., Gajda, J., Piwowar, P., Sroka, R., Stencel, M., & Zeglen, T. (2007). Measurements of Road Traffic Parameters Using Inductive Loops and Piezoelectric Sensors. Metrology and Measurement Systems. 14(2). 187–203. Retrieved from: https://depot.ceon.pl/bitstream/handle/123456789/6514/Measurements-of-Road-Traffic-Parameters-Using-Inductive-Loops-and-Piezoelectric-Sensors-Burnos-Gajda.pdf;sequence=1

Calabrese, F., Di Lorenzo, G., Liu, L., & Ratti, C. (2011). Estimating origin-destination flows using mobile phone location data. IEEE Pervasive Computing, 10, 36-44. Retrieved from: https://www.researchgate.net/publication/297712660_Estimating_Origin-Destination_flows_using_opportunistically_collected_mobile_phone_location_data_from_one_million_users_in_Boston_Metropolitan_Area

Chatterjee, R. (2012). An Analytical Assessment on Document Clustering. International Journal of Computer Network and Information Security (IJCNIS), 4, 63-71. Retrieved from:

http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=78B03393FD2B0DDA0782EFA4175FDFD5?doi=10.1.1.1016.4710&rep=rep1&type=pdf

Chen, J., Hu, T., Zhang, P., & Shi, W. (2014). Trajectory Clustering for People's Movement Pattern Based on Crowd Souring Data. ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. XL-2. 55-62. Retrieved from: https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XL-2/55/2014/isprsarchives-XL-2-55-2014.pdf

Colak, S., Alexander, L.P., Alvim, B.G., Mehndiratta, S.R., & Gonzalez, M.C. (2015). Analyzing cell phone location data for urban travel: current methods, limitations, and opportunities. Transportation Research Record: Journal of the Transportation Research Board, 2526, 126-135. Retrieved from: http://humnetlab.mit.edu/wordpress/wp-content/uploads/2010/10/TRB_finaldraft.pdf

CTI Reviews (2016). Sustainable Transportation, Problems and Solutions: Engineering, Urban studies and planning. Retrieved from: http://books.google.com

Ding, L., Jahnke, M., Wang, S. & Karja, K. (2016). Understanding Spatiotemporal Mobility Patterns related to Transport Hubs from Floating Car Data. In: Proc. LBS 2016, Vienna. Retrieved from: http://lbs2016.lbsconference.org/wp-content/uploads/2016/11/3_9.pdf

Ding, L., Meng, L., Yang, J., & Krisp, J. (2018). Interactive visual exploration and analysis of origin-destination data. Proceedings of the ICA. 1. 1-5. Retrieved from: https://www.proc-int-cartogr-assoc.net/1/29/2018/ica-proc-1-29-2018.pdf

Dogru, N. & Subasi, A. (2014). Comparison of clustering techniques for traffic accident detection. Turkish Journal of Electrical Engineering and Computer Sciences. 23. 10.3906/elk-1304-234. Retrieved from: http://journals.tubitak.gov.tr/elektrik/issues/elk-15-23-sup.1/elk-23-sup.1-7-1304-234.pdf

Ester, M., Kriegel, H., Sander, J., & Xu, X. (1996). A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. KDD. Retrieved from: https://www.aaai.org/Papers/KDD/1996/KDD96-037.pdf

Fahim, A., Saake, G., M.Salem, A., Torkey, F.A., & Ramadan, M. (2008). K-Means for Spherical Clusters with Large Variance in Sizes. Int J Comput Sci. 2. 113-118.

Fang, J., Xue, M., & Qiu, T.Z. (2014). Anonymous Cellphone-Based LargeScale Origin–Destination Data Collection: Case Studies in China. 93rd Annual Meeting of the Transportation Research Board, Washington, D.C. Retrieved from: https://businessdocbox.com/Logistics/74054561-93rd-annual-meeting-final-program.html

French, S., Barchers, C., & Zhang, W. (2015). Moving beyond Operations: Leveraging Big Data for Urban Planning Decisions. 56th Annual Conference of Association of College Schools of Planning (ACSP), Portland, OR. Retrieved from: http://web.mit.edu/cron/project/CUPUM2015/proceedings/Content/pss/194_french_h.pdf

Ghuman, S.P. (2016). Clustering Techniques - A Review. International Journal of Computer Science and Mobile Computing, Vol.5 Issue.5, May- 2016, 524-530. Retrieved from: https://www.ijcsmc.com/docs/papers/May2016/V5I5201699a5.pdf

