

KOCAELİ ÜNİVERSİTESİ * FEN BİLİMLERİ ENSTİTÜSÜ

**YAPAY SİNİR AĞLARI İLE KOCAELİ İLİNİN PUANT YÜK
TAHMİNİ**

YÜKSEK LİSANS TEZİ

DEMET ÖZDEMİR

**Anabilim Dalı : Elektrik Mühendisliği
Danışman: Yrd. Doç. Dr. Oya DEMİRER**

KOCAELİ, 2011

KOCAELİ ÜNİVERSİTESİ * FEN BİLİMLERİ ENSTİTÜSÜ

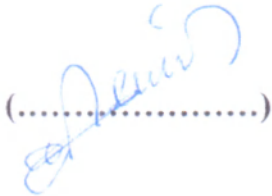
**YAPAY SİNİR AĞLARI İLE KOCAELİ İLİNİN PUANT YÜK
TAHMİNİ**

**YÜKSEK LİSANS TEZİ
DEMET ÖZDEMİR**

**Tezin Enstitüye Verildiği Tarih: 07 Ocak 2011
Tezin Savunulduğu Tarih: 24 Şubat 2011**

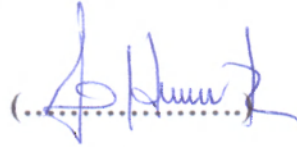
Tez Danışmanı

Yrd.Doç.Dr.Oya DEMİRER

(.....)


Üye

Prof.Dr.Semra ÖZTÜRK

(.....)


Üye

Yrd.Doç.Dr.Mehmet BAYRAK

(.....)


KOCAELİ, 2011

ÖNSÖZ ve TEŞEKKÜR

Elektrik enerjisi ekonominin ve sosyal yaşamın vazgeçilmez bir unsurudur. Üretildiği an tüketilmesi gereken elektrik enerjisinin tüketimi aylara, günlere ve hatta günün saatlerine göre değişkenlik gösterdiğinden tesisin kurulu gücü en yüksek talebi karşılayabilecek şekilde planlanmalıdır. Enerji santrallerine yapılması gereken eklemeler ve/veya yeni enerji santrallerinin kurulması, puant güçler göz önüne alınarak, öngörülen enerji talebini karşılamak üzere belirlenir. Bu belirlemede elektrik enerjisi tüketimini önemli ölçüde etkileyen nüfus artışı da göz önüne almak gerekir. Dolayısıyla yıllar itibariyle nüfus artışı ve artış trendi, nüfus planlaması dikkate alınmalıdır. Nüfusun yanı sıra mevsimsel farklılıkların getirdiği hava sıcaklıkları da yükü etkileyen çevresel faktörlerdendir. Bu yüzden yapay sınır ağı kullanılarak ileriye yönelik yapılan yük tahmininde, geçmiş yük bilgilerinin yanı sıra yükü nüfus ve hava sıcaklık bilgileri gibi çevresel faktörlere de bu çalışmada yer bulmuştur.

Kocaeli ilinin geçmiş yıllarına ait puant yük bilgilerini temin etmemde bana yardımcı olan Sakarya TEİAŞ yetkililerine çok teşekkür ederim.

Çalışmalarım boyunca beni destekleyen sayın danışman hocam Yrd. Doç.Dr.Oya DEMİRER'e, yine bu süre içerisinde değerli yardım ve katkılarıyla beni yönlendiren hocam Teknik Eğitim Fakültesi Doç.Dr.Melih İNAL'a, tez çalışmamın süresince gösterdiği anlayıştan dolayı MSB İzmit İnşaat Emlak Bölge Başkanlığına ve çalışma arkadaşlarıma ve hayatım boyunca beni destekleyen, bana güç veren çok değerli AİLEM ve eşim Cengizhan ÖZDEMİR'e şükran ve teşekkürlerimi sunarım.

İÇİNDEKİLER

ÖNSÖZ ve TEŞEKKÜR.....	i
İÇİNDEKİLER.....	ii
ŞEKİLLER DİZİNİ.....	i
TABLolar DİZİNİ.....	ii
SİMGELER ve KISALTMALAR.....	iii
ÖZET.....	iv
ABSTRACT.....	v
BÖLÜM 1. GİRİŞ.....	1
BÖLÜM 2. YÜK TAHMİNİ VE METODLARI.....	3
2.1. Yüklerin Özellikleri ve Sınıflandırılması.....	3
2.2. Yük Tahmini İçin Önemli Etkenler.....	4
2.3. Yük Tahmin Metodları.....	4
2.3.1. Uzman sistemlerle yapılan yük tahmini.....	6
2.3.2. Ekonometrik metodlar.....	7
2.3.3. Ortalama artış yüzdeleri ile yapılan tahmin.....	7
2.3.4. Son kullanıcı modeli.....	7
2.3.5. Gayri safi milli hasılaya dayalı yapılan tahmin.....	8
2.3.6. Yüzeysel yük tahmini.....	8
2.3.7. Yer renklendirme yöntemi.....	8
2.3.8. Arazi kullanım yöntemi.....	9
2.3.9. Regresyon analizi.....	9
2.3.10. Zaman serisi analizi.....	11
2.3.11. Bulanık mantık.....	13
2.3.12. Yapay sinir ağları.....	14
BÖLÜM 3. YAPAY SİNİR AĞLARI.....	15
3.1. Yapay Sinir Ağlarının Yapısı.....	16
3.1.1. Biyolojik nöron.....	16
3.1.2. Yapay sinir ağı hücresi.....	17
3.1.3. Girdiler.....	19
3.1.4. Ağırlıklar.....	19
3.1.5. Toplama fonksiyonu.....	19
3.1.6. Aktivasyon fonksiyonu.....	20
3.1.7. Hücre çıktısı.....	22
3.2. Yapay Sinir Ağı Bağlantı Yapısı.....	23
3.3. YSA Özellikleri.....	24
3.3.1. Doğrusal olmama.....	24
3.3.2. Analiz ve tasarım kolaylığı.....	24
3.3.3. Sınırsız sayıda değişken ve parametre.....	24
3.3.4. Hafıza.....	25
3.3.5. Genelleme.....	25
3.3.6. Öğrenme.....	25
3.3.7. Hata toleransı.....	25

3.3.8. Paralellik.....	26
3.4. Ağın Mimari Yapısına Göre YSA	26
3.4.1. İleri beslemeli çok katmanlı YSA	26
3.4.2. Geri beslemeli çok katmanlı YSA.....	27
3.5. YSA'da Eğitim	27
3.5.1. Eğitim algoritmaları	27
3.5.2. Eğitimin sınanması.....	28
3.6. Öğrenme Kuralları	29
3.7. YSA Modelleri	30
3.7.1. Tek katmanlı algılayıcı (perceptron).....	30
3.7.2. Çok katmanlı algılayıcı (multilayer perceptron).....	31
3.7.3. Hopfield ağı.....	31
3.7.4. Kohonen-self organizyon ağları.....	32
3.7.5. Uyarlanmış rezonans teorisi (ART)	33
3.7.6. Radyal tabanlı fonksiyon ağları (radial basis function networks-RBFN)	33
3.8. Geriye Yayılım Algoritması (Back Propagation Networks-BPN)	34
3.9. YSA Uygulama Alanları	35
3.10. YSA'ların Tarihsel Gelişimi.....	37
BÖLÜM 4. YAPAY SİNİR AĞLARI İLE KOCAELİ İLİNİN PUANT YÜK	
TAHMİNİNİN YAPILMASI	39
4.1. Yük Tahmin Verileri.....	39
4.1.1. Yük verileri.....	39
4.1.2. Nüfus verileri.....	41
4.1.3. Sıcaklık verileri.....	42
4.2. YSA Modeli ve Uygulaması.....	43
BÖLÜM 5. SONUÇ VE ÖNERİLER.....	51
KAYNAKLAR.....	52
EKLER.....	55
ÖZGEÇMİŞ	67

ŞEKİLLER DİZİNİ

Şekil 3.1: Sinir sisteminin blok diyagramı.....	16
Şekil 3.2: Biyolojik nöronun genel yapısı.....	17
Şekil 3.3: Yapay sinir ağı hücresi (yapay nöron).....	18
Şekil 3.4: En sık kullanılan aktivasyon fonksiyonları.....	21
Şekil 3.5: Basit bir YSA.....	23
Şekil 3.6: İleri beslemeli çok katmanlı YSA.....	27
Şekil 3.7: Geri beslemeli çok katmanlı YSA.....	27
Şekil 3.8: Perceptron.....	30
Şekil 3.9: Multilayer perceptron.....	31
Şekil 3.10: Hopfield ağı.....	32
Şekil 3.11: Kohonen ağı.....	32
Şekil 3.12: Basitleştirilmiş ART network.....	33
Şekil 3.13: Geri yayımlı ileri beslemeli ağ yapısı.....	34
Şekil 4.1: 2002-2010 yılları arası yıllık ortalama puant yük eğrisi.....	40
Şekil 4.2: 2002-2010 yılları arası aylık ortalama puant yük eğrisi.....	41
Şekil 4.3: 2002-2009 yılları arası aylık ortalama sıcaklık eğrisi.....	43
Şekil 4.4: YSA mimarisi.....	43
Şekil 4.5: Eğitim yük eğrisi.....	46
Şekil 4.6: Test yük eğrisi.....	46
Şekil 4.7: 2008 yılı tahmin ve gerçek puant yük eğrisi.....	48
Şekil 4.8: 2009 yılı tahmin ve gerçek değer puant yük eğrisi.....	48
Şekil 4.9: 2010 yılı tahmin ve gerçek değer puant yük eğrisi.....	49
Şekil 4.10: 2011-2020 yılları arası yıllık ortalama tahmini puant yük eğrisi.....	50
Şekil 4.11: 2011-2020 yılları arası aylık ortalama tahmini puant yük eğrisi.....	50

TABLolar DİZİNİ

Tablo 3.1: Toplama fonksiyonları	20
Tablo 4.1: Kocaeli ilinin aylık puant yük verileri	40
Tablo 4.2: Kocaeli nüfus sayım bilgileri.....	42
Tablo 4.3: 2002-2009 yılı aylık ortalama sıcaklık değerleri.....	42
Tablo 4.4: 2002-2007 yılları arası aylık puant yük verileri.....	45
Tablo 4.5: 2008-2010 yılları arası aylık puant yük verileri.....	45
Tablo 4.6: Gerçek ve tahmin puant yük değerleri ile hata oranları	47
Tablo 4.7: 2020 yılına kadar yapılan puant yük değerleri	49

SİMGELER ve KISALTMALAR

BPN : Geriye yayılım algoritması (Back Propagation Networks)

GSMH : Gayri safi milli hasıla

RBFN : Radyal tabanlı fonksiyon ağıları (Radial Basis Function Networks)

TEİAŞ : Türkiye Elektrik İletim Anonim Şirketi

YSA : Yapay sinir ağı

Π : Net girişi belirleyen çarpım fonksiyonu

f : Aktivasyon fonksiyonu

Σ : Net girişi belirleyen toplama fonksiyonu

YAPAY SINİR AĞLARI İLE KOCAELİ İLİNİN PUANT YÜK TAHMİNİ

Demet ÖZDEMİR

Anahtar Kelimeler: Yük Tahmini, Yük Tahmin Yöntemleri, Yapay Sinir Ağı, Geri-Yayılma Ağ

Özet: Elektrik enerjisinin yaşamın her alanında kullanılmasıyla birlikte ona olan bağımlılık her geçen gün artmaktadır. Bu da tüketicilere daha kaliteli, daha ekonomik ve daha güvenilir elektrik enerjisi sunmayı zorunlu kılmaktadır. Bu nedenle elektrik sistemlerinde geleceğe yönelik planlar yapılmalıdır. Bu planlamanın ilk ve en önemli adımı yük tahminidir. Bu çalışmada Yapay Sinir Ağları metodu kullanılarak Kocaeli ilinin puant yük tahmini yapılmıştır. Bu tahmin yapılırken geçmişteki veriler incelenerek gelecekte oluşabilecek durumlar tahmin edilmeye çalışılmıştır. Ayrıca en yakın sonuca ulaşabilmek için nüfus ve sıcaklık gibi çevresel etkenler de bu çalışmaya dahil edilmiştir.

