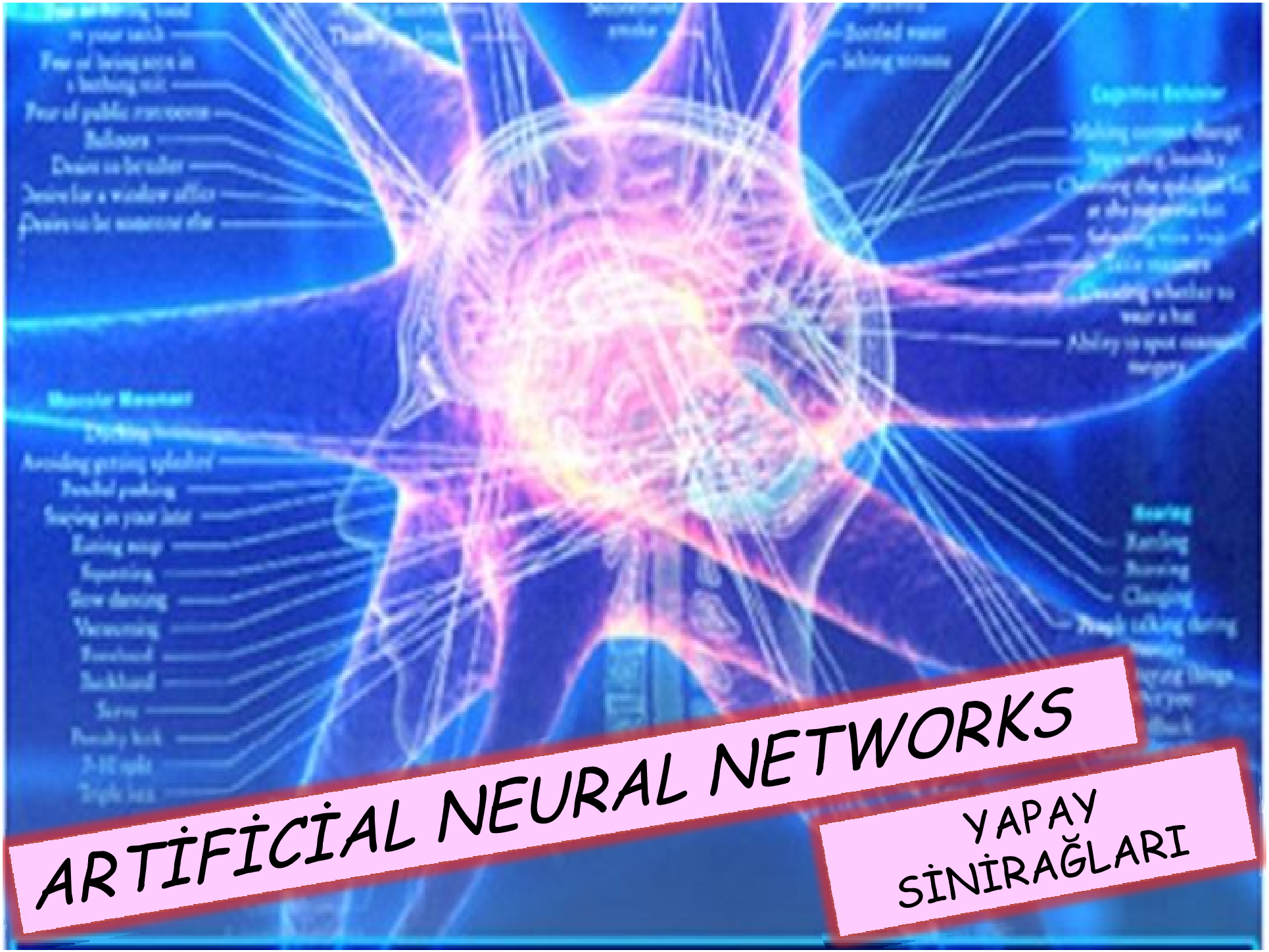
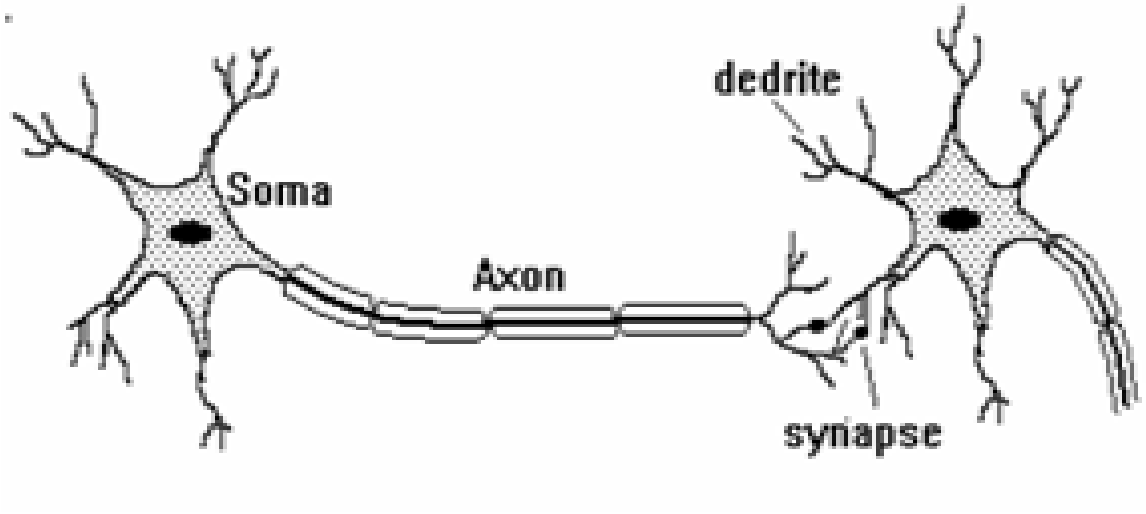


