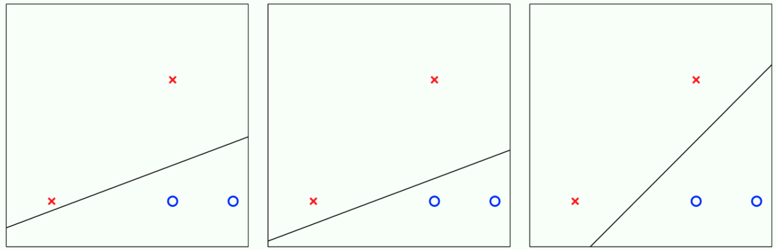
Support Vector Machine

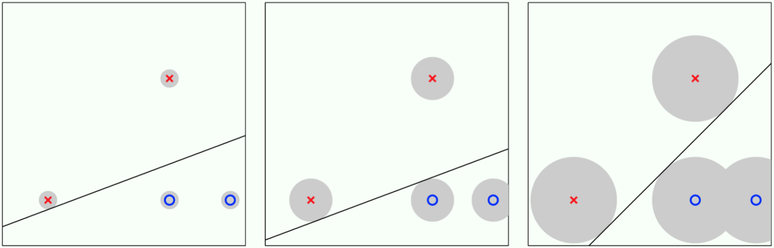
Section 1

(Linear) SVM with Hard-margin

Which Separator Do You Pick?

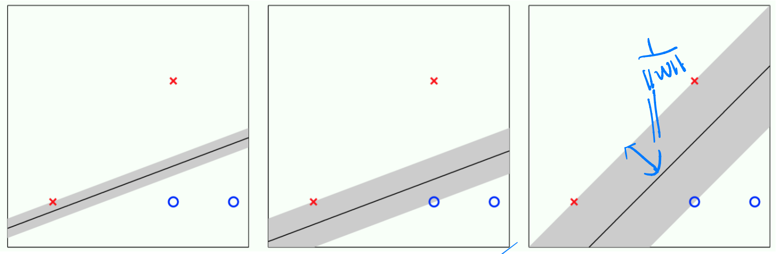


Robustness to Noisy Data



Being robust to noise (measurement error) is good (remember regularization).

Thicker Cushion Means More Robustness



We call such hyperplanes **fat**

Two Crucial Questions

1. Can we efficiently find the fattest separating hyperplane?
2. Is a fatter hyperplane better than a thin one?

Pulling Out the Bias

Before

$$\mathbf{x} \in \{1\} \times \mathbb{R}^d; \mathbf{w} \in \mathbb{R}^{d+1}$$

$$\mathbf{x} = \begin{bmatrix} 1 \\ x_1 \\ \vdots \\ x_d \end{bmatrix}; \mathbf{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_d \end{bmatrix}$$

$$\text{signal} = \mathbf{w}^T \mathbf{x}$$

Pulling Out the Bias

Before

$$\mathbf{x} \in \{1\} \times \mathbb{R}^d; \mathbf{w} \in \mathbb{R}^{d+1}$$

$$\mathbf{x} = \begin{bmatrix} 1 \\ x_1 \\ \vdots \\ x_d \end{bmatrix}; \mathbf{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_d \end{bmatrix}$$

signal $= \mathbf{w}^T \mathbf{x}$

After

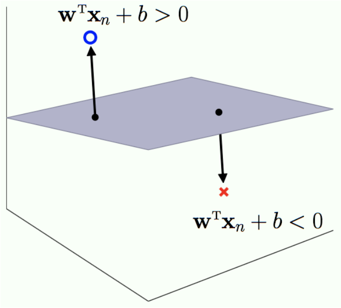
$$\mathbf{x} \in \mathbb{R}^d; b \in \mathbb{R}, \mathbf{w} \in \mathbb{R}^d$$

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix}; \mathbf{w} = \begin{bmatrix} w_1 \\ \vdots \\ w_d \end{bmatrix}$$

bias b

signal $= \mathbf{w}^T \mathbf{x} + b$

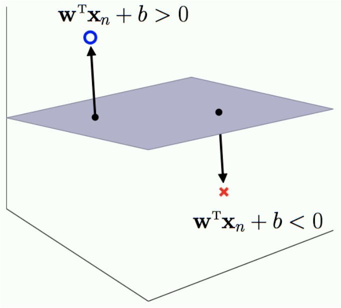
Separating The Data



Hyperplane $h = (b, \mathbf{w})$
h separates the data means:

$$y_n(\mathbf{w}^T\mathbf{x}_n + b) > 0$$

Separating The Data



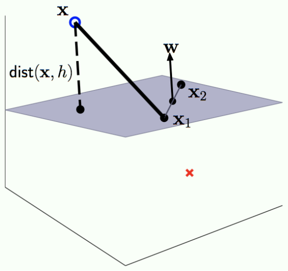
Hyperplane $h = (b, \mathbf{w})$
h separates the data means:

$$y_n(\mathbf{w}^T\mathbf{x}_n + b) > 0$$

By rescaling the weights and bias,

$$\min_{n=1,\ldots,N} y_n(\mathbf{w}^T\mathbf{x}_n + b) = 1$$

Distance to the Hyperplane

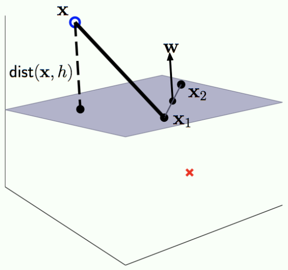


- $\mathbf{w}$ is normal to the hyperplane (why?)

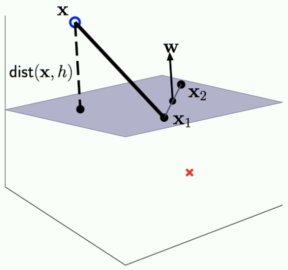
Distance to the Hyperplane

- $\mathbf{w}$ is normal to the hyperplane (why?)

  $\mathbf{w}^T(\mathbf{x}_2 - \mathbf{x}_1) = \mathbf{w}^T\mathbf{x}_2 - \mathbf{w}^T\mathbf{x}_1 = -b + b = 0$



Distance to the Hyperplane

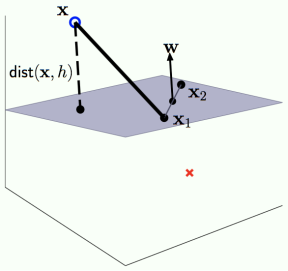


- $\mathbf{w}$ is normal to the hyperplane (why?)
  $\mathbf{w}^T(\mathbf{x}_2 - \mathbf{x}_1) = \mathbf{w}^T\mathbf{x}_2 - \mathbf{w}^T\mathbf{x}_1 = -b + b = 0$
- Scalar projection:

$$\mathbf{a}^T\mathbf{b} = \|\mathbf{a}\|\|\mathbf{b}\|\cos(\mathbf{a}, \mathbf{b})$$

$$\Rightarrow \mathbf{a}^T\mathbf{b}/\|\mathbf{b}\| = \|\mathbf{a}\|\cos(\mathbf{a}, \mathbf{b})$$

Distance to the Hyperplane

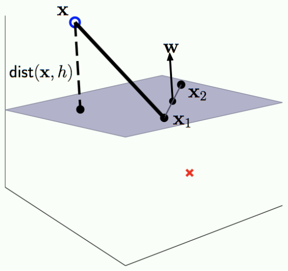


- $\mathbf{w}$ is normal to the hyperplane (why?)
  $\mathbf{w}^T(\mathbf{x}_2 - \mathbf{x}_1) = \mathbf{w}^T\mathbf{x}_2 - \mathbf{w}^T\mathbf{x}_1 = -b + b = 0$
- Scalar projection:

$$\mathbf{a}^T\mathbf{b} = \|\mathbf{a}\|\|\mathbf{b}\|\cos(\mathbf{a}, \mathbf{b})$$
$$\Rightarrow \mathbf{a}^T\mathbf{b}/\|\mathbf{b}\| = \|\mathbf{a}\|\cos(\mathbf{a}, \mathbf{b})$$

- let $\mathbf{x}_\perp$ be the orthogonal projection of $\mathbf{x}$ to h, distance to hyperplane is given by projection of $\mathbf{x} - \mathbf{x}_\perp$ to $\mathbf{w}$ (why?)

