Multiview Clustering in Economic Science

De Angelis A., Del Monte B., Di Marino E.

Computational Intelligence Class
Dep. of Computer Science - University of Salerno

{*anna.deangelis1, venturadelmonte, e.dimarino5*}*@gmail.com*

Prof. Roberto Tagliaferri & Prof. Michele La Rocca

March 3, 2015

Overview

A very simple scenario

- Suppose that a person wants to invest his saving in a company;
- After some time, the same person believes he has done a wrong investment;
- So, he decides to invest in an another company.

Problem Statement

- The problem is to evaluate whether two or more companies have the same behaviour.

Overview

Solution

- The proposed solution consists of combining "single-view" clustering in order to obtain a final "multi-view" clustering, that will let us understand better the stock options market.

Solution

- There are different representations or "views" describing the same object (asset) for a certain domain.
- Taken alone, these views could be incomplete.
- Therefore, our goal is to integrate multiple views in order to discover the underlying structures in our domain.

Solution

- We use a new method for clustering time series based on their structural characteristic;
- This method clusters on global features extracted from time series.

Overview

- A univariate time series is the simples form of temporal data and is a sequence of values collected regularity in time.
- We represent a time series as an ordered set of n real-valued variables $Y_t = Y_1, \ldots, Y_n$.
- There are **nine** statistical feature describing a time series' global characteristics;

Time Series

- The extracted measures become the inputs to the clustering process.
- Furthermore, for each feature, we have measured its grade of presence and we have normalized the metric to $[0, 1]$.

Features

Feature	RAW data	TSA data
Trend		√
Seasonality		√
Serial Correlation	√	√
Non-linearity	√	√
Skewness	√	√
Kurtosis	√	√
Self-similarity	√	
Chaotic	√	
Periodicity (frequency)	√	

Trend and seasonality

- The trend represents a general direction of a phenomenon during a certain period of time.
- The seasonality is a characteristic of a time series in which the data have regular and predictable changes that occur every calendar year.
- To extract trend and seasonality from a time series, we use a STL decomposition.

Periodicity

- Periodicity is very important for evaluating the seasonality and examining the cyclic pattern of a time series.
- It is calculated using an estimate of the spectral density.

Serial correlation

- If a time series is white noise, $Y_t = c + E_t$;
- we try to extract a measure which shows the degree of serial correlation to detect if the series can fit a white noise model.
- The used measure is "Box-Pierce statistic"

Non linear autoregressive structure

- The traditional linear models cannot handle the forecasting well compared to non linear models.
- We use Teraesvirta's neural network test for time series data non linearity characteristic identification and extraction.

Skewness

- It is a measure of symmetry.
- For a univariate time series, $S = \frac{1}{n\sigma^3}\sum_{t=1}^{n}(Y_t - \bar{Y}_t)^3$, where $\bar{Y}_t$ is the mean, σ is the standard deviation and n is the number of data points.

Kurtosis

- It is a measures of weather the data are peaked or flat, relative to a normal distribution.
- For a univariate time series, $K = \frac{1}{n\sigma^4}\sum_{t=1}^{n}(Y_t - \bar{Y}_t)^4$, where $\bar{Y}_t$ is the mean, σ is the standard deviation and n is the number of data points.

Self-similarity

- The definition is related to the Hurst exponent (H) (a measure of long-term memory of time series).

Chaos

- Non-linear dynamical systems often exhibit chaos, which is characterized by sensitive dependence on initial values, or more precisely by a positive Lyapunov Exponent.

Scaling transformations

- We present data to the clustering algorithms rescaled in the $[0, 1]$ range.
- In this way, certain features will not dominate on other ones.

Scaling transformations

- In order to map the raw measure $Q \in [0, \infty]$ range to a rescaled value $q \in [0, 1]$ range we use the transformation $q = \frac{e^{aQ}-1}{b+e^{aQ}}$ where a and b are costants to be chosen.

Scaling transformations

- In order to map the raw measure $Q \in [0, 1]$ range to a rescaled value $q \in [0, 1]$ range we use the transformation $q = \frac{(e^{aQ}-1)(b+e^{a})}{(b+e^{aQ})(e^{a}-1)}$ where a and b are costants to be chosen.

Overview

Dataset

- The S&P 500 (or the Standard & Poor's 500), is an American stock market index based on the market capitalizations of 500 (502) large companies;
- it is one of the best representations of the U.S. stock market, and a bellwether for the U.S. economy.

STANDARD
&POOR'S

Example of assets in the S&P 500 index

Sectors

Sector	Subsectors
Industrials	18
Health Care	8
Financials	13
Information Technology	12
Utilities	3
Materials	12
Consumer Staples	12
Consumer Discretionary	27
Energy	6
Telecommunications Services	2

Available data

- For each asset, the following data are available:
  - data;
  - opening price;
  - max price;
  - min price;
  - closing price;
  - volume;
  - adjusted closing price.

Available data

VALORI						
Data	Apertura	Massimo	Minimo	Chiusura	Volume	Chiusura aggiustata*
24 feb 2015	132,94	133,60	131,17	132,17	68.977.900	132,17
23 feb 2015	130,02	133,00	129,66	133,00	70.974.100	133,00
20 feb 2015	128,62	129,50	128,05	129,50	48.948.400	129,50
19 feb 2015	128,48	129,03	128,33	128,45	37.362.400	128,45
18 feb 2015	127,63	128,78	127,45	128,72	44.891.700	128,72
17 feb 2015	127,49	128,88	126,92	127,83	63.152.400	127,83
13 feb 2015	127,28	127,28	125,65	127,08	54.272.200	127,08
12 feb 2015	126,06	127,48	125,57	126,46	74.474.500	126,46
11 feb 2015	122,77	124,92	122,50	124,88	73.561.800	124,88
10 feb 2015	120,17	122,15	120,16	122,02	62.008.500	122,02

Computed data

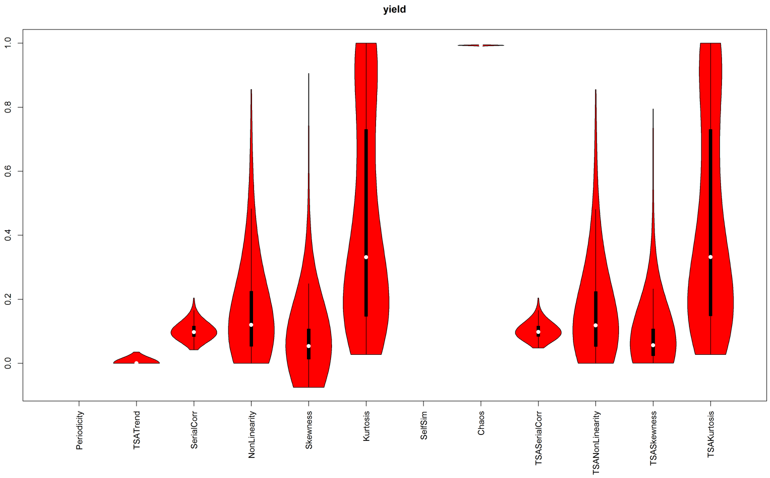
Starting from the previous data, we have computed (for the whole target period):

- yield;
  - computed as the difference of the natural logarithms of the closing price.
- range;
  - computed as the difference between the max price and the min price.
- volatility;
  - computed as the absolute value of the yield.
- volume;
- ~~adjusted closing price~~.

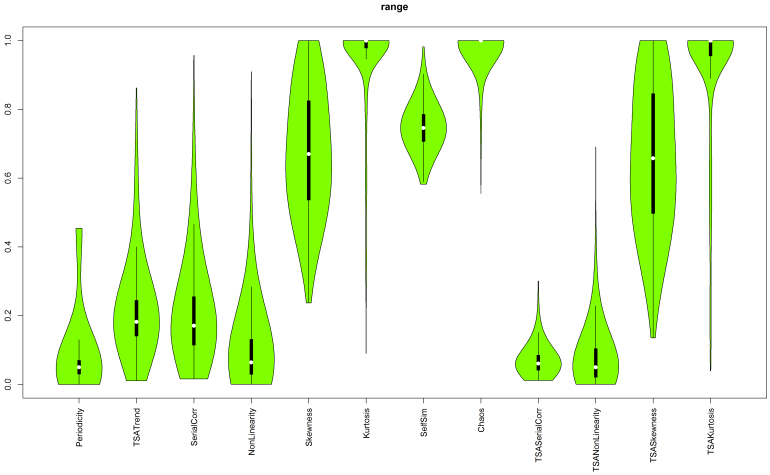
Feature searching

- A different selection from our limited number of global measures could affect the clustering result.
- The problem is to find the best set of n features from our set of m features.
- $m = 13$. There are $2^{13} - 1 = 8191$ possible subset of 13 measures.

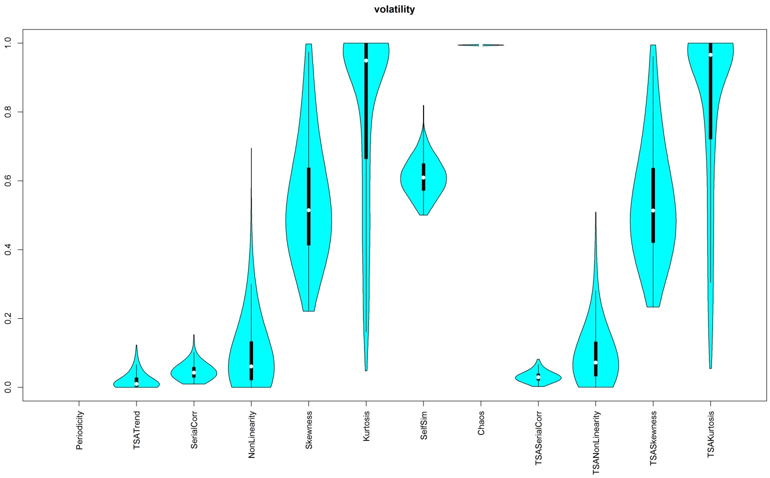
Violin plot S&P500 - Yield



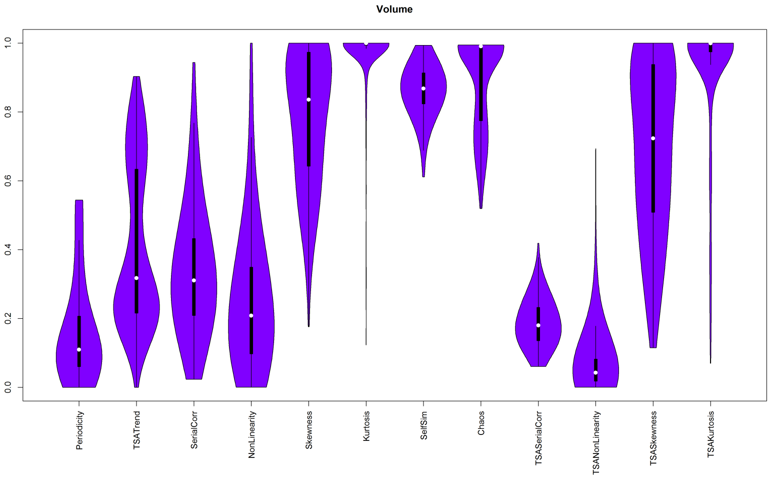
Violin plot S&P500 - Range



Violin plot S&P500 - Volatility

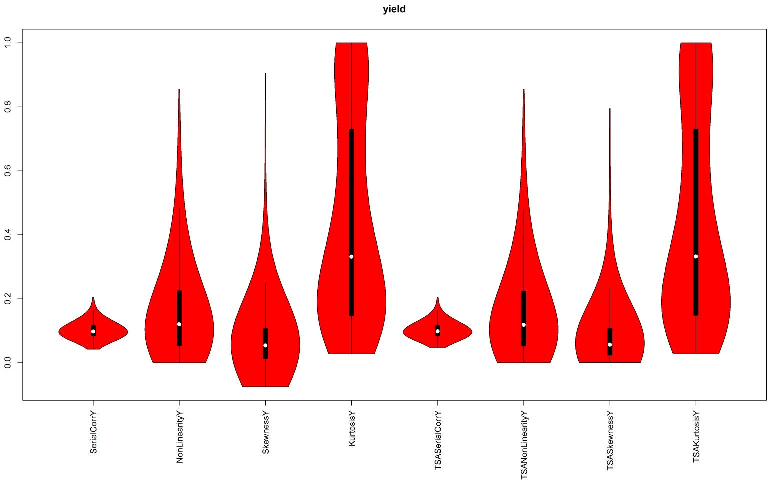


Violin plot S&P500 - Volume

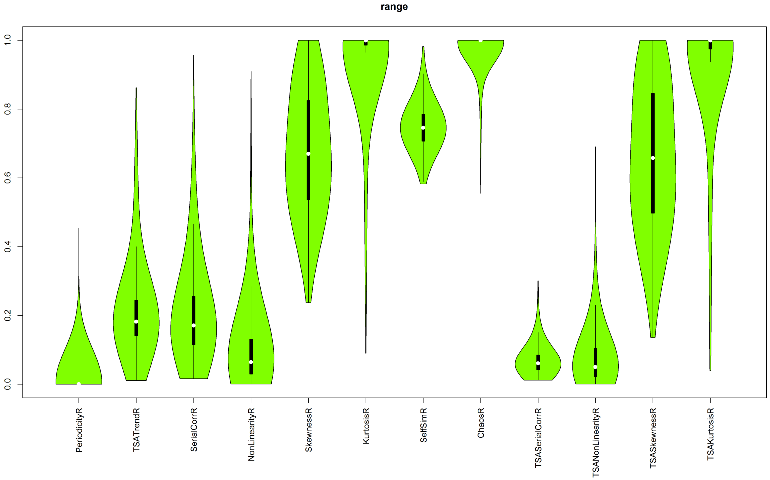


Overview

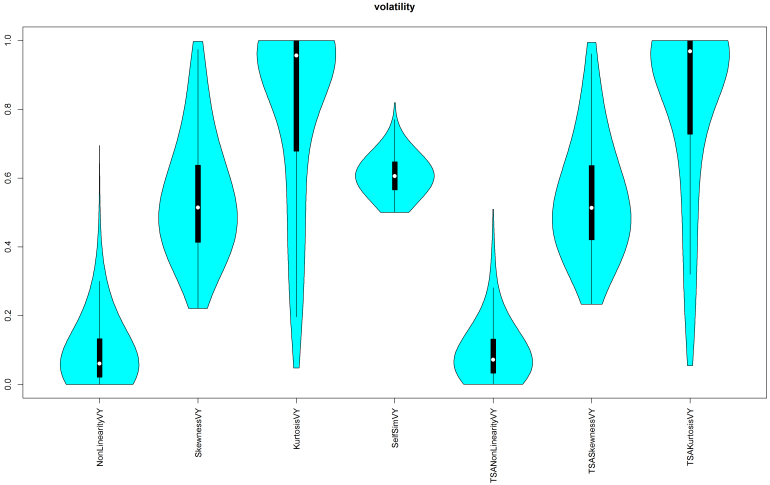
Features selection for the yield



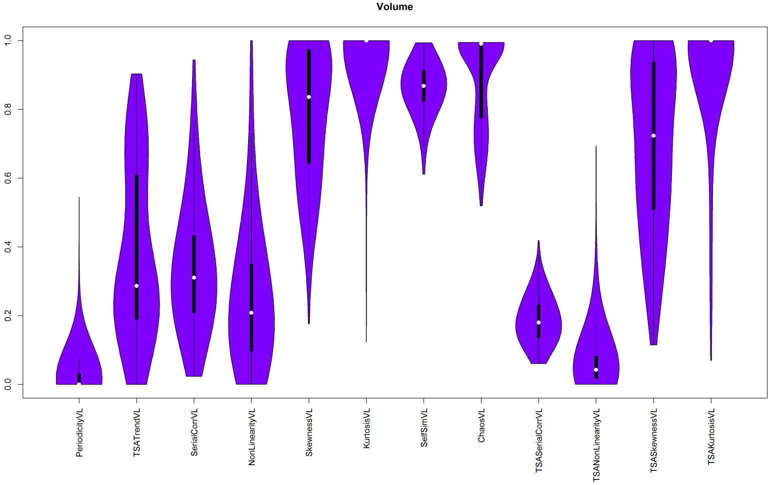
Features selection for the range



Features selection for the volatility



Features selection for the volume



What is clustering?

