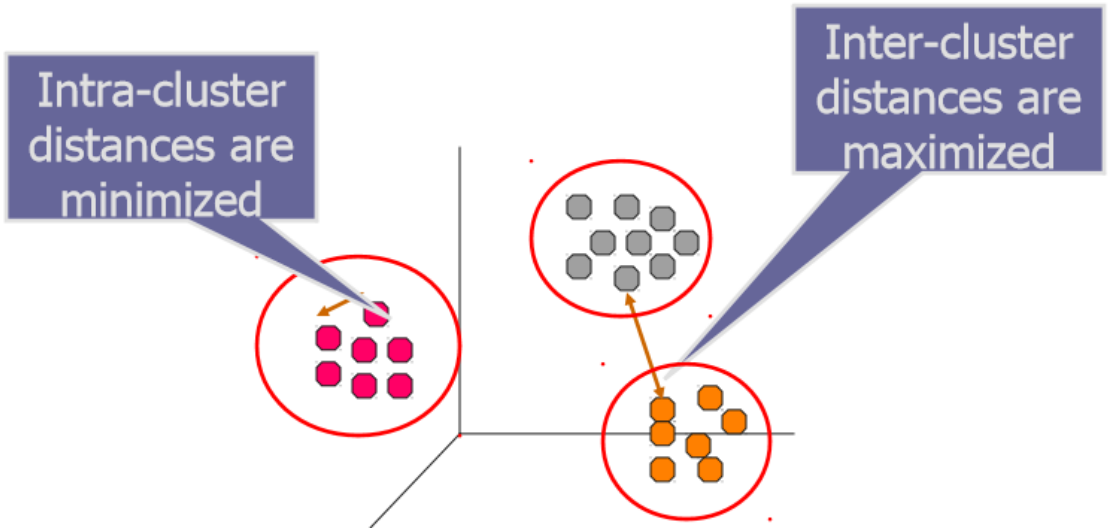
Cluster Analysis

Exercise 2

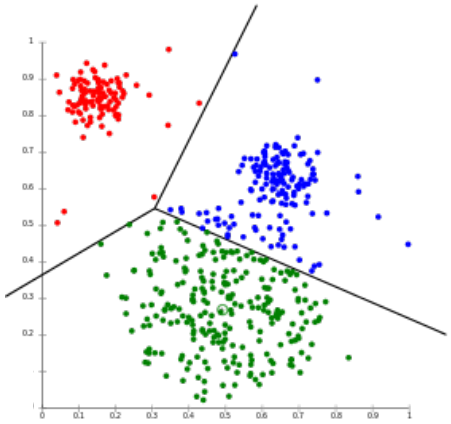
Recap: Cluster Analysis

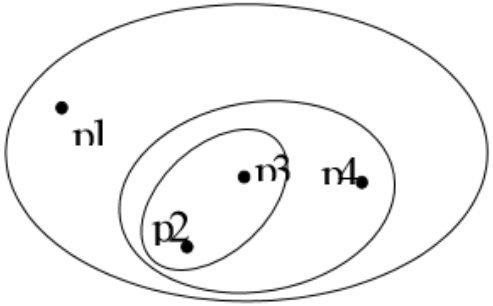
- Find groups of objects that are similar to each other and different from others
- Goal: Understand the data
  - Exploration of the data
  - The “correct” cluster assigned is not known -> unsupervised learning

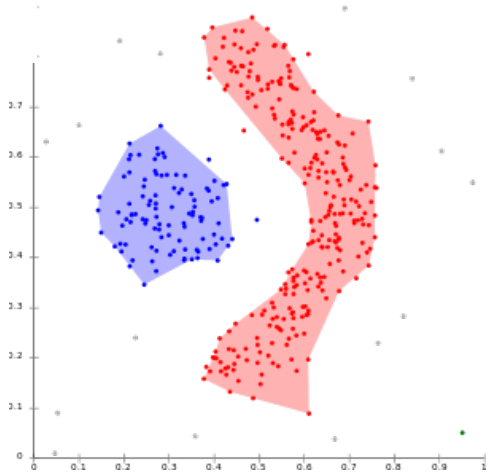


Types of Clusterings

- A “Clustering” is an assignment of examples to clusters
  - Partitional:
    - non-overlapping subsets, such that each example is in exactly one cluster
  - Hierarchical:
    - a set of nested clusters organised as a tree
  - Density based:
    - examples in dense areas form a cluster, examples in sparse areas are not assigned to a cluster