ELECTRICAL PEAK LOAD FORECASTING USING ARTIFICIAL NEURAL NETWORKS IN KOCAELİ

Demet ÖZDEMİR

Keywords: Load forecast, load forecasting methods, artificial neural networks, backpropagation network.

Abstract: Using electrical energy in all areas of life with each passing day it is increasing its dependence on. This higher quality to consumers, offering more economical and more reliable electrical power is required. Therefore, the electrical system should be making plans for the future. The first and most important step in estimation at load. In this study, a method using Artificial Neural Networks is presented for the peak load forecast for Kocaeli. This estimation is done by examining historical data to estimate future conditions that could have been created. Moreover, to achieve the best result, the population and environmental factors such as temperature is included in this study.

BÖLÜM 1. GİRİŞ

Kaliteli elektrik enerjisi üretimi, bütün yükler için vazgeçilmez bir ihtiyaç olup, gerçekleştirilmesi oldukça zor ve pahalıdır. Bütün ülkelerde gerek ulusal, gerekse özel sektör yatırımlarının önemli bir bölümü elektrik enerjisi sektörüne ayrılmaktadır. Bundan dolayı elektrik enerji sistemi planlı, ekonomik, güvenilir ve geleceğe yönelik olmalıdır.

Elektrik enerjisi sektörünün sağlıklı gelişimi için talep, arz, iletim, dağıtım ve fiyatlandırma konularında planlar yapılması büyük öneme sahiptir. Yapılacak planlama çalışmalarının en önemli sorunlarından biri, gelecekteki talep bilgileridir. Elektrik enerjisinin depolanamaması, talep tahminindeki doğruluk derecesinin önemini artırmaktadır. Yapılacak tahminler ne kadar doğru olursa, yapılan planlama çalışmalarının da o kadar geçerli olacağı açıktır. Ayrıca tahminin doğruluğu, mevcut elektrik sisteminin kapasite artırımında belirleyici etken olmasından dolayı oldukça önemlidir.

Gerektiğinden düşük yük tahminlerine dayalı yapılan planlama, sistem güvenilirliğinin azalmasına, tüketiciye sunulan enerji arzının kısıtlanmasına ve enerji kalitesinde düşmelere neden olacaktır. Gerektiğinden fazla yük tahminlerine dayalı yapılan planlama ise tam kapasite ile hizmet veremeyen, düşük kapasiteyle çalışan ve dolayısıyla ekonomik olmayan işletme koşullarına neden olacaktır. Bu durumda sisteme yapılan büyük ölçekli yatırımlar enerji kurumu içinde önemli mali sorunlar ortaya çıkaracaktır.

Yapılan yük tahmininin doğruluğu yükü etkileyen etmenlere, kullanılan yöntemlere bağlıdır. Nüfus yoğunluğu, sanayileşme, iklim şartları, şehir planlamaları, elektriksel araç ve gereçlere yönelme, teknolojik gelişmeler, gayri safi milli hasıla gibi etmenler yükü oldukça etkilemektedir.

Bu tez kısaca aşağıdaki bölümlerden oluşmaktadır.

İkinci bölümde yük tahmininden bahsedilerek, yük çeşitleri ve yükü etkileyen faktörlerden kısaca söz edilmektedir. Bu bölümde ayrıca yük tahmininde kullanılan yöntemler de ele alınmaktadır.

Üçüncü bölümde yük tahmininde kullanılan YSA ve yapısından bahsedilmektedir.

Dördüncü bölümde Kocaeli ilinin geçmiş yıllara ait yük bilgileri ile yükü etkileyen nüfus ve sıcaklık bilgileri, test sonuçları, hata oranları ve 2020 yılına kadar olan yük tahmin bilgileri sunulmaktadır.

BÖLÜM 2. YÜK TAHMİNİ VE METODLARI

Etkili bir sistem planlaması için, puant yük ve enerji ihtiyacının tahmin edilmesi gereklidir. Yük tahmini, geçmişteki koşulların incelenerek gelecekteki durumun öngörülmesidir. Yük tahminlerini kısa, orta ve uzun dönemli olmak üzere üç değişik boyutta incelemek mümkündür.

1. Kısa dönemli tahmin: saatlik, günlük veya haftalık.
2. Orta dönemli tahmin: aylık, üç aylık.
3. Uzun dönemli tahmin: yıllık veya daha uzun dönemler.

Kısa dönemli tahminler birkaç dakikadan bir güne kadar güç santralleri arasında yük paylaşımı, en iyi grup belirleme ve ekonomik işletmenin yapılabilmesinde, orta dönemli tahminler bir günden bir yıla kadar dönem için yakıt kaynaklarının dağılımını ve bakım işlemlerinin zamanlamasının belirlenmesinde, uzun dönemli tahminler ise iletim ağlarının ve yeni üretim kapasitelerinin ekonomik planlamasında ve sistemin modifiye edilmesinde önemli olmaktadır.

2.1. Yüklerin Özellikleri ve Sınıflandırılması

Yük sınıflandırmalarının yapılması için tüketici gruplarının özelliklerinin bilinmesi gerekir. Yükler;

1. Mesken,
 2. Ticarethaneler,
 3. Sanayi tipi tüketiciler,
 4. Diğer tüketiciler,
- şeklinde sınıflandırılır.

Mesken tipi tüketiciler, müstakil bina ve apartmanlardır. Ticari tipi tüketiciler çeşitli meslek gruplarının kullandıkları ofis, dükkan gibi yerlerdir. Sanayi tipi tüketiciler fabrikalar büyük atölyeler gibi endüstriyel amaçlı kurulmuş işletmelerdir. Diğer tüketiciler grubunda ise dernekler, hayır kurumları, müzeler, resmi kurslar, resmi okullar, resmi sağlık kuruluşları, resmi üniversite, resmi yurtlar, resmi yüksek okullar, vakıflar genel aydınlatma vb. kuruluşlar vardır. Yük profillerine göre ticarethane ve sanayi grupları ile meskeni ayıran ana etken puant ve gece tüketimleridir. Meskenlerdeki tüketim, sanayi ve ticarethaneye göre geceye kaymaktadır. Ticarethane ve diğer tüketicilerin tüketimleri ise mesai saatleri içerisindeydir.

2.2. Yük Tahmini İçin Önemli Etkenler

Doğru bir yük tahmini model oluşturmadan önce, yük tahmini yapılacak bölgenin yük karakteristiği incelenmeli ve hangi faktörlerin yükü ne şekilde etkilediği araştırılmalıdır. Yükü etkileyen faktörler; nüfus artışı, gayri safi milli hasıla, ekonomik veriler,coğrafik faktörler (sıcaklık, nem, yağış,...), geçmişteki veriler, endüstriyel ve şehir planları, arazi kullanımı, sosyo-kültürel faktörler, insanların hayat tarzlarındaki ve kullanım alışkanlıklarındaki değişiklikler, teknolojik gelişmeler olarak sıralanabilir. Yük tahmininde kullanılacak olan değişkenlerin belirlenmesi, öncelikle bu verilerin elde edilip edilememesine, daha sonra tahminin özelliğine, yonteme ve tüketici grubuna bağlıdır.

Kısa dönem yük tahmini için zaman etkeni, elektriksel veri ve müşteri sınıfları gibi etkenler vardır. Orta ve uzun dönem tahminler geçmiş verileri, elektriksel verileri, farklı müşteri sayılarını, uygulama alanını, ekonomik verileri ve onların tahminlerini hesaba katar [1].

2.3. Yük Tahmin Metodları

Yük tahmin çalışmalarının çoğunda birbirinden farklı modeller oluşturulmuştur. Her elektrik şebekesinin ve her tüketici bölgenin iklimsel ve coğrafi özellikleri birbirine

göre farklılık gösterdiği için fazla sayıda yük tahmini modelinin öne sürülmesi normal bir durumdur. Yük değişimlerinin yapısına bağlı olarak bir metot diğerine göre üstünlük gösterebilir. Yük tahmin metotları genel olarak ekstrapolasyon ve korelasyon tahmin metotlarıdır. Ekstrapolasyon, geçmiş verilerin ve bu verileri etkileyen güçlerin geçmişte olduğu gibi gelecekte de aynı oranda artacağı varsayılarak yapılan tahmindir. Birçok ekstrapolasyon yöntemi vardır. Bunların bazıları matematiksel büyüme eğrilerinin yorumlanmasından oluşur. Diğerleri ise geçmiş yıllardaki büyüme ortalamalarının ileri yıllar için kullanılmasıdır. Korelasyon, yüklerin diğer faktörlere bağlanmasıyla gerçekleşen tahmindir. Örneğin hava şartları veya ekonomik durum gibi. Korelasyonun en önemli avantajı büyümeyi etkileyen faktörleri önemlerine göre değerlendirmesidir. Korelasyon metodu aynı zamanda tahminlerin gerçek değerlerden sapması durumunda sebebin belirlenmesinde yardımcıdır[1].

Genel olarak istatistiksel yük tahmini teknikleri, ör.regresyon, zaman serileri, model tanıma, Kalman filtreleri vs, uzun zamandır pratikte kullanılmaktadır. Bu geleneksel metotlar ağırlıklı çoklu-model tahmin teknikleri kullanılarak birleştirilebilir. Fakat bu metotlar, yük ve onu etkileyen faktörler serisi arasında var olan kompleks doğrusal olmayan ilişkileri, uygun bir şekilde temsil edemezler, ki bunlar sistem değişikliklerine bağımlıdır (ör. Mevsim veya günün vakti).

Son yıllarda birçok tahmin metodu gelişmektedir. Uzman sistemler, yapay sinir ağları ve bulanık mantık gibi modern yük tahmin teknikleri son zamanlarda geliştirilmişlerdir ve cesaret verici sonuçlar vermektedirler. Bunlar arasında, yapay sinir ağları diğer metotlara göre oldukça dikkat çekicidir çünkü bunların, yük ile yükü doğrudan etkileyen faktörler arasındaki lineer olmayan ilişkileri değerlendirme kabiliyeti bulunmaktadır.

Genelde kullanılan tahmin metotlarını şöyle sıralayabiliriz;

1. Uzman Sistemlerle Yapılan Yük Tahmini
2. Ekonometrik modeller,
3. Ortalama artış yüzdeleri ile tahmin,

4. Son kullanıcı modeli,
5. Gayri safi milli hasılaya (GSMH) dayalı yapılan tahmin,
6. Yüzeysel yük tahmini,
7. Yer renklendirme yöntemi,
8. Arazi kullanım yöntemi,
9. Regresyon analizi,
10. Zaman serisi analizi,
11. Bulanık Mantık,
12. Yapay sinir ağları (YSA).

2.3.1. Uzman sistemlerle yapılan yük tahmini

Uzman Sistemler, belirli bir alanda sadece o alan ile ilgili bilgilerle donatılmış ve problemlere o alanda uzman bir kişinin getirdiği şekilde çözümler getirebilen bilgisayar programları olarak tarif edilebilir. Sistemi geliştiren kişi ile sistemin geliştirileceği alandaki uzman kişi ya da kişilerin karşılıklı fikir alışverişleri sonucu ortaya çıkarılan bilgi, sistemin bilgi tabanına uygun formatta girilir. Şayet ileride bilgide bir değişiklik yapmak gerekirse bu değişiklik bilgi tabanı üzerinde rahatça yapılabilir. Tamamlanan sistem, bilgi tabanındaki bilgiyi kendisine uç kullanıcılar tarafından belirtilen problemleri çözmede kullanır.

Elektrik enerjisi üretim genişletme planlaması, yüksek derecede kısıtlanmalı, doğrusal olmayan, ayrık yapılı, dinamik ve bilgisayar kullanımını gerekli kılan bir optimizasyon problemidir. Bu nedenle yapılacak olan kestirimlerin düzenli ve hızlı bir şekilde olması gerekmektedir. Uzman sistemlerle yapılan yük kestirimleri hem zaman açısından hem de doğruluk açısından büyük fayda taşımaktadır. Yük kestiriminde kullanılan uzman sistemlerden yapay sinir ağları, genetik algoritmalar ve bulanık mantık yöntemleri kullanılarak literatürde birçok çalışma yapılmıştır. Bu yöntemler ile yük kestirimine etki eden parametreler bir fonksiyon olarak tanımlanarak kestirim işleminde kullanılabilir. Ayrıca birçok parametre yük kestiriminde kullanıldığından yapılan yük kestirimi gerçeğe daha yakın sonuç verebilmektedir. Günümüzde yük kestiriminde daha çok uzman sistemler kullanılır hale gelmiştir[2].