- A way of grouping together data samples that are similar in some way - according to some choosen criteria.
- A form of **unsupervised learning** - you generally don't have examples demonstrating how the data should be grouped together.
- So, it's a method of **data exploration** - a way of looking patterns or structure in the data that are of interest.

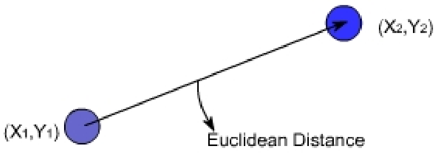
How to define "similarity"?

- Recall the goal is to group together "similiar" data.
- No single answer. It depends on what we want to find or emphasize in the data (this is one reason why clustering is an "art").
- The similarity measure is often more important than the clustering algorithm used.

(Dis)similarity measures

- We often equivalently refer to dissimilarity measures.
- Jagota defines a dissimilarity measure as a function $f(x, y)$ such that $f(x, y) > f(w, z)$ i.f.f. x is less similar to y than w is to z.

Euclidean distance



$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

- n is the number of dimensions in the data vector.

Pearson Correlation

- Pearson correlation measures the degree of the correlation (dependence) between two variables.
- It's always between -1 and 1 (perfectly anti-correlated and perfectly correlated).
- This is a similarity measure, but we can easily make it into a dissimilarity measure:

$$d_p = \frac{1 - r(x,y)}{2}$$

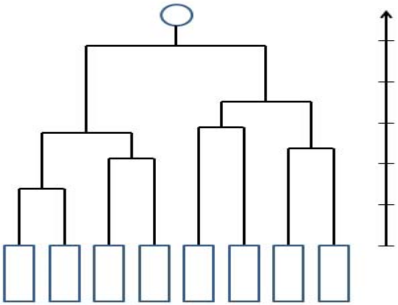
- $r(x, y)$ is the Pearson correlation.

Overview

Hierarchical Agglomerative Clustering

- We start with every data point in a separate cluster.
- We keep merging the most similar pairs of data points/clusters until we have one big cluster left.
- This is called a bottom-up or agglomerative method.

Hierarchical Clustering (cont.)

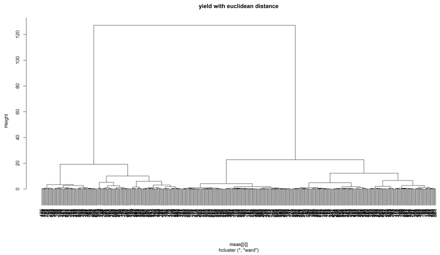


- This produces a binary tree or dendogram.
- The final cluster is the root and each data item is a leaf.
- The height of the bars indicate how close the items are.

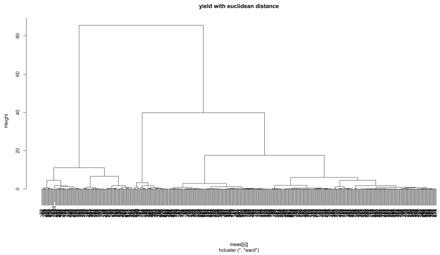
Ward's method

- Ward's method criterion minimizes the total within-cluster variance.
- At the initial step, all clusters are singletons (clusters containing a single point).
- At each step the pair of clusters with minimum cluster variance is merged.

Hierarchical clustering: Yield

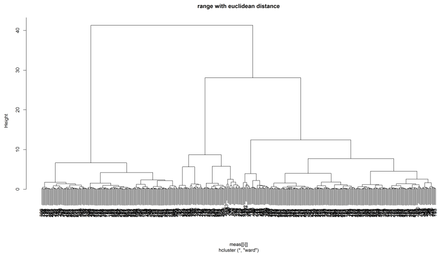


(a) Euclidean

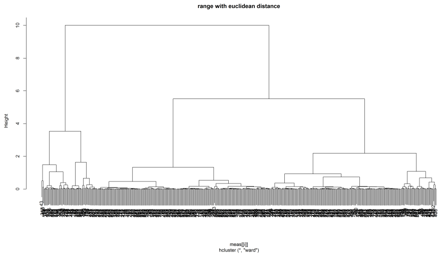


(b) Pearson

Hierarchical clustering: Range

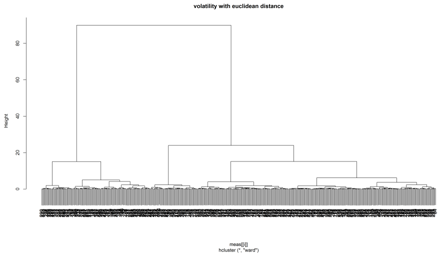


(a) Euclidean

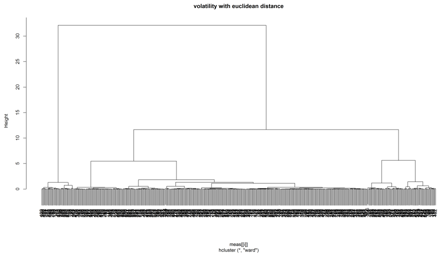


(b) Pearson

Hierarchical clustering: Volatility

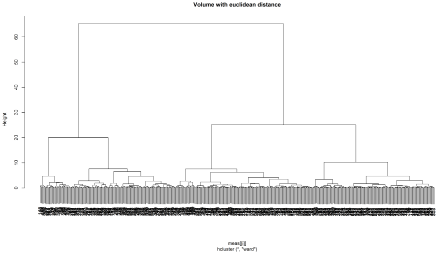


(a) Euclidean

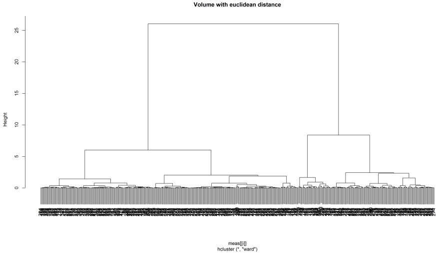


(b) Pearson

Hierarchical clustering: Volume



(a) Euclidean



(b) Pearson

Overview

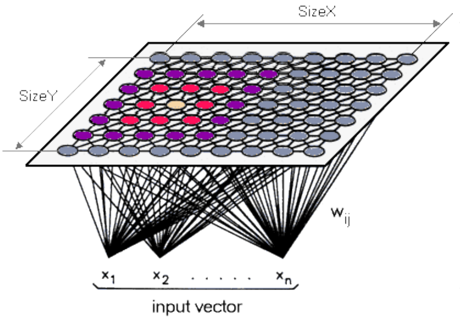
Non Hierarchical clustering

- These clustering methods do not possess tree-like structures.
- New clusters are formed in successive clustering either by merging or splitting clusters.

SOM

- SOM are based on the work of Kohonen on learning/memory in the human brain.
- We specify the number of clusters and also a topology i.e. a $2D$ grid that gives the geometric relationships between the clusters.
- The algorithm learns a mapping from the high dimensional space of the data points onto the points of the $2D$ grid, there is one grid point for each cluster.

Self-Organizing Maps (cont.)

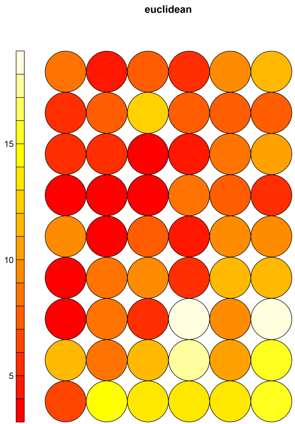


- The SOM network typically has two layers of nodes: the **input layer** and the **Kohonen layer**.
- The input layer is fully connected to a two-dimensional Kohonen layer.

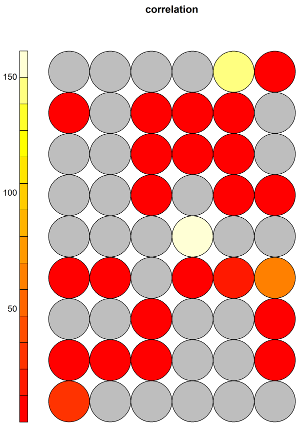
Self-Organizing Maps (cont.)

- Suppose we have a $r \times s$ grid with each grid point associated with a cluster mean.
- SOM algorithm moves the cluster means around in the high dimensional space, maintaining the topology specified by the $2D$ grid.
- A data point is put into the cluster with the closest mean.
- The effect is that nearby data points tend to map to nearby clusters (grid points).

Non Hierarchical clustering: yield

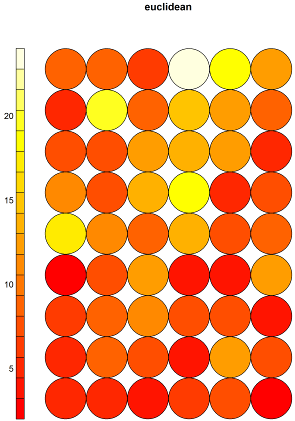


(a) Euclidean

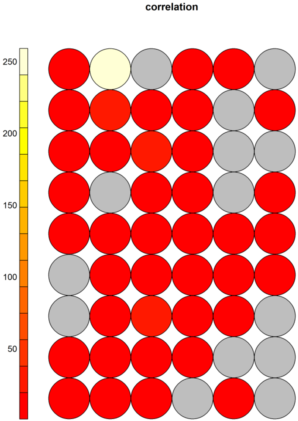


(b) Pearson

Non Hierarchical clustering: range

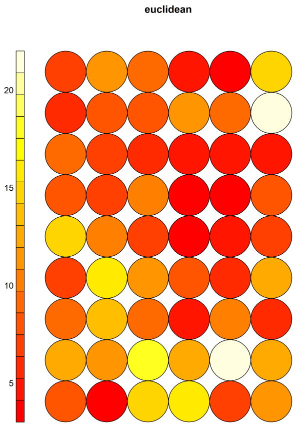


(a) Euclidean

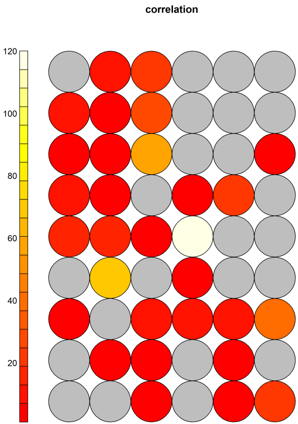


(b) Pearson

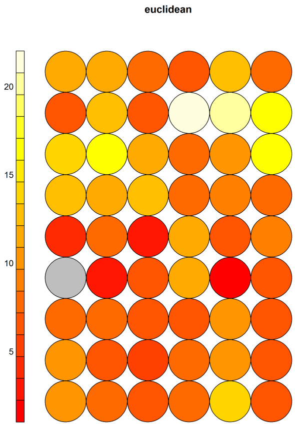
Non Hierarchical clustering: volatility



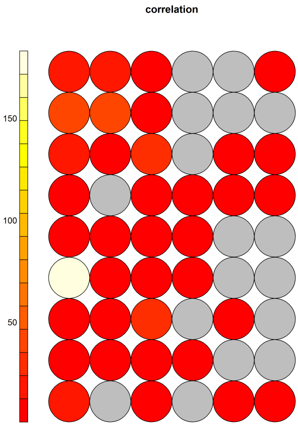
(a) Euclidean



(b) Pearson



(a) Euclidean



(b) Pearson

Overview

Determining the number of clusters

We have used two different approaches to determine the number of clusters in our dataset:

- Silhouette index,
- Correlation inter/intra clusters.

Silhouette index

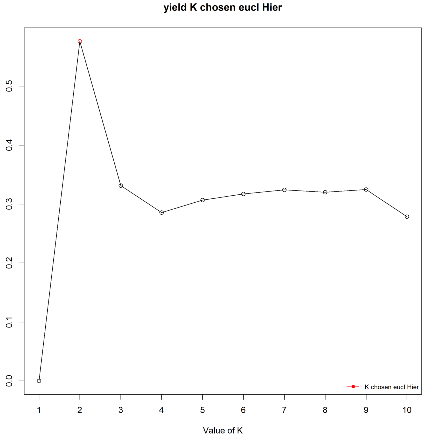
- It combines measures of cohesion and separation.
- Let $\mathbb{C}$ be a cluster, $\forall\ i \in \mathbb{C}$, we defined the silhouette width of the point i:

$$s_i = \frac{b_i - a_i}{max(a_i, b_i)}$$

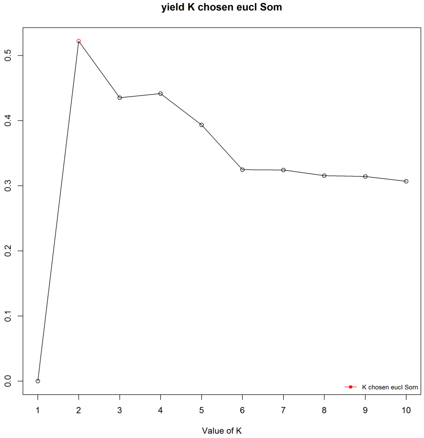
- a_i: the average distance between i and all the other points in the same class;
- b_i: the average distance between i and all other points in the nearest cluster.

