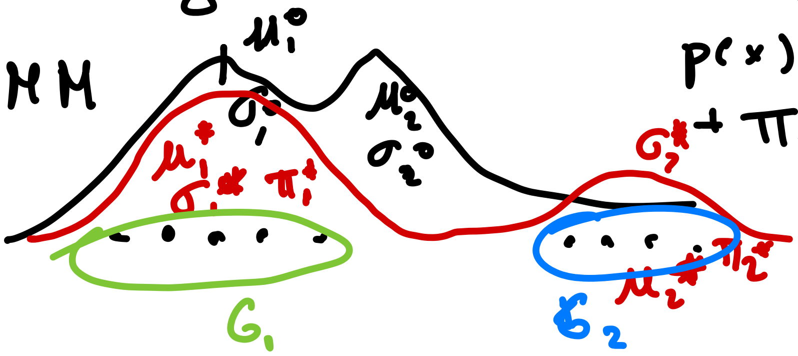
Expectation - Maximization

clustering : Parametric density estimation

GMM



$$p(x) = \pi_1 G(\mu_1, \sigma_1, x) + \pi_2 G(\mu_2, \sigma_2, x)$$

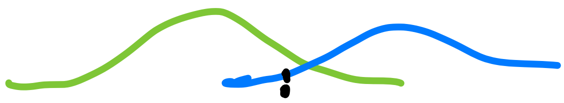
$$\theta_\ell = \{\mu_\ell, \Sigma_\ell\}$$

$$\underbrace{p(x)} = \sum_{\ell=1}^{K} \pi_\ell \underbrace{G(x \mid \theta_\ell)}$$

$$\sum_{\ell}^{K} \pi_\ell = 1$$

$$\mathcal{L}(X) = \prod p(x^i) = \prod_i \sum_{\ell=1}^{K} \pi_\ell G(x^i \mid \theta_\ell)$$

$$\log \mathcal{L}(X) = \sum \log\left(\sum_{\ell=1}^{K} \pi_\ell G(x^i \mid \theta_\ell)\right)$$



$$p(x) = \frac{G(x^i)}{G(x^i) + G(x^i)}$$

$$\omega_k^i = \frac{\pi_k G(x^i \mid \theta_k)}{\sum_\ell \pi_\ell G(x^i \mid \theta_\ell)}$$ Expectation

Effective population

LECTURE 10 :
- EXPECTATION MAX.
- DBSCAN
- DENSITY PEAKS
- VALIDATION METRICS

$$p(x) = \sum_{\ell}^{K} \pi_\ell \, F(x \mid \theta_\ell)$$

$$\theta_\ell = \{\underline{\mu}_\ell, \underline{\Sigma}_\ell\}$$

$$\mathcal{L}(\pi, \theta) = \sum^{N} \log\left(\sum_{\ell}^{K} \pi_\ell F(x_i, \theta_\ell)\right)$$

$$(\theta^0, \pi^0)$$

$$\downarrow$$

w_ℓ^i → probability of point i being generated by gaussian ℓ

$$\boxed{w_\ell^i = \frac{\pi_\ell F(x^i, \theta_\ell)}{\sum_j \pi_j F(x^i, \theta_j)}}$$

$$N_\ell = \sum_i w_\ell^i$$

$$\mu_\ell = \frac{1}{N_\ell} \sum_i^N w_\ell^i \, x^i$$

$$\Sigma_\ell = \frac{1}{N_\ell} \sum_i^N w_\ell^i \, (x^i - \mu_\ell)(x^i - \mu_\ell)^T$$

$$\pi_\ell = \frac{N_\ell}{\underbrace{\sum_j N_\ell}_{N}}$$

$$\begin{tikzcd} (\pi^0, \theta^0) \arrow[r] & w_\ell^i \arrow[l, bend left] \end{tikzcd}$$

