

İÇİNDEKİLER

BÖLÜM 1	1
GİRİŞ	1
1.1. YAPAY SİNİR AĞLARI	2
1.2. YAPAY SİNİR AĞLARININ TARİHÇESİ	3
1.3. YAPAY SİNİR AĞLARININ GENEL ÖZELLİKLERİ	7
1.4. YAPAY SİNİR AĞLARININ AVANTAJLARI VE DEZAVANTAJLARI	9
1.4.1. Avantajları	9
1.4.2. Dezavantajları	10
1.5. BİYOLOJİK NÖRON YAPISI	11
1.6. YAPAY SİNİR HÜCRESİ	16
1.7. BİR YAPAY SİNİR AĞININ TEMEL ELEMANLARI	18
1.7.1. Girişler	18
1.7.2. Ağırlıklar	19
1.7.3. Toplama İşlevi	19
1.7.4. Etkinlik İşlevi (Aktivasyon Fonksiyonu)	20
1.7.4.1. Doğrusal Fonksiyon	21
1.7.4.2. Basamak Fonksiyonu	22
1.7.4.3. Kutuplamalı Basamak Fonksiyonu	22
1.7.4.4. Parçalı Doğrusal Fonksiyon	23
1.7.4.5. Sigmoid Tipi Fonksiyon	24
1.7.4.6. Tanjant Hiperbolik Tipli Fonksiyon	24
1.7.4.7. Sinüs Tipli Fonksiyon	25
1.7.5. Çıkış İşlevi	25
1.7.6. Ölçekleme ve Sınırlama	26
1.8. YAPAY SİNİR HÜCRESİNİN ÇALIŞMA PRENSİBİ	26
1.9. YAPAY SİNİR AĞI YAPISI	27
1.9.1. Katmanlar	28
1.9.2. İletişim ve Bağlantı Çeşitleri	28
1.10. YAPAY SİNİR AĞLARI DONANIMI	30
1.11. YAPAY SİNİR AĞLARININ SINIFLANDIRILMASI-YAPILARINA GÖRE YAPAY SİNİR AĞLARI	31
1.11.1. İleri Beslemeli Ağlar	31
1.11.1.1. Çok Katmanlı Perseptron (MLP) Ağı	32
1.11.1.2. Öğrenme Vektör Nicelendirmesi (Learning Vector Quantization) Ağı	34
1.11.2. Geri Beslemeli Ağlar	35
1.11.2.1. Elman Ağı	36
1.11.2.2. Hopfield Ağı	38
1.11.2.3. Uyarlanırlar Rezonans (ART) Ağı	40
1.11.2.4. Som Ağı	41
1.12. YAPAY SİNİR AĞLARININ KRONOLOJİK SINIFLANDIRMASI	42
BÖLÜM 2	43
YAPAY SİNİR AĞLARINDA ÖĞRENME	43
2.1 TEMEL ÖĞRENME KURALLARI	44
2.1.1 Danışmanlı Öğrenme	44
2.1.1.1. Geri Yayılımlı Öğrenme Kuralı	46
2.1.1.1.1. Geri Yayılım Algoritması	46
2.1.1.1.2. Geri Yayılım Etkinlik İşlevleri	53
2.1.1.1.3. Öğrenme Oranı ve Ağ Üzerindeki Etkisi	53
2.1.1.1.4. Momentum Teriminin Ağ Üzerindeki Etkisi	53
2.1.1.1.5. Gizli Katmandaki Sinir Sayısının Ağ Üzerindeki Etkisi	53
2.1.1.1.6. Hata Farkı Değişkeninin Ağ Üzerindeki Etkisi	54

2.1.2. <i>Danışmansız Öğrenme</i>	54
2.1.3. <i>Takviyeli Öğrenme</i>	55
2.2. UYGULAMAYA GÖRE ÖĞRENME YAKLAŞIMLARI	56
2.3. ÖĞRENMEDE KULLANILAN DİĞER YAKLAŞIMLAR	57
2.4. YAPAY SİNİR AĞLARINI TEST ETME.....	57
2.5. YAPAY SİNİR AĞLARI TASARIMINDA DİKKAT EDİLMESİ GEREKEN HUSUSLAR.....	57
2.5.1. <i>Uygulama Başarısı</i>	58
2.5.2. <i>Uygulamada Karşılaşılan Güçlükler</i>	58
2.6. YAPAY SİNİR AĞI UYGULAMALARINDA TAKİP EDİLMESİ GEREKEN İŞLEMLER.....	59
BÖLÜM 3	63
YAPAY SİNİR AĞLARI UYGULAMALARI	63
3.1. ENDÜSTRİYEL UYGULAMALAR.....	63
3.2. MÜHENDİSLİK UYGULAMALARI	64
3.3. FİNANSAL UYGULAMALAR.....	65
3.4. ASKERİ UYGULAMALAR	65
3.5. SAĞLIK UYGULAMALARI.....	65
3.6. İLETİŞİM SANAYİ UYGULAMALARI.....	66
3.7. ULAŞTIRMA VE HAVACILIK SANAYİ UYGULAMALARI.....	66
BÖLÜM 4	67
TEMEL BİLEŞENLER ANALİZİ	67
BÖLÜM 5	70
GERİYAYILIMLI YAPAY SİNİR AĞI ÜZERİNE BİR UYGULAMA	70
5.1. UYGULAMA İÇİN YAPAY SİNİR AĞININ TASARIMI.....	77
5.2. SONUÇ.....	89
KAYNAKÇA	90
EKLER	92

ŞEKİLLER LİSTESİ

ŞEKİL 1.1 İNSAN BEYİNİ.....	12
ŞEKİL 1.2 BEYİN ÜZERİNDEKİ BEŞ TEMEL DUYU BÖLGESİ.....	12
ŞEKİL 1.3 BİYOLOJİK SİNİR SİSTEMİNİN BLOK GÖSTERİMİ.....	13
ŞEKİL 1.4 NÖRON.....	14
ŞEKİL 1.5 BİYOLOJİK VE YAPAY SİNİR AĞLARI.....	15
ŞEKİL 1.6 YAPAY SİNİR HÜCRESİNİN YAPISI.....	16
ŞEKİL 1.7 BİR YAPAY NÖRONUN TEMEL GRAFİK GÖSTERİMLERİ.....	17
ŞEKİL 1.8 DOĞRUSAL VEYA LİNEER FONKSİYON.....	21
ŞEKİL 1.9 BASAMAK FONKSİYONLARI.....	22
ŞEKİL 1.10 TEK KUTUPLAMALI BASAMAK FONKSİYONU.....	23
ŞEKİL 1.11 ÇİFT KUTUPLAMALI BASAMAK FONKSİYONU.....	23
ŞEKİL 1.12 PARÇALI DOĞRUSAL FONKSİYON.....	24
ŞEKİL 1.13 SİGMOİD TIPLI FONKSİYON.....	24
ŞEKİL 1.14 TANJANT HİPERBOLİK TIPLI FONKSİYON.....	25
ŞEKİL 1.15 SİNÜS TIPLI FONKSİYON.....	25
ŞEKİL 1.16 BİR YAPAY SİNİR AĞININ ÇALIŞMASI ÖRNEĞİ.....	26
ŞEKİL 1.17 ÇOK KATMANLI BİR YAPAY SİNİR AĞI.....	27
ŞEKİL 1.18 İLERİ BESLEMELİ AĞ.....	32
ŞEKİL 1.19 MLP YAPISI.....	33
ŞEKİL 1.20 LVQ AĞI.....	34
ŞEKİL 1.21 GERİ BESLEMELİ AĞ.....	35
ŞEKİL 1.22 ELMAN AĞ YAPISI.....	36
ŞEKİL 1.23 ELMAN AĞININ AYRINTILI GÖSTERİMİ.....	38
ŞEKİL 2.1 DANIŞMANLI ÖĞRENME.....	45
ŞEKİL 2.2 GERİ YAYILIM AĞI.....	47
ŞEKİL 2.3 BİR GERİ YAYILIM AĞI DÜĞÜMÜNDEKİ ÖĞRENME.....	48
ŞEKİL 2.4 BİR AĞDA KULLANILAN GERİYAYILIMLI ÖĞRENME ALGORİTMASI... ..	52
ŞEKİL 2.5 DANIŞMANSIZ ÖĞRENME.....	54
ŞEKİL 2.6 TAKVİYELİ ÖĞRENME YAPISI.....	55
ŞEKİL 2.7 TAKVİYELİ ÖĞRENME BLOK DİYAGRAMI.....	56
ŞEKİL 5.1 SOSYO-EKONOMİK GELİŞMİŞLİK ENDEKS TAHMİNİ İÇİN YAPAY SİNİR AĞI.....	78
ŞEKİL 5.2 GERÇEK ÇIKIŞLAR İLE YSA EĞİTİM ÇIKIŞLARININ KARŞILAŞTIRILMASI.....	84
ŞEKİL 5.3 GERÇEK ÇIKIŞLAR İLE YSA TEST ÇIKIŞLARININ KARŞILAŞTIRILMASI.....	85
ŞEKİL 5.4 GERÇEK ÇIKIŞLAR İLE YSA EĞİTİM ÇIKIŞLARININ KARŞILAŞTIRILMASI.....	86
ŞEKİL 5.5 GERÇEK ÇIKIŞLAR İLE YSA TEST ÇIKIŞLARININ KARŞILAŞTIRILMASI.....	86
ŞEKİL 5.6 GERÇEK ÇIKIŞLAR İLE YSA EĞİTİM ÇIKIŞLARININ KARŞILAŞTIRILMASI.....	87
ŞEKİL 5.7 GERÇEK ÇIKIŞLAR İLE YSA TEST ÇIKIŞLARININ KARŞILAŞTIRILMASI.....	88

TABLolar LİSTESİ

TABLO 1.1 SİNİR SİSTEMİ İLE YSA'NIN BENZERLİKLERİ.....	15
TABLO 1.2 İSTATİSTİKSEL YÖNTEMLERLE YAPAY SİNİR AĞLARININ BENZEŞİMLERİ.....	15
TABLO 1.3 TOPLAMA FONKSİYONU ÖRNEKLERİ	20
TABLO 2.1 YAPAY SİNİR AĞLARI VE UYGULAMA ALANLARI.....	60
TABLO 5.1 GİRİŞ DEĞİŞKENLERİ VE VARYANSLARI.....	74
TABLO 5.2 UYGULAMADA EĞİTİM İÇİN KULLANILAN GİRİŞ VERİLERİ.....	79
TABLO 5.3 UYGULAMADA TEST İÇİN KULLANILAN GİRİŞ VERİLERİ.....	81
TABLO 5.4 TRAİNGD ALGORİTMASI İLE ELDE EDİLEN MUTLAK HATALAR.....	84
TABLO 5.5 TRAİNGDM ALGORİTMASI İLE ELDE EDİLEN MUTLAK HATALAR	85
TABLO 5.6 TRAİNGDX ALGORİTMASI İLE ELDE EDİLEN MUTLAK HATALAR	87
TABLO 5.7 ORTALAMA MUTLAK HATA DEĞERLERİ.....	88
TABLO 5.8 HATA KARELER ORTALAMASI DEĞERLERİ.....	88

BÖLÜM 1

GİRİŞ

İnsanoğlu yaradılışından beri doğa ile iç içe yaşamış ve ondan esinlenerek karşılaştığı problemler için çözüm yöntemleri geliştirmiştir. Bu yöntemler bilgisayar kullanılarak yapılan sayısal hesaplamaların hızla gelişmesiyle çok ilerleme kaydetmiştir. Geliştirilen yöntemlerin bazıları canlı organizmayı örnek alarak yapılmıştır. Bu yolla geliştirilmiş olan yöntemlere örnek olarak Yapay Sinir Ağları (YSA) ve Genetik Algoritmalar (GA) gösterilebilir.¹

YSA kavramı beynin çalışma ilkelerinin sayısal bilgisayarlar üzerinde taklit edilmesi fikri olarak ortaya çıkmış ve ilk çalışmalar beyni oluşturan biyolojik hücrelerin, ya da literatürdeki ismi ile nöronların matematiksel olarak modellenmesi üzerinde yoğunlaşmıştır.²

Yapay sinir ağları, beynin fizyolojisinden yararlanılarak oluşturulan bilgi işleme modelleridir. Literatürde 100'den fazla yapay sinir ağı modeli vardır. Bazı bilim adamları, beynimizin güçlü düşünme, hatırlama ve problem çözme yeteneklerini bilgisayara aktarmaya çalışmışlardır. Bazıları ise, beynin fonksiyonlarını kısmen yerine getiren modelleri oluşturmaya çalışmışlardır.

YSA'ların öğrenme özelliği, araştırmacıların dikkatini çeken önemli özelliklerden birisidir. Çünkü herhangi bir olay hakkında girdi ve çıktılar arasındaki ilişkiyi, doğrusal olsun veya olmasın, elde bulunan mevcut örneklerden öğrenerek daha önce hiç görülmemiş olayları, önceki örneklerden çağrışım yaparak ilgili olaya çözümler üretebilme özelliği YSA'lardaki zeki davranışın da temelini teşkil eder.³

Yapay sinir ağları, bir sisteme ilişkin tek veya çoklu parametrelere bağlı olarak tanımlanan giriş verileri ile sistemin yine tek veya çoklu parametrelere bağlı olarak tanımlanabilen çıkışları arasında ilişki kurabilme yeteneğine sahiptir.

¹ Zekai Şen, *Yapay Sinir Ağları İlkeleri*, İstanbul: Su Vakfı Yayınları, 2004, s.7.

² M. Önder Efe ve Okyay Kaynak, *Yapay Sinir Ağları ve Uygulamaları*, İstanbul: Boğaziçi Üniversitesi Yayınları, 2000, s. 1.

³ Yapay Sinir Ağları ve Uygulamaları, 2000, http://batitrakya.dostweb.com/yapaysiniraglari/ek_dokuman.htm, (24.11.2005).

1.1. Yapay Sinir Ağları

Yapay sinir ağlarının herkes tarafından kabul edilen genel bir tanımı yoktur. Bazı bilim adamları tarafından yapılmış olan tanımlar aşağıdaki gibidir.

Haykin Sinir Ağlarına Detaylı Bakış (Neural Networks: A Comprehensive Foundation), isimli kitabında yapay sinir ağlarını “Bir sinir ağı, bilgiyi depolamak için doğal eğilimi olan basit birimlerden oluşan paralel dağıtılmış bir işlemcidir. O beyni, bilgi öğrenme işlemi yoluyla ağ tarafından elde edilir. Sinaptik ağırlıklar olarak bilinen nöronlar arası bağlantı kuvvetlerini, bilgiyi depolamak için kullanır gibi iki şekilde ifade eder.”

Zurada ise, Yapay Sinir Sistemlerine Giriş (Introduction to Artificial Neural Systems), kitabında Yapay sinir ağlarını “yapay sinir sistemleri veya sinir ağları deneysel bilgiyi alan, depolayan ve kullanan fiziksel hücreli sistemlerdir.” şeklinde tanımlamıştır.⁴

Yapay sinir ağları insan idrakinin ve biyolojik sinirlerin matematiksel modelinin geliştirilmesi olarak da tanımlanabilir.⁵

Basit tanımıyla yapay sinir ağları, birçok basit işlemci elemandan oluşan yapılardır. Bu elemanlar farklı formda ifade edilebilen nümerik verileri taşıyan “bağlantılar” veya “ağırlıklar” ile birbirine bağlıdır.⁶

Yapay sinir ağı şöyle de tanımlanabilir: Basit hesaplama elemanlarının yoğun bağlantılarından meydana gelmiş bir bilgi işleme elemanıdır.⁷

Yapay sinir ağları, insan beyninin özelliklerinden olan öğrenme yolu ile yeni bilgiler türetebilme, oluşturabilme ve keşfedebilme gibi yetenekleri herhangi bir yardım almadan otomatik olarak gerçekleştirmek amacı ile geliştirilen bilgisayar sistemleridir. Bu yetenekleri geleneksel programlama yöntemleri ile gerçekleştirmek oldukça zordur veya mümkün değildir. O nedenle, yapay sinir ağlarının, programlanması çok zor yada

⁴ Şeref Sağırođlu, Erkan Beşdok ve Mehmet Erler, *Mühendislikte Yapay Zeka Uygulamaları-I, Yapay Sinir Ağları*, Kayseri: Ufuk Kitap Kırtasiye, 2003, s. 25.

⁵ Laurene Fausett, *Fundamentals of Neural Networks*, New Jersey: Prentice-Hall, 1994, s. 3.

⁶ Sağırođlu ve diđerleri, *Age*, s.24.

⁷ Patrick K. Simpson, *Neural Networks Theory, Technology and Applications*, New York: The Institute of Electrical and Electronics Engineers Press, 1990, s. 1.

mümkün olmayan olaylar için geliştirilmiş adaptif bilgi işleme ile ilgilenen bir bilgisayar bilim dalı olduğu söylenebilir.

Yapay sinir ağları aynı zamanda, “bağlantılı ağlar (connectionist networks)”, “paralel dağıtılmış ağlar (parallel distributed networks)”, neuroformik sistemler (neuroformic systems)” olarak da adlandırılmaktadır. Yapay sinir ağları bilgisayar bilimine de bazı yenilikler getirmiştir. Algoritmik olmayan, adaptif, paralel programlama, dağıtılmış programlama vb. gibi tekniklerin gelişmesine katkıda bulunmuşlardır. Özellikle olaylar hakkında bilgilerin olmadığı fakat örneklerin bulunduğu durumlarda çok etkin olarak kullanılabilen bir karar verme aracı ve hesaplama yöntemi olarak görülebilir.⁸

Yapay sinir ağlarındaki gelişmelerin temel kaynağı, beynimizin rutin olarak gerçekleştirdiği karmaşık hesaplamaları yapay, belki zeki davranış sergileyen sistemlerin yapılabileceği ümididir.⁹

1940’lı yıllarda biyolojik sinir hücrelerinin yapısından esinlenerek tasarlanan yapay sinir hücre modeli VE, VEYA ve DEĞİL gibi mantıksal işlemlerin sayısal olarak modellenebileceğini göstermiştir. Böylece biyolojik sinir sisteminin incelenmesi ve onlara benzer şekilde çalışan yapay sinir ağları modellerinin geliştirilmesi farklı dallardaki araştırmacıların ortak çalışma konusu haline gelmiştir.

Bilgi işleme süreçleri olarak nitelendirebileceğimiz yapay sinir ağları verilen girdilere karşılık çıktılar üreten bir kara kutu olarak nitelendirilebilir. Yapay sinir ağları alışılmış bilgi işleme yöntemlerinden farklılık göstermektedir. Hatta sahip olduğu özelliklerden dolayı birçok yönetime göre daha sağlıklı sonuçlar üretmektedir.¹⁰

1.2. Yapay Sinir Ağlarının Tarihçesi

İnsan beyni hakkındaki ilk çalışmalar çok eski zamanlara dayanmaktadır. Modern elektroniğin gelişmesi ile birlikte, bu düşünce işlemi kullanmaya çalışmak doğal hale gelmiştir.¹¹

⁸ Ercan Öztemel, *Yapay Sinir Ağları*, İstanbul: Papatya Yayıncılık, 2003, s. 29-31.

⁹ Sağiroğlu, Age, s. 24

¹⁰ Şen, Age, s.9-10

¹¹ Çetin Elmas, *Yapay Sinir Ağları*, Ankara: Seçkin Yayıncılık, 2003, s. 27.

İlk yapay sinir ağı modeli 1943 yılında Warren McCulloch ve Walter Pitts tarafından gerçekleştirilmiştir. Bir nörobiyolojist olan Warren McCulloch ve bir istatistikçi olan Walter Pitts, “Sinir Aktivitesindeki Düşüncelere Ait Bir Mantıksal Hesap” başlıklı bir makale ile ilk dijital bilgisayarlara ışık tutmuştur. McCulloch ve Pitts, insan beyninin hesaplama yeteneğinden esinlenerek, elektrik devreleriyle basit bir sinir ağı modellemiştir. Yapay sinir hücreleri ile her türlü mantıksal ifadeyi formülize etmenin mümkün olduğunu göstermişlerdir. Hücrelerin birbiri ile paralel çalışması gerektiği fikrini ortaya atarak öğrenme kurallarını belirlemeye başlamışlardır.

1948 yılında Wiener “Cybernetics” isimli kitabında, sinirlerin çalışmasını ve davranış özelliklerini ele almıştır. 1949 ise, Donald Hebb “Organization of Behavior” isimli kitabında, öğrenme ile ilgili temel teoriyi ele almıştır. Hebb, yapay hücrelerden oluşan bir yapay sinir ağının değerlerini değiştiren bir öğrenme kuralı geliştirmiştir.¹²

John Von Neumann, Warren McCulloch ve Walter Pitts’in yazmış olduğu makaleyi, “elektronik beyinler” için bir kopya olarak görmüştür. John Von Neumann vakum tüplerini ve telgraf rölelerini kullanarak basit sinir işlevlerini taklit etmeyi önermiştir.

1951 yılında ilk neuro-bilgisayar üretilmiştir.

1954 yılında Farley and Clark tarafından rassal ağlar (random networks) ile adaptif tepki üretme kavramı ortaya atılmıştır ve bu kavram 1958 yılında Frank Rosenblatt ve 1961 yılında Caianiello tarafından geliştirilmiştir.

Bronx Yüksek Bilim Okulu’ndan Frank Rosenblatt, gözün hesaplamaları ile ilgilenmiştir. Bu bilim adamları, öğrenmenin ve zekânın herhangi bir özelliğinin simülasyonunda bilgisayarların aktif olarak nasıl kullanılabileceğini, 1956 yılında düzenlemiş oldukları ilk yapay zeka konferansında tartışmışlardır. Rosenblatt tarafından geliştirilen algılayıcı model (perceptron) yapay sinir ağları tarihinde önemli bir gelişmeye öncülük etmiştir. Çünkü bu model, daha sonraları geliştirilecek ve yapay sinir ağlarında devrim niteliğinde olacak olan çok katmanlı algılayıcıların temelini oluşturmaktadır.¹³

¹² http://batitrakya.dostweb.com/yapaysiniraglari/ek_dokuman.htm, Elmas, Age, s.27, Öztemel, Age, s. 37

¹³ Öztemel, Age, s. 37, http://batitrakya.dostweb.com/yapaysiniraglari/ek_dokuman.htm

1959'da, Stanford üniversitesinden Bernard Widrow, basit nöron benzeri elemanlara dayanan ve “adaline” (Adaptive Linear Neuron) olarak adlandırılan bir adaptif lineer elemanı geliştirmiştir. Adaline ve iki tabakalı biçimi olan “madaline” (Multiple Adaline); ses tanıma, karakter tanıma, hava tahmini ve adaptif kontrol gibi çok çeşitli uygulamalar için kullanılmıştır. Daha sonraları adaline, ayrık bir çıkış yerine sürekli bir çıkış üretmek için geliştirilmiştir. Widrow, telefon hatları üzerindeki ekoları elimine etmeye yarayan adaptif filtreleri geliştirmede, adaptif lineer eleman algoritmasını kullanmıştır. Bununla ilk defa Yapay Sinir Ağları, gerçek bir probleme uygulanmıştır.¹⁴

1960'lı yıllarda Grosberg, Kohonen, Rosenblatt, Widrow, Nilssons, Fukushima vb. gibi bilim adamlarının yapay sinir ağları üzerine çalışmaları olmuştur. Özellikle Nilssons tarafından yazılan ilke “Öğrenen Makineler” adlı kitap yapılan çalışmaların teorik çerçevesini oluşturmaya ve çalışmaları bir araya getirmeye neden olmuştur.

Yapay zekâ alanındaki araştırmacılar içerisinde istisnai bir yeri olan Marvin Minsky, aynı makaleden aldığı ilhamla makroskobik zekâ fikrini ortaya atmış ve uzman sistemlerin doğmasına neden olmuştur.

Minsky ve Papert'in “Perceptron” isimli kitaplarında özellikle yapay sinir ağlarına dayalı algılayıcılarının bilimsel değerinin olmadığını ve doğrusal (lineer) olmayan problemlere çözüm üretmediğini iddia etmişler ve bunu kanıtlamak için XOR probleminin çözülemediğini ispatlamışlardır.¹⁵ Bunun için iki katmanlı ileri beslemeli ağların kullanılabileceğini ileri sürmüşler ve tek katmanlı ağlardaki birçok sınırlamayı ortadan kaldırdığını göstermişlerdir. Fakat gizli katmanların ağırlıklarının nasıl değiştirileceği konusunda bir yöntem önerememişlerdir. Bu soruna Rumelhart ve arkadaşları geri yayılım yöntemi ile çözüm getirmişlerdir.¹⁶

Tek katmanlı ağların birçok konuda yetersiz kaldığının Minsky ve Papert tarafından gösterilmesi yapay sinir ağları üzerindeki çalışmaları bir süre durdurmuştur. Hatta bu olumsuz gelişmeler birçok araştırmacının hayal kırıklığına uğramasına sebep olmuştur. Bu durgunluk 1970'li yılların başından 1980'li yıllara kadar devam etmiştir.

¹⁴ http://batitrakya.dostweb.com/yapaysiniraglari/ek_dokuman.htm

¹⁵ Öztemel, Age, s. 38.

¹⁶ Elmas, Age,s.28.

Bu tarihler arasında bazı bilim adamları çalışmalarını kendi kişisel gayretleriyle sürdürmeye devam etseler de önemli bir ilerleme sağlanamamıştır.¹⁷

Helsinki Teknik Üniversitesi'nden Teuvo Kohonen, 1970'lerin ilk yıllarında adaptif öğrenme ve birleşik hafızalar üzerine temel çalışmalar yapmış ve yaptığı çalışmalar ile danışmansız öğrenme metotlarının gelişmesine ışık tutmuştur.

1982 yılında ilgi çeken bir başka gelişme, moleküller biyolojiden beyin kuramcılığına geçiş yapan bir model Caltech fizikçisi Hopfield tarafından sunulmuştur. Kendi adıyla anılan bir ağ yapısı mevcuttur ve birçok alana uygulanmıştır.¹⁸ Hopfield'in yaklaşımı beyne benzeyen basit bir model değil, kullanışlı bir alet yaratmaktı. Açık ve matematiksel analizleri kullanarak, böyle bir aletin nasıl çalışacağını ve ne yapabileceğini göstermiştir.

Hopfield tarafından yapılan bu çalışmalar ile yapay sinir ağlarının geliştirilebileceği ve özellikle geleneksel bilgisayar programlama ile çözülmesi zor olan problemlere çözüm üretebileceğini gösterdi. Gezgin satıcı problemine çözmesi buna iyi bir örnekti. Bu nedenle de yapay sinir ağlarına olan ilgi tekrar kurulmaya başlanmıştır.

1984'de Kohonen sinirlerin düzenli sıralanışına eşleme özelliği için danışmansız öğrenme yönteminin kullanıldığı ağları geliştirmiştir.¹⁹

1986 yılında Rumelhart ve arkadaşları tarafından çok katmanlı algılayıcı tipi ağlar için "geri yayılma" olarak adlandırılan öğrenme algoritması geliştirilmiştir. Bu algoritma güçlü olmakla birlikte oldukça karmaşık matematik esaslara dayanmaktaydı. Bu algoritmanın etkin bir öğrenmeyi mümkün kılma yeteneği dikkatleri üzerine çekmiştir. Günümüzde dahi en çok kullanılan öğretim sistemlerinden biri olan bu algoritmanın ortaya çıkarılması yapay sinir ağları alanında çığır açmıştır.²⁰

1987 yılında yapılan ilk yapay sinir ağları sempozyumundan sonra her sene değişik sempozyum ve konferanslar ile yapay sinir ağları tartışılmakta ve yeni modeller ve öğrenme teknikleri ortaya atılmaktadır.

¹⁷ Şen, Age, s.13

¹⁸ http://batitrakya.dostweb.com/yapaysiniraglari/ek_dokuman.htm, Simpson, Age, s.3.

¹⁹ Elmas, Age,s.29, Özetmel, Age, s. 39-40.

²⁰ Şen, Age, s.13.

Yapay sinir ağlarındaki bu ilerlemeye donanım teknolojisindeki gelişmelerin de çok katkısı olmuştur. Bilgisayarın boyutları küçülmüş kapasiteleri ve hızları artmıştır. Bu da yapay sinir ağlarının kullanımını kolaylaştırmıştır.²¹

Bazı araştırmacılar yapı tanımlama (pattern recognition) gibi problemlerin çözümüne yönelik çalışmalarını sürdürmüşlerdir. Bu dönem süresince bazı paradigmlar ortaya çıkmıştır. Grossberg ve Carpenter (1995) tarafından yapılan çalışmalar, yankı (resonating) algoritmaları araştıran bir düşünce okulunun temellerini atmıştır. Ayrıntılı kitaplar ve konferanslar çok farklı alanlarda uzmanlaşmış insanlara bir forum imkanı ve dolayısı ile bir etkileşim sağlamıştır. Akademik programlar oluşturulmuş ve en önemli üniversitelerde dersler açılmıştır. Artan ilgi ile beraber bu alandaki araştırmalara yönelik fonlar da artmış ve enstitüler kurulmuştur.

Sağlanan önemli ilerleme, günümüzde yapay sinir ağları alanında daha ileri araştırmalar için gerekli ilgi ve bilgi birikimini sağlamıştır. Sinir sistemi tabanlı işlemciler oluşturulmakta ve komplike problemlerin çözümüne yönelik uygulamalar gelişmektedir. Kısacası, bu alan günümüzde bir geçiş dönemi içinde olduğu görülmektedir.²²

1.3. Yapay Sinir Ağlarının Genel Özellikleri

Bir ağ yapısının çözebileceği problem uzayının, insan beyninin çözebildiği problem uzayının oldukça kısıtlanmış bir alt kümesi olacağı gözden kaçırılmamalıdır. Yapay sinir ağlarını çekici kılan aşağıda sıralanmış temel özelliklerin algılanışında bu noktanın gözden kaçırılmamasında da yarar vardır.

Birinci özellik sistemin paralelliği ve toplamsal işlevin yapısal olarak dağılmışlığıdır. Diğer bir deyişle birçok nöron eş zamanlı olarak çalışır ve karmaşık bir işlev çok sayıda küçük nöron aktivitesinin bir araya gelmesinden oluşur. Bu da, zaman içerisinde herhangi bir nöronun işlev dışı kalması durumunda ağın dikkate değer ölçüde etkilenmeyeceği anlamına gelir.

²¹ Öztemel, Age, s. 40.

²² Hasan Yurtoğlu, *Yapay Sinir Ağları Metodolojisi İle Öngörü Modellemesi: Bazı Makroekonomik Değişkenler İçin Türkiye Örneği*, Ankara: DPT Yayınları, 2005, s.39-41.

İkinci özellik ise genelleme yeteneği, diğer bir deyişle ağ yapısının, eğitim esnasında kullanılan nümerik bilgilerden eşleştirme esnasında kullanılan bilgileri çıkarması ve böylelikle eğitim sırasında kullanılmayan girdiler için de anlamlı yanıtlar üretebilmesidir.

Bir başka özellik ise ağ fonksiyonunun nonlinear oluşudur. Yapı üzerine dağılmış belli tipteki nonlinear alt birimler özelliklerle, istenen eşleştirmenin denetim ya da tanımlama işlemlerinde olduğu gibi nonlinear olması durumunda işlevin doğru biçimde yerine getirilebilmesini matematiksel olarak olası kılarlar. Burada, işlevin doğru biçimde gerçekleştirilmesi için yapısal bir esneklik gerekliliği vurgulanmalıdır. Yani ağ parametreleri, başarıyı artıracak –ya da maliyeti azaltacak- şekilde değiştirilebilmelidir.

