Czech Technical University in Prague
Faculty of Electrical Engineering

Department of Computer Science

# DIPLOMA THESIS AGREEMENT

Student: Chmelař Matej

Study programme: Open Informatics
Specialisation: Artificial Intelligence

Title of Diploma Thesis: Data-Driven Model of Taxi Passenger Demand

Guidelines:

The objectives of this thesis are:

1) Study state-of-the-art methods of passenger demand modelling.
2) Select and examine a relevant available dataset.
3) Design and implement a data-driven taxi passenger demand model based on the selected dataset. Focus on a selection and design of appropriate features as well as on selection of machine learning techniques and their hyperparameters.
4) Consider utilization of related available datasets to increase precision of the model, e.g., weather data, cultural/sport events, etc.
5) Experiment with the model limited to spatially-invariant features. Examine transferability of the model to a different location (e.g., different city or a part of the city).
6) Evaluate your models using the selected dataset.

Bibliography/Sources:


[1] de Brébisson, Alexandre, et al. "Artificial neural networks applied to taxi destination prediction." arXiv preprint arXiv:1508.00021 (2015).
[2] Xu, Jianmin, et al. "Modeling level of urban taxi services using neural network." Journal of Transportation Engineering 125.3 (1999): 216-223.
[3] Phithakkitnukoon, Santi, et al. "Taxi-aware map: Identifying and predicting vacant taxis in the city." International Joint Conference on Ambient Intelligence. Springer Berlin Heidelberg, 2010.
[4] Friedman, Jerome, Trevor Hastie, and Robert Tibshirani. The elements of statistical learning. Vol. 1. Springer, Berlin: Springer series in statistics, 2001.


Diploma Thesis Supervisor: Ing. Jan Drchal, Ph.D.

Valid until the end of the winter semester of academic year 2018/2019


prof. Dr. Michal Pěchouček, MSc.
Head of Department

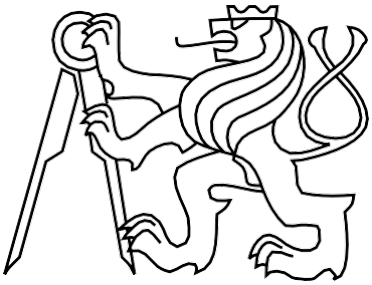
prof. Ing. Pavel Ripka,CSc.
Dean

Prague, Jun 23, 2017

Czech Technical University in Prague

Faculty of Electrical Engineering

Department of Computer Science



Master's Thesis

# Data-Driven Model of Taxi Passenger Demand

*Bc. Matej Chmelař*

Supervisor: Ing. Ján Drchal Ph.D.

Study Programme: Open Informatics

Field of Study: Artificial Intelligence

January 2018

## Acknowledgements

I would like to thank my supervisor, Ing. Ján Drchal Ph.D. for his goodwill, approach, and endless patience. More-over I would like to thank my parents and friends, for their unconditional support during my whole study.

## Declaration

I hereby declare that I have completed this thesis independently and that I have listed all the literature and publications used within the research and work. I have no objection to usage of this work in compliance with the Act §60 Law No. 121/2000Sb. (copyright law), and with the rights connected with the copyright act including the changes in the act.

In Prague …… ……..

# Abstract

This thesis introduces an overview of methods designed for modelling and prediction of taxi demand, using information available from datasets, consisting mostly of GPS localization, and timestamps. The objective is to develop a program, which manages to create data-driven model, able to predict passenger taxi demand. This is accomplished by selection of appropriate features, working with selection of data and configuring parameters of said model. Result of this thesis is model able to slightly predict the pickup location – latitude and longitude. Experiments show how well created model works, with comparison between single approaches.

Keywords: random forest, regression, machine learning, prediction, data-driven model, passenger, taxi, demand

# Abstrakt

Táto práca uvádza prehľad metód určených na modelovanie a predikciu taxi dopytu, s využitím informácii dostupných z dát, pozostávajúcich hlavne z GPS lokalizácie, a časových značiek. Úlohou je vyvinúť program, ktorý dokáže vytvoriť model založený na dátach, schopný predpovedať taxi dopyt cestujúceho. Toto je dosiahnuté selekciou vhodných vlastností, prácou s kolekciou dát, a konfiguráciou parametrov daného modelu. Výsledkom tejto práce je model schopný aspoň trochu predikovať polohu vyzdvihnutia – zemepisná dĺžka a šírka . Experimenty ukazujú ako schopne vytvorený model funguje, v porovnaní medzi jednotlivými metódami.

Keywords: random forest, regresia, strojové učenie, predikcia, model založený na dátach, cestujúci, taxi, dopyt

# Content

# List of Figures

# List of Tables

# Abbreviations

| | |
|---|---|
| NYC | New York City |
| dow | day of the week |
| hod | hour of the day |

# Introduction

Taxi service is a method commonly used in a central transportation, in almost every urban area. Nearly every mode of transportation uses GPS localization. This involves also other similar types of services, like Uber, Liftago, Lyft, etc. (will be called taxis in this paper). For all of these taxis which have enabled GPS, we are able to collect the geo-spatial location data for every trip. This means we can see time and location of the pickup, place where passenger entered the vehicle, and also the time and location of the drop off, place where passenger left the vehicle. This is something that gives us rich data which can be used to have some kind of insight on taxi passenger demand, meaning ability to see some pattern in time, mobility, or perhaps more significant places than the others.

The objective of every taxi company is to maximize number of customers delivered from one place to another every day. This means they need to plan the route of every taxi driver, so they spent as less time with a vacant space in the vehicle, as it is possible. Even though every experienced driver can estimate, where the next passenger could be found, as he usually goes the same places every day, he might not know where new passengers could be found in unknown locations. Also, he might not know of other irregular passenger demands because of some unusual features. But of course, there are new starting taxi drivers, or even companies, which would be able to use this knowledge to help them find new customers more efficiently. Many taxi drivers already use smartphones to aid them in the battles for customers, they track their own locations so when customer starts the application, they are able to see where some taxi drivers are. This helps them manage their plans so they can get the nearest cab in the most comfortable time. Other than that, it automatically calculates the fare considered if they input targeted destination, in addition to their starting location. Some applications have linked payment methods with taxi services, or if they wish, they can pay in person. All these uses of smartphones allow us to track every pickup locations and drop-offs, or even if it is needed, the path as whole can be tracked in some interval, which would result in several points the taxi driver travelled through. The tracking would not be whole without information about time when driver picked up the customer, total travelled distance, time of arrival to customers ordered location, and many more information about the trip the taxi driver made.

Considering, the objective is to decrease the total time without passengers, the important aspect of this is the ability to predict where the next passenger could be. This means that this problem is suited for a machine learning approach. In case of it being somewhat successful,

having at least some kind of additional information about passenger demand, at current time, over some spatial location, could aid the taxi services and improve the possibility of achieving their objective, ideally, predicting the starting locations (pickup latitude and longitude) of taxi trips based on all important information available to the taxi driver. However, to be able to predict this, we need to understand what the common citizen is going to do, and where is he going to be. But, there can also be some patterns that human is unable to notice or interpret, though, can be found by use of machine learning.

In this thesis, a machine learning model is constructed to estimate the taxi demand, more precisely, close pick up location (latitude and longitude). It is trained with historical data openly available, which consists of many information about the trip of the taxi. It usually has fields where we can see following information, which will be later more thoroughly looked into:

- the ID of Taxi,
- pickup datetime – the date and time when the meter was engaged,
- drop-off datetime – the date and time when the meter was disengaged,
- passenger count – number of passengers in the vehicle
- trip distance – elapsed trip distance in miles (for data available in NYC)
- pickup longitude and latitude – location where meter was engaged
- drop-off longitude and latitude – location where meter was disengaged
- fare amount – time and distance fare calculated by the meter

# 2. Formulation of task

Main objectives of this thesis are following:

1. Study state-of-the-art methods of passenger demand modelling.
2. Select and examine a relevant available dataset
3. Design and implement a data-driven taxi passenger demand model based on the selected dataset. Focus on a selection and design of appropriate features as well as on selection of machine learning techniques and their hyperparameters.
4. Consider utilization of related available datasets to increase precision of the model, e.g. weather data, cultural/sport events, etc.
5. Experiment with the model limited to spatial-invariant features. Examine transferability of the model to a different location (e.g., different city or a part of the city).
6. Evaluate your models using the selected datasets.

First part of this thesis will consist of overview of basic concepts, with related literature. This should provide some insight about what does this problem mean, how is it currently being solved, and other relevant information.

Second part will consist of information about design and implementation of data-driven model, which should be able to receive and process correct data, afterwards train itself on said input, and then be possibly able to predict some relevant features.

Last part consists of experiments in which I use implemented data-driven model. I will talk about what methods are used, how the dataset is tested, and what results are achieved. Finally, I will summarize the work in the conclusion part.

# 3. Background and related work overview

## 3.1. Basic overview

The transport industry is one of the greatest industries of the present day. In any developing city, road transport is an important factor [1]. As technology advances and as the world economy takes shape, it has become imperative to come up with workable solutions of transporting goods and people from one place to another. There are many decisions that can be made along transportation of goods and people. In this paper, the researcher is interested in modelling a data driven technique of estimating taxi customer demand. According to Aarhaug [2], a taxi is a vehicle with a designated driver for hire. There has been a challenge in determining the number of people that require taxi transportation in a given period of time.

In the taxi industry, all players are experiencing the challenge of determining the number of commuter to expect at a given point in time. When the number of commuters expected is higher than the real number, taxi owners are forced to go without commuters. The resulting decision in this case is reduction in the number of taxis that ply a certain route. When this happens, there is a likelihood that the number of taxis is going to be reduced in response to the shrank demand. This reduction may result into negative implications like commuters staying in long queues without getting taxi services.

In this thesis, the goal is to solve the current situation being experienced by commuters and taxi owners. Also according to Aarhaug [2], town commuters face so many challenges related to mobility. This explains why the taxi system was invented. In many towns globally, there is too much congestion occasioned by the current trend of rural-urban migration. As a result of this and many other challenges, it has been established that there is need for a convenient mobility solution. Stakeholder expectations for this project are cost effectiveness, convenience, safety, security and comfort. Every commuter using taxi service will expect to get all the above expectations in a single offer. In order to attain these requirements, the researcher will have to come up with a detailed model of determining taxi customer demand.

## 3.2. Data Driven Models

The main task in this project is to come up with a data driven model design that will address beforementioned problem. The expectation here is that all customers using taxi will get the expected satisfaction or what in business terms would be referred to as “value for money”. In order to serve customers as required, taxi owners should be able to predict the expected number of customers at any point in time. This is to say that taxi owners should be able to forecast any change in number of customers in order to make decisions that will keep their clients satisfied at all times. The objective of this study is to come up with several models that can be applied by taxi owners and other stakeholders in order to predict the number of customers that will be expected in any period of time.

It is impossible to forecast demand without looking at historical data. According to Armstrong & Green [3], demand, as the greatest determinant of utility, is forecasted using historical data. Before getting deeper into techniques of predicting customer requirements, it is important to start by recommending that taxi owners should be in a position to make utility based decisions as and when they are required. These decisions include variation in number and type of taxis and different times. If taxi owners are not in a position to do this, coming up with a customer prediction model may not be the best thing to start with. In order to ensure reachability, taxis can be offered online or through an app which can have iOS and Android support. These are the main mobile platforms that are used by customers.

The most important requirement in taxi industry is determination of stations and number of vehicles to be available in a station at any given time. According to research by Cho [4], one of the main determinants of demand is time.

## 3.3. Prediction

The goal of this paper as it has been stated on several occasions is to predict taxi customer demand in order to determine supply requirements. There are several dimensions that will be used to approach this technicality. The main goal is to ensure that any customer getting to a station will get ether an available vehicle for commuting. The technical issue here is to determine the most appropriate number of vehicles/parking spaces to be reserved in a station at any given point in time, or be able to determine any other places for expected said passenger demand at any given point in time. The first dimension that will be applied in making this determination is the strategic approach. Here it is required that proper forecasting will be done to establish demand at any given point.

In order to predict demand in a strategic manner, there are several sets of data that should be collected. To begin with, there is a robust system of collecting data about different market variables like time of the year, season, etc. When this data s collected, it should be matched with the respective numbers of taxi customers for those times. Strategically, one can look at variations in numbers of customers for different seasons of the year, different day of the week, or most importantly, time of the day and come up with decisions that are going to respond to them. The main assumption here is that every time is like the other. This is to say that what happens in a certain season, day, or time of the day, will also happen in another future point in time. Solomatine, See, and Abrah [5]s state that strategic prediction works by modelling real time data. The main control tool for this approach is to ensure that data collected is free from subjectivity and bias.

