Clustering

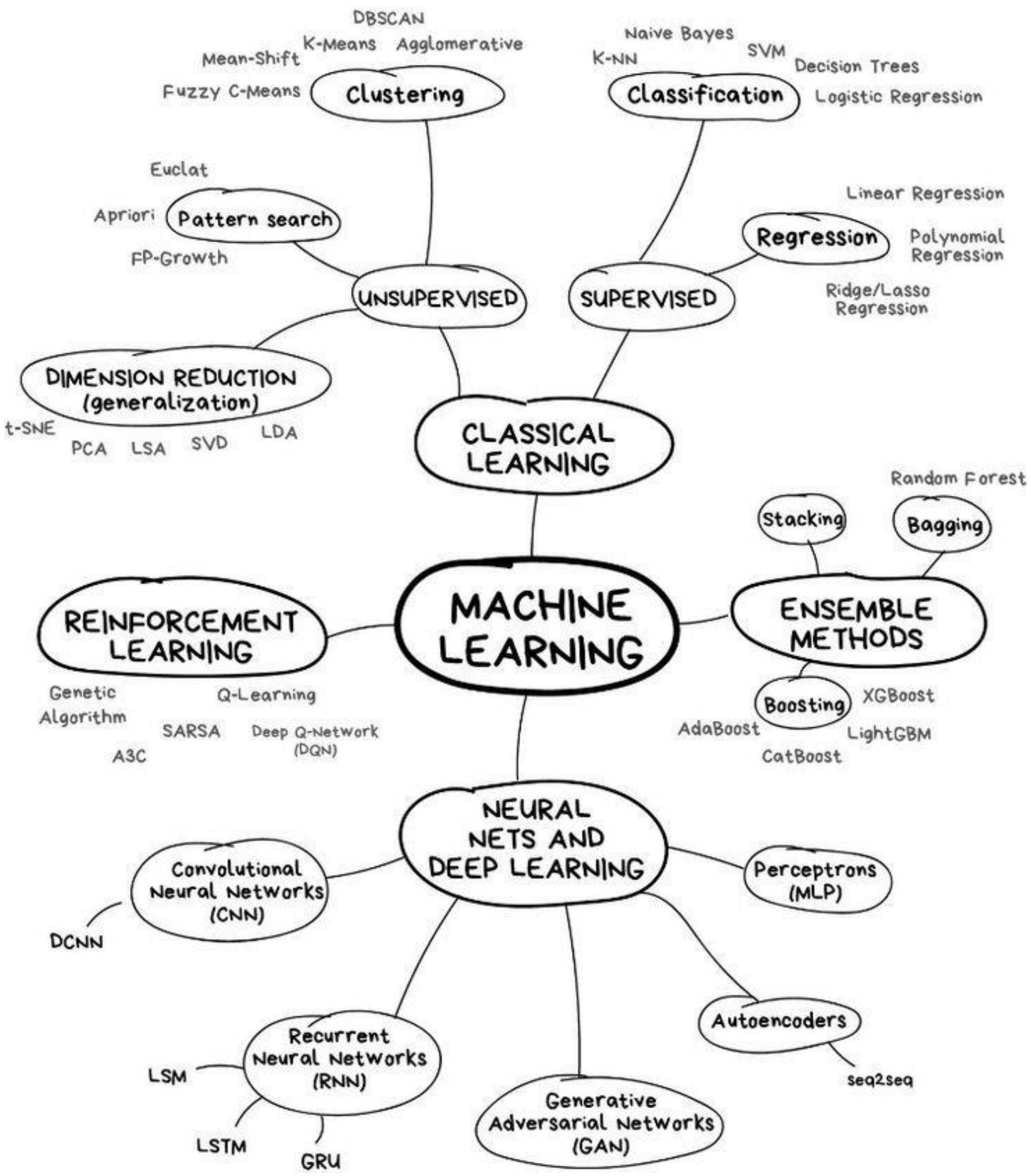
Extended from Kyuseok Shim's slides
and

한양대학교 ERICA
소프트웨어융합대학
COLLEGE OF COMPUTING

인공지능학과
Department of
Artificial Intelligence

정 우 환 (whjung@hanyang.ac.kr)

Fall 2021


MACHINE LEARNING
CLASSICAL LEARNING
UNSUPERVISED
SUPERVISED
Clustering
DBSCAN
K-Means
Agglomerative
Mean-Shift
Fuzzy C-Means
Classification
Naive Bayes
SVM
K-NN
Decision Trees
Logistic Regression
Pattern search
Euclat
Apriori
FP-Growth
Regression
Linear Regression
Polynomial Regression
Ridge/Lasso Regression
DIMENSION REDUCTION (generalization)
t-SNE
PCA
LSA
SVD
LDA
ENSEMBLE METHODS
Stacking
Bagging
Random Forest
Boosting
XGBoost
AdaBoost
LightGBM
CatBoost
REINFORCEMENT LEARNING
Genetic Algorithm
Q-Learning
SARSA
Deep Q-Network (DQN)
A3C
NEURAL NETS AND DEEP LEARNING
Convolutional Neural Networks (CNN)
DCNN
Perceptrons (MLP)
Recurrent Neural Networks (RNN)
LSM
LSTM
GRU
Generative Adversarial Networks (GAN)
Autoencoders
seq2seq

What is Cluster Analysis?

- Cluster: A collection of data objects
  - similar (or related) to one another within the same group
  - dissimilar (or unrelated) to the objects in other groups
- Cluster analysis (or *clustering*, *data segmentation, …*)
  - Finding similarities between data according to the characteristics found in the data and grouping similar data objects into clusters
- Unsupervised learning: no predefined classes (i.e., *learning by observations* vs. learning by examples: supervised)
- Typical applications
  - As a stand-alone tool to get insight into data distribution
  - As a preprocessing step for other algorithms

Clustering

- Given:
  - Data points and number of desired clusters K
- Group the data points into K clusters
  - Data points within clusters are more similar than across clusters

Data Clustering

Click log database

User	$News_1$	$News_2$	$News_3$	$News_4$	$News_5$	$News_6$
u_2	☑	☑	☑			
u_5		☑	☑			
u_6	☑	☑	☑		☑	
u_1				☑	☑	☑
u_3				☑		☑
u_4	☑			☑	☑	☑

$Cluster_1$
$News_1$, $News_2$, $News_3$

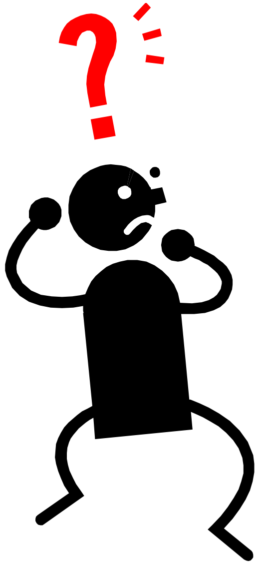
$Cluster_2$
$News_4$, $News_5$, $News_6$

Data Clustering

Clustering
Results

Cluster$_1$
News$_1$, News$_2$,
News$_3$

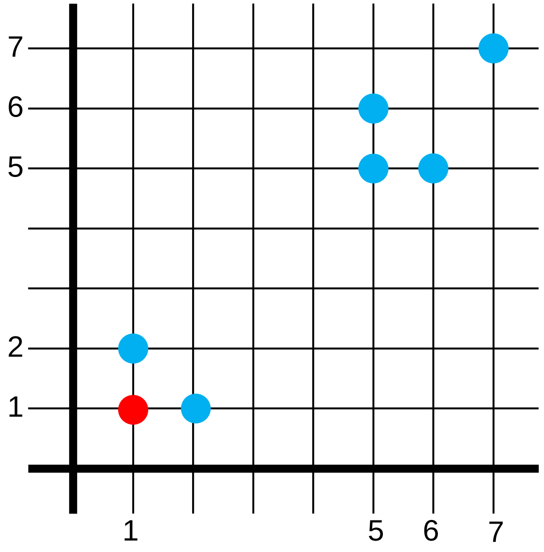
Cluster$_2$
News$_4$, News$_5$,
News$_6$



Recommend
News$_5$
for this new user

A new user
He clicked News$_4$ and News$_6$

What is Clustering?



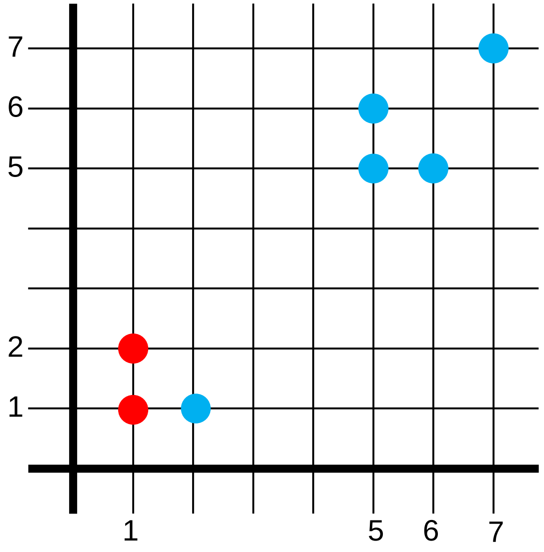
(1, 1) (1, 2) (2, 1) (5, 5) (5, 6) (6, 5) (7, 7)

A possible partition of 2 clusters

Cluster A	Cluster B
(1, 1)	(1, 2) (2, 1) (5, 5) (5, 6) (6, 5) (7, 7)

Even for K (# of desirable clusters) =2, there are too many possible partition of data!

What is Clustering?



(1, 1) (1, 2) (2, 1) (5, 5) (5, 6) (6, 5) (7, 7)

Another possible partition of 2 clusters

Cluster A

(1, 1) (1, 2)

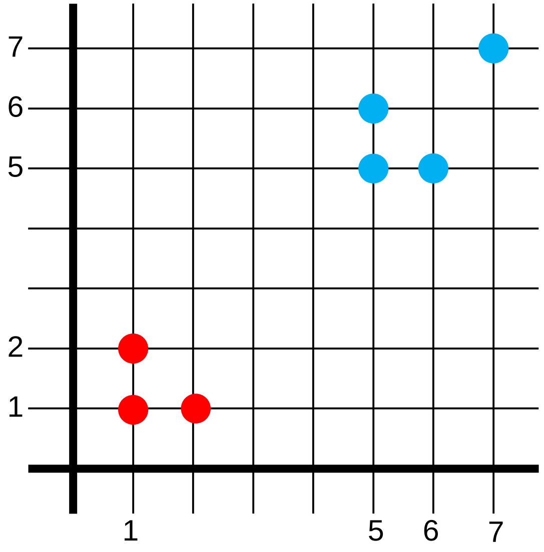
Cluster B

(2, 1)

(5, 5) (5, 6)

(6, 5) (7, 7)

What is Clustering?



