PGE 383 Subsurface Machine Learning

Lecture 7b: Advanced Clustering

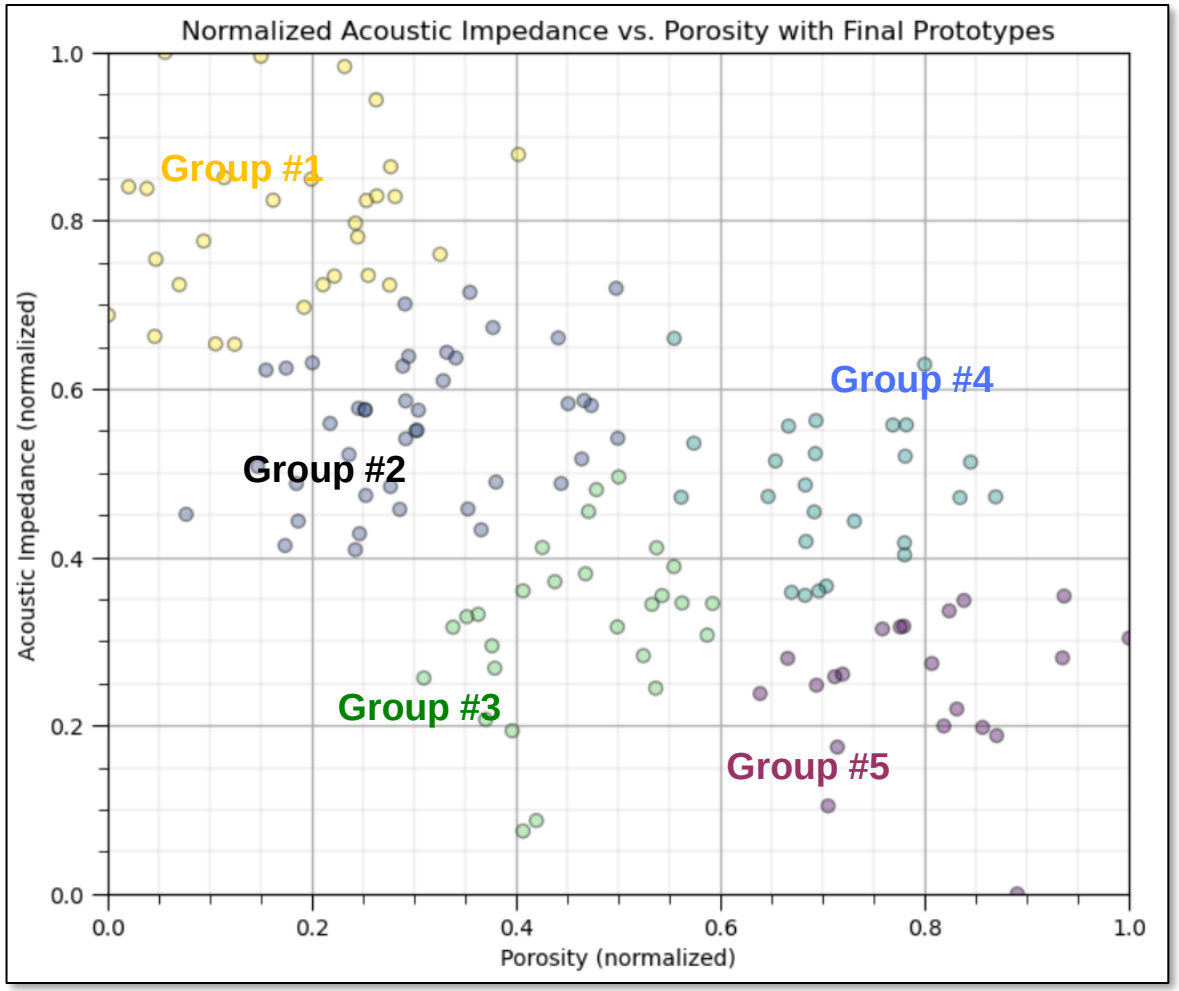
Lecture outline:

- **Centroid-based Clustering**
- **Density-based Clustering**
- **Density-based Clustering Hands-on**
- **Spectral Clustering**
- **Spectral Clustering Hands-on**

Motivation for More Clustering

There are many limitations and assumptions for k-means clustering, real data is complicated, we need other methods.

- spherical, convex, isotropic clusters
- minimize difference within clusters
- equal variance for all features
- reliable measures of distance in feature space
- equal prior probability for all clusters:



Cluster analysis assigns group membership to data. Modified from from Cluster Analysis chapter of Applied Machine Learning in Python e-book at, https://geostatsguy.github.io/MachineLearningDemos_Book/.

PGE 383 Subsurface Machine Learning

Lecture 7b: Advanced Clustering

Lecture outline:

- **Centroid-based Clustering**

k-means Clustering Assumptions

Assumptions of k-means clustering

1. spherical, convex, isotropic clusters

   minimize difference within clusters

   we are able to work with more complicated geometries.

2. equal variance for all features

$$\sigma_{X_1}^2 = \sigma_{X_2}^2 = \cdots = \sigma_{X_m}^2$$

   reliable measures of distance in feature space

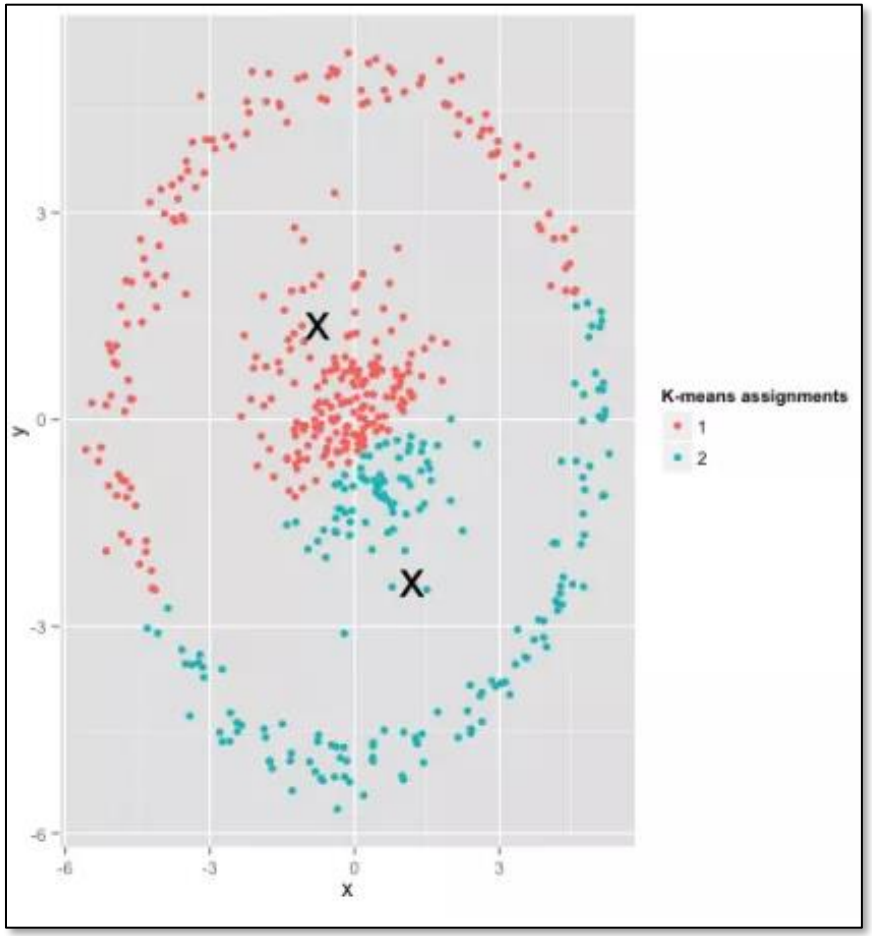
   we still use normalization or standardization

3. similar sized / frequency clusters

   larger cluster are divided to minimize overall variance within clusters

   we will work with local sample high densities or pairwise connections

   clusters with few samples in feature space are overwhelmed!

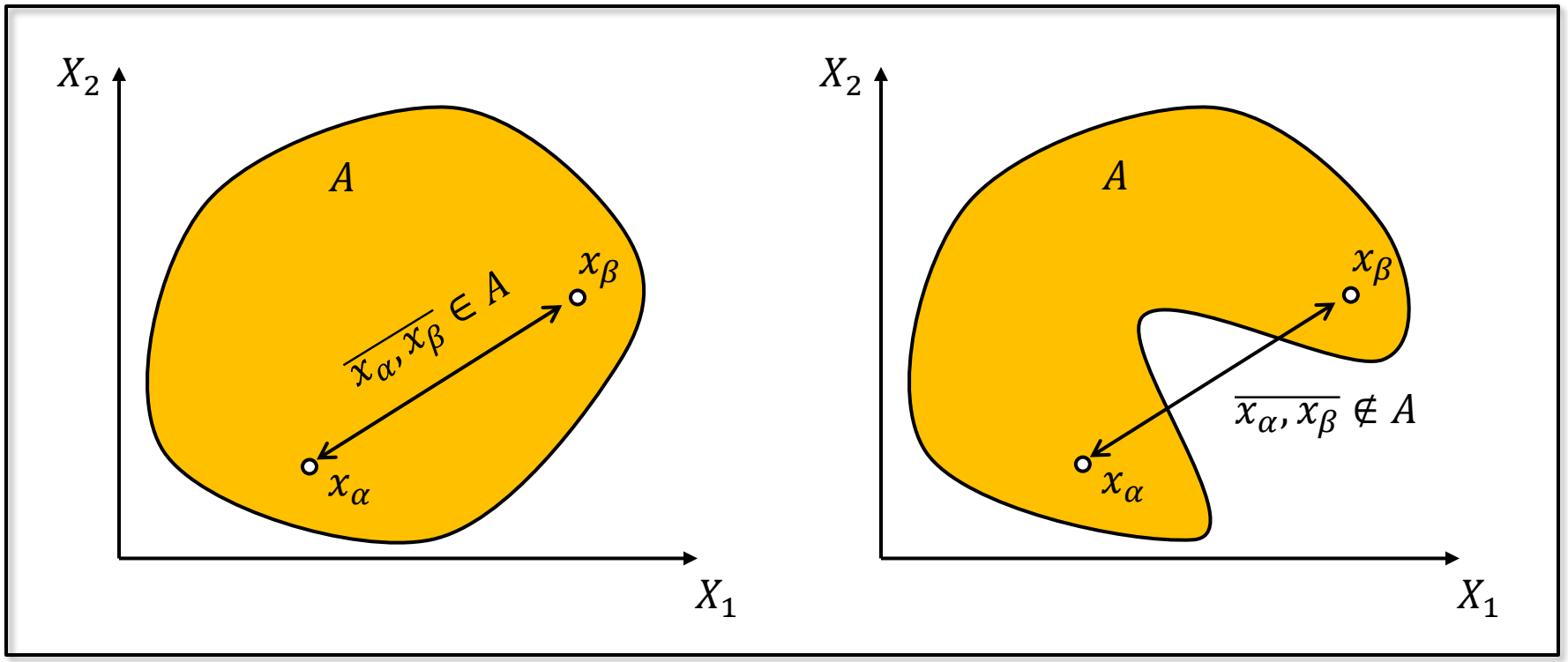


Nonspherical example figure from https://www.r-bloggers.com/k-means-clustering-is-not-a-free-lunch/.

Data Convexity

A subset (cluster) of Euclidean / Feature Space is convex if:

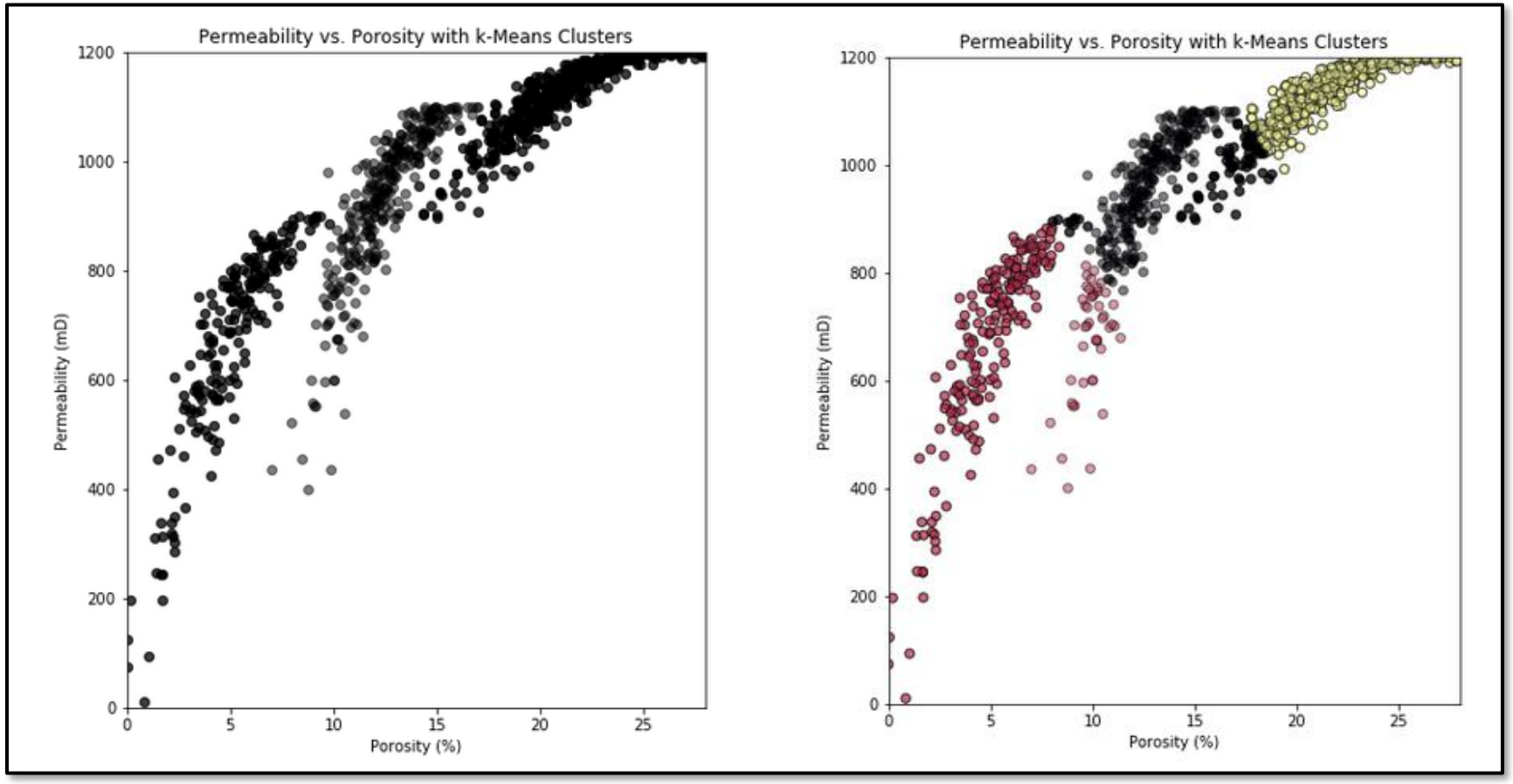
For any two points x_1 and x_2 within a subset A, the entire line segment $\overline{x_1, x_2}$ exists in A. $\overline{x_\alpha, x_\beta} \in A \; \forall \; \alpha, \beta$



Convex shape (left) and nonconvex shape (right) in feature space.

A More Complicated Dataset

A subset data set with anisotropic, nonconvex, density separable facies.



Non-convex datset, from GeoDataSets repository.

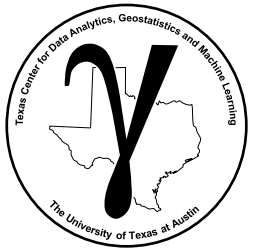
k-means clustering fails to accurately identify the facies.

