Machine Learning

Session 9

21st April 2018

Rama Krishna Bhupathi
ramakris@gmail.com

Agenda

Ensemble Learning

Support Vector Machines

Ensemble Learning

Ensemble is the art of combining diverse set of learners (individual models) together to improvise on the stability and predictive power of the model.

Ensemble modeling is a powerful way to improve the performance of your model. It usually pays off to apply ensemble learning over and above various models you might be building.

Ensemble Learning

Ensemble methods are meta-algorithms that combine several machine learning techniques into one predictive model in order to:

- Decrease variance (bagging)
- Reduce bias (boosting)
- Improve predictions (stacking).

Ex: Netflix's movie recommendation algorithm was Ensemble algorithm (winner)

Ensemble Learning

Bagging

Bagging stands for bootstrap aggregation. One way to reduce the variance of an estimate is to average together multiple estimates. For example, we can train M different trees on different subsets of the data (chosen randomly with replacement) and compute the ensemble:

$$f(x) = 1/M \sum_{m=1}^{M} f_m(x)$$

Ensemble Learning

A great resource is

https://blog.statsbot.co/ensemble-learning-d1dcd548e936

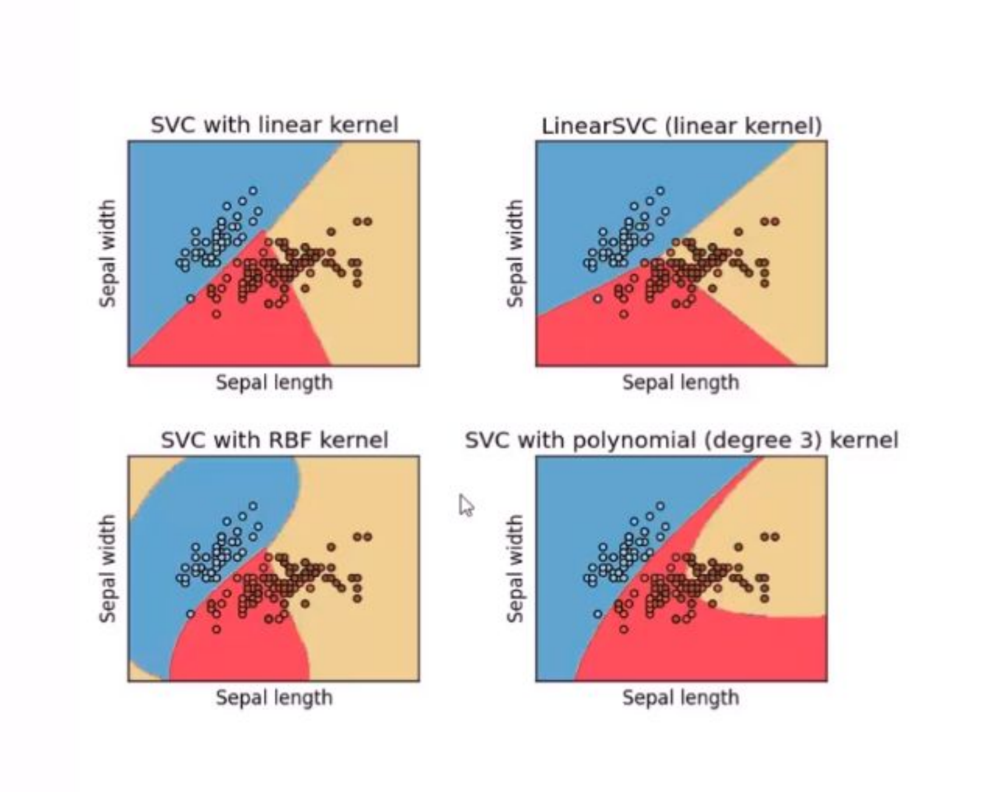
Support Vector Machines

- Supervised machine learning algorithm (K Means was unsupervised)
- Mostly used for classification
- Clustering High Dimensional Data(lots of features)
- Uses *kernel* trick to represent data in higher dimensional space to find hyperplanes that might not be apparent in lower dimensions.
- Finds higher-dimensional support vector across which to divide the data (mathematically these support vectors define hyperplanes
- Complicated math underneath.

