

T.C.  
SAKARYA ÜNİVERSİTESİ  
SOSYAL BİLİMLER ENSTİTÜSÜ

148203

**YAPAY SİNİR AĞLARI VE BİR İŞLETMEDE  
MALİYET/ÜRETİM MİKTARI İLİŞKİSİNİN YAPAY  
SİNİR AĞI İLE BELİRLENMESİ**

**YÜKSEK LİSANS TEZİ**

**Elk-Elektronik Müh. Ömer Engin POLAT**

148203

**Enstitü Anabilim Dalı : İŞLETME  
Enstitü Bilim Dalı : ÜRETİM YÖNETİMİ VE PAZARLAMA**

**Tez Danışmanı : Yrd. Doç. Dr. Harun Reşit YAZGAN**

**EYLÜL-2003**

T.C.  
SAKARYA ÜNİVERSİTESİ  
SOSYAL BİLİMLER ENSTİTÜSÜ

148203

**YAPAY SINIR AĞLARI VE BİR İŞLETMEDE  
MALİYET/ÜRETİM MİKTARI İLİŞKİSİNİN YAPAY  
SINIR AĞI İLE BELİRLENMESİ**

**YÜKSEK LİSANS TEZİ**

**Elk-Elektronik Müh. Ömer Engin POLAT**

**Enstitü Anabilim Dalı : İŞLETME  
Enstitü Bilim Dalı : ÜRETİM YÖNETİMİ VE PAZARLAMA**

**Bu tez .. / .. / 20.. tarihinde aşağıdaki jüri tarafından Oybirliği / Oyçokluğu ile kabul edilmiştir.**



**Doç. Dr. İ. Hakkı UDEMÖĞLÜ**  
**Jüri Başkanı**



**Trş. Doç. Dr.**  
**Jüri Üyesi**

**Hasan R. Yargan**

**Yrd. Doç. Dr.**  
**Jüri Üyesi**

**Mustafa 31920500**

## ÖNSÖZ

Günlük hayata ilişkin en basit olaylar arasındaki ilişkiler bile matematiksel ifadelerle dönüştürülmek istendiğinde, karşımıza çıkan fonksiyonlar çoğunlukla doğrusal olmayan yani üstel ifadeler olmaktadır. Aynı şeyi bir işletmede, bir prosesin incelenmesi, bakım ile üretim arasındaki ilişki, rotalama, eldekilerle optimum sonucun elde edilmesi, maliyet tahmini ile bütçe planlama..vb açılardan düşündüğümüzde, işletmelerdeki faaliyetlerin matematiksel ifadelerle dönüştürülerek çözülebilmesi çoğu zaman imkansız gibi gözükmektedir.

Günümüzde benzer ilişkilerin çözülebilmesi için Yapay Zeka tabanlı birçok uygulama geliştirilmiş ve matematiksel ifadelerle dönüştürülemeyen ve bilinen algoritmalar ile çözülemeyen (üstel (stokastik) ifadeler) ilişkilerin sonuçlarının tahmin edilebilmesi için doğadaki olaylardan esinlenerek bazı yaklaşımlar geliştirilmiştir. Özellikle bugünlerde işletmelerin sıklıkla kullanmaya başladığı bu yaklaşımlardan başlıcaları Yapay Sinir Ağları, Genetik Algoritmalar, Uzman Sistemler...vb'dir ve çoğunda da problemin çözülebilmesi için sezgisel (heuristic) bir yaklaşım kullanılmaktadır.

Bu ihtiyaçlardan doğan Yapay Sinir Ağları yaklaşımı günümüzde işletmelerde etkin bir şekilde uygulanmaya başlanmıştır. Optimizasyon, çizelgeleme, talep tahmini, maliyet tahmini...vb. konularla ilgili uygulamalar günümüzde mevcuttur.

Tez çalışmam sırasında ilgi, yardım ve desteğini esirgemeyen ve yardımlarıyla beni yönlendiren tez danışmanım Sayın Yrd. Doç. Dr. Harun Reşit YAZGAN Hocam'a, teşekkürü bir borç bilirim.

Çalışmayı yapmama izin veren TOPRAK Sağlık Ürünleri fabrikası yetkililerine, ilgi ve yardımlarından ötürü teşekkür ederim.

Tez çalışmam sırasında verdikleri destek ve gösterdikleri büyük sabır için eşim Tülay KORKUSUZ POLAT'a ve henüz 53 günlük olan kızım Gökçe POLAT'a teşekkürlerimin ve minnettarlığımın en özelini sunuyorum.



# İÇİNDEKİLER

KISALTMALAR.....	v
ŞEKİLLER LİSTESİ.....	vi
TABLOLAR LİSTESİ.....	vii
ÖZET.....	viii
SUMMARY.....	ix
GİRİŞ.....	1
1. YAPAY ZEKA.....	2
1.1. Yapay Zekanın Tanımı ve İçeriği .....	2
1.2. Yapay Zekanın Karakteristik Özellikleri .....	6
1.3. Yapay Zeka Teknolojileri.....	6
1.4. Yapay Zekanın Kapsamı .....	7
1.5. Yapay Zeka Araçlarının Ortak Özellikleri.....	7
1.6. Yapay Zeka ve Klasik Sistemlerin Karşılaştırılması .....	9
1.7. Yapay Zekanın Uygulamaları .....	10
1.8. Yapay Zekanın Uygulama Örnekleri .....	10
1.9. Yapay Zekanın Uygulamalarına Bazı Örnekler.....	11
2. YAPAY SİNİR AĞLARI.....	12
2.1. Biyolojik Sinir Ağları .....	13
2.2. Yapay Sinir Ağları Nedir? .....	16
2.3. Sinir Ağları Niçin Kullanılır?.....	18
2.4. Geleneksel Bilgisayarlar ve Yapay Sinir Ağları – Bir Karşılaştırma.....	18
2.5. Yapay Sinir Ağlarının Avantajları .....	19
2.6. Yapay Sinir Ağlarının Özellikleri.....	20
2.7. Yapay Sinir Ağlarının Uygulamaları .....	21
3. YAPAY SİNİR AĞLARININ YAPISI.....	23
3.1. Temel Bileşenleri .....	23
3.1.1. Girdi Elemanı .....	25
3.1.2. Ağırlıklar.....	25

3.1.3. Toplama Fonksiyonu.....	26
3.1.4. Aktivasyon Fonksiyonu.....	26
3.1.5. Çıktı Elemanı.....	26
3.2. Yapay Sinir Ağlarının Tipleri.....	27
3.2.1. Perseptron.....	29
Örnek yapı.....	30
3.2.2. Çok Katmanlı Perseptron.....	30
3.2.3. Geriye Yayınım Ağı (Backpropagation Net).....	32
3.2.4. Hopfield Ağı.....	33
3.2.5. Kohonen Özellik Haritası.....	34
3.2.6. Hücresel Yapay Sinir Ağı Modeli.....	36
3.3. Öğrenme.....	36
3.3.1. Öğrenmenin Genel Yapısı.....	36
3.3.2. Öğrenme Türleri.....	38
3.3.3. YSA da Kullanılan Önemli Öğrenme Algoritmaları.....	39
3.4. Temel Yapay Sinir Ağı Algoritmaları.....	40
3.4.1. İleri Yayınım.....	40
3.4.2. Geri Yayınım.....	41
4. MALİYET/ÜRETİM MİKTARI İLİŞKİSİNİN BELİRLENMESİ.....	47
4.1. TOPRAK Sağlık Ürünleri Fabrikasının Tanıtımı.....	47
4.2. Maliyet/Üretim Miktarı İlişkisinin Belirlenmesi.....	48
4.2.1. Uygulamada Kullanılan Program (Neurak v1.1).....	49
4.2.2. Verilerin Normalizasyonu, Ağın Eğitilmesi ve Testi.....	49
SONUÇ ve ÖNERİLER.....	54
KAYNAKLAR.....	58
EKLER.....	61
ÖZGEÇMİŞ.....	66

## KISALTMALAR

**YZ** : Yapay Zeka

**YSA** : Yapay Sinir Ağları

**HYSA** : Hücresel Yapay Sinir Ağı



## ŞEKİLLER LİSTESİ

	Sayfa No
Şekil 2.1. Beyin ve Sinir Sisteminin Temel Elemanları.....	14
Şekil 2.2. Sinir Hücresini Oluşturan Elemanlar .....	14
Şekil 3.2. Üç Katmanlı Bir Sinir Ağı Yapısı.....	24
Şekil 3.3. Çeşitli YSA Modelleri .....	28
Şekil 3.4. Bir Geri Yayınım Ağının yapısı.....	43
Şekil 3.5. Bias Değerinin Kullanımı .....	45
Şekil 4.1. 50-500. İterasyon Arası Öğrenme Oranı.....	51
Şekil 4.2. 750-2000. İterasyon Arası Öğrenme Oranı.....	52
Şekil 4.3. 2500-20000. İterasyon Arası Öğrenme Oranı.....	52





## TABLULAR LİSTESİ

	Sayfa No
Tablo 1.1. Yapay Zeka ve Klasik Programlama Karşılaştırılması .....	9
Tablo 3.1. Geliştirilen YSA Modelleri ve Özellikler .....	28
Tablo 3.2. Perseptronun Özellikleri .....	30
Tablo 3.3. Çok Katmanlı Perseptronun Özellikleri.....	31
Tablo 3.4. Geriye Yayınım Ağı Özellikleri.....	32
Tablo 3.5. Hopfield Ağının Özellikleri .....	34
Tablo 3.6. Kohonen Özellik Haritasının Özellikleri .....	35
Tablo 4.1. Çeşitli İterasyonlara Karşılık Gelen Hata Oranları .....	52
Tablo 4.2. Test Setinde Kullanılan Veriler.....	53



## ÖZET

**Anahtar Kelimeler:** Yapay Zeka, Yapay Sinir Ağları, Geriye Yayınım, Maliyet Tahmini

Yapay Zeka başlığı altında, son yıllarda üzerinde en çok araştırma yapılan dallardan birisi Yapay Sinir Ağı (Artificial Neural Networks) modelleridir. YSA arařtırmaları; optimizasyon, kontrol, görüntü ve imaj işleme, konuşulan dili anlama ve ayırma, doğal dil işleme ve tahmin gibi birçok alanla bağlantılıdır.

Yapay sinir ağlarının (YSA) ilham kaynağı biyolojik beynin gücü, esnekliğı ve duyarlılığıdır. YSA, beynin temel biyolojik bileşenlerinden sinir hücreleri, sinapsları ve dendritlerin matematiksel modeli olup, basit matematiksel elemanlardan oluşmaktadır. YSA'da öğrenme işlemi destekli (supervised) ve desteksiz (unsupervised) olmak üzere iki şekilde gerçekleşir. Gözetimli öğrenme işleminde, her girdi seti için bir çıktı seti gereklidir ve her ikisi birlikte öğrenme setini oluşturur. Genellikle belirli sayıdaki bu öğrenme çiftlerinin YSA'a tanıtılması ile öğrenme gerçekleştirilir. Öğrenme sürecinde; YSA'a bir girdi seti verilir ve çıktısı hesaplanır. Hesaplanan bu çıktı ile mevcut çıktı seti değeri arasındaki sapmayı en küçükleyen bir algoritma uyarınca istenilen sapma düzeyine ulaşılan dek YSA ağırlıkları değiştirilir. Böylece YSA eğitilmiş, dolayısıyla ağırlıklar en iyi değerlerini almış olur.

Bu çalışmada; Yapay Zeka tanımından, Yapay Sinir Ağlarının yapısından, bileşenlerinden, çeşitlerinden ve öğrenme türlerinden bahsedilmiştir. Bu çerçevede; TOPRAK SAĞLIK ÜRÜNLERİ fabrikası için mevcut ekonomik koşullar ve maliyetler değiştiğinde üretim miktarının nasıl değiştiğı ile ilgili olarak bir Yapay Sinir Ağı uygulaması yapılmıştır.

## SUMMARY

### **Artificial Neural Networks and A Cost Forecasting Application In An Enterprise**

**Keywords :** Artificial Intelligence, Artificial Neural Networks, Back Propagation, Cost Forecasting

Nowadays, the one of sections which are studied about is Artificial Neural Network (ANN) Models. ANN researchs are related to most field like optimisation, control, image processing, meaning and seperating language, natural language and forecasting.

The inspiration of the ANNs is the power, elasticity and sensitivity of the Biological Brain. ANN is the Mathematical Model of the nerve cells, sinaps and dentrits which are the main biological components of the Brain. ANN is formed from simple mathematical elements.

There are two kinds of learning processes in ANN; supervised and unsupervised. In the supervised learning process, the output set necessary for each input set, and both of them form the learning set. Usually, learning is used to realize by introduced to these pairs (input/output sets) to ANN. In the learning process, firstly, the input sets are given to ANN, and the output of them are computed. Afterwards, ANN change the weights, until the desired convergence criteria level between the computed outputs and the real outputs is proved. As a result, ANN is trained and the weights at the most suitable values.

In this study, An Artificial Intelligence, Structure of the ANN, Components of the ANN, Types of the ANN and Learning Strategies were described. And an application was carrioud out within context of Cost/Production Quantity relation in the TOPRAK SAGLIK URUNLERI factory.

## GİRİŞ

Bu çalışmada Yapay Zekanın teknolojilerinden biri olan Yapay Sinir Ağları anlatılmıştır.

Araştırmaların gelişimi göstermiştir ki; Yapay Zeka uygulamaları artık günümüzde kayda değer sonuçlar vermekte ve tıp, işletme , bankacılık, finans gibi alanlarda verimli bir şekilde kullanılmaktadır.

Bir işletmede uygun kararların alınabilmesi için o işletmedeki faaliyetler arasındaki ilişkilerin doğru tesbit edilmesi gereklidir. Örneğin, birim ürün maliyeti bütün işletmeler çok önemli bir parametredir ve bunun azaltılması pazarda o işletmeye büyük avantajlar sağlayacaktır. Üretilen bir ürünün maliyetini etkileyen çeşitli unsurlar vardır. İşletmedeki istikrar, işletmenin bulunduğu ülkedeki ekonomik durum, çalışan motivasyonu (işçi ücretlerine zam yaparak üretim miktarını arttırabilir), bakım faaliyetlerinin ne seviyede olduğu (minimum fire ve maksimum verimi verecek optimum bakım maliyeti) gibi bizim değiştirebileceğimiz, fakat en uygununu bulmakta zorlanacağımız birçok parametre vardır ve bunların hepsi de maliyetle ilişkilidir. Bu ilişkiler herhangi bir matematiksel ifadeyle açıklanamaz. Bir işletmedeki yöneticiye “Birim ürün maliyetinizi enazlayacak bakım maliyetiniz ne seviyededir?” sorusu sorulsa neredeyse hiçbiri bu soruya tam olarak bir cevap veremez.

İşte bu çalışmada TOPRAK Sağlık Ürünleri fabrikasında maliyet ve üretim miktarı arasındaki ilişkiyle ilgili bu ve buna benzer soruların cevabı aranmıştır.

# 1. YAPAY ZEKA

## 1.1. Yapay Zekanın Tanımı ve İçeriği

YZ nin konusu zeki davranışlar ile ilgilidir. Zekanın genel olarak şöyle bir tanımı verilebilir: “Zeka; olaylar, öneriler, bilgiler ve tüm bu hadiseler arasındaki ilişkileri anlayabilme yeteneğidir”.

YZ geniş bir alandır ve farklı insanlar için farklı anlamlar ifade eder. İnsan zekasına ihtiyaç duyulan görevleri yerine getiren bilgisayarlarla ilgilidir de denilebilir. Bununla birlikte, zekaya ihtiyacı olan (karmaşık bir aritmetik problemi gibi) problemler vardır ki bilgisayar bunları kolayca çözebilir. Bunun aksi de düşünülebilir; insanların neredeyse düşünmeksizin yaptığı işler vardır (bir yüzü tanımak gibi), bunları otomatikleştirmek ise oldukça zordur. Birinci örnek yapay zeka değildir, yapay zeka ikincisi gibi zor işlerle uğraşır.

İnsanlar farklı nedenlerden dolayı insan zekasını otomatikleştirmek isteyebilirler. Sebeplerden biri insan zekasını daha iyi anlamak için olabilir. Örneğin; insan davranışı açısından taklit etme girişiminde bulunacağımız programlar yazarak insan zekasının teorilerini artırabilir ve test edebiliriz. Başka bir sebep de, daha zeki programlar ve makinelere sahip olalım diye olabilir.

Bilinen herhangi bir algoritmik çözümü olmayan herhangi bir problem YZ için bir problem olarak kabul edilebilir. Burada algoritma sözcüğünden kastettiğimiz, makul bir zamanda bir bilgisayar tarafından uygulanabilir olan, hassaslıkla yapılan ardışık özellikli operasyonlardır. Satranç oyununu düşündüğümüzde hamle kombinasyonlarının sayısını kim bilebilir? Bunları hesaplamak ne kadar sürer? Satranç oynamak için algoritmalara ihtiyaç duyulmaz. Tıpkı, tıbbi bir teşhis yapmak, yazının bir parçasını özetlemek veya ana dilden başka bir dile çeviri yapmak için algoritmalara ihtiyaç duyulmadığı gibi. Yapay Zeka bu tip algoritmalarla çözülemeyen problemlerle ilgilenir.

Yapay Zeka ; insanın düşünme yapısını anlamak ve bunun benzerini ortaya çıkaracak bilgisayar işlemlerini geliştirmeye çalışmak olarak tanımlanabilir. Yani programlanmış bir bilgisayarın düşünme girişimidir. Daha geniş bir tanıma göre ise, YZ; bilgi edinme, algılama görme, düşünme ve karar verme gibi insan zekasına özgü kapasitelerle donatılmış bilgisayarlardır.

YZ'nın şöyle bir tanımı da yapılmıştır. “YZ, insan zekasını gerektiren durumların, bilgisayarları programlama yolu ile sonuçlandırıldığı bir bilgisayar bilim dalıdır”[Gevartel, 1985: s4].

Minsky (1968) YZ'yi; insan tarafından yapıldığı zaman zeki davranış gerektiren işlevlerin, programlar yardımı ile makineler tarafından yaptırılmasına imkan sağlayacak sistemler olarak tanımlamıştır.

Yapay zeka konusundaki ilk çalışma McCulloch ve Pitts tarafından yapılmıştır. Bu araştırmacıların önerdiği, yapay sinir hücrelerini kullanan hesaplama modeli, önermeler mantığı, fizyoloji ve Turing'in hesaplama kuramına dayanıyordu. Her hangi bir hesaplanabilir fonksiyonun sinir hücrelerinden oluşan ağlarla hesaplanabileceğini ve mantıksal “ve” ve “veya” işlemlerinin gerçekleştirilebileceğini gösterdiler. Bu ağ yapılarının uygun şekilde tanımlanmaları halinde öğrenme becerisi kazanabileceğini de ileri sürdüler. Hebb, sinir hücreleri arasındaki bağlantıların şiddetlerini değiştirmek için basit bir kural önerince, öğrenebilen yapay sinir ağlarını gerçekleştirmek de olası hale gelmiştir.

YZ çalışmaları 1950 li yıllarda başlamış ve 1960 lı yılların sonuna kadar hızlı bir şekilde devam etmiştir. Zeki davranışı üretmek için bu çalışmalarda kullanılan temel yapılardaki bazı önemli yetersizliklerin de ortaya konması ile birçok araştırmacı çalışmalarını durdurdu. Buna en temel örnek; sinir ağları konusundaki çalışmaların Minsky ve Papert'in 1969'da yayınlanan Perceptrons adlı kitaplarında tek katmanlı algılayıcıların bazı basit problemleri çözemeyeceğini gösterip aynı kısırlığın çok katmanlı algılayıcılarda da beklenilmesi gerektiğini söylemeleri ile bıçakla kesilmiş gibi durmasıdır.

Her sorunu çözecek genel amaçlı program yerine belirli bir uzmanlık alanındaki bilgiyle donatılmış programlar kullanma fikri yapay zeka alanında yeniden bir canlanmaya yol açtı. Kısa sürede uzman sistemler adı verilen bir metodoloji gelişti. Fakat burada çok sık rastlanan tipik bir durum, bir otomobilin tamiri için önerilerde bulunan uzman sistem programının otomobilin ne işe yaradığından haberi olmamasıydı.

İnsanların iletişimde kullandıkları Türkçe, İngilizce gibi doğal dilleri anlayan bilgisayarlar konusundaki çalışmalar bu sıralarda hızlanmaya başladı. Doğal dil anlayan programların dünya hakkında genel bilgiye sahip olması ve bu bilgiyi kullanabilmek için genel bir metodolojisi olması gerektiği belirtilmekteydi.

Uzman dizgelerin başarıları beraberinde ilk ticari uygulamaları da getirdi. Yapay zeka yavaş yavaş bir endüstri haline geliyordu. DEC tarafından kullanılan ve müşteri siparişlerine göre donanım seçimi yapan R1 adlı uzman sistem şirkete bir yılda 40 milyon dolarlık tasarruf sağlamıştı. Birden diğer ülkelerde yapay zekayı yeniden keşfettiler ve araştırmalara büyük kaynaklar ayrılmaya başlandı. 1988'de yapay zeka endüstrisinin cirosu 2 milyar dolara ulaşmıştı.