PGE 383 Subsurface Machine Learning

Lecture 7b: Advanced Clustering

Lecture outline:

- **Density-based Clustering**



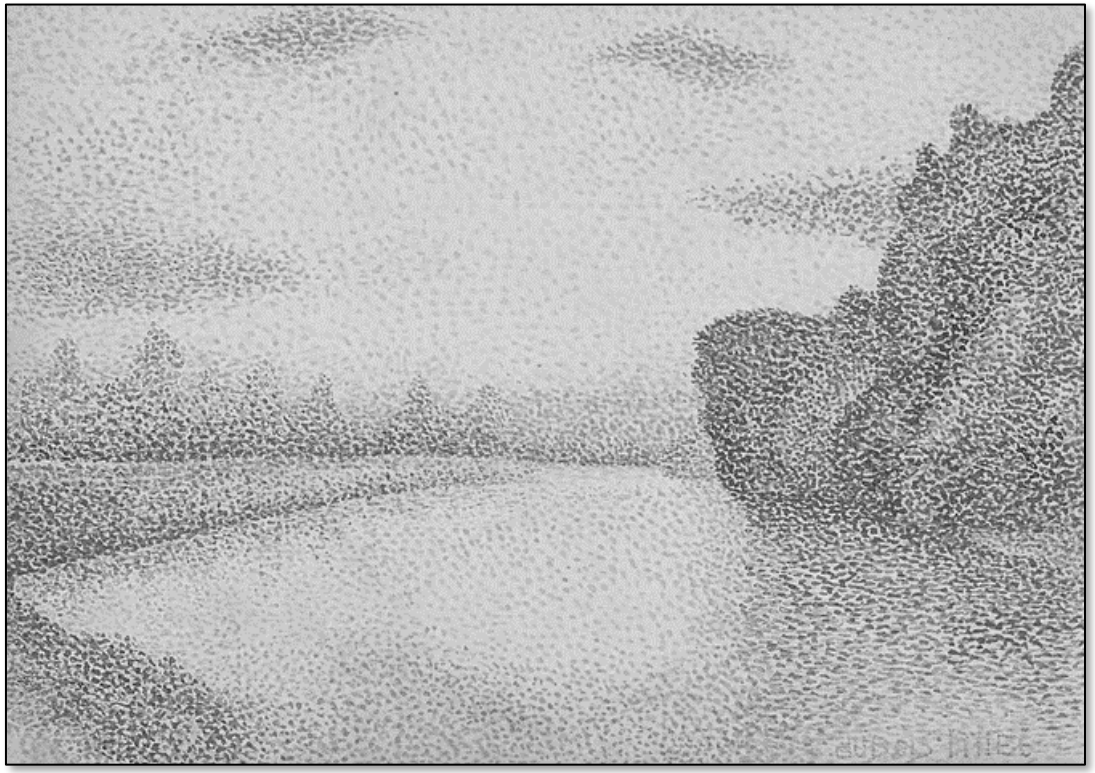
DBSCAN

Density-based: what is density in data over the predictor feature space?

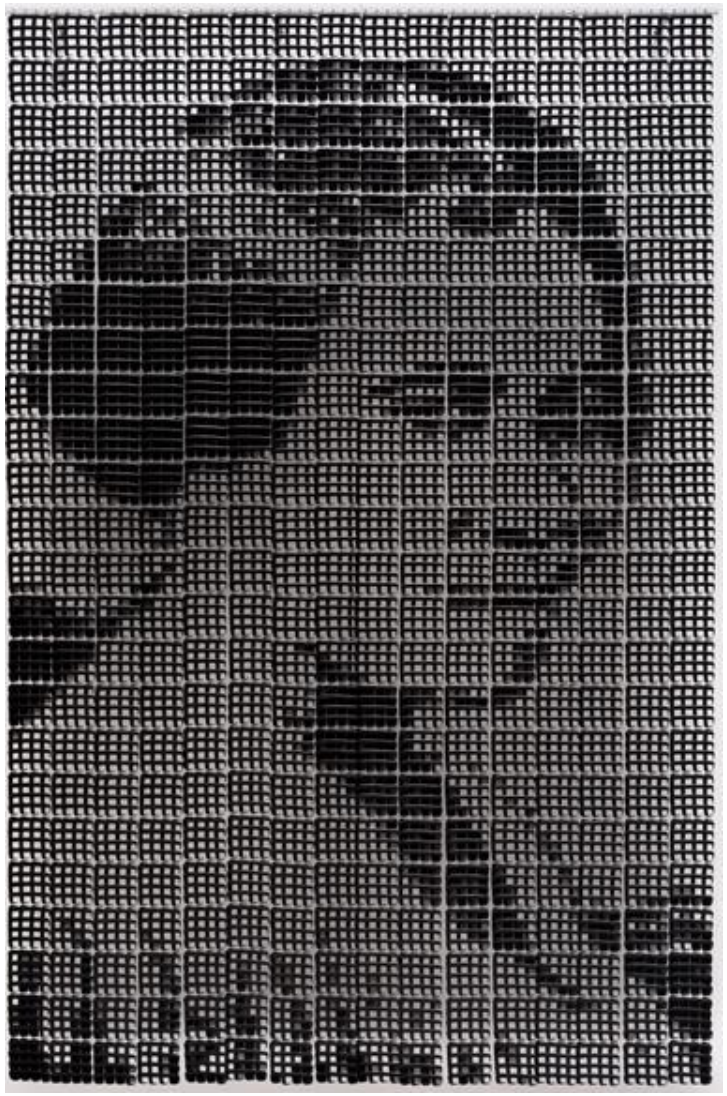
DBSCAN

Density-based: what is ‘density’ in data over the predictor feature space?

- Can you visualize density? Let’s look at art!



“The banks of the Marne at dawn.”, by Albert Dubois-Pillet (1886), pointillism, I removed the color to highlight point density.



“Madam C.J. Walker”, by Sonya Clark (2008), made from regularly spaced combs with teeth removed, in the Blanton Museum of Art.

DBSCAN

Density-based: groups form in the feature space at locations with sufficient point density determined by hyperparameters:

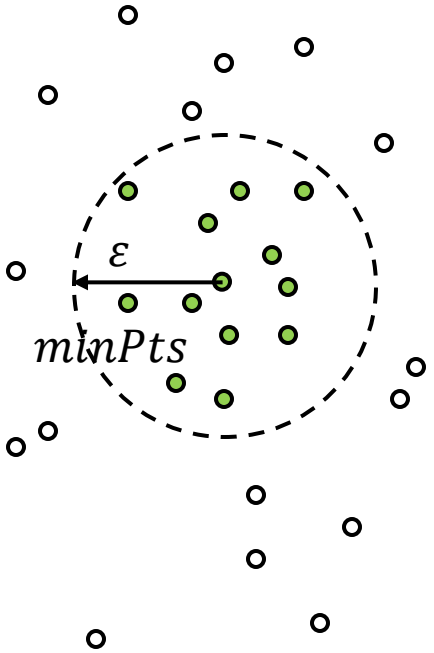
ε – radius of the local neighbourhood in the metric of normalized features

What is the scale / resolution of the clusters?

- Too small, too many samples are left as outliers. Too large, all the cluster merge to one cluster.

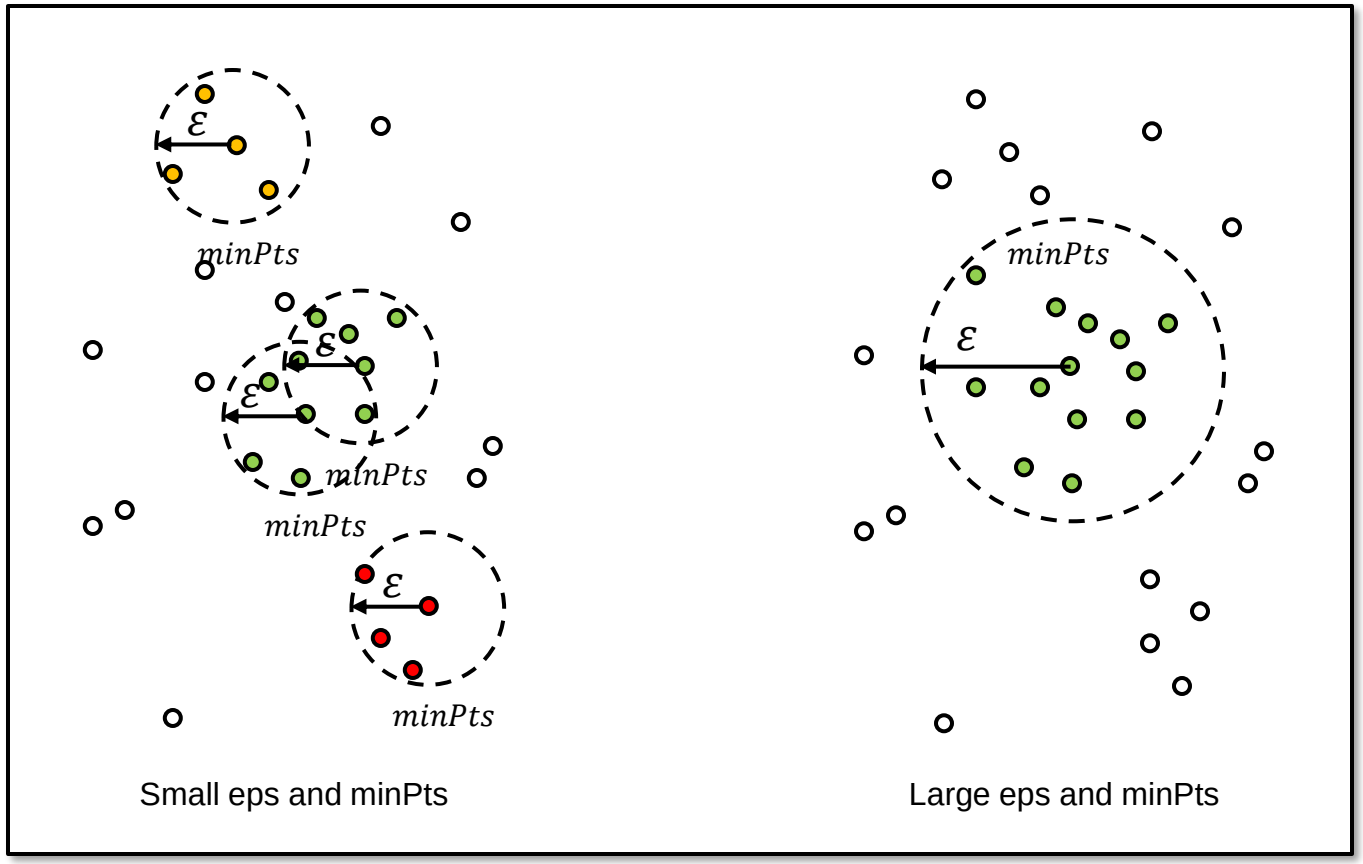
$minPts$ – minimum number of points to assign a core point, initialize or grow a cluster

- Too small, no outliers. Too large, all outliers.



DBSCAN

Density-based: groups form in the feature space at locations with sufficient point density determined by hyperparameters:



Schematic of DBSCAN impact of hyperparameters on clusters.

DBSCAN

DBSCAN, density-based spatial clustering of applications with noise (Ester et al.,1996). Advantages:

- Minimum domain knowledge to estimate hyperparameters, arbitrary shape clusters and efficient on large data sets

Aspects of DBSCAN,

- **Mutually exclusive** – all samples may only belong to one cluster

$$P\left(C_i \cap C_j \mid i \neq j\right) = 0.0$$

- **Non-exhaustive** – some samples may be unassigned / outliers

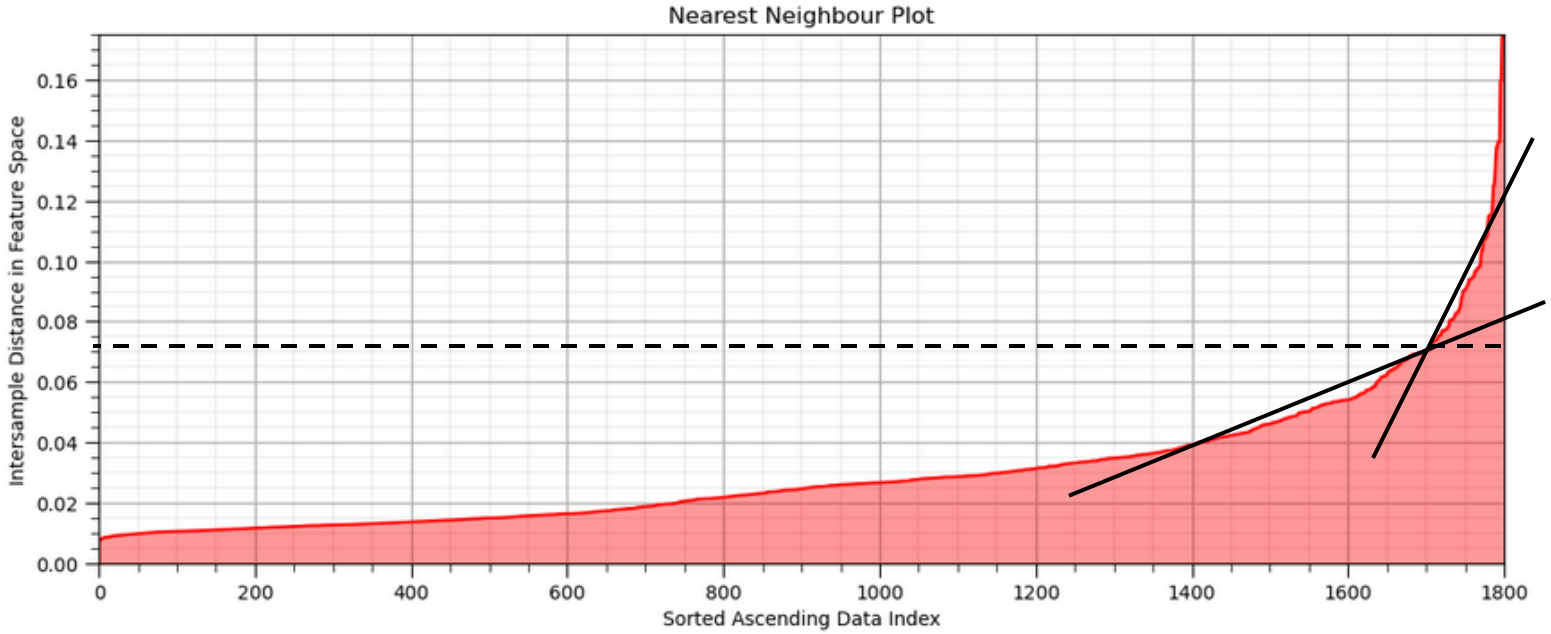
$$P(C_1 \cup C_2 \cup \cdots \cup C_K) \leq 1.0$$

DBSCAN

Parameter Estimation

ε – by k-distance graph (k = min sample, nearest neighbor only)

- Calculate the nearest neighbor distance in normalized feature space for all the sample data (1,700 in this case). Sort in ascending order and plot.
- Select the distance that maximizes the positive curvature (the elbow)



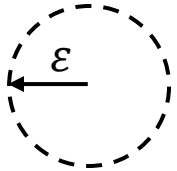
Nearest neighbour plot,
from MachineLearning density-based clustering chapter of e-book.