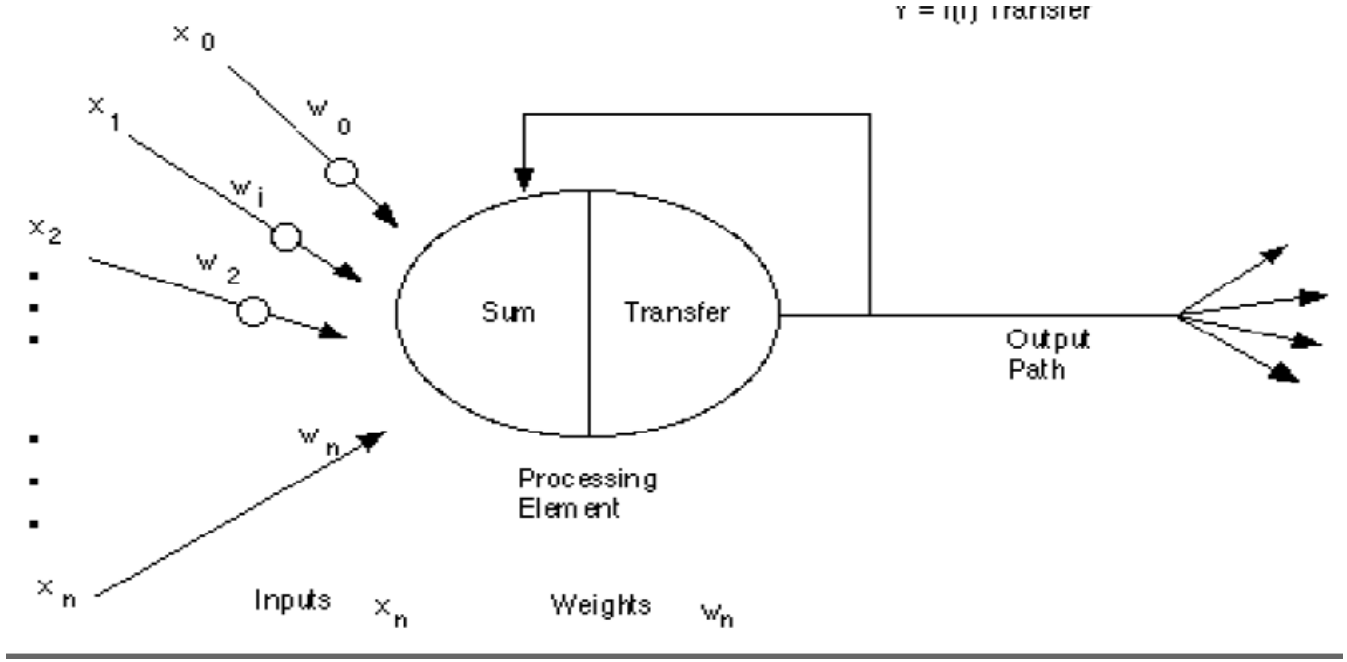
ARTİFİCİAL NEURAL NETWORKS

YAPAY SİNİRAĞLARI

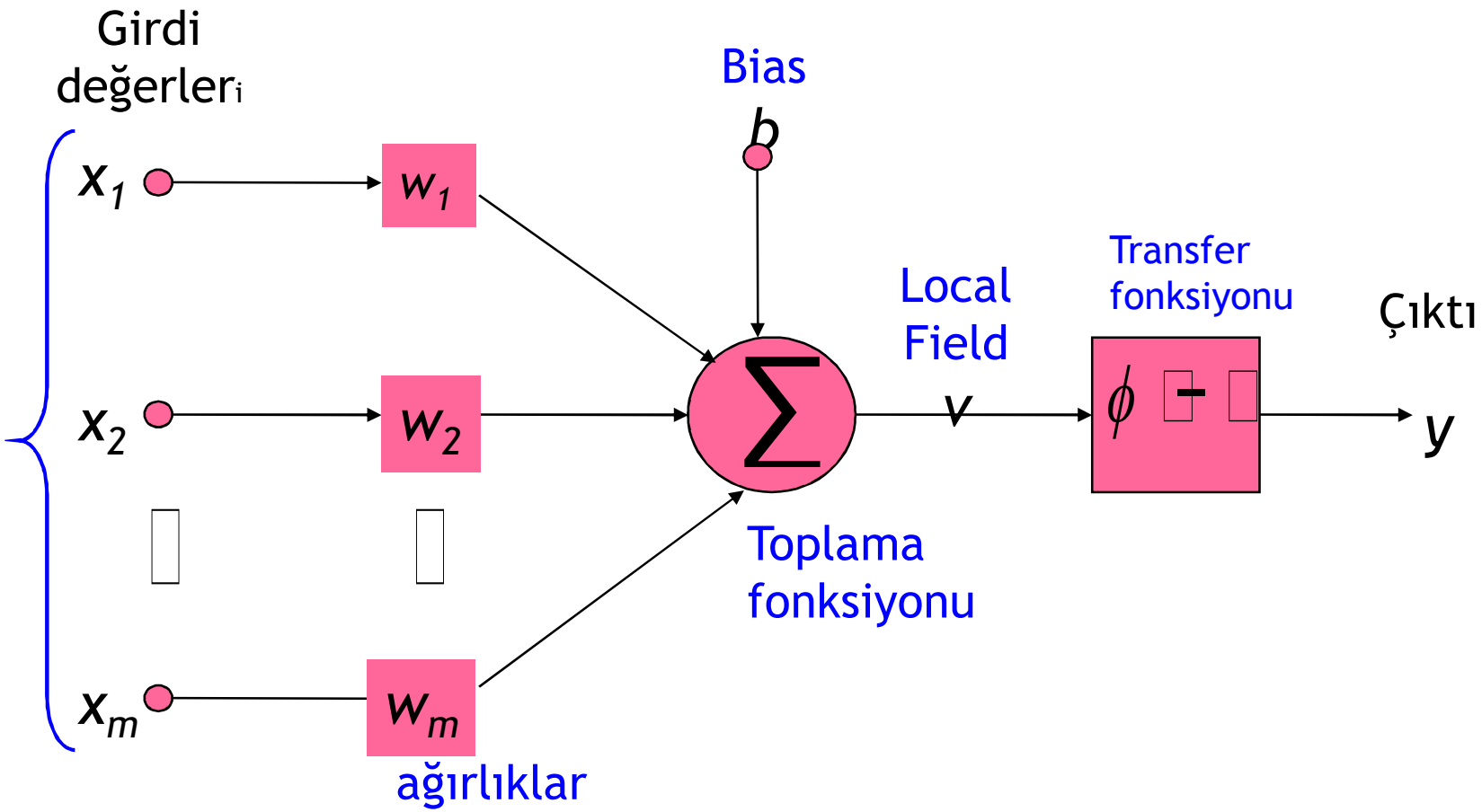
YSA KARŞILAŞTIRMA

BİYOLOJİK SİNİR SİSTEMİ	YAPAY SİNİR SİSTEMİ
NÖRON	İŞLEMCİ ELEMAN
DENDRİT	GİRDİLER
HÜCRE GÖVDESİ	TRANSFER FONKSİYONU
AKSONLAR	YAPAY NÖRON ÇIKIŞI
SİNAPSLAR	AĞIRLIKLAR


dedrite
Soma
Axon
synapse


x 0
x 1
x 2
x n
w 0
w i
w 2
w n
Sum
Transfer
Output
Path
Processing
Element
Inputs x n
Weights w n

