

T.C.
HALIÇ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
YÖNETİM BİLİŞİM SİSTEMLERİ PROGRAMI

KREDİ KARTI İÇİN
YAPAY SİNİR AĞLARI (YSA)
UYGULAMASI

YÜKSEK LİSANS TEZİ

Hazırlayan
YUSUF VURAL

Tez Danışmanı
Prof.Dr. ALİ OKATAN

Ocak 2008

İSTANBUL

T.C.
HALIÇ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ MÜDÜRLÜĞÜNE

Yönetim Bilişim Sistemleri Programı Yüksek Lisans öğrencisi Yusuf VURAL tarafından hazırlanan “**Kredi Kartı İçin Yapay Sinir Ağları (YSA) Uygulaması**” adlı bu çalışma jürimizce Yüksek Lisans Tezi olarak kabul edilmiştir.

Tez Savunma Tarihi : 14.01.2008

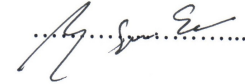
(Jüri Üyesinin Ünvanı , Adı , Soyadı ve Kurumu) :

İmzası :


Jüri Üyesi: Prof.Dr.Ali OKATAN
(Danışman-HÜ.Bil.Mühendisliği)



Jüri Üyesi : Prof.Dr.Sami ERCAN
(H.Ü.End.Mühendisliği)



Jüri Üyesi : Yrd.Doç Dr.Yüksel BAL
(H.Ü.Bil.Mühendisliği)



İÇİNDEKİLER

ÖNSÖZ	iii
ÖZET	iv
ABSTRACT	v
ŞEKİL LİSTESİ	vi
TABLO LİSTESİ	vii
GİRİŞ	1
1. YAPAY SİNİR AĞLARI	2
1.1. Yapay Sinir Ağları Tanım	2
1.2. Yapay Sinir Ağlarının Avantajları	2
1.3. Yapay Sinir Ağlarının Dezavantajları	4
1.4. YSA'ların Kullanıldığı Alanlar	5
1.5. Beyin Sinir Hücresinin Yapısı	7
1.6. Yapay Sinir Hücresi	8
1.6.1 Girdiler	8
1.6.2 Ağırlıklar	8
1.6.3 Toplama Fonksiyonu	8
1.6.4 Etkinlik Fonksiyonu	9
1.6.5 Hücrenin Çıktısı	9
1.7. Yapay Sinir Ağının Yapısı	9
1.7.1 Katmanlar	10
1.7.2 Katmanlar Arası Bağlantılar	10

2. AĞ YAPILARI	12
2.1. Perceptron (Basit Algılayıcı) Ağı	13
2.2. Çok Katmanlı Perceptron (Algılayıcı) Ağı	13
2.3. Hopfield Ağı	13
2.4. Hamming Ağı	14
2.5. Karşı – Yayma Ağı	14
2.6. Geri Yayılım Ağı	14
2.7. Delta Bar Delta	15
3. YAPAY SİNİR AĞLARINDA ÖĞRENME	15
3.1 Danışmalı Öğrenme	17
3.2 Danışmansız Öğrenme	18
3.3 Karma Öğrenmek Kuralı	19
3.4 Danışmanlı Öğrenme Yöntemlerinden Delta Öğrenme Kuralı	19
3.5 Geri Yayılımlı Öğrenme	22
3.5.1 Geri Yayılım Algoritması	25
4. BANKACILIK ALANINDA YAPAY SİNİR AĞLARI	26
4.1 Kredi Kartı Uygulaması	28
4.2 Sonuç	30
KAYNAKLAR.....	33
ÖZGEÇMİŞ.....	35

ÖNSÖZ

Bu tezin hazırlanması sırasında öncelikle danışmanlık yaparak değerli bilgi ve desteğini esirgemeyen sayın Prof. Dr. Ali Okatan'a, yardımları için Öğr. Gör. Oğuz Karan'a ve destekleri için eşim Neslihan Vural'a çok teşekkür ederim.

Yusuf VURAL
Ocak 2008

T.C.
HALIÇ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
YÖNETİM BİLİŞİM SİSTEMLERİ PROGRAMI
YÜKSEK LİSANS TEZİ

KREDİ KARTI İÇİN YAPAY SINIR AĞLARI (YSA) UYGULAMASI

YUSUF VURAL

Tez Danışmanı : Prof.Dr. ALİ OKATAN

Ocak 2008

ÖZET

Değişen koşullar ve tüketim alışkanlıkları (kredi kartının, kredi kartı ile taksitli alışverişin yaygınlaşması), bankaların müşterilerinin risklerini güncel olarak değerlendirmesini gerektirmektedir.

Kredi kartı talebinde bulunan müşteriler bir çok yönden değerlendirilerek uygun ise kart sahibi olabilmektedirler. Bu alandan YSA kullanılması ise günün hızlı değişen koşullarına göre müşterilerin davranış tahminlerini edinmemizi sağlayarak risklerimizi azaltabilir.

Bu çalışmada müşterinin cinsiyeti, medeni durumu, gelir durumu, meslek grubu ve kredi kartı borcu ödeme alışkanlıkları veri olarak alınmıştır. Bu verilere göre yeni müşteriler için davranış tahmini elde edilmiştir.

Anahtar kelimeler: Yapay sinir ağları, kredi kartları.

T.C.
HALIÇ UNIVERSITY
INSTITUTE OF SCIENCE
COMPUTER ENGINEERIN
MANAGEMENT OF INFORMATION SYSTEMS
MASTER THESIS

APPLICATION OF ARTIFICIAL NEURAL NETWORKS
FOR CREDIT CARDS

YUSUF VURAL

Supervisor : Prof.Dr. ALİ OKATAN

Ocak 2008

ABSTRACT

With the changing conditions and consumption patterns, e.g.usage of credit cards and hire-purchasing with credit cards those becoming widespread , banks get more need to evaluate the actual customer risks.

The customers demanding credit cards, are evaluated in many aspects and if the results are evaluated as appropriate, they become card-holders. The application of artificial neural networks can enable us reduce the risks by making it possible to foresee the customer behaviours influenced by rapidly changing daily conditions.

In this study, customer's gender, marital status, income, profession and credit card paying habits are taken as data. According to these data, new customers behaviors are predicted.

Keywords: Artificial neural networks, credit cards.

ŞEKİL LİSTESİ

Şekil 1.5.1 Sinir Hücresi Yapısı	6
Şekil 3.4.1 Hata değişimi, Ağırlık vektörü ve eğilimli inişin geometrik yorumu....	20
Şekil 3.5.1 İleri beslemeli çok katmanlı sinir ağı.	22
Şekil 4.2.1 Neurosolution Programına göre elde edilen Hata Grafiği	30
Şekil 4.2.1 Neurosolution Programına göre elde edilen Sonuç Tablosu.....	31

TABLO LİSTESİ

Tablo 1.1 Geleneksel algoritmalar ile YSA'lar	5
Tablo 2.1.1 XOR problemi	13
Tablo 4.1 Cinsiyet Referans Aralığı	24
Tablo 4.2 Medeni Durum Referans Aralığı	24
Tablo 4.3 Gelir Referans Aralığı	25
Tablo 4.4 Meslek Referans Aralığı	25
Tablo 4.5 Ödeme Referans Aralığı	25

GİRİŞ

Ülkemizde ekonomik gelişmeler, teknolojik gelişmelerle birlikte tüketim alışkanlıkları da değişmiştir. Bunun bir sonucu olarak kredi kartları nakit harcamaların yerini almaya başlamıştır. Bankacılık alanında bireysel kredilerin yaklaşık %30 nu kredi kartları harcamaları tutmaktadır (Türkiye Bankalar Birliği 2007 ilk 9 aylık istatistik raporlarına göre). Bankalar yeni müşteriler kazanmak için mücadele ederken risklerinin de artmamasını amaçlamaktadırlar. Bu nedenle kullandıkları değerlendirme kriterleri ve puanlama sistemleri vardır.

Bu alanda yapay sinir ağlarının kullanılması bankaların eski müşterilerinin alışkanlıklarına göre yeni müşterilerinin davranışlarını önceden tahmin etmesini sağlayacaktır. Değişen koşullara göre müşteri davranışlarında değişme olduğunda yeni verilerle beslenen ağ her zaman doğru sonuçlara yakın değerler elde edilmesini sağlayacaktır.

Çalışmada Türkiye’de özel bir bankanın 2007 yılına ait rastgele seçilmiş müşterilerinin; kredi kartı borçlarını düzenli ödeme bilgileri ve müşterilerin cinsiyetleri, medeni halleri, gelir durum aralıkları ve meslek grupları dikkate alınmıştır. Buradaki amaç YSA’larının bu alanda kullanılabilirliği göstermek olduğundan karar alma sürecini etkileyen tüm kriterler dikkate alınmayıp sadece bunlar dikkate alınmıştır.

1 YAPAY SİNİR AĞLARI

1.1. Yapay Sinir Ağları Tanım

Yapay sinir ağları (YSA), insan beyninin çalışmasından esinlenerek geliştirilmiş, öğrenme yolu ile yeni bilgiler türetebilme, geliştirebilme gibi yetenekleri gerçekleştirmek amacıyla geliştirilen bilgisayar sistemleridir. Yapay sinir ağları, biyolojik sinir ağlarını (insan beyni) taklit eden bilgisayar programlarıdır.

Yapay sinir ağları; insan beyninden esinlenerek, öğrenme sürecinin matematiksel olarak modellenmesi uğraşı sonucu ortaya çıkmıştır. Bu nedendir ki, bu konu üzerindeki çalışmalar ilk olarak beyni oluşturan biyolojik üniteler olan nöronların modellenmesi ve bilgisayar sistemlerinde uygulanması ile başlamış, daha sonraları bilgisayar sistemlerinin gelişimine de paralel olarak bir çok alanda kullanılabilir hale gelmiştir.

İnsan beyninin çalışma prensibini taklit ederek çalışan bu sistemler, her ne kadar bilgisayar teknolojisi hızlı bir gelişim göstermiş, işlem hızları nano saniyeler mertebesine inmiş olsa da, ilkel bir canlı beyninin fonksiyonları dahi baz alındığında, böyle bir organizmanın yanında çok ilkel kalmaktadır. Nano saniyeler bazındaki işlem hızları ile YSA'ları, mili saniyeler mertebesindeki işlem hızları ile işlem yapan insan beyninin işlevselliğinin henüz çok uzağındadır.

