

YAPAY SINIR AĞLARINDA KULLANILAN MATEMATIKSEL KAVRAMLAR.

Note Title

23.10.2020

KÜME:

X bir kume (set) olmak üzere :

$y \in X$: y , X kumesinin elementidir

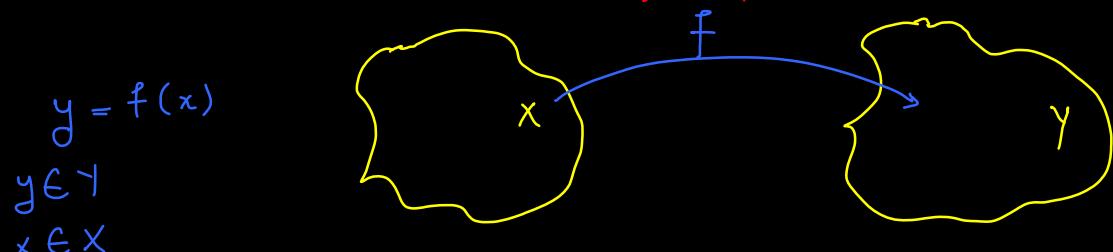
$y \notin X$: y , X kumesinin elementi değildir.

$\{x_1, x_2, x_3, \dots\}$: x_1, x_2 ve x_3 elementlerini içeren kume.

$\{x : x \in \mathbb{R}, x > 5\}$: Değeri 5'ten büyük olan reel sayılar kumesi.

FONKSİYON:

$f : X \rightarrow Y \Rightarrow f$ fonksiyonu X kumesinden Y kumesine tanımlı bir fonksiyondur.



$$\theta \in [0, 2\pi]$$

$$t = \sin(\theta)$$

$$t \in [-1, 1]$$

$f : \mathbb{R}^3 \rightarrow \mathbb{R}^2$: 3-boyutlu reel vektör uzayından 2-boyutlu reel vektör uzayına tanımlı bir fonksiyon.

$$t_i = f(t)$$

$$t_i = A \times t$$

$$2 \times 1 \quad 2 \times 3 \quad 3 \times 1$$

$$t_i = \begin{bmatrix} t_x \\ t_y \end{bmatrix}_{2 \times 1}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}$$

\uparrow

$$t = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}_{3 \times 1}$$

$X := Y \Rightarrow X$ başka bir deyişle Y 'dir.

$$\cosh x := \frac{e^x + e^{-x}}{2}$$

$X \triangleq Y \Rightarrow$ tanım eşitliği :

$$\langle x, y \rangle \triangleq \|x\| \|y\| \cos \theta$$

Vektörler :

$$a = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_n \end{bmatrix}_{n \times 1}$$

$\Rightarrow n \times 1$ boyutuna sahip olan vektörlerse

i) sütun matrisi

ii) n -vektör deplir.

$$t = \begin{bmatrix} -1 \\ 0 \\ 4 \\ 2 \end{bmatrix} \Rightarrow t$$
 vektör 4-vektördür.

Transpoz :

A $m \times n$ tipinde bir matris olmak üzere A matrisinin transpozu A^T ile gösterilir ve A^T matrisi $n \times m$ tipindedir.

$$A = \begin{bmatrix} -2 & 0 & 4 & 7 \\ 6 & 0 & 1 & 3 \end{bmatrix}_{2 \times 4}$$

$$A^T = \begin{bmatrix} -2 & 6 \\ 0 & 0 \\ 4 & 1 \\ 7 & 3 \end{bmatrix}_{4 \times 2}$$

$$(A^T)^T = A //$$

Eğer $A^T = A$ ise A matrisi simetrik matristir.

Noktasal Çarpımı (Dot Product) :

$x, y \in \mathbb{R}^n$ olmak üzere :

$$\langle x, y \rangle = \sum_{i=1}^n x_i y_i = x_1 y_1 + x_2 y_2 + \dots + x_n y_n$$

| SONUÇ DAIMA SKALERDIR.

ÖRNEK :

$$a = \begin{bmatrix} -1 \\ 0 \\ 3 \\ 4 \\ 7 \end{bmatrix}_{5 \times 1}, b = \begin{bmatrix} 2 \\ 3 \\ -4 \\ 6 \\ 0 \end{bmatrix}_{5 \times 1} \Rightarrow \langle a, b \rangle = (-1) \cdot 2 + 0 \cdot 3 + 3 \cdot (-4) + 4 \cdot 6 + 7 \cdot 0$$

$$\Rightarrow \langle a, b \rangle = 10 \text{ elde edilir.}$$

$$\langle x, y \rangle = x^T \cdot y$$

$1 \times n \quad n \times 1 = 1 \times 1$

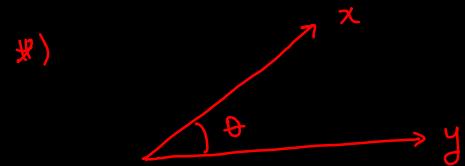
$$\langle a, b \rangle = a^T \cdot b$$

$$= \begin{bmatrix} -1 & 0 & 3 & 4 & 7 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \\ -4 \\ 6 \\ 0 \end{bmatrix} = 10 //$$

*) $\langle x, x \rangle \geq 0$, eğer $\langle x, x \rangle = 0$ ise $x = 0$ 'dır.

*) $\langle x, y \rangle = \langle y, x \rangle$

*) $\langle x, y \rangle = 0$ ise $x \perp y$ 'dır. \Rightarrow ortogonal // OFDM

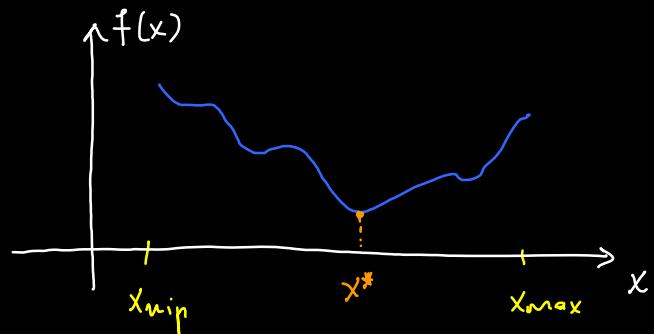


Eğer $\langle x, y \rangle$ sonucu ile
 $\langle t, y \rangle$ sonucu birbirine çok yakın
ise x vektörü t vektörü benzer
davranza sahiptir diyebiliriz.

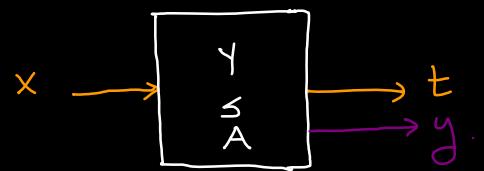
*) Bir vektörün euklid normu .

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$
$$\|x\| = \sqrt{x_1^2 + x_2^2 + x_3^2}$$
$$\|x\| = \sqrt{\langle x, x \rangle} = \sqrt{x^T x}$$

OPTIMIZASYON : (En iyileme)



matematiksel
bir fonksiyonu minimum
yapan değer kumesinin
elde edilmesi islenine
optimizasyon denir
optimizasyon = minimizasyon.



t : Hedef

y : YSA

$$\downarrow \text{Hata} = |t - y|$$

0 minimize

$$y \longrightarrow t$$



Convergence
Yakinsama //

SINIF

10 kisi

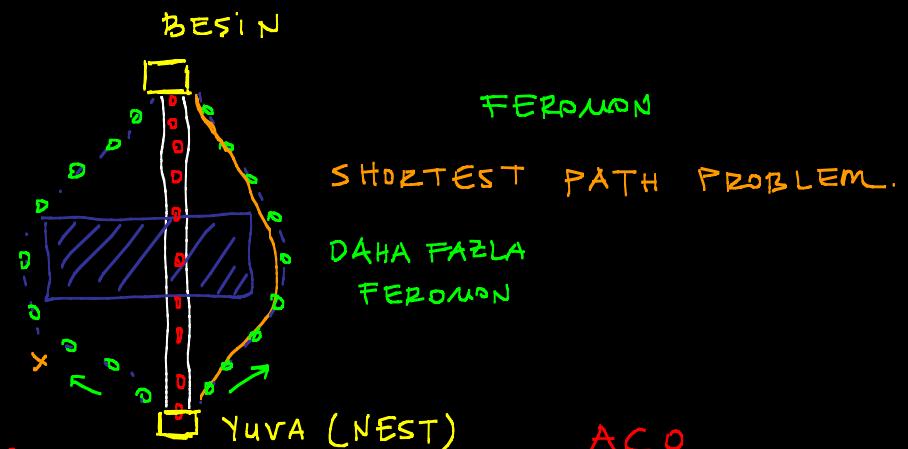
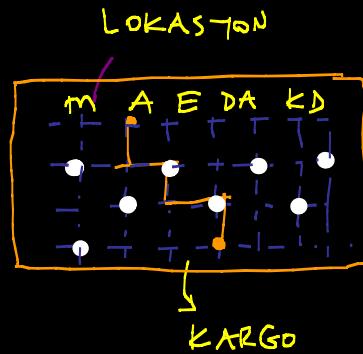
A	$\rightarrow 100\ 00001$
B	$\rightarrow 100\ 00010$
C	$\rightarrow 00011$
D	
E	
F	
G	

1000

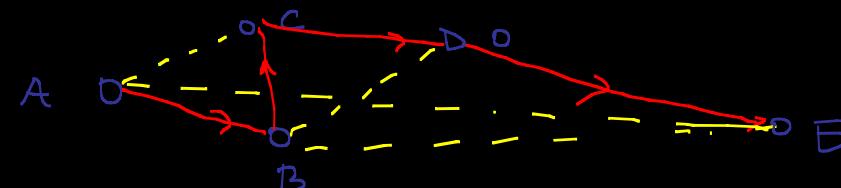
YAPAY ZEKA: Doğadaki zeki davranış gösteren biyolojik sistemlerin sahip oldukları üstün özelliklerini makinelere aktararak aynı davranışları sayısal sistemler ile gerçekleştirilmesini içeren kelimelerdir.

Temel fikir sayısal bilgisayarlar kullanılarak ve uygun programlama dilleri cinsinden ifade ederek zeka gerelitisen elementlerin sayısal ortamda gerçekleştirilmesidir.

SWARM INTELLIGENCE → SÜRÜ ZEKASI.



TRAVELLING SALESMAN PROBLEM



ACO
Ant
Colony
optimization

SAYISAL BİLGİSAYARLAR

Seri adımlar

Hızlı Birimler kullanılır

Sayısal

Tam tanımlı komut dizisi

Giriş verilerine duyarlıdır

BETİN

Paralel adılar.

Yavaş birimler kullanılır.

Analog.

Bilgi cereye adapte olabilir.

Giriş renklerine duyarlı
değildir.

