Optimal Taxi route prediction in Porto

IDS 494: Python for Data Science

Team:

Abhilash Chenreddy

Chiara Poletti

Prathyusha Dasari

**Introduction**

In today’s on-demand economy, optimizing travel and transportation routes is the key. If you are transporting people, delivering food from restaurants, or picking up and dropping off courier packages - the logistics of managing an on-demand fleet is all the same.

Generalizing even more, optimizing the shipments and all the transports has a huge impact on the financial health of people and businesses. Both for individuals and businesses like Amazon, it is crucial to find a way to optimize route and transport. Businesses like Amazon or Walmart, heavily depend on the efficiency of their shipments for their financial performance.

**What is Route Optimization?**

Route Optimization is the process of determining the most efficient route. It's more complex than simply finding the shortest distance path between two points. It needs to include all relevant factors such as the traffic congestion for the current time of the day and if the day is a holiday etc.

This tool can also account for changing road conditions, weather conditions, and other valuable points of data, which are customizable based on business needs and priorities. Only data fed into the tool will determine what are the limits and the features of it.

This means that routing isn’t just limited to point A to point B decisions, but to the whole gamut of variables that may influence routing.

Through the use of routing technology, companies and fleets can now see where there are school zones, road construction, and traffic delays along planned routes and then adjust for quicker navigation and reduced travel time.

**What’s the most expensive or dangerous mile?**

The mile you didn’t need to run — if we can eliminate that mile, we can eliminate the fuel used, the labor cost, and the accident — the elimination of unnecessary miles drives so many benefits — fuel savings, labor savings, productivity improvements, customer satisfaction, and it’s also the safest mile, because you’re not doing something that shouldn’t be done.

With the current dataset that we have, we are limited only to take into consideration the time of the day and type of the day (holiday or not). But this tool has a scope of adding additional features as well. While being able to recommend the optimal route between point A to point B routes, our tool can also allow for optimized, dynamic routing that:

• Takes into account variables such as real-time traffic conditions, road repairs etc.

• Improve productivity and safety.

• Integrates with other datasets.

**Optimal route for taxi trips**

One of the most frequent means of transportation in cities for passengers is taxis. But we cannot be sure that we can reliably predict which would be the best route to take for a taxi trip at a given time of the day.

One may think of using Google Maps to generate a prediction for the optimal taxi route, however, the optimal route between two points might not be the same based for different means of transportation. For example, most European cities, including Porto, have taxi reserved lanes downtown. Even just considering this, we can see that But Google Maps is a great tool but it's too generic. Like most of the off-the-shelf tools, it is great at doing one task, but it is not easy to customize it for a slight variation of the task it was created for. It doesn’t consider taxi-specific aspects of the trip, so may not provide an accurate recommendation of the route and duration.

Also, the prediction of the duration of a trip done by a generic tool like Google Maps doesn't take into account specific aspects of a trip by taxi. For example, in a taxi ride, you typically have some additional time slowdowns at the beginning and at the end of the ride when you have to explain your destination to the driver or you have to pay.

Therefore, we need a tool that:

1. Can deliver a highly customized route (e.g. by vehicle, type of day, etc.).

2. Is highly generalizable, scalable, and customizable to fit the specific needs of a business or an individual.

3. Can be used in variety of use cases (e.g. An individual who doesn’t want to be overcharged, the taxi driver, the company managing the taxis, etc.)

