Endotypes Discovery pipeline

Emilien JEMELEN (ENSAE), Emilie GERARD (Sanofi R&D)

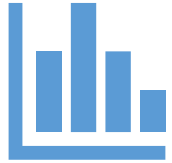
18th November 2022 - *Journées de biostatistique SFDS*

Contents

Introduction to the Endotypes Discovery project

Explanation of how the pipeline works

An example output of the pipeline



Performance assessment of the pipeline through simulated data

Introduction to the Endotypes Discovery project

The Endotypes Discovery (ED) project

❑ Clinical objective = Precision Medicine

Better understand heterogeneous diseases to identify which potential subgroups would better respond to the drug

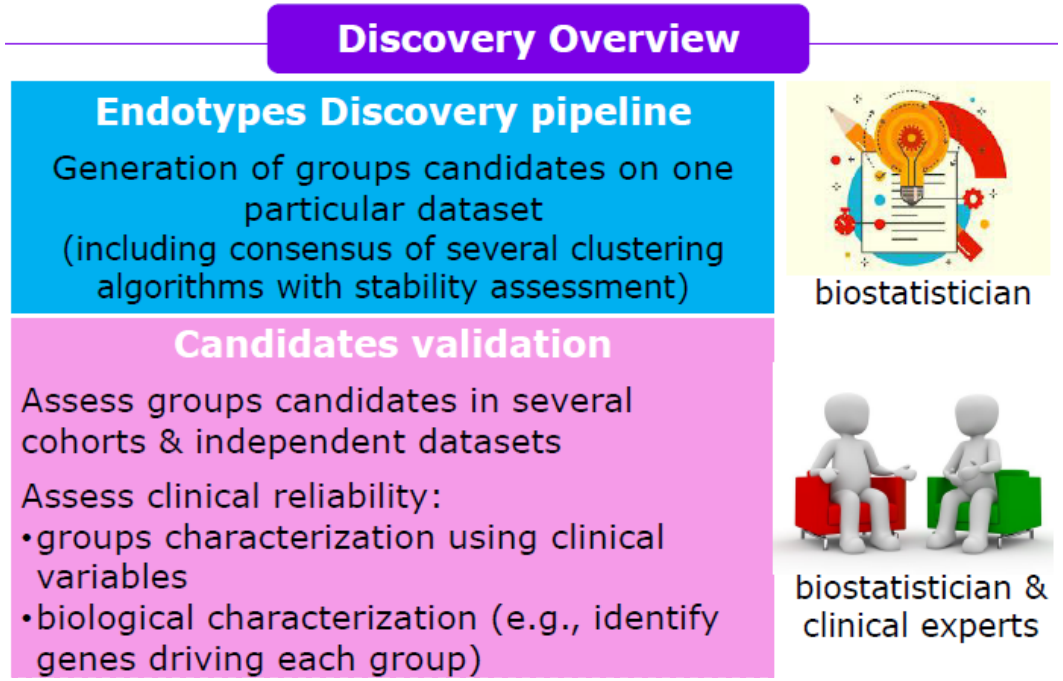
❑ Why using endotypes?

Phenotypes (groups of patients using clinical variables) are not always sufficient to identify these subgroups

⇒ look at omics/biomarker data (e.g. protein, gene)

Clustering : unsupervised ML problem

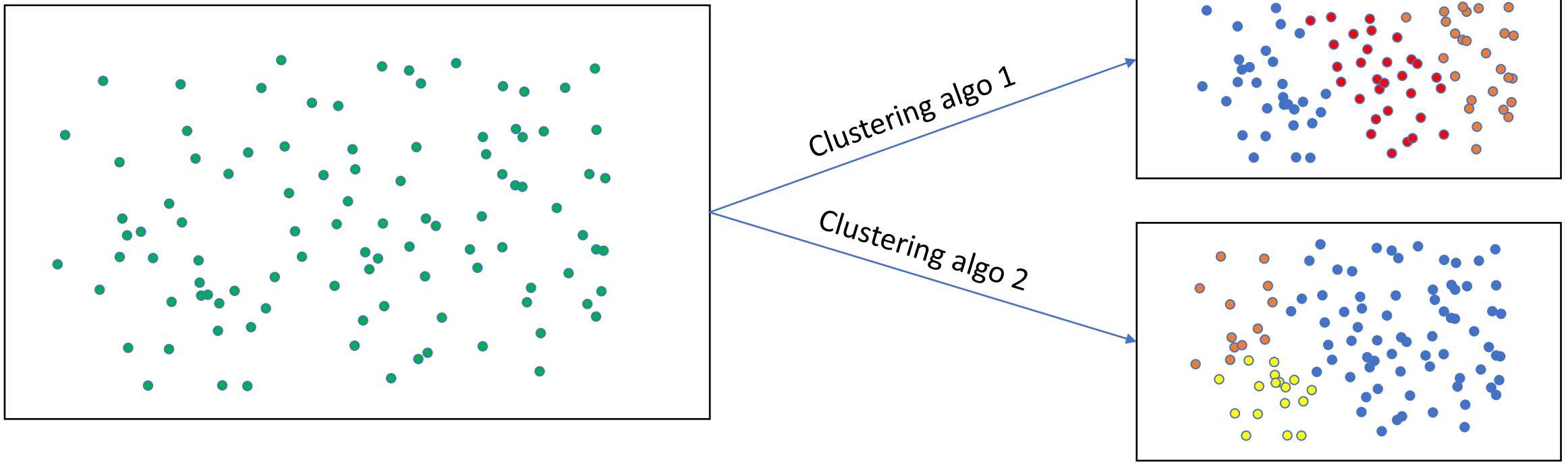
Challenging: not a lot of samples/patients, and **a lot of genes/features**
(e.g. 200 patients/5000 features)



Focus on the curse of high dimensionality for clustering

➔Fact: high dimensionality of the data tends to make the samples equidistant

➔Example of equidistant datapoints :



To sum up:

