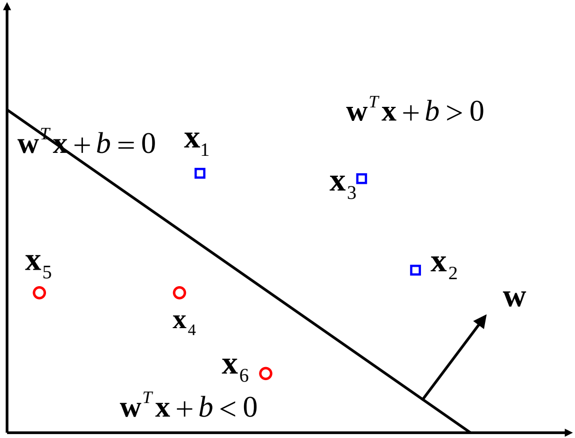
Support Vector Machines

Mingmin Chi

SCS Fudan University, Shanghai, China

Outline

Decision Boundary of Perceptron



Convergence of the Perceptron

- Suppose that there exists the optimal solution $(\mathbf{w}^*, b^*)$, which defines a decision boundary correctly classifying all the training samples, and every training sample is at least distance $\rho > 0$ from the decision boundary, i.e.,

Convergence of the Perceptron

- Suppose that there exists the optimal solution $(\mathbf{w}^*, b^*)$, which defines a decision boundary correctly classifying all the training samples, and every training sample is at least distance $\rho > 0$ from the decision boundary, i.e.,

$$|f(\mathbf{x}_i)| = |\mathbf{w}^{*\top}\mathbf{x}_i + b^*| = \rho$$

- Suppose that there exists a $\rho > 0$, and a weight vector $\mathbf{w}^*$ satisfying $||\mathbf{w}^*|| = 1$, and a threshold b^*, such that

Convergence of the Perceptron

- Suppose that there exists the optimal solution $(\mathbf{w}^*, b^*)$, which defines a decision boundary correctly classifying all the training samples, and every training sample is at least distance $\rho > 0$ from the decision boundary, i.e.,

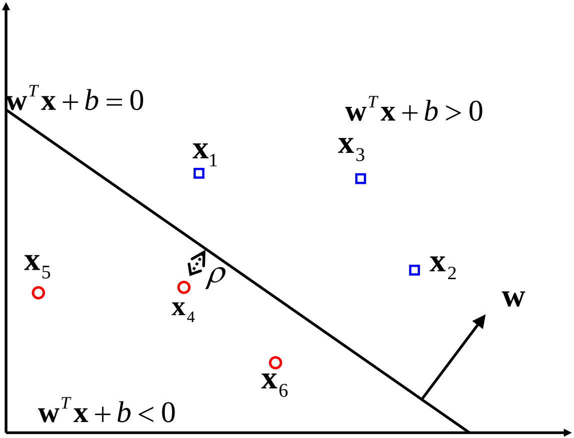
$$|f(\mathbf{x}_i)| = |\mathbf{w}^{*\top}\mathbf{x}_i + b^*| = \rho$$

- Suppose that there exists a $\rho > 0$, and a weight vector $\mathbf{w}^*$ satisfying $||\mathbf{w}^*|| = 1$, and a threshold b^*, such that

$$\forall_{i+1}^{n}, y_i f(\mathbf{x}_i) \geq \rho$$

- Then the perceptron algorithm converges after no more than $(b^{*2} + 1)(R^2 + 1)/\rho^2$ updates, where $R = \max_i ||\mathbf{x}_i||$. [Novikov, 1962]

Distance From Decision Boundary



Definition of Margin

- The quantity ρ plays a crucial role for the perceptron as it determines

Definition of Margin

- The quantity ρ plays a crucial role for the perceptron as it determines (1) how well the two classes can be separated;

Definition of Margin

- The quantity ρ plays a crucial role for the perceptron as it determines (1) how well the two classes can be separated; (2) How fast the perceptron learning algorithm converges
- This quantity ρ is henceforth what we call a *margin*

Definition of Margin

- The quantity ρ plays a crucial role for the perceptron as it determines (1) how well the two classes can be separated; (2) How fast the perceptron learning algorithm converges
- This quantity ρ is henceforth what we call a *margin*
- **Definition:** Denote by $f : \mathbb{R}^d \to \mathbb{R}$ used for the classification, then

$$\rho_f(\mathbf{x}, y) := \; yf(\mathbf{x})$$

Definition of Margin

- The quantity ρ plays a crucial role for the perceptron as it determines (1) how well the two classes can be separated; (2) How fast the perceptron learning algorithm converges
- This quantity ρ is henceforth what we call a *margin*
- **Definition:** Denote by $f : \mathbb{R}^d \to \mathbb{R}$ used for the classification, then

$$\rho_f(\mathbf{x}, y) := \ yf(\mathbf{x})$$

- Denote the minimum margin over the whole samples

$$\rho_f := \min_{1 \le i \le n} \rho_f(\mathbf{x}_i, y_i)$$

Maximum Margin Hyperplane

- It is desirable to have an estimator with a large margin
- Question?

