

POLITECNICO DI TORINO

Dipartimento di Elettronica e delle Telecomunicazioni

Corso di Laurea Magistrale in "ICT for Smart Societies"

Tesi di Laurea Magistrale

# User habits analysis in a bike sharing system



**Relator:**

Prof.re Marco Mellia

Danilo Giordano

Luca Vassio

**Candidate:**

Martina Mineo

*Ai miei zii Anna e Pippo*





# Acknowledgements

Il mio primo grazie va ai miei genitori che mi hanno permesso di vivere questa meravigliosa esperienza.

Ringrazio di cuore le mie tre amiche più care Greta, Mila e Francesca il quale aiuto è stato prezioso per lo svolgimento di questo lavoro, loro mi hanno sempre sostenuta e appoggiata a superare tutti gli ostacoli.

Un caloroso grazie va a Iacopo che mi ha sempre spronata ad andare avanti specialmente nei momenti più difficili. Il suo aiuto, non solo morale, è stato fondamentale per raggiungere questo traguardo.

Infine grazie ad Alessandra, Linda, Geppina, Jana, la piccola Clara, e tutte quelle persone che ho trovato durante il mio percorso, i quali consigli mi hanno aiutata a crescere e migliorare.



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Overview . . . . .	1
1.2	Goal of the thesis . . . . .	2
1.3	Motivation . . . . .	4
1.4	Literature review . . . . .	4
1.4.1	User habits analysis . . . . .	5
1.4.2	Reallocation problem . . . . .	8
1.4.3	Usage prediction problem . . . . .	11
1.5	Thesis organization . . . . .	12
<b>2</b>	<b>Datasets</b>	<b>13</b>
2.1	Dataset acquisition from real-time flows . . . . .	14
2.1.1	Real-time data acquisition . . . . .	15
2.1.2	Real-time data analysis . . . . .	23
2.1.3	Conclusions on real-time data . . . . .	25
2.2	Open historical dataset usage . . . . .	25
2.2.1	Dataset overview . . . . .	26
<b>3</b>	<b>Methodology</b>	<b>29</b>
3.1	Used tools . . . . .	29
3.1.1	Python . . . . .	29
3.1.1.1	Pandas . . . . .	30
3.2	Preparation of the data . . . . .	30
3.3	Exploration of the dataset . . . . .	32

	Contents
<b>4 Analysis and results</b>	<b>42</b>
4.1 Temporal analysis . . . . .	42
4.2 Spatial analysis . . . . .	49
<b>5 Conclusion</b>	<b>57</b>
<b>Appendices</b>	<b>60</b>
<b>A Additional plots and tables</b>	<b>61</b>
A.1 Statistical analysis . . . . .	61
A.2 Temporal analysis . . . . .	64
A.3 Spatial analysis . . . . .	76
<b>Bibliography</b>	<b>87</b>

# List of Figures

1.1	Example of station-based bike sharing . . . . .	2
1.2	Example of free-floating bike sharing . . . . .	2
1.3	Dockless Weekday Rider Peaks (October 2017 - January 2018) (Source: [5]) . . . . .	6
1.4	Dockless Weekend Rider Peaks (October 2017 - January 2018) (Source: [5]) . . . . .	6
1.5	Pattern of the demand factor, the O/D factor and the idle time factor (Source: [11]) . . . . .	9
2.1	Parameters for the first POST request . . . . .	15
2.2	Parameters for the second POST request . . . . .	16
2.3	Parameters for the third POST request . . . . .	16
2.4	Response to the third POST request . . . . .	17
2.5	Example of response . . . . .	17
2.6	“Ofo” coverage (Source: [16]) . . . . .	18
2.7	Perimeter of the city of Milan . . . . .	19
2.8	Grid of the city of Milan . . . . .	20
2.9	Final grid of the city of Milan . . . . .	21
2.10	Parallel requests script . . . . .	22
2.11	Response file example . . . . .	22
2.12	Example of crawler result . . . . .	24
2.13	Bookings data table (Source: [4]) . . . . .	27
2.14	Rental zone data table (Source: [4]) . . . . .	27
3.1	Scattered points of Munich . . . . .	33

## List of Figures

3.2	Map of Munchen with the rental zones . . . . .	33
3.3	Scattered points of Stuttgart . . . . .	33
3.4	Map of Stuttgart with the rental zones . . . . .	33
3.5	Scattered points of Frankfurt . . . . .	33
3.6	Map of Frankfurt with the rental zones . . . . .	33
3.7	Scattered points of Hamburg . . . . .	34
3.8	Map of Hamburg with the rental zones . . . . .	34
3.9	Number of vehicles by years in Munchen . . . . .	35
3.10	Number of vehicles by years in Stuttgart . . . . .	35
3.11	Number of vehicles by years in Frankfurt . . . . .	35
3.12	Number of vehicles by years in Hamburg . . . . .	35
3.13	Number of zones by years in Munchen . . . . .	36
3.14	Number of zones by years in Stuttgart . . . . .	36
3.15	Number of zones by years in Frankfurt . . . . .	36
3.16	Number of zones by years in Hamburg . . . . .	36
3.17	Number of bookings by years in Munchen . . . . .	38
3.18	Number of bookings by years in Stuttgart . . . . .	38
3.19	Number of bookings by years in Frankfurt . . . . .	38
3.20	Number of bookings by years in Hamburg . . . . .	38
3.21	Average duration for each city . . . . .	39
3.22	Median of duration for each city . . . . .	40
3.23	90th percentile of duration for each city . . . . .	41
4.1	Bookings during the day of Frankfurt . . . . .	43
4.2	Bookings during the week of Frankfurt . . . . .	44
4.3	Monthly bookings trend of Frankfurt by year . . . . .	45
4.4	Bookings by month of Frankfurt . . . . .	46
4.5	Zoom of the main peaks in monthly booking trend of Frankfurt . . . . .	47
4.6	Statistics of the peak hours of Frankfurt . . . . .	48
4.7	Statistics of the peak hours of Stuttgart . . . . .	48
4.8	Statistics of the peak hours of Hamburg . . . . .	48
4.9	Distance trip of Munchen . . . . .	50
4.10	Distance trip of Stuttgart . . . . .	50
4.11	Distance trip of Frankfurt . . . . .	50

4.12	Distance trip of Hamburg . . . . .	50
4.13	CDF of booking trip by city . . . . .	51
4.14	Origin/Destination matrix . . . . .	52
4.15	CDF of total bookings for each zone of Hamburg . . . . .	52
4.16	CDF of total parkings for each zone of Hamburg . . . . .	52
4.17	CDF of total bookings for each zone of Stuttgart . . . . .	53
4.18	CDF of total parkings for each zone of Stuttgart . . . . .	53
4.19	Voronoi map of bookings (Hamburg-8 a.m.) . . . . .	54
4.20	Voronoi map of parkings (Hamburg-8 a.m.) . . . . .	54
4.21	Voronoi map of bookings (Hamburg-12 p.m.) . . . . .	54
4.22	Voronoi map of parkings (Hamburg-12 p.m.) . . . . .	54
4.23	Voronoi map of bookings (Hamburg-5 p.m.) . . . . .	54
4.24	Voronoi map of parkings (Hamburg-5 p.m.) . . . . .	54
4.25	Voronoi booking zones (Hamburg-8 a.m.) . . . . .	55
4.26	Voronoi parking zones (Hamburg-8 a.m.) . . . . .	55
4.27	Voronoi booking zones (Hamburg-5 p.m.) . . . . .	55
4.28	Voronoi parking zones (Hamburg-5 p.m.) . . . . .	55
A.1	Bookings during the day of Munchen . . . . .	64
A.2	Bookings during the day of Stuttgart . . . . .	64
A.3	Bookings during the day of Hamburg . . . . .	65
A.4	Bookings during the week of Hamburg . . . . .	65
A.5	Bookings during the week of Stuttgart . . . . .	66
A.6	Bookings during the week of Hamburg . . . . .	66
A.7	Monthly booking trend of Hamburg 2014 . . . . .	67
A.8	Monthly booking trend of Hamburg 2015 . . . . .	67
A.9	Monthly booking trend of Hamburg 2016 . . . . .	68
A.10	Monthly booking trend of Stuttgart 2014 . . . . .	68
A.11	Monthly booking trend of Stuttgart 2015 . . . . .	69
A.12	Monthly booking trend of Stuttgart 2016 . . . . .	69
A.13	Bookings by month of Munchen . . . . .	70
A.14	Bookings by month of Stuttgart . . . . .	71
A.15	Bookings by month of Hamburg . . . . .	72
A.16	Statistics of the peak hours of Frankfurt-2014 . . . . .	73



A.17 Statistics of the peak hours of Frankfurt-2015 . . . . .	73
A.18 Statistics of the peak hours of Stuttgart-2014 . . . . .	74
A.19 Statistics of the peak hours of Stuttgart-2015 . . . . .	74
A.20 Statistics of the peak hours of Hamburg-2014 . . . . .	75
A.21 Statistics of the peak hours of Hamburg-2015 . . . . .	75
A.22 CDF of booking trip of Munchen . . . . .	77
A.23 CDF of booking trip of Stuttgart . . . . .	77
A.24 CDF of booking trip of Frankfurt . . . . .	77
A.25 CDF of booking trip of Hamburg . . . . .	77
A.26 CDF of total bookings for each zone of Munchen . . . . .	78
A.27 CDF of total parkings for each zone of Munchen . . . . .	78
A.28 CDF of total bookings for each zone of Stuttgart . . . . .	78
A.29 CDF of total parkings for each zone of Stuttgart . . . . .	78
A.30 Voronoi map of bookings (Frankfurt-8 a.m.) . . . . .	79
A.31 Voronoi map of parkings (Frankfurt-8 a.m.) . . . . .	79
A.32 Voronoi map of bookings (Frankfurt-12 p.m.) . . . . .	79
A.33 Voronoi map of parkings (Frankfurt-12 p.m.) . . . . .	79
A.34 Voronoi map of bookings (Frankfurt-5 p.m.) . . . . .	79
A.35 Voronoi map of parkings (Frankfurt-5 p.m.) . . . . .	79
A.36 Voronoi map of bookings (Munchen-8 a.m.) . . . . .	80
A.37 Voronoi map of parkings (Munchen-8 a.m.) . . . . .	80
A.38 Voronoi map of bookings (Munchen-12 p.m.) . . . . .	80
A.39 Voronoi map of parkings (Munchen-12 p.m.) . . . . .	80
A.40 Voronoi map of bookings (Munchen-5 p.m.) . . . . .	80
A.41 Voronoi map of parkings (Munchen-5 p.m.) . . . . .	80
A.42 Voronoi map of bookings (Stuttgart-8 a.m.) . . . . .	81
A.43 Voronoi map of parkings (Stuttgart-8 a.m.) . . . . .	81
A.44 Voronoi map of bookings (Stuttgart-12 p.m.) . . . . .	81
A.45 Voronoi map of parkings (Stuttgart-12 p.m.) . . . . .	81
A.46 Voronoi map of bookings (Stuttgart-5 p.m.) . . . . .	81
A.47 Voronoi map of parkings (Stuttgart-5 p.m.) . . . . .	81
A.48 Voronoi map of booking zones (Frankfurt-8 a.m.) . . . . .	82
A.49 Voronoi map of parking zones (Frankfurt-8 a.m.) . . . . .	82

## List of Figures

---

A.50 Voronoi map of booking zones (Frankfurt-12 p.m.) . . . . .	82
A.51 Voronoi map of parking zones (Frankfurt-12 p.m.) . . . . .	82
A.52 Voronoi map of booking zones (Frankfurt-5 p.m.) . . . . .	82
A.53 Voronoi map of parking zones (Frankfurt-5 p.m.) . . . . .	82
A.54 Voronoi map of booking zones (Munchen-8 a.m.) . . . . .	83
A.55 Voronoi map of parking zones (Munchen-8 a.m.) . . . . .	83
A.56 Voronoi map of booking zones (Munchen-12 p.m.) . . . . .	83
A.57 Voronoi map of parking zones (Munchen-12 p.m.) . . . . .	83
A.58 Voronoi map of booking zones (Munchen-5 p.m.) . . . . .	84
A.59 Voronoi map of parking zones (Munchen-5 p.m.) . . . . .	84
A.60 Voronoi map of booking zones (Stuttgart-8 a.m.) . . . . .	84
A.61 Voronoi map of parking zones (Stuttgart-8 a.m.) . . . . .	84
A.62 Voronoi map of booking zones (Stuttgart-12 p.m.) . . . . .	85
A.63 Voronoi map of parking zones (Stuttgart-12 p.m.) . . . . .	85
A.64 Voronoi map of booking zones (Stuttgart-5 p.m.) . . . . .	85
A.65 Voronoi map of parking zones (Stuttgart-5 p.m.) . . . . .	85
A.66 Voronoi map of booking zones (Hamburg-12 p.m.) . . . . .	86
A.67 Voronoi map of parking zones (Hamburg-12 p.m.) . . . . .	86

# List of Tables

3.1	RESULT_START . . . . .	31
3.2	RESULT_END . . . . .	31
3.3	Evolution of number of vehicles of Frankfurt . . . . .	35
3.4	Evolution of number of zones of Frankfurt . . . . .	37
3.5	Evolution of number of bookings of Frankfurt . . . . .	38
4.1	Cluster definition . . . . .	49
A.1	Development of bookings by years (Munchen) . . . . .	61
A.2	Development of bookings by years (Stuttgart) . . . . .	61
A.3	Development of bookings by years (Hamburg) . . . . .	62
A.4	Average, Median and 90th percentile of booking duration by city . . . . .	62
A.5	Development of vehicles by years (Munchen) . . . . .	62
A.6	Development of vehicles by years (Stuttgart) . . . . .	62
A.7	Development of vehicles by years (Hamburg) . . . . .	63
A.8	Development of stations by years (Munchen) . . . . .	63
A.9	Development of stations by years (Stuttgart) . . . . .	63
A.10	Development of stations by years (Hamburg) . . . . .	63
A.11	Development of bookings by month (Munchen) . . . . .	70
A.12	Development of bookings by month (Stuttgart) . . . . .	71
A.13	Development of bookings by month (Hamburg) . . . . .	72
A.14	Development of bookings by month (Frankfurt) . . . . .	72
A.15	Statistics of the peak hours of Frankfurt-2014 . . . . .	73
A.16	Statistics of the peak hours of Frankfurt-2015 . . . . .	73
A.17	Statistics of the peak hours of Frankfurt-2016 . . . . .	73

A.18 Statistics of the peak hours of Stuttgart-2014 . . . . .	74
A.19 Statistics of the peak hours of Stuttgart-2015 . . . . .	74
A.20 Statistics of the peak hours of Stuttgart-2016 . . . . .	74
A.21 Statistics of the peak hours of Hamburg-2014 . . . . .	75
A.22 Statistics of the peak hours of Hamburg-2015 . . . . .	75
A.23 Statistics of the peak hours of Hamburg-2016 . . . . .	75
A.24 Distance analysis by cluster (Munchen) . . . . .	76
A.25 Distance analysis by cluster (Stuttgart) . . . . .	76
A.26 Distance analysis by cluster (Frankfurt) . . . . .	76
A.27 Distance analysis by cluster (Hamburg) . . . . .	76



# Chapter 1

## Introduction

### 1.1 Overview

The concept of mobility is changing. Nowadays there are different systems to help people move in a sustainable way. The concept of sharing vehicles like cars or bicycles is more and more spreading. The bike sharing is growing up in the last years in many cities in different parts of the world.

As it is mentioned in many surveys, people prefer bikes because they are a healthy and a fast way to move especially across small distances. Moreover, people enjoy riding a bike.

There are two types of bike sharing: the station-based bike sharing and the free-floating bike sharing. The first one is usually public, the user has to do the registration on the website and then the service can be used. But it is based on stations, this means that the users must take and park the bike in a station which must be found through the mobile app.

The free-floating bike sharing is a relatively new system. The most important feature is that docking stations are not needed and the cost of installing and maintaining the racks are avoided. For this reason, the cost of the installation for the companies is less than the station-based system. Thanks to this new system searching a station for parking the bike is not necessary, hence it is easier for the user to leave the bike. In Figures 1.1 and 1.2 is showed an example of a station-based and a free-floating bike sharing system.



Figure 1.1: Example of station-based bike sharing



Figure 1.2: Example of free-floating bike sharing

Each bike is equipped by a GPS Tracker so that the user can find and book the bicycle through the smartphone app. Then he/she has up to 15 minutes to reach it and unlock the padlock. When the trip is finished the bike can be left in any place, inside the enabled area, as long as it is visible for the next customer.

In Italy different cities already have bike sharing, both station based and free-floating. For what concern the first class the majority is public (municipality) while the second class is usually private (big company especially Chinese). Some of the most important are Ofo [1] in Milan, and Mobike [2] that is located in major Italian cities such as Milan, Turin, Florence, Bergamo, etc.

## 1.2 Goal of the thesis

In the last few years not a lot of studies were made on the bike sharing system especially for what concern the free-floating bike sharing. The majority of these works deal with the reallocation of bikes from some places where the demand is low, for example because they are not easy to reach, to the places where the demand of rentals is high. The aim of this master thesis is to analyse the data coming from a free-floating bike sharing system to know the habits of the customers. To this purpose two types of data

research are made:

- *Real-time data*: to access this type of data some APIs are tested in order to have in real time the data of users movements in a certain city;
- *Historical data*: the website of Deutsche Bahn [4] makes available a dataset of a bike sharing system owned by them named “Call A Bike”. That dataset contains the historical data of the users in many German cities.

For what concern the first kind of data the API of Ofo system in the city of Milan is considered. Specifying some parameters such as the coordinates of a point, the data of the available bikes in a certain region can be retrieved. At the same time a crawler is built up in order to process these data.

With this first approach some issues occurred which prevented the continuation of the real-time analysis. In Chapter 2 the problems and the possible solutions found will be explained in detail.

Due to the problems encountered with the real-time data another approach is considered. We decided to analyse the data coming from the dataset of “Call A Bike” system selecting four main German cities: Munchen, Stuttgart, Frankfurt and Hamburg.

For this purpose an initial statistical analysis is made in order to understand which kind of data we are dealing with and finally two different types of analysis have been done.

- *Temporal analysis*: to know when people take a bike, hence which are the hours of the day, the days of the week and the months in which the system is mostly used;
- *Spatial analysis*: to understand which are the frequented zones and the patterns followed by the users.

To summarize, the goal of this work is to try to analyse data coming from different platforms to understand the habits of bike sharing customers, so when and how this kind of systems is utilized.

To reach this goal of the above mentioned methods are examined.



## 1.3 Motivation

In this section is explained the reasons why this study is performed. As mentioned before not many studies are done on bike sharing systems. In literature some surveys can be found to know the bike sharing user habits, in this way the customers can answer to some questions on their manner to use the system.

In this thesis the data, like the position through the coordinates and the time in which a bike is booked, are analysed in order to understand where and when the customers prefer to take and left the bike. Studying this pattern the habits of people can be understood. For example to minimize the disservice of the system during the maintenance of the fleet, the periods of inactivity of the bikes can be known. In this way the maintenance could be done in those periods, influencing the usage pattern of the customers in a minimal way and hence the companies can handle the service in a better way.

For what concern the historical data another motivation of this work is to determine how the system evolve in time knowing the bikes placed at the disposal in the different years. Another analysis could be done to understand if there are some peaks in the system usage during important events. In this way we can know if the service is influenced in some particular periods of the year. Moreover the usage in the various months can be investigated to understand if the seasons affect it. Hence we can know if the users' willingness to ride bike is influenced by the weather. In the end knowing the time intervals of the day and the areas in which the system is more used, a reallocation strategy can be implemented to help companies to handle the service in a better way.

## 1.4 Literature review

The majority of the studies made on free-floating bike sharing take into consideration the reallocation methods. Only few papers deal with the habits of the customers and these are based on surveys. In this section some papers about bike sharing are described differentiated by topics in order to understand the existing works.

### 1.4.1 User habits analysis

Here the papers found in literature that regard the users' habits are mentioned. All of them are based on surveys that take into consideration also socio-demographic aspects. In this thesis we want to study the same topic but on the basis of the only geographic and temporal data. In this way a more precise analysis can be performed. Below some of these papers are explained.

A study performed by the research university Virginia Tech [5] through a survey compared the users' answers of the five dockless services with the answers of one docked system in the District of Columbia. These companies are Ofo, Mobike, Jump, Spin and Lime Bike while the station-based one is Capital Bikeshare.

The aim of this work was to know if the two systems have a different geographic area coverage, the users' demographic profile and in the end thanks to the Geographic Information System (GIS) data, from District Department of Transportation (DDOT), understand where the systems are more used.

A first investigation showed that the most important reason why people prefer bike is because it is the fastest way to move both for the two systems.

The distance travelled by the half of the dockless customers is among 1 mile and 2.9 miles (about 1.6 km and 4.6 km).

For both the systems some historical data from DDOT (October 2017 - January 2018) were considered and it can be noticed that there is a drop in the usage in the cold months like December and January, even if the docked system is in any case the most used.

For what concern the trip pattern in weekdays there are two peaks, one in the morning (9 a.m.) and one in the afternoon (5 p.m.) while in the weekends only one peak has been noticed in the early afternoon as it is shown in the Figures 1.3 and 1.4

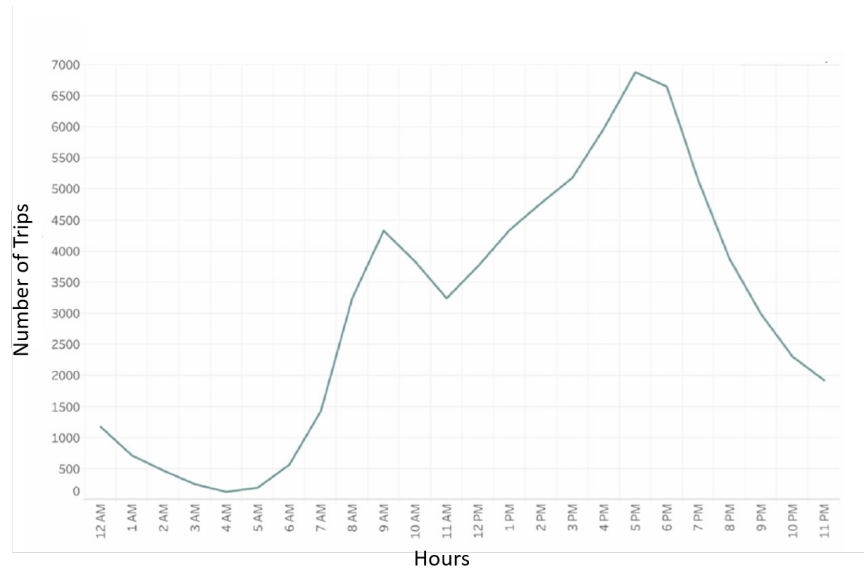


Figure 1.3: Dockless Weekday Rider Peaks (October 2017 - January 2018) (Source: [5])

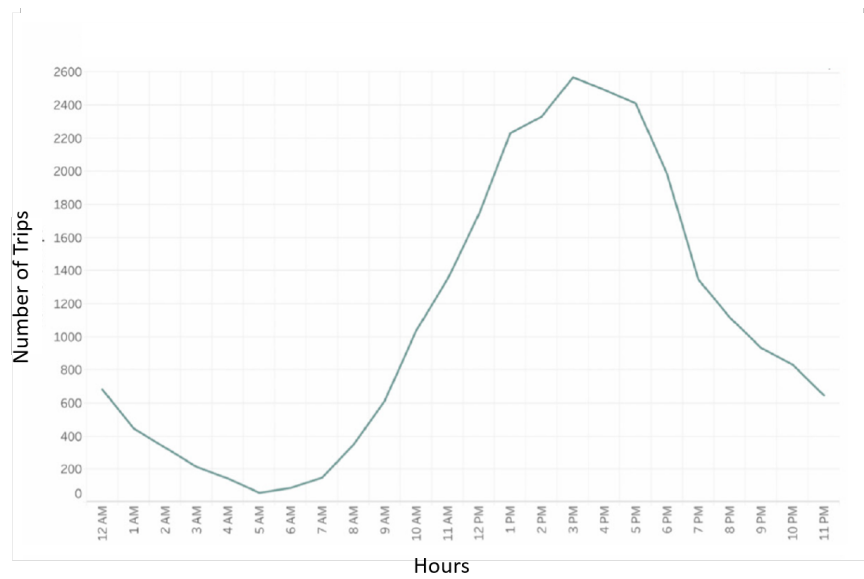


Figure 1.4: Dockless Weekend Rider Peaks (October 2017 - January 2018) (Source: [5])

Finally, thanks to GIS it has been noticed that the most of the rentals are made in the central of the District.

