Machine Learning

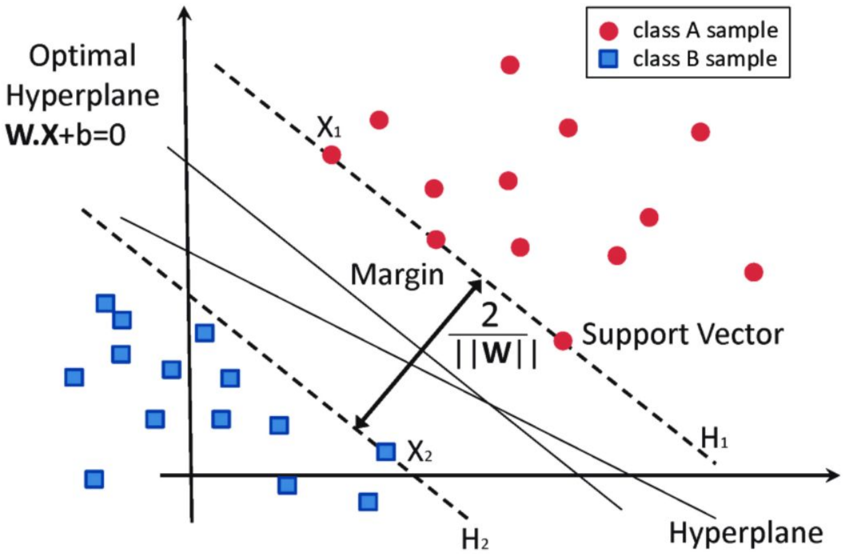
Support Vector Machine

Support Vector Machine

- Introduced in 1992 by Vapnik for classification of both linear and non-linear data.
- Applied to many areas including handwritten digit recognition, text classification, speaker identification ,object recognition as well as time series prediction.
- Although their training time is slow but they are highly accurate owing to their ability to complex non-linear decision boundaries (hyperplanes)

Support Vector Machine

- Let the dataset D be given as tuples in the form of {(X1,y1),(X2,y2),(X3,y3)...(Xn,yn)} where n is the number of data points in D
- Since there are two classes only, each yi ∈ {− 1, 1}
- Suppose each tuple Xi is a 2 dimensional vector representing the attributes x1 and x2.
- Scaling or normalization is performed to guard against the variables(attribute) with large variance.



Support Vector Machine

- Data is linearly separable as a straight line can be drawn to separate tuples of two classes -1 and +1.
- There can be infinite number of lines that can be drawn to separate the data.
- Aim is to find the best line that gives the minimum error rate on unknown tuples.
- If it was a 3D data, we would then find the best separating *plane.*
- *For n dimensions, we would* then find the best separating *hyperplane.*
- *"But how do we find the best line"? ,* Intuitively we can expect the hyperplane with the larger margin to be more accurate at classifying future data tuples than the hyperplane with the smaller margin.

Support Vector Machine

A separating hyperplane can be written as

$$\mathbf{W} \cdot \mathbf{X} + \mathbf{b} = \mathbf{0}$$

Where W is a n dimensional weight vector and b is referred to as bias.

For a 2D training tuple,if we think b as an additional weight , then the above equation can be re-written as

$$w_o + w_1 x_1 + w_2 x_2 = 0$$

Thus, any point that lies above the separating hyperplane satisfies

$$w_o + w_1 x_1 + w_2 x_2 > 0$$

Similarly, any point that lies below the separating hyperplane satisfies

$$w_o + w_1 x_1 + w_2 x_2 < 0$$

The tuples that belong to class yi= 1 satisfy the hyperplane

$$H1 : w_o + w_1 x_1 + w_2 x_2 \geq 1$$

And the tuples that belong to class yi=-1 satisfy the hyperplane

$$H2 : w_o + w_1 x_1 + w_2 x_2 \leq -1$$

Combining the two inequalities

$$y_i(w_o + w_1 x_1 + w_2 x_2) \geq 1$$

Support Vector Machine

Any training tuples that fall on hyperplanes H1 or H2 (i.e., the "sides" defining the margin) and satisfy the above equation are called **support vectors**. These give the most information about classification but are themselves difficult to classify.