$\log \mathcal{L} \to$ convergence check

Bayesian Model Selection

Dirichlet process

$$\underline{\pi} \;/\!/\; \underline{1-\pi}$$

$$\pi \sim Beta(\alpha, \beta)$$

Classical Methods

- k-means
- fuzzy c-means
- Agglomerative & Divisive hierarchical clustering
  - Single linkage
  - Ward's r
  - ↳ DIANA

Modern methods

- Kernel k-means
- Spectral clustering
- Affinity propagation
- Expectation Maximization

DBSCAN

Density Based Spatial Clustering of Applications with Noise

CLUSTERS: Regions of high density separated from other clusters by regions of low density

ε : distance

ρ_i = number of points within ε of point i

Classes of points:

- <u>Core points</u>: Has more than <u>MinPts</u> within <u>ε</u>
- <u>Border points</u>: Has less than MinPts within ε, but at least one of them is a core point.
- <u>Noise point</u>: it's not a core nor a border point.

Ideas:

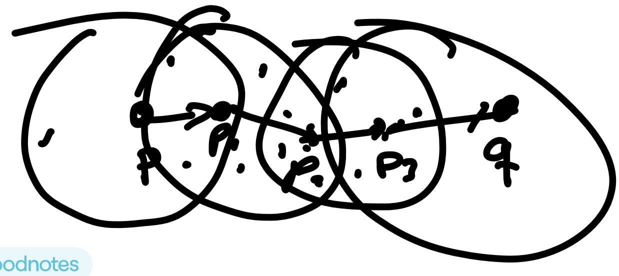
1. Any two core points close enough are put in the same cluster
(close enough = within ε)

2. Any border point that is close enough to a core point ...

3. Noise points are discarded

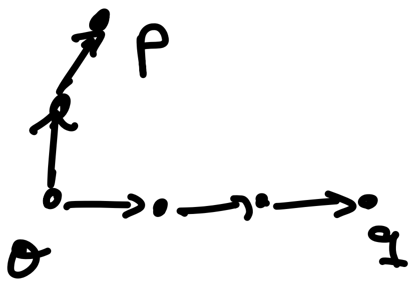
Concepts:

- Neighborhood: Points that are within ε.
- Reachability:
  - Direct: A point q is directly reachable from point p if q is in the neighborhood of p & p is a core object
  - Density-reachability: A point q is density reachable from point p if there is a chain of points direct-reachable from each other that connects them



Density connectivity : Points p & q are density connected if both are density reachable from a point o