Maximum Margin Hyperplane

- It is desirable to have an estimator with a large margin
- Question? Whether there exists such estimator with *maximum* margin,

Maximum Margin Hyperplane

- It is desirable to have an estimator with a large margin
- Question? Whether there exists such estimator with *maximum* margin, i.e., whether some f^* exists with

$$f^* := \arg\max_f \rho_f = \arg\max_f \min_i \rho_f(\mathbf{x}_i, y_i)$$

Maximum Margin Hyperplane

- It is desirable to have an estimator with a large margin
- Question? Whether there exists such estimator with *maximum* margin, i.e., whether some f^* exists with

$$f^* := \operatorname*{arg\,max}_f \rho_f = \operatorname*{arg\,max}_f \min_i \rho_f(\mathbf{x}_i, y_i)$$

- Without the constraints on the size of **w**, this maximum does not exist

Maximum Margin Hyperplane (Contd)

- If we define $f : \mathbb{R}^d \to \mathbb{R}$ by

$$f(\mathbf{x}) = \frac{\mathbf{w}^\top \mathbf{x} + b}{||\mathbf{w}||},$$

then the maximum margin f is defined by the weight vector and threshold that satisfy

$$\mathbf{w}^*, b^* = \arg\max_{\mathbf{w},b} \min_{i=1}^{m} \frac{y_i(\mathbf{w}^\top \mathbf{x}_i + b)}{||\mathbf{w}||} \tag{1}$$

Maximum Margin Hyperplane (Contd)

- If we define $f : \mathbb{R}^d \to \mathbb{R}$ by

$$f(\mathbf{x}) = \frac{\mathbf{w}^\top \mathbf{x} + b}{||\mathbf{w}||},$$

then the maximum margin f is defined by the weight vector and threshold that satisfy

$$\begin{aligned} \mathbf{w}^*, b^* &= \arg\max_{\mathbf{w},b} \min_{i=1}^{m} \frac{y_i(\mathbf{w}^\top \mathbf{x}_i + b)}{||\mathbf{w}||} \\ &= \arg\max_{\mathbf{w},b} \min_{i=1}^{m} y_i \mathrm{sgn}(\mathbf{w}^\top \mathbf{x}_i + b) \left\| \frac{\mathbf{w}^\top \mathbf{x}_i}{||\mathbf{w}||^2}\mathbf{w} + \frac{b}{||\mathbf{w}||^2}\mathbf{w} \right\| \end{aligned} \quad (1)$$

Maximum Margin Hyperplane (Cont)

- The Equ.(1) is equivalent to

$$\mathbf{w}^*, b^*, \rho^* = \arg\max_{\mathbf{w}, b, \rho} \rho \quad \text{s.t.} \quad \frac{y_i(\mathbf{w}^\top \mathbf{x}_i + b)}{||\mathbf{w}||} \geq \rho$$

Maximum Margin Hyperplane (Cont)

- The Equ.(1) is equivalent to

$$
\begin{aligned}
\mathbf{w}^*, b^*, \rho^* &= \arg\max_{\mathbf{w},b,\rho} \rho \quad \text{s.t.} \quad \frac{y_i(\mathbf{w}^\top \mathbf{x}_i + b)}{||\mathbf{w}||} \geq \rho \\
&= \arg\max_{\mathbf{w},b,\rho} \rho \quad \text{s.t.} \quad ||\mathbf{w}|| = 1 \text{ and } y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq \rho
\end{aligned}
$$

Maximum Margin Hyperplane (Cont)

