GENERAL ASSEMBLY

UNSUPERVISED LEARNING

Insert Instructor Name

Title, Company

LEARNING OBJECTIVES

- Understanding the difference between supervised and unsupervised learning.
- Understand clustering techniques k-means, hierarchical and DBSCAN clustering
- Apply those algorithms with sklearn

COURSE

PRE-WORK

PRE-WORK REVIEW

- Understand results from a confusion matrix and measure true positive rate and false positive rate
- Create and interpret results from a binary classification problem
- Know what a decision line is in logistic regression

OPENING

UNSUPERVISED LEARNING

SUPERVISED LEARNING

- All what we covered so far is considered supervised learning
- This includes Regressions and Classification techniques.
- It requires the data to be labeled
- Discover patterns in the data that relate data attributes with a target (class) attribute.

SUPERVISED LEARNING

- These patterns are then utilized to predict the values of the target attribute for future data instances.

- We can measure the accuracy of the model using different techniques (metrics) which will help us to satisfy a success criteria.

UNSUPERVISED LEARNING

- The data have no target attribute.
- We want to explore the data to find some intrinsic structures in them.
- Unsupervised learning is often much more challenging.
- It is hard to assess the results obtained from unsupervised learning methods because there is no true answer.

UNSUPERVISED LEARNING

- Unsupervised learning techniques can be used in several fields:
  - An online shopping site might try to identify groups of shoppers with similar browsing and purchase histories
  - Also identify items that are of particular interest to the shoppers within each group
  - Visualize high dimensional data
  - Fraud detection and Fault detection

ACTIVITY: KNOWLEDGE CHECK

EXERCISE

ANSWER THE FOLLOWING QUESTIONS

1. Can you think of any other use cases for unsupervised learning?

DELIVERABLE

Answers to the above question

UNSUPERVISED LEARNING TECHNIQUES

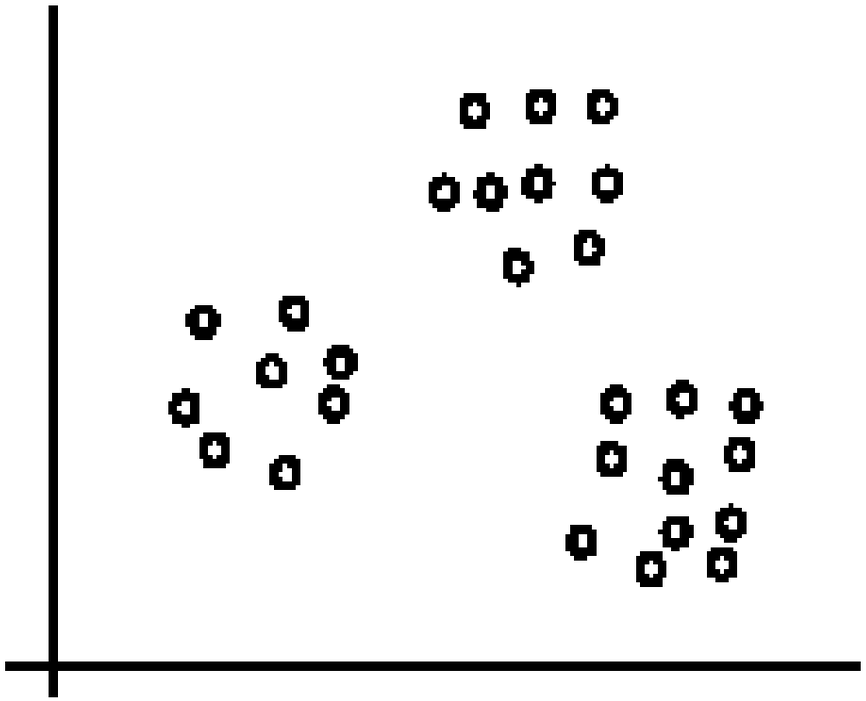
- Unsupervised learning techniques are becoming more interesting.
- Currently the following techniques are used:
  - Clustering:
    - K-Means
    - Hierarchical Clustering
    - DBSCAN Clustering
    - Etc..
  - Dimension reduction (t-SNE, PCA)

INTRODUCTION

CLUSTERING

CLUSTERING

- Clustering is a technique for finding similarity groups in data, called clusters.
- It groups data instances that are similar to (near) each other in one cluster and data instances that are very different (far away) from each other into different clusters.



CLUSTERING

- Types of clustering:
  - **Partitional Clustering:** is simply a division of the set of data objects into non-overlapping subsets (clusters) such that each data object is in exactly one subset.

  - **Hierarchical Clustering:** is a set of nested clusters that are organized as a tree.

