EE5907: Pattern Recognition
&
EE5026: Machine Learning for Data Analytics

Dr. WANG Si

Email: si.wang@nus.edu.sg

Office: E1a-04-01

Outlines

- Pattern Representation Learning
  - Unsupervised Representation Learning (week 7)
  - Supervised Representation Learning (week 8)
  - Unified Framework: Graph Embedding (week 8)
- Patter Recognition Models
  - Clustering (K-means, Agglomerative)(week 9)
  - Gaussian Mixture Model and Boosting (week 10)
  - Support Vector Machines (week 11)
- Deep Learning (week 12)
- Revision and Q&A (week 13)

Some slides are from Shuicheng Yan, Henry Lin

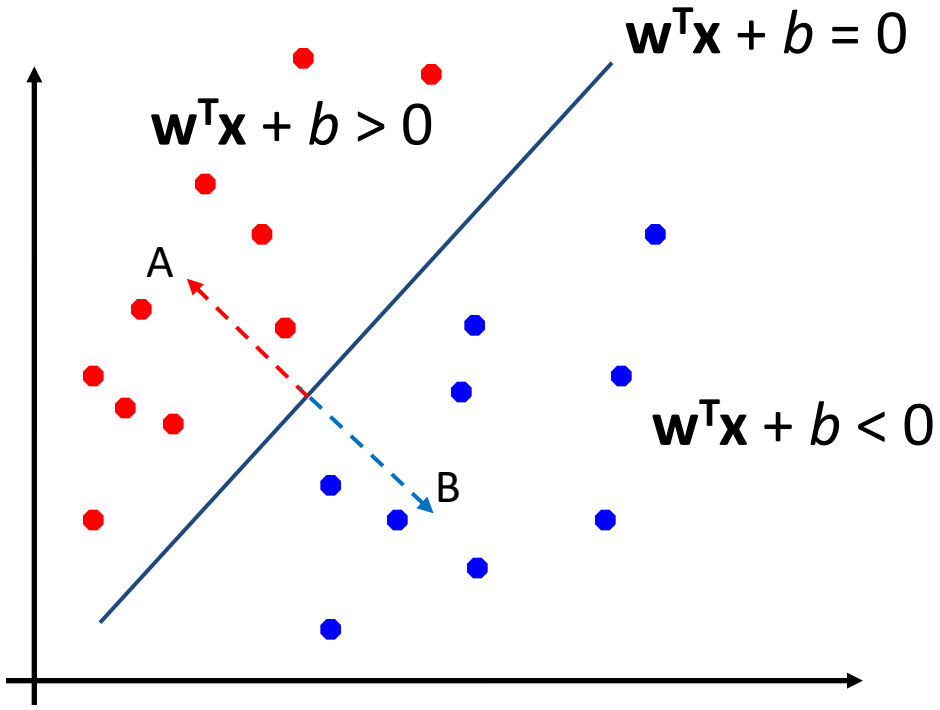
Support Vector Machine (SVM)

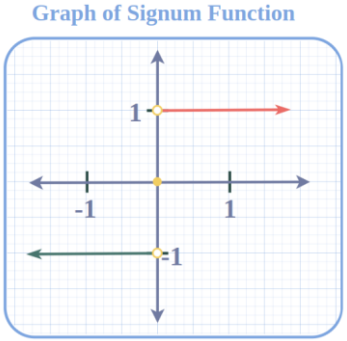
- We will discuss:
  - Hard-margin SVM (Primal and Dual formulations)
  - Soft-margin SVM (Primal and Dual formulations)
  - Kernel SVM
  - Multi-class SVM
- At the end of this lecture, you should be able to:
  - Understand different formulations for different SVMs
  - Elaborate the two tricks to deal with non linearly separable data
  - Apply SVM to the multi-class problem

Recap: Linear Classifier

Class labels:

- +1
- -1





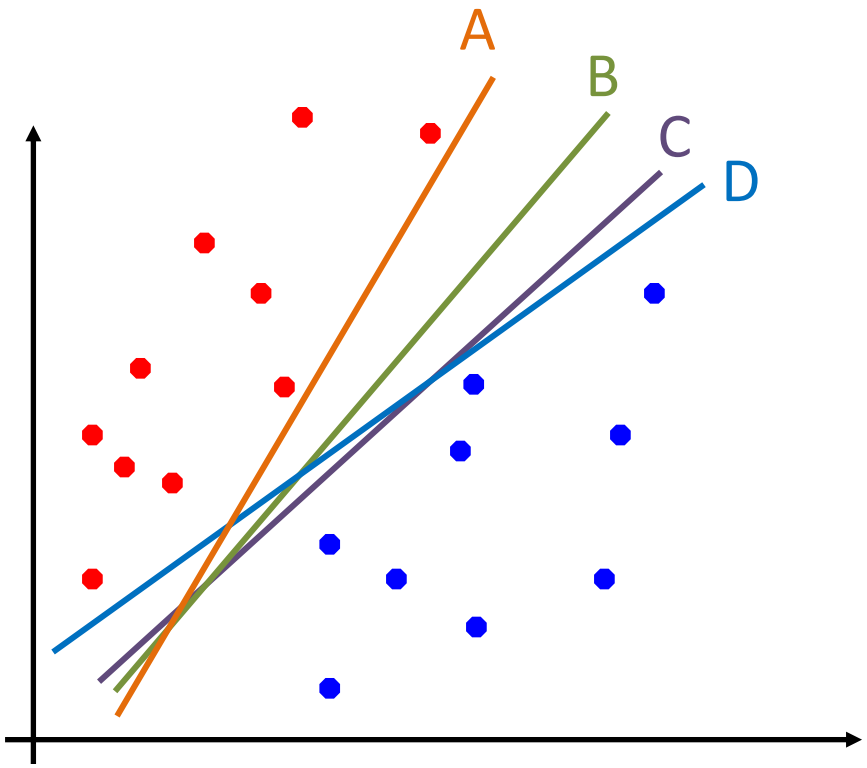
What is the direction of **w**?

$$f(\mathbf{x}) = \text{sign}(\mathbf{w}^T\mathbf{x} + b)$$

$$\text{sign}(z) = \begin{cases} -1, z < 0 \\ 0, z = 0 \\ +1, z > 0 \end{cases}$$

Recap: Linear Classifier

- Which one is optimal?

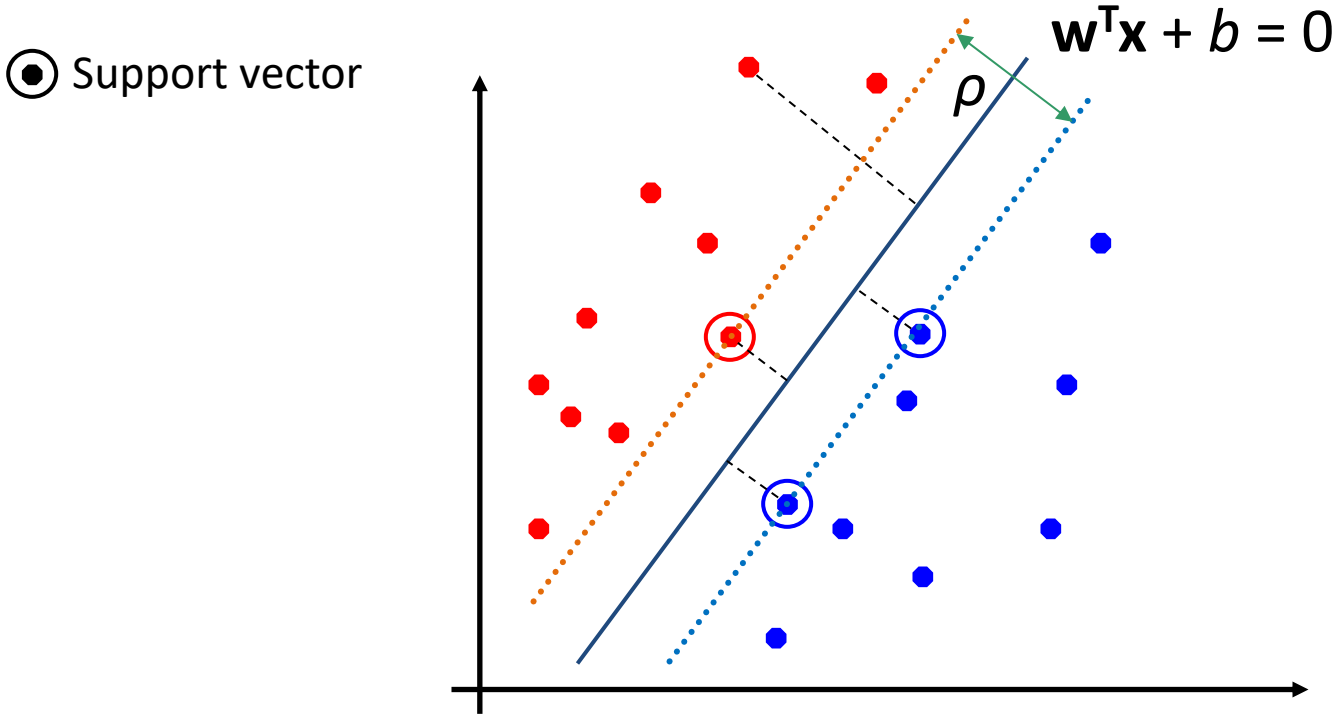


Support Vector Machine (SVM)

[Hard Margin]

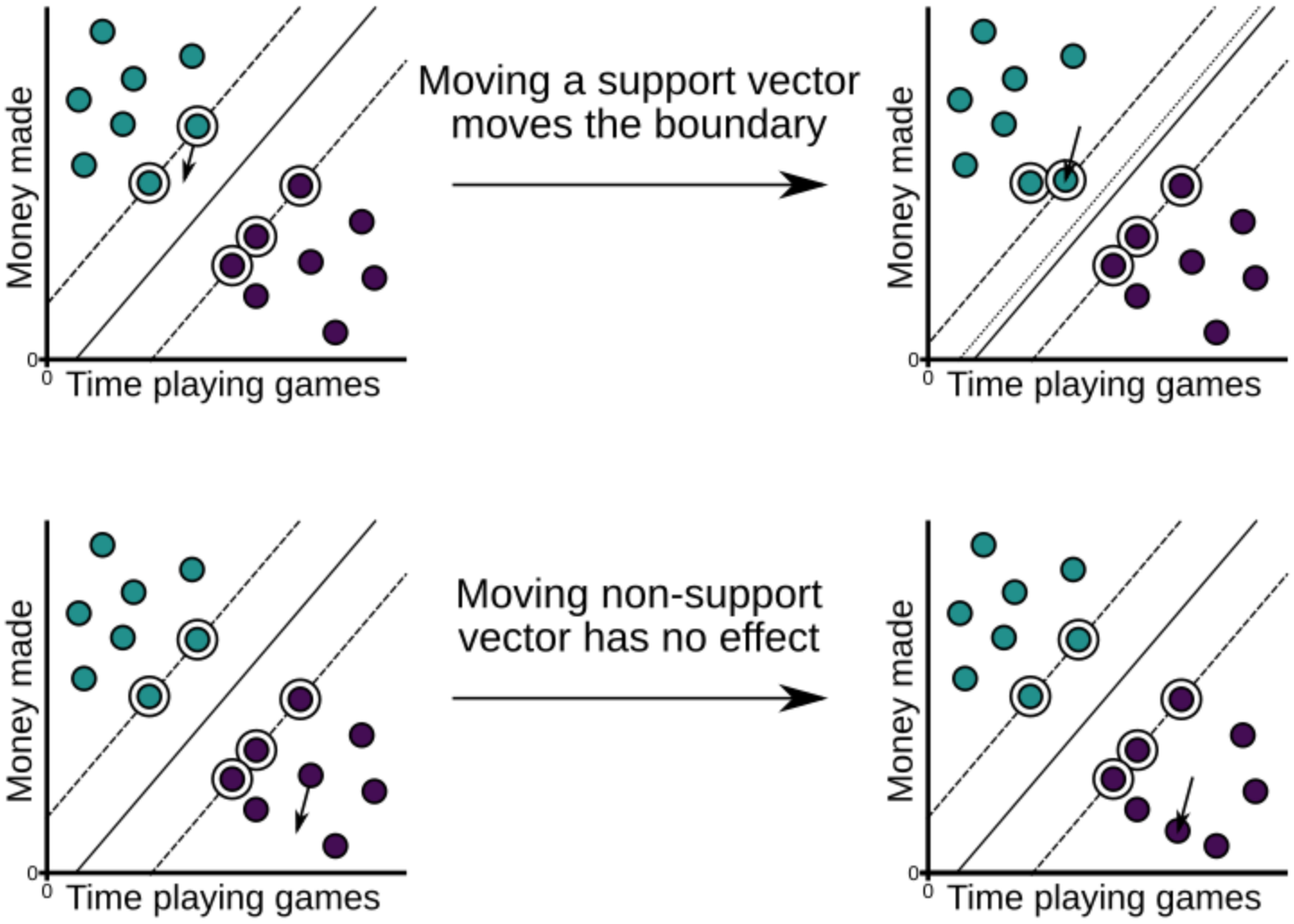
Hard-Margin Support Vector Machine (SVM)

