Kernels

The kernel function $K(\mathbf{x}, \mathbf{x}')$ is symmetric and positive semidefinite, and can be thought of as a measure of similarity between the samples $\mathbf{x}$ and $\mathbf{x}'$. In Machine Learning, kernels are employed when mapping non-linearly separable data in the input space into a higher-dimensional feature space where it can be linearly separated.

Consider $\phi(\mathbf{x})$, which is known as the feature map, and maps the sample $\mathbf{x}$ in the input space to a higher-dimensional feature space. For any two samples $\mathbf{x}$ and $\mathbf{x}'$ in the input space, the kernel function is defined as the dot product of $\phi(\mathbf{x})$ and $\phi(\mathbf{x}')$:

$$K(x, x') = \phi(\mathbf{x}) \cdot \phi(\mathbf{x}') \tag{1}$$

Since the dot product measures the similarity of vectors (if we normalise the vectors, the dot product will tell us the similarity of the directions of the vectors), K can be interpreted as a similarity measure between two elements of the input space.

The main advantage of using kernel functions in Machine Learning is efficiency. Instead of explicitly defining a feature map and computing the dot product $\phi(\mathbf{x}) \cdot \phi(\mathbf{x}')$, we can choose a kernel function that implicitly defines a feature map. When using a kernel function we operate in the original input space without having to compute the coordinates of the samples in the higher-dimension feature space. This is known as the kernel trick.

The Polynomial Kernel

The polynomial kernel is defined as

$$K(\mathbf{x}, \mathbf{x}') = (\mathbf{x} \cdot \mathbf{x}' + c)^d \tag{2}$$

where c is a constant, and d is the polynomial degree. Let's see how this works with an example:

Say we have a two-dimensional input space where

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \quad \mathbf{x}' = \begin{pmatrix} x_1' \\ x_2' \end{pmatrix}$$

For a second-degree polynomial kernel we obtain

$$\begin{aligned} K(\mathbf{x}, \mathbf{x}') &= (x_1 x_1' + x_2 x_2' + c)^2 \\ &= (x_1^2 x_1'^2 + x_2^2 x_2'^2 + 2x_1 x_1' x_2 x_2' + 2c x_1 x_1' + 2c x_2 x_2' + c^2) \end{aligned}$$

Therefore, from (1) we must have

$$\phi(\mathbf{x}) = \begin{pmatrix} x_1^2 \\ x_2^2 \\ \sqrt{2} x_1 x_2 \\ \sqrt{2c} x_1 \\ \sqrt{2c} x_2 \\ c \end{pmatrix}$$

Hence, in this example we have mapped from a two-dimensional input space to a six-dimensional feature space. However, we do not actually need to compute the mapping explicitly, or even know it (as long as we know that it exists). The kernel function implicitly defines the mapping, and computing (2) will yield the same result as $\phi(\mathbf{x}) \cdot \phi(\mathbf{x}')$.

Now, consider the homogeneous case (c = 0). The second-degree polynomial kernel is

$$\begin{aligned} K(\mathbf{x}, \mathbf{x}') &= (x_1 x_1' + x_2 x_2')^2 \\ &= (x_1^2 x_1'^2 + x_2^2 x_2'^2 + 2 x_1 x_1' x_2 x_2') \end{aligned}$$

Therefore, the feature map is

$$\phi(\mathbf{x}) = \begin{pmatrix} x_1^2 \\ x_2^2 \\ \sqrt{2} x_1 x_2 \end{pmatrix}$$

So, this time the mapping is to a three-dimensional feature space (see Figure 1).

