

**YAPAY SİNİR AĞLARI KULLANILARAK TÜRKİYE'DEKİ  
ÖZEL BİR SİGORTA ŞİRKETİNDE PORTFÖY  
DEĞERLENDİRMESİ**

**VALUATION OF PORTFOLIO WITH USING ARTIFICIAL  
NEURAL NETWORK AT A INSURANCE COMPANY IN  
TURKEY**

**GÜLAY DOĞAN**

Hacettepe Üniversitesi

Lisansüstü Eğitim – Öğretim ve Sınav Yönetmeliğinin

İSTATİSTİK Anabilim Dalı İçin Öngördüğü

**YÜKSEK LİSANS TEZİ**

olarak hazırlanmıştır.

2010

# YAPAY SİNİR AĞLARI KULLANILARAK TÜRKİYE'DEKİ ÖZEL BİR SİGORTA ŞİRKETİNDE PORTFÖY DEĞERLENDİRMESİ

Gülay Doğan

## ÖZ

Yapay zeka tekniklerinden biri olan yapay sinir ağları, son yıllarda öngörü problemlerinde tercih edilen bir modelleme tekniğidir. Özellikle doğrusal olmayan sistemlerde, öngörüler açısından istatistiksel tekniklere göre daha kolaylık sağlayan özelliklere sahiptir.

Çalışmada, Türkiye'deki bir Hayat Sigortası Şirketi'ne ait kredi karşılığı düzenlenen poliçeler için, değerlemede kullanılacak değişkenler yapay sinir ağları yardımıyla öngörülmüştür. Değerlemede kullanılacak ve ileriye dönük yapılan öngörüler için, TCMB USD efektif satış kuru, TCMB EUR efektif satış kuru, kredi karşılığı düzenlenen ve ortalama süresi 3 yıl olan poliçeler için 1 ile 3 yıllık iptal oranları tahmin edilmiştir.

Sigortacılık hakkında kısa bilgi ve tanımlamalar yapılmış olup, zaman serileri analizi için son yıllarda tercih edilen yapay sinir ağları, çalışmada detaylı olarak anlatılmıştır. Matlab paket programı içerisinde yer alan yapay sinir ağları araç paketi kullanılarak uygulama yapılmıştır. Çalışmanın son bölümünde ise, uygulama sonucuna bağlı olarak elde edilen bulgular değerlendirilmiştir.

**ANAHTAR KELİMELELER:** Yapay sinir ağları, hayat sigortası, poliçe iptal oranı, zaman serileri, öngörü.

Danışman: Prof. Dr. Hülya ÇINGİ, Hacettepe Üniversitesi, İstatistik Bölümü

# VALUATION OF PORTFOLIO WITH USING ARTIFICIAL NEURAL NETWORKS AT A INSURANCE COMPANY IN TURKEY

**Gülay Doğan**

## **Abstract**

Artificial Neural Networks is preferred modelling techniques for forecasting problems nowadays, which is one of the promising Artificial Intelligence techniques. Especially in the nonlinear systems, it has more convenient features according to the statistical techniques in terms of predictions.

In the study, for a Life Insurance Company in Turkey belong to a credit policy for the money held, variables used for the valuation were estimated with artificial neural networks. For forecasting will be used in the company valuation, Central Bank of Turkey USD and EUR exchange rate on banknotes selling and the lapse rates which belongs to policies with 3 average duration were forecasted.

Informations and discriptions have been made in the brief about life insurance. Artificial neural networks have been discribed which is a preferred for time series analysis in recent years. Application have been made with using Matlab program. In the last part of the study, findings were evaluated depending on the application results.

**KEY WORDS:** Artificial neural networks, life insurance, policy lapse rate, time series, forecasting.

Advisor: Prof. Dr. Hülya ÇINGI, Hacettepe University, Department of Statistics.

## TEŐEKKÜR

Çalıőmamda bana yardımcı olan ve desteęini esirgemeyen danıőmanım Sayın Prof. Dr. Hülya ÇINGI'ya, ayırmıő olduęu zaman, görüő ve katkılarından dolayı Sayın Dr. Çaędaő Hakan ALADAĖ'a, manevi desteęini esirgemeyen, bana her konuda yardımcı olan Sayın İ. Olgun KÜNTAY'a, her an yanımda olan aileme ve özellikle bu çalıőmamı bitirmemi çok isteyen anneme verdikleri destek ve gösterdikleri özveri için teőekkür ederim.

# İÇİNDEKİLER DİZİNİ

Sayfa

ÖZ.....	i
ABSTRACT.....	ii
TEŞEKKÜR.....	iii
İÇİNDEKİLER DİZİNİ.....	iv
ŞEKİLLER DİZİNİ.....	ix
SİMGELER (ve KISALTMALAR) DİZİNİ.....	xi
ÇİZELGELER DİZİNİ.....	xiii
GİRİŞ .....	1
BİRİNCİ BÖLÜM .....	3
1. ŞİRKET DEĞERİNİN GELİŞİMİ, NEDEN İHTİYAÇ DUYULDUĞU ve DEĞERLEMEDE KULLANILAN ANALİZLER.....	3
1.1. Şirket Değerlemesinin Gelişimi.....	3
1.2. Şirket Değerlemesine Duyulan İhtiyaç.....	3
1.3. Şirket Değerlemesinde Kullanılan Analizler .....	4
1.3.1. Teknik Analiz .....	4
1.3.2. Temel Analiz.....	4
1.3.3. Modern Yaklaşım (Rassal Yürüyüş Teorisi) .....	4
1.4. Genel Kabul Görmüş Şirket Değerleme Yöntemleri .....	4
1.4.1. Kayıtlı Değer.....	5
1.4.2. Piyasa Değeri .....	5
1.4.3. Ekonomik Değer.....	5
İKİNCİ BÖLÜM .....	7
2. SİGORTACILIKLA İLGİLİ TANIMLAMALAR .....	7
2.1. Aktüeryal Bugünkü Değer.....	8
2.2. Teknik Faiz .....	8
2.3. Mortalite Tablosu .....	8
2.4. Prim .....	10
2.5. İştira.....	10

2.6. Vade Gelimi.....	10
2.7. Aktüeryal Matematik Karşılık.....	10
ÜÇÜNCÜ BÖLÜM.....	12
3. YAPAY SİNİR AĞLARI.....	12
3.1. Yapay Sinir Ağlarının Tarihsel Gelişimi .....	15
Y.S.A. İçin Kullanılan Yazılım Kütüphaneleri ve Hazır Arayüzler.....	25
3.2. Yapay Sinir Ağları Süreçleri.....	26
3.3.Y.S.A.'ların Kullanım Alanları ve Yapısal Özellikleri .....	32
3.4. Y.S.A.'nın Kullanım Alanları .....	37
3.5. Y.S.A.'ların Sınıflandırılması.....	40
3.5.1. Y.S.A.'ların Mimari Yapılarına Göre Sınıflandırılması.....	40
3.5.2. Y.S.A.'ların Öğrenme Kuralları .....	46
3.5.3. Y.S.A.'nın Yapısındaki Katmanlara Göre Sınıflandırılması.....	49
3.6. Öğrenme Algoritmalarına Göre Y.S.A. ....	55
3.7. Y.S.A. ile Modelleme .....	65
DÖRDÜNCÜ BÖLÜM.....	76
4. UYGULAMA.....	76
4.1. Çözümlenen Veri.....	76
4.2. Ağ Yapısının Tanımı.....	83
Verilerin Normalleştirilmesi .....	83
Öğrenme Algoritması .....	89
Aktivasyon Fonksiyonu.....	89
Mimari Yapı .....	94
5. SONUÇ VE ÖNERİLER .....	111
KAYNAKLAR.....	118
ÖZGEÇMİŞ.....	124

## SİMGELER VE KISALTMALAR

$\gamma$	Momentum Katsayısı
$\mu$	Marquardt Parametresi
$\eta$	Öğrenme Katsayısı
Ç.K.A.	Çok Katmanlı Algılayıcı
Ç.K.Y.S.A.	Çok Katmanlı Yapay Sinir Ağı
G.B.Y.S.A.	Geri Beslemeli Yapay Sinir Ağı
G.Y.Ö.A.	Geri Yayılım Öğrenme Algoritması
H.K.O.	Hata Kareler Ortalaması
İ.B.Y.S.A.	İleri Beslemeli Yapay Sinir Ağı
L.M.Ö.A.	Levenberg-Marquart Öğrenme Algoritması
M.Y.S.A.	Modül Yapay Sinir Ağı
Ö.V.N.	Öğrenme Vektör Nicelendirmesi
R.T.F.Y.S.A.	Radyal Tabanlı Fonksiyon Yapay Sinir Ağı
R.T.F.	Radyal Tabanlı Fonksiyon
S.G.Y.Ö.A.	Standart Geri Yayılım Öğrenme Algoritması
T.G.B.Y.S.A.	Tam Geri Beslemeli Yapay Sinir Ağı
TCMB	Türkiye Cumhuriyeti Merkez Bankası
Y.S.A.	Yapay Sinir Ağı

## ŞEKİLLER DİZİNİ

	<u>Sayfa</u>
Şekil 3.1. Merkezi sinir sistemi yapısı.....	20
Şekil 3.2. Motor sinir hücre yapısı.....	20
Şekil 3.3. Motor sinir hücre yapısı.....	21
Şekil 3.4. Tek kutuplu adımsal aktivasyon fonksiyonu.....	27
Şekil 3.5. Çift kutuplu adımsal aktivasyon fonksiyonu.....	28
Şekil 3.6. Sigmoid aktivasyon fonksiyonu.....	29
Şekil 3.7. Parçalı doğrusal aktivasyon fonksiyonu.....	29
Şekil 3.8. Gaussian aktivasyon fonksiyonu.....	30
Şekil 3.9. Doğrusal aktivasyon fonksiyonu.....	31
Şekil 3.10. Hiperbolik tanjant aktivasyon fonksiyonu.....	31
Şekil 3.11. G.B.Y.S.A. için blok diyagram.....	43
Şekil 3.12. Danışmanlı öğrenme yapısı.....	47
Şekil 3.13. Danışmansız öğrenme yapısı.....	48
Şekil 3.14. Takviyeli öğrenme yapısı.....	49
Şekil 3.15. Tek katmanlı ağ yapısı.....	50
Şekil 3.16. İleri beslemeli ağ için blok diyagram.....	51
Şekil 3.17. Geri yayılım Ç.K.Y.S.A. yapısı.....	52
Şekil 3.18. Ç.K.İ.B.Y.S.A. genel yapısı.....	54
Şekil 3.19. Y.S.A.'da yer alan katmanlar.....	57
Şekil 3.20. Ç.K.İ.B.Y.S.A.'da geri yayılım akış şeması.....	59
Şekil 3.21. Hızlı ve yüksek performanli algoritmalar.....	61
Şekil 3.22. Momentum değerindeki değişime göre ağırlıkların değişim miktarı.....	73
Şekil 3.23. Eğitimin durdurulması için hata ve iterasyon değerleri.....	73
Şekil 4.1. TCMB USD efektif satış kuru veri grafiği.....	79
Şekil 4.2. TCMB EUR efektif satış kuru veri grafiği.....	82



Şekil 4.3. TCMB USD efektif satış kuruna ait aylar itibariyle orijinal ve tahmin değerleri.....	85
Şekil 4.4. Aylara göre 1 yıllık iptal oranlarına ait orijinal ve tahmin değerleri.....	86
Şekil 4.5. Aylara göre 2 yıllık iptal oranlarına ait orijinal ve tahmin değerleri.....	86
Şekil 4.6. Aylara göre 3 yıllık iptal oranlarına ait orijinal ve tahmin değerleri.....	87
Şekil 4.7. TCMB EUR efektif satış kuruna ait aylar itibariyle orijinal ve tahmin değerleri.....	87
Şekil 4.8. Y.S.A. modelinde 48. modele ait iterasyon sayısı, zaman ve performans değerleri.....	90
Şekil 4.9. TCMB USD efektif satış kuru için H.K.O. ve iterasyon sayıları.....	91
Şekil 4.10. Y.S.A. modelinde 6. modele ait iterasyon sayısı, zaman ve performans değerleri.....	91
Şekil 4.11. 1 Yıllık iptal oranı için H.K.O. ve iterasyon sayıları.....	92
Şekil 4.12. 2 Yıllık iptal oranı için H.K.O. ve iterasyon sayıları.....	93
Şekil 4.13. 3 Yıllık iptal oranı için H.K.O. ve iterasyon sayıları.....	93
Şekil 4.14. TCMB EUR efektif satış kuru için H.K.O. ve iterasyon sayıları.....	94
Şekil 4.15. TCMB USD efektif satış kuru için Y.S.A. modeli mimari yapısı.....	97
Şekil 4.16. Tanzim edilen poliçelerin 1 yıllık iptal oranına ilişkin Y.S.A. modeli mimari yapısı.....	99
Şekil 4.17. Tanzim edilen poliçelerin 2 yıllık iptal oranına ilişkin Y.S.A. modeli mimari yapısı.....	101
Şekil 4.18. Tanzim edilen poliçelerin 3 yıllık iptal oranına ilişkin Y.S.A. modeli mimari yapısı.....	103
Şekil 4.19. TCMB EUR efektif satış kuru için Y.S.A. modeli mimari yapısı.....	106

## ÇİZELGELER DİZİNİ

### Sayfa

Çizelge 3.1: İstatistik terminolojisinde kullanılan terimlerin Y.S.A. terminolojisindeki karşılıkları (Günay vd., 2007a).....	21
Çizelge 4.1: TCMB USD efektif satış kuru veri kümesi.....	77
Çizelge 4.2: TCMB EUR efektif satış kuru veri kümesi.....	80
Çizelge 4.3: TCMB USD efektif satış kuru için çalıştırılan modellere ait H.K.O. ve iterasyon sayıları.....	95
Çizelge 4.4: 1 Yıllık iptal oranı için çalıştırılan modellere ait H.K.O. ve iterasyon sayıları.....	98
Çizelge 4.5: 2 Yıllık iptal oranı için çalıştırılan modellere ait H.K.O. ve iterasyon sayıları.....	100
Çizelge 4.6: 3 Yıllık iptal oranı için çalıştırılan modellere ait H.K.O. ve iterasyon sayıları.....	101
Çizelge 4.7: TCMB EUR efektif satış kuru için çalıştırılan modelere ait H.K.O. ve iterasyon sayıları.....	103
Çizelge 4.8: TCMB USD efektif satış kuru tahmin değerleri.....	107
Çizelge 4.9: TCMB EUR efektif satış kuru tahmin değerleri.....	109

## GİRİŞ

Geleceğin öngörülmesi, özellikle ekonomi ve finans alanında yaygın olarak ihtiyaç duyulan bir konudur. Zamanımızda bu alanlarda faaliyet gösteren birçok şirket için gelecekle ilgili öngörülerde bulunmak önemli bir gereksinim olmuştur. Bu durum fon işleten ve elindeki portföye ait değerlendirme yapmak isteyen sigorta şirketleri için de geçerlidir. Şirket birleşmeleri, şirketin ilerideki kârının bugünkü değeri, portföye ait nakit akışları gibi gereksinimler için ileriye dönük tahminler yapmak gerekmektedir.

Öngörü için, kullanılan birçok istatistiksel analiz yöntemleri bulunmaktadır. Bunlar zaman serileri analizine dayalı yöntemlerdir. Son yıllarda bu alanda tercih edilen analiz yöntemlerinden biri de Yapay Sinir Ağları (Y.S.A.)'dır. Y.S.A.'lar, ağırlıklandırılmış şekilde birbirlerine bağlanmış birçok işlem biriminden (nöronlar) oluşan matematiksel modellerdir. Y.S.A., özellikle doğrusal olmayan yapıya sahip zaman serilerinin tahmininde kullanılmaktadır. Öğrenme hızı, az veriyle de orijinal sonuçlara yakın tahminlerde bulunabilmesi, kendi kendini eğitebilmesi gibi özelliklerinin yanında, değişken sayısının ve kullanılan parametrelerin sınırsız olması da diğer analiz yöntemlerine göre belirtilmesi gereken farklılıklardan birkaçıdır. Y.S.A.'lar, gerçek dünyaya ait ilişkileri tanıyabilir, sınıflandırma, kestirim ve işlev uydurma gibi görevleri yerine getirebilirler.

Hayat sigorta şirketi değerlemesi için gerekli olan belli başlı değişkenlerin ileriye dönük tahmin edilmesinde de Y.S.A.'lar tercih edilmiştir. Değerleme, banka kredileri nedeniyle üretilen ortalama poliçe süresi 3 yıl olan ihtiyaç kredileri için yapılacaktır. Bu doğrultuda TCMB USD ve EUR efektif satış kuru ile poliçe iptal oranları 1 - 2 ve 3 yıl için tahmin edilmiştir.

Çalışmamızın ilk bölümünde şirket değerlemesine neden ihtiyaç duyulduğu; ikinci bölümde ise sigorta ile ilgili kavramlar ve açıklamaları yer almıştır.

Y.S.A.'ların gelişimi, insan beyni yapısıyla olan benzerlikleri ve model içerisinde kullanılan terimler üçüncü bölümde açıklanmıştır. Y.S.A.'larının tarihsel gelişimi ve kullanım alanları verilmiştir.

Tahmin deęerleri ile orijinal deęerlerin karřılařtırılmasında kullanılan öğrenme algoritması incelenmiş ve uygulama olarak Çok Katmanlı Algılayıcı (Ç.K.A.) genel yapısı ve aę eğitimi ortaya konmuřtur. Yapılarına göre Y.S.A.'lardan, ileriye dönük Y.S.A.'ları; öğrenme algoritması olarak Levenberg-Marquardt öğrenme algoritması üzerinde detaylı açıklamalarda bulunulmuřtur. Ayrıca çalışmalarında tercih edilen dięer algoritmalar için de anlatımlar yer almaktadır.

Dördüncü bölümde ise uygulama için geçmişe dönük TCMB USD ve EUR günlük efektif satış kur deęerleri alınmış; üretilen ve iptal edilen poliçe sayıları ise iptal oranları için kullanılmıştır. Veriler, Matlab paket programı içerisinde yer alan ve hazır ara yüzlerden biri olan Neural Network araç paketi yardımıyla tahmin edilmiştir.

Sonuç olarak, elde edilen deęerler yorumlanmış, řirket deęerlemede kullanılacak ileriye dönük tahmin deęerleri belirlenmiştir. Çalışmanın bir sonraki adımı olarak, hayat ya da hayat dıřı sigorta řirketlerinin mevcut tüm portföy deęerlemeleri için gerekli parametre tahminlerinde, Y.S.A.'nın kullanabileceęi tavsiye edilmiştir.

## **BİRİNCİ BÖLÜM**

### **1. ŞİRKET DEĞERİNİN GELİŞİMİ, NEDEN İHTİYAÇ DUYULDUĞU ve DEĞERLEMEDE KULLANILAN ANALİZLER**

#### **1.1. Şirket Değerlemesinin Gelişimi**

Şirket değerlemesi, mutlak bir bilim dalı değildir. Şirket değerlemesinde, sosyo-ekonomik ve kültürel farklılıklar çoğu zaman önemli bir rol oynamaktadır. Dolayısıyla, kâr doğal olarak başarı ölçütü olarak kabul edilmiştir. Değerleme yöntemlerinin gelişmesi, sınai gelişme aşamaları ve ekonomik analiz ve bilgisayar işlem araçları arasında belirgin bir paralellik bulunmaktadır. Nakit akımlarına dayalı yöntemlerin önem kazanması, özellikle hızlı ekonomik büyüme, uluslararası rekabet ve izlenen ekonomi politikalarının sınai işletmeler arasında gruplaşmayı hızlandırdığı II. Dünya Savaşını izleyen dönemde söz konusu olmuştur.

Risk faktörünün giderek önem kazanması ise, daha istikrarsız ve belirsiz bir ekonomik döneme girildiği 1970'li yıllarda gerçekleşmiştir. Risk faktörünün sistemli bir şekilde analizlere dahil edilmesi, bilgisayar sistemlerinin dikkate değer bir gelişme göstermesiyle mümkün olmuştur.

#### **1.2. Şirket Değerlemesine Duyulan İhtiyaç**

Şirket değerlemesi, geleceğe yönelik stratejik planların oluşturulması ve bunların finansal sonuçları için yapılmaktadır. İhtiyaca göre şirket değerlemesi şirket satın alma, satma ya da birleşmelerinde, halka arz ya da başka nedenlerle hisse satışlarında kullanılabileceği gibi, hissedarlar için bir performans değerlendirme aracı olarak da kullanılabilir. Şirket değerlemesi amacına göre gelecek performansın bugün ifade ettiği değer ya da benzer şirketlerle karşılaştırılarak farklılıklar elde edilebileceği gibi, belli durumlarda sadece varlıkların ya da markaların değerlendirilmesi biçiminde de olabilmektedir.

### **1.3. Şirket Değerlemede Kullanılan Analizler**

Aksöyek (1994), şirket değerlemede; teknik analiz, temel analiz ve modern yaklaşım olmak üzere üç kuramsal model kullanıldığını belirtmiştir.

#### **1.3.1. Teknik Analiz**

Teknik analiz, fiyatların kendi içinde önceden izlenebilecek bir trende sahip olduğunu varsayar.

#### **1.3.2. Temel Analiz**

Bu analizde değerlemeye tabi tutulan şirketin gerçek değeri cari durumundaki ve gelecekteki ekonomik şartlar dikkate alınarak yapılan değer tespitine dayanmaktadır.

#### **1.3.3. Modern Yaklaşım (Rassal Yürüyüş Teorisi)**

Rassal yürüyüş teorisi ya da etkin pazar kuramı olarak da adlandırılan modern yaklaşımla belirlenir.

### **1.4. Genel Kabul Görmüş Şirket Değerleme Yöntemleri**

Bir firmanın değeri; o firmanın gelecekte sağlayacağı gelir akışı ile, bu gelir akışının elde edilmesindeki risk durumuna bağlıdır. Tüm finansal varlıklar, ekonominin ve işletmelerin tümünü etkisi altına alan sistematik risklerden aynı derecede etkilenir. Bu riskler faiz oranı riski, enflasyon riski ve piyasa riskidir. Akgüç (1994)'de belirtildiği gibi bu risklerden korunmak mümkün değildir. Finansal kararların odak noktasını firmanın değerini etkileyen karar ve faaliyetler oluşturmaktadır.

Finans yöneticisinden beklenen, firmanın değerini en yüksek düzeye ulaştıracak biçimde risk ve kârlılık arasında bir denge kurması, kârla risk arasında optimal bir değişim sağlamasıdır. Kâr ile risk arasında da dolaysız bir ilişki bulunmaktadır. Kârı yükseltmek için alınan kararlar genellikle riski de artırmakta; riski azaltan kararlar kârı da düşürmektedir.

## Şirket Değerleme Yöntemleri

### 1.4.1. Kayıtlı Değer

- **Temel PrensiP:** Bir şirketin değeri, sahip olduğu varlıklar ve yükümlülüklerden oluşur.
- **Değerleme Yaklaşımı:** Defter değeri/tasviye değeri/yenileme değeri/düzeltilmiş net aktif değeri olarak belirtilir.

### 1.4.2. Piyasa Değeri

- **Temel PrensiP:** Karşılaştırılabilir şirketlerin piyasalarda oluşmuş hisse senedi fiyatları ya da benzer işlerde ortaya çıkmış olan fiyatlar, bir şirketin değerinin göstergesidir.
- **Değerleme Yaklaşımı:** Bu yöntemle şirketin değeri, karşılaştırılabilir şirket çarpanları/karşılaştırılabilir işlem çarpanları ile belirlenmektedir.

### 1.4.3. Ekonomik Değer

- **Temel PrensiP:** Bir şirketin değeri ileriki yıllarda şirketin yaratacağı nakit akımlarının bugünkü değerinin toplamıdır. Bu nakit akımları, şirketin büyüme potansiyeli ve taşıdığı riskler (ülke, sektör ve şirket riski) gibi faktörlere bağlıdır.
- **Değerleme Yaklaşımı (Gelir Yaklaşımı) :** İndirgenmiş nakit akımları yöntemi (şirket değeri yaklaşımı, özsermaye değeri yaklaşımı)/indirgenmiş kâr payları yöntemi ile belirlenmektedir.

Çalışmada şirketin kredi karşılığı verilen poliçelere ait değerini belirlerken yatırım getirisi, iptal oranı, TCMB USD ve EUR efektif satış kur değerleri ve diğer varsayımlardan uygun olanlar ile duyarlılık analizleri yapılarak ekonomik değer yaklaşımı uygulanacaktır. Bu yaklaşımla ileriye doğru öngörü tahmininde, bir hayat sigortası portföyü için TCMB USD efektif satış kuru, TCMB EUR efektif satış kuru ve iptal oranına ait bilgiler Y.S.A. kullanılarak tahmin edilecektir.

Y.S.A. özellikle sınıflandırmada ve zaman serilerinin öngörülmesinde başarılı sonuçlar vermektedir. Y.S.A.'ların ekonomi ve finans alanlarında tahmin etme ve geleceği öngörme özellikleri, bu alanlarda tercih edilen bir modelleme yöntemi olmasını sağlamıştır.



## İKİNCİ BÖLÜM

### 2. SİGORTACILIKLA İLGİLİ TANIMLAMALAR

Sigorta kavramı hayatımızda karşılaştığımız, sık sık adını duyduğumuz bir kavramdır. Bu kavramın tanımı değişik biçimlerde yapılabilir. Mehr ve Cammack (1972)'de sigorta, fertlerin karşılaştıkları riskleri azaltmak ve riskin gerçekleşmesi halinde ortaya çıkan zararları belirginleştirmek ve telafi etmek gayesiyle örgütlenmiş toplumsal bir kurum olduğu belirtilmiştir. Başka bir tanımda ise sigorta, aynı türden tehlikeyle karşı karşıya olan kişilerin, belirli bir miktar para ödemesi yoluyla toplanan tutarın, sadece o tehlikenin gerçekleşmesi sonucu fiilen zarara uğrayanların zararını karşılamada kullanıldığı, bir risk transfer sistemidir. Bu sistem sayesinde kişiler, karşı karşıya buldukları tehlikelerin neden olabileceği, parayla ölçülebilen zararlarını, nispeten küçük miktarlarda ödemiş oldukları primler yoluyla paylaşmaktadırlar. Sigortanın temel işlevi, zararı ekonomik açıdan önemsiz bir duruma getirmektir. Kişiler tek başına karşılayamayacakları zararları bir organizasyon aracılığıyla aralarında paylaşmaktadırlar.

(<http://www.tsrbsb.org.tr/tsrbsb/Templates/innerpage>)

Sigortacı ise belirli bir prim karşılığında, sigortalıya ya da bir tazminatın yararlanacağı belirtilmiş olan kimseye, bir zarara uğraması durumunda belli bir para ya da gelir ödemeyi üstlenen kuruluştur.

Hayat Sigortaları Genel Şartları (1995)'de belirtildiği üzere, hayatı üzerine sigorta sözleşmesi yapılan kişiye "sigortalı", prim ödemek suretiyle sigortalının menfaatini sigortacı nezdinde sigortalayan kişiye "sigorta ettiren" ve sigorta sözleşmesine taraf olmamakla beraber lehine sigorta sözleşmesi yapılan ve rizikonun gerçekleşmesi halinde kural olarak sigorta tazminatını sigortacıdan isteme hakkına sahip olan tarafa da "lehdar" denir. Lehdar, sigorta sözleşmesine doğrudan taraf olabildiği gibi, sigorta ettiren konumundaki diğer bir kimsenin yapmış olduğu sözleşme ile de sigortadan doğrudan veya dolaylı olarak yararlanma konumunu elde edebilir. (<http://www.tsrbsb.org.tr/tsrbsb/Templates/innerpage>) Sigortalanan mal ya da cana ait belirleyici bilgileri, sigortanın başlangıç ve bitiş tarihlerini, ödenecek prim ve teminat tutarlarını vb. içeren sözleşmeye ait yazılı belge ise poliçe olarak isimlendirilir.

Sigortacılıkla ve şirket değerlemede kullanılan diğer terimler ise aşağıda belirtilmiştir.

### **2.1. Aktüeryal Bugünkü Değer**

Belirli bir olasılığa göre yapılacak bir miktar ödemenin, belirli bir iskonto faktörüyle hesaplama tarihine çekilmiş hali bir ödemenin aktüeryal bugünkü değeri olarak adlandırılır.

### **2.2. Teknik Faiz**

Teknik karşılıklar, primler ve kâr payı karşılıklarının garanti edilen kısmının hesaplanmasında kullanılan faiz oranıdır. Teknik faiz oranı; tarifenin teknik esaslarına ya da uygulanacak kâr payı teknik esaslarına, tarifenin endekslediği para birimine ya da varlığa, karşılıkların yatırıldığı varlıkların getirilerine, gelecekte yatırıma yönlendirilecek varlıkların beklenen getirilerine, ülkenin değişen ekonomik koşullarına ve şirketin mali bünyesine göre belirlenir.

Mortalite tablolarından yararlanılarak prim hesaplanmaktadır. Bu nedenle hayat sigortaları için tarife bilgileri doğrultusunda poliçe başlangıcında teknik faiz oranı belirlenmiş olacaktır. Değerleme süresince bu oran değişmeyecektir.

### **2.3. Mortalite Tablosu**

Bir ülkenin toplam nüfusunun gözlemlenmesi suretiyle elde edilen sonuçların, Hayat Sigortalarına uygulanmak üzere hazırlanan tablolarına denir. Hayat tabloları, aktüerya biliminin vazgeçilmez bir unsurudur. Bu tablolar istatistikî verilerden yararlanılarak meydana getirilir. Hayat tablolarında belirli bir süre içinde hayatta kalan ve ölen bireylerin sayılarından faydalanılarak zaman ve yaş aralığı bazında ölme ve yaşama olasılıkları ve bunlara bağlı olarak ölüm ve yaşam olasılıkları elde edilir. Hayat tablolarının bir adı da mortalite tablosudur. Yani mortalite tabloları ölüm gözlemlerine dayanarak, bir popülasyonda yaşayan insanların ölüm düzeylerini belirtir. Herhangi bir yaşta, bir yıl içerisinde kaç kişinin hayatta kalacağı ya da kaç kişinin öleceği bu tablolardan hareketle tahmin edilebilmektedir.

Mortalite tabloları, hayat sigortacılığındaki aktüeryal hesaplamaların ana kaynağını oluşturur. Bir aktüer, hayat sigortası tarifelerini hazırlarken içinde bulunulan ülkenin demografik yapısına en uygun mortalite tablosunu kullanmalıdır. Mevcut topluluğa uygun olan mortalite tablosunun kullanılması durumunda hem sigortacı hem de sigortalı zarar etme riskiyle karşı karşıyadır. Bu nedenle prim hesaplamasında kullanılan mortalite tablosu son derece önemlidir.

Mortalite tabloları ile ilgili aşağıdaki tanımlamalarda yapılabilir. Bunlar;

- Mortalite tabloları, büyük bir popülasyondaki gözlenen ölüm sayısının kayıtlarıdır. Eğer seçilen popülasyon bütün bir ülke ise buna mortalite tabloları denir.
- Mortalite tablosu, doğumları aynı zamana rastlayan bir grubun, yaşlar ilerlerken ölümler nedeniyle azalışını gösteren bir tablodur.
- Hayat tablosu, geçmiş gözlemlere ilişkin belirli büyüklükteki başlangıç grubundan, art arda gelen yaşlarda yaşayanların sayısını gösteren bir tablodur.
- Mortalite tablosu, bir ülkenin toplam nüfusunun gözlemlenmesi suretiyle elde edilen sonuçların, hayat sigortalarına uygulanmak üzere hazırlanan tablolardır.
- Hayat tablosu, tam yaş aralıklarına göre doğum ve yaşam sonu arasında ölme risklerini tanımlayan istatistiksel tablodur.

Mortalite tabloları, tablonun yapılış şekline ve yapıldığı yere göre isim almaktadır. Örneğin, Amerikan C.S.O. 1953-1958 Mortalite tablosu denildiği zaman, Amerika Birleşik Devletleri'nde 1953-1958 yılları arasında tam nüfusun gözlemlenmesi suretiyle elde edilen sonuçları gösteren tablo anlaşılmaktadır. Bunun yanısıra, bu tablolara "teknik faiz" yüklemesi yapılarak "Komütasyon Tabloları" elde edilmektedir.

Mortalite tabloları, tüm nüfus dikkate alınarak yapılabildiği gibi, cinsiyet ayrımı, belli bir bölgenin nüfusu gibi esaslara göre de yapılabilmektedir.

## **2.4. Prim**

Herhangi bir riske ilişkin olarak, sigortacının vermiş olduğu teminata karşılık olmak üzere, sigortalı ya da sigorta ettiren tarafından para olarak ödenen bedeldir. Sigorta sözleşmesinin en önemli unsurlarından birisidir ve sözleşmenin diğer bütün şartları yerine getirilmiş olsa dahi, primin ödenmemesi, birçok durumda sigorta sözleşmesinin yürürlüğe girmesini engelleyen bir durumdur.

Prim, risk primine ek olarak genel giderler, komisyonlar, sigorta şirketinin bu iş dolayısıyla katlanmak zorunda olduğu tüm maliyetler ve faaliyet kârını içermektedir.

Risk primi ise, eldeki istatistiklere dayanılarak hesaplanmış muhtemel hasar miktarı ve hasar masraflarını karşılamak üzere hesaplanmış net prim miktarıdır ve tehlike primi olarak da adlandırılabilir.

## **2.5. İştirâ**

İştirâ'nın sözlük anlamı satın alma olarak tanımlanmaktadır. Türk Ticaret Kanunu'nun 1327. maddesine göre, en az üç yıl prim ödenmesi halinde ve poliçenin iadesi kaydıyla "poliçede yazılı cetvele göre" iştirâ hakkı kullanılabilir. Poliçede yazılı cetvel, poliçenin ait olduğu tarife teknik esaslarıdır. Uygulamamızda uzun süreli hayat sigortası poliçelerine ait iştirâ oranı belirlenecektir. Bu orana aynı zamanda iptal oranı da denilmektedir.

## **2.6. Vade Gelimi**

Sigorta poliçesinin, başlangıçta belirlenmiş olan poliçe bitiş tarihine gelmesi durumu, vade gelimi olarak tanımlanır.

## **2.7. Aktüeryal Matematik Karşılık**

Şirketlerin üstlendikleri riskler için aldıkları risk primleri ile sigorta ettirenler ve lehdarlara olan yükümlülüklerinin peşin değerleri arasındaki farkı; birikim priminin de alındığı hayat sigortalarında ise tarife primlerinin birikime kalan kısımları ile teminatların bir yıldan uzun süreli verilmesi halinde tarifelerin onaylı teknik

esaslarında belirtilen formül ve esaslara göre ayrılan aktüeryal matematik karşılıklar toplamıdır.

## ÜÇÜNCÜ BÖLÜM

### 3. YAPAY SİNİR AĞLARI

#### Yapay Zekâ

Yapay zekâ kavramının geçmişi modern bilgisayar bilimi kadar eskidir. Fikir babası, "Makineler düşünebilir mi ?" sorusunu ortaya atarak Makine Zekâsını tartışmaya açan Alan Mathison Turing'dir. 1943'te İkinci Dünya Savaşı sırasında Kripto Analizi gereksinimleri ile üretilen elektromekanik cihazlar sayesinde bilgisayar bilimi ve yapay zekâ kavramları doğmuştur.

İnsanoğlu, mağara duvarlarına hayvan figürleri çizerek başladığı veri işleme sürecine artık insan gözüyle görülemeyecek küçüklükteki "chip"lerle devam etmektedir. Bu gelişim yaklaşık 4000 yıl gibi uzun bir süreye yayılmasına karşın gerçek gelişim son 50 yıl içinde yaşanmıştır. Günümüzde hayatımızın her alanına girmiş ve hayatımızın bir rutini haline dönüşmüş olan bu teknolojiyi insanoğlunun önceden tahmin etmesi bile olanaksızdı.

Örneğin; bilgisayar sektörünün önemli firmalarından IBM'in bir yöneticisi, "Bilgisayar ne kadar küçülürse küçülsün bir odadan daha küçük olamaz" demiştir. Sektörün önde gelen isimlerinden birinin böyle büyük bir yanığı içinde olması, bilgisayar teknolojisinin ne kadar hızlı geliştiğini çok güzel açıklamaktadır. Bu gelişim sürecinde artık kendi zekâsıyla yetinmeyen insanoğlu makinelere de zekâ verme uğraşı içindedir; artık amaç daha zeki, insan davranışlarını örnekleyebilen ve karar verebilen makineler üretmektir. Bu noktada karşımıza "Yapay Zekâ" kavramı çıkmaktadır.

