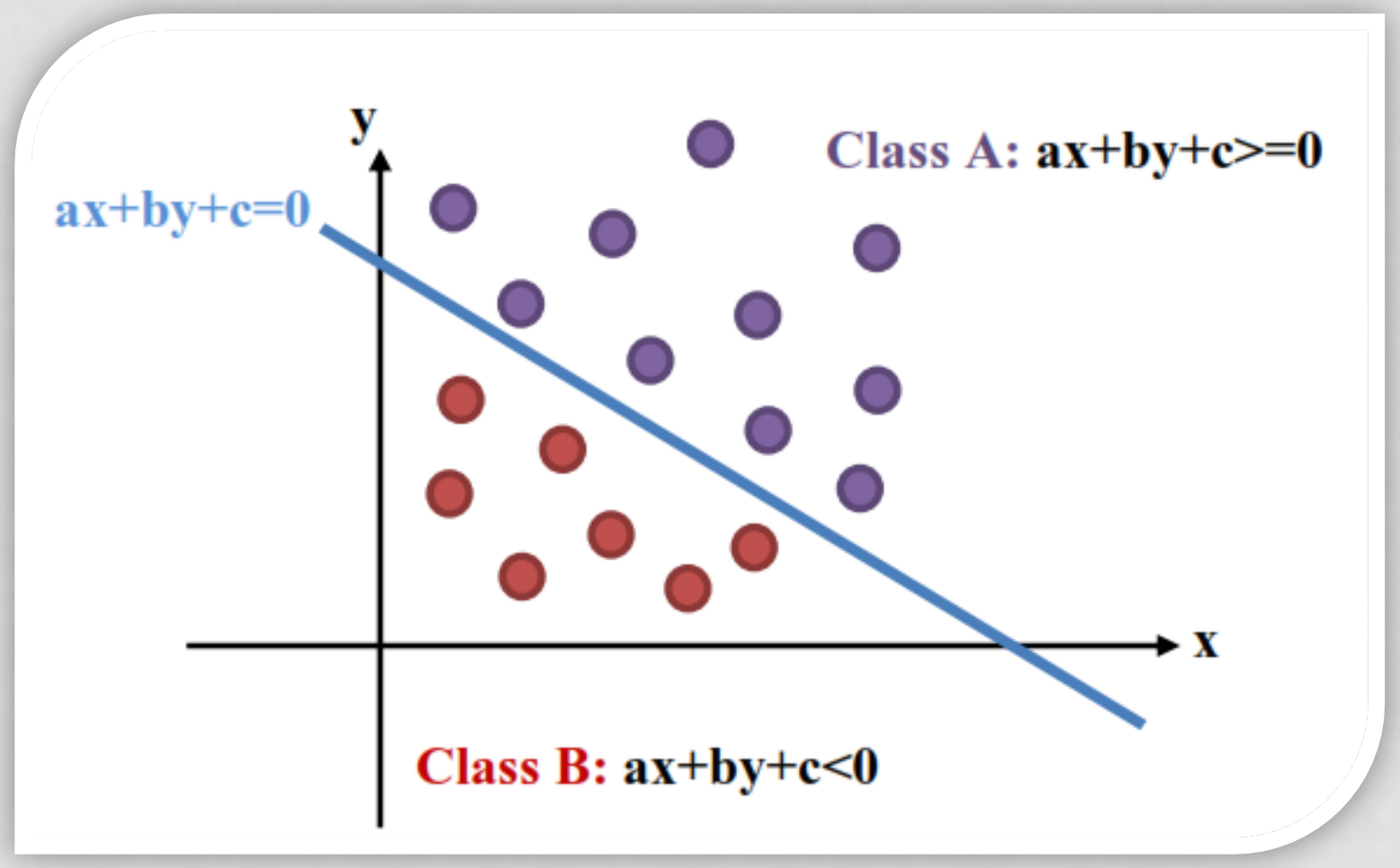
بسم الله الرحمن الرحیم

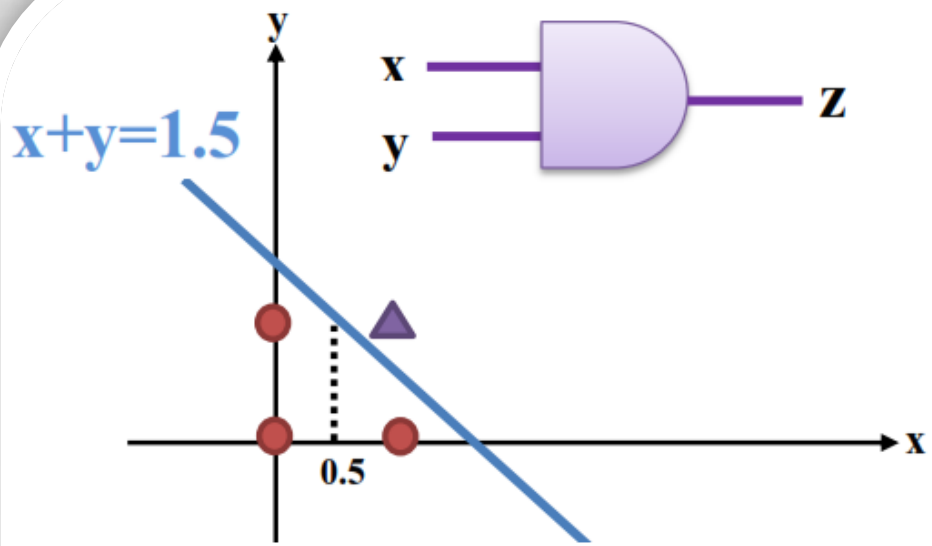
فهرست

- دسته بندی و جداسازی خطی توسط شبکه عصبی
- فرآیند یادگیری
- انواع یادگیری ماشین
  - یادگیری با نظارت
  - یادگیری بدون نظارت
  - یادگیری تقویتی
- ارزیابی یادگیری
- شبکه هب (یادگیری با نظارت)

