





Yapay Sinir Ağı (YSA)

 YSA, beynin bir işlevi yerine getirme yöntemini modellemek için tasarlanan bir sistem olarak tanımlanabilir.

 YSA, yapay sinir hücrelerinin birbirleri ile çeşitli şekillerde bağlanmasından oluşur ve genellikle katmanlar şeklinde düzenlenir.

 Donanım olarak elektronik devrelerle veya bilgisayarda yazılım olarak gerçekleştirilebilir.

 Beynin öğrenme sürecine uygun olarak YSA, bir öğrenme sürecinden sonra bilgiyi toplama, hücreler arasındaki bağlantı ağırlıkları ile bu bilgiyi saklama ve genelleme yeteneğine sahip paralel dağıtılmış bir işlemcidir.

 Öğrenme süreci, arzu edilen amaca ulaşmak için YSA ağırlıklarının yenilenmesini sağlayan öğrenme algoritmalarını içerir.

Yapay Sinir Ağının Özellikleri

- Doğrusal olmama
- Bilginin saklanması
- Öğrenme
- Görülmemiş örnekler hakkında bilgi üretebilme
- Eksik bilgi ile çalışabilme
- Örüntü ilişkilendirme ve sınıflama yapabilme
- Genelleme yapabilme
- Uyarlanabilirlik yeteneğine sahip olma
- Hata toleransına sahip olma
- Sadece nümerik bilgiler ile çalışabilme
- Donanım ve hız
- Analiz ve tasarım kolaylığı

Yapay Sinir Ağlarının Dezavantajları

- Donanıma bağlı çalışmaları
- Uygun ağ yapısının belirlenmesi
- Ağın parametre değerlerinin (öğrenme katsayısı, her katmandaki hücre sayısı, katman sayısı vb.)
- Ağın sadece nümerik olarak çalışabilmesi
- Ağın eğitilmesinin ne zaman sonlandırılacağı

YSA'nın Uygulama Alanları

- Arazi analizi ve tespiti
- Tıp alanında
- Savunma sanayi
- Haberleşme
- Üretim
- Otomasyon ve Kontrol
- Görüntü işleme
- İşaret işleme
- Desen tanıma
- Askeri sistemler
- Finansal sistemler
- Planlama, kontrol ve araştırma
- Yapay zeka
- Güç sistemleri

YSA'nın Kısa Bir Tarihçesi

- ✓ YSA'nın tarihçesi nörobiyoloji konusuna insanların ilgi duyması ve elde ettikleri bilgileri bilgisayar bilimine uygulamaları ile başlamaktadır.
- ✓ YSA' ile ilgili çalışmaları 1970 öncesi ve sonrası olarak ikiye ayırmak mümkündür.

1970 Öncesi Çalışmalar

- 1890-İnsan beyninin yapısı ve fonksiyonları ile ilgili ilk yayının yazılması
- 1911-İnsan beyninin bileşenlerinin belirli bir düzenek ile sinir hücrelerinden (nöronlar) oluştuğu fikrinin benimsenmesi
- 1943-Yapay sinir hücrelerine dayalı hesaplama teorisinin ortaya atılması ve eşik değerli mantıksal devrelerin geliştirilmesi
- 1949-Biyolojik olarak mümkün olabilen öğrenme prosedürünün bilgisayar tarafından gerçekleştirilecek biçimde geliştirilmesi
- 1956-1962-ADALINE ve Widrow öğrenme algoritmasının geliştirilmesi
- 1957-1962-Tek katmanlı algılayıcının (perceptron) geliştirilmesi
- 1965-İlk makine öğrenmesi kitabının yazılması
- 1967-1969-Bazı gelişmiş öğrenme algoritmalarının geliştirilmesi
- 1969-Tek katmanlı algılayıcıların problemleri çözme yeteneğinin olmadığı gösterilmesi

1970 Sonrası Çalışmalar

- 1969-1972-Doğrusal ilişkilerin geliştirilmesi
- 1972- Korelasyon matriks belleğinin geliştirilmesi
- 1974-Geriye yayılım modelinin geliştirilmesi
- 1978-ART modelinin geliştirilmesi
- 1982-Kohonen öğrenmesi ve SOM modelinin geliştirilmesi
- 1982-Hopppfield ağlarının geliştirilmesi
- 1982-Çok katmanlı algılayıcıların geliştirilmesi
- 1984-Boltzmann makinesinin geliştirilmesi
- 1988-RBF modelinin geliştirilmesi
- 1991-GRNN modelinin geliştirilmesi
- 1991-2000'li yıllar- Bu zaman diliminde sayısız çalışma ve uygulamalar hayata geçmiştir.

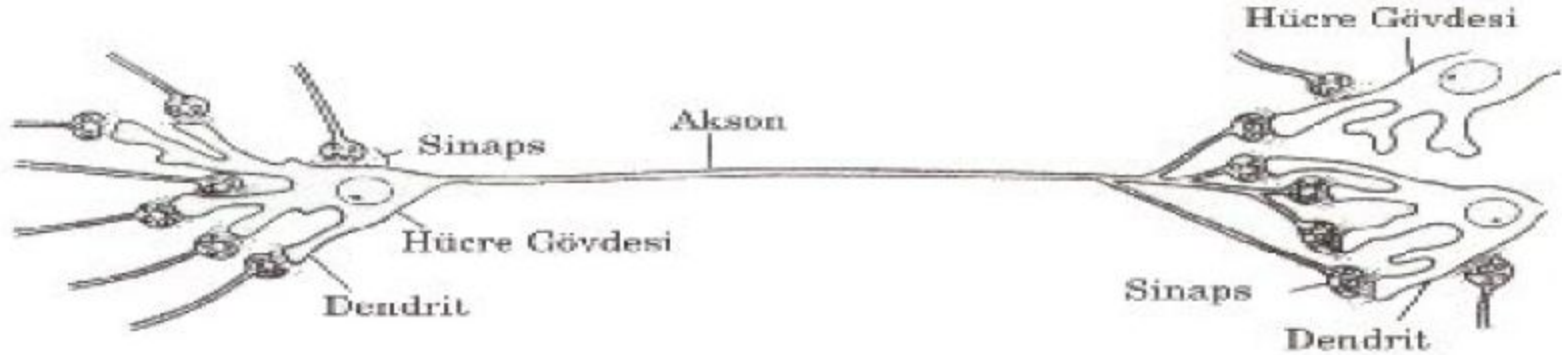
Biyolojik Sinir Sistemi

- ✓ Alıcı sinirler (receptor) organizma içerisinde ya da dış ortamlardan aldıkları uyarıları, beyne bilgi ileten elektriksel sinyallere dönüştürür.
- ✓ Tepki sinirleri (effector) ise, beyin ürettiği elektriksel darbeleri organizma çıktısı olarak uygun tepkilere dönüştürür.