Belirtilebilecek son özellik, sayısal ortamda tasarlanan sinir ağı yaklaşımlarının tüm devre gerçekleştirilebilirliklerinin olmasıdır. Bu da yakın gelecekte bu sistemlerin özellikle robotik uygulamaları ile birlikte düşünüldüğünde, günlük hayatta yaşam kalitesinin artırılmasında ne denli önemli bir rol oynayabileceklerine işaret eder.²³

Yapay sinir ağlarının karakteristik özellikleri uygulanan ağ modeline göre değişmektedir. Ancak tüm modeller için genel özellikler şunlardır:²⁴

- Yapay sinir ağları makine öğrenmesi gerçekleştirirler.
- Programları çalışma stili bilinen programlama yöntemlerine benzememektedirler.
- Bilgiyi saklarlar.
- Örnekleri kullanarak öğrenirler.
- Güvenle çalıştırılabilmeleri için önce eğitilmeleri ve performanslarının test edilmesi gerekmektedir.
- Görülmemiş örnekler hakkında bilgi üretirler. Bunu genelleme özelliği sayesinde yaparlar.
- Algılamaya yönelik olaylarda kullanılabilirler.
- Şekil ilişkilendirme ve sınıflandırma yapabilirler.
- Örüntü tamamlama gerçekleştirebilirler.
- Kendi kendine organize etme ve öğrenebilme yetenekleri vardır.

²³ Efe, M. Önder, Kaynak, Okyay, Age, s.1-3

²⁴ Öztemel, Age, s. 32-33

- Eksik bilgi ile çalışabilmektedirler.
- Hataya ve gürültüye karşı duyarlılığa ve toleransa sahiptirler.
- Belirsiz, tam olmayan bilgileri işleyebilmektedirler.
- Dereceli bozulma (Graceful degradation) gösterirler.
- Dağıtık belleğe sahiptirler. Veri dağıtılmış birleşik hafıza yapısı kullanılır ve bilgi farklı formlara dönüştürülerek işlenebilir.
- Sadece nümerik bilgiler ile çalışabilmektedirler.

Ayrıca:²⁵

- Doğrusal değildirler bu nedenle de hemen hemen her problemin çözümü için kullanılabilirler.
- Tanımı ya da parametreleri değişen probleme ve sisteme uygun çözümler sağlamak için tekrar tekrar eğitilebilirler.
- Paralel yapısı nedeniyle büyük ölçekli entegre devre (VLSI) teknolojisi ile gerçekleştirilebilir. Bu da yapay sinir ağlarının gerçek-zamanlı uygulamalarda kullanılmasını sağlar.
- Farklı yapıları için internet ortamından rahatlıkla bulunabilecek birçok yazılım geliştirilmiştir.

1.4. Yapay Sinir Ağlarının Avantajları ve Dezavantajları

Yapay sinir ağlarının en büyük avantajı, öğrenme kabiliyetinin olması ve farklı öğrenme algoritmaları kullanabilmesidir. Yapay sinir ağları uyarlanabilir ve esneyebilir. Tek bir yapay sinir ağı, herhangi bir işleve ilke olarak yaklaşabilir. Yapay sinir ağlarını yapısını ve öğrenme kuralını değiştirmek zorunda değiliz, sadece öğretim materyalini değiştirerek öğrenmeyi sağlayabiliriz. Bunlara karşılık dezavantajları ise sistemin çalışmasının analiz edilememesi ve öğrenme işleminde başarılı olunama riski olmasıdır. Yapay sinir ağlarının avantaj ve dezavantajları aşağıdaki gibi sıralanabilir:²⁶

1.4.1. Avantajları

- Matematiksel modele ihtiyaç duymazlar.

²⁵ Sağıroğlu ve diğerleri, Age, s. 40-42.

²⁶ Elmas, Age, s.26-27, Öztemel, Age, s. 207-209.

- Kural tabanı kullanımı gerektirmezler.
- Öğrenme kabiliyeti vardır ve farklı öğrenme algoritmalarıyla öğrenebilirler.
- Matematiksel olarak modellenmesi zor olan ya da mümkün olmayan problemleri rahatlıkla çözümleyebilir.
- Bir problem için ağ modelleneyeceği zaman herhangi bir bilgiye ihtiyaç duyulmaz sadece örnek gereklidir. Örnek bulmaksa bilgi bulmaktan kolaydır.
- Gerçek hayatta olaylar ve olayların arkasındaki değişik etkenlerin birbiri ile ilişkilerini bilmek zordur. Ancak bu yapay sinir ağları ile otomatik olarak yapılır.
- Olaylar ve olaylar arasındaki ilişkiler doğrusal olmadığı için bu tür problemleri modellemek de, çözmek de zordur. Çözüm için bazı varsayımlar yapmak gerekir. Bu da modellenen sistem ile gerçek sistem arasında farklılık olmasına sebep olur. Oysa yapay sinir ağları doğrusal olmayan ilişkileri içinde geleneksel yöntemlerden daha iyi ve gerçekçi çözümler üretir.
- Yapay sinir ağları uygulamaları hem pratik hem de ekonomiktir. Sadece problemle ilgili örneklerin toplanması ve basit bir programla problem çözülebilir.
- Yapay sinir ağları diğer sistemlere göre zaman bakımından da daha verimlidir. Örneklerin bulunması, probleme uygun ağın oluşturulması, ağın öğrenmesi, diğer örnekler için kullanıma geçmesi çok kısa zamanda yapılabilmektedir.
- Yapay sinir ağları aynı olay için yeni bilgilerin olması ve bazı değişiklikler söz konusu olduğunda yeniden eğitilip ortama uyum sağlayabilirler.
- Yapay sinir ağları paralel olarak çalıştığı için gerçek-zamanlı uygulamalar için kullanımı kolaydır.

1.4.2. Dezavantajları

- Sistem içerisinde ne olduğu bilinemez.
- Bazı ağlar hariç kararlılık analizi yapılamaz.
- Farklı sistemlere uyarlanması zor olabilir.
- Yapay sinir ağları ile bir uygulama geliştirilirken model seçiminde ve ağların topolojisini belirlerken belli bir kural yoktur. Doğru seçimlerin yapılması tamamen kullanıcının tecrübesine bağlıdır.

- Herhangi bir problem için örneklerin belirlenmesinde de yine bir kural söz konusu değildir. Kullanıcı tarafından örnekler belirlenip formülize edilecektir.
- Ağın davranışlarının açıklanması mümkün olmadığı için ağa güven azalmaktadır.
- Bazen eğitim gerçekleşmesi uzun zaman alabilir.
- Bulunan çözümün en uygun çözüm olduğunu söylemek mümkün değildir. Sadece iyi çözümlerden biri olduğu söylenebilir. Oysa geleneksel yöntemler en uygun sonuçlar üretebilirler.
- Örneklerin bulunmasının zor olduğu ya da doğru örnekler için karar verilmediği durumlarda sağlıklı çözüm üretilememektedir.

Yapay sinir ağlarının daha iyi anlaşılması için örnek teşkil eden doğal sinir ağı olan biyolojik sinir ağlarının yapısının ve çalışma ilkelerinin iyi bilinmesi gerekmektedir.²⁷

1.5. Biyolojik Nöron Yapısı

Biyolojik sinir ağları, beynimizde bulunan birçok sayıda sinir hücresinin bütünüdür. Bir sinir ağı milyarlarca sinir hücresinin bir araya gelmesiyle oluşur. Sinir hücreleri birbirleri ile bağlanarak işlevlerini yerine getirirler. Beynimizde yaklaşık olarak 10^{10} adet sinir hücresi ve bunların da yine yaklaşık olarak 6×10^{13} , ten fazla sayıda bağlantısı vardır. Diğer taraftan bir sinir hücresinin tepki hızı, günümüz bilgisayarlarına göre oldukça yavaş olmakla birlikte duyu bilgileri son derecede hızlı değerlendirebilmektedir. Bu nedenle insan beyni; öğrenme, birleştirme, uyarılma ve genelleştirme yeteneği nedeniyle son derece karmaşık, doğrusal olmayan ve paralel dağılmış bir bilgi işleme sistemi olarak tanımlanabilir. Biyolojik sinir ağlarının performansları, küçümsenemeyecek kadar yüksek ve karmaşık olayları işleyebilecek yetenektedir. Yapay sinir ağları ile bu yeteneğin bilgisayarlara kazandırılması amaçlanmaktadır.

Biyolojik sinir ağları insan beyninin çalışmasını sağlayan en temel taşlardan biridir. Biyolojik sinir ağları insanın beş duyu organından gelen bilgiler ışığında geliştirdiği algılama ve anlama mekanizmalarını çalıştırarak olaylar arasındaki ilişkileri öğrenir. Şekil 1.1 insan beyninin resmini göstermektedir.

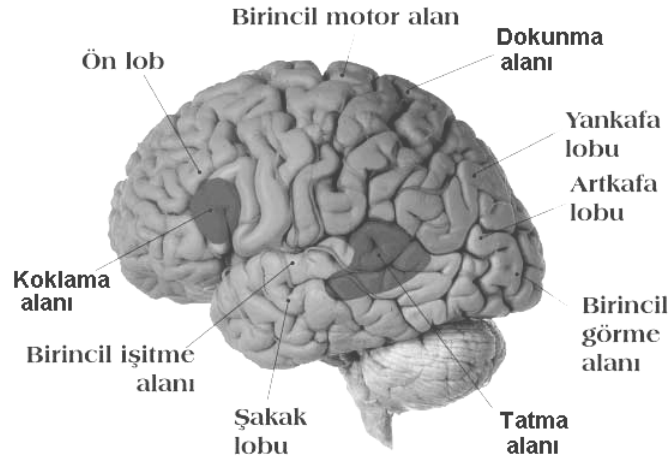
²⁷ Elmas, Age, s.29.



Şekil 1.1 İnsan beyni.

İnsan beyninin değişik bölgeleri değişik fonksiyonları yerine getirmektedir.

Şekil-1.2. ise beş duyunun beyin içindeki bölgelerini göstermektedir.



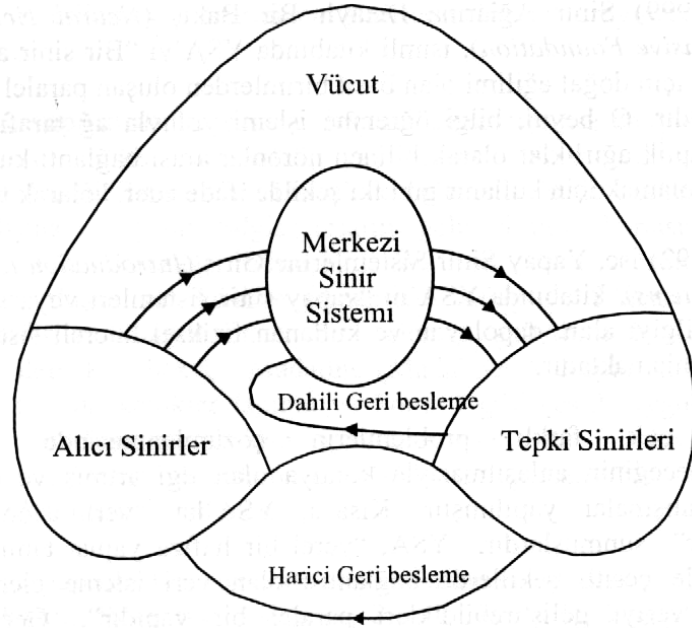
Şekil 1.2 Beyin üzerindeki beş temel duyu bölgesi.

Duyu organlarından gelen bilgiler (sinyaller) beyin sinir sistemi sayesinde beyine taşınır ve beynin oluşturduğu kararlar da yine sinir sistemi tarafından vücudun organlarına eylem olarak gönderilir.²⁸

Nöronlar buldukları yer ve işlevlerine göre alıcı nöronlar (afferent ya da receptor), tepki nöronları (efferent ya da effector), ara nöronlar (interneurons) olmak üzere üç grupta sınıflandırılırlar. Biyolojik sinir sistemi, merkezinde sürekli olarak bilgiyi alan, yorumlayan ve uygun bir karar üreten beynin (merkezi sinir ağı) bulunduğu 3 katmanlı bir sistem olarak açıklanmaktadır. Alıcı nöronlar çevreden hücrelere gelen

²⁸ Öztemel, Age, s. 45-46.

işaretleri (ses, ışık gibi) algırlarlar. Tepki nöronları beyinden vücudun bezlerine ve kaslarına giden çıkışları taşırlar. Bu iki tür nöron arasındaki ara nöron olarak bilinen diğer tüm nöronlar algılama, düşünme ve hafıza ile ilgili işlemleri üstlenirler.²⁹ Şekil-1.3'de biyolojik sinir sisteminin blok şeması görülmektedir.



Şekil 1.3 Biyolojik sinir sisteminin blok gösterimi.

Sinir hücreleri nöron olarak bilinir. Nöron, özellikle beyin olmak üzere sinir sisteminin temel birimidir ve başlıca üç kısımdan oluşur. Bunlar:

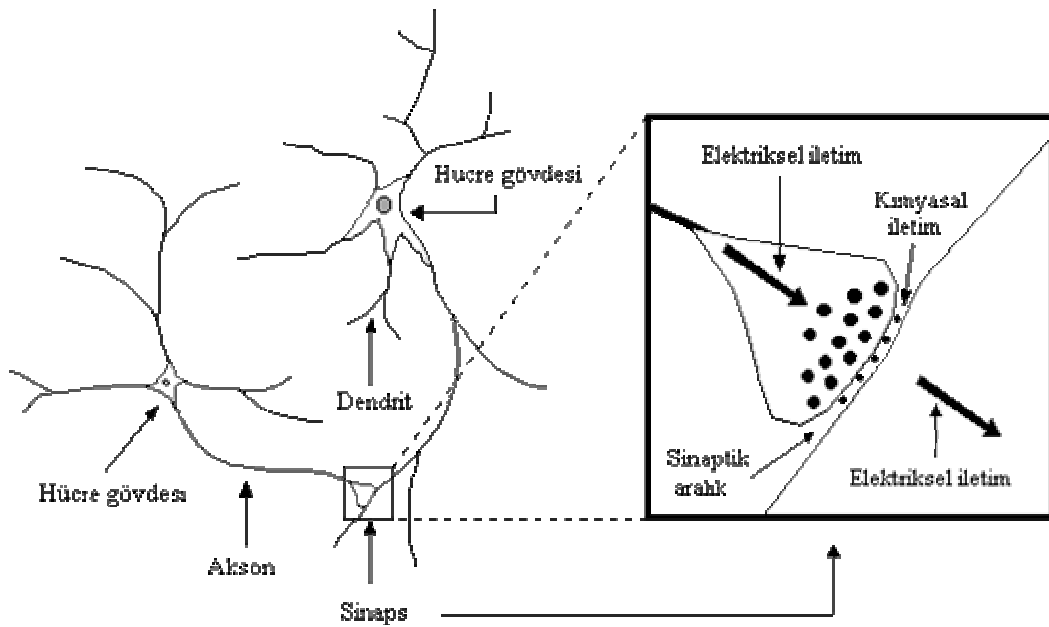
- Gövde (cell body),
- Gövdeye giren sinyal alıcı lifler (dendrit),
- Gövdeden çıkan sinyal iletilici lifler (axon).

Şekil-1.4'de bir nöron hücresinin yapısı verilmiştir. Bir nörondan yüzlerce, hatta binlerce dendrit çıkabilir. Bunların uzunluğu çoğunlukla bir milimetreden daha kısadır. Bazıları ise birkaç milimetre uzunluğunda olabilir. Dendritler çevre hücrelerden gelen sinyalleri (impulse) alıp, gövdeye ulaştırırlar. Her nöron, dendritler vasıtasıyla diğer nöronlardan gelen işaretleri alan ve birleştiren basit bir mikroişleme birimidir.³⁰

²⁹ S.Y. Kung ve K.I. Diamantaras, Principal Component Neural Networks, Theory and Applications, NewYork: Jhon Wiley & Sons, 1996, s. 5-7.

³⁰ [Http://batitrakya.dostweb.com/yapaysiniraglari/ek_dokuman.htm](http://batitrakya.dostweb.com/yapaysiniraglari/ek_dokuman.htm), Kung, Diamantaras, Age, s.5-7.

Beyin, sıkışık olarak ara bağlaşımlı milyarlarca nörondan oluşmaktadır. Her eleman kendi aralarında oldukça çok sayıda nörona (eleman başına yaklaşık olarak 104 bağlantı) bağlanmıştır. Bir nöronun aksonu (çıkış yolu) ayrıştırılmıştır ve bir sinaps olarak adlandırılan bir jonksiyon vasıtasıyla diğer nöronların dendritlerine bağlanmıştır. Bu jonksiyon uçlarındaki iletim doğal olarak kimyasaldır ve işaretin miktarı, akson tarafından serbest bırakılan kimyasalların büyüklüğüne bağlı olarak transfer edilir ve dendritler vasıtasıyla alınır. Bu sinaptik büyüklük, beyin öğrenirken neyin modifiye edildiğini belirtir. Bu sinaps, beynin temel hafıza mekanizmasına dayanarak nöron içerisindeki bilginin işlenmesi ile birleştirilir.³¹



Şekil 1.4 Nöron.

Verilen özellikte milyarlarca sinir hücresi bir araya gelerek sinir sistemini oluşturmaktadır. YSA'lar, insan beyninin çalışma prensibi örnek alınarak geliştirilmeye çalışılmıştır ve aralarında yapısal olarak bazı benzerlikler vardır.³² Bu benzerlikler Tablo-1.1'de ve istatistiksel prensiplerle yapay sinir ağları arasındaki terminolojik ilişkiler de Tablo-1.2 de verilmiştir.³³

³¹ http://batitrakya.dostweb.com/yapaysiniraglari/ek_dokuman.htm, Kung, Diamantaras, Age, s.5-7.

³² http://batitrakya.dostweb.com/yapaysiniraglari/ek_dokuman.htm, Judith E. Dayhoff, *Neural Network Architectures An Introduction*, New York: Van Nostrand Reinhold, 1990, s. 1-6.

³³ Sağiroğlu ve diğerleri, Age, s. 33

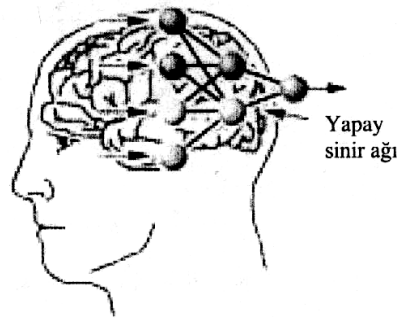
Tablo 1.1 Sinir sistemi ile YSA'nın benzerlikleri.

Biyolojik Sinir Sistemi	Yapay Sinir Ağları
Sinir Sistemi	Sinirsel Hesaplama Sistemi
Nöron (Sinir)	İşlem elemanı
Dendrit	Toplama fonksiyonu
Hücre gövdesi	Transfer fonksiyonu
Aksonlar	Eleman çıkışı
Sinapslar	Ağırlıklar

Tablo 1.2 İstatistiksel yöntemlerle yapay sinir ağlarının benzeşimleri.

İstatistiksel	Yapay Sinir Ağları
Model	Ağ
Tahmin	Öğrenme
Regresyon	Danışmanlı öğrenme
İnterpolasyon	Genelleştirme
Gözlem	Öğrenme algoritması
Parametre	Ağ parametreleri (ağırlıklar, katsayılar)
Bağımsız değişken	Giriş verileri
Bağımlı değişken	Çıkış verileri
Sınır regresyonu	Ağırlık budama işlemi

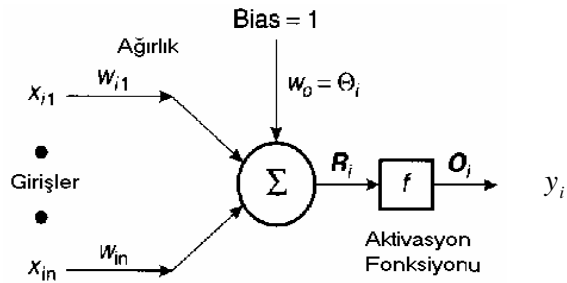
Biyolojik sinir ağlarının yapay sinir ağlarına bir dayanak olduğu şematik olarak Şekil-1.5'de gösterilmiştir.³⁴

**Şekil 1.5** Biyolojik ve yapay sinir ağları.

³⁴ Öztemel, Age, s. 48.

1.6. Yapay Sinir Hücresi

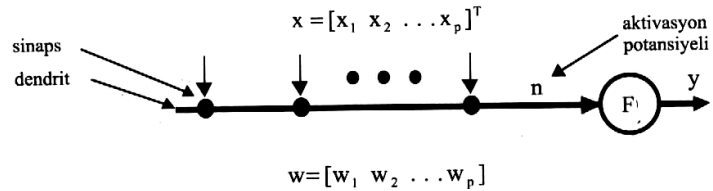
Biyolojik sinir ağlarının sinir hücreleri olduğu gibi yapay sinir ağlarının da yapay sinir hücreleri vardır. Yapay sinir hücreleri mühendislik biliminde işlem elemanları olarak da adlandırılmaktadır. Şekil-1.6'da yapay bir sinir (düğüm) gösterilmiştir.³⁵



Şekil 1.6 Yapay sinir hücresinin yapısı.

Girişler x_i sembolüyle gösterilmiştir. Bu girişlerin her biri ağırlık w_{ij} ile çarpılır. Basitçe, bu ürünler eşik değeri θ_j ile toplanır ve sonucu oluşturmak için etkinlik işlevi ile işlem yapılır ve y_i çıkışı alınır.

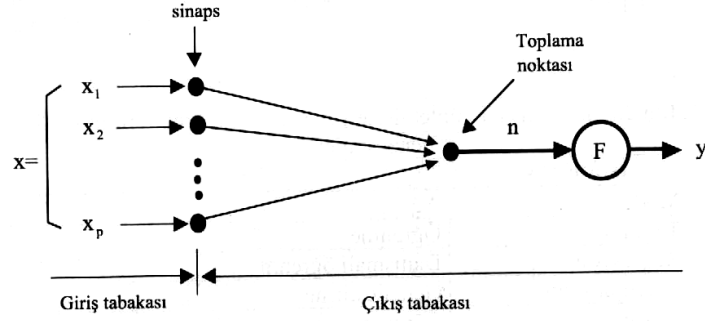
Yukarda verilen şeklin yanı sıra bir yapay nöronun farklı gösterim şekilleri vardır. Bunlar Şekil-1.7 a,b,c,d de verilmiştir.³⁶



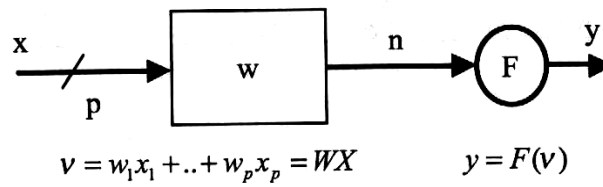
(a) Dendrit gösterim

³⁵ Öztemel, Age, s. 48.

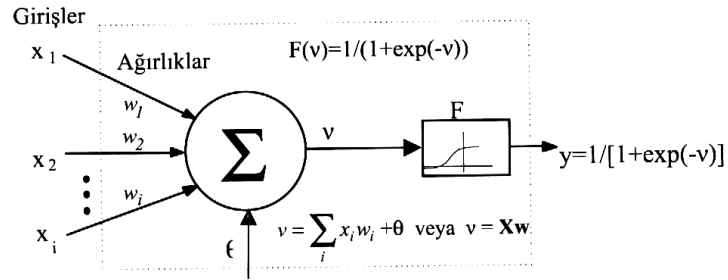
³⁶ Sağiroğlu ve diğerleri, Age, s. 34.



(b) İşaret akış grafi gösterimi



(c) Blok diyagram gösterimi



(d) Detaylı gösterim

Şekil 1.7 Bir yapay nöronun temel grafik gösterimleri.

Tüm yapay sinir ağları bu temel yapıdan üretilmiştir. Bu yapıdaki farklılıklar yapay sinir ağlarının farklı sınıflandırılmalarını sağlar. Bir yapay sinirin öğrenme yeteneği, seçilen öğrenme algoritması içerisinde ağırlıkların uygun bir şekilde ayarlanmasına bağlıdır.³⁷

³⁷ Elmas, Age, s.32.

1.7. Bir Yapay Sinir Ağının Temel Elemanları

Yapay sinir ağları aşağıdaki varsayımlar üzerine kurulmuştur:³⁸

- Bilgi işleme nöron olarak isimlendiren basit elemanlarda gerçekleştirilir
- İşaretler nöronlar arasındaki bağlantılardan geçer
- Her bağlantı birçok işareti taşıyan bir ağırlığa sahiptir
- Her nöron kendi giriş değerine çıkış işaretini belirlemek için aktivasyon fonksiyonu uygular.

Yapay sinir ağları, birbirine bağlı çok sayıda işlemci elemanlardan oluşmuş, genellikle paralel işleyen yapılar olarak adlandırılabilir. Yapay sinir ağlarındaki işlem elemanları (düğümler) basit sinirler olarak adlandırılırlar. Bir yapay sinir ağı birbirine bağlantılı, çok sayıda düğümlerden oluşur.

Yapay sinir ağları insan beyni gibi öğrenme, hatırlama ve genelleme yeteneğine sahiptirler.

İnsan beyninde öğrenme 3 şekilde olur;

- Yeni aksonlar üreterek,
- Aksonların uyarılmasıyla,
- Mevcut aksonların güçlerini değiştirerek.

Her aksonun üzerinden geçen işaretleri değerlendirebilecek yetenekte olduğu savunulmaktadır. Aksonun bu özelliği, bir işaretin belli bir sinir için ne kadar önemli olduğunu göstermektedir.

Yapay sinir ağlarının temel birimi işlem elemanı ya da düğüm olarak adlandırılan yapay bir sinirdir. Bir yapay sinir, biyolojik sinirlere göre daha basit olmasına karşın, biyolojik sinirlerin 5 temel işlemini taklit ederler.³⁹ Bunlar;

1.7.1. Girişler

Girişler tarafından bir yapay sinir hücresine bir başka yapay sinir hücresinden veya dış dünyadan bilgi alışı yapılır. Bunlar ağın öğrenmesi istenen örnekler tarafından belirlenir.

³⁸ Fausett, Age,s.3.

³⁹ Elmas, Age, s.31-32.

1.7.2. Ağırlıklar

Ağırlıklar bir yapay sinir hücresinin girişleri tarafından alınan bilgilerin önemini ve hücre üzerinde etkisi gösteren uygun katsayılardır. Her bir giriş için bir ağırlık vardır. Bu ağırlığın büyük olması bu girişin önemli olduğu ya da ağırlığın küçük olması girişin önemsiz olduğunu göstermez. Bir ağırlığın değerinin sıfır olması o ağ için en önemli olay olabilir. Eksi değerler de yine girişin önemsiz olduğunu göstermez. Ağırlığın artı ve eksi olması girişin etkisinin pozitif ya da negatif olduğunu gösterir. Ağırlıklar değişken ya da sabit olabilirler.⁴⁰

1.7.3. Toplama İşlevi

Toplama işlevi bir yapay sinirdeki her bir giriş ile o girişe ait olan ağırlığın çarpılarak bu çarpımların toplanmasıdır.

$$Net\ Toplam = \sum_i^n x_i w_i \quad (1.1)$$

Ancak birçok uygulama aşağıdaki gibi eşik değeri olan θ 'da bu toplamaya katılmıştır.

$$Net\ Toplam = \sum_i^n x_i w_i + \theta \quad \text{ya da} \quad Net\ Toplam = \sum_i^n x_i w_i - \theta \quad (1.2)$$

θ eşik değerinin girişlerden bağımsız olduğu için bütün girişlerin sıfır olması durumunda çıkış değerinin sıfır değil de eşik değerine eşit olduğu görülür ki bu da, belirtilen şartlar altında nöron çıkışının sıfır olması zorunluluğunu ortadan kaldırır. Eşik değerinin kullanımı, toplama fonksiyonuna +1 ya da -1 değerine sahip sabit bir girişin θ ağırlığına sahip bir bağlantı ile eklendiği şeklinde yorumlanır.⁴¹

Ayrıca her model ve her uygulama için bu toplama fonksiyonunun kullanılması şart değildir. Bazı modeller, kullanılacak toplama fonksiyonunu kendileri belirler. Çoğu zaman daha karmaşık olan değişik toplama fonksiyonları kullanılır. Bunlar Tablo-1.3'de gösterilmiştir. Bazı durumlarda girişlerin değeri önemli olurken, bazılarında sayısı önemli olabilir. Bir problem için en uygun toplama fonksiyonunu belirlemek için bir

⁴⁰ Öztemel, Age, s. 49.

⁴¹ Efe ve Kaynak, Age s.7.

formül geliştirilememiştir. Bu yüzden en uygun toplama fonksiyonunun bulunması deneme yanılma yoluyla belirlenir. Ayrıca aynı problem için kullanılan yapay sinir hücrelerinden hepsi aynı toplama fonksiyonunu kullanabileceği gibi her biri için farklı toplama fonksiyonu kullanılabilir.⁴²

Tablo 1.3 Toplama fonksiyonu örnekleri

Toplama İşlevi	Açıklama
Çarpım Net Girdi = $\prod_i x_i w_i$	Girişler ve ağırlıklar çarpılır sonra bu değerler de birbiriyle çarpılır
Maksimum (En Çok) Net Girdi = $\text{Max} (x_i w_i), i=1,2,\dots,N$	N adet giriş ve ağırlık birbiriyle çarpıldıktan sonra en büyüğü net girdi olarak kabul edilir.
Minimum (En Az) Net Girdi = $\text{Min} (x_i w_i), i=1,2,\dots,N$	N adet giriş ve ağırlık birbiriyle çarpıldıktan sonra en küçüğü net girdi olarak kabul edilir.
Çoğunluk Net Girdi = $\sum_i \text{sgn} (x_i w_i)$	N adet giriş ve ağırlık birbiriyle çarpıldıktan sonra pozitif ve negatif olanların sayısı bulunur. Bunlardan büyük olan net girdi olarak alınır.
Kümülatif Toplam Net Girdi = $\text{Net} (\text{eski}) + \sum_i (x_i w_i)$	Hücreye gelen bilgiler ağırlıklı olarak toplanır ve daha önce gelen bilgilere eklenerek hücrenin net girdisi bulunur.

1.7.4. Etkinlik İşlevi (Aktivasyon Fonksiyonu)

Yapay nöronun davranışını belirleyen önemli bir etken aktivasyon fonksiyonudur. Buna aynı zamanda “öğrenme eğrileri” de denir. Aktivasyon fonksiyonu hücreye gelen net girdiyi, diğer bir deyişle toplama fonksiyonunu işleyerek bu hücreye gelen girişlere karşılık olan çıkışı belirler.

Aktivasyon fonksiyonu da yapay sinir ağlarının farklı modelleri için farklı olabilir. En uygun aktivasyon fonksiyonunu belirlemek için geliştirilmiş bir fonksiyon yoktur. Toplama fonksiyonuna benzer şekilde hücrelerin hepsi için aynı aktivasyon

⁴² Öztemel, Age, s. 49-50.

fonksiyonu kullanma zorunluluğu yoktur. Bazıları aynı aktivasyon fonksiyonunu kullanırken bazıları kullanmayabilir.⁴³

Bazı modeller için özellikle de Çok Katmanlı Algılayıcı model için bu fonksiyon türevi alınabilir ve sürekli olmalıdır. Yapay sinir ağlarının kullanım amacına göre tek veya çift yönlü aktivasyon fonksiyonu kullanılabilir.

