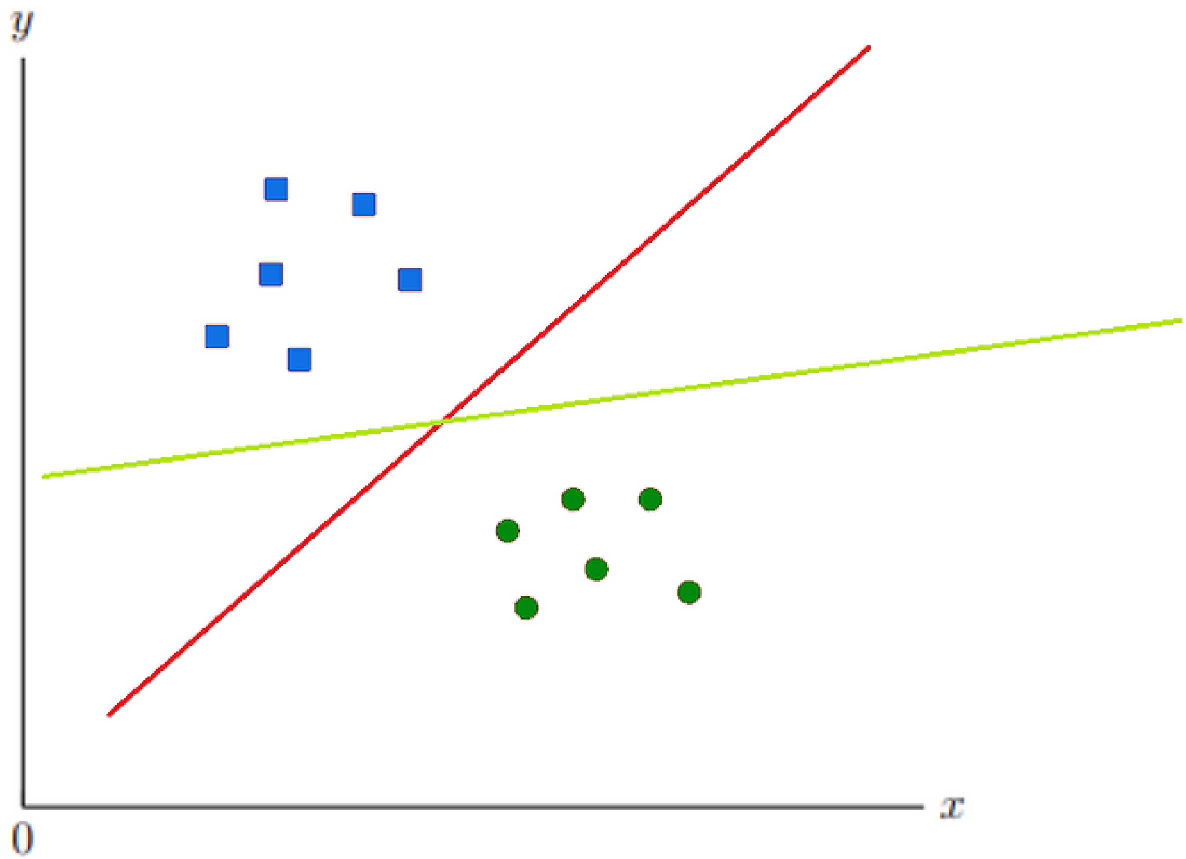