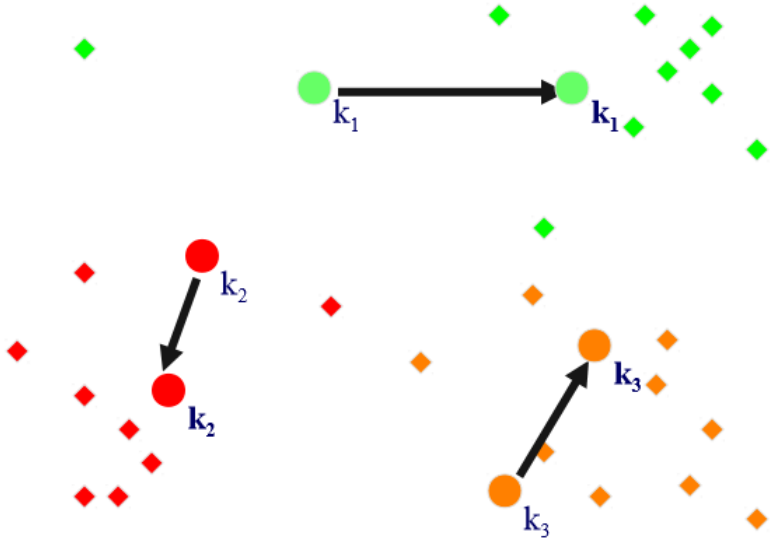






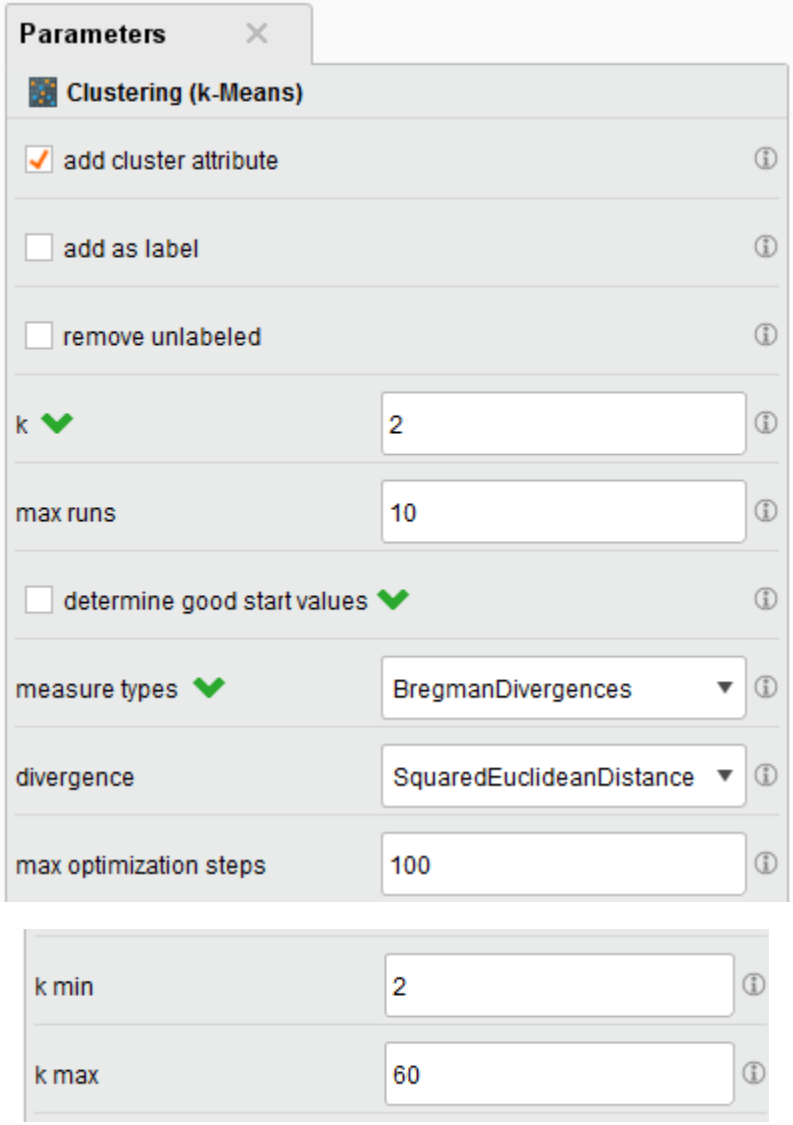
K-Means Clustering

- Partitional clustering approach
- Each example is assigned to its closest centroid
  - Requires a distance function!
- Number of clusters (k) must be specified manually
- Iteratively move the centroids to the centre of the clusters