İnsanlığın en büyük hayallerinden biri haline gelen, üzerine filmler çekilen, kitaplar yazılan yapay zekâ konusunu iyi anlayabilmek için felsefeden bilgisayar bilimine, elektrik-elektronikten biyoloji ve psikolojiye kadar çok geniş bir alanda çalışma yapmak gerekir.

Günümüzde bilgisayarlar, olaylar arasındaki ilişkileri öğrenebilmekte ve olaylar hakkında karar verebilmektedir. Elmas (2003)'de bilgisayarların bu özellikleri

kazanmasını ve yeteneklerinin geliştirilmesini sağlayan çalışmaların Yapay Zekâ çalışmaları olarak bilindiği belirtilmiştir.

Yapay zekâ ile ilgili çalışmalar insan bilinci kavramının keşfinin ilk adımlarıdır. Bu keşifle elde edilecek bulgular ekonomik yaşamımızda geniş uygulama alanları bulabilme potansiyeline sahiptir. Günümüzün bilgisayar teknolojisini önemli ölçüde yapay zekâ tasarımlarına ilişkin bilimsel çalışmalara borçluyuz.

Newell ve Simon, insan gibi düşünme yaklaşımına göre üretilmiş ilk program olan Genel Sorun Çözücü'yü geliştirmişlerdir. Simon, daha sonra fiziksel simge varsayımını ortaya atmış ve bu kuram, insandan bağımsız zeki sistemler yapma çalışmalarıyla uğraşanların hareket noktasını oluşturmuştur. Simon'ın bu tanımlaması bilim adamlarının Yapay Zekâ'ya yaklaşımlarında iki farklı akımın ortaya çıktığını belirginleştirmesi açısından önemlidir: Sembolik Yapay Zekâ ve Siberetik Yapay Zekâ (Ntv Bilim Dergisi,sayı:1,2009).

İnsanlar bilgiyi elde ederken sosyal çevrelerinden ve yaşam deneyimlerinden edindiği ve zamanla kişiliğinin derinliklerine kadar nüfuz eden yargılarının etkisi ile hareket etmektedir. İnsan zekâsı ile yapay zekâ kıyaslamalarının en ünlü olanı satranç müsabakalarıdır. Çabucak hamle yapılmasını gerektiren durumlarda satranç oynayan bilgisayarlar, insan satranççıya göre daha iyi performans göstermektedir. Çünkü bilgisayarların çalışma kuralları kesinliğe ve hızlı işlemeye dayalıdır. İnsan ise karar verme sürecini işletmek, yavaş ve bilinçli bir değerlendirme yapmak zorundadır. İnsan yargıları bu karar verme sürecini önemli ölçüde etkileyerek doğru karar olasılıklarını azaltırken, makine yargılardan bağımsız olarak olasılık hesaplarını hızla yapıp en etkili çözümü mutlaka bulmaktadır. Halaç vd., (2002), IBM tarafından tasarlanan Deep Blue isimli satranç oynayan bir bilgisayar programının 1991 yılında büyük satranç ustası Kasparov'u zor da olsa yenmesinin, bilgisayar ile insan zekâsı arasındaki kıyaslamaların ve rekabetin popüler hale gelmesini sağladığını belirtmiştir. Bu bakış açısıyla beynin üstün özellikleri, bilim adamlarını üzerinde çalışmaya zorlamış ve beynin nörofiziksel yapısından esinlenerek matematiksel modeli çıkarılmaya çalışılmıştır.

1970'li yıllarda büyük bilgisayar üreticileri olan Apple, Xerox, IBM gibi şirketler kişisel bilgisayar (PC-Personal Computer) modeli ile bilgisayarı popüler hale

getirdiler ve yaygınlaştırdılar. Ancak Yapay Zekâ çalışmaları, daha dar bir araştırma grubu tarafından geliştirilmeye devam etti.

Bugün, bu çalışmaları teşvik etmek amacı ile Alan Turing 'in adıyla anılan Turing Testi A.B.D 'de Loebner ödülleri adı altında Makine Zekâsına sahip yazılımların üzerinde uygulanarak başarılı olan yazılımlara ödüller dağıtılmaktadır. Testin içeriği kısaca şöyledir: Birbirini tanımayan birkaç insandan oluşan bir denek grubu birbirleri ile ve bir Yapay Zekâ diyalog sistemi ile geçerli bir süre sohbet etmektedirler. Birbirlerini yüz yüze görmeden yazışma yolu ile yapılan bu sohbet sonunda deneklere sorulan sorular ile hangi deneğin insan hangisinin Makine Zekâsı olduğunu saptamaları istenir. İlginçtir ki, şimdiye kadar yapılan testlerin bir kısmında Makine Zekâsı insan zannedilirken gerçek insanlar Makine zannedilmiştir.

Loebner Ödülünü kazanan Yapay Zekâ Diyalog sistemlerinin dünyadaki en bilinen örneklerinden biri A.L.I.C.E 'dir ve Carnegie Mellon Üniversitesi'nden Dr. Richard Wallace tarafından yazılmıştır. Bu ve benzeri yazılımlarının eleştiri toplamalarının nedeni, testin ölçtüğü kriterlerin konuşmaya dayalı olmasından dolayı programların ağırlıklı olarak diyalog sistemi olmalarıdır.

Yapay Zekâ teknikleri ise Yapay Sinir Ağları, Bulanık Mantık, Sezgisel Algoritmalar (Genetik Algoritmalar, Tabu Arama, Tavlama Benzetimi, Karınca Algoritması gibi) ve Uzman Sistemlerdir.

### **Yapay Sinir Ağları**

Haykin (1994), beynin bütün davranışlarını tam olarak modelleyebilmek için fiziksel bileşenlerinin doğru olarak modellenmesi gerektiği düşüncesi ile çeşitli yapay hücre ve ağ modelleri geliştirilmiştir. Böylece Y.S.A. denen yeni ve günümüz bilgisayarlarının algoritmik hesaplama yönteminden farklı, beynin fizyolojisinden yararlanılarak oluşturulan yapay bilgi işleme modelleri ortaya çıkmıştır. Y.S.A.'lar problemleri çözümede son yıllarda çok başarılı yaklaşımlar arasındadır (Çakar vd., 1996) Y.S.A., karmaşık yapıda içsel ilişkilere sahip problemlere çözüm bulabilmek için geliştirilmiş bir yapay zekâ tekniğidir.



Bilindiği gibi; öğrenme, hatırlama, düşünme gibi tüm insan davranışlarının temelinde sinir hücreleri bulunmaktadır. İnsan beyinde tahminen  $10^{11}$  adet sinir hücresi olduğu düşünülmektedir ve bu sinir hücreleri arasında sonsuz diyebileceğimiz sayıda sinaptik birleşme denilen sinirler arası bağ vardır. Ergezer vd. (2003), bu sayıdaki bir birleşimi gerçekleştirebilecek bir bilgisayar sisteminin dünya büyüklüğünde olması gerektiği söylenmektedir; ancak 50 yıl sonra bunun büyük bir yanılgı olmayacağını bu günden kimse söyleyemez. İnsan beyninin bu karmaşıklığı göz önüne alındığında, günümüz teknolojisinin 1,5 kilogramlık insan beynine oranla henüz çok geride olduğunu söylemek yanlış olmaz.

Y.S.A.; yapısı, bilgi işleme yöntemindeki farklılık ve uygulama alanları nedeniyle çeşitli bilim dallarının da kapsamına girmektedir. Y.S.A., beynin bir işlevi yerine getirme yöntemini modellemek için tasarlanan bir sistem olarak düşünüldüğünde, donanım olarak elektronik devrelerle ya da bilgisayarlarda yazılım olarak da hayata geçirilebilir.

Haykin (1999b)'da, Y.S.A., deneysel bilgiyi saklama ve kullanıma hazır hale getirme yeteneğine sahip basit işleme birimlerinden oluşan, çok yoğun, paralel ve dağılmış düzende çalışan bir işlemcidir. Bu işlemci iki yönden beyin ile benzerlik göstermektedir:

1. Bilgi ağ tarafından bir öğrenme süreciyle çevreden elde edilir.
2. Elde edilen bilgileri biriktirmek için sinaptik ağırlıklar olarak da bilinen nöronlar arası bağlantı güçleri kullanılır.

İnsan beyni; düşünme, var olan bilgi ve tecrübelerden sonuç çıkarma, karar verme gibi işlemleri yerine getirebilmesi ve bunları mili hatta nano saniyeler mertebesinde gerçekleştirme becerisine sahiptir. Beynin bilgi işleme yöntemine uygun olarak öğrenme süreci, arzu edilen amaca ulaşmak için Y.S.A. ağırlıklarının yenilenmesini sağlayan öğrenme algoritmalarını içerir.

### **3.1. Yapay Sinir Ağlarının Tarihsel Gelişimi**

Y.S.A.'nın tarihçesi nörobiyoloji konusuna insanların ilgi duyması ve elde ettikleri bilgileri bilgisayar bilimine uygulamaları ile başlamaktadır. Y.S.A. tarihini 1970 öncesi ve sonrası diye ikiye ayırmak gerekmektedir. 1970 yılında Y.S.A.

yaklaşımının tarihinde bir dönüm noktası başlamış ve o zamana kadar, kullanılamaz denen Y.S.A. yeniden doğmuştur.

Alan Turing, Nazi'lerin Enigma makinesinin şifre algoritmasını çözmeye çalışan matematikçilerin en ünlü olanlarından biriydi. İngiltere, Bletchley Park 'ta şifre çözme amacı ile başlatılan çalışmalar, Turing'in prensiplerini oluşturduğu bilgisayar prototipleri olan Heath Robinson, Bombe ve Colossus bilgisayarları, Boole cebirine dayanan veri işleme mantığı ile Makine Zekâsı kavramının oluşmasına sebep olmuştu. Modern bilgisayarın atası olan bu makineler ve programlama mantıkları aslında insan zekâsından ilham almışlardı. Ancak sonraları, modern bilgisayarlar daha çok uzman sistemler olarak adlandırılan programlar ile gündelik hayatımızın sorunlarını çözmeye yönelik kullanım alanlarında daha yaygınlaştılar.

Yine aynı yıllarda (1943'te) nörofizyolog McCulloch ve Walter Pitts ilk sinir ağı modelini geliştirerek birkaç ara bağlantı kurmaya çalıştılar. Bu araştırmacıların önerdiği, yapay nöronları kullanan hesaplama modeli, önermeler mantığı, fizyoloji ve Turing'in hesaplama kuramına dayanıyordu. Herhangi bir hesaplanabilir fonksiyonun sinir hücrelerinden oluşan ağlarla hesaplanabileceğini ve mantıksal ve/veya işlemlerinin gerçekleştirilebileceğini gösterdiler. Bu ağ yapılarının uygun şekilde tanımlanmaları halinde öğrenme becerisi kazanabileceğini de ileri sürdüler. Hebb, 1949 yılında sinir hücreleri arasındaki bağlantıların şiddetlerini değiştirmek için basit bir kural önerince, öğrenebilen Y.S.A. gerçekleştirmek de olası hale gelmiştir.

Yazıcı vd., (2007), 1950'li yıllarda hızlı bir gelişim gösteren bilgisayar teknolojisi üzerine çalışan uzmanların, sinirbilimcilerle temasa geçerek bu konuda çalışmalar yaptıklarını belirtmişlerdir.

Aynı yıllarda Shannon ve Turing bilgisayarlar için satranç programları yazıyorlardı. İlk yapay sinir ağı temelli bilgisayar SNARC, MIT'de Minsky ve Edmonds tarafından 1951'de yapıldı. Çalışmalarını Princeton Üniversitesi'nde sürdüren Mc Carthy, Minsky, Shannon ve Rochester'le birlikte 1956 yılında Dartmouth'da iki aylık bir açık çalışma düzenledi. Bu toplantıda birçok çalışmanın temelleri atılmakla birlikte, toplantının en önemli özelliği Mc Carthy tarafından önerilen

“Yapay Zekâ” adının konmasıdır. Kuram ispatlayan ilk programlardan Logic Theorist (Mantık kuramcısı) burada Newell ve Simon tarafından tanıtılmıştır.

1960'ların sonlarına doğru Y.S.A. çalışmaları durma noktasına gelmiştir. Buna en önemli etki; Minsky ve Pappert tarafından yazılan Algılayıcılar (Perceptrons) adlı kitaptır. Bu kitapta Y.S.A.'ların doğrusal olmayan problemleri çözemediğini meşhur XOR problemi ile ispatlaması; çalışmaların verimsiz olduğunu söylemesi, araştırmacıları uzun yıllar maddi kaynak bulmakta zorlanmış ve bu konudaki çalışmaları durma noktasına getirmiştir.

Ancak sınırlı imkanlara rağmen hala bu konuda çalışan az sayıda bilim adamının gayreti ile düzenlenen konferanslar ve bazı endüstriyel ve finansal kuruluşlarda başarılı ticari uygulamalar sayesinde 1980'lerin başlarından itibaren Y.S.A. çalışmaları yeniden ivme kazanmıştır (Papik et al. 1998).

Tüm bunlara rağmen Anderson, Amari, Cooper, Fukushima, Grossberg, Kohonen, Hopfield gibi bilim adamları çalışmalarını sürdürmüşler; 1972'de Kohonen ve Anderson çağrışımlı bellek konusunda benzer çalışmalar yayınlamışlardır.

1970'lerin sonlarına doğru Fukushima, NEOCOGNITRON modelini tanıtmıştır. Bu model şekil ve örüntü tanıma amaçlı geliştirilmiştir.

Kohonen daha sonra 1982 yılında kendi kendine öğrenme nitelik haritaları (Özörgütlemeli Haritalar) konusundaki çalışmasını yayınlamıştır. 1960'ların sonlarına doğru Grosberg, Carpenter ile birlikte Uyarlanabilir Rezonans Teorisi'ni geliştirmiştir.

1982-1984 yıllarında Hopfield tarafından yayınlanan çalışmalar ile Y.S.A.'ların geliştirilebileceğini ve gezgin satıcı problemi gibi çözümü zor problemlere çözüm üretebileceğini göstermiştir. Bu çalışmaların neticesi Hinton ve arkadaşları'nın geliştirdiği Boltzman Makinası'nın doğmasına yol açmıştır.

Öztemel (2003), 1988 yılında Broomhead ve Lowe'un radyal tabanlı fonksiyonlar modelini (R.T.F.) geliştirdiklerini ve özellikle filtreleme konusunda başarılı sonuçlar elde ettikleri belirtmiştir. Daha sonra Spect, bu ağların daha gelişmiş şekli olan Olasılıksal Sinir Ağları ve Genel Regresyon Sinir Ağları'nı geliştirmiştir.

Y.S.A.'nın tarihsel gelişimi incelendiğinde, günümüze kadar sayısız çalışma ve uygulamadan geçtiği görülmüştür. Aşağıda, yapılan çalışmalardan bazıları verilmiştir.

- 1890 - insan beyninin yapısı ve fonksiyonları ile ilgili ilk yayının yazılması,
- 1911 - insan beyninin sinir hücrelerinden oluştuğu fikrinin benimsenmesi,
- 1943 - yapay nöronlara dayanan hesaplama teorisinin ortaya atılması ve eşik değerli mantıksal devrelerin geliştirilmesi,
- 1949 - öğrenme prosedürünün bilgisayarlar tarafından gerçekleştirilecek şekilde geliştirilmesi,
- 1956 - 1962 Adaline ve Widrow-Hoff öğrenme algoritmasının geliştirilmesi,
- 1957 - 1962 Algılayıcının geliştirilmesi,
- 1965 - ilk makina öğrenmesi kitabının yayınlanması,
- 1967 - 1969 bazı gelişmiş öğrenme algoritmalarının geliştirilmesi (Grasberg vs.),
- 1969 - tek katmanlı algılayıcıların yetersizliklerinin ispatlanması,
- 1969 - 1972 doğrusal ilişkilendiricilerin geliştirilmesi,
- 1972 - korelasyon matris belleğinin geliştirilmesi,
- 1974 - geri yayılım modelinin geliştirilmesi,
- 1978 – Uyarlanabilir Rezonans Teorisi modelinin geliştirilmesi,
- 1982 yılında bir fizikçi olan Hopfield'in, Y.S.A.'nın genelleştirilebilceğini ve özellikle geleneksel bilgisayar programlama ile çözülmesi zor olan problemlere bile çözüm üretebileceğini göstermesi (Öztemel, 2003).
- 1982-1984 yıllarında Kohonen kendi adıyla anılan eğitimcisz öğrenen ağları geliştirilmesi (Hooman ve Zilouchian, 2001).

- 1984 - Boltzman Makinası'nın geliştirilmesi,
- 1988 - RBF modelinin geliştirilmesi,
- 1991 – Genel Regresyon Sinir Ağı modelinin geliştirilmesi.
- 2004 yılında Virjinya Eyalet Üniversitesi'nden Zhicun Wang'ın, 3 boyutlu konfigürasyonlarda aerodinamik güçlerin kontrolünü Y.S.A.'ları kullanılarak simüle etmiştir.
- 2006 - Jorgensen bir uçağın rüzgar tüneli verilerinden seçilen aerodinamik karakterleri öğrenerek “doğrusal geri besleme kontrolü”nde kullanılmasını sağlayacak yapay sinir ağı programı üzerinde çalışmasıdır.

### **Yapay Sinir Ağları ile Biyolojik Sinir Ağları Arasındaki Benzerlikler**

Y.S.A. biyolojik sinir ağlarının taklidi olduğu için, Y.S.A. çalışmasını anlayabilmek için öncelikle biyolojik sinir sisteminin yapısına bakmak gerekir. Bu nedenle, öncelikle biyolojik sinir ağlarının yapısı kısaca anlatılacaktır.

### **Biyolojik Sinir Ağlarının Yapısı**

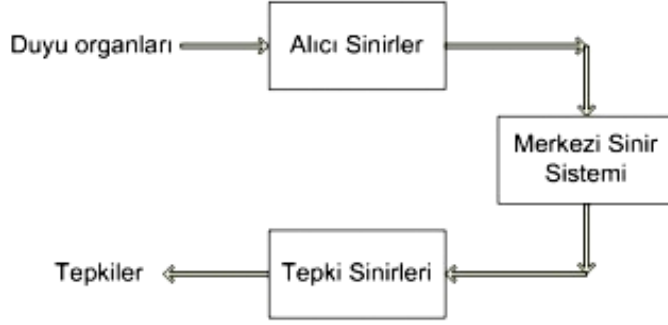
İnsan beyninin nasıl çalıştığı henüz tam olarak anlaşılabilmiş değildir. Ancak bu konuda beyni oluşturan milyarlarca sinir hücresinin önemli rol oynadığı bilinmektedir. Beynin çalışması genel olarak 3 aşamadan oluşur:

1. Bilgi girişi,
2. Sentezleme ve karşılaştırma,
3. Bilgi çıkışı ve eylem.

Beynin bu işlevlerini yerine getirebilmesini sağlayan temel yapı elemanının, sinir hücreleri yani nöronlardır Yazıcı vd., (2007).

Bir insanın beyinde yaklaşık olarak 10 milyar sinir hücresi ve bu nöronların birbirleriyle yaptığı bağlantı sayısının ise 60 trilyon olduğu tahmin edilmektedir. Bu sinirler girdi bilgilerini duyu organlarından alırlar. Daha sonra alıcı (taşıyıcı) sinirler bu sinyalleri işleyip bir sonraki sinire aktararak sinyalin merkezi sinir sistemine

kadar ulaşmasını sağlar. Merkezi sinir sistemi bu sinyalleri alıp yorumladıktan sonra tepki sinyallerini üretir. Bu sinyaller de tepkilerin oluşacağı organlara tepki sinirleri vasıtasıyla iletilir. Bu sayede duyu organlarından gelen bilgilere karşı, tepki organlarına uygun işaretler sinir sistemi vasıtasıyla yollar. Şekil 3.1.'de merkezi sinir sisteminin yapısı gösterilmiştir.



Şekil 3.1. Merkezi sinir sistemi yapısı

Biyolojik sinir sisteminin yapı taşı olan sinir hücreleri nöronlar, yapay sinir ağlarının da yapı taşıdır. Bir motor sinir hücresinin yapısı Şekil 3.2.'de gösterilmiştir.



Şekil 3.2. Motor sinir hücre yapısı

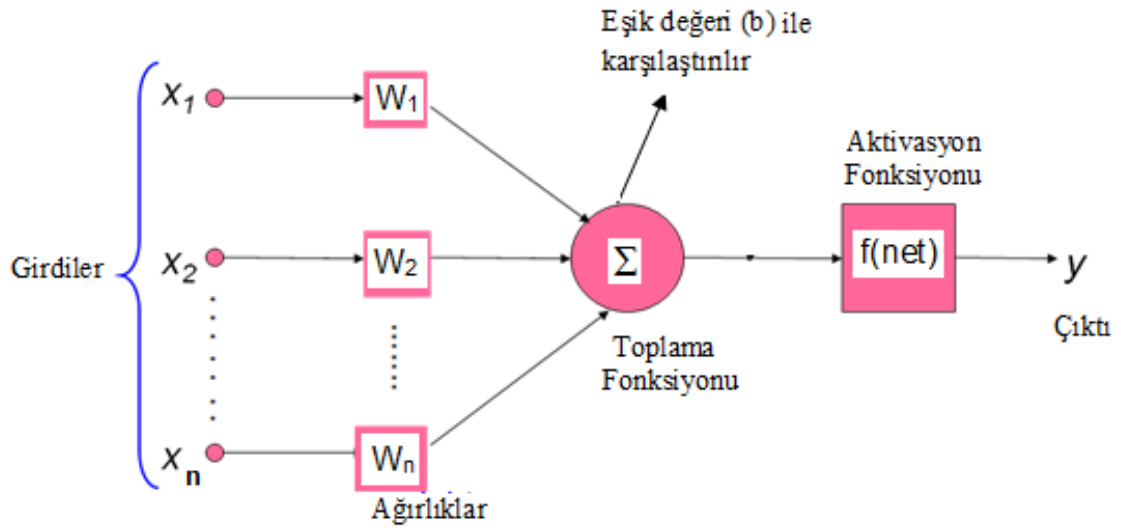
Motor sinir hücresinde, dendritler diğer hücrelerden girdileri alır. Daha sonra girdiler birleştirilir. Birleştirilen girdilerden genelde doğrusal olmayan bir şekilde sonuç üretilir. Bu sonuç çıktı olarak gönderilir.

### Yapay Nöronların Yapısı

Yapay nöronlar da biyolojik sinir hücrelerine benzer yapıdadır. Aynı biyolojik nöronlarda olduğu gibi, yapay nöronların da girdi sinyallerini aldıkları, bu sinyalleri

toplayıp işledikleri ve çıktılarını ilettikleri bölümleri bulunmaktadır. Y.S.A., kendisini oluşturan yapay nöronların birbirleriyle çeşitli katmanlar içerisinde paralel bağlantılar kurarak bir araya gelmelerinden oluşmaktadır. Yapay nöronlar, biyolojik sinir sisteminin doğal yapısının matematiksel olarak modellenmesiyle oluşturulmaya çalışılmaktadır.

Yapay nöronun yapısı Şekil 3.3.'te verilmiştir.



Şekil 3.3. Yapay nöron yapısı

Y.S.A.'nın istatistik terminolojisinde karşılık gelen terimleri ise Çizelge 3.1'de gösterilmiştir.

Çizelge 3.1: İstatistik terminolojisinde kullanılan terimlerin Y.S.A. terminolojisindeki karşılıkları (Günay vd., 2007a).

İSTATİSTİK TERMİNOLOJİSİ	Y.S.A. TERMİNOLOJİSİ
Model	Yapay Sinir Ağı
Parametre	Ağırlık
Bağımsız Değişken	Girdi
Tahmin Değeri	Çıktı
Bağımlı Değişken	Hedef
Artık	Hata

Çizelge 3.1. devam ediyor.

İSTATİSTİK TERMİNOLOJİSİ	Y.S.A. TERMİNOLOJİSİ
Güven Aralığı	Hata Çizgisi
Temel Fonksiyon	Nöron
Bağımsız Değişkenler Kümesi	Girdi Katmanı
Temel Fonksiyonlar Kümesi	Gizli Katman
Tahmin Değerleri Kümesi	Çıktı Katmanı
Gözlem	Örüntü
Kestirim ya da Optimizasyon	Eğitim, Öğrenme ya da Adaptasyon
Örneklem Adaptasyonu	Çevrimiçi Öğrenme
Grup Adaptasyonu	Çevrimdışı Öğrenme
Kestirim Ölçütü	Hata, Maliyet ya da Lyapunov Fonksiyonu
Dönüşüm	Fonksiyonel Bağlantı
Diskriminant Analizi	Sınıflama
Regresyon	Eşleme, Yaklaşım ya da Denetimli Öğrenme
Veri İndirgeme	Denetimsiz Öğrenme ya da Şifreleme
Genelleştirme	İnterpolasyon ya da Ekstrapolasyon

Yapay nöron aşağıdaki birimlerden oluşmaktadır.

- **Girdiler** :  $(x_1, x_2, x_3, \dots, x_n)$ , diğer hücrelerden ya da dış ortamlardan hücreye giren bilgilerdir. Bunlar ağın öğrenmesi istenen örnekler tarafından belirlenir.
- **Ağırlıklar** :  $(w_1, w_2, w_3, \dots, w_n)$ , hücreler arasındaki bağlantıların sayısal değeridir. Bir hücreye gelen bilginin değerini ve hücre üzerindeki etkisini gösterilirler.
- **Toplama (Birleştirme) fonksiyonu** : Hücreye gelen net girdinin hesaplanmasını sağlayan fonksiyondur. En yaygın kullanım şekli her girdi değerinin kendi ağırlığıyla çarpılarak toplanmasıdır.



- **Aktivasyon fonksiyonu** : Bu fonksiyon hücreye gelen net girdinin işlenmesiyle hücrenin bu girdiye karşılık üretileceği çıktıyı belirlemesini sağlar. Diğer bir anlatımla girdi kümesi ya da kendinden önceki bir katmandaki başka bir işlem elemanının bu işlem elemanı üzerindeki etkisini ifade eden değerlerdir.
- **Çıktılar** :  $(y_1, y_2, y_3, \dots, y_n)$  olarak gösterilen çıktı değerleri, ara katmanlardan gelen bilgileri işleyerek dış dünyaya gönderirler.

Efe ve Kaynak (2000)'de Y.S.A.'nın çalışma süreci şu şekilde de tanımlanır. Girdi katmanından alınan girdiler, girdi katmanı ve gizli katman arasında bulunan bağlantı ağırlıkları ile çarpılıp gizli katmana iletilmektedir. Gizli katmandaki nöronlara gelen girdiler toplanarak aynı şekilde gizli katman ile çıktı katmanı arasındaki bağlantı ağırlıkları ile çarpılarak çıktı katmanına iletilir. Çıktı katmanındaki nöronlar da, kendisine gelen bu girdileri toplayarak uygun bir çıktı üretirler.

### **Y.S.A.'nın Genel Özellikleri**

Y.S.A.'nın bazı genel özellikleri aşağıda sıralanmıştır:

1. Y.S.A. kabul edilebilir sonuçlar üretir. Optimum çözümü garanti etmez.
2. Ağ için nöron sayıları, katmanların sayısı, aktivasyon fonksiyonu, öğrenme katsayısı vb. belirlemede genel bir kural olmaması dezavantaj sayılabilir. Kullanıcının tecrübesine bağlıdır. Bu nedenle uygun ağ yapısının belirlenmesinde genellikle deneme-yanılma neticesinde karar söz konusudur. Deneme-yanılma kullanıldığında çoğu zaman daha iyi sonuç alınır.
3. Daha önce belirtildiği gibi ağın eğitiminde örnekler çok önemlidir. Ağın performansının düşmesi çoğu zaman örneklerin doğru sunulmamasından kaynaklanmaktadır. Kullanıcı bilgi eksikliği, uygun ağ yapısının sağlanamaması gibi nedenler çözülebilecek bir problemde, çözümsüzlüğe yol açabilir. Bu durum alternatif yaklaşımlar için de geçerlidir.

4. Eđitimin ne zaman bitirileceđi konusunda kesin bir sınır yoktur. Hata, istenen belli bir deđere ulařınca, eđitimin tamamlanması yeterli grldđ iin optimum đrenmenin gerekleřtiđi sylenemez. Ayrıca eđitimin ge sonlandırılması ađın ezberleme durumunu ortaya ıkardıđı iin, rnekleri ezberleyen ađ, yeni rneklerle karřılařtıđında uygun olmayan sonular retebilir.
5. Ađın davranıřları aıklanamaz. Bu sebeple bazı durumlarda ađların verdiđi sonuları deđerlendirmek zor olabilir. Ađ bir kara kutu gibidir.
6. Sinir ađlarının eđitilmesi ve test edilebilmesi iin ok fazla veri kmesine ihtiya duyulmamaktadır.
7. Basit olarak grlebilecek modelleme yapılarına rađmen zaman zaman uygulaması zor ve karmařık olabilmektedir.
8. Bazı u durumlarda eđitilmek iin uzun bir zamana ihtiya duyabilirler ve bundan dolayı zaman maliyeti ykselebilir.
9. đrenme yeteneđi sayesinde bilinen rnekleri kullanarak daha nce karřılařılmamıř durumlarda genelleme yapabilmektedir. Yani hatalı ya da kayıp veriler iin zm retebilmektedir. Eksik bilgileri de iřleyerek sonuca varabilir.
10. Sınırsız sayıda deđiřken ve parametre ile kolaylıkla alıřılabilmektedir.
11. Y.S.A., dođrusal olmayan, ok boyutlu ve zellikle problemin zmnde kesin bir matematiksel modelin ya da algoritmanın bulunmadıđı verilerin analizinde nemli bir aratır.
12. Y.S.A., ok sayıda hcrenin eřitli řekillerde bađlanmasından oluřtuđundan, paralel dađılmıř bir yapıya sahiptir ve ađın sahip olduđu bilgi, ađdaki btn bađlantılar zerine dađılmıř durumdadır. Bu nedenle, eđitilmiř Y.S.A.'nın bazı bađlantılarının hatta bazı hcrelerinin etkisiz hale gelmesi, ađın dođru bilgi retmesini nemli lde etkilemez. Bu nedenle hatayı tolere etme yetenekleri son derece yksektir.
13. Deđiřken uzayı karmařık ve verinin dađılımını bilinen istatistiksel dađılımlardan farklı Y.S.A., istatistiksel sınıflama yntemlerinden daha gvenilir sonular verirler.

15. Ağ ağırlıkları ve yapısı değiştiğinde somut bir problemin çözümü için kendini uyarlayabilme özelliğine sahiptir (Erdoğan, 2006; Yazıcı vd., 2007).

### **Y.S.A. İçin Kullanılan Yazılım Kütüphaneleri ve Hazır Arayüzler**

İnternet üzerinde Y.S.A. için bir çok yazılım kütüphanesi ve simülasyon ortamı bulunabilmektedir. Bunların bazıları sadece kütüphane kısmından oluşurken, bazıları Y.S.A. tasarlamayı sağlayan araçları da içerir.

Belli başlı Y.S.A. ara yüzleri aşağıda belirtilmektedir:

#### **Yazılım Kütüphanesi**

1. **Joone** : Açılımı Java Object Oriented Neural Engine olan kütüphane Java ortamında Y.S.A.'nın kullanılması için geliştirilmiştir. Ayrıca görsel olarak Y.S.A. oluşturulması ve eğitilmesini sağlayan bir araca sahiptir.

2. **Neural Network Toolbox for MATLAB**: MATLAB için yapay sinir ağı aracıdır.

3. **Annie** : Açık kaynak kodlu bir kütüphane olup C++ dili için geliştirilmiştir. Ç.K.A., RBF ve Hopfield modellerini desteklemektedir.

4. **FANN** : ANSI C ile gerçekleştirilmiş bir yapay sinir ağı kütüphanesidir. Benzerlerinin çoğundan daha hızlı çalışmaktadır.

#### **Simülasyon**

1. **Genesis** : Karmaşık sinir ağı simülasyon ortamıdır.

2. **Stuttgart Neural Network Simulator** : C dilinde yazılmış ve X tabanlı grafik arabirimi olan Y.S.A. uygulama simülasyon ortamıdır.

#### **Yazılım ve eklentiler**

1. **NeuroXL** : MS Excel için Y.S.A. eklentisidir. Tahminleme, sınıflandırma ve özellikle finansal tahminleme için geliştirilmiştir.

2. **NeuroSolutions** : Bir çok sinir ağı ve öğrenme algoritmasını destekleyen ticari bir yazılımdır.

**3. NeuralWorks Professional II/PLUS** : Kapsamlı sinir ağı geliştirme ortamı sunan ticari bir yazılımdır.

Çalışmada MATLAB içerisinde yer alan hazır ara yüzlerden biri olan Neural Network Araç Kiti'nden yararlanılmıştır.

### 3.2. Yapay Sinir Ağları Süreçleri

Hosein (2002), Y.S.A.'ların iki temel süreci içerdiğini belirtmiştir. Bunlar işlem ve eğitim süreçleridir.

#### İşlem Süreci

Toplama (birleştirme) ve aktivasyon olarak adlandırılan iki fonksiyondan oluşmaktadır.

#### Toplama Fonksiyonu

Toplama fonksiyonu, yapay nöronun ağırlıklarla çarpılması sonucu gelen girdileri toplayıp o hücrenin net girdisini hesaplayan bir fonksiyondur. Bu fonksiyon, adından da anlaşılacağı gibi, genelde toplama işlemini kullanmakla birlikte farklı işlemleri de kullanabilir. Hatta araştırmacının kendi kurduğu işlemi de kullanması mümkündür. Toplama fonksiyonunda kullanılan işlem, genellikle seçilen ağ mimarisine de bağlıdır.

Y.S.A. girilen girdi değerlerine toplama fonksiyonları uygulandıktan sonra, her bir nöron için çıktı değeri Eş. 3.1.'deki gibi bulunur.

$$\text{net} = \sum_{i=1}^n \sum_{j=1}^m w_{ij} x_i + b \quad (3.1.)$$

Burada  $x_i$  i'inci girdi,  $w_{ij}$  j'inci elemandan i'inci elemana bağlantı ağırlığını ve b yan değerini göstermektedir. Daha sonra elde edilen bu çıktı değerleri Eş. 3.2.'de belirtildiği gibi doğrusal olmayan türevlenebilir bir aktivasyon fonksiyonundan geçirilerek çıktı hesaplanır;

$$y = f(\text{net}) = f\left\{\sum_{i=1}^n \sum_{j=1}^m w_{ij} x_i + b\right\} \quad (3.2.)$$

$f(\text{net})$  : aktivasyon fonksiyonu,

$w_{ij}$  : i. girdi katmanı ile j. çıktı katmanı arasındaki ağırlık değeri,

$x_i$  : i. girdi değeri,

$b$  : yan değeridir.

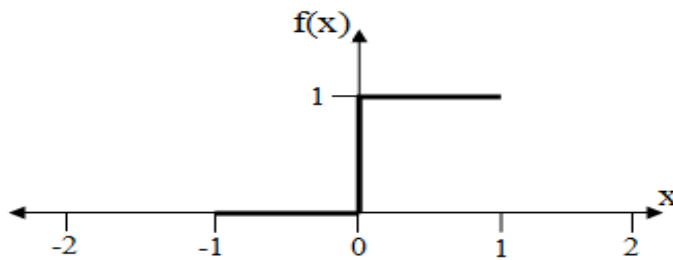
Nöronun arzu edilen çıktı değerini verebilmesi için  $w_{ij}$  ağırlık ve  $b$  yan değerlerinin optimum ayarlanması gerekmektedir.

### **Aktivasyon Fonksiyonları**

Yapay nöronun davranışını belirleyen önemli etkenlerden biri de aktivasyon fonksiyonudur. Mandic ve Chambers (2001), aktivasyon fonksiyonu hücreye gelen net girdiyi, diğer bir deyişle toplama fonksiyonunun ürettiği değeri işleyerek bu hücreye gelen girdilere karşılık olan çıktıyı belirlediğini belirtmiştir. Yani yapay nöron çıktısının büyüklüğünü sınırlandıran fonksiyondur. Bazı kaynaklarda transfer, eşik ya da sıkıştırma fonksiyonu olarak da isimlendirilmektedir. Aktivasyon fonksiyonu da Y.S.A.'ların farklı modelleri için farklı olabilir. En uygun aktivasyon fonksiyonunu belirlemek için geliştirilmiş bir yöntem yoktur. Hangi problemlerde hangi fonksiyonun kullanılacağı, Y.S.A.'nın çözmeye çalışacağı problemin şartlarına bağlıdır ve etkin bir şekilde deneme yanılma yoluyla tespit edilebilir. En çok kullanılan aktivasyon fonksiyonları ise aşağıda belirtilmiştir.