- It is a quantity between -1 and 1.

Silhouette indices - Yield

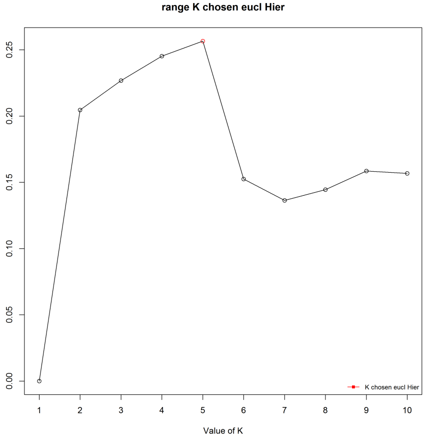


(a) Euclidean Hier

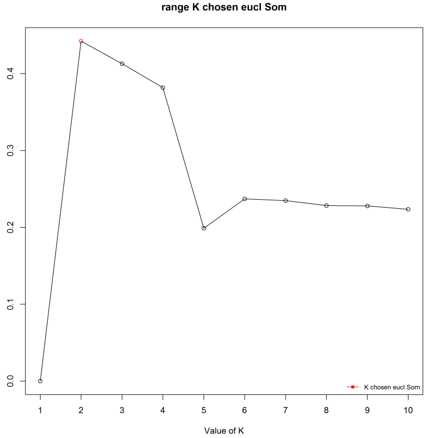


(b) Euclidean SOM

Silhouette indices- Range

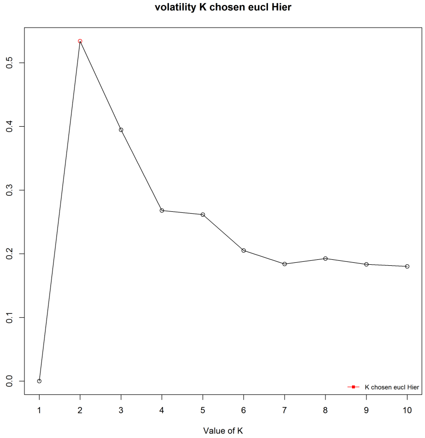


(a) Euclidean Hier

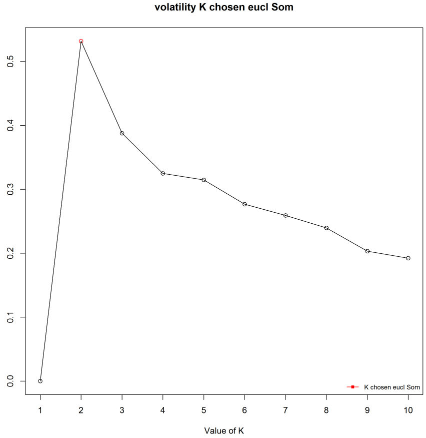


(b) Euclidean SOM

Silhouette indices - Volatility

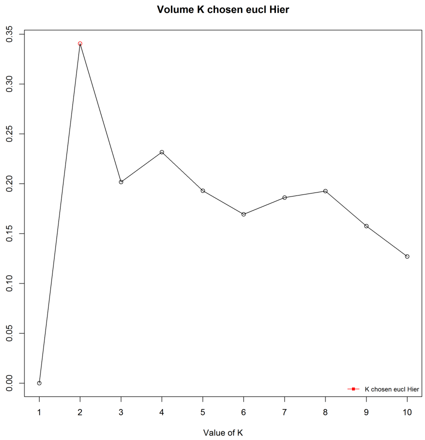


(a) Euclidean Hier

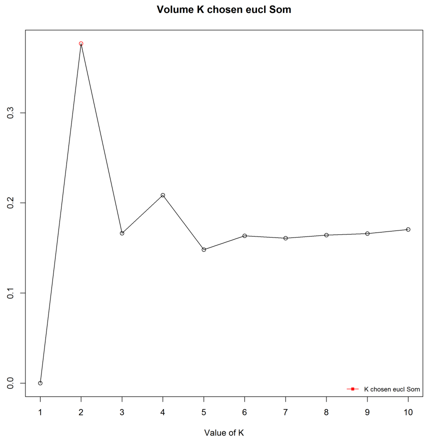


(b) Euclidean SOM

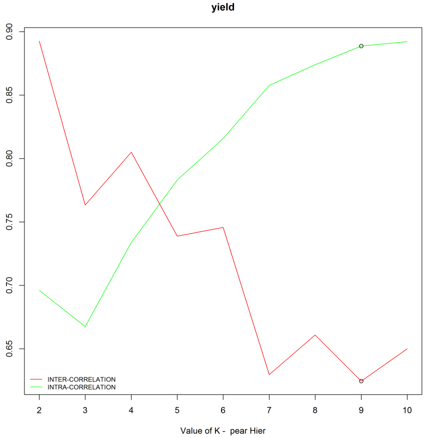
Silhouette indices- Volume



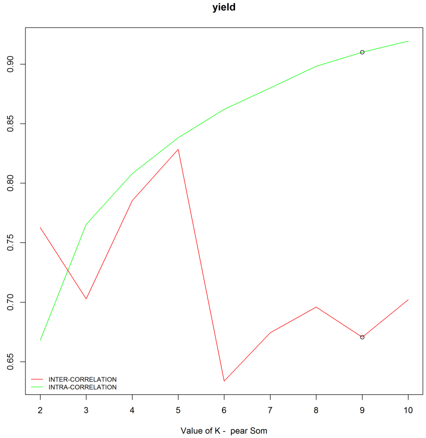
(a) Euclidean Hier



(b) Euclidean SOM

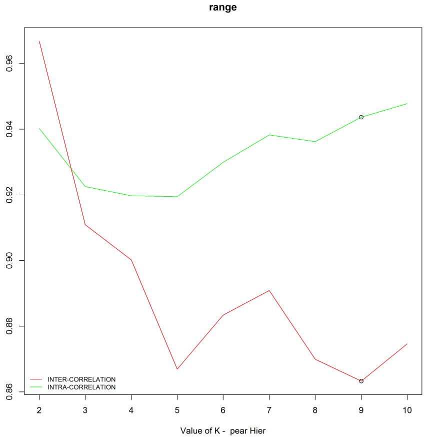

yield
0.90
0.85
0.80
0.75
0.70
0.65
INTER-CORRELATION
INTRA-CORRELATION
2
3
4
5
6
7
8
9
10
Value of K - pear Hier

(a) Hier

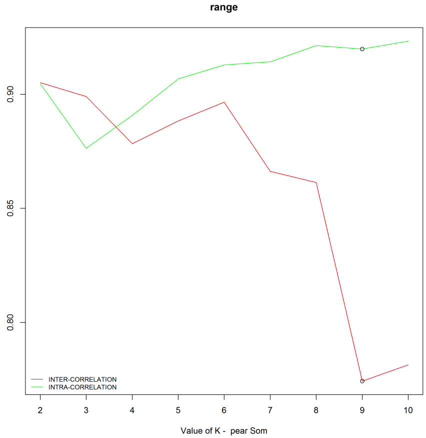

yield
0.90
0.85
0.80
0.75
0.70
0.65
INTER-CORRELATION
INTRA-CORRELATION
2
3
4
5
6
7
8
9
10
Value of K - pear Som

(b) SOM

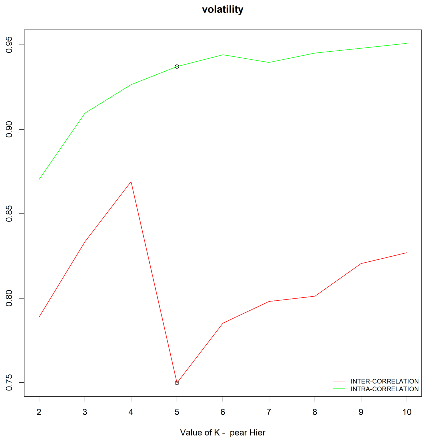
Correlation inter/intra - Range



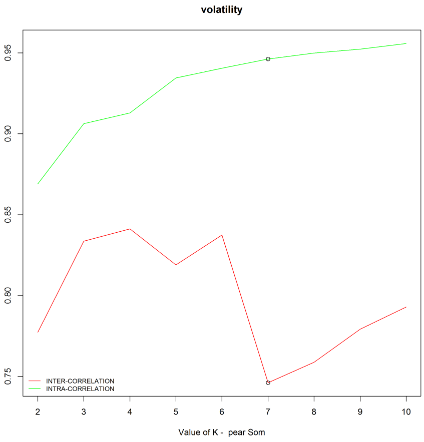
(a) Hier



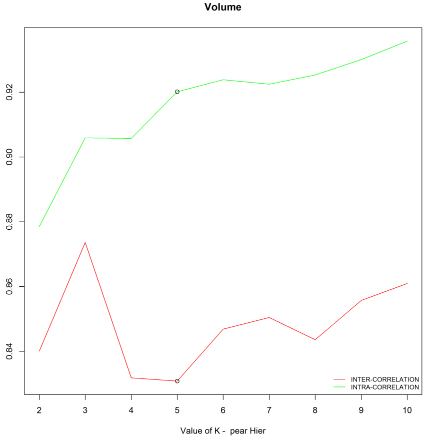
(b) SOM


volatility
0.75
0.80
0.85
0.90
0.95
2
3
4
5
6
7
8
9
10
Value of K - pear Hier
INTER-CORRELATION
INTRA-CORRELATION

(a) Hier

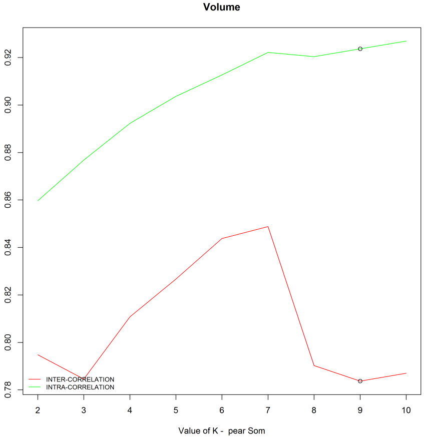

volatility
0.75
0.80
0.85
0.90
0.95
2
3
4
5
6
7
8
9
10
Value of K - pear Som
INTER-CORRELATION
INTRA-CORRELATION

(b) SOM



Volume
0.84
0.86
0.88
0.90
0.92
2
3
4
5
6
7
8
9
10
Value of K - pear Hier
INTER-CORRELATION
INTRA-CORRELATION

(a) Hier



Volume
0.78
0.80
0.82
0.84
0.86
0.88
0.90
0.92
2
3
4
5
6
7
8
9
10
Value of K - pear Som
INTER-CORRELATION
INTRA-CORRELATION

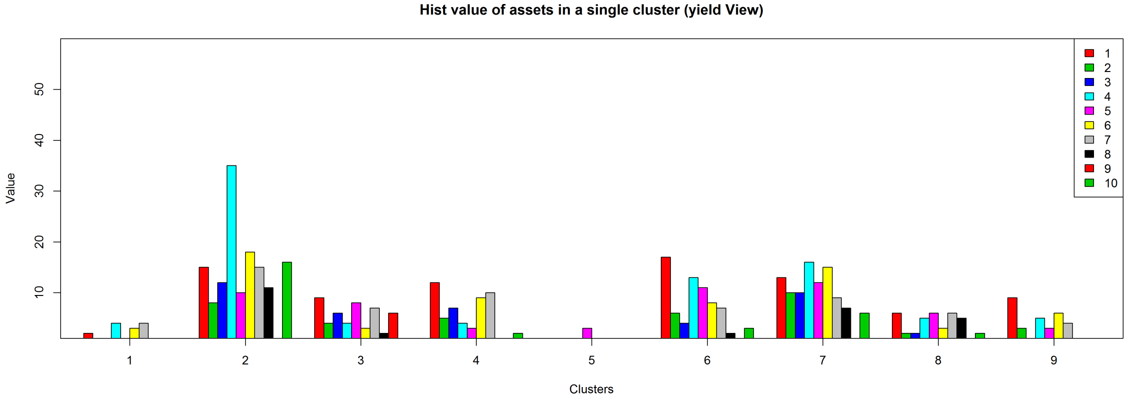
(b) SOM

Overview

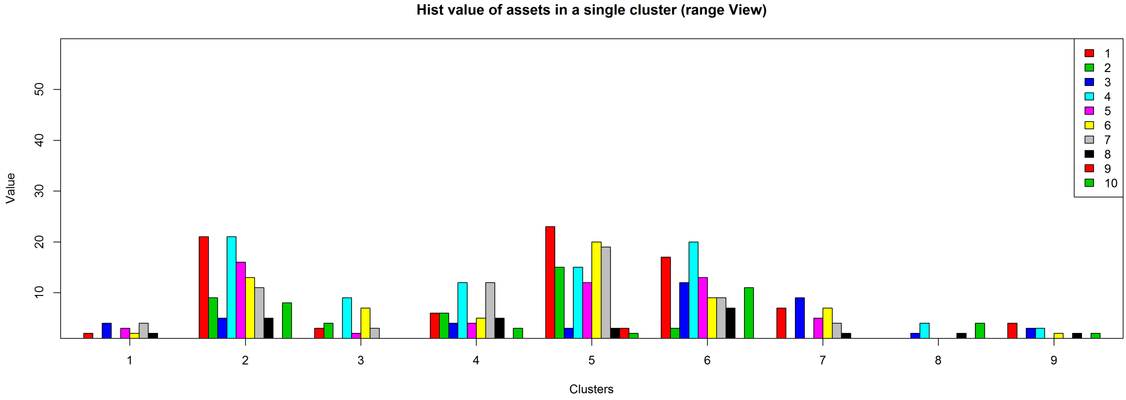
Results

	Yield	Range	Volatility	Volume
Hier Eucl	2	5	2	2
Hier Corr	**9**	**9**	5	5
Som Eucl	2	2	2	2
Som Corr	9	9	**7**	**9**