In [6] the users' characteristics were studied comparing both the station-based and the free-floating systems. The city of Hangzhou in China was considered and a survey was developed taking into account some parameters. The authors decided to refer to the purpose of [7] and [8] who thought that to improve the quality of the service the scope of the trip and the travel developments must be known. In addition they refer to [9] for the choice of the most important factors to look at. The results depict that the parameters that mostly influence the station-based system are:

- “*travel purpose*”: the frequency usage is different for different travel purpose;
- “*travel distance*”: the system allows to have the “first-hour free”;
- “*family car ownership*”: in some days in the peak hours the car can not be used in the center of the city;
- “*education background*”: young people go to boarding schools so they can not use the bikes in the weekdays.

For what concern the free-floating system, the results showed that only two parameters affect it:

- “*gender*”: the frequency usage is different between men and women;
- “*monthly cell phone data purchased*”: for booking a bike the phone is needed.

In [10] an interesting study that can be in line with the work done in this thesis is mentioned. The authors want to study the usage pattern of the station-base system in the city of Shenzhen. The data used were taken from an operator-run website every 10 minutes and then they were processed. The most relevant analysis was done for the city center named Luohu. The zones were divided into four clusters and the results were shown for workdays and for weekends. A first temporal analysis depicted that the number of available bikes is low in the ranges of time (07:00-09:00) and (18:00-19:00), this means that the peaks of bookings are in those intervals. For what concern the spatial study in workdays, clusters 1 and 2 represent the stations in the central part of the city in which many offices can be found. These two clusters are labeled as

“*morning destination*” and “*night origin*” this because at 08:00 the bikes are left in that stations, while at 17:00 bicycles are booked again. Cluster 3 and 4 represent the periphery which is a residential zone and in particular cluster 4 is labeled as “*morning origin, night destination*”. In this case an opposite behaviour is shown because in that stations bikes are booked from 08:00 to 17:00. In the end weekends were taken into account. In this case cluster 1 and 2 are not different from the ones on workdays. The behaviour of cluster 4, instead changes during the day, in fact user reserves a bike in the morning and left it in the early afternoon or in the evening.

### 1.4.2 Reallocation problem

This kind of problem is not strictly linked with the goal of this thesis, but it is mentioned because it is one of the most interested topic in a bike sharing system study, in fact the need to find a bike in the right place leads to an increase of the usage of the system. Here some papers that take into account that problem are mentioned.

In [11] the dataset of “Call a Bike” was used and the city of Munich has been studied in 2016.

Firstly, an analysis was made to model the demand of the rentals. The number of bookings have been counted in five different time intervals differentiating between weekdays and weekends, then the city was divided into 40 zones. In the morning (from 6 a.m. to 10 a.m.) the highest number of bookings was in the periphery of the city while in the evening (from 4 p.m. to 8 p.m.) the most of the bookings have been observed in the city center, this means that the fleet is not balanced because the next day the majority of the bikes are not in the right place, and this happens especially from Sunday to Monday.

Secondly, the demand model was built up using three factors that were calculated for each zone  $i$ :

- *Demand factor*  $D_i$ : represents the demand in a certain time interval. If this value is high this means that the demand in that zone is high;
- *O/D factor*  $O_i$ : represents the difference between the inflow and outflow in a zone;

- *Idle time factor  $I_i$* : represents the idle time of a bike; higher idle time corresponds to less demand in that zone.

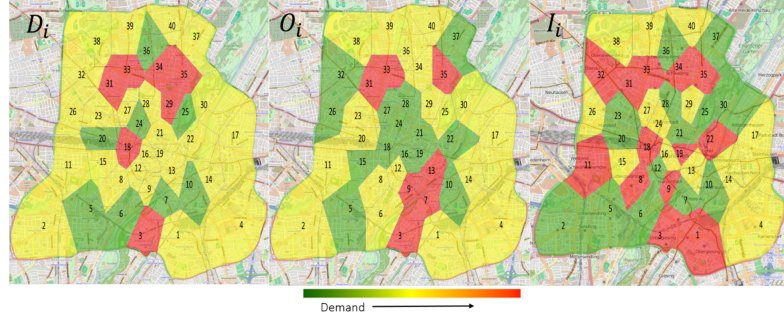


Figure 1.5: Pattern of the demand factor, the O/D factor and the idle time factor (Source: [11])

The strategies of reallocation are two: *User-based* and *Operator-based*.

The first one can be used when the imbalance of the fleet is low and this method can be encouraged through some discounts to the user. The advantage is that it is cheap because the presence of the staff is not necessary but it is not very efficient because it is not easy to know the willingness of the user. The second strategy is used when the need of redistribution is high. Operators can carry bikes on some vehicles but this is very costly even if the reallocation is efficient.

In [12] the dataset of “Share-A-Bull BSS” was used to understand the demand of rentals and the imbalance of the system in time influenced by some variables like some external factors which may not seem directly related to the system usage. The study on how these factors can affect the service is done in Tampa campus (University of Florida), hence the analysis depends mainly on the students’ habits.

The aim of this study was to understand the variation of the demand and of the imbalance, which is the difference between parkings and bookings, influenced by some variables like weather, holidays, etc. Both parkings and bookings were analysed daily and hourly, while imbalance was studied hourly.

For the analysis two approaches were considered. Due to the fact that the variances of daily and hourly rentals and parkings are larger than the means the “Negative binomial regression” model was used. This approach, as mentioned by the authors, is the one commonly used in the literature for this type of variables because they were interested

in the generation of non-zero count variables. For the analysis of imbalance, instead, the “Linear regression” model was adopted because in this case the authors were considered also negative values.

For the results only bookings were showed in order to avoid repetitions. It can be noticed that the Fall season has a positive effect both for daily and hourly bookings while Spring and holidays have a negative effect as well as the time range 23:00-6:00. The sky clouded and the humidity have in general a negative impact too. Moreover, in September on Tuesday bookings start from 8:00 and not from 7:00 and on Sunday from 22:00 to 23:00 the bikes were used a lot maybe for leisure activities of the students.

Hence the operator-based approach can be used from 23:00 to 6:00 except on Tuesday in September that can be extended to 8:00, while the hybrid approach is recommended in the morning and in the early afternoon from Monday to Friday according to the schedule of the student classes.

In [13] a new version of “Chemical Reaction Optimization” (CRO) called enhanced CRO was proposed to solve the reallocation problem. The objective function of this algorithm was to minimize the total unsatisfied demand and the total operational time and to maximize the probability of getting a bike.

For the analysis the graph theory was considered representing the bike sharing system by a complete directed graph  $G_0 = (V_0, A_0)$ , where  $V_0$  are the nodes representing the zones in which the bikes can be found and  $A_0$  are the edges that constitute all the possible links between the nodes. To simplify the system, nodes were divided into two categories: hardly accessed, which are difficult to reach or found, and easily accessed. The goal is to adjust the quantity of bikes to load or unload in a node with the help of some vehicles that can carry a certain number of bikes.

Considering the demand at hardly nodes equal to 0, a mathematical model was build up to construct the initial solution and based on this the number of loading/unloading quantity was adjust.

Three bike sharing systems were used for the analysis, the “Citybike” in Vienna, the “Share-A-Bull” in Tampa campus of Florida University and the “Citi Bike” in New York.

For all of them different scenarios were considered varying the number of vehicles used and the repositioning time. The results showed that the enhanced CRO always reaches a feasible solution especially when the repositioning time and the number of vehicles become high.

### 1.4.3 Usage prediction problem

Here a solution that takes into account the Machine Learning to solve a problem of predicting the rentals in a range of time. The work described in this thesis can be considered a usefull base for this type of approach. This is the reason why that paper is mentioned.

In [14] two Machine Learning models were tested to predict the number of rentals in one hour. In this way the companies that handle the system can manage it in a better manner.

The dataset used comes from the University of Irvine in California and represents the data of two years (2011-2012) of “Capital Bikeshare” system in Washington D.C.

By an initial analysis it can be noticed that the number of rentals per hour changes with temperature and humidity, for example if the temperature is too high or too low that number decreases. Moreover, an analysis month by month showed that the number of bookings increases in Summer.

After that two Regression algorithms were tested for the prediction: *Maximum Likelihood Estimate* and *Maximum A Posteriori*. The goal is to find the best weights  $\mathbf{w}$  such that given the features  $\mathbf{X}$ , the prediction  $\mathbf{Y}_{\text{predict}} = \mathbf{X} * \mathbf{w}$ , is much close as possible to the real result  $\mathbf{Y}_{\text{test}}$ .

The first model examined was MLE, as mentioned before the aim is to find the weights  $\mathbf{w}$  that maximize the likelihood function  $P(D|\mathbf{w})$ . That parameters were found from the training-set using Ordinary Least Squares (OLS) method and then tested on test-set in order to compare the  $y_{MLE}$  and the  $y_{Test}$ .

The second model MAP is based on Bayes’ Theorem and unlike the MLE this algorithm works on a posterior distribution. But both the mean and the variance of the weights were not know, hence different combination of these parameters were tried on



training-set to find the best one to be used on the test-set.

The results depicted that the two algorithm are really similar to each other. Moreover, there is a relationship among the count and the predictions, but both the models can predict the count in a better way when the number of the usage is low.

## 1.5 Thesis organization

In this section the topics of each chapter are briefly explained.

Chapter 2 describes the researches for finding the real-time data of the free-floating bike sharing system with the attempted study done over the collected data and the problems found in this approach. In addition the historical data used are introduced.

Chapter 3 explains in detail the historical data utilized, the methodologies and the tools used for the implementation of the work. A preliminary statistical analysis is also shown in order to understand which kind of data are considered and how they evolve in the different years.

Chapter 4 depicts the results obtained from the analysis of the historical dataset. This analysis is divided into two scenarios: in the first one all the temporal data are considered while in the second the spatial analysis is made, in order to understand when and where people prefer to move with bikes.

Chapter 5 is the conclusive chapter of the thesis in which a sum up of the study is described and future works are proposed.

## Chapter 2

### Datasets

This chapter deals with the aspect related to data. The development of this thesis has been possible thanks to the availability of *open data*. These data can be accessed in a free manner and any limitation is required for their use.

For what concern the free-floating bike sharing getting available data has been difficult since the companies are private and from their websites there were not data put at the disposal, for this reason a research on APIs that could reach data has been done.

An **Application Programming Interface** (API) defines a set of methods that allow the communication between services. Moreover, it is often used by programmers to access some data. Since the aim of this thesis is not to develop a new API to access data from the bike sharing companies, it was decided to search an appropriate API from the GitHub platform.

The data needed for the analysis of this work are related to the usage of the bike sharing system and they are explained below. All these data used for the development of this study, concern only the information related to bikes.

The useful data regard three main information:

- *ID*: represents a unique code that identifies a single bike, in this way the number of vehicles used or parked can be counted;
- *time*: in which a bicycle is booked or parked in order to understand the mean time of the usage or the hours of the day in which the system is mostly used;

- *coordinates*: represent the point in which a bike can be found, in terms of latitude and longitude. They are useful to know which are the zones with the majority of bookings and the general pattern of the mobility.

This chapter is divided into two sections, the first one regards the real-time data, this kind of information deals with the real-time usage of the system, hence it is possible to have information on how the customers use the service immediately. With this first approach some problems, that are explained in detail later, occurred. For this reason another approach has been followed with a different kind of information: historical data. In this way old data have been analysed in order to reach the goal of this thesis.

## 2.1 Dataset acquisition from real-time flows

As mentioned previously, the first part of the work is focused on the research of real-time data. The goal is to collect some data from a system to be then processed.

Since any kind of official data are available a GitHub page [15] is found in which a list of APIs of some bike sharing platforms is described. Each API is related to one bike sharing system.

The majority of the APIs taken from the website are tested in order to understand which parameters are taken into account and which kind of data can be obtained.

All of them require two important parameters: the coordinates of a point in terms of latitude and longitude. In this way the data got from this request are all the bicycles available in a certain radius of the given point.

The focus is placed on Italy, so an Italian city is taken into consideration for the collection of data of the “Ofo” system. “Ofo” is a free-floating bike sharing company born in 2014 in Beijing, China. This system has grown up in a short time and it is developed in many countries. “Ofo” came in Italy in 2017 in two cities: Milan and Varese. For the purpose of this study Milan is chosen because the large number of users.

In order to understand in a better way how the data are collected, all the steps done are explained below.

### 2.1.1 Real-time data acquisition

The API found performs a request to a web server, hence the techniques of the HTTP protocol are used, in particular POST requests are needed. A POST request is an HTTP method to retrieve information from a web server specifying in the body the right parameters.

First of all to access to Ofo platform the registration is mandatory, so the app has been downloaded and installed in an android smartphone. Three POST requests are required in which the first two are necessary to obtain the *token* that is needed in the last request in order to get the bikes. For all the requests the **header key** is set to *Content-Type* and the **header value** to *application/x-www-form-urlencoded*.

The first POST request is made to *https://one.ofo.com/verifyCode\_v2* to receive the *OTP code* through an SMS for the second request. In the Figure 2.1 the parameters to use are shown. The ‘tel’ parameter represents the phone number in which the “Ofo” app is installed and in which the code will be retrieved, ‘ccc’ is the country code from which the request is made, in the end ‘lat’ and ‘lng’ are the coordinates of the point for which the bikes are shown.

Key	Value
tel	333123456
type	1
ccc	39
lat	45.516338
lng	9.10416

Figure 2.1: Parameters for the first POST request

Once the *OTP code* is received the second POST request can be done to *https://one.ofo.com/api/login\_v2* in order to obtaine the *token*. The Figure 2.2 depicts the parameters for this request that are the same of before with in addition the *OTP code*.

## 2.1. Dataset acquisition from real-time flows

---

Key	Value
tel	333123456
code	1225
ccc	39
lat	45.516338
lng	9.104160

Figure 2.2: Parameters for the second POST request

Finally, the last POST request to <https://one.ofo.com/nearbyofocar> is the most important because from this a list of bicycles can be obtained. The coordinates represent the chosen point, that is the center of a circle in which the bikes are parked. In the following Figure 2.3 the parameters used for the third request are listed. The ‘source’ constitutes the smartphone type (1 for android or 2 for iOS) while ‘token’ is the parameter received with the second POST.

Key	Value
lat	45.516338
lng	9.104160
source	1
token	2db25860-f09b-11e8-b128-efec9f286c73

Figure 2.3: Parameters for the third POST request

An example of response to this request is shown in Figure 2.4 with the zoom on the most important fields in order to understand which kind of information are displayed from the request made.

## 2.1. Dataset acquisition from real-time flows

```
{
  "errorCode": 200,
  "msg": "附近车辆位置",
  "values": {
    "cars": [
      {
        "carNo": "eDg3g1",
        "bomNum": "5CA",
        "userIdLast": "1",
        "lng": 9.1047016030321,
        "lat": 45.522189527284
      },
      {
        "carNo": "eDg3g1",
        "bomNum": "5CA",
        "userIdLast": "1",
        "lng": 9.1047016030321,
        "lat": 45.522189527284
      }
    ],
    "expPrice": {
      "price": "0.60",
      "actualPrice": "0.60",
      "orderTime": 1200,
      "currency": "€",
      "type": 1
    },
    "icon": "http://ofo-testmeixi-image.oss-us-west-1.aliyuncs.com//report/6fc78646df3a375416f9c1884728fa50.png",
    "bikeIcon": [
      {
        "bomNum": "0",
        "icon": "http://ofo-testmeixi-image.oss-us-west-1.aliyuncs.com//report/6fc78646df3a375416f9c1884728fa50.png",
        "animationUrl": ""
      }
    ]
  }
}
```

Figure 2.4: Response to the third POST request

What can be noticed is that the most interesting information are in the first block. The field “carNo” represents the bicycle ID while “lng” and “lat” indicate the precise position of the bike. Drawing these coordinates in a fusion table it has been noticed that the farther distance from the given point is about 500 meters. So a circle with a radius equal to 500 meter can be considered (Figure 2.5).

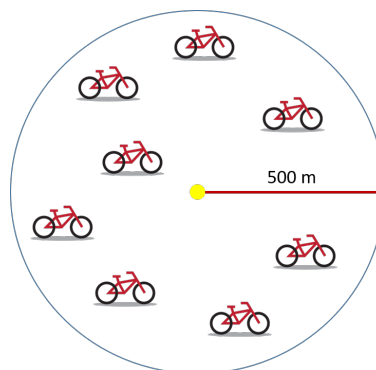


Figure 2.5: Example of response

## 2.1. Dataset acquisition from real-time flows

To collect the bikes for the entire city of Milan, more parallel requests must be done in different points. To select the area of the city in which the requests must be done in order to retrieve the majority of the available bikes, the map of the “Ofo” system is considered to understand where the service is active.

Looking at this map depicted in the Figure 2.6 it can be noticed that the green part represents the zone in which the system is active without any fees while in the area outside the green zone “Ofo” is active with a payment of an extra fee.



Figure 2.6: “Ofo” coverage (Source: [16])

For the analysis in this study, to simplify the model, the city is inserted into a square and looking at the map (Figure 2.7) four points with the respective coordinates which represent the angles of the square are taken into account. In this way the total of the green area is covered and a large part of the outside zone is considered, this because in the green part the system is mostly used by users and the majority of the bicycles are present. In fact, with a preliminary analysis in the border of the city only a small number of bikes is visible that can be considered negligible. Moreover, taking into account this square the best trade-off between the amount of coverage and the number of points in which the requests are done has been found; in order to cover a

## 2.1. Dataset acquisition from real-time flows

representative area of the system with less computational time.

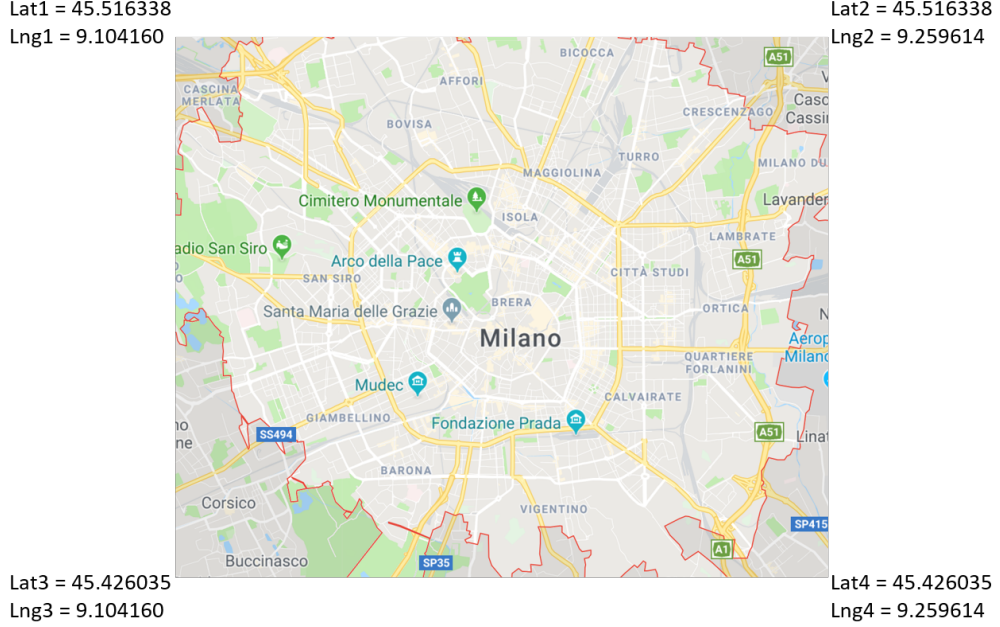


Figure 2.7: Perimeter of the city of Milan

A grid is created taking into account a set of latitudes from 45.516338 to 45.426035 and a set of longitudes from 9.104160 to 9.259614. To know the number of steps which are the number of points in which the request must be done, the area must be divided in equal parts considering a conversion factor of 0.0092 for latitude and 0.013 for longitude. those factors are used when we want to move from a point of the globe to another on a distance equal to 1 km. In this way the number of steps can be found as follow:

$$\frac{Lat3 - Lat1}{0.0092} = \frac{45.426035 - 45.516338}{0.0092} = 10steps \quad (2.1)$$

$$\frac{Lng2 - Lng1}{0.013} = \frac{9.259614 - 9.104160}{0.013} = 12steps \quad (2.2)$$



## 2.1. Dataset acquisition from real-time flows

In this way a  $1\text{km} * 1\text{km}$  grid is created as it is shown in Figure 2.8. But in this grid some areas are not covered, then other points are considered in order to have the total coverage.

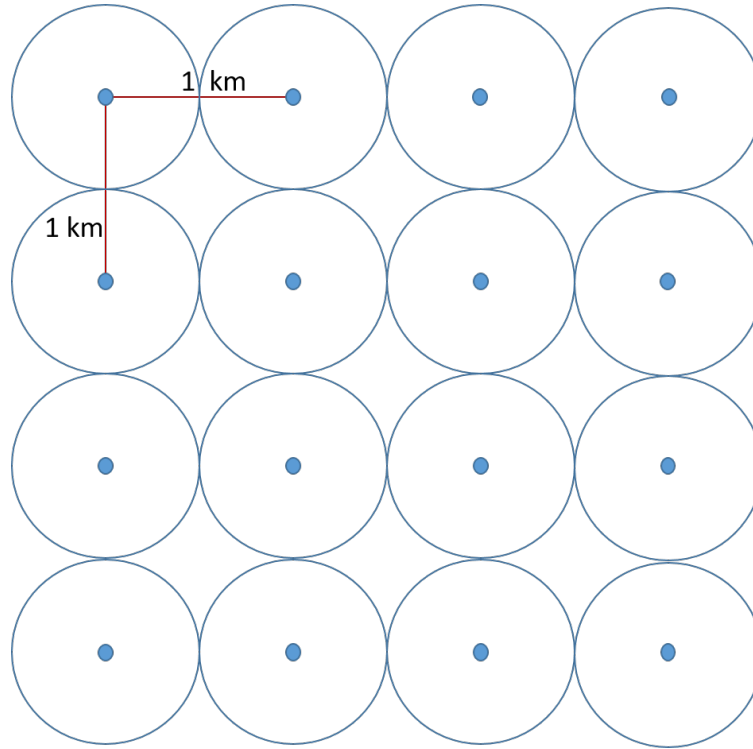


Figure 2.8: Grid of the city of Milan

Another grid is built on the before mentioned, but this time the starting point is in the middle of the first four circles, and it is calculated summing or subtracting the conversion factor, that corresponds to 500 meters to the coordinates of the first point.

$$Lat1 - 0.0046 = 45,511738 \quad (2.3)$$

$$Lng1 + 0.0065 = 9,11066 \quad (2.4)$$

In this case the number of steps to consider are one less with respect to the previous ones, so 9 steps for the latitude and 11 steps for longitude. In the Figure 2.9 the final grid used is depicted, but it is visible that there are some overlapped zones, the bikes in that areas are considered just once.

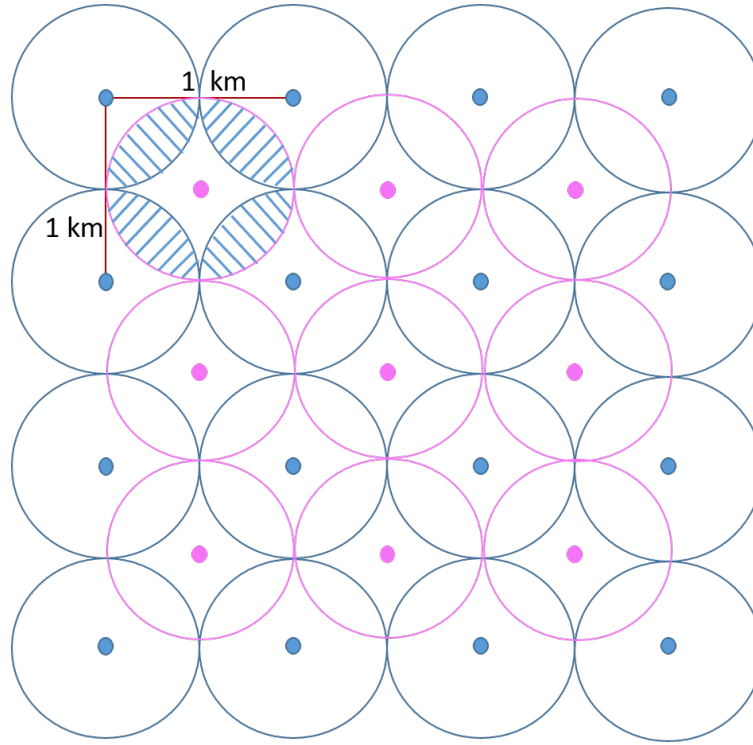


Figure 2.9: Final grid of the city of Milan

The final idea is to make parallel requests to these points in order to obtain all the data in a real-time manner and in the end process these data to understand how many bikes are booked and when the users start the rental.