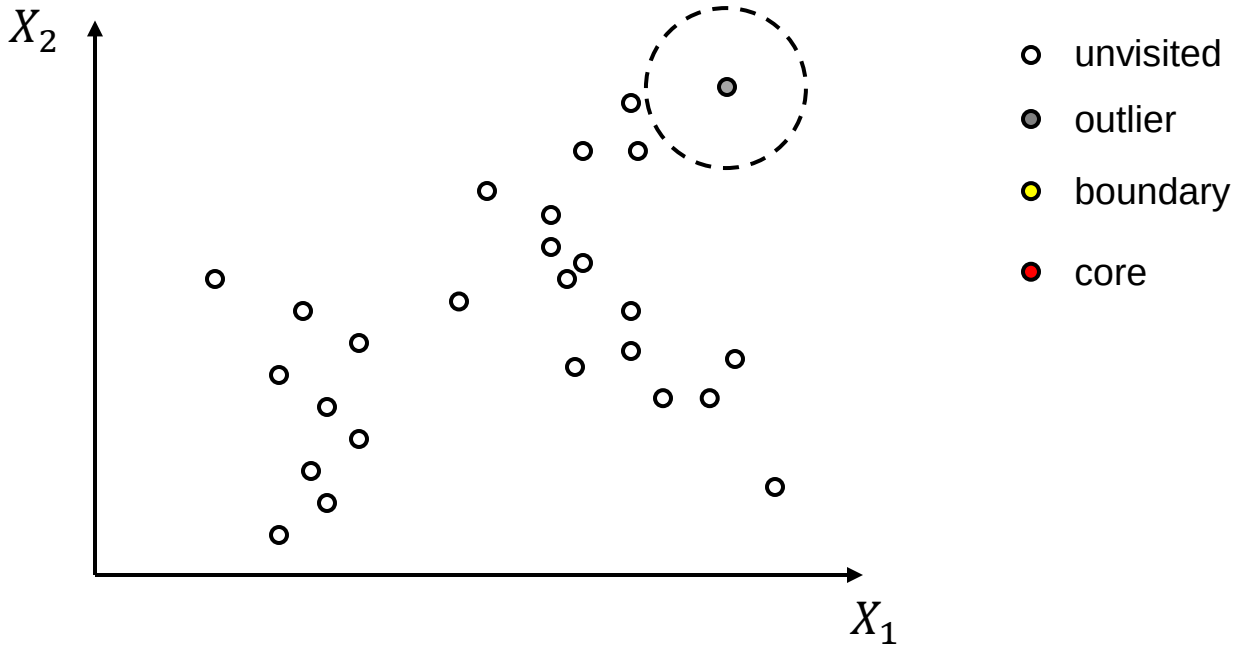
Elbow Method: point of diminishing returns, or break in slope.

DBSCAN Method

Let's Assume $\min\_samples = 4$ **and**

- Initialize all data as unvisited
- Randomly visit an unvisited sample
  - If < min sample within radius of eps → outlier

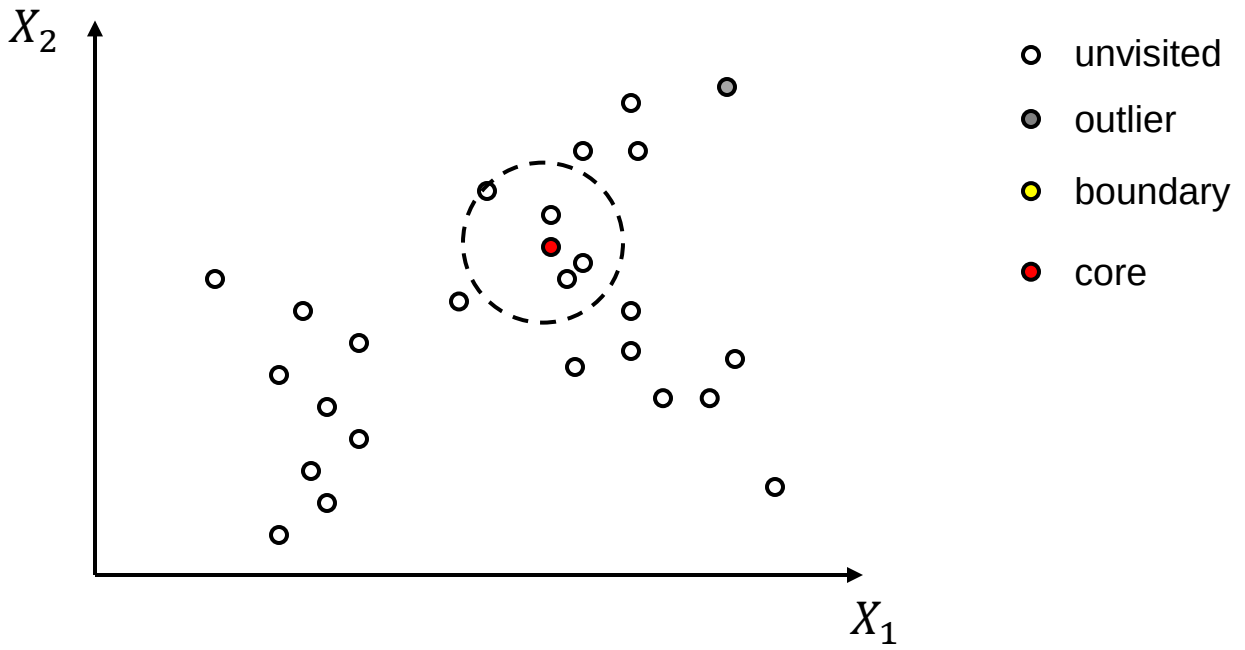




DBSCAN Method

Randomly visit an unvisited sample

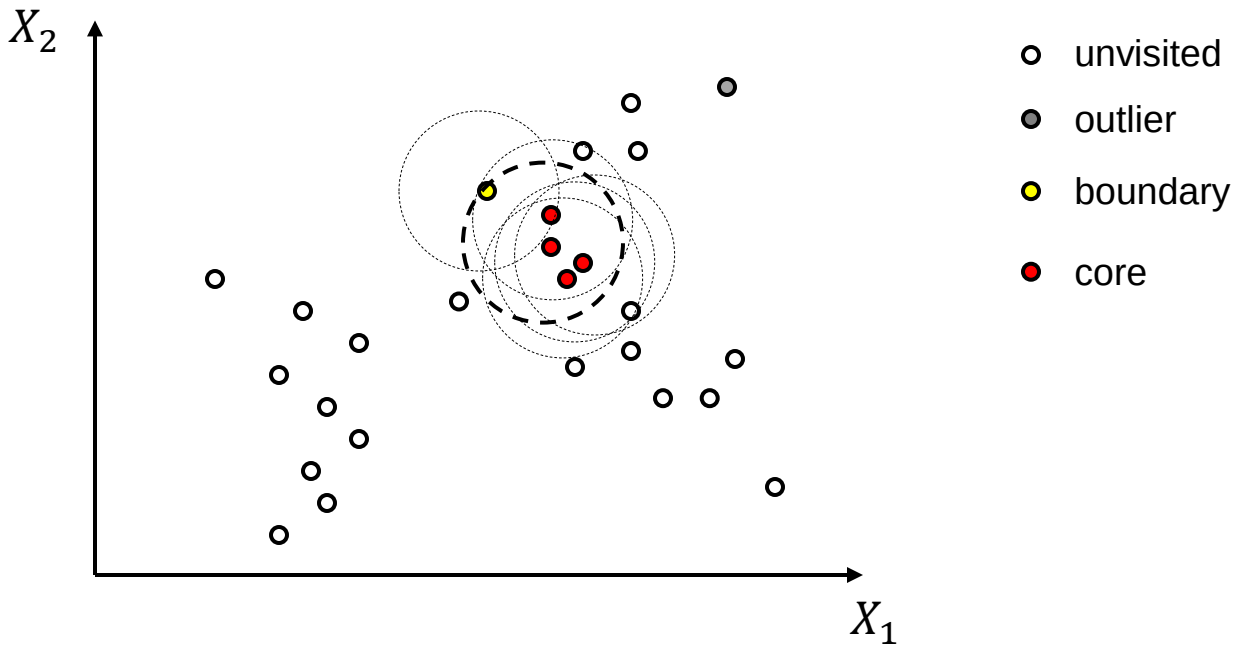
- If > min sample within radius of eps → core
- Now visit the samples within the radius



DBSCAN Method

Now visit the samples within the radius

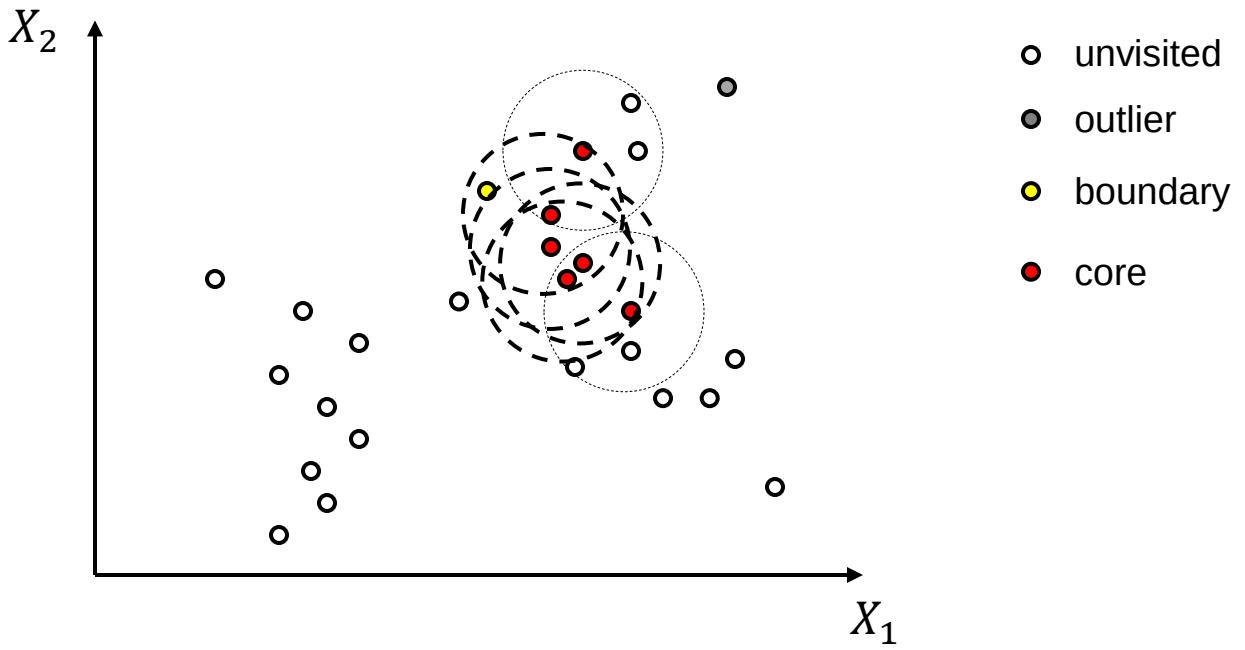
- If > min sample within radius of eps → core
- If < min sample or core → boundary



DBSCAN Method

Randomly visit an unvisited sample

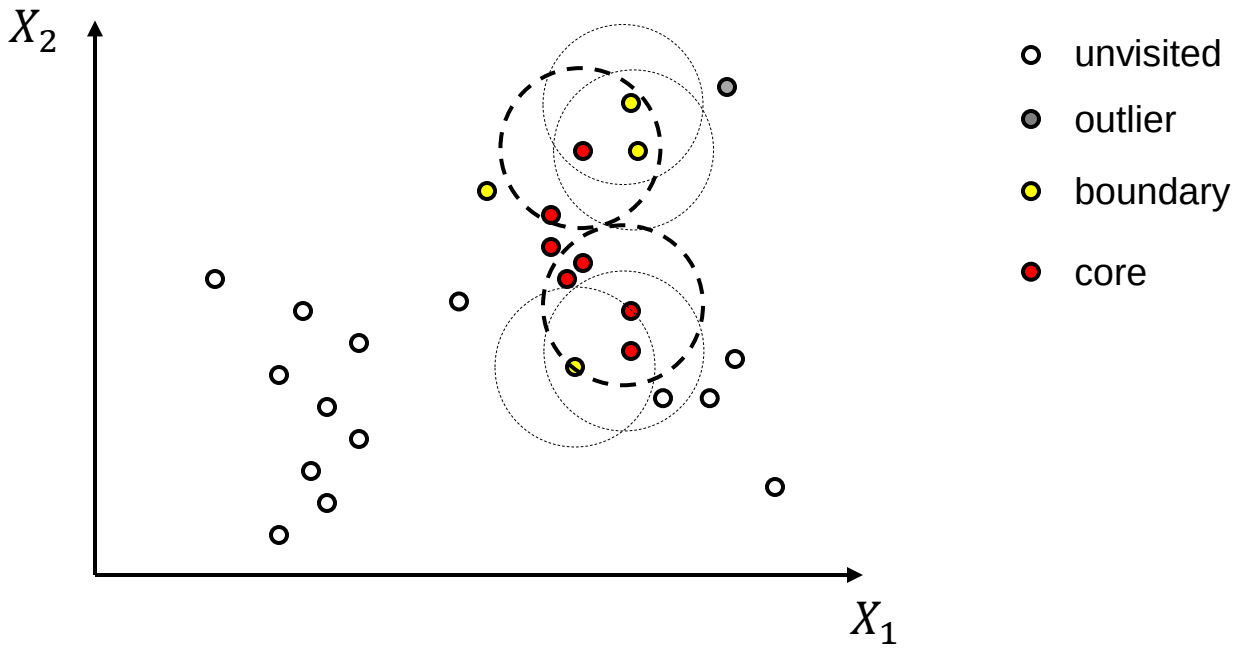
- If > min sample within radius of eps → core
- Now repeat for the new core locations



DBSCAN Method

Randomly visit an unvisited sample

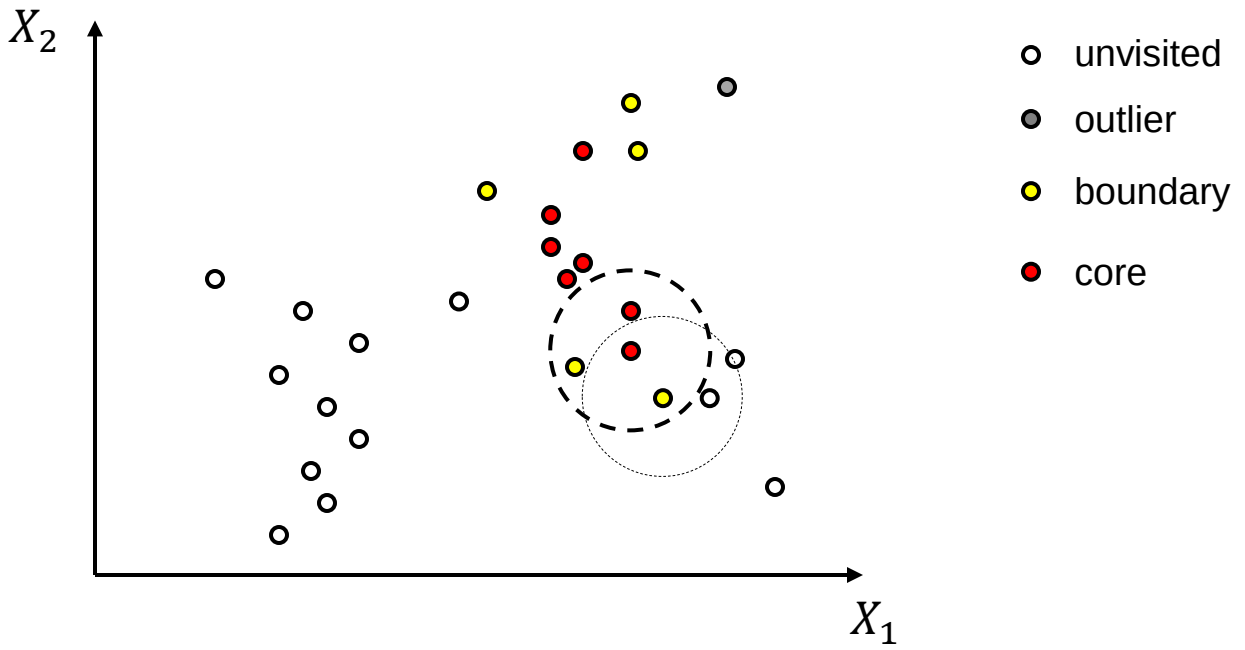
- If > min sample within radius of eps → core
- Now repeat for the new core locations



