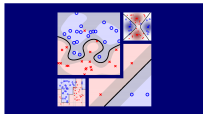


Machine Learning Techniques (機器學習技法)



Lecture 1: Linear Support Vector Machine

Hsuan-Tien Lin (林軒田)

htlin@csie.ntu.edu.tw

Department of Computer Science
& Information Engineering

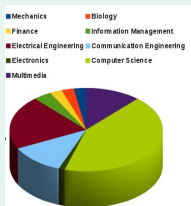
National Taiwan University
(國立台灣大學資訊工程系)



Course History

NTU Version

- 15-17 weeks (2+ hours)
- highly-praised with **English and blackboard teaching**



Coursera Version

- 8 weeks of 'foundations' (previous course) + 8 weeks of 'techniques' (**this course**)
- **Mandarin teaching** to reach more audience in need
- **slides teaching** improved with Coursera's quiz and homework mechanisms

goal: **try** making Coursera version even better than NTU version

Course Design

from Foundations to Techniques

- mixture of philosophical illustrations, key theory, core algorithms, usage in practice, and hopefully jokes :-)
- three major techniques surrounding **feature transforms**:
 - Embedding Numerous Features: how to **exploit** and **regularize** numerous features?
 - inspires **Support Vector Machine** (SVM) model
 - Combining Predictive Features: how to **construct** and **blend** predictive features?
 - inspires **Adaptive Boosting** (AdaBoost) model
 - Distilling Implicit Features: how to **identify** and **learn** implicit features?
 - inspires **Deep Learning** model

allows students to **use ML professionally**

Fun Time

Which of the following description of this course is true?

- ① the course will be taught in Taiwanese
- ② the course will tell me the techniques that create the android Lieutenant Commander Data in Star Trek
- ③ the course will be 16 weeks long
- ④ the course will focus on three major techniques

Fun Time

Which of the following description of this course is true?

- ① the course will be taught in Taiwanese
- ② the course will tell me the techniques that create the android Lieutenant Commander Data in Star Trek
- ③ the course will be 16 weeks long
- ④ the course will focus on three major techniques

Reference Answer: ④

- ① no, my Taiwanese is unfortunately not good enough for teaching (yet)
- ② no, although what we teach may serve as building blocks
- ③ no, unless you have also joined the previous course
- ④ yes, **let's get started!**

Roadmap

1 Embedding Numerous Features: Kernel Models

Lecture 1: Linear Support Vector Machine

- Course Introduction
- Large-Margin Separating Hyperplane
- Standard Large-Margin Problem
- Support Vector Machine
- Reasons behind Large-Margin Hyperplane

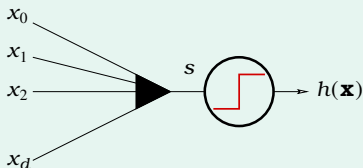
2 Combining Predictive Features: Aggregation Models

3 Distilling Implicit Features: Extraction Models

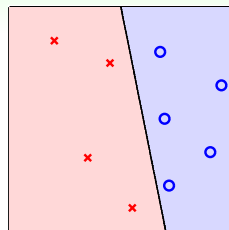
Linear Classification Revisited

PLA/pocket

$$h(\mathbf{x}) = \text{sign}(s)$$



plausible err = 0/1
(small flipping noise)
minimize **specially**

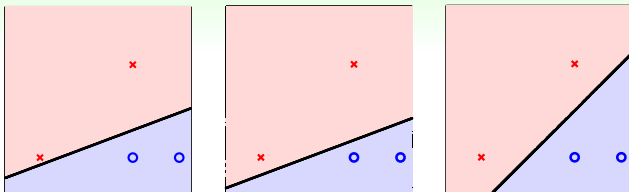


(linear separable)

linear (hyperplane) classifiers:

$$h(\mathbf{x}) = \text{sign}(\mathbf{w}^T \mathbf{x})$$

Which Line Is Best?

