Statistical Learning Theory, Spring Semester 2021
Tutorial session
Constant Shift Embeddings

Teaching Assistant: Evgenii Bykovetc, ebykovetc@inf.ethz.ch, CAB F63.1

26th April 2021

Pairwise clustering lecture's highlights

From lecture (12th April 2021) you know about:

- Clustering principles
- Proximity Data and their Grouping
- **Dissimilarities** and Similarities
- **K-means clustering**
- Correlation clustering
- Shifted correlation clustering
- Graph partitioning (graph cuts)
- **Pairwise data clustering**
- **Properties of pairwise clustering costs**
- **Constant Shift Embedding**
- Alternative costs functions (normalized cut, average cut, min-max cut, adaptive radio cut, Cheeger cut)

Overview

Content[1]:

- **Data invariants and constant shift embeddings flow**
  Invariants that preserve pairwise clustering. Building pipeline for constant shift embeddings flow
- **Reconstructing embedding points**
  Process of reconstructing embedding points based on pipeline from previous section
- **Predicting cluster membership of new data**
  Dealing with new data objects

[1]Some material of this tutorial based on notes prepared by Dr. Paolo Penna and Luca Corinzia and also paper "Optimal Cluster Preserving Embedding of Nonmetric Proximity Data" V.Roth, J. Laub, M. Kawanabe, JM Buhmann. `www.researchgate.net/publication/3193640_Optimal_Cluster_Preserving_Embedding_of_Nonmetric_Proximity_Data`

Data invariants and Constant Shift Embeddings

Problem setting and pairwise dissimilarity

Input

We have N objects

$$O = \{o_1 \ldots, o_i, \ldots, o_N\}$$

which are **not** necessarily vectors. We want to cluster them based on their **pairwise dissimilarity** represented by a matrix D where

$$D_{ij} = \text{dissimilarity between object } o_i \text{ and object } o_j$$

Note: here we have only assumption that "self-dissimilarities" are equal to zero ($D_{ii} := 0 \ \forall i$)

Output

We want to map these objects into K clusters $\alpha_1, \ldots, \alpha_k$ so that each cluster contains "similar" objects.

Pairwise clustering (PC)

Our cost function sums all pairwise dissimilarities **inside** each cluster α:

Pairwise clustering cost function

$$\mathcal{R}^{pc}(c, D) := \sum_{\alpha} \sum_{(i,j)\in\mathcal{E}_{\alpha\alpha}} \frac{D_{ij}}{|\mathcal{G}_\alpha|} \tag{1}$$

where

$$\begin{aligned} \mathcal{G}_\alpha &= \{o \in O : \ c(o) = \alpha\} && \text{(objects in cluster } \alpha) \\ \mathcal{E}_{\alpha\beta} &= \{(i,j) : \ o_i \in \mathcal{G}_\alpha \text{ and } o_j \in \mathcal{G}_\beta\} && (\mathcal{E}_{\alpha\alpha} = \text{ all pairs in cluster } \alpha) \end{aligned}$$

Return the clustering c_D minimizing $\mathcal{R}^{pc}$ above
($c_D = \arg\min_c \mathcal{R}^{pc}(c, D)$).

Invariance: symmetrization

If D_{ij} is **not** symmetric

$$D_{ij} \neq D_{ji}$$

then we can make it symmetric by considering

$$D^S_{ij} := \frac{D_{ij} + D_{ji}}{2}$$

This will not change the cost function and thus the clustering:

Invariance of PC from dissimilarity symmetrization

$$\mathcal{R}^{pc}(c, D^S) = \mathcal{R}^{pc}(c, D) \tag{2}$$

Invariance: off-diagonal shift

Consider adding the same quantity d_0 to **all** elements **not** in the diagonal:

$$\tilde{D}_{ij} = D_{ij} + \begin{cases} d_0 & \text{if } i \neq j \\ 0 & \text{otherwise} \end{cases}$$

Now the cost function changes, but the computed **clustering does not change**:

Invariance of PC from dissimilarity off-diagonal shift

$$\mathcal{R}^{pc}(c, \tilde{D}) = \mathcal{R}^{pc}(c, D) + \underbrace{(N - K)d_0}_{\text{constant shift}} \tag{3}$$

One special case: Dissimilarity $\equiv$ Euclidean Distance [1]

Consider the case

$$D_{ij} = ||x_i - x_j||^2$$

so we can think of our objects as points $x_1, \ldots, x_N \in R^d$. Now we have two clustering methods (cost functions):

- K-means: $\mathcal{R}^{km}(c, x) = \sum_i ||x_i - y_{c(i)}||$ where $y_\alpha = \frac{1}{|\mathcal{G}_\alpha|} \sum_{i:c(i)=\alpha} x_i$.
- pairwise clustering: $\mathcal{R}^{pc}(c, D)$ for $D_{ij} = ||x_i - x_j||^2$.