- Maximize the **margin** around the decision boundary



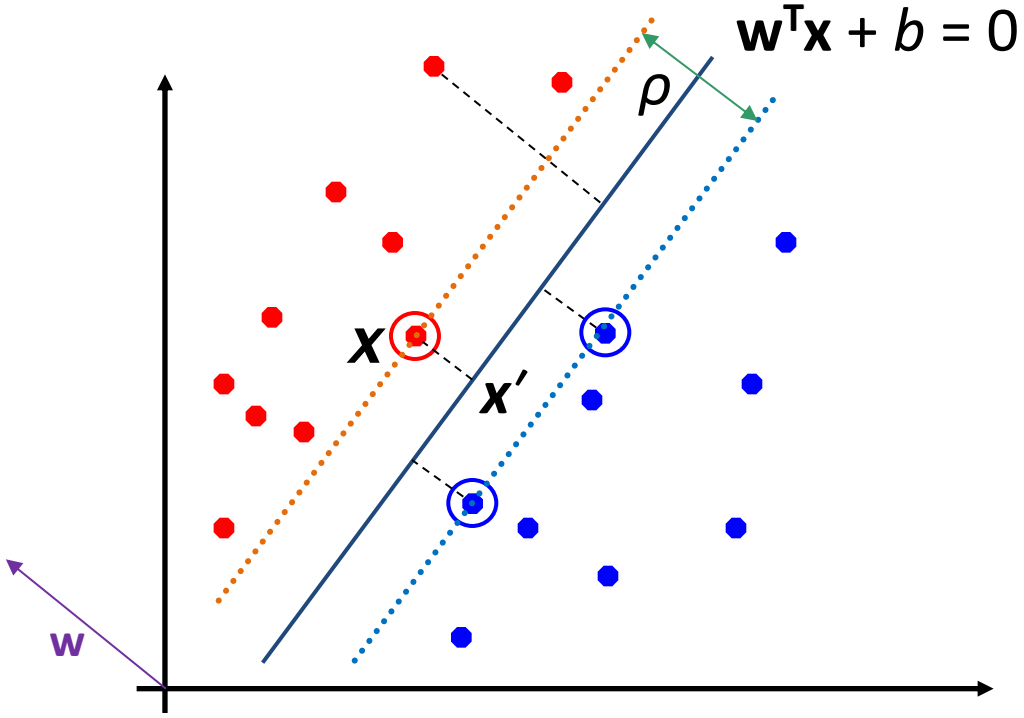
Hard-Margin Support Vector Machine (SVM)

- Move Support Vectors



Hard-Margin Support Vector Machine (SVM)

Equations for orange dashed line and blue dashed line?



Orange dashed line:

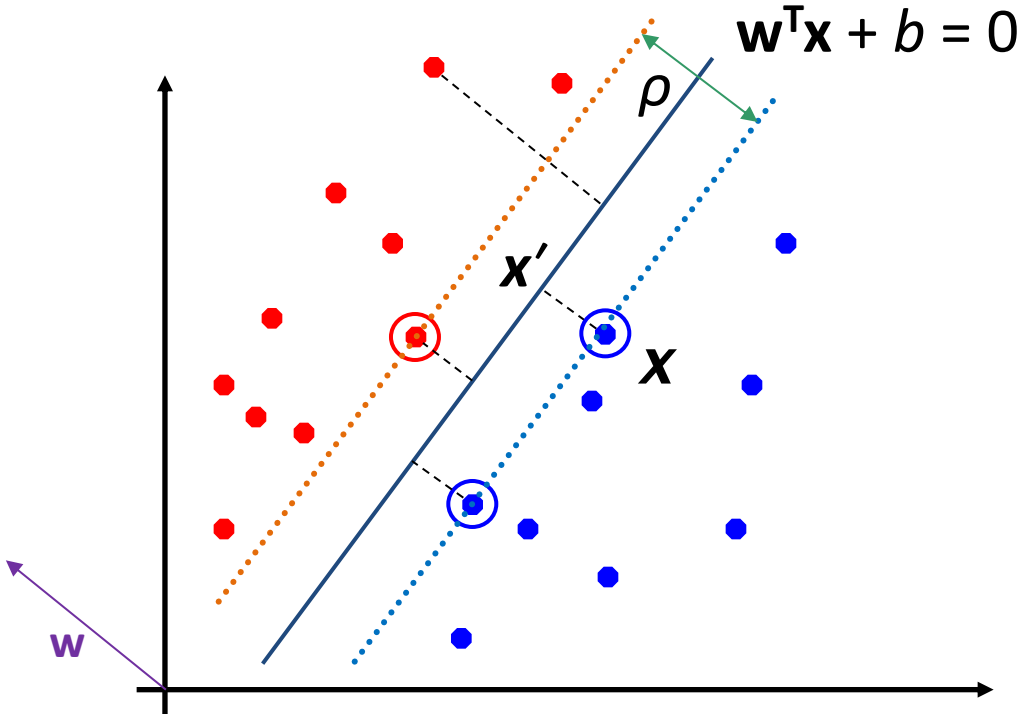
$$\begin{cases} \mathbf{w}^T\mathbf{x}' + b = 0 \\ \dfrac{\boldsymbol{x} - \boldsymbol{x}'}{\rho/2} = \dfrac{\boldsymbol{w}}{\|\boldsymbol{w}\|} \rightarrow \boldsymbol{x}' = \boldsymbol{x} - \dfrac{\rho \boldsymbol{w}}{2\|\boldsymbol{w}\|} \end{cases}$$

$$\boldsymbol{w}^T\left(\boldsymbol{x} - \frac{\rho \boldsymbol{w}}{2\|\boldsymbol{w}\|}\right) + b = 0$$

$$\boldsymbol{w}^T\boldsymbol{x} + b = \frac{\rho\|\boldsymbol{w}\|}{2}$$

Hard-Margin Support Vector Machine (SVM)

Equations for orange dashed line and blue dashed line?



Blue dashed line:

$$\begin{cases} \mathbf{w}^T\mathbf{x}' + b = 0 \\ \dfrac{\boldsymbol{x} - \boldsymbol{x}'}{\rho/2} = -\dfrac{\boldsymbol{w}}{\|\boldsymbol{w}\|} \rightarrow \boldsymbol{x}' = \boldsymbol{x} + \dfrac{\rho \boldsymbol{w}}{2\|\boldsymbol{w}\|} \end{cases}$$

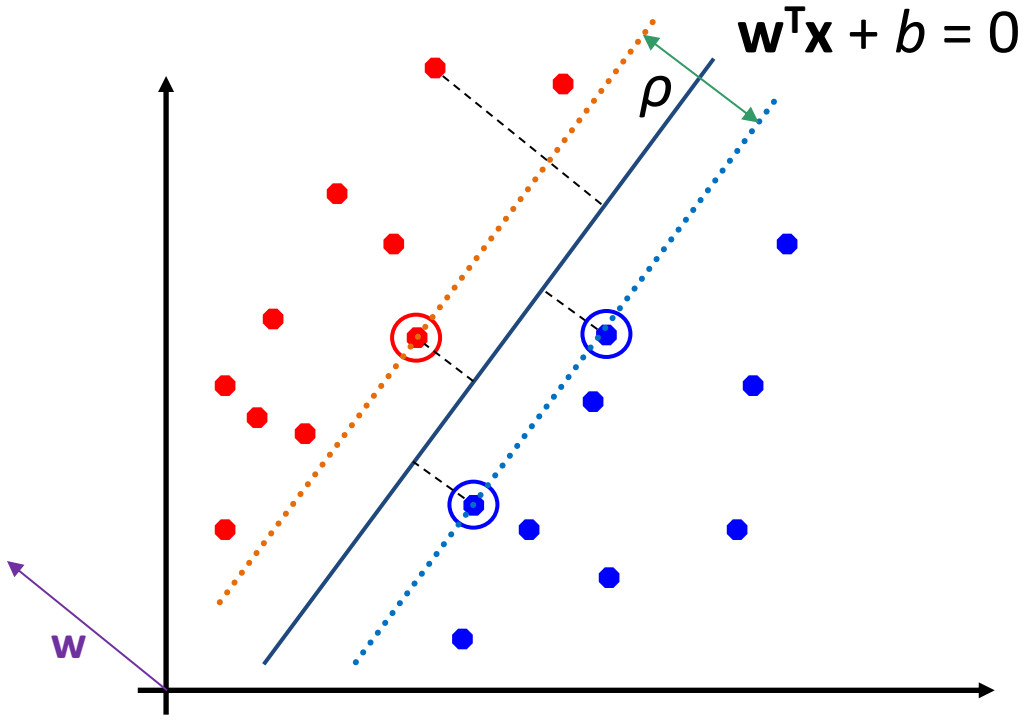
$$\boldsymbol{w}^T\left(\boldsymbol{x} + \frac{\rho \boldsymbol{w}}{2\|\boldsymbol{w}\|}\right) + b = 0$$

$$\boldsymbol{w}^T\boldsymbol{x} + b = -\frac{\rho\|\boldsymbol{w}\|}{2}$$

Hard-Margin Support Vector Machine (SVM)

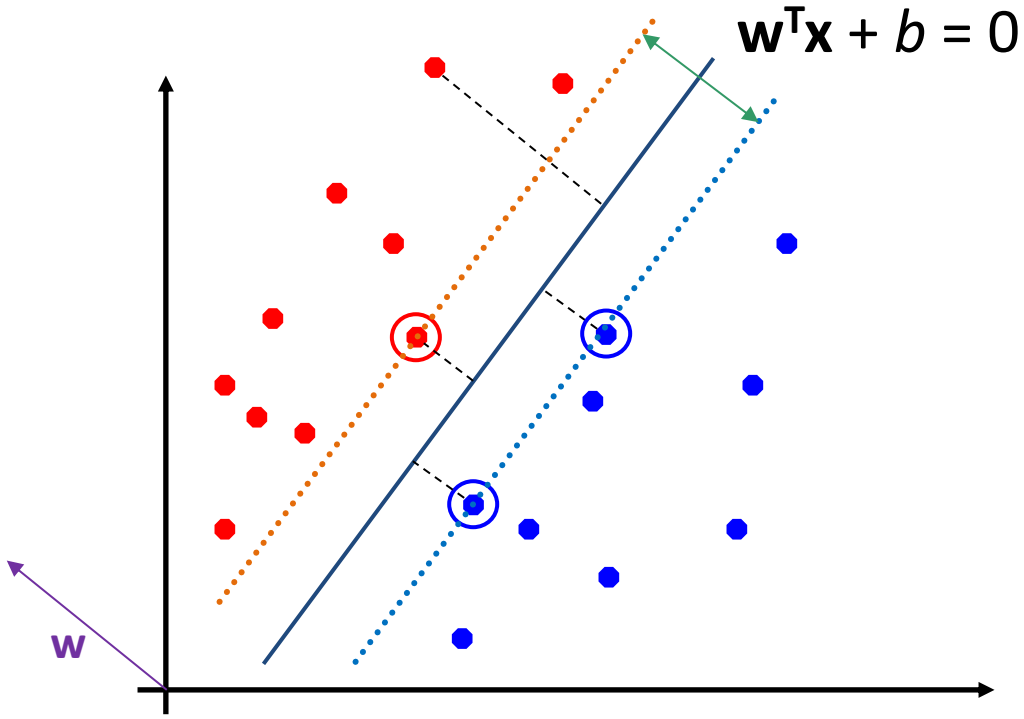
- Plus Plane: $\mathbf{w}^T\mathbf{x} + b = 1$
- Minus Plane: $\mathbf{w}^T\mathbf{x} + b = -1$

$$\rho = \frac{2}{\|w\|}$$



Hard-Margin Support Vector Machine (SVM)

- Objective: find **w**, b such that
  - ✓ All positive training points $(\boldsymbol{x}_i, y_i = 1)$ in the red zone ⇨ $\boldsymbol{w}^T\boldsymbol{x}_i + b \geq 1, if\, y_i = 1$
  - ✓ All negative training points $(\boldsymbol{x}_i, y_i = -1)$ in the blue zone ⇨ $\boldsymbol{w}^T\boldsymbol{x}_i + b \leq -1, if\, y_i = -1$
  - ✓ The margin ρ is maximized ⇨ $max\, \frac{2}{\|\boldsymbol{w}\|}$



Hard-Margin SVM Primal as Quadratic Program (QP)

- Objective: find **w**, b such that

$$\boldsymbol{w}^T\boldsymbol{x}_i + b \geq 1, if\, y_i = 1$$

$$\boldsymbol{w}^T\boldsymbol{x}_i + b \leq -1, if\, y_i = -1$$

$$\Rightarrow\ y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1$$

$$max\ \frac{2}{\|\boldsymbol{w}\|} \Rightarrow min\|\boldsymbol{w}\|, \text{ or } min\|\boldsymbol{w}\|^2, \text{ or } min\ \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w}$$

Reformulated as a quadratic optimization problem:

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

Subject to $y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1$, for all $(\boldsymbol{x}_i, y_i), i = 1 \ldots N$

Quadratic objective function

Linear constraints

Quadratic program

- QP solver libraries:

  cvxpy, *quadprog*, *OSQP*, *scipy.optimize*, *etc.*

Hard-Margin SVM Primal & Dual

Primal Problem

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

Subject to $y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1$, for all $(\boldsymbol{x}_i, y_i), i = 1 \ldots N$

d+1 variables

N constraints

Which one is better?