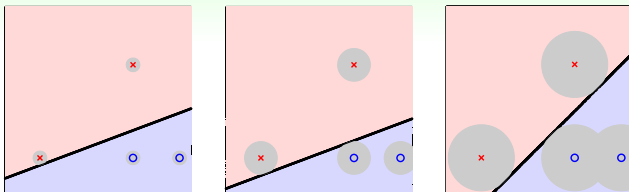


- PLA? depending on randomness
- VC bound? whichever you like!

$$E_{\text{out}}(\mathbf{w}) \leq \underbrace{E_{\text{in}}(\mathbf{w})}_0 + \underbrace{\Omega(\mathcal{H})}_{d_{\text{VC}}=d+1}$$

You? **rightmost one, possibly :-)**

Why Rightmost Hyperplane?



informal argument

if (Gaussian-like) noise on future $\mathbf{x} \approx \mathbf{x}_n$:

\mathbf{x}_n further from hyperplane

distance to closest \mathbf{x}_n

\iff tolerate more noise

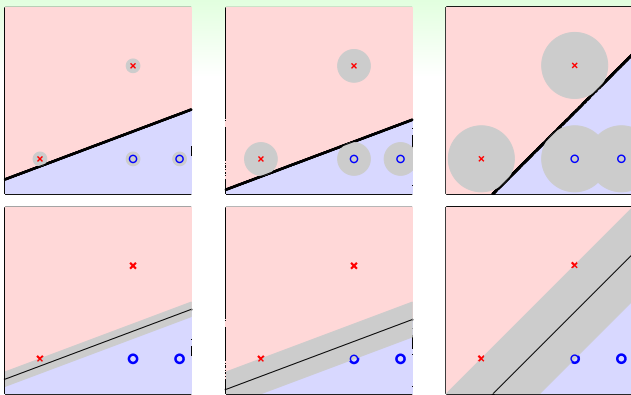
\iff amount of noise tolerance

\iff more robust to overfitting

\iff robustness of hyperplane

rightmost one: **more robust**
because of **larger distance to closest \mathbf{x}_n**

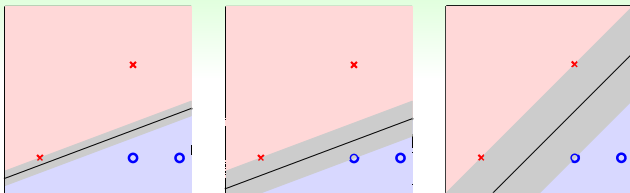
Fat Hyperplane



- **robust** separating hyperplane: **fat**
—far from both sides of examples
- **robustness** \equiv **fatness**: distance to closest \mathbf{x}_n

goal: find **fattest** separating hyperplane

Large-Margin Separating Hyperplane

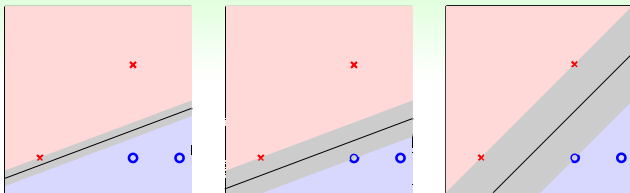


$$\begin{aligned}
 & \max_{\mathbf{w}} \quad \text{fatness}(\mathbf{w}) \\
 & \text{subject to} \quad \mathbf{w} \text{ classifies every } (\mathbf{x}_n, y_n) \text{ correctly} \\
 & \quad \text{fatness}(\mathbf{w}) = \min_{n=1, \dots, N} \text{distance}(\mathbf{x}_n, \mathbf{w})
 \end{aligned}$$

- fatness: formally called **margin**
- correctness: $y_n = \text{sign}(\mathbf{w}^T \mathbf{x}_n)$

goal: find **largest-margin**
separating hyperplane

Large-Margin Separating Hyperplane



$$\begin{aligned}
 & \max_{\mathbf{w}} \quad \text{margin}(\mathbf{w}) \\
 & \text{subject to} \quad \text{every } y_n \mathbf{w}^T \mathbf{x}_n > 0 \\
 & \quad \text{margin}(\mathbf{w}) = \min_{n=1, \dots, N} \text{distance}(\mathbf{x}_n, \mathbf{w})
 \end{aligned}$$

- fatness: formally called **margin**
- **correctness**: $y_n = \text{sign}(\mathbf{w}^T \mathbf{x}_n)$

goal: find **largest-margin separating** hyperplane

Fun Time

Consider two examples $(\mathbf{v}, +1)$ and $(-\mathbf{v}, -1)$ where $\mathbf{v} \in \mathbb{R}^2$ (without padding the $v_0 = 1$). Which of the following hyperplane is the **largest-margin separating** one for the two examples? You are highly encouraged to visualize by considering, for instance, $\mathbf{v} = (3, 2)$.