İnsan beyninde yaklaşık 10-12 Milyar arası sinir hücresinin varlığından bahsedilmekle birlikte, bu sayının bilgisayar ortamında modellenmesi şu an için mümkün görünmemektedir. Yine de, YSA'ları yapısallıkları ve hassas eşleştirmeleri başarı ile gerçekleştirebilmeleri ile gün geçtikçe daha fazla uygulama alanı bulmaktadır.

1.2. Yapay Sinir Ağlarının Avantajları

YSA'ları, makina öğrenmesi gerçekleştirebilirler. Yapay sinir ağlarının temel işlevi bilgisayarın öğrenmesini sağlamaktır. Olayları öğrenerek benzer olaylar karşısında mantıklı kararlar verebilirler.

Bilgi işleme yöntemleri geleneksel programlamadan farklıdır. Bu nedenle geleneksel programlamanın getirdiği bir çok olumsuzluk ortadan kaldırılabilir.

Bilgiler ağın tamamında saklanır. Geleneksel programlamada olduğu gibi, bilgiler veri tabanları yada dosyalarda belli bir düzende tutulmaz, ağın tamamına yayılarak değerler ile ölçülen ağ bağlantılarında saklanmaktadır. Hücrelerden bazılarının işlevini yitirmesi, anlamlı bilginin kaybolmasına neden olmaz.

Örnekleri kullanarak öğrenirler. YSA'nın öğrenebilmesi için örneklerin belirlenmesi, bu örneklerin ağa gösterilerek istenilen çıktılara göre ağın eğitilmesi gerekmektedir. Ağın başarısı, seçilen örnekler ile doğru orantılıdır, ağa olay bütün yönleri ile gösterilemezse ağ yanlış çıktılar üretebilir.

Daha önce görülmemiş örnekler hakkında bilgi üretebilirler. YSA'ları eğitimleri sırasında kendilerine verilen örneklerden genellemeler çıkarırlar ve bu genellemeler ile yeni örnekler hakkında bilgi üretebilirler.

Algılamaya yönelik olaylarda kullanılabilirler. YSA'larının en başarılı oldukları alanlar, algılamaya yönelik uygulama alanlarıdır. Bu alanlarda başarıları kanıtlanmıştır.

Örüntü (pattern) ilişkilendirme ve sınıflandırma yapabilirler. YSA'ları kendilerine örnekler halinde verilen örüntüleri kendisi veya diğerleri ile ilişkilendirebilir. Ayrıca kendisine verilen örneklerin kümelenmesi ile, bir sonraki verinin hangi kümeyle dahil olacağını karar verilmesi konusunda kullanılabilirler.

Örüntü tamamlama yapabilirler. Ağ eksik bilgileri içeren örüntüler verildiğinde eksik bilgilerin tamamlanması konusunda başarılıdırlar.

Kendi kendine öğrenebilme ve organize etme yetenekleri vardır. YSA'ları online olarak öğrenebilirler ve kendi kendilerini eğitebilirler.

Eksik bilgi ile çalışabilmektedirler. Geleneksel sistemlerin aksine YSA'ları eğitildikten sonra veriler eksik bilgi içerse dahi, çıktı üretebilirler. Bu durum bir performans kaybı yaratmaz, performans kaybı eksik bilginin önemine bağlıdır. Burada bilgilerin önem dereceleri eğitim sırasında öğrenilir.

Hata toleransına sahiptirler. YSA'larının, eksik bilgilerle çalışabilmeleri ve bazı hücreleri bozulsa dahi çalışabilmeleri, onları hatalara karşı toleranslı yapar.

Dereceli bozulma (Graceful degradation) gösterirler. Bir ağ, zaman içerisinde yavaş ve göreceli bir bozulmaya uğrar. Ağlar, problemin ortaya çıktığı anda hemen bozulmazlar.

Dağıtık belleğe sahiptirler. YSA'larında bilgi, ağa dağılmış bir şekilde tutulur. Hücrelerin bağlantı ve ağırlık dereceleri, ağın bilgisini gösterir. Bu nedenle tek bir bağlantının kendi başına anlamı yoktur.

Yukarıda çok temel bazı avantajlardan bahsedilmekle beraber, YSA'larının daha pek çok avantajı vardır.

1.3. Yapay Sinir Ağlarının Dezavantajları

YSA'larının, pek çok avantajının yanında bazı dezavantajları da vardır. Belli başlı dezavantajları;

Donanım bağımlıdır. YSA'larının en önemli sorunu donanım bağımlı olmalarıdır. YSA'larının en önemli özellikleri ve varoluş nedenlerinden birisi olan paralel işlem yapabilme yeteneği, paralel çalışan işlemciler ile performans gösterir.

Uygun ağ yapısının belirlenmesinde belli bir kural yoktur. YSA'larında probleme uygun ağ yapısının belirlenmesi için geliştirilmiş bir kural yoktur. Uygun ağ yapısı deneyim ve deneme yanılma yolu ile belirlenmektedir.

Ağın parametre değerlerinin belirlenmesinde belli bir kural yoktur. YSA'larında öğrenme katsayısı, hücre sayısı, katman sayısı gibi parametrelerin belirlenmesinde belirli bir kural yoktur. Bu değerlerin belirlenmesi için belirli bir standart olmamakla birlikte, her problem için farklı bir yaklaşım söz konusu olabilmektedir.

Öğrenilecek problemin ağa gösterimi önemli bir problemdir. YSA'ları nümerik bilgiler ile çalışabilmektedirler. Problemler YSA'larına tanıtılmadan önce nümerik değerlere çevrilmek zorundadırlar. Burada belirlenecek gösterim mekanizması, ağın performansını doğrudan etkileyecektir. Bu da kullanıcının yeteneğine bağlıdır.

Ağın eğitiminin ne zaman bitirilmesi gerektiğine ilişkin belli bir yöntem yoktur. Ağın örnekler üzerindeki hatasının belirli bir değerin altına indirilmesi, eğitimin tamamlandığı anlamına gelmektedir. Burada optimum neticeler veren bir mekanizma henüz yoktur ve YSA'ları ile ilgili araştırmaların önemli bir kolunu oluşturmaktadır.

Ağın davranışları açıklanamamaktadır. Bu sorun YSA'larının en önemli sorunudur. YSA'ları bir probleme çözüm ürettiği zaman, bunun neden ve nasıl olduğuna ilişkin bir ipucu vermez. Bu durum ağa olan güveni azaltıcı bir unsurdur.

Geleneksel Algoritmalar	Yapay Sinir Ağları
➤ Çıktılar; koyulan kurallara, girişlerin uygulanması ile elde edilir.	➤ Öğrenme esnasında giriş çıkış bilgileri verilerek, kurallar koyulur.
➤ Hesaplama; merkezi, eş zamanlı ve ardışıldır.	➤ Hesaplama; toplu, eş zamansız ve öğrenmeden sonra paraleldir.
➤ Bellek paketlenmiş ve hazır bilgi depolanmıştır.	➤ Bellek ayrılmış, ve ağa yayılmıştır. Dahilidir.
➤ Hata toleransı yoktur.	➤ Hata toleransı vardır.
➤ Nisbeten hızlıdır.	➤ Yavaş ve donanıma bağımlıdır.
➤ Bilgiler ve algoritmalar kesindir.	➤ Deneyimden yararlanır.

Tablo 1.1 Geleneksel algoritmalar ile YSA'ları

1.4. YSA'larının Kullanıldığı Alanlar

Yapay sinir ağları başlıca; **Sınıflandırma, Modelleme ve Tahmin** uygulamaları olmak üzere, pek çok alanda kullanılmaktadır. Başarılı uygulamalar incelendiğinde, YSA'larının çok boyutlu, gürültülü, karmaşık, kesin olmayan, eksik, kusurlu, hata olasılığı yüksek sensör verilerinin olması ve problemi çözmek için matematiksel modelin ve algoritmaların bulunmadığı, sadece örneklerin var olduğu durumlarda yaygın olarak kullanıldıkları görülmektedir. Bu amaçla geliştirilmiş ağlar genellikle şu fonksiyonları gerçekleştirmektedirler;

- Muhtemel fonksiyon kestirimleri
- Sınıflandırma
- İlişkilendirme veya örüntü eşleştirme
- Zaman serileri analizleri
- Sinyal filtreleme
- Veri sıkıştırma
- Örüntü tanıma
- Doğrusal olmayan sinyal işleme
- Doğrusal olmayan sistem modelleme
- Optimizasyon
- Kontrol

YSA'ları pek çok sektörde değişik uygulama alanları bulmuştur. Bunlardan bazıları;

Bankacılık: Kredi uygulamaları geliştirilmesi, müşteri analizi ve kredi müracaat değerlendirilmesi, bütçe yatırım tahminleri vs.

Savunma: Silah yönlendirme, hedef seçme, radar, sensör sonar sistemleri, sinyal işleme, görüntü işleme vs.

Elektronik: Kod sırası öngörüsü, çip bozulma analizi, non-lineer modelleme vs.

Üretim: Üretim işlem kontrolü, ürün dizaynı, makina yıpranmalarının tespiti, dayanıklılık analizi, kalite kontrolü, iş çizelgeleri hazırlanması vs.

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

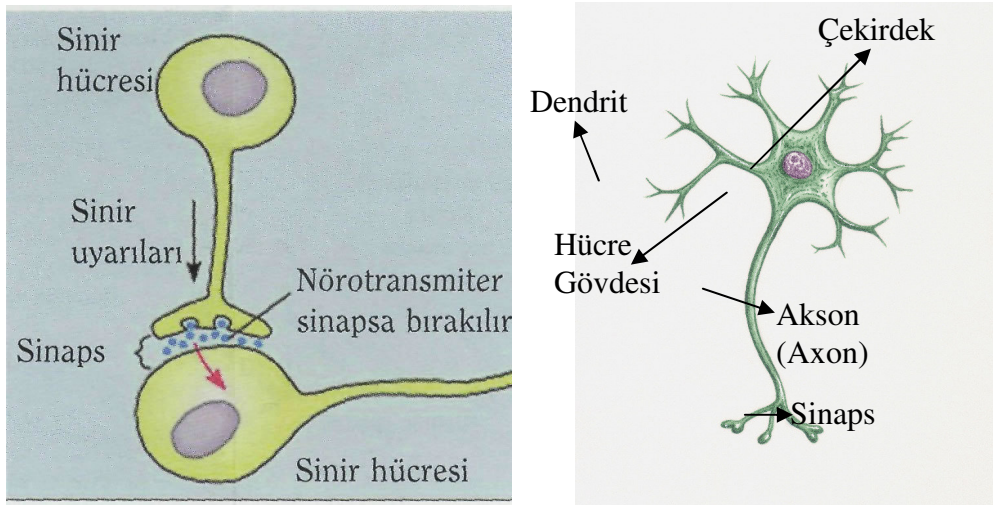
Telekomünikasyon: Görüntü ve data karşılaştırma, filtreleme, eko ve gürültü sönmülendirilmesi, ses ve görüntü işleme, trafik yoğunluğunun kontrolü ve anahtarlama vs.

