در فصل ۲: طراحی طبقه‌بندها براساس چگالی احتمال و توابع احتمال

LINEAR CLASSIFIERS

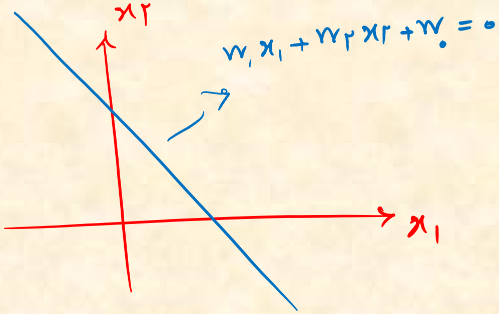
در این فصل: طراحی طبقه بندهای خطی صرفنظر از توزیع های توصیف کننده داده های آموزشی   برتری: سادگی، سبک محاسبه

- The Problem: Consider a two class task with ω_1, ω_2

معادله خطی

➢ $g(\underline{x}) = \underline{w}^T \underline{x} + w_0 = 0 =$ فضای l بعدی

$w_1x_1 + w_2x_2 + \ldots + w_lx_l + w_0 = 0$



➢ Assume $\underline{x}_1, \underline{x}_2$ on the decision hyperplane :

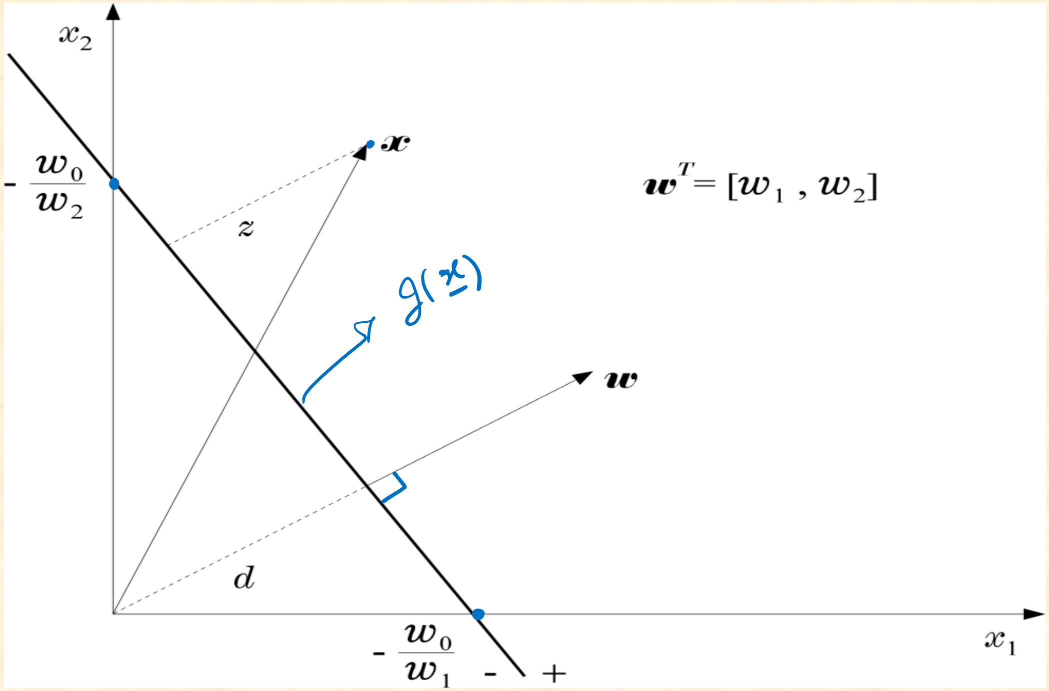
$0 = \underline{w}^T \underline{x}_1 + w_0 = \underline{w}^T \underline{x}_2 + w_0 \Rightarrow$

$\underline{w}^T(\underline{x}_1 - \underline{x}_2) = 0 \;\; \forall \underline{x}_1, \underline{x}_2$ بردار وزن‌ها $\underline{w}$ بر این صفحه تصمیم عمود است.

- Hence:

$$\boxed{\underline{w} \perp \text{on the hyperplane}}$$

$$g(\underline{x}) = \underline{w}^T \underline{x} + w_0 = 0$$



$w_۱x_۱+w_۲x_۲+w_۰=۰$

فاصله عمودی $d = \frac{|w_0|}{\sqrt{w_1^2 + w_2^2}}$, $z = \frac{|g(\underline{x})|}{\sqrt{w_1^2 + w_2^2}}$

۳۸

❖ The Perceptron Algorithm

الگوریتم Perceptron :

جدایی پذیر خطی

فرض: کلاس‌ها جدایی پذیر خطی باشند.

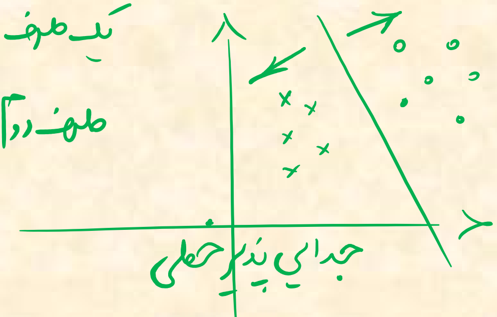
- Assume linearly separable classes, i.e.,

$$\exists\ \underline{w}^*:\ \underline{w}^{*T}\,\underline{x} > 0\quad \forall\,\underline{x}\in\omega_1$$

یک طرف ابر صفحه

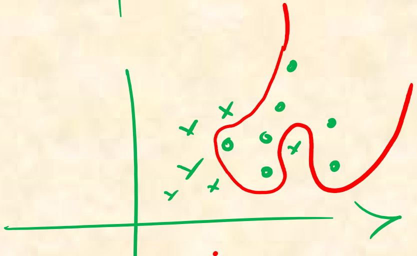
$$\underline{w}^{*T}\,\underline{x} < 0\quad \forall\,\underline{x}\in\omega_2$$

طرف دوم ابر صفحه



جدایی پذیر خطی

- The case $\underline{w}^{*T}\,\underline{x} + w_0^*$ falls under the above formulation, since



X جدایی پذیر خطی نیست

- $\underline{w}' \equiv \begin{bmatrix}\underline{w}^*\\ w_0^*\end{bmatrix},\ \underline{x}' = \begin{bmatrix}\underline{x}\\ 1\end{bmatrix}$ → $\underline{w}^{*T}\underline{x} + w_0^* = 0$

- $\underline{w}^{*T}\,\underline{x} + w_0^* = \boxed{\underline{w}'^{T}\,\underline{x}' = 0}$

هدف: راه حلی برای به دست آوردن ابرصفحه $\underline{w}$ که دو کلاس را کاملاً از هم جدا کند.

- Our goal: Compute a solution, i.e., a hyperplane $\underline{w}$, so that

$$\underline{w}^T \underline{x} (><) 0 \quad \underline{x} \in \begin{cases} \omega_1 \\ \omega_2 \end{cases}$$

  - The steps
    - Define a cost function to be minimized
    - Choose an algorithm to minimize the cost function
    - The minimum corresponds to a solution

➢ The Cost Function

تابع هزینه perceptron:

$$\Rightarrow \quad J(\underline{w}) = \sum_{\underline{x}\in Y} (\delta_x \underline{w}^T \underline{x})$$

- Where Y is the subset of the vectors wrongly classified by $\underline{w}$. When Y=(empty set) a solution is achieved and

Y: زیرمجموعه‌ای از بردارها که به اشتباه طبقه‌بندی شده‌اند.

وقتی $Y=\phi$ شود به جواب رسیده‌ایم:

- $Y=\phi \rightarrow \quad J(\underline{w}) = 0$

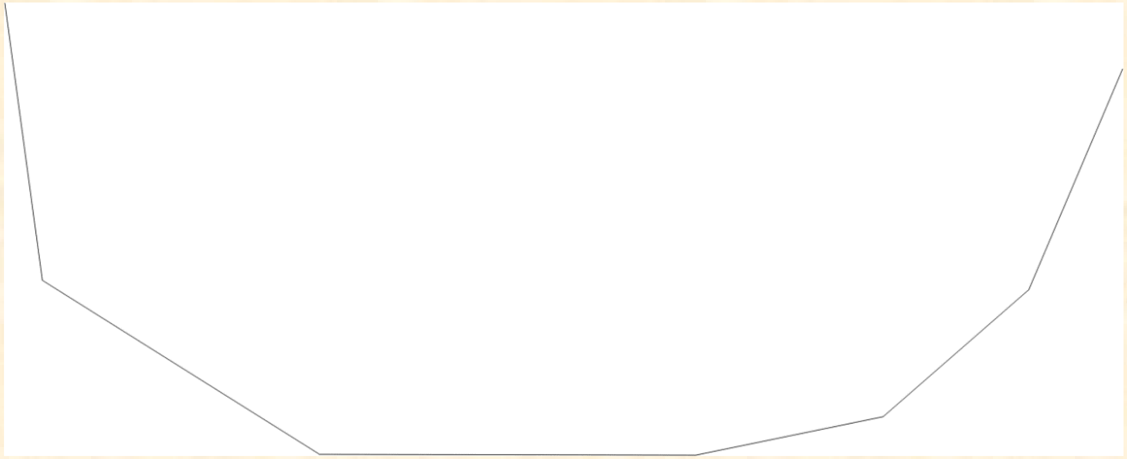
- $$\left\{ \begin{array}{l} \delta_x = -1 \text{ if } \overbrace{\underline{x}\in Y \text{ and } \underline{x}\in\omega_1}^{\underline{w}^T\underline{x}<0} \\ \delta_x = +1 \text{ if } \underbrace{\underline{x}\in Y \text{ and } \underline{x}\in\omega_2}_{\underline{w}^T\underline{x}>0} \end{array} \right.$$

$\underline{w}^T\underline{x} > 0 \rightarrow x\in\omega_1$

$\underline{w}^T\underline{x} < 0 \rightarrow x\in\omega_2$

- $J(\underline{w}) \geq 0$

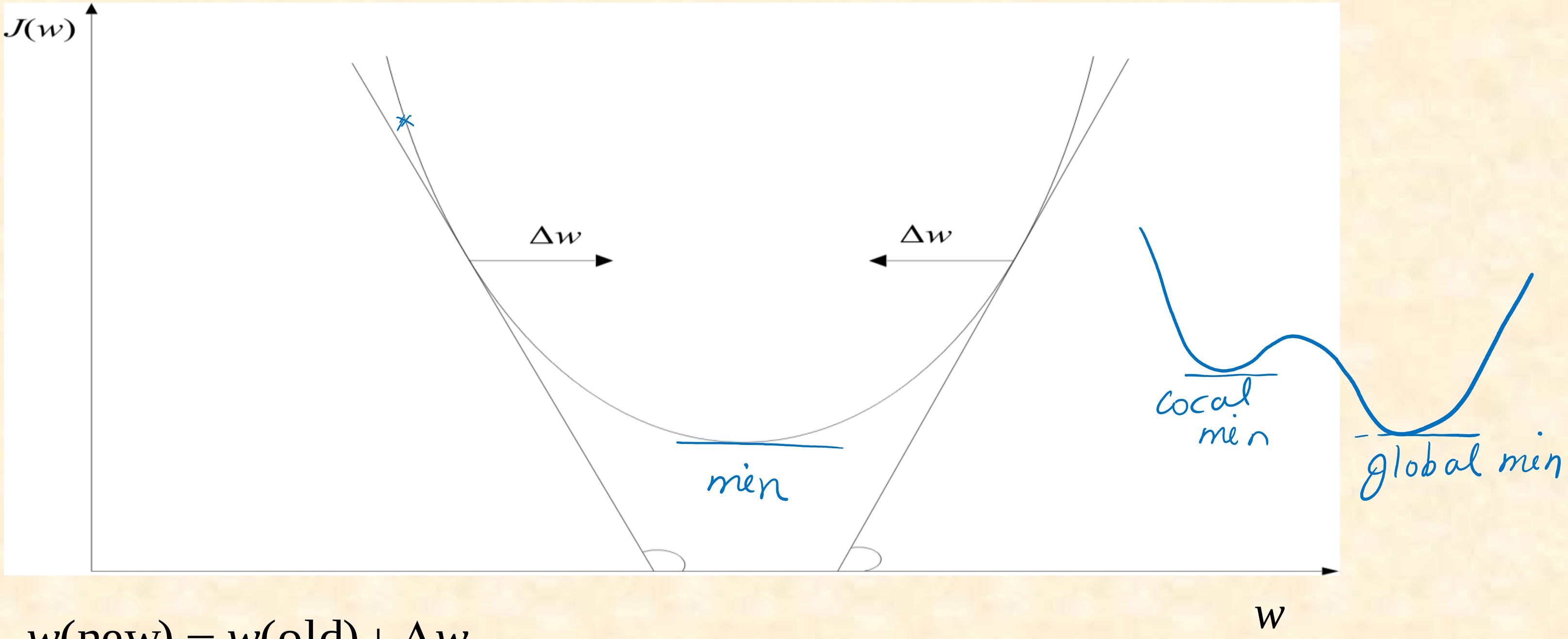
- $J(\underline{w})$ is piecewise linear (WHY?)