- The Equ.(1) is equivalent to

$$\begin{aligned}
\mathbf{w}^*, b^*, \rho^* &= \arg\max_{\mathbf{w},b,\rho} \rho \quad \text{s.t.} \quad \frac{y_i(\mathbf{w}^\top \mathbf{x}_i + b)}{||\mathbf{w}||} \geq \ \rho \\
&= \arg\max_{\mathbf{w},b,\rho} \rho \quad \text{s.t.} \quad ||\mathbf{w}|| = 1 \text{ and } y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq \ \rho \\
&= \arg\max_{\mathbf{w},b} ||\mathbf{w}||^2 \quad \text{s.t.} \quad y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq \ 1
\end{aligned}$$

where $1 \leq i \leq n$

Canonical Hyperplanes

- **Definition:** The hyperplane is in *canonical* form w.r.t. $\mathbf{X}$ if $\min_{\mathbf{x}_i \in \mathbf{X}} |\mathbf{w}^\top \mathbf{x}_i + b| = 1$

Canonical Hyperplanes

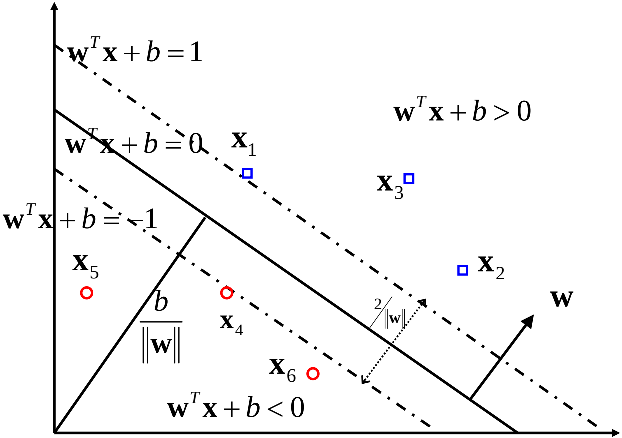
- **Definition:** The hyperplane is in *canonical* form w.r.t. $\mathbf{X}$ if $\min_{\mathbf{x}_i \in \mathbf{X}} |\mathbf{w}^\top \mathbf{x}_i + b| = 1$
- For canonical hyperplanes, the distance of the closest point to the hyperplane ("margin") is $1/||\mathbf{w}||$: Assume we have two points on the two margins: i.e., $\mathbf{x}^+$ on the positive margin and $\mathbf{x}^-$ on the negative margin, so we have

$$f(\mathbf{x}^+) =< \frac{\mathbf{w}}{||\mathbf{w}||}, \mathbf{x}^+ > + \frac{b}{||\mathbf{w}||} = \frac{1}{||\mathbf{w}||}$$
$$f(\mathbf{x}^-) =< \frac{\mathbf{w}}{||\mathbf{w}||}, \mathbf{x}^- > + \frac{b}{||\mathbf{w}||} = -\frac{1}{||\mathbf{w}||}$$

  then the geometric margin ρ is then the functional margin of the resulting classifier

$$\rho = \frac{1}{2}\left(f(\mathbf{x}^+) - f(\mathbf{x}^-)\right) = \frac{1}{||\mathbf{w}||}$$

Separable (Hard-Margin) SVMs



Separable (Hard-Margin) SVMs

For the ease of computation, finally we can write down the objective function with the constraints:

$$\min_{\mathbf{w}} \left\{ \frac{1}{2} \|\mathbf{w}\|^2 \right\}$$
$$\text{s.t.} \qquad \forall_{i=1}^{n} : y_i \left(\mathbf{w}^\top \mathbf{x}_i + b \right) \geq 1$$

This problem can be solved in the primal and dual formulations. usually dealt with by the Lagrange theory, where we can introduce Lagrange multipliers $\alpha_i \geq 0$ and a *Lagrangian* as follows:

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2} ||\mathbf{w}||^2 - \sum_{i=1}^{n} \alpha_i \left(y_i (\mathbf{w}^\top \mathbf{x}_i + b) - 1 \right)$$

Dual Problems

- This problem is usually transformed to its corresponding dual form by introducing Lagrange multipliers $\alpha_i \geq 0$.
- The primal *Lagrangian* is:

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}||\mathbf{w}||^2 - \sum_{i=1}^{n} \alpha_i \left(y_i(\mathbf{w}^\top \mathbf{x}_i + b) - 1 \right),$$

  here, we define **w** and b as the *primal variables* and α_i as the *dual variables*
- The corresponding dual is found by differentiating with respect to the primal variables **w** and b due to the *Karush-Kuhn-Tucher (KKT) conditions*

Dual Problems (Contd)

- According to the KKT conditions, at the optimal point, the derivatives of the Lagrangian $L(\mathbf{w}, b, \boldsymbol{\alpha})$ with respect to the primal variables must vanish
- Hence we have:

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\alpha})}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i = 0 \Rightarrow \quad \mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i$$

and

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\alpha})}{\partial b} = 0 \quad \Rightarrow \quad \sum_{i=1}^{n} \alpha_i y_i = 0$$

Dual Lagrangian

- Substituting

$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i \text{ and } \sum_{i=1}^{n} \alpha_i y_i = 0$$

into the primal $L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}||\mathbf{w}||^2 - \sum_{i=1}^{n} \alpha_i \left(y_i(\mathbf{w}^\top \mathbf{x}_i + b) - 1 \right)$,

to obtain

$$L(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{l=1}^{n} y_i y_l \alpha_i \alpha_l < \mathbf{x}_i, \mathbf{x}_l >$$

$$\text{s.t.} \quad \begin{cases} \alpha_i \geq 0, 1 \leq i \leq n \\ \sum_{i=1}^{n} y_i \alpha_i = 0 \end{cases}$$

Computing b

$$f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + b = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i^\top \mathbf{x} + b$$

Since any support vector $\mathbf{x}_j$ satisfies $y_j f(x_j) = 1$, we have

$$y_j f(\mathbf{x}_j) = y_j \left(\sum_{i \in \mathcal{S}} \alpha_i y_i \mathbf{x}_i^\top \mathbf{x}_j + b \right) = 1$$

$$\Rightarrow f(\mathbf{x}_j) = \sum_{i \in \mathcal{S}} \alpha_i y_i \mathbf{x}_i^\top \mathbf{x}_j + b = y_j$$

By averaging these over all support vectors

$$b^* = \frac{1}{N_{\mathcal{S}}} \sum_{j \in \mathcal{S}} \left(y_j - \sum_{i \in \mathcal{S}} \alpha_i y_i \mathbf{x}_i^\top \mathbf{x}_j \right)$$

Decision Function

$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i$$

The decision function

$$\begin{aligned} f(\mathbf{x}) &= \mathbf{w}^\top \mathbf{x} + b \\ &= \operatorname{sgn}\left[\sum_{i=1}^{n} \alpha_i^* y_i \mathbf{x}_i^\top \mathbf{x} + b^*\right] \end{aligned}$$

Support Vectors

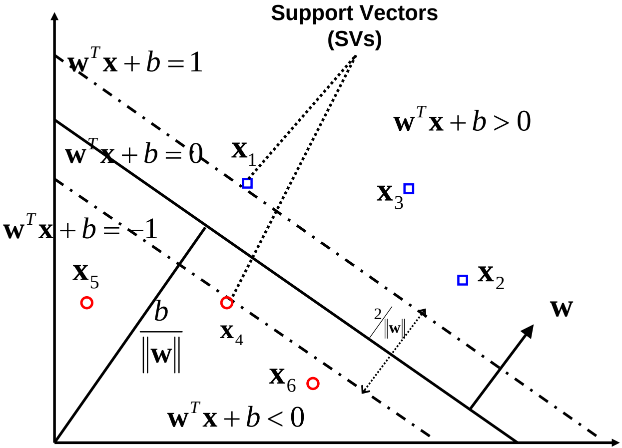
Karush-Kuhn-Tuck (KKT) conditions:

$$\alpha_i \geq 0$$
$$y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1$$
$$\alpha_i \left(y_i(\mathbf{w}^\top \mathbf{x}_i + b) - 1 \right) = 0$$