دسته بندی ساده



ترکیب عطفی AND



x	y	x AND y = z
0	0	0
0	1	0
1	0	0
1	1	1

$$Z = \begin{cases} 0 & x + y \leq 1.5 \\ 1 & x + y > 1.5 \end{cases}$$

دسته بندی ساده با شبکه عصبی

- الگوی ورودی عضو دسته مورد نظر باشد، پاسخ «بله» و اگر ورودی عضو آن دسته نباشد، پاسخ «خیر»

$$y_in = b + \sum_i x_i w_i$$

- تابع فعال‌سازی پله‌ای

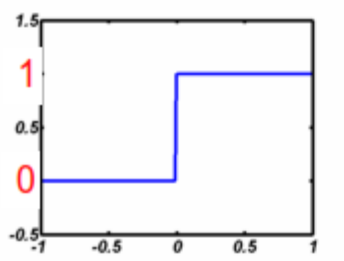
$$f(y_in) = \begin{cases} 1 & if \quad y_in \geq 0 \\ -1 & if \quad y_in < 0 \end{cases}$$

- مرز تصمیم‌گیری (Decision Boundary) = مرز بین ناحیه‌ای که در آن y_in>0 و ناحیه‌ای که در آن y_in<0 است

$$b + \sum_i x_i w_i = 0$$

  - پاسخ این معادله یک خط، یک صفحه و یا یک ابرصفحه است

تبیین عملکرد یک نورون در فضای m بعدی ورودی



تابع غیر خطی: سخت (پله‌ای)

$$y = \begin{cases} 1 & Y \geq 0 \\ 0 & Y < 0 \end{cases}$$

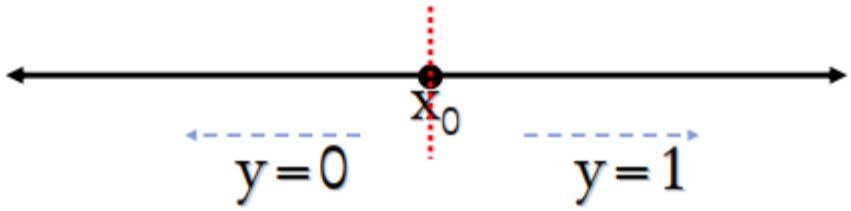
تابع فعالیت نورون: غیرخطی سخت

m = 1

مرز تصمیم یک **نقطه** است.

$$y = f(w_0 x_0 - \theta)$$

مرز تصمیم $w_0 x_0 - \theta = 0 \Rightarrow x_0 = \theta / w_0$



تبیین عملکرد یک نورون در فضای m بعدی ورودی

تابع فعالیت نورون: غیرخطی سخت

m = 2

مرز تصمیم یک خط است که فضای ورودی را به دو طبقه تقسیم

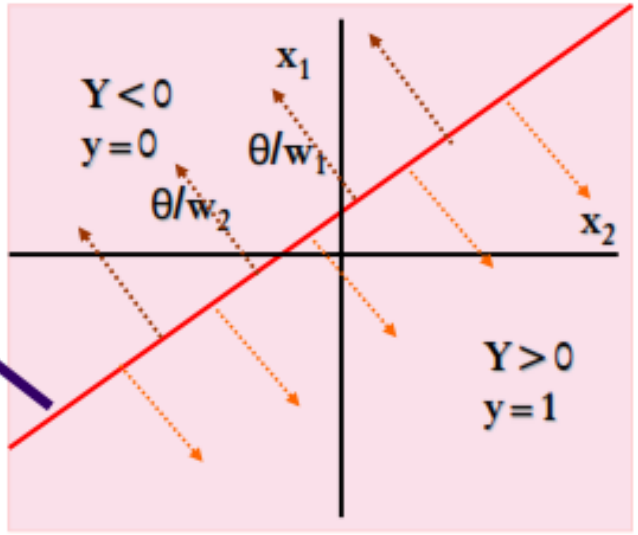
$$y = f(w_1x_1 + w_2x_2 - \theta)$$

$$w_1x_1 + w_2x_2 - \theta = 0$$ مرز تصمیم

مثلاً :