➢ The Algorithm

- The philosophy of the gradient descent is adopted.

گرادیانِ نزولی



$$\underline{w}(\text{new}) = \underline{w}(\text{old}) + \Delta\underline{w}$$

$$\Delta\underline{w} = -\mu \frac{\partial J(\underline{w})}{\partial \underline{w}} \Big|_{\underline{w} = \underline{w}(\text{old})}$$

$$\Delta\underline{w} = -\mu \frac{\partial J(\underline{w})}{\partial \underline{w}} \Big|_{\underline{w} = \underline{w}_{old}}$$

μ: پارامتر تنظیم سرعت همگرایی

- Wherever valid

$$\Rightarrow \underline{w}_{new} = \underline{w}_{old} + \Delta\underline{w}$$

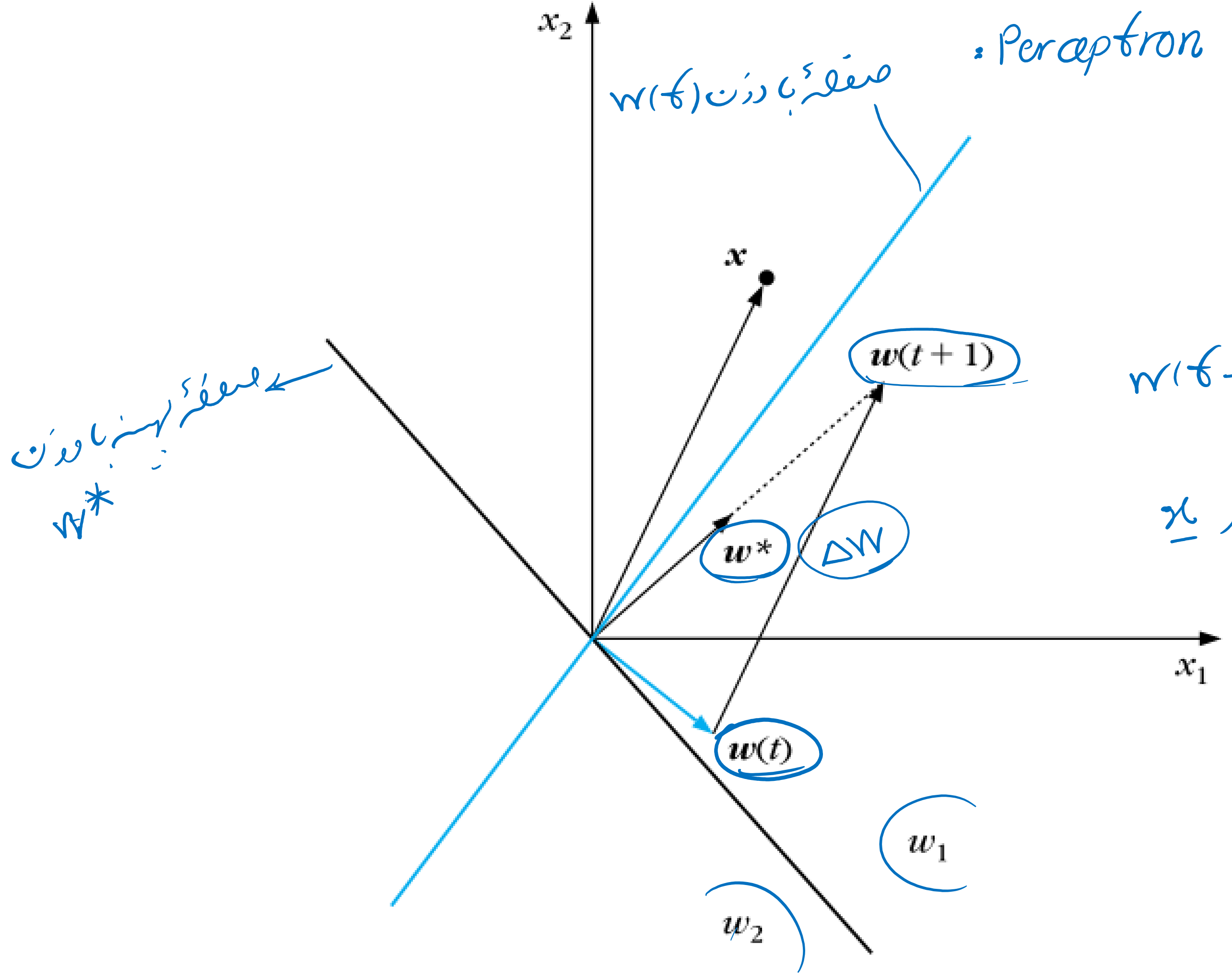
$$\frac{\partial J(\underline{w})}{\partial \underline{w}} = \frac{\partial}{\partial \underline{w}} \Big(\sum_{\underline{x}\in Y} \delta_x \underline{w}^T \underline{x}\Big) = \sum_{\underline{x}\in Y} \delta_x \underline{x}$$

- $$\boxed{\underline{w}(t+1) = \underline{w}(t) - \rho_t \sum_{\underline{x}\in Y} \delta_x \underline{x}}$$

ρ_t: پارامتر تنظیم
دنباله‌ای از اعداد حقیقی
مثبت

This is the celebrated Perceptron Algorithm

تفسیر هندسی الگوریتم Perceptron :



$$w(t+1) = w(t) + \Delta W$$

ΔW در جهت بردار $\underline{x}$

FIGURE 3.2

Geometric interpretation of the perceptron algorithm. The update of the weight vector is in the direction of $\boldsymbol{x}$ in order to turn the decision hyperplane to include $\boldsymbol{x}$ in the correct class.

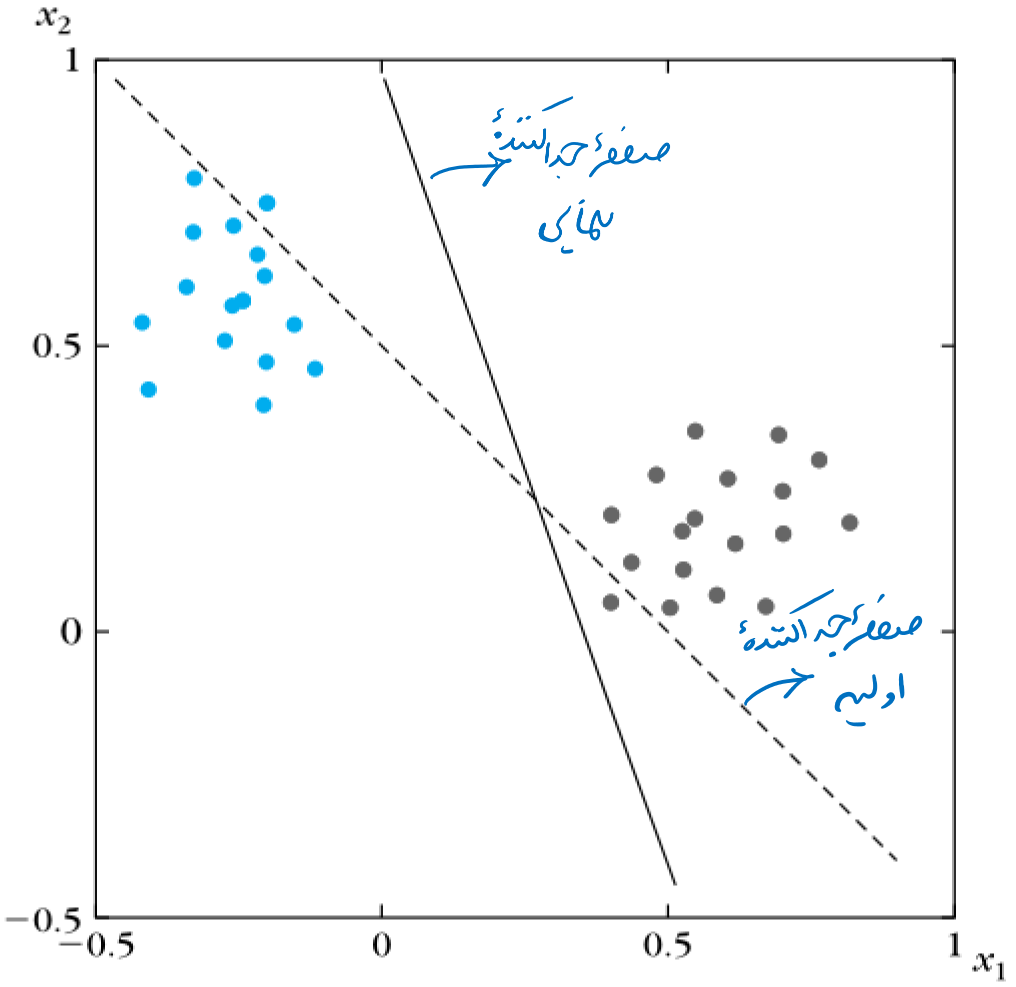


FIGURE 3.3

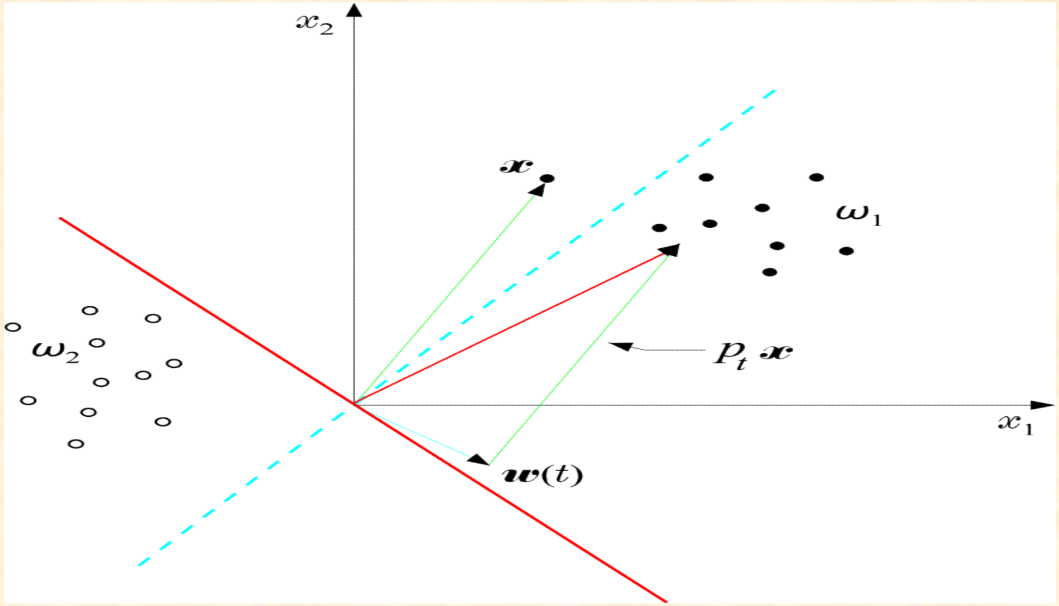
An example of the perceptron algorithm. After the update of the weight vector, the hyperplane is turned from its initial location (dotted line) to the new one (full line), and all points are correctly classified.