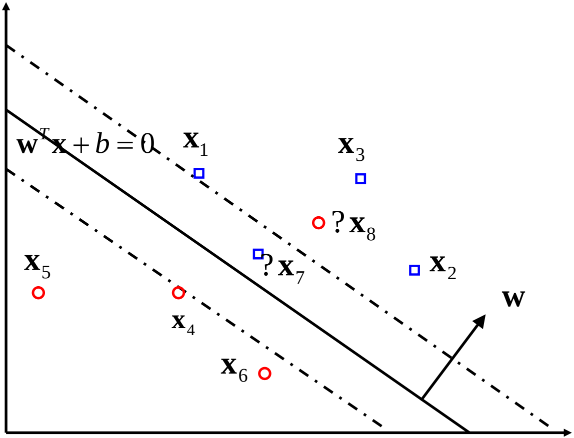
Data points are

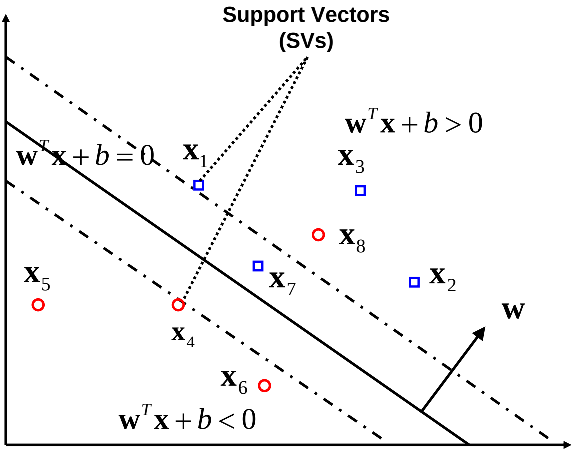
1. no support vectors if $\alpha_i = 0 : y_i(\mathbf{w}^\top \mathbf{x}_i + b) > 1$
2. support vectors if $\alpha_i > 0 : y_i(\mathbf{w}^\top \mathbf{x}_i + b) = 1$

Support Vectors (Contd)



Question?


$\mathbf{w}^T\mathbf{x} + b = 0$
$\mathbf{x}_1$
$\mathbf{x}_3$
$? \mathbf{x}_8$
$\mathbf{x}_5$
$? \mathbf{x}_7$
$\mathbf{x}_2$
$\mathbf{w}$
$\mathbf{x}_4$
$\mathbf{x}_6$


Support Vectors
(SVs)
$\mathbf{w}^T\mathbf{x}+b>0$
$\mathbf{w}^T\mathbf{x}+b=0$
$\mathbf{x}_1$
$\mathbf{x}_3$
$\mathbf{x}_8$
$\mathbf{x}_2$
$\mathbf{x}_5$
$\mathbf{x}_7$
$\mathbf{x}_4$
$\mathbf{w}$
$\mathbf{x}_6$
$\mathbf{w}^T\mathbf{x}+b<0$

Soft-Margin SVMs

- To overcome the sensitivity to the noisy data, a standard approach is to allow for the possibility of example violating the critical constraints by introducing “slack variable":

$$\xi_i \geq 0, \quad \text{for all} \quad 1 \leq i \leq n, \tag{2}$$

along with the relaxed constraints:

$$y_i \left(\mathbf{w}^\top \mathbf{x}_i + b \right) \geq 1 - \xi_i, \quad \text{for all} \quad 1 \leq i \leq n \tag{3}$$

- By making ξ_i large enough, the constraint on $(\mathbf{x}_i, y_i)$ can always be met.

Soft-Margin SVMs (Contd)

- In order not to obtain the trivial solution where all ξ_i take on large values, we thus need to penalize them in the objective function.
- To this end, a term $\sum_i \xi_i$ is included in the objective function as

$$\min_{\mathbf{w},\xi} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i$$

$$\text{s.t.} \qquad \forall_{i=1}^{n} : y_i \left(\mathbf{w}^\top \mathbf{x}_i + b\right) \geq 1 - \xi_i, \xi_i \geq 0$$

Dual Problems

By introducing Lagrange multipliers $\alpha_i \geq 0$ and $\mu_i \geq 0$. The primal *Lagrangian* is:

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}\alpha_i\left(y_i(\mathbf{w}^\top\mathbf{x}_i + b) - 1 + \xi_i\right) - \sum_{i=1}^{n}\mu_i\xi_i,$$

$$\frac{\partial L}{\partial \mathbf{w}} = 0 \quad \Rightarrow \quad \mathbf{w} = \sum_{i=1}^{n}\alpha_i y_i \mathbf{x}_i$$

$$\frac{\partial L}{\partial b} = 0 \quad \Rightarrow \quad \sum_{i=1}^{n}\alpha_i y_i = 0$$

$$\frac{\partial L}{\partial \xi_i} = 0 \quad \Rightarrow \quad \alpha_i = C - \mu_i$$

Solution

With the Lagrange theory and the KKT conditions, we can obtain the dual Lagrangian:

$$L(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{l=1}^{n} y_i y_l \alpha_i \alpha_l \mathbf{x}_i^\top \mathbf{x}_l$$

$$\text{s.t.} \begin{cases} C \geq \alpha_i \geq 0, 1 \leq i \leq n \\ \sum_{i=1}^{n} y_i \alpha_i = 0 \end{cases}$$

- the box constraint
- the optimal value of w in terms of the optimal value of α^*