The distance of any point on H1 form the separating hyperplane is $1/\|W\|$.

If W = {w1,w2...wn}, then $\|W\|$ is $\sqrt{w_1^2 + w_2^2 + ...w_n^2}$. This is also equal to the distance of any point on H2 from separating hyperplane. Therefore the maximal marginal distance is $2/\|W\|$.

The MMH can be rewritten as the decision boundary using Lagrangian formulation as

$$\mathrm{d}(X^T) = \sum_{i=1}^{l} y_i \alpha_i X^i X^T + b_o$$

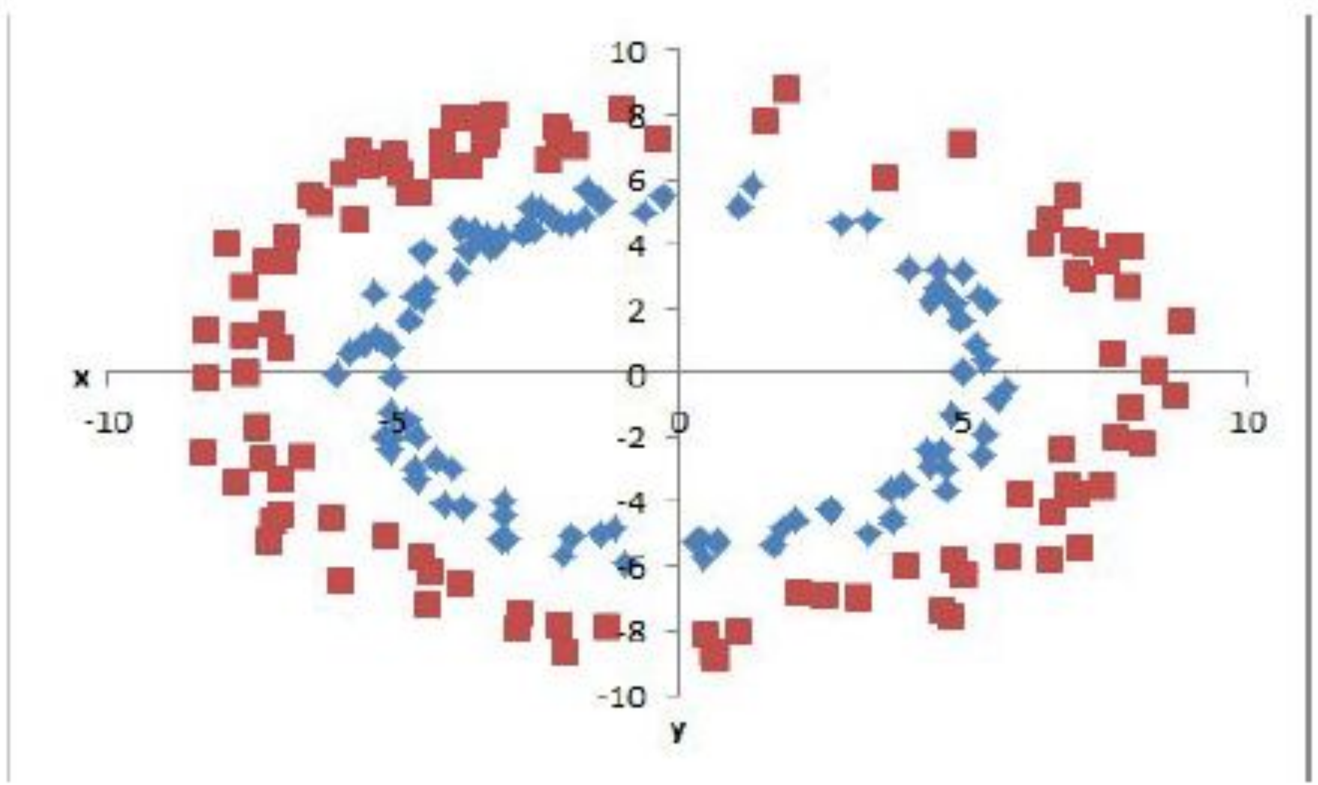
Where y_i is the class label of support vector X^i; X^T is a test tuple; α_i and b_o are numeric parameters that were determined automatically by the optimization and 1 is the number of support vectors. A test tuple X^T