Biyolojik Sinir Sisteminin Blok Gösterimi


Biyolojik Sinir Hücresi



- ✓ **Dendritler**, sinaptik sinyalleri girdi olarak almakta
- ✓ **Hücre gövdesi**, bu sinyali analog bir yöntemle işlemekte
- ✓ **Akson**, üretilen denetim sinyalleri aksonlar aracılığı ile denetlenecek hedef hücrelere iletilmektedir.
- ✓ **Sinaps** (akson uçları)

 Bir sinir hücresi, gelen elektrik darbelerinden üç şekilde etkilenir:

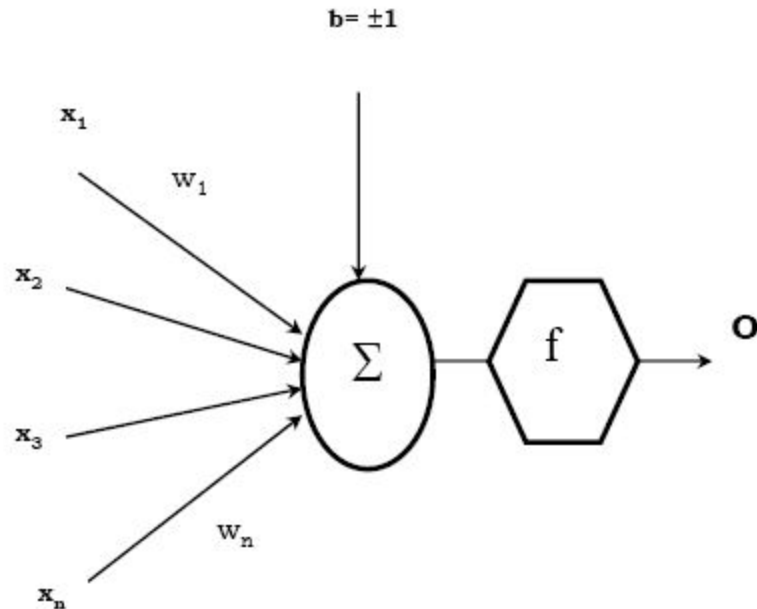
- Gelen darbelerden bazıları nöronu uyarır.
- Bazıları bastırır.
- Geri kalanı da değişikliğe yol açar.

 İnsan beyninin 10 milyar sinir hücresinden ve 60 trilyon synapse bağlantısından oluşur.


Yapay Hücre Modelleri


✓ Temel bir yapay sinir ağı hücresi biyolojik sinir hücresine göre çok daha basit bir yapıya sahiptir. Yapay sinir ağı hücresinde;


- Girişler
- Ağırlıklar
- Toplama fonksiyonu
- Aktivasyon fonksiyonu
- Çıktılar bulunur.



Temel yapay sinir ağı hücresi

 **Girdiler:** Yapay sinir hücresine gelen girişler ağın öğrenmesi istenen örnekler tarafından belirlenir. Giriş verileri, dış ortamdan girilen bilgiler olabildiği gibi başka hücrelerden veya kendi kendisinden de bilgiler olabilir.

 **Ağırlıklar:** Ağırlıklar bir yapay hücreye gelen bilginin önemini ve hücre üzerindeki etkisini gösterir. Her bir giriş kendine ait bir ağırlığa sahiptir. Ağırlıklar pozitif, negatif, sıfır, sabit veya değişken değerler olabilir.

 **Toplam Fonksiyonu:** Bir hücreye gelen net girişi hesaplar. Bunun için değişik fonksiyonlar kullanılmaktadır. Genellikle deneme-yanılma yolu ile toplama fonksiyonu belirlenir.



En yaygın toplama fonksiyonu ağırlıklar toplamıdır.

$$\text{Net Girdi} = \sum_i X_i W_i$$



Bir yapay sinir ağında, her hücre elemanı bağımsız olarak farklı bir toplama fonksiyonuna sahip olabileceği gibi hepsi aynı toplama fonksiyonuna da sahip olabilir.

Toplama Fonksiyonu Örnekleri

Net Giriş	Açıklama
Çarpım Net Girdi $= \prod_i X_i W_i$	Ağırlık değerleri girdiler ile çarpılır ve daha sonra bulunan değerler birbirleri ile çarpılarak net girdi hesaplanır.
Maksimum Net Girdi $= \text{Max}(X_i W_i) \quad (i=1 \dots N)$	N adet girdi içinden ağırlıklar ile çarpıldıktan sonra en büyüğü yapay sinir hücresinin net girdisi olarak kabul edilir.
Minimum Net Girdi $= \text{Min}(X_i W_i) \quad (i=1 \dots N)$	N adet girdi içinden ağırlıklar ile çarpıldıktan sonra en küçüğü yapay sinir hücresinin net girdisi olarak kabul edilir.
Çoğunluk Net Girdi $= \sum_i \text{sgn} X_i W_i$	N adet girdi içinden ağırlıklar ile çarpıldıktan sonra pozitif ve negatif olanların sayısı bulunur. Büyük olan sayı hücrenin net girdisi olarak kabul edilir.
Kümülatif toplam Net Girdi $= \text{Net}(\text{eski}) + \sum_i \text{sgn} X_i W_i$	Hücreye gelen bilgiler ağırlıklı olarak toplanır ve daha önce gelen bilgilere eklenerek hücrenin net girdisi bulunur.



