Machine Learning I: Fractal 2

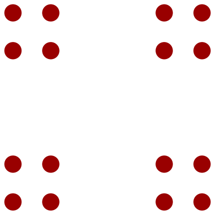
Rajendra Nagar

Assistant Professor
Department of Electircal Engineering
Indian Institute of Technology Jodhpur

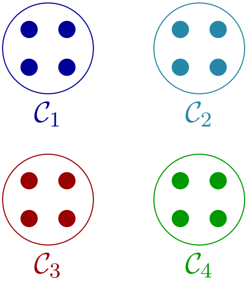
These slides are prepared from the following book:
Shalev-Shwartz, Shai, and Shai Ben-David. Understanding machine learning: From theory to algorithms. Cambridge university press, 2014.

Clustering

Input: A set of elements, $\mathcal{X}$, and a distance function to measure similarity.
Objective: A partition of the input domain $\mathcal{X}$ into groups $\{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_k\}$ of similar elements such that $\cup_{i=1}^{k} \mathcal{C}_i = \mathcal{X}$, and $\mathcal{C}_i \cap \mathcal{C}_j = \emptyset, \forall i \neq j$.

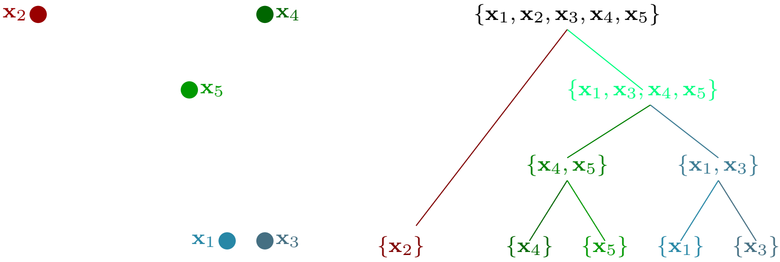


$\mathcal{X}$



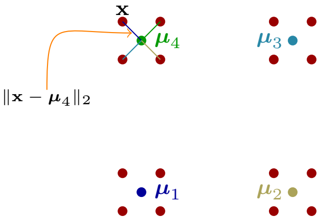
$\mathcal{X} = \mathcal{C}_1 \cup \mathcal{C}_2 \cup \mathcal{C}_3 \cup \mathcal{C}_4$

Linkage-Based Clustering Algorithm



k-Means [1,2]

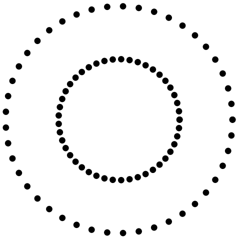
Let $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ be the set data-points, where $\mathbf{x}_i \in \mathbb{R}^d$. We want to partition $\mathcal{X}$ in groups $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_k$ containing similar objects. Let $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_k$ be their respective group representatives (centres), where $\boldsymbol{\mu}_i \in \mathbb{R}^d$.



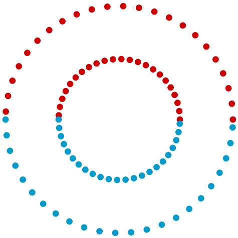
- $f(\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k) = \sum_{i=1}^{k} \sum_{\mathbf{x} \in \mathcal{C}_i} \|\mathbf{x} - \boldsymbol{\mu}_i\|_2^2$
- $\underset{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k}{\arg\min}\ f(\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k)$.
- $\frac{\partial f}{\partial \boldsymbol{\mu}_i} = \mathbf{0} \Rightarrow \boldsymbol{\mu}_i = \frac{\sum_{\mathbf{x} \in \mathcal{C}_i} \mathbf{x}}{|\mathcal{C}_i|}$
- $\boldsymbol{\mu}_i = \text{mean}(\mathcal{C}_i)\ \forall i \in \{1, 2, \ldots, k\}$.

[1] MacQueen, James. "Some methods for classification and analysis of multivariate observations." Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, 1967.

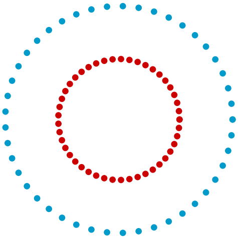
[2] Lloyd, Stuart P. "Least squares quantization in PCM." IEEE Transactions on Information Theory, 1982.



$\mathcal{X}$



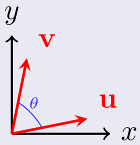
k-means



Desired

Orthogonal Vectors

Two unit norm vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ are called orthogonal vectors if $\langle \mathbf{u}, \mathbf{v} \rangle = \mathbf{u}^\top \mathbf{v} = \cos(\theta) = 0$.



