

4. YAPAY SİNİR AĞLARI

4.1 Giriş

Yapay sinir ağı ya da kısaca YSA; insan beyninin çalışma sisteminin yapay olarak benzetimi çabalarının bir sonucu olarak ortaya çıkmıştır. En genel anlamda bir YSA insan beynindeki birçok nöronun (sinir hücresinin), ya da yapay olarak basit işlemcilerin birbirlerine değişik etki seviyeleri ile bağlanması sonucu oluşan karmaşık bir sistem olarak düşünülebilir. Önceleri temel tıp birimlerinde insan beynindeki nöronların matematiksel modelleme çabaları ile başlayan çalışmalar, geçtiğimiz on sene içerisinde, disipline bir şekil almıştır. YSA bugün fizik, matematik, elektrik ve bilgisayar mühendisliği gibi çok farklı bilim dallarında araştırma konusu haline gelmiştir. YSA'nın pratik kullanımı genelde, çok farklı yapıda ve formlarda bulunabilen bilgi verilerini hızlı bir şekilde tanımlama ve algılama üzerinedir. Aslında mühendislik uygulamalarında YSA'nın geniş çaplı kullanımının en önemli nedeni, klasik tekniklerle çözümü zor problemler için etkin bir alternatif oluşturmaktır. Çünkü bilgisayarlar insanın beyinsel yeteneğinin en zayıf olduğu çarpma, bölme gibi matematiksel ve algoritmik hesaplama işlemlerinde hız ve doğruluk açısından yüzlerce kat başarılı olmalarına rağmen insan beyninin öğrenme ve tanıma gibi işlevlerini hala yeteri kadar gerçekleştirememektedir. Çizelge 4.1'de bilgisayar ile insan beyni arasındaki çalışma sistem yapısı karşılaştırmalı olarak verilmiştir.

Çizelge 4.1 Bilgisayar ile insan beyni arasındaki çalışma sistem yapısının karşılaştırılması

BİLGİSAYAR	İNSAN BEYNİ
Sayısal	Analog
Seri	Paralel
Komut Kümeli	Bilgiye Adapte Olma
Yanlış Hesaplamalar Sonucu Etkiler	Birimlerin Ana İşlemlere Etkisi Azdır
Giriş Verilerindeki Hatalar Sonucu Etkiler	Giriş Verilerindeki Hatalara Her Zaman Duyarlı Değil

4.2 YSA'nın Tanımı ve Modeli

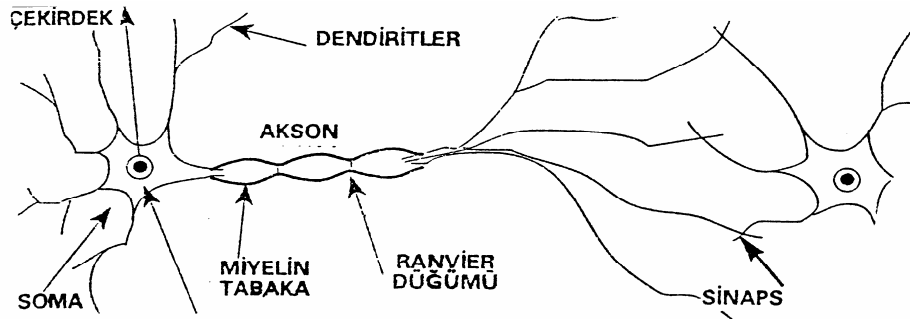
4.2.1 YSA'nın tanımı

YSA paralel dağılmış bir bilgi işleme sistemidir. Yani, YSA'nın temelinde, zeka gerektiren işlemlerden oluşan bilgi işleme işlevi vardır. Bu sistem tek yönlü işaret kanalları (bağlantılar) ile birbirine bağlanan işlem elemanlarından oluşur. Çıkış işareti bir tane olup isteğe göre çoğaltılabilir. YSA yaklaşımının temel düşüncesiyle, insan beyninin fonksiyonları arasında benzerlik vardır. Bu yüzden YSA sistemine insan beyninin modeli denilebilir. YSA çevre şartlarına göre davranışlarını şekillileyebilir. Girişler ve istenen çıkışların sisteme verilmesi ile

kendisini farklı cevaplar verebilecek şekilde ayarlayabilir. Ancak son derece karmaşık bir içyapısı vardır. Onun için bugüne kadar gerçekleştirilen YSA; biyolojik fonksiyonların temel nöronlarını örnek alarak yerine getiren kompozite elemanlar olmuştur.