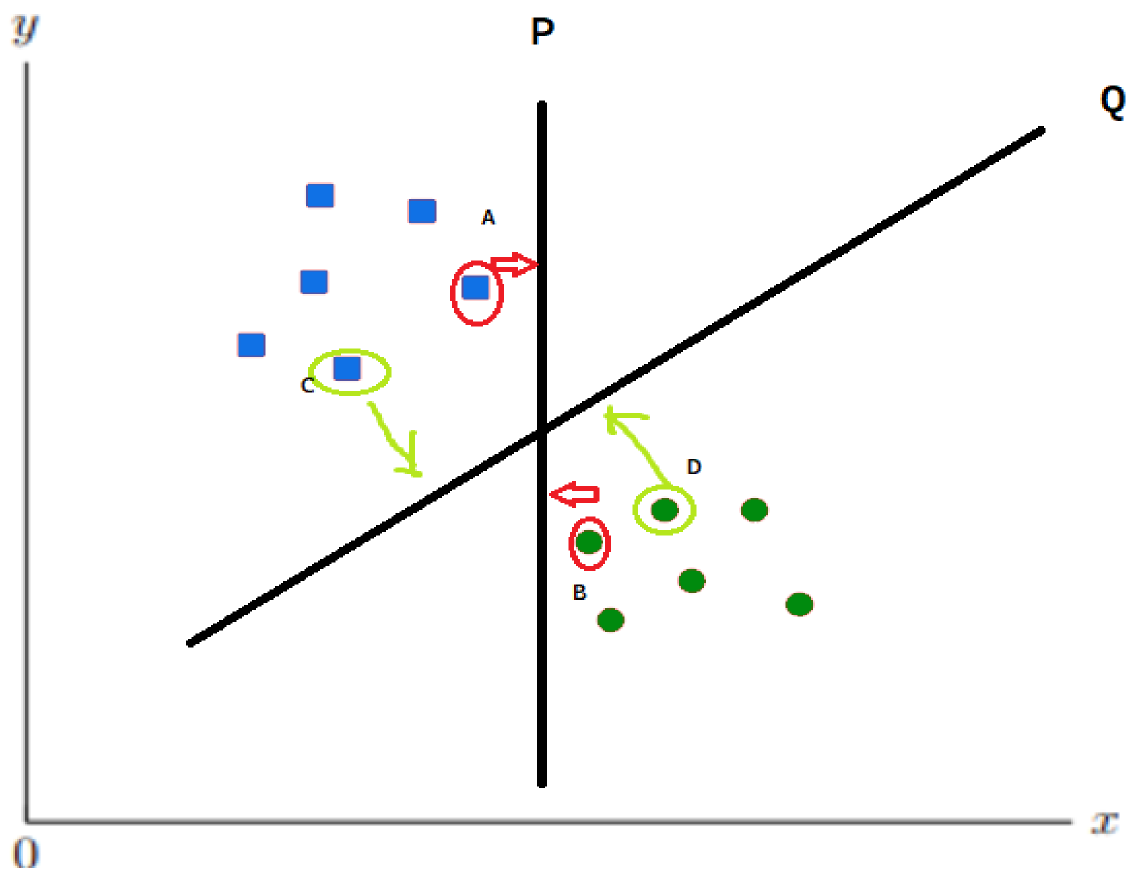
Support Vector Machine(SVM)