4. Delivers reliable and stable results

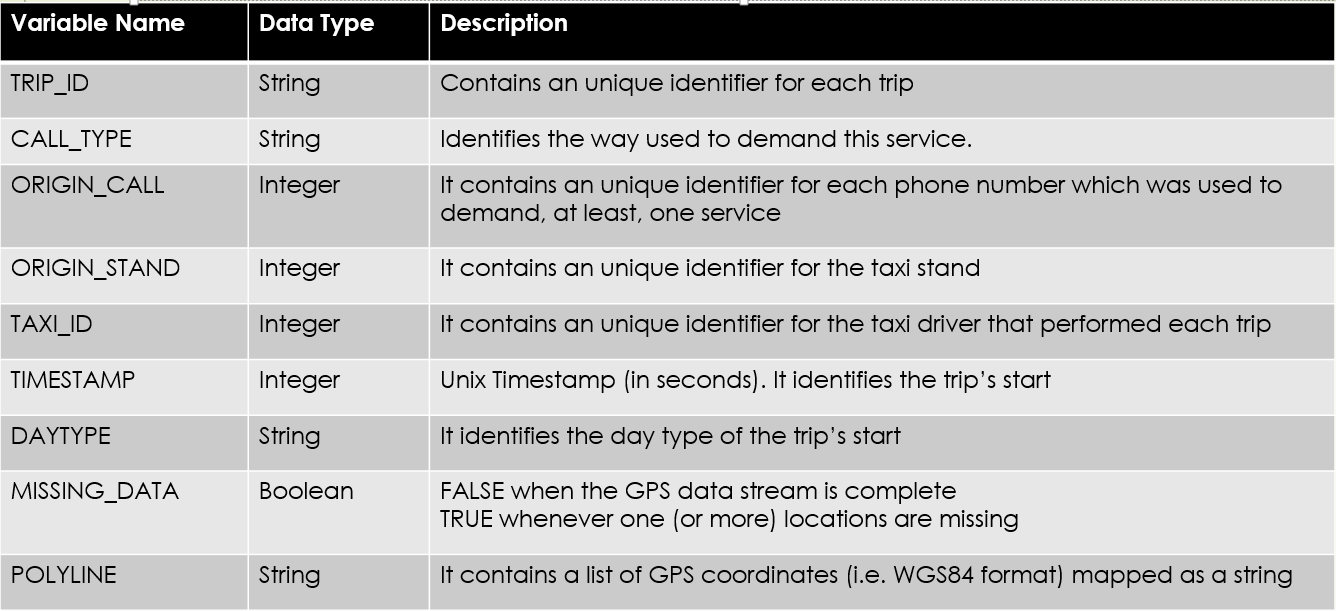
5. To provide accurate optimizations for a very specific type of vehicle or for any other kind of customization of the trip (like the fact that it is a bank holiday or not).

In this project, we will discuss the results of this type of tool used for the optimization of taxi routes, but again, this is just a general framework that can be applied to a variety of other route optimization tasks.

**Dataset Description:**

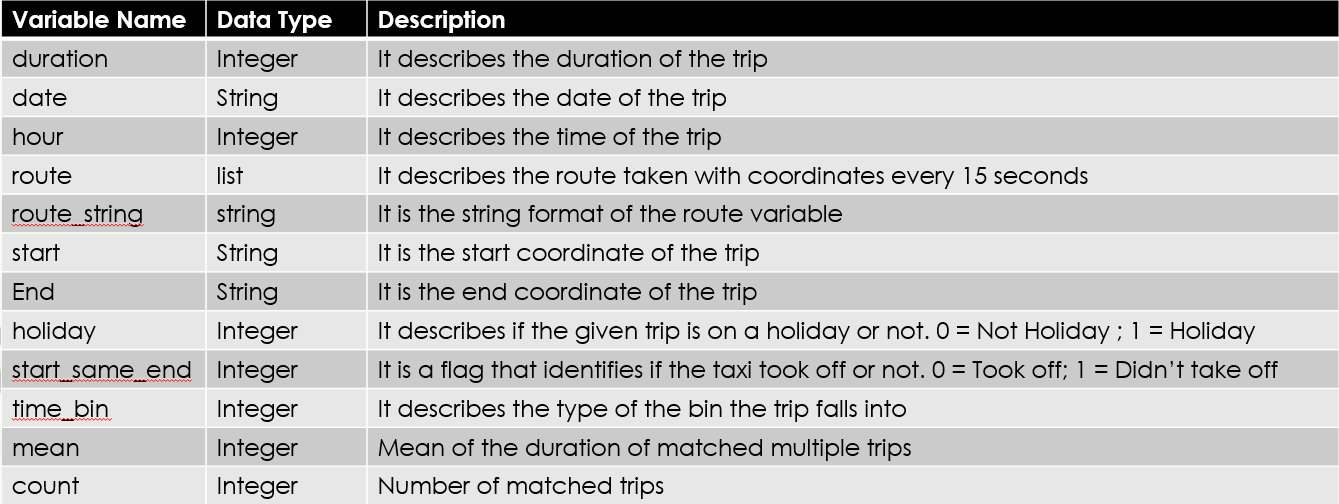
As the primary aim of this project is to suggest a route from a starting to an ending point based on the time of the day, we searched for datasets that have the trajectory information. The taxi data for the city of Porto, Portugal is one such dataset which has the information of timestamp and polyline for each trip. The flowing are the attributes contained in the dataset we found. We modified some attributes and removed some in order to make the dataset usable for our purpose.