Support Vector Machines

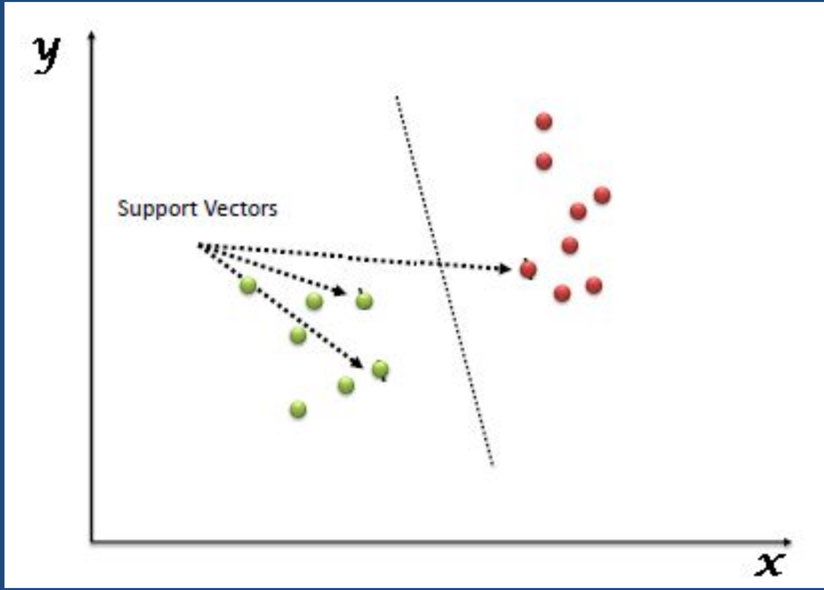
Classifying the iris data
Set with various kernels

Some kernels will work better than the others.



Support Vector Machines

A Support Vectors can simply be the coordinates of an individual observation. Support Vector Machine is a frontier which best segregates the two classes (hyper-plane/ line).