Exercise 1

Which method should we choose? And why?

One special case: Dissimilarity $\equiv$ Euclidean Distance [2]

Equivalence of PC and k-means clustering

They are (essentially) the same:

$$\mathcal{R}^{km}(c, x) = \frac{1}{2}\mathcal{R}^{pc}(c, D) \qquad \text{for } D_{ij} = ||x_i - x_j||^2 \tag{4}$$

Decomposition [1]

Look back at the case of "geometric dissimilarities":

$$D_{ij} = ||x_i - x_j||^2 = \underbrace{||x_i||^2}_{S_{ii}} + \underbrace{||x_j||^2}_{S_{jj}} - 2\underbrace{x_i x_j}_{S_{ij}}$$

where $S_{ij} = x_i x_j$. This suggests the following idea:

Decomposition of dissimilarity

Decomposition of D with zero diagonal: find another matrix S such that

$$D_{ij} = S_{ii} + S_{jj} - 2S_{ij} \tag{5}$$

Decomposition [2]

Exercise 2

Check the following:

1. The "diagonal" elements are ok, i.e., (5) can be satisfied for $i = j$;
2. There is always one solution of (5);
3. There are infinitely many solutions of (5).

Centralization [1]

Centralized matrix

A **centralized** matrix is a matrix M such that the **sum** of the elements in each **row**, and the **sum** of the elements in each **column**, equals to zero:

$$\text{for all } i \text{ and } j \qquad \sum_k M_{ik} = 0 \qquad \text{and} \qquad \sum_k M_{kj} = 0 \tag{6}$$

Centralization [2]

An **example of a centralized matrix** is the matrix which has $1 - 1/N$ in the diagonal, and $-1/n$ off diagonal:

$$Q = \begin{pmatrix} 1-1/n & -1/n & \cdots & -1/n \\ -1/n & 1-1/n & \cdots & -1/n \\ & & \vdots & \\ -1/n & -1/n & \cdots & 1-1/n \end{pmatrix} = I_n - \frac{1}{n} O_n$$

where I is the identity n x n matrix and $O := e_n e_n^T$ is the matrix with all entries equal to 1, $e_n = \underbrace{(1, \ldots, 1)}_{n}$

Centralization of matrix

$$M \Longrightarrow M^c := QMQ \textbf{ centralized} \tag{7}$$

Centralization and decomposition [1]

Lemma 1

Relationship between centralization and decomposition
For decomposition $D_{ij} = S_{ii} + S_{jj} - 2S_{ij}$ it holds:

$$S^c = -\frac{1}{2} D^c \tag{8}$$

Moreover, matrix S^c is still a decomposition for the original D, that is,

$$D_{ij} = S^c_{ii} + S^c_{jj} - 2S^c_{ij} \tag{9}$$

This is interesting because different decompositions give the same centralization (see next slide).

Centralization and decomposition [2]

Note one interesting fact about "uniqueness" implied by (8).

Uniqueness of centralized decomposition

Take **two** different decompositions S and T of D

$$D_{ij} = S_{ii} + S_{jj} - 2S_{ij} \qquad \text{and} \qquad D_{ij} = T_{ii} + T_{jj} - 2T_{ij}$$

and centralize the two of them ($S^c = QSQ$ and $T^c = QTQ$).
Equation (8) says that $S^c = T^c$ and we could have arrived to this result in this way: first centralize D by computing $D^c = QDQ$ and then apply (8).

Theorem 1

Dissimilarity D derives from a squared Euclidean distance if and only if $S^c = -\frac{1}{2}D^c$ is positive semidefinite.

From dissimilarity to geometric distances [1]

Given our dissimilarity matrix D, let us decompose it using the matrix S as in (5).
Then our goal is to write S as

$$S = XX^T = \begin{pmatrix} | & | & | & | \\ x_1 & x_2 & \cdots & x_N \\ | & | & | & | \end{pmatrix} \begin{pmatrix} -x_1- \\ -x_2- \\ \vdots \\ -x_N- \end{pmatrix}$$

that is $S_{ij} = x_i x_j$ which then implies $D_{ij} = ||x_i - x_j||^2$.

From dissimilarity to geometric distances [2]

We need two conditions (symmetry and positive semidefinite):

- If S is **symmetric** then can be decomposed as

$$U \Lambda U^T$$

where Λ contains the eigenvalues in the main diagonal and zeros off diagonal.