الگوریتم پرسپترون :

```
- مقدار دهی اولیه $\underline{w}(0)$ به صورت تصادفی
- انتخاب $\rho_0$
- $t=0$
حلقهٔ تکرار
    - $Y=\phi$
    - for $i=1$ to $N$
          if $\delta_{x_i} w(t)^T \underline{x}_i \geq 0$ then $Y = Y \cup \{\underline{x}_i\}$
      end
    - $\boxed{w(t+1) = w(t) - \rho_t \sum_{x\in Y} \delta_x \underline{x}}$
    - تنظیم $\rho_t$
    - $t=t+1$
تا زمانی که $Y=\phi$
```

حلقهٔ for برای تشخیص نمونه های به اشتباه طبقه بندی شده با بردار وزن های $w(t)$

- An example:



$$\underline{w}(t+1) = \underline{w}(t) + \rho_t \underline{x}$$
$$= \underline{w}(t) - \rho_t \delta_x \underline{x} \quad (\delta_x = -1)$$

همراهی :

- The perceptron algorithm **converges** in a **finite** number of iteration steps to a solution if

$$\lim_{t\to\infty} \sum_{k=0}^{t} \rho_k \to \infty, \qquad \lim_{t\to\infty} \sum_{k=0}^{t} \rho_k^{\,2} < +\infty$$

$$\text{e.g.,:} \boxed{\rho_t = \frac{c}{t}}$$

❖ A useful variant of the perceptron algorithm

فرم‌های دیگر الگوریتم پرسپترون:

⇒ Reward & Punishment تنبیه و پاداش

$$\underline{w}(t+1) = \underline{w}(t) + \rho \underline{x}_{(t)} \ , \begin{cases} \underline{w}^T(t)\underline{x}_{(t)} \leq 0 \\ \underline{x}_{(t)} \in \omega_1 \end{cases}$$

طبقه‌بندی به اشتباه

$$\underline{w}(t+1) = \underline{w}(t) - \rho \underline{x}_{(t)}, \begin{cases} \underline{w}^T(t)\underline{x}_{(t)} \geq 0 \\ \underline{x}_{(t)} \in \omega_2 \end{cases}$$

$$\underline{w}(t+1) = \underline{w}(t) \quad \text{otherwise} \longrightarrow$$ طبقه‌بندی به درستی

نمونه‌ها را یک به یک به الگوریتم اعمال می‌کنیم؛ اگر پس از پایان یافتن نمونه‌ها، الگوریتم همگرا نشد، دوباره نمونه‌ها را اعمال خواهیم کرد.

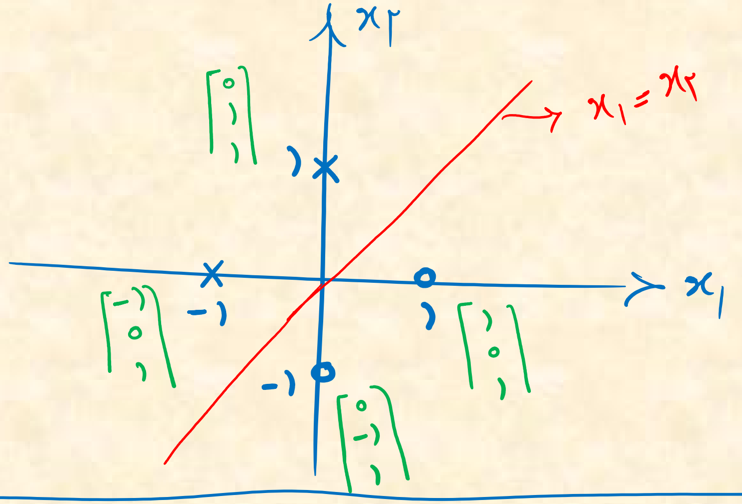
- It is a reward and punishment type of algorithm

مثال 3.2 :

$\times : \omega_1$
$\circ : \omega_2$

طبقه بند خطی با استفاده از الگوریتم پرسپترون به فرم پاداش و تنبیه طراحی کنید.

$$\underline{w}^T\underline{x} \underset{\omega_2}{\overset{\omega_1}{\gtrless}} 0$$



x_2
x_1
$x_1 = x_2$
1
-1
1
-1
$\begin{bmatrix}0\\1\\1\end{bmatrix}$
$\begin{bmatrix}-1\\0\\1\end{bmatrix}$
$\begin{bmatrix}1\\0\\1\end{bmatrix}$
$\begin{bmatrix}0\\-1\\1\end{bmatrix}$

حل:

$\underline{w}(0) = [0, 0, 0]$ تصادفی , $\rho = 1$

گام اول: $$\underline{w}^T(0)\begin{bmatrix}-1\\0\\1\end{bmatrix} = 0 \xrightarrow{\times} \underline{w}(1) = \underline{w}(0) + 1\cdot\begin{bmatrix}-1\\0\\1\end{bmatrix} = \begin{bmatrix}-1\\0\\1\end{bmatrix}$$

گام دوم: $$\underline{w}^T(1)\begin{bmatrix}0\\1\\1\end{bmatrix} = 1 > 0 \xrightarrow{\checkmark} \underline{w}(2) = \underline{w}(1)$$

گام سوم: $$\overset{\omega_2 \leftarrow}{\underline{w}^T(2)}\begin{bmatrix}0\\-1\\1\end{bmatrix} = 1 > 0 \xrightarrow{\times} \underline{w}(3) = \underline{w}(2) - 1\cdot\begin{bmatrix}0\\-1\\1\end{bmatrix} = \begin{bmatrix}-1\\1\\0\end{bmatrix}$$

گام چهارم: $$\overset{\omega_2 \leftarrow}{\underline{w}^T(3)}\begin{bmatrix}1\\0\\1\end{bmatrix} = -1 < 0 \xrightarrow{\checkmark} \underline{w}(4) = \underline{w}(3)$$

گام پنجم: $$\underline{w}^T(4)\begin{bmatrix}-1\\0\\1\end{bmatrix} = 1 > 0 \xrightarrow{\checkmark} \underline{w}(5) = \underline{w}(4)$$

$$\underline{w}^T\underline{x} = 0 \rightarrow [-1\ \ 1\ \ 0]\begin{bmatrix}x_1\\x_2\\1\end{bmatrix}$$

همگرا $-x_1 + x_2 = 0$

$x_1 = x_2$

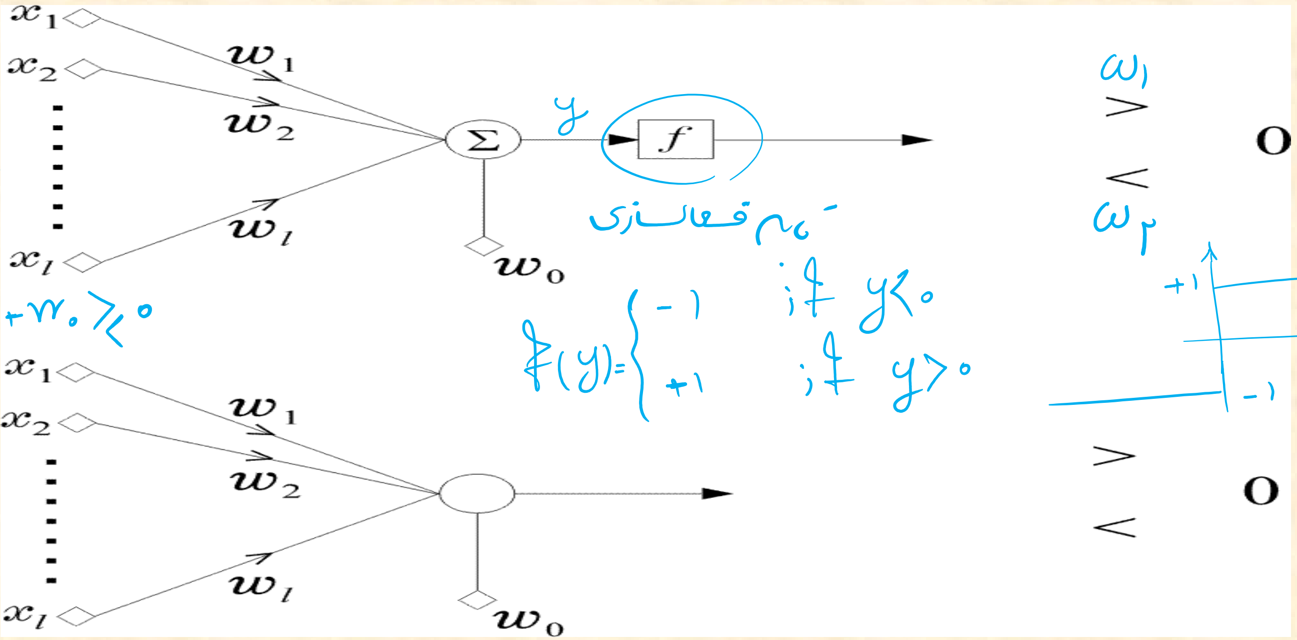
$$\ldots \xrightarrow{\text{گام هفتم}} \underline{w}(7) = \underline{w}(6) = \begin{bmatrix}-1\\1\\0\end{bmatrix}$$

❖ The perceptron

واحد پایه ای شبکه های عصبی (نورون)

$$\underline{w}^T \underline{x} + w_0 \overset{\omega_1}{\underset{\omega_2}{\gtrless}} 0$$

$$w_1 x_1 + w_2 x_2 + \dots + w_0 \gtrless 0$$



$$f(y) = \begin{cases} -1 & \text{if } y < 0 \\ +1 & \text{if } y > 0 \end{cases}$$

از روی مدل شبکهٔ عصبی انسان

w_i's synapses or synaptic weights

w_0 threshold

- ➢ The network is called perceptron or neuron
- ➢ It is a learning machine that learns from the training vectors via the perceptron algorithm

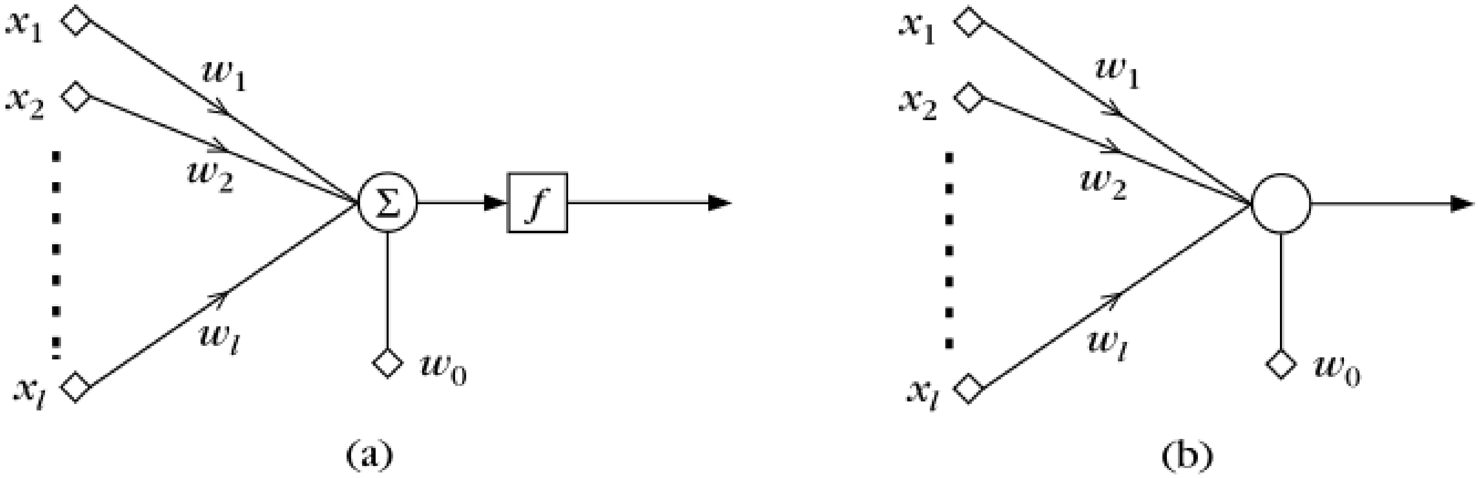


FIGURE 3.5

The basic perceptron model. (a) A linear combiner is followed by the activation function. (b) The combiner and the activation function are merged together.

