Nonconvex Clustering

Sibylle Hess

Technische Universiteit Eindhoven University of Technology

The k-means Lecture..

The Most Important Slide of the k-means Lecture

Theorem (Equivalent k-means objectives)

The following objectives are equivalent

$$\min_{Y,X} \sum_{s=1}^{r}\sum_{i=1}^{n} Y_{is}\|D_{i\cdot} - X_{\cdot s}^\top\|^2 \quad s.t.\ X \in \mathbb{R}^{d\times r}, Y \in \mathbb{1}^{n\times r} \tag{1}$$

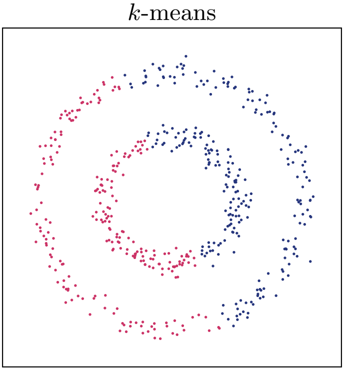
$$\min_{Y} \|D - YX^\top\|^2 \quad s.t.\ X = D^\top Y(Y^\top Y)^{-1}, Y \in \mathbb{1}^{n\times r} \tag{2}$$

$$\min_{Y,X} \|D - YX^\top\|^2 \quad s.t.\ X \in \mathbb{R}^{d\times r}, Y \in \mathbb{1}^{n\times r} \tag{3}$$

1

Informal Problem Description

Problem: k-means can Only Identify Convex Clusters



The cluster-separating boundary between two centroids is always linear.

What do we do if we have nonlinearly separated clusters?

Feature Transformation and Kernel Trick

How was that Again with Kernels?

Use a feature transformation to map points to a space where clusters are linearly separable:

$$x \to \phi(x).$$

Problem: Computing $\phi(x)$ for every data point might be costly or impossible, $\phi(x)$ might be infinite-dimensional (see RBF kernel).

Solution: We don't need ϕ, we just need the inner product

$$\phi(x)^\top \phi(y)$$

The Kernel Matrix

Defining for $D \in \mathbb{R}^{n \times d}$ the row-wise applied feature transformation

$$\phi(D) = \begin{pmatrix} -- & \phi(D_{1\cdot}) & -- \\ & \vdots & \\ -- & \phi(D_{n\cdot}) & -- \end{pmatrix},$$

the kernel matrix is given by

$$K = \phi(D)\phi(D)^\top \in \mathbb{R}^{n \times n}.$$

2

Derive the Formal Problem Definition

The Kernel k-means Objective

Given: a data matrix $D \in \mathbb{R}^{n \times d}$, a feature transformation $\phi : \mathbb{R}^d \rightarrow \mathbb{R}^p$ mapping into a p-dimensional feature space, where $p \in \mathbb{N} \cup \{\infty\}$, and the number of clusters r.

Find: clusters indicated by the matrix $Y \in \mathbb{1}^{n \times r}$ which minimize the within cluster scatter in the transformed feature space

$$\min_Y \|\phi(D) - YX^\top\|^2 \text{ s.t. } X = \phi(D^\top)Y(Y^\top Y)^{-1}, Y \in \mathbb{1}^{n \times r} \quad (4)$$

3 Optimization

If we want to apply the **kernel trick**, then we need to state the kernel k-means objective with respect to the **inner product of data points**.

