

## **DENEY NO:7**

### **Lineer Ayrılabilen Paternlerin Yapay Sinir Ağı ile Sınıflandırılması**

#### **Deneyin Amacı:**

Yapay zeka biliminin araştırma alanlarından biri olan Yapay Sinir Ağları (YSA), tahmin, sınıflandırma, veri ilişkilendirme, veri yorumlama bilgisayarların öğrenmesine yönelik çalışmaları kullanılmaktadır. Bu deneyde YSA ile Matlab programı ile harfların tek-katmanlı bipolar ayırık perceptronun patern sınıflayıcı olarak kullanılması deneyi gerçekleştirilecektir.

#### **1. Biyolojik Sinirin Yapısı**

YSA(Yapay Sinir Ağı) biyolojik sinir ağlarından esinlenerek geliştirilmiş bilgi işlem sistemleridir. Bir sinir ağı, bilgiyi depolamak için doğal eğilimi olan basit birimlerden oluşan paralel dağıtılmış bir işlemcidir. Bu ağlar, sinaptik ağırlıklar olarak bilinen nöronlar arası bağlantı kuvvetlerini, bilgiyi depolamak için kullanırlar. Benzer olarak YSA da birçok işlemciden oluşur ve farklı biçimlerde ifade edilebilen nümerik verileri taşıyan bağlantı yada ağırlıklarla birbirine bağlı yapılardır. Biyolojik sinir ağlarında girdi sinyallerini alan, yorumlayan ve uygun çıktıyı

iletken temel işlemci nöron olarak adlandırılır. Bir nöron, gövde (soma), gövdeye giren sinyal alıcıları (dendrit) ve gövdeden çıkan sinyal iletilici (akson) olmak üzere üç kısımdan oluşur[1].

a)Eklemler: Geçmiş deneyimlerin (bilgi tabanı) saklandığı alandır. Diğer sinirlerden bilgileri alır ve onları ağırlıklandırarak somaya iletir.

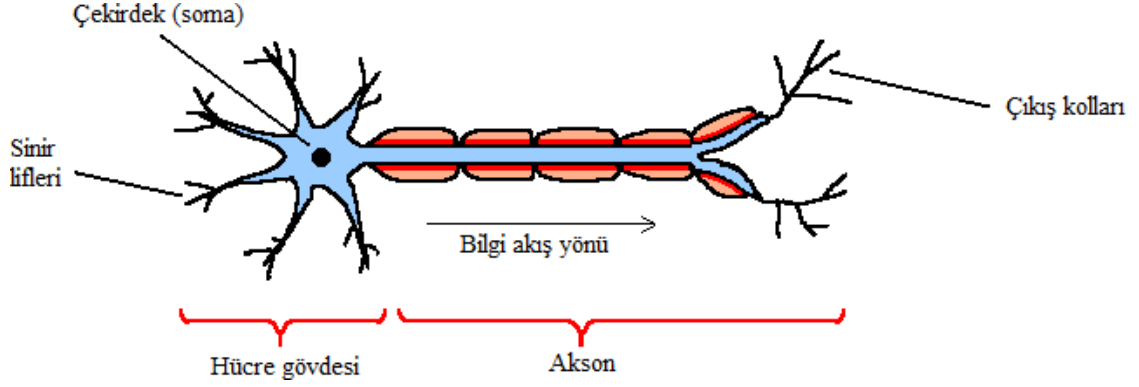
b) Soma: Sinirin gövdesidir. Eklemlerden gelen ağırlıklandırılmış bilgileri alır, onları toplar, eşiklendirir ve genelde doğrusal olmayan bir aktivasyon fonksiyonundan geçirir. Eklemlerden gelen ağırlıklandırılmış girişlerin toplamı belirli bir eşik değerini aşarsa soma bir çıkış işareti üretir.

c) Akson: Somanın ürettiği işareti diğer sinirlere ileten bir fiber yoldur.

Sinirler, çözölmüş kimyasal iyonlar içeren akışkanlarla doldurulmuş ve kuşatılmıştır. Bu kimyasal iyonlar sodyum ( $Na^+$ ), kalsiyum ( $Ca^{++}$ ), potasyum ( $K^+$ ) ve klördür ( $Cl^-$ ).  $K^+$  iyonları sinir hücresinin içinde yoğun olarak bulunurken,  $Na^+$  iyonları ise hücre membranının dışında yoğunlaşmıştır.  $Na^+$  ve  $K^+$  iyonları sinir darbesi adı verilen aktif sinir cevabının üretilmesinden sorumludur.