Güvenlik: Parmak izi tanıma, kredi kartı hileleri saptama, retina tarama, yüz eşleştirme vs.

Görüldüğü gibi YSA'ları günlük hayatımızda farkında olmadığımız pek çok alanda kullanılmaktadır. Gün geçtikçe uygulama alanları genişlemekte ve gelişmektedir.

1.5. Beyin Sinir Hücresinin Yapısı

İnsanın hatırlama, düşünme, her harekette daha önceki deneyimlere başvurma yeteneğini, beyninin en temel parçası olan sinir hücreleri sağlar.



Şekil 1.5.1 Sinir Hücresi Yapısı

Sinir hücresi, sinir sisteminin temel fonksiyonel birimidir. Kısa uzantıları dendrit, uzun uzantıları akson olarak adlandırılan bu hücreler sinir sistemini oluşturur (Şekil 1.5.1).

Görevlerine Göre Sinir Hücresi Tipleri:

a- Duyu Nöronları: Duyu reseptöründen aldığı uyarıları merkezi sinir sistemine (beyin ve omurilik) taşır.

b- Ara Nöronlar: Bunlar, merkezi sinir sisteminde bulunurlar. Duyu nöronlarıyla gelen bilgileri değerlendirirler ve sonuçlarını motor nöronlara aktarırlar.

c- Motor Nöronlar: Bunlar, merkezi sinir sistemindeki ara nöronlardan aldıkları bilgiyi, kas ve salgı bezi gibi yapılara götürürler. Motor nöronların etkisiyle kas veya salgı bezleri faaliyete geçer.

Sinaps'lar sinir hücreleri arasındaki bağlantıları sağlar. Bunlar fiziksel olmayıp, bir hücreden diğerine elektrik sinyallerinin geçmesini sağlayan boşluklardır. Bu sinyaller hücre gövdesine gider. Hücre gövdesinde işleme tabi tutulup, kendi elektrik

sinyalini oluşturur. Bu sinyal akson aracılığıyla dendrit'lere, dendrit'lerde snaps lara gönderir. Sinyal buradan diğer hücrelere iletilir.

Beynimiz 10-12 Milyar arası sinir hücresi içerir. Bu hücreler bir araya gelerek sinir sistemini oluşturmaktadır. Yapay sinir ağları bu hücreler ve özelliklerinden faydalanılarak geliştirilmiştir.

1.6. Yapay Sinir Hücresi

Yapay sinir ağları, birbirine bağlı çok sayıda işlem elemanlarından oluşmuş, paralel işleyen yapılar olarak tanımlanabilir. Her işlem elemanının 5 temel elemanı vardır. Bunlar:

1.6.1. Girdiler

Bir yapay sinir hücresine dış dünyadan alınan bilgilerdir. Yapay sinir hücresine dış dünyadan olduğu gibi başka hücrelerden veya kendi kendisinden de bilgiler gelebilir.

1.6.2. Ağırlıklar

Yapay sinir hücresine gelen bilgilerin, önemini ve hücre üzerindeki etkisini belirleyen katsayılarıdır. Her bir giriş kendine ait bir ağırlığa sahiptir. Ağırlıklar, değişkenler veya sabit değerler olabilir. Bir ağırlığın değerinin büyük olması, o girişin yapay sinire güçlü bağlanması ya da önemli olmasıdır. Bir ağırlığın değerinin küçük olması, zayıf bağlanması ya da önemli olmamasıdır.

1.6.3. Toplama Fonksiyonu

Bu fonksiyon, bir hücreye gelen girdiyi hesaplar. En yaygın olarak kullanılan yöntem ağırlıklı toplamı bulmaktır. Yani her gelen girdi değeri, kendi ağırlığı ile çarpılarak toplanır. Formül olarak;

$$\text{ToplamGirdi} = \sum_i^n G_i A_i$$

şeklinde ifade edilebilir. Burada G girdileri, A ise ağırlıkları, n ise bir hücreye gelen toplam girdi sayısını göstermektedir. Bazı durumlarda toplama fonksiyonu bu kadar basit işlem olmayıp, daha karmaşık olabilir.

1.6.4. Etkinlik Fonksiyonu

Bu fonksiyon, hücreye gelen toplam girdiyi işleyerek, hücrenin bu girdiye karşılık üreteceği çıktıyı belirler. Bir etkinlik fonksiyonunun kullanım amacı, zaman söz konusu olduğunda, toplama fonksiyonunun çıkışının değişmesine izin vermektir.

1.6.5. Hücrenin Çıktısı

Etkinlik fonksiyonu tarafından belirlenen çıktı değeridir. Üretilen çıktı, dış dünyaya veya başka bir hücreye gönderilir. Bir yapay sinir hücresinden çıkan tek bir çıktı değeri vardır. Aynı değer birden fazla hücreye girdi olarak gitmektedir.

1.7 Yapay Sinir Ağının Yapısı

Yapay sinir hücreleri bir araya gelerek, yapay sinir ağını oluştururlar. Genel olarak hücreler 3 katman halinde ve her katman içinde paralel olarak bir araya gelerek, yapay sinir ağını oluştururlar. Sadece giriş ve çıkış katmanı olan ağlar, karmaşık işlevleri hesaplama yeteneğinden yoksundur. Bu nedenle, karmaşık hesaplamalar için oluşturulan ağlarda, en az bir ara katman olmalıdır.

YSA'ları, uygulanan ağ modeline göre değişik karakteristik özellikler göstermelerine karşın, temel birkaç ortak özelliğe sahiptirler.

Birinci özellik; YSA'larında sistemin paralelliği ve toplamsal işlevin yapısal olarak dağılımlılığıdır. YSA'ları bir çok hücrelerden meydana gelir ve bu hücreler eş zamanlı olarak çalışarak karmaşık işlevleri yerine getirir. Diğer bir deyişle; karmaşık işlevler, bir çok hücrenin eş zamanlı çalışması ile meydana getirilir. Süreç içerisinde bu hücrelerden her hangi biri işlevini yitirse dahi, sistem güven sınırları içerisinde çalışmasına devam edebilir.

İkinci özellik ise genelleme yeteneği, diğer bir deyişle ağ yapısının, eğitim esnasında kullanılan nümerik bilgilerden eşleştirmeyi betimleyen kaba özellikleri

çıkarması ve böylelikle eğitim sırasında kullanılmayan girdiler için de, anlamlı yanıtlar üretebilmesidir.

Üçüncü olarak; ağ fonksiyonları non-lineer olabilmektedir. Yapı üzerinde dağılmış belli tipteki non-lineer alt birimler özellikle, istenen eşleştirmenin denetim ya da tanımlama işlemlerinde olduğu gibi non-lineer olması durumunda, işlevin doğru biçimde yerine getirilebilmesini matematiksel olarak olası kılarlar.

Dördüncü özellik ise; sayısal ortamda tasarlanan YSA'larının, donanımsal gerçekleştirilebilirlikleridir. Bu özellik belki de YSA'larının, günlük hayatta daha da fazla yaşamımızın içine gireceğinin (girebileceğinin) göstergesidir.

1.7.1 Katmanlar

Giriş katmanından alınan veriler, giriş katmanı ve ara katman arasında bulunan bağlantı ağırlıkları ile çarpılıp ara katmana iletilir. Ara katmandaki sinirlere gelen girişler, toplanarak aynı şekilde ara katman ile çıkış katmanı arasındaki bağlantı ağırlıkları ile çarpılarak çıkış katmanına iletilir. Çıkış katmanı da, bu gelen girişleri toplayarak buna uygun çıkış üretir.

Giriş Katmanı : Bu katmandaki işlem elemanları, dış dünyadan gelen bilgileri (verileri) alarak ara katmana iletmekle sorumludurlar.

Ara Katmanlar : Giriş katmanından gelen bilgileri, işleyerek çıkış katmanına iletir. Bu bilgilerin işlenmesi ara katmanda gerçekleşir. Bir ağ için birden fazla ara katman olabilir.

Çıkış Katmanı : Bu katmandaki işlem elemanları, ara katmandan gelen bilgileri işleyerek çıktı üretirler. Üretilen çıktı dış dünyaya iletilir.

1.7.2 Katmanlar Arası Bağlantılar

Tam bağlantılı , İki katmandaki her sinir, ikinci katmandaki her sinire bağlıdır.

Kısmi bağlantılı, İki katmanın bir siniri, ikinci katmandaki her sinirle bağlantılı olmalıdır.

İleri Besleme bağı, İlk katmandaki sinirler, çıkışlarını ikinci kat sinirlerine gönderir. İkinci kat sinirlerinden geri bir giriş almazlar.

Çift Yönlü bağlar, ikinci katman sinirlerinin çıkışlarını, birinci katman sinirlerine taşıyan başka bir bağlantı kümesi vardır.

Hiyerarşik bağlar, hiyerarşik yapı varsa düşük katmanların bir sonraki seviyedeki katmanların sinirlerine iletilebilir.

Rezonans bağlar, çift yönlü katmanlar arası bağıdır. Kesin durum oluşuncaya kadar, bağlantı üzerinden defalarca mesajı göndermeye devam edebilir.

2. AĞ YAPILARI

Öğrenme Yöntemlerine göre ağ yapıları; Danışmanlı öğrenme yöntemleri ve Danışmansız öğrenme yöntemleri olmak üzere iki gruba ayrılır (Elmas, 2003,s: 69).