Beyin:

- i) En karmaşık organıdır.
- ii) Kişi den kişiye, cinsiyetler arası farklılıklar gösterir.
- iii) 10^{10} sinir hücresi vardır
- iv) Her sinir hücresi yaklaşık 10^4 sinaptik bağlantı yapar.
- v) $1.3 \text{ kg}, 0.15 \text{ m}^2$, yaklaşık 100 Hz çalışma frekansına sahip.
- vi) Beynin farklı bölgeleri farklı işlevlerden sorumludur.

vii) Beynin temel yapı taşı, sinir hücresi dir.

NÖRON → Neuron

İki tip sinir hücresi vardır

i) İla Sinir Hücreleri : Kendi öz giriş ve öz çıkış bağlantıları vardır.

ii) Çıkış hücreleri :

a) Beynin bir bölgesini diğerine bağlayan hücreler.

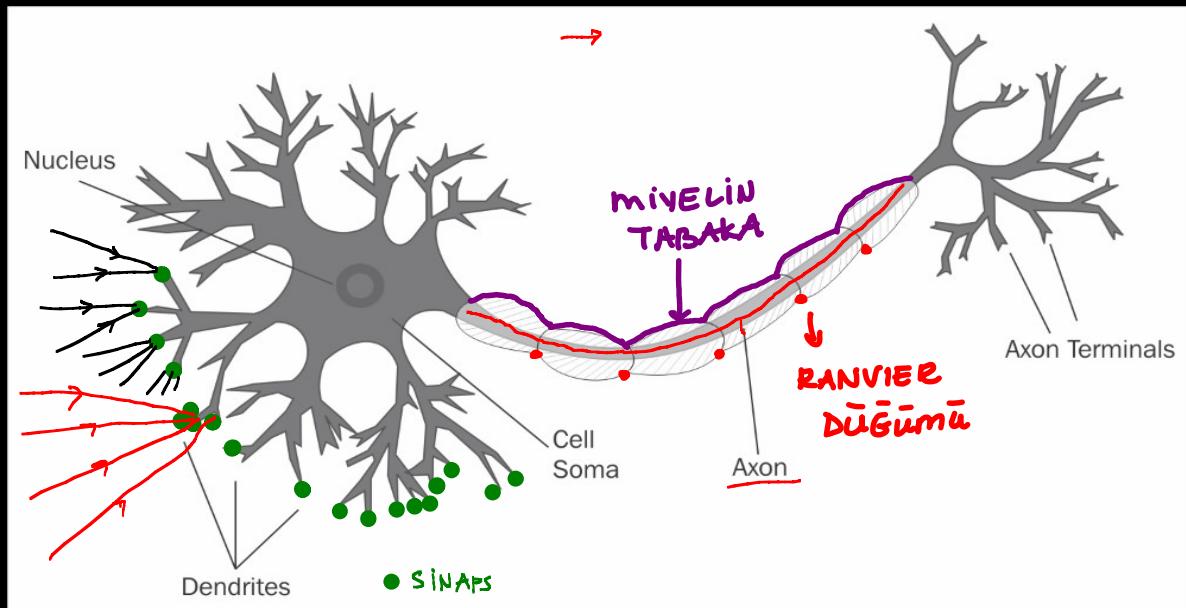
b) Beyni kaslara bağlayan hücreler.

c) Duyu organlarını beyne bağlayan hücreler.

Sinir hüresinin blok diyagramı :



BİYOLOJİK NÖRON :



AKSON : Çikış davranışının üretildiği elektriksel olarak aktif görevde uzantılarıdır.

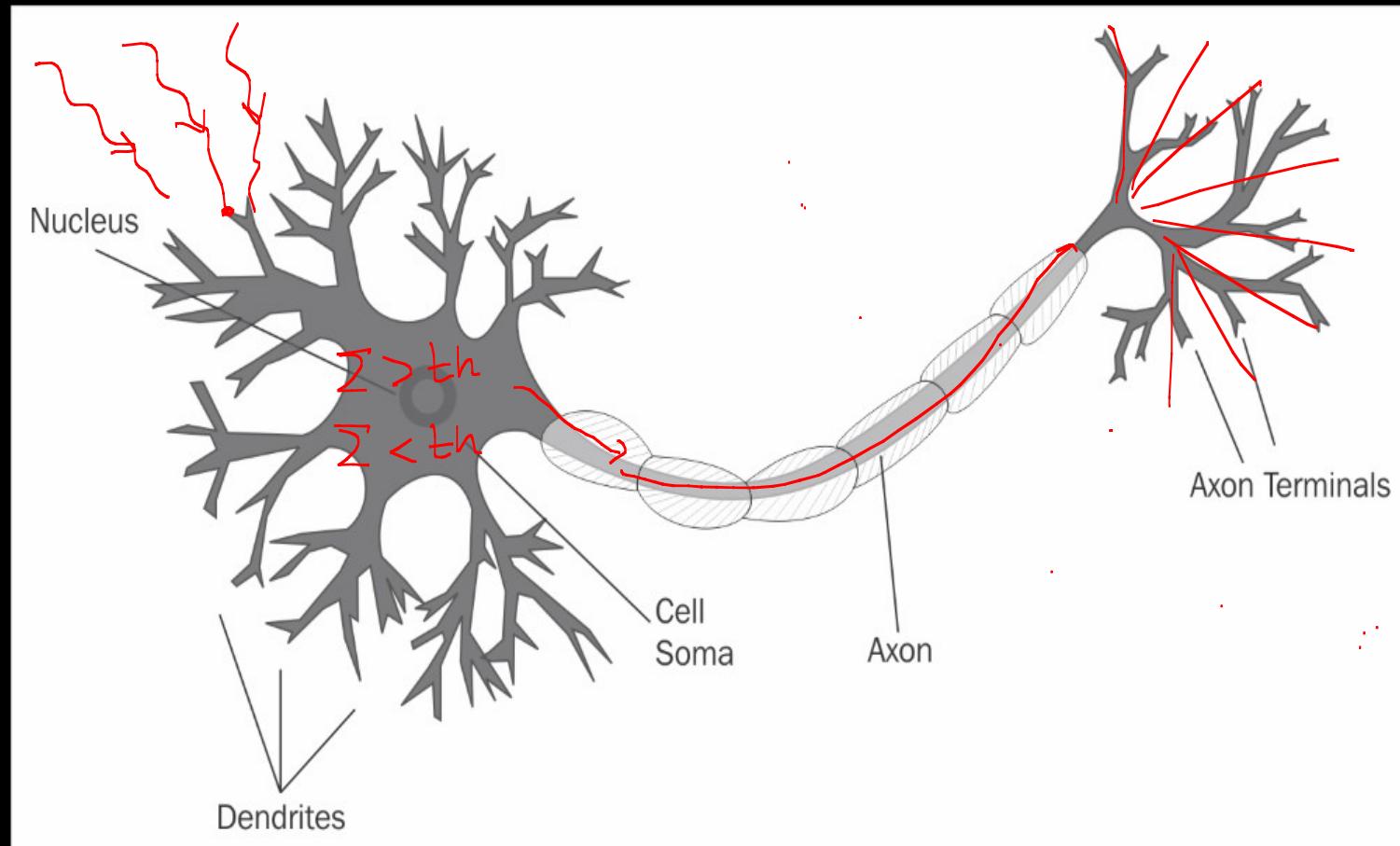
SOMA : Sinir hücresinin görevdesidir. 5-100 μm çaplı şekiptir.

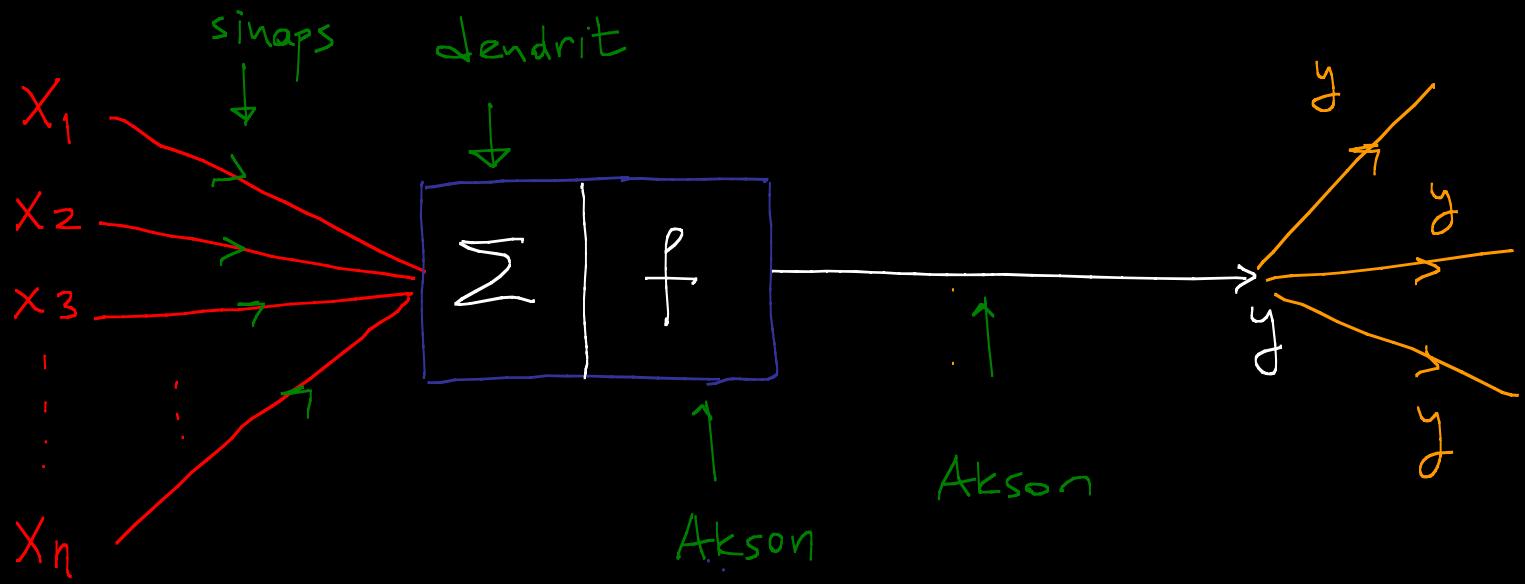
DENDRİT : Diğer hücrelerden gelen elektriksel işaretleri toplayan elektriksel olarak pozitif kollardır.

SİNAPS : Bir hücrenin aksonunun diğer hücrenin dendritine bıçaklı noktaları.