DBSCAN Method

Randomly visit an unvisited sample

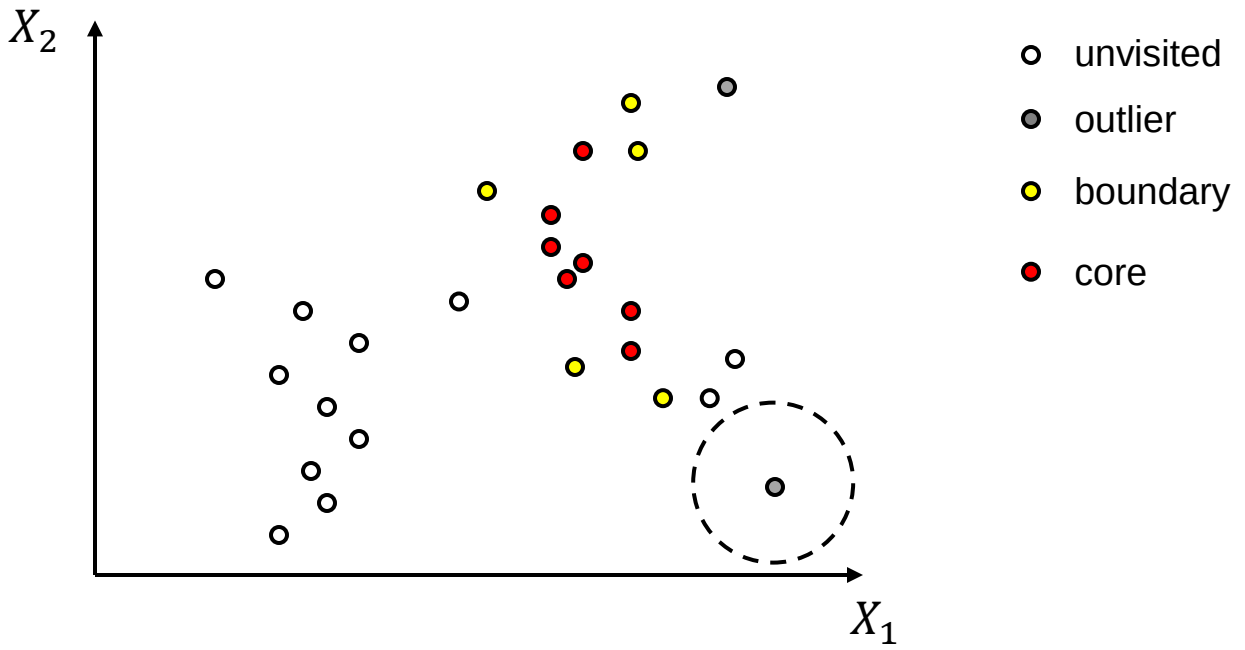
- If > min sample within radius of eps → core
- Now repeat for the new core locations



DBSCAN Method

Randomly visit an unvisited sample

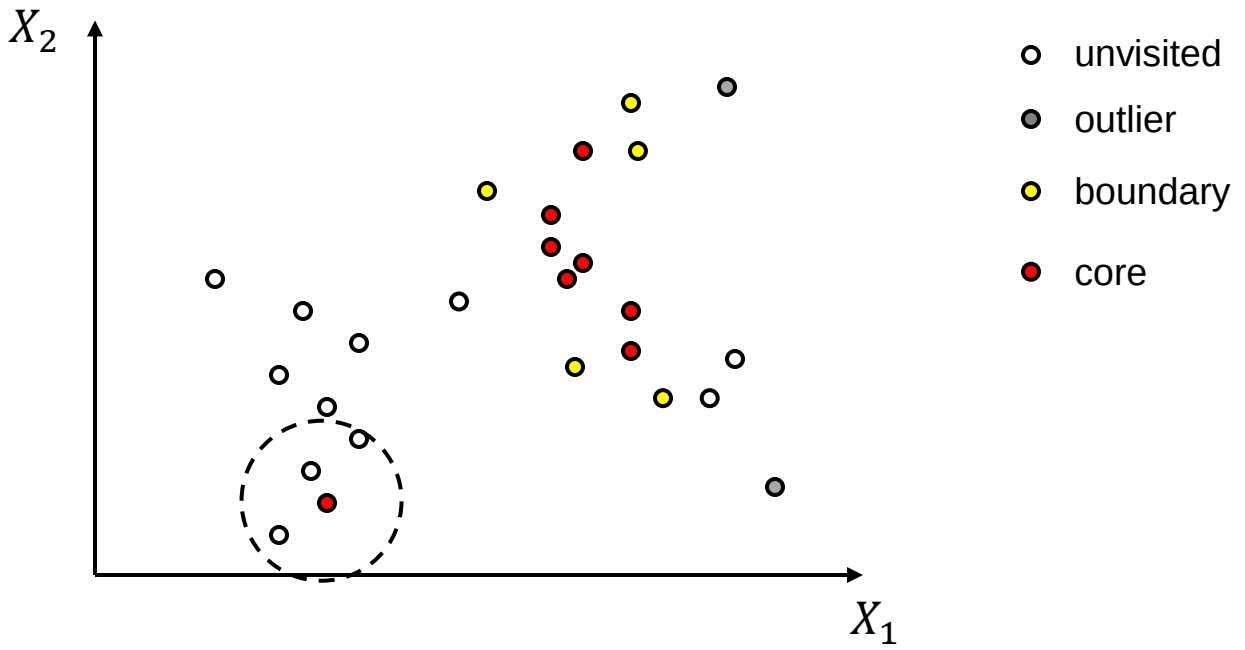
- If < min sample within radius of eps → outlier



DBSCAN Method

Randomly visit an unvisited sample

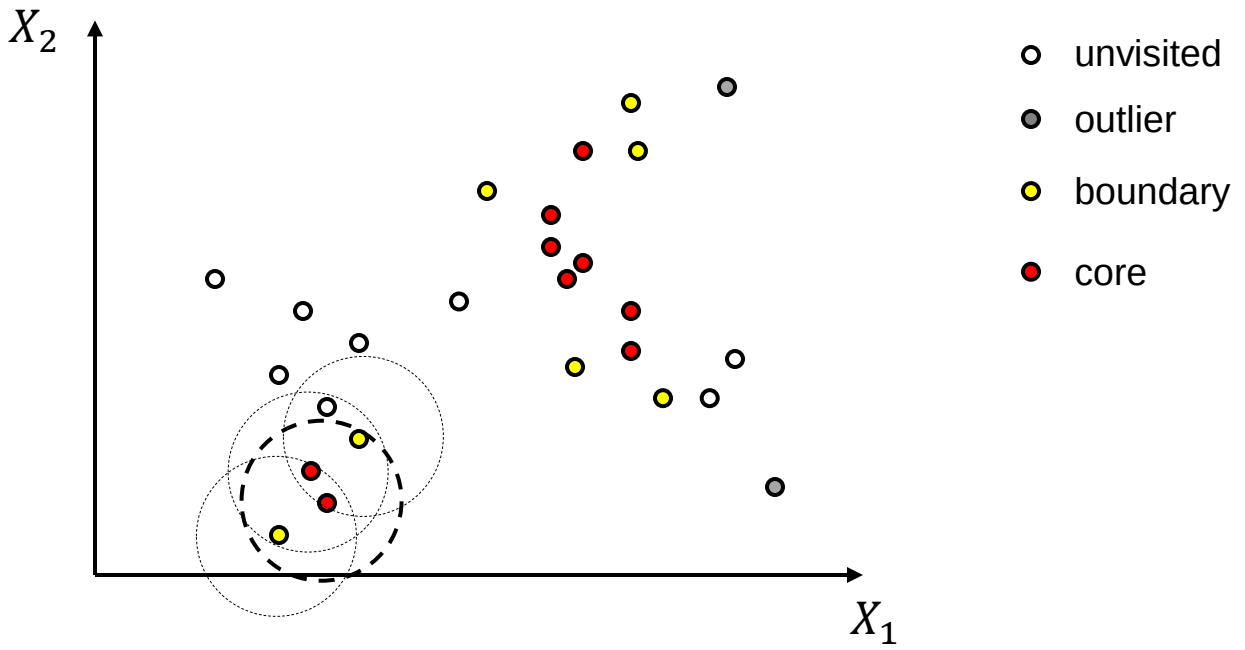
- If > min sample within radius of eps → core
- Now visit the samples within the radius



DBSCAN Method

Now visit the samples within the radius

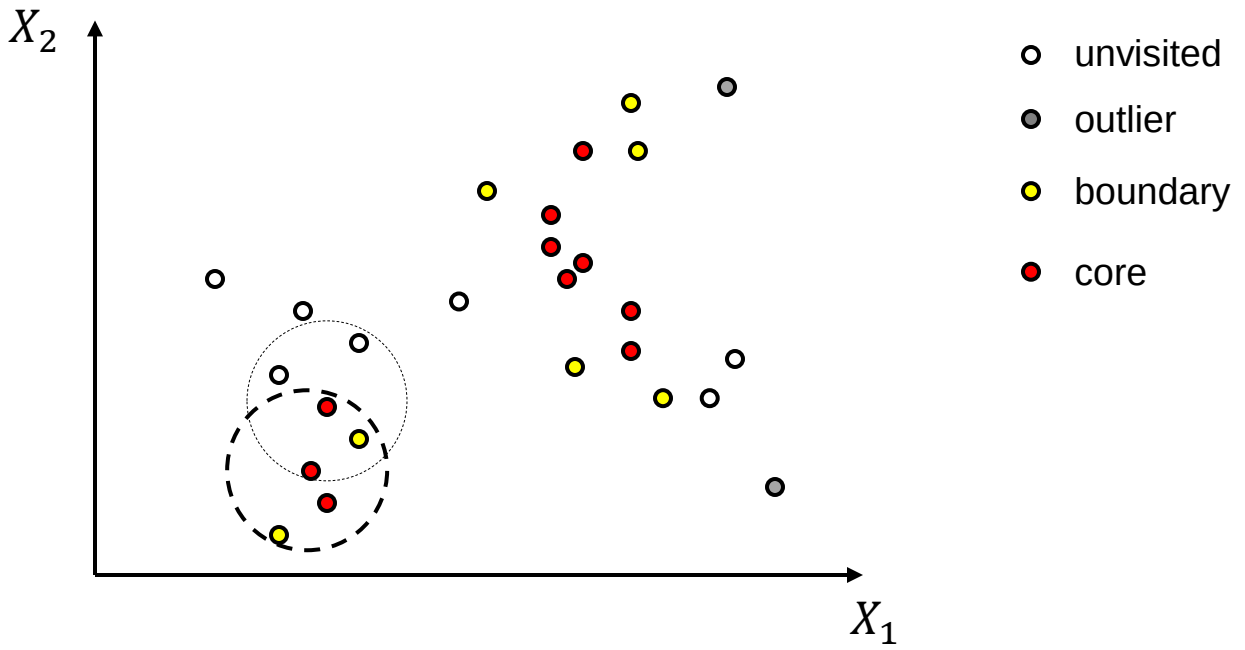
- If > min sample within radius of eps → core
- If < min sample or core → boundary



DBSCAN Method

Now visit the samples within the radius

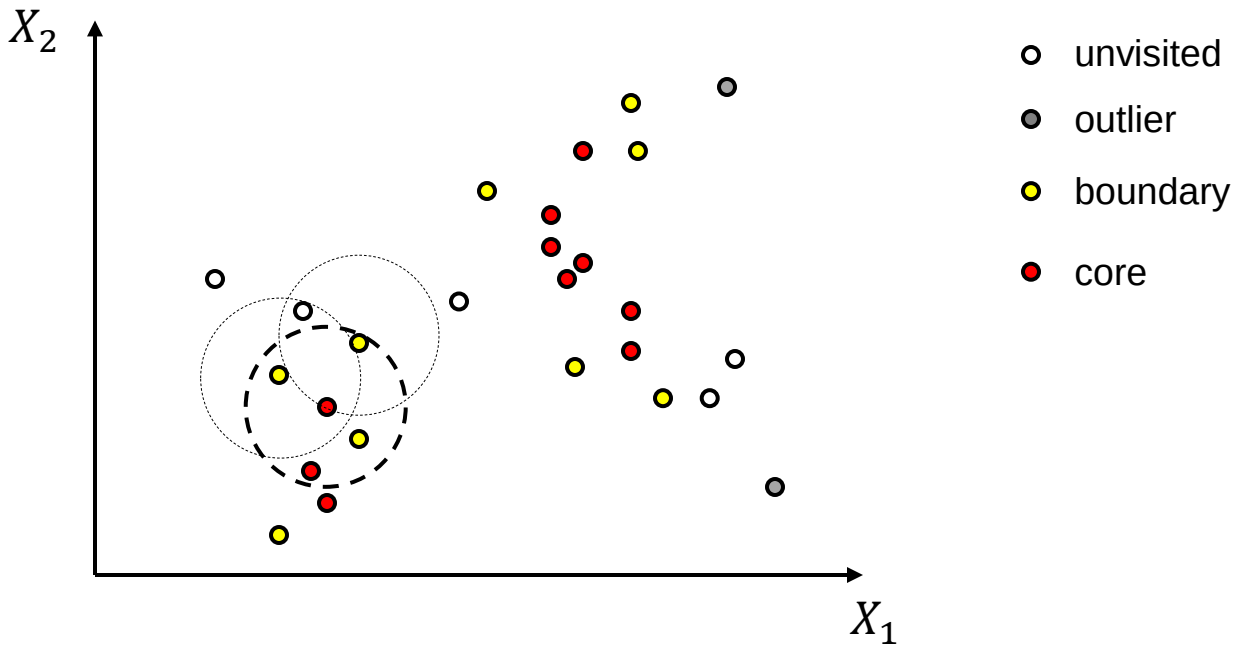
- If > min sample within radius of eps → core
- If < min sample or core → boundary



DBSCAN Method

Now visit the samples within the radius

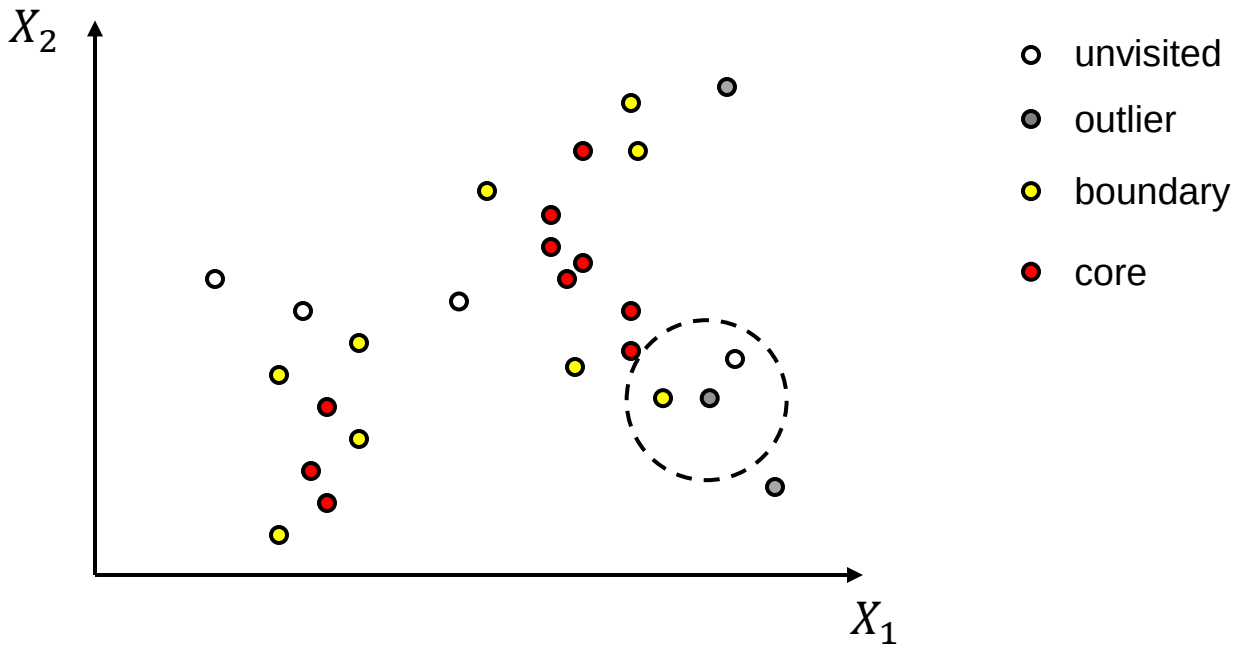
- If > min sample within radius of eps → core
- If < min sample or core → boundary