Aktivasyon (Etkinlik) Fonksiyonu: Bu fonksiyon, hücreye gelen net girdiyi işleyerek hücrenin bu girdiye karşılık üreteceği çıktıyı belirler.

Genelde aktivasyon fonksiyonu doğrusal olmayan bir fonksiyondur ve toplama fonksiyonunda olduğu gibi ağın hücre elemanlarının hepsi aynı fonksiyonu kullanması gerekmez. En çok kullanılan aktivasyon fonksiyonu sigmoid fonksiyondur.

$$f(x) = \text{lojistik}(x) = \frac{1}{1 + \exp(-\beta x)}$$

Burada, x hücre elemanlarına gelen net girdi değerini göstermektedir.

Aktivasyon Fonksiyonu Örnekleri

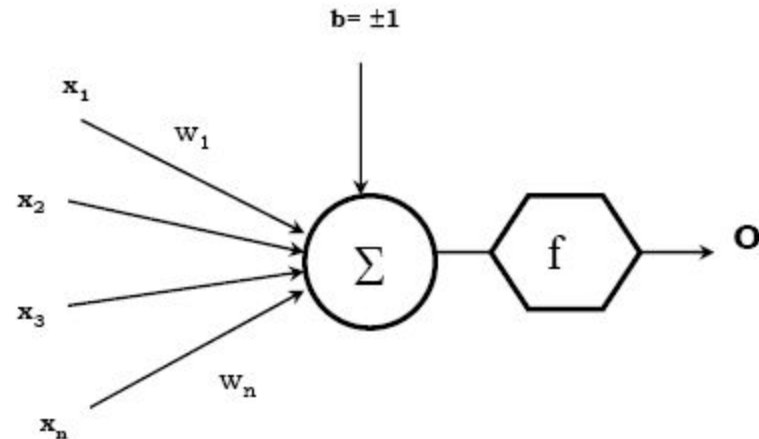
Aktivasyon Fonksiyonu	Açıklama
Doğrusal fonksiyon $f(\text{Net}) = \text{Net}$	Gelen girdiler olduğu gibi hücrenin çıkışı olarak kabul edilir
Step fonksiyonu $f(\text{Net}) = \begin{cases} 1 & \text{eğer } \text{Net} > \text{eşik değeri} \\ 0 & \text{eğer } \text{Net} \leq \text{eşik değeri} \end{cases}$	Gelen Net girdi değeri belirlenen bir eşik değerinin altında veya üstünde olmasına göre hücrenin çıkışı 1 veya 0 değerini alır.
Doyumlu doğrusal fonksiyon $f(\text{Net}) = \begin{cases} 1 & \text{eğer } \text{Net} > 1 \\ \text{Net} & \text{eğer } -1 < \text{Net} < 1 \\ -1 & \text{eğer } \text{Net} \leq -1 \end{cases}$	Gelen bilgilerin 0 veya 1' den büyük veya küçük olmasına göre 0 ve 1 değerini alır. 0 ve 1 arasında ise Net girdi çıkışa gönderilir
Sinus fonksiyonu $F(\text{Net}) = \sin(\text{Net})$	Öğrenilmesi düşünülen olayların sinüs fonksiyonuna uygun dağılım gösterdiği durumlarda kullanılır.
Hiperbolik tanjant fonksiyonu $F(\text{Net}) = \frac{e^{\text{Net}} - e^{-\text{Net}}}{e^{\text{Net}} + e^{-\text{Net}}}$	Gelen Net girdi değerini tanjant fonksiyonundan geçirilmesi ile hesaplanır.



Hücre Çıkışı: Aktivasyon fonksiyonu tarafından belirlenen çıktı değeridir. Üretilen çıkış dış dünyaya veya bir başka hücreye giriş olarak gönderilebilir. Hücre kendi çıktısını kendisine girdi olarak da gönderebilir.

Örnek

$$\begin{array}{ll} x_1 = 0.5 & w_1 = -0.2 \\ x_2 = 0.6 & w_2 = 0.6 \\ x_3 = 0.2 & w_3 = 0.2 \\ x_4 = 0.7 & w_4 = -0.1 \end{array}$$



$$\text{Net} = \sum x_i w_i \quad i=1 \dots 4$$

$$\text{Net} = 0.5 * (-0.2) + 0.6 * 0.6 + 0.2 * 0.2 + 0.7 * (-0.1)$$




$$\text{Net} = 0.23$$

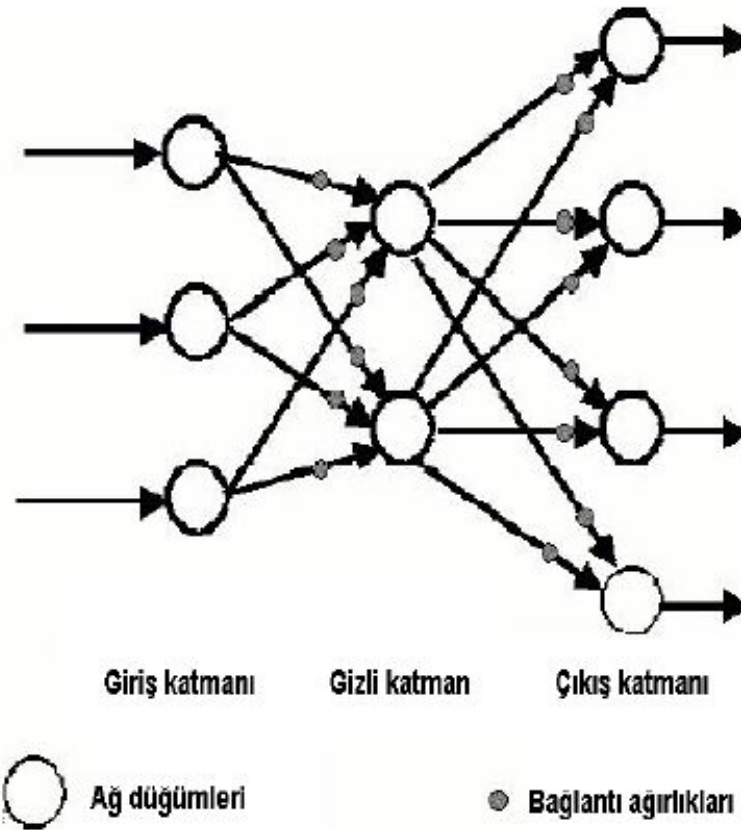
Hücrenin sigmoid aktivasyon fonksiyonuna göre çıkışı