(1, 1) (1, 2) (2, 1) (5, 5) (5, 6) (6, 5) (7, 7)

Another possible partition of 2 clusters

Cluster A

(1, 1)

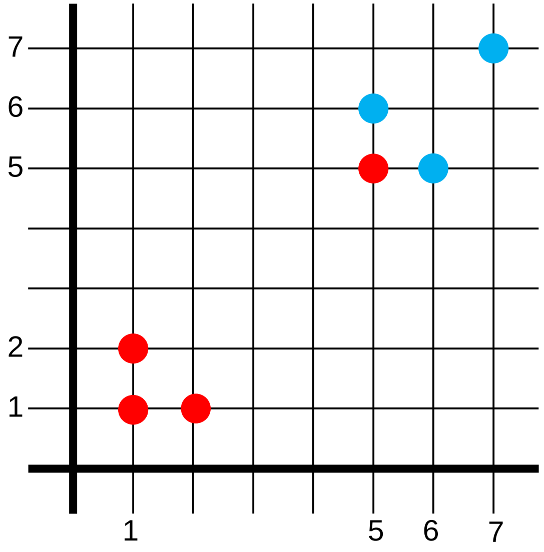
(1, 2) (2, 1)

Cluster B

(5, 5) (5, 6)

(6, 5) (7, 7)

What is Clustering?



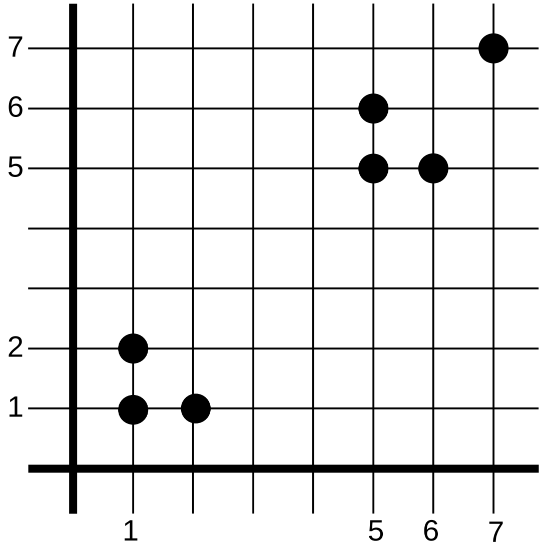
(1, 1) (1, 2) (2, 1) (5, 5) (5, 6) (6, 5) (7, 7)

Another possible partition of 2 clusters

Cluster A	Cluster B
(1, 1) (1, 2)	(5, 6)
(2, 1) (5, 5)	(6, 5) (7, 7)

⋮

What is Clustering?

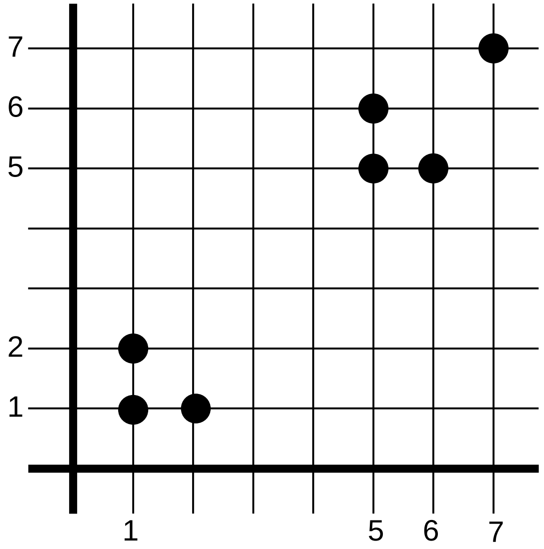


(1, 1) (1, 2) (2, 1) (5, 5) (5, 6) (6, 5) (7, 7)

How many all possible partitions of 2 clusters?

$= 2^7$ (# of subsets)

What is Clustering?



(1, 1) (1, 2) (2, 1) (5, 5) (5, 6) (6, 5) (7, 7)

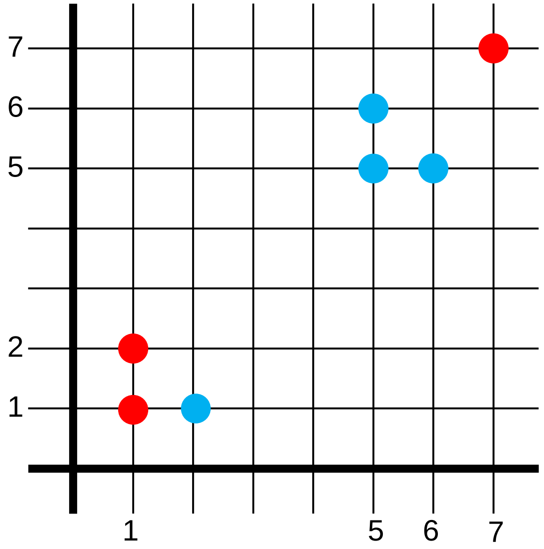
Popular goodness measure of clustering?

minimal sum of squared distances

$$\Sigma_{m=1}^{k}\Sigma_{t_{mi}\in Km}(C_m - t_{mi})^2$$

Cluster center

What is Clustering?



A possible partition

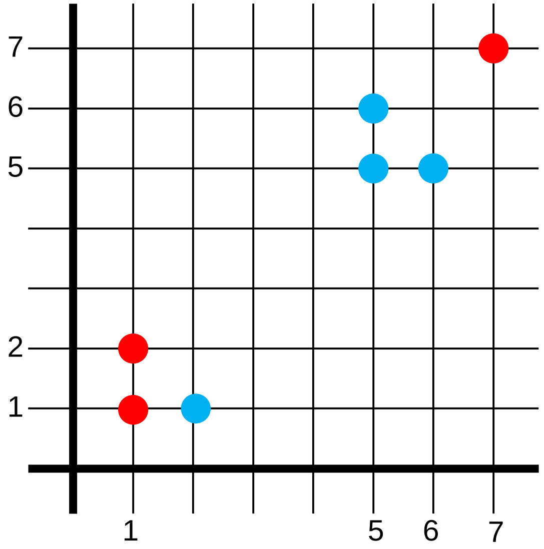
Cluster A

(1, 1) (1, 2) (7, 7)

Cluster B

(2, 1) (5, 5) (5, 6) (6, 5)

What is Clustering?



A possible partition

Cluster A

(1, 1) (1, 2) (7, 7)

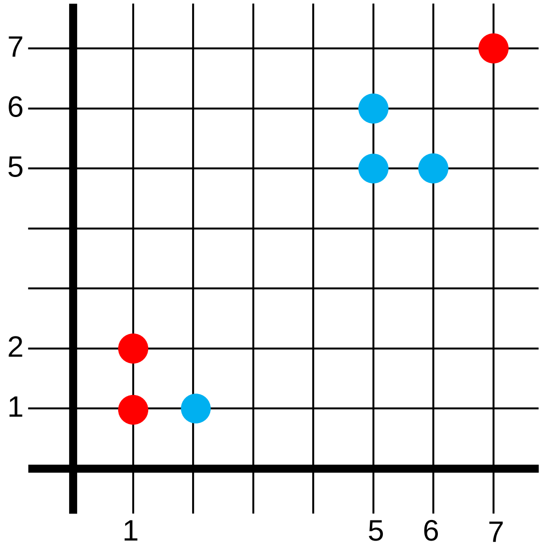
Cluster B

(2, 1) (5, 5) (5, 6) (6, 5)

Center : (9/3, 10/3) Center : (18/4, 17/4)

What is Clustering?

A possible partition



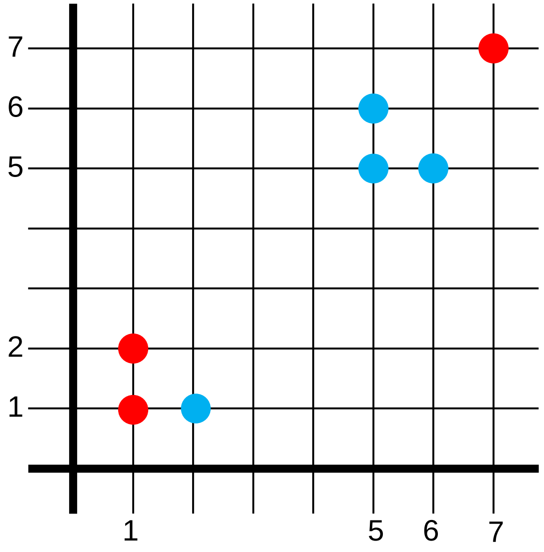
Cluster A		
(1, 1)	(1, 2)	(7, 7)

Cluster B			
(2, 1)	(5, 5)	(5, 6)	(6, 5)

Center : (9/3, 10/3)   Center : (18/4, 17/4)

$(6/3)^2+(7/3)^2$

What is Clustering?