The initial part of the study is focused on the parallel requests in each centroid of the grid. For this purpose a Python script is written in which starting from the initial coordinates of the first two centroids (the first blue and the first pink in Figure 2.9), all the points have been considered from left to right and with the help of the *wget* function the parallel requests are made in each point.

As can be seen from Figure 2.4 in the response the time is not present, for this reason these responses are saved in a text file and in the name of this file a timestamp, which includes the time and the date, is inserted. Every time that the requests are done in the entire city a new file is created, in this way in each file we store the total bikes present in the whole city.

## 2.1. Dataset acquisition from real-time flows

To work in a better way the data are transformed into a *Json* format in order to access the useful information. In Figure 2.10 the process of parallel requests with the *wget* function and the timestamp in the file name are shown.

```
while True:
    now = datetime.now()
    now_string = now.strftime("%d_%m_%Y_%H_%M_%S")
    now_res = now_string+".txt"

    for i in range(len(lat1)):
        for j in range(len(lng1)):
            cmd = "wget http://one.ofo.com/nearbyofocar --post-data='{\"content-type=application/x-www-form-urlencoded&lat="+str(lat1[i])+ "&lng="+str(lng1[j])+ "&source=1&token=a74c68b0-bce7-11e8-a5d7-9576d81197d0\"}' -O - >> "+now_res
```

Figure 2.10: Parallel requests script

In the Figure 2.11 an example of response saved in the file is depicted. Each block represent the response in a given point, so since the city is composed by two grids, the first one with a number of steps equal to  $10 * 12 = 120$  and the second one have  $9 * 11 = 99$  steps, in total the amount of centroids are 219. Hence, in a file 219 blocks are present.

```
1 {"errorCode":200,"msg":"附近车辆位置","values":{"cars":[{"carNo":"eDg3g1","bomNum":"5CA","userIdLast":"1","lng":9.1045955114173,"lat":45.522005463758}],{"price":"0.60","actualPrice":"0.60","orderTime":1200,"currency":"€","type":1,"icon":"http://ofo-testmeixi-image.oss-us-west-1.aliyuncs.com/report/6fc78646df3a375416f9c1884728fa50.png","bikeIcon":{"bomNum":"0","icon":"http://ofo-testmeixi-image.oss-us-west-1.aliyuncs.com/report/6fc78646df3a375416f9c1884728fa50.png","animationUrl":""}}}}
2 {"errorCode":200,"msg":"附近车辆位置","values":{"cars":[{"carNo":"m9gZ2x","bomNum":"30A","userIdLast":"1","lng":9.1193231727987,"lat":45.517117555824}, {"carNo":"OMgRd5","bomNum":"3DA","userIdLast":"1","lng":9.1198025121032,"lat":45.517378790163}, {"carNo":"eDdNIN","bomNum":"0","userIdLast":"1","lng":9.1160017472432,"lat":45.518741875714}],{"price":"0.60","actualPrice":"0.60","orderTime":1200,"currency":"€","type":1,"icon":"http://ofo-testmeixi-image.oss-us-west-1.aliyuncs.com/report/6fc78646df3a375416f9c1884728fa50.png","bikeIcon":{"bomNum":"0","icon":"http://ofo-testmeixi-image.oss-us-west-1.aliyuncs.com/report/6fc78646df3a375416f9c1884728fa50.png","animationUrl":""}}}}
3 {"errorCode":200,"msg":"附近车辆位置","values":{"cars":[{"carNo":"6lO0ZY","bomNum":"5CA","userIdLast":"1","lng":9.1303632239936,"lat":45.519872414036}, {"carNo":"KrV1nR","bomNum":"3DA","userIdLast":"1","lng":9.134198268715,"lat":45.513735822638}, {"carNo":"38zLlg","bomNum":"0","userIdLast":"1","lng":9.1348806143152,"lat":45.519477052573}],{"price":"0.60","actualPrice":"0.60","orderTime":1200,"currency":"€","type":1,"icon":"http://ofo-testmeixi-image.oss-us-west-1.aliyuncs.com/report/6fc78646df3a375416f9c1884728fa50.png","bikeIcon":{"bomNum":"0","icon":"http://ofo-testmeixi-image.oss-us-west-1.aliyuncs.com/report/6fc78646df3a375416f9c1884728fa50.png","animationUrl":""}}}}
```

Figure 2.11: Response file example

In each block the bikes parked in the radius of the centroids are listed with the identifier, the coordinates of their position and some other additional information which are not useful for our analysis.

Simultaneously, a crawler is built up to process the data saved in the files. The goal is to compare every block in all the files. If a bike that is visible in a block of a certain file is not present in the next one, this means that a booking started at the time of the corresponding file name.

A crawler is a software that analyses automatically the content of a database or the content of a network.

### 2.1.2 Real-time data analysis

For the analysis four collections of data are taken into account:

- *activeBookings*: represents the bikes that are currently booked and not available;
- *activeParkings*: represents the bikes that are currently parked and available;
- *permanentBookings*: represents the history of the bookings so the bookings already recorded;
- *permanentParkings*: represents the history of the parkings so the parkings already recorded.

As mentioned before a crawler is built up in Python language in order to process the data and insert the data in the right collection.

The data should be stored in a database but initially, to test the crawler a local implementation is done, in fact the four collections are represented by dictionaries. A dictionary is a Python's structure which consists in a set of key-value pairs inside curly brackets. Unlike the lists, dictionaries are not ordered and each element can be accessed through the key.

At the beginning the previous mentioned duplicate bikes in the intersected zones are removed with a control on the bike ID and on the coordinates. If those values are repeated this means that the two bikes are the same, so one of them must be cancelled. Then the dictionaries are analysed taking into account the bicycles ID. If a new ID appears this means that a bike is parked so it must be stored in *activeParkings* dictionary, dropped from the *activeBookings* and at the same time it must be inserted in *permanentBookings* in order to update the history of bookings. On the contrary, if the element is not anymore in the response this means that a bike is booked so the ID must be added in *activeBookings*, removed from *activeParkings* and the history of parkings must be adjusted putting the ID in *permanentParkings*.

## 2.1. Dataset acquisition from real-time flows

In Figure 2.12 an example of the result is shown to better know how the crawler works.

```
*****
sto aggiungendo il parking: Y4AOWa
sto aggiungendo il parking: m9QnRD
Active Parkings: 2
Active Bookings: 0
Permanent Parkings: 0
Permanent Bookings: 0
*****
sto aggiungendo il parking: NmLLGK
sto aggiungendo un booking: Y4AOWa
sto rimuovendo un parking: Y4AOWa
Active Parkings: 2
Active Bookings: 1
Permanent Parkings: 1
Permanent Bookings: 0
*****
sto aggiungendo il parking: Y4AOWa
sto rimuovendo un booking: Y4AOWa
sto aggiungendo il parking: beWAOK
Active Parkings: 4
Active Bookings: 0
Permanent Parkings: 1
Permanent Bookings: 1
```

Figure 2.12: Example of crawler result

From this example it can be noticed that initially two IDs “Y4AOWa” and “m9QnRD” are present in the API response consequently two bikes are parked and hence the *activeParkings* collection is set to 2.

Then in the second block the new ID “NmLLGK” appears and added to *activeParkings* while the previous “Y4AOWa” disappears, this means that the bike is booked. Hence, the ID “Y4AOWa” is dropped from *activeParkings*, added to *activeBookings* and the *permanentParkings* collection is updated since the parking is just finished.

In the end in the third block a new bike with ID “beWAOK” is parked and added to *activeParkings*. Moreover, the same bike of previous block is parked, so the ID “Y4AOWa” is in the API response again with different coordinates, hence the collections must be upgraded according to the result.

### 2.1.3 Conclusions on real-time data

For the completion of this analysis a large amount of data are needed (at least one month) in order to achieve a good result, but some problems with the collection of the data have been occurred. In fact, the *token* required for the POST request to retrieve the bikes has a deadline of half an hour approximatively. Hence, to obtain some data another method was developed, the *token* was changed manually each time the deadline occurred. But also in this case any results can be achieved due to a lack in the collected data, in fact some points in which the request must be done were lost.

At this point a step back has been taken and another API has been tested, the one related to another free-floating bike sharing system present in Italy: Mobike, but even in this case the same issues have been found.

In the end as last option the most important companies of free-floating bike sharing systems in Italy have been contacted through e-mail in order to have data in a continuous manner for the real-time analysis. Since no response has been received another approach was conducted considering historical data made available by other companies in order to achieve the goal of this work which consists in analysing data to understand the habits of the customers.

## 2.2 Open historical dataset usage

This section deals with the explanation of historical data. As previously mentioned due to the issues found with the acquisition of real-time data this kind of data are taken into account and processed to understand how the customers use a bike sharing system. The historical data found are open data from past years already saved in a database. From the researches made a dataset put at the disposal for a hackaton by Deutsche Bahn was found and take into consideration.

Deutsche Bahn [3] is one of the most important railway company in the world. It provides both the services of carsharing, called “Flinkster” with a fleet of about 4000 vehicles in over 300 cities and a free-floating bikesharing called “Call A Bike”, with 13000 bikes in about 50 cities.

The “Call A Bike” system is active since 2000. The bicycles are equipped with **Global Positioning System** (GPS) device for the localization. The user can rental a bike in different ways, the most common is through the application installed in the smartphone.

In the app a map with all available bikes is visualized and from this a bike can be selected and reserved. The other methods to book the vehicle are through phone, calling the number visualized in the display and putting the given code in the screen or with a card at the terminal. Bicycles can be found in places easy to reach like a cross road in the town. Once the user locates it the lock on the wheel is unloked and the rental begins.

### 2.2.1 Dataset overview

The Deutsche Bahn website [4] provides different datasets in a CSV form with the data of “Call A Bike” from the past years (in particular 2014, 2015, 2016), for example booking data, rental zones data, vehicles information or data on the price.

As said previously these data were put at the disposal by the company for different purposes such us Hackathons. We decide to use them to study the evolution of the system and how customers use it.

For this reason all of these tables are analysed to understand which data could be useful for the development of this work. Two tables are selected from four available in the website, the booking and the rental zone tables which contain the most significant fields.

The Figures 2.13 and 2.14 show the chosen tables with the different fields and the primary keys.

## 2.2. Open historical dataset usage


PK	Attribut
	BOOKING_HAL_ID
	CATEGORY_HAL_ID
	VEHICLE_HAL_ID
	CUSTOMER_HAL_ID
	DATE_BOOKING
	DATE_FROM
	DATE_UNTIL
	COMPUTE_EXTRA_BOOKING_FEE
	TRAVERSE_USE
	DISTANCE
	START_RENTAL_ZONE
	START_RENTAL_ZONE_HAL_ID
	END_RENTAL_ZONE
	END_RENTAL_ZONE_HAL_ID
	RENTAL_ZONE_HAL_SRC
	CITY_RENTAL_ZONE
	TECHNICAL_USER_NAME

Figure 2.13: Bookings data table  
(Source: [4])



PK	Attribut
	RENTAL_ZONE_HAL_ID
	RENTAL_ZONE_HAL_SRC
	NAME
	CODE
	TYPE
	CITY
	COUNTRY
	LATITUDE
	LONGITUDE
	POI_AIRPORT_X
	POI_LONG_DISTANCE_TRAINS_X
	POI_SUBURBAN_TRAINS_X
	POI_UNDERGROUND_X
	ACTIVE_X
	COMPANY
	COMPANY_GROUP

Figure 2.14: Rental zone data table  
(Source: [4])

To have a lighter tables to work with only the most important attributes are selected. In the booking table the following attributes are considered:

- *BOOKING\_HAL\_ID*: This is the primary key and represents the identifier of the booking;
- *VEHICLE\_HAL\_ID*: Represents the identifier of the bike;
- *DATE\_BOOKING*: Date and time in which the booking occurred that coincides with DATE FROM;



- *DATE\_FROM*: Date and time in which the booking starts;
- *DATE\_UNTIL*: Date and time in which the booking ends;
- *START\_RENTAL\_ZONE\_HAL\_ID*: Identifier of the zone in which the rental starts;
- *END\_RENTAL\_ZONE\_HAL\_ID*: Identifier of the zone in which the rental ends.

While in the rental zone table these fields are included:

- *RENTAL\_ZONE\_HAL\_ID*: This is a primary key and represents the identifier of the zone;
- *CITY*: This is the list of the cities;
- *LATITUDE*: Latitude of the zones in which a bike can be booked or left;
- *LONGITUDE*: Longitude of the zones in which a bike can be booked or left.

All of these fields are useful for the analysis, in fact *BOOKING\_HAL\_ID* represents the primary key, this means that each booking has a unique number that identifies it, thanks to this ID the number of bookings made by users can be counted.

With *VEHICLE\_HAL\_ID* all bikes can be identified and counted in order to know if the system grew up from 2014 to 2016.

*DATE\_FROM* and *DATE\_UNTIL* are also important because they represent the timestamp, useful to understand when the rental has started and finished.

In the end the fields *LATITUDE* and *LONGITUDE* are needed to know the position of the zone in which the bicycle can be booked or parked.

# Chapter 3

## Methodology

In this chapter the initial phases of the work are explained, including the used tools, the adjustment of the data found and an initial analysis to understand which kind of data will be used.

### 3.1 Used tools

#### 3.1.1 Python

Python [17] is a widely popular, high-level, object oriented programming language. High-level means that it is portable: this implies that a script written in Python can be run in any kind of machine without modifications. Moreover, it is easy to understand because the syntax is very simple and readable. That is why the usage of Python has grown up in the last few years.

It is also really efficient thanks to the large amount of available libraries that can be used to deal with some different kinds of tasks like graphical interface, database managing, calculations.

For the purpose of this work, since the amount of data is not huge the use of a database is not needed so the data are managed with a Python library useful for this type of data manipulation. The most important library used for the development of this work is Pandas which is explained below.

### 3.1.1.1 Pandas

Pandas [18] is a useful Python library for managing data like CSV, SQL database, etc. After loading the file, a Dataframe (a Python object) can be created with pandas with rows and columns which are easy to access, just like a table.

This makes simpler to work with this kind of data rather than a list or a dictionary because dataframe are more flexible and data can be also displayed with Excel that is human readable.

## 3.2 Preparation of the data

As mentioned in the previous chapter the data used coming from the Deutsche Bahn dataset. In this section how these data are prepared and handled for the future analysis is explained.

The dataset contains data from 01-01-2014 to 15-05-2017, for the analysis the three years 2014, 2015 and 2016 are considered to have a complete overview on how evolve the system during the entire years.

Starting from the two tables shown in Figures 2.13 and 2.14 which contain the booking and the zone information, some operations are made on them in order to have cleaning data to work with.

Since in the dataset some empty cells are present an operation on the rows is done so after eliminating the unnecessary columns, a filter on the rows is added to delete those rows in which the cells, corresponding to attributes "START\_RENTAL\_ZONE\_HAL\_ID", "END\_RENTAL\_ZONE\_HAL\_ID" for the first table and "LATITUDE", "LONGITUDE" for the second one, are empty.

After that, four main German cities to be analysed are selected: **Munchen, Stuttgart, Frankfurt** and **Hamburg**.

To work with all the data together a join operation is made between the two tables.

For this purpose the booking table is divided into two tables: BOOKING\_START and BOOKING\_END. In the first one the column END\_RENTAL\_ZONE\_HAL\_ID is eliminated while in the second one the attribute START\_RENTAL\_ZONE\_HAL\_ID is dropped. The rental zone table is instead duplicated and the column RENTAL\_ZONE\_HAL\_ID has been appropriately renamed with the attributes START\_RENTAL\_ZONE\_HAL\_ID

### 3.2. Preparation of the data

---

and `END_RENTAL_ZONE_HAL_ID` in order to have the same attribute name with the previous tables.

As second step a *join* is performed between `ZONE_START` and `BOOKING_START` on the key `START_RENTAL_ZONE_HAL_ID` and between `ZONE_END` and `BOOKING_END` on the key `END_RENTAL_ZONE_HAL_ID`.

At this point two tables has been created called `RESULT_START` and `RESULT_END` in which the fields `LATITUDE` and `LONGITUDE` are charged in `LATITUDE_START-LONGITUDE_START` and `LATITUDE_END-LONGITUDE_END`. An example of the two result tables is depicted below.

Table 3.1: `RESULT_START`

<b>BOOKING_HAL_ID</b>	<b>LATITUDE_START</b>	<b>LONGITUDE_START</b>
22685553	48,16194444	11,58611111
34503871	48,16027778	11,57527778

Table 3.2: `RESULT_END`

<b>BOOKING_HAL_ID</b>	<b>LATITUDE_END</b>	<b>LONGITUDE_END</b>
22685553	48,16239178	11,58755563
34503871	48,15055556	11,59527778

In the end an *inner join* is performed between these two result tables on the key `BOOKING_HAL_ID` which is the primary key, in order to obtain a big booking final table and from this four CSV are created, one for each city. Each CSV is made by the following fields:

- *`START_RENTAL_ZONE_HAL_ID`*
- *`LONGITUDE_START`*
- *`LATITUDE_START`*
- *`VEHICLE_HAL_ID`*
- *`BOOKING_HAL_ID`*
- *`DATE_BOOKING`*

- *DATE\_FROM*
- *DATE\_UNTIL*
- *END\_RENTAL\_ZONE\_HAL\_ID*
- *LONGITUDE\_END*
- *LATITUDE\_END*

Moreover, other two columns are created. The first one is called *TIME\_DIFFERENCE* and it has been built doing the subtraction between the two columns *DATE\_UNTIL* and *DATE\_FROM* which contains the duration of each booking. In this way how many minutes a trip lasts can be known.

The second column is called *DISTANCE* and contains the geographic distance in meters from a starting point (*LATITUDE\_START*-*LONGITUDE\_START*) to an end point (*LATITUDE\_END*-*LONGITUDE\_END*). The values in this column are calculated using the Vincenty's formulae which allows to calculate the distance between two points on the surface of the Earth. In this way we can understand how long is a trip.

### 3.3 Exploration of the dataset

In this section an exploration of the dataset is done with some statics in order to understand which kind of data can be useful for the work. Even if "Call A Bike" is a free-floating bike sharing system the vehicles must be taken and left in certain zones. That areas are defined such that the bicycles are easy to found for example intersections and central zones.

Before analysing some statistics, all the points from the rental zone table are plotted for each city, in order to know how scattered they are. These points represent those places in which a bike can be booked or parked. The points are then put on the map of each city to understand which are the areas involved.

### 3.3. Exploration of the dataset

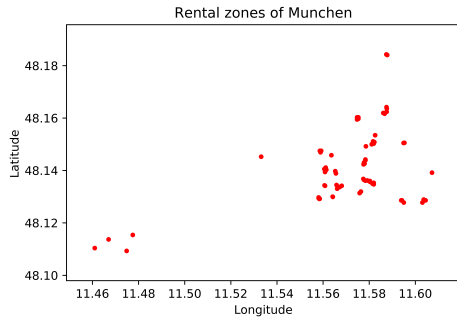


Figure 3.1: Scattered points of Munich

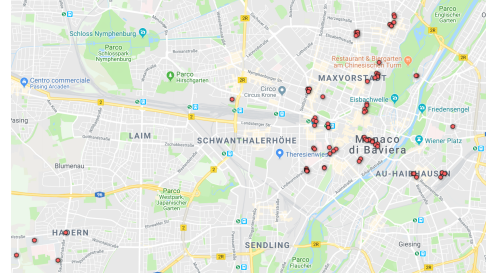


Figure 3.2: Map of Munchen with the rental zones

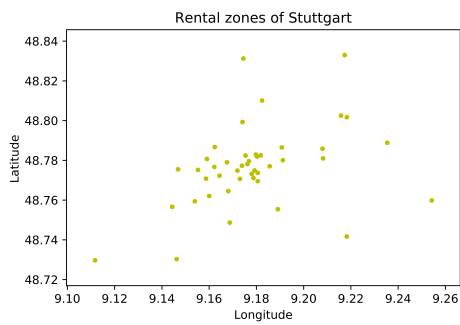


Figure 3.3: Scattered points of Stuttgart

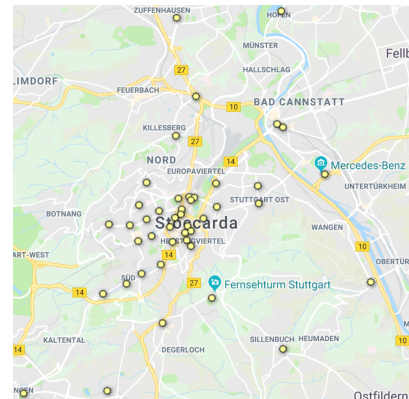


Figure 3.4: Map of Stuttgart with the rental zones

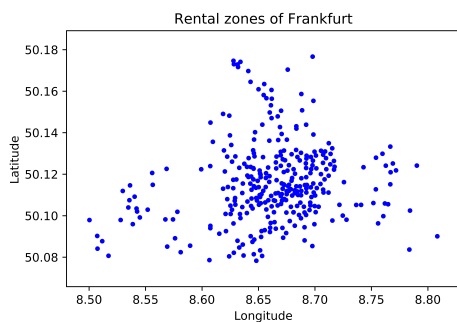


Figure 3.5: Scattered points of Frankfurt

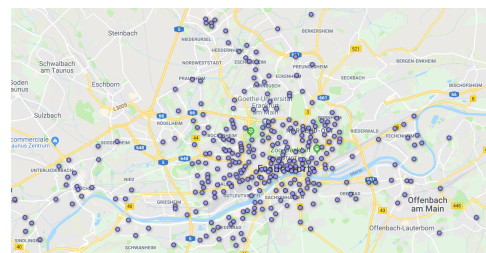


Figure 3.6: Map of Frankfurt with the rental zones

### 3.3. Exploration of the dataset

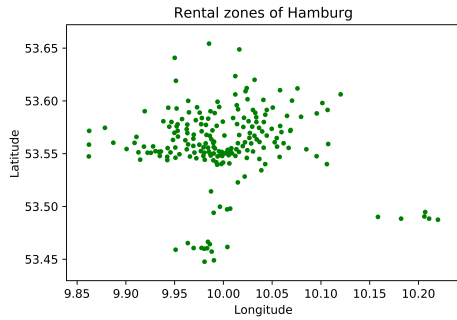


Figure 3.7: Scattered points of Hamburg

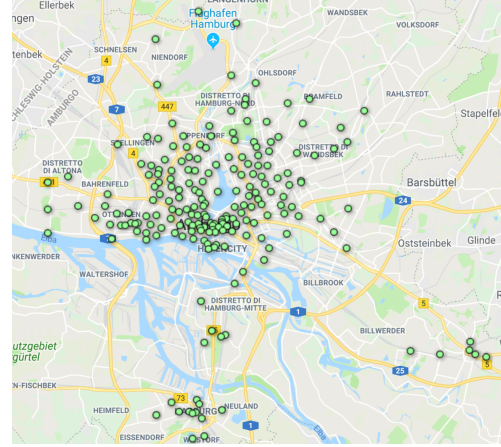


Figure 3.8: Map of Hamburg with the rental zones

As it can be noticed from the figures above in Frankfurt and in Hamburg there are a number of “stations” higher than Munchen and Stuttgart, this maybe because the service in the latter two cities is not widely used as well as the other two.

As said before for all the analysis three years are taken into account (2014, 2015 and 2016) and a filter is applied to the data in order to eliminate the outliers. In fact, all the bookings that last less than 3 minutes and more than 90 minutes are not considered real rentals. This because in the first case the customer could have delete the booking while the second case could regard a system error or a maintenance period.

Once the data are ready to be analysed some statistics are taken into account in order to know the evolution of the data during the different years.

First of all the number of vehicles is counted in each city to understand if the system grew up from 2014 to 2016. The plots in Figures A.43, A.44, A.45 and A.46 show the amount of bikes in each year for all the cities.