YAPAY SİNİR HÜCRESİ



YAPAY SİNİR HÜCRELERİNİN TEMEL ELEMANLARI

- Girdiler
- Ağırlıklar
- Toplama fonksiyonu
- Aktivasyon fonksiyonu
- Çıktılar

YAPAY SINIR HÜCRELERİNİN TEMEL ELEMANLARI

- **GİRDİLER :**

  Yapay sinir ağlarına dışarıdan verilen bilgilerdir.

- **AĞIRLIKLAR :**

Hücreler arasındaki bağlantıların sayısal değeridir.

Bir hücrenin üzerine gelen bilginin değerini ve hücre üzerindeki etkisini gösterir.

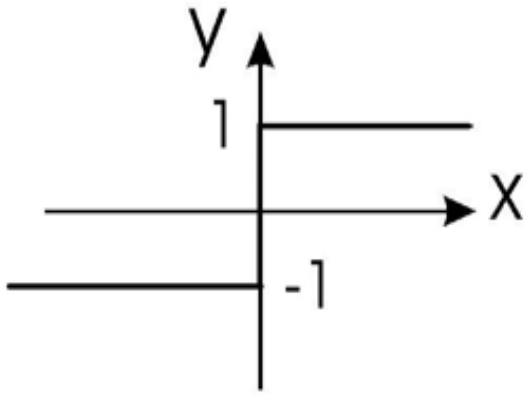
- TOPLAMA FONKSİYONU :

  Hücreye gelen net girdinin hesaplanmasını sağlayan fonksiyondur.

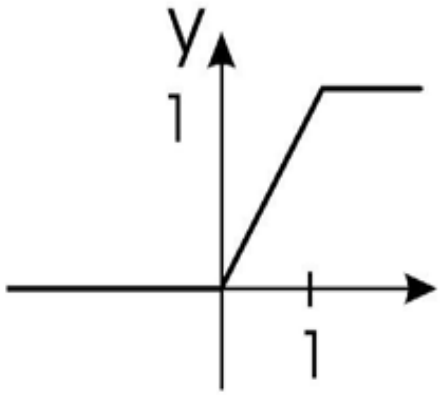
  En yaygın kullanım şekli her girdi değerinin kendi ağırlığıyla çarpılarak toplanmasıdır.

- AKTİVASYON FONKSİYONU :

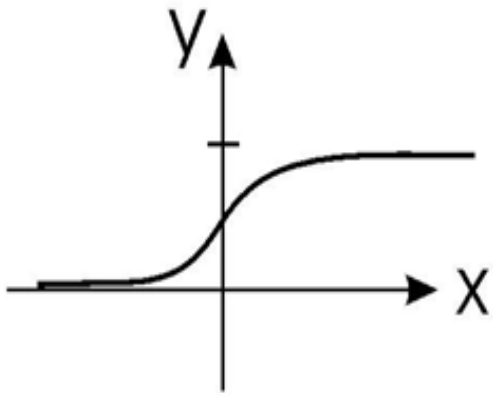
  Bu fonksiyon hücreye gelen net girdinin işlenmesiyle hücrenin bu girdiye karşılık üretileceği çıktıyı belirlemesini sağlar.



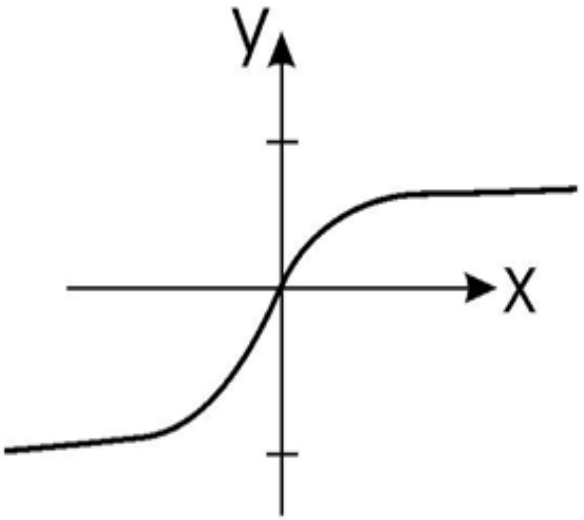
Adım (Step) Fonksiyon



Eşik (Threshold) Fonksiyon



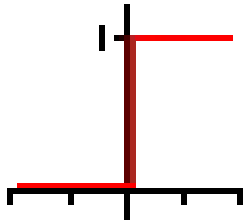
Sigmoid Fonksiyon



Hiperbolik Tanjant Fonksiyon

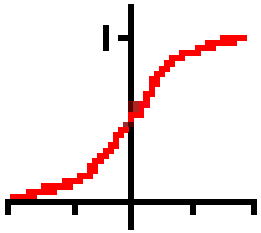
AKTIVASYON FONKSIYONLARI

Unit Step



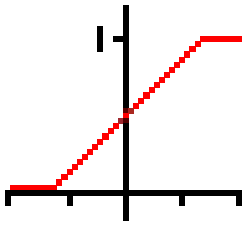
$$f(x) = \begin{cases} 0 \text{ if } 0 > x \\ 1 \text{ if } x \geq 0 \end{cases}$$

