شبکه‌های عصبی مصنوعی

جلسه سوم:

پرسپترون (Perceptron)

محمد فرخی (farrokhi AT iust.ac.ir)

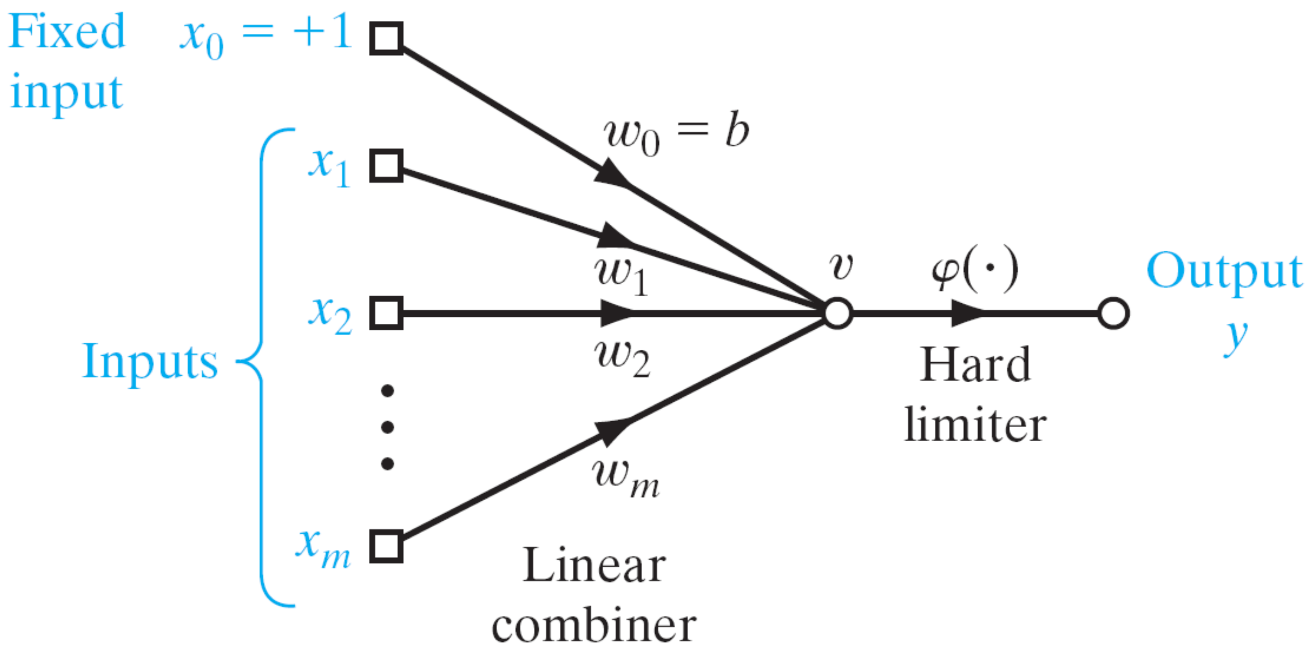
پرسپترون (Perceptron)

پرسپترون (Perceptron)

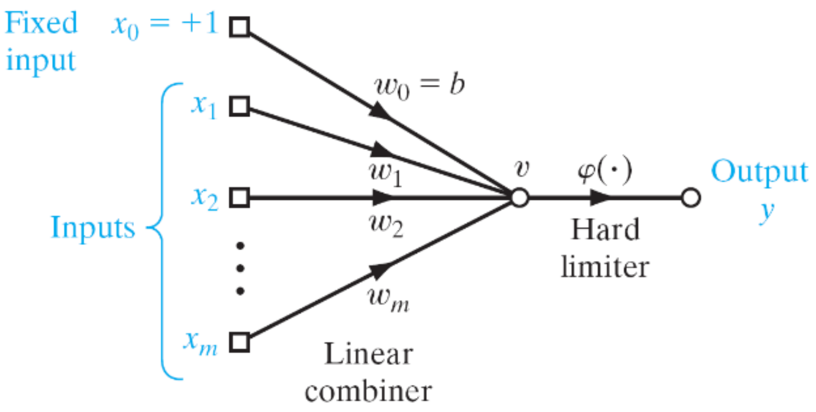
پرسپترون در سال ۱۹۵۸ توسط فرانک روزنبلات ابداع شد.

پرسپترون (Perceptron)

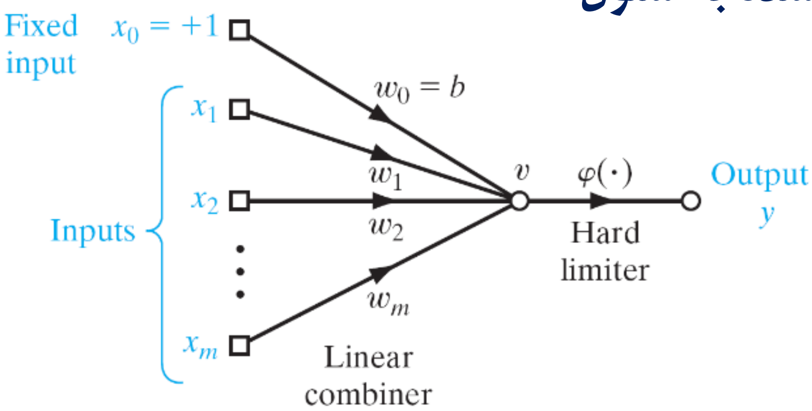
پرسپترون در سال ۱۹۵۸ توسط فرانک روزنبلات ابداع شد.



پرسپترون (Perceptron)



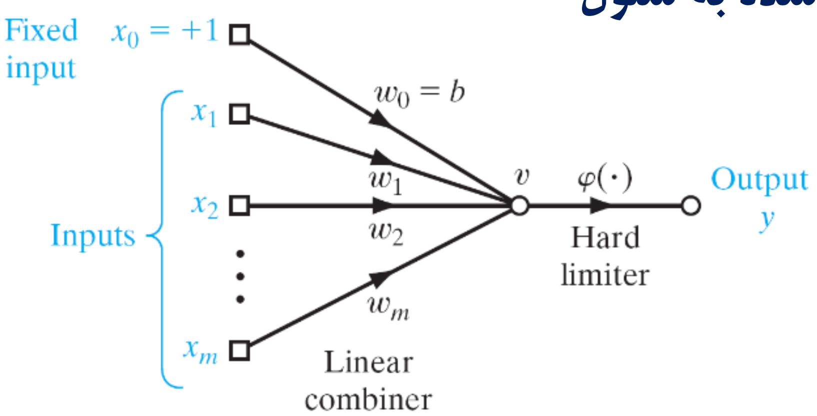
پرسپترون (Perceptron)



گره جمع‌کننده، ترکیب خطی از ورودی‌های اعمال‌شده به سلول را محاسبه می‌کند:

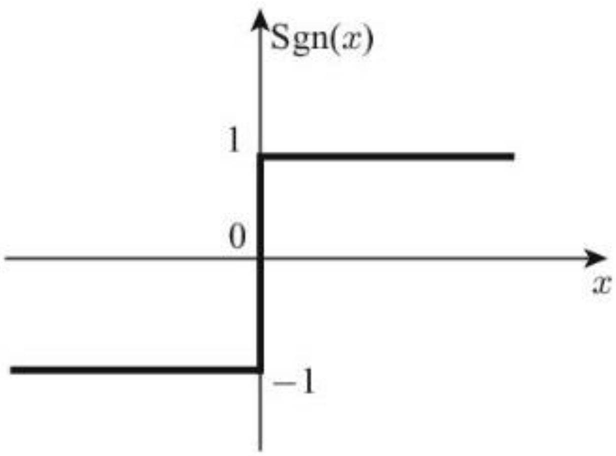
$$v = \sum_{i=1}^{m} w_i x_i + w_0$$

پرسپترون (Perceptron)



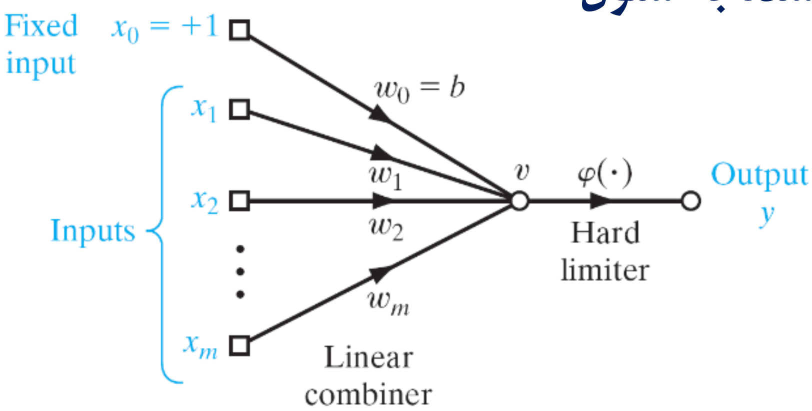
گره جمع‌کننده، ترکیب خطی از ورودی‌های اعمال‌شده به سلول را محاسبه می‌کند:

$$v = \sum_{i=1}^{m} w_i x_i + w_0$$



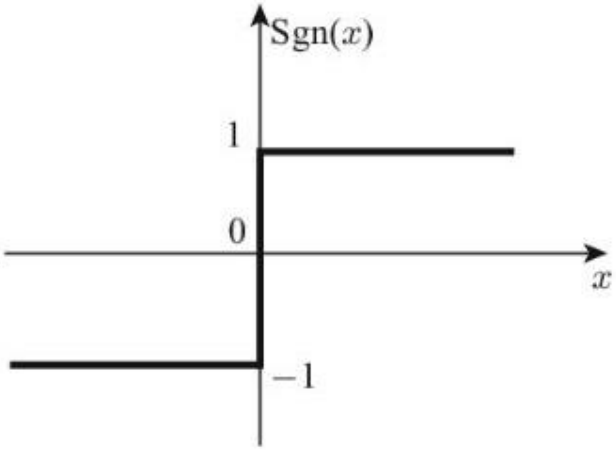
عبور این سیگنال از تابع غیرخطی $\phi(\cdot)$

پرسپترون (Perceptron)



گره جمع‌کننده، ترکیب خطی از ورودی‌های اعمال‌شده به سلول را محاسبه می‌کند:

$$v = \sum_{i=1}^{m} w_i x_i + w_0$$



عبور این سیگنال از تابع غیرخطی $\phi(\cdot)$

خروجی پرسپترون

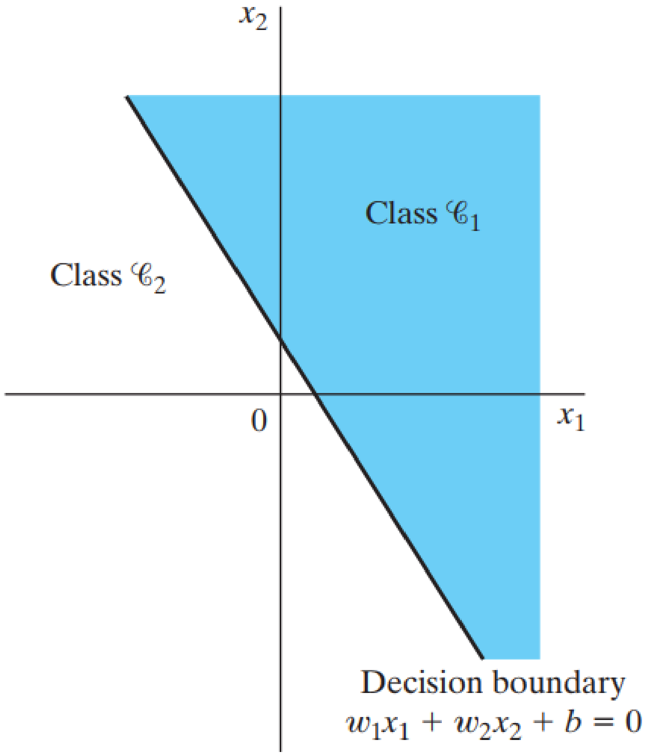
$$y = \operatorname{sgn}(v) = \begin{cases} +1 & v > 0 \\ -1 & v \le 0 \end{cases}$$

پرسپترون (Perceptron)

– وظیفه پرسپترون جداکردن الگوهای ورودی به دوکلاس C_1 و C_2

پرسپترون (Perceptron)

