16. Spectral Clustering

Spectral Clustering: Motivation

Motivation

Clustering is the process of organizing objects into different groups, where members of a group are similar in some way and members from different groups are dissimilar.

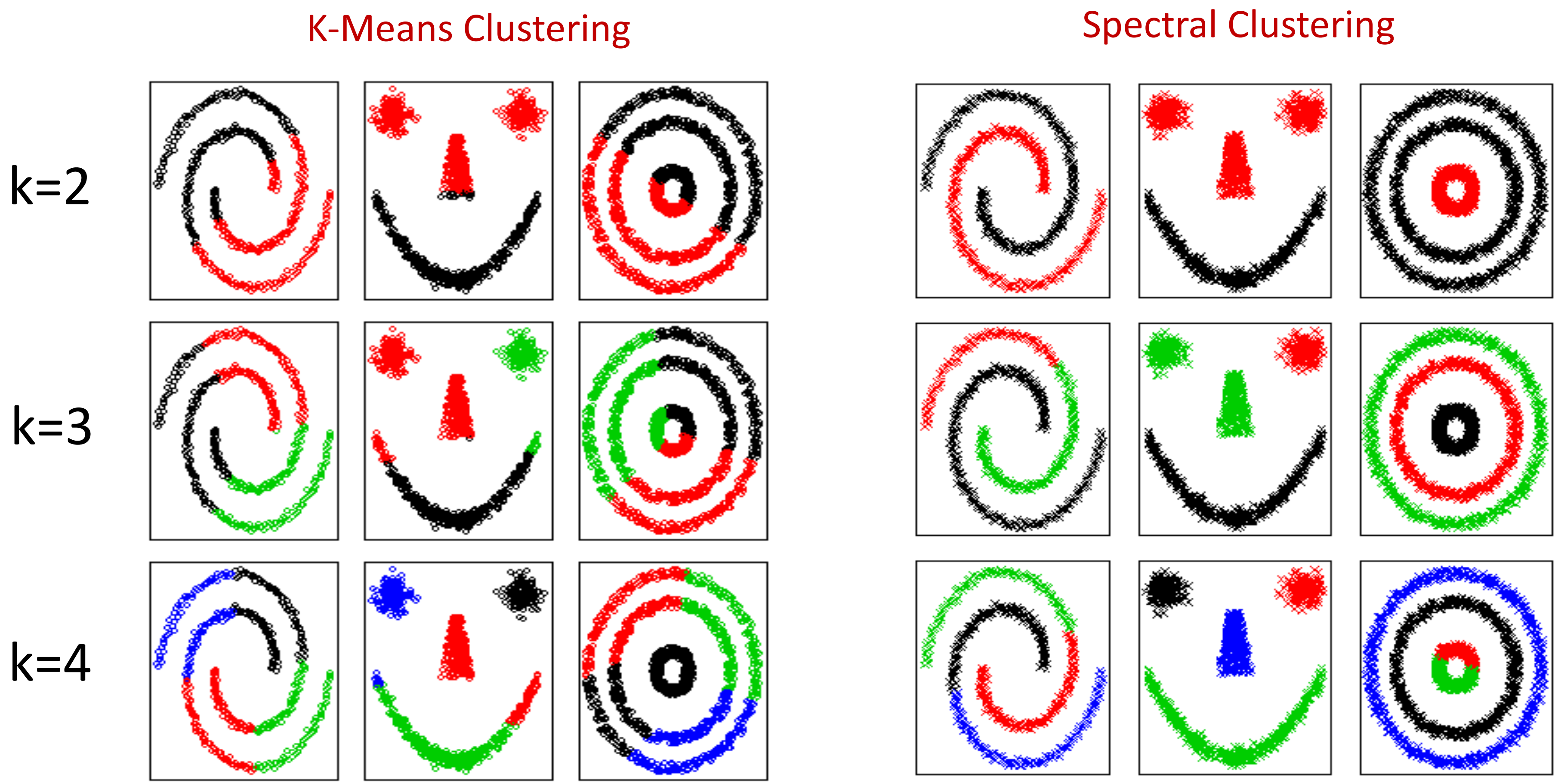
Spectral clustering is such that, given an underlying graph, one partitions the vertices such that those connected by edges with high weights are grouped together but separated from the ones connected by low weights.