High dimensionality ⇒ Equidistant points ⇒ Arbitrary clustering

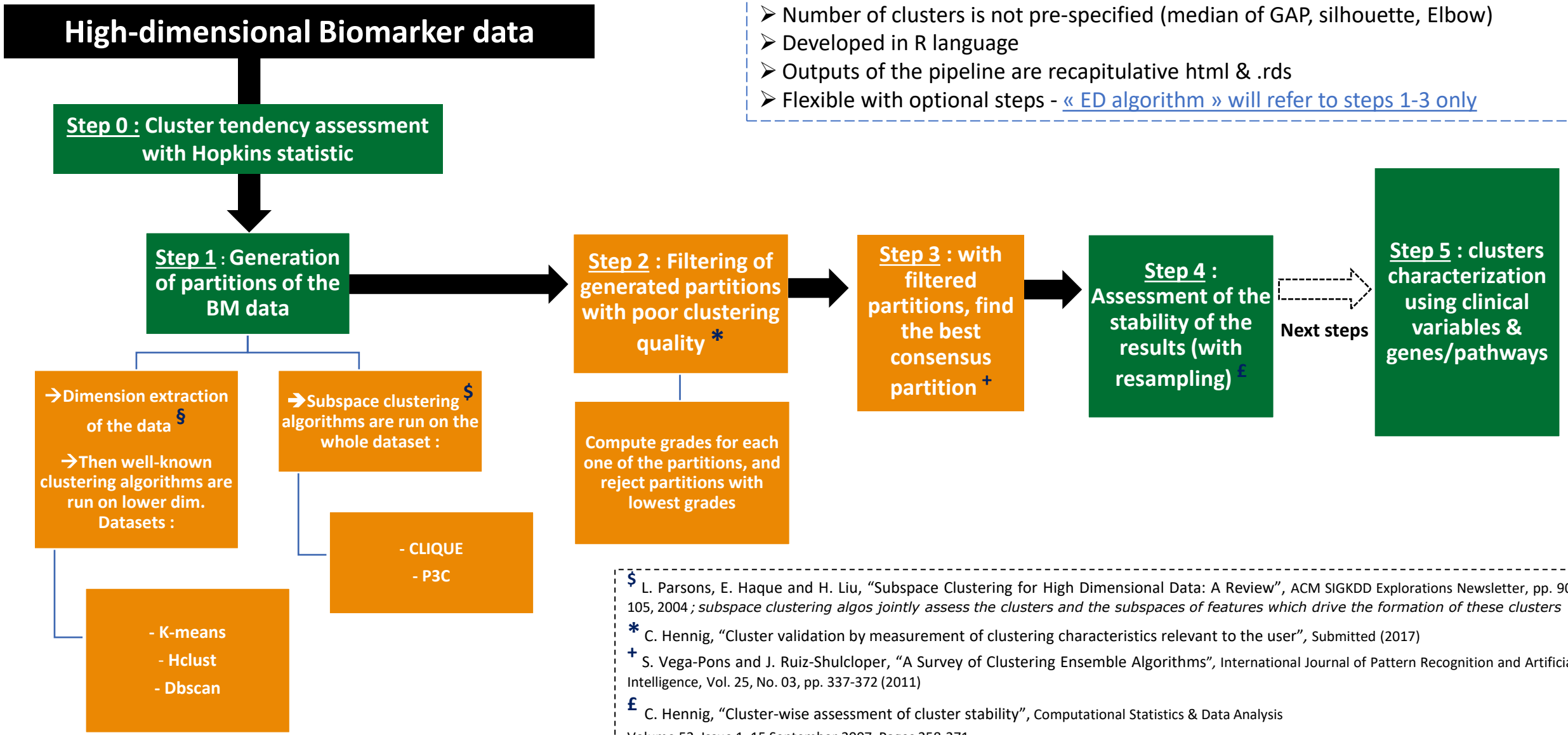
➔Which clustering partition is better?

None, both are arbitrary
Data are not separable

Explanation of how the pipeline works

Overview of the ED pipeline

- Number of clusters is not pre-specified (median of GAP, silhouette, Elbow)
- Developed in R language
- Outputs of the pipeline are recapitulative html & .rds
- Flexible with optional steps - « ED algorithm » will refer to steps 1-3 only



$ L. Parsons, E. Haque and H. Liu, "Subspace Clustering for High Dimensional Data: A Review", ACM SIGKDD Explorations Newsletter, pp. 90-105, 2004 ; *subspace clustering algos jointly assess the clusters and the subspaces of features which drive the formation of these clusters*

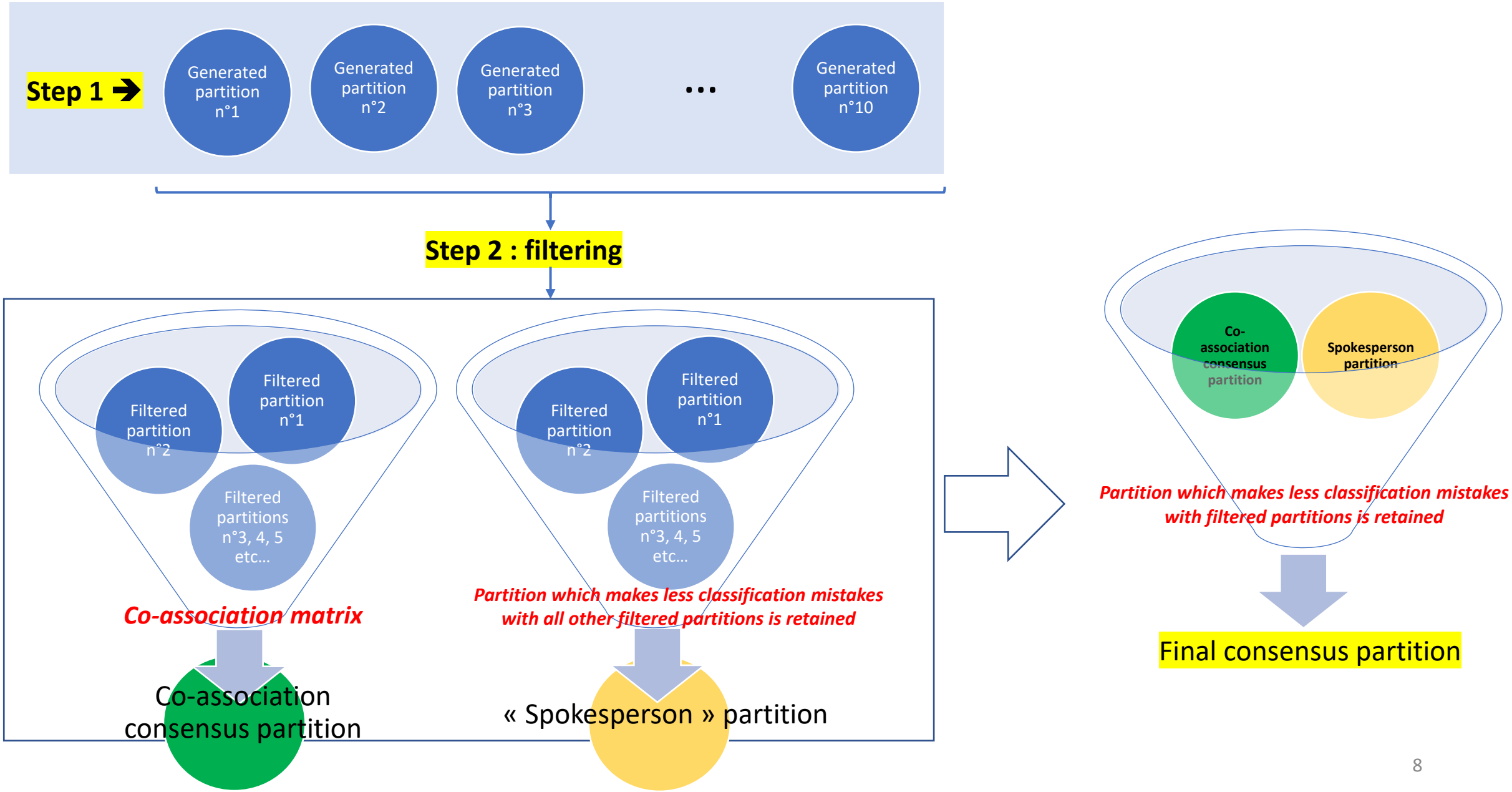
* C. Hennig, "Cluster validation by measurement of clustering characteristics relevant to the user", Submitted (2017)

\+ S. Vega-Pons and J. Ruiz-Shulcloper, "A Survey of Clustering Ensemble Algorithms", International Journal of Pattern Recognition and Artificial Intelligence, Vol. 25, No. 03, pp. 337-372 (2011)