The objective of the SVM algorithm is to **create a decision boundary or line** that can effectively segregate a given data set into different classes. Once the decision boundary is established, new examples can be classified into the appropriate classes with relative ease. In the SVM algorithm, this decision boundary is known as a **hyperplane**. The challenge then becomes drawing this hyperplane with precision.



Suppose we are given a data set with two classes drawn in blue and green colors, respectively. The data set is linearly separable, meaning we can draw a straight line to separate the two classes. However, we need to determine which line is the best fit or the optimal boundary. In the given diagram, two lines are drawn in red and green colors, respectively. In this case, we need to draw all possible lines to determine the one with the best accuracy. This line is called the Hyperplane, whose dimension depends on the features present in the data set. If there are only two features, we can draw a straight line, but if there are more than two features, we need to draw a plane, which will be the two-dimensional hyperplane in this case.

- Support vectors are data points that are closer to the hyperplane and influence the position and orientation of the hyperplane. Using these support vectors, we maximize the margin of the classifier. Deleting the support vectors will change the position of the hyperplane. These are the points that help us build our SVM.



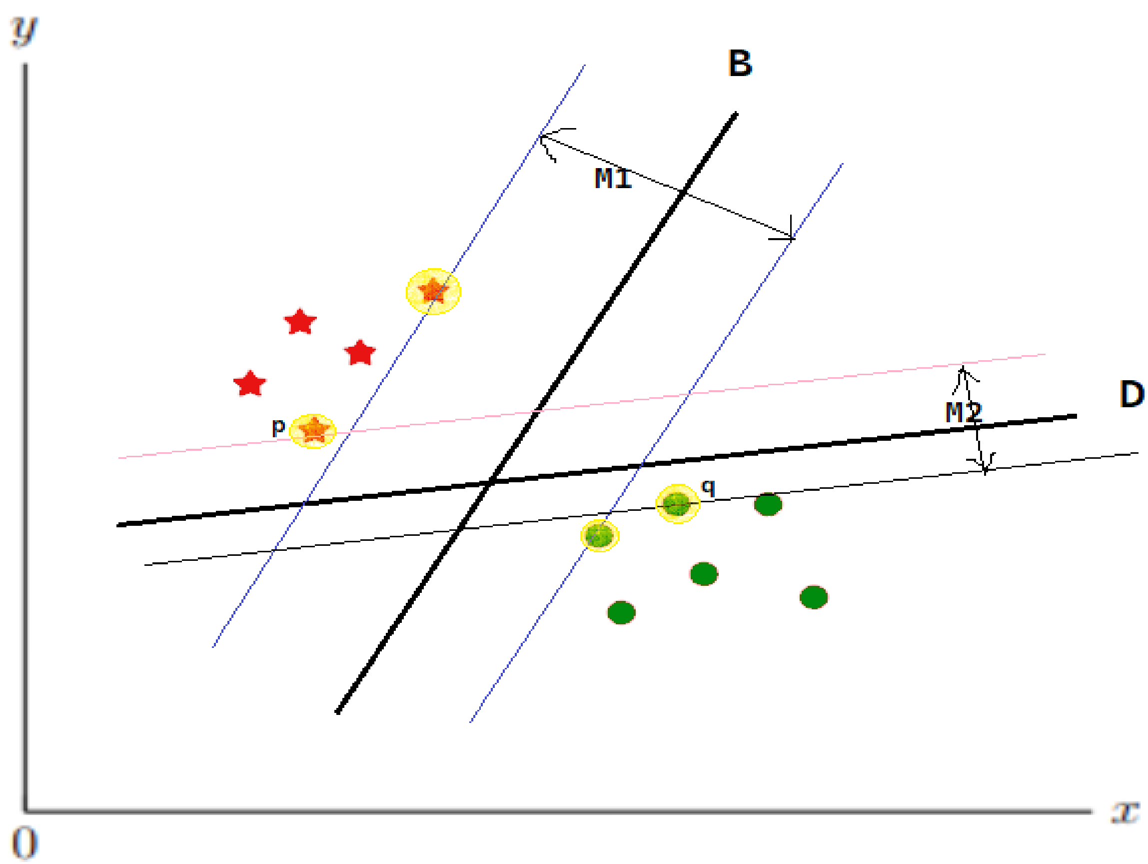
In this graph, A and B points are nearest for this P hyperplane. So A and B are referred to as support vectors to the P hyperplane in this case.

And C and B are the other nearest points to the Q hyperplane.