SYMMETRIC PROPERTY

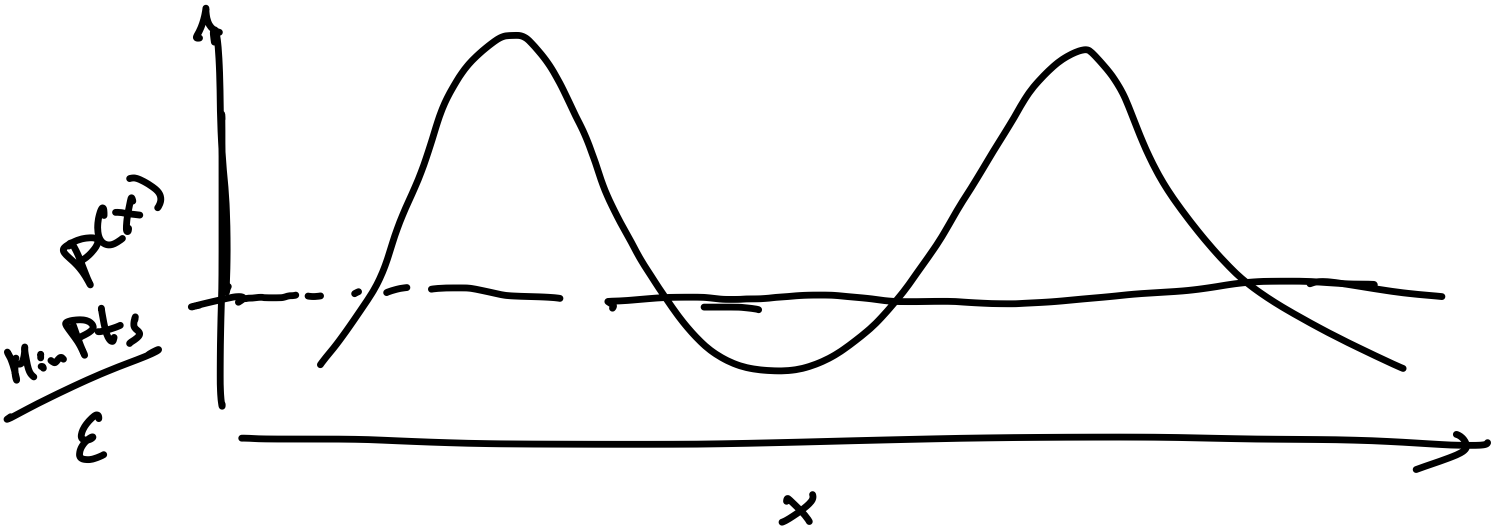


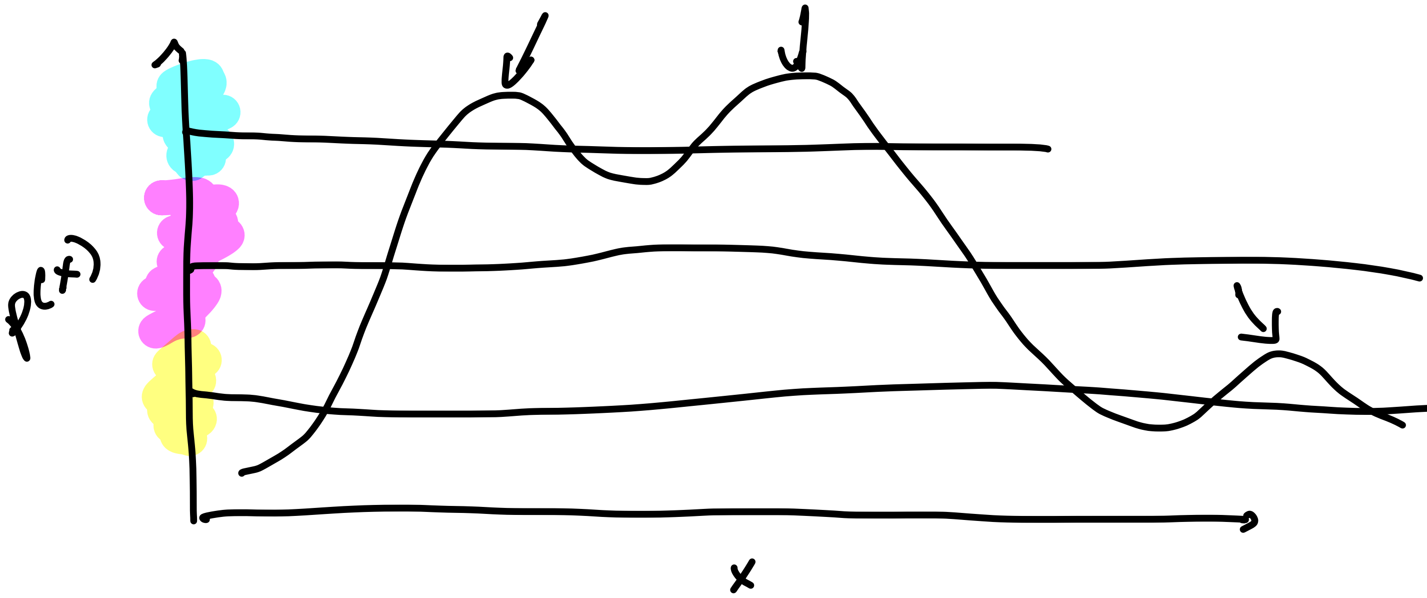
Definition of cluster :