Spectral clustering is motivated by connectivity whereas k-means clustering is motivated by compactness [it is ideal for discovering global clusters where all members of each cluster are in close proximity to each other (in the Euclidean sense)].

Spectral clustering is useful for hard non-convex clustering problems whereas *k*-means clustering is able to recover only convex clusters (it divides the space in *k* clusters with the property that the line connecting any two points belonging to the same cluster does not intersect any other cluster).

Spectral Clustering: Motivation

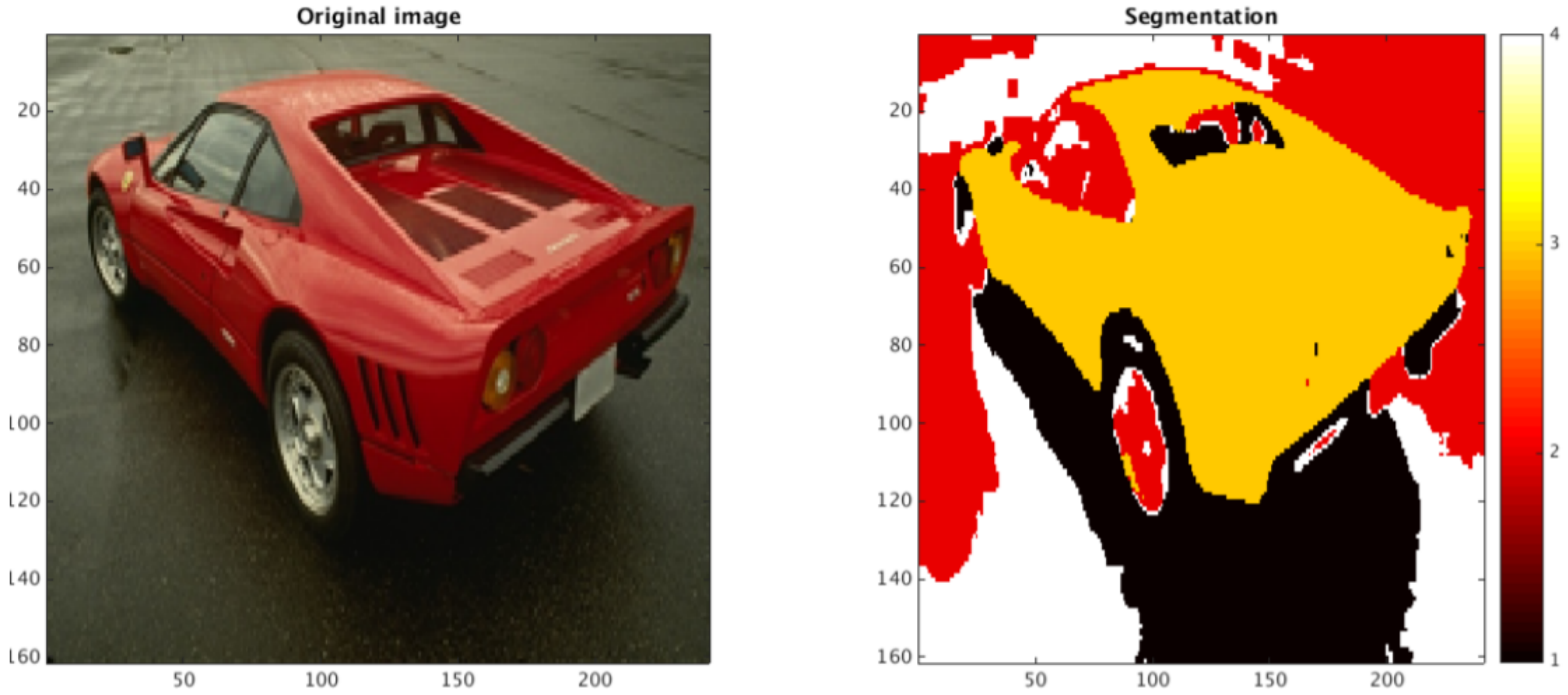
Spectral Clustering vs. K-Means Clustering



https://www.machinelearningtutorial.net/2017/02/27/the-5-most-common-clustering-methods-with-code-in-r/