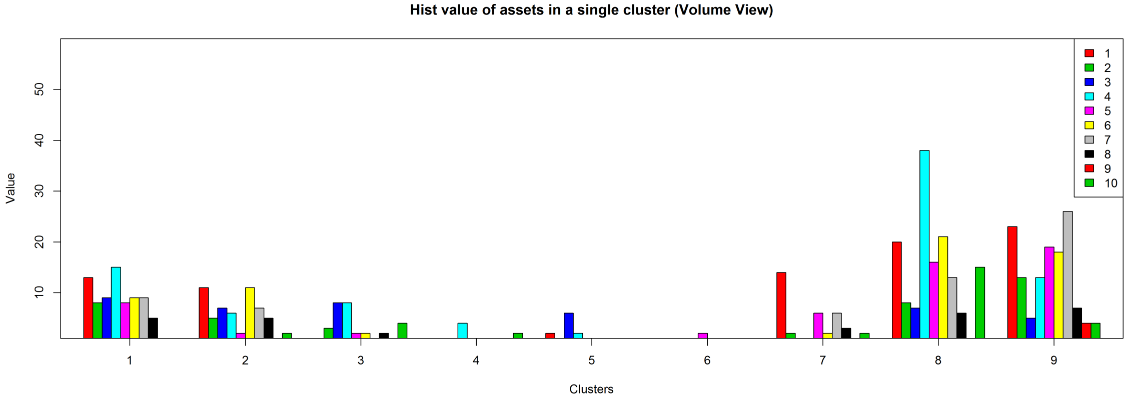
The resulting plot - Yield



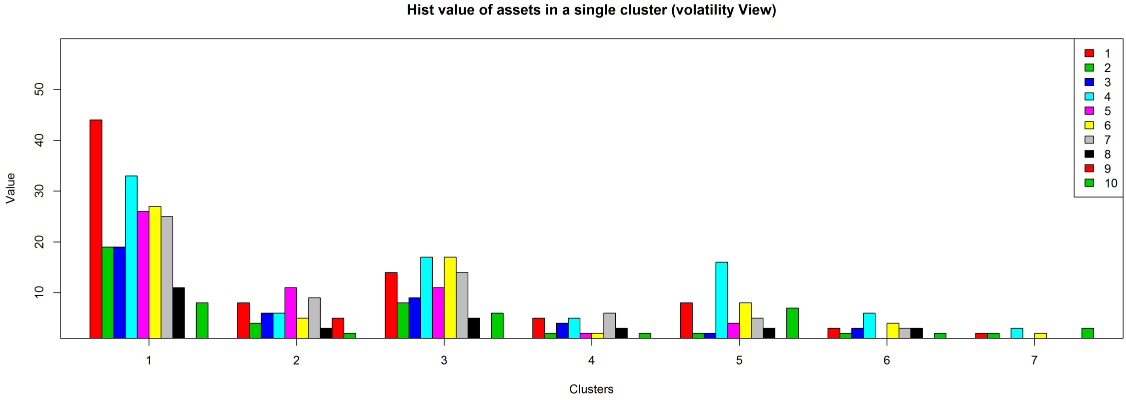
The resulting plot - Range



The resulting plot - Volume



The resulting plot - Volatility



Clusters' impurity

- Typical objective functions in clustering formalize the goal of attaining high intra-cluster similarity and low inter-cluster similarity.
- To compute **purity** , each cluster is assigned to the class which is most frequent in the cluster, and then the accuracy of this assignment is measured by counting the number of correctly assigned documents and dividing by N

$$\textbf{Impurity} = 1 - \text{purity}$$

	Impurity
Yield	0.2215569
Range	0.2115768
Volatility	0.2055888
Volume	0.2475050

Overview

Why Multi-view Clustering?

- There are different representations/views of the same set of objects for a given domain.
- Taken alone, these views are often incomplete, therefore we need to integrate these multiple views in order to discover the underlying structures in the domain.

What's multiview clustering?

- It's an unsupervised algorithm, that allows to combine informations from different views using a *late integration* strategy.
- Each view is represented throughout a clustering.
- Through IMF, single-view clusters are projected into *meta-clusters*, yielding to a complete representation of the given domain.

The Multiview algorithm - Setup

- Let $\mathbb{C} = \{C_1, \ldots, C_v\}$, where C_h represents the clustering $\{c_h^1, \ldots, c_h^{k_h}\}$ for the h-th view. Each view represents the same set of n objects.
- Let $\mathbb{M} = \{M_1, \ldots, M_v\}$, where $M_h \in \mathbb{R}^{n \times k_h}$ represents the cluster membership of objects in C_h generated on the h-th view.
- We build $\mathbf{X} \in \mathbb{R}^{l \times n}$ by transposing the matrices in $\mathbb{M}$ and stacking them vertically, where $l = \sum_{i=1}^{v} k_i$.

The Multiview algorithm - Factorization process

- **Goal**: To project the clusters in $\mathbb{C}$ to a set of $k^{'} < l$ meta-clusters.
- The process involves producing an approximation of $\mathbf{X}$ in the form of the product of two non-negative factors: $\mathbf{X} \approx \mathbf{PH}$ such that $\mathbf{P} \geq 0, \mathbf{H} \geq 0$.
- $\mathbf{P}$ and $\mathbf{H}$ have been initialized through Non Negative Double Singular Value Decomposition.
- The rows of $\mathbf{P} \in \mathbb{R}^{l \times k^{'}}$ represent the projection of the original clusters to $k^{'}$ meta-clusters.
- The columns in $\mathbf{H} \in \mathbb{R}^{k^{'} \times n}$ can be viewed as the membership of the n objects with respect to the $k^{'}$ meta-clusters.

The Multiview algorithm - Factorization process

- To measure the reconstruction error between the original matrix **X** and **PH** we can compute the Frobenius norm:

$$\|X - PH\|_F^2 = \sum_{i=1}^{l} \sum_{j=1}^{n} [X_{ij} - (PH)_{ij}]^2$$

- To minimize the value of the norm, we iteratively apply the following multiplicative rules:

$$P_{ic} \leftarrow P_{ic} \frac{(XH^T)_{ic}}{(PHH^T)_{ic}} \text{ e } H_{cj} \leftarrow H_{cj} \frac{(P^T X)_{cj}}{(P^T PH)_{cj}}$$

- The rules are applied until the change in the norm between one iteration and the next is below an arbitrarily small value.

The Multiview algorithm - The contribution matrix

- Based on $\mathbf{P}$ we can calculate $\mathbf{T} \in \mathbb{R}^{v \times k'}$ indicating the contribution of the h-th view to each meta-cluster:

$$T_{hf} = \frac{\sum_{c_f^j \in C_f} P_{jf}}{\sum_{g=1}^{l} P_{gf}}$$

- A value of T_{hf} close to 0 indicates that the h-th view has made little contribution to the f-th meta-cluster, meanwhile a value close to 1 indicates that the h-th view has made predominant contribution to the f-th meta-cluster.

The Multiview algorithm - An Example

$$\mathcal{C}_1 = \{\{x_1, x_2, x_3\}, \{x_4, x_5\}\}$$

$$\mathcal{C}_2 = \{\{x_6, x_7\}, \{x_1, x_2\}\}$$

$$\longrightarrow \quad \mathbf{X} \quad \begin{array}{c|ccccccc} & x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 \\ \hline c_1^1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ c_1^2 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ \hdashline c_2^1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ c_2^2 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{array}$$

$$\mathbf{X} \approx \mathbf{PH} \longrightarrow \quad \mathbf{P} \quad \begin{array}{c|ccc} c_1^1 & 1.2 & 0.0 & 0.0 \\ c_1^2 & 0.0 & 1.2 & 0.0 \\ \hdashline c_2^1 & 0.0 & 0.0 & 1.2 \\ c_2^2 & 0.9 & 0.0 & 0.0 \end{array} \qquad \mathbf{H}^\mathsf{T} \quad \begin{array}{c|ccc} x_1 & 1.0 & 0.0 & 0.0 \\ x_2 & 1.0 & 0.0 & 0.0 \\ x_3 & 0.5 & 0.0 & 0.0 \\ x_4 & 0.0 & 0.8 & 0.0 \\ x_5 & 0.0 & 0.8 & 0.0 \\ x_6 & 0.0 & 0.0 & 0.8 \\ x_7 & 0.0 & 0.0 & 0.8 \end{array}$$

$$\mathbf{T} \quad \begin{array}{c|ccc} V_1 & 0.6 & 1.0 & 0.0 \\ \hdashline V_2 & 0.4 & 0.0 & 1.0 \end{array}$$

Figure: Example of IMF applied to clustering from two views in a domain of 7 data objects, in order to obtain k' meta-clusters.