4.2.2 Nöronun biyolojik yapısı ve nöron modeli

İnsanın bilgi işleme olayı beyinde gerçekleşir. Gerçektende en karmaşık sinir ağı Cerebral Cortex denilen “beyin”dir. Sinir sisteminin en basit yapısı nöronlardır. Beyinde yaklaşık olarak 10^{10} sinir hücresi vardır. Yine hücre başına bağlantı sayısı ise 10^4 mertebesindedir. Beyin için çalışma frekansı 100 Hz’dir. Fiziksel boyutları ise 1.3 kg ve 0.15 m^2 kesitlidir. Vücudun değişik yerleri ile bilgi alışverişi yapan nöron hücresidir. Şekil 4.1 de basit bir nöron hücresi görülmektedir.

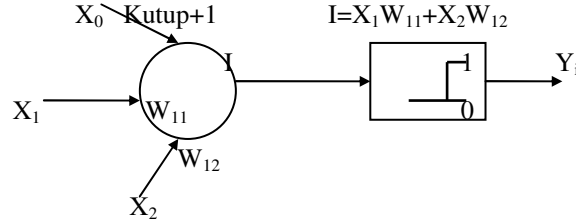


Şekil 4.1 Basit bir nöron yapısı

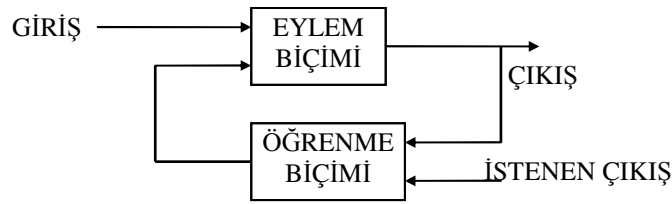
Nöron, soma adı verilen hücre gövdesi dentrit denilen kıvrımlı uzantılar ve somanın dalları sayesinde nöronu dallarına bağlayan tek sinir fiberli aksondan oluşur. Dendrit’ler hücreye gelen girişleri toplarlar. Dendrit tarafından alınan işaretler hücrede birleştirilerek bir çıkış darbesi üretilip üretilmeyeceğine karar verilir. Eğer bir iş yapılacaksa üretilen çıkış darbesi aksonlar tarafından taşınarak diğer nöronlarla olan bağlantılara veya terminal organlara iletilir. Beyindeki kortekste her nöronun bir karşılığı vardır. Bir nöronun çıkışı ona bağlı olan bütün nöronlara iletilir. Fakat korteks, işin yapılabilmesi için hangi nöron harekete geçirilecekse, sadece ona komut gönderir.

Somanın içinde ve çevresinde sodyum, kalsiyum, potasyum ve klor iyonları vardır. Potasyum yoğunluğu nöronun içinde, sodyum yoğunluğu dışındadır. Somanın zarı elektriksel olarak uyarılınca (söz konusu uyarı genellikle bir gerilim düşmesidir) zar, Na ve Ca gibi diğer iyonların içeri geçmesine izin verir ve somanın iç durumunu değiştirir. Nöronlar arasındaki bağlantılar hücre gövdesinde veya "sinaps" adı verilen dendritlerdeki geçişlerde olur. Yardımcı bir benzetme aksonlarla, dendritleri elektrik sinyallerini nörona ileten değişik empedansdaki yalıtılmış iletken olmasıdır. Sinir sistemi milyarlarca nöron ile tek bir nöronun çıkan aksonun 10000 kadar diğer nöronu bağlayan bir ağdır. Sinapslarla düzeltilen işaretleri

taşıyan aksonlar ve dendritlerle içiçe geçmiş nöronlar bir sinir ağı oluştururlar. Şekil 4.2'de en basit formda gösterilen nöron modeli, bir eşik birimi olarak algılanabilir. Şekil 4.3'de ise YSA'nın genel blok şeması gösterilmektedir.