Biyolojik sistemlerde öğrenme, nöronlar arasındaki sinaptik (synaptic) bağlantıların ayarlanması ile olur. İnsanlar doğumlarından itibaren bir yaşayarak öğrenme süreci içerisine girerler. Bu süreç içinde beyin sürekli bir gelişme göstermektedir. Yaşayıp tecrübe ettikçe sinaptik bağlantılar ayarlanır ve hatta yeni bağlantılar oluşur. Bu sayede öğrenme gerçekleşir. Bu durum YSA için de

geçerlidir. Öğrenme, eğitime yoluyla örnekler kullanarak olur; başka bir deyişle, gerçekleşme girdi/çıkı verilerinin işlenmesiyle, yani eğitime algoritmasının bu verileri kullanarak bağlantı ağırlıklarını (weights of the synapses) bir yakınsama sağlanana kadar, tekrar tekrar ayarlamasıyla olur[2].



Şekil 1. Biyolojik sinirin yapısı

İnsan beyni sinir adı verilen yaklaşık  $10^{11}$  hesaplama elemanından oluşur. Bu sinirlerin her biri  $10^4$  tane sinir eklemine sahiptir. Merkezi sinir sistemi reseptörlerden aldığı bilgileri kendi içinde işler ve efektörleri kumanda ederek çeşitli aksiyonlar biçiminde cevaplar üretir.

## 2. Yapay Sinir Modeli

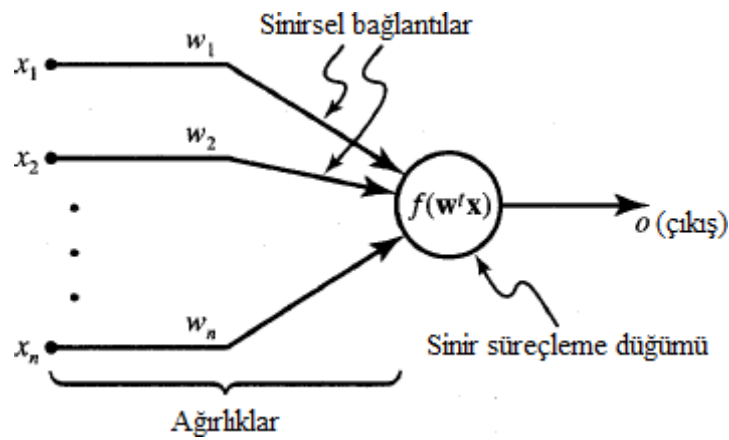
Son zamanlarda, veri madenciliği, değişkenler veya nedenler arasındaki korelasyonun altını çizme ve nihayetinde performansın tahmin edilmesi gibi karmaşık iş durumlarını inceleme potansiyelini ortaya çıkarmıştır. Talebi tahmin etmek için Yapay sinir ağları (YSA) en çok kullanılan makine öğrenme tekniğidir. Örnek olarak turistlerin geçmiş varış verilerini veya oda doluluk verilerini kullanarak turizm veya otel talebini tahmin etmek için YSA'yı kullanmıştır[3].

Yapay zekâ teknolojisinde önemli bir yeri olan YSA tekniği insan beyninin çalışma şeklini taklit eden bir çalışma yapısına sahiptir. YSA tekniği çok sayıda değişkenle çalışarak bunları öğrenme ve genelleme yapabilme yeteneğine sahiptir. YSA, gerçek dünyadaki karmaşık sistemlerde doğrusal olmayan girdi çıktı arasındaki ilişkilerini temsil etmektedir ve bunu örneklerle öğrenebilir[4].

Girişler ( $x_1, x_2, \dots, x_n$ ) : Giriş katmanındaki hücreler için, kullanıcı tarafından örnekler ile oluşturulmuş veri kümesidir. Diğer katmandaki hücreler için, herhangi bir katmandaki hücrenin çıkışı olabilir.

Ağırlıklar ( $w_1, w_2, \dots, w_n$ ) : Girişlerin, çıkışa ne oranda aktarılacağını gösterir. Örneğin  $w_1$  ağırlığı,  $x_1$  girişinin, çıkışa olan etkisini göstermektedir.

Toplama Fonksyonu: Bir hücrenin net girdisini hesaplamak için kullanılır. Bu amaç ile değişik fonksiyonlar kullanılmaktadır. En fazla tercih edilen, ağırlıklı toplam fonksiyonudur[2]. Giriş hücrelerindeki bilgiler ara katman hücrelerine aktarılırken ilgili ağırlıklar ile çarpılmakta ve ağırlıklı toplam fonksiyonuna göre her ara katman hücresinin NET girdisi hesaplanmaktadır[4]. Toplama Fonksyonu: Bir hücrenin net girdisini hesaplamak için kullanılır. Bu amaç ile değişik fonksiyonlar kullanılmaktadır. En fazla tercih edilen, ağırlıklı toplam fonksiyonudur[2]. Giriş hücrelerindeki bilgiler ara katman hücrelerine aktarılırken ilgili ağırlıklar ile çarpılmakta ve ağırlıklı toplam fonksiyonuna göre her ara katman hücresinin NET girdisi hesaplanmaktadır[4]. Her sinir modeli çok sayıda sinirsel giriş bağlantıları ve tek bir çıkışı olan bir süreçleme elemanından ibarettir; Şekil 2.



