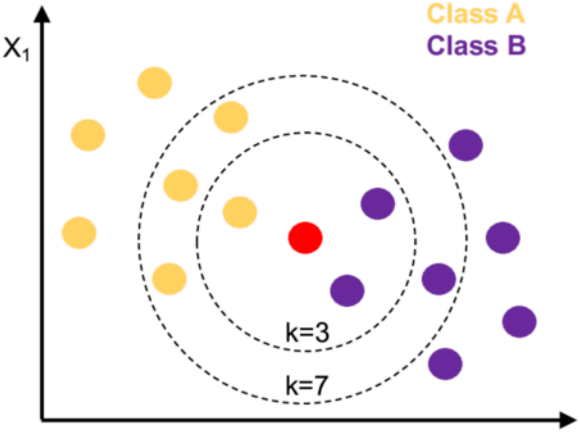
K-Nearest Neighbors (KNN)



For k = 3, what class is it? Class B

For k = 7, what class is it? Class A

In K-Nearest Neighbors (KNN), the primary factor that determines the class of a data point is the majority class of the nearest K neighbors.

The "K" represent the number of nearest neighbors to consider when making a prediction

KNN algorithm measure the similarity between data points by calculating the distance between data points

A major disadvantage of the KNN algorithm is that it requires a large amount of memory and computation as the dataset grow

KNN algorithm is very fast during the training phase but slow during prediction

K-Nearest Neighbor

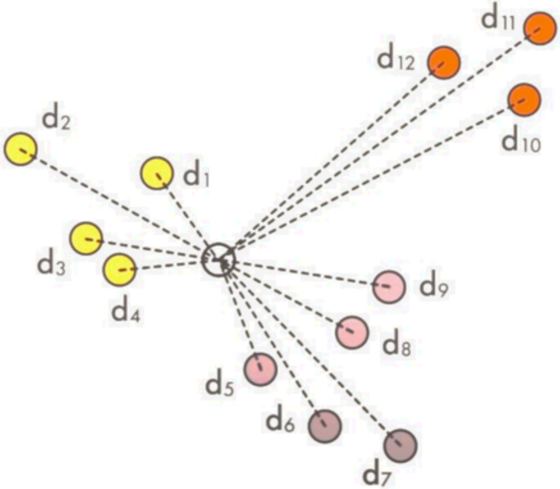
Distance Calculation

Euclidean:

$$d(x,y) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2}$$

Manhattan/city - block:

$$d(x,y) = \sum_{i=1}^{m} |x_i - y_i|$$



KNN algorithm relies on distance for classification, if the features represent different physical units or points are scaled differently, features with larger scales may dominate in the distance calculations.
Normalizing the training data can improve its accuracy dramatically

K-Nearest Neighbor

Effect of k Values

The best way to choose the optimal value of "K" for a KNN classifier is by using **cross-validation** to evaluate the performance of the model for different values of K.

K-Nearest Neighbors (KNN)

You have the following data points: (2, 3), (5, 8), (4, 5), and (6, 7). If a new point (4, 6) is given, what is the Euclidean distance to the nearest neighbor? 1.41

K-Nearest Neighbors (KNN)

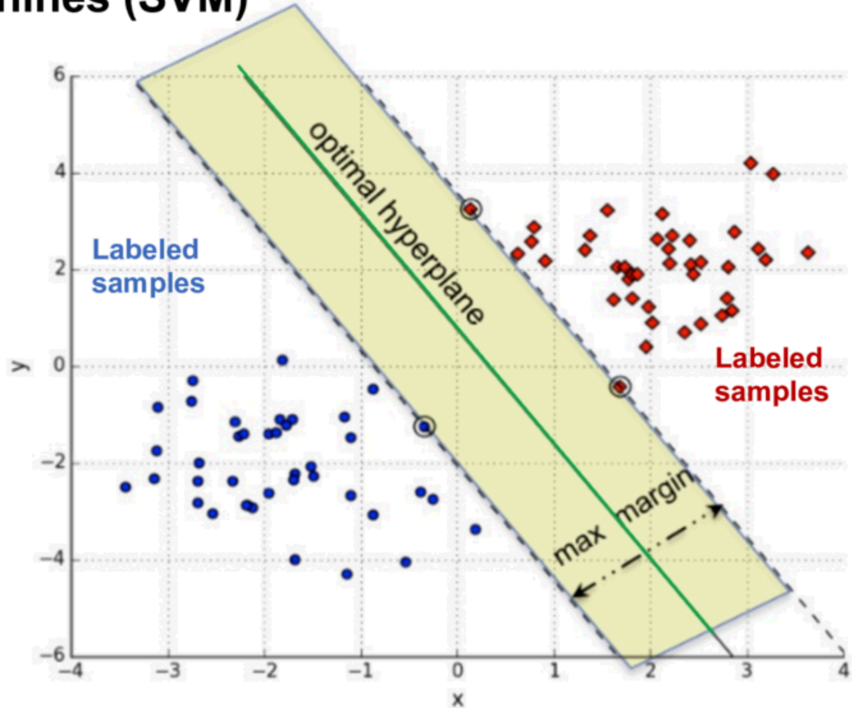
Given a dataset with points classified as Class X and Class Y, where Class X has points (1, 1), (2, 2), (3,1) and Class Y has points (4, 4), (5, 4). If a new point (3, 3) is to be classified using K=3, which class will it belong to based on the nearest neighbors?