A possible partition

Cluster A		
(1, 1)	(1, 2)	(7, 7)

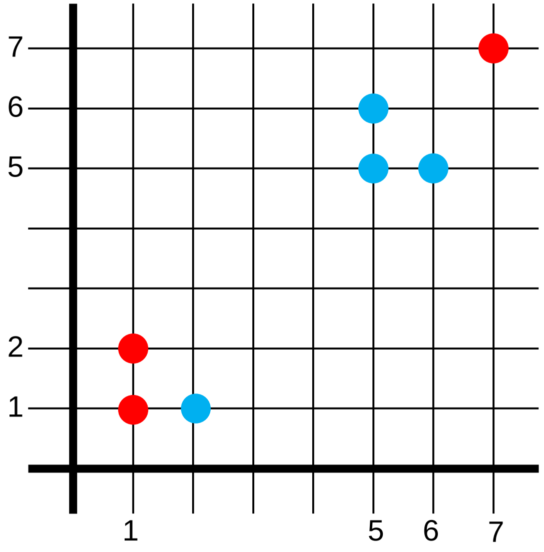
Cluster B			
(2, 1)	(5, 5)	(5, 6)	(6, 5)

Center : (9/3, 10/3) Center : (18/4, 17/4)

$(10/4)^2+(13/4)^2+(2/4)^2+(3/4)^2+(2/4)^2+(7/4)^2+(6/4)^2+(3/4)^2$

$(6/3)^2+(7/3)^2+(6/3)^2+(4/3)^2+(12/3)^2+(11/3)^2$

What is Clustering?



A possible partition

Cluster A		
(1, 1)	(1, 2)	(7, 7)

Cluster B			
(2, 1)	(5, 5)	(5, 6)	(6, 5)

Center : (9/3, 10/3)   Center : (18/4, 17/4)

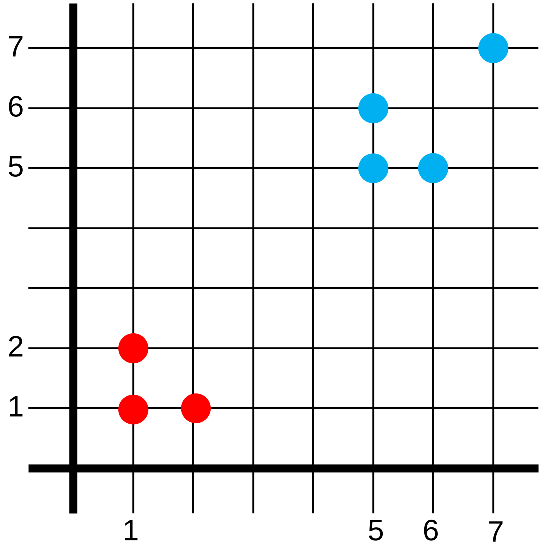
$(10/4)^2+(13/4)^2+(2/4)^2+(3/4)^2+(2/4)^2+(7/4)^2+(6/4)^2+(3/4)^2=23.75$

$(6/3)^2+(7/3)^2+(6/3)^2+(4/3)^2+(12/3)^2+(11/3)^2=44.67$

$$\Sigma_{m=1}^{k}\Sigma_{t_{mi}\in Km}(C_m - t_{mi})^2 = 68.42$$

What is Clustering?

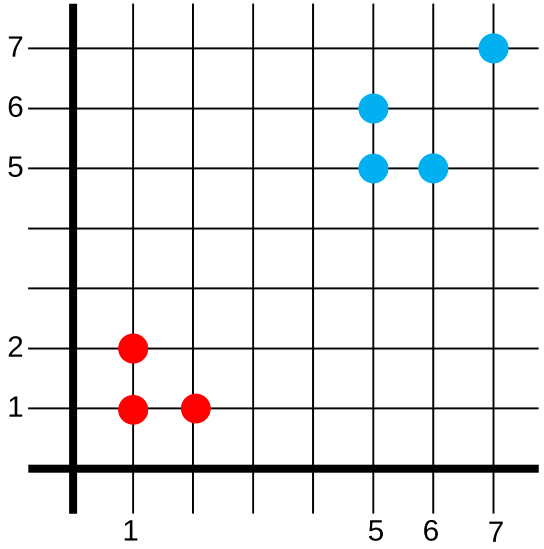
Another possible partition



Cluster A		
(1, 1)	(1, 2)	(2, 1)

Cluster B			
(5, 5)	(5, 6)	(6, 5)	(7, 7)

What is Clustering?



Another possible partition

Cluster A

(1, 1) (1, 2) (2, 1)

Cluster B

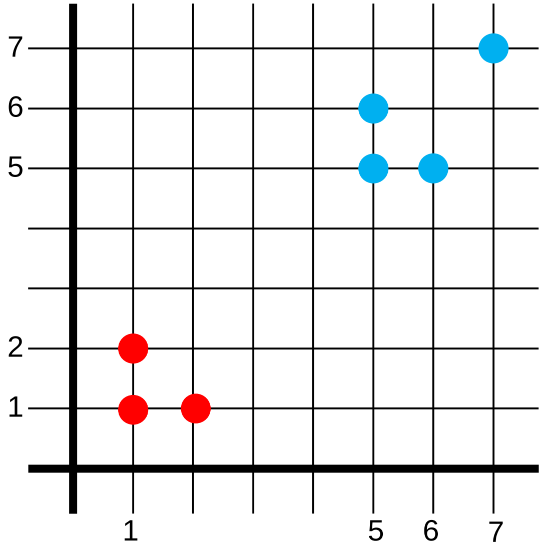
(5, 5) (5, 6) (6, 5) (7, 7)

Center : (4/3, 4/3)

Center : (23/4, 23/4)

What is Clustering?

Another possible partition



Cluster A

(1, 1) (1, 2) (2, 1)

Cluster B

(5, 5) (5, 6) (6, 5) (7, 7)

Center : (4/3, 4/3)

Center : (23/4, 23/4)

$(3/4)^2+(3/4)^2+(3/4)^2+(1/4)^2+(1/4)^2+(3/4)^2+(5/4)^2+(5/4)^2=5.5$

$(1/3)^2+(1/3)^2+(1/3)^2+(2/3)^2+(2/3)^2+(1/3)^2=1.33$

$$\Sigma_{m=1}^{k}\Sigma_{t_{mi}\in Km}(C_m - t_{mi})^2 = 6.83$$

This partition is better than previous one!

Considerations for Cluster Analysis

- Partitioning criteria
  - Single level vs. hierarchical partitioning (often, multi-level hierarchical partitioning is desirable)
- Separation of clusters
  - Exclusive (e.g., one customer belongs to only one region) vs. non-exclusive (e.g., one document may belong to more than one class)
- Similarity measure
  - Distance-based (e.g., Euclidian, road network, vector) vs. connectivity-based (e.g., density or contiguity)
- Clustering space
  - Full space (often when low dimensional) vs. subspaces (often in high-dimensional clustering)

Requirements and Challenges

- Scalability
  - Clustering all the data instead of only on samples
- Ability to deal with different types of attributes
  - Numerical, binary, categorical, ordinal, linked, and mixture of these
- Constraint-based clustering
  - User may give inputs on constraints
  - Use domain knowledge to determine input parameters
- Interpretability and usability
- Others
  - Discovery of clusters with arbitrary shape
  - Ability to deal with noisy data
  - Incremental clustering and insensitivity to input order
  - High dimensionality

Major Clustering Approaches (I)

- Partitioning approach:
  - Construct various partitions and then evaluate them by some criterion, e.g., minimizing the sum of square errors
  - Typical methods: k-means, k-medoids, CLARANS
- Hierarchical approach:
  - Create a hierarchical decomposition of the set of data (or objects) using some criterion
  - Typical methods: Diana, Agnes, BIRCH, CAMELEON
- Density-based approach:
  - Based on connectivity and density functions
  - Typical methods: DBSACN, OPTICS, DenClue
- Grid-based approach:
  - based on a multiple-level granularity structure
  - Typical methods: STING, WaveCluster, CLIQUE

Major Clustering Approaches (II)

- Model-based:
  - A model is hypothesized for each of the clusters and tries to find the best fit of that model to each other
  - Typical methods: EM, SOM, COBWEB
- Frequent pattern-based:
  - Based on the analysis of frequent patterns
  - Typical methods: p-Cluster
- User-guided or constraint-based:
  - Clustering by considering user-specified or application-specific constraints
  - Typical methods: COD (obstacles), constrained clustering
- Link-based clustering:
  - Objects are often linked together in various ways
  - Massive links can be used to cluster objects: SimRank, LinkClus

PARTITIONAL CLUSTERING ALGORITHMS

Partitioning Algorithms: Basic Concept

- Partitioning method: Partitioning a database ***D*** of ***n*** objects into a set of ***k*** clusters, such that the sum of squared distances is minimized (where c_i is the centroid or medoid of cluster C_i)

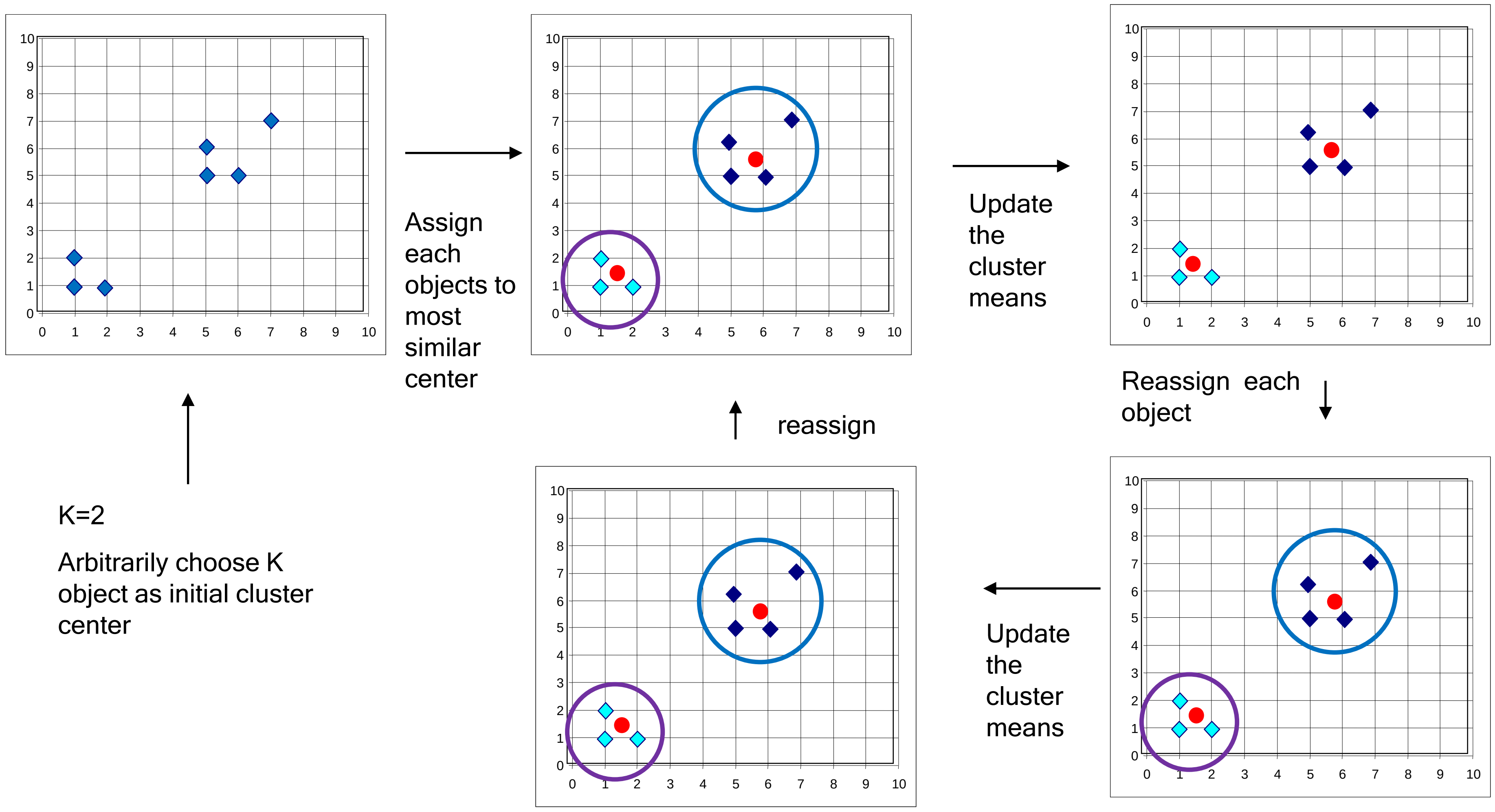
$$E = \Sigma_{i=1}^{k} \Sigma_{p \in C_i} (p - c_i)^2$$