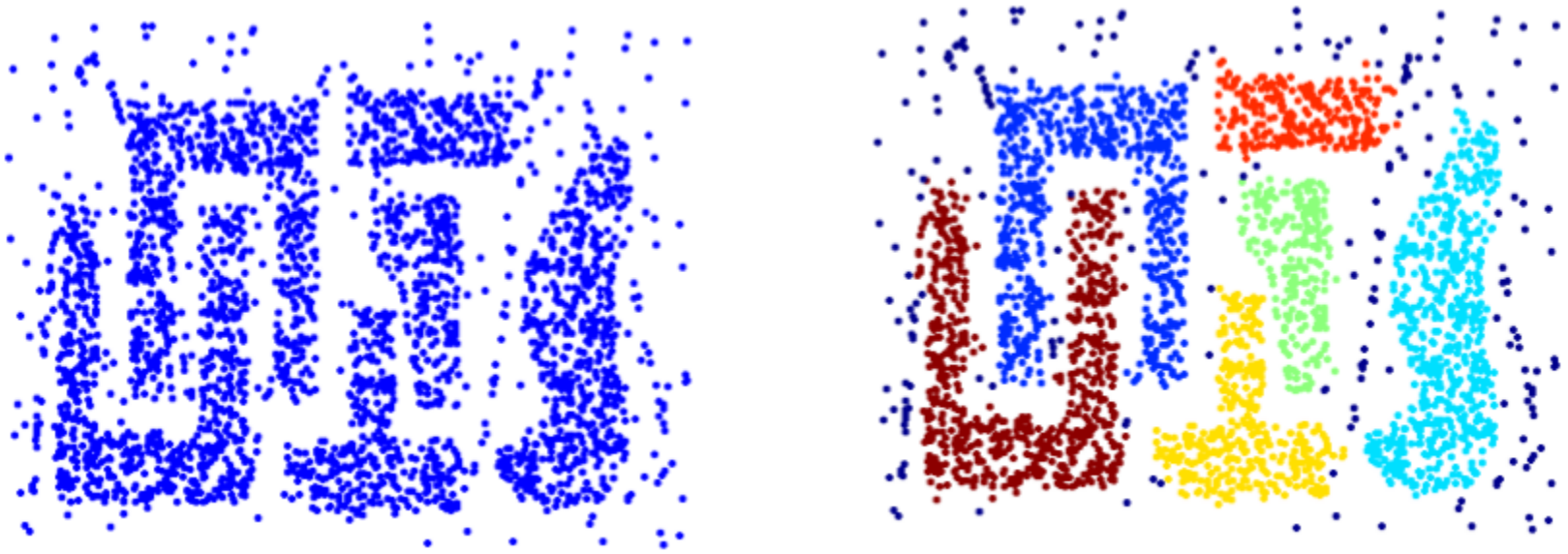
Operators: K-Means/K-Medoids/X-Means

- Input port: Example Set
- Output ports:
  - Cluster Model
  - Clustered Example Set
- Parameters:
  - K
  - Similarity Measure



