CS 383: Machine Learning

Prof Adam Poliak

Fall 2024

11/26/2024

Lecture 28

Announcements – Remaining Assignments

HW07: due Wednesday 11/27

HW08: due Friday 12/06 (might extend this too)

No Project Presentations – due end of finals period

Outline

Deep Learning Review

Midterm Review practice problems

Unsupervised learning

Deep Learning

FNN

Solutions to fixed-length input problem

Attention/Self-Attention/Transformers

Outline

Deep Learning Review

Midterm Review practice problems

Unsupervised learning

Question 1

- First compute weighted leaf labels

$$P(+ \mid \text{sun}) = \frac{\frac{1}{3}}{\frac{1}{3} + \frac{1}{8} + \frac{1}{8}} = \frac{4}{7} \geq 0.5 \quad \Rightarrow +$$

Question 1

- First compute weighted leaf labels

$$P(+ \mid \text{sun}) = \frac{\frac{1}{3}}{\frac{1}{3} + \frac{1}{8} + \frac{1}{8}} = \frac{4}{7} \geq 0.5 \quad \Rightarrow +$$

$$P(+ \mid \text{rain}) = \frac{\frac{1}{12}}{\frac{1}{12} + \frac{1}{6} + \frac{1}{6}} = \frac{1}{5} < 0.5 \quad \Rightarrow -$$

Question 1

- Based on these labels, we can say which training points are misclassified

$$\epsilon_t = \frac{1}{8} + \frac{1}{8} + \frac{1}{12} = \frac{1}{3}$$

Question 1

- Based on these labels, we can say which training points are misclassified

$$\epsilon_t = \frac{1}{8} + \frac{1}{8} + \frac{1}{12} = \frac{1}{3}$$

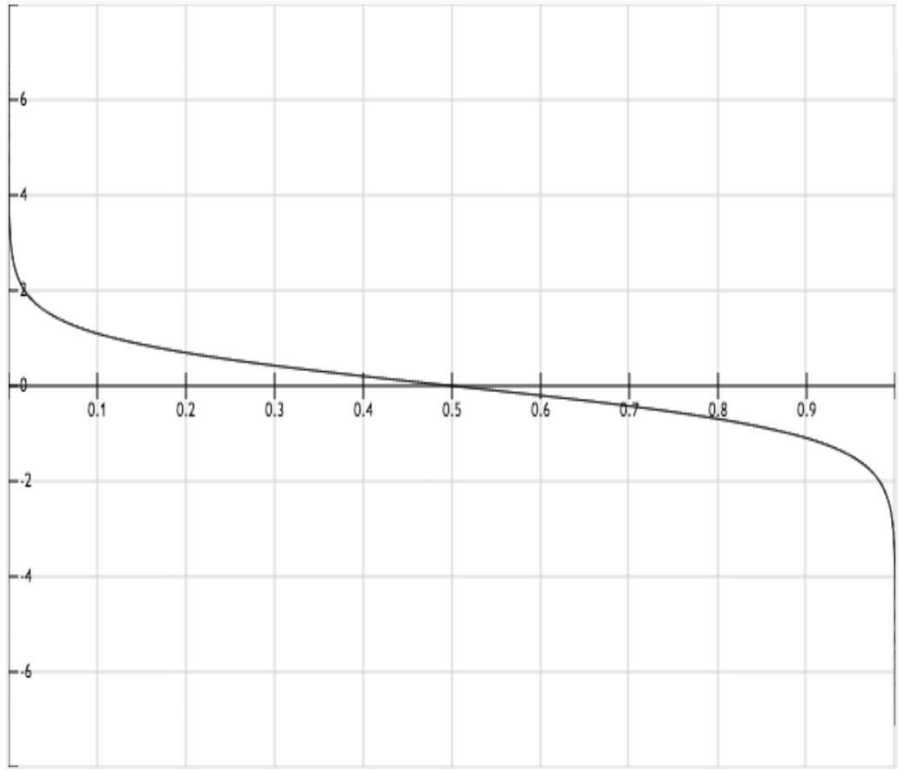
- Note if this was > 0.5, we should have chosen different leaf labels! So this “flipping” step should happen automatically
  - (exception for pathological cases)

Question 1

- Score function:

$$\alpha_t = \frac{1}{2} \ln \left(\frac{1-\epsilon_t}{\epsilon_t} \right)$$

- Fraction: accuracy/error
- As error -> 0, score becomes high
- As error -> ½, score goes to 0



Question 2

- r = 1/3, probability of one classifier being wrong
- T = 5, number of classifiers
- R = number of votes for the wrong class
- If R=3,4,5 then we will vote for the wrong class overall

Question 2

- This analysis assumed classifiers were independent!
- What if they are not? How did Random Forests help us decorrelate classifiers?