Set of all the points that are density connected

1. • Set ε & MinPts
2. • Choose a point p not processed
3. • Retrieve all points that are density reachable from p
4. • if p was a core point → Put all these points in a cluster
5. • Continue until all the points have been processed : < chosen point in 2 / clustered in 4

General idea





Density peak clustering

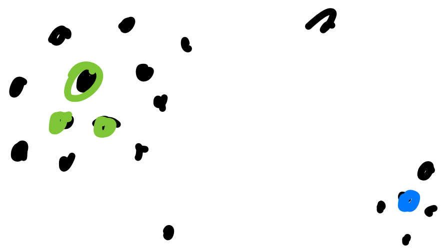
Clusters are Peaks in the density of points (already present in Mean-Shift)

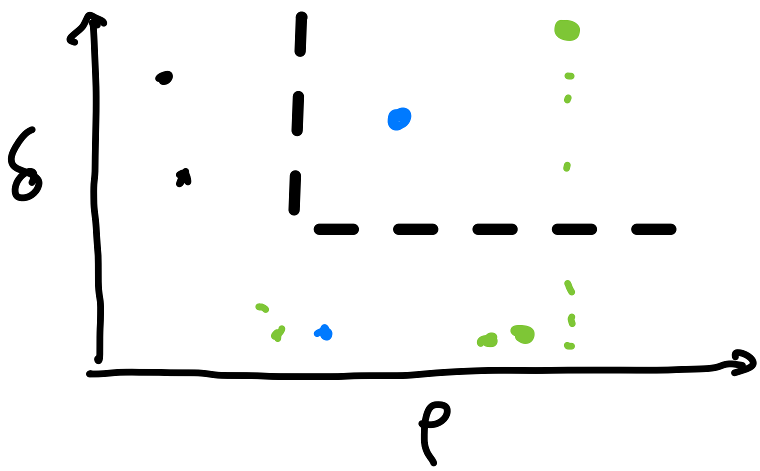
$\rho^i = \underbrace{\chi(\varepsilon)}_{\text{Number of points within } \varepsilon \text{ of } i}$

δ^i = Minimum distance from a point with higher density

δ of point with highest density $\Rightarrow$ arbitrary high value

Plot δ as a function of ρ



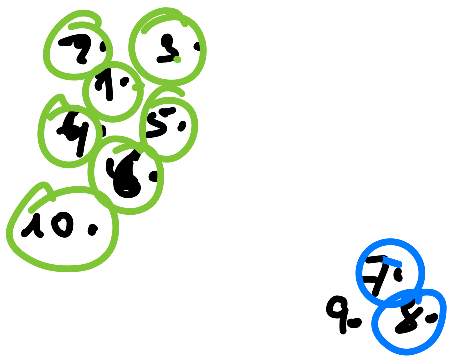


1. Compute the densities for all the points

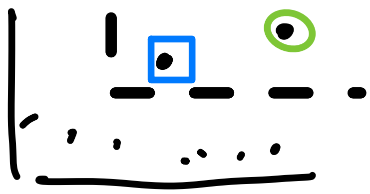
2. Compute δ and assign the δ of the highest density point to an arbitrary high number

