Clustering practicum

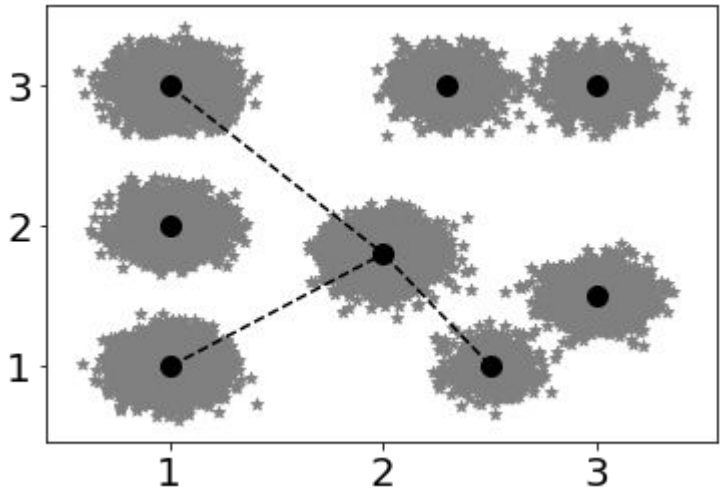
Stanislav Protasov

Agenda: clustering

- Problem statement
- How we measure quality
- Couple of algorithms
  - K-means
  - DBScan
  - Louvain modularity

Why do we cluster?

- Any **modelling** is done to **simplify data**
- simplify because we cannot **make decisions** based on millions of numbers
  - E.g. regression brings **few parameter numbers** to describe a domain instead of holding samples
  - "*Terminator and similar*" is a good way to describe customer's preferences
- Clustering is a way to bring **limited number of entities** (clusters or representatives) while **preserving** general **idea** about the structure.



Clustering - what is this?

Set partitioning - grouping of set's elements into non-empty subsets, such that every element is included in **one and only one** of the subsets (disjoint).

Number of partitionings - **Bells number** (~ e^x) $B_{n+1} = \sum_{k=0}^{n} \binom{n}{k} B_k$

Number of non-empty partitionings of size k- **Stirling number of second kind**

$$\left\{ {n \atop k} \right\} = \frac{1}{k!} \sum_{i=0}^{k} (-1)^i \binom{k}{i} (k-i)^n.$$

NB: for any metric introduced, we <u>cannot</u> solve a problem with brute force

Clustering - what can we do then?

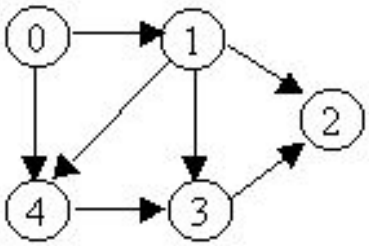
Thus we put **limitations**:

- [Optionally] Predefine number ***k*** of clusters
- Implement **iterative** approaches with **convergence**
- **Rely on distance** to avoid considering obviously bad case

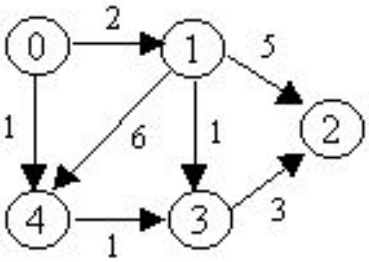
But even then clustering is usually slow.

Clustering - what is the **object**?

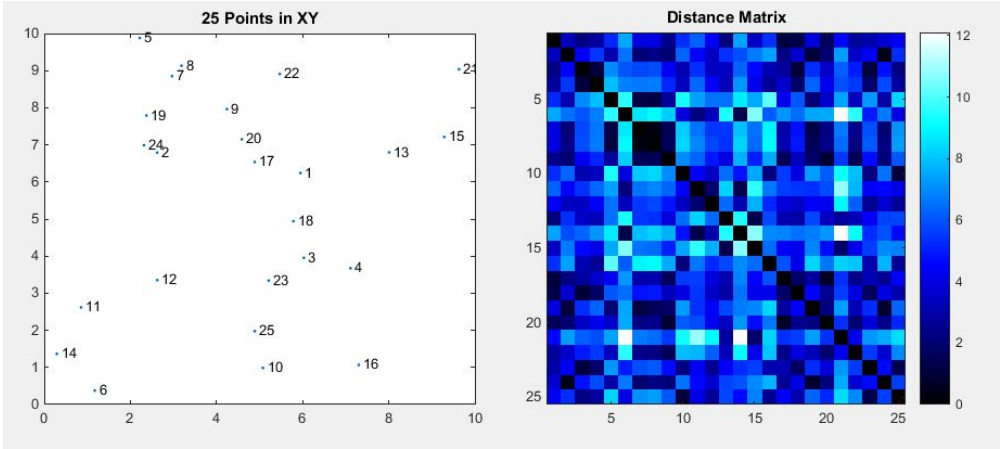
As we don't have any idea about cluster form, we usually rely on **distance** and its representation in graph or matrix form. There are 2 major approaches to define distance: *metric* and *vector* spaces.