Şekil 2. Girişler, ağırlıklandırıcılar, ve süreçleyiciden oluşan bir sinirin genel sembolü.

Sinirin çıkış işareti aşağıdaki bağıntı ile verilir.

$$o = f(\mathbf{w}^t \mathbf{x}) \quad (1)$$

veya

$$o = f\left(\sum_{i=1}^n w_i x_i\right) \quad (2)$$

Burada  $\mathbf{w}$  ağırlık vektörüdür ve

$$\mathbf{w} \triangleq [w_1 \ w_2 \ \dots \ w_n]^t$$

ile tanımlanır ve  $\mathbf{x}$  ise giriş vektörü olup aşağıdaki gibi verilir.

$$\mathbf{x} \triangleq [x_1 \ x_2 \ \dots \ x_n]^t \quad (3)$$

Ağırlıklarla giriş vektörünün skaler çarpımı net olarak adlandırılır.

$$NET = \sum_{i=0}^n w_i^t x_i \quad (4)$$

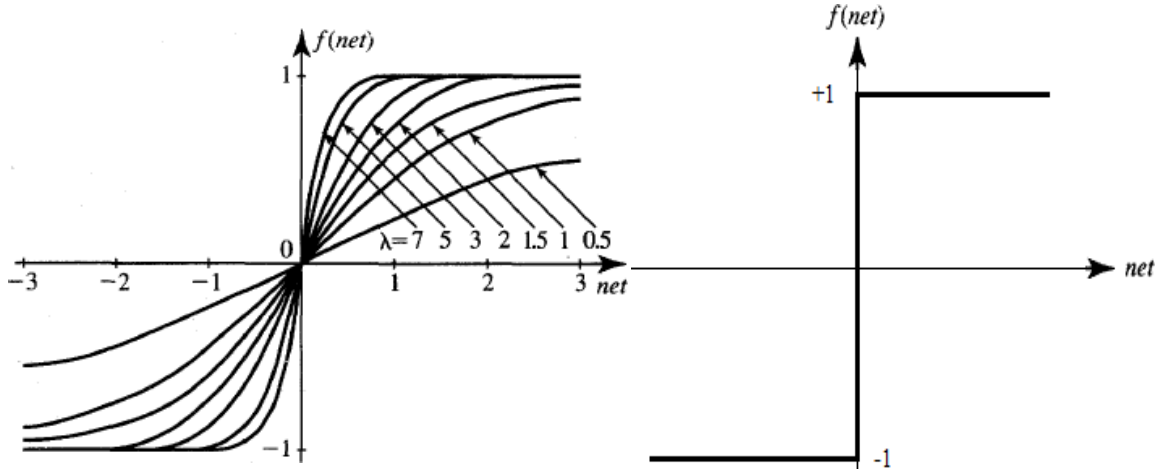
$(\mathbf{w}^t \mathbf{x})$  fonksiyonu aktifleme fonksiyonu olarak bilinir ve net argümenti biyolojik sinirin mebran potansiyeline benzer.

Sinir, süreçleme düğümü olarak, kendi ağırlıklandırılmış girişlerinin toplanması işlemini veya net'i elde etmek için skaler çarpma işlemini gerçekleştirir. Ardından, kendi aktivasyon fonksiyonu sayesinde  $f(\text{net})$  lineer olmayan işlemini yapar. Kullanılan aktivasyon fonksiyonları

$$f(\text{net}) \triangleq \frac{2}{1 + \exp(-\lambda \text{net})} - 1 \quad (5)$$

$$f(\text{net}) \triangleq \text{sgn}(\text{net}) = \begin{cases} +1, & \text{net} > 0 \\ -1, & \text{net} < 0 \end{cases} \quad (6)$$

$\lambda > 0$  katsayısı  $\text{net} = 0$  yakınında  $f(\text{net})$  sürekli aktivasyon fonksiyonunun derinliğini belirleyen sinir kazancı ile orantılıdır.  $\lambda \rightarrow \infty$  için sürekli aktivasyonun limiti (6) bağıntısında tanımlanan  $\text{sgn}(\text{net})$  fonksiyonu haline gelir. (5) ve (6) bağıntıları sürekli ve ayrık bipolar aktivasyon fonksiyonları gösterir; şekil 3.