Representing Data by the Inner Product Only

Theorem (k-means trace objective)

The k-means objective in Eq. (1) is equivalent to

$$\max_{Y} \operatorname{tr}(Z^\top DD^\top Z) \qquad s.t.\ Z = Y(Y^\top Y)^{-1/2}, Y \in \mathbb{1}^{n\times r} \tag{5}$$

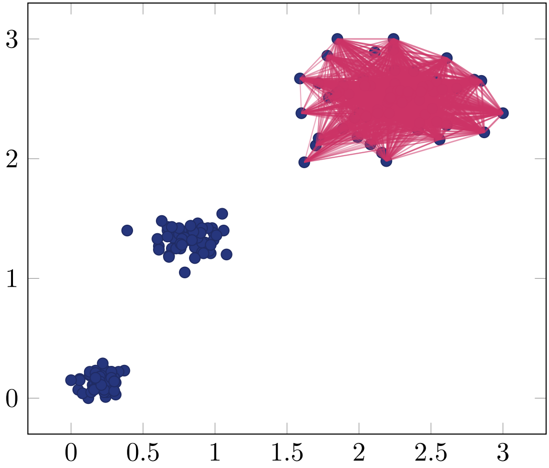
Interpretation: Clusters are now defined with respect to the inner product similarity:

$$sim(i,j) = D_{i\cdot}D_{j\cdot}^\top = \cos(\sphericalangle(D_{i\cdot}, D_{j\cdot}))\|D_{i\cdot}\|\|D_{j\cdot}\|$$

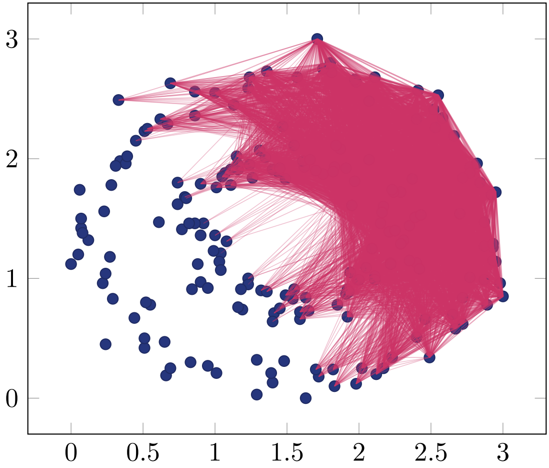
Points within one cluster need to be similar:

$$\operatorname{tr}(Z^\top DD^\top Z) = \sum_{s=1}^{r} \frac{Y_{\cdot s}^\top DD^\top Y_{\cdot s}}{|Y_{\cdot s}|} = \sum_{s=1}^{r} \frac{1}{|\mathcal{C}_s|} \sum_{i,j\in\mathcal{C}_s} D_{i\cdot}D_{j\cdot}^\top$$

The Inner Product Similarity and Convex Clusters



The Inner Product Similarity and Nonconvex Clusters



Kernel k-means

Theorem (Equivalent kernel k-means objectives)

Given the kernel matrix $K = \phi(D)\phi(D)^\top$, the following objectives are equivalent:

$$\min_{Y} \|\phi(D) - YX^\top\|^2 \ \ s.t.\ \ X = \phi(D^\top)Y(Y^\top Y)^{-1}, Y \in \mathbb{1}^{n\times r} \quad (6)$$

$$\max_{Y} \operatorname{tr}(Z^\top KZ) \qquad s.t.\ \ Z = Y(Y^\top Y)^{-1/2}, Y \in \mathbb{1}^{n\times r} \quad (7)$$

Problem: We do not know how to optimize Eq. (7), we only know how to optimize Eq. (6), but we do not want to compute ϕ!
Idea: We go the other way round: from the kernel matrix to the inner product.

Eigendecomposition of Symmetric Matrices

Theorem (Eigendecomposition of symmetric matrices)

For every symmetric matrix $K = K^\top \in \mathbb{R}^{n\times n}$ *there exists an orthogonal matrix* $V \in \mathbb{R}^{n\times n}$ *and a diagonal matrix* $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$ *where* $|\lambda_1| \geq \ldots \geq |\lambda_n|$ *such that*

$$K = V\Lambda V^\top$$

Every symmetric matrix $K \in \mathbb{R}^{n\times n}$ has a symmetric decomposition $K = A^\top A$ if and only if the eigenvalues of K are nonnegative. This is equivalent to K being positive semi-definite.

Kernel matrices are positive semi-definite!