### 3.3. Exploration of the dataset

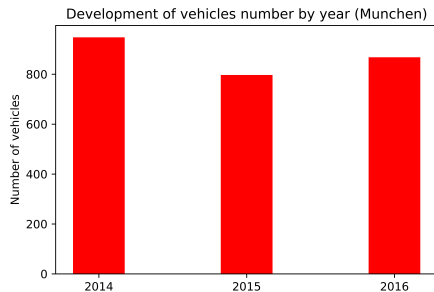


Figure 3.9: Number of vehicles by years in Munchen

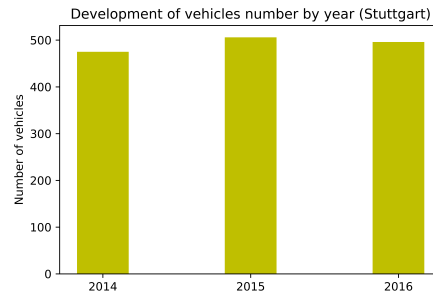


Figure 3.10: Number of vehicles by years in Stuttgart

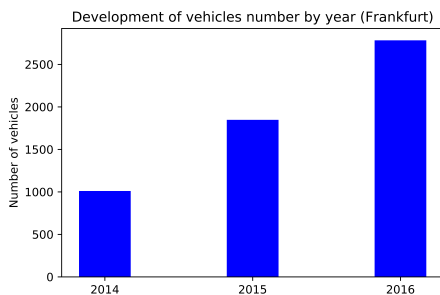


Figure 3.11: Number of vehicles by years in Frankfurt

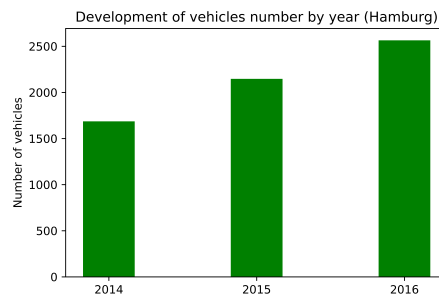


Figure 3.12: Number of vehicles by years in Hamburg

What is interesting to notice from these bar plots is that the city of Frankfurt is the one that mostly grew up with respect the other city in which the number of vehicles available is almost constant, in particular the precise number of bikes in Frankfurt in the different years is shown in Table 3.3

Year	Number of vehicles
2014	1010
2015	1847
2016	2783

Table 3.3: Evolution of number of vehicles of Frankfurt

This means that from 2014 to 2015 there is a growth of 82,9% while from 2015 to 2016 the growth of vehicles is of 50,68%. The city of Hamburg instead has the larger



### 3.3. Exploration of the dataset

number of vehicles this maybe because it is the bigger city among the four chosen, while both Munchen and Stuttgart have a less number of bikes that is constant in time. Since the amount of available bikes changes in some of this cities another interesting statistic is the development of the number of zones in years. Due to the growth of the vehicles it is possible that the number of zones in which they can be booked or parked is changed too. This is the reason why that kind of data is analysed and plotted in the following figures.

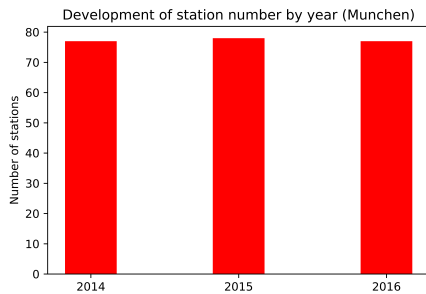


Figure 3.13: Number of zones by years in Munchen

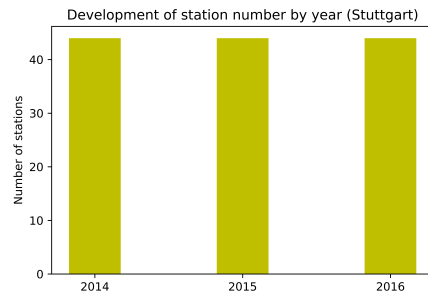


Figure 3.14: Number of zones by years in Stuttgart

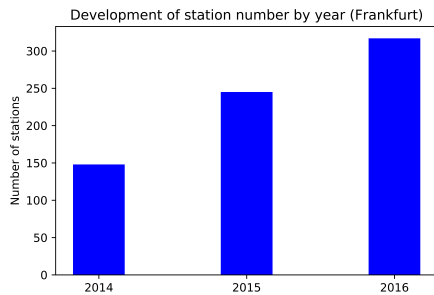


Figure 3.15: Number of zones by years in Frankfurt

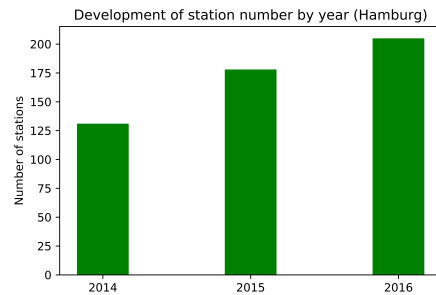


Figure 3.16: Number of zones by years in Hamburg

As expected, the evolution of the number of zones reflects the pattern of the vehicles. In fact, in Frankfurt there is an increase in the number of “stations” from 2014 to 2016. In particular here the precise number of zones is depicted in Table 3.4:

Year	Number of zones
2014	148
2015	245
2016	317

Table 3.4: Evolution of number of zones of Frankfurt

Hence, the growth in this case for Frankfurt is of 65,5% from 2014 and 2015 while between the last two years the number of zones increases of 29,4%. Another interesting thing to notice is the number of stations of Hamburg, even if the number of bikes is higher with respect to Frankfurt, the number of available zones is less.

Since the identifier of the booking is present in the table it is also used for the statistical analysis, in fact we decided to know the development of the number of bookings during the three years. This is another statistics that help this study to understand in which year the system is mostly used.

From the Figures 3.17, 3.18, 3.19 and 3.20 it can be noticed that the majority of rentals are in Hamburg this means that the system is most used in this city respect to the others, maybe due to the high number of population.

### 3.3. Exploration of the dataset

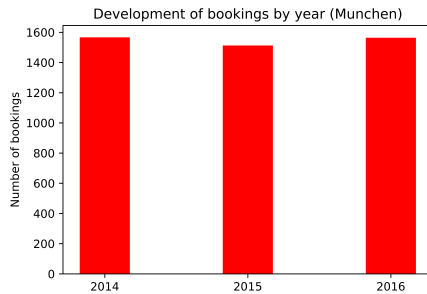


Figure 3.17: Number of bookings by years in Munchen

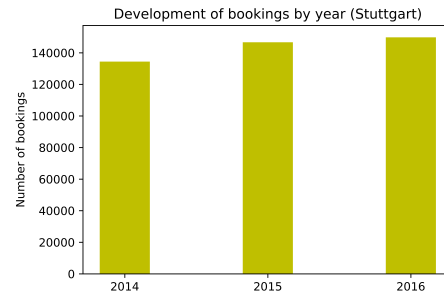


Figure 3.18: Number of bookings by years in Stuttgart

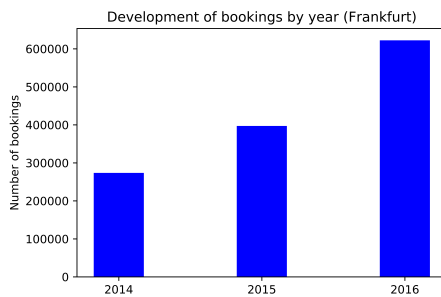


Figure 3.19: Number of bookings by years in Frankfurt

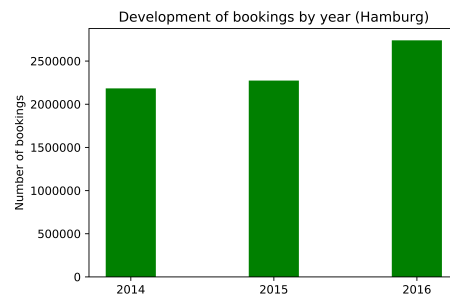


Figure 3.20: Number of bookings by years in Hamburg

Additionally it is visible that the number of bookings increases for the most again in Frankfurt as can be seen from the Table 3.5.

Year	Number of bookings
2014	273597
2015	397166
2016	622496

Table 3.5: Evolution of number of bookings of Frankfurt

In conclusion Frankfurt is the city from the ones chosen which has the larger growth regarding this bike sharing system.

Given the duration of each booking some interesting statistics can be done on this data, hence for each city three important parameters are taken into account the average, the median and the 90th percentile of the booking duration. In this way the usage of the system can be understood. To know better what these parameters represent they are explained in detail below with the help of a graphical representation.

- *Average of duration*: represents the arithmetic mean of some values in this case the duration of bookings is considered. From the Figure 3.21 it is visible that the average duration for all the cities is between 11 and 17 minutes. Stuttgart is the city in which the average duration is smaller than the other cities even if the first half hour is free, while Hamburg has the higher value, this maybe because the first one is the smallest city so a trip lasts less than the second one which is the biggest city.

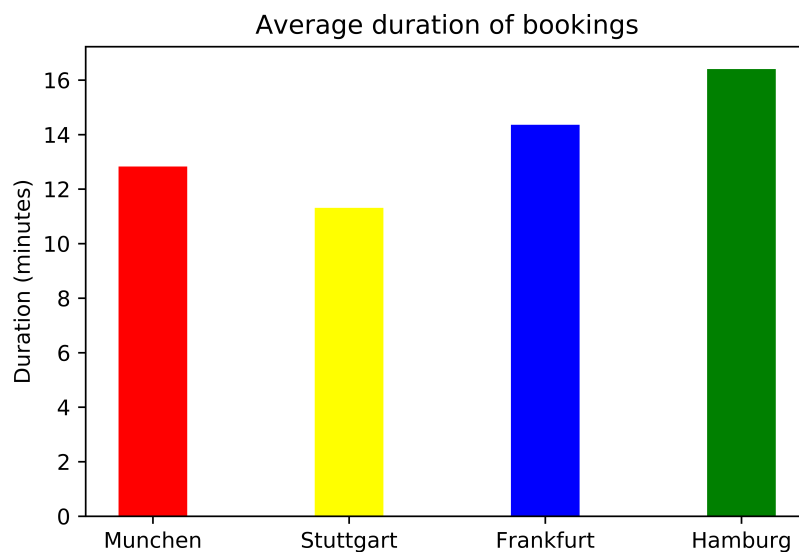


Figure 3.21: Average duration for each city

- *Median of duration*: in statistics the median indicates the central value of a ordered set of values, so it corresponds to the 50th percentile. If the number of values in the set is odd the median corresponds to the central value while if this number is even to obtain the median the arithmetic mean between the two values in the middle is performed.

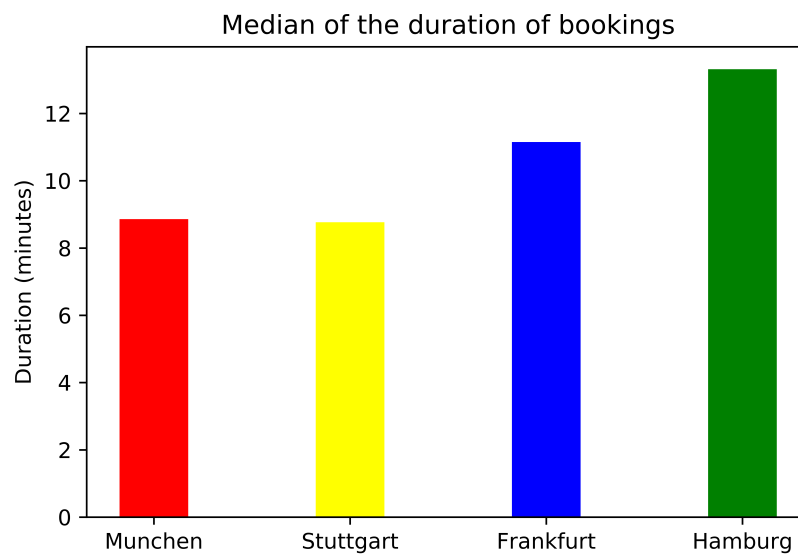


Figure 3.22: Median of duration for each city

- *90th percentile of duration*: given an ordered set of values the percentile in statistics indicates the probability under which a value can be found. In this case means that the 90% of the bookings last less than a certain value shown in Figure 3.23

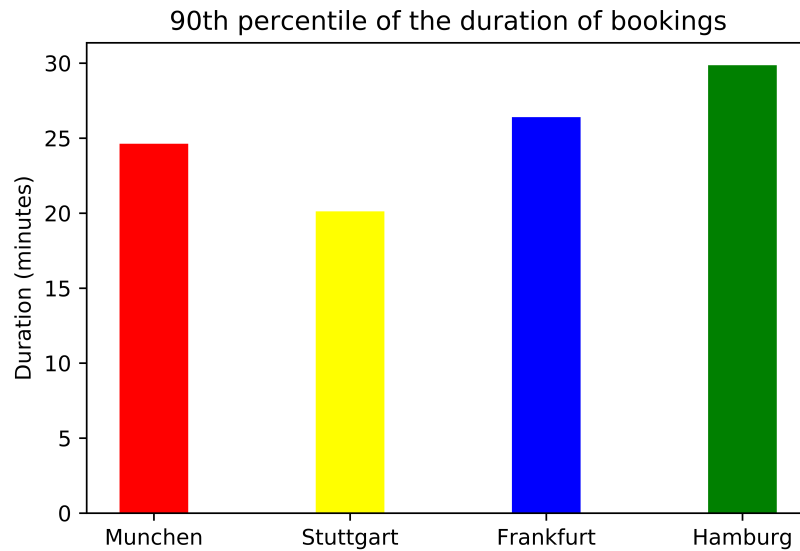


Figure 3.23: 90th percentile of duration for each city

From the Figure 3.23 it can be noticed that the higher value can be observed for Hamburg this means that the 90% of the bookings in this city last less than 30 minutes this maybe due to the big area of the city. It is also interesting to notice the value of Stuttgart which is the lower given that the first half an hour is free.

Once the statistics are done in order to have a general overview of the dataset, other kind of analysis is taken into account that can be divided into temporal and spatial analysis. This can help to understand better the habits of the users in terms of time in which the system is mostly used and which are the zones frequented a lot.

# Chapter 4

## Analysis and results

In this chapter all the results obtained applying the analysis on the data described in Chapter 3 are explained. As mentioned previously the analysis is divided into two types of data. The first kind of study takes into consideration how the usage of the system changes in time, so a temporal analysis is conducted while in a second step a spatial overview of the system is described in order to understand the most frequented zones.

### 4.1 Temporal analysis

In this section the temporal analysis of data is taken into account. For this kind of assessment the number of bookings is considered in order to understand how this value evolves daily, weekly and monthly. It is said in advance that for this type of analysis the city of Munchen is not mentioned due to the small amount of data present in the dataset, in fact considering the hourly average number of bookings this value during a day is less than one, this means that on average in a day there is a number of bookings less than one. For this reason, only the other three cities are analysed. To avoid repetitions for the most important plots only the city of Frankfurt is shown which is the most significant in terms of evolution of the system during the years. The plot of the other cities can be found in the appendix.

First of all the data from the dataset are grouped by year to have an estimation for all the years taken into account. To know the daily trend during the entire year, the

number of bookings is averaged upon the 365 days. This trend is very similar for all the cities but the city of Frankfurt is shown in order to understand better the growth of the system during the years.

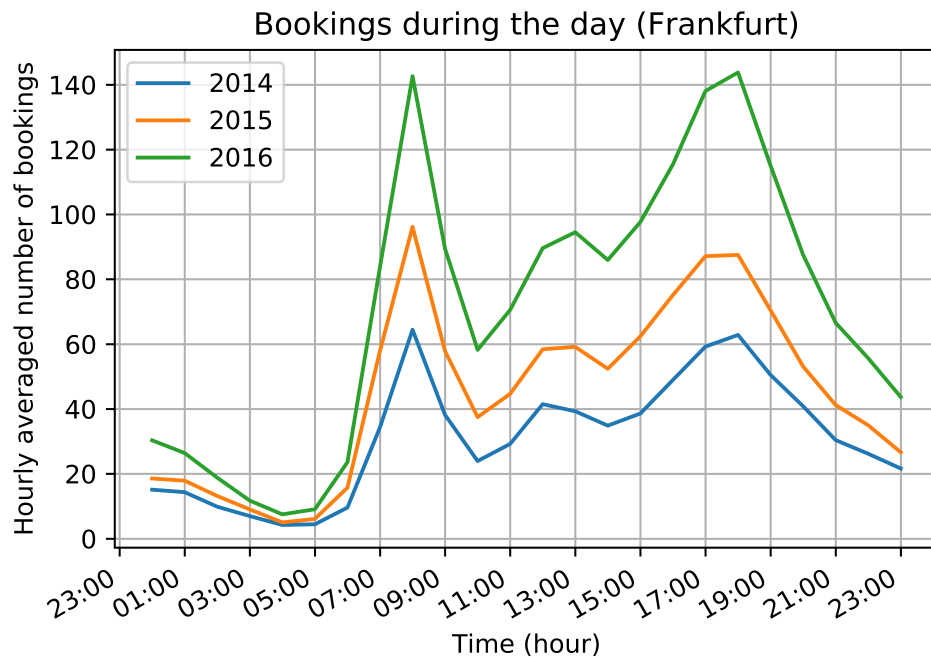


Figure 4.1: Bookings during the day of Frankfurt

From the Figure 4.1 it can be noticed that two main peaks are visible in two intervals, this means that on average the customers mostly use the service in the morning at 8 a.m. when work and school start and in the evening around 5 p.m. when people come back to home. Another small peak is present at noon maybe people use the service to go for lunch or run some errands. During the night hours the system is almost not used.

The second analysis is implemented to know the weekly trend of the usage upon the whole year but in this case the number of bookings is averaged on the 52 weeks present in a year. The city of Frankfurt is again depicted since the trend is equal for the other cities.



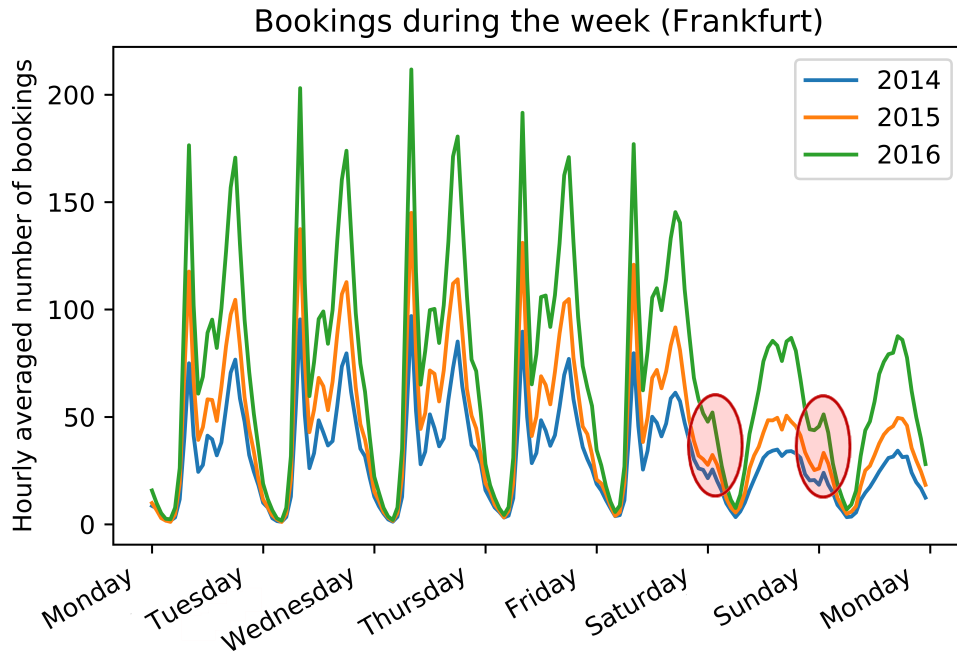


Figure 4.2: Bookings during the week of Frankfurt

As shown in Figure 4.2 even here the growth of the system in this city can be noted, moreover the trend during a day of the week reflects the daily trend seen before, in fact for each day the two main peaks are present. What is interesting to notice is that in the weekend the service is not used in the same way as weekdays and the peaks are not so discernible this because users do not have to go to work or school and hence the system is less used. Another interesting detail to highlight is the slight peak visible in the Friday and Saturday night hours because young people may go out in these hours for leisure activities.

In the third analysis the daily patten for each month is considered doing the average of the booking number upon the days of every month. The monthly trend reflects the daily one for all the months, in fact the three peaks are visible in each curve. Even here the result is the same for all the cities hence, the city of Frankfurt is again chosen for the illustration.

## 4.1. Temporal analysis

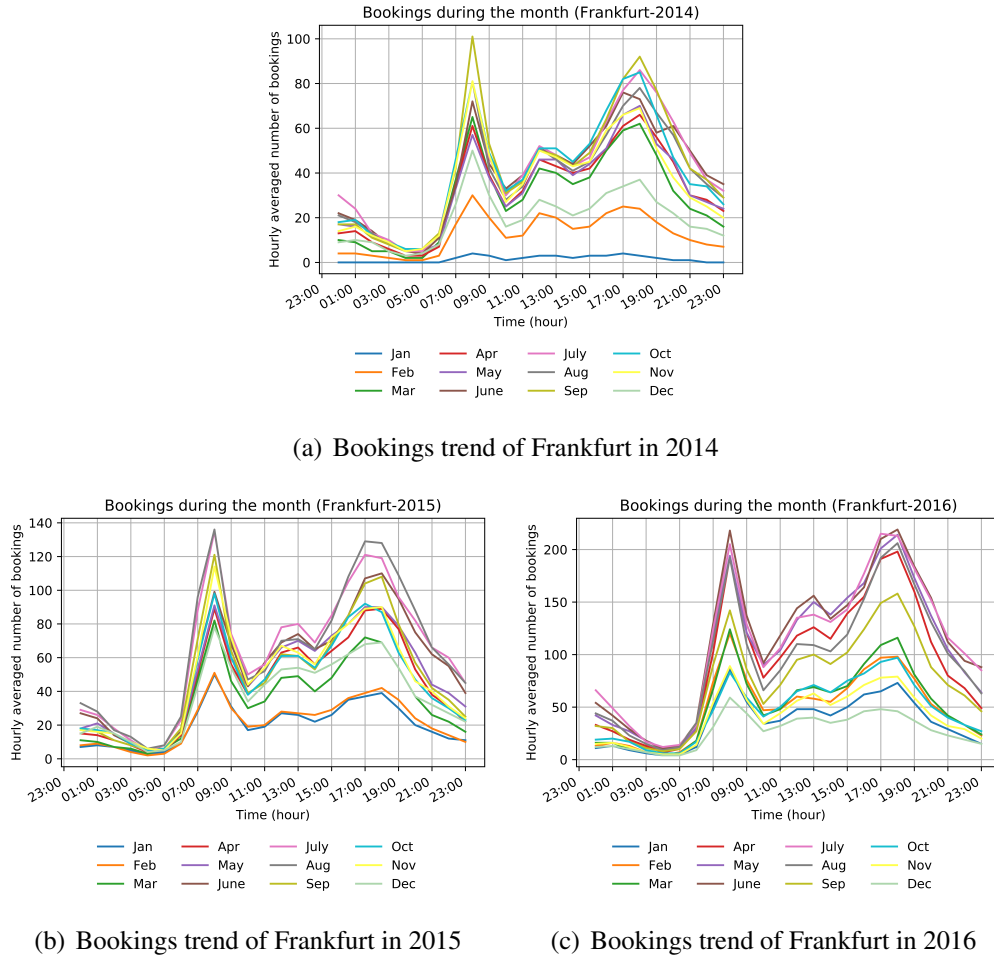


Figure 4.3: Monthly bookings trend of Frankfurt by year

As it is depicted in the Figure 4.3 the averaged number of bookings is lower in the cold month such as December, January and February. This is reasonable given that the bad weather conditions can influence the usage of the bike sharing system. On the contrary the highest number of rentals are done in Summer when the weather is good. Once again the development of the service in this city can be noticed looking at the increase of the averaged number of booking during the years.

At this point the effective number of rentals is calculated and plotted to understand the distribution in the three years. As before the trend is similar for all the cities.

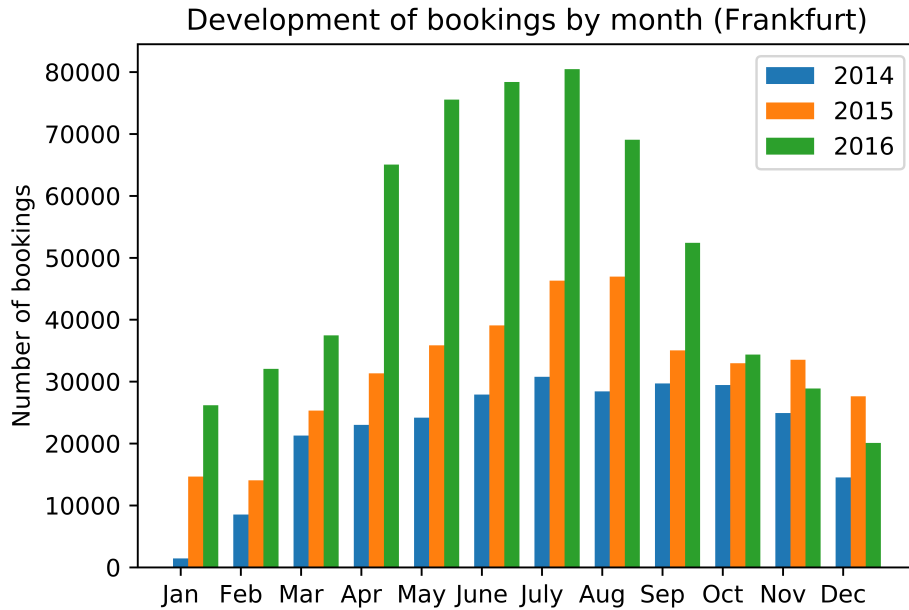


Figure 4.4: Bookings by month of Frankfurt

From the plot in Figure 4.4 the growth of the usage of the system in 2016 is evident once again, moreover for all the years the Summer months are the ones with the larger number of rental, hence the willingness of the users to use the system is higher in those months.