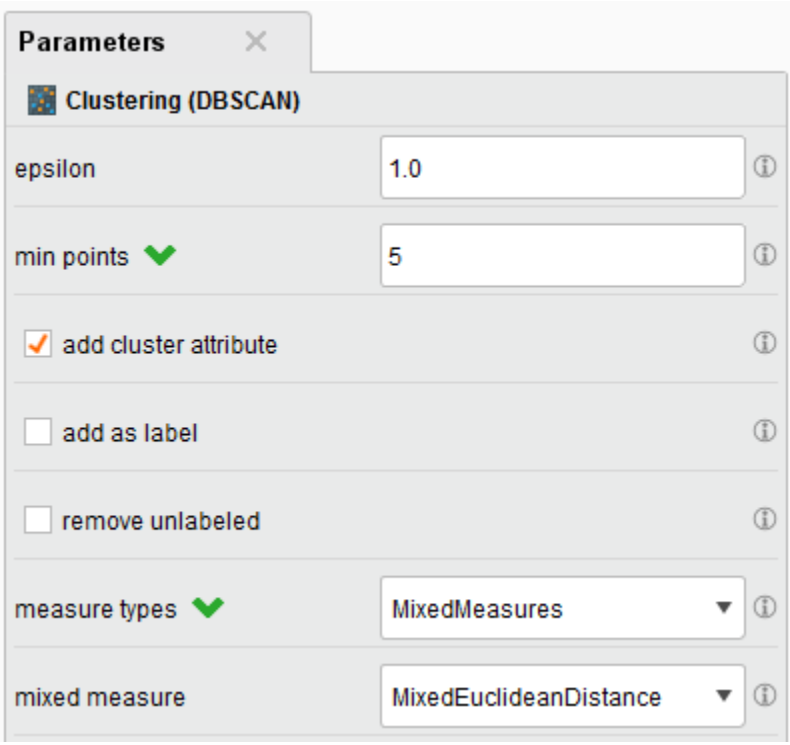
DBSCAN Clustering

- **D**ensity-**B**ased **S**patial **C**lustering of **A**pplications with **N**oise
- Examples separated into Core, Border and Noise Points
- Can handle clusters of different shapes and sizes



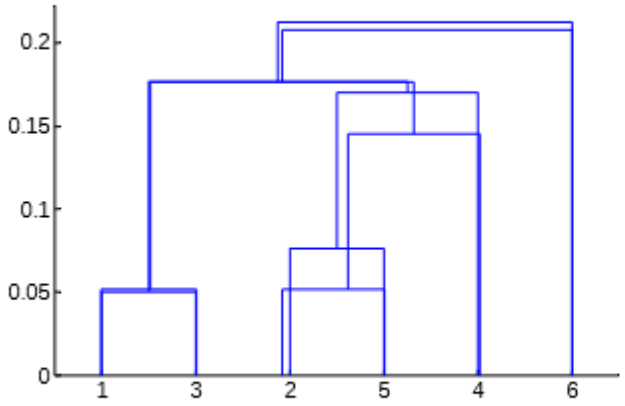
Operators: DBSCAN

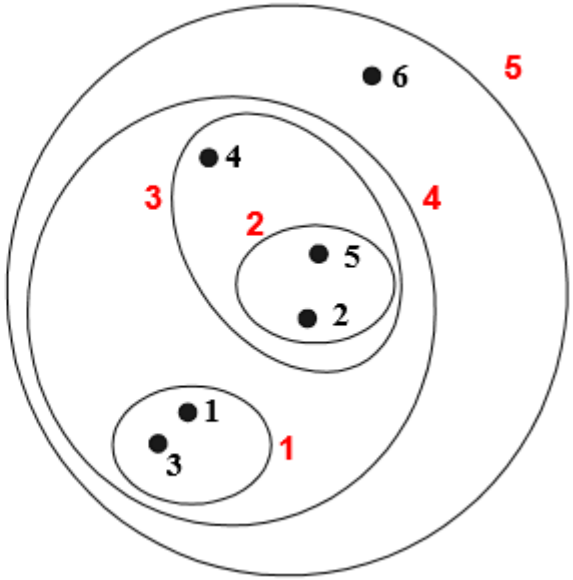
- Input port: Example Set
- Output ports:
  - Cluster Model
  - Clustered Example Set
- Parameters
  - Epsilon
  - Min points
  - Similarity Measure



Hierarchical Clustering

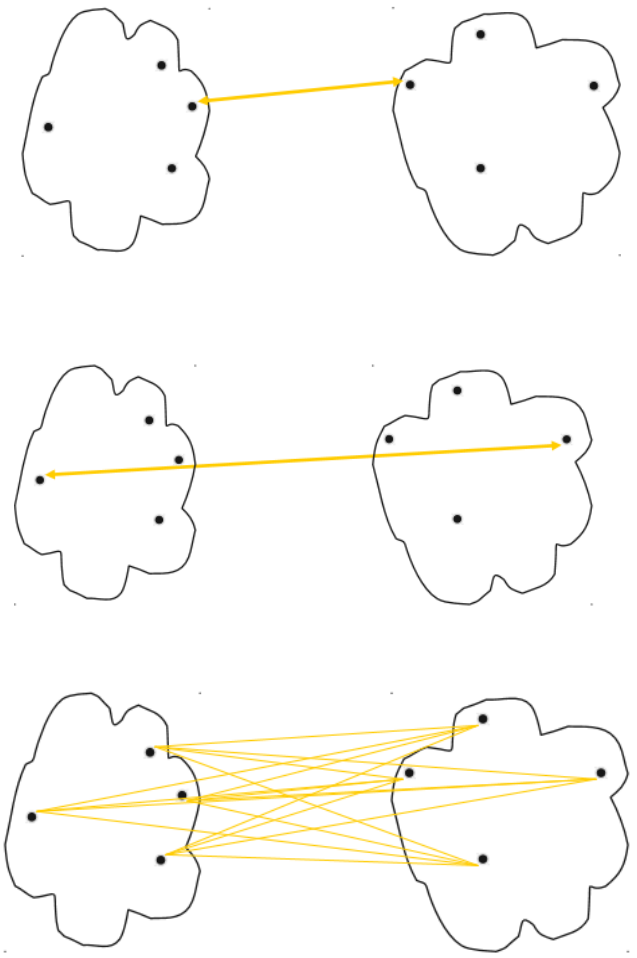
- Produces a set of nested clusters organised as tree
- Can be visualised as Dendrogram
  - Y-axis shows the distance between merged clusters
- Agglomerative: Bottom-Up
- Divisive: Top-Down





