Similarity Preserving Representation Learning for Time Series Clustering.

Qi Lei

joint work with
Jinfeng Yi, JD AI Research
Roman Vaculin, Lingfei Wu, IBM Research
Inderjit S. Dhillon, UT Austin & Amazon
• Time series clustering is important
  ○ Biology: Multiple gene expression profile alignment
  ○ Energy: Discovering energy consumption pattern
  ○ Finance: Personal income pattern/ Discovery patterns from stock time-series/ retail pattern/ etc
  ○ Medicine: Detecting brain activity
  ○ Speech/voice recognition
  ○ User analysis
  ○ Psychology
  ○ ...

Machine learning clusterings are effective, robust, efficient, and easy to use.

They are not directly usable for time series data, due to its temporal nature, usually unequal lengths, and complex properties.

Solution: extract static features via representation learning.

Target: Given a set of $n$ time series $\mathcal{T} = \{T_1, \cdots, T_n\}$, find a mapping $f: \mathcal{T} \rightarrow \mathbb{R}^d$, s.t.,

$$S(T_i, T_j) \approx \langle f(T_i), f(T_j) \rangle \quad \forall i, j \in [n],$$
Time Series Distance Measure

- Dynamic Time Warping (DTW)
- Move-split-merge (MSM)
- KL distance
- Euclidean distance (ED)
- Pearson’s correlation coefficient and related distances
- Cosine wavelets
- ....
- **Dynamic Time Warping (DTW)**
- Move-split-merge (MSM)
- KL distance
- Euclidean distance (ED)
- Pearson’s correlation coefficient and related distances
- Cosine wavelets
- ....
Distance Measure: Dynamic Time Warping

- one of the most commonly used measure for time series similarities
- optimal global alignment between two time series
- exploit temporal distortions

Euclidean Matching

Dynamic Time Warping Matching
Pros:
• Behave well in dealing with temporal drift
• Better accuracy than Euclidean norm
• ....

Cons:
• Less efficient than Euclidean norm:
  ○ Time complexity cubic to length
• ....
Distance Matrix: Pair-wise DTW Distance

\[ S(T_i, T_j) \approx \langle f(T_i), f(T_j) \rangle \ \forall i, j \in [n], \]

\[ D_{ij} = \text{DTW}(T_i, T_j) \]
• We introduce DTW similarity from DTW distance:

\[ S(T_i, T_j) = \frac{\text{DTW}(T_i, 0)^2 + \text{DTW}(T_j, 0)^2 - \text{DTW}(T_i, T_j)^2}{2}, \]

• Intuition behind: for vector space, we have:

\[ \langle x, y \rangle = \left( \| x \|^2 + \| y \|^2 - \| x - y \|^2 \right) / 2 \]
$S_{ij} = S(T_i, T_j)$ is the induced DTW similarity.
Up to now, everything is quite standard. The idea is close to traditional kernel method. The only difference is the way we induce similarity from distance measure is different.
Theorem (informal):

Given $n$ time series generated from $k$ clusters. Suppose the clusters are distinguishable, meaning the distance within each cluster is negligible compared to the distance between each cluster, and the distance measure satisfies some relaxed version of triangle inequality. Then our generated similarity matrix is close to a low rank (of rank $k(k-1)+2$) matrix.

Other intuition for low rankness:

• DTW measures co-movement between time series, which is driven by small number of factors.

Consequence:

• Partial observation ensures good enough matrix recovery[1]

More Efficient Method

- Partial observation of a low rank plus noise matrix is enough for good recovery via matrix completion methods.

Therefore we don't need to compute all pairwise similarities.

\[ S_{ij} \approx \langle x_i, x_j \rangle, \forall (i, j) \in \Omega \]
We also propose an efficient algorithm to extract features from the partially observed similarity matrix.

Advantages:
- Exact coordinate descent -- no need to choose learning rate
- Works very efficient in practice
- Theoretical guarantee of convergence
Experimental Result: average NMI

Baseline methods
- k-Shape
- CLDS-kMeans
- Gaussian-kernels instead of our way of generating pairwise DTW similarities (Laplace-DTW-kMeans)
- kMedoids-DTW

Dataset:
- 85 UCR time series dataset

<table>
<thead>
<tr>
<th>method</th>
<th>SPIRAL-DTW-kMeans</th>
<th>Laplace-DTW-kMeans</th>
<th>k-Shape</th>
</tr>
</thead>
<tbody>
<tr>
<td>NMI</td>
<td>0.332</td>
<td>0.171</td>
<td>0.281</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>method</th>
<th>kMedoids-DTW</th>
<th>kMeans-DTW</th>
<th>CLDS-kMeans</th>
</tr>
</thead>
<tbody>
<tr>
<td>NMI</td>
<td>0.291</td>
<td>0.217</td>
<td>0.285</td>
</tr>
</tbody>
</table>

The overall clustering performance of all the proposed and baseline methods.
Baseline methods
- k-Shape
- CLDS-kMeans
- Gaussian-kernels instead of our way of generating pairwise DTW similarities (Laplace-DTW-kMeans)
- kMedoids-DTW
We proposed a framework SPIRAL that
• is flexible to multiple time series distance or similarity measures
• is very efficient to learn
• yields good clustering performance with simple static clustering algorithm like kMeans clustering
Thank you!
• perfect align the signal shapes -- high and lows

• speaking slow or fast has little influence