In theory, in case of equality of arrival rate of vacant taxis, and rate of passenger demand appearance, total equilibrium will be achieved. This means every time, passenger would be of need of taxi, the very same moment vacant taxi would arrive to that exact location, at the needed given time. [6] But in reality, most often disequilibrium states occur, which are either oversupply, that means there are more vacant taxis than are currently people in demand of taxi transportation, or the opposite situation can appear, overfull demand, which indicates that there are many waiting passengers, and not enough of vacant taxis to transfer them to their desired destinations. So obviously, both event are highly undesirable, as it either makes taxi service companies waste their resources on excess of vacant taxis, or causes people to be unhappy, and spent their time waiting. This raises a need for at least some kind of forecasting of short-term passenger demand. [6] This ability to predict passenger demand is quite difficult, mostly because of three kind of dependencies mentioned by Zhang [7]:

1. Spatial dependency: passenger demand in one specific zone was not determined by only the variables of this zone, but also by the zonal variables in the network. There is also significant difference between nearby zones, and locations further away. [8]
2. Time dependency: every passenger demand is strongly repeating occurrence. For instance, rush hours in the morning, may affect use of taxi service. Then also, higher passenger demand could be foreseen in later hours, or even night hours.
3. External dependency: some other factors may have some kind of effect on the passenger demand. To name some, such as, weather, sport events, airport travels.

In definition, short-term passenger demand is time-series prediction problem. According to this research [6], nearest historical passenger demand, could have the most influence on predicting demand in early time. This means, that the travel time of trips affects the forecast of the taxi demand, considering it effect on congestion level of trips and nearby locations. Then obviously, time of the day, day of the week, and weather reflects the short-term passenger demand.

## 3.4. Machine Learning

Just like humans, machine can be taught to operate on themselves. Solomatine, See and Abrah [5] refer to the concept as artificial intelligence. AI as it is commonly referred to, is a form of learning where machines are programmed to collect real time data, analyse it and make decisions based on it. In the taxi industry, it has already been stated that the biggest challenge is to predict the number of people that will be willing to commute in a given period of time. If such a decision is not made adequately, it might result to wastage on the part of taxi owners or inconvenience on the part of commuters. Wastage for taxi owners arises when they are not able to get customers. There are a number of fixed costs that are associated with taxis, these include salaries and depreciation. Whether there are customers for taxis or not, taxi owners must pay these costs.

In application of machine learning, cars used to provide taxi services can be programmed in a manner that they will be collecting usage data on real time basis. Using modern technologies like GPS, it is also possible to ensure that location data is collected on real time basis. Using these two approaches, it is possible to determine the number of trips made by a taxi as well as the locations where these trips are made. If this is done repeatedly, it becomes possible for the intelligent unit of the taxi to determine the relationship between number of trips, location and other variables like time. Time in this case can refer to season or time of the day. Basically, the main idea is to come up with a decision on number of people to expect in a certain location considering the time of the day or day of the week.

Application of artificial intelligence is very important as it reduces cost of employing complex research methodologies. Apart from reduction of cost, AI increases reliability as it is not subjected to biasness or what is commonly referred to as “human error”. In this way, it is possible to make decisions about the number of taxi cars that will be required to meet demand. When these decisions are made from an informed point, there is maximization of income for taxi owners as well as value for the commuters.

## 3.5. Random Forests

This method is part of ensemble methods, which uses decision trees. As Breiman [9] is developer of Random Forest, his insight is definitely most valuable. According to him, "random forests for regression are formed by growing trees depending on a random vector Θ such that the tree predictor $h(\mathbf{x}, \Theta)$ takes on numerical values as opposed to class labels. The output values are numerical and we assume that the training set is independently draw from the distribution of the random vector $Y,\mathbf{X}$. The mean-squared generalization error for any numerical predictor $h(\mathbf{x})$ is

$$E_{\boldsymbol{X},Y}\big(Y - h(\mathbf{X})\big)^2$$

The random forest predictor is formed by taking the average over k of the trees of the trees $\{h(\mathbf{x}, \Theta_k)\}$." [9] This method is really suitable for any task where there is huge number of data available, and also higher number of features or attributes is present.

In random forest method, we can change some of the parameters. To name few, there is option to change number of estimators, which is basically number of "trees in the forest". Often, the higher number, the better and overall more precise results are achieved, but after some point, the difference is not significantly changed, and only increases time needed for the model training part. [10]

There is also relation between, strength of the individual trees in the forest, difference in between pair of trees in the forest, and error. [10] To simply say it, random forests operate with the bagging method, to produce random samples out of the training sets for every tree. When the tree is built, it uses new subset, and most importantly, random attribute selection. Next, the best split is used on the node.

Important note is, that random forests do not overtrain [9]. But in case of large datasets, all of the data is used, even though not all of them are equally important for learning.

## 3.6. Related work

### 3.6.1. Analysis of the passenger pick-up pattern for taxi location recommendation

This work deals with evaluation of pick-up patterns of taxi services based on real-life location history data, from Republic of Korea. The point is to collect effective background data which are needed to design location recommendation service for empty taxi cars. It is said that the useful analysis is the pick-up pattern classification which is related to income of the taxi business. This could be improved by reducing the amount of empty taxi cars by navigating them to spots where many customers could be possibly waiting for a taxi. It is reported that up to 80% times there is no customer present in the taxi.

The analysis starts with creating suitable pick-up records from the raw history data. It contains huge amount of widely scattered data which was collected for a long time. Spatial grouping is used because of differences and uniqueness of areas. Then, make a refined cluster for a location recommendation. They propose to use refined clustering performed by k-means method. Also, within each cluster, temporal analysis used creates time dependency pattern changing along the time axis.

Their cluster and spatial-temporal pick-up frequency helped to suggest the empty taxi cars to go to the nearby cluster locations which results in the reduction of the empty taxi ratio, and improving income of the taxi business. [11]

### 3.6.2. A predictive model for the passenger demand on a taxi network

In this work, they focused on the problem of choosing which taxi is the best to drive after a passenger in a given location and time, in the city Porto, Portugal. They bring up four different key factors: expected price for the service, distance or cost relation with each stand, number of taxis available in each stand, and passenger demand for each stand.

Their goal is to predict how many services will be demanded during time period for each taxi stand, reusing real service extracted from the data. Each data has following attributes: type of event, taxi stand, timestamp, ID of taxi, latitude and longitude retrieved from to the GPS. [12]

### 3.6.3. Artificial Neural Networks Applied to Taxi Destination Prediction

Their approach uses fully automated and is based on artificial neural network, a variant of multi-layer perceptron architecture. They tried more sophisticated alternative approaches, but

they did not perform as well as the simpler model they used. The task was to predict the destination of a taxi given a prefix of its trajectory. Other architecture that did not perform as well were recurrent neural network, bidirectional recurrent neural network and memory network.

Dataset is composed of taxis running for a complete year in the city Porto, Portugal, containing 1.7 million data points, where each one represents a taxi ride composed of following attributes: sequence of GPS positions, latitude and longitude, ID of the taxi, client or taxi stand ID, timestamp when the ride began.

MLP architecture consists of input layer, which receives a representation of the taxi's prefix with the metadata. Hidden layer used consists of a matrix multiplication followed by bias and a nonlinearity, and in their approach, a single hidden layer of 500 neurons is used. The output layer predicted the destination of the taxi in latitude and longitude. [13]

### 3.6.4. Context-aware taxi demand hotspots prediction

This research reports that over 60 – 73% of the operation hours, taxi drivers from Taiwan, were without any passenger, which not only wastes energy but also pollutes the environment. Main reason of the unoccupied driving is the search for potential customer, wandering around in the city. Goal of this research is to predict the areas with potential demand from context and past history. Analysis of the data includes the time and location passenger got on taxi, which gives us suggestions on the demand distribution. On primitive data clustering methods are applied to find these high-density locations. Afterwards, the hotness scores are calculated for each identified cluster. This results in defining hotspots, meaning drivers can adjust their driving strategies for improvement. [14]

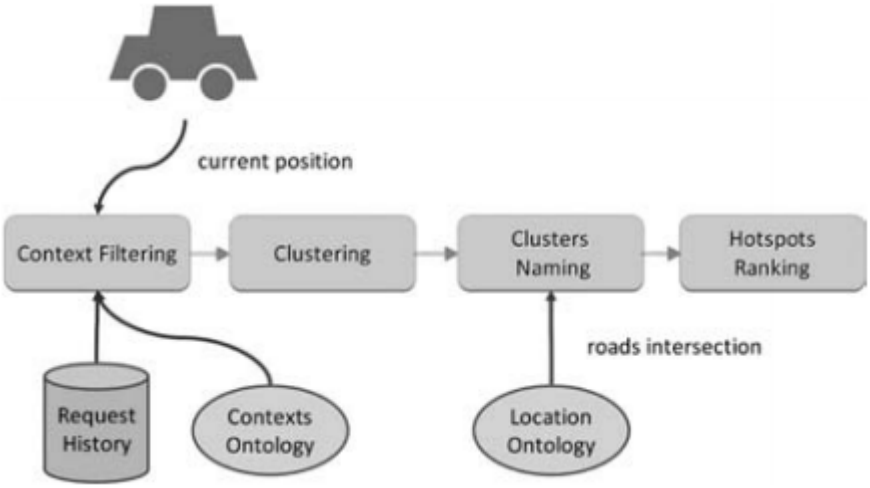



**Figure 1:** Proposed solution for hotspot ranking

Their proposed solution can be seen in figure above, where taxi driver takes a passenger, then, according to the context, request records are retrieved and filtered, which are then arranged into clusters. For each cluster, road fitting the distribution is used for annotation. Afterwards hotness scores are calculated. The geometry of the clusters, semantic meanings, and hotness scores form the hotspots. [14]

### 3.6.5. Hunting or Waiting? Discovering Passenger-Finding Strategies from a Large-scale Real-world Taxi Dataset

In this paper, they study feature sets which have the most significant impact on taxi drivers' performance, which can lead to improvement of driving strategies of others taxi drivers. This is based on the pattern analysis. Firstly, they symbolize the strategies to find passengers into a collection of features represented by time, location and strategy. The idea behind this what behaviour in certain time and location leads to good performance. The taxi performance predictor build on these selected features can achieve an accuracy of 85.3%.

The data are retrieved from taxis in a large city in China, Hangzhou, deployed with GPS device, that recorded information every 1 to 7 minutes. Their dataset consists of several fields used in their research, and they are following: ID of a vehicle, longitude and latitude, speed, state of occupancy, and timestamp. In comparison to the other research, this one does not investigate the driving trajectories, meaning they use only pick-up and drop-off events for each taxi ride, extracting the GPS data and timestamp. Used hotspot analysis contains partitioning metropolitan area into grid with equal intervals. They select only the top 99 regions, and then each of these cells in the grid is numbered. [15]

Taxi drivers most of the times use two different strategies, for finding new passengers, called hunting and waiting. In the comparison of these two strategies can be seen that depending on the hotspot and time of the day, different strategy may increase average pickup rate per hour.

In the taxi-pattern discovery there are three steps. Firstly, they design a set of taxi patterns with relation with time, location, and driving behaviour. Secondly, they use feature selection tool L1-Norm SVM to select the most discriminative features and learn a predictor based on these selected patterns. Lastly, the prediction of the set of taxis' performance based on the selected patterns.[15]

### 3.6.6. Modeling Level of Urban Taxi Services Using Neural Network

In this study, they apply neural network for analysis of the taxi service in urban area of Hong Kong. The neural network they present consist from basic three layers, input, hidden and output, which are connected completely forming a hierarchical network. For the hidden layer, a sigmoid activation function is used, and as for output layer, a linear function is chosen. Training algorithm based on back-propagation is used, where in training session the data, transformed to normalized vectors between the range from 0 to 1, are given to input and output layers. The input consists of several variables including number of taxis, taxi fare, average occupied time, average income, and total population. Output consists of taxi passenger demand, passenger waiting time, taxi availability, and taxi waiting time. They demonstrate that proposed neural network is able to produce favourable results, and can be used as tool to assist with decisions concerning taxi service structure. [16]

### 3.6.7. Taxi-Aware Map: Identifying and predicting vacant taxis in the city

This research uses data collected from taxis in Lisbon, Portugal. Each taxi has information about their location, service status, and timestamp. In their analysis, they use square grid, where each cell is assigned a number starting from the left bottom corner, increasing left to right, and then upwards after reaching rightmost cell.