£ C. Hennig, "Cluster-wise assessment of cluster stability", Computational Statistics & Data Analysis Volume 52, Issue 1, 15 September 2007, Pages 258-271

§ Several types of dimension extraction : auto-encoder, MDS (SKS model), PCA

Focus on Step 3 : finding a consensus partition of the data



An example output of the pipeline

Example of the .html output of the ED pipeline

Stability assessment of each one of the clusters

Source : *Cluster-wise assessment of cluster stability*, Christian Hennig

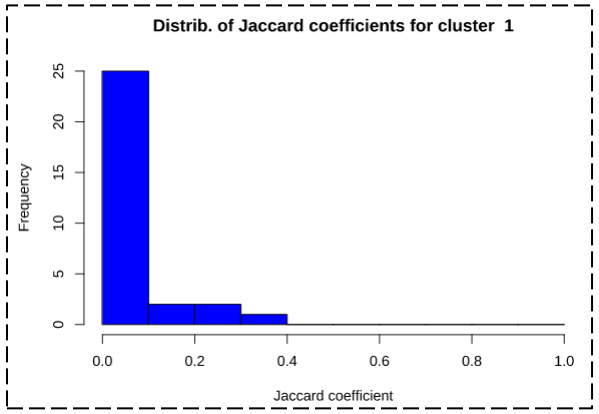
In this part we want to test whether the final partition returned by the Endotypes Discovery pipeline is robust in the face of resampling or not. Indeed, if a patient was classified in cluster 2, then it should be classified in cluster 2 again even if we only consider a smaller subset of the data. Therefore, for every cluster of the original partition, we compute Jaccard stability coefficients over iterated bootstraps of the data.

Interpretation : a cluster is deemed stable if its mean Jaccard coef. is >0.5

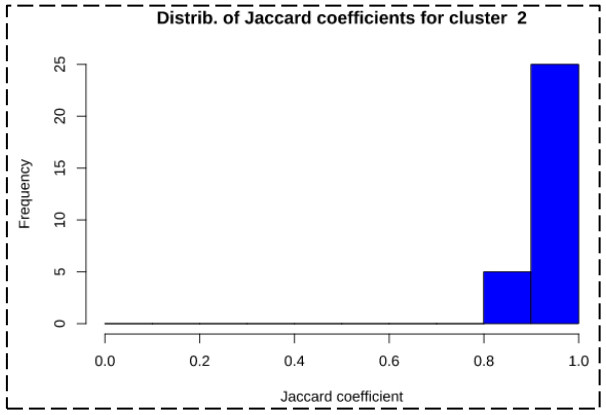
```
##     Cluster1          Cluster2        Cluster3
##  Min.   :0.009501   Min.   :0.8079   Min.   :0.2235
##  1st Qu.:0.019583   1st Qu.:0.9530   1st Qu.:0.8719
##  Median :0.032094   Median :0.9791   Median :0.9218
##  Mean   :0.064948   Mean   :0.9547   Mean   :0.8303
##  3rd Qu.:0.051439   3rd Qu.:0.9880   3rd Qu.:0.9523
##  Max.   :0.333333   Max.   :0.9975   Max.   :0.9744
```

Jaccard coef formula :
Given two cluster C1 and C2,

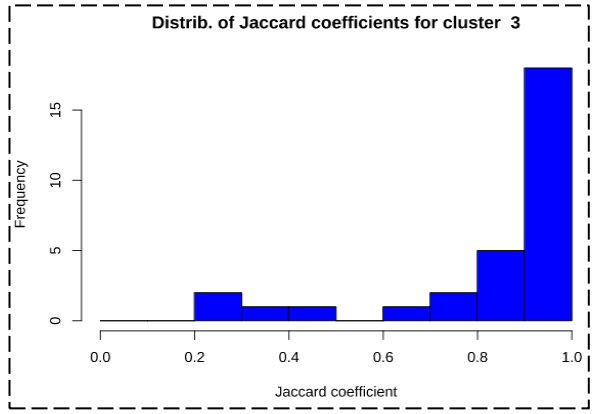
$$J: \ (C_1, C_2) \subseteq E^2 \mapsto \frac{card(C_1 \cap C_2)}{card(C_1 \cup C_2)} \in [0,1]$$



Unstable cluster



Stable cluster



Stable cluster

Performance assessment of the ED algorithm

Methodology for testing the performance of the algorithm (1/3)

➔With a program dedicated to simulating data (gaussian distribution with blocks of correlated features), we simulated 50 biomarkers datasets for each one of the 27 scenarii below :

	Number of patients	Number of genes	Number of driving genes
1	50	200	None
2	100	200	None
3	200	200	None
4	50	2000	None
5	100	2000	None
6	200	2000	None
7	50	5000	None
8	100	5000	None
9	200	5000	None
10	50	200	Some
11	100	200	Some
12	200	200	Some
13	50	2000	Some
14	100	2000	Some
15	200	2000	Some
16	50	5000	Some
17	100	5000	Some
18	200	5000	Some
19	50	200	Many
20	100	200	Many
21	200	200	Many
22	50	2000	Many
23	100	2000	Many
24	200	2000	Many
25	50	5000	Many
26	100	5000	Many
27	200	5000	Many

Number of genes	Driving genes category	Number of driving genes
200	Some	10
200	Many	50
2000	Some	20
2000	Many	200
5000	Some	50
5000	Many	500

Table of #driving genes given #genes and considered category

➔ For each scenario, each one of the 50 datasets simulated comes along with a **binary classification** of the patients driven by the active genes (those with a higher beta coef in the formula below*).

