

شبکه عصبی مصنوعی

MLP و RBF

در نرم افزار MATLAB

پایه چهارم



مدرس دوره

قربان علیزاده

دانشجوی دکتری زلزله



شبکه عصبی مصنوعی در نرم افزار متلب

هر نیاز دلیل اختراع یک ابزار



۱- نیاز فیزیکی

۲- نیاز عاطفی و اجتماعی

۳- نیاز به دقت و سرعت

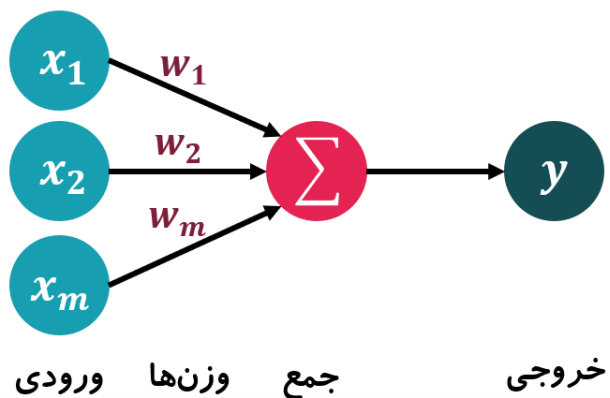
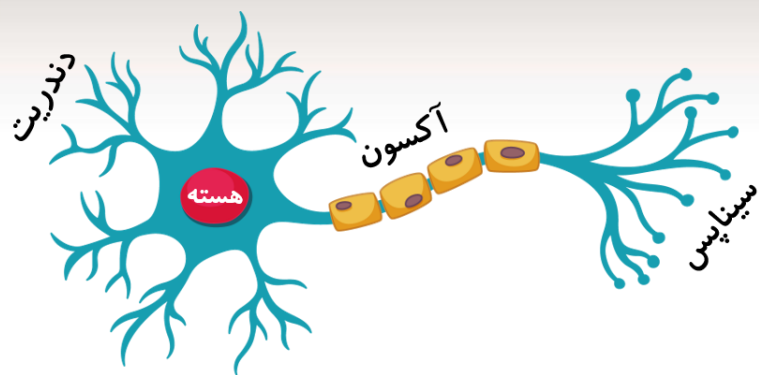
۴- نیاز به تصمیم گیری سریع





شبکه عصبی مصنوعی در نرم افزار متلب

طبیعت بهترین الگو





مقایسه مغز و کامپیوتر

- سرعت پاسخگویی هر نورون ۱ میلی ثانیه

- سرعت پاسخگویی یک ترانزیستور سیلیکونی ۱ نانو ثانیه

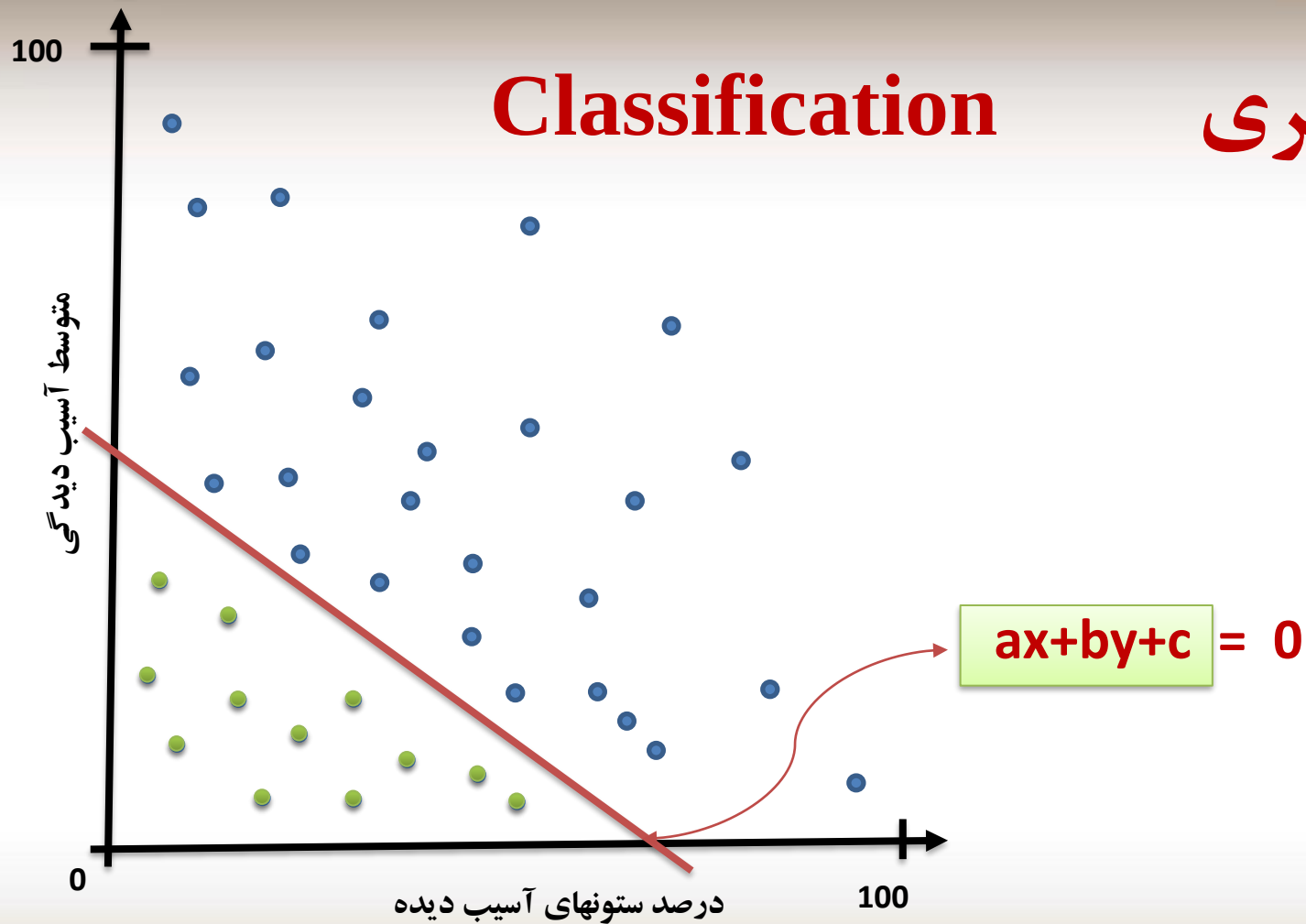
اما عملکرد کلی مغز بسیار سریعتر از کامپیوتر است. چرا؟