Their objective is to predict vacant taxis in each grid cell in exact time. For that they use inference engine that estimates probabilities for cells given the historical data. To improve the overall performance, predictor needs to learn from errors, meaning error-based learning is part of the system. To reduce the amount of data in repository and maintaining the amount of useful information, method to detect adequacy of historical data is used.

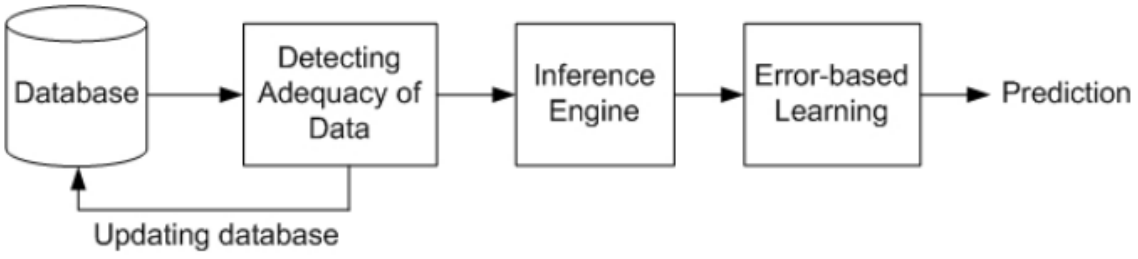



**Figure 2:** Flowchart of proposed algorithm

There are several variables that are important for the inference engine, and these are time of the day, day of the week, weather condition, and with them, computing number of vacant taxis. [17]

### 3.6.8. Modelling Taxi Trip Demand by Time of Day in New York City

This study utilizes a large database of taxi trips tracked by GPS to build a trip generation model at different time of day. It says that taxi trips are more numerous in places where transit is more accessible. They have six major factors: population, education, age, income, transit access time, and total jobs. From these features, income and total jobs influences taxi service the most. It is reported that influential factors vary temporally and spatially. [18]

### 3.6.9. Where to Find My Next Passenger?

Most of the taxi cars have GPS sensors for dispatching, which means they also store with a certain frequency a lot of information about geo-position, its timestamp, and occupancy information. This gives information about where and when most of passengers get on and off a taxi, and the other important part of information is where the taxi drivers usually go and how fast they find new passengers. Objective is to recommend locations with high probability of picking up a passenger, and suggestion of location where passengers could find a vacant taxi.

They state that good recommended parking place should bring high probability to get a passenger, with a short waiting time, and a long distance of the next trip, which would earn the most money for the taxi driver. In this approach, instead of typical grid-based partition of the map, they have recommendation system on road-segment level, which is more accurate and meaningful understanding for taxi drivers. [19]

### 3.6.10. Predicting Taxi Pickups in New York City

There is used dataset containing around 170 million of taxi trips, where each data of the trip has information about ID number of taxi driver, latitude and longitude, timestamp of pick-up and timestamp of drop off, duration, fare, and number of passengers. They select only specific region of New York City with a rectangle, that contained Manhattan. This part of the city is then divided into a grid.

They report that their decision tree regression model performs well, and is able to successfully predict pick-ups for taxi dispatchers and determine where to position taxi cars to

maximize the performance. It is also said that neural networks could also achieve good results. [20]

# 4. Data analysis and model implementation

## 4.1. Which data is used

For this task, I decided to work with data available for open public on the Internet. The first dataset used is from New York City [21] and second being Chicago [22]. There is absolutely huge amount of data available, which also comes with disadvantage. Large datasets mean large files, which led to not even using whole datasets.

Each city has its own set of names used to label individual variables. Some difference can be found in them. Some are the same, but important thing is there are many columns to choose from. To name few, there is:

“Trip Start Timestamp” in the Chicago dataset, but “Lpep Pickup Datetime” in the NYC dataset. Same goes for the final time, “Trip End Timestamp” and “Lpep Pickup Datetime”, respectively.

Another important variable used in this work is pickup location, and that is defined by its latitude and longitude. While In NYC they annotated simply as “Pickup longitude” and “Pickup latitude”, Chicago uses word Centroid in the middle of the string, marking it as center of pickup area, because of hiding the exact spot for privacy reasons, making it “Pickup Centroid Latitude” and “Pickup Centroid Longitude”.

Then in both datasets there is number of variables indicating fare amount – time-and-distance fare calculated by the meter. Extra charges, mostly in case of night trips, tip amount, only by credit cards as cash tips are not included, and many more.

I also used information about weather from National Weather Service [23] for both cities. There I was able to download data about maximum, minimum and average temperature for each day. In addition to that, there are features like precipitation, snow fall or even snow depth.

These datasets are then obviously connected by the day of the year. This means even though some trip is made early morning, average temperature is most probably lower than of some trip later in the day. Same goes for precipitation, which often differs throughout the day. It would be definitely better if the weather data were more detailed, if not by every hour, at least by parts of the day, for example, every 6 hours.

## 4.2. What can be seen in the data

In this part I would like to talk about what can be seen by human eye in the dataset. If there is some repeating pattern, notable differences, or important features that could be worked upon. The first, and probably main part is pickup location. This will be split in three different parts to analyse the data. First overall view will be showing individual days of the week's number of trips, by the hour of the day. Next will be overall number of pickups, on every day of the week. And lastly, pickups over hour of the day. Second part, will consists of analysis of weather, main focus will be set on average temperature, and precipitation.

### 4.2.1. Overall Pickups

As mentioned, first I will talk about the main picture, overall pickups. The graphs will look for all instances followingly:

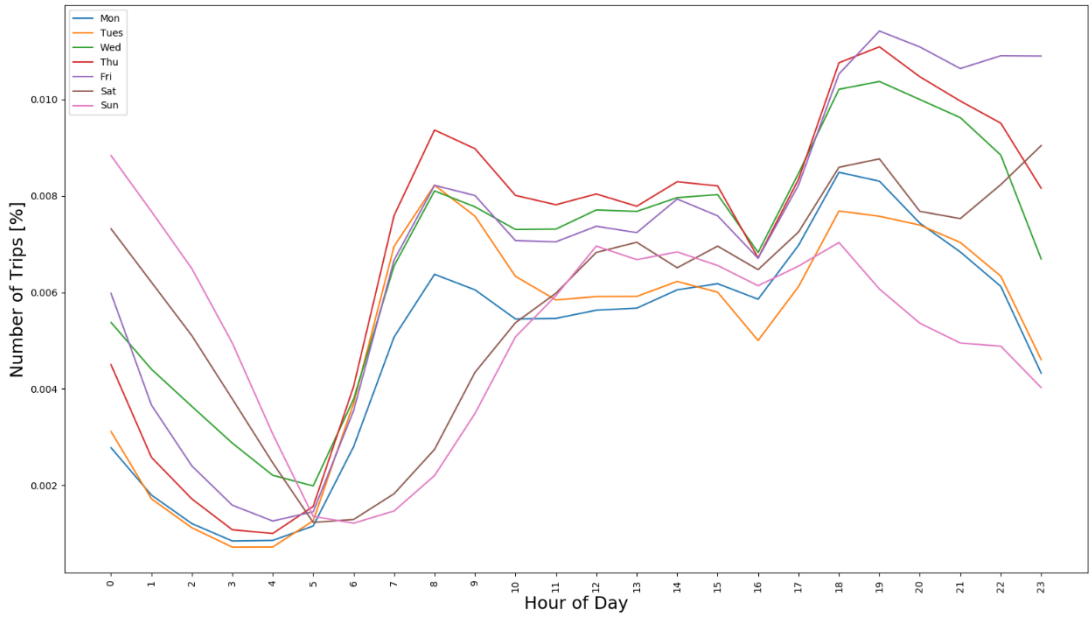



**Figure 3:** NYC pickups on January 2014 - each line represents one day, as can be seen, x-axis represent every hour of the day, and on y-axis there is number of trips by percentage

Several information can be read from the graph. First that could be pointed out, is that for every day in the week, the day for the taxi demand starts around 5 o'clock, in the morning, then follows really high growth in small amount of time, getting to the first peak at 8 o'clock. But there is also important to note, that even though there is some growth, this peak is not fully achieved in the Saturday and Sunday, the weekend, in comparison to other days. Then during the lunch

hours, nearly every day has approximately the same number of trips made. For this dataset, sudden drop for every day, on 16th hour can be noticed. Personally, I am not sure what causes this phenomenon. Following this event in time, every day achieves its own peak at 18th, 19th or 20th hour. At these hours, Friday is the only day that manages to keep the value of number of trips. Interestingly, Saturday is the only day that records significant rise in the taxi demand, all other days except these two, show decrease in the need of taxi service, Sunday being the lowest one from them. After this we get to the start of the graph, the zero hour mark, where every line representing day, no matter which, finds itself declining, till the morning at 5 o'clock.

Now let's see if this pattern applies to other datasets. To not overwhelm this part, I chosen to have only four distinct data, three being from NYC, and one from Chicago. In addition to that, the first one from NYC will be omitted in this part of comparison.

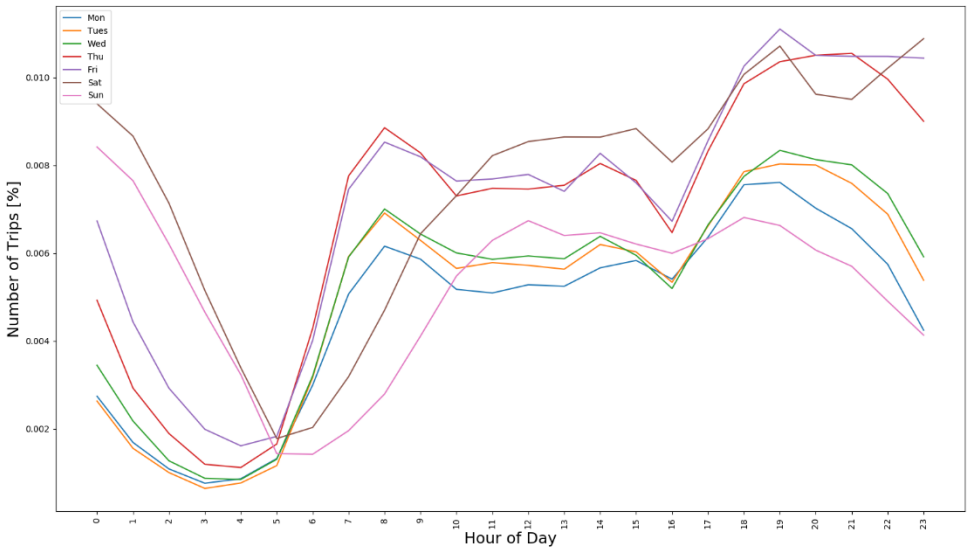



**Figure 4:** NYC pickups on October 2016

Even with the first different dataset overview, we can spot huge similarities in the graphs. Probably the main notable difference so far, is that Saturday managed to overcome taxi demand in the night hours, and has overall higher values compared to the dataset in January 2014.

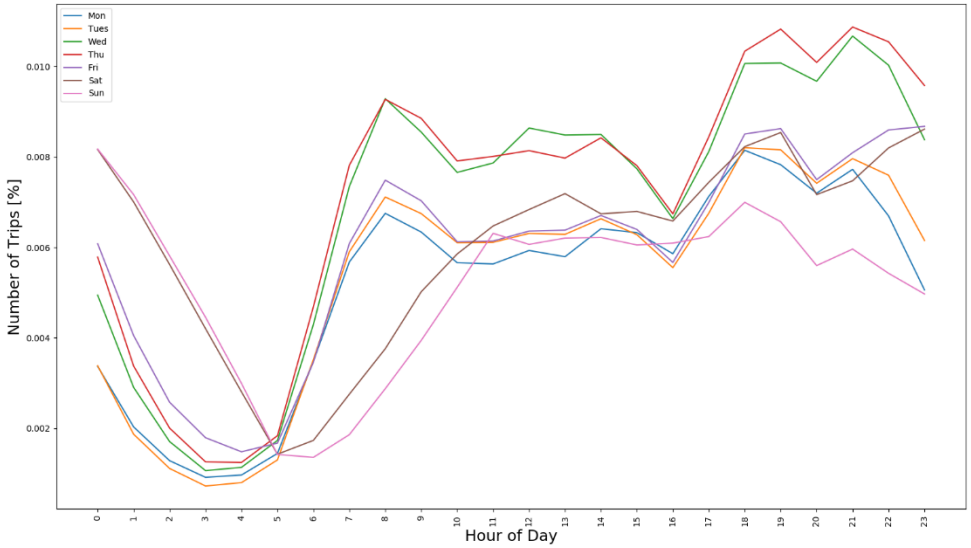



**Figure 5:** NYC pickups on June 2016

This dataset is a little bit different in comparison to the two previous ones. Here we can see probably the effect of vacations, or summer break, as two days, Wednesday and Thursday is much higher in comparison to previous month, and also, achieves greater values than other days in this current month. Other information seem similar in relation with the rest of the NYC dataset.