مثال:

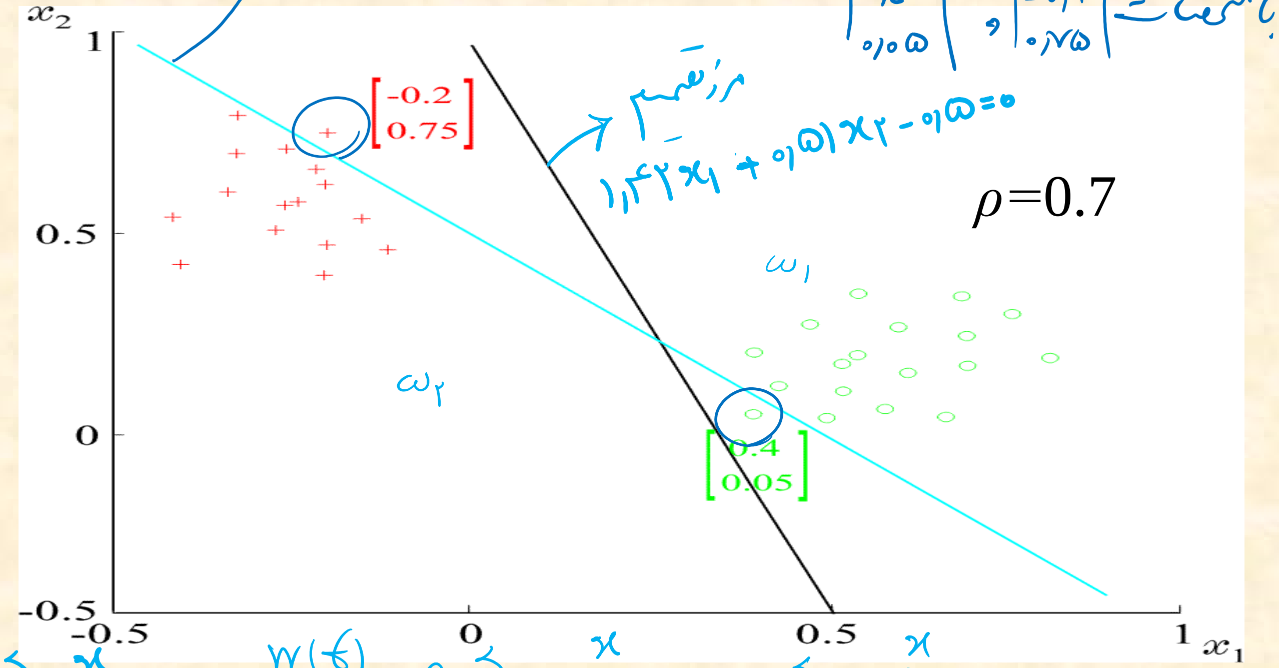
➢ Example: At some stage t the perceptron algorithm results in

$$w(t) \rightarrow \quad w_1 = 1, \; w_2 = 1, \; w_0 = -0.5$$

$$\text{مرز تصمیم} \rightarrow \quad x_1 + x_2 - 0.5 = 0$$

The corresponding hyperplane is

همه نمونه‌ها به درستی طبقه‌بندی شده‌اند
به جز دو نمونه با مختصات $\begin{bmatrix}-۰٫۲\\ ۰٫۷۵\end{bmatrix}$ و $\begin{bmatrix}۰٫۴\\ ۰٫۰۵\end{bmatrix}$



ρ=0.7

$$\underline{w}(t+1) = \underline{w}(t) - \rho_t \sum_{\underline{x}\in Y} \delta_x \underline{x}$$

$$\underline{w}(t+1) = \overset{w(t)}{\begin{bmatrix}1\\1\\-0.5\end{bmatrix}} \overset{-\rho_t\delta_x}{-0.7(-1)} \overset{\underline{x}}{\begin{bmatrix}0.4\\0.05\\1\end{bmatrix}} \overset{-\rho_t\delta_x}{-0.7(+1)} \overset{\underline{x}}{\begin{bmatrix}-0.2\\0.75\\1\end{bmatrix}} = \begin{bmatrix}1.42\\0.51\\-0.5\end{bmatrix}$$

w_0 w_0

الگوریتم Pocket : (برای حالت کلاسهایی که جدایی پذیر خطی نیستند)

شرط اصلی الگوریتم پرسپترون جدایی پذیر خطی بودن کلاسهاست. اگر این فرض برقرار نباشد که در عمل معمولاً این چنین است. الگوریتم پرسپترون همگرا نخواهد شد. فرم های دیگر الگوریتم که به راه حل بهینه همگرا شوند در این موارد استفاده می شود. که یکی از آنها الگوریتم Pocket است.

تعداد نمونه هایی که به درستی طبقه بندی شده اند.

دو گام الگوریتم Pocket :

$w_s = w(0)$

۱ـ مقدار دهی اولیه بردار وزن $w(0)$ به صورت تصادفی و ذخیره آن با عنوان w_s ، مقدار دهی شمارنده $h_s = 0$

۲ـ در تکرار t ام، با استفاده از قانون پرسپترون $w(t+1)$ را حساب می کنیم. با استفاده از این بردار وزن ، بردارهای آموزشی را تست می کنیم و تعداد بردارهایی که درست طبقه بندی شده اند را h می نامیم. اگر $h > h_s$ باشد، w_s را با $w(t+1)$ جایگزین می کنیم، h_s را با h جایگزین می کنیم.

$w_s = w(t+1)$ $\qquad$ $h_s = h$

حالتِ چند کلاسه :

تاکنون حالتِ دوکلاسه را بررسی کردیم. تعمیم به حالت $\underline{\underline{M}}$ کلاسه سرراست است :

تابع جداکننده خطی $\underline{w}_i \,;\ i=1,2,\dots,M$ برای هر یک از کلاسها تعریف شده است.

بردار ویژگی‌ها $\underline{x}$ با ابعاد $(l+1)$ بعدی به صورت زیر در کلاس ω_i طبقه‌بندی می‌شود اگر :

$$\underline{w}_i^T \underline{x} > \underline{w}_j^T \underline{x} \ ;\ \forall j\neq i$$

این شرط به ساختار kesler منجر می‌شود :

برای هر یک از بردارهای آموزشی متعلق به کلاس ω_i که $i=1,2,\dots,M$ باشد، $\boxed{\underline{M-1}}$ بردار آموزشی به صورت زیر تعریف می‌کنیم :

$$\underline{x}_{ij} = [\underline{0}^T, \underline{0}^T, \dots, \underline{x}^T, \dots, -\underline{x}^T, \dots, 0^T]^T_{(l+1)M\times 1}$$

همهٔ بلوک‌ها بردار صفرند غیر از بلوک i اُم که برابر با $\underline{x}^T$ و بلوک j اُم که برابر با $-\underline{x}^T$ است.

$$\underline{w}^T \underline{x}_{12} \overset{\omega_1}{\underset{\omega_2}{\gtrless}} 0$$

بردار وزن‌ها : $\underline{w} = [\underline{w}_1^T, \dots, \underline{w}_M^T]^T$

$$\underline{w}^T \underline{x}_{13} \overset{\omega_1}{\underset{\omega_3}{\gtrless}} 0$$

با این تعریف : if $\underline{x}\in\omega_i \rightarrow \underline{w}^T \underline{x}_{ij} > 0 \quad \forall j=1,2,\dots,M \ , j\neq i$

تعداد نمونه‌ها : $(M-1)N$

بنابراین مسأله به طراحی طبقه‌بند خطی در فضای $M(l+1)$ بعدی تبدیل شد.

مثال : مسأله ۳ کلاسه در فضای دو بعدی : بردارهای آموزشی :

$\omega_۱ :\ [۱,۱]^T ,\quad [۲,۲]^T ,\quad [۲,۱]^T$

$\omega_۲ :\ [۱,-۱]^T ,\quad [۱,-۲]^T ,\quad [۲,-۲]^T$

$\omega_۳ :\ [-۱,۱]^T ,\quad [-۱,۲]^T ,\quad [-۲,۱]^T$

مسأله جدایی پذیر خطی است.

ساختار kesler :

$$\underbrace{(l+1)}_{3}\underbrace{M}_{3}\times 1 = 9\times 1$$

$$\begin{cases} \underline{x} = [۱,۱,۱]^T \\ \omega_۱ \rightarrow i=۱ \end{cases}$$

$$\begin{cases} x_{۱۲} = [\overbrace{۱,۱,۱}^{\underline{x}^T}, \overbrace{-۱,-۱,-۱}^{-\underline{x}^T}, 0,0,0]^T_{9\times1} \\ x_{۱۳} = [\overbrace{۱,۱,۱}^{\underline{x}^T}, 0\ 0,0, \overbrace{-۱,-۱,-۱}^{-\underline{x}^T}]_{9\times1} \end{cases}$$

$$\begin{cases} \underline{x} = [۱,-۲,۱]^T \\ \omega_۲ \rightarrow i=۲ \end{cases}$$

$$x_{۲۱} = [\overbrace{-۱,۲,-۱}^{-\underline{x}^T}, \overbrace{۱,-۲,۱}^{\underline{x}^T}, 0,0,0]_{9\times1}$$

$$x_{۲۳} = [0,0,0, \overbrace{۱,-۲,۱}^{\underline{x}^T}, \overbrace{-۱,۲,-۱}^{-\underline{x}^T}]_{9\times1}$$

$$\begin{cases} \underline{x} = [-۲,۱,۱]^T \\ \omega_۳ \rightarrow i=۳ \end{cases}$$

$$x_{۳۱} = [۲,-۱,-۱, 0,0,0, -۲,۱,۱]_{9\times1}$$

$$x_{۳۲} = [0,0,0,۲,-۱,-۱,-۲,۱,۱]_{9\times1}$$

بردار وزن ها :

$$\begin{cases} \underline{w}_۱ = [w_{۱۱}, w_{۱۲}, w_{۱0}]^T \\ \underline{w}_۲ = [w_{۲۱}, w_{۲۲}, w_{۲0}]^T \\ \underline{w}_۳ = [w_{۳۱}, w_{۳۲}, w_{۳0}]^T \end{cases}$$

$$\rightarrow \underline{w} = [w_۱^T, w_۲^T, w_۳^T]^T$$

با ۱۸ بردار آموزشی و بردار $\underline{w}$ الگوریتم پرسپترون را اجرا کنیم :

$\underline{w}_۱ = [۵,۱۳ , ۳,۶۰ , ۱,۰۰]^T$

$\underline{w}_۲ = [-۰,۰۵ , -۳,۱۶ , -۰,۴۱]^T$

$\underline{w}_۳ = [-۳,۸۴ , ۱,۲۸ , ۰,۶۹]^T$

3.4 روشهای کمینه مربعات :

جذابیت طبقه‌بند های خطی در سادگی آنهاست بنابراین در بسیاری از موارد با وجود اینکه فرض جدایی پذیر خطی برقرار نیست. از این طبقه‌بندها استفاده می‌شود. در این مواقع طبقه‌بندهای خطی به راه حل زیر بهینه (sub optimal) منجر می‌شوند.

3.4.1 تخمین خطای میانگین مربعات :

مسأله دو کلاسه : خروجی $y = \pm 1$

حل با استفاده از خطاها

بردار ورودی $\underline{x}$ ——— طبقه‌بند $\underline{w}^T \underline{x}$

تابع هزینه $J(\underline{w}) = E[|y - \underline{w}^T \underline{x}|^2]$ MSE : Mean Square Error

خروجی مطلوب ← $y(\underline{x}) = y = \pm 1$ ، خروجی واقعی ← $\underline{w}^T \underline{x}$

MSE بین خروجی های واقعی و مطلوب

true desired

$$\text{minimize} \longrightarrow \hat{\underline{w}} = \arg\min_{\underline{w}} J(\underline{w})$$

$$3.28 \longrightarrow J(\underline{w}) = \underbrace{p(\omega_1)\int (1 - \underline{x}^T \underline{w})^2 p(\underline{x}|\omega_1)\, d\underline{x}}_{y=+1} + \underbrace{p(\omega_2)\int (1 + \underline{x}^T \underline{w})^2 p(x|\omega_2)\, d\underline{x}}_{y=-1}$$

$$\frac{\partial J(\underline{w})}{\partial \underline{w}} = 0 \longrightarrow 2E[\underline{x}(y - \underline{x}^T \underline{w})] = 0 \rightarrow \hat{\underline{w}} = R_x^{-1} E[\underline{x} y]$$

❖ Least Squares Methods

- If classes are linearly separable, the perceptron output results in ± 1

- If classes are NOT linearly separable, we shall compute the weights $w_1, w_2, \ldots, w_0$

  so that the difference between

  - The actual output of the classifier, $\underline{w}^T \underline{x}$, and

  - The desired outputs, e.g.