Overview

First meta-cluster

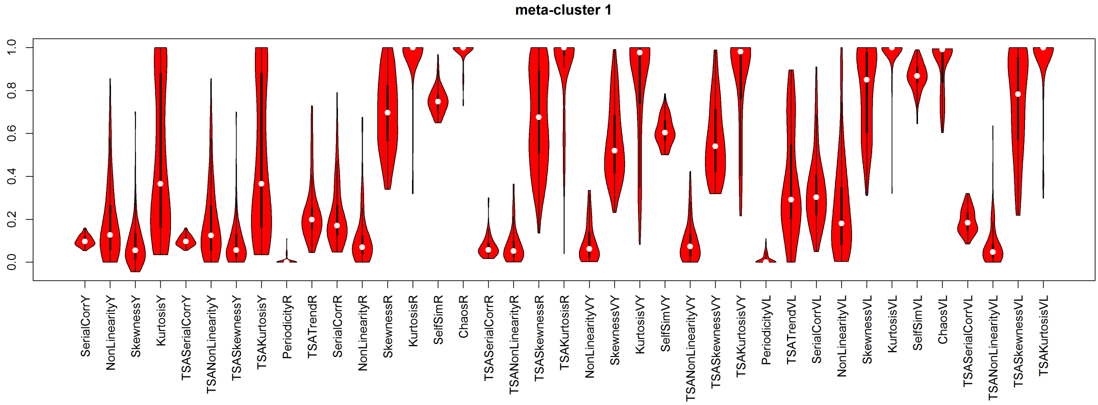


Figure: First meta-cluster

Second meta-cluster

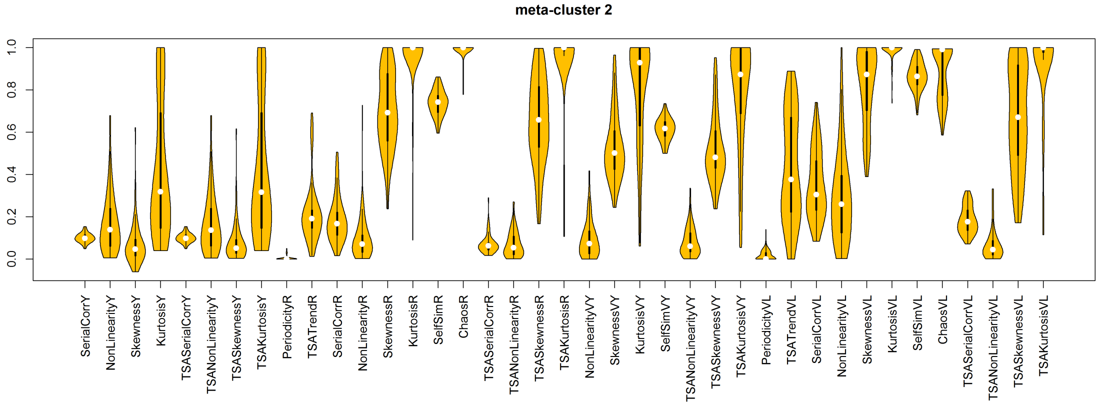


Figure: Second meta-cluster

Third meta-cluster

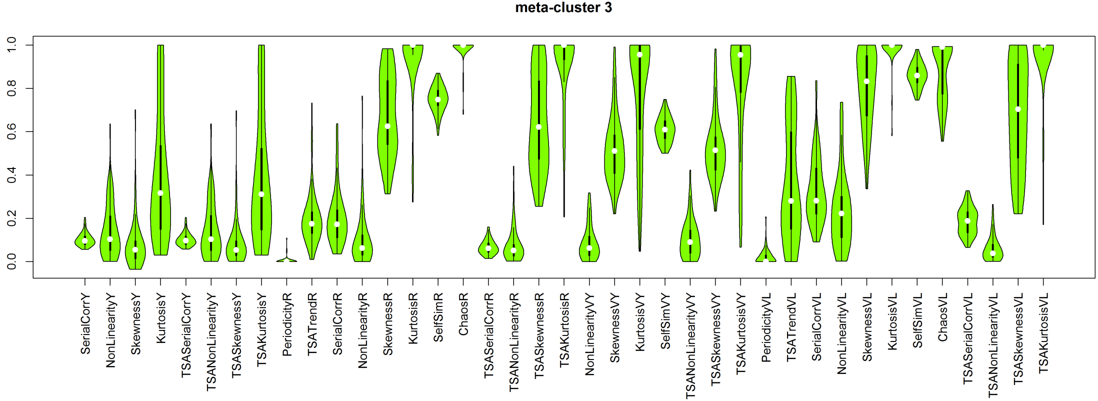


Figure: Third meta-cluster

Fourth meta-cluster

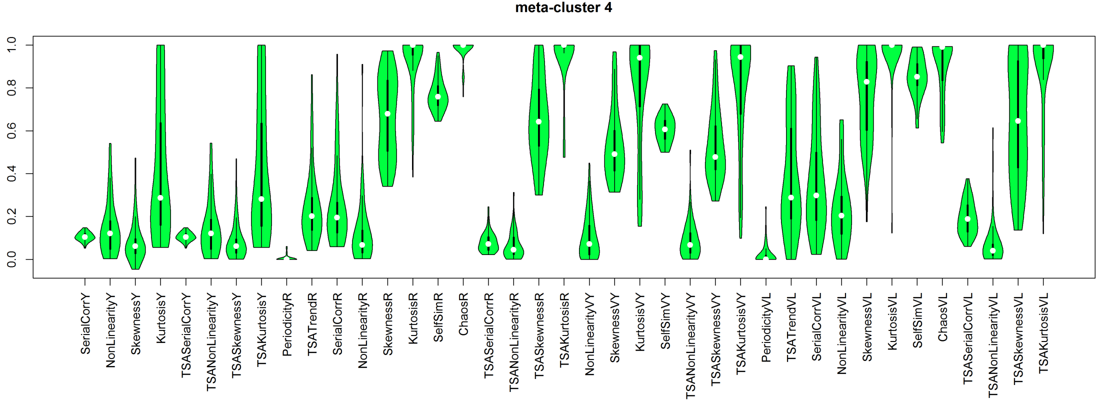


Figure: Fourth meta-cluster

Fifth meta-cluster

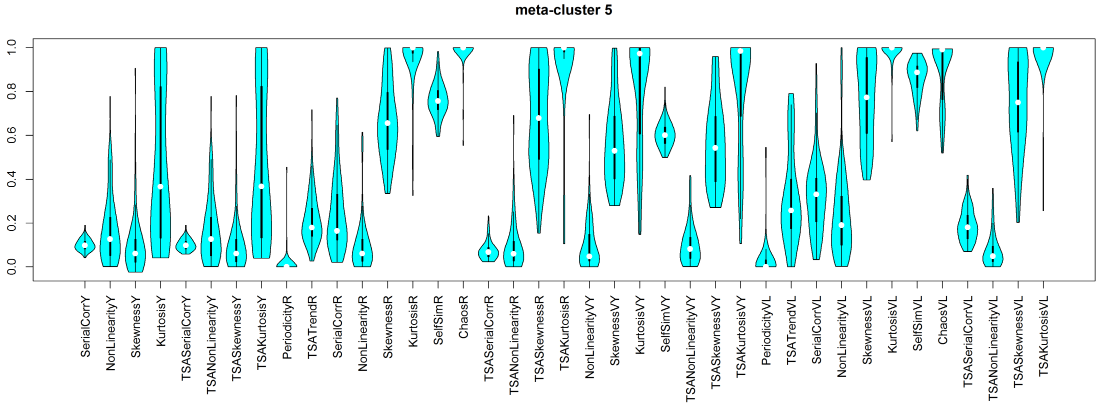


Figure: Fifth meta-cluster

Sixth meta-cluster

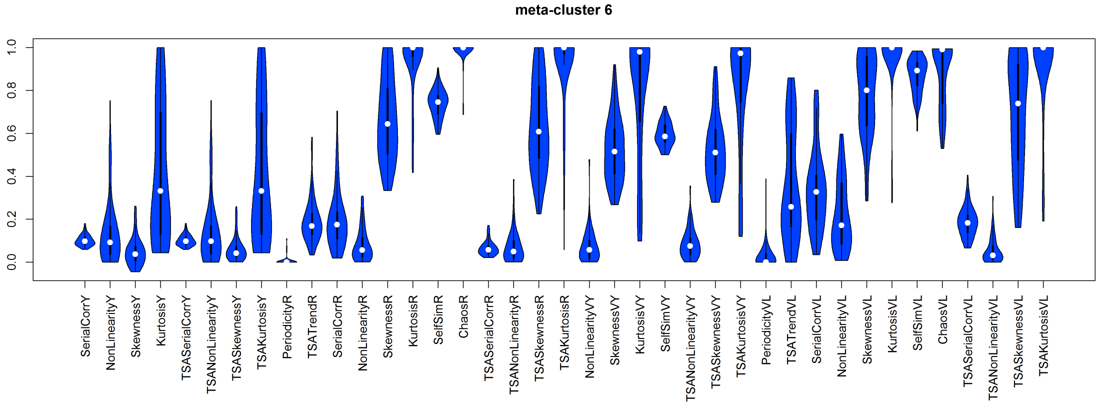


Figure: Sixth meta-cluster

Seventh meta-cluster

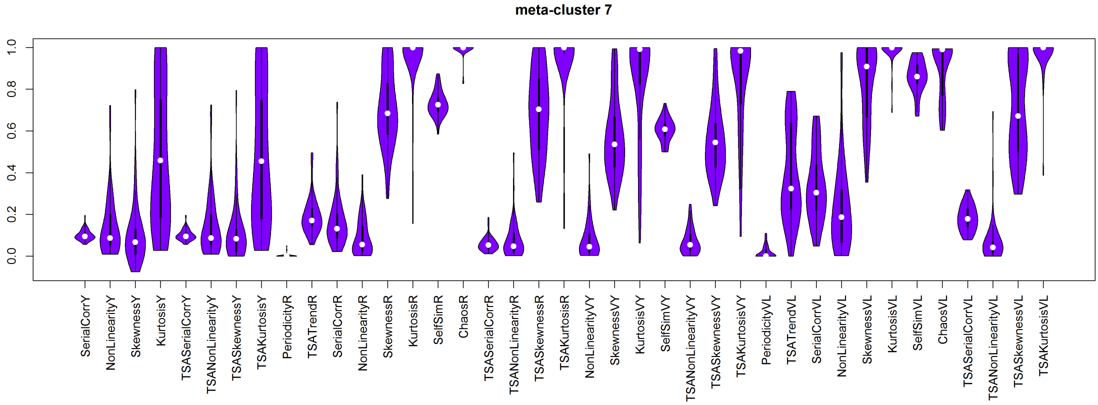


Figure: Seventh meta-cluster

Eighth meta-cluster

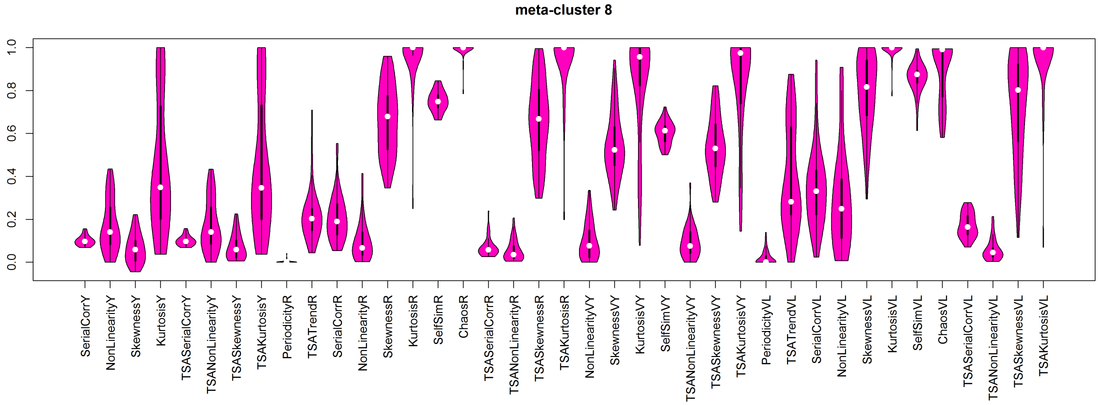
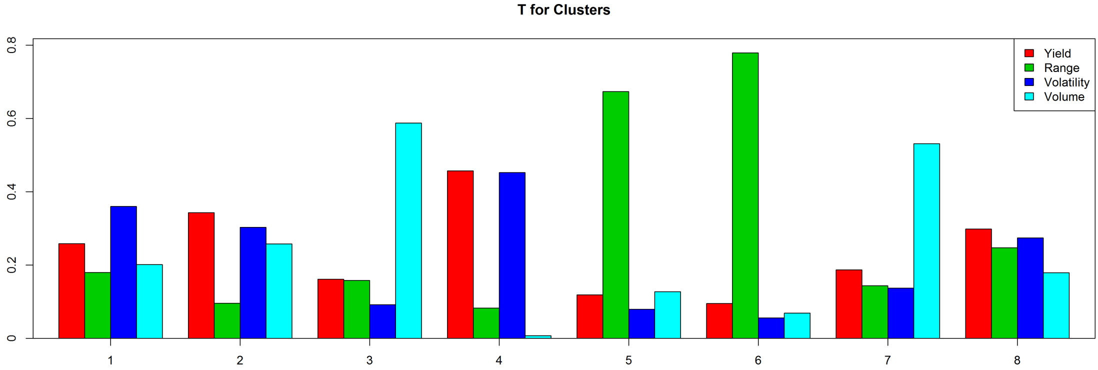
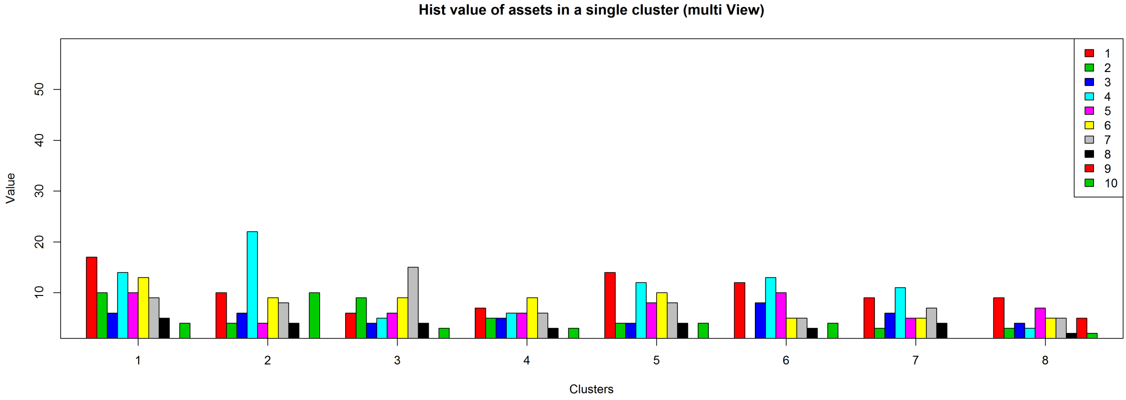


Figure: Eighth meta-cluster

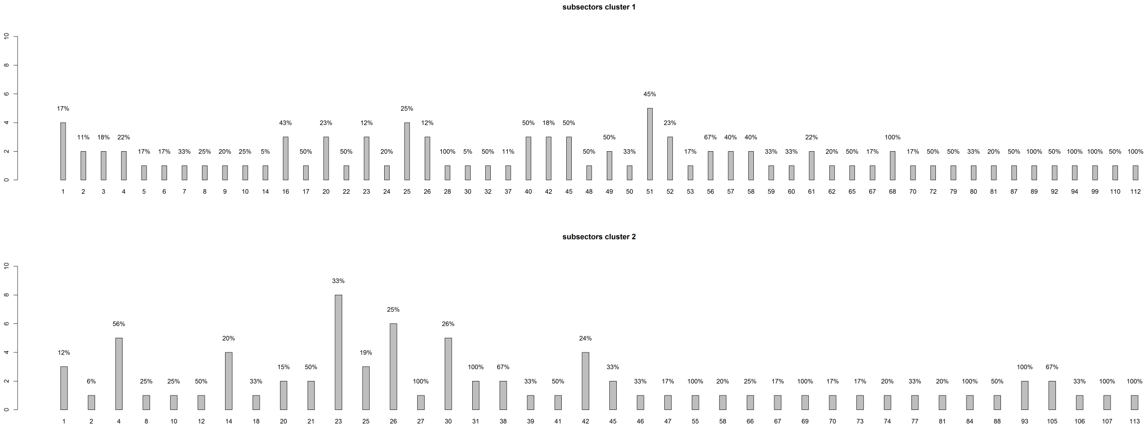
Contribution matrix



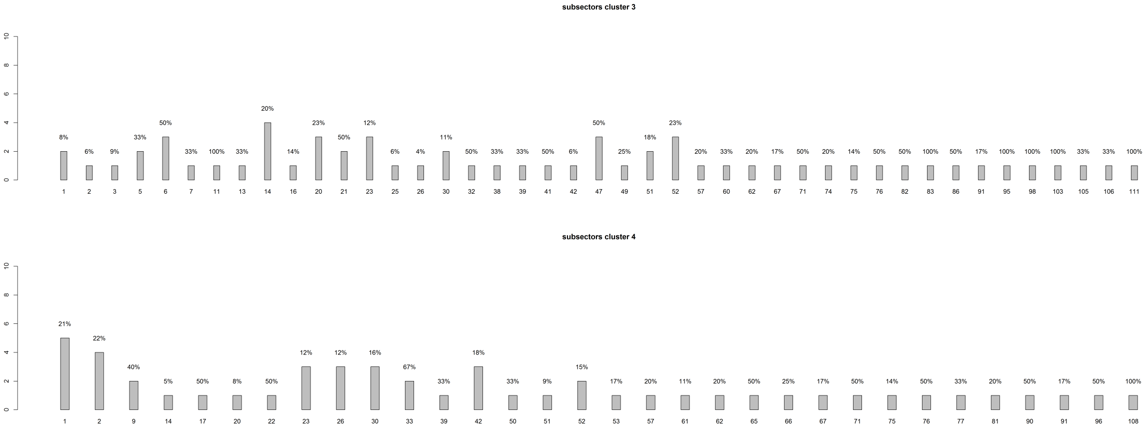
Sectors distribution



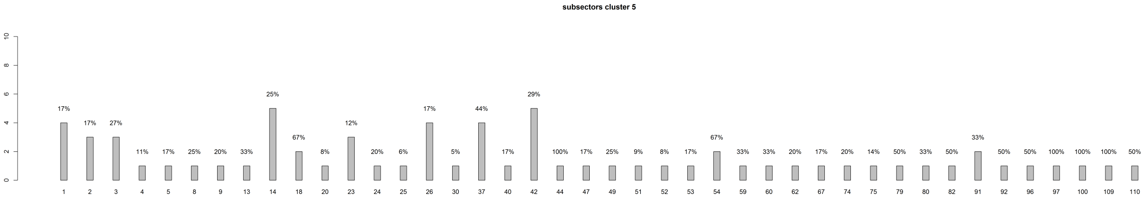
Subsectors distribution

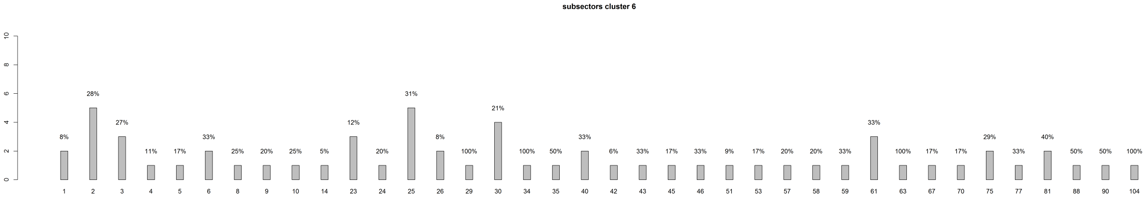


Subsectors distribution (cont.)