$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$



$$A = \begin{pmatrix} \infty & 2 & \infty & \infty & 1 \\ \infty & \infty & 5 & 1 & 6 \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & 3 & \infty & \infty \\ \infty & \infty & \infty & 1 & \infty \end{pmatrix}$$



Clustering - how to understand success?

General idea: ... include groups with **small distances between cluster members**, dense areas of the data space ...

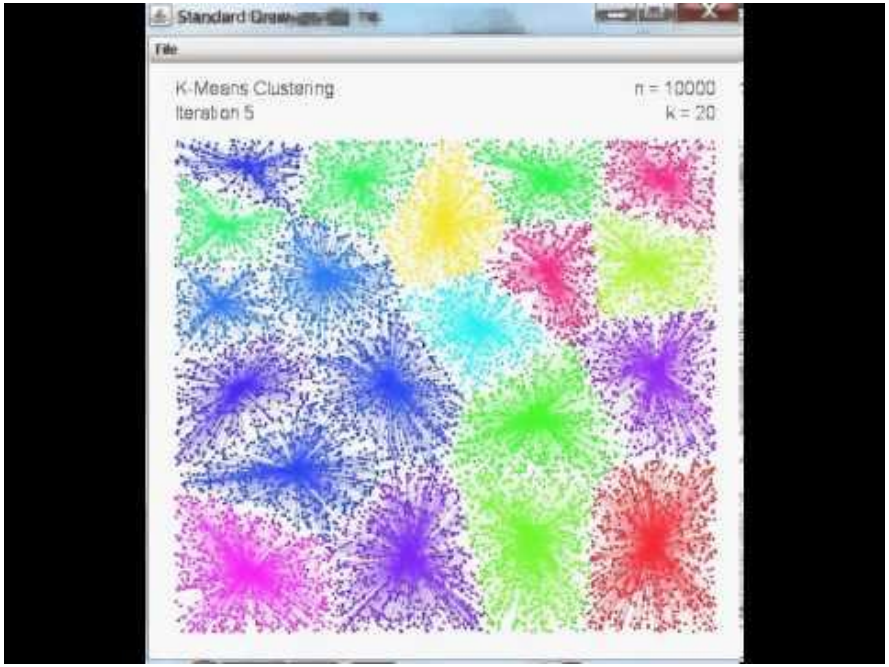
Also: maximize between-cluster variance, minimize within-class variance.

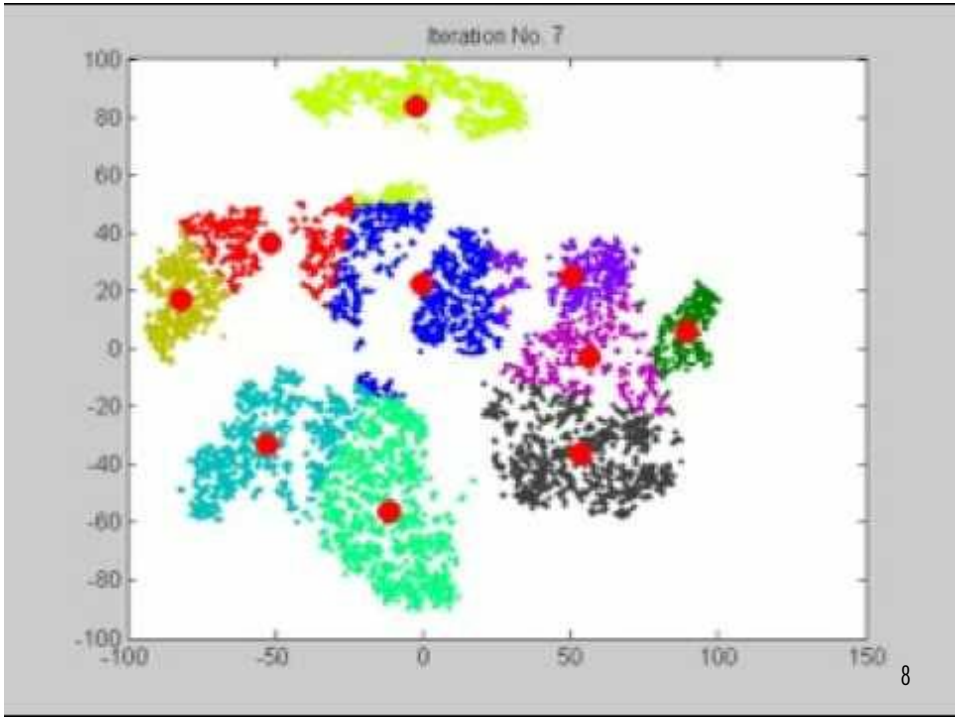
Internal evaluation (on the training data).