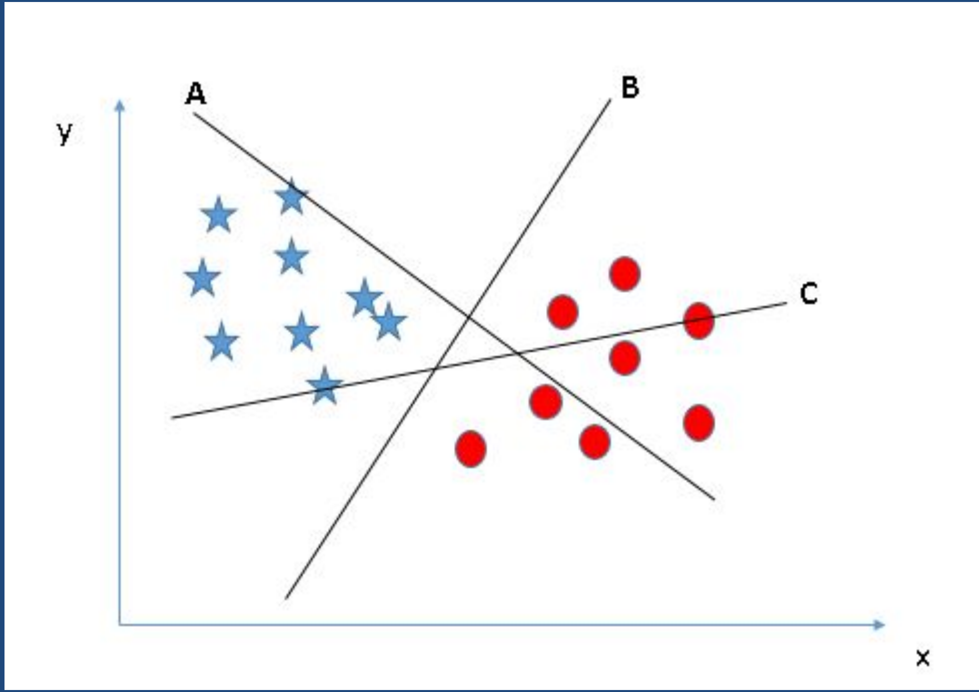


SVM (Identify the right hyper plane)

Scenario 1

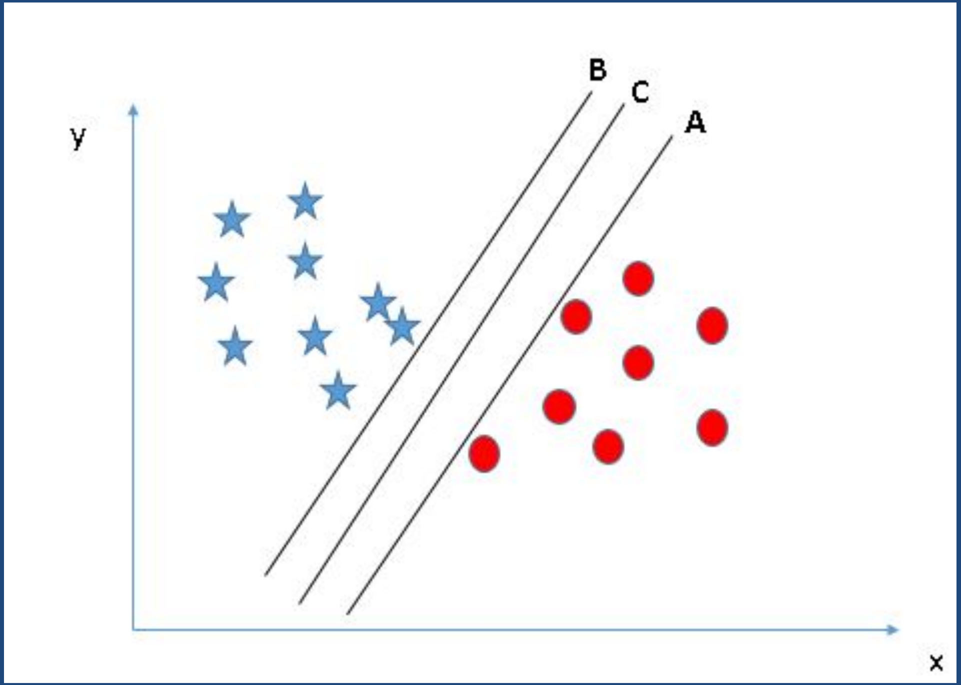
Here, we have three hyper-planes (A, B and C). Now, identify the right hyper-plane to classify star and circle.