① $x_1 = 0$

② $x_2 = 0$

③ $v_1 x_1 + v_2 x_2 = 0$

④ $v_2 x_1 + v_1 x_2 = 0$

Fun Time

Consider two examples $(\mathbf{v}, +1)$ and $(-\mathbf{v}, -1)$ where $\mathbf{v} \in \mathbb{R}^2$ (without padding the $v_0 = 1$). Which of the following hyperplane is the **largest-margin separating** one for the two examples? You are highly encouraged to visualize by considering, for instance, $\mathbf{v} = (3, 2)$.

- ① $x_1 = 0$
- ② $x_2 = 0$
- ③ $v_1 x_1 + v_2 x_2 = 0$
- ④ $v_2 x_1 + v_1 x_2 = 0$

Reference Answer: ③

Here the **largest-margin separating** hyperplane (line) must be a perpendicular bisector of the line segment between \mathbf{v} and $-\mathbf{v}$. Hence \mathbf{v} is a normal vector of the largest-margin line. The result can be extended to the more general case of $\mathbf{v} \in \mathbb{R}^d$.

Distance to Hyperplane: Preliminary

$$\begin{aligned}
 & \max_{\mathbf{w}} \quad \text{margin}(\mathbf{w}) \\
 & \text{subject to} \quad \text{every } y_n \mathbf{w}^T \mathbf{x}_n > 0 \\
 & \quad \quad \quad \text{margin}(\mathbf{w}) = \min_{n=1, \dots, N} \text{distance}(\mathbf{x}_n, \mathbf{w})
 \end{aligned}$$

‘shorten’ \mathbf{x} and \mathbf{w}

distance needs w_0 and (w_1, \dots, w_d) differently (to be derived)

$$\begin{aligned}
 b &= w_0 \\
 \begin{bmatrix} | \\ \mathbf{w} \\ | \end{bmatrix} &= \begin{bmatrix} w_1 \\ \vdots \\ w_d \end{bmatrix} ; \quad \begin{bmatrix} | \\ \mathbf{x} \\ | \end{bmatrix} = \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix}
 \end{aligned}$$

~~x_0~~

for this part: $h(\mathbf{x}) = \text{sign}(\mathbf{w}^T \mathbf{x} + b)$

Distance to Hyperplane

want: distance(\mathbf{x} , b , \mathbf{w}), with hyperplane $\mathbf{w}^T \mathbf{x}' + b = 0$

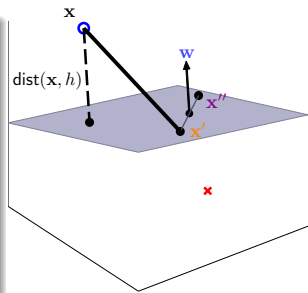
consider \mathbf{x}' , \mathbf{x}'' on hyperplane

① $\mathbf{w}^T \mathbf{x}' = -b$, $\mathbf{w}^T \mathbf{x}'' = -b$

② $\mathbf{w} \perp$ hyperplane:

$$\begin{pmatrix} \mathbf{w}^T & \underbrace{(\mathbf{x}'' - \mathbf{x}')}_{\text{vector on hyperplane}} \end{pmatrix} = 0$$

③ distance = project $(\mathbf{x} - \mathbf{x}')$ to \perp hyperplane



$$\text{distance}(\mathbf{x}, b, \mathbf{w}) = \left| \frac{\mathbf{w}^T}{\|\mathbf{w}\|} (\mathbf{x} - \mathbf{x}') \right| \stackrel{\textcircled{1}}{=} \frac{1}{\|\mathbf{w}\|} |\mathbf{w}^T \mathbf{x} + b|$$

Distance to **Separating** Hyperplane

$$\text{distance}(\mathbf{x}, \mathbf{b}, \mathbf{w}) = \frac{1}{\|\mathbf{w}\|} |\mathbf{w}^T \mathbf{x} + \mathbf{b}|$$

- separating** hyperplane: for every n

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

- distance to **separating** hyperplane:

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

$$\begin{array}{ll} \max_{\mathbf{b}, \mathbf{w}} & \text{margin}(\mathbf{b}, \mathbf{w}) \\ \text{subject to} & \text{every } y_n(\mathbf{w}^T \mathbf{x}_n + \mathbf{b}) > 0 \\ & \text{margin}(\mathbf{b}, \mathbf{w}) = \min_{n=1, \dots, N} \frac{1}{\|\mathbf{w}\|} y_n(\mathbf{w}^T \mathbf{x}_n + \mathbf{b}) \end{array}$$