$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i$$

Support Vectors

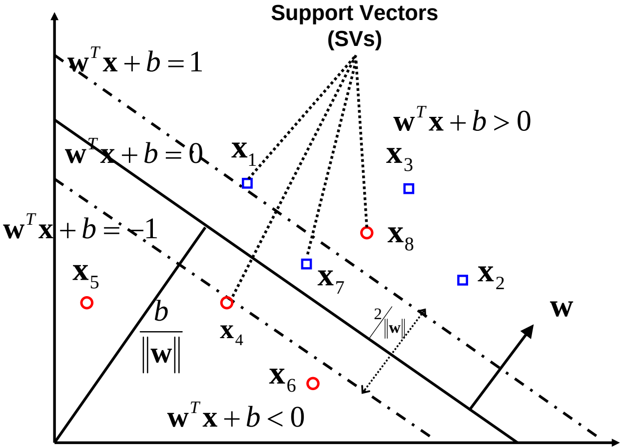
with KKT conditions:

$$
\begin{aligned}
&\alpha_i \geq 0;\ \ \mu_i \geq 0;\ \ \xi_i \geq 0;\ \ \mu_i \xi_i = 0 \\
&y_i(\mathbf{w}^\top \mathbf{x}_i + b) - 1 + \xi_i \geq 0 \\
&\alpha_i \left(y_i(\mathbf{w}^\top \mathbf{x}_i + b) - 1 + \xi_i \right) = 0 \\
&\alpha_i = C - \mu_i
\end{aligned}
$$

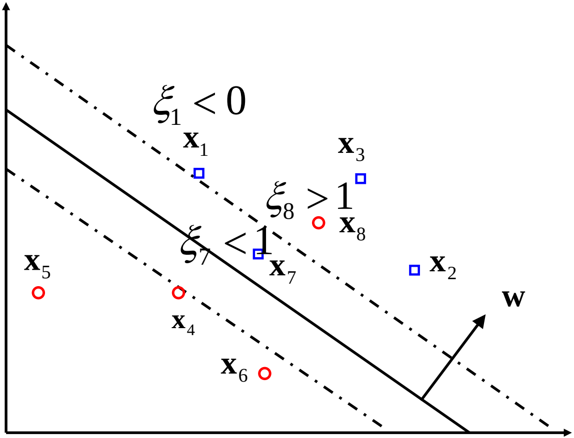
The data points are

1. no contribution to **w** if $\alpha_i = 0$
2. support vectors if $\alpha_i > 0 \quad \Rightarrow \quad y_i(\mathbf{w}^\top \mathbf{x}_i + b) = 1 - \xi_i$
   - $\alpha_i < C \quad \Rightarrow \quad \xi_i = 0$, on the margin
   - $\alpha_i = C \quad \Rightarrow \quad \xi_i > 0$
     1. correctly classified if $\xi_i \leq 1$
     2. misclassified if $\xi_i > 1$

Soft-Margin SVMs: Geometric Illustration

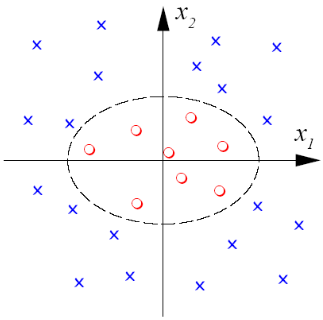


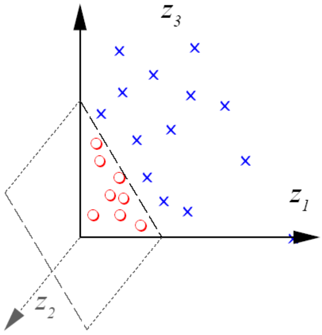
Soft-Margin SVMs: Geometric Illustration (cont)



Nonlinear Mapping

$$\phi : \mathbb{R}^2 \rightarrow \mathbb{R}^3$$
$$(\mathbf{x}_1, \mathbf{x}_2) \rightarrow (\mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3) \; : (\mathbf{x}_1^2, \sqrt{2}\mathbf{x}_1\mathbf{x}_2, \mathbf{x}_2^2)$$





Feature Spaces

- Preprocess the data with

$$\begin{aligned} \phi: \mathbf{X} &\to \mathcal{H} \\ \mathbf{x} &\to \phi(\mathbf{x}), \end{aligned}$$

  where $\mathcal{H}$ is a dot product space and learn the mapping from $\phi(\mathbf{x})$ to the output y
- Usually, $\dim(\mathbf{X}) \ll \dim(\mathcal{H})$

Dual Lagrangian with Kernels

- In linear separable case, the dual formulation is

$$L(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{l=1}^{n} y_i y_l \alpha_i \alpha_l \mathbf{x}_i^\top \mathbf{x}_l$$

$$\text{s.t.} \quad \begin{cases} \alpha_i \geq 0, 1 \leq i \leq n \\ \sum_{i=1}^{n} y_i \alpha_i = 0 \end{cases}$$