In this scenario, hyper-plane “B” has excellently performed this job



SVM (Identify the right hyper plane)
Scenario 2

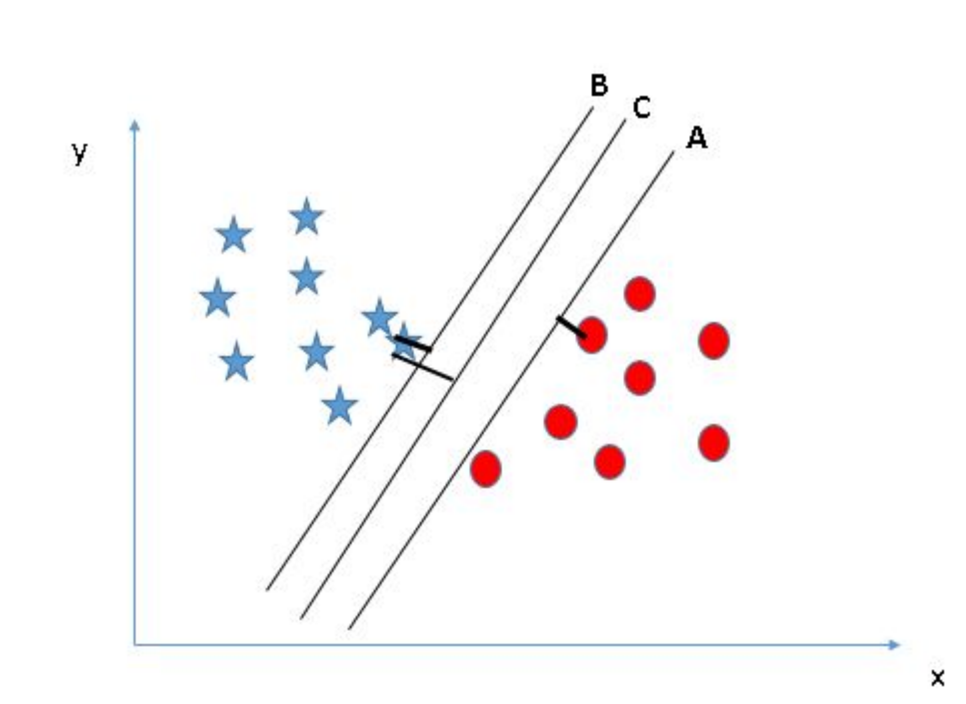
Here, we have three hyper-planes (A, B and C) and all are segregating the classes well.
Now, How can we identify the right hyper-plane?



SVM (Identify the right hyper plane)

Scenario 2...cont

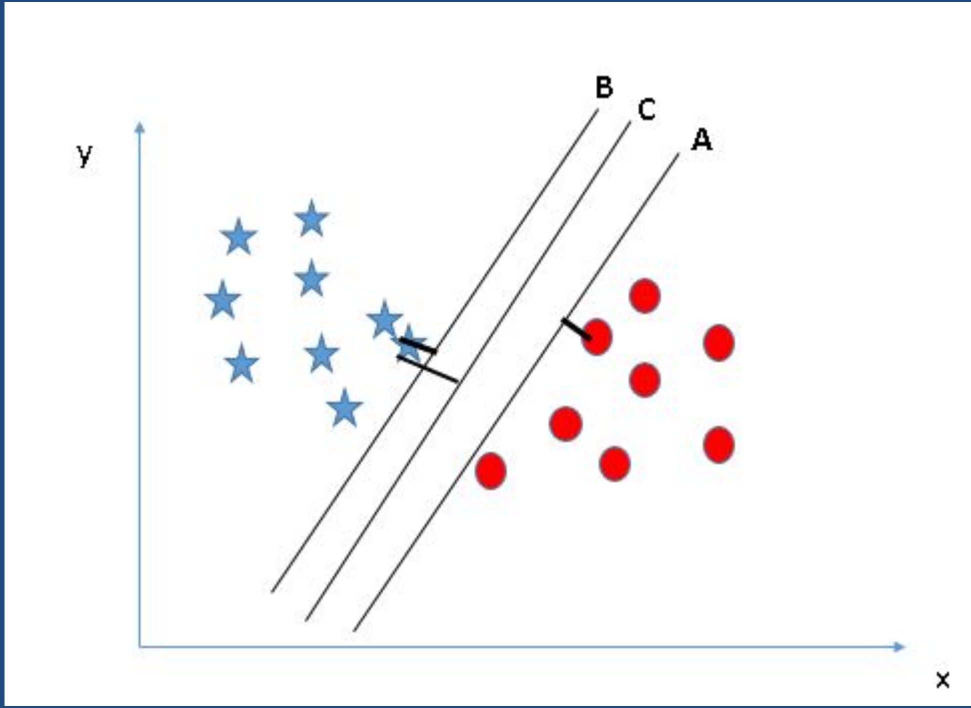
Here, maximizing the distances between nearest data point (either class) and hyper-plane will help us to decide the right hyper-plane. This distance is called as Margin. Let’s look at the snapshot:



SVM (Identify the right hyper plane)

Scenario 2...cont

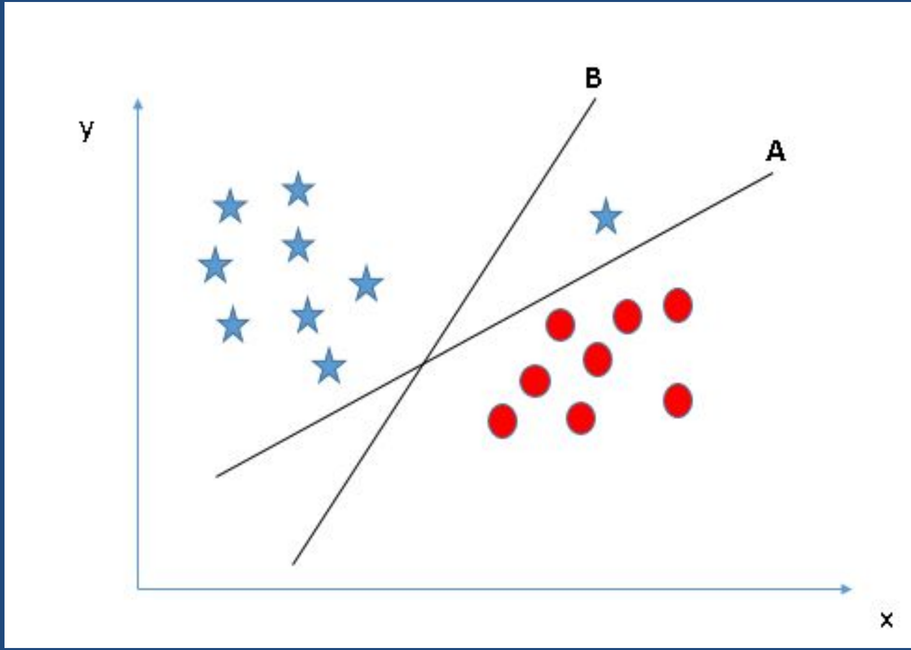
you can see that the margin for hyper-plane C is high as compared to both A and B. Hence, we name the right hyper-plane as C. Another lightning reason for selecting the hyperplane with higher Margin is robustness. If we select a hyper-plane having low margin then there is high chance of mis-classification.



SVM (Identify the right hyper plane)

Scenario 3

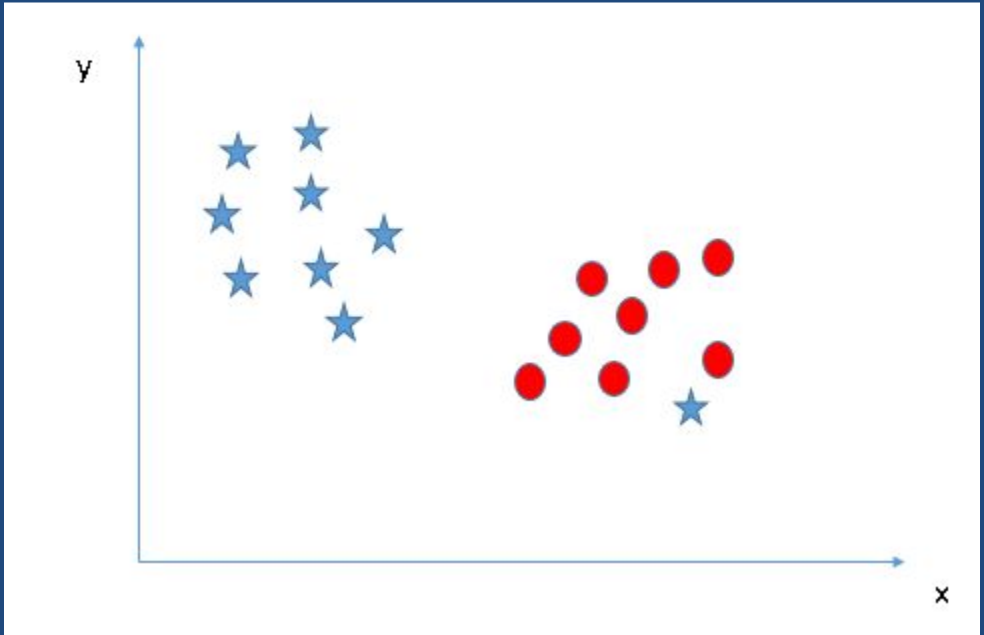
Some of you may have selected the hyper-plane B as it has higher margin compared to A. But, here is the catch, SVM selects the hyper-plane which classifies the classes accurately prior to maximizing margin. Here, hyper-plane B has a classification error and A has classified all correctly. Therefore,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