CLUSTERING

- Real life examples:
  - Example 1: groups people of similar sizes together to make “small”, “medium” and “large” T-Shirts.
  - Example 2: In marketing, segment customers according to their similarities
  - Example 3: Given a collection of text documents, we want to organize them according to their content similarities

CLUSTERING

- Clustering depends on the following:
  - Clustering algorithm: whether K-means, Hierarchical Clustering, etc..
  - Distance (similarity, or dissimilarity) function
  - Clustering quality:
    - Inter-clusters distance → maximized
    - Intra-clusters distance → minimized

K-MEANS

K-MEANS

- K-means is a partitional clustering algorithm.
- Let the set of data points (or instances) D be {x1, x2, …, xn}, where xi = (xi1, xi2, …, xir) is a vector in a real-valued space and r is the number of attributes (dimensions) in the data.
- The *k*-means algorithm partitions the given data into *k* clusters.
  - Each cluster has a cluster center, called centroid
  - k is specified by the user

K-MEANS ALGORITHM

- Given k, the k-means algorithm works as follows:
  - Randomly choose k data points (seeds) to be the initial centroids, cluster centers from the dataset
  - Assign each data point to the closest centroid
  - Re-compute the centroids using the current cluster memberships.
  - If a convergence criterion is not met, iterate again

K-MEANS ALGORITHM

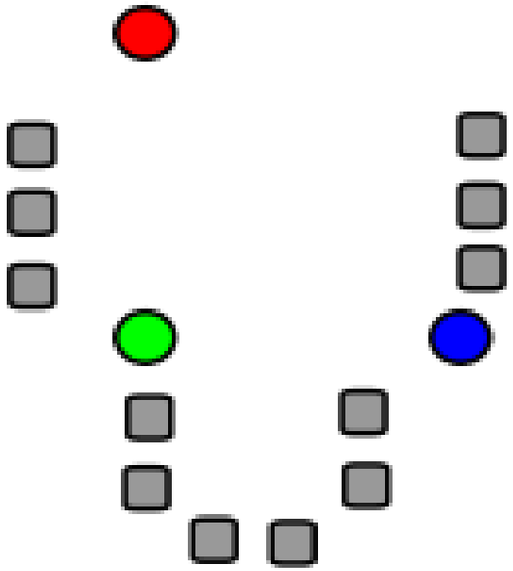
- Stopping/convergence criterion:
  - no (or minimum) re-assignments of data points to different clusters
  - no (or minimum) change of centroids, or
  - minimum decrease in the sum of squared error (SSE)

$$SSE = \sum_{j=1}^{k} \sum_{\mathbf{x} \in C_j} dist(\mathbf{x}, \mathbf{m}_j)^2$$

  - Ci is the jth cluster, mj is the centroid of cluster Cj (the mean vector of all the data points in Cj), and dist(x, mj) is the distance between data point x and centroid mj.

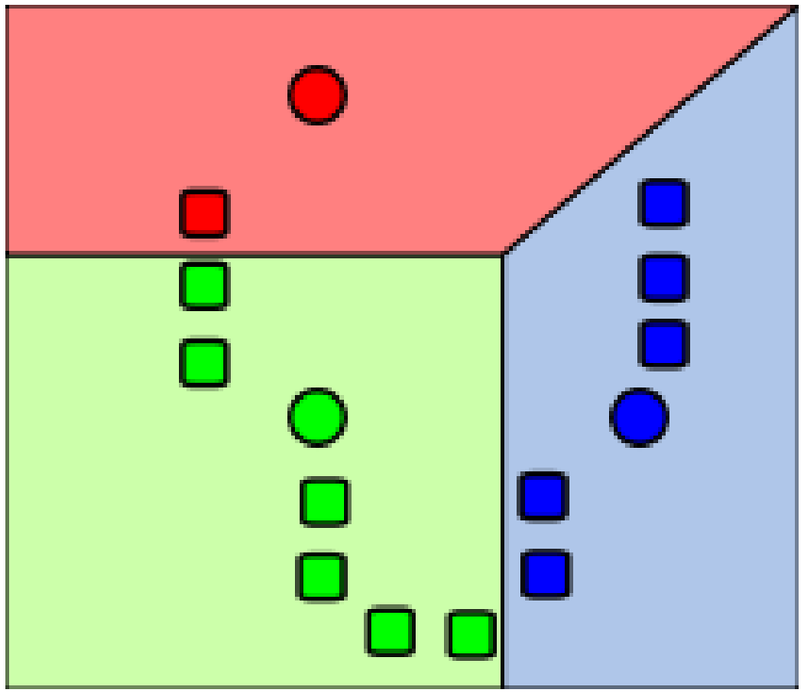
K-MEANS EXAMPLE

- k initial "means" (in this case k=3) are randomly generated within the data domain (shown in color).