Bütün bu çalışmaların sonunda yapay zeka araştırmacıları iki guruba ayrıldılar. Bir gurup insan gibi düşünen sistemler yapmak için çalışırken, diğer gurup ise rasyonel karar verebilen sistemler üretmeyi amaçlamaktaydı.

Bu yaklaşımları kısaca inceleyelim:

- İnsan gibi düşünen sistemler

İnsan gibi düşünen bir program üretmek için insanların nasıl düşündüğünü saptamak gerekir. Bu da psikolojik deneylerle yapılabilir. Yeterli sayıda deney yapıldıktan sonra elde edilen bilgilerle bir kuram oluşturulabilir. Daha sonra bu kurama dayanarak bilgisayar programı üretilebilir. Eğer programın giriş/çıkış ve zamanlama davranışı insanlarınkine eşse programın düzeneklerinden bazılarının insan beyninde de mevcut

olabileceği söylenebilir. İnsan gibi düşünen sistemler üretmek bilişsel bilimin araştırma alanına girmektedir. Bu çalışmalarda asıl amaç genellikle insanın düşünme süreçlerini çözümlenmede bilgisayar modellerini bir araç olarak kullanmaktır.

- İnsan gibi davranan sistemler

Yapay zeka araştırmacılarının baştan beri ulaşmak istediği ideal, insan gibi davranan sistemler üretmektir. Turing zeki davranışı, bir sorgulayıcıyı kandırarak kadar bütün bilişsel görevlerde insan düzeyinde başarı göstermek olarak tanımlamıştır. Bunu ölçmek için de Turing testi olarak bilinen bir test önermiştir. Turing testinde denek, sorgulayıcıyla bir terminal aracılığıyla haberleşir. Eğer sorgulayıcı, deneğin insan mı yoksa bir bilgisayar mı olduğunu anlayamazsa denek Turing testini geçmiş sayılır. Turing, testini tanımlarken zeka için bir insanın fiziksel benzetiminin gereksiz olduğunu düşündüğü için sorgulayıcıyla bilgisayar arasında doğrudan fiziksel temastan söz etmekten kaçınmıştır. Burada vurgulanması gereken nokta, bilgisayarda zeki davranışı üreten sürecin insan beynindeki süreçlerin modellenmesiyle elde edilebileceği gibi tamamen başka prensiplerden de hareket edilerek üretilmesinin olası olmasıdır.

- Rasyonel düşünen sistemler

Bu sistemlerin temelinde mantık yer alır. Burada amaç çözülmesi istenen sorunu mantıksal bir gösterimle betimledikten sonra çıkarım kurallarını kullanarak çözümünü bulmaktır. Yapay Zeka 'da çok önemli bir yer tutan mantıkçı gelenek zeki sistemler üretmek için bu çeşit programlar üretmeyi amaçlamaktadır. Bu yaklaşımı kullanarak gerçek sorunları çözmeye çalışınca iki önemli engel karşımıza çıkmaktadır. Mantık, formel bir dil kullanır. Gündelik yaşamdan kaynaklanan, çoğu kez de belirsizlik içeren bilgileri mantığın işleyebileceği bu dille göstermek hiç de kolay değildir. Bir başka güçlük de en ufak sorunların dışındaki sorunları çözerken kullanılması gereken bilgisayar kaynaklarının üstel olarak artmasıdır.

- Rasyonel davranan sistemler



Amaçlara ulaşmak için inançlarına uygun davranan sistemlere rasyonel denir. Bir ajan algılayan ve harekette bulunan bir şeydir. Bu yaklaşımda yapay zeka, rasyonel ajanların incelenmesi ve oluşturulması olarak tanımlanmaktadır. Rasyonel bir ajan olmak için gerekli koşullardan biri de doğru çıkarımlar yapabilmek ve bu çıkarımların sonuçlarına göre harekete geçmektir. Ancak, yalnızca doğru çıkarım yapabilmek yeterli değildir. Çünkü bazı durumlarda doğruluğu ispatlanmış bir çözüm olmadığı halde gene de bir şey yapmak gerekebilir. Bunun yanında çıkarımdan kaynaklanmayan bazı rasyonel davranışlar da vardır. Örneğin, sıcak bir şeye değince insanın elini çekmesi bir refleks harekettir ve uzun düşünce süreçlerine girmeden yapılır. Bu yüzden yapay zekayı rasyonel ajan tasarımı olarak gören araştırmacılar, iki avantaj öne sürerler. Birincisi “düşünce yasaları” yaklaşımından daha genel olması, ikincisi ise bilimsel geliştirme yöntemlerinin uygulanmasına daha uygun olmasıdır.

## **1.2. Yapay Zekanın Karakteristik Özellikleri**

YZ nin karakteristik özelliklerini şu şekilde sıralayabiliriz:

- Bilgiyi kullanma,
- Sembolik bilgi işleme,
- Sezgisel problem çözme,
- Eksik, belirsiz ve tam olmayan bilgi işleme,
- Açık olarak formülize edilemeyen problemleri çözebilme,
- Tecrübe ile öğrenme,
- Yeni durumlara adapte olabilme,
- En fazla geliştiren kadar zeki olma,
- Hata yapabilme.

## **1.3. Yapay Zeka Teknolojileri**

Yapay Zeka teknolojilerini şu şekilde sıralayabiliriz:

- Uzman Sistemler (Uzmanlık bilgisini işler),
- Yapay Sinir Ağları,
- Genetik Algoritmalar,
- Bulanık Mantık,
- Vaka Tabanlı Muhakeme,
- Monoton Olmayan Muhakeme,
- Kalitatif Muhakeme,
- Model Tabanlı Muhakeme, vb.

#### **1.4. Yapay Zekanın Kapsamı**

YZ da herhangi bir iyi tanımlanmış metot olmaksızın çalıştığımız alanlar çok çeşitlidir, fakat bunlar genelde iki yaygın karakteristiğe sahiptirler:

- Hepsi (harfler, kelimeler, işaretler, çizimlerle ifade edilen) sembolik bilgilerle ilgilidir.
- Daima çok seçeneklidirler.

YZ araştırmacılarının karşı karşıya kaldıkları ilk problem; bilginin elde edilmesidir. Var olan yazılım ve donanım teknolojileri; görebilen, tat alan, koklayabilen, üretebilen veya nesnelerin fiziksel kullanımlarını ve konuşmayı anlayabilen insan organlarına rakip olmaktan çok uzaktırlar.

#### **1.5. Yapay Zeka Araçlarının Ortak Özellikleri**

YZ; mantık, matematik, psikoloji, dilbilim ve fizik gibi bilim dallarından yoğun olarak etkilenen bir disiplindir. Bütün YZ Sistemleri arasında, bu disiplinin temel özelliklerini daha iyi sınırlandırma olanağı sağlayan ortak noktalar bulunmaktadır. Bu ortak noktaları üç madde halinde ele alabiliriz:

- Sembolik bilgiler kullanma,
- Bilgilerin eksik, yanlış ve çelişkili olduğu durumlarda sonuçlar üretilmesi,

- Bilgi kavramı ile yoğun ilişki.

YZ programları sembolik bilgiler kullanır ve bu bilgileri sezgisel yöntemlerle işler. Bu bilgileri, kavramları, kuralları nesnelere düşünürken bir insanın göz önünde bulundurabildiği değerlerle olayları temsil etmektedir. Sayısal işlem yöntemleri kullanılabilirle beraber sonuçların işlenmesi genellikle sembole dayalı olarak yapılmaktadır. Örneğin 40° C olan hava sıcaklığı 'hava çok sıcak' kavramı ile ifade edilir.

YZ da buluna diğer bir kavram ise sezgisel yöntemler kavramıdır. Program yapmak için belli bir programlama dilinde kodlanmış olan bir algoritma, verilmiş bir problemi çözmeye yönelik işlemler dizisinin eksiksiz betimlemesinden oluşmaktadır. Sezgisel, determinist olmayan yolları izleyen ve başarının garantili olmadığı, ancak çalıştığı zaman genellikle işlem süresinde büyük tasarruf sağlayan bir çözüm yöntemidir. Başarısızlık durumunda tekrar geriye dönüp bir başka çözüm yolunu denemek gerekmektedir. Bir problemin çözümünün sezgisel araştırması, genellikle sadece ihtimali fazla olanları göz önünde bulundurarak çözüm yollarının tamamının kısaltılması olarak görülmektedir. YZ programları sezgisel yöntemler ile algılama, karar verme, muhakeme gibi insana benzer özellikle göstererek problemleri çözer.

YZ'nin bir başka özelliği de işlenen verilerin ve bilgilerin eksik, yanlış ve çelişkili olduğu durumlarda sonuçlar üretmesidir. Bir insan için örneğin tıpta, iş idaresinde, bankacılık sektöründe bu tip durumların söz konusu olduğu düşünüldüğünde problemleri tam olarak çözmeseyse bile etkili yöntemlerle kabul edilebilir sonuçlar üretir.

Önemli bir YZ kavramı da bilgiyle yoğun ilişkidir. Bilgi kavramı YZ'nin hammaddesi sayılabilir. Günümüzde YZ az sayıda ve iyice sınırlanmış alanlarda, bu alanlara özgü çok miktardaki yoğun bilgiye dayanarak problemleri çözebilen sistemleri ortaya koyar. Bu bilgiler bir programlama dilinde prosedürler biçiminde kodlandıkları zaman, doğal biçimlerine daha yakın olarak YZ sistemine katkıda bulunmaktadır [Turban, 1992: s681-714], [Winstanley & Graham, 1991].

## 1.6. Yapay Zeka ve Klasik Sistemlerin Karşılaştırılması

YZ programları, insanın akıl davranışını kabaca taklit ettiği gibi, aynı zamanda programlarda oluşturulan iç dinamikler sayesinde insana benzer karakteristik özellikler gösterir. Bazen insanın karşısına iyi tanımlanmış problemler çıkar. Çözüm algoritması net olan problemlerin aksine bu tip problemlerde sezgisel, algoritmik olmayan yöntemlerle sonuca gidilir. Bu yönüyle YZ programları klasik programlardan bazı farklılıklar ortaya koyar. Tablo 1.1 de YZ ve klasik bilgisayar programcılığı karşılaştırılmıştır [Gevartel, 1985: s4].

Tablo 1.1. Yapay Zeka ve Klasik Programlama Karşılaştırılması

Yapay Zeka Programlama	Klasik Bilgisayar Programlama
Genelde sembolik işlemler yaparlar.	Sayısal işlemler yaparlar.
Sezgisel araştırma yapısına sahiptir.	Algoritmik araştırma yapısı vardır.
Belli tanım aralığındaki belirli problemleri çözebilir.	Belli tanım aralığındaki yalnızca bir problemi çözebilir.
Genellikle değiştirmek, güncelleştirmek ve genişletmek kolaydır.	Değiştirilmesi zordur.
Sonuçlar kesin olmayabilir.	Sonuçlar kesindir.
Tatminkar sonuçlar genelde kabul edilebilir.	Mümkün olan en iyi sonucu verir.
Bilgiyi işler.	Veriyi işler

Kısacası; Yapay Zeka programları karmaşıklık, belirsizlik ve çelişki içeren, sezgisel yöntemlerle çözümler üretmeyi hedefleyen; klasik programlara göre daha esnek programlardır.

### 1.7. Yapay Zekanın Uygulamaları

Yapay Zekanın uygulama alanlarını şu şekilde sıralayabiliriz:

- Yorumlama : sensörlerden gelen bilgilerin yorumlanması,
- Tahmin : bilinen durumlardan muhtemel durumları tayin etme,
- Tasarım : kısıtlara bağlı olarak nesnelere tasarlanması,
- Planlama : aktivitelerin belirlenmesi,
- Gözleme : gözlemlerin beklentiler ile karşılaştırılması,
- Tamir : hataların düzeltilmesi,
- Talimat oluşturma : eylem ve planların uygulanmasını koordine etme,
- Kontrol : sistem davranışını bütün olarak kontrol etme.

### 1.8. Yapay Zekanın Uygulama Örnekleri

YZ nin uygulama örneklerini şu şekilde sıralayabiliriz:

- Robotik : yol ve önerge planlama (Navigation-Gemi İşletmesi); bilgisayar görme; imalat kontrol (CAM); imalat teşhis sistemleri (Manufacturing Diagnostic Systems); imalat; imalat çizelgeleme
- Uzman Sistemler : tıbbi teşhis (MYCIN); savaş alanı yönetimi (Pilot's Associate); jeolojik başarı (PROSPECTOR); bilgisayar konfigürasyon(XCON); güç sistemleri kontrolü; üretim planlama ve çizelgeleme; tasarım
- Oyun Oynama (Game Playing)
- Ana Dili Anlama
- Teorem İspatlama
- Bilgisayar Yardımı ile Eğitim/Öğrenme

- Otomatik Karar Verme
- Otomatik Yazılım Oluřturma

### 1.9. Yapay Zekanın Uygulamalarına Bazı Örnekler

YZ nin uygulamalarına örnek olması açısında řu örnekler verilebilir:

- Kalite Kontrol (Pham ve Öztemel, 1992, 1995, 1996),
- Muayene (Wu and Rodd, 1989),
- Deneysel Tasarım (Öztemel ve Aydın, 1996),
- Proses Planlama (Singh ve Mahanty, 1991),
- Optimizasyon (Davidor, 1991),
- Esnek İmalat Sistemleri (Mellichamp ve Arkadařları, 1990),
- Karar Destek Sistemleri (Schrunder ve Arkadařları, 1994),
- Montaj Hattı Uygulamaları (Dađlı ve Vellanki, 1993),
- Bakım (Mojstorovic, 1990),
- İř Çizelgeleme (Cedimođlu, 1993; Öztemel ve Düğenci, 1996; Tařgetiren, 1996),
- Hata Teřhisi (Roy, 1991),
- Grup Teknolojisi (Kusiak ve Chung, 1991),
- Benzetim (Sharma ve Arkadařları, 1994),
- Bilgi Sistemleri (Etzioni ve Weld, 1994),
- Veri Analizi (Lin ve Lin, 1993), vb.

## 2. YAPAY SİNİR AĞLARI

Yapay Sinir Ağları (Artificial Neural Networks – YSA) teknolojisi bilgisayar dünyasında insan beyninin ve sinir sisteminin davranışlarını taklit etme esası üzerine kurulmuş yeni bir bilgi işleme yaklaşımıdır. Bu ağlar birbirlerine paralel olarak bağlanmış işlem elemanlarından yapay sinir hücresi ve onların hiyerarşik bir organizasyonundan oluşurlar. YSA, daha çok biyolojik sistemlerin hücreler üzerinde dağıtılmış bilgiyi paralel olarak işleme özelliklerinden yararlanan bir mekanizmadır. Hücreler birbirine bağlı ve paralel çalıştıkları için bazılarının işlevini yitirmesi ile sinir sistemi fonksiyonunu yitirmez. Bu ağların temel amacı gerçek dünyadaki nesnelere ve olaylara karşı biyolojik sinir sisteminin davrandığı gibi davranmaktır. Fakat günümüzde bu noktaya ulaşıldığını söylemek mümkün değildir [Lippmann, 1987].

YSA farklı branşlara sahip araştırmacıların yoğun ilgi gösterdikleri bir araştırma ve uygulama alanıdır. İnsanın biyolojik bazı yapı taşlarından esinlenerek geliştirilen YSA na yoğun ilginin nedeni, klasik yöntemlerin başarısız olduğu veya zayıf kaldığı problemlerde, paralel bir yapıya sahip sinir ağlarının gösterdiği başarı ve üstünlüktür.

YSA nın mühendislik başta olmak üzere birçok bilimsel alanda, karmaşık ve belirsiz veriler altında problemlere çözümler ürettikleri ispatlanmıştır [Tosun, 1997]. Algoritmasız, tamamıyla paralel, adaptif, öğrenebilen ve paralel dağıtılmış bir hafızaya sahip olma, bu sistemlerin ana özelliklerinin başında gelmektedir. bu teknoloji, insana benzer yaklaşımlarla robotların oluşturulmasında ilk adım olarak görülebilir [Lippmann, 1987: s4-24].

YSA ile ilgili çalışmalara 1940 larda başlanılmış ama hem teknolojik yetersizlik ve hem de beyin karmaşık yapısından dolayı çalışmalar yavaş ilerlemiştir. McCulloch ve Pitts 1943 yılında yayınladıkları makalede, basit bir yapıya sahip sinir ağlarının bile matematiksel ve mantıksal bazı işlemleri gerçekleştirebileceğini ortaya koymuşlardır [McCulloch, 1943: s115-133]. 1960 lı yıllarda Minsky, perseptronun mantıksal XOR işlemini gerçekleştiremeyeceğini gösterdikten sonra YSA üzerine

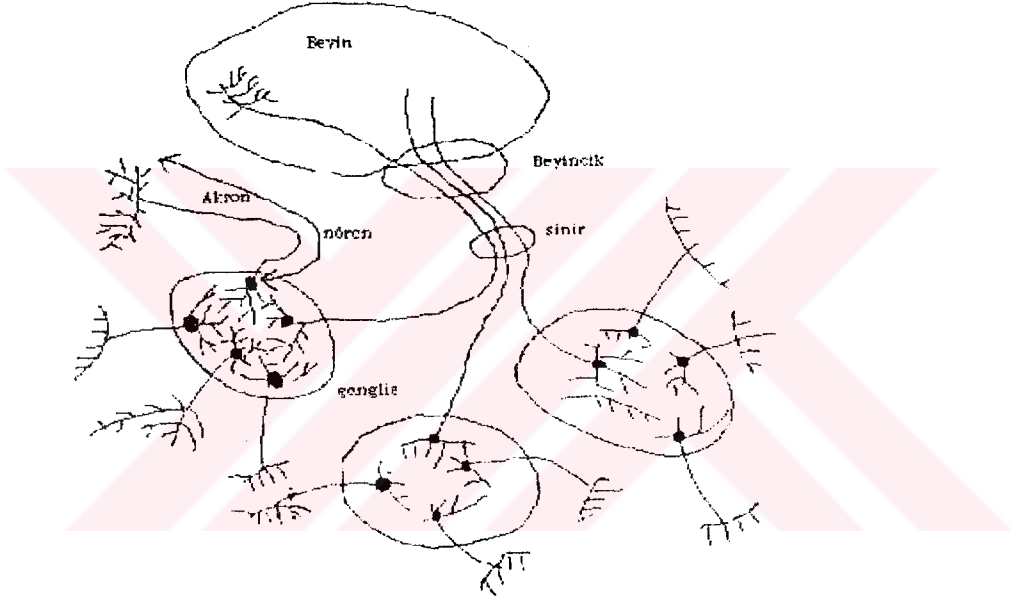
yapılan çalışmalarda bir duraksama meydana gelmiştir [Mehra & Benjamin, 1992]. Hopfield 1982 yılında yayınladığı makale ile YSA üzerinde yoğun çalışmaların tekrar başlamasını sağlamıştır [Hopfield, 1982]. Ayrıca son yıllardaki teknolojik ve nörofizyolojik gelişmeler nedeni ile elde edilen başarılı sonuçlar dikkatleri yeniden YSA na çevirmiştir. YSA, olayları öğrenerek karar verme prensibi üzerine kurulmuşlardır. Öğrenme, zeki sistemlerin bilgi, yetenek ve tecrübelerini artırma olayı olarak düşünülebilir. Farklı tanımlar yapılmakla birlikte en genel şekli ile öğrenme; sistemlerin aynı veya benzeri işleri yaptıklarında, o işi veya işleri bir önceki yapıldığı şekliyle daha verimli ve etkin olarak gerçekleştirecek değişiklikleri oluşturma süreci olarak tanımlanmaktadır. Araştırmacıların zeki robotlar veya benzeri nesnelere oluşturma yönündeki çalışmalarının önemli bir noktasını da bu öğrenme sürecinin bilgisayarlaştırılması oluşturur. Dolayısıyla ortaya atılan öğrenme metot ve yöntemlerinin sayısı her geçen gün artmaktadır. Bunun temel nedeni insanoğlunun programlanabilir makineler yerine, eğitilebilir makinelere sahip olma arzusudur. YSA insanoğlunun bu merakını giderebilmek için başlatılan çalışmaların ortaya çıkarttığı bir tür bilimsel öğrenme mekanizmasıdır [Lippmann, 1987: s4-24]. YSA her geçen gün ilgi odağı olmaktadır da insan beyninin fonksiyonları ile ilgili çalışmalar pek de yeni sayılmaz. Bu ağların paralel yapıları ve bilgisayarları geleneksel yöntemlerden çok daha farklı kullanarak özellikle seri bilgisayarlarda bilinen yöntemlerle yapılması mümkün olmayan veya çok zor olan bir takım işlevleri rahatlıkla yapmaları önemlerini daha da arttırmaktadır. YSA nı daha iyi kavramak için önce biyolojik sinir ağlarının genel olarak bilinmesinde yarar vardır [Öztemel, 1996, s134-140].