- If S is not only **symmetric**, but also **positive semidefinite** then all eigenvalues are positive and Λ can be further decomposed as

$$U \Lambda^{1/2} \Lambda^{1/2} U^T$$

where $\Lambda^{1/2}$ contains the roots $\sqrt{\lambda_i}$ of the eigenvalues in the main diagonal (these roots exist because $\lambda_i \geq 0$).

S matrix representation

S could be represented as $S = \underbrace{U\Lambda^{1/2}}_{X} \underbrace{\Lambda^{1/2} U^T}_{X^T}$

Relationship between positive semi-definiteness am off-diagaonal shift [1]

Lemma 2

Make S positive semidefinite

$$S \Rightarrow \tilde{S} := S - \lambda_{\min} I \tag{10}$$

(subtract the minimum eigenvalue $\lambda_{\min} = \lambda_{\min}(S)$ of S from the diagonal – here I denotes the identity matrix)

Relationship between positive semi-definiteness am off-diagaonal shift [2]

This transformation (10) affects D which changes into $\tilde{D}$ as

$$\tilde{D}_{ij} = \tilde{S}_{ii} + \tilde{S}_{jj} - 2\tilde{S}_{ij}$$

which however is only an **off-diagonal** shift of D:

1. For $i = j$ we have $\tilde{D}_{ii} = S_{ii} - \lambda_{\min} + S_{jj} - \lambda_{\min} - 2(S_{ii} - \lambda_{\min}) = D_{ii}$
2. For $i \neq j$ we have $\tilde{D}_{ii} = S_{ii} - \lambda_{\min} + S_{jj} - \lambda_{\min} - 2(S_{ij}) = D_{ij} - 2\lambda_{\min}$

Theorem 2

Minimal shift

$$D_0 = -2\lambda_{min} \tag{11}$$

is the minimal constant such that:

$$\tilde{D} = D^S - 2\lambda_{min}(O - I) \tag{12}$$

derive from squared Euclidian distance.

Flow of constant shift embeddings [1]

The whole pipeline for dissimilarity matrix changes is the following:

Constant Shift Embeddings Flow

$$D^{\text{“generic”}} \xrightarrow[D^S_{ij} := \frac{D_{ij}+D_{ji}}{2}]{\text{Symmetrization}} D^S \xrightarrow[D_{ij}=S_{ii}+S_{jj}-2S_{ij}]{\text{Decomposition}} S \xrightarrow[S^c=QSQ]{\text{Centralization}} S^c = -\frac{1}{2}D^c$$

$$\text{Diagonal shift} \;\Big\downarrow\; \tilde{S} := S - \lambda_{\min} I$$

$$\tilde{D} \xleftarrow[\tilde{D}_{ij}=\tilde{S}_{ii}+\tilde{S}_{jj}-2\tilde{S}_{ij}]{\text{Off-diagonal shifted}} \tilde{S} = XX^T$$

Flow of constant shift embeddings [2]

Note that D and $\tilde{D}$ still give the **same clustering**:

1. Symmetrization doesn't change clustering (Equation 2);
2. S^c is also a decomposition for D^S (Equation 9);
3. $\tilde{D}$ is obtained from D^S via an **off-diagonal shift** (we are applying transformation (10) to a decomposition S^c of D^S) and doesn't change clustering.

Since S^{ps} is positive semidefinite, $\tilde{D}$ comes from a Euclidean distance, i.e.,

$$\tilde{D}_{ij} = ||x_i - x_j||^2$$

for some $x_1, \ldots, x_N \in R^d$. In particular, these vectors can be obtained by the eigenvalues of $\tilde{S}$ and the decomposition described above.

Flow of constant shift embeddings [3]

Constant Shift embedding

For any D with zero diagonal we can map our objects into points of the Euclidean space

$$o_i \to x_i \in R^d \qquad \text{(embedding)}$$

such that

$$\mathcal{R}^{pc}(c, D) = \sigma \mathcal{R}^{km}(c, X) + \sigma' \qquad \text{(constant shift)}$$

and thus K-mean clustering produces the same result.

