ML: Clustering

CPE 232: Data Models

Dr. Sansiri Tarnpradab

Department of Computer Engineering, KMUTT

Announcement

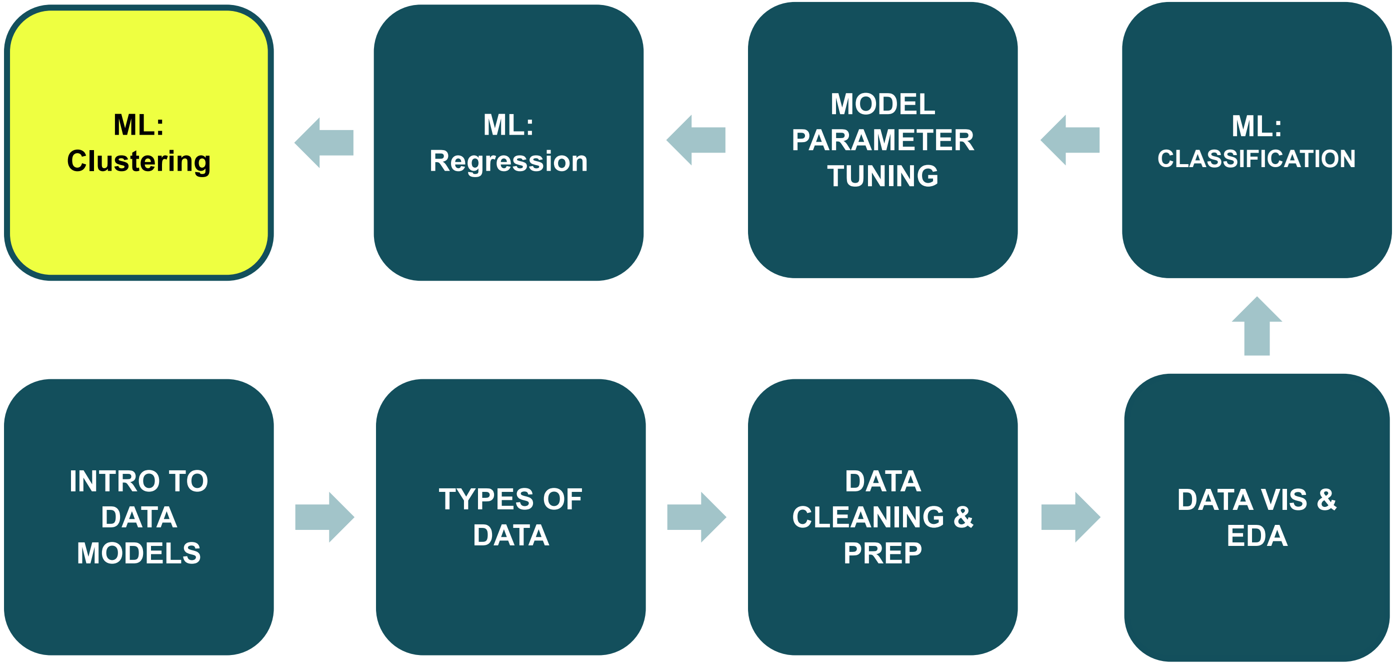
- No class for the next 3 weeks

- Class resumes on 23/4

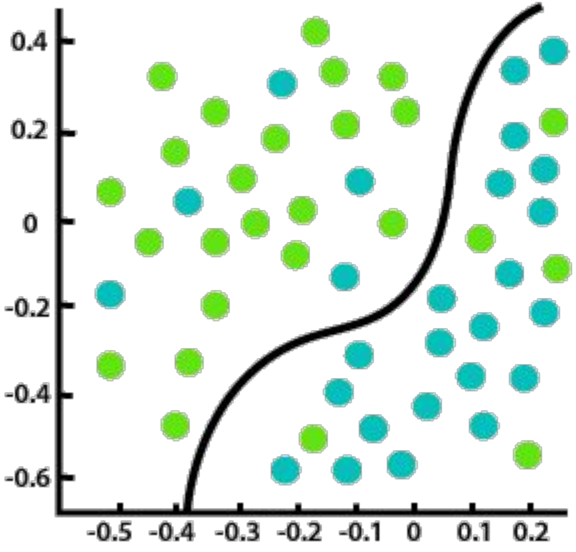
- One-page progress report also due 23/4
    - Method
    - Challenges
    - Results (preliminary to final)
    - Workload

- Proposal: If there's any change, update to LEB by (5/4)

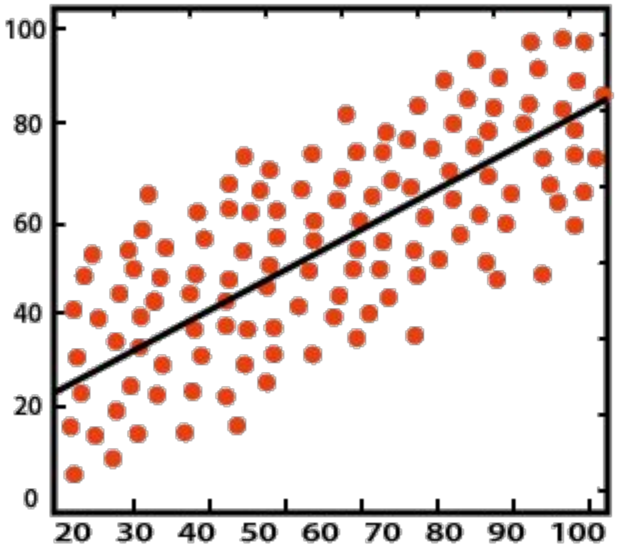
Review


INTRO TO DATA MODELS
TYPES OF DATA
DATA CLEANING & PREP
DATA VIS & EDA
ML: CLASSIFICATION
MODEL PARAMETER TUNING
ML: Regression
ML: Clustering

So far…



Classification



Regression

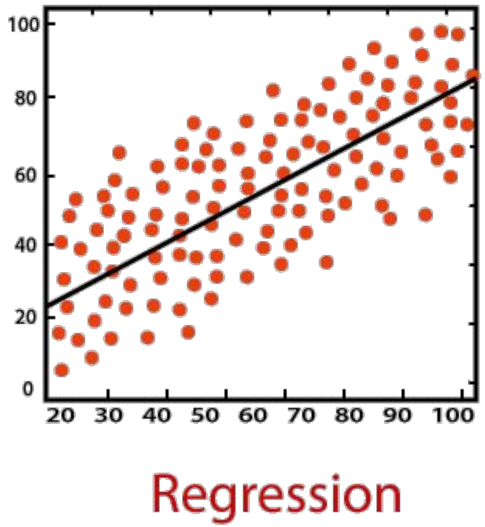
Ref: https://www.javatpoint.com/regression-vs-classification-in-machine-learning