For example, the dataset has attributes like call\_type, origin\_stand which give us the information about the origin of the ride request and the taxi stand information respectively. This information is of no use for our purpose as we are just interested in the start point, end point and the trajectory in between, irrespective of the nature of the ride. Hence these attributes are removed. Also, the attribute daytype identifies the type of the day on which the ride has taken place, which can be quite useful for our analysis. But in the metadata, it is specified that this is not an accurate information. Hence this attribute has been removed and will then be replaced by a manually computed feature that shows whether the day was a business day or not.



After removing the above said attributes, the following new attributes are created. The start and end coordinates are extracted from the poyline. The coordinates in the polyline are 15 seconds apart. Using that information, we have calculated the duration from the start point to end point of each trip. The polyline is converted into a list of coordinates (array of 2-dimensional arrays) and is named as a new attribute “route”. This will be useful for all the processing steps done on this variable that are described below (eg. rounding of the coordinates). Also the routes where the start and end points are same and the time duration is less than a threshold are removed as they are considered as outliers and noise (maybe deriving from the recording of the “trip” in moments where there was actually no trip as the taxi did not really move from start to end). In order to identify them easily, a dummy attribute, start\_same\_end is constructed which takes the value 1 when true and 0 when false.

Besides these, variables holiday, time\_bin, mean and count are also created whose significance and functionality are explained in the next section. The below tables display the variables, their data types and their description.



**Data Handling:**

Some more operations are performed on the given attributes to extract further information. For example, date and hour are extracted from the time stamp. It is our understanding that traffic differs significantly on a holiday compared to a regular weekday. So identifying the day type is another important task. For this, using web scraping methods, we have obtained the information about all the days in the years 2013-14 to identify the weekend. Following this, we have compared the date we extracted with the dates in the above extracted list. Days that are non-business days, are flagged as “holiday” with a value 1 in the corresponding dummy variable.

Following this, time bins are created. In order to have a rough first idea of the number of bins that would be ideal to set, we plotted the hour of the day (24 hours)versus the average duration. By observing this graph, initially 9 time bins were considered. But as our target here is to suggest an optimal route based on the previous observations, it would be better to get multiple matches for a set of start points, end points and time bin. When using 9 time bins, there were cases where only a single trip existed for a set of points. To combat this, the time bins were re-organized and 4 final bins were formed. Below is the description of each time bin.