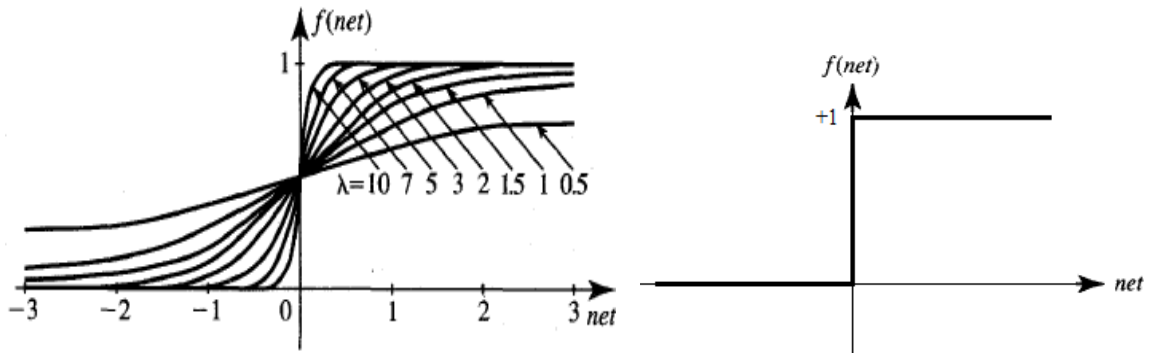
Şekil 3. Sürekli ve ayırık bipolar aktivasyon fonksiyonları

Aktivasyon fonksiyonları tek kutuplu olabilir, bu durumda sinirlerin çıkışları sadece pozitif değerler alır.

$$f(net) \triangleq \frac{1}{1 + \exp(-\lambda net)} \quad (7)$$

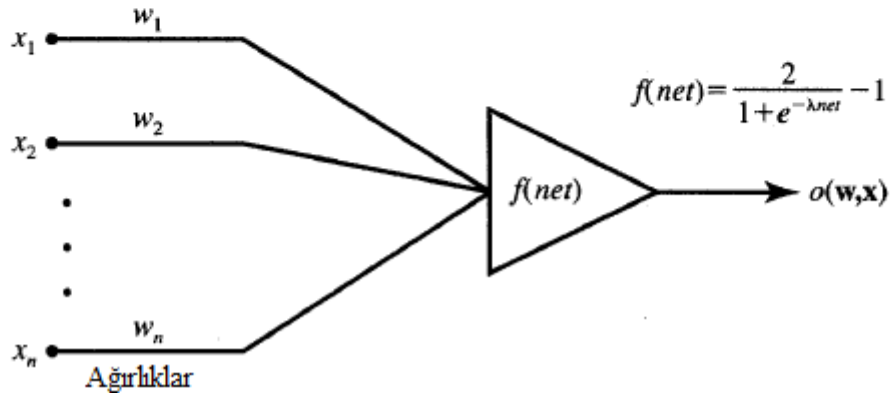
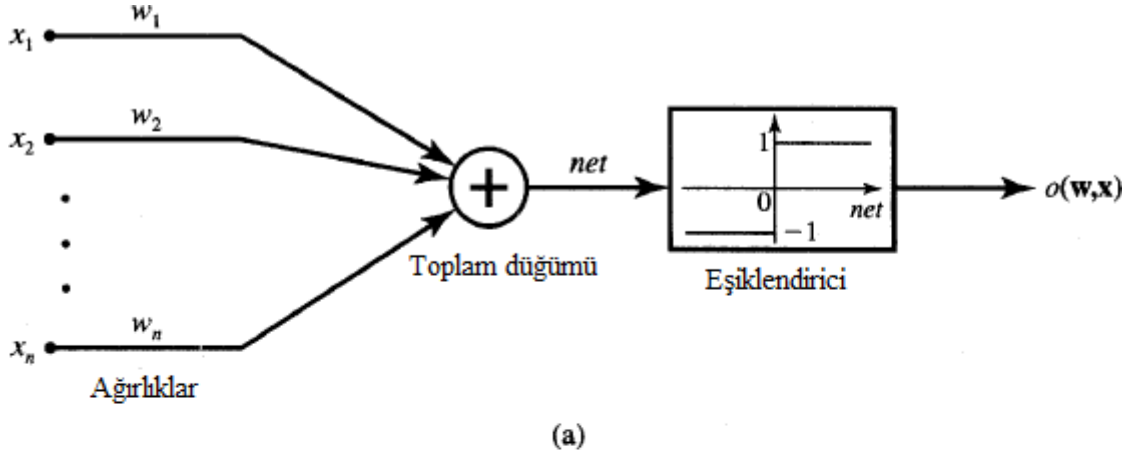
$$f(net) \triangleq \begin{cases} 1, & net > 0 \\ 0, & net < 0 \end{cases} \quad (8)$$

(7) ve (8) bağıntılarına karşı düşen aktivasyon fonksiyonları şekil 4'te gösterilmiştir



Şekil 4. Tek kutuplu sürekli ve ayırık aktivasyon fonksiyonları

Ayrık ve sürekli aktivasyon fonksiyonuna sahip sinirlerin sembolik gösterimi şekil 5'te gösterilmiştir.