Classification

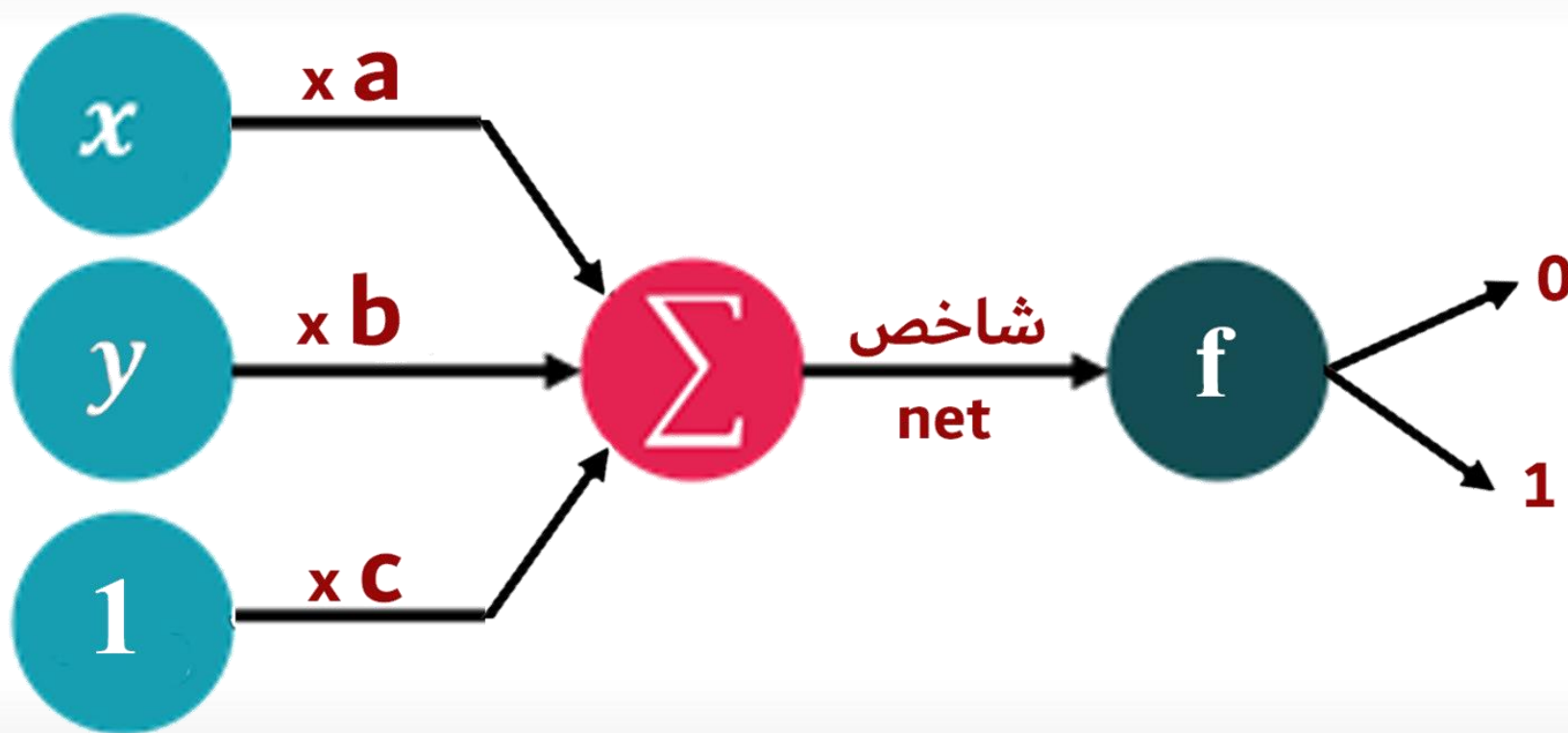
طبقه بندی مهمترین مهارت بشری

- سرعت پاسخگویی هر نورون ۱ میلی ثانیه





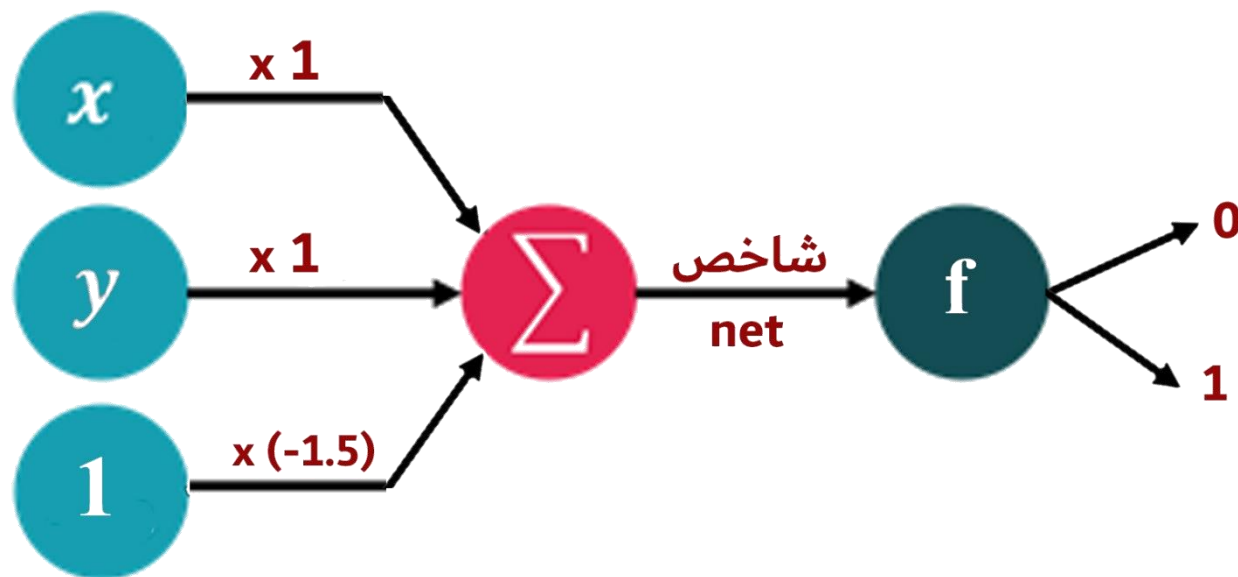
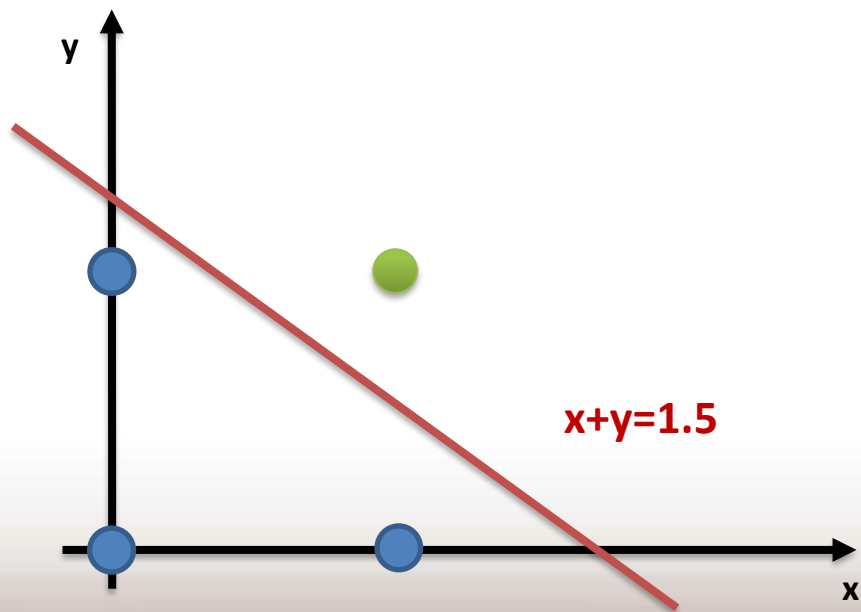
مدل ریاضی نورون