Distance to the Hyperplane



- $\mathbf{w}$ is normal to the hyperplane (why?)
  $\mathbf{w}^T(\mathbf{x}_2 - \mathbf{x}_1) = \mathbf{w}^T\mathbf{x}_2 - \mathbf{w}^T\mathbf{x}_1 = -b + b = 0$
- Scalar projection:

$$\mathbf{a}^T\mathbf{b} = \|\mathbf{a}\|\|\mathbf{b}\|\cos(\mathbf{a}, \mathbf{b})$$
$$\Rightarrow \mathbf{a}^T\mathbf{b}/\|\mathbf{b}\| = \|\mathbf{a}\|\cos(\mathbf{a}, \mathbf{b})$$

- let $\mathbf{x}_\perp$ be the orthogonal projection of $\mathbf{x}$ to h, distance to hyperplane is given by projection of $\mathbf{x} - \mathbf{x}_\perp$ to $\mathbf{w}$ (why?)

$$\begin{aligned}\text{dist}(\mathbf{x}, h) &= \frac{1}{\|\mathbf{w}\|} \cdot |\mathbf{w}^T\mathbf{x} - \mathbf{w}^T\mathbf{x}_\perp| \\ &= \frac{1}{\|\mathbf{w}\|} \cdot |\mathbf{w}^T\mathbf{x} + b|\end{aligned}$$

Fatness of a Separating Hyperplane

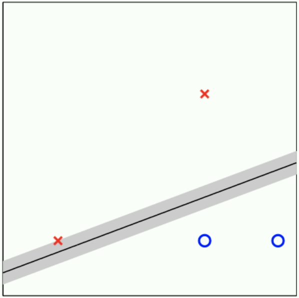
$$\text{dist}(\mathbf{x}, h) = \frac{1}{\|\mathbf{w}\|} \cdot |\mathbf{w}^T\mathbf{x} + b| = \frac{1}{\|\mathbf{w}\|} \cdot |y_n(\mathbf{w}^T\mathbf{x} + b)| = \frac{1}{\|\mathbf{w}\|} \cdot y_n(\mathbf{w}^T\mathbf{x} + b)$$

Fatness of a Separating Hyperplane

$$\text{dist}(\mathbf{x}, h) = \frac{1}{\|\mathbf{w}\|} \cdot |\mathbf{w}^T\mathbf{x} + b| = \frac{1}{\|\mathbf{w}\|} \cdot |y_n(\mathbf{w}^T\mathbf{x} + b)| = \frac{1}{\|\mathbf{w}\|} \cdot y_n(\mathbf{w}^T\mathbf{x} + b)$$

Fatness

= Distance to the closest point



$$\begin{aligned}\text{Fatness} &= \min_n \text{dist}(\mathbf{x}_n, h) \\ &= \frac{1}{\|\mathbf{w}\|} \min_n y_n(\mathbf{w}^T\mathbf{x} + b) \\ &= \frac{1}{\|\mathbf{w}\|}\end{aligned}$$

Maximizing the Margin

- Formal definition of margin:

$$\text{margin: } \gamma(h) = \frac{1}{\|\mathbf{w}\|}$$

Maximizing the Margin

- Formal definition of margin:

$$\text{margin: } \gamma(h) = \frac{1}{\|\mathbf{w}\|}$$

  - NOTE: Bias b does not appear in the margin.

Maximizing the Margin

- Formal definition of margin:

$$\text{margin: } \gamma(h) = \frac{1}{\|\mathbf{w}\|}$$

  - NOTE: Bias b does not appear in the margin.
- Objective maximizing margin:

$$\min_{b,\mathbf{w}} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w}$$

$$\text{subject to: } \min_{n=1,\ldots,N} y_n(\mathbf{w}^T\mathbf{x}_n + b) = 1$$

Maximizing the Margin

- Formal definition of margin:

$$\text{margin: } \gamma(h) = \frac{1}{\|\mathbf{w}\|}$$

  - NOTE: Bias b does not appear in the margin.
- Objective maximizing margin:

$$\min_{b,\mathbf{w}} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w}$$

$$\text{subject to: } \min_{n=1,\ldots,N} y_n(\mathbf{w}^T\mathbf{x}_n + b) = 1$$

- An equivalent objective:

$$\min_{b,\mathbf{w}} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w}$$

$$\text{subject to: } y_n(\mathbf{w}^T\mathbf{x}_n + b) \geq 1 \text{ for } n = 1, \ldots, N$$

Example - Our Toy Data Set

$$\min_{b,\mathbf{w}} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w}$$

subject to: $y_n(\mathbf{w}^T\mathbf{x}_n + b) \geq 1$ for $n = 1, \ldots, N$

Training Data:

$$X = \begin{bmatrix} 0 & 0 \\ 2 & 2 \\ 2 & 0 \\ 3 & 0 \end{bmatrix}, \mathbf{y} = \begin{bmatrix} -1 \\ -1 \\ +1 \\ +1 \end{bmatrix}$$

What is the margin?

Example - Our Toy Data Set

$$\min_{b,\mathbf{w}} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w}$$

subject to: $y_n(\mathbf{w}^T\mathbf{x}_n + b) \geq 1$ for $n = 1, \ldots, N$

$$X = \begin{bmatrix} 0 & 0 \\ 2 & 2 \\ 2 & 0 \\ 3 & 0 \end{bmatrix}, \mathbf{y} = \begin{bmatrix} -1 \\ -1 \\ +1 \\ +1 \end{bmatrix} \quad \Rightarrow \begin{cases} (1) : -b \geq 1 \\ (2) : -(2w_1 + 2w_2 + b) \geq 1 \\ (3) : 2w_1 + b \geq 1 \\ (4) : 3w_1 + b \geq 1 \end{cases}$$

$$\begin{cases} (1) + (3) & \rightarrow w_1 \geq 1 \\ (2) + (3) & \rightarrow w_2 \leq -1 \end{cases} \Rightarrow \frac{1}{2}\mathbf{w}^T\mathbf{w} = \frac{1}{2}(w_1^2 + w_2^2) \geq 1$$

Example - Our Toy Data Set

$$\min_{b,\mathbf{w}} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w}$$

subject to: $y_n(\mathbf{w}^T\mathbf{x}_n + b) \geq 1$ for $n = 1, \ldots, N$

$$X = \begin{bmatrix} 0 & 0 \\ 2 & 2 \\ 2 & 0 \\ 3 & 0 \end{bmatrix}, \mathbf{y} = \begin{bmatrix} -1 \\ -1 \\ +1 \\ +1 \end{bmatrix} \quad \Rightarrow \begin{cases} (1) : -b \geq 1 \\ (2) : -(2w_1 + 2w_2 + b) \geq 1 \\ (3) : 2w_1 + b \geq 1 \\ (4) : 3w_1 + b \geq 1 \end{cases}$$

$$\begin{cases} (1) + (3) & \rightarrow w_1 \geq 1 \\ (2) + (3) & \rightarrow w_2 \leq -1 \end{cases} \Rightarrow \frac{1}{2}\mathbf{w}^T\mathbf{w} = \frac{1}{2}(w_1^2 + w_2^2) \geq 1$$

Thus: $w_1 = 1, w_2 = -1, b = -1$

Example - Our Toy Data Set

- Given data $X = \begin{bmatrix} 0 & 0 \\ 2 & 2 \\ 2 & 0 \\ 3 & 0 \end{bmatrix}$
- Optimal solution

$$\mathbf{w}^* = \begin{bmatrix} w_1 = 1 \\ w_2 = -1 \end{bmatrix}, b^* = -1$$

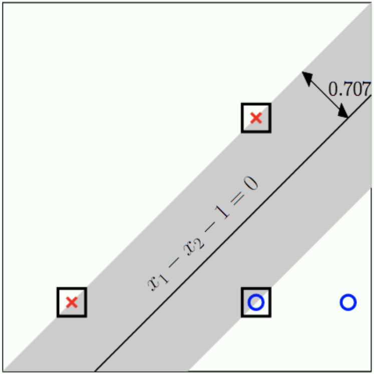
- Optimal hyperplane
  $g(\mathbf{x}) = \mathsf{sign}(x_1 - x_2 - 1)$

Example - Our Toy Data Set

- Given data $X = \begin{bmatrix} 0 & 0 \\ 2 & 2 \\ 2 & 0 \\ 3 & 0 \end{bmatrix}$
- Optimal solution

$$\mathbf{w}^* = \begin{bmatrix} w_1 = 1 \\ w_2 = -1 \end{bmatrix}, b^* = -1$$

- Optimal hyperplane
  $g(\mathbf{x}) = \text{sign}(x_1 - x_2 - 1)$
- margin:
  $\frac{1}{\|w\|} = \frac{1}{\sqrt{2}} \approx 0.707$

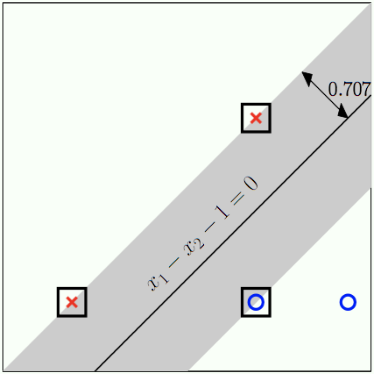


Example - Our Toy Data Set

- Given data $X = \begin{bmatrix} 0 & 0 \\ 2 & 2 \\ 2 & 0 \\ 3 & 0 \end{bmatrix}$
- Optimal solution

$$\mathbf{w}^* = \begin{bmatrix} w_1 = 1 \\ w_2 = -1 \end{bmatrix}, b^* = -1$$

- Optimal hyperplane
  $g(\mathbf{x}) = \text{sign}(x_1 - x_2 - 1)$
- margin:
  $\frac{1}{\|w\|} = \frac{1}{\sqrt{2}} \approx 0.707$



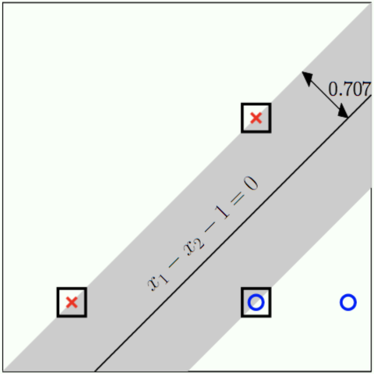
For data points (1), (2) and (3) $y_n(\mathbf{x}_n^T \mathbf{w}^* + b^*) = 1$

Example - Our Toy Data Set

- Given data $X = \begin{bmatrix} 0 & 0 \\ 2 & 2 \\ 2 & 0 \\ 3 & 0 \end{bmatrix}$
- Optimal solution

$$\mathbf{w}^* = \begin{bmatrix} w_1 = 1 \\ w_2 = -1 \end{bmatrix}, b^* = -1$$

- Optimal hyperplane
  $g(\mathbf{x}) = \text{sign}(x_1 - x_2 - 1)$
- margin:
  $\frac{1}{\|w\|} = \frac{1}{\sqrt{2}} \approx 0.707$