2.3.2. Ekonometrik metodlar

Ekonometrik yaklaşımlar elektrik tahmini için ekonomik verileri ve istatistiksel teknikleri birleştirir, değişkenler arasında bağlantı kurarak bunlardan matematiksel bir fonksiyon elde edilmesini sağlar. Bu yaklaşım, enerji tüketimi (bağımlı değişken) ve tüketimi etkileyen faktörler arasındaki ilişkiyi hesaplar. Elektrik enerjisi talebi sadece zamanın bir fonksiyonu olmayıp, ekonomik ve sosyal değişimlerden, teknolojik gelişmelerden ve çevre şartlarından da etkilenmektedir[1].

2.3.3. Ortalama artış yüzdeleri ile yapılan tahmin

Bu yöntemde geçmiş yıllardaki tüketilen enerji değerindeki yıllık artış oranlarının ortalamaları bulunarak ileriye yönelik tahminler yapılır.

Gelecekteki enerji tüketiminin belirlenmesi için aşağıdaki formül kullanılır;

$$E(t) = E_0 (1+\varepsilon)^{\exp(t)} \quad (2.1)$$

$E(t)$ t yıl sonraki enerji tüketimini, E_0 en son yıldaki enerji tüketimini, ε ortalama yıllık artış yüzdesini ifade etmektedir.

Ancak bu yöntemle sabit bir artış değeri öngörüldüğünden, yapılacak yük kestirim sonuçları ile gerçek değerler arasında büyük farklılıklara rastlanmaktadır.

2.3.4. Son kullanıcı modeli

Bu metot ile mal ve hizmetlerin üretilmesi için gerekli enerji tüketimi kapsamlı enerji tüketim bilgisini, müşteri kullanımını, onların yaşlarını, evlerin büyüklüğünü gibi detaylı hedeflere bölünmektedir. Müşteriler hakkındaki istatistiksel bilgiler ile birlikte değişen dinamikler tahmin için temeli oluşturur. Bu hedefler için bulunan tüketim miktarları toplanarak toplam tahmin elde edilmektedir. Son kullanıcı modeli konutlar, ticari ve endüstriyel sektörlerdeki elektrik tahmini için de kullanılır.

İdeal olarak bu yaklaşım oldukça kesindir. Bununla beraber, son-kullanıcı verilerinin kalitesi ve miktarının hassas olması oldukça önemlidir. Bu yüzden uygulama açısından oldukça zor bir yöntemdir[1].

2.3.5. Gayri safi milli hasılaya dayalı yapılan tahmin

Türkiye’de uzun dönemli tahminler yapılırken kullanılan yöntemlerden biri de GSMH büyüme hızıyla, elektrik enerjisindeki büyüme hızı arasındaki ilişkiyi bir katsayı ile belirlemek ve kalkınma planı ile hedeflenen büyüme hızlarına göre elektrik enerjisi tahminini bulunan bu katsayıya göre hesaplamaktır. Bu yöntem ekonomik ve siyasi açıdan istikrara sahip ülkelerde geçerli ve başarılı olur.

2.3.6. Yüzeysel yük tahmini

Bu yöntemde önemli olan iki unsur yük tahmini ve büyüme karakteristiğidir. Bu yöntemde tüketici karakteristikleri sınıflandırılarak yük kestirimi gerçekleştirilmektedir. Buna göre yerleşim alanı, yerleşim yeri (kırsal yerleşim yeri ve tek katlı müstakil evler), ticarethaneler, ofisler, depolar, resmi daireler büyük ölçekli fabrikalar, küçük ölçekli fabrikalar vb. şekilde sınıflara ayrılmıştır.

2.3.7. Yer renklendirme yöntemi

Bu yöntemde ilk olarak her bir tüketici sınıfının yerleşim yerleri belirlenir ve bunun için ele alınan bölge küçük eşit parçalara bölünerek tüketici gruplarına karşı düşen alan hesaplanır. Ayrıca incelenen bölgenin ölçekli haritası (otoyolları, demir yolları, yerleşim yerlerinin ve bölgenin elektrik iletim-dağıtım sistemini içeren harita) kullanılır[3].

Bu yöntem küçük ve orta ölçekli yerleşim yerlerine uygulandığında çok iyi sonuç verebilir[3].

2.3.8. Arazi kullanım yöntemi

Bu yöntem veri tabanı oluşturmada mükemmel bir yük tahmini sağlamaktadır. Güncelleştirilmiş coğrafik verileri toplamak için harcanacak çok uzun sürelerden (binlerce saat) tasarruf sağlar. Sonuçta diğer yöntemlere göre daha kullanışlıdır ve daha yaklaşık sonuçlar verir. Yer renklendirme ve arazi kullanım yöntemleriyle, hem tüketici grubundaki yıllık artış değerlerinde hem de yerleşim yerinin artış değerlerinden yararlanılarak tepe yük değerleri başarıyla tahmin edilir[3].

2.3.9. Regresyon analizi

Regresyon analizi iki veya daha fazla değişken arasındaki ilişkileri ölçmek için kullanılır. Regresyon analizinde, etkileyen değişkenlere bağımsız(serbest) değişken, etkilenen değişkene de bağımlı değişken adı verilir. Bağımlı ve bağımsız değişkenler arasındaki dağılım diyagramının çizilmesiyle iki değişken arasında bir bağlantının kurulup kurulamayacağına, eğer kuruluyorsa nasıl bir fonksiyonun ele alınacağına karar verilir. Hem tanımlayıcı hem de çıkarımsal istatistik sağlar.

n tane x serbest değişkeninin ($x_1, x_2, x_3, \dots, x_n$) değerlerine karşılık gelen n tane y değişkeninin ($y_1, y_2, y_3, \dots, y_n$) olduğunu varsayalım. Bu durum dağılım diyagramında n adet nokta belirtir. Noktaların genel durumuna göre değişkenler arasındaki ilişki doğruyla gösterilebiliyorsa; doğrusal regresyon göz önüne alınır. Noktalar arasından geçen doğrunun denklemi $y = f(x) = a+bx$ dir. Doğrunun her x_i değerine karşılık gelen nokta $f(x_i)$ 'dir. Böylece her gerçek y_i değerine doğru üzerinde bir $f(x_i)$ teorik değeri karşılık gelecektir. Noktalar arasından geçirilen doğrunun iki değişken arasındaki ilişkiyi en iyi şekilde temsil etmesi için a ve b katsayılarının, n nokta için gerçek ve teorik koordinatlarının aralarındaki farkların karelerinin toplamı minimum olacak şekilde seçilmesi gerekmektedir[1].

$$Q = \sum [y(i) - f(x_i)]^2 \quad (2.2)$$

Minumum olmalı ve bunu gerçekleştirmek için aşağıdaki fonksiyonun a ve b'ye göre kısmi türevleri alınır.

$$Q = \sum [y(i) - (a + bx)]^2 \quad (2.3)$$

$$y(i) = y_i$$

$$dQ/da = -2 \sum [y_i - a - bx_i] = 0 \quad (2.4)$$

$$dQ/db = -2 \sum x_i [y_i - a - bx_i] = 0 \quad (2.5)$$

$$na + \sum bx_i = \sum y_i \quad (2.6)$$

$$a \sum x_i + \sum bx_i^2 = \sum x_i y_i \quad (2.7)$$

$$y'' = \sum y_i / n \text{ ve } x'' = \sum x_i / n \quad (2.8)$$

(n=toplum gözlem sayısı)

$$a = y'' - b x'' \quad (2.9)$$

$$b = (\sum x_i y_i - y'' \sum x_i) / (\sum x_i^2 - x'' \sum x_i) \quad (2.10)$$

olarak bulunur. Eğer regresyon denklemi üstel bir ifade ise ve iki değişken arasındaki ilişki üstel bir fonksiyon ise;

$$y = a * b \exp(x) \quad (2.11)$$

olarak yazılabilir. Bu denklemin logaritması alınır,

$$\log y = \log a + x \log b \quad (2.12)$$

$$Y = \log y \quad A = \log a \quad B = \log b$$

$$Y = A + Bx$$

doğru denklemi elde edilir. Noktaların genel durumuna göre değişkenler arası ilişki parabolle gösteriliyorsa;

$$F(x) = a + bx + cx^2 \quad (2.13)$$

$$Q = \sum [y(i) - f(x_i)]^2 = \sum (y_i - f(a + bx_i + cx_i^2))^2 \quad (2.14)$$

minimum olmalıdır. Fonksiyonun a, b, c'ye göre kısmi türevleri alınır;

$$dQ/da = -2 \sum [y_i - a - bx_i - cx_i^2] = 0 \quad (2.15)$$

$$dQ/db = -2 \sum x_i [y_i - a - bx_i - cx_i^2] = 0 \quad (2.16)$$

$$dQ/dc = -2 \sum x_i^2 [y_i - a - bx_i - cx_i^2] = 0 \quad (2.17)$$

$$na + \sum bx_i + \sum cx_i = \sum y_i, \quad (2.18)$$

$$a\sum x_i + \sum bx_i^2 + c\sum x_i^3 = \sum x_i y_i \quad (2.19)$$

$$a\sum x_i^2 + b\sum x_i^3 + c\sum x_i^4 = \sum x_i^2 y_i \quad (2.20)$$

Bu denklemlerden a,b ve c katsayıları bulunur.

Eğer bağımsız değişken sayısı birden fazla ise elektrik enerjisi talep artışına birden fazla faktör etki ediyorsa çok değişkenli regresyon analizi yapılır. Bu durumda fonksiyon aşağıdaki gibi olur[1].

$$y_i = f(x_i) = a_0 + a_1 x_{i1} + a_2 x_{i2} + a_3 x_{i3} + \dots + a_n x_{in} \quad (2.21)$$

Model matris;

$$[Y] = [X] * [A] \quad (2.22)$$

2.3.10. Zaman serisi analizi

Zamanın periyodik noktalarında, bir cevap değişkeni gözlemlenmesi yoluyla verilerin toplanması zaman serisi olarak adlandırılır. Bir değişken üzerine tekrarlanan gözlemler zaman serisini oluşturursa, değişken, zaman serisi değişkeni olarak tanımlanır. Zaman serilerinin analizi seriyi oluşturan bileşenlerden ayrışımını gerektirir. Bir seriyi bileşenlerine ayırmak için kapsadığı dört bileşen arasında belli bir ilişki bulunduğu varsayılmalıdır. Genellikle izlenen yol bir zaman serisinin birkaç bileşenini toplamı ya da çarpımından meydana geldiği varsayımdır[4].

Bir Klasik Modelde, Zaman Serisi Dört Elemana Sahiptir.

1. Uzun – dönemli genel trend(eğilim) (T)
2. Dönemsel dalgalanmaları (C)
3. Mevsimsel dalgalanmalar (S)
4. Varyasyon ve düzensiz rast gele hareketler (I)

$$Y = (T)(C)(S)(I) \quad (2.23)$$

İstatistiksel yönden zaman serilerinin amacı; bahsedilen dört elemandan her birinin, olayın aldığı değerler üzerinde ne ölçüde etkili olduğunu araştırmaktır.

Olayın bağlı olduğu temel ve taşıdığı özellikler, uzun dönemde çok fazla değişmeyen belirli bir yön taşıdığından, olayın bağlı olduğu temel ve taşıdığı özellikler, uzun – dönemli genel trend(eğilim) olarak adlandırılır. Eğilim, zaman serilerindeki değişkenlerin değerlerinin uzun vadede oluşturdukları kararlı durumdur. Eğilim analizi bir uzun dönem analizi olduğundan verilerin aylık veya mevsimlik olarak verilmiş olması tahlilin sonucunu etkilemeyecektir. Eğilimin yön ve şiddeti her zaman sabit kalmaz. Eğilim doğrusal ya da eğrisel olabilir

Dönemsel değişimler, bir eğilim doğrusu etrafındaki uzun dönem dalgalanmalara denir. Bu hareketler mevsimsel değişimlere benzer şekilde dönemsel olarak tekrarlınsalar da, dönemlerin uzunluğu ve sürelerinin belirsizliği ile dikkati çekmektedir[2].