## **2.1. Biyolojik Sinir Ağları**

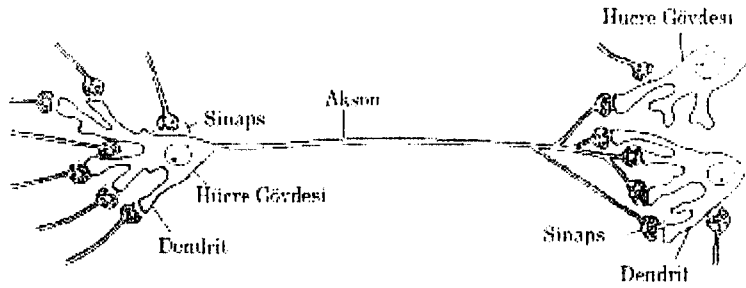
Biyolojik beynin en önemli özelliklerinden birisi de öğrenme olayıdır. İnsanlar ve hayvanlar sürekli olarak içlerinde buldukları çevre ile ilişkiler neticesinde bir öğrenme işlemi içerisinde bulunurlar. Öğrenilen her yeni bilgi beynin fonksiyonlarını etkileyecek davranışlarda kendini gösterir. Bu özellik YSA nın da temelini oluşturmaktadır [Öztemel, 1996, s134-140].



Şekilde 2.1’de beyin ve sinir sisteminin temel yapı elemanları görülmektedir. Beyin ve sinir sisteminde fiziksel katmana bakıldığında, işlemci, sinyal iletim ortamı ve yol verici olarak, sinir sisteminin temel ögesi olan sinir hücresi, ya da sinir hücresi görülmektedir. Sinir hücresini oluşturan Dendrit, hücre gövdesi, akson ve akson uçları (sinaps) şekil 2.2’de gösterilmiştir. Dendritler sinaptik sinyalleri girdi olarak almakta, hücre gövdesi bu sinyalleri bilindiği kadarıyla analog bir yöntemle işlemekte ve üretilen denetim sinyali ya da sinyalleri aksonlar aracılığı ile denetlenecek hedef hücrelere iletilmektedir.



Şekil 2.1. Beyin ve Sinir Sisteminin Temel Elemanları



Şekil 2.2. Sinir Hücresini Oluşturan Elemanlar

Bir sinir hücresi sinir ağlarının en temel elemanlarından biri olup sinir sisteminde fonksiyon ve görevlerine göre değişik şekil ve büyüklükte olabilir. Şekil 2.2 de bir hücrenin bir ucunda “dendrit” adı verilen ve hücreye diğer hücrelerden veya dış dünyadan bilgiler (sinyaller) getiren bağlantı elemanları, diğer ucunda ise bir life benzeyen, “akson (axon)” adı verilen ve hücrelerden diğerlerine veya dış dünyaya bilgiler taşıyan bağlantı elemanları görülmektedir. Akson daha sonra diğer hücrelerle birleşme esnasında dağınık dallara ayrılmaktadır. İki uçtaki bağlantı noktalarının elektro fizyolojik olarak hücrelerdeki bilgileri işlemede önemli yeri vardır.

Tipik bir sinir hücresi, hücre gövdesi ve dendritleri üzerine dış kaynaklardan gelen elektrik darbelerinden üç şekilde etkilenir. Gelen darbelerden bazıları sinir hücrelerini uyarır, bazıları bastırır, geri kalanı da davranışında değişikliğe yol açar. Sinir hücresi yeterince uyarıldığında çıkış kablosundan (aksonundan) aşağı bir elektriksel işaret göndererek tepkisini gösterir. Genellikle bu tek akson üzerinde çok sayıda dallar olur. Aksondan inmekte olan elektrik işareti dallara ve alt dallara ve sonunda başka sinir hücrelerine ulaşarak onların davranışını etkiler. Sinir hücresi, çok sayıda başka sinir hücrelerinden genellikle elektrik darbesi biçiminde gelen verileri alır. Yaptığı iş bu girdilerin karmaşık ve dinamik bir toplamını yapmak ve bu bilgiyi aksonundan aşağı göndererek bir dizi elektrik darbesi biçiminde çok sayıda başka sinir hücresine iletmektir. Sinir hücresi, bu etkinlikleri sürdürmek ve molekül sentezlemek için de enerji kullanır fakat başlıca işlevi işaret alıp işaret göndermek, yani bilgi alışverişidir .

Sinyaller bir hücrenin aksonundan diğerinin dendritine gönderilir. Bir akson birden fazla dendrit ile bağlantı yapabilir. Bağlantının yapıldığı yere “sinaps” adı verilir. Hücreler, elektrik sinyallerini hücre duvarlarındaki gerilimi değiştirerek üretirler. Bu ise hücrenin içinde ve dışında bulunan dağılmış iyonlar sayesinde olmaktadır. Bu iyonlar sodyum, potasyum, kalsiyum ve klorin gibi iyonlardır. Bir hücre diğer hücreye elektrik sinyalini bu kimyasal iyonlar sayesinde transfer eder. Bazı iyonlar elektrik ve manyetik kutuplaşmaya neden olurken bazıları da kutuplaşmadan kurtulup hücre zarını geçerek iyonların hücreye geçmesini sağlarlar. Sinyaller hücrenin etkinliğini belirlerler. Bir hücrenin etkinliği hücreye gelen sinaps sayısı, sinapslardaki iyonların konsantrasyonu ve sinapsın sahip olduğu güç olmak üzere üç faktöre bağlıdır. Bir

hücre sahip olduğu uyarı miktarınca diğer hücreleri etkilemektedir. Bazı hücreler diğerlerinin uyarılarını pozitif yönde, bazı hücrelerde negatif yönde etkilemektedir. İnsan beyni bu şekilde çalışan sayısız hücrenin bir araya gelmesinde oluşmaktadır[Öztemel, 1996, s134-140].

## 2.2. Yapay Sinir Ağları Nedir?

Bir YSA; biyolojik sinir sistemlerinden etkilenecek yapılan beyin, proses işleme gibi bir bilgi işleme paradigmasıdır. Bu paradigmanın anahtar kelimesi; bilgi işleme sistemlerinin yeni yapısıdır (novel structure).

YSA da tıpkı insanlar gibi örneklerden öğrenir. Bir YSA, bir öğrenme prosesi aracılığı ile spesifik bir uygulama için konfigüre edilir, örneğin pattern recognition veya veri sınıflandırma gibi. Biyolojik sistemlerde öğrenme, sinir hücreleri arasında olan sinaptik bağlantılar için düzenlemeler içerir. Bu YSA için de doğrudur.

Yapay sinir ağlarının temel yapısı, beyne, sıradan bir bilgisayarinkinden daha çok benzemektedir. Yine de birimleri gerçek sinir hücreleri kadar karmaşık değil ve ağların çoğunun yapısı, beyin kabuğundaki bağlantılarla karşılaştırıldığında büyük ölçüde basit kalmaktadır.

Yapay sinir ağlarındaki her bir işlem birimi, basit anahtar görevi yapar ve şiddetine göre, gelen sinyalleri söndürür ya da iletir. Böylece sistem içindeki her birim belli bir yüke sahip olmuş olur. Her birim sinyalin gücüne göre açık ya da kapalı duruma geçerek basit bir tetikleyici görev üstlenir. Yükler, sistem içinde bir bütün teşkil ederek, karakterler arasında ilgi kurmayı sağlar. Yapay sinir ağları araştırmalarının odağındaki soru, yüklerin, sinyalleri nasıl değiştirmesi gerektiğidir. Bu noktada herhangi bir formdaki bilgi girişinin, ne tür bir çıkışa çevrileceği, değişik modellerde farklılık göstermektedir. Diğer önemli bir farklılık ise, verilerin sistemde depolanma şeklidir. Sinirsel bir tasarımda, bilgisayarda saklı olan bilgiyi, tüm sisteme yayılmış küçük yük birimlerinin birleşerek oluşturduğu bir bütün evre temsil etmektedir.

Ortama yeni bir bilgi aktarıldığında ise, yerel büyük bir deęişiklik yerine tüm sistemde küçük bir deęişiklik yapılmaktadır .

Yapay sinir aęları beynin bazı fonksiyonlarını ve özellikle öğrenme yöntemlerini benzetim yolu ile gerçekleştirmek için tasarlanır ve geleneksel yöntem ve bilgisayarların yetersiz kaldığı sınıflandırma, kümeleme, duyu-veri işleme, çok duyulu makine gibi alanlarda başarılı sonuçlar verir. Yapay sinir aęlarının özellikle tahmin problemlerinde kullanılabilmesi için çok fazla bilgi ile eğitilmesi gerekir. Aęların eğitimi için çeşitli algoritmalar geliştirilmiştir .

YSA nın karakterize edilmesinin en uygun yöntemi, anlama ve hesaplamanın diğer modelleri ile YSA nın ayırt edilmesidir. YZ ve muhakeme ile ilgili bilimlerde kullanılan yaygın düşünme modelleri sonuç çıkarmada bilgiyi ayırır. Aynı şekilde yaygın olarak bilgisayarları programlamak için kullanılan hesaplamanın modeli veri ve programlara ayırır. YSA bu tip farklılıkları derinleştirir. Şöyle ki ; YSA' nda algoritmalar statik olarak temsil edilir ve YSA nın verileri aęlardaki sistemin dinamikleri ile gösterilir. Hopfield bu durumu 'çıkan hesaplama (emergent computation)' diye ifade etti[Hopfield, 1982]. Başka bir ifade ile aę çıktısı gibi tekrar ortaya çıkmadan önce, fark edilemeyen girdi verisi gerçeğine deęindi.

YSA sembolik ve herhangi bir şeyle ilgili bilgiyi ikiye ayırır. Buradaki fark bilgiyi temsil etme yöntemindedir. Bilgi ilk şekilde sembollerle gösterilir. İkinci şekilde ise bilgi doğrudan temsil edilerek, hiçbir işleme tabi tutulmadan doğrudan işlev görür[Kocabaş, 1992: s4-7].

YSA, lineer olmayan sistemlere ve proses deęişikliğine adapte olabilir, bilgi sensörleri ile birleşebilir ve problemlerin çözülmesinde etkinlik gösterir. Bu gibi özellikleri ve diğer güçlü yönleri YSA nın, geleceğin günümüze göre çok daha modern ve kompleks üretim sistemlerinde kullanılmasına olanak verir [White & Sofge, 1992].

### **2.3. Sinir Ağları Niçin Kullanılır?**

Sinir ağları ve karmaşık verilerden türetilen yetenekleri; ya diğer bilgisayar teknikleri yada insanlar tarafından fark edilerek ortaya çıkan çok karmaşık trendleri ve numuneleri (pattern) elde etmek için kullanılır. Eğitilen bu sinir ağı; verilen analizdeki bilgi kategorisinde bir “uzman” olarak düşünülebilir. Bu uzman daha sonra, “what-if” sorularına cevap veren, ilgi alanları için yeni durumlar oluşturan projeksiyonlar sağlamak için kullanılabilir.

### **2.4. Geleneksel Bilgisayarlar ve Yapay Sinir Ağları – Bir Karşılaştırma**

Sinir ağları; problem çözümünde geleneksel bilgisayarlardan daha farklı bir yaklaşım içindedirler. Geleneksel bilgisayarlar problem çözümü için algoritmik bir yaklaşım kullanırlar.

Sinir ağları bilgiyi insan beynine benzer bir şekilde işler. Ağ; spesifik bir problemi çözmek için paralel olarak çalışan, birbirleriyle bağlantılı çok sayıda proses elemanından (sinir hücresinden) oluşur. Sinir ağları örneklerden öğrenir. Spesifik bir görevi yerine getirmek için programlanmayabilir. Örnekler dikkatlice seçilmelidir, aksi takdirde zamanı boşa harcamış oluruz.

Diğer yandan geleneksel bilgisayarlar problem çözümünde kognitif bir yaklaşım kullanırlar.

YSA ve geleneksel algoritmik bilgisayarlar rekabet halinde değildirler ve birbirlerine de benzemezler. Geleneksel bilgisayarların görevleri aritmetik operasyonlar gibi algoritmik yaklaşımlar için daha uygundur. Algoritmik olmayan yaklaşımlar ise YSA için daha uygundur. Daha zor görevlerde daha fazla verim alabilmek için bu iki yaklaşımın bir kombinasyonunu kullanan sistemlere ihtiyaç duyulur (normalde geleneksel bir bilgisayar sinir ağlarını desteklemek için kullanılır).

## 2.5. Yapay Sinir Ağlarının Avantajları

YSA'nın iki temel avantajı vardır :

- Lineer veya değil; basit veya karmaşık herhangi bir fonksiyonu gösterebilme yeteneği.
- "hatalı geriye yayılım (error backpropagation)" tarafından sunulan örneklerden öğrenme yeteneği (Le Cun, McClelland, 1986; Werbos, 1974). Ayrıca model oluşturma ve öğrenme otomatiktir.

Bununla birlikte sinirsel model oluşturma veri analizine aittir ve bir sihir değildir. Veri açık ve yeterli miktarda olmalıdır.

YSA'nın avantajlarını genel olarak şu şekilde sıralayabiliriz :

- Herhangi bir fonksiyonel bağımlılık için hesaplama yeteneği. Ağ; herhangi bir gecikmeye sebep olmadan yapıyı keşfeder (=öğrenir, modeller).
- Aracı olmadan, kayıt etmeden, basitleştirme veya soruyu yorumlama olmadan model için veriyi düzgün bir biçimde çalıştırır.
- Veride güvenirsizlik veya azaltma olmasına aldırmaz.
- Değişkenlerin önceden bilinmesi şartı yoktur.
- Kullanım kolaylığı vardır, geleneksel istatistiksel analize göre insan çalışması daha azdır.
- Segmentasyonda (parçalama); ağ, her bir sınıfta kaç tane küme olacağını kendi kendine belirler.
- Acemi kullanıcılar için; öğrenme fikri, çok değişkenli istatistiklerin karmaşıklığını anlamaktan daha kolaydır.
- Kullanım hızı; 1GHz bilgisayarda birkaç mili saniyedir.
- Uzaysal ilişkiler kolay analiz edilir ve modellenir.
- Son model; sürekli, türetilir ve sonraki çalışmalar için de elverişlidir.

## 2.6. Yapay Sinir Ağlarının Özellikleri

YSA'nın hem yaygın olarak kullanılmasını sağlayan, hem de geleneksel bilgi işleme metodlarında ayrılan özellikleri vardır. YSA'nın bir takım özellikleri kullanılan sinir ağı modeline bağlı olsa da bunun yanında bir takım genel özellikleri de bulunmaktadır. Bu özellikler:

- YSA, olaylar arasındaki ilişkileri belirli bir algoritmaya dayanarak çözmek yerine, o ilişkiyi gösteren örnekleri incelemek suretiyle çözümler üretmeyi sağlarlar. Olay ile ilgili sinir ağına örneklerden başka hiçbir ön bilginin verilmemiş olması önemlidir. Ağ kendisine tanımlanan örnekleri tekrar tekrar inceleyerek ağdaki ilişkiyi kavramaya çalışır. Her yeni örnek, ağın sahip olduğu bilgiye bir yenisini ekler ve bu işlem tekrar ettikçe ilgili problem hakkında genellemeler yapılır [Öztemel, 1996, s134-140].
- YSA kendisine tanımlanan bir şekli, daha önce öğrendikleri ile mukayese ederek aradaki benzerlikleri ortaya koyma ve eksik şekilleri tanımlama, benzer şekilleri oluşturma veya şekilleri belirli sınıflara ayırma özelliklerine sahiptir.
- Bir ağ öğrenme esnasında sahip olduğu bilgileri temsil etme şeklini kendisi belirleyebilir. Bu daha çok kodlanması zor veya mümkün olmayan olayların üzerindeki çalışmalarda önemlidir. Bu özellikleri neticesinde sinir ağları, kendilerine sunulan örneklerden genelleme yapabilirler. Benzeri olayları değerlendirmekte de bu genellemeden yararlanırlar. Eksik, gürültülü, doğruluğu belli olmayan olaylarda bu genelleme özelliği oldukça faydalıdır. Genelleme sonunda eksik bilgiler tanımlanabilir, gürültülü bilgiler süzülerek ayrıştırılabilir. Özellikle görüntü tanıma, sınıflandırma ve sinyal analizinde kullanılabilir. Özellikle görüntü tanıma, sınıflandırma ve sinyal analizinde kullanılabilir.
- Verilerde bir eksik söz konusu olursa geleneksel yöntemler çalışmazlar. İyi eğitilmiş, genelleme kapasitesi yüksek bir sinir ağı kendisine tanımlanan veriler eksik olsa bile karar verme işlemine devam edebilir. Aynı şekilde sinir ağı üzerinde bir takım problemler ve bozukluklar da olabilir. Geleneksel sistemlerin tersine sinir ağları bu durumda da çalışmalarına devam ederler. Verilerdeki eksiklik veya sinir ağlarındaki yapısal bozukluk arttıkça sinir ağının performansı yavaş yavaş azalmaya başlar. Fakat sistem, fonksiyonunu tamamen durdurmaz. Her durumda bir sonuç üretebilir. Bu özellik sinir ağının yapısından

kaynaklanmaktadır. Çünkü ağıın sahip olduđu bilgi, ađ üzerindeki hücrelerin birbirleri ile olan bađlantıları üzerine dađıtılmıřtır. Böyle bir durumda tek bađlantı ve onun üzerindeki bilgi tek başına hiçbir anlam ifade etmez. Ancak bir grup halinde veya tam olarak bađlantıların birlikte düşünülmesi sonucu anlamlı bilgi üretilmektedir.

## **2.7. Yapay Sinir Ağlarının Uygulamaları**

YSA uygulamaları ile genelde řu konularda karşılaşılmaktadır:

- Satıř Tahminleri,
- Endüstriyel Proses Kontrol,
- Müřteri Arařtırması,
- Veri Geçerliliđi,
- Risk Yönetimi.

### **Tıpta Yapay Sinir Ağları**

YSA ları tıpta “sıcak” bir arařtırma alanıdır ve önümüzdeki yıllarda biyomedikal sistemler için daha geniş bir uygulama alanı bulacađı düşünölmektedir. řu anda arařtırmalar, daha ziyade insan vücudunun parçalarının modellenmesinde ve çeřitli taramalarda rahatsızlıkların fark edilmesinde yoğunlaşmaktadır (örneğin, cardiogram, CAT tarama, ultrasonic tarama). Kalp ve damar hastalıklarının teřhisinde ve modellenmesinde, elektronik burun uygulamalarında yapay sinir ağları kullanılmaktadır.

### **İřletmelerde Yapay Sinir Ağları**

İřletme; muhasebe ve finans analizi gibi uzmanlık gerektiren pek çok alana bölünmüřtür. İřletmelerde kaynak elde etme ve çizelgeleme gibi iřletme amaçları için



sinir ađlarının kullanıldıđı pek ok konu vardır. İřletmeler ayrıca veri tabanı madenciliđinde sinir ađlarının kullanımı iin gl bir potansiyele de sahiptirler. Sinir ađlarının en ok kullanıldıđı alan optimizasyon ve izelgelemedir ve bunlar iin Hopfield-Tank ađı kullanılır. Ayrıca YSA; daha iyi retim planlama yapmak iin rn veya hizmetin talebinin nceden belirlenmesinde; makinelerin ve kimyasal reaktrlerin yardımcı kontrolnde (servo-control) kullanılır.

### **Banka İřlemlerinde Yapay Sinir Ađları**

Kredi sonularının nceden tahmin edilmesi (kredi skortlama), kt kredilerin kurtarılması iin nceden tahmin yapılması, yeni mřterilerin davranıřlarının nceden belirlenmesi, “iyi risklerin (good risks)” tanımlanması ve kredilerin sunulmasında kullanılır.

### **Finans İřlemlerinde Yapay Sinir Ađları**

cret paylarının nceden belirlenmesi, pozisyon almaya yardımcı olma, portfy ve deđer ynetimi gibi konularda kullanılır.

YSA ayrıca ařađıda sıralanan veri analiz uygulamalarında da kullanılır :

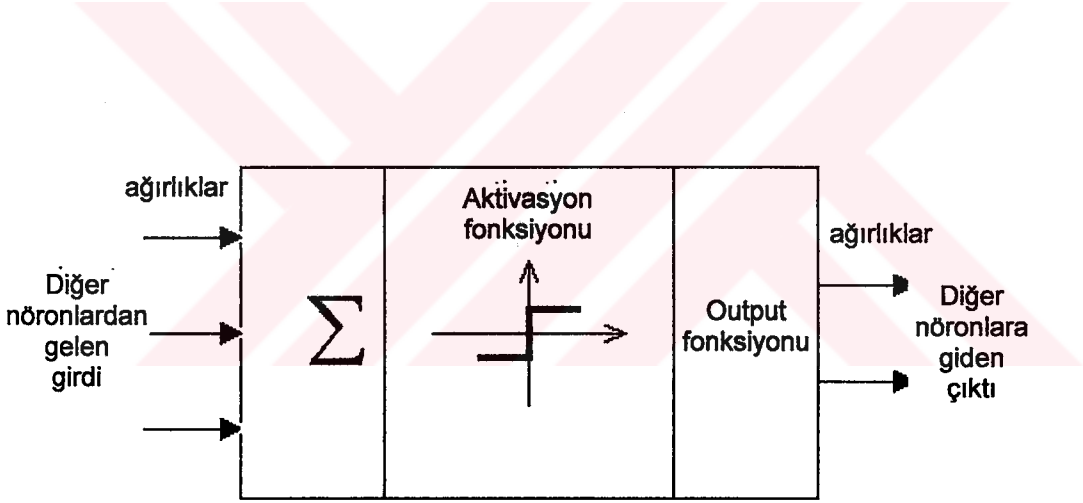
- Deđiřken Seimi: nemli/belirgin olmayan deđiřkenlerin tanımlanması ve elimine edilmesi,
- Tipik profiller,
- Mřterilerin sosyodemografik analizi,
- Veri tabanındaki veri kmeleri iin veri madenciliđi, vb.