Kernel k-means Inside Out

Theorem (Equivalent kernel k-means objectives)

Given a kernel matrix and its symmetric decomposition $K = AA^\top$, the following objectives are equivalent:

$$\min_Y \|A - YX^\top\|^2 \qquad s.t.\ X = A^\top Y(Y^\top Y)^{-1}, Y \in \mathbb{1}^{n\times r} \qquad (8)$$

$$\max_Y \operatorname{tr}(Z^\top KZ) \qquad s.t.\ Z = Y(Y^\top Y)^{-1/2}, Y \in \mathbb{1}^{n\times r} \qquad (9)$$

Algorithm Idea: Use the objective in Eq. (8): compute a symmetric decomposition $AA^\top = K$ by means of the eigendecomposition $A = V\Lambda^{1/2}$ and run k-means on A.

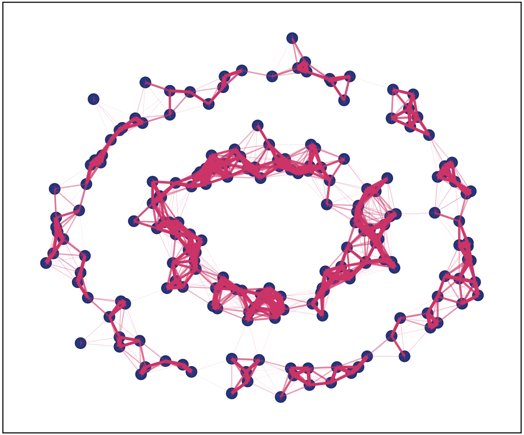
The Kernel k-means Algorithm

function KERNELKMEANS(r, K)
    $(V, \Lambda) \leftarrow$ EIGENDECOMPOSITION(K)
    $A \leftarrow V\Lambda^{1/2}$ $\triangleright AA^\top = K$
    $(X, Y) \leftarrow$KMEANS(A,r)
    return Y
end function

Let's try this kernel k-means idea on the two circles dataset.

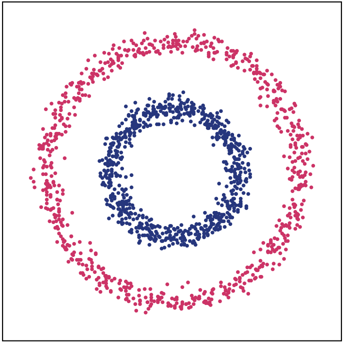
1. The Inner Product Similarity of the RBF Kernel

$$K_{ij} = \exp\left(-\epsilon \| D_{i\cdot} - D_{j\cdot} \|^2\right)$$



2. Apply k-means on the Symmetric Factor Matrix

We apply k-means on the matrix $V\Lambda^{1/2}$ and obtain a perfect clustering for a suitable choice of $\epsilon = 0.3$ as depicted below:



Ok, so in theory we have a method to solve kernel k-means, but in practice this method is not often employed.

Drawbacks of kernel k-means is a lack of robustness and the requirement of a full eigendecomposition.

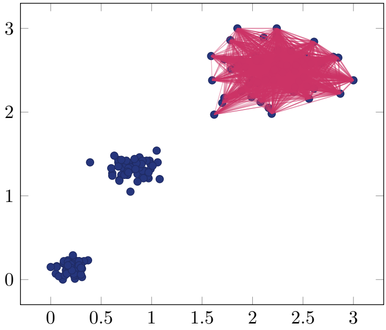
A related method based on a graph representation of the data facilitates nonconvex clustering based on a truncated eigendecomposition.

