Clustering

Chloé-Agathe Azencott

Center for Computational Biology (CBIO)
Mines Paris – Institut Curie – INSERM U1331
PSL Research University & PR[AI]RIE, Paris, France

Sep 17, 2026

http://cazencott.info chloe-agathe.azencott@minesparis.psl.eu @cazencott@lipn.info

1. Clustering

unsupervised

$\mathcal{D} = \{x^i\}_{i=1..n}$ unlabeled samples

Goal: group the samples in homogeneous subsets

Ex) market segmentation topic clustering
disease subtyping

→ better understand data
→ visualization
→ outliers = samples alone in their cluster

preprocessing for supervised learning

2. Evaluation

2. Evaluation

2.1 Cluster shape

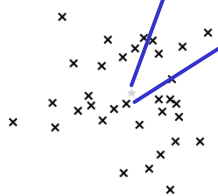
clustering = partitioning $\mathcal{D}$ into $C_1, C_2, \dots, C_K$

C_k

for cluster k : $\mu_k = \frac{1}{|C_k|} \sum_{x \in C_k} x$

centroid = mean of the samples in that cluster of a cluster

medoid = sample that is closest to the centroid



⊛ Cluster homogeneity or tightness

$$T_k = \frac{1}{|C_k|} \sum_{x \in C_k} d(\mu_k, x)$$

ideally small

overall $T = \frac{1}{K} \sum_{k=1}^{K} T_k$

Ideally:



⊛ Cluster separability $S_{k,\ell} = d(\mu_k, \mu_\ell)$

overall $S = \frac{2}{K(K-1)} \sum_{k=1}^{K} \sum_{\ell > k} S_{k,\ell}$

ideally large

Silhouette coefficient

Evaluate, for each sample, whether it's well located in its cluster

$$s(x) = \frac{b(x) - a(x)}{\max(a(x), b(x))} \quad \in [-1, 1]$$

$a(x)$ = average distance between x and the cluster $C(x)$ it's been assigned to

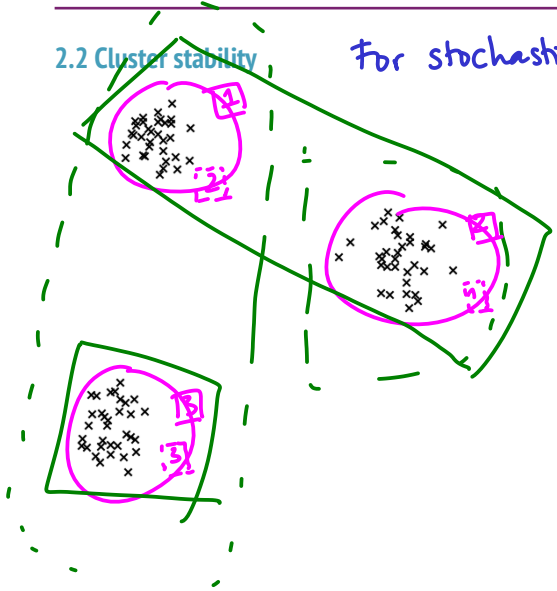
$$= \frac{1}{|C(x)| - 1} \sum_{\substack{u \text{ in } C(x) \\ u \neq x}} d(u, x)$$

$b(x)$ = the smallest value a would take if x was assigned to another cluster

$$\min_{C \neq C(x)} \frac{1}{|C|} \sum_{u \text{ in } C} d(u, x)$$

Overall: $s = \frac{1}{n} \sum_{x \in D} s(x)$

2.2 Cluster stability



For stochastic algorithms (≠ initialisation
↓
≠ result)

reasonable clustering algo, K=3

K=2

→ right-ish
very ≠ clusters

2.3 When a ground truth exists (either labels we didn't use, or another clustering)

The names we give to clusters don't necessarily match the labels

Rand index Look at all pairs (x_i, x_ℓ) of samples

- If they have the same label, are they in the same cluster?
- if they don't, are they in $\neq$ clusters?

↓

proportion of such pairs

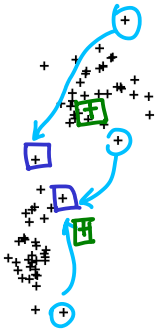
3. K-means – Lloyd's algorithm

K is fixed (hyperparam)

Can only find convex clusters



can be kernelized





In general: dim red before clustering (curse of dimensionality).

Goal: minimize intra cluster variance

$$\underset{C_1,\dots,C_K \text{ partition of } \mathcal{D}}{\operatorname{argmin}} \sum_{k=1}^{K} \sum_{x \in C_k} \|\vec{x} - \vec{\mu}_k\|^2$$

Corresponds to the tightness we've defined earlier.

Solved with a heuristic (Lloyd's)

1) Pick K centroids at random

reiterate until convergence:

2) Assign each point to its closest centroid

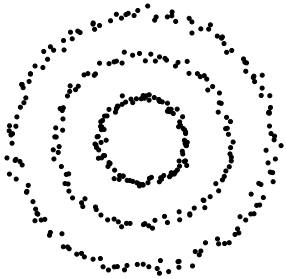
3) Recompute the centroids

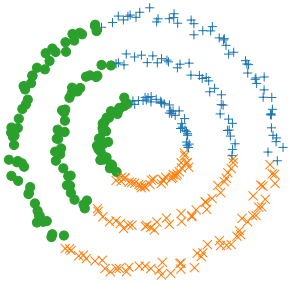
by design, each sample is closer to its own centroid than to the others

4. Density-based clustering – DBSCAN

hyperparam = neighborhood size ε

Again: make sense in low dimension

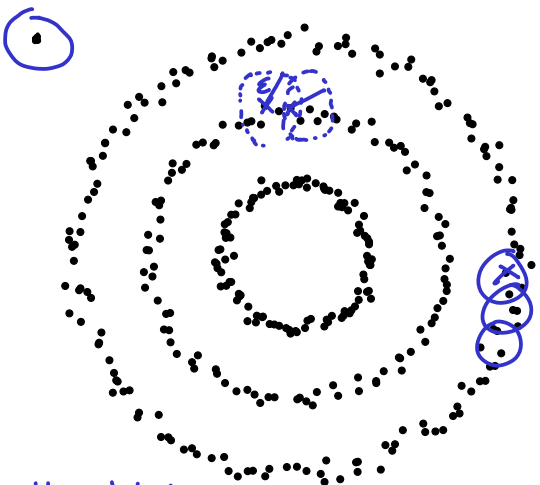




k-means solution with k=3

4. Density-based clustering – DBSCAN

hyperparam = neighborhood size ε



1) pick one datapoint at random to initialize a first cluster
2) add to this cluster all points that are within a distance ε of its points

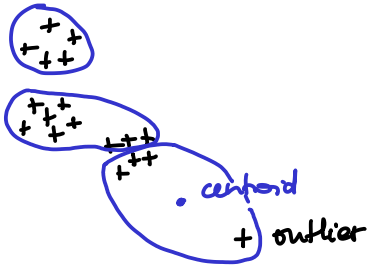
↑ repeat until no more points to add

3) pick a point at random among those that haven't been assigned to a cluster yet

ε too small: many small clusters
too large: one single cluster

Outlier detection is built-in:
outliers are points that are alone in their cluster

K=3



K means is sensitive to outliers