Bin 1 – (6-10 hours & 16-20 hours) - Non Holiday Days – Rush hour

Bin 2 – (10 – 16 hours) - Non Holiday Days – Not rush hour

Bin 3 – Non Holiday Days – Night

Bin 4 - Holiday Days

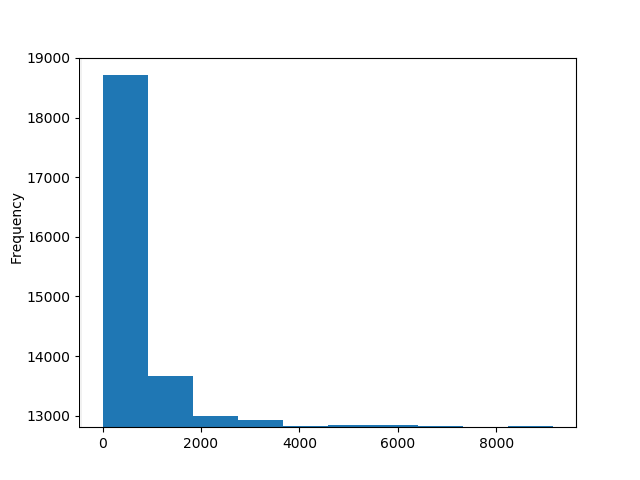
**Removing Outliers:**

The data set obtained is cleaned further so as to get better results. The dataset has a column called missing\_data which is true if there is any missing point in the polyline (set of coordinates). So all the observations where this attribute has the value “true” are removed since they would introduce only noise and potential mistakes in the predictions.

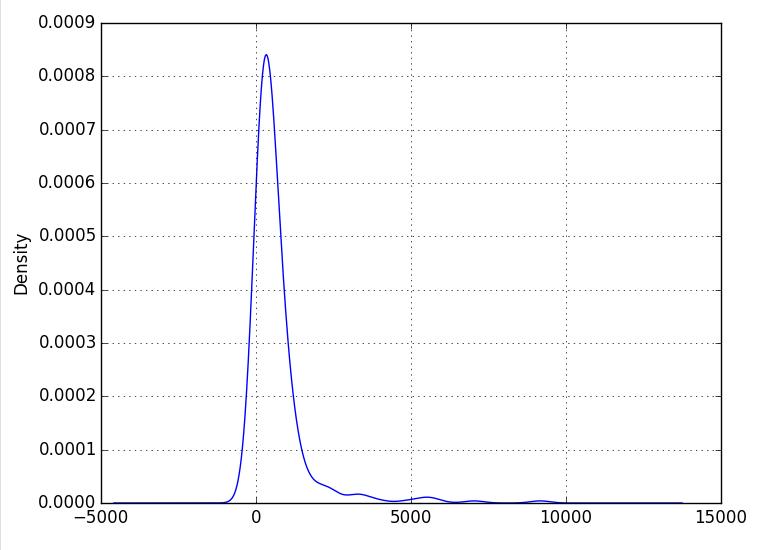
We have also created a new variable start\_same\_end which denotes the trips that did not take off or the ones that have same start and ending points . Observations with such trip polylines are removed.

As we need at least two trips between a set of start and end points to compare and suggest an optimized route, those routes that had only one route doesn’t serve our purpose. Hence such observations are removed as well.

**Spread of duration across trips**



Trip Duration



In order to understand the nature and distribution of the trips, the above graphs are plotted. The first graph shows the frequency of trip duration and the second one shows the density of the same. Few interesting observations can be made here which helped in cleaning the data further. As can be seen, the majority of the trips had duration under 500 seconds but there are some observations where the durations are in the ranges of above 8000 sec. Such observations are considered outliers and are removed.

**Descriptive statistics post data cleaning & pre-processing**

Total number of rows (trips) used - 1,648,110

Duration of trip:

Mean - 731

Standard Deviation - 661

Minimum - 30

1st Quartile - 420

2nd Quartile - 615

3rd Quartile - 870

Maximum - 58,200

Percent of Non-business days - 28%

**Analysis:**

Now that the data is processed, the following analysis is done in order to get multiple routes between start and end points. The usual way to proceed is to divide the city into grids of convenient size and consider the coordinates in each grid as a single observation. We have employed a new method to achieve a very similar effect. We have tried and reduced each coordinate to a certain decimal digits, which effectively would produce the same result as forming grids. In order to decide on the ideal number of digits to which each coordinate should be rounded off, we have iterated on a range of digits on a sample set and calculated the distance of the farthest point to the new rounded off point. It was observed that for 3 decimal digits, this (worst case) distance turned out to be about 160 meters, which is a reasonable assumption.