Sigmoid



$$f(x) = \frac{1}{1+e^{-\beta x}}$$

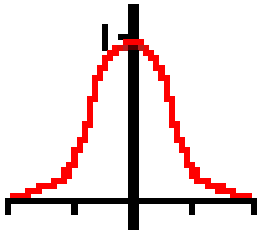
Piecewise Linear



$$f(x) = \begin{cases} 0 & \text{if } x \leq x_{min} \\ mx+b \text{ if } x_{max} > x > x_{min} \\ 1 & \text{if } x \geq x_{max} \end{cases}$$

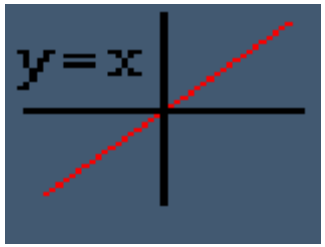
AKTIVASYON FONKSIYONLARI

Gaussian



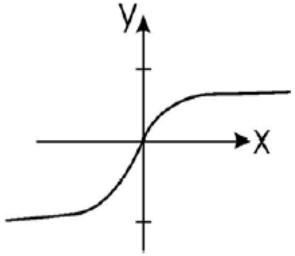
$$f(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$$

Identity

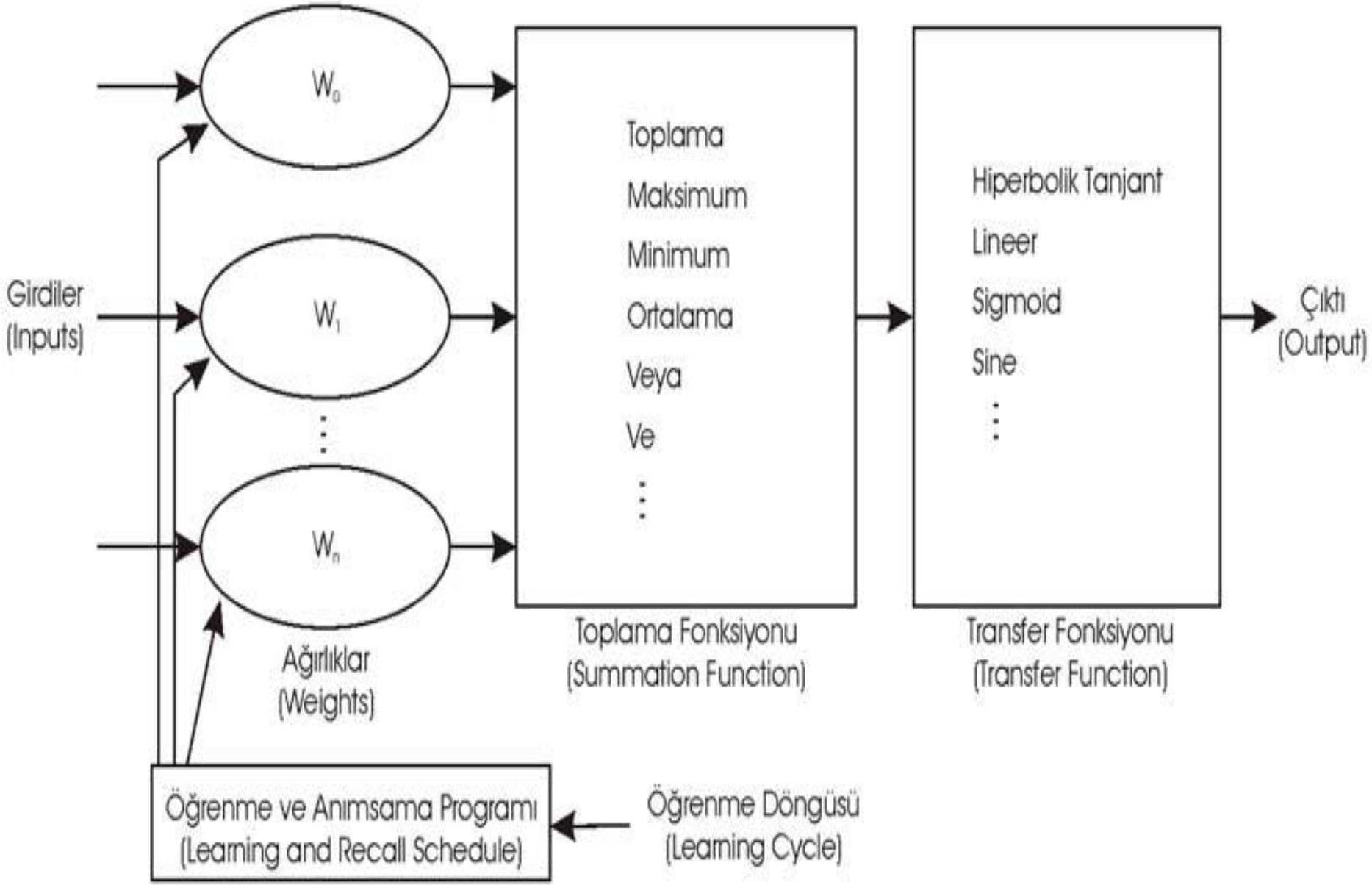


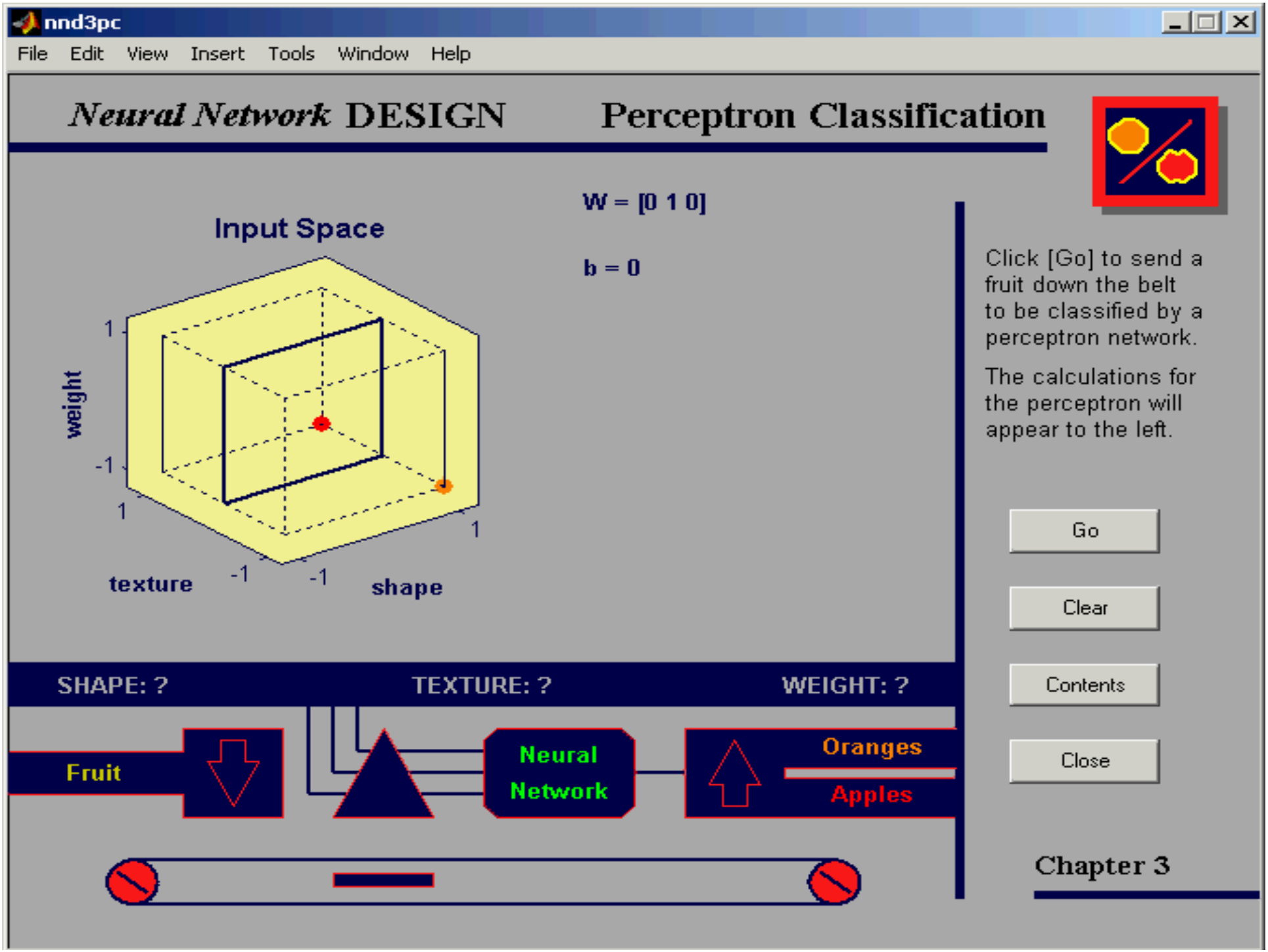
$$f(x) = x$$

Hiberbolik Tanjant

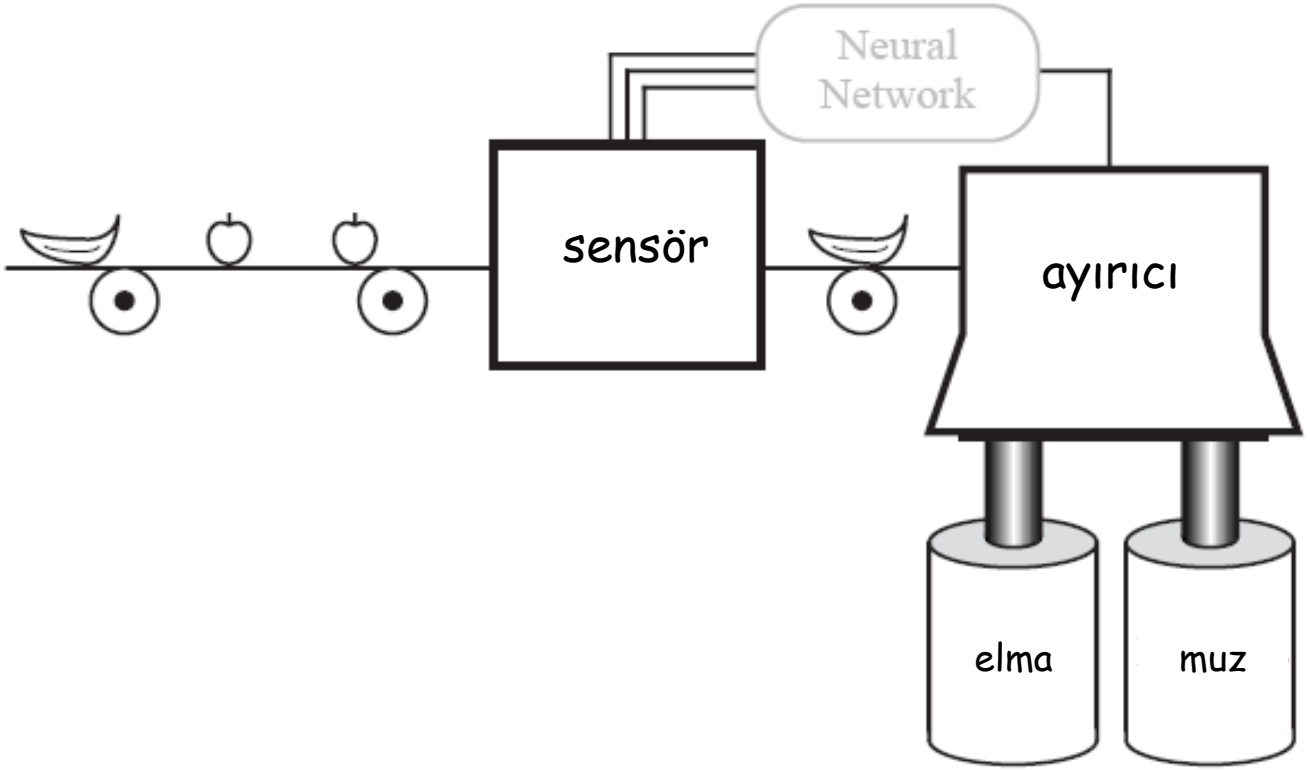


$$f(s) = (1 - \exp(-2s))/(1 + \exp(-2s))$$


W0
W1
Wn
Girdiler
(Inputs)
Toplama
Maksimum
Minimum
Ortalama
Veya
Ve
Hiperbolik Tanjant
Lineer
Sigmoid
Sine
Çıktı
(Output)
Ağırlıklar
(Weights)
Toplama Fonksiyonu
(Summation Function)
Transfer Fonksiyonu
(Transfer Function)
Öğrenme ve Anımsama Programı
(Learning and Recall Schedule)
Öğrenme Döngüsü
(Learning Cycle)


nnd3pc
File Edit View Insert Tools Window Help
Neural Network DESIGN
Perceptron Classification
W = [0 1 0]
b = 0
Input Space
weight
texture
shape
1
-1
1
-1
-1
1
Click [Go] to send a fruit down the belt to be classified by a perceptron network.
The calculations for the perceptron will appear to the left.
Go
Clear
Contents
Close
SHAPE: ?
TEXTURE: ?
WEIGHT: ?
Fruit
Neural Network
Oranges
Apples
Chapter 3

