MoF-DAC
Ministry of Finance
Data Analytics Community

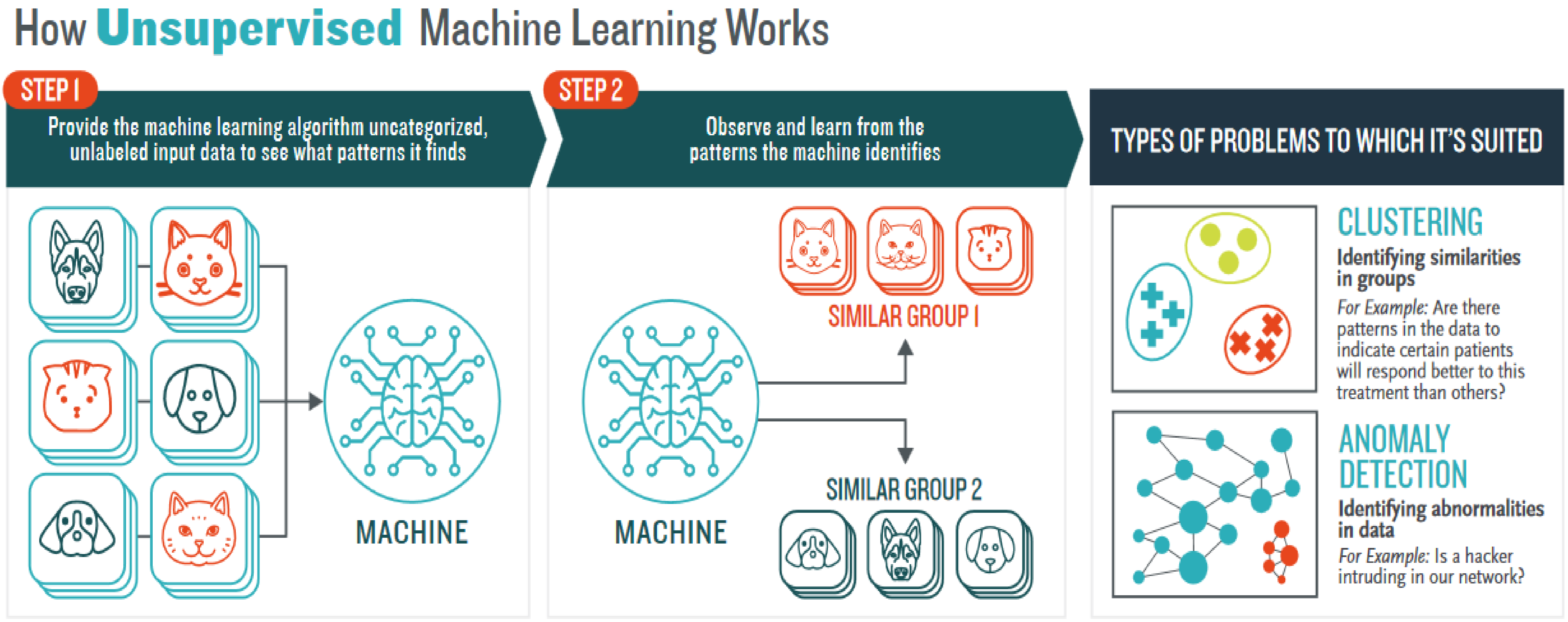
MACHINE LEARNING BASIC CLUSTERING

Sindhu Wardhana
Ade Satya Wahana
Aris Budi Santoso
Leonard Yulianus

Definition

- ***Clustering*** adalah sekumpulan teknik untuk membagi data ke dalam grup atau cluster;
- ***Cluster*** adalah kumpulan objek data
  - Memiliki kemiripan dengan antara objek dalam satu cluster;
  - Memiliki perbedaan dengan objek lain di luar cluster;
- Clustering termasuk dalam kategori **Unsupervised Learning**
  - Data tidak memiliki label/ predefined class

UNSUPERVISED LEARNING



The Advantages

Growing importance in a number of fields

- subgroups of breast cancer patients grouped by their gene expression measurements
- groups of shoppers characterised by their browsing and purchase histories
- movies grouped by the ratings assigned by movie viewers
- topic modelling of text document (NLP)

Easier to obtain unlabeled data than labelled data

The Challenges

more subjective than supervised learning

- No simple goal for analysis
- The computer have to learn how to do something that we don’t tell it how to do

Have some issues

- The number of subgroups (clusters)
- The different results via K-means with different random initialisations
- How to assess the performance of the unsupervised learning methods?

The learning (or inference) procedure is hard

- Clustering Algorithm

Clustering Approaches

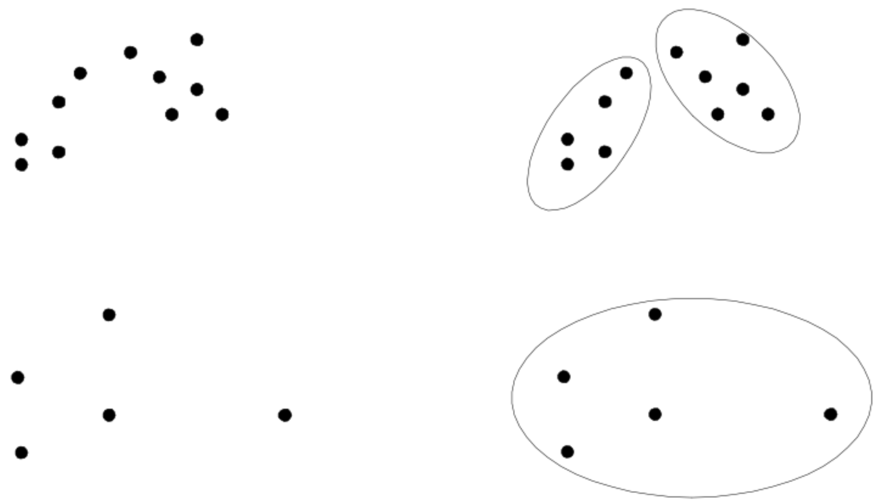
- Partitional Clustering
  - Perlu menentukan jumlah kluster
  - Iterasi untuk menempatkan data ke dalam cluster
  - K-Means
- Hierarchical Clustering

  Pembentukan cluster dilakukan secara hirarki
  - **Agglomerative :** Bottom-up

    Menggabungkan dua titik yang memiliki kemiripan ke dalam sebuah cluster
  - **Divisive :** Top-down

    Mulai dari sebuah cluser besar kemudian dibagi
- Density Based Clustering
  - Pembentukan cluster dilakukan berdasar kepadatan titik data pada suatu area;
  - Antara cluster dipisahkan oleh area dengan kepadatan titik data yang rendah;
  - Density-Based Spatial Clustering of Applications with Noise (DBSCAN)

Partitioning Clustering (K-means)



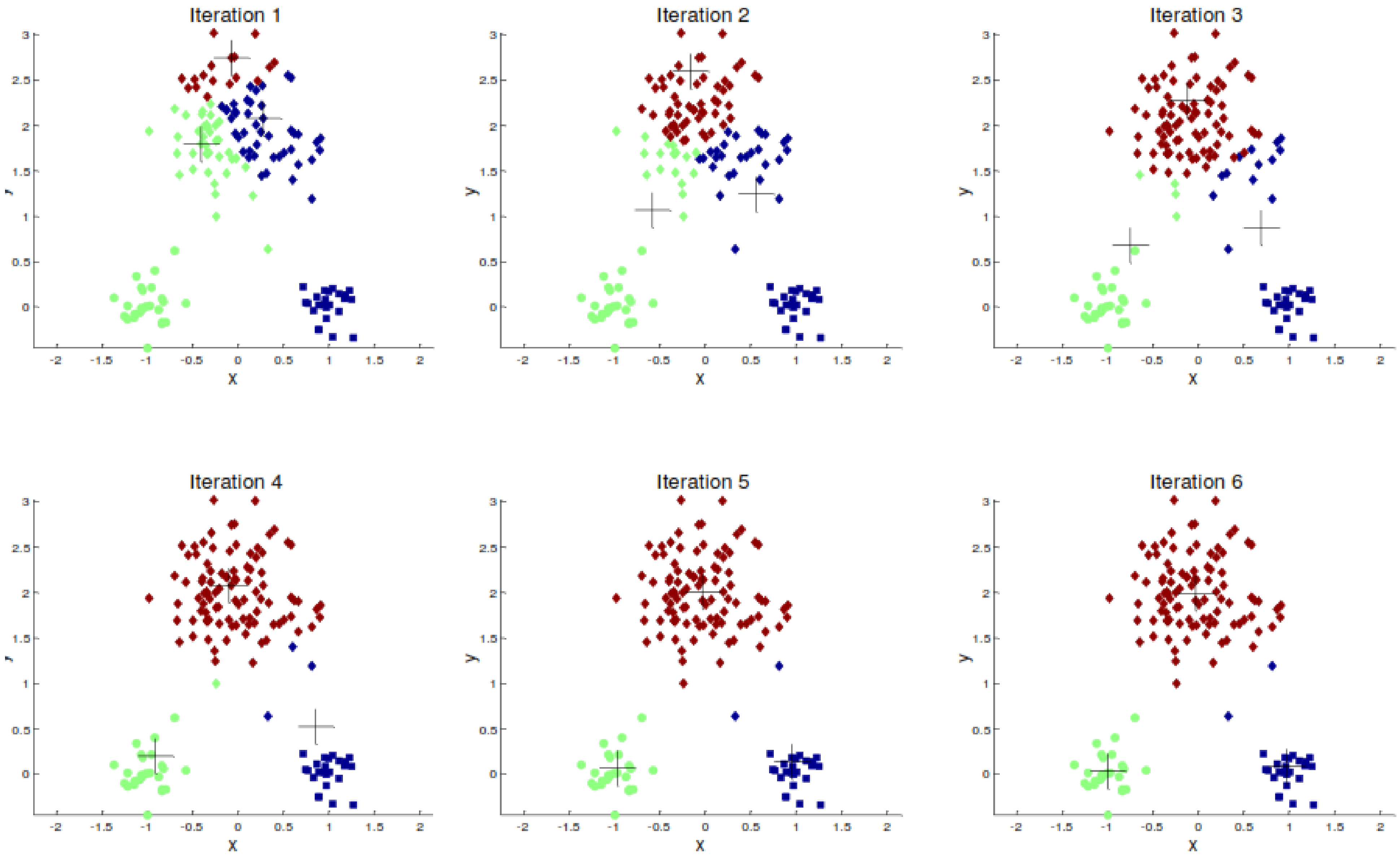
Original Points A Partitional Clustering