After deciding on the number of decimal points, it is applied to all the data points in the train dataset and the following steps are applied:

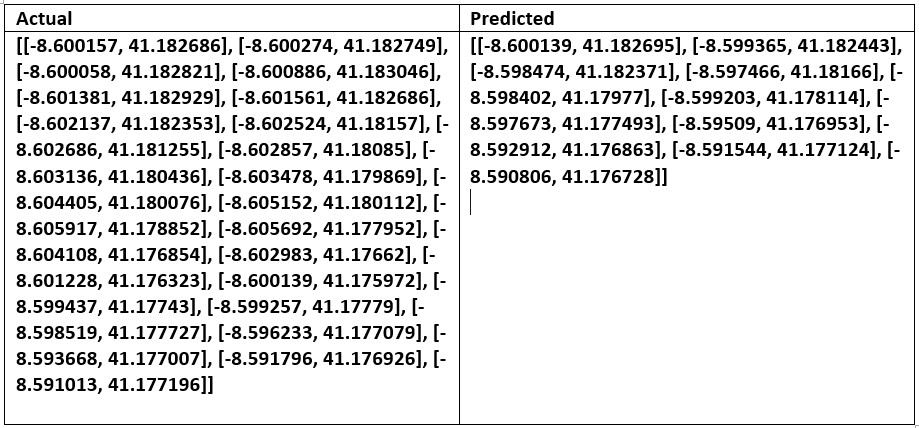
**Step 1:** From the timestamp of the trip, the date and hour of each trip is extracted.

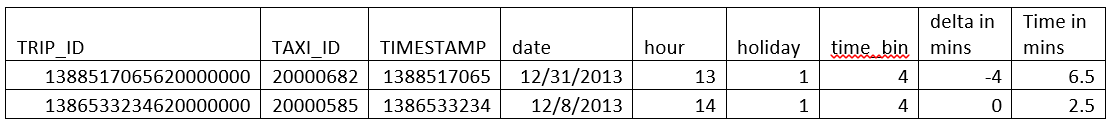
**Step 2:** Depending on the date, the type of day i.e., holiday or non-holiday is determined. And then the trip is categorized into one of the four bins that we have created.

**Step 3:** Search for a trip with start point, end point and the same bin type as search variables and all the routes that match this criterion are considered.

**Step 4:** The above steps result in a data frame that has the start and end points, date and time and time bin along with all the routes that are satisfying the above conditions. Using the above dataframe, when a new test observation comes, it is looked up and the route with shortest duration is suggested as the optimal route.

Below, is a demonstration and result of one such trip.