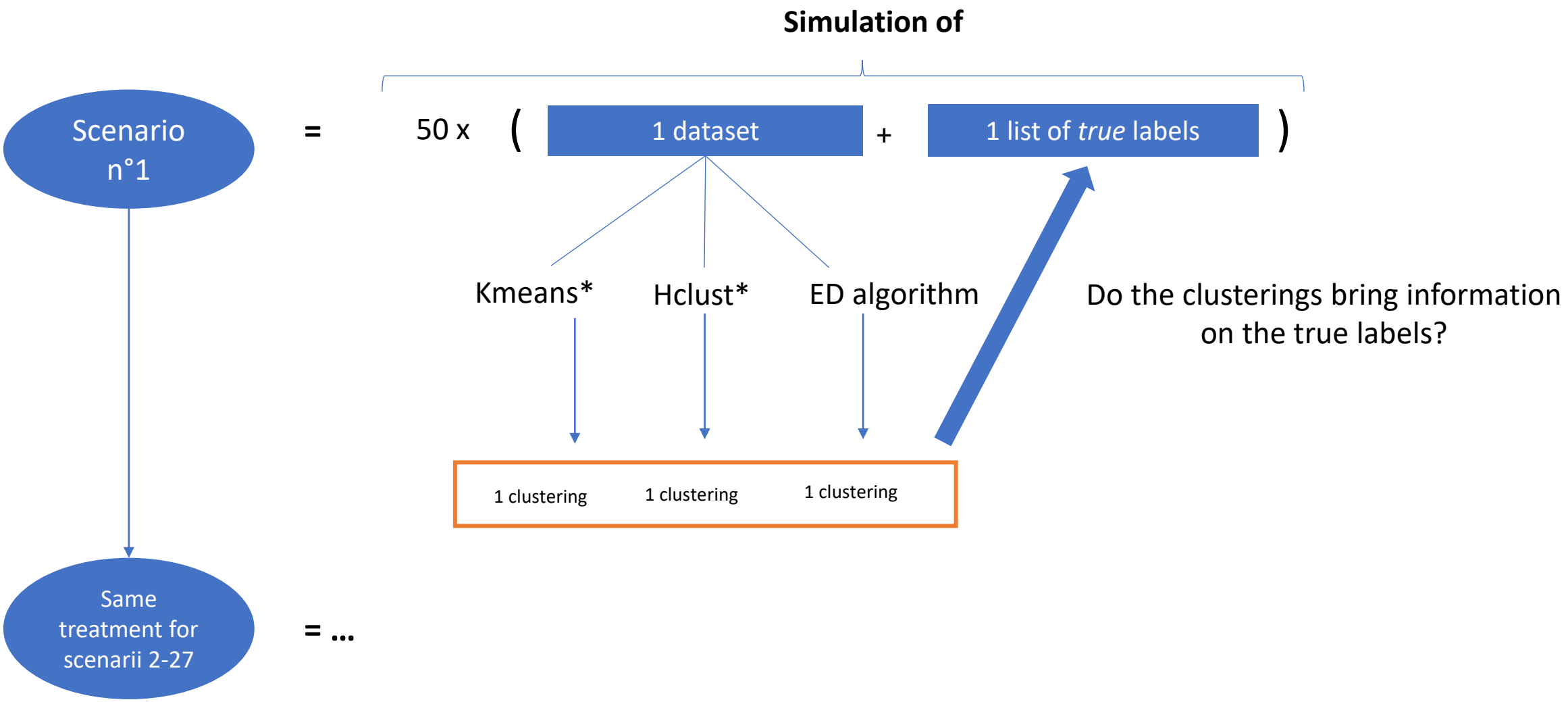
➔ This 0/1 classification allows us to get out of the unsupervised dead end and test for the ability of our ED algorithm to discover that 0/1 « true » classification.

* Binary classification model :

$$p(x) = \frac{\exp(\beta_0 + \beta_1 x)}{1 + \exp(\beta_0 + \beta_1 x)}, \qquad label(x) = \mathbb{1}(p(x) \geq \frac{1}{2})$$

➔Active genes have a coef. in β_1 set to 3, while other genes coef. in β_1 is set to 1

Methodology for testing the performance of the algorithm (2/3)



***:with number of cluster k=2**

Methodology for testing the performance of the algorithm (3/3)

➔For the scenarii under the null hypothesis H0 (no gene actually driving the simulated labels):

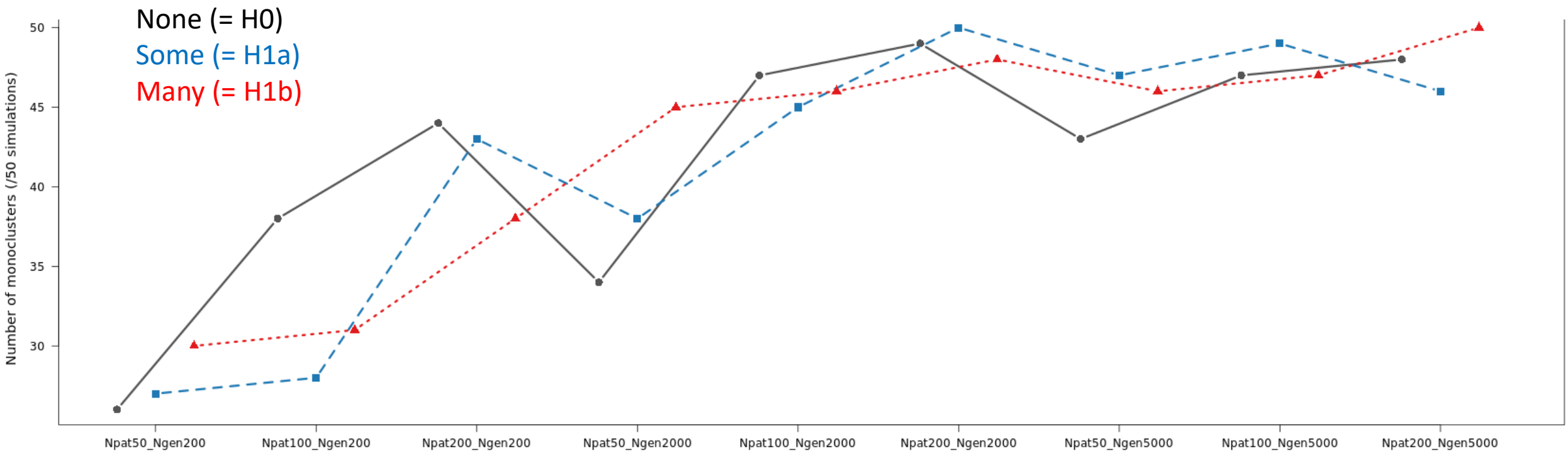
- We compute the **proportion of monocluster partitions** returned by the algorithms

➔For the scenarii in which there are active genes (H1), we test two things:

1. **Are the groups of patients (simulated labels = 1 or 0) stable ?**
   - i.e. have they been dissolved (we use a dissolution metric : Jaccard coefficient) resp. in the ED, Kmeans & Hclust clusterings

2. **Do the ED, Kmeans and Hclust partitions bring any information about the simulated labels?**
   - Fisher exact test with H0 : « clustering result and true labels are independent »
   - We count the significant p-values

In most scenarii, ED algorithm returns monocluster



➔Tendency that follows the growth of number of patients and genes

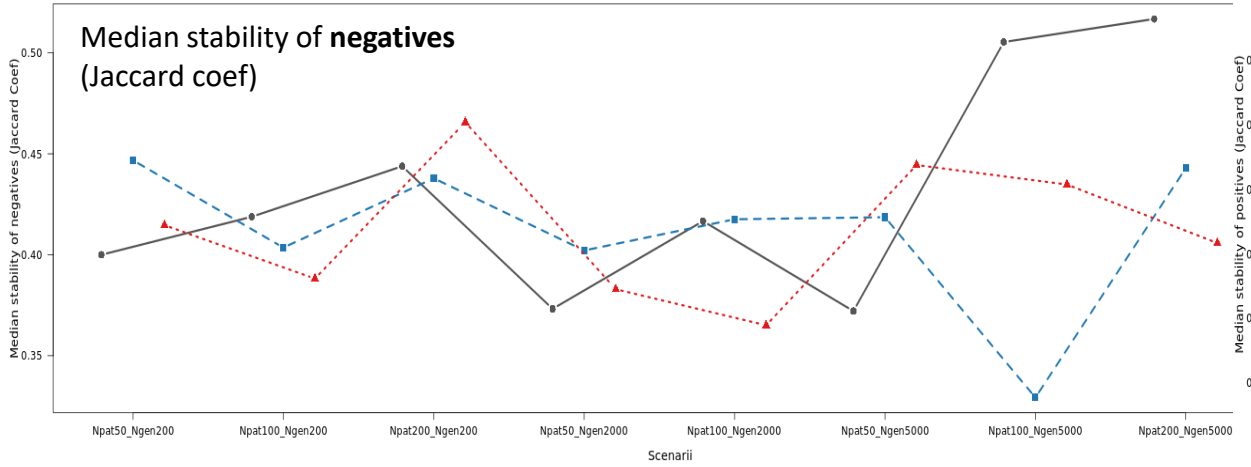
➔Indicator that the separation signal gets weaker ?