SVM (Identify the right hyper plane)

Scenario 4

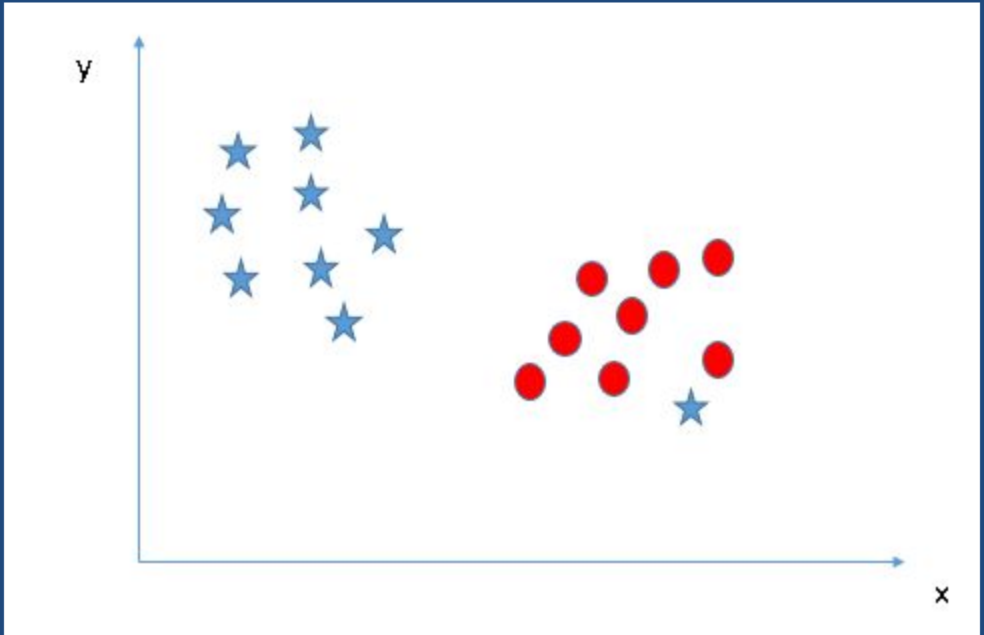
Below, I am unable to segregate the two classes using a straight line, as one of star lies in the territory of other(circle) class as an outlier.