تابع عطفی AND

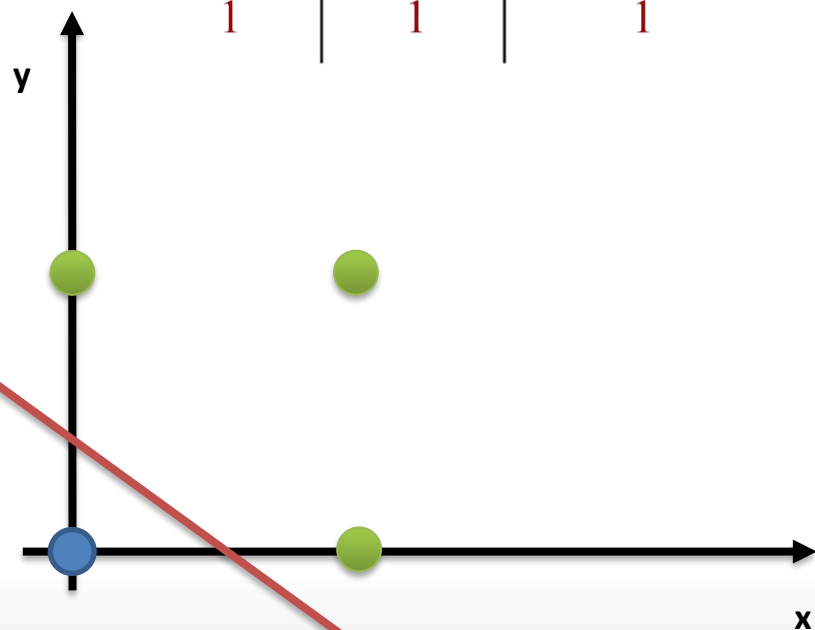
x	y	$x \text{ AND } y = z$
0	0	0
0	1	0
1	0	0
1	1	1