#### **- Adımsal Aktivasyon Fonksiyonu**

Bu aktivasyon fonksiyonları, mantıksal çıktı verir ve sınıflandırıcı ağlarda tercih edilir. Çift kutuplu ve tek kutuplu olarak ikiye ayrılır. Eş. 3.3.'te tek kutuplu adımsal aktivasyon fonksiyonunun matematiksel gösterimi ve Şekil 3.4.'te grafiksel gösterimi yer almaktadır.



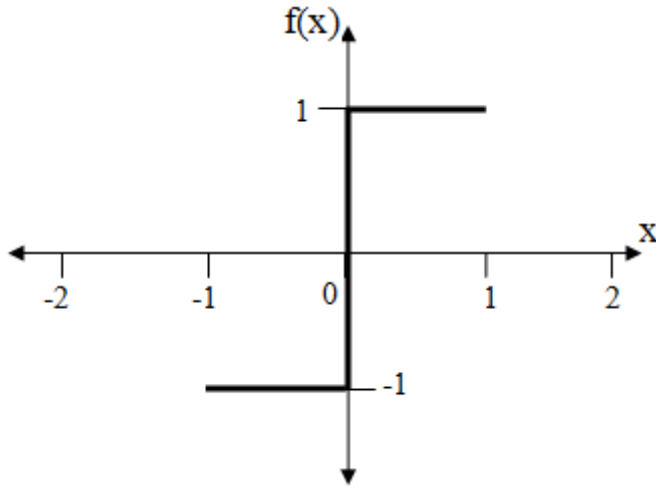
Şekil 3.4. Tek kutuplu adımsal aktivasyon fonksiyonu

$$f(x) = \begin{cases} 0, & 0 > x \\ 1, & x \geq 0 \end{cases} \quad (3.3.)$$

$f(x)$  : tek kutuplu aktivasyon fonksiyonu,

$x$  : girdi değeridir.

Çift kutuplu aktivasyon fonksiyonuna ait fonksiyon denklemini Eş. 3.4.'te; grafiksel gösterimi ise Şekil 3.5.'te bulunmaktadır.

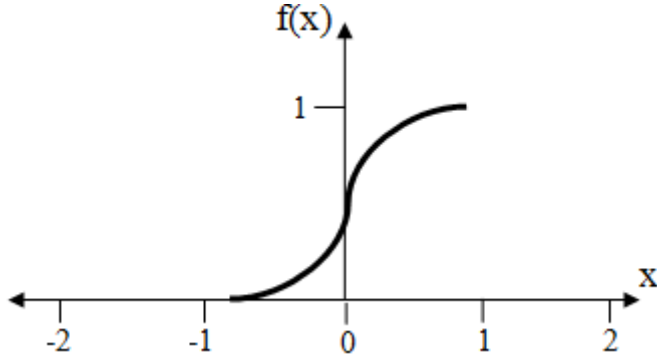


Şekil 3.5. Çift kutuplu adımsal aktivasyon fonksiyonu

$$f(x) = \begin{cases} 0, & 0 > x \\ 1, & x \geq 0 \\ -1, & x < 0 \end{cases} \quad (3.4.)$$

#### - Sigmoid Aktivasyon Fonksiyonu

Sürekli ve doğrusal olmayan bir fonksiyon olması nedeniyle doğrusal olmayan problemlerin çözümünde kullanılan Y.S.A.'larda tercih edilir. Girdi değerleri hangi aralıkta olursa olsun, çıktı 0 ile 1 arasında olmaktadır. Türevlenebilir olduğu için geri yayılım algoritmaları ile kullanılabilir. Sürekli ve her bir değer birbirinden farklıdır. Asimptotik olarak doygun noktalara yaklaşır. Sigmoid aktivasyon fonksiyonuna ait grafiksel gösterim Şekil 3.6.'da ve ilgili fonksiyon ise Eş. 3.5.'te belirtilmiştir.



Şekil 3.6. Sigmoid aktivasyon fonksiyonu

$$f(x) = \frac{1}{1+e^{-\lambda \text{net}}} \quad (3.5.)$$

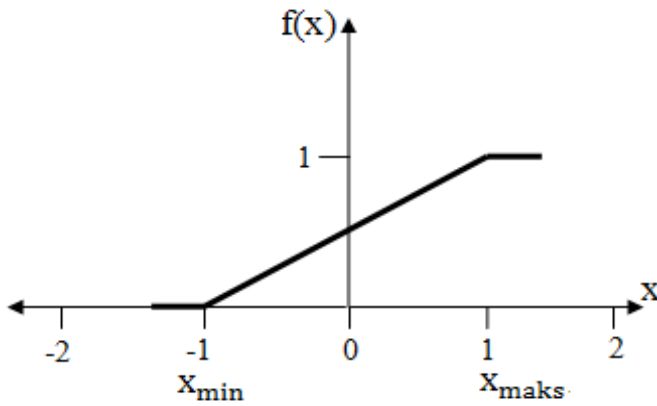
$f(x)$  : sigmoid aktivasyon fonksiyonu,

$\lambda$  : eğim parametresi,

net : aktivasyon değeridir.

#### - Parçalı Doğrusal Aktivasyon Fonksiyonu

Parçalı doğrusal aktivasyon fonksiyonu doğrusaldır ve net aktivasyon değerini minimum girdi değerinden küçük değere sahip girdiler için sıfıra; büyük değerler için bire eşitler. Parçalı doğrusal aktivasyon fonksiyonunun Eş. 3.6.'da matematiksel gösterimi, Şekil 3.7.'de ise grafiği görülmektedir.



Şekil 3.7. Parçalı doğrusal aktivasyon fonksiyonu

$$f(x) = \begin{cases} 0, & x \leq x_{\min} \\ mx + b, & x_{\max} > x > x_{\min} \\ 1, & x \geq x_{\max} \end{cases} \quad (3.6.)$$

$x_{\min}$  : veri kümesindeki minimum değere sahip değer,

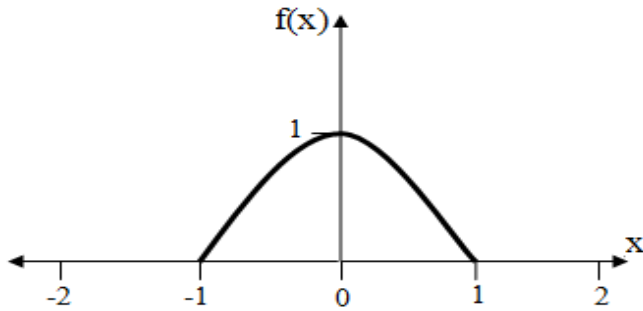
$x_{\max}$  : veri kümesindeki maksimum değere sahip değer,

$x$  : girdi değeri,

$b$  : yan değeridir.

### - Gaussian Aktivasyon Fonksiyonu

Şekil 3.8'de görüldüğü gibi çan şeklindedir.  $|net|$  büyük olduğunda  $f(net)$  asimptotik olarak 0 ya da bazı sabit değerlere yakınsar. Eş. 3.7'de gaussian aktivasyon fonksiyonuna ait eşitlik yer alırken, Şekil 3.8.'de grafiksel gösterimi verilmiştir.



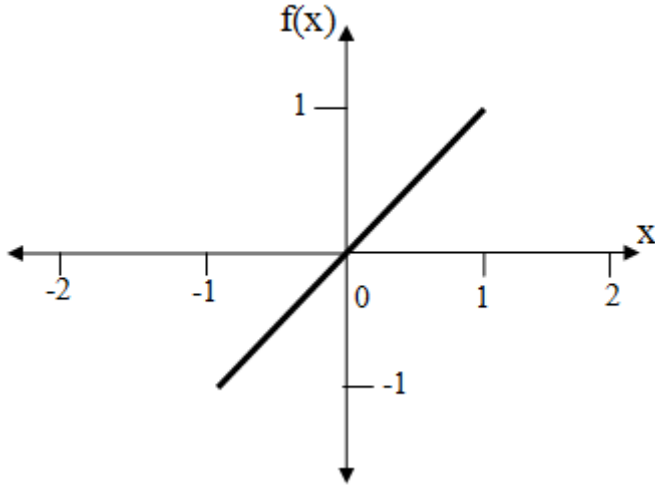
Şekil 3.8. Gaussian aktivasyon fonksiyonu

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (3.7.)$$

### - Doğrusal Aktivasyon Fonksiyonu

Hücrenin net girdisini doğrudan hücre çıktısı olarak veren ve doğrusal bir problemi çözmek amacıyla kullanılan bir aktivasyon fonksiyonudur. Doğrusal süzgeç problemlerinde sıklıkla kullanılmaktadır. Eş. 3.8.'de doğrusal aktivasyon fonksiyonu; Şekil 3.9.'da ise fonksiyonun grafiği bulunmaktadır.





Şekil 3.9. Doğrusal aktivasyon fonksiyonu

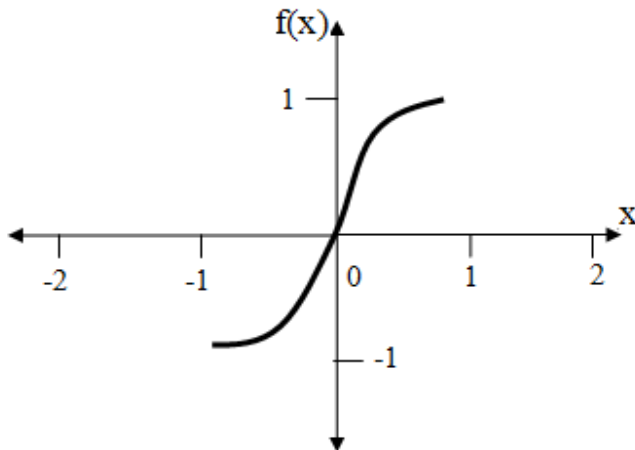
$$f(x) = x \quad (3.8.)$$

$f(x)$  : doğrusal aktivasyon fonksiyonu,

$x$  : girdi değeridir.

#### - Hiperbolik Tanjant Aktivasyon Fonksiyonu

Uygulamada en çok kullanılan aktivasyon fonksiyonlarından biridir. Bu fonksiyon, Şekil 3.10.'da görüldüğü gibi çift kutuplu bir fonksiyondur. Aktivasyon fonksiyonunun matematiksel gösterimi Eş. 3.9.'da verilmiştir.



Şekil 3.10. Hiperbolik tanjant aktivasyon fonksiyonu

$$f(x) = (1 - \exp(-2x)) / (1 + \exp(-2x)) \quad (3.9.)$$

$f(x)$  : hiperbolik tanjant aktivasyon fonksiyonu,

$x$  : girdi değeridir.

### **Eğitim Süreci**

Alkan (2001), Y.S.A.'nın en önemli özelliklerinden birisinin, ilgili probleme ait örnekler verildiğinde girdiler ile çıktılar arasındaki ilişkiyi öğrenebilmesi olduğunu söylemiştir. Y.S.A.'nın diğer bir süreci olan eğitim sürecinin temel amacı girdi ve hedef değer arasında bir eşleştirme yapmaktır. Danışmanlı öğrenme olarak adlandırılan yöntem ile ağ yeterince eğitildiğinde daha önceden görmediği girdilere karşı uygun çıktılar üretebilmektedir. Bu özellik Y.S.A.'nın genelleme kapasitesi olarak adlandırılır ve çoğunlukla ağın yapısına ve eğitim için uygulanan yineleme sayısına bağlıdır.

Y.S.A.'nın, veri yapısındaki ilişkiyi öğrenmesi, probleme ait örnekler yardımı ile ağ ağırlıklarının en uygun değerlerinin belirlenmesine dayanır.

### **3.3.Y.S.A.'ların Kullanım Alanları ve Yapısal Özellikleri**

Y.S.A., özellikle doğrusal olmayan sistemlerde öngörüler açısından istatistiksel tekniklere göre daha kolaylık sağlayan bir özelliğe sahiptir. Bundan dolayı başta işletme ve finans olmak üzere birçok değişik alanda kullanım imkanı bulur.

#### **Y.S.A.'ların Kullanım Alanları**

Mevcut verilerden öğrenerek karmaşık konularla başa çıkmak için eğitilmiş olan uygun sistemlerin tasarımı, optimizasyon problemleri ve kontrol uygulamalarına kadar pek çok alanda Y.S.A. kullanılmaktadır.

Y.S.A.'nın temel kullanım alanları aşağıdaki şekilde belirtilebilir.

- **Öngörü ve tahminleme:** Gelecekteki satışlar, hava tahminleri, at yarışları, çevresel risk, v.b.
- **Sınıflandırma ve Kümeleme:** Müşteri profilleri, tıbbi teşhis, ses ve şekil tanıma, hücre tipleri, v.b.

- **Kontrol** : Erken uyarı için uçaklarda ses ve titreşim düzeyleri, v.b.

Ayrıca, veri birleştirme, kavramsallaştırma ve filtreleme için de kullanılır. Bu özellikleri, Y.S.A.'nın karmaşık problemleri çözebilme yeteneğini gösterir.

### **Yapay Sinir Ağları'nın Yapısal Özellikleri**

Y.S.A.'nın hesaplama ve bilgi işleme gücünü, öğrenebilme ve genelleme yeteneğinden aldığı söylenebilir. Genelleme, öğrenme sürecinde karşılaşılmayan veriler için de Y.S.A.'nın uygun tepkileri üretmesi olarak tanımlanır. Günümüzde birçok bilim alanında Y.S.A., aşağıdaki özellikleri nedeniyle etkin olmuş ve uygulama yeri bulmuştur.

#### **A) Doğrusal Olmama**

Y.S.A.'nın temel işlem elemanı olan hücre, doğrusal olmayabilir. Bu durumda hücrelerin birleşmesinden meydana gelen Y.S.A. modeli de doğrusal olmaz. Beltratti, Margarita ve Terna (1996) doğrusal olmayan modellerde kullanılan belirli fonksiyonel yapılar, veriyi üreten fonksiyonun genellikle Y.S.A.'ların ima ettiği farklı olduğu ve bu yüzden Y.S.A.'ların kullanılması için gerekli ekonometrik teorinin eksik tanımlı doğrusal olmayan modeller için olduğunu ima etmektedir. Y.S.A.'lar herhangi bir sürekli fonksiyona ya da türevlerine yakınsama yeteneğine sahiptir ve bu yüzden Evrensel Fonksiyon Yakınsayıcı Yöntem (Universal Function Approximator Method) olarak tanımlanmaktadır. Bu özelliği ile Y.S.A., doğrusal olmayan karmaşık problemlerin çözümünde en önemli araçlardan biri olmuştur. (Yurtoğlu, H., 2005)

#### **B) Öğrenme**

Y.S.A.'nın arzu edilen davranışı gösterebilmesi için amaca uygun olarak ayarlanması, herhangi bir problemin çözümü için probleme yönelik bir algoritmanın geliştirilmesi gerekmektedir. Y.S.A.'nın karmaşık yapısı nedeniyle bağlantılar ve ağırlıklar önceden ayarlı olarak verilemez ya da tasarlanamaz. Bu nedenle Y.S.A., istenen davranışı gösterecek şekilde ilgilendiği problemde aldığı eğitim örneklerini kullanarak problemi öğrenmelidir. Böylece olayları öğrenerek benzer olaylar karşısında benzer kararlar vermeye çalışır; kendisine gösterilen

örneklerden genellemeler yaparak daha önce görmediği örnekler hakkında bilgiler üretebilir.

### C) Öğrenme Hızı

Y.S.A. yapısı, eğitim süresi boyunca N parametrelili bir ağ için N+1 boyutlu bir uzayda, N değişkenli bir yüzey üzerinde gezen bir noktanın, hatayı en aza indiren noktayı aramasını gerektirir. Süreç içerisinde eğitim işlemi, çok uzun sürebilmektedir. Bu durumu engellemek için sisteme  $\eta$  öğrenme katsayısı,  $\mu$  momentum terimi ve her bir hücre için bir eşik değeri ilave edilir (Bkz, Eş. 3.10.).

$$w_{pq,k}(N+1) = w_{pq,k}(N) - \eta_{p,q} \delta_{pq,k} \Phi_{p,j}(I) + \mu \Delta w_{pq,k}(N) \quad (3.10.)$$

Yukarıdaki denklemde;

w : ağırlık değeri,

p,q : hücre numarası

k : katman sayısı,

N : devir sayacı,

$\eta$  : öğrenme katsayısı,

$\delta$  : geri yayılacak hata terimi,

$\mu$  : momentum değeri,

$\Phi(I)$  : ağırlık bulduğu çıktı değeridir.

### D) Genelleme

Y.S.A., ilgilendiği problemi öğrendikten sonra eğitim sırasında karşılaşmadığı test örnekleri için de arzu edilen tepkiyi üretebilir. Örneğin, karakter tanıma amacıyla eğitilmiş Y.S.A., bozuk karakter girdilerinde de doğru karakterleri verebilir ya da bir sistemin eğitilmiş Y.S.A. modeli, eğitim sürecinde verilmeyen girdi sinyalleri için de sistemle aynı davranışı gösterebilir.

## **E) Uyarlanabilirlik**

Y.S.A., ilgilendiği problemdeki değişikliklere göre ağırlıklarını ayarlar. Yani, belirli bir problemi çözmek amacıyla eğitilen Y.S.A., problemdeki değişimlere göre tekrar eğitilebilir ve değişimler devamlı ise gerçek zamanda da eğitime devam edilebilir. Bu özelliği ile Y.S.A., uyarlamalı örnek tanıma, sinyal işleme, sistem tanılama ve denetim gibi alanlarda etkin olarak kullanılır.

## **F) Hata Toleransı**

Y.S.A., çok sayıda hücrenin çeşitli şekillerde bağlanmasından oluştuğu için paralel dağılmış bir yapıya sahiptir ve ağırlık sahibi olduğu bilgi, ağıdaki bütün bağlantılar üzerine dağılmış durumdadır. Bu nedenle, eğitilmiş Y.S.A.'nın bazı bağlantılarının hatta bazı hücrelerinin etkisiz hale gelmesi, önemli ölçüde ağına doğru bilgi üretmesine sebep olmaz. Bu nedenle, geleneksel yöntemlere göre hatayı tolere etme yeteneği son derece yüksektir.

## **G) Hata Değişimi**

Hata değişimini inceleyen iki çeşit kural vardır.

1- Hata düzeltme kuralları,

2- Gradyan kuralları.

Hata düzeltme kurallarında, her bir girdi görüntüsünde ağırlıklar yeniden belirlenerek hata en aza indirilmeye çalışılır. Gradyan kurallarında ise, ağırlıklar yeniden ayarlanarak Hata Kareler Ortalaması (H.K.O.) en aza indirilmeye çalışılır. Bu noktada gradyan kuralından kısaca bahsedecek olursak, hatayı minimize etmenin geometrik bir yorumunu yapmak mümkündür. Bunu yapabilmek için ağırlıkların mümkün olan tüm değerleri, hataların kareleri toplamına karşı gelecek şekilde üç boyutlu koordinat sisteminde çizilir. Bu çizim sonunda hata yüzeyi küresel bir top şeklindedir. Bu yüzeyi bir tase benzetmek mümkündür. Tasın en alt kısmı hataların kareleri toplamının en küçük değerlerine karşı gelmektedir. Eğitim sırasında amaç ağırlıklar kümesinin en iyisini bulmak olan, en alt kısmına ulaşmaktır. Geri yayılım algoritması o andaki ağırlıklar yerine, yüzey hatasının eğimini hesaplayarak amacına ulaşır. Daha sonra da ağırlıkları tasın alt kısmına

dođru artırarak deđiřtirir. İřte bu tasın üřt kısmından alt kısmına dođru ilerleme iřlemine “gradyan iniř” denir.

Bishop (1995), hata yüzeyinin çođu kez düzensiz ve karmařık bir yapıya sahip olması nedeniyle, sınır ađının optimum olmayan “yerel minimum” deđerine yakınsayabildiđini belirtmiřtir. Ađ ađırlıkları istenen ve gerçekleřen ađ sonuçları arasındaki fark ölçüsüne dayanan bir hata deđerini minimum yapacak řekilde dengelenmektedir.

Ađırlık vektörü ile çalıřan Y.S.A.'da önemli noktalardan birisi, bir öğrenme kuralı geliřtirip, yan deđeri ile  $w$  ađırlık vektörünü istenilen Y.S.A. performansını verecek noktaya yöneltmektir. Genellikle öğrenme kuralı için bir performans ya da maliyet fonksiyonu tanımlanır. Minimizasyon ya da maksimizasyon ile " $w$ " vektörü bulunur ve H.K.O. tanımlanır.

Öte yandan yaygın olarak kullanılmakta olan hata düzeltme kuralı uzun zaman gerektirmektedir. Ayrıca ađırlık güncellemelerini ardıřık ve yerel olarak hesaplandıđı için yerel minimuma yakalanma riski vardır. Bu nedenle algoritmayı hızlandırmak için çeřitli arařtırmalar yapılmaktadır.

#### **H) Deđiřken Sayısının ve Kullanılan Parametrelerin Sınırsız Olması**

Y.S.A. modelleri sınırsız sayıda deđiřken ve parametre ile çalıřabilmektedir. Bu sayede mükemmel bir öngörü dođruluđu ile genel çözümler sađlanabilmektedir.

#### **i) Analiz ve Tasarım Kolaylıđı**

Ergezer, Dikmen ve Özdemir (2003), Y.S.A.'nın temel iřlem elemanı olan hücrenin yapısı ve modelinin, bütün Y.S.A. yapılarında yaklaşık aynı olduđunu belirtmiřtir. Dolayısıyla, Y.S.A.'nın farklı uygulama alanlarındaki yapıları da standart yapıdaki bu hücrelerden oluşacaktır. Bu nedenle, farklı uygulama alanlarında kullanılan Y.S.A.'ları benzer öğrenme algoritmalarını ve teorilerini paylaşabilirler. Bu özellik, problemlerin Y.S.A. ile çözümünde önemli bir kolaylık getirecektir.

## **J) Genelleştirme Yeteneği**

Y.S.A.'ların en önemli özelliklerinden biri de bilgiyi depolamalarıdır. Y.S.A.'da bilgi, ağırlıklar üzerine dağıtılmıştır. Bağlantıların ağırlıkları ise ağırlık hafıza birimidir. Bu ağırlıklar ağırlık o andaki sahip olduğu bilgiyi ya da uygulanan örneklerden öğrenmiş olduğu davranışı verir. Bu bilgiler, ağırlık bir çok ağırlıklar üzerine (hafıza birimine) dağıtılır.

Eğitilmiş ağırlık, eğitimde kullanılmamış herhangi farklı bir girdi uygulanırsa ağırlık daha önceki girdilerden öğrenmiş olduğu davranış doğrultusunda beklenen çıktıya uygun bir çıktı değeri üretebilecektir. Y.S.A.'ya uygulanan veri eksik, gürültülü ya da daha önce hiç karşılaşmamış olsa bile, ağırlık kabul edilebilir en uygun çıktıyı üretecektir. Bu özelliğe, genelleştirme yeteneği denir.

### **3.4. Y.S.A.'nın Kullanım Alanları**

#### **İstatistik ve Ekonomi**

Ekonomik değişkenlerin modellenmesi ve tahmin edilmesi konusu ekonomi alanı için oldukça önem taşımaktadır ve genellikle istatistiksel yöntemlerin kullanımını gerektirmektedir.

Y.S.A. ve istatistik alanları arasında önemli sayılabilecek bir kesişim kümesi vardır. İstatistik, veri analizi ile uğraşan bir alandır. Benzer şekilde, sinir ağı terminolojisindeki hatalı veriden öğrenerek genelleştirme kavramı (ya da yeteneği), istatistiksel çıkarım ile aynı anlamı taşımaktadır. Dolayısıyla, Y.S.A.'ları da genelde veri analizini içermektedir.

Zaman serilerinin tahmin edilmesi sınıflama konusunda sıklıkla kullanılıyor olması nedeniyle, Y.S.A.'larla istatistik ve dolayısıyla ekonomi bilimlerinde de kullanılmaktadır.

#### **Sigorta ve Finans**

Bankacılıkta kredi uygulamaları geliştirilmesi, müşteri analizi ve kredi müracaat değerlendirilmesi, bütçe yatırım tahminlerinde uygulanabilir. Bunun yanında sigortacılıkta ürün optimizasyonu, uygulama politikası geliştirme, kıymet biçme,

pazar performans analizi, bütçe kestirimi, hedef belirleme, tahmin yöntemleri olarak da kullanılmaktadır.

### **Uzay, Otomotiv ve Haberleşme**

Arıza analizi ve tespiti bir sistemin, cihazın ya da elemanın düzenli ve doğru çalışma şeklini öğrenen Y.S.A. yardımıyla bu sistemlerde meydana gelebilecek arızaların tanımlanma olanağı vardır. Bu amaçla Y.S.A.; elektrik makinelerinin, uçakların ya da bileşenlerinin, entegre devrelerin v.s. arıza analizinde kullanılmaktadır.

Savunma sanayi silahların otomasyonu ve hedef izleme, nesnelere/görüntüleri ayırma ve tanıma, yeni algılayıcı tasarımı ve gürültü önleme v.s gibi alanlara uygulanmaktadır.

Haberleşme, görüntü ve veri sıkıştırma, otomatik bilgi sunma servisleri, konuşmaların gerçek zamanda çevirisi v.s gibi alanlarda uygulama örnekleri vardır.

Üretim sistemlerinin optimizasyonu, ürün analizi ve tasarımı, ürünlerin (entegre, kağıt, kaynak v.s.) kalite analizi ve kontrolü, planlama ve yönetim analizi v.s. alanlarına uygulanmaktadır.

Uçaklarda otomatik pilot sistemi otomasyonu, ulaşım araçlarında otomatik yol bulma/gösterme, robot sistemlerin kontrolü, doğrusal olmayan sistem modelleme ve kontrolü, elektrikli sürücü sistemlerin kontrolü v.s. gibi yaygın bir uygulama alanı bulmuştur.

### **Tıp**

EEG ve ECG gibi tıbbi sinyallerin analizi, kanserli hücrelerin analizi, protez tasarımı, transplantasyon zamanlarının optimizasyonu ve hastanelerde giderlerin optimizasyonu v.s gibi uygulama alanları vardır.

### **Mühendislik**

Kimya mühendisliği; kimyasal reaktör seçimi, dinamik işlemlerde hata belirlenmesi, endüstriyel polimerizasyonda eritme akış indisi tahmini, endüstriyel mayalama



işleminin modellenmesi, biyokimyasal işlemlerde mikrobik konsantrasyonların tahmininde uygulanmaktadır.

İnşaat ve yapı mühendisliği; konstrüksiyon projelerinde kaynak seviyelerini belirleme, bir rezervuardan çıktığı kontrol, biyolojik bilgiler yardımıyla nehir suyu kalitesinin sınıflandırılması, nehirlerin akışının tahmin edilmesi, sonlu-eleman-temelli yapısal analiz işleminin modellenmesi, yapı malzemelerinin iç yapılarındaki çatlakların tespit edilmesi, depreme maruz betonarme çerçevelerde emniyetli yatay taşıyıcı tahmininde uygulanmaktadır.

Elektrik ve elektronik mühendisliği; hastaların alarmla kontrolü, gürültülü resimlerin kalitesini artırma, görüntülerin sıkıştırılması, gürültü filtreleme ve resimlerdeki kenar bilgisinin çıkarılması, güç sisteminde harmoniklerin tahmini, gezgin haberleşme sisteminde kanal dağıtımı, ultrasonik müziklerden objelerin sınıflandırılması, optik okuyucu sistemler için resimlerin ön işleme alanlarında kullanılmaktadır.

İmalat ve makine mühendisliği; metal kesme tezgahının kontrolü, akustik salınım ve iş parçası kuvvetiyle iş parçası yatağının kontrolü ve güç tüketimi ve iş parçası ivmesi, hücreli imalat için grup teknolojisi parça gruplarının tasarımı, hareket eden nesnelere için engelsiz yol planlaması, makine parametrelerinin optimizasyonu, makine arızalarının sınıflandırılması, malzemelerin ısı transferinin belirlenmesi, uçak kanat kutularının yapısının tasarımında uygulanmaktadır.

Sistem ve kontrol mühendisliği; esnek kollu robotun kontrolü, bir model helikopterin havada kontrolü, çok değişkenli robotun yörünge koordinasyonu, iki sıvı tank sisteminin akış seviye kontrolü, anestezi derinliği kontrolü ve ölçülmesi, bir robot için optimal yolun bulunması, banyo suyu sıcaklığının kontrolü, endüstriyel robot kontrolü, sistem kimliklendirme gibi birçok alanda uygulanmaktadır.

Anten ve uygulamaları; Y.S.A.'larda bir dinamik öğrenme algoritması kullanılarak anten dizi elemanlarından elde edilen işaretler arasındaki faz farklılıklarının karşılaştırılmasıyla radar izleme gerçekleştirilmektedir.

## **Dil**

Sözcük tanıma, yazı ve konuşma çevrimi, dil tercüme vs.

## **Telekomünikasyon**

Görüntü ve veri karşılaştırma, filtreleme, eko ve gürültü giderme, ses ve görüntü işleme, trafik yoğunluğunun kontrolü ve anahtarlama vs.

## **Güvenlik**

Parmak izi tanıma, kredi kartı hileleri saptama, retina tarama, yüz eşleştirme vs. Verilen örnekler çoğaltılabilir. Görüldüğü gibi Y.S.A.'lar, sınıflandırma, olasılıksal fonksiyon kestirimi, örüntü eşleştirme, örüntü tanıma, optimizasyon, doğrusal olmayan sistem modelleme, zaman serisi analizi, tahmin ve sinyal işleme gibi farklı alanlarda başarı ile kullanılmaktadır. Gün geçtikçe uygulama alanları genişlemekte ve gelişmektedir.

### **3.5. Y.S.A.'ların Sınıflandırılması**

Hücre çıktıları, ağırlıklar üzerinden diğer hücrelere ya da kendisine girdi olarak bağlanabilir ve bağlantılarda gecikme birimi de kullanılabilir. Hücrelerin bağlantı şekillerine, öğrenme kurallarına ve aktivasyon fonksiyonlarına göre çeşitli Y.S.A. yapıları geliştirilmiştir. Her bir sinir hücresi arasındaki bağlantıların yapısı ağın yapısını belirler. İstenilen hedefe ulaşmak için bağlantıların nasıl değiştirileceği öğrenme algoritması tarafından belirlenir. Kullanılan bir öğrenme kuralına göre, hatayı sifra yakınsayacak şekilde, ağın ağırlıkları değiştirilir. Y.S.A.'lar yapılarına ve öğrenme algoritmalarına göre sınıflandırılırlar. Günümüzde, belirli amaçlarla ve değişik alanlarda kullanılmaya uygun birçok yapay sinir ağı modeli (Perceptron, Adaline, Ç.K.A., Öğrenme Vektör Nicelendirmesi (Ö.V.N.), Hopfield, Recurrent, SOM, Uyarlanabilir Rezonans Teorisi vb.) geliştirilmiştir.

#### **3.5.1. Y.S.A.'ların Mimari Yapılarına Göre Sınıflandırılması**

Genel olarak Y.S.A.'lar mimari yapılarına göre, ileri beslemeli ve geri beslemeli ağlar olmak üzere sınıflandırılırlar. Bu iki ağ yapısı dışında bellek tabanlı, R.T.F. ve modül yapay sinir ağları da bulunmaktadır.

## A) İleri Beslemeli Yapay Sinir Ağları

İleri Beslemeli Yapay Sinir Ağları'nda (İ.B.Y.S.A.) işlemci elemanlar genellikle katmanlara ayrılmışlardır ve bir katmandaki hücrelerin çıktıları bir sonraki katmana ağırlıklar üzerinden girdi olarak verilir. Yani girdiler, girdi katmanından çıktı katmanına doğru tek yönlü bağlantılarla iletilir.

İ.B.Y.S.A.'da temel olarak giriş, gizli ve çıkış katmanları olmak üzere üç çeşit katman bulunur. Sırasıyla girdi katmanı, Y.S.A.'ya giren verileri tutan katman, işlemlerin yapıldığı ve istenilen sonuca göre kendisini eğiten gizli katman ya da katmanlar ve son olarak çıktı değerlerini gösteren çıkış katmanıdır.

Nöronlar, girdi katmanından çıkış katmanına doğru tek yönlü bağlantılarla iletilir. Girdi katmanı, dış ortamlardan aldığı bilgileri hiçbir değişikliğe uğratmadan gizli katmandaki hücrelere iletmektedir (Zurada, 1992). Nöronlar bir katmandan diğer bir katmana bağlantı kurarlarken, aynı katman içerisinde bağlantıları bulunmaz. Bir girdiye verdikleri bilgi, ağın önceki durumundan bağımsızdır. Bu nedenle bu bilgi bellekte tutulmaz. Bilgi, gizli ve çıkış katmanından işlenerek ağ çıktısı belirlenir. Bu yapısı ile ileri beslemeli ağlar doğrusal olmayan statik bir işlevi gerçekleştirerek yalnızca bir çıktı seti üretirler.

Kaç tane gizli katman kullanılacağı probleme göre belirlenmektedir. Her katmanda bir ya da daha çok sayıda nöron bulunabilir.

İ.B.Y.S.A.'larını içerdiği katman sayısına göre sınıflandırabiliriz.

### - Tek Katmanlı İleri Beslemeli Yapay Sinir Ağları

Bir giriş ve bir çıkış katmanından oluşan en basit ağ yapısına sahiptir. Girdi katmanı veri üzerinde hiçbir işlem yapmadan veriyi çıkış katmanına iletir.

### - Çok Katmanlı İleri Beslemeli Yapay Sinir Ağları

Çok Katmanlı İleri Beslemeli Yapay Sinir Ağları, Ç.K.A. olarak da bilinirler. Bir ya da daha fazla gizli katman içermektedir. Gizli katmanlara ihtiyaç duyulmasının sebebi girdi katmanlarından gelen genellikle işlenmemiş sinyallerin özelliklerini belirlemek, ağırlıklandırmak ve sonuçları çıktı

katmanına yönlendirmektir. Gizli katmanlara “gizli” denilmesinin sebebi istenilen çıktı değerini saklamasıdır.

Tezin ilerleyen bölümlerinde tek katmanlı ve çok katmanlı Y.S.A.’lar daha detaylı olarak açıklanacaktır.

## **B) Geri Beslemeli Yapay Sinir Ağları**

Geri Beslemeli Yapay Sinir Ağı (G.B.Y.S.A.), çıkış ve gizli katmanlardaki çıktıların, girdi katmanına ya da önceki gizli katmanlara geri beslendiği bir ağ yapısıdır. Diğer bir deyişle, geri beslemeli sinir ağının, ileri beslemeli bir ağın çıktılarının girdilere bağlanması ile elde edildiği söylenebilir.

Geri besleme, bir katmandaki hücreler arasında olduğu gibi katmanlar arasındaki hücreler arasında da olabilir. Geri beslemenin yapılış şekline göre farklı yapıda ve davranışta G.B.Y.S.A. yapıları elde edilebilir. Bunlar;

### **- Tam Geri Beslemeli Yapay Sinir Ağları (T.G.B.Y.S.A.)**

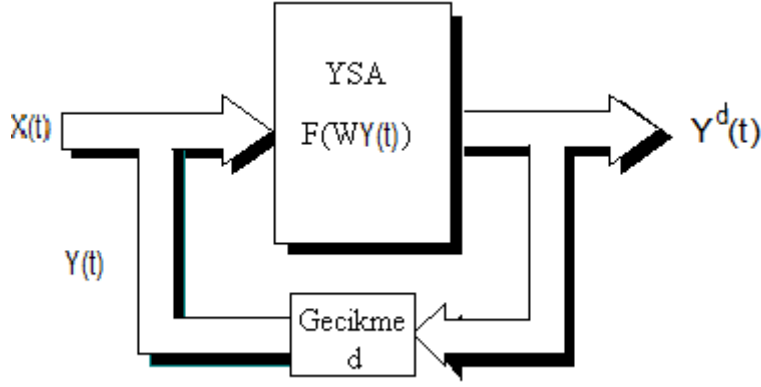
Bu ağlar gelişi güzel ileri ve geri bağlantıları olan ağlardır. Bu ağların hepsi eğitilebilir.

### **- Kısmi Geri Beslemeli Yapay Sinir Ağları (K.G.B.Y.S.A.)**

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

G.B.Y.S.A., doğrusal olmayan dinamik bir davranış gösterir. Geri beslemeli sinir ağlarında gecikmeler olduğu da söylenebilir.

Şekil 3.11.’de geri beslemeli ağ için blok diyagram görülmektedir. Bu çeşit sinir ağlarının dinamik hafızaları vardır ve bir andaki çıktı hem o andaki hem de önceki girdileri yansıtır. Bundan dolayı, özellikle önceden tahmin uygulamaları için uygundur. Bu ağlar çeşitli tipteki zaman serilerinin tahmininde oldukça başarı sağlamışlardır. Bu ağlara örnek olarak Hopfield, SOM, Elman ve Jordan ağları verilebilir.