- Partitional clustering approach
- Each cluster is associated with a centroid (center point)
- Each point is assigned to the cluster with the closest centroid
- Number of clusters, K, must be specified
- The basic algorithm is very simple

Algorithm 1 Basic K-means Algorithm.

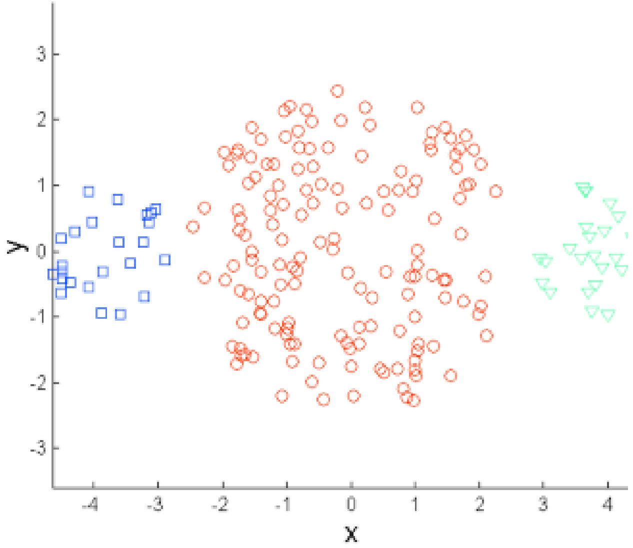
Select K points as the initial centroids.
repeat
    Form K clusters by assigning all points to the closest centroid.
    Recompute the centroid of each cluster.
until The centroids don't change

K-Means Algorithm

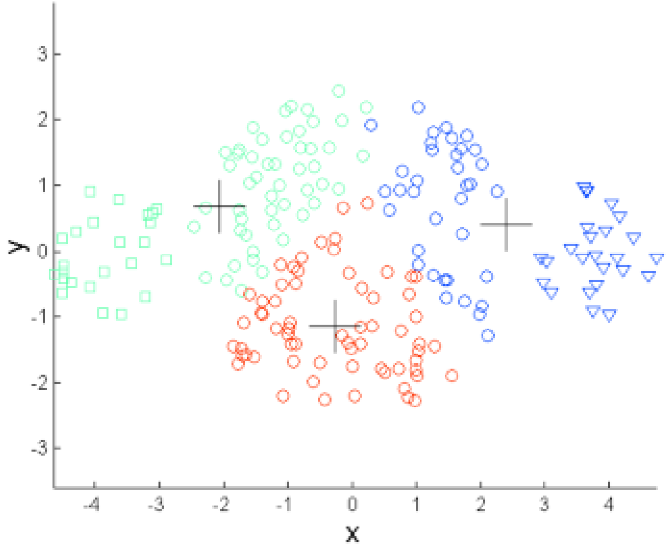


K-Means Limitation

Try kmeans with labeled data (original points)
The result is different



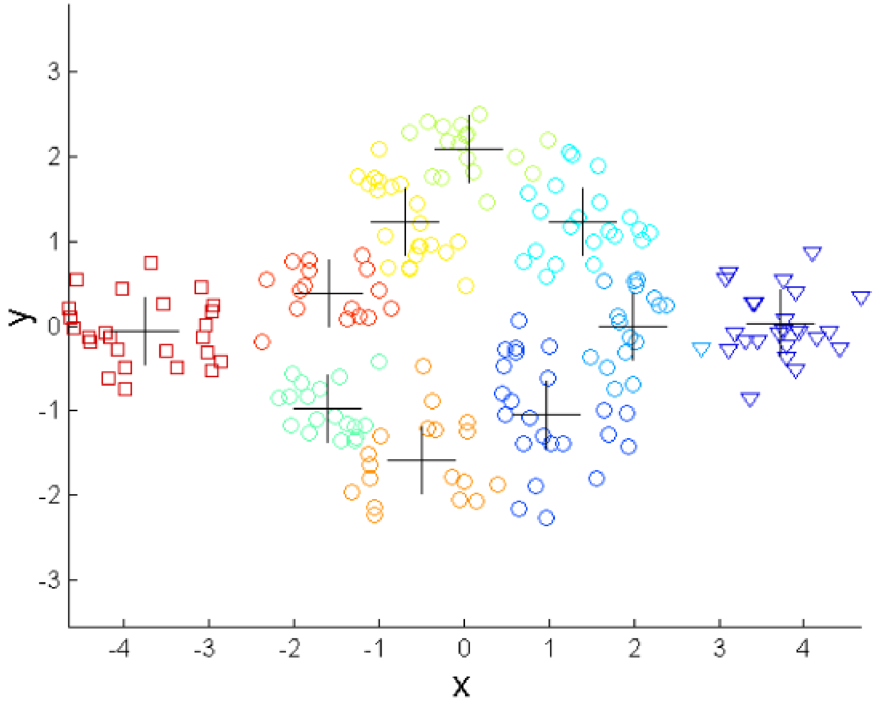
Original Points



K-means (3 Clusters)

Size difference

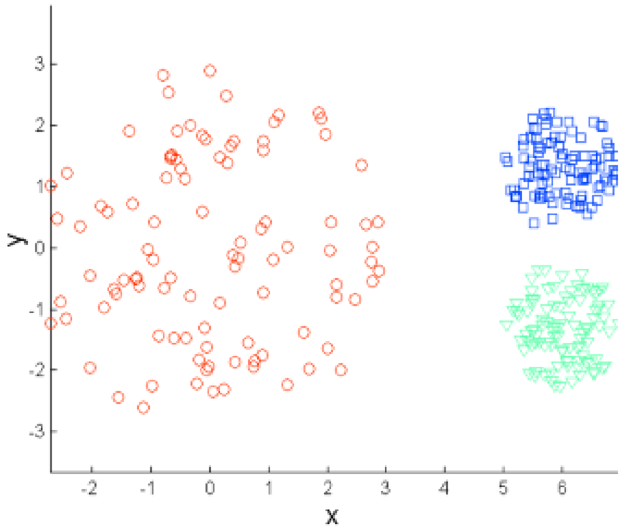
One solution is to use many clusters.
Find parts of clusters, but need to put together.