- Given *k*, find a partition of *k clusters* that optimizes the chosen partitioning criterion
  - Global optimal: exhaustively enumerate all partitions
  - Heuristic methods: *k-means* and *k-medoids* algorithms
  - *k-means* (MacQueen'67, Lloyd'57/'82): Each cluster is represented by the center of the cluster
  - *k-medoids* or PAM (Partition around medoids) (Kaufman & Rousseeuw'87): Each cluster is represented by one of the objects in the cluster

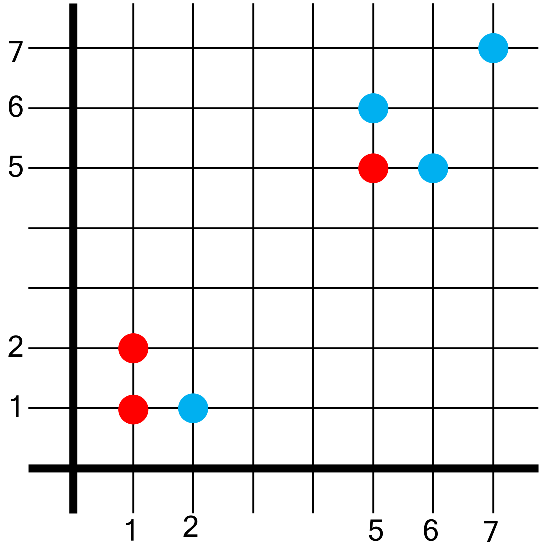
The *K-Means* Clustering Method

- Given *k*, the *k-means* algorithm is implemented in four steps:
  - Partition objects into *k* nonempty subsets
  - Compute seed points as the centroids of the clusters of the current partitioning (the centroid is the center, i.e., *mean point*, of the cluster)
  - Assign each object to the cluster with the nearest seed point
  - Go back to Step 2, stop when the assignment does not change

An Example for *K-Means* Clustering Method



K-Means Clustering Example

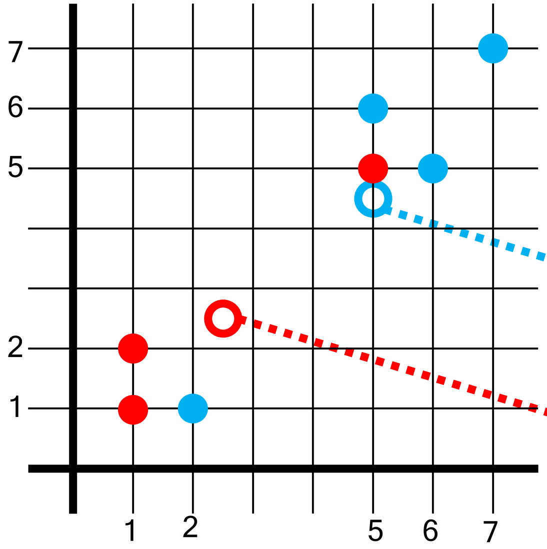


(1, 1) (2, 1) (1, 2) (5, 5) (6, 5) (5, 6) (7, 7)

Class A	Class B
(1, 1) (1, 2) (5, 5)	(2, 1) (6, 5) (5, 6) (7, 7)

Assume we randomly partitioned the objects into 2 nonempty subsets as above!

K-Means Clustering Example



(1, 1) (2, 1) (1, 2) (5, 5) (6, 5) (5, 6) (7, 7)

Class A

(1, 1) (1, 2) (5, 5)

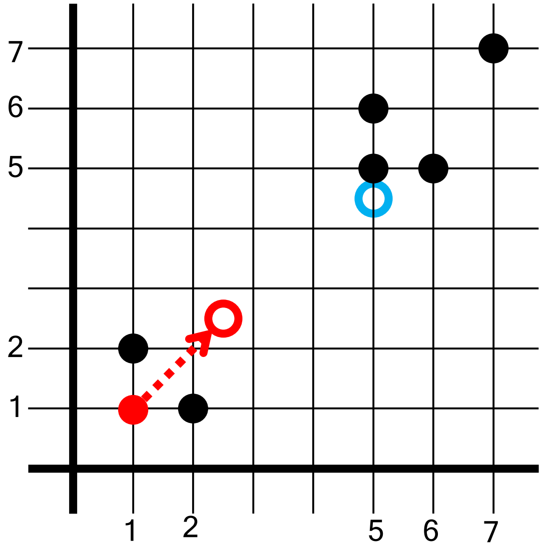
Class B

(2, 1) (6, 5) (5, 6) (7, 7)

Center : (7/3, 8/3)

Center : (20/4, 19/4)

K-Means Clustering Example

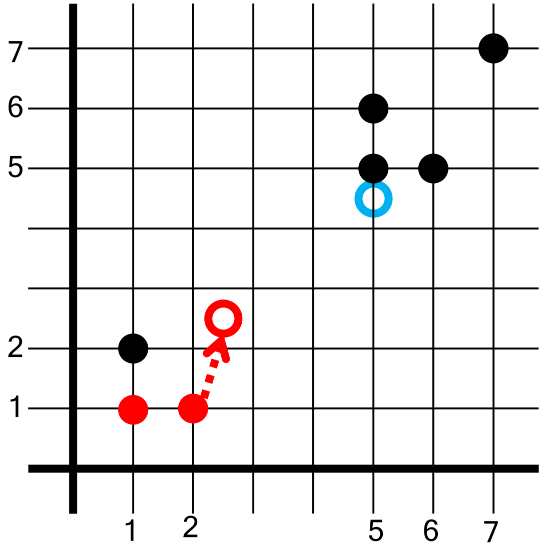


(1, 1) (2, 1) (1, 2) (5, 5) (6, 5) (5, 6) (7, 7)

Class A	Class B
(1, 1)	

Center : (7/3, 8/3)   Center : (20/4, 19/4)

K-Means Clustering Example

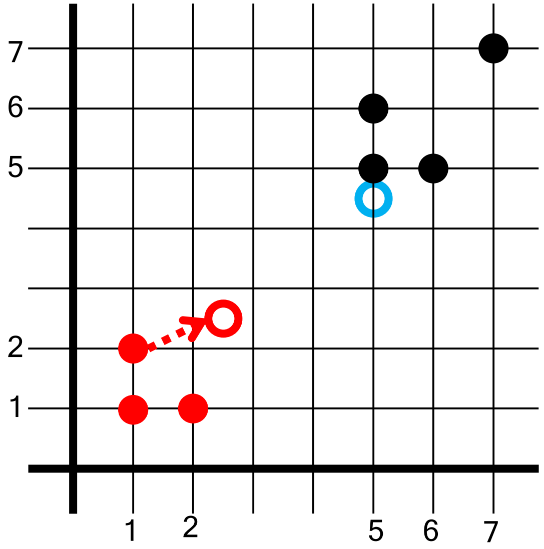


(1, 1) (2, 1) (1, 2) (5, 5) (6, 5) (5, 6) (7, 7)

Class A	Class B
(1, 1) (2, 1)	

Center : (7/3, 8/3) Center : (20/4, 19/4)

K-Means Clustering Example



(1, 1) (2, 1) *(1, 2)* (5, 5) (6, 5) (5, 6) (7, 7)

Class A

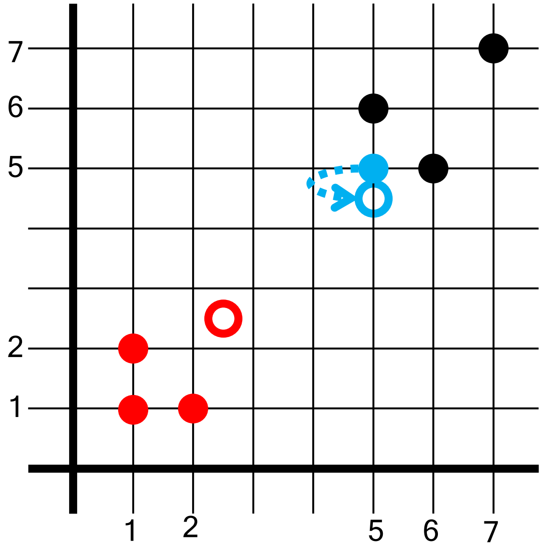
(1, 1) (2, 1) (1, 2)