$$Y = x_1 + x_2 - 1 = 0$$

$$w_1=1;\quad w_2=1;\quad \theta = 1$$

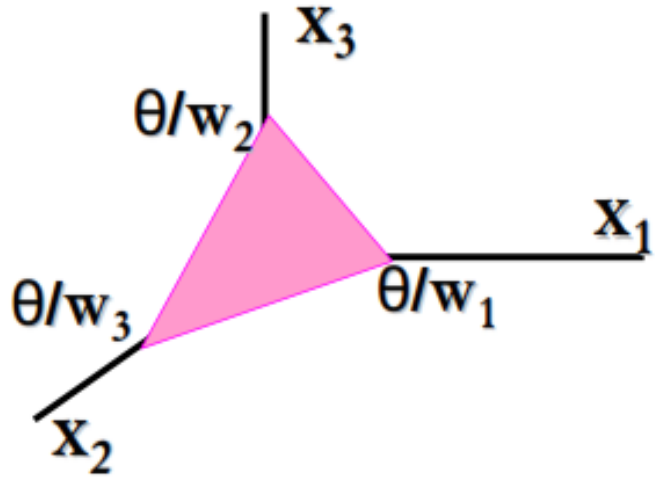


تبیین عملکرد یک نورون در فضای m بعدی ورودی

تابع فعالیت نورون: غیرخطی سخت

m = 3

مرز تصمیم یک صفحه است.

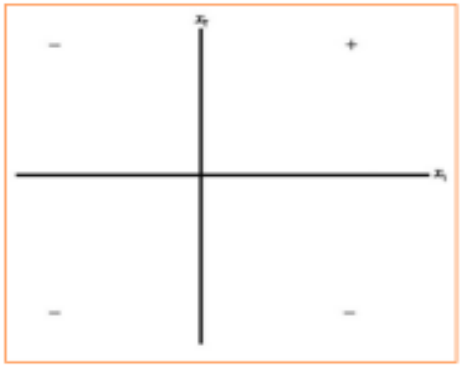


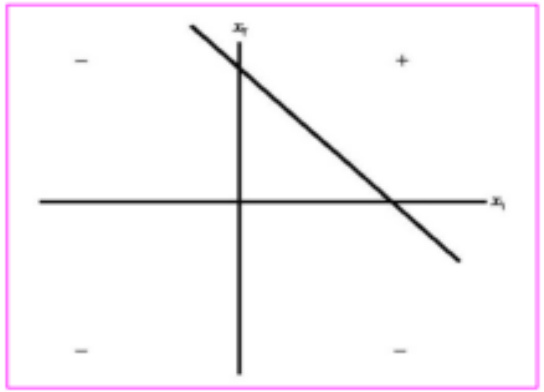
جدا سازی خطی

- **مسئله خطی تفکیک‌پذیر (Linearly Separable)**
  - حل یک مسئله توسط شبکه‌های یک لایه پس از تنظیم وزن‌ها (و بایاس)
  - تمام بردارهای ورودی آموزش که پاسخ صحیح برای آنها ۱+ (عضویت در دسته) است، در یک طرف مرز تصمیم‌گیری و تمام برادرهای ورودی آموزش که پاسخ صحیح برای آنها ۱- (عدم عضویت در دسته) است در سمت دیگر مرز تصمیم‌گیری قرار می‌گیرند
  - نشان داده شده است که شبکهٔ یک لایه فقط می‌تواند مسائلی را حل کنند که به صورت خطی تفکیک‌پذیر باشند
  - شبکه‌های چندلایه‌ای که از توابع فعال‌سازی خطی استفاده می‌کنند، از شبکه‌های یک لایه قوی‌تر نیستند زیرا ترکیب چند تابع خطی نیز خطی است