Dual Lagrangian with Kernels

- In linear separable case, the dual formulation is

$$L(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{l=1}^{n} y_i y_l \alpha_i \alpha_l k(\mathbf{x}_i, \mathbf{x}_l)$$

$$\text{s.t.} \quad \begin{cases} \alpha_i \geq 0, 1 \leq i \leq n \\ \sum_{i=1}^{n} y_i \alpha_i = 0 \end{cases}$$

- The solution

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i^* y_i \mathbf{x}^\top \mathbf{x}_i + b^*$$

can be formulated as

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i^* y_i k(\mathbf{x}, \mathbf{x}_i) + b^*$$

Positive Definite Kernels

- **Definition (Gram Matrix)**: Given a function $k : \mathcal{X}^2 \to \mathbb{K}$ (where $\mathbb{K} = \mathcal{C}$ or $\mathbb{K} = \mathbb{R}$) and patterns $(\mathbf{x}_i, \cdots, \mathbf{x}_n) \in \mathcal{X}$, the $n \times n$ matrix $\mathbf{K}$ with elements:

$$\mathbf{K}_{ij} := k(\mathbf{x}_i, \mathbf{x}_j)$$

  is called the Gram matrix (or kernel matrix) of k w.r.t $(\mathbf{x}_1, \cdots, \mathbf{x}_n)$
- **Definition (Positive Definite Matrix)**: A complex $n \times n$ matrix $\mathbf{K}$ satisfying

$$\sum_{ij} c_i \bar{c}_j \mathbf{K}_{ij} \geq 0$$

  1. for all $c \in \mathcal{C}$, the matrix $\mathbf{K}$ is called positive definite
  2. for all $c \in \mathcal{R}$, the real symmetric $n \times n$ matrix $\mathbf{K}$ is called positive definite

Positive Definite Kernels (Contd)

- **Definition (Positive Definite Kernel)**: Let $\mathcal{X}$ be a nonempty set. A function k on $\mathcal{X} \times \mathcal{X}$ which for all $n \in \mathbb{N}$, and all $(\mathbf{x}_i, \cdots, \mathbf{x}_n) \in \mathcal{X}$ gives rise to a positive definite Gram matrix is called a positive definite (pd) kernel
- A number of different terms are used for pd kernels, such as reproducing kernel, Mercer kernel, admissible kernel, Support Vector kernel, nonnegative definite kernel, and covariance function
- Conditions of kernels:
  1. positivity on the diagonal of $\mathbf{K}$, i.e., $k(\mathbf{x}, \mathbf{x}) > 0$, for all $\mathbf{x} \in \mathcal{X}$
  2. Symmetry, i.e., $k(\mathbf{x}_i, \mathbf{x}_j) = \overline{k}(\mathbf{x}_j, \mathbf{x}_i)$ or $\mathbf{K}_{ij} = \overline{\mathbf{K}}_{ji}$

Mercer Theorem

- **Mercer Theorem** If k is a continuous kernel of a positive definite integral operator on $L_2(\mathcal{X})$,

$$\int_{\mathcal{X}^2} k(\mathbf{x}, \mathbf{x}^{'}) f(\mathbf{x}) f(\mathbf{x}^{'}) d\mathbf{x} d\mathbf{x}^{'} \geq 0, \quad \forall \quad f$$

It can be expanded as

$$k(\mathbf{x}, \mathbf{x}^{'}) = \sum_{i=1}^{\infty} \lambda_i \psi_i(\mathbf{x}) \psi_i(\mathbf{x}^{'})$$

using eigenfunctions ψ_i and eigenvalues $\lambda_i \geq 0$

Mercer Theorem (Contd)

- In this case,

$$\phi(\mathbf{x}) := \begin{pmatrix} \sqrt{\lambda_1}\psi_1(\mathbf{x}) \\ \sqrt{\lambda_2}\psi_2(\mathbf{x}) \\ \cdot \\ \cdot \\ \cdot \end{pmatrix}$$

- **Proposition (Mercer Kernel Map)** If k is a kernel satisfying the above conditions (in the previous slide), we can construct a mapping ϕ into a space where k acts as a dot product,

$$< \phi(\mathbf{x}), \phi(\mathbf{x}^{'}) >= \phi(\mathbf{x})^{\top}\phi(\mathbf{x}^{'}) = k(\mathbf{x}, \mathbf{x}^{'})$$

