

Welcome to **INTERNSHIP STUDIO**

Module 04 | Lesson 04

K nearest neighbour

K- Nearest Neighbor (KNN)

Objective:

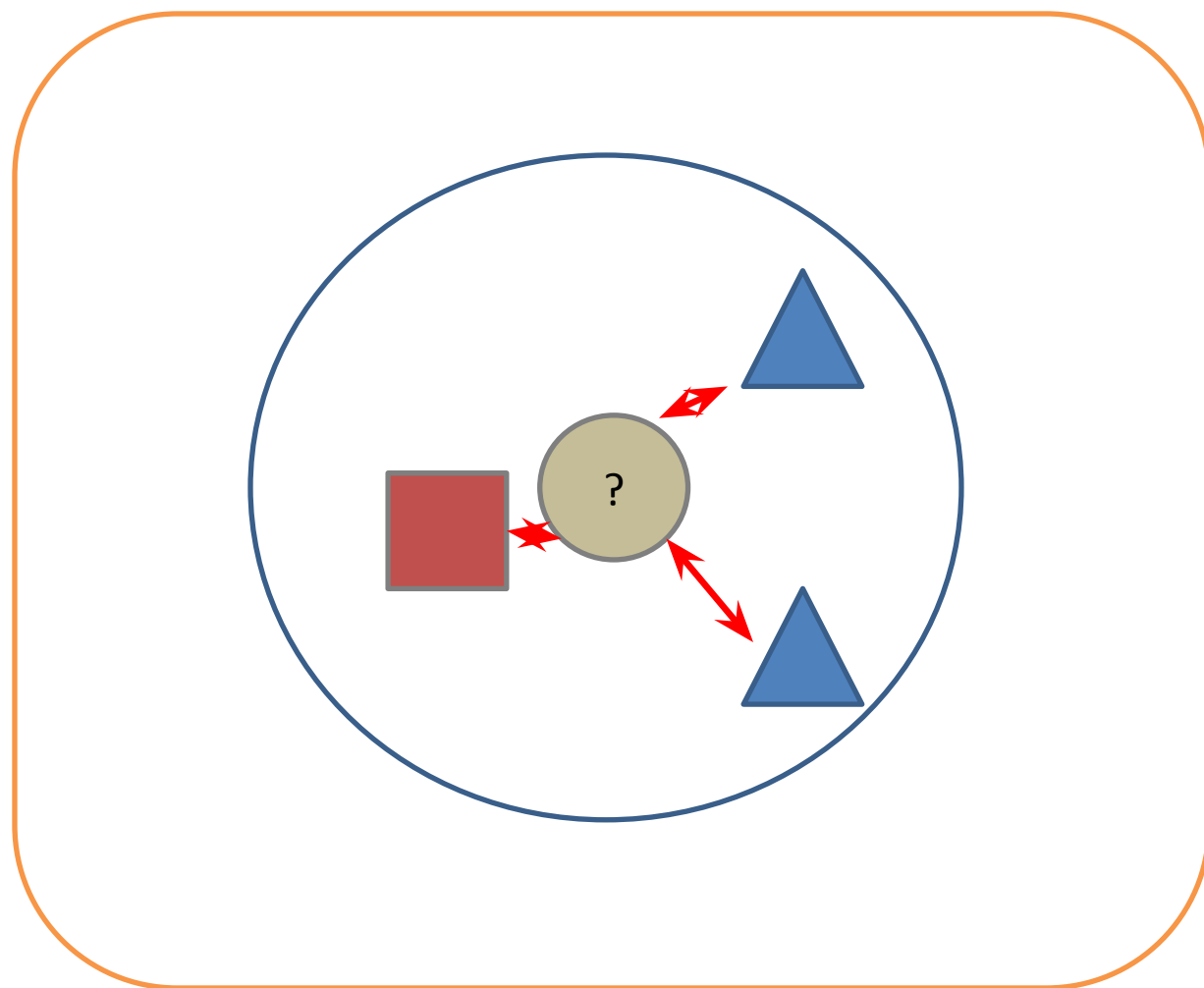
K Nearest Neighbor is an instance based learning approach that classifies unlabeled data to any of the classes based on their similarity and majority voting.

K- Nearest Neighbor (KNN)

Requirements:

1. The set of stored patterns.
2. Distance metric to compute distance between patterns.
3. The value of k (number of nearest neighbor)
 - *Usually we take distance as Euclidian distance.*
 - *K should be any positive integer (odd numbers to avoid ties).*
 - *Attributes may have to be scaled to prevent distance measures from being dominated by one of the attributes*

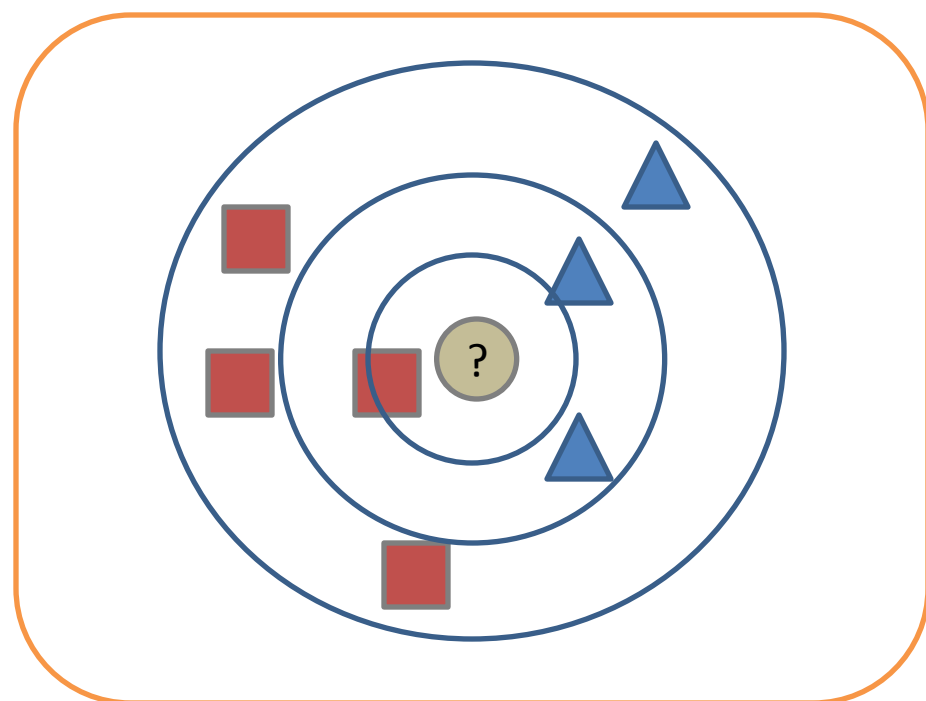
K- Nearest Neighbor (KNN)



Classification Process:

1. Take unlabeled data point .
2. Measure its distance from each of the class.
3. Use class labels of nearest neighbors to determine the class label of unknown pattern (e.g., by taking majority vote)

K- Nearest Neighbor (KNN)



□ $k = 1$:

- Belongs to square class

□ $k = 3$:

- Belongs to triangle class

□ $k = 7$:

- Belongs to square class

Choosing the value of k :

- If k is too small, sensitive to noise points
- If k is too large, neighborhood may include points from other classes

K- Nearest Neighbor (KNN)

Advantages

- ✓ Doesn't require apriori probability and distribution function.
- ✓ Easy to implement.

Disadvantages

- ✓ If there are n elements complexity will be $O(n)$ (computationally extensive)
- ✓ Sometime majority can dominate the classification.
- ✓ Accuracy can be severely degraded by the presence of noisy or irrelevant features