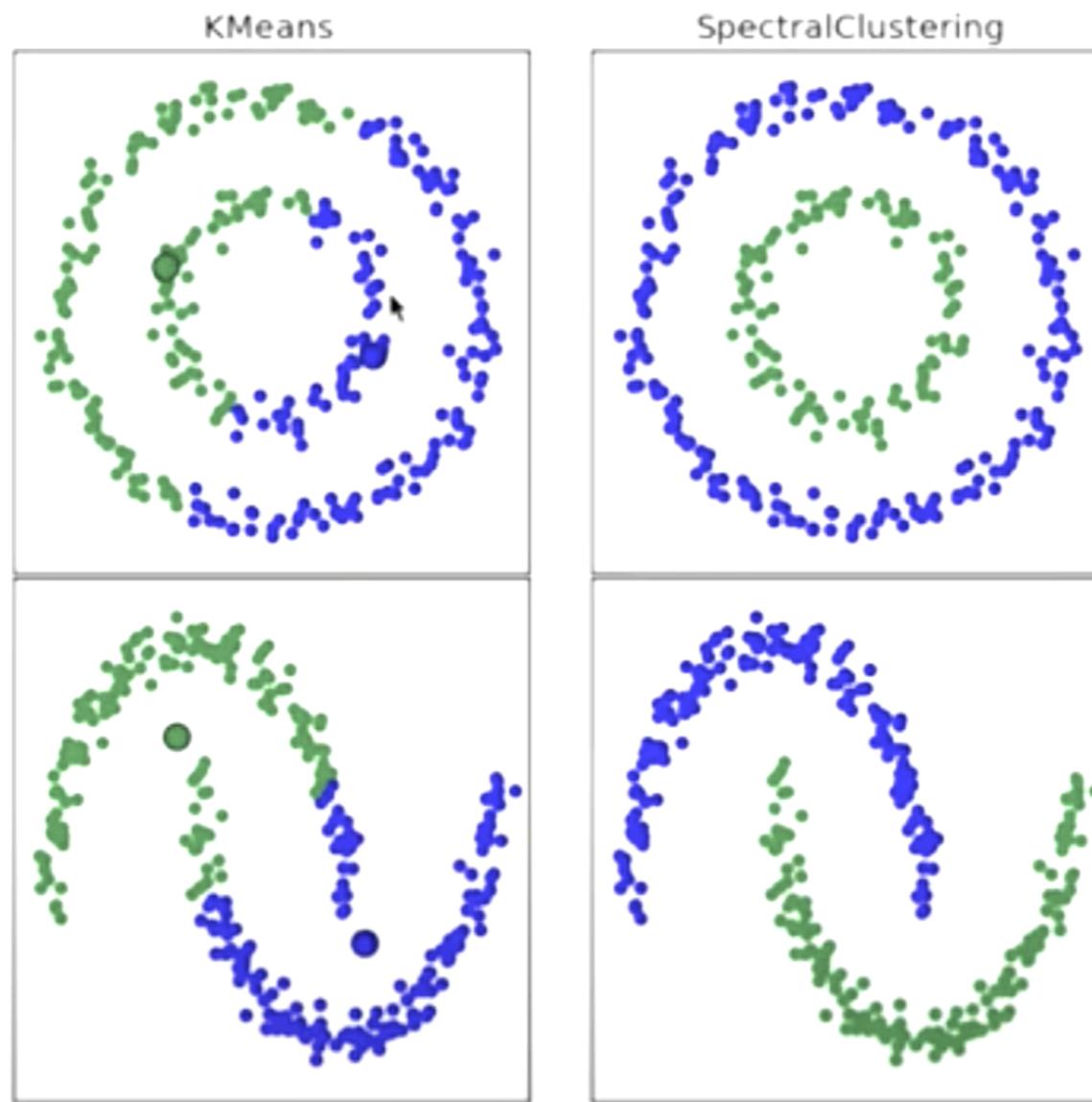


# Difference between **Spectral Clustering** and Conventional Clustering Techniques

Hannah Do  
Jan 4th, 2021

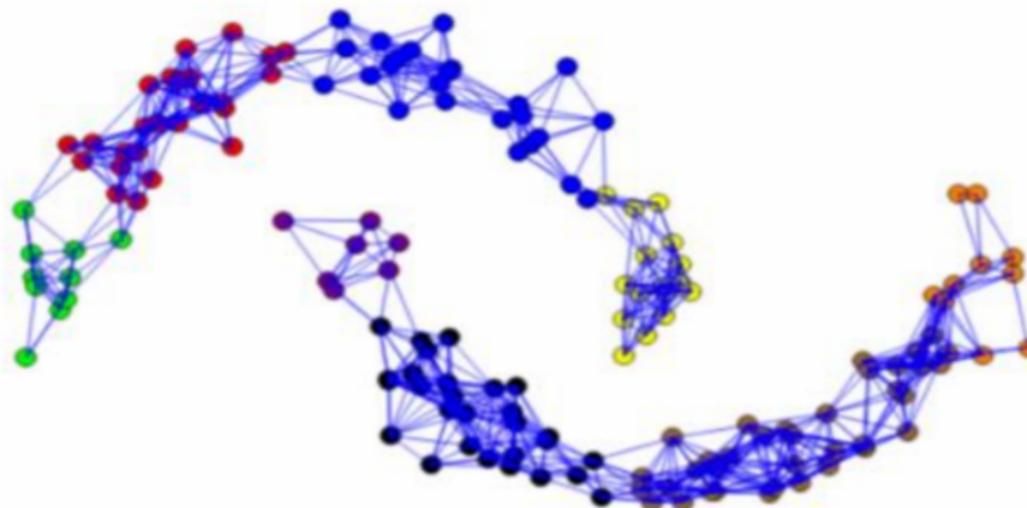
# 1. Introduction

## K Means vs. Spectral Clustering



# 1. Introduction

## Spectral Clustering

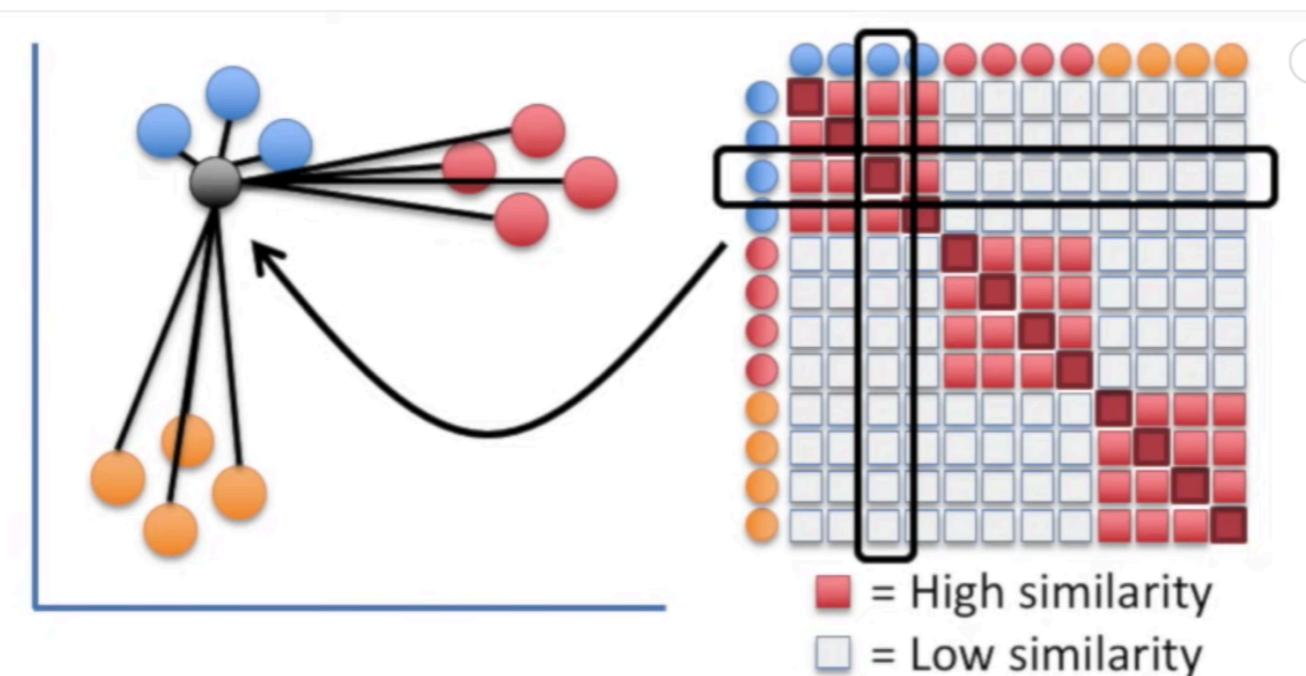


- Spectral Clustering uses information from the eigenvalues of special matrices derived from the **graph** of the dataset.
- Spectral clustering treats the data clustering as a **graph** partitioning problem without making any assumption on the form of the data clusters.

# 2. Principle

## A. Construct a similarity graph for all the data points - same principle as t-SNE

$$A_{ij} = \begin{cases} w_{ij} & : \text{weight of edge } (i, j) \\ 0 & : \text{if no edge between } i, j \end{cases}$$



Statquest: t-SNE, Clearly Explained  
<https://www.youtube.com/watch?v=NEaUSP4YerM>

### Different Methods of determining the edges:

- $\varepsilon$ -neighbor
- KNN
- Fully connected

### Method of determining the weight of the edges :

- Gaussian Kernel

$$w_{ij} = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)$$

## 2. Principle

B. Create a Laplacian graph from the adjacency matrix and diagonal matrix of degrees

$$d_i = \sum_{\{j|(i,j) \in E\}} w_{ij}$$

$$\begin{aligned} d_1 &= w_{12} + w_{13} \\ d_2 &= w_{12} + w_{23} + w_{24} \\ d_3 &= w_{13} + w_{23} \\ d_4 &= w_{24} \end{aligned}$$

$$L_{ij} = \begin{cases} d_i & : \text{if } i = j \\ -w_{ij} & : \text{if } (i, j) \text{ is an edge} \\ 0 & : \text{if no edge between } i, j \end{cases}$$