Spectral Clustering: Applications

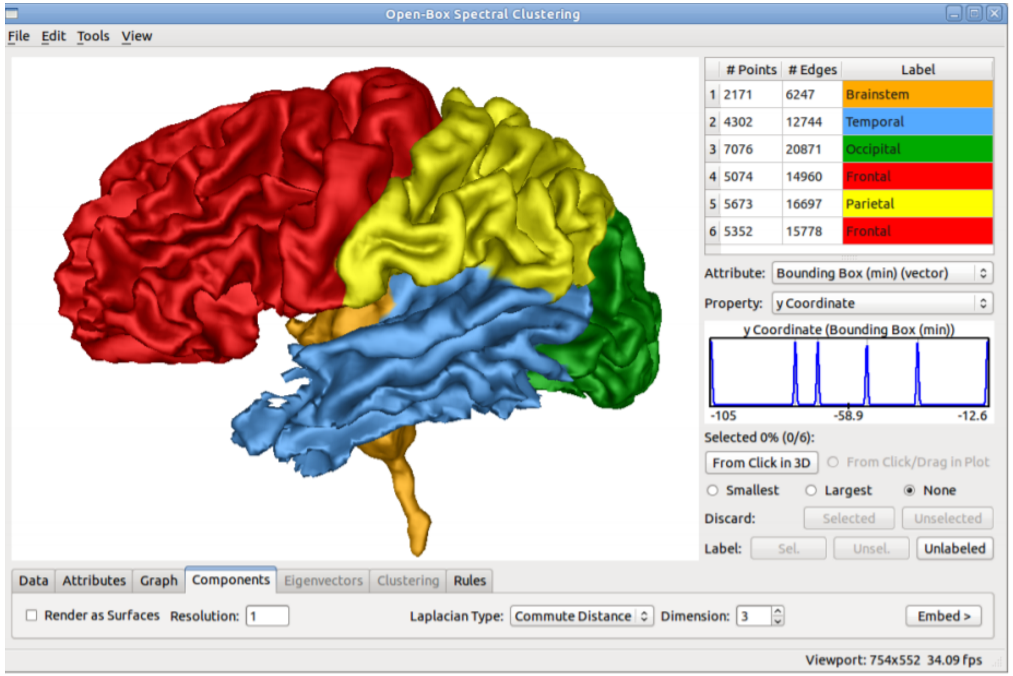
Applications: Image Segmentation

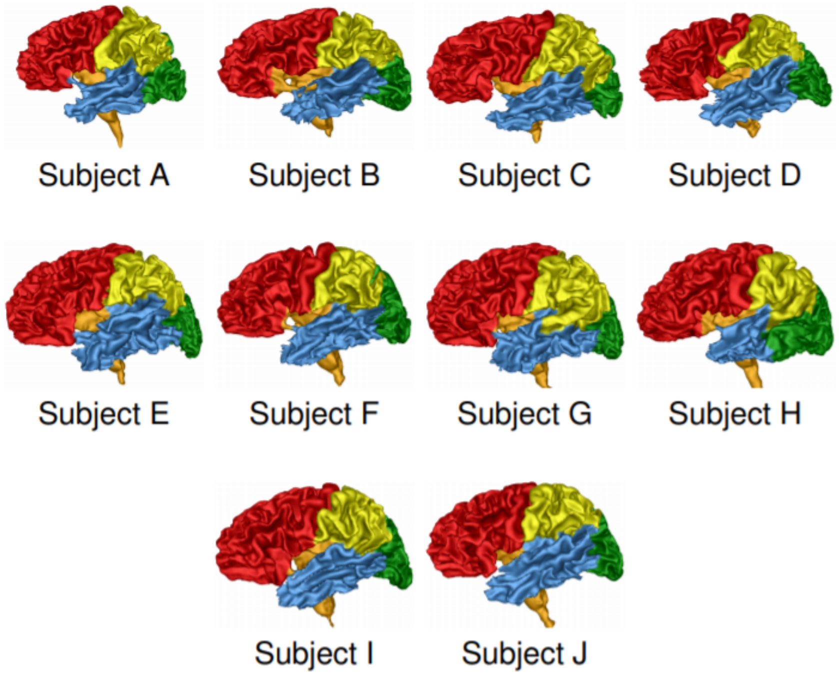


https://arxiv.org/pdf/1505.00477.pdf

Spectral Clustering: Applications

Applications: Image Segmentation





Developing segmentation protocols in chest CT and brain MRI that can then be successfully applied to other datasets in an automated manner

https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6634089

Spectral Clustering: Warm-Up

v_1

Graphs: Basic Definitions

Graph:

A graph G is a collection of nodes $V = \{v_1, v_2, \dots v_m\}$, called vertices, together with a set of lines $E = \{e_1, e_2, \dots e_n\}$, called edges, connecting certain pairs of vertices. (V cannot be empty).

Connected Graph:

A connected graph is a graph where it is possible to get from every vertex in the graph to every other vertex through a series of edges, called a path.

Edge Cut:

An edge cut is the set of edges whose removal makes a graph disconnected.

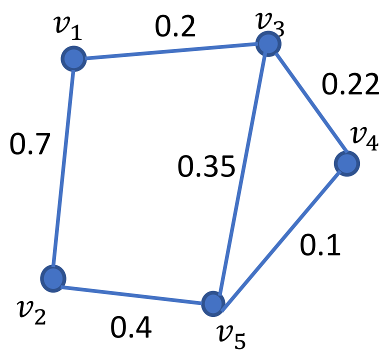
Spectral Clustering: Warm-Up

Graphs: Matrix Representations – Similarity and Adjacency Matrices