Question 2

- This analysis assumed classifiers were independent!
- What if they are not? How did Random Forests help us decorrelate classifiers?
- Note about Bagging: choosing *n* with resampling actually does produce a very different dataset
  - As *n* increases, roughly 0.37 not chosen each time

Question 3

- False Positive Rate: FP / (FP + TN)

- True Positive Rate: TP / (TP + FN)

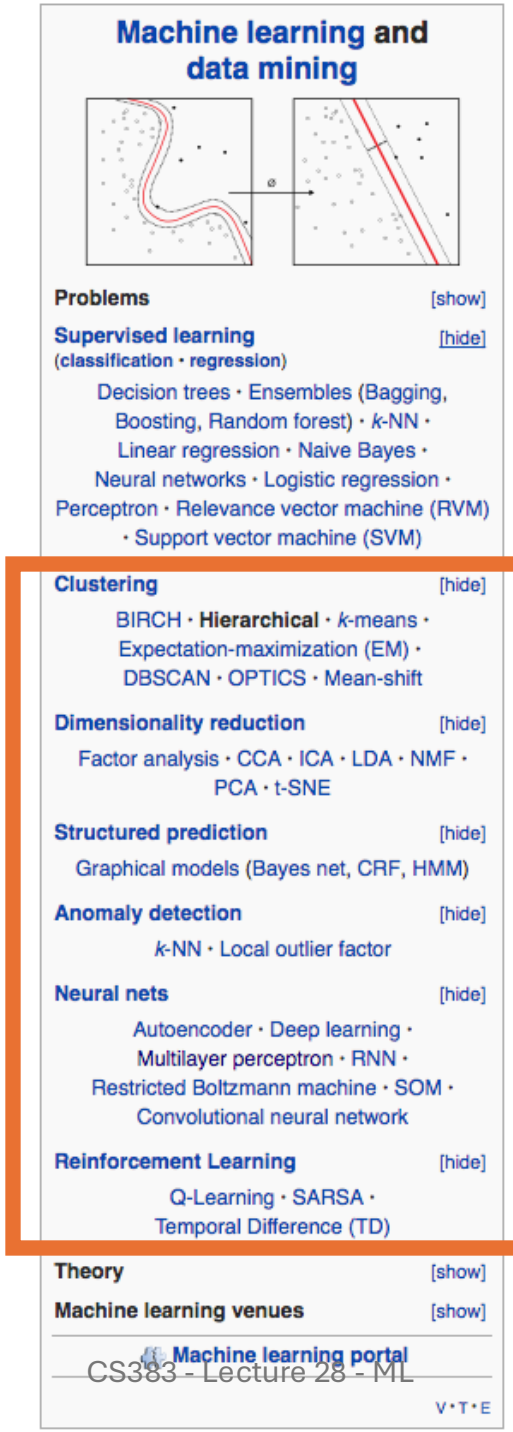
Outline

Deep Learning Review

Midterm Review practice problems

Unsupervised learning

Supervised Learning:
makes use of examples where we know the underlying "truth" (label/output)



Unsupervised Learning:
Learn underlying structure or features without labeled training data

Image: wikipedia

Unsupervised learning: 3 main areas

1) Clustering: group data points into clusters based on features only

2) Dimensionality reduction: remove feature correlation, compress data, visualize data

3) Structured prediction: model latent variables (example: Hidden Markov Models)

Unsupervised Algorithms

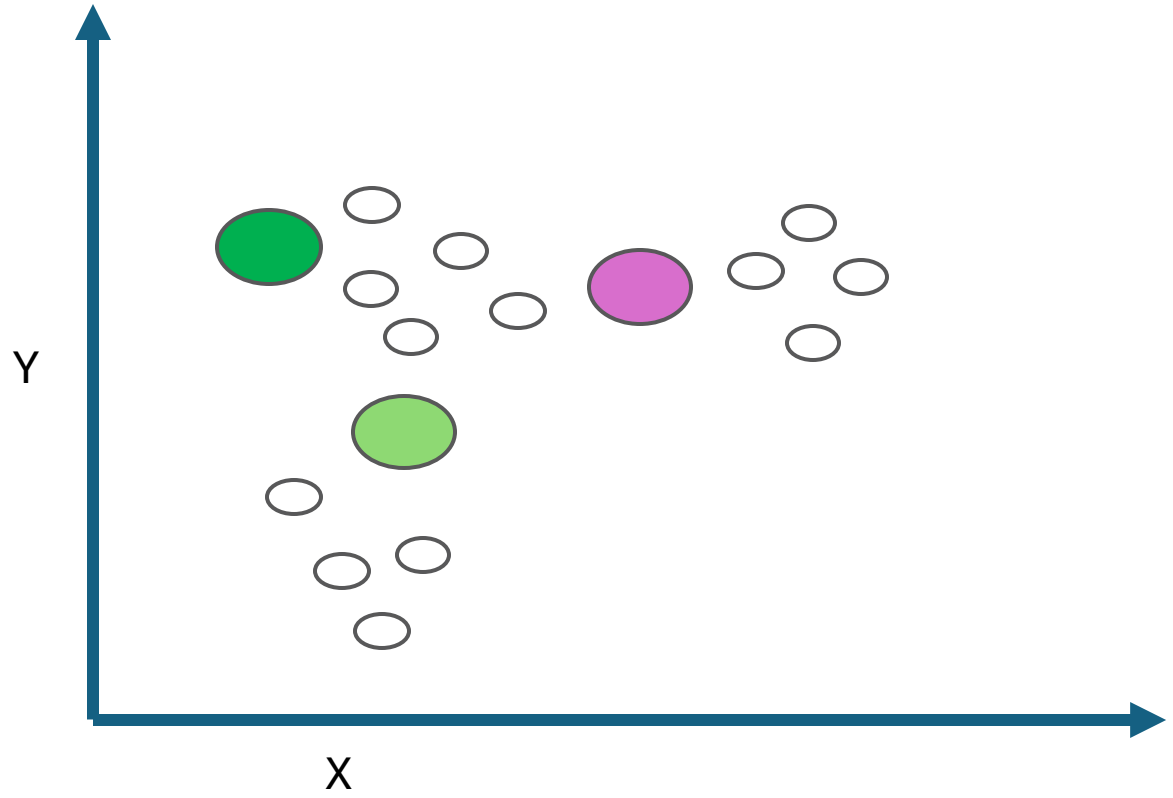
- K-means
- Gaussian Mixture Models (GMM)
- Principle Component Analysis (PCA)