SVM (Identify the right hyper plane)

Scenario 4...cont

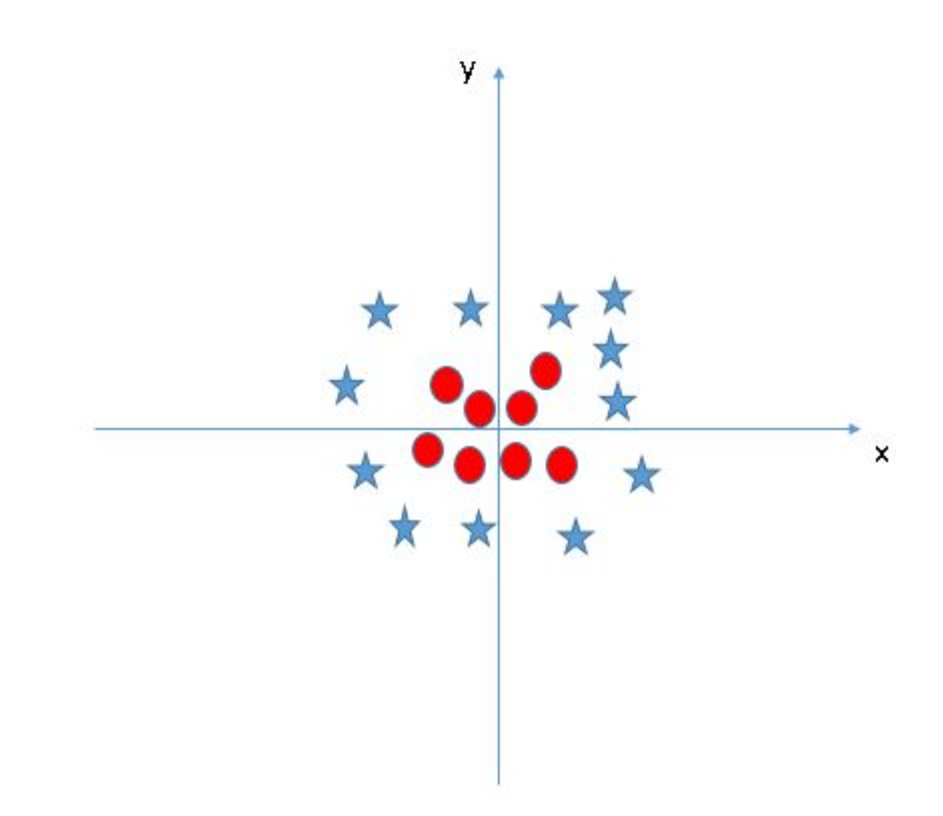
As I have already mentioned, one star at other end is like an outlier for star class. SVM has a feature to ignore outliers and find the hyper-plane that has maximum margin. Hence, we can say, SVM is robust to outliers.



SVM (Identify the right hyper plane)

Scenario 5

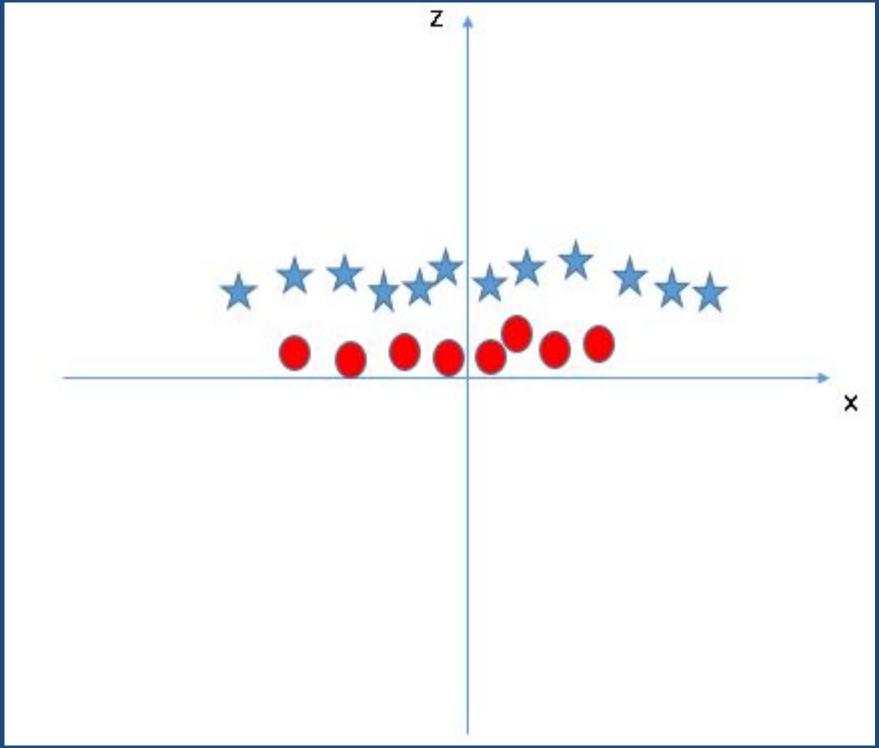
In the scenario below, we can’t have linear hyperplane between the two classes, so how does SVM classify these two classes? Till now, we have only looked at the linear hyperplane.