For data points (1), (2) and (3) $y_n(\mathbf{x}_n^T \mathbf{w}^* + b^*) = 1$
Support Vectors

Solver: Quadratic Programming

$$\min_{\mathbf{u}\in\mathbb{R}^q} \quad \frac{1}{2}\mathbf{u}^T Q\mathbf{u} + \mathbf{p}^T\mathbf{u}$$

$$\text{subject to: } A\mathbf{u} \geq \mathbf{c}$$

$$\mathbf{u}^* \leftarrow QP(Q, \mathbf{p}, A, \mathbf{c})$$

($Q = 0$ is linear programming.)
http://cvxopt.org/examples/tutorial/qp.html

Maximum Margin Hyperplane is QP

$$\min_{b,\mathbf{w}} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w}$$

subject to: $y_n(\mathbf{w}^T\mathbf{x}_n + b) \geq 1, \forall n$

$$\min_{\mathbf{u}\in\mathbb{R}^q} \quad \frac{1}{2}\mathbf{u}^T Q\mathbf{u} + \mathbf{p}^T\mathbf{u}$$

subject to: $A\mathbf{u} \geq \mathbf{c}$

$$\mathbf{u} = \begin{bmatrix} b \\ \mathbf{w} \end{bmatrix} \in \mathbb{R}^{d+1} \Rightarrow \frac{1}{2}\mathbf{w}^T\mathbf{w} = [b, \mathbf{w}^T]\begin{bmatrix} 0 & \mathbf{0}_d^T \\ \mathbf{0}_d & I_d \end{bmatrix}\begin{bmatrix} b \\ \mathbf{w}^T \end{bmatrix} = \mathbf{u}^T\begin{bmatrix} 0 & \mathbf{0}_d^T \\ \mathbf{0}_d & I_d \end{bmatrix}\mathbf{u}$$

Maximum Margin Hyperplane is QP

$$\min_{b,\mathbf{w}} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w}$$

subject to: $y_n(\mathbf{w}^T\mathbf{x}_n + b) \geq 1, \forall n$

$$\min_{\mathbf{u}\in\mathbb{R}^q} \quad \frac{1}{2}\mathbf{u}^T Q\mathbf{u} + \mathbf{p}^T\mathbf{u}$$

subject to: $A\mathbf{u} \geq \mathbf{c}$

$$\mathbf{u} = \begin{bmatrix} b \\ \mathbf{w} \end{bmatrix} \in \mathbb{R}^{d+1} \Rightarrow \frac{1}{2}\mathbf{w}^T\mathbf{w} = [b, \mathbf{w}^T] \begin{bmatrix} 0 & \mathbf{0}_d^T \\ \mathbf{0}_d & I_d \end{bmatrix} \begin{bmatrix} b \\ \mathbf{w}^T \end{bmatrix} = \mathbf{u}^T \begin{bmatrix} 0 & \mathbf{0}_d^T \\ \mathbf{0}_d & I_d \end{bmatrix} \mathbf{u}$$

$$Q = \begin{bmatrix} 0 & \mathbf{0}_d^T \\ \mathbf{0}_d & I_d \end{bmatrix}, \mathbf{p} = \mathbf{0}_{d+1}$$

Maximum Margin Hyperplane is QP

$$\min_{b,\mathbf{w}} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w}$$

subject to: $y_n(\mathbf{w}^T\mathbf{x}_n + b) \geq 1, \forall n$

$$\min_{\mathbf{u}\in\mathbb{R}^q} \quad \frac{1}{2}\mathbf{u}^T Q\mathbf{u} + \mathbf{p}^T\mathbf{u}$$

subject to: $A\mathbf{u} \geq \mathbf{c}$

$$\mathbf{u} = \begin{bmatrix} b \\ \mathbf{w} \end{bmatrix} \in \mathbb{R}^{d+1} \Rightarrow \frac{1}{2}\mathbf{w}^T\mathbf{w} = [b, \mathbf{w}^T]\begin{bmatrix} 0 & \mathbf{0}_d^T \\ \mathbf{0}_d & I_d \end{bmatrix}\begin{bmatrix} b \\ \mathbf{w}^T \end{bmatrix} = \mathbf{u}^T\begin{bmatrix} 0 & \mathbf{0}_d^T \\ \mathbf{0}_d & I_d \end{bmatrix}\mathbf{u}$$

$$Q = \begin{bmatrix} 0 & \mathbf{0}_d^T \\ \mathbf{0}_d & I_d \end{bmatrix}, \mathbf{p} = \mathbf{0}_{d+1}$$

$$y_n(\mathbf{w}^T\mathbf{x}_n + b) \geq 1 = [y_n, y_n\mathbf{x}_n^T]\mathbf{u} \geq 1 \Rightarrow \begin{bmatrix} y_1 & y_1\mathbf{x}_1^T \\ \vdots & \vdots \\ y_N & y_N\mathbf{x}_N^T \end{bmatrix}\mathbf{u} \geq \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$$

Maximum Margin Hyperplane is QP

$$\min_{b,\mathbf{w}} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w}$$

subject to: $y_n(\mathbf{w}^T\mathbf{x}_n + b) \geq 1, \forall n$

$$\min_{\mathbf{u}\in\mathbb{R}^q} \quad \frac{1}{2}\mathbf{u}^T Q\mathbf{u} + \mathbf{p}^T\mathbf{u}$$

subject to: $A\mathbf{u} \geq \mathbf{c}$

$$\mathbf{u} = \begin{bmatrix} b \\ \mathbf{w} \end{bmatrix} \in \mathbb{R}^{d+1} \Rightarrow \frac{1}{2}\mathbf{w}^T\mathbf{w} = [b, \mathbf{w}^T]\begin{bmatrix} 0 & \mathbf{0}_d^T \\ \mathbf{0}_d & I_d \end{bmatrix}\begin{bmatrix} b \\ \mathbf{w}^T \end{bmatrix} = \mathbf{u}^T\begin{bmatrix} 0 & \mathbf{0}_d^T \\ \mathbf{0}_d & I_d \end{bmatrix}\mathbf{u}$$

$$Q = \begin{bmatrix} 0 & \mathbf{0}_d^T \\ \mathbf{0}_d & I_d \end{bmatrix}, \mathbf{p} = \mathbf{0}_{d+1}$$

$$y_n(\mathbf{w}^T\mathbf{x}_n + b) \geq 1 = [y_n, y_n\mathbf{x}_n^T]\mathbf{u} \geq 1 \Rightarrow \begin{bmatrix} y_1 & y_1\mathbf{x}_1^T \\ \vdots & \vdots \\ y_N & y_N\mathbf{x}_N^T \end{bmatrix}\mathbf{u} \geq \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$$

$$A = \begin{bmatrix} y_1 & y_1\mathbf{x}_1^T \\ \vdots & \vdots \\ y_N & y_N\mathbf{x}_N^T \end{bmatrix}, \boldsymbol{c} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$$

Back To Our Example

Exercise:

$$X = \begin{bmatrix} 0 & 0 \\ 2 & 2 \\ 2 & 0 \\ 3 & 0 \end{bmatrix}, \mathbf{y} = \begin{bmatrix} -1 \\ -1 \\ +1 \\ +1 \end{bmatrix} \qquad \begin{cases} (1) : -b \geq 1 \\ (2) : -(2w_1 + 2w_2 + b) \geq 1 \\ (3) : 2w_1 + b \geq 1 \\ (4) : 3w_1 + b \geq 1 \end{cases}$$

Show the corresponding $Q, \mathbf{p}, A, \mathbf{c}$.

Back To Our Example

Exercise:

$$X = \begin{bmatrix} 0 & 0 \\ 2 & 2 \\ 2 & 0 \\ 3 & 0 \end{bmatrix}, \mathbf{y} = \begin{bmatrix} -1 \\ -1 \\ +1 \\ +1 \end{bmatrix} \qquad \begin{cases} (1) : -b \geq 1 \\ (2) : -(2w_1 + 2w_2 + b) \geq 1 \\ (3) : 2w_1 + b \geq 1 \\ (4) : 3w_1 + b \geq 1 \end{cases}$$

Show the corresponding $Q, \mathbf{p}, A, \mathbf{c}$.

$$Q = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{p} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, A = \begin{bmatrix} -1 & 0 & 0 \\ -1 & -2 & -2 \\ 1 & 2 & 0 \\ 1 & 3 & 0 \end{bmatrix}, \mathbf{c} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Back To Our Example

Exercise:

$$X = \begin{bmatrix} 0 & 0 \\ 2 & 2 \\ 2 & 0 \\ 3 & 0 \end{bmatrix}, \mathbf{y} = \begin{bmatrix} -1 \\ -1 \\ +1 \\ +1 \end{bmatrix} \qquad \begin{cases} (1) : -b \geq 1 \\ (2) : -(2w_1 + 2w_2 + b) \geq 1 \\ (3) : 2w_1 + b \geq 1 \\ (4) : 3w_1 + b \geq 1 \end{cases}$$

Show the corresponding $Q, \mathbf{p}, A, \mathbf{c}$.

