Clustering market regimes using the Wasserstein distance [1]

Seminar for the Phd course in Quantitative Finance (SNS)

Alessandro Batignani

October 6, 2024

Link for the code: https://github.com/alebati3

Clusters analysis

- Cluster analysis or Clustering is an unsupervised technique used to group *objects* into *clusters*.
- It's fundamental to define a way to quantify the similarity among objects.
    - It extends the concept of distance.
- "The definition of an optimal clustering is not well defined, and in the case of financial data, this is certainly true." [1]
- "Heuristically, we would like individual clusters to contain objects that are similar to each other whilst being distinct from objects in other clusters". [1]

k-means algorithm

- Suppose $X = \{(x_1, \ldots, x_N) : x_i \in V\}$, where $(V, \|\cdot\|_V)$ is a normed vector space . We further assume that each $x_i = (x_i^1, \ldots, x_i^d)$ has been standardized coordinate-wise, that is,

$$\mathbb{E}[(x_i^j)_{1 \le i \le N}] = 0 \quad \text{and} \quad \mathrm{Var}((x_i^j)_{1 \le i \le N}) = 1 \quad \text{for } j = 1, \ldots, d.$$

- The *k-means clustering algorithm* is an unsupervised vector quantization method which assigns elements of X to k disjoint clusters. Each of these clusters is defined by central elements $\bar{x} = \{\bar{x}_j\}_{j=1,\ldots,k}$ called *centroids*, which are initially sampled from X.

k-means algorithm

- At each step $n \in \mathbb{N}$ of the algorithm, one first calculates the *nearest neighbours*

$$C_l^n := \left\{ x_i \in X : \arg \min_{j=1,\dots,k} d(x_i, \bar{x}_j^{n-1}) = l \right\}$$

  associated to each $\bar{x}_l^{n-1}$ for $l = 1, \dots, k$.
- Each set C_l^n is then aggregated into a new centroid x_l^n for $l = 1, \dots, k$ via a function $\alpha : 2^V \to V$, so

$$\bar{x}_l^n := \alpha(C_l^n) \quad \text{for } l = 1, \dots, k.$$

- In the classical k-means on $\mathbb{R}^d$, we take as new centroid the barycenter of C_l

$$\alpha(C_l) = \left(\frac{1}{|C_l|} \sum_{x_j \in C_l} x_j \right)_{1 \le j \le d}.$$

  where, $|C_l|$ denotes the cardinality of the set C_l.

k-means algorithm

- For a given tolerance level $\epsilon > 0$ and a loss function $l : V^k \times V^k \to [0, +\infty)$, the k-means algorithm terminates at step $n \in \mathbb{N}$ if the stopping condition

$$l(\bar{x}^n, \bar{x}^{n-1}) < \epsilon$$

  is satisfied.
- The loss function l is given by

$$l(x, y) = \sum_{i=1}^{k} \|x_i - y_i\|_V,$$

- At the end, the algorithm outputs the final clusters $C^* = \{C_l^n\}_{l=1,\dots,k}$ and the k-quantization $\bar{x}^n = \{x_l^n\}_{l=1,\dots,k}$.

The market regime clustering problem (MRCP)

Given the return series of a security price $\mathbf{r} = (r_0, r_1, \ldots, r_N)$

- The MRCP is defined as the task of classifying segments of return series $(I_i)_{i=1}^M$, where

$$I_i = \left(r_i^1, \ldots, r_i^n\right) \quad \text{for} \quad n \in \mathbb{N}$$

Any vector $I_i \in \mathbb{R}^n$ can be associated to an empirical probability measure

$$\mu_i = \frac{1}{n} \sum_{j=1}^{n} \delta_{r_i^j}$$

for $i = 1, \ldots, M$ with n atoms. Thus the problem of classifying market regimes is equivalent to assigning a label to probability measures $(\mu_i)_{i=1}^M$.

Problem setting and notation

Given the return series $\mathbf{r} = (r_j)_{j=0}^{N-1}$, where $r_j = \log(s_{j+1}) - \log(s_j)$, we define the segments of the return series as follows:

- if $h_1, h_2 \in \mathbb{N}$ with $h_1 > h_2$ then

$$l_i = \left(r_{(h_1-h_2)(i-1)}, \ldots, r_{(h_1-h_2)(i-1)+h_1}\right) \quad \text{for} \quad i = 1, \ldots, M$$

  where M is the maximum number of partitions with length $h_1 + 1$ that can be extracted from the return series with sliding offset parameter h_2.