Zaman serilerindeki değişkenlerin değerlerinin mevsimlere göre değişkenlik gösterdiği durumdur. Mevsimsel dalgalanmaların dalga uzunluğu 12 ay olmaktadır. Mevsimlik değişimler, bu on iki aylık dönem içerisinde oluşan az ya da çok düzenli iniş ve çıkışlar olarak tanımlanır. Bu dalgalanmalar doğal sosyal veya sanal nedenlerle ortaya çıkmaktadır. Yıllık olarak düzenlenen zaman serisinde mevsimlik dalgalanmalar görünmemektedir.

Daha önceden kestirilemeyen, hata terimi yardımıyla açıklanabilen değişmelerin durum olan varyasyon ve düzensiz rasgele hareketler, doğal ve sosyo – ekonomik nedenlerden dolayı ortaya çıkabilir. Ancak, sözü edilen hareketlerin ne zaman, nasıl bir dalga şiddeti ile meydana geleceği belirsizlik taşıdığı için tahmini mümkün olmaz. Bu nedenle bu tip hareketleri bir istatistiksel ölçüyle ifade edebilmek zordur.

Zaman serisi yaklaşımından elektrik tüketimi, ekstrapolasyon teknikleri kullanılarak tahmin edilmektedir. Ekstrapolasyon işlemi gerçekleştirilirken, büyüme trendini yansıtmak için tarihsel veriler düzenlenir ve bunlar için en uygun fonksiyon elde edilmeye çalışılır.

2.3.11. Bulanık mantık

Bulanık mantık (Fuzzy Logic) kavramı ilk kez 1965 yılında Lofti A. Zadeh'in bu konu üzerine ilk makalesini yayınlaması ile duyuldu. O tarihten sonra önemi gittikçe artarak günümüze kadar gelen bulanık mantık belirsizliklerin anlatımı ve belirsizliklerle çalışabilmesi için kurulmuş bir matematiksel düzen olarak tanımlanabilir.

Mühendislik ve diğer bilim dallarında olaylar ve sistemler kesin matematik modeller kullanılmak suretiyle ele alınıp incelenirler. Oluşturulan bu modelin kullanılması ile sistemin veya olayın gelecekte alacağı durum veya göstereceği davranış biçimi tahmin edilir. Fakat günlük hayatta karşılaşılan problemlerin çeşitli nedenlerden dolayı tam olarak modellenemeyebilir veya kesinlik içermeyebilir. Bu tip problemlerin incelenmesinde ve çözülmesinde bulanık mantık yaklaşımı uygulanabilir.

Klasik mantık işlemlerinde doğruluk değeri olarak sadece $\{0,1\}$ veya “doğru” ve “yanlış” değerlerini kullanırken, bulanık mantıkta $[0,1]$ aralığındaki sonsuz değeri doğruluk değeri olarak kullanılır. Gerçek hayata baktığımızda hemen hemen hiçbir şey kesinlikle doğru veya kesinlikle yanlış değildir; önermeler genelde kısmen doğru veya kısmen yanlıştır.

Bulanık kuramın temel kavramı bulanık kümelerdir. Örneğin “orta yaş” kavramını inceleyecek olursak, bu kavramın sınırlarının kişiden kişiye değişiklik gösterdiğini görürüz. Kesin sınırlar söz konusu olmadığı için kavramı matematiksel olarak kolayca ifade edemeyiz. Ama genel olarak 35–50 yaşları orta yaşlılık sınırları olarak düşünülebilir. Bu durum grafik olarak düşünülürse aitlik eğrisi adı verilen ve kavram içinde hangi değerlerin hangi ağırlıkta olduğunu gösteren bir eğri çıkar.

Bulanık mantık sistemine örnek olarak; Bir ifade tamamen yanlış ise klasik mantıkta olduğu gibi 0 değerinde; tamamen doğru ise 1 değerindedir. Bunların dışında tüm ifadeler 0'dan büyük 1'den küçük değerler alırlar. Yani değeri 0,25 olan bir ifadenin anlamı %25 doğru, %75 yanlış demektir. Bulanık mantığın uygulama alanları

geniştir. Sağladığı en büyük fayda ise insana özgü tecrübe ile öğrenme olayını kolayca modelleyebilmesi ve belirsiz kavramların bile matematiksel olarak ifade edilebilmesine olanak tanınmasıdır. Bu nedenle doğrusal olmayan sistemlere yaklaşım yapabilmek için özellikle uygundur.

Bulanık sistem yaklaşımları yük tahmininde sıcaklık, gün tipi (tatil, iş günü,...) gibi yük değişim faktörleri içeren günlere ait yükün tahmin edilmesinde iyi sonuçlar vermektedir. Bununla birlikte YSA ve bulanık mantık tekniklerinin beraber kullanımını ile daha etkin sistemler tasarlamak mümkündür.

2.3.12. Yapay sinir ağları

Teknolojik gelişmelerin önemli boyutlara ulaştığı günümüzde, insanoğlu kendisini tanımaya yönelik bir takım çalışmalar yapmıştır. Bu çalışmalardan bir tanesi de Yapay Sinir Ağlarıdır. YSA üçüncü bölümde detaylı olarak anlatılacaktır.

BÖLÜM 3. YAPAY SİNİR AĞLARI

Yapay Sinir Ağları (YSA), insan beyninin özelliklerinden olan öğrenme yolu ile yeni bilgiler türetebilme, yeni bilgiler oluşturabilme ve keşfedebilme gibi yetenekleri, herhangi bir yardım almadan otomatik olarak gerçekleştirebilmek amacı ile geliştirilen bilgisayar sistemleridir.

YSA'lar insan beyninden esinlenerek, öğrenme sürecinin matematiksel olarak modellenmesi uğraşı sonucu ortaya çıkmıştır. Bu nedendir ki, bu konu üzerindeki çalışmalar ilk olarak beyni oluşturan biyolojik üniteler olan nöronların modellenmesi ve bilgisayar sistemlerinde uygulanması ile başlamış, daha sonraları bilgisayar sistemlerinin gelişimine de paralel olarak birçok alanda kullanılır hale gelmiştir.

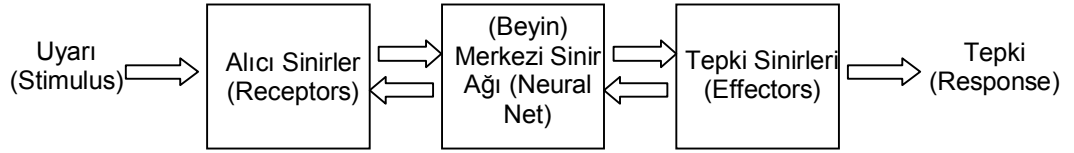
İnsan beyninin çalışma prensibini taklit ederek çalışan bu sistemler, her ne kadar bilgisayar teknolojisi hızlı bir gelişim göstermiş, işlem hızları nano saniyeler mertebesine inmiş olsa da, bırakalım insan beynini, ilkel bir canlı beyninin fonksiyonları dahi baz alındığında, böyle bir organizmanın yanında çok ilkel kalmaktadır. Nano saniyeler bazındaki işlem hızları ile YSA'lar, mili saniyeler mertebesinde işlem yapan insan beyninin işlevselliğinin henüz çok uzağındadır. Çünkü bilgisayarlar insanın beyinsel yeteneğinin en zayıf olduğu çarpma, bölme gibi matematiksel ve algoritmik hesaplama işlemlerinde hız ve doğruluk açısından yüzlerce kat başarılı olmalarına rağmen insan beyninin öğrenme, tanıma gibi işlevlerini hala yeteri kadar gerçekleştirememektedir.

Burada kısa bir hatırlatma yapmak gerekirse; insan beyninde yaklaşık 10^{11} sinir hücresinin varlığından bahsedilmekle birlikte, bu sayının bilgisayar ortamında modellenmesi şu an için mümkün değildir. Fakat karar hızı açısından insan beyni ile henüz yarışmasalar bile, YSA'lar yapısallıkları ve hassas eşleştirmelerdeki başarıları açısından gün geçtikçe daha fazla uygulama alanı bulmaktadır.

3.1. Yapay Sinir Ağlarının Yapısı

Daha önceden de bahsedildiği üzere yapay sinir ağları, insan beyninden esinlenilmiş yapılardır. Bu yüzden, YSA'ların yapısını anlamak için öncelikle insan beyninin yapısını ve çalışma şeklini anlamak gerekmektedir.

İnsan beyni, sinir sisteminin merkezini oluşturan temel elemandır. En basit şekilde, sürekli olarak iletilen bilgiyi alır, idrak eder (işler) ve uygun kararları vererek gerekli yerlere iletir. Çok basit görünmekle birlikte aslında oldukça karmaşık olan bu yapının basit bir gösterimi şu şekildedir.



Şekil 3.1: Sinir sisteminin blok diyagramı[8]

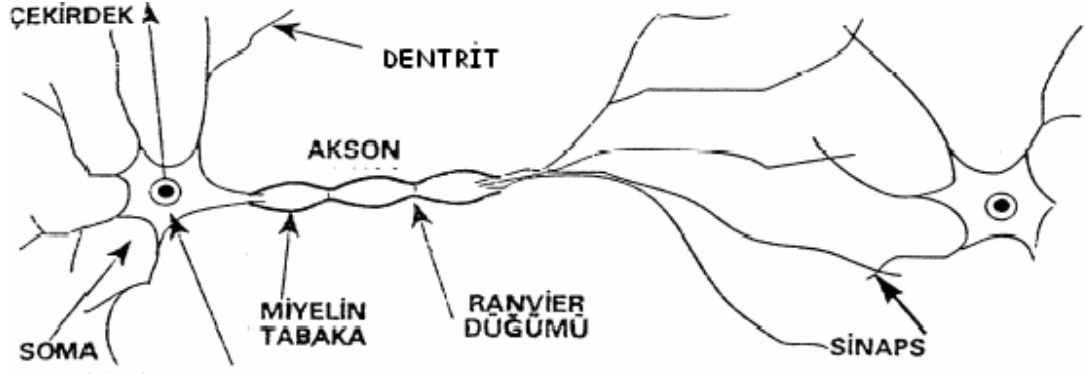
Şekilden de anlaşılacağı üzere biyolojik sinir sistemi, merkezinde sürekli olarak bilgiyi alan, yorumlayan ve uygun bir karar üreten beyin (merkezi sinir ağı) bulunduğu üç katmanlı bir sistem olarak açıklanır. Alıcı sinirler (receptor) organizma içerisinden ya da dış ortamlardan algıladıkları uyarıları, beyine bilgi ileten elektriksel işaretlere dönüştürür. Tepki sinirleri (effector) ise, beyinin ürettiği elektriksel darbeleri organizma çıktısı olarak uygun tepkilere dönüştürür. Şekilde Merkezi Sinir Ağı olarak gösterilen ortadaki bölüm, yani beyin, sinir sisteminin merkezi konumundadır. Beynin temel yapı taşları ise sinir hücreleri, diğer bir ifadeyle nöronlardır.

3.1.1. Biyolojik nöron

Yapay Sinir Ağları, beyinden esinlenildiklerinden dolayı benzer yapıya sahiptirler. Doğal olarak, YSA'ların temel elemanları da yapay nöronlardır. Yapay nöronları tanımak için öncelikle simüle ettikleri biyolojik nöronları tanımak gerekir.

Beyinde yaklaşık olarak 10^{11} sinir hücresi vardır. Yine hücre başına bağlantı sayısı ise 10^4 mertebesindedir. Beyin için çalışma frekansı 100 Hertz'dir. Fiziksel boyutları

ise 1.3 kg ve 0.15 m² kesitlidir. Vücudun değişik yerleri ile bilgi alış-verişi yapan nöron hücresidir.