$$f(\text{Net}) = 1 / (1 + e^{-0.23})$$


$$f(\text{Net}) = 0.56$$


Yapay Sinir Ağlarının Yapısı


-  Yapay sinir hücreleri bir araya gelerek yapay sinir ağını oluşturmaktadır.
-  Sinir hücrelerinin bir araya gelmesi rastgele olmaz.
-  Genel olarak hücreler 3 katman halinde ve her katman içinde paralel olarak bir araya gelerek ağı oluştururlar.




Yapay sinir ağı yapısı

 **Girdi Katmanı:** Bu katmandaki hücre elemanları dış dünyadan bilgileri alarak ara katmanlara iletmekle sorumludur. Bazı ağlarda girdi katmanında herhangi bir bilgi işleme olmaz.

 **Ara Katman:** Girdi katmanından gelen bilgileri işleyerek çıkış katmanına gönderirler. Bir ağ için birden fazla ara katman olabilir.

 **Çıktı Katmanı:** Bu katmandaki hücre elemanları ara katmandan gelen bilgileri işleyerek ağın girdi katmanında sunulan girdi seti için üretmesi gereken çıktıyı üretir. Üretilen çıktı dış dünyaya iletilir.




Yapay Sinir Ağları Modelleri

 Hücrelerin bağlantı şekillerine, öğrenme kurallarına ve aktivasyon fonksiyonlarına göre çeşitli YSA yapıları geliştirilmiştir.

 Yapay sinir ağı modelleri temel olarak iki grupta toplanmaktadır. Bunlar:

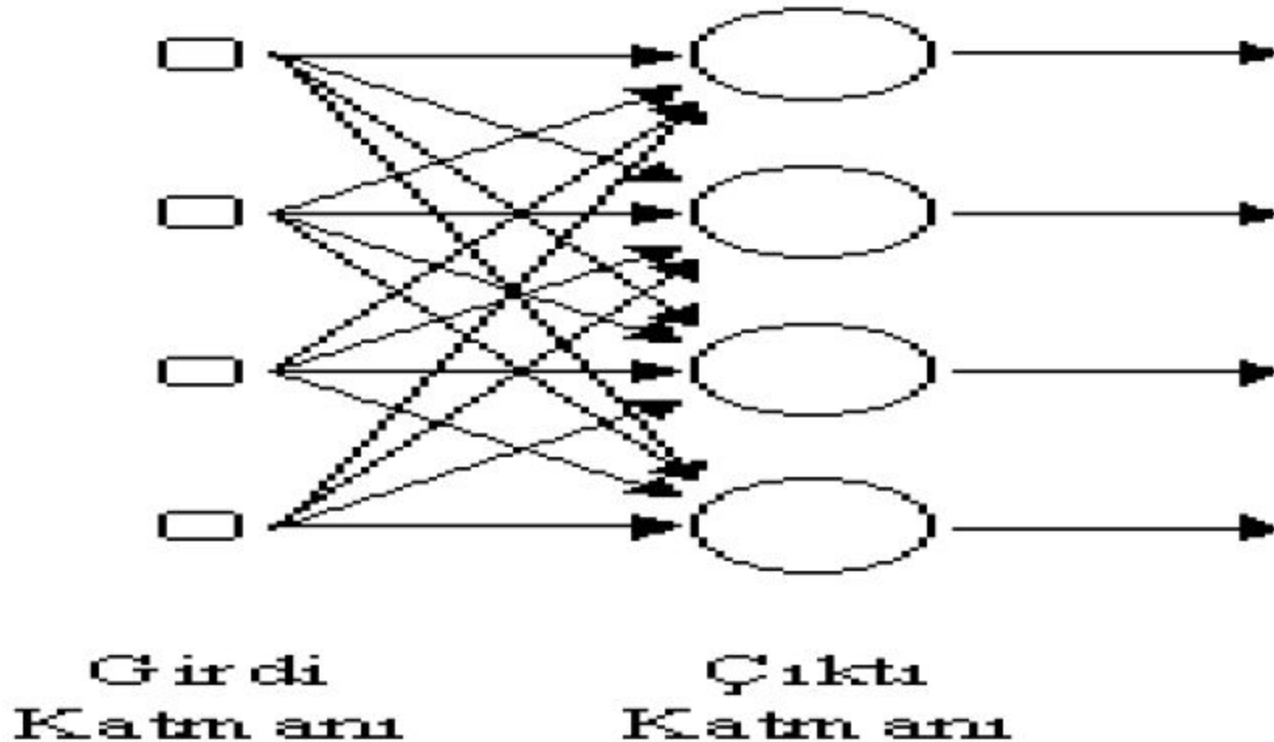
1. İleri beslemeli yapay sinir ağları
2. Geri beslemeli yapay sinir ağları

I) İleri Beslemeli Yapay Sinir Ağları

-  İşlem girişlerden çıkışlara doğru ilerler.
-  Çıkış değerleri öğreticiden alınan istenen çıkış değerleri ile karşılaştırılır ve bir hata sinyali elde edilerek ağ ağırlıkları güncellenir.
-  İleri beslemeli yapay sinir ağlarında gecikmeler yoktur. Kendi aralarında:
 - Tek katmanlı ileri beslemeli ağlar
 - Çok katmanlı ileri beslemeli ağlar olarak ayrılırlar.