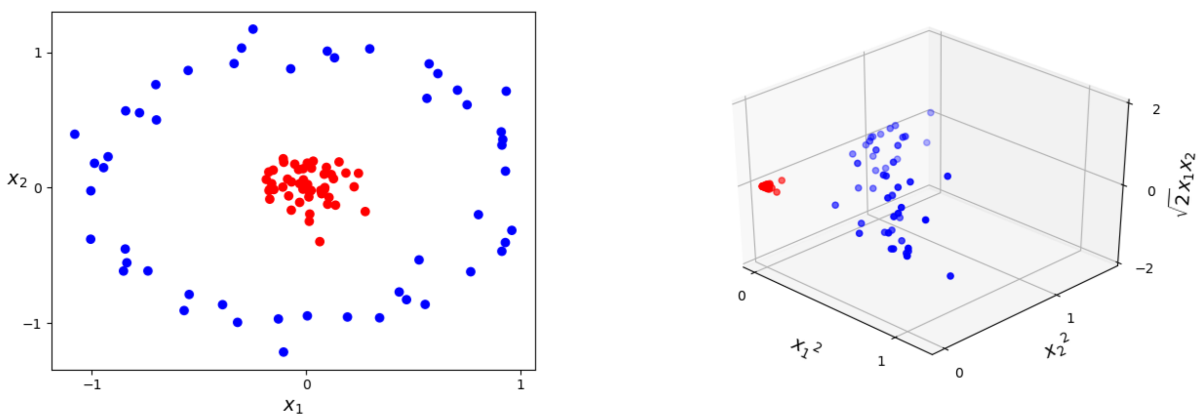

Figure 1: (left) A non-linearly separable, two-dimensional data set. (right) The data is mapped to a three-dimensional feature space, where it is linearly separable.

The Radial Basis Function Kernel

The Radial Basis Function (RBF) kernel is defined as

$$K(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{||\mathbf{x} - \mathbf{x}'||^2}{2\sigma^2}\right) \tag{3}$$

where σ is a scaling parameter. The RBF kernel measures the Euclidean distance between the samples $\mathbf{x}$ and $\mathbf{x}'$ and returns a dot product in the feature space that depends on how close together the samples are. From equation (3) it is clear that the smaller the distance between the samples (the more similar the samples are to one another), the larger the value of the kernel function. The RBF kernel can be expanded as

$$\exp\left(-\frac{||\mathbf{x}-\mathbf{x}'||^2}{2\sigma^2}\right) = \exp\left[-\frac{(\mathbf{x}-\mathbf{x}')(\mathbf{x}-\mathbf{x}')}{2\sigma^2}\right] \tag{4}$$

$$= \exp\left(\frac{\mathbf{x}\cdot\mathbf{x}'}{\sigma^2}\right)\exp\left(-\frac{||\mathbf{x}||^2}{2\sigma^2}\right)\exp\left(-\frac{||\mathbf{x}'||^2}{2\sigma^2}\right) \tag{5}$$

$$= \left[1+\frac{\mathbf{x}\cdot\mathbf{x}'}{\sigma^2}+\frac{1}{2!}\left(\frac{\mathbf{x}\cdot\mathbf{x}'}{\sigma^2}\right)^2+...\right]\exp\left(-\frac{||\mathbf{x}||^2}{2\sigma^2}\right)\exp\left(-\frac{||\mathbf{x}'||^2}{2\sigma^2}\right) \tag{6}$$

$$= \exp\left(-\frac{||\mathbf{x}||^2}{2\sigma^2}\right)\exp\left(-\frac{||\mathbf{x}'||^2}{2\sigma^2}\right)\sum_{n=0}^{\infty}\frac{1}{n!}\left(\frac{\mathbf{x}\cdot\mathbf{x}'}{\sigma^2}\right)^n \tag{7}$$

In the last line the two exponential terms are constant. The final term is just an infinite sum of the polynomial kernel (with $c = 0$) divided by some constants. Therefore, the mapping to this infinite-dimensional feature space can be expressed as the dot product of the feature maps, as is required.