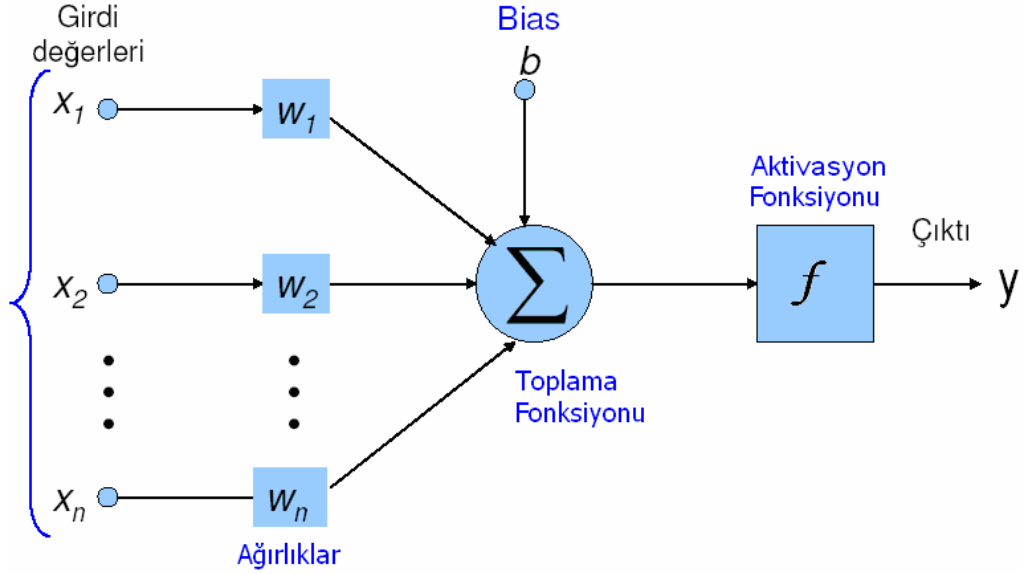


Şekil 3.2: Biyolojik nöronun genel yapısı[9]

Şekilde de görüldüğü gibi bir nöron üç ana bileşenden oluşur. Bunlar; soma olarak adlandırılan çekirdek, akson ve dentritlerdir. Dentritler, çekirdeğin saça benzeyen uzantılarıdır ve girdi kanalları olarak işlev görürler. Bu girdi kanalları diğer nöronların sinapsları aracılığıyla girdilerini alırlar. Daha sonra çekirdek, gelen bu sinyalleri zaman içinde işler. Çekirdek, bu işlenmiş değeri bir çıktıya dönüştürdüktan sonra bu çıktıyı akson ve sinapslar aracılığıyla diğer nöronlara gönderir. Sinapslar, nöronlar arasındaki elektrokimyasal bağlantıyı sağlamaktadır. Bir biyolojik nöron, temel olarak, diğer kaynaklardan girdiler alır, belirli bir şekilde bunları birleştirir, sonuç üzerinde bir işlem (genelde doğrusal olmayan) uygular ve kesin sonucu üretir.

3.1.2. Yapay sinir ağı hücresi

Yapay sinir ağı hücresi (yapay nöron) biyolojik nörona göre daha basit yapıda olup matematiksel modeli aşağıdaki şekilde gibidir.



Şekil 3.3: Yapay sinir ağı hücresi (yapay nöron)[4]

YSA hücresinde; temel olarak dış ortamdan ya da diğer nöronlardan alınan veriler yani girdiler (x_n), ağırlıklar (w_n), toplama fonksiyonu (Σ), aktivasyon fonksiyonu (f), aktivasyon fonksiyonunun eşik değeri (b) ve çıkışlar (y) bulunmaktadır.

YSA hücresinin net çıkışı aşağıdaki gibi hesaplanır.

$$y = f \sum_{i=1}^n (x_i w_i + b) \quad (3.1)$$

Kısaca basit bir sinir hücresinin işleyişi şu şekildedir.

- 1-) Giriş sinyallerinin ağırlıklandırılmış toplamı alınır.
- 2-) Bir eşik değeri seçilir.
- 3-) Girdilerin toplamı eşik değeri ile karşılaştırılır. Bu değer eşik değerinden küçükse çıktı -1, büyükse +1 olur. (ikinci durumda sinir hücresi aktifleşmiştir.)

Aktivasyon fonksiyonu işlem süresince aynı zamanda nöron çıkışı da olan net çıkışı hesaplayan bir fonksiyondur. YSA'da aktivasyon fonksiyonu olarak birçok fonksiyon kullanılmaktadır. Ağda hangi aktivasyon fonksiyonunun kullanılacağı probleme göre değişmektedir. Ancak aktivasyon fonksiyonu genellikle doğrusal olmayan bir fonksiyondur.

3.1.3. Girdiler

Bir YSA hücresine dışardan verilen ve x_1, x_2, \dots, x_n olarak gösterilen verilerdir. Bu veriler ağın öğrenmesi istenen örneklerden belirlenir. Yapay sinir ağı hücresine dışarıdan bilgi verildiği gibi, bir işlem biriminden diğerine de bilgi gelebilir.

3.1.4. Ağırlıklar

Yapay sinir hücrelerini birbirine bağlayan bağlantıların her birinin sayısal bir ağırlığı vardır ve ağırlıklar w_1, w_2, \dots, w_n olarak ifade edilir ve ilgili girişlerin işlem biriminin net çıkışı üzerindeki etkisini gösterir. Yapay sinir ağlarında bu ağırlıklar, uzun dönemlik hafızaya karşılık gelir. Bir yapay sinir ağı bu ağırlıkların tekrar tekrar ayarlanması sayesinde öğrenir.

3.1.5. Toplama fonksiyonu

Toplama Fonksiyonu Σ simgesi ile ifade edilen, hücreye gelen net girdinin toplanmasını sağlamaktadır. Toplama fonksiyonu, adından da anlaşılacağı gibi, genelde toplama işlemi yapılmaktadır. Fakat birçok farklı işlem çeşidi toplama fonksiyonu için kullanılabilir. Toplama fonksiyonu, bu basit çarpımlar toplamına ek olarak, minimum, maksimum, mod, çarpım veya çeşitli normalizasyon işlemlerinden birisi olabilir. Aşağıda yaygın olarak kullanılan aktivasyon fonksiyonları verilmiştir.

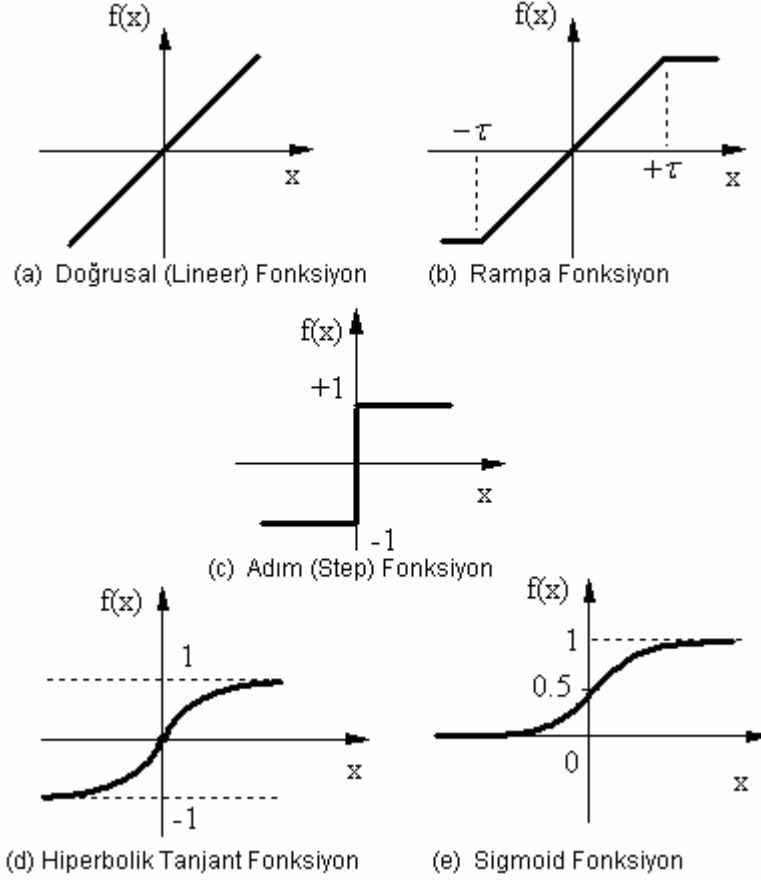
Tablo 3.1: Toplama fonksiyonları[2]

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

3.1.6. Aktivasyon fonksiyonu

Bu fonksiyon, f simgesi ile ifade edilir ve hücreye gelen net girdiyi bir fonksiyondan geçirerek hücrenin bu girdiye karşılık gelen çıktıyı belirler. Bu fonksiyon, aldığı değeri bir algoritma ile gerçek bir çıktıya dönüştürür. YSA'da çeşitli aktivasyon fonksiyonları kullanılmaktadır. Problem için en uygun aktivasyon fonksiyonu seçmek tasarımcının belirleyebileceği bir durumdur.

Aktivasyon fonksiyonu genellikle doğrusal olmayan bir fonksiyondur. Doğrusal fonksiyonlar genelde tercih edilmez çünkü doğrusal fonksiyonlarda çıktı, girdi ile orantılıdır. Bu durum, ilk YSA denemelerinin başarısızlıkla sonuçlanmasının da temel nedenidir.



Şekil 3.4: En sık kullanılan aktivasyon fonksiyonları[5]

Lineer fonksiyonun denklemi aşağıdaki gibidir.

$$f(x) = \alpha \cdot x \quad (3.2)$$

α : işlem elemanının x aktivitesini ayarlayan reel değerli bir sabittir[5].

Lineer fonksiyon $[-\tau, +\tau]$ sınırları arasında kısıtlandığında rampa aktivasyon fonksiyonu olur ve denklemi aşağıdaki gibi olur.

$$f(x) = \begin{cases} +\tau & : \text{Eğer } x \geq \tau \text{ ise} \\ x & : \text{Eğer } |x| < \tau \text{ ise (yani } -\tau < x < \tau) \\ -\tau & : \text{Eğer } x \leq -\tau \text{ ise} \end{cases} \quad (3.3)$$

$+\tau / -\tau$ işlem elemanının maksimumu (minimumu) çoğu zaman doyma seviyesi olarak adlandırılan çıkış değeridir. Eğer aktivasyon fonksiyonu bir giriş işaretine bağlı ise yaydığı $+\tau$ giriş toplamı pozitif, bağlı değilse adım fonksiyonu $[-\delta]$ olarak adlandırılır.

Aşağıda adım aktivasyon fonksiyonunun denklemini gösterilmiştir[5].

$$f(x) = \begin{cases} +\tau & : \text{Eğer } x > 0 \text{ ise} \\ -\delta & : \text{Diğer durumlar} \end{cases} \quad (3.4)$$

Hiperbolik tanjant fonksiyonunun matematiksel gösterimi aşağıdaki gibidir.

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} = \frac{e^{2x} - 1}{e^{2x} + 1} \quad (3.5)$$

Sigmoid fonksiyonu seviyeli, doğrusal olmayan (non-linear) sınırlı monoton artan logaritmik bir fonksiyondur. Giriş değerleri hangi aralıkta olursa olsun, çıkış 0 ile 1 arasında olmaktadır. Türevlenebilir olduğu için geriye yayılım algoritmaları ile kullanılabilir. Doğrusal olmayan problemlerin çözümünde tercih edilmektedir. Denklemi aşağıda verilmiştir.

$$f(x) = \frac{1}{1 + e^{-x}} \quad (3.6)$$

Her işlem elemanı kendisine verilen yerel veriye göre, kendisini ayarlayacak bütün YSA'nın enformasyon bölgesinin öğrenmesini sağlar. (Enformasyon bölgesi olasılık yoğunluk fonksiyonu ile de tanımlanabilir). Enformasyon bölgesi birçok uygulamada, gerçek değer "0" ile "1" arasında normalize edilmesi gerekir. (Normalize etmek: gerçek değeri 85 olan bir girişi 0.85 şeklinde ağa uygulamaktır.) Normalizasyon aynı anda bütün girişlere uygulanabilir.