Şekil 3.11. G.B.Y.S.A. için blok diyagram

Burada;

$X(t)$  : t nöronlu girdi katmanı,

$Y(t)$  : t nöronlu girdi katmanına ait gizli katman,

$d$  : gecikme sayısı,

$Y^d(t)$  : d gecikmeli t nöronlu çıktılar,

$F(W Y(t))$  : ağırlıkların birimlerinde tercih edilen aktivasyon fonksiyonudur.

### C) Bellek Tabanlı Yapay Sinir Ağları

Doğrusal olmayan sistemlerin tanımlanması ve denetiminde, katmanlı Y.S.A. yapıları etkin olarak kullanılmaktadır. Bellek Tabanlı Yapay Sinir Ağları'nda hem öğrenme çok hızlıdır, hem de gereken Y.S.A. öğrenme sırasında oluşturulur; ağ yapısı ile ilgili her hangi bir ön kabule gerek yoktur. Y.S.A. ile sistem tanımlamada, doğru model yapısının seçilebilmesiyle model girdilerinin belirlenebilmesi için sistemin girdi ve çıktısının gecikme derecelerinin bilinmesi gerekir. Örneğin, Alpaydın (1990)'da, belirtildiği gibi büyü ve öğren yönteminde öğrenme şu şekilde olur:

**Aşama 1.** Öğrenme kümesinden gelişmiş güzel bir örnek alınır.

**Aşama 2.** Eldeki Y.S.A.'nın bu girdiye verdiği tahmin hesaplanır.

**Aşama 3.** Eğer verilen tahmin, orijinal değere yeterince yakınsa herhangi bir şey yapılmaz. Eğer arada tanımlanmış bir tolerans değerinden daha büyük bir fark var ise, bu girdi ve çıktı Y.S.A.'nın belleğine eklenir. Bu değişiklik, Y.S.A.'nın bir sonraki seferde benzer bir girdi ile karşılaştığında doğru çıktı verebilmesini sağlar.

**Aşama 4.** Öğrenme kümesinde görülmemiş örnek kaldı ise, Aşama 1'e dönülür.

Doğrusal olmayan bir sistemi modelleme ve denetim yeteneği, sadece sistemin o andaki girdi ve bir önceki çıktı verileri model girişi alınarak incelenmiş ve tatmin edici sonuçlar alındığı gösterilmiştir.

#### **D) Radyal Tabanlı Fonksiyon Yapay Sinir Ağları**

Radyal Tabanlı Fonksiyon Yapay Sinir Ağları (R.T.F.Y.S.A.) tasarımı ise, çok boyutlu uzayda eğri uydurma yaklaşımıdır ve bu nedenle R.T.F.Y.S.A.'nın eğitimi, çok boyutlu uzayda eğitim verilerine en uygun bir yüzeyi bulma problemine dönüşmektedir. Verleysen ve Hlavackova (1994)'de belirtildiği gibi, R.T.F.Y.S.A.'da temel fikir, bir grup radyal taban fonksiyonu istenen f fonksiyonuna yaklaşacak şekilde ağırlıklandırarak toplamaktan ibarettir. Radyal tabanlı fonksiyon 3 katmanlı bir yapıdır. Girdi katmanı girdi vektör uzayı ile, çıktı katmanı da örüntü sınıfları ile ilişkilidir. Böylelikle tüm yapı, gizli katmanın yapısı ve gizli katman ile çıkış katmanı arasındaki ağırlıkların belirlenmesine indirgenir. Gizli katmandaki nöronların aktivasyon fonksiyonları bir  $C_j$  merkezi ve  $\sigma_j$  bant genişliği ile belirlenir. Aktivasyon fonksiyonu, Eş. 3.11'de tanımlanan bir Gauss eğrisidir.

$$\varphi_j(x) = \exp\left(-\frac{\|x-C_j\|^2}{2\sigma_j^2}\right) \quad (3.11.)$$

#### **E) Modül Yapay Sinir Ağları**

Modül Yapay Sinir Ağları (M.Y.S.A.) çok sayıda Y.S.A. yapısının birleşiminden oluşur. Eğer, bir ağın yapması gereken işlemler birbirleriyle de haberleşmeksizin

iki ya da daha fazla modüle (alt yapıya) ayrılabilirse bu ağlar M.Y.S.A. olarak adlandırılır. Modüller arasında seyrek bağlantılar vardır.

Modüllerin çıktıları, modüllerden bilgi geri beslenmemek üzere bir birleştirme birimi ile birleştirilir ve birleştirme birimi;

- M.Y.S.A. çıktısını elde etmek için modül çıktılarının nasıl birleştirilmesi gerektiğini,
- Hangi eğitim örneklerini hangi modülün öğreneceğini kararlaştırmalıdır.

## F) Hopfield Ağı

Hopfield Ağı (H.A.), tek katmanlı ve geri dönüşümlü ağıdır. Süreç elemanlarının tamamı hem girdi hem de çıktı elemanlarıdır. Ağın bağlantı değerleri, bir enerji fonksiyonu olarak saklanmaktadır.

Hopfield yapay sinir ağı, ağ içerisinde geriye doğru dönüşe izin veren yani ağ içerisinde yönlü çember oluşturabilen bir Y.S.A. yapısına sahiptir. Tekrar besleme kabiliyeti denilen bu özelliği ile diğer Y.S.A. modellerinden ayrılmaktadır. Ağın bu kabiliyeti sayesinde girdi örüntüsü Hopfield mimarisine verildiğinde, mimari işleme bir başlangıç enerjisi ile başlanılmış olur. Bu başlangıç konumundan itibaren ağ, girdi örüntüsünü bir başka girdi örüntüsüne doğru yönlendirmeye başlar. Bu süreçte girdi örüntüsünde yapılan her ufak değişimin ardından enerji tekrar tekrar hesaplanarak girdi örüntüsünün morfolojik dönüşümünün kontrolü sağlanır. Girdi örüntüsünün bir başka öğrenilmiş olan girdi örüntüsüne benzetilme işlemi (morfolojisi), enerji durağan olana dek sürer. Enerji durağanlaşması ise enerjinin minimuma ulaştığı ve değişmediği yerdir.

Günümüzde geliştirilmiş iki tür Hopfield ağı vardır. Bunlar;

- **Kesikli Hopfield Ağı:** Çağrışımlı bellek olarak kullanılır. (Ör: Görüntü tanıma ve onarma)
- **Sürekli Hopfield Ağı:** Kombinatoriyal optimizasyon problemlerinde kullanılır (Ör: Gezgin satıcı problemi)

Hopfield Ağı'nın özellikleri aşağıda verilmiştir.

1. Aynı-uyum sınıflandırmasında kullanılan ilk ağlardan biridir.
2. Tek katmanlı ve geri dönüşümlü ağıdır.
3. Süreç elemanlarının tamamı hem girdi hem de çıktı elemanlarıdır.
4. Hücreler açık (+1) ya da kapalı (-1) olarak ikili mantığa göre çalışır.
5. İşlemci fonksiyonu, süreksiz formda eşik fonksiyonu, sürekli formda sigmoid ve tanjant hiperbolik fonksiyonu olarak düşünülmüştür.
6. Ağın bağlantı değerleri bir enerji fonksiyonu olarak saklanmaktadır.
7. Ağın öğrenmesi Hebb kuralına göre yapılır.

### **3.5.2.Y.S.A.'ların Öğrenme Kuralları**

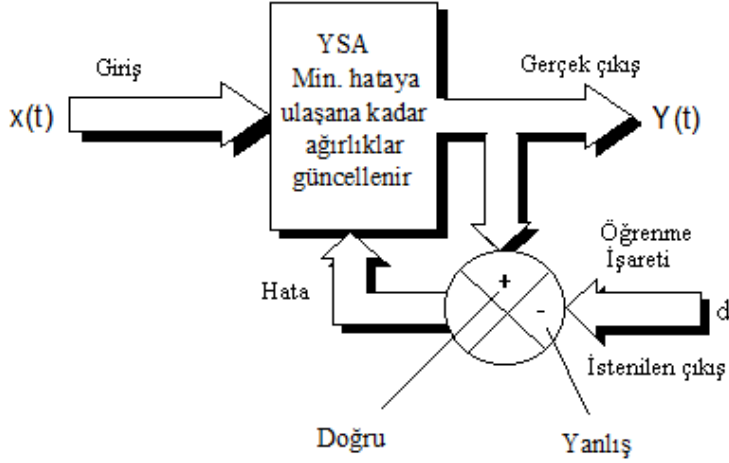
Öğrenme; gözlem, eğitim ve hareketin doğal yapıda meydana getirdiği davranış değişikliği olarak tanımlanmaktadır. O halde, birtakım yöntem ve kurallar baz alınarak gözlem ve eğitime göre ağıdaki ağırlıkların değiştirilmesi sağlanmalıdır. Bunun için genel olarak üç öğrenme yönteminden ve bunların uygulandığı değişik öğrenme kurallarından söz edilebilir. Bu öğrenme kuralları aşağıda açıklanmaktadır.

#### **Danışmanlı Öğrenme (Denetimli Öğrenme)**

Bu tip öğrenmede, ağa verilen girdi değerleri için çıktı değerleri de verilir. İstenilen ve gerçek çıktı arasındaki farka (hataya) göre ağ, bağlantıların ağırlığını en uygun çıktıyı elde etmek için sonradan düzenleyebilir. Her iterasyonda örneğe ait çıktı değerleri ile ağın çıktı değerleri karşılaştırılarak ağın hatası hesaplanır. Ağın çıktıları ile beklenen çıktılar arasındaki hata hesaplandıktan sonra, ağın yeni ağırlıkları bu hata payına göre düzenlenir. Hata payı hesaplanırken ağın bütün çıktıları ile beklenen çıktıları arasındaki fark hesaplanır ve bu farka göre her nörona düşen hata payı bulunur. Daha sonra her nöron kendine gelen ağırlıkları günceller. Bu hata minimum olana kadar ağ, nöronlar arasındaki ağırlıkları düzelterek iterasyona devam eder. Bu sebeple danışmanlı öğrenme algoritmasının bir "öğretmene" ya da "danışmana" ihtiyacı vardır. Şekil 3.12.'de danışmanlı öğrenme yapısı gösterilmiştir.



Widrow-Hoff tarafından geliştirilen delta kuralı ve Rumelhart ve McClelland tarafından geliştirilen Genelleştirilmiş Delta Kuralı ya da Geri Yayılım Algoritması (G.Y.A.) danışmanlı öğrenme algoritmalarına örnek olarak verilebilir. Y.S.A.'da en yaygın öğrenme biçimidir.



Şekil 3.12. Danışmanlı öğrenme yapısı

Fausett (1994), hatayı en aza indirmek için bağlantı ağırlıklarının yeniden düzenlenerek güncellendiğini belirtmiştir. Böylece ağı gerçek çıktı değerlerine en yakın çıktı değerlerini üretmesi amaçlanır.

Danışmanlı öğrenmeyi kullanan modellere örnek olarak Ö.V.N. modeli verilebilir. Bu modelde ağı sadece üretilen çıktının doğru olup olmadığı bilgisi verilir. Diğer bir ifadeyle, ağı üretmiş olduğu çıktılar hakkında ağı doğru ya da yanlış türünden geri bildirimler verilerek ağı eğitmeye çalışılmaktadır. Bu bağlamda bakıldığında Ö.V.N. modelini kullanan Y.S.A., danışmanlı öğrenmeyi kendi içerisinde uygulamaktadır.

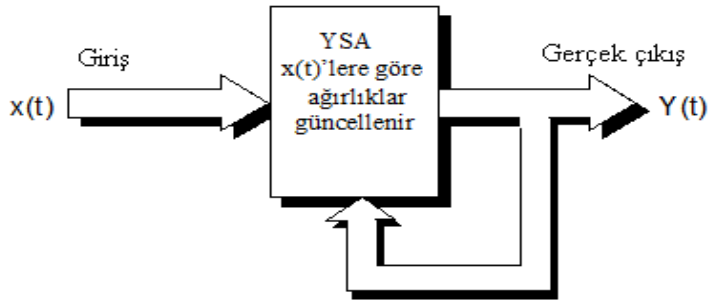
### **Danışmansız Öğrenme (Denetimsiz Öğrenme)**

Girdiye verilen örnekten elde edilen çıktı bilgisine göre ağı sınıflandırma kurallarını kendi kendine geliştirmektedir. Yani bu öğrenme algoritmalarında, istenilen çıktı değerinin bilinmesine gerek yoktur. Danışmansız öğrenmede ağı öğrenme sırasında sadece örnek girdiler verilmektedir. Herhangi bir beklenen çıktı bilgisi verilmez. Girdiye verilen bilgilere göre ağı her bir örneği kendi arasında sınıflandıracak şekilde kendi kurallarını oluşturarak, bağlantı ağırlıklarını aynı

özellikleri gösteren desenler oluşturmak üzere ayarlar. Böylece ağ öğrenme işlemini tamamlamış olur.

Özetlediğimizde Haykin (1999a)'da belirtildiği gibi, danışmansız öğrenmede ağa sadece girdi veri grubunun sunulduğu ve bu veri grubuna uyumlu bir çıktı değeri üretecek şekilde ağın kendisinin uygun ağırlıkları düzenlemesini istediği belirtilmiştir. Şekil 3.13.'te danışmansız öğrenme yapısı gösterilmiştir.

Danışmansız öğrenmede, girdiye verilen örneklerden elde edilen çıktı bilgisine göre ağ sınıflandırma kurallarını kendi kendine yapar. Bu öğrenme tipine Grossberg tarafından geliştirilen Uyarlanabilir Rezonans Teorisi ve Kohonen tarafından geliştirilen Özdüzenleyici Haritalar örnek olarak verilebilir.



Şekil 3.13. Danışmansız öğrenme yapısı

### Takviyeli Öğrenme

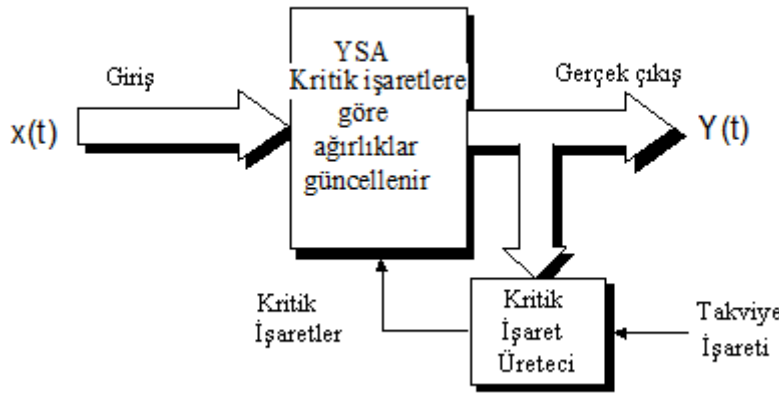
Bu öğrenme kuralı danışmanlı öğrenmeye yakın bir yöntemdir. Danışmansız öğrenme algoritması, istenilen çıktının bilinmesine gerek duymaz. Hedef çıktıyı vermek için bir “öğretmen” yerine, burada Y.S.A.’ya bir çıktı verilmemektedir. Bu öğrenme yaklaşımında ağın her iterasyonu sonucunda elde ettiği sonucun iyi ya da kötü olup olmadığına dair bir bilgi verilir. Ağ bu bilgilere göre kendini yeniden düzenler. Bu sayede ağ herhangi bir girdi dizisiyle hem öğrenerek hem de sonuç çıkararak işlemeye devam eder.

Örneğin satranç oynayan Y.S.A., yaptığı hamlenin iyi ya da kötü olduğunu anlık olarak ayırt edememesine rağmen yine de hamleyi yapar. Eğer oyun sonunda program oyunu kazanırsa, Y.S.A. yaptığı hamlelerin iyi olduğunu varsayacaktır ve

bundan sonraki oyunlarında benzer hamleleri iyi olarak değerlendirerek oynayacaktır.

Şekil 3.14.'te takviyeli öğrenme yapısı gösterilmiştir. Optimizasyon problemlerini çözmek için Hinton ve Sejnowski'nin geliştirdiği Boltzmann Kuralı takviyeli öğrenmeye örnek olarak verilebilirler.

Özellikle optimizasyon problemlerini çözmeye hedef çıktıyı belirlemek için, istenilen ve elde edilen çıktı arasındaki iyiliği değerlendiren bir ölçüt bulunmaktadır. Bu yöntem en iyi örnek yine Boltzmann Kuralı ve Genetik Algoritma verilebilir.



Şekil 3.14. Takviyeli öğrenme yapısı

Çalışmamızda öğrenme yapısı olarak danışmanlı yani denetimli öğrenme yöntemi kullanılacaktır.

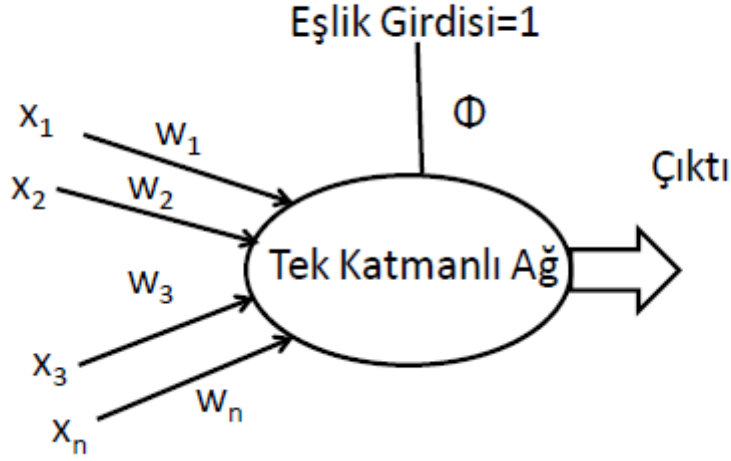
### 3.5.3.Y.S.A.'nın Yapısındaki Katmanlara Göre Sınıflandırılması

Y.S.A., içerdikleri katman sayısına göre tek katmanlı ve çok katmanlı olarak da sınıflandırılabilirler.

#### Tek Katmanlı Yapay Sinir Ağları

Tek Katmanlı Yapay Sinir Ağları (T.K.Y.S.A.) sadece girdi ve çıktı katmanlarından oluşur. Ara katmanlar bulunmaz. Her ağın bir ya da daha fazla girdisi çıktısı vardır. Şekil 3.15.'te tek katmanlı ağ yapısı gösterilmiştir.

Tek katmanlı algılayıcılarda çıktı fonksiyonu doğrusal bir fonksiyondur. Diğer bir deyişle ağı gönderilen örnekler iki sınıf arasında paylaştırılarak iki sınıfı birbirinden ayıran doğru bulunmaya çalışılır.



Şekil 3.15. Tek katmanlı ağ yapısı

Ağın çıktısı, ağırlıklandırılmış girdi değerlerinin eşik değeri ile toplanması sonucu bulunur. Bu girdi değeri aktivasyon fonksiyonundan geçirilerek ağın çıktısı hesaplanır. T.K.Y.S.A. temel olarak doğrusal özellik gösteren olayları çözmekte başarılı iken doğrusal özellik göstermeyen olayların çözümlenmesinde yetersiz kalmaktadırlar.

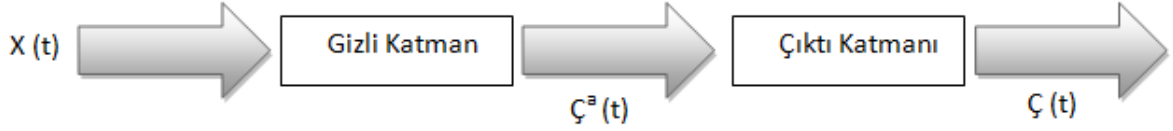
### Çok Katmanlı Yapay Sinir Ağları

Bu ağ modeli özellikle mühendislik uygulamalarında en çok kullanılan sinir ağ modelidir. Bir çok öğretim algoritmasının bu ağı eğitmede kullanılabilir olması, bu modelin yaygın kullanılmasının sebebidir.

White (1990), günümüzde en çok bilinen ve yaygın biçimde kullanılan yapay sinir ağı sınıflandırma yönteminin Çok Katmanlı Yapay Sinir Ağı (Ç.K.Y.S.A.) olduğunu ifade etmektedir. Genel amaçlı, esnek ve çoklu katmanlar halinde organize edilmiş nöronlardan oluşan doğrusal olmayan modellerdir. Ç.K.Y.S.A. hemen hemen her fonksiyonu tahmin etme/yaklaşım özelliğine sahip bir ağdır. Özellikle tanıma sistemlerinde kullanılmaktadır.

Ç.K.Y.S.A. modeli, bir giriş, bir ya da daha fazla gizli katman ve bir de çıkış katmanından oluşur. Bir katmandaki bütün işlem elemanları bir üst katmandaki bütün işlem elemanlarına bağlıdır.

Şekil 3.16.'da İ.B.Y.S.A. için blok diyagram gösterilmiştir. İleri beslemeli ağlara örnek olarak Ç.K.A. ve Öğrenme Vektör Nicelendirme ağları verilebilir.



Şekil 3.16. İleri beslemeli ağ için blok diyagram

Burada;

$X(t)$  :  $t$  nöronlu girdi katmanı,

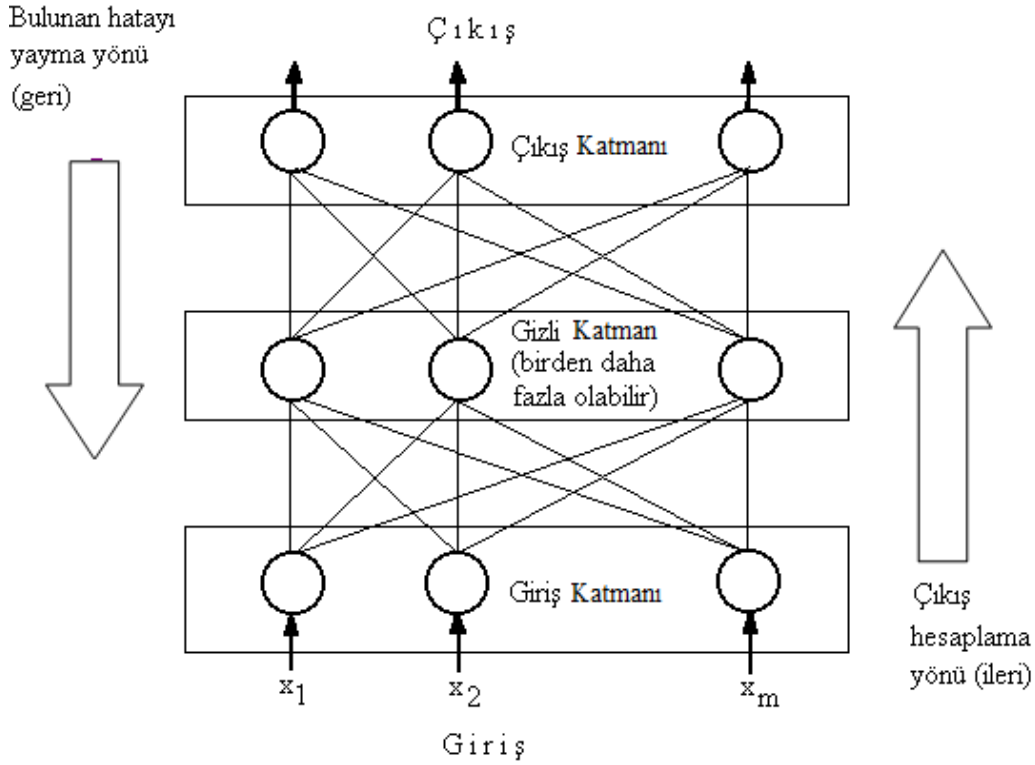
$Ç^a(t)$  :  $t$  nöronlu gizli katman,

$Ç(t)$  :  $t$  nöronlu çıktı katmanına ait çıktıları göstermektedir.

Anlaşılması kolay ve matematiksel olarak ispatlanabilir olmasından dolayı Ç.K.Y.S.A.'larının eğitiminde Geri Yayılım Algoritması (G.Y.A.) kullanılmaktadır.

G.Y.A., daha önce de bahsedildiği gibi danışmanlı öğrenme yapısına sahip ve birçok uygulamada kullanılan öğrenme algoritmasıdır.

İleri beslemeli, danışmanlı öğrenme kuralını uygulayan geri yayılım öğrenme algoritmasını kullanan örnek bir Ç.K.Y.S.A. modeli Şekil 3.17.'de verilmiştir.



Şekil 3.17. Geri yayılım Ç.K.Y.S.A. yapısı

Ç.K.Y.S.A.'larında girdi katmanında herhangi bir bilgi işleme yapılmaz. Buradaki işlem elemanı sayısı tamamen uygulanan problemin girdi sayısına bağlıdır. Gizli katman sayısı ve gizli katmanlardaki işlem elemanı sayısı ise, deneme-yanılma yolu ile bulunabilir. Çıkış katmanındaki eleman sayısı ise yine uygulanan probleme göre deneme-yanılmayla belirlenebilir.

Ç.K.Y.S.A.'larının zaman serisi tahmininde kullanılabilmesi için ağın mimari yapısının belirlenmesi gerekmektedir. Ağ mimari yapısının belirlenme süreci, ağın kaç katmandan oluşacağını, her katmanda kaç işlem elemanı bulunacağını, bu işlem elemanlarının hangi transfer fonksiyonuna sahip olacağını belirlenmesi gibi işlemlerini kapsar. Kaç dönemlik tahmin yapılacağına bağlı olarak çıktı sayısı da belirlenir.

Ç.K.Y.S.A.'larında, danışmanlı öğrenme kuralı uygulanarak ağa bir örnek gösterilir ve örnek neticesinde nasıl bir sonuç üreteceği de bildirilir. Örnekler girdi katmanına uygulanır, ara katmanlarda işlenir ve çıkış katmanından da çıktılar elde edilir.

Ç.K.Y.S.A.'nın çalışma mekanizması aşağıdaki adımlarla özetlenebilir. Bunlar;

**ADIM 1.** Örneklerin toplanması (veri toplama, birleştirme, temizleme ve dönüştürme işlemleri neticesinde verilerin analiz ve modelleme aşamasına uygun hale getirilmesi hedeflenmiştir.)

**ADIM 2.** Ağın mimari yapısının belirlenmesi,

**ADIM 3.** Öğrenme algoritmasının belirlenmesi,

**ADIM 4.** Ağırlıkların başlangıç değerlerinin atanması,

**ADIM 5.** Öğrenme seti örneklerin seçilmesi ve ağa gösterilmesi,

**ADIM 6.** Girdilerin nöronlar tarafından doğrusal olmayan fonksiyon ile dönüştürülmesi,

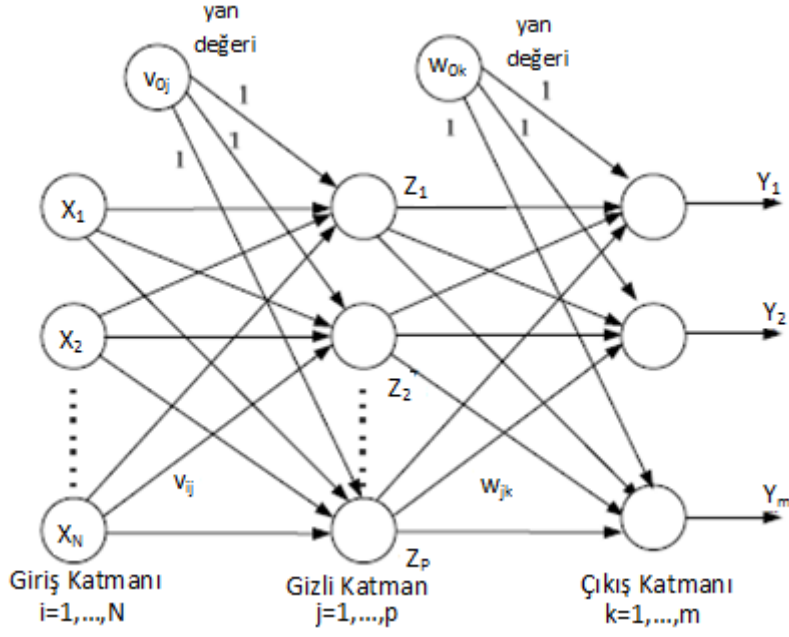
**ADIM 7.** Dönüştürülen verilere uygun parametrelerin belirlenmesi,

**ADIM 8.** Tahmin edilen çıktı ile hedeflenen çıktı arasındaki farkın karesi belirli bir minimum değere ulaşmaya kadar parametrelerin yeniden belirlenmesi,

**ADIM 9.** Ağırlıkların değiştirilmesi,

**ADIM 10.** Tahmin edilen çıktıların bu çıktılara karşılık gelen parametrelerle birlikte elde edilmesi.

Şekil 3.18.'de Ç.K.İ.B.Y.S.A.'nın genel yapısı verilmiştir.



Şekil 3.18. Ç.K.İ.B.Y.S.A. genel yapısı

Şekilde belirtilen Y.S.A. yapısı incelendiğinde;

$N$  : girdi nöron sayısı,

$p$  : gizli nöron sayısı,

$m$  : çıktı nöron sayısı,

$X_i$  : girdi nöronları,  $i = 1, \dots, N$

$v_{ij}$  :  $i$ . girdi katmanındaki  $i$  nöronu ile gizli katmandaki  $j$  nöronu arasındaki ağırlık değeri,  $i = 1, \dots, N$   $j = 1, \dots, p$

$Z_j$  : gizli katman nöronları,  $j = 1, \dots, p$

$w_{jk}$  :  $j$  gizli katmanındaki  $j$  nöronu ile çıkış katmandaki  $k$  nöronu arasındaki ağırlık değeri,  $j = 1, \dots, p$   $k = 1, \dots, m$

$Y_k$  : çıktı nöronlarıdır.  $k = 1, \dots, m$

İleri beslemeli yapay sinir ağında her katmanda ne kadar nöron olacağına aşağıdaki basit kurallara göre karar verilebilir:



Öncelikle girdi katmanı için nöron sayısına sistemin girdisi olan verinin sayısına göre kolayca karar verilebilir. Örneğin, sistemin öğrenmesi ve daha sonra sınıflandırılması istenilen örüntünün kaç veri ünitesinden oluştuğuna göre, girdi katmanındaki sinir hücresi sayısı belirlenebilir. Kısaca girdi katmanındaki her sinir hücresi, sonucu etkilemesi istenen bir değişkene karşılık gelmektedir.

Benzer uygulama çıkış katmanı için de kullanılabilir. Buna göre çıktı bilgisinin nasıl gösterilmesi istendiğine karar verildikten sonra bu çıktıda bulunması istenen her değişken için bir nöron bulundurulması gerekir. Tezin ilerleyen bölümlerinde bu işlemlerin nasıl yapılacağı detaylı olarak açıklanacaktır.

Herhangi bir problemi çözmek amacıyla kullanılan Y.S.A. da, katman sayısı ve gizli katmandaki hücre sayısı gibi kesin belirlenememiş bilgilere rağmen nesne tanıma ve sinyal işleme gibi alanların yanı sıra İ.B.Y.S.A., sistemlerin tanınması ve denetimi ile örüntü tanıma yaygın olarak kullanılmaktadır. Örneğin bir sınıflandırma problemi için çıkış katmanında farklı sınıfların gösterilmesini sağlamaya yeterli miktarda sinir hücresi bulunması yeterlidir. Ya da bir filtreleme problemi için girdi ve çıktındaki nöronların sayısı genelde eşit olur.

### **3.6. Öğrenme Algoritmalarına Göre Y.S.A.**

Y.S.A.'nın eğitilmesi kadar, eğitilmesinde kullanılacak yöntemin de önemi büyüktür. Y.S.A.'nın, veri yapısındaki ilişkiyi öğrenmesi, probleme ait örnekler yardımı ile ağ ağırlıklarının en uygun değerlerinin belirlenmesine dayanır.

Y.S.A.'da öğrenme, deneme-yanılma yoluyla tekrarlamalar sonucunda istenilen değer ile çıktı değeri arasındaki farkın minimum olması esasına dayanmaktadır. Y.S.A.'nın eğitimi için birçok değişik optimizasyon yöntemi bulunmakla birlikte optimum sonucu garanti eden genel bir algoritma bulunmamaktadır. Yapay sinir ağlarında bilinen ve yaygın olarak kullanılan belli başlı öğrenme algoritmaları aşağıda yer almaktadır:

- Geri Yayılım Öğrenme Algoritması (Genelleştirilmiş Delta Kuralı),
- Hebb Öğrenme Kuralı,
- Hopfield Öğrenme Kuralı,

- Kohonen Öğrenme Kuralı,
- Levenberg – Marquardt Öğrenme Algoritması.

### **Geri Yayılım Öğrenme Algoritması**

Günay vd., (2007a)'de, karmaşık verilerin sınıflandırılmasında kullanılan Y.S.A. modellerinden birisinin; ilk olarak Werbos tarafından düzenlenen daha sonra Parker, Rummelhart ve McClelland tarafından geliştirilen geri yayılım ağı olduğunu belirtmiştir. Geri Yayılım Öğrenme Algoritması (G.Y.Ö.A.), genelleştirilmiş delta kuralı olarak da bilinir.

Birçok uygulamada kullanılan en yaygın öğrenme algoritmasıdır. Aydoğmuş ve Çöteli (2005), bu algoritmanın hataları geriye doğru çıkıştan girişe azaltmaya çalışmasından dolayı geri yayılım ismini aldığını belirtmektedir. Bu algoritma ile, i. ve j. kat işlem elemanları arasındaki ağırlıklardaki  $\Delta W_{ij}$  değişikliği hesaplanır. Herhangi bir ağırlığın yeni değeri, eski değeri ile ağırlık değerlerinin değişim miktarının ( $\Delta W_{ij}$ ) toplamı olarak belirlenir ( $w_{yeni} = w_{eski} + \Delta W_{ij}$ ). Ara katmanlardaki işlemci elemanlar için herhangi bir hedef çıktı yoktur. Ara katmandaki düğümlerin her biri toplam hatanın sadece hesaplanan bir kısmını içerir. Bu süreç, her katmandaki düğümler toplam hatanın belirli bir kısmını içerecek şekilde girdi katmanına kadar tekrarlanır. Elde edilen hata sinyalleri temel alınarak, bağlantı ağırlıkları her düğümde yeniden düzenlenir. Bu düzenleme tüm verilerin kodlanabileceği bir duruma ağırlıkların yakınsamasını sağlar. Daha sonra bütün bağlantılar için ağırlıkların güncelleştirilmesi gerçekleştirilir. Geri yayılım algoritmasında eğitime rastgele bir ağırlık kümesi ile başlanır. Birçok uygulamada ağırlıkların başarısının ağırlıkların atanmış ilk ağırlık değerlerinin uygun seçilmesine bağlı olduğu Nabiyev (2003)'te belirtilmiştir. Geri yayılım, bir katmandaki hücreler arasında olduğu gibi katmanlar arasındaki hücreler arasında da olabilir. Bu yapı ile geri beslemeli Y.S.A., doğrusal olmayan dinamik bir davranış gösterir. Dolayısıyla, geri beslemenin yapılış şekline göre farklı yapıda ve davranışta geri beslemeli Y.S.A. yapıları elde edilebilir.

Tipik geri yayılım ağı, daima; bir giriş, bir çıkış ve en az bir gizli katmana sahiptir. Geri yayımlı ağ modelinde, problemin özelliklerine göre gizli katman sayısını

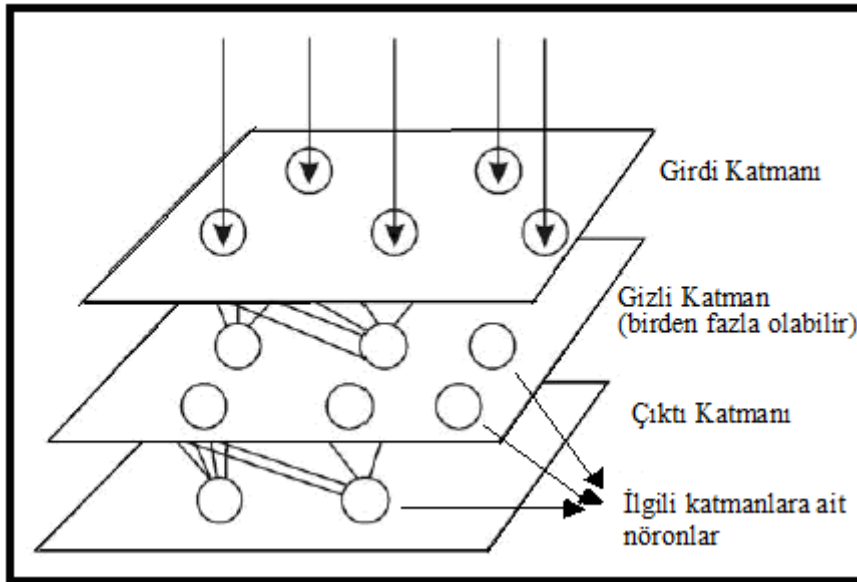
artırabilmek mümkündür. Gizli katmanların sayısında teorik olarak bir sınırlama yoktur. Genel olarak bir ya da iki tane bulunur. Geri yayılım algoritması, gradyan azalan ve Ç.K.Y.S.A.'ları eğitmede en çok kullanılan temel bir algoritmadır.