A graph with m vertices can be represented by an $m \times m$ similarity matrix W whose entry W_{ij} represents a pairwise affinity between nodes v_i and v_j that is positive if there is an edge between v_i and v_j and 0 if there is no edge (symmetric) .

The graph similarity matrix is known as the graph adjacency matrix A when the presence/absence of edges between a pair or nodes is represented by 1/0, respectively.

Example



$$W = \begin{bmatrix} 0 & 0.7 & 0.2 & 0 & 0 \\ 0.7 & 0 & 0 & 0 & 0.4 \\ 0.2 & 0 & 0 & 0.22 & 0.35 \\ 0 & 0 & 0.22 & 0 & 0.1 \\ 0 & 0.4 & 0.35 & 0.1 & 0 \end{bmatrix}$$

$$A = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix}$$

Spectral Clustering: Warm-Up

v_1

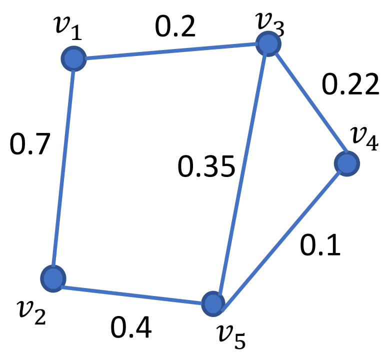
Graphs: Matrix Representations – Degree Matrix

The degree of a vertex v_i, denoted as $d(v_i)$, corresponds to the number of edges connected to it.

The degree of a vertex v_i can be determined by summing over the i-th column or row of the similarity matrix.

The degree matrix D contains the degree of each vertices along its diagonal and zeros everywhere else.