DBSCAN Method

Randomly visit an unvisited sample

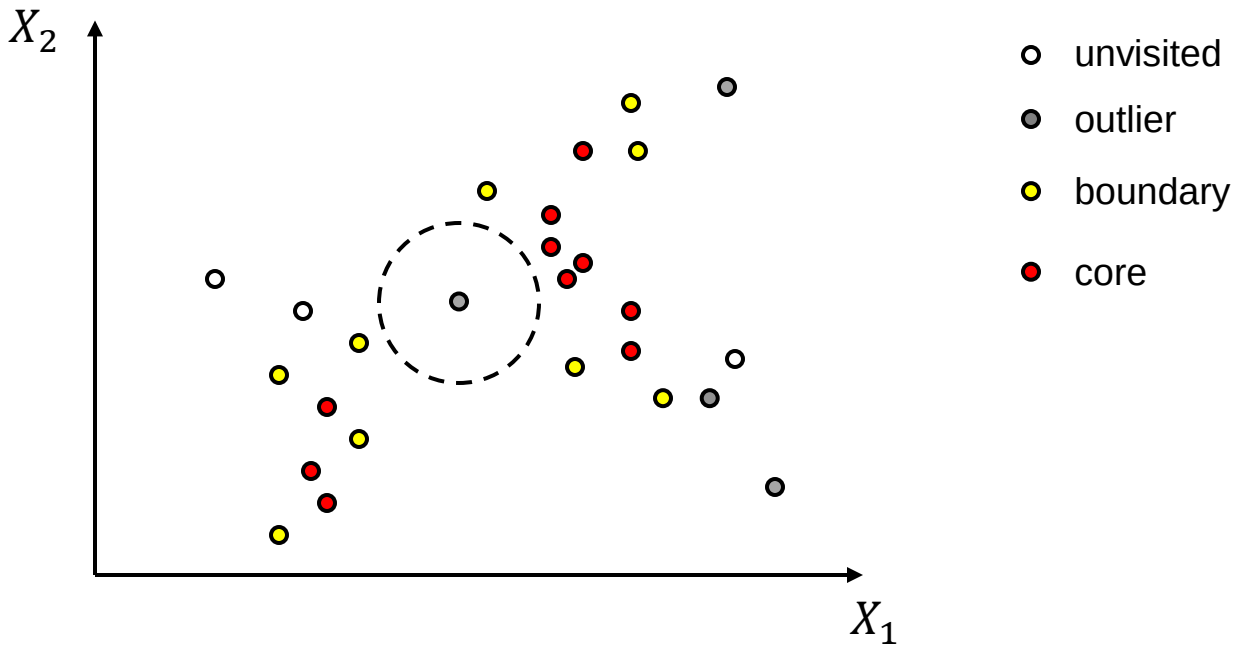
- If < min sample within radius of eps → outlier



DBSCAN Method

Randomly visit an unvisited sample

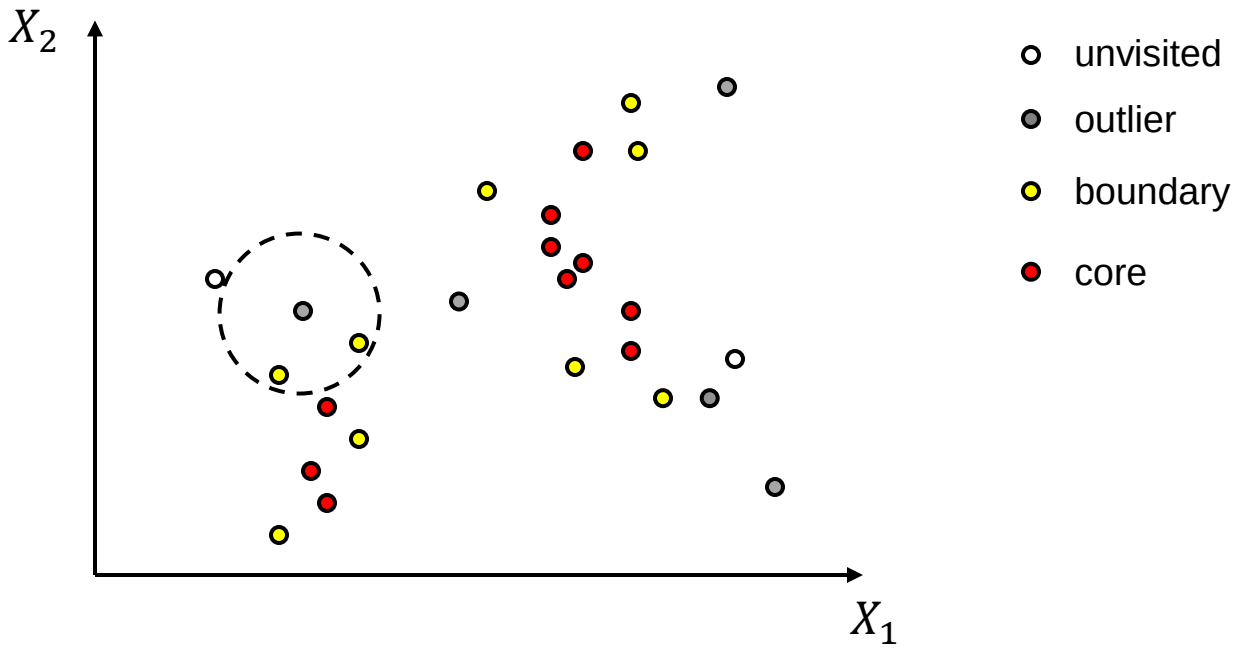
- If < min sample within radius of eps → outlier



DBSCAN Method

Randomly visit an unvisited sample

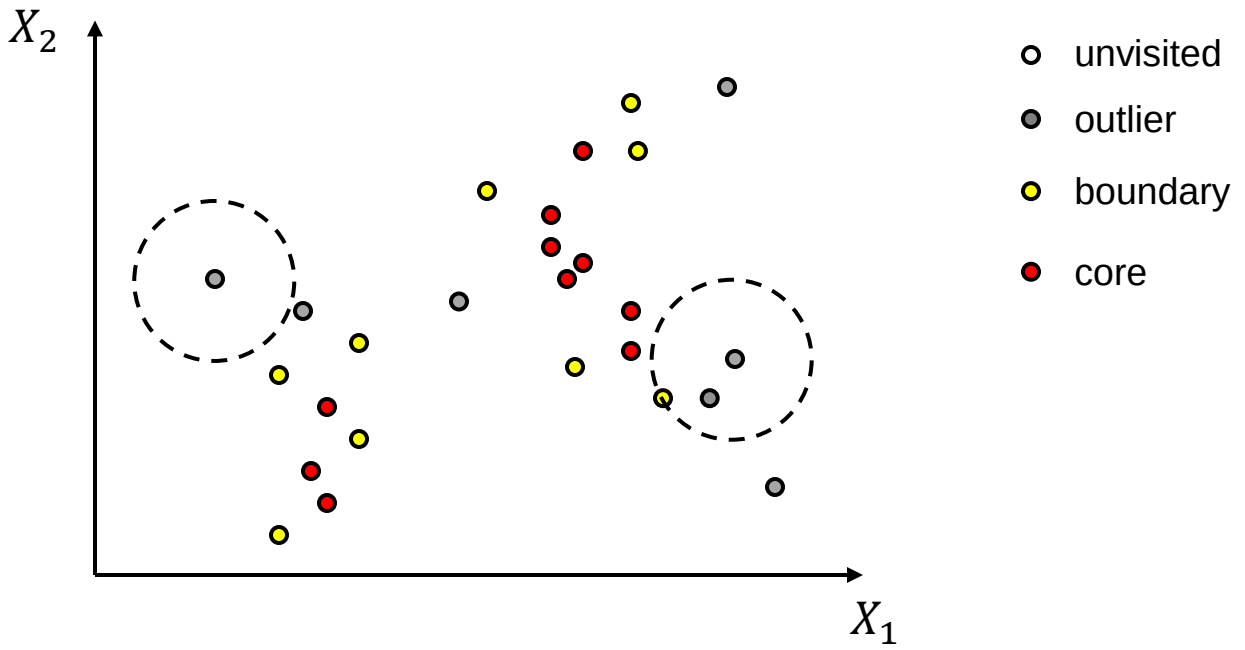
- If < min sample within radius of eps → outlier



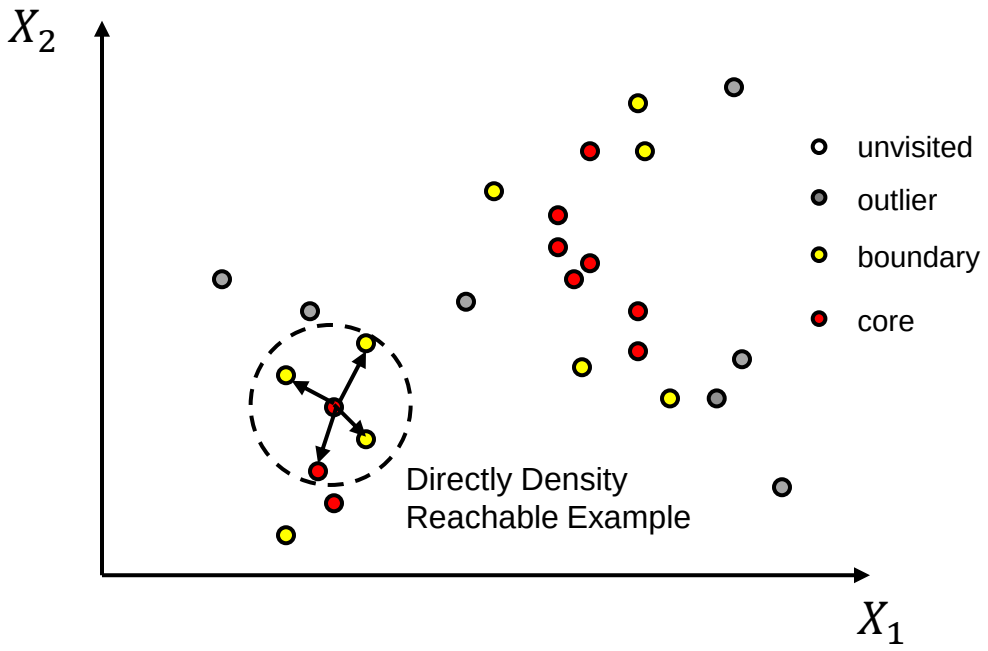
DBSCAN Method

Randomly visit an unvisited sample

- If < min sample within radius of eps → outlier



DBSCAN Method



The Workflow

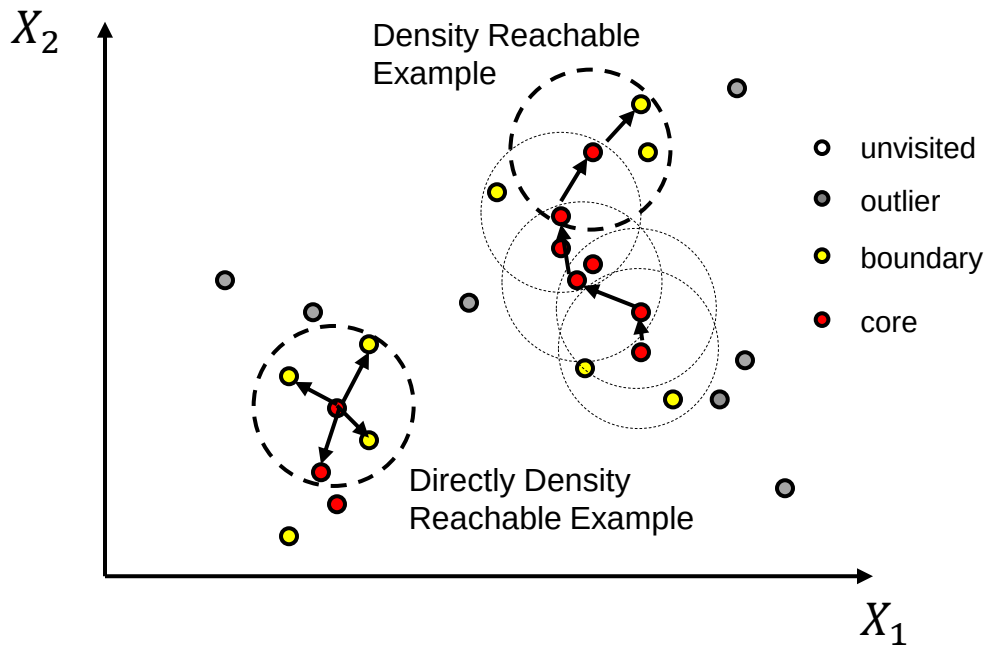
- Now we need to assign clusters

Let's first define some concepts:

- **Directly Density Reachable** - sample x_α is directly density reachable from x_β , if x_β is a core point and x_α belongs to the neighborhood from x_β

$$\left\| x_\alpha, x_\beta \right\| \leq \varepsilon$$

DBSCAN Method



The Workflow

- Now we need to assign clusters

Let's first define some concepts:

- **Directly Density Reachable** - sample x_α is directly density reachable from x_β , if x_β is a core point and x_α belongs to the neighborhood from x_β

$$\left\| x_\alpha, x_\beta \right\| \leq \varepsilon$$

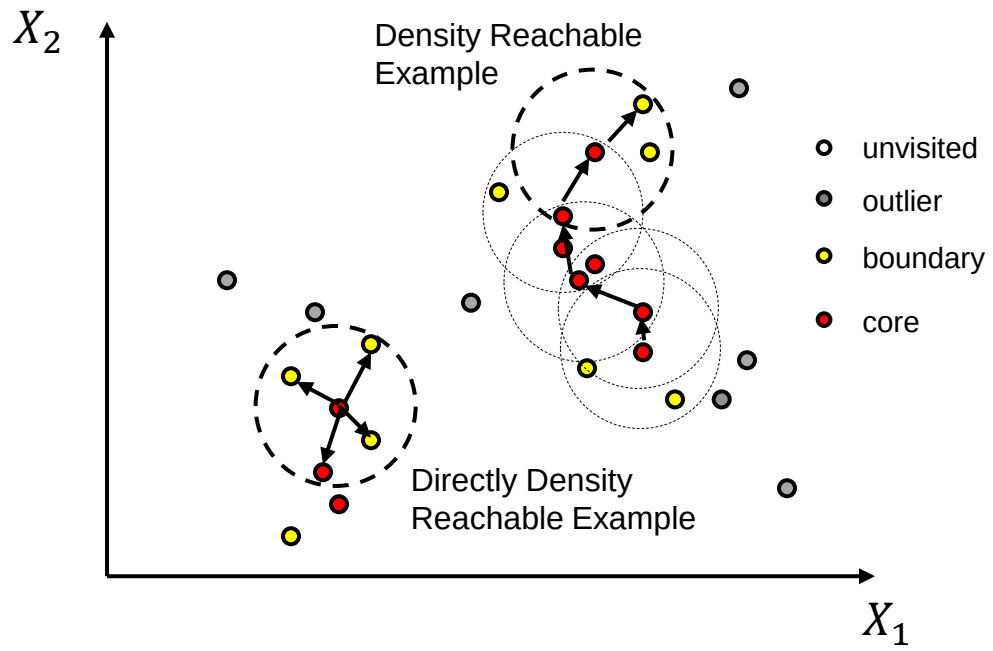
- **Density Reachable** - point x_α is density reachable from x_β if x_α belongs to a neighborhood of a core point that can reached from x_β. This would require a chain of core points each belonging the previous core points and the last core point including point x_α.

