Kernel Method

Tuan Nguyen

November 22, 2022

Overview

Dual representation

Constructing kernels

Kernel function

Gaussian Kernel

Dual representation

- Many linear models for regression and classification can be reformulated in terms of a dual representation in which the kernel function arises naturally.
- This concept will play an important role when we consider support vector machines.

L2 regularization loss

$$J(W) = \frac{1}{2}\sum_{i=1}^{N}(w^T\phi(x_n) - t_n)^2 + \frac{\lambda}{2}w^T w \tag{1}$$

where $\lambda > 0$

Dual representation (cont.)

If we set the gradient of $J(w)$ with respect to w equal to zero

$$w = -\frac{1}{\lambda}\sum_{i=1}^{N}(w^T\phi(x_n) - t_n)\phi(x_n) \tag{2}$$

$$= \sum_{n=1}^{N} a_n\phi(x_n) = \Phi^T a \tag{3}$$

where Φ is the design matrix, whose n^{th} row is given by $\phi(x_n)^T$. Here the vector a $= (a_1, ..., a_N)^T$, and we have defined

$$a_n = -\frac{1}{N}(w^T\phi(x_n) - t_n) \tag{4}$$

Instead of working with the parameter vector w, we can now reformulate the least squares algorithm in terms of the parameter vector a, giving rise

Dual representation (cont.)

to a **dual representation**. If we substitute $w = \Phi^T a$ into J(w), we obtain

$$J(a) = \frac{1}{2} a^T \Phi \Phi^T \Phi \Phi^T a - a^T \Phi \Phi^T t + \frac{1}{2} t^T t + \frac{\lambda}{2} a^T \Phi \Phi^T a \tag{5}$$

where t $= (t_1, ..., t_N)^T$. We define the Gram matrix K $= \Phi \Phi^T$, which is an NxN symmetric matrix with elements

$$K_{nm} = \Phi(x_n)^T \Phi(x_m) = k(x_n, x_m) \tag{6}$$

where k is the kernel function. We have

$$J(a) = \frac{1}{2} a^T K K a - a^T K t + \frac{1}{2} t^T t + \frac{\lambda}{2} a^T K a \tag{7}$$

Setting the gradient of J(a) with respect to a to zero, we obtain the following solution

$$a = (K + \lambda I_N)^{-1} t \tag{8}$$

Dual representation (cont.)

The prediction

$$y(x) = w^T \phi(x) = a^T \Phi \phi(x) = k(x)^T (K + \lambda I_N)^{-1}) t$$

where we have defined the vector k(x) with elements $k_n(x) = k(x_n, x)$. The dual formulation allows the solution to the least-squares problem to be expressed entirely in terms of the kernel function $k(x, x')$.

- We determine the parameter vector a by inverting an N $\times$N matrix, whereas in the original parameter space formulation we had to invert an M $\times$ M matrix in order to determine w. Because N is typically much larger than M, the dual formulation does not seem to be particularly useful.
- The advantage of the dual formulation is that it is expressed entirely in terms of the kernel function k(x, x'). We can therefore work directly in terms of kernels and avoid the explicit introduction of the feature vector $\phi(x)$, which allows us implicitly to use feature spaces of high, even infinite, dimensionality.

Constructing kernels

We must ensure that the function we choose is a valid kernel, in other words that it corresponds to a scalar product in some (perhaps infinite dimensional) feature space. For example: Suppose $x = (x_1, x_2)$

$$k(x, z) = (x^T z)^2 = (x_1 z_1 + x_2 z_2)^2 \tag{9}$$
$$= x_1^2 z_1^2 + 2 x_1 z_1 x_2 z_2 + x_2^2 z_2^2 \tag{10}$$
$$= (x_1^2, \sqrt{2} x_1 x_2, x_2^2)(z_1^2, \sqrt{2} z_1 z_2, z_2^2)^T \tag{11}$$
$$= \phi(x)^T \phi(z) \tag{12}$$

A necessary and sufficient condition for a function k(x, x') to be a valid kernel is that the Gram matrixK, whose elements are given by $k(x_n, x_m)$, should be positive semidefinite for all possible choices of the set x_n.

Kernel function

Some commonly used Kernels are:

- Linear Kernel: $K(x, x') =< x, x' >$
- Polynomial Kernel: $K(x, x') =< x, x' >^d$ where d is the degree of the polynomial
- Gaussian RBF: $K(x, x') = e^{-\frac{\|x-x'\|}{2\sigma^2}}$

Gaussian Kernel

Gaussian kernel form

$$k(x, x') = e^{-\frac{\|x-x'\|}{2\sigma^2}}$$

We can expand the square

$$\|x - x'\| = x^T x + (x')^T (x') - 2x^T x' \tag{13}$$

The Gaussian kernel is not restricted to the use of Euclidean distance. If we use kernel substitution to replace $x^T x$ with a nonlinear kernel $k(x, x')$.