### 3. YAPAY SİNİR AĞLARININ YAPISI

#### 3.1. Temel Bileşenleri

Pek çok farklı tip sinir ağı vardır, fakat genelde hemen hemen aynı bileşenlere sahiptirler. Tıpkı insan beyni gibi; bir sinir ağıda sinirlerden ve onlar arasındaki bağlantılardan oluşur. Sinirler; giren bilgiyi diğer sinirlere gidecek olan bilgiye çevirirler. Sinir ağındaki bu bağlantılar ağırlıklar olarak adlandırılır. “elektriksel” bilgi ağırlıklarda depolanan spesifik değerler ile simüle edilir. Bağlantı yapısının bu ağırlık değerlerinde basit değişiklikler yaparak da simüle edilebilir.

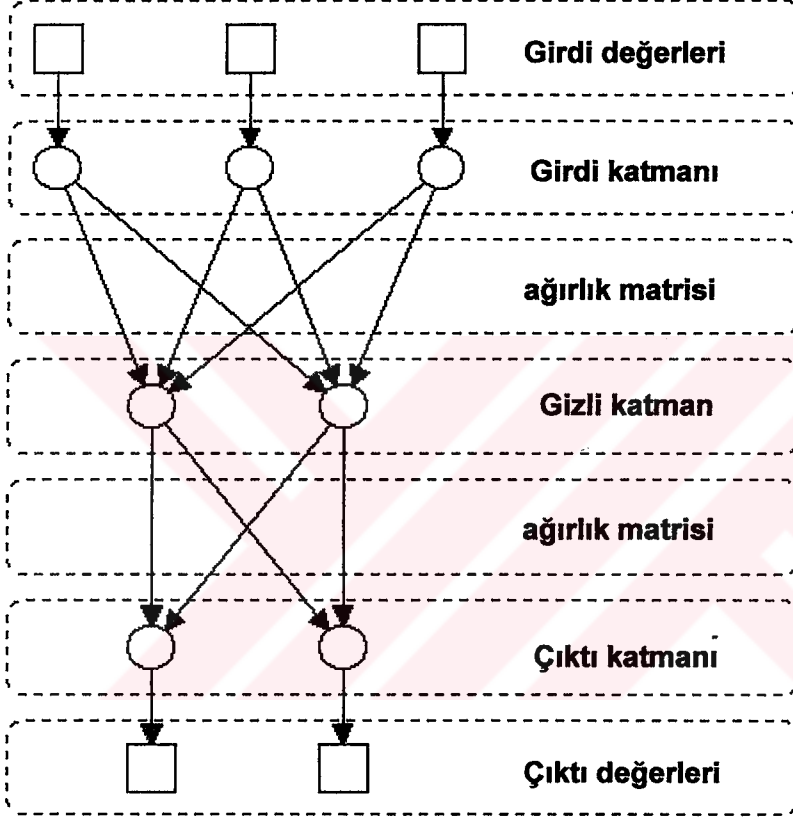
Şekil 3.1; bir sinir ağının idealize edilmiş bir sinirini gösterir.



Şekil 3.1. Bir Sinir Ağının İdealize Edilmiş Bir Siniri

Görüldüğü gibi bir yapay sinir biyolojik bir sinire benzer. Ve aynı şekilde çalışır. Girdi olarak adlandırılan bilgi ağırlıklandırılarak sinire gönderilir. Bu girdi tüm giren ağırlıkların değerlerini toplayarak bir yayılım fonksiyonu aracılığı ile işlenir. Sonuç değeri; sinirin aktivasyon fonksiyonu tarafından belirli bir eşik değeri ile karşılaştırılır. Eğer girdi eşik değerini aşarsa; sinir aktive edilecektir. Aksi takdirde engellenecektir (inhibit).

Bir sinir ađında, katmanlar gruplandırılır, sinir katmanları olarak adlandırılır. Genellikle bir katmanın her bir siniri sonraki ve önceki katmanın tüm sinirleri ile bağlantılıdır (ađın girdi ve çıktı katmanları hariç). Bir sinir ađına verilen bilgi; bir veya daha fazla ara katman aracılığı ile girdi katmanından çıktı katmanına doğru yayılım gösterir. Şekil 3.2. de üç katmanlı bir sinir ađı gösterilmektedir.



**Şekil 3.2.** Üç Katmanlı Bir Sinir Ađı Yapısı

Bu bir sinir ađının genel yapısı deđildir. Örneđin, bazı sinir ađı tipleri gizli/ara katman içermezler veya bir katmandaki sinirler bir matris şeklinde düzenlenir. Genel olan; tüm sinir ađı tiplerinde iki sinir ađı katmanını birbirine bađlayan en azından bir ađırlık matrisi olmasıdır.

Bir YSA modeli birbirinden bađımsız ve paralel olarak çalışabilen proses elemanlarının (yapay sinir hücrelerinin, nöronların) hiyerarşik bir şekilde

organizasyonundan oluşur. YSA nı oluşturan proses elemanlarından her biri beş temel parçadan oluşur [Öztemel, 1992: s14].

### 3.1.1. Girdi Elemanı

YSA da girdi, bir dış kaynaktan gelebileceği gibi diğer bir sinirden de gelebilir. Bir sinir hücresine birden fazla girdi gelebilir. Sinirsel hesaplama sadece sayıları işler. Eğer problem sayılardan oluşmuyorsa problemin girdileri sayısallaştırılır. 0 ve 1 arasında temsil edilir. Örneğin problem onaylama veya onaylamama diye ifade ediliyorsa; onaylama 1 olarak, onaylamama 0 olarak ifade edilir. Bunun gibi YSA da girdi karakterlerinin veya grafiklerinin piksel değeri de 0-1 arasında ağa verilir. Örneğin 5\*10 piksellik bir karakter 50 bit vektör girişi ile ağa girdi olarak verilebilir.

Giriş veri gruplarının ağa sunulduğu terminallerdir. Bu katmanda sinir hücresi sayısı, giriş veri sayısı kadardır ve her bir giriş sinir hücresi bir veri alır. Burada veri işlenmeden bir sonraki katman olan gizli/ara katmana geçer.

Ara / Gizli Katman : Ağın temel işlevini gören katmandır. Bazı uygulamalarda ağda birden fazla gizli katman bulunabilir. Gizli katman sayısı ve katmandaki sinir hücresi sayısı, probleme göre değişir, tamamen ağ tasarımcısının kontrolündedir ve onun tecrübesine bağlıdır. Bu katman;giriş katmanından aldığı ağırlıklandırılmış veriyi probleme uygun bir fonksiyonla işleyerek bir sonraki katmana iletir. Bu katmanda gereğinden az sinir hücresi kullanılması giriş verilerine göre daha az hassas çıkış elde edilmesine sebep olur. Aynı şekilde gerektiğinden daha çok sayıda sinir hücresi kullanılması durumunda da aynı ağda yeni tip veri gruplarının işlenmesinde zorluklar ortaya çıkar.

### 3.1.2. Ağırlıklar

YSA nın en önemli unsurlarındandır. Sinir hücresinde girdilerin etkisini tespit eder. Ağırlıklar bir proses elemanının her bir girdisinin önemini gösterir. Sabit veya

değişken olabilirler. Ağırlık değerinin eksi (-) olması etkinin ters yönde olduğunu gösterir. Ağırlıklar bir anlamda YSA'nın hafızaları olarak ifade edilebilir.

Problemlerde ağırlıklarla oynanarak en uygun ağırlıklar bulunur. Uygun ağırlıkların bulunması problemin çözüldüğünü gösterir.

### **3.1.3. Toplama Fonksiyonu**

Sinir hücrelerine net girdiyi açıklar. Toplama fonksiyonu her bir ağırlığı girdi elemanı ile çarpıp, daha sonra bunları tek bir proses elemanı olacak şekilde toplar. Kısacası bir proses elemanından gelen bilgileri birleştirme işlevini yerine getirir. Toplama fonksiyonunda girdiler tek veya çok miktarda olabilirler. Bu yüzden tek bir terim halinde temsil edilme ihtiyacı duyarlar. Bunun için bazı fonksiyonlar kullanılır. En çok kullanılan toplama fonksiyonu tipleri maksimum, minimum, çoğunluk, çarpım, toplam ve kümülatif toplam fonksiyonlarıdır.

### **3.1.4. Aktivasyon Fonksiyonu**

Aktivasyon fonksiyonu toplama fonksiyonunun sonucunu alır, ilgili fonksiyon ile bunu işler, proses elemanının nihai çıktısını üretir. Burada elde edilen çıktılar diğer proses elemanlarına veya dış dünyaya iletilir. Aktivasyon fonksiyonları her bir proses elemanının ürettiği çıktının (0,1) aralığında olmasını sağlar. Bunun sebebi; yapılan işlemlerde aşırı değerlerin üretilmesinin önlenmesi ve belli sınırlar içinde çalışmayı gerçekleştirmektir. Aktivasyon ve toplama fonksiyonları problemin yapısına göre tercih edilir.

### **3.1.5. Çıktı Elemanı**

Çıktılar ağların sonuçlarıdır. Her bir proses elemanı sadece bir çıktı üretebilir. Ama birden fazla girdiye sahip olabilir.

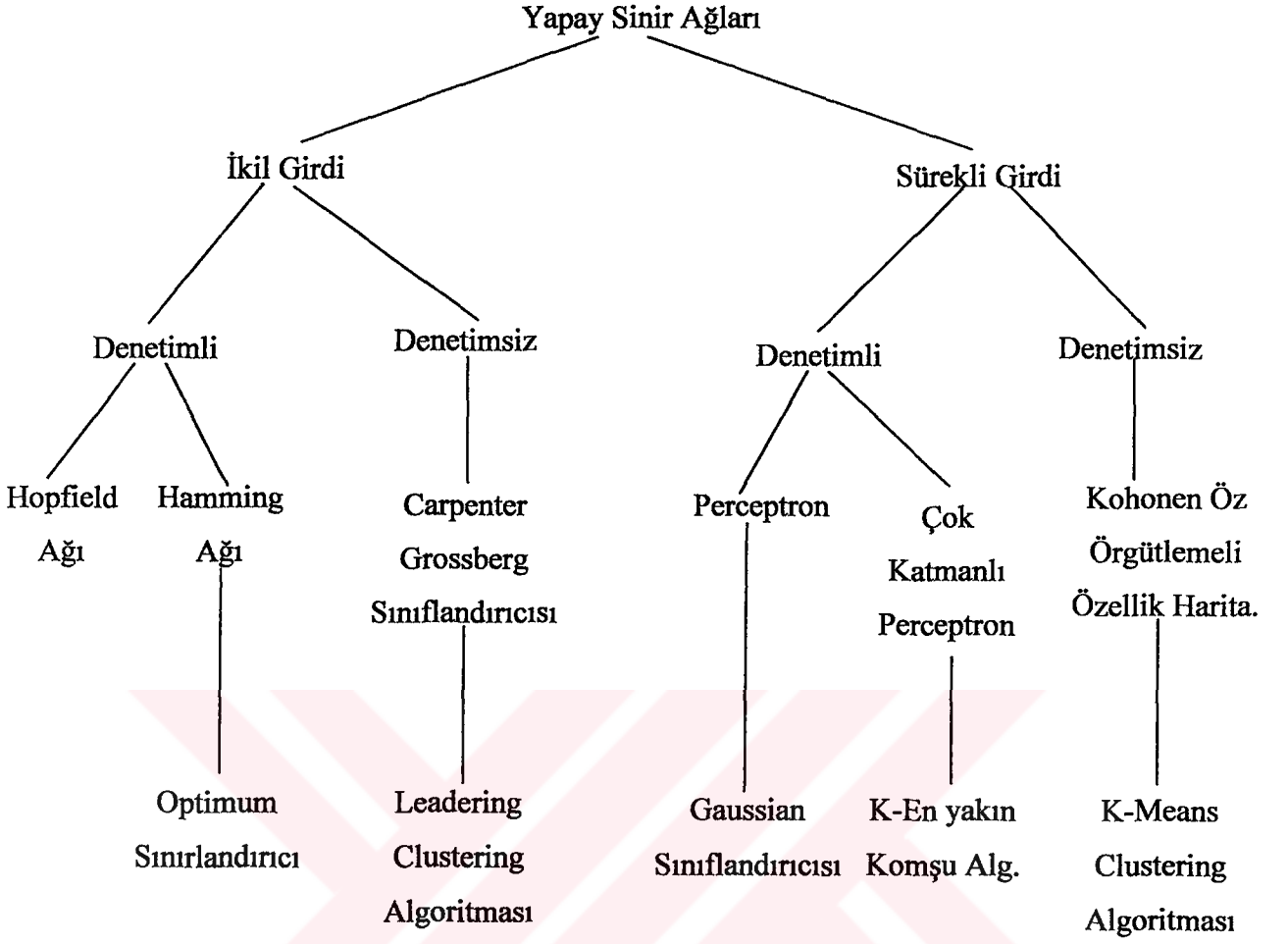
Çıktı katmanı; ağıın en uç katmanıdır.Gizli katmandan aldığı veriyi ağıın kullandığı fonksiyonla işleyerek çıktısını verir.Çıkış katmanındaki sinir hücresi sayısı , ağa sunulan her verinin çıkış sayısı kadardır.Bu katman dan elde edilen değerler yapay sinir ağıının söz konusu problemleri için çıkış değerleridir.

### 3.2. Yapay Sinir Ağlarının Tipleri

YSA nın çeşitli tipleri vardır. Bunlar; tipleri (ileri besleme veya geri besleme), yapıları ve kullandıkları öğrenme algoritmalarına göre sınıflara ayrılabilir. Sinir ağıının tipi; ağ katmanlarının birisindeki sinir hücrelerinin birbirleri ile bağlantılı olup olmadığına göre belirlenir. İleri besleme sinir ağları yalnızca iki farklı katman arasındaki sinirlerin bağlantılı olmasına izin verir. Geri beslemeli sinir ağları ise; ayrıca aynı katmandaki sinirler arasında da bağlantı olmasına izin verir.

Zamandan bağımsız olarak örüntülerin sınıflandırılmasında kullanılabilen altı önemli YSA nın aile ağacı şekil 3.3 de verilmektedir. Bu ağaç ilk olarak ikil veya sürekli girdi değerleri ile çalışanlar olmak üzere ikiye ayrılmaktadır. Bu ayırım kendi içerisinde eğitimin denetimli olması veya olmamasına göre tekrar ikiye ayrılmıştır.

Hopfield YSA ve Perceptron gibi denetimli eğitilen ağlar, çağrışımıcı bellek veya sınıflandırıcı olarak adlandırılırlar. Denetimli eğitimde her girdi vektörü için çıktının ne olacağı baştan sisteme verilmektedir. Eğitim sırasında verilen çıktı değerlerine göre ağ, fiili çıktıyı olması gereken çıktı ile karşılaştırır ve aradaki farkı gidermek için ağırlıklar matrisi elemanlarının değerlerini değiştirir. Ancak bazı araştırmacılar, bir biyolojik sistem içerisinde uygulanan girdilere karşı gelen çıktının baştan bilinemeyeceği için, bu şekildeki denetimli eğitimin biyolojik kurallara uygun olmayacağını savunmuş, kendi modellerini bilinen biyolojik kurallarla ters düşmeyecek şekilde kısıtlamışlardır.



**Şekil 3.3. Çeşitli YSA Modelleri**

Tablo 3.1 de yeni ağların geliştirilmesine temel teşkil eden ve/veya günümüzde halen çeşitli uygulama alanlarında etkin bir şekilde kullanılan YSA temel özellikleri ile birlikte verilmektedir.

**Tablo 3.1. Geliştirilen YSA Modelleri ve Özellikler**

YSA Adı	Geliştirenler	İlk Uygulamalar	Sınırlar	Yorumlar
<b>Adaptive Resonance Theory</b>	Gail Carpenter Stephen Grossberg	Şekil tanıma radar veya sonar çıktıların tanımlanması	Bozuk veya eksik verilere karşı duyarlı	Çok karmaşık
<b>Avalanche</b>	Stephen Grossberg	Sürekli söz tanıma, robot kollarına motor komutlarının	Hareket dizilerinin aynen tekrarlanması, hızı değiştirmek	YSA sınıfı, bütün bu işlevi yapan tek bir ağ bulunmamakta

		öğretiminde	veya yeni hareket eklemek için basit bir yöntem yok	
<b>Backpropagation (Geriye Yayımlama Ağı)</b>	Paul Werbos, David Parker, Davir Rumelhart	Yazılı metinden sözsentezi, robot kollarının kontrolü, çeşitli işletme uygulamaları	Denetimli eğitim, girdi ve çıktı örnekleri çok sayıda olmalı	Günümüzde en yaygın kullanılan YSA, öğrenimi kolay, sonuçları etkin
<b>Bidirectional</b>	Bart Kasko	İçeriği adreslenebilir bellek	Düşük saklama yoğunluğu, bilginin düzgün kodlanması gerektirir	Öğrenimi en kolay ağ, eğitim için uygun
<b>Boltzmann Makinesi Cauchy Makinesi</b>	Jeffrey Hinton, Terry Sejnowski, John Hopkins, Harold Szu	Sonar, radar ve görüntüler için şekil tanıma	Uzun eğitim süresi	Basit ağ modelleri
<b>Brain State in a box</b>	James Anderson	Veri bankalarından bilgi çıkarımı	Bir adımda karar verme	Parçalanmış girdileri tanıma
<b>Cerebellatron</b>	Andreas Pellionez, David Mar, James Albus	Robot kolları motor hareketlerinin kontrolü	Girdilerde karmaşık kontrollere gerek duymaktadır.	Avalanche ağına benzer. İstenen hareketleri elde etmek için, farklı ağırlıklar ile birçok komut birlikte kullanılabilir.
<b>Counter Propagation</b>	Robert Hecht-Nielsen	Görüntü sıkıştırma, istatistik analiz	Çok sayıda düğüm ve bağlantı gereklidir	Geriye yayılım ağının benzeri, daha basit, daha güçlüdür
<b>Hopfield</b>	John Hopfield	Bozuk verilerden bilgi çıkarımı, karakter tanıma	Öğrenme yok, ağırlıklar baştan belirlenir.	Geniş ölçüde tümleşik olarak gerçekleştirilebilir.
<b>Madaline</b>	Bernard Widrow	Birden fazla adaline düğümünden meydana gelir. Uzak telefon görüşmelerinde ekonun giderilmesi	Girdi ve çıktı değerleri arasında doğrusal ilişki varsayımına dayanır	Öğrenme kuralları oldukça güçlüdür
<b>Neocognitron</b>	Kunihiko Fukushima	El yazısı karakteri tanıma	Çok sayıda düğüm ve bağlantı gerekir	Geliştirilen en karmaşık ağ, Kanji karakterlerini tanıyabilir
<b>Perceptron</b>	Frank Rosenblatt	Karakter tanıma	Kanji gibi karmaşık karakterleri tanıyamaz	Bilinen en eski YSA, günümüzde kullanımı yok
<b>Self Organizing Maps</b>	Teuvo Kohonen		Çok fazla eğitime gerek duyar	Sayısal aerodinamik hesaplamalarda çoğu algoritmadan üstün

### 3.2.1. Perseptron

Perseptron ilk kez 1958 yılında F. Rosenblatt tarafından ortaya atıldı. Bu yalnızca başlangıç girdi ve çıktı değerlerini (0,1) kabul eden iki sinir katmanında oluşan çok basit bir sinir ağı tipidir. Öğrenme prosesi destekli öğrenmedir ve ağ AND ve OR gibi temel mantıksal operasyonlar ile çözülebilir. Ayrıca örnek sınıflandırma amaçları



(pattern classification purposes) için kullanılır. Daha karmaşık mantıksal operasyonlar (XOR problemi gibi) bir perseptron ile çözülemez.