SVM (Identify the right hyper plane)

Scenario 5...cont

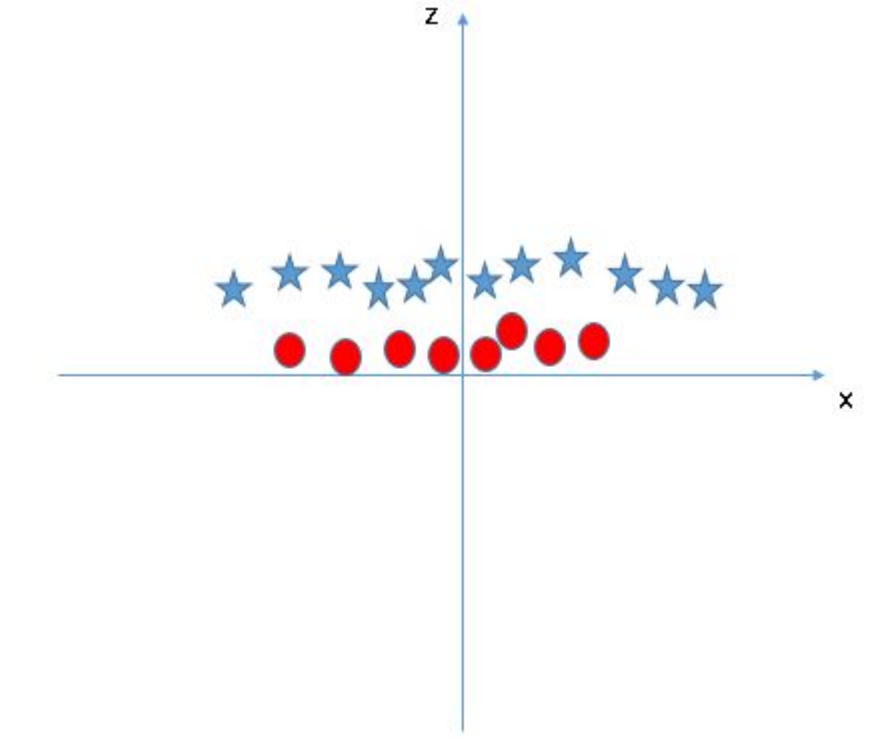
SVM can solve this problem. Easily! It solves this problem by introducing additional feature. Here, we will add a new feature $z=x^2+y^2$.
 Now, let's plot the data points on axis x and z:



SVM (Identify the right hyper plane)

Scenario 5...cont

All values for z would be positive always because z is the squared sum of both x and y.In the original plot, red circles appear close to the origin of x and y axes, leading to lower value of z and star relatively away from the origin result to higher value of z.



Pros and Cons associated with SVM

Pros:

- It works really well with clear margin of separation
- It is effective in high dimensional spaces.
- It is effective in cases where number of dimensions is greater than the number of samples.
- It uses a subset of training points in the decision function (called support vectors), so it is also memory efficient.

Cons:

- It doesn't perform well, when we have large data set because the required training time is higher
- It also doesn't perform very well, when the data set has more noise i.e. target classes are overlapping