Distance

Mahdi Akbari Zarkesh
9612762638

INSTRUCTIONS

we want to see effect distance on result of clustering

The DataSet : wine

Classes : 3
Samples per class : [59,71,48]
Samples total : 178
Dimensionality : 13
Features : real, positive

Sqeuclidean Distance

Compute the squared Euclidean distance between two 1-D arrays.

The squared Euclidean distance between *u* and *v* is defined as

$$||u - v||_2^2$$
$$\left(\sum (w_i |(u_i - v_i)|^2)\right)$$

sqeuclidean : 0.7592592592592593

Braycurtis Distance

Compute the Bray-Curtis distance between two 1-D arrays.

Bray-Curtis distance is defined as

$$\sum |u_i - v_i| / \sum |u_i + v_i|$$

The Bray-Curtis distance is in the range [0, 1] if all coordinates are positive, and is undefined if the inputs are of length zero.

braycurtis : 0.7962962962962963

Canberra Distance

Compute the Canberra distance between two 1-D arrays.

The Canberra distance is defined as

$$d(u, v) = \sum_i \frac{|u_i - v_i|}{|u_i| + |v_i|}.$$

Canberra : 0.9444444444444444

Chebyshev Distance

Compute the Chebyshev distance.

Computes the Chebyshev distance between two 1-D arrays *u* and *v*, which is defined as

$$\max_i |u_i - v_i|.$$

chebyshev : 0.7592592592592593

cityblock Distance

Compute the City Block (Manhattan) distance.

Computes the Manhattan distance between two 1-D arrays *u* and *v*, which is defined as

$$\sum_i |u_i - v_i|.$$

cityblock : 0.7592592592592593

correlation Distance

Compute the correlation distance between two 1-D arrays.

The correlation distance between *u* and *v*, is defined as

$$1 - \frac{(u - \bar{u}) \cdot (v - \bar{v})}{||(u - \bar{u})||_2 ||(v - \bar{v})||_2}$$

where $\bar{u}$ is the mean of the elements of *u* and $x \cdot y$ is the dot product of x and y.

correlation : 0.6666666666666666

cosine Distance

Compute the Cosine distance between 1-D arrays.

The Cosine distance between *u* and *v*, is defined as

$$1 - \frac{u \cdot v}{||u||_2 ||v||_2}.$$

where $u \cdot v$ is the dot product of u and v.

cosine : 0.6851851851851852

euclidean Distance

Computes the Euclidean distance between two 1-D arrays.

The Euclidean distance between 1-D arrays *u* and *v*, is defined as

$$||u - v||_2 = \left(\sum (w_i |(u_i - v_i)|^2)\right)^{1/2}$$

euclidean : 0.7592592592592593

mahalanobis Distance

Compute the Mahalanobis distance between two 1-D arrays.

The Mahalanobis distance between 1-D arrays *u* and *v*, is defined as

$$\sqrt{(u - v)V^{-1}(u - v)^T}$$

where `V` is the covariance matrix. Note that the argument *VI* is the inverse of `V`.

mahalanobis : 0.9814814814814815

seuclidean Distance

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

seuclidean : 1.0

The result :

As you can see, different definitions of distance have a big impact on our outcome.

By repeating the same procedure on non-standardized data we found that:

1. You have to look at your data to select the distance definition
2. Manhattan and seuclidean distance works best in most cases

Thank you :)