3.1.7. Hücre çıktısı

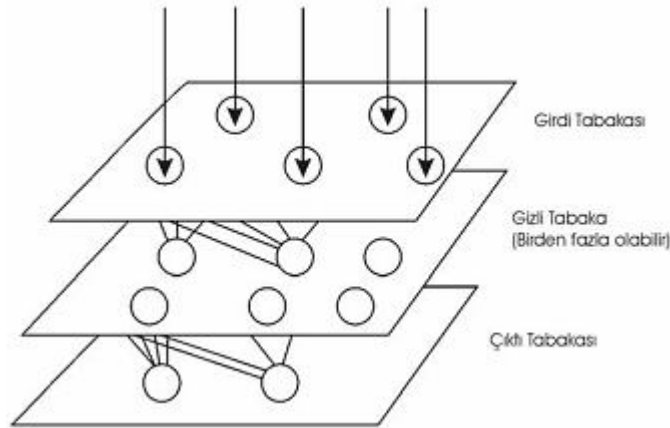
Aktivasyon fonksiyonu tarafından belirlenen çıktı değeridir ve "y" simgesi ile ifade edilmektedir. Temel olarak, bir yapay sinir hücresinin iç aktivasyonu yada ham çıktısı, girdilerin ağırlıklandırılmış toplamıdır. Ancak genelde son değeri belirlemek için bir eşik fonksiyonu da kullanılır. Çıktı 1 ise, sinir hücresi harekete geçer (aktif hale gelir). Çıktı 0 ise, sinir hücresi harekete geçmez.

Üretilen çıktı dışarıya veya başka bir hücreye gönderilir. Hücre kendi çıktısını kendisine girdi olarak da gönderebilir. Bir işlem biriminden çıkan tek bir çıktı elemanı vardır. Aynı değer birden fazla işlem birimine girdi olarak gitmektedir.

3.2. Yapay Sinir Ağı Bağlantı Yapısı

Tüm YSA' lar, temel elemanlarından bahsettiğimiz nöronlardan oluşturulurlar. Bu nöronların dizaynı, sinir ağının, başka bir deyişle mimarisinin oluşturulmasının ilk aşamasıdır. İkinci aşaması ise bu işlem elemanlarının kümelenmesi ve birbirleri arasındaki bağlantıların oluşturulmasını içerir.

YSA' ların ilk yıllarında, bazı araştırmacılar, nöronlar arasındaki bağlantıları rassal olarak oluşturmuşlar ve olumsuz sonuçlarla karşılaşmışlardır. Bir yapıyı dizayn etmenin en kolay yolu elemanları katmanlandırmaktır. Burada katmanlandırmanın üç bölümü vardır. Bunlar, nöronları katmanlar halinde gruplandırmak, katmanlar arasındaki bağlantıları gruplandırmak ve son olarak ise toplama ve aktivasyon fonksiyonlarını gruplandırmaktır. Genel olarak YSA' ların genel yapısı aşağıdaki gibidir.



Şekil 3.5: Basit bir YSA[8]

Çoğu uygulamalarda YSA' lar en az üç katmandan oluşur. Girdi katmanı, dışarıdan girdileri alan nöronları içerir. Çıktı katmanı ise çıktıları dışarı ileten nöronları içeren katmandır. Girdi ve çıktı katmanları tek katmandan oluşurken bu iki katman arasında birden fazla gizli katman bulunabilir. Bu gizli katmanlar çok sayıda nöron içerirler ve bu nöronlar tamamen ağ içindeki diğer nöronlarla bağlantılıdır.

3.3. YSA Özellikleri

Yapay sinir ağı modellerini klasik sistemlerden ayıran çeşitli özellikler vardır. Bunlardan bazıları;

3.3.1. Doğrusal olmama

Yapay sinir ağlarının temel elemanı olan yapay sinir hücrelerinin doğrusal sonuçlar vermemesinden dolayı bu özellik ağı da yansımıştır. Doğrusal olmamalarından dolayı yapay sinir ağları doğrusal olmayan sistemlerde tahmin yapma açısından istatistik hesaplamalarına göre daha kolay ve doğru sonuç vermesinden dolayı sık kullanılan bir yöntem haline gelmiştir.

3.3.2. Analiz ve tasarım kolaylığı

YSA' nın temel işlem elemanı olan hücrenin yapısı ve modeli, bütün YSA yapılarında yaklaşık aynıdır. Dolayısıyla, YSA' nın farklı uygulama alanlarındaki yapıları da standart yapıdaki bu hücrelerden oluşacaktır. Bu nedenle, farklı uygulama alanlarında kullanılan her bir YSA, benzer öğrenme algoritmalarını ve teorilerini paylaşabilirler.

3.3.3. Sınırsız sayıda değişken ve parametre

YSA modelleri sınırsız sayıda değişken ve parametre ile çalışabilirler. Bu sayede mükemmel bir öngörü doğruluğu ile genel çözümler sağlanabilmektedir. Bununla birlikte, YSA'ların kullanımında göz önünde bulundurulması gereken bazı dezavantajları da bulunmaktadır. Bunlar arasında en önemlisi sinir ağlarının eğitilebilmesine ve test edilebilmesine yetecek kadar geniş veri seti gereksinimidir. Ancak yine de, yeterli veri seti genişliği için belirli bir kural yoktur. Bununla ilgili belirli bir standart olmamakla birlikte her problem için farklı bir yaklaşım söz konusu olabilmektedir.

3.3.4. Hafıza

YSA'nın önemli bir özelliği bilgiyi saklama şeklidir. Bilgiler ağın tamamında saklanır. Geleneksel programlamada olduğu gibi bilgiler veri tabanları ya da dosyalarda belli bir düzende tutulmaz, ağın tamamına yayılarak ağın bağlantılarının değerleri ile ölçülür ve ağ bağlantılarında saklanır. Nöronlardan bazılarının işlevini yitirmesi, anlamlı bilginin kaybolmasına neden olmaz.

3.3.5. Genelleme

Öğrenme yeteneği sayesinde bilinen örnekleri kullanarak daha önce karşılaşılmamış durumlarda genelleme yapabilmektedir. Diğer bir deyişle ağ yapısının, eğitim esnasında kullanılan nümerik bilgilerden eşleştirmeyi betimleyen kaba özelliklerden çıkarım yapması ve böylelikle eğitim sırasında kullanılmayan girdiler için de, anlamlı yanıtlar oluşturabilmektedir. Aynı zamanda hatalı (noisy) veya kayıp veriler için de çözüm üretebilmektedir. Bu durum bir performans kaybı yaratmaz, performans kaybı eksik bilginin önemine bağlıdır. Burada bilgilerin önem dereceleri eğitim sırasında öğrenilir.

3.3.6. Öğrenme

Yapay sinir ağlarının temel işlevi bilgisayarın öğrenmesini sağlamaktır. Esin kaynağı olarak insan beyninin çalışma sistemi olan bu yöntemde, eğitime veya başlangıç tecrübesi sayesinde veriyi kullanarak öğrenme yeteneğine sahip olur. Böylece olayları öğrenerek benzer olaylar karşısında mantıklı kararlar verebilir. Bu sayede geleneksel teknikler göre çok karmaşık olan problemlere çözüm sunabilmektedirler.

3.3.7. Hata toleransı

YSA'da ağın sahip olduğu bilgi, ağdaki bütün bağlantılar üzerine dağılmıştır ve paralel dağılmış bir yapıya sahiptir. Bundan dolayı iyi eğitilmiş ve genelleme kapasitesi yüksek bir YSA' da bazı bağlantılarının hatta bazı nöronların etkisiz hale

gelmesi, ađın dođru bilgi üretmesini önemli ölçüde etkilemez. Bu da YSA'nın geleneksel yöntemlere göre daha toleranslı olmasını sağlar.

3.3.8. Paralellik

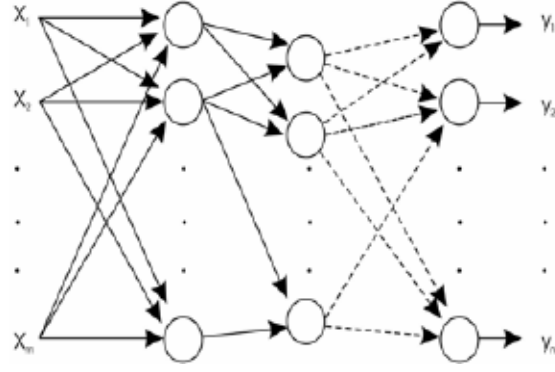
YSA'lar bilgi işleme yöntemleri bakımından geleneksel işlemcilerden farklıdır. Geleneksel işlemcilerde, tek bir merkezi işlem elemanı her hareketi sırasıyla gerçekleştirir. YSA'da ise çok sayıdaki nöron eş zamanlı şekilde çalışarak karmaşık işlevleri yerine getirir. Ayrıca YSA, hesaplamaları paralel olarak yürütülebildiğinden gerçek zamanlı işlem yapabilir.

3.4. Ađın Mimari Yapısına Göre YSA

İleri beslemeli sinir ađlarında, işlem elemanları arasındaki bağlantılar bir döngü oluşturmazlar ve bu ađlar girdi veriye genellikle hızlı bir şekilde karşılık üretirler. Geri beslemeli ađlarda (Recurrent Networks) ise bağlantılar döngü içerirler ve hatta her seferinde yeni veri kullanabilmektedirler. Bu ađlar, döngü sebebiyle girdinin karşılığını yavaş bir şekilde oluştururlar. Aşağıda ileri beslemeli çok katmanlı ađ şekli gösterilmektedir.

3.4.1. İleri beslemeli çok katmanlı YSA

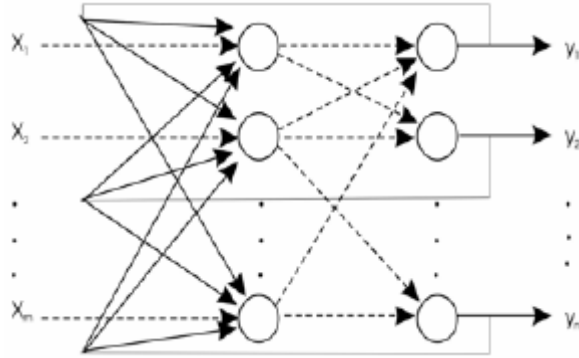
İleri beslemeli YSA'da, işlem birimleri katmanlar şeklinde düzenlenir ve bir katmandaki hücrelerin çıkışları bir sonraki katmana ağırlıklar üzerinden giriş olarak verilir. Giriş katmanı, dış ortamlardan aldığı bilgileri hiçbir değişikliğe uğratmadan orta (gizli) katmandaki işlem birimlerine iletir.



Şekil 3.6: İleri beslemeli çok katmanlı YSA[9]

3.4.2. Geri beslemeli çok katmanlı YSA

Geri beslemeli yapay sinir ağlarında, en az bir işlem biriminin çıkışı kendisine ya da diğer işlem birimlerine giriş olarak verilir ve genellikle geri besleme bir geciktirme elemanı üzerinden yapılır. Geri besleme, bir katmandaki işlem birimleri arasında olduğu gibi katmanlar arasındaki işlem birimleri arasında da olabilir.



Şekil 3.7: Geri beslemeli çok katmanlı YSA[9]

3.5. YSA'da Eğitim

3.5.1. Eğitim algoritmaları

Bir ağ yapılandırıldıktan sonra eğitime hazır durumdadır. Bu aşamada ağ ağırlıkları belirlenir. Bunu yaparken YSA'nın vazgeçilmez bir parçası olan eğitime algoritmaları kullanılır. Eğitim algoritması, eldeki problemin özelliğine göre öğrenme kuralının

YSA'ya nasıl uyarlanacağını belirtir. Yaygın olarak kullanılan üç çeşit eğitim algoritması vardır. Bunlar:

1- Öğreticili eğitim (Supervised training): Giriş-çıkış örnek çiftlerinin alındığı, bilinmeyen bir giriş fonksiyonuna bu örnekler yardımıyla yakınsanmasını sağlayan bir bağlantı ağırlık değişim kuralıdır. Öğreticili eğitimde, elde doğru örnekler vardır. Bu yöntemde hem giriş hem de çıkış verileri kullanılır. Öncelikle, ağırsal olarak belirlenen başlangıç ağırlıklarını kullanarak giriş değerlerini işler ve çıkış değerini istenilen çıkış değeri ile karşılaştırır. Elde edilen hatalar sistem içinde geriye gönderilir ve bu hatalar kullanılarak ağı kontrol eden bağlantı ağırlıkları güncellenir. Bu işlem defalarca tekrarlanır ve bağlantı ağırlıkları sürekli olarak ayarlanır.