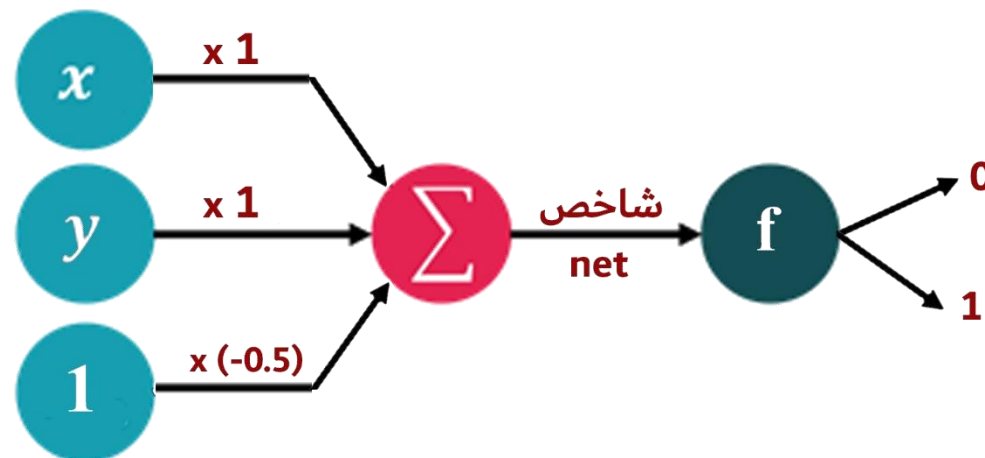


تابع فصلی OR

x	y	$x \text{ OR } y = z$
0	0	0
0	1	1
1	0	1
1	1	1

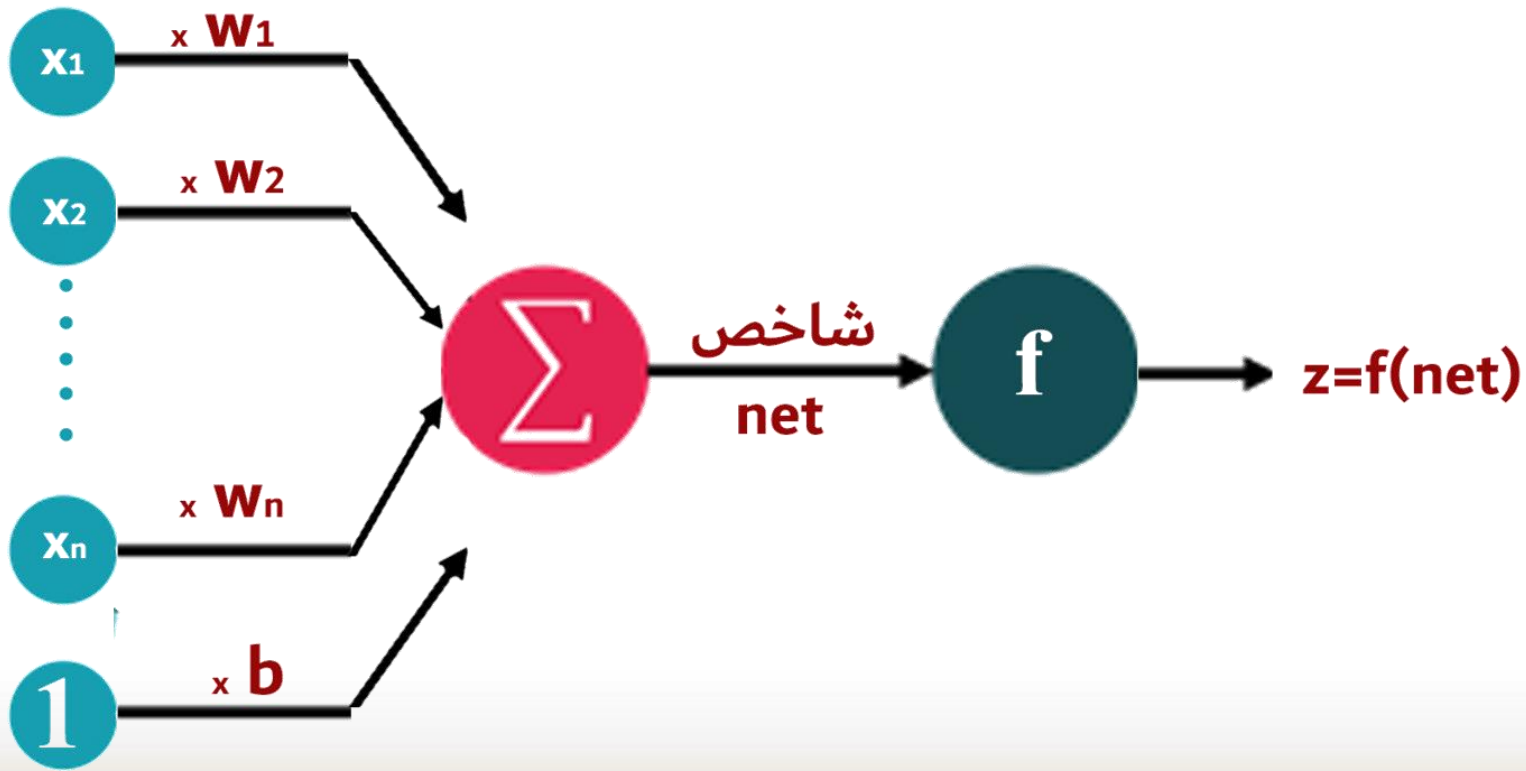


$$x+y=0.5$$





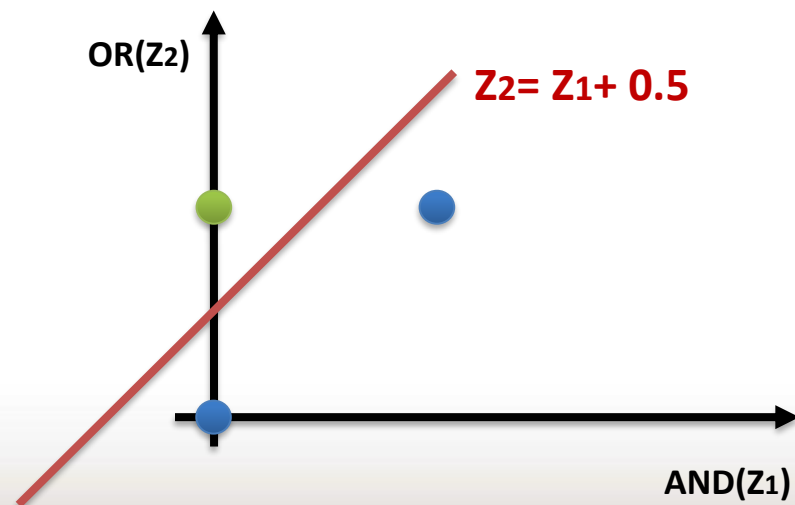
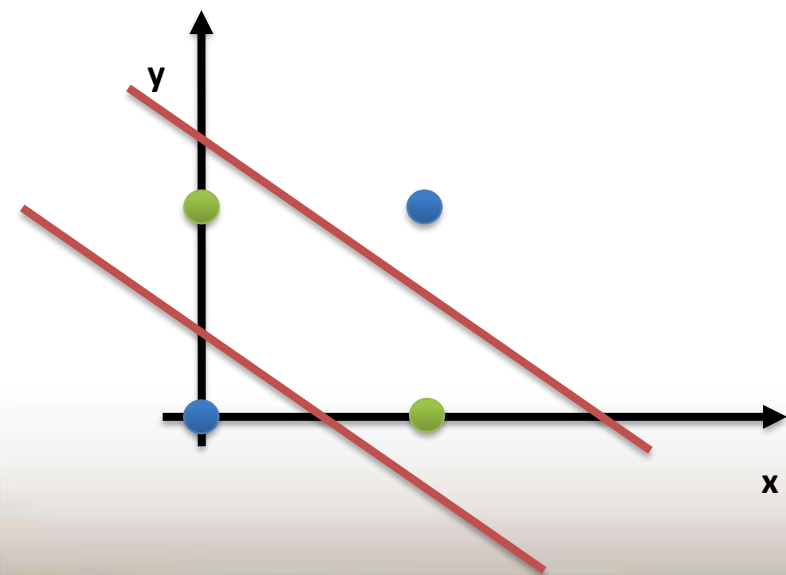
مدل ریاضی نورون در حالت کلی

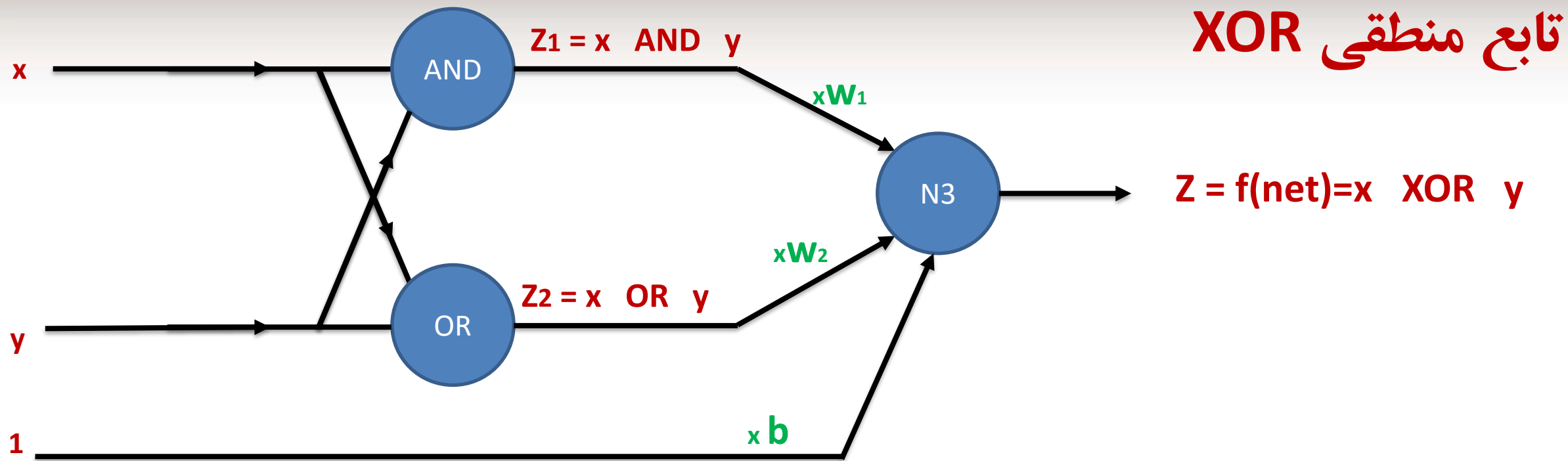




تابع منطقی XOR

x	y	$x \text{ XOR } y = z$
0	0	0
0	1	1
1	0	1
1	1	0



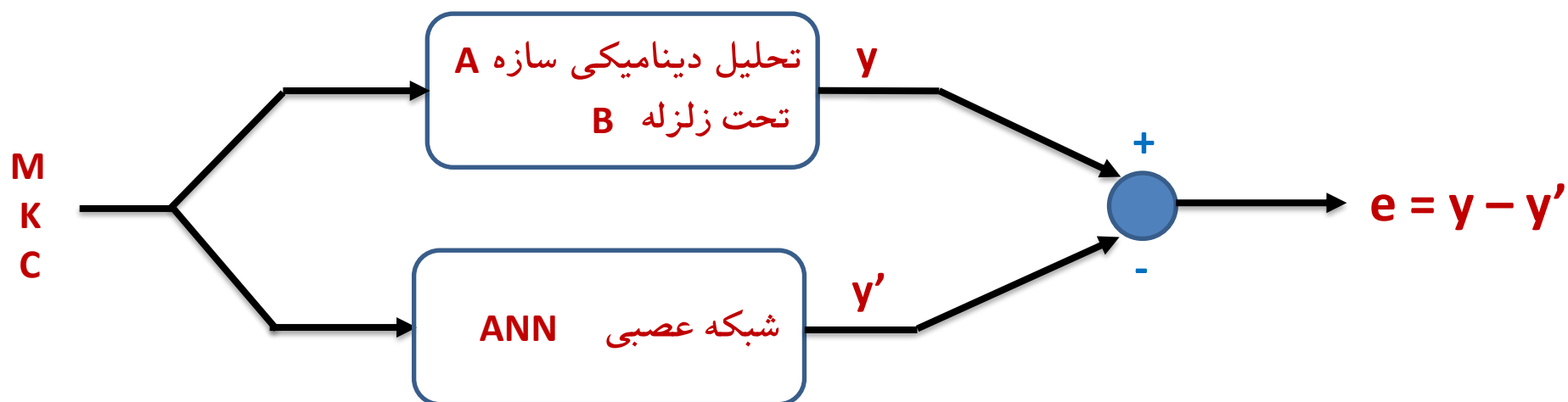


$$\vec{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \quad \vec{w} = \begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix} \quad \longrightarrow \quad \text{net} = [w_1 \ w_2 \ \dots \ w_n] \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} + b = w^T \cdot x + b \quad \longrightarrow \quad Z = f(\text{net}) = f(w^T \cdot x + b)$$



Training

آموزش شبکه عصبی



$$e_i = y_i - y'_i \longrightarrow \min \frac{1}{N} \sum_{i=1}^N e_i^2 \quad \text{MSE} \quad \text{میانگین مربعات خطا}$$