- **<u>different types of SVM</u>**

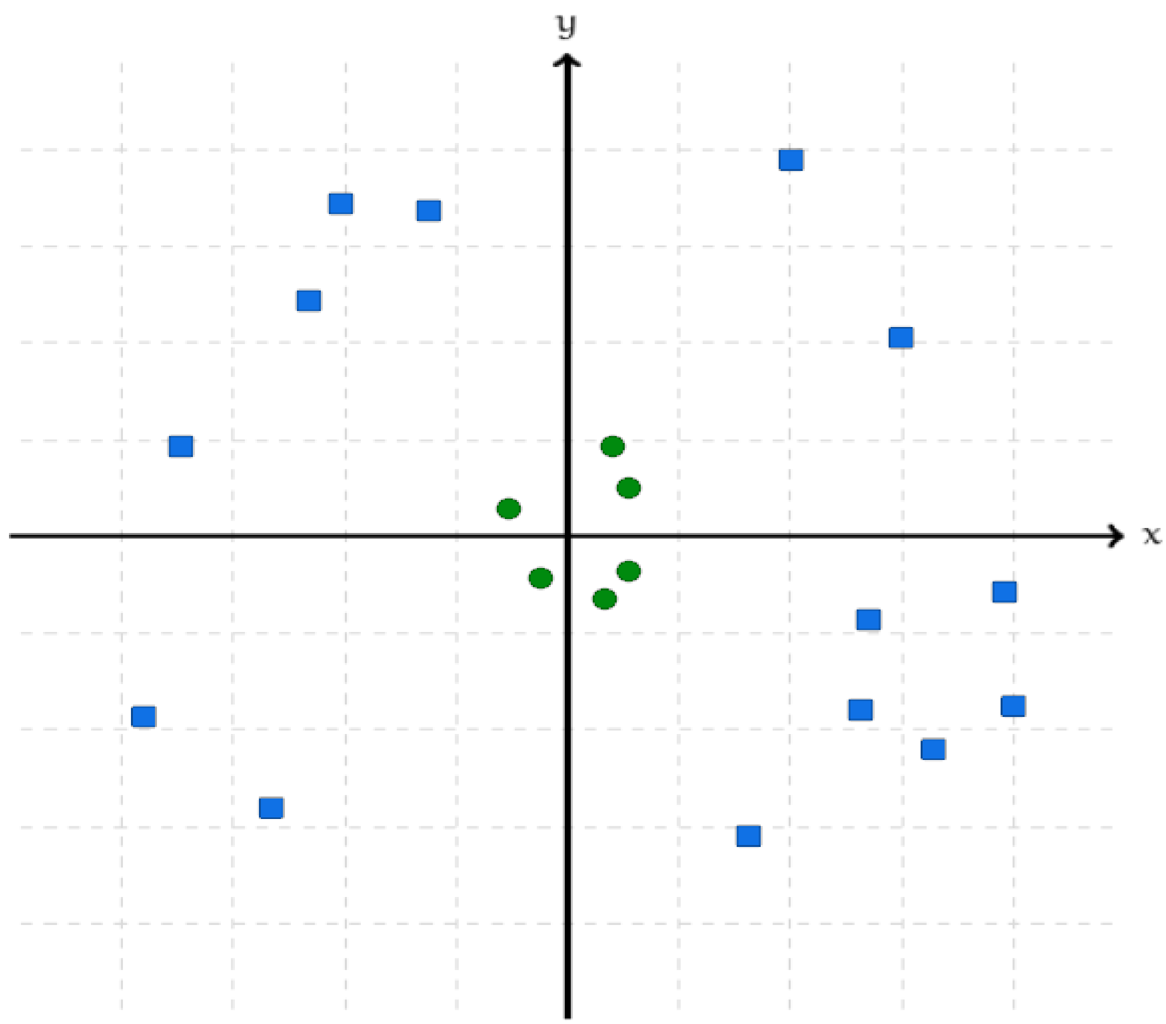
1- Linear SVM — Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and classifier is used called as Linear SVM classifier.

Suppose we have a dataset that has two classes(stars and dots) and the dataset has two features x and y. we can separate this dataset using B and D straight lines. When we draw multiple straight lines to separate the data which one has to be considered as the hyperplane. To understand we can use this example.



Assume D as the one hyperplane and these p and q are the support vectors. So draw a parallel line to a hyperplane with the support of support vectors. And the M2 distance we need to calculate. This distance we called Margin. Similarly, we need to do this to B hyperplane also. And calculate the margin M1. When comparing the M1 and M2, the M2 margin is smaller when compared to the M1. So using this we can say B is the best fit line(hyperplane) which divides the data into two classes. We should do this calculation for all possible hyperplanes and the one who gives the maximum margin will be the hyperplane.