Support Vector Machine

belongs to class +1 if the sign of the result obtained from the above equation is positive and the class prediction is -1 if the sign is negative. The complexity of the classifier depends upon the number of support vectors rather than the dimensionality of data. This makes SVM less prone to overfitting.

"What if the data is not linearly separable"? In such a case no straight line can be found to separate the class. We obtain a non linear SVM by extending the approach of linear SVM. the original input data is mapped into a higher dimensional space using some function Φ. Then SVM finds a linear separating

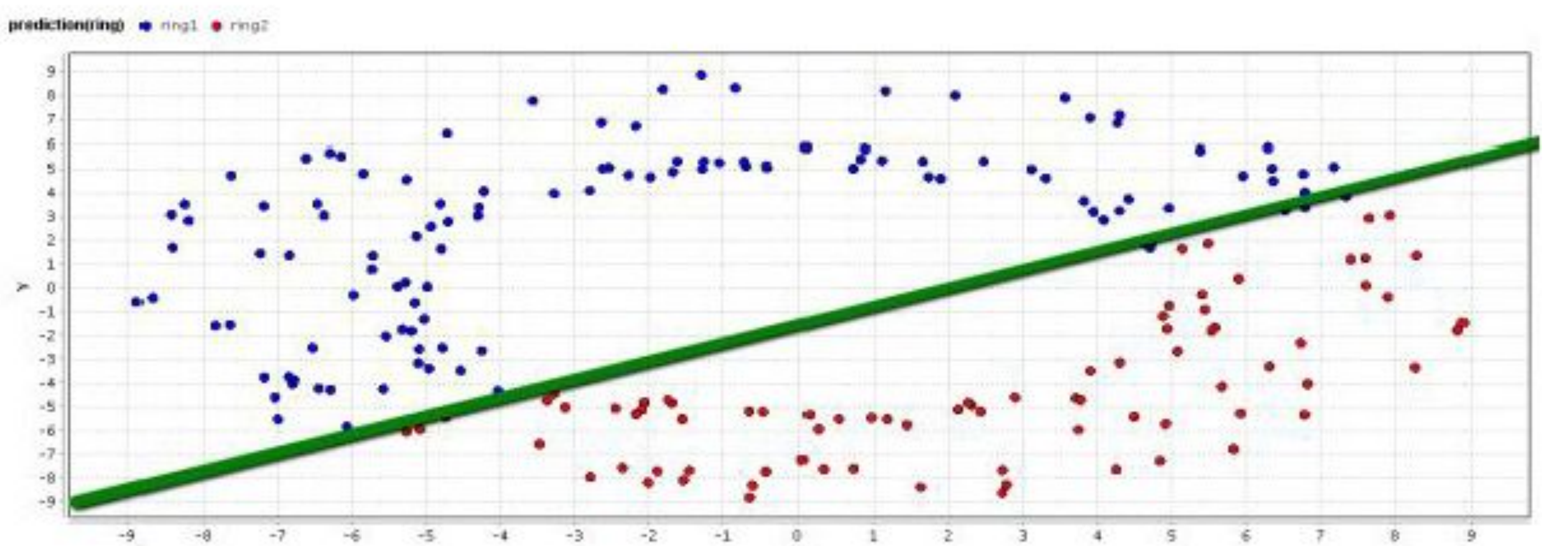
Support Vector Machine



Support Vector Machine

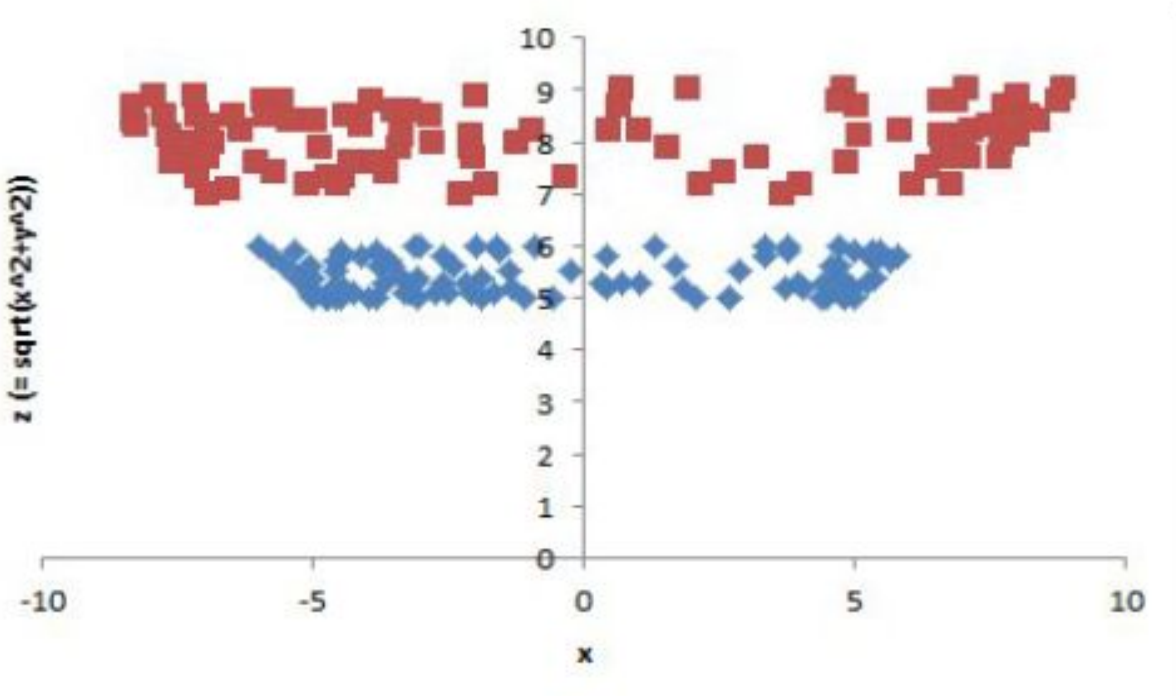
hyperplane with the maximal margin in this higher dimension space .Consider the following example. The figure below shows the data points represented by X=({x,y}) belonging to two classes. It is clear that a straight line cannot be drawn to split them. But intuitively a circular or elliptical hyperplane can separate them.

A linear SVM will classify half the inner ring and half the outer ring correctly giving an accuracy of not more than 50%.



Support Vector Machine

But if we transform the features x and y into a new feature space involving x , y and a new variable $z = \sqrt{x^2 + y^2}$. The data transformed results in a new feature space involving x and z as shown below. Clearly data is now linearly separable and SVM can be applied.



Support Vector Machine

"How do we choose the nonlinear mapping to a higher dimensional space"? There are many kernel functions that can be used to transform the original data into higher feature space. Some of the commonly used kernels are :

- Linear kernel: $K(x_i, x_j) = x_i^T x_j$
- Polynomial kernel: $K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \gamma > 0$
- RBF kernel : $K(x_i, x_j) = \exp(-\gamma \|x_i, - x_j\|^2), \gamma > 0$
- Sigmoid kernel: $K(x_i, x_j) = \tanh(\gamma x_i^T x_j + r)$

Here, γ, r and d are kernel parameters.

So far, we have described linear and nonlinear SVMs for binary (i.e., two-class) classification. SVM classifiers can also be combined for the multiclass case.