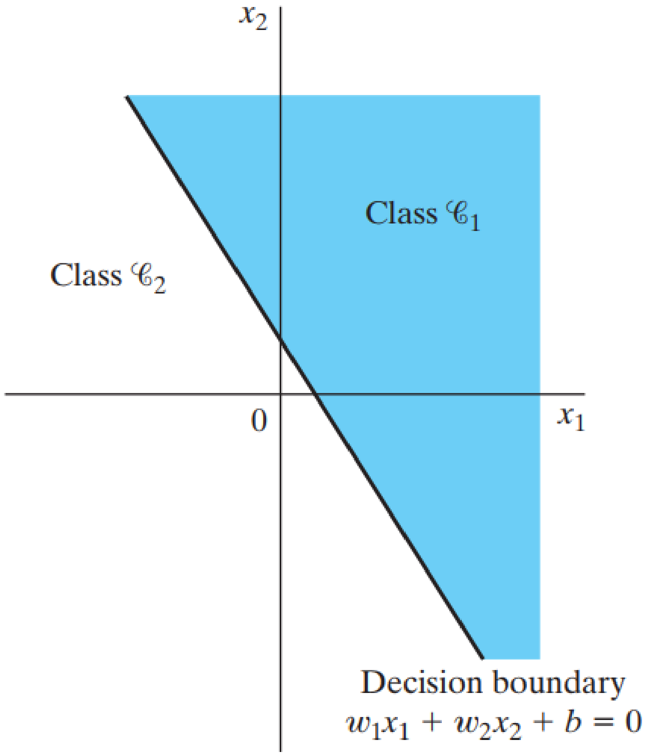
– وظیفه پرسپترون جداکردن الگوهای ورودی به دوکلاس C_1 و C_2



حالت خاص: الگوهای ورودی با دو درایه x_1 و x_2

پرسپترون (Perceptron)

– وظیفه پرسپترون جداکردن الگوهای ورودی به دوکلاس C_1 و C_2



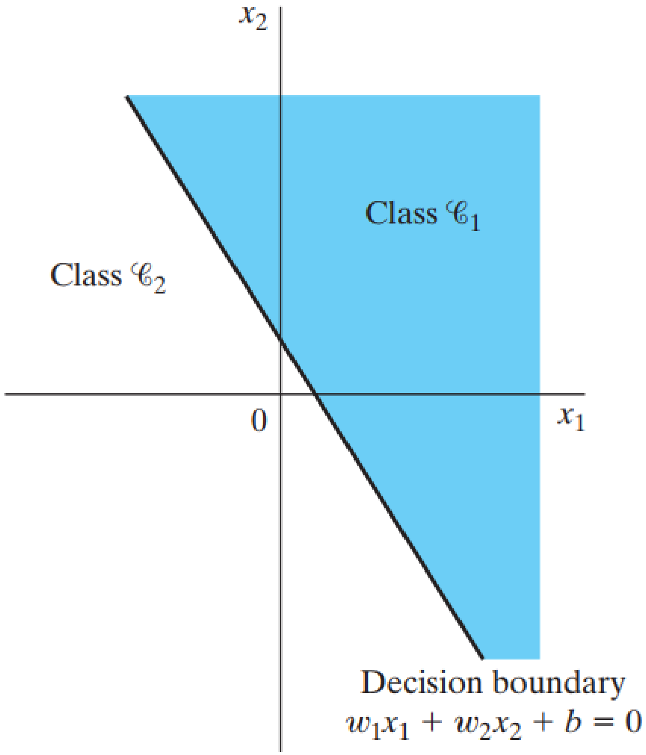
– جداکردن این دو کلاس توسط سطحی با معادله

$$\sum_{i=1}^{m} w_i x_i + w_0 = 0$$

حالت خاص: الگوهای ورودی با دو درایه x_1 و x_2

پرسپترون (Perceptron)

– وظیفه پرسپترون جداکردن الگوهای ورودی به دوکلاس C_1 و C_2



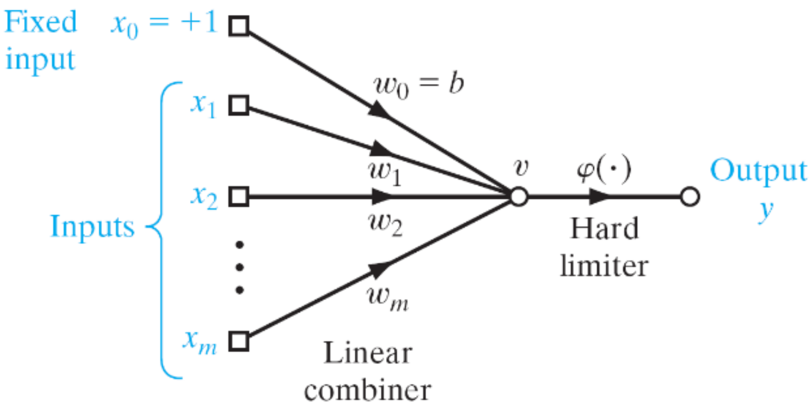
– جداکردن این دو کلاس توسط سطحی با معادله

$$\sum_{i=1}^{m} w_i x_i + w_0 = 0$$

که به آن مرز تصمیم‌گیری (Decision boundary) می‌گویند.

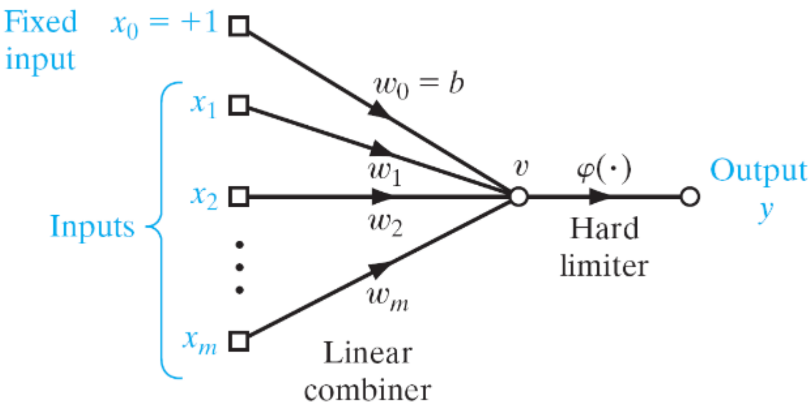
حالت خاص: الگوهای ورودی با دو درایه x_1 و x_2

پرسپترون (Perceptron)



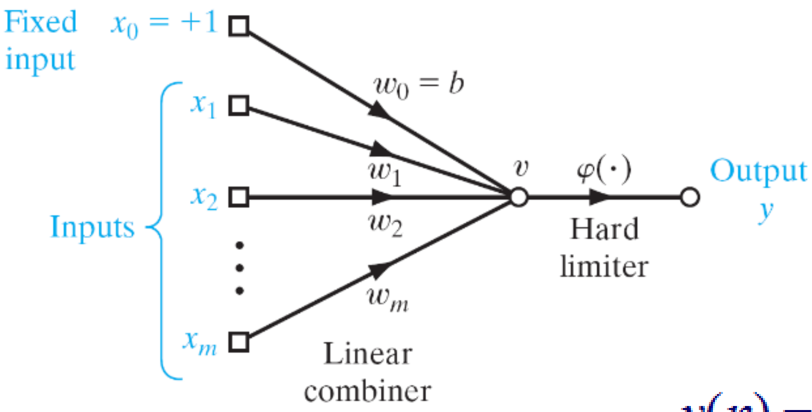
پرسپترون (Perceptron)

الگوریتم پرسپترون:



پرسپترون (Perceptron)

الگوریتم پرسپترون:



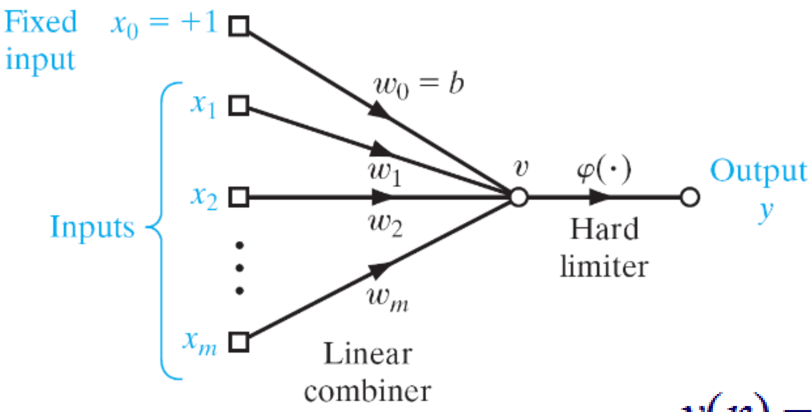
ابتدا معادلات را به‌صورت برداری زیر می‌نویسیم:

$$v(n) = \mathbf{w}^T(n)\ \mathbf{x}(n)$$

$$\mathbf{w}^T(n)\ \mathbf{x}(n)\ = 0$$

پرسپترون (Perceptron)

الگوریتم پرسپترون:



ابتدا معادلات را به‌صورت برداری زیر می‌نویسیم:

$$v(n) = \mathbf{w}^T(n)\ \mathbf{x}(n)$$

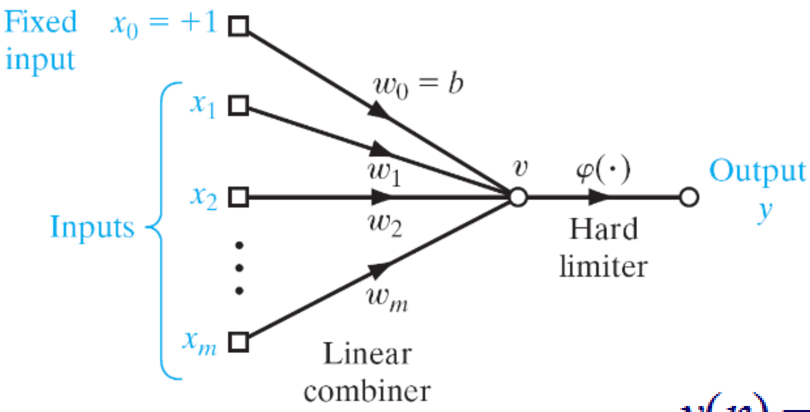
$$\mathbf{w}^T(n)\ \mathbf{x}(n) = 0$$

که در آن‌ها

$$\mathbf{w}(n) = \left[w_0(n) \quad w_1(n) \quad \cdots \quad w_m(n)\right]^T$$

$$\mathbf{x}(n) = \left[+1 \quad x_1(n) \quad \cdots \quad x_m(n)\right]^T$$

پرسپترون (Perceptron)



الگوریتم پرسپترون:

ابتدا معادلات را به‌صورت برداری زیر می‌نویسیم:

$$v(n) = \mathbf{w}^T(n)\ \mathbf{x}(n)$$

$$\mathbf{w}^T(n)\ \mathbf{x}(n) = 0$$

که در آن‌ها

$$\mathbf{w}(n) = [w_0(n) \quad w_1(n) \quad \cdots \quad w_m(n)]^T$$

$$\mathbf{x}(n) = [+1 \quad x_1(n) \quad \cdots \quad x_m(n)]^T$$

قرارداد برای سطح تصمیم‌گیری:

$$1)\ \mathbf{w}^T\mathbf{x} > 0 \Rightarrow \mathbf{x} \in C_1$$

$$2)\ \mathbf{w}^T\mathbf{x} \le 0 \Rightarrow \mathbf{x} \in C_2$$

پرسپترون (Perceptron)

الگوریتم پرسپترون:

$$1)\ \mathbf{w}^T\mathbf{x} > 0 \Rightarrow \mathbf{x} \in C_1$$

$$2)\ \mathbf{w}^T\mathbf{x} \le 0 \Rightarrow \mathbf{x} \in C_2$$

پرسپترون (Perceptron)

الگوریتم پرسپترون:

$$1)\ \mathbf{w}^T\mathbf{x} > 0 \Rightarrow \mathbf{x} \in C_1$$

$$2)\ \mathbf{w}^T\mathbf{x} \le 0 \Rightarrow \mathbf{x} \in C_2$$

صورت مساله الگوریتم:

با داشتن دو دسته الگوی ورودی که به‌طور خطی جداپذیر باشند، مساله عبارت است از یافتن بردار وزن (w) به‌طوری که دو شرط بالا برآورده شوند.