DBSCAN Method

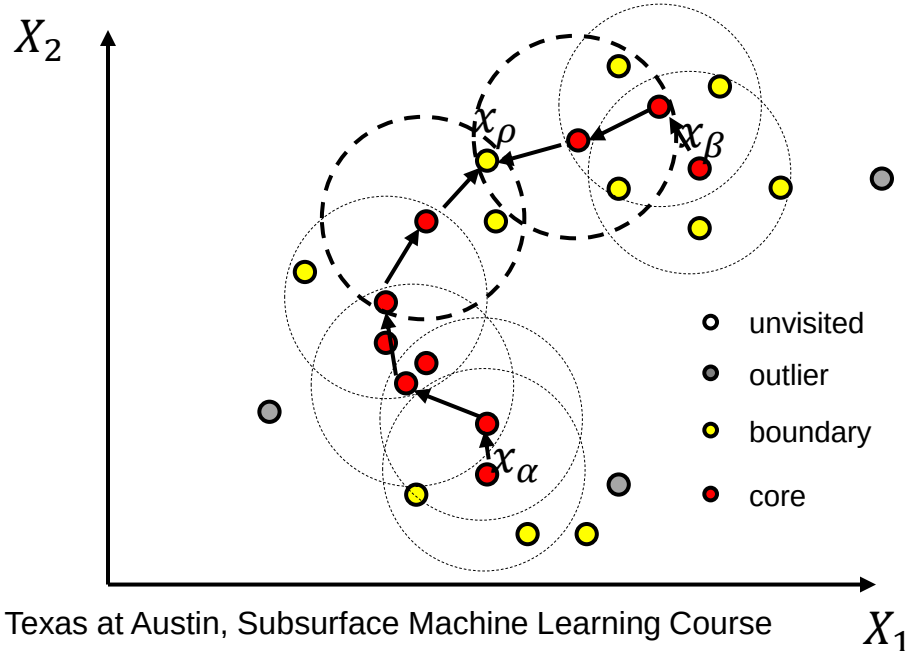
The Workflow

- Now we need to assign clusters

Let's first define some concepts:



- **Density Connected** - x_α and x_β are density connected if there is a common point, x_ρ, density reachable to both.



We now can use density connected clusters as our categories.

DBSCAN Method

The Workflow

- Now we can assign connected clusters are our categories.

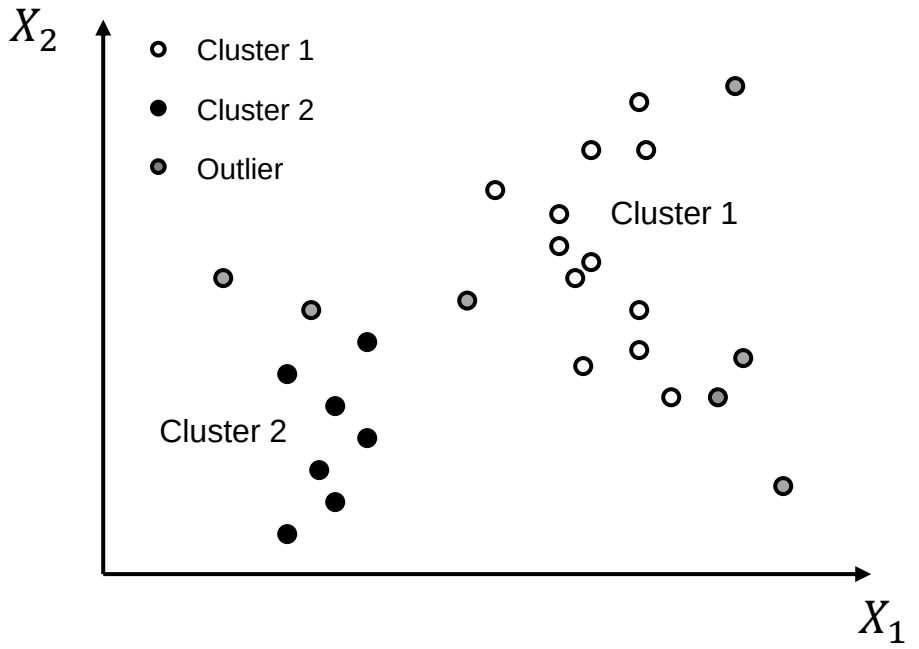
- Note, no need for a priori assignment of number of clusters.

- **Density Connected Clusters**

  - Here's our density connected clusters

  - If we increased ε we could have joined these clusters and included the outliers.

  - ε is a measure of the scale

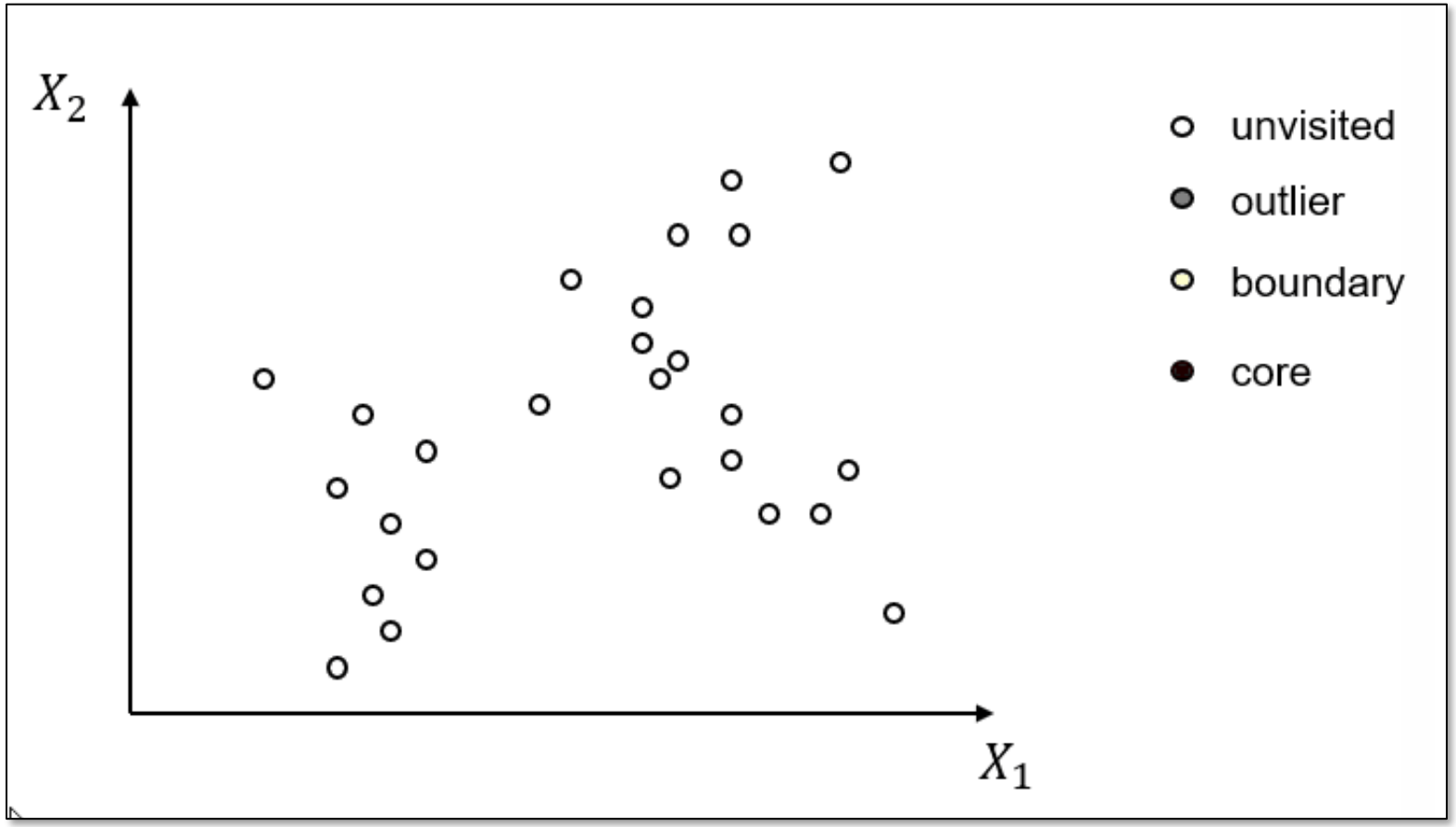


- min_samples a measure of required density.

DBSCAN Method

Hierarchical, agglomerative clustering with DBSCAN

- Let's watch DBSCAN assign core, boundary and outliers samples in the predictor feature space.



Animated .gif of DBSCAN from the previous slides.

PGE 383 Subsurface Machine Learning

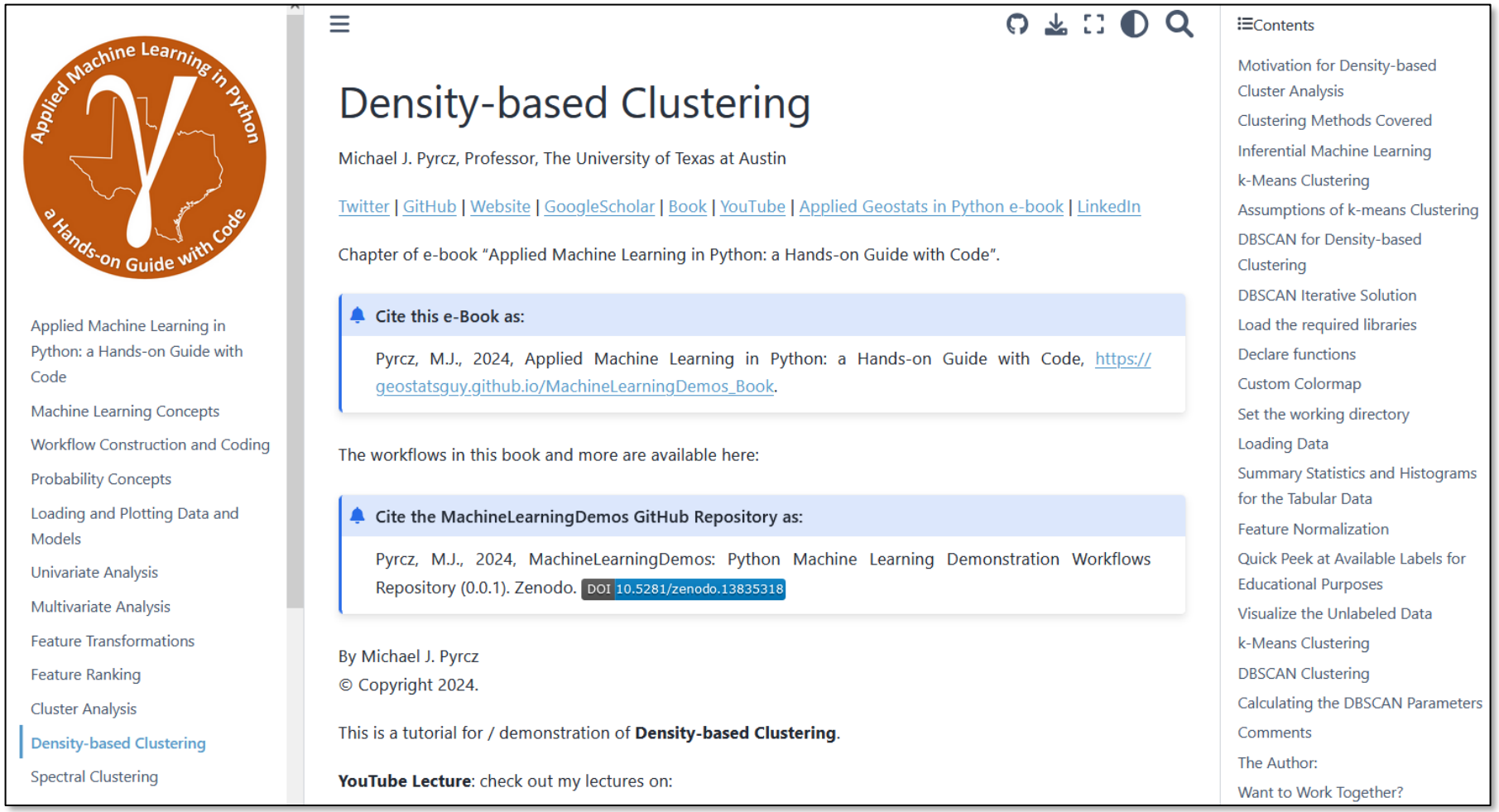
Lecture 7b: Advanced Clustering

Lecture outline:

- **Density-based Clustering Hands-on**

DBSCAN Hands-on

Demonstration workflow with DBSCAN for unsupervised clustering / segmentation of sample data.



Applied Machine Learning in Python: a Hands-on Guide with Code

Machine Learning Concepts
Workflow Construction and Coding
Probability Concepts
Loading and Plotting Data and Models
Univariate Analysis
Multivariate Analysis
Feature Transformations
Feature Ranking
Cluster Analysis
Density-based Clustering
Spectral Clustering

Density-based Clustering

Michael J. Pyrcz, Professor, The University of Texas at Austin

Twitter | GitHub | Website | GoogleScholar | Book | YouTube | Applied Geostats in Python e-book | LinkedIn

Chapter of e-book "Applied Machine Learning in Python: a Hands-on Guide with Code".

Cite this e-Book as:

Pyrcz, M.J., 2024, Applied Machine Learning in Python: a Hands-on Guide with Code, https://geostatsguy.github.io/MachineLearningDemos_Book.

The workflows in this book and more are available here:

Cite the MachineLearningDemos GitHub Repository as:

Pyrcz, M.J., 2024, MachineLearningDemos: Python Machine Learning Demonstration Workflows Repository (0.0.1). Zenodo. DOI 10.5281/zenodo.13835318

By Michael J. Pyrcz
© Copyright 2024.

This is a tutorial for / demonstration of **Density-based Clustering**.

YouTube Lecture: check out my lectures on:

Contents

Motivation for Density-based Cluster Analysis
Clustering Methods Covered
Inferential Machine Learning
k-Means Clustering
Assumptions of k-means Clustering
DBSCAN for Density-based Clustering
DBSCAN Iterative Solution
Load the required libraries
Declare functions
Custom Colormap
Set the working directory
Loading Data
Summary Statistics and Histograms for the Tabular Data
Feature Normalization
Quick Peek at Available Labels for Educational Purposes
Visualize the Unlabeled Data
k-Means Clustering
DBSCAN Clustering
Calculating the DBSCAN Parameters
Comments
The Author:
Want to Work Together?

MachineLearning Density-based Clustering chapter of e-book.

PGE 383 Subsurface Machine Learning

Lecture 7b: Advanced Clustering

Lecture outline:

- **Spectral Clustering**

Spectral Clustering

General Comments:

- Spectral Clustering, utilize the spectrum (eigenvalues) of a matrix that represents the pairwise relationships between the data.