Orthonormal Matrix

A matrix $\mathbf{A} = \begin{bmatrix} \mathbf{a}_1 & \mathbf{a}_2 & \cdots & \mathbf{a}_n \end{bmatrix}$ of size $n \times n$ is called an orthonormal matrix if $\langle \mathbf{a}_i, \mathbf{a}_j \rangle = 1$ if $i = j$ and $\langle \mathbf{a}_i, \mathbf{a}_j \rangle = 0$ if $i \neq j$. If $\mathbf{A}$ is an orthonormal matrix, then $\mathbf{A}^\top \mathbf{A} = \mathbf{I}$.

Spectral Theorem

Let $\mathbf{A} \in \mathbb{R}^{n \times n}$ be a symmetric matrix, i.e. $\mathbf{A}^\top = \mathbf{A}$. Then, $\mathbf{A}$ has exactly n orthonormal eigenvectors $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ with corresponding eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$. $\mathbf{A}\mathbf{x}_i = \lambda_i \mathbf{x}_i, \forall i \in [n]$.

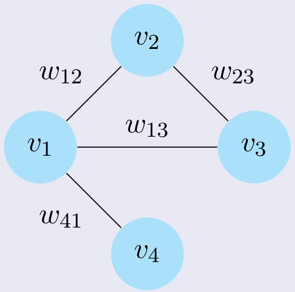
Trace of a Matrix

The trace of a matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ is defined as: Trace$(\mathbf{A}) = \sum_{i=1}^{n} a_{ii}$.

Let $\mathbf{B} \in \mathbf{R}^{n \times n}$ be a matrix, then Trace$(\mathbf{B}^\top \mathbf{A} \mathbf{B}) = \sum_{i=1}^{n} \mathbf{b}_i^\top \mathbf{A} \mathbf{b}_i$.

Graph

A graph is a pair $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is the set of vertices and $\mathcal{E}$ is the set of edges between the vertices.



$$\mathcal{V} = \{v_1, v_2, v_3, v_4\}$$

$$\mathcal{E} = \{(v_1, v_2), (v_2, v_3), (v_4, v_1), (v_1, v_3)\}$$

Adjacency Matrix

$$\mathbf{W} = \begin{bmatrix} 0 & w_{12} & w_{13} & w_{14} \\ w_{12} & 0 & w_{23} & 0 \\ w_{13} & w_{23} & 0 & 0 \\ w_{14} & 0 & 0 & 0 \end{bmatrix}$$

Degree Matrix

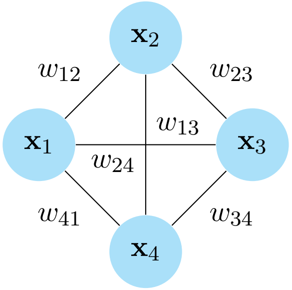
$$\mathbf{D} = \begin{bmatrix} d_1 & 0 & 0 & 0 \\ 0 & d_2 & 0 & 0 \\ 0 & 0 & d_3 & 0 \\ 0 & 0 & 0 & d_4 \end{bmatrix}, d_i = \sum_{j=1}^{4} w_{ij}$$

Laplacian Matrix

$$\begin{aligned} \mathbf{L} &= \mathbf{D} - \mathbf{W} \\ &= \begin{bmatrix} d_1 & -w_{12} & -w_{13} & -w_{14} \\ -w_{12} & d_2 & -w_{23} & 0 \\ -w_{13} & -w_{23} & d_3 & 0 \\ -w_{14} & 0 & 0 & d_4 \end{bmatrix} \end{aligned}$$

Spectral Clustering

- Represent the relationships between points in a data set $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ by a similarity graph.
- A vertex represents a data point, and every two vertices are connected by an edge whose weight is their similarity $\mathbf{W}_{i,j} = s(\mathbf{x}_i, \mathbf{x}_j)$.

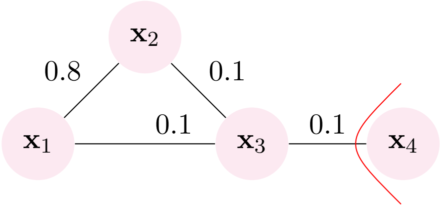


- For example, we can set $\mathbf{W}_{i,j} = e^{-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{\sigma^2}}$, where σ is a parameter.
- Partition of the graph such that the edges between different groups have low weights and the edges within a group have high weights.