Margin of **Special** Separating Hyperplane

$$\begin{aligned}
 & \max_{b, \mathbf{w}} \quad \text{margin}(\mathbf{b}, \mathbf{w}) \\
 & \text{subject to} \quad \text{every } y_n(\mathbf{w}^T \mathbf{x}_n + b) > 0 \\
 & \quad \quad \quad \text{margin}(\mathbf{b}, \mathbf{w}) = \min_{n=1, \dots, N} \frac{1}{\|\mathbf{w}\|} y_n(\mathbf{w}^T \mathbf{x}_n + b)
 \end{aligned}$$

- $\mathbf{w}^T \mathbf{x} + b = 0$ same as $3\mathbf{w}^T \mathbf{x} + 3b = 0$: scaling does not matter
- **special** scaling: only consider separating (b, \mathbf{w}) such that

$$\min_{n=1, \dots, N} y_n(\mathbf{w}^T \mathbf{x}_n + b) = 1 \implies \text{margin}(\mathbf{b}, \mathbf{w}) = \frac{1}{\|\mathbf{w}\|}$$

$$\begin{aligned}
 & \max_{b, \mathbf{w}} \quad \frac{1}{\|\mathbf{w}\|} \\
 & \text{subject to} \quad \text{every } y_n(\mathbf{w}^T \mathbf{x}_n + b) > 0 \\
 & \quad \quad \quad \min_{n=1, \dots, N} y_n(\mathbf{w}^T \mathbf{x}_n + b) = 1
 \end{aligned}$$

Standard Large-Margin Hyperplane Problem

$$\max_{b, \mathbf{w}} \frac{1}{\|\mathbf{w}\|} \quad \text{subject to} \quad \min_{n=1, \dots, N} y_n(\mathbf{w}^T \mathbf{x}_n + b) = 1$$

necessary constraints: $y_n(\mathbf{w}^T \mathbf{x}_n + b) \geq 1$ for all n

original constraint: $\min_{n=1, \dots, N} y_n(\mathbf{w}^T \mathbf{x}_n + b) = 1$
 want: optimal (b, \mathbf{w}) **here (inside)**

if optimal (b, \mathbf{w}) outside, e.g. $y_n(\mathbf{w}^T \mathbf{x}_n + b) > 1.126$ for all n
 —can scale (b, \mathbf{w}) to “more optimal” $(\frac{b}{1.126}, \frac{\mathbf{w}}{1.126})$ **(contradiction!)**

final change: $\max \implies \min$, remove $\sqrt{}$, add $\frac{1}{2}$

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

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

Fun Time

Consider three examples $(\mathbf{x}_1, +1)$, $(\mathbf{x}_2, +1)$, $(\mathbf{x}_3, -1)$, where $\mathbf{x}_1 = (3, 0)$, $\mathbf{x}_2 = (0, 4)$, $\mathbf{x}_3 = (0, 0)$. In addition, consider a hyperplane $x_1 + x_2 = 1$. Which of the following is not true?

- ① the hyperplane is a separating one for the three examples
- ② the distance from the hyperplane to \mathbf{x}_1 is 2
- ③ the distance from the hyperplane to \mathbf{x}_3 is $\frac{1}{\sqrt{2}}$
- ④ the example that is closest to the hyperplane is \mathbf{x}_3

Fun Time

Consider three examples $(\mathbf{x}_1, +1)$, $(\mathbf{x}_2, +1)$, $(\mathbf{x}_3, -1)$, where $\mathbf{x}_1 = (3, 0)$, $\mathbf{x}_2 = (0, 4)$, $\mathbf{x}_3 = (0, 0)$. In addition, consider a hyperplane $x_1 + x_2 = 1$. Which of the following is not true?

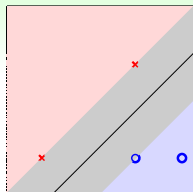
- ① the hyperplane is a separating one for the three examples
- ② the distance from the hyperplane to \mathbf{x}_1 is 2
- ③ the distance from the hyperplane to \mathbf{x}_3 is $\frac{1}{\sqrt{2}}$
- ④ the example that is closest to the hyperplane is \mathbf{x}_3

Reference Answer: ②

The distance from the hyperplane to \mathbf{x}_1 is $\frac{1}{\sqrt{2}}(3 + 0 - 1) = \sqrt{2}$.

