**Broadly, there are 3 types of Machine Learning Algorithms:**

**1. Supervised Learning:**

This algorithm consists of a target / outcome variable (or dependent variable) which is to be predicted from a given set of predictors (independent variables). Using these set of variables, we generate a function that map inputs to desired outputs. The training process continues until the model achieves a desired level of accuracy on the training data.

Supervised learning is commonly used in real world applications, such as face and speech recognition, products or movie recommendations, and sales forecasting. Supervised learning can be further classified into two types: Regression and Classification.

* Regression trains on and predicts a continuous-valued response, for example predicting real estate prices.
* Classification attempts to find the appropriate class label, such as analyzing positive/negative sentiment, male and female persons, benign and malignant tumors, secure and unsecure loans etc.

**Examples of Supervised Learning**: SVM, Regression, Decision Tree, Random Forest, KNN, ANN, Logistic Regression etc.

**ALGORITHMS:**

**1.1) Support Vector Machine(SVM):**

It is a supervised [machine learning algorithm](https://courses.analyticsvidhya.com/courses/introduction-to-data-science-2?utm_source=blog&utm_medium=understandingsupportvectormachinearticle) which can be used for both classification or regression challenges. However, it is mostly used in classification problems. In the SVM algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well (look at the below snapshot).

Support Vectors are simply the co-ordinates of individual observation. The SVM classifier is a frontier which best segregates the two classes (hyper-plane/ line).

**Identify the right hyper-plane (Scenario-1):** Here, we have three hyper-planes (A, B and C). Now, identify the right hyper-plane to classify star and circle.

You need to remember a thumb rule to identify the right hyper-plane: “Select the hyper-plane which segregates the two classes better”. In this s cenario, hyper-plane “B” has excellently performed this job.

**Identify the right hyper-plane (Scenario-2):** Here, we have three hyper-planes (A, B and C) and all are segregating the classes well. Now, how can we identify the right hyper-plane?

Here, maximizing the distances between nearest data point (either class) and hyper-plane will help us to decide the right hyper-plane. This distance is called as **Margin**.

Above, you can see that the margin for hyper-plane C is high as compared to both A and B. Hence, we name the right hyper-plane as C. Another lightning reason for selecting the hyperplane with higher margin is robustness. If we select a hyper-plane having low margin then there is high chance of miss-classification.

**Identify the right hyper-plane (Scenario-3):**Hint:Use the rules as discussed in previous section to identify the right hyper-plane

Some of you may have selected the hyper-plane **B** as it has higher margin compared to **A.** But, here is the catch, SVM selects the hyper-plane which classifies the classes accurately prior to maximizing margin. Here, hyper-plane B has a classification error and A has classified all correctly. Therefore, the right hyper-plane is **A.**

**Can we classify two classes (Scenario-4)?:** Below, I am unable to segregate the two classes using a straight line, as one of the stars lies in the territory of other(circle) class as an outlier.

As I have already mentioned, one star at other end is like an outlier for star class. The SVM algorithm has a feature to ignore outliers and find the hyper-plane that has the maximum margin. Hence, we can say, SVM classification is robust to outliers.

A support vector machine is a directed learning algorithm that sorts data into two classes. It is

prepared with a trained of data previously arranged into two classifications, fabricating the

model as it is at first prepared. The assignment of the SVM algorithm is to figure out which

class a new data point has a place in.

In this paper let’s discuss how the SVM algorithm is used to approach classification problems.

The main aim of the classification problem is to separate data into two classes. If you see

graph (A) the data points were easily classified using Hyperplane where the points above the

hyperplane belong to one class and the points below the hyperplane belong to another class. A

hyperplane is a boundary that classifies the data point, it can be a simple straight line in 2

dimensions.

In SVM along with the hyperplane, it creates two marginal planes which are parallel to

hyperplanes. These planes are known as marginal hyperplanes (they are represented as dotted

lines in the graph (A)). While creating the marginal hyperplane algorithm makes sure it passes

through one of the nearest positive or negative point. The distance between the marginal

hyperplanes is called Marginal distance. The significance of this marginal distance is it plays a

major role in choosing the best-suited hyperplane as hyperplane can be drawn in multiple

ways for classifying data. The hyperplane with maximum marginal distance is the most

efficient.

Chart, diagram, scatter chart

Description automatically generated

Observe the picture (B) both graphs in the picture represent the same data point but the

hyperplane is drawn in different ways. We can see that the marginal distance (D\_1) is greater

than (D\_2). So here in this example, the most efficient hyperplane is the one with marginal

distance (D\_1).

Diagram

Description automatically generated

As shown in picture C.1, there are data points represented on a straight line where positive

points are sandwiched between negative. Here it is impossible to classify negative and positive

points in two different classes using a hyperplane. By using the polynomial kernel, we convert

1D to 2D which means we map points from the straight line to a polynomial (as shown in

C.2). As shown now after implementing polynomial kernel we can easily classify the negative

and positive points using a straight line (hyperplane).

Diagram

Description automatically generated

**Reference Link**

1. Stefan Kahl, Thomas Wilhelm-Stein, Hussein Hussein, Holger Klinck, Danny Kowerko,

Marc Ritter, and Maximilian Eibl Large-Scale Bird Sound Classification using

Convolutional Neural Networks

1. Bo Zhao, Jiashi Feng Xiao Wu Shuicheng Yan A Survey on Deep Learning-based Fine-

grained Object Classication and Semantic Segmentation