Doğrusal olmayan fonksiyonların kullanılması yapay sinir ağlarının çok karmaşık ve farklı problemlere uygulanmasını sağlamıştır. En çok kullanılan aktivasyon fonksiyonları şunlardır:⁴⁴

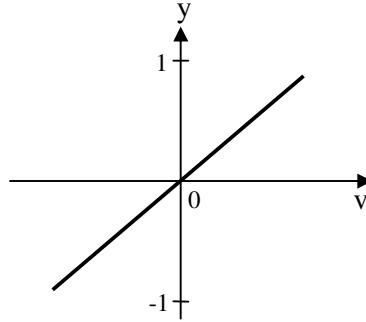
1.7.4.1. Doğrusal Fonksiyon

Doğrusal fonksiyon, hücreye gelen girişleri olduğu gibi çıkışa verir. Fonksiyonun şekli Şekil-1.8'de verilmiştir. Çoğunlukla ADALİNE olarak bilinen doğrusal modelde, klasik işaret işleme ve regresyon analizinde kullanılır. Denklemi;

$$v = \sum_i^n x_i w_i \quad \text{veya} \quad v = \sum_i^n x_i w_i + \theta \quad \text{olmak üzere;}$$

$$y = F(v) = Av \quad (1.3)$$

Formüldeki A sabit bir katsayıdır.



Şekil 1.8 Doğrusal veya lineer fonksiyon.

⁴³ Öztemel, Age, s. 50.

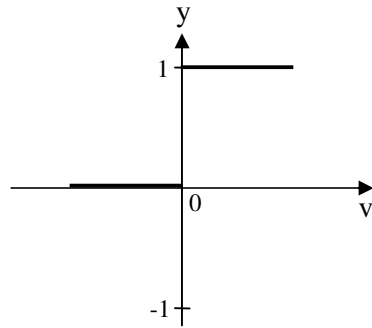
⁴⁴ Öztemel, Age, s. 50-51, Sağıroğlu ve diğerleri, Age, s. 37-40, Simon Haykin, *Neural Network, A Comprehensive Foundation*, New York: Macmillan College Publishing Company, 1994, s. 10-12.

1.7.4.2. Basamak Fonksiyonu

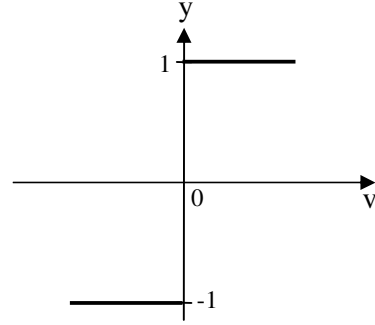
Basamak fonksiyonu tek veya çift kutuplu olabilir. Bu fonksiyonların şekli Şekil-1.9'da, matematiksel ifadeleri de Denklem 1.4 ve 1.5'de verilmiştir. Perceptron (Basit Algılayıcı Model) olarak bilinen yapay sinir hücresi aktivasyon fonksiyonu olarak bu fonksiyonu kullanır.

$$y = F(v) = \begin{cases} 1 & v \geq 0 \\ 0 & v < 0 \end{cases} \quad (1.4)$$

$$y = F(v) = \begin{cases} +1 & v \geq 0 \\ -1 & v < 0 \end{cases} \quad (1.5)$$



(a) Tek kutuplu



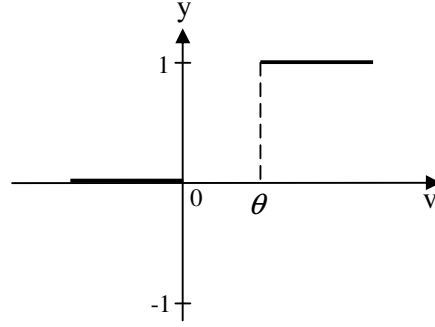
(b) Çift kutuplu

Şekil 1.9 Basamak fonksiyonları.

1.7.4.3. Kutuplamalı Basamak Fonksiyonu

Kutuplama değeri tek kutuplu ve çift kutuplu basamak fonksiyonunun her ikisine de eklenebilir. Aktivasyon fonksiyonu eşik değeri olan θ 'yı aştığı zaman nöron aktif olur. Tek kutuplu basamak fonksiyonu için Denklem 1.6, 1.7 ve Şekil-1.10,1.11 aşağıda verilmiştir.

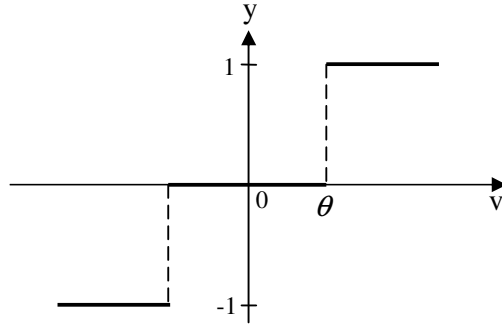
$$y = F(v) = \begin{cases} 1 & w.x \geq \theta \\ 0 & w.x < \theta \end{cases} \quad (1.6)$$



Şekil 1.10 Tek kutuplamalı basamak fonksiyonu.

Çift kutuplu basamak fonksiyonu ise;

$$y = F(v) = \begin{cases} +1 & w.x \geq \theta \\ -1 & w.x < \theta \end{cases} \quad (1.7)$$

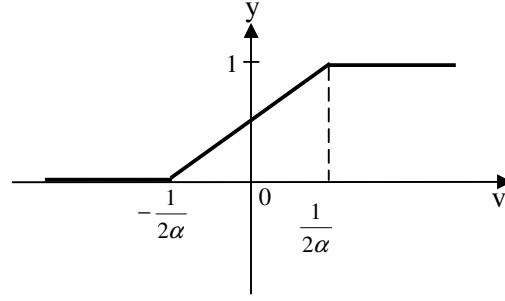


Şekil 1.11 Çift kutuplamalı basamak fonksiyonu.

1.7.4.4. Parçalı Doğrusal Fonksiyon

Bu fonksiyon, küçük aktivasyon potansiyeli için, α kazancı olan bir doğrusal toplayıcı (Adaline) olarak çalışır. Büyük aktivasyon potansiyeli için, nöron doyuma ulaşır ve çıkış işareti 1 olur. Büyük kazançlar için, $\alpha \rightarrow \infty$ iken, parçalı doğrusal fonksiyon basamak fonksiyonu gibi davranır. Denklem 1.8'de fonksiyonu, Şekil-1.12'de grafiği gösterilmiştir.

$$y = F(v) = \begin{cases} 0 & v \leq -1/2\alpha \\ \alpha v + \frac{1}{2} & |v| < 1/2\alpha \\ 1 & v \geq 1/2\alpha \end{cases} \quad (1.8)$$

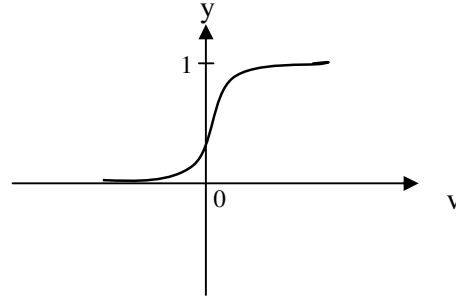


Şekil 1.12 Parçalı doğrusal fonksiyon.

1.7.4.5. Sigmoid Tipi Fonksiyon

Uygulamalarda en çok kullanılan aktivasyon fonksiyonlarından biridir. Fonksiyonun formülü Denklem 1.9'da, şekli ise Şekil-1.13'de gösterilmiştir. Fonksiyonun en aktif bölgesi 0,2 ile 0,8 arasındadır. Tek kutuplu fonksiyon olarak da adlandırılır.

$$y = F(v) = \frac{1}{1 + e^{-v}} = \frac{1}{2} [\tanh(v/2) + 1] \quad (1.9)$$

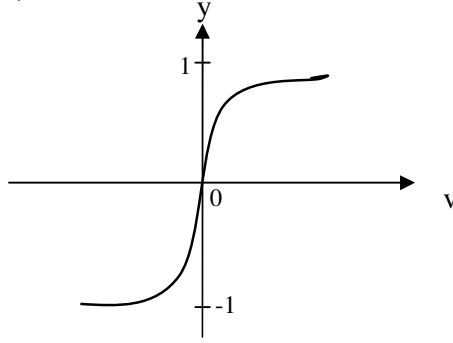


Şekil 1.13 Sigmoid tipli fonksiyon.

1.7.4.6. Tanjant Hiperbolik Tipli Fonksiyon

Uygulamalarda çok kullanılan aktivasyon fonksiyonlarından biri de Tanjant Hiperbolik fonksiyondur. Bu fonksiyon çift kutuplu fonksiyon olarak da bilinir. Giriş uzayının genişletilmesinde etkin bir şekilde kullanılan bir fonksiyondur. Fonksiyonun şekli Şekil-1.14'de verilmiştir. Formülü ise Denklem 1.10'da gösterildiği gibidir.

$$y = \frac{1 - e^{-2v}}{1 + e^{2v}} = \tanh(\beta v) \quad (1.10)$$

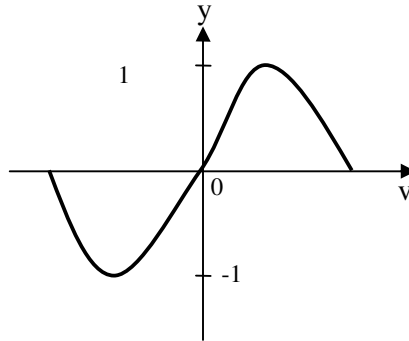


Şekil 1.14 Tanjant hiperbolik tipli fonksiyon.

1.7.4.7. Sinüs Tipli Fonksiyon

Öğrenilmesi düşünülen olaylar sinüs fonksiyonuna uygun dağılım gösteriyorsa bu gibi durumlarda aktivasyon fonksiyonu olarak sinüs fonksiyonu kullanılır. Fonksiyonun şekli Şekil-1.15’de ve formülü Denklem 1.11 aşağıda verilmiştir.

$$y = f(v) = \text{Sin}(v) \quad (1.11)$$



Şekil 1.15 Sinüs tipli fonksiyon.

1.7.5. Çıkış İşlevi

Çıkış $y = f(v)$, aktivasyon fonksiyonunun sonucunun dış dünyaya veya diğer sınırlara gönderilmesidir. Bu sinirin çıkışı kendine ve kendinden sonra gelen bir ya da daha fazla sayıda sinire giriş olabilir.

Biyolojik sinirde olduğu gibi bir sinirin birden fazla girişi varken tek bir çıkış etkinliği vardır. Ancak ağ şeklinde gösterildiğinde bir sinirin birden fazla çıkışı varmış gibi gözükür. Bu sadece şekil üzerinde böyledir.

Her bir düğümün çıkışı aktivasyon fonksiyonunun sonucuna eşittir. Fakat bazı ağ yapılarında komşu düğümler arasında yarışma oluşturmak için aktivasyon fonksiyonlarını yeniden düzenlenebilir. Bunun sonucunda yarışmacı düğümlerin girişleri, hangi düğümlerin öğrenme işlemine katılacağına karar verilmesine yardımcı olurlar.⁴⁵

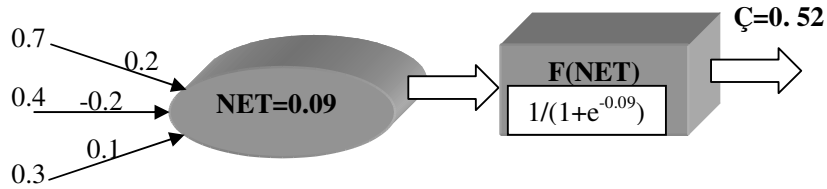
Bir yapay sinir ağının bu 5 temel elemanı dışında zaman zaman ihtiyaç duyulduğunda kullanılan bir elemanı daha vardır. Bu eleman ölçekleme ve sınırlama olarak adlandırılır.

1.7.6. Ölçekleme ve Sınırlama

Etkinlik işlevinin sonuçları ölçek veya sınır işlemlerinden geçirilebilir. Ölçeklendirme basitçe bir ölçek etmeni ile aktivasyon değerinin çarpılması sonucudur. Sınırlama ise, ölçeklenen bu sonuçların en az ve en çok sınırlarını aşmasına engel olmaktır.⁴⁶

1.8. Yapay Sinir Hücresinin Çalışma Prensipleri

Şekil-1.16'da girişleri ve ağırlıkları verilmiş olan bir yapay sinir hücresinin çalışması şöyledir:



Şekil 1.16 Bir yapay sinir ağının çalışması örneği.

Hücreye gelen net girdi, ağırlıklarla girişler çarpılarak aşağıdaki gibi hesaplanır.

$$\text{NET Girdi} = 0.7 * 0.2 + 0.4 * (-0.2) + 0.3 * 0.1$$

⁴⁵ Öztemel, Age, s. 51, Elmas, Age, s.35.

⁴⁶ Elmas, Age, s.35

$$\text{NET Girdi} = 0.14 - 0.08 + 0.03$$

$$\text{NET Girdi} = 0.09$$

Hücresin sigmoid tipli aktivasyon fonksiyonuna göre çıkışı $y=f(v)$ şöyledir;

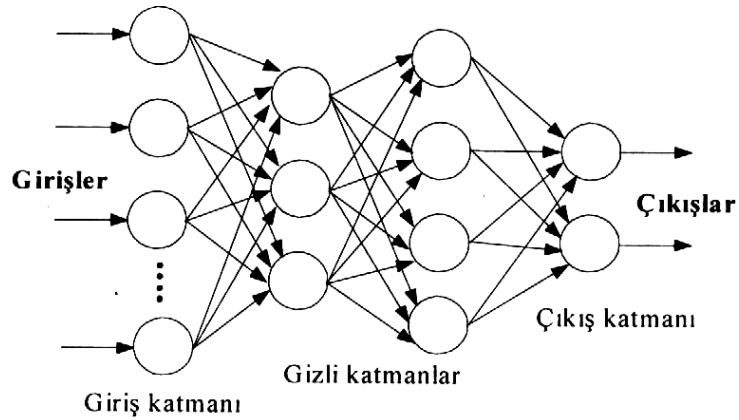
$$y = f(v) = 1 / (1+e^{-0.09})$$

$$y = 0.52$$

Sonuçta verilen girdilere karşılık yukarıda işlemler sonucunda y çıkış değeri bulunur.

1.9. Yapay Sinir Ağı Yapısı

Birden fazla yapay sinir hücresi bir araya gelerek yapay sinir ağını oluşturur. Bu hücrelerin bir araya gelmesi rasgele olmaz. Ağdaki sinir hücreleri katmanlar içine yerleştirilir. Çoğunlukla hücreler 3 katman halinde ve her katman içinde paralel olacak şekilde bir araya gelerek ağı oluştururlar. Her katmandaki düğümler sadece kendinden bir önceki katmanda bulunan düğümlerden giriş alabilir. Bir düğüm kendinden sonraki bir sinir hücresine bağlanırken kendine asla bağlanamaz. Son katmandaki işaretler ağın çıkışıdır. Bu yapının daha iyi anlaşılması için Şekil-1.17'de çok katmanlı bir yapay sinir ağı verilmiştir.⁴⁷



Şekil 1.17 Çok katmanlı bir yapay sinir ağı.

⁴⁷ Elmas, Age, s.44-45, Öztemel, Age, s. 52

1.9.1. Katmanlar

Genel olarak tüm yapay sinir ağlarının basit bir topolojik yapısı vardır. Sinirlerin bazıları gerçek dünyadan girişlerini alırken, bazıları da çıkışlarını dış dünyaya verirler. Diğerleri ise görünmediğinden ve dış dünyayla doğrudan bağlantısı olmadığından gizli katman olarak adlandırılır.

Şekil-1.17’de görüldüğü gibi sinirler katmanlarda toplanmışlardır. 3 çeşit katman vardır:

1. Giriş Katmanı
2. Ara (Gizli) Katman(lar)
3. Çıkış Katmanı

Giriş katmanı girişlerini dış dünyadan alan sinirlerin bulunduğu katmandır. Çıkış katmanı, ağınc çıkışını dış dünyaya ileten katmanlardır. Bu iki katman arasında genelde birkaç gizli (ara) katman vardır.

Ara katmanı olmayan yapay sinir ağları karmaşık işlevleri hesaplayabilme yeteneğine sahip değildir. Bu yüzden karmaşık hesaplamalar için ağınc en az bir tane ara katmanı olmak zorundadır.

Giriş katmanı ile dış dünyadan alınan girişler, giriş katmanı ile gizli katman arasındaki bağlantı ağırlıklarıyla çarpılıp gizli katmandaki sinirlere giriş olarak iletilir. En son gizli katmana gelen girişler yine, bu gizli katmanla çıkış katmanı arasındaki bağlantı ağırlıkları ile çarpıldıktan sonra tüm çıkış katmanındaki bu çarpımlar toplanarak bir çıkış üretilir.

Gizli sinirler, ağınc en iyi sonucu üretebilecek sayıda seçilmelidir. Sayı çok fazla olursa ağıncda genelleşme sorunu ortaya çıkar.

1.9.2. İletişim ve Bağlantı Çeşitleri

Bir sinirin çıkışı başka bir sinire giriş olarak yollarla bağlanır. Bu yollar normalde yönlü değildir. İki sinir arasında iki yönlü bağlantı vardır. Bir sinir birden çok hücreden giriş alır ancak diğer sinire sadece bir çıkış gönderir. Ayrıca bir katmandaki sinirler kendi aralarında da bağlantı kurabilir ya da hiç bağlantı olmayabilir.

Katmanlar arasında kullanılan deęişik türde bağlantılar vardır. Bu bağlantılara **Katmanlar Arası Bağlantı** denir. Çeşitleri aşağıda verilmiştir.

Tam Bağlantılı: Bir katmandaki her sinir kendinden sonra gelen katmandaki her sinire bağlıdır.

Kısmi Bağlantılı: Bir katmandaki bir sinirin sonraki katmandaki her sinirle bağlantılı olmama durumudur.

İleri Besleme: Bir katmandaki sinirler sonraki katmandaki sinirlere çıkışlarını gönderirler, fakat sonraki katmandaki sinirlerden bir giriş almazlar.

Çift Yönlü: Bir katmandaki sinirler sonraki katmandaki sinirlere çıkışlarını gönderirler, aynı zamanda sonraki katmandaki sinirlerin çıkışları da önceki katmandaki sinirlere taşınır.

İleri beslemeli ve çift yönlü bağlantılar tam veya kısmi bağlantılı olabilir.

Hiyerarşik: Bu bağlantı çeşidinde düşük katmandaki sinirler bir sonraki seviyedeki katmanın sinirlerine iletilebilir.

Rezonans: Çift yönlü bağlantılı katmanlar kesin durum oluşuncaya kadar girişleri ya da çıkışları bir katmandaki sinirden diğer katmandaki sinire defalarca göndermeye devam ediyorsa bu bağlantılar rezonans bağlantıdır.

Karmaşık yapıdaki yapay sinir ağlarında aynı katman içindeki sinirler kendi aralarında haberleşebilirler. Bu haberleşme **Sinirler Arası Bağlantı** ile sağlanır. İki çeşit sinirler arası bağlantı vardır.⁴⁸

1.Tekrarlamalı: Aynı katman içindeki sinirler tam veya kısmi bağlantılı olabilir. Bu katman başka bir katmandan giriş aldığında, çıkışları başka bir katmana göndermeden önce aynı katman içindeki sinirler arasında bağlantı meydana gelir ve bağlantılarla bir sinir çıkışı diğerine defalarca aktarılır.

2.Merkezde / Çevre Dışı (On Center / Off Surround): Aynı katmandaki iki sinir arasında uyarıcı, yasaklayıcı olmak üzere iki çeşit bağlantı vardır. Uyarıcı bağlantıda bir sinirin çıkışı bağlı bulunduğu sinirin faaliyet potansiyelini artırır,

⁴⁸ Elmas, Age, s.45-47, Öztemel, Age, s. 52

yasaklayıcı sinir ise faaliyet potansiyelini azaltır. Katmandaki sinirin kendisi ve komşuları arasında uyarıcı bağlantı kurulurken, diğer sinirler arasında yasaklayıcı bağlantı kurulur. Bu durum bir yarışma ortamı gibi düşünülebilir. Birkaç işaret değişiminden sonra etkin çıkış değerli sinir kazanır.

1.10. Yapay Sinir Ağları Donanımı

Her ne kadar yapay sinir ağlarında çoğu uygulama yazılım kullanarak yapılsa da, yapay sinir ağlarının paralel ve hızlı işlem yapmasını sağlayan yapay sinir ağları donanımlarıdır. Bu donanımları analog, sayısal ve karma olmak üzere üç grupta toplayabiliriz. Analog donanımlar, işaretleri sürekli değişen ve transistörün davranışına bağlı olarak çarpıcı veya toplayıcı olarak davranan çiplerdir. Sayısal olanlar ise, mantıksal işlemlere dayalı fonksiyonları kullanırlar. Analog ve sayısal donanımların birleşmesinden karma sistem oluşur.

Yapay sinir ağları için donanım geliştirilmesinin üstünlükleri;⁴⁹

- İşlem hızlarının yüksek olması,
- İşlem hassasiyetinin yüksek olması bu nedenle güvenilirliğin artması,
- Yüksek seviyeli gürültülerden etkilenmemeleri,
- Var olan sistemlerle tasarlanabilmeleri,
- Programlanabilir bileşenlere sahip olmaları,
- Çoğullama ve tekilleme kolaylığı sağlamaları,
- Sabit ve uyarlanabilir ağırlıkların korunabilmesi,
- Sistemin güvenliğinin daha rahat kontrol edilmesi,
- Probleme uygun tasarlandığı için sistemin boyutları ve ağırlığı konusunda en uygun seçimin yapılabilmesi

şeklinde sıralanabilir. Donanımın sınırlamaları ise;

- Geliştirilmesinin zor olması,
- Geliştirilmesinin uzun zaman alması,
- Isıl kararlılığının düşük olması,

⁴⁹ Sağıroğlu ve diğerleri, Age, s. 48, Öztemel, Age, s. 197-198.

- Gürültü oluşması,
- Analog bağlantılarda problemlerle karşılaşılması,
- Sınırlı doğruluğa sahip olması,
- Testinin zor oluşudur.

Yapay sinir ağları donanım çeşitleri şunlardır:⁵⁰

- Nörobilgisayarlar
- PC hızlandırıcılar ve kartları
- Yongalar (Chips)
- Gömülü mikroişlemciler

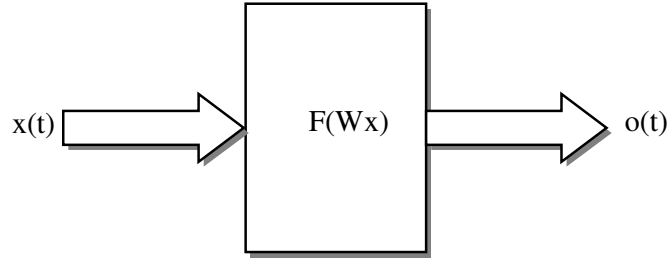
1.11. Yapay Sinir Ağlarının Sınıflandırılması-Yapılarına Göre Yapay Sinir Ağları

Yapay sinir ağları daha önce de bahsedildiği gibi birbirleri ile bağlantılı işlemcilerden (nöronlardan) oluşmuştur. Bu nöronlar arasındaki bağlantılar yapay sinir ağlarının yapılarını belirler. Bağlantıların nasıl olacağına öğrenme algoritmaları ile karar verilebilir. Yine öğrenme algoritmasına göre ağırlıkların değeri belirlenir. Genelde yapay sinir ağları yapılarına göre iki grupta incelenmektedir.

1.11.1. İleri Beslemeli Ağlar

İleri beslemeli ağlarda nöronlar katmanlarda bulunur. Girişler bir katmandan sonraki katmana tek yönlü bağlantı ile gönderilir. Bağlantı tek yönlü olduğu için sonraki katmandan alınan çıkışların geriye giriş olarak dönüşü söz konusu değildir. Şekil-1.18'de ileri beslemeli bir ağ blok yapıda gösterilmiştir. Bu ağlar Statik Ağlar olarak da isimlendirilirler.

⁵⁰ Öztemel, Age, s. 198.



Şekil 1.18 İleri beslemeli ağ.

İleri beslemeli ağlarda ilk katmandaki sinirler için x girişleri ile w ağırlık değerlerinin bir işlevi hesaplanır ve bu sonuç sonraki tüm komşu sinirlere iletilir. İlk gizli katman gibi ikinci gizli katman sinirleri de önceki katmana tam bağlıdır. Buradaki sinirler de aynı şekilde girişler ve ağırlık değerlerinin işlevini alarak sonraki sinirlere gönderirler. Bu işlem arada tüm ara katmanlarda ve çıkış katmanında yapıldıktan sonra tamamlanır. Bu ağlara “Çok Katmanlı İleri Beslemeli Ağlar” da denir.

Rosentblatt’ın perceptronu ortaya atmasından sonraki yıllarda, değişik çok katmanlı ileri beslemeli ağ yapısı üzerinde çalışılmıştır. Bu ağlar için geri yayılım algoritması uygulanabilir duruma gelmiştir.

İleri beslemeli ağlara örnek olarak sıkça kullanılan Çok Katmanlı Perseptron (Multilayered Perceptron-MLP) ve Öğrenme Vektör Nicelendirmesi (Learning Vector Quantization) ağları verilebilir.⁵¹

1.11.1.1. Çok Katmanlı Perseptron (MLP) Ağı

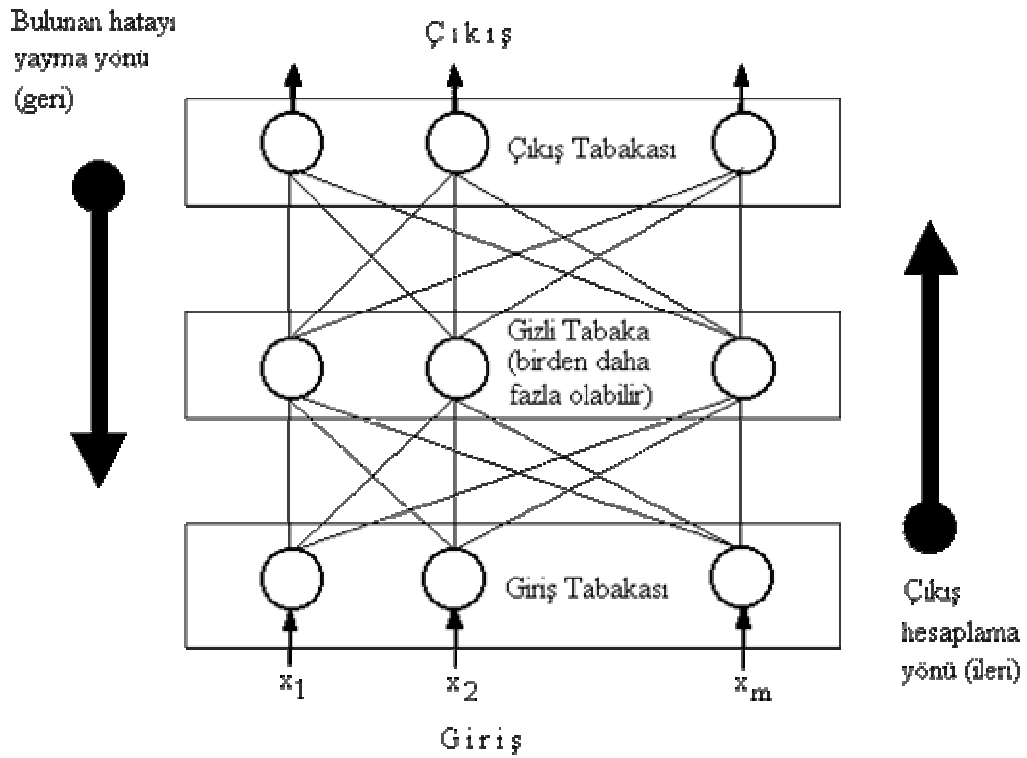
Bir yapay sinir ağından çözmesi istenilen problem doğrusal değilse ilk başta tasarlanmış yapay sinir ağları ile bunu modellemek mümkün değildir. Bu tür problemler için daha gelişmiş bir ağ türüne ihtiyaç duyulmuştur. Çok katmanlı perseptron da bu amaçla geliştirilen bir ağıdır. Bir çok katmanlı perseptron sinir ağı (MLP) modeli, Şekil-1.19’da gösterilmiştir. Birçok öğrenme algoritması bu ağı eğitmede kullanılabilir olması, bu modelin yaygın kullanılmasının sebebidir. Bir MLP modeli, bir giriş, bir veya daha fazla ara ve bir de çıkış katmanından oluşur. Her bir katmanda da bir veya daha fazla sayıda işlem elemanı bulunur. Bir katmandaki bütün işlem elemanları bir üst katmandaki bütün işlem elemanlarına bağlıdır. Bilgi akışı ileri doğru olup geri besleme

⁵¹ Elmas, age, s.62, Sağıroğlu ve diğerleri, Age, s. 42-43.

yoktur. Bunun için bu tip ağlar ileri beslemeli sinir ağı modeli olarak adlandırılır. Giriş katmanında herhangi bir bilgi işleme yapılmaz. Giriş ve çıkış katmanındaki işlemci elemanı sayısı tamamen uygulanan probleme bağlıdır. Ara katman sayısı ve ara katmanlardaki işlemci elemanı sayısı ise, deneme-yanılma yolu ile bulunur.

MLP ağlarında, ağı bir örnek gösterilir ve örnek neticesinde nasıl bir sonuç üreteceği de bildirilir (danışmanlı öğrenme). Örnekler giriş katmanına uygulanır, ara katmanlarda işlenir ve çıkış katmanından da çıkışlar elde edilir. Kullanılan öğrenme algoritmasına göre, ağın çıkışı ile arzu edilen çıkış arasındaki hata tekrar geriye doğru yayılarak hata minimumuna düşünceye kadar ağın ağırlıkları değiştirilir.

İleri beslemeli ağlar, en genel anlamıyla giriş uzayıyla çıkış uzayı arasında statik haritalama yapar. Bir andaki çıkış, sadece o andaki girişin bir fonksiyonudur.⁵²



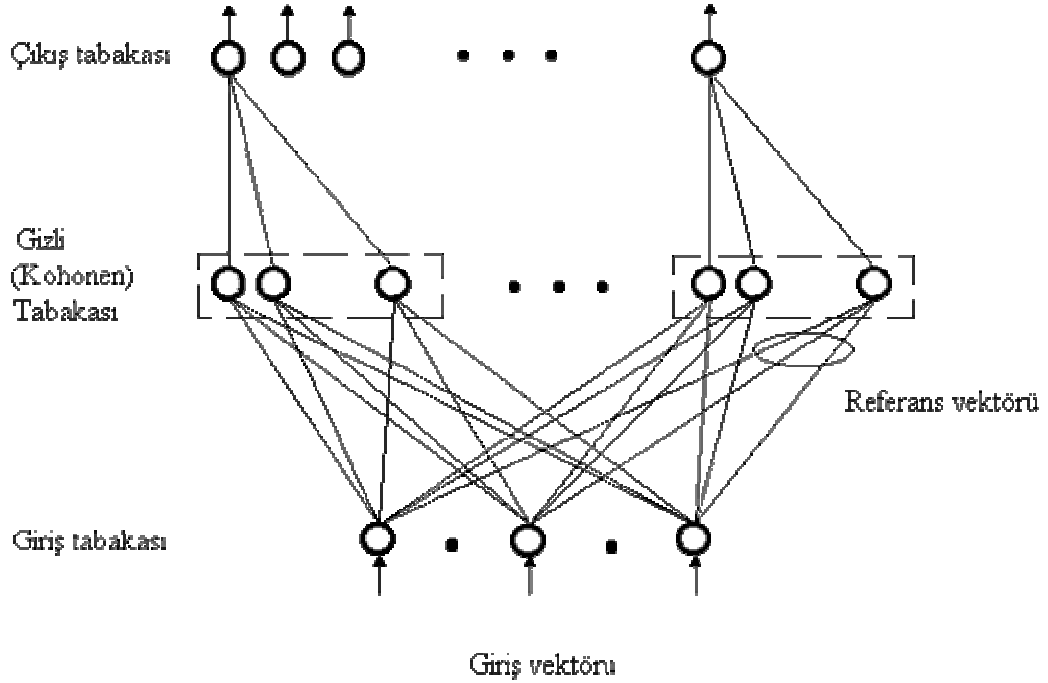
Şekil 1.19 MLP yapısı.