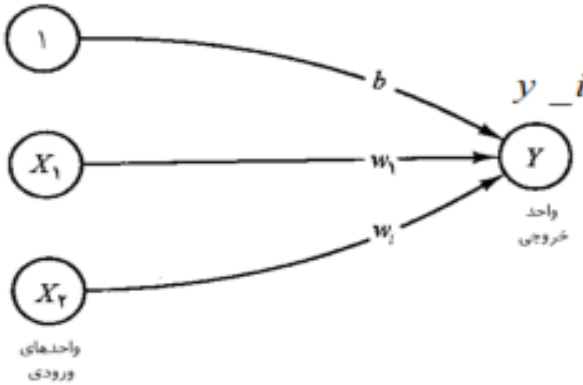
جدا سازی خطی

INPUT(x_1, x_2)	OUTPUT
(1 , 1)	+1
(1 , −1)	−1
(−1 , 1)	−1
(−1 , −1)	−1





- مثال: تابع AND

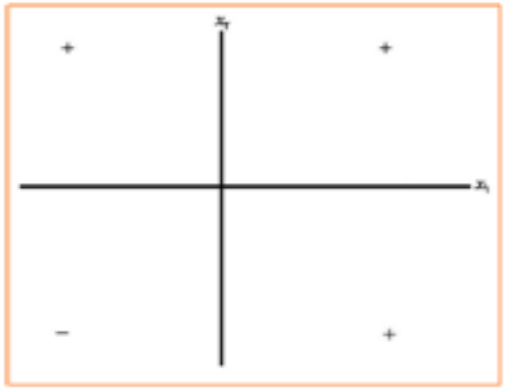


$$y_in = 1 \times b + w_1 x_1 + w_2 x_2$$

- مرز تصمیم‌گیری $b + w_1 x_1 + w_2 x_2 = 0$
- پاسخ $b = -1,\ w_1 = 1, w_2 = 1$

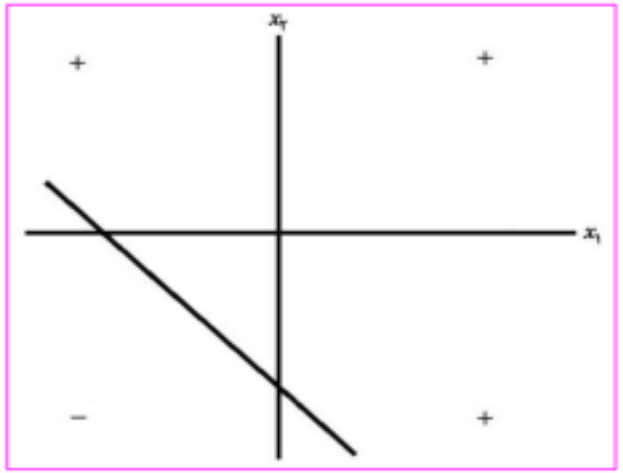
$$x_2 = -x_1 + 1$$

جدا سازی خطی



- مثال: تابع OR

INPUT(x_1, x_2)	OUTPUT
(1, 1)	+1
(1, −1)	+1
(−1, 1)	+1
(−1, −1)	−1



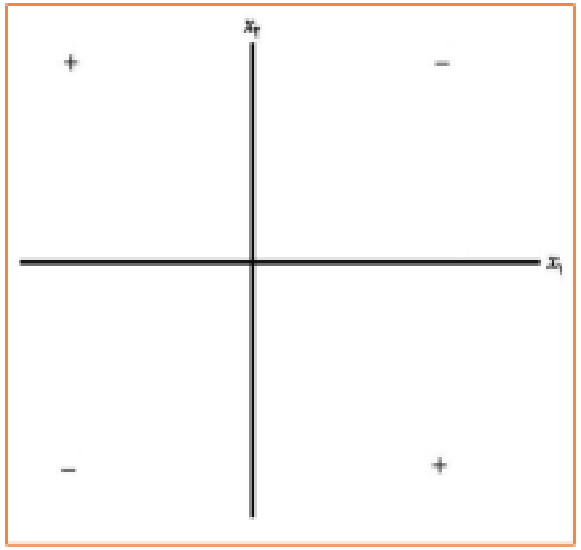
- مرز تصمیم‌گیری

$$b = 1,\ w_1 = 1, w_2 = 1$$

$$x_2 = -x_1 - 1$$

- اگر وزن بایاس وجود نداشت، مرز تصمیم‌گیری باید از مبدأ عبور می‌کرد

جدا سازی خطی



- **مثال: تابع XOR**

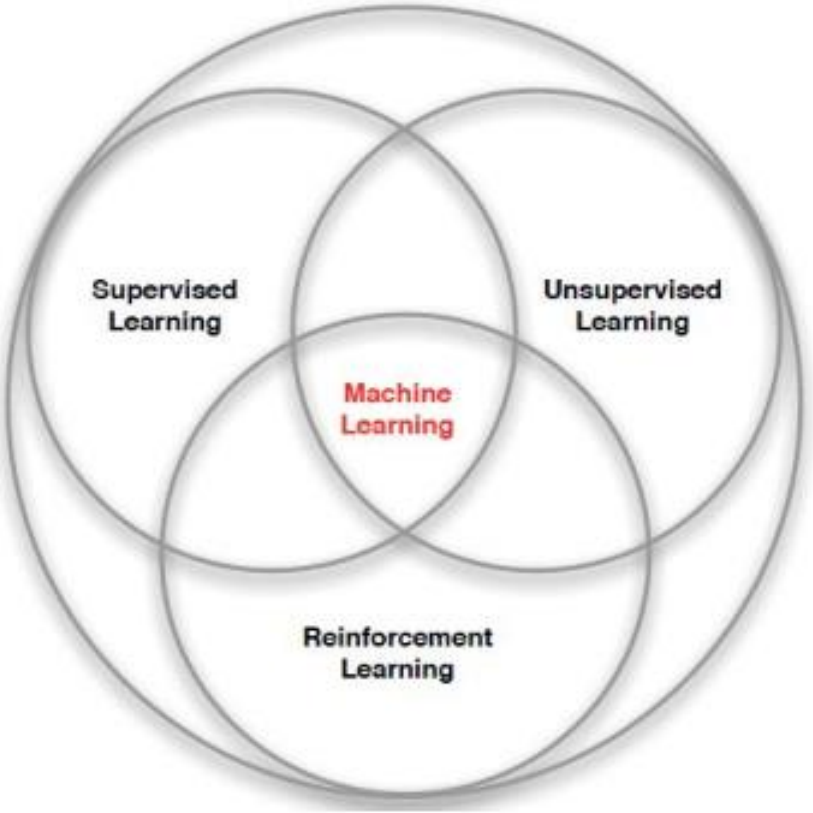
INPUT(x_1, x_2)	OUTPUT
(1 , 1)	−1
(1 , −1)	+1
(−1 , 1)	+1
(−1 , −1)	−1

- حل؟
- فضای داده‌های ورودی به صورت خطی جدایی‌پذیر نیست.
- هیچ خط مستقیم نمی‌تواند نقاط مثبت و منفی را جدا کند

فرآیند یادگیری

- سیستم یادگیرنده توسط محیط تحریک شود.
- قانون یادگیری با رجوع به نتیجه تحریک، پارامترهای سیستم یادگیری را تغییر دهد.
- سیستم یادگیرنده به خاطر تغییراتی که در ساختار داخلی آن اتفاق افتاده است، پاسخ مناسبتری به محیط بدهد.