$$Q = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{p} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, A = \begin{bmatrix} -1 & 0 & 0 \\ -1 & -2 & -2 \\ 1 & 2 & 0 \\ 1 & 3 & 0 \end{bmatrix}, \mathbf{c} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Use your QP-solver to give

$$\boldsymbol{u}^* = [b^*, w_1^*, w_2^*]^T = [-1, 1, -1]$$

Link to Regularization

$$\min_{\mathbf{w}} \quad E_{in}(\mathbf{w})$$
$$\text{subject to: } \mathbf{w}^T\mathbf{w} \leq C$$

	optimal hyperplane	regularization
minimize	$\mathbf{w}^T\mathbf{w}$	E_{in}
subject to	$E_{in} = 0$	$\mathbf{w}^T\mathbf{w} \leq C$

Section 2

Dual Problem of SVM

Primal and Dual in Optimization

Primal Problem

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

Primal and Dual in Optimization

Primal Problem

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

Generalized Lagrangian

$$\mathcal{L}(\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = f(w) + \sum_{i=1}^{k} \boldsymbol{\alpha}_i g_i(w) + \sum_{i=1}^{l} \boldsymbol{\beta}_i h_i(\mathbf{w})$$

Primal and Dual in Optimization

Primal Problem

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

Generalized Lagrangian

$$\mathcal{L}(\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = f(w) + \sum_{i=1}^{k} \boldsymbol{\alpha}_i g_i(w) + \sum_{i=1}^{l} \boldsymbol{\beta}_i h_i(\mathbf{w})$$

Dual Problem

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

Primal and Dual in Optimization

Primal Problem

$$\begin{array}{lll} \min_{\mathbf{w}} & f(\mathbf{w}) & \\ \text{s.t.} & g_i(\mathbf{w}) \leq 0, & i = 1, \ldots, k \\ & h_i(\mathbf{w}) = 0, & i = 1, \ldots, l \end{array}$$

Generalized Lagrangian

$$\mathcal{L}(\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = f(w) + \sum_{i=1}^{k} \boldsymbol{\alpha}_i g_i(w) + \sum_{i=1}^{l} \boldsymbol{\beta}_i h_i(\mathbf{w})$$

Dual Problem

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

Under some "certain assumptions", there must exist $\mathbf{w}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*$ so that $\mathbf{w}^*$ is the solution to the primal problem, $\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*$ are the solution to the dual problem. (check more details in `https://web.stanford.edu/~boyd/cvxbook/bv_cvxbook.pdf`).

KKT Conditions

Assuming these "certain assumptions" are satisfied, the optimal solutions $\mathbf{w}^*$, $\boldsymbol{\alpha}^*$ and β^* satisfy the **Karush-Kuhn-Tucker (KKT) conditions**, which are as follows:

$$
\begin{aligned}
\frac{\partial}{\partial \mathbf{w}_i} \mathcal{L}\left(\mathbf{w}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*\right) &= 0, i = 1, \ldots, d \\
\frac{\partial}{\partial \boldsymbol{\beta}_i} \mathcal{L}\left(\mathbf{w}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*\right) &= 0, i = 1, \ldots, l \\
\boldsymbol{\alpha}_i^* g_i\left(\mathbf{w}^*\right) &= 0, i = 1, \ldots, k \\
g_i\left(\mathbf{w}^*\right) &\leq 0, i = 1, \ldots, k \\
\boldsymbol{\alpha}_i^* &\geq 0, i = 1, \ldots, k
\end{aligned}
$$

Back to the SVM: Dual Problem

SVM: Primal

$$\min_{b,\mathbf{w}} \frac{1}{2}\mathbf{w}^T\mathbf{w}$$

subject to: $y_n\left(\mathbf{w}^T\mathbf{x}_n + b\right) \geq 1$ for $n = 1, \ldots, N$

Back to the SVM: Dual Problem

SVM: Primal

$$\min_{b,\mathbf{w}} \frac{1}{2}\mathbf{w}^T\mathbf{w}$$

subject to: $y_n\left(\mathbf{w}^T\mathbf{x}_n + b\right) \geq 1$ for $n = 1, \ldots, N$

We can write the constraints as

$$g_n(\mathbf{w}) = -y_n\left(\mathbf{w}^T\mathbf{x}_n + b\right) + 1 \leq 0$$

Back to the SVM: Dual Problem

SVM: Primal

$$\min_{b,\mathbf{w}} \frac{1}{2}\mathbf{w}^T\mathbf{w}$$

$$\text{subject to: } y_n\left(\mathbf{w}^T\mathbf{x}_n + b\right) \geq 1 \text{ for } n = 1, \ldots, N$$

We can write the constraints as

$$g_n(\mathbf{w}) = -y_n\left(\mathbf{w}^T\mathbf{x}_n + b\right) + 1 \leq 0$$

Lagrangian

$$\mathcal{L}(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^{m} \boldsymbol{\alpha}_n\left[y_n\left(\mathbf{w}^T\mathbf{x}_n + b\right) - 1\right]$$

SVM Dual: KKT Conditions

Lagrangian

$$\mathcal{L}(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^{m} \boldsymbol{\alpha}_n \left[y_n \left(\mathbf{w}^T \mathbf{x}_n + b \right) - 1 \right]$$

SVM Dual: KKT Conditions

Lagrangian

$$\mathcal{L}(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^{m} \boldsymbol{\alpha}_n \left[y_n \left(\mathbf{w}^T\mathbf{x}_n + b\right) - 1\right]$$

KKT conditions:

$$\nabla_{\mathbf{w}}\mathcal{L}(\mathbf{w}, b, \boldsymbol{\alpha}) = \mathbf{w} - \sum_{n=1}^{N} \boldsymbol{\alpha}_n y_n \mathbf{x}_n = 0 \tag{1}$$

$$\frac{\partial}{\partial b}\mathcal{L}(\mathbf{w}, b, \boldsymbol{\alpha}) = \sum_{n=1}^{N} \boldsymbol{\alpha}_n y_n = 0 \tag{2}$$

$$\boldsymbol{\alpha}_n g_n\left(\mathbf{w}\right) = 0, i = 1, \ldots, N \tag{3}$$

$$g_n\left(\mathbf{w}\right) \leq 0, i = 1, \ldots, N \tag{4}$$

$$\boldsymbol{\alpha}_n \geq 0, i = 1, \ldots, N \tag{5}$$

SVM Dual

From Eq. (1), we have

$$\mathbf{w} = \sum_{n=1}^{N} \boldsymbol{\alpha}_n y_n \mathbf{x}_n \tag{6}$$

SVM Dual

From Eq. (1), we have

$$\mathbf{w} = \sum_{n=1}^{N} \boldsymbol{\alpha}_n y_n \mathbf{x}_n \tag{6}$$

Replace Eq. (6) back to the Lagrangian:

$$\mathcal{L}(\mathbf{w}, b, \boldsymbol{\alpha}) = \sum_{n=1}^{N} \boldsymbol{\alpha}_n - \frac{1}{2} \sum_{n,m=1}^{N} y_n y_m \boldsymbol{\alpha}_n \boldsymbol{\alpha}_m \mathbf{x}_n^T \mathbf{x}_m - b \sum_{n=1}^{N} \boldsymbol{\alpha}_n y_n$$

SVM Dual

From Eq. (1), we have

$$\mathbf{w} = \sum_{n=1}^{N} \boldsymbol{\alpha}_n y_n \mathbf{x}_n \tag{6}$$

Replace Eq. (6) back to the Lagrangian:

$$\mathcal{L}(\mathbf{w}, b, \boldsymbol{\alpha}) = \sum_{n=1}^{N} \boldsymbol{\alpha}_n - \frac{1}{2} \sum_{n,m=1}^{N} y_n y_m \boldsymbol{\alpha}_n \boldsymbol{\alpha}_m \mathbf{x}_n^T \mathbf{x}_m - b \sum_{n=1}^{N} \boldsymbol{\alpha}_n y_n$$

From Eq. (2), the last term must be zero, so we obtain

$$\mathcal{L}(\mathbf{w}, b, \boldsymbol{\alpha}) = \sum_{n=1}^{N} \boldsymbol{\alpha}_n - \frac{1}{2} \sum_{n,m=1}^{N} y_n y_m \boldsymbol{\alpha}_n \boldsymbol{\alpha}_m \mathbf{x}_n^T \mathbf{x}_m$$

SVM: Dual

The Dual problem for SVM now becomes

$$\max_{\boldsymbol{\alpha}\in\mathbb{R}^N} \sum_{n=1}^{N} \boldsymbol{\alpha}_n - \frac{1}{2}\sum_{m=1}^{N}\sum_{n=1}^{N} \boldsymbol{\alpha}_n \boldsymbol{\alpha}_m y_n y_m \mathbf{x}_n^T \mathbf{x}_m$$

$$\text{subject to } \sum_{n=1}^{N} y_n \boldsymbol{\alpha}_n = 0, \boldsymbol{\alpha}_n \geq 0, \forall n$$

which is also a QP problem.

SVM Dual: Prediction

After we solve the dual problem and denote $\boldsymbol{\alpha}^*$ as the optimal of the dual problem.

- We can obtain the primal solution:

$$\mathbf{w}^* = \sum_{n=1}^{N} y_n \boldsymbol{\alpha}_n^* \mathbf{x}_n$$

where for support vectors $\boldsymbol{\alpha}_n > 0$ (why?)

SVM Dual: Prediction

After we solve the dual problem and denote $\boldsymbol{\alpha}^*$ as the optimal of the dual problem.

- We can obtain the primal solution:

$$\mathbf{w}^* = \sum_{n=1}^{N} y_n \boldsymbol{\alpha}_n^* \mathbf{x}_n$$

  where for support vectors $\boldsymbol{\alpha}_n > 0$ (why?)