K-means Algorithms

1. Initialize: Randomly pick K points as cluster centers

Randomly pick K points as centers

- Example: 2D point patterns



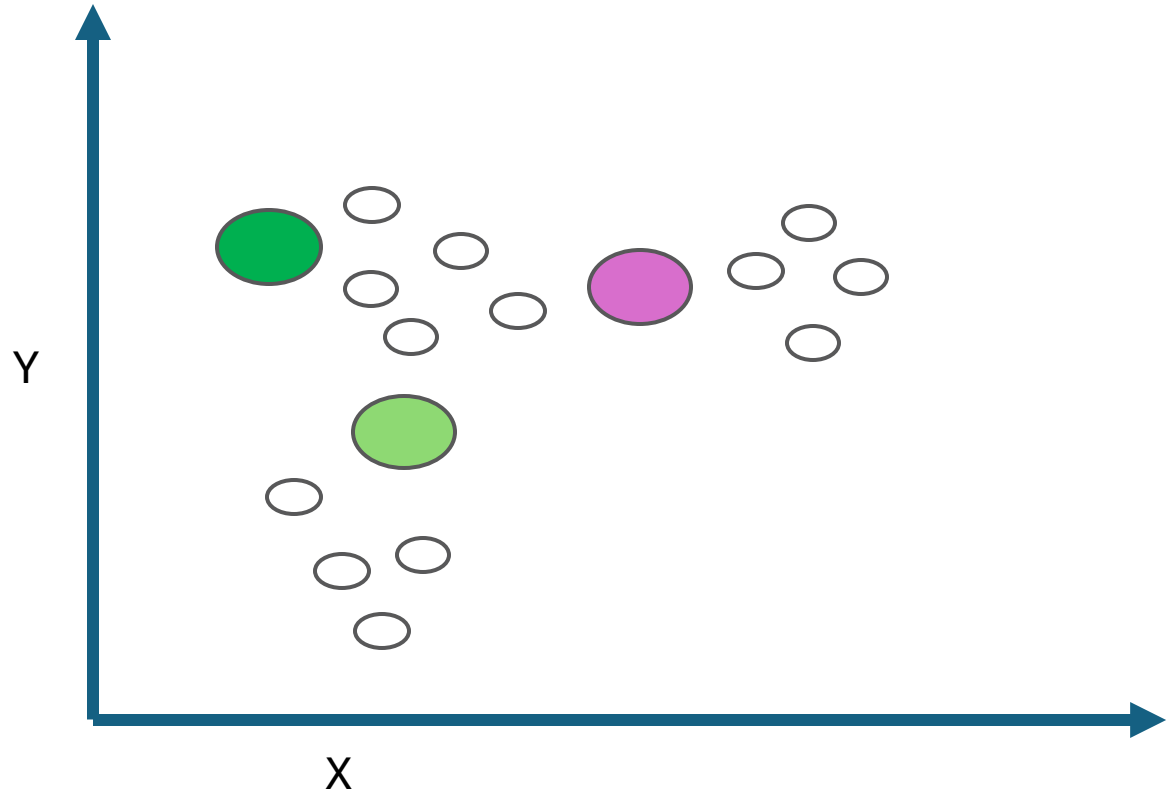
K-means Algorithms

1. Initialize: Randomly pick K points as cluster centers

2. Assign data points to each cluster
    1. Based on distance between point and cluster's center

