Discussion 8

王欣奕，wangxy6@shanghaitech.edu.cn

Review

- Kernels methods
- Support Vector Machines

Kernels methods

- **A kernel K is a legal def of dot-product:**

$$K(\boldsymbol{x}, \boldsymbol{y}) = <\phi(\boldsymbol{x}), \phi(\boldsymbol{y})>$$

- ϕ **is an implicit mapping.**

Why we need it?

- Many algorithms interact with data only via dot-products
- Useful when data is not linearly separable in the original space
- We only care about K, not ϕ.

Kernels methods

- Conceptually, to apply kernel function, we
  1. find $\phi(\cdot)$ and calculate $\phi(\mathbf{x})$,
  2. measure the similarity by some kernel $k(\mathbf{x}, \mathbf{x}')$.

  However, $\mathbf{x}$ is typically mapped to a high-dimensional space by $\phi(\cdot)$ so that the computational complexity for inner product is huge.
- In practice, we can therefore work directly in terms of kernels, which allows us implicitly to use feature spaces of high, even infinite dimensionality.

Kernels methods

Theorem (Mercer)

K is a kernel if and only if:

- K is symmetric
- For any set of training points $x_1, x_2, \dots, x_m$ and for any $a_1, a_2, \dots, a_m \in R$, we have:

$$\sum_{i,j} a_i a_j K(x_i, x_j) \geq 0$$

$$a^T K a \geq 0$$

I.e., $K = (K(x_i, x_j))_{i,j=1,\dots,n}$ is positive semi-definite.

Kernels methods

- **Properties:**

Given valid kernels $k_1(\mathbf{x}, \mathbf{x}')$, $k_2(\mathbf{x}, \mathbf{x}')$, the following new kernels will also valid:

$$k(\mathbf{x}, \mathbf{x}') = ck_1(\mathbf{x}, \mathbf{x}') \tag{1}$$
$$k(\mathbf{x}, \mathbf{x}') = f(\mathbf{x})k_1(\mathbf{x}, \mathbf{x}') f(\mathbf{x}') \tag{2}$$
$$k(\mathbf{x}, \mathbf{x}') = q(k_1(\mathbf{x}, \mathbf{x}')) \tag{3}$$
$$k(\mathbf{x}, \mathbf{x}') = \exp(k_1(\mathbf{x}, \mathbf{x}')) \tag{4}$$
$$k(\mathbf{x}, \mathbf{x}') = k_1(\mathbf{x}, \mathbf{x}') + k_2(\mathbf{x}, \mathbf{x}') \tag{5}$$
$$k(\mathbf{x}, \mathbf{x}') = k_1(\mathbf{x}, \mathbf{x}') k_2(\mathbf{x}, \mathbf{x}') \tag{6}$$
$$k(\mathbf{x}, \mathbf{x}') = k_3(\phi(\mathbf{x}), \phi(\mathbf{x}')) \tag{7}$$
$$k(\mathbf{x}, \mathbf{x}') = \mathbf{x}^{\mathrm{T}}\mathbf{A}\mathbf{x}' \tag{8}$$

where $c > 0$ is a constant, $f(\cdot)$ is any function, $q(\cdot)$ is a polynomial with nonnegative coefficients, $k_3(\cdot)$ is a valid kernel and $\mathbf{A}$ is a symmetric positive semidefinite matrix.

Kernels methods

- **Proof for (1) and (5): K(x,x') = $\sum_{i=1}^{m} a_i k_i(x, x')$ is a valid kernel.**

- **Symmetry:**

K(x',x) = $\sum_{i=1}^{m} a_i k_i(x', x)$. Since k_i is a valid kernel function, $k_i(x, x') = k_i(x', x)$.
Then K(x',x)= $\sum_{i=1}^{m} a_i k_i(x', x) = \sum_{i=1}^{m} a_i k_i(x, x')$ = K(x,x').

- **Positive Semidefinite:**

Let $u \in R^n$. Let G be the Gram matrix of K, i.e.
$G_{i,j}$ = K(x_i, x_j) = $\sum_{i=1}^{m} a_i k_i\left(x_i, x_j\right)$. We can get G = $\sum_{i=1}^{m} a_i G_i$. G_i is the Gram matrix of k_i . $u^T G u = u^T (\sum_{i=1}^{m} G_i) u = \sum_{i=1}^{m} a_i u^T G_i\, u \geq 0$.