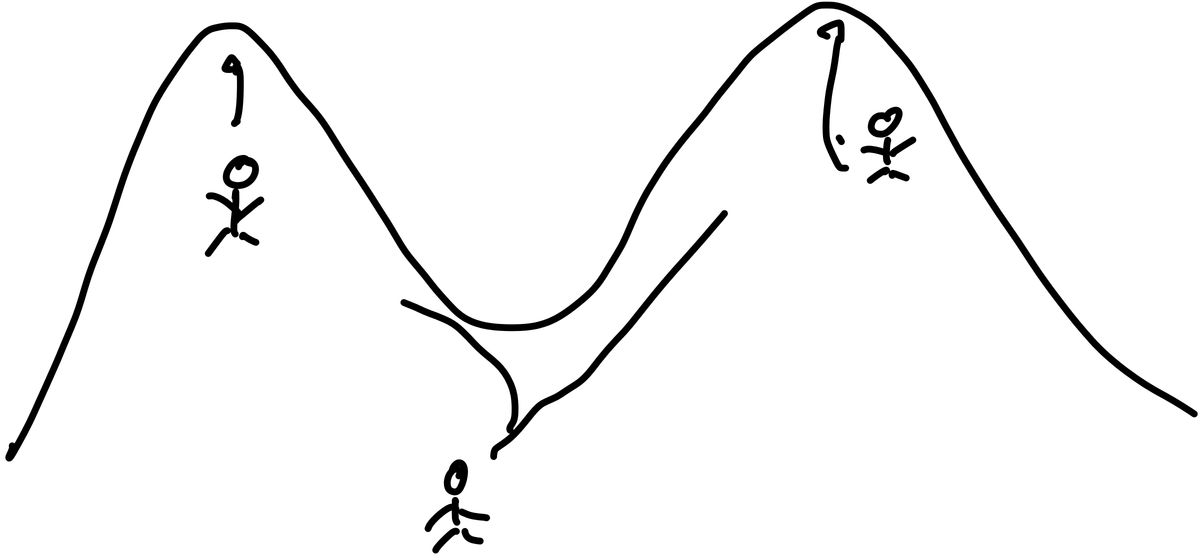
3. Plot δ as function of ρ (Decision graph). Identify the cluster as outliers

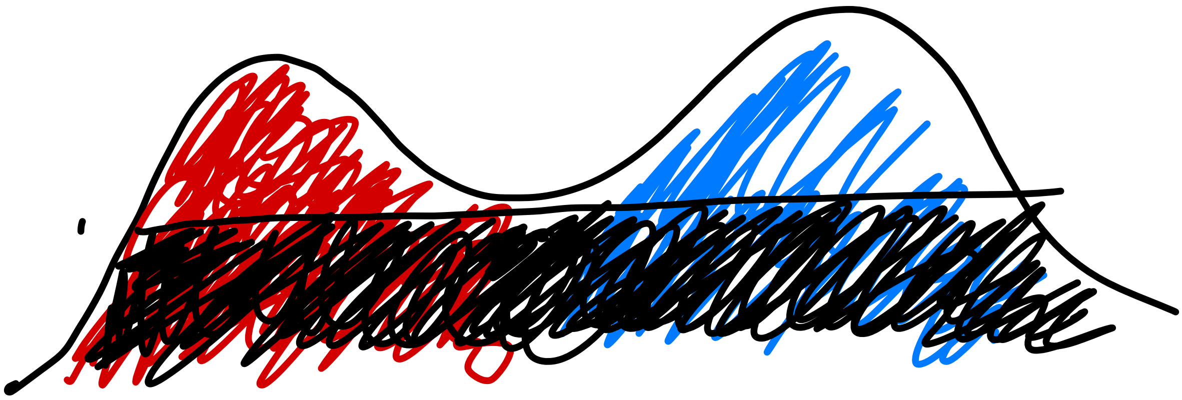
4. Assign in decreasing order of density each point to the same cluster as it's nearest neighbor with higher density