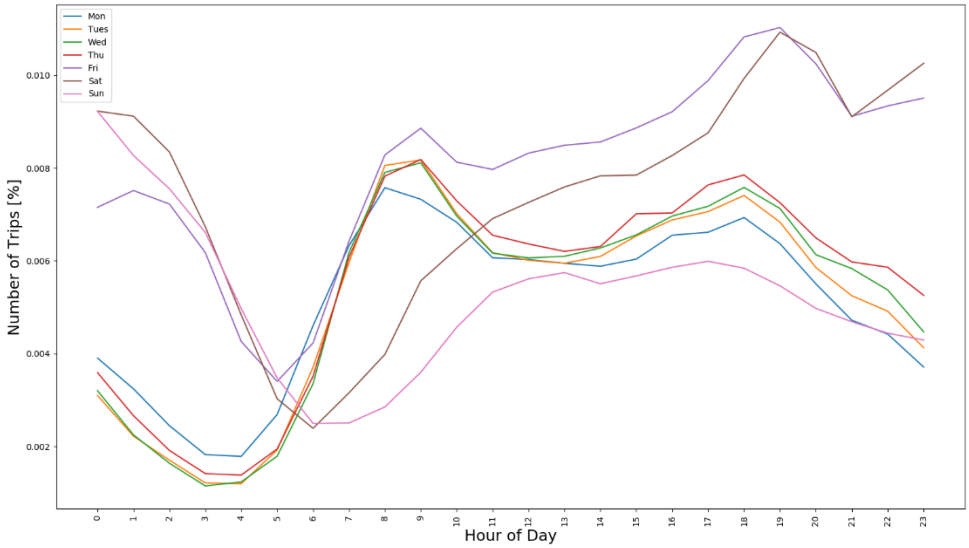



**Figure 6:** Chicago pickups on January 2016

This being the last dataset to compare, we can see what can be similar, and what is different to others. To start with similarities, all graphs follow the same curvature throughout the day. Day "starts" around 5'oclock, followed by rise and reaching its first peak during 9$^{th}$ hour. During work

hours, there a small portion of stagnation, which is succeeded by the same rise and fall like in previous dataset in NYC. Now on first sight, there is one unusual difference available to spot, and that is there is no “dip” in passenger demand on 16$^{th}$ hour, rather than that, all days continue to grow in numbers. Furthermore, in comparison to other NYC data, Chicago has higher passenger demand during night hours on Friday and Saturday in contrast to other days. Other than that, all similarities stay the same. What is interesting to point out in all of the datasets, is the huge rise of need of taxi service on Sunday between 23th hour and midnight, as everyone is rushing from their nightlives to their homes.

### 4.2.2. Pickups by Day of the Week

In comparison to the previous part, in this section we will try to detect if there are some kind of patterns in the dataset, when split only over days of the week.

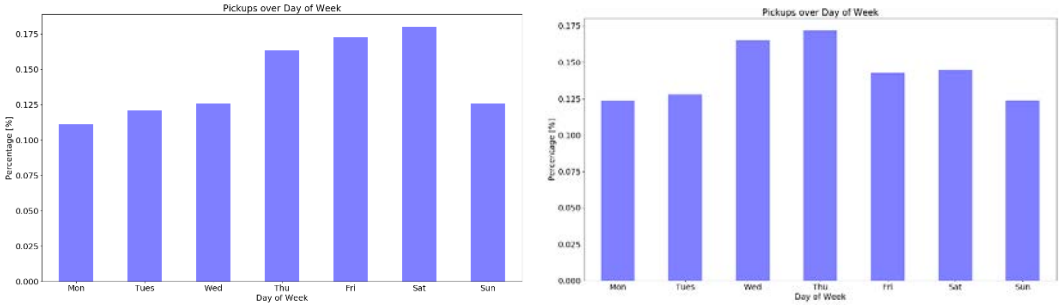



**Figure 7:** NYC dow comparison - difference between October 2015, (left), June 2016 (right)

In the first pair of graphs, we can see that the biggest similarity is that four days, Monday, Tuesday, Thursday and Sunday, keep the ratio nearly same between them. Other days has different values of passenger demand. Important to note, is that the bar graph on the right is related to the one on Figure 5: **NYC pickups on June 2016** in previous section. As there was much higher rise in Wednesday and Thursday during day, and most noticeably during night hours, we can see confirmation of that fact in this Figure 7: **NYC dow comparison**, that there is indeed higher demand in taxi service.

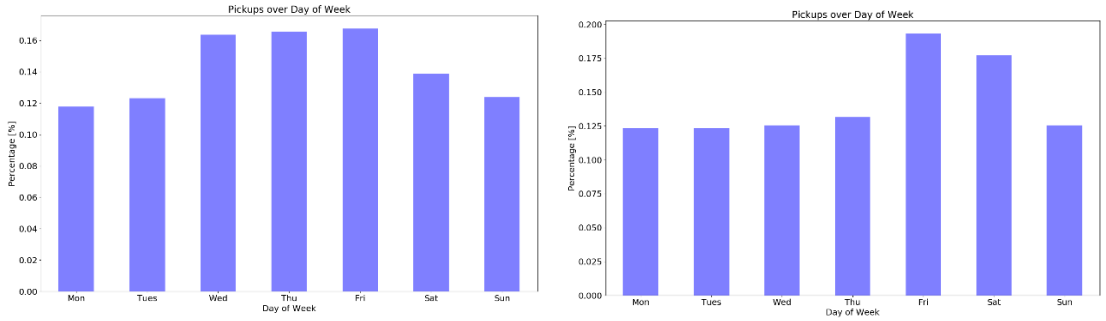



**Figure 8:** NYC and Chicago dow comparison - NYC January 2014 (left), Chicago January 2016 (right)

For the second pair, I chose to show both datasets taking place during the month January. What we can instantly notice, is that there is nearly no similarity between them, the only being same number of pickups on Friday. Next, in graph containing Chicago data, we can see that all work days have nearly the same value, with sudden rise in nearly of double the pickups on the last workday. After that, there is decrease in demand, getting to a point of nearly the same values like in previous workdays during the Sunday. On the left graph, strangely enough, pickup total on Wednesday reaches the same values like on Friday, and even higher number than on Saturday.

What is seen on all four figures is that last workday has never the lowest sum of pickups over the day, on the contrary, usually has the highest number, which is obvious as it is often the most favourite day of every human.

### 4.2.3. Pickups by Hour of the Day

In this last part of comparison taxi demand by pickup I will show data no matter the dow, depending fully on hour of the day. As can be seen on Figure 9: **Comparison of hod of all four datasets**, all four figures follow the same pattern, which could also be seen on Figure 3: **NYC pickups on January 2014**. Lowest point of demand being at 5$^{th}$ hour is equal for all cases. Ignoring the data about which day passenger is in, after reaching higher values at 9$^{th}$ hour, the need of taxi services keeps the same till 16$^{th}$ hour. The sudden decrease can be seen again in NYC datasets, and not in the Chicago one. Otherwise, there is not much to be pointed out in case of similarities or differences.

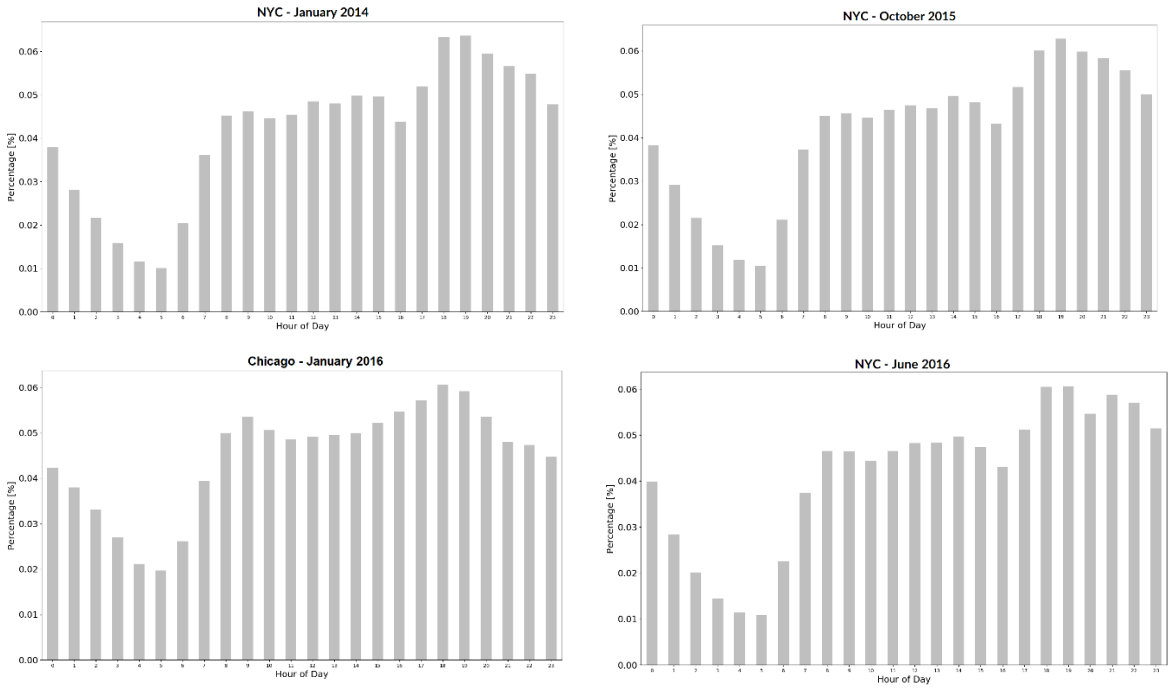



**Figure 9:** Comparison of hod of all four datasets - NYC 01-2014 (top-left), NYC 10-2015 (top-right), Chicago 01-2016 (bottom-left), NYC 06-2016 (bottom-right)

### 4.2.4. Weather

In my work, I decided to work mostly with two features, them being average and precipitation. This means it will be split into two parts, trying to bring spotlight on them, showing if there are present any similarities or differences.

#### 4.2.4.1. Precipitation

Precipitation represents weather states like rain, rainfall, snow, hail or others. In this case, it is measured in inches (1 in = 2.54 cm). Many days it does not rain, snow, or anything else, which makes the value of precipitation in the day equal zero.

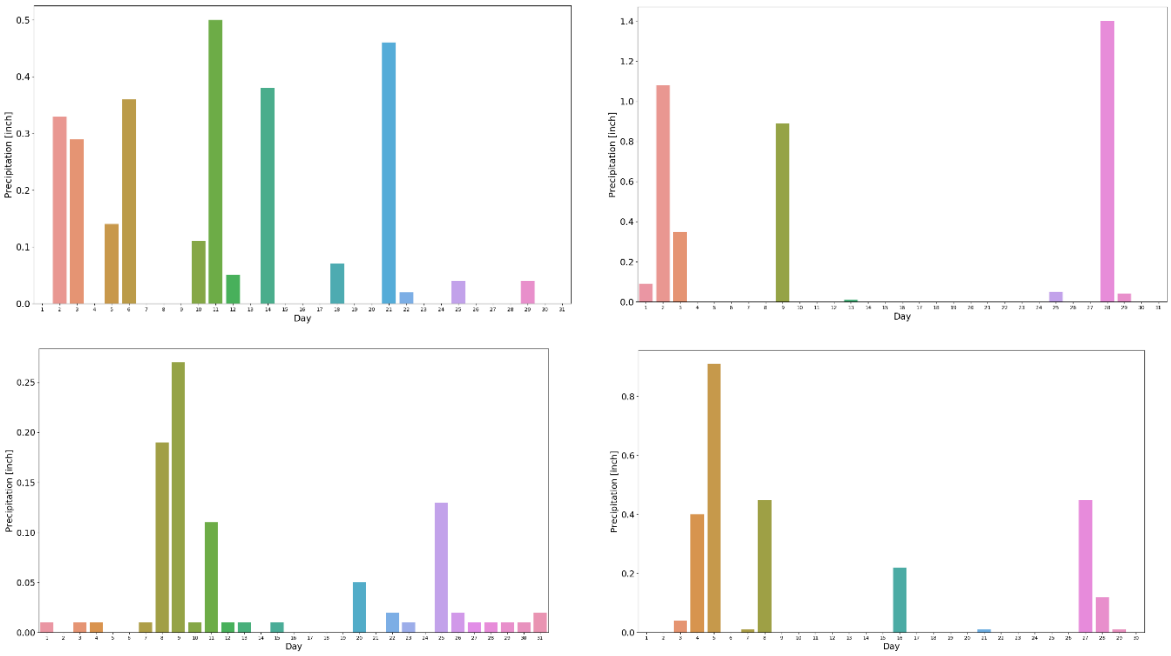



**Figure 10:** Comparison of precipitation of all four datasets - NYC 01-2014 (top-left), NYC 10-2015 (top-right), Chicago 01-2016 (bottom-left), NYC 06-2016 (bottom-right)

As can be seen in all four graphs, for human, precipitation seems random and unpredictable. It is hard to make any conclusions and mention similarities and differences. Even when we compare January in NYC and in Chicago, there are no obvious similarities.

