**Final report**

Analysis of bus public transport in Malaga using graphs

Learning from Networks, University of Padova, 2022-23

Jesús Moncada Ramírez, 2075280

# 1. Introduction

The **motivation** for this project was to analyze some of the main features of the bus public transport system of the city of Málaga, Spain using networks. Particularly, its frequency and speed. This analysis has taken place by creating a graph data structure with all the information about bus lines, stops, routes, and timetables… and extracting interesting characteristics from it using graph algorithms.

The initial **objectives** were seven for this project, and finally, we had time to complete four of them: (1) “Create a graph representing all the bus lines and stops…”, (2) “Compute some graph analytics at the node level…”, (6) “Do some experiments with random graphs…” and (7) “Create an analysis report with the conclusions”. This way, our work has been centered on computing graph analytics for nodes, especially **closeness and betweenness centrality**. After this, we studied some **random graphs,** following diverse models, to discover whether the centrality measures were relevant. Thus, the project title could have changed to “Analysis of node centralities and its significance in the bus public transport of Málaga”.

The **dataset** used was obtained from Malaga’s council open data site (*[Datos abiertos, Ayuntamiento de Málaga](https://datosabiertos.malaga.eu/)*). On this web page, the administration of the city provides several CSV files about the bus system organization. The availability of this data has been a very relevant motivation to choose the city of Málaga as the object of our study. We tried to get this data for the city of Padova, but it was not found.



Figure 1: Urban Buses Map of Málaga

In section 2 we will explain all the work that has been done. In section 3 we will discuss some problems we found during the project development and the proposed solutions. In section 4 we will show some results and relevant information obtained after we studied the graph. Finally, in section 5 we will present the conclusions.

The project has been uploaded to [this repository](https://github.com/jemoncadar/malaga-bus-graph.git) in GitHub to maintain higher version control and to make it public.

# 2. Work done

All the code developed for this project can be found in the Jupyter Notebook file *ProjectNotebook.ipynb.* We have used *Python 3* as the programming language and *NetworkX* as the main library to work with graphs. This file also contains some explanations, more technical details, and comments in a notebook way to facilitate its understanding. We have divided it into some parts that will be explained separately.

## Part 0: Installation issues

This section, which has no code, contains the **versions** of the libraries used that must be installed to execute all the code properly. This has been discovered by going into bugs and searching on the internet. Some of these libraries present incompatibilities among them so, in some cases, they have to be downgraded to previous versions.

## Part 1: Importing the datasets

We created the proper Python data structures for the CSV files of our dataset. For this, we have used the library *pandas*; this way the data structures are *pandas.DataFrames*, which allows us to perform a lot of already implemented operations. After importing, we renamed some column labels of the data, translating from Spanish to English. Finally, we provided a brief view of each dataset file showing some samples (using the library *tabulate* to print it nicely). In a nutshell, this section oversees data preprocessing.

## Part 2: Creating the graph

Here, we created the graph, using a *NetworkX* data structure called *MultiDiGraph* (a graph that allows multiple, directed edges between nodes). We added first the nodes (all the bus stops in the city), and then the edges (all the connections between two stops belonging to the same line). Before adding the edges, we chose a measure of the quality of each edge, this is, its weight (which has been called ***velocity***). We decided that the *velocity* of an edge (the connection between two stops) is inversely proportional to the **distance** between the two stops and directly proportional to the **number of buses** in its line (in some time interval). It is also influenced by the product by a constant that has been set empirically. The formula of the *velocity* of an edge belonging to the line , , is:

were is a constant, is the number of buses of line (in some time interval) and is the distance between the stops and . All the values we got were normalized and applied to a threshold to ensure stability.