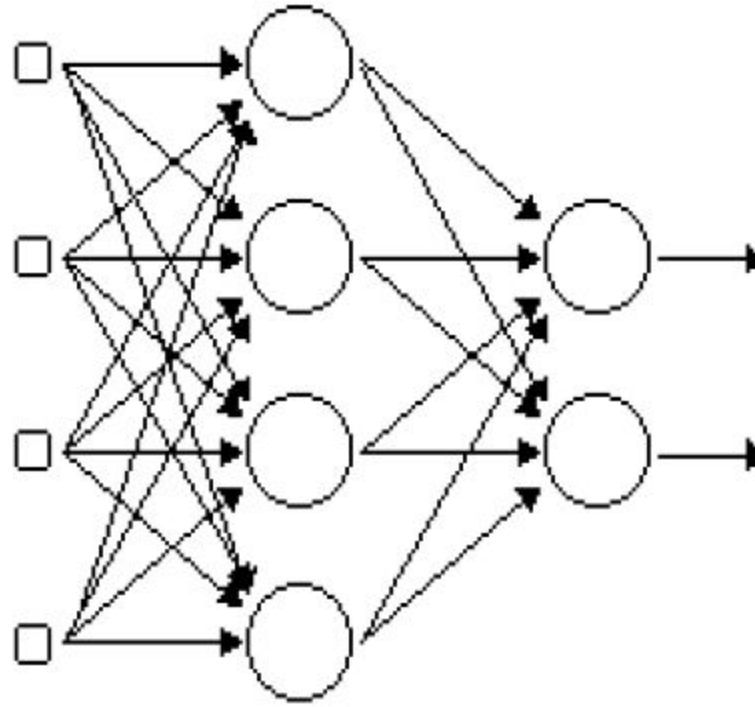
Tek Katmanlı İleri Beslemeli Ağlar

- ✓ En basit ağ tipi olup bir çıktı katmanı ve buna bağlı bir girdi katmanından oluşmaktadır.



Çok Katmanlı İleri Beslemeli Ağlar

- ✓ Girdi katmanı dış ortamlardan aldığı bilgileri hiçbir değişikliğe uğratmadan orta (gizli) katmandaki hücrelere iletir.
- ✓ Bilgi, orta ve çıkış katmanında işlenerek ağ çıkışı belirlenir.
- ✓ Çok katmanlı ağlar tek katmanlı ağlara göre daha karmaşık problemlerin çözümünde kullanılırlar.
- ✓ Çok katmanlı ağların eğitilmesi zordur.