Tablo 3.2. Perseptronun Özellikleri

Perseptron özellikleri	
Örnek yapı	
Tipi	İleri besleme
Sinir katmanları	1 girdi katmanı 1 çıktı katmanı
Girdi değer tipi	Başlangıç (binary)
Aktivasyon fonksiyonu	hard limiter
Öğrenme Metodu	Destekli
Öğrenme algoritması	Hebb Öğrenme Kuralı
Başlıca kullanım alanları	Basit mantıksal operasyonlar Örnek sınıflandırma (pattern classification)

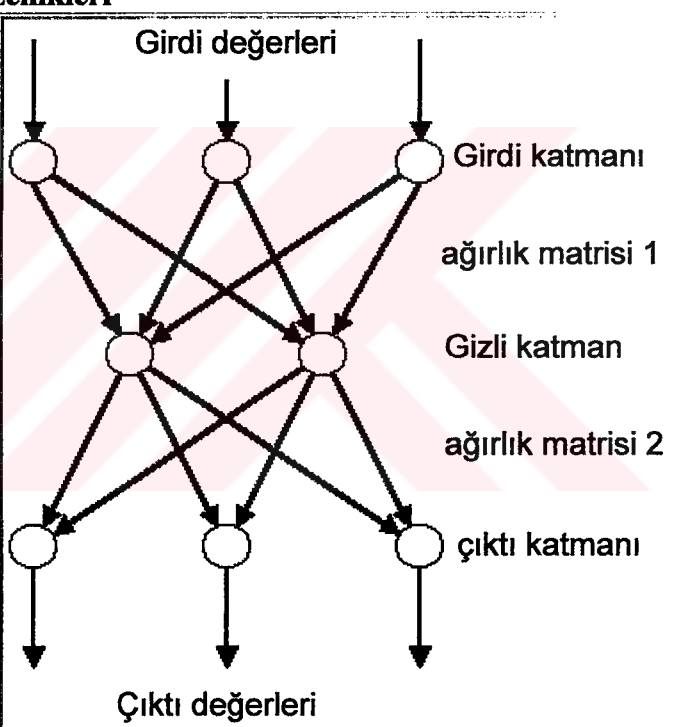
### 3.2.2. Çok Katmanlı Perseptron

Çok katmanlı perseptron ilk defa 1969 yılında M. Minsky ve S. Papert tarafından ortaya atıldı. Perseptron un genişletilmiş bir halidir. Girdi ve çıktı katmanları arasında

bir veya daha fazla ara sinir katmanı olabilir. Geniş bir yapı olan çok katmanlı bir perseptron; XOR problemini de kapsayan her türlü mantıksal operasyon çözebilir.

Çok katmanlı perseptron, ilk geliştirildiği yıllarda etkin bir eğitime algoritması olarak yoğun bir şekilde çok kullanılmıştır. En yeni eğitime algoritmalarından biri geri yayılım (backpropagation) dır. Çok katmanlı perseptron yapısındaki bir ağa bu algoritmanın uygulanması ile Geri Yayılım Yapay Sinir Ağı geliştirilmiştir.

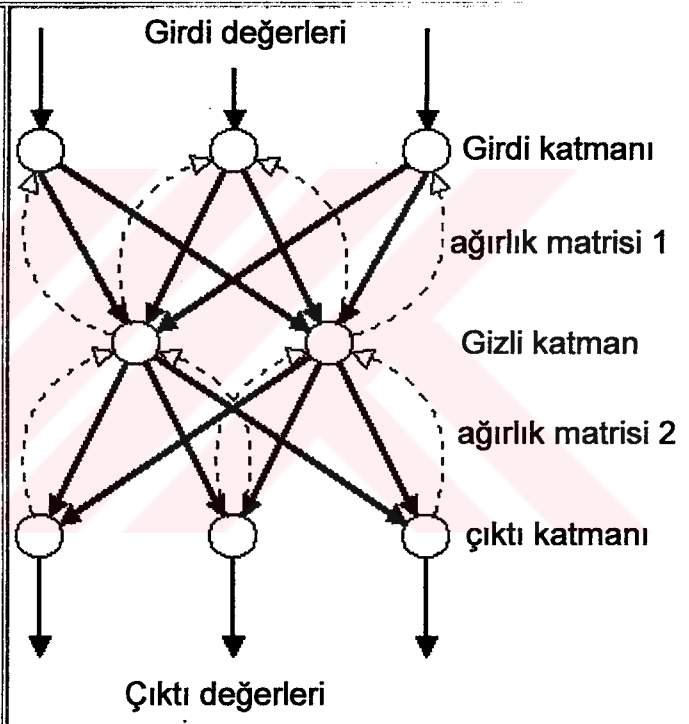
Tablo 3.3. Çok Katmanlı Perseptronun Özellikleri

Çok katmanlı perseptron özellikleri	
Örnek yapı	
Tip	İleri besleme
Sinir katmanları	1 girdi katmanı 1 veya daha çok ara/gizli katman 1 çıktı katmanı
Girdi değeri tipi	Başlangıç (Binary)
Aktivasyon fonksiyonu	Hard limiter / sigmoid
Öğrenme metodu	Destekli
Öğrenme algoritması	Delta Öğrenme Kuralı, Backpropagation
Başlıca kullanım alanları	Karmaşık mantıksal operasyonlar Örnek sınıflandırma (pattern classification)

### 3.2.3. Geriye Yayınım Ağı (Backpropagation Net)

Geriye yayınım ağı ilk kez 1986 yılında, E. Rumelhart ve R.J. Williams tarafında ortaya atıldı. Ve sinir ağları tiplerinin en güçlülerinden biri olarak kabul edilir. Çok katmanlı perseptron ile aynı yapıdadır ve geriye yayınım öğrenme algoritması için kullanılır .

Tablo 3.4. Geriye Yayınım Ağı Özellikleri

Geri Yayınım Ağı Özellikleri	
Örnek yapı	
Tip	İleri besleme
Sinir katmanları	1 girdi katmanı 1 veya daha çok ara/gizli katman 1 çıkı katmanı
Girdi değeri tipi	Başlangıç (Binary)
Aktivasyon fonksiyonu	Sigmoid
Öğrenme metodu	Destekli
Öğrenme algoritması	Backpropagation (genel olarak)
Başlıca kullanım alanları	Karmaşık mantıksal operasyonlar Örnek sınıflandırma (pattern classification) Konuşma analizi

### 3.2.4. Hopfield Ađı

Hopfield ađı ilk kez 1982 yılında, fizikçi J. J. Hopfield tarafından ortaya atıldı. “Termodinamik modeller” olarak adlandırılan sinir ađı tiplerine dahildir. Her bir siniri diđer tüm sinirlere bađlı bir sinirler kümesinden oluşur. Girdi ve çıktı sinirleri arasında farklılık yoktur. Bir Hopfield ađının temel uygulama alanı; örnekleri tanıma ve depolamadır (imaj dosyalama gibi).

Hopfield ađı, tek katmanlı ve geri beslemeli bir YSA dır [Lippmann, 1987: s4-22]. Genellikle giriş deđerlerinin ikili düzende (0 ve 1) temsil edilebileceđi uygulamalarda kullanılır. Örnek olarak; siyah beyaz görüntülerin işlenmesi, 8 bitlik ASCII karakterleri ile temsil edilen metinlerin işlenmesi verilebilir.

Sürekli Hopfield Ađı modeli, optimizasyon problemlerinin çözümünde başarılı uygulamalar verir. Örnek olarak Gezgin Satıcı Problemi (Traveling Salesman Problem) verilebilir. Hopfield modeli YSA nı pek çok optimizasyon problemine uygulamak mümkündür. Yapısında bir enerji fonksiyonunu minimize etme özelliđi bulunduğu için gerçek dünyadaki pek çok probleme uygunluk göstermektedir.

Hopfield modeli, yoğun bađlantılı, geribeslemeli ve bunun sonucu olarak da dinamik bir yapıya sahip olduđu için gerçek biyolojik sistemlere en çok uyan modellerden biridir. Hopfield modeli, gezgin satıcı gibi zor optimizasyon problemlerinin çözümünde bile kullanılacak bir YSA dır. Hopfield modelinde donanım tavlama yönteminin kullanılması ile global minimum bulunması sağlanabilir. Hopfield modelini çeşitli uygulamalarda kullanmada en önemli unsur, devre parametrelerinin belirlenmesidir.

Problemin enerji fonksiyonu olarak iyi ifadesi alınır ve parametreler uygun seçilirse devre problemi için geçerli ve iyi çözümler bulunur. Hopfield modelini herhangi bir probleme uygulamada kullanabilmek için problemin çözümü olacak şekilde devre

parametrelerini belirlenmesini sağlayacak bir koşul bulunması üzerinde çalışılması gerekmektedir [Hopfield & Tank, 1985: s141-152].

Tablo 3.5. Hopfield Ağının Özellikleri

Hopfield Ağı Özellikleri	
Örnek yapı	<p>The diagram shows a Hopfield neural network with 6 nodes arranged in a hexagonal pattern. Each node is connected to every other node, forming a complete graph. The central area is labeled 'ağırlık matrisi' (weight matrix). Four nodes are labeled 'Girdi değeri' (input value) with dashed arrows pointing to them.</p>
Tip	Geri besleme
Sinir katmanları	1 matrix
Girdi değeri tipi	Başlangıç (Binary)
Aktivasyon fonksiyonu	Signum / hard limiter
Öğrenme metodu	Desteksiz
Öğrenme algoritması	Delta öğrenme kuralı Simulated annealing (genel olarak)
Başlıca kullanım alanları	Örnek ilişkisi (pattern association) Optimizasyon problemleri

### 3.2.5. Kohonen Özellik Haritası

Kohonen özellik haritası ilk kez 1982 yılında Helsinki Üniversitesinden Prof. Teuvo Kohonen tarafından ortaya atıldı. Eğer insan beyninin öğrenme prosesi simüle edilecekse belki de en kullanışlı sinir ağı tipidir. Bu tip sinir ağının “kalbi” özellik

haritasıdır. Bir sinir katmanı, belirli girdi değerlerine uygun olarak düzenlenir. Bu sinir ağı tipi hem ileri besleme (özellik haritasının girdi katmanı) hem de geri besleme (özellik haritası) içindir. Bir Kohonen Özellik Haritası, sample applet için kullanılır.

Tablo 3.6. Kohonen Özellik Haritasının Özellikleri

<b>Kohonen Özellik Haritası Özellikleri</b>	
<b>Örnek Yapı</b>	
<b>Tip</b>	İleri besleme / Geri Besleme
<b>Sinir katmanları</b>	1 girdi katmanı 1 harita katmanı
<b>Girdi değeri tipi</b>	Başlangıç (binary), gerçek (real)
<b>Aktivasyon fonksiyonu</b>	sigmoid
<b>Öğrenme metodu</b>	Desteksiz
<b>Öğrenme algoritması</b>	Selforganization
<b>Başlıca kullanım alanları</b>	Örnek sınıflandırma (pattern classification) Optimizasyon problemleri, simülasyon

### **3.2.6. Hücresel Yapay Sinir Ağı Modeli**

1988 yılında Chua ve Yang Hücresel Yapay Sinir Ağı (HYSA) adı verilen yeni bir model önerdiler. Büyük çaplı tümlşik devre gerçeklemesine uygun olan bu ađ modeli; görüntü işleme, model tanıma, robot görmesi ve hareketli cisimlerin saptanması gibi birçok uygulamada başarılı sonuç vermiştir.

HYSA nın temel yapı taşına hücre denir. Hücreler lineer olan ve lineer olmayan devre elemanlarından oluşmaktadır. Her bir hücre sadece yakın komşuluğunda bulunan hücelere bađlıdır. Komşu olan hüceler birbirlerini doğrudan etkilerken; birbirine bađlı olmayan hüceler yani komşu olmayan hüceler başlangıç durumundan kararlı duruma giderken zamanda yayılım ile birbirlerini etkilemektedirler. HYSA, hücelerinin iki boyutlu bir ızgara oluşturacak şekilde birbirlerine bađlanmasından oluşmuş tek katmanlı bir ađdır [Chua & Yang, 1988: s1273-1290].

## **3.3. Öğrenme**

### **3.3.1. Öğrenmenin Genel Yapısı**

Bilgi işleme hızı bilgisayar teknolojisinde halen önemli bir etkindir. Sistemlerin her geçen gün biraz daha karmaşık olması nedeni ile daha çok bilgiyi daha verimli bir şekilde işleme gerekliliđi yeni yazılım/donanım sistemlerini zorunlu hale getirmiştir. Halbuki insan beyni, oldukça fazla bilgiyi gerçek zamanlı olarak oldukça hızlı bir biçimde işleyebilmektedir. Bu durum, yapısındaki hücelerin paralel olarak çalışması ile açıklanmaktadır. YSA da yine birbirlerine bađlı ve paralel işlem elemanlarından oluştuğundan, hızlı işleyebilme yetenekleri, bu ađlara özellikle endüstride gerçek zamanlı çalışma kabiliyeti de kazandırır.

Simon (1983) tarafından öğrenme işlevi, bir sistemin bir görevi daha etkili ve verimli yapabilme yeteneğini kazanması şeklinde tanımlanmıştır. Michalski (1986) ise, öğrenmenin deneyimin bir göstergesi olduğunu belirtmiştir [Baklavacı, 1994].

Genel olarak öğrenme, bir bilginin çözümü demektir. Bir istem aldığı bilgiyi kendi bünyesinde çözerek öğrenmektedir. Bunu da öğrenme süreci boyunca yapısını değiştirerek yerine getirmektedir.

Sonuçta öğrenme, sistemin parametrelerini değiştirerek verilen giriş değerlerine karşılık istenen çıkış değerlerinin alınması veya üretilmesini sağlayacak bir kendini uyarılma sürecidir [Karamahmut, 1994: s17].

YSA da değişebilen sistem parametreleri, hücreler arası bağlantıları sağlayan sinapsları temsil eden bağlantıların ağırlık katsayılarıdır. En uygun ağırlıkların bulunması problemin çözüldüğü anlamına gelir. Öğrenme sırasında oluşabilecek değişimler şu şekilde sıralanabilir [Güzeliş, 1991]:

- Yeni bağlantıların oluşması,
- Varolan bağlantıların kaybolması,
- Varolan bağlantıların ağırlıklarının değişimi.

YSA da öğrenme olayı aşama aşama gerçekleştirilen bir işlem, genelde bir süreçtir. Öğrenme sırasında bir eğitim seti kullanılır. Eğitim kümesinde, ilgili olaylar arasındaki ilişkileri YSA na en uygun şekilde gösterebilecek girdiler bulunur.

Öğrenme özelliği YSA nın en önemli özelliklerindedir. Kendisine verilen öğrenme seti sayesinde öğrenme işlemi gerçekleştirildiği zaman YSA, iç dinamikleri ile öğrenme setinde belirlediği ilişkilerden yola çıkarak genelleme yapar ve yeni problemler karşısında sonuçlar üretir. Öğrenme aşamasında her adımda eğitim setinden sırayla veya rast gele örnekler alınmakta ve ağa uygulanmaktadır. Adım sonunda belirli bir amaç ölçütü gereğince ağın bağlantı ağırlık katsayıları değiştirilmektedir.



Adım adım eğitim dışında, kümesel eğitim denilen öğrenme kuralları da vardır. Bunlarda eğitim tek bir adımda gerçekleşmektedir.

Bağlantı ağırlık katsayılarının son değerleri eğitim kümesindeki tüm örnekler göz önüne alınarak tek bir adımda bulunmaktadır.

### 3.3.2. Öğrenme Türleri

YSA öğrenme durumlarına göre genel olarak üç türlü öğrenme algoritmalarına sahiptirler[Öztemel, 1996, s134-140]:

a) **Öğretmenli Öğrenme:** Bu öğrenme türünde, dışarıdan bir öğretmenin sinir ağının öğrenmesine müdahalesi söz konusudur. Öğretmen, sinir ağının ilgili girdi için öğrenmesi gereken sonucu sinir ağı sistemine tanımlar. Diğer bir ifade ile; ağa, girdi-çıkı ikilisinden oluşan örnekler sunulur ve girdi çıkı bilgisinin ağa tanımlanması gereklidir. Ağ girdi kısmını alır ve o anki bağlantı ağırlıklarının tanımladığı bilgi ile bir çıkı oluşturulur. Bu çıkı, hedef çıkı ile karşılaştırılır ve ağıdaki hata tekrar ağa aktarılarak ağırlıklar bu hatayı azaltacak biçimde değiştirilir.

Öğretmenli öğrenme basit anlamda öğrenciler ile öğretmenin bir sınıftaki dersine benzetilebilir. Öğrenciler ile öğretmen ders boyunca karşılıklı etkileşim içindedirler. Öğrencilerin sorulan sorulara verdikleri cevaplar öğretmen tarafından denetlenmektedir ve gerekirse doğru cevap öğretmen tarafından öğrencilere iletilmektedir.

b) **Takviyeli Öğrenme:** Bu tür öğrenmede, yine bir öğretmene ihtiyaç vardır. Öğretmenli öğrenmeden farkı ise bu durumda öğretmenin ağı üretmesi gereken sonuç yerine, onun ürettiği sonucun sadece doğru veya yanlış olduğunu söylemesidir. Bu ise, ağa bir takviye sinyalinin gönderilmesi ile

sağlanır. Bu tür öğrenme, örnek için beklenen çıktının oluşturulamadığı durumlarda çok faydalıdır.

- c) **Öğretmensiz Öğrenme:** Bu durumda hiçbir öğretmene ihtiyaç yoktur. Onun için buna çoğu zaman kendi kendine organize öğrenme (self-organized learning) de denilmektedir. Ağ kendisine gösterilen örnekleri alır ve belli bir kritere göre sınıflandırır. Bu kriter önceden bilinmeyebilir. Ağ kendi öğrenme kriterlerini kendisi tanımlamaktadır.

### 3.3.3. YSA da Kullanılan Önemli Öğrenme Algoritmaları

Burada öğretmenli ve öğretmensiz öğrenmeye yönelik yedi önemli öğrenme algoritması verilecektir [Zurada, 1991]:

- a) **Hebb Öğrenme Algoritması:** Hebb'in öğrenme algoritmasında temel fikir, hücrenin darbe üretmesine sebep olan bağlantıların ağırlık katsayılarını büyüterek etkilerini arttırmaktır. Dışarıdan istenilen bir çıkış uygulanmadığından öğretmensiz bir öğrenme algoritmasıdır.
- b) **Algılayıcı (Perceptron) Öğrenme Algoritması:** Rosenblatt (1958) tarafından önerilen Algılayıcı Öğrenme Algoritması'nda ağırlık ( $w$ ) değişimi; gerçekleşen çıktı ile istenen çıktı arasındaki fark ile orantılıdır. Dolayısı ile öğretmenli bir algoritmadır. Devrenin çıkışı iki kutuplu ve  $w$ ' nun oluşturduğu bir hiper yüzey tarafından belirlenmektedir. Ağın ürettiği çıkış istenen çıktıdan farklı ise, hiper yüzey hatayı azaltacak yönde hareket etmektedir. Hata sıfır olduğunda  $w$  değişimi durmaktadır.
- c) **Eğim-Düşme Öğrenme Algoritması:** Eğim-düşme (Gradient Descent) Öğrenme Algoritması veya "Delta Kuralı" Öğrenme Algoritması sadece üretilebilir fonksiyonlara sahip olan ağlara uygulanabilmektedir. Öğretmeni bir öğrenme algoritmasıdır.  $w$  değişimi hatanın eğiminin ters yönündedir. Böylece

hata fonksiyonunun oluşturduğu çanağın dibine, yani hatanın minimum olduğu yere doğru hareket edilmektedir.

- d) **Widrow-Hof Öğrenme Algoritması:** Widrow-Hof Öğrenme Algoritması öğretmenli bir öğrenme algoritmasıdır.  $w$  değişimi ağıın çıkışından bağımsızdır. Aslında bu algoritma Eğitim-Düşme algoritmasının özel bir halidir.
- e) **İlinti Öğrenme Algoritması:** Hebb Öğrenme Algoritmasının Öğretmenli Uyarlamasıdır.
- f) **Kazanan Her Şeyi Alır Öğrenme Algoritması:** Yarışma türü öğrenme algoritmalarına örnek teşkil etmekte olan bu algoritmada genel kural; giriş değerine en yakın olan bağlantı ağırlık katsayılarını bulmaktır. Bu  $w'$  lara ilişkin hücre, kazanan hücre olarak adlandırılmaktadır. Sadece kazanan hücre bir çıkış üretmekte ve bu hücreye ilişkin  $w'$  lar değişime uğramaktadır. Öğretmensiz öğrenmenin tipik bir örneği olan bu algoritma sonuçta ağa gelen girişleri sınıflandırmaktadır.
- g) **Outstar Öğrenme Algoritması :** Öğretmenli öğrenme algoritma türü olan bu öğrenme algoritmasında amaç;  $w'$  ları istenen çıktıya benzetmektir.

### 3.4. Temel Yapay Sinir Ağı Algoritmaları

#### 3.4.1. İleri Yayınım

İleri yayınım, destekli bir öğrenme algoritmasıdır. Ve bir sinir ağı aracılığı ile girdi katmanından çıktı katmanına doğru “bilgi akışını” tanımlar.