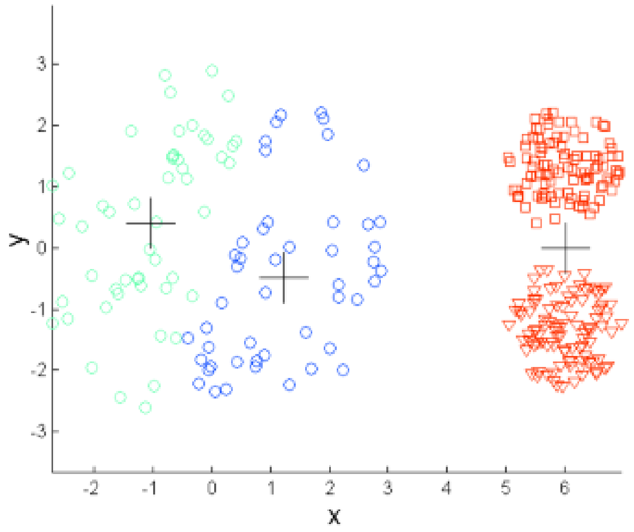
Overcoming K-means

K-Means Limitation

Try kmeans with labeled data (original points)
The result is different

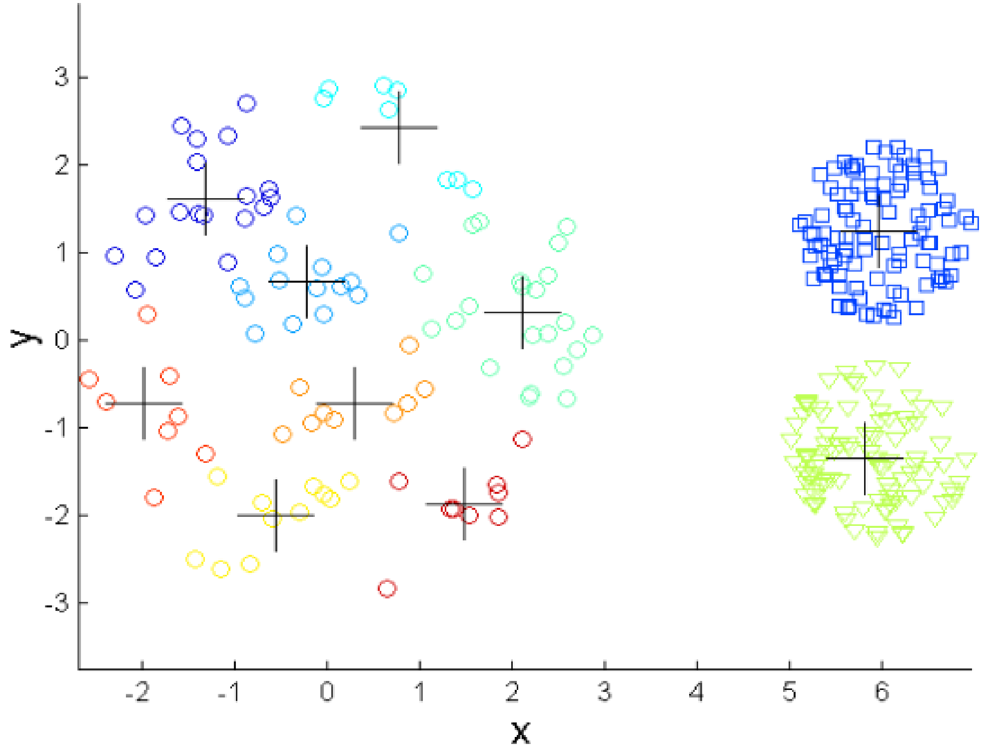


Original Points



K-means (3 Clusters)

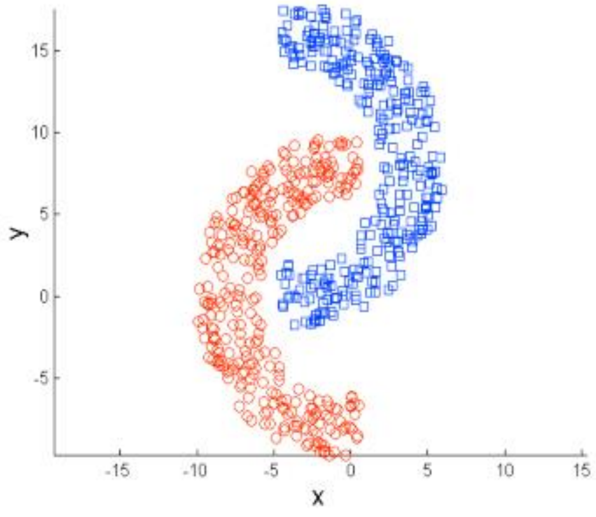
Density difference



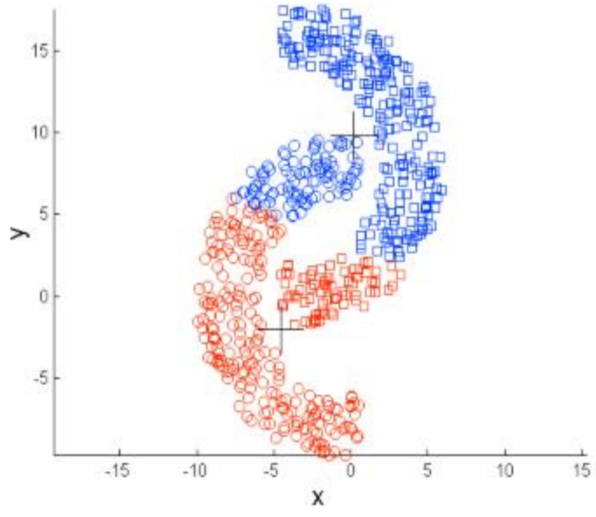
Overcoming K-means

K-Means Limitation

Try kmeans with labeled data (original points)
The result is different

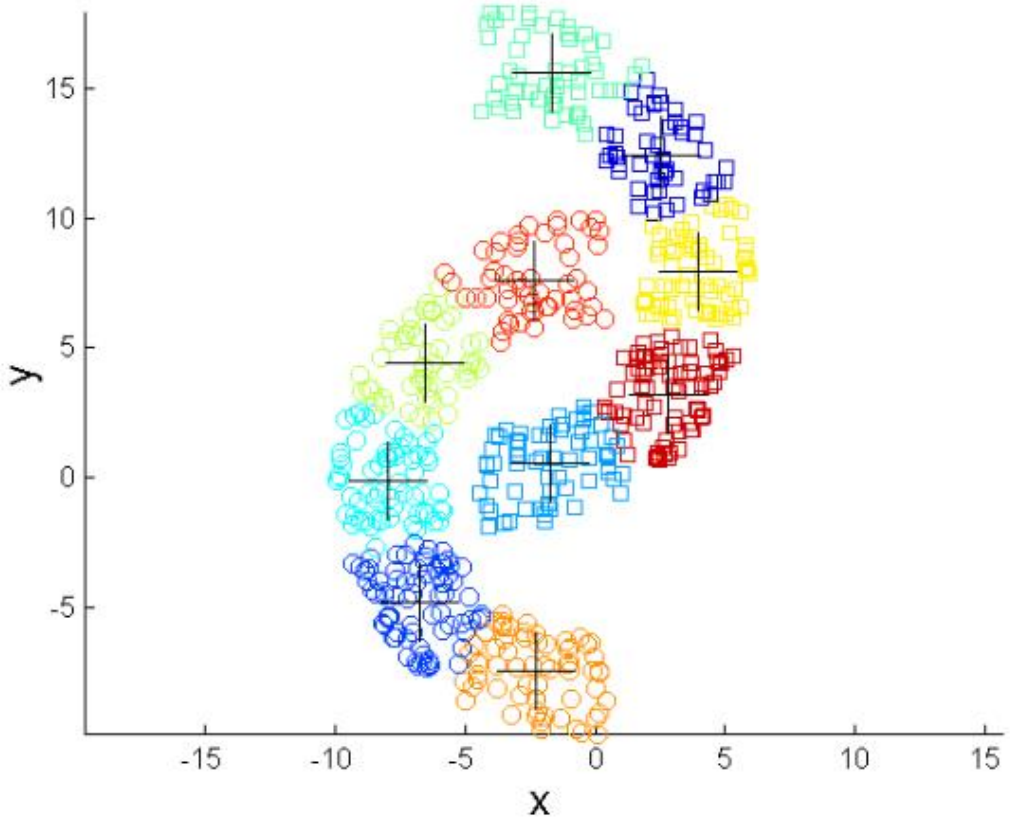


Original Points



K-means (2 Clusters)

Non-globular Shapes



Overcoming K-means

Pre and Post Processing

Pre-processing

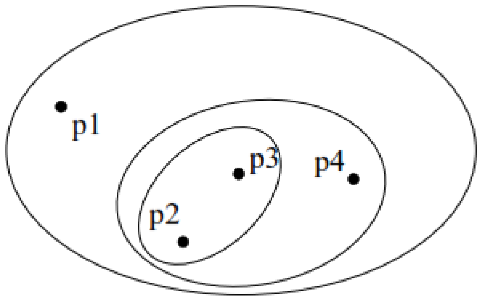
- Normalize the data
- Eliminate outliers

Post-processing

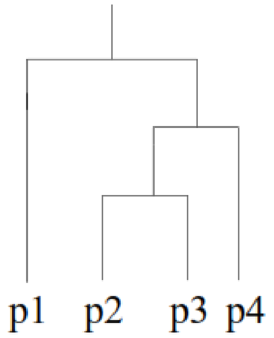
- Eliminate small clusters that may represent outliers
- Split ‘loose’ clusters, i.e., clusters with relatively high SSE
- Merge clusters that are ‘close’ and that have relatively low SSE

Hierarchical Clustering

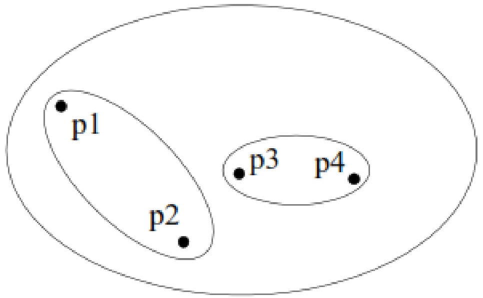
- Produces a set of nested clusters organized as a hierarchical tree
  - Can be visualized as a dendogram
  - A tree like diagram that records the sequences of merges or splits
- Do not have to assume any particular number of clusters
  - Any desired number of clusters can be obtained by 'cutting' the dendogram at the proper level