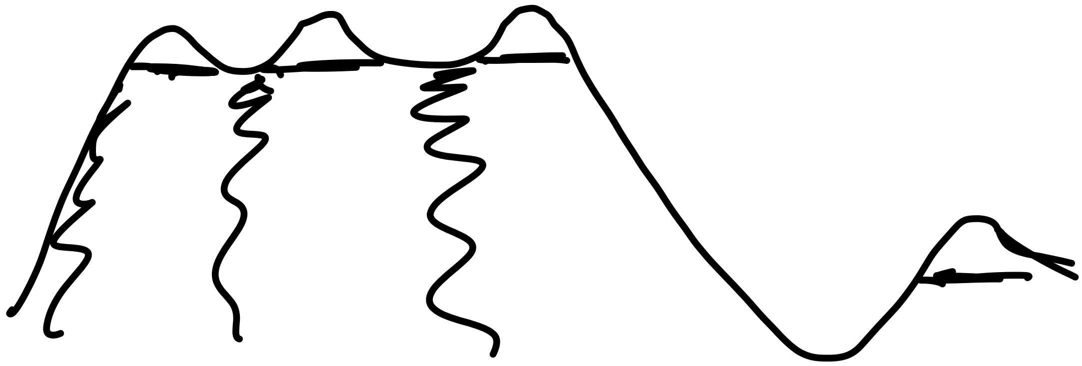
1
2
3
4
5
6
7
8
9
10



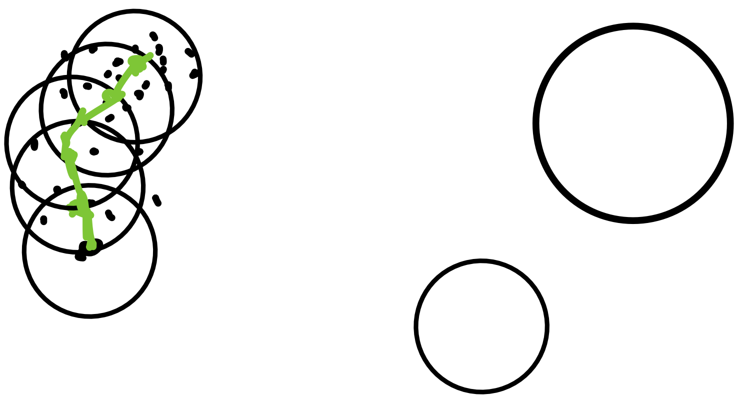




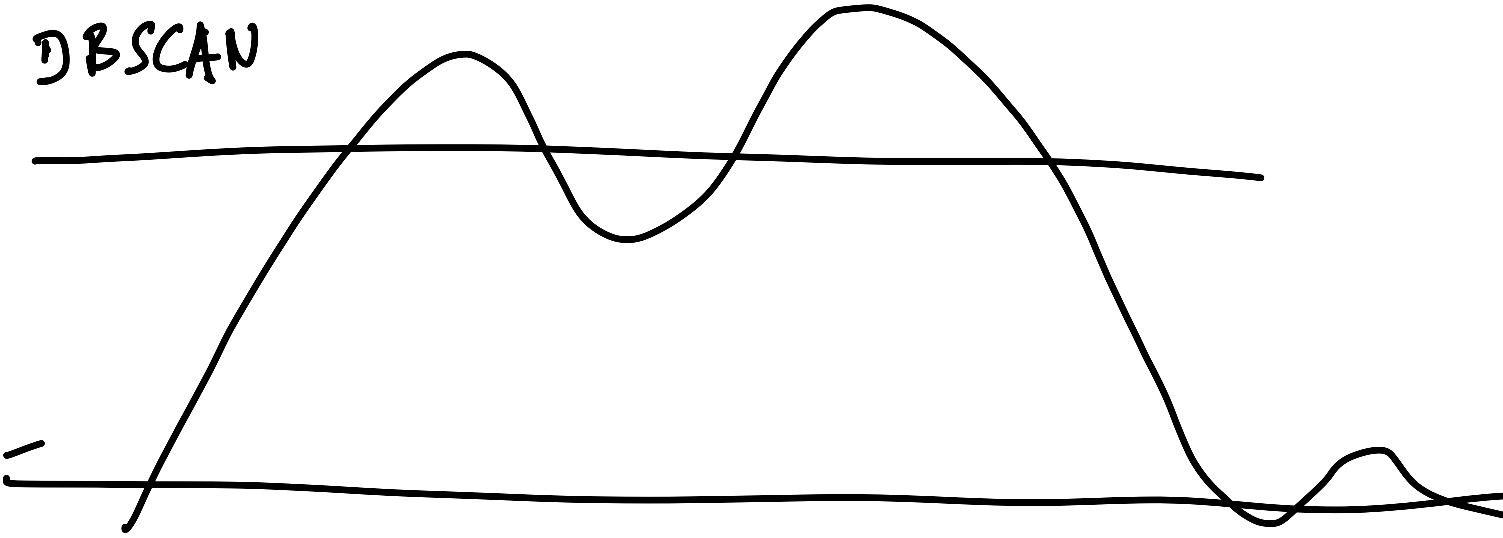
- i that belongsto cluster A is a border point if it has a neighbor (within ε) that belongs to another cluster B
- The border density of cluster A will be the highest density among the border points belonging to A
- Points belonging to cluster A will be labeled as "halo" if their density is lower than the border density of A