Finally, we displayed an **image** of the generated graph to check if the elements were properly created. Two images have been generated, one of them where the edge color depends on the bus line, and the other where the edge color depends on the *velocity* attribute. Some tests were made to display our graph over an *HTML* interactive map, but finally, this feature was classified as optional. As can be observed in *Figure 2*, the generated image for the graph resembles fairly well the Málaga map in *Figure 1.*

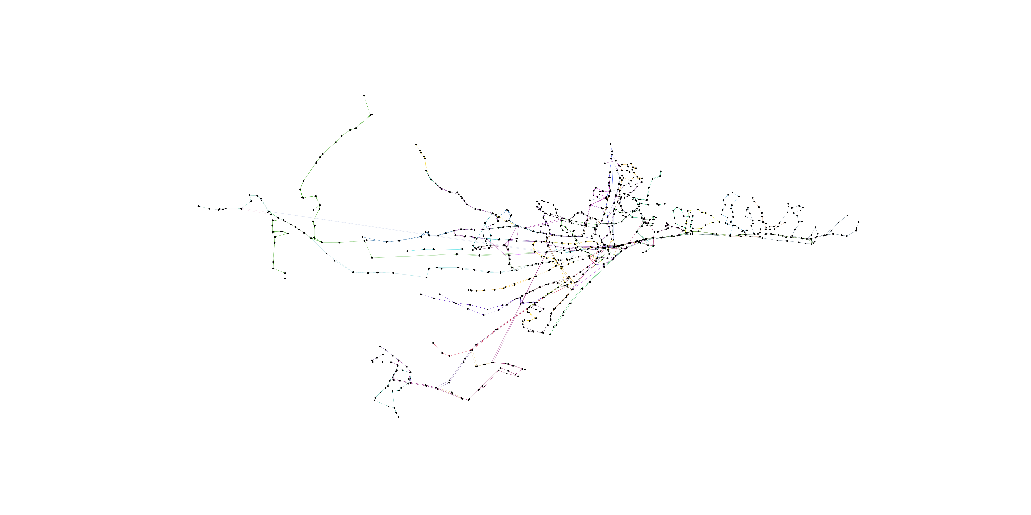


Figure 2: Line-based image generated of the graph

## Part 3: Node centralities

In this section, we calculated **closeness centrality** and **betweenness centrality** for all nodes in our graph. We had to define a new version of our initial graph as the *NetworkX* algorithms for centralities expect a **distance** weight for each edge, this is, the higher the value the poorer (further away) the connection between two stops. This way we defined the *distance* of an edge, , as the inverse of its *velocity*. If an edge was “very fast”, its *velocity* value was high, so now the *distance* will be low.

After this, we calculated the closeness centrality for nodes in two cases: (1) considering every edge weight as 1 (not taking into account the *distance* attribute) and (2) considering the *distance* attribute as weight.

The betweenness centrality for every node was also computed in the new version of the graph. But in this case, the graph was also forced to have another modification, as *NetworkX* does not implement the betweenness centrality algorithm for the graph data type. Only directed graphs (without multiple edges between nodes) support the calculation of this measure. Thus, we created a function to convert a *MultiDiGraph* to a *DiGraph*. In this conversion, we need to specify what to do with the edge attributes, as some previous edges may be converted into only one new edge. The implementation decision was to calculate the **maximum** among the *velocities* of the (disappearing) previous edges and assign it to the new edge (the other attributes were discarded as they don’t provide useful information anymore).

Again, the betweenness centrality was computed for two cases: (1) not considering *distance* and (2) considering *distance*. All the results will be commented on in section 4.

## Part 4: Random graphs

Now, to evaluate the significance of the measures obtained in the previous section, we generated several random graphs and calculated the same measures on them. These, were generated following two models. (1) The **Chung-Lu model**, this is, probability of edge to appear is:

(2) The **Given Degree Sequence model**, this is, each node will maintain the same degree in the random graphs. The generation of graphs following these two models was already implemented in *NetworkX*.