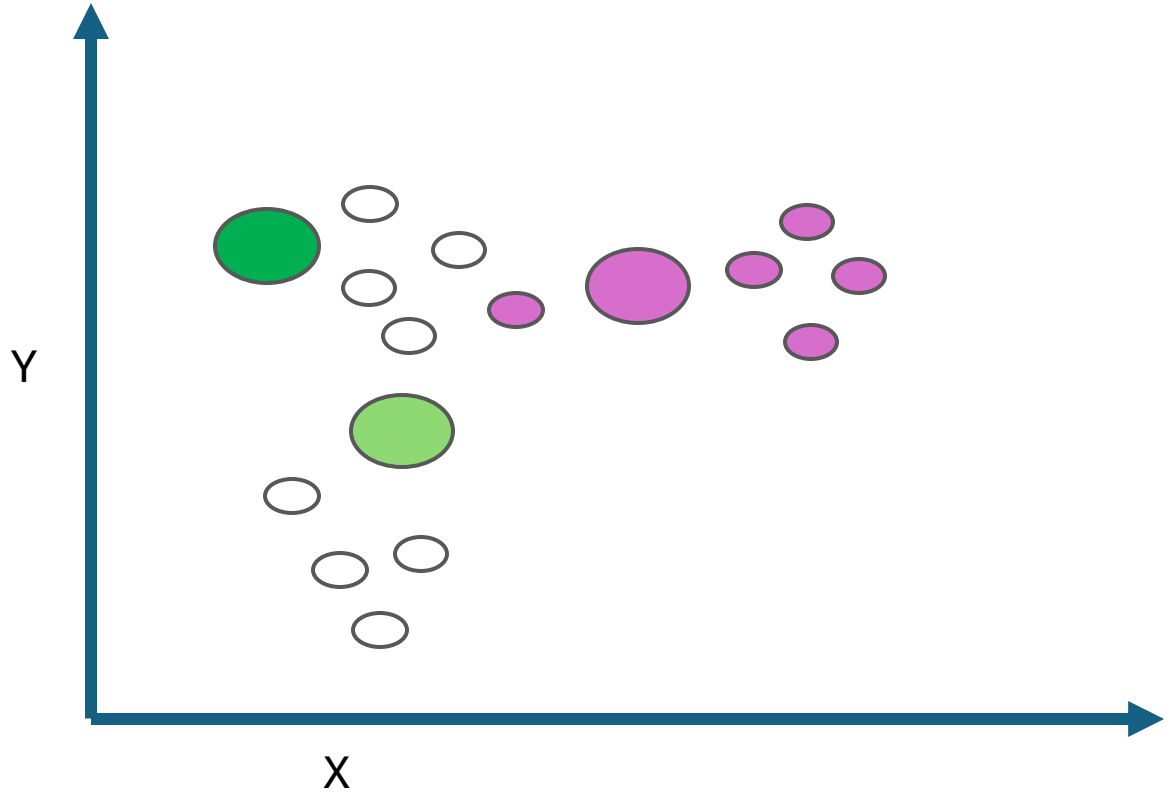
Assign data points to each cluster

- Example: 2D point patterns



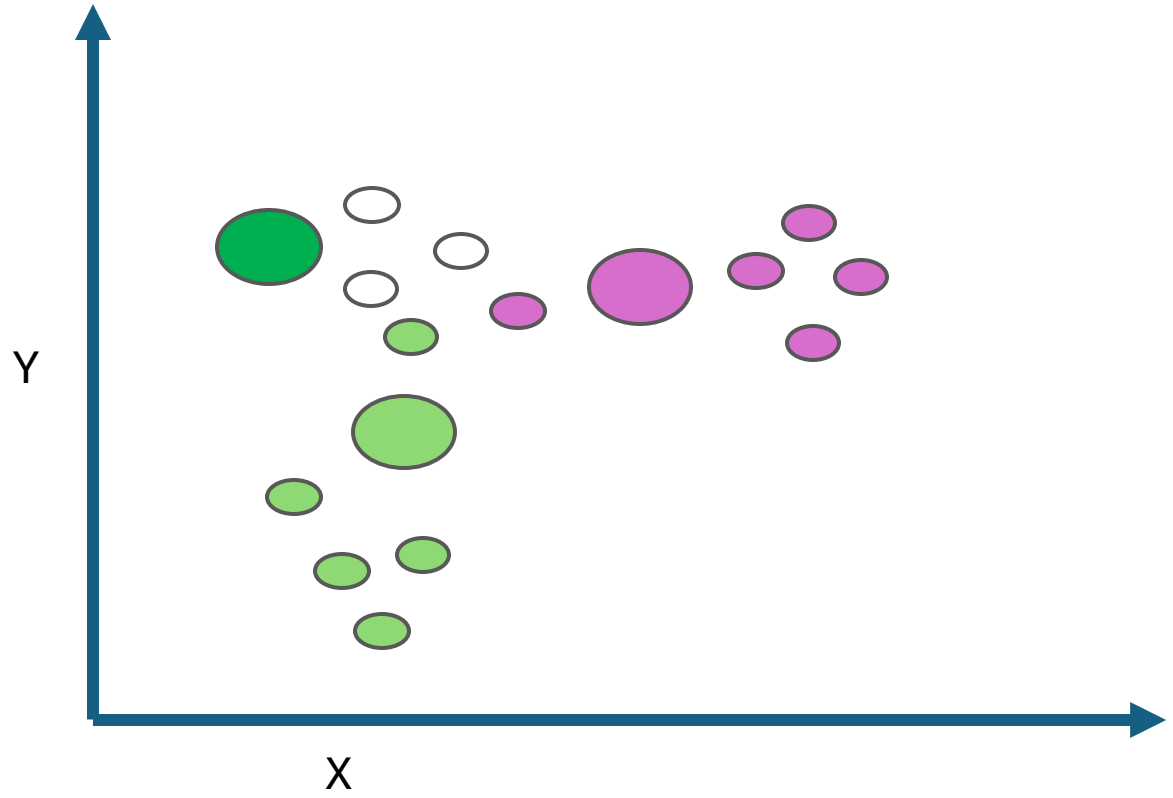
Assign data points to each cluster

- Example: 2D point patterns



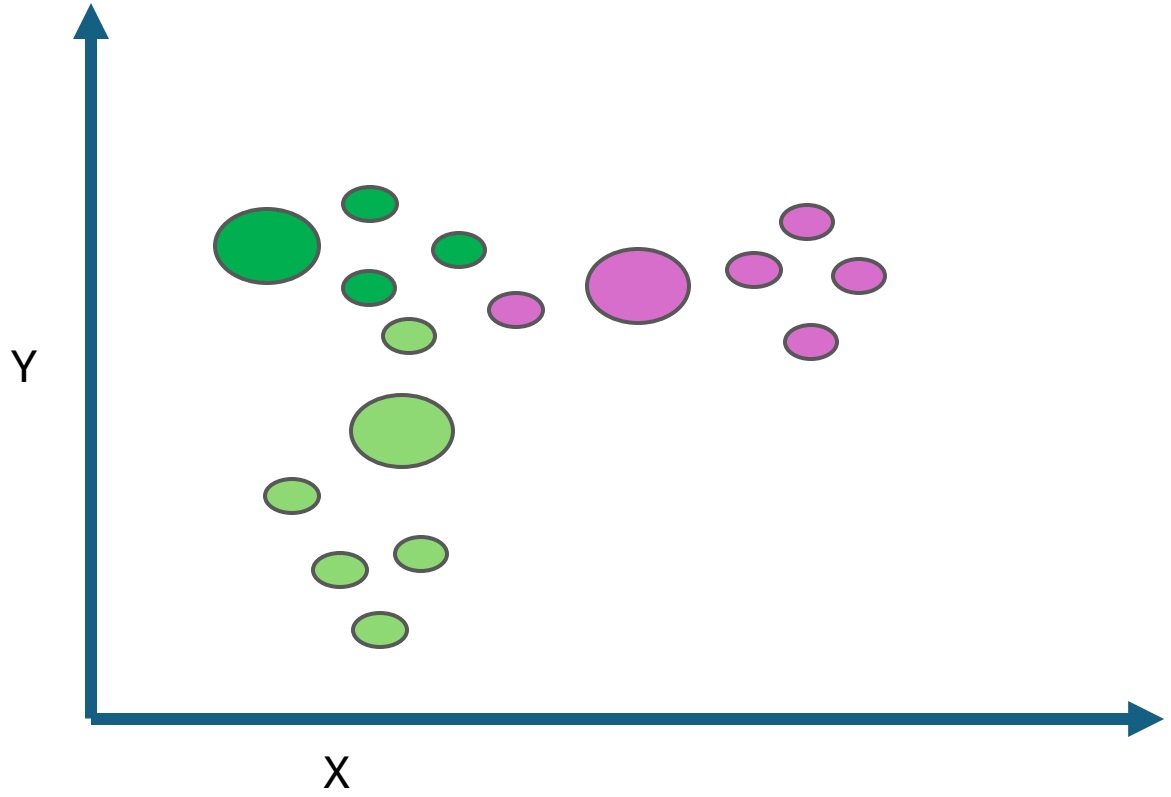
Assign data points to each cluster

- Example: 2D point patterns