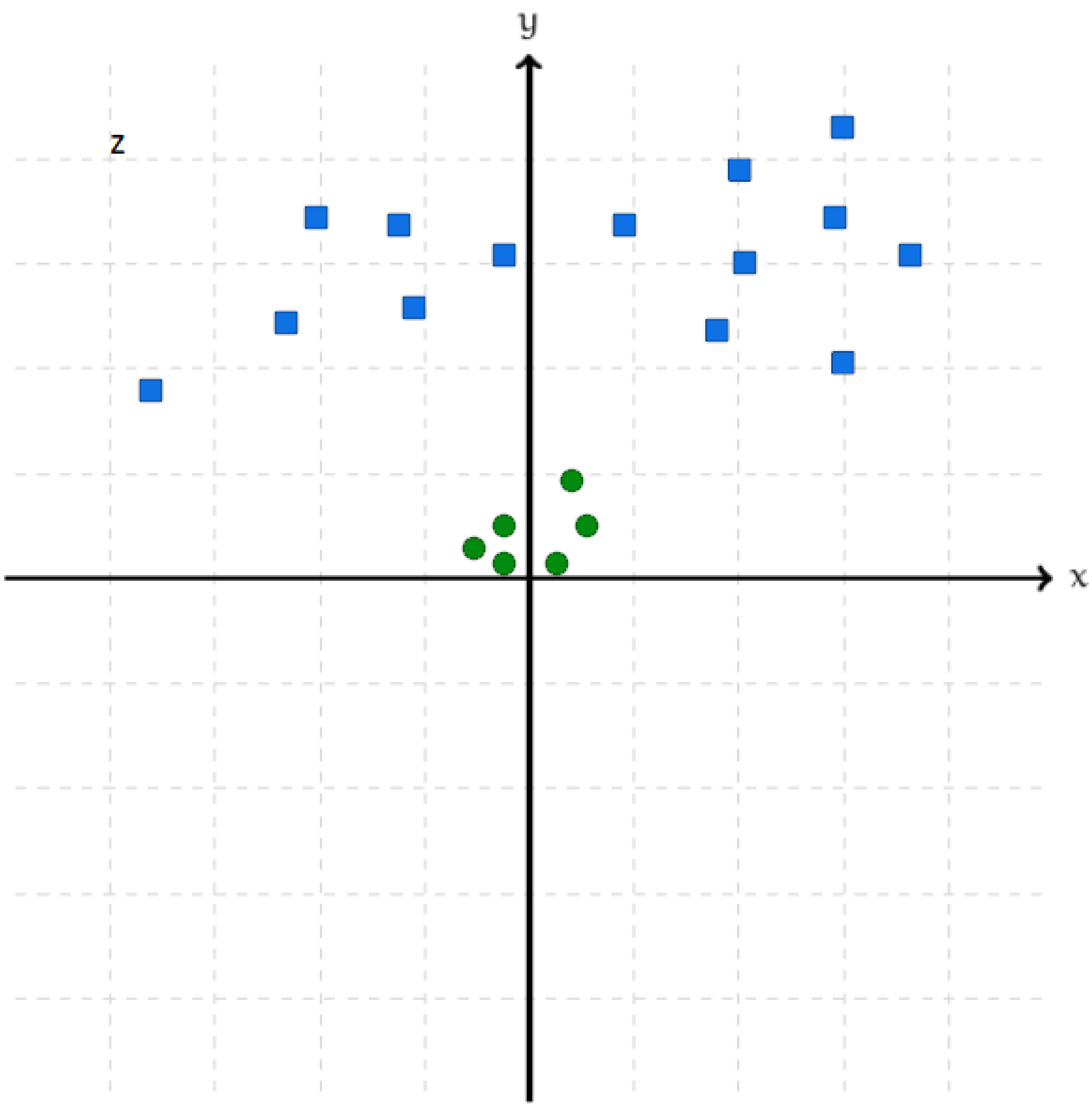
2- Non-linear SVM — Non-linear SVM is used for non-linearly separated data, which means if a dataset cannot be classified by using a straight line, then such data is termed as Non-linear data, and the classifier is called a Non-linear SVM classifier.



Here we are not able to draw a straight line to device these data into 2 classes. We need to convert particular data into linear data. For that, we can use a mapping function.