Şekil 5. Ayırık ve sürekli aktivasyon fonksiyonuna sahip sinirlerin sembolik gösterimi

### 3. Yapay Sinir Ağ Modelleri

Yapay sinir ağı, sinirlerin kendi aralarında bağlantısı olarak tanımlanabilir. Sinir çıkışları, ağırlıklar üzerinden, kendileri de dâhil olmak üzere tüm diğer sinirlere hem gecikmesiz hem de gecikmeli bağlantılara izin verecek şekilde bağlanarak bir ağ oluşturulabilir. Ağ mimarileri genelde ileri beslemeli ağlar ve geri beslemeli ağlar olmak üzere iki sınıfa ayrılabilir.

#### 3.1. İleri Beslemeli Ağ

n girişli m sinirin ilkel ileri beslemeli mimarisi şekil 6'da gösterilmiştir. Çıkış ve giriş vektörleri sırasıyla aşağıdaki gibidir.

$$\mathbf{o} = [o_1 \ o_2 \ \dots \ o_m]^t \quad (9)$$

$$\mathbf{x} = [x_1 \ x_2 \ \dots \ x_n]^t$$

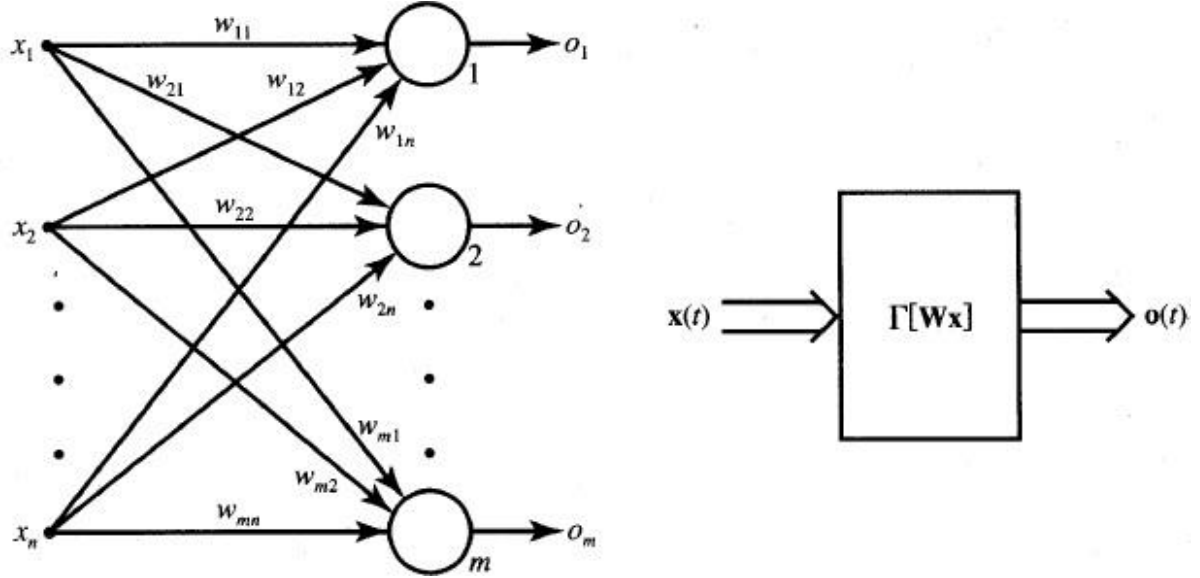
$w_{ij}$  ağırlığı i. siniri j. girişe bağlar, yani çift indislerden birincisi hedefi ikincisi de kaynak düğümünü işaret eder. O zaman i. sinirin aktivasyon değeri aşağıdaki gibi yazılabilir

$$net_i = \sum_{j=1}^n w_{ij}x_j, \quad \text{for } i = 1, 2, \dots, m \quad (10)$$

Ağıdaki  $m$  sinirin her biri tarafından gerçekleştirilen transformasyon aşağıdaki gibi ifade edilen güçlü bir doğrusal olmayan iz düşümdür

$$o_i = f(\mathbf{w}_i^t \mathbf{x}), \quad \text{for } i = 1, 2, \dots, m \quad (11)$$

Tek katmanlı ileri beslemeli ağıın ara bağlantı şeması Şekil 6’da verilmiştir.