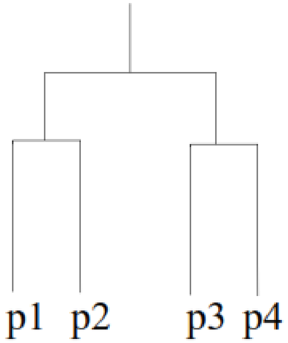
Traditional Hierarchical Clustering



Traditional dendogram



Non-traditional Hierarchical Clustering



Non-traditional dendogram

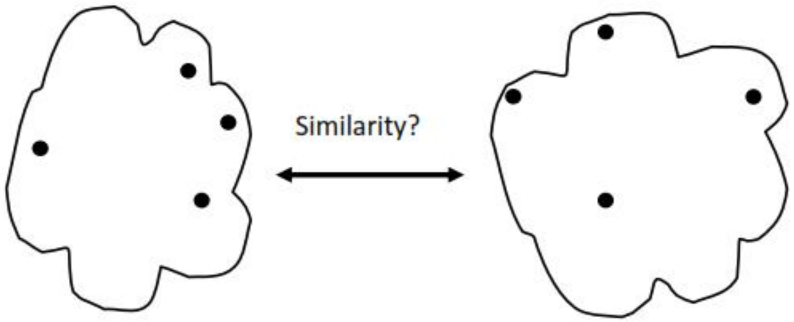
Hierarchical Clustering

Two main types of hierarchical clustering

- Agglomerative:
  - Start with the points as individual clusters
  - At each step, merge the closest pair of clusters until only one cluster (or k clusters) left
- Divisive:
  - Start with one, all-inclusive cluster
  - At each step, split a cluster until each cluster contains a point (or there are k clusters)

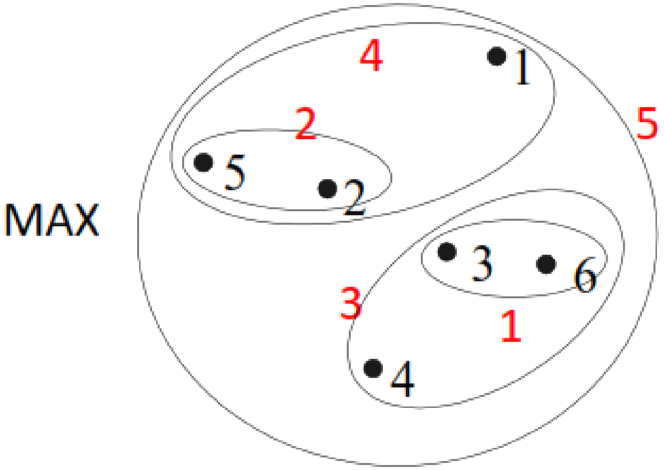
How to decide closest pair??

- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward's Method uses squared error

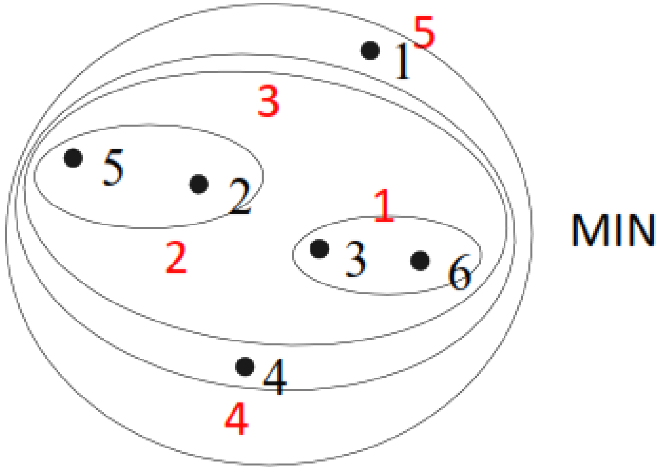


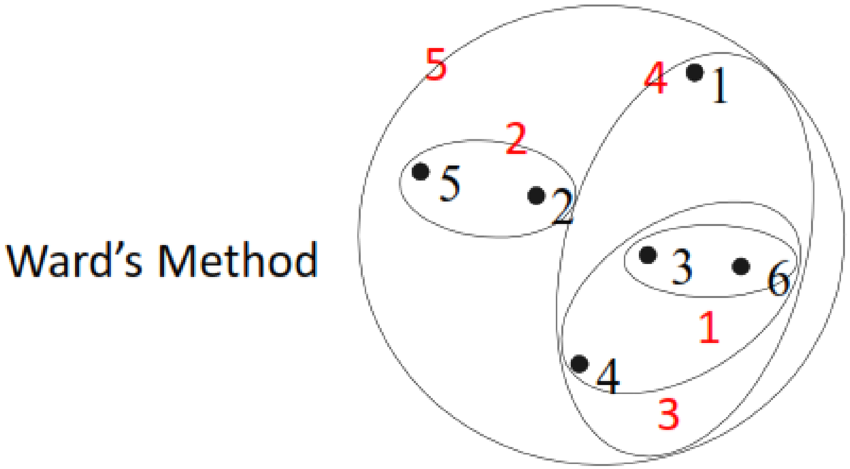
Similarity Approach Comparison

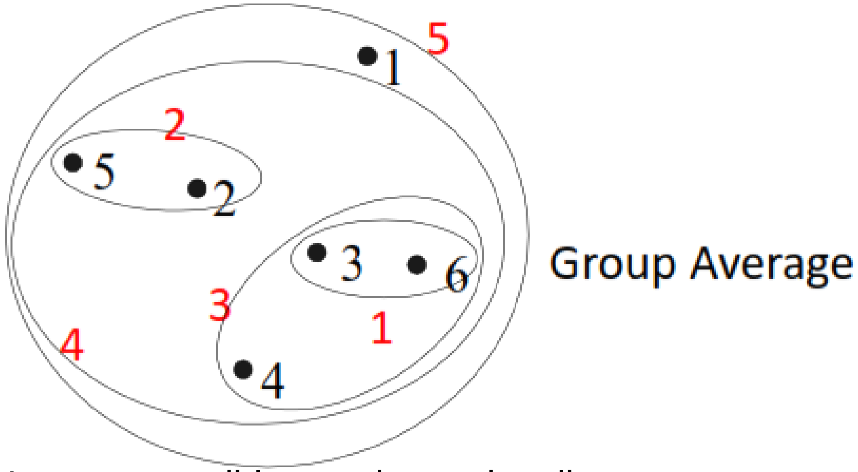
Less susceptible to noise and outliers
Bias towards globular clusters
Tends to break large cluster



Can handle non-elliptical shapes
Sensitive to noise and outliers







Less susceptible to noise and outliers
Bias towards globular clusters

Density Based Clustering

DBSCAN is a density-based algorithm.

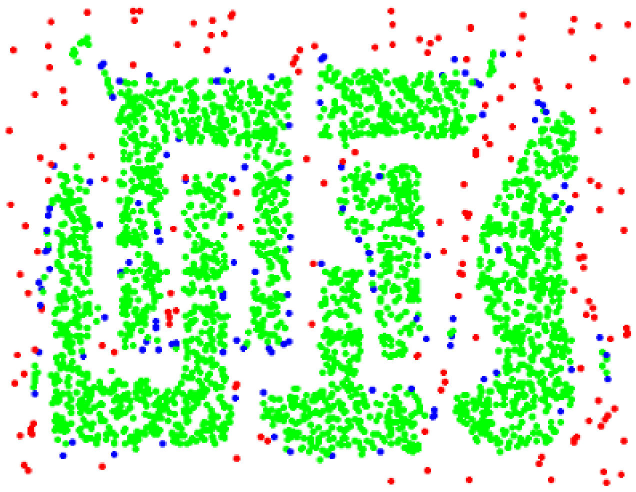
- Density = number of points within a specified radius (Eps)
- A point is a core point if it has more than a specified number of points (MinPts) within Eps
  - These are points that are at the interior of a cluster
- A border point has fewer than MinPts within Eps, but is in the neighborhood of a core point
- A noise point is any point that is not a core point or a border point.