Assign data points to each cluster

- Example: 2D point patterns



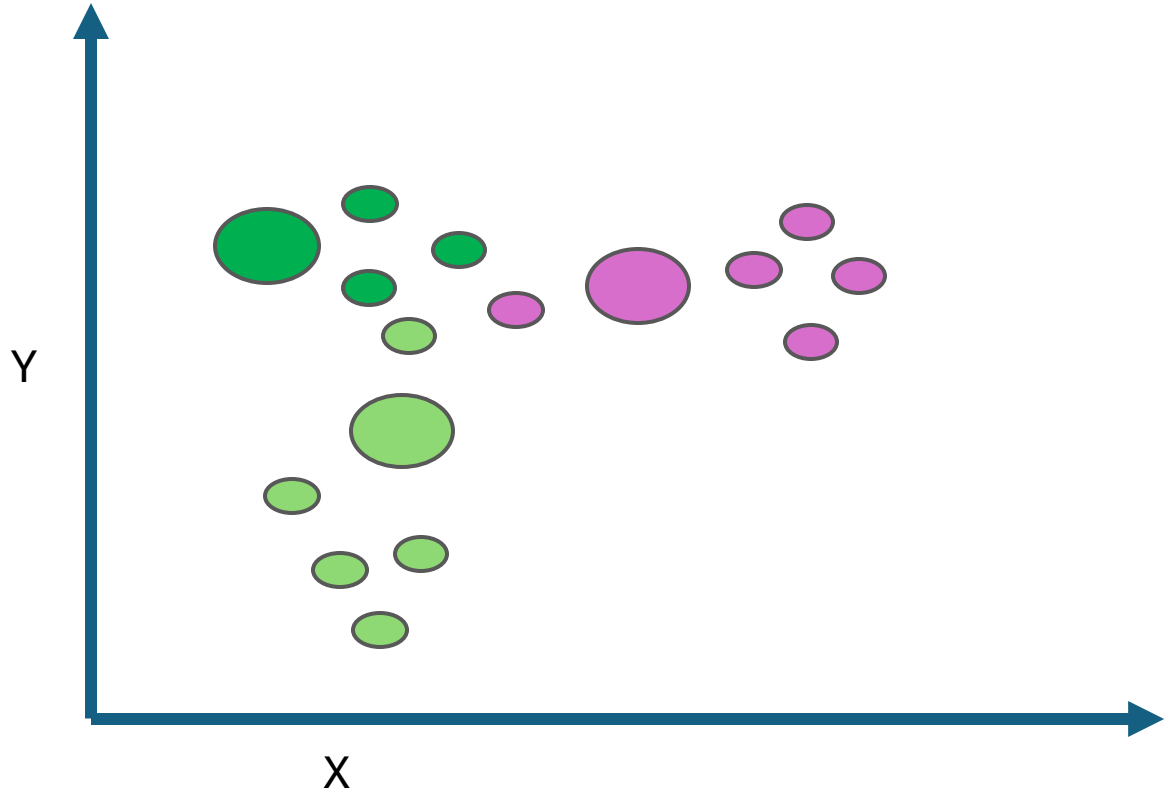
K-means Algorithms

1. Initialize: Randomly pick K points as cluster centers

2. Assign data points to each cluster
    1. Based on distance between point and cluster's center
3. Update the center of each cluster
    1. The average of its assigned points