پرسپترون (Perceptron)

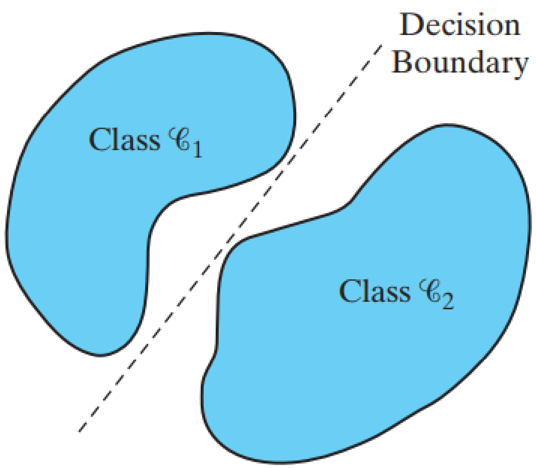
الگوریتم پرسپترون:

$$1)\ \mathbf{w}^T\mathbf{x} > 0 \Rightarrow \mathbf{x} \in C_1$$

$$2)\ \mathbf{w}^T\mathbf{x} \le 0 \Rightarrow \mathbf{x} \in C_2$$

صورت مساله الگوریتم:

با داشتن دو دسته الگوی ورودی که به‌طور خطی جداپذیر باشند، مساله عبارت است از یافتن بردار وزن (w) به‌طوری که دو شرط بالا برآورده شوند.



جداپذیر به‌صورت خطی

پرسپترون (Perceptron)

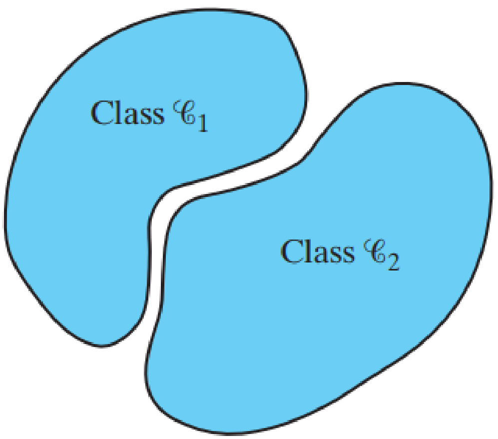
الگوریتم پرسپترون:

$$1)\ \mathbf{w}^T\mathbf{x} > 0 \Rightarrow \mathbf{x} \in C_1$$

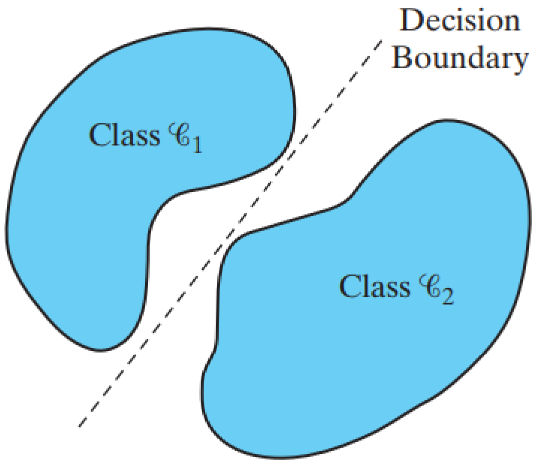
$$2)\ \mathbf{w}^T\mathbf{x} \le 0 \Rightarrow \mathbf{x} \in C_2$$

صورت مساله الگوریتم:

با داشتن دو دسته الگوی ورودی که به‌طور خطی جداپذیر باشند، مساله عبارت است از یافتن بردار وزن (w) به‌طوری که دو شرط بالا برآورده شوند.



جداپذیر به‌صورت غیرخطی



جداپذیر به‌صورت خطی

پرسپترون (Perceptron)

الگوریتم پرسپترون:

آموزش وزن‌ها (یا به‌عبارتی یافتن مقدار بهینه ورن‌ها):

پرسپترون (Perceptron)

الگوریتم پرسپترون:

آموزش وزن‌ها (یا به‌عبارتی یافتن مقدار بهینه ورن‌ها):

۱– کلاسه‌بندی صحیح در لحظه تکرار nام

$$\begin{cases} a)\ \mathbf{w}(n+1) = \mathbf{w}(n) \ \text{ if } \ \mathbf{w}^T(n)\mathbf{x}(n) > 0 \ \text{ and } \ \mathbf{x}(n) \in C_1 \\ b)\ \mathbf{w}(n+1) = \mathbf{w}(n) \ \text{ if } \ \mathbf{w}^T(n)\mathbf{x}(n) \le 0 \ \text{ and } \ \mathbf{x}(n) \in C_2 \end{cases}$$

پرسپترون (Perceptron)

الگوریتم پرسپترون:

آموزش وزن‌ها (یا به‌عبارتی یافتن مقدار بهینه ورن‌ها):

۱– کلاسه‌بندی صحیح در لحظه تکرار nام

$$\begin{cases} a)\ \mathbf{w}(n+1)=\mathbf{w}(n) \ \text{ if } \ \mathbf{w}^T(n)\,\mathbf{x}(n)>0 \ \text{ and } \ \mathbf{x}(n)\in C_1 \\ b)\ \mathbf{w}(n+1)=\mathbf{w}(n) \ \text{ if } \ \mathbf{w}^T(n)\,\mathbf{x}(n)\le 0 \ \text{ and } \ \mathbf{x}(n)\in C_2 \end{cases}$$

۲– کلاسه‌بندی غلط در لحظه تکرار nام

$$\begin{cases} a)\ \mathbf{w}(n+1)=\mathbf{w}(n)-\eta(n)\mathbf{x}(n) \ \text{ if } \ \mathbf{w}^T(n)\,\mathbf{x}(n)>0 \ \text{ and } \ \mathbf{x}(n)\in C_2 \\ b)\ \mathbf{w}(n+1)=\mathbf{w}(n)+\eta(n)\mathbf{x}(n) \ \text{ if } \ \mathbf{w}^T(n)\,\mathbf{x}(n)\le 0 \ \text{ and } \ \mathbf{x}(n)\in C_1 \end{cases}$$

پرسپترون (Perceptron)

الگوریتم پرسپترون:

آموزش وزن‌ها (یا به‌عبارتی یافتن مقدار بهینه ورن‌ها):

۱– کلاسه‌بندی صحیح در لحظه تکرار nام

$$\begin{cases} a)\ \mathbf{w}(n+1)=\mathbf{w}(n) \ \ \text{if} \ \ \mathbf{w}^T(n)\,\mathbf{x}(n)>0 \ \text{ and } \ \mathbf{x}(n)\in C_1 \\ b)\ \mathbf{w}(n+1)=\mathbf{w}(n) \ \ \text{if} \ \ \mathbf{w}^T(n)\,\mathbf{x}(n)\le 0 \ \text{ and } \ \mathbf{x}(n)\in C_2 \end{cases}$$

۲– کلاسه‌بندی غلط در لحظه تکرار nام

$$\begin{cases} a)\ \mathbf{w}(n+1)=\mathbf{w}(n)-\eta(n)\mathbf{x}(n) \ \ \text{if} \ \ \mathbf{w}^T(n)\,\mathbf{x}(n)>0 \ \text{ and } \ \mathbf{x}(n)\in C_2 \\ b)\ \mathbf{w}(n+1)=\mathbf{w}(n)+\eta(n)\mathbf{x}(n) \ \ \text{if} \ \ \mathbf{w}^T(n)\,\mathbf{x}(n)\le 0 \ \text{ and } \ \mathbf{x}(n)\in C_1 \end{cases}$$

$\eta(n)>0$ ضریب آموزش (یادگیری)؛ یعنی کنترل مقدار تغییرات وزن‌ها در هر مرتبه تکرار آموزش

پرسپترون (Perceptron)

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

فرض‌های ساده کننده

$$\mathbf{w}(1) = 0, \quad \eta(n) = 1$$

$$\mathbf{w}^T(n)\,\mathbf{x}(n) \le 0 \ \text{ and } \ \mathbf{x}(n) \in C_1$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

فرض‌های ساده‌کننده

$$\mathbf{w}(1) = 0, \quad \eta(n) = 1$$

$$\mathbf{w}^T(n)\,\mathbf{x}(n) \le 0 \ \text{ and } \ \mathbf{x}(n) \in C_1$$

بنابراین

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \eta(n)\mathbf{x}(n) \quad \Rightarrow$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

فرض‌های ساده‌کننده

$$\mathbf{w}(1) = 0, \quad \eta(n) = 1$$

$$\mathbf{w}^T(n)\,\mathbf{x}(n) \le 0 \ \text{ and } \ \mathbf{x}(n) \in C_1$$

بنابراین

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \eta(n)\mathbf{x}(n) \quad \Rightarrow \quad \mathbf{w}(n+1) = \mathbf{w}(n) + \mathbf{x}(n)$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

فرض‌های ساده‌کننده

$$\mathbf{w}(1)=0,\quad \eta(n)=1$$

$$\mathbf{w}^T(n)\,\mathbf{x}(n)\le 0 \text{ and } \mathbf{x}(n)\in C_1$$

بنابراین

$$\mathbf{w}(n+1)=\mathbf{w}(n)+\eta(n)\mathbf{x}(n) \quad\Rightarrow\quad \mathbf{w}(n+1)=\mathbf{w}(n)+\mathbf{x}(n)$$

با تکرار الگوریتم

$$n=1 \qquad \mathbf{w}(2)=\mathbf{w}(1)+\mathbf{x}(1)$$

$$n=2 \qquad \mathbf{w}(3)=\mathbf{w}(2)+\mathbf{x}(2)=\mathbf{x}(1)+\mathbf{x}(2)$$

.

.

.

$$n=n \qquad \mathbf{w}(n+1)=\mathbf{x}(1)+\cdots+\mathbf{x}(n)$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

$$\mathbf{w}(n+1) = \mathbf{x}(1) + \cdots + \mathbf{x}(n)$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

$$\mathbf{w}(n+1) = \mathbf{x}(1) + \cdots + \mathbf{x}(n)$$

برطبق فرض جداپذیری خطی، همواره جوابی مثل $\mathbf{w}_0$ وجود دارد به‌طوری‌که

$$\mathbf{w}_0^T \mathbf{x} > 0$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

$$\mathbf{w}(n+1) = \mathbf{x}(1) + \cdots + \mathbf{x}(n)$$

برطبق فرض جداپذیری خطی، همواره جوابی مثل w_0 وجود دارد به‌طوری‌که

$$\mathbf{w}_0^T \mathbf{x} > 0$$

کوچکترین این حاصلضرب

$$\alpha := \min_{\mathbf{x}(n) \in C_1} \mathbf{w}_0^T \mathbf{x}(n)$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

$$\mathbf{w}(n+1) = \mathbf{x}(1) + \cdots + \mathbf{x}(n)$$

برطبق فرض جداپذیری خطی، همواره جوابی مثل $\mathbf{w}_0$ وجود دارد به‌طوری‌که

$$\mathbf{w}_0^T \mathbf{x} > 0$$

کوچکترین این حاصلضرب

$$\alpha := \min_{\mathbf{x}(n) \in C_1} \mathbf{w}_0^T \mathbf{x}(n)$$

از طرفی، با ضرب‌کردن دو طرف رابطه تنظیم وزن‌ها در $\mathbf{w}_0^T$

$$\mathbf{w}_0^T \mathbf{w}(n+1) = \mathbf{w}_0^T \mathbf{x}(1) + \cdots + \mathbf{w}_0^T \mathbf{x}(n)$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

$$\mathbf{w}(n+1) = \mathbf{x}(1) + \cdots + \mathbf{x}(n)$$

برطبق فرض جداپذیری خطی، همواره جوابی مثل $\mathbf{w}_0$ وجود دارد به‌طوری‌که

$$\mathbf{w}_0^T \mathbf{x} > 0$$

کوچکترین این حاصلضرب

$$\alpha := \min_{\mathbf{x}(n) \in C_1} \mathbf{w}_0^T \mathbf{x}(n)$$

از طرفی، با ضرب‌کردن دو طرف رابطه تنظیم وزن‌ها در $\mathbf{w}_0^T$

$$\mathbf{w}_0^T \mathbf{w}(n+1) = \mathbf{w}_0^T \mathbf{x}(1) + \cdots + \mathbf{w}_0^T \mathbf{x}(n)$$