Prediction

Numerical values

Time-series

Relationship



Data = Model + Error

Independent variables (x)
Aka Predictors

Linear Regression

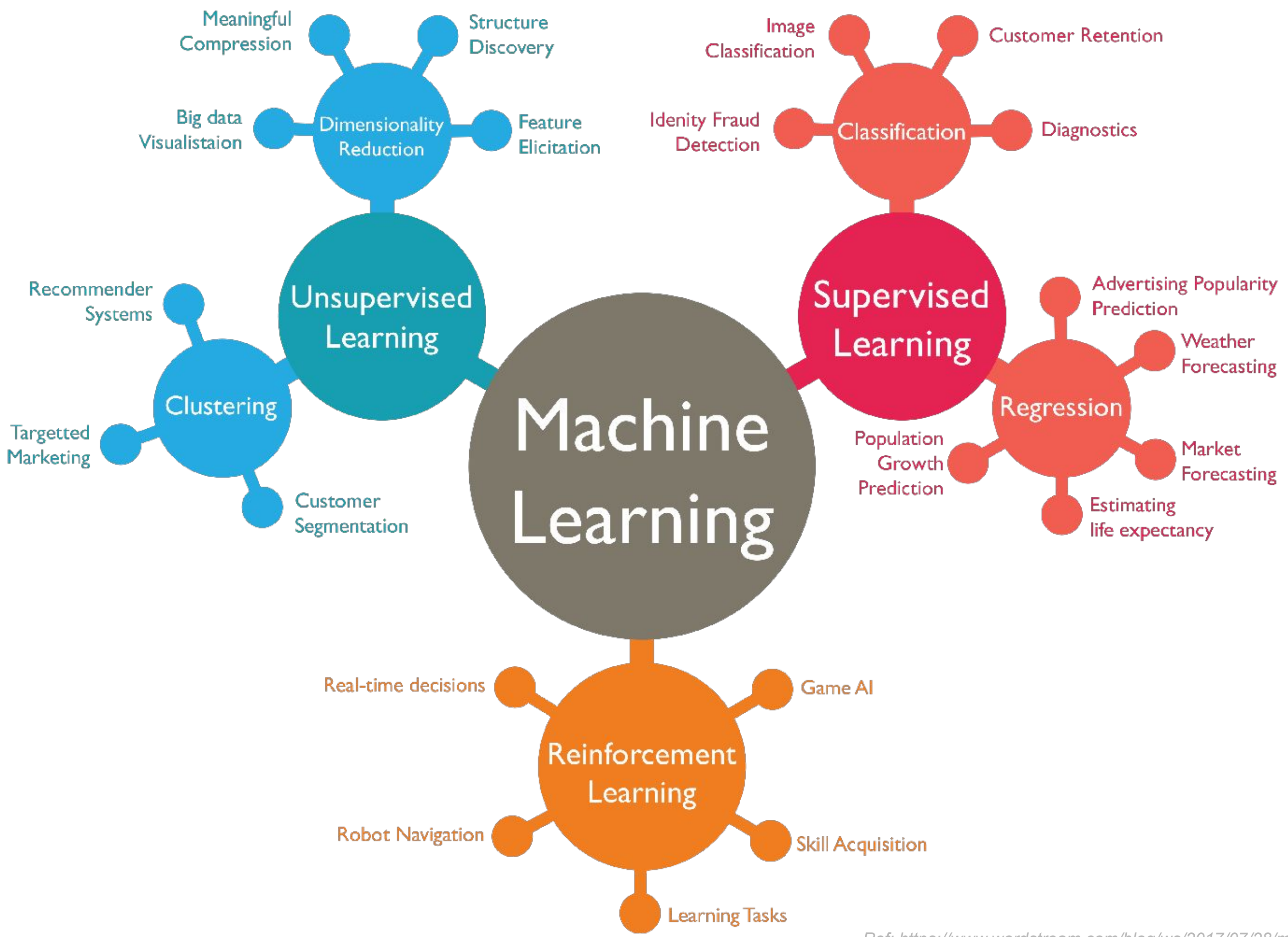
Dependent variables (y)
Aka Outcome

Polynomial Regression
(degree>1)

Decision Tree Regression

Evaluation
MSE, RMSE, MAE

Ref: https://www.javatpoint.com/regression-vs-classification-in-machine-learning



Ref: https://www.wordstream.com/blog/ws/2017/07/28/machine-learning-applications

Basic Types of Machine Learning

Labeled Data

Example inputs
Example outputs

Supervised Learning Algorithms

Unseen Inputs

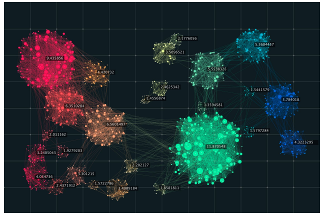
Prediction Model
(Classification/Regression)

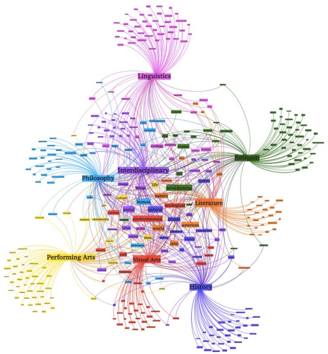
Predicted Outputs

Unlabeled Input Data

Unsupervised Learning Algorithms

Hidden/Unusual patterns
Alternative representations





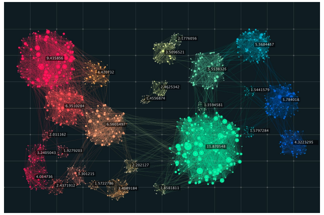
Refs:
[Face Detection](#)
[Spam Detection](#)
[Topic Modeling](#)
[Social Networks](#)

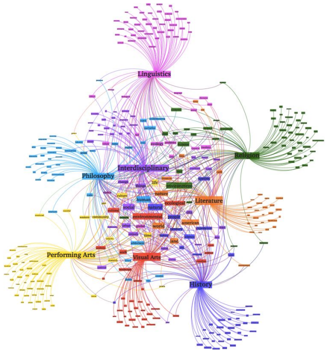
Outline

- Unsupervised-learning
- Revisiting Data Points
- Clustering Concept
  - Similarity Measures
  - Distance Functions
  - Quality of Clustering
- Clustering Approaches
  - Partitioning-based
  - Hierarchical-based
  - Density-based

Unsupervised-learning

Unlabeled Input Data → **Unsupervised Learning Algorithms** → Hidden/Unusual patterns Alternative representations