Example



$$D = \begin{bmatrix} 0.9 & 0 & 0 & 0 & 0 \\ 0 & 1.1 & 0 & 0 & 0 \\ 0 & 0 & 0.78 & 0 & 0 \\ 0 & 0 & 0 & 0.32 & 0 \\ 0 & 0 & 0 & 0 & 0.85 \end{bmatrix}$$

v_1

Spectral Clustering: Warm-Up

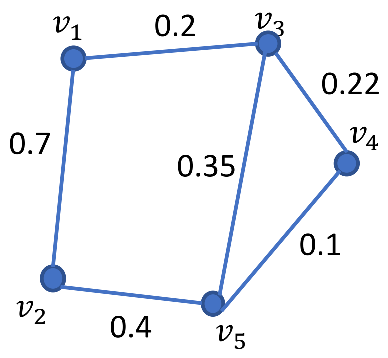
Graphs: Matrix Representations – Laplacian Matrix

The un-normalized graph Laplacian matrix L corresponds to the difference between the degree matrix D and the similarity matrix W of the graph: $L = D - W$

It is a positive semi-definite matrix i.e. the eigenvalues of L are all non-negative and the eigenvectors are all orthogonal.

The eigenvectors of the Laplacian matrix provide an embedding of the data based on similarity.

Example



$L =$

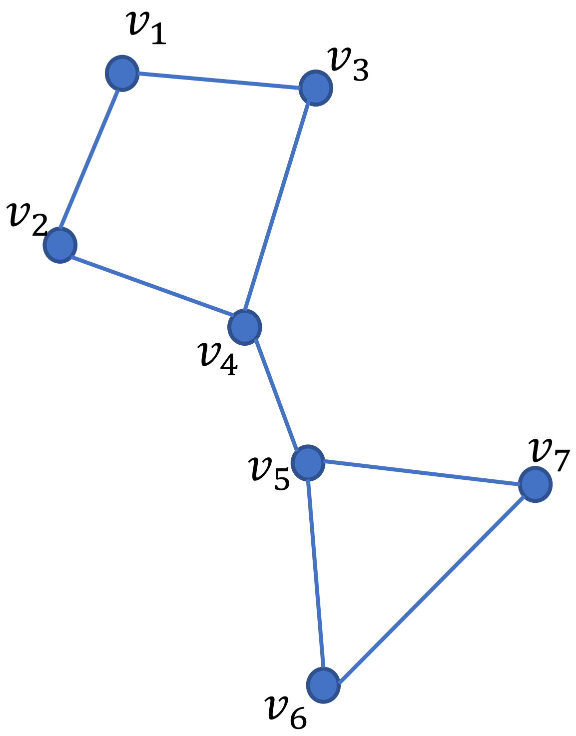
0.9	-0.7	-0.2	0	0
-0.7	1.1	0	0	-0.4
-0.2	0	0.78	-0.22	-0.35
0	0	-0.22	0.32	-0.1
0	-0.4	-0.35	-0.1	0.85

Spectral Clustering: Warm-Up

v_1

Exercise:

Determine the degree and adjacency matrices of the graph:



Spectral Clustering: Procedure

Graph Construction

A set of data points $\mathcal{X} = \{x_1, x_2, \dots . x_m\}$, can be modelled using a graph structure where vertices are given by data points and edges are given by constructing a similarity matrix according to various possible approaches:

ϵ-Neighbourhood: The vertices x_i and x_j are connected by an edge with weight given by

$$W_{ij} = \begin{cases} 1, & \|x_i - x_j\| < \epsilon \\ 0, & \text{otherwise} \end{cases}$$

Gaussian Kernel: The vertices x_i and x_j are connected by an edge with weight given by a Gaussian kernel i.e. $W_{ij} = \exp\left(-\|x_i - x_j\|^2/\sigma^2\right)$

The clustering problem reduces to find a partition of the graph such that edges within a group have high weights and edges between different groups have low weight.