Danışmanlı öğrenme yöntemlerine göre **ağ yapıları**;

- Perceptron ağı (Basit algılayıcı ağı)
- Çok katmanlı Perceptron ağı
- Geri Yayılım ağı
- Yüksek düzeyli sinir ağı
- İşlevsel Bağ ağı

Danışmansız öğrenme yöntemlerine göre **ağ yapıları**;

- Hopfeild ağı
- Olasılık sinir ağları
- Uyarlanır rezonans ağı
- Özörgütlemeli harita ağı
- Boltzman makinesi ağı
- Hamming ağı
- Geri yayma içine özörgütlemeli harita ağı
- İki yönlü çağrışım belleği ağı
- Yığın ağı
- Karşı yayma ağı
- Öğrenme vektör nicelendirmesi
- Rekabetçi öğrenme ağları

Bu ağların bazılarının kısa tanımlarını yapalım.

2.1 Perceptron (Basit Algılayıcı) Ağı

Perceptron bir sinir hücresinin birden fazla girdiyi alarak bir çıktı üretmesi yöntemine dayanmaktadır. Ağın çıktısı 1 veya 0 şeklinde mantıksal değerlerdir. Girdiler işlem elemanına gönderilir. Her girdi değerine karşı çıktı değeri de ağa gönderilir. Daha sonra öğrenme kuralına göre ağın çıktı değeri hesaplanır. Eğer ağın çıktı değeri olması gerekenden farklı ise ağırlıklar ve eşik değeri değiştirilir. Değişikliğin nasıl yapılacağını öğrenme kuralı belirler. Eşik değeri, etkinlik fonksiyonunun konumunu belirlemek için kullanılır.

2.2 Çok Katmanlı Perceptron (Algılayıcı) Ağı

Girdi ve çıktılar arasında doğrusal ilişkilerin olmadığı durumlarda kullanılan yöntemdir. XOR problemi doğrusal ilişki olmayan duruma güzel bir örnektir. XOR işlemi için iki girdi olduğundan, dört farklı durum söz konusudur. Bu durumlar ve sonuçları aşağıdaki tabloda belirtilmiştir.

P	Q	Çıktı
1	1	0
1	0	1
0	1	1
0	0	0

Tablo – 2.2.1 XOR problemi

XOR problemini, çok katmanlı algılayıcı modeli ile çözmek mümkün olmuştur. Çok katmanlı algılayıcı ağında bilgiler, girdi katmanından ağa verilir. Ara katmanlardan geçerek, çıktı katmanına gelir. Buradan sonuç bilgisi alınır.

Danışmanlı öğrenme yöntemi olduğundan ağa, hem veriler hem de verilerden elde edilmesi istenilen çıktı bilgileri verilir. Ağ kendisine gönderilen örneklerden genellemeler yaparak problem uzayını temsil eden bir çözüm uzayı üretmektedir.

2.3 Hopfield Ağı

1982’de John Hopfield tarafından geliştirilen bir ağıdır. Hopfield ağında ikili değerler kullanılmaktadır. Hopfield ağında bir giriş katman, bir Hopfield katmanı ve bir çıkış katmanı olmak üzere 3 katman kullanılmasına rağmen farklı katmanlar yoktur. Bu ağda bir hücre diğer tüm hücelere bağlıdır. Ayrıca bağlantılar çift

yönlüdür. Her iki yönde akan veriye uygulanan ve bağlantı için hesaplanan bir ağırlık değeri vardır.

2.4 Hamming Ağı

Bu ağ Richard Lippman tarafından geliştirilmiştir. Hamming ağı, giriş vektörleri için en az ikili sayı hatasının temel sınıflandırılmasını yerine getirmektedir. Burada, Hamming tarafından hata, hata aralığı tanımlanmaktadır.

2.5 Karşı – Yayma Ağı

Robert Hecht-Nielsen bu ağ, danışmansız bir Kohonen katmanını, öğretilebilir bir çıktı katmanı ile birleştirmek için bir araç olarak geliştirmiştir.

Tek yönlü karşı yayma ağı, üç katmana sahiptir. Bir girdi tampon katmanı, bir Kohonen katmanı ve Delta kuralını kullanan bir çıktı katmanını içerir.

Rekabetçi Kohonen katmanı hiç bir denetim olmadan öğrendiği için, hangi sınıfa tepki gösterdiğini bilmez. Bu, Kohonen katmanındaki bir işleme elemanının, farklı sınıflara ait iki yada daha fazla eğitim girdisinin, sorumluluğunu almayı öğrenmesinin mümkün olduğu anlamına gelir. Böyle bir durumda ağın çıktısı, bu işleme elemanını aktive eden her girdi için belirsizdir. Bu problem oluşturabilir.

2.6 Geri Yayılım Ağı

Geri yayılım ağının, bir girdi katmanı, en az bir gizli katmanı ve bir çıktı katmanı vardır. Gizli katmanların sayısı için teorik olarak bir sınırlama yoktur. Yapılan çalışmalar sonucuna göre; herhangi bir derecede karmaşıklığa sahip problemleri çözmek için, en az dört katman (üç gizli katman ve bir çıktı katman) bulunması gerekmektedir.

Geri yayılım ağlarında katman sayısı ve her katmandaki hücre sayısı dikkatle seçilmelidir. Bu sayıların ne olacağı hakkında kesin bir yöntem yoktur. Bu konuda takip edilecek genel kurallar bulunmaktadır.

1. Kural : Girdi verisi ve istenilen çıktı arasındaki ilişkinin karmaşıklığı artınca, gizli katmanlardaki işleme elemanlarının sayısı da artmalıdır.

2. Kural : Eđer ele alınan süreç bir çok aşamalara ayrılabilir, fazla sayıda gizli katman kullanılmalıdır. Eđer süreç aşamalara ayrılamıyorsa, çok fazla sayıda gizli katman kullanılmamalıdır. Kullanılması durumunda ağda yalnızca ezberleme ortaya çıkar ve yanlış genel sonuçlara yol açar.

3. Kural : Ağda kullanılan eğitim verisinin miktarı, gizli katmanlardaki işleme elemanlarının sayısı için, üst bir sınır oluşturmaktadır. Gizli katmanların, az miktarda işleme elemanına sahip olması önemli bir faktördür.

Bir ağ oluşturmak için yukarıda belirtilen kurallar uygulandıktan hemen sonra, öğretim süreci başlar. Bu süreçte eğitim verisi, ağın giriş katmanına uygulanır ve istenen çıktılar çıkış katmanında karşılaştırılır. Öğrenme süreci esnasında, ağ boyunca bir ileri tarama yapılır ve her bir hücrenin çıktısı katman katman hesaplanır. En son katmanın çıktısı ve istenilen çıktı arasındaki fark, genellikle aktarım işlevinin türevi tarafından değiştirilen bir önceki katmana geri yayılır ve bağlantı bağıl değerleri Delta Kuralını kullanacak şekilde ayarlanır.

Geri yayılım ağı için, öğrenme kurallarının birçok çeşidi vardır. Farklı hata işlevleri, aktarım işlevleri ve hatta aktarım işlevinin türevinin değiştirme yöntemi de kullanılabilir.

2.7 Delta Bar Delta

Delta bar delta, Robert Jacobs tarafından, standart ileri beslemeli, geri yayılım ağlarının öğrenme oranını iyileştirmek amacıyla geliştirilmiştir. Delta bar delta ağı, bir geri yayılım ağı olarak aynı mimariden yararlanmaktadır. Delta bar delta ağının farklılığı, yalnızca kendine has algoritmik öğrenme metodudur. Delta bar delta, her bir bağıl değer kendine has kendini uyarlayabilen katsayıya sahip olduğu bir öğrenme metodu kullanmaktadır. Geri yayılım mimarisinin momentum faktörünü kullanmaz.

Bu yöntem bir çok uygulamayı çözmekte başarılıdır ancak, bazı pratik problemlerde kullanılmak için çok yavaştır.

İleri beslemeli çağrışım gibi sebeplerin geriye kalan çalışmaları, normal geri yayılım mimarisine benzerdir. Delta bar delta yönteminde; eski hata değerlerinin,

gelecek hesaplanmış hata deęerlerini tahmin etmek için kullanılabileceęi demektir. Muhtemel hataları bilmek, sistemin baęıl deęerlerini ayarlarken zeka adımları atmasını saęlar. Bir aęın her bir baęlantı baęıl deęerinin kendi öğrenme oranı olmalıdır. Her bir baęlantıya bir öğrenme oranı atayarak ve bu öğrenme oranının zaman içinde sürekli deęişmesine izin vererek, yakınsamaya harcanan zamanı azaltmak için daha fazla özgürlük derecesi girilebilir.

Bu algoritmayı doğrudan uygulayan kurallar, ileri doğru düzdür ve uygulaması kolaydır. Her bir baęlantı baęıl deęerinin kendi öğrenme oranı vardır. Bu öğrenme oranları, standart geri yayılım ile birlikte bulunan mevcut hataya baęlı olarak deęişir. Doğrusal olarak artırma, öğrenme oranlarını çok büyük ve çok hızlı hale gelmesini önler.

Bir aęda, her bir baęlantı baęıl deęeri için farklı oranlar olmasına izin vererek, bir çok dik iniş aramasını yapmaya (negatif iniş yönde) gerek kalmaz. Bunun yerine, baęlantı baęıl deęerleri, baęıl deęerin kendisi bakımından hatanın kısmi türevi bazında güncelleştirilir. Ayrıca, baęıl deęer deęişiklikleri, bölgesellik sınırlamasını karşılar. Yani, bu deęişiklikler yalnızca baęlı oldukları işleme elemanlarından gelecek bilgiye ihtiyaç duyarlar.

3. YAPAY SINİR AĞLARINDA ÖĞRENME

Yapay sinir ağlarında bilgi, ağdaki bağlantıların ağırlıklarında depolanır. Bir ağ da öğrenme, istenen bir işlevi yerine getirecek şekilde ağırlıkların ayarlanması sürecidir. Yapay sinir ağlarında öğrenme, sinirler arasındaki ağırlıkların değiştirilmesi ile gerçekleştirilmektedir.

Temelde öğrenme yöntemleri danışmanlı ve danışmansız olmak üzere iki gruba ayrılmıştır. Bunlar özet olarak (Elmas, 2003,s: 95);

Danışmanlı Öğrenme Yöntemleri,

- Perceptron,
- Delta öğrenme,
- Geri yayımlımlı,
- Eğimli iniş.

Danışmasız Öğrenme Yöntemleri,

- Rekabete dayalı öğrenme,
- Hebbian öğrenme,
- Kendi kendine öğrenme.

