Support Vector Machines

The Support Vector Machine is a supervised learning algorithm mostly used for classification but it can be used also for regression. The main idea is that based on the labeled data (training data) the algorithm tries to find the optimal hyperplane which can be used to classify new data points. In two dimensions the hyperplane is a simple line.

Usually a learning algorithm tries to learn the most common characteristics (what differentiates one class from another) of a class and the classification is based on those representative characteristics learnt (so classification is based on differences between classes). The SVM works in the other way around. It finds the most similar examples between classes. Those will be the support vectors. This margin is orthogonal to the boundary and equidistant to the support vectors.

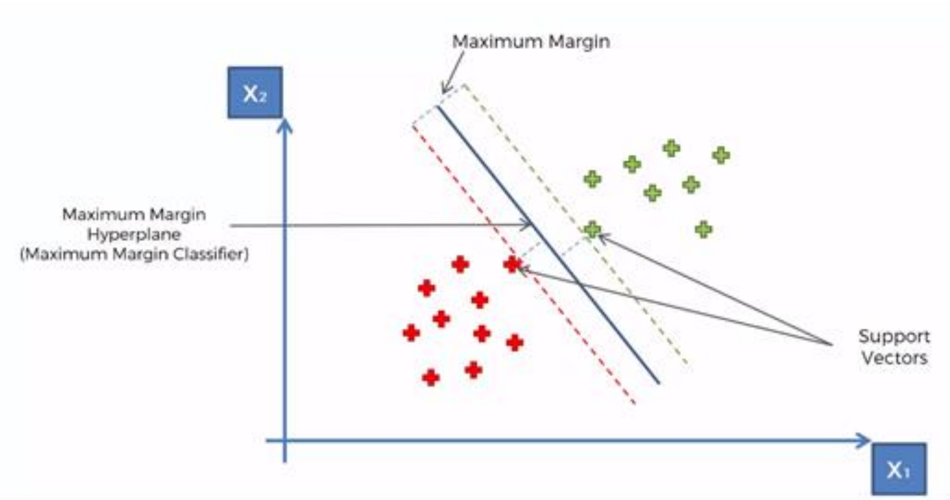

Figure 1: Linearly Separable Data

Hyper-Plane

A hyperplane is a decision boundary that differentiates the two classes in SVM. A data point falling on either side of the hyperplane can be attributed to different classes. The dimension of the hyperplane depends on the number of input features in the dataset. If we have 2 input features the hyper-plane will be a line. likewise, if the number of features is 3, it will become a two-dimensional plane.

Support Vectors

Support vectors are the data points that are nearest to the hyper-plane and affect the position and orientation of the hyper-plane. We have to select a hyperplane, for which the margin, i.e the distance between support vectors and hyper-plane is maximum. Even a little interference in the position of these support vectors can change the hyper-plane.

SVM for Non-Linear Data Sets

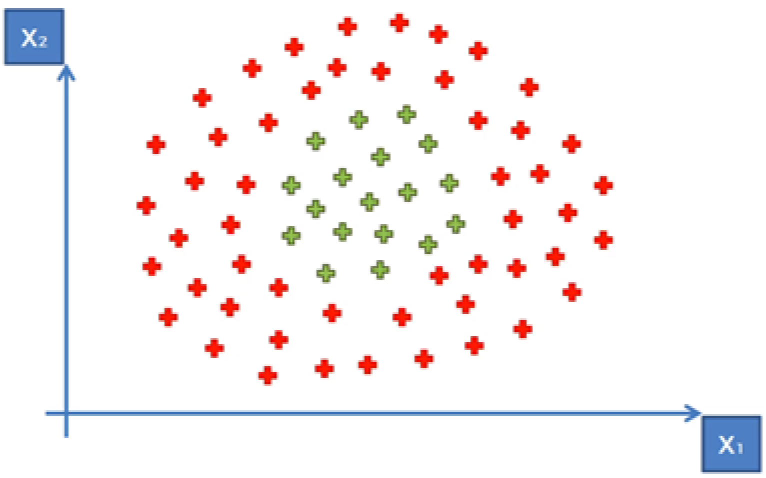

Figure 2: Linearly Non-separable Data

Lets look at a 1D plot that would be easier to understand.

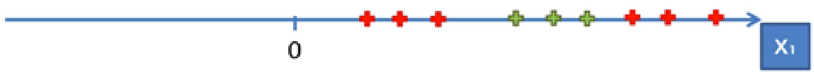

Figure 3: 1-D Plot

We can't have a separable point in the above graph, so we need to do some operations so that they are linearly separable.

Now we apply Change of Origin in the data in such a way that the green data points comes closer to the Origin.