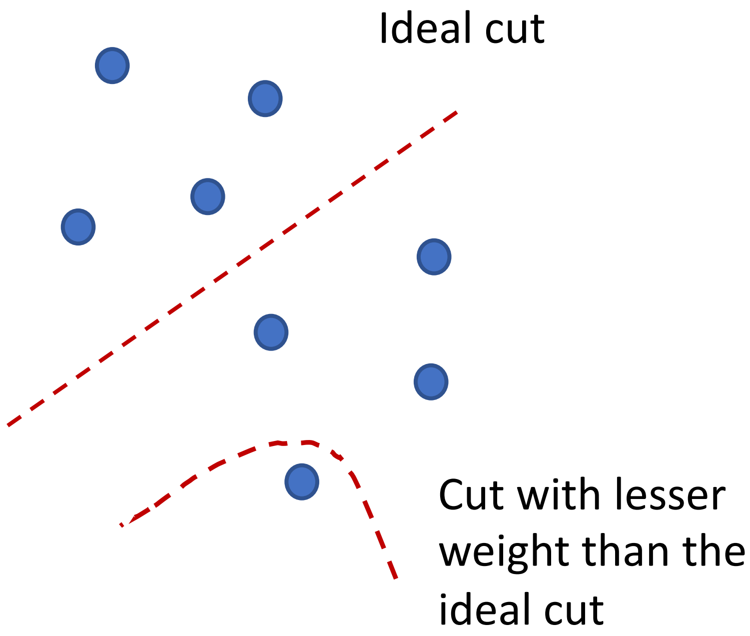
Spectral Clustering: Procedure

Graph Partition

The simplest method to construct a partition of a graph involves solving the mincut problem.

The mincut problem – which yields a partition $C_1, \ldots, C_k$ such that weight of edges connecting vertices in one partition to the vertices in the other partitions is minimum – aims to minimize the objective given by:

$$cut(C_1, \ldots C_k) = \sum_{i=1}^{k} \sum_{r \in C_i, s \notin C_i} W_{rs}$$



This problem can be solved efficiently for two partitions but it does not often lead in practice to satisfactory partitions: it can separate one individual vertex from the rest of the graph.

Spectral Clustering: Procedure

v_1

Spectral Clustering: Graph Partition

Other popular methods to construct a partition of the graph involve solving the ratio-cut problem

The ratio-cut problem addresses the limitations of the mincut problem by aiming to minimize the objective given by:

$$ratiocut(C_1, .. C_k) = \sum_{i=1}^{k} \frac{1}{|C_i|} \sum_{r \in C_i, s \notin C_i} W_{rs}$$

This results in a computationally very hard optimization problem that cannot be solved efficiently.

Spectral clustering is a way to relax the ratio-cut optimization problem. It leverages the fact that

$$ratiocut(C_1, .. C_k) = trace(H^t L H)$$

where H is a matrix with entries $H_{i,j} = 1/\sqrt{|C_j|} \times \mathbb{I}(i \in C_j)$

v_1

Spectral Clustering: Algorithm

Algorithm: Unnormalized Spectral Clustering

(this is a relaxation of the ncut algorithm)

Algorithm Input:

Degree matrix $D \in R^{m\times m}$; Similarity matrix $W \in R^{m\times m}$
Number of clusters k

Procedure:

(1) Initialize: Compute the unnormalized graph Laplacian L

(2) Let $U \in R^{m\times k}$ be the matrix whose columns are the eigenvectors of L corresponding to the k smallest eigenvalues of L. **Let** $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m$ be the rows of U