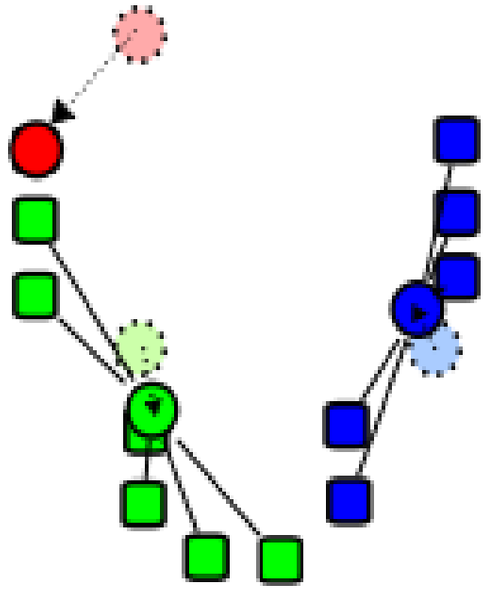
K-MEANS EXAMPLE

- k clusters are created by associating every observation with the nearest mean. The calculation will be based on calculating the distance from the mean



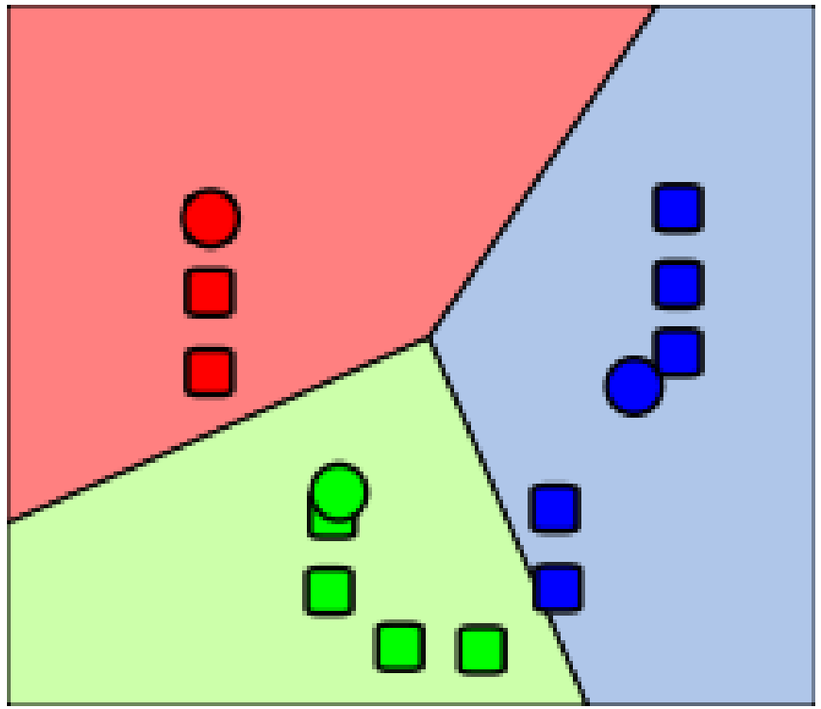
K-MEANS EXAMPLE

- The centroid of each of the k clusters becomes the new mean.



K-MEANS EXAMPLE

- Steps 2 and 3 are repeated until convergence has been reached



K-MEANS ASSUMPTIONS

- Assumptions are important! k-Means assumes:
  - k is the correct number of clusters
  - Data is isotropically distributed (circular/spherical distribution)
  - Variance is the same for each variable
  - clusters are roughly the same size

- Nice counterexamples / cases where assumptions are not met:
  - http://varianceexplained.org/r/kmeans-free-lunch/
  - Scikit-Learn Examples

K-MEANS STRENGTHS

- Simple: easy to understand and to implement

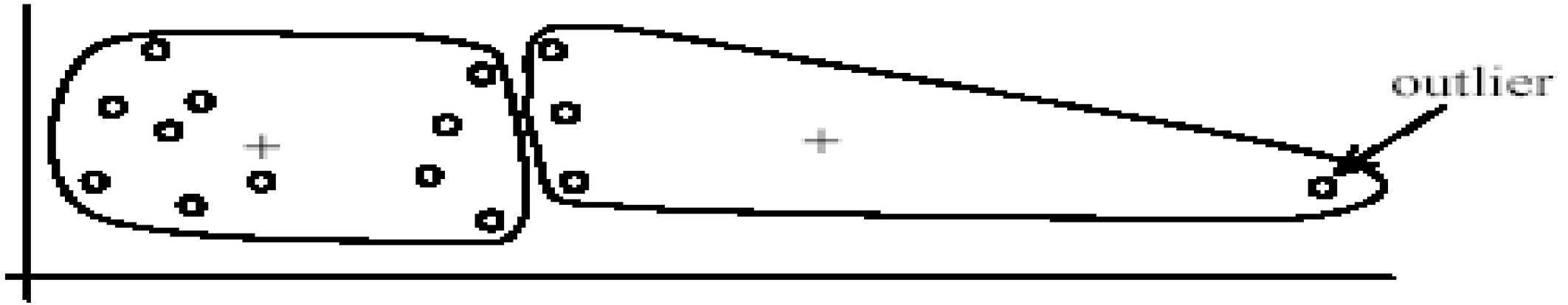
- Efficient in terms of execution: Time complexity is O(tkn) where:
  - n is the number of data points
  - k is the number of clusters
  - t is the number of iterations.