**A =**

	1	2	3	4
1	0	w <sub>12</sub>	w <sub>13</sub>	0
2	w <sub>12</sub>	0	w <sub>23</sub>	w <sub>24</sub>
3	w <sub>13</sub>	w <sub>23</sub>	0	0
4	0	w <sub>24</sub>	0	0

**L =**

	1	2	3	4
1	d <sub>1</sub>	-w <sub>12</sub>	-w <sub>13</sub>	0
2	-w <sub>12</sub>	d <sub>2</sub>	-w <sub>23</sub>	-w <sub>24</sub>
3	-w <sub>13</sub>	-w <sub>23</sub>	d <sub>3</sub>	0
4	0	-w <sub>24</sub>	0	d <sub>4</sub>

Laplacian graph (L) is formed by subtracting the adjacency matrix (A) from the Diagonal matrix of degrees (D)

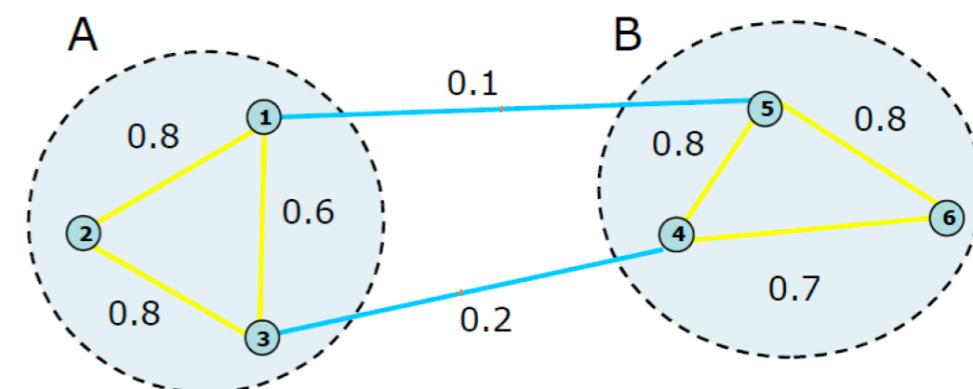
$$\mathbf{L} = \mathbf{D} - \mathbf{A}$$

## 2. Principle

C-a. Calculate eigenvectors and eigenvalues from the Laplacian graph, and choose the Fiedler value (smallest eigenvalue other than zero) to partition the cluster

### Minimum Cut Method

$$\Lambda^T = [0.0 \quad 0.3 \quad 2.2 \quad 2.3 \quad 2.5 \quad 3.0]$$
$$X = \begin{bmatrix} 0.4 & 0.2 & 0.1 & 0.4 & -0.2 & -0.9 \\ 0.4 & 0.2 & 0.1 & 0.0 & 0.4 & 0.3 \\ 0.4 & 0.2 & -0.2 & 0.0 & -0.2 & 0.6 \\ 0.4 & -0.4 & 0.9 & 0.2 & -0.4 & -0.6 \\ 0.4 & -0.7 & -0.4 & -0.8 & -0.6 & -0.2 \\ 0.4 & -0.7 & -0.2 & 0.5 & 0.8 & 0.9 \end{bmatrix}$$



→ Recursive Bipartitioning

## 2. Principle

C-b. Calculate eigenvectors and eigenvalues from the Laplacian graph, and choose the # of K values (# of eigenvalues like in PCA), then perform K-means

**K=3** : Choose 3 sets of Principal Components

$$\Lambda^T = \begin{bmatrix} 0.0 & 0.3 & 2.2 & 2.3 & 2.5 & 3.0 \end{bmatrix}$$
$$X = \begin{bmatrix} 0.4 & 0.2 & 0.1 & 0.4 & -0.2 & -0.9 \\ 0.4 & 0.2 & 0.1 & 0.0 & 0.4 & 0.3 \\ 0.4 & 0.2 & -0.2 & 0.0 & -0.2 & 0.6 \\ 0.4 & -0.4 & 0.9 & 0.2 & -0.4 & -0.6 \\ 0.4 & -0.7 & -0.4 & -0.8 & -0.6 & -0.2 \\ 0.4 & -0.7 & -0.2 & 0.5 & 0.8 & 0.9 \end{bmatrix}$$

# 3. Conclusion

## Comparison to traditional clustering

Affinity Based Clustering	Graph-Clustering
K means	Spectral Clustering
All members of each cluster are in close proximity to each other (euclidean distance-wise)	A similarity matrix is constructed for each data point
Ideal for globular clusters	Ideal for any shape of clusters
$N \times P$ matrix	$P \times P$ matrix Indifferent to # of features
Inconsistent result	Consistent result

$N$  : Number of features/dimensions

$P$  : Number of datapoints