با مقایسه دو رابطه اخیر

$$\mathbf{w}_0^T \mathbf{w}(n+1) \geq n\alpha$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

$$\mathbf{w}(n+1) = \mathbf{x}(1) + \cdots + \mathbf{x}(n)$$

برطبق فرض جداپذیری خطی، همواره جوابی مثل $\mathbf{w}_0$ وجود دارد به‌طوری‌که

$$\mathbf{w}_0^T \mathbf{x} > 0$$

کوچکترین این حاصلضرب

$$\alpha := \min_{\mathbf{x}(n) \in C_1} \mathbf{w}_0^T \mathbf{x}(n)$$

از طرفی، با ضرب‌کردن دو طرف رابطه تنظیم وزن‌ها در $\mathbf{w}_0^T$

$$\mathbf{w}_0^T \mathbf{w}(n+1) = \mathbf{w}_0^T \mathbf{x}(1) + \cdots + \mathbf{w}_0^T \mathbf{x}(n)$$

با مقایسه دو رابطه اخیر

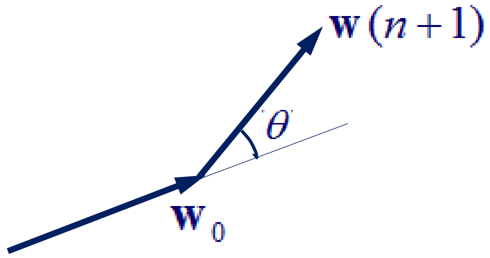
$$\mathbf{w}_0^T \mathbf{w}(n+1) \geq n\alpha$$

با استفاده از نابرابری کوشی-شوارتز (Cauchy-Schwartz)

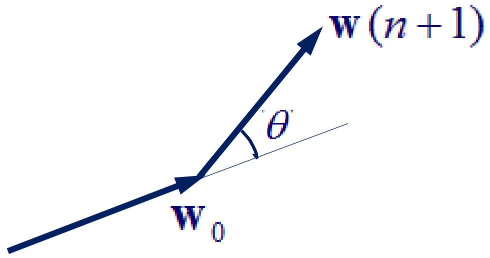
$$\left\|\mathbf{w}_0^T\right\|^2 \left\|\mathbf{w}(n+1)\right\|^2 \geq n^2 \alpha^2$$

نابرابری کوشی-شوارتز (Cauchy-Schwartz):

نابرابری کوشی-شوارتز (Cauchy-Schwartz):

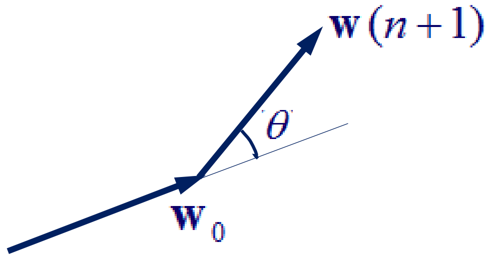


نابرابری کوشی-شوارتز (Cauchy-Schwartz):



برای دو بردار $\mathbf{w}_0$ و $\mathbf{w}(n+1)$ داریم:

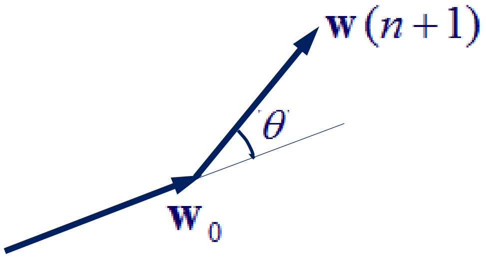
نابرابری کوشی-شوارتز (Cauchy-Schwartz):



برای دو بردار $\mathbf{w}_0$ و $\mathbf{w}(n+1)$ داریم:

$$\mathbf{w}_0^T\mathbf{w}(n+1) = \left\|\mathbf{w}_0^T\right\|\left\|\mathbf{w}(n+1)\right\|\cos(\theta)$$

نابرابری کوشی–شوارتز (Cauchy-Schwartz):

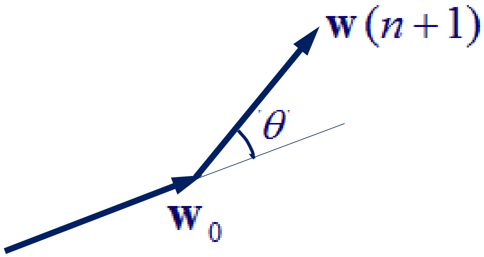


برای دو بردار $\mathbf{w}_0$ و $\mathbf{w}(n+1)$ داریم:

$$\mathbf{w}_0^T\mathbf{w}(n+1) = \left\|\mathbf{w}_0^T\right\|\left\|\mathbf{w}(n+1)\right\|\cos(\theta)$$

$$\left[\mathbf{w}_0^T\mathbf{w}(n+1)\right]^2 = \left\|\mathbf{w}_0^T\right\|^2\left\|\mathbf{w}(n+1)\right\|^2\cos^2(\theta)$$

نابرابری کوشی-شوارتز (Cauchy-Schwartz):



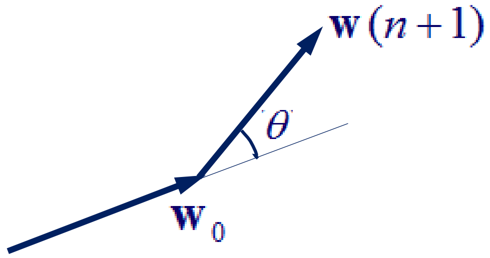
برای دو بردار $\mathbf{w}_0$ و $\mathbf{w}(n+1)$ داریم:

$$\mathbf{w}_0^T \mathbf{w}(n+1) = \left\| \mathbf{w}_0^T \right\| \left\| \mathbf{w}(n+1) \right\| \cos(\theta)$$

$$\left[\mathbf{w}_0^T \mathbf{w}(n+1) \right]^2 = \left\| \mathbf{w}_0^T \right\|^2 \left\| \mathbf{w}(n+1) \right\|^2 \cos^2(\theta)$$

$$\cos(\theta) \le 1$$

<u>نابرابری کوشی-شوارتز (Cauchy-Schwartz):</u>



برای دو بردار $\mathbf{w}_0$ و $\mathbf{w}(n+1)$ داریم:

$$\mathbf{w}_0^T\mathbf{w}(n+1) = \left\|\mathbf{w}_0^T\right\| \left\|\mathbf{w}(n+1)\right\| \cos(\theta)$$

$$\left[\mathbf{w}_0^T\mathbf{w}(n+1)\right]^2 = \left\|\mathbf{w}_0^T\right\|^2 \left\|\mathbf{w}(n+1)\right\|^2 \cos^2(\theta)$$

$$\cos(\theta) \le 1$$

$$\left[\mathbf{w}_0^T\mathbf{w}(n+1)\right]^2 \le \left\|\mathbf{w}_0^T\right\|^2 \left\|\mathbf{w}(n+1)\right\|^2$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

$$\mathbf{w}(n+1)=\mathbf{x}(1)+\cdots+\mathbf{x}(n)$$

برطبق فرض جداپذیری خطی، همواره جوابی مثل $\mathbf{w}_0$ وجود دارد به‌طوری‌که

$$\mathbf{w}_0^T\mathbf{x}>0$$

کوچکترین این حاصلضرب

$$\alpha:=\min_{\mathbf{x}(n)\in C_1}\mathbf{w}_0^T\mathbf{x}(n)$$

از طرفی، با ضرب‌کردن دو طرف رابطه تنظیم وزن ها

$$\mathbf{w}_0^T\mathbf{w}(n+1)=\mathbf{w}_0^T\mathbf{x}(1)+\cdots+\mathbf{w}_0^T\mathbf{x}(n)$$

با مقایسه دو رابطه اخیر

$$\mathbf{w}_0^T\mathbf{w}(n+1)\geq n\alpha$$

با استفاده از نابرابری کوشی-شوارتز (Cauchy-Schwartz)

$$\left\|\mathbf{w}_0^T\right\|^2\left\|\mathbf{w}(n+1)\right\|^2\geq n^2\alpha^2$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

$$\left\|\mathbf{w}_0^T\right\|^2 \left\|\mathbf{w}(n+1)\right\|^2 \geq n^2\alpha^2$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

$$\left\|\mathbf{w}_0^T\right\|^2 \left\|\mathbf{w}(n+1)\right\|^2 \geq n^2\alpha^2$$

$$\Rightarrow \left\|\mathbf{w}(n+1)\right\|^2 \geq \frac{n^2\alpha^2}{\left\|\mathbf{w}_0^T\right\|^2}$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

$$\left\|\mathbf{w}_0^T\right\|^2 \left\|\mathbf{w}(n+1)\right\|^2 \geq n^2\alpha^2$$

$$\boxed{\Rightarrow \left\|\mathbf{w}(n+1)\right\|^2 \geq \frac{n^2\alpha^2}{\left\|\mathbf{w}_0^T\right\|^2}}$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

$$\left\|\mathbf{w}_0^T\right\|^2 \left\|\mathbf{w}(n+1)\right\|^2 \geq n^2\alpha^2$$

$$\boxed{\Rightarrow \left\|\mathbf{w}(n+1)\right\|^2 \geq \frac{n^2\alpha^2}{\left\|\mathbf{w}_0^T\right\|^2}}$$

حال، $\left\|\mathbf{w}(n+1)\right\|^2$ را از راه دیگری نیز به‌دست می‌آوریم.

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

قبلا داشتیم

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mathbf{x}(k)$$

پرسپترون (Perceptron)

همگرايي الگوريتم پرسپترون:

قبلا داشتیم

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mathbf{x}(k)$$

طول (اندازه اقلیدسی) بردارهای فوق

$$\left\|\mathbf{w}(k+1)\right\|^2 = \left\|\mathbf{w}(k)\right\|^2 + \left\|\mathbf{x}(k)\right\|^2 + 2\mathbf{w}^T(k)\mathbf{x}(k)$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

قبلا داشتیم

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mathbf{x}(k)$$

طول (اندازه اقلیدسی) بردارهای فوق

$$\|\mathbf{w}(k+1)\|^2 = \|\mathbf{w}(k)\|^2 + \|\mathbf{x}(k)\|^2 + 2\mathbf{w}^T(k)\mathbf{x}(k)$$

فرض‌شده که بردار ورودی $\mathbf{x}(k)$ غلط کلاسه بندی شده، یعنی

$$\mathbf{w}^T(k)\,\mathbf{x}(k) \le 0$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

قبلا داشتیم

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mathbf{x}(k)$$

طول (اندازه اقلیدسی) بردارهای فوق

$$\|\mathbf{w}(k+1)\|^2 = \|\mathbf{w}(k)\|^2 + \|\mathbf{x}(k)\|^2 + 2\mathbf{w}^T(k)\mathbf{x}(k)$$

فرض‌شده که بردار ورودی $\mathbf{x}(k)$ غلط کلاسه بندی شده، یعنی

$$\mathbf{w}^T(k)\ \mathbf{x}(k) \leq 0$$

با مقایسه دو رابطه اخیر

$$\|\mathbf{w}(k+1)\|^2 \leq \|\mathbf{w}(k)\|^2 + \|\mathbf{x}(k)\|^2$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

قبلا داشتیم

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mathbf{x}(k)$$

طول (اندازه اقلیدسی) بردارهای فوق

$$\|\mathbf{w}(k+1)\|^2 = \|\mathbf{w}(k)\|^2 + \|\mathbf{x}(k)\|^2 + 2\mathbf{w}^T(k)\mathbf{x}(k)$$

فرض‌شده که بردار ورودی $\mathbf{x}(k)$ غلط کلاسه بندی شده، یعنی

$$\mathbf{w}^T(k)\,\mathbf{x}(k) \leq 0$$

با مقایسه دو رابطه اخیر

$$\|\mathbf{w}(k+1)\|^2 \leq \|\mathbf{w}(k)\|^2 + \|\mathbf{x}(k)\|^2$$

$$\Rightarrow \|\mathbf{w}(k+1)\|^2 - \|\mathbf{w}(k)\|^2 \leq \|\mathbf{x}(k)\|^2 \quad k = 1, \ldots, n$$