#### 4.2.4.2. Average Temperature

This factor is definitely more easily observable, as it is clear that during different season, or even month, the temperature changes.

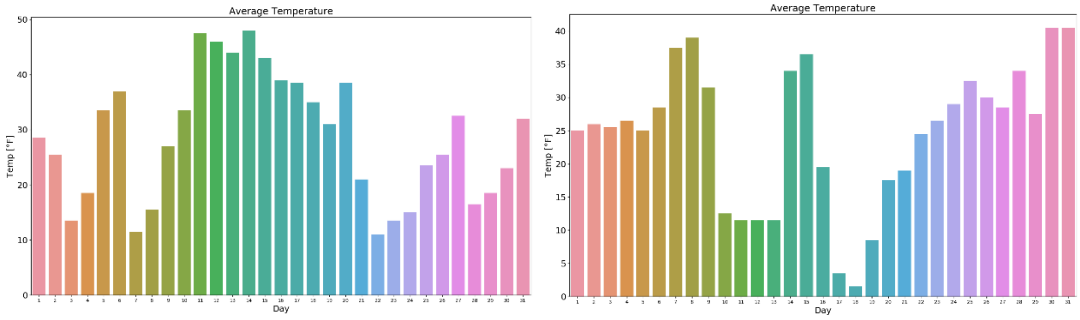



**Figure 11:** NYC and Chicago temperature comparison - NYC 01-2014 (left), Chicago 01-2016 (right)

As can be seen, in both cases the temperature is around 25°F, reaching peaks on around 40-45°F. There are some similarities, for example even though it is different year, in nearly two thirds of the month the temperature suddenly drops, and then rises again.

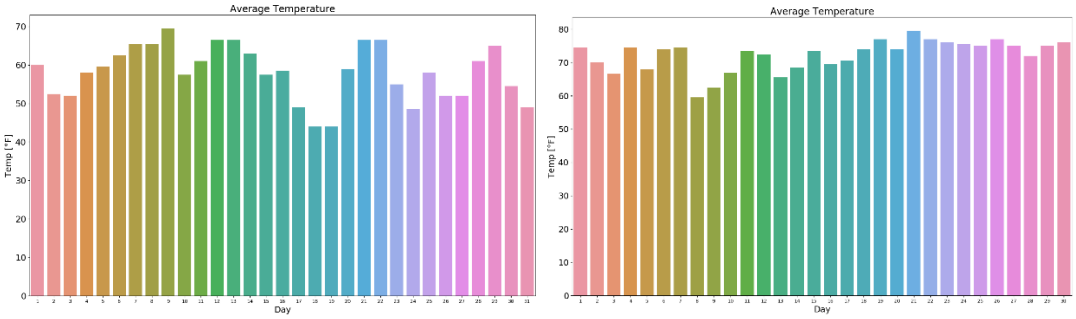



**Figure 12:** NYC in October and June comparison - NYC 10-2015 (left), NYC 06-2016 (right)

In these two months, we can obviously see approximately two time larger temperatures, whereas in June it is around 10°F higher temperature than in October. Moreover, we can see that June is more consistent, keeping nearly the same temperature every day, as in October the differences are more noticeable.

### 4.2.5. Pickup Heatmaps

In this section, we will try to look at graphical representation of data, on real life map, where increase of pickups on same location, or close area, will result in *stronger* use of colour. In this case, the highest *heat* is represented by colour red, gradually fading to yellow and green colours. This should provide some kind of idea, what locations are highly congested, where is taxi demand absent, or if there any patterns throughout the day.

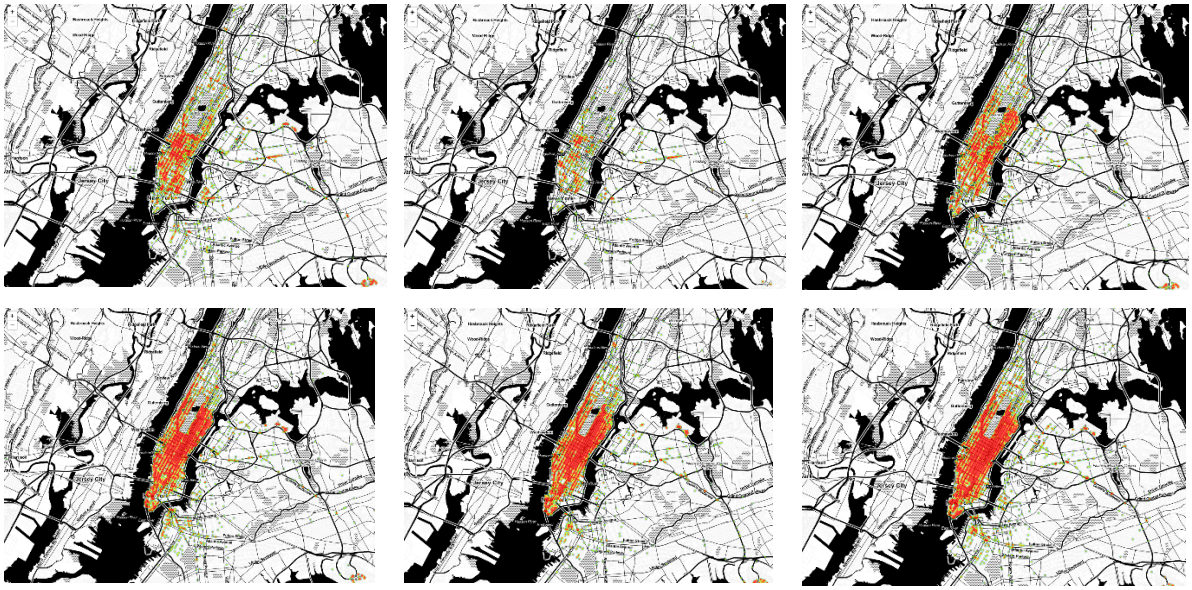


**Figure 13:** Heatmap for NYC, 6 different hours - NYC 06-2016, Monday; first row consists of hours: $0^{th}$, $3^{rd}$, $6^{th}$; second row consists of hours: $9^{th}$, $14^{th}$, $19^{th}$

As can be seen, the most action happens in central area of NYC, the Manhattan. The rural area, outside the centre, is much less used. The area that somewhat reaches near values of *heat* like in the centre, are airfields, whereas one is in bottom right corner, and to the East side from the centre. What can also be seen, that during night hours, the airfields are not used nearly at all.

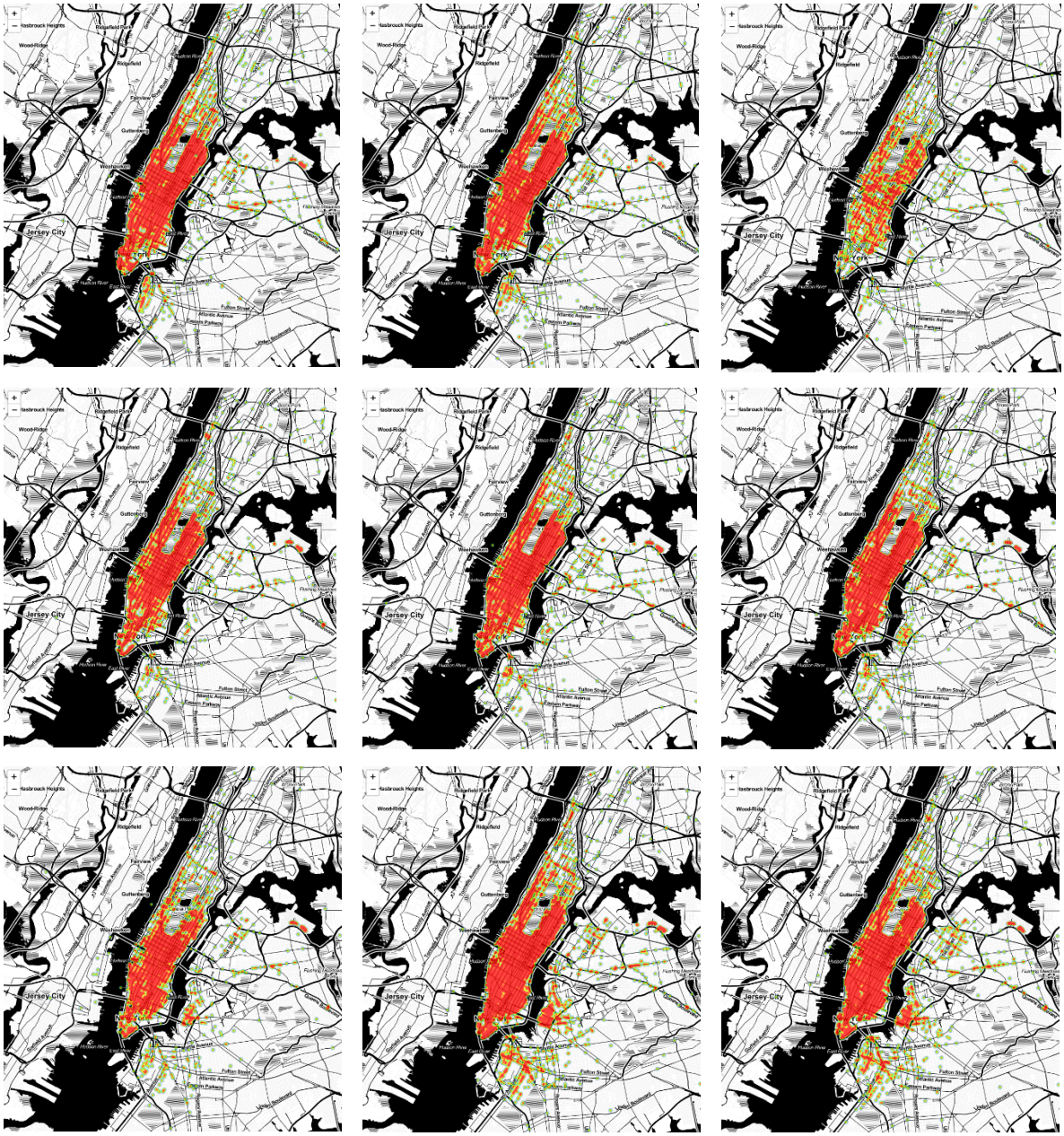


**Figure 14:** Heatmap for NYC, 3 different days, 3 different hours - NYC 01-2014, columns [days]: Tuesday, Friday, Saturday; rows [hours]: 7th, 16th, 23rd

In this comparison, we can see more clearly how days affect the taxi demand with connection with hour of the day. First row shows us Tuesday and Friday, both workdays, that in early morning, people are using taxi service, in comparison to weekend – Saturday, where is clearly

visible less of the pickup congestion. In the second row, we see three nearly identical cases of heatmap, as every image have red coverage nearly the same. This means during the day, in the evening, it does not have much of an effect whether it is Tuesday or Saturday, people use it nearly the same. And in the last row, we see a bit of an opposite to the first row, as on Monday, people do not tend to stay late in the city, which means we see less of a use of taxi services, in comparison to higher usage on Friday or Saturday. In addition to that, much higher activity can be seen in the rural area, also in cases of Friday and Saturday, the activity in rural area is higher in comparison to the Monday one.