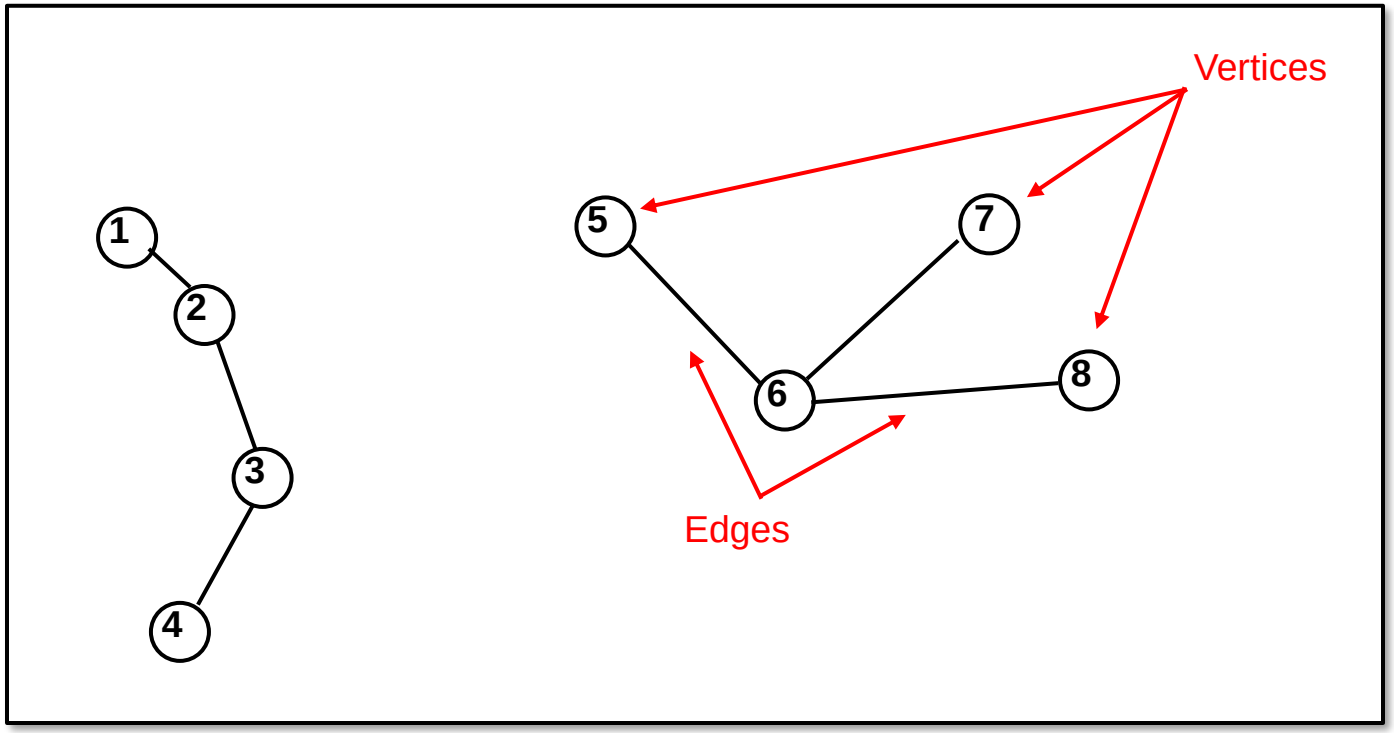
- Dimensionality reduction from data samples pairwise relationships characterized by the graph Laplacian matrix
  - eigenvalues, eigenvectors are equivalent to principal component analysis dimensionality reduction by linear, orthogonal feature projection / rotation to best describe the variance (more in next lecture).

- Advantages:
  - the ability to encode pariwise relationships, integrate expert knowledge.
  - eigenvalues provide useful information on the number of clusters, based on the degree of ‘cutting’ required to make k clusters
  - lower dimensional representation for the sample data pairwise relationships
  - the eigenvalues and vectors can be interpreted!

Graph

A diagram that represents data in an organized manner

- Set of nodes/objects (samples) with vertices (indicating relationships)
- Undirected graph, vertices are bidirectional (connection goes both ways)

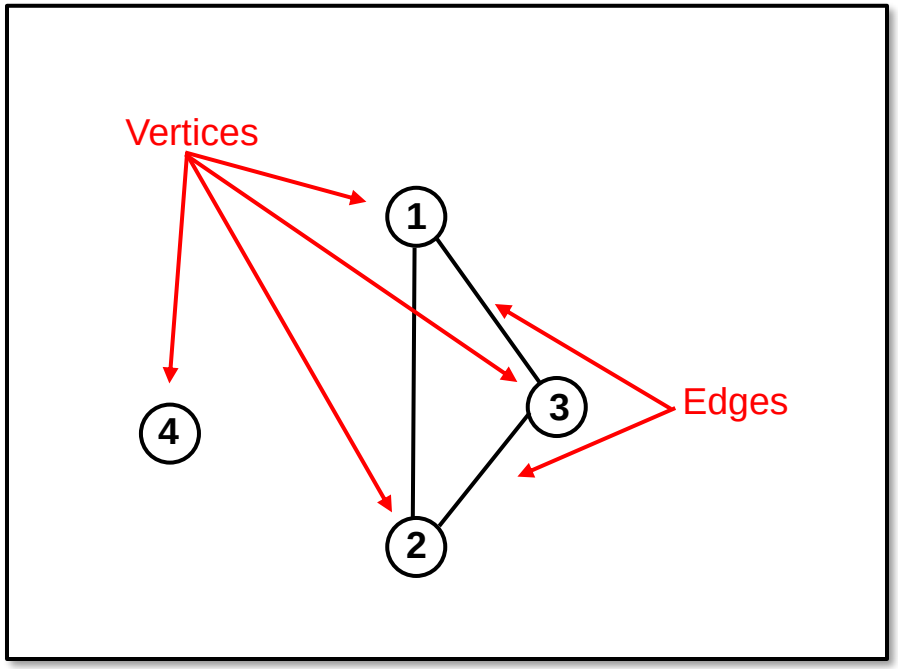


Example Undirected Graph

Graph from the e-book

A diagram that represents data in an organized manner

- Set of nodes/objects (samples) with vertices (indicating relationships)
- Undirected graph, vertices are bidirectional (connection goes both ways)

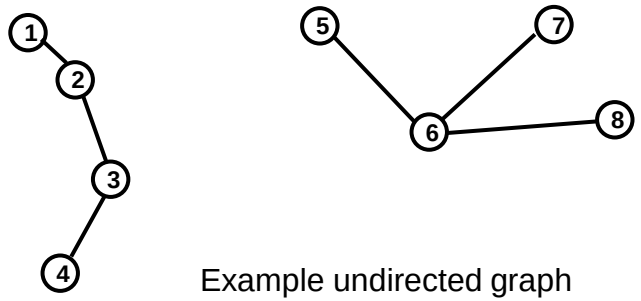


Example Undirected Graph

Adjacency Matrix

We can represent the graph as an adjacency matrix

- A matrix with the pairwise connections between all combinations of nodes
  - 0 if not connected
  - 1 if connected
- Note, node self connections are set to 0



Example undirected graph

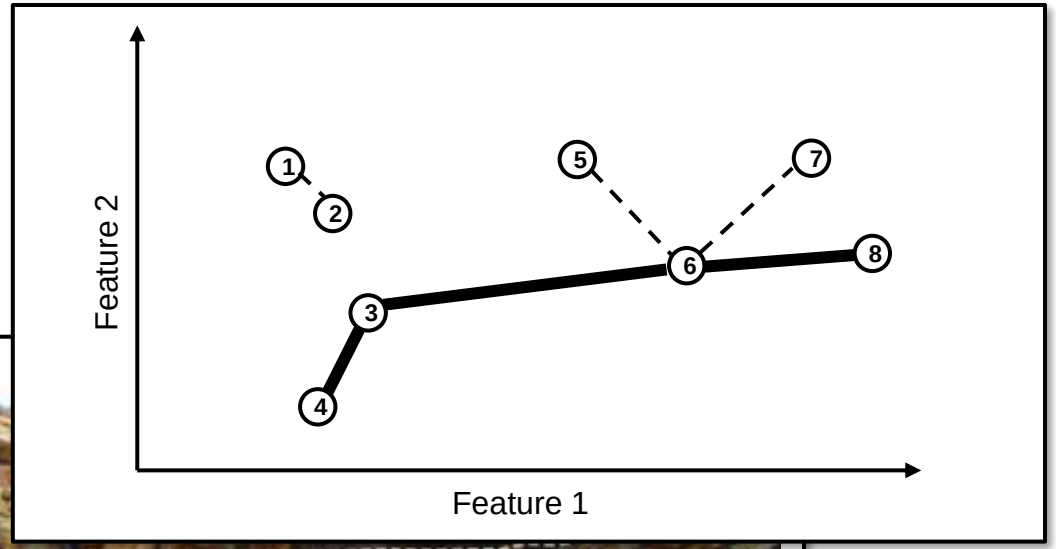
Resulting adjacency matrix

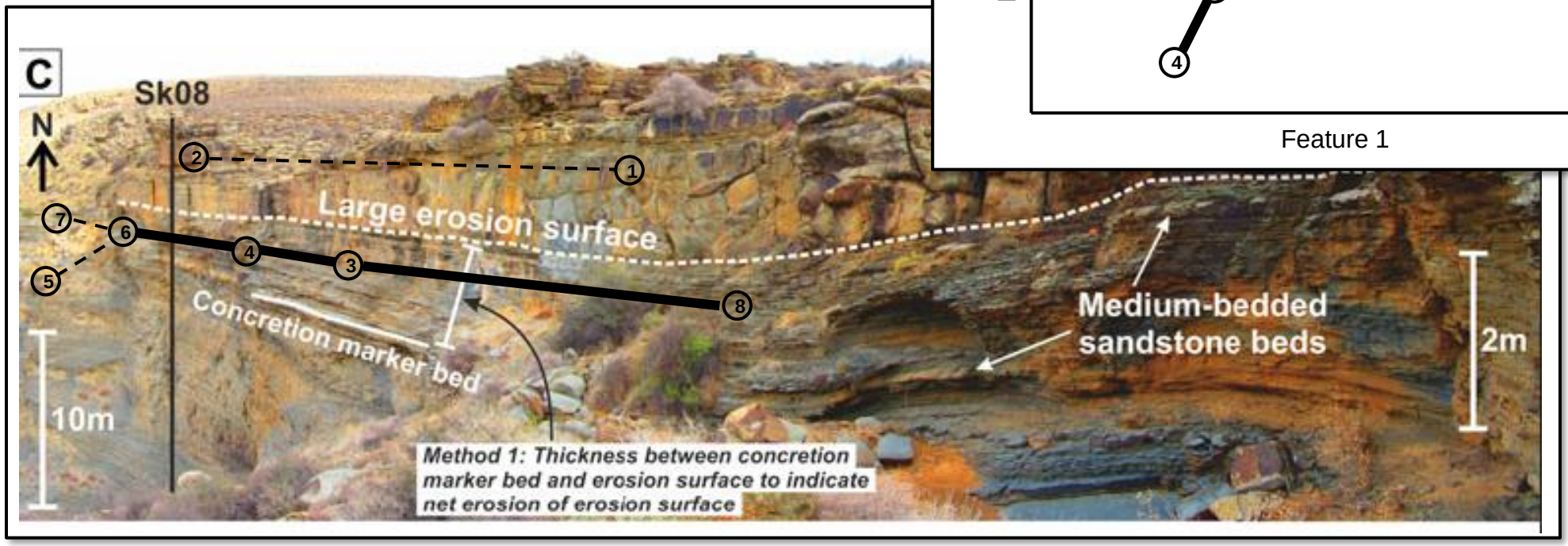
x_2 \ x_1	1	2	3	4	5	6	7	8
1	0	1	0	0	0	0	0	0
2	1	0	1	0	0	0	0	0
3	0	1	0	1	0	0	0	0
4	0	0	1	0	0	0	0	0
5	0	0	0	0	0	1	0	0
6	0	0	0	0	1	0	1	1
7	0	0	0	0	0	1	0	0
8	0	0	0	0	0	1	0	0

Adjacency Matrix

We can represent the graph as an adjacency matrix

- How would we integrate expert knowledge? Here's an example





Data in regular space and with connectivity information
(schematic based on Skoorsteenberg Fm, Karoo Basin, South Africa, Hansen et al., 2021).

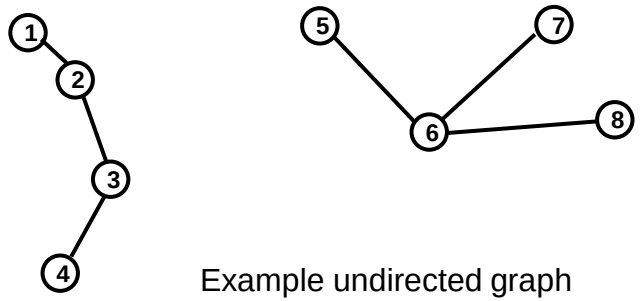
Image from https://www.frontiersin.org/articles/10.3389/feart.2021.737932/full

Degree Matrix

Diagonal Matrix with the 'Amount of Connection' for Each Node

- Number of connections if based on an adjacency matrix

$$D_{i,i} = \sum_{j=1}^{n} A_{i,j}$$



Example undirected graph

Resulting degree matrix

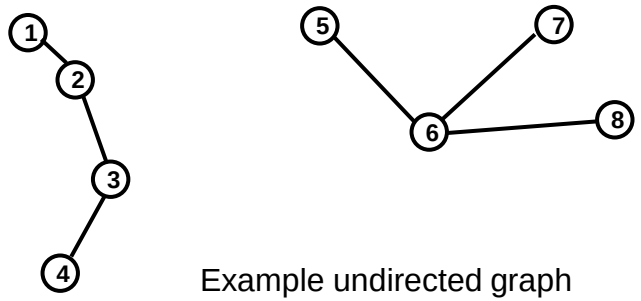
x_2 \ x_1	1	2	3	4	5	6	7	8
1	1	0	0	0	0	0	0	0
2	0	2	0	0	0	0	0	0
3	0	0	2	0	0	0	0	0
4	0	0	0	1	0	0	0	0
5	0	0	0	0	1	0	0	0
6	0	0	0	0	0	3	0	0
7	0	0	0	0	0	0	1	0
8	0	0	0	0	0	0	0	1

Graph Laplacian

Matrix Accounting for

- Degree Matrix, D - (degree of connection for each node) and
- Adjacency Matrix, A - specific connections between nodes

$$L = D - A$$



Example undirected graph

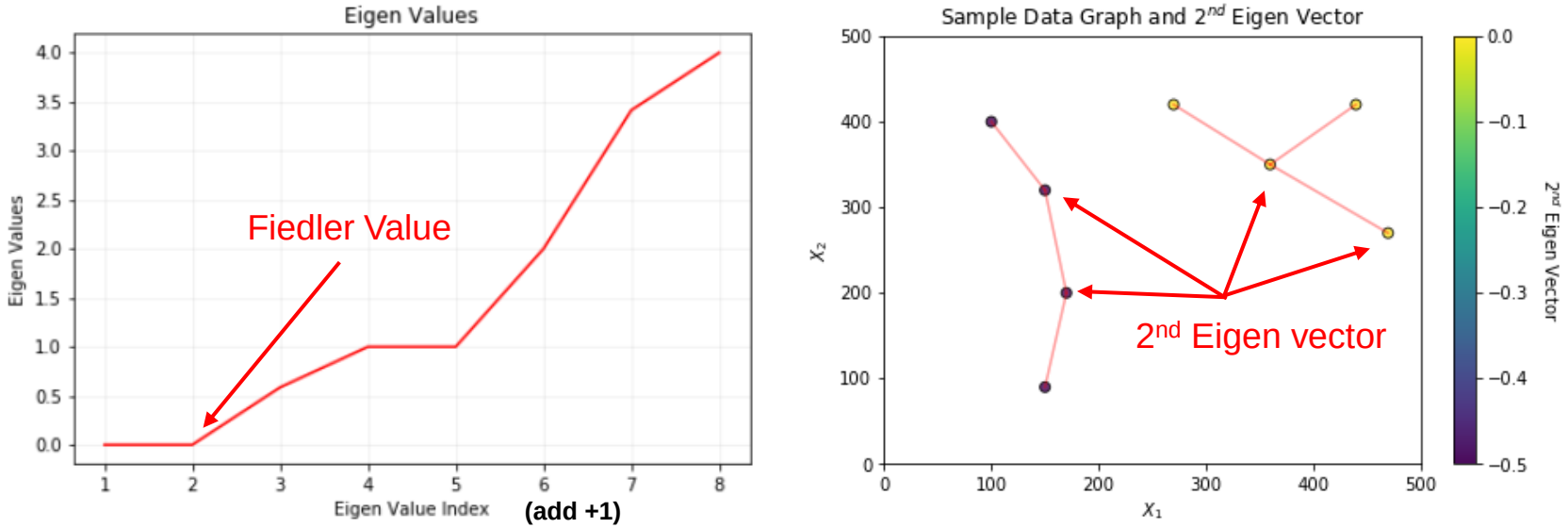
Resulting graph Laplacian

x_2 \ x_1	1	2	3	4	5	6	7	8
1	1	-1	0	0	0	0	0	0
2	-1	2	-1	0	0	0	0	0
3	0	-1	2	-1	0	0	0	0
4	0	0	-1	1	0	0	0	0
5	0	0	0	0	1	-1	0	0
6	0	0	0	0	-1	3	-1	-1
7	0	0	0	0	0	-1	1	0
8	0	0	0	0	0	-1	0	1