پرسپترون (Perceptron)

<u>همگرایی الگوریتم پرسپترون:</u>

$$\left\|\mathbf{w}(k+1)\right\|^2-\left\|\mathbf{w}(k)\right\|^2\le\left\|\mathbf{x}(k)\right\|^2\quad k=1,\ldots,n$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

$$\|\mathbf{w}(k+1)\|^2 - \|\mathbf{w}(k)\|^2 \le \|\mathbf{x}(k)\|^2 \quad k = 1,\dots,n$$

با تکرار الگوریتم

$$\begin{array}{ll}
k=1 & \|\mathbf{w}(2)\|^2 - \|\mathbf{w}(1)\|^2 \le \|\mathbf{x}(1)\|^2 \\
k=2 & \|\mathbf{w}(3)\|^2 - \|\mathbf{w}(2)\|^2 \le \|\mathbf{x}(2)\|^2 \\
 & \vdots \\
k=n & \|\mathbf{w}(n+1)\|^2 - \|\mathbf{w}(n)\|^2 \le \|\mathbf{x}(n)\|^2 \\
\hline
 & \|\mathbf{w}(n+1)\|^2 \le \sum_{k=1}^{n} \|\mathbf{x}(k)\|^2
\end{array}$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

$$\|\mathbf{w}(k+1)\|^2 - \|\mathbf{w}(k)\|^2 \le \|\mathbf{x}(k)\|^2 \quad k = 1,\ldots,n$$

با تکرار الگوریتم

$$\begin{array}{ll}
k=1 & \|\mathbf{w}(2)\|^2 - \|\mathbf{w}(1)\|^2 \le \|\mathbf{x}(1)\|^2 \\
k=2 & \|\mathbf{w}(3)\|^2 - \|\mathbf{w}(2)\|^2 \le \|\mathbf{x}(2)\|^2 \\
 & \vdots \\
k=n & \|\mathbf{w}(n+1)\|^2 - \|\mathbf{w}(n)\|^2 \le \|\mathbf{x}(n)\|^2 \\
\hline
 & \|\mathbf{w}(n+1)\|^2 \le \sum_{k=1}^{n} \|\mathbf{x}(k)\|^2
\end{array}$$

فرض کنید

$$\beta := \max_{\mathbf{x}(k)\in C_1} \|\mathbf{x}(k)\|^2$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

$$\|\mathbf{w}(k+1)\|^2 - \|\mathbf{w}(k)\|^2 \le \|\mathbf{x}(k)\|^2 \quad k = 1,\dots,n$$

با تکرار الگوریتم

$$\begin{aligned}
k &= 1 \qquad \|\mathbf{w}(2)\|^2 - \|\mathbf{w}(1)\|^2 \le \|\mathbf{x}(1)\|^2 \\
k &= 2 \qquad \|\mathbf{w}(3)\|^2 - \|\mathbf{w}(2)\|^2 \le \|\mathbf{x}(2)\|^2 \\
&\ \vdots \\
k &= n \qquad \|\mathbf{w}(n+1)\|^2 - \|\mathbf{w}(n)\|^2 \le \|\mathbf{x}(n)\|^2
\end{aligned}$$

$$\|\mathbf{w}(n+1)\|^2 \le \sum_{k=1}^{n} \|\mathbf{x}(k)\|^2$$

فرض کنید

$$\beta := \max_{\mathbf{x}(k)\in C_1} \|\mathbf{x}(k)\|^2$$

$$\Rightarrow \boxed{\|\mathbf{w}(n+1)\|^2 \le n\beta}$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

$$\left\| \mathbf{w}(n+1) \right\|^2 \le n\beta$$

$$\left\| \mathbf{w}(n+1) \right\|^2 \ge \frac{n^2\alpha^2}{\left\| \mathbf{w}_0^T \right\|^2}$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

$$\|\mathbf{w}(n+1)\|^2 \le n\beta$$

$$\|\mathbf{w}(n+1)\|^2 \ge \frac{n^2\alpha^2}{\|\mathbf{w}_0^T\|^2}$$

– دو رابطه بالا را با هم مقایسه کنید!

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

$$\|\mathbf{w}(n+1)\|^2 \ge \frac{n^2\alpha^2}{\|\mathbf{w}_0^T\|^2}$$

$$\|\mathbf{w}(n+1)\|^2 \le n\beta$$

- دو رابطه بالا را با هم مقایسه کنید!
- این دو رابطه با هم در تناقض‌اند، مگر تساوی آن‌ها

پرسپترون (Perceptron)

همگرايي الگوريتم پرسپترون:

$$\|\mathbf{w}(n+1)\|^2 \le n\beta$$

$$\|\mathbf{w}(n+1)\|^2 \ge \frac{n^2\alpha^2}{\|\mathbf{w}_0^T\|^2}$$

- دو رابطه بالا را با هم مقايسه كنيد!
- اين دو رابطه با هم در تناقض‌اند، مگر تساوی آن‌ها

$$\frac{n^2\alpha^2}{\|\mathbf{w}_0^T\|^2} = n\beta$$

پرسپترون (Perceptron)

همگرایی الگوریتم پرسپترون:

$$\|\mathbf{w}(n+1)\|^2 \geq \frac{n^2\alpha^2}{\|\mathbf{w}_0^T\|^2}$$

$$\|\mathbf{w}(n+1)\|^2 \leq n\beta$$

– دو رابطه بالا را با هم مقایسه کنید!

– این دو رابطه با هم در تناقض‌اند، مگر تساوی آن‌ها

$$\frac{n^2\alpha^2}{\|\mathbf{w}_0^T\|^2} = n\beta$$

$$n_{\max} = \frac{\beta\|\mathbf{w}_0^T\|^2}{\alpha^2}$$

پرسپترون (Perceptron)

الگوریتم پرسپترون:

پرسپترون (Perceptron)

الگوریتم پرسپترون:

برای حالت کلی $\mathbf{w}(1) \neq 0$ و $0 < \eta(n) < 1$ فقط تغییر مقدار $n_{\max}$ را خواهیم داشت.

پرسپترون (Perceptron)

الگوریتم پرسپترون:

برای حالت کلی $\mathbf{w}(1) \neq 0$ و $0 < \eta(n) < 1$ فقط تغییر مقدار $n_{\max}$ را خواهیم داشت.

پرسپترون (Perceptron)

الگوریتم پرسپترون:

برای حالت کلی $\mathbf{w}(1) \neq 0$ و $0 < \eta(n) < 1$ فقط تغییر مقدار $n_{\max}$ را خواهیم داشت.

خلاصه الگوریتم پرسپترون:

پرسپترون (Perceptron)

الگوریتم پرسپترون:

برای حالت کلی $\mathbf{w}(1) \neq 0$ و $0 < \eta(n) < 1$ فقط تغییر مقدار $n_{\max}$ را خواهیم داشت.

خلاصه الگوریتم پرسپترون:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \eta(n)\left[d(n) - y(n)\right]\mathbf{x}(n)$$

پرسپترون (Perceptron)

الگوریتم پرسپترون:

برای حالت کلی $\mathbf{w}(1) \neq 0$ و $0 < \eta(n) < 1$ فقط تغییر مقدار $n_{\max}$ را خواهیم داشت.

خلاصه الگوریتم پرسپترون:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \eta(n)\left[d(n) - y(n)\right]\mathbf{x}(n)$$

که در آن

$$d(n) = \begin{cases} +1 & \mathbf{x}(n) \in C_1 \\ -1 & \mathbf{x}(n) \in C_2 \end{cases}$$

$$y(n) = \operatorname{sgn}\left[\mathbf{w}^T(n)\,\mathbf{x}(n)\right]$$

پرسپترون (Perceptron)

<u>تمرین:</u> نشان دهید که با استفاده از معیار کارآیی (Performance Index) زیر می‌توان همان الگوریتم پرسپترون را استخراج کرد

$$J = -E\left[e(n)v(n)\right]$$

که مقدار لحظه‌ای آن برابر است با

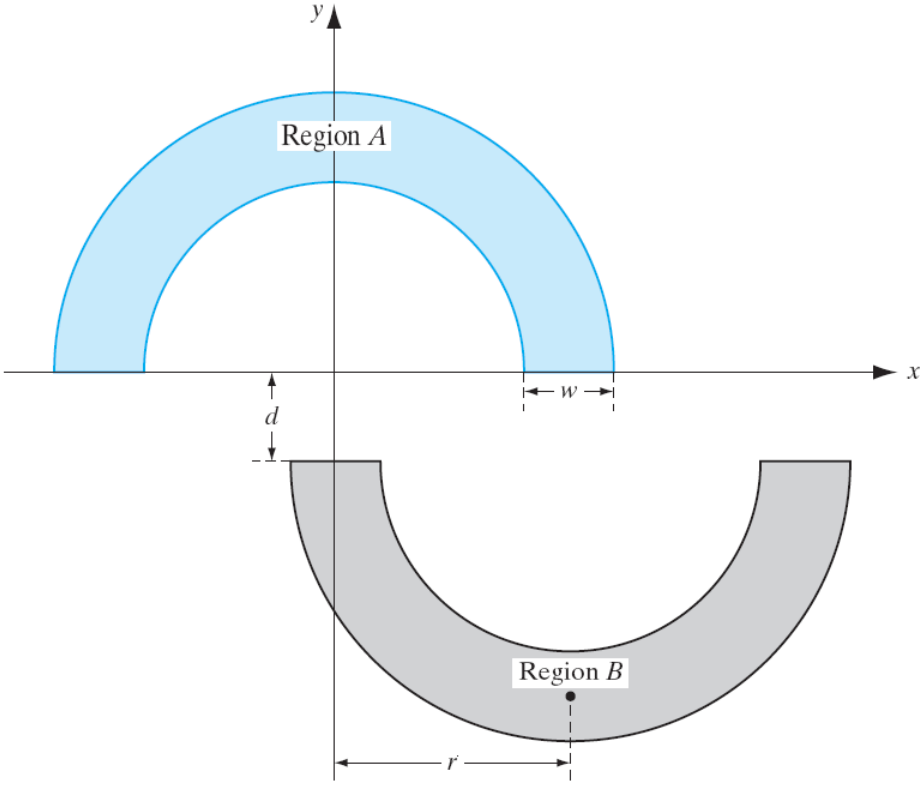
$$\begin{aligned}\hat{J}(n) &= -e(n)v(n)\\ &= -\left[d(n) - y(n)\right]v(n)\end{aligned}$$

پرسپترون (Perceptron)

مساله کلاسه‌بندی (Classification Problem)

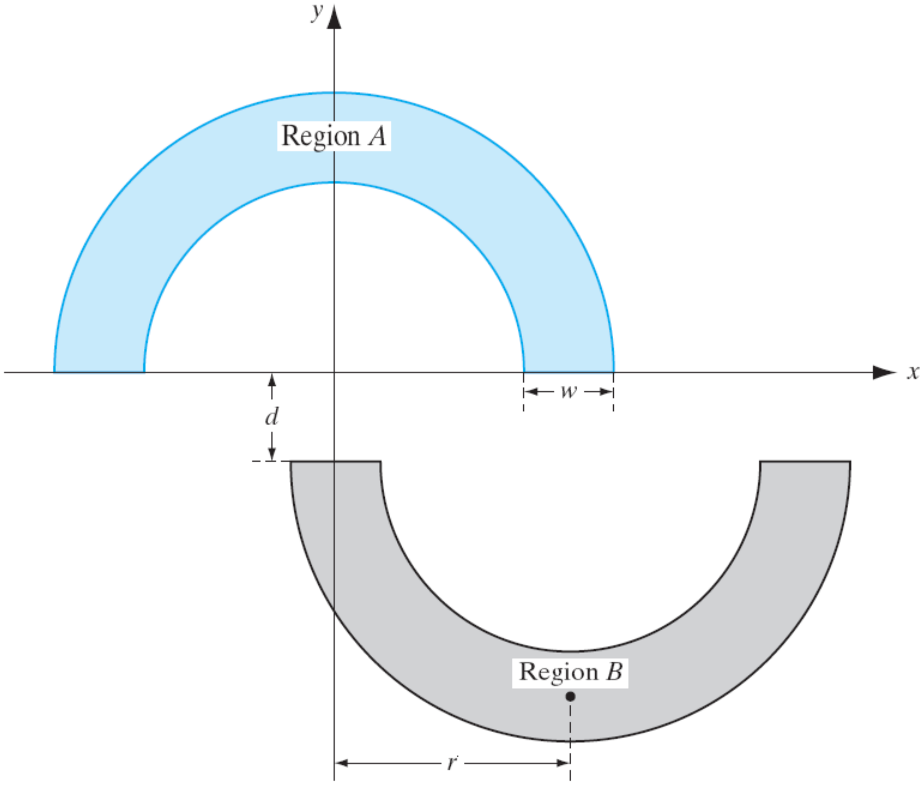
پرسپترون (Perceptron)