Şekil 4.2 Nöron modeli

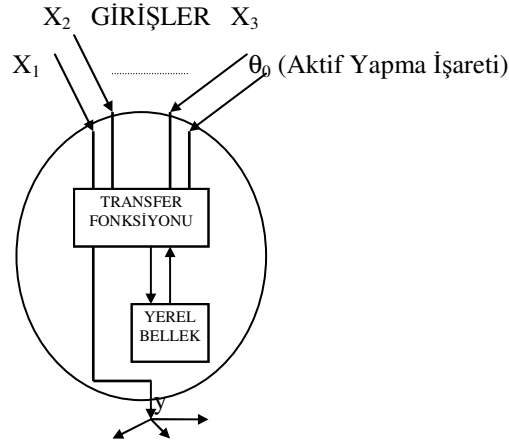


Şekil 4.3 YSA'nın genel blok şeması

Eşik birimi, çıkışları toplayan ve sadece girişin toplamı iç eşik değerini aştığında bir çıkış üreten işlem elemanıdır. Bir eşik birimi olarak nöron sinapslarındaki işaretleri alır ve hepsini toplar. Eğer toplanan işaret gücü eşiği geçecek kadar güçlü ise diğer nöronları ve dendritleri uyaran akson boyunca bir işaret gönderilir. Kesişen dendritlerden gelen sinapslarla kapılanan bütün işaretleri soma toplar. Toplam işaret daha sonra nöronun iç eşik değeri ile karşılaştırılır ve eşik değerini aşmışsa aksona bir işaret yayar. YSA, bu basit nöronların (düğümün ya da ünitelerin) bağlanarak bir ağ'a dönüştürülmesiyle meydana getirilir.

4.3 YSA'nın Yapısı ve İşlem Elemanı

YSA temel olarak, basit yapıda ve yönlü bir graf biçimindedir. Her bir düğüm hücre denilen n. dereceden lineer olmayan bir devredir. Düğümler işlem elemanı olarak tanımlanır. Düğümler arasında bağlantılar vardır. Her bağlantı tek yönlü işaret iletim yolu (gecikmesiz) olarak görev yapar. Her işlem elemanı istenildiği sayıda giriş bağlantısı ve tek bir çıkış bağlantısı alabilir. Fakat bu bağlantı kopya edilebilir. Yani bu tek çıkış birçok hücreyi besleyebilir. Ağ'daki tek gecikme, çıkışları ileten bağlantı yollarındaki iletim gecikmeleridir. İşlem elemanının çıkışı istenilen matematiksel tipte olabilir. Kısmen sürekli çalışma konumunda "aktif" halde eleman bir çıkış işareti üretir. Giriş işaretleri YSA'na bilgi taşır. Sonuç ise çıkış işaretlerinden alınabilir. Şekil 4.4 'de genel bir işlem elemanı (nöron, düğüm) gösterilmiştir.