An introduction to mean-shift clustering

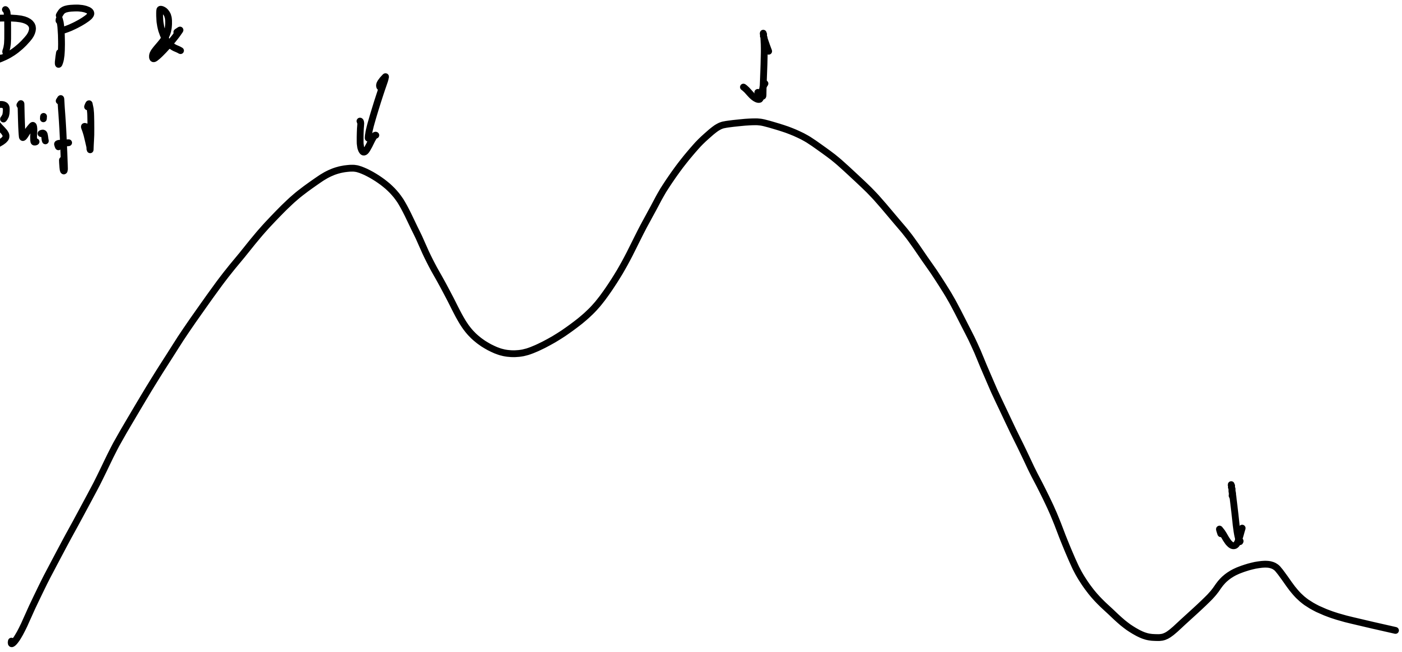


DBSCAN



DP &

Mean-Shift



VALIDATION

- EXTERNAL (Compare with a classification)
- INTERNAL VALIDATION (CHECK CLUSTER PROPERTIES)
    - Minimize the intra cluster variance
    - Maximize the inter-cluster variance

$$\sigma^2(X) = \sum^{N} (x^i - \bar{x})^2 = \sum_{i}^{N} \|x^i - c_\ell\|^2 \delta(z^i, \ell) + \sum_{\ell}^{k} n_\ell \|c_\ell - \bar{x}\|^2$$