⁵² Sağiroğlu ve diğerleri, Age, s. 54-55, Richard Lippmann, "An Introduction to Computing with Neural Nets", 1987, Vol.4, s. 24-25.

1.11.1.2. Öğrenme Vektör Nicelendirmesi (Learning Vector Quantization) Ağı

Learning Vector Quantization (LVQ) ağı 1984 yılında Kohonen tarafından tasarlanmıştır. Bu nedenle Kohonen ağı olarak da bilinir. Şekil-1.20’de bir LVQ ağı verilmiştir. Şekil-1.20’de görüldüğü gibi bir giriş, bir çıkış ve bir Kohonen katmanından oluşur. Giriş katmanındaki nöronlar ara katmandaki tüm nöronlarla bağlıdır. Ancak ara katmandaki farklı sayıdaki nöron grupları çıkış katmanındaki her bir nörona bağlıdır. Bu ağın temel prensibi n boyutlu bir vektörü, bir vektörler setine haritalamaktır. Bu ağın öğrenme işlemi, girdi vektörünün hangi vektör seti tarafından temsil edileceğinin bulunmasını sağlamaktadır.

LVQ ağları özellikle sınıflandırma problemleri olmak üzere, sınıflandırma ve görüntü işleme problemlerinin çözümünde kullanılır. Bu ağ için kullanılan öğrenme takviyeli öğrenmedir.⁵³



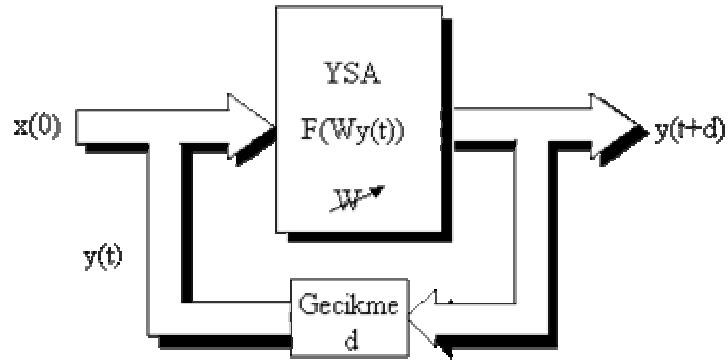
Şekil 1.20 LVQ ağı.

⁵³ Sağiroğlu ve diğerleri, Age, s. 58-59.

1.11.2. Geri Beslemeli Ağlar

Geri beslemeli ağlarda da işlem elemanları yani nöronlar ileri beslemeli ağlarda olduğu gibi katmanlarda bulunur. Ancak katmanlar arası bağlantılar tek yönlü değil çift yönlüdür. Bağlantıların çift yönlü olması ara katmanların ya da çıkış katmanının çıkışlarının bir önceki katmana giriş olarak gönderilebileceği anlamına gelir. Bir geri beslemeli ağ tipi Şekil-1.21’de verilmiştir. Bu ağlar Dinamik Ağlar olarak da adlandırılabilir. Dinamik bir hafızaya sahiptirler. Bir andaki çıkış hem o andaki giriş hem de bir önceki giriş olarak kullanılır. Bu nedenle bu tür ağlar tahmin uygulamaları için kullanılır.

Geri dönüşümlü ağlarda, ağın işlem elemanlarının çıkışları tekrar ağa belirli işlemler sonucunda giriş olarak gönderilir. Dinamik sistemlerde geri dönüşümlerden kaynaklanan zaman gecikmesi önemli olduğu için bu konuda bir takım çalışmalar yapılarak değişik geri dönüşümlü ağ modelleri tasarlanmıştır.⁵⁴



Şekil 1.21 Geri beslemeli ağ.

Geri dönüşümlü ağlar iki çeşittir:⁵⁵

Tam geri dönüşümlü ağlar: Bu ağlar rasgele ileri ve geri bağlantılara sahip olan ağlardır. Bu bağlantıların hepsi eğitilebilir.

Kısmi geri dönüşümlü ağlar: Bu ağlarda ağın işlem elemanlarına ek olarak, içerik elemanları da vardır. Bu ağlar, temelde ileri beslemeli ağlardır. İleri bağlantılar eğitilebilir. Geri dönüşüm olayı sadece içerik elemanları tarafından yapılır ve bağlantılar eğitilemezler. İçerik elemanları ara katman elemanlarının geçmiş durumlarını

⁵⁴ Sağiroğlu ve diğerleri, Age, s. 42-43.

⁵⁵ Öztemel, Age, s. 165.

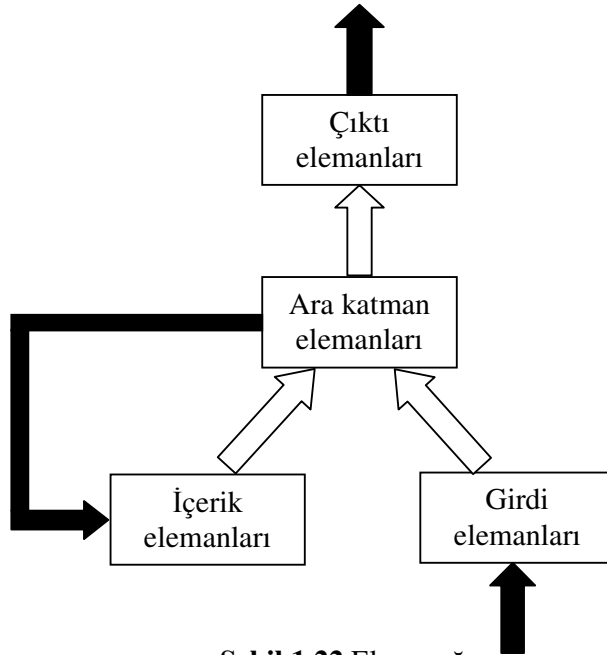
hatırlamak için kullanılırlar. Ađın ıktısı hem nceki durumu hem de ađın o andaki durumuna bađlı olarak oluřturulmaktadır. Gemiř durumları hatırlayabilmeleri bu ađlara dinamik hafızaya sahip olma zelliđi kazandırmaktadır.

Geride dnüşümlü ađlara rnek olarak Elman, Hopfield, Uyarlanır Rezonans (ART), zrgütlemeleli Harita (SOM) Ađları verilebilir. Bu ađlardan en basit yapıya sahip olan ve kullanılması en kolay ađ Elman ađıdır.

1.11.2.1. Elman Ađı

Elman ađı daha nce anlatılan ok katmanlı algılayıcı ađının đrenme kuralına gre đrenmektedir. Elman ađının yapısı Őekil-1.22'de gsterilmiřtir. Őekil-1.22'de grüldüđu gibi Elman ađının 4 eřit iřlemci elemanları vardır:

- Girdi elemanları,
- Ara katman elemanları,
- ıktı elemanları,
- İerik elemanları.



Őekil 1.22 Elman ađ yapısı.

Bunlardan giriş ve ıkıř elemanları dıř dnya ile etkileřim halindedir. Giriř elemanları dıř dnyadan bilgileri alır ara katmanlara iletirler. ok katmanlı algılayıcılarda olduđu gibi, Elman ađında da giriş elemanlarının bilgi iřleme zelliđi

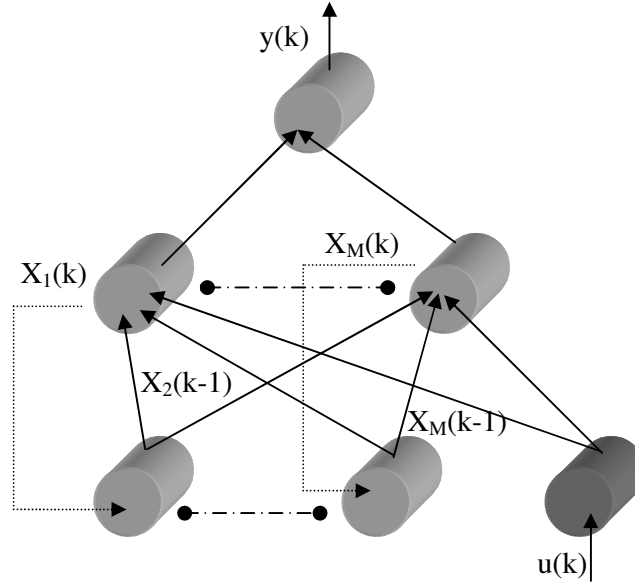
yoktur. Girişleri olduğu gibi ara katman elemanlarına gönderirler. Çıkış elemanları ise, ağın çıkışını dış dünyaya iletir. Çıkış birimlerinin bilgi işleme fonksiyonları doğrusaldır. Sadece kendilerine gelen bilgileri toplarlar. Ara katman elemanları ise, hem doğrusal hem doğrusal olmayan aktivasyon fonksiyonlarına sahip olabilirler. İçerik elemanları ara katman elemanlarının önceki aktivite değerlerini hatırlamak için kullanılmaktadır. Bu elemanlar bir adım gecikmeyi (one step time delay) içermektedirler. Bir önceki iterasyondaki aktivasyon değerlerini bir sonraki iterasyona giriş olarak taşırlar. Şekildeki ileri beslemeli bağlantıların ağırlıkları (boş oklar) eğitim sırasında değiştirilebilirler. Geri dönüşümlerin (içerik elemanlarının) bağlantı ağırlıkları ise (dolu oklar) sabittir. Bu ağırlıklar değiştirilemezler. Geri dönüşümlerin ağırlıkları sabit olduklarından Elman ağına “Kısmi Geri Dönüşümlü Ağ” da denilebilir.

Elman ağındaki, herhangi t. zamanında hem t. zamanındaki giriş değerleri, hem de ara katmanların (t-1). zamandaki (önceki) aktivite değerleri ağa giriş olarak verilirler. Ağın girişleri belirlendikten sonra ağ artık ileri beslemeli bir çok katmanlı algılayıcıya dönüşmektedir. Bu girdiler kullanılarak ileri doğru ağın çıktıları belirlenir. Bunu belirlemek için çok katmanlı algılayıcıların kullandıkları öğrenme kuralı kullanılmaktadır. Bu ileri doğru hesaplamadan sonra ağın ara katmanlarının aktivasyon değerleri geriye doğru içerik elemanlarına girdi olarak ve orada bir sonraki iterasyonda kullanılmak üzere saklanır.

Başlangıçta ara katmanların aktivasyon değerleri bilinmediğinden içerik elemanlarının başlangıç değerlerinin belirlenmesi gerekir. Bunun için, genel olarak bir ara katmanın alabileceği maksimum değer yarısı içerik elemanlarının başlangıç girdi değerleri olarak atanır. Eğer sigmoid fonksiyonu kullanılacak ise genellikle bu elemanlar başlangıçta giriş değeri olarak 0,5 değerini alırlar. Hiperbolik Tanjant fonksiyonu için ise 0 değeri başlangıç değeri olarak atanmaktadırlar.

Şekil-1.23’de bir giriş, M ara katman elemanı ve bir çıkıştan oluşan Elman Ağı’nın daha ayrıntılı bir şekilde gösterimi verilmiştir. Şekildeki $u(k)$, k. zaman dilimindeki dış dünyadan gelen girdiyi; $y(k)$, k. zaman diliminde üretilen çıkışı; $x(k)$ ise, k.zaman dilimindeki ara katman elemanlarının çıkışlarını; $x(k-1)$ ise bir önceki zaman diliminde ara katman elemanlarının çıkışlarını göstermektedirler.⁵⁶

⁵⁶ Sağıroğlu ve diğerleri, Age, s. 62, Öztemel, Age, s. 166-167.



Şekil 1.23 Elman ağıının ayrıntılı gösterimi.

1.11.2.2. Hopfield Ağı

Hopfield ağı, John Hopfield tarafından 1982 yılında geliştirilmiştir. En çok optimizasyon problemleri için kullanılır. Hopfield ağı tek katmanlı ve geri dönüşümlü bir ağıdır. Şekil-1.24'de bir Hopfield ağı şekli gösterilmiştir. Bu ağı genellikle ikili (0 veya 1) ve bipolar (+1 veya -1) girişler kabul eder. Tek katmandaki işlemci elemanlarının her biri bir diğerine bağlanmıştır. Ayrıca her hücrenin çıkışı, bir ağırlıkla işleme tabi tutularak diğer hücrelere giriş değeri olarak dönmektedir. Ağı bağlantı değerleri bir enerji fonksiyonu olarak saklanır. Hopfield ağıının eğitilmesi sadece bir adım alır ve w_{ij} ağırlıkları aşağıdaki gibi hesaplanır:

$$w_{ij} = \begin{cases} \frac{1}{N} \sum_{c=1}^p x_i^c x_j^c, & i \neq j \\ 0, & i = j \end{cases} \quad (1.12)$$

Burada;

w_{ij} : i işlemci elemanından j işlemci elemanına olan bağlantının ağırlığını,

x_i^c : c sınıfı için eğitime giriş deseninin i'inci elemanını,

P: sınıf sayısını,

N: işlemci eleman sayısını gösterir.

Denklem 1.12'de $w_{ij} = w_{ji}$ ve $w_{ii} = 0$ durumları ağırlıkların kararlılığını garantiler.

Bilinmeyen bir bilgi ağına girildiğinde, ağırlıkların çıkışları bilinmeyen bir desen elemanlarına eşitlenir, yani;

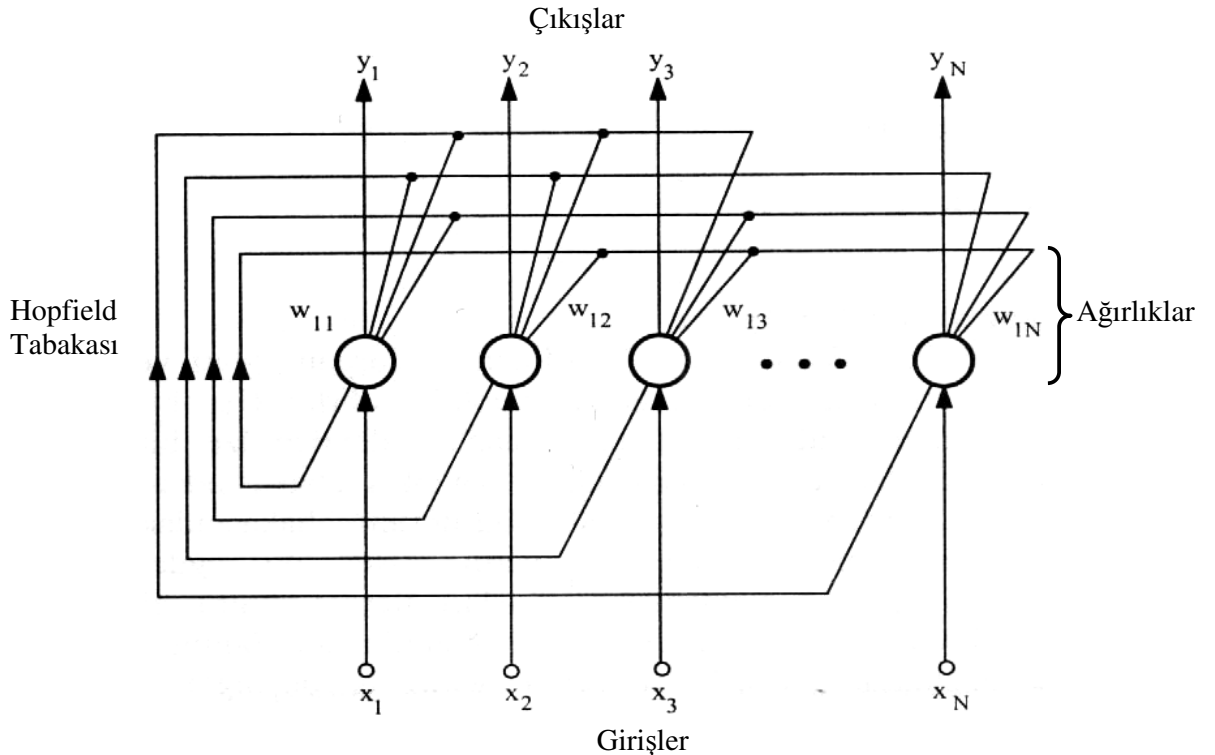
$$y_i(0) = x_i \quad 1 \leq i \leq N \quad (1.13)$$

Bu başlangıç değerleri ile başlayarak Hopfield ağı bir sonraki denklemi kullanarak minimum enerji durumuna geçmek için döngüye girer.

$$y_i(k+1) = f \left[\sum_{j=1}^N w_{ij} y_j(k) \right] \quad 1 \leq i \leq N \quad (1.14)$$

burada f aşağıdaki gibi tanımlanan keskin sınırlama (hard limiting) fonksiyonudur.

$$f(x) = \begin{cases} -1 & x < 0 \\ +1 & x > 0 \end{cases} \quad (1.15)$$



Şekil 1.24 Hopfield ağı.

Günümüze kadar iki tür Hopfield ağı geliştirilmiştir. Bunlar:⁵⁷

Kesikli (Discrete) Hopfield Ağı: Hopfield ağının bu türü çağrışumlu bellek olarak kullanılır. Aktivasyon fonksiyonu olarak signum fonksiyonu kullanılır. Hücre +1 veya -1 değerini alabilir.

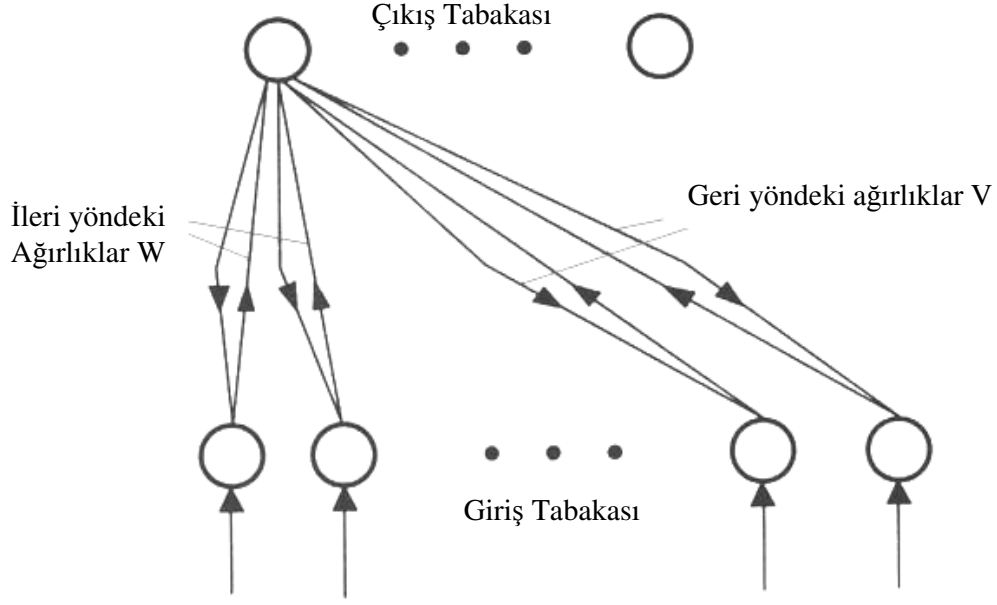
Sürekli (Continuous) Hopfield Ağı: Bu ağlar da kombinatoriyel optimizasyon problemlerinin çözümünde kullanılır. Kesikli ağdan farklı olarak signum fonksiyonu yerine sigmoid fonksiyonu kullanılır. Ağın çıkış değerleri 0-1 arasında sürekli değerler olabilir.

1.11.2.3. Uyarlanırlı Rezonans (ART) Ağı

ART ağları 1976 yılında Grossberg tarafından geliştirilmiştir. Birçok farklı ART ağı vardır. Şekil-1.25’de bir ART-1 ağı verilmiştir. ART-1 ağının giriş ve çıkış olmak üzere iki katmanı vardır. İki katman birbirleriyle tamamen bağlantılıdır ve bağlantılar ileri ve geri yöndedir. i 'nci çıkış işlemci elemanının ileri yöndeki bağlantılarının ağırlıklarının oluşturduğu W vektörü, temsil ettiği sınıfın bir örneğini oluşturur. İleri bağlantılarının W_i vektörlerinin toplamı ART ağının uzun-dönem (long-term) hafızasını oluşturur. Bu vektörler, giriş desenine benzerlikleri ölçüsünde, çıkış işlemci elemanlarını belirlerler. Çıkış işlemci elemanlarının geri bağlantılarının V_i vektörleri ise, giriş deseninin, hafızadaki desenine yeterince benzeyip benzemediğini (o sınıfa ait olup olmadığını) kesin olarak belirlemek amacıyla kullanılır. Bu test vektörleri diyebileceğimiz V_i vektörleri ART ağının kısa-dönem (short-term) hafızasını oluştururlar. W_i , V_i 'nin normalize edilmiş halidir. Bu ağlar belli giriş verisini analiz eder, olası özellikleri kontrol eder veya giriş vektörü içindeki bu özellikleri sınıflandırır. Birçok ağın temelini oluşturur. Özöğütlemeli ağlara iyi bir örnektir. ART ağları eğitmek için danışmansız öğrenme yöntemi kullanılır.⁵⁸

⁵⁷ Sağiroğlu ve diğerleri, Age, s. 61-62, Öztemel, Age, s. 170-173.

⁵⁸ Sağiroğlu ve diğerleri, Age, s. 65-66, Phil Picton, *Intoduction to Neural Networks*, London: Macmillan Pres, 1994, s. 102.



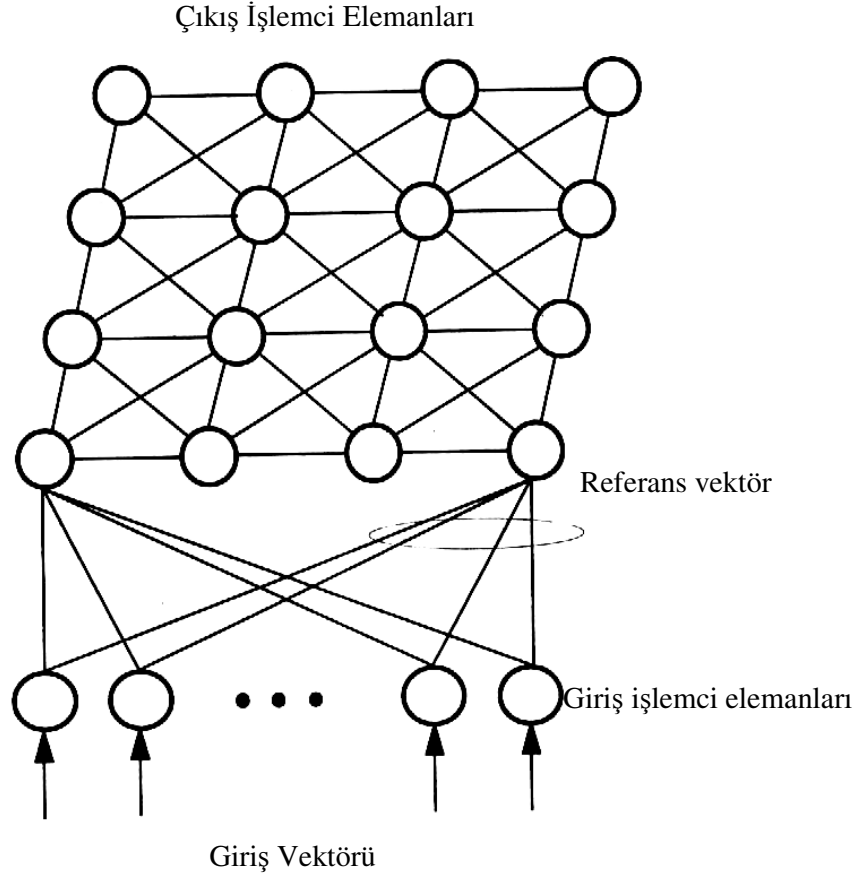
Şekil 1.25 ART-1 ağı.

1.11.2.4. Som Ağı

SOM tipi ağlar Kohonen tarafından geliştirilmiştir. Şekil-1.26'da bir SOM ağı gösterilmiştir.⁵⁹ Bu ağlar bir giriş ve bir çıkış katmanından oluşur. Ancak diğer ağlardan farklı olarak çıkış katmanı 2 boyutlu bir düzlemdir. Som tipi ağlarda yarışmayı kazanan işlemci elemanı 1 değerini alırken, diğer elemanlar 0 değerini alır. Sadece 1 değerini alan işlemci ve komşularının ağırlıkları değişir. Genellikle sınıflandırma amaçlı kullanılırlar. Bu ağlar öğrenmek için öğretmene ihtiyaç duymazlar. Yani çıktıların bilinmesi zorunlu değildir. Çıktıları belli olmayan problemlerin çözümü için uygundur.⁶⁰

⁵⁹ Sağiroğlu ve diğerleri, Age, s. 64.

⁶⁰ Öztemel, Age, s. 180-181.



Şekil 1.26 SOM ağı.

1.12. Yapay Sinir Ağlarının Kronolojik Sınıflandırması⁶¹

1. Perceptronlar
2. Hopfield Ağı
3. Uyarlanır Rezonans Ağı
4. Kohonen Harita Ağı
5. Geriyayılım Ağı
6. Counterpropagation Ağı

⁶¹ Simpson, Age, s. 6.

BÖLÜM 2

YAPAY SINİR AĞLARINDA ÖĞRENME

Yapay sinir ağlarının en ayırt edici ve önemli özelliği öğrenme yeteneğine sahip olmasıdır.

Yapay sinir ağlarından iyi bir sonuç etmek için ağ uyarlanabilir olmalıdır. Bu ise ancak uygun değerli ağırlıklar ve doğru bağlantılarla mümkündür. Uygun ağırlıklar ve bağlantıların elde edilebilmesi için sistemin davranışlarını öğrenmeli ya da kendi kendini örgütlemelidir. Öğrenme yapay sinir ağının çok önemli bir parçasıdır.

Yapay sinir ağlarında bilgi, ağdaki sinirlerin bağlantılarının ağırlıklarında tutulur. Bu nedenle ağırlıkların nasıl belirleneceği önemlidir. Bilgi tüm ağda saklandığı için bir düğümün sahip olduğu ağırlık değeri tek başına bir şey ifade etmez. Tüm ağdaki ağırlıklar uygun değerler almalıdır. Ancak yapay sinir ağlarında başlangıçta en uygun ağırlıkların değerlerinin bulunması için geliştirilmiş bir formül yoktur. Ağdaki işlemci elemanlar zamanla bir takım kurallar yardımıyla en uygun ağırlık değerlerini belirlerler. Bu işleme “ağın eğitilmesi” denir. Buna göre bir ağın eğitilebilir olabilmesi için ağırlık değerlerinin belirli bir kural dâhilinde dinamik olarak değiştirilebilir olması gerekmektedir.⁶²

Ağırlıkların belirlenmesi işlemi sadece mevcut ağırlıkların değiştirilmesini içermez. Aynı zamanda;

- Yeni bağlantıların yapılması,
- Bazı bağlantı ağırlıklarının yok edilmesi

gibi durumlar da söz konusu olabilir.

Yapay sinir ağlarında genel olarak öğrenme olayı iki aşamada gerçekleşir. Birinci aşamada ağırlık değerleri rasgele alınır ve ağa gösterilecek olan örnek için ağın üreteceği çıkış bulunur. Bu çıkış değerinin doğruluğuna göre ikinci aşama olarak da ağa farklı örnekler gösterilerek ağırlık değerleri değiştirilir. Buradaki amaç bu örnekler için doğru çıkışı elde etmeye yarayacak olan ağırlık değerlerini bulmaktır. Bu aşamada örneklerin seçimi önemlidir. Seçim yapılırken örnek eğitim kümesi ne kadar büyük olursa o kadar iyi olur şeklinde bir yanlış kanaat mevcuttur. Oysa bu küme en az bilgi

⁶² Öztemel, Age, s. 55, Şen, Age, s.90.

ile öğrenmeyi sağlayacak şekilde seçilmelidir. Eğitim kümesi oluşturulurken birbirine yakın bilgiler yerine farklı ve bağımsız bilgilerin seçilmesi daha iyi sonuçlar verir. Bu amaçla örnekler ağa tekrar tekrar gösterilir. Ağın doğru çıktığı üretecek ağırlık değerlerini bulması, ağın örneklerin temsil ettiği olaylar hakkında genelleme yapabilme yeteneğine sahip olduğunu gösterir. Ağın genelleme yapabilmesi olayına ise “ağın öğrenmesi” denir. Ağırlıkların değerlerinin değişmesi belli kurallara göre yapılmaktadır. Bu kurallar “öğrenme kuralları” olarak adlandırılır.

McCulluch ve Pitts’in sinir ağlarının temel problemi olan öğrenmenin nasıl yapılacağına, Donald Hebb 1949 yılında yayınlanmış olan “The Organization of Behavior” adlı kitabında açıklık getirmiştir. Bu kitapta Hebb kuralı adındaki öğrenme kuralını anlatmıştır. Bu kurala göre sinirler arasındaki ağırlık değerleri, sinirlerin aynı zamanda etkinleşmesine bağlıdır. Hebb, sinirlerin hafızada basit bir yerde yerleşebileceğini varsayarak, birbirlerini uyardıkları ve bunun sonucunda aralarındaki ağırlık değerlerinin kendi etkinlikleri çarpımı oranında artacağını göstermiştir.

Hebb’den sonra birçok araştırmacı, Hebb kurallarını esas alarak öğrenme değişkenleri ve ağırlık değerlerinin belirlenmesi konusunda çalışmalar yapmışlardır ve bir takım öğrenme kuralı geliştirmişlerdir. Temelde öğrenme yöntemleri, danışmanlı (supervised) ve danışmansız (unsupervised) olmak üzere iki grupta toplanmıştır.⁶³

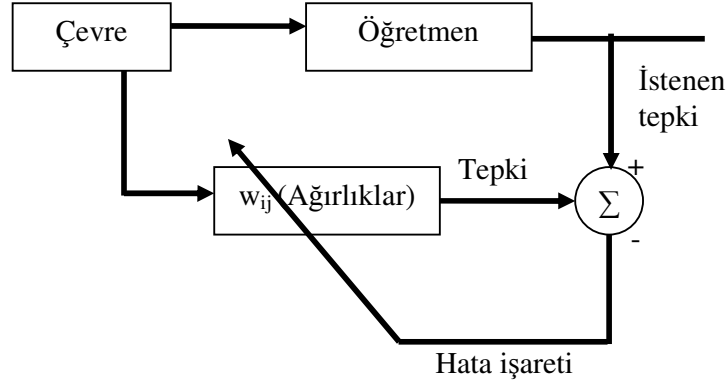
2.1 Temel Öğrenme Kuralları

2.1.1 Danışmanlı Öğrenme

Danışmanlı öğrenmede, yapay sinir ağı kullanılmadan önce eğitilmesi gerekir. Eğitim işlemi için giriş bilgilerinin yanında çıkış bilgileri de verilmelidir. Çoğu uygulama için ağa gerçek örnek kümesi verilme zorunluluğu vardır. Bu örnek kümesi ile ağ eğitilip istenen istatistiksel doğruluk elde edildiğinde eğitim işlemi tamamlanmış olur. Ağ kullanılmaya başladığında eğitim sonucunda elde edilen ağırlık değerleri çoğunlukla sabit kalır, bir daha değiştirilmez. Ancak bazı ağ modellerinde kullanım

⁶³ Elmas, Age, s.95, Öztemel, Age, s. 55, Şen, Age, s.90.

aşamasında da az da olsa eğitime devam edilir. Buradaki amaç ağıın değişen koşullara uyum göstermesini sağlamaktır. Şekil-2.1’de bir danışmanlı öğrenme gösterilmiştir.⁶⁴



Şekil 2.1 Danışmanlı öğrenme.