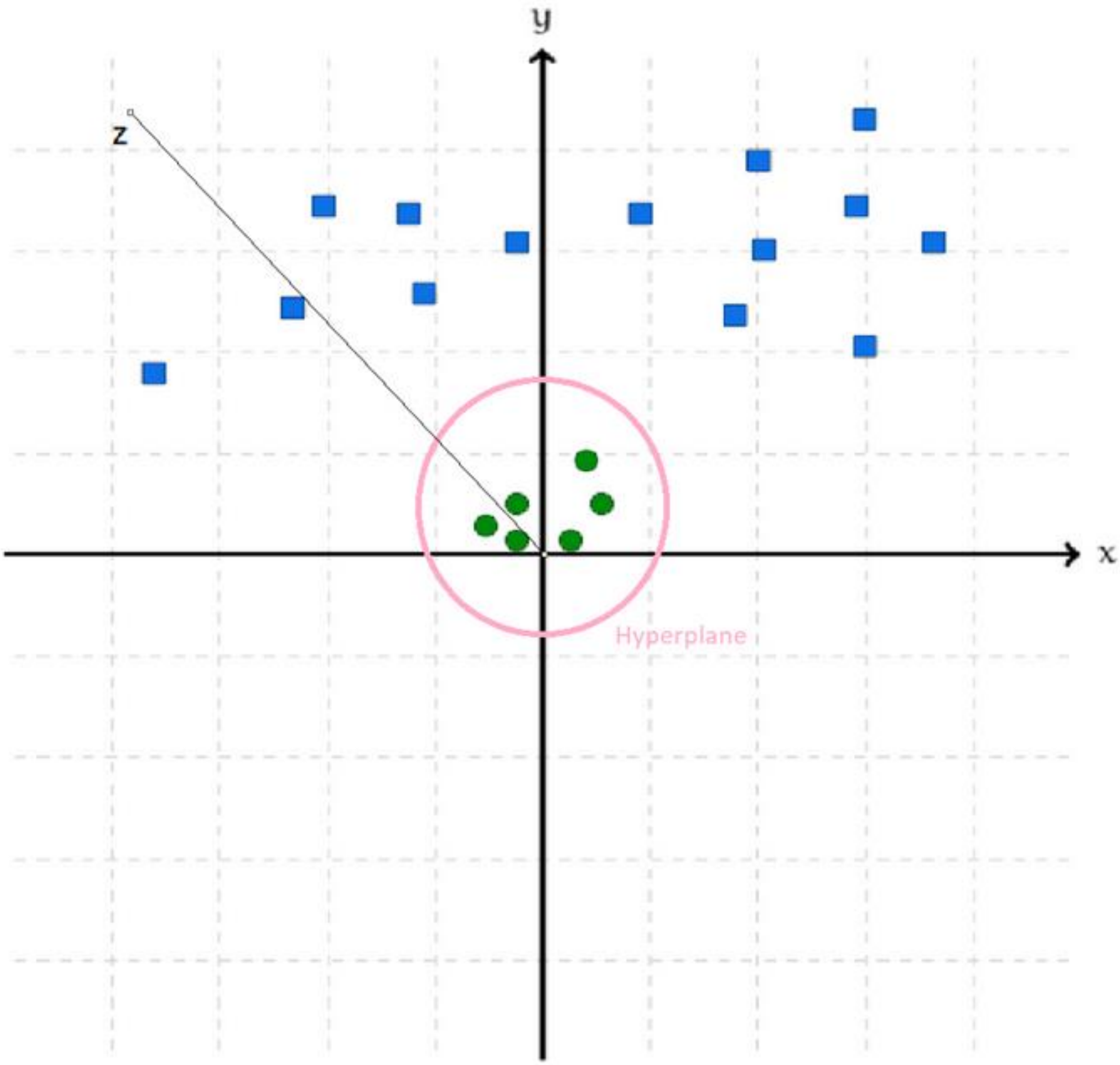
```
It can be calculated as:

z = x^2 + y^2
```

Once we apply this all the data points will be like this. And we will get another axis called z.



The above depiction of data points is linearly separable and can be separated by a straight-line hyperplane. Now we can draw all possible lines that will separate data into two classes and we need to identify one straight line that will give the maximum margin and that will be considered as the best hyperplane.