Algoritma şu şekilde çalışır:

1. Tüm ağırlıkları  $-1.0$  dan  $+1.0$  a doğru rastgele olarak ata,
2. Ağın girdi katmanının sınırları için bir girdi örneği (başlangıç değeri) ata,
3. Sonraki katmanın herbir sinirini aktive et:
  - Önceki sinirin çıktı değerleri ile bu sinire yol gösteren bağlantıların çoklu ağırlıklarını hesapla,
  - Bu değerleri topla,
  - Bu sinir hücresinin çıktı değerini hesaplayan bir aktivasyon fonksiyonu için sonucu onayla.
4. Çıktı katmanına ulaşıncaya kadar bu adımları tekrar et,
5. Hesaplanan çıktı ile hedef çıktıyı karşılaştır ve bir hata değeri hesapla,
6. Eski ağırlık değerlerine bu hata değerini ekleyerek tüm ağırlıkları değiştir,
7. 2. adıma git,
8. Eğer çıktı değeri, hedeflenen çıktıya uyuyorsa algoritmayı bitir.

### 3.4.2. Geri Yayınım

Geri yayılım bir destekli öğrenme algoritmasıdır. Ve genelde ağın ara sinir katmanlarını bağlayan ağırlıkları değiştirmek için çok katmanlı perseptronlar tarafından kullanılır. Geriye yayılım algoritması; geriye doğru yönlendirmede ağırlıkları değiştirmek için, hesaplanmış bir çıktı hatasını kullanır. İleri yönlendirmeli yayılım yapılırken; sinir hücreleri *sigmoid aktivasyon fonksiyonu* kullanılarak aktive edilir. Sigmoid fonksiyonunun formülü şu şekildedir:

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

Algoritma şu şekilde çalışır :

- 1) Bir girdi örneği için ileri yayılım aşamasını gerçekleştir ve çıktı hatasını hesapla.
- 2) Aşağıdaki formülü kullanarak her bir ağırlık matrisi için tüm ağırlık değerlerini değiştir.
- 3) 1. adıma git.

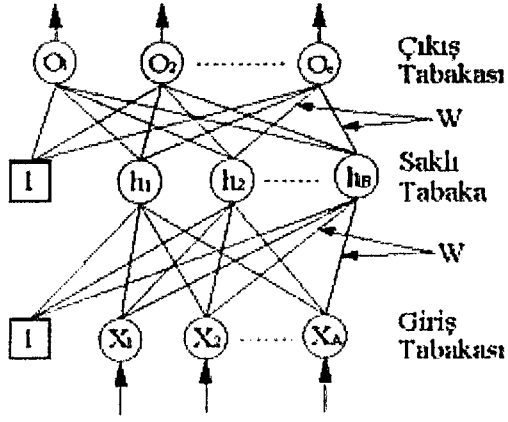
Ağırlık (eski) + öğrenme oranı \* çıktı hatası \* çıktı (i.sinir hücresinin) \* çıktı ((i+1). Sinir hücresinin) \* (1-çıkıtı) ((i+1). Sinir hücresinin)

- 4) Eğer tüm çıktı örnekleri hedef örneklerle uyuyorsa algoritmayı bitir.

Geri Yayılım Yapay Sinir Ağı; karmaşık modellerin teşhisinde başarı ile kullanılmaktadır. Karmaşık bir modeli teşhis edebilmek için modeldeki tüm noktaların paralel olarak incelenmesi gerekmektedir. Böyle bir incelemeyi yapabilecek sistem kolayca programlanamaz. Eğer sisteme birden fazla örnek model tanıtılıp bunlar arasındaki ilişkiyi öğrenmek için kendini uyarlayabilmesi sağlanırsa aynı ilişkiyi yeni modellere uygulayacak olan sistem bu zor problemi çözümler. İşte geriye yayılım yapay sinir ağı böyle bir sistemdir. Geriye yayılım algoritmalarında gizli/ara katman sayısının artırılması öğrenme oranını genelde yükseltir.

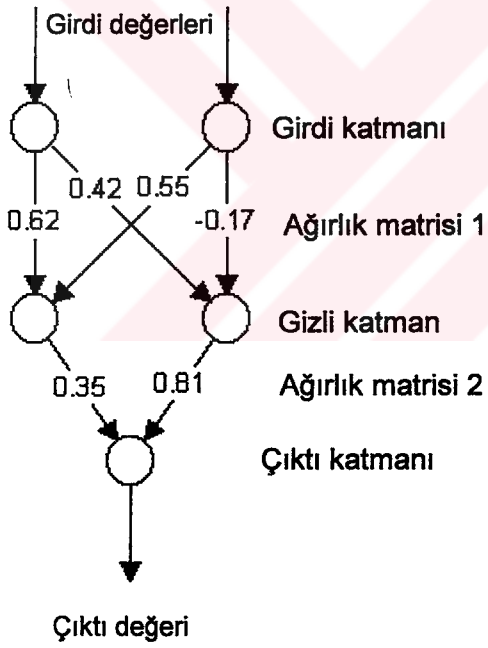
Geri yayılım ağında; bir katmandan bir başka katmana, aradaki katmanı atlayarak geçebilmek mümkün değildir.

Geriye yayılım algoritmasında YSA'nın eğitilmesindeki amaç; giriş değerleri ile, ağıın üretebileceği cevabın istenilen çıktı değerinin aynısını veya istenen hassasiyetteki yakın değerini sağlayacak bir ağırlıklar kümesi bulmaktır. İstenilen ağırlıkların bulunması problemin çözümü anlamına gelir. Şekil 3.4. de bir geri yayılım ağıının yapısı verilmektedir.



Şekil 3.4. Bir Geri Yayımlı Ağın yapısı

Bir geri yayımlı örneği vermek gerekirse; 3 katmanlı bir çok katlı perseptron olduğunu farzedelim.



Öğrenme örnekleri:

Girdi	Hedef
0 1	0
1 1	1

Başlangıçta, ağırlık değerleri 1. ağırlık matrisi için: 0.62, 0.42, 0.55, -0.17 olarak; 2. ağırlık matrisi için: 0.35, 0.81 olarak rast gele bir şekilde atansın. Ağırlık öğrenme oranı 0.25 olsun. Sonra birinci örnek girdisi (0 1) girdi katmanının sinir hücrelerine verilsin (girdi katmanının çıktısı girdinin aynısıdır).

Ara katmandaki sinir hücreleri aktivite edilir:

1. ara katmanın girdisi	$0*0.62 + 1*0.55 = 0.55$
2. ara katmanın girdisi	$0*0.42 + 1*(-0.17) = -0.17$
1. ara katmanın çıktısı	$1 / (1+\exp(-0.55)) = 0.634$
2. ara katmanın çıktısı	$1 / (1+\exp(+0.17)) = 0.457$

Çıktı katmanındaki sinir hücreleri aktivite edilir:

Çıktı sinir hücresinin girdisi	$0.634135591*0.35 + 0.4576020$
Çıktı sinir hücresinin çıktısı	$1 / (1+\exp(-0.592605124))$
Hedeflenen çıktıya göre hata değeri	$0 - 0.643962658 = -0.643962658$

Şimdi çıktı hatasını geri yayılım ile ağırlık matrisindeki ağırlıkların değişimi ile başlayalım:

1. ağırlığın değişim değeri	$0.25 * (-0.643962658) * 0.634135591 * 0.643962658 * (1-0.643962658) = -0.023406638$
2. ağırlığın değişim değeri	$0.25 * (-0.643962658) * 0.457602059 * 0.643962658 * (1-0.643962658) = -0.016890593$
1. ağırlığın yeni değeri	$0.35 + (-0.023406638) = 0.326593362$
2. ağırlığın yeni değeri	$0.81 + (-0.016890593) = 0.793109407$

1. ağırlık matrisindeki ağırlıkları değiştirelim:

1. ağırlığın değişim değeri	$0.25 * (-0.643962658) * 0 * 0.634135591 * (1-0.634135591) = 0$
2. ağırlığın değişim değeri	$0.25 * (-0.643962658) * 0 * 0.457602059 * (1-0.457602059) = 0$
3. ağırlığın değişim değeri	$0.25 * (-0.643962658) * 1 * 0.634135591 * (1-0.634135591) = -0.037351064$
4. ağırlığın değişim değeri	$0.25 * (-0.643962658) * 1 * 0.457602059 * (1-0.457602059) = -0.039958271$
1. ağırlığın yeni değeri	$0.62 + 0 = 0.62$ (değişmedi)
2. ağırlığın yeni değeri	$0.42 + 0 = 0.42$ (değişmedi)
3. ağırlığın yeni değeri	$0.55 + (-0.037351064) = 0.512648936$
4. ağırlığın yeni değeri	$-0.17 + (-0.039958271) = -0.209958271$

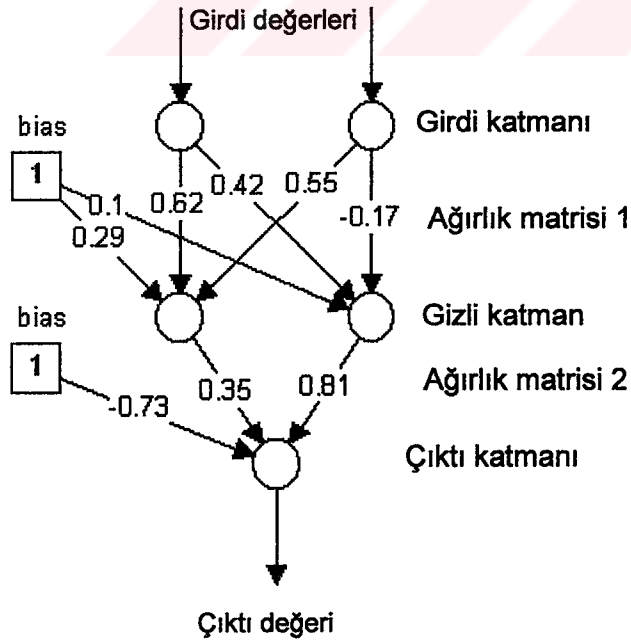
Birinci girdi örneği ağı yayıldıktan sonra aynı prosedürü sonraki girdi örneği içinde yapalım, fakat değiştirilmiş ağırlık değerleri ile. İkinci örneğin ileri ve geri doğru yayılımından sonra öğrenme adımı tamamlanmış olur ve ağı hatası; her bir örneğin çıktı hatalarının karesi eklenerek hesaplanabilir. Bu prosedürü tekrar çalıştırarak, bu hata değerini küçültebiliriz.

Eğer ağı hata değeri 0 veya kabul edilebilir bir değere ise algoritma başarı ile bitirilmiş olur.

Bu algoritma birden fazla ara katmanı olan Çok Katmanlı Perseptronlar için uygulanabilir.

Eğer girdi katmanındaki tüm değerler sıfır ise bu örnek için 1. ağırlık matrisindeki değerler asla değişmeyecektir. Bunun düzeltmek için "bias (eşik)" adı verilen ve sabit 1 çıktısı olan bir değere kullanılır.

Bu, ağı yapısını aşağıdaki şekilde değiştirir:



Şekil 3.5. Bias Değerinin Kullanımı



Bu eklenmiş ağırlıklar, ara katmanının ve çıktı katmanının sinir hücreleri başlangıçta rast gele değerler alırlar ve diğer ağırlıklarla aynı şekilde değiştirilirler.sonraki sinir hücresine sabit 1 çıktısı gönderilerek, tüm sinir hücrelerinin girdi değerlerinin daima sıfırdan farklı olması garanti edilir.



## 4. MALİYET/ÜRETİM MİKTARI İLİŞKİSİNİN BELİRLENMESİ

### 4.1. TOPRAK Sağlık Ürünleri Fabrikasının Tanıtımı

TOPRAK Holding'in Sakarya 1. Organize Sanayi Bölgesindeki kampüsünde bulunan üç fabrikasından biri olan TOPRAK Sağlık Ürünleri Müessesesinin temeli, Ocak 1995'te atılmıştır. Toplam 22.500 m<sup>2</sup> kapalı alan üzerine kurulmuştur. Bebek Bezi (Baby Diaper) ve Hijyenik Kadın Pedi (Sanitary Lady Napkin) üretimi yapan tesisin kurulması bir sene gibi kısa bir zamanda tamamlanmış ve ilk deneme üretimini Ocak 1996'da yapmıştır.

TOPRAK Sağlık Ürünleri Fabrikası, sektöründe dünyaca tanınmış, merkezi İsveç'te olan SCA Mölnlycke firmasının lisansı altında, Libero markasıyla Bebek Bezi ve Libresse markasıyla Hijyenik Kadın Pedi üretmektedir.

İlk kurulduğunda biri Bebek Bezi ve biri Hijyenik Kadın Pedi makinesi olmak üzere toplam 2 makine ile üretime başlayan fabrika, zamanla makine sayısını yükseltmiştir. Bugün fabrikada 4'ü Bebek Bezi ve 3'ü Hijyenik Kadın Pedi makinesi olmak üzere 7 adet makine çalışmaktadır. Piyasada standart olarak bulunan formatlarda (Mini, Midi, Maxi ve Maxi Plus) Bebek Bezi; 4 çeşit formatta da (Ultra İnce Normal, Ultra İnce Uzun, Reform Normal ve Reform Uzun) Hijyenik Kadın Pedi üretmektedir.

Ürünün çıkarılabilmesi için gerekli operasyonların tamamına yakını makineler tarafından yapılmakta (hijyenik), insan müdahalesi sadece, hammaddeler makineler üzerindeki yerlerine takılırken veya üretim ayarı, bakım, onarım... vb. durumlarda gerekmektedir.

Bebek Bezi Makinelerinin maksimum hızları 450 ped/dakika; Kadın Pedi Makinelerinin maksimum hızları ise 600 ped/dakikadır. Yani fabrika tam kapasiteyle çalışırsa, yaklaşık günde 2.400.000 adet Bebek Bezi ve yine 2.400.000 adet Hijyenik

Kadın Pedi üretebilme kapasitesine sahiptir. Bu da ayda yaklaşık 60.000.000 Çocuk Bezi ve 60.000.000 Hijyenik Kadın Pedi üretimi demektir.

Fabrikada şu anda ISO-9000 Kalite Yönetim Sistemine geçmek için çalışmalar devam etmektedir.

Çalışan personel sayısı bugün itibariyle toplam 158 kişidir. Bunlardan 124'ü sendikaya bağlı işçiler (Holdingde kapsam içi olarak anılmakta), 34'ü ise memurlardır (Holdingde kapsam dışı olarak anılmakta).

#### **4.2. Maliyet/Üretim Miktarı İlişkisinin Belirlenmesi**

Bu çalışmada; fabrikanın mevcut maliyet bilgileri Yapay Sinir Ağına girdi olarak verilmiş ve ağdan çıktı olarak üretim miktarını vermesi istenmiştir. Bu bölümde Toprak Sağlık Ürünleri Fabrikasından alınan maliyet bilgileri kullanılarak, bunlarla üretim miktarını belirleyecek olan bir Yapay Sinir Ağı Uygulamasından bahsedilecektir.

Bilindiği gibi üretim yapan bütün kuruluşlarda maliyetten söz edilir ve önemli bir kavramdır. İşletmeler için maliyet ve üretim miktarı birbirleriyle doğrudan ilişkilidir; karlılık ve dolayısıyla pazarda rekabet için üzerinde durulması gereken kavramlardır. Doğru maliyet hesabı fiyat stratejisini belirlemede çok önemli bir araçtır.

Maliyetle üretim miktarı arasındaki ilişkiyi etkileyen birçok faktör vardır ve bunu herhangi bir fonksiyonla ifade edebilmek çok güçtür. Üretim için gerekli girdiler kabaca bellidir, fakat bu girdilerle elde edilebilecek maksimum çıktı için birçok parametreden söz edilebilir. Örneğin, makinalardaki fireler, verimlilik, kapasite kullanım oranı, tamir/bakım...vb. Fireler düştükçe, verimlilik ve kapasite kullanım oranı arttıkça aynı maliyetle daha fazla üretim miktarı elde edilebilir ve birim ürün maliyeti düşürülebilir.

Bakım faaliyetleri bu parametrelerle direkt ilişkilidir. Girdilerden biri de bakım maliyetleridir ve maliyet/üretim miktarı ilişkisinin en karmaşık kısımlarından biri bakım maliyeti ile üretim miktarı arasındadır. Bakım maliyetinin artırılması belli bir seviyeye kadar verimliliği artırır (fire düşer, kapasite kullanım oranı artar, işçilik maliyeti düşer). Verimliliği maksimize eden bir optimum bakım maliyeti vardır. Bu seviyeden sonra yapılan bakım, verimliliğe etki etmez ve ekstra maliyettir. Aynı zamanda şu an yapılan bakımdan daha sonra sonuç alınabilir. Bu ilişkinin bir fonksiyonla ifade edilebilmesi çok zordur. Bu sebeple bu alanda bir Yapay Sinir Ağı uygulaması yapılması uygun görülmüştür.

#### **4.2.1. Uygulamada Kullanılan Program (Neurak v1.1)**

Uygulamada *Neurak v1.1* (Internet'te [www.quaternions.net](http://www.quaternions.net) adresinden temin edilebilir) programı kullanılmıştır.

*Neurak v1.1*; *Neurak* ve *Neurak Sim* isminde iki programdan oluşmakta ve Windows 98, 2000, XP işletim sistemleri üzerinde çalıştırılabilmektedir. *Neurak* programı çalıştırılarak ağ oluşturulur, *Neurak Sim* çalıştırılıp *Neurak*'da oluşturulan ağ bu programda çağrılarak kullanılır.

#### **4.2.2. Verilerin Normalizasyonu, Ağın Eğitilmesi ve Testi**

Kullanacağımız ağ eğitmek ve test etmek için Toprak Sağlık Ürünleri Fabrikasından geçmişte gerçekleşmiş 62 aya ait veri grubu kullanılmıştır. Ağın girdileri “Hammadde, Ambalaj, İşçilik, Enerji, Personel, Bakım, Dışarıdan Sağlanan Fayda ve Hizmetler, Amortisman, Sarf Malzeme ve Diğer “ olarak 1/1.000.000 oranında kısaltılarak alınmış ve çıktı ise “Üretim Miktarı” olarak belirlenmiştir.

Kullanılan Veri grubu Ek A'da verilmiştir. Bu değerlerin ağı girilebilmesi için normalize edilmesi (0-1 aralığına çekilmesi) gerekmektedir.

Normalizasyon Input veri gruplarına ve Output veri grubuna uygulanmıştır. Yani örneğin Input-1'e ait bütün veriler kendi arasında normalize edilmiş, bu grup için  $X_{min}$  Input-1'in aldığı minimum değer,  $X_{maks}$  Input-1'in aldığı maksimum değer olarak alınmış ve  $X_{nor}$  buna göre hesaplanmıştır.

Normalizasyon için kullanılan denklem aşağıda verilmiştir:

$$X_{nor} = \frac{X_{reel} - X_{min}}{X_{maks} - X_{min}}$$

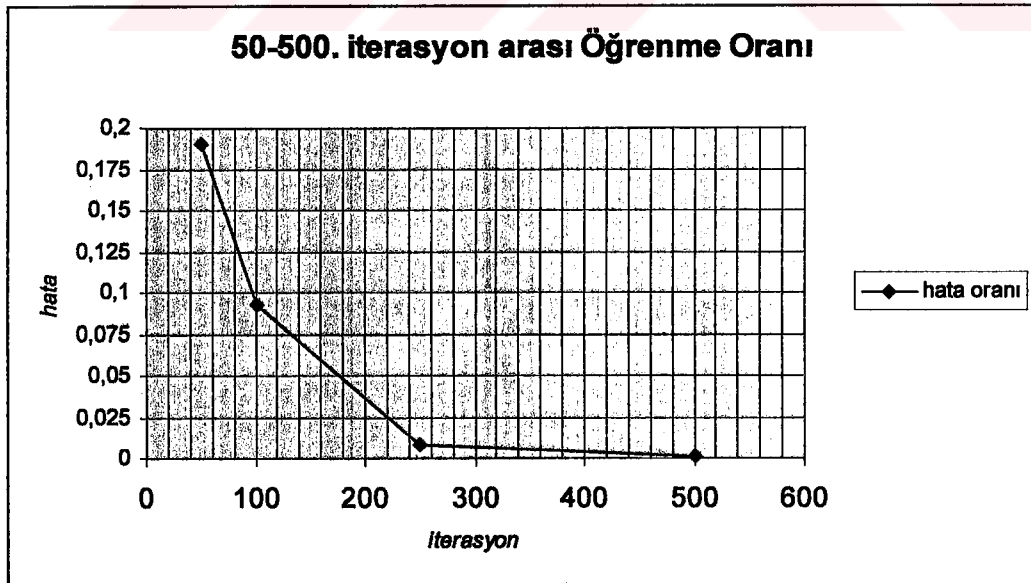
$X_{nor}$  : Normalize edilmiş değer  
 $X_{reel}$  : Gerçek değer  
 $X_{min}$  : Minimum değer  
 $X_{maks}$  : Maksimum değer

Normalize edilmiş veriler ise Ek B'de verilmiştir.