3.1 Danışmalı Öğrenme

Danışmanlı öğrenmede, yapay sinir ağı kullanılmadan önce eğitilmelidir. Eğitim işlemi, sinir ağına giriş ve çıkış bilgileri sunmaktan oluşur. Yani, her bir giriş kümesi için uygun çıkış kümesi ağına sağlanmalıdır. Yapay Sinir ağlarında gerçek çıkış değeri, istenen çıkış değeri ile kıyaslanır. Öğrenme yöntemi, bütün işleme elemanlarının anlık hatalarını en aza indirmeye çalışır. Hata azaltma işlemi, kabul edilebilir doğruluğa ulaşana kadar ağırlıklar devamlı olarak derlenir.

Sinir ağı, belirli bir sıralamadaki girişler için istenen istatistiksel doğruluğu elde ettiği zaman eğitim işlemi tamamlanmış olur. Öğrenme aşaması tamamlandıktan sonra ağ kullanılmaya başlandığında, bulunan ağırlıkların değeri sabit olarak alınır ve bir daha değiştirilmez.

Giriş ve çıkış bilgilerinin nasıl sunulacağı veya nasıl kodlanacağı, bir ağın başarılı bir şekilde yönlendirmek için önemlidir. Yapay sinir ağları sadece sayısal giriş bilgileri ile çalıştığından, ham bilgiler ölçeklendirilmelidir.

Yapay sinir ağlarındaki öğrenme oranı denetlenebilir bir kaç etkene bağlıdır. Daha düşük bir öğrenme oranı, yeterli derecede eğitilmemiş bir sistem üretmek için, çok daha fazla zaman harcanacağı anlamına gelmektedir.

Bu etkenler, bir ağın eğitilmesinde ne kadar süre alacağını belirlemede önemli rol oynar. Bu etkenlerin herhangi biri değiştirilirse, eğitim süresi artabilir ve büyük hatalara sebep olabilir.

Öğrenme yöntemlerinde genellikle öğrenme oranı veya öğrenme sabiti kullanılmaktadır. Bu oran sıfır ile bir arasında bir değerdir. Küçük öğrenme oranı en iyi sonuca ulaşma şansını yükseltmektedir.

3.2 Danışmansız Öğrenme

Danışmansız öğrenmede sistemin doğru çıkış hakkında bilgisi yoktur ve girişlere göre kendi kendisini örnekler. Danışmansız olarak eğitilebilen ağlar istenen ya da hedef çıkış olmadan giriş bilgilerinin özelliklerine göre ağırlık değerlerini ayarlar.

Danışmansız öğrenmede ağ istenen dış verilerle değil, girilen bilgilerle çalışır. Bu tür öğrenmede gizli sınırlar dışarıdan yardım almaksızın kendilerini örgütlemek için bir yol bulmalıdırlar. Bu yaklaşımda, verilen giriş vektörleri, önceden bilinebilen performansı ölçebilecek ağ için, hiçbir çıkış örneği sağlamaz. Yani ağ yaparak öğrenmektedir.

Danışmansız öğrenme için belli başlı örneklerden birisi Kohonen ağıdır. Kohonen ağında, giriş katmanına ek olarak, birbiriyle topolojik olarak ilişkili sınırlardan oluşan tek bir çıkış katmanı vardır. Her bir giriş, çıkış katmanındaki her bir sinire bağlıdır. Ağ rastgele ağırlıklarla çalışmaya başlar. Herhangi bir giriş uygulandığında, giriş vektörüne Öklid uzaklığı en az olan sinir seçilir ve bu sinire gelen bağlantı giriş ağırlıkları, giriş vektörüne yaklaşacak şekilde yenilenir. Bu kazanan sinirle birlikte, onun topolojik komşuluğunda bulunan belli sayıda sinire gelen ağırlıklar da benzeri şekilde değiştirilir.

Danışmansız öğrenme gelecek için büyük ümitler vaat etmektedir. Danışmansız öğrenme ile gelecekte bilgisayarların kendi kendilerine gerçek robotik hareketleri öğrenebilecekleri öne sürülmektedir.

3.3 Karma Öğrenme Kuralı

Karma öğrenme kuralında değişik uygulamalarda kullanılacak iki çeşit öğrenme örneği vardır. Bunlardan birincisi çevrim dışı öğrenme ya da öbek öğrenme olarak adlandırılır. Çevrim dışı öğrenmede değişkeni güncelleştirme hareketi sadece bütün eğitim veri kümesi sunulduktan sonra yerine getirilir. Başka bir deyişle her eğitim çağından sonra yerine getirilir. İkincisi ise çevrim içi öğrenme ya da örnekle öğrenme olup, değişkenler her bir giriş, çıkış ve veri parçalarının sunulmasından sonra derhal güncelleştirilebilir.

Her iki öğrenme tarzını da birleştirmek ve değişkenleri belli bir eğitim veri girişi yapıldıktan sonra güncelleştirmek mümkündür.

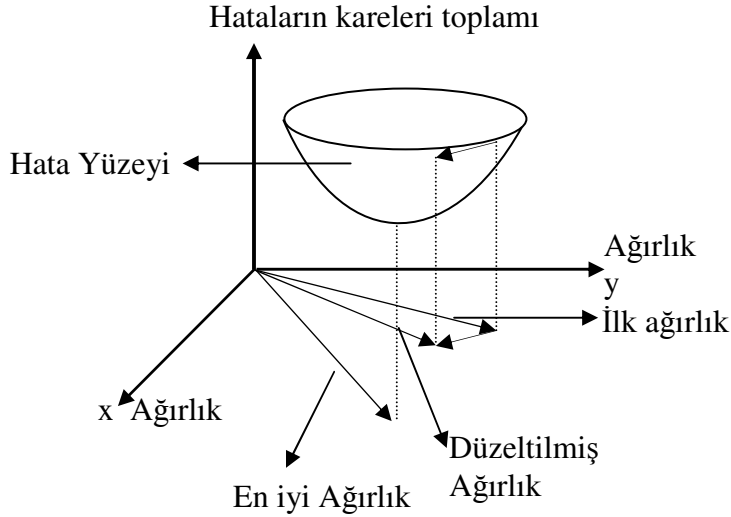
3.4 Danışmanlı Öğrenme Yöntemlerinden Delta Öğrenme Kuralı

Yapay sinir ağlarında en yaygın kullanılan öğrenme kuralı Widrow Hoff Kuralı veya en küçük kareler olarak ta bilinen delta öğrenme kuralıdır.

Eğitim işlemi sırasında giriş kümesi ağa uygulanır ve çıkış kümesi oluşturulur. Daha sonra gerçek çıkış değeri ile istenilen çıkış değeri karşılaştırılarak hata değeri hesaplanır. Eğer hata değeri sıfır ise ağdaki ağırlıklarda değişiklik yapılmadan eğitim işlemi sonlandırılır. Hata değeri sıfırdan büyükse, hata kabul edilebilecek bir değere düşünceye kadar ağırlıklarda değiştirme işlemi devam ettirilir.

Gizli katmanı olan bir ağda birden fazla iyi nokta olacağından en iyi noktayı bulmak zordur. Bu durumda en iyi değer elde edilinceye kadar eğilimli iniş yöntemi ile ağırlıklar değiştirilir.

Eğilimli iniş yönteminde, ağırlık çıktığı değerinin her bir ağırlık değerine göre kısmi türevi alınır. Bu elde edilen değer bir önceki değerden küçük olacaktır (Şekil 3.4.1).



Şekil 3.4.1 Hata değişimi, ağırlık vektörü ve eğimli inişin geometrik yorumu

n adet giriş, m adet çıkış hücresi olan tek katmanlı bir ağda her bir i-inci hücrenin çıkışı şu şekilde ifade edilebilir;

$$y_i = (w_i, x) = w_{i1}x_1 + \dots + w_{in}x_n, \quad i = 1..m$$

Eğitim kümesi $\{(x^1, y^1), \dots, (x^K, y^K)\}$ şeklinde ifade edilir. K eğitim çifti sayısını göstermektedir. Tek katmanlı ileri beslemeli bir ağda toplam ağ hatası

$$E = \frac{1}{2} \sum_{k=1}^K \sum_{i=1}^m (v_i^k - z_i^k)^2$$

olarak tanımlanır. Burada i çıkış hücresinin sırasını, k eğitim çifti sırasını, v_i^k i-inci çıkış hücresinin k-inci istenilen eğitim çıkışını gösterir. z_i^k ise i-inci çıkış değerinin k-inci eğitim çıkışına göre hesaplanan değerini gösterir.

E sistemin toplam hata değerini gösterir. Burada ağırlıkların değişimi için eğilimli iniş yöntemi kullanıldığından, her bir ağırlık sistem hatasının kendisine göre kısmi türevi alınarak güncelleştirilir. Bu;

$$w_{ij}^{(k)} = w_{ij}^{(k-1)} + \eta \frac{\partial E_k}{\partial w_{ij}}$$

şeklinde gösterilir. Burada η öğrenme sabiti olup $\eta > 0$ olarak alınır. Ağırlık değerine göre kısmi türev şu şekilde ifade edilir;

$$\frac{\partial E_k}{\partial w_{ij}^{(k)}} = \frac{\partial E_k}{\partial y_i} \frac{\partial y_i}{\partial w_{ij}^{(k-1)}} = -(v_i - z_i) x_i$$

burada $y_i = (w_i, x) = w_{i1}x_1 + \dots + w_{in}x_n$, $i = 1..m$ alınmıştır.