Class B

Center : (7/3, 8/3)

Center : (20/4, 19/4)

K-Means Clustering Example

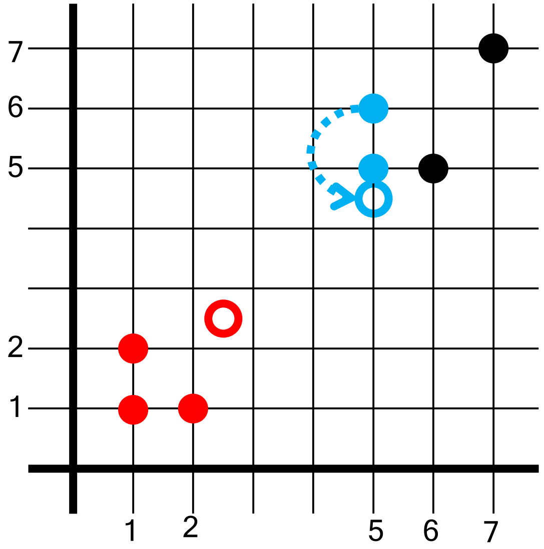


(1, 1) (2, 1) (1, 2) *(5, 5)* (6, 5) (5, 6) (7, 7)

Class A	Class B
(1, 1) (2, 1) (1, 2)	(5, 5)

Center : (7/3, 8/3) Center : (20/4, 19/4)

K-Means Clustering Example



(1, 1) (2, 1) (1, 2) (5, 5) *(6, 5)* (5, 6) (7, 7)

Class A

(1, 1) (2, 1) (1, 2)

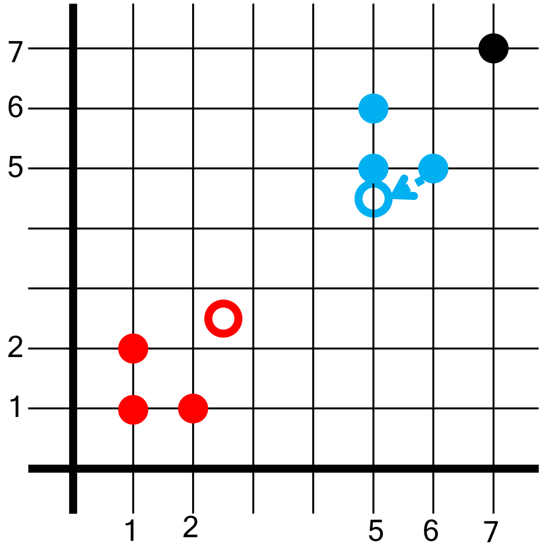
Class B

(5, 5) (6, 5)

Center : (7/3, 8/3)

Center : (20/4, 19/4)

K-Means Clustering Example

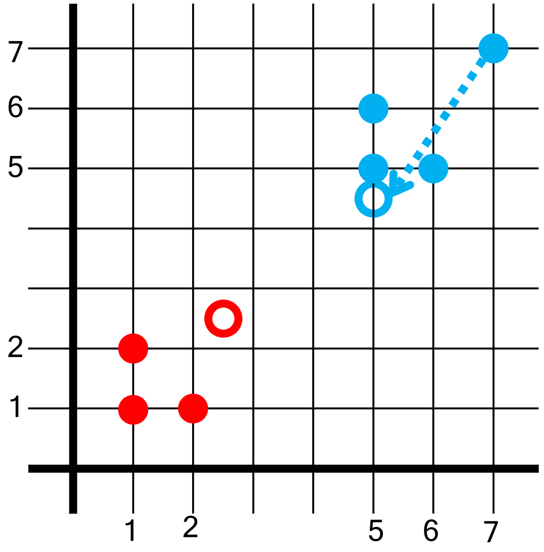


(1, 1) (2, 1) (1, 2) (5, 5) (6, 5) *(5, 6)* (7, 7)

Class A	Class B
(1, 1) (2, 1) (1, 2)	(5, 5) (6, 5) (5, 6)

Center : (7/3, 8/3) Center : (20/4, 19/4)

K-Means Clustering Example

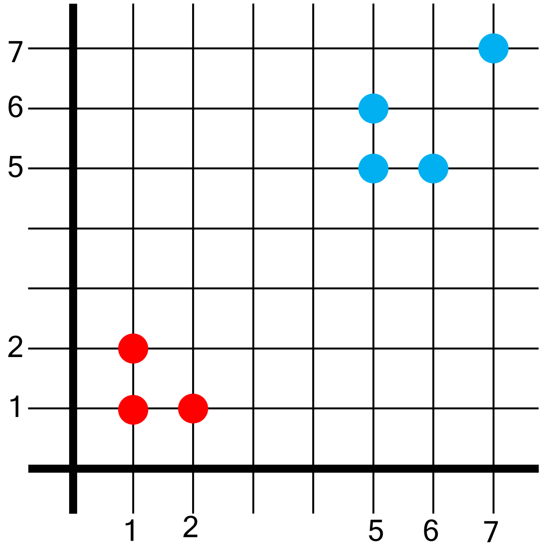


(1, 1) (2, 1) (1, 2) (5, 5) (6, 5) (5, 6) *(7, 7)*

Class A	Class B
(1, 1) (2, 1) (1, 2)	(5, 5) (6, 5) (5, 6) (7, 7)

Center : (7/3, 8/3) Center : (20/4, 19/4)

K-Means Clustering Example



(1, 1) (2, 1) (1, 2) (5, 5) (6, 5) (5, 6) (7, 7)

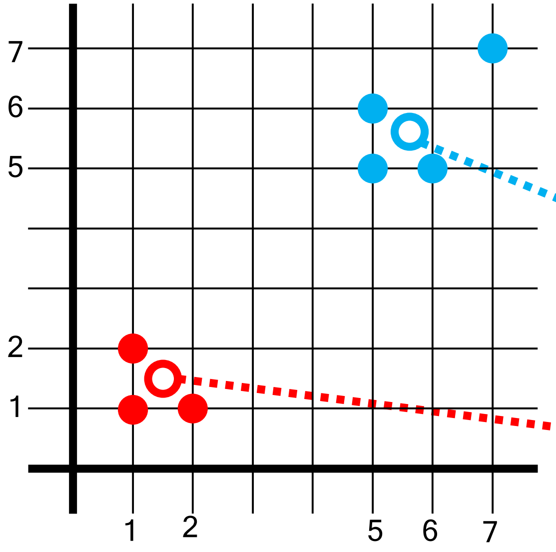
Class A

(1, 1) (2, 1) (1, 2)

Class B

(5, 5) (6, 5) (5, 6) (7, 7)

K-Means Clustering Example



(1, 1) (2, 1) (1, 2) (5, 5) (6, 5) (5, 6) (7, 7)

Class A

(1, 1) (2, 1) (1, 2)

Class B

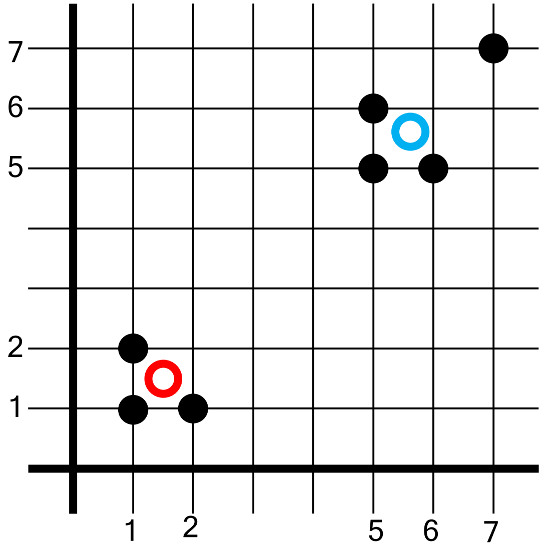
(5, 5) (6, 5) (5, 6) (7, 7)

Center : (4/3, 4/3)

Center : (23/4, 23/4)

Update the cluster means

K-Means Clustering Example



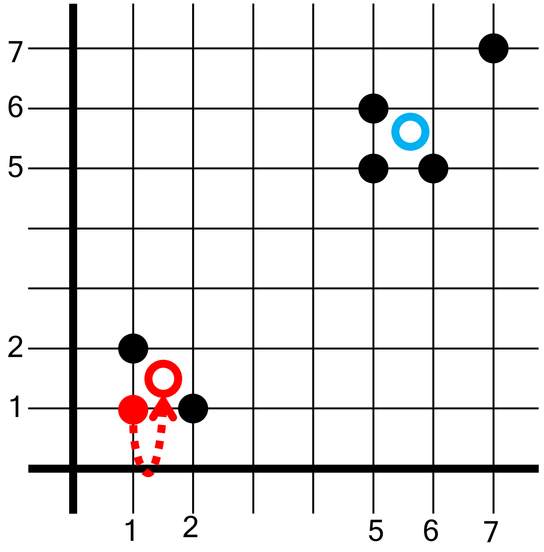
(1, 1) (2, 1) (1, 2) (5, 5) (6, 5) (5, 6) (7, 7)

Class A	Class B
(1, 1)	

Center : (4/3, 4/3)

Center : (23/4, 23/4)

K-Means Clustering Example



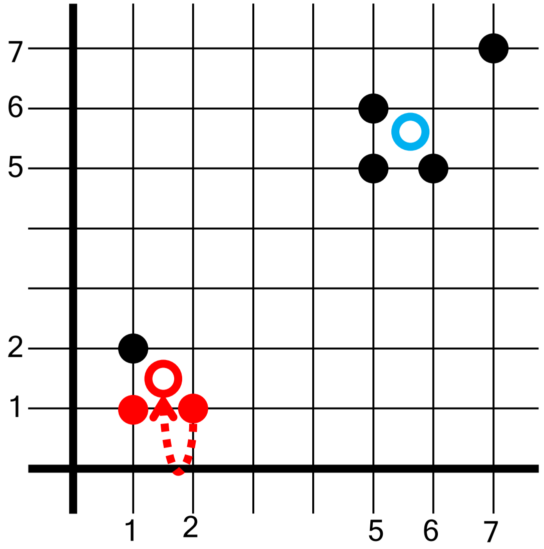
(1, 1) (2, 1) (1, 2) (5, 5) (6, 5) (5, 6) (7, 7)