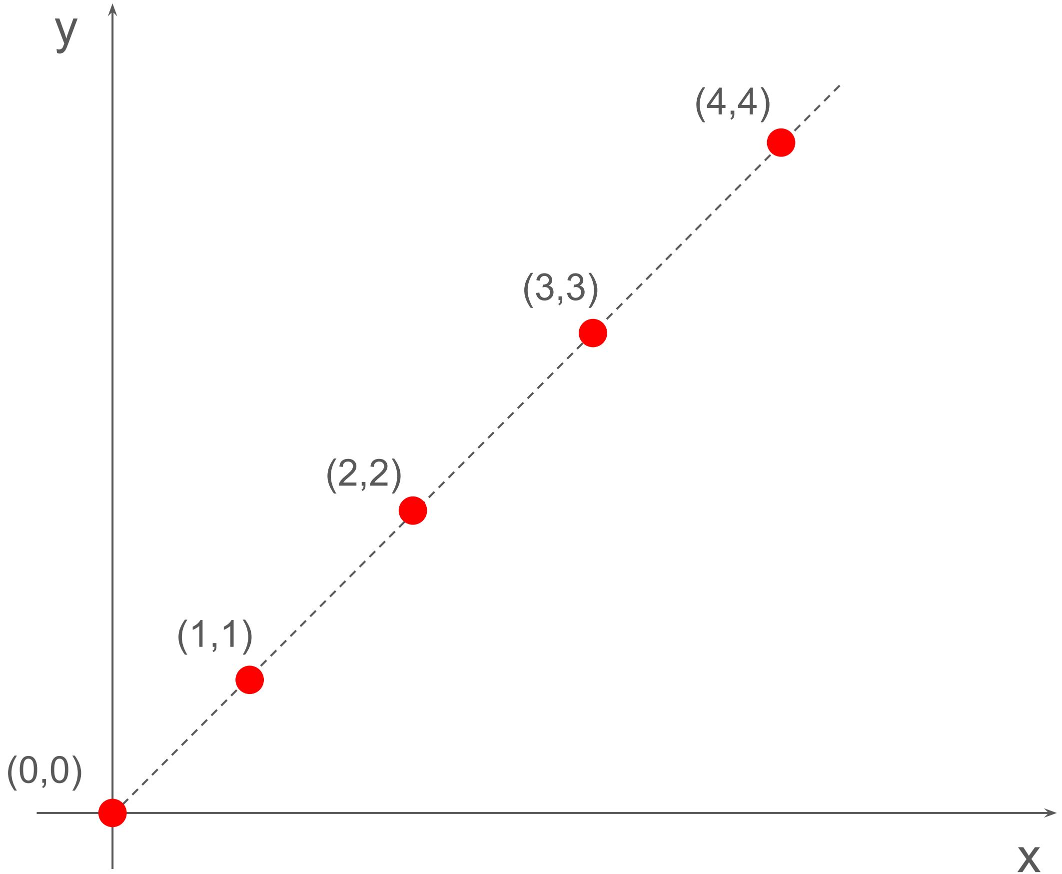


- No predefined classes
- No labels needed
- Applications:
  - Stand-alone tool to get insight into data distribution
  - A preprocessing step for other algorithms

A ***data point*** represents a single piece of information.

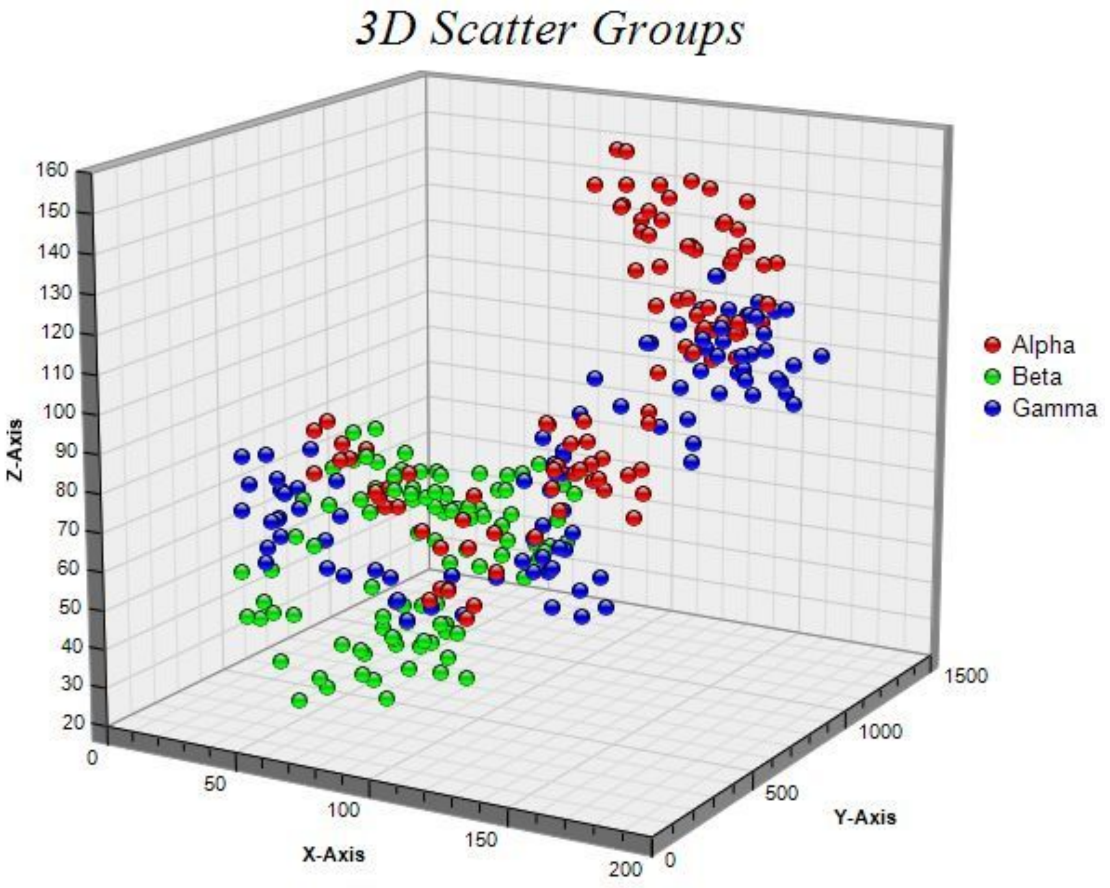
Data Points
2D

x	y
0	0
1	1
2	2
3	3
4	4



Data Points
3D

x	y	z
?	?	?
?	?	?
?	?	?
?	?	?
?	?	?

