Support Vector Machine

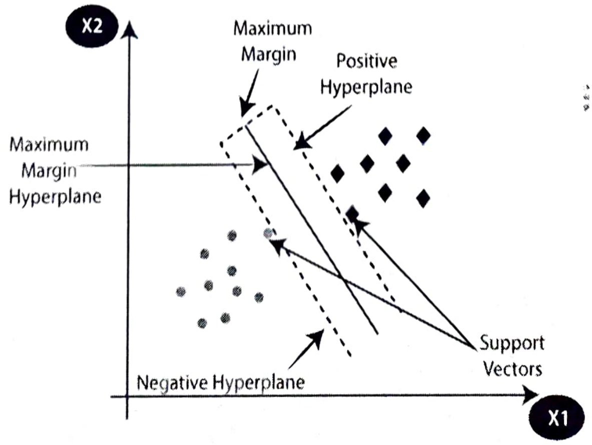
Support Vector Machine, abbreviated as SVM can be used for both regression and classification tasks, but generally, they work best in classification problems. They were very famous around the time they were created, during the 1990s, and keep on being the go-to method for a high-performing algorithm with a little tuning.

It is a supervised machine learning problem where we try to find a hyperplane that best separates the two classes. SVM does this by finding the maximum margin between the hyperplanes that means maximum distances between the two classes.

Linear SVM: When the data is perfectly linearly separable only then we can use Linear SVM. Perfectly linearly separable means that the data points can be classified into 2 classes by using a single straight line(if 2D).

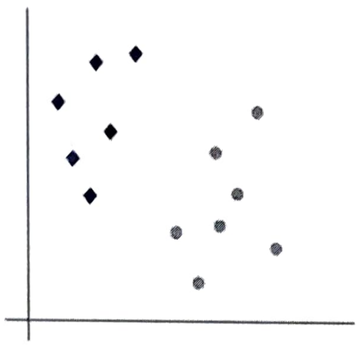
Important Terms

- Support Vectors: These are the points that are closest to the hyperplane. A separating line will be defined with the help of these data points.
- Margin: it is the distance between the hyperplane and the observations closest to the hyperplane (support vectors). In SVM large margin is considered a good margin. There are two types of margins hard margin and soft margin. I will talk more about these two in the later section.

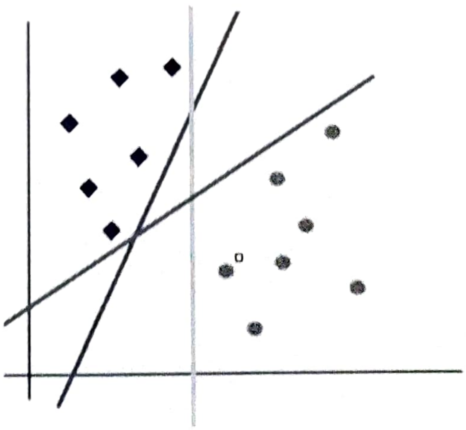


How Does Support Vector Machine Work?

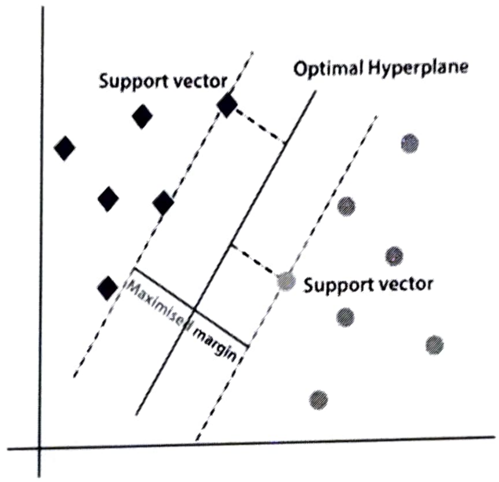
SVM is defined such that it is defined in terms of the support vectors only, we don't have to worry about other observations since the margin is made using the points which are closest to the hyperplane (support vectors), whereas in logistic regression the classifier is defined over all the points. Hence SVM enjoys some natural speed-ups. Suppose we have a dataset that has two classes (green and blue). We want to classify that the new data point as either blue or green.



To classify these points, we can have many decision boundaries. Since we are plotting the data points in a 2-dimensional graph we call this decision boundary a straight line but if we have more dimensions, we call this decision boundary a "hyperplane".

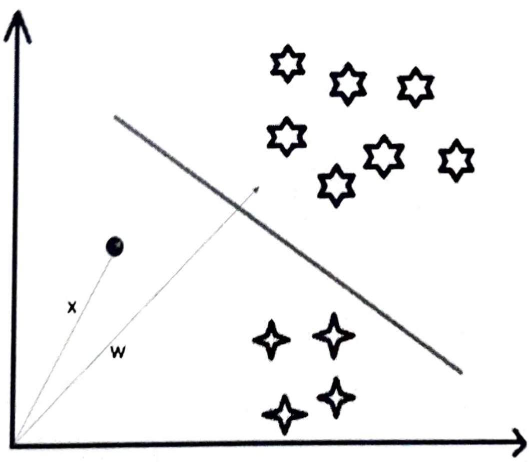


The best hyperplane is that plane that has the maximum distance from both the classes, and this is the main aim of SVM. This is done by finding different hyperplanes which classify the labels in the best way then it will choose the one which is farthest from the data points or the one which has a maximum margin.