  - Recall Eq.(3) in KKT conditions:

$$\boldsymbol{\alpha}_n g_n(\mathbf{w}) = 0, i = 1, \ldots, N$$

$$g_n(\mathbf{w}) = -y_n \left(\mathbf{w}^T \mathbf{x}_n + b\right) + 1$$

SVM Dual

- The optimal hyperplane:

$$
\begin{aligned}
h(\mathbf{x}) &= \mathbf{w}^{*T}\mathbf{x} + b^* \\
&= \sum_{n=1}^{N} y_n \boldsymbol{\alpha}_n^* \mathbf{x}_n^T \mathbf{x} + b^* \\
&= \sum_{\boldsymbol{\alpha}_n^* > 0} y_n \boldsymbol{\alpha}_n^* \mathbf{x}_n^T \mathbf{x} + b^*
\end{aligned}
$$

SVM Dual

- The optimal hyperplane:

$$
\begin{aligned}
h(\mathbf{x}) &= \mathbf{w}^{*T}\mathbf{x} + b^* \\
&= \sum_{n=1}^{N} y_n \boldsymbol{\alpha}_n^* \mathbf{x}_n^T \mathbf{x} + b^* \\
&= \sum_{\boldsymbol{\alpha}_n^* > 0} y_n \boldsymbol{\alpha}_n^* \mathbf{x}_n^T \mathbf{x} + b^*
\end{aligned}
$$

Only Support Vectors Matter

Dual SVM: Summary

$$\max_{\boldsymbol{\alpha} \in \mathbb{R}^N} \sum_{n=1}^{N} \boldsymbol{\alpha}_n - \frac{1}{2} \sum_{m=1}^{N} \sum_{n=1}^{N} \boldsymbol{\alpha}_n \boldsymbol{\alpha}_m y_n y_m \mathbf{x}_n^T \mathbf{x}_m$$

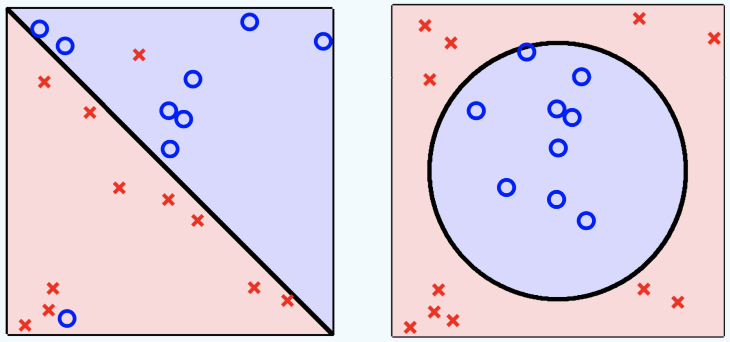
$$\text{subject to } \sum_{n=1}^{N} y_n \boldsymbol{\alpha}_n = 0, \boldsymbol{\alpha}_n \geq 0, \forall n$$

$$\mathbf{w}^* = \sum_{n=1}^{N} y_n \boldsymbol{\alpha}_n^* \mathbf{x}_n$$

Section 3

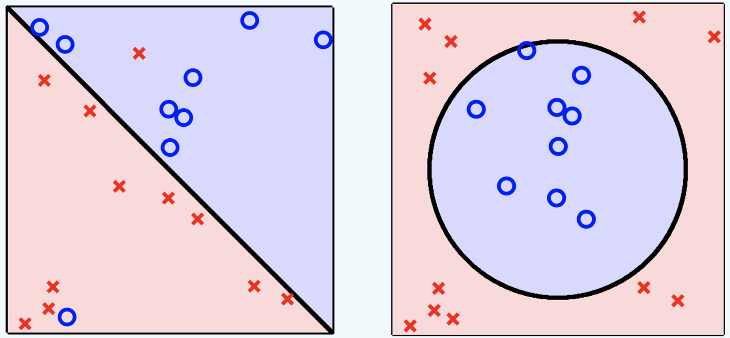
Non-linear SVM

How to Handle Non-(Linear)Separable Data?



(a) Few noisy data. (b) Nonlinearly separable.

How to Handle Non-(Linear)Separable Data?



(a) Few noisy data. (b) Nonlinearly separable.

(a) Tolerate noisy data points: soft-margin SVM.

(b) Inherent nonlinear boundary: non-linear transformation.

Non-Linear Transformation

$$\mathbf{\Phi}_1(\mathbf{x}) = (x_1, x_2)$$
$$\mathbf{\Phi}_2(\mathbf{x}) = (x_1, x_2, x_1^2, x_1x_2, x_2^2)$$
$$\mathbf{\Phi}_3(\mathbf{x}) = (x_1, x_2, x_1^2, x_1x_2, x_2^2, x_1^3, x_1^2x_2, x_1x_2^2, x_2^3)$$

Non-Linear Transformation

- Using the nonlinear transform with the optimal hyperplane using a transform $\mathbf{\Phi}$: $\mathbb{R}^d \to \mathbb{R}^{\tilde{d}}$:

$$\mathbf{z}_n = \mathbf{\Phi}(\mathbf{x}_n)$$

Non-Linear Transformation

- Using the nonlinear transform with the optimal hyperplane using a transform $\mathbf{\Phi}$: $\mathbb{R}^d \to \mathbb{R}^{\tilde{d}}$:

$$\mathbf{z}_n = \mathbf{\Phi}(\mathbf{x}_n)$$

- Solve the hard-margin SVM in the $\mathcal{Z}$-space $(\tilde{\mathbf{w}}^*, \tilde{b}^*)$:

$$\min_{\tilde{b}, \tilde{\mathbf{w}}} \quad \frac{1}{2}\tilde{\mathbf{w}}^T\tilde{\mathbf{w}}$$

$$\text{subject to: } y_n(\tilde{\mathbf{w}}^T\mathbf{z}_n + \tilde{b}) \geq 1, \forall n$$

Non-Linear Transformation

- Using the nonlinear transform with the optimal hyperplane using a transform $\mathbf{\Phi}$: $\mathbb{R}^d \to \mathbb{R}^{\tilde{d}}$:

$$\mathbf{z}_n = \mathbf{\Phi}(\mathbf{x}_n)$$

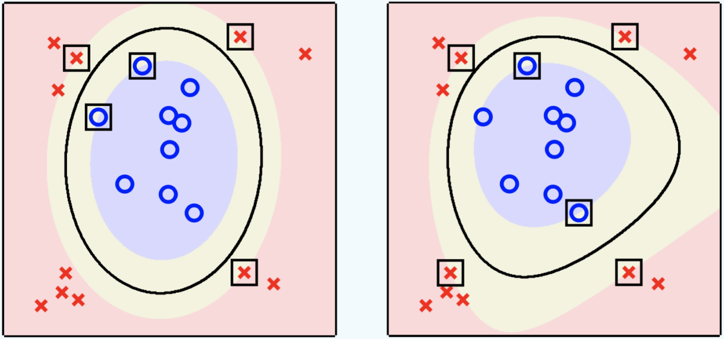
- Solve the hard-margin SVM in the $\mathcal{Z}$-space $(\tilde{\mathbf{w}}^*, \tilde{b}^*)$:

$$\min_{\tilde{b}, \tilde{\mathbf{w}}} \quad \frac{1}{2}\tilde{\mathbf{w}}^T \tilde{\mathbf{w}}$$
$$\text{subject to: } y_n(\tilde{\mathbf{w}}^T \mathbf{z}_n + \tilde{b}) \geq 1, \forall n$$

- Final hypothesis:

$$g(\mathbf{x}) = \text{sign}(\tilde{\mathbf{w}}^{*T} \mathbf{\Phi}(\mathbf{x}) + \tilde{b}^*)$$

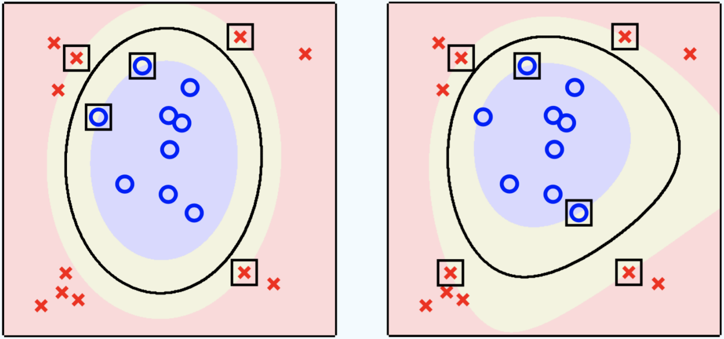
SVM and non-linear transformation



(a) SVM with Φ_2 (b) SVM with Φ_3

The margin is shaded in yellow, and the support vectors are boxed.

SVM and non-linear transformation

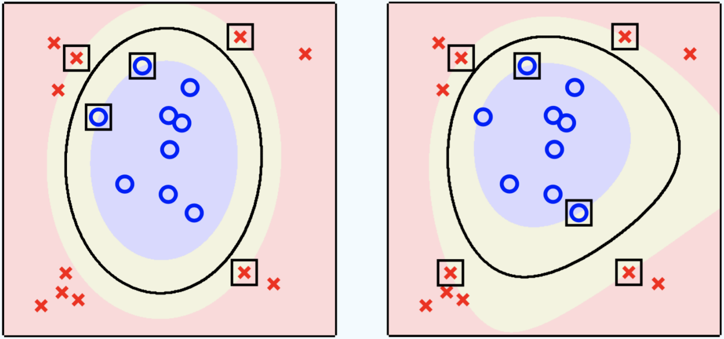


(a) SVM with Φ_2 (b) SVM with Φ_3

The margin is shaded in yellow, and the support vectors are boxed.