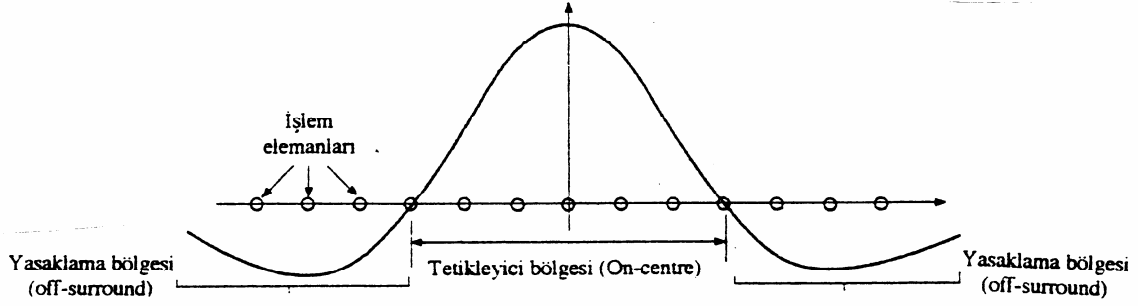
Şekil 4.4 Genel işlem elemanı yapısı

YSA birtakım alt kümelerle ayrılabilir. Bu alt kümelerdeki elemanların transfer fonksiyonları aynıdır. Bu küçük gruplara "katman" layer adı verilir. Ağ katmanların birbirlerine hiyerarşik bir şekilde bağlanmasından oluşmuştur. Dış dünyadan alınan bilgi giriş katmanı ile taşınır. Bir transfer fonksiyonları yoktur. YSA transfer fonksiyonu ve yerel bellek elemanı, bir öğrenme kuralı ile giriş çıkış işareti arasındaki bağıntıya göre ayarlanır. Girişi aktif yapabilmek için yukarıda adı geçen kural ve işaretlerin bir zamanlama fonksiyonu tanımlaması gerekebilir. Kısaca bir YSA'dan beklenen görev, gerçek dünyadaki nesneler ile biyolojik sinir ağının yaptığı işlevi, benzer bir yolla yerine getirmesidir. YSA'nın giriş veri tipleri ikili (binary) 0-1 veya sürekli değerlerdir. Bu giriş durumlarından başka, işlem elemanlarına ait girişleri matematiksel olarak da sınıflamak gerekmektedir. Çünkü bir işlem elemanına gelen girişlerin bir kısmı azaltıcı uyarma girişleri olmaktadır.

Bu arttırıcı veya azaltıcı girişler "giriş sınıflarını" oluşturur.

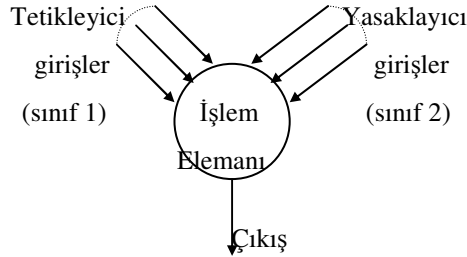
4.3.1 Giriş işareti sınıfları

İşlem elemanının transfer fonksiyonu, girişine gelen bütün işaretler için tanımlanır. Bazen değişik katman davranışlarının farklı olması tabiidir. İşaretlerin hangi bölgelerden geldiğinin bilinmesi gerekir. Değişik bölgelere göre işaretlerin sınıfları tamamlanabilir. Sıkça izlenen bir yapı ise merkezde evet/çevrede hayır (on centre/off surround) yapısıdır. Şekil 4.5'de bu yapı gösterilmektedir. Meksika şapkasına benzer bağlantı tipindedir.



Şekil 4.5 Komşu hücrelerin merkez hücreye etkisi

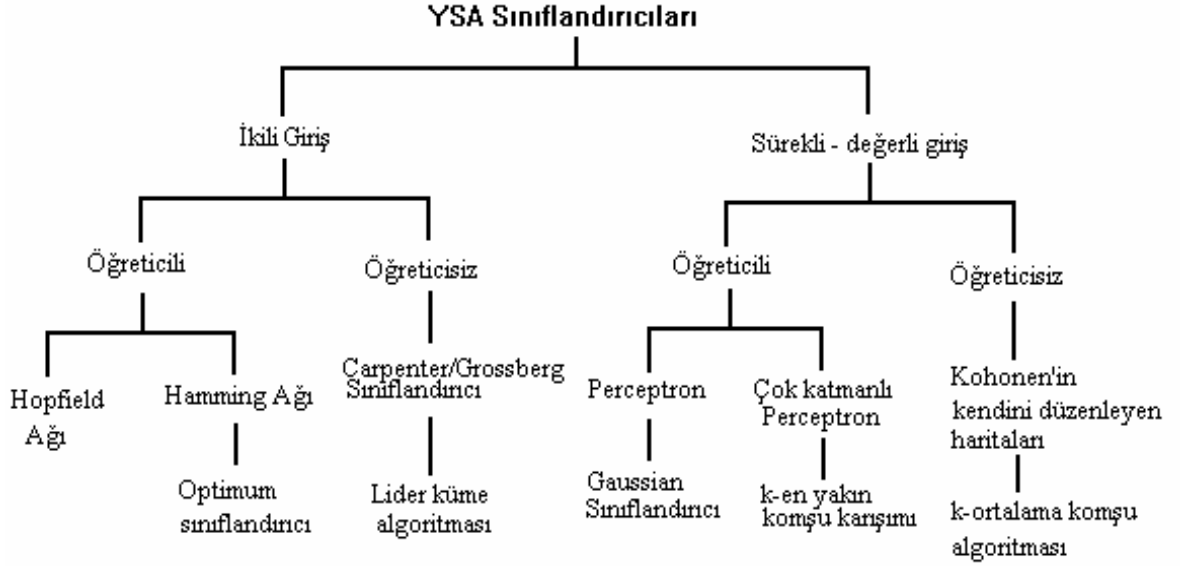
İşlem elemanı tetikleyici girişlerin kendine yakın komşu girişlerden yasaklanan girişlerini daha uzaktan alır. Böylece işlem elemanına gelen girişler sınıflarına göre değerlendirilmiş olur. Tetikleyici bölgeden gelen girişler yasaklanan sınıfı oluşturur. Şekil 4.6 böyle bir işlem elemanını gösterir.