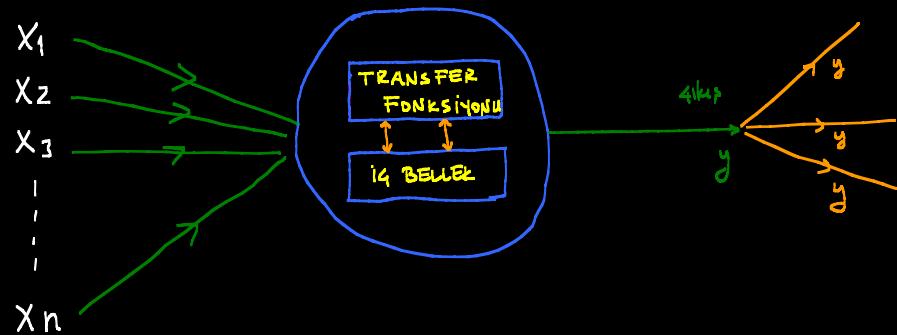
MİYELİN TABAKA : Sitoplazma ile hücreler arası sivi arasındaki kapasiteyi düşürerek sinyallerin yayılma hızını artırıcı yalıtım maddesi.

RANVIER DÜĞÜMÜ : Akson üzerinde miyelin tabaskayla kaplı birkaç mm'de bir yer olan ve işaretlerin periyodik olarak yeniden ırtılmasını sağlayan boğumlardır.





İşlem Birimleri : (Processing Unit)



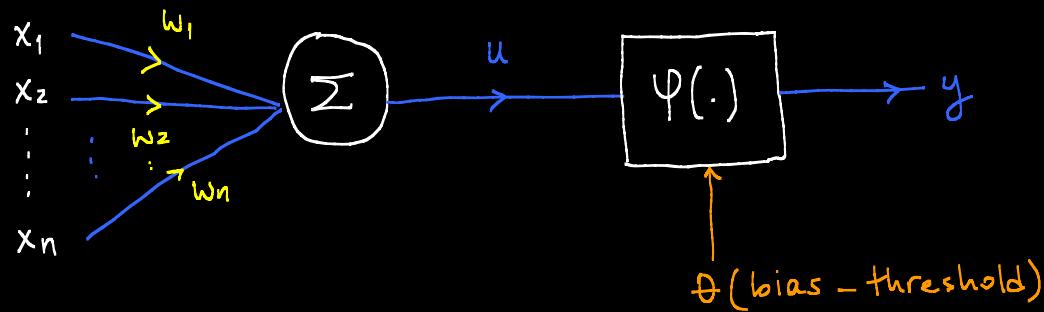
Giriş Birimleri : Sisteme dışardan bilgi almakı sağlamak

Gılkıp Birimleri : Sistemin aksine ve bağlı olduğu birimlere bilgi göndermek.

Saklı Birimler : Çok karmaşık işlevlerin yapıldığı, işlevsel birlik

HIDDEN UNIT en fazla olduğu birimdir.

TEMEL NÖRON MODELİ (McCulloch-Pitts Modeli - 1943)



x_1, x_2, \dots, x_n : Giriş sinyalleri

w_1, w_2, \dots, w_n : Sinaptik ağırlık

$\psi(\cdot)$: Aktivasyon Fonksiyonu

y : Çıktı

$$u = \sum_{i=1}^n w_i x_i = w_1 x_1 + w_2 x_2 + \dots + w_n x_n$$

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}_{n \times 1}, \quad W = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix}_{n \times 1}$$

$$y = \psi(u - \theta)$$

$$u = x^T \cdot w = w^T \cdot x$$

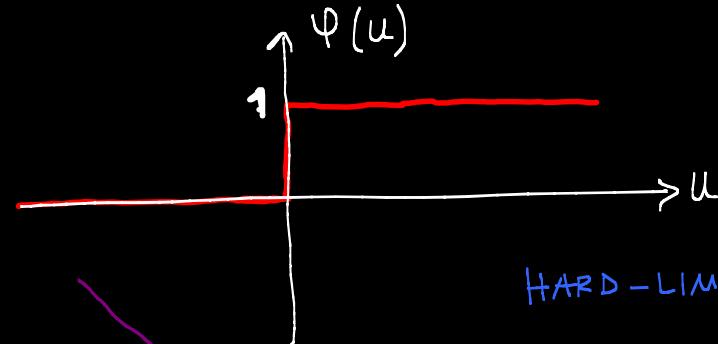
$$y = \psi(w^T \cdot x - \theta)$$

$$1 \times n \quad n \times 1 \rightarrow 1 \times 1$$

giriş ile çıkış arasındaki matematiksel ilişki!

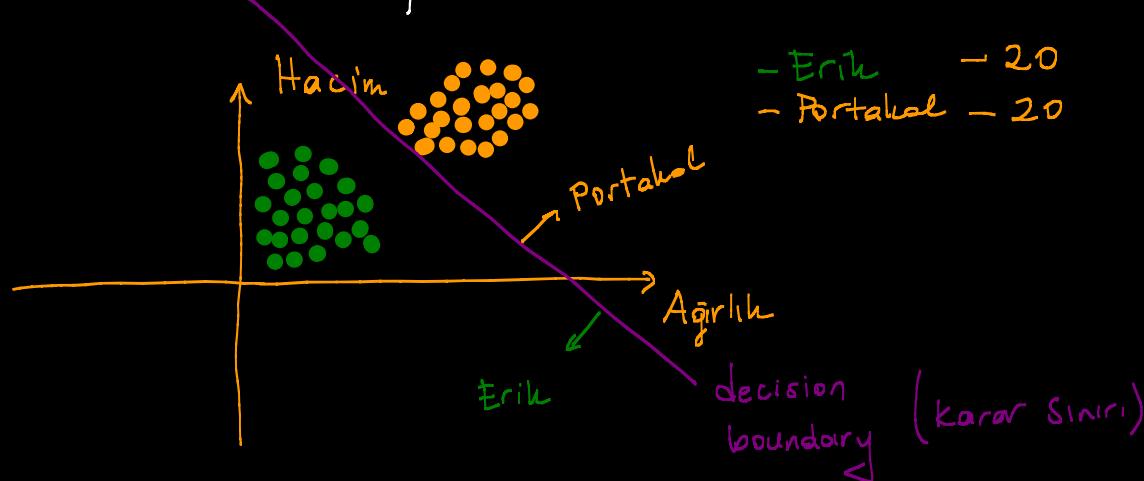
- | Temel nöron modeli (Perceptron-Algoritması) iki sınıfı ve doğrusal olarak sınıflandırılabilen sınıflandırma problemini rahatlıkla çözmektedir.

Perceptron modeli, aktivasyon Fonksiyonu olarak HARD LIMITING (keskin Geçişli) FUNCTION kullanılır.

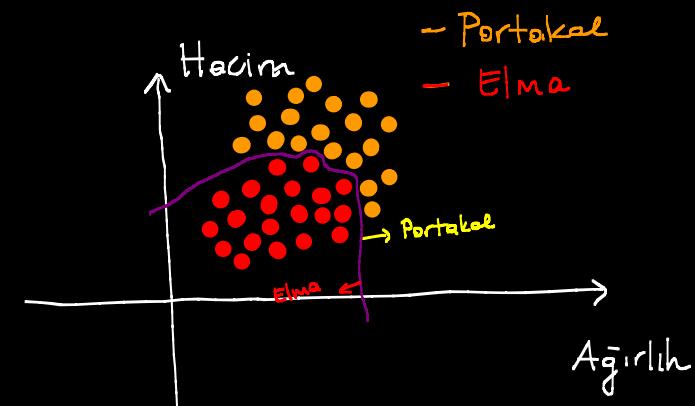


$$\phi(u) = \begin{cases} 1, & u \geq 0 \\ 0, & u < 0 \end{cases}$$

HARD-LIMITING FUNCTION (hardlim)

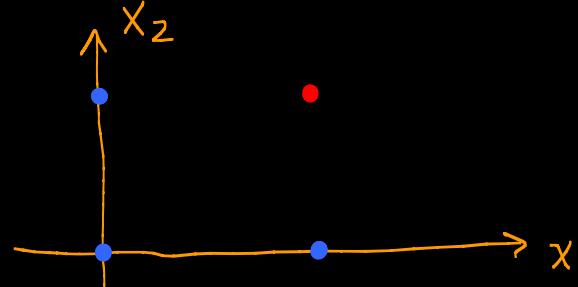


Degraded data
cannot be classified



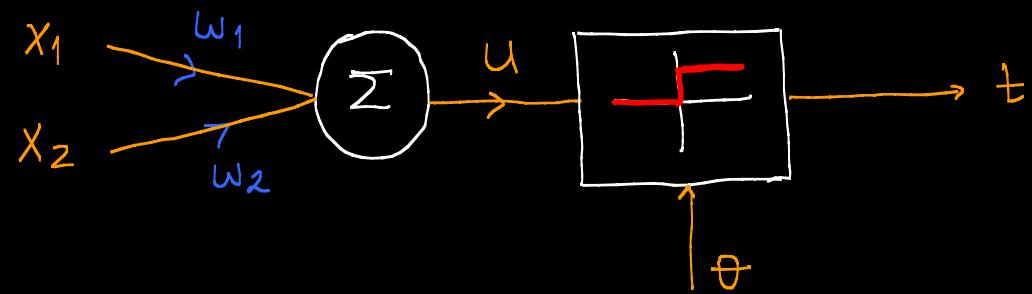
For example AND problem to solve:

x_1	x_2	t
0	0	0
0	1	0
1	0	0
1	1	1



HASTA	SAGLIKLI
0	1
ELMA	PORTAKAL
0	1
KADIN	EZGEEK
0	1

=

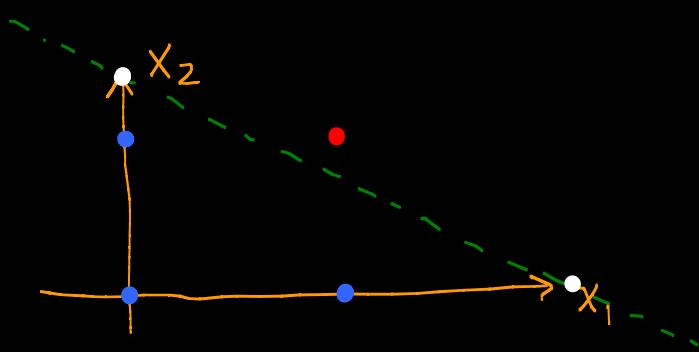


$$t = \varphi(u - \theta)$$

$$t = \varphi(\omega_1 x_1 + \omega_2 x_2 - \theta)$$

$$\omega_1 x_1 + \omega_2 x_2 - \theta \geq 0 \Rightarrow t = 1$$

$$\omega_1 x_1 + \omega_2 x_2 - \theta < 0 \Rightarrow t = 0$$



$$\omega_1 x_1 + \omega_2 x_2 - \theta = 0$$

$$\omega_1 x_1 + \omega_2 x_2 = \theta$$

Burada temel problem en uygun
 ω_1, ω_2 ve θ değerlerinin bulunmasıdır.

ω_1 ve ω_2 burada doğrunun ekseni kestiği noktaları ve doğrunun eğimini belirler.