Class A	Class B
(1, 1)	

Center : (4/3, 4/3)

Center : (23/4, 23/4)

K-Means Clustering Example



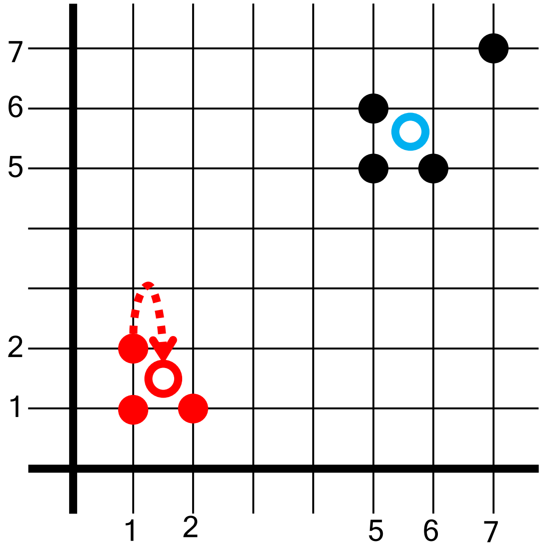
(1, 1) *(2, 1)* (1, 2) (5, 5) (6, 5) (5, 6) (7, 7)

Class A	Class B
(1, 1) (2, 1)	

Center : (4/3, 4/3)

Center : (23/4, 23/4)

K-Means Clustering Example



(1, 1) (2, 1) *(1, 2)* (5, 5) (6, 5) (5, 6) (7, 7)

Class A

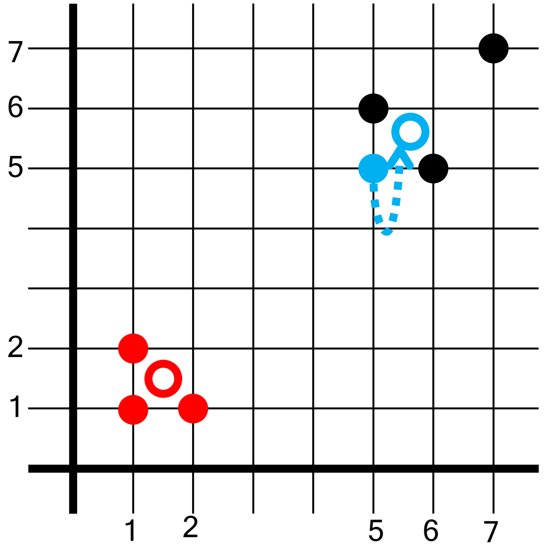
(1, 1) (2, 1) (1, 2)

Class B

Center : (4/3, 4/3)

Center : (23/4, 23/4)

K-Means Clustering Example



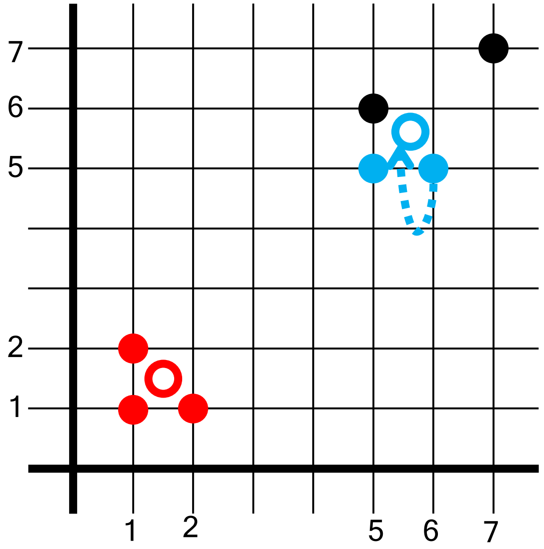
(1, 1) (2, 1) (1, 2) *(5, 5)* (6, 5) (5, 6) (7, 7)

Class A	Class B
(1, 1) (2, 1) (1, 2)	(5, 5)

Center : (4/3, 4/3)

Center : (23/4, 23/4)

K-Means Clustering Example



(1, 1) (2, 1) (1, 2) (5, 5) *(6, 5)* (5, 6) (7, 7)

Class A

(1, 1) (2, 1) (1, 2)

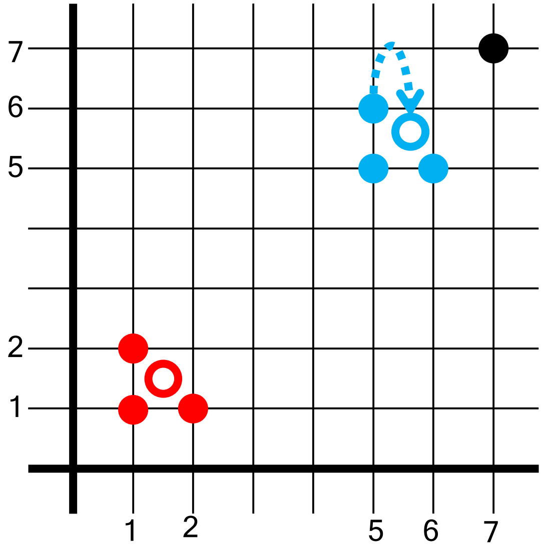
Class B

(5, 5) (6, 5)

Center : (4/3, 4/3)

Center : (23/4, 23/4)

K-Means Clustering Example



(1, 1) (2, 1) (1, 2) (5, 5) (6, 5) *(5, 6)* (7, 7)

Class A

(1, 1) (2, 1) (1, 2)

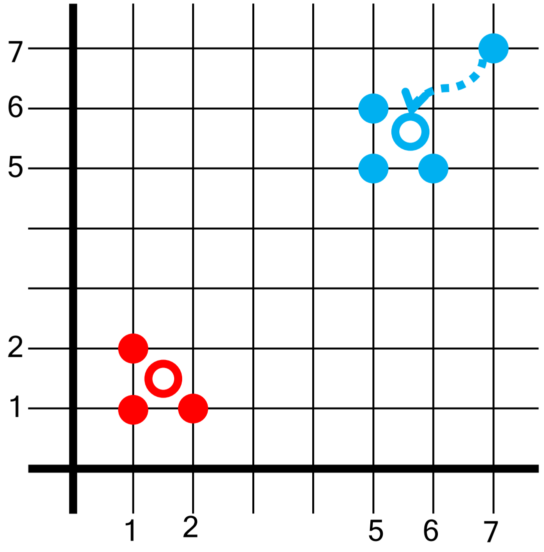
Class B

(5, 5) (6, 5) (5, 6)

Center : (4/3, 4/3)

Center : (23/4, 23/4)

K-Means Clustering Example



(1, 1) (2, 1) (1, 2) (5, 5) (6, 5) (5, 6) *(7, 7)*

Class A

(1, 1) (2, 1) (1, 2)

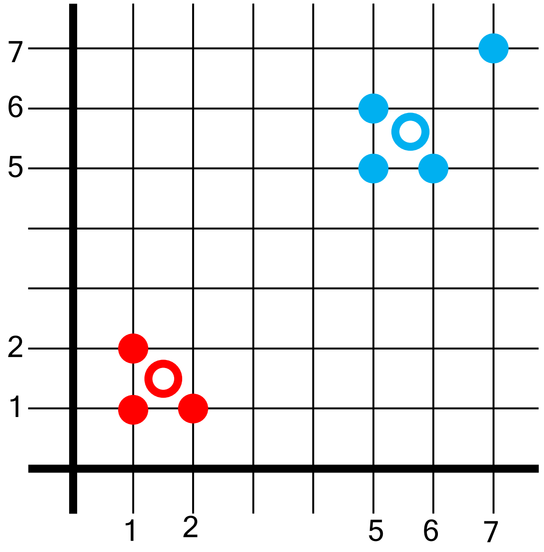
Class B

(5, 5) (6, 5) (5, 6) (7, 7)

Center : (4/3, 4/3)

Center : (23/4, 23/4)

K-Means Clustering Example



(1, 1) (2, 1) (1, 2) (5, 5) (6, 5) (5, 6) (7, 7)

Class A	Class B
(1, 1) (2, 1) (1, 2)	(5, 5) (6, 5) (5, 6) (7, 7)

Center : (4/3, 4/3) Center : (23/4, 23/4)

No change $\Rightarrow$ STOP

What is the Problem of the K-Means Method?

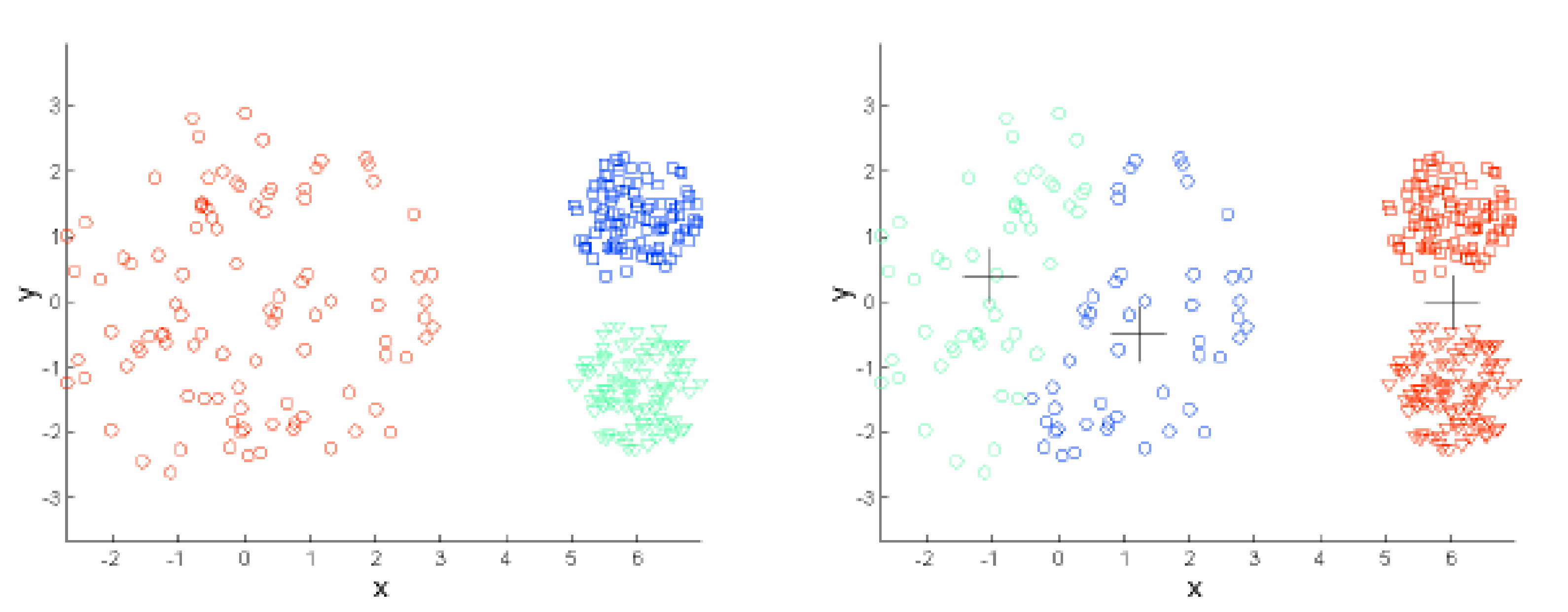
- The k-means algorithm is sensitive to outliers !
  - Since an object with an extremely large value may substantially distort the distribution of the data.
- K-Medoids: Instead of taking the **mean** value of the object in a cluster as a reference point, **medoids** can be used