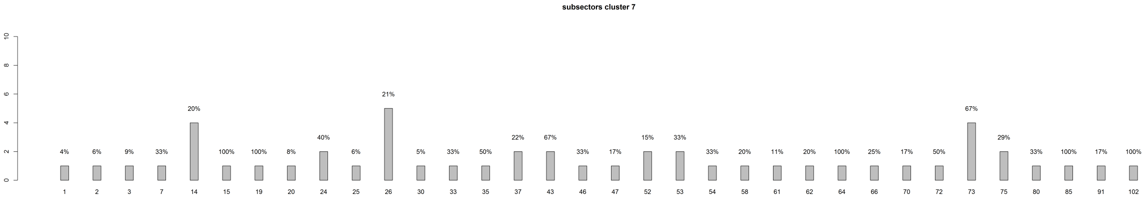


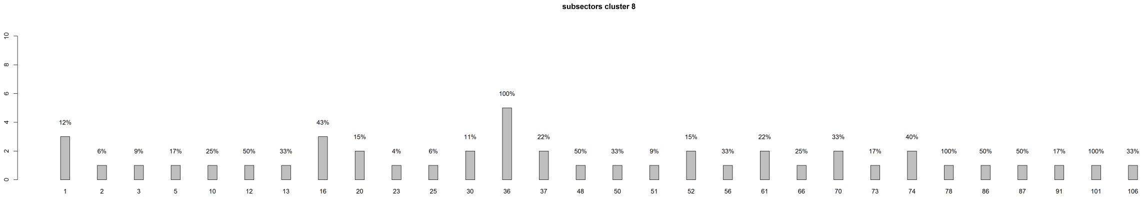
Subsectors distribution (cont.)





Subsectors distribution (cont.)