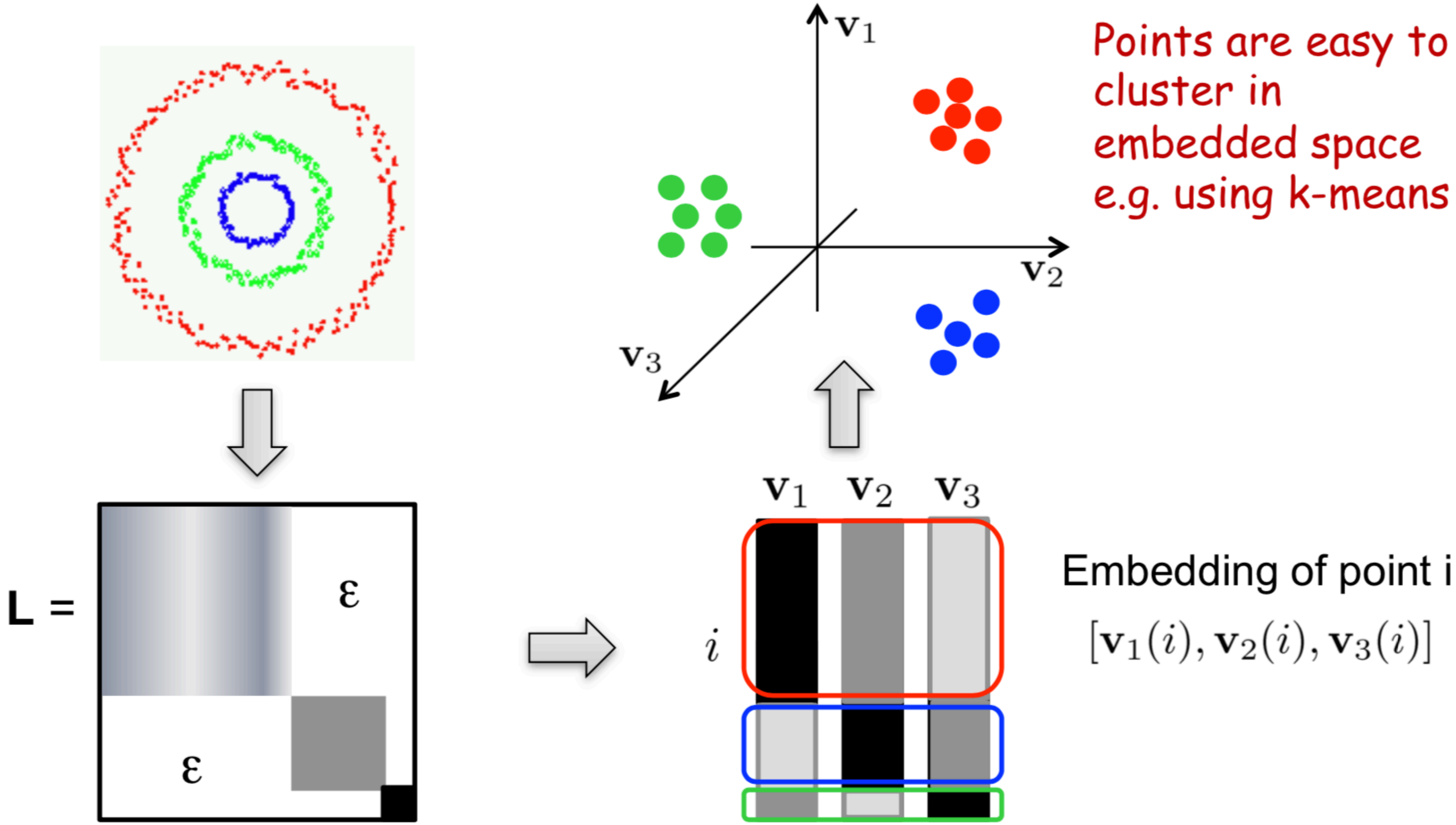
(3) Cluster the points $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m$ using k-means clustering

Algorithm Output:

Clusters $C_1, \ldots, C_K$ of the k-means algorithm

Spectral Clustering: Intuition

Algorithm: Unnormalized Spectral Clustering



Source: Aarti Singh, CMU

Spectral Clustering: Challenges

v_1

Challenges

Choice of number of clusters k.
Choice of similarity metric.