1

Informal Problem Description

Clustering a Graph Indicated by a Similarity Matrix

Interpretation of the Data as a Graph

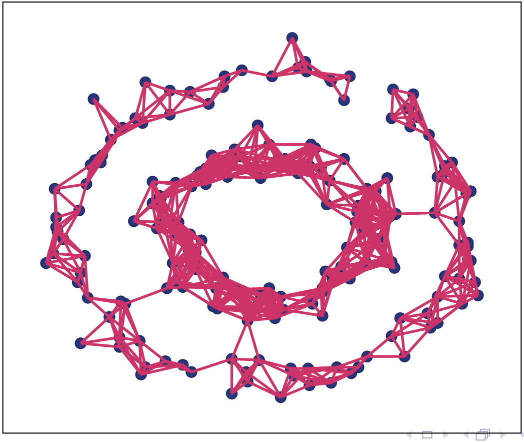


Every data point is a node.

The weight of an edge reflects the similarity between connected nodes.

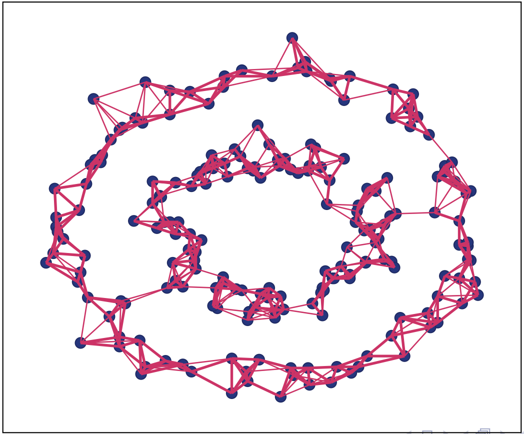
Similarity Measures: Epsilon Neighborhood

$$W_{ij} = \begin{cases} 1 & \text{if } \|D_{i\cdot} - D_{j\cdot}\| < \epsilon \\ 0 & \text{otherwise} \end{cases}$$



Similarity Measures: K-nearest neighbors (K=5)

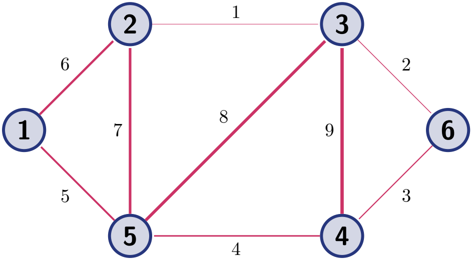
$$N_{ij} = \begin{cases} 1 & \text{if } D_{i\cdot} \in KNN(D_{j\cdot}) \\ 0 & \text{otherwise} \end{cases}, \quad W = \frac{1}{2}(N + N^\top)$$



2

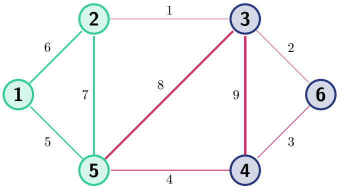
Derive the Formal Problem Definition

The Weighted Adjacency Matrix



$$W = \begin{pmatrix} 0 & 6 & 0 & 0 & 5 & 0 \\ 6 & 0 & 1 & 0 & 7 & 0 \\ 0 & 1 & 0 & 9 & 8 & 2 \\ 0 & 0 & 9 & 0 & 4 & 3 \\ 5 & 7 & 8 & 4 & 0 & 0 \\ 0 & 0 & 2 & 3 & 0 & 0 \end{pmatrix}$$

Computing the Similarity Within a Cluster



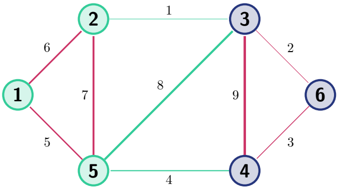
$$Y_{\cdot s}^\top = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}$$

$$Y_{\cdot s}^\top W Y_{\cdot s} = 2(5+6+7)$$

$$Sim(Y;W) = \mathrm{tr}(Y^\top W Y (Y^\top Y)^{-1})$$

$$= \sum_{s=1}^{r} \frac{Y_{\cdot s}^\top W Y_{\cdot s}}{|Y_{\cdot s}|} = \sum_{s=1}^{r} \frac{1}{|\mathcal{C}_s|} \sum_{i,j \in \mathcal{C}_s} W_{ji}$$

