Machine Learning

Lecture 14
Linear Support Vector Machine

深度取特徵
再做SVM分類

Chen-Kuo Chiang (江 振 國)
ckchiang@cs.ccu.edu.tw

中正大學 資訊工程學系

Course Introduction

- 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

The Storyline

1. Embedding Numerous Features: Kernel Models

   Linear Support Vector Machine

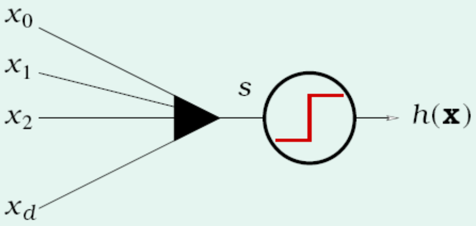
   - Course Introduction
   - Large-Margin Separating Hyperplane
   - Standard Large-Margin Problem
   - Support Vector Machine
   - Reasons behind Large-Margin Hyperplane

2. Combining Predictive Features: Aggregation Models
3. Distilling Implicit Features: Extraction Models

Linear Classification Revisited

PLA/pocket

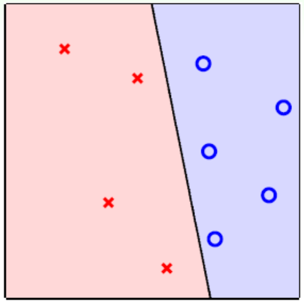
$$h(\mathbf{x}) = \text{sign}(s)$$



plausible $\boxed{\text{err} = 0/1}$

(small flipping noise)

minimize **specially**

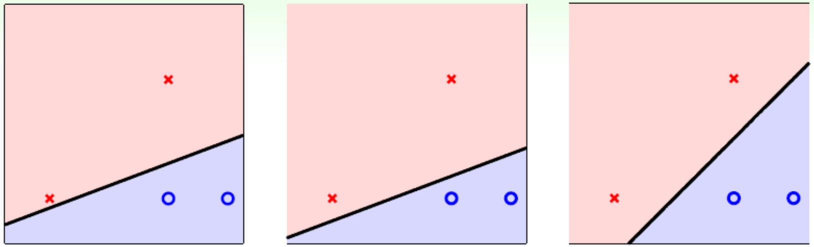


(linear separable)

linear (hyperplane) classifiers:

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

Which Line Is Best?

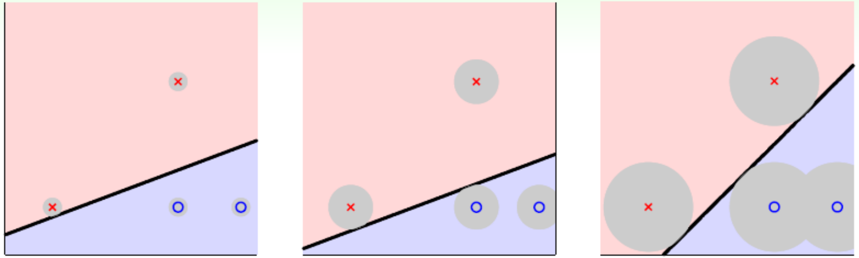


- PLA? depending on randomness 一條線將兩者分開
- VC bound? whichever you like!

$$E_{\text{out}}(\mathbf{w}) \leq \underbrace{E_{\text{in}}(\mathbf{w})}_{0} + \underbrace{\Omega(\mathcal{H})}_{d_{\text{VC}}=d+1}$$

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

Why Rightmost Hyperplane?



線距離sample的大小

informal argument

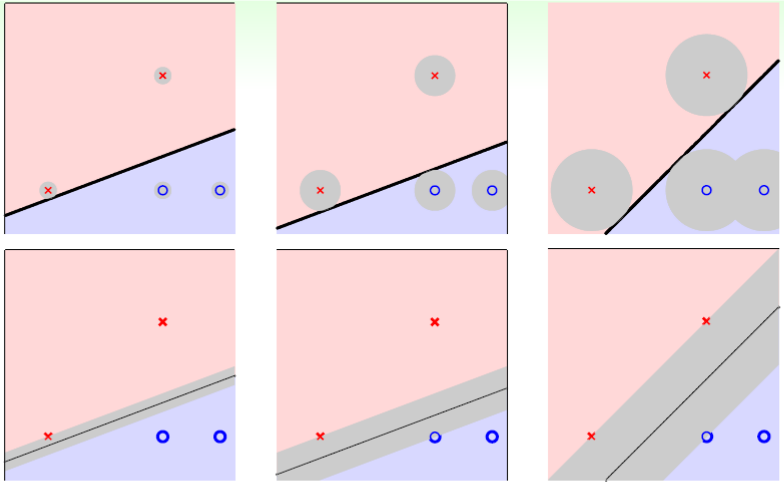
if (Gaussian-like) noise on future $\mathbf{x} \approx \mathbf{x}_n$:

X距離分隔線越遠，可以容忍更多noise

$\mathbf{x}_n$ further from hyperplane	distance to closest $\mathbf{x}_n$
$\Longleftrightarrow$ tolerate more noise	$\Longleftrightarrow$ amount of noise tolerance
$\Longleftrightarrow$ more robust to overfitting	$\Longleftrightarrow$ robustness of hyperplane

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

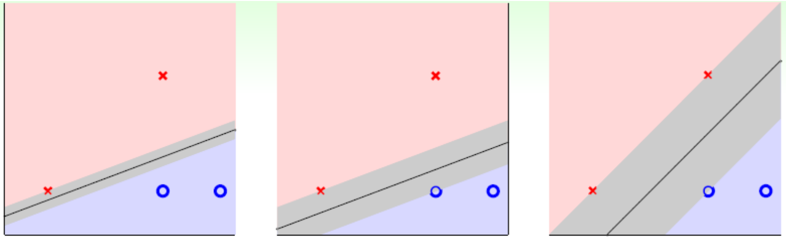
Fat Hyperplane



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

goal: find **fattest** separating hyperplane

Large-Margin Separating Hyperplane



$$\max_{\mathbf{w}} \quad \text{fatness}(\mathbf{w})$$ 找最肥的

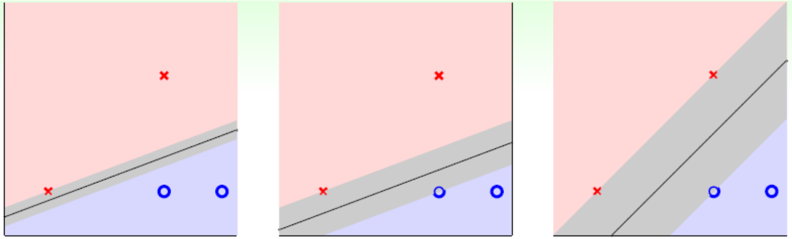
subject to $\quad \mathbf{w}$ classifies every $(\mathbf{x}_n, y_n)$ correctly

$$\text{fatness}(\mathbf{w}) = \min_{n=1,\ldots,N} \text{distance}(\mathbf{x}_n, \mathbf{w})$$ 要找到限制條件

- fatness: formally called **margin** 最大margin
- correctness: $y_n = \text{sign}(\mathbf{w}^T \mathbf{x}_n)$ 分類預測結果要與標準一致

goal: find $\boxed{\textbf{largest-margin}}$ **separating** hyperplane

Large-Margin Separating Hyperplane



$$\max_{\mathbf{w}} \quad \text{margin}(\mathbf{w})$$ 將概念轉換為數學式，再計算出w。

subject to every $y_n \mathbf{w}^T \mathbf{x}_n > 0$ 代表資料做對

$$\text{margin}(\mathbf{w}) = \min_{n=1,\ldots,N} \text{distance}(\mathbf{x}_n, \mathbf{w})$$

- 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. $v_2 x_1 + v_1 x_2 = 0$

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

$$\max_{\mathbf{w}} \quad \text{margin}(\mathbf{w})$$

$$\text{subject to} \quad \text{every } y_n \mathbf{w}^T \mathbf{x}_n > 0$$

$$\text{margin}(\mathbf{w}) = \min_{n=1,\ldots,N} \text{distance}(\mathbf{x}_n, \mathbf{w})$$

'shorten' $\mathbf{x}$ and $\mathbf{w}$

distance needs $\boxed{w_0}$ and $(w_1, \ldots, w_d)$ differently (to be derived)

將常數獨立出來

$$\boxed{b} = w_0 \qquad\qquad \cancel{x_0 = 1}$$

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

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

Distance to Hyperplane

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

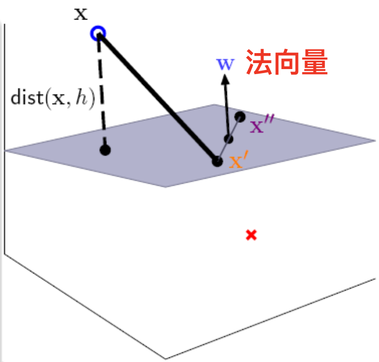
consider $\mathbf{x}'$, $\mathbf{x}''$ on hyperplane

1. $\mathbf{w}^T\mathbf{x}' = -b$, $\mathbf{w}^T\mathbf{x}'' = -b$ 落在平面上結果為0
2. $\mathbf{w} \perp$ hyperplane:
垂直

$$\left(\mathbf{w}^T \underbrace{(\mathbf{x}'' - \mathbf{x}')}_{\text{vector on hyperplane}}\right) = 0$$