θ ise doğrunun eğimini değiştirmeksizin düzleme kaydırılmasını sağlar.

$$\omega_1 x_1 + \omega_2 x_2 - \theta = 0$$

$$i) x_1 + x_2 - 1 = 0$$

$$ii) 2x_1 + x_2 - 1 = 0$$

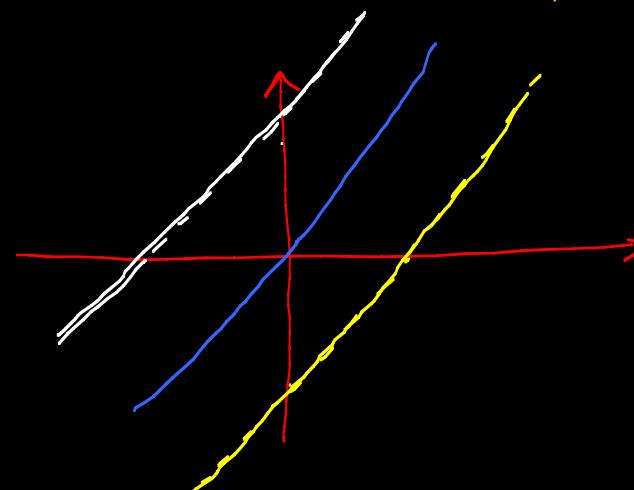
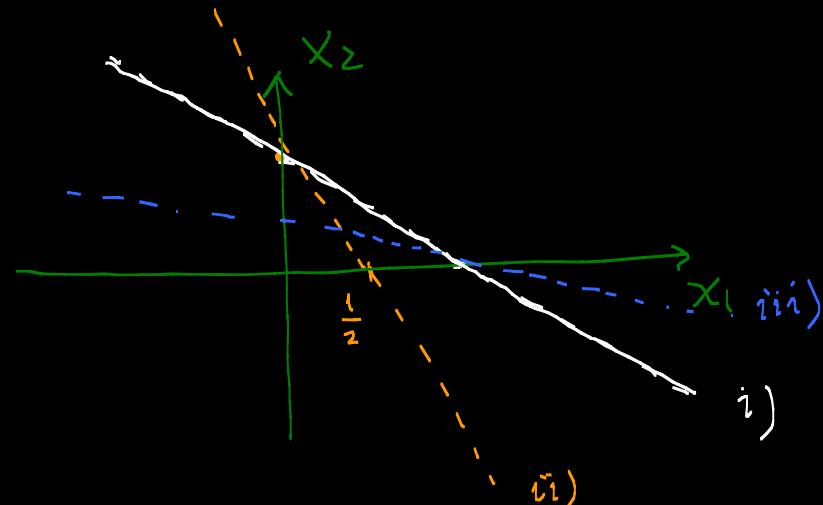
$$iii) x_1 + 2x_2 - 1 = 0$$

$$\omega_1 x_1 + \omega_2 x_2 - \theta = 0$$

$$x_1 - x_2 - 1 = 0$$

$$x_1 - x_2 + 1 = 0$$

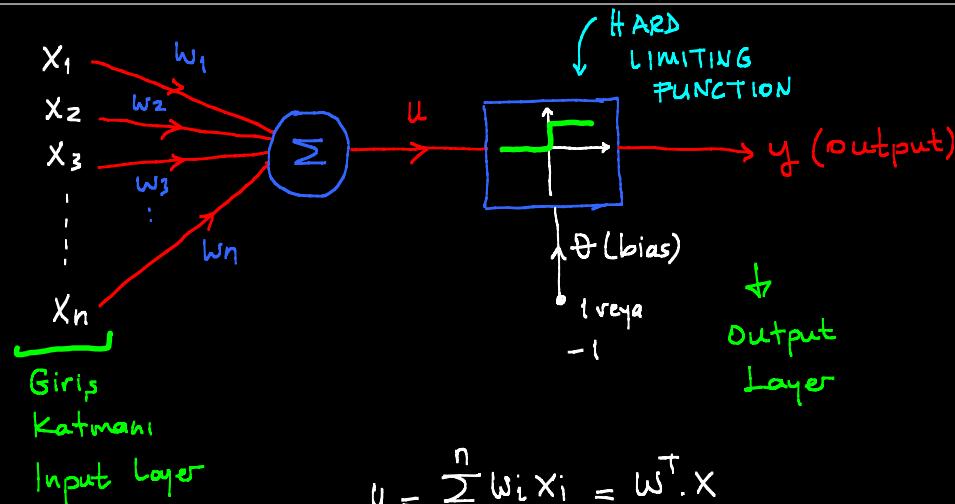
$$x_1 - x_2 = 0$$



PERCEPTRON ÖĞRENME KURALLI :

Note Title

20.11.2020



$$u = \sum_{i=1}^n w_i x_i = \mathbf{w}^\top \mathbf{x}$$

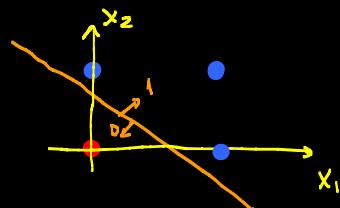
$$y = \Phi(\mathbf{w}^\top \mathbf{x} - \theta)$$

! Perceptronlar iki sınıfı ve doğrusal bir şekilde sınıflandırabilen problemleri kesersiz bir şekilde sınıflandırır.

Örneğin OR problemini ele alalım:

x_1	x_2	t
0	0	0
0	1	1
1	0	1
1	1	1

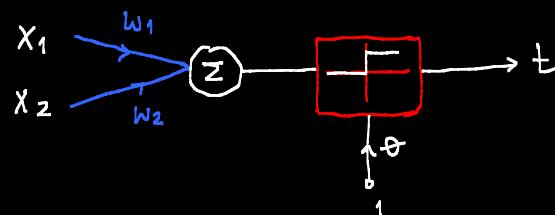
→ 1.sınıf
→ 2.sınıf }
OR problemi iki sınıfı bir sınıflandırma problemi olarak ele alınabilir



Eğitim giriş verisinde OR problemi için girişleri ve dört örneklüten oluşturulan için girişler sütun şeklinde ifade edilmelidir.

			sütun sayısı - örnek sayısı	
	0	1	0	1
1)	0	0		
2)	0	1		
3)	1	0		
4)	1	1		

satır sayısı
giriş sayısı



$$t = \text{hardlim}(w_1x_1 + w_2x_2 - \theta)$$

problem kusursuz bir sınıflandırma yapmak için en uygun w_1, w_2 ve θ değerlerinin bulunması.

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

\downarrow
doğru denklemi olduğu için

w_1, w_2 ve θ değerlerine göre bir
değeri gibidir ve bu değeri sınırıdır!

PERCEPTRON ÖĞRENME KURALI :

Adım 1 : Ağırlık (w) ve bias (θ) için başlangıç değerleri ata.

İpuçu : Başlangıç değerleri sıfırda olsun veya sıfırın yakını
klaiki real sayılar şeklinde seçilmelidir.

Adım 2 : Eğitim verisindeki her örnek (j) için aşağıdaki
işlemleri gerçekleştir:

a) J. örnek için ağılnın çıkışını ($y_j(t)$) hesapla.

$$y_j(t) = \psi(w^T \cdot x - \theta)$$

b) Ağırlık ve bias değerlerini güncelle.

$$w_i(t+1) = w_i(t) + \eta \overbrace{(t_j - y_j(t))}^{\text{Hedef çıkış}} \cdot x_j$$

$$\theta(t+1) = \theta(t) + \eta (t_j - y_j(t))$$

Adım 3 : Hedef çıkışlar ile ağılnın çıkışları arasındaki hatayı hesapla.

$$E = \frac{1}{n} \cdot \sum_{i=1}^n (t_i - y_i)^2$$

ortalama
karesel
hata

mean
square
error.

Adım 4 : Eğer hata sıfıra esit ise algoritmayı sonlandır.

Aksi durumda Adım 2'ye dön.

				iterasyon	
		x_1	x_2	t	y
→	→	0	0	0	0
→	→	0	1	1	0
→	→	1	0	1	1
→	→	1	1	1	1

w, θ güncellendi
w, θ " " $y = \varphi(w^T x + \theta)$

$w, \theta \leftarrow$

$\frac{1}{n} \sum_{i=1}^n (t_i - y_i)^2 = \frac{2}{4} = 0.5$

$$0 + 1 + 1 + 0 = 2$$

AKTİVASYON FONKSIYONLARI

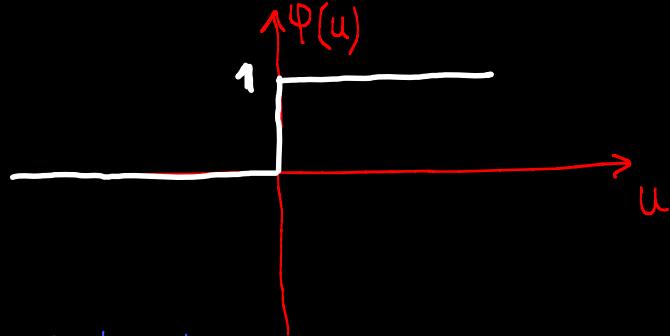
Note Title

04.12.2020

Aktivasyon fonksiyonları, yapay sinir ağlarında gelen bilginin işlenmesi ve çıkış işaretinin oluşturulması amacıyla kullanılan fonksiyonlar.