Böylece yeni ağırlık değeri

$$w_{ij}^{(k)} = w_{ij}^{(k-1)} + \eta (v_i - z_i) x_i \quad i=1, \dots, n$$

eşitliğiyle hesaplanır. Sistemin hata değeri δ_i^k ile gösterilirse

$$\delta_i^k = \frac{\partial E_k}{\partial y_i} = (v_i - z_i)$$

olur. Ağırlık değeri kısaca,

$$w_{ij}^{(k)} = w_{ij}^{(k-1)} + \eta \delta_i^k$$

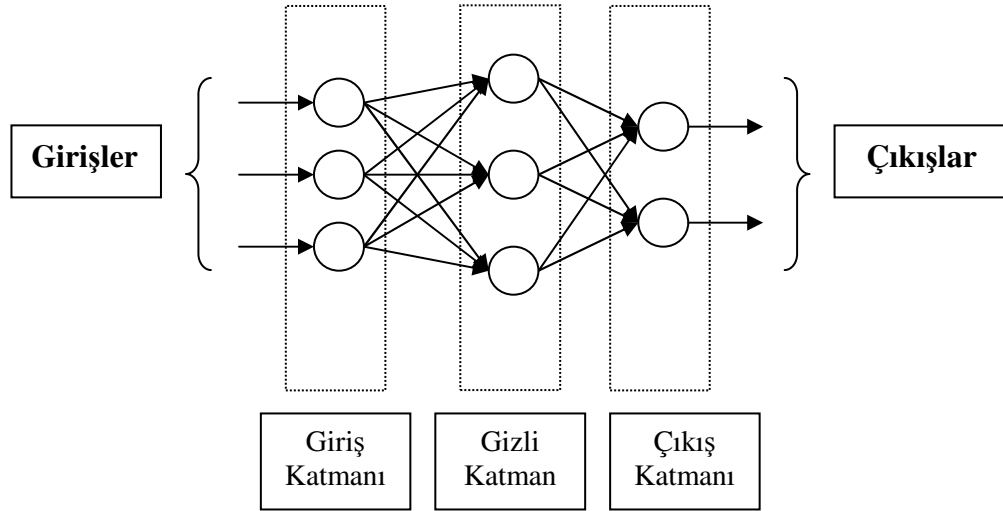
şeklinde yazılabilir.

3.5 Geri Yayımlı Öğrenme

Geri yayılım algoritması, sinir ağının eğitimci sınıfına giren genel bir algoritmadır. Giriş, çıkış ve en az bir katmandan oluşur. Girişlerle çıkışlar arasındaki hata sinyali bulunarak, ağırlıklar bu hata sinyaliyle güncellenmektedir. Hata yani $e(k)$, arzu edilen çıkış (gerçek çıkış - $y(k)$) ile sinir ağının çıkışı ($o(k)$) arasındaki farktır.

$$e(k) = y(k) - o(k)$$

Aşağıdaki şekilde birçok sinir hücresinin bir birine bağlandığı ileri yönlü çok katmanlı bir yapay sinir ağı görülmektedir. Giriş katmanı ile çıkış katmanı arasındaki katman veya katmanlar gizli katman olarak adlandırılır. Sinir ağlarında kaç tane gizli katman kullanılacağı ve her bir gizli katmanda kaç nöron olacağı bugüne kadar belirlenememiştir, probleme göre değişen bu nitelikler deneme-yanılma yoluyla bulunur.



Şekil 3.5.1 İleri beslemeli çok katmanlı sinir ağı.

Şekil 3.5.1'deki çok katmanlı ağı düşünürsek;

$k+1$. katmanda i . birime net giriş;

$$n^{k+1}(i) = \sum_{j=1}^{S_k} w^{k+1}(i, j) o^k(j) + b^{k+1}(i)$$

(1)

Birim i 'nin çıkışı şöyle olacaktır.

$$o^{k+1}(i) = f^{k+1}(n^{k+1}(i)) \quad (2)$$

M katmanlı bir ağ matris biçiminde ifade edilirse;

$$\begin{aligned} \underline{o}^0 &= \underline{x} \\ (3) \\ \underline{o}^{k+1} &= f^{k+1}(W^{k+1} \underline{o}^k + \underline{b}^{k+1}) \end{aligned}$$

$$k=0,1,\dots,M-1 \quad (4)$$

Ağın temel görevi giriş-çıkış çiftleri arasındaki ilişkiyi öğrenmektir.

$$\{(x_1, y_1), (x_2, y_2), \dots, (x_Q, y_Q)\}$$

Ağın performansı şöyledir;

$$E = \frac{1}{2} \sum_{q=1}^Q (y_q - o_q^M)^T (y_q - o_q^M) = \frac{1}{2} \sum_{q=1}^Q e_q^T e_q \quad (5)$$

Buradaki \underline{o}_q^M , q. giriş (\underline{x}_q) olduğunda ağın çıkışıdır, $\underline{e}_q = y_q - \underline{o}_q^M$ ise q.girişin hatasıdır. Standart geri yayılım algoritması için yaklaşık adım düşümü algoritmasını kullanır. Performans indeksiyle şöyle yaklaşılır;

$$E = \frac{1}{2} e_q^T e_q \quad (6)$$

burada toplam karelerin toplamı tek giriş-çıkış çifti için karesel hatayla yer değiştirilir. Yaklaşık adım (eğim) düşümü algoritması bundan sonra şöyle olur;

$$\Delta w^k(i, j) = -\alpha \frac{\partial E}{\partial w^k(i, j)} \quad (7)$$

$$\Delta b^k(i) = -\alpha \frac{\partial E}{\partial b^k(i)} \quad (8)$$

burada α öğrenme oranıdır, şöyle tanımlanır;

$$\delta^k(i) \equiv \frac{\partial E}{\partial n^k(i)} \quad (9)$$

performans indeksinin duyarlılığı k.katmanda i. birimin net girişinde değiştirilir. Şimdi (1),(6) ve (9) kullanılarak,

$$\frac{\partial E}{\partial w^k(i, j)} = \frac{\partial E}{\partial n^k(i)} \frac{\partial n^k(i)}{\partial w^k(i, j)} = \delta^k(i) o^{k-1}(j) \quad (10)$$

$$\frac{\partial E}{\partial b^k(i)} = \frac{\partial E}{\partial n^k(i)} \frac{\partial n^k(i)}{\partial b^k(i)} = \delta^k(i) \quad (11)$$

Hassaslık yeterliği aşağıdaki yinelemeli ilişkiyle de gösterilebilir.

$$\underline{\delta}^k = \dot{F}^k(\underline{n}^k) W^{k+1} \underline{\delta}^{k+1} \quad (12)$$

burada;

$$\dot{F}^k(\underline{n}^k) = \begin{bmatrix} \dot{f}^k(n^k(1)) & 0 & \dots & 0 \\ 0 & \dot{f}^k(n^k(2)) & \dots & 0 \\ \cdot & \cdot & \dots & \cdot \\ 0 & 0 & \dots & \dot{f}^k(n^k(Sk)) \end{bmatrix} \quad (13)$$

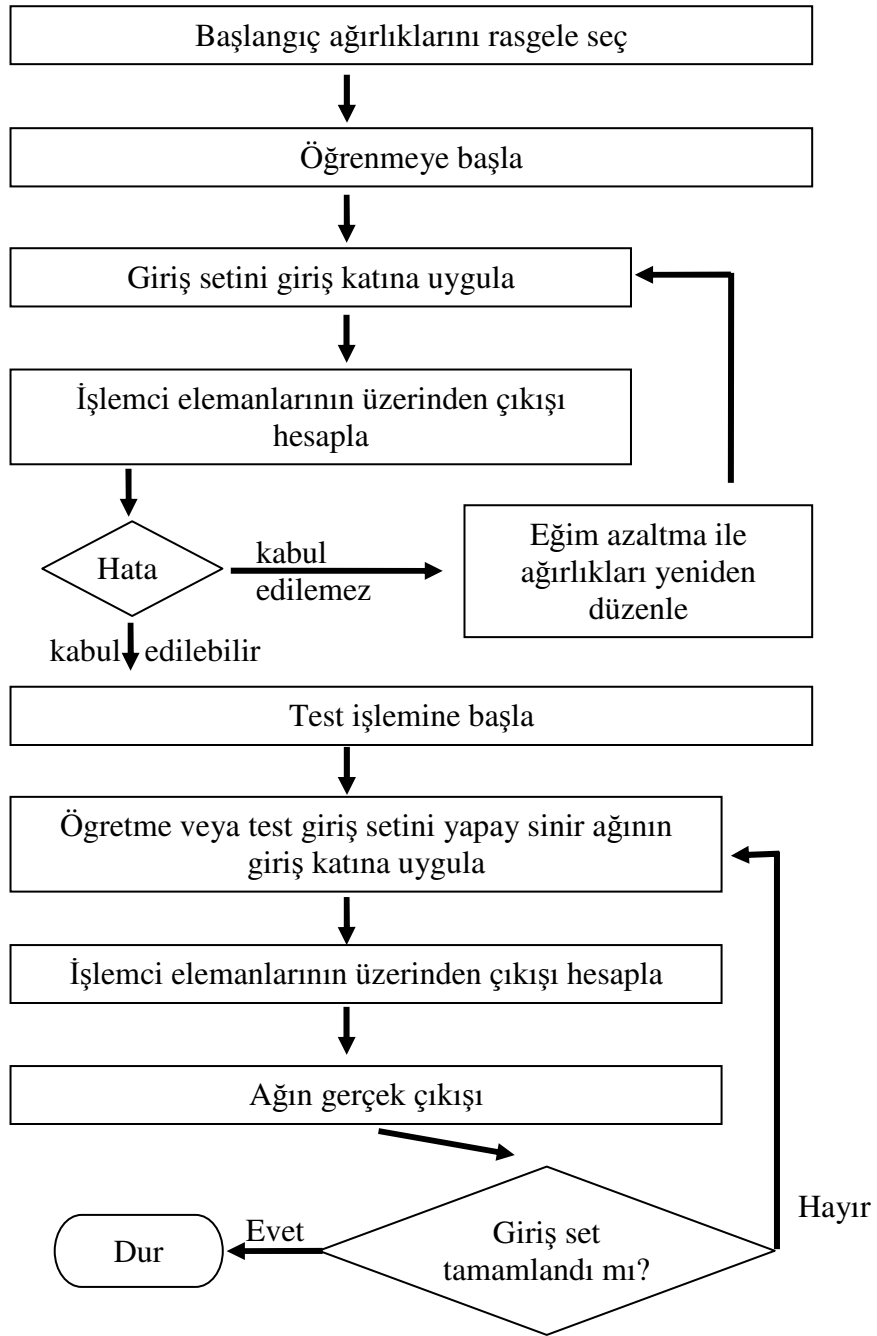
ve

$$f^k(n) = \frac{df^k(n)}{dn} \quad (14)$$

$$\underline{\delta}^M = -\dot{F}^M(\underline{n}^M)(\underline{y}_q - \underline{o}_q) \quad (15)$$

Tüm öğrenme algoritmalarında şu adımlar izlenir. İlk önce (3)-(4) kullanılarak giriş ileri yönde yayılır, sonra (15) ve (12) kullanılarak geri yayılır ve son olarak ağırlıklar ve denge (7), (8), (10) ve (11) kullanılarak güncellenir.