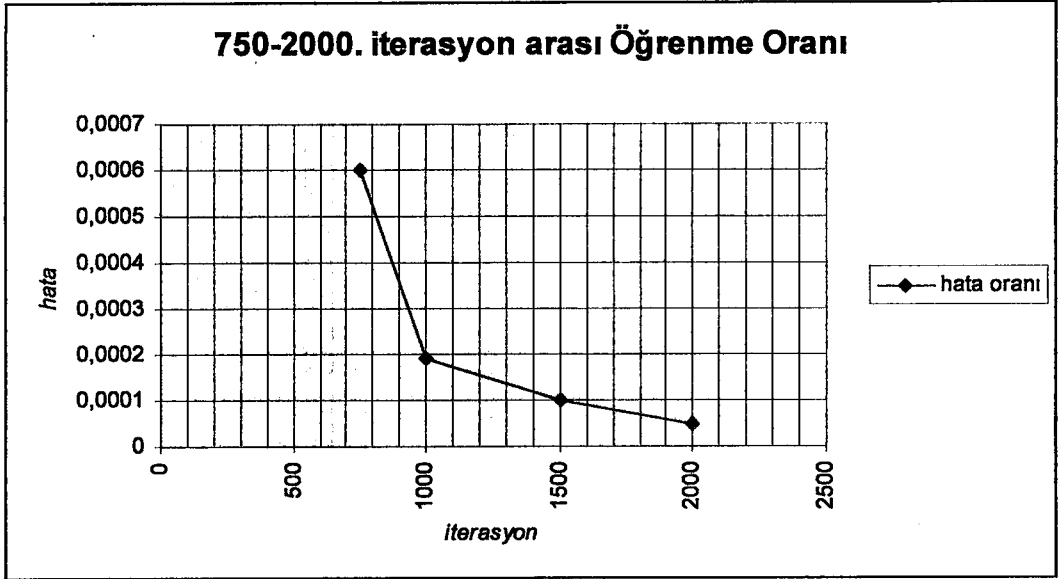
Neurak programı, oluşturacağı ağı için ihtiyaç duyduğu eğitim ve test setlerini bir txt dosyasından alabilmektedir. Bu yüzden ilk olarak normalize edilmiş veriler kopyalanarak bir txt dosyası oluşturulmuştur. Daha sonra program çalıştırılıp, Girdi Sayısı, veri grubu sayısı ve çıktı sayısı programa girilmiş ve karşımıza çıkan Excel tablosuna benzeyen bölümde daha önce kaydedilen txt dosyası çağırılmıştır. Dolayısıyla, programa eğitim için ihtiyaç duyduğu veriler kolaylıkla girilmiştir. Veri grubunun bir kısmı kopyalanarak test setine aktarılmıştır. Program bu aşamadan sonra veri grubu içinde birbiriyle aynı verilerin kullanıldığı grupların olup olmadığını test etmiştir. Program artık ağı eğitmeye hazır hale gelmiştir.

Program eğitim aşamasında bize değiştirebileceğimiz dört adet parametre sunmuştur. Bunlar Gizli Katman Nöron Sayısı (Number of Neurons), İterasyon Sayısı (Maximum Number of Epochs), Momentum Faktörü (Momentum Factor) ve İstenilen Hata oranı (Convergence Criterium) dır. Bir sonraki aşamada girilen bu dört parametre kullanılarak, program ağa verileri atmış, oluşan çıktı ile gerçek çıktı arasındaki hata oranını geriye doğru ağırlıkları değiştirerek dağıtmış, aynı veriler tekrar ağa atılmıştır. Bu işlem aslında istenen hata oranı yakalanana dek –ki ağ o zaman eğitilmiştir-, ya da girdiğimiz iterasyon sayısı tamamlanana kadar devam eder. Programda geriye doğru giderek bu dört parametreyi tekrar değiştirme şansımız vardır. En düşük hata oranını yakalamak için diğer üç parametrenin uygun değerlere getirilmesi gerekmektedir.

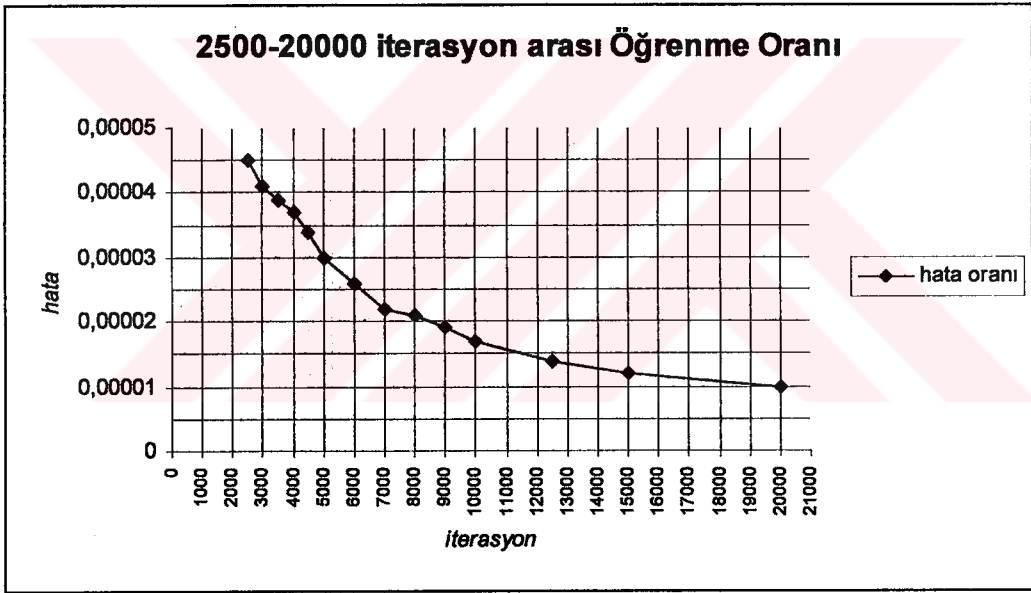
Bu çalışmada ağ Gizli Katman Nöron Sayısı: 12, Momentum Faktörü: 0.64, değerleri girilerek eğitilmiş ve 0,00001 hata oranını 20100 iterasyon sonunda yakalamıştır. Bazı iterasyon değerlerindeki hata oranları Tablo 4.1 de, Öğrenme Oranları da Şekil 4.1, Şekil 4.2 ve Şekil 4.3’de gösterilmiştir.



**Şekil 4.1. 50-500. İterasyon Arası Öğrenme Oranı**



**Şekil 4.2. 750-2000. İterasyon Arası Öğrenme Oranı**



**Şekil 4.3. 2500-20000. İterasyon Arası Öğrenme Oranı**

**Tablo 4.1. Çeşitli İterasyonlara Karşılık Gelen Hata Oranları**

İterasyon	50	100	250	500	750	1000	1500	2000	2500	3000
Hata Oranı	0,2	0,09	0,009	0,001	0,0006	0,0002	0,000098	0,00005	0,000045	0,000041

İterasyon	3500	4000	4500	5000	6000	7000
Hata Oranı	0,000039	0,000037	0,000034	0,00003	0,000026	0,000022

İterasyon	8000	9000	10000	12500	15000	20000
Hata Oranı	0,000021	0,000019	0,000017	0,000014	0,000012	0,00001

Daha sonra program test setine kopyalanan ve Tablo 4.1’de verilen verileri ağı atarak test etmiş ve %0,17 hata oranıyla testi tamamlamıştır.

Tablo 4.2. Test Setinde Kullanılan Veriler

	INP-1	INP-2	INP-3	INP-4	INP-5	INP-6	INP-7	INP-8	INP-9	INP-10	OUTPUT
NO	HAM MADDE	AMBALAJ	İŞÇİLİK	SU & ELEKTRİK	SARF MALZEME	PERSONEL	DIŞ. SAĞL. FAY. HİZM.	AMORTİS.	BAKIM	DIĞER	ÜRETİM MİKTARI
1	0,2917539	0,3012266	0,3348322	0,2671006	0,2840169	0,3319911	0,3001248	0,2175421	0,1332206	0,0853121	0,2072341
2	0,0507831	0,0340250	0,1201430	0,0187022	0,0122652	0,1114094	0,0307116	0,0529892	0,0266441	0,0169227	0,1019167
3	0,9062819	0,2332185	0,3163842	0,6528208	0,8417018	0,8427293	0,3253433	0,4801560	0,5191372	0,2808639	1,0000000
4	0,2525944	0,2080148	0,6026750	0,3199008	0,5070908	0,5516779	0,2589263	0,3282566	0,0566716	0,7029655	0,1106760
5	0,2437426	0,2547677	0,4485184	0,2231866	0,8899962	0,6402685	0,5038702	0,3614666	0,1454853	0,2824756	0,1716818
6	0,0249690	0,0164244	0,0360890	0,0043397	0,0000000	0,0272931	0,0112360	0,0088819	0,0050751	0,0059633	0,0386439
7	0,6731562	0,3018987	0,5462931	0,5876214	0,6814872	0,5458613	0,8439451	0,4469748	0,4929160	0,1716987	0,5928483
8	0,0744686	0,0592288	0,1421653	0,0242819	0,1640475	0,1836689	0,0474407	0,0533923	0,0211461	0,0105834	0,1076876
9	0,0531712	0,0363774	0,0769053	0,0116760	0,0091989	0,0606264	0,0244694	0,0250191	0,0105731	0,0124637	0,0907873
10	0,2874420	0,2956818	0,5279603	0,3334367	0,8351859	0,6612975	0,3280899	0,1984395	0,5316134	0,1639089	0,2001237
11	0,1070549	0,1084601	0,1807910	0,0852449	0,5599847	0,2798658	0,2284644	0,1913427	0,2315500	0,1943698	0,0618302
12	0,0940815	0,0825002	0,1274069	0,0464972	0,0996550	0,1326622	0,0431960	0,0957721	0,0820469	0,0156334	0,2827700
14	0,0786219	0,0638915	0,1057304	0,0888613	0,0682254	0,1731544	0,0973783	0,0882002	0,0365828	0,2583539	0,0197857
15	0,2476392	0,0603629	0,1255621	0,1777227	0,2717516	0,3366890	0,3670412	0,2049462	0,2467752	0,1085205	0,2856554
16	0,6203715	0,2425859	0,5984089	0,6451746	0,7301648	0,3250559	0,1023720	0,2283674	0,3121167	0,2046309	0,5627576
17	0,4371896	0,4938251	0,3447481	0,4007026	0,2621694	0,3689038	0,6521848	0,2456274	0,1552125	0,0792414	0,3567601
18	0,3454472	0,1307654	0,7257004	0,3999793	0,6255270	0,5246085	0,4898876	0,3554062	0,2622119	0,0911679	0,2849340
19	0,1270255	0,1757120	0,1623429	0,2007646	0,3522422	0,1187919	0,0721598	0,2334058	0,0604779	0,1375846	0,0511129
20	0,0205532	0,0197009	0,0320535	0,0095061	0,0122652	0,0375839	0,0109863	0,0291361	0,0006344	0,0121951	0,0792457
21	0,1262929	0,0954381	0,1340943	0,0447406	0,0923726	0,1156600	0,0511860	0,0762088	0,1689575	0,0309982	0,2742168
22	0,0726573	0,0633874	0,1443560	0,0308948	0,0678421	0,1449664	0,0501873	0,0634834	0,0647071	0,0082196	0,1917766



## SONUÇ ve ÖNERİLER

### Sonuçlar

EK B'deki normalize edilmiş veriler kullanılarak ağda en uygun sonucu verecek Gizli Katman Nöron Sayısı ve Momentum Faktörü aşağıda verilmiştir. Bu değerleri bulabilmek için öncelikle ağ İterasyon Sayısı 2000 olarak verilmiştir. Daha sonra Gizli Katman Nöron Sayısı değiştirilerek 2000 iterasyonda en düşük hata oranını veren Nöron Sayısı bulunmuştur. Daha sonra aynı iterasyon ve Nöron sayısı kullanılarak değişik Momentum Faktörü değerleri girilmiş ve yine en düşük hata oranını veren Momentum Faktörü seçilmiştir.

Gizli Katman Nöron Sayısı : 12  
Momentum Faktörü : 0,64

Daha sonra ağ yukarıdaki değerler girilmiş, hata oranı 0,00001 olarak verilmiş ve iterasyon sayısı artırılarak bu hata oranı yakalanmaya çalışılmıştır. Ağ 20100 iterasyon sonucu istenen hata değerini sağlamıştır.

Veri grubunun bir kısmı Neurak programının test setinde kullanılmış ve;

Test Seti Hata Oranı : 0,0017 olarak bulunmuş ve ağ kaydedilmiştir.

Ağ kaydedildikten sonra Neurak txt uzantılı iki ve nk uzantılı bir adet dosya oluşturmuştur. Bunlardan birincisi Ağın adı, Girdi Sayısı, Gizli Katmandaki Nöron Sayısı ve Output Sayısını vermiştir. Çıktısı aşağıdadır;

Neurak - Neural Networks Development Environment  
<http://www.quaternions.net>

Project Name: tez  
Description:

Number of Inputs: 10  
Number of Outputs: 1  
Number of Neurons: 12

Program ikinci dosyada; programın bulduğu ve ağı sağlayan ağırlıkları vermiştir. aşağıda verilmiştir;

```
w1 = [  
2,639 1,129 0,5322 0,4661 0,3171 1,431 0,6385 -0,7771 1,087 0,251 -0,2717 1,33  
-0,1445 0,2534 0,265 0,07127 0,5453 -0,1076 -0,2143 1,265 0,1618 0,142 1,592 1,448  
0,3089 0,26 0,3432 0,4677 0,2046 0,619 -0,4106 0,8015 0,4467 0,1722 0,1242 0,7236  
-1,273 1,216 0,6021 0,4248 0,3748 0,2864 0,3935 2,17 1,058 0,528 0,2409 1,656  
-0,2756 1,017 0,494 0,2648 0,5001 0,9379 0,2409 -0,3456 0,9672 0,6244 2,326 0,5516  
0,1875 0,955 0,2759 0,2501 0,5662 -0,2099 0,4549 -1,209 0,6306 0,3946 1,298 0,5052  
0,758 0,4627 0,4817 0,1856 0,6409 1,773 -0,2221 0,1076 0,1294 0,08273 1,051 0,9724  
0,1625 1,234 0,4556 0,1777 0,5572 0,894 -0,6784 1,611 1,122 0,8895 0,6964 0,2345  
0,3606 0,3079 0,4401 0,2682 0,2476 2,068 -0,5318 1,239 0,1242 0,1491 0,8502 -0,324  
0,4962 0,7226 0,1047 -0,3751 0,5077 1,502 -0,8599 2,567 0,3592 0,5159 -0,1277 2,857  
];  
b1 = [  
0,2274 0,2032 0,7294 0,3579 -0,005559 -0,3332 -0,4714 0,3281 0,3297 -0,3426 -0,2058  
-1,659  
];  
w2 = [  
2,154  
-1,397  
-0,4647  
-0,6013  
0,4767  
-1,818  
1,017  
2,495  
-1,335  
0,7247  
1,939  
-0,7112  
];  
b2 = [  
-0,2386  
];
```

w1 Matrisi (weight-1) : Girdi Katmanı ile Gizli Katma arası ağırlıkları,  
b1 vektörü (bias-1) : Girdi Katmanı ile Gizli Katman arasındaki eşik değerlerini,  
w2 vektörü (weight-2) : Gizli Katman ile Çıktı Katmanı arası ağırlıkları,  
b2 değeri (bias-2) : Gizli Katman ile Çıktı Katmanı arası eşik değerini,

ifade etmektedir.

Programın üçüncü çıktısı ise Neurak v1.1'in simülasyonda kullanmak için nk uzantısıyla kaydettiği ağıdır (tez.nk).

Kaydedilen ağı Neurak Programının Simülasyon kısmında (*Neurak Sim*) çalıştırılarak veri grubundan çeşitli vektörler seçilmiş; bunların sadece girdi değerleri ağı girilmiş ve çıktıdaki 0,0017 hata oranının sağlandığı görülmüştür.

## Öneriler

Bir Yapay Sinir Ağı programı, diğer programlama dillerinden farklı olarak; sadece herhangi bir matematiksel fonksiyonun çözümü için değil, aralarında matematiksel bir fonksiyonla ifade edilemeyecek bir ilişki mevcut olan, birçok problemin çözümünde kullanılabilir. Yani bir Yapay Sinir Ağı Programı kullanılarak birden fazla problemin çözümü için o probleme ait ağlar oluşturulabilir.

Fakat bir problemin Yapay Sinir Ağı ile çözülebilmesi için yeterli sayıda veri olması, çıktı katmanı ve girdi katmanı arasındaki ilişkinin iyi bir şekilde tesbit edilmesi, çıktıya etki eden girdilerin tamamının girdi katmanında mevcut olması...vb önemli ayrıntılara dikkat edilmelidir.

Nitekim bu tezdeki uygulamada Toprak Sağlık Ürünleri Fabrikasından alınan 62 aylık maliyet bilgileri girdi, üretim miktarları ise çıktı olarak alınmıştır. Bu 62 vektör kullanılarak ağ eğitilmiş ve test setinde kullanılan vektörler de bu 62 vektör arasından rasgele seçilmiştir. Eğitilen ağ kullanıldığında verilen girdi değerleri de bu 62 vektör arasından seçilmiştir.

Uygulamada 62 vektörün bir kısmı eğitim için, kalan kısmı test için kullanıldığında hata payının en fazla %23'e düşürülebildiği görülmüştür. Bunun sebebi araştırılmış ve neticesinde ağın bu şekilde kullanılamamasının birkaç sebebi olduğu tesbit edilmiştir. Verilerin alındığı 62 ay düşünüldüğünde bu dönemde ülkemiz iki ciddi ekonomik kriz geçirmiş ve neticesinde Toprak Sağlık Ürünleri Fabrikası bu dönemlerde üretiminde ve maliyetlerinde (personele ücretsiz izinler, özellikle hammadde maliyetinde olmak üzere maliyetlerdeki büyük artış...) büyük dengesizlikler yaşamıştır.

Ayrıca işletmede bugün alınan bir hammadde 8-10 ay sonra kullanılabilen ve dolayısıyla maliyetlere 8-10 ay sonra düşebilmektedir. Üretimde de istikrar olmadığı düşünüldüğünde %23 hata oranının sebebi açıkça görülmektedir.

Görülüyor ki; biz bu uygulamayla sadece bir Yapay Sinir Ağı'nın öğrenebildiğini görebilmekteyiz. Faaliyette bir istikrar olduğunda, veri sayısı yeterli olduğunda, girdi çıktı arasındaki ilişki iyi tesbit edildiğinde bir Yapay Sinir Ağı uygulaması bu alanda çok kullanışlı olacaktır.

Bir başka husus, Yapay Sinir Ağları insan beyinde işleyen öğrenme fonksiyonunu taklit eden bilgisayar programlarıdır. Bir Yapay Sinir Ağı eğitilirken, uygun sonucu verebilmesi için momentum faktörü ve gizli katman nöron sayısının en uygun değerlere getirilmesi anlamında dışarıdan bir müdahaleye ihtiyaç duymaktadır. Fakat insan beyni öğrenme fonksiyonunu dışarıdan hiçbir müdahale olmaksızın kendisi yapmaktadır.

Bu anlamda Yapay Sinir Ağı oluşturmak için kullanılan program geliştirilerek bu iki parametreyi de kendisi belirleyebilir. Dolayısıyla program, kullanıcı müdahalesi olmaksızın, sadece veri grubu ve istenen hata oranı verilerek istenen ağı oluşturacak hale getirilebilir. Bu, mevcut programlara göre hem zaman kaybını engellemiş olur, hem de bu sayede daha uygun sonuçlar elde edilebilir.

## KAYNAKLAR

### Kitaplar

**CHUA, L.O., and YANG, L.,** “Celluar Neural Networks : Application”, IEEE Transaction on Circuits and Systems, vol.35, pp. 1273-1290, October 1988.

**HOPFIELD, J.J.,** “NNs and Physical System with Emergent Collective Computational Abilities”, Proc. Nat’l Academy Science, Vol. 79, pp2254-2258, USA, April, 1982.

**HOPFIELD ,J.J., and TANK, D.W.,** “Neural Computation of Decisions in Optimization Problems”, Biological Cybernetics, vol.52, pp. 141-152, 1985.

**McCULLOCH, W.S., and PITTS, W.,** “A Logical Calculus of the Ideas Immanent in Nervous Activity”, Bull. Math. Biophysics, Vol. 5,pp. 115-133, 1943.

**TOBORG, T.S., and HWANG.,K.,** “Exploring Neural Network and Optical Computing Technologies”, Parallel Processing for Supercomputers and Artificial Intelligence, der. Hwang K., D., Degroot, McGraw-Hill, s: 609-656, 1989.