Şekil 6. İleri beslemeli bir ağ modeli ve blok diyagramı

#### 4. Sinir Ağı Öğrenme Kuralları

Yapay sinir ağı, girişleri ile çıkışları arasındaki ilişkisi bilinen her sistemin yerini alabilir. Eğer bir sistemin girişi ile çıkışı arasındaki ilişkisi matematiksel olarak modellenilebiliyorsa, o zaman yapay sinir ağından ziyade matematiksel izdüşüm bağıntısının hesaplanması daha uygun olabilir. Yapay sinir ağı daha ziyade matematiksel olarak ifade edilemeyen ama girişlerine karşılık çıkışları bilinen sistemlere eğitim yoluyla istenilen giriş-çıkış izdüşüm yeteneğinin kazandırılması için kullanılır. Birçok farklı eğitim algoritması vardır. Burada karakter sınıflama işlemi yapılacağı için sınıflamaya yönelik bir ağıın seçimi ve eğitimi anlatılacaktır.

##### 4.1. Perceptron Öğrenme Kuralı

Bu deneyde iki karakterin sınıflamasını yapan bir ağıın tasarımı ve eğitimi anlatılacaktır. Bipolar ayrık aktivasyon fonksiyonuna sahip bir sinir +1 ve -1 olmak üzere iki çıkış değeri verdiği için bir sinirli ve tek katmanlı bir ağ iki farklı karakterin sınıflandırılması için uygun bir ağıdır.

Tek katmanlı böyle bir ağıın danışmanlı eğitimi için en uygun eğitim kuralı Perceptron Öğrenme kuralıdır, Şekil 7. Bu kural için  $r$  öğrenme işareti, arzulanan ile gerçek sinir cevabı arasındaki farktır. Bu yüzden öğrenme danışmanlıdır ve öğrenme işareti

$$r \triangleq d_i - o_i \quad (13)$$

değerine eşittir. Burada  $o_i = \text{sgn}(\mathbf{w}_i^t \mathbf{x})$  ve  $d_i$  ise arzulanan cevaptır. Şekil 7'den  $\Delta \mathbf{w}_i$  için

$$\Delta \mathbf{w}_i = c [d_i - \text{sgn}(\mathbf{w}_i^t \mathbf{x})] \mathbf{x}$$

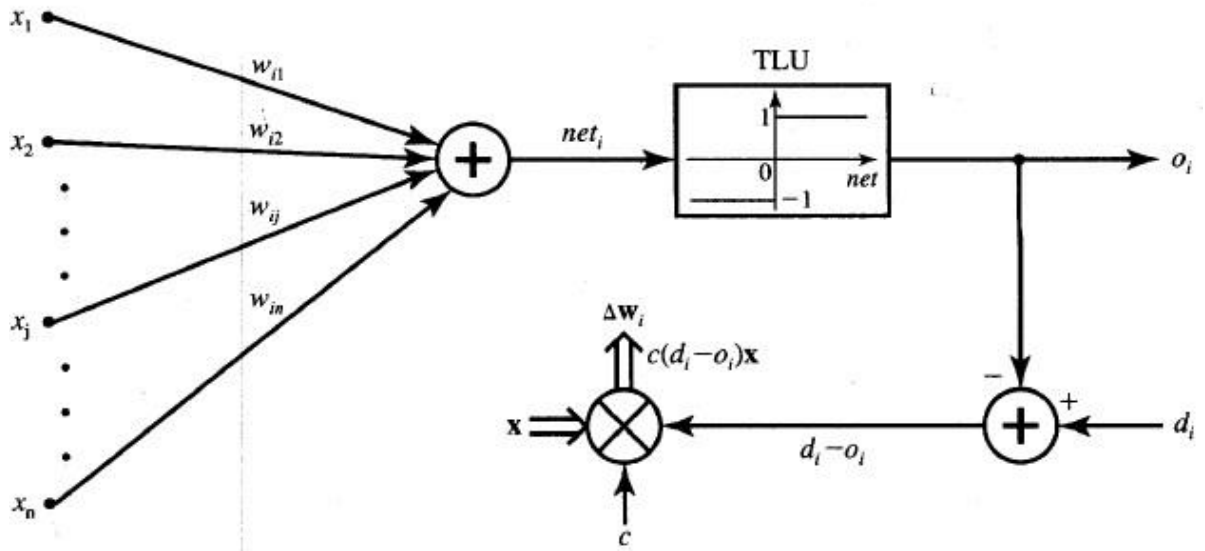
$$\Delta w_{ij} = c [d_i - \text{sgn}(\mathbf{w}_i^t \mathbf{x})] x_j, \quad \text{for } j = 1, 2, \dots, n$$
(14)

yazılabilir. Bu kuralda, yalnız  $o_i$  cevabı doğru değilse ağırlık ayarlaması yapılır.

Arzulanan cevap ya +1 veya -1 olduğundan ağırlık ayarı

$$\Delta \mathbf{w}_i = \pm 2c \mathbf{x}$$
(15)

bağıntısına indirgenir. Arzulanan ve gerçek cevaplar aynı ise ağırlık ayarı yapılmaz. Bu yöntemde ağırlıklar herhangi bir değere ilişkilendirilebilir.