Girdi
Katmanı

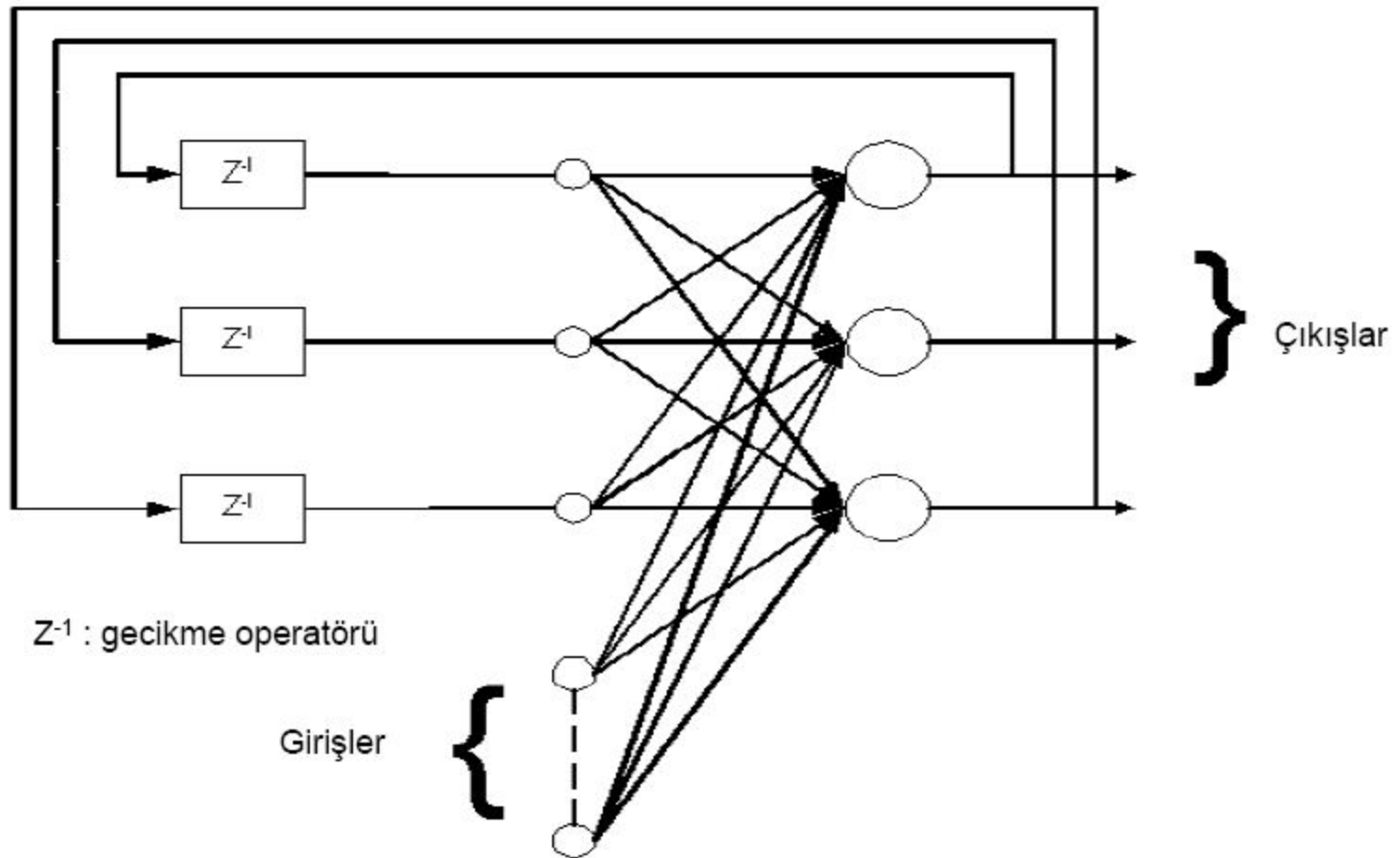
Gizli
Katman

Çıktı
Katmanı

Katmanlı İleri Beslemeli Ağ Yapısı

II)Geri Beslemeli Yapay Sinir Ağları

-  Geri beslemeli sinir ağı, ileri beslemeli bir ağın çıkışlarının girişlere bağlanması ile elde edilir.
-  Geri beslemeli sinir ağlarında gecikmeler vardır.
-  Geri beslemeli sinir ağları, hücreler arası veya katmanlar arası geri besleme yapılış şekline göre farklı isimlerle söylenir.
 - Tam geri beslemeli ağlar
 - Kısmi geri beslemeli ağlar



Geri beslemeli yapı

Tam Geri Beslemeli Ağlar






Bu ağlar gelişmiş güzel ileri ve geri bağlantıları olan ağlardır.



Bu bağlantıların hepsi eğitilebilir.

Kısmi geri beslemeli ağlar

-  Bu ağlarda, ağın hücre elemanlarına ek olarak içerik (context) elemanları vardır.
-  Geri besleme sadece içerik elemanları üzerinde yapılır ve bu bağlantılar eğitilemezler.
-  İçerik elemanları ara katman elemanlarının geçmiş durumlarını hatırlamak için kullanılır.

Yapay Sinir Ağlarında En Çok Kullanılan Modeller

- Algılayıcılar (Perceptronlar)
- Çok Katmanlı Algılayıcılar
- Lineer Vektör Quantization Modeli (LVQ)
- Kendi Kendini Organize Eden Model (SOM)
- Adaptive Rezorans Teorisi Modeli (ART)
- Hopfield Ağı
- CounterPropogation Ağı

Yapay Sinir Ağlarında En Çok Kullanılan Modeller

- Neocognitron Ağı
- Boltzman Makinesi
- Probabilistic Ağı
- Elman Ağı
- Jordan Ağı
- Radyal Temelli Ağ(RBN)
- Kohonen Ağı

Yapay Sinir Ağlarının Eğitilmesi

- ✓ YSA'da hücre elemanlarının bağlantılarının ağırlık değerlerinin belirlenmesi işlemine “ağın eğitilmesi” denir.
- ✓ Başlangıçta bu ağırlık değerleri rastgele alınır.
- ✓ YSA'lar, kendilerine örnekler gösterildikçe bu ağırlık değerlerini yenileyerek amaca ulaşmaya çalışırlar.

Yapay Sinir Ağlarının Eğitilmesi

- ✓ Amaca ulaşmanın veya yaklaşmanın ölçüsü de yine dışarıdan verilen bir değerdir.
- ✓ Eğer yapay sinir ağı verilen giriş-çıkış çiftleriyle amaca ulaşmış ise ağırlık değerleri saklanır.
- ✓ Ağırlıkların sürekli yenilenip istenilen sonuca ulaşılan kadar geçen zamana “**öğrenme**” denir.
- ✓ Ağırlık değerlerinin değişmesi belirli kurallara göre yürütülmektedir. Bu kurallara “**öğrenme kuralları**” denir.



Yapay Sinir Ağlarının Eğitilmesi


- ✓ Yapay sinir ağı öğrendikten sonra daha önce verilmeyen girişler uygulanarak ağ çıkışları gözlemlenir.
- ✓ Genelde eldeki örneklerin %80'i ağı verilip ağ eğitilir. Daha sonra geri kalan %20'lik kısım verilip ağın davranışları incelenir ve bu işleme “ağın test edilmesi” denir.
- ✓ Eğitimde kullanılan örnekler setine “eğitim seti”, test için kullanılan sete ise “test seti” denir.


Öğrenme Stratejileri

- Danışmanlı (Supervised) Öğrenme
- Danışmansız (Unsupervised) Öğrenme
- Takviyeli (Reinforcement) Öğrenme
- Karma Stratejiler