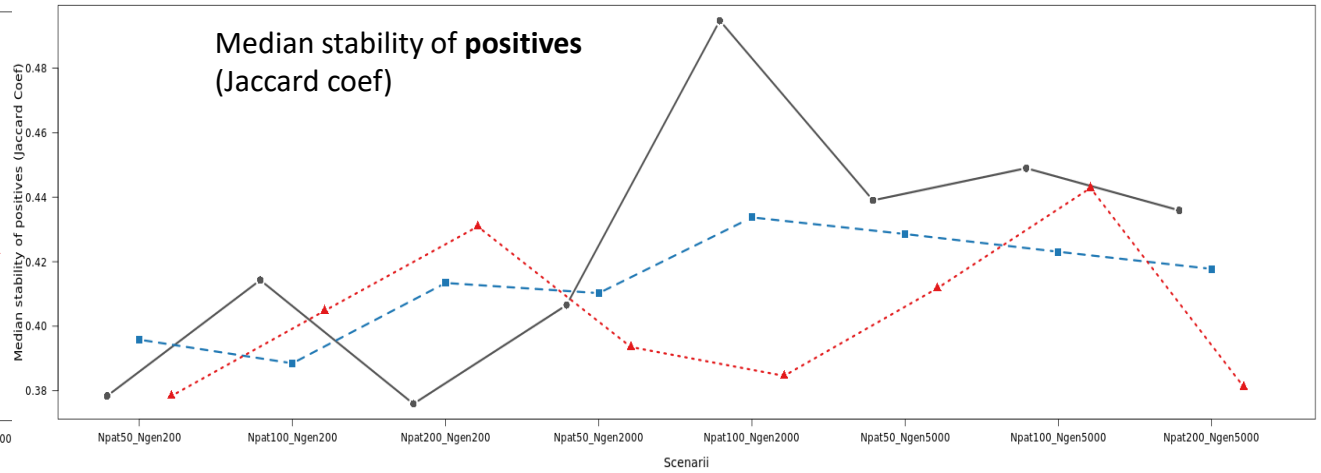
NB : Kmeans and Hclust were run with 2 clusters by default

Focus on alternative hypothesis scenarii

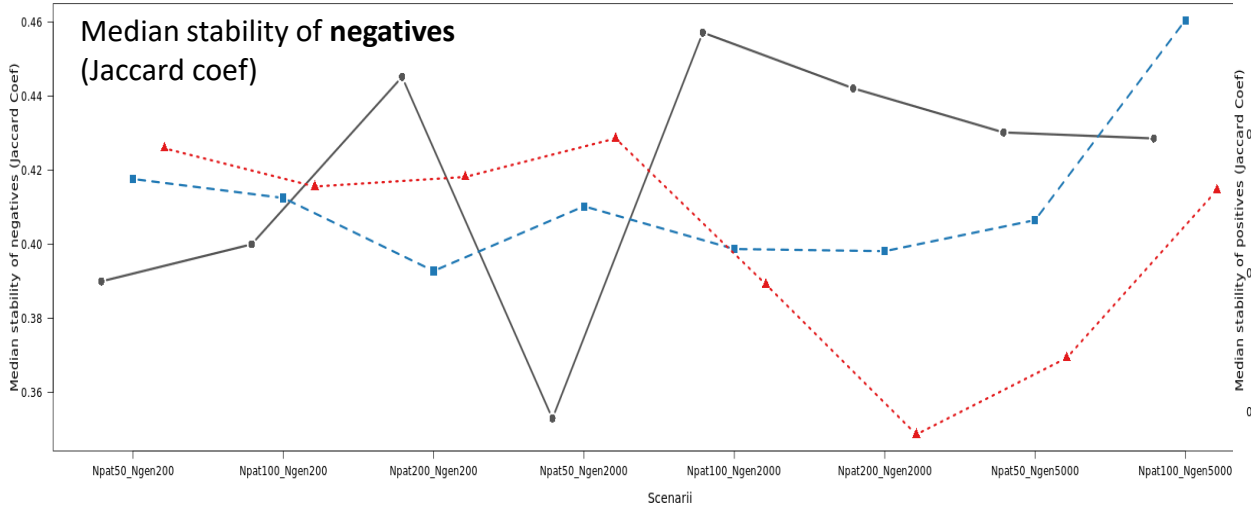
ED
Hclust
Kmeans

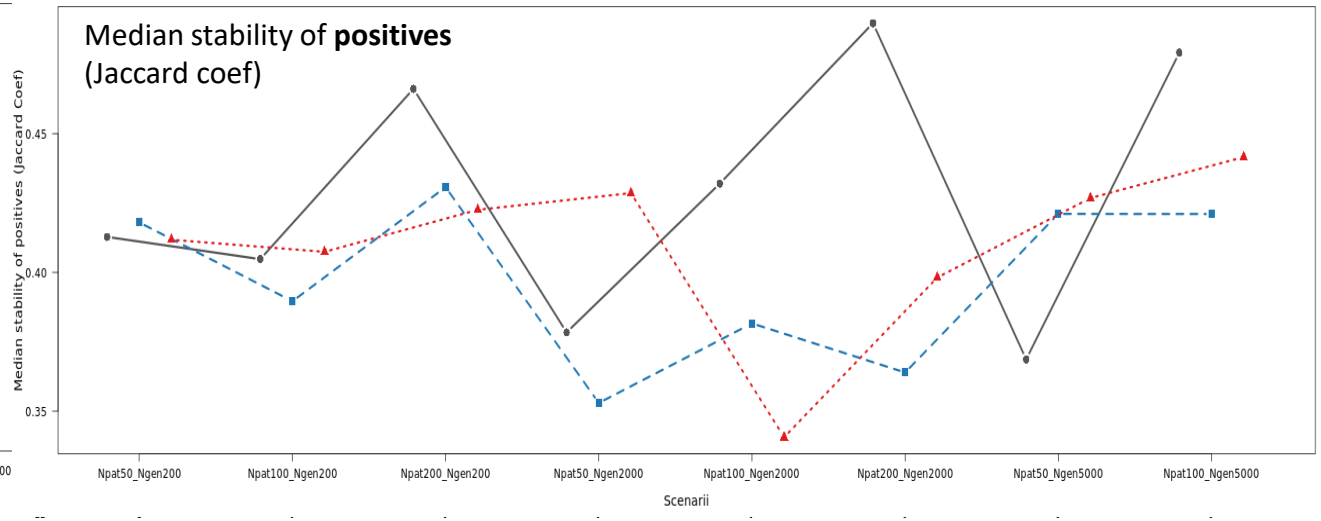
Some scenarii





Many scenarii





➔All coefficients are below 0.5 (not good)

➔Removing ED monoclusters ⇒ ED tends to be better with higher gene numbers and higher patients numbers

Conclusions

ED pipeline

1. User-friendly format for automation of endotypes discovery in high dimensional settings

2. Developed with optimally determined parameters to avoid arbitrary choices from the user

3. Flexible with independent functions which can be run separately by the user

4. Quality & stability assessments of generated partitions

5. Characterization of clusters using clinical variables

6. ED algorithm (steps 1-3) is optional, ED pipeline can be run on any clustering algorithm

Simulation results - Synthesis on ED algorithm

Properties	ED algorithm	Kmeans	Hclust
Number of clusters pre-specified	No	Yes	Yes
May return mono-cluster partitions	Yes	No	No
Simulation results after withdrawal of ED mono-cluster partitions **≤ 2000 genes**	+	+	+
Simulation results after withdrawal of ED mono-cluster partitions : very high-dim **> 2000 genes**	++	+	+

ED algorithm (steps 1-3) tends to be slightly better than classical clustering algorithms such as Kmeans and Hclust in some scenarii

Limitation: number of clusters of Kmeans & Hclust were pre-specified

Food for thought and next steps

➔ **Monocluster partitions:** corrective steps have been tried

- Increasing the association coefficient in the co-association matrix consensus method (step 3)

Generation → filtering poor quality → **consensus** → stability assessment → cluster characterization

Result: higher number of clusters generated by co-association matrix but the spokesperson partition was often selected (and was often a monocluster …)

➔ **On toy datasets or other high-dimensional data (internal project), ED algorithm returned ≥ 2 clusters:**

- Signal not strong enough in the simulated data ?