1) Keskin Geçişli (HARD LIMITING) fonksiyonlar:

$$f(u) = \begin{cases} 1, & u \geq 0 \\ 0, & u < 0 \end{cases}$$



İki sınıfı sınıflandırıcılarında kullanılır

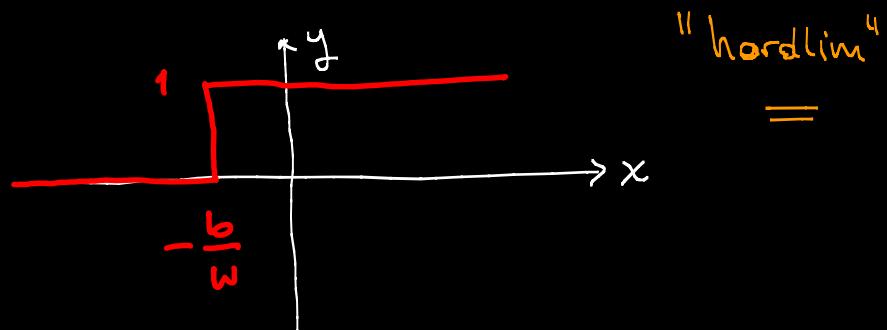
↳ Birinci sınıf etiketi : 0 (Hasta), (Kırmızı)

↳ ikinci sınıf etiketi : 1 (Sağlıklı), (Kırmızı olmayan)

$$y = \text{hardlim}(wx+b)$$

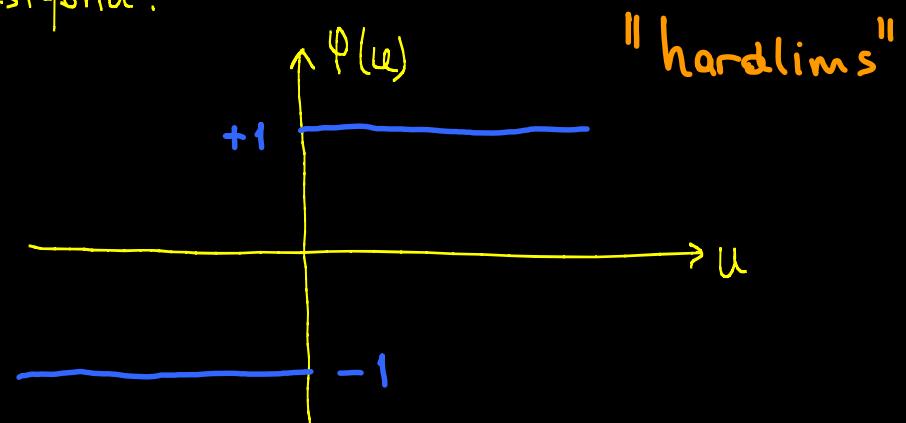
$$wx+b = 0$$

$$x = -\frac{b}{w}$$



^{)} Simetrik hardlim fonksiyonu:

$$\Psi(u) = \begin{cases} 1, & u \geq 0 \\ -1, & u < 0 \end{cases}$$

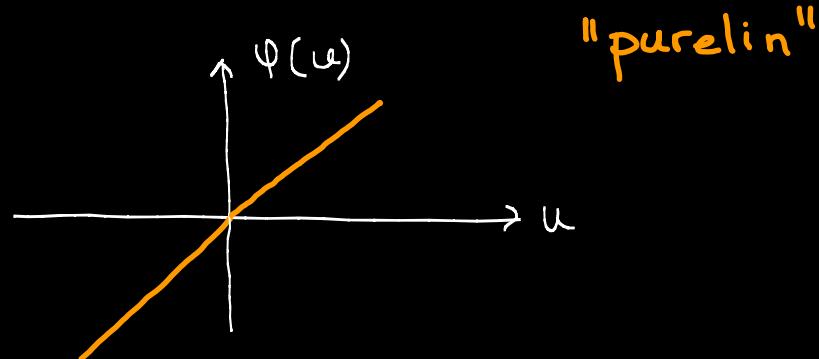


2) Paralel Lineer (Piecewise Linear) Fonksiyonlar:

Gök katmanlı ağların sıkıştırma katında kullanılabilek fonksiyonlardır.

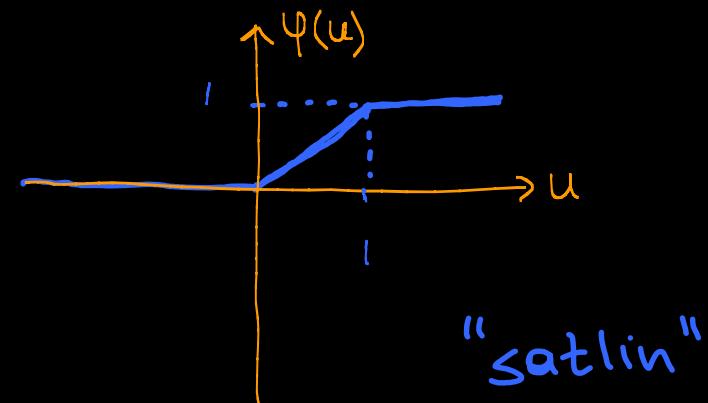
^{)} Pure Linear Function:

$$\Psi(u) = u$$



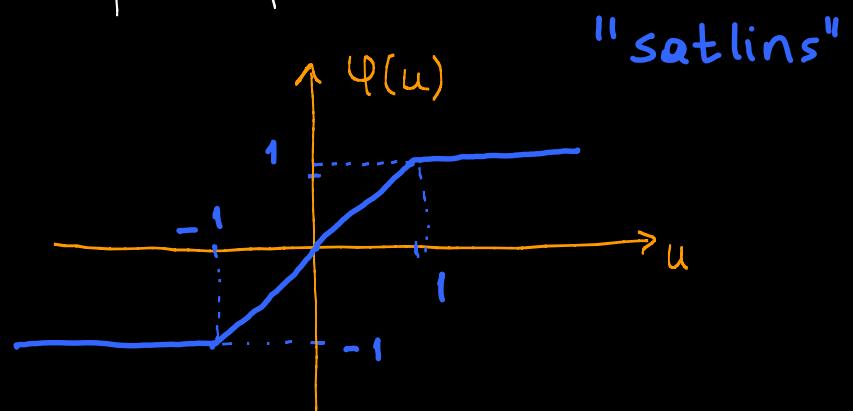
*) Doymalı Lineer (Saturating Linear) Fonksiyonlar:

$$\Psi(u) = \begin{cases} 0, & u < 0 \\ u, & 0 \leq u \leq 1 \\ 1, & u > 1 \end{cases}$$



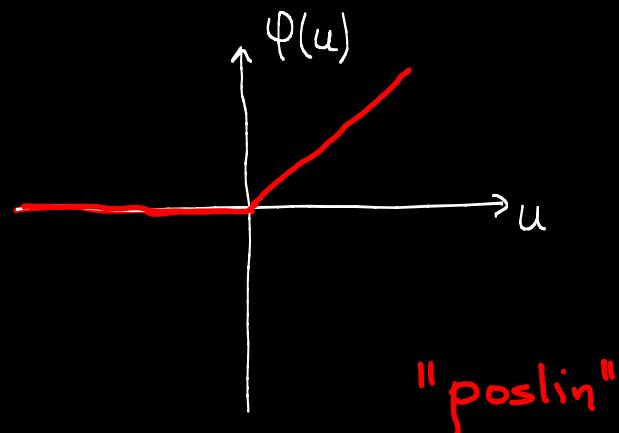
*) Simetrik Doymalı Lineer Fonksiyon:

$$\Psi(u) = \begin{cases} -1, & u < -1 \\ u, & -1 \leq u \leq 1 \\ 1, & u > 1 \end{cases}$$



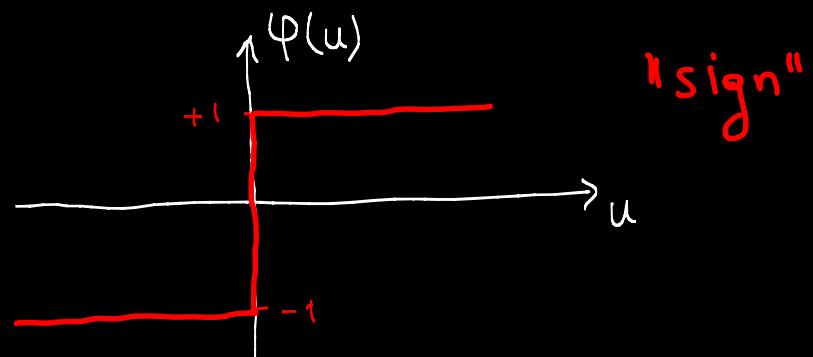
*¹) Pozitif Lineer Fonksiyonlar:

$$\psi(u) = \begin{cases} 0, & u < 0 \\ u, & u \geq 0 \end{cases}$$



*²) Signum Fonksiyonu:

$$\psi(u) = \begin{cases} -1, & u < 0 \\ 0, & u = 0 \\ 1, & u > 0 \end{cases}$$

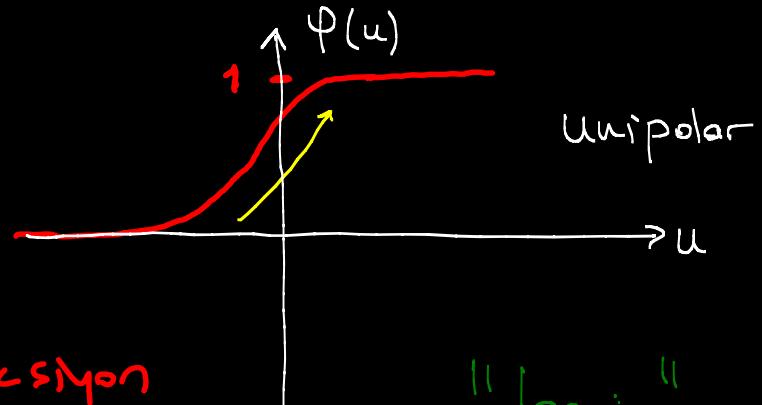


3) Sürekli Fonksiyonlar:

Sürekli fonksiyonlar, yapay sinir ağlarının özellikleri gizli katmanlarında kullanılır. En önemli nedeni türevi olabilmeler olmasıdır.

*) Sigmoid (Logaritmik Sigmoid) Fonksiyonlar.

$$\varphi(u) = \frac{1}{1 + e^{-\lambda u}}$$



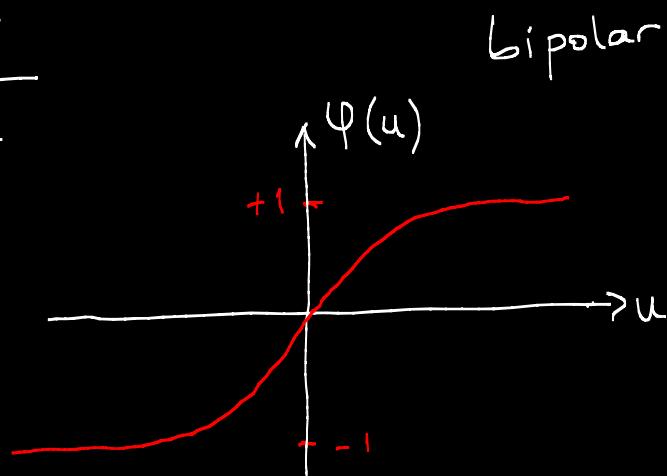
$\lambda \rightarrow \infty$ giderse sigmoid fonksiyon
hardlim fonksiyonu yakınsor.