- Given a graph with adjacency matrix $\mathbf{W}$, the simplest way partition of the graph is to solve the *mincut* problem, which chooses a partition $\mathcal{C}_1, \ldots, \mathcal{C}_k$ that minimizes the objective

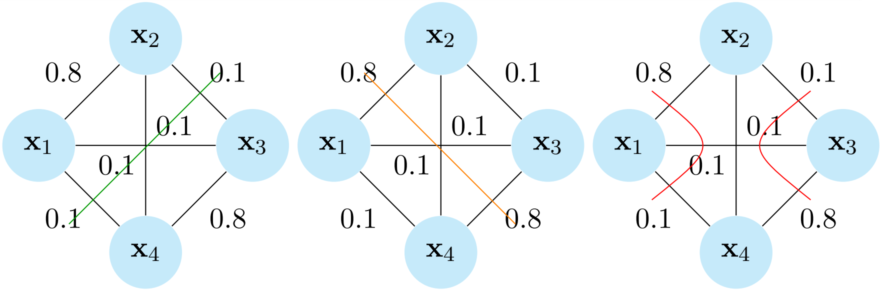
$$\text{Cut}(\mathcal{C}_1, \ldots, \mathcal{C}_k) = \sum_{i=1}^{k} \sum_{r \in \mathcal{C}_i} \sum_{s \notin \mathcal{C}_i} \mathbf{W}_{r,s}.$$

- The problem is that in many cases, the solution of mincut simply separates one individual vertex from the rest of the graph.



- A simple solution is to normalize the cut and define the normalized *mincut* objective as follows

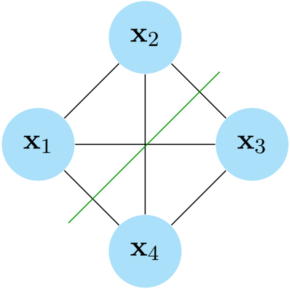
$$\text{RatioCut}(\mathcal{C}_1, \dots, \mathcal{C}_k) = \sum_{i=1}^{k} \frac{1}{|\mathcal{C}_i|} \sum_{r \in \mathcal{C}_i} \sum_{s \notin \mathcal{C}_i} \mathbf{W}_{r,s}.$$



$\text{RatioCut}(\mathcal{C}_1, \mathcal{C}_2) = 0.2$ $\quad$ $\text{RatioCut}(\mathcal{C}_1, \mathcal{C}_2) = 0.9$ $\quad$ $\text{RatioCut}(\mathcal{C}_1, \mathcal{C}_2) = 1.0$

$$\min_{\mathcal{C}_1, \dots, \mathcal{C}_k} \text{RatioCut}(\mathcal{C}_1, \dots, \mathcal{C}_k)$$

Consider the Graph Laplacian matrix $\mathbf{L}$ of the graph constructed on $\mathcal{X}$.



Cluster Assignment Matrix

Let $\mathcal{C}_1, \ldots, \mathcal{C}_k$ be the clustering and $\mathbf{H} \in \mathbb{R}^{n \times k}$ be a matrix such that

$$\mathbf{H}_{i,j} = \frac{1}{\sqrt{|\mathcal{C}_j|}} \mathbb{1}_{[i \in \mathcal{C}_j]}.$$

$$\mathbf{H} = \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & 0 \\ 0 & \frac{1}{\sqrt{2}} \\ 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \in \mathbb{R}^{n \times k}. \text{ Here, } k = 2.$$

Claim

The columns of the matrix $\mathbf{H}$ are orthonormal to each other and

$$\text{RatioCut}(\mathcal{C}_1, \ldots, \mathcal{C}_k) = \text{trace}(\mathbf{H}^\top \mathbf{L} \mathbf{H}).$$

Proof

Let $\mathbf{h}_1, \ldots, \mathbf{h}_k$ be the columns of the matrix $\mathbf{H}$. Then, it is easy to observe that $\text{trace}(\mathbf{H}^\top \mathbf{L}\mathbf{H}) = \sum_{i=1}^{k} \mathbf{h}_i^\top \mathbf{L}\mathbf{h}_i$ and for any vector $\mathbf{v}$ we have

$$
\begin{aligned}
\mathbf{v}^\top \mathbf{L}\mathbf{v} &= \frac{1}{2}\left(\sum_r \mathbf{D}_{r,r} v_e^2 - 2\sum_r \sum_s v_r v_s \mathbf{W}_{r,s} + \sum_s \mathbf{D}_{s,s} v_s^2\right) \\
&= \frac{1}{2}\sum_r \sum_s \mathbf{W}_{r,s}(v_r - v_s)^2.
\end{aligned}
$$