In the end to have a wider view of the booking trend a zoom on the three peaks is made from the Figure 4.3 and with the values allocated in those intervals some statistics are calculated. In Figure 4.5 the zoom of the ranges of the two main peaks is shown.

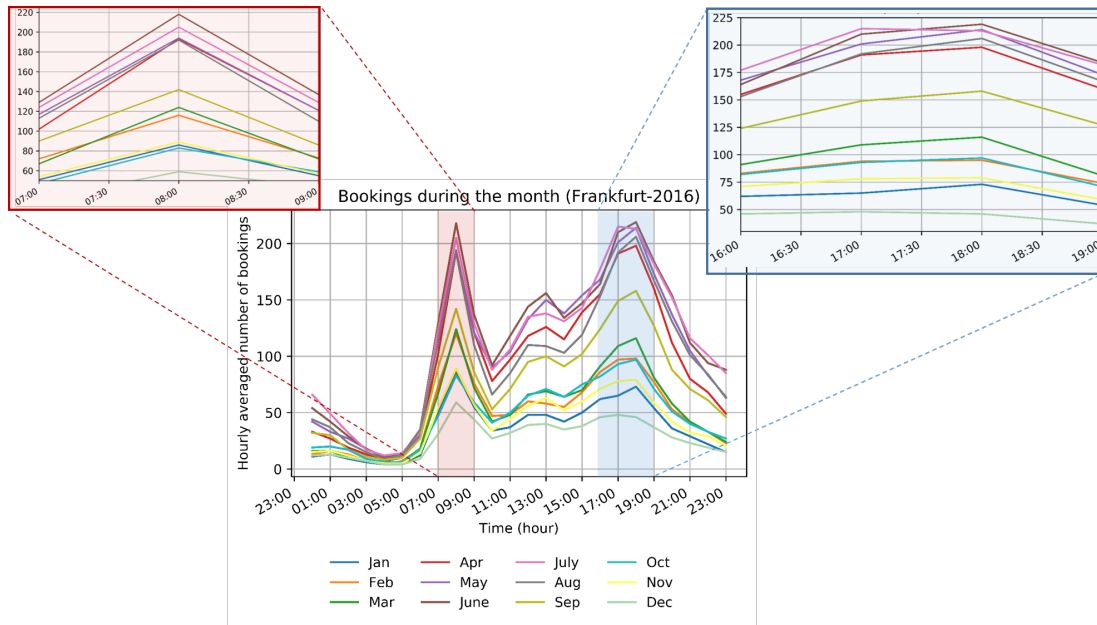


Figure 4.5: Zoom of the main peaks in monthly booking trend of Frankfurt

The considered statistics are the median, the 25th, the 75th and the 90th percentile of the number of bookings allocated in the three ranges of the peaks from 7 a.m. to 9 a.m., from 12 p.m. to 2 p.m. and from 4 p.m to 7 p.m.

For both Frankfurt and Stuttgart the trend is similar for all years and from Figure 4.6 it can be noticed that for all the statistics the values obtained at 8 a.m. and at 5 p.m. are really similar this means that the system usage is analogue in these two intervals while at 12 p.m. the value is lower. Another interesting thing to notice is that for all the cities the values for the 75th and the 90th are very similar to each other this means that between these two statistics the values do not grow a lot.

## 4.1. Temporal analysis

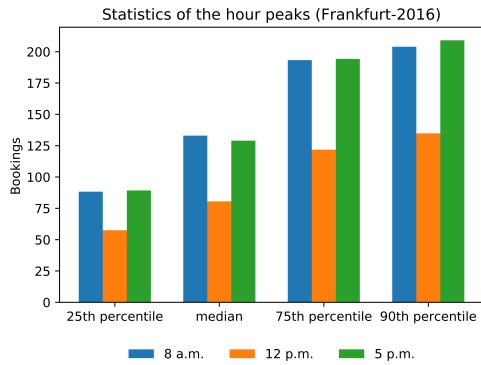


Figure 4.6: Statistics of the peak hours of Frankfurt

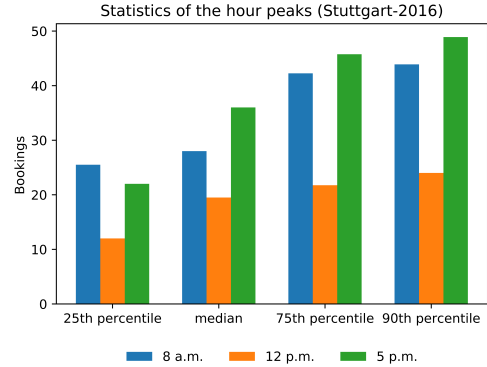


Figure 4.7: Statistics of the peak hours of Stuttgart

In the case of Hamburg only for the 25% of the customers the usage of the system is similar on both 8 a.m. and 5 p.m., while the majority of the users prefer to take a bike at 5 p.m.

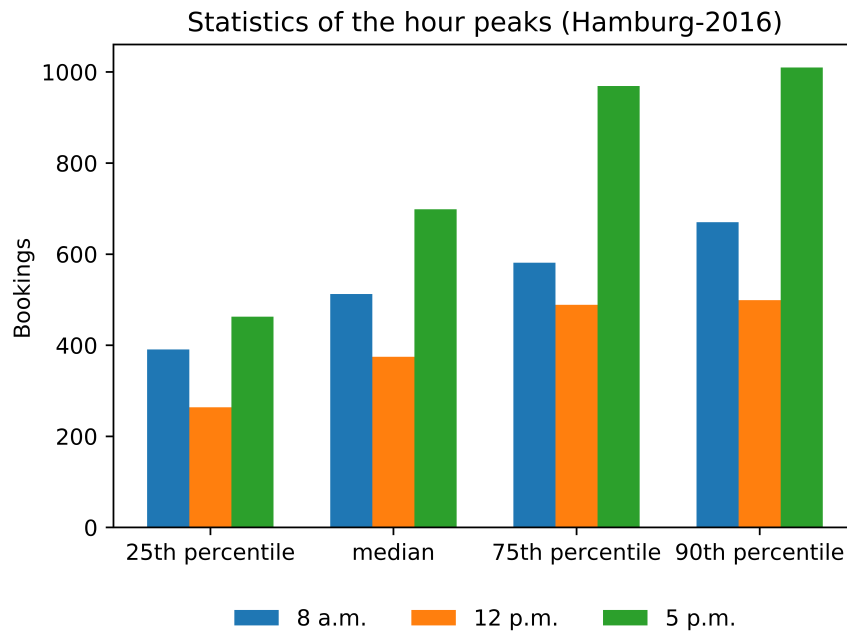


Figure 4.8: Statistics of the peak hours of Hamburg

In the end based on the temporal analysis we can conclude that the users prefer to take the bicycles in some specific intervals of time in particular around 8 a.m. and around 5 p.m during weekdays. This means that people are inclined to use the service to go to work or school and come back to home at the end of the day. Moreover, in weekends the system is not so used such as in workdays and there is not evident peaks except for Friday and Saturday night when young people have leisure activities. Another consideration can be done on the monthly usage, in fact the system is mostly used in Summer months due to the good weather conditions.

After the temporal analysis another kind of study over spatial data is made in order to understand how customers move and which are the most frequented areas.

## 4.2 Spatial analysis

In this section a spatial analysis is conducted for all the cities in order to understand how people move. For this purpose the distance in meters of each booking is considered from the column `DISTANCE` of the CSV.

Initially the distance of the trips is taken into account to know how long the trips are. Three clusters are generated and explained in Table 4.1 which represent a certain range of distance in meter.

Cluster	Meters
A	less than 100
B	between 100 and 500
C	more than 500

Table 4.1: Cluster definition

What can be noticed from the Figures 4.9, 4.10, 4.11 and 4.12 is that only in Munchen the majority of the rentals are done for a trip distance less than 100 meters and more than 500 meters, while for the other three cities the mostly of the bookings fall into Cluster C.

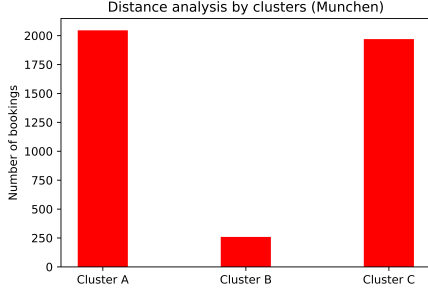


Figure 4.9: Distance trip of Munchen

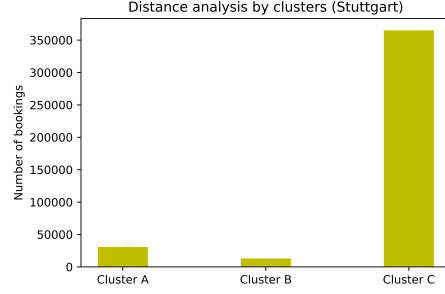


Figure 4.10: Distance trip of Stuttgart

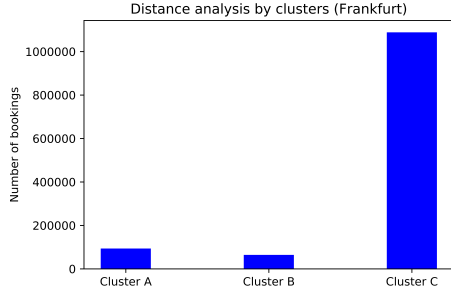


Figure 4.11: Distance trip of Frankfurt

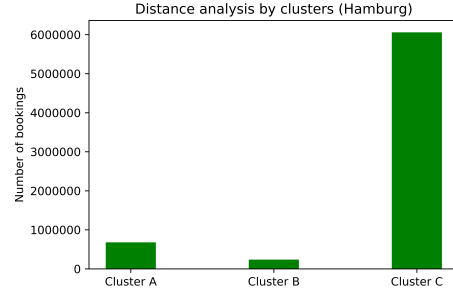


Figure 4.12: Distance trip of Hamburg

To explore in a better way in which city the trips are longer, the **Cumulative Distribution Function (CDF)** of the distance travelled for each city, is calculated. To this purpose a filter is applied to those values to eliminate the outliers. Hence, too short and too long travel (less than 2 meters and more than 4000 meters) are not treated because they can be considered as a system error or an accident.

The CDF of a real random variable  $X$  is represented by the function:

$$F_X(x) = P(X \leq x) \quad (4.1)$$

where  $P(X \leq x)$  constitutes the probability that the random variable  $X$  takes values less or equal to  $x$ .

Looking at the plot in the Figure 4.13 it can be noticed that the longer trips are made in Hamburg while Munchen seems the city with the shorter travel distance covered.

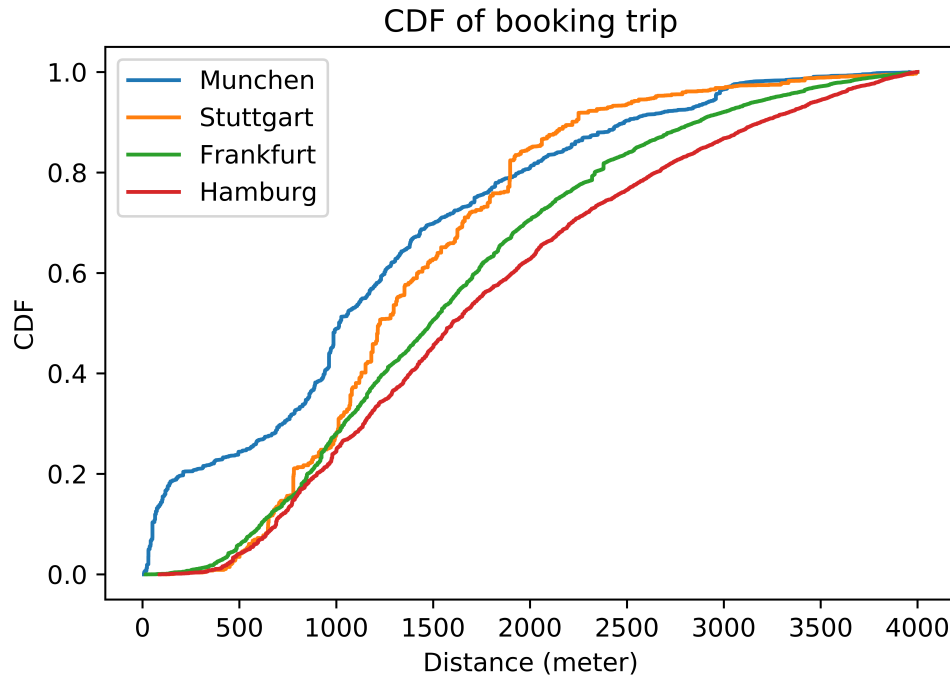


Figure 4.13: CDF of booking trip by city

To understand where people move, an analysis on the zones is done in order to know from which areas users start the rental and in which areas they park the bike. For this purpose an **Origin/Destination** matrix is built up in which the total number of bookings from the origin, represented by the rows, to the destination, represented by the columns, is evaluated.

As can be seen from the example depicted in Figure 4.14 the matrix is sparse and since the coordinates in the CSV identify a zone and not the precise point in which the bike is taken or left, it is possible to have bookings from a zone to the same area.



## 4.2. Spatial analysis

Index	354	388	1532	1559	1589	1610	3531
354	16	0	0	2	4	3	0
388	0	17	0	1	0	0	1
1532	0	0	1	0	0	0	0
1559	5	0	0	15	1	1	0
1589	1	0	0	1	11	0	0
1610	5	0	0	0	1	16	1
3531	0	0	0	1	1	3	13

Figure 4.14: Origin/Destination matrix

Once the matrix is prepared, to have the number of bookings started in each zone, the sum of all the values in every row is calculated, and to have the number of parkings done in each area the same calculation is evaluated by column. As second step the three main time intervals (8a.m., 12p.m., 5p.m.) are selected and for each city a new CSV is created per interval with the ID of the zone, the coordinates and the sum of the bookings/parkings for each area. In this way the total number of bookings done in a zone and the total parkings in the same area are extracted.

With these information the CDF of the load of bookings/parkings for every zone in each time interval is calculated for the four cities. In this way we can understand how spread are the bookings/parkings in all the zones. In Figures 4.15 and 4.16 the city of Hamburg is depicted while the other cities are shown in the appendix.

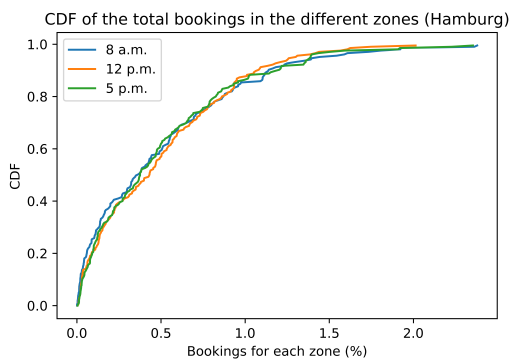


Figure 4.15: CDF of total bookings for each zone of Hamburg

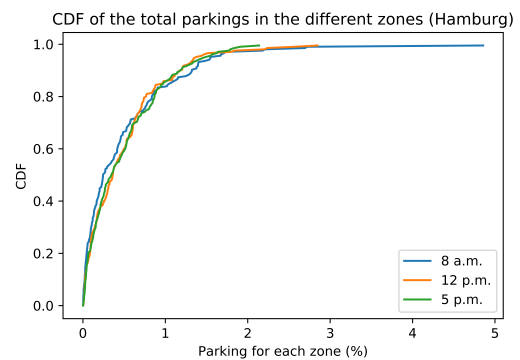


Figure 4.16: CDF of total parkings for each zone of Hamburg

From the above figures we can conclude that in Hamburg the percentage of bookings and parkings in the three time intervals in the different zones is almost the same. This means that the system usage is very spread in all the areas considered. This result is the same also for Frankfurt and Munchen. In Stuttgart the three curves are a bit different as it can be noticed in Figures 4.17 and 4.18. This means that for the same zone the same percentage of bookings/parkings is reached in a different way during the three time intervals.

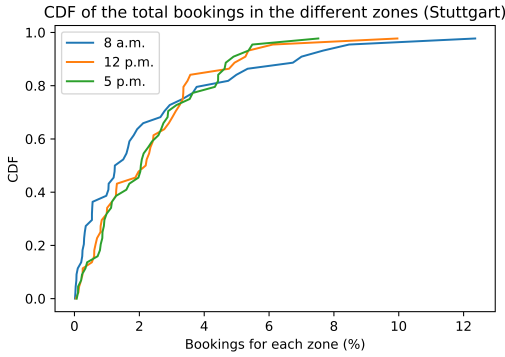


Figure 4.17: CDF of total bookings for each zone of Stuttgart

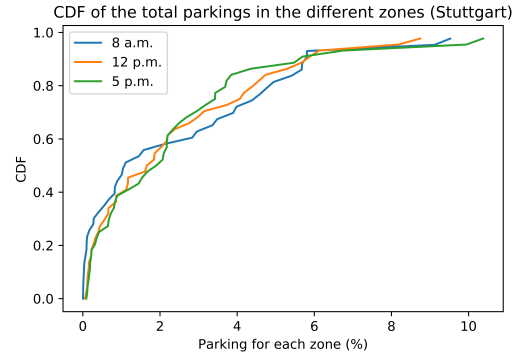


Figure 4.18: CDF of total parkings for each zone of Stuttgart

For the last analysis the real zones in the three main time intervals are considered in order to know which are the patterns followed by the users. Since the available coordinates represent the area in which the bike can be taken or left and not the precise points in which it can be found, even if they are near to each other, we decided to divide each zone with the help of the Voronoi diagram.

Giving a finite set of points a **Voronoi Diagram** defines a division of a plane into regions on the basis of euclidean distance between points. For example given two points, in a Voronoi diagram, a line is traced between the two. Then, its perpendicular segment, passing through the middle point of the aforementioned line, is traced. In this way two regions are created.

After the creation of the Voronoi map, each cell has been gradually coloured from white to red on the basis of how many bookings/parkings are found in that region during the three years, so to understand which are the zones more frequented.

## 4.2. Spatial analysis

In the following figures the Voronoi map of bookings and parkings in the peak time intervals of Hamburg is shown.

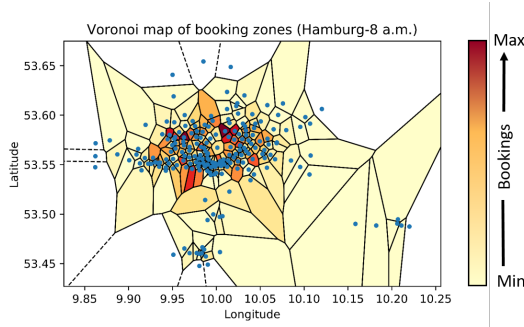


Figure 4.19: Voronoi map of bookings (Hamburg-8 a.m.)

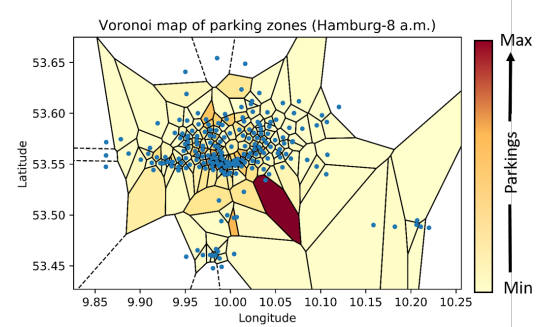


Figure 4.20: Voronoi map of parkings (Hamburg-8 a.m.)

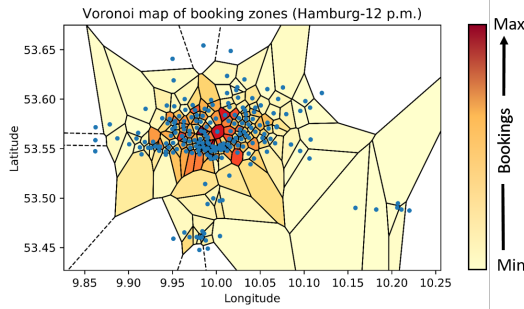


Figure 4.21: Voronoi map of bookings (Hamburg-12 p.m.)

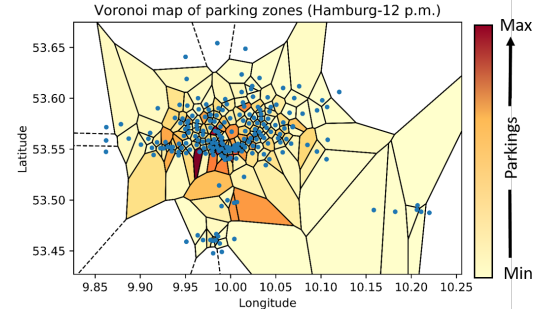


Figure 4.22: Voronoi map of parkings (Hamburg-12 p.m.)

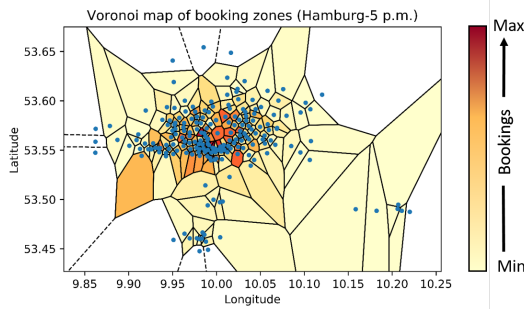


Figure 4.23: Voronoi map of bookings (Hamburg-5 p.m.)

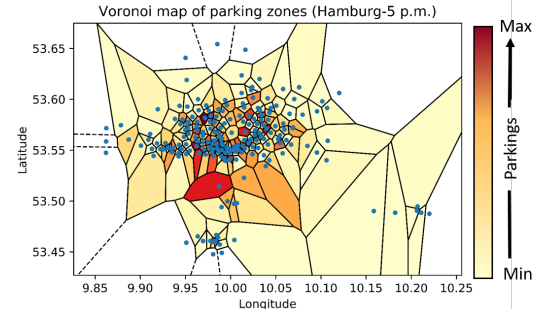


Figure 4.24: Voronoi map of parkings (Hamburg-5 p.m.)

## 4.2. Spatial analysis

In the end to have a better idea of the frequented zones the Voronoi diagram is represented onto a geographical map for each city. In the figures below the city of Hamburg in the two main time intervals is shown in order to have a comparison with the previous plots. The green polygons represent those areas in which few bicycles are taken or parked while the red polygons are the most frequented zones.

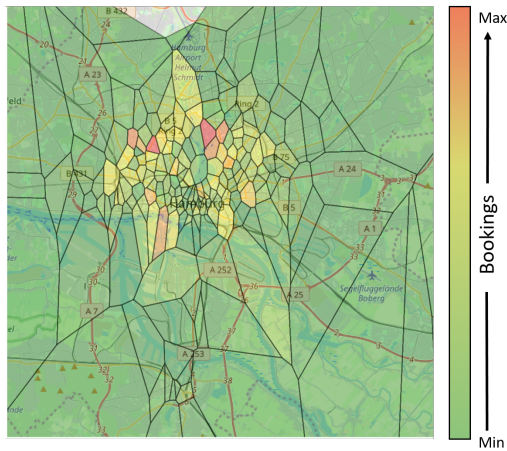


Figure 4.25: Voronoi booking zones (Hamburg-8 a.m.)

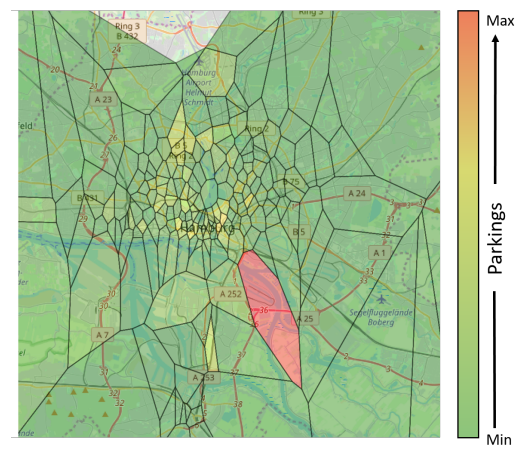


Figure 4.26: Voronoi parking zones (Hamburg-8 a.m.)



Figure 4.27: Voronoi booking zones (Hamburg-5 p.m.)