"logsig"

*) Hiperbolik Tangent (Tangent Sigmoid) Fonksiyonlar:

$$\varphi(u) = \tanh\left(\frac{u}{2}\right) = \frac{1 - e^{-u}}{1 + e^{-u}}$$

"tansig"



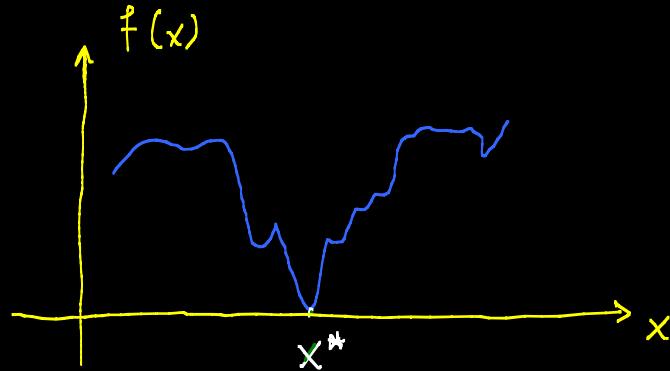
OPTIMIZASYON

Note Title

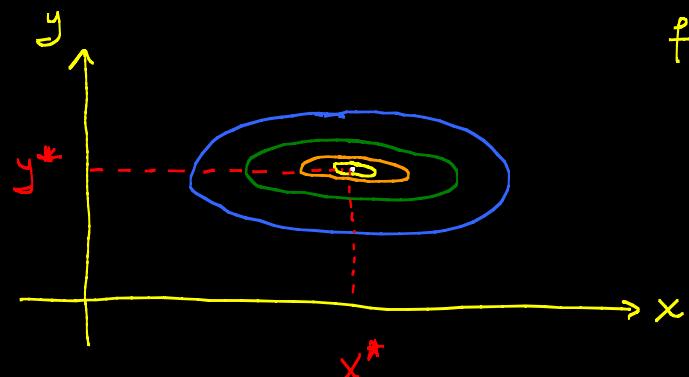
11.12.2020

Matematiksel Tanım: Verilen bir $f(x)$ fonksiyonunu minimum yapan x^* değer kinnesinin elde edilmesidir.

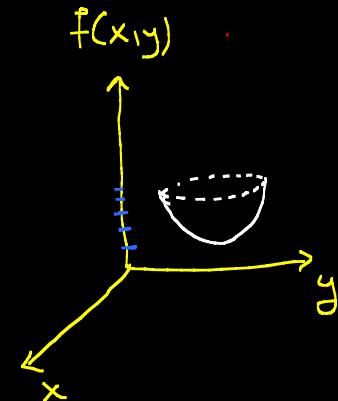
OPTIMIZASYON \leftrightarrow MINIMIZASYON



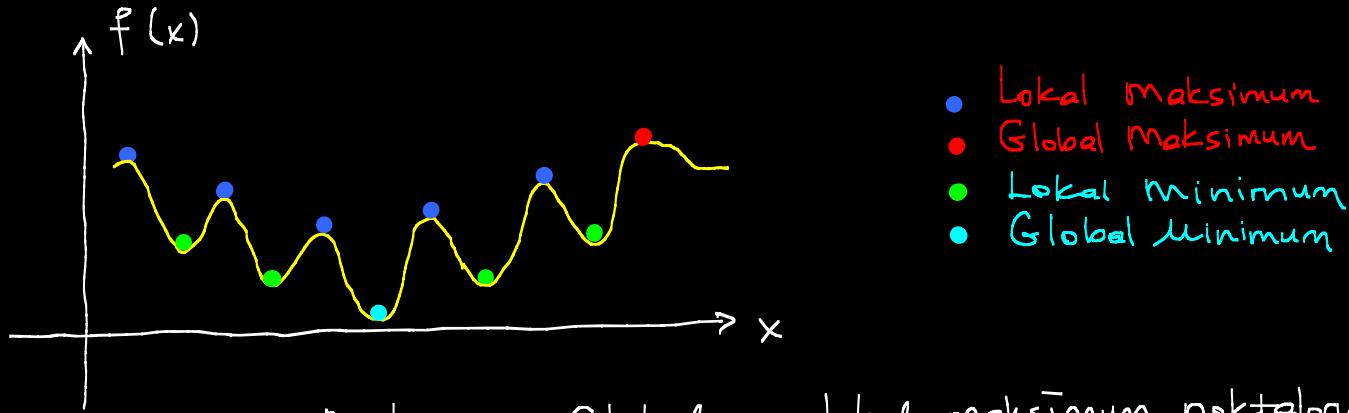
x^* $f(x)$ fonksiyonunu minimum yapan değerdir.



$f(x,y)$



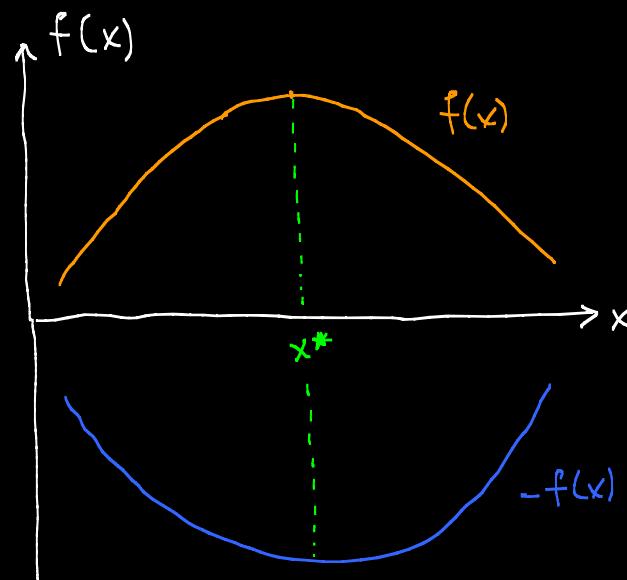
$X = \begin{bmatrix} x^* \\ y^* \end{bmatrix}$ noktası $f(x,y)$ fonksiyonunu minimum yapan değerdir.



maksima : Global ve lokal maksimum noktaların hepsi
 minima : Global ve lokal minimum noktaların hepsi

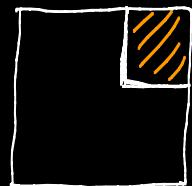
*)

\boxed{K} \rightarrow Kararlı Kısıtlı maksimize et
 $f(x)$



$$\max f(x) = \min -f(x)$$

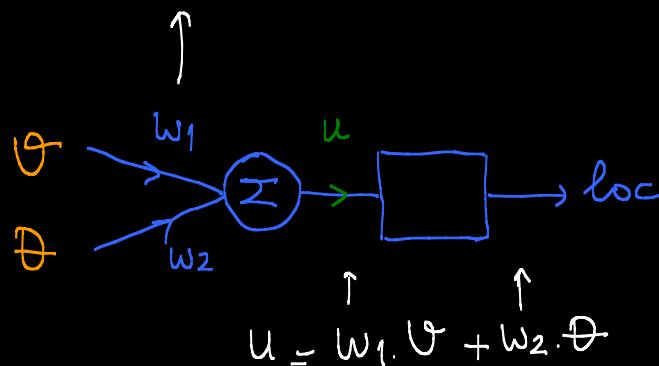
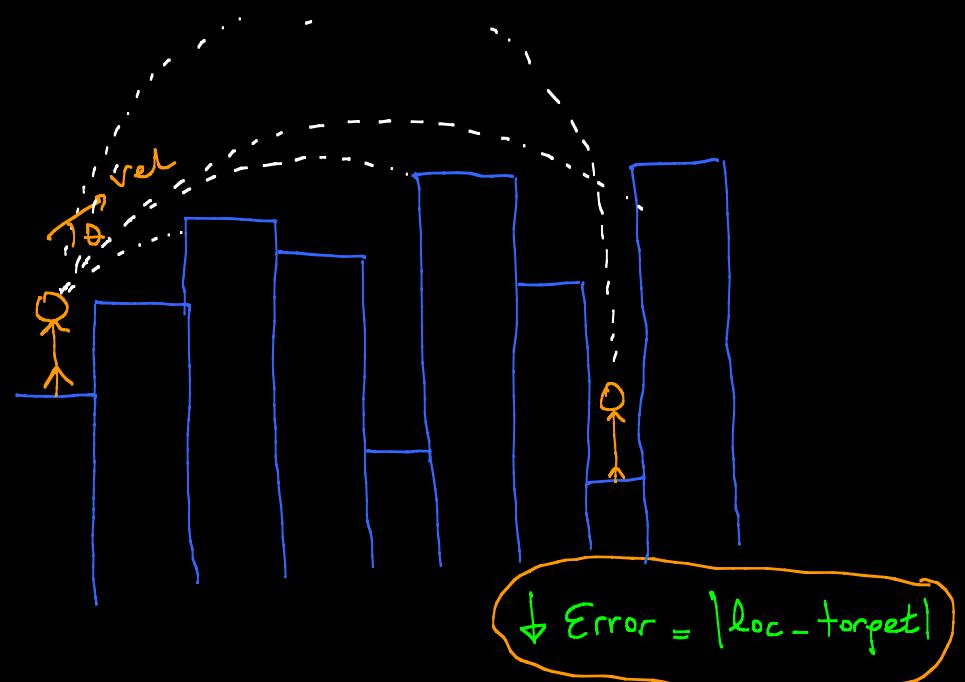
Mühendislik Tanımı : Belirli şartlar (kısıtlar) (kısıtlar) altında bir sistemi en iyi duruma getiren çözümün elde edilmesidir.



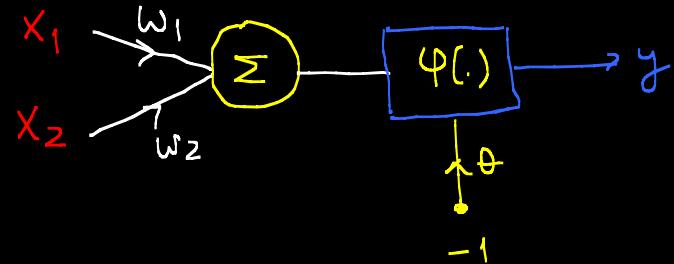
- $\rightarrow f = 5.8 \text{ GHz}$
- Katana \geq
- Yanlıslılığın
- Hızıne geri
- S.L.L
- B.G

Fiziksel kısıtlar : Tasarım Uzayı.