3.5.1 Geri Yayılm Algoritması



4. BANKACILIK ALANINDA YAPAY SİNİR AĞLARI

İkinci Dünya Savaşı öncesinde bir borçlunun kredibilitesinin değerlendirilmesi 'işletme veya özel kredi alanında' çok subjektif ve kalitatif bir sahaydı. Çoğu banka, müşterilerle bizzat görüşüp, borçlunun kredibilitesi hakkında subjektif kanaatlerine ve uzun yıllara dayalı tecrübelerine göre değerlendirmelerini yapan bir dizi kredi uzmanına başvuruyordu. İkinci Dünya Savaşı sırasında ve sonrasında bu uzmanların bir çoğuna ulaşılabilmesi söz konusu olmadığından, -özellikle İngiltere ve ABD'de- kredibilite değerlendirmesinde kantitatif metotlar kullanılmaya başlandı. Finans sektörünün diğer alanlarında olduğu gibi, risk yönetimi bilimi de tedricen kalitatiften kantitatife doğru bir seyir izlemiştir.

Bu arada borçluyu değerlendiren bir çok metot grubu mevcuttur. Bunlardan en önemli ve tanınan üç kategori:

1. Klasik skorlama modelleri
2. Merton modelleri
3. Yapay sinir ağları'dır.

Türkiye'de akademik anlamda,

- Bankalarda mali başarısızlığın öngörülmesi lojistik regresyon ve yapay sinir ağı karşılaştırması,
- Türk bankacılık sektöründe finansal güç derecelerinin Tahmininde yapay sinir ağları ve çok değişkenli istatistiksel Analiz tekniklerinin performanslarının karşılaştırılması

alanlarında yapılan çalışmalarda Yapay Sinir Ağları incelenmiş ve bu sistem başarılı olarak bulunmuştur.

4.1 Kredi Kartı Uygulaması

Kredi kartı müşterilerinin ödeme alışkanlıklarının düzenli olup olmayacağı bilgilerini YSA'ları ile çözümlenip çözülemeyeceği araştırıldı. Bunun için danışmalı öğrenme yöntemi olan İleri Beslemeli Geri yayımlı Öğrenme yöntemi kullanıldı.

Bir bankanın müşterilerine ait Cinsiyet, Medeni durum, Gelir aralığı durumu, Meslek grup bilgileri ve bu müşterilerin düzenli ödeme yapıp yapmadıkları bilgileri alındı.

Programın çalışması için 3 dosya gerekmektedir. Birinci dosyada mevcut olan yani bilinen veriler, sisteme bu bilgiler girdi olarak alınmaktadır. İkinci dosya da ise müşterilerin gösterdikleri eğilimler yer almaktadır. Sonuç dosyası olarak, bu dosyada “Krediyi düzenli öder (1)” ve “Krediyi düzenli ödemez (0)” bilgileri yer almaktadır. Üçüncü dosya ise kontrol dosyasıdır. Krediyi düzenli ödeyip ödemeyeceği öğrenilmek istenen müşteri kriterlerinin yer aldığı dosyadır. Ağ eğitilmişse bu dosyadaki verileri yorumlayarak program bize sonuçları verecektir.

Yapay sinir ağlarının eğitilmesi sözel verilerin sayısal değerlerle ifade edilmesi ile mümkündür. Yani mevcut verilerin, ağa rakamsal olarak tanıtılması gerekmektedir. Bu nedenle değerlere dönüşüm yapmak gerekmektedir. Aşağıda örnek uygulama için yapılan dönüşüm tabloları yer almaktadır.

Giriş Dosyası Bilgileri (KriterB.txt);

Değer	Anlamı
1	Erkek
0	Kadın

Tablo 4.1 Cinsiyet Referans Aralığı

Tablo 4.1 de görüldüğü şekilde Erkek olan müşteriler “1”, Kadın olan müşteriler “0” olarak değiştirildi.

Değer	Anlamı
1	Evli
2	Bekar
3	Dul/Boşanmış

Tablo 4.2 Medeni Durum Referans Aralığı

Tablo 4.2 de görüldüğü şekilde Evli olan müşteriler “1”, Bekar olan müşteriler “2”, Dul veya Boşanmış müşteriler için “3” olarak değiştirildi.

Değer	Anlamı
1	0-500
2	501-1,000
3	1,001-1,500
4	1,501-2,000
5	2,001-2,500
6	2,501-3,000
7	3,001-4,000
8	4,001-5,000
9	5,001-7,500
10	7,501-10,000
11	10,001-15,000
12	15,000 üzeri

Tablo 4.3 Gelir Referans Aralığı

Tablo 4.3 de görüldüğü şekilde müşteriler maaş durumları hangi değer aralığında ise ona göre “1” den “12” ye kadar sınıflandırıldı.

Değer	Anlamı
1	Girişimci (Doktor/Dış Hekimi/Eczacı/Danışman/Müşavir)
2	Finans/Sigorta/Bilişim
3	Eğitimci (Öğretim Görevlisi/Öğretmen)
4	Esnaf/Tüccar/Muhasebeci/Mali Müşavir/Mühendis/Mimar
5	Tarım (Çiftçi/Balıkçı/Ormancı)/ Lojistik
6	Yapı, inşaat çalışanı (Boyacı/Tesisatçı/Tamirci)
7	Turizm (Rehber/Tur Operatörü, Şoför)
8	Satış - Pazarlama Temsilcisi, sekreter
9	Diğer Meslekler

Tablo 4.4 Meslek Referans Aralığı

Tablo 4.4 de görüldüğü şekilde müşteriler hangi meslek grubuna giriyorsa ona göre “1” den “9” a kadar sınıflandırıldı.

Çıkış Dosyası bilgileri (CikisB.txt);

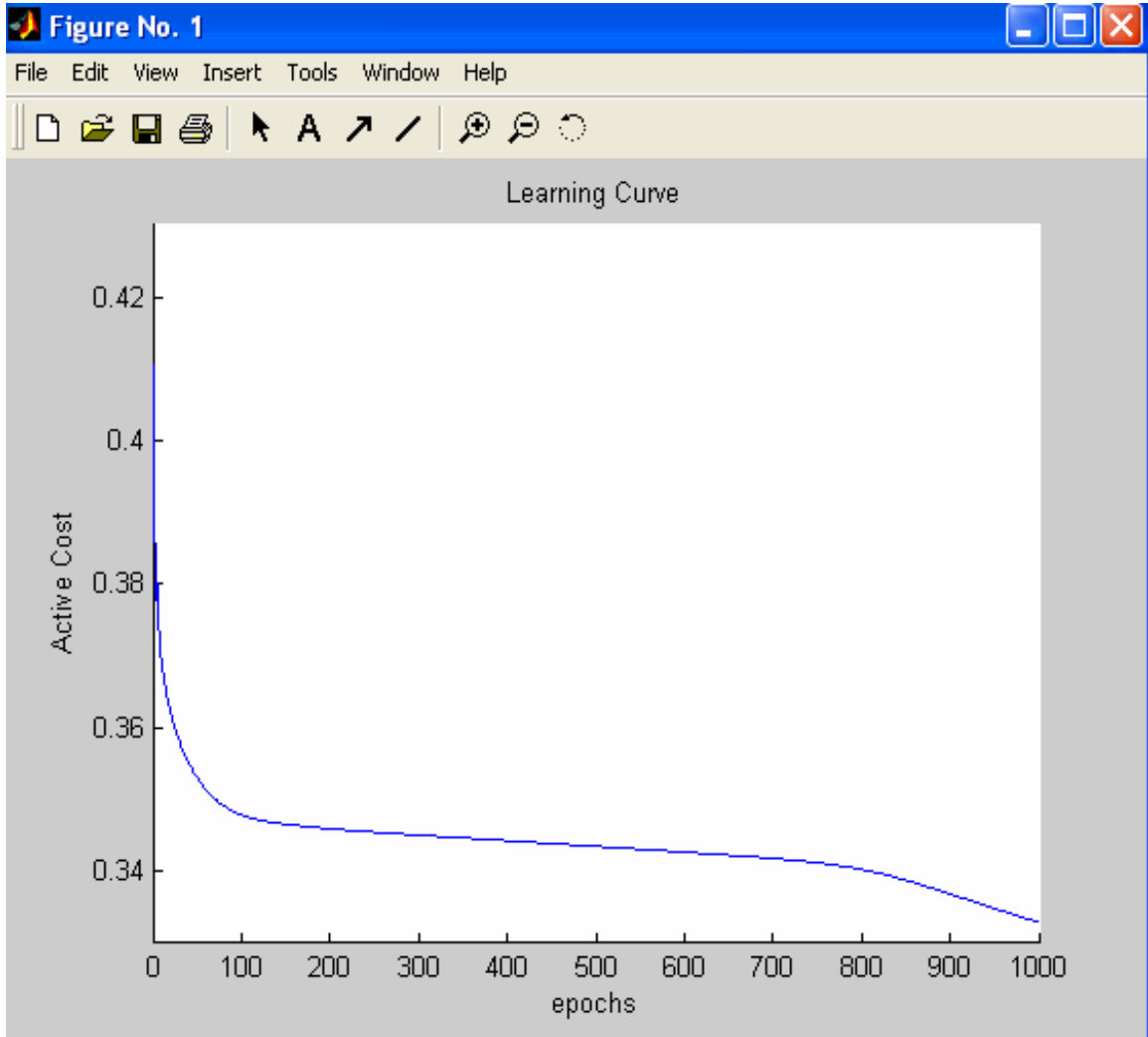
Değer	Anlamı
1	Düzenli Ödeme Yapıyor
0	Düzenli Ödeme Yapmıyor

Tablo 4.5 Ödeme Referans Aralığı

Tablo 4.5 de görüldüğü şekilde müşterilerin kredi kartı borçlarını düzenli ödemeleri durumu “1” , düzenli ödeme yapmamaları durumu “0” olarak değiştirildi.