Danışmanlı öğrenmede eğitim kümesi için çıkışlar bilindiğine göre, sanki bir öğretmen çıkışları olması gerekenle karşılaştırarak kabul edilip edilmeyeceğine karar verir. Elde edilen çıkışlarla aynı girişler için önceden bilinen çıkışlar arasındaki farklar hata olarak kabul edilir. Bu hataların karelerinin en küçük olacak şekilde sınırlar arası bağlantılardaki ağırlık değerleri yeniden düzenlenir ve en küçük hata ile çıkışa yaklaşılr. Bu öğrenme türünde girişlerden çıkışlara doğru ileri bağlantı ve hatanın istenilen değerden büyük olması durumunda çıkışlardan girişlere doğru bir geri besleme olacaktır. Bulunan hata değeri seçilecek olan hata değerinden küçük olması durumunda yapay sinir ağı eğitilmiş demektir. Bundan sonra ağıın verilecek olan bilinmeyen çıkışa sahip girişler için en küçük hata ile çıkış tahmininde bulunması beklenir.

Bu öğrenme yönteminde öğrenmeye dışarıdan müdahale eden bir öğretmen vardır. Öğrenme eğitici kontrolündedir. Öğretmen, eğitim kümesini ve hata değerini belirleyerek eğitimin ne kadar devam edeceğine karar verir. Bu yöntemin en önemli özelliği eğitim esnasında gerçek giriş, çıkış değerlerin kullanılmasıdır. Danışmanlı öğrenme kuralları şunlardır:⁶⁵

- 1) Perceptron Öğrenme Kuralı
- 2) Delta Öğrenme Kuralı

⁶⁴ Elmas, Age, s.96-97.

⁶⁵ Şen, Age, s.98-99, Elmas, Age, s.96.

- 3) Genişletilmiş Delta Öğrenme Kuralı
- 4) Geri Yayılımlı Öğrenme Kuralı

2.1.1.1. Geri Yayılımlı Öğrenme Kuralı

Geri yayılımlı öğrenme kuralını kullanan ağlar hiyerarşik yapıdadır. Giriş, çıkış ve en az bir tane gizli katmandan oluşurlar. Geri yayılım öğrenme kuralı çok katmanlı ağlarda kullanılan delta kuralının genelleştirilmiş halidir. Geri yayılım ağında hatalar, ileri besleme aktarım işlevinin türevi tarafından, ileri beslemeli ağlar için kullanılan bağlantılar yardımıyla geriye doğru yayılırlar. Öğrenme işlemi, basit çift yönlü hafıza birleştirmeye dayanır.⁶⁶

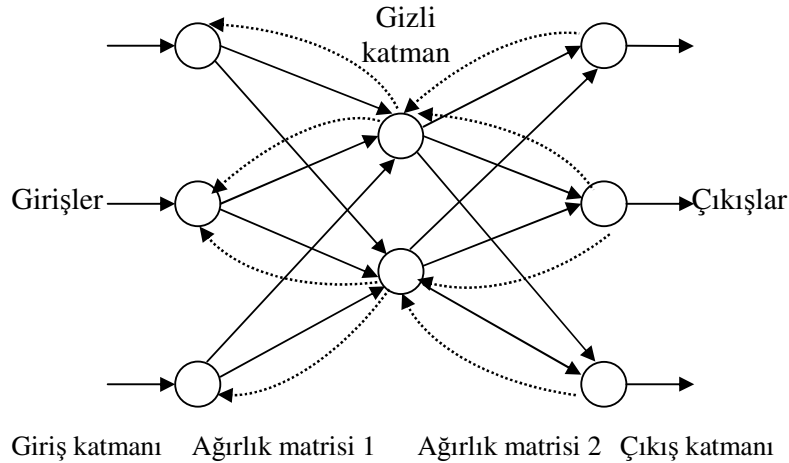
Geri yayılımlı öğrenme yöntemi, türevi alınabilir etkinlik işlevi kullanılarak herhangi bir ağa uygulanabilir. Bu yöntemde geriye doğru ağırlık ayarlamaları yapıldığı için “geri yayılım” olarak isimlendirilmiştir.

2.1.1.1.1. Geri Yayılım Algoritması

Geri yayılım ağı, Geoffrey Hinton ve James McClelland tarafından geliştirilmiştir. Giriş, çıkış ve en az bir gizli katmandan oluşur. Gizli katmandaki düğüm sayısı değişebilir. Düğüm sayısı artarsa ağın hatırlama yeteneği artar, ancak öğrenme süresi uzar. Düğüm sayısı az olursa öğrenme süresi kısalmış olur ancak ağın hatırlama yeteneği azalır. Bir katmandaki her düğüm kendinden sonraki katmandaki her düğüme bağlıdır. Fakat aynı katman içindeki hiçbir düğüm diğerine bağlı değildir. Yine her katmandaki düğümlerin çıkışları kendinden önceki katmana giriş olarak döner. Şekil-2.2’de bir geri yayılım ağı verilmiştir.⁶⁷

⁶⁶ Kishan Mehrota, Chilukuri K. Mohan ve Sanjay Ranka, *Elements of Artificial Neural Networks*, Londond: Massachusetts Institute of Tecnology, 1997, s. 70.

⁶⁷ Stamatios V. Kartalopoulos, *Understanding Neural Networks and Fuzzy Logic*, NewYork: The Institute of Electrical and Electronics Engineers Press, s. 78-80.



Şekil 2.2 Geri yayılım ağı.

Geril yayılım algoritması ağın gerçek çıkışları ile elde etmek istediğimiz çıkışlar arasındaki hataların karelerinin toplamının en aza indirilmesi ile ağın ağırlıklarının yeniden düzenlenmesi olarak genelleştirilebilir. Geril yayılım, girişlerin yanında istenen çıkışların da bilindiği ağlarda danışmanlı öğrenme kuralı olarak kullanılır. Önce öğrenmede, ağ ağırlıkları rasgele seçilir ve yeni girişler için çıkış değerlerinin hesaplanması amacıyla kullanılır. İleri besleme işlemi giriş katmanına gelen verilerin ilk gizli katmana giriş olarak aktarılmasında geçerlidir.

Geril yayılım öğrenme yöntemi, türevi alınabilir özelliğe sahip olan etkinlik fonksiyonunu çok katmanlı herhangi bir ağa uygulayabilir. En çok kullanılan algoritmadır. Bu algoritma ile çok katmanlı ağlarda öğrenme olayını gerçekleştirilir. Bu sayede önceden başarı elde edilememiş olan giriş-çıkış tasvirleri, sınıflandırma ve genelleme gibi problemlere daha iyi çözümler getirilmesi sağlanmıştır. Bu ağda iki türlü bağlantı vardır. Birinci ileri yönlü bağlantıdır ve ağırlıklar kullanılarak girişlere karşılık çıkışlar elde edilir. İkinci bağlantı geriye doğrudur ve elde edilen bu çıkışlarla beklenen çıkışlar arasındaki hata değerinin geriye yayılmasını sağlar. Bu işlem tüm katmanlar için yapılır. Tüm ağlarda olduğu gibi burada da amaç en iyi çıkışa ulaşabilmek için en uygun ağırlık değerlerini belirlemektir. Toplam hata en aza indirilmeye kadar bu işleme devam edilir.⁶⁸

⁶⁸ Mehrota, Mohan ve Ranka, Age, s. 70, Elmas, Age, s. 124-125.

Geriyayılım ağı algoritmasının akış diyagramı Şekil-2.4'de verilmiştir.⁶⁹

Bu geriyayılım öğrenme algoritması için aşağıdaki parametreler kullanılacaktır.

l katmanlar ve N_l l katmanındaki düğümler,

w_{ij}^l = l . katmandaki j . düğümlerle $l+1$. katmandaki i . düğüm arasındaki ağırlık değeri,

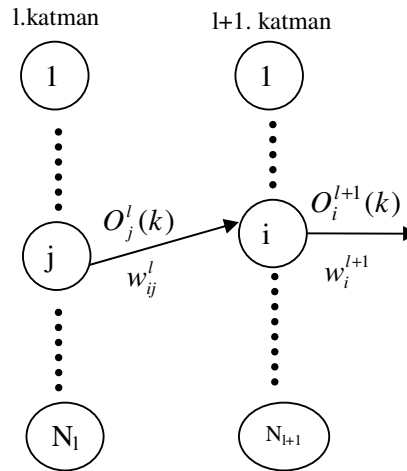
$O_j^l(x_p)$ = gerçek çıkış (l . katmandaki j . düğüme gelen x_p giriş değerleri için),

$T_{l,j}(x_p)$ = beklenen ya da hedeflenen çıkış (l . katmandaki j . düğüme gelen x_p giriş değerleri için),

$a_{l,j}(x_p)$ = aktivasyon fonksiyonunun çıkışı (l . katmandaki j . düğüme gelen x_p giriş değerleri için) bir önceki,

P = eğitim örnekleri ve $x_p=p$. eğitim örneğidir.

Şekil-2.3'de w_{ij}^l ile gösterilen ağırlık değeri $l+1$. katmandaki i . düğüme, l . katmandaki j . düğümden alınan işarettir.



Şekil 2.3 Bir geri yayılım ağı düğümündeki öğrenme.

⁶⁹ Sağıroğlu ve diğerleri, Age, s. 80.

l+1. katmandaki i düğümdeki çıkış değeri $O_j^{l+1}(k)$ aşağıdaki gibi hesaplanır.⁷⁰

$$\begin{aligned} O_j^{l+1}(k) &= f\left(\sum_{j=1}^{N_l} w_{ij}^l O_j^l(k) - \theta_i^{l+1}\right) \\ &= f\left(\sum_{j=1}^{N_l+1} w_{ij}^l O_j^l(k)\right), \end{aligned}$$

Eşik değeri θ_i^{l+1}

Aktivasyon fonksiyonu olarak $f(x) = 1/(1 + \exp(-\beta x))$ sigmoid kullanıldığında, onun türevi;

$$f'(x) = \beta f(x)(1 - f(x))$$

Ağa verilen tüm K örnek için toplam hata E, l. katmandaki beklenen çıkış ile gerçek çıkış arasındaki farkların kareleri toplamı olarak tanımlanabilir.

$$E = \sum_{k=1}^K E_k = \sum_{k=1}^K \left(\frac{1}{2} \sum_{i=1}^{N_L} [T_i(k) - O_i^L(k)]^2 \right)$$

Amaç tüm ağ için hatayı en aza indirecek ağırlık değerlerini hesaplamaktır. Yeni ağırlık değeri hatanın ağırlık değerine oranının türevine yaklaşık bir değerdir.

$$\Delta w_{nm}^l \approx - \frac{\partial E_k}{\partial w_{nm}^l}$$

l. katmandaki nm. nöronun ağırlığının E_k hatasına oranı aşağıdaki gibi hesaplanır.

$$\frac{\partial E_k}{\partial w_{nm}^l} = \frac{\partial E_k}{\partial O_i^L(k)} \frac{\partial O_i^L(k)}{\partial w_{nm}^l}$$

O zaman;

⁷⁰ Kartalopoulos, Stamatios V., Understanding Neural Networks and Fuzzy Logic, s. 78-80.

$$-\frac{\partial E_k}{\partial w_{nm}^l} = \sum_{i=1}^{N_L} (T_i(k) - O_j^L(k)) \frac{\partial O_i^L(k)}{\partial w_{nm}^l}$$

sigmoid fonksiyonu ve onun türevi arasındaki ilişki aşağıdaki gibidir. $l=L-1$ için;

$$-\frac{\partial E_k}{\partial w_{nm}^l} = (T_n - O_n^L) \beta O_n^L (1 - O_n^L) O_m^{L-1}$$

çıkış katmanının ağırlıklarını ayarlamak için yapılan işlem;

$$\Delta w_{nm}^L = \eta [(T_n - O_n^L) O_n^L (1 - O_n^L)] O_m^{L-1}$$

şekindedir. Burada η öğrenme katsayısı olarak bilinen faktördür.

Bunun yanında $l \neq L-1$ olduğunda O_m^{L-1} hala w_{nm}^l ' e bağlıdır ve hatanın ağırlıklara oranı aşağıdaki gibi hesaplanır.

$$-\frac{\partial E_k}{\partial w_{nm}^l} = \sum_{i=1}^{N_L} (T_i - O_i^L) f'(O_i^L) \sum_{j=1}^{N_{L-1}+1} w_{ij}^{L-1} \frac{\partial O_j^{L-1}(k)}{\partial w_{nm}^l}$$

Eğer $l = L-2$ ise son gizli katman ağırlıkları şöyle hesaplanır:

$$-\frac{\partial E_k}{\partial w_{nm}^l} = \sum_{i=1}^l (T_i - O_i^L) f'(O_i^L) w_{in}^{L-1} f'(O_n^{L-1}) O_m^{L-2}$$

$$= f'(O_m^{L-1}) \left[\sum_{i=1}^N (T_i - O_i^L) f'(O_i^L) w_{in}^L \right] O_m^{L-2}$$

sonuç olarak, son gizli katmanın ağırlıklarını ayarlamak için yapılan işlem;

$$\Delta w_{nm}^{L-2} = \eta [f'(O_n^{L-1}) \sum_{i=1}^{N_L} (T_i - O_i^L) f'(O_i^L) w_{in}^L] O_{in}^{L-2}$$

Bu ifade şöyle özetlenir;

$$\Delta w_{ij}^l = \eta \delta_i^l O_j^{l-1}$$

burada çıkış katmanı ağırlıkları,

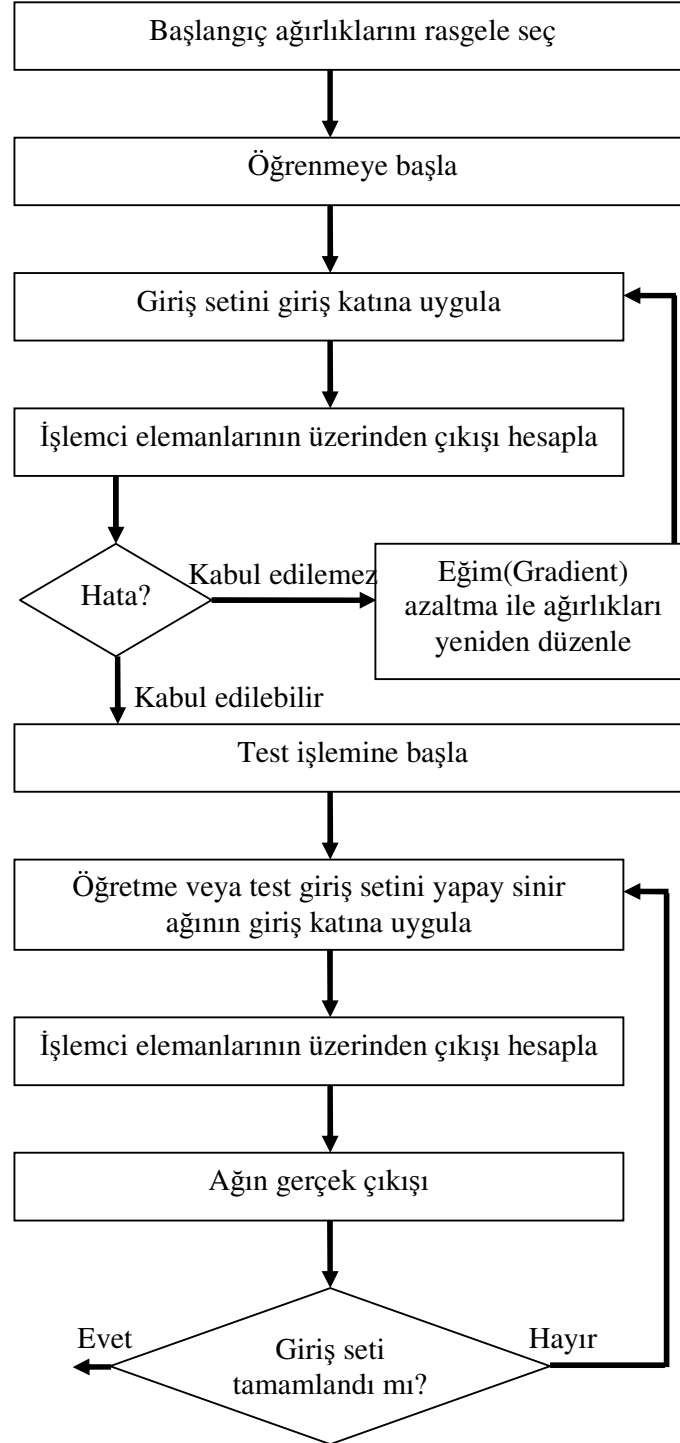
$$\delta_i^L = (T_i - O_i^L) O_i^L (1 - O_i^L),$$

ve gizli katmanın ağırlıkları,

$$\delta_i^l = \left(\sum_{r=1}^N \delta_r^{l+1} w_{ri}^{l+1} \right) O_i^L (1 - O_i^L).$$

Bu ağırlıkların ayarlanması ve eğitim alma işlemleri en küçük hata bulununcaya kadar tekrarlanır.⁷¹

⁷¹ Kartalopoulos, Stamatios V., Understanding Neural Networks and Fuzzy Logic, s. 78-80.



Şekil 2.4 Bir ağda kullanılan geriyayılımlı öğrenme algoritması.

2.1.1.1.2. Geri Yayılım Etkinlik İşlevleri

Geri yayılım çalışması süreci, etkinlik işlevlerinin sınırlı ve türevlenebilir olması gerekir. Bunu sağlayan etkinlik işlevi yukarıda anlatılmış olan sigmoid tipli olanıdır.

2.1.1.1.3. Öğrenme Oranı ve Ağ Üzerindeki Etkisi

Birçok eğitime işleminde öğrenme oranı ya da öğrenme katsayısı kullanılır. Öğrenme oranı ağın performansı üzerinde etkilidir. Bu değer genellikle pozitif bir değer olup sıfır ile bir arasında bir değer alır. Öğrenme katsayısı birden büyük alınırsa öğrenme algoritmasını ayarlamak kolaylaşır. Fakat ağda salınımlar meydana gelir. Küçük değerler ise, anlık hataların hızla düzeltilmesini sağlayamazken en iyi sonuca ulaşılmasını sağlar.

Öğrenme oranı küçük olduğunda eğitime işleminin zaman alırken, öğrenme oranı büyütüldüğünde eğitime zamanı kısalmıştır. Öğrenme oranı büyütüldüğünde eğitime işleminin için gerekli adım sayısı azalır ve aynı zamanda toplam hata üzerinde bir iyileştirme olur. Ancak öğrenme oranına çok büyük değerler verilirse, yakınsama olayı meydana gelmemektedir.⁷²

2.1.1.1.4. Momentum Teriminin Ağ Üzerindeki Etkisi

Hesaplamalara momentum teriminin katılmasının ağ üzerinde adım sayısının azalması ve toplam hatanın düşmesi gibi önemli etkileri vardır. Momentum katsayısı çok yüksek alınırsa ağın toplam hatası sıfıra doğru daha fazla bir eğilimle yaklaşır.

2.1.1.1.5. Gizli Katmandaki Sinir Sayısının Ağ Üzerindeki Etkisi

Gizli katmandaki sinir sayısı artırıldığında simülasyon sırasında hafıza kullanımı, işlemci yükü artar ve öğrenme süresi uzar. Ancak öğrenme işlemi daha hassas yapılmaktadır. Gizli katmandaki sinir sayısı az alındığında, ağın hatırlama yeteneğini azalmakta fakat öğrenme süresi kısalmaktadır. Gizli katmandaki sinir sayısı, giriş ve çıkış katmanındaki sinir sayıları dikkate alınarak belirlenir.

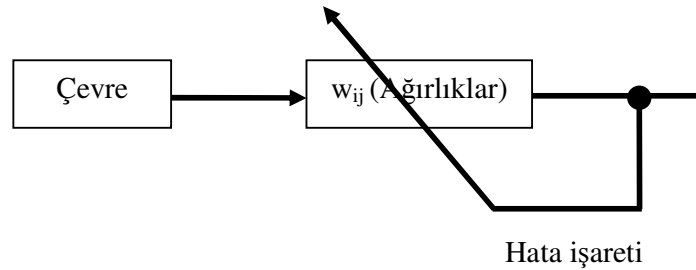
⁷² Elmas, Age, s.142-143, Mehrota, Mohan, Ranka, Age, s.79-86.

2.1.1.1.6. Hata Farkı Değişkeninin Ağ Üzerindeki Etkisi

Hata farkı değişkeninin değeri ya da tolerans, beklenen çıkışların hata değerlerinin hassasiyeti ile ilgilidir. Tolerans değerleri küçük alınırsa, ağın toplam hatası daha fazla sifıra yaklaşır ve bağlantı ağırlıklarındaki hassasiyet de artar. Aynı zamanda eğitim süresi ve adım sayısı artar.⁷³

2.1.2. Danışmansız Öğrenme

Danışmansız öğrenmede ağa çıkışlar verilmez. Bu nedenle hata dikkate alınmaz. Ağ girişlere göre kendi kendini örnekler. İstenen ya da hedeflenen bir çıkış bilgi olmadığı için ağırlık değerleri giriş verileri arasındaki matematiksel ilişkilere göre ayarlanır. Çıkışları kontrol edecek bir öğretmen yoktur ve bu nedenle danışmansız öğrenme olarak nitelendirilmiştir. Bir danışmana ihtiyaç duyulmadığından kendiliğinden öğrenme olarak da adlandırılır. Kohonen danışmansız öğrenme yöntemini kullanarak geliştirdiği özörgütlemeli harita ağ yapısını biyolojik sistemlerdeki öğrenmeden esinlenmiştir. Bu yöntemde sinirler öğrenmek için elverişli duruma gelmek amacıyla yarışır. Sonuçta en büyük çıkışa sahip olan sinir kazanır ve komşularına bağlantı ağırlık değerlerini güncellemek için uyarı gönderir. Şekil-2.5’de danışmansız öğrenme verilmiştir.⁷⁴



Şekil 2.5 Danışmansız öğrenme.

Danışmansız öğrenmede ağ, çıkış bilgisi olmadığı için girilen bilgilerle çalışır. Bu öğrenme yönteminde sinirler dışardan bir yardım alamadıkları için kendi kendilerini örgütleyebilmenin yolunu bulmalıdırlar. Bu öğrenmede ağ, yaparak öğrenmektedir.

⁷³ Elmas, Age, s.142-143, Mehrota, Mohan, Ranka, Age, s.79-86.

⁷⁴ Elmas, Age, s.149.

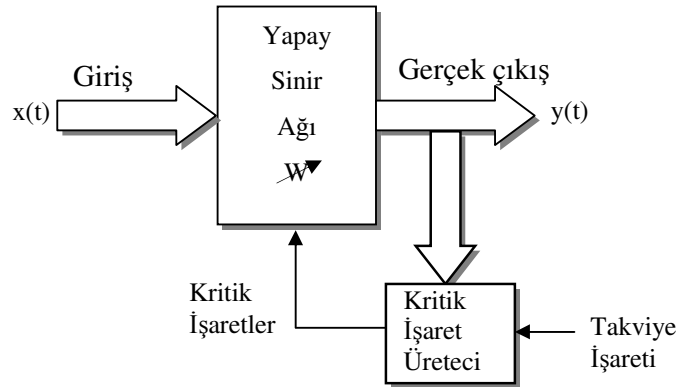
Danışmansız öğrenme gelecek için ümitler vaat etmektedir. Bazı araştırmacılar tarafından bu öğrenme kuralı ile bilgisayarların kendi kendine robotik hareketleri öğrenebilecekleri düşünülmektedir. Danışmansız öğrenme kuralları şunlardır:⁷⁵

- 1) Karma Öğrenme Kuralı
- 2) Yarışmacı Öğrenme Kuralı
- 3) Yarışmacı Sinirsel İşaretler Kuralı
- 4) Özörgütlemeli Harita Ağı Kuralı

Bazı araştırmacılar 3. bir öğrenme kuralı olduğunu düşünmektedirler. Bu kural Takviyeli Öğrenme'dir

2.1.3. Takviyeli Öğrenme

Bu öğrenme kuralı danışmanlı öğrenme kuralının özel bir şeklidir. Şekil-2.6'da takviyeli öğrenme yapısı vardır. Bu öğrenmede giriş değerlerine karşılık istenen çıkışın bilinmesine gerek yoktur. Çıkış değerlerinin bilinmemesine rağmen elde edilen çıkışın verilen girişlere uygunluğunu değerlendiren bir kriter kullanılır.⁷⁶

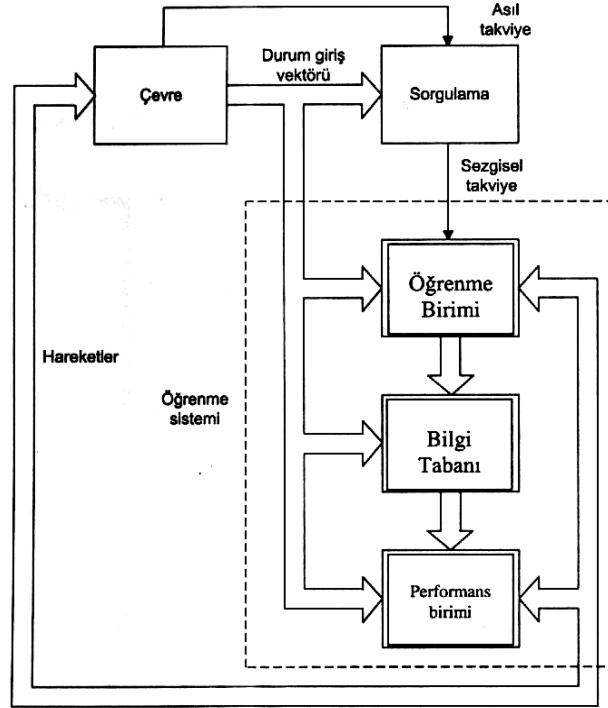


Şekil 2.6 Takviyeli öğrenme yapısı.

⁷⁵ Elmas, Age, s.147.

⁷⁶ Sağıroğlu ve diğerleri, Age, s. 79-81.

Optimizasyon problemlerine çözüm getirmek için Hinton ve Sejnowski'nin geliştirdiği Boltzman Kuralı veya Genetik Algoritma takviyeli öğrenmeye örnektir. Şekil-2.7'de takviyeli öğrenme algoritmasının blok diyagramı ayrıntılı olarak verilmiştir. Bu öğrenme çevre, sorgulama ve öğrenme sistemi olmak üzere üç modülden oluşur. Öğrenme sisteminde de öğrenme birimi, bilgi tabanı, ve performans birimi modülleri vardır.



Şekil 2.7 Takviyeli öğrenme blok diyagramı.

2.2. Uygulamaya Göre Öğrenme Yaklaşımları⁷⁷

- **Off-line öğrenme:** Bu yaklaşımda ağ farklı bir ortamda eğitilir. Eğitim sonucunda elde edilen ağırlıklar ile ağın yapısı, daha sonra on-line uygulamalarda kullanılır.
- **On-line öğrenme:** On-line öğrenmede, sistem üzerinde bir öğrenme algoritması ve bu yapıyı destekleyen donanım ve yazılımlar mevcuttur.

⁷⁷ Sağiroğlu ve diğerleri, Age, s. 83.

2.3. Öğrenmede Kullanılan Diğer Yaklaşımlar

- **Desen tabanlı eğitim:** Bu yaklaşımda, eğitim verilerinin ağa her uygulandığında ağırlıkları da değiştirilir.
- **Grup eğitim:** Burada eğitim veri setinin tamamı ağa uygulandıktan sonra ağın ağırlıkları değiştirilir.
- **Artırımlı öğrenme:** Bu tür öğrenmede herhangi bir ağırlık kümesi için amaç fonksiyonu hesaplanır. Bu sayede, eğitimde gelişmenin olup olmadığı görülebilir ve istenilen doğruluk için amaç fonksiyonun minimum değeri hesaplanabilir.
- **Yapısal öğrenme:** Ağ öğrenmeye kısa süreli eğitilen, ara katmanı olmayan bir yapı ile başlar. Sonra ağırlıklar değiştirilmeksizin ağa bir ya da daha çok ara katman eklenir, eğitim tekrarlanır ve işlem devam eder.
- **Ardışıl öğrenme:** Bir çok on-line öğrenme algoritması, öğrenme oranı, amaç fonksiyonu yaklaşımı ya da Hessian matrisi gibi ağırlıklara ilave edilen bilginin depolanmasına gereksinim duyarlar.⁷⁸

2.4. Yapay Sinir Ağlarını Test Etme

Ağın eğitilme işlemi tamamlandıktan sonra yani en iyi çıkışı elde edebilecek ağırlıklar belirlendikten sonra ağın öğrenme olayını gerçekleştirip gerçekleştirmediğini ölçmek için yapılan denemelere “ağın test edilmesi” denir. Test aşamasında ağın görmediği örnekler kullanılır. Bu örnekler ağa gösterilir ve ağ eğitim sonrasında belirlemiş olduğu en uygun ağırlık değerlerini kullanarak çıkışı üretir. Elde edilen bu çıkışın doğruluğuna bakılarak ağın öğrenmesi hakkında bilgi sahibi olunur. Sonuç ne kadar iyiye ağ eğitiminin performansı o kadar iyidir. Eğitimde kullanılan örnek kümeyle “eğitim seti”, test için kullanılan kümeyle “test seti” adı verilir. Yapay sinir ağların bu şekilde bilinen örnekler kullanılarak eğitildikten sonra bilinmeyen örnekler için yorum yapabilme (genelleme) yeteneğine “Adaptif Öğrenme” denir.⁷⁹

2.5. Yapay Sinir Ağları Tasarımında Dikkat Edilmesi Gereken Hususlar

Bir yapay sinir ağ modelinin başarılı olabilmesi için dikkat edilecek bazı hususlar vardır. Bunlar;

⁷⁸ Sağıroğlu ve diğerleri, Age, s. 83-85.

⁷⁹ Öztemel, Age, s. 55-56.

2.5.1. Uygulama Başarısı

Yapay sinir ağlarıyla çözülemeyecek problem azdır. Ancak bunun için uygun yapının, bu yapıya uygun öğrenme algoritmasının, seçilen algorithmada uygun parametrelerin, yapıya uygun giriş, ara katman ve çıkış sayısının, ara katmandaki nöron sayısının, nöronlar için aktivasyon fonksiyonunun, eğitim ve test kümesinin ve kümelere uygulanacak olan normalizasyon seviyelerinin belirlenmesi zor ve zaman alıcıdır. Bunlar da ancak tecrübe ile elde edilir. Bunlar belirlenirken dikkat edilmezse karmaşıklık olur. Bu karmaşıklığın en aza indirgenmesi hedeflenir.

Uygun parametrelerle tasarlanmış ağda istikrarlı sonuçlar üretilecektir. Bunun dışında ağın tepki süresi de önemlidir. Bu sürenin kısa olabilmesi için ağın yeterince küçük olması gerekir.

2.5.2. Uygulamada Karşılaşılan Güçlükler

Yapay sinir ağlarında uygulamanın başarılı olabilmesi için uygun parametrelerin seçilmesi gereklidir. Ancak bu parametreleri seçmek kolay değildir. Seçimde karşılaşılan zorluklar şunlardır;⁸⁰

- Problemin çözümüne uygun yapay sinir ağı yapısının seçilmesi,
- En uygun çözüm için yapay sinir ağı giriş ve çıkış sayılarının ne kadar olacağıının belirlenmesi,
- Ara katmanda ne kadar nöron bulunacağı,
- Ara katman sayısının ne kadar olacağı,
- Kullanılacak öğrenme algoritmasının seçilen yapay sinir ağına uygun olması,
- Seçilen öğrenme algoritma için en uygun parametrelerin seçilmesi,
- Seçilen veri kodlama yapısı,
- Veri normalizasyon yaklaşımı,
- Seçilen aktivasyon fonksiyonunun yapısı,
- Toplama fonksiyonunun seçimi,
- Uygun olan performans fonksiyonu seçimi,
- Uygun iterasyon ve adım sayısı seçimi,
- Ön işleme ve son işleme işlemleri,
- Uygun veri tipinin ve sayısının belirlenmesi,

⁸⁰ Sağıroğlu ve diğerleri, Age, s. 105-107.

