# Machine Learning and Applications

## Assignment 1, solution

1. (3) Soft margin hyperplanes. The function of the slack variables used in the optimmization problem for soft margin hyperplanes has the form: $\xi \to \sum_{i=1}^{m} \xi_i$. Instead, we could use $\xi \to \sum_{i=1}^{m} \xi_i^p$, with $p > 1$.

   (a) Give the dual formulation of the problem in this general case.
   **Solution:**

   *Proof.*

$$\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m} \xi_i^p$$
$$\text{s.t. } y_i(w \cdot x_i + b) \geq 1 - \xi_i$$
$$\xi_i \geq 0, i \in [1, m]$$

   **Lagrangian** for the problem: $\forall w, b, \alpha_i \geq 0, \beta_i \geq 0$:

$$L(w, b, \xi, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m} \xi_i^p - \sum_{i=1}^{m} \alpha_i[y_i(w \cdot x_i + b) - 1 + \xi_i] - \sum_{i=1}^{m} \beta_i \xi_i$$

   **KKT conditions**:

$$\nabla_w L = w - \sum \alpha_i y_i x_i = 0 \quad \Leftrightarrow \quad w = \sum_i \alpha_i y_i x_i$$

$$\nabla_b L = -\sum_i \alpha_i y_i = 0 \quad \Leftrightarrow \quad \sum_i \alpha_i y_i = 0$$

$$\nabla_{\xi_i} L = Cp\xi_i^{p-1} - \alpha_i - \beta_i = 0 \quad \Leftrightarrow \quad \xi_i^{p-1} = \frac{\alpha_i + \beta_i}{Cp}$$

   $\forall i, \alpha_i[y_i(w \cdot x_i + b) - 1 + \xi_i] = 0 \Rightarrow \alpha_i = 0$ or $y_i(w \cdot x_i + b) = 1 - \xi_i$

   **Dual optimization problem:** plugging optimal $w$ and $\xi$ in $L$:

$$L = \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) + \sum_i \left(\frac{\alpha_i + \beta_i}{Cp}\right)^{\frac{1}{p-1}} (\alpha_i + \beta_i)\left(\frac{1}{p} - 1\right)$$

Dual problem

$$\max_{\boldsymbol{\alpha},\boldsymbol{\beta}} L$$

$$\text{s.t. } \sum_i \alpha_i y_i = 0, \quad \alpha_i \geq 0, \quad \beta_i \geq 0 \quad \forall i$$

■

(b) How does this more general formulation ($p > 1$) compare to the standard setting ($p = 1$)? In the case $p = 2$ is the optimization still convex?

*Proof.* $p > 1$ imposes **larger loss** for points that violate the margin → more noise-sensitive. The problem is still **convex**. ■

Sparse SVM. One can give two types of arguments in favor of the SVM algorithm: one based on the sparsity of the support vectors, another based on the notion of margin. Suppose that instead of maximizing the margin, we choose instead to maximize sparsity by minimizing the $L_p$ norm of the vector $\boldsymbol{\alpha}$ that defines the weight vector $\mathbf{w}$, for some $p \geq 1$. First, consider the case $p = 2$. This gives the following optimization problem:

$$\min_{\boldsymbol{\alpha},b} \frac{1}{2} \sum_{i=1}^{m} \alpha_i^2 + C \sum_{i=1}^{m} \xi_i \tag{1}$$

$$\text{subject to } y_i \left( \sum_{j=1}^{m} \alpha_j y_j \mathbf{x}_i \cdot \mathbf{x}_j + b \right) \geq 1 - \xi_i, i \in [1, m]$$

$$\xi_i, \alpha_i \geq 0, i \in [1, m].$$

(a) Show that modulo the non-negativity constraint on $\boldsymbol{\alpha}$, the problem coincides with an instance of the primal optimization problem of SVM.

*Proof.* **Primal SVM problem**:

$$\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C \sum_i \xi_i$$

$$\text{s.t. } y_i(w \cdot \phi(x_i) + b) \geq 1 - \xi, \quad \forall i$$

Let's consider $\phi(x) = \begin{pmatrix} y_1(x_1 \cdot x) \\ \cdots \\ y_m(x_m \cdot x) \end{pmatrix}$. Pluggin such $\phi(x)$ into primal optimization of SVM we obtain sparse SVM problem modulo non-negative constraint on $w$ ■

(b) Derive the dual optimization of problem of 1.