$$x_{\text{medoid}} = \arg\min_{y \in \mathcal{X}} \sum_{i=1}^{n} d(y, x_i).$$

Drawbacks of K-Means

- K-Means has problems when clusters are of differing
  - Sizes
  - Densities
  - Non-spherical shapes
- Problems with outliers

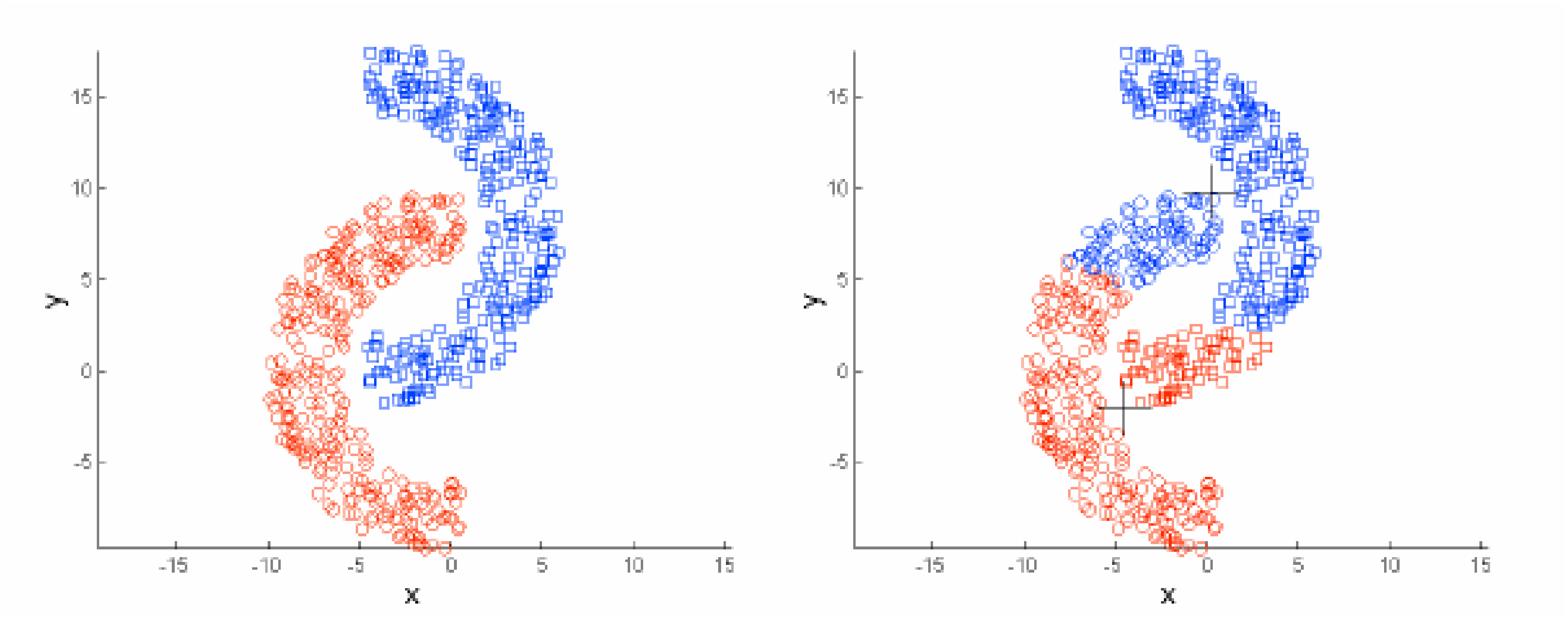
Drawbacks of K-Means



Original Points

K-means (3 Clusters)

Drawbacks of K-Means



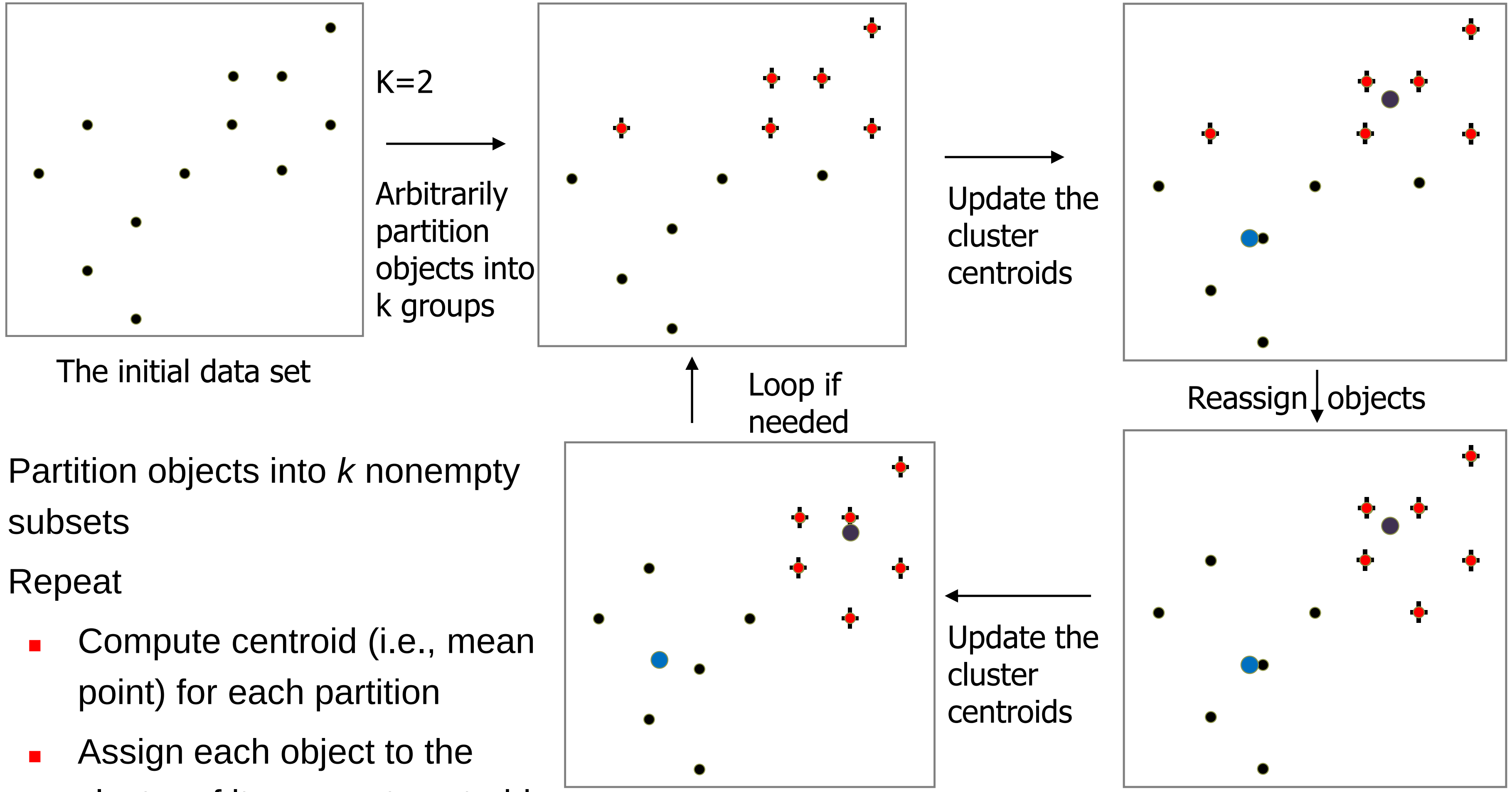
Original Points

K-means (2 Clusters)

Drawbacks of K-Means

- K-Means has problems when clusters are of differing
  - Sizes
  - Densities
  - Non-spherical shapes
- Problems with outliers

An Example of *K-Means* Clustering

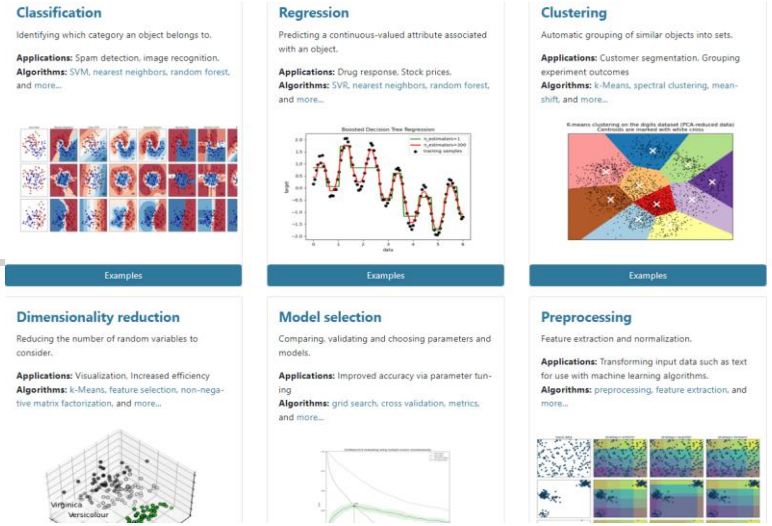


- Partition objects into *k* nonempty subsets
- Repeat
  - Compute centroid (i.e., mean point) for each partition
  - Assign each object to the cluster of its nearest centroid
- Until no change