Closure Properties of Kernel Functions

Constructing more complex kernels from simpler ones

Let $k_1(\mathbf{x}, \mathbf{y})$ and $k_2(\mathbf{x}, \mathbf{y})$ be the kernels functions. Then the following are all kernels:

1. $k_1(\mathbf{x}, \mathbf{y}) + k_2(\mathbf{x}, \mathbf{y}) \quad \Leftarrow \quad \phi(\mathbf{x}) = (\phi_1(\mathbf{x}), \phi_2(\mathbf{x}))\top$
2. $\alpha k_1(\mathbf{x}, \mathbf{y}), \quad \alpha > 0 \quad \Leftarrow \quad \phi(\mathbf{x}) = \sqrt{\alpha}\phi_1(\mathbf{x})$
3. $k_1(\mathbf{x}, \mathbf{y})k_2(\mathbf{x}, \mathbf{y}) \quad \Leftarrow \quad \phi(\mathbf{x})_{ij} = \phi_i(\mathbf{x})_i\phi_2(\mathbf{x})_j$ (tensor product)
4. $f(\mathbf{x})f(\mathbf{y}), \quad \forall f \quad \Leftarrow \quad \phi(\mathbf{x}) = f(\mathbf{x})$
5. $\mathbf{x}^\top \mathbf{A}\mathbf{y}, \quad$ for $\mathbf{A} \succeq 0 \quad \Leftarrow \quad \phi(\mathbf{x}) = L^\top \mathbf{x}$ for $\mathbf{A} = LL^\top$ (Cholesky)

Kernel Tricks

- any algorithm that only depends on dot products can benefit from the kernel trick
- can be extended to non-vectorial data
- The kernel is as a nonlinear similarity measure (examples)
- Examples of common kernels used
  1. Gaussian kernels: $k(\mathbf{x}, \mathbf{x}^{'}) = \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}^{'}\|^2}{2\sigma^2}\right)$
  2. Polynomial kernels: $k(\mathbf{x}, \mathbf{x}^{'}) = \left(\mathbf{x}^\top \mathbf{x}^{'} + c\right)^d$

One-vs-Rest

- To get C-class binary classifiers, it is common to construct a set of binary classifiers $f^1, \cdots, f^C$, each trained to separate one class from the rest, and combine them by doing the multi-class classification according to the maximal output before applying the sgn function; i.e., by taking:

$$\arg \max_{j=1,\cdots,C} f^j(\mathbf{x}), \text{ where } f^j(\mathbf{x}) = \sum_{i=1}^{n} y_i \alpha^j k(\mathbf{x}, \mathbf{x}_i) + b^j$$

One-vs-Rest: Objective Function

- Multi-class objective functions

$$\min_{\mathbf{w}^j, \xi^j} \left\{ \frac{1}{2} ||\mathbf{w}^j||^2 + C \sum_{i=1}^{n} \xi_i^j \right\}$$

$$\text{s.t.} \quad \begin{cases} < \mathbf{w}^j, \mathbf{x}_i > +b \geq 1 - \xi_i^j, & y_i = j \\ < \mathbf{w}^j, \mathbf{x}_i > +b \leq -1 + \xi_i^j, & y_i \neq j \\ \forall_{i=1}^n : \xi_i^j > 0, j = 1, \cdots, C \end{cases}$$

One-vs-One

- This method constructs $C \times (C-1)/2$ classifiers where each one is trained on data from two classes. For training data from the l-th and the j-th classes, we solve the following binary classification problem:

$$\min_{\mathbf{w}^{lj},\xi^{lj}} \left\{ \frac{1}{2}||\mathbf{w}^{lj}||^2 + C\sum_{i=1}^{n}\xi_i^{lj} \right\}$$

$$\text{s.t.} \quad \begin{cases} <\mathbf{w}^{lj}, \mathbf{x}_i> + b \geq 1 - \xi_i^{lj}, & y_i = l \\ <\mathbf{w}^{lj}, \mathbf{x}_i> + b \leq -1 + \xi_i^{lj}, & y_i = j \\ \forall_{i=1}^{n} : \xi_i^{lj} > 0, l, j = 1, \cdots, C \end{cases}$$

One-vs-One

- Assume the voting strategy used
  1. if sgn $\left[\sum_{i=1}^{j} \alpha_i^{lj} y_i < \mathbf{x}, \xi > + b^{lj}\right]$ says that the pattern **x** belongs to the class j, then the vote for the class j is added by one. Otherwise the class l is added by one
  2. Then we predict the **x** is in the class with the largest vote