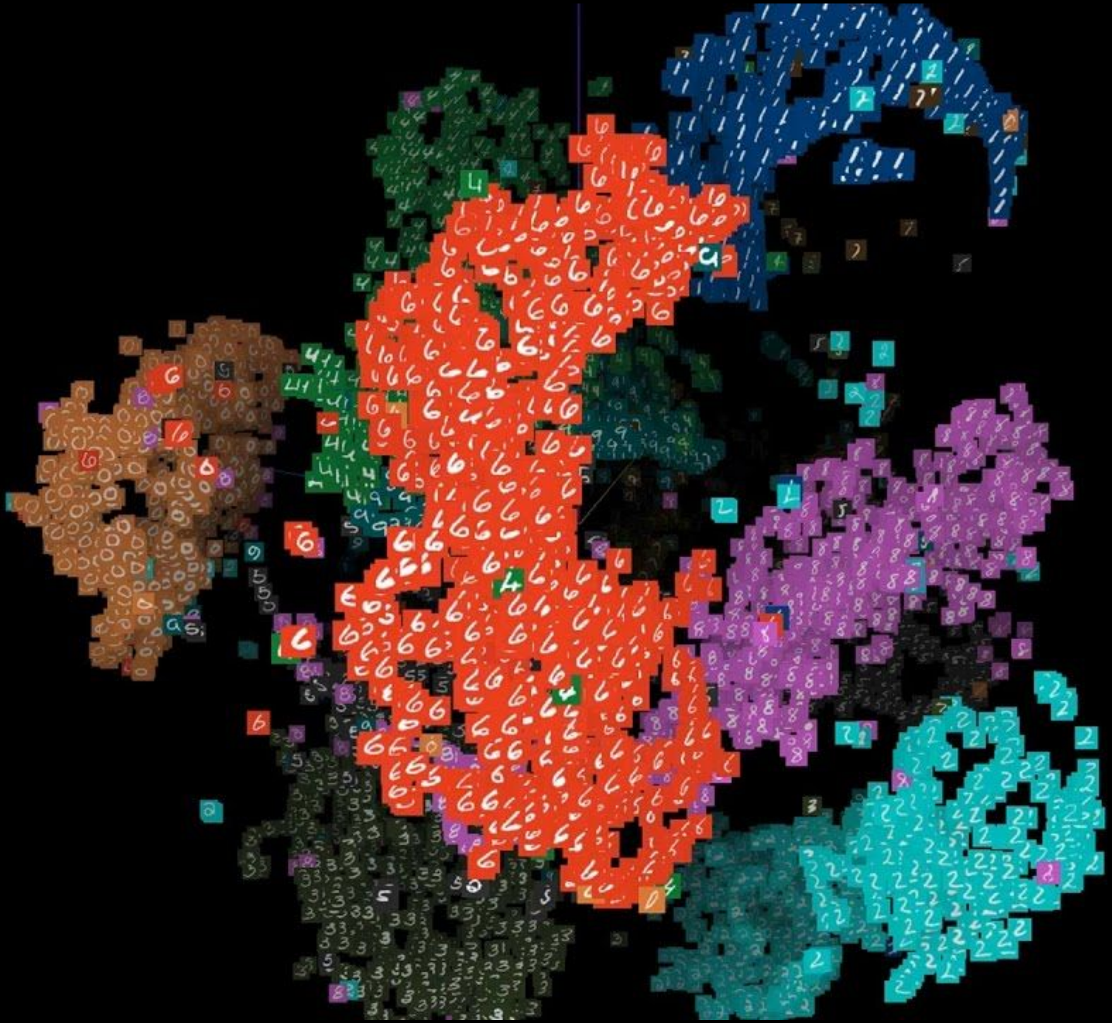


Ref: https://forum.posit.co/t/3d-scatter-plot-with-data-points-colored-according-to-their-group/99145

Data Points

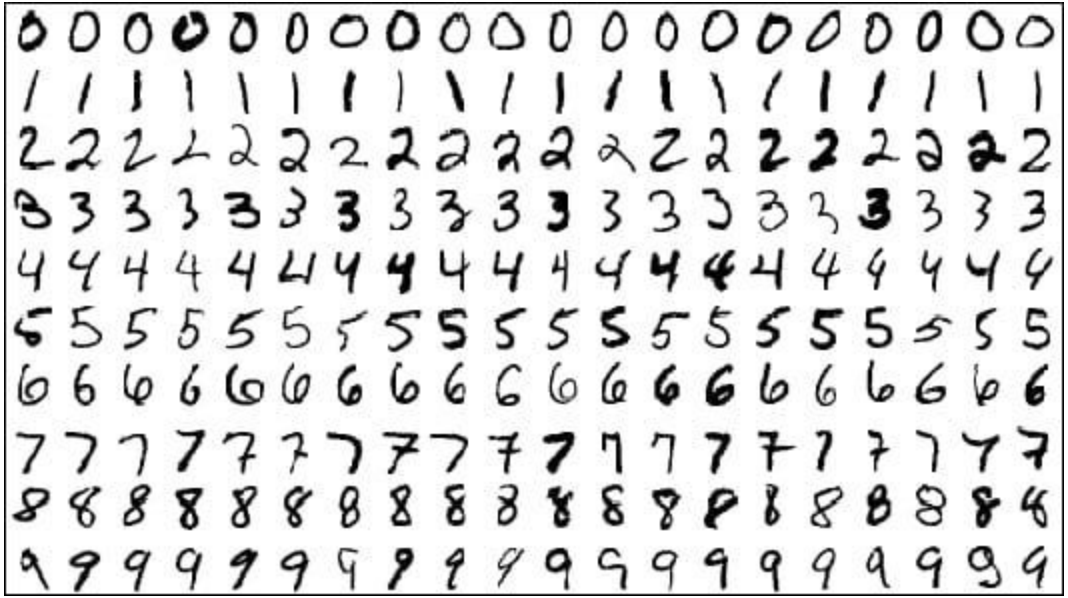
High-dimensional Space

Attribute 1	Attribute 2	…	Attribute k
x_{11}	x_{12}	…	x_{1k}
x_{21}	x_{22}	…	x_{2k}
x_{31}	x_{32}	…	x_{3k}
…	…	…	…
x_{n1}	x_{n2}	…	x_{nk}



Ref: https://www.youtube.com/watch?v=wvsE8jm1GzE

MNIST 0-9



Ref: https://medium.com/@navyashree.raghupatro/recognizing-handwritten-digits-with-scikit-learn-8d248dc01b6d