Guo, D. & Zhu, X. (2014). Origin-Destination Flow Data Smoothing and Mapping. IEEE Transactions on Visualization and Computer Graphics, 20, 2043-2052. Retrieved from: http://vis.cs.ucdavis.edu/vis2014papers/TVCG/papers/2043_20tvcg12-guo-2346271.pdf

Guo, D., Zhu, X., Jin, H., Gao, P., & Andris, C. (2012). Discovering Spatial Patterns in Origin-Destination Mobility Data. Trans. GIS, 16, 411-429. Retrieved from:

https://www.researchgate.net/publication/262895654_Discovering_Spatial_Patterns_in_Origin-Destination_Mobility_Data

Hahsler, M., & Piekenbrock, M. (2018). dbscan: Density Based Clustering of Applications with Noise (DBSCAN) and Related Algorithms. R package version 1.1-3. https://CRAN.R-project.org/package=dbscan

Han, J.W., Kamber, M. and Pei, J. (2012). Data Mining: Concepts and Techniques. 3rd Edition, Morgan Kaufmann Publishers, Waltham.

Hayfield, T. & Racine, J.S. (2008). Nonparametric Econometrics: The np Package. Journalbof Statistical Software 27(5). http://www.jstatsoft.org/v27/i05/

Hogg, V.R. & Tanis, A.E. (2005). Probability and statistical inference. Upper Saddle River, USA: Pearson Education.

Hu, J., Qian, Q., Pei, J., Jin, R., & Zhu, S. (2015). Finding Multiple Stable Clusterings. IEEE International Conference on Data Mining, Atlantic City, NJ, 2015, pp. 171-180. Retrieved from: https://www.cs.sfu.ca/~jpei/publications/Clusterability%20ICDM15.pdf

Huang, R. & Sun, S. (2013). Kernel Regression with Sparse Metric Learning. Journal of Intelligent and Fuzzy Systems, 24, 775-787. Retrieved from: https://arxiv.org/pdf/1712.09001.pdf

Huang, W., Li, S., & Xu, S. (2016). A three-step spatial-temporal-semantic clustering method for human activity pattern analysis. ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. XLI-B2. 549-552. Retrieved from: https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLI-B2/549/2016/isprs-archives-XLI-B2-549-2016.pdf

Huber, W., Lädke, M., & Ogger, R. (1999). Extended floating-car data for the acquisition of traffic information. In Proceedings of the 6th World Congress on Intelligent Transport Systems.

INRIX. (2018). About INRIX. Retrieved from: http://inrix.com/about/

Iqbal, M.S., Choudhury, C.F., Wang, P., & González, M.C. (2014). Development of Origin–Destination Trip Matrices Using Mobile Phone Call Data. Transportation Research Part C: Emerging Technologies, 2014, 63–74. Retrieved from: https://pdfs.semanticscholar.org/5100/694c0919bf1268f776f8f89c18853f0e68b3.pdf

Jambhorkar, S. & Jondhale, V. (2015). Data Mining Technique: Fundamental Concept and Statistical Analysis. Retrieved from: http://books.google.com

Jones, M., Geng, Y., Nikovski, D., & Hirata, T. (2013). Predicting link travel times from floating car data. IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC. 1756-1763. Retrieved from: https://merl.com/publications/docs/TR2013-095.pdf

Kassambara, A. (2017). Determining The Optimal Number Of Clusters: 3 Must Know Methods. Retrieved from: https://www.datanovia.com/en/lessons/determining-the-optimal-number-of-clusters-3-must-know-methods/

Kaur, M. & Kaur, U. (2013). Comparison between K-mean and hierarchical algorithm using Query Redirection. Int. J. Adv. Res. Comput. Sci. Software Eng.. 3. 1454-1459. Retrieved from: https://pdfs.semanticscholar.org/5246/5a9ccef4f5dbc0b767c4c4d7aa91cbdfabc5.pdf

Kaushik, M. & Mathur, B. (2014). Comparative Study of K-Means and Hierarchical Clustering Techniques. Internatioonal journal of Software and Hardware Research in Engineering. 2. 93-98. Retrieved from: https://www.researchgate.net/publication/293061584_Comparative_Study_of_K_Means_and_Hierarchical_Clustering_Techniques

Kerner, B.S., Demir, C., Herrtwich, R.G., Klenov, S.L., Rehborn, H., Aleksic, M., & Haug, A. (2005). Traffic state detection with floating car data in road networks. Proceedings. 2005 IEEE Intelligent Transportation Systems, Vienna, 44-49. Retrieved from: https://ieeexplore-ieee-org.eaccess.ub.tum.de/document/1520133