Şekil 4.6 Tetikleyici ve yasaklanan girişlere sahip bir işlem elemanı

Bir işlem elemanına gelen girişler matematiksel tiplerine göre etiketlenilerek sınıflandırılır. YSA, giriş veri tiplerine göre ikili giriş (0,1) ve sürekli değerli giriş olmak üzere aşağıdaki gibi sınıflandırılır (Şekil 4.7).

Burada giriş işareti olarak seçilen I , w ve yük olarak kullanılan DC motordan gelen yükün P sayısal değerleri, bu değerlerin ölçümler boyunca okunması sırasında sürekli-değer de (reel sayı) olduğundan, sınıflandırıcı olarak öğreticili öğrenmeye sahip olan çok katmanlı perseptrona bağlı olarak ileri-besleme sinir ağı ve hızlı hatanın geriye-yayılımı algoritması (Fast Backpropagation Algorithm) kullanılmıştır. Bu arada klasik hatanın geriye-yayılımı algoritmasıyla yapılan eğitme sonuçları da karşılaştırmalı olarak sunulmuştur.



Ş

Şekil 4.7 YSA Sınıflandırıcıları

4.3.2 Bağlantı geometrileri

Bağlantılarda taşınan işaret verisinin cinsi tanımlanmalıdır. Bağlantı geometrisi YSA için çok önemlidir.bağlantı işareti her cinsten olabilir. Bağlantının nerede başlayıp nerede bittiğini bilmesi gerekir. 1'den N'e kadar olan bir işlem elemanı kümesinin bağlantıları aşağıda tanımlandığı gibi NxN boyutlu matris biçiminde gösterilebilir.

$$\begin{bmatrix} w_{ij} \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & \cdot & \cdot & w_{1n} \\ w_{21} & w_{22} & \cdot & \cdot & w_{2n} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ w_{n1} & w_{n2} & \cdot & \cdot & w_{nn} \end{bmatrix}$$

$w_{ij} = w_{ji} = 1 \Leftarrow i, j$ işlem elemanı j işlem elemanına bağlı

$w_{ij} = w_{ji} = 0 \Leftarrow$ bağılı değil

En fazla N^2 bağlantı olur. Bağlantılar çeşitli geometrik bölgeler arasında demetler halinde düşünülebilir. Bu bağlantı demetlerinin uyması gereken kurallar şunlardır.

- 1- Bağlantı demetini oluşturan işlem elemanları aynı bölgeden çıkmalıdır.
- 2- Bağlantı demetinin işaretleri aynı matematiksel tipten olmalıdır.
- 3- Bağlantı demetinin işaretleri aynı sınıftan olmalıdır.
- 4- Bağlantı demetinin bir seçim fonksiyonu (σ) olmalıdır.

$$\sigma : T \rightarrow 2^S \quad T: \text{Hedef belgesi} \quad S: \text{kaynak bölgesi}$$

Hedef bölgesindeki her işlem elemanı kaynak bölgesindeki her elemana giderse "tam" (full) bağlıdır. (örn:çok katmanlı perceptron). Eğer her hedef bölgesi elemanı N kaynak bölgesi

elemanına bağılı ise " düzgün dağılmış" (uniform) olasıdır. Ayrıca her bir elemana, yine bir kaynak elemanı bağılı ise buna "bire-bir" bağılı denir.