Performans kısıtları

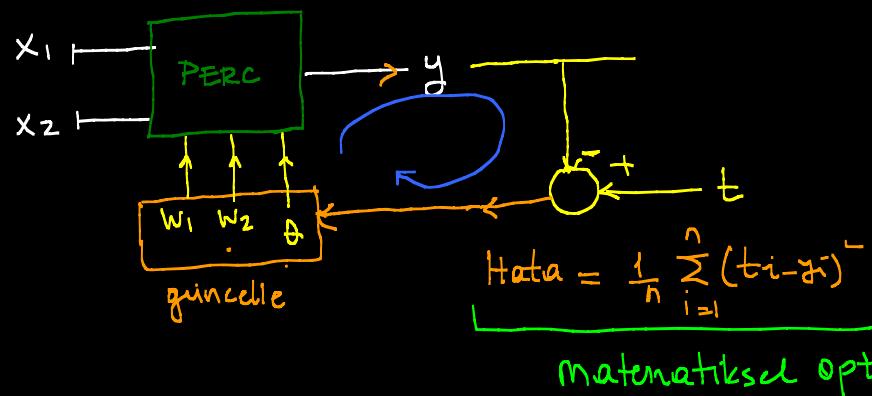
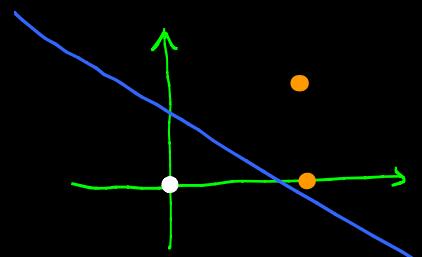


<i>x</i> ₁	<i>x</i> ₂	<i>t</i>	<i>y</i>
0	0	0	0
0	1	1	0
1	0	1	1
1	1	1	1



$$\text{Hata} = \frac{1}{n} \sum_{i=1}^n (t_i - y_i)^2 \Rightarrow \text{minimize edilecek fonksiyon.}$$

Aşağı hata} \rightarrow 0



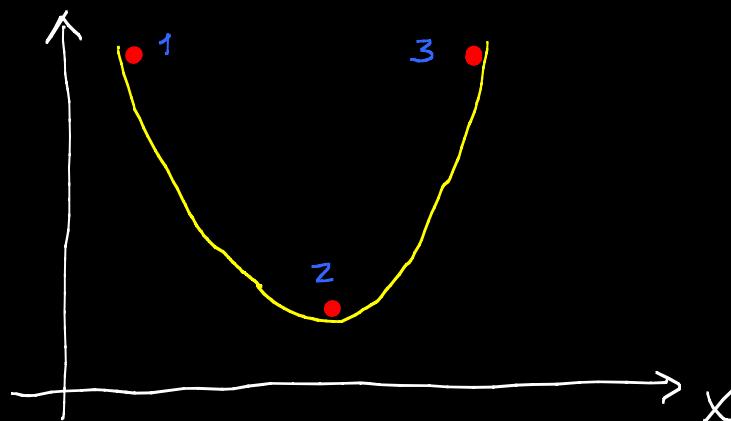
Optimizasyon Algoritmaları

- i) Matematiksel Opt. Yöntemleri : Tek baslangic noktasina sahiptir.
gradient, hessian matrisleri
1. ve 2. dereceden tane.
- ii) Sezgisel (dogadan ilham alin) optimizasyon yontemleri
: Goklu baslangic noktasina sahiptir.

: optimum noktası sosyal ve bilişsel bilgi kullanımına elde eder. Türev bilesi kullanma.

$$f(x)$$

ÖRNEK :



- 1 - türer →
- 2 - turev ≈ 0
- 3 - türev +

1) x_i ve η değerini gir.

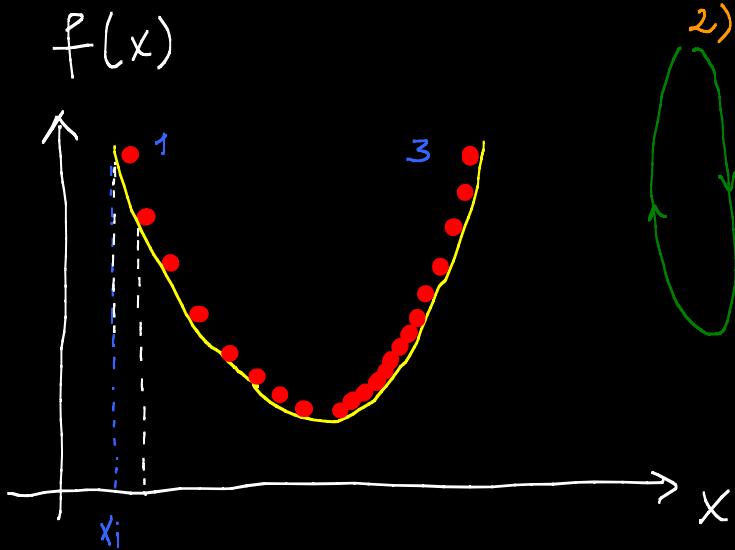
2) x_i noktasında türevi hesapla.

2.1) Eğer türev $< \epsilon$ ise

$$\text{sonuç} = x_i$$

Eğer türev $> \epsilon$ ise

$$x_i^{(t+1)} = x_i - \eta \cdot \frac{df}{dx} \Big|_{x=x_i}$$

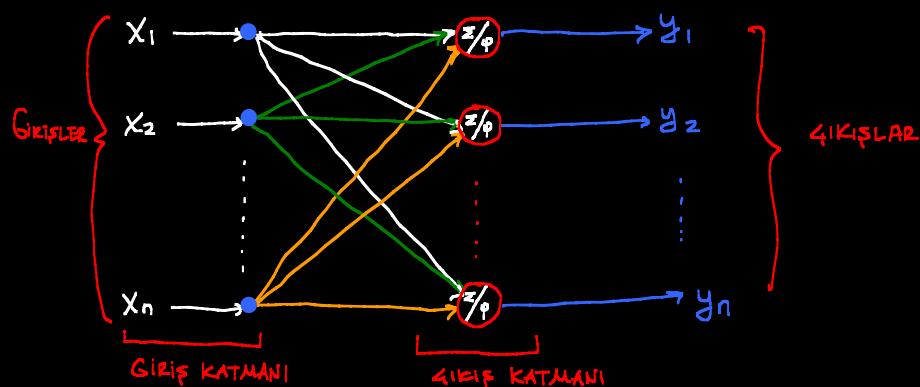


- Çok katmanlı algılayıcılar en yaygın kullanılan yapay sinir ağları modelidir.
- Çok katmanlı algılayıcılar ileri beslemeli ağlar sınıfına dahildir.

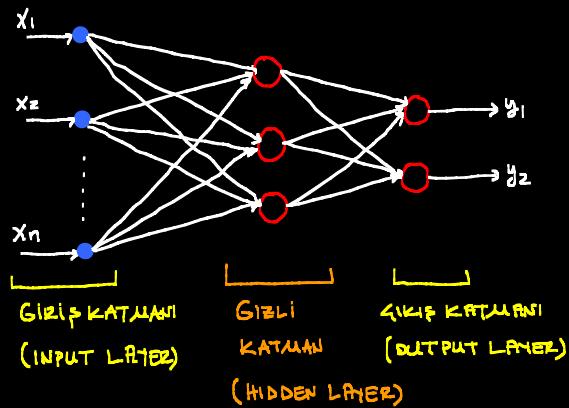
İLERİ BESLEMELİ AĞLAR (Feedforward Networks)

- Girişten çıkışa doğru bilgi akışı daima ileriye doğruudur.
- Tek katmanlı veya çok katmanlı olabilirler.
- Ağ içerisinde geribesleme yoktur.
- Bir katmandaki nöronlar bir önceki katmanda bulunan nöronlar tarafından beslenirler.

$$\Sigma/\phi$$



- * Yandaşı oğ yapısı tek katmanlı olarak kabul edilir. Katman sayısı nöron içeren katman sayısı ile belirlenir.
- * TEK KATMANLI \Rightarrow SINGLE LAYER
- * Bu oğ yapısı sadece linear sınırlar oluşturabilir.



4OK KATMAKLı AĞ (MULTI-LAYER)

- * En yaygın kullanılan ağı yapısıdır.
- * Karmazılı ve non-linear problemlerin çözümünde kullanılır.
- * Problemin karmazılığının boşa olarak gizli katman sayısını ve gizli katman sayısını artırmaktır.

$3 \times 10 \times 2$
 3-girişli 2-çıkışlı
 Tek gizli katman var.
 Gizli katmande 10 nöron var.

 $4 \times 15 \times 8 \times 2$
 4Girişli 2-çıkışlı
 2 gizli katman var.
 1. gizli katmande 15
 2. " 8
 nöron var.

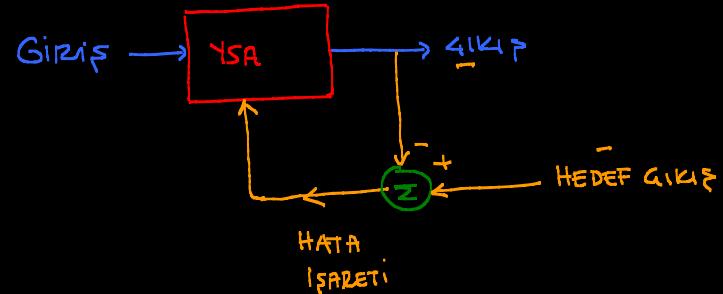
GİZLİ KATMAN, 19 ağız ile ağı ağızı arasında bulunan bilgiyi anlamlı hale getirir. İstensel yükün en yoğun olduğu katmandır. Gizli katman sayısının artışı, ağıın daha yüksek dereceli istatistiksel bilgiye erişimini sağlar.

- | Gizli katmanlarda genellikle tangent sigmoid veya logaritmik sigmoid aktivasyon fonksiyonları kullanılır.

YAPAY SINIR AĞLARININ EĞİTİLMESİ (ÖĞRENME)

- i) Öğrenme işlemi
 - ↳ Öğrenme yöntemleri
 - ↳ Öğrenme kuralları
- ii) Öğrenme yöntemleri
 - ↳ Egitsicili (SUPERVISED) Öğrenme
 - ↳ Egitsicisiz (UNSUPERVISED) Öğrenme
 - ↳ Zorlamalı (REINFORCEMENT) Öğrenme
- iii) Öğrenme kuralları
 - ↳ Hata düzeltme li (error correction)
 - ↳ Boltzman
 - ↳ Yarızımsız
 - ↳ Hebb

Eğitsicili öğrenme:

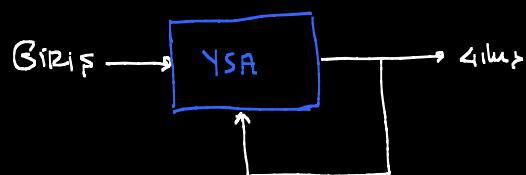


- Ağı, giriş verilerine karşılık akıması gereken hedef çıkış bilgilerine sahiptir.
- Amaç hatalı minimize ederek girişle hedef çıkış arasında ilişkisi matematiksel olarak modelleme
- En yaygın kullanılan öğrenme yöntemidir

- ↓
- Sınıflandırma (Classification) , Tanıma (Recognition) , Tespit etme (Detection) gibi problemlerde kullanılır.