- For $\mathbf{\Phi}_2$, $\tilde{d}_2 = 5$ and for $\mathbf{\Phi}_3$, $\tilde{d}_3 = 9$

SVM and non-linear transformation



(a) SVM with Φ_2 (b) SVM with Φ_3

The margin is shaded in yellow, and the support vectors are boxed.

- For $\mathbf{\Phi}_2$, $\tilde{d}_2 = 5$ and for $\mathbf{\Phi}_3$, $\tilde{d}_3 = 9$
- $\tilde{d}_3$ is nearly double $\tilde{d}_2$, yet the resulting SVM separator is not severely overfitting with $\mathbf{\Phi}_3$ (regularization?).

Common SVM Basis Functions

- $\mathbf{z}_k$ = polynomial terms of $\mathbf{x}_k$ of degree 1 to q
- $\mathbf{z}_k$ = radial basis function of $\mathbf{x}_k$

$$\mathbf{z}_k(j) = \phi_j(\mathbf{x}_k) = \exp(-|\mathbf{x}_k - \mathbf{c}_j|^2/\sigma^2)$$

- $\mathbf{z}_k$ = sigmoid functions of $\mathbf{x}_k$

Quadratic Basis Functions

$$\mathbf{\Phi}(\mathbf{x}) = \begin{bmatrix} 1 \\ \sqrt{2}x_1 \\ \vdots \\ \sqrt{2}x_d \\ x_1^2 \\ \vdots \\ x_d^2 \\ \sqrt{2}x_1x_2 \\ \vdots \\ \sqrt{2}x_1x_d \\ \sqrt{2}x_2x_3 \\ \vdots \\ \sqrt{2}x_{d-1}x_d \end{bmatrix}$$

- Including Constant Term, Linear Terms, Pure Quadratic Terms, Quadratic Cross-Terms
- The number of terms is approximately $d^2/2$.
- You may be wondering what those $\sqrt{2}$s are doing. You'll find out why they're there soon.

Dual SVM: Non-linear Transformation

$$\max_{\boldsymbol{\alpha} \in \mathbb{R}^N} \sum_{n=1}^{N} \boldsymbol{\alpha}_n - \frac{1}{2} \sum_{m=1}^{N} \sum_{n=1}^{N} \boldsymbol{\alpha}_n \boldsymbol{\alpha}_m y_n y_m \boldsymbol{\Phi}(\mathbf{x}_n)^T \boldsymbol{\Phi}(\mathbf{x}_m)$$

$$\text{subject to } \sum_{n=1}^{N} y_n \boldsymbol{\alpha}_n = 0, \boldsymbol{\alpha}_n \geq 0, \forall n$$

$$\mathbf{w}^* = \sum_{n=1}^{N} y_n \boldsymbol{\alpha}_n^* \Phi(\mathbf{x}_n)$$

- Need to prepare a matrix Q, $Q_{nm} = y_n y_m \boldsymbol{\Phi}(\mathbf{x}_n)^T \boldsymbol{\Phi}(\mathbf{x}_m)$
- Cost?

Dual SVM: Non-linear Transformation

$$\max_{\boldsymbol{\alpha} \in \mathbb{R}^N} \sum_{n=1}^{N} \boldsymbol{\alpha}_n - \frac{1}{2} \sum_{m=1}^{N} \sum_{n=1}^{N} \boldsymbol{\alpha}_n \boldsymbol{\alpha}_m y_n y_m \boldsymbol{\Phi}(\mathbf{x}_n)^T \boldsymbol{\Phi}(\mathbf{x}_m)$$

$$\text{subject to } \sum_{n=1}^{N} y_n \boldsymbol{\alpha}_n = 0, \boldsymbol{\alpha}_n \geq 0, \forall n$$

$$\mathbf{w}^* = \sum_{n=1}^{N} y_n \boldsymbol{\alpha}_n^* \Phi(\mathbf{x}_n)$$

- Need to prepare a matrix Q, $Q_{nm} = y_n y_m \boldsymbol{\Phi}(\mathbf{x}_n)^T \boldsymbol{\Phi}(\mathbf{x}_m)$
- Cost?
  - We must do $N^2/2$ dot products to get this matrix ready.
  - Each dot product requires $d^2/2$ additions and multiplications, The whole thing costs $N^2 d^2/4$.

Quadratic Dot Products

$$\mathbf{\Phi}(\mathbf{a})^T\mathbf{\Phi}(\mathbf{b}) = \begin{bmatrix} 1 \\ \sqrt{2}a_1 \\ \vdots \\ \sqrt{2}a_m \\ a_1^2 \\ \vdots \\ a_m^2 \\ \sqrt{2}a_1a_2 \\ \vdots \\ \sqrt{2}a_1a_d \\ \sqrt{2}a_2a_3 \\ \vdots \\ \sqrt{2}a_{d-1}a_d \end{bmatrix}^T \begin{bmatrix} 1 \\ \sqrt{2}b_1 \\ \vdots \\ \sqrt{2}b_d \\ b_1^2 \\ \vdots \\ b_d^2 \\ \sqrt{2}b_1b_2 \\ \vdots \\ \sqrt{2}b_1b_d \\ \sqrt{2}b_2b_3 \\ \vdots \\ \sqrt{2}b_{d-1}b_d \end{bmatrix}$$

- Constant Term 1
- Linear Terms

$$\sum_{i=1}^{d} 2a_ib_i$$

- Pure Quadratic Terms

$$\sum_{i=1}^{d} a_i^2b_i^2$$

- Quadratic Cross-Terms

$$\sum_{i=1}^{d}\sum_{j=i+1}^{d} 2a_ia_jb_ib_j$$

Quadratic Dot Product

- Does $\mathbf{\Phi}(\mathbf{a})^T\mathbf{\Phi}(\mathbf{b})$ look familiar?

$$\mathbf{\Phi}(\mathbf{a})^T\mathbf{\Phi}(\mathbf{b}) = 1 + 2\sum_{i=1}^{d} a_i b_i + \sum_{i=1}^{d} a_i^2 b_i^2 + \sum_{i=1}^{d}\sum_{j=i+1}^{d} 2a_i a_j b_i b_j$$

Quadratic Dot Product

- Does $\mathbf{\Phi}(\mathbf{a})^T\mathbf{\Phi}(\mathbf{b})$ look familiar?

$$\mathbf{\Phi}(\mathbf{a})^T\mathbf{\Phi}(\mathbf{b}) = 1 + 2\sum_{i=1}^{d} a_i b_i + \sum_{i=1}^{d} a_i^2 b_i^2 + \sum_{i=1}^{d}\sum_{j=i+1}^{d} 2a_i a_j b_i b_j$$

- Try this: $(\boldsymbol{a}^T\boldsymbol{b} + 1)^2$

Quadratic Dot Product

- Does $\mathbf{\Phi}(\mathbf{a})^T \mathbf{\Phi}(\mathbf{b})$ look familiar?

$$\mathbf{\Phi}(\mathbf{a})^T \mathbf{\Phi}(\mathbf{b}) = 1 + 2\sum_{i=1}^{d} a_i b_i + \sum_{i=1}^{d} a_i^2 b_i^2 + \sum_{i=1}^{d} \sum_{j=i+1}^{d} 2 a_i a_j b_i b_j$$

- Try this: $(\boldsymbol{a}^T \boldsymbol{b} + 1)^2$

$$\begin{aligned}
(\boldsymbol{a}^T \boldsymbol{b} + 1)^2 &= (\boldsymbol{a}^T \boldsymbol{b})^2 + 2\boldsymbol{a}^T \boldsymbol{b} + 1 \\
&= \left(\sum_{i=1}^{d} a_i b_i\right)^2 + 2\sum_{i=1}^{d} a_i b_i + 1 \\
&= \sum_{i=1}^{d} \sum_{j=1}^{d} a_i b_i a_j b_j + 2\sum_{i=1}^{d} a_i b_i + 1 \\
&= \sum_{i=1}^{d} a_i^2 b_i^2 + 2\sum_{i=1}^{d} \sum_{j=i+1}^{d} a_i a_j b_i b_j + 2\sum_{i=1}^{d} a_i b_i + 1
\end{aligned}$$

Quadratic Dot Product

- Does $\mathbf{\Phi}(\mathbf{a})^T\mathbf{\Phi}(\mathbf{b})$ look familiar?

$$\mathbf{\Phi}(\mathbf{a})^T\mathbf{\Phi}(\mathbf{b}) = 1 + 2\sum_{i=1}^{d} a_i b_i + \sum_{i=1}^{d} a_i^2 b_i^2 + \sum_{i=1}^{d}\sum_{j=i+1}^{d} 2a_i a_j b_i b_j$$

- Try this: $(\boldsymbol{a}^T\boldsymbol{b}+1)^2$

$$\begin{aligned}(\boldsymbol{a}^T\boldsymbol{b}+1)^2 &= (\boldsymbol{a}^T\boldsymbol{b})^2 + 2\boldsymbol{a}^T\boldsymbol{b} + 1 \\ &= \left(\sum_{i=1}^{d} a_i b_i\right)^2 + 2\sum_{i=1}^{d} a_i b_i + 1 \\ &= \sum_{i=1}^{d}\sum_{j=1}^{d} a_i b_i a_j b_j + 2\sum_{i=1}^{d} a_i b_i + 1 \\ &= \sum_{i=1}^{d} a_i^2 b_i^2 + 2\sum_{i=1}^{d}\sum_{j=i+1}^{d} a_i a_j b_i b_j + 2\sum_{i=1}^{d} a_i b_i + 1\end{aligned}$$

- They're the same! And this is only $O(d)$ to compute!