Dual Problem

$$\max_{\boldsymbol{\lambda}} -\frac{1}{2}\sum_{i,j} \lambda_i\lambda_j y_i y_j \boldsymbol{x}_i^T\boldsymbol{x}_j + \sum_i \lambda_i$$

s.t. $\lambda_i \geq 0, \sum_i \lambda_i y_i = 0, i = 1, \ldots, N$

N variables

N+1 constraints

λ_i is the Lagrange Multiplier associated with each constraint in the primal problem

Hard-Margin SVM Primal -> Dual

Primal Problem

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

Subject to $y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1$, for all $(\boldsymbol{x}_i, y_i), i = 1 \dots N$

- Inequality Constraints:

  Karush-Kuhn-Tucker (KKT) conditions

$$f(\boldsymbol{w}, b) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w},$$
$$g_i(\boldsymbol{w}, b) = y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) - 1$$

1) Define the Lagrangian function:

$$\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\lambda}) = f(\boldsymbol{w}, b) - \sum_i \lambda_i g_i(\boldsymbol{w}, b)\,, \text{where } \lambda_i \geq 0, \forall i$$

2) KKT conditions for optimality:

✓ Stationarity:

$$\begin{cases} \dfrac{\partial \mathcal{L}}{\partial \boldsymbol{w}} = \boldsymbol{0} \longrightarrow \boldsymbol{w} - \sum_i \lambda_i y_i \boldsymbol{x}_i = \boldsymbol{0} \longrightarrow \boldsymbol{w} = \sum_i \lambda_i y_i \boldsymbol{x}_i \\ \dfrac{\partial \mathcal{L}}{\partial b} = 0 \longrightarrow \sum_i \lambda_i y_i = 0 \end{cases}$$

Hard-Margin SVM Primal -> Dual

- Inequality Constraints:

  Karush-Kuhn-Tucker (KKT) conditions

$$f(\boldsymbol{w}, b) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w},$$
$$g_i(\boldsymbol{w}, b) = y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) - 1 \geq 0$$

1) Define the Lagrangian function:

$$\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\lambda}) = f(\boldsymbol{w}, b) - \sum_i \lambda_i g_i(\boldsymbol{w}, b)\,, \text{where } \lambda_i \geq 0, \forall i$$

2) KKT conditions for optimality:

✓ Stationarity: $\boldsymbol{w}^* = \sum_i \lambda_i y_i \boldsymbol{x}_i\,, \sum_i \lambda_i y_i = 0$

✓ Primal Feasibility: $g_i(\boldsymbol{w}, b) = y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) - 1 \geq 0, \forall i$

✓ Dual Feasibility: $\lambda_i \geq 0, \forall i$

✓ Complementary Slackness: $\lambda_i g_i(\boldsymbol{w}, b) = 0, \forall i$

If $g_i(\boldsymbol{w}, b) > 0$ (inactive), $\lambda_i = 0$

If $g_i(\boldsymbol{w}, b) = 0$ (active), $\lambda_i > 0$

λ_i only plays a role when a constraint is exactly binding

Hard-Margin SVM Primal -> Dual

$$\min_{w,b} \mathcal{L}(w, b, \lambda) = \min_{w,b} f(w, b) - \sum_i \lambda_i g_i(w, b)$$

$$= \min_{w,b} \frac{1}{2} w^T w - \sum_i \lambda_i (y_i (w^T x_i + b) - 1)$$

$$= \min_{w,b} \frac{1}{2} w^T w - \sum_i \lambda_i y_i w^T x_i - \sum_i \lambda_i y_i b + \sum_i \lambda_i$$

$$= \frac{1}{2} w^{*T} w^* - w^{*T} w^* + \sum_i \lambda_i$$

$$= -\frac{1}{2} \sum_i \lambda_i y_i x_i^T \sum_i \lambda_i y_i x_i + \sum_i \lambda_i$$

$$= -\frac{1}{2} \sum_{i,j} \lambda_i \lambda_j y_i y_j x_i^T x_j + \sum_i \lambda_i$$

$$f(w, b) = \frac{1}{2} w^T w,$$

$$g_i(w, b) = y_i (w^T x_i + b) - 1 \geq 0$$

$$\begin{cases} w^* = \sum_i \lambda_i y_i x_i \\ \sum_i \lambda_i y_i = 0 \end{cases}$$

Primal

$$\min_{w,b} \frac{1}{2} w^T w, s.t. y_i (w^T x_i + b) \geq 1$$

WHY?

Dual

$$\max_{\lambda} -\frac{1}{2} \sum_{i,j} \lambda_i \lambda_j y_i y_j x_i^T x_j + \sum_i \lambda_i$$

$$\text{s.t. } \lambda_i \geq 0, \sum_i \lambda_i y_i = 0$$

Lagrange Duality

Prove $\min_{\boldsymbol{w},b} f(\boldsymbol{w},b)\,, s.t.\, g_i(\boldsymbol{w},b) \geq 0$ $=$ $\max_{\boldsymbol{\lambda}} \min_{\boldsymbol{w},b} \mathcal{L}(\boldsymbol{w},b,\boldsymbol{\lambda})$ s.t. $\lambda_i \geq 0$

$$\mathcal{L}(\boldsymbol{w},b,\boldsymbol{\lambda}) = f(\boldsymbol{w},b) - \sum_i \lambda_i g_i(\boldsymbol{w},b)$$

$$f(\boldsymbol{w},b) \geq \mathcal{L}(\boldsymbol{w},b,\boldsymbol{\lambda})$$

$$l(\boldsymbol{\lambda}) = \min_{\boldsymbol{w},b} \mathcal{L}(\boldsymbol{w},b,\boldsymbol{\lambda})\,, s.t.\, \lambda_i \geq 0$$

$$p^* = \min_{\boldsymbol{w},b} f(\boldsymbol{w},b)\,, s.t.\, g_i(\boldsymbol{w},b) \geq 0$$

$$p^* \geq l(\boldsymbol{\lambda})$$

$$p^* = \max_{\boldsymbol{\lambda}} l(\boldsymbol{\lambda})$$

$$f(\boldsymbol{w},b) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w},$$

$$g_i(\boldsymbol{w},b) = y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) - 1$$

$$\boldsymbol{w}^* = \sum_i \lambda_i y_i \boldsymbol{x}_i$$

$$\sum_i \lambda_i y_i = 0$$

Lagrange Duality

Primal

Dual

$\min_{\boldsymbol{x}} f(\boldsymbol{x}), s.t. g_i(\boldsymbol{x}) \geq 0$ $\Leftrightarrow$ $\max_{\boldsymbol{\lambda}} \min_{\boldsymbol{x}} f(\boldsymbol{x}) - \sum_i \lambda_i g_i(\boldsymbol{x}), \text{s.t. } \lambda_i \geq 0$

$\min_{\boldsymbol{x}} f(\boldsymbol{x}), s.t. g_i(\boldsymbol{x}) \leq 0$ $\Leftrightarrow$

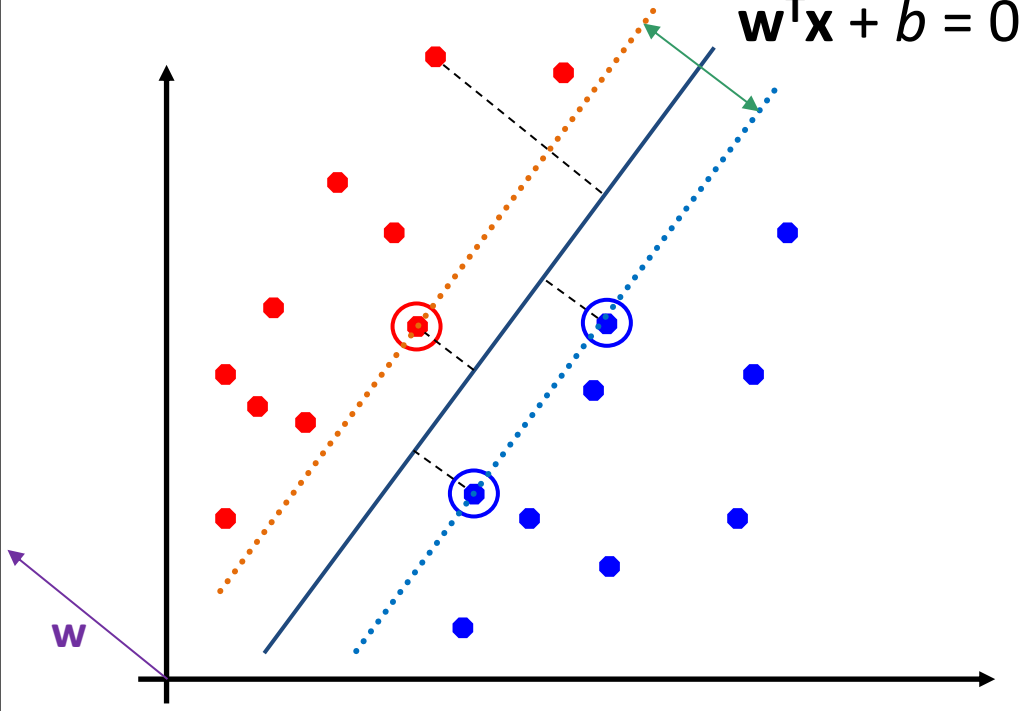
$\max_{\boldsymbol{x}} f(\boldsymbol{x}), s.t. g_i(\boldsymbol{x}) \geq 0$ $\Leftrightarrow$

$\max_{\boldsymbol{x}} f(\boldsymbol{x}), s.t. g_i(\boldsymbol{x}) \leq 0$ $\Leftrightarrow$

Lagrange Duality

Primal		Dual
$\min_{x} f(x), s.t.\ g_i(x) \geq 0$	$\equiv$	$\max_{\lambda} \min_{x} f(x) - \sum_i \lambda_i g_i(x)$, s.t. $\lambda_i \geq 0$
$\min_{x} f(x), s.t.\ g_i(x) \leq 0$	$\equiv$	$\max_{\lambda} \min_{x} f(x) + \sum_i \lambda_i g_i(x)$, s.t. $\lambda_i \geq 0$
$\max_{x} f(x), s.t.\ g_i(x) \geq 0$	$\equiv$	$\min_{\lambda} \max_{x} f(x) + \sum_i \lambda_i g_i(x)$, s.t. $\lambda_i \geq 0$
$\max_{x} f(x), s.t.\ g_i(x) \leq 0$	$\equiv$	$\min_{\lambda} \max_{x} f(x) - \sum_i \lambda_i g_i(x)$, s.t. $\lambda_i \geq 0$

Hard-Margin SVM: Decision Boundary



$$\begin{cases} \boldsymbol{w}^* = \sum_i \lambda_i y_i \boldsymbol{x}_i \\ \sum_i \lambda_i y_i = 0 \end{cases}$$

$$\boldsymbol{w}^{*T}\boldsymbol{x} + b^* = \sum_i \lambda_i y_i \boldsymbol{x}_i^T \boldsymbol{x} + b^* = 0$$

for any support vector $\boldsymbol{x}_s$ ($\lambda_i > 0$),

$$y_s(\boldsymbol{w}^{*T}\boldsymbol{x}_s + b^*) = 1$$
$$\downarrow$$
$$y_s^2(\boldsymbol{w}^{*T}\boldsymbol{x}_s + b^*) = y_s$$
$$\downarrow$$
$$b^* = y_s - \sum_i \lambda_i y_i \boldsymbol{x}_i^T \boldsymbol{x}_s$$

Quick Question

For SVM with the following objective function:

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

$$\text{s.t. } y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1, \forall i$$

or

$$\max_{\boldsymbol{\lambda}} -\frac{1}{2}\sum_{i,j} \lambda_i\lambda_j y_i y_j \boldsymbol{x}_i^T\boldsymbol{x}_j + \sum_i \lambda_i$$

$$\text{s.t. } \lambda_i \geq 0, \textstyle\sum_i \lambda_i y_i = 0, \forall i$$

Which of the following statements correctly describe *support vectors*?