- Davies–Bouldin index $DB = \frac{1}{n}\sum_{i=1}^{n}\max_{j\neq i}\left(\frac{\sigma_i + \sigma_j}{d(c_i, c_j)}\right)$
- Dunn index
- Silhouette Coefficient $s = \frac{b-a}{max(a,b)}$ $SC = \max_k \tilde{s}\,(k)$

Purity, coverage, Differential edit distance - rely on **pre-defined** clusters (compare with validation set)

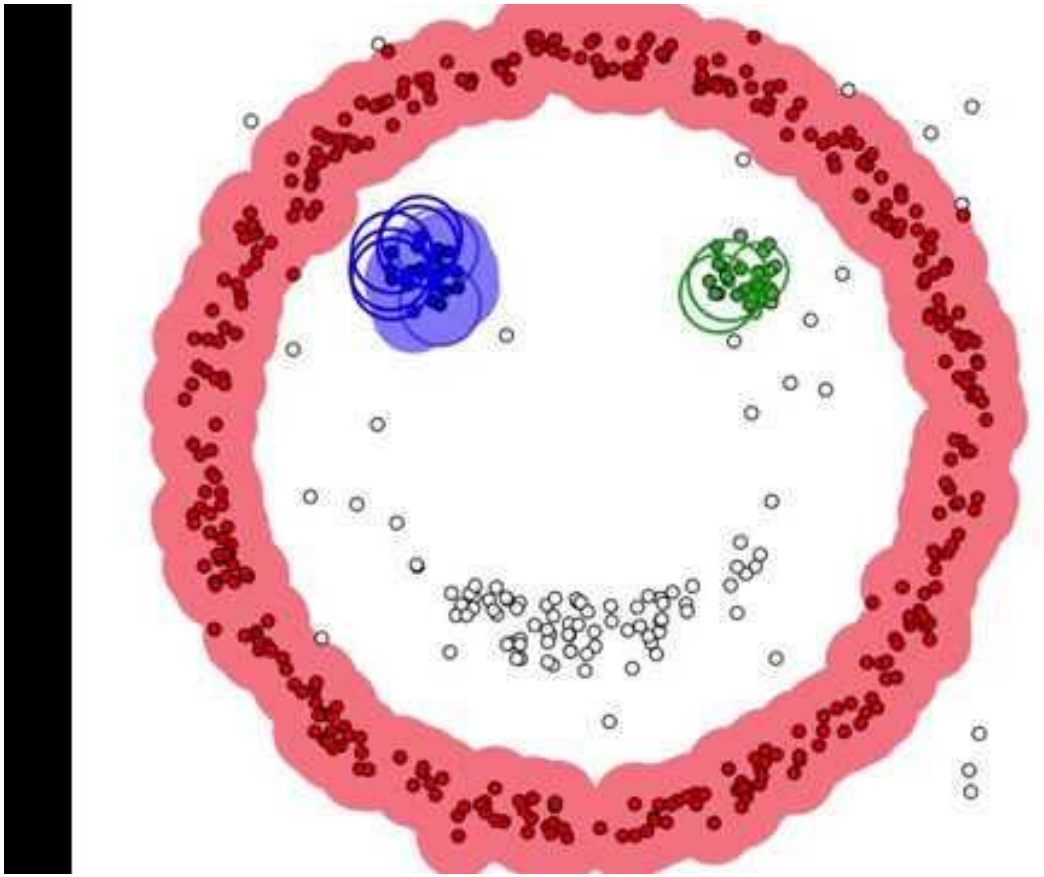
K-Means

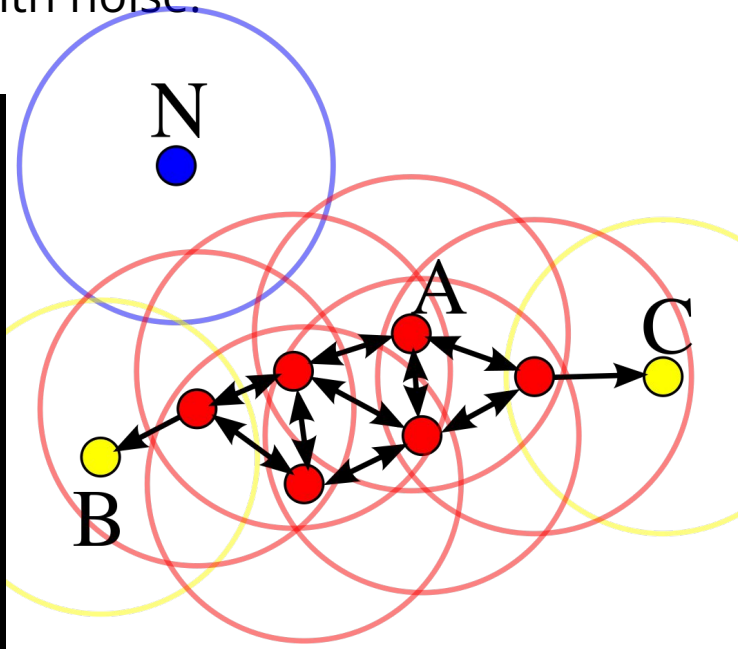

File
K-Means Clustering
Iteration 5
n = 10000
k = 20


Iteration No. 7
100
80
60
40
20
0
-20
-40
-60
-80
-100
-100
-50
0
50
100
150

DBScan

Density-based spatial clustering of applications with noise.





Lab #1. Clustering 2D points

- Consider clustering example.
- What is silhouette score for k={2, 3}?
- Will DBScan provide better results?
- What about this example?

Lab #2. Multidimensional case. Multiple subscriptions