Şekil 7. Perceptron öğrenme kuralı

## 5. Tek-Katmanlı Bipolar Ayrık Perceptronun Patern Sınıflayıcı Olarak Kullanılması

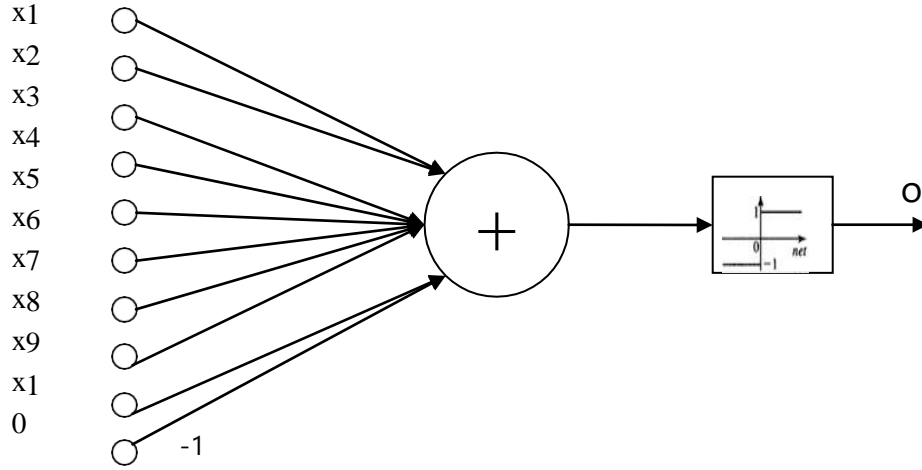
Şekil 8'de gösterilen L ve I karakterleri için bir sınıflayıcının tasarlanması ve eğitilmesi istenmektedir. Siyah pikseller 1 ve beyaz pikseller 0(sıfır) ile temsil edilmektedir. L karakteri için  $o=1$  ve I karakteri için de  $o=-1$  çıkışını üretecek eşik ile birlikte 10 girişe sahip ayrık bipolar perceptrondan oluşan bu sınıflayıcının ağırlıklarını eğitimle belirleyelim.

1	2	3
4	5	6
7	8	9

1	2	3
4	5	6
7	8	9

Şekil 8. L ve I karakterleri (3x3 boyutunda)





$$y_L = [1 \ 0 \ 0 \ 1 \ 0 \ 0 \ 1 \ 1 \ 1 \ -1]^T$$

$$y_I = [0 \ 1 \ 0 \ 0 \ 1 \ 0 \ 0 \ 1 \ 0 \ -1]^T$$

$$w^1 = w + \frac{c}{2} (d - o) y \quad w^1 = w \pm y, \quad o = \text{sgn}(w^T y) \quad w = 0$$

$$w^1 = w_{\text{initial}} = [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ -1] \text{ olsun.}$$

Adım 1:  $y_L$  giriftir,  $d_L = +1$  varsayılıyor

$$o = \text{sgn}(w^{1T} y_L) = +1 \text{ olduđundan } w^2 = w^1 \text{ olur.}$$

Adım 2:  $y_I$  giriftir ve istenen çıkış  $d_I = -1$  dir.

$$o = \text{sgn}(w^{2T} y_I) = \text{sgn}(1) = +1 \text{ olduđundan } w^3 = w^2 - y_I = [0 \ -1 \ 0 \ 0 \ -1 \ 0 \ 0 \ -1 \ 0 \ 0]$$

Adım 3:  $y_L$  giriftir.

$$o = \text{sgn}(w^{3T} y_L) = \text{sgn}(-1) = -1 \quad w^4 = w^3 + y_L = [1 \ -1 \ 0 \ 1 \ -1 \ 0 \ 1 \ 0 \ 1 \ -1]$$