Girdi katmanı; girdi veri kümelerinin ağıya sunulduğu terminallerdir. Bu katmanda nöron sayısı, girdi veri sayısı kadardır ve her bir girdi nöronu bir veri alır. Burada veri, işlenmeden bir sonraki katman olan gizli katmana geçer.

Gizli katman; ağın temel işlevini gören katmandır. Bazı uygulamalarda ağda birden fazla gizli katman bulunabilir. Gizli katman sayısı ve katmandaki nöron sayısı, probleme göre değişir. Bu katman; girdi katmanından aldığı ağırlıklandırılmış veriyi probleme uygun bir fonksiyonla işleyerek bir sonraki katmana iletir.

Çıkış katmanı; Y.S.A.'nın en uç katmanıdır. Gizli katmandan aldığı veriyi ağın kullandığı fonksiyonla işleyerek çıktısını verir. Çıkış katmanındaki nöron sayısı, ağıya sunulan her verinin çıktı sayısı kadardır. Bu katmandan elde edilen değerler Y.S.A.'nın söz konusu problem için çıktı değerleridir.

Y.S.A.'da yer alan katmanlar Şekil 3.19.'da gösterilmiştir.



Şekil 3.19. Y.S.A.'da yer alan katmanlar

Eđitme iřlemi ve eđitimden sonraki test iřlemi bu akıřa gre yapılır. Y.S.A.'nın eđitimi iin birok deđiřik optimizasyon yntemi bulunmakla birlikte optimum sonucu garanti eden genel bir algoritma bulunmamaktadır.

G.Y..A., đrenmesi ve uygulaması kolay bir modeldir. Kaastra ve Boyd (1996) gnmzdeki uygulamaların yaklaşık % 80'inde kullanıldıđı tahmin edilen G.Y..A.'nın, zellikle mhendislik alanında neredeyse her trl problemin zm iin kullanılabilir kadar gl olduđu belirtmektedir. G.Y..A., finansal zaman serilerinin gelecekteki davranıřlarını tahmin etmede de sıklıkla tercih edilmektedir.

Geri yayılım ađındaki đrenme ařađındaki adımlardan oluřur:

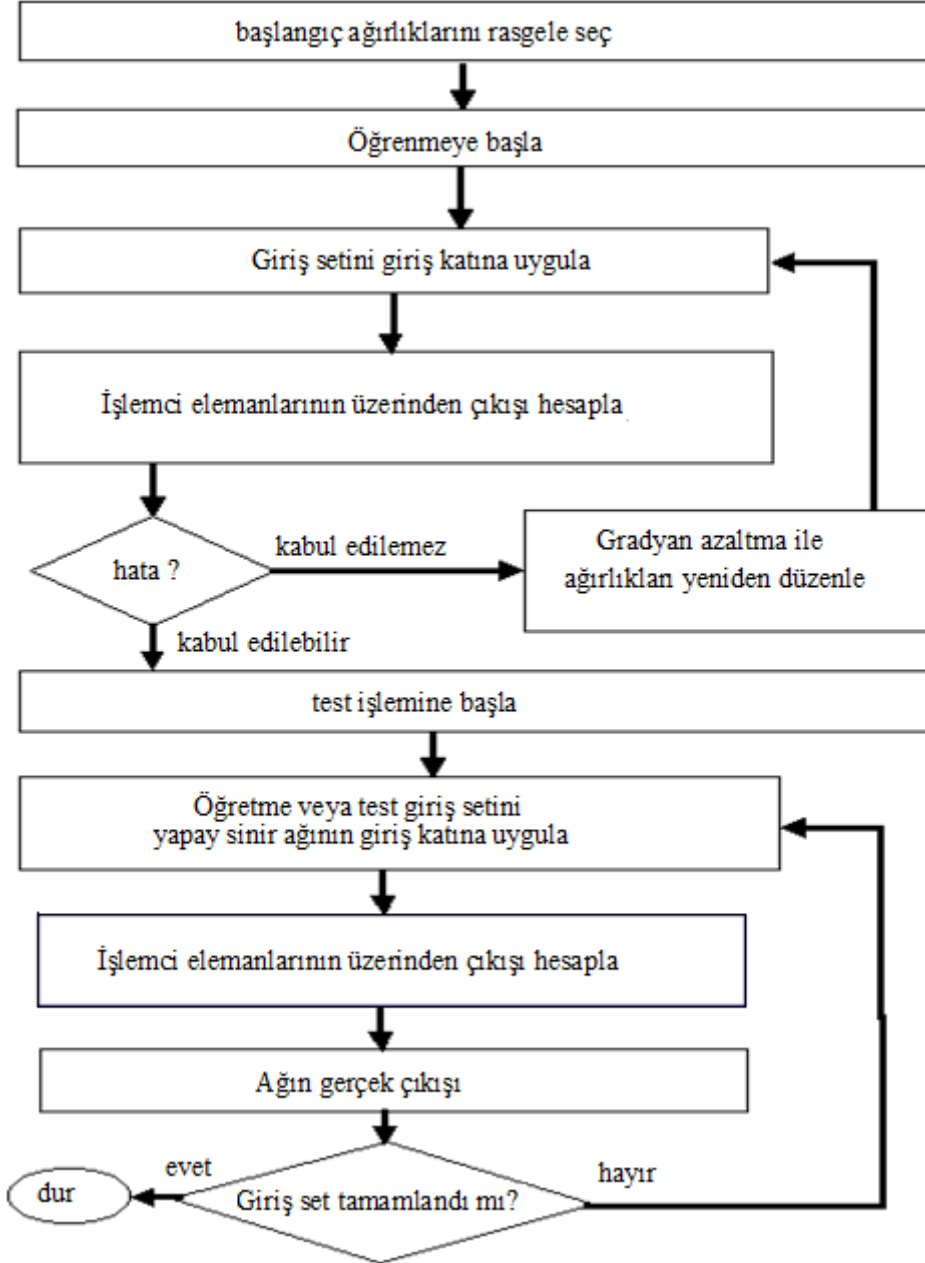
**ADIM 1.** Eđitim kmesinden bir sonraki rneđi seme ve ađ giriřine girdi vektr uygulama,

**ADIM 2.** Ađın ıktısını hesaplama,

**ADIM 3.** Ađın ıktısı ile istenen hedef vektr arasındaki hatayı hesaplama,

**ADIM 4.** Hatayı kltecek řekilde ađın ađırlıklarını ayarlama.

ok katmanlı ileri beslemeli Y.S.A. modelindeki geri yayılım řekil 3.20.'de verilmiřtir.



Şekil 3.20. Ç.K.İ.B.Y.S.A.'da geri yayılım akış şeması

### Standart Geri Yayılım Öğrenme Algoritması

Standart Geri Yayılım Öğrenme Algoritması (S.G.Y.Ö.A.), birçok pratik uygulama için çok geri kalmaktadır. S.G.Y.Ö.A.'daki en büyük problem, çok uzun eğitim süresine sahip olmasıdır. Pratik problemlerde temel S.G.Y.Ö.A.'nın kullanılması, Y.S.A. eğitiminin günlerce hatta haftalarca sürebileceğinden dolayı mümkün değildir. Bu sebeple algoritmada yakınsamayı hızlandırmak için bazı yöntemler geliştirilmiştir.

## **Standart Geri Yayılım Öğrenme Algoritmasının Dezavantajları**

Çok kullanılmasına karşın birçok istenilmeyen yönleri vardır. Bunlardan en önemlisi hata fonksiyonunu minimize ederken iterasyon sürecinin çok yavaş işlemesidir. Dolayısıyla istenilen doğrulukta bir hata düzeyine erişmek için kullanıcının uzun bir zaman beklemesi gerekmektedir. Bunun nedeni ise, Minai ve Williams (1990) tarafından da vurgulandığı gibi, S.G.Y.Ö.A. metodunun yalnızca yerel gradyan kullanması nedeniyle,  $w$  ağırlık uzayının istenmeyen bölgelere sıçramayı engellemek için öğrenme adımının (oranının) çok küçük seçilmesinden kaynaklanmaktadır. Eğer öğrenme adımı büyük seçilirse, ağırlık değişimleri salınımlı ve kararsız olacaktır.

S.G.Y.Ö.A.'sı ağ boyunca gradyan hesaplamalarını geri doğru yapar. En basit geri yayılım öğrenme algoritması gradyan azalma algoritmasıdır. Bu algortmada ağırlıklar, performans fonksiyonunun azalması yönünde ayarlanır. Fakat bu yöntem, pek çok problem için çok yavaş kalmaktadır.

## **Momentum Geri Yayılım Öğrenme Algoritması**

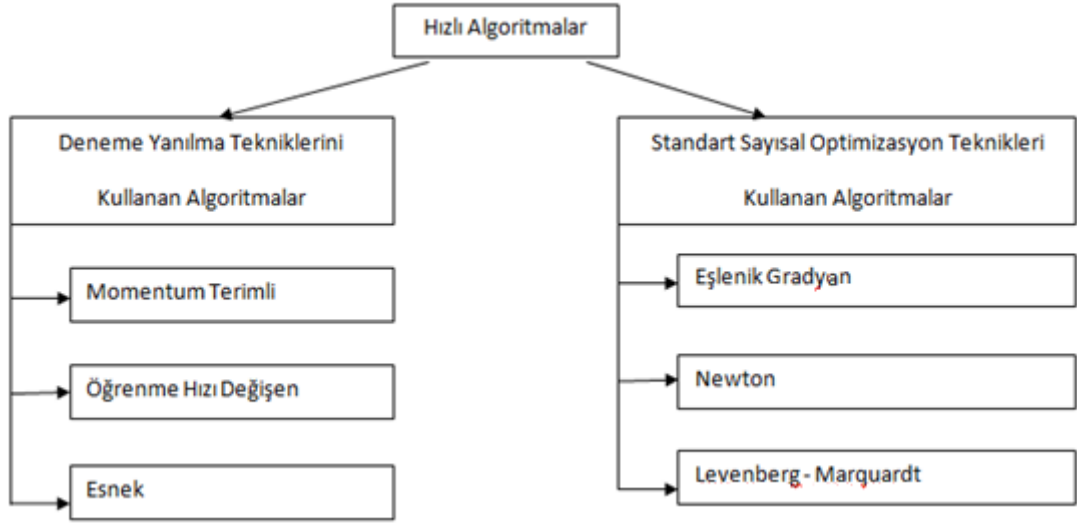
Çok katmanlı Y.S.A.'da bir başka geri yayılım yöntemi olan Momentum Geri Yayılım Öğrenme Algoritması, aslında işleyiş bakımından S.G.Y.Ö.A.'dan çok büyük farklılıklar barındırmamaktadır. S.G.Y.Ö.A.'na ek olarak ağırlık katsayılarının güncellenebilmesi için alfa ( $\alpha$ ) ve beta ( $\beta$ ) gibi reel sabit değerlere gereksinim duyulmaktadır. Algoritmayı S.G.Y.Ö.A.'dan ayıran en önemli özellik ise ağırlık katsayısı güncellemesi yapılırken iki iterasyon öncesine ait ağırlık katsayısı değerlerinin de kullanılmasıdır.

## **Levenberg-Marquardt Öğrenme Algoritması**

Levenberg-Marquardt Öğrenme Algoritması (L.M.Ö.A.) son yıllarda kullanılan en popüler ve diğer algoritmalara alternatif olarak geliştirilmiş algoritmalarındandır. Şekil 3.21.'de hızlı ve yüksek performanslı algoritmaların 2'ye ayrıldığı görülmektedir. Bunlar;

1. Deneme yanılma tekniklerini kullanan algoritmalar,

2. Standart sayısal optimizasyon yöntemlerini kullanan algoritmalarıdır. L.M.Ö.A. da bu kategori içerisinde yer almaktadır.



Şekil 3.21. Hızlı ve yüksek performanlı algoritmalar

L.M.Ö.A.'na göre, hata değeri bulunduktan sonra nöronlar kendi hatalarını azaltmak için ağırlıklarını ayarlamak durumundadır. Ağırlık değiştirme denklemleri de ağdaki performans fonksiyonunu en küçük yapacak şekilde düzenlenirler. Diğer bir deyişle bu algoritmaya, maksimum komşuluk fikri üzerine kurulmuş bir en küçük kareler tahmin metodu da denilebilir.

Sağiroğlu (1996), L.M.Ö.A., Gauss-Newton tekniğinin ve En hızlı iniş metodunun en iyi özelliklerini birleştirerek bu algoritmaların sınırlamalarının çoğundan kaçınmıştır. L.M.Ö.A.'nın yakınsama hızı genellikle yüksektir.

Özellikle ileri beslemeli ağlarda kullanılan öğrenme algoritmaları, performans ölçütünü en küçük yapacak ağırlıkları ayarlayabilmek için, performans fonksiyonunun türevini kullanırlar.

L.M.Ö.A.'da, Newton ve Gradyan Azalması algoritmalarının en iyi özelliklerinden oluşur.

Eşlenik gradyan öğrenme algoritmasına alternatif olarak sunulan Newton yöntemlerinde temel adım Hessian matrisini elde etmektir. Eş. 3.12.'de yer alan hessian matrisi, performans fonksiyonunun ağırlıklara göre ikinci dereceden

türevlerinden oluşan bir matristir. Hessian matrisi, ağırlık uzayının farklı doğrultularındaki gradyan değişimini gösterir.

$$H(n) = \frac{\partial^2 E(n)}{\partial w^2(n-1)} \quad (3.12.)$$

Burada;

$H(n)$  : Hessian matrisi,

$E(n)$  : performans fonksiyonu,

$w(n)$  : ağırlık matrisi (Ağın sinaptik ağırlığı),

$n$  : iterasyon sayısıdır.

Performans fonksiyonu, duruma göre toplam ani hata ya da ortalama karesel hata olarak alınabilir. Bu çalışmada, ileri beslemeli ağların çoğunda olduğu gibi, performans fonksiyonu olarak, hata kareler ortalaması kullanılmıştır.

Hessian matrisi hesaplandıktan sonra, tersi bulunarak ağırlıklar yenilenebilir. Ancak hessian matrisi çok karmaşık ve ileri beslemeli Y.S.A. için hesaplanması zor bir matristir. Newton yöntemlerinin içinde, ikinci dereceden türevlerin hesaplanmadan işlem yapılan bir sınıf vardır. Bu sınıftaki yöntemler, Quasi-Newton yöntemleri olarak adlandırılırlar. Quasi-Newton yöntemleri, algoritmanın her iterasyonunda, Hessian matrisinin yaklaşık bir şeklini kullanır.

L.M.Ö.A.'da, Quasi-Newton yöntemleri gibi, Hessian matrisinin yaklaşık değerini kullanır. L.M.Ö.A. için Hessian matrisinin yaklaşık değeri Eş. 3.13. yardımıyla bulunur.

$$H(n) = J^T(n) J(n) + \mu I \quad (3.13.)$$

Burada;

$\mu$  : marquardt parametresi,

$I$  : birim matris,



$J(n)$  : jakobiyen matris,

$J^T(n)$  : jakobiyen matrisin tersi,

$n$  : iterasyon sayısı.

$\mu$  Marquardt parametresi skaler bir sayıdır. Eğer  $\mu$  sıfırsa, bu yöntem yaklaşık Hessian matrisini kullanan Newton algoritması; eğer  $\mu$  büyük bir sayı ise, küçük adımlı gradyan azalması yöntemi haline gelir. Her iterasyonda H.K.O. azaldığı için, Newton yöntemlerinde en küçük hataya daha hızlı ulaşılır.

Eş. 3.14.'te yer alan  $J(n)$  Jakobiyen matrisi, ağ hatalarının ağırlıklara göre birinci dereceden türevlerinden oluşur:

$$J(n) = \frac{\partial e(n)}{\partial w(n-1)} \quad (3.14.)$$

Burada;

$e(n)$  : ağ hataları vektörü,

$J(n)$  : jakobiyen matrisi,

$w(n)$  : ağırlık matrisidir (Ağın sinaptik ağırlığı).

Jakobien matrisi, hesaplamada Hessian matrisinden daha kolay olduğu için tercih edilir. Ağın gradyanı Eş. 3.15.'de verilen şekilde hesaplanır.

$$g(n) = J^T(n) e(n) \quad (3.15.)$$

Algoritmada ağırlıklar Eş. 3.16.'da belirtilen denkleme göre değiştirilir:

$$w(n+1) = w(n) - [H(n)]^{-1} g(n) \quad (3.16.)$$

Haykin (1994), her başarılı adımda yani performans fonksiyonunun azaltılmasında  $\mu$  değeri azalırken, hataların karelerinin toplamının arttığı adımlarda  $\mu$  değerinin artırıldığını belirtmiştir. Bu yöntemde  $\mu$  parametresi hesaplama süresince otomatik olarak güncellenmekte ve böylece yavaş yakınsama probleminden etkilenmediği

için hızlı bir yaklaşım elde edilmektedir. Yaklaşım oranının hızlanması ile eğitim kısa sürmektedir. Ancak fazla işlem yapmakta ve sonucunda çok fazla hafıza kullanmayı gerektirmektedir.

Bolat ve Kalenderli (2003), genel olarak L.M.Ö.A.'nın yavaş yakınsama probleminde etkilenmediğini ifade etmişlerdir. Burada hedef, performans fonksiyonunu en küçük yapacak ağırlık değerini bulmaktadır.

### **Hebbian Öğrenme Algoritması**

Hopfield Ağı, Hebbian Öğrenme Algoritmasını uygular. Bu şekilde bir öğrenme ilk defa Hebb tarafından ifade edilen sinaptik modülasyon yönteminin matematiksel bir çıkarımıdır.

Hebb'in kuralına göre, eğer alıcı nöron harekete geçerken bir nöron başka bir nöronu uyarıyorsa, iki hücre arasındaki bağlantı ağırlıklandırılır.

Matematiksel olarak:  $w_{ij} = a_i a_j$  şeklinde ifade edilebilir.

Bir ağırlıktaki değişim, bağlandığı birimlerin aktivasyonları çarpımına eşittir. Böylelikle, eğer iki birim de açıksa ( $a_i = 1$ ) ya da her iki birim de kapalıysa ( $a_i = -1$ ) ağırlığın gücü artar, aksi halde azalır. Hebb kuralının matematiksel tanımına göre ağırlık değeri, eğer her iki birimin aktivasyonları kapalıysa artar; birimlerin aktivasyonları 1 ve  $-1$  ise azalır.

$a_i = 1$  ya da  $a_i = -1$  olması durumu fiziksel olarak açıklanamaz ve değiştirilemez. Bununla birlikte sisteme eğer Hebbian kuralı uygulanmışsa anlaşılması daha kolay olur.

Eğer birimler dizisine bir deseni anında verip üstteki kuralı uygularsak, o desen ağın hafızasına aldığı, öğrendiği desen haline gelir. Bu da Hopfield ağının ilgilendiği Hebbian Öğrenme Algoritması'nın önemli bir özelliğidir.

Sonuç olarak, eğer ağın aktivasyonunun deseni, ağın öğrendiği bir desene yakınsa, o desene doğru gitmeye çalışacaktır.

### 3.7. Y.S.A. ile Modelleme

Y.S.A.'nın modellenmesi anlatılmadan önce model ve modellemenin tanımlamaları yapılmalıdır. Öner (1998), sistemin ve sürecin en iyi şekilde anlaşılmasını sağlayacak ve bunu gerçekleştirecek temsilini "model", bu modeli gerçekleştirme ortaya çıkarma sürecini de "modelleme" olarak tanımlamıştır.

Modellemeyi gerçekliğin basit gösterimi ya da soyutlaması olarak tanımlayabiliriz. Daniel (1995)'e göre modelleme süreci ile gerçek sistem ya da durumun, bir model aracılığı ile en iyi şekilde temsil edilmesi gerçekleşir. Gerçek durumun tam olarak modellenmesi çok zordur fakat istenilen sonuca ulaşmak için modeller geliştirilir.

Modelleme işleminde kullanılan iki temel yaklaşımdan ilki, kümelendirilmiş parametre modellemesi, ikincisi ise sistem tanımadır. Kümelendirilmiş parametre modellemesi yaklaşımında, sistem, girdi-çıkıtı ilişkisi basitçe ifade edilebilen bileşenlerle yapılandırılmaya çalışılır. Sistem tanıma yaklaşımında ise deneysel olarak elde edilmiş ya da hipotetik olarak üretilmiş girdi-çıkıtı verilerinin kullanılarak sistemin matematik modelinin kurulmasına çalışılır.

Modelin kurulabilmesi için üç temel gereksinim vardır. Bunlar;

1. Girdi-çıkıtı verisi,
2. Model adaylarının belirlenmesi,
3. Modelin seçim kriteridir.

Gelecekteki verilerin istatistiksel özelliklerinin geçmiştekilerle uyumlu olacağı varsayımından hareketle, modellemeye konu olan sisteme ait geçmişe dönük verilerin istatistiksel özelliklerinden yararlanılarak kurulan ve zamana bağımlı modele "zaman-serisi modeli" denir. Modelin, geçmişteki verileri yeterli doğrulukta sağlaması kadar gelecekte tekrarlanma ihtimali yüksek özellikleri tanımlaması da gereklidir.

Özellikle, geleceğe dönük kestirim zordur. Sistemin karmaşıklığı ya da hedefin hassaslığına bağlı olarak, tasarlanan modellerin yoğun işlem gerektiren algoritmalarının geçerliliği, gelişen teknolojinin harikası ve vazgeçilmez unsuru bilgisayarların elverdiği kolaylıkla ve hızla sınanabilmektedir. Sınamayı başarıyla

geçen modeller öngörü işleminde kullanılmaktadır. Başarılı modellerin sahip olması istenilen diğer özellikler:

- Parametre sayısı asgariye indirilmeli,
- Parametrelerin kestirimi kolay olmalı,
- Parametreler anlamlıca yorumlanabilir olmalıdır.

### **Y.S.A. İçin Yapılması Gerekenler**

Y.S.A. ile öngöründe bulunmak için aşağıda belirtilen faktörlerin belirlenmesi gerekmektedir. Her problem için geçerli olacak genel bir Y.S.A. modeli belirlenememiştir. Bu nedenle iyi öngörü sonuçları elde edebilmek amacıyla her bir problem için tüm faktörlerin gözden geçirilmesi gerekmektedir.

1. Verilerin hazırlanması, bir ön işlemden geçirilmesi [(0,1) aralığına dönüştürme)],
2. Eğitim ve test verilerinin ayrıştırılması,
3. Y.S.A.'nın tasarlanması,
  - a. Gizli katman sayısının belirlenmesi,
  - b. Gizli nöron sayısının belirlenmesi,
  - c. Çıktı sayısının belirlenmesi,
4. Aktivasyon fonksiyonuna karar verilmesi,
5. Performans ölçütünün hesaplanması,
6. Y.S.A.'nın eğitilmesi,
7. Uygulama.

### **Modelleme Adımında Yapılacak İşlemler**

#### **Girdi Nöron Sayısı**

Girdi nöron sayısını belirlemek için kullanılan genel bir kural yoktur. Deneme-yanılma ile girdi nöron sayısının belirlenmesi sıklıkla kullanılan bir yöntemdir. Bu katmanda nöron sayısı , girdi veri sayısı kadardır ve her bir girdi nöronu bir veri alır. Burada veri, işlenmeden bir sonraki katman olan gizli katmana geçer.

## **Gizli Katman Sayısı**

Gizli katman ađın temel işlevini gören katmandır. Bazı uygulamalarda ađda birden fazla gizli katman bulunabilir. Gizli katman sayısı ve katmandaki nöron sayısı, probleme göre deđişir. Tamamen ađ tasarımcısının kontrolündedir ve onun tecrübesine bađlıdır.

Y.S.A.'da hiç gizli katman bulunmuyorsa ya da girdi katmanındaki birimler ile çıktı katmanındaki birimler arasında doğrusal bađlantı varsa, bu durumda girdilerin bir kısmının çıktılara direk bađlı olduđu anlaşılır. Hangi girdinin hangi çıktıya bađlı olduđuna göre de bir sonuca varılır. Gizli katman bulunmaması durumunun en klasik örneđi, yerine koyma şifrelemesi olarak gösterilebilir. Bu şifreleme yönteminde her harf başka bir harf ile yer deđiştirecektir ve her girdi bir çıktıya bađlıdır.

Y.S.A.'da tek katman bulunması durumu genelde problemlerin büyük çođunluđunda karşılaşılan bir durumdur ve kısaca girdi ve çıktı arasında sonlu bir kümeden yine sonlu bir kümeye bađlantı bulunması durumunda kullanılır.

Y.S.A.'da birden fazla katmanın kullanılması durumu çok sık rastlanan bir durum deđildir. Genelde kesin sonuç beklenmeyen ama bir deđere yaklařması istenen problemlerde kullanılan iki katmanlı çözümler, katman sayısı arttıkça karmaşıklaşmakta ve hem tasarımı hem de analizi zorlaşmaktadır.

## **Gizli Katmandaki Nöron Sayısı**

Y.S.A.'nın en kritik noktası gizli katmanında kaç nöron olacađına karar vermektir. Chowdary (2007), genellikle sonucu direk olarak vermemesine karşın sonuca direk etkisi olması açısından gizli katmandaki nöron sayısının oldukça önemli olduđunu belirtmiştir. Çünkü gizli katmanda bulunan nöron sayısı Y.S.A.'nın bir süreci modelleme doğruluđunu ve genelleştirme yeteneđini belirlemektedir.

Ayrıca gizli katman nöron sayısı, öğrenme sırasında bellekte fazla bilgi barındırmak haricinde öğrenme işleminin daha iyi yapılmasını da sağlamaktadır.

Çelebi ve Bayraktar (2008), gizli katmandaki nöron sayısının belirlenmesinde genel bir kuralın bulunmadığına dikkat çekmektedir. Bu nedenle, gizli katman nöron sayısı belirlenirken deneme-yanılma sık kullanılan bir yöntemdir.

Gizli katman; girdi katmanından aldığı ağırlıklandırılmış veriyi probleme uygun bir fonksiyonla işleyerek bir sonraki katmana iletir. Bu katmanda gereğinden az nöron kullanılması, girdi verilerine göre daha az hassas çıktı elde edilmesine sebep olurken, test kümesindeki verilerle doğru sonucu vermemesine sebep olabilir. Normalde iki ayrı nöron üzerinden taşınacak veri tek nöron üzerinden taşınmış olur. Bu da taşınan verinin daha karmaşık olması demektir.

Aynı şekilde gerektiğinden daha çok sayıda nöron kullanılması durumunda da aynı ağda yeni tip veri gruplarının işlenmesinde zorluklar ortaya çıkar ve uyum aşımı olur. Basitçe ifade edilirse, gizli katmanda bulunan her sinir hücresini eğitecek kadar eğitim kümesi bulunmayabilir. Diğer bir problem ise, yeterli miktarda eğitim verisi bulunması durumunda bile karşılaşılabilen geç öğrenmedir. Buna göre sistemdeki bütün sinir hücrelerinin eğitimi normalden çok daha uzun zaman almaktadır.

Lipmann (1987), gizli nöron sayıları için  $2n+1$ ,  $2n$ ,  $n$ ,  $n/2$  değerlerini ileri sürmüştür. Zhang ve Patuwo (1998) ise bu yaklaşımların bütün problemler için iyi çalışmayacağını belirtmiştir.

Gizli katmanda bulunan nöron sayısı Y.S.A.'nın bir süreci modelleme doğruluğunu ve genelleştirme yeteneğini belirlediğinden uygun nöron sayısını belirlemek için birçok çalışmada olduğu gibi bu çalışmada da deneme-yanılma metodundan yararlanılmıştır.

### **Çıktı Nöron Sayısı**

Çıktı nöron sayısı, zaman serisi tahmin problemlerinde tahmin döneminin uzunluğuna eşit alınabilir. Buna göre çalışmada kaç dönemlik tahmin yapılırsa, çıktı nöron sayısı da o kadar alınabilir. Bu yöntem doğrudan öngörü yöntemidir. Diğer bir yöntem ise iteratif öngörü yöntemi olup, ilk öngörünün veri olarak kullanılıp, bir sonraki öngörünün elde edilmesi yöntemidir. Hangi öngörü

yönteminin daha iyi olduğu konusunda, varılmış genel bir sonuç bulunmamaktadır (Günay vd., 2007a).

### **Aktivasyon Fonksiyonu**

Aktivasyon fonksiyonu yapay nöron çıktısının büyüklüğünü sınırlandıran fonksiyondur. Aynı zamanda girdi ve çıktı birimleri arasındaki eğrisel eşleşmeyi sağlar. Transfer, eşik ya da sıkıştırma fonksiyonu olarak da isimlendirilmektedir. (Mandic ve Chambers, 2001). Seçilen aktivasyon fonksiyonu, doğrusal olmadığında eğitim parametresinin belirlenmesi gerekir (Aladağ, vd., 2007).

Zaman serileri için sigmoid aktivasyon fonksiyonunun kullanılabileceği, Tebelkis (1995)'de belirtilmiştir. Çünkü hem doğrusal olmayan bir fonksiyondur; hem de her yerde türevlenebilirdir. İkili (binary) değişkenler için “adım” fonksiyonu da önerilmektedir. S.G.Y.Ö.A.'da ise çoğunlukla girdi katmanı için doğrusal aktivasyon fonksiyonu, diğer katmanlar için ise sigmoid fonksiyon kullanılır.

### **Normalleştirme İşlemi**

Saritas (2008), girdi ve çıktı verilerinin normalleştirilmesinin, yakınsama ve öğrenme işlemi açısından oldukça önemli olduğunu belirtmiştir. Bu nedenle ağ eğitilmeden önce veriler eğitim karakteristiğini iyileştirmek için Eş. 3.17. kullanılarak 0-1 değerleri arasında normalize edilir.

$$X_i' = \frac{X_i - X_{\min}}{X_{\max} - X_{\min}} \quad i = 1, \dots, N \quad (3.17.)$$

Burada,

$X_i$  : N sayıdaki veri kümesinde yer alan i.nci veri,

$X_{\min}$  : N sayıdaki veri kümesinde yer alan minimum değere sahip veri,

$X_{\max}$  : N sayıdaki veri kümesinde yer alan maksimum değere sahip veri,

$X_i'$  : N sayıdaki veri kümesinde yer alan normalleştirilmiş i.nci veriyi göstermektedir.

## **Eđitim ve Test Kumesi**

Eldeki tım verilerin ađın eđitimi iin kullanılması dođru bir yntem olmayacaktır. Eđitim, test ve geerlilik olmak üzere, veri u gruba ayrılabilir. Bu durumda eđitim kumesi iin veri kumesinden daha az bir grupla alıřılacaktır. Bundan kaınmak iin veri kumesi eđitim ve test olmak üzere ikiye de ayrılabilir. Oransal dađılım olarak da % 90 eđitim kumesi ve %10 test kumesi ya da %80 eđitim kumesi ve %20 test kumesi řeklinde bir dađılım kullanılabilir (Günay vd., 2007a).

## **Ađın Eđitimi**

Y.S.A.'nın en önemli özelliklerinden birisi, ilgili probleme ait örnekler verildiđinde girdiler ile ıktılar arasındaki iliřkiyi öğrenebilmesi; girdi ve ıktı arasında bir eřleřtirme yapmasıdır. Ađ yeterince eđitildiđinde daha önceden görmediđi girdilere karřı uygun ıktılar üretebilmektedir. Bu özellik Y.S.A.'nın genelleme kapasitesi olarak adlandırılır ve çođunlukla ađın yapısına ve eđitim iin uygulanan yineleme sayısına bađlıdır (Alkan, 2001).

Y.S.A.'da katmanlar arasındaki nöronların bađlantı ađırlık deđerlerinin belirlenmesi iřlemine "ađın eđitilmesi" denir. Bařlangıta bu ađırlık deđerleri rasgele alınır. Y.S.A.'lar, kendilerine örnekler gösterildike bu ađırlık deđerlerini yenileyerek amaca ulařmaya alıřırlar. Amaca ulařmanın ya da yaklařmanın ölçüsü dıřarıdan verilen deđerdir. Eđer Y.S.A. verilen girdi ıktı çiftleriyle amaca ulařmış ise ađırlık deđerleri saklanır. Ađırlık deđerlerinin deđiřmesi belirli kurallara göre yürütölmektedir. Bu kurallara "öđrenme kuralları" denilmektedir. Y.S.A. öğrendikten sonra daha önce verilmeyen girdiler uygulanarak ađ ıktıları gözlemlenir.

Bu alıřmada, ileri beslemeli Y.S.A. mimarisi göz önüne alınmıştır. Bunun anlamı, nöronlardan ıkan bütö bađlantılar bir sonraki katmandaki nöronlara gitmektedir ve önceki ya da aynı katmanlar arasında bađlantı mümkün olmamaktadır. Buradaki mimari, ok katmanlı Y.S.A. olarak da anılmaktadır. Tipik olarak ok katmanlı, ileri beslemeli Y.S.A., girdi katmanı, gizli katman ve ıktı katmanından oluřmaktadır (Öztemel, 2003).

Geliřtirilen Y.S.A. eđitim yntemi, birden fazla gizli katmana sahip mimariler iin de kolaylıkla uygulanabilir. özölecek olan problem tipik danıřmanlı Y.S.A. eđitimidir.



Verilen girdi vektör seti ve onlara karşılık gelen çıktı değerine karşılık, hesaplanan çıktı değerleri ve orijinal çıktı değerlerinin birbirine yaklaşmasını sağlayacak olan nöronlar arası optimum bağlantı ağırlıkları bu yöntemin ulaşmayı hedeflediği problemdir. İterasyonlar arası optimum bağlantı ağırlıkları belirli bir “beklenen çıktı ve hesaplanan çıktı” fark duyarlılığında birbirlerine yaklaştıklarında, Y.S.A.’nın öğrendiği sonucuna varılmaktadır.

Bu tür danışmanlı öğrenmeyle Y.S.A. ağırlarının eğitilmesinde, Trafalis ve Couellan (1998)’de belirtildiği üzere, klasik yaklaşımlar Werbos tarafından ortaya konulan, hatanın geri doğru yayılımını kullanmaktadırlar. Bu yaklaşım gerçek ve hesaplanan çıktı değerinin olduğu çıktı katmanındaki hata değerini geri yaymaya başlayarak ve her aşamada hatayı bir önceki katmana yayarak ağırlıkların iteratif bir biçimde güncelleştirir. Güncelleştirilmiş ağırlıklar azalan gradyan yöntemi kullanılarak türetilmektedir. Battiti (1992)’e göre, Quasi-Newton yöntemini kullanan teknikler de önerilen diğer teknikler arasındadır. Geri yayılım algoritması oldukça iyi sonuçlar verebilmekte, güç ve geçerlilik için bir kıyaslama algoritması olarak kullanılmaktadır (Trafalis ve Couellan, 1998).

Yine de, geri yayılım algoritması, uzun eğitim zamanları gerektirmekte, dolayısıyla hızlı eğitim zamanlarına ihtiyaç duyan uygulamalarda çok da uygun bir yöntem olamamaktadır.

Çok katmanlı ve danışmanlı öğrenme, Y.S.A.’larda hızlı sonuç verebilecek bir yapıya sahiptir. Böylece problem, kısıtlı doğrusal olmayan optimizasyon problemine dönüştürülmektedir. Kısıtlarda genellikle sigmoid ya da hiperbolik tanjant fonksiyonları kullanılabilir. Hangisinin kullanılacağı, çok katmanlı danışmanlı Y.S.A.’nın uygulanmasında kullanılan aktivasyon fonksiyonu ile ilgilidir. Dolayısıyla Y.S.A.’da kullanılan aktivasyon fonksiyonu doğrusal olmayan modelde aynen yansımalıdır.

En küçük karelerle yapılmış hata fonksiyonun yerine mutlak hatalar toplam fonksiyonu da kullanılabilir. Mutlak hataların toplam fonksiyonunun amaç fonksiyonu olarak kullanılması, optimum sonucu değiştirmemekle birlikte, amaç fonksiyonunun doğrusal bir forma dönüştürülmesi bakımından bir avantaj olarak görülebilir.