2- Skor ile eğitim(Graded training): Skor ile eğitimde giriş değerlerine karşılık gelen çıkış değerleri tam olarak bilinmemektedir. Çıkış değeri yerine skor verilir ve ağı değerlendirilmesi yapılır. Özellikle kontrol uygulamaları için idealdir.

3- Kendini düzenleme ile eğitim (Self-organization training): Bu eğitim tipine adaptif eğitim de denir. Giriş örneklerinin kümelendirilmesini sağlayan bir bağlantı ağırlık değişim kuralıdır. Probleme ait veri girişleri ağı verilir ancak buna karşın istenen çıkışlar ağı verilmez. Sistem giriş verilerini gruplandırmak için hangi özellikleri kullanacağına kendi kendisine karar verir ve böylelikle giriş verilerine göre kendini düzenleyerek organize eder. Bu eğitim tipi her ne kadar olasılık yoğunluk fonksiyonlarına, sınıflandırma ve şekil tanıma problemlerine uygulanabilse de günümüzde bu yöntem deney alanlarıyla sınırlı kalmıştır.

3.5.2. Eğitimin sınaması

Eğitim sırasında eğitim kümesine ihtiyaç vardır. Eğitim kümesi, YSA' ya öğrenme için uygulanacak olan girişlerden ve istenen çıkışlardan oluşmaktadır. Bunların bir kısmı eğitim aşamasında kullanılırken bir kısmı da sınamaya aşamasında ağı genelleştirme yeteneğinin teyidi amacıyla kullanılır. Sınamanın başarısızlığı durumunda, sınamaya amacıyla kullanılan verilerin bir kısmı eğitim verisine katılarak eğitim ve sınamaya işlemleri kabul edilebilir bir performans kriterine kadar tekrarlanır. Yapay sinir ağı eğitiminde karşılaşılan temel bir sorun ezberlemedir. Yapay sinir

ağının eğitim sürecindeki hata seviyesi, test sürecindeki hata seviyesine göre bariz farklılıklar gösterdiği takdirde ezberleme sorunu ile karşılaşmış olur. Bu da tanımlanması istenen fonksiyel ilişkiden ziyade eğitim verisindeki gürültü gibi tuhaflıkların da öğrenildiği anlamına gelir. Ezberlemeyi azaltmak için yapılabilecekler:

- Eğitimde kullanılan verilerin sayısını arttırarak gürültünün ortalamasının kendiliğinden düşmesini sağlamak,
- Serbest parametre olan nöronların sayısını kullanılması gerekenin asgarisi ile sınırlamak,
- Eğitimi, ezberleme başlamadan kesmek. Bu aşamada, ağ veri setinden hatalı ilişkiler çıkartmaya başlayabilir. Bu yüzden eğitime uzunluğu iyi ayarlanmalıdır [6].

3.6. Öğrenme Kuralları

Bilginin kurallar şeklinde açıklandığı klasik uzman sistemlerin tersine, YSA gösterilen örnekten öğrenerek kendi kurallarını oluşturur. Öğrenme; giriş örneklerine veya (tercihen) bu girişlerin çıkışlarına bağlı olarak ağın bağlantı ağırlıklarını değiştiren veya ayarlayan öğrenme kuralı ile gerçekleştirilir. Günümüzde kullanılan birçok öğrenme kuralı vardır. Bilinen ve en çok kullanılan öğrenme kuralları şunlardır:

- Rastlantısal (Hebb) öğrenme kuralı,
- Performans (Widrow ve ADALINE) öğrenme kuralı,
- Kompetitif (Kohonen) öğrenme,
- Filtreleme (Grossberg),
- Spotitemporal öğrenme,
- Genelleştirilmiş delta kuralı öğrenme.

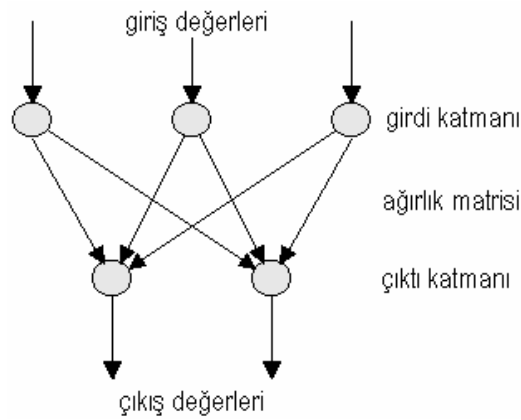
3.7. YSA Modelleri

Günümüzde pek çok YSA modeli kullanılmaktadır. Bunlar kullandıkları aktivasyon fonksiyonları, öğrenme biçimi ve öğrenme kuralları bakımından farklılık gösterirler ve bu farklılıklar ağın modelini belirler. Buna göre en çok kullanılan YSA modellerinden bazıları aşağıda verilmiştir. Bunlar:

- Tek Katmanlı Algılayıcı(Perceptron),
- Çok Katmanlı Algılayıcı(Multilayer Perceptron)
- Hopfield Ağı
- Kohonen-Self Organizyon Ağları
- Radyal Tabanlı Fonksiyon Ağları (Radial Basis Function Networks-RBFN)
- Uyarlanmış Rezonans Teorisi (ART)

3.7.1. Tek katmanlı algılayıcı (perceptron)

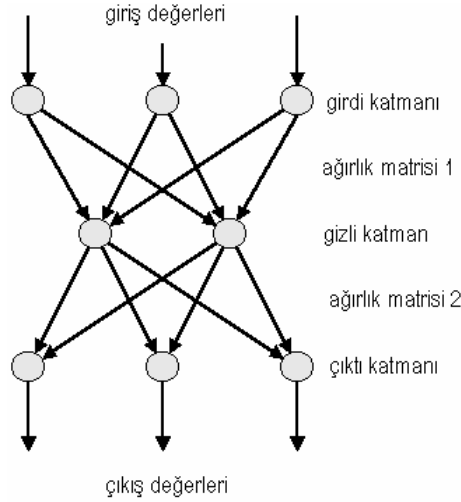
Bir giriş katmanı ve bir çıkış katmanı vardır ve bilgi girişten çıkışa doğru ilerler yani ağ ileri beslemedir. Tek katmanlı olarak isimlendirilmesinin sebebi, giriş katmanının veri üzerinde hiçbir işlem yapmadan veriyi çıkış katmanına iletmesidir yani ağırlıklandırılmış girdiler aktivasyon fonksiyonuna uygulanır ve +1 ya da -1 şeklinde bir çıktı alınır. Perceptron'un amacı girdileri sınıflandırmaktır.



Şekil 3.8: Perceptron[7]

3.7.2. Çok katmanlı algılayıcı (multilayer perceptron)

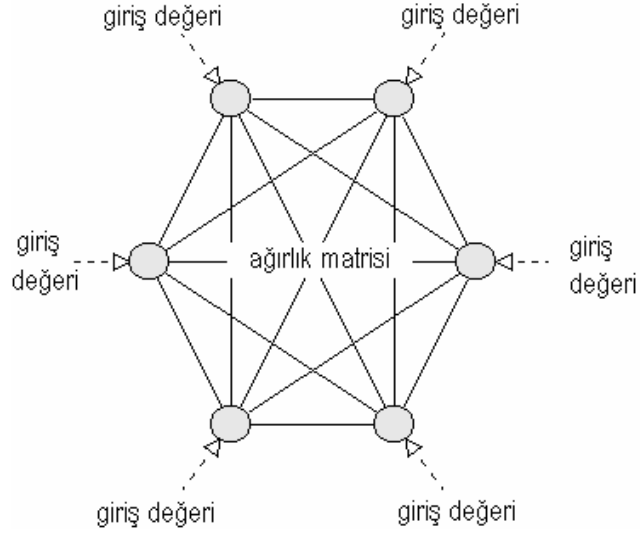
Çok katmanlı perceptron giriş ve çıkış katmanları arasında birden fazla katmanın kullanıldığı ileri yayımlı (feed-forward) YSA sistemidir. Gizli katman (hidden layer) olarak isimlendirilen bu ara katmanlarda, düğümleri giriş ve çıkış katmanlarına doğrudan bağlı olmayan işlem elemanları vardır. Çok katmanlı yapılarda (n). katmanın çıkış sinyalleri (n+1). katmanın giriş sinyalleri olarak kullanılır. Çok katmanlı ağlarda, veriler giriş katmanı tarafından kabul edilirler. Ağ içinde yapılan işlemler sonucunda çıkış katmanında oluşan sonuç değer, istenen cevap ile karşılaştırılır. Bulunan cevap ile istenen cevap arasında herhangi bir ayrılık varsa, ağırlıklar bu farkı azaltacak şekilde yeniden düzenlenir. Ağırlık düzeltme işlemi çıkış seviyesine bağlı ağırlıklardan başlar ve işlem ters yönde, giriş seviyesine varana kadar devam eder. Sonuçta sistem hatalar yapar, ama bu hatalardan birşeyler öğrenip isteneni bulana kadar işleme devam eder.



Şekil 3.9: Multilayer perceptron[10]

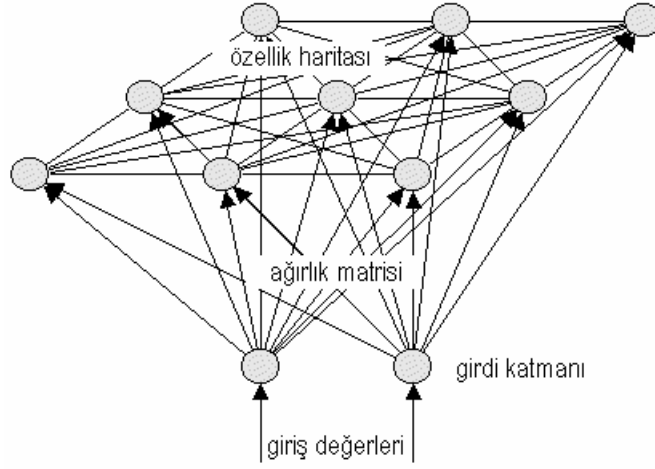
3.7.3. Hopfield ağı

Tek katmanlı, geri dönüşümlü bir ağıdır. Proses elemanlarının tamamı hem girdi hem de çıktı elemanıdır. Ağın bağlantıları enerji fonksiyonudur. Her bir düğüm diğerleriyle bağlanır ve başlangıç değerleri +1 ya da -1'e ayarlanır, bir noktada birleşene kadar bir dizi devir alır ve stabil(kararlı) hale gelir.



Şekil 3.10: Hopfield ağı[10]

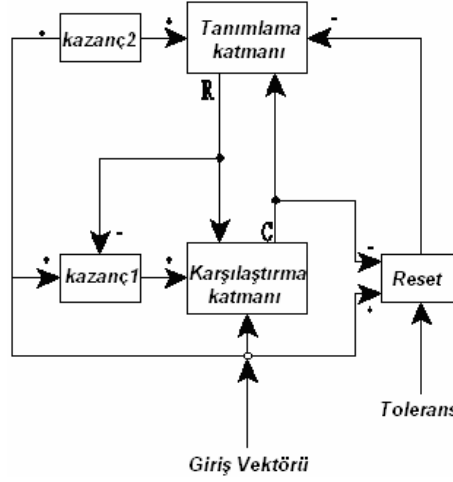
3.7.4. Kohonen-self organizyon ağları



Şekil 3.11: Kohonen ağı[10]

Bir girdi katmanı ve bir özellik haritası katmanından oluşur. Ayrı bir çıktı katmanı yoktur. Şebekede her bir düğüm çıktı düğümünün kendisidir. Bu algorithmada giriş verisi sınırlandırılabilir, şebeke düğümleri organize olur. Özellik haritasında her düğümde depolanan girdi örnekleri karşılaştırma işleri tarafından organize edilir ve düğümde biriktirilir.