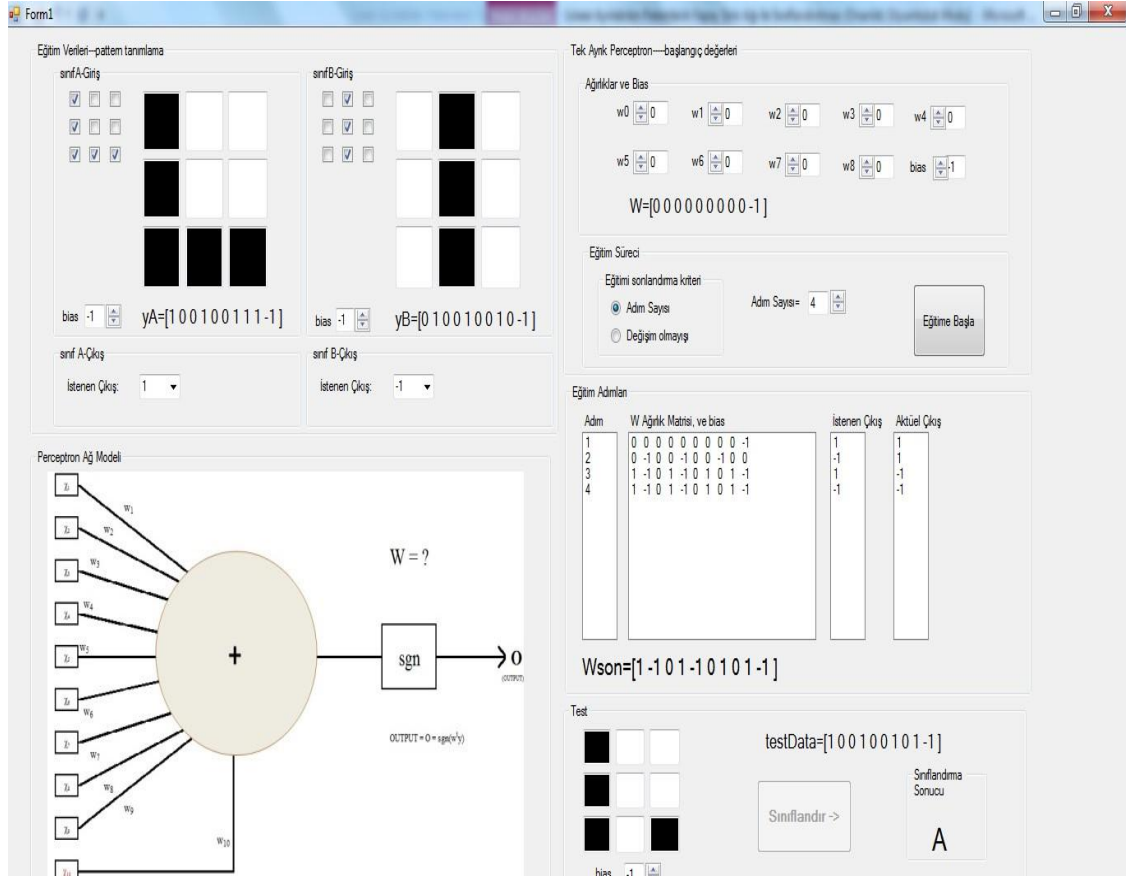
Adım 4:  $y_I$  giriftir.

$$o = \text{sgn}(w^{4T} y_I) = \text{sgn}(-1) = -1 \quad w^5 = w^4 \text{ olur.}$$

Eđitim bitmiřtir. L karakteri  $o = +1$  çıkışına ve I karakteri de  $o = -1$  çıkışına izdüşürölmüřtür.

$$W_{\text{son}} = [1 \ -1 \ 0 \ 1 \ -1 \ 0 \ 1 \ 0 \ 1 \ -1] \text{ olur.}$$

## 6. Pattern Sınıflandırma Programının Ekran Görüntüsü



Şekil 9. Pattern Sınıflandırma Programının Ekran Görüntüsü

## KAYNAKLAR

- [1] M. Hacıbeyoğlu, “Çoklu Etmen Mimarisi Ve Takviyeli Öğrenme,” *Chest*, vol. 25, no. 1, pp. 1–14, 2006.
- [2] M. Z. Yıldırım, A. Çavuşoğlu, B. Şen, and İ. Budak, “Yapay Sinir Ağları ile Ağ Üzerinde Saldırı Tespiti ve Paralel Optimizasyonu,” pp. 671–677, 2014.
- [3] A. Akgün, “Veri Madenciliği Sınıflandırma Yöntemlerinin Etkinliğinin Karşılaştırılması Olarak İncelenmesi: Seyahat Acentası Örneği,” *AKDENİZ ÜNİVERSİTESİ Sos. BİLİMLER ENSTİTÜSÜ DOKTORA TEZİ*, 2019.
- [4] M. KAYAKUŞ and M. TERZİOĞLU, “Yapay Sinir Ağları ve Çoklu Doğrusal Regresyon Kullanarak Emeklilik Fonu Net Varlık Değerlerinin Tahmin Edilmesi,” *BİLİŞİM Teknol. DERGİSİ*, 2021.
- [5] İstanbul Ticaret Üniversitesi Bilgisayar Mühendisliği Bölümü Bilgisayar Sistemleri Laboratuvarı Deney Föyü  
<http://ww3.ticaret.edu.tr/mckasapbasi/files/2015/09/Lineer-Ayrilabilen-Paternlerin-Yapay-Sinir-Agi-ile-Siniflandirilmasi2.pdf>