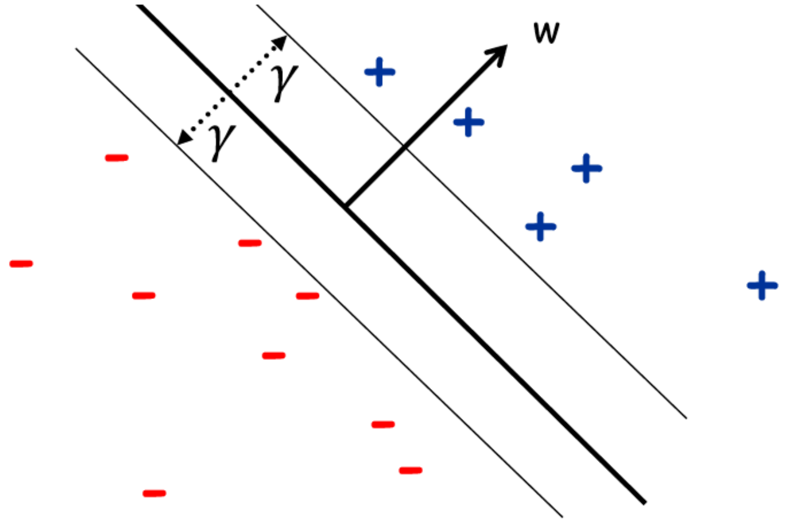
Support vector machines

Definition: The margin of example x w.r.t. a linear sep. w is the distance from x to the plane $w \cdot x = 0$.

Definition: The margin γ_w of a set of examples S wrt a linear separator w is the smallest margin over points $x \in S$.

Definition: The margin γ of a set of examples S is the maximum γ_w over all linear separators w.

$$\gamma = \boxed{\max_{w} \boxed{\min_{x \in S} \boxed{\frac{y w^T x}{||w||}}}}$$



Support vector machines

- Optimize for the maximum margin Separator.

Input: S={(x_1, y_1), …,(x_m, y_m)};

Maximize γ under the constraint:

- $||w||^2 = 1$ Non-linear, non-convex
- For all i, $y_i w \cdot x_i \geq \gamma$

$$w' = \frac{w}{\gamma}$$

Input: S={(x_1, y_1), …,(x_m, y_m)};

Minimize $||w'||^2$ under the constraint:

- For all i, $y_i w' \cdot x_i \geq 1$

- The objective is convex (quadratic)
- All constraints are linear
- Can solve efficiently (in poly time) using standard **quadratic programing** (QP) software

Support vector machines(with noise)

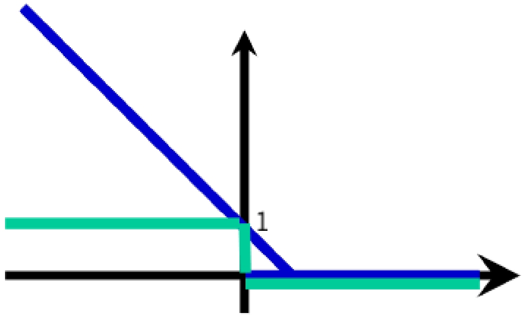
Input: S={(x_1, y_1), …,$(\mathrm{x}_\mathrm{m}, y_\mathrm{m})$};

Find $\mathrm{argmin}_{\mathrm{w},\xi_1,\ldots,\xi_m} ||w||^2 + C\sum_i \xi_i$ s.t.:

- For all i, $y_i w \cdot x_i \geq 1 - \xi_i$

  $\xi_i \geq 0$

Replace the number of mistakes with the hinge loss



$l(w, x, y) = \max(0, 1 - y\, w \cdot x)$

Consider the following, which we'll call the **primal** optimization problem:

$$\begin{aligned} \min_w \quad & f(w) \\ \text{s.t.} \quad & g_i(w) \leq 0, \quad i = 1, \ldots, k \\ & h_i(w) = 0, \quad i = 1, \ldots, l. \end{aligned}$$

To solve it, we start by defining the **generalized Lagrangian**

$$\mathcal{L}(w, \alpha, \beta) = f(w) + \sum_{i=1}^{k} \alpha_i g_i(w) + \sum_{i=1}^{l} \beta_i h_i(w).$$

- primal

$$\min_{w} \max_{\alpha,\beta\,:\,\alpha_i \geq 0} \mathcal{L}(w, \alpha, \beta),$$

- dual

$$\max_{\alpha,\beta\,:\,\alpha_i \geq 0} \min_{w} \mathcal{L}(w, \alpha, \beta).$$

$$d^* = \max_{\alpha,\beta\,:\,\alpha_i \geq 0} \min_{w} \mathcal{L}(w, \alpha, \beta) \leq \min_{w} \max_{\alpha,\beta\,:\,\alpha_i \geq 0} \mathcal{L}(w, \alpha, \beta) = p^*.$$

$d^* = p^*$ when strong duality holds(KKT conditions)