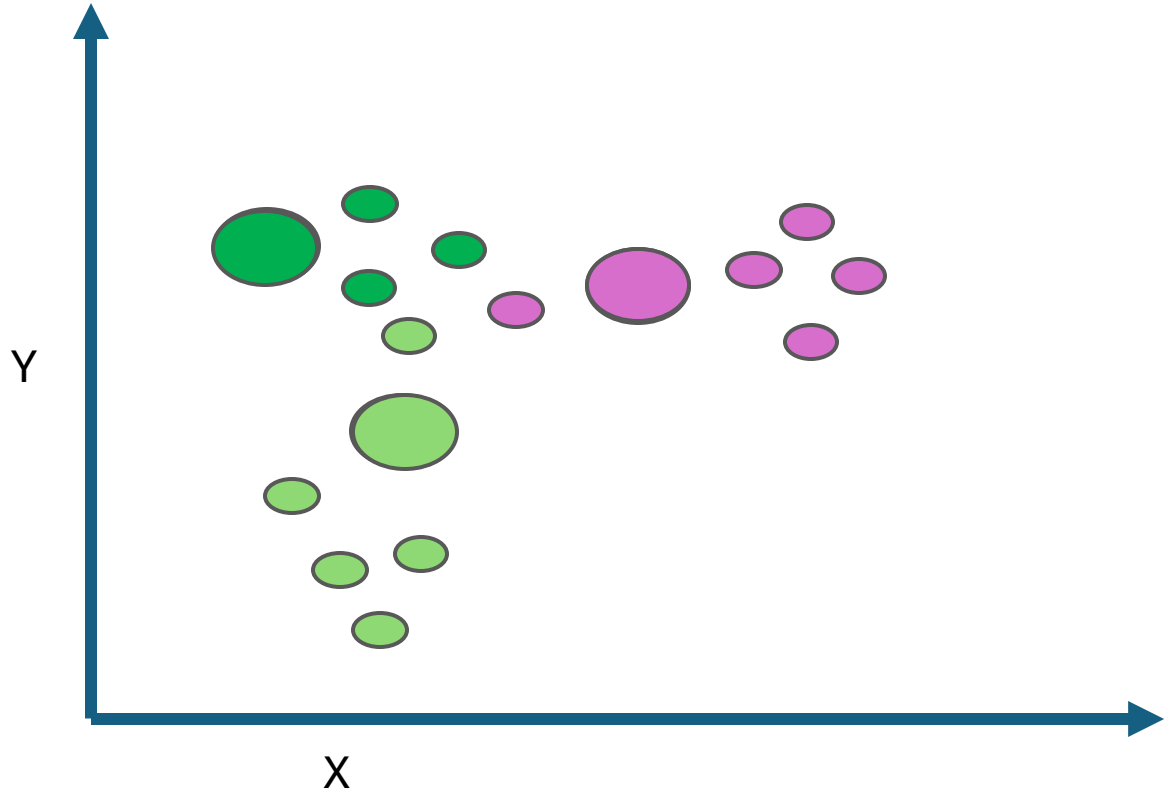
Update Centers

- Example: 2D point patterns



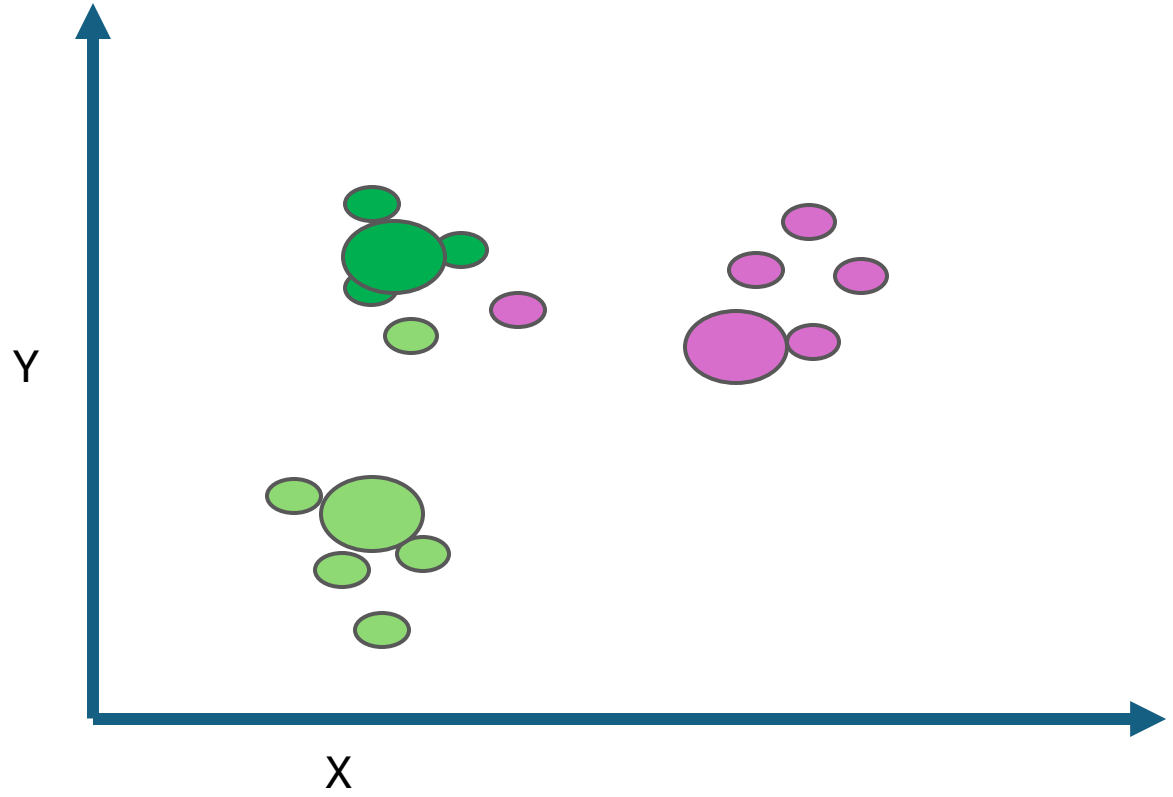
Update Centers

- Example: 2D point patterns



Updated Centers

- Example: 2D point patterns



K-means Algorithms

