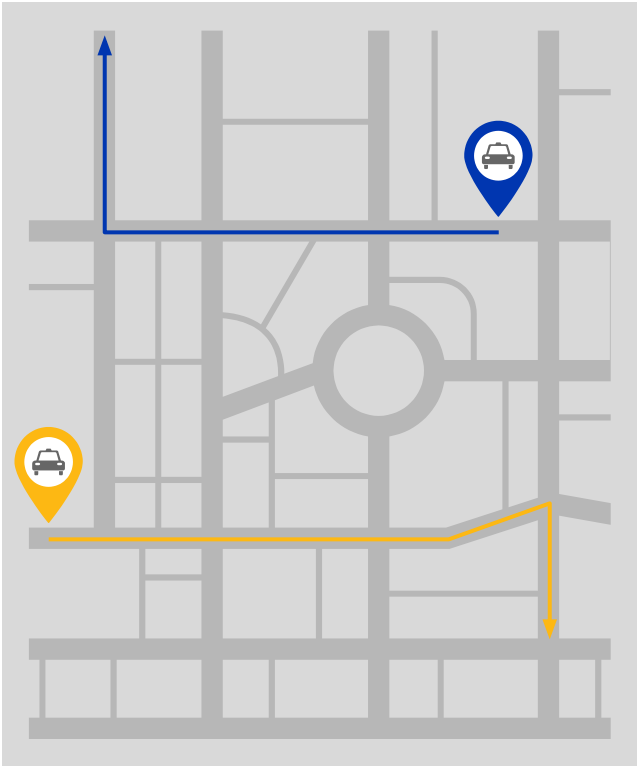
# Taxi Trajectory Analysis

## **EDAA** - G04

Diogo Rodrigues
Eduardo Correia
**João Sousa**

# Goals

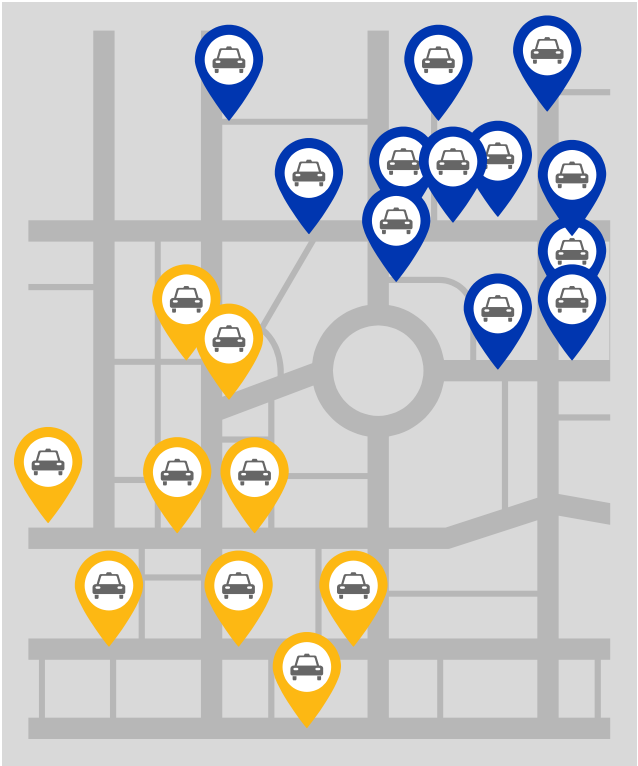
The goal of this project is to analyze **taxi trajectories**. Taking into account the **size** of the data we will be dealing with, we have to implement **efficient algorithms**.



# Problem definition

**Clustering** of the **processed data points** in the first part of the project.

**Analysis of the network** (taxi logs): coverage and other metrics.



# Data - map matching results

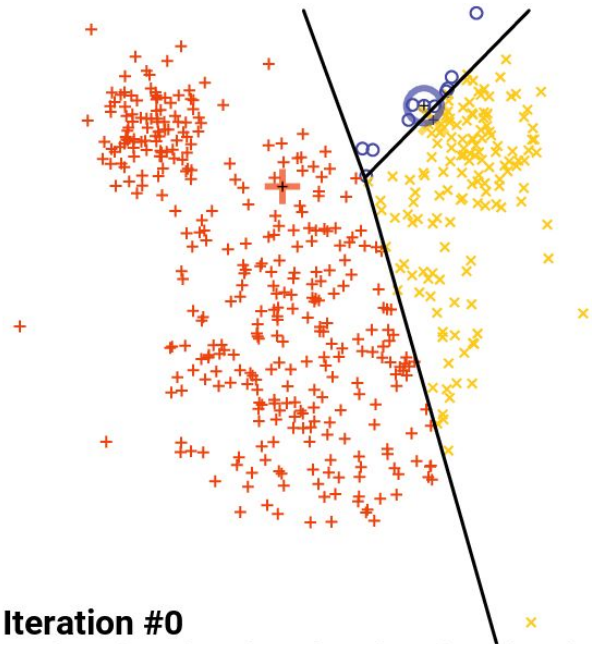
.txt file

```
1372636858620000589 23          // trip ID, N number of matches
111479505                       // matched graph nodes
9581616760
3391597627
674753639
9581592139
...
4468690341
1372637303620000596 19
9581698290
9038311044
9038733834
9581692451
...
```