    $+1 \text{if } \underline{x} \in \omega_1$

    $-1 \text{if } \underline{x} \in \omega_2$

  to be SMALL

- SMALL, in the mean square error sense, means to choose $\underline{w}$ so that the cost function

  - $J(\underline{w}) \equiv E[(y - \underline{w}^T \underline{x})^2]$ is minimum
  - $\hat{\underline{w}} = \arg\min_{\underline{w}} J(\underline{w})$
  - y the corresponding desired responses

- Minimizing

$J(\underline{w})$ w.r.t. to $\underline{w}$ resultsin :

$$\frac{\partial J(\underline{w})}{\partial \underline{w}} = \frac{\partial}{\partial \underline{w}} E[(y - \underline{w}^T x)^2] = 0$$

$$= 2E[\underline{x}(y - \underline{x}^T \underline{w})] \Rightarrow$$

$$E[\underline{x}\underline{x}^T]\underline{w} = E[\underline{x}y] \Rightarrow$$

$$\boxed{\hat{\underline{w}} = R_x^{-1} E[\underline{x}y]}$$

where R_x is the autocorrelation matrix

$$R_x \equiv E[\underline{x}\underline{x}^T] = \begin{bmatrix} E[x_1 x_1] & E[x_1 x_2]\ldots & E[x_1 x_l] \\ \ldots\ldots\ldots\ldots. & \ldots\ldots\ldots\ldots\ldots & \ldots\ldots\ldots\ldots. \\ E[x_l x_1] & E[x_l x_2]\ldots & E[x_l x_l] \end{bmatrix}$$

and $E[\underline{x}y] = \begin{bmatrix} E[x_1 y] \\ \ldots \\ E[x_l y] \end{bmatrix}$ the crosscorrelation vector

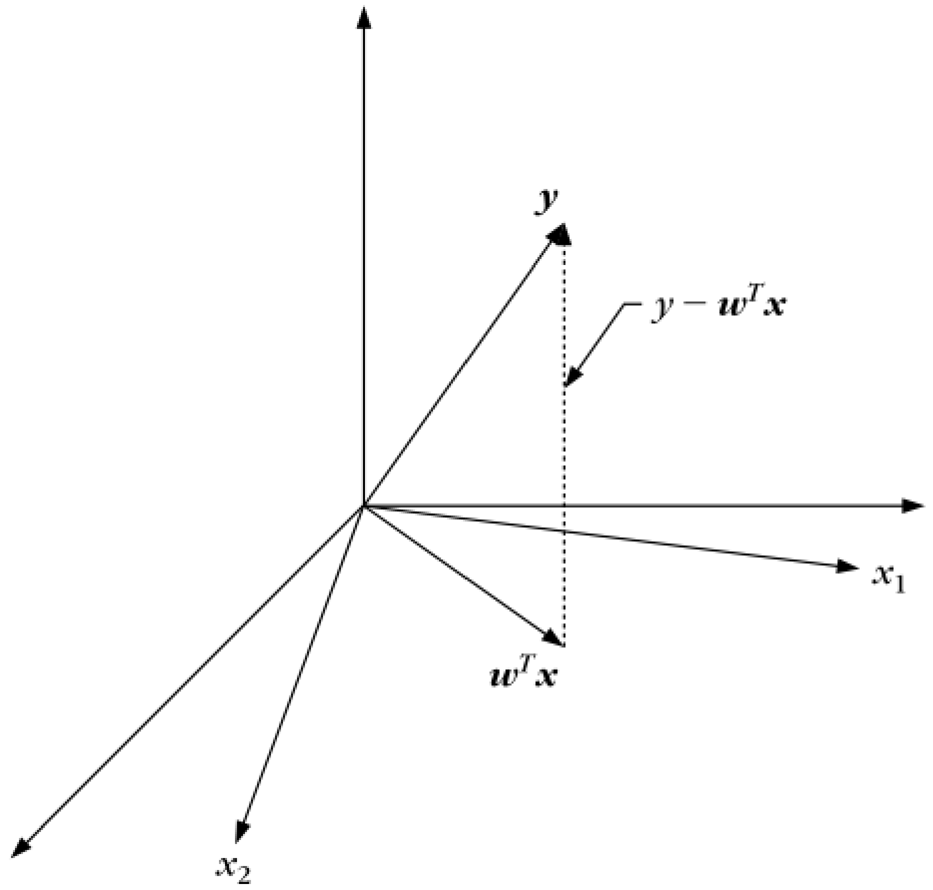

$\boldsymbol{y}$
$y - \boldsymbol{w}^T\boldsymbol{x}$
x_1
$\boldsymbol{w}^T\boldsymbol{x}$
x_2

FIGURE 3.6

Interpretation of the MSE estimate as an orthogonal projection on the input vector elements' subspace.

➢ Multi-class generalization

تعمیم چندکلاسه

- The goal is to compute M linear discriminant functions:

$$g_i(\underline{x}) = \underline{w}_i^T \underline{x}$$

؛ i = ۱, ۲, …, M

according to the MSE.

- Adopt as desired responses y_i:

خروجی های مطلوب

$$y_i = 1 \quad \text{if} \quad \underline{x} \in \omega_i$$

$$y_i = 0 \quad \text{otherwise}$$

- Let

$$\underline{y} = [y_1, y_2, \ldots, y_M]^T$$

- And the matrix

ماتریس وزن

$$W = [\underline{w}_1, \underline{w}_2, \ldots, \underline{w}_M]$$

- The goal is to compute W:

$$\hat{W} = \arg\min_{W} E\left[\left\|\underline{y} - W^T \underline{x}\right\|^2\right] = \arg\min_{W} E\left[\sum_{i=1}^{M}\left(y_i - \underline{w}_i^{\,T} \cdot \underline{x}\right)^2\right]$$

معادل M مسأله کمینه سازی MSE

- The above is equivalent to a number M of MSE minimization problems. That is:

  Design each $\underline{w}_i$ so that its desired output is 1 for $\underline{x} \in \omega_i$ and 0 for any other class.

➢ Remark: The MSE criterion belongs to a more general class of cost function with the following important property:

- The value of $g_i(\underline{x})$ is an estimate, in the MSE sense, of the a-posteriori probability $P(\omega_i \mid \underline{x})$, provided that the desired responses used during training are $y_i = 1, \underline{x} \in \omega_i$ and 0 otherwise.

X

- Mean square error regression: Let $\underline{y} \in \Re^M$, $\underline{x} \in \Re^\ell$ be jointly distributed random vectors with a joint pdf $p(\underline{x}, \underline{y})$
  - The goal: Given the value of $\underline{x}$ estimate the value of $\underline{y}$. In the pattern recognition framework, given $\underline{x}$ one wants to estimate the respective label $y = \pm 1$.
  - The MSE estimate $\underline{\hat{y}}$ of $\underline{y}$ given $\underline{x}$ is defined as:

$$\underline{\hat{y}} = \arg\min_{\tilde{y}} E\left[\|\underline{y} - \underline{\tilde{y}}\|^2\right]$$

  - It turns out that:

$$\underline{\hat{y}} = E\left[\underline{y} \,|\, \underline{x}\right] \equiv \int_{-\infty}^{+\infty} \underline{y}\, p(\underline{y} \,|\, \underline{x})\, d\underline{y}$$

The above is known as the regression of $\underline{y}$ given $\underline{x}$ and it is, in general, a non-linear function of $\underline{x}$. If $p(\underline{x}, \underline{y})$ is Gaussian the MSE regressor is linear.

3.4.3 تخمین مجموع مربعات خطا   Sum of Error Squares Estimation

معیاری نزدیک به MSE با عنوان مجموع مربعات خطا یا LS (Least Squares)

تابع هزینه
$$J(\underline{w}) = \sum_{i=1}^{N} (y_i - \underline{x}_i^T \underline{w})^2 \equiv \sum_{i=1}^{N} e_i^2$$

در MSE میانگین خطاها، در LS مجموع خطا

در MSE برای E{} به pdf نیاز است، ولی در LS این مشکل وجود ندارد.

$$\frac{\partial J(\underline{w})}{\partial \underline{w}} = 0 \text{ Minimizing} \longrightarrow \sum_{i=1}^{N} \underline{x}_i (y_i - \underline{x}_i^T \underline{\hat{w}}) = 0 \rightarrow \left(\sum_{i=1}^{N} \underline{x}_i \underline{x}_i^T\right) \hat{w} = \sum_{i=1}^{N} \underline{x}_i y_i$$

ماتریس
$$X = \begin{bmatrix} \underline{x}_1^T \\ \underline{x}_2^T \\ \vdots \\ \underline{x}_N^T \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1l} \\ x_{21} & x_{22} & \cdots & x_{2l} \\ \vdots & \ddots & & \vdots \\ x_{N1} & x_{N2} & & x_{Nl} \end{bmatrix}_{N\times l}, \quad \underline{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}_{N\times 1}$$

$$(X^T X)\hat{w} = X^T \underline{y} \longrightarrow \boxed{\hat{w} = \underbrace{(X^T X)^{-1} X^T}\, \underline{y}}$$

$X\hat{w} = \underline{y}$
دستگاه معادلات
حل

X^+ شبه وارون ماتریس X

اگر X مربعی باشد: $X^+ = X^{-1}$

$N = l$ معمولا اینطور نیست

❖ SMALL in the sum of error squares sense means

- $$J(\underline{w}) = \sum_{i=1}^{N} (y_i - \underline{w}^T \underline{x}_i)^2$$

  $(y_i, \underline{x}_i)$: training pairs that is, the input $\underline{x}_i$ and its corresponding class label $\underbrace{y_i \ (\pm 1)}$.

  برچسب‌های

  +۱ → $\omega_۱$

  -۱ → $\omega_۲$

- $$\frac{\partial J(\underline{w})}{\partial \underline{w}} = \frac{\partial}{\partial \underline{w}} \sum_{i=1}^{N} (y_i - \underline{w}^T \underline{x}_i)^2 = 0 \Rightarrow$$

$$\boxed{(\sum_{i=1}^{N} \underline{x}_i \underline{x}_i^T)\underline{w} = \sum_{i=1}^{N} \underline{x}_i y_i}$$

❖ Pseudoinverse Matrix

- Define

$$X = \begin{bmatrix} \underline{x}_1^T \\ \underline{x}_2^T \\ \cdots \\ \underline{x}_N^T \end{bmatrix} \quad \text{(an } Nxl \text{ matrix)}$$

$$\underline{y} = \begin{bmatrix} y_1 \\ \cdots \\ y_N \end{bmatrix} \quad \text{corresponding desired responses}$$

- $X^T = [\underline{x}_1, \underline{x}_2, \ldots, \underline{x}_N]$ (an lxN matrix)

- $X^T X = \sum_{i=1}^{N} \underline{x}_i \underline{x}_i^T$

- $X^T \underline{y} = \sum_{i=1}^{N} \underline{x}_i y_i$

Thus $(\sum_{i=1}^{N} \underline{x}_i^T \underline{x}_i)\hat{\underline{w}} = (\sum_{i=1}^{N} \underline{x}_i y_i)$

$$(X^T X)\hat{\underline{w}} = X^T \underline{y} \Rightarrow$$

$$\hat{\underline{w}} = (X^T X)^{-1} X^T \underline{y}$$

$$= X^{\neq} \underline{y}$$

$\boxed{X^{\neq} \equiv (X^T X)^{-1} X^T}$ Pseudoinverse of X

- Assume $N=l$ ⇒ X square and invertible. Then

$$(X^T X)^{-1} X^T = X^{-1} X^{-T} X^T = X^{-1} \Rightarrow$$

$$\boxed{X^{\neq} = X^{-1}}$$

- Assume $N>l$. Then, in general, there is no solution to satisfy all equations simultaneously:

$$X\underline{w}=\underline{y}: \quad \begin{array}{c} \underline{x}_1^T\underline{w}=y_1 \\ \underline{x}_2^T\underline{w}=y_2 \\ \ldots. \\ \underline{x}_N^T\underline{w}=y_N \end{array} \quad N \text{ equations} > l \text{ unknowns}$$

- The "solution" $\underline{w}=X^{\neq}\underline{y}$ corresponds to the minimum sum of squares solution

➢ Example:

$$\omega_1 : \begin{bmatrix}0.4\\0.5\end{bmatrix}, \begin{bmatrix}0.6\\0.5\end{bmatrix}, \begin{bmatrix}0.1\\0.4\end{bmatrix}, \begin{bmatrix}0.2\\0.7\end{bmatrix}, \begin{bmatrix}0.3\\0.3\end{bmatrix}$$

$$\omega_2 : \begin{bmatrix}0.4\\0.6\end{bmatrix}, \begin{bmatrix}0.6\\0.2\end{bmatrix}, \begin{bmatrix}0.7\\0.4\end{bmatrix}, \begin{bmatrix}0.8\\0.6\end{bmatrix}, \begin{bmatrix}0.7\\0.5\end{bmatrix}$$