Cluster of Data Points

Cluster Analysis

Cluster

A collection of data objects

Similarity

Dissimilarity

Cluster Analysis

Finding Similarities

Characteristics

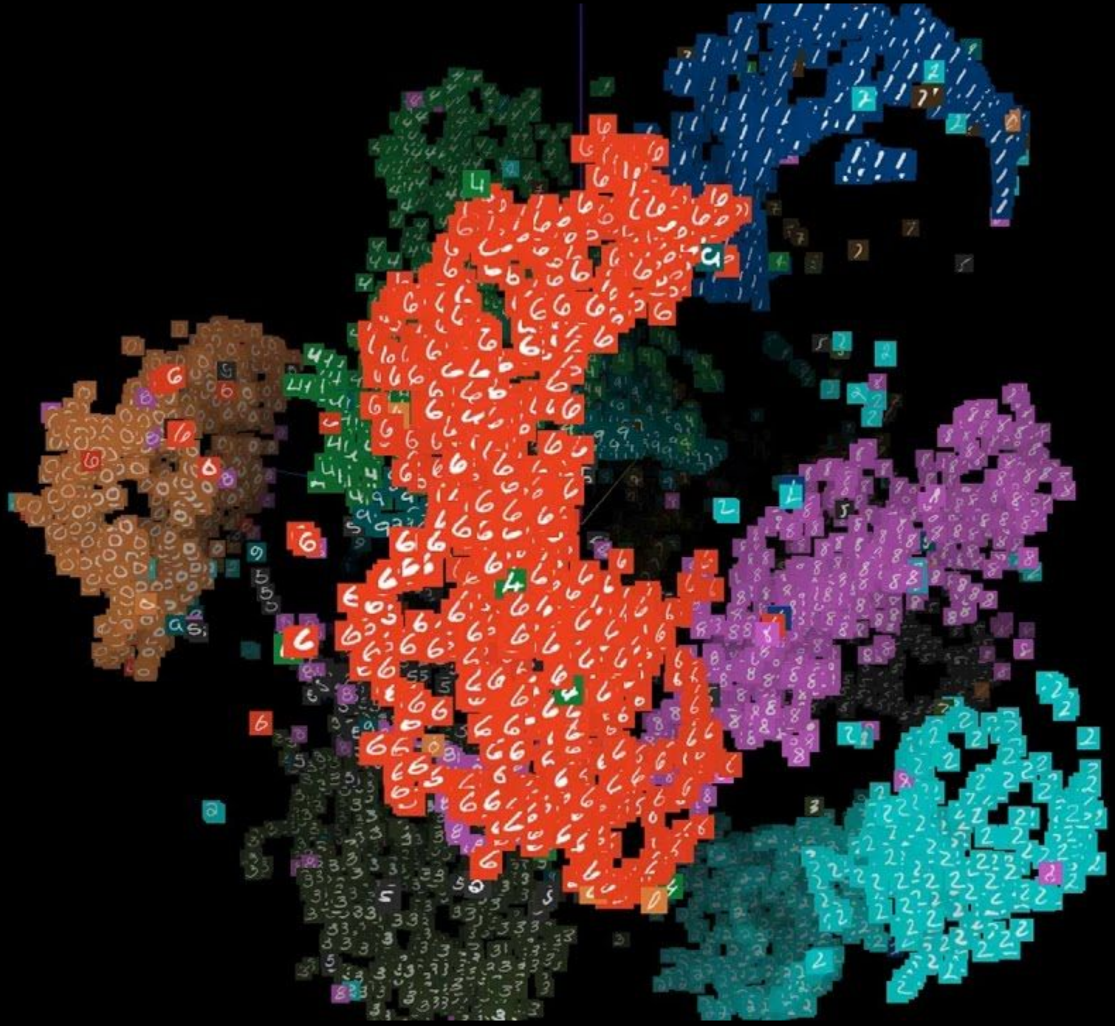
Grouping based on similarities

Similarity

- If two things are similar in some ways, they often share other characteristics

- Applications:
    - Recommendations (e.g. Netflix, Amazon)
    - Troubleshooting
    - Knowledge management
    - Customer segmentation
    - Even classification or regression

Determine Similarities



Distance Functions

Euclidean Distance (L2 norm)

$$d_{Euclidean}(X,Y) = ||X - Y||_2 = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \ldots}$$

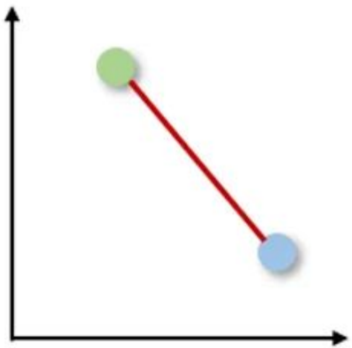
Manhattan Distance (L1 norm)

$$d_{Manhattan}(X,Y) = ||X - Y||_1 = |x_1 - y_1| + |x_2 - y_2| + \ldots$$

Cosine Distance

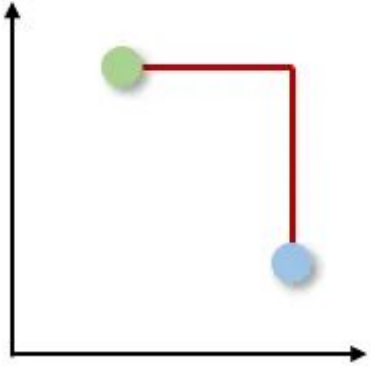
$$d_{Cosine}(X,Y) = \frac{X \cdot Y}{||X|| \times ||Y||}$$

Euclidean



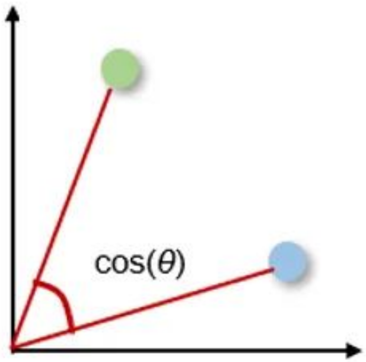
- Shortest distance between two real-valued vectors
- Most common

Manhattan



- Taxicab or City-block distance
- Shortest distance between two real-valued vectors
- Right angles

Cosine

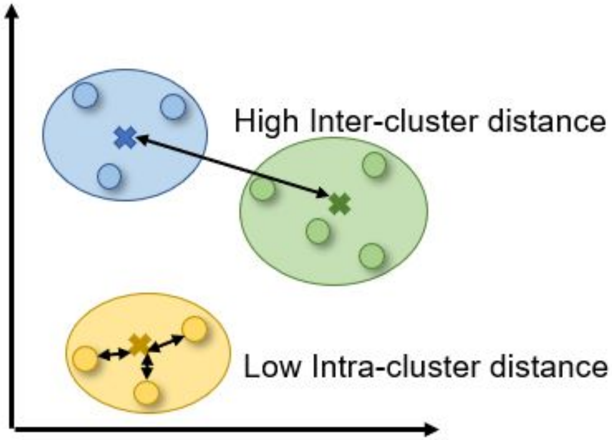


- Cosine between two vectors
- Often used in higher dimensionality
- Measured in Θ
    - Θ = 0° → similar (overlap)
    - Θ = 90° → dissimilar

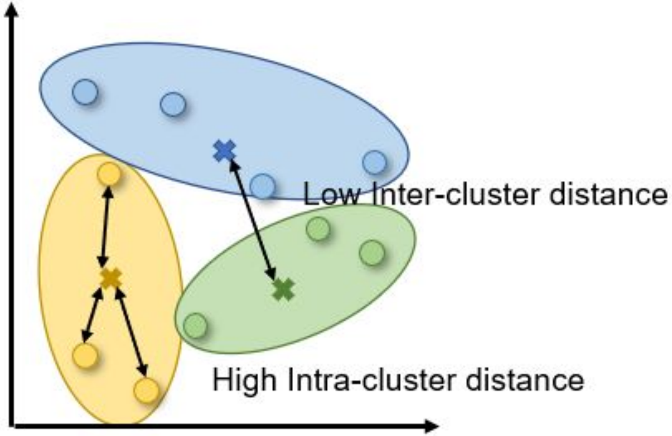
Ref: https://medium.com/@jodancker/a-brief-introduction-to-distance-measures-ac89cbd2298