Support Vector Machines (SVM)

SVM is a supervised learning method that looks at data and sorts it into one of the two categories.

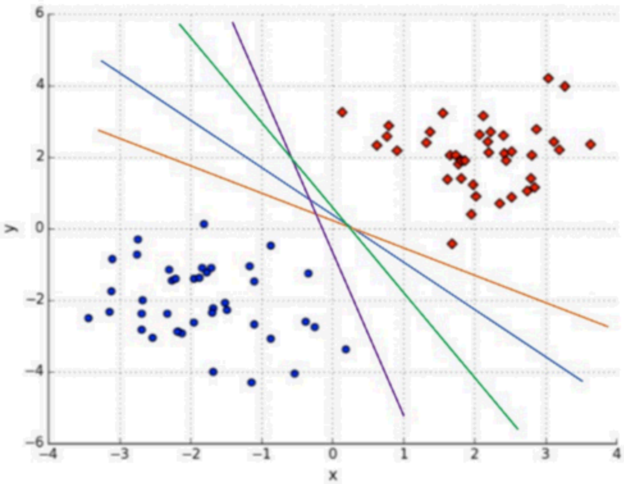
The primary goal of SVM in classification is to find the best-fit line that maximizes the margin between two classes.

A common advantage of using SVM is that it works well with high-dimensional data and can handle large datasets.

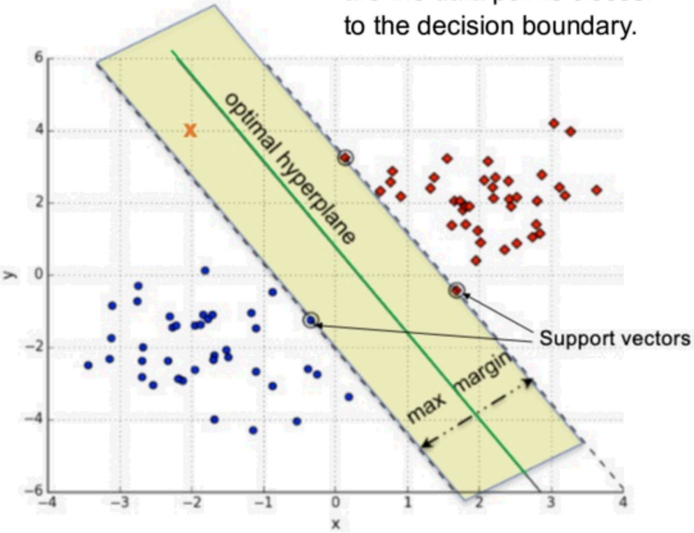


Support Vector Machines (SVM)

"Support Vectors" in SVM are the data points closest to the decision boundary.



In logistic regression, any of these lines might work well for this set of data.

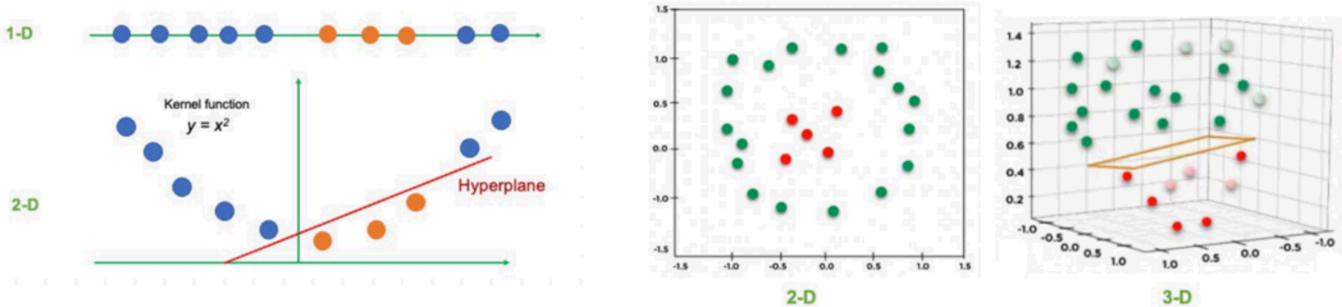


SVM will converge to a line that has the maximum margin to separate the two classes

Support Vector Machines (SVM)

In the case of non-linearly separable data, SVM handle the classification problem by using kernel functions to map the data into a higher-dimensional space, where a linear decision boundary can be found.

Transformation



Naive Baves

Bayes' Theorem

$$P(A|x) = \frac{P(x|A)P(A)}{P(x)}$$

Where:
$P(A|x)$ = Probability of class A given the feature vector x (Posterior Probability)
$P(x|A)$ = Probability of observing x given class A (Likelihood)
$P(A)$ = Probability of class A before observing x (Prior Probability)
$P(x)$ = Probability of observing x for all classes (Evidence)

Naïve Bayes Assumption

Naive Bayes assumes that features in x = {x1,x2,…,xn} are **conditionally independent** given the class label A. This simplifies the likelihood computation as:

$$P(x|A) = P(x_1|A) \cdot P(x_2|A) \cdot \cdots \cdot P(x_n|A)$$

To classify a new data point x, the algorithm computes the posterior probability for each class and selects the class with the highest probability

Naive Bayes

In a Naive Bayes classifier, if the prior probabilities are $P(A) = 0.3$ and $P(B) = 0.7$, and the likelihoods for feature F are $P(F|A) = 0.5$ and $P(F|B) = 0.6$, what is the posterior probability of Class A given feature F?