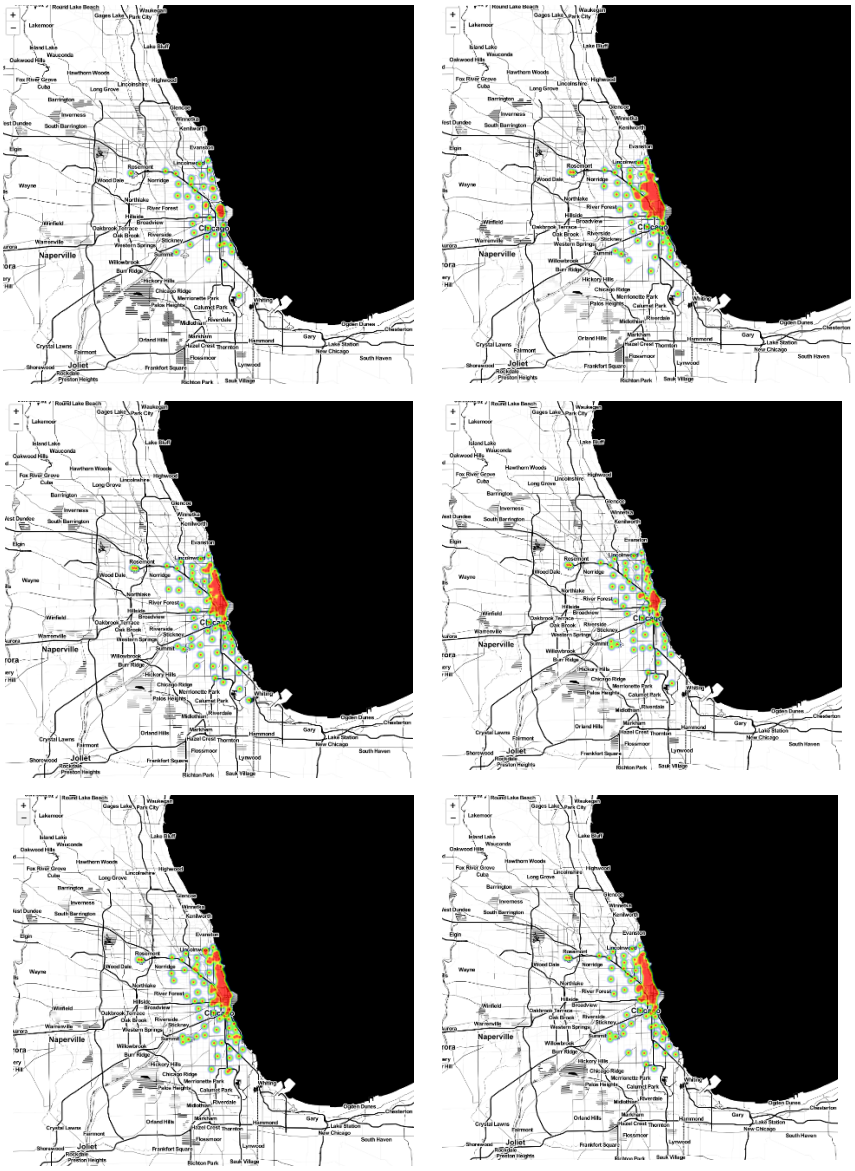


**Figure 15:** Heatmap for Chicago, 2 different days, 3 different times - Chicago 01-2016, columns [days]: Thursday, Sunday; rows [hours]: 3rd, 7th, 16th

On this figure, we can clearly see the difference between workday and weekend day. While in very early morning, on 3 o'clock, nearly noone uses taxi service on Thursday, there is really high demand on Sunday, most probably people still coming from pubs or clubs, or less probably people waking up really soon and traveling. Considering huge decrease of need of taxi in the morning at 7 o'clock, on Sunday people were going home from their nightlife, while on Thursday, there is increase in pickup congestion, as people are going to work. Lastly, in last row, we can see that no matter which of these two days, people are using nearly the same amount of taxi service at 16$^{th}$ hour.

## 4.3. Data-driven model implementation

In this section, the model and its implementation is analysed. The implementation is programmed in language Python 3.5.4, in integrated development environment Visual Studio 2015. For model implementation, I used open source libraries for Machine Learning in Python, scikit-learn.

First part of implementation is dedicated to selection and examination of available dataset. Important part of this process is to make sure that all of the data used is relevant and correct. In addition to that, I made sure that in case of huge outliers, the instances will be removed, then, preserve only full data, deleting ones with missing information or incorrect in that matter. In case of working with datetime, there was need of splitting the data into separate features, naming pickup time hour, and pickup time day. In connection to this, I was able to add weather data, joining every instance of pickup by day, with corresponding data information from weather dataset. Upon further inspection, there was need of converting some of the attributes from weather file, to values, according to the website's [23] description.

Because I was working with two different sources of data, I had to make sure to unify the names of the columns, representing each feature, as mentioned earlier.

Important thing to point out is that downloaded dataset were too large to work properly with. This lead to random selection of rows from the data. In the end, I took and used only 0.0005% of every file, to keep the ratio of the data same. This was essential to ensure that the taxi demand ratio also stays the same inside the city, with connection to rural and other areas. This was done by creating random sample out of whole dataset – its length. After this each line corresponding to the values in created random sample are saved.

### 4.3.1. Random Sampling

Some of the datasets were still too large to be comfortable worked with, so I used reservoir sampling, which is algorithm, that selects desired amount of rows out of dataset, where each line has exactly the same probability to be in final data selection. In my case, I used reservoir sampling using geometric approximation [24], which has much faster sampling times. The sampling probability is *R/j* where *R* is desired length of output dataset, and *j* is index of currently selected element in the original dataset.

### 4.3.2. Data Normalization

As in most cases of Machine Learning, data normalization is applied on dataset. Firstly, there is part where I normalize pickup location – latitude and longitude. This is the usual case of normalization where I apply formula:

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

This is also called, Min-Max scaling, changing values of these features in between range of <0; 1>, but keeping its distribution.

I also tried to normalize time, to make sure that repeating pattern of hours in the day, and days in the week is present. In default dataset, hours go from value zero upwards, to last 23rd hour. This means that even though hours 23 and 0 are neighbours, the value difference between them is huge. Same goes for days, where in dataset they are marked as indices from zero to six, whereas in words it is from Monday to Sunday, and even though there is transition from Sunday to Monday, it is not properly recorded in the values.

Because of this reasoning, to normalize time, I transformed them to individual points on circle. To accomplish this, I needed to create two new features for each attribute, one for sinuses, other for cosines.

$$time\_sin = \sin\left(\frac{2 * \pi * \left(hour + \frac{minutes}{60}\right)}{24}\right);\ \ tim\_cos = \cos\left(\frac{2 * \pi * \left(hour + \frac{minutes}{60}\right)}{24}\right)$$

$$day\_sin = \sin\left(\frac{2 * \pi * weekday}{7}\right);\ \ day\_cos = \cos\left(\frac{2 * \pi * weekday}{7}\right)$$

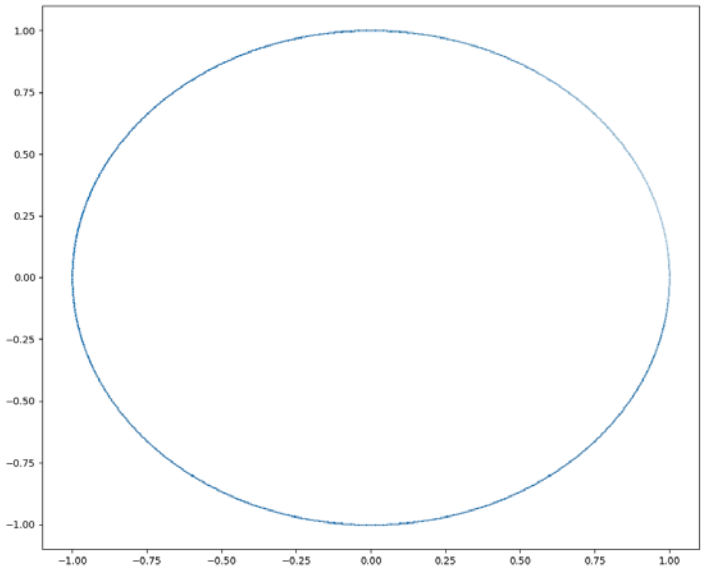



**Figure 16:** Representation of normalized time - sin and cos, making whole circle

As can be seen in graph above, after we normalize time, we achieve full continuous cycle, ensuring that repeating pattern. Because there is enough data, at every hour, there are huge number of pickups, every minute, which is the cause of this circle.

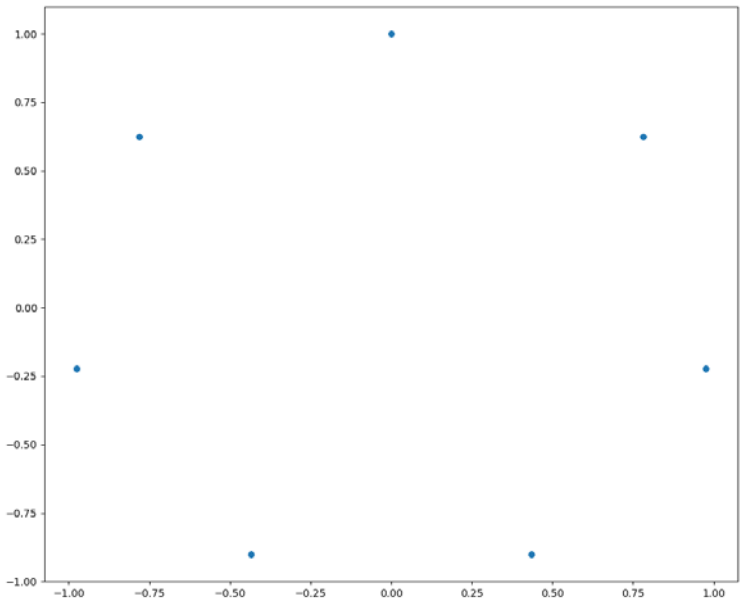



**Figure 17:** Representation of normalized days - sin and cos, making 6 points

Similarly, it goes for days, with the exception that now we get 7 remote points, each representing one day, with the one on top being Monday, going right getting to Tuesday and to the rest of the days, fulfilling the *"circle"*.

### 4.3.3. Preparing data for transferability / spatial invariance

To be able to somehow transfer the model to different location, like different part of the city or some other city, we need some feature that ensures that machine learning algorithm knows in what city we are currently *"in".* Like in real life, taxi drivers have their own knowledge about at what time, in current day, they can expect passenger in need of taxi, machine learning algorithm learns the same. So, I create new feature that can be called *taxi station*. I randomly create 200 stations, within 10 km radius around centre of all of the pickup locations – somewhere in the centre of the city.

Then I divide data into 200 parts, assigning each part its own taxi station. This way, when transferring to different unknown location, algorithm can make predictions accordingly with the new location, otherwise it would still predict in its *home* city where it learned to predict pickup location.

### 4.3.4. Machine learning model

For this task, I tried three different machine learning models. Random Forest Regression, Multi-Layer Perceptron, and Linear Regression. Considering unsatisfactory results from the last two named methods, I decided to work only with Random Forest Regression, with proper selection of its hyperparameters and overall design.

First important parameter for selection, is number of estimators, or what can be called "*number of trees in the forest".* Usually, the higher the number, the better and more precise will be the results, but at a cost of time. There is a point, where increasing number of estimators becomes wasteful, to find that limit, I had to try several number of estimators, in many runs.

After number of runs, I found out that in my case, after 150 estimators, it becomes pointless to increase. The limit at first was 500 estimators, which was then lowered to 200 estimators, and in the end, I was trying only between values of 150 and 250, where 150 number of estimators gave satisfactory results.

Next parameter that was tweaked was min samples leaf. As name indicates, this value indicates how many samples are required at minimum to be at any leaf node. Initially, the range I tried to create model in was from 5, to 80. This was changed fast to lower numbers, as higher number of min samples in leaves resulted in unsatisfactory output. In the last runs it was run with two different values, either 5, or 25 samples in the leaf nodes.

Last parameter that was being changed was max features. This is the number of features to be considered when looking for the best split. There were three options used, "*auto*", "*sqrt"* and "*log2*".

$$auto: \max_{features} = n\ features$$

$$sqrt: \max_{features} = sqrt(n)\ features$$

$$log2: \max_{features} = log2(n)\ features$$

I tried to always pick the best combination of parameters, but later it became obvious that this parameter seemed to be picked at *random*, or had little to no influence on the training process in this case.

Splitting data was made in usual manner, having train size from 80% and test size to 20%, or splitting it into two thirds for the train data and one third for the test data.