Resistant to Noise

Can handle clusters of different shapes and sizes

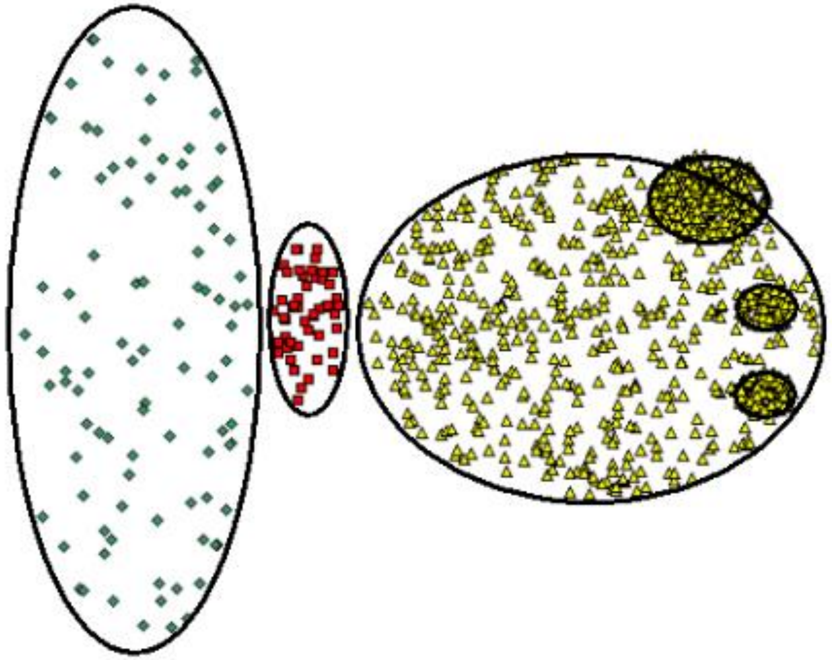
Eps = 10, MinPts = 4

Original Points



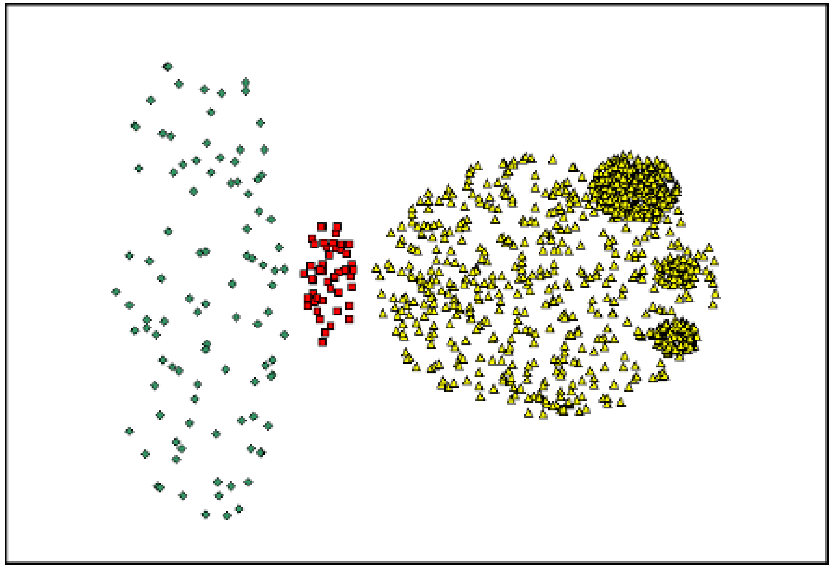
Point types: core, border and noise

Doesn't Work Well

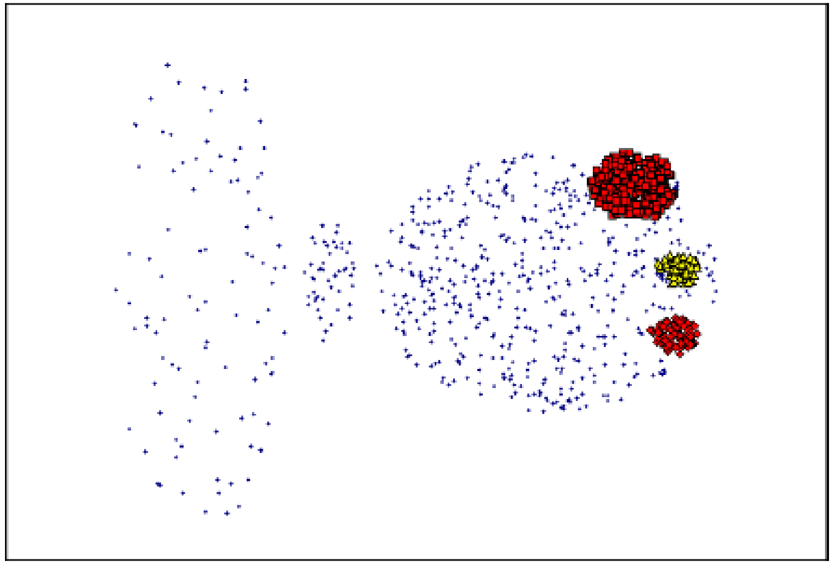


Original Points

- Varying densities
- High-dimensional data



(MinPts=4, Eps=9.75).



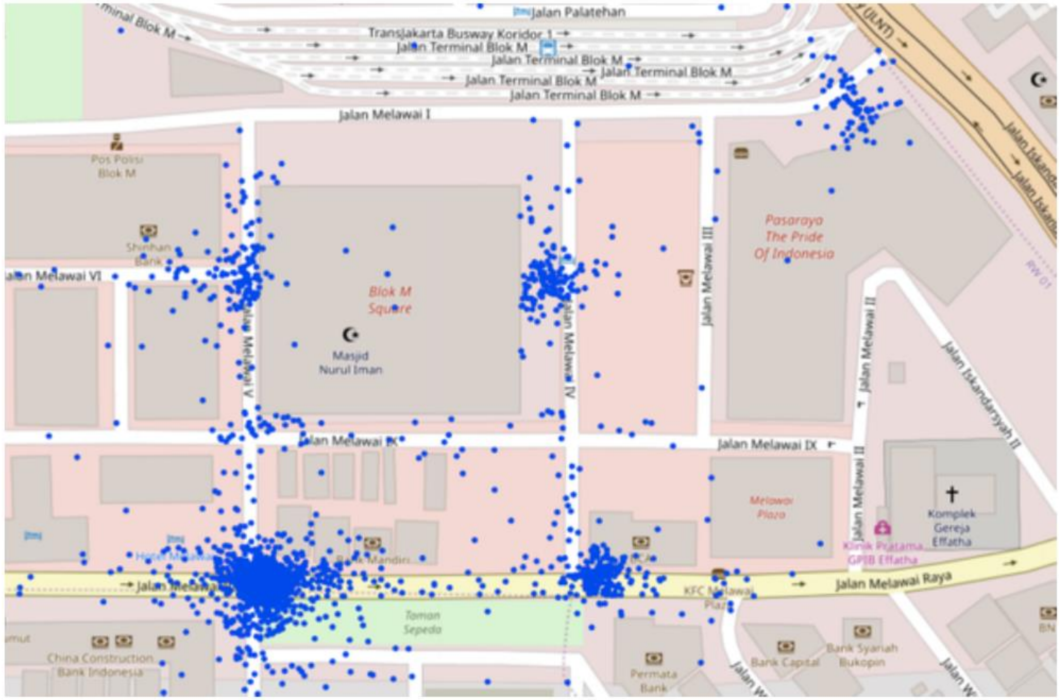
(MinPts=4, Eps=9.92)

- Use Case Clustering

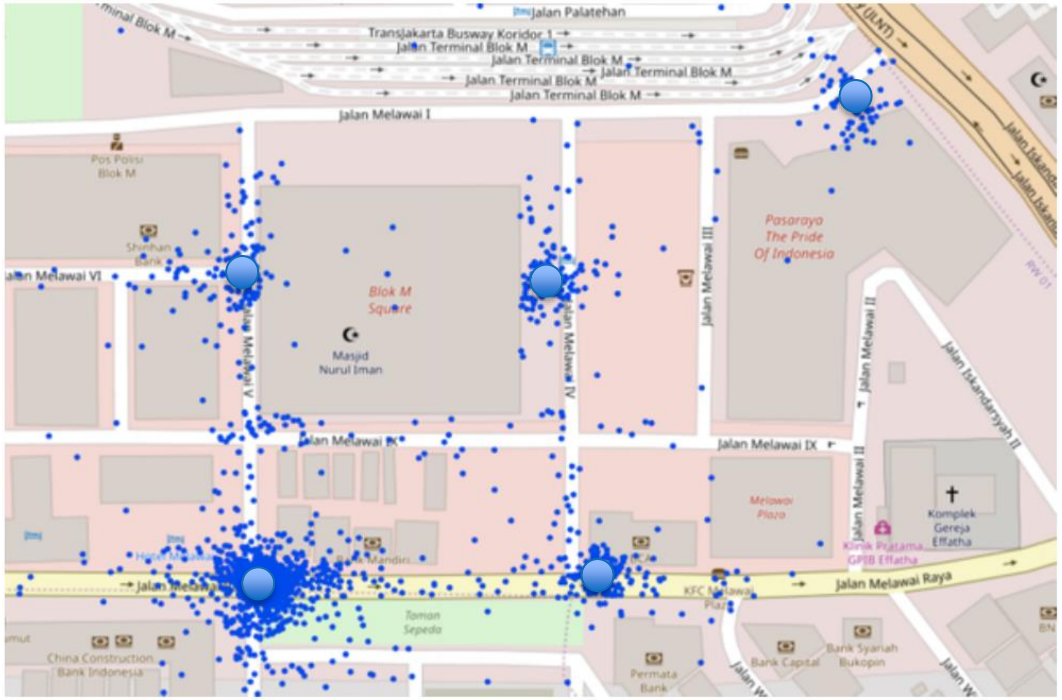
Study Case (GOJEK Meeting Points)