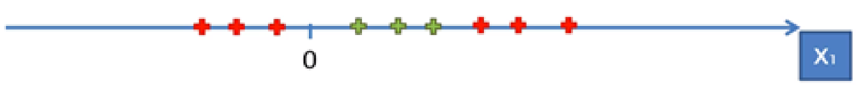

Figure 4: After applying change of origin. $y_1 = (x - c)$

Mapping To Higher Dimension

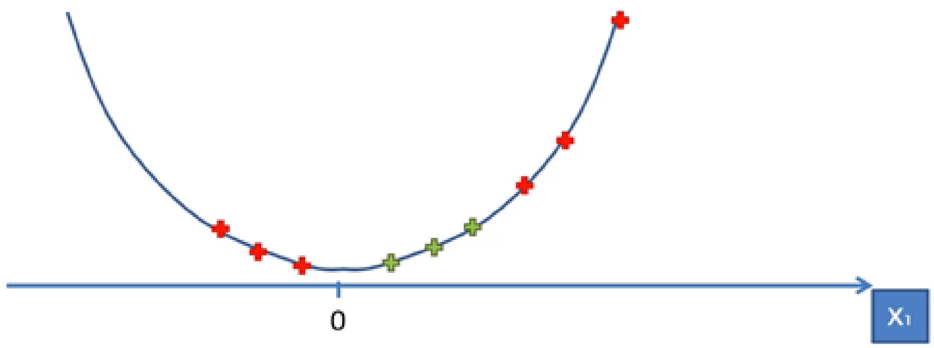

Figure 5: Squared the data to form a parabolic curve $y = (x - c)^2$

Now its linearly separable.

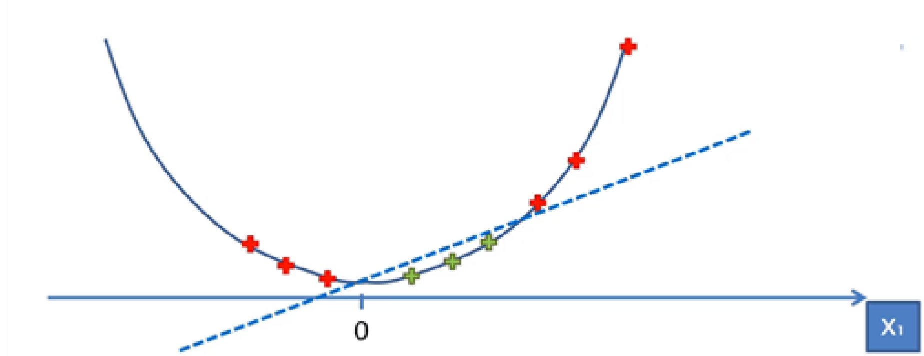

Figure 6: Linearly Separable Data

Similarly, we can use the same technique for Linearly Non-separable 2-D Data. But this process is too compute-intensive, instead we will use a technique called ‘Kernel Trick’.

Kernel Trick

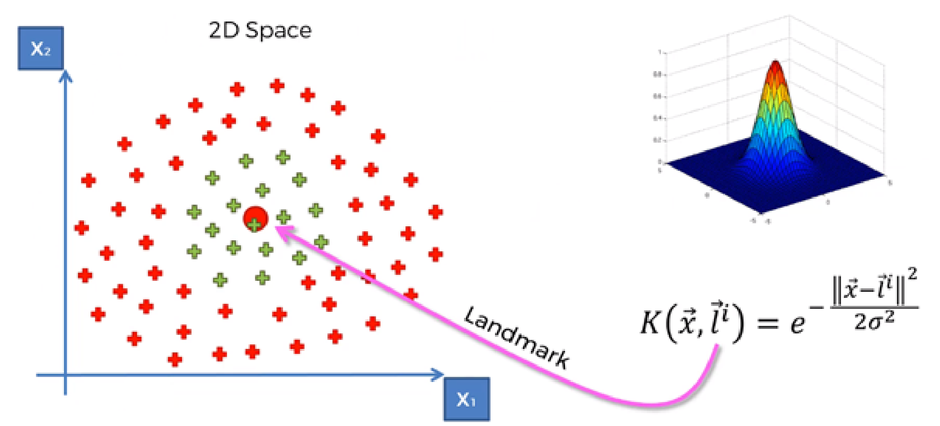

Figure 7: Kernel Trick using Gaussian RBF (Radial Basis Function) Kernel on fig. 2

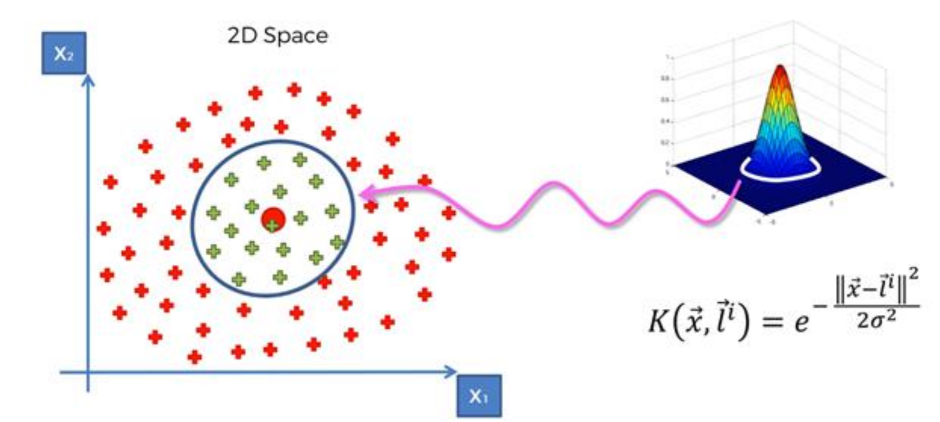

Figure 8: Mapping Hyperplane in 2-D decision boundary can be changed by changing σ.

Resources:

Towards data science

machine learning a-z udemy

analytics vidya