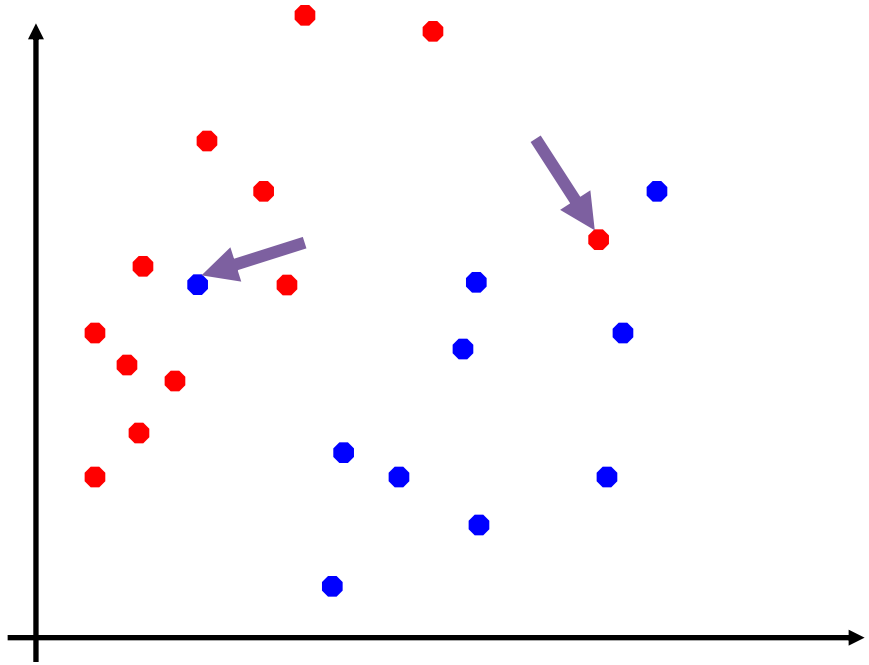
A. Data points that lie exactly on the margin boundaries.
B. Data points that have $\lambda_i > 0$.
C. Data points that determine the decision boundary
D. Data points inside the margin
E. Data points that are misclassified

SVM: linearly non-separable data

- What if the data is not linearly separable?

  Trick 1: Allow a few points to violate the margin (Soft-margin SVM)

  Trick 2: Map data to a higher dimensional space using a kernel function, do linear classification there (Kernel SVM)



Support Vector Machine (SVM)

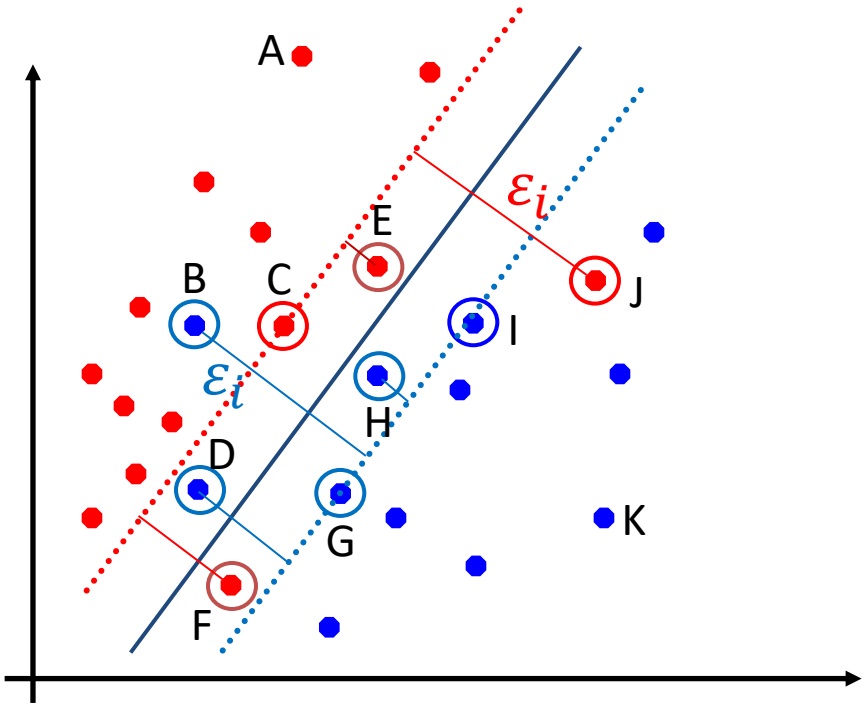
[Soft Margin]

Trick 1: Soft-margin SVM---Slack Variable

- Relax hard-margin constraints by introducing ***slack variables*** ε_i

$$y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1 - \varepsilon_i, \varepsilon_i \geq 0, \forall i$$

Measures how much a data point violates the margin



$if\ \varepsilon_i = 0$: $y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1$

Correctly classified, on or outside the margin

$if\ 0 < \varepsilon_i \leq 1$: $0 \leq y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) < 1$

Correctly classified, inside the margin

$if\ \varepsilon_i > 1$: $y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) < 0$

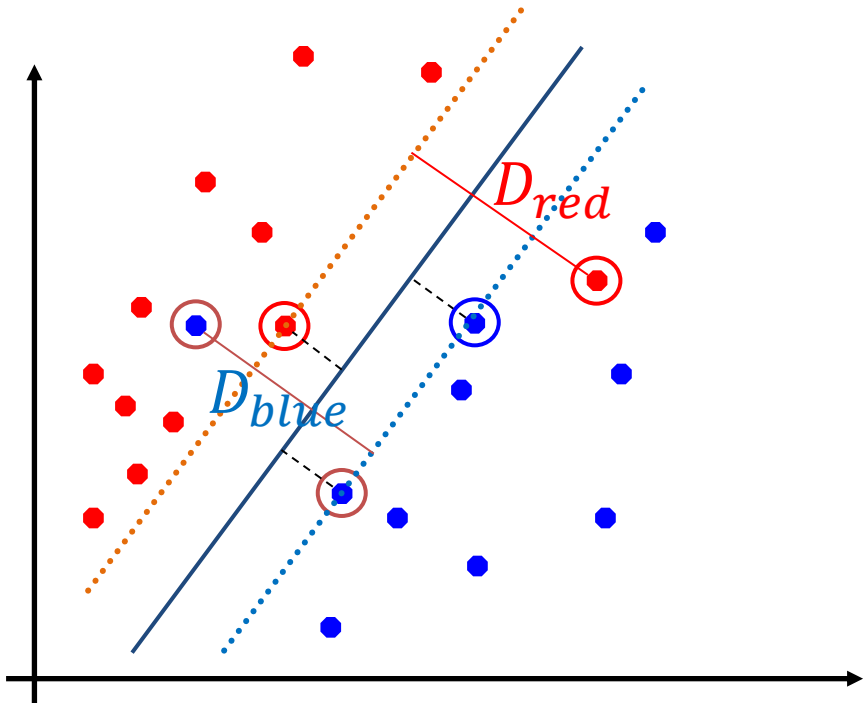
Misclassified, wrong side

ε_i is not the distance to the corresponding margin boundary!

Trick 1: Soft-margin SVM---Slack Variable

- ***Slack variables*** ε_i are not the distances to the corresponding margin boundary!

D=0 If the data points do not violate the margin



$if\ y_i = -1$ (blue):

$$\begin{cases} \mathbf{w}^T\mathbf{x}' + b = -1 \\ \dfrac{\boldsymbol{x}_i - \boldsymbol{x}'}{D_{blue}} = \dfrac{\boldsymbol{w}}{\|\boldsymbol{w}\|} \end{cases}$$

$if\ y_i = 1$ (red):

$$\begin{cases} \mathbf{w}^T\mathbf{x}' + b = 1 \\ \dfrac{\boldsymbol{x}_i - \boldsymbol{x}'}{D_{red}} = \dfrac{-\boldsymbol{w}}{\|\boldsymbol{w}\|} \end{cases}$$

$$\begin{cases} \mathbf{w}^T\mathbf{x}' + b = y_i \\ \dfrac{\boldsymbol{x}_i - \boldsymbol{x}'}{D} = \dfrac{-y_i\boldsymbol{w}}{\|\boldsymbol{w}\|} \end{cases} \rightarrow \boldsymbol{x}' = \boldsymbol{x}_i + \frac{D y_i \boldsymbol{w}}{\|\boldsymbol{w}\|}$$

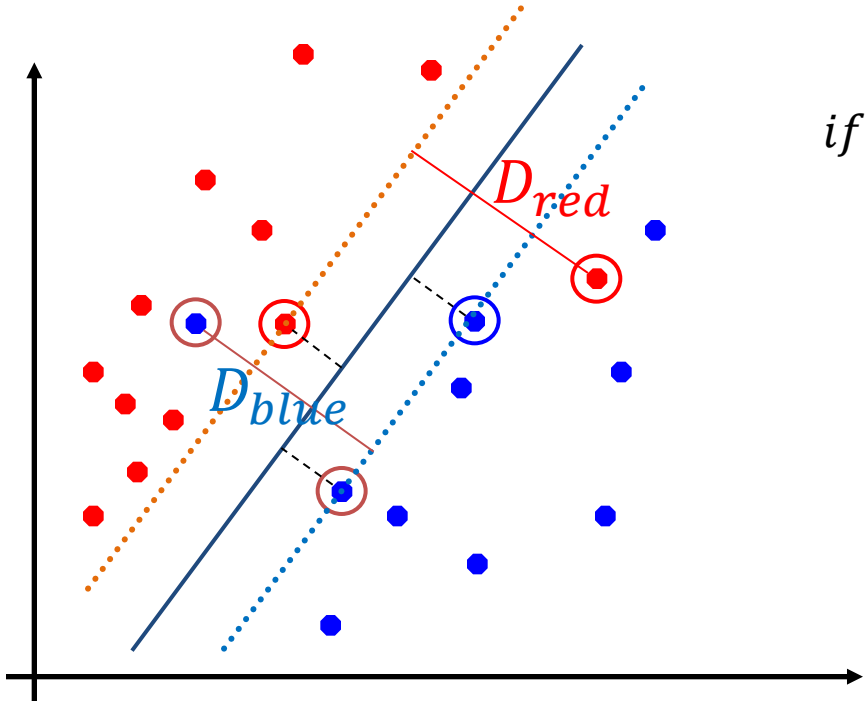
$$\boldsymbol{w}^T\left(\boldsymbol{x}_i + \frac{D y_i \boldsymbol{w}}{\|\boldsymbol{w}\|}\right) + b = y_i$$

$$D = \frac{y_i - \boldsymbol{w}^T\boldsymbol{x}_i - b}{y_i\|\boldsymbol{w}\|}$$

Trick 1: Soft-margin SVM---Slack Variable

- ***Slack variables*** ε_i are not the distances to the corresponding margin boundary!

$$D = \frac{y_i - \boldsymbol{w}^T \boldsymbol{x}_i - b}{y_i \|\boldsymbol{w}\|}$$



$if D = 0$: $\boldsymbol{w}^T \boldsymbol{x}_i + b = y_i$, or $y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) > 1$

Correctly classified, on or outside the margin

$if\ 0 < D \leq 1$: $0 < \frac{y_i - \boldsymbol{w}^T \boldsymbol{x}_i - b}{y_i \|\boldsymbol{w}\|} \leq 1$

$if\ y_i = -1$: $-1 < \boldsymbol{w}^T \boldsymbol{x}_i + b \leq \|\boldsymbol{w}\| - 1$

$if\ y_i = 1$: $1 - \|\boldsymbol{w}\| \leq \boldsymbol{w}^T \boldsymbol{x}_i + b < 1$

$1 - \|\boldsymbol{w}\| \leq y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) < 1$

Correctly classified inside the margin or misclassified

$if D > 1$: $y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) < 1 - \|\boldsymbol{w}\|$

Correctly classified inside the margin or misclassified

Trick 1: Soft-margin SVM---Slack Variable

- ***Slack variables*** ε_i are not the distances are not the distances to the corresponding margin boundary!

$$\varepsilon_i \neq D = \frac{y_i - \boldsymbol{w}^T \boldsymbol{x}_i - b}{y_i \|\boldsymbol{w}\|}$$

$if\ \varepsilon_i = 0: y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) \geq 1$

Correctly classified, on or outside the margin

$if\ D = 0:\ y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) \geq 1$

Correctly classified, on or outside the margin

$if\ 0 < \varepsilon_i \leq 1: 0 \leq\ y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) < 1$

Correctly classified, inside the margin

$if\ 0 < D \leq 1:\ 1 - \|\boldsymbol{w}\| \leq y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) < 1$

Inside the margin or misclassified

$if\ \varepsilon_i > 1: y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) < 0$

Misclassified, wrong side

$if D > 1:\ y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) < 1 - \|\boldsymbol{w}\|$

Inside the margin or misclassified

What is the relationship between ε_i and D?