Python – K-Means Clustering

Packages

- Scikit-learn (https://scikit-learn.org/)
  - Machine learning tool



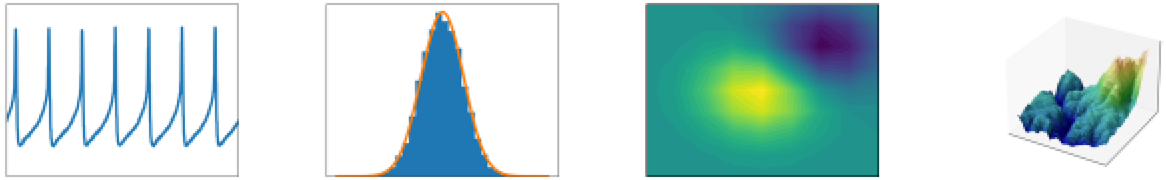
- Pandas (https://pandas.pydata.org/)
  - Data analysis and manipulation tool

	BandName	WavelengthMax	WavelengthMin
0	CoastalAerosol	450	430
1	Blue	510	450
2	Green	590	530
3	Red	670	640
4	NearInfrared	880	850
5	ShortWaveInfrared_1	1650	1570
6	ShortWaveInfrared_2	2290	2110
7	Cirrus	1380	1360

- Matplotlib (https://matplotlib.org/)

Matplotlib: Visualization with Python

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python.



Download the Dataset

- Download cluster2.csv from
- https://hyu-my.sharepoint.com/:f:/g/personal/whjung_hanyang_ac_kr/Ev34n7L_Z0BErxWCad88rsAB6IdZCa0cUn7_Rd0ryYYYWQ?e=bqVe6k
- PWD: ai202102
- Save the csv file in the same directory as the source file (.ipynb)

Import Libraries

A *magic command* to make figures are visible in the *jupyter notebook*

```
%matplotlib inline
```

Import libraries

```
import pandas as pd
import matplotlib.pyplot as plt
```

A Python plotting library

Provides a MATLAB-like plotting framework

Import the Dataset

- Import the dataset from the csv file

```
df = pd.read_csv('cluster2.csv')
print("Dimensions of the data = {}".format(
    df.shape))
```

The value is printed in "{}"

The printed result:

2-dimensional 1300 data points

```
Dimensions of the data = (1300, 2)
```

Import the Dataset

```
df[:5]
```

	X	Y
0	1.070487	1.328147
1	1.072777	1.191249
2	0.328029	1.261713
3	0.600926	1.254465
4	0.759281	1.284541

Import the Dataset

- Convert `df` to array

```
X = df.values
X[:5]
```

```
array([[1.07048688, 1.3281469 ],
       [1.07277723, 1.19124898],
       [0.3280287 , 1.26171275],
       [0.60092577, 1.2544653 ],
       [0.75928098, 1.28454059]])
```

Plotting the Dataset

```
# Set the size of the figure
plt.figure(figsize=(5, 5))

# Plot the data points
plt.scatter(X[:, 0], X[:, 1], s=4)

plt.show()  # Print the figure
```

Plotting the Dataset

```
# Set the size of the figure
plt.figure(figsize=(5, 5))

# Plot the data points
plt.scatter(X[:, 0], X[:, 1], s=4)

plt.show()  # Print the figure
```

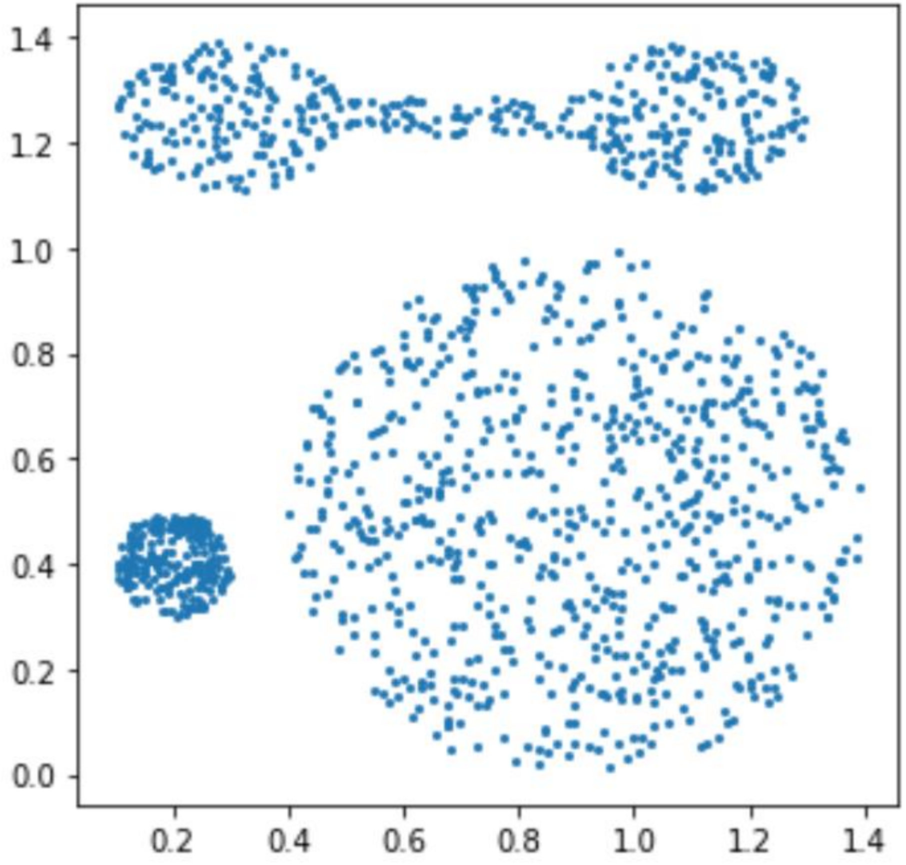
width, height in inches

x and y coordinates

Performs a scatter plot

The size of each points in the figure

The Plotted Data

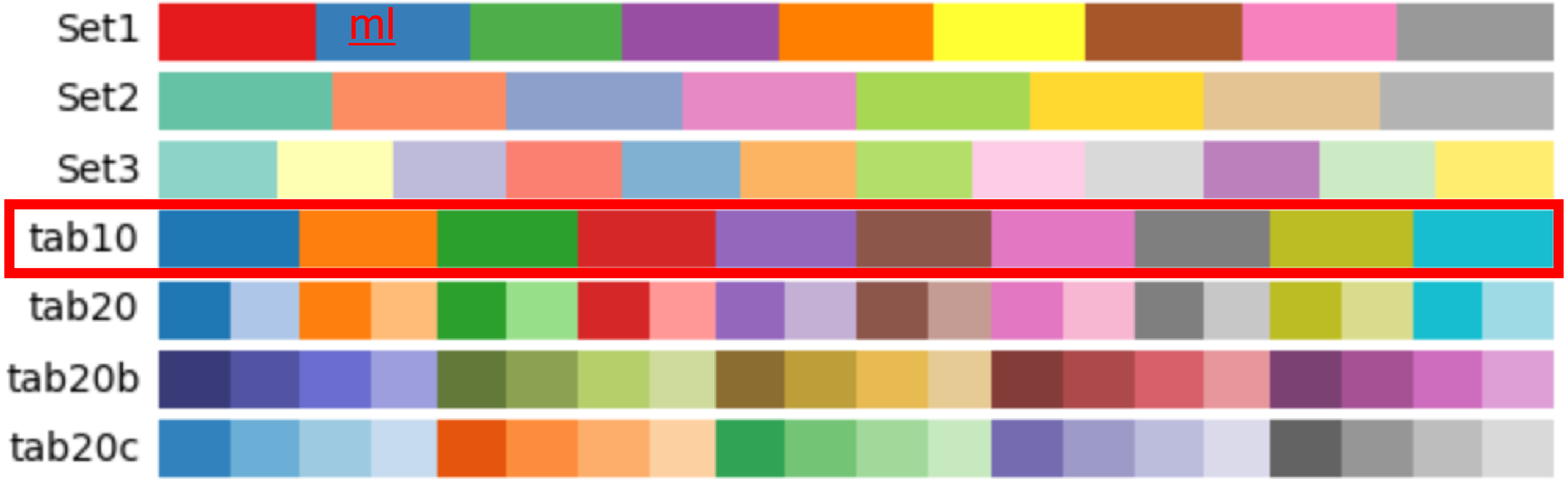


Choosing Colormaps

```
cmap = "tab10"
```

- Store the name of *colormap* in global variable *cmap*
  - For mapping cluster indices to colors

https://matplotlib.org/tutorials/colors/colormaps.html



K-Means Clustering

```
from sklearn.cluster import KMeans
k_means = KMeans(n_clusters=4,
                 random_state=0)
```

- Create an object 'k_means' for K-means clustering
  - n_clusters: the number of clusters
  - random_state: the random seed for centroid initialization
  - max_iter: maximum number of iterations

Predicted Cluster Indices

- Perform clustering and output the cluster index for each data point

```
y_pred = k_means.fit_predict(X)
print(y_pred[:10])
```

```
[0 0 3 3 0 3 3 0 3 0]
```

The cluster index of the first 10 points

Cluster Centers

```
print(k_means.cluster_centers_)
```

```
[[1.01379946 1.15444051]
 [1.03377029 0.43895496]
 [0.37933677 0.43852558]
 [0.34274065 1.2515179 ]]
```

Plotting Clustering Results

```
plt.figure(figsize=(5, 5))
plt.scatter(X[:, 0], X[:, 1],
            c=y_pred, s=4, cmap=cmap)
# 'c': The color of each point

plt.show()
```

The color of each point is determined by the cluster index

Set the colormap

The Clustering Result