Dual SVM: Non-linear Transformation

$$\max_{\boldsymbol{\alpha}\in\mathbb{R}^N} \sum_{n=1}^{N} \boldsymbol{\alpha}_n - \frac{1}{2}\sum_{m=1}^{N}\sum_{n=1}^{N} \boldsymbol{\alpha}_n \boldsymbol{\alpha}_m y_n y_m \boldsymbol{\Phi}(\mathbf{x}_n)^T \boldsymbol{\Phi}(\mathbf{x}_m)$$

$$\text{subject to } \sum_{n=1}^{N} y_n \boldsymbol{\alpha}_n = 0, \boldsymbol{\alpha}_n \geq 0, \forall n$$

$$\mathbf{w}^* = \sum_{n=1}^{N} y_n \boldsymbol{\alpha}_n^* \Phi(\mathbf{x}_n)$$

- Need to prepare a matrix Q, $Q_{nm} = y_n y_m \boldsymbol{\Phi}(\mathbf{x}_n)^T \boldsymbol{\Phi}(\mathbf{x}_m)$
- Cost?
  - We must do $N^2/2$ dot products to get this matrix ready.
  - Each dot product requires d additions and multiplications.

Higher Order Polynomials

	$\mathbf{\Phi}(\mathbf{x})$	Cost	100dim
Quadratic	$d^2/2$ terms	$d^2N^2/4$	$2.5kN^2$
Cubic	$d^3/6$ terms	$d^3N^2/12$	$83kN^2$
Quartic	$d^4/24$ terms	$d^4N^2/48$	$1.96mN^2$

	$\Phi(\mathbf{a})^T\Phi(\mathbf{b})$	Cost	100dim
Quadratic	$(\mathbf{a}^T\mathbf{b}+1)^2$	$dN^2/2$	$50N^2$
Cubic	$(\mathbf{a}^T\mathbf{b}+1)^3$	$dN^2/2$	$50N^2$
Quartic	$(\mathbf{a}^T\mathbf{b}+1)^4$	$dN^2/2$	$50N^2$

Dual SVM with Quintic basis functions

$$\max_{\boldsymbol{\alpha}\in\mathbb{R}^N} \sum_{n=1}^{N} \boldsymbol{\alpha}_n - \frac{1}{2}\sum_{m=1}^{N}\sum_{n=1}^{N} \boldsymbol{\alpha}_n\boldsymbol{\alpha}_m y_n y_m \underbrace{\boldsymbol{\Phi}(\mathbf{x}_n)^T\boldsymbol{\Phi}(\mathbf{x}_m)}_{(\mathbf{x}_n^T\mathbf{x}_m+1)^5}$$

subject to $\sum_{n=1}^{N} y_n\boldsymbol{\alpha}_n = 0, \boldsymbol{\alpha}_n \geq 0, \forall n$

Classification:

$$g(\mathbf{x}) = \mathrm{sign}(\mathbf{w}^{*T}\boldsymbol{\Phi}(\mathbf{x}) + b^*) = \mathrm{sign}\left(\sum_{\boldsymbol{\alpha}_n^*>0} y_n\boldsymbol{\alpha}_n^*\boldsymbol{\Phi}(\mathbf{x}_n)^T\boldsymbol{\Phi}(\mathbf{x}) + b^*\right)$$

$$= \mathrm{sign}\left(\sum_{\boldsymbol{\alpha}_n^*>0} y_n\boldsymbol{\alpha}_n^*(\mathbf{x}_n^T\mathbf{x}+1)^5 + b^*\right)$$

Section 4

Kernel Tricks

Dual SVM with general kernel functions

$$\max_{\boldsymbol{\alpha}\in\mathbb{R}^N} \sum_{n=1}^{N} \boldsymbol{\alpha}_n - \frac{1}{2}\sum_{m=1}^{N}\sum_{n=1}^{N} \boldsymbol{\alpha}_n \boldsymbol{\alpha}_m y_n y_m K(\mathbf{x}_n, \mathbf{x}_m)$$

$$\text{subject to } \sum_{n=1}^{N} y_n \boldsymbol{\alpha}_n = 0, \boldsymbol{\alpha}_n \geq 0, \forall n$$

Classification:

$$g(\mathbf{x}) = \text{sign}(\mathbf{w}^{*T}\boldsymbol{\Phi}(\mathbf{x}) + b^*) = \text{sign}\left(\sum\nolimits_{\boldsymbol{\alpha}_n^*>0} y_n \boldsymbol{\alpha}_n^* \boldsymbol{\Phi}(\mathbf{x}_n)^T \boldsymbol{\Phi}(\mathbf{x}) + b^*\right)$$

$$= \text{sign}\left(\sum\nolimits_{\boldsymbol{\alpha}_n^*>0} y_n \boldsymbol{\alpha}_n^* K(\mathbf{x}_n, \mathbf{x}_m) + b^*\right)$$

Kernel Tricks

- Replacing dot product with a kernel function

Kernel Tricks

- Replacing dot product with a kernel function
- Not all functions are kernel functions!
  - Need to be decomposable $K(\mathbf{a}, \mathbf{b}) = \mathbf{\Phi}(\mathbf{a})^T \mathbf{\Phi}(\mathbf{b})$
  - Could $K(\mathbf{a}, \mathbf{b}) = (\mathbf{a} - \mathbf{b})^3$ be a kernel function?
  - Could $K(\mathbf{a}, \mathbf{b}) = (\mathbf{a} - \mathbf{b})^4 - (\mathbf{a} + \mathbf{b})^2$ be a kernel function?

Kernel Tricks

- Replacing dot product with a kernel function
- Not all functions are kernel functions!
  - Need to be decomposable $K(\mathbf{a}, \mathbf{b}) = \mathbf{\Phi}(\mathbf{a})^T \mathbf{\Phi}(\mathbf{b})$
  - Could $K(\mathbf{a}, \mathbf{b}) = (\mathbf{a} - \mathbf{b})^3$ be a kernel function?
  - Could $K(\mathbf{a}, \mathbf{b}) = (\mathbf{a} - \mathbf{b})^4 - (\mathbf{a} + \mathbf{b})^2$ be a kernel function?
- Mercer's condition
  - To expand Kernel function $K(\mathbf{a}, \mathbf{b})$ into a dot product, i.e., $K(\mathbf{a}, \mathbf{b}) = \mathbf{\Phi}(\mathbf{a})^T \mathbf{\Phi}(\mathbf{b})$, $K(\mathbf{a}, \mathbf{b})$ has to be positive semi-definite function.

Kernel Tricks

- Replacing dot product with a kernel function
- Not all functions are kernel functions!
  - Need to be decomposable $K(\mathbf{a}, \mathbf{b}) = \mathbf{\Phi}(\mathbf{a})^T \mathbf{\Phi}(\mathbf{b})$
  - Could $K(\mathbf{a}, \mathbf{b}) = (\mathbf{a} - \mathbf{b})^3$ be a kernel function?
  - Could $K(\mathbf{a}, \mathbf{b}) = (\mathbf{a} - \mathbf{b})^4 - (\mathbf{a} + \mathbf{b})^2$ be a kernel function?
- Mercer's condition
  - To expand Kernel function $K(\mathbf{a}, \mathbf{b})$ into a dot product, i.e., $K(\mathbf{a}, \mathbf{b}) = \mathbf{\Phi}(\mathbf{a})^T \mathbf{\Phi}(\mathbf{b})$, $K(\mathbf{a}, \mathbf{b})$ has to be positive semi-definite function.
  - kernel matrix K is always symmetric PSD for any given $\mathbf{x}_1, \ldots, \mathbf{x}_N$.

Commonly Used SVM Kernel Functions

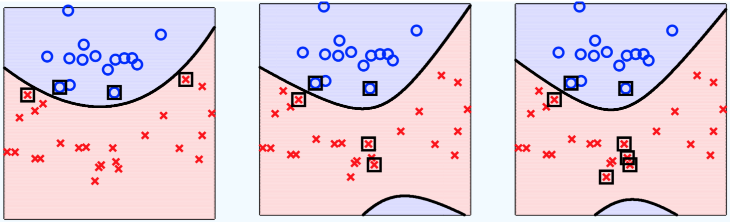
- $K(\mathbf{a}, \mathbf{b}) = (\boldsymbol{\alpha} \cdot \mathbf{a}^T\mathbf{b} + \boldsymbol{\beta})^Q$ is an example of a SVM kernel function.
- Beyond polynomials there are other very high dimensional basis functions that can be made practical by finding the right Kernel Function
  - Radial-basis style kernel (RBF)/Gaussian kernel function

$$K(\mathbf{a}, \mathbf{b}) = \exp\left(-\gamma\|\mathbf{a} - \mathbf{b}\|^2\right)$$

  - Sigmoid functions

2nd Order Polynomial Kernel

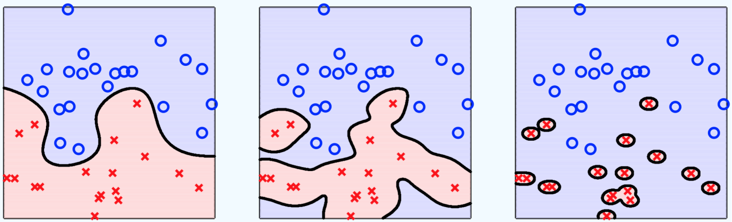
$$K(\mathbf{a}, \mathbf{b}) = (\boldsymbol{\alpha} \cdot \mathbf{a}^T \mathbf{b} + \boldsymbol{\beta})^2$$