Quality of Clustering

- High *intra*-class similarity
- Low *inter*-class similarity



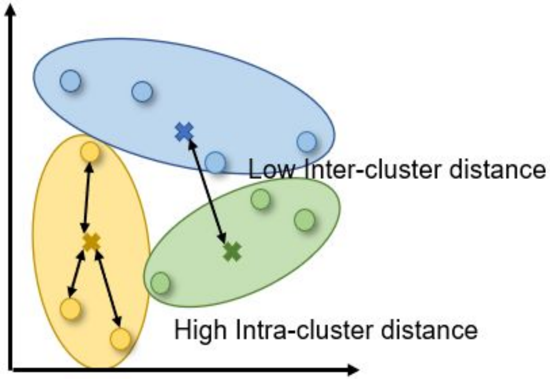
Good Example

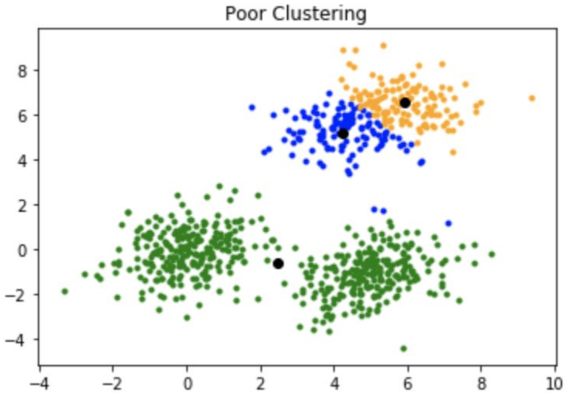


Bad Example

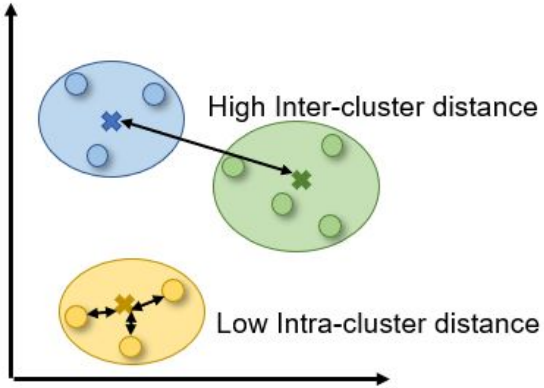
Ref: https://medium.com/@jodancker/a-brief-introduction-to-cluster-validation-ca4215295b06

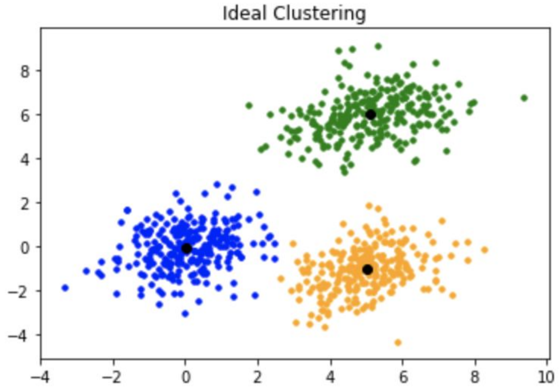
Bad
Example





Good
Example





Refs:
https://www.geeksforgeeks.org/ml-k-means-algorithm/
https://medium.com/@jodancker/a-brief-introduction-to-cluster-validation-ca4215295b06

Clustering Approaches

- **Partitioning** Approach
- **Hierarchical** Approach
- **Density-based** Approach

Partitioning: Basic Concept

- Breaking down a large group of data points into partitions
- While still taking into account the distance → minimum

Basic Concept

Construct a partition of a database D of n objects into a set of k clusters, such that sum of squared distance is minimal

Partitioning: Brute-force

Finding a global optimal clustering:

1. Exhaustively enumerate <u>ALL</u> the clusterings

2. Return the clustering with the min score

$$\hat{C} = \arg\min_{C}\{SSE(C)\}$$

Basic Concept

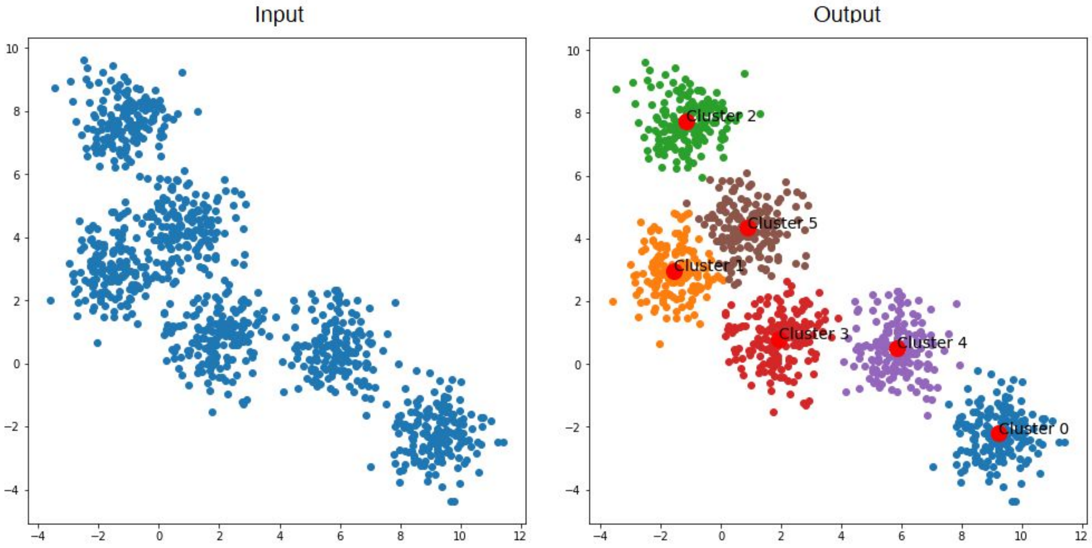
Construct a partition of a database D of n objects into a set of k clusters, such that sum of squared distance is minimal

k=2 k=3 k=4 k=5 k=...

Computationally Infeasible

Partitioning: K-means

- Each cluster is represented by the center of the cluster
- **Centroid** → Center of the cluster → Average



Ref: https://towardsai.net/p/l/centroid-neural-network-an-efficient-and-stable-clustering-algorithm

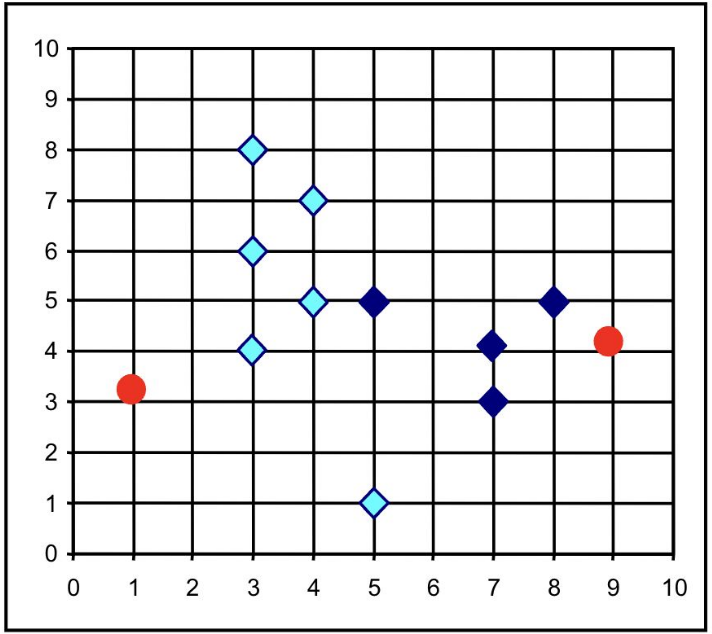
K-means Steps

1. Partition objects into k non-empty subsets.
2. Compute seed points as the centroids of the clusters of the current partition.
3. Assign each object to the cluster with the nearest seed point.
4. Go back to Step 2, stop when no more new assignment.

