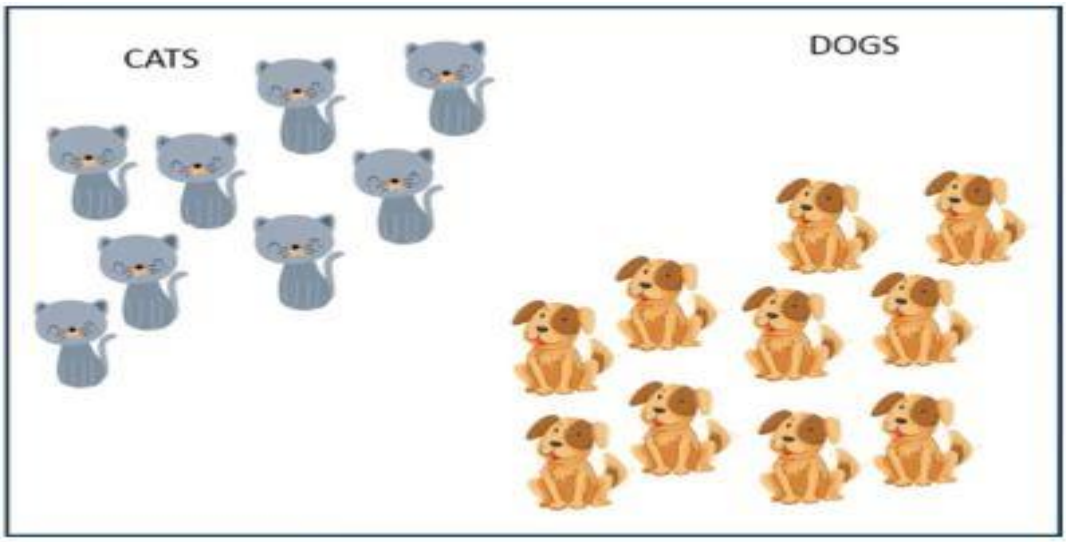
1b)Explain the KNN algorithm.

KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.

The KNN algorithm is useful when you are performing a pattern recognition task for classifying objects based on different features.

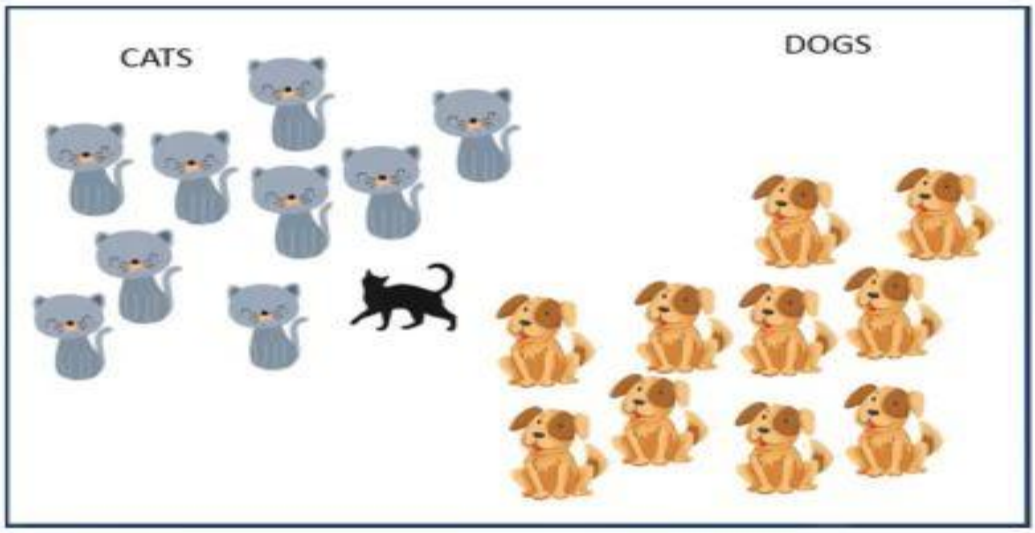
Suppose there is a dataset that contains information regarding cats and dogs. There is a new data point and you need to check if that sample data point is a cat or dog. To do this, you need to list the different features of cats and dogs.

Now, let us consider two features: claw sharpness and ear length. Plot these features on a 2D plane and check where the data points fit in.

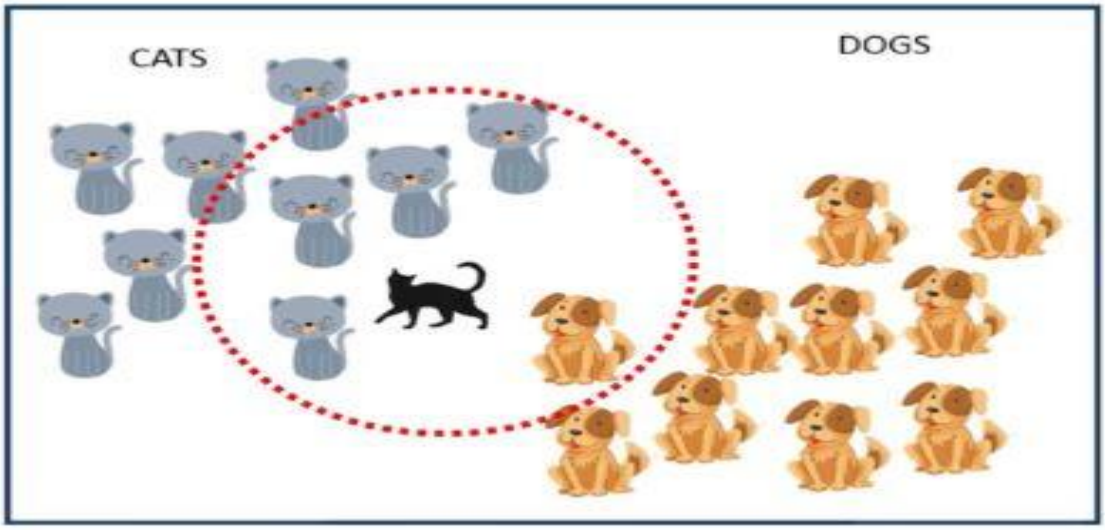


The sharpness of claws is significant for cats, but not so much for dogs. On the other hand, the length of ears is significant for dogs, but not quite when it comes to cats.

Now, if we have a new data point based on the above features, we can easily determine if it's a cat or a dog.



The new data point features indicate that the animal is, in fact, a cat.



Since KNN is based on feature similarity, we can perform classification tasks using the KNN classifier. The image below—trained with the KNN algorithm—shows the predicted outcome, a black cat.

When Do We Use the KNN Algorithm?

The KNN algorithm is used in the following scenarios:

1. Data is labeled
2. Data is noise-free
3. Dataset is small, as KNN is a lazy learner

Pros and Cons of Using KNN

pros:

1. Since the KNN algorithm requires no training before making predictions, new data can be added seamlessly, which will not impact the accuracy of the algorithm.
2. KNN is very easy to implement. There are only two parameters required to implement KNN—the value of K and the distance function (e.g. Euclidean, Manhattan, etc.)

Cons:

1. The KNN algorithm does not work well with large datasets. The cost of calculating the distance between the new point and each existing point is huge, which degrades performance.
2. Feature scaling (standardization and normalization) is required before applying the KNN algorithm to any dataset. Otherwise, KNN may generate wrong predictions.