Time series clustering

Xinhe

April 2022

1. What metric is available in the package?

In tslearn, we have dtw (dynamic time warping) and Euclidean distance.

In Julia package, I can't find the metric they use.(probably Euclidean distance)

2 and 3. What metric is commonly used in the literature? How are metrics mathematically defined?

External measures are used when the class labels are available for individual data points. (For our project, labels for the data are unknown.)

Rand Index (RI)

Adjusted Rand Index (ARI)

Adjusted Mutual Information (AMI)

Fowlkes Mallows index (FMS)

Internal measures quantify the goodness of clusters based on an optimization objective for the clustering output, without the need for class labels. (Just the case in our project)

Define our problem:

The dataset has the dimension of (n, T, d)

Where n is the number of points of the full data. n=364 for one year and n = 364*64000 for the whole 64k simulation data.

$T = 24$ and $d = 2$

T is the number of time series in one data point.

d is the dimension of data in one time period, which are LMP (P/($/MWh)) and dispatched power (D/(Mw))

So we can write the data for one year as :

$T = \{[(P_{1,1}, D_{1,1}), (P_{1,2}, D_{1,2}), ..., (P_{1,24}, D_{1,24})],$

$[(P_{2,1}, D_{2,1}), (P_{2,2}, D_{2,2}), ..., (P_{2,24}, D_{2,24})], ...,$

$[(P_{364,1}, D_{364,1}), (P_{364,2}, D_{364,2}), ..., (P_{364,24}, D_{364,24})]\}$

Let C_I be the cluster I.

Euclidean distance

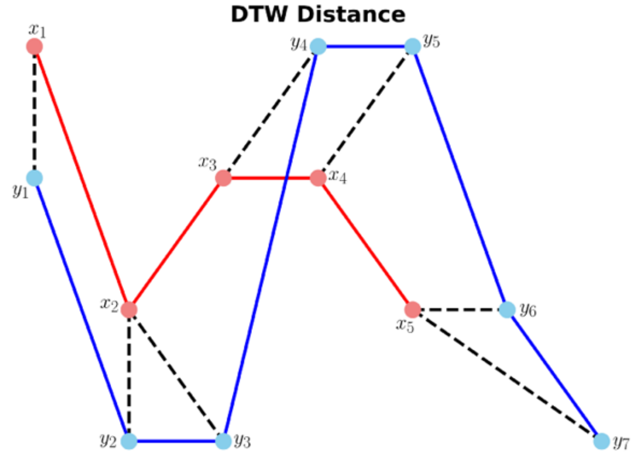
In out problem, the Euclidean distance between two point a and b is:

$point_a = [(P_{a,1}, D_{a,1}), (P_{a,2}, D_{a,2}), ..., (P_{a,24}, D_{a,24})]$

$point_b = [(P_{b,1}, D_{b,1}), (P_{b,2}, D_{b,2}), ..., (P_{b,24}, D_{b,24})]$

$d(i, j) = \sqrt[2]{\sum_{i=1}^{24} [(P_{a,i} - P_{b,i})^2 + (D_{a,i} - D_{b,i})^2]}$

Dynamic Time Warping(DTW)



In out problem, all the time series data are in the same length.

Define two time series point a and b:

$point_A = [A_1, A_2, ..., A_{24}] = [(P_{A,1}, D_{A,1}), (P_{A,2}, D_{A,2}), ..., (P_{A,24}, D_{A,24})]$

$point_B = [B_1, B_2, ..., B_{24}] = [(P_{B,1}, D_{B,1}), (P_{B,2}, D_{B,2}), ..., (P_{B,24}, D_{B,24})]$

W is the number of warping path from a to b.

$d_{DTW} = \frac{\sum_{k=1}^{W} w_k}{W}$

w_k is the length of warping path.

$w_k = d(A_i, B_j) + min(d(A_{i-1}, B_j), d(A_i, B_{j-1}), d(A_{i-1}, B_{j-1}))$

Here the $d(A_i, B_j)$ is defined as Euclidean distance:

$d(A_i, B_j) = \sqrt{(P_{A,i} - P_{B,j})^2 + (D_{A,i} - D_{B,j})^2}$

Silhouette score (available in tslearn package):

Silhouette score $s(i) = (b(i) - a(i))/max(a(i), b(i))$

a(i) = The intra-cluster distance. The average distance between point i and other points in cluster C.

$a(i) = \frac{1}{|C_I|-1} \sum_{j \in C_I, i \neq j} d(i,j)$

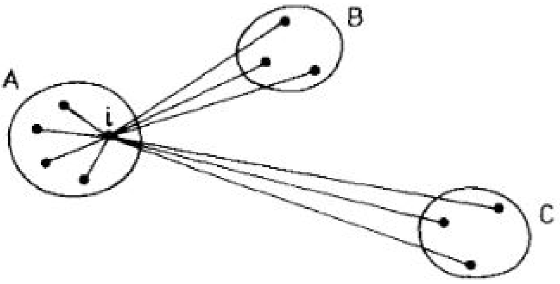
Here $d(i,j)$ is the Euclidean distance between point (i,j).

b= The inter-cluster distance. For point $i \in C_I$ and any point $J \in C_J$ in any another cluster C_J,

Calculate distance between point i and j, $d(i,j)$. The smallest mean distance of i to all points in any other cluster is $b(i)$

The cluster with this smallest mean dissimilarity is said to be the "neighboring cluster" of i because it is the next best fit cluster for point i.