As I mentioned that I was using reservoir sampling to select random data from the dataset, which was still large. This made it possible that I had opportunity to test influence of size of data on the fitting process. After while it was clear that 100 thousand of rows was not enough for proper use of machine learning algorithm. But going over 400 thousand data started to be unnecessary, as at high cost of time, the results were not improving by large margin.

For testing the predictions and receiving score, I used root mean square error and R2 scoring.

$$RMSE = \sqrt{\sum \frac{\left(y_{pred} - y_{test}\right)^2}{N}}$$

R2 scoring is regression function, where best possible score is 1.0. A constant model which always predicts expected value, disregarding the input features gets score of 0.0. Considering the model can be worse, it can get negative values too. Because these scoring techniques did not give me good enough perception how is the model working, I decided to work with regular distance between points – using haversine formula for calculation between two sets of points containing latitude and longitude.

# 5. Experiments and evaluation

## 5.1. Types of experiments and evaluation

There was some room for different types of experiments considering of availability of various datasets. There could be basic training and testing within the same dataset, using only one part of the city. For example, training it, and testing it, on the NYC *Yellow* 2014-01 dataset, without use of NYC *Green* 2014-01 dataset. To clarify, *Yellow* dataset consists of central part of the NYC, whereas *Green* dataset consists mainly of the rural areas in NYC. Next, with the use of *Green* data, we expand the area of prediction, making it harder for proposed model to foresee the taxi demand.

Another type of experiment was training it on dataset in January 2014, and testing it on later datasets in real time, like October 2015 or June 2016. Different combinations can be made, or even combining two datasets from different years, to predict the pickup location.

Lastly, crossover between NYC and Chicago, where the proposed model would train on dataset containing NYC information, and trying to predict taxi demand in Chicago and testing the ability to do so. In this case I used my proposed transferability.

Something else to consider during experiments was use of features. In some cases, more features were tried that in others. For example, in some cases only pickup time – day, pickup time – hour, average temperature and precipitation was used, in other tries, I added to the feature list attributes like trip duration, passenger count, or maximum and minimum temperature. In case of transferability, created taxi station location had to be inserted into said feature list.

For evaluation, I decided to have five thresholds by which I compared my predictions. Where I would calculate haversine distance between predicted location and actual location which happened in real life. Considering how close the prediction was, it would get assigned to the specific threshold counter. The thresholds I used were *smaller than 0.5 km* as the best prediction, next number was *1.5 km*, then *2.5 km*, and *7 km*, and lastly, all values above *7 km* were considering as wrong predictions. In addition to that, it contains mean of prediction distance.

Another decision I made in evaluation process, was comparing predicted pickup location, with more than one taxi demand that happened at exact time as it was supposed to predict. This way I compared the dataset within set amount of time with my predicted value, to see, if there will be some other need of taxi service in close distance. For example, proposed model could have predicted that at some location will be taxi demand, but after comparing to real value of the

pickup location, we would receive *8 km* distance, which can be considered high. But if we try and check whether within 30 seconds, at predicted location in *1 km* radius, will be some other passenger waiting for pickup, we can consider that prediction, by my evaluation, as more successful that it was before.

## 5.2. Experiments

I started with simple experiments containing 100 thousand data using reservoir sampling, from NYC January 2014 dataset. Features used were pickup time hour, pickup time day, average temperature and precipitation.

| |
|---|
| Number of estimators: 150<br>Min Samples Leaf: 5<br>Max Features: auto |
| Feature importance:<br>pickup_time_hour 0.725009<br>average temperature 0.145372<br>pickup_time_day 0.0749043<br>precipitation 0.0547138 |
| Evaluation (normal):<br>Mean: 3.134499 km<br>[<0.5]: 03.4467 %<br>[<1.5]: 21.8033 %<br>[<2.5]: 25.4667 %<br>[<7.0]: 43.4567 %<br>[>7.0]: 05.8267 % |
| Evaluation (within 30 seconds):<br>Mean: 1.373547 km<br>[<0.5]: 14.6433 %<br>[<1.5]: 52.2767 %<br>[<2.5]: 22.4967 %<br>[<7.0]: 10.1533 %<br>[>7.0]: 00.4300 % |

**Table 1:** Experiment 1 – NYC 01-2014, 100k

From the evaluation, it is visible that the predictions made may not be exactly correct in the tested time, but when increasing the range in which we consider the results as feasible even only by half a minute, we can see that the prediction is much better, having over half of the predictions inside 1.5 km radius of the tested pickup location. From the feature importance, we can see that if we use only these four features, pickup time hour has the most influence on the prediction, by large margin.

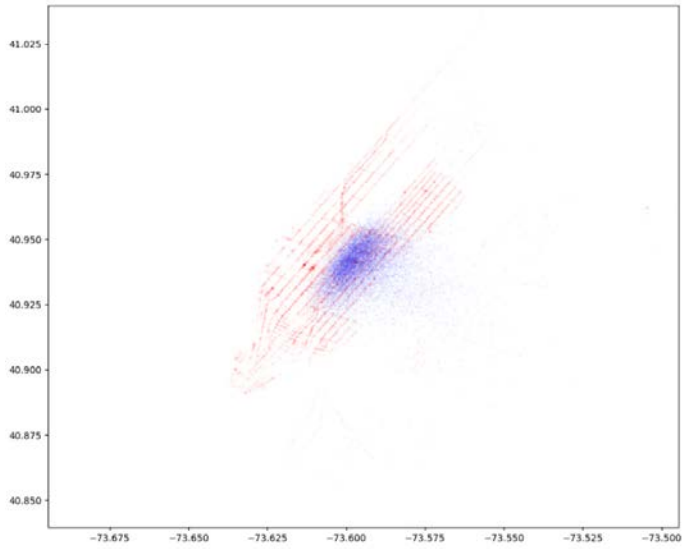



**Figure 18:** Experiment 1, actual and predicted pickup location - red dots are real locations, blue are predicted locations

As we can see in the graph, predicted locations are bundled inside the centre of the city. Important to note that this is exact predicted location at actual time stamp. Evaluation within 30 seconds achieves better precision.

---

In the next experiment, I doubled the data on which the model was trained. Training data used was NYC January 2014 dataset, but predicted and tested dataset was NYC October 2015. Features used were pickup time hour, pickup time day, average temperature and precipitation.

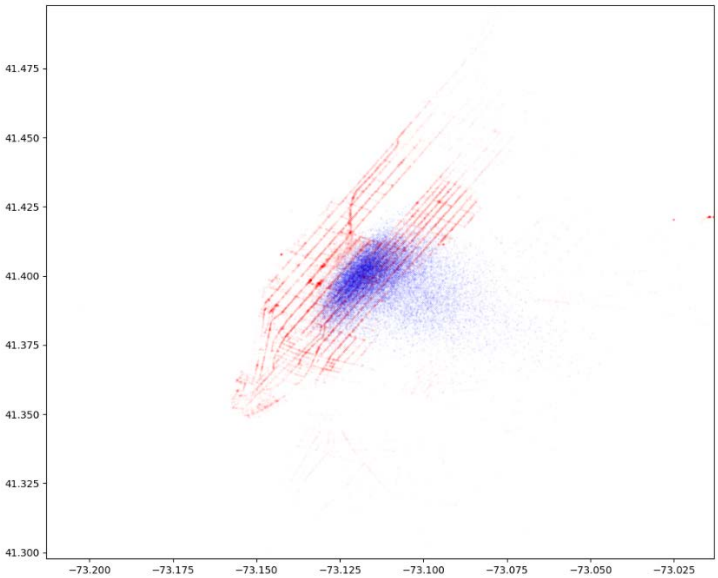



**Figure 19:** Experiment 2, actual and predicted pickup location

| |
|---|
| Number of estimators: 200<br>Min Samples Leaf: 5<br>Max Features: auto |
| Feature importance:<br>pickup_time_hour 0.673962<br>average temperature 0.180867<br>pickup_time_day 0.0941976<br>precipitation 0.0509734 |
| Evaluation (normal):<br>Mean: 3.361282 km<br>[<0.5]: 03.2083 %<br>[<1.5]: 19.9850 %<br>[<2.5]: 24.3667 %<br>[<7.0]: 45.3417 %<br>[>7.0]: 07.0983 % |
| Evaluation (within 30 seconds):<br>Mean: 1.040675 km<br>[<0.5]: 22.5467 %<br>[<1.5]: 58.0833 %<br>[<2.5]: 15.0500 %<br>[<7.0]: 04.2217 %<br>[>7.0]: 00.0983 % |

**Table 2:** Experiment 2 – NYC 01-2014 & NYC 10-2015, 200k

As we can see in the graph and from the evaluation, prediction actually improved in comparison to the experiment 1, and all that changed was number of rows used.

---

In this next experiment, transferability was tried, where we trained the proposed model on data available from Chicago, and tested its prediction on NYC dataset.

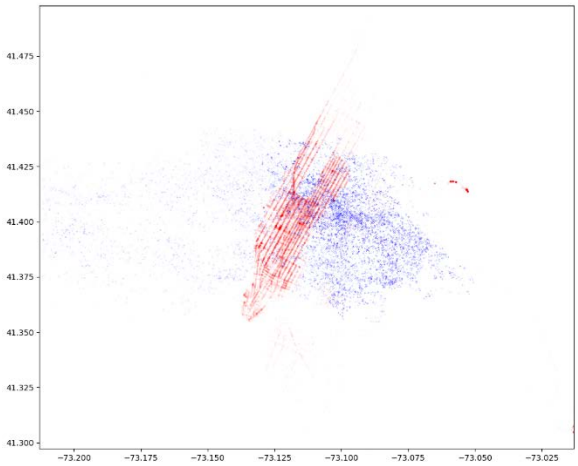



**Figure 20:** Experiment 3 - predicting NYC data, learned by Chicago data

| |
|---|
| Number of estimators: 200<br>Min Samples Leaf: 25<br>Max Features: log2 |
| Feature importance:<br>trip_duration 0.377063<br>pickup_time_hour 0.284417<br>taxi_longitude 0.0969928<br>average temperature 0.0755587<br>taxi_latitude 0.0644562<br>pickup_time_day 0.0626264<br>precipitation 0.0388857 |
| Evaluation (normal):<br>Mean: 5.205818 km<br>[<0.5]: 009167 %<br>[<1.5]: 074517 %<br>[<2.5]: 136700 %<br>[<7.0]: 639367 %<br>[>7.0]: 140250 % |
| Evaluation (within 30 seconds):<br>Mean: 1.886761 km<br>[<0.5]: 070117 %<br>[<1.5]: 355017 %<br>[<2.5]: 326650 %<br>[<7.0]: 245700 %<br>[>7.0]: 002517 % |

**Table 3:** Experiment 3 – Chicago 01-2016 & NYC 06-2016, 200k

What can be seen on Figure 20: **Experiment 3**, is obvious that model learned about Chicago data its taxi demand distribution in the area. However, is still able to predict somewhat precise pickup locations close to the actual real one.

---

This next experiment on Table 4 was done on two datasets, where it was trained on NYC October 2015, and was tested on June 2016. Both *Yellow* and *Green* taxi sets were used. Number of rows used for training was 300 thousand.

| |
|---|
| Number of estimators: 150<br>Min Samples Leaf: 25<br>Max Features: sqrt |
| Feature importance:<br>trip_duration 0.565063<br>pickup_time_hour 0.346848<br>average temperature 0.0426335<br>pickup_time_day 0.0326101<br>precipitation 0.0128453 |
| Evaluation (normal):<br>Mean: 2.696682 km<br>[<0.5]: 04.9289 %<br>[<1.5]: 27.4889 %<br>[<2.5]: 29.9567 %<br>[<7.0]: 32.9700 %<br>[>7.0]: 04.6556 % |
| Evaluation (within 30 seconds):<br>Mean: 0.673213 km<br>[<0.5]: 42.9711 %<br>[<1.5]: 52.2389 %<br>[<2.5]: 04.2244 %<br>[<7.0]: 00.5378 %<br>[>7.0]: 00.0278 % |

**Table 4:** Experiment 4 – NYC 10-2015 & NYC 06-2016, 300k

We can notice its ability to learn and predict taxi demand, while being in the same city.

---

This experiment was done on 100 thousand rows of NYC June 2016 dataset. It was trained on *Yellow* data, and then tested its prediction on *Green* data.