- Ağda kullanılacak bağlantı türünün belirlenmesidir.

2.6. Yapay Sinir Ağı Uygulamalarında Takip Edilmesi Gereken İşlemler

Yapay sinir ağlarının seçilen probleme en uygun çözümü bulabilmeleri için model tasarımı aşağıdaki gibi yapılmalıdır;⁸¹

Verilerin toplanması: Probleme ilgili verilerin toplanmalı ve hangilerinin giriş hangilerinin çıkış olarak kullanılacağına ve kaçar tane kullanılacağına karar verilmelidir. Bu sayede giriş-çıkış katmanlarında kaç hücre olacağına karar vermiş oluruz.

Verilerin Alt Verilere Bölünmesi: Toplanan verilerin bir kısmı yapay sinir ağlarının eğitilmesinde, bir kısmı uygunluğunun araştırılmasında, bir kısmı da test işleminde kullanılacağı için 3 alt gruba ayrılır. Bu gruplandırma %40, %30, %30 şeklinde olabileceği gibi duruma göre rasgele gruplama da yapılabilir.

Yapay Sinir Ağları Yapısı Seçimi: Uygulama problemine göre yapılır. Bu nedenle hangi tür uygulamada hangi model kullanılacağına bilinmesi gerekir. Tablo-2.1' de bazı alanlar için kullanılacak ağ modelleri verilmiştir.⁸²

⁸¹ Şen, Age, s.80-82, Sağıroğlu ve diğerleri, Age, s. 107-116.

⁸² Elmas, Age, s.176.

Tablo 2.1 Yapay sinir ağıları ve uygulama alanları.

Uygulama Alanı	Ağ Türü
Tahmin	Geri Yayılım Ağı
	Delta Bar Delta
	Genişletilmiş Delta Bar Delta
	Yüksek Seviyeli Ağlar
	Özörgütlemeli Harita Ağı
	Perceptron
Sınıflandırma	Geri Yayılım Ağı
	Perceptron
	Öğrenme Vektörü Niclemesi
	Olasılıksal Sinir Ağları
	Kohonen Ağı
	Boltzman Makinesi
Veri Birleştirme	Hopfield
	Boltzman Makinesi
	Hamming Ağı
	İki Yönlü Çağrışımli Bellek
	Yığın Ağları
	Vektör Niclemesi Ağı
Veri Kavramlaştırılması	Uyarlanırlı Rezonans Ağı
	Özörgütlemeli Harita Ağı
Veri Süzülmesi	Yeniden Dolaşım
Resim veya Görüntü İşleme	Geri-Yayılım Ağı
	Perceptron

Yapay sinir ağıları algoritması seçimi: Ağın başarılı olabilmesi için önemli bir etken de öğrenme algoritmasıdır. Seçilen ağ yapısına uygun olan öğrenme algoritmasını seçmek gerekmektedir. Yapay sinir ağlarının az veriyle öğrenmesi ve genelleme yapabilmesi istenir. Ağın genellemesi yapılan testlerle kontrol edilmelidir. Aşırı-öğrenme söz konusu ise ağ kabul edilebilir çıkışlar üretmez.

En uygun öğrenme seviyesini yakalayabilmek için öğrenme fonksiyonunun kontrolü için, kullanılan performans fonksiyonunun beklenen değere ulaşım ulaşılmadığına bakılır.

Aktivasyon fonksiyonunun seçimi: Seçilecek aktivasyon fonksiyonunu da ağın öğrenmesi için önemlidir. Problem uygun olan fonksiyonu belirlemek, ağın başarısı

artırır. Bazı problemler için tek tip fonksiyon kullanmak yerine, değişik fonksiyonlar kullanılarak daha iyi sonuçlar elde edilir.

Başlangıç Ağırlık Değerlerinin Belirlenmesi: Hücreler arasındaki bağlantıların ağırlık değerleri başlangıçta rasgele ve küçük seçilir. Tekli Doğrusal Algılayıcı (TDA)' lar için 0 ve 1 veya -1 ve 1 sayıları arasındaki değerler alınır.

Ön ve son veri işleme: Yapay sinir ağlarının diğer yöntemlerden ayıran en önemli özelliklerinden biri olan doğrusal olmama özelliğini anlamlı yapan, verilerin normalizasyona tabi tutulmasıdır. Normalizasyon değeri çok büyük olan kümülatif toplamların oluşturacağı olumsuzluklara engel olur. Ön veri işleme işlemi, verinin en doğru ve en hızlı şekilde hazırlanmasını ifade eder. Genellikle veri $[0,1]$ veya $[-1,1]$ aralıklarında ölçeklenir. Eğitim aşamasında normalize edilmiş veriler kullanılır. Eğitim bittiğinde alınan çıkış değerleri ters normalize işlemine tabi tutularak gerçek çıkışlar elde edilir. Transfer fonksiyonu aynı zamanda ölçekleme fonksiyonu olarak da kullanılabilir.

Ara katman ve ara katmandaki nöron sayısının belirlenmesi: Bir yapay sinir ağında uygun ara katman ve ara katmandaki nöron sayısını belirlemek için bazı yaklaşımlar vardır, kesin kurallar yoktur. En basit yolu en iyi sonucu alana kadar deneme yapmaktır.

Bazı araştırmacılar tarafından her türlü problem için en fazla iki ara katmanın kullanılması söylenebilir de, bazı problemler için bu her zaman bu doğru olmayabilir. Burada dikkat edilmesi gereken ağın olabildiğince basit yapıda olmasıdır. Ağ yapısı basit olmazsa öğrenme süresi artmaktadır.

Bağlantı türünün belirlenmesi: Ara bağlantılar çoğunlukla kısmen veya tamamen ileri beslemeli veya çift yönlüdür.

Veri kodlama: Veri kodlama bir verinin bilgiye dönüştürülmesi için işlenebilmesini kolaylaştırır ve verinin mimarisini değiştirir.

Dilsel kodlama ve sayısal kodlama olmak üzere iki farklı veri kodlama yaklaşımı vardır. Bu veri kodlama işlemi genellikle veri sıkıştırmaya uygun problemler için kullanılır. Veri kodlama ile ağdaki giriş ve çıkış katmanındaki işlemci eleman sayıları

kontrol edilir. Sınıflandırma problemlerinde her sınıfın tek bir çıkış değeri ile ifade edildiği ikili kodlama tercih edilmelidir.

Performans fonksiyonu seçimi: İleri beslemeli ağlar için kullanılan performans fonksiyonları aşağıdaki gibidir.

$$\text{Hata Kareler Ortalaması: } MSE = \frac{1}{N} \sum_{i=1}^N (t_i - td_i)^2 \quad (2.1)$$

$$\text{Hata Kareler Toplamı: } SSE = \sum_{i=1}^N (t_i - td_i)^2 \quad (2.2)$$

$$\text{Hata Kareler Ortalaması Karekökü: } RMS = \sqrt{\frac{1}{N} \sum_{i=1}^N (t_i - td_i)^2} \quad (2.3)$$

Performans fonksiyonuna ağın biaslarının ve ağırlıklarının kareleri toplamının ortalama değeri eklenerek genelleştirme iyileştirilir. Genelleştirmeyi iyileştirmenin bir yolu da öğrenmeyi erken sonlandırmadır.

Yenileme: Geri besleme sonrasında yeniden belirlenmiş olan bağlantı değerleri aynı giriş değerleri ile yeniden ileri besleme işlemi yapılır ve bu işlem istenen çıkışlarla elde edilen çıkışlar arasındaki hata değerinin beklenen hata değerinden küçük olana kadar tekrarlanır.

Durma: İstenen çıkışlarla elde edilen çıkışlar arasındaki hata değeri önceden belirlenen hata değerinden küçük olduğu zaman eğitim bitmiş demektir.

Kontrol: Eldeki bütün eğitim örnek seti ile eğitim tamamlandıktan sonra uygunluk için ayrılan verilerle işlem yapılarak çıkışlar karşılaştırılır hata değeri istenilen kadar küçükse ağ kullanıma hazırdır. Değilse uygunluk verileri ile de eğitime devam edilir.

Kullanım: Bu aşamada artık yapay sinir ağlarının önceden görmediği girişlerden çıkışlar üretebilir.⁸³

⁸³ Şen, Age, s.80-82, Sağıroğlu ve diğerleri, Age, s. 107-116.

BÖLÜM 3

Yapay Sinir Ağları Uygulamaları

Yapay sinir ağları üzerinde binlerce uygulama yapılmış ve başarılı sonuçlar üretmiştir. Geniş bir uygulama alanına sahiptir. Temel olarak yapay sinir ağları uygulamaları ile aşağıdaki işlemler yapılmaktadır:⁸⁴

Tahmin: Ağa sunulan giriş değerlerine karşılık çıkış değerlerini tahmin edebilirler.

Sınıflandırma: Kendilerine sunulan bilgileri kategorize ederler.

Veri ilişkilendirme: Ağlar, ağa sunulan verilerin hatalı ya da eksik olup olmadığını belirler. Eksik bilgi varsa tamamlar.

Veri filtreleme: Ağlar birçok veri arasından uygun verileri bulurlar.

Tanıma ve eşleştirme: Değişik şekil ve örüntülerin tanınması, eksik, belirsiz bilgilerin eşlenmesi işlemini yaparlar.

Teşhis: Ağlar sistemlerin olumsuzluklarını ortaya çıkarır ve problemlerin teşhis edilmesini sağlar.

Yorumlama: Bir olay hakkında toplanan verilerle ağ eğitildikten sonra bu bilgileri kullanarak yeni olayları yorumlarlar.

Yukarıda verilen fonksiyonları gerçekleştiren yapay sinir ağları çok farklı alanlarda kullanılmaktadır.

Alanlarına göre yapay sinir ağları uygulamaları:⁸⁵

3.1. Endüstriyel Uygulamalar

- Kalite denetimi
- Kalite geliştirme
- Hidroelektrik santralinde bulunan 4 büyük senkron kompanzatorün güç dağıtım şebekelerinde kullanımı

⁸⁴ Elmas, Age, s. 175, Öztemel, Age, s. 36, Clifford Lau, *Artificial Neural Networks, Paradigms, Applications and Implementations*, New York: The Institute of Electrical and Electronics Engineers Press, 1992, s. 3.

⁸⁵ Sağıroğlu ve diğerleri, Age, s. 117-340, Öztemel, Age, s. 205-206., Elmas, Age, s.177-178.

- Ark fırın denetim sistemi ve bu sistemin elektrik tasarrufu
- Güneş enerjisi pilleri üretiminde konum kalınlık hatalarının saptanması
- Petrol endüstrisi için rafine işlemleri
- Elektrik motorlarında arıza tespiti
- Konuşmayı metne çevirme
- Otomatik dil çevirisi
- İşitme ve bedensel engelliler için protez geliştirilmesi
- Sesle kumanda etme
- Doğal dil işleme
- Fırınlarda ürettiği gaz miktarını tahmin etme
- Ürün tasarımı
- Makinelerin bakımı
- Kimyasal proseslerin dinamik modellenmesi
- Otomobillerde otomatik rehber sisteminin geliştirilmesi
- Robotlarda görme sistemi ve kontrolü
- Cep telefonlarında ses ile çalışabilme
- Araba pistonlarının üretim şartlarının belirlenmesi
- Elektronik yonga hata analizleri
- Optimizasyon çalışmaları
- Müşteri tahmini, pazar verilerinin değerlendirilmesi ve analiz edilmesi
- Kömür güç istasyonları için çevrimiçi karbon akımı ölçülmesi
- İşlerin makinelere atanması ve çizelgeleme
- Gezgin satıcı probleminin çözümü

3.2. Mühendislik Uygulamaları

- Kontrol akış örüntülerinin (paternlerinin) sınıflandırılması
- Günlük sıcaklık tahmini
- Endüstriyel robot sensör modellemede sinerji
- Sistem kimliklendirme
- Robotik uygulamalar
- Mikroşerit antenler
- Dairesel mikroşerit anten tasarımı
- Dikdörtgen mikroşerit anten tasarımı

- Sayısal görüntü işleme
- Görüntü tanıma
- Raster verilerde geometrik dönüşüm
- Katı modelleme
- Görüntülerin otomatik olarak yöneltilmesi
- Sayısal görüntülerin kodlanması
- Uzaktan algılama
- İşaret işleme

3.3. Finansal Uygulamalar

- Makro ekonomik tahminler
- Borsa benzetim çalışmaları endekslerinin tahmini
- Kredi kartı hilelerinin tespiti
- Kredi kartı kurumlarında iflas tahminleri
- Banka kredilerinin değerlendirilmesi
- Emlak kredilerinin yönetilmesi
- Döviz kuru tahminleri
- Risk analizleri
- El yazısı ve imza tanıma

3.4. Askeri Uygulamalar

- Hedef tanıma ve takip sistemleri
- Yeni sensörlerin performans analizleri
- Radar ve görüntü sinyalleri işleme
- Sensör fizyonu
- Askeri uçakların uçuş yörüngelerinin belirlenmesi (optimizasyonu)
- Mayın dedektörleri

3.5. Sağlık Uygulamaları

- Solunum hastalıklarının teşhisi
- EEG, ECG analizleri
- Transplant zamanlarının optimizasyonu
- Hastalıkların teşhisi ve resimlerden tanınması

- Kardiovascular sistemlerin modellenmesi ve teşhisi
- Tıbbi resim işleme
- CTG izleme
- Hamile kadınların karınlarındaki çocukların kalp atışlarının izlenmesi
- Yumurtalık kanserinin izlenmesi
- Üroloji uygulamaları
- RNA ve DNA'daki aminoasit incelemesi
- Hastaların ilaçlara karşı tepkilerinin incelenmesi
- Kazalarda sakatlıklardan korunma

3.6. İletişim Sanayi Uygulamaları

- Faks ve telefon hatlarının güncellenmesi
- Hatlarda olabilecek yankıların giderilmesi
- Veri sıkıştırma

3.7. Ulaştırma ve Havacılık Sanayi Uygulamaları

- Fren denetimi
- Mühendislik hataları
- Etkin süspansiyon
- Uzay mekiğinde manevra denetimi
- Uçaklarda titreşim seviyeleri ve sesin görüntülenerek motor sorunlarının erken uyarı sistemi
- Kara taşıtın sürücüsüz seyri
- Bir helikopterin denetimi
- Hava alanlarında bomba tanıma ve ortaya çıkarma
- Uçak parçalarının hata teşhislerinin yapılması
- Rotalama sistemleri

BÖLÜM 4

TEMEL BİLEŞENLER ANALİZİ

Çok değişkenli istatistiksel analizde (n) tane bireye (nesne) ilişkin (p) tane değişken (özellik) incelenmektedir. Bu özelliklerden birçoğunun birbiriyle ilişkili (bağımlı) ve değişken sayısının (p) çok büyük olması, çeşitli değerlendirmeler yapılmasını güçleştirmektedir. Böyle durumlarda başvurulan tekniklerden en önemlisi; Temel Bileşenler Analizidir (Principal Component Analysis). Genel olarak değişkenler arasındaki bağımlılık yapısının yok edilmesi ve/veya boyut indirgeme amacıyla kullanılan bu teknik başlı başına bir analiz olduğu gibi başka analizler için veri hazırlama tekniği olarak da kullanılmaktadır.⁸⁶

Temel Bileşen Analizi yapmanın üç amacı vardır:

1. Veri indirgemesi yapmak.
2. Tahminleme yapmak.
3. Veri setini bazı yöntemlerin analiz edebileceği forma sokmak.⁸⁷

Temel Bileşenler Analizi bir değişkenler setinin varyans-kovaryans yapısını, bu değişkenlerin doğrusal birleşimleri vasıtasıyla açıklayarak, boyut indirgenmesi ve yorumlanmasını sağlayan, çok değişkenli bir istatistik yöntemidir. Yöntemde karşılıklı bağımlılık yapısı gösteren, ölçüm sayısı (n) olan (p) adet değişken; doğrusal, ortogonal ve birbirinden bağımsız olma özelliklerini taşıyan k ($k \leq p$) tane yeni değişkene dönüştürülmektedir.

(n) tane birime ait (p) özellik ölçüldüğünde toplam değişkenlik (varyans), (p) adet değişkenin tümü tarafından açıklanmaktadır. Toplam değişkenliğin önemli bir kısmı k ($k \leq p$) bileşen tarafından açıklanabilir. Bu durumda, (k) adet bileşen gerçek (p) adet değişkeni temsil edebilmektedir. Böylece (n) ölçümdeki (p) değişken, önemli bir bilgi (varyans) kaybı olmadan, (n) ölçümdeki, (p) (k) değişkene indirgenmektedir. Söz konusu (k) adet yeni değişken, gerçek değişkenlerin bazı kısıtlamalarına bağlı kalınarak oluşturulmuş çeşitli doğrusal birleşimleridir.

Temel Bileşenler Analizi ile ulaşılmak istenen ilk sonuç; X_1, X_2, \dots, X_p gibi (p) tane değişkeni, önemli bir bilgi kaybına neden olmaksızın, bu değişkenleri temsil

⁸⁶ Bülent Dinçer, Metin Özasan ve Erdoğan Satılmış, *İllerin Sosyo-Ekonomik Gelişmişlik Sıralaması Araştırması (1996)*. Ankara: DPT. Bölgesel Gelişme ve Yapısal Uyum Genel Müdürlüğü, 1996, s.33.

⁸⁷ Kazım Özdamar, *Paket Programlar İle İstatistiksel Veri Analizi-2*, Eskişehir: Kaan Kitabevi, 2002, s.215.

edebilen daha az sayıda değişkene indirgemektir. Daha sonra indirgenmiş yeni değişkenler ile çalışmanın amacı doğrultusunda çeşitli sonuçlara ulaşılabilmektedir.

$\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_p$ vektörlerinin standartlaştırılmış hali olan $\mathbf{Z}_1, \mathbf{Z}_2, \dots, \mathbf{Z}_p$ vektörlerinin (p) tane doğrusal birleşimi, ya da temel bileşeni;

$$\mathbf{Y}_1 = (\mathbf{a}_1)^t \mathbf{Z} = a_{11} \mathbf{Z}_1 + a_{21} \mathbf{Z}_2 + \dots + a_{p1} \mathbf{Z}_p$$

$$\mathbf{Y}_2 = (\mathbf{a}_2)^t \mathbf{Z} = a_{12} \mathbf{Z}_1 + a_{22} \mathbf{Z}_2 + \dots + a_{p2} \mathbf{Z}_p$$

$$\dots \quad \dots \quad \dots$$

$$\dots \quad \dots \quad \dots$$

$$\mathbf{Y}_p = (\mathbf{a}_p)^t \mathbf{Z} = a_{1p} \mathbf{Z}_1 + a_{2p} \mathbf{Z}_2 + \dots + a_{pp} \mathbf{Z}_p$$

Burada; $\mathbf{Z}_1, \mathbf{Z}_2, \dots, \mathbf{Z}_p$ 'ler standartlaştırılmış veri matrisinin satır vektörleri ((p) değişkene ait (p) tane satır vektör), $\mathbf{Y}_1, \mathbf{Y}_2, \dots, \mathbf{Y}_p$ 'ler temel bileşenler, a_{ij} ler ise her bir temel bileşenin hangi değişkenle, hangi oranda ilişkilendirildiğini gösteren sabit sayılardır. a_{ij} sabit sayıları temel bileşen yükleridir. Temel bileşen yükleri, temel bileşenlerin değişkenlere varyans katkısını gösteren ağırlıklardır ve temel bileşenleri, değişkenlerin hangi ağırlıklarla tanımladıklarını göstermektedir. Temel bileşenler ortogonal seçileceğinden, a_{ij} ağırlıkları değişkenler ile temel bileşenler arasındaki korelasyon katsayısıyla orantılıdır. a_{ij} : (i)'inci değişkenin (j)'inci temel bileşendeki ağırlığıdır.

Temel bileşenlerin varyansları ve kovaryansları;

$$\text{Var}(\mathbf{Y}_i) = \text{Var}((\mathbf{a}_i)^t \mathbf{Z}) = (\mathbf{a}_i)^t \mathbf{S} \mathbf{a}_i = (\mathbf{a}_i)^t \mathbf{R} \mathbf{a}_i \quad i = 1, 2, \dots, p$$

$$\text{Cov}(\mathbf{Y}_i, \mathbf{Y}_k) = (\mathbf{a}_i)^t \mathbf{S} \mathbf{a}_k = (\mathbf{a}_i)^t \mathbf{R} \mathbf{a}_k \quad \text{dır.} \quad k = 1, 2, \dots, p$$

Eşitlikteki (S), standartlaştırılmış veri matrisinin kovaryans matrisi, (R) standartlaştırılmış veri matrisinin korelasyon matrisidir. Standartlaştırılmış veri matrisi kullanıldığından $\mathbf{R} = \mathbf{S}$ 'dir.

$\mathbf{Y}_1, \mathbf{Y}_2, \dots, \mathbf{Y}_p$ temel bileşenleri, orijinal değişkenlerin birbirinden bağımsız ve varyansları toplam sistem varyansını mümkün olabilecek en fazla bir biçimde açıklayan doğrusal birleşimleri olacak şekilde seçilecektir.

Bunun için izlenecek yol; birinci temel bileşen (\mathbf{Y}_1), toplam varyansa katkısı maksimum olacak şekilde $\mathbf{Z}_1, \mathbf{Z}_2, \dots, \mathbf{Z}_p$ 'lerin doğrusal birleşimleri olarak belirlenmektedir. İkinci temel bileşen (\mathbf{Y}_2), birinci temel bileşenden bağımsız olarak, birinci temel bileşenin açıkladığı varyanstan sonra geriye kalan toplam varyansa katkısı maksimum olacak şekilde, benzer şekilde üçüncü ve daha sonraki temel bileşenler her birinin toplam varyansa katkısı maksimum olacak şekilde ve birbirinden bağımsız olarak aşağıda gösterildiği gibi saptanmaktadır.

Toplam varyansa katkısı en fazla olan birinci temel bileşen,

$\mathbf{Y}_1 = a_{11}\mathbf{Z}_1 + a_{21}\mathbf{Z}_2 + \dots + a_{p1}\mathbf{Z}_p$ doğrusal birleşimidir.

$\text{MaxVar}(\mathbf{Y}_1) = (\mathbf{a}_1)^t \mathbf{R} \mathbf{a}_1$ eşitliğinden \mathbf{a}_1 vektörü birinci temel bileşenin varyansını maksimum yapacak şekilde belirlenmektedir. Ancak, \mathbf{a}_1 vektörü herhangi bir sabit sayı ile çarpılarak, değişkenlik hiçbir kısıtlamaya bağlı kalmaksızın artırılabilir. Bundan dolayı \mathbf{a}_i vektörlerinin birim uzunlukta $((\mathbf{a}_i)^t \cdot \mathbf{a}_i = 1)$ seçilmesi uygun olacaktır. Bu şekilde seçilen;

Birinci temel bileşen, $\text{max Var}((\mathbf{a}_1)^t \mathbf{Z})$ ve $(\mathbf{a}_1)^t \cdot \mathbf{a}_1 = 1$ şartlarını sağlayan $(\mathbf{a}_1)^t \mathbf{Z}$ doğrusal birleşimidir.

İkinci temel bileşen, $\text{max Var}((\mathbf{a}_2)^t \mathbf{Z})$ ile $(\mathbf{a}_2)^t \cdot \mathbf{a}_2 = 1$ ve

$\text{Cov}((\mathbf{a}_1)^t \mathbf{Z}, (\mathbf{a}_2)^t \mathbf{Z}) = \text{Cov}(\mathbf{Y}_1, \mathbf{Y}_2) = 0$ şartlarını sağlayan $(\mathbf{a}_2)^t \mathbf{Z}$ doğrusal birleşimidir.

(i) inci temel bileşen, $\text{max Var}((\mathbf{a}_i)^t \mathbf{Z})$, $(\mathbf{a}_i)^t \cdot \mathbf{a}_i = 1$ ve $k < i$ için $\text{Cov}(\mathbf{Y}_i, \mathbf{Y}_k) = 0$ şartlarını sağlayan $(\mathbf{a}_i)^t \mathbf{Z}$ doğrusal birleşimidir.

Amaç, değişkenlerin doğrusal birleşimlerinin oluşmasını sağlayan a_{ij} ($(i)=1,2,\dots,(p)$; $(j)=1,2,\dots,(p)$) katsayılarını, belirtilen şartlara bağlı kalarak tespit etmektir.

Teorik kısmı açıklanan temel bileşenler analizi, kısaca özetlenecek olursa;

- (n) ölçümdeki (p) değişkene ait veri matrisi standartlaştırılır,
- Standartlaştırılmış veri matrisinin korelasyon matrisi bulunur,
- Korelasyon matrisinin öz değerleri ve standartlaştırılmış öz vektörleri hesaplanır,
- Öz değerlerden temel bileşenlerin toplam varyansı açıklama oranları elde edilir,

Her bir öz vektörün transpozesi ile standartlaştırılmış veri matrisinin transpozesi çarpılarak temel bileşen değerleri bulunur.⁸⁸

⁸⁸ Dinçer ve diğerleri, Age, s.33-37.

BÖLÜM 5

GERİYAYILIMLI YAPAY SİNİR AĞI ÜZERİNE BİR UYGULAMA

Bu uygulamada bir geriyayılımlı yapay sinir ağı kullanılarak her il için verilen aşağıdaki giriş değişkenleri ile sosyo-ekonomik gelişmişlik endeksi tahmin edilmeye çalışılmıştır. Bu endeksi belirleyen tüm değişkenler:⁸⁹

Demografik Göstergeler

- Şehirleşme oranı
- Yıllık ortalama nüfus artış hızı
- Nüfus yoğunluğu
- Doğurganlık hızı
- Ortalama hane halkı büyüklüğü
- Toplam nüfus

İstihdam Göstergeleri

- Tarım işkolunda çalışanların toplam istihdama oranı
- Sanayi işkolunda çalışanların toplam istihdama oranı
- Ticaret işkolunda çalışanların toplam istihdama oranı
- Mali kurumlar işkolunda çalışanların toplam istihdama oranı
- Ücretli çalışanların toplam istihdama oranı
- Ücretli çalışan kadınların toplam istihdama oranı
- İşverenlerin toplam istihdama oranı

Eğitim Göstergeleri

- Okur yazar nüfus oranı
- Okur yazar kadın nüfusun toplam kadın nüfusuna oranı
- Üniversite bitirenlerin okul bitirenlere oranı
- İlköğretim okullaşma oranı
- Liseler okullaşma oranı

⁸⁹ Cemalettin Kaymak, Rasim Akpınar ve Ahmet Kındap, *İller ve Bölgeler İtibarıyla Gayri Safi Yurtiçi Hâsıladaki Gelişmeler (1987-2000)*, Ankara: Devlet İstatistik Enstitüsü Matbaası, 2003, s. 49.

- Meslek liseleri okullaşma oranı

Sağlık Göstergeleri

- Bebek ölüm oranı
- Onbin kişiye düşen hekim sayısı
- Onbin kişiye düşen diş hekimi sayısı
- Onbin kişiye düşen eczane sayısı
- Onbin kişiye düşen hastane yatağı sayısı

Sanayi Göstergeleri

- Organize sanayi bölgesi parsel sayısı
- Küçük sanayi sitesi işyeri sayısı
- İmalat sanayi işyeri sayısı
- İmalat sanayi yıllık çalışanlar ortalama sayısı
- İmalat sanayi kurulu güç kapasite miktarı
- Fert başına imalat sanayi elektrik tüketimi
- Fert başına imalat sanayi katma değeri

Tarım Göstergeleri

- Kırsal nüfus başına tarımsal üretim değeri
- Tarımsal üretim değerinin Türkiye içindeki payı

İnşaat Göstergeleri

- Daire sayısı
- Borulu su tesisatı bulunan daire oranı

Mali Göstergeler

- Gayri safi yurt içi hâsıla içindeki payı
- Fert başına gayri safi yurt içi hâsıla
- Banka şube sayısı
- Fert başına banka mevduatı
- Toplam banka mevduatı içindeki payı
- Toplam banka kredileri içindeki payı

- Kırsal nüfus başına tarımsal kredi miktarı
- Fert başına sanayi, ticari ve turizm kredileri miktarı
- Fert başına belediye giderleri
- Fert başına genel bütçe gelirleri
- Fert başına gelir ve kurumlar vergisi miktarı
- Fert başına kamu yatırımları miktarı
- Fert başına teşvik belgeli yatırım tutarı
- Fert başına ihracat miktarı
- Fert başına ithalat miktarı

Altyapı Göstergeleri

- Kırsal yerleşmelerde asfalt yol oranı
- Yeterli içme suyu götürülen nüfus oranı
- Tck asfalt yol oranı

Diğer Göstergeler

- Yeşil kartlı kişi oranı
- Onbin kişiye düşen özel otomobil sayısı
- Onbin kişiye düşen motorlu kara taşıtı sayısı
- Fert başına elektrik tüketim miktarı
- Fert başına telefon kontör değeri

Bu değişkenlerin Sivas ili için değerleri Ek-1'de verilmiştir.

Giriş Değişkenleri

Yukarıda verilen 58 değişkenden 4'ünün değeri kaynaklarda yer almadığı için bu 4 değişken kullanılamamıştır. Bu 4 değişken aşağıda verilmiştir.

- İmalat sanayi yıllık çalışanlar ortalama sayısı
- İmalat sanayi kurulu güç kapasite miktarı
- Fert başına imalat sanayi elektrik tüketimi
- Fert başına imalat sanayi katma değeri

Kalan 54 değişken temel bileşenler analizine tabi tutulmuştur. Temel Bileşenler Analizi ile 54 giriş değişkeninin indirgeme işlemi Matlab'de prepca fonksiyonu kullanılarak yapılmıştır.

Matlab'de Temel Bileşenler Analizi yapmak için prepca fonksiyonu kullanılır. Prepca fonksiyonuna ait arka planda yapılan işlemler şunlardır:

```
function [Ptrans,TransMat] = prepca(P,min_frac)
```

PREPCA Principal component analysis(Temel Bileşen Analizi)

Matlab'de temel bileşen analizi min_frac parametresi kullanılarak 54x81 boyutundaki giriş verisi 6x81 boyutuna indirilmiştir. Matlab'deki bu işleme ait algoritma aşağıdaki gibidir:

Kullanılan giriş parametreleri:

P : RxQ boyutundaki giriş matrisi.

min_frac : Çıkış bileşen sayısını belirlemek için kullanılan parametre.

Çıkış parametreleri ise:

Ptrans: Çıkış değerleri

TransMat : Transformasyon matrisidir.

Algoritma:

Temel Bileşenleri hesaplamak için singular value decomposition(svd) kullanılır. Giriş veri seti giriş covaryans matrisinin özvektörlerini içeren bir matrisle çarpılır. Böylece bağımsız bileşenlerden oluşan giriş veri setinden varyans büyüklüğüne göre sıralanmış bir dizi oluşturulur. Elde edilen bu dizinin her elemanı toplam varyansa bölünür. Dizinin her elemanın min_frac değişkeninden büyük olup olmadığına bakılır. Büyük olduğundaki değerler toplanır. Bu toplam indirgenmiş giriş değişkeni sayısını verir. Transformasyon matrisi boyutu istenen değişken sayısı elde edilecek şekilde azaltılır. Sonuçta 54x81 boyutundaki Giriş vektörü, transformasyon matrisi ile çarpılarak 6x81 boyutundaki indirgenmiş giriş veri seti elde edilir.