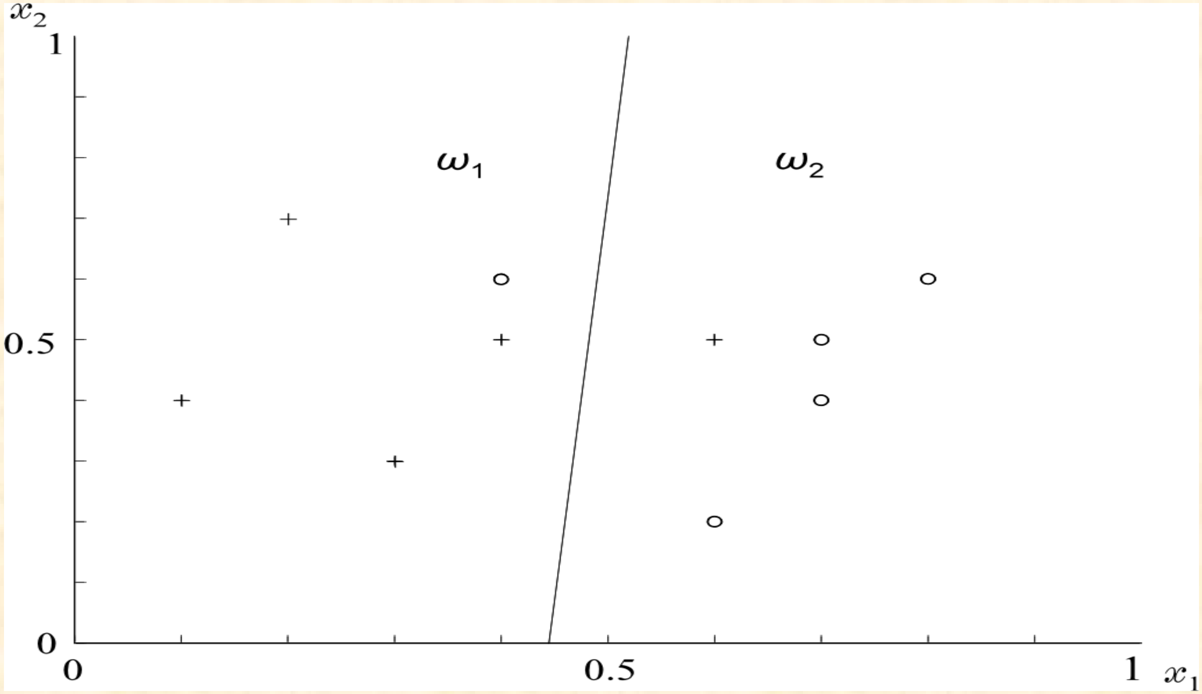
مثال:

مسأله دوکلاسه و دوبعدی

$w_1 x_1 + w_2 x_2 + w_0 = 0$

طبقه‌بند خطی بهینه با معیار مجموع مربعات خطا (LS) طراحی کنید.

حل: $\hat{w} = (X^T X)^{-1} X^T \underline{y}$



$$X = \begin{bmatrix} 0.4 & 0.5 & 1 \\ 0.6 & 0.5 & 1 \\ 0.1 & 0.4 & 1 \\ 0.2 & 0.7 & 1 \\ 0.3 & 0.3 & 1 \\ 0.4 & 0.6 & 1 \\ 0.6 & 0.2 & 1 \\ 0.7 & 0.4 & 1 \\ 0.8 & 0.6 & 1 \\ 0.7 & 0.5 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ -1 \\ -1 \\ -1 \\ -1 \\ -1 \end{bmatrix} = \underline{y}$$

- $$X^T X = \begin{bmatrix} 2.8 & 2.24 & 4.8 \\ 2.24 & 2.41 & 4.7 \\ 4.8 & 4.7 & 10 \end{bmatrix}, X^T \underline{y} = \begin{bmatrix} -1.6 \\ 0.1 \\ 0.0 \end{bmatrix}$$

$$\underline{w} = (X^T X)^{-1} X^T \underline{y} = \begin{bmatrix} -3.13 \\ 0.24 \\ 1.34 \end{bmatrix}$$

❖ The Bias – Variance Dilemma

A classifier $g(\underline{x})$ is a learning machine that tries to predict the class label y of $\underline{x}$. In practice, a finite data set D is used for its training. Let us write $g(\underline{x}; D)$. Observe that:

- For some training sets, $D = \{(y_i, \underline{x}_i), i = 1, 2, ..., N\}$, the training may result to good estimates, for some others the result may be worse.

- The average performance of the classifier can be tested against the MSE optimal value, in the mean squares sense, that is:

$$E_D\left[\left(g(\underline{x}; D) - E[y \mid \underline{x}]\right)^2\right]$$

where E_D is the mean over all possible data sets D.

- The above is written as:

$$E_D\left[\left(g(\underline{x};D)-E[y\,|\,\underline{x}]\right)^2\right]=$$

$$\left(E_D\left[g(\underline{x};D)\right]-E[y\,|\,\underline{x}]\right)^2+E_D\left[\left(g(\underline{x};D)-E_D\left[g(\underline{x};D)\right]\right)^2\right]$$

- In the above, the first term is the contribution of the bias and the second term is the contribution of the variance.

- For a finite D, there is a trade-off between the two terms. Increasing bias it reduces variance and vice verse. This is known as the bias-variance dilemma.

- Using a complex model results in low-bias but a high variance, as one changes from one training set to another. Using a simple model results in high bias but low variance.

❖ LOGISTIC DISCRIMINATION

- Let an M-class task, $\omega_1, \omega_2, \ldots, \omega_M$. In logistic discrimination, the logarithm of the likelihood ratios are modeled via linear functions, i.e.,

$$\ln\left(\frac{P(\omega_i \mid \underline{x})}{P(\omega_M \mid \underline{x})}\right) = w_{i,0} + \underline{w}_i^{\,T} \underline{x},\ \ i = 1,\ 2,\ \ldots,\ M\text{-}1$$

- Taking into account that

$$\sum_{i=1}^{M} P(\omega_i \mid \underline{x}) = 1$$

it can be easily shown that the above is equivalent with modeling posterior probabilities as:

$$P\left(\omega_M \mid \underline{x}\right) = \frac{1}{1 + \sum_{i=1}^{M-1} \exp\left(w_{i,0} + \underline{w}_i^{\,T}\, \underline{x}\right)}$$

$$P\left(\omega_i \mid \underline{x}\right) = \frac{\exp\left(w_{i,0} + \underline{w}_i^{\,T}\, \underline{x}\right)}{1 + \sum_{i=1}^{M-1} \exp\left(w_{i,0} + \underline{w}_i^{\,T}\, \underline{x}\right)}, \iota = 1,2,\ldots \mathrm{M} - 1$$

- For the two-class case it turns out that

$$P\left(\omega_2 \mid \underline{x}\right) = \frac{1}{1 + \exp\left(w_0 + \underline{w}^T\, \underline{x}\right)}$$

$$P\left(\omega_1 \mid \underline{x}\right) = \frac{\exp\left(w_0 + \underline{w}^T\, \underline{x}\right)}{1 + \exp\left(w_0 + \underline{w}^T\, \underline{x}\right)}$$

- The unknown parameters $\underline{w}_i, w_{i,0},\ i=1, 2, \ldots, M\text{-}1$ are usually estimated by maximum likelihood arguments.

- Logistic discrimination is a useful tool, since it allows linear modeling and at the same time ensures posterior probabilities to add to one.

3.7 طبقه بند ماشین بردار پشتیبان

Support Vector Machine (SVM)

3.7.1 : کلاس های جدایی پذیر خطی

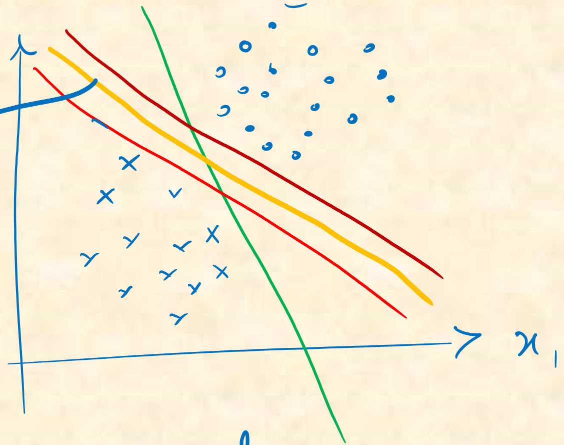
Seperable Classes

فرض کنید N بردار آموزشی $\underline{x}_i$, $i=1,2,\dots,N$ از کلاسهای ω_1 و ω_2 جدایی پذیر خطی

هدف : طراحی ابر صفحه ای که همه ی بردارهای آموزشی را به درستی طبقه بندی کند.

$$g(\underline{x}) = \underline{w}^T \underline{x} + w_0 = 0$$

همانطورکه قبلاً گفته شد این ابرصفحه جداکننده کلاسها یکتا نیست. برای مثال الگوریتم پرسپترون به یکی از این ابرصفحه‌ها همگرا شود.



طبقه بند بهتری است

زیرا فضای بیشتری را در اختیار هر کلاس قرار دهد.

generalization performance

کارایی تعمیم یافته

100% → 70% → Train
30% - Test
overfit

70%
30%
cross validation

❖ Support Vector Machines

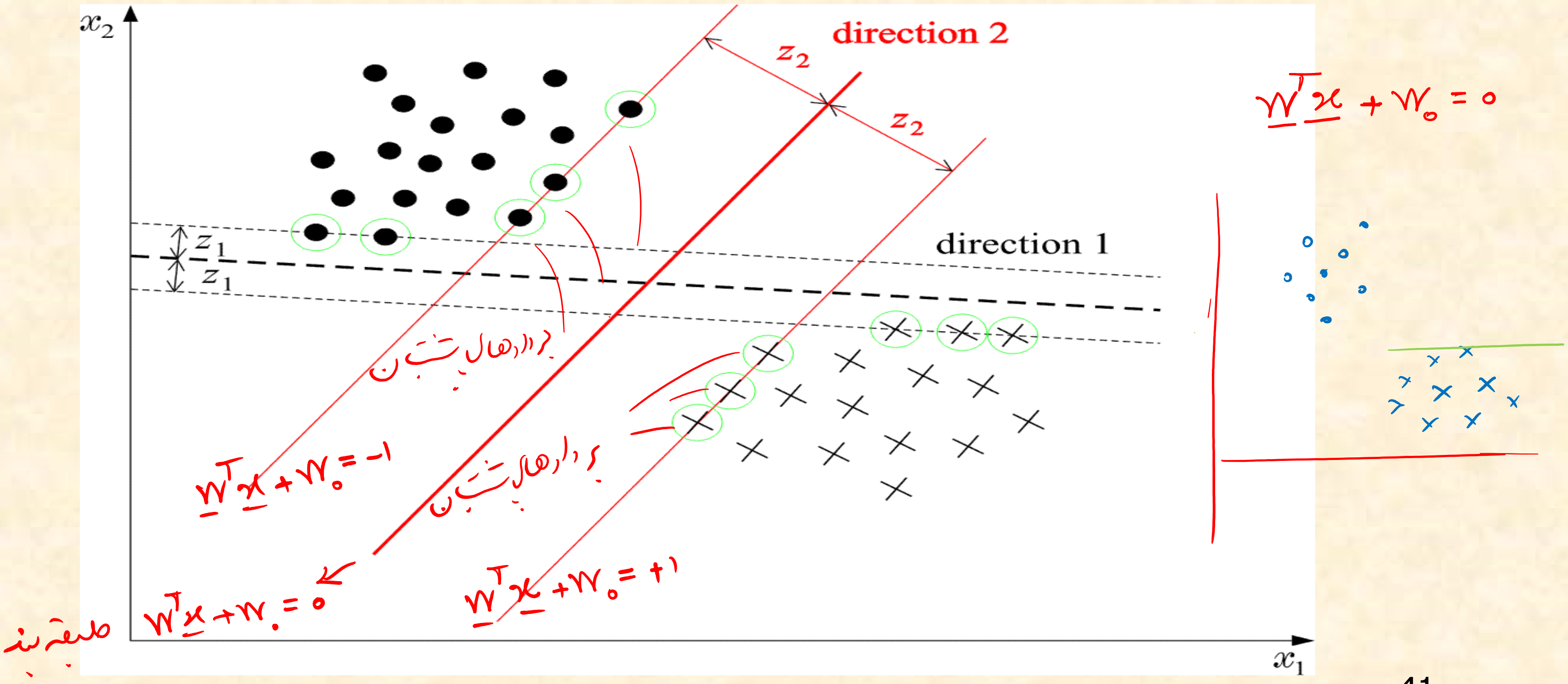
- ➢ The goal: Given two linearly separable classes, design the classifier

$$g(\underline{x}) = \underline{w}^T \underline{x} + w_0 = 0$$

Margin

SVM براساس بیشترین حاشیه از هم [illegible]
عمل می کند

that leaves the maximum margin from both classes



x_2
x_1
direction 2
z_2
z_2
direction 1
z_1
z_1
$\underline{w}^T\underline{x} + w_0 = 0$
بردارهای پشتیبان
بردارهای پشتیبان
$\underline{w}^T\underline{x} + w_0 = -1$
$\underline{w}^T\underline{x} + w_0 = 0$
$\underline{w}^T\underline{x} + w_0 = +1$
طبقه بند

- Margin: Each hyperplane is characterized by
  - Its direction in space, i.e., $\underline{w}$
  - Its position in space, i.e., w_0
  - For EACH direction, $\underline{w}$, choose the hyperplane that leaves the SAME distance from the nearest points from each class. The margin is twice this distance.