Trick 1: Soft-margin SVM---Slack Variable

- ***Slack variables*** ε_i are not the distances!

$$\varepsilon_i \neq D = \frac{y_i - \boldsymbol{w}^T \boldsymbol{x}_i - b}{y_i \|\boldsymbol{w}\|}$$

$if\ \varepsilon_i = 0: y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) \geq 1$

Correctly classified, on or outside the margin

$if\ D = 0:\ y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) \geq 1$

Correctly classified, on or outside the margin

$if\ 0 < \varepsilon_i \leq 1: 0 \leq y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) < 1$

Correctly classified, inside the margin

$if\ 0 < D \leq 1:\ 1 - \|\boldsymbol{w}\| \leq y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) < 1$

Inside the margin or misclassified

$if\ \varepsilon_i > 1: y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) < 0$

Misclassified, wrong side

$if D > 1:\ y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) < 1 - \|\boldsymbol{w}\|$

Inside the margin or misclassified

$$\varepsilon_i = D, when\ \|\boldsymbol{w}\| = 1$$

Trick 1: Soft-margin SVM Primal

Soft-margin Primal

$$\min_{\boldsymbol{w},b,\boldsymbol{\varepsilon}} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_i \varepsilon_i$$

$$\text{s.t. } y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1 - \varepsilon_i,\ \varepsilon_i \geq 0, \forall i$$

d+1+*N* variables

2N constraints

Hard-margin Primal

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

$$\text{s. t. } y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1, \forall i$$

d+1 variables

N constraints

C	ε_i	Tolerance for misclassification	Margin
Larger			
Smaller			

Trick 1: Soft-margin SVM Primal

Soft-margin Primal

$$\min_{\boldsymbol{w},b,\boldsymbol{\varepsilon}} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_i \varepsilon_i$$

$$\text{s.t. } y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1 - \varepsilon_i,\ \varepsilon_i \geq 0, \forall i$$

d+1+*N* variables

2N constraints

Hard-margin Primal

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

$$\text{s. t. } y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1, \forall i$$

d+1 variables

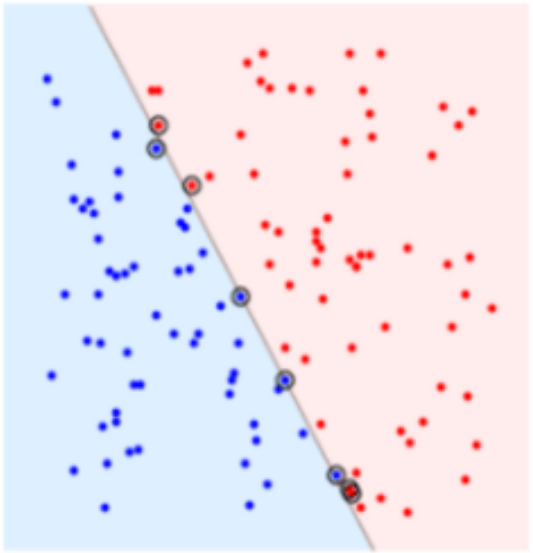
N constraints

C	ε_i	Tolerance for misclassification	Margin
Larger	Smaller	Less	Smaller
Smaller	Larger	More	Larger

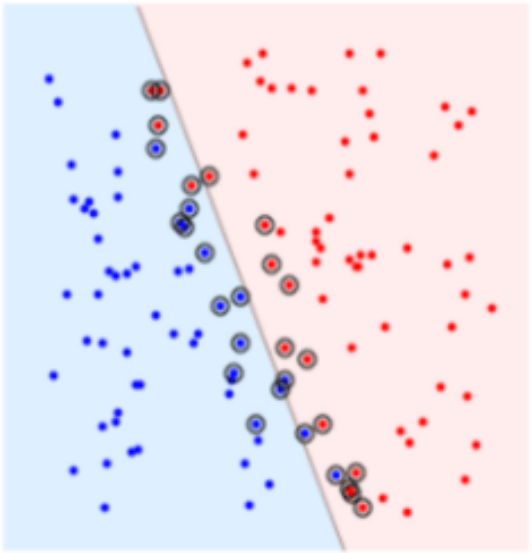
Trick 1: Soft-margin SVM Primal

Different values of C

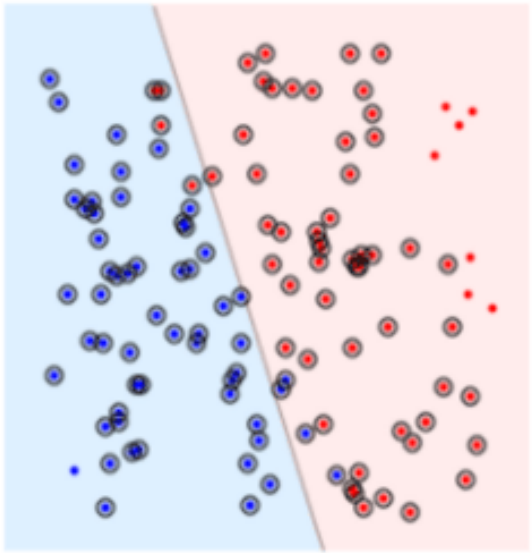
C=1000



C=10



C=0.1



The data points circled are the points lying within the margin region including the margin boundaries

Trick 1: Soft-margin SVM Primal->Dual

Soft-margin Primal

$$\min_{\boldsymbol{w},b,\boldsymbol{\varepsilon}} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_i \varepsilon_i$$

$$\text{s.t. } y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1 - \varepsilon_i, \varepsilon_i \geq 0, \forall i$$

- Inequality Constraints:

  Karush-Kuhn-Tucker (KKT) conditions

  1) Define the Lagrangian function:

$$\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\varepsilon}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) - \sum_i \lambda_i g_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) + \mu_i h_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) \text{, where } \lambda_i, \mu_i \geq 0, \forall i$$

$$f(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_i \varepsilon_i,$$

$$g_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) + \varepsilon_i - 1 \geq 0,$$

$$h_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = \varepsilon_i \geq 0$$

Trick 1: Soft-margin SVM Primal->Dual

- Inequality Constraints:

  Karush-Kuhn-Tucker (KKT) conditions

  1) Define the Lagrangian function:

$$\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\varepsilon}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) - \sum_i \lambda_i g_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) + \mu_i h_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) \text{, where } \lambda_i, \mu_i \geq 0, \forall i$$

$$f(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_i \varepsilon_i, \quad g_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) + \varepsilon_i - 1 \geq 0, \quad h_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = \varepsilon_i \geq 0$$

  2) KKT conditions for optimality:

  ✓ Stationarity:

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{w}} = \mathbf{0} \longrightarrow \boldsymbol{w} - \sum_i \lambda_i y_i \boldsymbol{x}_i = \mathbf{0} \longrightarrow \boldsymbol{w} = \sum_i \lambda_i y_i \boldsymbol{x}_i$$

$$\frac{\partial \mathcal{L}}{\partial b} = 0 \longrightarrow \sum_i \lambda_i y_i = 0$$

$$\frac{\partial \mathcal{L}}{\partial \varepsilon_i} = 0 \longrightarrow C - \lambda_i - \mu_i = 0 \longrightarrow C = \lambda_i + \mu_i$$

$$(\lambda_i, \mu_i \leq C)$$

Trick 1: Soft-margin SVM Primal->Dual

- Inequality Constraints:

  Karush-Kuhn-Tucker (KKT) conditions

  1) Define the Lagrangian function:

$$\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\varepsilon}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) - \sum_i \lambda_i g_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) + \mu_i h_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) \text{, where } \lambda_i, \mu_i \geq 0, \forall i$$

$$f(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_i \varepsilon_i, \quad g_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) + \varepsilon_i - 1 \geq 0, \quad h_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = \varepsilon_i \geq 0$$

  2) KKT conditions for optimality:

  ✓ Stationarity: $\boldsymbol{w}^* = \sum_i \lambda_i y_i \boldsymbol{x}_i, \ \sum_i \lambda_i y_i = 0, \ C = \lambda_i + \mu_i \ (\lambda_i, \mu_i \leq C)$

  ✓ Primal Feasibility: $g_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) + \varepsilon_i - 1 \geq 0, \ h_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = \varepsilon_i \geq 0$

  ✓ Dual Feasibility: $\lambda_i, \mu_i \geq 0, \forall i$

  ✓ Complementary Slackness: $\lambda_i g_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = 0, \qquad \mu_i h_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = 0, \forall i$

  If $g_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) > 0$ (inactive), $\lambda_i = 0$

  If $g_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = 0$ (active), $\lambda_i > 0$

  If $h_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) > 0$ (inactive), $\mu_i = 0$

  If $h_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = 0$ (active), $\mu_i > 0$

How to interpret?

Trick 1: Soft-margin SVM Primal->Dual

- Interpretation of Complementary Slackness

$\lambda_i g_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = 0,$ $\quad$ $\mu_i h_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = 0, \forall i$ $\quad$ $C = \lambda_i + \mu_i$

If $g_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) > 0$ (inactive), $\lambda_i = 0$

If $g_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = 0$ (active), $\lambda_i > 0$

If $h_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) > 0$ (inactive), $\mu_i = 0$

If $h_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = 0$ (active), $\mu_i > 0$

(a) When $\lambda_i = 0,\ \mu_i = C > 0$

$g_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) + \varepsilon_i - 1 > 0$

$h_i(\boldsymbol{w}, b, \boldsymbol{\varepsilon}) = \varepsilon_i = 0$

$y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) > 1$

Correctly classified and outside the margin

(b) When $\lambda_i > 0$,

$y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) = 1 - \varepsilon_i$

Case 1: $\varepsilon_i = 0, \mu_i > 0$

On the margin boundaries

Case 2: $0 < \varepsilon_i \leq 1, \mu_i = 0$

Correctly classified, inside the margin

Case 3: $\varepsilon_i > 1, \mu_i = 0$

Misclassified

Trick 1: Soft-margin SVM Primal->Dual

$$\min_{\boldsymbol{w},b,\boldsymbol{\varepsilon}} \mathcal{L}(\boldsymbol{w},b,\boldsymbol{\varepsilon},\boldsymbol{\lambda},\boldsymbol{\mu}) = \min_{\boldsymbol{w},b,\boldsymbol{\varepsilon}} f(\boldsymbol{w},b,\boldsymbol{\varepsilon}) - \sum_i \lambda_i g_i(\boldsymbol{w},b,\boldsymbol{\varepsilon}) + \mu_i h_i(\boldsymbol{w},b,\boldsymbol{\varepsilon}) \text{, where } \lambda_i, \mu_i \geq 0, \forall i$$

$$= \min_{\boldsymbol{w},b,\boldsymbol{\varepsilon}} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_i \varepsilon_i - \sum_i \lambda_i (y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) + \varepsilon_i - 1) + \mu_i\varepsilon_i$$

$$= \min_{\boldsymbol{w},b,\boldsymbol{\varepsilon}} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_i \varepsilon_i - \sum_i \lambda_i y_i \boldsymbol{w}^T\boldsymbol{x}_i - \sum_i \lambda_i y_i b - \sum_i \lambda_i\varepsilon_i + \sum_i \lambda_i - \sum_i \mu_i\varepsilon_i$$

$$= \frac{1}{2}\boldsymbol{w}^{*T}\boldsymbol{w}^* - \boldsymbol{w}^{*T}\boldsymbol{w}^* + \sum_i \lambda_i$$

$$= -\frac{1}{2}\sum_i \lambda_i y_i \boldsymbol{x}_i^T \sum_i \lambda_i y_i \boldsymbol{x}_i + \sum_i \lambda_i$$

$$= -\frac{1}{2}\sum_{i,j} \lambda_i\lambda_j y_i y_j \boldsymbol{x}_i^T\boldsymbol{x}_j + \sum_i \lambda_i$$