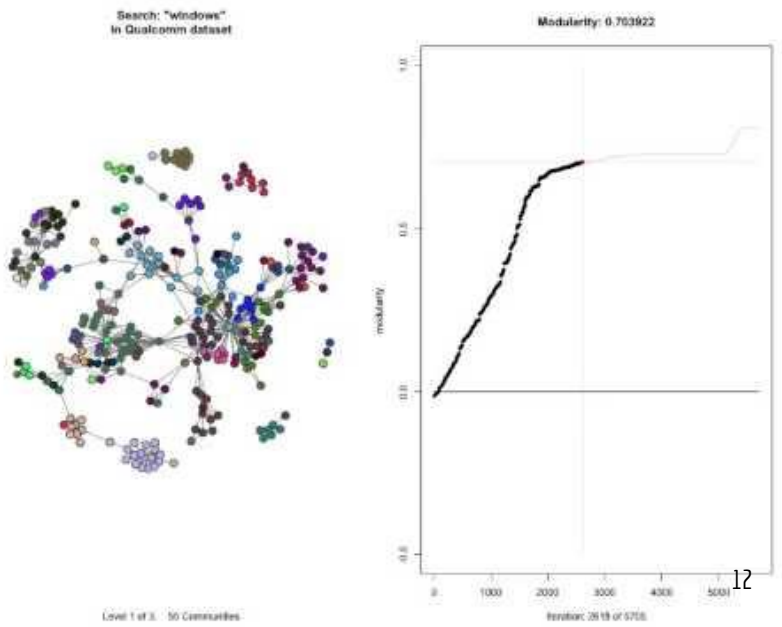
multiple subscriptions

Louvain modularity

$$Q = \frac{1}{2m} \sum_{ij} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j),$$

- A_{ij} represents the edge weight between nodes i and j;
- k_i and k_j are the sum of the weights of the edges attached to nodes i and j, respectively;
- $2m$ is the sum of all of the edge weights in the graph;
- c_i and c_j are the communities of the nodes; and
- δ is a simple delta function.

- Graph-based
- Considers only existing edges (no centroids)
- Starts with community number == number of nodes.
- Searches for communities. Change element assignment if this improves **modularity Q**



Lab #3

Complete the tutorial using Louvain modularity algorithm. Did it perform ok?

Can you measure **silhouette score** somehow?