Example of k-Nearest Neighbors (k-NN)

1. k-Nearest Neighbors (k-NN)

Break down the **k-Nearest Neighbors (k-NN)** algorithm step by step with specific values and calculations, focusing on how we can compute distances and determine the class of a new data point.

Example Scenario

Suppose we have the following 2D dataset where each point is classified as either **Red (R)** or **Blue (B)**:

Data Point	x_1 (feature 1)	x_2 (feature 2)	Class
d_1	2	4	Red
d_2	4	6	Red
d_3	4	2	Blue
d_4	6	4	Blue
d_5	6	6	Red

Now, we are given a new data point:

$$d_{\text{new}} = (5, 5)$$

Steps of k-NN

Step 1: **Calculate the Distance Between d_{new} and All Existing Data Points**

We'll calculate the **Euclidean distance** between d_{new} and each data point. The formula for Euclidean distance between two points $P(x_1, y_1)$ and $Q(x_2, y_2)$ is:

$$\text{Distance} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Let's compute this for each data point in the dataset:

- **Distance between d_{new} and $d_1(2, 4)$:**

$$\text{Distance}(d_1, d_{\text{new}}) = \sqrt{(5-2)^2 + (5-4)^2} = \sqrt{9+1} = \sqrt{10} \approx 3.16$$

- **Distance between d_{new} and $d_2(4, 6)$:**

$$\text{Distance}(d_2, d_{\text{new}}) = \sqrt{(5-4)^2 + (5-6)^2} = \sqrt{1+1} = \sqrt{2} \approx 1.41$$

- **Distance between d_{new} and $d_3(4, 2)$:**

$$\text{Distance}(d_3, d_{\text{new}}) = \sqrt{(5-4)^2 + (5-2)^2} = \sqrt{1+9} = \sqrt{10} \approx 3.16$$

- **Distance between d_{new} and $d_4(6, 4)$:**

$$\text{Distance}(d_4, d_{\text{new}}) = \sqrt{(5-6)^2 + (5-4)^2} = \sqrt{1+1} = \sqrt{2} \approx 1.41$$

- **Distance between d_{new} and $d_5(6, 6)$:**

$$\text{Distance}(d_5, d_{\text{new}}) = \sqrt{(5-6)^2 + (5-6)^2} = \sqrt{1+1} = \sqrt{2} \approx 1.41$$

Step 2: **Sort the Distances**

Now that we have all the distances, we can sort them in ascending order:

Data Point	Distance	Class
d_2	1.41	Red
d_4	1.41	Blue
d_5	1.41	Red
d_1	3.16	Red
d_3	3.16	Blue

Step 3: **Choose the Value of k and Determine the Nearest Neighbors**

Let's set $k = 3$, which means we will consider the 3 nearest neighbors. From the sorted list of distances, the three nearest neighbors are:

- d_2 (Red, distance = 1.41)
- d_4 (Blue, distance = 1.41)
- d_5 (Red, distance = 1.41)

Step 4: Majority Voting

Now, we check the class labels of the 3 nearest neighbors. Out of the three:

- Two points (d_2 and d_5) are **Red**.
- One point (d_4) is **Blue**.

Since the majority of the neighbors are classified as **Red**, we classify the new point d_{new} as **Red**.

Summary of Steps:

1. **Compute distances** from the new point to all existing data points using the chosen distance metric (Euclidean distance in this case).
2. **Sort the distances** in ascending order.
3. Choose **k** (number of neighbors), and **select the k nearest neighbors**.
4. **Perform majority voting** among the k-nearest neighbors to assign the class label to the new point.

Final Classification:

In this example, $d_{\text{new}} = (5, 5)$ is classified as **Red** because the majority of the nearest neighbors (for $k = 3$) belong to the Red class.

2. Rocchio classifier

The Rocchio classifier works by calculating centroids (average points) for each class and then determining which class a new point is closer to, based on distance (usually Cosine Similarity or Euclidean Distance).

Example Scenario (See the example above)

Steps of Centroid-Based Classification

Step 1: Compute the Centroid for Each Class

The centroid for a class is the **mean of all points** in that class. Let's compute the centroid for both the Red and Blue classes.

- **Centroid for Red class:**

The Red class consists of the points $d_1 = (2, 4)$, $d_2 = (4, 6)$, and $d_5 = (6, 6)$. The formula for the centroid of a class C is:

$$c_{\text{red}} = \frac{1}{|C_{\text{red}}|} \sum d_i \quad \text{for all } d_i \in C_{\text{red}}$$

So, for the Red class:

$$c_{\text{red}} = \left(\frac{2+4+6}{3}, \frac{4+6+6}{3}\right) = (4, 5.33)$$

- **Centroid for Blue class:**

The Blue class consists of the points $d_3 = (4, 2)$ and $d_4 = (6, 4)$. Similarly, the centroid is:

$$c_{\text{blue}} = \frac{1}{|C_{\text{blue}}|} \sum d_i \quad \text{for all } d_i \in C_{\text{blue}}$$

So, for the Blue class:

$$c_{\text{blue}} = \left(\frac{4+6}{2}, \frac{2+4}{2}\right) = (5, 3)$$

Now we have the centroids for both classes:

- Red class centroid: $c_{\text{red}} = (4, 5.33)$
- Blue class centroid: $c_{\text{blue}} = (5, 3)$

Step 2: **Calculate the Distance Between d_{new} and the Centroids**

Next, we calculate the **Euclidean distance** between $d_{\text{new}} = (5, 5)$ and each centroid.

- **Distance between d_{new} and $c_{\text{red}}(4, 5.33)$:**

$$\text{Distance}(d_{\text{new}}, c_{\text{red}}) = \sqrt{(5-4)^2 + (5-5.33)^2} = \sqrt{1+0.1089} = \sqrt{1.1089} \approx 1.05$$

- **Distance between d_{new} and $c_{\text{blue}}(5, 3)$:**

$$\text{Distance}(d_{\text{new}}, c_{\text{blue}}) = \sqrt{(5-5)^2 + (5-3)^2} = \sqrt{0+4} = 2$$

Step 3: **Assign the Class Based on the Closest Centroid**

- The distance between d_{new} and c_{red} is approximately **1.05**.
- The distance between d_{new} and c_{blue} is **2**.

Since d_{new} is closer to the **Red** centroid (c_{red}), the new point d_{new} is classified as **Red**.

Summary of Steps:

1. **Compute the centroids** of each class by averaging the points in each class.
2. **Calculate the distance** between the new point and each class centroid (in this case, we used Euclidean distance).
3. **Classify the new point** based on the closest centroid.

Final Classification:

In this example, $d_{\text{new}} = (5, 5)$ is classified as **Red** because it is closer to the Red class centroid than the Blue class centroid.