$b(i) = \min_{J \neq I} \frac{1}{|C_J|} \sum_{j \in C_J} d(i,j)$



Calinski–Harabasz index (available in scikit learn)

https://scikit-learn.org/stable/modules/generated/sklearn.metrics.calinski_harabasz_score.html

Calinski–Harabasz index is the ratio of the sum of between-clusters dispersion and of inter-cluster dispersion for all clusters.

CH index is defined as:

$s = \frac{\mathrm{tr}(B_k)}{\mathrm{tr}(W_k)} \times \frac{n_E - k}{k-1}$

where n_E is the number of all data points (364 in one year). k is the number of clusters.

$tr(B_k)$ is defined as the trace of the between group dispersion matrix:

$tr(B_k) = \sum_{i=1}^{k} n_i (c_i - c_E)(c_i - c_E)^T$

where c_i is center of cluster C_i and n_i is the number of data points in cluster C_i. c_E is the center of all data points.

In our problem, $c_i = [(P_{i,c,1}, D_{i,c,1}), (P_{i,c,2}, D_{i,c,2}), ..., (P_{i,c,24}, D_{i,c,24})]$

where $(P_{i,c,1}, D_{i,c,1}) = (\frac{\sum_{j=1}^{n_i} P_{i,j,1}}{n_i}, \frac{\sum_{j=1}^{n_i} D_{i,j,1}}{n_i})$,...,$(P_{i,c,24}, D_{i,c,24}) = (\frac{\sum_{j=1}^{n_i} P_{i,j,24}}{n_i}, \frac{\sum_{j=1}^{n_i} D_{i,j,24}}{n_i})$

c_E is the center of all data points.

$c_E = [(P_{E,1}, D_{E,1}), (P_{E,2}, D_{E,2}), ..., (P_{E,24}, D_{E,24})]$

where $(P_{E,1}, D_{E,1}) = (\frac{\sum_{i=1}^{n_E} P_{i,1}}{n_E}, \frac{\sum_{i=1}^{n_E} D_{i,1}}{n_E})$,...,$(P_{E,24}, D_{E,24}) = (\frac{\sum_{i=1}^{n_E} P_{i,24}}{n_E}, \frac{\sum_{i=1}^{n_E} D_{i,24}}{n_E})$

$tr(W_k)$ is defined as the trace of the within-cluster dispersion matrix:

$W_k = \sum_{i=1}^{k} \sum_{x=1}^{n_i} (c_x - c_i)(c_x - c_i)^T$

where c_x is the data points in cluster C_i.

In our problem, $W_k = \sum_{i=1}^{k} \sum_{x=1}^{n_i} \sum_{j=1}^{24} ((P_{i,x,j} - P_{i,c,j})^2 + (D_{i,x,j} - D_{i,c,j})^2)$

Davies–Bouldin index (available in scikit learn):

https://scikit-learn.org/stable/modules/generated/sklearn.metrics.davies_bouldin_score.html

X_j is a data point assigned to cluster C_i. A_i is the centroid of C_i and N_i is the number of data points in the cluster i.

A_i is defined just the same as c_i in Calinski–Harabasz index.

$A_i = [(P_{A,i,1}, D_{A,i,1}), (P_{A,i,2}, D_{A,i,2}), ..., (P_{A,i,24}, D_{A,i,24})]$

$X_j = [(P_{j,1}, D_{j,1}), (P_{j,2}, D_{j,2}), ..., (P_{j,24}, D_{j,24})]$

Let $p = 2, q = q$

S_i is the average distance between the data points in cluster i and the centroid of the cluster.

$S_i = \left(\frac{1}{N_i} \sum_{j=1}^{N_i} \|X_j - A_i\|_p^q\right)^{1/q} = (\frac{1}{N_i} \sum_{j=1}^{N_i} [\sum_{k=1}^{24} (P_{j,k} - P_{A,i,k})^2 + (D_{j,k} - D_{A,i,k})^2]^{1/2})$

$M_{i,j}$is a measure of separation between cluster C_i and cluster C_j.

$A_j = [(P_{A,j,1}, D_{A,j,1}), (P_{A,j,2}, D_{A,j,2}), ..., (P_{A,j,24}, D_{A,j,24})]$

$M_{i,j} = \|A_i - A_j\|_p = \left(\sum_{k=1}^{24} (P_{A,i,k} - P_{A,j,k})^2 + (D_{A,i,k} - D_{A,j,k})^2\right)^{\frac{1}{2}}$

Let $R_{i,j}$ be a measure of how good the clustering scheme is.

$R_{i,j} = \frac{S_i + S_j}{M_{i,j}}$

$D_i \equiv \max_{j \neq i} R_{i,j}$

$DB \equiv \frac{1}{N} \sum_{i=1}^{N} D_i$

DB is called the Davies–Bouldin index.

4. What algorithms are available in the package?

tslearn

Only k-means and kernel k-means.

Kernel k-means: Kernel trick allows us to project our data into a higher dimensional space to achieve linear separability and solve the K-Means problem in a more efficient way.

Julia

K-means/medoids and hierarchical clustering with centroid/medoid representation

5. algorithms are recommended in the literature?

For our case, we have the consistent length for the time series data. One of the most widely used clustering techniques is k-means. I believe k-means is a promising algorithm for our project.

6. How are the algorithms mathematically defined?

k-means:

step 1: Initialize n cluster centers := $\{c_1, c_2, .., c_n\}$.

Step 2: Calculate the distance of point i to each cluster center, $d(i, c_i)$. Here we need to specify which distance metric we are using.

Step 3: Assign the point i to the closest cluster.

Step 4: Calculate the new cluster center $\{c_1, c_2, .., c_n\}$.

Repeat step 2 to step 4. Until all cluster centers no longer move. (SSE is minimized)