Karush-Kuhn-Tucker (KKT) conditions, which are as follows:

$$\frac{\partial}{\partial w_i}\mathcal{L}(w^*, \alpha^*, \beta^*) \;=\; 0, \;\; i = 1, \ldots, n$$

$$h_i(w^*) = 0, \;\; i = 1, \ldots, l.$$

$$\begin{aligned} \alpha_i^* g_i(w^*) &= 0, \;\; i = 1, \ldots, k \\ g_i(w^*) &\leq 0, \;\; i = 1, \ldots, k \\ \alpha^* &\geq 0, \;\; i = 1, \ldots, k \end{aligned}$$

Support vector machines(No-noise)

$$L(w,a) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{m} a_i(y_i w^T x_i - 1)$$

We need to solve : $\max_{a_i} \min_{w} L(w,a)$.

$$\nabla_w L(w,a) = w - \sum_{i=1}^{m} a_i y_i x_i = 0$$

$$w = \sum_{i=1}^{m} a_i y_i x_i$$

Plug this into $L(w,a)$ we can get the dual problem:

$$\min_{a_i} \frac{1}{2}\sum_{i,j=1}^{m} a_i a_j y_i y_j x_i \cdot x_j + \sum_{i=1}^{m} a_i, \quad s.t.\ a_i \geq 0$$

Support vector machines(with-noise)

$$L(w,\xi,a,\lambda) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m}\xi_i - \sum_{i=1}^{m}a_i(y_i w^T x_i - 1 + \xi_i) - \sum_{i=1}^{m}\lambda_i\xi_i$$

We need to solve : $\max_{a_i,\lambda_i}\min_{w,\xi} L(w,\xi,a,\lambda)$.

$$\nabla_w L(w,\xi,a,\lambda) = w - \sum_{i=1}^{m}a_i y_i x_i = 0 \qquad \nabla_{\xi_i} L(w,\xi,a,\lambda) = C - a_i - \lambda_i = 0$$

$$w = \sum_{i=1}^{m}a_i y_i x_i \qquad C = a_i + \lambda_i$$

Plug these into $L(w,\xi,a,\lambda)$ we can get the dual problem:

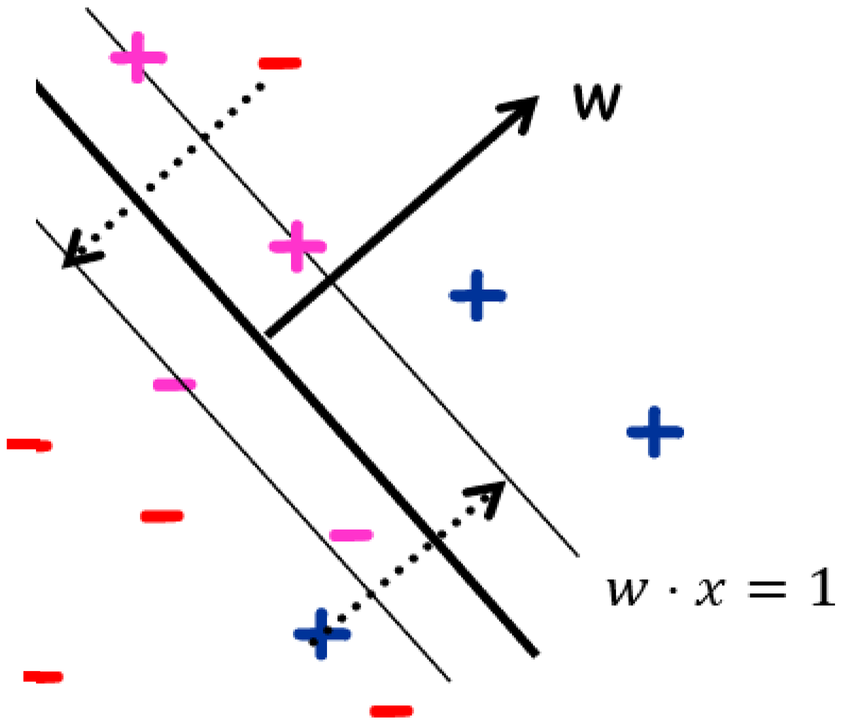
$$\min_{a_i}\frac{1}{2}\sum_{i,j=1}^{m}a_i a_j y_i y_j x_i \cdot x_j + \sum_{i=1}^{m}a_i, \quad s.t.\ 0 \le a_i \le \mathrm{C}$$

Support vector machines(with-noise)

$y_i w^T x_i > 1 \rightarrow a_i = 0$

$y_i w^T x_i < 1 \rightarrow a_i = C$

$y_i w^T x_i = 1 \rightarrow a_i \in (0,C)$



Support vector machines(Kernel)

$$\min_{a_i} \frac{1}{2} \sum_{i,j=1}^{m} a_i a_j y_i y_j \boxed{x_i \cdot x_j} + \sum_{i=1}^{m} a_i, \quad s.t.\ 0 \le a_i \le C$$

$$K(x_i, x_j) = <\phi(x_i), \phi(x_j)>$$