ELMA - MUZ AYIRICI



PROTOTİP VEKTÖRLER

tanıtıcı

vektör

$$\mathbf{p} = \begin{bmatrix} \text{şekil} \\ \text{yüzey} \\ \text{ağırlık} \end{bmatrix}$$

Prototip Muz

$$\mathbf{p}_1 = \begin{bmatrix} -1 \\ 1 \\ -1 \end{bmatrix}$$

Prototip Elma

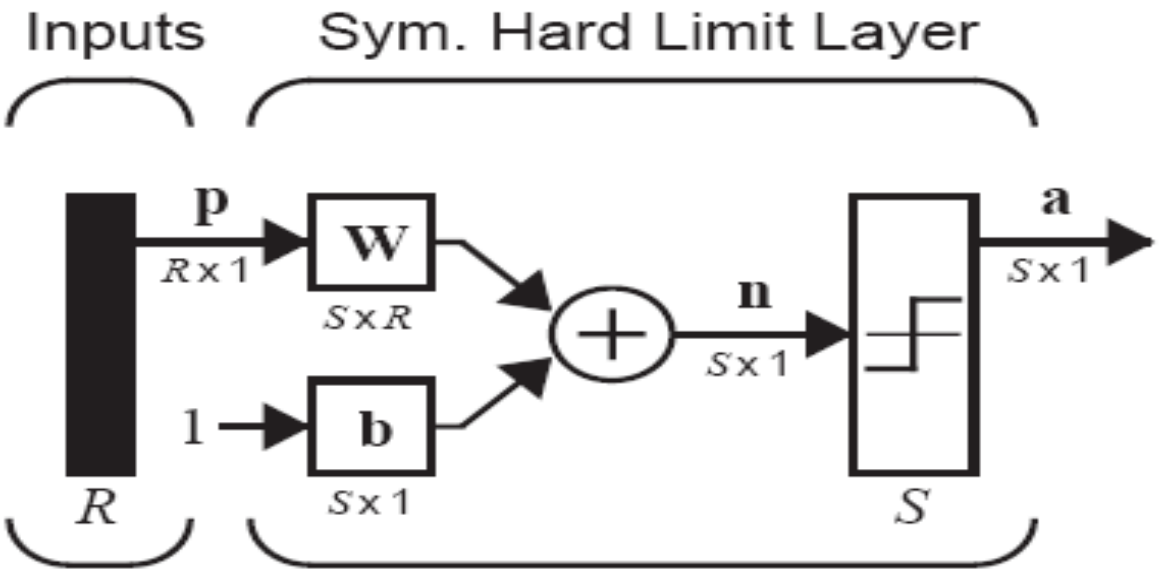
$$\mathbf{p}_2 = \begin{bmatrix} 1 \\ 1 \\ -1 \end{bmatrix}$$

Şekil : {1:yuvarlak; -1:eliptik }

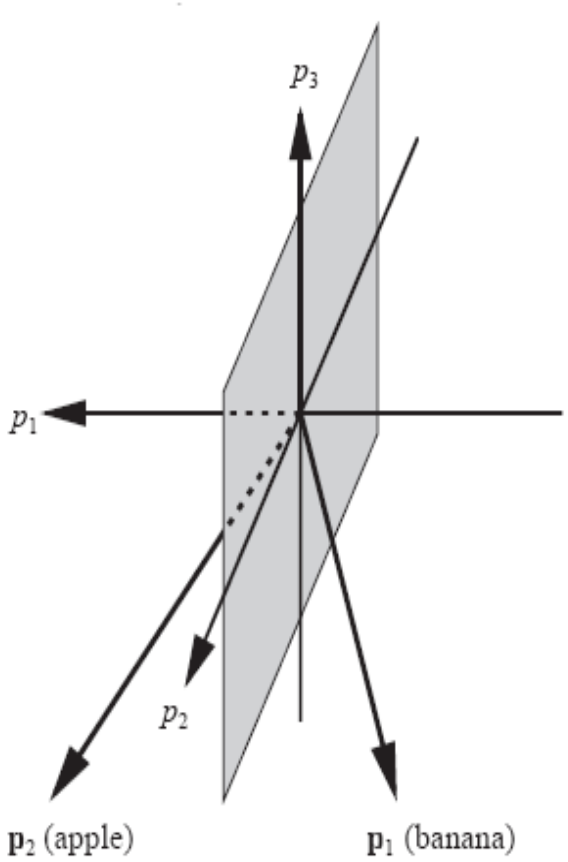
Yüzey : {1:pürüzsüz; -1: pürüzlü }

Ağırlık : {1:>0,5kg ; -1 :< 0,5kg}

ALGILAMA SİSTEMİ



$$\mathbf{a} = \mathbf{hardlims}(\mathbf{Wp} + \mathbf{b})$$



$$\mathbf{a} = \mathbf{hardlims}(\mathbf{Wp} + \mathbf{b})$$

$$a = hardlims\left(\begin{bmatrix} w_{1,1} & w_{1,2} & w_{1,3} \end{bmatrix}\begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} + b\right)$$

$$\begin{bmatrix} 1 & 0 & 0 \end{bmatrix}\begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} + 0 = 0$$

YAPAY AĞIN KONTROLÜ

muz

$$a = hardlims\left(\begin{bmatrix}1 & 0 & 0\end{bmatrix}\begin{bmatrix}-1\\1\\-1\end{bmatrix} + 0\right) = \text{-1 (muz)}$$