Computing the Cut of a Cluster



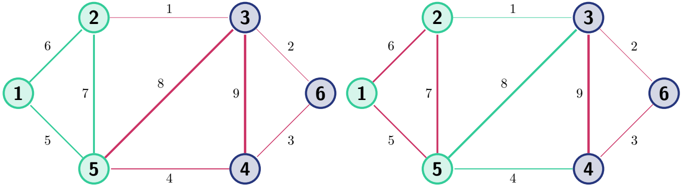
$$Y_{\cdot s}^\top = \begin{pmatrix}1 & 1 & 0 & 0 & 1 & 0\end{pmatrix}$$

$$Y_{\cdot s}^\top W(1 - Y_{\cdot s}) = 1 + 8 + 4$$

$$Cut(Y;W) = \mathrm{tr}((1-Y)^\top WY(Y^\top Y)^{-1})$$

$$= \sum_{s=1}^{r} \frac{(1-Y_{\cdot s})^\top WY_{\cdot s}}{|Y_{\cdot s}|} = \sum_{s=1}^{r} \frac{1}{|\mathcal{C}_s|} \sum_{i \notin \mathcal{C}_s} \sum_{j \in \mathcal{C}_s} W_{ij}$$

Maximum Similarity vs. Minimum Cut



There are principally two ways to define clusters of graphs:

1. maximize the sum of weights within clusters
2. minimize the sum of weights between clusters

Maximum Similarity Graph Clustering

Given: a graph indicated by a symmetric, nonnegative similarity matrix $W \in \mathbb{R}_+^{n \times n}$, and the number of clusters r.

Find: clusters indicated by the matrix $Y \in \mathbb{1}^{n \times r}$ which maximize the similarity of points within a cluster

$$\max_Y Sim(Y; W) = \mathrm{tr}(Y^\top W Y (Y^\top Y)^{-1}) \qquad \text{s.t. } Y \in \mathbb{1}^{n \times r}$$

Minimum Cut Graph Clustering

Given: a graph indicated by a symmetric, nonnegative similarity matrix $W \in \mathbb{R}_+^{n\times n}$, and the number of clusters r.

Find: clusters indicated by the matrix $Y \in \mathbb{1}^{n\times r}$ which minimize the cut of all clusters

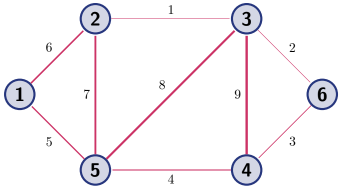
$$\min_{Y} Cut(Y;W) = \operatorname{tr}((1-Y)^\top WY(Y^\top Y)^{-1}) \quad \text{s.t. } Y \in \mathbb{1}^{n\times r}$$

3

Optimization

The Degree Matrix

We have $Y_{\cdot s}^{\top} W Y_{\cdot s} \leq Y_{\cdot s} I_W Y_{\cdot s}$ where I_W is the degree matrix:



$$I_W = \begin{pmatrix} 11 & 0 & 0 & 0 & 0 & 0 \\ 0 & 14 & 0 & 0 & 0 & 0 \\ 0 & 0 & 20 & 0 & 0 & 0 \\ 0 & 0 & 0 & 16 & 0 & 0 \\ 0 & 0 & 0 & 0 & 24 & 0 \\ 0 & 0 & 0 & 0 & 0 & 5 \end{pmatrix}$$

$Y_{\cdot s}^{\top} W Y_{\cdot s} = Y_{\cdot s}^{\top} I_W Y_{\cdot s}$ if and only if $Y_{\cdot s}$ indicates a connected component. This is equivalent to

$$Y_{\cdot s}^{\top} \underbrace{(I_W - W)}_{=L} Y_{\cdot s} = 0$$

The matrix $L = I_W - W$ is called graph Laplacian.