Project's Target

Determine meeting point (gates) of any Place of Interest (POI) based on historical driver pick up data



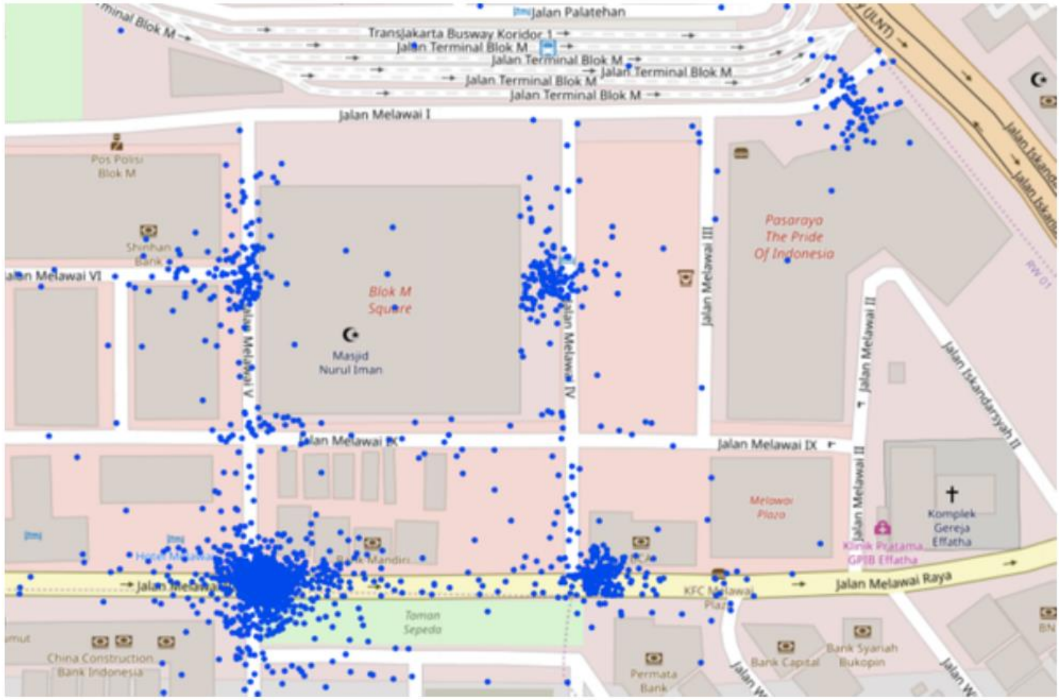
Clustering



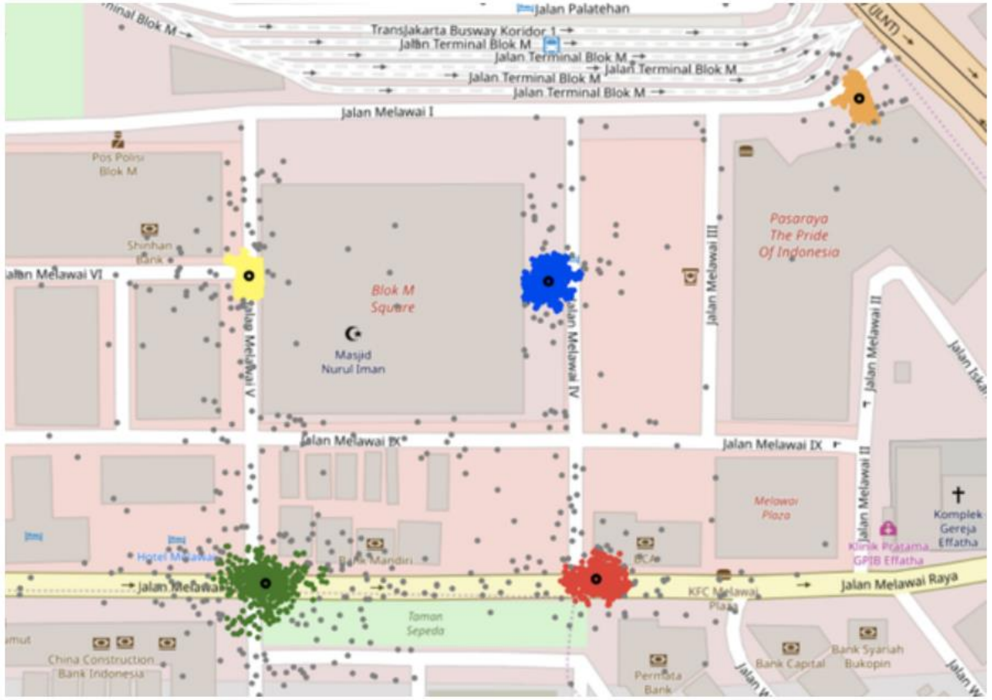
POI Example: Blok M Square

Using DBSCAN

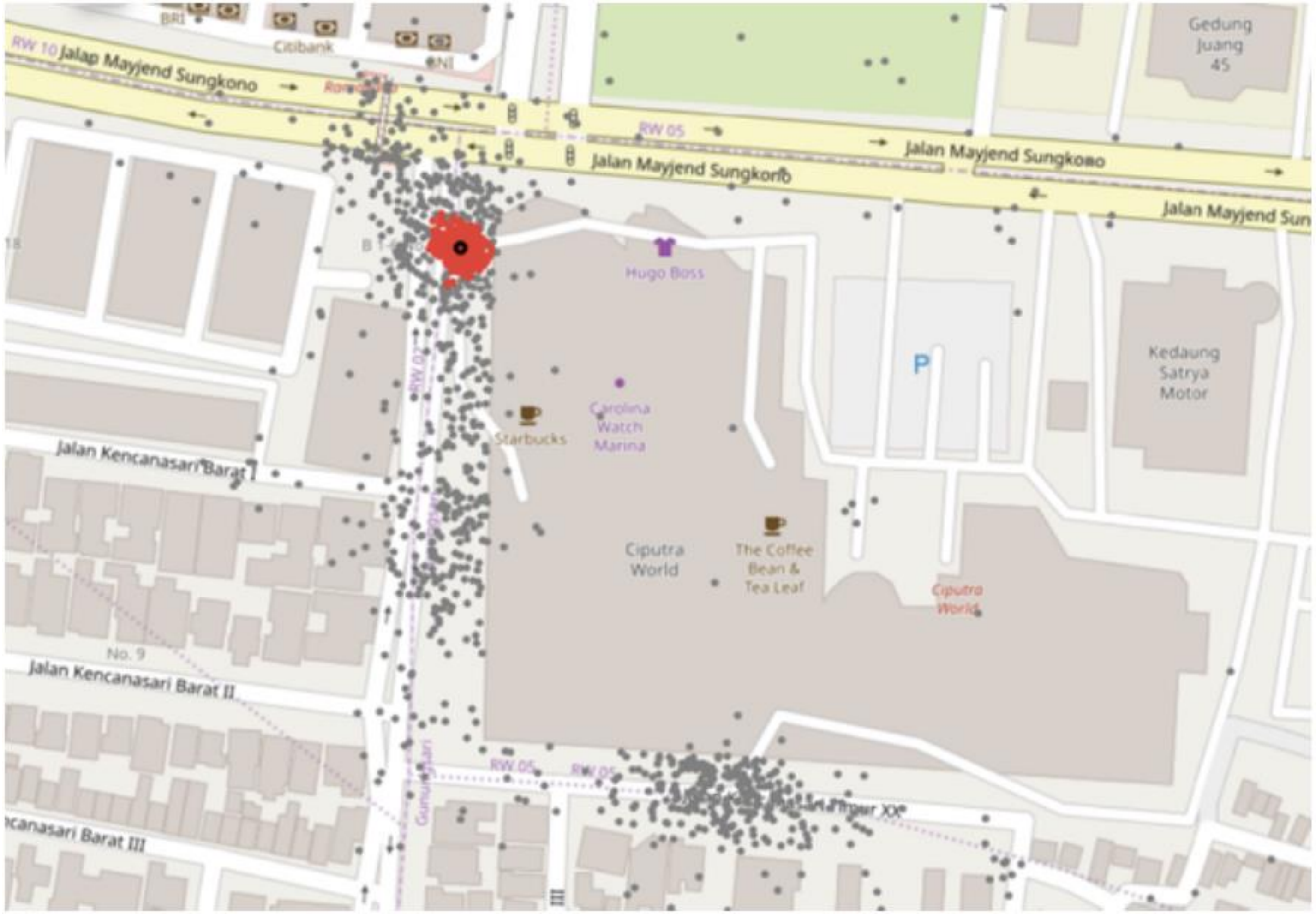
After some hyperparameter tuning (eps and min_samples)
The result is 5 clusters for determining gates on Blok M Square