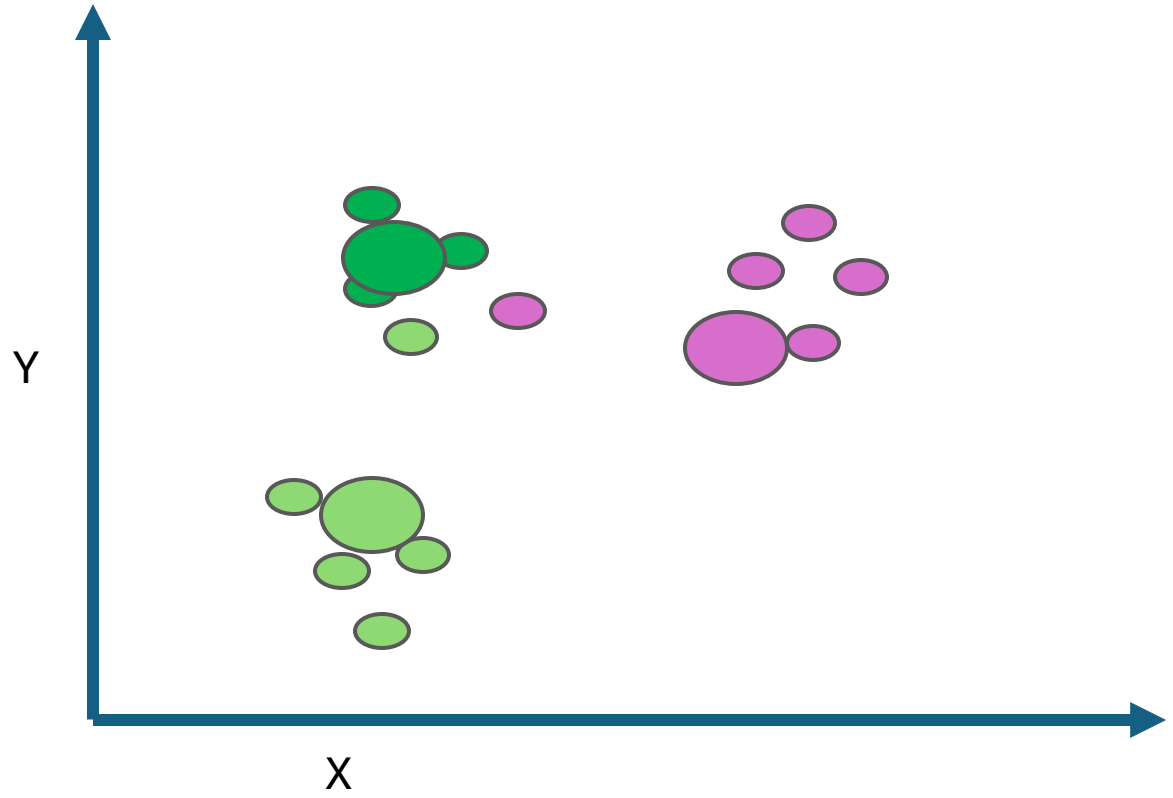
1. Initialize: Randomly pick K points as cluster centers

2. Assign data points to each cluster
    1. Based on distance between point and cluster's center
3. Update the center of each cluster
    1. The average of its assigned points