مساله کلاسه‌بندی (Classification Problem)



پرسپترون (Perceptron)

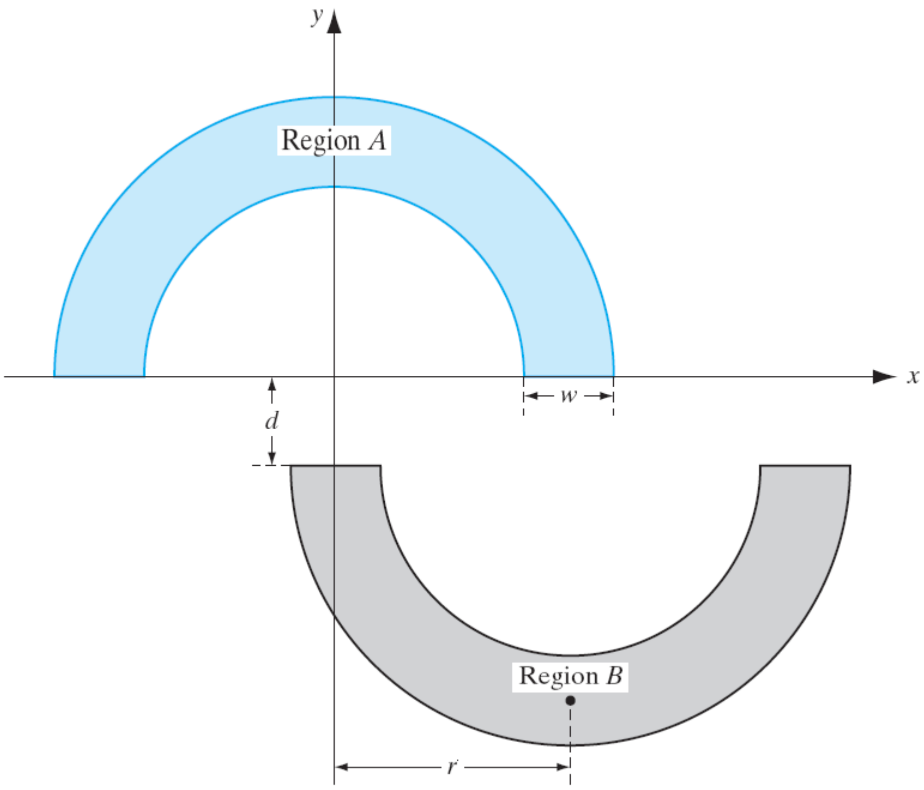
مساله کلاسه‌بندی (Classification Problem)



- دو کلاس از داده‌ها که در دو ناحیه A و B به شکل ماه که به‌صورت نامتقارن در مقابل یکدیگر قرارگرفته‌اند.

پرسپترون (Perceptron)

مساله کلاسه‌بندی (Classification Problem)



- دو کلاس از داده‌ها که در دو ناحیه A و B به شکل ماه که به‌صورت نامتقارن در مقابل یکدیگر قرارگرفته‌اند.
- شعاع و پهنای هر دو ناحیه

  $r = 10, \quad w = 6$

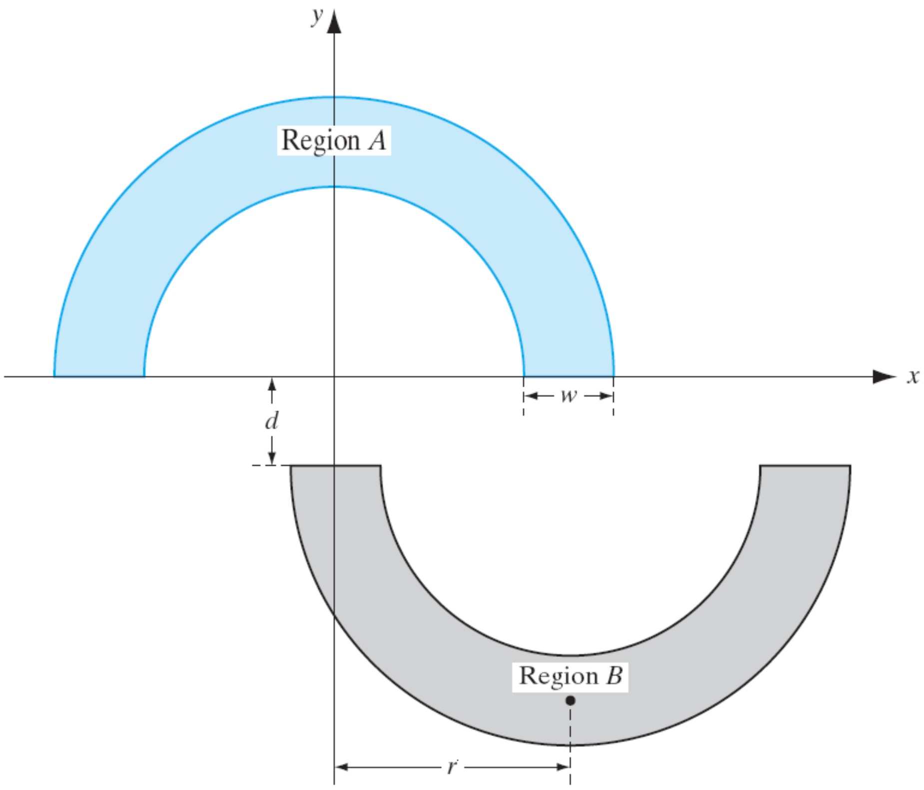
- $d > 0$ افزایش جداپذیری کلاس‌ها از یکدیگر

  $d < 0$ کاهش جداپذیری کلاس‌ها از یکدیگر

- ۱۰۰۰ نمونه برای آموزش وزن‌ها برابر با یک دوره (epoch) از داده‌ها
- ۲۰۰۰ نمونه برای آزمایش

پرسپترون (Perceptron)

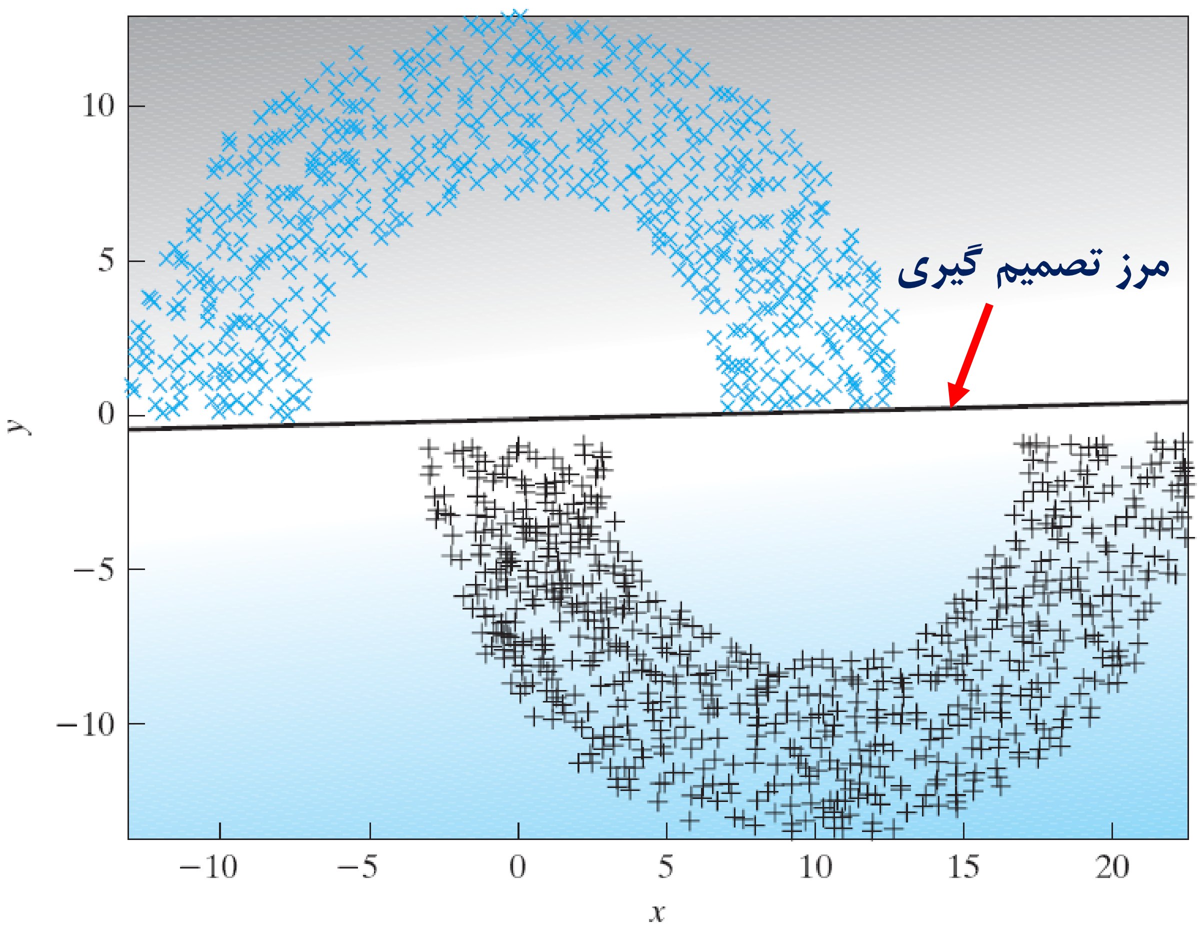
مساله کلاسه‌بندی (Classification Problem)



- ساختار پرسپترون:
  - دو ورودی $(y \quad x)$
  - ضریب آموزش (η) از 0.1 تا 10^{-5} به‌طور خطی کاهش می‌یابد.
  - $\mathbf{w}(0) = \mathbf{0}$

پرسپترون (Perceptron)

مساله کلاسه‌بندی (Classification Problem)