Looks ok



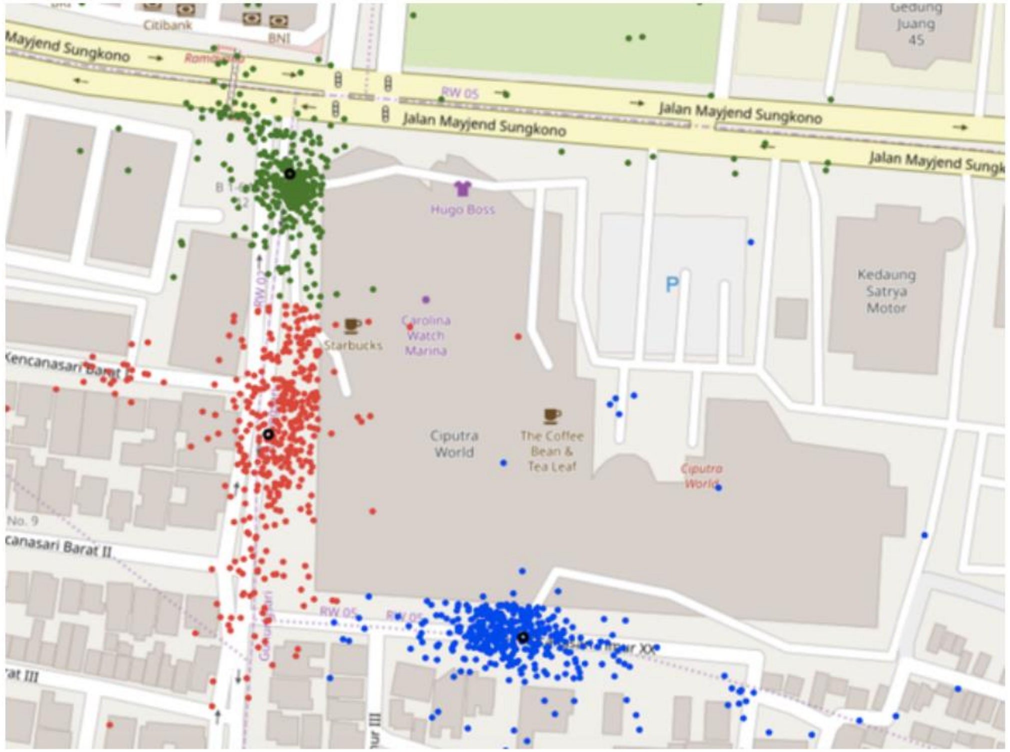
POI : Ciputra World



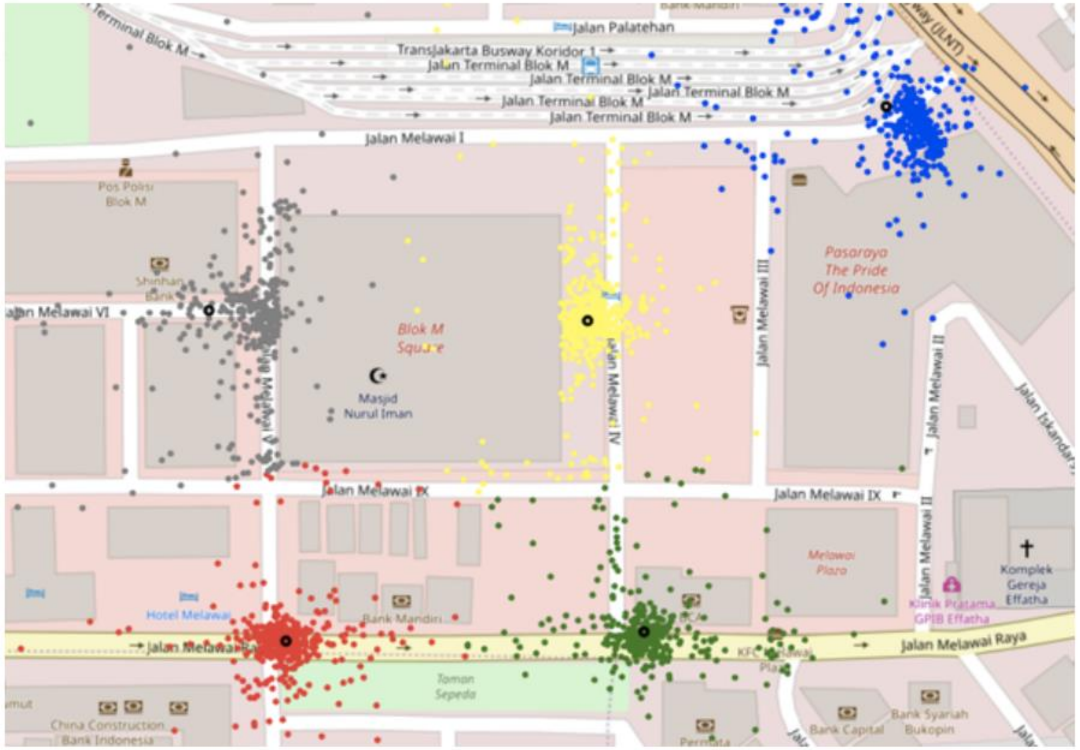
Datapoints are spread so model cant determine cluster

Using KMeans

Ciputra World



Blok M Square



Different POI may have different number of gates, but Kmean has a fix value of K
So, how to find best K value for each POI

- Clustering Evaluation

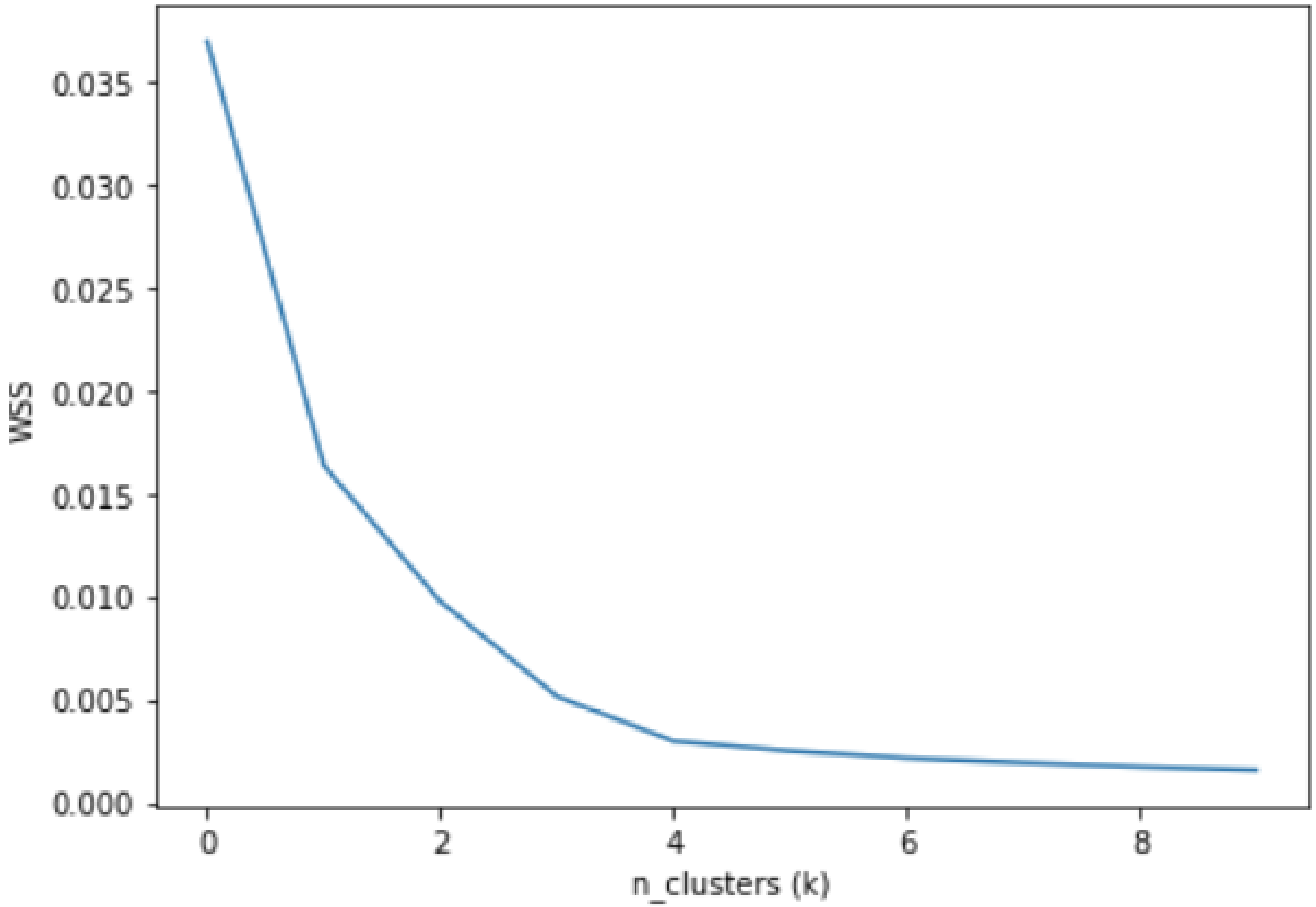
Clustering Evaluation

- Mengukur kedekatan dalam cluster, dan pemisahan antar cluster yang berbeda
- Cluster yang baik : *Compact, Separated*
- Ukuran evaluasi clustering:
  - Sum of Squared Error
  - Silhouette Index
  - Davies Bouldien Index