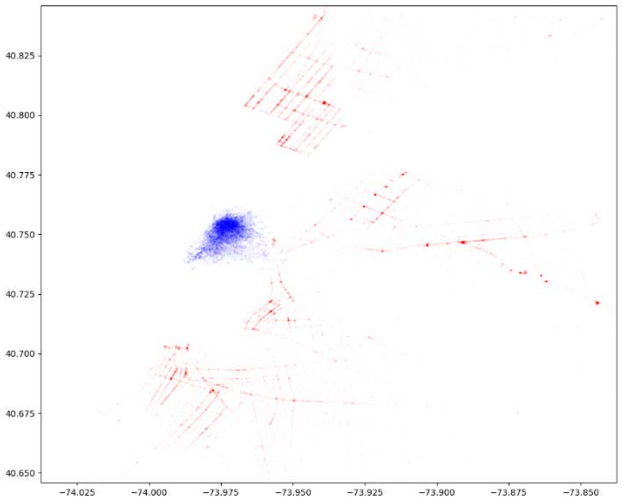



**Figure 21:** Experiment 5 - predicting *Green* data, trained on *Yellow* data

| |
|---|
| Number of estimators: 150<br>Min Samples Leaf: 25<br>Max Features: log2 |
| Feature importance:<br>pickup_time_hour 0.875984<br>pickup_time_day 0.0423334<br>average temperature 0.0368592<br>taxi_longitude 0.0234768<br>taxi_latitude 0.0213463 |
| Evaluation (normal):<br>Mean: 7.311765 km<br>[<0.5]: 00.0031 %<br>[<1.5]: 00.2781 %<br>[<2.5]: 01.5575 %<br>[<7.0]: 50.4404 %<br>[>7.0]: 47.7209 % |
| Evaluation (within 30 seconds):<br>Mean: 4.703078 km<br>[<0.5]: 00.0093 %<br>[<1.5]: 01.1960 %<br>[<2.5]: 07.1109 %<br>[<7.0]: 83.9519 %<br>[>7.0]: 07.7320 % |

**Table 5:** Experiment 5 – NYC 06-2016 Yellow & NYC 06-2016 Green, 100k

It can be obviously seen, that in this case, model was unable to learn any relevant information, which made him useless for predicting the taxi demand in this area of the city.

---

In this last experiment, I tried to put more available features to learn, while also giving the model 400 thousand of data rows to learn from. It was learned on NYC January 2014 dataset, while it was tested with its ability to predict taxi demand on NYC October 2015 dataset.

What can also be seen on Figure 22: **Experiment 6**, that it was strongly trained over bigger area of the city, than any of the previous models.

| |
|---|
| Number of estimators: 150<br>Min Samples Leaf: 5<br>Max Features: sqrt |
| Feature importance:<br>trip_distance 0.463171<br>trip_duration 0.210148<br>pickup_time_hour 0.198743<br>passenger_count 0.0249126<br>maximum temperature 0.0245435<br>minimum temperature 0.024444<br>average temperature 0.0221778<br>pickup_time_day 0.0198895<br>precipitation 0.0119707 |
| Evaluation (normal):<br>Mean: 1.398230 km<br>[<0.5]: 17.2756 %<br>[<1.5]: 49.2967 %<br>[<2.5]: 22.0667 %<br>[<7.0]: 10.4333 %<br>[>7.0]: 00.9278 % |
| Evaluation (within 30 seconds):<br>Mean: 0.339363 km<br>[<0.5]: 83.8444 %<br>[<1.5]: 15.5622 %<br>[<2.5]: 00.4411 %<br>[<7.0]: 00.1478 %<br>[>7.0]: 00.0044 % |

**Table 6:** Experiment 6 – NYC 01-2014 & NYC 10-2015, 400k

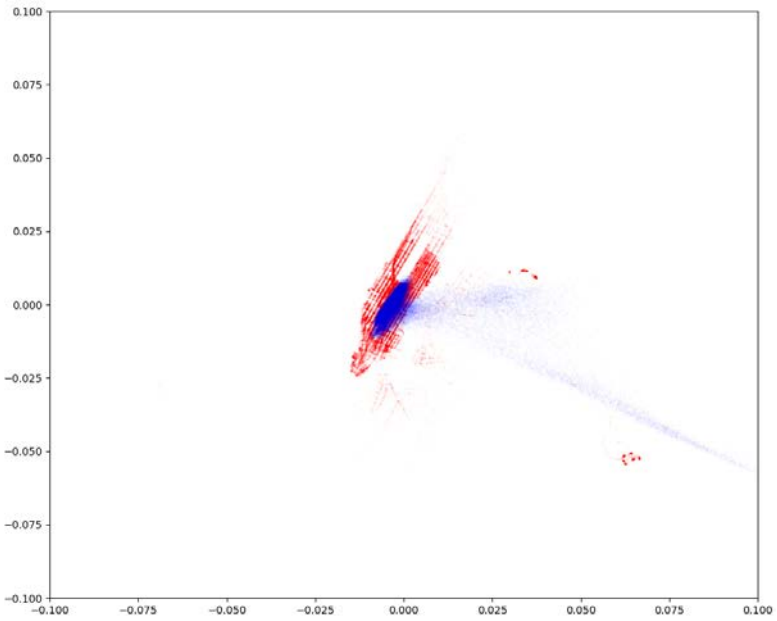



**Figure 22:** Experiment 6 Text

### 5.2.1. Recapitulation

Many experiments were done, some were less successful than others, either because of number of data used for training, or different feature set. Some failed to predict taxi demand completely, showing inability to be robust.

What I noticed on every experiment I made, even the ones that are not mentioned here, is that nearly always, pickup time hour has the most influence on its ability to predict taxi demand.

Division of the prediction by distance proved to be useful indicator of how successful the proposed model is. In addition to that, combining it with comparison within set time from the predicted data showed that even when the prediction is not exactly correct, within short amount of time there will be other passenger available for transport.

# Conclusion

In this work, methods of passenger demand modelling were introduced, briefly shown how they work and what they do. Datasets were thoroughly examined, and described, trying to point out all important features and patterns that can be seen by human.

The objective of this thesis was to design and create data-driven model of taxi passenger demand. After careful examination of features and attributes, I was able to design and implement data processing model, with selection of learning techniques and parameters. In addition to that, weather data was implemented, in effort to improve model's ability to predict passenger taxi demand.

Experiments and their evaluation show, that there is possibility to predict approximate pickup location of passenger taxi demand. If we consider that the prediction does not have to be 100% exact latitude and longitude, and also in consideration with some time difference (for example half minute), the results seem much better. In my opinion, these predictions could be improved with addition of features, like Taxi ID, where it could perhaps learn patterns each individual makes. Also considering weather data available, was limited only to daily history, it may have hindered the learning process.

In terms of transferability of the model, it showed that it is definitely not reliable for proper prediction, as it had huge mistakes and deviated from correct solution to the prediction problem. It could be seen that even with addition of artificially created taxi station locations, it did not produce feasible results. Perhaps if in the real dataset were some information about taxi location before the actual pickup of passenger, it could be easier to transfer the model to different location and predict the taxi demand.

To summarise, the model was able to somewhat predict the passenger taxi demand, but it is not fully usable for transfer to different areas. Also in case of some new features from datasets, its learning process could be improved.

# Bibliography


[1] D. Pojani and D. Stead, “Sustainable Urban Transport in the Developing World: Beyond Megacities,” *Sustainability*, vol. 7, no. 6, pp. 7784–7805, 2015.

[2] J. Aarhaug, *Taxis As Urban Transport*, no. 1308/2014. 2014.

[3] J. S. Armstrong and K. C. Green, “Department of Econometrics and Business Statistics Demand Forecasting : Evidence-based Methods,” no. September, 2005.

[4] G. Cho *et al.*, “Synopsis of Traffic Simulation Models,” *Accid. Anal. Prev.*, vol. 1, no. 4, pp. 99–110, 2014.

[5] D. Solomatine, L. M. See, and R. J. Abrahart, “Data-Driven Modelling : Concepts , Approaches and Experiences,” pp. 17–31.

[6] J. Ke, H. Zheng, H. Yang, and X. (Michael) Chen, “Short-term forecasting of passenger demand under on-demand ride services: A spatio-temporal deep learning approach,” *Transp. Res. Part C Emerg. Technol.*, vol. 85, pp. 591–608, 2017.

[7] J. Zhang, Y. Zheng, and D. Qi, “Deep Spatio-Temporal Residual Networks for Citywide Crowd Flows Prediction,” 2016.

[8] H. Yang, C. W. Y. Leung, S. C. Wong, and M. G. H. Bell, “Equilibria of bilateral taxi-customer searching and meeting on networks,” *Transp. Res. Part B Methodol.*, vol. 44, no. 8–9, pp. 1067–1083, 2010.

[9] L. Breiman, “Randomforest2001,” pp. 1–33, 2001.

[10] T. M. Oshiro, P. S. Perez, and J. A. Baranauskas, “How many trees in a random forest?,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 7376 LNAI, pp. 154–168, 2012.

[11] J. Lee, I. Shin, and G. L. Park, “Analysis of the passenger pick-up pattern for taxi location recommendation,” *Proc. - 4th Int. Conf. Networked Comput. Adv. Inf. Manag. NCM 2008*, vol. 1, pp. 199–204, 2008.

[12] L. Moreira-Matias, J. Gama, M. Ferreira, and L. Damas, “A predictive model for the passenger demand on a taxi network,” *IEEE Conf. Intell. Transp. Syst. Proceedings, ITSC*, no. April 2017, pp. 1014–1019, 2012.

[13] A. De Brébisson, É. Simon, A. Auvolat, P. Vincent, and Y. Bengio, “Artificial neural networks applied to taxi destination prediction,” *CEUR Workshop Proc.*, vol. 1526, pp. 1–12, 2015.

[14] H. W. Chang, Y. C. Tai, and J. Y. J. Hsu, “Context-aware taxi demand hotspots prediction,” *Int. J. Bus. Intell. Data Min.*, vol. 5, no. 1, p. 3, 2010.

[15] B. Li *et al.*, “Hunting or waiting? Discovering passenger-finding strategies from a large-scale real-world taxi dataset,” *2011 IEEE Int. Conf. Pervasive Comput. Commun. Work. PERCOM Work. 2011*, pp. 63–68, 2011.

[16] B. Moufida and A. Djamel, “T Hermal M Odeling of U Rban M Icroclimate ,” *Direct*, vol. 4, no. June, pp. 647–652, 2011.

[17] “Ambient Intelligence,” 2005.

[18] C. Yang and E. Gonzales, “Modeling Taxi Trip Demand by Time of Day in New York City,” *Transp. Res. Rec. J. Transp. Res. Board*, vol. 2429, no. 14, pp. 110–

120, 2014.

[19] J. Yuan, Y. Zheng, L. Zhang, Xi. Xie, and G. Sun, “Where to find my next passenger,” *Proc. 13th Int. Conf. Ubiquitous Comput. - UbiComp ’11*, p. 109, 2011.

[20] J. Grinberg, A. Jain, and V. Choksi, “Predicting Taxi Pickups in New York City,” 2014.

[21] “New York City - open dataset @ www.nyc.gov.” New York City.

[22] “Chicago - open dataset @ data.cityofchicago.org.” .

[23] “Weather dataset @ www.weather.gov.” .

[24] “Reservoir Sampling @ erikerlandson.github.io.” .

# 6. CD Content

- DataAnalysis
  - Chicago-2016-01
    - HeatMaps
      - dataPerDay.txt
      - heatMap1.html
      - heatMap2.html
      - heatMap3.html
      - heatMap4.html
      - heatMap5.html
      - heatMap6.html
    - HM
      - Screenshot_1.png
      - Screenshot_2.png
      - Screenshot_3.png
      - Screenshot_4.png
      - Screenshot_5.png
      - Screenshot_6.png
    - Pickups.png
    - pickups_dayofweek_percentage.png
    - pickups_hourofday.png
    - precipitation.png
    - temperature.png
  - NYC-2014-01
    - Same as previous folder
  - NYC-2015-10
    - Same as previous folder
  - NYC-2016-06
    - Same as previous folder
- Project
  - TaxiDemandPrediction
    - Data
      - yellow_tripdata_2014-01_reduced.csv
      - green_tripdata_2014-01_reduced.csv
      - weather_nyc_2014-01.csv
    - DataAnalysis.py … file with method for plotting, analysis, etc.
    - DataProcessing.py … file for data processing, normalization, etc.
    - PredictionModel.py … file for creation of model, with parametrization
    - Utils.py
    - TaxiDemandPrediction.py … main file, containing basic overview how the program works
    - TaxiDemandPrediction.pyproj
  - TaxiDemandPrediction.sln
- Thesis.pdf
- Thesis.docx