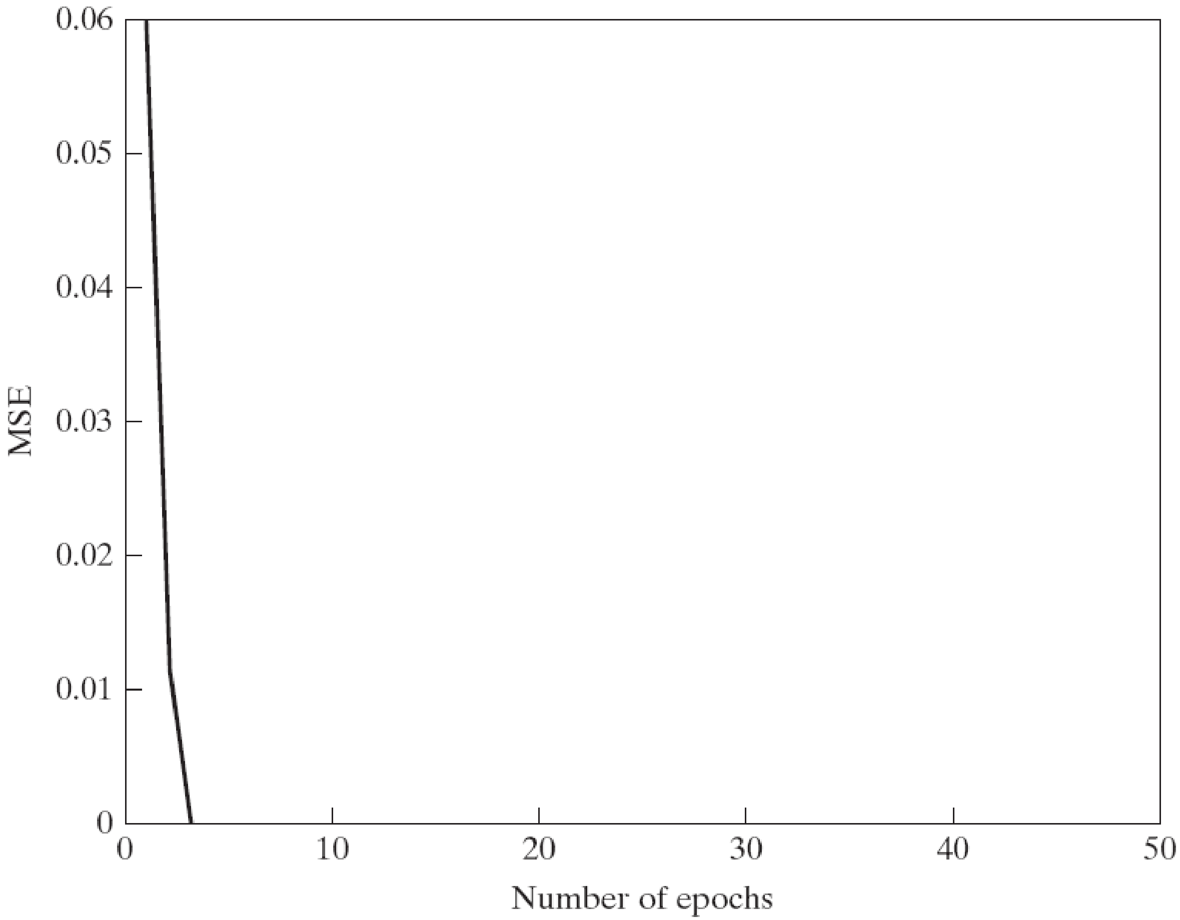


کلاسه‌بندی داده‌های آزمایش

برای $d = 1$

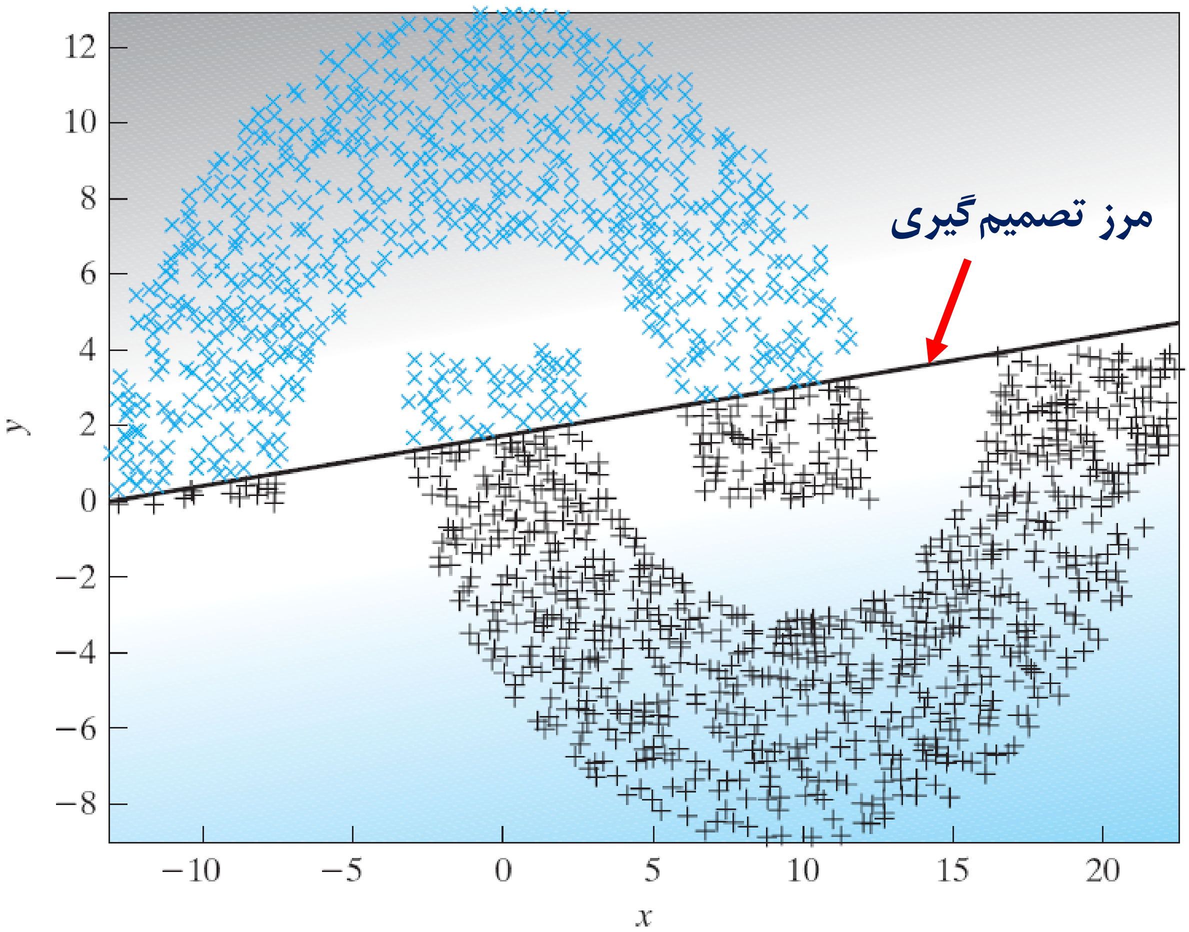
پرسپترون (Perceptron)

مساله کلاسه‌بندی (Classification Problem)



پرسپترون (Perceptron)

مساله کلاسه‌بندی (Classification Problem)

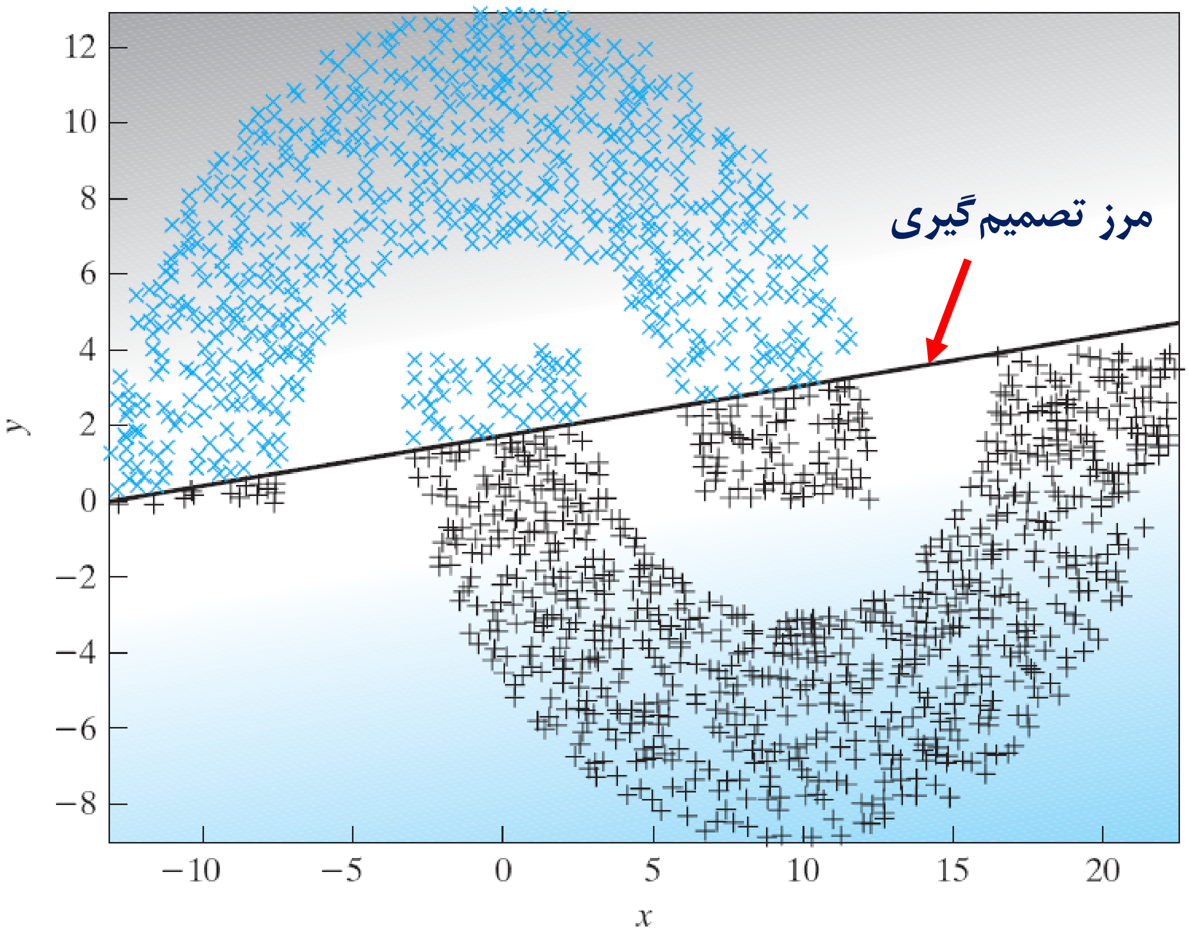


کلاسه‌بندی داده‌های آزمایش

برای $d = -4$

پرسپترون (Perceptron)

مساله کلاسه‌بندی (Classification Problem)



کلاسه‌بندی داده‌های آزمایش

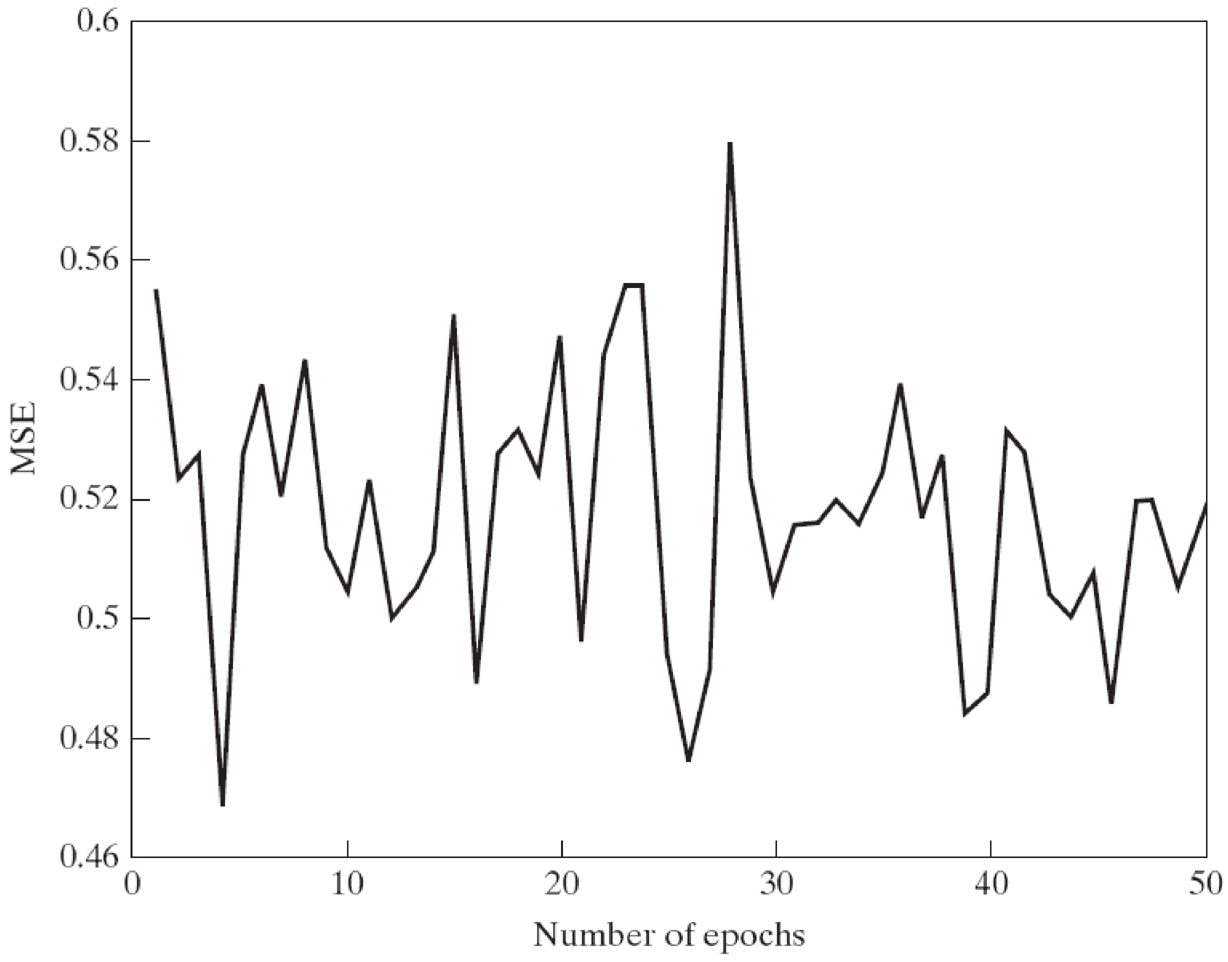
برای $d = -4$

خطای کلاسه‌بندی

$$\frac{186}{2000} \times 100\% = 9.3\%$$

پرسپترون (Perceptron)