- K-Means may produce tighter clusters than other methods

K-MEANS WEAKNESSES

- The number of clusters (which is K) must be determined before hand.
- The algorithm is sensitive to outliers as it depends on calculating the distance from the cluster mean or centroid. Very far data from the centroid may pull the centroid away from the real one.
- Different initial partitions can result in different final clusters

K-MEANS WEAKNESSES



(A): Undesirable clusters

(B): Ideal clusters

K-MEANS WEAKNESSES

- To deal with outliers:
  - One method is to remove some data points in the clustering process that are much further away from the centroids than other data points. We have to be careful in this solution as we might need to monitor those outliers points before we decide to exclude them
  - Another method is to perform sampling (random, etc..) when we have large dataset where we can use samples to generate the clusters. Sampling will reduce the chance of picking up those outliers.

DEMO

K-MEANS PRACTICE WITH SKLEARN

K-MEANS PRACTICE WITH SKLEARN

- Code snippet:

```
import pandas as pd
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt

iris = datasets.load_iris()
irisdf = pd.DataFrame(iris.data, columns=iris.feature_names)
kmeans = KMeans(n_clusters=3, random_state=0).fit(iris.data[:,2:])
```

INTRODUCTION

DBSCAN CLUSTERING

DBSCAN CLUSTERING

- DBSCAN: Density-based spatial clustering of applications with noise (1996)

- Main idea: Group together closely-packed points by identifying
  - Core points
  - Reachable points
  - Outliers (not reachable)

- Two parameters:
  - min_samples
  - eps

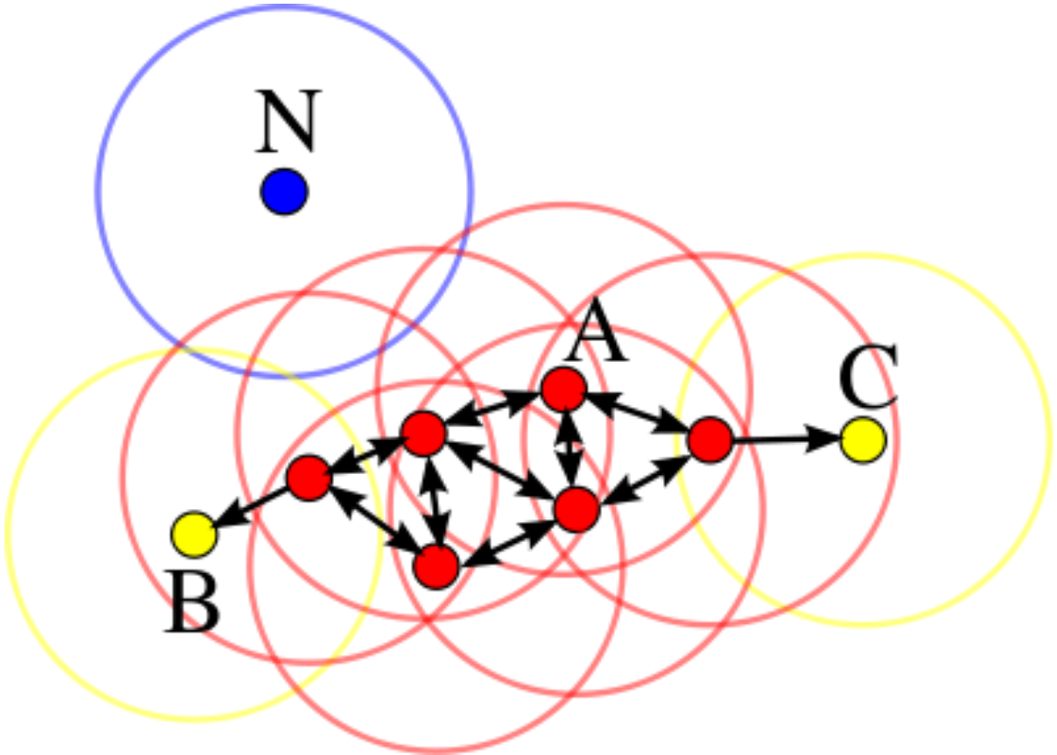
DBSCAN CLUSTERING

- Core points: at least min_samples points within eps of the core point
  - Such points are directly reachable from the core point

- Reachable: point q is reachable from p if there is a path of core points from p to q