- In general the different partitions l_i may overlap.

Problem setting and notation

- We can associate to each segment of data l_i the empirical measure μ_i for $i = 1, \ldots, M$. This gives us a family of measures:

$$\mathcal{K} = \{\mu_1, \ldots, \mu_M\}.$$

  It is this family $\mathcal{K}$ which will be the subject of our clustering algorithm.

- Let Q^j be the function which extracts the j-th order statistic of l_i, for $j = 1, \ldots, h_1 + 1$. Then, the cumulative distribution function of the *empirical measure* μ_i associated to l_i is defined as:

$$\mu_i((-\infty, r]) = \frac{1}{h_1 + 1} \sum_{j=1}^{h_1+1} \chi(Q^j(l_i) \leq r),$$

  where $\chi : \mathbb{R} \to [0, 1]$ is the indicator function.

p-Wasserstein distance (W_p)

Main properties:

- W_p is a metric in the set of probability measures having the first p moments finite, denoted by $\mathcal{P}_p(\mathbb{R}^d)$.
- Convergence with respect to W_p is equivalent to the usual weak convergence of measures plus convergence of the first p-th moments.
- Intuition: connection with the optimal transport problem.

Connection with the optimal transport problem

- The *optimal transport problem* aims to transform the distribution of mass $\mu(x)$ into another distribution $\nu(y)$ on a space X, while minimizing the cost.
- The function $c(x, y) \geq 0$ represents the cost of moving mass from point x to point y. A transport plan $\gamma(x, y)$ defines how much mass is moved from x to y.
- The total cost of the transport plan γ is:

$$\int \int c(x, y)\gamma(x, y)\, dx\, dy = \int c(x, y)\, d\gamma(x, y).$$

- The *optimal transport plan* minimizes the total cost:

$$C = \min_{\gamma \in \Gamma(\mu,\nu)} \int c(x, y)\, d\gamma(x, y).$$

- If $c(x, y) = d(x, y)^p$, then C corresponds to the W_p.

W_p for empirical probability measure

- Suppose $\mu, \nu \in \mathcal{P}_p(\mathbb{R}^d)$ and let $d = 1$. Moreover, suppose that μ, ν are absolutely continuous with respect to the Lebesgue measure on $\mathbb{R}$. Then, the p-Wasserstein distance $W_p(\mu, \nu)$ is given by

$$W_p(\mu, \nu) = \left(\int_0^1 \left| F_\mu^{-1}(z) - F_\nu^{-1}(z) \right|^p dz \right)^{1/p},$$

where the quantile function $F_\mu^{-1} : [0, 1) \to \mathbb{R}$ is defined as

$$F_\mu^{-1}(z) = \inf\{x : F_\mu(x) \geq z\}.$$

- If μ, ν are empirical measures with equal numbers of atoms $N \in \mathbb{N}$, with $(\alpha_i)_{1 \leq i \leq N}$ and $(\beta_i)_{1 \leq i \leq N}$ their corresponding order statistics, then

$$W_p(\mu, \nu)^p = \frac{1}{N} \sum_{i=1}^{N} |\alpha_i - \beta_i|^p .$$

W_p for empirical probability measure

- Calculating the Wasserstein distance between two empirical measures can be done in linear time, assuming the atoms of each measure are already sorted ascending. If not, is an $\mathcal{O}(N \log N)$ operation, where N is the number of atoms.

- Suppose that $\{\mu_i\}_{1 \leq i \leq M}$ are a family of empirical probability measures, each with order statistics $\{\alpha_j^i\}_{1 \leq j \leq N}$. We can define the Wasserstein barycenter as the probability measure $\bar{\mu}$ that minimizes the sum of p-Wasserstein distances to each μ_i.
- In particular $\bar{\mu}$ is charaterazed by the following the order statistics

$$a_j = \text{Median}\left(\alpha_j^1, \ldots, \alpha_j^M\right) \quad \text{for } j = 1, \ldots, N.$$

Wasserstein k-means algorithm

$$\text{Set of vectors} \longrightarrow \mathcal{K} = \{\mu_1, \dots, \mu_M\}$$

$$\|\cdot\|_V \longrightarrow p\text{-Wasserstein distance}$$

$$\begin{matrix}\text{Aggregation function} \\ \text{to update centroids}\end{matrix} \longrightarrow \text{Wasserstein barycenter}$$

The last specification we need to make is regarding the loss function.