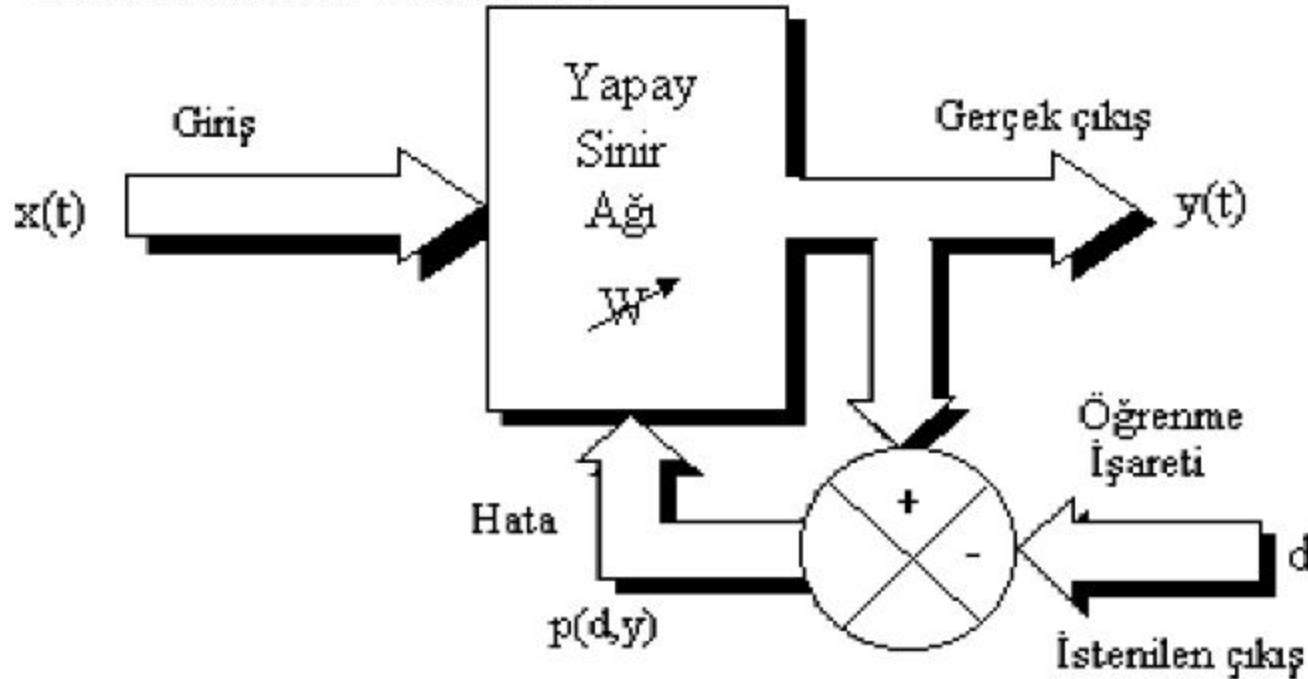
I) Danışmanlı (Supervised) Öğrenme

-  Eğitim sırasında sisteme bir girdi ve bir hedef çıktı vektörlerinin çift olarak verilmesi ve bunlara göre sistemdeki ağırlık değerlerinin güncellenmesi ve değiştirilmesi yapılır.
-  Yapay sinir ağlarının eğitilmesinde kullanılan hedef çıktı 1 veya 0 olabileceği gibi bir örüntü de olabilir.




 Belli bir bilgi kümesine karşılık, ilgili çıktı kümesini hatırlayacak şekilde eğitilmiş sistemlere “**çağrışimli bellek**” denir.

 Eğer, girdi vektörü ile çıktı vektörü aynı ise buna “**öz çağrışimli bellek**”, çıktı vektörü farklı ise buna da “**karşıt çağrışimli bellek**” denir.

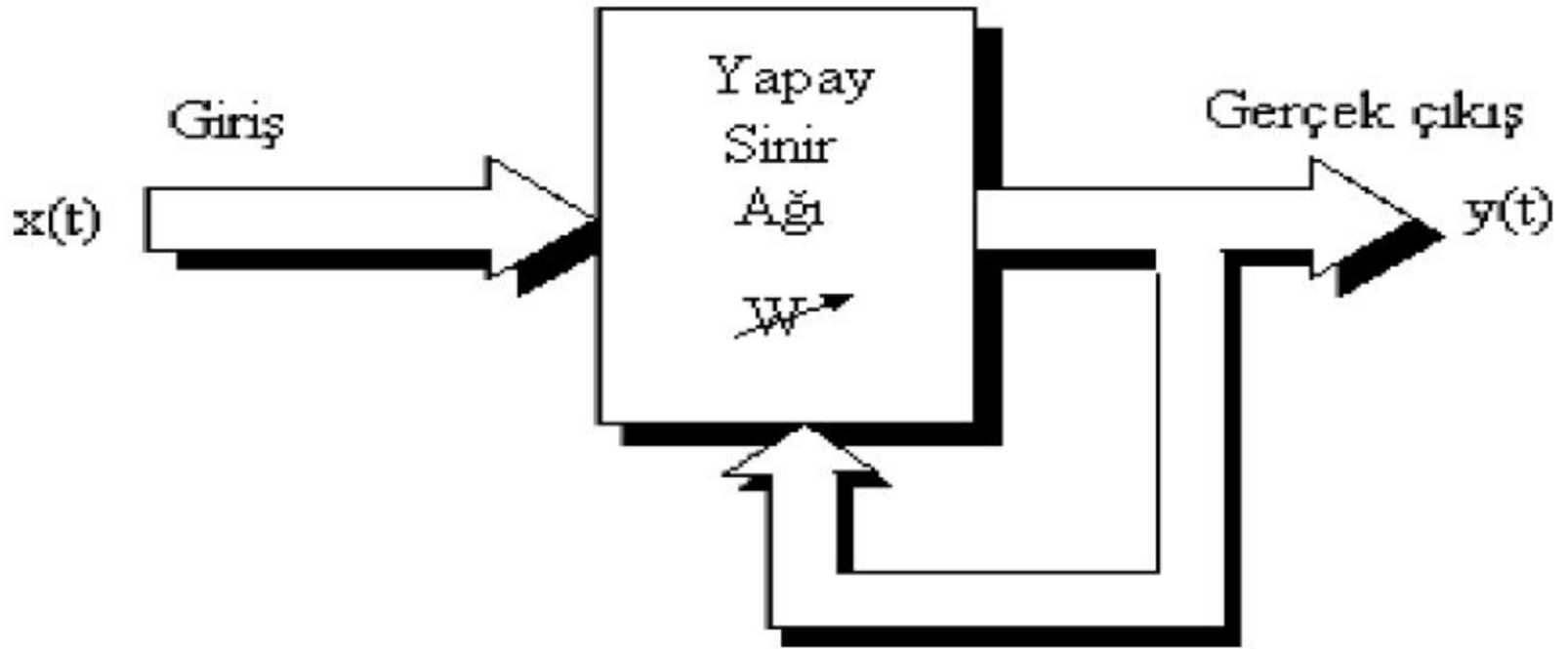
- Widrow-Hoff tarafından geliştirilen '*Delta Kuralı*' ve Rumelhart ve McClelland tarafından geliştirilen '*Genelleştirilmiş Delta Kuralı*' veya '*Geri Besleme (back propagation) algoritması*' danışmanlı öğrenme algoritmalarına örnek olarak verilebilir.



II)Danışmansız Öğrenme



-  Bu sistemlerde, bir grup girdi vektörü sisteme verilir, ancak hedef çıktılar belirtilmez.
-  Sistem girdiler içerisinde birbirine en çok benzeyenleri gruplar ve her bir grup için farklı bir örüntü tanımlar.
-  Özdüzenlemeli özellik haritaları bu yöntemi kullanarak sınıflama işlemini yerine getirir.