Algoritma Adımları:

1. Giriş seti PREMNMX fonksiyonu kullanılarak ölçeklendirilir.
2. $[R,Q]=\text{size}(P)$; Giriş verisinin satır ve sütun sayısı hesaplanır.
3. If $R > Q$, Hata; end $R>Q$ ise hata verir ve işlem sonlanır.
4. Değilse; $[\text{TransMat},s,v] = \text{svd}(P,0)$; $R<Q$ ise Singular Value Decomposition işlemi yapılarak temel bileşenler bulunur.
5. $\text{var} = \text{diag}(s).^2/(Q-1)$; Her temel bileşen için varyans hesaplanır.
6. $\text{total_variance} = \text{sum}(\text{var})$; Toplam varyans hesaplanır.
7. $\text{frac_var} = \text{var}./\text{total_variance}$; Dizinin her elemanı toplam varyansa bölünür
8. $\text{greater} = (\text{frac_var} > \text{min_frac})$; Dizinin her elemanın min_frac değişkeninden büyük olup olmadığına bakılır. Greater fonksiyonu büyükse 1, değilse 0 değerini döndürür.
9. $\text{size_pc} = \text{sum}(\text{greater})$; Büyük olduğundaki değerleri yani 1'ler toplanır. Bu olması istenilen giriş değişkeni sayısını verir.
10. $\text{TransMat} = \text{TransMat}(:,1:\text{size_pc})'$; Transformasyon matrisi boyutu istenen değişken sayısını elde edecek şekilde azaltılır.
11. $\text{Ptrans} = \text{TransMat}*P$; 54×81 boyutundaki Giriş vektörü, transformasyon matrisi ile çarpılarak 6×81 boyutundaki indirgenmiş giriş veri seti elde edilir.

Matlab'de her temel bileşen için hesaplanan varyanslar Tablo-5.1'de gösterilmiştir.

Tablo 5.1 Giriş değişkenleri ve varyansları.

Değişken Adı	Varyans Değeri
Toplam Nüfus	24,019
Şehirleşme Oranı	2,5483
Yıllık Ortalama Nüfus Artış Hızı	0,50771
Nüfus Yoğunluğu	0,316
Doğurganlık Hızı	0,2475
Ortalama Hane halkı Büyüklüğü	0,21112
Tarım İşkolunda Çalışanların Toplam İstihdama Oranı	0,19859
Sanayi İşkolunda Çalışanların Toplam İstihdama Oranı	0,18641
Ticaret İşkolunda Çalışanların Toplam İstihdama Oranı	0,15729
Mali Kurumlar İşkolunda Çalışanların Toplam İstihdama Oranı	0,14173
Ücretli Çalışanların Toplam İstihdama Oranı	0,10672
Ücretli Çalışan Kadınların Toplam İstihdama Oranı	0,095195
İşverenlerin Toplam İstihdama Oranı	0,084399
Okur Yazar Nüfus Oranı	0,068965
Okur Yazar Kadın Nüfusun Toplam Kadın Nüfusuna Oranı	0,065316
Üniversite Bitirenlerin Okul Bitirenlere Oranı	0,057792
İlköğretim Okullaşma Oranı	0,054306

Liseler Okullaşma Oranı	0,04199
Meslek Liseleri Okullaşma Oranı	0,039384
Bebek Ölüm Oranı	0,037582
Onbin Kişiye Düşen Hekim Sayısı	0,03365
Onbin Kişiye Düşen Diş Hekimi Sayısı	0,025612
Onbin Kişiye Düşen Eczane Sayısı	0,024302
Onbin Kişiye Düşen Hastane Yatağı Sayısı	0,022069
Organize Sanayi Bölgesi Parsel Sayısı	0,020336
Küçük Sanayi Sitesi İşyeri Sayısı	0,016491
İmalat Sanayi İşyeri Sayısı	0,016329
Kırsal Nüfus Başına Tarımsal Üretim Değeri	0,01421
Tarımsal Üretim Değerinin Türkiye İçindeki Payı	0,010874
Daire Sayısı	0,0099861
Borulu Su Tesisatı Bulunan Daire Oranı	0,0092804
Gayri Safi Yurt İçi Hasıla İçindeki Payı	0,0082453
Fert Başına Gayri Safi Yurt İçi Hasıla	0,0070693
Banka Şube Sayısı	0,0053047
Fert Başına Banka Mevduatı	0,0049151
Toplam Banka Mevduatı İçindeki Payı	0,0046227
Toplam Banka Kredileri İçindeki Payı	0,0041777
Kırsal Nüfus Başına Tarımsal Kredi Miktarı	0,0039759
Fert Başına Sınai, Ticari Ve Turizm Kredileri Miktarı	0,0025655
Fert Başına Belediye Giderleri	0,0022776
Fert Başına Genel Bütçe Gelirleri	0,0018633
Fert Başına Gelir Ve Kurumlar Vergisi Miktarı	0,0016559
Fert Başına Kamu Yatırımları Miktarı	0,0014803
Fert Başına Teşvik Belgeli Yatırım Tutarı	0,0013574
Fert Başına İhracat Miktarı	0,0011333
Fert Başına İthalat Miktarı	0,00061346
Kırsal Yerleşmelerde Asfalt Yol Oranı	0,00050885
Yeterli İçme Suyu Götürülen Nüfus Oranı	0,00039082
TCK Asfalt Yol Oranı	0,00026543
Onbin Kişiye Düşen Özel Otomobil Sayısı	0,00014849
Onbin Kişiye Düşen Motorlu Kara Taşıtı Sayısı	9,12E-05
Fert Başına Elektrik Tüketim Miktarı	4,77E-05
Fert Başına Telefon Kontör Değeri	2,29E-05
Yeşil Kartlı Kişi Oranı	1,71E-05

Bu aşamada algoritmada da verildiği gibi toplam varyans hesaplanır. Toplam varyans 29,396' dır. İlk 6 değişken için varyans toplamı 27,439 olup 54 değişkenin %93'ünü temsil ettiği için indirgenmiş giriş değişkeni sayısı (size_pc değeri) 6 olarak belirlenir. Algoritmanın son iki aşamasındaki işlemlerle 6x81 boyutundaki indirgenmiş giriş veri seti bulunur.

Çıkış Değişkeni

Y_1 = İlin Sosyo-Ekonomik Gelişmişlik Endeksi

Gelişme olgusu, ülke genelinde toplumsal, siyasal, kültürel ve benzeri kurumlardaki yapısal değişimleri kapsayan çok boyutlu bir öze sahiptir. Bu anlayışla gelişme; ülkenin ekonomik, sosyal, siyasal ve kültürel yapılarındaki ilerlemeyi kapsamakta ve bir bütün oluşturmaktadır. Belirtilen kapsamda bu çalışmaya konu olan sosyo-ekonomik gelişme olgusu da; kişi başına düşen milli gelirin artırılması şeklinde özetlenebilecek iktisadi büyüme kavramıyla beraber, yapısal ve insani gelişmeyi içine alan ve ölçebildiğimiz bütün sosyal değişkenleri içermektedir.

Son elli yıllık kalkınma deneyiminden çıkarılacak başlıca sonuçlar şunlardır:

- Niceliksel büyüme, gelişme için gerekli olmakla birlikte büyüme için gerekli olan makroekonomik istikrar ortamı, sosyal, kültürel, çevresel ve mekansal yapıların da uyumunu gerekli kılmaktadır.
- İnsani ve toplumsal ihtiyaçların karşılanmasında büyümeden ziyade gelişme daha çok gerekli olmaktadır.
- Gelişme sürecinde tek bir politika yerine, geniş kapsamlı bir politika yaklaşımı gerekmektedir.
- Gelişme sürecinde kurumlar önem taşımaktadır.

Tüm bu anlayış farklılıkları, kalkınma düşüncesinde sürdürülebilir gelişme doğrultusunda bir değişim yaşandığını göstermektedir. Sürdürülebilir gelişme ise, iktisadi büyüme yanında, sosyo-kültürel, mekansal, çevresel ve benzeri boyutları içine alan dengeli bir süreci temel almaktadır. Kalkınma düşüncesindeki değişimi aşağıdaki dört öge ile daha geniş bir çerçevede tanımlayabiliriz:

- Sürdürülebilir gelişme, çok boyutlu bir amaç taşımaktadır. Sürdürülebilir gelişme anlayışı beşeri sermayeye yapılan yatırımların ekonomik büyümeyi hızlandıracağını öngörmektedir.
- Gelişme politikaları birbirine bağımlıdır ve bu nedenle bütüncül bir yaklaşımla ele alınması gerekmektedir. Bu süreçte geniş toplumsal katılım, toplumsal motivasyon ve kurumsal ortamlar son derece önem taşımaktadır.
- Sürdürülebilir gelişme sürecinde kamu sektörü önemli bir yere sahiptir. Bununla birlikte kamu strateji ve politikaları, ülkelerin özgün koşullarına göre

değişkenlik göstermektedir. Genel kalkınma politikaları, zamana ve mekana göre biçimlenmektedir.

- Süreçler de politikalar kadar önem taşımaktadır. Diğer bir deyişle; uzlaşma, katılımcılık ve saydam süreçlere dayalı politikaların getirileri daha güçlü ve uzun sürelidir.⁹⁰

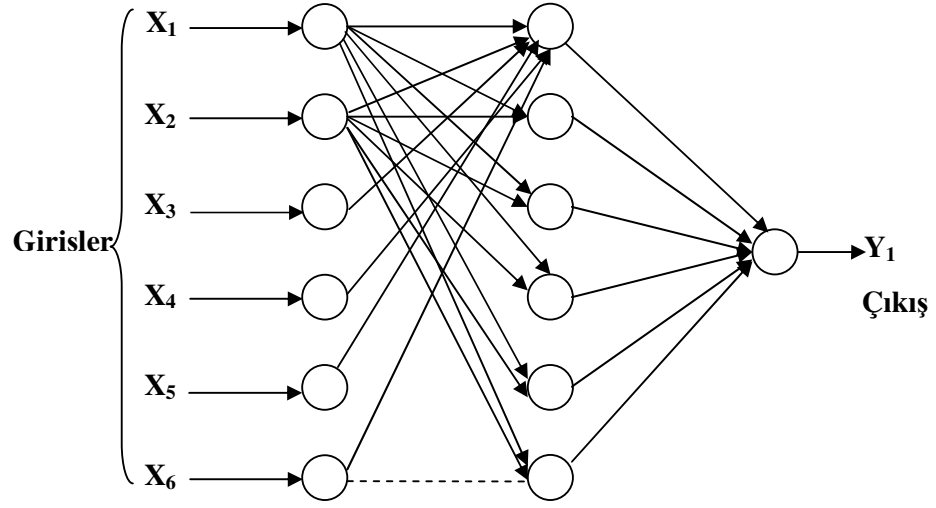
Çıkış değişkeni olarak alınan Y değerleri Ek-2’de verilmiştir.

5.1. Uygulama İçin Yapay Sinir Ağının Tasarımı

Bu uygulama ileri beslemeli bir ağ olan çok katmalı perseptron (MLP) ağı ile yapılmıştır. MLP’ler birçok öğrenme algoritması kullanılarak eğitilebilirler. Bu çalışmada ağı eğitmek için danışmanlı bir öğrenme algoritması olan geri-yayılım (BP: back-propagation), delta-bar-delta (DBD), genişletilmiş delta-bar-delta (EDBD-extended delta-bar-delta) öğrenme algoritmaları kullanılmıştır. Uygulama sonucunda elde edilen çıkış değeri ile gerçek çıkış değerleri arasındaki fark sifıra ya da en aza indirilmeye çalışılmıştır. Bu çıkışlar arasındaki farka göre ağırlık değerleri yeniden hesaplanarak ağa uygulanmıştır. Bu işlem istenilen çıkışlar elde edilinceye kadar tekrarlanmıştır.

Uygulama için geriyayılımlı bir ağ olan Şekil-5.1’deki ağ kullanılmıştır. Bu ağ oluşturulurken de en iyi sonucu üretecek olan katman sayısı şekilde görüldüğü gibi biri gizli katman olmak üzere üç katmandır. Giriş katmanında giriş verisi olarak 6 değişken alındığı için 6 nöron ve çıkışta sadece bir endeks değeri alınacağı için çıkış katmanında 1 nöron bulunmaktadır. Gizli katmandaki nöron sayısı yine en iyi sonucu ürettiği için 6 olarak belirlenmiştir.

⁹⁰ Dinçer ve diğerleri, Age, s.7-8.



Şekil 5.1 Sosyo-Ekonomik gelişmişlik endeks tahmini için yapay sinir ağı.

Bu uygulamada eğitim için kullanılan 54 olan giriş değişkeni sayısı temel bileşen analizi yardımıyla 6'ya indirilmiş ve Tablo-5.2' de gösterilmiştir. Aynı şekilde test için kullanılan giriş değişkeni sayısı 54'den 6'ya indirilmiş ve Tablo-5.3'de verilmiştir.

Tablo 5.2 Uygulamada eğitim için kullanılan giriş verileri.

Sıra No	İl	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆
1	Adana	-3,5683	1,6564	-0,7128	0,13537	0,61365	0,59461
2	Adıyaman	-5,5014	-1,1692	-0,3188	0,10875	0,06316	-0,2314
3	Afyon	-5,1443	0,37465	0,53316	0,82721	-0,2745	-0,5525
4	Ağrı	-5,5127	-2,7214	-0,7843	0,46235	-0,1015	0,07949
5	Amasya	-5,2348	-0,0458	0,18269	0,73207	-0,4812	-0,459
6	Ankara	-5,2348	-0,0458	0,18269	0,73207	-0,4812	-0,459
7	Antalya	-0,2789	4,1071	-1,012	0,02465	2,4379	-0,3026
8	Artvin	-3,858	2,2483	-0,5023	0,90747	0,92145	0,20804
9	Aydın	-5,5357	-1,7771	0,31223	-0,5607	-0,1573	-0,3354
10	Balıkesir	-4,5878	-0,2138	0,78424	-1,9999	0,45013	0,50432
11	Bilecik	-4,4591	1,6541	0,1332	0,52597	0,41598	0,19347
12	Bingöl	-4,2371	1,8427	0,34975	0,65514	0,48482	0,59038
13	Bitlis	-5,2508	-0,1267	0,33282	-0,8782	-0,2242	-0,3814
14	Bolu	-5,3889	-1,917	-1,5283	-0,015	0,05492	0,28343
15	Burdur	-5,7406	-0,7472	0,6298	-0,3097	0,01583	-0,8508
16	Bursa	-4,6718	1,6126	0,3089	-0,8002	-0,8937	0,19321
17	Çanakkale	-5,2166	-2,5315	0,01831	-0,6316	0,43424	0,30163
18	Çankırı	-5,4854	-2,3427	-0,6085	0,28469	0,0798	-0,1788
19	Çorum	-4,2219	2,1161	1,2558	-0,0557	0,60924	0,89151
20	Denizli	-4,9264	1,654	0,59191	0,22694	0,12806	-0,0446
21	Diyarbakır	-2,9444	2,7788	-0,6472	0,2845	-0,5204	0,50488
22	Edirne	-4,7093	1,7052	0,38337	-0,1128	-0,0773	0,22124
23	Elazığ	-5,501	-0,1136	0,36663	-0,0077	-0,2334	-0,8765
24	Erzincan	-5,0795	0,00814	0,53402	0,10565	0,11963	-0,1216
25	Erzurum	-4,0344	1,8447	-0,4352	-0,0166	-0,2013	0,10618
26	Eskişehir	-4,8571	-1,72	-1,1219	0,26535	0,56772	0,57832
27	Gaziantep	-5,3514	0,31588	0,04834	-0,6567	-0,6266	-0,6673
28	Giresun	-4,1995	1,7032	0,35059	-0,5218	0,02364	0,32534
29	Gümüşhane	-4,9618	0,22421	-0,2338	-0,0651	0,68611	0,04275
30	Hakkâri	-5,3294	-0,2662	0,34036	-0,4338	0,34817	-0,3907
31	Hatay	-4,9641	-0,9299	0,02264	-0,0997	0,58118	-0,1454
32	Isparta	-3,7331	2,8476	-0,0528	-0,4898	0,49883	0,49234
33	Mersin	-4,1994	1,1619	-1,1454	0,82417	-0,4763	0,3935
34	İstanbul	-4,6575	-0,7047	0,55373	-0,3032	0,41589	-0,7864
35	İzmir	-5,5991	-1,3166	0,73185	-0,2639	0,3016	-0,8849
36	Kars	-5,4162	-2,4653	-1,2551	-0,1746	0,07571	0,66329
37	Kastamonu	-4,6629	0,7428	-0,1186	0,55644	-0,4552	0,27655
38	Kayseri	-5,1144	-1,6308	-0,7227	-0,0176	-0,2199	0,30742
39	Kırklareli	-4,7962	1,4329	0,66533	0,24049	0,33019	-0,5328
40	Kırşehir	1,6104	3,6504	-3,3972	-0,4807	0,23959	-2,4291
41	Kocaeli	-1,607	3,589	-0,8011	0,95253	0,36606	0,58979
42	Konya	-4,744	0,83138	-0,5612	-1,4608	0,16631	0,07175
43	Kütahya	-4,9399	0,95935	0,58089	0,49364	-0,3273	0,12139
44	Malatya	-5,4281	-1,1895	0,04725	-0,3115	0,10589	-0,1552
45	Manisa	-4,941	-0,3405	0,73435	-0,542	0,58234	0,03491
46	K.Maraş	-4,4158	1,5125	-0,274	0,12555	-0,0921	0,1155
47	Mardin	-5,2321	0,03635	-0,3332	-0,3918	-0,4146	0,12536
48	Muğla	-4,8535	0,57676	-0,402	-0,4785	-0,2247	-0,1444
49	Muş	-3,9647	2,3433	0,21681	-0,4832	-1,3462	0,434
50	Nevşehir	-5,1833	0,56723	0,57346	0,22827	-0,0642	-0,3786
51	Niğde	-5,0138	-0,2613	-0,086	0,26875	-0,2257	0,01587

52	Ordu	-2,4846	2,2322	-0,6582	-1,1092	-1,4335	0,21102
53	Rize	-4,0695	0,84205	0,75812	2,2221	0,13896	0,15234
54	Sakarya	-5,2685	0,71522	0,54189	0,14102	-0,246	-0,495
55	Samsun	-5,0753	0,00716	-0,2519	0,0815	0,5257	-0,1295
56	Siirt	-4,4652	1,4234	0,20291	0,81205	-0,0014	0,3295
57	Sinop	-5,5292	-1,8951	-0,7738	0,46905	-0,2624	0,00325
58	Sivas	-4,1168	1,2709	-0,3127	0,7565	-0,2465	0,1672
59	Tekirdağ	-4,1019	2,267	0,12862	0,11974	0,5655	0,32751
60	Tokat	-5,5181	-3,1111	-0,3417	0,50075	-0,298	0,05594
61	Trabzon	-5,0221	0,97693	0,46861	0,42143	-0,3212	-0,1875
62	Tunceli	-5,2215	0,21614	0,58654	0,47221	-0,6055	-0,2657
63	Şanlıurfa	-5,1863	-1,071	0,23058	-0,041	0,33298	-0,3198
64	Uşak	-5,4597	0,2915	-0,5841	0,03701	-0,1663	-0,3769
65	Van	-5,0822	0,17504	-0,0409	-0,6808	0,30979	-0,2113
66	Yozgat	-4,3087	1,5673	0,36481	0,42836	-0,4497	0,35464
67	Zonguldak	-4,192	0,31127	0,69329	0,51694	0,69896	0,36836
68	Aksaray	-5,268	-1,6735	-0,6574	1,0835	0,21366	0,37135
69	Bayburt	-5,3676	-2,1028	-0,9474	-0,0828	0,14649	0,70934
70	Karaman	-5,1757	-0,3826	0,78068	-0,6183	0,37318	-0,3224
71	Kırıkkale	-5,1594	-2,7036	-1,517	0,14401	-0,1412	0,80946
72	Batman	-4,7299	-0,4566	1,0493	0,23368	0,31513	-0,0259
73	Şırnak	-3,7953	2,2172	-0,3036	-0,5077	-1,7312	0,61403
74	Bartın	-5,4777	-0,4987	0,47264	0,46262	-0,11	-0,68
75	Ardahan	-4,9953	-0,0752	0,14775	-0,1933	0,76335	-0,6416
76	Iğdır	-4,7434	-0,7366	0,18318	-2,2696	1,0253	0,89572
77	Yalova	-4,796	1,2513	-0,1546	-0,2128	-0,1739	0,03875
78	Karabük	-5,2729	-2,4661	-0,843	0,46531	0,0642	0,25239
79	Kilis	-4,2623	2,1881	-0,9276	-1,0792	-0,4996	-0,1003
80	Osmaniye	-5,5836	-0,6347	0,49458	0,54383	-0,3448	-0,9009
81	Düzce	-4,0175	1,0007	0,18319	-0,8813	-0,3517	0,2208

Tablo 5.3 Uygulamada test için kullanılan giriş verileri.

Sıra No	İl	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆
1	Adana	-3,6	1,3393	-0,5942	0,5954	-0,275	0,06742
2	Adıyaman	-5,6074	-1,0084	-0,6582	-0,0058	-0,0418	-0,0209
3	Afyon	-5,1929	0,28949	-0,2918	-0,1127	0,16908	0,06308
4	Ağrı	-5,4739	-2,4736	-0,4809	-0,2222	0,02036	-0,1066
5	Amasya	-5,3969	-0,0236	-0,1489	-0,2021	0,0395	0,01325
6	Ankara	-5,1989	0,53345	0,44363	0,03896	0,32966	0,26621
7	Antalya	0,27553	3,8327	0,1687	2,3569	-1,4001	1,1941
8	Artvin	-3,5497	2,2641	-0,6031	-0,1185	-1,1947	0,08239
9	Aydın	-5,8917	-1,6839	1,1095	0,57726	0,15793	-0,5535
10	Balıkesir	-5,313	-0,0175	1,092	0,38519	0,2173	-0,7585
11	Bilecik	-4,1829	1,9508	-0,1653	-0,1794	-0,5707	0,33104
12	Bingöl	-4,3727	1,9293	0,25468	-0,1425	-0,0946	0,39933
13	Bitlis	-5,5445	0,02389	0,11538	-0,1295	0,40222	-0,8675
14	Bolu	-5,3621	-1,8647	-1,3573	-0,0133	-0,5809	-0,8235
15	Burdur	-5,67	-1,0582	0,69804	0,5559	0,5623	0,30389
16	Bursa	-4,4333	1,8022	0,08536	-0,9858	0,19771	-0,1947
17	Çanakkale	-5,2877	-2,4452	0,23163	-0,1325	0,13105	-0,0677
18	Çankırı	-5,6283	-2,0894	-0,7707	0,20622	0,03503	0,28169
19	Çorum	-4,8171	1,1668	0,29016	-0,1704	0,17473	0,21868
20	Denizli	-4,9017	1,324	0,44403	0,10921	0,12823	0,60872
21	Diyarbakır	-2,7596	2,5931	-0,9859	-0,1059	0,18337	-0,0793
22	Edirne	-4,6596	1,7789	0,55502	-0,4466	-0,0367	0,35164
23	Elazığ	-5,2834	-0,2835	0,35523	-0,3051	0,07915	-0,0172
24	Erzincan	-5,3832	-0,0076	0,23545	-0,2584	0,52324	0,65644
25	Erzurum	-4,2212	1,6078	-0,3485	-0,1841	-0,5661	-0,7147
26	Eskişehir	-5,1831	-1,4107	-1,196	0,16027	-0,6111	-0,4979
27	Gaziantep	-4,276	1,5922	0,59168	0,26342	-0,8279	-0,0266
28	Giresun	-4,5774	0,47677	0,03723	-0,0362	-0,3443	-0,0871
29	Gümüşhane	-5,2056	-0,1018	0,64852	0,38038	0,29675	0,1285
30	Hakkâri	-5,2129	-0,9279	0,24999	0,53529	0,533	0,19183
31	Hatay	-3,6346	2,511	-0,1367	0,38663	-0,1349	-0,3738
32	Isparta	-4,2782	0,56991	-1,44	0,02768	-0,1171	-0,2871
33	Mersin	-4,5934	-0,4036	0,88797	0,76382	-1,3652	0,05111
34	İstanbul	-5,6182	-1,0026	0,86542	0,511	0,29085	-0,0088
35	İzmir	-5,3324	-2,4229	-1,3119	0,07695	-0,1245	0,82085
36	Kars	-4,649	0,73001	-0,4905	-0,0688	0,33257	0,0112
37	Kastamonu	-5,8501	-1,3968	0,69775	0,46793	0,2213	-0,0465
38	Kayseri	-4,7691	1,6071	-0,2139	0,16595	0,08818	0,43776
39	Kırklareli	2,2951	3,8895	-2,117	2,3578	1,9628	-0,837
40	Kırşehir	-0,4792	3,3404	-0,7146	0,10504	-0,5692	0,18644
41	Kocaeli	-4,974	0,23478	0,45797	-0,2921	-0,2089	-0,5555
42	Konya	-5,5867	-0,8988	0,83081	0,48802	0,07947	-0,2697
43	Kütahya	-5,1628	-0,1505	0,76966	-0,1847	-0,2386	-0,2885
44	Malatya	-4,1957	1,4978	-0,4835	0,33662	0,48532	0,38404
45	Manisa	-4,8192	1,058	0,28592	0,34371	0,28351	-0,597
46	K.Maraş	-4,4054	2,1924	0,47839	-0,4724	0,26155	0,37336
47	Mardin	-5,1406	0,80832	0,59857	0,24223	0,26321	0,46373
48	Muğla	-5,3124	-0,2769	-0,1848	-0,1073	-0,0475	-0,541
49	Muş	-2,6915	2,6373	-0,9514	-1,4517	0,78356	-0,1919
50	Nevşehir	-4,5827	0,71193	-0,3468	-0,1302	-0,0349	-0,1603
51	Niğde	-4,945	0,96321	0,25123	-0,0639	0,42384	0,28468
52	Ordu	-4,8989	0,2143	0,02504	0,26399	-0,2134	-0,8302

53	Rize	-4,5234	1,28	-0,207	-0,2145	-0,2028	0,1348
54	Sakarya	-5,6029	-1,6174	-1,0741	0,11993	-0,1819	0,36908
55	Samsun	-4,1976	1,556	-0,9123	-0,0751	-0,3636	-0,3972
56	Siirt	-3,9281	2,033	0,46678	-0,9783	-0,9023	-0,2512
57	Sinop	-5,8486	-2,5136	-0,4848	0,10584	0,40262	0,6035
58	Sivas	-5,0175	1,0208	0,35368	-0,2984	0,41507	0,86362
59	Tekirdağ	-5,2671	0,44143	0,08827	-0,4559	0,39638	0,71106
60	Tokat	-5,5369	-0,3697	-0,1573	0,32249	-0,1429	-0,6143
61	Trabzon	-4,8931	0,48795	1,2413	0,07814	-0,2632	-0,2919
62	Tunceli	-4,8225	1,1677	-0,1425	-0,2155	0,12341	-0,1434
63	Şanlıurfa	-4,8573	0,27564	0,03626	0,15973	-0,1542	-0,457
64	Uşak	-5,4408	-2,1452	-0,5694	0,41867	-0,0174	0,36315
65	Van	-5,4468	0,06152	-0,1871	-0,4784	0,08812	0,46771
66	Yozgat	-5,2075	-0,3714	0,49443	0,27016	0,43694	0,1698
67	Zonguldak	-5,2913	-1,1646	-1,2482	-0,1164	-0,6618	-0,0907
68	Aksaray	-5,3516	-2,9389	-1,5078	0,04724	-0,7216	0,11184
69	Bayburt	-3,6374	2,3776	-0,1231	-1,2881	0,05763	-0,0116
70	Karaman	-5,5271	-0,2565	0,05239	0,06513	0,58095	0,44567
71	Kırıkkale	-4,9245	0,48043	0,63242	0,55784	-0,1661	-0,193
72	Batman	-5,432	-1,0087	1,2831	0,79617	-0,1211	-0,7286
73	Şırnak	-4,5862	1,3231	-0,1961	-0,1413	0,2711	0,10614
74	Bartın	-5,55	-1,9675	-1,0908	0,04654	0,09686	0,50226
75	Ardahan	-5,6973	-0,3997	0,30151	-0,1054	0,49285	0,62824
76	Iğdır	-4,4633	1,1193	0,25919	-0,506	0,44354	-0,9645

Bu ağ için temel bileşenler analizi ile 6'ya indirilen giriş değerleri ve çıkış değerleri -1 ile +1 arasında ölçeklendirilmiştir. Ölçekleme iki farklı yöntemle yapılmıştır. İlki Excelde oluşturulan aşağıdaki formülle yapılmıştır. İkincisi ise Matlab programıyla yapılmıştır. Bu şekilde farklı ölçekleme yöntemleri kullanılarak ölçekleme olayının da sonucu doğrudan etkilediği görülmüştür.

$$\text{Yeni deęer} = \text{Yeni deęerlerin bař. deęeri} + (\text{Eski deęer} - \text{MİN}(\text{Eski deęerler})) / (\text{MAK}(\text{Eski deęerler}) - \text{MİN}(\text{Eski deęerler})) * (\text{MAK}(\text{Yeni deęerler}) - \text{MİN}(\text{Yeni deęerler})) \quad (5.1)$$

Ağırlık deęerleri bařlangıçta rasgele seçilmiştir ve ağırlık deęerleri kullanılan öğrenme algoritması ile en uygun çıkışı elde edebilecek şekilde yeniden düzenlenmiştir. Öğrenme hızını ve doğruluk derecesini etkileyen öğrenme katsayısı ve momentum katsayısı için birçok farklı deęer alınarak yapay sinir ağının eğitimi ve testi yapılmıştır.

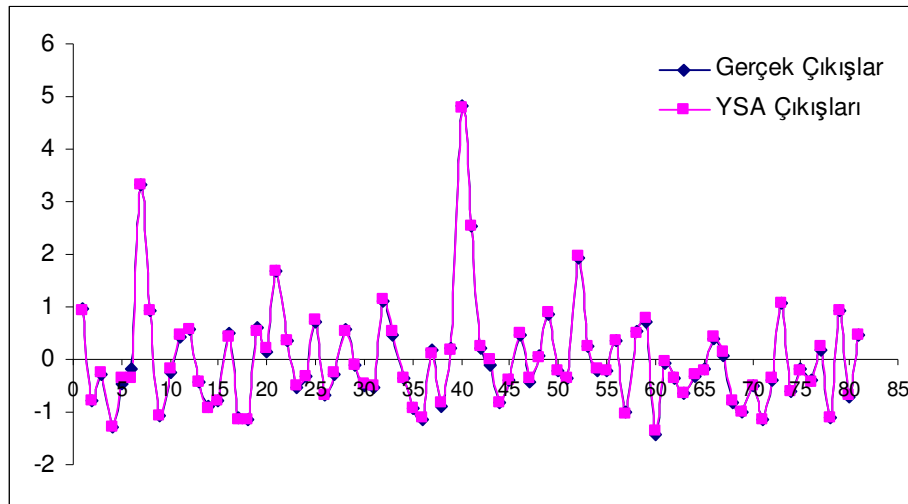
Aynı uygulama birkaç farklı geriyayılımlı öğrenme algoritması kullanılarak yapılmıştır. Burada kullanılan geriyayılımlı öğrenme algoritmaları traingd, traingdm ve traingdx'dir.

Eğitim *Traingd* (Gradyant Azaltma) öğrenme algoritması ile farklı öğrenme katsayıları kullanılarak yapılmıştır. Eğitimde kullanılan her öğrenme katsayısına karşılık gelen yapay sinir ağının üretmiş olduğu çıkışlar ile beklenen çıkışlar arasındaki ortalama mutlak hatalar Tablo-5.4’de verilmiştir.