کاربردهای شبکه عصبی

۱- مدل سازی و تقریب توابع

۲- طبقه بندی و خوشه بندی

۳- استخراج ویژگی

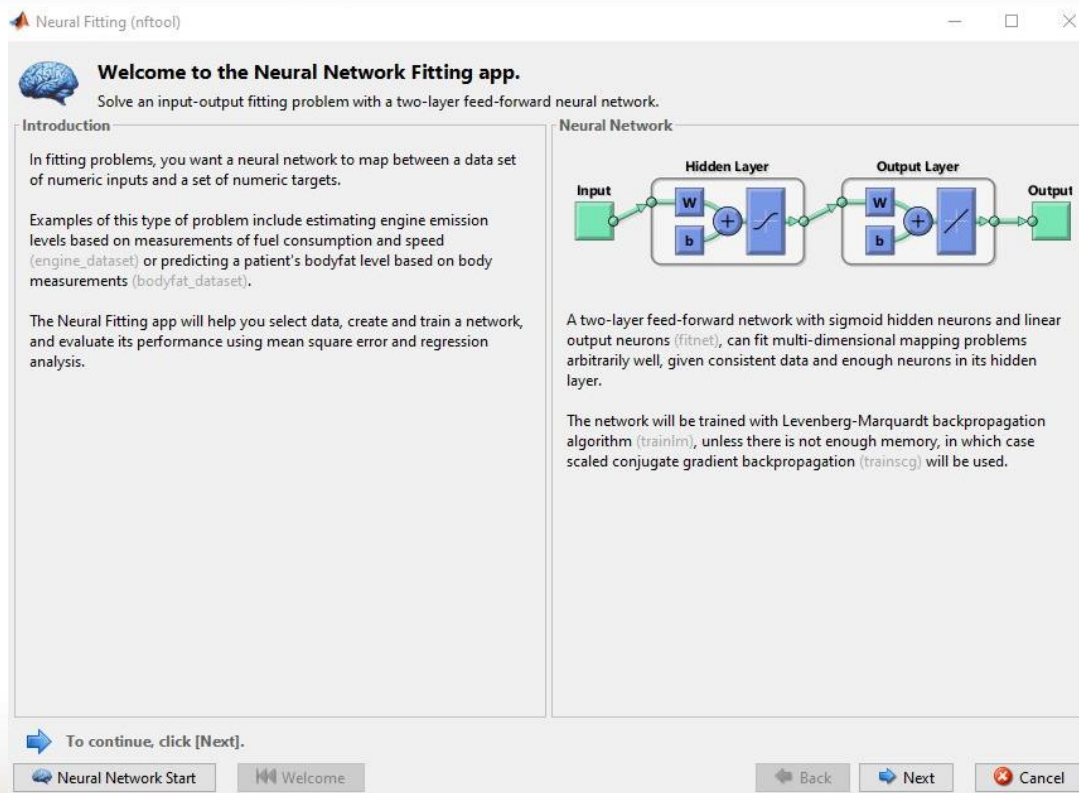
۴- حافظه های انجمنی

۵- فشرده سازی

۶- کنترل و شناسایی سیستم

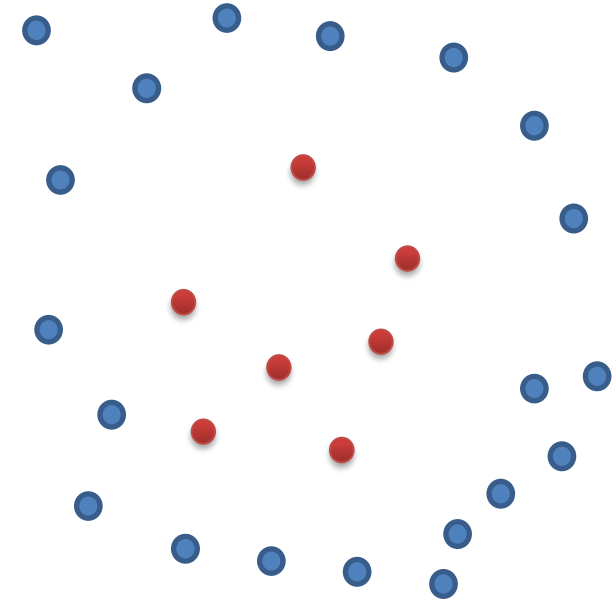
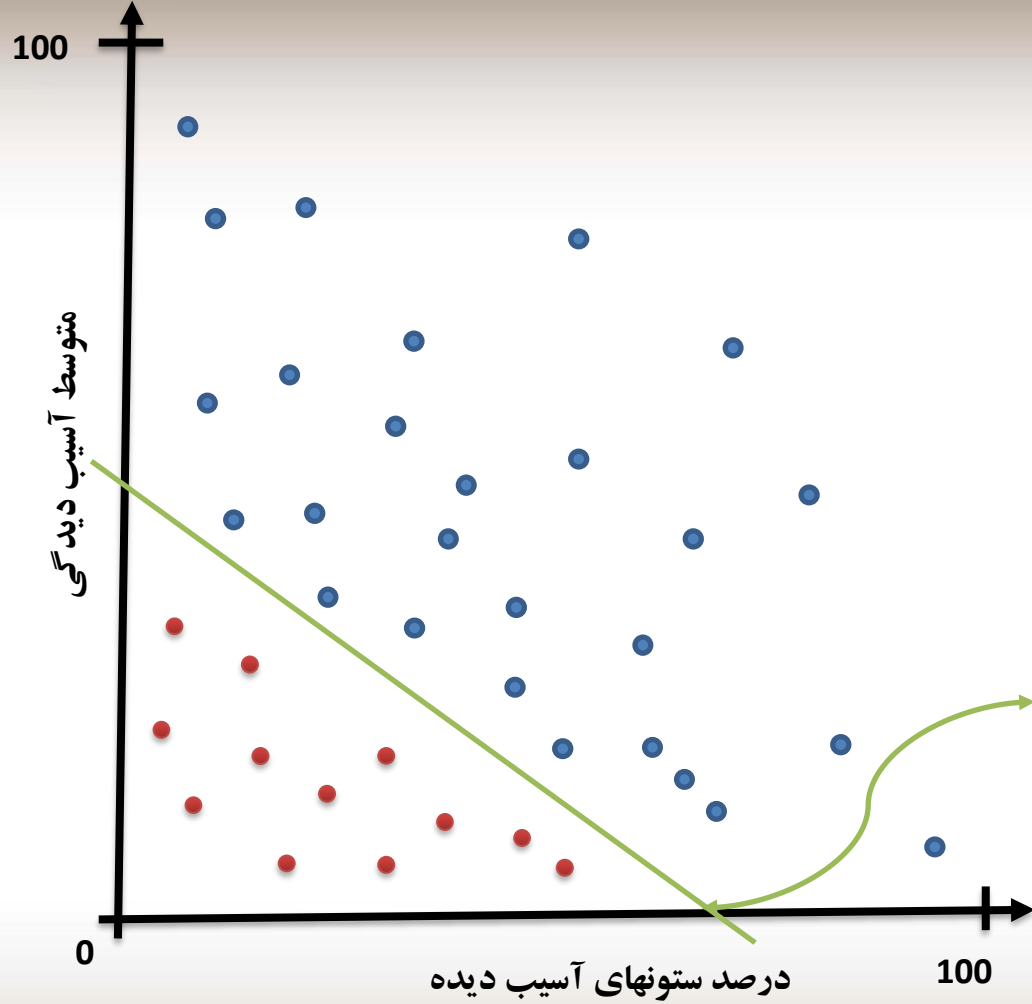