4. Repeat 2 & 3 until the assignments stop changing

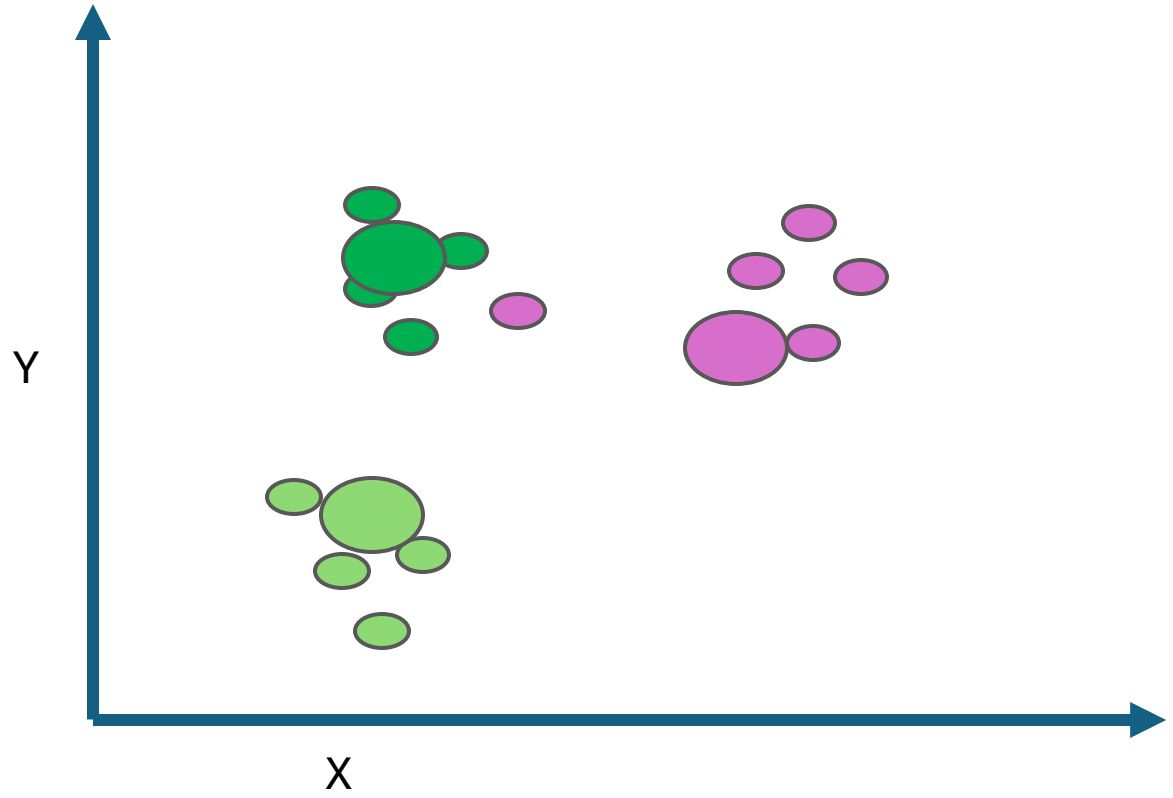
Reassign data points to each cluster

- Example: 2D point patterns



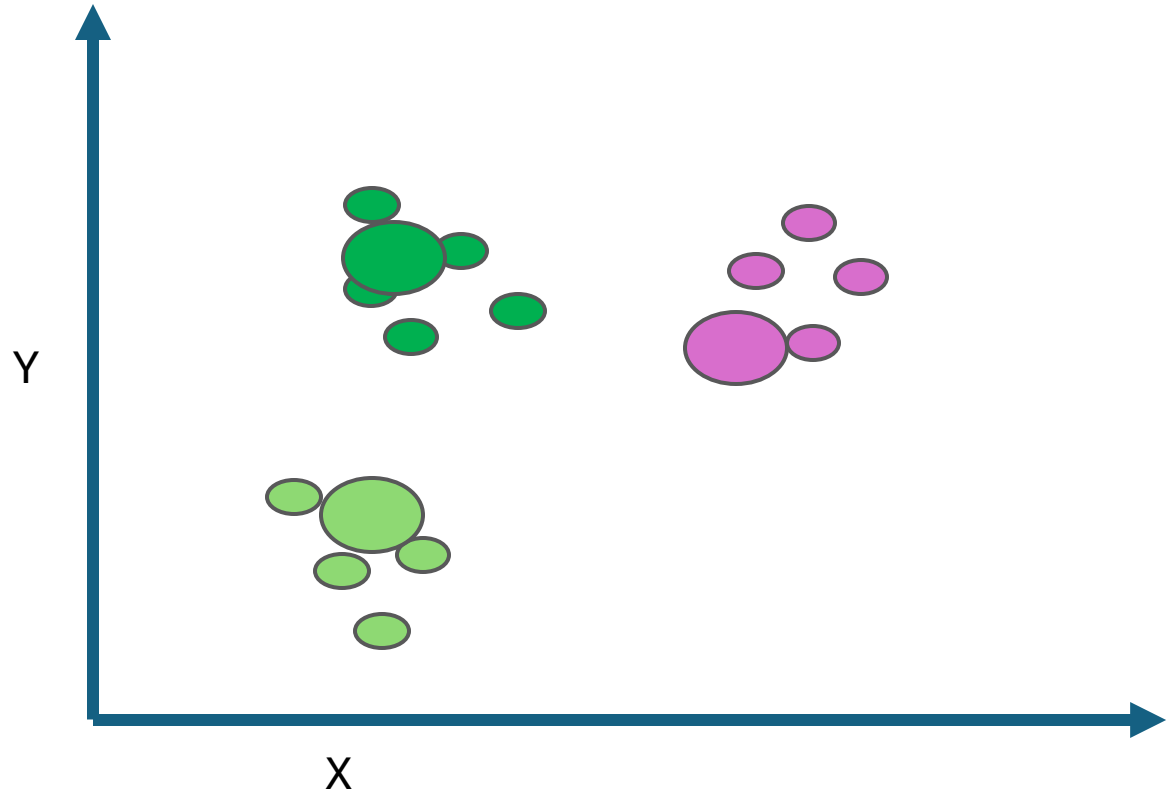
Reassign data points to each cluster

- Example: 2D point patterns



Reassign data points to each cluster

- Example: 2D point patterns



K-means Algorithms

1. Initialize: Randomly pick K points as cluster centers

2. Assign data points to each cluster
    1. Based on distance between point and cluster's center
3. Update the center of each cluster
    1. The average of its assigned points

4. Repeat 2 & 3 until the assignments stop changing

K-means Algorithms

1. Initialize: Randomly pick K points as cluster centers

2. Assign data points to each cluster
    1. Based on **distance between** point and cluster's center
3. Update the center of each cluster
    1. The average of its assigned points

4. Repeat 2 & 3 until the assignments stop changing