The **procedure** in both cases was, once generated the random graphs, calculate the closeness centrality and betweenness centrality for **each node** in these new graphs. With this data, for each node , we have calculated the **z-scores**:

were is the real measure for in our original graph and is the random variable corresponding to the value of feature (when the null hypothesis is true). In our case, takes the value of the centrality calculated for node in the original graph and are, respectively, the mean and standard deviation of all the centralities calculated for node in the random graphs.

Equivalently, we have calculated the p-values, specifically:

Practically, these p-values have been calculated by counting, for each node, the number of times that the centrality in a random graph was higher or equal (first case) or lower or equal (second case) than the value in the original graph.

Note that, in the case of the Given Degree Sequence model we have only studied the closeness centrality, due to problems with the implementation of algorithms for our graph data structures.

# 3. Problems found

During the development of this project, some problems had been found. In every case, a solution has been proposed.

* **Inconsistencies in the dataset.** We have discovered that the data offered by the Málaga council are not perfectly correlated. An example of this is the dataset “*lines\_and\_stops.csv”* where you can find bus lines never referenced in “*routes.csv”* and stops never referenced in “*stops.csv”*. The solution to this has been creating the graph using only the data from *“lines\_and\_stops.csv”* and using the other files to get extra information. In some cases where the absent information was needed, we simply set a default value.
* **Values of the *velocity* attribute (weight)**. After setting the attribute *velocity* for all the nodes, we found that the data contained extreme values (especially very high *velocities*). This is the case of an edge connecting two stops where important lines pass, that are very close geographically. To avoid this, we have normalized the data (against the maximum) and applied thresholding with an empirical value.
* **Library *Mplleaflet*.** *Mplleaflet* is a very interesting library to display information on dynamic maps using Python. Unfortunately, it presents bugs, and the installation is quite complicated. Finally, we managed to create the map, but this section has been labeled as optional in the Jupyter notebook of the project.
* **Algorithms not implemented.** As we have already commented, the *NetworkX* library does not implement some algorithms (especially betweenness centrality) for certain types of graphs. Every time we faced this, a new version of the graph (fulfilling the requirements) was created from the original. Anyway, the results obtained in these simplified versions were always contrasted with the original graph.
* **Computational cost working with random graphs.** Generating random graphs was not a computational problem for us but calculating the centralities on them. To solve this issue, we reduced the number of random graphs generated to 50.

# 4. Results

Firstly, we will highlight important bus stops from the city based on their centrality measures obtained, secondly, we will explain the significance of this data.

## Closeness centrality

Calculating the **closeness centrality** for every node in the graph without considering the *distance* attribute, we got that the stop with the **highest value** is *(164*, *Paseo del Parque)*. This makes sense because the “Paseo del Parque” avenue is located in the very city center, where all the bus lines converge. The majority of bus lines in the city provide a connection between a remote neighborhood and the city center. This way, somebody that lives in the east countryside can travel to the western countryside by taking only two different lines. Knowing this, it makes sense that the “more central” stop is located there. Secondly, there are several stops with the **lowest value** for this measure (0) as the library approximates very low values to 0. We have discovered that they are located in remote parts of the city, and they are usually the first/last stops of a line. For example, stop (*152, Avda. de las Postas (Lorenza Correa))* is the first/last of line 1; and stop *(351, Puerta Blanca (Gregorio Diego))* is the first/last of line 3.

Executing the **closeness centrality** algorithm but now considering the *distance* attribute outputs approximately the same results. Now, the stop with the **highest value** is *(1359, Paseo de la Farola (Comandancia de Marina)),* which is another stop near the city center. In this case, the location of the most centered stop is slightly displaced to the east of the city. The reason for this could be that, as it is well known, the western part of the city presents a higher quality and *velocity* of public transport. Considering now a quality measure, the most central stop is now displaced to the east, where the public transport does not present the same good characteristics. The nodes presenting the **lowest value** for this measure are, again, the starting/ending stops of lines, located in remote neighborhoods.