K-means Steps (1-2)

1. Partition objects into k non-empty subsets. (k=2)

2. Compute seed points as the centroids of the clusters of the current partition.

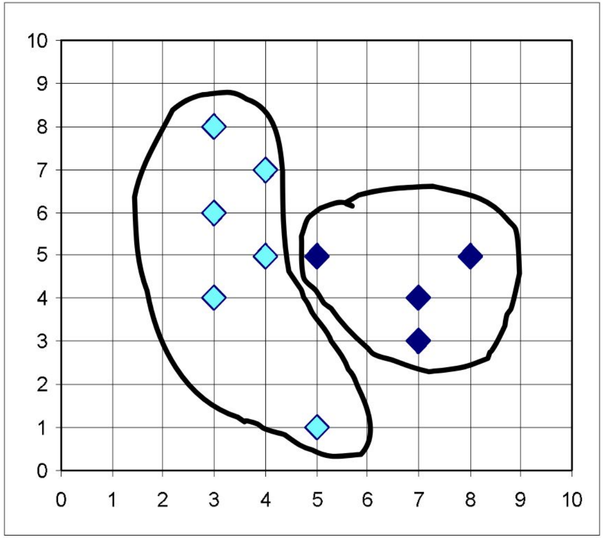


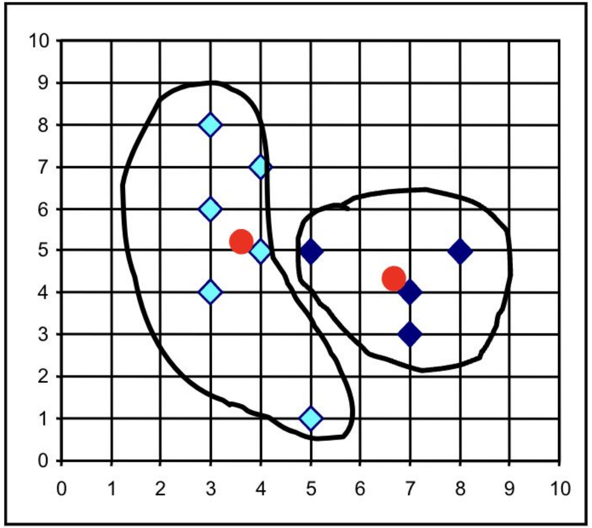
K-means Steps (3-4)

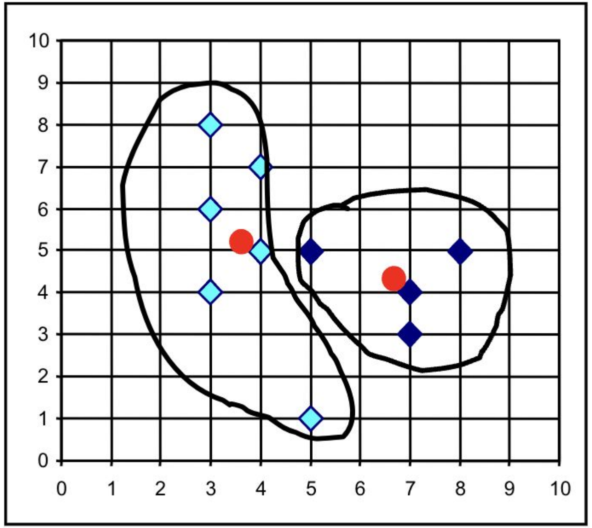
Step 2: Compute seed points as the centroids of the clusters of the current partition.

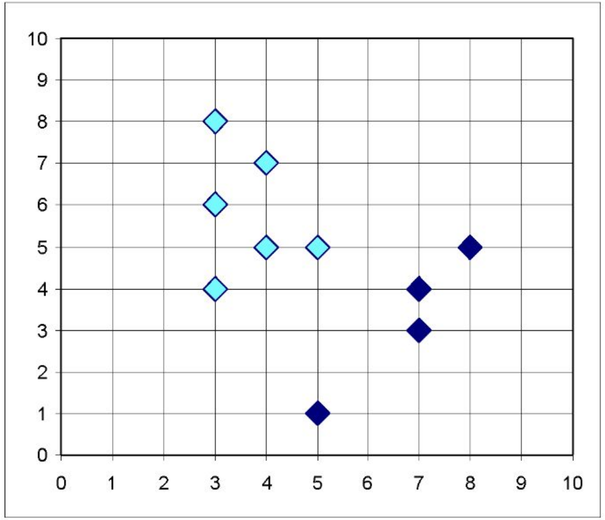
3. Assign each object to the cluster with the nearest seed point.

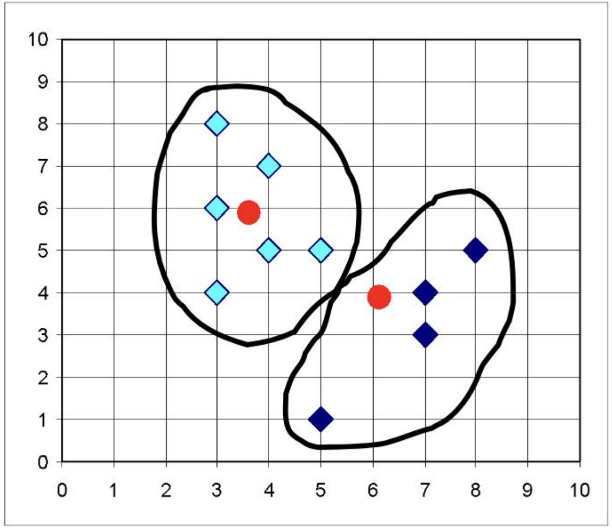
4. Go back to **Step 2**, stop when no more new assignment.





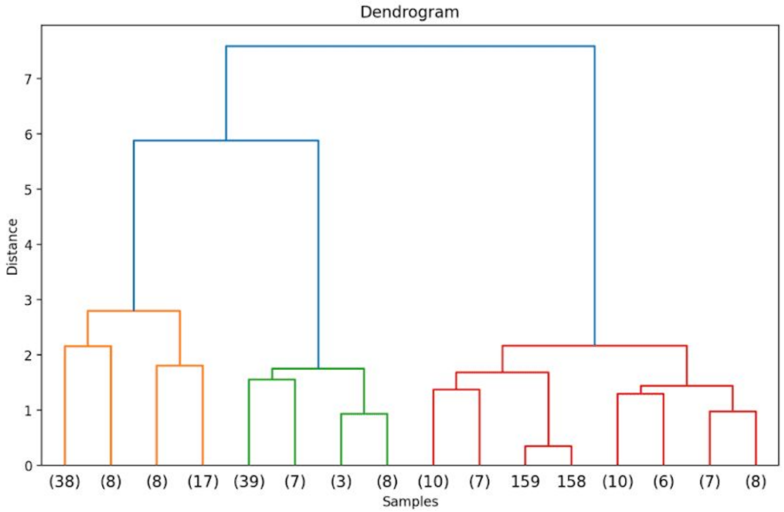

10
9
8
7
6
5
4
3
2
1
0
0 1 2 3 4 5 6 7 8 9 10


10
9
8
7
6
5
4
3
2
1
0
0 1 2 3 4 5 6 7 8 9 10


10
9
8
7
6
5
4
3
2
1
0
0 1 2 3 4 5 6 7 8 9 10

Hierarchical Methods

- Produces a set of nested clusters
- Organized as a hierarchical tree



Ref: https://www.kdnuggets.com/unveiling-hidden-patterns-an-introduction-to-hierarchical-clustering