- The distance of a point $\hat{\underline{x}}$ from a hyperplane is given by

$$z_{\hat{x}} = \frac{g(\hat{\underline{x}})}{\|\underline{w}\|}$$

فاصله‌ی نقطه‌ی $\hat{x}$ از ابرصفحه‌ی $g(\underline{x})=0$:

$$g(\underline{x}) = \underline{w}^T\underline{x} + w_0 = 0$$

- Scale, $\underline{w}, \underline{w}_0$, so that at the nearest points from each class the discriminant function is ±1:

$$|g(x)| = 1 \ \{g(\underline{x}) = +1 \text{ for } \omega_1 \text{ and } g(\underline{x}) = -1 \text{ for } \omega_2\}$$

$\underline{w}$ و w_0 را طوری مقیاس دهی می‌کنیم که نزدیک‌ترین نقاط هر کلاس مقدار $g(\underline{x})=\pm1$ را داشته باشند.

- Thus the margin is given by

$$\frac{1}{\|\underline{w}\|} + \frac{1}{\|\underline{w}\|} = \frac{2}{\|\underline{w}\|}$$

- Also, the following is valid

$$\underline{w}^T\underline{x} + w_0 \geq 1 \ \forall \underline{x} \in \omega_1$$

$$\underline{w}^T\underline{x} + w_0 \leq -1 \ \forall \underline{x} \in \omega_2$$

- SVM (linear) classifier

بهینه سازی مقید

$$g(\underline{x}) = \underline{w}^T \underline{x} + w_0$$

maximize $\frac{۲}{\|\underline{w}\|}$

- Minimize

$$J(\underline{w}) = \frac{1}{2}\|\underline{w}\|^2$$

- Subject to

یک رابطه →

$$y_i(\underline{w}^T \underline{x}_i + w_0) \geq 1,\ i = 1,2,\ldots,N$$

$$y_i = 1, \text{for } \underline{x}_i \in \omega_1,$$

$$y_i = -1, \text{for } \underline{x}_i \in \omega_2$$

- The above is justified since by minimizing $\|\underline{w}\|$

  the margin $\frac{2}{\|w\|}$ is maximised

KKT

- The above is a quadratic optimization task, subject to a set of linear inequality constraints. The Karush-Kuhh-Tucker conditions state that the minimizer satisfies: پیوست (ج)

تابع لاگرانژ ضرایب لاگرانژ

- (1) $\frac{\partial}{\partial \underline{w}} \mathrm{L}(\underline{w}, w_0, \underline{\lambda}) = \underline{0}$
- (2) $\frac{\partial}{\partial w_0} L(\underline{w}, w_0, \underline{\lambda}) = 0$
- (3) $\lambda_i \geq 0, i = 1,2,\ldots, N$
- (4) $\lambda_i \left[y_i (\underline{w}^T \underline{x}_i + w_0) - 1 \right] = 0, i = 1,2,\ldots, N$
- Where $L(\bullet,\bullet,\bullet)$ is the Lagrangian

$$L(\underline{w},\ w_0,\ \underline{\lambda}) \equiv \frac{1}{2} \underline{w}^T \underline{w} - \sum_{i=1}^{N} \lambda_i [y_i (\underline{w}^T \underline{x}_i + w_0) - 1]$$

- The solution: from the above, it turns out that

- $$\underline{w} = \sum_{i=1}^{N} \lambda_i y_i \underline{x}_i$$

- $$\sum_{i=1}^{N} \lambda_i y_i = 0$$

- Remarks:
  - The Lagrange multipliers can be either zero or positive. Thus,
    - $\underline{w} = \sum_{i=1}^{N_s} \lambda_i y_i \underline{x}_i$

      where $N_s \leq N_0$, corresponding to positive Lagrange multipliers

    - From constraint (4) above, i.e.,

      $$\lambda_i [y_i (\underline{w}^T \underline{x}_i + w_0) - 1] = 0, \quad i = 1,2,\ldots, N$$

      the vectors contributing to $\underline{w}$ satisfy

      $$\underline{w}^T \underline{x}_i + w_0 = \pm 1$$

- These vectors are known as SUPPORT VECTORS and are the closest vectors, from each class, to the classifier.

- Once $\underline{w}$ is computed, w_0 is determined from conditions (4).

- The optimal hyperplane classifier of a support vector machine is UNIQUE.

- Although the solution is unique, the resulting Lagrange multipliers are not unique.

- Dual Problem Formulation
  - The SVM formulation is a convex programming problem, with
    - Convex cost function
    - Convex region of feasible solutions
  - Thus, its solution can be achieved by its dual problem, i.e.,
    - maximize $\underset{\underline{\lambda}}{} \quad L(\underline{w}, w_0, \underline{\lambda})$
    - subject to

$$\underline{w} = \sum_{i=1}^{N} \lambda_i y_i \underline{x}_i$$

$$\sum_{i=1}^{N} \lambda_i y_i = 0$$

$$\underline{\lambda} \geq \underline{0}$$

- Combine the above to obtain
  - $\underset{\underline{\lambda}}{\text{maximize}}\ (\sum_{i=1}^{N}\lambda_i - \frac{1}{2}\sum_{ij}\lambda_i\lambda_j y_i y_j \underline{x}_i^T \underline{x}_j)$
  - subject to

$$\sum_{i=1}^{N}\lambda_i y_i = 0$$

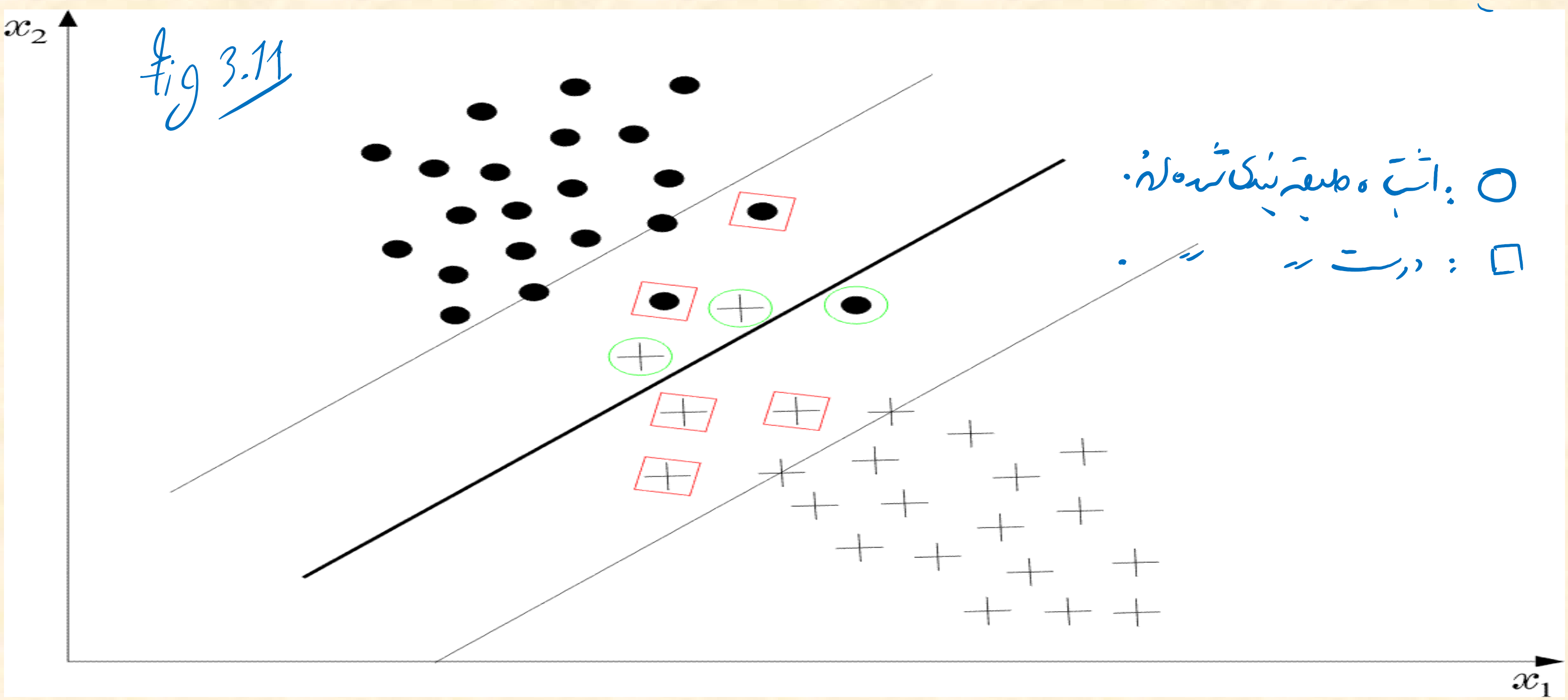
$$\underline{\lambda} \geq \underline{0}$$

- ➢ Remarks:
  - Support vectors enter via inner products

- ➢ Non-Separable classes

حالت دوم : کلاس های جدایی ناپذیر خطی :

در این حالت از آنجایی که هیچ ابرصفحه‌ای وجود ندارد که کاملاً دو کلاس ها را از یکدیگر جدا کنند ، همواره نمونه هایی به اشتباه طبقه بندی می شوند.



داخل حاشیه ها، تعدادی نمونه قرار گرفته اند:

سه دسته بردار وجود دارد :

۱- بردارهایی که خارج از باند قرار گرفته اند و به درستی طبقه بندی شده اند.

۲- بردارهایی که داخل باند قرار گرفته اند و درست طبقه بندی شده اند (□) و معادله زیر را برآورده می کنند:

$$0 \leq y_i(\underline{w}^T\underline{x}+w_0) < 1$$

۳- بردارهایی که داخل باند قرار گرفته اند و به اشتباه طبقه بندی شده اند (○) و معادله زیر را برآورده می کنند:

$$y_i(\underline{w}^T\underline{x}+w_0) < 0$$

هر سه دسته را می توان با رابطه زیر بیان کرد:

$$y_i(\underline{w}^T\underline{x}+w_0) \geq 1-\xi_i$$

برای دسته اول : $\xi_i = 0$

ـ ـ دوم : $0 < \xi_i \leq 1$

ـ ـ سوم : $\xi_i > 1$

به ξ_i ها متغیرهای slack گفته می شود.