Kim, J. & Mahmassani, H. (2015). Spatial and Temporal Characterization of Travel Patterns in a Traffic Network Using Vehicle Trajectories. Transportation Research Procedia. 9. 164-184. Retrieved from: https://www.sciencedirect.com/science/article/pii/S2352146515001702

Kumar, D., Wu, H., Lu, Y., Krishnaswamy, S., & Palaniswami, M. (2016). Understanding Urban Mobility via Taxi Trip Clustering. 17th IEEE International Conference on Mobile Data Management (MDM), Porto, 2016, 318-324. Retrieved from: https://ieeexplore.ieee.org/document/7517811

Kumar, D., Wu, H., Rajasegarar, S., Leckie, C., Krishnaswamy, S., & Palaniswami, M. (2018). Fast and Scalable Big Data Trajectory Clustering for Understanding Urban Mobility. IEEE Transactions on Intelligent Transportation Systems. 1-14. Retrieved from: https://ieeexplore.ieee.org/document/8424474

Kumar, S., Nageswara Rao, K., Govardhan, A., & Sandhya, N. (2015). Subset K-Means Approach for Handling Imbalanced-Distributed Data. Advances in Intelligent Systems and Computing. 338. 497-508.

Landeshauptstadt München Planungsreferat (2013). Parkraummanagement in der Landeshauptstadt München. [PDF map]. Retrieved from: https://www.muenchen.de/rathaus/Stadtverwaltung/Kreisverwaltungsreferat/Verkehr/Parkraummanagement/Strassenauskunft.html

Latecki, L.J. & Lakämper, R. (2000). Shape similarity measure based on correspondence of visual parts. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(10), 1185–1190.

Leduc, G. (2008). Road Traffic Data: Collection Methods and Applications. Working Papers on Energy, Transport and Climate Change, 1(55). Retrieved from: ftp://ftp.jrc.es/pub/EURdoc/JRC47967.TN.pdf

Levinson, D., & Shanjiang, Z. (2013). A Portfolio Theory of Route Choice. Transportation Research Part C: Emerging Technologies, 232–243. Retrieved from: https://pdfs.semanticscholar.org/4a52/489a610d0a19ad5d2a060c6b30b8c99d83c9.pdf

Lian, J., Li, Y., Gu, W., Huang, S.L., & Zhang, L. (2018). Joint Mobility Pattern Mining with Urban Region Partitions. Retrieved from: https://www.researchgate.net/publication/329182399_Joint_Mobility_Pattern_Mining_with_Urban_Region_Partitions

Lin, L. (2015). Data science application in intelligent transportation systems: An integrative approach for border delay prediction and traffic accident analysis (Doctoral dissertation, SUNY at Buffalo, Buffalo, NY, 2015). Retrieved from: https://pqdtopen.proquest.com/doc/1658211320.html?FMT=AI

Liu, J., Li, J., Li, W., & Wu, J. (2015). Rethinking big data: A review on the data quality and usage issues. ISPRS Journal of Photogrammetry and Remote Sensing, 115, 134–142. Retrieved from: https://pdfs.semanticscholar.org/245c/e366bd869fea87b6b8d50cc824d817f08fbd.pdf?_ga=2.188166632.1536350496.1544648163-1820800008.1542668704

Liu, L., Biderman, A., & Ratti, C. (2009). Urban Mobility Landscape : Real Time Monitoring of Urban Mobility Patterns. Proc. 11th Int. Conf. Comput. Urban Plan. Urban Manag, 1–16.

Liu, X. & Ban, Y. (2013). Uncovering Spatio-Temporal Cluster Patterns Using Massive Floating Car Data. ISPRS International Journal of Geo-Information. 2. 371-384. Retrieved from: https://pdfs.semanticscholar.org/2884/6b55fe1710373763d9ad67ffc15a4364fd91.pdf

Mahmoud, H.F. (2014). Parametric versus Semi/nonparametric Regression Models. [Class handout]. Retrieved from: https://www.colorado.edu/lab/lisa/services/short-courses/parametric-versus-seminonparametric-regression-models

Maimon, O. & Rokach, L. (2005). Data Mining and Knowledge Discovery Handbook: Springer-Verlag. Retrieved from: https://www.cs.swarthmore.edu/~meeden/cs63/s16/reading/Clustering.pdf