4.3.3 Ağ tipleri

Üç Çeşit Ağ Tipi Vardır

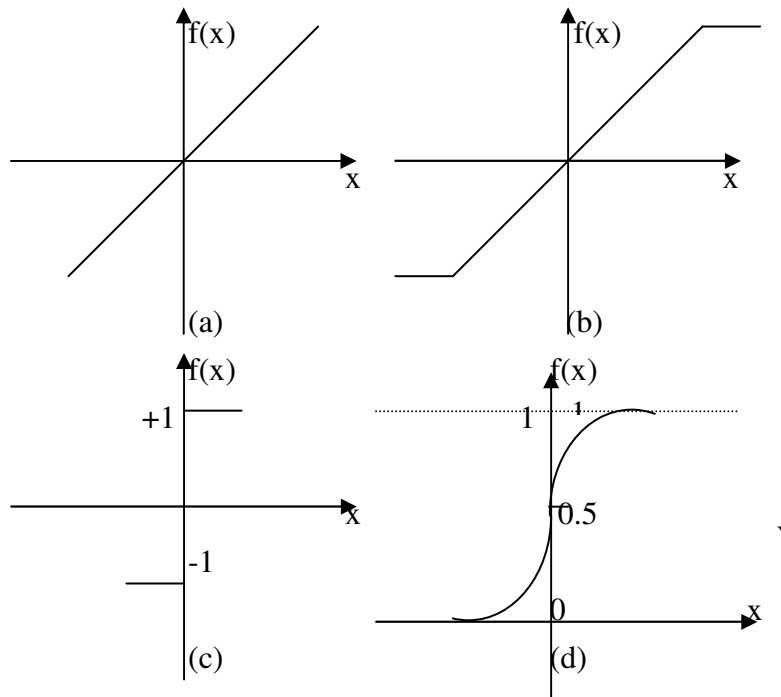
1- İleri beslemeli ağ: Her bir katmandaki hücreler sadece bir önceki katmanın hücrelerince beslenir.

2- Kaskat bağılantılı ağ: Hücreler sadece önceki katmanlardaki hücrelerce beslenir.

3- Geri beslemeli ağ: En az bir hücre sonraki katmanlardaki hücrelerce de beslenir. (Bu çalışmada hem ileri, hem de geri beslemeli ağ tipi birlikte uygulanmalıdır).

4.3.4 Eşik fonksiyonları

Transfer veya işaret fonksiyonları olarak da adlandırılan eşik fonksiyonları, muhtemel sonsuz domen girişli işlem elemanlarını önceden belirlenmiş sınırdaki çıkış olarak düzenler. Dört tane yaygın eşik fonksiyonu vardır. Bunlar, rampa, basamak ve sigmoid fonksiyonudur. Şekil 4.8'de bu fonksiyonlar gösterilmiştir.



Şekil 4.8 Sıkça kullanılan dört eşik fonksiyonu

Şekil 4.8 (a)'da gösterilen lineer fonksiyonun denklemi aşağıdaki gibidir.

$$f(x) = \alpha.x$$

α : işlem elemanının x aktivitesini ayarlayan reel değerli bir sabittir. Lineer fonksiyon $[-\tau, +\tau]$ sınırları arasında kısıtlandığında (b)'deki rampa eşik fonksiyonu olur ve denklemi;

$$f(x) = \begin{cases} +\tau & : \text{eğer } x \geq \tau \text{ ise} \\ x & : \text{eğer } |x| < \tau \text{ ise} \\ -\tau & : \text{eğer } x \leq -\tau \text{ ise} \end{cases} \quad \text{şeklini alır.}$$

$+\tau$ ($-\tau$) işlem elemanının maksimumu (minimumu) çoğu zaman doyma seviyesi olarak adlandırılan çıkış değeridir. Eğer eşik fonksiyonu bir giriş işaretiyle bağlı ise yaydığı $+\tau$ giriş toplamı pozitif, bağlı değilse eşik basamak fonksiyonu $[-\delta]$ olarak adlandırılır. Şekil 4.8 (c), basamak eşik fonksiyonunu gösterir ve denklemi;

$$f(x) = \begin{cases} +\tau & : \text{eğer } x > 0 \text{ ise} \\ -\delta & : \text{diğer durumlar} \end{cases} \quad \text{şeklindedir.}$$