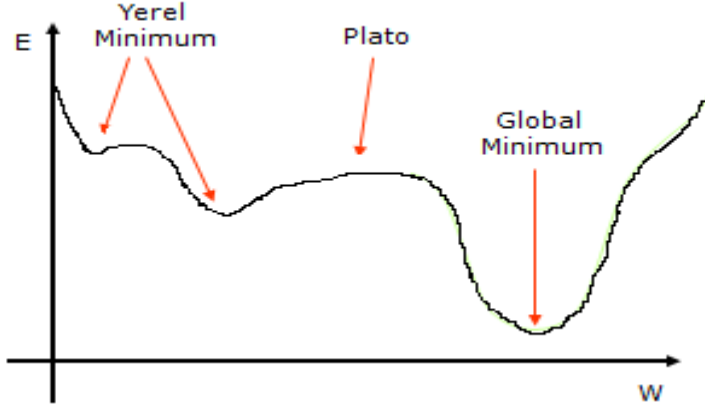
## Öğrenme Katsayısı ( $\eta$ )

G.Y.Ö.A.'da öğrenme katsayısı, ağırlıkların değişim miktarını belirler. Eğer öğrenme katsayısı gereğinden büyük olursa problem uzayında rasgele gezinme olur. Bunun da ağırlıkları rasgele değiştirmekten farkı olmaz. Eğer öğrenme katsayısı çok küçük olursa, çözüme ulaşmak daha uzun sürer. Sabit olarak 1 değerinden çok daha küçük bir değerdir. Başlangıçta büyük  $\eta$  ile başlanır ve dereceli olarak bu değer düşürülür. Ya da küçük  $\eta$  ile başlanılır ve hata artmaya başlayana kadar bu değer büyütülür. Böylece öğrenimin her bir aşamasında maksimum güvenli adım bulunur. Asıl amaç, öğrenme oranı büyürken hatanın minimumda kalmasına çalışılır.

## Momentum Katsayısı ( $\gamma$ )

G.Y.Ö.A.'da öğrenme katsayısı, momentum, öğrenme sürecini hızlandıran alternatif ağırlık ayarlama yöntemidir. Bu yöntemde  $t+1$  anındaki ağırlıklar ayarlanırken,  $t$  ve  $t-1$  anındaki ağırlıkların farkı da hesaba katılır. Momentum katsayısı  $\gamma$  ile gösterilir.

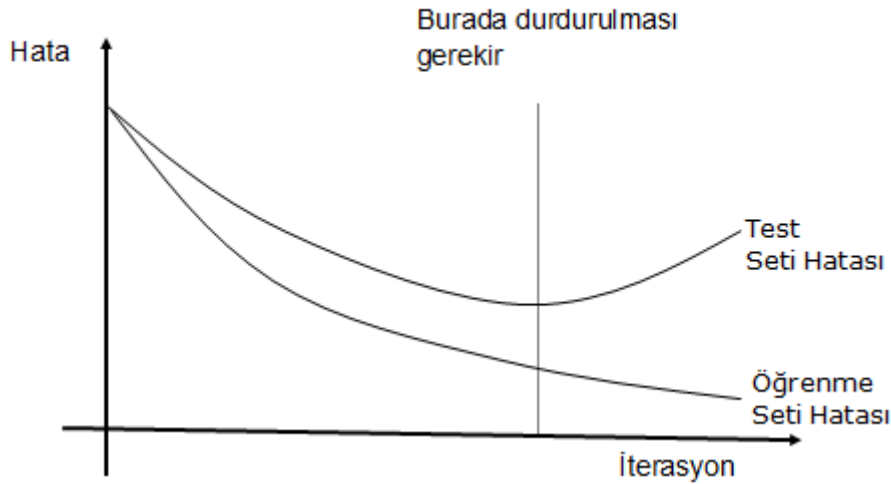
Momentum katsayısı, yerel çözümlere takılmayı önler.  $\gamma$  momentum katsayısı  $0 < \gamma < 1$  aralığında seçilir. Momentum kullanıldığında, ağırlıklara göre ardışık iki iterasyonda, aynı işaretli ise ağırlıklara uygulanacak düzeltme artar, aksi taktirde azalır. Momentum katsayısı ile Y.S.A.'nın öğrenme oranında belirli bir hızlanma elde edilir. Bu değer çok küçük seçilmesi yerel çözümlerden kurtulmayı zorlaştırır. Değer çok büyük seçilmesi ise tek bir çözüme ulaşmada sorunlar yaratabilir. Algoritma dengede iken momentum kullanarak daha büyük bir öğrenme hızı elde edilebilir. Momentumun diğer bir özelliği ise yörünge tutarlı bir yönde hareket ettiğinde yakınsamanın hızlanmasıdır. Hata belli bir eşiğin altında artmışsa momentum değeri değiştirilmez. Ağırlık parametrelerinin güncellenmesi iptal edilir. Hata azalıyorsa parametre güncellemesi yapılır. Öğrenme hızı ve momentum değeri bir miktar artırılır. Momentum değerindeki değişime göre ağırlıkların değişim miktarı Şekil 3.22.'de gösterilmiştir.



Şekil 3.22. Momentum değerindeki değişime göre ağırlıkların değişim miktarı

### Eğitimin Durdurulması

Hatanın belli bir değerin altına düşmesi sonucunda eğitim sonlandırılır ya da belirli sayıda iterasyondan sonra eğitim durdurulur. Her iki yönteminde dezavantajları olabilir. Örneğin başta belirlenen iterasyon sayısı yüksek ise ağ hatayı minimize etse dahi boşuna öğrenmeye çalışmaya devam eder. Diğer durumdaki dezavantaj ise; ağın gereğinden fazla eğitilmesiyle, problemi öğrenmek yerine verileri ezberlemeye başlamasıdır. Bu da ağın hatalı sonuçlar üretmesine neden olur. Şekil 3.23.'te eğitimin durdurulması için hata ve iterasyon sayısı gösterilmiştir. Çalışmamızın uygulama aşamasında, her veri kümesi için kullanıcı tarafından belirlenmiş bir hata değerine ulaşılması sonucu eğitim durdurulmuştur.



Şekil 3.23. Eğitimin durdurulması için hata ve iterasyon değerleri

## Ağın Test Edilmesi

Ağın eğitimi tamamlandıktan sonra öğrenip öğrenmediğini test etmek amacıyla yapılan denemeler ağın test edilmesi aşamasını oluşturmaktadır. Öztemel (2003), test aşamasında ağa öğrenme aşamasında görmediği verilerin uygulanması gerektiği belirtmiştir. Bu işlem test kümesindeki verilerin kullanılması ile gerçekleşmektedir.

## Performans Ölçütleri

En iyi Y.S.A. mimarisinin belirlenmesinde çeşitli performans ölçütlerinden yararlanılır (Aladağ vd., 2009). Y.S.A.'da genellikle H.K.O. performans ölçütü olarak tercih edilir. H.K.O. Eş. 3.18.'de belirtildiği gibi hesaplanır.

$$H.K.O. = \sum_{i=1}^N \frac{(e(n))^2}{N} \quad (3.18.)$$

Burada;

N : test kümesi büyüklüğü,

e(n) : orijinal değer ile tahmin edilen değer arasındaki farktır.

Diğer performans ölçütleri ise Eş.3.19. ile Eş. 3.22. arasındaki eşitliklerde verilmiştir.

$$\text{Ortalama Mutlak Hata (OMH)} = \sum_{i=1}^N \frac{|e(n)|}{N} \quad (3.19.)$$

$$\text{Ortalama Mutlak Yüzde Hata (OMYH)} = \left(\frac{1}{N}\right) \sum_{i=1}^N \left| \frac{e(n)}{y(n)} \right| (100) \quad (3.20.)$$

$$\text{Hata Kareler Toplamı (HKT)} = \sum_{i=1}^N (e(n))^2 \quad (3.21.)$$

$$\text{Hata Kareler Ortalaması Karekökü (H.K.O.K)} = \sqrt{HKO} \quad (3.22.)$$

Burada;

N : test kümesi büyüklüğü,

$e(n)$  : orijinal deęer ile tahmin edilen deęer arasındaki fark,

$y(n)$  :  $n$  dönemine ait gözlem deęeri,

H.K.O. : hata kareler ortalamasıdır.

## DÖRDÜNCÜ BÖLÜM

### 4. UYGULAMA

Türkiye’de bir sigorta şirketine ait kredi karşılığı verilen uzun süreli hayat sigortası portföyünün, 3 yıllık kârının bugünkü değeri yani gömülü değerini bulmak için kullanılacak parametrelerin öngörülmesi istenmektedir. Hayat sigorta şirketi portföyündeki ağırlık, ihtiyaç kredileri için verilen ve süresi ortalama 3 yıl olan poliçelerden oluşmaktadır. Bu değer için ileriye dönük tahmin edilmesi gereken aşağıdaki parametreler, Y.S.A. ile öngörülmüştür. Bunlar;

- TCMB USD efektif satış kuru,
- TCMB EUR efektif satış kuru,
- İptal Oranlarıdır.

Uzun süreli hayat sigortası poliçeleri yatırım amacıyla değil koruma amacıyla tanzim edilen poliçelerdir. Teminatın güvence altına alınması amacıyla, sigortalı tarafından kullanılan kredi tutarı, sigorta poliçesi için teminat tutarı olarak poliçede yer alır. Portföyde teminat tutarları; USD, EUR ve TL para birimlerine göre dağılmaktadır. Poliçenin süresi ise kullanılan kredinin süresine eşit olup; sigortalının kredi borcunu kapatması durumunda poliçeye iştirah işlemi uygulanmaktadır. Uygulamada hem azalan teminatlı hem de sabit teminatlı poliçeler için iştirah oranı ileriye dönük 3 yıl için öngörülmüştür.

Uygulamada, TCMB USD ve EUR efektif satış kuru ile 1, 2 ve 3 yıllık iptal oranlarının öngörülmesinde kullanılan yöntem ve işleyiş detaylı olarak anlatılmış; çalışmada elde edilen bilgiler çizelge ve şekillerle gösterilmiştir.

#### 4.1. Çözümlenen Veri

##### - TCMB USD ve EUR Efektif Satış Kuru

31.07.1991 – 30.09.2009 arasında yer alan her ay sonuna ait TCMB USD ve EUR efektif satış kuru değerleri alınmış, her iki parametrenin öngörülmesinde toplam 219 gözlem kullanılmıştır. TCMB USD efektif satış kuru için kullanılan veri kümesi Çizelge 4.1’de; verilere ilişkin grafikler ise Şekil 4.1.’de yer almaktadır.

Çizelge 4.1: TCMB USD efektif satış kuru veri kümesi.

Yıl	USD Efektif Satış Kuru Değeri	Yıl	USD Efektif Satış Kuru Değeri	Yıl	USD Efektif Satış Kuru Değeri	Yıl	USD Efektif Satış Kuru Değeri
31.07.1991	0,00442	29.02.1996	0,06604	30.09.2000	0,67010	30.04.2005	1,39320
31.08.1991	0,00463	31.03.1996	0,07114	31.10.2000	0,68637	31.05.2005	1,37430
30.09.1991	0,00473	30.04.1996	0,07483	30.11.2000	0,68721	30.06.2005	1,34980
31.10.1991	0,00495	31.05.1996	0,07840	31.12.2000	0,67602	31.07.2005	1,32960
30.11.1991	0,00506	30.06.1996	0,08193	31.01.2001	0,68346	31.08.2005	1,35930
31.12.1991	0,00510	31.07.1996	0,08367	28.02.2001	0,91190	30.09.2005	1,34910
31.01.1992	0,00551	31.08.1996	0,08649	31.03.2001	1,02702	31.10.2005	1,35580
29.02.1992	0,00588	30.09.1996	0,09166	30.04.2001	1,16791	30.11.2005	1,35820
31.03.1992	0,00627	31.10.1996	0,09608	31.05.2001	1,16385	31.12.2005	1,35030
30.04.1992	0,00663	30.11.1996	0,10214	30.06.2001	1,26070	31.01.2006	1,33030
31.05.1992	0,00689	31.12.1996	0,10837	31.07.2001	1,32824	28.02.2006	1,31960
30.06.1992	0,00690	31.01.1997	0,11618	31.08.2001	1,37721	31.03.2006	1,35120
31.07.1992	0,00709	28.02.1997	0,12224	30.09.2001	1,52828	30.04.2006	1,32380
31.08.1992	0,00709	31.03.1997	0,12704	31.10.2001	1,59979	31.05.2006	1,54650
30.09.1992	0,00735	30.04.1997	0,13546	30.11.2001	1,49394	30.06.2006	1,61300
31.10.1992	0,00788	31.05.1997	0,14074	31.12.2001	1,44868	31.07.2006	1,50490
30.11.1992	0,00832	30.06.1997	0,14775	31.01.2002	1,32920	31.08.2006	1,47750
31.12.1992	0,00860	31.07.1997	0,15997	28.02.2002	1,40817	30.09.2006	1,50660
31.01.1993	0,00886	31.08.1997	0,16663	31.03.2002	1,34626	31.10.2006	1,45930
28.02.1993	0,00923	30.09.1997	0,17407	30.04.2002	1,33989	30.11.2006	1,46920
31.03.1993	0,00950	31.10.1997	0,18156	31.05.2002	1,42184	31.12.2006	1,41450
30.04.1993	0,00970	30.11.1997	0,19484	30.06.2002	1,57908	31.01.2007	1,43110
31.05.1993	0,01017	31.12.1997	0,20625	31.07.2002	1,69251	28.02.2007	1,40100
30.06.1993	0,01092	31.01.1998	0,21550	31.08.2002	1,63161	31.03.2007	1,38890

Çizelge 4.1. devam ediyor.

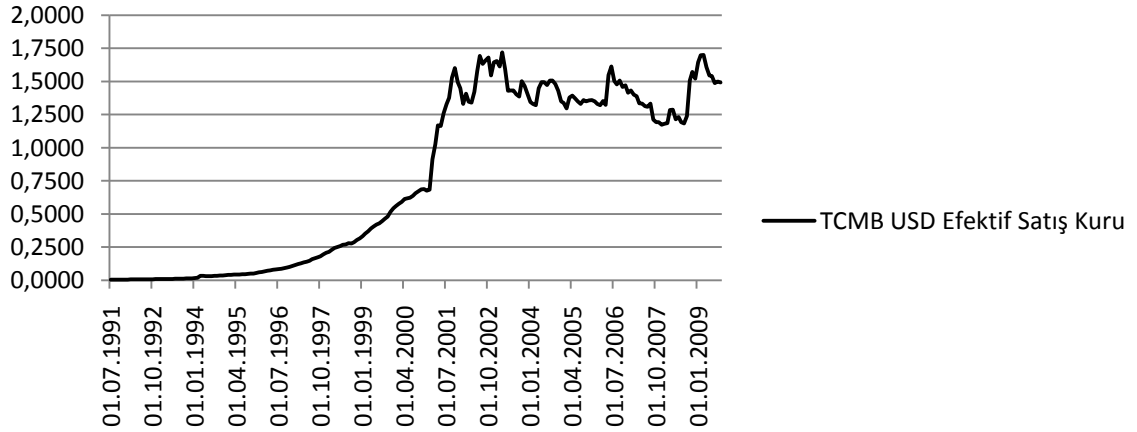
Yıl	USD Efektif Satış Kuru Değeri	Yıl	USD Efektif Satış Kuru Değeri	Yıl	USD Efektif Satış Kuru Değeri	Yıl	USD Efektif Satış Kuru Değeri
31.07.1993	0,01150	28.02.1998	0,22990	30.09.2002	1,65911	30.04.2007	1,33580
31.08.1993	0,01181	31.03.1998	0,24247	31.10.2002	1,68073	31.05.2007	1,33370
30.09.1993	0,01214	30.04.1998	0,25024	30.11.2002	1,54506	30.06.2007	1,31290
31.10.1993	0,01303	31.05.1998	0,25781	31.12.2002	1,64485	31.07.2007	1,30890
30.11.1993	0,01379	30.06.1998	0,26700	31.01.2003	1,65261	31.08.2007	1,33260
31.12.1993	0,01453	31.07.1998	0,27023	28.02.2003	1,61294	30.09.2007	1,21240
31.01.1994	0,01729	31.08.1998	0,27858	31.03.2003	1,71903	31.10.2007	1,19330
28.02.1994	0,01835	30.09.1998	0,27791	30.04.2003	1,59243	30.11.2007	1,19210
31.03.1994	0,03426	31.10.1998	0,28718	31.05.2003	1,42875	31.12.2007	1,17210
30.04.1994	0,03426	30.11.1998	0,30342	30.06.2003	1,43072	31.01.2008	1,17970
31.05.1994	0,03189	31.12.1998	0,31502	31.07.2003	1,43085	29.02.2008	1,18420
30.06.1994	0,03132	31.01.1999	0,33270	31.08.2003	1,40161	31.03.2008	1,28460
31.07.1994	0,03117	28.02.1999	0,35293	30.09.2003	1,38542	30.04.2008	1,28560
31.08.1994	0,03312	31.03.1999	0,36971	31.10.2003	1,50084	31.05.2008	1,21430
30.09.1994	0,03421	30.04.1999	0,39020	30.11.2003	1,46450	30.06.2008	1,23140
31.10.1994	0,03600	31.05.1999	0,40510	31.12.2003	1,40467	31.07.2008	1,19280
30.11.1994	0,03651	30.06.1999	0,42084	31.01.2004	1,34547	31.08.2008	1,18210
31.12.1994	0,03888	31.07.1999	0,42863	29.02.2004	1,32967	30.09.2008	1,23940
31.01.1995	0,04060	31.08.1999	0,44601	31.03.2004	1,31959	31.10.2008	1,50590
28.02.1995	0,04156	30.09.1999	0,46269	30.04.2004	1,44925	30.11.2008	1,57230
31.03.1995	0,04220	31.10.1999	0,48034	31.05.2004	1,49474	31.12.2008	1,52190
30.04.1995	0,04261	30.11.1999	0,51598	30.06.2004	1,49532	31.01.2009	1,64490
31.05.1995	0,04287	31.12.1999	0,54352	31.07.2004	1,47191	28.02.2009	1,69800
30.06.1995	0,04424	31.01.2000	0,55928	31.08.2004	1,50667	31.03.2009	1,69860



Çizelge 4.1. devam ediyor.

Yıl	USD Efektif Satış Kuru Değeri	Yıl	USD Efektif Satış Kuru Değeri	Yıl	USD Efektif Satış Kuru Değeri	Yıl	USD Efektif Satış Kuru Değeri
31.07.1995	0,04525	29.02.2000	0,57666	30.09.2004	1,50718	30.04.2009	1,60690
31.08.1995	0,04813	31.03.2000	0,59204	31.10.2004	1,47946	31.05.2009	1,54420
30.09.1995	0,04928	30.04.2000	0,61316	30.11.2004	1,42838	30.06.2009	1,53980
31.10.1995	0,05121	31.05.2000	0,61755	31.12.2004	1,35060	31.07.2009	1,48630
30.11.1995	0,05469	30.06.2000	0,62290	31.01.2005	1,33710	31.08.2009	1,49940
31.12.1995	0,05998	31.07.2000	0,63822	28.02.2005	1,29660	30.09.2009	1,49130
31.01.1996	0,06289	31.08.2000	0,65545	31.03.2005	1,37930		

### TCMB USD Efektif Satış Kuru



Şekil 4.1. TCMB USD efektif satış kuru veri grafiği

TCMB EUR efektif satış kuru için kullanılan veri kümesi Çizelge 4.2'de verilere ilişkin grafik ise Şekil 4.2.'de yer almaktadır.

Çizelge 4.2: TCMB EUR efektif satış kuru veri kümesi.

Yıl	EUR Efektif Satış Kuru Değeri	Yıl	EUR Efektif Satış Kuru Değeri	Yıl	EUR Efektif Satış Kuru Değeri	Yıl	EUR Efektif Satış Kuru Değeri
31.07.1991	0,00496	29.02.1996	0,08871	30.09.2000	0,58757	30.04.2005	1,80560
31.08.1991	0,00518	31.03.1996	0,09429	31.10.2000	0,58118	31.05.2005	1,71850
30.09.1991	0,00549	30.04.1996	0,09553	30.11.2000	0,59481	30.06.2005	1,62690
31.10.1991	0,00575	31.05.1996	0,10031	31.12.2000	0,62229	31.07.2005	1,60890
30.11.1991	0,00612	30.06.1996	0,10525	31.01.2001	0,62812	31.08.2005	1,67320
31.12.1991	0,00657	31.07.1996	0,11072	28.02.2001	0,83560	30.09.2005	1,62630
31.01.1992	0,00675	31.08.1996	0,11449	31.03.2001	0,90659	31.10.2005	1,64490
29.02.1992	0,00703	30.09.1996	0,11766	30.04.2001	1,05443	30.11.2005	1,60480
31.03.1992	0,00745	31.10.1996	0,12453	31.05.2001	0,99817	31.12.2005	1,59760
30.04.1992	0,00783	30.11.1996	0,13024	30.06.2001	1,06977	31.01.2006	1,60730
31.05.1992	0,00827	31.12.1996	0,13618	31.07.2001	1,16239	28.02.2006	1,56390
30.06.1992	0,00887	31.01.1997	0,13867	31.08.2001	1,25537	31.03.2006	1,63130
31.07.1992	0,00933	28.02.1997	0,14184	30.09.2001	1,39643	30.04.2006	1,66110
31.08.1992	0,00986	31.03.1997	0,14843	31.10.2001	1,45154	31.05.2006	1,98730
30.09.1992	0,01004	30.04.1997	0,15312	30.11.2001	1,32906	30.06.2006	2,02220
31.10.1992	0,01005	31.05.1997	0,16195	31.12.2001	1,27576	31.07.2006	1,90910
30.11.1992	0,01015	30.06.1997	0,16711	31.01.2002	1,14617	31.08.2006	1,89380
31.12.1992	0,01042	31.07.1997	0,17000	28.02.2002	1,21765	30.09.2006	1,90840
31.01.1993	0,01077	31.08.1997	0,18167	31.03.2002	1,17435	31.10.2006	1,85670
28.02.1993	0,01100	30.09.1997	0,19324	30.04.2002	1,21019	30.11.2006	1,93330
31.03.1993	0,01140	31.10.1997	0,20724	31.05.2002	1,33269	31.12.2006	1,86320
30.04.1993	0,01204	30.11.1997	0,21604	30.06.2002	1,57023	31.01.2007	1,85490
31.05.1993	0,01248	31.12.1997	0,22496	31.07.2002	1,66357	28.02.2007	1,85140
30.06.1993	0,01260	31.01.1998	0,23640	31.08.2002	1,60779	31.03.2007	1,85000

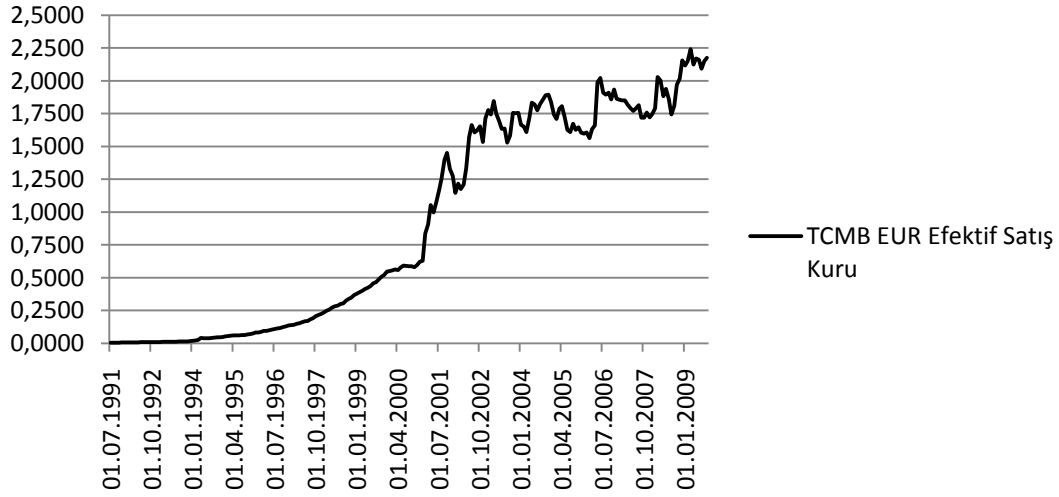
Çizelge 4.2. devam ediyor.

<b>Yıl</b>	<b>EUR Efektif Satış Kuru Değeri</b>	<b>Yıl</b>	<b>EUR Efektif Satış Kuru Değeri</b>	<b>Yıl</b>	<b>EUR Efektif Satış Kuru Değeri</b>	<b>Yıl</b>	<b>EUR Efektif Satış Kuru Değeri</b>
31.07.1993	0,01305	28.02.1998	0,24841	30.09.2002	1,62211	30.04.2007	1,82000
31.08.1993	0,01385	31.03.1998	0,25876	31.10.2002	1,65233	31.05.2007	1,79140
30.09.1993	0,01473	30.04.1998	0,27204	30.11.2002	1,53424	30.06.2007	1,76970
31.10.1993	0,01519	31.05.1998	0,28266	31.12.2002	1,71426	31.07.2007	1,78900
30.11.1993	0,01573	30.06.1998	0,28802	31.01.2003	1,77805	31.08.2007	1,81360
31.12.1993	0,01641	31.07.1998	0,29913	28.02.2003	1,74311	30.09.2007	1,71940
31.01.1994	0,01956	31.08.1998	0,30396	31.03.2003	1,84486	31.10.2007	1,71850
28.02.1994	0,02093	30.09.1998	0,32463	30.04.2003	1,74705	30.11.2007	1,75750
31.03.1994	0,02589	31.10.1998	0,33985	31.05.2003	1,69121	31.12.2007	1,72100
30.04.1994	0,04026	30.11.1998	0,34816	30.06.2003	1,63388	31.01.2008	1,74490
31.05.1994	0,03790	31.12.1998	0,36891	31.07.2003	1,63574	29.02.2008	1,78850
30.06.1994	0,03875	31.01.1999	0,37880	31.08.2003	1,52915	31.03.2008	2,02830
31.07.1994	0,03821	28.02.1999	0,38903	30.09.2003	1,58118	30.04.2008	2,00200
31.08.1994	0,04105	31.03.1999	0,39991	31.10.2003	1,75628	31.05.2008	1,88240
30.09.1994	0,04319	30.04.1999	0,41365	30.11.2003	1,75388	30.06.2008	1,93930
31.10.1994	0,04710	31.05.1999	0,42442	31.12.2003	1,75612	31.07.2008	1,85980
30.11.1994	0,04567	30.06.1999	0,43611	31.01.2004	1,66622	31.08.2008	1,74240
31.12.1994	0,04899	31.07.1999	0,45850	29.02.2004	1,65012	30.09.2008	1,80920
31.01.1995	0,05245	31.08.1999	0,46612	31.03.2004	1,60990	31.10.2008	1,97120
28.02.1995	0,05582	30.09.1999	0,48836	30.04.2004	1,71460	30.11.2008	2,01660
31.03.1995	0,05900	31.10.1999	0,50680	31.05.2004	1,83270	31.12.2008	2,15430
30.04.1995	0,06033	30.11.1999	0,52140	30.06.2004	1,81831	31.01.2009	2,11600
31.05.1995	0,06039	31.12.1999	0,54536	31.07.2004	1,77395	28.02.2009	2,14950
30.06.1995	0,06201	31.01.2000	0,55061	31.08.2004	1,82277	31.03.2009	2,23990

Çizelge 4.2. devam ediyor.

Yıl	EUR Efektif Satış Kuru Değeri	Yıl	EUR Efektif Satış Kuru Değeri	Yıl	EUR Efektif Satış Kuru Değeri	Yıl	EUR Efektif Satış Kuru Değeri
31.07.1995	0,06390	29.02.2000	0,55654	30.09.2004	1,85684	30.04.2009	2,12470
31.08.1995	0,06416	31.03.2000	0,56405	31.10.2004	1,88971	31.05.2009	2,16910
30.09.1995	0,06779	30.04.2000	0,55867	30.11.2004	1,89347	30.06.2009	2,16050
31.10.1995	0,07111	31.05.2000	0,57680	31.12.2004	1,83840	31.07.2009	2,09090
30.11.1995	0,07451	30.06.2000	0,59326	31.01.2005	1,74300	31.08.2009	2,15150
31.12.1995	0,08187	31.07.2000	0,59062	28.02.2005	1,70930	30.09.2009	2,17400
31.01.1996	0,08276	31.08.2000	0,58730	31.03.2005	1,78700		

### TCMB EUR Efektif Satış Kuru



Şekil 4.2. TCMB EUR efektif satış kuru veri grafiği

#### - Yıllık İptal Oranları

01.08.2006 – 01.08.2009 arasında tanzim edilen poliçe sayısı ile bu poliçelerden 1 yıl sonra iptal olan poliçe sayıları alınmıştır. Her ay sonunda ilgili aya ait tanzim edilen poliçe sayısı, 1 yıl sonra iptal olmayan poliçe sayısından çıkarılarak, 1 yıl

içerisinde iptal edilen poliçe sayısı bulunmuştur. Bulunan bu değer tanzim edilen poliçe sayısına oranlanarak, 1 yıllık iptal oranı bulunmuştur.

İptal oranlarına ilişkin öngörü portföy ortalaması olan 3 yıl için yapılacaktır. Böylece 30.09.2009 ve sonrasında tanzim edilecek poliçelerin 3 yıl boyunca yıllık iptal oranları belirlenmiş olacaktır.

$\rho_{(ij)}$  : i. dönemde tanzim edilen poliçelerin j. yıldaki iptal oranı,

j : iptal dönemine ilişkin yıl sayısı,  $j = 1,2,3$

i : poliçelerin tanzim edildiği tarihe ilişkin ay cinsinden değerlerdir.  $i = 1, \dots, 37$

i indisine ilişkin değerler
31.08.2006 = 1
30.09.2006 = 2
...
31.08.2009 = 37

Yukarıda yer alan bilgiler doğrultusunda Y.S.A. modelimizde  $j = 1, 2, 3$  olmak üzere yıllık iptal oranları belirlenecektir. İlk olarak 1 yıllık iptal oranı için yapılan çalışma anlatılmaktadır.  $j = 1$  için toplam 20 adet model üzerinde uygulama gerçekleştirilmiştir.

#### 4.2. Ağ Yapısının Tanımı

Çalışmada basit bir mimari yapıya ve algoritmaya sahip olan ve birçok farklı problem türünü çözümede başarısı kanıtlanmış olan Ç.K.İ.B.Y.S.A. modeli kullanılmıştır.

#### Verilerin Normalleştirilmesi

Verilerin 0–1 aralığına dönüştürülmesi işlemi veri normalleştirilmesi olarak isimlendirilmektedir. Bu işlem girdi nöronları ile çıktı nöronlarının ağına kullanımına uygun bir şekilde analizinin yapılması ve uygulamada kullanılması için gereklidir.

Böylece veriler bir ön işlemde geçirilerek Y.S.A.'da kullanılacak şekilde ölçeklenmiş olacaktır.

Uygulamada tek kutuplu lojistik aktivasyon fonksiyonu kullanılacağı için, veri normalleştirilmesi Eş. 4.1.'de yer alan formülasyon kullanılarak her veri için ayrı ayrı yapılır.

$$x_i' = \frac{x_i - \min(x_i)}{\max(x_i) - \min(x_i)} \quad (4.1.)$$

Burada;

$x_i'$  : normalleştirilen  $x_i$  giriş ya da çıkış nöronu,

$\min(x_i)$  : giriş nöronlarına ait veri kümesindeki minimum giriş nöron değeri,

$\max(x_i)$  : giriş nöronlarına ait veri kümesindeki maksimum değere sahip giriş nöronu değeridir.

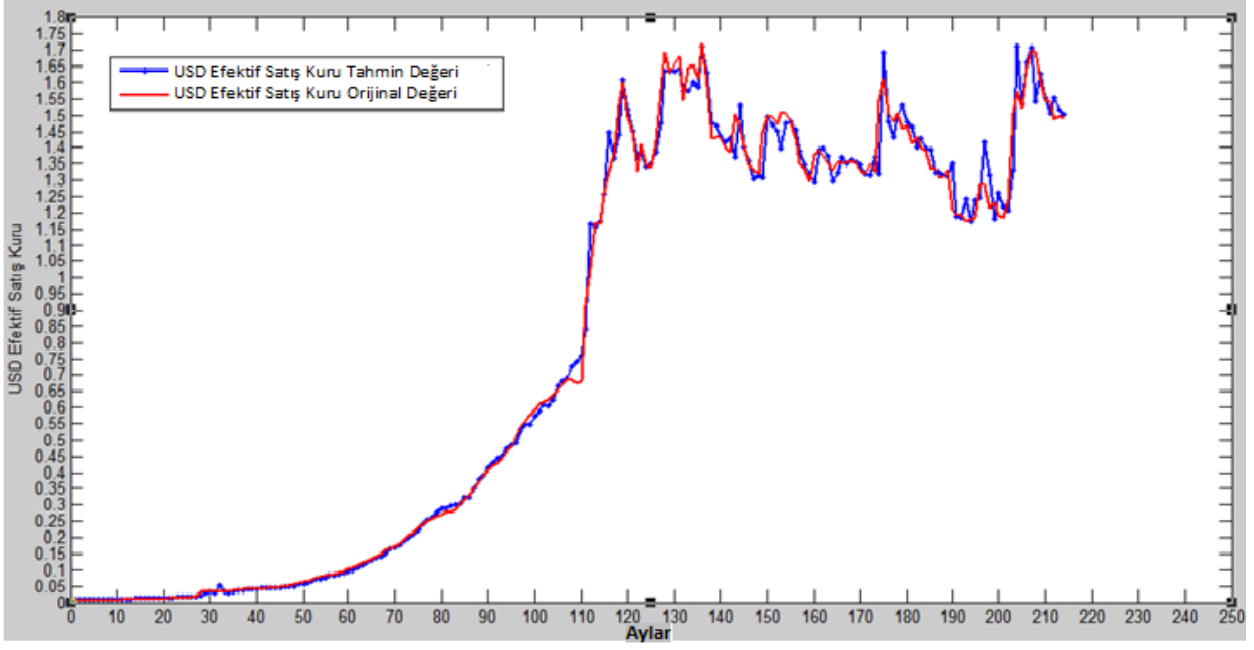
### **Eğitim Kümesi ve Test Kümesinin Belirlenmesi**

En iyi ağırlık değerlerinin bulunması için eğitim kümesi; ağın genelleştirme yeteneğini ölçmek içinse test kümesinin belirlenmesi gerekmektedir. Bu kümeler tahmin edilecek 5 veri kümesi için de ayrı ayrı belirlenmiştir.

#### **- TCMB USD Efektif Satış Kuru**

Çalışmada kullanılan veri kümesinin %85'i eğitim, %5'i onaylama (validation) ve %10'unu test amacıyla kullanılmıştır. Buna göre ilk 186 gözlem eğitim, sonraki 11 gözlem onaylama ve son 22 gözlem ise test kümesi için kullanılmıştır.

Eğitilmiş olan ağ, tüm verilere göre çalıştırılır. Ağın tahmini değerleri orijinal değerlerine uygun olacak şekilde yeniden ölçeklendirilir. Bu işlem sonucunda 5-8-1 mimari yapıya sahip, eğitim, onaylama ve test kümesine ait tahmin değerleri ve orijinal değerler Şekil 4.3.'te gösterilmiştir.