Relation of Minimum Cut and Maximum Similarity

Theorem (Minimum Cut and Maximum Similarity)

Given a symmetric similarity matrix $W \in \mathbb{R}_+^{n \times n}$, *the degree matrix* I_W *and the Graph Laplacian* $L = I_W - W$, *then the following objectives are equivalent:*

$$\min_Y Cut(Y; W) = \operatorname{tr}((1 - Y)^\top WY(Y^\top Y)^{-1}) \quad s.t\ Y \in \mathbb{1}^{n \times r}$$

$$\max_Y Sim(Y; -L) = \operatorname{tr}(Y^\top (-L)Y(Y^\top Y)^{-1}) \quad s.t\ Y \in \mathbb{1}^{n \times r}$$

The maximum similarity objective is equal to the kernel k-means objective. However, note that $-L$ is not a kernel matrix (it's negative semi-definite).

Eigenvalues of Graph Laplacians

Proposition (Connected Components and Eigenvectors)

Given a graph indicated by the symmetric matrix $W \in \mathbb{R}_+^{n\times n}$, then the indicator vectors of the connected components are eigenvectors of the Laplacian $L = I_W - W$ to the smallest eigenvalue 0.

Proof (sketch): For every connected component there exists an order of columns and rows such that W has a block-diagonal form:

$$Wv = \left(\begin{array}{ccc|c} W_{11} & \dots & W_{1c} & \\ \vdots & & \vdots & 0 \\ W_{c1} & \dots & W_{cc} & \\ \hline & 0 & & \widehat{W} \end{array}\right)\left(\begin{array}{c} 1 \\ \vdots \\ 1 \\ \hline 0 \end{array}\right) = \left(\begin{array}{c} \vert W_{1\cdot} \vert \\ \vdots \\ \vert W_{c\cdot} \vert \\ \hline 0 \end{array}\right) = I_W v.$$

The standard method to solve the minimum cut objective is called Spectral Clustering.

The idea of Spectral Clustering is the same as of kernel k-means with few alterations.

Instead of using the full eigendecomposition, Spectral Clustering uses only the first r meaningful eigenvectors which are not indicating the connected component.

The Spectral Clustering Algorithm

Requirement

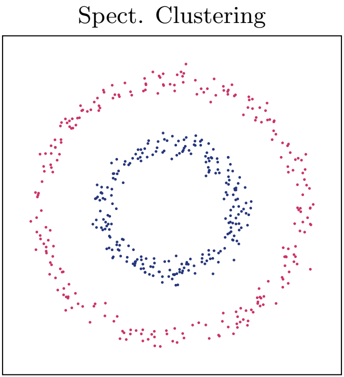
The parameters of the similarity measure should be chosen such that the graph is connected!

```
function SpectralClustering(r, D, Sim)
    W ← sim(D)                                   ▷ Compute Similarity matrix
    L ← I_W − W                                  ▷ Compute Graph Laplacian
    (V, Λ) ← TruncatedEigendecomposition(L, r + 1)
    A ← V_{·{2,...,r+1}}                          ▷ Remove connected component
    (X, Y) ← kMeans(A,r)
    return Y
end function
```

Spectral Clustering with 10NN Similarity Matrix and L_{sym}

In practice, the weighted adjacency matrix is often normalized.
The corresponding Graph Laplacian is often denoted by

$$L_{sym} = I - I_W^{-1/2} W I_W^{-1/2}$$



The Most Important Slide of this Lecture

Theorem (Equivalent k-means objectives)

The following objectives are equivalent

$$\min_{Y} \|D - YX^\top\|^2 \qquad s.t.\ X = D^\top Y(Y^\top Y)^{-1}, Y \in \mathbb{1}^{n\times r}$$

$$\min_{Y,X} \|D - YX^\top\|^2 \qquad s.t.\ X \in \mathbb{R}^{d\times r}, Y \in \mathbb{1}^{n\times r}$$

$$\max_{Y} \operatorname{tr}(Z^\top DD^\top Z) \qquad s.t.\ Z = Y(Y^\top Y)^{-1/2}, Y \in \mathbb{1}^{n\times r}$$