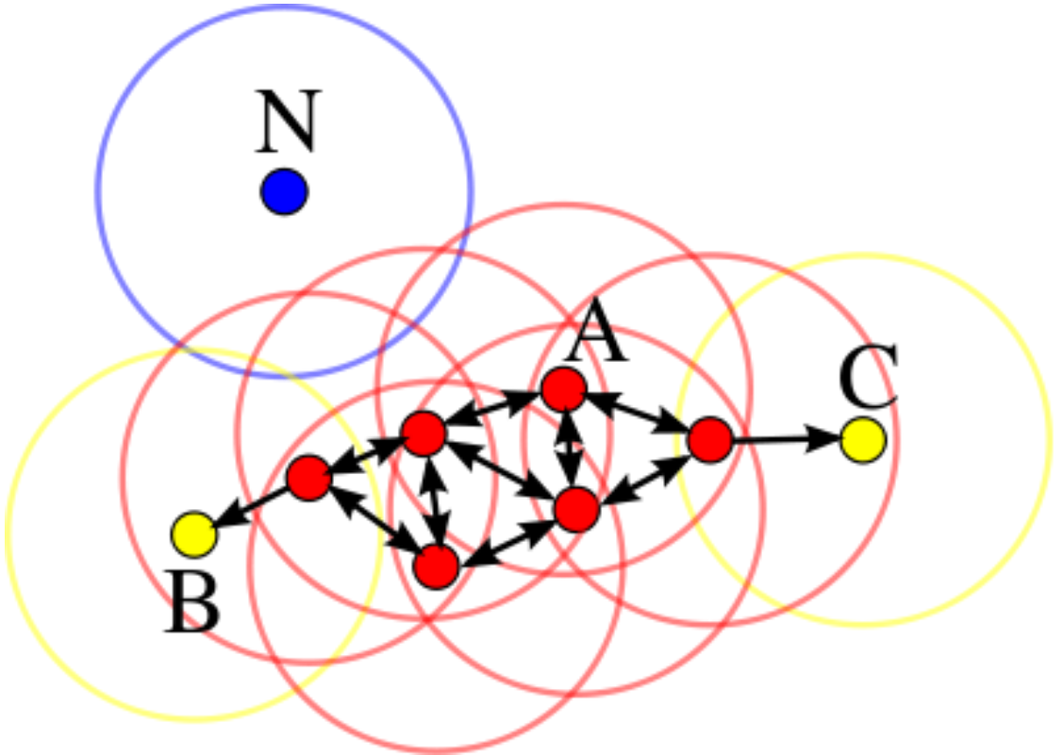
- Outlier: not reachable

- In this example:
  - minPts = 4



DBSCAN CLUSTERING

- A cluster is a collection of connected core and reachable points
- DBSCAN Demo



DBSCAN CLUSTERING

- DBSCAN advantages:
  - Can find arbitrarily-shaped clusters
  - Don’t have to specify number of clusters
  - Robust to outliers

- DBSCAN disadvantages:
  - Doesn’t work well when clusters are of varying densities
  - Hard to chose parameters that work for all clusters
  - Can be hard to chose correct parameters regardless