Solving a Particular Standard Problem

$$\begin{aligned} \min_{b, \mathbf{w}} \quad & \frac{1}{2} \mathbf{w}^T \mathbf{w} \\ \text{subject to} \quad & y_n(\mathbf{w}^T \mathbf{x}_n + b) \geq 1 \text{ for all } n \end{aligned}$$



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

$$\begin{aligned} -b &\geq 1 & (i) \\ -2w_1 - 2w_2 - b &\geq 1 & (ii) \\ 2w_1 + b &\geq 1 & (iii) \\ 3w_1 + b &\geq 1 & (iv) \end{aligned}$$

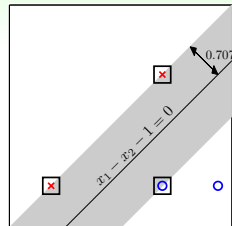
- $\left\{ \begin{array}{ll} (i) & \& (iii) \\ (ii) & \& (iii) \end{array} \right\} \Rightarrow \begin{array}{l} w_1 \geq +1 \\ w_2 \leq -1 \end{array} \Rightarrow \frac{1}{2} \mathbf{w}^T \mathbf{w} \geq 1$
- $(w_1 = 1, w_2 = -1, b = -1)$ at **lower bound** and satisfies (i) – (iv)

$$g_{\text{SVM}}(\mathbf{x}) = \text{sign}(x_1 - x_2 - 1): \text{SVM? :-)}$$

Support Vector Machine (SVM)

optimal solution: $(w_1 = 1, w_2 = -1, b = -1)$

$$\text{margin}(b, \mathbf{w}) = \frac{1}{\|\mathbf{w}\|} = \frac{1}{\sqrt{2}}$$



- examples on boundary: 'locates' fattest hyperplane
other examples: **not needed**
- call boundary example **support vector** (candidate)

support vector machine (SVM):
learn **fattest hyperplanes**
(with help of **support vectors**)

Solving General SVM

$$\begin{aligned} \min_{b, \mathbf{w}} \quad & \frac{1}{2} \mathbf{w}^T \mathbf{w} \\ \text{subject to} \quad & y_n(\mathbf{w}^T \mathbf{x}_n + b) \geq 1 \text{ for all } n \end{aligned}$$

- **not easy manually, of course :-)**
 - gradient decent? **not easy with constraints**
 - luckily:
 - (convex) quadratic objective function of (b, \mathbf{w})
 - linear constraints of (b, \mathbf{w})
- **quadratic programming**

quadratic programming (QP):
'easy' optimization problem

Quadratic Programming

optimal $(\mathbf{b}, \mathbf{w}) = ?$

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

subject to $y_n(\mathbf{w}^T \mathbf{x}_n + \mathbf{b}) \geq 1,$
for $n = 1, 2, \dots, N$

optimal $\mathbf{u} \leftarrow \text{QP}(\mathbf{Q}, \mathbf{p}, \mathbf{A}, \mathbf{c})$

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

subject to $\mathbf{a}_m^T \mathbf{u} \geq \mathbf{c}_m,$
for $m = 1, 2, \dots, M$

objective function: $\mathbf{u} = \begin{bmatrix} \mathbf{b} \\ \mathbf{w} \end{bmatrix}; \mathbf{Q} = \begin{bmatrix} \mathbf{0} & \mathbf{0}_d^T \\ \mathbf{0}_d & \mathbf{I}_d \end{bmatrix}; \mathbf{p} = \mathbf{0}_{d+1}$

constraints: $\mathbf{a}_n^T = y_n \begin{bmatrix} 1 & \mathbf{x}_n^T \end{bmatrix}; \mathbf{c}_n = 1; M = N$

SVM with general QP solver:
easy **if you've read the manual :-)**

SVM with QP Solver

Linear Hard-Margin SVM Algorithm

- 1 $Q = \begin{bmatrix} 0 & \mathbf{0}_d^T \\ \mathbf{0}_d & \mathbf{I}_d \end{bmatrix}$; $\mathbf{p} = \mathbf{0}_{d+1}$; $\mathbf{a}_n^T = y_n [1 \quad \mathbf{x}_n^T]$; $c_n = 1$
- 2 $\begin{bmatrix} b \\ \mathbf{w} \end{bmatrix} \leftarrow \text{QP}(Q, \mathbf{p}, A, \mathbf{c})$
- 3 return b & \mathbf{w} as g_{SVM}

- **hard-margin**: nothing violate 'fat boundary'
- **linear**: \mathbf{x}_n

want **non-linear**?