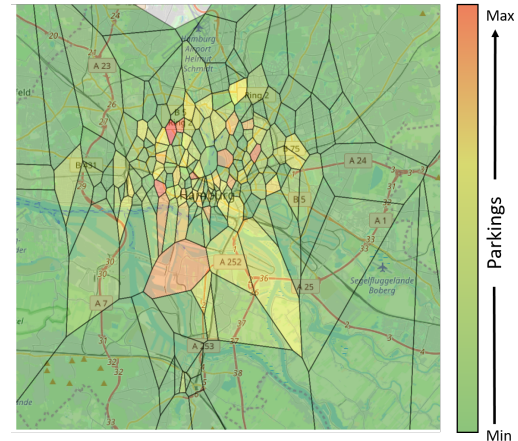


Figure 4.28: Voronoi parking zones (Hamburg-5 p.m.)

In Figures 4.25 and 4.26 are compared the zones in which the majority of bookings and parkings occur at 8 a.m. It can be noticed that the customers take the bikes in the peripheral zones which are a residential areas and in proximity of the central railway station and then left them both in centre or in the areas near the university of Hamburg (yellow area in proximity of the city centre) and in the red polygon where the golf and football club and a public school can be found.

At 5 p.m. we have an opposite pattern the majority of bookings occur in the central zone and close to the university and they finish in the peripheral areas or in proximity of railway stations such as Dammtor which is near the university.

This means that users use the system to approach to work areas or to stations if they are commuters and to come back towards home at the end of the day.

In the city of Stuttgart it is also used in association with the public transport system in fact customers approach to bus terminals or to underground with the bike in order to move then with the means of public transport.

This kind of analysis in association with the temporal one give us some ideas. In fact given the hours in which the system is utilized it can be noticed that the service is mostly used by workers or students. Moreover, the zones visited are often in the centre of the city, which is a commercial area, or in proximity of schools. In addition it seems that in some cities the service is used to reach the desired zones while in other cases such as Stuttgart the users prefer use the system to approach to stations, bus terminals or underground.

Another consideration can be done on the customers of the service. Since the registration is required to use the service and a fee must be paid, it is less used by tourists. This explains why there are not anomalous peaks during the years for important events.

## Chapter 5

### Conclusion

During the last few years the free-floating bike sharing system has grown a lot and the majority of the studies found on it regard how the bicycles can be repositioned in order to help customers in finding them.

For this reason this thesis wants to analysed data available on a free-floating system in order to understand the habits of the customers in terms of when and how they move.

In the first part of the work a real-time analysis is conducted in order to have an immediate estimation of the usage of the system. To obtain available real-time data has been difficult since the companies are private and there were not data put at the disposal, for this reason a GitHub page is found in which a list of APIs of some bike sharing platforms described and each API is related to one bike sharing system.

As second step, the city of Milan is chosen and the service of “Ofo”, which is one of the most important bike sharing system, is considered for the analysis. Thanks to a request to the server specifying the coordinates of a point of the city, the data of available bikes placed in a circle with a radius of 500 meters are retrieved. For this reason a Python script has been implemented to collect data in a parallel way for all the points of the city. At the same time, a crawler was built up in order to process these data. The aim of the crawler is to create four collections (*activeParkings*, *permanentParkings*, *activeBookings* and *permanentBookings*) and to insert the bikes in the right collection in order to have an estimation of the usage of the service.

But with this kind of data some problems occur such as the coverage of the city, in fact since the requests give back all the bikes present in the aforementioned circles the

---

division of the city is not so trivial and the *token* used as a parameter for the request has a deadline, so a lack of data occurs.

Due to the issues found with the acquisition of real-time data, another kind of data are taken into account: the historical data.

Deutsche Bahn is one of the most important railway company in the world. It provides different datasets in a CSV form with the data of “Call A Bike system from the past years (2014, 2015, 2016). In this work the four main German cities are considered for the analysis: Munchen, Stuttgart, Frankfurt and Hamburg. After choosing the interesting data from the website, the tables have been clean and adjusted for the purpose of this study.

Initially a statistical analysis is made in order to have a general overview of the dataset and to understand how the system evolves during the different years. Even if “Call A Bike is a free-floating bike sharing system the vehicles must be taken and left in certain zones. That areas are defined such that the bicycles are easy to found for example intersections and central zones. Hence, the development of the number of vehicles, the number of zones and the number of booking is calculated in years. From this analysis it can be noticed that Frankfurt is the city with the highest growth of the system. Moreover, another statistical analysis is made in order to know the average, the median and the 90th percentile on the duration of the bookings. This analysis shows that Stuttgart is the city with the lower booking duration even if the first half hour is free.

In the end, a temporal and a spatial analysis is made to understand when and where people move. The trend of the hourly averaged number of bookings is analysed during the day, during the week and during the months. The results show that for all these configurations two peaks are relevant during the day, one in the morning and one in the evening. Another smaller peak can be noticed at noon. This means that customers mostly use the system around 8 a.m. when the work and school start and around 5 p.m to come back to home. In addition during the weekend the service is not used such as in workdays and there is not a clear distinction of peaks, except for Friday and Saturday night when young people go out for leisure activities. Looking at the development of bookings during the months it can be noted that the system usage grows in the Summer months this maybe because of the good weather.

---

For what concern the spatial analysis users make trips of more than 500 meters except for Munchen in which there are also a lot of trips less than 100 meters. Moreover, with the help of a Voronoi diagrams the movements in the peak hours are analysed and the results show that the higher number of trips is made in some particular zones. In fact at 8 a.m. customers use the bike to go from a peripheral area to work/school in the centre of the city or to activity areas and at 5 p.m. the pattern is opposite, hence people use bicycles to come back to home.

Another way to use the system is in association with public transport system in fact users approach to bus terminals, to underground or to railway stations with the bike in order to move then with the means of public transport.

As future works, the real-time system can be implemented with the help of the bike sharing companies. If the data can be retrieved in a real-time manner the crawler can be adjusted on the basis of the needs and the data can be consequently processed, to reach results in a real-time manner.

For what concern the analysis done with the help of the historical data, the results could be useful for the companies of the bike sharing systems. In fact with this analysis they can have a knowledge on both the time range in which the system is most used and the areas more frequented by the customers. With these information a reallocation strategy can be implemented on the basis of the results obtained and the management of the service can be enhanced.

The companies can also minimize the disservice during the maintenance of the fleet. In fact with this analysis the inactivity periods can be known and in these particular intervals the maintenance can be done without affecting the bicycle usage.

Another interested future work could be use this analysis for the integration of the bike sharing system with the public transport to improve the latter. To minimize the cost, the public transport companies could decide to eliminate the means of transport in the hours of the day in which the demand is low and substitute them with the bike sharing system. In this way the buses unused could be useful in other zones in which the demand is high or they can be eliminated in order to have a positive effect on the environment. To encourage this new type of transportation a unique ticket could be created for both means of transport and bicycles.



# **Appendices**

# Appendix A

## Additional plots and tables

### A.1 Statistical analysis

Year	Number of bookings
2014	1567
2015	1513
2016	1564

Table A.1: Development of bookings by years (Munchen)

Year	Number of bookings
2014	134424
2015	146672
2016	149816

Table A.2: Development of bookings by years (Stuttgart)

Year	Number of bookings
2014	2183882
2015	2274375
2016	2740432

Table A.3: Development of bookings by years (Hamburg)

City	Average	Median	90th percentile
Munchen	12.83	8.86	24.62
Stuttgart	11.31	8.77	20.12
Frankfurt	14.36	11.15	26.4
Hamburg	16.41	13.32	29.87

Table A.4: Average, Median and 90th percentile of booking duration by city

Year	Number of vehicles
2014	948
2015	797
2016	868

Table A.5: Development of vehicles by years (Munchen)

Year	Number of vehicles
2014	475
2015	506
2016	496

Table A.6: Development of vehicles by years (Stuttgart)

Year	Number of vehicles
2014	1686
2015	2147
2016	2565

Table A.7: Development of vehicles by years (Hamburg)

Year	Number of stations
2014	77
2015	78
2016	77

Table A.8: Development of stations by years (Munchen)

Year	Number of stations
2014	44
2015	44
2016	44

Table A.9: Development of stations by years (Stuttgart)

Year	Number of stations
2014	131
2015	178
2016	205

Table A.10: Development of stations by years (Hamburg)