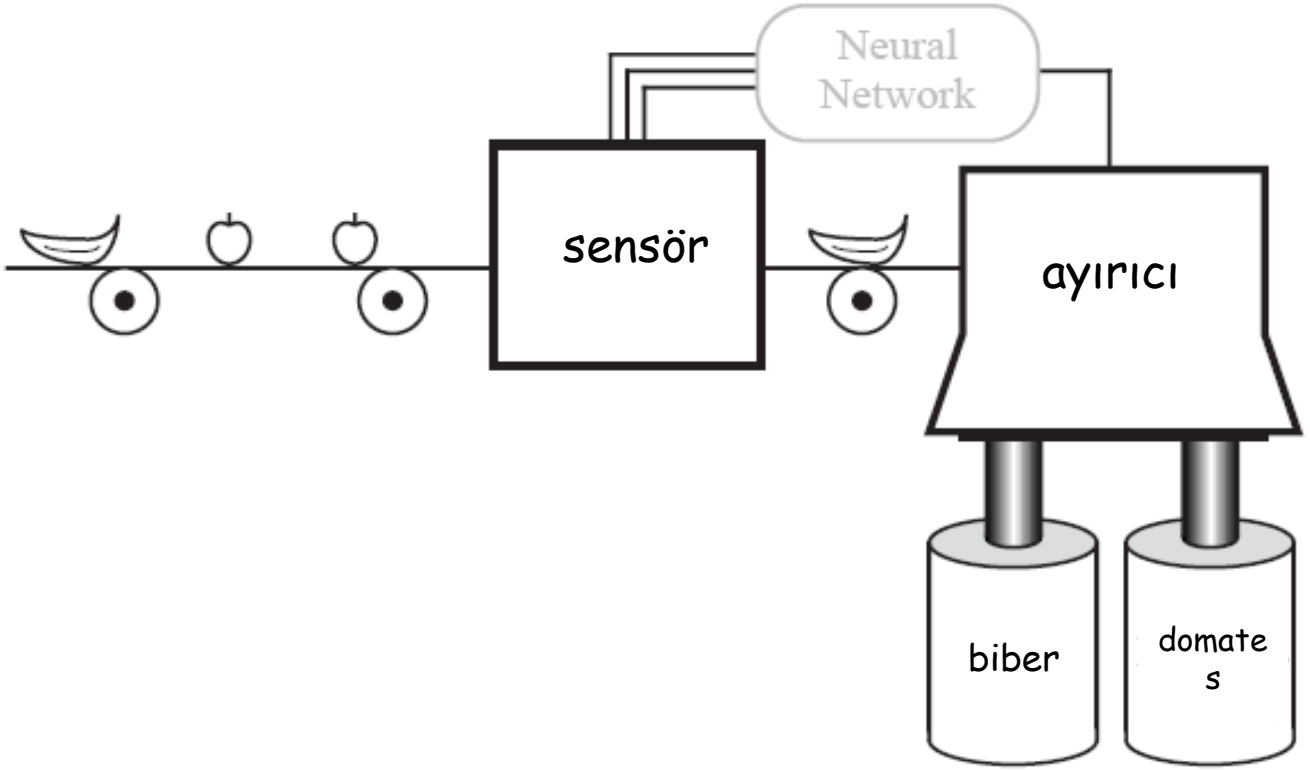
elma

$$a = hardlims\left(\begin{bmatrix}1 & 0 & 0\end{bmatrix}\begin{bmatrix}1\\1\\-1\end{bmatrix} + 0\right) = \text{1 (elma)}$$

"pürüzlü" muz

$$a = hardlims\left(\begin{bmatrix}1 & 0 & 0\end{bmatrix}\begin{bmatrix}-1\\-1\\-1\end{bmatrix} + 0\right) = \text{-1 (muz)}$$

DOMATES - BIBER AYIRICI



PROTOTİP VEKTÖRLER

tanıtıcı vektör

$$\mathbf{p} = \begin{bmatrix} \text{şekil} \\ \text{yüzey} \\ \text{ağırlık} \\ \text{renk} \end{bmatrix}$$

Şekil : {1:yuvarlak; -1:eliptik }

Yüzey : {1:pürüzsüz; -1: pürüzlü }

Ağırlık : {1:>20g; -1 :< 20g}

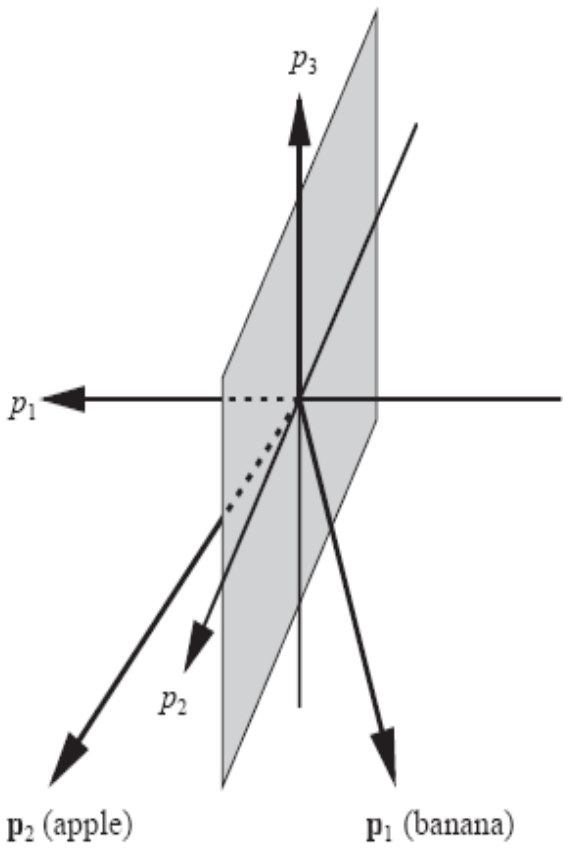
Renk : {1:kırmızı; -1:yeşil }

Prototip biber

$$\mathbf{p}_1 = \begin{bmatrix} -1 \\ 1 \\ -1 \\ -1 \end{bmatrix}$$

Prototip domates

$$\mathbf{p}_2 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$



$$\mathbf{a} = \mathbf{hardlims}(\mathbf{Wp} + \mathbf{b})$$

$$a = hardlims\left(\begin{bmatrix} \text{W1.1 w1.2 w1.3 w1.4} \end{bmatrix}\begin{bmatrix} \text{p1} \\ \text{p2} \\ \text{p3} \\ \text{P4} \end{bmatrix} + b\right)$$

$$\mathbf{[1\ 0\ 1\ 0]}\ \begin{bmatrix} \text{p1} \\ \text{p2} \\ \text{p3} \\ \text{P4} \end{bmatrix} + 0 = 0$$