پیاده سازی شبکه عصبی MLP در تولباکس نرم افزار متلب





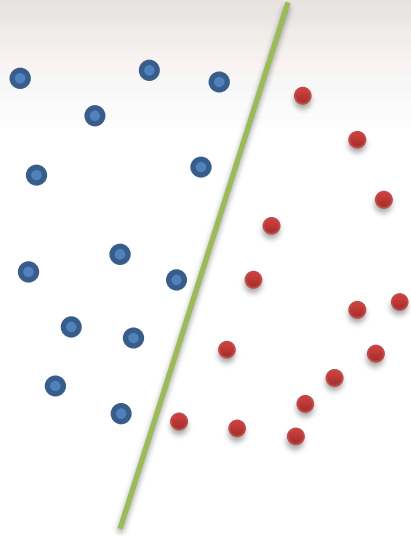
شبکه عصبی مصنوعی در نرم افزار متلب - مبانی شبکه های عصبی RBF



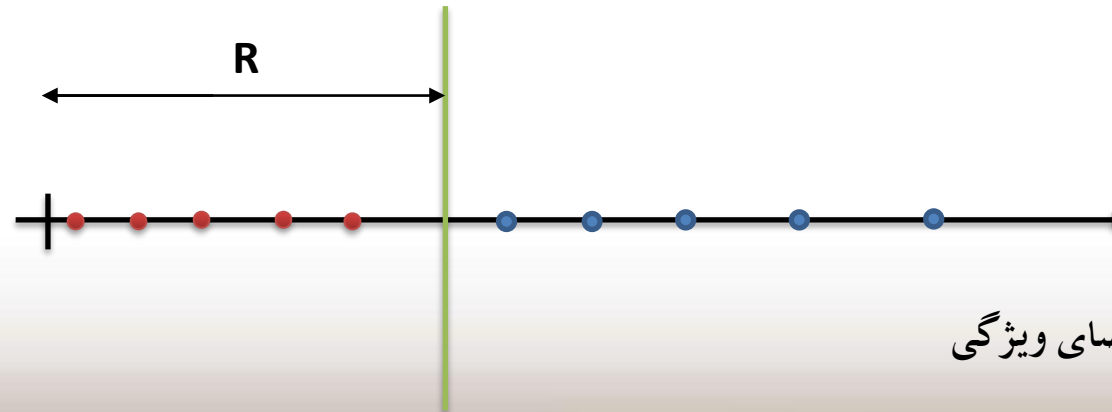
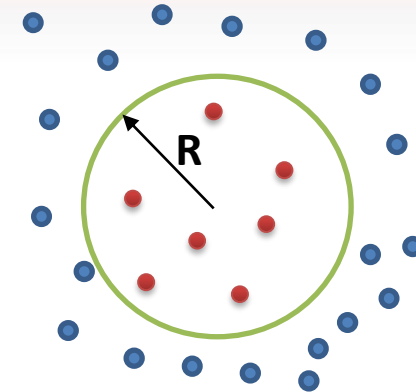


شبکه عصبی مصنوعی در نرم افزار متلب - مبانی شبکه های عصبی RBF

قابل تفکیک بصورت خطی در فضای ورودی

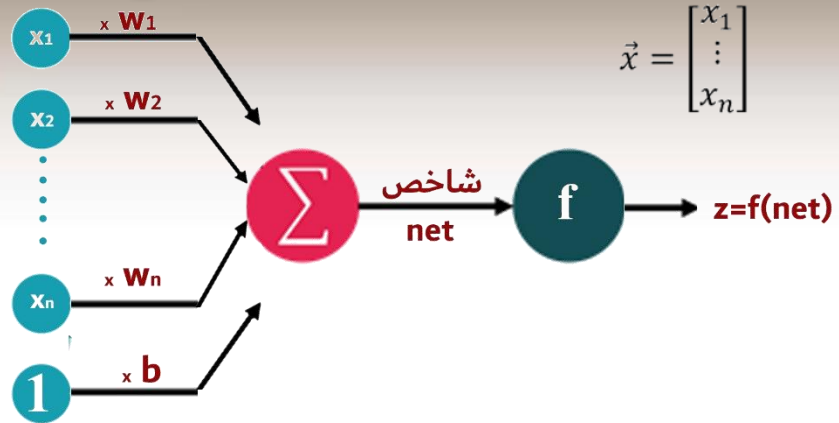


غیر قابل تفکیک بصورت خطی در فضای ورودی

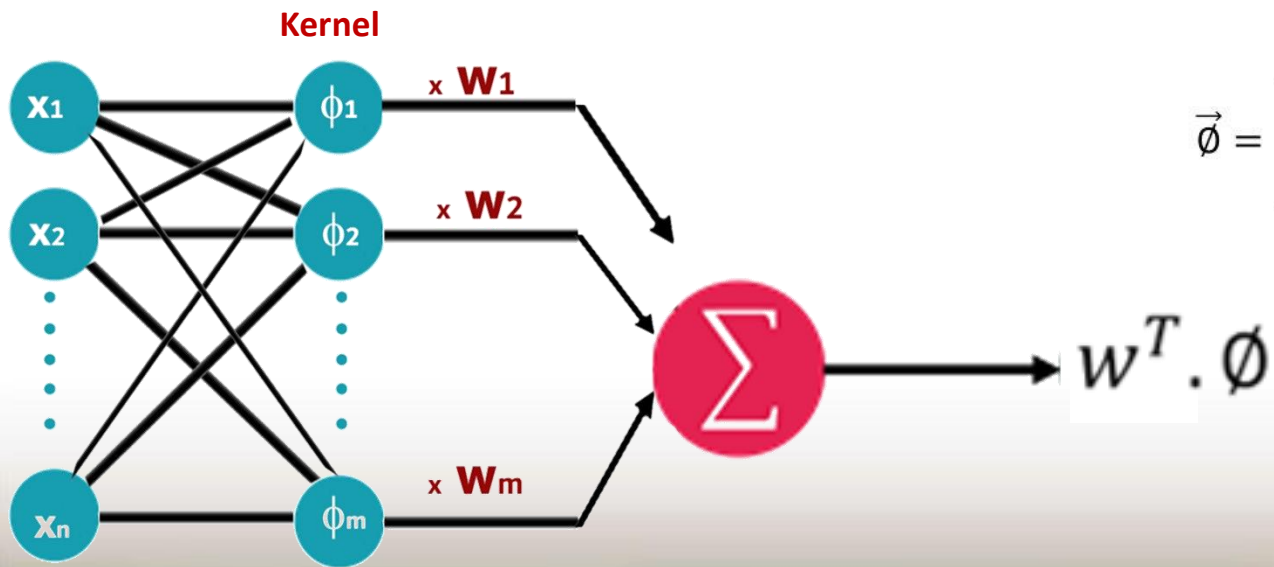


قابل تفکیک بصورت خطی در فضای ویژگی

شبکه عصبی مصنوعی در نرم افزار متلب - مبانی شبکه های عصبی RBF



$$\vec{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \quad \vec{w} = \begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix} \quad \longrightarrow \quad net = [w_1 \ w_2 \ \dots \ w_n] \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} + b = w^T \cdot x + b \quad \longrightarrow \quad Z = f(net) = f(w^T \cdot x + b)$$