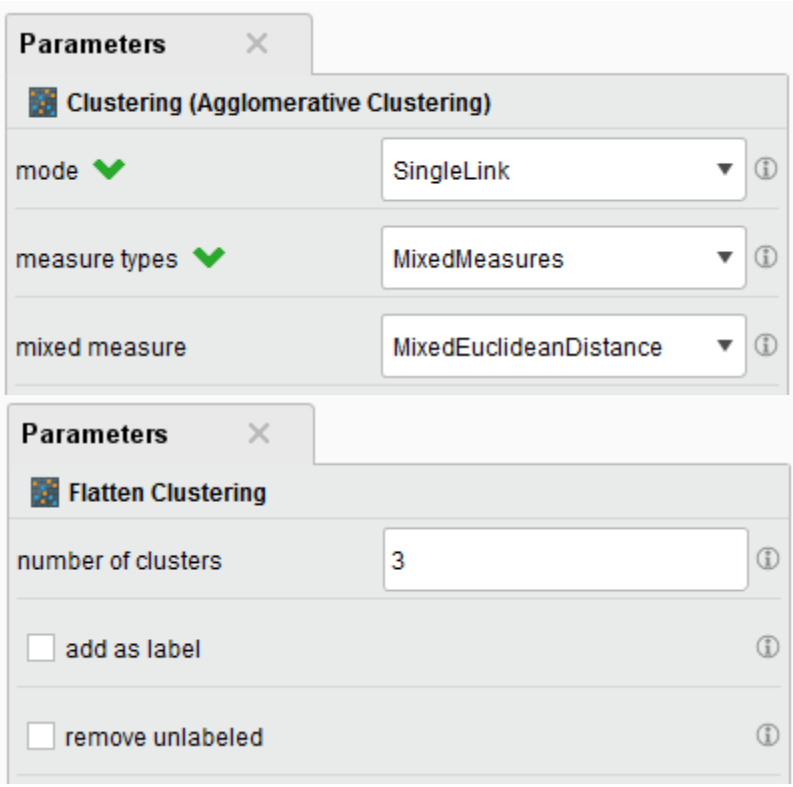
Hierarchical Clustering: Cluster Similarity

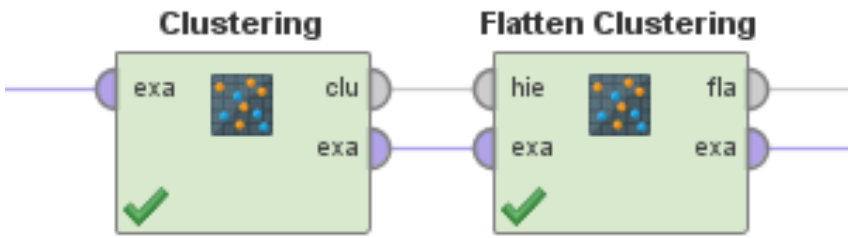
- Given two clusters with n examples, how do we define their similarity?
  - **Single Link/Min**: use the shortest distance of any two examples in the two clusters
  - **Complete Link/Max**: use the longest distance between any two examples in the two clusters
  - **Group Average**: use the average of all pair-wise distances



Operators: Agglomerative Clustering

- Input port: Example Set
- Output ports:
  - Cluster Model
  - Original Example Set
- Parameters
  - Linkage Mode
  - Similarity Measure
- Flatten Clustering cuts off the hierarchical Model
  - Assigns each example to a single cluster