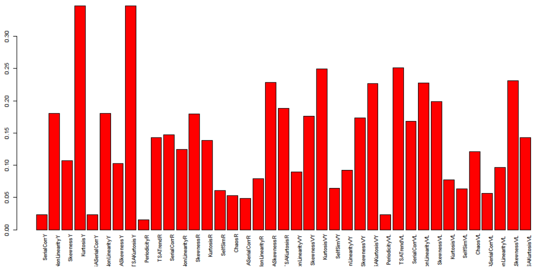




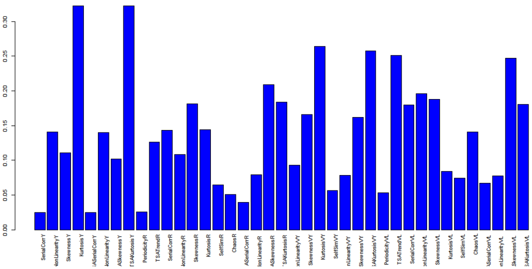
Overview

Optimization: First meta-cluster

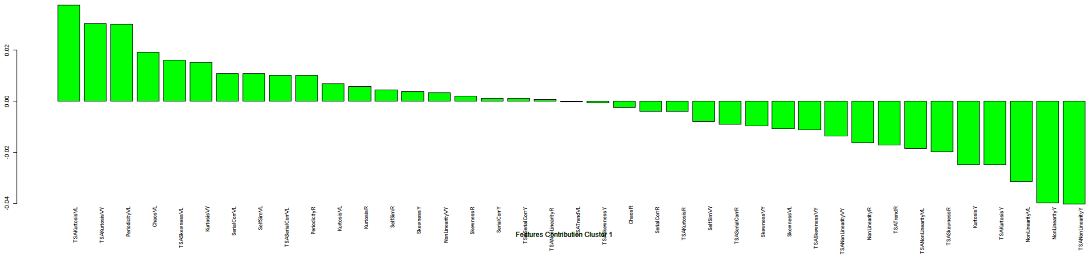
Standard Deviation in cluster 1



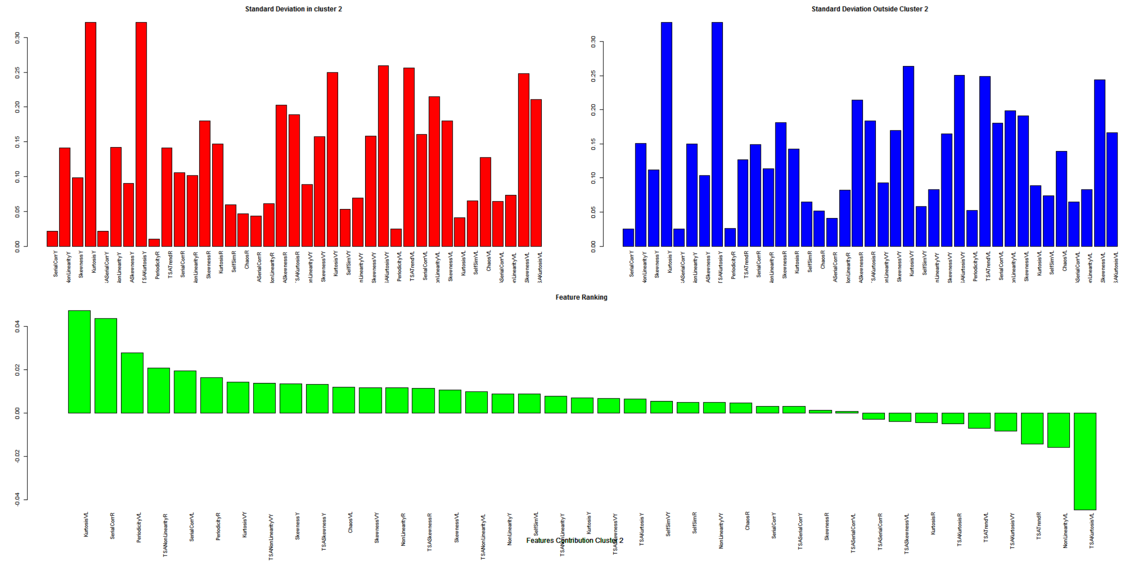
Standard Deviation Outside Cluster 1



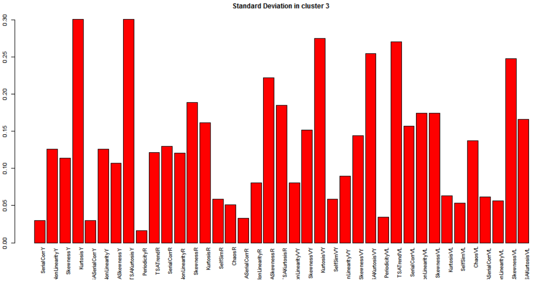
Feature Ranking

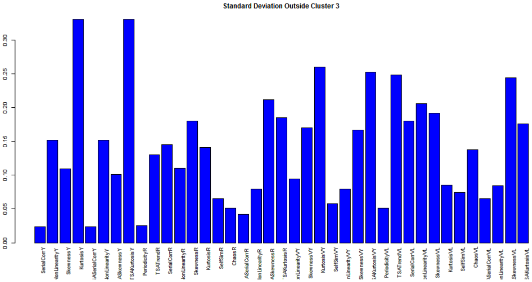


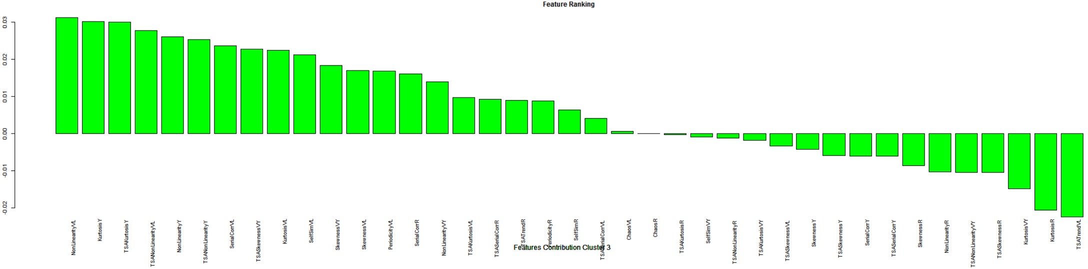
Optimization: Second meta-cluster



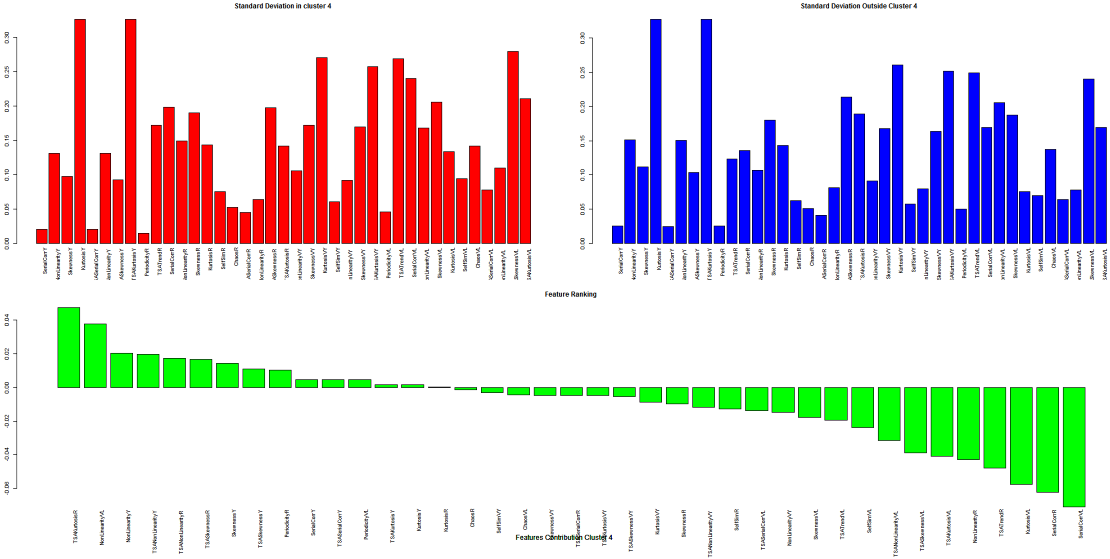
Optimization: Third meta-cluster



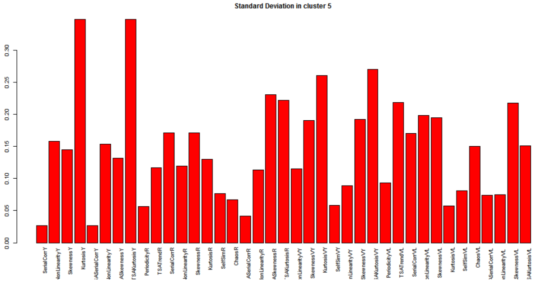


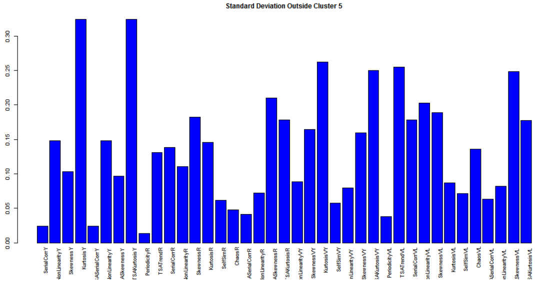


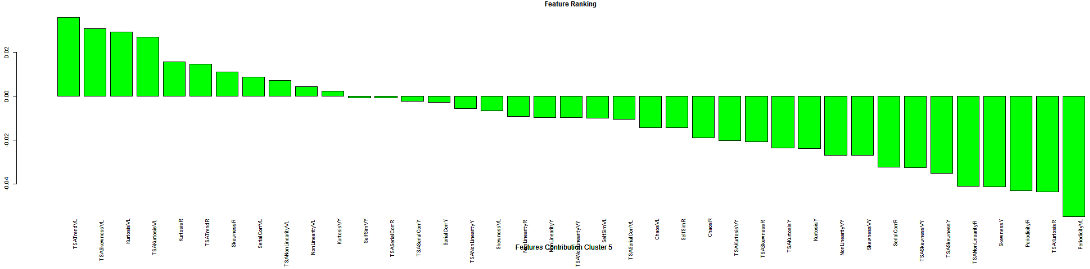
Optimization: Fourth meta-cluster



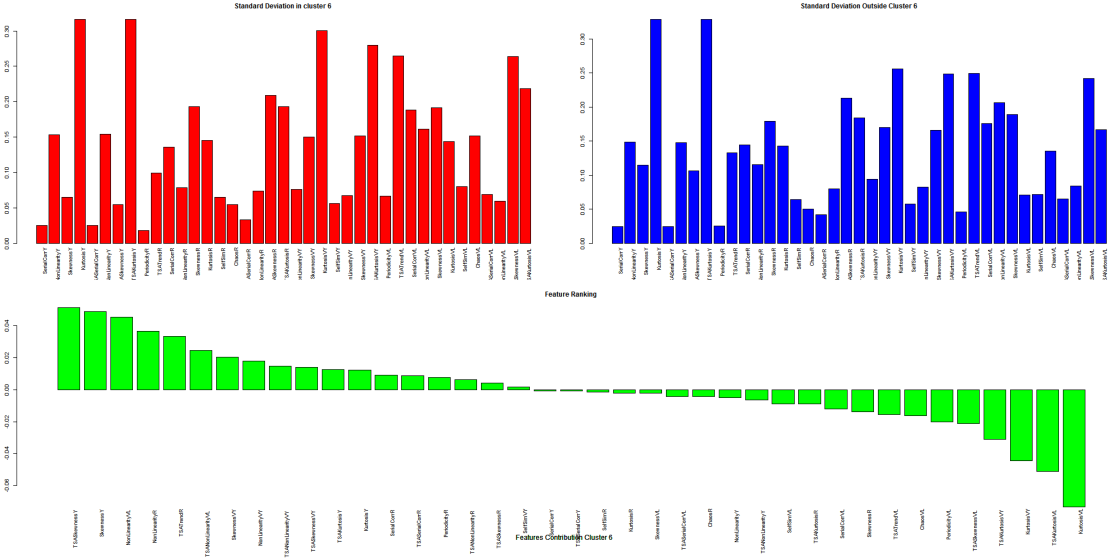
Optimization: Fifth meta-cluster



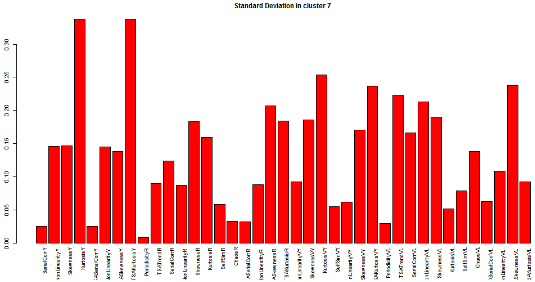


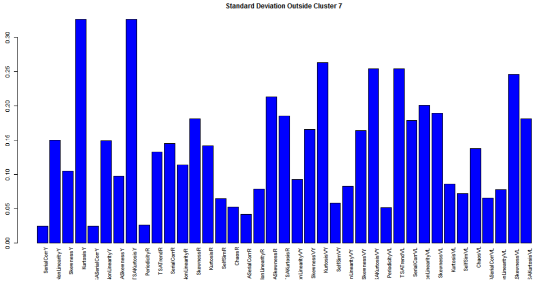


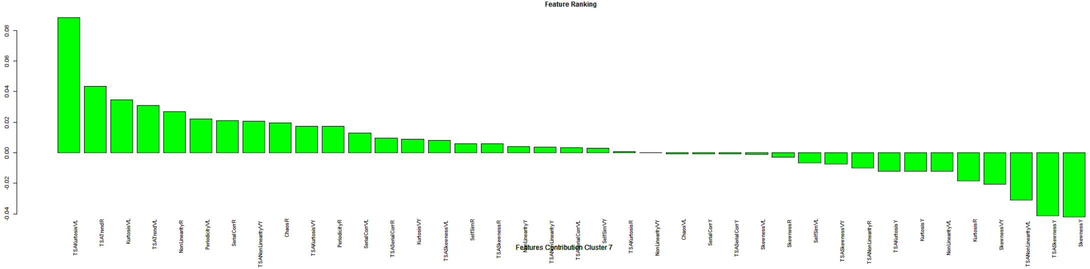
Optimization: Sixth meta-cluster



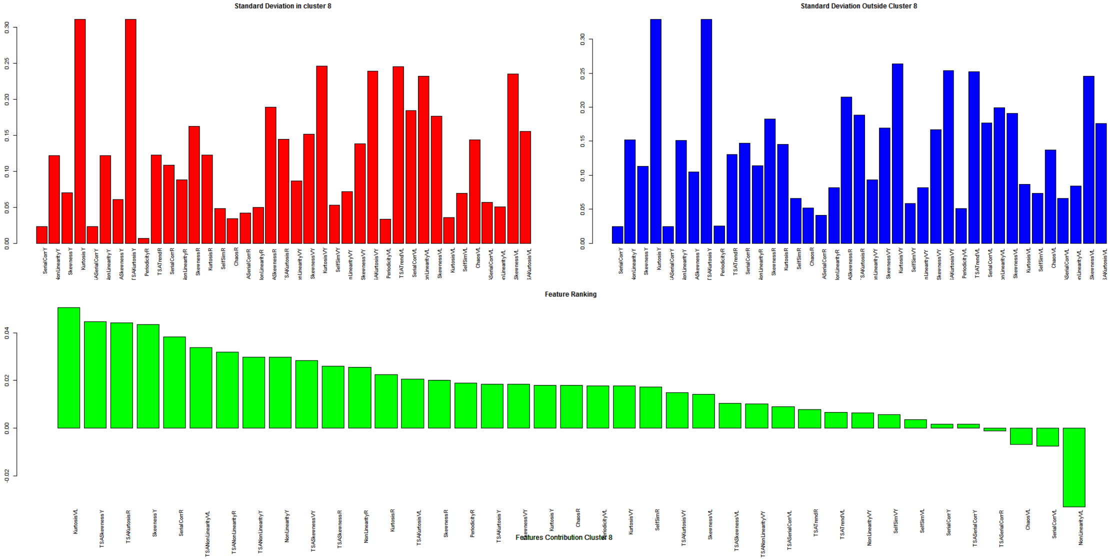
Optimization: Seventh meta-cluster







Optimization: Eighth meta-cluster



Relevant features: first meta-cluster

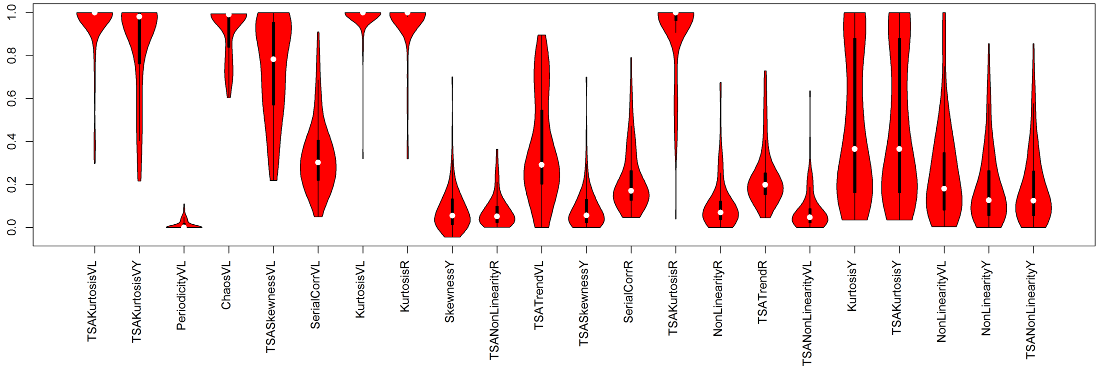


Figure: First meta-cluster

Relevant features: second meta-cluster

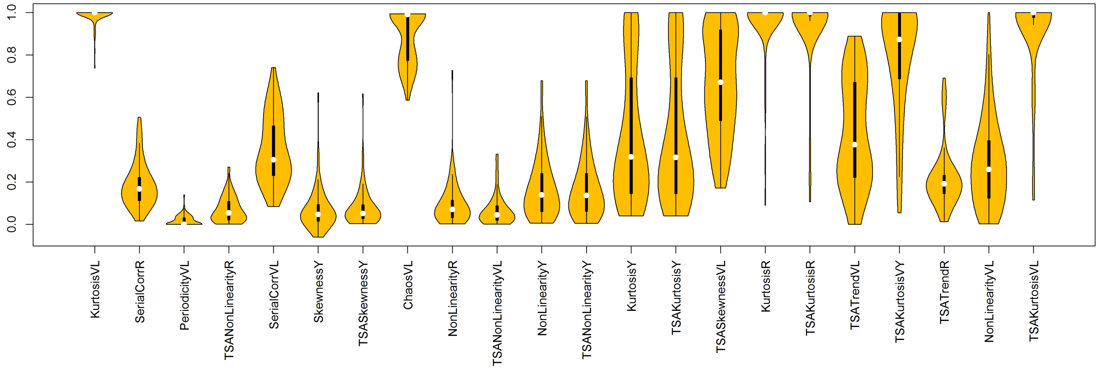


Figure: Second meta-cluster

Relevant features: third meta-cluster

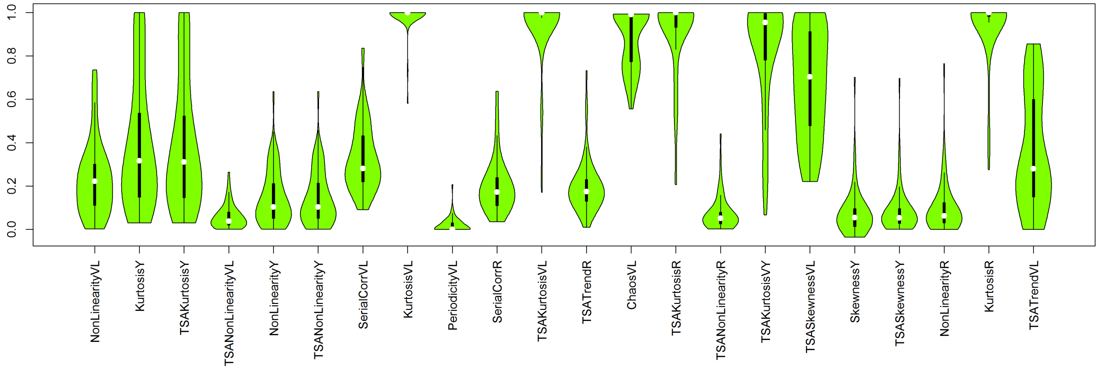


Figure: Third meta-cluster

Relevant features: fourth meta-cluster

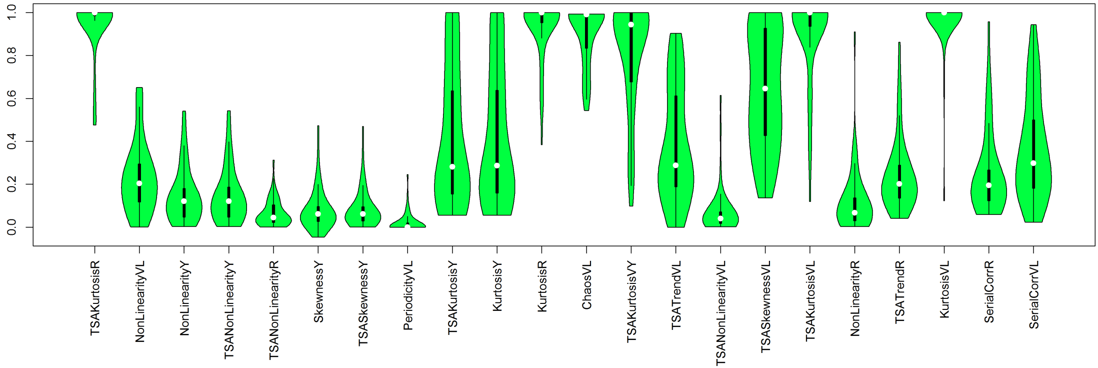


Figure: Fourth meta-cluster

Relevant features: fifth meta-cluster

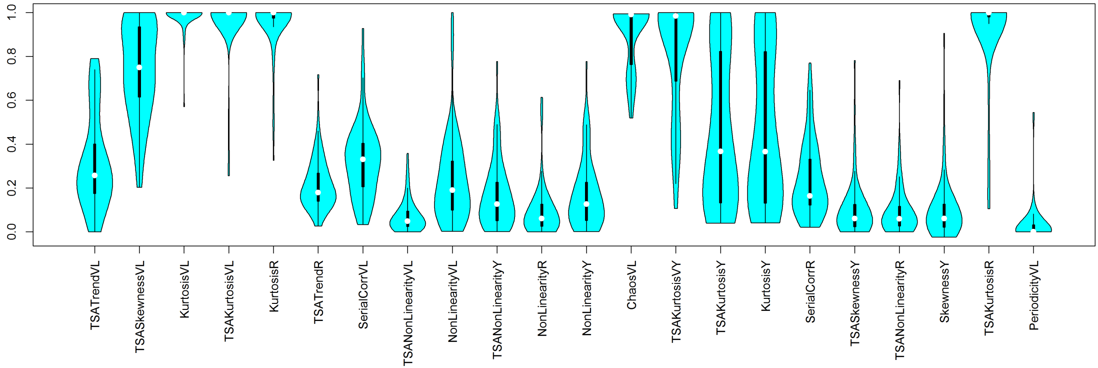


Figure: Fifth meta-cluster

Relevant features: sixth meta-cluster

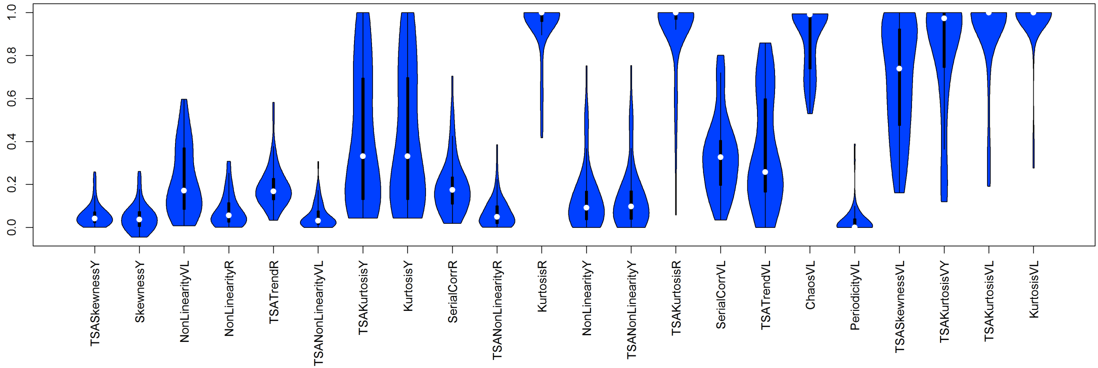


Figure: Sixth meta-cluster

Relevant features: seventh meta-cluster

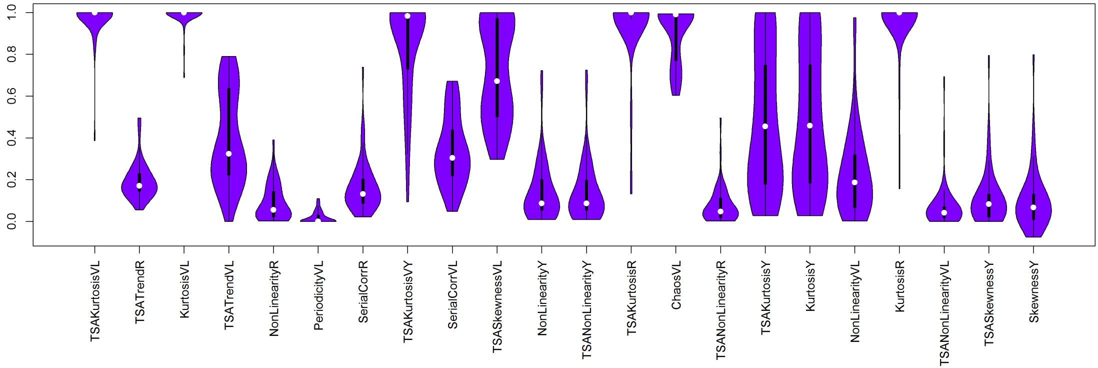


Figure: Seventh meta-cluster

Relevant features: eighth meta-cluster

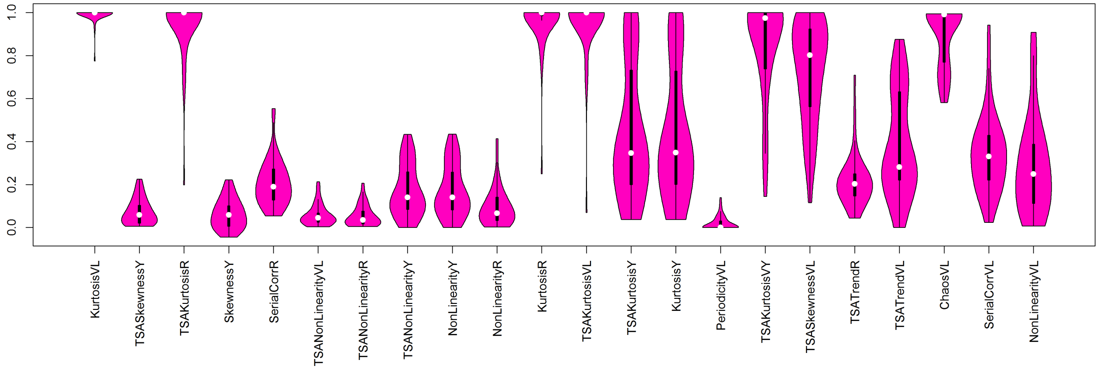
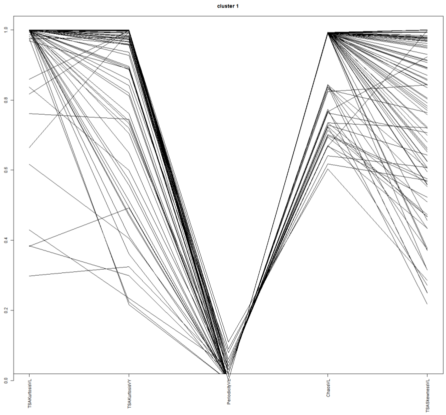
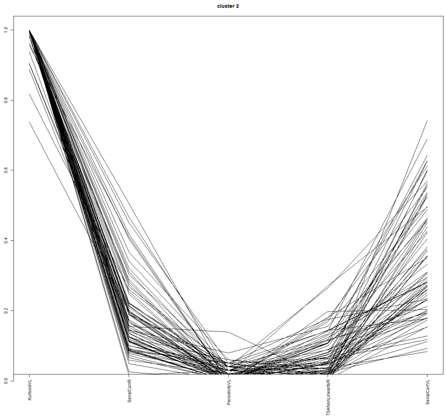


