Machine Learning: THEORY AND APPLICATIONS

ARNAK DALALYAN (14/04/2016)

LECTURE 4: support vector machines (SVM)

1) CONVEXIFICATION

Let us focus here on the binary classification problem: we observe $\underbrace{X_1, \dots, X_n}_{\text{features}}$ and $\underbrace{Y_1, \dots, Y_n}_{\text{labels}}$ with Y_i taking only two values ± 1.

The goal is to find a prediction rule

$$g : \mathcal{X} \to \mathcal{Y} = \{\pm 1\}$$

such that the expected classification error

$$R(g) = \mathbb{E}\left[\mathbb{1}(Y \neq g(X))\right] = \mathbb{P}(Y \neq g(X))$$

is small. During the second lecture, we have seen that one can solve this problem by empirical risk minimization (ERM):

$$(1) \quad \widehat{g}_n \in \arg\min_{g \in \mathcal{G}} \widehat{R}_n(g) = \arg\min_{g \in \mathcal{G}} \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}(Y_i = g(X_i)).$$

Here $\mathcal{G}$ is a set of functions mapping $\mathcal{X}$ to $\{\pm 1\}$.

Let us stress that (1) is a non-convex problem and the non-convexity has two origins:

origine 1: the set $\mathcal{G}$ is not convex. Indeed, if $g_1 \neq g_2$, g_1 and $g_2 \in \mathcal{G}$, then $(g_1 + g_2)/2 \notin \mathcal{G}$ since it takes the values $\{-1, 0, 1\}$.

origine 2: the mapping $g \mapsto \widehat{R}_n(g)$ is nonconvex.

We will now fix these two problems in order to get a problem of convex optimisation.

First step: search space convexification

The set $\mathcal{F} \triangleq \{ f : \mathcal{X} \to \{\pm 1\} \}$ being non-convex, we replace it by its convex hull:

$$\mathcal{H} = \{ h : \mathcal{X} \longrightarrow [-1; +1] \}$$

It is clear that if $h_1, h_2 \in \mathcal{H}$, then $\frac{h_1 + h_2}{2} \in \mathcal{H}$. (more generally, $h_1, \dots, h_k \in \mathcal{H}$ and $\alpha_1, \dots, \alpha_k > 0$ imply that $\frac{\alpha_1 \cdot h_1 + \dots + \alpha_k \cdot h_k}{\alpha_1 + \dots + \alpha_k} \in \mathcal{H}$.)

Well, $\mathcal{H}$ is convex, but can we interpret an element $h \in \mathcal{H}$ as a predictor? Yes, we can define the followin rule:

- for $x \in \mathcal{X}$
  - predict $+1$ if $h(x) \geq 0$
  - predict -1 if $h(x) < 0$

This corresponds to defining the prediction function

$$g(x) = \operatorname{sign}(h(x))$$

along with the convention that $\operatorname{sign}(0) = +1$.

Second step: cost function convexification

Note first that if $g(x) = \operatorname{sign}(h(x))$ and $y \in \{\pm 1\}$ then

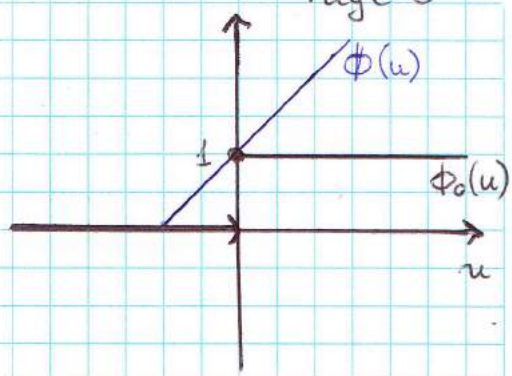
$$\mathbb{1}(y \neq g(x)) = \mathbb{1}(1 \neq y \cdot g(x)) = \mathbb{1}(-y \cdot g(x) \geq 0)$$
$$= \mathbb{1}(-y \cdot h(x) \geq 0)$$

Therefore, the empirical risk of $g = \operatorname{sign}(h)$ is

$$\widehat{R}_n(g) = \frac{1}{n} \sum_{i=1}^{n} \phi_0(-Y_i h(X_i))$$

where $\phi_0(u) = \mathbb{1}(u \geq 0)$. Clearly, this function ϕ_0 is not convex.

In order to get a convex cost function, we replace ϕ_0 by a convex surrogate ϕ.



The most common choices of ϕ are

- hinge loss : $\phi(u) = (1+u)_+$
- logistic loss : $\phi(u) = \log(1+e^u)/\log 2$
- exponential loss: $\phi(u) = e^u$
- quadratic loss: $\phi(u) = (1+u)^2$

THEOREM

If $\mathcal{H}_0$ is a convex subset of $\mathcal{H} = \{h : \mathcal{X} \to [-1;+1]\}$ and ϕ is one of the above convex surrogates, then the predictor

$$\hat{h} \in \arg\min_{h \in \mathcal{H}_0} \frac{1}{n} \sum_{i=1}^{n} \phi(-Y_i h(X_i))$$

can be computed by convex optimisation.