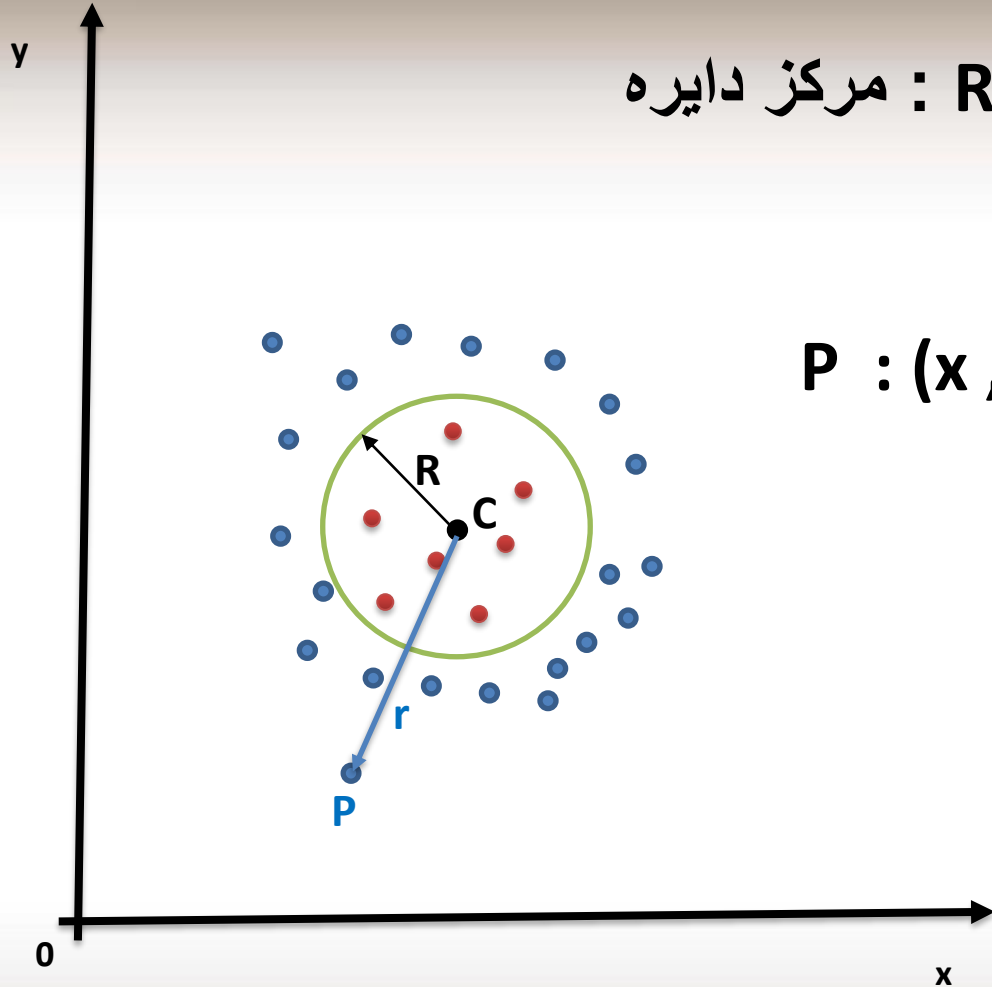
$$\vec{\phi} = \begin{bmatrix} \phi_1 \\ \vdots \\ \phi_m \end{bmatrix} \quad \vec{w} = \begin{bmatrix} w_1 \\ \vdots \\ w_m \end{bmatrix} \quad \longrightarrow \quad net = [w_1 \ w_2 \ \dots \ w_m] \begin{bmatrix} \phi_1 \\ \vdots \\ \phi_m \end{bmatrix} = w^T \cdot \phi$$

شبکه عصبی مصنوعی در نرم افزار متلب - مبانی شبکه های عصبی RBF

مرکز دایره : $R : (x_c, y_c) \longrightarrow R = \sqrt{(x - x_c)^2 + (y - y_c)^2}$

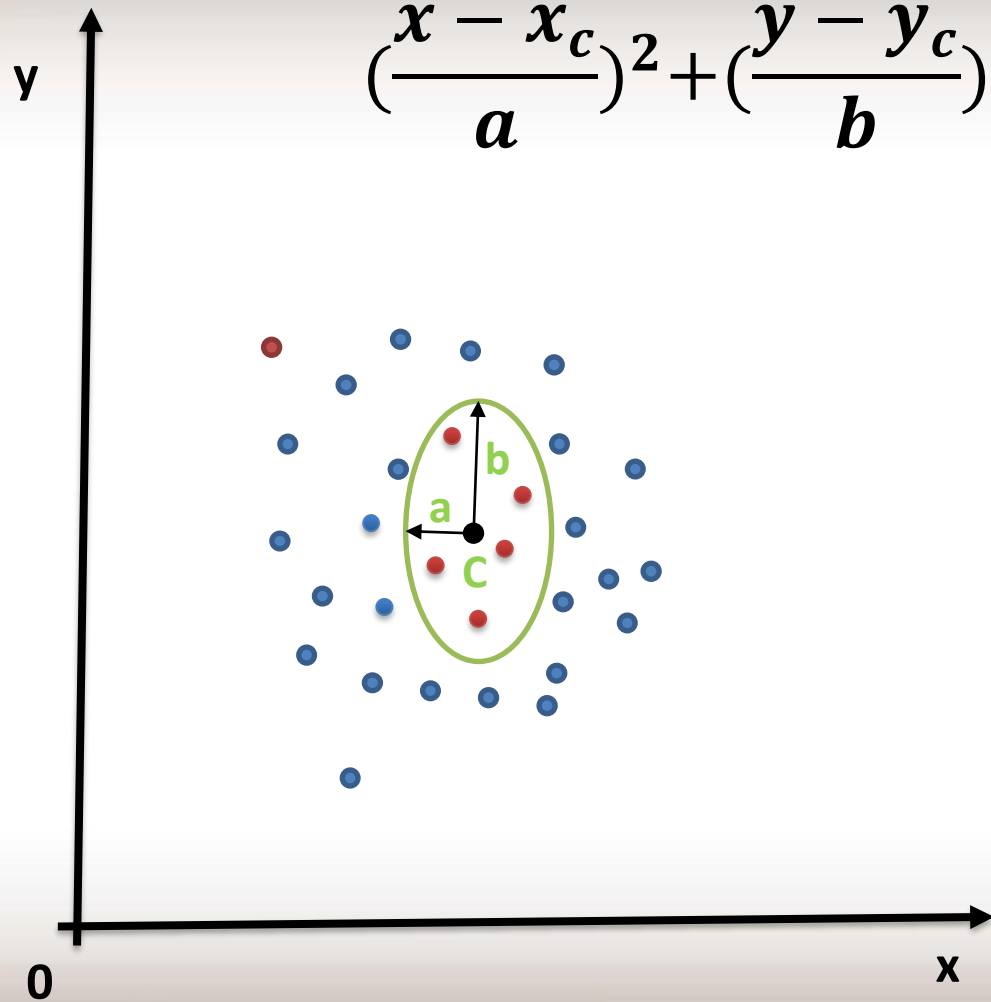
$P : (x, y) \longrightarrow r = \sqrt{(x - x_c)^2 + (y - y_c)^2} = \phi(x, y) = \phi(p)$

$$\begin{cases} \phi \leq R \rightarrow \text{red dot} \\ \phi > R \rightarrow \text{blue dot} \end{cases}$$