If we define the homogeneous polynomial kernel of degree n as

$$\phi^n(\mathbf{x})\cdot\phi^n(\mathbf{x}') = (\mathbf{x}\cdot\mathbf{x}')^n \tag{8}$$

we can write the expanded RBF kernel as

$$\exp\left(-\frac{||\mathbf{x}-\mathbf{x}'||^2}{2\sigma^2}\right) = \exp\left(-\frac{||\mathbf{x}||^2}{2\sigma^2}\right)\exp\left(-\frac{||\mathbf{x}'||^2}{2\sigma^2}\right)\sum_{n=0}^{\infty}\frac{\phi^n(\mathbf{x})\cdot\phi^n(\mathbf{x}')}{n!\,\sigma^{2n}} \tag{9}$$

and the feature map as

$$\phi_{RBF}(\mathbf{x}) = \exp\left(-\frac{||\mathbf{x}||^2}{2\sigma^2}\right)\left(1,\ \frac{\phi^1(\mathbf{x})}{\sigma},\ \frac{\phi^2(\mathbf{x})}{\sqrt{2!}\,\sigma^2},\ \frac{\phi^3(\mathbf{x})}{\sqrt{3!}\,\sigma^3},\ ...\right)^T \tag{10}$$

For a more concrete example, let's say that $\mathbf{x}$ and $\mathbf{x}'$ are vectors in one-dimensional space with components x and x', respectively. From equation (7) we obtain

$$\exp\left(-\frac{||\mathbf{x}-\mathbf{x}'||^2}{2\sigma^2}\right) = \exp\left(-\frac{x^2}{2\sigma^2}\right)\exp\left(-\frac{x'^2}{2\sigma^2}\right)\sum_{n=0}^{\infty}\frac{1}{n!}\left(\frac{xx'}{\sigma^2}\right)^n \tag{11}$$

$$= \exp\left(-\frac{x^2}{2\sigma^2}\right)\exp\left(-\frac{x'^2}{2\sigma^2}\right)\left(1+\frac{xx'}{\sigma^2}+\frac{(xx')^2}{2!\,\sigma^4}+\frac{(xx')^3}{3!\,\sigma^6}+...\right) \tag{12}$$

Therefore, the feature map is

$$\phi_{RBF}(\mathbf{x}) = \exp\left(-\frac{x^2}{2\sigma^2}\right)\left(1,\ \frac{x}{\sigma},\ \frac{x^2}{\sqrt{2!}\,\sigma^2},\ \frac{x^3}{\sqrt{3!}\,\sigma^3},\ ...\right)^T \tag{13}$$

Support Vector Machines (SVMs) and Kernels

The dual form of the soft-margin SVM optimisation problem is

$$\begin{aligned}
&\min_{\alpha} \quad \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} \alpha^i\alpha^j y^i y^j \mathbf{x}^i \cdot \mathbf{x}^j - \sum_{i=1}^{m} \alpha^i \\
&\text{subject to} \quad \sum_{i=1}^{m} \alpha^i y^i = 0 \\
&0 \leq \alpha^i \leq C, \quad i = 1, 2, ..., m
\end{aligned} \tag{14}$$

In the function to be minimised, the dot product occurs only between the samples $\mathbf{x}^i$ and $\mathbf{x}^j$, and does not involve any of the other parameters. Therefore, if the data set is non-linearly separable, as well as employing the soft-margin approach, we can choose to map the samples to a higher-dimensional space, with the feature map $\phi(\mathbf{x}^i)$ for the sample $\mathbf{x}^i$. Then, the function to be minimised (subject to the same constraints as above) would become

$$\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} \alpha^i\alpha^j y^i y^j \phi(\mathbf{x}^i) \cdot \phi(\mathbf{x}^j) - \sum_{i=1}^{m} \alpha^i \tag{15}$$

which is

$$\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} \alpha^i\alpha^j y^i y^j K(\mathbf{x}^i, \mathbf{x}^j) - \sum_{i=1}^{m} \alpha^i \tag{16}$$

and the weight vector solution to the soft-margin SVM problem would be

$$\mathbf{w} = \sum_{i=1}^{m} \alpha^i y^i \phi(\mathbf{x}^i) \tag{17}$$

References

Deisenroth, M. S., Faldo, A. A., Ong, C. S. (2020). *Mathematics for Machine Learning*. Cambridge University Press.

Mohri, M., Rostamizadeh, A., Talwalker, A. (2012). *Foundations of Machine Learning*. The MIT Press, Cambridge, Massachusetts.

Shalev-Shwartz, S., Ben-David, S. (2014). *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, New York, USA.