In this case, there is no hyperplane such that

$$\underline{w}^T \underline{x} + w_0 (><) 1, \ \forall \underline{x}$$

- Recall that the margin is defined as twice the distance between the following two hyperplanes

$$\underline{w}^T \underline{x} + w_0 = 1$$

$$\text{and}$$

$$\underline{w}^T \underline{x} + w_0 = -1$$

- The training vectors belong to one of three possible categories

1) Vectors outside the band which are correctly classified, i.e.,

$$y_i(\underline{w}^T \underline{x} + w_0) > 1$$

2) Vectors inside the band, and correctly classified, i.e.,

$$0 \leq y_i(\underline{w}^T \underline{x} + w_0) < 1$$

3) Vectors misclassified, i.e.,

$$y_i(\underline{w}^T \underline{x} + w_0) < 0$$

- All three cases above can be represented as

$$y_i(\underline{w}^T \underline{x} + w_0) \geq 1 - \xi_i$$

1) $\rightarrow \xi_i = 0$
2) $\rightarrow 0 < \xi_i \leq 1$
3) $\rightarrow 1 < \xi_i$

ξ_i are known as slack variables

- The goal of the optimization is now two-fold

هدف در این حالت:

  - Maximize margin

۱- بیشینه کردن حاشیه

  - Minimize the number of patterns with $\xi_i > 0$,
    One way to achieve this goal is via the cost

$$J(\underline{w}, w_0, \underline{\xi}) = \frac{1}{2}\|\underline{w}\|^2 + \underline{C}\sum_{i=1}^{N} I(\xi_i)$$

۲- کمینه کردن تعداد نمونه‌هایی که $\xi_i > 0$ دارند. (داخل باند قرار دارند)

where C is a constant and

ترم دومش تعداد نمونه‌های داخل باند $\xi_i > 0$

$$I(\xi_i) = \begin{cases} 1 & \xi_i > 0 \\ 0 & \xi_i = 0 \end{cases}$$

Regularization Parameter

  - $I(.)$ is not differentiable. In practice, we use an approximation

پارامتر C (پارامتر تنظیم) ثابتی است که تاثیر هر یک از دو عبارت را در J تعیین می‌کند:

بهینه سازی این تابع هزینه به دلیل ناپیوستگی و در نتیجه مشتق‌ناپذیری $I(\xi_i)$ دشوار است. بنابراین در عمل متداول است که به جای آن از توابع تقریبی استفاده شود:

  - $$J(\underline{w}, w_0, \underline{\xi}) = \frac{1}{2}\|\underline{w}\|^2 + C\sum_{i=1}^{N} \xi_i$$

  - Following a similar procedure as before we obtain

➢ KKT conditions

حل مسئله :

$$(1)\quad \underline{w} = \sum_{i=1}^{N} \lambda_i y_i \underline{x}_i$$

$$(2)\quad \sum_{i=1}^{N} \lambda_i y_i = 0$$

$$(3)\quad C - \mu_i - \lambda_i = 0, i = 1,2,\ldots, N$$

$$(4)\quad \lambda_i [y_i (\underline{w}^T \underline{x}_i + w_0) - 1 + \xi_i] = 0, \quad i = 1,2,\ldots, N$$

$$(5)\quad \mu_i \xi_i = 0, \quad i = 1,2,\ldots, N$$

$$(6)\quad \mu_i, \lambda_i \geq 0, \quad i = 1,2,\ldots, N$$

- The associated dual problem

Maximize $\underline{\lambda}(\sum_{i=1}^{N}\lambda_i - \frac{1}{2}\sum_{i,j}\lambda_i\lambda_j y_i y_j \underline{x}_i^T \underline{x}_j)$

subject to

$$0 \le \lambda_i \le C,\ i = 1,2,\ldots,N$$

$$\sum_{i=1}^{N}\lambda_i y_i = 0$$

- Remarks: The only difference with the separable class case is the existence of C in the constraints

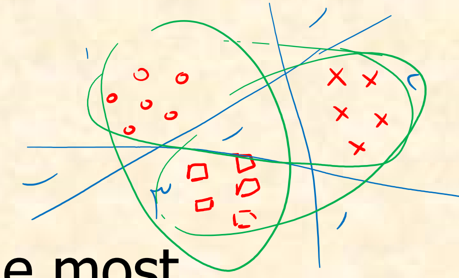
- Training the SVM

  A major problem is the high computational cost. To this end, decomposition techniques are used. The rationale behind them consists of the following:

  - Start with an arbitrary data subset (working set) that can fit in the memory. Perform optimization, via a general purpose optimizer.
  - Resulting support vectors remain in the working set, while others are replaced by new ones (outside the set) that violate severely the KKT conditions.
  - Repeat the procedure.
  - The above procedure guarantees that the cost function decreases.
  - Platt's SMO algorithm chooses a working set of two samples, thus analytic optimization solution can be obtained.

- Multi-class generalization

- مسئله M کلاسه:



One against All:

- یکی در برابر همه:

Although theoretical generalizations exist, the most popular in practice is to look at the problem as M two-class problems (one against all).

در این حالت برای هر یک از کلاس ها به دنبال طراحی تابع جداکننده بهینه $g_i(\underline{x})$ هستیم که:

$g_i(\underline{x}) > g_j(\underline{x})$ باشد ; $i=1,\dots,M$

$\forall j \neq i$ if $\underline{x} \in \omega_i$

در SVM: $g(\underline{x}) = 0$ را طوری طراحی کنیم که کلاس ω_i را از همه کلاس های دیگر

assign $\underline{x}$ in ω_i if $i = \arg\max_k [g_k(\underline{x})]$ جداکننده

دو مشکل: ۱- نواحی نامشخص: نواحی که چند $g_i(\underline{x})>$ دارند.

۲- آموزش نامتقارن assymetric training

تعداد نمونه‌های منفی که ممکن است بیشتر شود به دلیل تعداد کلاس های زیاد

- یکی در برابر دیگری:

برای هر دو کلاس یک طبقه‌بند طراحی می‌کنیم

One against One:

Binary Classifiers

M کلاس

مشکل: تعداد زیاد طبقه‌بندها

تعداد طبقه‌بندها: $\frac{M(M-1)}{2}$

$\binom{M}{2}$

مثال:

3.5

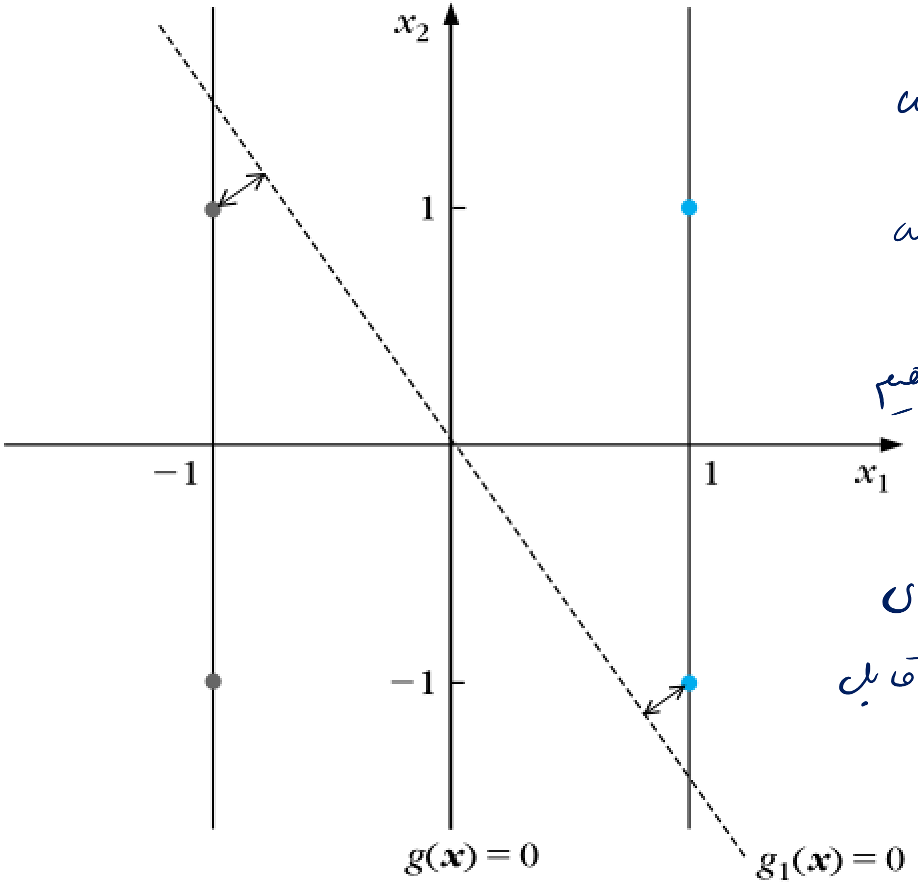
دو کلاسه

$\omega_1 : [1, 1]^T, [1, -1]^T$

$\omega_2 : [-1, 1]^T, [-1, -1]^T$

با روش SVM می خواهیم نشان دهیم که خط $x_1 = 0$ ابرصفحه جداکننده است و این نتیجه به ازای مجموعه های مختلف از ضرایب لاگرانژ قابل دستیابی است.

$$g(\underline{x}) = w_1 x_1 + w_2 x_2 + w_0 = 0$$



$w_1 = 1$

$w_2 = w_0 = 0$

FIGURE 3.12

In this example all four points are support vectors. The margin associated with $g_1(x) = 0$ is smaller compared to the margin defined by the optimal $g(x) = 0$.

شرایط نامساوی :

۴ نقطه

$$w_1 + w_2 + w_0 - 1 \geq 0$$

$$w_1 - w_2 + w_0 - 1 \geq 0$$

$$w_1 - w_2 - w_0 - 1 \geq 0$$

$$w_1 + w_2 - w_0 - 1 \geq 0$$

$$\mathcal{L}(w_1, w_2, w_0, \lambda) = \frac{w_1^2 + w_2^2}{2} - \lambda_1 (w_1 + w_2 + w_0 - 1) - \lambda_2 (w_1 - w_2 + w_0 - 1) - \lambda_3 (w_1 - w_2 - w_0 - 1) - \lambda_4 (w_1 + w_2 - w_0 - 1)$$

KKT :

$$\frac{\partial \mathcal{L}}{\partial w_1} = 0 \rightarrow w_1 = \lambda_1 + \lambda_2 + \lambda_3 + \lambda_4$$

$$\frac{\partial \mathcal{L}}{\partial w_2} = 0 \rightarrow w_2 = \lambda_1 - \lambda_2 - \lambda_3 + \lambda_4$$

$$\frac{\partial \mathcal{L}}{\partial w_0} = 0 \rightarrow \lambda_1 + \lambda_2 - \lambda_3 - \lambda_4 = 0$$

$$\lambda_1 (w_1 + w_2 + w_0 - 1) = 0$$

$$\lambda_2 (w_1 - w_2 + w_0 - 1) = 0$$

$$\lambda_3 (w_1 - w_2 - w_0 - 1) = 0$$

$$\lambda_4 (w_1 + w_2 - w_0 - 1) = 0$$

$$\lambda_1, \lambda_2, \lambda_3, \lambda_4 \geq 0$$

$$w_1 = 1, \quad w_2 = w_0 = 0 \rightarrow \begin{cases} \lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 1 \\ \lambda_1 - \lambda_2 - \lambda_3 + \lambda_4 = 0 \\ \lambda_1 + \lambda_2 - \lambda_3 - \lambda_4 = 0 \end{cases}$$

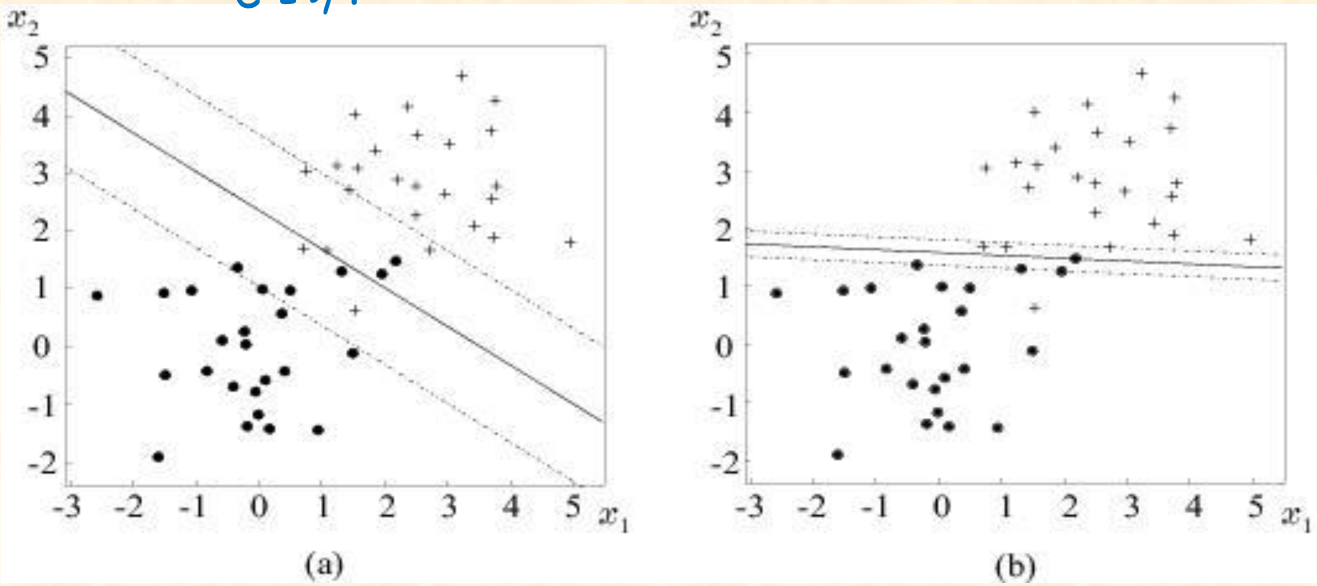
۳ معادله چهار مجهول

بی نهایت جواب

مثال: تأثیر پارامتر تنظیم C درحالت کلاس‌های جدایی ناپذیر خطی

- Example:

C = ۰/۲ C = ۱۰۰۰



(a) (b)

- Observe the effect of different values of C in the case of non-separable classes.

C کمتر ← margin بیشتر

C بیشتر ← تعداد نمونه‌های داخل باند کمتر