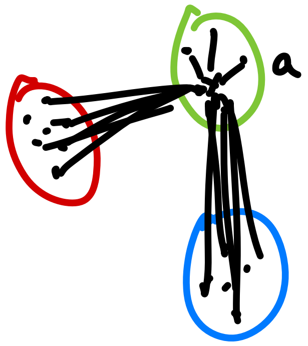
$\downarrow$ $\uparrow$

F-ratio test

$$F = \frac{k \sum^{N} \|x^i - c_\ell\|^2 \delta(z^i, \ell)}{\sum_{\ell}^{k} n_\ell \|c_j - \bar{x}\|}$$

Silhouette Coefficient

- Cohesion (b^i): average distance to other points in the same cluster
- Separation (a^i): minimum average distance to points in one different cluster



a

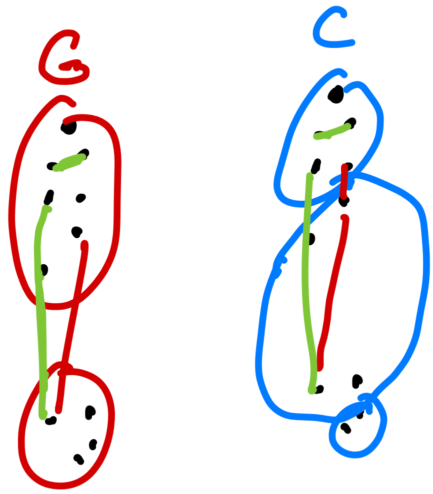
$$s^i = \frac{b^i - a^i}{\max(b^i, a^i)}$$

$\{-1, 1\}$

$$s(z^i) = \langle s^i \rangle$$

External Validation

$\{C, G\}$



- a++ (same cluster & same G

b: (same cluster but different G)

c: (Sames G but differen C)

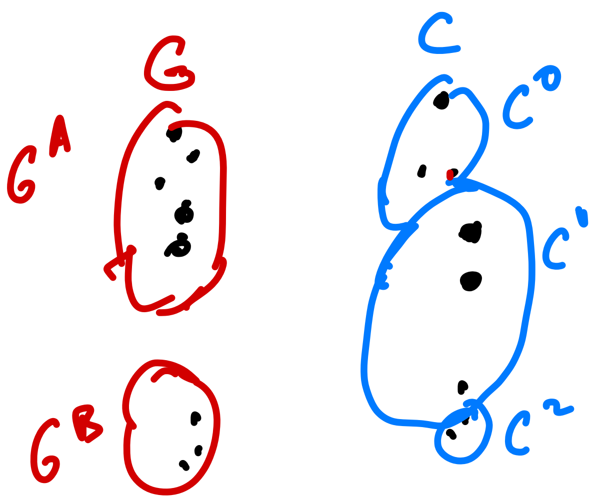
d: different G
& " C

Rand index:

$$RI = \frac{a+d}{a+b+c+d} = \frac{2(a+d)}{(N^2-N)}$$

Normalized Mutual Information

$$MI(G,C) = \sum_{l=1}^{K} \sum_{j=1}^{G} p(l,j) \log \frac{p(l,j)}{p(l)\,p(j)}$$



$$p(G^A) = \frac{5}{8}$$

$$p(C^1) = \frac{3}{8}$$

$$p(G^A, C^1) = \frac{2}{8}$$

$$\frac{2}{8} \log \frac{2/8}{5/8 \cdot 3/8}$$

$$NMI(G,C) = \frac{MI(G,C)}{[H(G) + H(C)]/2}$$

$$H(G) = -\sum_{j}^{G} p(j) \log(p(j))$$