Figure: Eighth meta-cluster

Features' distribution in the meta-clusters

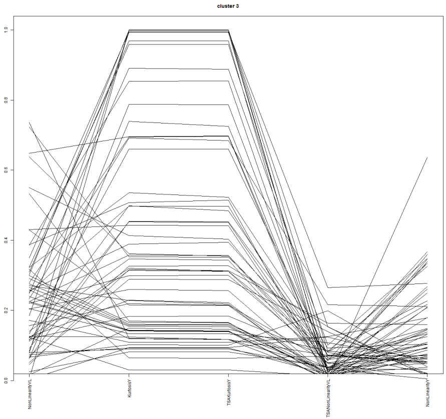


(a) First meta-cluster

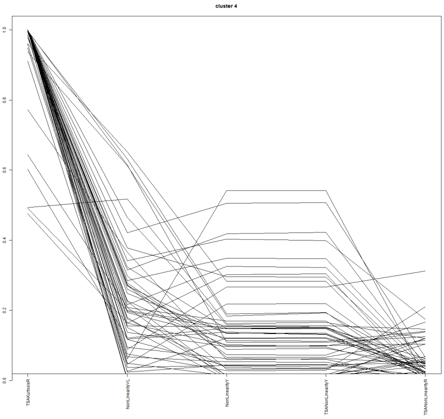


(b) Second meta-cluster

Features' distribution in the meta-clusters

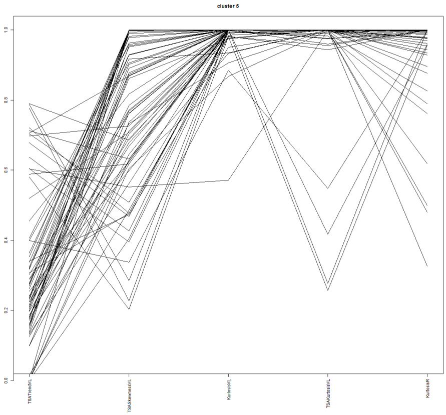


(c) Third meta-cluster

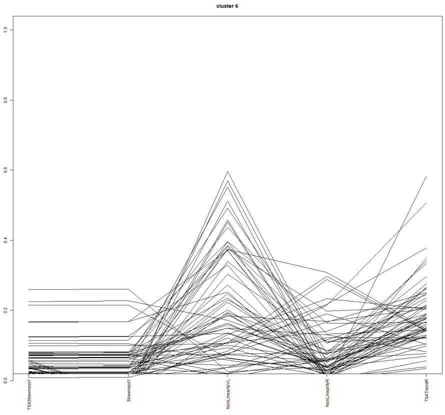


(d) Fourth meta-cluster

Features' distribution in the meta-clusters

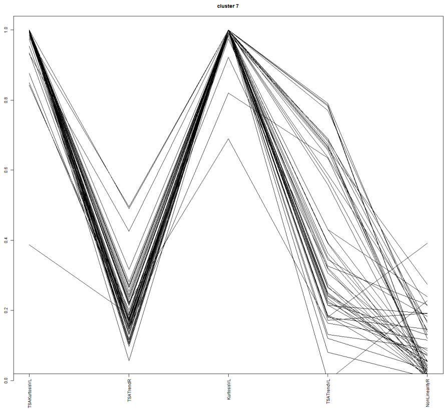


(e) Fifth meta-cluster

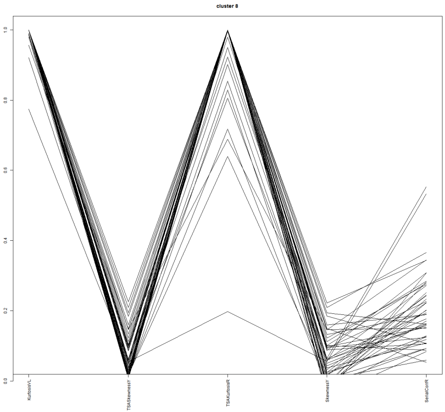


(f) Sixth meta-cluster

Features' distribution in the meta-clusters

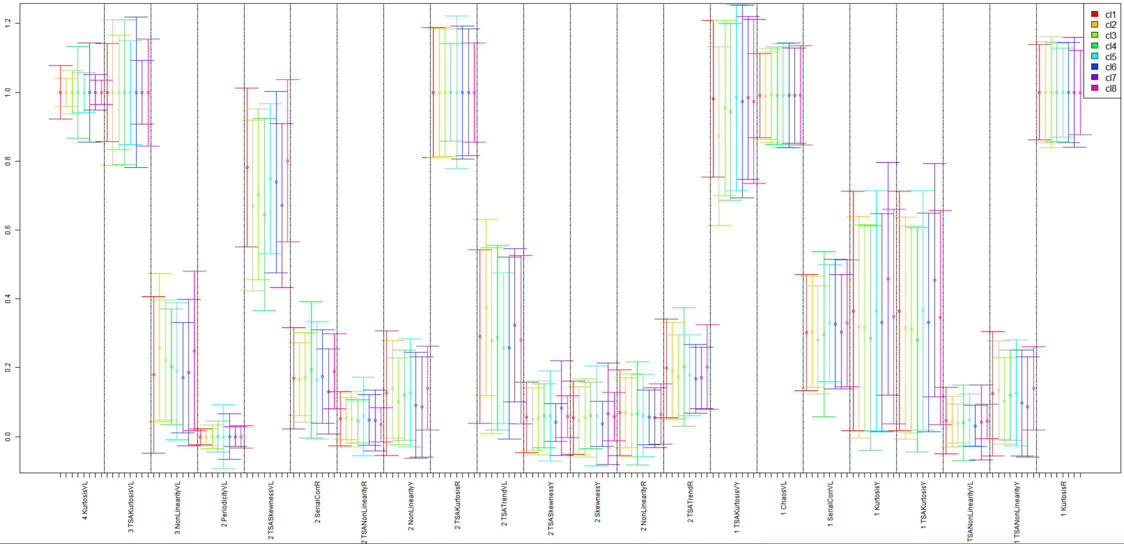


(g) Seventh meta-cluster

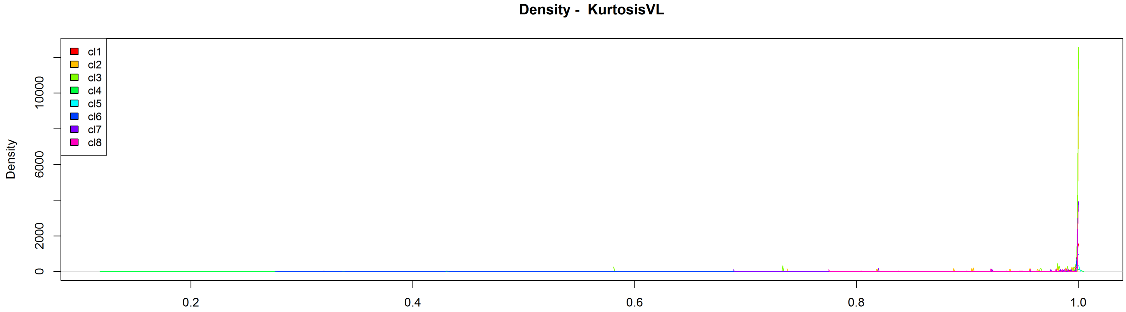


(h) Eighth meta-cluster

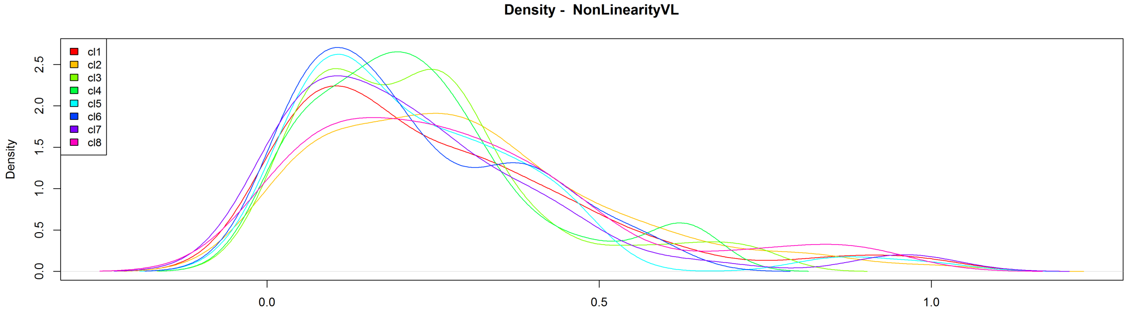
Other selection of features



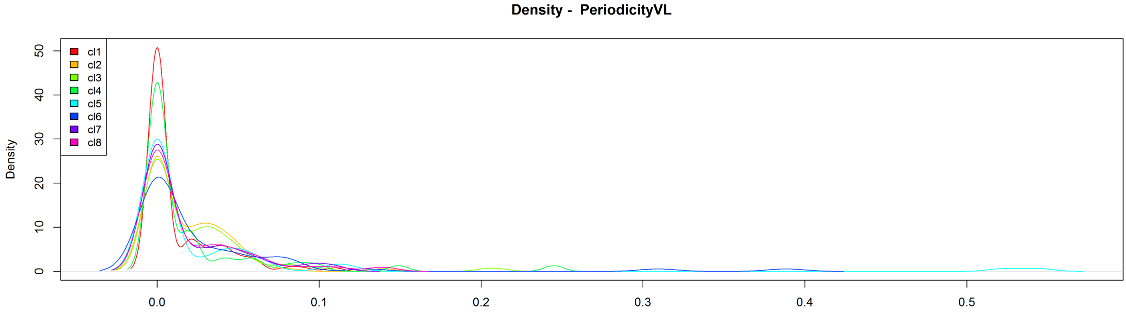
Density plot: KurtosisVL



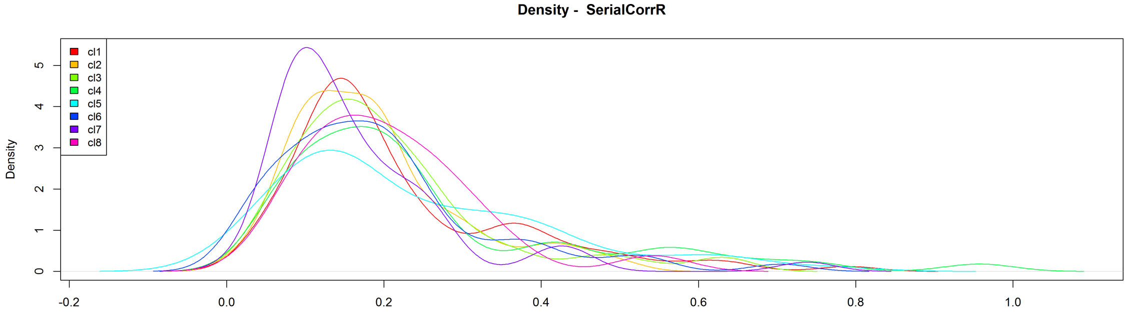
Density plot: NonLinearityVL



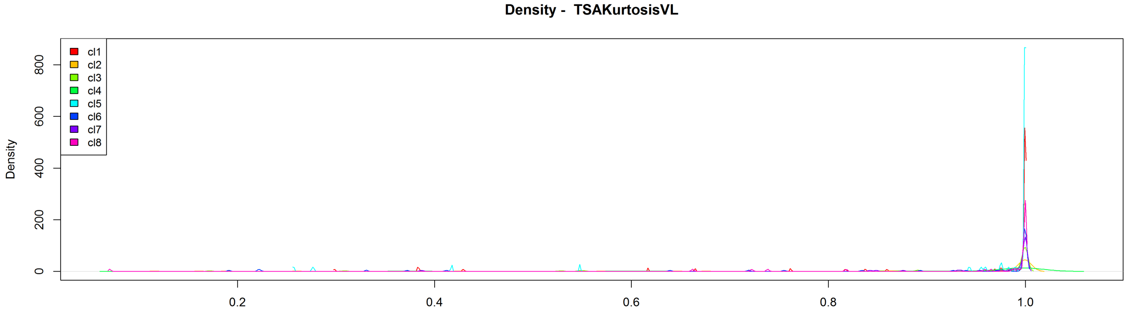
Density plot: PeriodicityVL



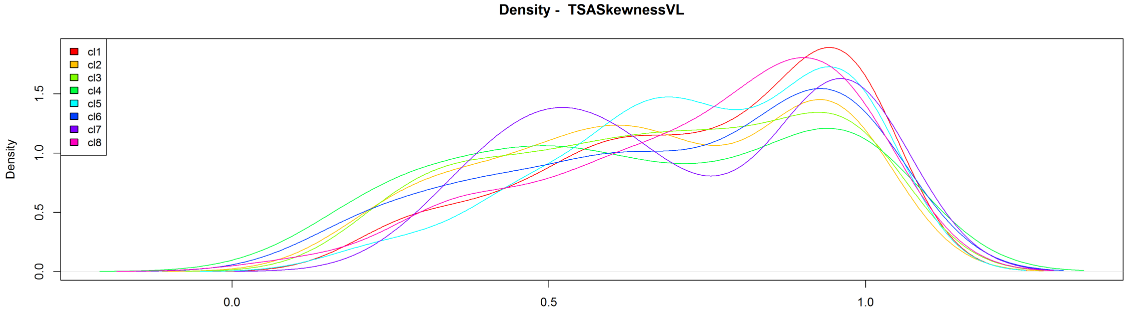
Density plot: SerialCorrR



Density plot: TSAKurtosisV



Density plot: TSASkewnessVL



Thank you for your time!