This representation is in 3-D with a z-axis. In 2-D, the graph looks like this:-



This is what a non-linear SVM does! It takes the data points to a higher dimension to be linearly separable in that dimension, and then the algorithm classifies them.

- **Math behind the Support Vector Machine Classifier**

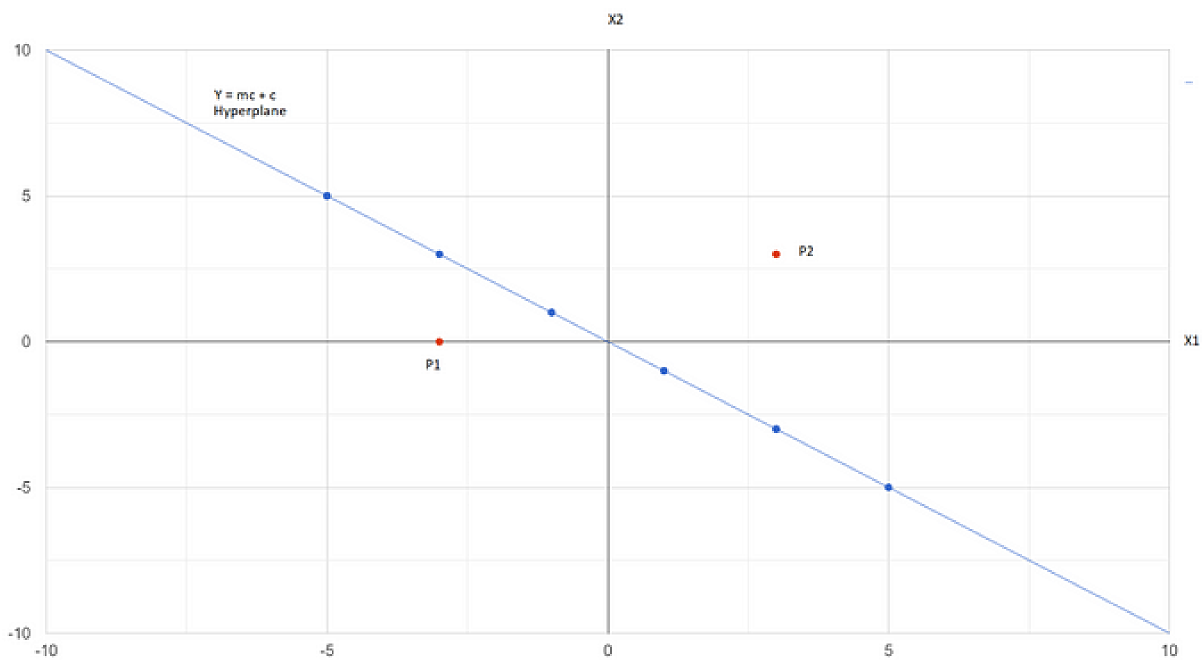
Let's say we have two features X1 and X2. We are trying to plot the data points in these two features in a graph. Let's say we have two points P1 and P2.

```
P1 - (-3, 0)   P2 - (3, 3)
```

To draw a hyperplane that can separate these two points is a straight line because this is a 2-dimensional space. Let's take this line(blue color line) which will act as our hyperplane for our supporting machine classifier. We know that the equation of a straight line is,

```
y = mx +c
```

```
In the below scatter plot diagram,

Slope  =  -1 (becuase of the negative slope)
Intercept  = 0 (becuase this line go through the origin of the graph)
```



take these two parameters slop and Intercept and store them in a variable called W. W is the weight vector that contains the coefficients of x and y.

```
For example, for a line given in the diagram
-x - y = 0,
W -> parameter of the line
     (-1, -1)

We can form the equation like this ,
-x - y + b = 0,

W = [-1, -1], X = [x, y] and bias b = 0

We can represent the same equation in the following form:
f(x) = W.Transpose . X
```

Let’s take the P1(-3. 0) point and we try to find the corresponding W.Transpose value for that.