Furthermore, the solution of the problem

$$h^* \in \arg\min_{h \in \mathcal{H}} \mathbb{E}\left[\phi(-Y h(X))\right]$$

coincides with the Bayes predictor $\text{sign}(h^*) = g^*$.

(That is $h^*(x) \geq 0$ iff $\mathbb{P}(Y=1 \mid X=x) \geq 1/2$)

PROOF: omitted.

2 Kernel function and SVM

Let us consider a functional space, $\mathcal{H}_0 \subset \mathcal{H}$, on which a scalar product is defined:

$\forall\ h, h' \in \mathcal{H}$ $\quad \langle h, h' \rangle$ is the scalar product $\in \mathbb{R}$.

Then, this scalar product defines a norm:

$$\| h \| = \sqrt{\langle h, h \rangle}$$

Examples

1) $\mathcal{H}_0 = \{ h : \mathbb{R}^p \to \mathbb{R} \quad h$ is an affine function, i.e. $h(x) = a \cdot x + b \quad \forall x \in \mathbb{R}^p \}$

Here $a = (a_1, \dots, a_p) \in \mathbb{R}^p$. The scalar product can be defined as $\langle h, \bar{h} \rangle = b \cdot \bar{b} + a \cdot \bar{a} = b\bar{b} + \sum_{j=1}^{p} a_j \bar{a}_j$.

2) $\mathcal{H}_0 = \{ h : \mathbb{R}^p \to \mathbb{R} \quad h$ is differentiable and $h' \in L_2 \}$

$\langle h, \bar{h} \rangle = \int h \cdot \bar{h} + \int h' \cdot \bar{h}'$

Then $\|h\|^2 = \int h^2 + \int (h')^2$

If $\mathcal{H}_0$ is endowed with a scalar product, then the corresponding norm satisfies the triangle inequality

$$\| h + \bar{h} \| \leq \|h\| + \|\bar{h}\|$$

which implies that the function

$$h \longmapsto \|h\|$$

is convex. Therefore, the set $\mathcal{H}_t = \{ h \in \mathcal{H}_0 : \|h\| \leq t \}$ is a convex subset of $\mathcal{H}$.

One can therefore define the predictor

$$\text{(P1)} \qquad \hat{h}_t \in \arg\min_{\|h\| \leq t} \frac{1}{n} \sum_{i=1}^{n} \phi(-Y_i h(X_i))$$

or, using the Lagrange multipliers,

$$\text{(P2)} \qquad \hat{h}_\lambda \in \arg\min_{h \in \mathcal{H}_0} \left\{ \frac{1}{n} \sum_{i=1}^{n} \phi(-Y_i h(X_i)) + \lambda \|h\|^2 \right\}$$

Here, t and λ are >0 tuning parameters. If t is large (λ is small) then the set $\mathcal{H}_t$ is large and $\hat{h}_t$ may overfit. If t is small (λ is large) then $\mathcal{H}_t$ is too small and $\hat{h}_t$ (resp. $\hat{h}_\lambda$) will underfit. One can choose t and λ by cross-validation but how to choose $\mathcal{H}_0$?

The SVM corresponds to (P2) with a set $\mathcal{H}_0$ defined through a kernel function.

Let $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a function such that

a) $K(x, \bar{x}) = K(\bar{x}, x)$

b) K is semi-definite positive that is

$$\sum_{i,j=1}^{m} \alpha_i \alpha_j K(x_i, x_j) \geq 0 \quad \forall \alpha_1, \dots, \alpha_m \in \mathbb{R}$$
$$\forall x_1, \dots, x_m \in \mathcal{X}$$

We say that K is a kernel and define the set

$$\mathcal{H}_0 = \left\{ \sum_{j=1}^{m} \alpha_j K(x_j, \cdot) : m \in \mathbb{N}, \alpha_1, \dots, \alpha_m \in \mathbb{R}, x_1, \dots, x_m \in \mathcal{X} \right\}$$

The set $\mathcal{H}_0$ is convex, we can define a scalar product on this set $\mathcal{H}_0$ by

$$\langle h, \bar{h} \rangle = \sum_{i,j} \alpha_i \bar{\alpha}_j K(x_i, x_j)$$

$$\text{if} \quad h(x) = \sum_{i=1}^{n} \alpha_i K(x_i, x) \text{ and } \bar{h}(x) = \sum_{j=1}^{m} \bar{\alpha}_j K(x_j, x)$$

The function K measures the similarity between x and x'.
The set $\mathcal{H}_0$ described above is called reproducing kernel Hilbert space. It is very convenient to use an RKHS as $\mathcal{H}_0$ in (P2) because of the following theorem:

If $\mathcal{H}_0$ is the RKHS induced by K and $\hat{h}_\lambda$ is a solution of (P2), then there are $\alpha_1, \dots, \alpha_n \in \mathbb{R}$ such that $$\hat{h}_\lambda(x) = \sum_{i=1}^{n} \alpha_i K(X_i, x).$$

This means that the SVM predictor $\hat{h}_\lambda$ is a linear combination of the terms $\{K(X_i, \cdot) : i = 1, \dots, n\}$ corresponding to the sample $X_1, \dots, X_n$. So, we first map each $X_i \in \mathcal{X}$ to a $K(X_i, \cdot) \in \mathcal{H}_0$ and then apply a linear classifier.