انواع یادگیری ماشین



- یادگیری ماشین
  - یادگیری با نظارت
  - یادگیری بدون نظارت
  - یادگیری تقویتی

یادگیری با نظارت
SUPERVISED LEARNING

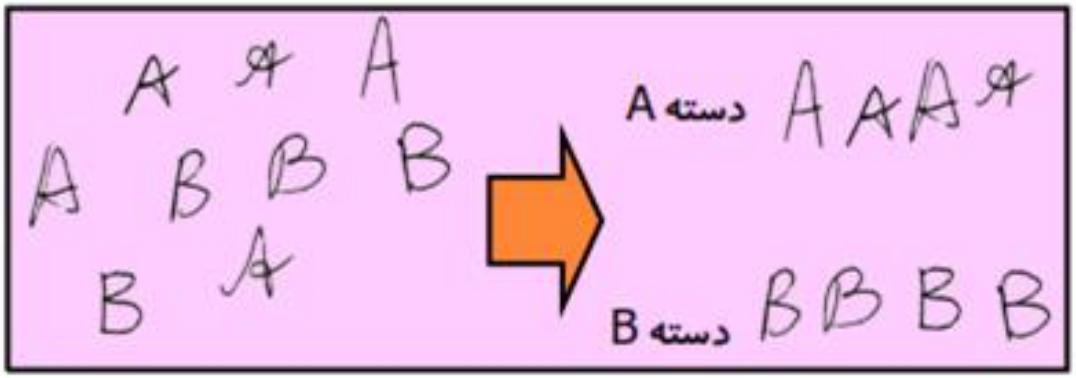
- در این حالت در هر تکرار الگوریتم یادگیری جواب مطلوب سیستم یادگیرنده از قبل آماده است.
  - به عبارت دیگر الگوریتم به جواب مطلوب دسترسی دارد.

- مثلا اگر هدف یادگیری تابع $y=x^2$ باشد، پاسخ مطلوب به ورودی ۰.۵ برابر ۰.۲۵ می باشد.
- به طور کل جوابی را که سیستم یادگیر با وضعیت فعلی پارامترهایش می دهد، جواب واقعی در نظر می گیریم.

بنابر این الگوریتم هم به جواب مطلوب و هم به جواب واقعی دسترسی دارد. یعنی خطای یادگیری که تفاوت این دو مقدار می باشد، در دسترس است.

یادگیری با نظارت
SUPERVISED LEARNING

- یادگیری با نظارت (Supervised)
  - داده‌های آموزش دارای برچسپ هستند. برچسپ‌ها بیانگر دسته هر داده هستند.
  - دسته‌بندی (طبقه‌بندی) (Classification)
    - نوعی از بازشناسی برای دسته‌بندی الگو
    - دسته‌ها از قبل مشخص



کاربردهایی از یادگیری تحت نظارت

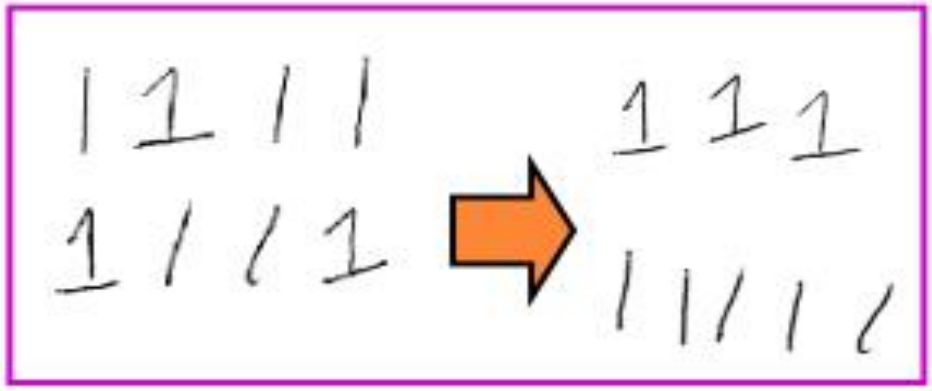
- تحلیل‌ها و پیش‌بینی‌های مالی و تجاری
  - پیش‌بینی بورس
  - پیش‌بینی قیمت طلا
- پزشکی
  - تشخیص بیماری‌ها از روی تصاویر و سیگنال‌ها

یادگیری بدون نظارت
UNSUPERVISED LEARNING

- در این حالت جواب مطلوب برای سیستم یادگیرنده موجود نیست.
- یعنی : عدم دسترسی به خطای یادگیری
- سوال
  - چگونه می توان یک سیستم یادگیرنده را آموزش داد، اگر ندانیم که این سیستم قرار است چه کاری انجام دهد؟
- خواهیم دید که بیشتر الگوریتمهای بدون ناظر عمل خوشه بندی را انجام می دهند.
  - یعنی می آموزند که الگوهای ورودی را به تعداد متناهی از گروهها تقسیم کنند.