- ✓ Bu Grossberg tarafından geliştirilen **ART (Adaptive Resonance Theory)** veya Kohonen tarafından geliştirilen **SOM(Self Organizing Map)** öğrenme kuralı danışmansız öğrenmeye örnek olarak verilebilir.



Danışmansız Öğrenme Yapısı

III) Takviyeli Öğrenme

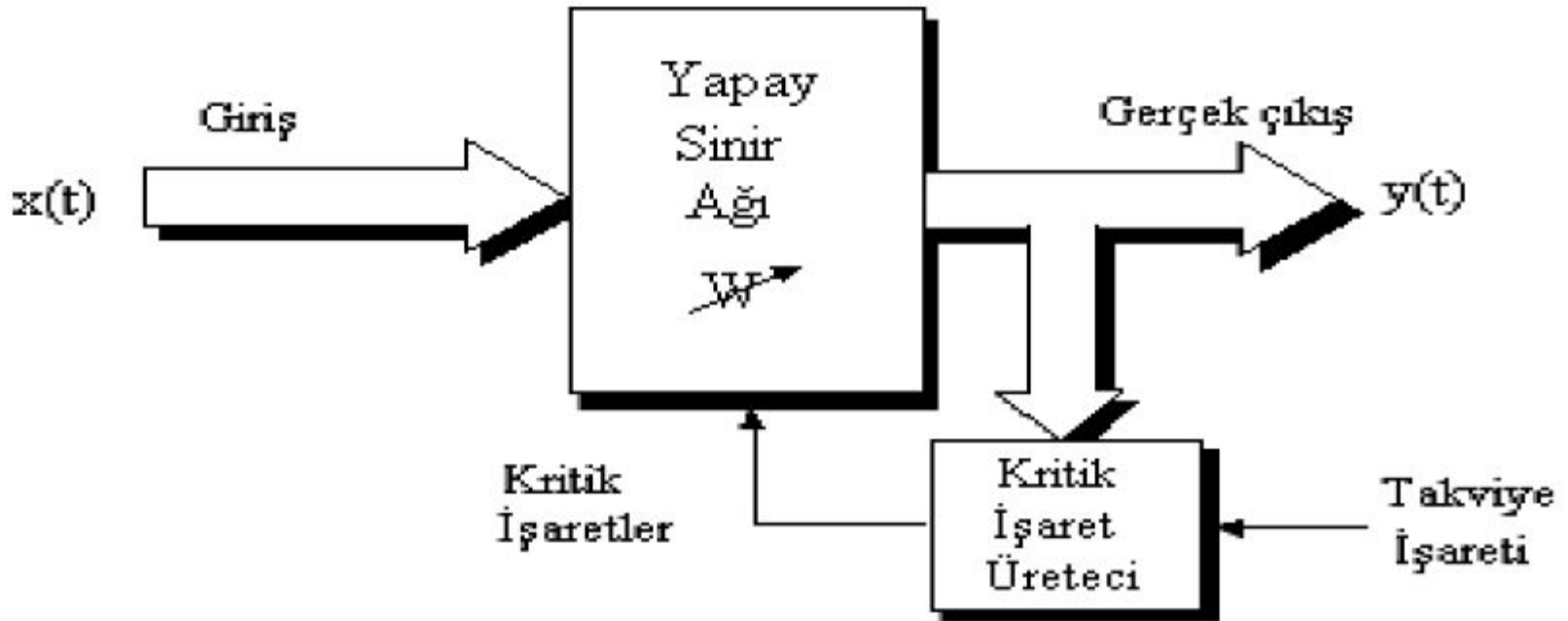
-  Bu öğrenme kuralı danışmanlı öğrenmeye yakın bir metottur.
-  Hedef çıktıyı vermek için bir “öğretmen” yerine, burada yapay sinir ağına bir çıkış verilmemekte fakat elde edilen çıkışın verilen girişe karşılık iyiliğini değerlendiren bir kriter kullanılmaktadır.

 Takviyeli öğrenmede, ağın davranışlarının uygun olup olmadığını belirten bir özyetenek bilgisine ihtiyaç duyulur.

 Bu bilgiye göre ağırlıklar ayarlanır.



 Gerçek zamanda öğrenme olup, deneme-yanılma esasına göre sinir ağı eğitilmektedir.

- ✓ Optimizasyon problemlerini çözmek için Hinton ve Sejnowski'nin geliştirdiği Boltzman Kuralı veya Genetik Algoritmalar takviyeli öğrenmeye örnek olarak verilebilir.





Takviyeli Öğrenme Yapısı

IV) Karma Öğrenme

-  Danışmanlı, danışmansız veya takviyeli öğrenme stratejilerinden birkaçını birlikte kullanarak geliştirilen yapıdır.
-  Radial Tabanlı (Radial Basis Network) ağlar ve Olasılık Tabanlı (Probabilistic Neural Network) ağlar bunlara örnek olarak verilebilir.

Öğrenme Kuralları

 **Çevrimiçi (On-line) Öğrenme Kuralları:** Bu kurallara göre öğrenen sistemler gerçek zamanda çalışırken bir taraftan fonksiyonlarını yerine getirmekte diğer taraftan ise öğrenmeye devam etmektedir. (ART ve Kohonen öğrenme kuralı bu sınıfa girmektedir.

 **Çevrimdışı (Off-line) Öğrenme Kuralları:** Bu kuralları kullanan sistemler eğitildikten sonra gerçek hayatta kullanıma alındığında artık öğrenme olmamaktadır. Sistemin öğrenmesi gereken yeni bilgiler söz konusu olduğunda sistem kullanımdan çıkarılmakta ve çevrimdışı olarak yeniden eğitilmektedir.

Öğrenme Algoritmaları

Geri Yayınım Algoritması

Levenberg-Marquardt öğrenme algoritması

Delta kuralı

Delta-Bar-Delta (DBD)

Genişletilmiş delta-bar-delta (EDBD-extended delta-bar-delta)

Genetik Algoritma (GA)