- The most natural choice is to replace the distance induced by the norm on V with p-Wasserstein distance

$$I(\bar{\mu}^{n-1}, \bar{\mu}^n) = \sum_{i=1}^{k} W_p(\bar{\mu}_i^{n-1}, \bar{\mu}_i^n).$$

where $\bar{\mu}^n = (\bar{\mu}_i^n)_{1 \le i \le k}$ are the centroids obtained after step n of the Wasserstein k-means algorithm

Alternative clustering algorithms as benchmarks

- k-means with statistical moments (Moment k-means)
- Hidden Markov models
- Agglomerative Hierarchical clustering
  - W_p based (W-Hierchical clustering)
  - statistical moments based (Moment-Hierchical clustering)

Moment k-means

- A natural and more classical approach to clustering regimes may involve studying the first $p \in \mathbb{N}$ raw moments associated to each measure $\mu \in \mathcal{K}$
- each empirical probability measure μ_i is mapped in vector of $\mathbb{R}^p$, whose components are the corresponding first p moments

$$\varphi^p(\mu_i) = \left(\int_{\mathbb{R}} x^n \mu_i(dx) \right)_{1 \leq n \leq p},$$

- Thus, for a given $p \geq 1$, we obtain

$$\varphi^p(\mathcal{K}) = \{\varphi^p(\mu_1), \ldots, \varphi^p(\mu_M) : \varphi^p(\mu_i) \in \mathbb{R}^p \text{ for } i = 1, \ldots, M\}.$$

- After standardising each element of $\varphi^p(\mathcal{K})$ component-wise, we obtain a clustering set on $\mathbb{R}^p$, which we can apply the standard k-means algorithm to.

Hidden Markov models

- A more classical approach to market regime clustering involves fitting a *Hidden Markov model (HMM)* to observed time series data $\mathbf{r}$
  - We assume the existence of $k \in \mathbb{N}$ hidden latent states $\{1, \ldots, k\}$ which govern the dynamics of $\mathbf{r}$. The transitions between the latent states are assumed Markovian.
  - The aim is to determine the sequence of hidden states that most likely produced the observed data.
- HMM does not cluster segments of return series; instead, it associates a given latent state to a single return at time t.

Hidden Markov models

Model parameters:

- **Transition probabilities A**: probabilities of transitioning from one hidden state to another.
- **Emission probability densities B**: represented by $f(r_t|z_t)$, where z_t is the given latent state and r_t is the observed return.
  - $f(x|z_t)$ are assumed to be gaussians (*Gaussian HMM*)
- **Initial state distribution π**: the probabilities of starting in a particular hidden state.

The triplet $\lambda = (\mathbf{A}, \mathbf{B}, \boldsymbol{\pi})$ defines the Hidden Markov model.

Solution:

- find the best model λ^* by maximizing the Likelihood $\mathcal{L}(\lambda|\mathbf{r})$.
- Compute the most probable sequence of hidden states using the *Viterbi algorithm*.

Agglomerative Hierarchical clustering

- Start with the points as individual clusters;
- at each step, merge most similar pair of clusters until only one cluster (or k clusters) left;
- a key point is to define the Inter-Cluster Similarity.
    - Different approaches to defining the similarity between clusters disinguish the different algorithms
- The standard algorithm for Hierarchical Agglomerative clustering (HAC) has a complexity of $\mathcal{O}(N^3)$, where N is the number of objects.

Agglomerative Hierarchical clustering

Let's consider two different clusters, A and B, and a distance d to quantify the similarity among objects.
There are several popular way to define the Inter-Cluster distance:

- **Complete-linkage clustering:**

$$\max_{a \in A, b \in B} d(a, b)$$

- **Minimum or single-linkage clustering:**

$$\min_{a \in A, b \in B} d(a, b)$$

- **Average linkage clustering:**

$$\frac{1}{|A| \cdot |B|} \sum_{a \in A} \sum_{b \in B} d(a, b)$$

W and Moment Hierarchical

W Hierarchical

- We refer to the same setting of the W k-means.