4.2. Sonuç

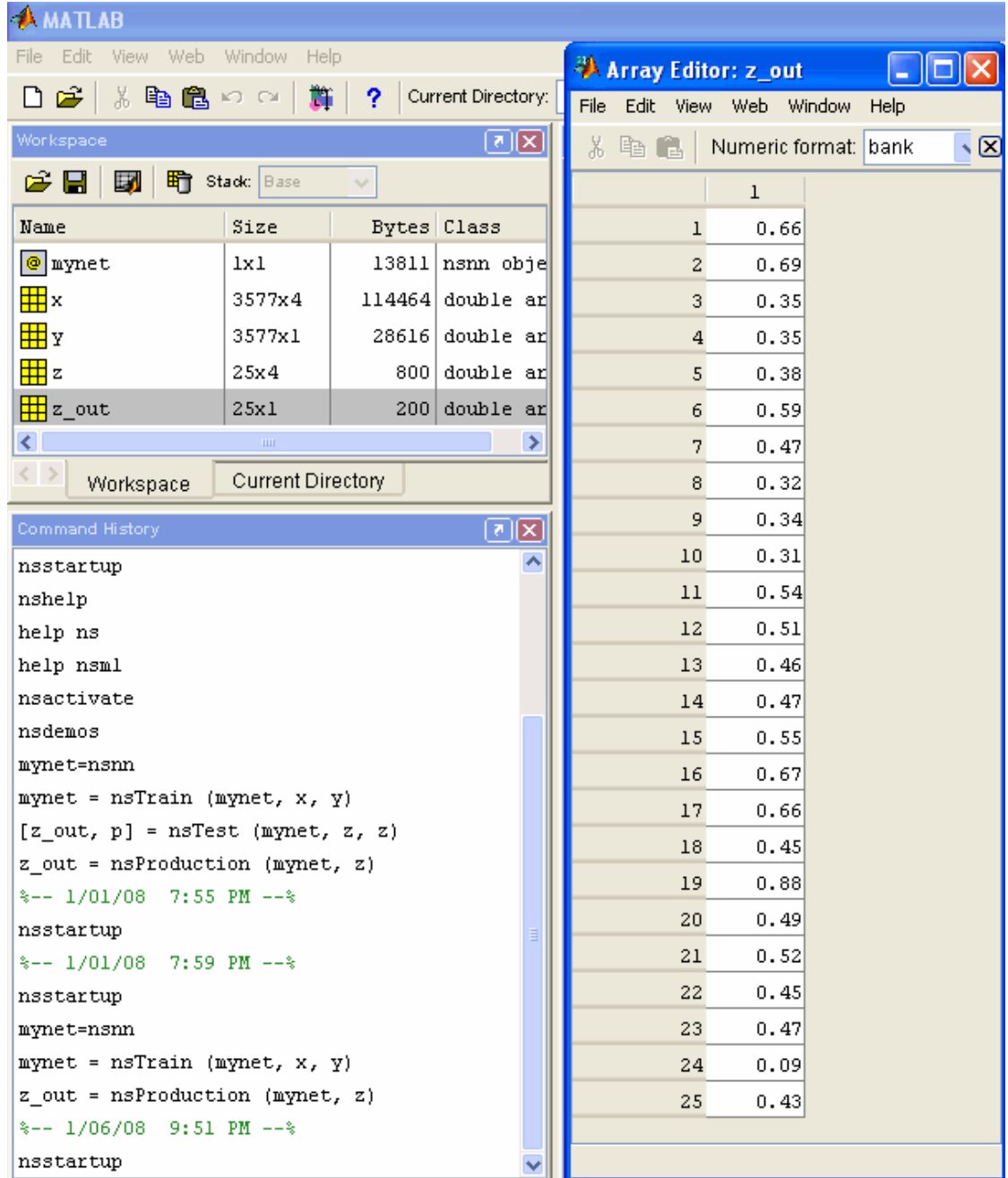
Matlab programı üzerine kurulan Neurosolitions programı yardımıyla mevcut veriler analiz edildi. Yapay Sinir Ağı programı çalıştırıldıktan sonra giriş, çıkış ve sinama dosyaları programa aktarıldı. 3577 satırlık kriter ve çıkış dosyaları yardımıyla eğitim denendi. Program eğitim sonrası hata grafiği oluşturdu(Şekil 4.2.1).



Şekil 4.2.1 Neurosolition Programına göre elde edilen Hata Grafiği

Hata grafiğinde adımlar ilerledikçe hata oranları azalır, ağ öğrenme işlemini başarı ile gerçekleştirmiştir. Bu şekilde anlaşılacağı gibi ağ öğrenme işlemini başarı ile gerçekleştirmiştir.

Öğrenme işlemi başarı ile gerçekleştiği için sınama dosyası ile program tekrar çalıştırıldı. Belirlenen kriterlere uygun kredi ödeme ihtimal yüzdeleri tablosunu YSA programı oluşturdu (Şekil 4.2.2).



The image displays the MATLAB environment. The main window shows the workspace with the following variables:

Name	Size	Bytes	Class
@mynet	1x1	13811	nsnn obje
x	3577x4	114464	double ar
y	3577x1	28616	double ar
z	25x4	800	double ar
z_out	25x1	200	double ar

The Command History window shows the following commands:

```

nsstartup
nshelp
help ns
help nsml
nsactivate
nsdemos
mynet=nsnn
mynet = nsTrain (mynet, x, y)
[z_out, p] = nsTest (mynet, z, z)
z_out = nsProduction (mynet, z)
%-- 1/01/08 7:55 PM --%
nsstartup
%-- 1/01/08 7:59 PM --%
nsstartup
mynet=nsnn
mynet = nsTrain (mynet, x, y)
z_out = nsProduction (mynet, z)
%-- 1/06/08 9:51 PM --%
nsstartup

```

The Array Editor window for 'z_out' shows the following data:

Index	Value
1	0.66
2	0.69
3	0.35
4	0.35
5	0.38
6	0.59
7	0.47
8	0.32
9	0.34
10	0.31
11	0.54
12	0.51
13	0.46
14	0.47
15	0.55
16	0.67
17	0.66
18	0.45
19	0.88
20	0.49
21	0.52
22	0.45
23	0.47
24	0.09
25	0.43

Şekil 4.2.2 Neurosolition Programına göre elde edilen Sonuç Tablosu

Bu çalışma ile kredi kartı tahsisi, bireysel kredi tahsisi gibi bankacılık konularında aktif olarak yapay sinir ağlarını kullanabileceği sonucuna varılır. Her kurumun müşterisinin kendine has özellikleri olabilir. Kurum kendi müşterilerinin ödeme alışkanlıklarını ve müşterilerin davranışlarını izlemesine yarayacak kişisel özelliklerini (yaş, cinsiyet, medeni hal, gelir durumu, çalıştığı yer, gayrimenkul bilgisi, taşıt bilgisi, vadeli/vadesiz hesap bilgisi v.s.) dikkate alarak YSA'larını kullanıp daha az riskli tahsis işlemi yapabileceklerdir.

KAYNAKLAR

BAZARAA, M.S. – SHERALI, H.D. – SHETTY, C.M. **Nonlinear Programming: Theory and Algorithms**, John Wiley and Sons Publishing, N.Y, 1993

BATTİTİ, R. **First and Second-Order Methods for Learning Between Steepest Descent, and Newton's Method**, Neural Computation, 1992

EFE, M. ÖNDER – KAYNAK, O. **Yapay Sinir Ağları ve Uygulamaları**, Boğaziçi Üniversitesi, İstanbul 2004

ELMAS, ÇETİN **Yapay Sinir Ağları Kuram, Mimari, Eğitim, Uygulama**, Seçkin Yayıncılık, Ankara 2003

ERDEM, SABRİ – ÇAKIR, ŞEN **Doğrusal Olmayan Kısıtlı Programlama ile Yapay Sinir Ağlarının Eğitilmesi**, Dokuz Eylül Üniversitesi ,2005

HAYKIN, S. **Neural Networks**, Macmillan Collage Printing Company, New Jersey 1994

KAASTRA, IEBELİNG – BOYD, MİLTON **Designing a neural network for forecasting financial and economic time series**, *Neurocomputing* (syf. 215-236), Elsevier, 1996

KARA, YAKUP - ACAR BOYACIOĞLU, MELEK “Türk bankacılık sektöründe finansal güç derecelerinin Tahmininde yapay sinir ağları ve çok değişkenli istatistiksel Analiz tekniklerinin performanslarının karşılaştırılması” , **10. Ulusal Finans Sempozyum Programı**, 2006

KESKİN BENLİ, YASEMİN, “Bankalarda mali başarısızlığın öngörülmesi lojistik regresyon ve yapay sinir ağı karşılaştırması”, **Gazi Üniversitesi Endüstriyel Sanatlar Eğitim Fakültesi Dergisi** Yıl: 2005 Sayı:16, s31-46

OKATAN, ALİ, **Yapay Sinir Ağları Ders Notları**, İstanbul, 2007

ÖZTEMEL, ERCAN **Yapay Sinir Ağları**, Papatya Yayıncılık, İstanbul, 2003

SAGIROGLU, ŞEREF **Mühendislikte Yapay Zeka Uygulamaları**, Ufuk Kitabevi, Ağustos 2003

TOKEL, ÖMER EMRE – KARATEPE, YALÇIN “Yapay Sinir Ağları kullanılarak kredi taleplerinin değerlendirilmesi”, **Active Dergisi** Yr:2005 No:45

TOKTAŞ, PERAL - BAŞAK DEMİRHAN, MELEK “Bankacılık Sektöründe Başarısızlık Tahminine Veri Madenciliği Yaklaşımı” **Yöneylem Araştırması /Endüstri Mühendisliği XXIV Ulusal Kongresi**, Gaziantep - Adana 2004,

www.bankaciyim.com (ERAY Yazar, Donnerstag, 2007)

www.backpropagation.netfirms.com

www.neurosolutions.com

www.yapay-zeka.org

www.finansbilim.com/ufs2006/Makaleler/TURKBANKACILIK.pdf

ÖZGEÇMİŞ

Doğum Tarihi : 02.02.1974

Doğum Yeri : Ladik / Samsun

Öğrenim Bilgileri :

Lisans : 1991-1995 Mimar Sinan Üniversitesi Matematik Bölümü

Yüksek Lisans : 2006-2008 Haliç Üniversitesi Yönetim Bilişim Sistemleri

Çalıştığı Kurumlar :

1997-2004 Pamukbank T.A.Ş.
Organizasyon Bölümü'nde Sistem Analist/Yetkili olarak görev yaptım.

2004- Türkiye Halk Bankası A.Ş.
Organizasyon Daire Başkanlığı'nda Kıdemli Uzman /Yönetici olarak görev yapmaktayım.