Son ve en önemli eşik fonksiyonu (bu çalışmada kullanılan) sigmoid fonksiyonudur. Şekil 4.8 (d) de gösterilen S biçimindeki sigmoid fonksiyonu; seviyeli, lineer olmayan çıkış veren, sınırlı, monoton artan fonksiyondur. Denklemi;

$$f(x) = \frac{1}{1 + e^{-x}} \quad \text{biçimindedir}$$

Her işlem elemanı kendisine verilen yerel veriye göre, kendisini ayarlayacak bütün YSA'nın enformasyon bölgesinin öğrenmesini sağlar. (Enformasyon bölgesi olasılık-yoğunluk fonksiyonu ile de tanımlanabilir). Enformasyon bölgesi birçok uygulamada, gerçek değerler "0" ile "1" arasında normalize edilmesi gerekir. (Normalize etmek:gerçek değeri 85 olan bir girişi 0.85 şeklinde ağa uygulamaktır.) Normalizasyon aynı anda bütün girişlere uygulanabilir.

4.3.5 Ağırlık uzayı

Bir çok YSA öğrenme işlemi, işlem elemanlarının ağırlığı değiştirilerek sağlanır. Böylece tanımlanan ağırlık değiştirilerek öğrenmede iyi bir model kullanıp, ağırlıkların bu modele göre değiştirilmesi esastır. Basit bir matematiksel model olarak her bir işlem elemanının "n" adet gerçek ağırlığı olduğu düşünülerek ve N adet işlem elemanı gözönüne alınırsa;

$$w = (w_{11}, w_{12}, \dots, w_{1n}, w_{21}, w_{22}, \dots, w_{2n}, \dots, w_{N1}, w_{N2}, \dots, w_{Nn})^T$$

$$w = (w_1^T, w_2^T, w_3^T, \dots, w_N^T)$$

w_1, w_2, \dots, w_N : işlem elemanlarının ağırlık vektörleridir.

$$w_1 = \begin{pmatrix} w_{11} \\ w_{12} \\ \vdots \\ w_{1n} \end{pmatrix} \quad \dots \quad w_N = \begin{pmatrix} w_{N1} \\ w_{N2} \\ \vdots \\ w_{Nn} \end{pmatrix}$$

YSA ağırlık vektörü N, n boyutlu orkid uzayında yayılır. YSA'nın enformasyon işleme performansı, ağırlık vektörünün belirli bir değeri ile bulunacaktır. Hata değişimini inceleyen iki çeşit kural vardır.

- 1- Hata düzeltme kuralları ,
- 2- Gradyen kuralları.

Hata düzeltme kuralları; Her bir giriş örüntüsünde ağırlıkları yeniden ağırlayarak çıktı hatasını en aza indirmeye çalışırlar. Gradyen kurallarında ise, ağırlıklar yeniden ayarlanarak ortalama karesel hatayı (MSE) en aza indirilmeye çalışılır.

Bu noktada gradyen kuralından kısaca bahsedecek olursak, hatayı düzeltmenin (yani minimize etmenin) geometrik bir yorumunu yapmak mümkündür. Bunu yapabilmek için ağırlıkların mümkün olan tüm değerleri, hataların kareleri toplamına karşı gelecek şekilde üç boyutlu koordinat sisteminde çizilir. Bu çizim sonunda hata yüzeyi küresel bir top şeklindedir. Bu yüzeyi bir tasa da benzetmek mümkündür. Tasın en alt kısmı hataların kareleri toplamının en küçük değerlerine karşı gelmektedir. Eğitim sırasında amaç ağırlıklar kümesinin en iyisini bulmak olan, en alt kısmına ulaşmaktır. Geriye-yayılım algoritması o andaki ağırlıklar yerine, yüzey hatasının eğimini hesaplayarak amacına ulaşır. Daha sonra da bu ağırlıkları tasın alt kısmına doğru artımsal olarak değiştirir. İşte bu artımsal olarak tasın üst kısmından alt kısmına doğru ilerleme işlemine “gradyen iniş” denir.