*Proof.* Lagrangian:

$$L = \frac{1}{2}\|\alpha\|^2 + C\sum_i \xi_i - \sum_i \alpha_i'[y_i(\alpha_i \cdot \phi(x_i) + b) - 1 + \xi_i] - \sum_i \beta_i'\xi_i - \sum_i \gamma_i\alpha_i$$

KKT conditions:

$$\nabla_w L = \alpha - \sum_i \alpha_i' y_i \phi(x_i) - \gamma' = 0$$

$$\nabla_b L = -\sum_i \alpha_i' y_i = 0$$

$$\nabla_{\xi_i} L = C - \alpha_i' - \beta_i' = 0$$

$$\alpha_i[y_i(\alpha \cdot \phi(x_i) + 1) - 1 + \xi_i] = 0$$

Plugging optimal $\alpha$ and $\xi$ we obtain dual optimization problem

$$\min_{\alpha_i', \gamma_i'} -\frac{1}{2}\left\|\sum \alpha_i' y_i \phi(x_i)\right\|^2 + \sum_i \alpha_i' + \frac{1}{2}\|\gamma'\|^2$$

$$\text{s.t.} \sum_i \alpha_i' y_i = 0$$

$$0 \le \alpha_i' \le C, \quad \gamma_i' \ge 0, \quad \forall i \in [1, m]$$

■

(c) Setting $p = 1$ will induce a more sparse $\boldsymbol{\alpha}$. Derive the dual optimization in this case

*Proof.* Let's set $p = 1$ in $L_p$ norm:

$$\max_{\alpha, b, \xi} \sum_i |\alpha_i| + C\sum_i \xi_i$$

$$\text{s.t. } y_i(\alpha \cdot \phi(x_i) + b) \ge 1 - \xi_i, \quad i \in [1, m]$$

$$\xi_i, \alpha_i \ge 0, \quad i \in [1, m]$$

Lagrangian:

$$L = \sum_i |alpha_i| + C\sum_i \xi_i - \sum_i \alpha_i'[y_i(\alpha_i \cdot \phi(x_i) + b) - 1 + \xi_i] - \sum_i \beta_i'\xi_i - \sum_i \gamma_i\alpha_i$$

KKT conditions:

$$\nabla_w L = 1 - \sum_i \alpha'_i y_i \phi(x_i) - \gamma' = 0$$

$$\nabla_b L = -\sum_i \alpha'_i y_i = 0$$

$$\nabla_{\xi_i} L = C - \alpha'_i - \beta'_i = 0$$

$$\alpha_i [y_i(\alpha \cdot \phi(x_i) + 1) - 1 + \xi_i] = 0$$

Dual optimization:

$$\max_{\alpha', \gamma'} \sum_i \alpha'_i$$

$$\text{s.t. } \sum_i \alpha'_i y_i = 0$$

$$0 \leq \alpha'_i \leq C, \quad \gamma'_i \geq 0, \quad i \in [1, m]$$

■

2. (2) Importance weighted SVM. Suppose you wish to use SVMs to solve a learning problem where some training data points are more important than others. More formally, assume that each training point consists of a triplet $(x_i, y_i, p_i)$, where $0 \leq p_i \leq 1$ is the importance of the $i$-th point. Rewrite the primal SVM constrained optimization problem so that the penalty for mis-labeling a point $x_i$ is scaled by the priority $p_i$ . Then carry this modification through the derivation of the dual solution.

*Proof.*

$$\min_{w,b,\xi} \|w\|^2 + C \sum_i p_i \xi_i$$

$$\text{s.t. } y_i(w \cdot x_i + b) \geq 1 - \xi_i, \quad i \in [1, m]$$

$$\xi_i \geq 0, \quad i \in [1, m]$$

Lagrangian:

$$L = \frac{1}{2}\|w\|^2 + C \sum_i p_i \xi_i - \sum_i \alpha_i [y_i(w \cdot x_i + b) - 1 + \xi_i] - \sum_i \beta_i \xi_i$$

KKT conditions:

$$\nabla_w L = w - \sum \alpha_i y_i x_i = 0$$
$$\nabla_b L = -\sum_i \alpha_i y_i = 0$$
$$\nabla_{\xi_i} L = Cp_i - \alpha_i - \beta_i = 0$$
$$\forall i, \alpha_i[y_i(w \cdot x_i + b) - 1 + \xi_i] = 0$$

Dual optimization problem:

$$\max_{\alpha,\beta} -\frac{1}{2}\left\|\sum_i \alpha_i y_i x_i\right\|^2 + \sum_i \alpha_i$$
$$\text{s.t. } \sum_i \alpha_i y_i = 0, \quad i \in [1, m]$$
$$0 \leq \alpha_i \leq Cp_i, \quad i \in [1, m]$$

■

3. (3) Show that the following kernels $K$ are PDS:

   (a) $K(x, y) = \cos(x - y)$ over $\mathbb{R} \times \mathbb{R}$.

   *Proof.*

   $K(x, y) = \cos(x-y) = \sin(x)\sin(y)+\cos(x)\cos(y) = \langle \Phi(x), \Phi(y) \rangle$,

   where $\Phi(x) = (\sin(x), \cos(x))$. $K(x, y)$ is a dot product, therefore, it is a PDS. ■

   (b) $K(x, y) = (x + y)^{-1}$ over $(0, +\infty) \times (0, +\infty)$.

