Kernels

It is not possible to find a hyperplane or a linear decision boundary for some classification problems. If we project the data in to a higher dimension from the original space, we may get a hyperplane in the projected dimension that helps to classify the data.

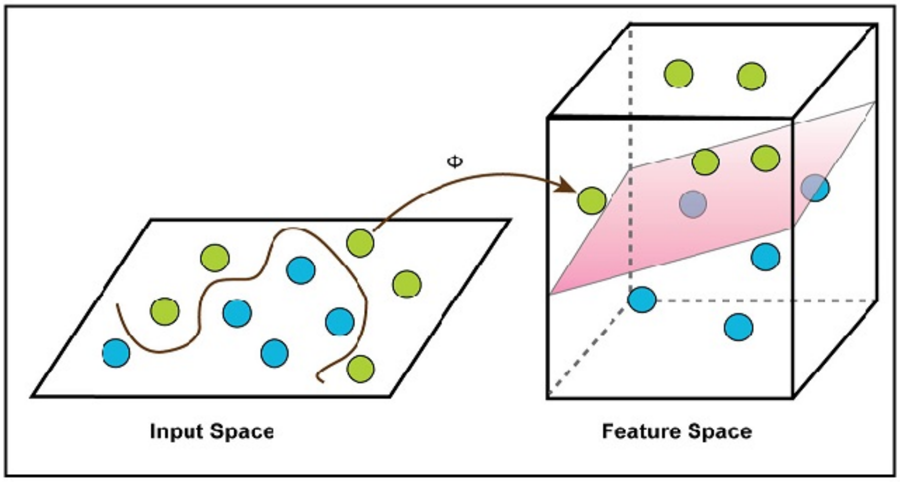

Figure 1

As shown in the above figure, it is impossible to find a single line to separate the two classes (green and blue) in the input space. But, after projecting the data in to a higher dimension (i.e. feature space in the figure), we could able to find the hyperplane which classifies the data. Kernel helps to find a hyperplane in the higher dimensional space.

Mathematical definition:

$K(x, y) =< \phi(x), \phi(y) >$. Here, K is the kernel function, x, y are n dimensional inputs. ϕ is a map from n-dimension to m-dimension space. $< x, y >$ denotes the dot product. Usually, m is much larger than n.

Type of Kernels

- linear: $K(x, y) = x^T y$.
- polynomial: $K(x, y) = (\gamma x^T y + r)^d, \gamma > 0$.
- radial basis function (RBF): $K(x, y) = \exp(-\gamma||x - y||^2)\gamma > 0$.
- sigmoid: $K(x, y) = \tanh(\gamma x^T y + r)$.

Example: Polynomial Kernel

Suppose $\phi : \mathbb{R}^2 \to \mathbb{R}^3$ is a function which is projecting the data from 2D to 3D i.e.,
ϕ is taking $(x_1, x_2) \mapsto (z_1, z_2, z_3) := (x_1^2, \sqrt{2}x_1x_2, x_2^2)$ which can be seen in the following figure.

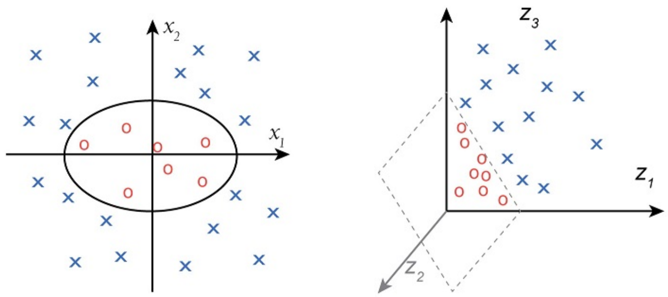

Figure 2

We get above projection by using polynomial kernel where $\gamma = 1$, $r = 0$ and $d = 2$.
Now, in this 3D space, we might be able to find a hyper-plane $az_1 + bz_2 + cz_3 = 0$ to separate the data into two classes with a large margin because the data is more separable in higher dimension space.

Mathematical derivation for above polynomial kernel (exception)

Let $x = (x_1, x_2)$ and $x^{'} = (x_1^{'}, x_2^{'})$

$$\begin{aligned}
K(x, x^{'}) &= \phi(x).\phi(x^{'}) \\
&= \phi(x_1, x_2).\phi(x_1^{'}, x_2^{'}) \\
&= (x_1^2, \sqrt{2}x_1x_2, x_2^2).(x_1^{'2}, \sqrt{2}x_1^{'}x_2^{'}, x_2^{'2}) \\
&= (x_1x_1^{'})^2 + 2x_1x_2x_1^{'}x_2^{'} + (x_2x_2^{'})^2 \\
&= (x_1x_1^{'} + x_2x_2^{'})^2 \\
&= ((x_1, x_2).(x_1^{'}, x_2^{'}))^2 \\
&= (x.x^{'})^2
\end{aligned} \tag{1}$$

References:

For more details on Kernels,
https://en.wikipedia.org/wiki/Support_vector_machine#Kernel_trick