# *k*-means clustering

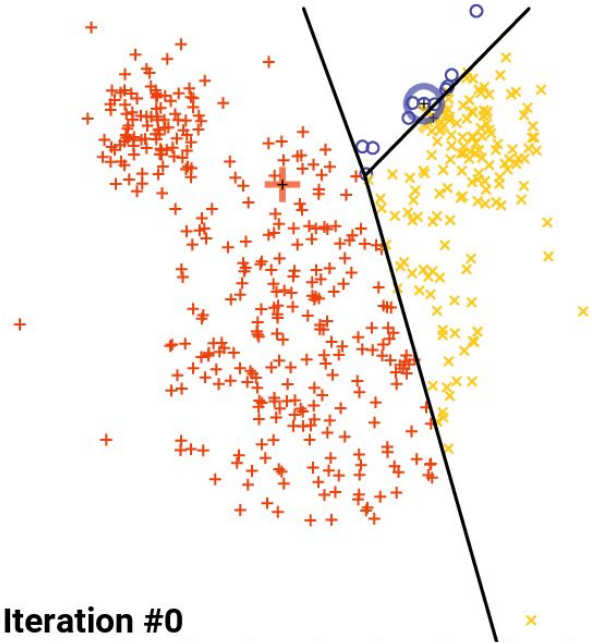
Iterative approach that **splits a set of *n* observations** into a **predetermined number *k* of partitions**.

Progressively minimizes the sum of distances between the points and their respective cluster centroid.



# *k*-means clustering

- Choose the number of clusters
- Initialize centroids (at random)
- Assign each data point to the closest cluster centroid
- Recompute the centroids of the newly formed clusters
- Repeat previous 2 steps



# *k*-means clustering

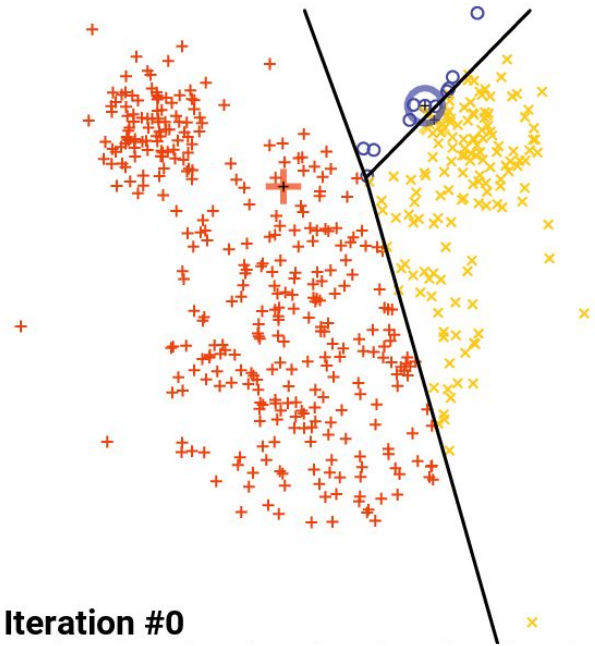
Stopping criteria alternatives:

- Centroids did not change
- Points remained in the same clusters
- Maximum number of iterations reached

Time complexity: *O(NKI)*

Space complexity: *O(N(D+K))*

N number of points, D number of dimensions, K number of centroids, I number of iterations.

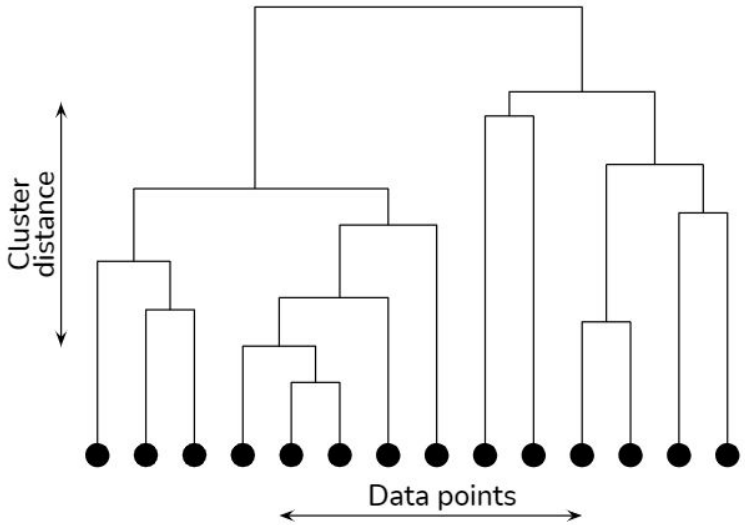


# Hierarchical clustering

Two main approaches:

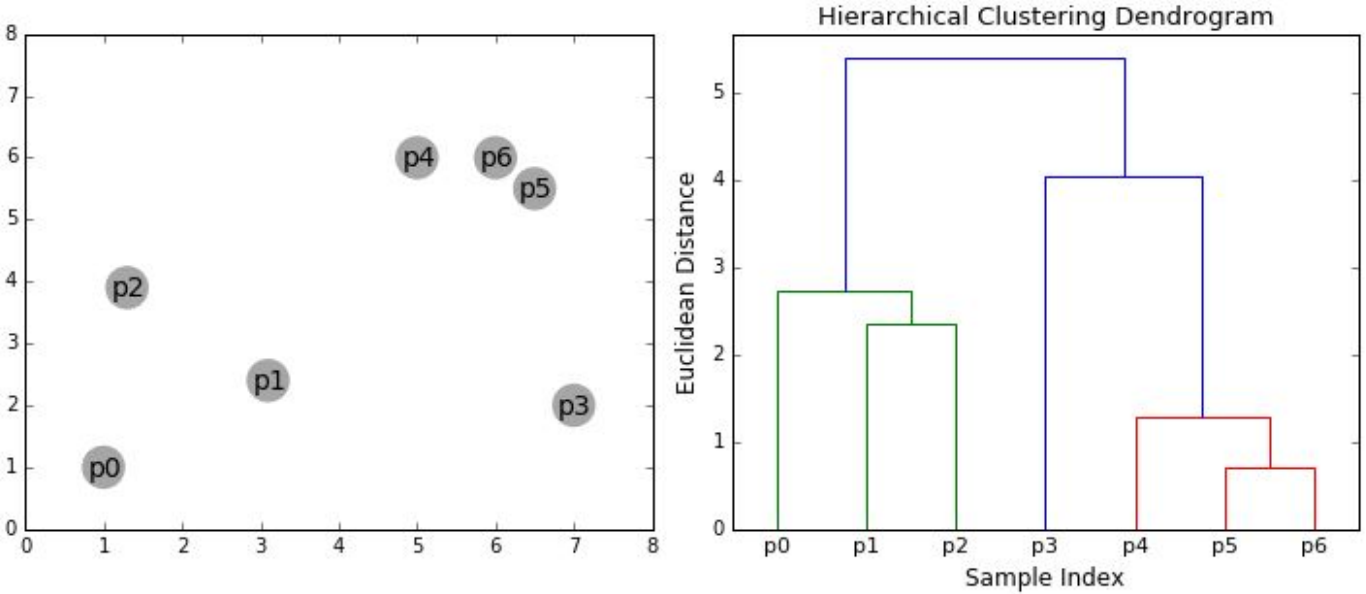
- “top-down” (divisive)
- “bottom-up” (agglomerative)

Time complexity: $O(n^3)$ but can be optimized to $O(n^2)$.
Space complexity: $O(n^2)$.




n is the number of data points.

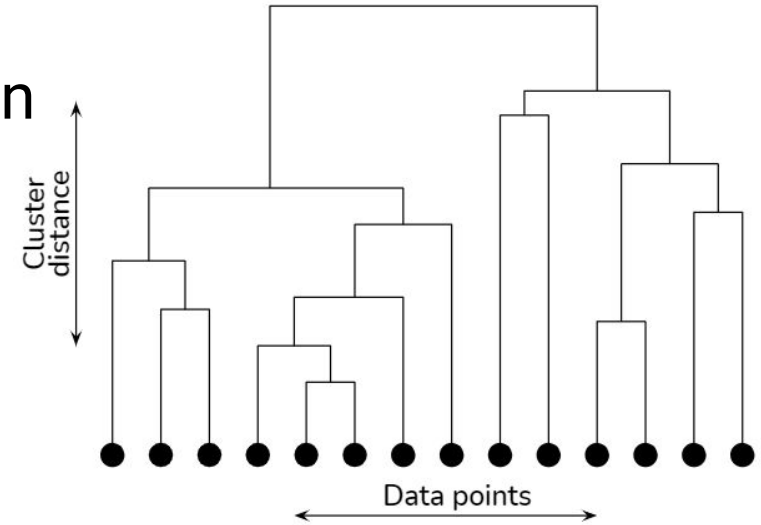
# Hierarchical clustering



# Hierarchical clustering

- Divisive: all observations start in one cluster.
- Agglomerative: each observation starts in its own cluster.

Various alternatives for metrics to choose for splitting or combining clusters: euclidean, squared euclidean, manhattan,...



# Q&A