DEMO

DBSCAN CLUSTERING

DBSCAN PRACTICE WITH SKLEARN

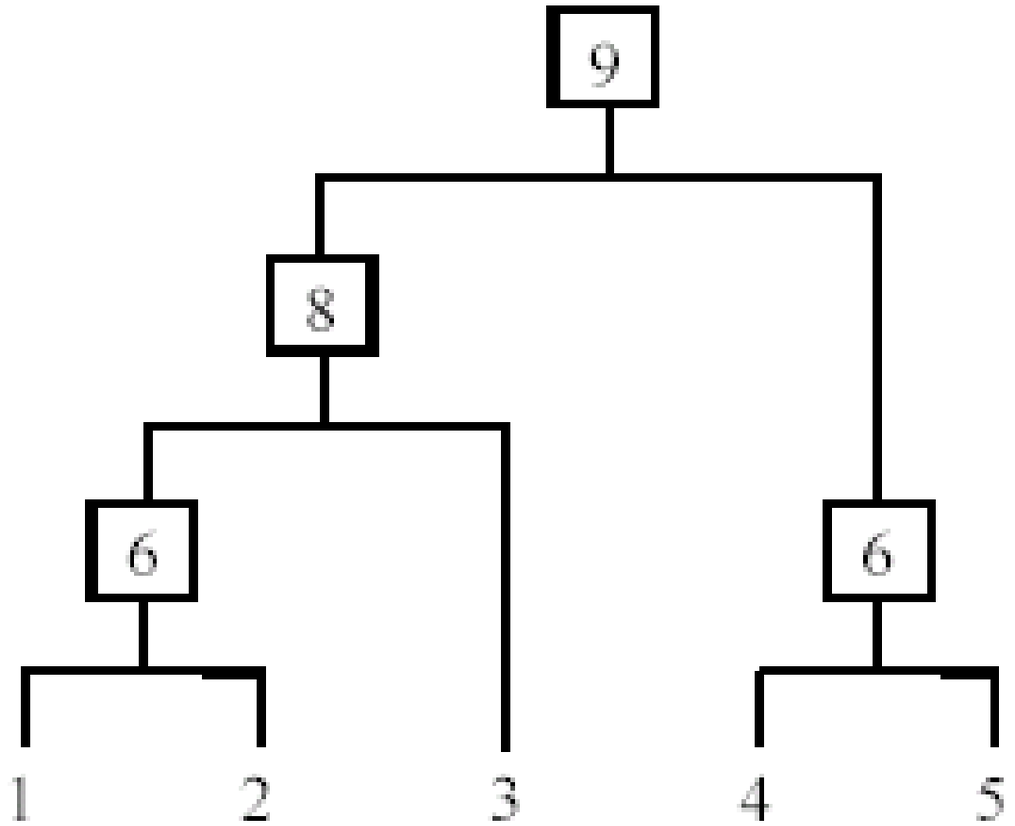
We can fit the model with sklearn

- from sklearn.cluster import DBSCAN
- est = DBSCAN(eps=0.5, min_samples=10)
- est.fit(X)
- labels = est.labels_

HIERARCHICAL CLUSTERING

HIERARCHICAL CLUSTERING

- Produce a nested sequence of clusters, a tree, also called Dendrogram.



HIERARCHICAL CLUSTERING

- Types of hierarchical clustering:
- Agglomerative (bottom up) clustering: It builds the dendrogram (tree) from the bottom level
  - Merges the most similar (or nearest) pair of clusters
  - Stops when all the data points are merged into a single cluster (i.e., the root cluster)

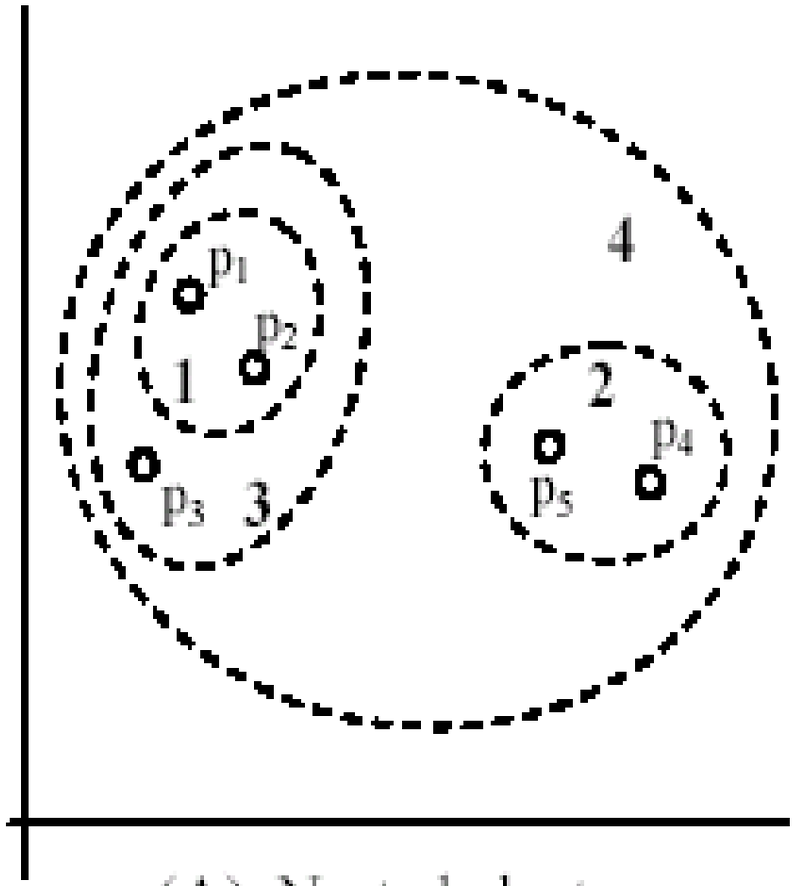
HIERARCHICAL CLUSTERING

- Types of hierarchical clustering:
- Divisive (top down) clustering: It starts with all data points in one cluster, the root
  - Splits the root into a set of child clusters. Each child cluster is recursively divided further
  - Stops when only singleton clusters of individual data points remain, i.e., each cluster with only a single point

