

Lecture Notes on Machine Learning

February 27, 2014

We will first introduce how to apply kernel trick on SVM. Suppose we map a set of observations $x_i \in \mathbb{R}^p$ to $X_i = \phi(x_i)$, where X_i are in a Hilbert space H . Then, kernel SVM is to find a separator with maximum margin in H :

$$G(X) = \langle U, X \rangle_H + b$$

Let $\{X_i\}$ be the training set mapped into Hilbert space H , and $\{y_i\}$ be the corresponding label set. SVM requires that:

$$\begin{cases} G(X_i) \geq 1, & \text{when } y_i = 1 \\ G(X_i) \leq -1, & \text{when } y_i = -1 \end{cases}$$

Introduce $\xi_i = \max(0, 1 - y_i G(X_i))$, then we need to solve the following minimization problem:

$$\min \left\{ \frac{1}{2} \|U\|^2 + c \sum \xi_i \right\}$$

with constraints

$$\begin{cases} \xi_i \geq 0 \\ \xi_i - (1 - y_i G(x_i)) \geq 0 \end{cases}$$

There is an important theorem saying that the separator U will be the linear combination of X_i , i.e. $U = \sum \lambda_i X_i$, therefore, our separator can be written as:

$$L(x) = \sum \lambda_i k(x_i, x) + b$$

which is obviously a nonlinear classifier. Support vectors are those x_j with $\lambda_i \neq 0$.

Next, we move to the topic “k-fold cross-validation”. It works as follows: given a data set, we separate it into k equal parts randomly. Test on i th part and train on the others, we can do this k times. Each time, we can get the good proportion P_i^+ , P_i^- , which are the proportions of correct classification into + and - classes. Then find the average P^+ , P^- to evaluate the performance of the classify method.

One of the most famous cross-validation method is “leave one-out”, which is: if the data set is small, each time we take one observation out as test set, and the others compose training set to build classifier. We can do this many times (same as the number of observations), and take the average of good proportions to evaluate the performance.

(For the example on biomedical data, I really did not understand well, so I ignore this part...)