$\mathbf{z}_n = \Phi(\mathbf{x}_n)$ —**remember? :-)**

Fun Time

Consider two negative examples with $\mathbf{x}_1 = (0, 0)$ and $\mathbf{x}_2 = (2, 2)$; two positive examples with $\mathbf{x}_3 = (2, 0)$ and $\mathbf{x}_4 = (3, 0)$, as shown on page 17 of the slides. Define \mathbf{u} , Q , \mathbf{p} , c_n as those listed on page 20 of the slides. What are \mathbf{a}_n^T that need to be fed into the QP solver?

① $\mathbf{a}_1^T = [-1, 0, 0]$, $\mathbf{a}_2^T = [-1, 2, 2]$, $\mathbf{a}_3^T = [-1, 2, 0]$, $\mathbf{a}_4^T = [-1, 3, 0]$

② $\mathbf{a}_1^T = [1, 0, 0]$, $\mathbf{a}_2^T = [1, -2, -2]$, $\mathbf{a}_3^T = [-1, 2, 0]$, $\mathbf{a}_4^T = [-1, 3, 0]$

③ $\mathbf{a}_1^T = [1, 0, 0]$, $\mathbf{a}_2^T = [1, 2, 2]$, $\mathbf{a}_3^T = [1, 2, 0]$, $\mathbf{a}_4^T = [1, 3, 0]$

④ $\mathbf{a}_1^T = [-1, 0, 0]$, $\mathbf{a}_2^T = [-1, -2, -2]$, $\mathbf{a}_3^T = [1, 2, 0]$, $\mathbf{a}_4^T = [1, 3, 0]$

Fun Time

Consider two negative examples with $\mathbf{x}_1 = (0, 0)$ and $\mathbf{x}_2 = (2, 2)$; two positive examples with $\mathbf{x}_3 = (2, 0)$ and $\mathbf{x}_4 = (3, 0)$, as shown on page 17 of the slides. Define \mathbf{u} , Q , \mathbf{p} , c_n as those listed on page 20 of the slides. What are \mathbf{a}_n^T that need to be fed into the QP solver?

① $\mathbf{a}_1^T = [-1, 0, 0]$, $\mathbf{a}_2^T = [-1, 2, 2]$, $\mathbf{a}_3^T = [-1, 2, 0]$, $\mathbf{a}_4^T = [-1, 3, 0]$

② $\mathbf{a}_1^T = [1, 0, 0]$, $\mathbf{a}_2^T = [1, -2, -2]$, $\mathbf{a}_3^T = [-1, 2, 0]$, $\mathbf{a}_4^T = [-1, 3, 0]$

③ $\mathbf{a}_1^T = [1, 0, 0]$, $\mathbf{a}_2^T = [1, 2, 2]$, $\mathbf{a}_3^T = [1, 2, 0]$, $\mathbf{a}_4^T = [1, 3, 0]$

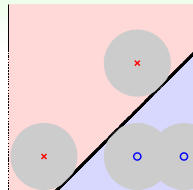
④ $\mathbf{a}_1^T = [-1, 0, 0]$, $\mathbf{a}_2^T = [-1, -2, -2]$, $\mathbf{a}_3^T = [1, 2, 0]$, $\mathbf{a}_4^T = [1, 3, 0]$

Reference Answer: ④

We need $\mathbf{a}_n^T = y_n \begin{bmatrix} 1 & \mathbf{x}_n^T \end{bmatrix}$.

Why Large-Margin Hyperplane?

$$\begin{aligned} \min_{b, \mathbf{w}} \quad & \frac{1}{2} \mathbf{w}^T \mathbf{w} \\ \text{subject to} \quad & y_n(\mathbf{w}^T \mathbf{z}_n + b) \geq 1 \text{ for all } n \end{aligned}$$



	minimize	constraint
regularization	E_{in}	$\mathbf{w}^T \mathbf{w} \leq C$
SVM	$\mathbf{w}^T \mathbf{w}$	$E_{\text{in}} = 0$ [and more]