YAPAY AĞIN KONTROLÜ

biber

$$a = hardlims\left(\begin{bmatrix} 1 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \\ -1 \\ -1 \end{bmatrix} + 0 \right) = -2 \text{ (biber)}$$

domates

$$a = hardlims\left(\begin{bmatrix} 1 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} + 0 \right) = 2 \text{ (domates)}$$

Yeşil domates

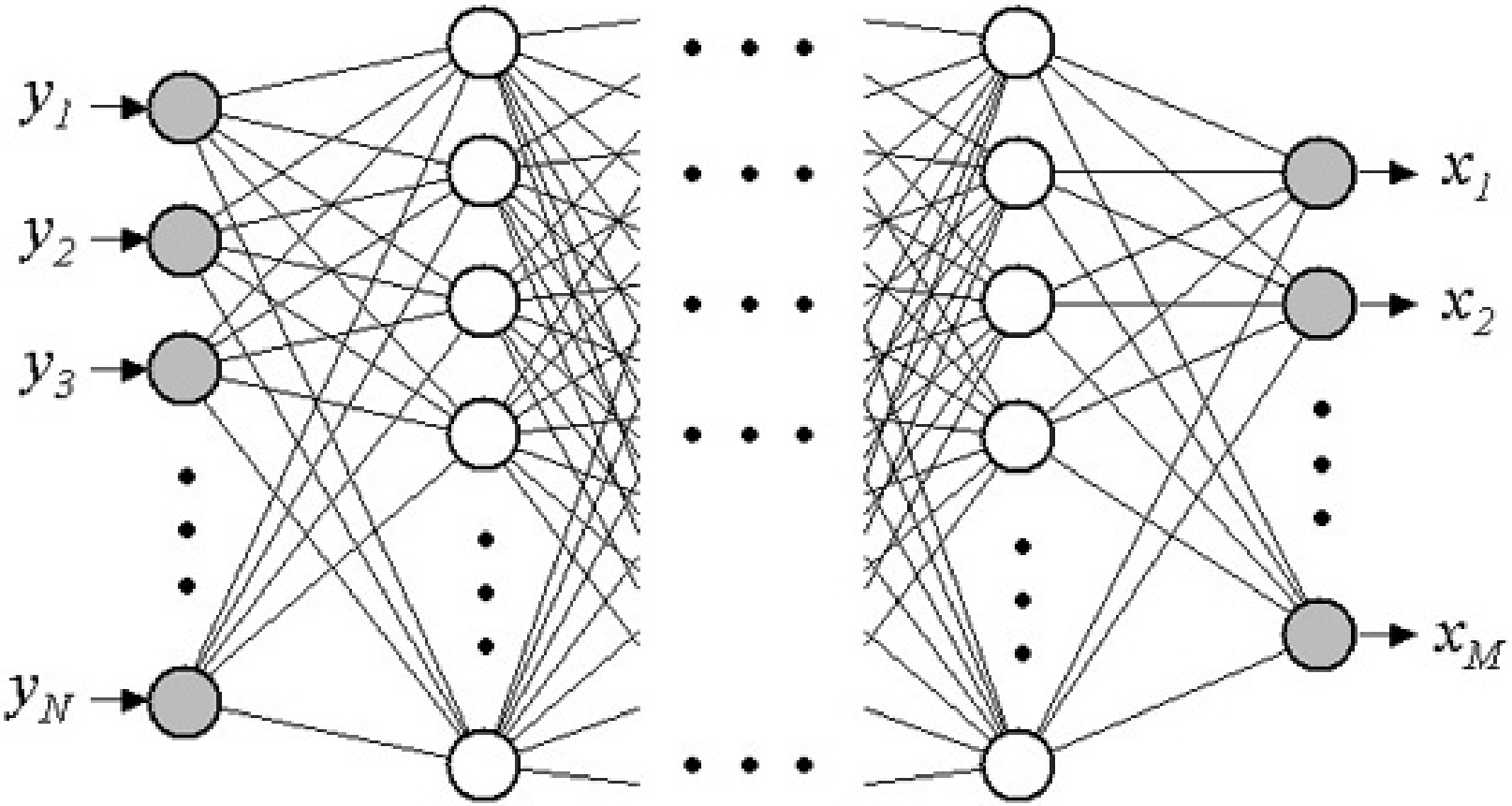
$$a = hardlims\left(\begin{bmatrix}1 & 0 & 1 & 0\end{bmatrix}\begin{bmatrix}1\\1\\-1\\-1\end{bmatrix} + 0\right) = \text{-2 (biber)}$$

Kırmızı biber

$$a = hardlims\left(\begin{bmatrix}1 & 0 & 1 & 0\end{bmatrix}\begin{bmatrix}-1\\1\\-1\\1\end{bmatrix} + 0\right) = \text{- 2 (domates)}$$


Boston Dynamics

YAPAY SİNİR AĞLARININ GENEL YAPISI



KAYNAKLAR

- http://muhendislik.erciyes.edu.tr/tr/bilgisayar/bilakad/ss/index.htm
- Yapay Sinir Ağları Metodolojisi İle Öngörü Modellemesi: Bazı Makroekonomik Değişkenler İçin Türkiye Örneği

  Hasan YURTOĞLU

  Uzmanlık Tezi
- Hagan- Howard

KAYNAKLAR

- İnternet Üzerinde Yapay Zeka ve Yapay Sinir Ağları

  Y. Doç. Dr. Aybars UĞUR

  Ahmet Cumhur KINACI

  Ege Üniversitesi Bilgisayar Mühendisliği Bölümü

  KASIM 2005
- *Yapay Sinir Ağlarında Sınıflandırma*

  Neslihan Bozer