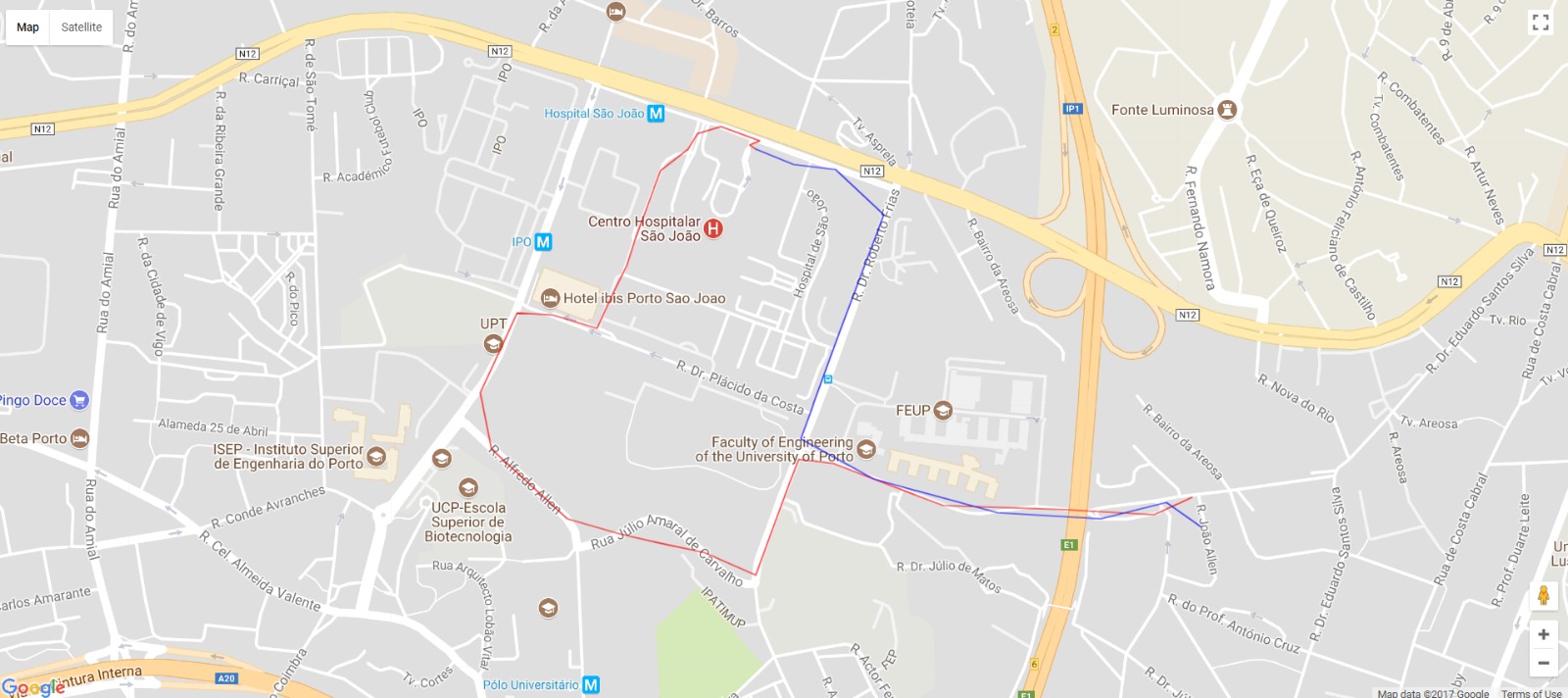


From the table above, we can see that the actual trip took place on 12/31/2013 which is a public holiday in Porto. And the trip took place at 13 hours. Therefore, this trip falls into the category of Bin 4. After looking for a similar trip with same start and end points that occurred on a holiday we got the optimal route which has a duration of 2.5 mins, while the actual trip has a duration of 6.5 minutes. Therefore, by using the suggested route, we can reduce the duration for the route by 4 mins.

For illustration purposes, in the tables above we reported all the variables that were taken into account in the prediction. The actual trip to be optimized is in the first row, while the optimal route is shown in the second row. The bigger table above shows the full sequence of coordinates in the actual trip and compares it with the same sequence of coordinates of a trip corresponding to the optimized route (right quadrant above). Specifically, we first rounded the coordinates to find a match from the new trip into the optimized routes table, and then we did a step backwards and obtained the “true” trip that corresponded to this rounded optimal trip. So, in the right side above we can see the coordinates of a trip that actually took place in the type of day and hour corresponding to the new trip to optimize, but that took a shorter amount of time and thus was deemed the optimal route for this combination of start point, end point, type of day, and hour of the day.

Now that we have the sets of coordinates for these two trips (the one to be optimized, and the optimal one), we also wanted to give a graphical representation to see if out tool was actually working and leading to reasonable results. In the map below, the red line indicates the actual route and the blue line indicates the optimal route.