Now, let $\mathbf{v} = \mathbf{h}_i$ and noting that $(h_{ir} - h_{is})^2$ is non zero only if $r \in \mathcal{C}_i$, $s \notin \mathcal{C}_s$, we have that

$$
\mathbf{h}_i^\top \mathbf{L}\mathbf{h}_i = \frac{1}{|\mathcal{C}_i|}\sum_{r\in\mathcal{C}_i, s\notin\mathcal{C}_i} \mathbf{W}_{r,s} \Rightarrow \sum_{i=1}^{k} \mathbf{h}_i^\top \mathbf{L}\mathbf{h}_i = \sum_{i=1}^{k} \frac{1}{|\mathcal{C}_i|}\sum_{r\in\mathcal{C}_i}\sum_{s\notin\mathcal{C}_i} \mathbf{W}_{r,s}.
$$

$$
\Rightarrow \text{trace}(\mathbf{H}^\top \mathbf{L}\mathbf{H}) = \text{RatioCut}(\mathcal{C}_1, \ldots, \mathcal{C}_k).
$$

Spectral Clustering

Problem Formulation

$$\min_{\mathcal{C}_1,\ldots,\mathcal{C}_k} \text{RatioCut}(\mathcal{C}_1,\ldots,\mathcal{C}_k) \Leftrightarrow \min_{\mathbf{H}\in\mathbb{R}^{n\times k},\mathbf{H}^\top\mathbf{H}=\mathbf{I}} \text{trace}(\mathbf{H}^\top\mathbf{L}\mathbf{H}).$$

Rayleigh quotient

$$\mathbf{v}^\star = \arg\min_{\mathbf{v}\in\mathbb{R}^n,\mathbf{v}^\top\mathbf{v}=1} \mathbf{v}^\top\mathbf{L}\mathbf{v}$$

$$\begin{aligned} f(\mathbf{v}) &= \mathbf{v}^\top\mathbf{L}\mathbf{v} + \lambda(1-\mathbf{v}^\top\mathbf{v}) \\ \nabla f &= 2\mathbf{L}\mathbf{v} - 2\lambda\mathbf{v} \\ \mathbf{L}\mathbf{v} &= \lambda\mathbf{v} \\ \mathbf{v}^\top\mathbf{L}\mathbf{v} &= \lambda. \end{aligned}$$

Therefore, we have to minimize λ such that $\mathbf{L}\mathbf{v} = \lambda_1\mathbf{v}$. Hence, $\mathbf{v}^\star =$ eigenvector of the matrix $\mathbf{L}$ corresponding to the smallest eigenvalue $= \mathbf{u}_1$.

Rayleigh quotient

$$\mathbf{v}^\star = \underset{\mathbf{v}^\top \mathbf{u}_1 = 0, \mathbf{v}^\top \mathbf{v} = 1}{\arg\min} \mathbf{v}^\top \mathbf{L}\mathbf{v}$$

Solution

$$\begin{aligned} f(\mathbf{v}) &= \mathbf{v}^\top \mathbf{L}\mathbf{v} + \lambda(1 - \mathbf{v}^\top \mathbf{v}) \\ \nabla f &= 2\mathbf{L}\mathbf{v} - 2\lambda\mathbf{v} \\ \mathbf{L}\mathbf{v} &= \lambda\mathbf{v} \\ \mathbf{v}^\top \mathbf{L}\mathbf{v} &= \lambda \end{aligned}$$

Therefore, we have to minimize λ such that $\mathbf{L}\mathbf{v} = \lambda\mathbf{v}$ and $\mathbf{v}^\top \mathbf{u}_1 = 0$. $\mathbf{v}^\star =$ eigenvector of the matrix $\mathbf{L}$ corresponding to the second smallest eigenvalue $= \mathbf{u}_2$.

Rayleigh quotient

$$\mathbf{v}^\star = \underset{\mathbf{v}^\top \mathbf{u}_i = 0, \forall i < k, \mathbf{v}^\top \mathbf{v} = 1}{\arg\min} \mathbf{v}^\top \mathbf{L}\mathbf{v}$$

Solution

$$\begin{aligned} f(\mathbf{v}) &= \mathbf{v}^\top \mathbf{L}\mathbf{v} + \lambda(1 - \mathbf{v}^\top \mathbf{v}) \\ \nabla f &= 2\mathbf{L}\mathbf{v} - 2\lambda\mathbf{v} \\ \mathbf{L}\mathbf{v} &= \lambda\mathbf{v} \\ \mathbf{v}^\top \mathbf{L}\mathbf{v} &= \lambda \end{aligned}$$

We have to minimize λ such that $\mathbf{L}\mathbf{v} = \lambda\mathbf{v}$ and $\mathbf{v}^\top \mathbf{u}_i = 0, \forall i < k$. $\mathbf{v}^\star =$ eigenvector of the matrix $\mathbf{L}$ corresponding to the k^{th} smallest eigenvalue $= \mathbf{u}_k$.