یادگیری بدون نظارت
UNSUPERVISED LEARNING

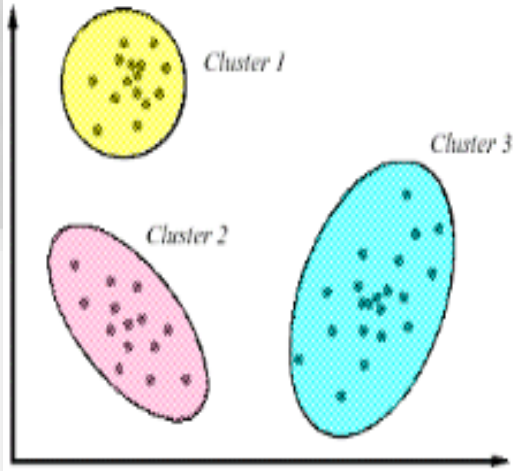
- خوشه‌بندی (Clustering)
  - دسته‌بندی الگوها بر اساس شباهت
  - دسته‌ها از قبل نامشخص



کاربردهایی از یادگیری بدون نظارت

- دسته‌بندی خودکار داده‌های حجیم
  - متون وب‌سایت
  - رفتار مشتری‌ها برای بازاریابی هدف‌مند
- داده‌کاوی و استخراج دانش از داده‌ها
- کاهش ابعاد ویژگی‌ها
- قطعه‌بندی تصویر (Segmentation)

یادگیری بدون نظارت

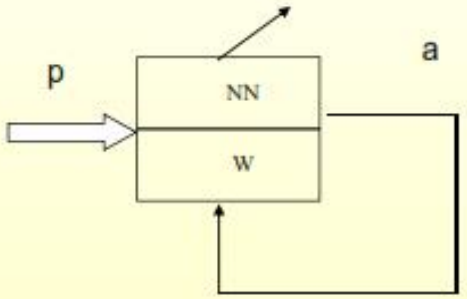


- روش‌ها
  - شبکه عصبی
    - نگاشت‌های خودسازمانده کوهونن (SOM)
    - نظریهٔ نوسان وفقی (ART)
  - K میانگین (K-means) و حالت فازی آن (FCM)
  - روش‌های سلسله مراتبی

یادگیری بدون نظارت

یادگیری بدون ناظر در شبکه عصبی

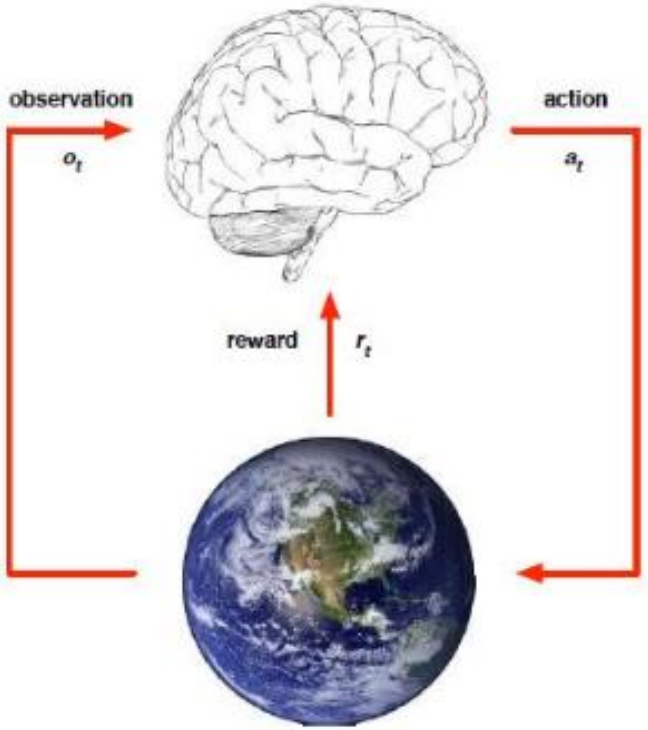
- یادگیری خود سازمانده
- در این نوع یادگیری، هیچ سیگنالی که اطلاعاتی در مورد مطلوبیت جواب شبکه به خود شبکه وارد نماید موجود نیست.
- شبکه در اینجا بایستی تنها با ارائهٔ صرف بردارهای ورودی، ارتباطات موجود بین الگوهای ورودی را پیدا کرده و در خروجی شبکه کُد نماید.