Ağırlık vektörü ile çalışan YSA'da önemli noktalardan birisi, bir öğrenme kuralı geliştirip, enformasyon bölgesi kullanarak (eşik fonksiyonu ile) ağırlık vektörü "w" ıyı istenilen YSA performansı verecek noktaya yöneltmektir. Genellikle öğrenme kuralı için bir performans ya da maliyet fonksiyonu tanımlanır. Minimizasyon veya maksimizasyon ile "w" vektörü bulunur. Bir performasyon çeşidi olarak bilinen, MSE (karesel ortalama hata) şu şekilde tanımlanır.

$$F(w) = \int_A |f(x) - G(x, w)|^2 \rho(x) dv(x)$$

Amaç F 'i küçültmeye çalışmaktır.

$y=G(w,x)$:sistemin giriş çıkış fonksiyonu.

y :çıkış işareti vektörü

x :giriş işareti vektörü

w :ağırlık vektörü

$\rho(x)$:olasılık yoğunluk fonksiyonu

4.4 YSA'da Eğitim (Training)

4.4.1 Eğitim algoritmaları

Eğitim algoritmaları YSA'nın ayrılmaz bir parçasıdır. Eğitim algoritması eldeki problemin özelliğine göre öğrenme kuralını YSA'na nasıl adapte edeceğimizi belirtir. Üç çeşit eğitim algoritması yaygın olarak kullanılmaktadır.

- 1- Öğreticili eğitim (supervised training).
- 2- Skor ile eğitim (graded training).
- 3- Kendini düzenleme ile eğitim (self-organization training)

Öğreticili eğitimde, elimizde doğru örnekler vardır. Yani (X_1, X_2, \dots, X_N) şeklindeki giriş vektörünün, (y_1, y_2, \dots, y_N) şeklindeki çıkış vektörü, tam ve doğru olarak bilinmektedir. Herbir $(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)$ çifti için ağ doğru sonuçları verecek şekilde seçilen bir öğrenme kuralı ile beraber eğitilir.

Ne tür eğitim yöntemi kullanılırsa kullanılsın, herhangi bir ağ için gerekli karakteristik özellik, ağırlıkların verilen eğitim örneğine nasıl ayarlanacağını belirtilerek öğrenme kuralının oluşturulmasıdır. Öğrenme kuralının oluşturulması için bir örneğin, ağa defalarca tanıtılması gerekebilir. Öğrenme kuralı ile ilişkili parametreler ağın zaman içinde gelişme kaydetmesiyle değişebilir.

4.4.2 Bellek

YSA'nın önemli bir özelliği bilgiyi saklama şeklidir. Bağlantı ağırlıkları YSA bellek biçimleridir. Ağırlıkların değerleri ağın o anki bilgi durumunu temsil eder. Mesela; bir giriş/istenen çıkış çiftinin belirtilen bilgi parçası ağın içinde birçok bellek biçimine dağıtılmıştır. Bellek üniteleri ile diğer saklı bilgiler, bu bilgiyi paylaşırlar. Bazı YSA bellekleri ilişkilidir. Öyleki eğitilen ağa bir kısmı uygulanırsa, ağ bu girişe belleğindeki en yakın çıkışı bu giriş için seçer ve tam girişe bağlı çıkış ortaya çıkar. Eğer YSA oto-ilişkili ise, kısmi giriş vektörlerinin ağa verilmesi bu girişlerin tamamlanması ile sonuçlanır. YSA belleğinin yapısı; eksik, gürültülü ve tam seçilemeyen bir giriş uygulandığı zaman bile mantıklı çıkış üretmeye uygundur. Bu kurala "genelleme " adı verilir. Bir genellemenin kalitesi ve anlamı, uygulama çeşidine, ağın tipine ve karmaşıklığına dayanır. Lineer olmayan çok katmanlı ağlar (özellikle geri beslemeli ağlar) gizli katmandaki özelliklerden öğrenirler ve bunları çıkışlar üretmek için birleştirirler. Gizli katmandaki bilgi, yeni giriş örüntülerine akılcı çözümler oluşturmak için kullanılabilir.