## A.2 Temporal analysis

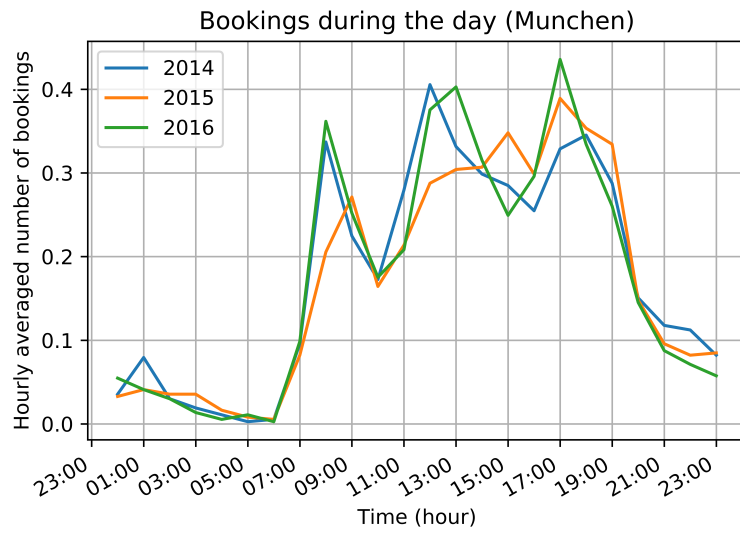


Figure A.1: Bookings during the day of Munchen

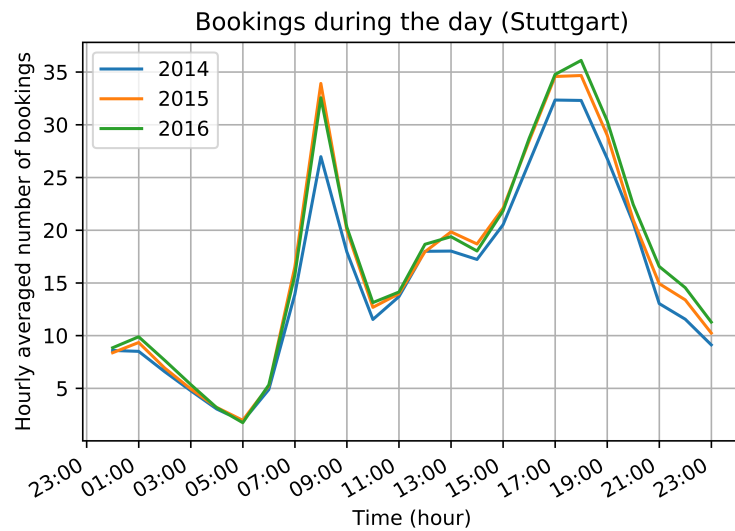


Figure A.2: Bookings during the day of Stuttgart

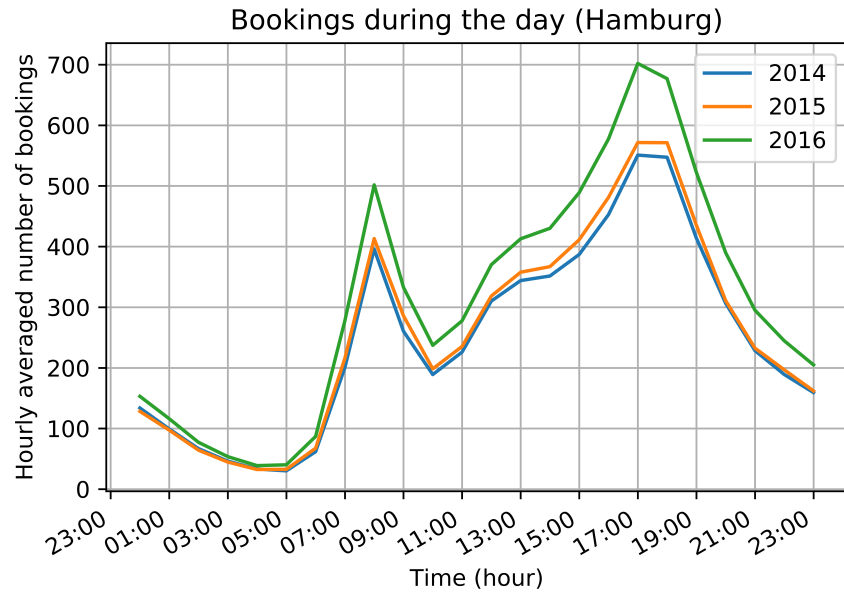


Figure A.3: Bookings during the day of Hamburg

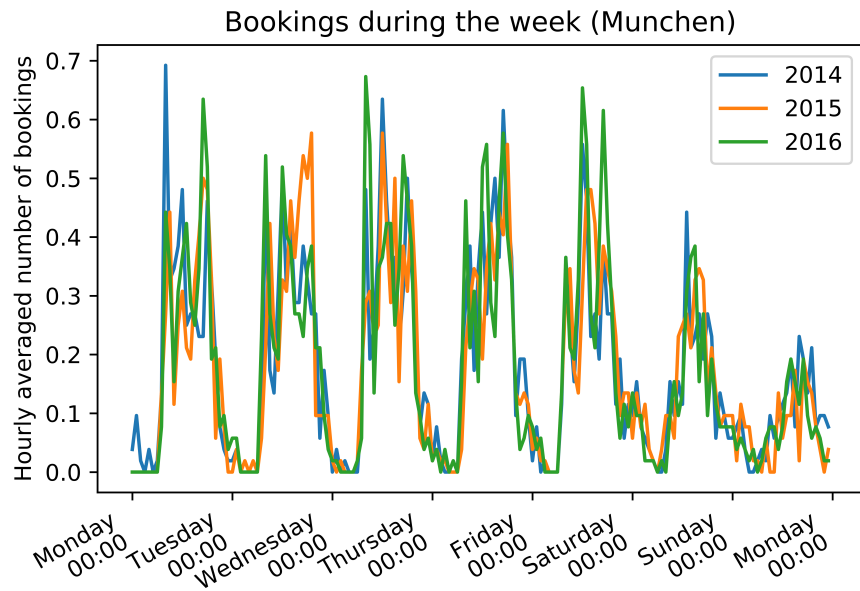


Figure A.4: Bookings during the week of Hamburg

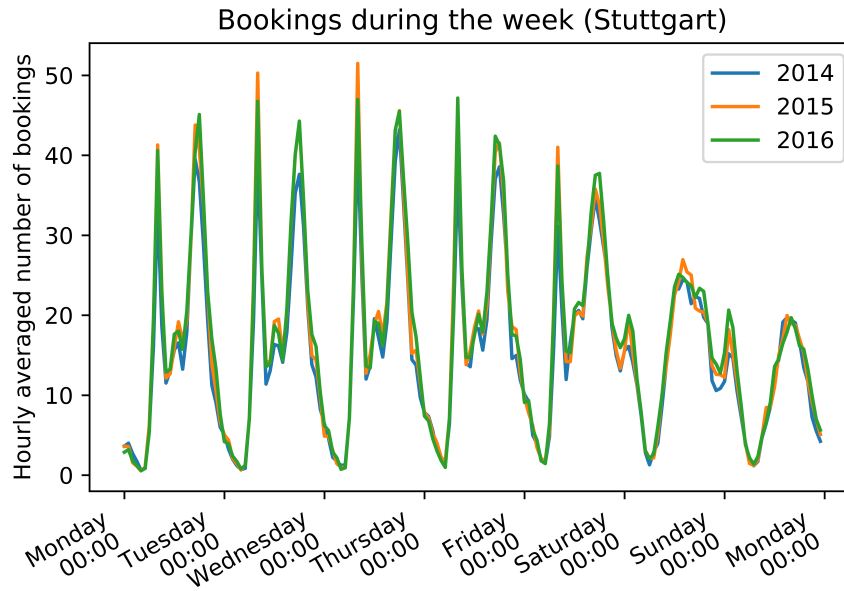


Figure A.5: Bookings during the week of Stuttgart

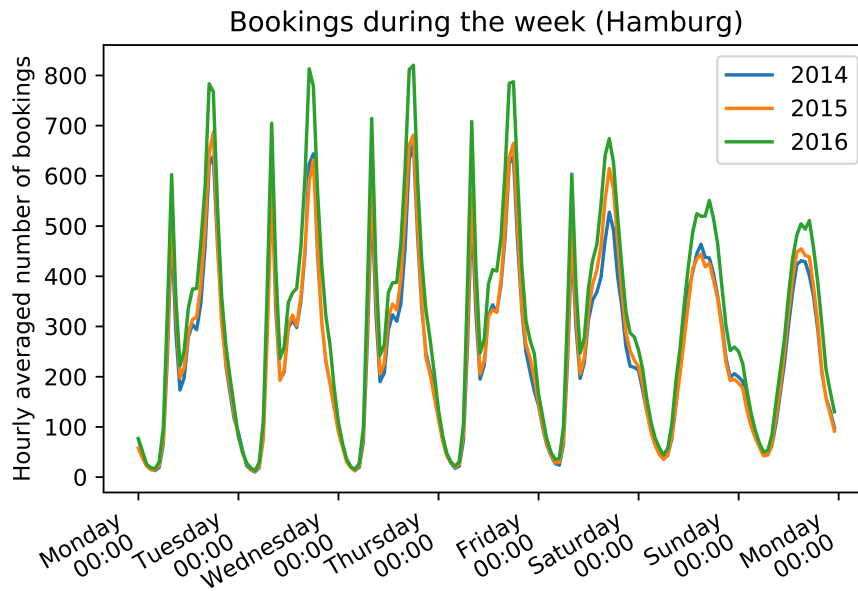


Figure A.6: Bookings during the week of Hamburg

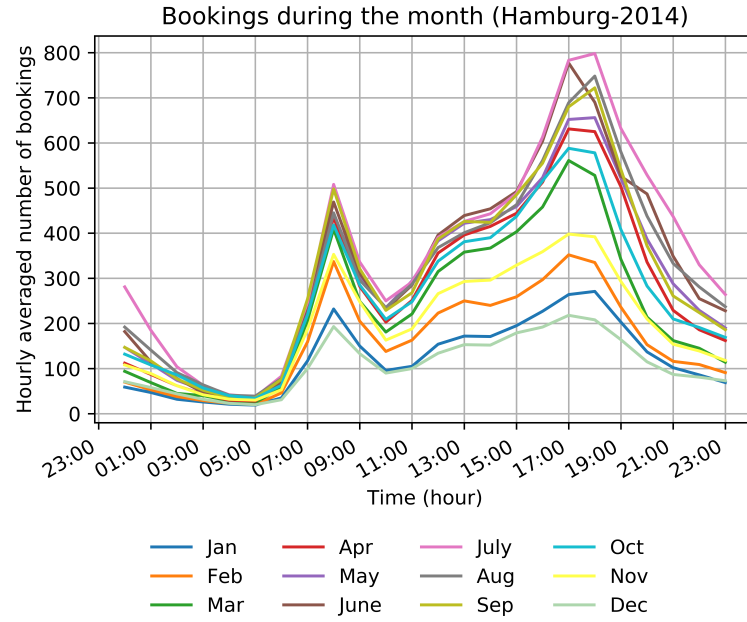


Figure A.7: Monthly booking trend of Hamburg 2014

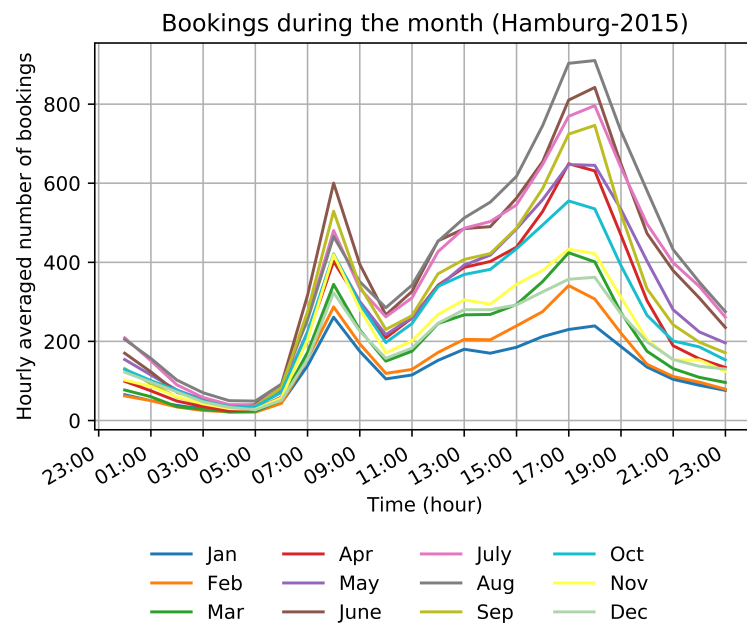


Figure A.8: Monthly booking trend of Hamburg 2015



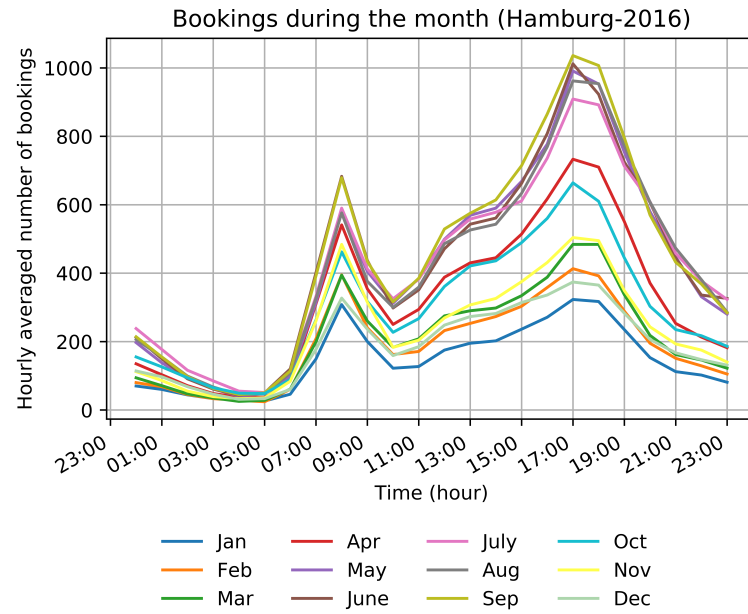


Figure A.9: Monthly booking trend of Hamburg 2016

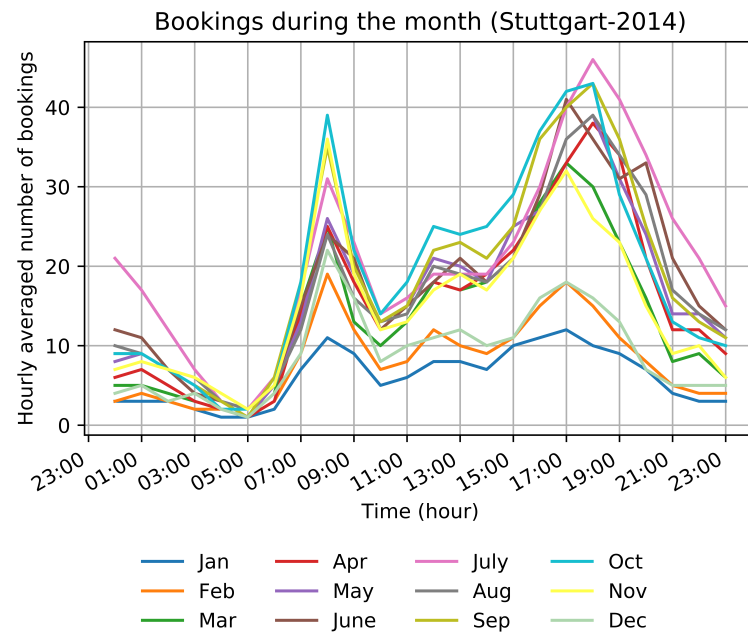


Figure A.10: Monthly booking trend of Stuttgart 2014

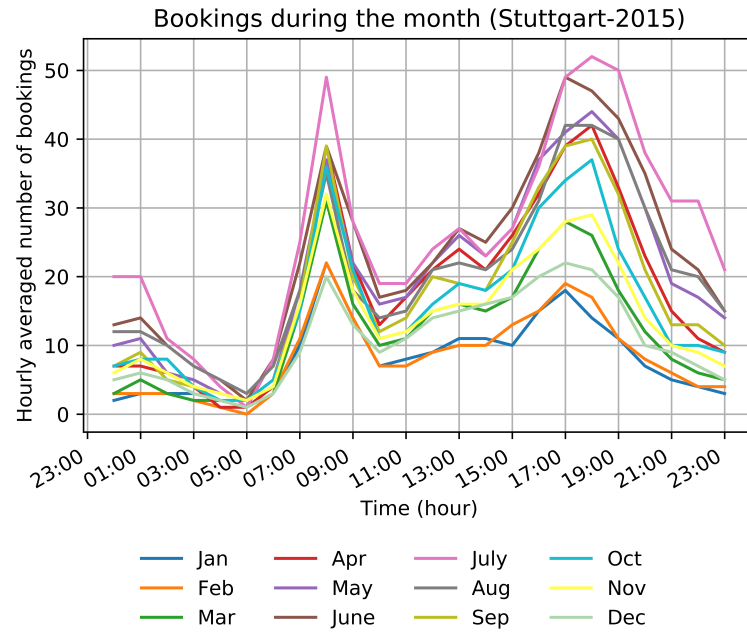


Figure A.11: Monthly booking trend of Stuttgart 2015

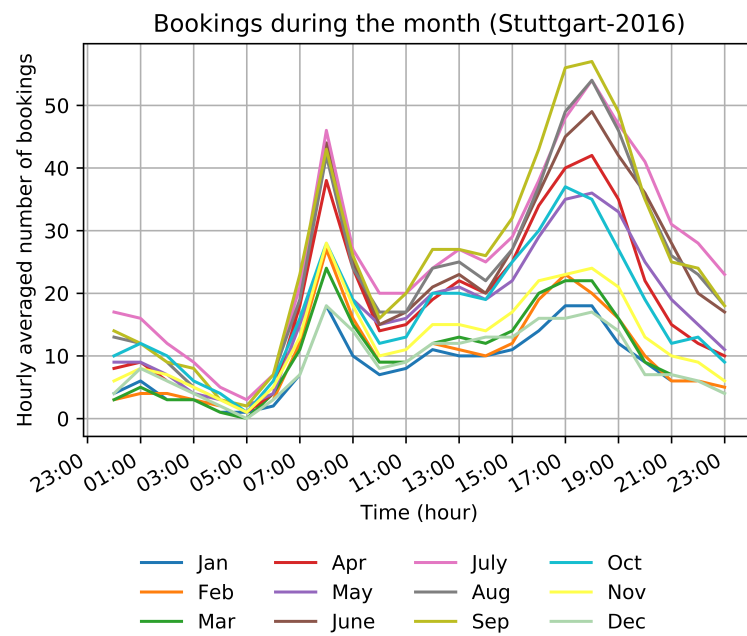


Figure A.12: Monthly booking trend of Stuttgart 2016

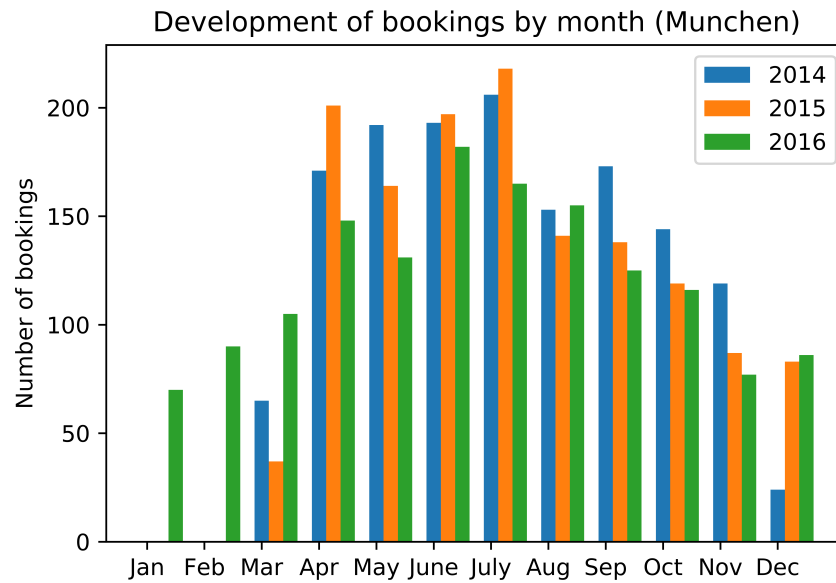


Figure A.13: Bookings by month of Munchen

Year	Jan	Feb	Mar	Apr	May	June	July	Aug	Sep	Oct	Nov	Dec
2014	0	0	65	171	192	193	206	153	173	144	119	24
2015	0	0	37	201	164	197	218	141	138	119	87	83
2016	70	90	105	148	131	182	165	155	125	116	77	86

Table A.11: Development of bookings by month (Munchen)

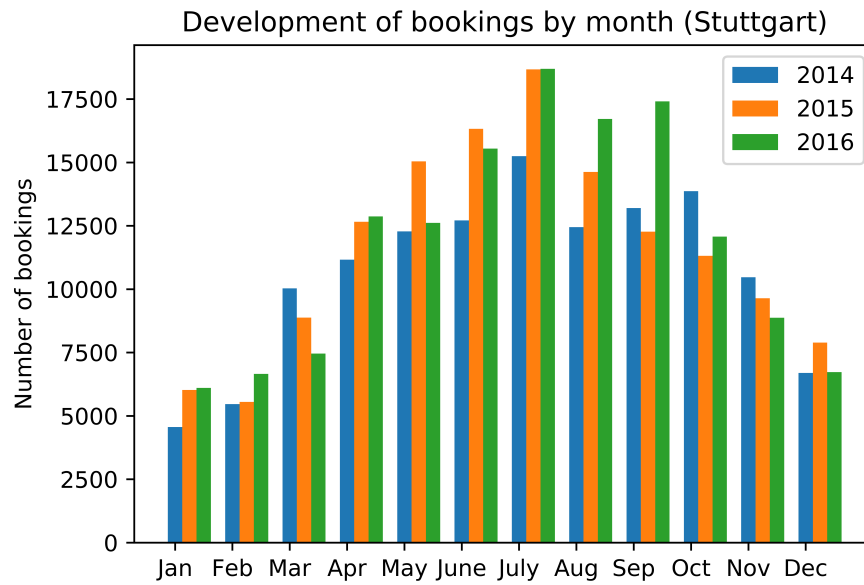


Figure A.14: Bookings by month of Stuttgart

Year	Jan	Feb	Mar	Apr	May	June	July	Aug	Sep	Oct	Nov	Dec
2014	4560	5463	10031	11165	12282	12714	15247	12445	13199	13869	10472	6696
2015	6025	5553	8877	12661	15042	16325	18670	14623	12270	11313	9640	7893
2016	6102	6658	7457	12868	12613	15548	18695	16717	17407	12074	8870	6726

Table A.12: Development of bookings by month (Stuttgart)

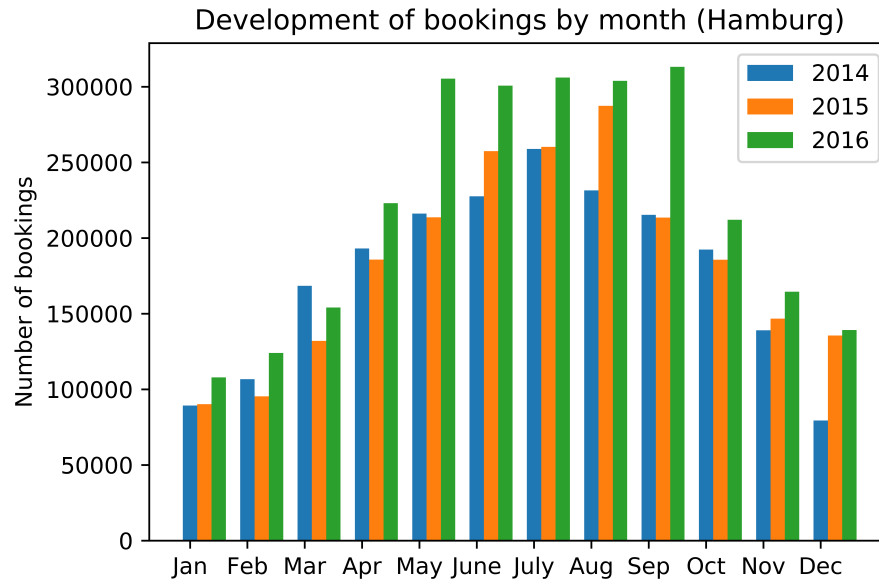


Figure A.15: Bookings by month of Hamburg

Year	Jan	Feb	Mar	Apr	May	June	July	Aug	Sep	Oct	Nov	Dec
2014	89321	106716	168416	193109	216160	227579	258914	231474	215303	192382	139060	79379
2015	90163	95380	132003	185725	213719	257482	260257	287363	213549	185638	146733	135556
2016	107888	124033	154055	223082	305324	300719	306071	303899	313203	212077	164493	139177

Table A.13: Development of bookings by month (Hamburg)

Year	Jan	Feb	Mar	Apr	May	June	July	Aug	Sep	Oct	Nov	Dec
2014	1449	8540	21299	23011	24171	27914	30785	28415	29696	29455	24922	14530
2015	14664	14058	25312	31346	35860	39080	46303	46965	35035	32960	33532	27633
2016	26179	32055	37469	65050	75544	78381	80474	69055	52407	34369	28885	20090

Table A.14: Development of bookings by month (Frankfurt)

## A.2. Temporal analysis

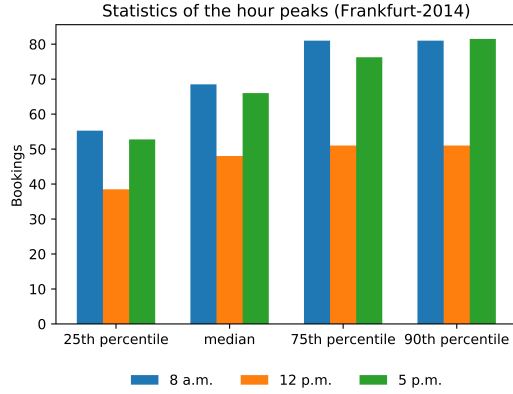


Figure A.16: Statistics of the peak hours of Frankfurt-2014

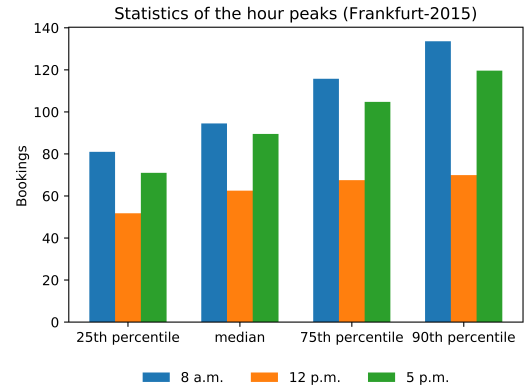


Figure A.17: Statistics of the peak hours of Frankfurt-2015

Hour	25th percentile	Median	75th percentile	90th percentile
8 a.m.	55.25	68.5	81.0	81.0
12 p.m.	38.5	48.0	51.0	51.0
5 p.m.	52.75	66.0	76.25	81.5

Table A.15: Statistics of the peak hours of Frankfurt-2014

Hour	25th percentile	Median	75th percentile	90th percentile
8 a.m.	81.0	94.5	115.75	133.6
12 p.m.	51.75	62.5	67.5	69.9
5 p.m.	71.0	89.5	104.75	119.6

Table A.16: Statistics of the peak hours of Frankfurt-2015

Hour	25th percentile	Median	75th percentile	90th percentile
8 a.m.	88.25	133.0	193.25	203.9
12 p.m.	57.5	80.5	121.75	134.8
5 p.m.	89.25	129.0	194.25	209.1

Table A.17: Statistics of the peak hours of Frankfurt-2016

## A.2. Temporal analysis

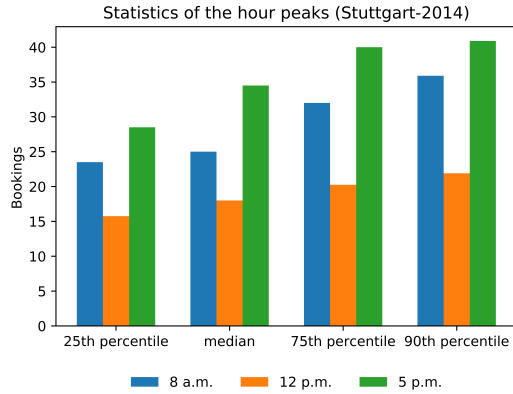


Figure A.18: Statistics of the peak hours of Stuttgart-2014

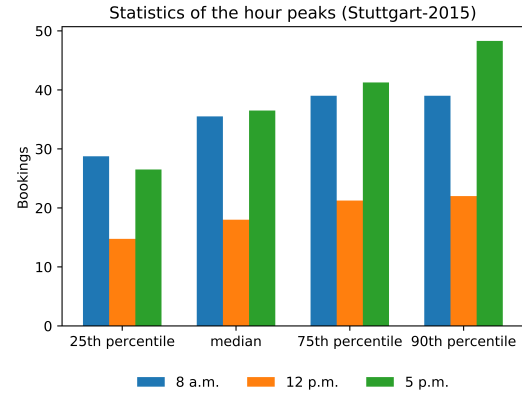


Figure A.19: Statistics of the peak hours of Stuttgart-2015

Hour	25th percentile	Median	75th percentile	90th percentile
8 a.m.	23.5	25.0	32.0	35.9
12 p.m.	15.75	18.0	20.25	21.9
5 p.m.	28.5	34.5	40.0	40.9

Table A.18: Statistics of the peak hours of Stuttgart-2014

Hour	25th percentile	Median	75th percentile	90th percentile
8 a.m.	28.75	35.5	39.0	39.0
12 p.m.	14.75	18.0	21.25	22.0
5 p.m.	26.5	36.5	41.25	48.3

Table A.19: Statistics of the peak hours of Stuttgart-2015

Hour	25th percentile	Median	75th percentile	90th percentile
8 a.m.	25.5	28.0	42.25	43.9
12 p.m.	12.0	19.5	21.75	24.0
5 p.m.	22.0	36.0	45.75	48.9

Table A.20: Statistics of the peak hours of Stuttgart-2016

## A.2. Temporal analysis

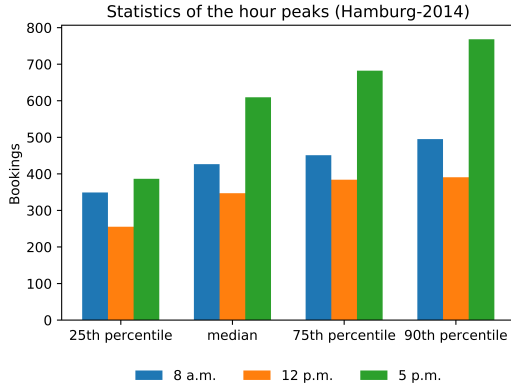


Figure A.20: Statistics of the peak hours of Hamburg-2014

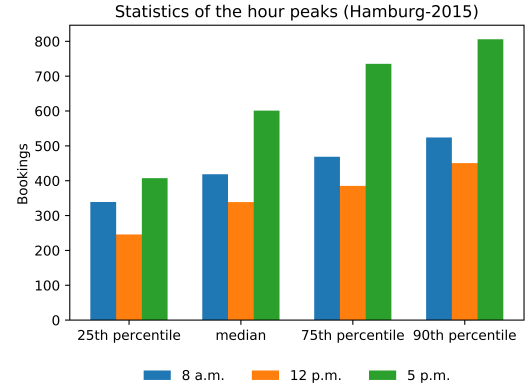


Figure A.21: Statistics of the peak hours of Hamburg-2015

Hour	25th percentile	Median	75th percentile	90th percentile
8 a.m.	349.0	426.5	451.0	495.1
12 p.m.	255.25	347.0	384.0	390.6
5 p.m.	386.5	609.5	682.25	768.2

Table A.21: Statistics of the peak hours of Hamburg-2014

Hour	25th percentile	Median	75th percentile	90th percentile
8 a.m.	338.75	418.5	468.75	524.1
12 p.m.	245.75	338.5	385.0	450.4
5 p.m.	407.25	601.0	735.25	805.9

Table A.22: Statistics of the peak hours of Hamburg-2015

Hour	25th percentile	Median	75th percentile	90th percentile
8 a.m.	390.75	512.5	581.0	670.1
12 p.m.	263.75	374.5	488.75	498.8
5 p.m.	462.5	698.5	969.25	1009.9

Table A.23: Statistics of the peak hours of Hamburg-2016



## A.3 Spatial analysis

Year	Number of bookings
2014	2046
2015	259
2016	1970

Table A.24: Distance analysis by cluster (Munchen)

Year	Number of bookings
2014	30802
2015	12880
2016	365088

Table A.25: Distance analysis by cluster (Stuttgart)

Year	Number of bookings
2014	93791
2015	64449
2016	1088653

Table A.26: Distance analysis by cluster (Frankfurt)

Year	Number of bookings
2014	679209
2015	236549
2016	6059644

Table A.27: Distance analysis by cluster (Hamburg)

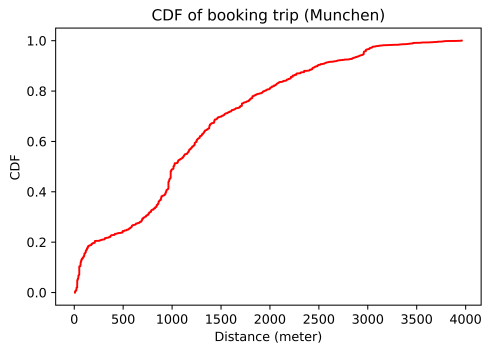


Figure A.22: CDF of booking trip of Munchen

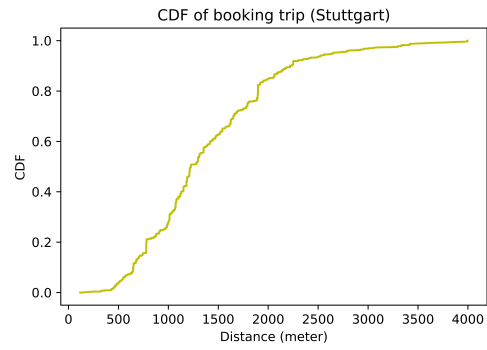


Figure A.23: CDF of booking trip of Stuttgart

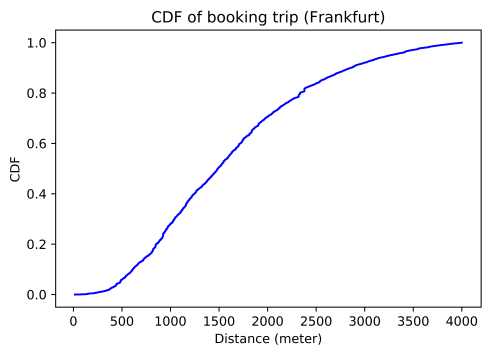


Figure A.24: CDF of booking trip of Frankfurt

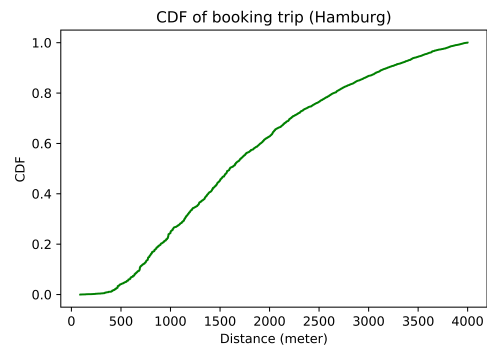


Figure A.25: CDF of booking trip of Hamburg

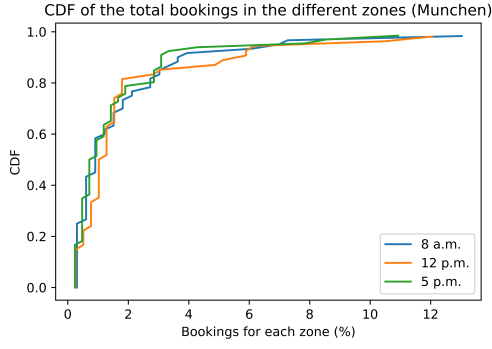


Figure A.26: CDF of total bookings for each zone of Munchen

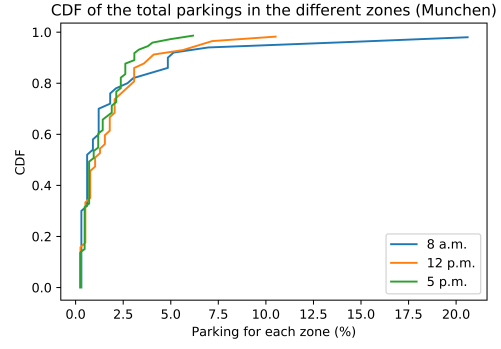


Figure A.27: CDF of total parkings for each zone of Munchen

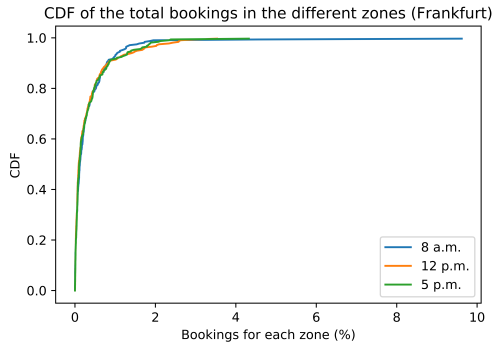


Figure A.28: CDF of total bookings for each zone of Stuttgart

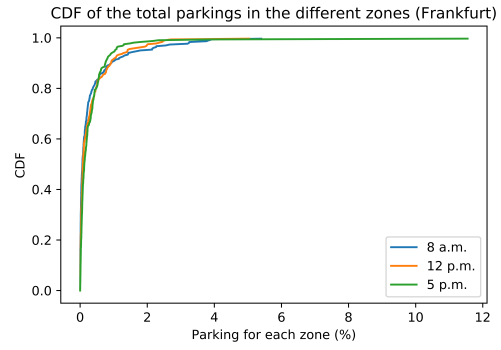


Figure A.29: CDF of total parkings for each zone of Stuttgart

### A.3. Spatial analysis

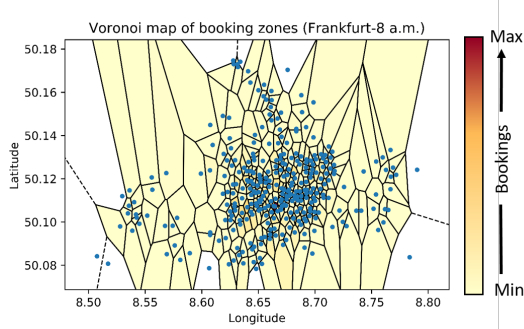


Figure A.30: Voronoi map of bookings  
(Frankfurt-8 a.m.)

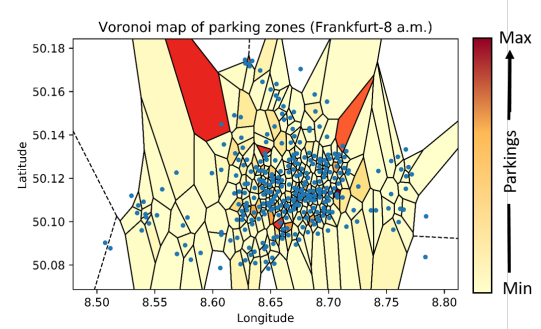


Figure A.31: Voronoi map of parkings  
(Frankfurt-8 a.m.)

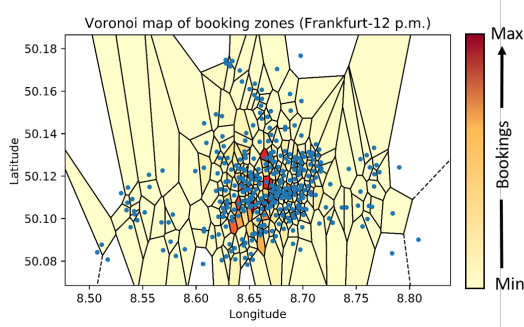


Figure A.32: Voronoi map of bookings  
(Frankfurt-12 p.m.)

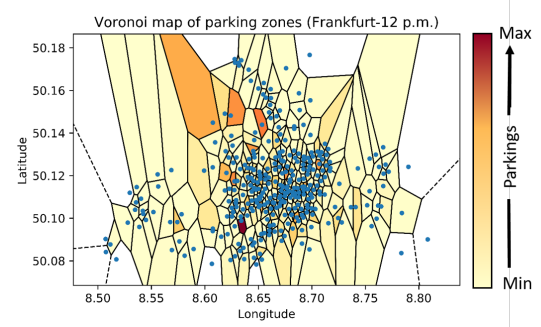


Figure A.33: Voronoi map of parkings  
(Frankfurt-12 p.m.)

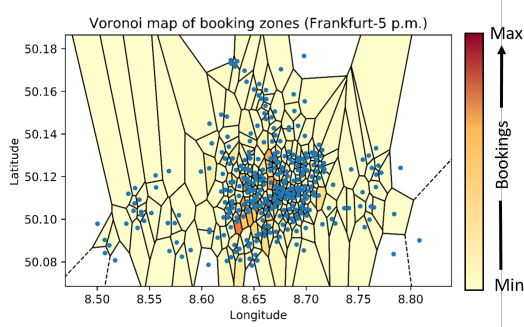


Figure A.34: Voronoi map of bookings  
(Frankfurt-5 p.m.)