   *Proof.* If $\phi(x, y)$ is an NDS and $\phi(x, y) > 0 \quad \forall x, y$, then $\frac{1}{\phi(x,y)}$ is a PDS, because:

   $$\frac{1}{\phi(x, y)} = \int_0^\infty e^{-t\phi(x,y)} dt,$$

   and $e^{-t\phi(x,y)}$ is a PDS $\forall t > 0$.

   $\phi(x, y) = x + y$ is an NDS, therefore, $\frac{1}{\phi(x,y)}$ is a PDS. ■

(c) $\forall \lambda > 0, K(x,y) = \exp(-\lambda[\sin(y-x)]^2)$ over $\mathbb{R} \times \mathbb{R}$ (*Hint:* rewrite $[\sin(y-x)]^2$) as the square of the norm of the difference of two vectors.)

*Proof.*

$$[\sin(y-x)]^2 = 1-[\cos(y-x)]^2 = 1-(\cos(x)\cos(y)+\sin(x)\sin(y))^2 = 1-\langle \Phi(x), \Phi(y)\rangle,$$

where $\Phi(x) = \begin{pmatrix}\cos(x)\\ \sin(x)\end{pmatrix}$. Therefore, $K(x,y) \sim \exp(-\langle \Phi(x), \Phi(y)\rangle)$ and is a PDS kernel. ■

4. (2) Show that the following kernels K are NDS:

(a) $K(x,y) = [\sin(x-y)]^2$ over $\mathbb{R} \times \mathbb{R}$.

*Proof.*

$$[\sin(y-x)]^2 = 1-[\cos(y-x)]^2 = 1-(\cos(x)\cos(y)+\sin(x)\sin(y))^2 = 1-\langle \Phi(x), \Phi(y)\rangle$$

By definition, $K(x,y)$ is an NDS if $\forall c_i \in \mathbb{R}, \sum_i c_i = 0$ and $\forall x_i \in \mathcal{X} \rightarrow$

$$\sum_{i,j=1}^{m} c_i c_j K(x_i, x_j) \leq 0$$

Then we have

$$\sum_{i,j=1}^{m} c_i c_j K(x_i, x_j) \leq 0 = \tag{2}$$

$$= \sum_{i,j} c_i c_j - \sum_{i,j} c_i c_j \langle \Phi(x_i), \Phi(x_j)\rangle = \tag{3}$$

$$= -\sum_{i,j} c_i c_j \langle \Phi(x_i), \Phi(x_j)\rangle \leq 0. \tag{4}$$

■

(b) $K(x,y) = \log(x+y)$ over $(0,+\infty) \times (0,+\infty)$.

*Proof.* $\log(x+y) = \log(\phi(x,y))$ where $\phi(x,y) = x+y$ is an NDS. Let us use the following expression:

$$\log(1+\phi(x,y)) = \int_0^\infty (1-e^{-t\phi(x,y)})\frac{e^{-t}}{t}dt$$

If $\phi(x, y)$ is an NDS, then the above expression is an NDS.

$$\log(\phi + 1/c) = \log(1 + c\phi) - \log(c).$$

Taking limit $c \to \infty$ we obtain that $\log(\phi)$ is an NDS. ■

5. (2) Explicit polynomial kernel mapping. Let $K$ be a polynomial kernel of degree $d$, i.e., $K : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}, K(\mathbf{x}, \mathbf{x}') = (\mathbf{x} \cdot \mathbf{x}' + c)^d$, with $c > 0$. Show that the dimension of the feature space associated to $K$ is

$$\binom{N+d}{d}$$

Write $K$ in terms of kernels $k_i : (\mathbf{x}, \mathbf{x}') \to (\mathbf{x} \cdot \mathbf{x}')^i, i \in [0, d]$. What is the weight assigned to each $k_i$ in that expression? How does it vary as a function of $c$?

*Proof.* A kernel should contain all monomials $x_1^{k_1}, \ldots, x_N^{k_N}, \quad \sum_j k_j \leq d$. So, we need to calculate the number of such monomials $f(d, N)$. For $f(d, N)$ we have the following recursive expression:

$$f(d, N) = f(d, N-1) + f(d-1, N),$$

i.e. we first fix the power of $N$-th component to be 0 ($f(d, N-1)$) and then add all remaining monomials for which the power of $N$-th component is greater than 0 (so, the sum of powers is not greater than $d-1$, $f(d-1, N)$.

By induction:

- $N = 0, d = 1; f(d, N) = 1$
- $N = 1, d = 0; f(d, N) = 1$
- Induction step:

$$f(d, N) = \binom{N-1+d}{d} + \binom{N+d-1}{d-1} = \binom{N+d}{d}$$

Using Newton's formula:

$$K(x, y) = \sum_{i=0}^{d} \binom{d}{i} c^{d-i} (x \cdot y)^i$$

The weight assigned to $k_i$ is $\binom{d}{i} c^{d-i}$ and it is increasing with $c$. ■