Machine Learning

Support Vector Machines

Fabio Vandin

November 27, 2017
Kernel Trick for SVM

What if we want to apply a nonlinear transformation before using SVM?

Let $\psi()$ be the nonlinear transformation

Considering the dual formulation $\Rightarrow$ we only need to be able to compute $\langle \psi(x), \psi(x') \rangle$ for some $x, x'$.

Definition

A kernel function is a function of the type:

$$K_\psi(x, x') = \langle \psi(x), \psi(x') \rangle$$

where $\psi(x)$ is a transformation of $x$.

Intuition: we can think of $K_\psi$ as specifying similarity between instances and of $\psi$ as mapping the domain set $\mathcal{X}$ into a space where these similarities are realized as inner products.
Some Kernels

The following are the most commonly used kernels

- linear kernel: $\psi(x) = x$
- sigmoid: $K(x, x') = \tanh(\gamma \langle x, x' \rangle + \zeta)$ (for $\gamma, \zeta > 0$)
- degree-$Q$ polynomial kernel
- Gaussian-radial basis function (RBF) kernel
Choice of Kernel

Notes

- polynomial kernel: usually used with $Q \leq 10$
- Gaussian-RBF kernel: usually $\gamma \in [0, 1]$
- many other choices are possible!

Mercer’s condition

$K(x, x')$ is a valid kernel function if and only if the kernel matrix

$$K = \begin{bmatrix}
K(x_1, x_1) & K(x_1, x_2) & \cdots & K(x_1, x_m) \\
K(x_2, x_1) & K(x_2, x_2) & \cdots & K(x_2, x_m) \\
\vdots & \vdots & \ddots & \vdots \\
K(x_m, x_1) & K(x_m, x_2) & \cdots & K(x_m, x_m)
\end{bmatrix}$$

is always symmetric positive semi-definite for any given $x_1, x_2, \ldots, x_m$. 
SVMs can be also used for regression. The function to be minimized will be

$$\frac{\lambda}{2} \|w\|^2 + \sum_{i=1}^{m} V_\varepsilon(y_i - \langle x_i, w \rangle - b)$$

where

$$V_\varepsilon(r) = \begin{cases} 0 & \text{if } |r| < \varepsilon \\ |r| - \varepsilon & \text{otherwise} \end{cases}$$
One can prove that the solution has the form:

\[ w = \sum_{i=1}^{m} (\alpha_i^* - \alpha_i) x_i \]

and that the final model produced in output is

\[ h(x) = \sum_{i=1}^{m} (\alpha_i^* - \alpha_i) \langle x_i, x \rangle + b \]

where \( \alpha_i^*, \alpha_i \geq 0 \) and are the solution to a suitable QP.

**Definition**

*Support vector:* \( x_i \) such that \( \alpha_i^* - \alpha_i \neq 0 \)

One can define kernels, similarly to SVM for classification.
Bibliography [UML]

SVM: Chapter 15
- no sections 15.1.2, 15.2.1, 15.2.2, 15.2.3,

Kernels: Chapter 16
- no section 16.3