## Betweenness centrality

Calculating the **betweenness centrality** in our graph without considering the *distance* attribute, outputs interesting results. The stop with the **highest value** is *(662, Avda. de Andalucía)*. Checking the lines that go through this stop (either through code or looking at the [EMT app](https://www.emtmalaga.es/es/descargaapp)) we obtained lines: 4, 7, 8, 14, 19, 21, 23, 38, 92, A, N2, and N3. This is the stop with more lines going through it in the whole system. Again, this makes sense. Based on the definition of betweenness centrality, if a node has a high value it means that it is present in a high quantity of shortest paths between other nodes. In our context, this happens when a bus stop has a lot of line connections, as it will be present in a high number of trips. As happened in the case of closeness centrality, the nodes with the **lowest values** are bus stops corresponding to the beginning/end of lines, situated in remote places of the city.

When computing the **betweenness centralities** but now considering the *distance* attribute, the stop with the **highest value** is the same, and the stops with the **lowest values** are very similar.

## Analysis of significance

As it has been explained earlier, we have generated some random graphs and calculated z-scores and p-values, for every node, to check the significance of our measures. The results for the Chung-Lu model can be summarized in the following table. Note that, as we are not able to work with the z-score/p-value obtained for each node we will use the mean.

|  |  |  |
| --- | --- | --- |
| **Chung-Lu model** | | |
|  | **Closeness centrality** | **Betweenness centrality** |
| Mean  of z-score | -2.028 | 1.476 |
| Mean  of p-value | 0.853 | 0.289 |
| Mean  of p-value | 0.164 | 0.765 |

Looking at this data, we can conclude that the closeness centrality for a node in the original graph, on average, is 2.028 standard deviations below the closeness centrality of a node in a random graph. This information is confirmed by the p-values, as they show that, on average, the probability of finding a node in a random graph with closeness centrality higher than in our graph is 85.3%, and the probability of finding a node in a random graph with a lower value 16.4%

In the same way, the betweenness centrality of a node in our original graph is, on average, 1.476 standard deviations higher than the betweenness centrality for a node in a random graph. This is again confirmed by the p-values, as the probability of finding a node with betweenness centrality higher in a random graph is 28.9%, while the probability of finding a node with a lower value is 76.5%.

In a nutshell, what this data is telling us is that the **closeness centrality of the nodes in our graph is (on average) lower than the expectation**. An explanation for this could be the disposition of the bus stops on a city map; there are some constraints not allowing the stops to be as connected as we would like, for example, streets with different directions and blocks of buildings. In a random graph, it is more likely for two stops to be connected, as these intrinsic limitations do not exist.

Analogously, the **betweenness centrality of nodes in our graph seems to be (on average) higher than the expectation**. This fact is a desired feature for a public transport system, as we would like to maximize the number of routes to which each bus stop can access, this is, the number of shortest paths between other nodes in which a node is present.

For the Given Degree Sequence model, the results are as follows.

|  |  |
| --- | --- |
| **Given Degree Sequence model** | |
|  | **Closeness centrality** |
| Mean of z-score | -4.992 |
| Mean  of p-value | 0.993 |
| Mean  of p-value | 0.007 |

As we can observe, what we discovered with the Chung-Lu model is confirmed here. Data are even more exaggerated; the z-score, in this case, is -4.992 and the probability of finding a node with a closeness centrality higher in a random graph (on average) is 99.3% and lower a 0.7%.

# 5. Conclusion

In a nutshell, we have discovered which are the more relevant bus stops in Málaga, in terms of closeness and betweenness centrality. For this, we have created a huge graph with multiple and directed edges. The information obtained has been contrasted with the characteristics of the city and its public transport. We have also proved that the centrality measures obtained are significant, by computing the same measures on random graphs and calculating some measures of significance.

However, there is still a long way to go to elaborate a complete analysis. The next steps could be calculating some graphs level features and comparing them to other public transport systems in other cities or looking for the presence of some network patterns and motifs. Anyway, we consider that this project has been a very interesting and complete application of the theory studied in class.