$(1 + 0.001\mathbf{x}^{\mathrm{T}}\mathbf{x}')^2$ | $1 + (\mathbf{x}^{\mathrm{T}}\mathbf{x}') + (\mathbf{x}^{\mathrm{T}}\mathbf{x}')^2$ | $(1 + 1000\mathbf{x}^{\mathrm{T}}\mathbf{x}')^2$

Gaussian Kernels

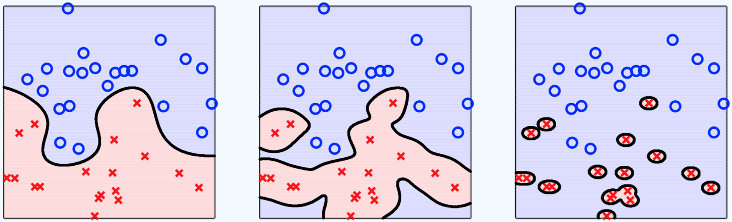
$$K(\mathbf{a}, \mathbf{b}) = \exp\left(-\gamma \|\mathbf{a} - \mathbf{b}\|^2\right)$$



$\exp(-1\|\mathbf{x} - \mathbf{x}'\|^2)$ | $\exp(-10\|\mathbf{x} - \mathbf{x}'\|^2)$ | $\exp(-100\|\mathbf{x} - \mathbf{x}'\|^2)$

Gaussian Kernels

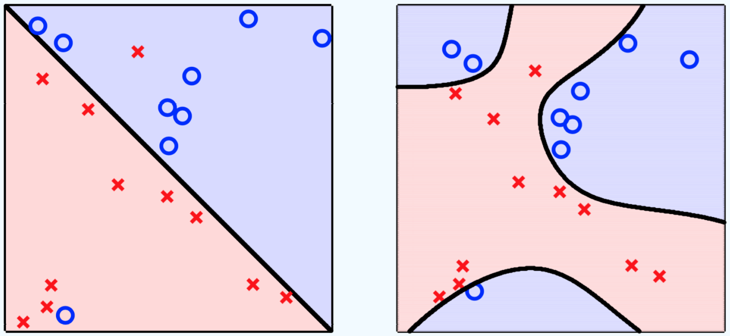
$$K(\mathbf{a}, \mathbf{b}) = \exp\left(-\gamma \|\mathbf{a} - \mathbf{b}\|^2\right)$$



$\exp(-1\|\mathbf{x} - \mathbf{x}'\|^2)$ $\exp(-10\|\mathbf{x} - \mathbf{x}'\|^2)$ $\exp(-100\|\mathbf{x} - \mathbf{x}'\|^2)$

When γ is large, we clearly see that even the protection of a large margin cannot suppress overfitting. However, for a reasonably small γ, the sophisticated boundary discovered by SVM with the Gaussian-RBF kernel looks quite good.

Gaussian Kernels



(a) linear classifier (b) Gaussian-RBF kernel

For (a) a noisy data set that linear classifier appears to work quite well, (b) using the Gaussian-RBF kernel with the hard-margin SVM leads to overfitting.

Section 5

Soft-margin SVM

From hard-margin to soft-margin

- When there are outliers, hard margin SVM + Gaussian-RBF kernel result in an unnecessarily complicated decision boundary that overfits the training noise.

From hard-margin to soft-margin

- When there are outliers, hard margin SVM + Gaussian-RBF kernel result in an unnecessarily complicated decision boundary that overfits the training noise.
- Remedy: a soft formulation that allows small violation of the margins or even some classification errors.

From hard-margin to soft-margin

- When there are outliers, hard margin SVM + Gaussian-RBF kernel result in an unnecessarily complicated decision boundary that overfits the training noise.
- Remedy: a soft formulation that allows small violation of the margins or even some classification errors.
- Soft-margin: margin violation $\varepsilon_n \geq 0$ for each data point $(\mathbf{x}_n, y_n)$ and require that

$$y_n(\mathbf{w}^T\mathbf{x}_n + b) \geq 1 - \varepsilon_n$$

  - ε_n captures by how much $(\mathbf{x}_n, y_n)$ fails to be separated.

Soft-Margin SVM

We modify the hard-margin SVM to the soft-margin SVM by allowing margin violations but adding a penalty term to discourage large violations:

$$\min_{b,\mathbf{w},\boldsymbol{\varepsilon}} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{n=1}^{N}\varepsilon_n$$

$$\text{subject to: } y_n(\mathbf{w}^T\mathbf{x}_n + b) \geq 1 - \varepsilon_n \text{ for } n = 1, \ldots, N$$

$$\varepsilon_n \geq 0, \text{ for } n = 1, \ldots, N$$

The meaning of C?

Soft-Margin SVM

We modify the hard-margin SVM to the soft-margin SVM by allowing margin violations but adding a penalty term to discourage large violations:

$$\min_{b,\mathbf{w},\boldsymbol{\varepsilon}} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{n=1}^{N}\varepsilon_n$$

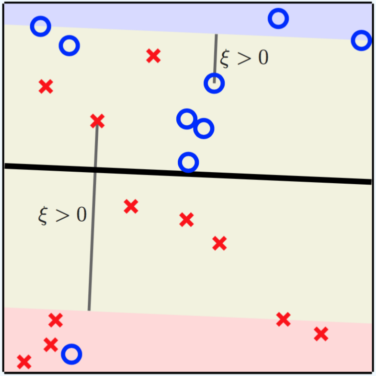
$$\text{subject to: } y_n(\mathbf{w}^T\mathbf{x}_n + b) \geq 1 - \varepsilon_n \text{ for } n = 1, \ldots, N$$

$$\varepsilon_n \geq 0, \text{ for } n = 1, \ldots, N$$

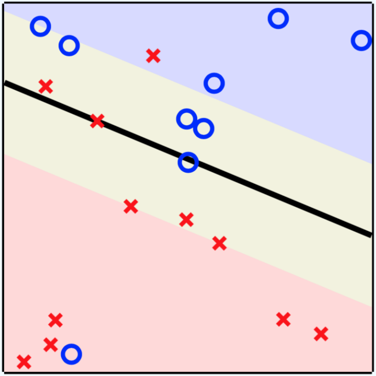
The meaning of C?

- When C is large, it means we care more about violating the margin, which gets us closer to the hard-margin SVM.
- When C is small, on the other hand, we care less about violating the margin.

Soft Margin Example



(a) $C = 1$



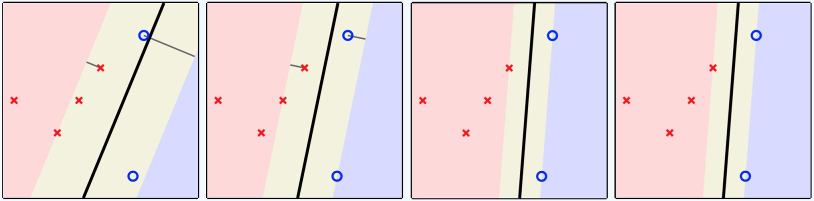
(b) $C = 500$

Soft Margin and Hard Margin

$$\min_{b,\mathbf{w},\boldsymbol{\varepsilon}} \quad \underbrace{\frac{1}{2}\mathbf{w}^T\mathbf{w}}_{\text{margin}} + \underbrace{C\sum_{n=1}^{N}\varepsilon_n}_{\text{error tolerance}}$$

subject to: $y_n(\mathbf{w}^T\mathbf{x}_n + b) \geq 1 - \varepsilon_n, \varepsilon_n \geq 0, \forall N$

Soft-margin			**Hard-margin**
small C	medium C	large C	



The Hinge Loss

- We have $\varepsilon_n \geq 0$, and that
  $y_n(\mathbf{w}^T\mathbf{x}_n + b) \geq 1 - \varepsilon_n \Rightarrow \varepsilon_n \geq 1 - y_n(\mathbf{w}^T\mathbf{x}_n + b)$

The Hinge Loss

- We have $\varepsilon_n \geq 0$, and that $y_n(\mathbf{w}^T\mathbf{x}_n + b) \geq 1 - \varepsilon_n \Rightarrow \varepsilon_n \geq 1 - y_n(\mathbf{w}^T\mathbf{x}_n + b)$
- The SVM loss (aka. Hinge Loss) function

$$E_{\mathsf{SVM}}(b, \mathbf{w}) = \frac{1}{N}\sum_{n=1}^{N} \max(1 - y_n(\mathbf{w}^T\mathbf{x}_n + b), 0)$$

The Hinge Loss

- We have $\varepsilon_n \geq 0$, and that $y_n(\mathbf{w}^T\mathbf{x}_n + b) \geq 1 - \varepsilon_n \Rightarrow \varepsilon_n \geq 1 - y_n(\mathbf{w}^T\mathbf{x}_n + b)$
- The SVM loss (aka. Hinge Loss) function

$$E_{\mathsf{SVM}}(b, \mathbf{w}) = \frac{1}{N}\sum_{n=1}^{N} \max(1 - y_n(\mathbf{w}^T\mathbf{x}_n + b), 0)$$

- The soft-margin SVM can be re-written as the following optimization problem:

$$\min_{b,\mathbf{w}} E_{\mathsf{SVM}}(b, \mathbf{w}) + \lambda \mathbf{w}^T\mathbf{w}$$