⇒ More simulations with different parameters are needed (more active genes with higher beta coef. for instance, no pre-specification of number of clusters for Kmeans & Hclust, …)

Questions / Feedbacks

Appendix

- Model of genes correlation for the simulation of biomarker data
- Additional simulation results of ED algorithm
- Subspace clustering algorithms overview (step 1)
- Overview of the discrimination of poor quality generated partitions (step 2)
- Formula of Mirkin distance and median partition approach (step 3)
- Clusters stability assessment (Jaccard coefficients bootstrap) (step 4)

Model of genes correlation for the simulation of biomarker data

$$M_{exemple} = \left[m_{i,j}\right]_{(i,j)\in[\![1,7]\!]^2}$$

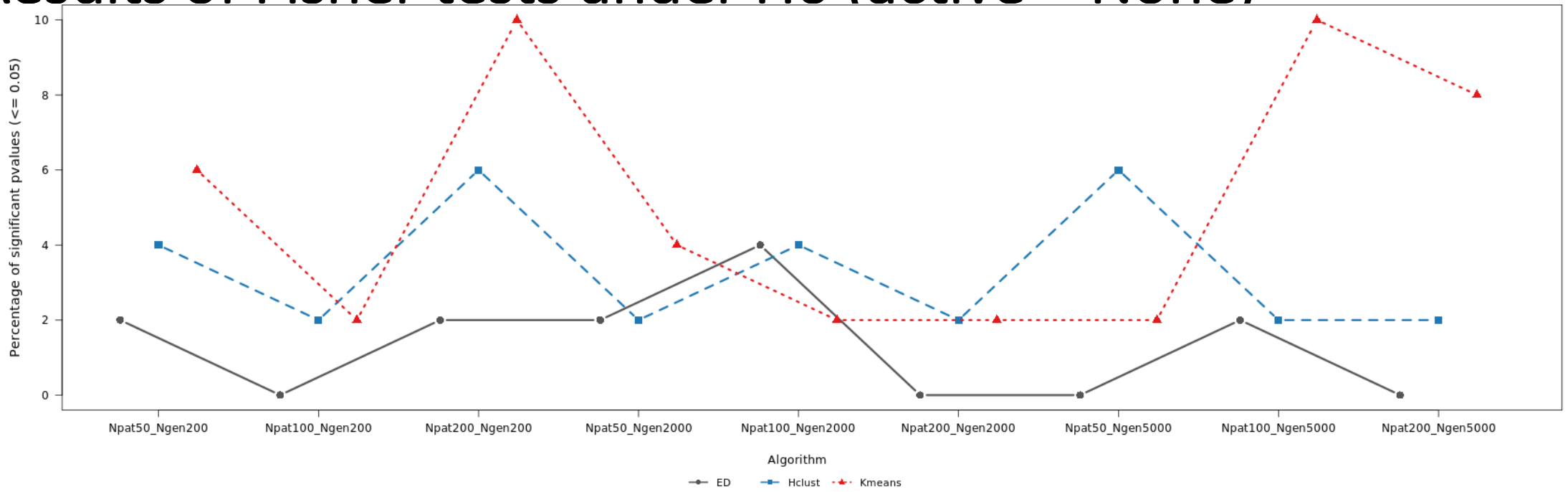
$$\text{Où } \forall (i,j) \in [\![1,7]\!]^2, m_{i,j} \overset{\text{def}}{\equiv} corr(biomarqueur\ n°i, biomarqueur\ n°j)$$

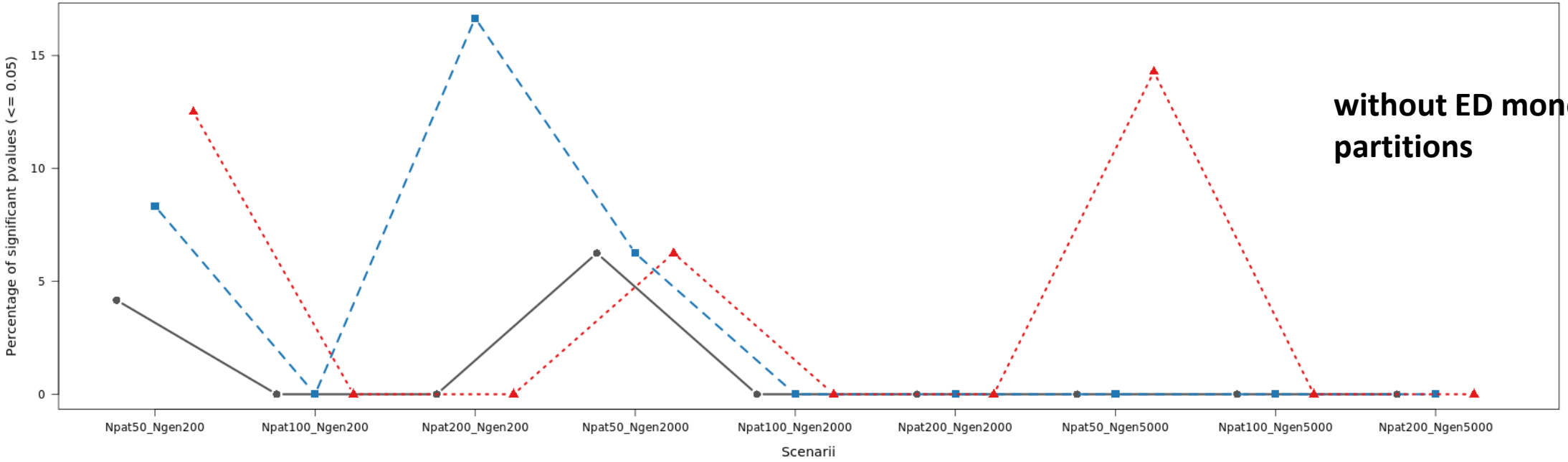
$$M_{exemple} = \begin{pmatrix} 1 & \rho & \rho^2 & 0 & 0 & 0 & 0 \\ \rho & 1 & \rho & 0 & 0 & 0 & 0 \\ \rho^2 & \rho & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \rho & \rho^2 & \rho^3 \\ 0 & 0 & 0 & \rho & 1 & \rho & \rho^2 \\ 0 & 0 & 0 & \rho^2 & \rho & 1 & \rho \\ 0 & 0 & 0 & \rho^3 & \rho^2 & \rho & 1 \end{pmatrix}$$