$f(\boldsymbol{w},b,\boldsymbol{\varepsilon}) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_i \varepsilon_i$,

$g_i(\boldsymbol{w},b,\boldsymbol{\varepsilon}) = y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) + \varepsilon_i - 1 \geq 0$,

$h_i(\boldsymbol{w},b,\boldsymbol{\varepsilon}) = \varepsilon_i \geq 0$

$$\boldsymbol{w}^* = \sum_i \lambda_i y_i \boldsymbol{x}_i$$

$$\sum_i \lambda_i y_i = 0$$

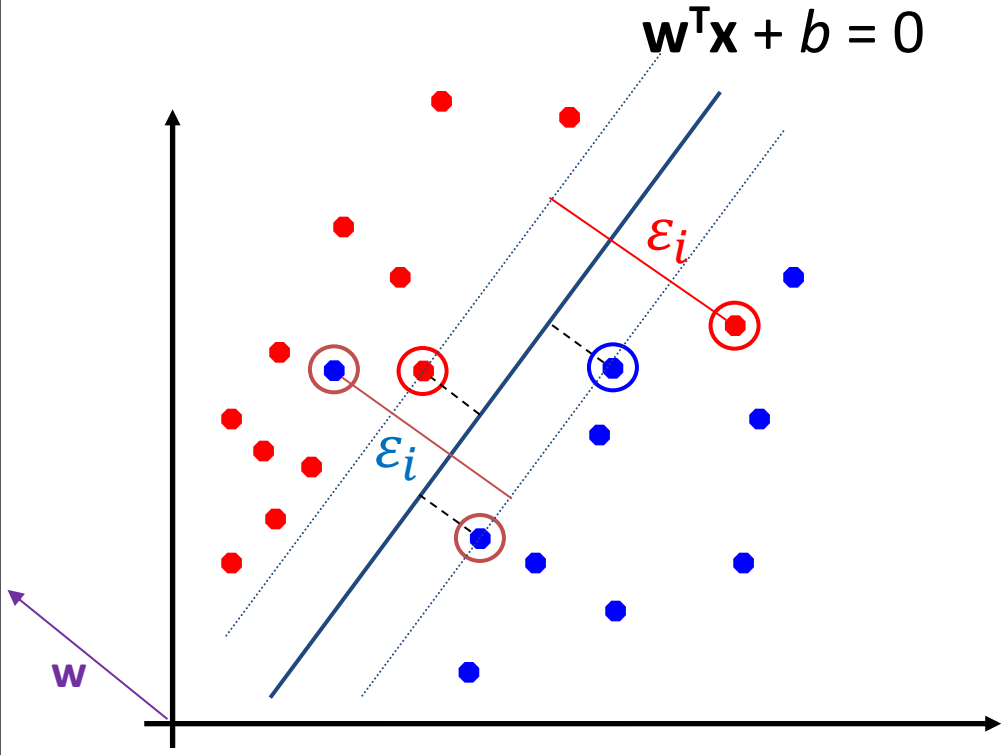
$$C = \lambda_i + \mu_i$$

Soft-margin Dual

$$\max_{\boldsymbol{\lambda}} -\frac{1}{2}\sum_{i,j} \lambda_i\lambda_j y_i y_j \boldsymbol{x}_i^T\boldsymbol{x}_j + \sum_i \lambda_i$$

s.t. $0 \leq \lambda_i \leq C, \sum_i \lambda_i y_i = 0, \forall i$

Soft-margin SVM: Decision Boundary



$$\begin{cases} \boldsymbol{w}^* = \sum_i \lambda_i y_i \boldsymbol{x}_i \\ \sum_i \lambda_i y_i = 0 \end{cases}$$

$$\boldsymbol{w}^{*T}\boldsymbol{x} + b^* = \sum_i \lambda_i y_i \boldsymbol{x}_i^T \boldsymbol{x} + b^* = 0$$

for any support vector $\boldsymbol{x}_s$ ($\lambda_i > 0$),

$$y_s(\boldsymbol{w}^{*T}\boldsymbol{x}_s + b^*) = 1 - \varepsilon_s$$
$$\downarrow$$
$$y_s^2(\boldsymbol{w}^{*T}\boldsymbol{x}_s + b^*) = y_s(1 - \varepsilon_s)$$
$$\downarrow$$
$$b^* = y_s(1 - \varepsilon_s) - \sum_i \lambda_i y_i \boldsymbol{x}_i^T \boldsymbol{x}_s$$

Quick Question

For Soft-margin SVM with the following objective function:

$$\min_{\boldsymbol{w},b,\boldsymbol{\varepsilon}} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_i \varepsilon_i$$

$$\text{s.t. } y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1 - \varepsilon_i,\ \varepsilon_i \geq 0, \forall i$$

or

$$\max_{\boldsymbol{\lambda}} -\frac{1}{2}\sum_{i,j} \lambda_i\lambda_j y_i y_j \boldsymbol{x}_i^T\boldsymbol{x}_j + \sum_i \lambda_i$$

$$\text{s.t. } 0 \leq \lambda_i \leq C,\ \textstyle\sum_i \lambda_i y_i = 0, \forall i$$

Which of the following data points are considered support vectors?

A. Data points that lie exactly on the margin boundaries.
B. Data points that have $\lambda_i > 0$.
C. Data points that determine the decision boundary
D. Data points inside the margin
E. Data points that are misclassified
F. Data points that have $\varepsilon_i \geq 0$

Soft-margin VS Hard-margin SVMs

Hard-margin Primal:

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

$$\text{s.t. } y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1, \forall i$$

Optimize over *d*+1 variables

N constraints

Hard-margin Dual:

$$\max_{\boldsymbol{\lambda}} -\frac{1}{2}\sum_{i,j} \lambda_i\lambda_j y_i y_j \boldsymbol{x}_i^T\boldsymbol{x}_j + \sum_i \lambda_i$$

$$\text{s.t. } \lambda_i \geq 0, \textstyle\sum_i \lambda_i y_i = 0, \forall i$$

Optimize over *N* variables

N+1 constraints

Soft-margin Primal:

$$\min_{\boldsymbol{w},b,\boldsymbol{\varepsilon}} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_i \varepsilon_i$$

$$\text{s.t. } y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1 - \varepsilon_i, \varepsilon_i \geq 0, \forall i$$

Optimize over *d*+1+*N* variables

2N constraints

Soft-margin Dual:

$$\max_{\boldsymbol{\lambda}} -\frac{1}{2}\sum_{i,j} \lambda_i\lambda_j y_i y_j \boldsymbol{x}_i^T\boldsymbol{x}_j + \sum_i \lambda_i$$

$$\text{s.t. } 0 \leq \lambda_i \leq C, \textstyle\sum_i \lambda_i y_i = 0, \forall i$$

Optimize over *N* variables

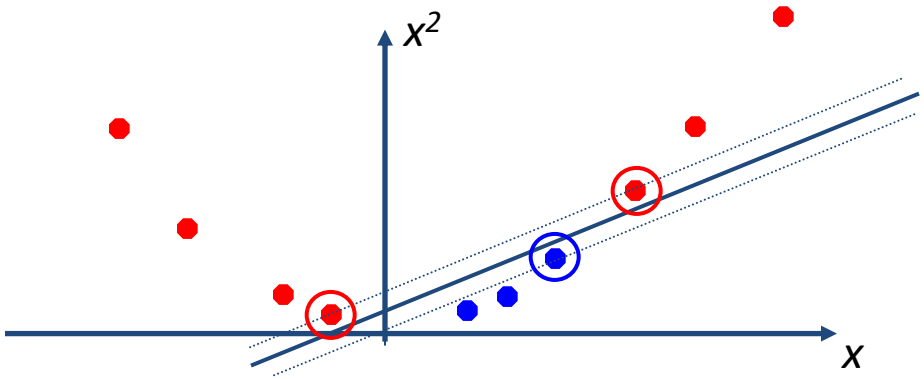
(2)*N*+1 constraints

Kernel SVM

Trick 2: Kernel SVM

- Here's a linearly non-separable dataset in 1-D space

- We can map data x from 1-D to 2-D by $x \rightarrow (x, x^2)$

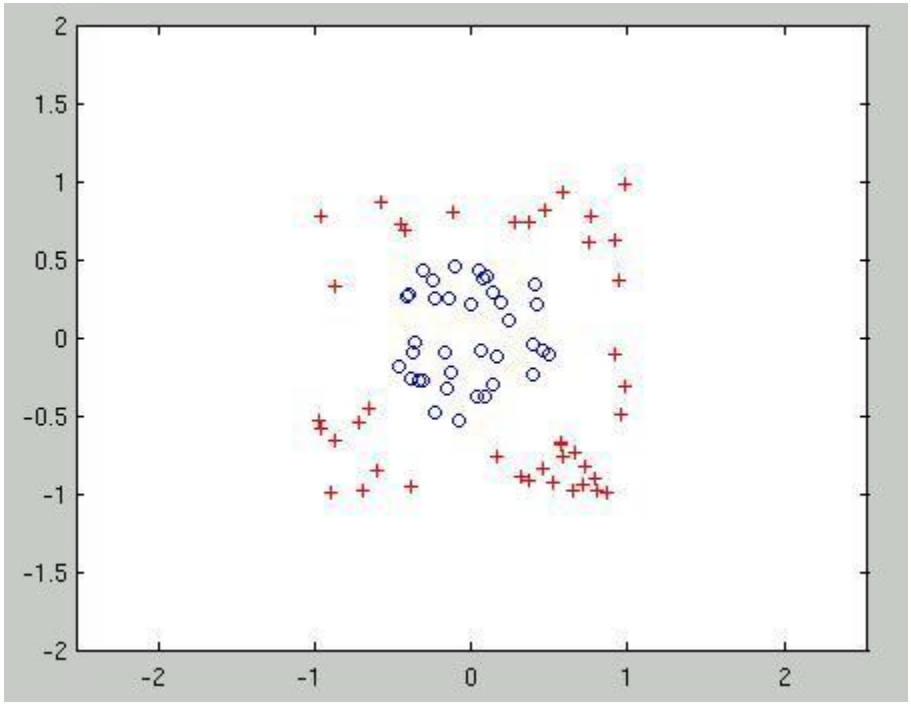


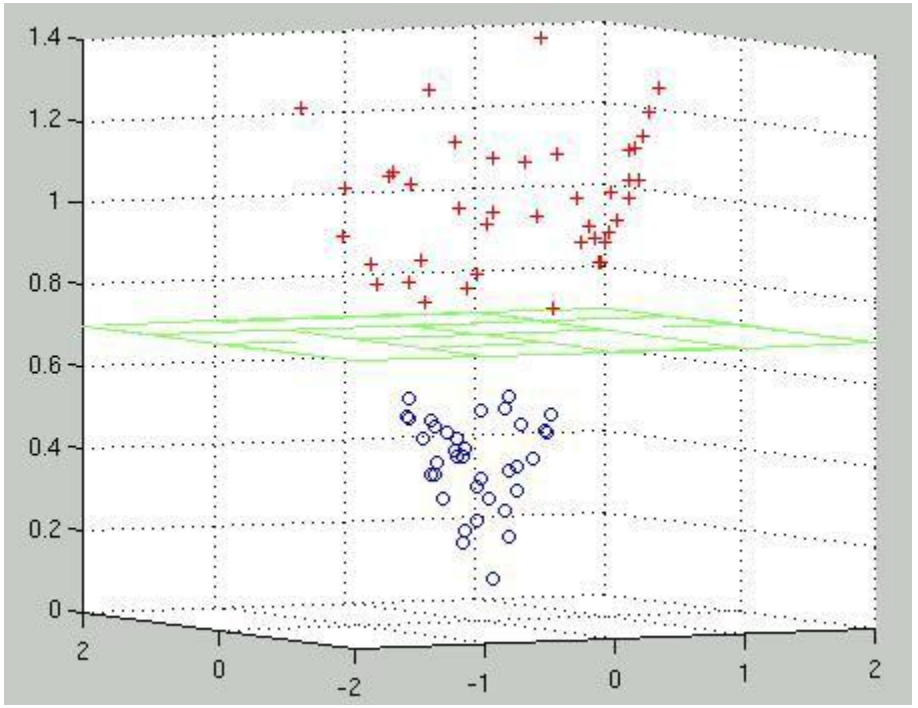
- ❖ Now the data is linearly separable in the new space
- ❖ We can run SVM in the new space
- ❖ The linear decision boundary in the new space corresponds to a non-linear decision boundary in the old space