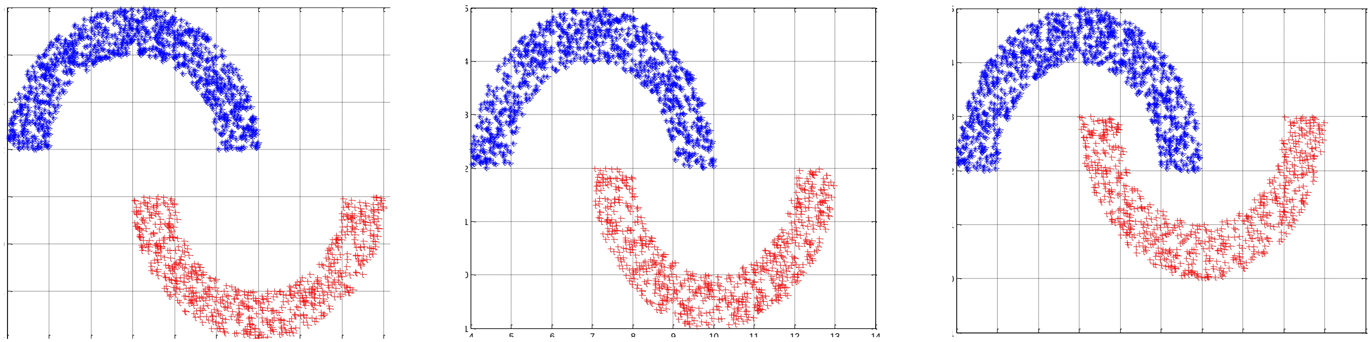
مساله کلاسه‌بندی (Classification Problem)



پرسپترون (Perceptron)

تمرین: الگوریتم پرسپترون را برای دو کلاس ماه‌شکل برای موارد زیر اجرا کرده و نتایج را گزارش کنید:

۱- مقادیر مختلف فاصله دو کلاس از هم (d). یعنی جداپذیری خطی و غیرخطی



۲- ضریب آموزش ثابت و متغیر با زمان

پرسپترون (Perceptron)

پرسپترون – مدل Mark I

Fig. 1.4. • Frank Rosenblatt (the inventor of the perceptron and designer of the Mark I Perceptron neurocomputer) with the 400 pixel (20 × 20) Mark I Perceptron image sensor. Photo courtesy of Arvin Calspan Advanced Technology Center.

- آرایه ۲۰x۲۰ از حسگرهای نوری

پرسپترون (Perceptron)

پرسپترون – مدل Mark I

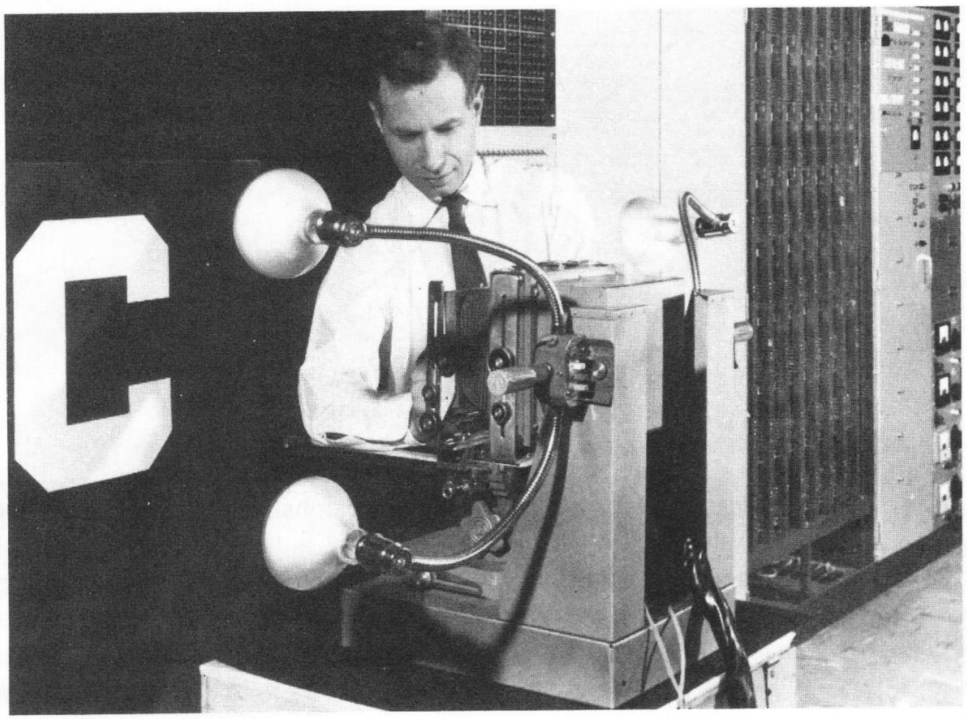


Fig. 1.3. • The Mark I Perceptron image input system being adjusted by Charles Wightman, Mark I Perceptron project engineer. A printed character was mounted on the board and illuminated with four floodlights. The image of the character was focused on a 20 × 20 array of CdS photoconductors — which then provided 400 pixel values for use as inputs to the neural network (which then attempted to classify the figure into one of *M* classes — "A", "B", etc.). Photo courtesy of Arvin Calspan Advanced Technology Center.

- سیستم تصویربرداری ورودی به‌صورت آرایه ۲۰x۲۰
- از ۴۰۰ پیکسل بدست‌آمده به‌عنوان ورودی به پرسپترون استفاده‌می‌شود

پرسپترون (Perceptron)

پرسپترون – مدل Mark I

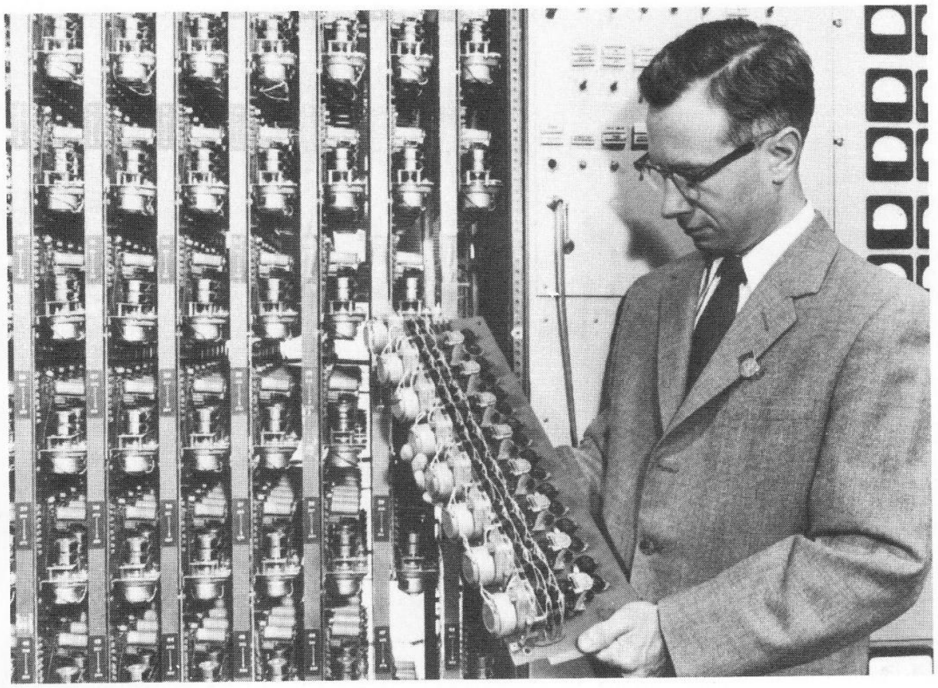


Fig. 1.5. • Charles Wightman holding a subrack of 8 motor/potentiometer pairs. Each motor/potentiometer pair functioned as a single adaptive weight value. The perceptron learning law was implemented in analog circuits that (when properly wired through the patchboard shown in Figure 1.6) would control the motor of each potentiometer (the resistance of which functioned to implement one weight). Photo courtesy of Arvin Calspan Advanced Technology Center.

- وزن‌های تطبیقی پرسپترون به‌صورت موتورهای DC با پتانسیومتر در انتهای محور آن‌ها ساخته‌شده‌اند

پرسپترون (Perceptron)

پرسپترون – مدل Mark I

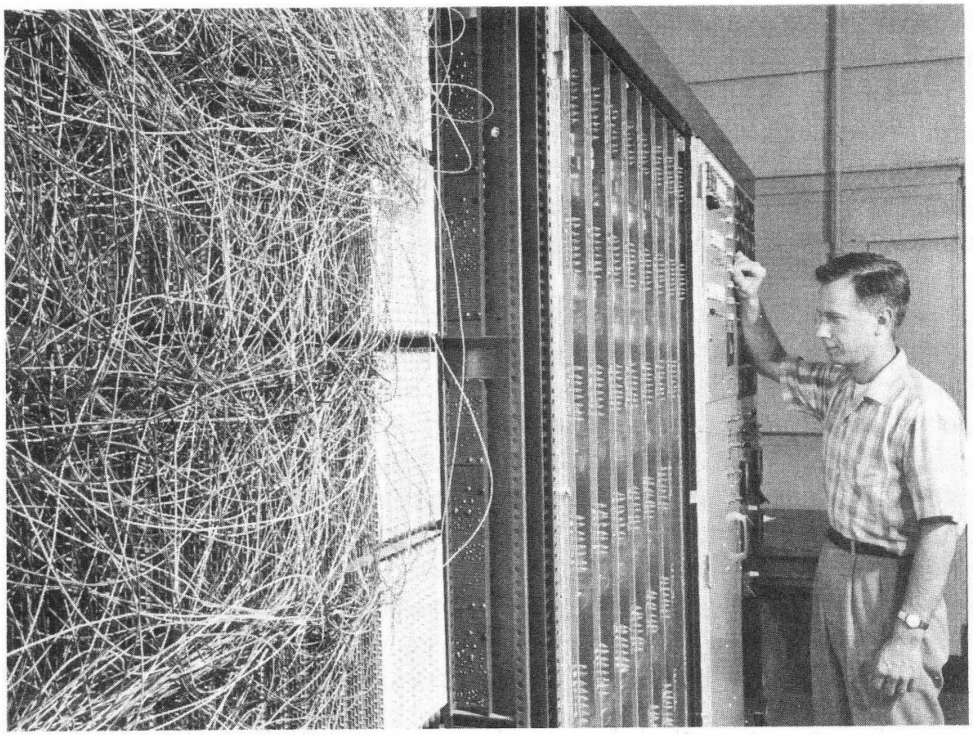


Fig. 1.6. • The Mark I Perceptron patchboard. The connection patterns were typically "random", so as to illustrate the ability of the perceptron to learn the desired pattern without need for precise wiring (in contrast to the precise wiring required in a programmed computer). Photo courtesy of Arvin Calspan Advanced Technology Center.

- اتصالات در پرسپترون (برخلاف رایانه‌های متداول) کاملاً اتفاقی است.
- این سیستم، با استفاده از روش آموزش روزنبلات، قادر به تشخیص حروف یا اعداد می‌باشد.

پرسپترون (Perceptron)

مسائل فصل‌های مقدمه و ۱ را حل کنید!