Eigenvalues and Eigenvectors of the Laplacian

Calculate the Eigenvectors and Associated Eigenvalues

- Sort eigenvectors in order of ascending eigenvalues
- Number of first nonzero eigenvalue -1 is the number of connected parts



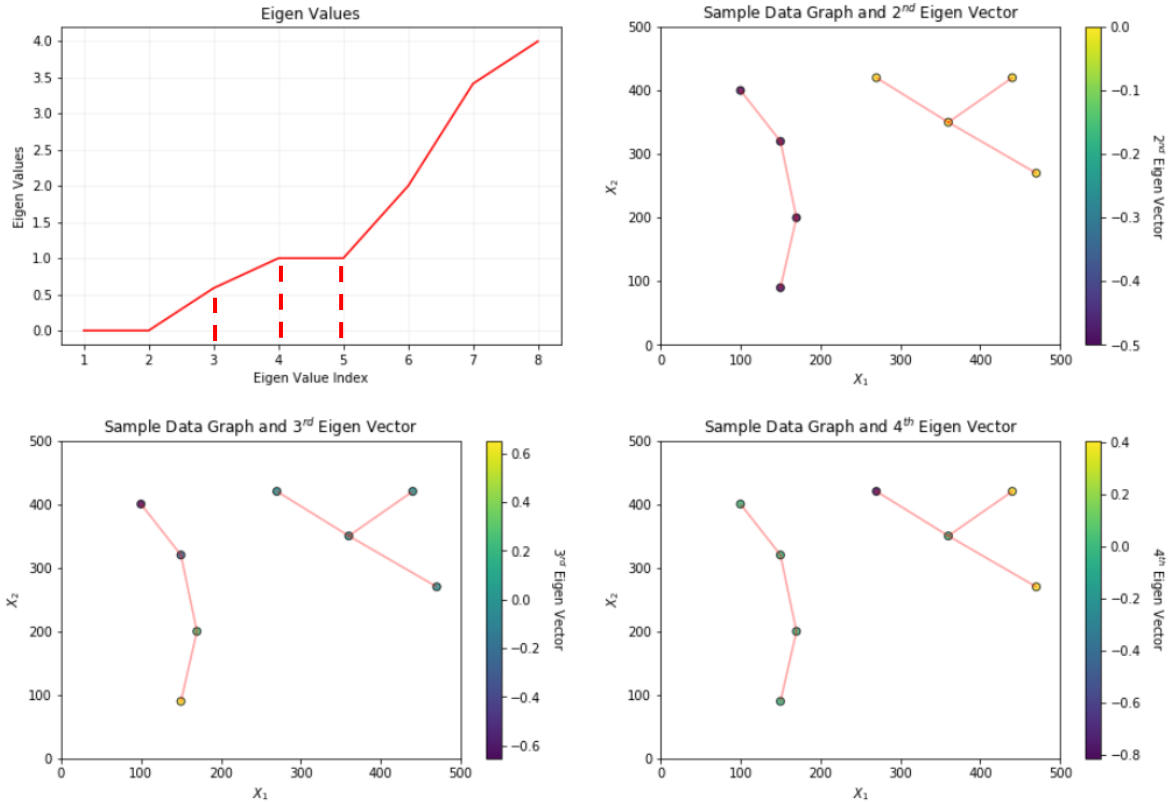
Sorted Eigenvalue plot and sample data, connections and 2nd Eigenvectors plotted.

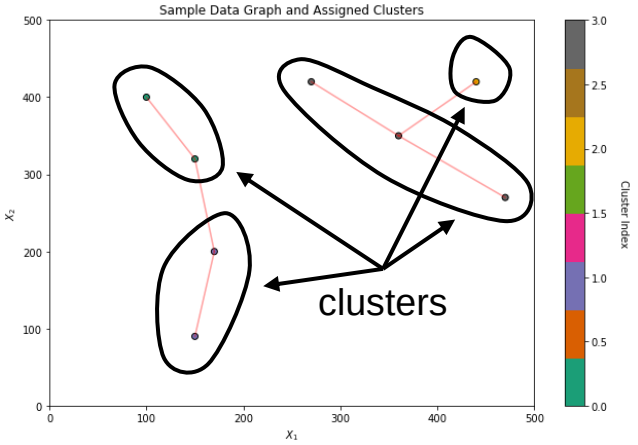
- 1st nonzero eigenvalue is spectral gap – density of connections, fully connected would be equal to n
- 2nd eigenvalue is Fiedler value, ‘amount of cut’ to make 2 groups, etc.
- 2nd eigenvector can be used directly to specify 2 clusters!

Eigenvalues and Eigenvectors of the Laplacian

Apply k-means Clustering on the Eigenvectors (up to the k number of clusters)

- For k = 2, cluster by the 2nd Eigenvector below
- For k = 3, cluster by the 2nd and 3rd Eigenvector below



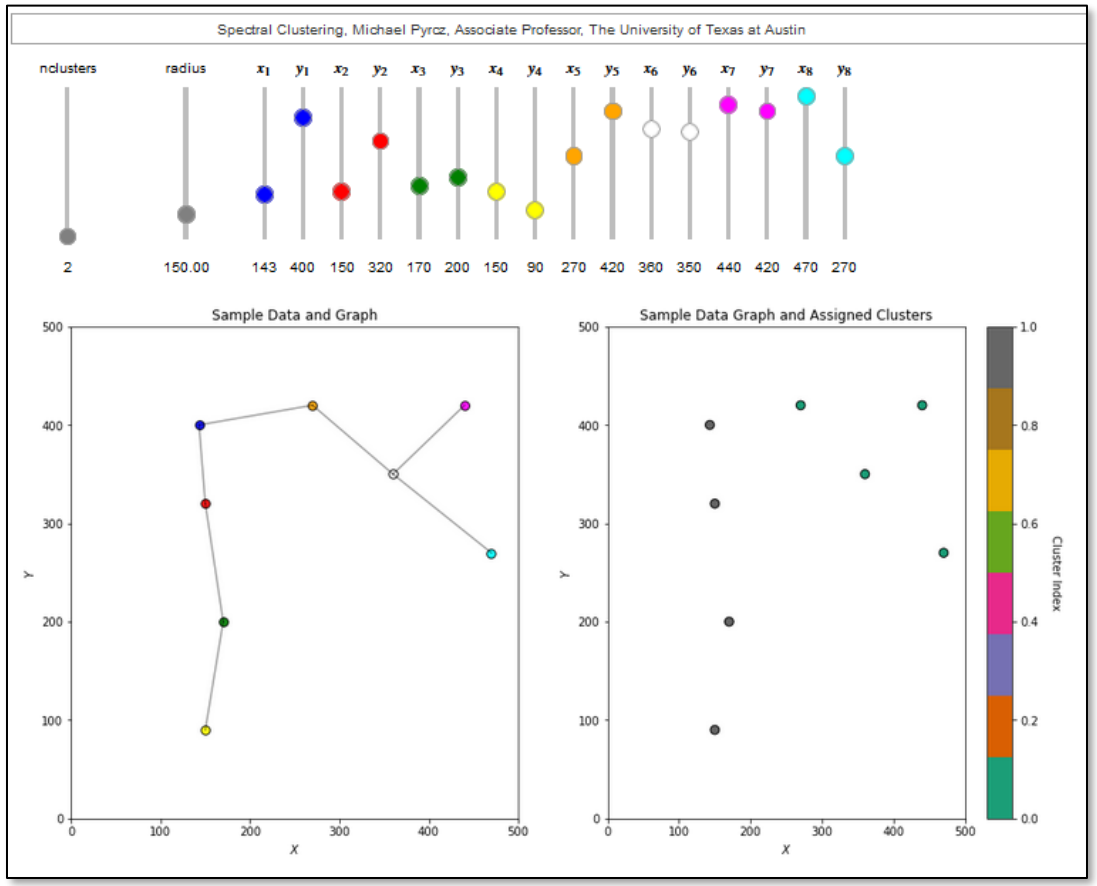


Sorted Eigen values, plotted Eigen vectors and k-means clusters of 2nd, 3rd and 4th Eigen vectors for cluster assignment.

Spectral Clustering Hands-on

Try out interactive spectral clustering for automatic categorical assignment, clustering.

Note, adjacency based on maximum separation distance.



Interactive spectral clustering dashboard, file is Interactive_Spectral_Clustering.ipynb.

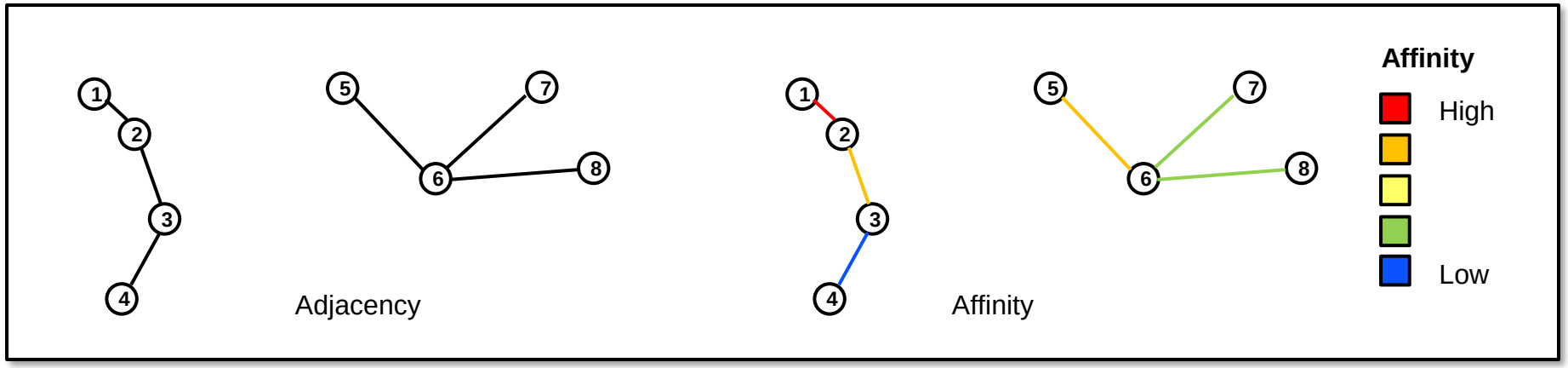
Adjacency and Affinity

Adjacency methods

- Radius, used in my interactive demonstration
- K-nearest neighbors

Affinity instead of adjacency matrix

- Indicate the degree of connection instead of binary indicator, yes (1) or no (0)
- Radial basis function – smooth isotropic function
- Precomputed matrix
- Precomputed function, i.e., $Affinity = f(Pairwise\ Distance)$



Schematic of adjacency vs. affinity.

PGE 383 Subsurface Machine Learning

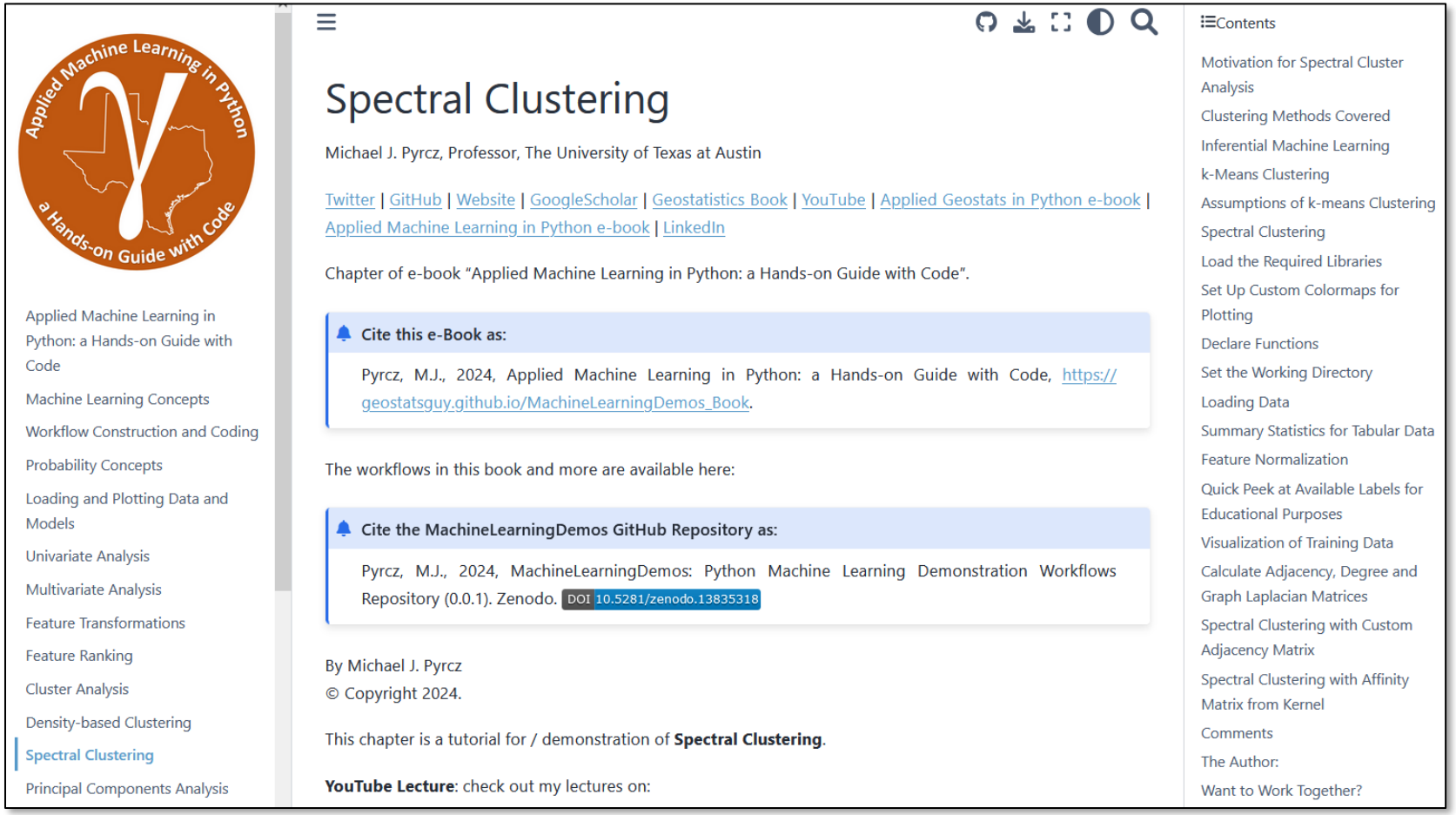
Lecture 7b: Advanced Clustering

Lecture outline:

- **Spectral Clustering Hands-on**

Spectral Clustering Hands-on

Demonstration workflow with spectral clustering for unsupervised clustering / segmentation of sample data.



MachineLearning Density-based Clustering chapter of e-book.

PGE 383 Subsurface Machine Learning

Lecture 7b: Advanced Clustering

Lecture outline:

- **Centroid-based Clustering**
- **Density-based Clustering**
- **Density-based Clustering Hands-on**
- **Spectral Clustering**
- **Spectral Clustering Hands-on**