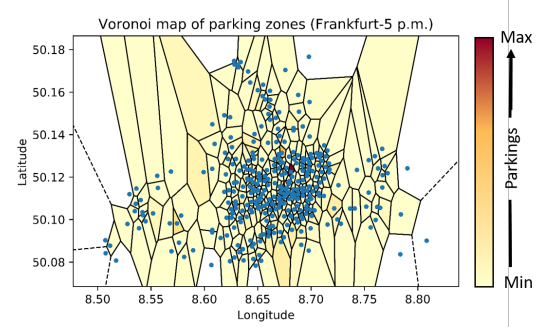


Figure A.35: Voronoi map of parkings  
(Frankfurt-5 p.m.)

### A.3. Spatial analysis

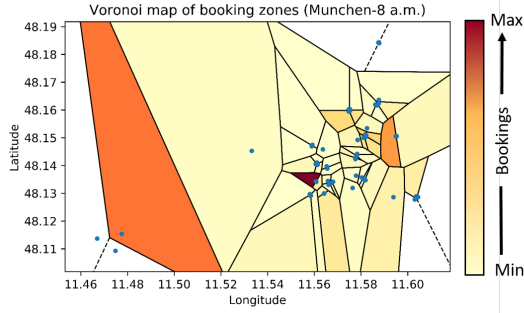


Figure A.36: Voronoi map of bookings  
(Munchen-8 a.m.)

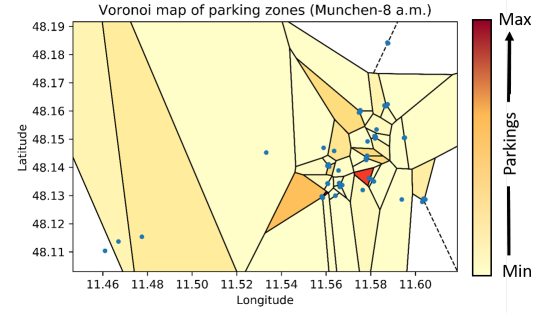


Figure A.37: Voronoi map of parkings  
(Munchen-8 a.m.)

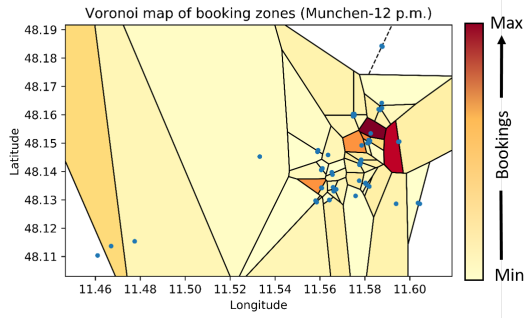


Figure A.38: Voronoi map of bookings  
(Munchen-12 p.m.)

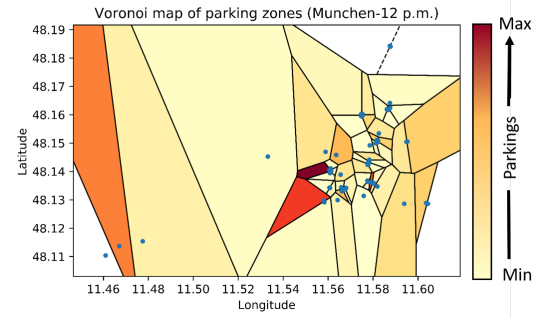


Figure A.39: Voronoi map of parkings  
(Munchen-12 p.m.)

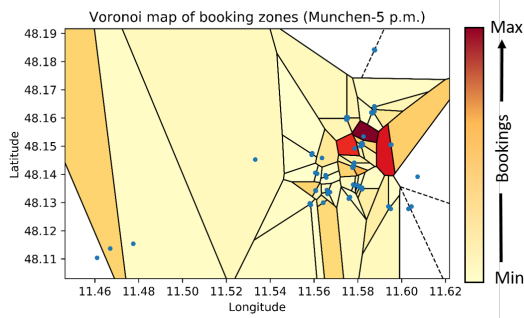


Figure A.40: Voronoi map of bookings  
(Munchen-5 p.m.)

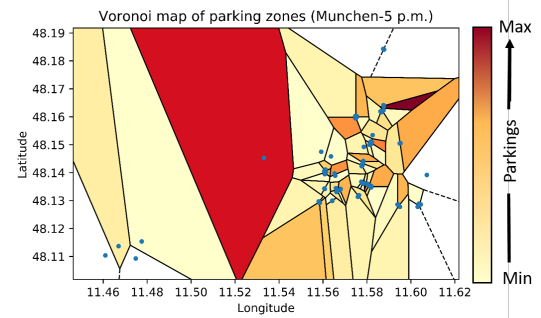


Figure A.41: Voronoi map of parkings  
(Munchen-5 p.m.)

### A.3. Spatial analysis

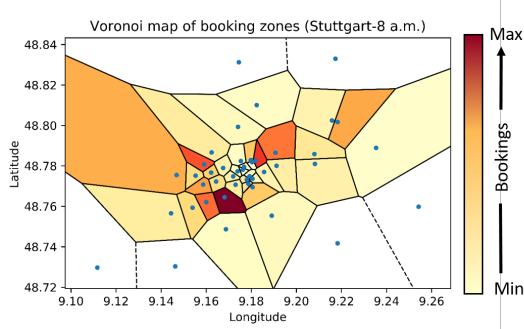


Figure A.42: Voronoi map of bookings  
(Stuttgart-8 a.m.)

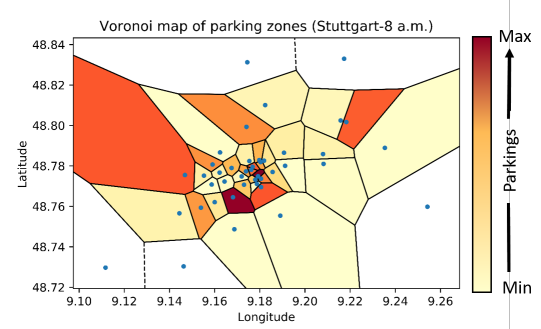


Figure A.43: Voronoi map of parkings  
(Stuttgart-8 a.m.)

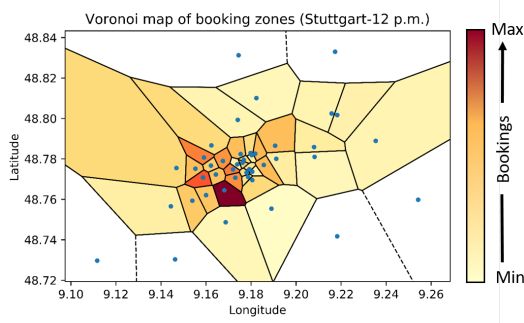


Figure A.44: Voronoi map of bookings  
(Stuttgart-12 p.m.)

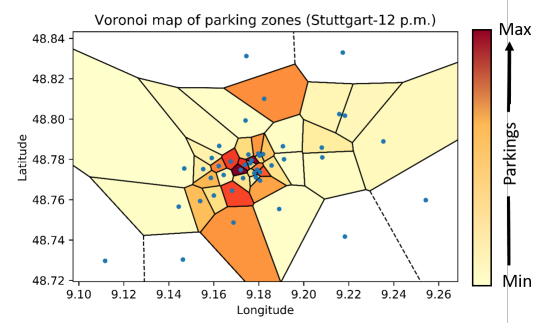


Figure A.45: Voronoi map of parkings  
(Stuttgart-12 p.m.)

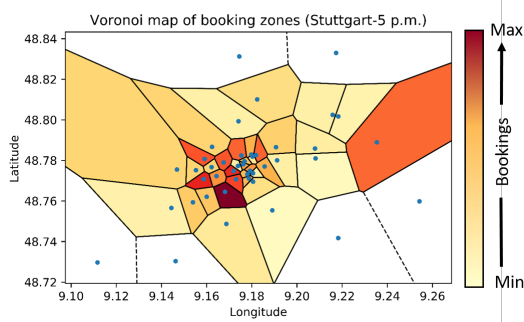


Figure A.46: Voronoi map of bookings  
(Stuttgart-5 p.m.)

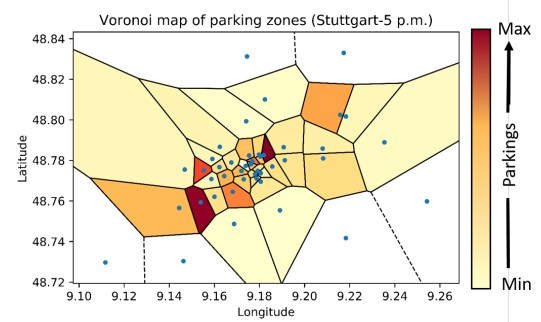


Figure A.47: Voronoi map of parkings  
(Stuttgart-5 p.m.)

### A.3. Spatial analysis

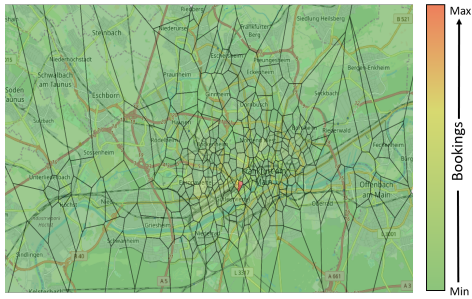


Figure A.48: Voronoi map of booking zones (Frankfurt-8 a.m.)

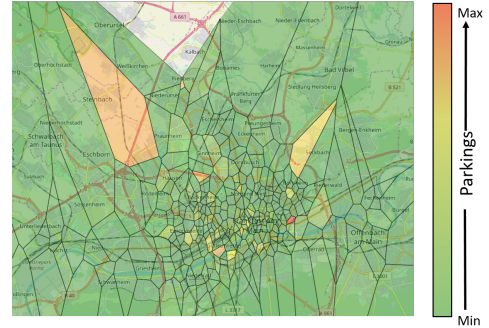


Figure A.49: Voronoi map of parking zones (Frankfurt-8 a.m.)

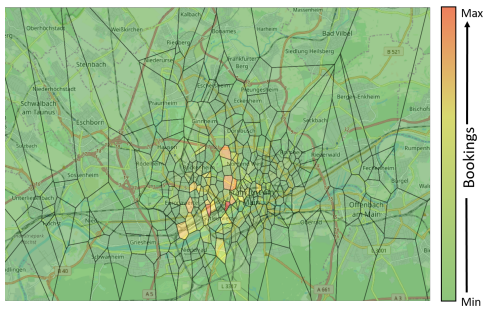


Figure A.50: Voronoi map of booking zones (Frankfurt-12 p.m.)

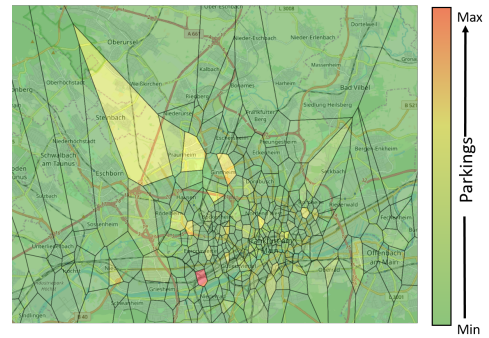


Figure A.51: Voronoi map of parking zones (Frankfurt-12 p.m.)

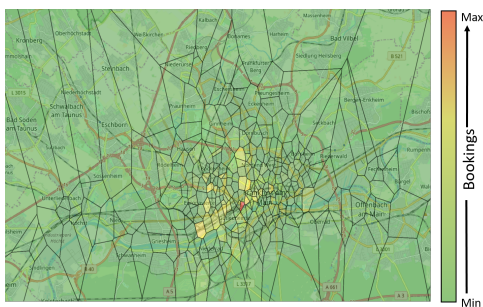


Figure A.52: Voronoi map of booking zones (Frankfurt-5 p.m.)

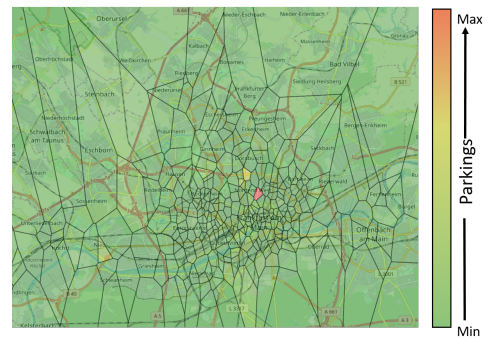


Figure A.53: Voronoi map of parking zones (Frankfurt-5 p.m.)



### A.3. Spatial analysis



Figure A.54: Voronoi map of booking zones (Munich-8 a.m.)

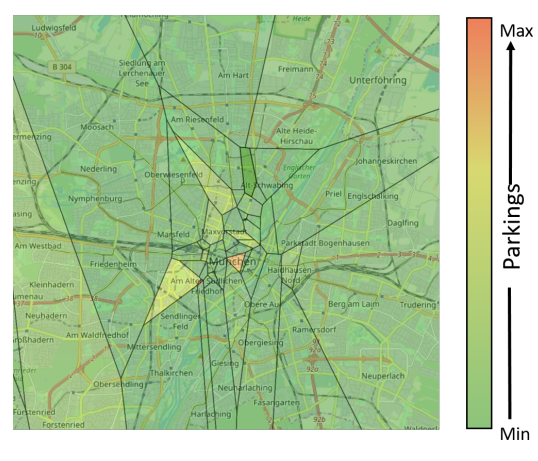


Figure A.55: Voronoi map of parking zones (Munich-8 a.m.)

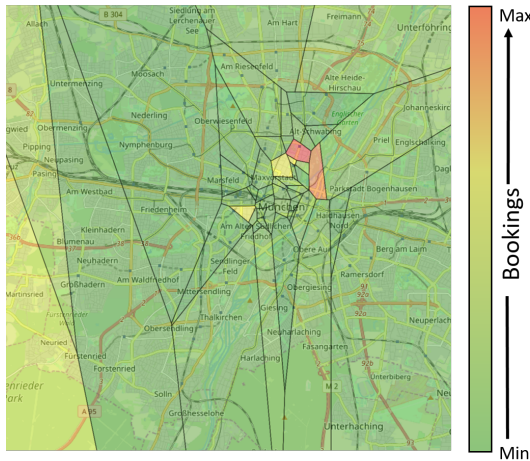


Figure A.56: Voronoi map of booking zones (Munich-12 p.m.)

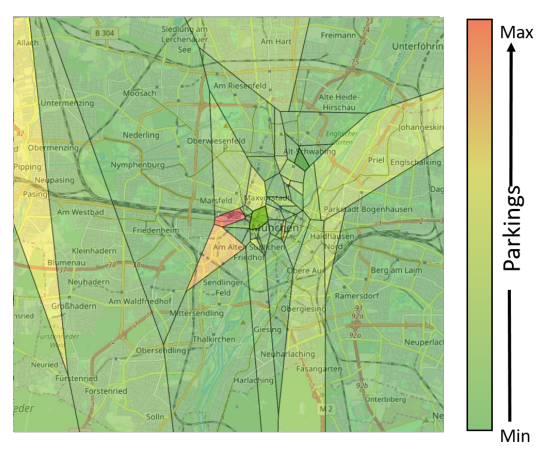


Figure A.57: Voronoi map of parking zones (Munich-12 p.m.)



### A.3. Spatial analysis



Figure A.58: Voronoi map of booking zones (Munich-5 p.m.)



Figure A.59: Voronoi map of parking zones (Munich-5 p.m.)

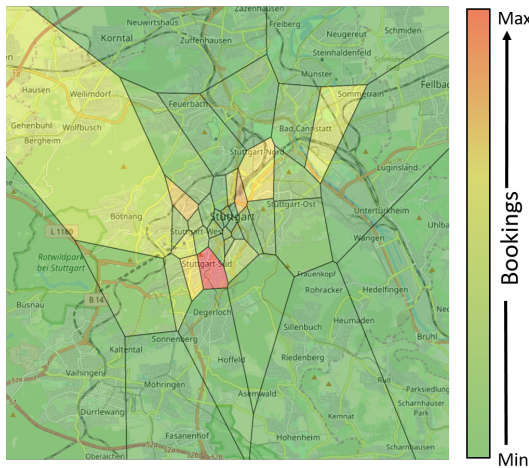


Figure A.60: Voronoi map of booking zones (Stuttgart-8 a.m.)

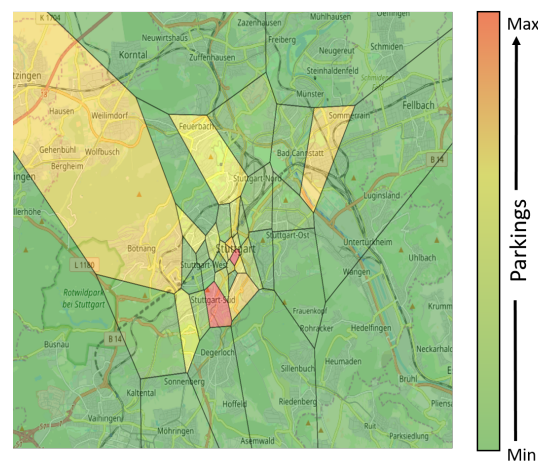


Figure A.61: Voronoi map of parking zones (Stuttgart-8 a.m.)

### A.3. Spatial analysis

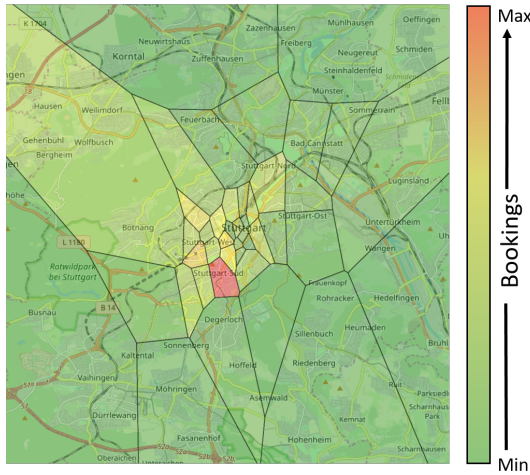


Figure A.62: Voronoi map of booking zones (Stuttgart-12 p.m.)

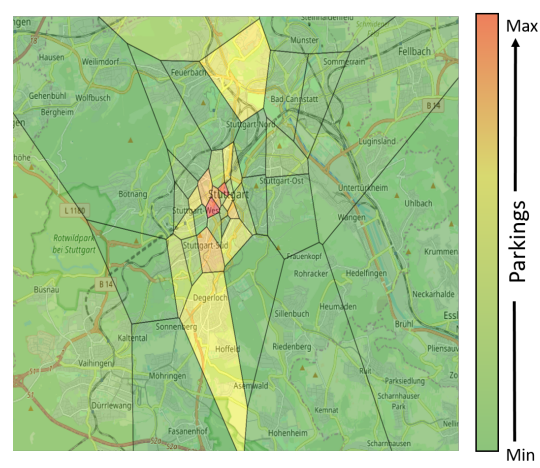


Figure A.63: Voronoi map of parking zones (Stuttgart-12 p.m.)

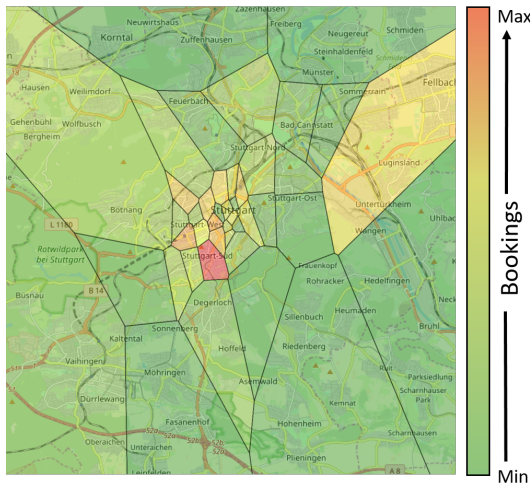


Figure A.64: Voronoi map of booking zones (Stuttgart-5 p.m.)

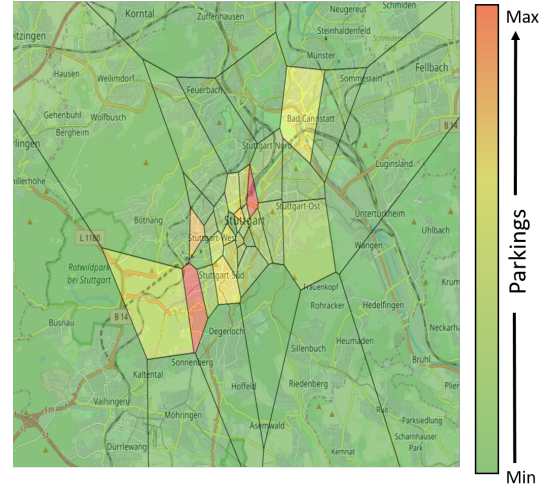


Figure A.65: Voronoi map of parking zones (Stuttgart-5 p.m.)

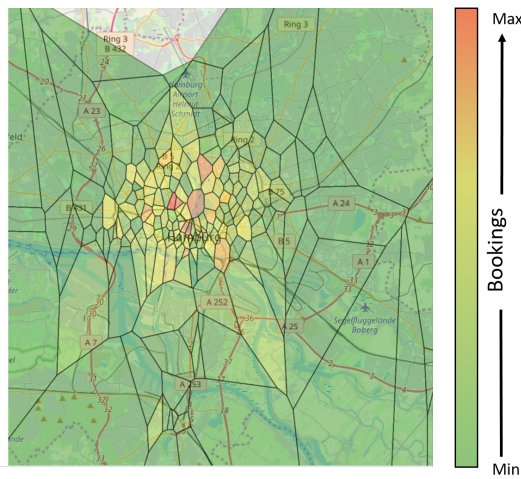


Figure A.66: Voronoi map of booking zones (Hamburg-12 p.m.)



Figure A.67: Voronoi map of parking zones (Hamburg-12 p.m.)

# Bibliography

- [1] Ofo website. Available at <https://www.ofo.com/>. [Accessed 30 September 2018]
- [2] Mobike website. Available at: <https://mobike.com/it/>. [Accessed 3 July 2018]
- [3] Deutsche Bahn website (German website). Available at: <https://www.bahn.de/p/view/index.shtml>. [Accessed 4 October 2018]
- [4] Deutsche Bahn dataset: Call A Bike. Available at: <https://data.deutschebahn.com/dataset/data-call-a-bike>. [Accessed 2 October 2018]
- [5] Professor Ralph Buehler , Virginia Tech. Technical report: *D.C. Dockless Bike-share: A First Look*, Spring 2018.
- [6] Chen Mengwei and Wang Dianhai and Sun Yilin and Waygood E. Owen D. and Yang Wentao, *A comparison of users' characteristics between station-based bike-sharing system and free-floating bikesharing system: case study in Hangzhou, China*, journal: "Transportation", August 2018.
- [7] Regue, R., Recker, W.: Proactive vehicle routing with inferred demand to solve the bikesharing rebalancing problem. *Transp. Res. Part E* 72, 192-209 (2014)
- [8] Bao, J., et al.: Exploring bikesharing travel patterns and trip purposes using smart card data and online point of interests. *Netw. Spat. Econ.* 4(17), 1231-1253 (2017)
- [9] Shaheen, S., Guzman, S., Zhang, H.: Bikesharing in Europe, the Americas, and Asia. *Transp. Res. Rec. J. Transp. Res. Board* (2143), 159-167 (2011)



- [10] Wu Jiansheng and Wang Luyi and Li Weifeng, *Usage Patterns and Impact Factors of Public Bicycle Systems: Comparison between City Center and Suburban District in Shenzhen*, Journal of Urban Planning and Development, 2018.
- [11] S. Reiss and K. Bogenberger, *Optimal bike fleet management by smart relocation methods: Combining an operator-based with an user-based relocation strategy*, 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), November 2016.
- [12] Pal Aritra, Zhang Yu and Kwon Changhyun, *Analysis of Free-floating Bike Sharing and Insights on System Operations or Analyzing Mobility Patterns and Imbalance of Free Floating Bike Sharing Systems*, University of South Florida, Tampa, January 2018.
- [13] Yung-lung Liu and W. Y. Szeto and Sin C. Ho, *A static free-floating bike repositioning problem with multiple heterogeneous vehicles, multiple depots, and multiple visits*, 2018.
- [14] Nadir Nibras, *Predicting number of Bike-share Users - A classic regression problem solved with Supervised Machine Learning*, November 2018.
- [15] GitHub page of bike sharing platforms. Documentation of Bike Sharing APIs. Available at: <https://github.com/ubahnverleih/WoBike>. [Accessed 2 July 2018]
- [16] Ofo coverage map in Milan. Available at: <https://www.facebook.com/ofoitalia/>. [Accessed 28 July 2018]
- [17] Python.org (2018). Welcome to Python.org. Available at: <https://www.python.org/>. [Accessed 1 July 2018]
- [18] Pandas.pydata.org. Python Data Analysis Library. Available at: <https://pandas.pydata.org/>. [Accessed 3 October 2018]