

Support Vector Machines - Kernel Explained

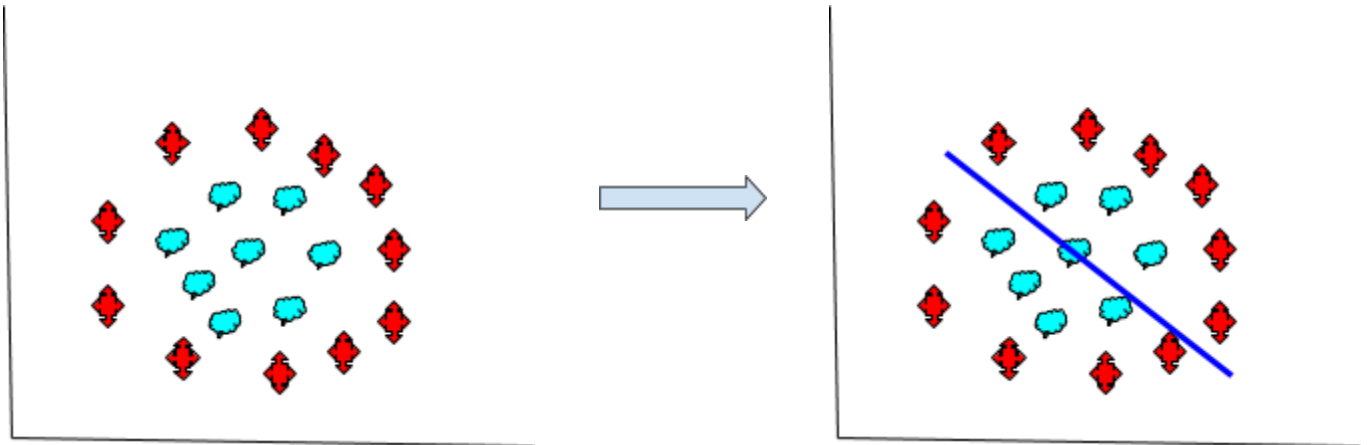
In the last post we saw what actually an SVM is and what it does. Please follow the link below to catch up with the happenings.

Link: <https://codingmachinelearning.wordpress.com/2016/07/13/support-vector-machines-svm/>

I ended the last post, by saying if we have a not-linearly separable data set then the linear SVM decision boundary will fail miserably !! So what to do then? To help us solve this problem we have SVMs which can give non linear decision boundary. Hurray we have a solution - but before that you need to have a question in your mind.

SVM gives hyper-plane as its decision boundary, then how can it be nonlinear ??

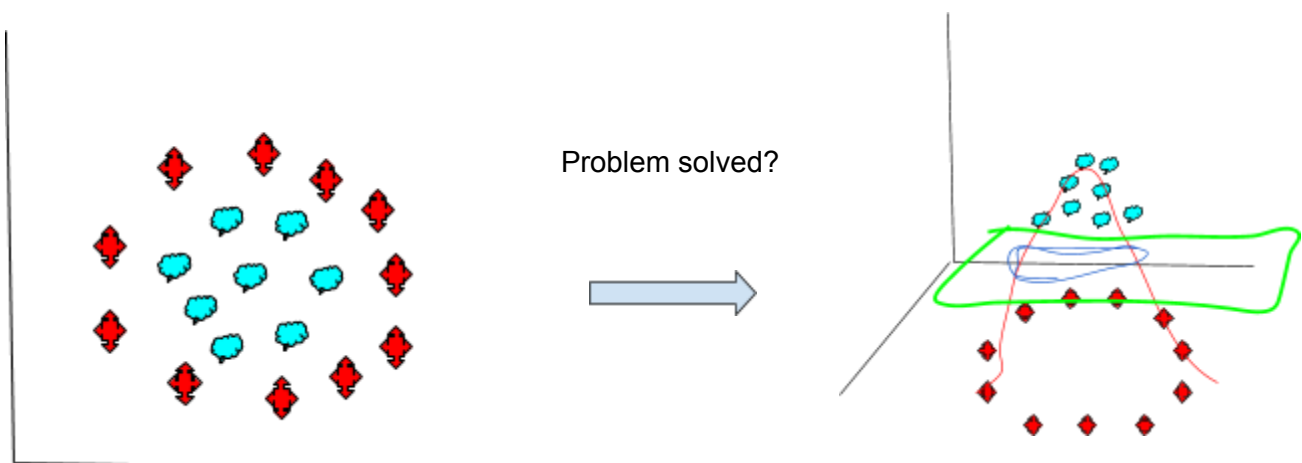
Answer to the above question is - you are right. This is what we are going to deal in this post - SVM Kernels

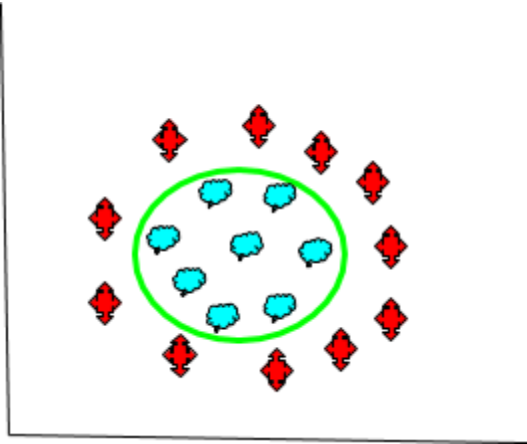


This was not giving us good results. Now how do get a non-linear decision boundary for the above data set? The answer is to use kernels.

Kernel: This actually is a **function** that maps the data to a higher dimension where the data is separable

In simple words, the data is not linearly separable in this 2-dimension. So I will project the data up by some number of dimensions using a mapping function. In that dimension plane the data will be separable.





Once we get the hyper plane in the $n+k$ dimensional space - where k is the projection of the data up from n dimensional space we can project the hyper-plane back down again to get a decision boundary that can separate points.

In very simple terms this is what SVM Kernel does. It projects the data up by k dimensions such that the data points are now separable in the higher dimensional plane.

Some popular choice for kernel functions are:

- Gaussian Kernel: https://en.wikipedia.org/wiki/Gaussian_function
- Polynomial Kernel: https://en.wikipedia.org/wiki/Polynomial_kernel

Recap:

- SVMs are powerful in the sense that they can generate non-linear decision boundaries.
- Kernels help in projecting data to a higher dimensional space where the points can be linearly separated.
- Popular ones are RBF (also called the gaussian) and polynomial

In practise we rarely write our own kernel function. For most practical purposes we use the default gaussian kernel. Nevertheless we can write our own kernel function and see how it works. We will deal with this in our next post. Till then Bye.

Reference:

- <http://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html#sklearn.svm.SVC>
- <https://www.quora.com/What-are-Kernels-in-Machine-Learning-and-SVM>
- <http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-034-artificial-intelligence-fall-2010/lecture-videos/lecture-16-learning-support-vector-machines/>