Similarity Measures

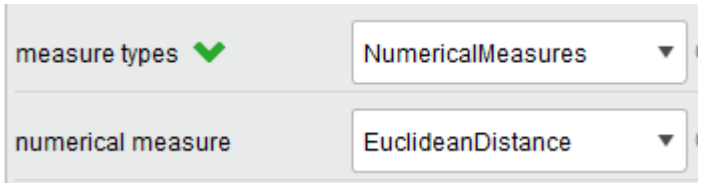
- Between two values, we can measure similarity and dissimilarity (=distance)
  - We can convert one into the other
  - Dissimilarity = max – similarity
  - 70% = 100% - 30%

Attribute Type	Dissimilarity	Similarity
Nominal	$d = \begin{cases} 0 & \text{if } p = q \\ 1 & \text{if } p \neq q \end{cases}$	$s = \begin{cases} 1 & \text{if } p = q \\ 0 & \text{if } p \neq q \end{cases}$
Ordinal	$d = \frac{\lvert p-q \rvert}{n-1}$ (values mapped to integers 0 to $n-1$, where n is the number of values)	$s = 1 - \frac{\lvert p-q \rvert}{n-1}$
Interval or Ratio	$d = \lvert p - q \rvert$	$s = -d$, $s = \frac{1}{1+d}$ or $s = 1 - \frac{d - min\_d}{max\_d - min\_d}$

Similarity Measures

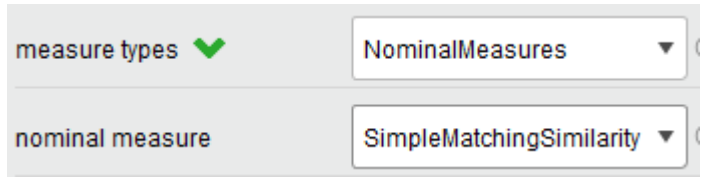
- Euclidean Distance

$$dist = \sqrt{\sum_{k=1}^{n} (p_k - q_k)^2}$$



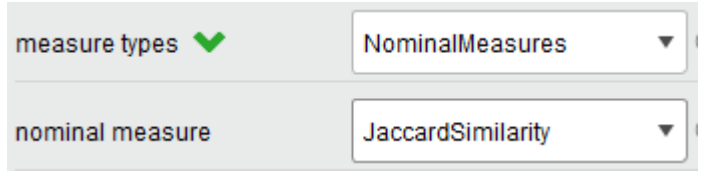
- Simple Matching Coefficient

$$SMC(x_i, x_j) = \frac{M_{11} + M_{00}}{M_{01} + M_{10} + M_{11} + M_{00}}$$



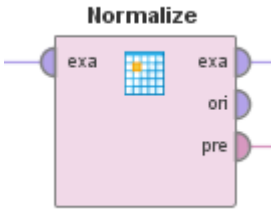
- Jaccard Coefficient

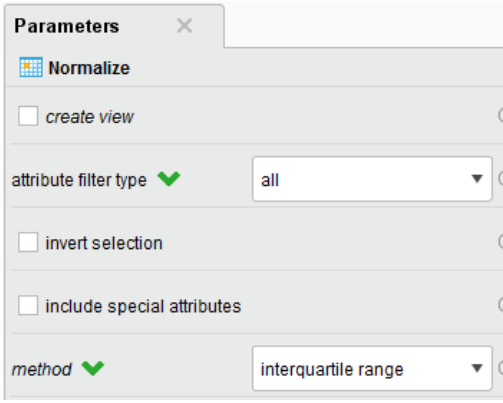
$$J(x_i, x_j) = \frac{M_{11}}{M_{01} + M_{10} + M_{11}}$$



Operators: Normalise

- Input Port: Example Set
- Output Ports:
  - Example Set
  - Original Example Set
  - Preprocessing Model





- Z-Transformation (="Statistical normalization")
  - Convert into Normal distribution with mean = 0 and variance = 1
  - The range -3 to +3 will contain 99.9% of the data
  - "Subtract the mean and divide by the standard deviation"
- Range Transformation
  - Normalises all values to the specified range.
  - "Subtract min and divide by the absolute difference between min and max"
- Proportion Transformation
  - Each value is normalised as the proportion of the attribute
  - "Divide each value by the sum of all original values"
- Interquartile Range
  - Uses the value range of the middle 50% of the data to normalise
  - "Divide by the absolute difference between the 25^{th} and 75^{th} percentile"