شبکه عصبی مصنوعی در نرم افزار متلب - مبانی شبکه های عصبی RBF



$$\left(\frac{x - x_c}{a}\right)^2 + \left(\frac{y - y_c}{b}\right)^2 = 1 \quad \longrightarrow \quad \frac{1}{a^2} (x - x_c)^2 + \frac{1}{b^2} (y - y_c)^2 = 1$$

$$\underbrace{[x - x_c \quad y - y_c]}_{(P - C)^T} \begin{bmatrix} \frac{1}{a^2} & 0 \\ 0 & \frac{1}{b^2} \end{bmatrix} \underbrace{\begin{bmatrix} x - x_c \\ y - y_c \end{bmatrix}}_{(P - c)} = 1$$

$$\underbrace{(P - C)^T \sum_{\emptyset} (P - c)}_{\emptyset} = 1$$



شبکه عصبی مصنوعی در نرم افزار متلب – مبانی شبکه های عصبی RBF

$$\Sigma = \begin{bmatrix} \frac{1}{R^2} & 0 \\ 0 & \frac{1}{R^2} \end{bmatrix} = \frac{1}{R^2} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \longrightarrow \text{دایره}$$

$$\Sigma = \begin{bmatrix} \frac{1}{a^2} & 0 \\ 0 & \frac{1}{b^2} \end{bmatrix} \longrightarrow \text{بیضی بدون دوران}$$

$$\Sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{bmatrix} \longrightarrow \text{بیضی عمومی}$$



شبکه عصبی مصنوعی در نرم افزار متلب - مبانی شبکه های عصبی RBF

تقریب بر اساس مقادیر توابع هسته ای (کرنل)

$$y = \sum_{i=1}^m w_i \cdot \phi_i(x)$$

شکل کاملتر

$$y = \sum_{i=1}^m w_i \cdot \phi_i(x - x_i)$$

اوزان
مرکز تابع ϕ_i
تابع غیر خطی
گوسی

شاخص عملکرد

$$e_i = y_i - y'_i \longrightarrow MSE = \frac{1}{N} \sum_{i=1}^N e_i^2$$

تابع مبتنی بر شعاع (RBF) : تابعی است که مقدار آن به فاصله ورودی از یک نقطه مرکزی ارتباط دارد.



پیاده سازی شبکه عصبی RBF در تولباکس نرم افزار متلب

The screenshot shows the MATLAB Documentation window for the `newrb` function. The window has a blue header with the title "Documentation" and a search bar. Below the header, there are tabs for "All", "Examples", "Functions", and "Apps". The main content area is titled "newrb" and "Design radial basis network". It includes sections for "Syntax", "Description", and "Examples".

Syntax

```
net = newrb(P,T,goal,spread,MN,DF)
```

Description

Radial basis networks can be used to approximate functions. `newrb` adds neurons to the hidden layer of a radial basis network until it meets the specified mean squared error goal.

`net = newrb(P,T,goal,spread,MN,DF)` takes two of these arguments,

Argument	Description
<code>P</code>	R-by-Q matrix of Q input vectors
<code>T</code>	S-by-Q matrix of Q target class vectors
<code>goal</code>	Mean squared error goal (default = 0.0)
<code>spread</code>	Spread of radial basis functions (default = 1.0)
<code>MN</code>	Maximum number of neurons (default is Q)
<code>DF</code>	Number of neurons to add between displays (default = 25)

and returns a new radial basis network.

The larger spread is, the smoother the function approximation. Too large a spread means a lot of neurons are required to fit a fast-changing function. Too small a spread means many neurons are required to fit a smooth function, and the network might not generalize well. Call `newrb` with different spreads to find the best value for a given problem.

Examples

Here you design a radial basis network, given inputs `P` and targets `T`.

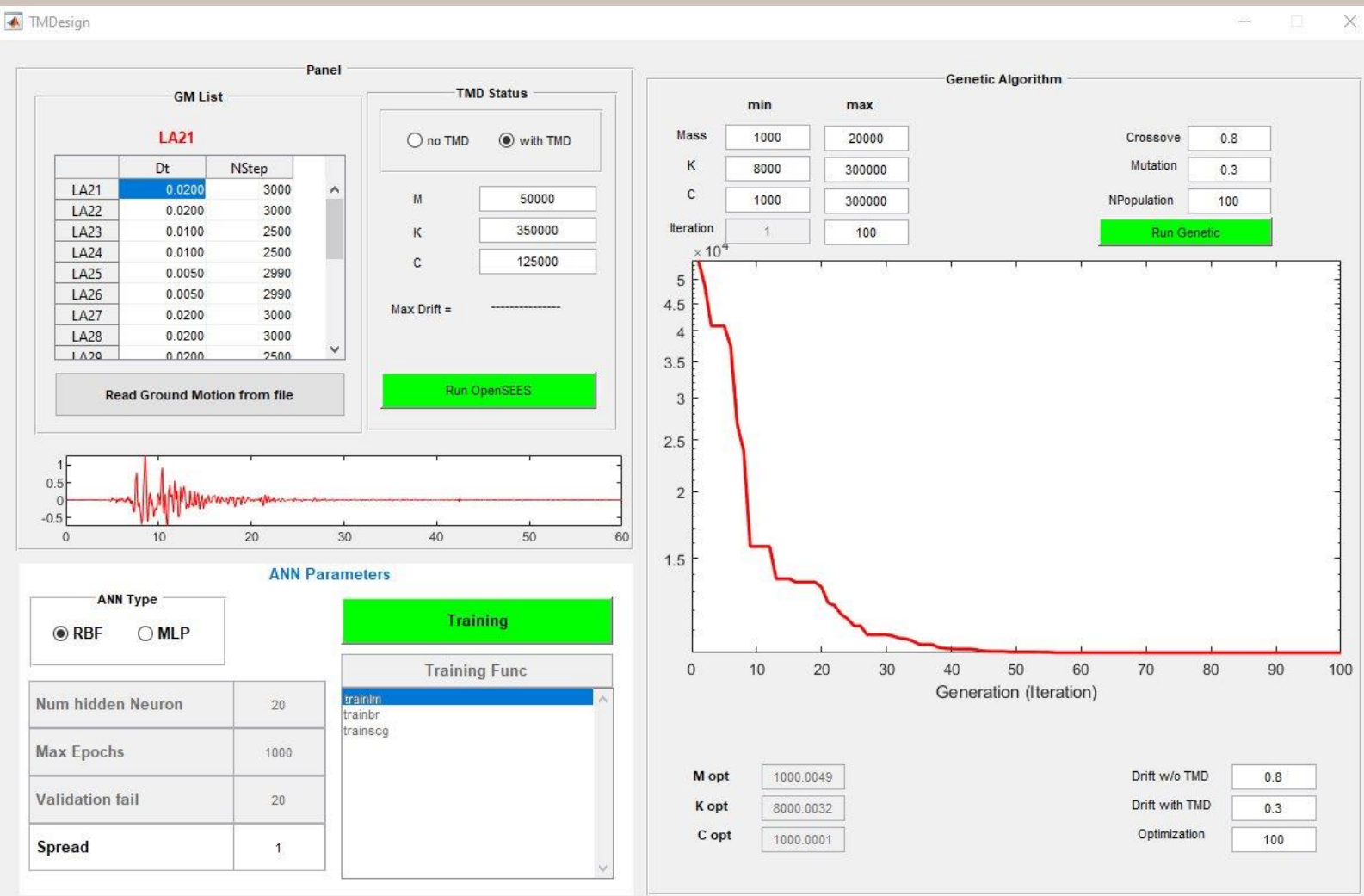
```
P = [1 2 3];  
T = [2.0 4.1 5.9];  
net = newrb(P,T);
```



شبکه عصبی مصنوعی در نرم افزار متلب

طراحی پل شبکه عصبی

در GUI متلب





شبکه عصبی مصنوعی در نرم افزار متلب

کدنویسی پل شبکه عصبی

در GUI متلب