یادگیری تقویتی
REINFORCEMENT LEARNING

- یادگیری بر مبنای آزمون و خطا در تعامل با محیط
- انجام یک عمل (action) و دریافت بازخورد از محیط
- یادگیری بر اساس پاداش (Reward) و جریمه (Penalty)
- عدم وجود ناظر (supervisor)
- یادگیری در طول زمان

مراحل یک یادگیری تقویتی



- At each step t the agent:
  - Executes action a_t
  - Receives observation o_t
  - Receives scalar reward r_t
- The environment:
  - Receives action a_t
  - Emits observation o_t
  - Emits scalar reward r_t

ارزیابی سیستم های یادگیری

- دقت/نرخ خطا
  - درصد بازشناسی درست/نادرست دسته‌ها
  - نرخ قبول اشتباه/رد اشتباه
- سرعت پردازش
- هزینه طراحی و ساخت
- مقاوم‌بودن (Robustness)

شبکه هب (یادگیری با نظارت)

- اولین (و ساده‌ترین) قانون یادگیری برای شبکهٔ عصبی
- ایده اصلی یادگیری هب
  - یادگیری با تغییر استحکامات سیناپس‌های نرون‌ها (وزن‌های شبکه‌های عصبی) است
  - اگر دو نرون متصل به هم به طور هم‌زمان «فعال» باشند، وزن بین آنها باید افزایش یابد
  - هب دربارهٔ نرون‌هایی که به طور هم‌زمان برانگیخته نمی‌شوند، چیزی نمی‌گوید
    - یادگیری قوی‌تر = اگر دو نرون به طور هم‌زمان «غیرفعال» باشند، وزن‌ها افزایش یابد

- شبکهٔ هب یک لایه است
- به‌روز شدن (Update) وزن‌ها

$$w_i(new) = w_i(old) + x_i y$$

خروجی

وزن جدید

وزن قبلی

ورودی

- برای داده دودویی، اگر ورودی یا خروجی (یا هر دو) «غیرفعال» باشند، یادگیری صورت نمی‌گیرد

الگوریتم شبکه هب

- مرحله ۰- به تمام وزن‌ها مقدار اولیه صفر بدهید $w_i = 0 \quad (i = 1,\ldots,n)$
- مرحله ۱- برای هر بردار آموزش ورودی و خروجی هدف، s : t، مراحل ۲ تا ۴ را انجام بده
- مرحله ۲- فعال‌سازی‌های واحدهای ورودی را تعیین کن $x_i = s_i \quad (i = 1,\ldots,n)$
- مرحله ۳- برای واحد خروجی فعال‌سازی را تعیین کن $y = t$
- مرحله ۴- وزن‌ها و بایاس را به‌روز کن

$$w_i(new) = w_i(old) + x_i y \quad (i = 1,\ldots,n)$$

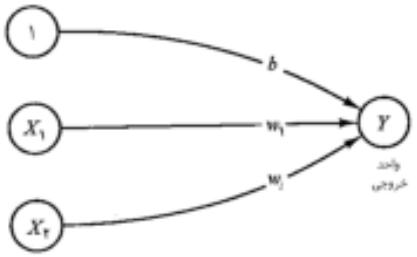
$$b(new) = b(old) + y$$

$$\mathbf{w}(new) = \mathbf{w}(old) + \mathbf{x}.y \quad \Rightarrow \quad \mathbf{w}(new) = \mathbf{w}(old) + \Delta\mathbf{w} \qquad \Delta w = x.y$$

داده‌های آموزشی فقط یک بار به شبکه نشان داده شده و آموزش به اتمام می‌رسد

کاربرد شبکه هب (تابع AND)

تابع AND با ورودی‌ها و هدف‌های دودویی . . .



INPUT			TARGET
$(x_1$	x_2	1)	
(1	1	1)	1
(1	0	1)	0
(0	1	1)	0
(0	0	1)	0

- تغییر وزن

$$\Delta w_1 = x_1 t, \quad \Delta w_2 = x_2 t, \quad \Delta b = 1.t = t$$

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

کاربرد شبکه هب (تابع AND)

- برای ورودی اول

$$x_1 = 1, \quad x_2 = 1, \quad b = 1, \quad t = 1$$

INPUT			TARGET	WEIGHT CHANGES			WEIGHTS		
x_1	x_2	1	t	Δw_1	Δw_2	Δb	w_1	w_2	b
							0	0	0
1	1	1	1	1	1	1	1	1	1

مقدار اولیه

$$x_2 = -x_1 - 1$$

کاربرد شبکه هب (تابع AND)

- تابع AND با ورودی‌ها و هدف‌های دودویی
  - برای دومین، سومین و چهارمین ورودی

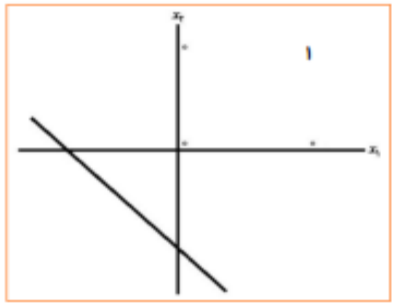
INPUT			TARGET	WEIGHT CHANGES			WEIGHTS		
x_1	x_2	1	t	Δw_1	Δw_2	b	w_1	w_2	b
1	0	1	0	0	0	0	1	1	1
0	1	1	0	0	0	0	1	1	1
0	0	1	0	0	0	0	1	1	1

یادگیری رخ نمی‌دهد
وزن‌ها تغییر نمی‌کند

الگوهایی با مقدار هدف صفر یا «غیرفعال»

استفاده از نمایش دودویی

پاسخ نادرست



$$x_2 = -x_1 - 1$$

کاربرد شبکه هب (تابع AND)