Mitchell, D. (2014). New traffic data sources–An overview. BITRE/IPA joint workshop: Exploring New Sources of Traffic Data, Sydney. Retrieved from: https://bitre.gov.au/events/2014/files/NewDataSources-BackgroundPaper-April%202014.pdf

Moise, I. & Pournaras, E. (2017). Density-Based Clustering. [Powerpoint slides]. Retrieved from: https://www.ethz.ch/content/dam/ethz/special-interest/gess/computational-social-science-dam/documents/education/Spring2017/Data_science/course3_1.pdf

Montello, D.R. & Goulias, K.G. (2018). Handbook of Behavioral and Cognitive Geography. Retrieved from: http://books.google.com

Mueller, S. T., Perelman, B. S., & Veinott, E. S. J. B. R. M. (2016). An optimization approach for mapping and measuring the divergence and correspondence between paths. 48(1), 53-71. Retrieved from: https://link.springer.com/article/10.3758/s13428-015-0562-7

Mueller, S.T., Perelman, B.S., & Veinott, E.S. (2016). An optimization approach for mapping and measuring the divergence and correspondence between paths. Behav Res 48-53. Retrieved from: https://doi.org/10.3758/s13428-015-0562-7

Neethu, C.V. & Surendran, S. (2013). Review of Spatial Clustering Methods. International Journal of Information Technology Infrastructure, Volume 2, No.3, May–June 2013. Retrieved from: http://warse.org/pdfs/2013/ijiti01232013.pdf

Rai, P. (2011). Machine Learning, Data Clustering: K-means and Hierarchical Clustering. [Powerpoint slides]. Retrieved from: https://www.cs.utah.edu/~piyush/teaching/4-10-print.pdf

Rohde, J. (2007). Evaluation of interest point detectors in content-based image retrieval (Doctoral dissertation, ITU university of Copenhagen). Retrieved from: https://pdfs.semanticscholar.org/626a/95070387fa055cfce0c3ba8348729bd9d1b9.pdf

Rojas, Mario & Sadeghvaziri, Eazaz & Jin, Xia. (2016). Comprehensive Review of Travel Behavior and Mobility Pattern Studies That Used Mobile Phone Data. Transportation Research Record: Journal of the Transportation Research Board. 2563. 71-79. 10.3141/2563-11.

Rokib, S.A., Karim, M.A., Qiu, T.Z., & Kim, A. (2015) Origin–Destination Trip Estimation from Anonymous Cell Phone and Foursquare Data. 94th Annual Meeting of the Transportation Research Board, Washington, D.C.

Rousseeuw, P., Ruts, I., & Tukey, J.W. (1999). The Bagplot: A Bivariate Boxplot. American Statistician - AMER STATIST. 53. 382-387. Retrieved from:

https://www.researchgate.net/profile/Peter_Rousseeuw/publication/247788661_The_Bagplot_A_Bivariate_Boxplot/links/5b2bbfb9a6fdcc8506b709cf/The-Bagplot-A-Bivariate-Boxplot.pdf

Santini, M. (2016). Machine Learning for Language Technology: Advantages & Disadvantages of k-Means and Hierarchical clustering [Powerpoint slides]. Retrieved from: http://stp.lingfil.uu.se/~santinim/ml/2016/Lect_10/10c_UnsupervisedMethods.pdf

Seif, G. (2018). The 5 Clustering Algorithms Data Scientists Need to Know. Retrieved from: https://towardsdatascience.com/the-5-clustering-algorithms-data-scientists-need-to-know-a36d136ef68

Sonagara, D. & Badheka, S. (2014). Comparison of Basic Clustering Algorithms. International Journal of Computer Science and Mobile Computing, Vol.3 Issue.10, October- 2014, pg. 58-61. Retrieved from: https://pdfs.semanticscholar.org/f0a4/d6bfb37b6c1102f7ef6b0d0f2ef861da6aca.pdf

Spissu, E., Meloni, I., & Sanjust, B. (2011). Behavioral Analysis of Choice of Daily Route with Data from Global Positioning System. In Transportation Research Record: Journal of the Transportation Research Board, No. 2230, Transportation Research Board of the National Academies, Washington, D.C., 96–103. Retrieved from: https://journals.sagepub.com/doi/pdf/10.3141/2230-11

Stiftelsen for industriell og teknisk forskning [SINTEF], (n.d.). Travel Behaviour. Retrieved from: https://www.sintef.no/en/traffic-behaviour/

Sun, J., Zhang, C., Zhang, L., Chen, F., & Peng, Z.R. (2014). Urban travel behavior analyses and route prediction based on floating car data. Transportation Letters, 6, 118-125. Retrieved from: https://www.researchgate.net/publication/280786953_Urban_travel_behavior_analyses_and_route_prediction_based_on_floating_car_data

Tan, P., Kumar, V., & Steinbach, M. (2005). Introduction to data mining (1st ed). Pearson Addison Wesley, Boston.