In simulations, $\rho = 0.7$

Results of Fisher tests under H0 (active = None)

ED
Hclust
Kmeans

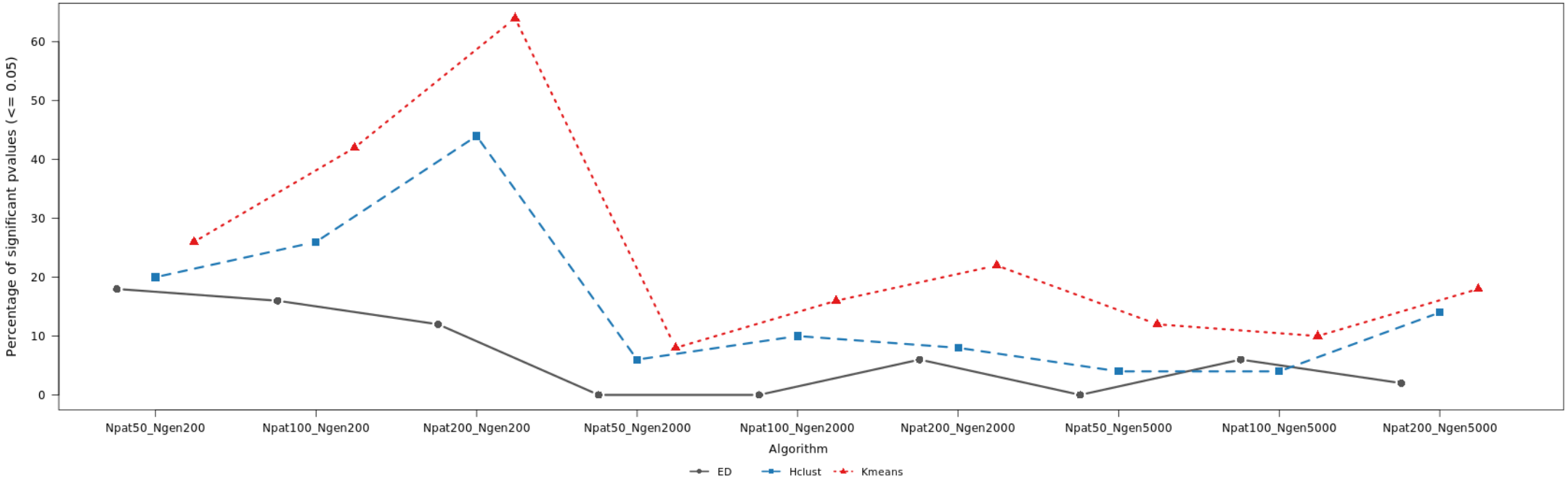


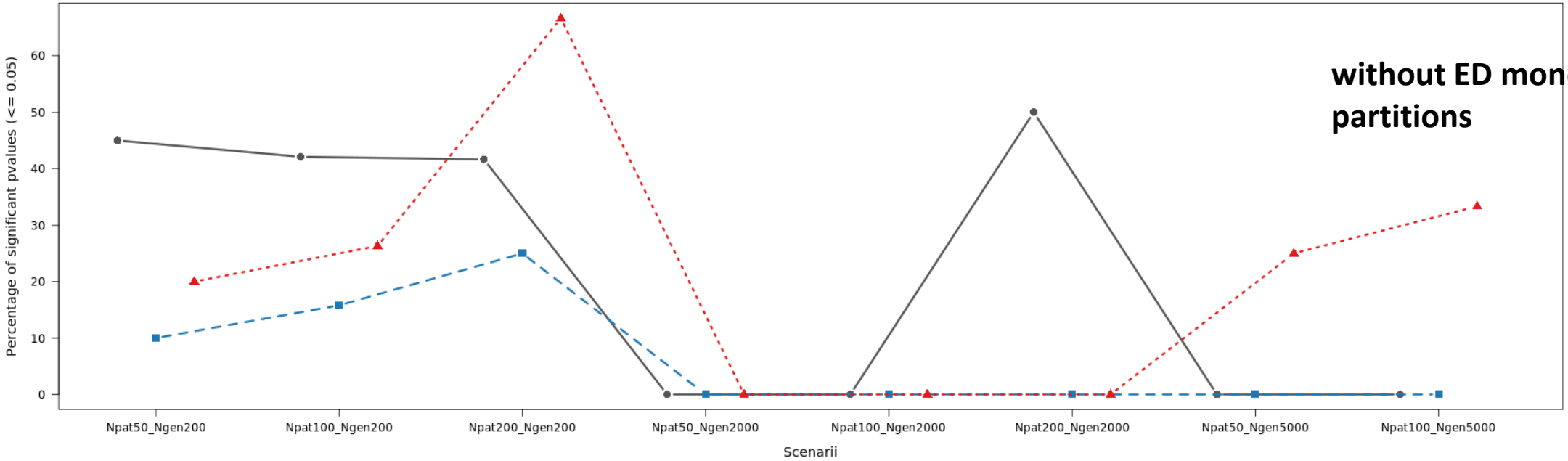


without ED monocluster partitions

Results of Fisher tests under H1 (active = Many)