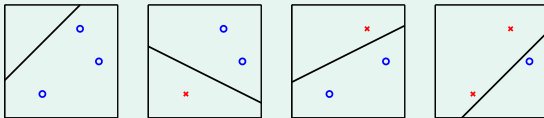
SVM (large-margin hyperplane):
'weight-decay regularization' within $E_{\text{in}} = 0$

Large-Margin Restricts Dichotomies

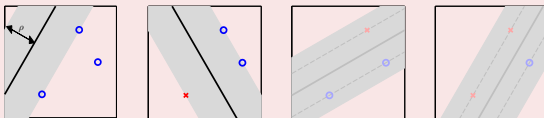
consider 'large-margin algorithm' \mathcal{A}_ρ :

either **returns g with $\text{margin}(g) \geq \rho$ (if exists)**, or 0 otherwise

\mathcal{A}_0 : like PLA \implies shatter 'general' 3 inputs



$\mathcal{A}_{1.126}$: more strict than SVM \implies cannot shatter any 3 inputs



fewer dichotomies \implies smaller 'VC dim.' \implies **better generalization**

VC Dimension of Large-Margin Algorithm

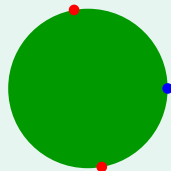
fewer dichotomies \implies smaller ‘**VC dim.**’

considers $d_{\text{VC}}(\mathcal{A}_\rho)$ [**data-dependent, need more than VC**]

instead of $d_{\text{VC}}(\mathcal{H})$ [data-independent, covered by VC]

$d_{\text{VC}}(\mathcal{A}_\rho)$ when \mathcal{X} = unit circle in \mathbb{R}^2

- $\rho = 0$: just perceptrons ($d_{\text{VC}} = 3$)
- $\rho > \frac{\sqrt{3}}{2}$: cannot shatter any 3 inputs
($d_{\text{VC}} < 3$)
—some inputs must be of **distance** $\leq \sqrt{3}$



generally, when \mathcal{X} in **radius- R** hyperball:

$$d_{\text{VC}}(\mathcal{A}_\rho) \leq \min \left(\frac{R^2}{\rho^2}, d \right) + 1 \leq \underbrace{d+1}_{d_{\text{VC}}(\text{perceptrons})}$$

Benefits of Large-Margin Hyperplanes

	large-margin hyperplanes	hyperplanes	hyperplanes + feature transform Φ
#	even fewer	not many	many
boundary	simple	simple	sophisticated

- **not many** good, for d_{VC} and generalization
- **sophisticated** good, for possibly better E_{in}

a new possibility: non-linear SVM

	large-margin hyperplanes + numerous feature transform Φ
#	not many
boundary	sophisticated

Fun Time

Consider running the 'large-margin algorithm' \mathcal{A}_ρ with $\rho = \frac{1}{4}$ on a \mathcal{Z} -space such that $\mathbf{z} = \Phi(\mathbf{x})$ is of 1126 dimensions (excluding z_0) and $\|\mathbf{z}\| \leq 1$. What is the upper bound of $d_{\text{VC}}(\mathcal{A}_\rho)$ when calculated by $\min\left(\frac{R^2}{\rho^2}, d\right) + 1$?

- ① 5
- ② 17
- ③ 1126
- ④ 1127

Fun Time

Consider running the 'large-margin algorithm' \mathcal{A}_ρ with $\rho = \frac{1}{4}$ on a \mathcal{Z} -space such that $\mathbf{z} = \Phi(\mathbf{x})$ is of 1126 dimensions (excluding z_0) and $\|\mathbf{z}\| \leq 1$. What is the upper bound of $d_{\text{VC}}(\mathcal{A}_\rho)$ when calculated by $\min\left(\frac{R^2}{\rho^2}, d\right) + 1$?

- ① 5
- ② 17
- ③ 1126
- ④ 1127

Reference Answer: ②

By the description, $d = 1126$ and $R = 1$. So the upper bound is simply 17.

Summary

① Embedding Numerous Features: Kernel Models

Lecture 1: Linear Support Vector Machine

- Course Introduction
from foundations to techniques
- Large-Margin Separating Hyperplane
intuitively more robust against noise
- Standard Large-Margin Problem
minimize 'length of w ' at special separating scale
- Support Vector Machine
'easy' via quadratic programming
- Reasons behind Large-Margin Hyperplane
fewer dichotomies and better generalization

- **next: solving non-linear Support Vector Machine**

② Combining Predictive Features: Aggregation Models

③ Distilling Implicit Features: Extraction Models