I do a simple matrix multiplication. For that, I need Wt(transpose of w) and multiply it with x to find the Wt. X value.

Why do we use Wt value?

We are trying to project the P1 point into the Hyperplane. Fo that we need to use matrix multiplication(Projection of vectors).

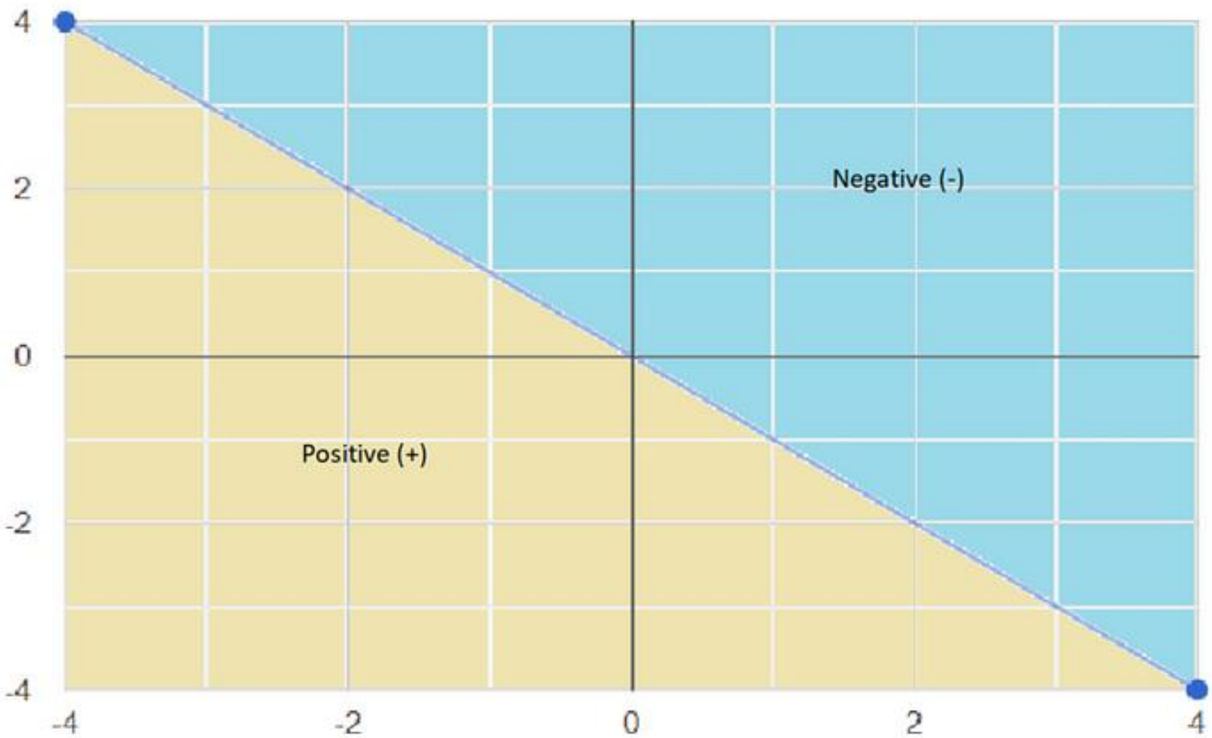
```
W = [-1, -1]
Wt = [-1 ]
     [-1 ]

X = [x, y] = [-3  0]

Wt . X =  [-1 ]
          [-1 ] . [-3  0 ]

Wt . x  = +3 (Positive value)
```

Now you can see the value of multiplication is 3. It is a positive number. So we can say that all the points that lie on the left side of the hyperplane will have positive values.

Let's take the P2(3. 3) point and we try to find the corresponding y value for that.

```
Wt . x  = [-1]
          [-1 ] . [3  3]

Wt . x  = -6 (Negative value)
```

Now you can say, for all the points that lie on the left side of the hyperplane Wt. x value will be Positive and For all the points that lie on the right side of the hyperplane Wt. x value will be Negative.



This is how our support vector machine classifier will try to classify data points. Wt will act as a Label.