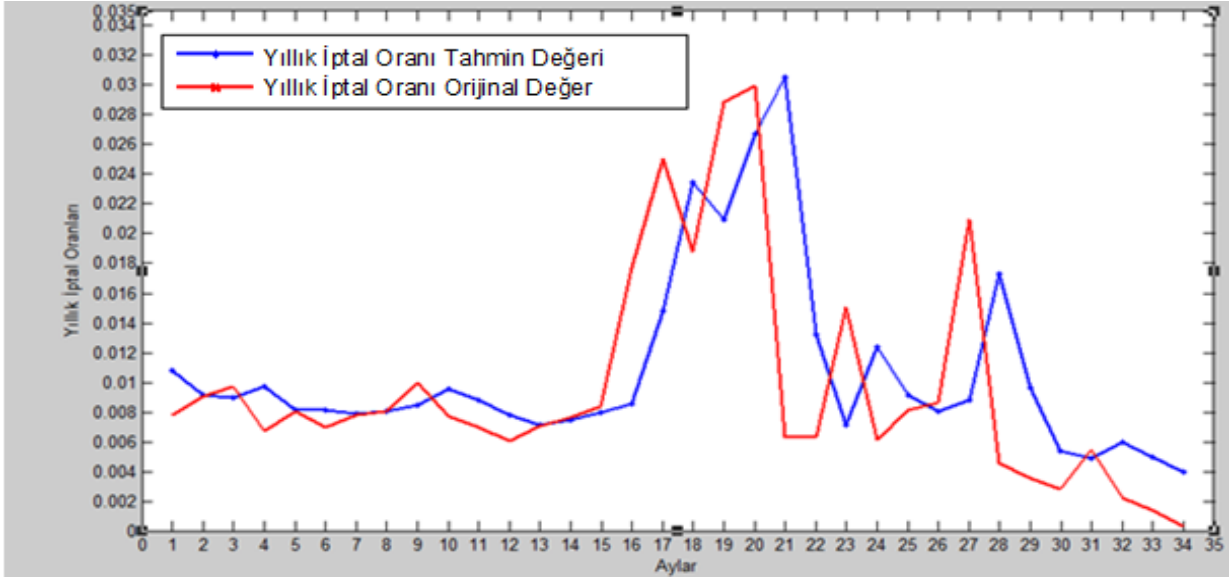


Şekil 4.3. TCMB USD efektif satış kuruna ait aylar itibariyle orijinal ve tahmin değerleri

#### - Yıllık İptal Oranları

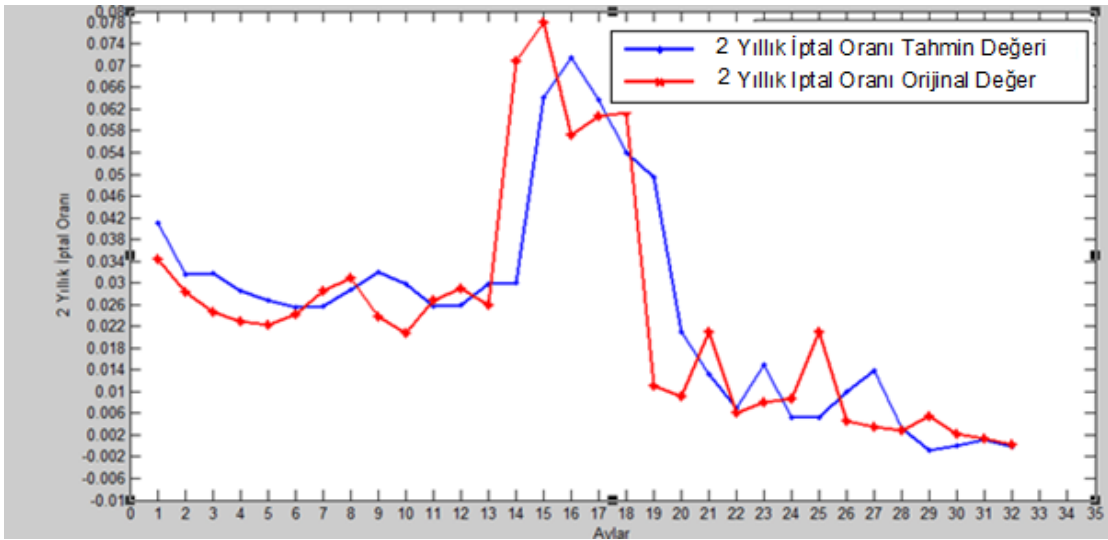
Çalışmada kullanılacak gözlemlerin %80'i eğitim, %5'i onaylama (validation) ve %15'i test kümesi için kullanılmıştır. Böylece eğitim kümesinde 27; onaylama kümesinde 2 ve test kümesinde ise 5 adet gözlem kullanılmıştır.

Eğitilmiş olan ağ, tüm verilere göre çalıştırıldıktan sonra, 1 yıllık iptal oranlarına ait 2-1-1 mimari yapıdaki tahmin değerleri (eğitim, onaylama ve test kümesine ait değerler) ile orijinal değerlere ait grafik Şekil 4.4.'de verilmiştir.



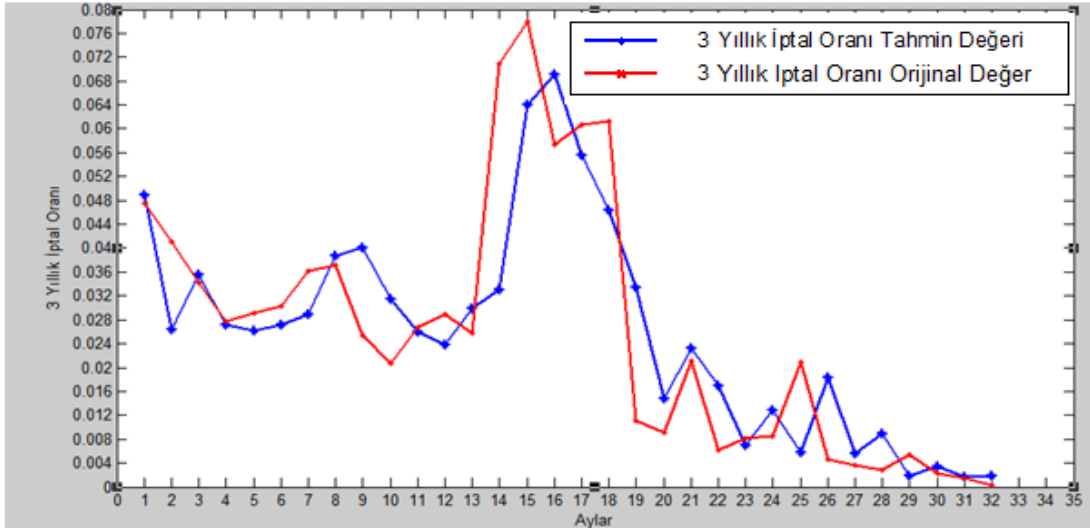
Şekil 4.4. Aylara göre 1 yıllık iptal oranlarına ait orijinal ve tahmin değerleri

Aynı şekilde 4-4-1 mimari yapıya sahip 2 yıllık iptal oranlarına ait orijinal ve tahmin değerleri Şekil 4.5.'te; 3 yıllık iptal oranlarına ait değerler ise Şekil 4.6.'te yer almaktadır.



Şekil 4.5. Aylara göre 2 yıllık iptal oranlarına ait orijinal ve tahmin değerleri

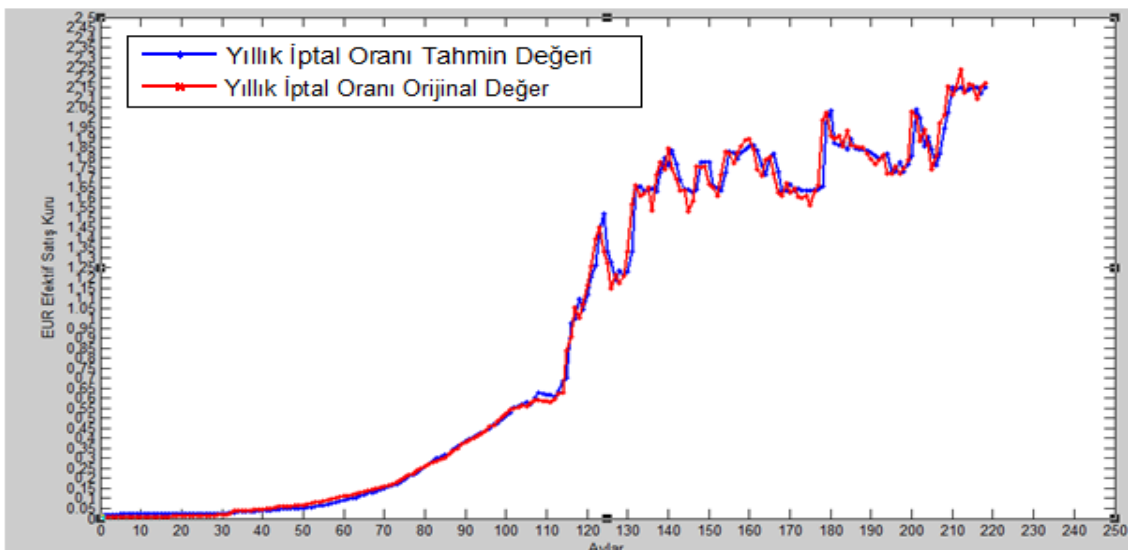




Şekil 4.6. Aylara göre 3 yıllık iptal oranlarına ait orijinal ve tahmin değerleri

#### - TCMB EUR Efektif Satış Kuru

Çalışmada kullanılacak veri kümeleri için, TCMB USD efektif satış kurunu belirlemede kullandığımız dağılım kullanılmıştır. Böylece veri kümesinin %85'i eğitim, %5'i onaylama ve %10'u test kümesi olarak kullanılmıştır. Buna göre ilk 186 gözlem eğitim, sonraki 11 gözlem onaylama ve son 22 gözlem ise test kümesi için kullanılmıştır. 1-10-1 mimari yapıya sahip TCMB EUR efektif satış kuruna ait aylar itibariyle orijinal ve tahmin değerleri ise (eğitim, onaylama ve test kümesine ait değerler) Şekil 4.7.'de gösterilmiştir.



Şekil 4.7. TCMB EUR efektif satış kuruna ait aylar itibariyle orijinal ve tahmin değerleri

## **Girdi Birimleri Sayısının Belirlenmesi**

### **- TCMB USD Efektif Satış Kuru**

Çalışmada 1 ile 5 arasında girdi birimleri kullanılmıştır. Bu sayı aralığı kullanıcı tarafından belirlenmiş olup, uygun modelin bulunmasında yararlanılmıştır. Bu aralıktaki girdi birimleri alınarak Matlab üzerinde yapılan çalışma sonrasında, TCMB USD efektif satış kuru girdi birim sayısı olarak 5 bulunmuş ve çalışmada kullanılmıştır.

### **- Yıllık İptal Oranları**

Çalışmada 1 ile 4 arasında girdi birimleri alınmıştır. En küçük H.K.O.'ya sahip modelin girdi birim sayısı belirlenmiş ve çalışmada kullanılmıştır. Bu değerler 1 yıllık iptal oranı için 2; 2 ve 3 yıllık iptal oranları için 4 olarak bulunmuştur.

### **- TCMB EUR Efektif Satış Kuru**

Çalışmada 1 ile 5 arasında girdi birimleri alınmıştır. En küçük H.K.O.'ya sahip modelin girdi birim sayısı 1 olarak belirlenmiş ve TCMB EUR efektif satış kuru için girdi birim sayısı olarak kullanılmıştır.

## **Gizli Katman ve Gizli Katman Birim Sayısının Belirlenmesi**

Günay vd. (2007a)'de, öngörü problemlerinde, teorik çalışmalar, yapay sinir ağı modelinde tek gizli tabakanın kullanılmasının yeterli olacağını ortaya koymuştur. Bu bilgi doğrultusunda gizli katman sayısı 1 alınmıştır. Gizli katman sayısı için yine kullanıcı tarafından belirlenmiş aralıklardaki girdi birim sayıları kullanılmıştır. Bunlar;

TCMB USD efektif satış kuru için 1–10 arasında gizli katman sayısı,

TCMB EUR efektif satış kuru için 1–10 arasında gizli katman sayısı,

1 Yıllık iptal oranı için 1–5 arasında gizli katman sayısı,

2 Yıllık iptal oranı için 1–5 arasında gizli katman sayısı,

3 Yıllık iptal oranı için 1–5 arasında gizli katman sayısıdır.

Bu deęerler kullanılarak öngörüde kullanılacak gizli katmandaki birim sayısı için Matlab paket programı yardımıyla en küçük H.K.O.'ya sahip modeldeki birim sayısı alınmıştır. Buna göre;

TCMB USD efektif satış kuru için gizli katman sayısı 1 ve gizli katman birim sayısı 8,

TCMB EUR efektif satış kuru için gizli katman sayısı 1 ve gizli katman birim sayısı 8,

1 Yıllık iptal oranı için gizli katman sayısı 1 ve gizli katman birim sayısı 1,

2 Yıllık iptal oranı için gizli katman sayısı 1 ve gizli katman birim sayısı 4,

3 Yıllık iptal oranı için gizli katman sayısı 1 ve gizli katman birim sayısı 5 olarak alınmıştır.

### **Çıktı Katmanı Birim Sayısının Belirlenmesi**

Modelimizde Box-Jenkins yönteminde de kullanılan iteratif öngörü yöntemi ile çıktı deęerleri belirlenmektedir. Yani ilk öngörüye ait veriler kullanılarak bir sonraki öngörü elde edilmektedir. Bu nedenle yalnızca 1 çıktı katmanı ve 1 çıktı deęerinin kullanılması yeterli olacaktır. TCMB USD ve EUR efektif satış kuru ile iptal oranları için modelimizde yalnızca 1 çıktı katmanı ve 1 çıktı deęeri kullanılmıştır.

### **Öğrenme Algoritması**

Birçok çalışmada geriye yayılım öğrenme algoritması uygulanmıştır. Fakat bu algoritma, pek çok problem için yavaş kalmaktadır. Bu algoritmalarından daha hızlı ve daha yüksek performanslı algoritmalar da bulunmaktadır. Hem hızlı ve yüksek performanslı olması hem de MATLAB'da hazır ara yüze sahip olması nedeniyle, çalışmada L.M.Ö.A'sı kullanılmıştır.

### **Aktivasyon Fonksiyonu**

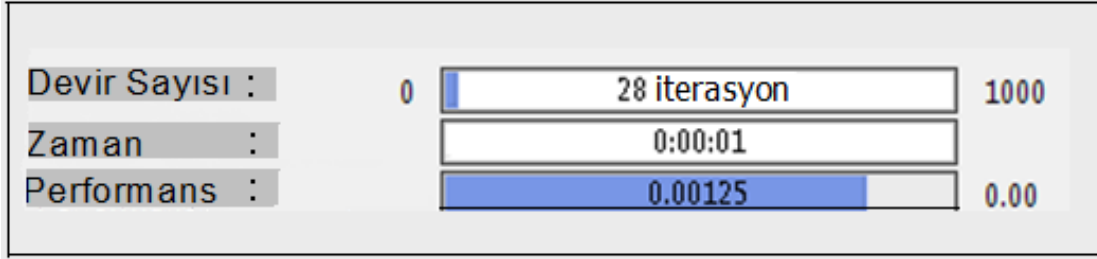
Tüm veri kümeleri için girdi katmanında yer alan birimler ile gizli katman birimleri arasında tek kutuplu lojistik aktivasyon fonksiyonu; gizli katman birimleri ile çıktı katmanı birimleri arasında ise doğrusal aktivasyon fonksiyonu kullanılmıştır.

## Performans Ölçütü

### - TCMB USD Efektif Satış Kuru

Test kümesi tahminleri için performans ölçütü olarak H.K.O. kullanılmıştır. Y.S.A. modelinde en iyi modelin H.K.O.'sı en küçük değere sahip 48. model olduğu görülmüştür. Kullanılan 50 model içerisinde 5-8-1 mimari yapısına sahip 48. Model; 0,00125'lik değerle en küçük H.K.O.'ya sahiptir.

48. modele ait bilgilerin Matlab'ta yer alan görüntüsü ise Şekil 4.8.'de yer almaktadır.



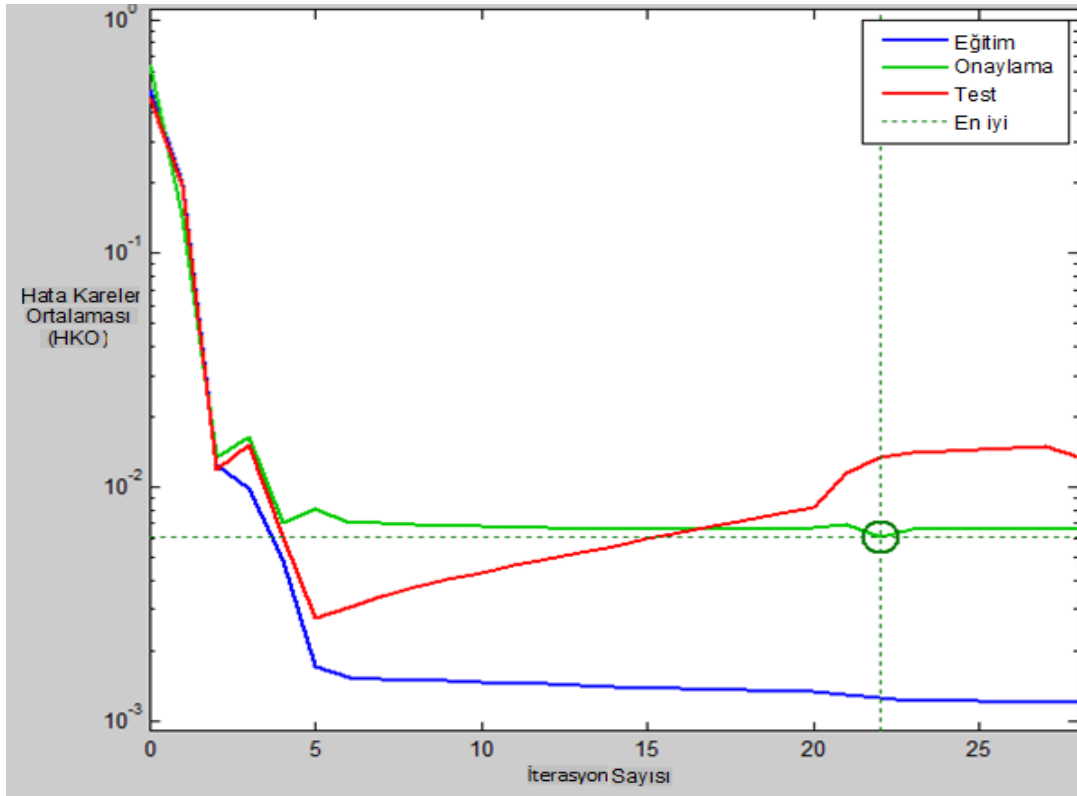
Şekil 4.8. Y.S.A. modelinde 48. modele ait iterasyon sayısı, zaman ve performans değerleri

Devir Sayısı : eğitimde kullanılan iterasyon sayısı,

Zaman : modelin eğitimi için geçen süre,

Performans : H.K.O. değeridir.

H.K.O. ve iterasyon sayısına ilişkin grafiksel gösterim Şekil 4.9.'da yer almaktadır.

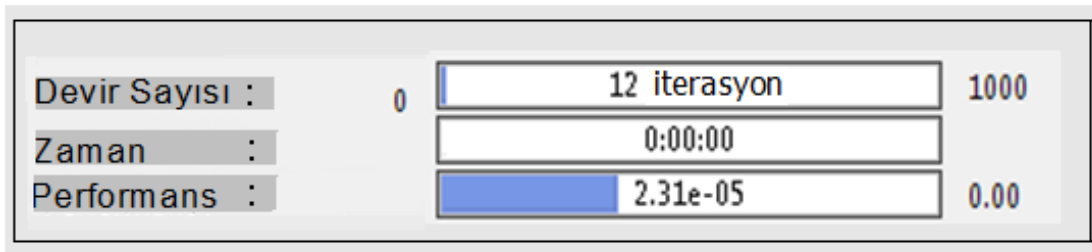


Şekil 4.9. TCMB USD efektif satış kuru için H.K.O. ve iterasyon sayıları

#### - Yıllık İptal Oranları

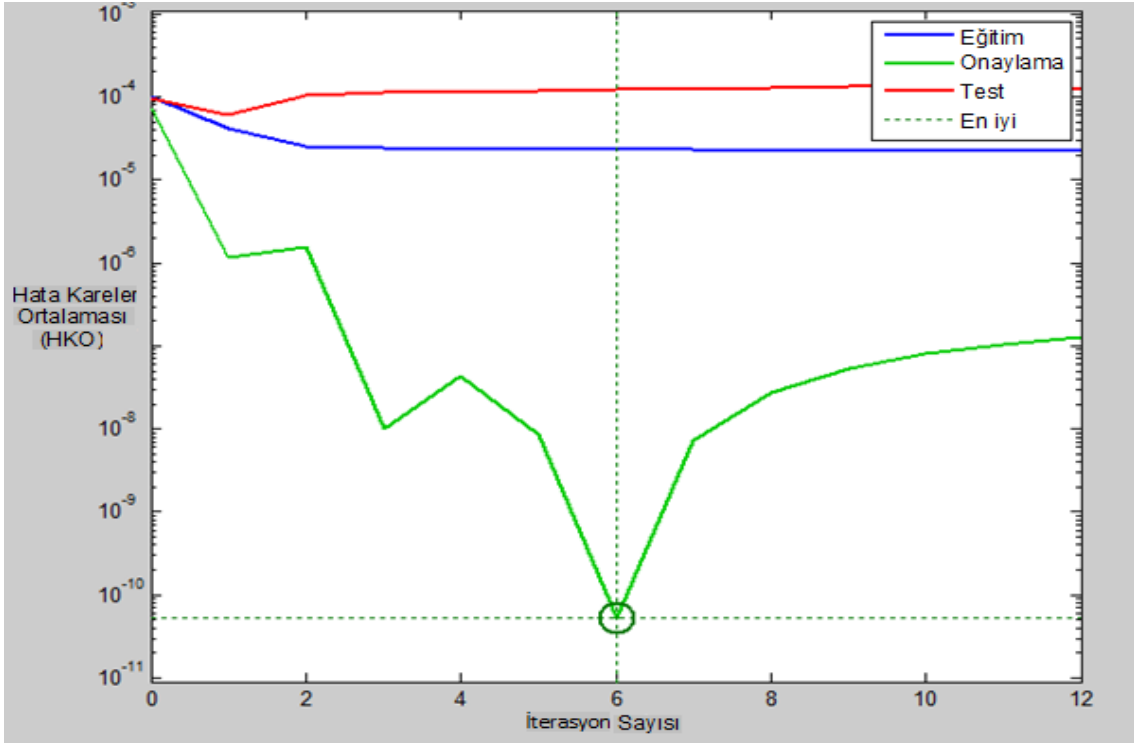
Performans ölçütü olarak H.K.O. kullanılmıştır. Y.S.A. modelinde 1 yıllık iptal oranı için en iyi modelin H.K.O. değeri en küçük olan 6. model olduğu görülmüştür. Kullanılan 20 model içerisinde 2-1-1 mimari yapısına sahip 6. Model; 0,000121'lik değerle en küçük H.K.O.'na sahiptir.

6. modele ait bilgilerin Matlab'taki görüntüsü Şekil 4.10.'da yer almaktadır.



Şekil 4.10. Y.S.A. modelinde 6. modele ait iterasyon sayısı, zaman ve performans değerleri

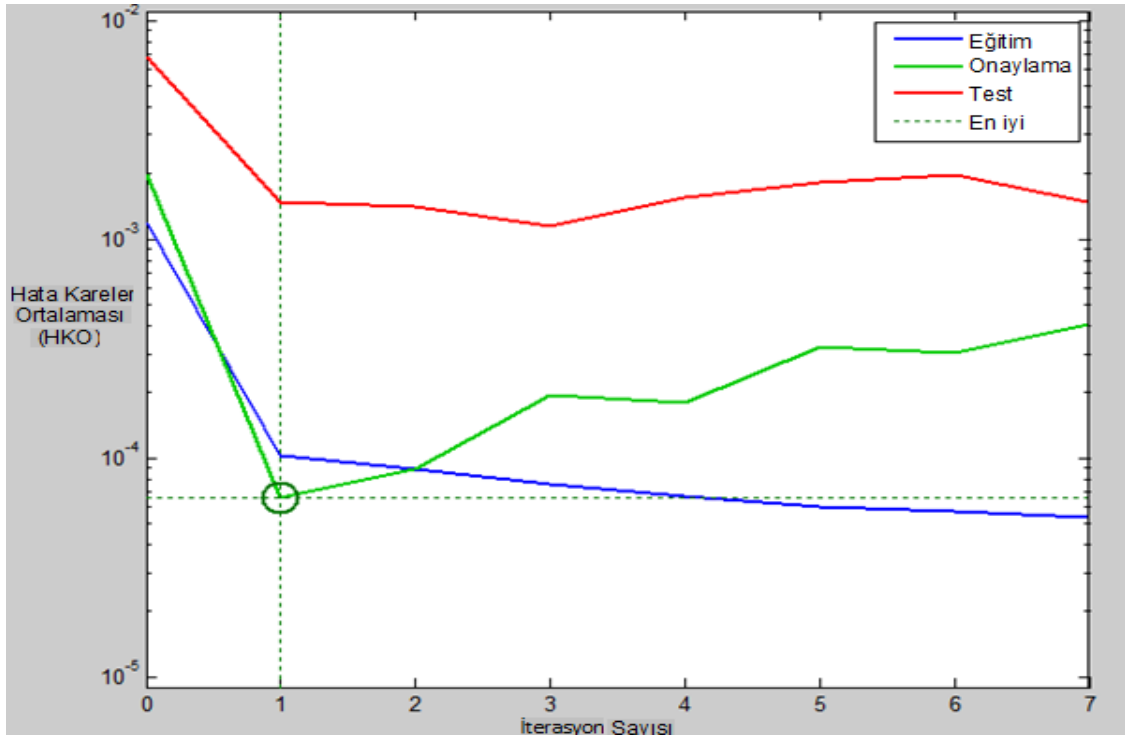
H.K.O. ve iterasyon sayısına ilişkin grafiksel gösterim ise Şekil 4.11.'de yer almaktadır.



Şekil 4.11. 1 Yıllık iptal oranı için H.K.O. ve iterasyon sayıları

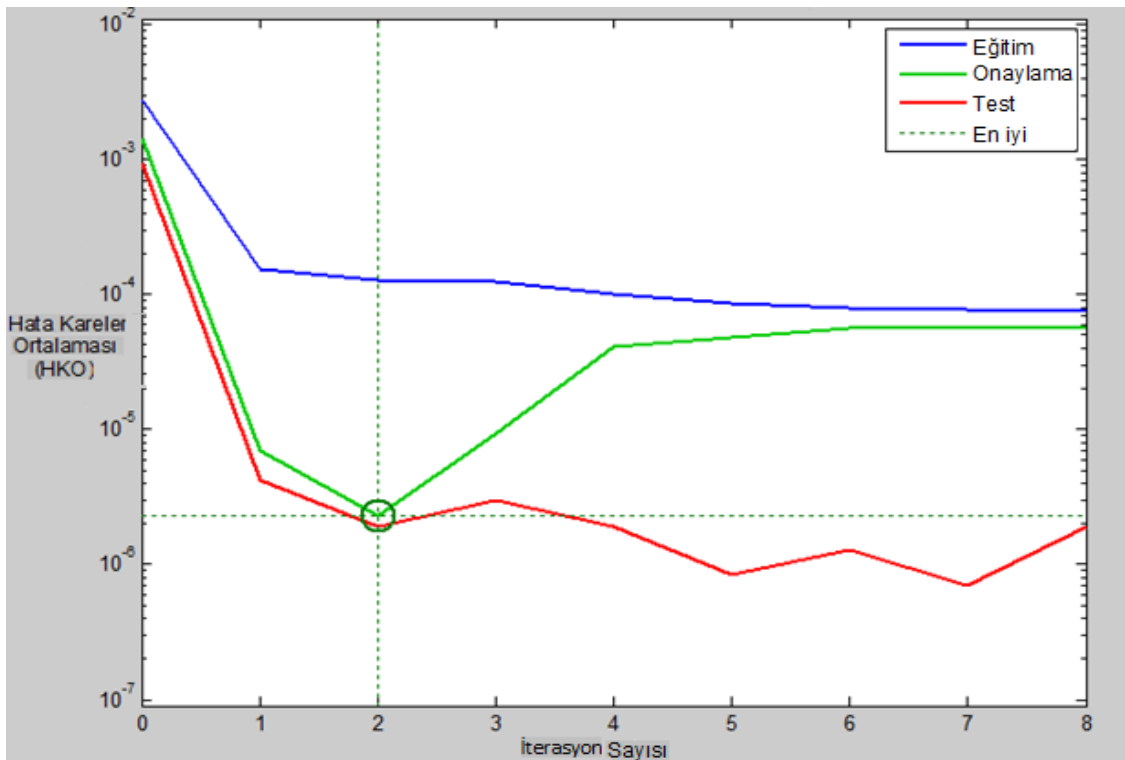
Aynı uygulama  $j = 2$  için yani 2 yıllık iptal oranını öngörmek için yapılmıştır.

H.K.O. ve iterasyon sayısının grafiksel olarak gösterimi ise Şekil 4.12'de verilmiştir.



Şekil 4.12. 2 Yıllık iptal oranı için H.K.O. ve iterasyon sayıları

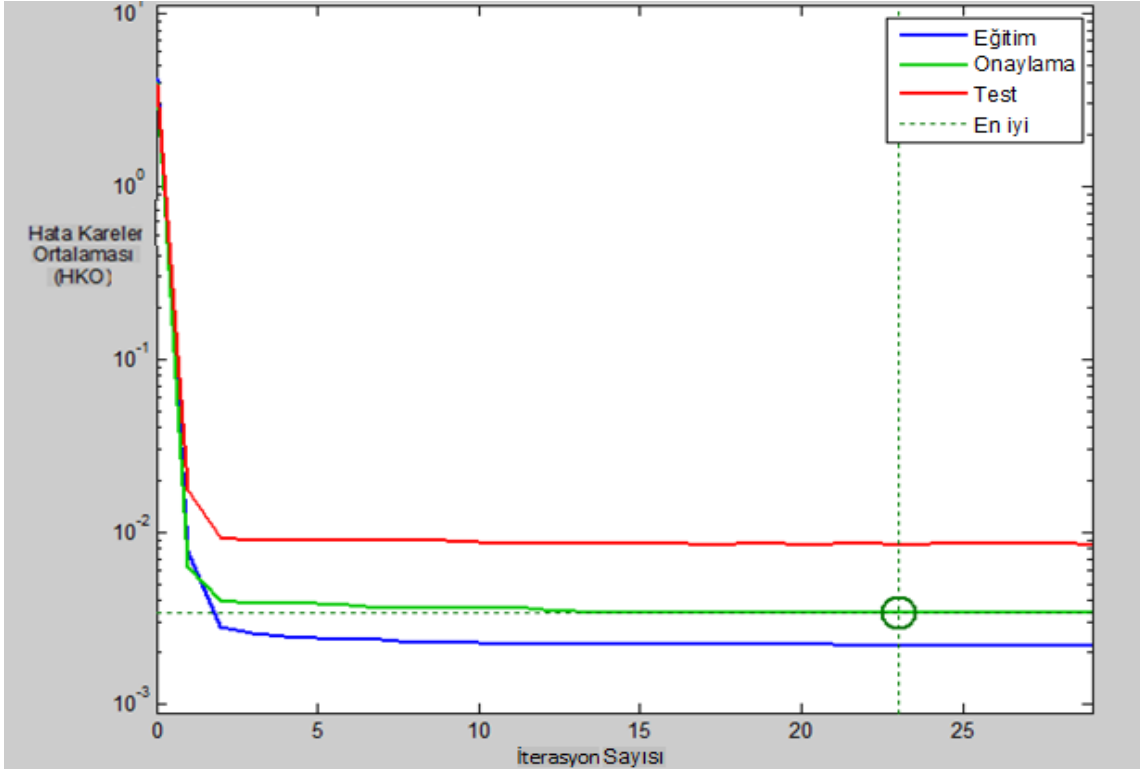
Uygulama  $j = 3$  için yapıldığında bulunan sonuçlar ve ilgili grafikler Şekil 4.13'te verilmiştir.



Şekil 4.13. 3 Yıllık iptal oranı için H.K.O. ve iterasyon sayıları

## - TCMB EUR Efektif Satış Kuru

Performans ölçütü olarak H.K.O. kullanılmıştır. Y.S.A. modelinde en iyi modelin H.K.O.'sı en küçük değere sahip 10. model olduğu görülmektedir. Kullanılan 50 model içerisinde 10. Model; 0,00222'lik değerle ve 1-10-1 mimari yapısı ile en küçük H.K.O. sahiptir. Tüm modellere ilişkin H.K.O. değerleri Şekil 4.14.'te yer almaktadır.



Şekil 4.14. TCMB EUR efektif satış kuru için H.K.O. ve iterasyon sayıları

## Mimari Yapı

### - TCMB USD Efektif Satış Kuru

Y.S.A.'lar ile modellemede öngörüle bulunmak için, mimari yapının belirlenmesi önemlidir. Çok katmanlı modelin uygulanmasında başarı için gerekli en önemli parametrelerden biridir. Fakat en iyi yapının belirlenmesi açık ve zor bir problemdir. Bu doğrultuda, basitlik sağlanması amacıyla da, tek girdi, tek gizli ve bir çıktı katmanına sahip çok katmanlı bir ağ yapısı seçilebilir.



Yapı girdi katmanı, tek gizli katman ve çıktı katmanı olmak üzere üç katmandan oluşmaktadır. Mimari yapı için girdi tabakasındaki birimlerin sayısı, gizli katman sayısı, gizli katmandaki birim sayısı ve çıktı tabakasındaki birimlerin sayısı belirlenmektedir.

Uygulamada en doğru mimari yapıyı bulmak için, Matlab paket programında yer alan hazır araç paketi Neural Networks üzerinde Ç.K.İ.B.Y.S.A. incelenerek, H.K.O. en düşük olan model kullanılmıştır. Girdi katmanındaki birimler 1 ile 5, gizli katmandaki birimler 1 ile 10 arasında değiştirilerek, toplam 50 mimari denenmiştir. TCMB USD efektif satış kuru için incelenen tüm mimarilere ilişkin test kümesi üzerinden hesaplanan H.K.O. değerleri Çizelge 4.3'te yer almaktadır.

Çizelge 4.3: TCMB USD efektif satış kuru için çalıştırılan modellere ait H.K.O. ve iterasyon sayıları.

Model Sayısı	Girdi Katmanı Birim Sayısı	Gizli Katman Birim Sayısı	H.K.O.	İterasyon Sayısı
1	1	1	0,00219	7
2	1	2	0,00563	17
3	1	3	0,00288	9
4	1	4	0,00208	15
5	1	5	0,00322	11
6	1	6	0,00274	10
7	1	7	0,00231	11
8	1	8	0,00252	9
9	1	9	0,00234	14
10	1	10	0,00157	36
11	2	1	0,00209	18
12	2	2	0,00273	17
13	2	3	0,00207	23
14	2	4	0,00244	12
15	2	5	0,00237	12
16	2	6	0,00399	12

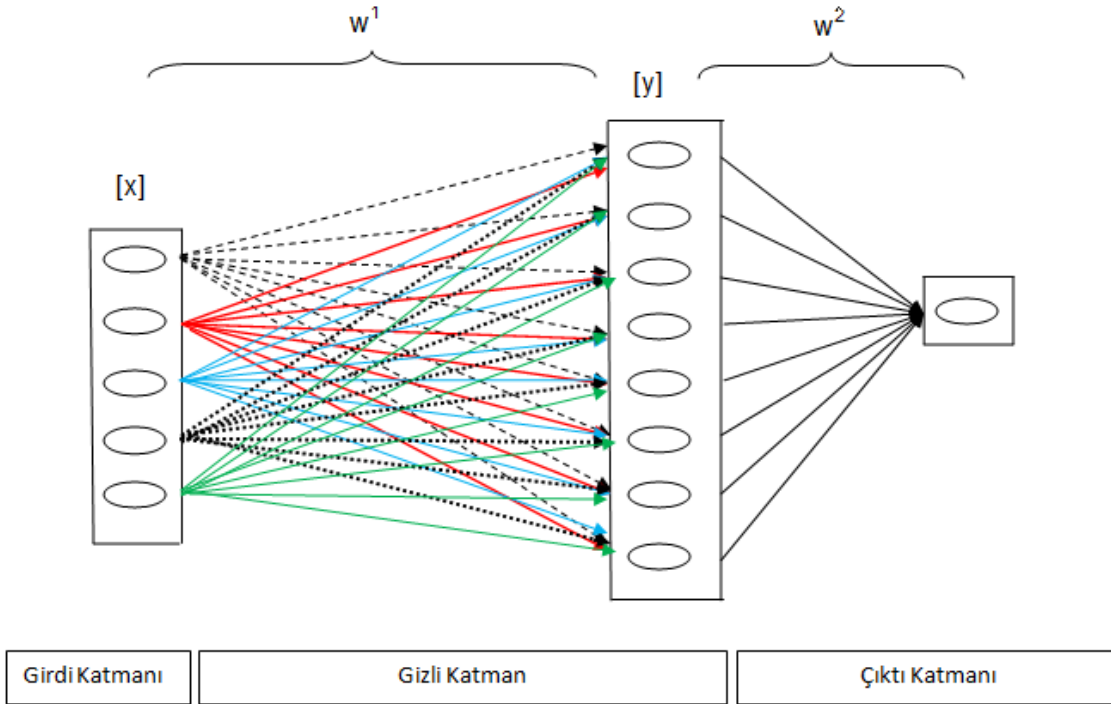
Çizelge 4.3. devam ediyor.