Trick 2: Kernel SVM

Another Example

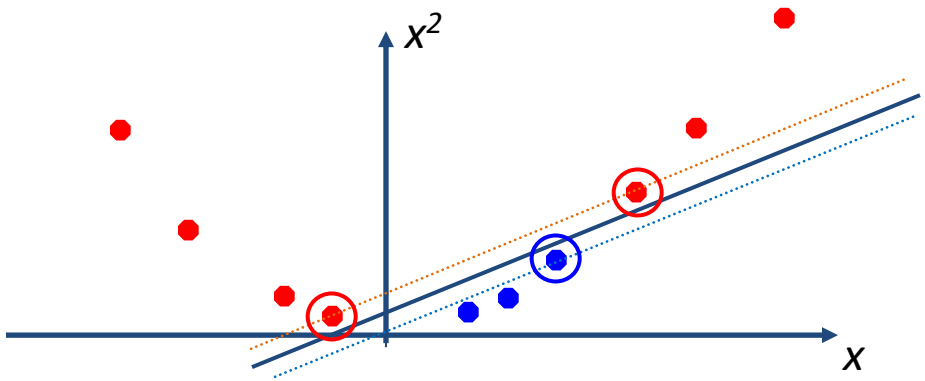
$$(x_1, x_2) \Rightarrow (x_1, x_2, \sqrt{x_1^2 + x_2^2})$$





Trick 2: Kernel SVM

- **General idea**: map the original feature space $\boldsymbol{x} = (x_1, \ldots, x_d)^T$ to some higher-dimensional feature space $\boldsymbol{\Phi}(\boldsymbol{x})$ where the data is separable



Decision Boundary:

$$\boldsymbol{w}^T \boldsymbol{\Phi}(\boldsymbol{x}) + b = 0$$

Plus Plane:

$$\boldsymbol{w}^T \boldsymbol{\Phi}(\boldsymbol{x}) + b = 1$$

Minus Plane:

$$\boldsymbol{w}^T \boldsymbol{\Phi}(\boldsymbol{x}) + b = -1$$

Trick 2: Kernel SVM

- Optimization Formulation after mapping

Hard-margin Primal:

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

$$\text{s.t. } y_i(\boldsymbol{w}^T\boldsymbol{\Phi}(\boldsymbol{x}_i) + b) \geq 1, \forall i$$

Hard-margin Dual:

$$\max_{\boldsymbol{\lambda}} -\frac{1}{2}\sum_{i,j}\lambda_i\lambda_j y_i y_j \boldsymbol{\Phi}(\boldsymbol{x}_i)^T\boldsymbol{\Phi}(\boldsymbol{x}_j) + \sum_i \lambda_i$$

$$\text{s.t. } \lambda_i \geq 0, \textstyle\sum_i \lambda_i y_i = 0, \forall i$$

Soft-margin Primal:

$$\min_{\boldsymbol{w},b,\boldsymbol{\varepsilon}} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_i \varepsilon_i$$

$$\text{s.t. } y_i(\boldsymbol{w}^T\boldsymbol{\Phi}(\boldsymbol{x}_i) + b) \geq 1 - \varepsilon_i, \varepsilon_i \geq 0, \forall i$$

Soft-margin Dual:

$$\max_{\boldsymbol{\lambda}} -\frac{1}{2}\sum_{i,j}\lambda_i\lambda_j y_i y_j \boldsymbol{\Phi}(\boldsymbol{x}_i)^T\boldsymbol{\Phi}(\boldsymbol{x}_j) + \sum_i \lambda_i$$

$$\text{s.t. } 0 \leq \lambda_i \leq C, \textstyle\sum_i \lambda_i y_i = 0, \forall i$$

Trick 2: Kernel SVM

- Optimal decision boundary after mapping

Hard-margin SVM:

$$\boldsymbol{w}^{*T}\boldsymbol{\Phi}(\boldsymbol{x}) + b^* = \sum_i \lambda_i y_i \boldsymbol{\Phi}(\boldsymbol{x}_i)^T \boldsymbol{\Phi}(\boldsymbol{x}) + b^* = 0$$

where $b^* = y_s - \sum_i \lambda_i y_i \boldsymbol{\Phi}(\boldsymbol{x}_i)^T \boldsymbol{\Phi}(\boldsymbol{x}_s)$

for any support vector $\boldsymbol{x}_s$ $(\lambda_i > 0)$

$$\boldsymbol{w}^* = \sum_i \lambda_i y_i \boldsymbol{\Phi}(\boldsymbol{x}_i)$$

$$\sum_i \lambda_i y_i = 0$$

Soft-margin SVM:

$$\boldsymbol{w}^{*T}\boldsymbol{\Phi}(\boldsymbol{x}) + b^* = \sum_i \lambda_i y_i \boldsymbol{\Phi}(\boldsymbol{x}_i)^T \boldsymbol{\Phi}(\boldsymbol{x}) + b^* = 0$$

where $b^* = y_s(1 - \varepsilon_s) - \sum_i \lambda_i y_i \boldsymbol{\Phi}(\boldsymbol{x}_i)^T \boldsymbol{\Phi}(\boldsymbol{x}_s)$

for any support vector $\boldsymbol{x}_s$ $(\lambda_i > 0)$

Trick 2: Kernel SVM

- The solution relies on the inner product $\boldsymbol{\Phi}(\boldsymbol{x}_i)^T\boldsymbol{\Phi}(\boldsymbol{x}_j)$
- A *kernel function* K: $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{\Phi}(\boldsymbol{x}_i)^T\boldsymbol{\Phi}(\boldsymbol{x}_j)$

Hard-margin Dual:

$$\max_{\boldsymbol{\lambda}} -\frac{1}{2}\sum_{i,j}\lambda_i\lambda_j y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) + \sum_i \lambda_i$$

$$\text{s.t. } \lambda_i \geq 0, \textstyle\sum_i \lambda_i y_i = 0, \forall i$$

$$\boldsymbol{w}^{*T}\boldsymbol{\Phi}(\boldsymbol{x}) + b^* = \sum_i \lambda_i y_i K(\boldsymbol{x}_i, \boldsymbol{x}) + b^* = 0$$

where $b^* = y_s - \sum_i \lambda_i y_i K(\boldsymbol{x}_i, \boldsymbol{x}_s)$

for any support vector $\boldsymbol{x}_s$ $(\lambda_i > 0)$

Soft-margin Dual:

$$\max_{\boldsymbol{\lambda}} -\frac{1}{2}\sum_{i,j}\lambda_i\lambda_j y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) + \sum_i \lambda_i$$

$$\text{s.t. } 0 \leq \lambda_i \leq C, \textstyle\sum_i \lambda_i y_i = 0, \forall i$$

$$\boldsymbol{w}^{*T}\boldsymbol{\Phi}(\boldsymbol{x}) + b^* = \sum_i \lambda_i y_i K(\boldsymbol{x}_i, \boldsymbol{x}) + b^* = 0$$

where $b^* = y_s(1 - \varepsilon_s) - \sum_i \lambda_i y_i K(\boldsymbol{x}_i, \boldsymbol{x}_s)$

for any support vector $\boldsymbol{x}_s$ $(\lambda_i > 0)$

Examples of Kernel Functions

- Linear: $K(\mathbf{x}_i,\mathbf{x}_j) = \mathbf{x}_i^T\mathbf{x}_j$
  Mapping Φ: $\mathbf{x} \rightarrow \boldsymbol{\phi}(\mathbf{x})$, where $\boldsymbol{\phi}(\mathbf{x})$ is $\mathbf{x}$ itself

- Polynomial of power p: $K(\mathbf{x}_i,\mathbf{x}_j) = (1+ \mathbf{x}_i^T\mathbf{x}_j)^p$
  Mapping Φ: $\mathbf{x} \rightarrow \boldsymbol{\phi}(\mathbf{x})$, where $\boldsymbol{\phi}(\mathbf{x})$ has $\binom{d+p}{p} = \frac{(d+p)!}{p!\cdot d!}$ dimensions

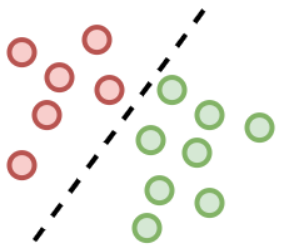
- Gaussian (radial-basis function): $K(\mathbf{x}_i,\mathbf{x}_j) = \exp(\frac{\|\boldsymbol{x}_i-\boldsymbol{x}_j\|^2}{2\sigma^2})$
  Mapping Φ: $\mathbf{x} \rightarrow \boldsymbol{\phi}(\mathbf{x})$, where $\boldsymbol{\phi}(\mathbf{x})$ is *infinite-dimensional*: every point is mapped to *a function* (a Gaussian)

Right Kernel for SVM?

Right Kernel for SVM

- Linear: $K(\mathbf{x}_i,\mathbf{x}_j) = \mathbf{x}_i^T\mathbf{x}_j$

  When to Use? ✓ The data is linearly separable

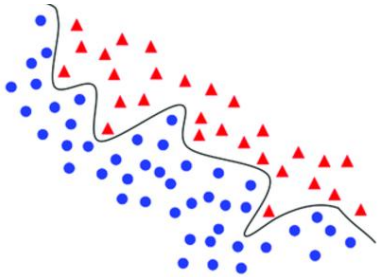


- Polynomial : $K(\mathbf{x}_i,\mathbf{x}_j) = (1+ \mathbf{x}_i^T\mathbf{x}_j)^p$ (need to choose the right p)

  When to Use? ✓ Nonlinear and polynomial-like decision boundary

  ✓ Capture feature interactions

  ✓ Low-dimensional data

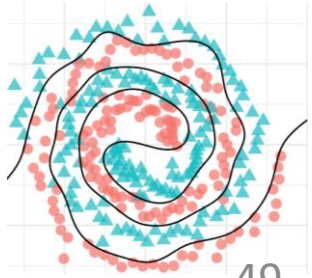


- Gaussian (RBF) : $K(\mathbf{x}_i,\mathbf{x}_j) = \exp(\frac{\|x_i - x_j\|^2}{2\sigma^2})$

  When to Use? ✓ Highly complex and nonlinear decision boundary

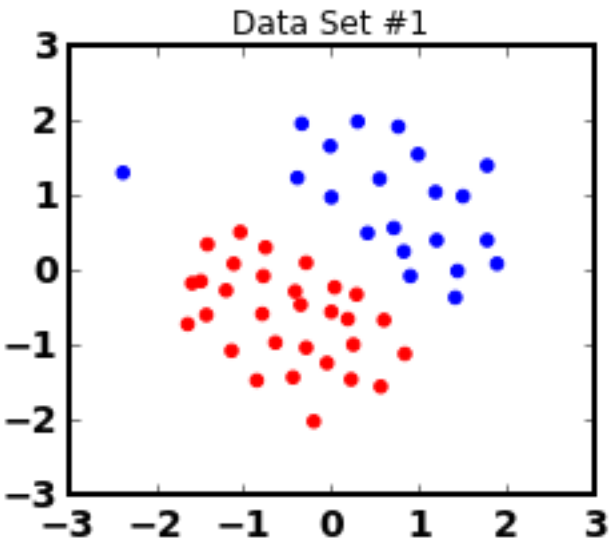
  ✓ Unknown underlying feature transformations