- تابع AND با ورودی‌های دودویی و مقادیر هدف دوقطبی

INPUT $(x_1\ x_2\ 1)$	TARGET t
(1 1 1)	1
(1 0 1)	−1
(0 1 1)	−1
(0 0 1)	−1

- اولین ورودی

INPUT $(x_1\ x_2\ 1)$	TARGET t	WEIGHT CHANGES $(\Delta w_1\ \Delta w_2\ \Delta b)$	WEIGHTS $(w_1\ w_2\ b)$
			(0 0 0)
(1 1 1)	1	(1 1 1)	(1 1 1)

$$x_2 = -x_1 - 1$$

- ارائه دومین، سومین و چهارمین

پاسخ نادرست

INPUT $(x_1\ x_2\ 1)$	TARGET t	WEIGHT CHANGES $(\Delta w_1\ \Delta w_2\ b)$	WEIGHTS $(w_1\ w_2\ b)$
(1 0 1)	−1	(−1 0 −1)	(0 1 0)
(0 1 1)	−1	(0 −1 −1)	(0 0 −1)
(0 0 1)	−1	(0 0 −1)	(0 0 −2)

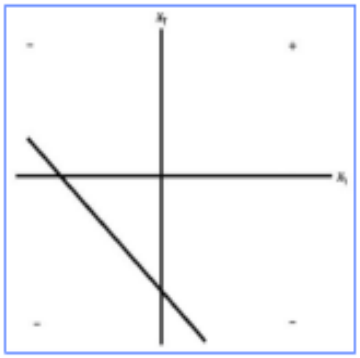
کاربرد شبکه هب (تابع AND)

INPUT			TARGET
x_1	x_2	1	t
1	1	1	1
1	−1	1	−1
−1	1	1	−1
−1	−1	1	−1

- تابع AND برای ورودی‌ها و هدف‌های دوقطبی . . .
  - اولین ورودی

INPUT			TARGET	WEIGHT CHANGES			WEIGHTS		
x_1	x_2	1	t	Δw_1	Δw_2	Δb	w_1	w_2	b
							0	0	0
1	1	1	1	1	1	1	1	1	1

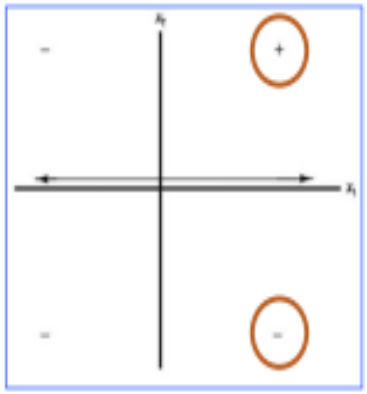
$$x_2 = -x_1 - 1$$



کاربرد شبکه هب (تابع AND)

INPUT			TARGET	WEIGHT CHANGES			WEIGHTS		
x_1	x_2	1	t	Δw_1	Δw_2	Δb	w_1	w_2	b
							1	1	1
1	−1	1	−1	−1	1	−1	0	2	0

$x_2 = 0$



- دومین ورودی

پاسخ درست برای
دو نمونه آموزش

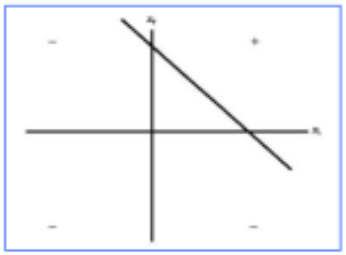
کاربرد شبکه هب (تابع AND)

- تابع AND برای ورودی‌ها و هدف‌های دوقطبی

- سومین ورودی

INPUT			TARGET	WEIGHT CHANGES			WEIGHTS		
x_1	x_2	1	t	Δw_1	Δw_2	Δb	w_1	w_2	b
							0	2	0
−1	1	1	−1	1	−1	−1	1	1	−1

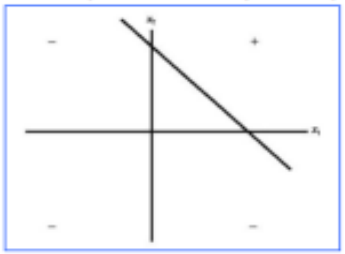
$$x_2 = -x_1 + 1$$



- چهارمین ورودی

INPUT			TARGET	WEIGHT CHANGES			WEIGHTS		
x_1	x_2	1	t	Δw_1	Δw_2	Δb	w_1	w_2	b
							1	1	−1
−1	−1	1	−1	1	1	−1	2	2	−2

$$x_2 = -x_1 + 1$$



پاسخ درست

نمایش داده ها در شبکه هب

- شکل نمایش داده‌ها می‌تواند مسئله قابل‌حل را به مسئله‌ای غیرقابل‌حل تبدیل کند
  - در قانون هب بسیار موثر است
    - برای برخی الگوها منجر به جواب درست نمی‌شود، ممکن است برای نمایش متفاوتی از همان الگوها پاسخ درستی را نتیجه دهد
- نمایش دوقطبی بهتر از نمایش دودویی است
  - افزایش قابلیت تعمیم شبکه
  - امکان تمایز داده‌های گم‌شده (Missing Data) از داده‌های اشتباه (Mistaken Data)
    - مقادیر گم‌شده = «۰»
    - اشتباهات = قرینه مقدار ورودی از ۱+ به ۱- و یا برعکس