Hierarchical: Basic Concept

- Merge or split one cluster at a time
- Merge → Agglomerative
- Split → Divisive

Ref: https://harshsharma1091996.medium.com/hierarchical-clustering-996745fe656b

Hierarchical: Agglomerative

Compute the proximity matrix

Let each data point be a cluster

Repeat

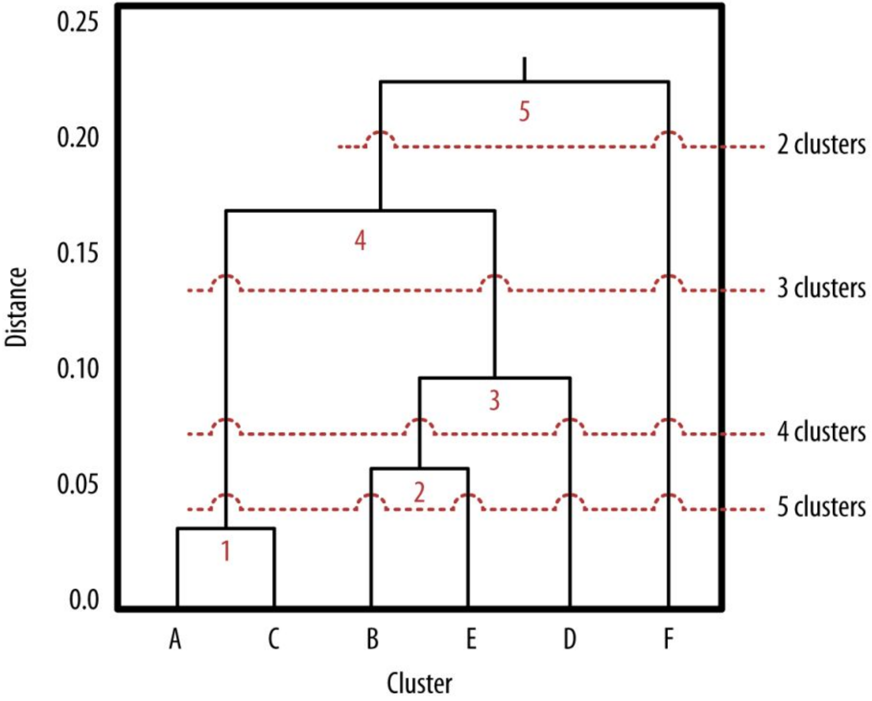
Merge the two closest clusters

Update the proximity matrix

Until only a single cluster remains

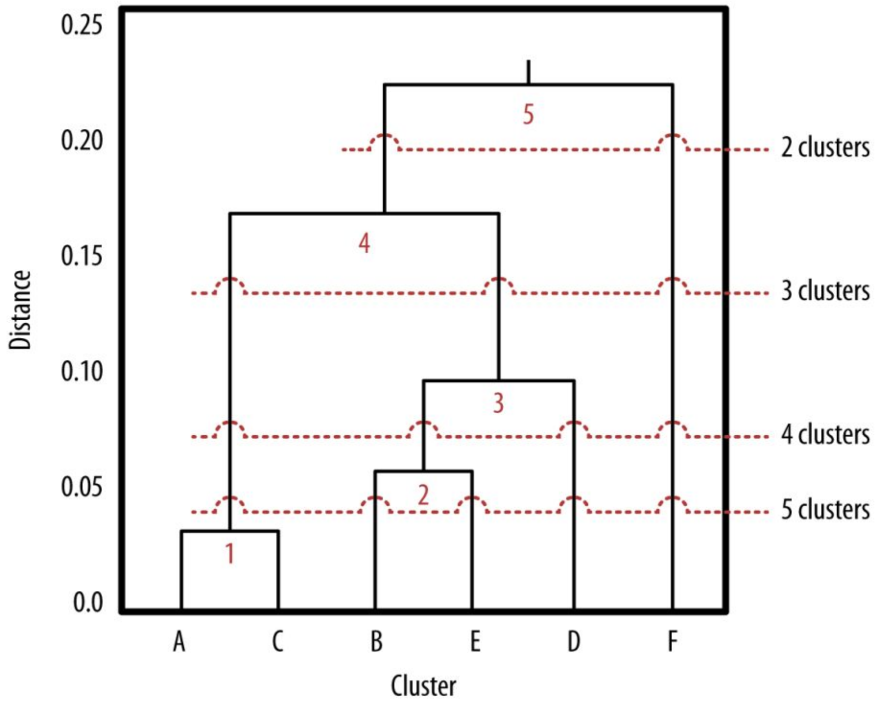
	p1	p2	p3	p4	p5
p1					
p2					
p3					
p4					
p5					

Proximity Matrix

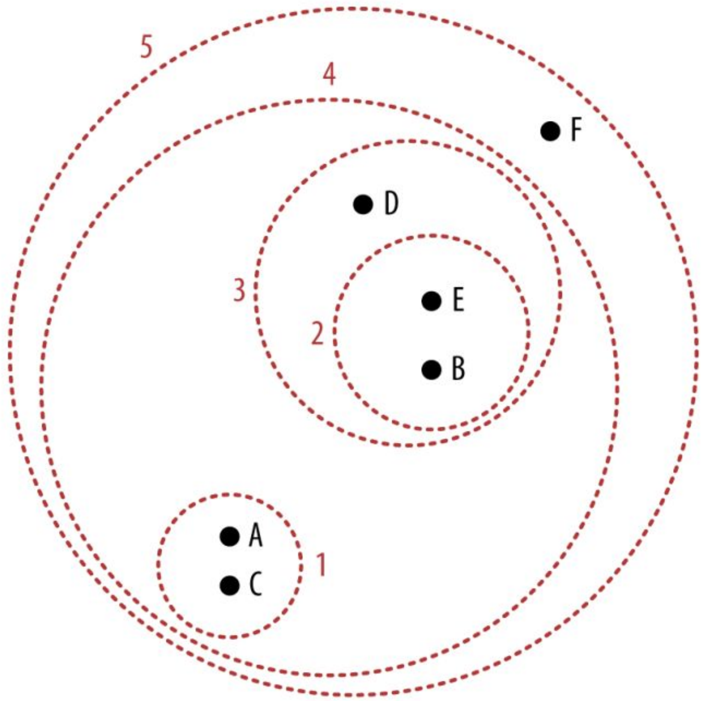


Nearest neighbor pairs are grouped to clusters

- A and C are closest so they are grouped first
- Followed by B and E
- The diagram is known as dendrogram



Dendrogram



Nested Clusters

In Summary

- Unsupervised-learning
- Clustering Concept
  - Similarity Measures
  - Distance Functions
  - Quality of Clustering
- Clustering Approaches
  - Partitioning-based
  - Hierarchical-based
  - Density-based

Q & A