Moment Hierarchical

- As for the Moment k-means each μ_i is mapped to a vector of $\mathbb{R}^p$, whose components are the first p moments. In order to quantify the similarity we take the Euclidean distance.
- In this case we define another criterion for merging clusters, the so called **Ward's method**, a variance-minimizing approach. In particular we merge the clusters that minimize the following quantity

$$\sum_{x \in A \cup B} \|x - \mu_{A \cup B}\|^2 - \sum_{x \in A} \|x - \mu_A\|^2 - \sum_{x \in B} \|x - \mu_B\|^2$$

where μ_C represents the centroid of the cluster C.

Validation of the clustering algorithms

"The validation of clustering structures is the most difficult and frustrating part of cluster analysis.

Without a strong effort in this direction, cluster analysis will remain a black art accessible only to those true believers who have experience and great courage." [2]

Validation on synthetic data

Generating synthetic price path helps to define a validation procedure.

- We know both the underlying probabilistic structure and the regime change periods *a priori*.
- It's possible to evaluate both how accurately either algorithm is classifying sequences of returns into regimes, and how closely the centroids $\{\bar{\mu}_l\}$ of each cluster correspond to the true distributions $\{P_l\}$ associated to the synthetic data.

We assume 2 different regimes: a standard regime (regime-off) and the regime change (regime-on). They can be interpreted from an economic point of view as bull and bear regimes.

Validation on synthetic data

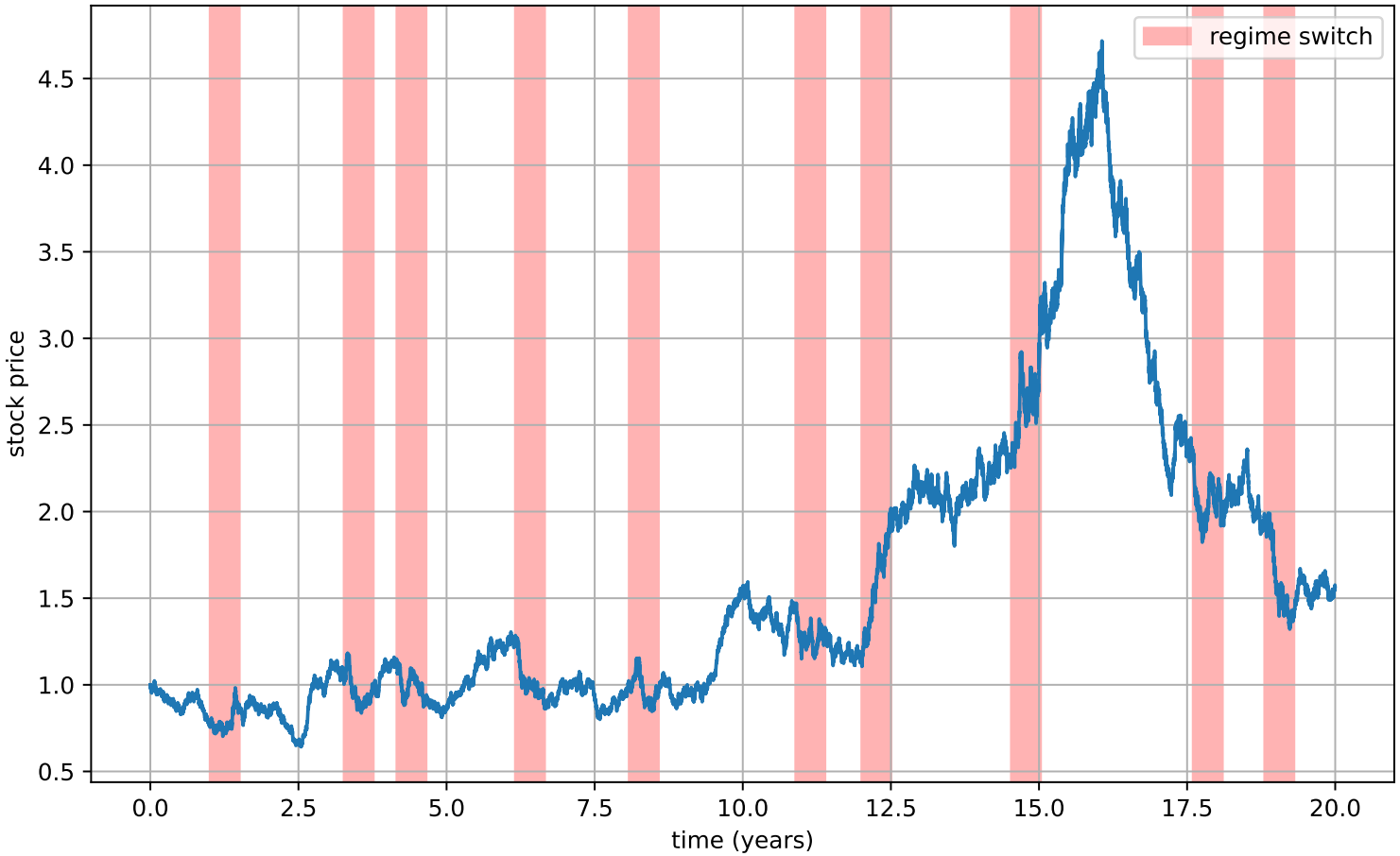
The methodology is as follows:

- For a given time interval $[0, T]$ with $T \in \mathbb{N}$, we define a mesh so that each time increment roughly represents one market hour. That is, with $n := 252 \times 7$, we set

$$\Delta = \left\{ \left[\frac{i-1}{n}, \frac{i}{n} \right] : i = 1, 2, \ldots, nT \right\}.$$

  then $\Delta t = \frac{1}{n}$ represents 1 market hour.
- Next, we define the number of regime changes $r \in \mathbb{N}$ we wish to observe. We specify their starting points and the length of each interval.
- **Note**: we simulate a path over T = 20 years with r = 10 regime changes, randomly chosen with a duration of 1.5 years.

Validation with synthetic data



Validation on synthetic data

- For $i = 0, \ldots, N-1$, associate to each log-return r_i the empirical measures $M_i = \{\mu_{j(i)}, \ldots, \mu_{j(i)+v_i-1}\}$ it was a member of.
- $j(i) \in \mathbb{N}$ is the first measure that r_i is a member of. With $h_1 = 35$ and $h_1 = 28$, one has that $v_i \in [1, 6]$. Note that if the overlap parameter $h_2 = 0$, then $v_i = 1$.
- We then calculate which cluster each $\mu \in M_i$ is associated to, which gives us our predicted labels $\bar{y}^i = \{\bar{k}_1, \ldots, \bar{k}_v\}$. We then aggregate these labels into the row vector

$$\bar{Y}^i = \left(\sum_{j=1}^{v} \chi_{\{x=l\}}(\bar{k}_j) \right)_{l=1}^{k} \quad \text{for } i = 0, \ldots, N-1,$$

where $k = 2$ is the number of clusters. In what follows we assume the assignment $\bar{k} = 1$ corresponds to the standard regime and $\bar{k} = 2$ the regime change.

Accuracy scores

For a given vector of log-returns $\mathbf{r}$ and cluster assignments $C = \{C_l\}_{l=1}^{2}$, we define:

- **regime-off accuracy score (ROFS)**

$$\mathrm{ROFS}(\mathbf{r}, C) = \frac{\sum_{r_i \in off} \bar{Y}_1^i}{\sum_{r_i \in off} \sum_{k=1,2} \bar{Y}_k^i} \in [0, 1]$$

- **regime-on accuracy score (RONS)**

$$\mathrm{RONS}(\mathbf{r}, C) = \frac{\sum_{r_s^i \in on} \bar{Y}_2^i}{\sum_{r_i \in on} \sum_{k=1,2} \bar{Y}_k^i} \in [0, 1]$$

- **total accuracy (TA)**

$$\mathrm{TA}(\mathbf{r}, C) = \frac{\sum_{r_i \in off} \bar{Y}_1^i + \sum_{r_i \in on} \bar{Y}_2^i}{\sum_{i=1}^{N-1} \sum_{k=1,2} \bar{Y}_k^i} \in [0, 1]$$

Models for generating price paths

- Geometric Brownian motion (GBM)
- Merton jump diffusion model (MJD)
- Heston model

Validation on real data

- "Heuristically, we would like individual clusters to contain objects that are similar to each other whilst being distinct from objects in other clusters" [1]
- Evaluating derived k-means clusters is typically done by evaluating the final total cluster variation $\mathrm{TC}(C^*)$.
  - Associate to each C_l its centroid $\bar{x}_l$ for $l = 1, \ldots, k$. Then, for a given C_l, the within-cluster variation is defined as

$$\mathrm{WC}(C_l) = \sum_{x \in C_l} \|x - \bar{x}_l\|_V^2 \quad \text{for } l = 1, \ldots, k.$$

  - Then, the total cluster variation is given by

$$\mathrm{TC}(C) = \sum_{i=1}^{k} \mathrm{WC}(C_i)$$

- Since the total cluster variation depends on V, one cannot use it in evaluation between clusterings on different choices of V.