WSS (Within-cluster Sum of Squared) Approach

Sum of squared error of each datapoint on a same cluster

Find diminishing point (elbow) on WSS curve

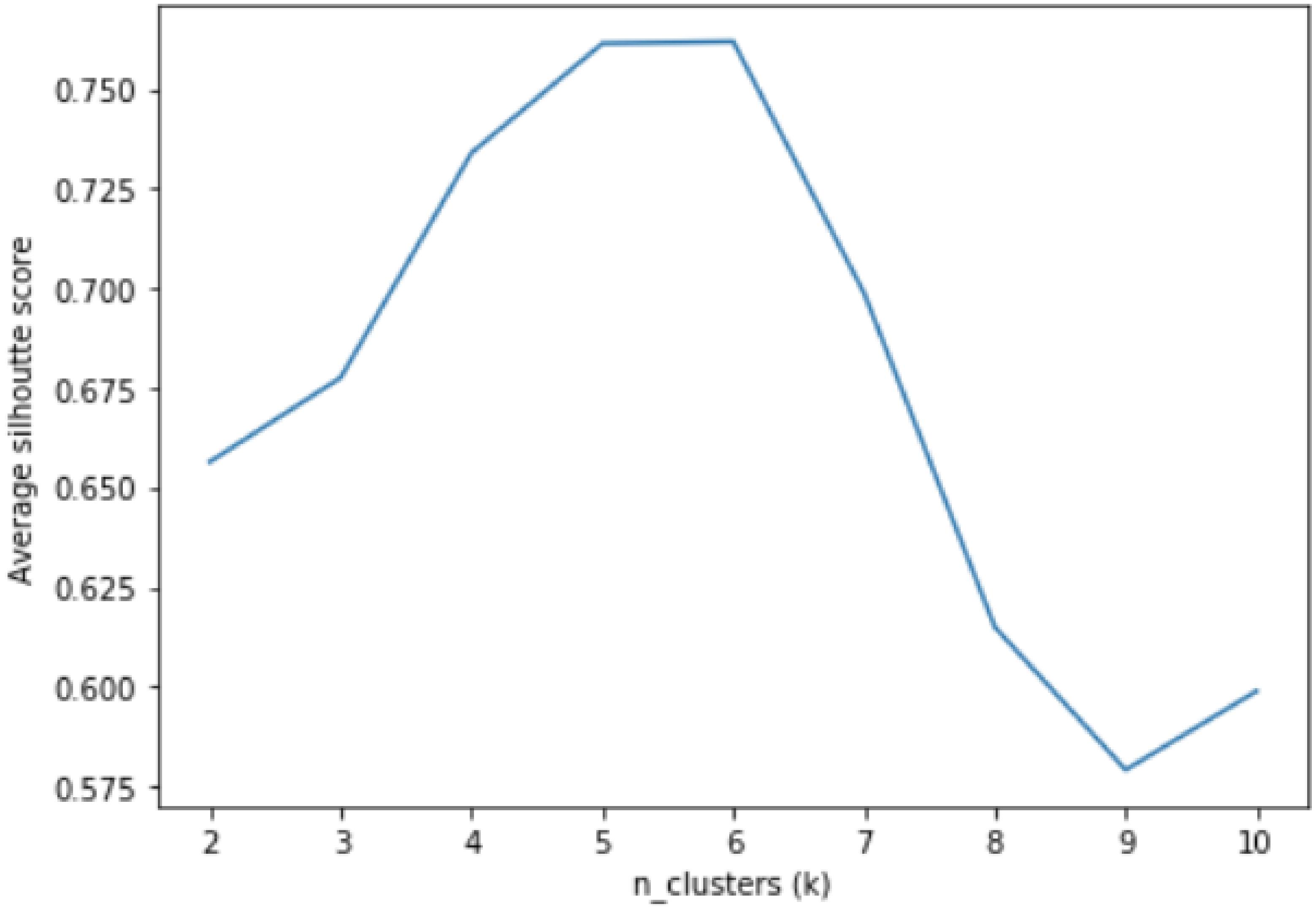


Optimal K value is between 4 and 6

Silhouette Score Approach

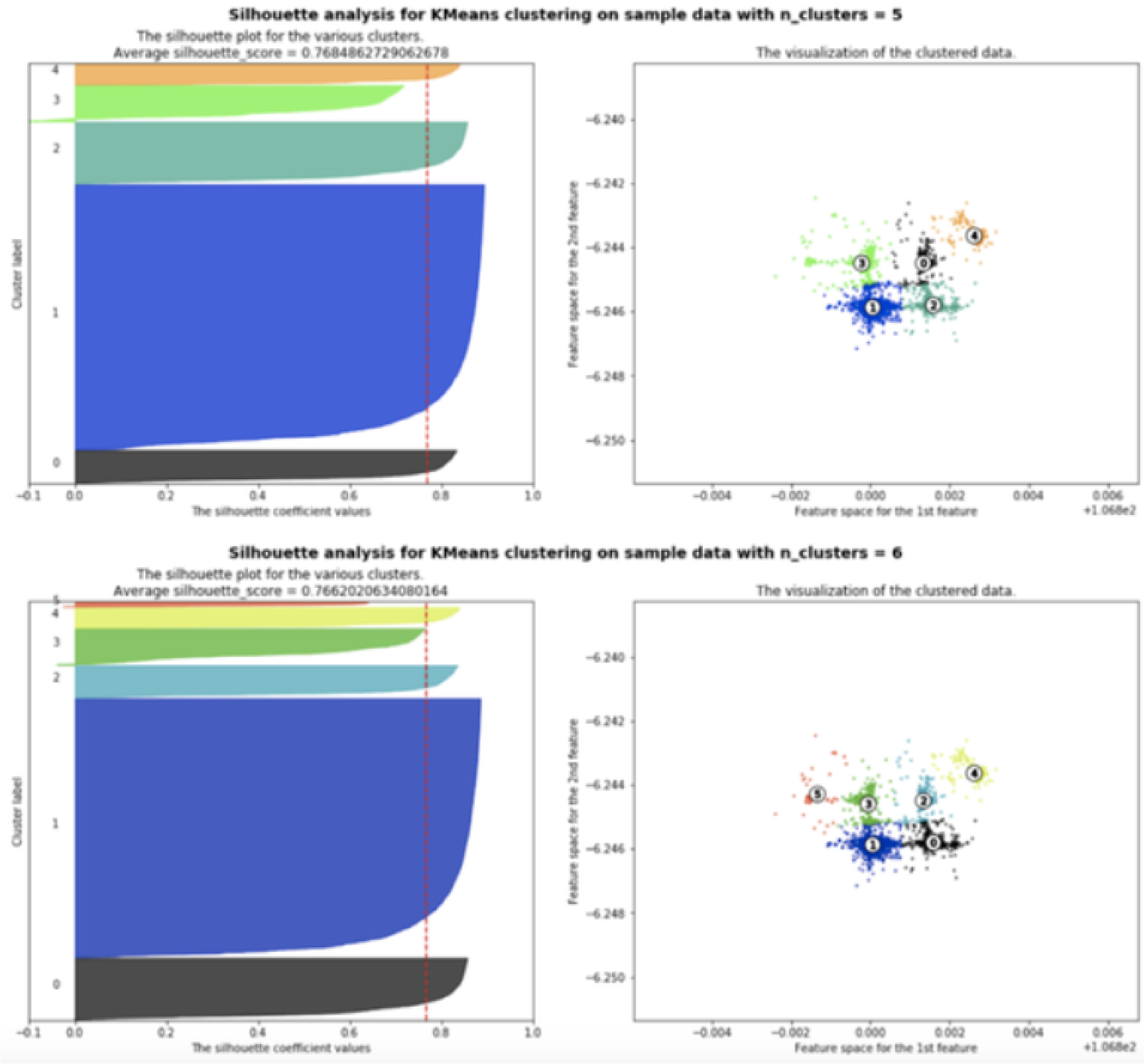
The silhouette value measures how similar a point is to its own cluster (cohesion) compared to other clusters (separation).

Find the maximum point of average silhouette score



Optimal K value is 5 or 6

Compare K = 5 and K = 6



- K=5 gives a little bit more silhouette point
- K=6 makes 1 additional but not too significant cluster (red #5)
- So, better to choose K = 5

Next Step

Automate the process of finding optimal K-value for analysing different POI

Give name of each gate (cluster centroid/mean), so it is easier for drivers and passengers to locate the gate

NLP case

CONTOH PENERAPAN PADA PYTHON

klik di sini

TERIMA KASIH