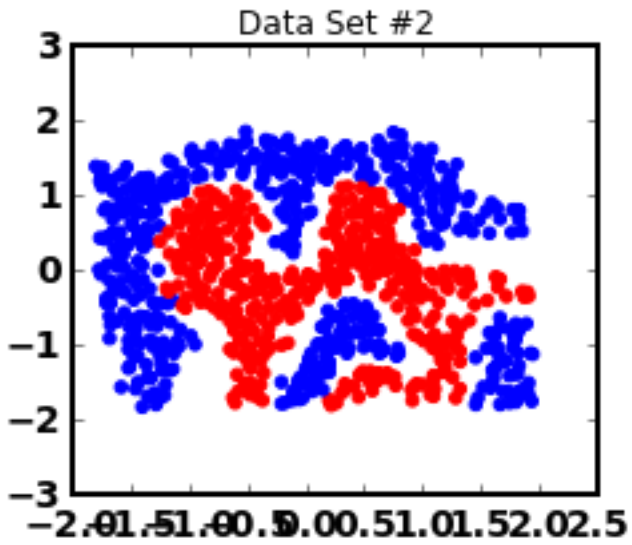
  ✓ Moderate-sized dataset

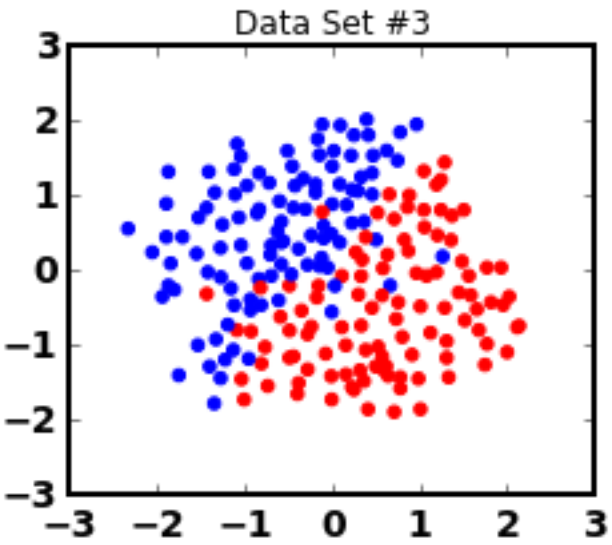


Right Kernel for SVM

Which Kernel to Use?







What Functions Can be Kernels?

- Can we always find out a *K* (kernel) for a given $\boldsymbol{\Phi}$?
- Given a *K*, can we always find out a $\boldsymbol{\Phi}$ such that $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{\Phi}(\boldsymbol{x}_i)^T \boldsymbol{\Phi}(\boldsymbol{x}_j)$?

What Functions Can be Kernels?

- Mercer's Theorem

  For $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ to be a valid kernel, it must satisfy:

  i. Symmetry: $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = K(\boldsymbol{x}_j, \boldsymbol{x}_i)$

  ii. The kernel matrix $\boldsymbol{G}$: $\boldsymbol{v}^T \boldsymbol{G} \boldsymbol{v} \geq 0$, *for any non-zero vector* $\boldsymbol{v}$ (Positive Semi-Definiteness)

$$\boldsymbol{G} = \begin{bmatrix} K(\boldsymbol{x}_1, \boldsymbol{x}_1) & \cdots & K(\boldsymbol{x}_1, \boldsymbol{x}_N) \\ \vdots & \ddots & \vdots \\ K(\boldsymbol{x}_N, \boldsymbol{x}_1) & \cdots & K(\boldsymbol{x}_N, \boldsymbol{x}_N) \end{bmatrix}$$

What Functions Can be Kernels?

- Mercer's Theorem

  For $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ to be a valid kernel, it must satisfy:

  i. Symmetry: $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = K(\boldsymbol{x}_j, \boldsymbol{x}_i)$ $\quad \boldsymbol{\Phi}(\boldsymbol{x}_i)^T \boldsymbol{\Phi}(\boldsymbol{x}_j) = \boldsymbol{\Phi}(\boldsymbol{x}_j)^T \boldsymbol{\Phi}(\boldsymbol{x}_i)$

  ii. The kernel matrix $\boldsymbol{G}$: $\boldsymbol{v}^T \boldsymbol{G} \boldsymbol{v} \geq 0$, *for any non-zero vector* $\boldsymbol{v}$

$$\boldsymbol{G} = \begin{bmatrix} K(\boldsymbol{x}_1, \boldsymbol{x}_1) & \cdots & K(\boldsymbol{x}_1, \boldsymbol{x}_N) \\ \vdots & \ddots & \vdots \\ K(\boldsymbol{x}_N, \boldsymbol{x}_1) & \cdots & K(\boldsymbol{x}_N, \boldsymbol{x}_N) \end{bmatrix}$$

(Positive Semi-Definiteness)

$$\boldsymbol{v}^T \boldsymbol{G} \boldsymbol{v} = \begin{bmatrix} v_1 & \cdots & v_N \end{bmatrix} \begin{bmatrix} K(\boldsymbol{x}_1, \boldsymbol{x}_1) & \cdots & K(\boldsymbol{x}_1, \boldsymbol{x}_N) \\ \vdots & \ddots & \vdots \\ K(\boldsymbol{x}_N, \boldsymbol{x}_1) & \cdots & K(\boldsymbol{x}_N, \boldsymbol{x}_N) \end{bmatrix} \begin{bmatrix} v_1 \\ \vdots \\ v_N \end{bmatrix}$$

$$= \begin{bmatrix} \sum_{i=1}^{N} v_i K(\boldsymbol{x}_i, \boldsymbol{x}_1) & \cdots & \sum_{i=1}^{N} v_i K(\boldsymbol{x}_i, \boldsymbol{x}_N) \end{bmatrix} \begin{bmatrix} v_1 \\ \vdots \\ v_N \end{bmatrix} = \sum_{j=1}^{N} \sum_{i=1}^{N} v_i K(\boldsymbol{x}_i, \boldsymbol{x}_j) v_j$$

What Functions Can be Kernels?

$$\boldsymbol{v}^T \boldsymbol{G}\boldsymbol{v} = \sum_{j=1}^{N}\sum_{i=1}^{N} v_i K(\boldsymbol{x}_i, \boldsymbol{x}_j) v_j$$

If K is a valid kernel, $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{\Phi}(\boldsymbol{x}_i)^T \boldsymbol{\Phi}(\boldsymbol{x}_j)$

$$\boldsymbol{v}^T \boldsymbol{G}\boldsymbol{v} = \sum_{j=1}^{N}\sum_{i=1}^{N} v_i K(\boldsymbol{x}_i, \boldsymbol{x}_j) v_j = \sum_{j=1}^{N}\sum_{i=1}^{N} v_i \boldsymbol{\Phi}(\boldsymbol{x}_i)^T \boldsymbol{\Phi}(\boldsymbol{x}_j) v_j$$

Because of the Linearity of the inner product:

$$(a\boldsymbol{\Phi}(\boldsymbol{x}_i))^T \boldsymbol{\Phi}(\boldsymbol{x}_j) = a\boldsymbol{\Phi}(\boldsymbol{x}_i)^T \boldsymbol{\Phi}(\boldsymbol{x}_j)$$

$$(\boldsymbol{\Phi}(\boldsymbol{x}_i) + \boldsymbol{\Phi}(\boldsymbol{x}_k))^T \boldsymbol{\Phi}(\boldsymbol{x}_j) = \boldsymbol{\Phi}(\boldsymbol{x}_i)^T \boldsymbol{\Phi}(\boldsymbol{x}_j) + \boldsymbol{\Phi}(\boldsymbol{x}_k)^T \boldsymbol{\Phi}(\boldsymbol{x}_j)$$

$$\boldsymbol{v}^T \boldsymbol{G}\boldsymbol{v} = \sum_{j=1}^{N}\sum_{i=1}^{N} v_i \boldsymbol{\Phi}(\boldsymbol{x}_i)^T \boldsymbol{\Phi}(\boldsymbol{x}_j) v_j = \sum_{j=1}^{N}(\sum_{i=1}^{N} v_i \boldsymbol{\Phi}(\boldsymbol{x}_i))^T \boldsymbol{\Phi}(\boldsymbol{x}_j) v_j = (\sum_{i=1}^{N} v_i \boldsymbol{\Phi}(\boldsymbol{x}_i))^T (\sum_{j=1}^{N} \boldsymbol{\Phi}(\boldsymbol{x}_j) v_j)$$

Because of the Positive Semi-definiteness of the inner product: $\boldsymbol{\Phi}(\boldsymbol{x}_i)^T \boldsymbol{\Phi}(\boldsymbol{x}_i) \geq 0$

$$\boldsymbol{v}^T \boldsymbol{G}\boldsymbol{v} = (\sum_{i=1}^{N} v_i \boldsymbol{\Phi}(\boldsymbol{x}_i))^T (\sum_{j=1}^{N} \boldsymbol{\Phi}(\boldsymbol{x}_j) v_j) \geq 0$$

What Functions Can be Kernels?

- Can we always find out a *K* (kernel) for a given Φ ?

  Yes, such *K* always satisfies Mercer's conditions

- Given a *K*, can we always find out a Φ such that $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{\Phi}(\boldsymbol{x}_i)^T \boldsymbol{\Phi}(\boldsymbol{x}_j)$?

  No, only for *K* that satisfies Mercer's conditions

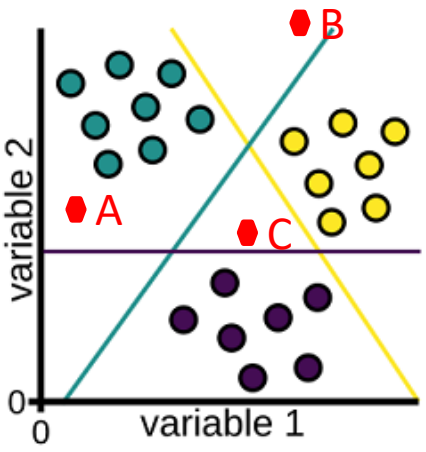
Multi-class SVM

SVM: more than two classes

- N-class problem

1. One-vs-All approach

Training: train a binary SVM classifier for each class



- Classifier 1: Class 1 vs. the rest (class 2--*N*)

  Label: +1 Label: -1
- Classifier 2: Class 2 vs. the rest (class 1, 3--*N*)

  Label: +1 Label: -1
- …
- Classifier *N*: Class *N* vs. the rest (class 1--(*N*-1))

  Label: +1 Label: -1

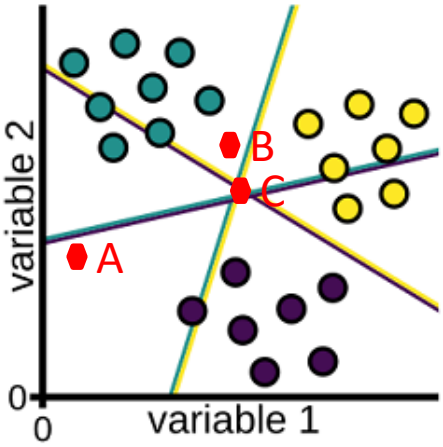
Prediction: choose the class whose classifier outputs the largest value

SVM: more than two classes

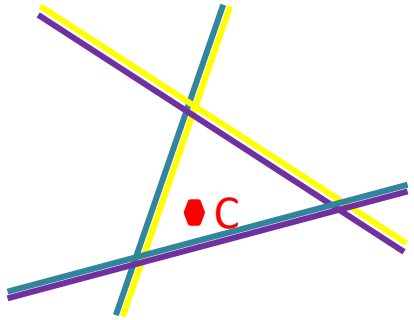
- N-class problem

  2. One-vs-One approach

     Training: train a binary SVM classifier for each possible pair of classes



- Classifier (1, 2): Class 1 vs. Class 2
- Classifier (1, 3): Class 1 vs. Class 3
- …
- Classifier (*N*-1, *N*): Class *N*-1 vs. Class *N*



Prediction: Majority voting among all the classifiers

More References

- Christopher J. C. Burges

  A Tutorial on Support Vector Machines for Pattern Recognition (1998).

- Kayleigh Calder, et al.

  Quadratic programming
  https://optimization.cbe.cornell.edu/index.php?title=Quadratic_programming

- S. Boyd and L. Vandenberghe

  Convex Optimization, Chapter 5 (2004)

- John C. Platt

  Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines (1998)