Maximum mean discrepancy (MMD)

- Let $(\mathcal{X}, d)$ be a metric space and $\mathcal{F}$ be a class of functions $f : \mathcal{X} \to \mathbb{R}$. If $\mu, \nu \in \mathcal{P}(\mathcal{X})$ are Borel measures, the **maximum mean discrepancy (MMD)** between μ and ν is defined as

$$\mathrm{MMD}[\mathcal{F}, \mu, \nu] := \sup_{f \in \mathcal{F}} \left(\mathbb{E}_\mu[f(x)] - \mathbb{E}_\nu[f(y)] \right).$$

- If as $\mathcal{F}$ we take the Gaussian kernel

$$\kappa_G : \mathbb{R}^d \times \mathbb{R}^d \to [0, +\infty), \quad \kappa_G(x, y) = \exp\left(-\frac{\|x - y\|^2_{\mathbb{R}^d}}{2\sigma^2} \right)$$

  the MMD is a metric on $\mathcal{P}(\mathcal{X})$.
- This last property makes MMD the building block of the validation procedure.

Maximum Mean Discrepancy (MMD)

- If we draw samples $\mathbf{x} = (x_1, \ldots, x_n)$ and $\mathbf{y} = (y_1, \ldots, y_m)$ where $x_i \sim \mu$ for $i = 1, \ldots, n$ and $y_j \sim \nu$ for $j = 1, \ldots, m$, a biased empirical estimate of the previous MMD is given by:

$$\mathrm{MMD}_b^2[\kappa_G, \mathbf{x}, \mathbf{y}] = \left[\frac{1}{n^2} \sum_{i,j=1}^{n} k(x_i, x_j) - \frac{2}{mn} \sum_{i,j=1}^{m,n} k(x_i, y_j) + \frac{1}{m^2} \sum_{i,j=1}^{m} k(y_i, y_j) \right].$$

Cluster validation via MMD

Between-cluster evaluation

Repeat the following 3 steps for each clustering algorithm:

- given the two cluster C_1, C_2, draw $n \in \mathbb{N}$ pairwise samples $(\mu_i, \nu_i) \in C_1 \times C_2$ for $i = 1, \dots, n$.
  - We represent each empirical measure $\mu_i, \nu_i \in \mathcal{P}_p(\mathbb{R})$ by its corresponding vector of log-returns $\mathbf{x}_i, \mathbf{y}_i$.
- For each pair we compute $\mathrm{MMD}_b[\kappa_G, \mathbf{x}_i, \mathbf{y}_i]$, where we choose a gaussian kernel with $\sigma = 0.1$.
- Finally, compute the similarity score

$$\mathrm{Sim} = \mathrm{Median}\left(\left(\mathrm{MMD}_b^2[\kappa_G, \mathbf{x}_i, \mathbf{y}_i]\right)_{1 \le i \le n}\right),$$

Then,

- we compare the associated distribution of the MMD among the histograms generated from all the methods by reporting the similarity score.

Cluster validation via MMD

Within-cluster evaluation is performed much in the same way as the between-cluster case. Repeat the following steps for each clustering algorithm:

- for each cluster C_l, $l = 1, 2$, we draw $n \in \mathbb{N}$ pairwise samples $(\mu_i^1, \mu_i^2) \in C_l \times C_l$ and evaluate the biased MMD.
- We report the similarity score associated to the empirical distribution of each within-cluster MMD and plot the resulting histograms.

So, for each algorithm we get 1 between-cluster similarity score and 2 within-cluster similarity scores.

- "We stated that a clustering algorithm was successful if the *self-similarity* and *distinctness* of derived clusters were appropriately traded off against each other." [1]

References

- Blanka Horvath, Zacharia Issa and Aitor Muguruza. Clustering market regimes using the Wasserstein distance. October 2021.
- Pang-Ning Tan, Michael Steinbach and Vipin Kumar. Introduction to Data Mining. Pearson, 2018 (Second edition).