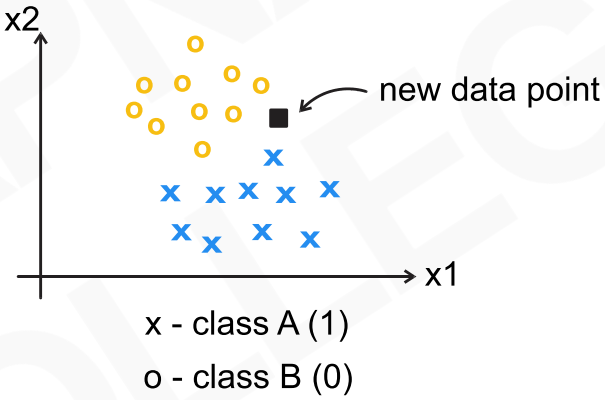
KNN - K Nearest Neighbors

Intuition & Logic

KNN (K-Nearest Neighbors) is a supervised ML algorithm used for classification and regression problems.

KNN makes predictions by looking at the **K closest data points** to a new data point and using their information. Basically on the logic - *"Tell me who your neighbors are, and I'll tell you who you are."*

Let's suppose we create a scatter plot of data split into 2 categories:



How do we predict which class the new data point belongs to? We use KNN.

How does KNN work?

1. We choose an odd number as K (like 3, 5, 7, 9 etc.)
2. Measure the **distance** between the new point and all existing points.

   There are two popular ways to do that:

   a. Euclidean Distance (most common)

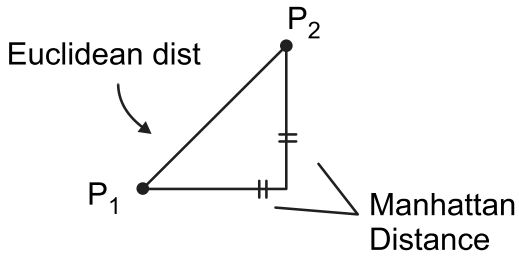
   This is the shortest distance between two points $P1(x1, y1)$ & $P2(x2, y2)$.

   $$d = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}$$

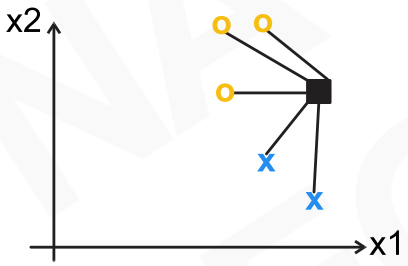
   b. Manhattan Distance

   This is the distance we'd travel in a grid-like city, moving only horizontally and vertically.

   $$d = |x2 - x1| + |y2 - y1|$$



3. Find the **K nearest neighbors** i.e. neighbors with the least distance.
4. Make a prediction.
   - For Classification problems we take a majority vote, so whichever class majority of the KNNs belong to is our the new point's class. (That's why K is odd)



   - For Regression problems we take the average of K neighbors' values to predict the new point's value.

KNN is also called as a **lazy learner algorithm** because it does not learn from the training set immediately instead it stores the entire dataset and performs computations only at the time of classification.

Note - **Feature scaling** is required to get the better performance of the KNN algorithm.

Let's suppose we have a dataset having m number of instances and n number of features.

There is one feature having values ranging between **0 and 1**.

Meanwhile, there is also a feature that varies from **-999 to 999**.

When these values are substituted in the formula of Euclidean distance, this will affect the performance by giving higher weightage to variables having a higher magnitude.

Keep Learning & Keep Exploring!