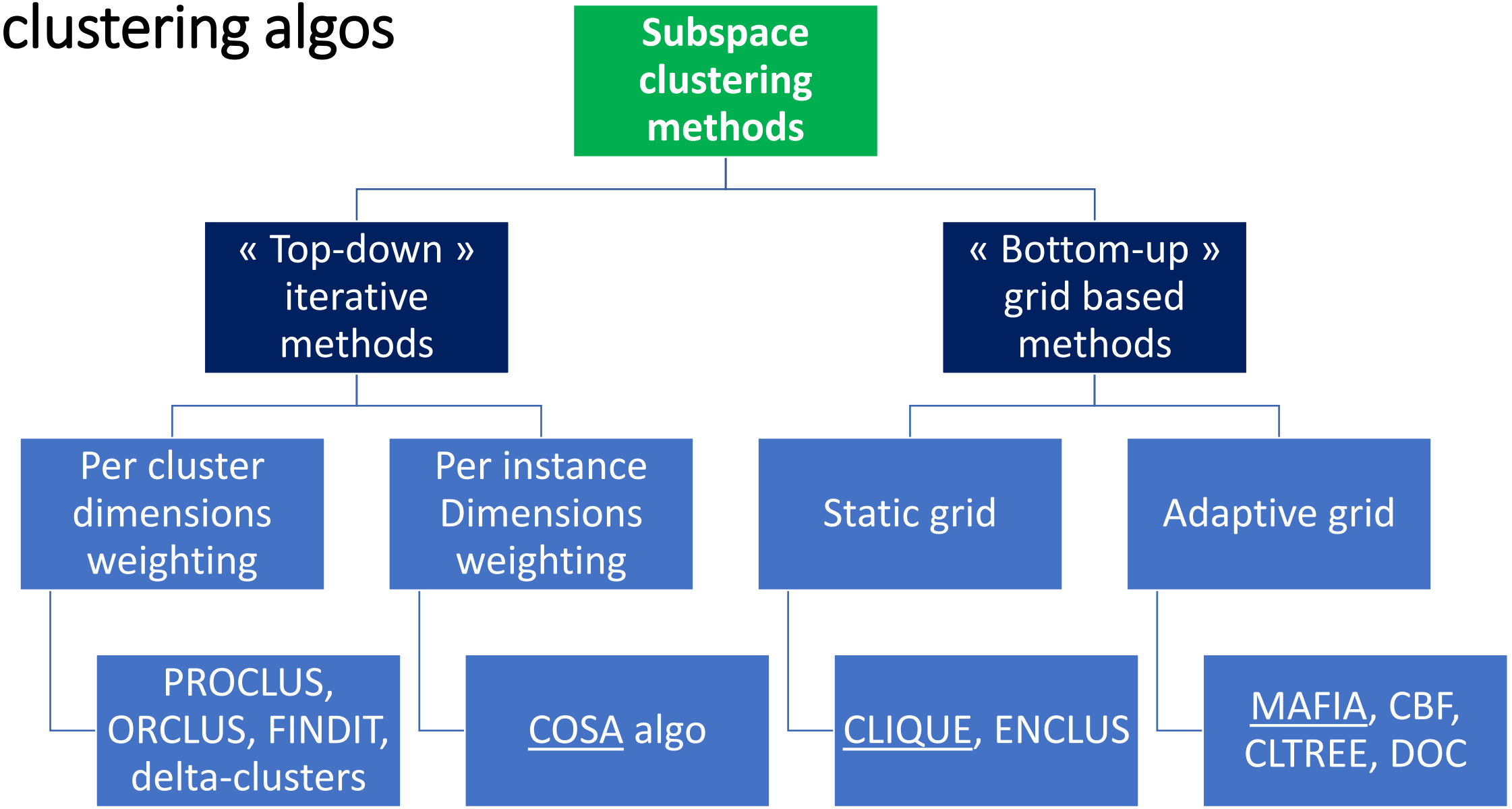
ED
Hclust
Kmeans





without ED monocluster partitions

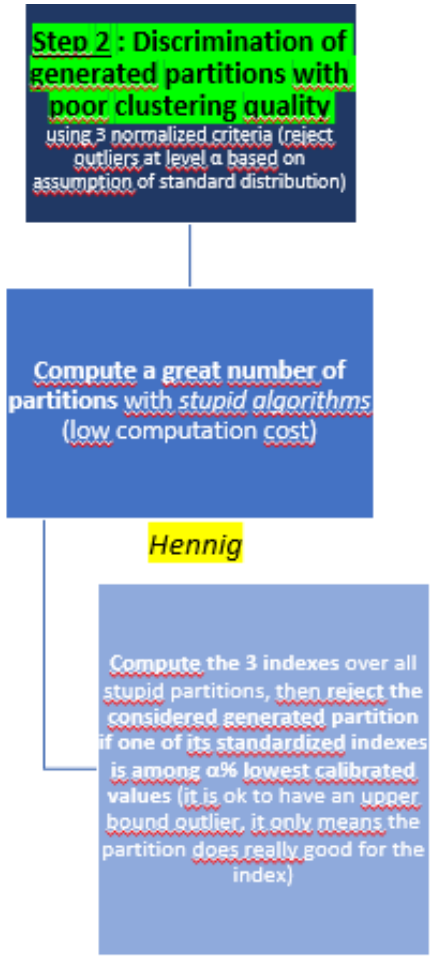
Clustering adapted for high-dim data : review of subspace clustering algos



Discrimination of generated partitions: density modes and valleys

Cf **Cluster validation by measurement of clustering characteristics relevant to the user**, C. Hennig. Section 3.6. “Density modes and valleys”.

Architecture of step 2



Meaning of each index

➔**I_densdec**: For every cluster, starting from the cluster mode, i.e., the observation with the highest density, construct a growing sequence of observations that eventually covers the whole cluster by always adding the closest observation that is not yet included. Optimally, in this process, the within-cluster density of newly included points should always decrease. Whenever actually the density goes up, a penalty of the squared difference of the densities is incurred. The index Idensdec aggregates these penalties.

➔**I_densbound**: index that penalises a high contribution of points from different clusters to the density values in a cluster, because this means that the cluster border cuts through a high density region.

➔**I_highdgap**: an issue with Idensdec is that it is possible that there is a large gap between two observations with high density, which does not incur penalties if there are no low-density observations in between. This is picked up during computation of I_densdec by I_highdgap.

NB : all three indexes are normalized such that a higher index is better.

A median partition approach : consensus obtained with Mirkin distance as similarity measure

Generally speaking, obtaining a consensus via median partition approach means resolving the following problem :

$$argmax_{P\in\mathbb{P}_X}\left(\sum_{j=1}^{m}\Gamma(\mathrm{P},\mathrm{P_j})\right)$$

Where $\mathbb{P}_X$ is a subset of $\mathbb{N}^n$, $(P_j)_{j\in[\![1,m]\!]}$ is a collection of partitions we think might contain information on the true classification of the data (partitions generated by clustering algos for instance), Γ is a similarity measure between two partitions.

Mirkin distance is merely a similarity measure, defined as below :

$$\begin{cases} M:\ (\mathbb{N}^n)^2 \to \mathbb{R}_- \\ M: P_a, P_b \mapsto (-1).card\left(\{(x,y)\in data: \left(P_a(x)=P_a(y)\right)\wedge\left(P_b(x)\ \neq P_b(y)\right)\}\right) \\ \qquad -card\left(\{(x,y)\in data: \left(P_b(x)=P_b(y)\right)\wedge\left(P_a(x)\ \neq P_a(y)\right)\}\right) \end{cases}$$

Qualitatively, Mirkin distance counts classification mistakes for pairs of points between two partitions. Indeed, for a given pair of point, two partitions A and B disagree either if the pair is in the same cluster in A but not in B or if the pair is in the same cluster in B but not in A.

NB: $P_a(x)$ is the label of sample x in partition P_a

Cluster-wise assessment of cluster stability – HENNIG

C. Hennig, "Cluster-wise assessment of cluster stability", Computational Statistics & Data Analysis - Volume 52, Issue 1, 15 September 2007, Pages 258-271

Algo :
Let B = #bootstrap draws, let C be a cluster of $E_n(X_n)$.

For i in [|1, B|] :

1. Draw a bootstrap sample $X_{i,n}$ of n points with replacement from the original dataset X_n
2. Let $X_{i,n}$* be $X_{i,n}$ without duplicates (keep 1 out of 2)
3. Compute the clustering $E_n(X_{i,n})$
4. Let $C_i^* = C \cap X_{i,n}^*$ (original cluster restricted to the context of the bootstrap sample)
5. Si $C_i^* \neq \emptyset$,
   $\gamma_{C,i} = \max(\gamma(C_i^*, D))$, $D \in E_n(X_{i,n})$
   Else, $\gamma_{C,i} = 0$

This generates $(\gamma_{C,i})_{i \in \{1, \dots, B\}}$ on which it is possible to compute the mean as a stability measure for cluster C :

$$\overline{\gamma}_C = \frac{1}{B^*} \sum_{i=1}^{B} \gamma_{C,i}$$

Where B* = $card(\{i \in [\![1, B]\!], C_i^* \neq \emptyset\})$

➔According to Hennig **B = 50** is enough most of the time
$\overline{\gamma}_C$ <0.6 (should not be trusted) ; $\in [0.6, 0.75[$ (uncertain labels) ; $\in [0.75, 0.85[$ (valid, stable cluster), >0.85 (highly stable cluster)