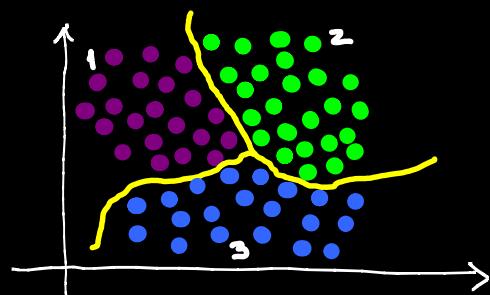
D	M	M	M	K	D	K	D	K	D	K
●	●	●	●	●	●	●	●	●	●	●

Eğitsiz Öğrenme:

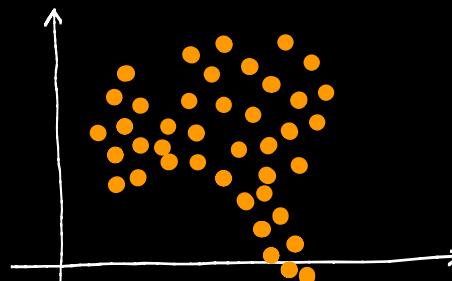


- ağ herdef ailmə bilgisine sahip deyildir.
Sadece giriş verisi tanınır.
- ağ kendi içindəki özelliklərə baxaraq
ortak özellik arar.
- Ağrılıqla KÜMELEMƏ (CLUSTERING)
işləmlərində kullanılır. (əgər ürettiiyi)
- Ağrılıklar qeyncəllərken sadece ailmə
bilgisine baxılır.

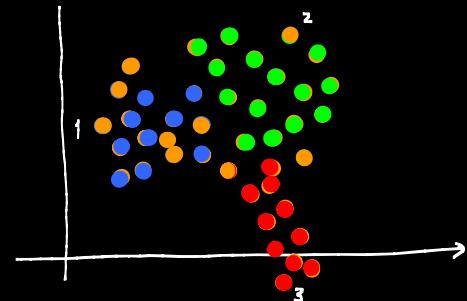
CLASSIFICATION



CLUSTERING

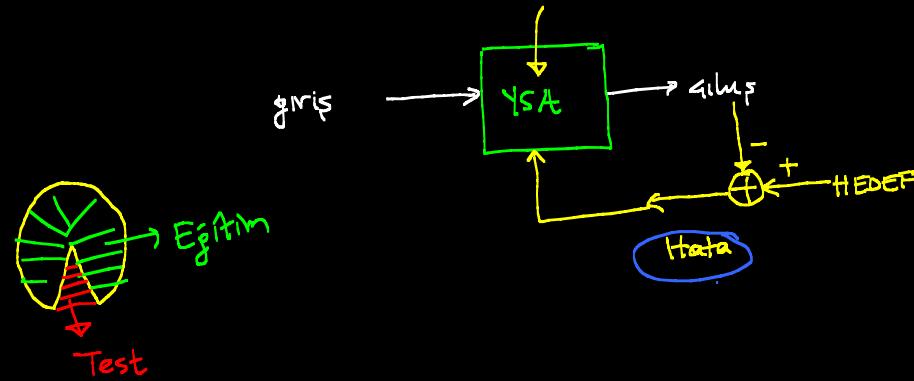


k-means clustering



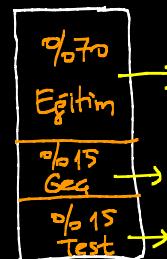
k-means clustering

- Eğiticiili öğrenme
- Eğiticisi ile öğrenme → kümelenme (hedef silme bilgisine sahip değil)
- Takvimeli öğrenme
- Kendi-kendine öğrenme



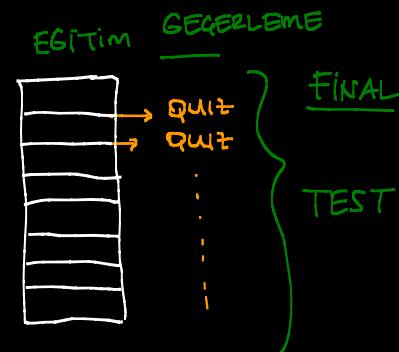
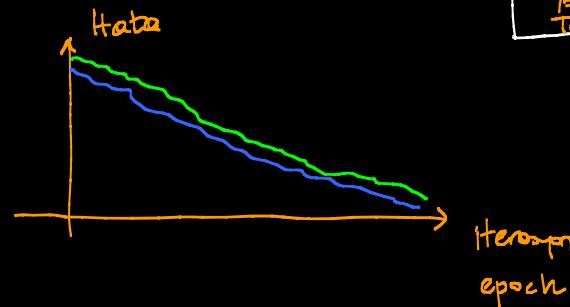
YSA'da teneel olarak üç faz bulunmaktadır:

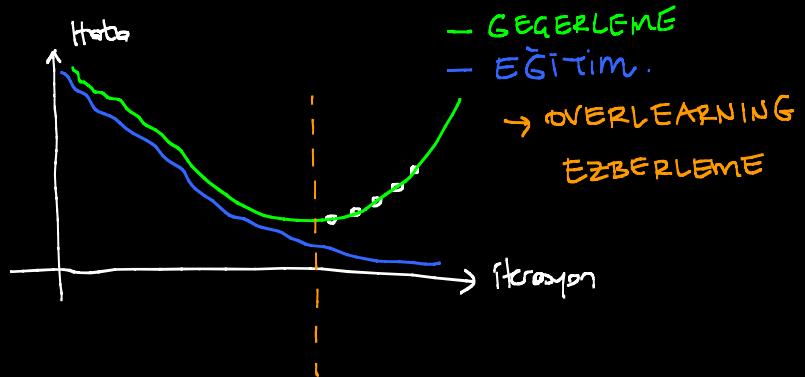
- 1) Eğitim (Training)
 - 2) Geçerlene (Validation)
 - 3) Test (Test)
- Eğitim aranızda
Eğitim bitinden sonra -



w ve θ sadece
eğitim verisi ile güncellentir.

Eğitim, geçerlene ve
test verileri mutlaka
birbirlerinden farklı
olmalıdır.





Eğiticiyi
öprene



- Toplam verinin büyük bir kısım eğitilmə formada kullanılır (%70)
- Gecikmə ve Test formada kalıcı kılınanları verir kullanır.

Təkrarıyla
öprene



- Oyular ve simulyapalar

Kendi-kendine
öprene



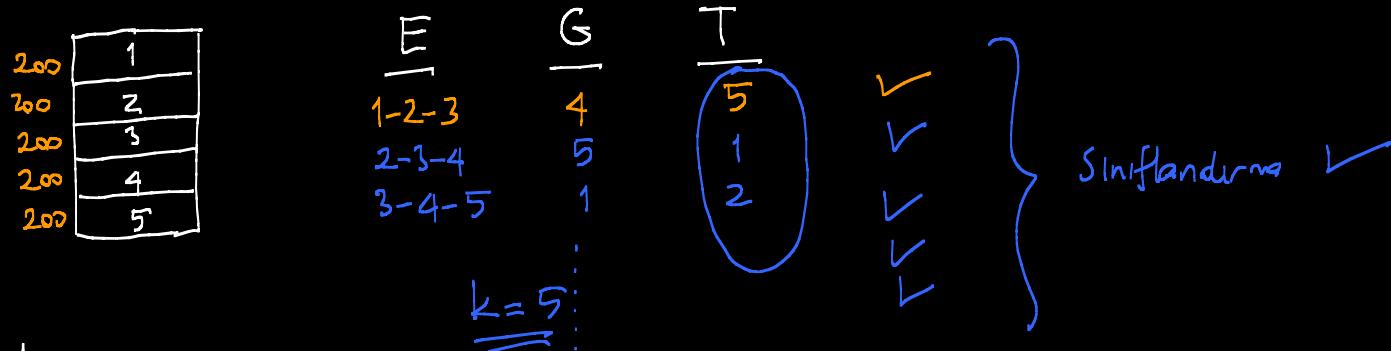
- NLP natural language processing

10sn. 300 frame

**SELF-SUPERVISED
LEARNING**



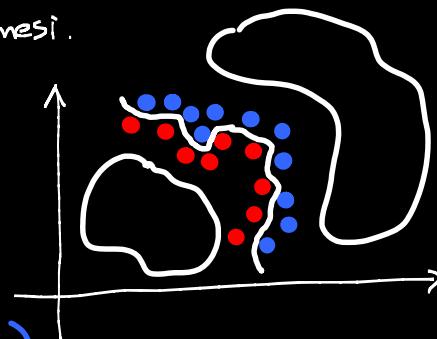
k -fold cross validation (k -siniflu yapay gecitlene)



YSA'da temel amaciımız

1) Eğitim aşamasında eğitim ve gecitlene heterozini minimizeden bir yapı oluşturtmak

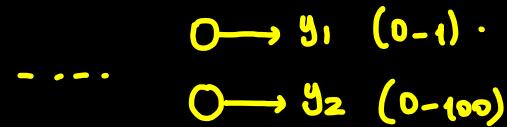
2) Bu ağır önceleri \rightarrow korelasyonlu test verisindeki lençer bozukluğunu göstermesi.



x	y	z	a	b		$\%1$	$\frac{100}{99}$	1	$\frac{\%2}{0.8}$
100			1						
90			0.9						
10			0.1						

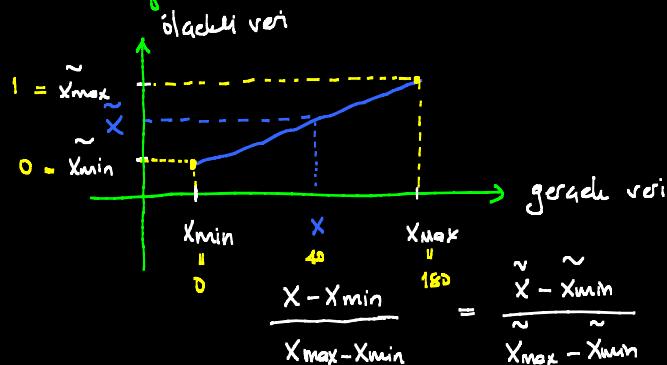
$\overline{10.1} \quad \overline{1.2}$

$2 \times 4 \times 2$



YSP'de çıkışların enzit derecede hata minimizasyonuna dahil edilmesi ve çıkışların genellikle bir zakıde minimize edilmesi için çıkış verileri 0-1 aralığına ölçeklendirilir. Bu isleme veri ölçeklendirme denir. En genel veri ölçeklendirme yöntemi ise "LINEER OLÇEKLENDİRME" dir.

Veri aralığı 0-100 arasında olan vertiği 0-1 aralığına ölçeklendirilmek



$$\tilde{x} = \tilde{x}_{\min} + (\tilde{x}_{\max} - \tilde{x}_{\min}) \cdot \left(\frac{x - x_{\min}}{x_{\max} - x_{\min}} \right) \quad \text{ÖLÇEKLENDİRME}$$

$$x = x_{\min} + (x_{\max} - x_{\min}) \cdot \left(\frac{\tilde{x} - \tilde{x}_{\min}}{\tilde{x}_{\max} - \tilde{x}_{\min}} \right) \quad \text{GERİ ÖLÇEKLENDİRME}$$

Veri ölçeklendirirken hangi aralığa ölçeklendirileceği
gizli katmandan kullanılan aktivasyon fonksiyonları göre seçilmesi daha
mantıklı olacaktır.