兩個向量相互垂直為0

3. distance = project ($\mathbf{x} - \mathbf{x}'$) to $\perp$ hyperplane
投影在垂直處



$$\text{distance}(\mathbf{x}, b, \mathbf{w}) = \left|\underbrace{\frac{\mathbf{w}^T}{\|\mathbf{w}\|}}_{\text{長度}}(\mathbf{x} - \mathbf{x}')\right| \overset{①}{=} \frac{1}{\|\mathbf{w}\|}\left|\underline{\mathbf{w}^T\mathbf{x} + b}\right|$$

Distance to **Separating** Hyperplane

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

- **separating** hyperplane: for every n

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

- distance to **separating** hyperplane:

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

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

Margin of **Special** Separating Hyperplane

$$\max_{b,\mathbf{w}} \quad \text{margin}(b, \mathbf{w})$$

$$\text{subject to} \quad \text{every } y_n(\mathbf{w}^T\mathbf{x}_n + b) > 0$$

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

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

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

$$\max_{b,\mathbf{w}} \quad \frac{1}{\|\mathbf{w}\|}$$

$$\text{subject to} \quad \text{every } y_n(\mathbf{w}^T\mathbf{x}_n + b) > 0$$

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

Standard Large-Margin Hyperplane Problem

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

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

original constraint: $\min_{n=1,\ldots,N} \quad y_n(\mathbf{w}^T\mathbf{x}_n + b) = 1$
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 $\Longrightarrow$ min, remove $\sqrt{\ }$, add $\frac{1}{2}$

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

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

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 $\frac{1}{\sqrt{2}}(3 + 0 - 1) = \sqrt{2}$.

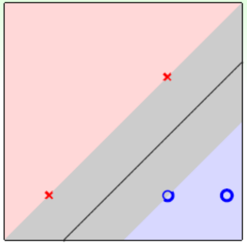
點$P(x_o, y_o)$到直線$L: ax + by + c = 0$的距離寫

$$d = \frac{|ax_o + by_o + c|}{\sqrt{a^2 + b^2}}$$

Solving a Particular Standard Problem

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

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



$$\mathrm{X} = \begin{bmatrix} 0 & 0 \\ 2 & 2 \\ 2 & 0 \\ 3 & 0 \end{bmatrix} \quad \mathbf{y} = \begin{bmatrix} -1 \\ -1 \\ +1 \\ +1 \end{bmatrix} \quad \begin{array}{rrrl} & & -b \geq 1 & (i) \\ -2w_1 & -2w_2 & -b \geq 1 & (ii) \\ 2w_1 & & +b \geq 1 & (iii) \\ 3w_1 & & +b \geq 1 & (iv) \end{array}$$

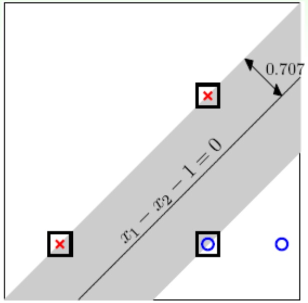
- $\left\{ \begin{array}{lllll} (i) & \& & (iii) & \Longrightarrow & w_1 \geq +1 \\ (ii) & \& & (iii) & \Longrightarrow & w_2 \leq -1 \end{array} \right\} \Longrightarrow \frac{1}{2}\mathbf{w}^T\mathbf{w} \geq 1$
- $(w_1 = 1, w_2 = -1, b = -1)$ at **lower bound** and satisfies $(i) - (iv)$

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

Support Vector Machine (SVM)

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

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

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

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

- **not easy manually, of course :-)**
- gradient descent? **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 $(b, \mathbf{w}) = ?$

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

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

$$\text{for } n = 1, 2, \ldots, N$$

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

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

$$\text{subject to} \quad \mathbf{a}_m^T\mathbf{u} \geq c_m,$$

$$\text{for } m = 1, 2, \ldots, M$$

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

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

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

SVM with QP Solver

Linear Hard-Margin SVM Algorithm

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

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

want **non-linear**?
$\mathbf{z}_n = \mathbf{\Phi}(\mathbf{x}_n)$—**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. $\mathbf{a}_1^T = [-1, 0, 0]$, $\mathbf{a}_2^T = [-1, -2, -2]$, $\mathbf{a}_3^T = [1, 2, 0]$, $\mathbf{a}_4^T = [1, 3, 0]$

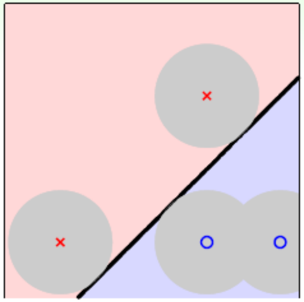
Reference Answer: 4

We need $\mathbf{a}_n^T = y_n \left[\ 1 \quad \mathbf{x}_n^T\ \right]$.

Why Large-Margin Hyperplane?

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

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



	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 $E_{\text{in}} = 0$

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**

Summary

1. Embedding Numerous Features: Kernel Models

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**