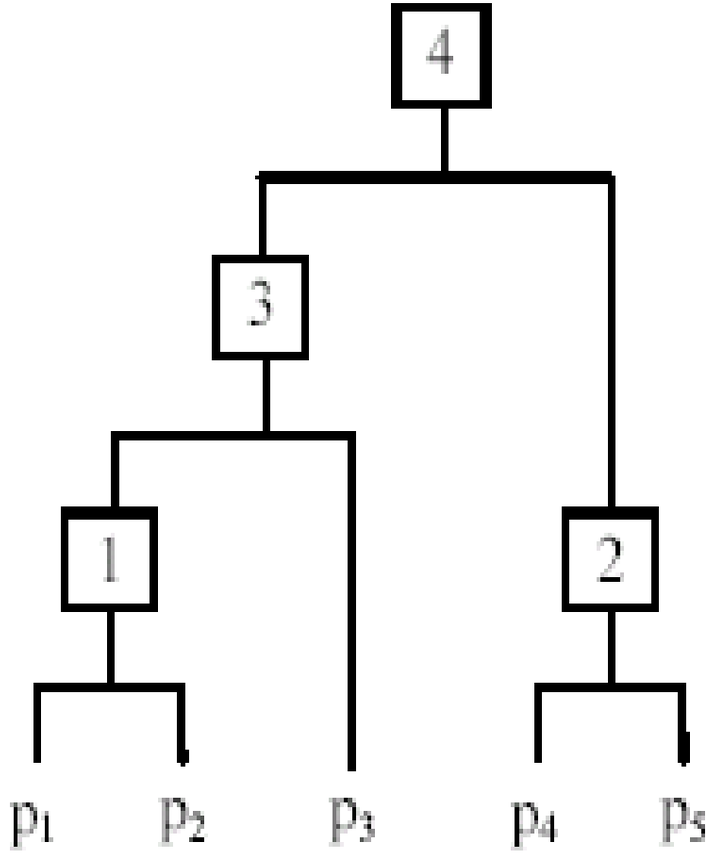
AGGLOMERATIVE CLUSTERING

- It is more popular then divisive methods
- At the beginning, each data point forms a cluster (also called a node)
- Merge nodes/clusters that have the least distance
- Go on merging
- Eventually all nodes belong to one cluster

AGGLOMERATIVE CLUSTERING EXAMPLE



(A). Nested clusters



(B) Dendrogram

MEASURING DISTANCE OF TWO CLUSTERS

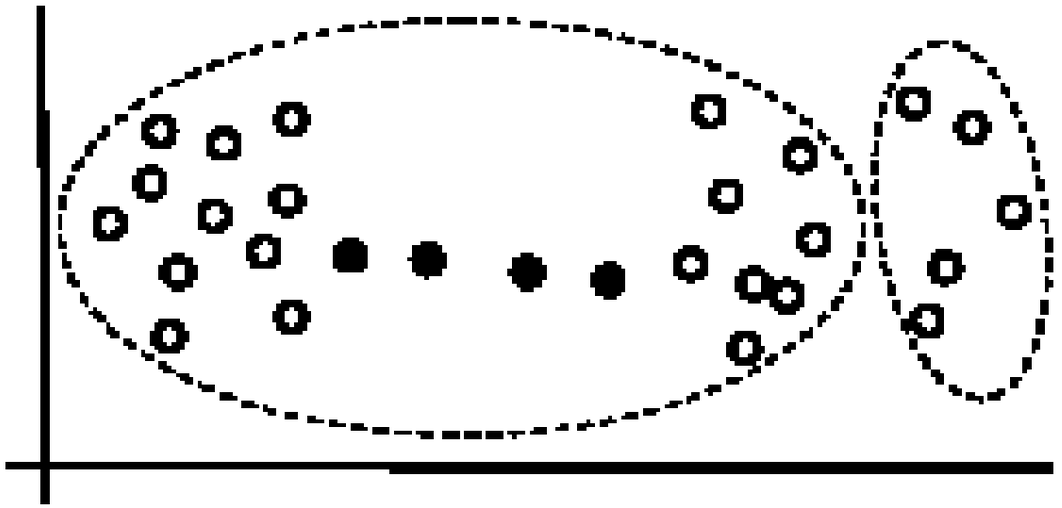
- A few ways to measure distances of two clusters.

- Results in different variations of the algorithm:
  - Single link
  - Complete link
  - Average link
  - Centroids
  - Etc..

SINGLE LINK

- The distance between two clusters is the distance between two closest data points in the two clusters, one data point from each cluster.