Tawfik, A.M. & Rakha, H.A. (2012). Network Route-Choice Evolution in a Real-World Experiment: Necessary Shift from Network- to DriverOriented Modeling. In Transportation Research Record: Journal of the Transportation Research Board, No. 2322, Transportation Research Board of the National Academies, Washington, D.C., 70–81. Retrieved from: https://journals.sagepub.com/doi/pdf/10.3141/2322-08

Tibshirani, R., Walther, G., & Hastie, T. (2001). Estimating the Number of Clusters in a Data Set Via the Gap Statistic. Journal of the Royal Statistical Society Series B. 63. 411-423. Retrieved from: https://statweb.stanford.edu/~gwalther/gap

Tibshirani, R. & Wasserman, L. (2015). Statistical Machine Learning: Nonparametric Regression [Class handout]. Retrieved from: http://www.stat.cmu.edu/~larry/=sml/nonpar.pdf

Toohey, K. (2015). SimilarityMeasures: Trajectory Similarity Measures. R package version 1.4. https://CRAN.R-project.org/package=SimilarityMeasures

Wang, M.H., Schrock, S.D., Broek, N.V., & Mulinazzi, T. (2013). Feasibility of Using Cellular Phone Location Data in Traffic Survey on Intercity Trips. 92nd Annual Meeting of the Transportation Research Board, Washington, D.C. Retrieved from: https://www.sciencedirect.com/science/article/pii/S2352146517307366

Wang, S. (2015). Spatiotemporal Visual Analysis of Traffic Flow Patterns Related to Transport Hubs from Floating Car Data (Master's Thesis). Retrieved from:

http://cartographymaster.eu/wp-content/theses/2016_Wang_Thesis.pdf

Wang, Z., He, S., & Leung, Y. (2017). Applying mobile phone data to travel behaviour research: A literature review. Travel Behaviour and Society. Retrieved from: http://isiarticles.com/bundles/Article/pre/pdf/113266.pdf

Wolf, J., Oliveira, M., & Thompson, M. (2003). Impact of Underreporting on Mileage and Travel Time Estimates: Results from Global Positioning System–Enhanced Household Travel Survey. In Transportation Research Record: Journal of the Transportation Research Board, No. 1854, Transportation Research Board of the National Academies, Washington, D.C., 2003, 189–198.

Ye, Y., Zheng, Y., Chen, Y., Feng, J., & Xie, X. (2009). Mining individual life pattern based on location history. The 10th International Conference on Mobile Data Management: Systems, Services and Middleware, IEEE. Retrieved from: https://www.researchgate.net/publication/221422556_Mining_Individual_Life_Pattern_Based_on_Location_History

Yue, Y., Lan, T., Yeh, A.G., & Li, Q.Q. (2014). Zooming into individuals to understand the collective: A review of trajectory-based travel behaviour studies. Travel Behaviour and Society, 1, 69-78.

Yue, Y., Zhuang, Y., Li, Q., & Mao, Q. (2009). Mining time-dependent attractive areas and movement patterns from taxi trajectory data. 17th International Conference on Geoinformatics, Fairfax, VA, 2009, 1-6. Retrieved from: https://ieeexplore-ieee-org.eaccess.ub.tum.de/document/5293469

Zaiane, O. (1999). Principles of Knowledge Discovery in Databases: Data Clustering [Powerpoint slides], University of Alberta. Retrieved from: https://webdocs.cs.ualberta.ca/~zaiane/courses/cmput690/slides/Chapter8/index.html

Zamani, Z., Pourmand, M., & Saraee, M. H. (2010). Application of data mining in traffic management: Case of city of Isfahan. 2nd International Conference on Electronic Computer Technology, Kuala Lumpur, 2010, 102-106. Retrieved from: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.389.1640&rep=rep1&type=pdf

Zhang, L., Hong, J.H., Nasri, A., & Shen, Q. (2012). How built environment affects travel behavior: A comparative analysis of the connections between land use and vehicle miles traveled in US cities. Journal of Transport and Land Use, 5, 40–52.