Reconstructing Vector Embeddings

Reconstructing Vector Embeddings [1]

Algorithm for reconstruction of embedded vectors

1. Calculate the centralized dot product matrix $\tilde{S}^c = -\frac{1}{2} Q \tilde{D} Q$ from the matrix of squared Euclidean distances $\tilde{D}$.
2. Express $\tilde{S}^c$ in its eigenbasis: $\tilde{S}^c = V \Lambda V^T$,where $V = (v_1, \ldots, v_n)$ contains the eigenvectors v_i and $\Lambda = diag(\lambda_1, \ldots, \lambda_n)$ is a diagonal matrix of eigenvalues $\lambda_1, \geq, \ldots \lambda_p > \lambda_{p+1} = 0 = \cdots = \lambda_n$. Notice that, due to the centralization which introduces a linear dependency between all vectors, at least one eigenvalue equals zero, i.e., $p \leq n - 1$.
3. Calculate the n x p map matrix:

$$X_p = V_p (\Lambda_p)^{1/2}, \Lambda_p = diag(\lambda_1, \ldots, \lambda_n), \tag{13}$$

where $V_p = (v_1, \ldots, v_p)$. **The rows of X_p contain the vectors $\{x_i\}_{i=1}^n$ in p dimensional space, whose mutual distances are given by $\tilde{D}$.**

Reconstructing Vector Embeddings [2]

Scheme for reconstruction of embedded vectors

$$D^{\text{“generic”}} \xrightarrow{\text{CSE Flow}} \tilde{D}^S \xrightarrow[\tilde{D}_{ij}=\tilde{S}_{ii}+\tilde{S}_{jj}-2\tilde{S}_{ij}]{\text{Decomposition}} \tilde{S} \xrightarrow[\tilde{S}^c=Q\tilde{S}Q]{\text{Centralization}} \tilde{S}^c = -\frac{1}{2}\tilde{D}^c$$

$$\Big\downarrow \text{loss-free reconstruction}$$

$$X_t = V_t(\Lambda_t)^{1/2}, t < p \xleftarrow{\text{Approximation and denoising}} X_p = V_p(\Lambda_p)^{1/2}$$

Predicting the cluster membership of new data

Predicting the cluster membership of new data [1]

Problem

We are given M new objects and the corresponding $M \times N$ matrix of pairwise dissimilarities D_{ij}^{new} between these new objects and all n original objects. We would like to predict the cluster membership $\hat{c}$ of the new objects.

Note

Because of eigenvalue equation:

$$\tilde{S}^c V_p = V_p \Lambda_p \tag{14}$$

We have:

$$X_p = \tilde{S}^c V_p (\Lambda_p)^{-1/2} \tag{15}$$

Predicting the cluster membership of new data [2]

Algorithm for clustering new data

1. Compute the matrix S^{new} defined by:

$$D_{ij}^{new} = S_{ii}^{new} + \tilde{S}_{jj}^{c} - 2S_{ij}^{new} \tag{16}$$

2. Calculate

$$(S^{new})^{c} := \frac{1}{2}[D^{new}(I_n - \frac{1}{n}O_n) - \frac{1}{n}e_m e_n^T + \tilde{D}(I_n - \frac{1}{n}e_n e_n^T)] \tag{17}$$

3. Project the objects represented by $(S^{new})^{c}$ into the coordinate system spanned by the eigenvectors V_p of the matrix $\tilde{S}$:

$$X_p^{new} = (S^{new})^{c} V_p (\Lambda_p)^{-1/2} \tag{18}$$

4. Assigning objects to the cluster with the closest centroid vector:

$$\hat{c}_i = \underset{c}{\operatorname{argmin}} ||(x_p^{new})_i - y_{c(i)}|| \tag{19}$$

Predicting the cluster membership of new data [3]

Scheme for clustering new data

$$D^{new} \xrightarrow[D^{new}_{ij} = S^{new}_{ii} + \tilde{S}^{c}_{jj} - 2S^{new}_{ij}]{\text{Decomposition}} S^{new} \xrightarrow[\begin{array}{c}(S^{new})^c := \frac{1}{2}[D^{new}(I_n - \frac{1}{n}O_n) - \\ - \frac{1}{n}e_m e_n^T + \tilde{D}(I_n - \frac{1}{n}e_n e_n^T)]\end{array}]{\text{Centralization}} (S^{new})^c$$

$$\downarrow \text{new data} \quad \text{CSE embeddings}$$

$$\hat{c}_i = \underset{c}{\operatorname{argmin}} ||(x_p^{new})_i - y_{c(i)}|| \xleftarrow[\text{to closest centroids} \{y_1, \ldots, y_k\}]{\text{Assignments } \hat{c}_i \text{ to } (x_p^{new})_i} X_p^{new} = (S^{new})^c V_p (\Lambda_p)^{-1/2}$$

Questions

Thank you for your attention!

Questions?

Link to the original paper:
www.researchgate.net/publication/3193640_Optimal_Cluster_Preserving_Embedding_of_Nonmetric_Proximity_Data