**TURBAN, E.,** “Decision Support and Expert Systems: Management Support System”, Prentice-Hall, pp.681-714, 1992.

**WINSTANLEY, GRAHAM,** “AI in Engineering”, Courier Int. Ltd., England, 1991.

**ZURADA, J. M.,** “Introduction to Artificial Neural Systems”, West Publishing Company, St. Paul, 1991.

## **Makaleler**

**GEVARTEL, W. B.**, “Intelligent Machines”, New Jersey: Prentice Hall Inc., pp4, 1985.

**KOCABAŞ, Ş.**, “Yarınlarımıza Yapay Zeka mı Hükmedecek?”, Çizgi Üstü, İstanbul, Sayı:1, İstanbul, s.4-7, 1992.

**LIPPMANN, R.P.**, “An Introduction to Computing with Neural Nets”, Acoustics, Speech and Signal Processing Magazine, vol.4, No: 2, pp.4-22, Apr 1987.

**LIPPMANN, R.P.**, “An Introduction to Computing with Neural Nets”, IEEE Asp Magazine, vol.4, pp.4-24, 1987.

**OZTEMEL, E.**, “Integrating Expert Systems and NNs for Intelligent On-Line Statistical Process Control”, University of Wales, PhD., School of Electrical – Electronic and Systems Engineering, Cardif, December, pp. 14, 1992.

**ÖZTEMEL, E.**, “Bilgisayarda Öğrenme ve Yapay Sinir Ağları”, Otomasyon Dergisi, s.134-140, 1996.

## **Diğer Kaynaklar**

**BAKLAVACI, S.**, “Yapay Sinir Ağlarında Öğrenme Algoritmalarının Analizi”, Yüksek Lisans Tezi, İTÜ, Ocak 1994.

**GÜZELİŞ, C.**, “Yapay Sinir Ağları, Yüksek Lisans Ders Notları”, İTÜ, 1991.

**KARAMAHMUT, S.**, “HYSA için İki Öğrenme Algoritması ve Görüntü İşleme Uygulamaları”, Yük. Lis. Tezi, İTÜ, Elektrik-Elektronik Bölümü, s.17, 1994.

**MEHRA, P., and BENJAMIN, W.W.**, “Artificial Neural Networks: Concepts and Theory”, IIEE Com. Soc. Press Tutorial, Los Vaqueros, USA, 1992.

**TOSUN M.**, “Geriyansız Yapay Sinir Ağının Görüntü Sıkıştırma Kullanılması”, İTÜ, FBE, Y.Lisans Tezi, 1997.

**WHITE, A.D., and SOFGE, D.A.**, (edt. By), “Handbook of Intelligence Control”, Van Nostrand Reinhold, s.11-235, NewYork, 1992.

[www.members.tripod.com/~Bagem/bagem/yz3.html](http://www.members.tripod.com/~Bagem/bagem/yz3.html)

[www.pmsi.fr/neurin2a.htm](http://www.pmsi.fr/neurin2a.htm)

<http://rfhs8012.fh-regensburg.de/%7Esaj39122/jfroehl/diplom/e-index.html>

[www.doc.ic.ac.uk/~nd/surprise\\_96/journal/vol4/cs11/report.html](http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html)

## EKLER

### EK A. Ağda Kullanılan Veri Grubu

	INP-1	INP-2	INP-3	INP-4	INP-5	INP-6	INP-7	INP-8	INP-9	INP-10	OUTPUT
AY	HAMMADDE	AMBALAJ	İŞÇİLİK	SU & ELEKTRİK	SARF MALZEME	PERSONEL	DIŞ. SAĞL. FAY. HİZM.	AMORTİS	BAKIM	DİĞER	ÜRETİM MİKTARI
1	67.403	2.414	1.284	996	465	1.583	414	22.014	767	5.717	2.436
2	42.530	910	731	655	327	600	1.961	8.544	777	769	1.658
3	144.413	2.962	2.562	2.638	1.162	2.133	876	22.897	1.757	2.580	4.471
4	88.683	1.604	1.193	1.762	736	1.558	1.487	14.937	1.173	2.052	2.991
5	287.494	5.873	2.581	5.058	2.053	3.721	1.978	32.195	716	4.533	9.420
6	317.041	5.719	2.848	6.360	2.223	3.820	1.320	34.055	2.461	5.260	9.923
7	226.656	4.435	5.909	4.211	1.534	3.841	972	21.355	1.054	4.553	6.847
8	136.650	3.943	4.176	3.570	1.120	4.523	990	11.162	1.227	5.036	3.695
9	172.452	3.697	7.844	3.391	2.460	3.662	994	24.716	1.374	6.407	4.977
10	244.332	5.340	6.113	6.910	2.636	3.562	779	28.721	3.245	5.135	7.082
11	174.596	5.119	4.327	5.023	2.491	1.594	510	14.263	1.938	2.554	4.405
12	217.913	5.942	5.294	6.286	1.932	1.506	427	16.564	1.482	3.841	5.680
13	122.594	3.280	6.398	3.913	1.659	2.398	1.979	25.389	1.246	1.729	2.984
14	236.214	7.354	4.842	5.729	1.805	2.493	3.397	31.750	2.337	3.228	5.972
15	299.771	8.288	5.962	5.917	1.644	2.483	3.566	35.243	1.506	1.667	7.364
16	349.534	9.355	7.953	6.655	1.287	4.122	3.028	35.531	2.780	2.021	7.854
17	194.587	4.844	6.812	3.659	1.080	3.490	2.145	21.836	4.735	2.101	4.209
18	65.714	4.170	2.788	2.013	828	1.634	843	11.002	524	1.103	1.425
19	96.290	8.242	5.974	3.154	1.008	2.772	2.949	16.268	2.411	2.541	2.191
20	154.402	11.923	3.094	3.920	711	1.702	2.629	17.763	740	1.507	3.681
21	103.978	7.338	3.008	2.627	768	1.537	1.219	15.812	636	1.620	2.230
22	316.673	23.973	8.400	6.936	1.891	4.418	4.022	37.575	1.948	3.645	6.763
23	99.544	7.220	4.003	2.730	1.147	2.285	2.957	14.243	1.102	2.104	2.056
24	102.483	7.206	4.683	3.269	2.206	3.009	1.331	14.485	2.520	3.083	2.161



25	104.500	7.100	6.019	2.870	1.205	2.395	2.609	16.515	1.043	2.012	2.161
26	46.865	4.350	1.512	1.985	946	584	306	16.914	292	2.593	715
27	156.379	15.575	5.151	7.368	1.491	3.283	1.581	54.260	791	14.107	1.765
28	264.547	23.909	8.777	9.720	2.594	3.158	3.752	70.167	2.195	18.646	2.758
29	63.462	6.116	1.600	2.392	531	458	574	15.573	337	2.440	671
30	87.332	6.232	3.994	2.202	2.349	2.915	2.035	25.810	694	5.290	1.885
31	39.941	2.749	1.672	867	1.488	1.304	932	13.992	1.101	3.650	819
32	59.323	3.826	2.598	927	784	4.241	924	22.037	540	7.316	1.175
33	78.171	3.856	4.217	3.043	389	2.955	1.665	23.881	290	7.067	1.118
34	30.083	1.688	1.021	902	205	827	407	6.827	179	4.841	411
35	90.401	5.119	5.331	3.138	1.350	2.519	1.054	23.503	274	13.117	1.293
36	102.373	6.801	2.075	3.635	1.136	2.043	845	21.947	207	15.816	1.381
37	45.569	2.967	1.397	1.864	344	2.708	1.688	16.380	728	12.846	620
38	20.431	977	1.146	223	59	551	140	4.381	132	347	1.208
39	52.501	2.512	2.945	574	152	1.415	360	11.257	338	890	3.103
40	21.259	1.033	771	155	51	324	115	2.438	56	264	1.100
41	11.481	558	417	84	27	175	62	1.317	30	143	594
42	78.768	4.923	2.863	576	187	1.198	429	9.147	207	989	4.011
43	28.643	1.577	1.337	277	455	874	207	4.409	106	229	1.264
44	9.950	636	382	134	59	221	61	2.724	9	259	988
45	6.266	400	242	85	37	139	47	1.716	6	162	622
46	19.266	1.183	715	211	86	391	109	4.638	165	301	1.792
47	24.588	1.510	981	290	119	535	151	5.958	226	409	2.287
48	11.668	731	402	165	103	223	89	2.808	48	213	1.030
49	10.723	675	496	150	87	225	82	2.561	65	153	936
50	4.224	266	177	53	31	80	29	999	23	55	369
51	32.084	1.928	1.191	485	281	636	185	6.707	389	322	2.682
52	35.443	2.131	1.209	492	287	646	190	7.353	394	323	2.963
53	2.824	167	104	42	27	53	17	700	34	32	219
54	38.495	2.052	1.581	417	269	801	173	6.484	121	515	2.742

55	34.060	1.816	1.453	383	247	733	158	5.755	109	471	2.426
56	29.699	1.782	1.479	342	304	732	210	5.377	282	259	2.237
57	28.015	1.676	1.356	341	204	701	218	5.110	312	185	2.080
58	10.261	559	451	126	117	228	53	1.965	180	72	648
59	57.781	3.023	1.628	610	343	732	284	7.453	1.037	791	3.570
60	46.611	2.439	1.267	475	268	570	222	5.994	805	609	2.880
61	9.860	479	334	82	93	167	70	1.264	126	151	537
62	20.000	971	926	228	259	463	197	2.648	348	418	1.090



## EK B. Normalize Edilmiş Veri Grubu

NO	INP-1 HAM MADDE	INP-2 AMBALAJ	INP-3 İŞÇİLİK	INP-4 SU & ELEKTRİK	INP-5 SARF MALZEME	INP-6 PERSONEL	INP-7 DIŞ. SAĞL. FAY. HİZM.	INP-8 AMORTİS.	INP-9 BAKIM	INP-10 DİĞER	OUTPUT ÜRETİM MİKTARI
1	0,1862623	0,0943880	0,1360544	0,0985741	0,1678804	0,3422819	0,0991261	0,3068219	0,1609220	0,3054153	0,2284625
2	0,1145222	0,0312106	0,0722933	0,0633395	0,1149866	0,1223714	0,4853933	0,1129169	0,1630366	0,0395939	0,1482894
3	0,4083788	0,1174074	0,2834083	0,2682372	0,4350326	0,4653244	0,2144819	0,3195330	0,3702686	0,1368862	0,4381698
4	0,2476392	0,0603629	0,1255621	0,1777227	0,2717516	0,3366890	0,3670412	0,2049462	0,2467752	0,1085205	0,2856554
5	0,8210608	0,2396875	0,2855990	0,5182889	0,7765427	0,8205817	0,4896380	0,4533807	0,1501374	0,2418072	0,9481657
6	0,9062819	0,2332185	0,3163842	0,6528208	0,8417018	0,8427293	0,3253433	0,4801560	0,5191372	0,2808639	1,0000000
7	0,6455885	0,1792825	0,6693186	0,4307708	0,5776159	0,8474273	0,2384519	0,2973354	0,2216113	0,2428817	0,6830173
8	0,3859883	0,1586155	0,4695031	0,3645381	0,4189345	1,0000000	0,2429463	0,1506039	0,2581941	0,2688299	0,3582028
9	0,4892504	0,1482819	0,8924248	0,3460426	0,9325412	0,8073826	0,2439451	0,3457181	0,2892789	0,3424842	0,4903133
10	0,6965706	0,2172982	0,6928398	0,7096508	1,0000000	0,7850112	0,1902622	0,4033714	0,6849228	0,2741485	0,7072341
11	0,4954342	0,2080148	0,4869134	0,5146725	0,9444232	0,3447427	0,1230961	0,1952438	0,4085430	0,1354894	0,4313685
12	0,6203715	0,2425859	0,5984089	0,6451746	0,7301648	0,3250559	0,1023720	0,2283674	0,3121167	0,2046309	0,5627576
13	0,3454472	0,1307654	0,7257004	0,3999793	0,6255270	0,5246085	0,4898876	0,3554062	0,2622119	0,0911679	0,2849340
14	0,6731562	0,3018987	0,5462931	0,5876214	0,6814872	0,5458613	0,8439451	0,4469748	0,4929160	0,1716987	0,5928483
15	0,8564708	0,3411325	0,6754295	0,6070469	0,6197777	0,5436242	0,8861423	0,4972577	0,3171918	0,0878371	0,7362943
16	1,0000000	0,3859531	0,9049925	0,6833023	0,4829437	0,9102908	0,7518102	0,5014035	0,5865934	0,1068551	0,7867890
17	0,5530934	0,1964631	0,7734348	0,3737342	0,4036029	0,7689038	0,5313358	0,3042596	1,0000000	0,1111529	0,4111707
18	0,1813908	0,1681509	0,3094662	0,2036578	0,3070142	0,3536913	0,2062422	0,1483006	0,1095369	0,0575373	0,1242786
19	0,2695798	0,3392002	0,6768131	0,3215540	0,3760061	0,6082774	0,7320849	0,2241064	0,5085642	0,1347910	0,2032152
20	0,4371896	0,4938251	0,3447481	0,4007026	0,2621694	0,3689038	0,6521848	0,2456274	0,1552125	0,0792414	0,3567601
21	0,2917539	0,3012266	0,3348322	0,2671006	0,2840169	0,3319911	0,3001248	0,2175421	0,1332206	0,0853121	0,2072341
22	0,9052205	1,0000000	0,9565318	0,7123373	0,7144500	0,9765101	1,0000000	0,5308276	0,4106576	0,1941012	0,6743611
23	0,2789651	0,2962698	0,4495561	0,2777433	0,4292833	0,4993289	0,7340824	0,1949559	0,2317615	0,1113141	0,1893034
24	0,2874420	0,2956818	0,5279603	0,3334367	0,8351859	0,6612975	0,3280899	0,1984395	0,5316134	0,1639089	0,2001237
25	0,2932595	0,2912291	0,6820016	0,2922091	0,4515140	0,5239374	0,6471910	0,2276621	0,2192853	0,1063715	0,2001237
26	0,1270255	0,1757120	0,1623429	0,2007646	0,3522422	0,1187919	0,0721598	0,2334058	0,0604779	0,1375846	0,0511129
27	0,4428918	0,6472318	0,5819209	0,7569746	0,5611345	0,7225951	0,3905119	0,7710136	0,1659970	0,7561513	0,1593157
28	0,7548758	0,9973116	1,0000000	1,0000000	0,9839019	0,6946309	0,9325843	1,0000000	0,4628886	1,0000000	0,2616447
29	0,1748954	0,2498950	0,1724893	0,2428188	0,1931775	0,0906040	0,1390762	0,2141017	0,0699937	0,1293650	0,0465787
30	0,2437426	0,2547677	0,4485184	0,2231866	0,8899962	0,6402685	0,5038702	0,3614666	0,1454853	0,2824756	0,1716818
31	0,1070549	0,1084601	0,1807910	0,0852449	0,5599847	0,2798658	0,2284644	0,1913427	0,2315500	0,1943698	0,0618302
32	0,1629575	0,1537007	0,2875591	0,0914445	0,2901495	0,9369128	0,2264669	0,3071530	0,1129203	0,3913184	0,0985161
33	0,2173200	0,1549609	0,4742304	0,3100847	0,1387505	0,6492170	0,4114856	0,3336980	0,0600550	0,3779413	0,0926422
34	0,0786219	0,0638915	0,1057304	0,0888613	0,0682254	0,1731544	0,0973783	0,0882002	0,0365828	0,2583539	0,0197857
35	0,2525944	0,2080148	0,6026750	0,3199008	0,5070908	0,5516779	0,2589263	0,3282566	0,0566716	0,7029655	0,1106760
36	0,2871247	0,2786692	0,2272570	0,3712544	0,4250671	0,4451902	0,2067416	0,3058575	0,0425037	0,8479639	0,1197444
37	0,1232875	0,1176174	0,1490834	0,1882620	0,1215025	0,5939597	0,4172285	0,2257187	0,1526750	0,6884066	0,0413232
38	0,0507831	0,0340250	0,1201430	0,0187022	0,0122652	0,1114094	0,0307116	0,0529892	0,0266441	0,0169227	0,1019167
39	0,1432811	0,0985046	0,3275683	0,0549700	0,0479111	0,3046980	0,0856429	0,1519714	0,0702051	0,0460943	0,2971970
40	0,0531712	0,0363774	0,0769053	0,0116760	0,0091989	0,0606264	0,0244694	0,0250191	0,0105731	0,0124637	0,0907873
41	0,0249690	0,0164244	0,0360890	0,0043397	0,0000000	0,0272931	0,0112360	0,0088819	0,0050751	0,0059633	0,0386439
42	0,2190419	0,1997816	0,3181137	0,0551767	0,0613262	0,2561521	0,1028714	0,1215973	0,0425037	0,0514129	0,3907667
43	0,0744686	0,0592288	0,1421653	0,0242819	0,1640475	0,1836689	0,0474407	0,0533923	0,0211461	0,0105834	0,1076876
44	0,0205532	0,0197009	0,0320535	0,0095061	0,0122652	0,0375839	0,0109863	0,0291361	0,0006344	0,0121951	0,0792457
45	0,0099276	0,0097874	0,0159114	0,0044431	0,0038329	0,0192394	0,0074906	0,0146256	0,0000000	0,0069840	0,0415293
46	0,0474229	0,0426783	0,0704485	0,0174623	0,0226140	0,0756152	0,0229713	0,0566888	0,0336223	0,0144515	0,1620981
47	0,0627729	0,0564143	0,1011184	0,0256251	0,0352626	0,1078300	0,0334582	0,0756906	0,0465215	0,0202536	0,2131080

48	0,0255083	0,0236915	0,0343595	0,0127092	0,0291299	0,0380313	0,0179775	0,0303453	0,0088814	0,0097239	0,0835738
49	0,0227827	0,0213392	0,0451977	0,0111593	0,0229973	0,0384787	0,0162297	0,0267897	0,0124762	0,0065005	0,0738871
50	0,0040380	0,0041586	0,0084169	0,0011366	0,0015332	0,0060403	0,0029963	0,0043042	0,0035948	0,0012356	0,0154575
51	0,0843933	0,0739729	0,1253315	0,0457739	0,0973553	0,1304251	0,0419476	0,0864727	0,0809896	0,0155797	0,2538129
52	0,0940815	0,0825002	0,1274069	0,0464972	0,0996550	0,1326622	0,0431960	0,0957721	0,0820469	0,0156334	0,2827700
53	0,0000000	0,0000000	0,0000000	0,0000000	0,0000000	0,0000000	0,0000000	0,0000000	0,0059209	0,0000000	0,0000000
54	0,1028843	0,0791817	0,1702986	0,0387477	0,0927558	0,1673378	0,0389513	0,0832626	0,0243180	0,0259482	0,2599959
55	0,0900926	0,0692683	0,1555402	0,0352346	0,0843235	0,1521253	0,0352060	0,0727684	0,0217805	0,0235844	0,2274320
56	0,0775143	0,0678400	0,1585380	0,0309981	0,1061709	0,1519016	0,0481898	0,0673269	0,0583633	0,0121951	0,2079555
57	0,0726573	0,0633874	0,1443560	0,0308948	0,0678421	0,1449664	0,0501873	0,0634834	0,0647071	0,0082196	0,1917766
58	0,0214502	0,0164664	0,0400092	0,0086795	0,0344960	0,0391499	0,0089888	0,0182101	0,0367942	0,0021489	0,0442086
59	0,1585100	0,1199698	0,1757177	0,0586898	0,1211192	0,1519016	0,0666667	0,0972116	0,2180165	0,0407758	0,3453215
60	0,1262929	0,0954381	0,1340943	0,0447406	0,0923726	0,1156600	0,0511860	0,0762088	0,1689575	0,0309982	0,2742168
61	0,0202936	0,0131059	0,0265191	0,0041331	0,0252970	0,0255034	0,0132335	0,0081190	0,0253753	0,0063930	0,0327700
62	0,0495400	0,0337730	0,0947769	0,0192188	0,0889230	0,0917226	0,0449438	0,0280421	0,0723197	0,0207371	0,0897568



## ÖZGEÇMİŞ

10 Ekim 1974 tarihinde Kayseri’de doğdu. İlk ve orta öğrenimini Kayseri’de tamamladı.1992 yılında İstanbul Teknik Üniversitesi’ne bağlı Sakarya Mühendislik Fakültesi Elektrik-Elektronik Mühendisliği Bölümünde lisans öğrenimine başladı, 1997 yılı Temmuz ayında bu bölümden mezun oldu.

Mart 1998 - Haziran 1999 tarihleri arasında Yedek Subay olarak askerlik görevini tamamladı.

Ağustos 1999’dan Haziran 2002’ye kadar özel sektörde çeşitli firmalarda Elektrik Mühendisi, Otomasyon Mühendisi, Şantiye Şefi pozisyonlarında görevler yaptı.

Eylül 2001’de Sakarya Üniversitesi, Sosyal Bilimler Enstitüsü, İşletme Anabilim Dalına bağlı Üretim Yönetimi ve Pazarlama Bölümünde Yüksek Lisansa başladı.

Haziran 2002’de TOPRAK Holdinge bağlı TOPRAK Sağlık Ürünleri Fabrikasında Elektronik Bakım Mühendisi olarak göreve başladı. Halen bu fabrikada Makine Bakım ve Yardımcı İşletmeler Şefi olarak görevine devam etmektedir.