**Result - Optimized Route**

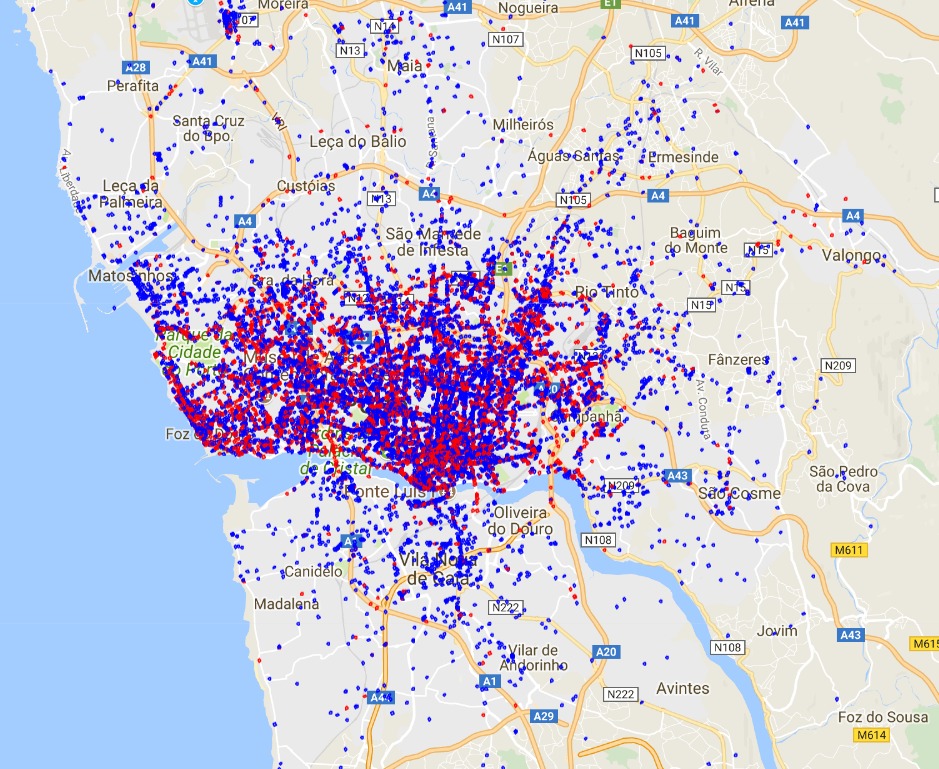


**Red – Actual Route**

**Blue – Optimized Route**

From the map above, we can see that the actual trip took basically a detour compared to the optimal route, and resulted in a very significant increase in duration as mentioned above. This map is obviously just an example of the capabilities of the tool. For instance, it could happen that for a trip taking place in rush hours, a path that has longer distance, could still be the optimal route leading to the shortest duration. This better route would be the one recommended by the tool since it results in a shorter trip time.

**Distribution of trips’ starting and ending points**

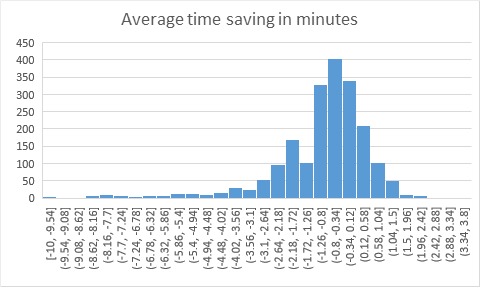


**Red – Start Point**

**Blue – End Point**

In the map above, we tried to understand whether the pickup and the drop off points are uniformly distributed across the city or not. From the map we can see that this is not the case but there are some very interesting relations between the neighborhood and the frequency of pick-ups versus drop-offs. In correspondence to the main train station of the city (middle portion of the map), there are a significantly higher number of pick-ups than drop-offs. On the contrary, in the outskirts of the city, there are almost only drop-offs. Part of the explanation for this phenomenon could be that the taxi trips considered do not differentiate between the type of ride, so it would be very reasonable that a higher proportion of the trips were starting from taxi stations. This is probably a portion of the explanation for the phenomenon represented in the map, it is also likely that there is more to it. A separate analysis would be needed here to uncover more interesting insights on this.

**Conclusion - Performance metrics**



The above plot shows the average time saved in minutes. It is evident from the plot that using this tool has saved on an average a maximum of 400 minutes by opting for the route suggested by our tool.

Also it can be seen from the plot that in most of the cases, there is a reduction of half a minute in travel time. This is a very significant result considering that the average trip duration is extremely low.

**Future developments**

* Our project is just a framework that allows us to create a generalizable and fully customizable tool. The taxi is just an example.
* Companies can derive a very significant amount of value from using an optimization tool that shows satisfactory performances and is specifically designed for their goals.
* As one would expect from any tool, also the taxi trip optimizer developed in this analysis does not have perfect performance, but this is something we'd expect from any tool. Increasing the dataset size even further, improving the quality of this data, deploying more sophisticated techniques to match different trips are just some of the potential future advancements that would make this tool even better than it currently is.