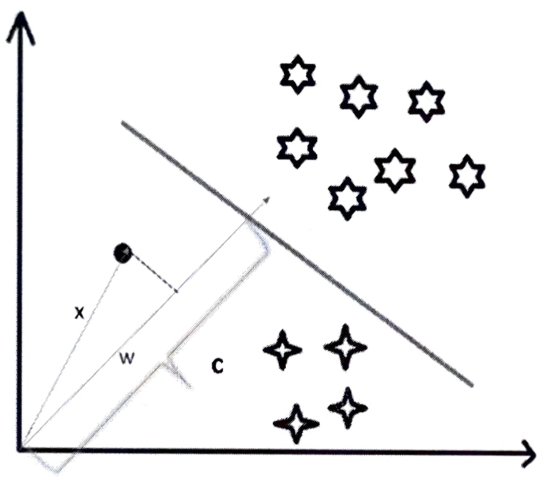


Use of Dot Product in SVM

Consider a random point X and we want to know whether it lies on the right side of the plane or the left side of the plane (positive or negative).



To find this first we assume this point is a vector (X) and then we make a vector (w) which is perpendicular to the hyperplane. Let's say the distance of vector w from origin to decision boundary is 'c'. Now we take the projection of X vector on w.



Projection of any vector or another vector is called dot-product. Hence, we take the dot product of x and w vectors. If the dot product is greater than 'c' then we can say that the point lies on the right side. If the dot product is less than 'c' then the point is on the left side and if the dot product is equal to 'c' then the point lies on the decision boundary.

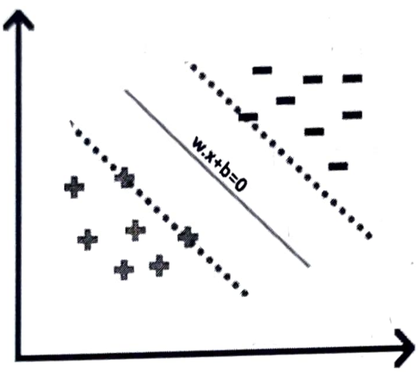
$$\vec{X}.\vec{w} = c \text{ (the point lies on the decision boundary)}$$

$$\vec{X}.\vec{w} > c \text{ (positive samples)}$$

$$\vec{X}.\vec{w} < c \text{ (negative samples)}$$

Margin in Support Vector Machine

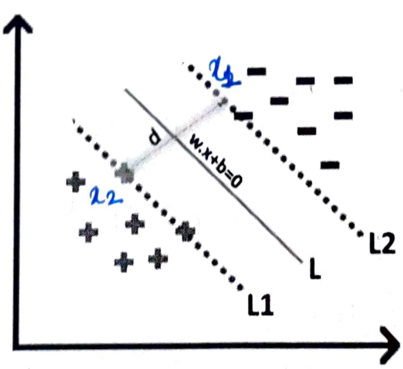
Equation of a hyperplane is w.x+b=0 where w is a vector normal to hyperplane and b is an offset.



$$y = \begin{cases} +1 & \text{if } \vec{X}.\vec{w} + b \geq 0 \\ -1 & \text{if } \vec{X}.\vec{w} + b < 0 \end{cases}$$

If the value of w.x+b>0 then we can say it is a positive point otherwise it is a negative point.

Now we need (w,b) such that the margin has a maximum distance. Let's say this distance is 'd'.



Hyperplane equation: $w^T x + b = 0$
For negatives: $w^T x + b = -1$
For positives: $w^T x + b = 1$

For x_1: $w^T x_1 + b = -1$
For x_2: $w^T x_2 + b = 1$

$$\begin{array}{r} w^T x_2 + b = 1 \\ -\quad - \quad - \\ \hline w^T (x_2 - x_1) = 2 \end{array}$$

Unit vector $= \frac{w}{\|w\|}$ (has a ~~mag~~ magnitude of 1)

So, from above equation, as we want distance betn x_1 and x_2 i.e. $(x_2 - x_1)$, remove w^T by unit vector.

$$\therefore \frac{w^T}{\|w\|}(x_2 - x_1) = \frac{2}{\|w\|}$$

$$\therefore x_2 - x_1 = \frac{2}{\|w\|}$$

$\therefore$ Update (w^*, b^*) such that $(w^*, b^*) \max \frac{2}{\|w\|}$

having condition,

$$y_i \begin{cases} +1 & w^T x + b \geq 1 \\ -1 & w^T x + b \leq -1 \end{cases}$$

$$\therefore \quad y_i * w^T x_i + b \geq 1$$

Above is the generalized equation.
If value is not coming as ≥ 1, then it is misclassification.

$$\boxed{\therefore (w^*, b^*) = \min \frac{\|w\|}{2} + C_i \sum_{i=1}^{n} \xi_i}$$

C_i = how many errors
ξ_i = value of the error.