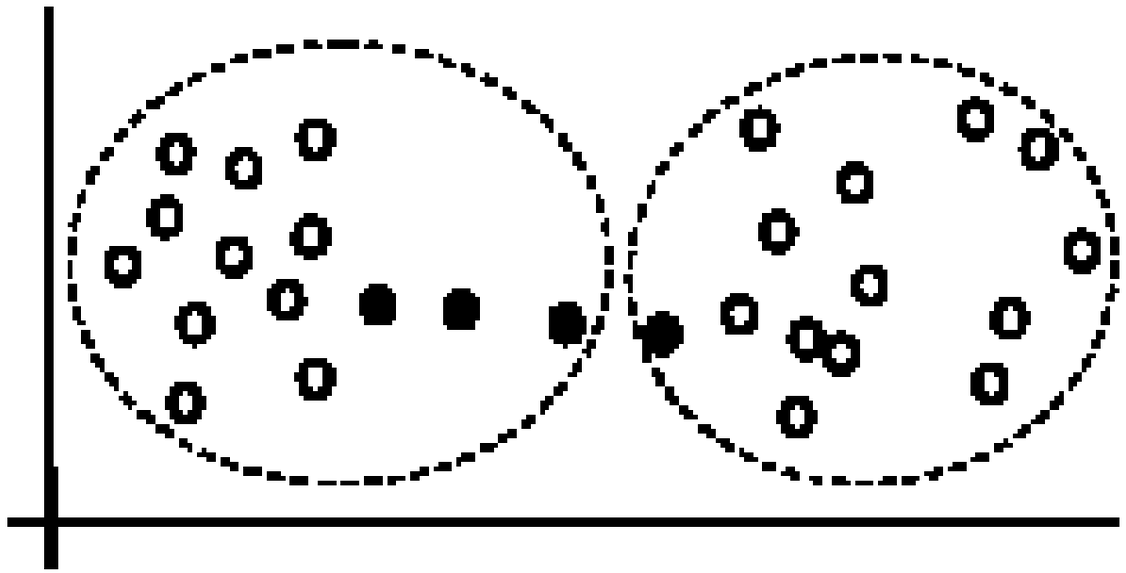
- It can find arbitrarily shaped clusters, but It may cause the undesirable “chain effect” by noisy points



COMPLETE LINK

- The distance between two clusters is the distance of two furthest data points in the two clusters.

- It is sensitive to outliers because they are far away



AVERAGE LINK VS CENTROID METHODS

- Average link: A compromise between
  - the sensitivity of complete-link clustering to outliers and
  - the tendency of single-link clustering to form long chains that do not correspond to the intuitive notion of clusters as compact, spherical objects.
  - In this method, the distance between two clusters is the average distance of all pair-wise distances between the data points in two clusters.
- Centroid method: In this method, the distance between two clusters is the distance between their centroids

DEMO

HIERARCHICAL CLUSTERING

HIERARCHICAL PRACTICE WITH SKLEARN

We can fit the model with sklearn

- from sklearn.cluster import AgglomerativeClustering
- est = AgglomerativeClustering(n_clusters=4)
- est.fit(X)
- labels = est.labels_

DEMO

CHOOSING K IN K-MEANS AND HIERARCHAL

CHOOSING K AND BEST MODEL

- There are different method to determine which k to be chosen
  - The elbow method

$$D_k = \sum_{x_i \in C_k} \sum_{x_j \in C_k} ||x_i - x_j||^2 = 2n_k \sum_{x_i \in C_k} ||x_i - \mu_k||^2$$

$$W_k = \sum_{k=1}^{K} \frac{1}{2n_k} D_k$$

  - The silhouette method

$$s = \frac{b - a}{max(a, b)}$$

CHOOSING CLUSTERING ALGORITHM

CHOOSING CLUSTERING ALGORITHM

- Clustering research has a long history. A vast collection of algorithms are available. We have introduced three algorithms out of them.

- Choosing the “best” algorithm is a challenge.
  - Every algorithm has limitations and works well with certain data distributions.

  - It is very hard, if not impossible, to know what distribution the application data follow. The data may not fully follow any “ideal” structure or distribution required by the algorithms.

  - One also needs to decide how to standardize the data, to choose a suitable distance function and to select other parameter values.

CHOOSING CLUSTERING ALGORITHM

- Due to these complexities, the common practice is to :
  - run several algorithms using different distance functions and parameter settings, and
  - then carefully analyze and compare the results.
- The interpretation of the results must be based on insight into the meaning of the original data together with knowledge of the algorithms used.
- Clustering is highly application dependent and to certain extent subjective (personal preferences).

PROJECT PRACTICE

ACTIVITY: PROJECT PRACTICE

DIRECTIONS (30 minutes)

Using the data set from either titanic or flights delays. Build a model that uses different features to cluster the dataset:

1- Try to build the model using k-means and Hierarchal Clustering

2- Try to change the number of clusters and see the impact on the results

DELIVERABLE

New models and performance statement

CONCLUSION

TOPIC REVIEW

REVIEW AND NEXT STEPS

- Supervised vs Unsupervised learning
- Clustering algorithms including K-means and Hierarchal Clustering
- Applying clustering techniques with sklearn

BEFORE NEXT CLASS

BEFORE NEXT CLASS

UPCOMING

- Project: Unit Project 4

LESSON

CREDITS

THANKS FOR THE FOLLOWING

CITATIONS

- Title, Author: link
- Title, Author: link
- Title, Author: link

LESSON

Q & A

LESSON

EXIT TICKET

DON’T FORGET TO FILL OUT YOUR EXIT TICKET

THANKS!

NAME

- Optional Information:
- Email?
- Website?
- Twitter?