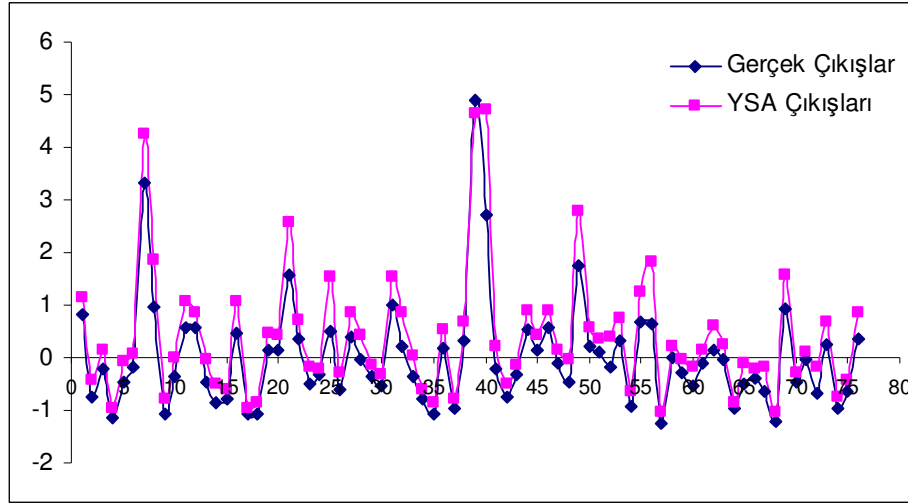
Tablo 5.4 Traingd algoritması ile elde edilen mutlak hatalar

Öğrenme Katsayısı	Ortalama Mutlak Hata	
	Eğitim	Test
0,01	0,02332	0,20873
0,1	0,01049	0,14701
0,2	0,00967	0,13462
0,3	0,00959	0,13111
0,4	0,01195	0,11992
0,5	0,01635	0,14347

En iyi sonuç öğrenme katsayısı 0.3 alındığı zaman elde edilmiştir. Eğitim seti için gerçek çıkışlarla yapay sinir ağının üretmiş olduğu eğitim çıkışları arasındaki ilişki Şekil-5.2’ de, test seti gerçek çıkışları ve YSA test çıkışları arasındaki ilişki ise Şekil-5.3’de gösterilmiştir.



Şekil 5.2 Gerçek çıkışlar ile YSA eğitim çıkışlarının karşılaştırılması.



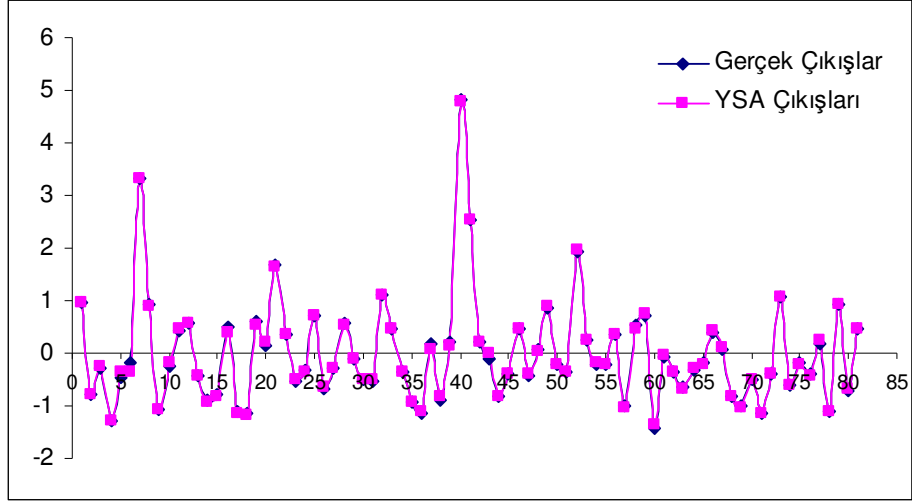
Şekil 5.3 Gerçek çıkışlar ile YSA test çıkışlarının karşılaştırılması.

Eğitim *Traindm* (Momentumlu Gradyant Azaltma) öğrenme algoritması ile farklı öğrenme katsayıları kullanılarak yapılmıştır. Eğitimde kullanılan her öğrenme katsayısına karşılık gelen yapay sinir ağının üretmiş olduğu çıkışlar ile beklenen çıkışlar arasındaki ortalama mutlak hatalar Tablo-5.5’de verilmiştir.

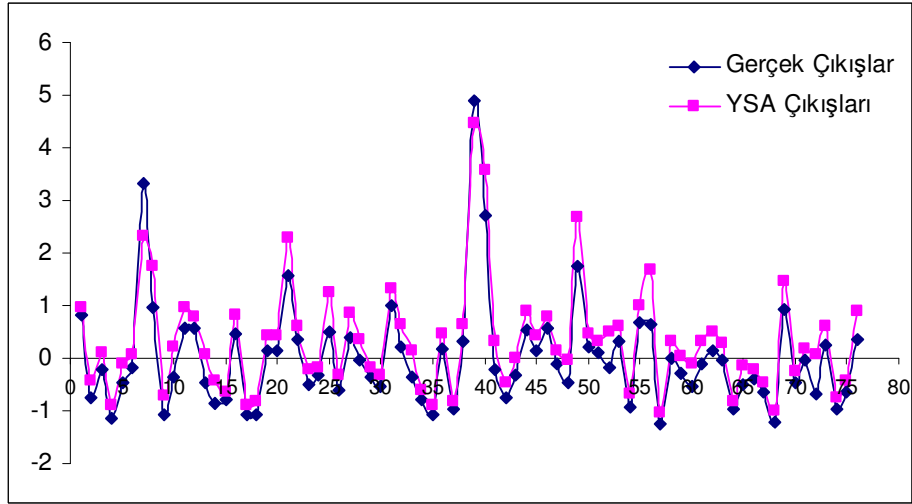
Tablo 5.5 Traindm algoritması ile elde edilen mutlak hatalar

Öğrenme Katsayısı	Momentum Katsayısı	Ortalama Mutlak Hata	
		Eğitim	Test
0,4	0,9	0,00852	0,12185
	0,7	0,00922	0,13095
0,3	0,9	0,00878	0,12030
	0,7	0,00943	0,13306
0,2	0,5	0,00894	0,19466
	0,9	0,00968	0,1267

Traindm öğrenme algoritması kullanarak eğitim yapıldığında en iyi sonuç öğrenme katsayısı 0.3 ve momentum katsayısı tüm katmanlar için 0.9 alındığı zaman elde edilmiştir. Eğitim seti için gerçek çıkışlarla yapay sinir ağının üretmiş olduğu eğitim çıkışları arasındaki ilişki Şekil-5.4’ de, test seti gerçek çıkışları ve YSA test çıkışları arasındaki ilişki ise Şekil-5.5’de gösterilmiştir.



Şekil 5.4 Gerçek çıkışlar ile YSA eğitim çıkışlarının karşılaştırılması.



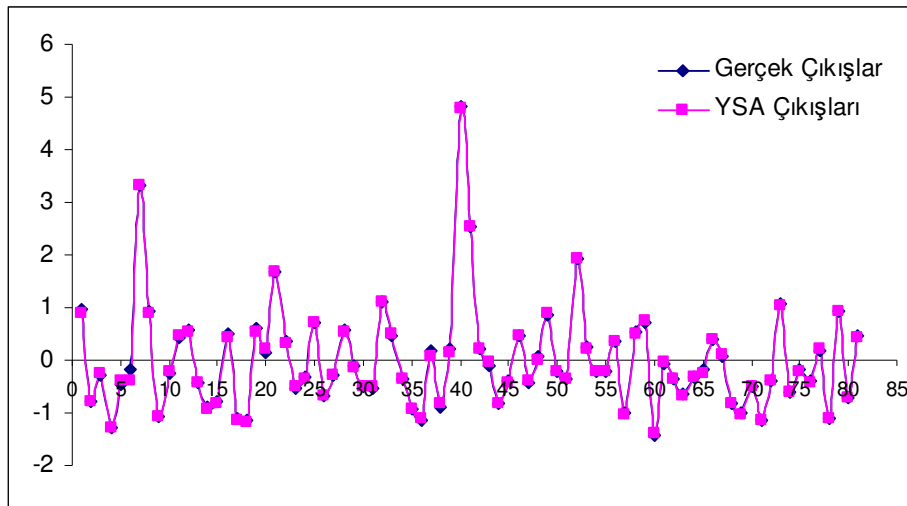
Şekil 5.5 Gerçek çıkışlar ile YSA test çıkışlarının karşılaştırılması.

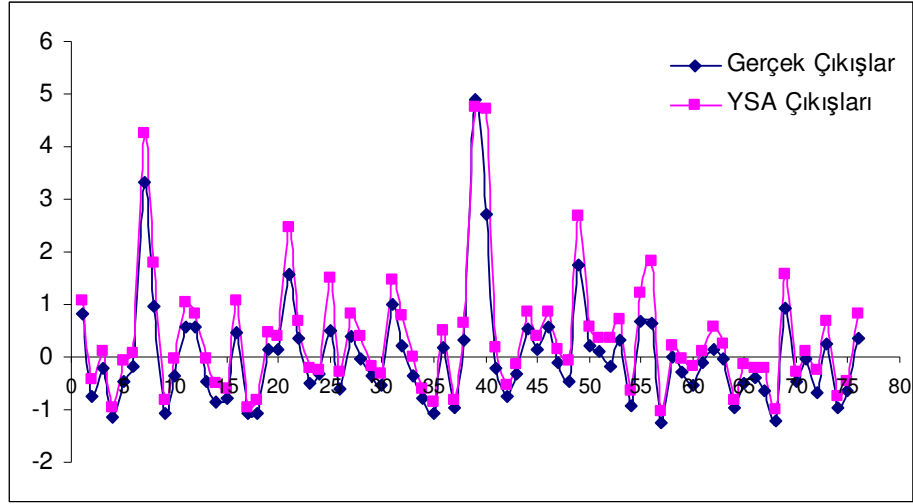
Eğitim *Trainidx* (Adaptif Öğrenme Oranı) öğrenme algoritması ile farklı öğrenme katsayıları kullanılarak yapılmıştır. Eğitimde kullanılan her öğrenme katsayısına karşılık gelen yapay sinir ağının ürettiği çıkışlar ile beklenen çıkışlar arasındaki ortalama mutlak hatalar Tablo-5.6'de verilmiştir.

Tablo 5.6 Traingdx algoritması ile elde edilen mutlak hatalar

Öğrenme Katsayısı	Momentum Katsayısı	Azaltma faktörü	Artırma faktörü	Mutlak Hata	
				Eğitim	Test
0,01	0,9	1,05	0,7	0,00822	0,14082
0,2				0,00814	0,14703
0,3				0,00802	0,13810
0,5				0,00814	0,12946
0,3	0,7			0,00866	0,12781
0,3	0,5			0,00939	0,13037
0,01	0,7			0,00822	0,02496
0,01	0,5			0,00857	0,12733
	0,3			0,00908	0,12485
0,01	0,5			1,00	0,7
	0,3	1,05	0,6	0,00877	0,12793
			0,8	0,00889	0,12615

Traingdx öğrenme algoritması ile eğitim yapılırken öğrenme katsayısı 0.01, öğrenme katsayısı azaltma faktörü 0.7, artırma faktörü 1.05 ve momentum katsayısı 0.3 alındığı zaman en iyi sonuç elde edilmiştir. Eğitim seti için gerçek çıkışlarla yapay sinir ağının üretmiş olduğu eğitim çıkışları arasındaki ilişki Şekil-5.6' de, test seti gerçek çıkışları ve YSA test çıkışları arasındaki ilişki ise Şekil-5.7' de gösterilmiştir.

**Şekil 5.6** Gerçek çıkışlar ile YSA eğitim çıkışlarının karşılaştırılması.



Şekil 5.7 Gerçek çıkışlar ile YSA test çıkışlarının karşılaştırılması.

Her öğrenme algoritması ile elde edilen eğitim ve test için bulunan ortalama mutlak hatalar Tablo-5.7’te ve hata karelerinin ortalaması Tablo-5.8’de verilmiştir.

Tablo 5.7 Ortalama mutlak hata değerleri.

Öğrenme Algoritması	Ortalama Mutlak Hata	
	Eğitim	Test
Traingd	0,00959	0,13111
Traingdm	0,00878	0,12030
Traingdx	0,00908	0,12485

Tablo 5.8 Hata Kareler Ortalaması değerleri.

Öğrenme Algoritması	Hata Karelerinin Ortalaması	
	Eğitim	Test
Traingd	0,000180	0,026109
Traingdm	0,000175	0,018919
Traingdx	0,000168	0,024369

5.2. Sonuç

Yapılan bu çalışma ile illerin sosyo-ekonomik gelişmişlik endeksi belirlenmeye çalışılmıştır. Öncelikle bu endeksi belirleyen 54 giriş değişkeni sayısı temel bileşenler analizi ile 6'ya indirilmiş ve bu veri seti, seçilen yapay sinir ağına sunulmuştur.

Birçok yapay sinir ağı modeli arasından çok kullanışlı ve yaygın olarak kullanılması sebebiyle bu çalışma için ileri beslemeli-geriyayılımlı bir yapay sinir ağı seçilmiştir. Bu seçimin çıkışlar üzerinde doğrudan etkisi olduğu görülmüştür.

Çalışma esnasında seçilen ağ türünün yanı sıra çıkışların doğruluğunu etkileyen birçok faktör olduğu görülmüştür. Bunlar; öğrenme katsayısı, momentum katsayısı, ara katman sayısı, ara katmanlardaki nöron sayısı ve iterasyon sayısıdır. Sonuçta bu uygulama için en uygun sonucu elde edilecek şekilde bu faktörlerin değerleri belirlenmiş ve çalışmada sunulmuştur.

Şekil 5.2, Şekil 5.3, Şekil 5.4, Şekil 5.5, Şekil 5.6, Şekil 5.7'deki grafiklerle Tablo-5.6'daki ortalama mutlak hatalar ve Tablo-5.7'deki hata kareler ortalamasından da görüldüğü gibi çıkışların doğruluğunun kabul edilebilir olduğu da söylenebilir.

Traingd öğrenme algoritmasına göre en küçük hata kareler ortalaması eğitim verileri için 0,000180, test verileri için 0,026109' dir.

Traingdm öğrenme algoritmasına göre en küçük hata kareler ortalaması eğitim verileri için 0,000175, test verileri için 0,018919' dir.

Traingdx öğrenme algoritmasına göre en küçük hata kareler ortalaması eğitim verileri için 0,000168, test verileri için 0,024369' dir.

Bu çalışma ile illerin sosyo-ekonomik gelişmişlik endeksinin çeşitli istatistikî analizler sonucunda elde edilebileceği gibi yapay sinir ağları kullanarak da bulunabileceği gösterilmiştir. Bu sayede çok fazla işlem yapılmaksızın alınan örnek veri seti yardımıyla kolayca çıkış tahmin edilebilir.

KAYNAKÇA

- Dayhoff, Judith E. *Neural Network Architectures An Introductio*. NewYork: Van Nostrand Reinhold, 1990.
- Dinçer, Bülent, Metin Özaslan ve Erdoğan Satılmış. *İllerin Sosyo-Ekonomik Gelişmişlik Sıralaması Araştırması (1996)*. Ankara: DPT. Bölgesel Gelişme ve Yapısal Uyum Genel Müdürlüğü, 1996.
- Efe, M. Önder ve Okyay Kaynak. *Yapay Sinir Ağları ve Uygulamaları*. İstanbul: Boğaziçi Üniversitesi, 2000.
- Elmas, Çetin. *Yapay Sinir Ağları*. Ankara: Seçkin Yayıncılık, 2003.
- Fausett, Laurene. *Fundamentals of Neural Networks*. New Jersey: Prentice-Hall, 1994.
- Haykin, Simon, *Neural Network, A Comprehensive Foundation*. New York: Macmillan College Publishing Company, 1994.
- Kaymak, Cemalettin, Rasim Akpınar ve Ahmet Kındap, *İller ve Bölgeler İtibariyle Gayri Safi Yurtiçi Hâsıladaki Gelişmeler (1987-2000)*. Ankara: Devlet İstatistik Enstitüsü Matbaası, 2003.
- Kartalopoulos, Stamatios V. *Understanding Neural Networks and Fuzzy Logic*. NewYork: The Institute of Electrical and Electronics Engineers Press,
- Kung, S.Y. ve K.I. Diamantaras. *Principal Component Neural Networks*. Theory and Applications, NewYork: Jhon Wiley & Sons, 1996.
- Lau, Clifford. *Artificial Neural Networks. Paradigms, Applications and Implementations*. (Çev. Edgar Sánchez-Sinencio), NewYork: The Institute of Electrical and Electronics Engineers Press, 1992.
- Lippmann, Richard. "An Introduction to Computing with Neural Nets", 1987.
- Mehrota, Kishan, MOHAN, Chilukuri. K. ve RANKA Sanjay. *Elements of Artificial Neural Networks*. London: Massachusetts Institute of Tecnology, 1997.
- Özdamar, Kazım, *Paket Programlar İle İstatistiksel Veri Analizi-2*, Eskişehir: Kaan Kitabevi, 2002.

Öztemel, Ercan. *Yapay Sinir Ağları*. İstanbul: Papatya Yayıncılık, 2003.

Picton, Phil. *Intoduction to Neural Networks*. London: Macmillan Press, 1994.

Sağiroğlu, Şeref, BEŞDOK, Erkan ve ERLER, Mehmet. *Mühendislikte Yapay Zeka Uygulamaları I: Yapay Sinir Ağları*. Kayseri: Ufuk Kitap Kırtasiye, 2003.

Simpson Patrick K. *Neural Networks Theory, Technology and Applications*. NewYork: The Institute of Electrical and Electronics Engineers Press, 1990.

Şen, Zekai. *Yapay Sinir Ağları ve İlkeleri*, İstanbul: Su Vakfı Yayınları, 2004.

“Yapay Sinir Ağları ve Uygulamaları.” 2000, http://batitrakya.dostweb.com/yapay-siniraglari/ek_dokuman.htm. (24.11.2005)

Yurtoğlu, Hasan. *Yapay Sinir Ağları Metodolojisi İle Öngörü Modellemesi: Bazı Makroekonomik Değişkenler İçin Türkiye Örneği*, Ankara: DPT Yayınları, 2005.

EKLER

EK-1. İllerin Gelişmişlik Performansları - Sivas İli Örneği

SİVAS İLİ GELİŞMİŞLİK PERFORMANSI

Sosyo-ekonomik Gelişmişlik Sıralaması (2003) (81 il içinde)53						
DEĞİŞKEN	YIL	BİRİM	SİVAS	İÇ ANADOLU BÖLGESİ	TÜRKİYE	SIRA (81 il içinde)
DEMOGRAFIK GÖSTERGELER						
1 Toplam Nüfus	2000	Kişi	755.091	11.608.868	67.803.927	29
2 Şehirleşme Oranı	2000	Yüzde	55,86	69,25	64,90	39
3 Yıllık Ortalama Nüfus Artış Hızı	1990-2000	Binde	-1,54	15,78	18,28	69
4 Nüfus Yoğunluğu	2000	Kişi/Km2	26,45	63	88	78
5 Doğurganlık Hızı	2000	Çocuk Sayısı	2,76	2,54	2,53	33
6 Ortalama Hanehalkı Büyüklüğü	2000	Kişi	5,27	4,42	4,50	24
İSTİHDAM GÖSTERGELERİ						
7 Tarım İşkolunda Çalışanların Toplam İstihdama Oranı	2000	Yüzde	66,46	46,81	48,38	27
8 Sanayi İşkolunda Çalışanların Toplam İstihdama Oranı	2000	Yüzde	5,43	10,55	13,35	51
9 Ticaret İşkolunda Çalışanların Toplam İstihdama Oranı	2000	Yüzde	5,09	8,90	9,67	55
10 Mali Kurumlar İşkolunda Çalışanların Toplam İstihdama Oranı	2000	Yüzde	1,27	4,01	3,11	51
11 Ücretli Çalışanların Toplam İstihdama Oranı	2000	Yüzde	28,46	45,07	43,52	54
12 Ücretli Çalışan Kadınların Toplam İstihdama Oranı	2000	Yüzde	3,31	8,52	8,81	58
13 İşverenlerin Toplam İstihdama Oranı	2000	Yüzde	1,31	2,53	2,61	48
EGİTİM GÖSTERGELERİ						
14 Okur Yazar Nüfus Oranı	2000	Yüzde	85,40	90,32	87,30	52
15 Okur Yazar Kadın Nüfusun Toplam Kadın Nüfusuna Oranı	2000	Yüzde	77,52	84,96	80,62	53
16 Üniversite Bitirenlerin Okul Bitirenlere Oranı	2000	Yüzde	6,06	10,31	8,42	37
17 İlköğretim Okullaşma Oranı	2000-2001	Yüzde	83,56	92,95	98,01	63
18 Liseler Okullaşma Oranı	2000-2001	Yüzde	30,01	41,58	36,92	52
19 Meslek Liseleri Okullaşma Oranı	2000-2001	Yüzde	20,61	21,69	20,49	37
SAGLIK GÖSTERGELERİ						
20 Bebek Ölüm Oranı	2000	Binde	53,00	41,77	43,00	12
21 Onbin Kişiyeye Düşen Hekim Sayısı	2000	Kişi	11	17	13	20
22 Onbin Kişiyeye Düşen Diş Hekimi Sayısı	2000	Kişi	0,3	2,9	2,2	73
23 Onbin Kişiyeye Düşen Eczane Sayısı	2000	Adet	1,5	3,1	2,9	57
24 Onbin Kişiyeye Düşen Hastane Yatağı Sayısı	2000	Hastane Yatağı	30	26	23	11
SANAYİ GÖSTERGELERİ						
25 Organize Sanayi Bölgesi Parsel Sayısı	2000	Parsel	202	16.399	28.726	26
26 Küçük Sanayi Sitesi İşyeri Sayısı	2000	Adet	1.586	17.106	81.302	15
27 İmalat Sanayi İşyeri Sayısı	2000	Adet	16	1.620	11.118	57
28 İmalat Sanayi Yıllık Çalışanlar Ortalama Sayısı	2000	Kişi	4.314	148.070	1.130.488	42
29 İmalat Sanayii Kurulu Güç Kapasite Miktarı	2000	Beygir Gücü	35.716	1.711.732	13.478.078	48
30 Fert Başına İmalat Sanayi Elektrik Tüketimi	2000	Kws	108	348	550	55
31 Fert Başına İmalat Sanayi Katma Değeri	2000	Milyon TL.	48	251	350	54
TARIM GÖSTERGELERİ						
32 Kırsal Nüfus Başına Tarımsal Üretim Değeri	2000	Milyon TL.	1.137	1.410	1.124	37
33 Tarımsal Üretim Değerinin Türkiye İçindeki Payı	2000	Yüzde	1,42	18,81	100,00	23
İNŞAAT GÖSTERGELERİ						
34 Daire Sayısı	2000	Adet	117.149	2.783.819	16.235.830	34
35 Borulu Su Tesisatı Bulunan Daire Oranı	2000	Yüzde	98	96	97	19
MALİ GÖSTERGELER						
36 Gayri Safi Yurt İçi Hasıla İçindeki Payı	2000	Yüzde	0,67	16,96	100,00	37
37 Fert Başına Gayri Safi Yurt İçi Hasıla	2000	Milyon TL.	1.098	1.820	1.837	53
38 Banka Şube Sayısı	2000	Adet	57	1266	7786	27
39 Fert Başına Banka Mevduatı	2000	Milyon TL.	276	1.347	939	44
40 Toplam Banka Mevduatı İçindeki Payı	2000	Yüzde	0,33	24,57	100,00	26
41 Toplam Banka Kredileri İçindeki Payı	2000	Yüzde	0,28	21,29	100,00	30
42 Kırsal Nüfus Başına Tarımsal Kredi Miktarı	2000	Milyon TL.	118	192	138	17
43 Fert Başına Sınai, Ticari Ve Turizm Kredileri Miktarı	2000	Milyon TL.	57	489	392	57
44 Fert Başına Belediye Giderleri	2000	Milyon TL.	42	90	82	62
45 Fert Başına Genel Bütçe Gelirleri	2000	Milyon TL.	80	449	464	45
46 Fert Başına Gelir Ve Kurumlar Vergisi Miktarı	2000	Milyon TL.	55	198	165	35
47 Fert Başına Kamu Yatırımları Miktarı	1995-2000	Milyon TL.	682	276	248	4
48 Fert Başına Teşvik Belgelii Yatırım Tutarı	1995-2000	Milyon TL.	724	2.002	2.668	52
49 Fert Başına İhracat Miktarı	1995-2000	ABD Doları	7	407	2.249	57
50 Fert Başına İthalat Miktarı	1995-2000	ABD Doları	80	1.345	3.967	43
ALTYAPI GÖSTERGELERİ						
51 Kırsal Yerleşmelerde Asfalt Yol Oranı	2000	Yüzde	31,39	60,74	45,23	57
52 Yeterli İçme Suyu Götürülen Nüfus Oranı	2000	Yüzde	96,17	94,90	84,98	19
53 TCK Asfalt Yol Oranı	2000	Yüzde	90,17	95,80	91,28	52
DİĞER GÖSTERGELER						
54 Onbin Kişiyeye Düşen Özel Otomobil Sayısı	2000	Adet	303	884	652	54
55 Onbin Kişiyeye Düşen Motorlu Kara Taşıtı Sayısı	2000	Adet	472	1.288	1.056	60
56 Fert Başına Elektrik Tüketim Miktarı	2000	Mws	0,7	1,1	1,4	56
57 Fert Başına Telefon Kontür Değeri	2000	Adet	993	1.860	1.852	55
58 Yeşil Kartlı Kişi Oranı	2000	Yüzde	21	14	15	34

EK-2. İllerin Sosyo-Ekonomik Gelişmişlik Endeksi ve Sıralaması

İllerin Sosyo-Ekonomik Gelişmişlik Sıralaması – 2003

Plaka No	İller	Sosyo-Ekonomik Gelişmişlik Endeksi	Sosyo-Ekonomik Gelişmişlik Sıralaması (81 il içinde)
1	Adana	0,94901	8
2	Adıyaman	-0,77647	65
3	Afyon	-0,27246	44
4	Ağrı	-1,28116	80
5	Amasya	-0,18591	39
6	Ankara	3,31483	2
7	Antalya	0,91480	10
8	Artvin	-0,26018	43
9	Aydın	0,42025	22
10	Balıkesir	0,56540	15
11	Bilecik	0,50429	18
12	Bingöl	-1,12469	76
13	Bitlis	-1,15736	79
14	Bolu	0,60860	14
15	Burdur	0,14395	31
16	Bursa	1,67890	5
17	Çanakkale	0,36924	24
18	Çankırı	-0,51917	59
19	Çorum	-0,32761	46
20	Denizli	0,71624	12
21	Diyarbakır	-0,66993	63
22	Edirne	0,56234	16
23	Elazığ	-0,10131	36
24	Erzincan	-0,49288	58
25	Erzurum	-0,53286	60
26	Eskişehir	1,10368	6
27	Gaziantep	0,46175	20
28	Giresun	-0,36696	50
29	Gümüşhane	-0,92501	71
30	Hakkâri	-1,13956	77
31	Hatay	0,19613	29
32	Isparta	0,21187	28
33	Mersin	0,51934	17
34	İstanbul	4,80772	1
35	İzmir	2,52410	3
36	Kars	-0,81944	67
37	Kastamonu	-0,37558	51
38	Kayseri	0,47748	19
39	Kırklareli	0,86287	11
40	Kırşehir	-0,22870	42

41	Kocaeli	1,94329	4
42	Konya	0,25254	26
43	Kütahya	-0,20684	40
44	Malatya	-0,22627	41
45	Manisa	0,34165	25
46	K.Maraş	-0,34968	48
47	Mardin	-0,98944	72
48	Muğla	0,71238	13
49	Muş	-1,43956	81
50	Nevşehir	-0,07483	34
51	Niğde	-0,35582	49
52	Ordu	-0,64489	62
53	Rize	-0,17840	37
54	Sakarya	0,40404	23
55	Samsun	0,08791	32
56	Siirt	-1,00644	73
57	Sinop	-0,48518	57
58	Sivas	-0,40597	53
59	Tekirdağ	1,05893	7
60	Tokat	-0,59010	61
61	Trabzon	-0,18582	38
62	Tunceli	-0,40003	52
63	Şanlıurfa	-0,83158	68
64	Uşak	0,16867	30
65	Van	-1,09297	75
66	Yozgat	-0,71652	64
67	Zonguldak	0,44906	21
68	Aksaray	-0,45183	56
69	Bayburt	-0,80176	66
70	Karaman	-0,09852	35
71	Kırıkkale	0,05851	33
72	Batman	-0,90456	70
73	Şırnak	-1,13979	78
74	Bartın	-0,41550	55
75	Ardahan	-1,07318	74
76	Iğdır	-0,89089	69
77	Yalova	0,93541	9
78	Karabük	0,21332	27
79	Kilis	-0,41175	54
80	Osmaniye	-0,33321	47
81	Düzce	-0,27995	45

İllerin Sosyo-Ekonomik Gelişmişlik Sıralaması – 1996

Plaka No	İller	Sosyo-Ekonomik Gelişmişlik Endeksi	Sosyo-Ekonomik Gelişmişlik Sıralaması (76 il içinde)
1	Adana	0,825002	9
2	Adıyaman	-0,752853	61
3	Afyon	-0,228109	41
4	Ağrı	-1,134534	74
5	Amasya	-0,193947	39
6	Ankara	3,324472	2
7	Antalya	0,979019	7
8	Artvin	-0,361540	46
9	Aydın	0,572214	12
10	Balıkesir	0,566499	13
11	Bilecik	0,474944	17
12	Bingöl	-1,060746	73
13	Bitlis	-1,056951	71
14	Bolu	0,147192	28
15	Burdur	0,144998	29
16	Bursa	1,561681	5
17	Çanakkale	0,351583	20
18	Çankırı	-0,506919	53
19	Çorum	-0,338263	44
20	Denizli	0,501473	16
21	Diyarbakır	-0,614462	57
22	Edirne	0,408019	18
23	Elazığ	-0,024586	33
24	Erzincan	-0,369077	47
25	Erzurum	-0,550649	56
26	Eskişehir	1,010243	6
27	Gaziantep	0,199953	25
28	Giresun	-0,342129	45
29	Gümüşhane	-0,783385	63
30	Hakkâri	-1,053626	70
31	Hatay	0,189559	26
32	Isparta	0,337425	21
33	Mersin	4,879015	10
34	İstanbul	2,707983	1
35	İzmir	-0,450686	3
36	Kars	-0,754675	62
37	Kastamonu	-0,331970	43
38	Kayseri	0,530593	15
39	Kırklareli	0,554468	14
40	Kırşehir	-0,116527	36
41	Kocaeli	1,745641	4
42	Konya	0,220072	24

43	Kütahya	0,093397	31
44	Malatya	-0,179552	38
45	Manisa	0,308470	22
46	K.Maraş	-0,916083	50
47	Mardin	0,692054	66
48	Muğla	0,625896	11
49	Muş	-1,244671	76
50	Nevşehir	-0,006389	32
51	Niğde	-0,280378	42
52	Ordu	-0,535689	55
53	Rize	-0,122267	37
54	Sakarya	0,154779	27
55	Samsun	-0,042239	35
56	Siirt	-0,970848	68
57	Sinop	-0,512526	54
58	Sivas	-0,408015	48
59	Tekirdağ	-0,657586	8
60	Tokat	0,912105	52
61	Trabzon	-0,481332	34
62	Tunceli	-0,034803	60
63	Şanlıurfa	-0,694920	59
64	Uşak	0,249609	23
65	Van	-0,955459	67
66	Yozgat	-0,639394	58
67	Zonguldak	0,362269	19
68	Aksaray	-0,449240	49
69	Bayburt	-0,798578	64
70	Karaman	-0,225250	40
71	Kırıkkale	0,141061	30
72	Batman	-0,869404	65
73	Şırnak	-1,224524	75
74	Bartın	-0,472164	51
75	Ardahan	-1,057505	72
76	Iğdır	-0,980015	69