<b>Model Sayısı</b>	<b>Girdi Katmanı Birim Sayısı</b>	<b>Gizli Katman Birim Sayısı</b>	<b>H.K.O.</b>	<b>İterasyon Sayısı</b>
17	2	7	0,0028	9
18	2	8	0,00234	10
19	2	9	0,00187	13
20	2	10	0,00233	13
21	3	1	0,0021	10
22	3	2	0,00297	10
23	3	3	0,00299	10
24	3	4	0,00237	12
25	3	5	0,00182	15
26	3	6	0,00176	23
27	3	7	0,00221	31
28	3	8	0,00212	29
29	3	9	0,00225	14
30	3	10	0,00163	47
31	4	1	0,00208	23
32	4	2	0,00246	36
33	4	3	0,00212	12
34	4	4	0,00227	17
35	4	5	0,00191	9
36	4	6	0,00198	20
37	4	7	0,00193	10
38	4	8	0,00204	11
39	4	9	0,00186	12
40	4	10	0,00205	10
41	5	1	0,00209	23
42	5	2	0,00213	14
43	5	3	0,0018	17

Çizelge 4.3. devam ediyor.

Model Sayısı	Girdi Katmanı Birim Sayısı	Gizli Katman Birim Sayısı	H.K.O.	İterasyon Sayısı
44	5	4	0,00202	12
45	5	5	0,00243	12
46	5	6	0,00227	11
47	5	7	0,00245	9
48	5	8	0,00124	28
49	5	9	0,00205	14
50	5	10	0,00224	11

Matlab üzerinde çalıştırılan modeller içerisinde en küçük H.K.O.'ya sahip model 1 Girdi, 1 Gizli, 1 Çıktı Katmanı ve 1 Çıktı'dan oluşmaktadır. Mimari yapı olarak, en küçük H.K.O.'sı 0.00124 olan ve girdi katmanında 5 birim, gizli katmanda 8 birim ve çıktı katmanında 1 birim bulunan 5-8-1 mimari yapısına sahip 48. model kullanılmıştır (Bkz, Şekil 4.15.).



Şekil 4.15. TCMB USD efektif satış kuru için Y.S.A. modeli mimari yapısı

[x] : girdi katmanı birimleri vektörü,

[y] : gizli katman birimleri vektörü,

$w^1$  : girdi katmanı ile gizli katman arasındaki ağırlıklar,

$w^2$  : girdi katmanı ile gizli katman arasındaki ağırlıkları göstermektedir.

### - Yıllık İptal Oranları

Mimari yapı için, Matlab paket programında yer alan hazır araç paketi Neural Networks üzerinde İ.B.Ç.K.Y.S.A. incelenerek, H.K.O.'sı en küçük olan model, uygulamada kullanılmıştır. İptal oranları 1, 2 ve 3 yıllık iptal oranları olarak incelenmiştir.

#### 1 Yıllık İptal Oranları için;

Tüm modellere ilişkin H.K.O. değerleri Çizelge 4.4'te yer almaktadır.

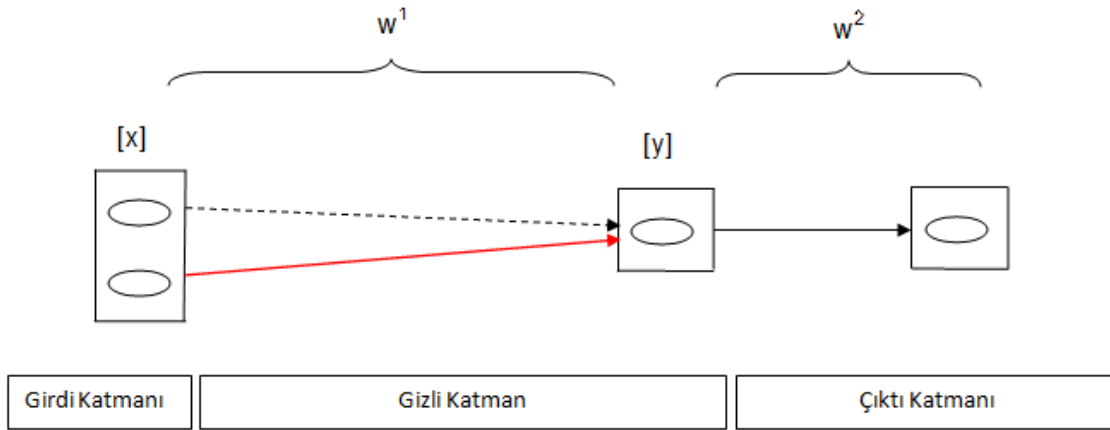
Çizelge 4.4: 1 Yıllık iptal oranı için çalıştırılan modellere ait H.K.O. ve iterasyon sayıları.

Model No	Girdi Katmanı Birim Sayısı	Gizli Katman Birim Sayısı	H.K.O.	İterasyon Sayısı
1	1	1	0,000325	7
2	1	2	0,000363	8
3	1	3	0,000242	10
4	1	4	0,000295	7
5	1	5	0,000145	8
6	2	1	0,000121	12
7	2	2	0,000610	9
8	2	3	0,00111	12
9	2	4	0,000319	7
10	2	5	0,000646	9
11	3	1	0,000210	8
12	3	2	0,000323	7

Çizelge 4.4. devam ediyor.

Model No	Girdi Katmanı Birim Sayısı	Gizli Katman Birim Sayısı	H.K.O.	İterasyon Sayısı
13	3	3	0,000523	7
14	3	4	0,000498	12
15	3	5	0,000257	8
16	4	1	0,000143	10
17	4	2	0,000153	13
18	4	3	0,000299	8
19	4	4	0,000176	10
20	4	5	0,00155	11

En iyi mimari yapı olarak, Şekil 4.16.'da de gösterildiği gibi, girdi katmanında 2 birim, gizli katmanda 1 birim ve çıktı katmanında 1 birim bulunan, 2-1-1 mimari yapısı bulunmuştur.



Şekil 4.16. Tanzim edilen poliçelerin 1 yıllık iptal oranına ilişkin Y.S.A. modeli mimari yapısı

[x] : girdi katmanı birimleri vektörü,

[y] : gizli katman birimleri vektörü,

$w^1$  : girdi katmanı ile gizli katman arasındaki ağırlıklar,

$w^2$  : girdi katmanı ile gizli katman arasındaki ağırlıkları göstermektedir.

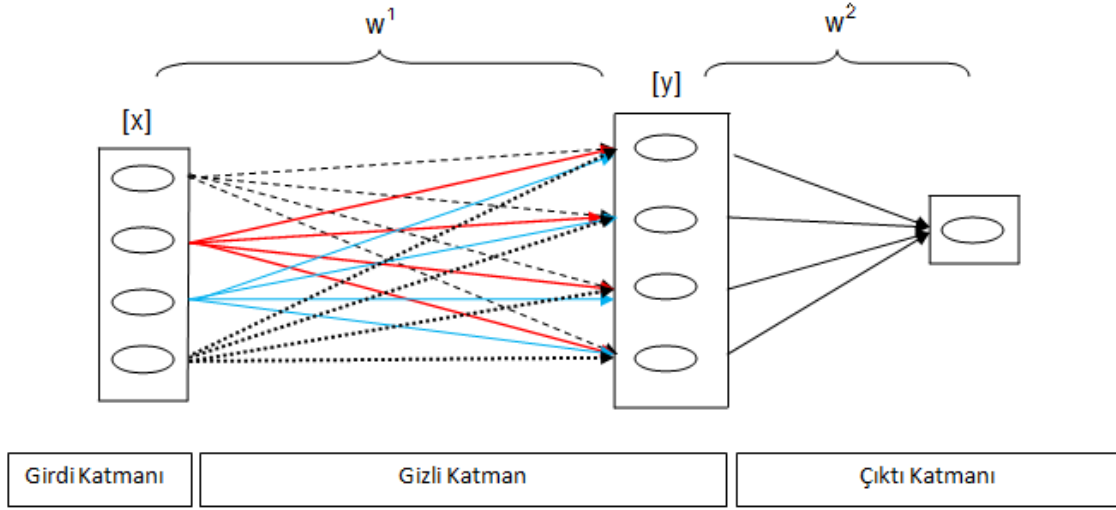
## 2 Yıllık İptal Oranları için;

Tüm modellere ilişkin H.K.O. değerleri Çizelge 4.5'de yer almaktadır.

Çizelge 4.5: 2 Yıllık iptal oranı için çalıştırılan modellere ait H.K.O. ve iterasyon sayıları.

Model No	Girdi Katmanı Birim Sayısı	Gizli Katman Birim Sayısı	H.K.O.	İterasyon Sayısı
1	1	1	0,000171	97
2	1	2	0,000163	10
3	1	3	0,000174	7
4	1	4	0,000164	7
5	1	5	0,000160	8
6	2	1	0,000579	6
7	2	2	0,000134	14
8	2	3	0,000160	8
9	2	4	0,000164	7
10	2	5	0,000131	10
11	3	1	0,000176	10
12	3	2	0,000162	9
13	3	3	0,000172	10
14	3	4	0,000385	6
15	3	5	0,000107	9
16	4	1	0,000146	16
17	4	2	0,000421	7
18	4	3	0,000187	7
19	4	4	0,000103	7
20	4	5	0,000115	8

Mimari yapımızda; girdi katmanında 4 birim, gizli katmanda 4 birim ve çıktı katmanında 1 birim bulunan, 4–4-1 mimari yapısı kullanılmıştır. Mimari yapı, Şekil 4.17.'de gösterilmiştir.



Şekil 4.17. Tanzim edilen poliçelerin 2 yıllık iptal oranına ilişkin Y.S.A. modeli mimari yapısı

[x] : girdi katmanı birimleri vektörü,

[y] : gizli katman birimleri vektörü,

$w^1$  : girdi katmanı ile gizli katman arasındaki ağırlıklar,

$w^2$  : girdi katmanı ile gizli katman arasındaki ağırlıkları göstermektedir.

### 3 Yıllık İptal Oranları için;

Tüm modellere ilişkin H.K.O. değerleri Çizelge 4.6'da yer almaktadır.

Çizelge 4.6: 3 Yıllık iptal oranı için çalıştırılan modellere ait H.K.O. ve iterasyon sayıları.

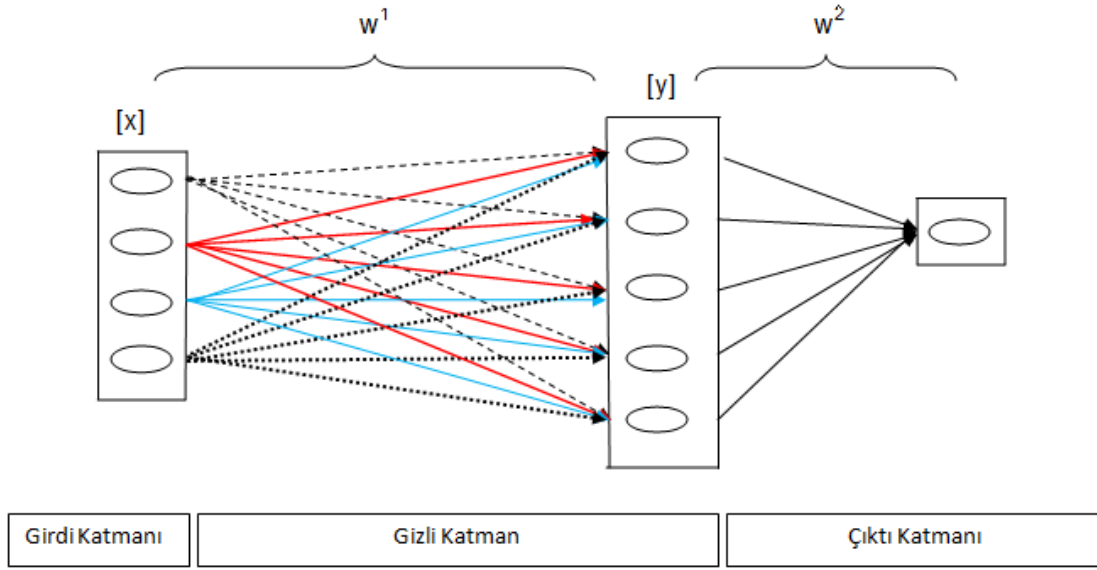
Model No	Girdi Katmanı Birim Sayısı	Gizli Katman Birim Sayısı	H.K.O.	İterasyon Sayısı
1	1	1	0,000190	21
2	1	2	0,000186	10

Çizelge 4.6. devam ediyor.

<b>Model No</b>	<b>Girdi Katmanı Birim Sayısı</b>	<b>Gizli Katman Birim Sayısı</b>	<b>H.K.O.</b>	<b>İterasyon Sayısı</b>
3	1	3	0,000189	7
4	1	4	0,000182	7
5	1	5	0,000183	8
6	2	1	0,000176	9
7	2	2	0,000144	10
8	2	3	0,000146	10
9	2	4	0,000157	7
10	2	5	0,000129	13
11	3	1	0,000176	11
12	3	2	0,000177	8
13	3	3	0,000231	8
14	3	4	0,000417	6
15	3	5	0,000139	8
16	4	1	0,000167	16
17	4	2	0,000174	10
18	4	3	0,000408	7
19	4	4	0,00111	6
20	4	5	0,000128	8

Mimari yapıda; girdi katmanında 4 birim, gizli katmanda 5 birim ve çıktı tabakasında da 1 birimden oluşan, 4–5-1 mimari yapısı kullanılmıştır. Y.S.A. modeli mimari yapısı Şekil 4.18.'de gösterilmiştir.





Şekil 4.18. Tanzim edilen poliçelerin 3 yıllık iptal oranına ilişkin Y.S.A. modeli mimari yapısı

[x] : girdi katmanı birimleri vektörü,

[y] : gizli katman birimleri vektörü,

$w^1$  : girdi katmanı ile gizli katman arasındaki ağırlıklar,

$w^2$  : girdi katmanı ile gizli katman arasındaki ağırlıkları göstermektedir.

#### - TCMB EUR Efektif Satış Kuru

Mimari yapı için, Matlab paket programında yer alan hazır araç paketi Neural Networks üzerinde İBÇ.K.Y.S.A. incelenerek, H.K.O.'sı en düşük olan model, uygulamada kullanılmıştır (Bkz, Çizelge 4.7).

Çizelge 4.7: TCMB EUR efektif satış kuru için çalıştırılan modellere ait H.K.O. ve iterasyon sayıları.

Model No	Girdi Katmanı Birim Sayısı	Gizli Katman Birim Sayısı	H.K.O.	İterasyon Sayısı
1	1	1	0,00396	15
2	1	2	0,00363	16

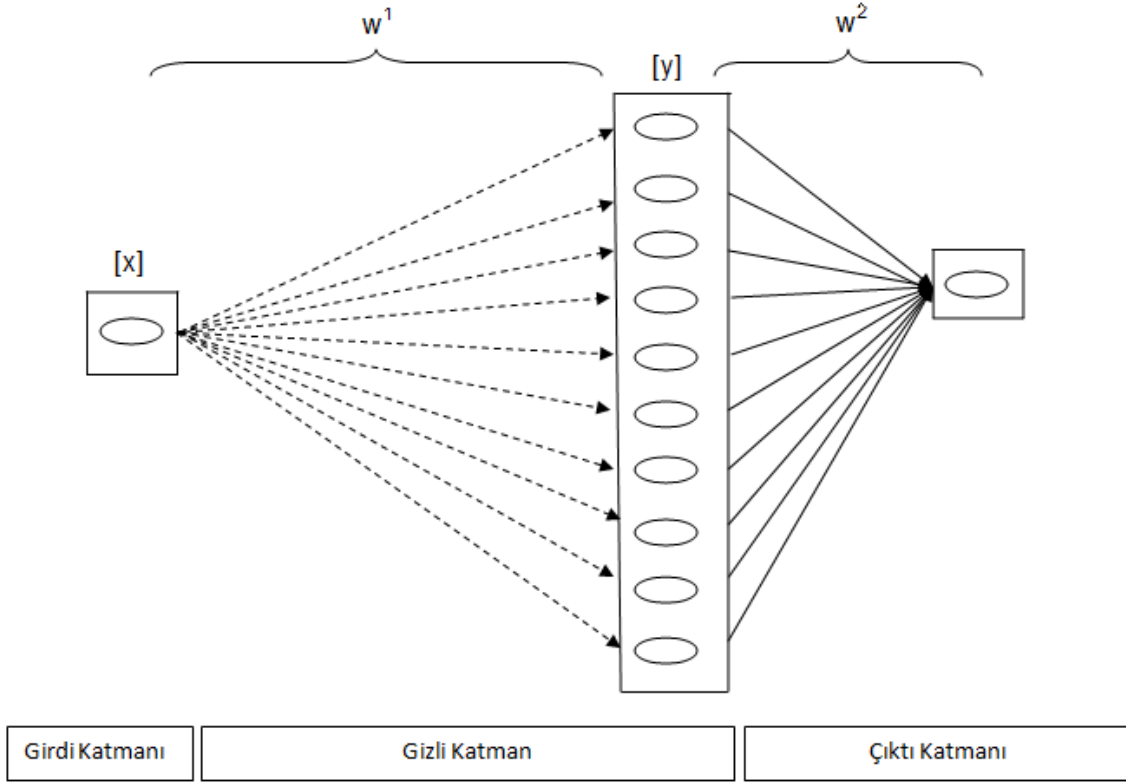
Çizelge 4.7. devam ediyor.

<b>Model No</b>	<b>Girdi Katmanı Birim Sayısı</b>	<b>Gizli Katman Birim Sayısı</b>	<b>H.K.O.</b>	<b>İterasyon Sayısı</b>
3	1	3	0,00290	22
4	1	4	0,00281	14
5	1	5	0,00421	10
6	1	6	0,00324	10
7	1	7	0,00410	8
8	1	8	0,00348	8
9	1	9	0,00340	13
10	1	10	0,00222	10
11	2	1	0,00378	24
12	2	2	0,00388	17
13	2	3	0,00339	18
14	2	4	0,00373	17
15	2	5	0,00332	55
16	2	6	0,00295	18
17	2	7	0,00333	9
18	2	8	0,00318	23
19	2	9	0,00255	14
20	2	10	0,00336	13
21	3	1	0,00376	20
22	3	2	0,00317	57
23	3	3	0,00546	9
24	3	4	0,00328	14
25	3	5	0,00355	9
26	3	6	0,00314	12
27	3	7	0,00371	12
28	3	8	0,00321	31
29	3	9	0,00306	17
30	3	10	0,00327	13
31	4	1	0,00378	29
32	4	2	0,00326	20
33	4	3	0,00305	13
34	4	4	0,00299	14

Çizelge 4.7. devam ediyor.

<b>Model No</b>	<b>Girdi Katmanı Birim Sayısı</b>	<b>Gizli Katman Birim Sayısı</b>	<b>H.K.O.</b>	<b>İterasyon Sayısı</b>
35	4	5	0,00398	11
36	4	6	0,00533	11
37	4	7	0,00328	13
38	4	8	0,00317	27
39	4	9	0,00391	9
40	4	10	0,00227	14
41	5	1	0,00372	36
42	5	2	0,382	16
43	5	3	0,00271	12
44	5	4	0,00262	21
45	5	5	0,00332	13
46	5	6	0,00301	24
47	5	7	0,00252	29
48	5	8	0,00248	13
49	5	9	0,00328	11
50	5	10	0,00300	10

Mimari yapı olarak Şekil 4.19.'da belirtilen, TCMB EUR efektif satış kuru için çalıştırılan modellere ait H.K.O. ve iterasyon sayılarının yer aldığı çizelge sonucuna göre, girdi katmanında 1, gizli katmanda 10 ve çıktı katmanında 1 birim yer alan, 1–10-1 mimari yapısı kullanılmıştır.



Şekil 4.19. TCMB EUR efektif satış kuru için Y.S.A. modeli mimari yapısı

[x] : girdi katmanı birimleri vektörü,

[y] : gizli katman birimleri vektörü,

$w^1$  : girdi katmanı ile gizli katman arasındaki ağırlıklar,

$w^2$  : girdi katmanı ile gizli katman arasındaki ağırlıkları göstermektedir.

### Tahmin Değerleri

#### - TCMB USD Efektif Satış Kuru

Kurulan Y.S.A. modeline ilişkin öngörü değerleri, çıktı değerinden elde ettiğimiz son 32 değer olacaktır. Böylece 30.09.2009 tarihinden itibaren 32 aylık ay sonu TCMB USD efektif satış kuru değerlerine ulaşılmış olacaktır. Bu değerler Çizelge 4.8'de gösterilmiştir.

Çizelge 4.8: TCMB USD efektif satış kuru tahmin değerleri.

<b>Tahmin Edilen Tarihler</b>	<b>Tahmin Değerleri (TCMB USD Efektif Satış Kuru)</b>
31.10.2009	1,3177
30.11.2009	1,3689
31.12.2009	1,3194
31.01.2010	1,6927
28.02.2010	1,4804
31.03.2010	1,4344
30.04.2010	1,4961
31.05.2010	1,5284
30.06.2010	1,4775
31.07.2010	1,4641
31.08.2010	1,4007
30.09.2010	1,4302
31.10.2010	1,3979
30.11.2010	1,3918
31.12.2010	1,3239
31.01.2011	1,3266
28.02.2011	1,3170
31.03.2011	1,3133
30.04.2011	1,3526
31.05.2011	1,1857
30.06.2011	1,1857
31.07.2011	1,2419
31.08.2011	1,1703
30.09.2011	1,2373
31.10.2011	1,2476
30.11.2011	1,4164
31.12.2011	1,3177

Çizelge 4.8. devam ediyor.

<b>Tahmin Edilen Tarihler</b>	<b>Tahmin Değerleri (TCMB USD Efektif Satış Kuru)</b>
31.01.2012	1,1802
29.02.2012	1,2600
31.03.2012	1,2188
30.04.2012	1,2081
31.05.2012	1,3317
30.06.2012	1,7142
31.07.2012	1,5353
31.08.2012	1,6638
30.09.2012	1,7102

**- Yıllık İptal Oranları**

Kurulan Y.S.A. modeline ilişkin öngörü değerleri çıktı değerinden elde ettiğimiz son değer olacaktır. Böylece 30.09.2009 tarihinden itibaren 1 yıl için yıllık iptal oranı olarak 0,017297 değerine ulaşılmış olunacaktır. Bu değer aşağıda gösterilmiştir.

<b>İptal Edilme Dönemi</b>	<b>Tahmin Değerleri (Yıllık İptal Oranı)</b>
30.09.2009 -30.09.2010	0,017297

2 yıllık iptal oranları için, 19. modelin en H.K.O.'ya sahip olduğu görülmüştür. Bu değer, 0,00103'tür. Böylece kullanılacak mimari yapı 4-4-1 olarak belirlenmiştir. Elde edilen modelde, poliçenin 2 yıl içerisinde iptal olma oranı ise aşağıdaki tabloda gösterilmiştir.

<b>İptal Edilme Dönemi</b>	<b>Tahmin Edilen Değerler (2 Yıllık İptal Oranı)</b>
30.09.2009 - 30.09.2011	0,0012

3 yıllık iptal oranları içinse, 20. modelin en küçük H.K.O.'ya sahip olduğu görülmüştür. Bu değer, 0,000128'dir. Böylece kullanılacak mimari yapı 4-5-1 olarak belirlenmiştir. Y.S.A. modeliyle elde edilen poliçenin 3 yıl içerisinde iptal olma oranı ise aşağıdaki tabloda gösterilmiştir.

<b>İptal Edilme Dönemi</b>	<b>Tahmin Edilen Değerler (3 Yıllık İptal Oranı)</b>
30.09.2009 -30.09.2012	0,0017

#### - TCMB EUR Efektif Satış Kuru

Kurulan Y.S.A. modeline ilişkin öngörü değerleri çıktı değerinden elde ettiğimiz son 32 değer olacaktır. Böylece 30.09.2009 tarihinden itibaren 32 aylık ay sonu TCMB EUR efektif satış kur değerlerine ulaşılmış olacaktır. Bu değerler Çizelge 4.9'da gösterilmiştir.

Çizelge 4.9: TCMB EUR efektif satış kuru tahmin değerleri.

<b>Tahmin Edilen Tarihler</b>	<b>Tahmin Edilen Değerler (TCMB EUR Efektif Satış Kuru)</b>
31.10.2009	1,87200
30.11.2009	1,84100
31.12.2009	1,89596
31.01.2010	1,84401
28.02.2010	1,84019
31.03.2010	1,83863
30.04.2010	1,83802
31.05.2010	1,82476
30.06.2010	1,80878
31.07.2010	1,79170
31.08.2010	1,80714
30.09.2010	1,82164
31.10.2010	1,73045

Çizelge 4.9. devam ediyor.

<b>Tahmin Edilen Tarihler</b>	<b>Tahmin Edilen Değerler (TCMB EUR Efektif Satış Kuru)</b>
30.11.2010	1,72914
31.12.2010	1,77959
31.01.2011	1,73277
28.02.2011	1,76508
31.03.2011	1,80679
30.04.2011	2,04259
31.05.2011	1,99836
30.06.2011	1,85404
31.07.2011	1,90288
31.08.2011	1,84242
30.09.2011	1,76197
31.10.2011	1,81937
30.11.2011	1,94713
31.12.2011	2,02328
31.01.2012	2,15176
29.02.2012	2,13830
31.03.2012	2,15084
30.04.2012	2,12650
31.05.2012	2,14264
30.06.2012	2,15327
31.07.2012	2,15264
31.08.2012	2,12094
30.09.2012	2,15125



## 5. SONUÇ VE ÖNERİLER

Y.S.A.'lar, ekonomi ve finans alanında tercih edilen, hem eğrisel hem de doğrusal yapıyı modelleyebilen ve diğer yöntemlere göre daha hızlı yakınsayan bir yöntemdir. Y.S.A.'na ait özellikler de kullanım açısından bir diğer tercih nedeni olmaktadır. Bu özelliklerden en önemlileri bir kez eğitildikten sonra eksik veriyle de çalışıyor olması, kendi kendine öğreniyor olmasıdır.

Yapılan çalışmada Y.S.A. modeli kullanılarak TCMB USD efektif satış kuru, TCMB EUR efektif satış kuru ve ileriye dönük 1,2 ve 3 yıllık iptal oranları öngörülme çalışılmıştır.

USD ve EUR verileri için geçmişe ait veriye ulaşmanın daha az zaman alıcı olması ve daha çok veri kullanmak amacıyla 2003 yılından itibaren ay sonlarına ait veriler kullanılmış; iptal oranları için hem poliçe sürelerinin ortalama 3 yıl olması hem de geçmiş veriye ulaşma zorluğu nedeniyle 31.08.2006 ve sonrası aylara ait veriler kullanılmıştır.

Elde edilen sonuçlar ve tahmin değerleri incelendiğinde, uygun tahmin değerlerinin elde edildiği görülmüştür. Bu veriler kullanılarak sigorta şirketindeki kredi kaynaklı tanzim edilen uzun süreli hayat sigorta portföyünün 3 yıl sonraki kârı bulunabilir.

Çalışmanın bir sonraki aşaması için, hayat sigortası ya da hayat dışı sigorta şirketinde tüm portföy için daha uzun vadeli kârların bugünkü değeri için kullanılacak parametreler de aynı yöntem kullanılarak hesaplanabilir.

## KAYNAKLAR DİZİNİ

- Aksöyek, İ., 1994, Halka Açılan Şirketlerde Firma Değerinin Belirlenmesine Yönelik Yaklaşımların Mukayeseli Olarak İncelenmesi, Doktora Tezi, Gazi Üniversitesi, Sosyal Bilimler Enstitüsü, Ankara,
- Akgüç, Ö., 1994, Finansal Yönetim, 6. Baskı, Avcıol Basım Yayın, İstanbul, s. 141.
- Aladağ, Ç.H., Eğrioğlu, E., Kadılar, C. 2009, Forecasting Nonlinear Time Series with a Hybrid Methodology, Applied Mathematics Letters, 22, p. 1467-1470.
- Aladağ, Ç.H., Eğrioğlu, E., Günay, S., 2007, Yapay Sinir Ağları ile Türkiye Elektrik Tüketiminin Tahmini, TÜİK 16. İstatistik Araştırma Sempozyumu Bildiriler Kitabı, s. 181-191.
- Alkan, A., 2001, Predictive Data Mining with Neural Networks and Genetic Algorithms, Ph.D. Thesis, İTÜ, İstanbul, p. 51.
- Alpaydın, E., 1990, Neural Models of Incremental Supervised and Unsupervised Learning, Doktora tezi, Ecole Polytechnique Federale de Lausanne, İsviçre, p. 839.
- Aydoğmus, Z., Çöteli, R., 2005, Yapay Sinir Ağları Yardımıyla İzolatör Yüzeyinde Potansiyel Tahmini, F. Ü. Fen ve Müh. Bil. Dergisi, 17;2, s. 239-246.
- Battiti R., 1992, First and Second-Order Methods for Learning Between Steepest Descent and Newton's Method, Neural Computation, Vol. 4, p. 141–166.
- Beltratti, A., Margarita, S., Terna P., 1996, Neural Networks for Economic and Financial Modelling. Thomson-ITCP, London.
- Bishop, C.M., 1995, Neural Networks for Pattern Recognition, Oxford University Press.
- Bolat, S., Kalenderli, Ö., 2003, Levenberg-Marquardt Algoritması Kullanılan Yapay Sinir Ağı ile Elektrot Biçim Optimizasyonu, İstanbul Üniversitesi Elektrik - Elektronik Fakültesi, Elektrik Mühendisliği Bölümü, İstanbul.
- Chowdary, B.V., 2007, Back-Propagation Artificial Neural Network Approach for Machining Centre Selection, Journal of Manufacturing Technology Management, 18 (3), p. 315-332.
- Çakar, T., Türker, A.K., Toraman, A., 1996, İmalat Sistemlerinin Tasarlanmasında Yapay Sinirsel Ağların Kullanılması, Sakarya.
- Çelebi, D., Bayraktar, D., 2008, An Integrated Neural Network and Data Envelopment Analysis for Supplier Evaluation Under Incomplete Information, Expert Systems with Applications, 35, p. 1698–1710.
- Daniel B., 1995, The Digital MBA, McGraw - Hill Company, California, p. 308-310.

- Efe, Ö., Kaynak O., 2000, Yapay Sinir Ağları ve Uygulamaları, Boğaziçi Üniversitesi Yayınları, Basım, İstanbul, s. 148.
- Elmas, Ç., 2003, Yapay Sinir Ağları, Seçkin Yayıncılık, Ankara, s. 21-39.
- Ergezer, H., Dikmen, M., Özdemir, E., 2003, Yapay Sinir Ağları ve Tanıma Sistemleri, Pivolka, s. 14-17.
- Erdoğan, G., 2006, Yapay Sinir Ağları İle İktisadi Ve Finansal Zaman Dizilerinin Kestirimi, Yüksek Lisans Tezi, Gazi Üniversitesi Fen Bilimleri Enstitüsü, Ankara.
- Fausett, L., 1994, Fundamentals of Neural Networks: Architectures, Algorithms and Applications, Prentice Hall, p. 294-296.
- Günay, S., Eğrioğlu, E., Aladağ, Ç.H., 2007a, Tek Değişkenli Zaman Serileri Analizine Giriş, Hacettepe Üniversitesi Yayınları, Ankara.
- Günay, S., Eğrioğlu, E., Aladağ, Ç.H., 2007b, Yapay Sinir Ağları ile Türkiye Elektrik Tüketiminin Tahmini, TÜİK 16. İstatistik Araştırma Sempozyumu Bildiriler Kitabı.
- Halaç, A., Erbil, T., Falay, T., 2002, Yapay Zeka , PC NET, s. 59.
- Hayat Sigortası Genel Şartları, 1995, s. 5.
- Haykin, S., 1994, Neural networks: A comprehensive foundation, MacMillan College Publishing Comp. Inc., New York.
- Haykin, S., 1999a, Learning processes; single – layer perceptrons; multilayer perceptrons. Neural Networks: A Comprehensive Foundation. 2'nd edition, Prentice Hall International Inc, USA, p. 14-68.
- Haykin, S., 1999b, Neural Networks: a Comprehensive Foundation, Upper Saddle River, N.J., Prentice Hall.
- Hooman, Y., Zilouchian A., 2001, Neural Network Architecturer, Chapter 3, CRC Press LLC.
- Hosein, P., 2002, Stock Price Prediction By Artificial Neural Networks: A Study of Tehran's Stock Exchange, Management Studies Quarterly Journal of Management & Accounting School, p. 31-32.
- Jorgensen C., 2006. DCS-Neural-Network Program for Aircraft Control and Testing, Ames Research Center, NASA, Moffett Field, California.
- Kaastra I., Boyd, M., 1996, Designing a Neural Network for Forecasting Financial and Econometric Time Series, Neurocomputing, Vol. 10, p. 215-236.
- Lipmann, R. P., 1987, An Introduction to Computing with Neural Nets. IEEE ASSP Magazine, April, 4-22.

- Mandic, D.P., Chambers, J. A., 2001, Recurrent Neural Networks for Prediction - Learning Algorithms Architectures and Stability, John Wiley & Sons Ltd.
- Mehr R., Cammack E., 1972, Principles of Insurance Fifth Edition, Richard D Irwin Inc., Ontario, p.34.
- Minai, A. A., Williams, R.D., 1990. Back-propagation Heuristics: A Study of The Extended Delta-Bar-Delta Algorithm. Proceedings of the International Joint Conference on Neural Networks, vol. 1, San Diego, CA, 17-21 June, p. 595-600.
- Nabiyev, V.V., 2003, Yapay Zeka, Seçkin Yayıncılık, Ankara.
- Ntv Bilim Dergisi, 2009, sayı:1,2009.
- Öner, E., 1998, İşletme Yönetiminde Sistem Yaklaşımı, 3.Basım, Alfa A.Ş, İstanbul.
- Öztemel E., 2003, Yapay Sinir Ağları, Papatya Yayıncılık, s. 67-97-39, İstanbul.
- Papik K., Molnar B., Schaefer R., Dombovari Z., Tulassay Z., 1998, Feher Journal Application of Neural Networks in medicine-a review. Med Sci Mon, s. 538-46.
- Saritas, İ., 2008, The Design of Magnetic Filter for Cleaning Industrial Liquids and Adaptive Control of Filter by Artificial Intelligence Methods, Phd Thesis, Selcuk University Institute of the Natural and Applied Sciences.
- Sağiroğlu S., 1996, Levenberg-Marquardt Metodunun Bir Robot Sensörün Yapay Sinir Ağı Modellenmesinde Kullanılması, İstanbul, s. 323-328.
- Tebelkis, J., 1995, Speech Recognition Using Neural Networks, Ph.D. Thesis, Carnegie Mellon University, Pennsylvania.
- Trafalis, T.B., Couellan, N., 1998, An Incremental Nonlinear Primal-Dual Algorithm and Applications to Artificial Neural Networks Training, Proceedings of LSS'98, preprints, Volume II, p. 1053-1059.
- Verleysen, M., Hlavackova, K., 1994, An Optimized RBF Network For Approximation of Functions, Proceedings European Symposium on Artificial Neural Networks, Brussels, Belgium, p. 175-180.
- White, H., 1990, Connectionist Non-parametric Regression : Multilayer Feedforward Networks Can Learn Arbitrary Mappings, Neural Networks, p. 535-550.
- Yazıcı, A.C., Ögüş, E., Ankaralı, Seyit, Canan, S., Ankaralı, H., Akkuş, Z., 2007, Yapay Sinir Ağlarına Genel Bakış.
- Yurtoğlu, H., 2005, Yapay Sinir Ağları Metodolojisi İle Öngörü Modellemesi: Bazı Makroekonomik Değişkenler İçin Türkiye Örneği, Uzmanlık Tezi. DPT.

Zhang, G., Patuwo, B. E., HU, Y. M., 1998, Forecasting with Artificial Neural Networks. International Journal of Forecasting, 14, p. 35-62.

Zurada, M. J., 1992, Introduction to Artificial Neural Systems. West Publishing Company, Inc. New York.

İnternet (www/ftp) kaynakları:

<http://www.tsrbs.org.tr/tsrsb/Templates/innerpage>

## ÖZGEÇMİŞ

Adı Soyadı : Gülay DOĞAN

Doğum Yeri : Ankara

Doğum Yılı : 1978

Medeni Hali : Bekar

Eğitim ve Akademik Durumu:

Lise 1992 - 1996 Çankaya Lisesi, Ankara

Lisans 1996 - 2001 Hacettepe Üniversitesi, İstatistik Bölümü, Ankara

Yabancı Dil: İngilizce

İş Tecrübesi:

2003 - 2006 Axa Oyak Hayat Sigorta A.Ş., İstanbul, Uzman Yardımcısı,

2006 - 2009 Axa Hayat Sigorta A.Ş., İstanbul, Uzman,

2009 - ... Axa Hayat Sigorta A.Ş., İstanbul, Müdür Yardımcısı,