Rayleigh quotient

$$\underset{\substack{\mathbf{v}_1,\ldots,\mathbf{v}_k \\ \mathbf{v}_i^\top \mathbf{v}_j = \delta_{ij}}}{\arg\min} \sum_{i=1}^{k} \mathbf{v}_i^\top \mathbf{L} \mathbf{v}_i$$

Here, $\delta_{ij} = 1$, if $i = j$ and $\delta_{ij} = 0$, if $i \neq j$.

Solution

$$\begin{aligned}
f(\mathbf{v}_1, \ldots, \mathbf{v}_k) &= \sum_{i=1}^{k} \mathbf{v}_i^\top \mathbf{L} \mathbf{v}_i + \sum_{i=1}^{k} \lambda_i (1 - \mathbf{v}_i^\top \mathbf{v}_i) \\
\nabla_{\mathbf{v}_i} f &= 2\mathbf{L}\mathbf{v}_i - 2\lambda \mathbf{v}_i \\
\mathbf{L}\mathbf{v}_i &= \lambda_i \mathbf{v}_i \\
\mathbf{v}_i^\top \mathbf{L} \mathbf{v}_i &= \lambda_i
\end{aligned}$$

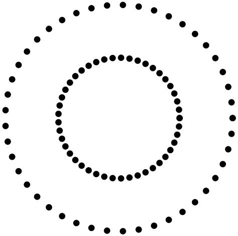
Therefore, we have to minimize $\sum_{i=1}^{k} \lambda_i$ such that $\mathbf{L}\mathbf{v}_i = \lambda \mathbf{v}_i$ and $\mathbf{v}_i^\top \mathbf{v}_j = 0$ if $i \neq j$. Hence, $\mathbf{v}_i^\star =$ eigenvector of the matrix $\mathbf{L}$ corresponding to the i^{th} smallest eigenvalue $= \mathbf{v}_i$.

Problem

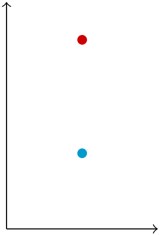
$$\mathbf{H}^\star = \underset{\mathbf{H}\in\mathbb{R}^{n\times k},\mathbf{H}^\top\mathbf{H}=\mathbf{I}}{\arg\min} \ \mathrm{trace}(\mathbf{H}^\top\mathbf{L}\mathbf{H}).$$

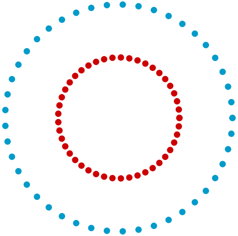
Solution

Let $\mathbf{L}\mathbf{u}_i = \lambda_i\mathbf{u}_i, \forall i \in \{1, 2, \ldots, n\}$ be the EVD of the matrix $\mathbf{L}$. Here, we assume that the eigenvalues are such that $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$. Then, the solution to the above problem is $\mathbf{H}^\star = \begin{bmatrix}\mathbf{u}_1 & \mathbf{u}_2 & \cdots & \mathbf{u}_k\end{bmatrix}$.



$$\mathbf{H} = \begin{bmatrix} 1 & 0.5 \\ 1 & 0.5 \\ \vdots & \vdots \\ 1 & 0.5 \\ 1 & -0.5 \\ 1 & -0.5 \\ \vdots & \vdots \\ 1 & -0.5 \end{bmatrix}$$





Algorithm 1 Spectral Clustering

1: **Input:** $\mathbf{W} \in \mathbb{R}^{n \times n}$, Number of clusters k.
2: **Initialize:** Compute the graph Laplacian $\mathbf{L}$.
3: $\mathbf{H} \leftarrow$ matrix whose columns are the eigenvectors of $\mathbf{L}$ corresponding to the k-smallest eigenvalues.
4: $\mathbf{r}_1, \ldots, \mathbf{r}_n$ be the rows of $\mathbf{H}$.
5: Cluster the points $\mathbf{r}_1, \ldots, \mathbf{r}_n$ using k-means algorithm.
6: **Output:** Clusters $\mathcal{C}_1, \ldots, \mathcal{C}_k$ of the k-means algorithm.