3.7.5. Uyarlanmış rezonans teorisi (ART)



Şekil 3.12: Basitleştirilmiş ART network[7]

Art networku bir vektör sınıflamasıdır. Giriş vektörü alınır kategori numaraları birbirlerine göre sınıflandırılır, benzer örneklere bağlı olarak saklanır. Sınıflandırma tek katmanlı tanıma olarak yapılır. Eğer giriş vektörü herhangi bir depolanmış deseni eşlemezse yeni kategori giriş vektörü istenilen deseni depolayarak oluşturur. Depolanan desen özelleşmiş tolerans içinde giriş vektörünü eşleyenbulduğundan o desen giriş vektörü olarak ayarlanır. Ağ her bir sınıflandırma kategorisi için tanıma katmanı sinir hücresi ile ilgili ağırlıklarda bir takım desen depolanır. Ağırlık ile tanıma katman sinir hücresini en iyi eşleme uygulanan vektör ateşler. Giriş vektörü X karşılaştırma katmanı uyarım vektörü P ile karşılaştırılır. Tanıma katmanında yukarı-aşağı geri besleme olması engellenir. Giriş vektörü X reset blokla karşılaştırılır ve reset sinyali benzerden daha düşük düzeyde ortaya çıkarsa, uygulanır[7].

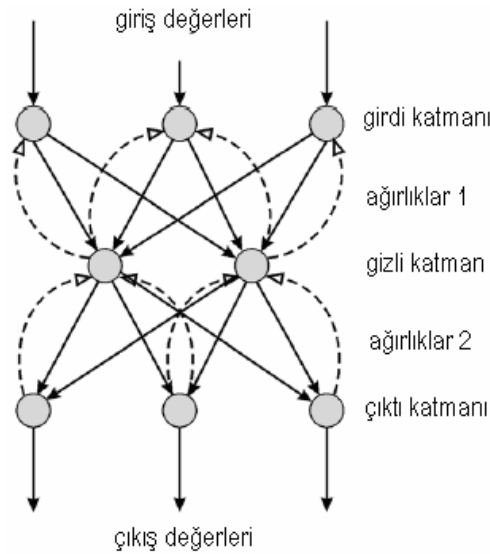
3.7.6. Radyal tabanlı fonksiyon ağları (radial basis function networks-RBFN)

Radyal tabanlı fonksiyon ağları (Radial Basis Function Networks-RBFN), çok boyutlu uzayda eğri uydurma yaklaşımıdır. RBFN'nin eğitimi, çok boyutlu uzayda eğitim verilerine en uygun bir yüzeyi bulma problemidir. RBFN, yapı olarak BPN ile benzerdir. Girdi katmanı, ara katman ve çıktı katmanından oluşur. Ara katmandaki radyal tabanlı aktivasyon fonksiyonu (gauss, üstel) girdileri dönüştürmeyi sağlar. Çıktı katmanında doğrusal bir aktivasyon fonksiyonu vardır. RBFN'nin eğitimi,

BPN'nin eğitiminden daha az bir zaman gerektirir. Ancak, BPN kadar doğru sonuçlar verebilmesi için daha fazla veriye ihtiyaç duymaktadır.

3.8. Geriye Yayılım Algoritması (Back Propagation Networks-BPN)

Uygun ağırlıkların bulunması için en sık kullanılan yöntem geri yayılma algoritmasıdır. Geriye yayılım algoritması, bir hedef fonksiyonu minimize etmek üzere tasarlanmış optimizasyon tekniğidir yani elde edilen çıkışlar ile elde edilen hedef çıkışlar arasındaki hataların karesinin ortalamasını minimum yapmak için geliştirilmiş iteratif bir gradyan algoritmasıdır. Hataları çıkıştan girişe geriye doğru azaltma eğilimindedir.



Şekil 3.13: Geri yayılmalı ileri beslemeli ağ yapısı[8]

Şekilde, bir girdi, bir gizli ve bir çıktı katmanı olmak üzere 3 katmanlı bir geri yayılma ağ yapısı gösterilmektedir. Katmanlar halinde düzenlenmiş daireler işlem elemanlarını yani sinir hücrelerini temsil etmektedir. Girdi katmanında üç nöron bulunmakla birlikte bunlar ağa üç değişkenli girdi olarak tanıtılmaktadır. Bunun yanında gizli katmanda iki, çıktı katmanında ise üç nöron bulunmaktadır. Dolayısıyla, ağdan üç değişken olarak çıktı alınmaktadır. Girdi katmanından, gizli katmana iletilen değerler "Ağırlıklar 1" ağırlık seti ile, gizli katmandan çıktı katmanına iletilen değerler ise "Ağırlıklar 2" ağırlık seti ile ağırlıklandırılmaktadır. Ağ yapısında, kalın oklar kullanma sırasındaki bilgi akışını simgelemektedir.

Kullanma, eğitilmiş bir ağa yeni girdi verilerin sunulması ve çıktısının alınması işlemidir. Eğitim aşamasının bir basamağı hem ilerleme hem de geri yayılma safhalarını içerirken, kullanma aşamasında sadece ilerleme işlemi uygulanır. Zaten gerçek sonuçlar bilinmediğinden hatanın hesaplanıp geri yansıtılması mümkün değildir. Kısacası geri yayılma sadece eğitim esnasında kullanılır. Kullanma aşamasındaki algoritma eğitime aşamasına göre daha basittir. Dolayısıyla YSA'lar kullanma aşamasında eğitime aşamasına göre daha hızlı çalışırlar.

Geri yayılma ağlarda öğrenme kuralı olarak genellikle en küçük kareler yöntemine dayalı Delta Öğrenme Kuralının genelleştirilmiş hali olan genelleştirilmiş delta kuralı kullanılmaktadır Delta Kuralı iki safhadan oluşur. 1. safhaya, ileri doğru hesaplama safhası adı verilir ve bu safhada ağa gösterilen örnek seti için ağın çıktısı hesaplanır. 2. safhaya geriye doğru hesaplama safhası adı verilir; bu safhada ise ağın ağırlıkları güncellenir. Öğrenme işlemi süresince, ağ içinde ileri doğru bir bilgi akışı vardır ve katman katman her işlem elemanının çıktısı hesaplanır. Çıktı katmanına ulaşıldığında, bu katmanın çıktısı ile istenilen çıktı arasındaki fark hesaplanır ve bu hata önceki katmanlara iletilir (geri yayılma). Hatanın iletilmesi, katman katman geriye doğru olur ve bu süreçte Delta Kuralı ile bağlantı ağırlıkları ayarlanır. Bu işlem, girdi katmanına ulaşılan kadar devam eder ve bu noktada yeni bir döngüye başlar[8].

Geriye yayılma algoritması ilk olarak 1974 yılında Werbos tarafından geliştirilmesine rağmen ancak 1986'daki Rumelhart, Hinton ve Williams'ın yaptığı katkılarla uygulamalarda yer bulmuştur. Doğrusal olmayan yapı içeren problemlerde etkili olmalarından dolayı günümüzde pek çok uygulama alanında kullanılmaktadır.

3.9. YSA Uygulama Alanları

Yapay sinir ağları başlıca; sınıflandırma, modelleme ve tahmin uygulamaları olmak üzere, pek çok alanda kullanılmaktadır[9].

Bunlardan bazıları;

Uzay: Uçuş simülasyonları, otomatik pilot uygulamaları, komponentlerin hata denetimleri vs.

Otomotiv: Otomatik yol izleme, rehber, garanti aktivite analizi, yol koşullarına göre sürüş analizi vs.

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

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

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

Eğlence: Animasyonlar, özel efektler, pazarlama öngörüsü vs.

Finans: Kıymet biçme, pazar performans analizi, bütçe kestirimi, hedef belirleme vs.

Sigortacılık: Ürün optimizasyonu, uygulama politikası geliştirme vs.

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

Sağlık: Göğüs kanseri erken teşhis ve tedavisi, EEG, ECG, MR, kalite artırımı, ilaç etkileri analizi, kan analizi sınıflandırma, kalp krizi erken teşhis ve tedavisi vs.

Petro kimya: Arama, verim analizi vs.

Robotik: Yörünge kontrol, forklift robotları, görsel sistemler, uzaktan kumandalı sistemler, optimum rota belirleme vs.

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

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

Güvenlik: Parmak izi tanıma, kredi kartı hileleri saptama, retina tarama, yüz eşleştirme vs.[9].

3.10. YSA'ların Tarihsel Gelişimi

YSA'lar ile ilgili ilk adım 1943 yılında bir sinir hekimi olan Warren McCulloch ile bir matematikçi olan Walter Pitts tarafından gerçekleştirilen yapay sinir ağı modeli ile başlamıştır. McCulloch ve Pitts, insan beyninin hesaplama yeteneğinden esinlenerek, elektrik devreleriyle basit bir sinir ağı modellemiştir. 1949 yılında ise Hebb, hücre bağlantılarını ayarlamak için ilk öğrenme kuralını önerdi. Hebb kuralı; sinir ağının bağlantı sayısı değiştirilirse, öğrenebileceğini ön görmekteydi.

1958 yılında Rosenblatt Perceptron modeli ve öğrenme kuralını geliştirdi[8].

1959 yılında Bernard Widrow ve Marcian Hoff (Stanford Üniversitesi) ADALINE (Adaptive Linear Neuron) modelini geliştirmişler ve bu model YSA'ların mühendislik uygulamaları için başlangıç kabul edilmiştir.

Minsky ve Papert, 1969'da Algılayıcıların (Perceptron) yetersizliğini görmüşler ve çok katmanlı modeli önermişlerdir. Ancak, gizli katmanın ağırlıklarının nasıl değiştirileceği konusunda bir yöntem önerememişler. Bu durum kaynakların kesilmesine ve YSA çalışmalarının durmasına neden olmuştur[10].

Tüm bu olumsuzluklara rağmen bazı bilim adamları çalışmalarına devam etmiş, 1976'da Grosberg Carpenter Adaptif Rezonans Teorisi (ART), 1982'de Hopfield, Optimizasyon gibi teknik problemleri çözmek için lineer olmayan dinamik Hopfield ağını geliştirmiştir[8].

1982-1984 yıllarında Kohonen Kendi Kendine Öğrenme Nitelik Haritaları (Self Organizing Feature Maps SOM) konusundaki çalışmasını yayınlamıştır. Kendi adını taşıyan eğitimci olmayan öğrenen bir ağ geliştirmiştir[10].

1986'da Rumelhart, esas olarak 1974 yılında Werbos tarafından bulunan çok katmanlı algılayıcı tipi ağlar için geriye yayılım algoritmasını geliştirmiştir.

1988 yılında, Broomhead ve Lowe radyal tabanlı fonksiyonlar modelini (Radial Basis Functions RBF) geliştirmişler ve özellikle filtreleme konusunda başarılı

sonular elde etmiřlerdir. Daha sonra Spect, bu ađların daha geliřmiř řekli olan Probabilistik ađlar (PNN) ve Genel Regresyon Ađlarını (GRNN) geliřtirmiřtir[10].

YSA'lar ile ilgili alıřmalara 1950'li yıllarda bařlanmasına rađmen, ancak 1980'li yılların ortalarında genel amalı kullanım iin yeterli seviyeye gelmiřlerdir. Bugün ise bilgisayar sistemlerinin geliřimine de paralel olarak pek ok alanda kullanılır hale gelmiřtir.

BÖLÜM 4. YAPAY SİNİR AĞLARI İLE KOCAELİ İLİNİN PUANT YÜK TAHMİNİNİN YAPILMASI

Bu çalışmada Kocaeli ilinin puant yük tahmini yapılmıştır. Bu tahmin yapılırken yapay sinir ağları(YSA) metodu kullanılmıştır. Bunun sebebi yük tahmininin doğrusal yapıda olmaması ve YSA'ların da bu tür doğrusal olmayan yapıları dikkate alabilmesidir. YSA'ların bu özelliği bu çalışmanın içeriği açısından da ayrıca önem taşımaktadır.

Yapılan birçok çalışma geleceği tahminde, yapay sinir ağlarının (YSA) en az geleneksel metotlar kadar, hatta daha iyi neticeler verdiğini göstermiştir[11]. YSA'nın geleceği tahmin için kullanıldığı alanlardan biri de elektrik enerjisi talep tahminidir. Bu çalışmada YSA modeli ile geçmişteki veriler incelenerek gelecekte oluşabilecek durumlar tahmin edilmeye çalışılmıştır. Aynı şekilde bu tahmin modeli MATLAB programının Sinir Ağları Alt Yazılımı (Neural Network Toolbox) kullanılarak oluşturulmuştur. Ayrıca nüfus ve sıcaklık gibi çevresel etkenler de bu çalışmaya dahil edilerek daha gerçekçi sonuçlara ulaşılması hedeflenmiştir.

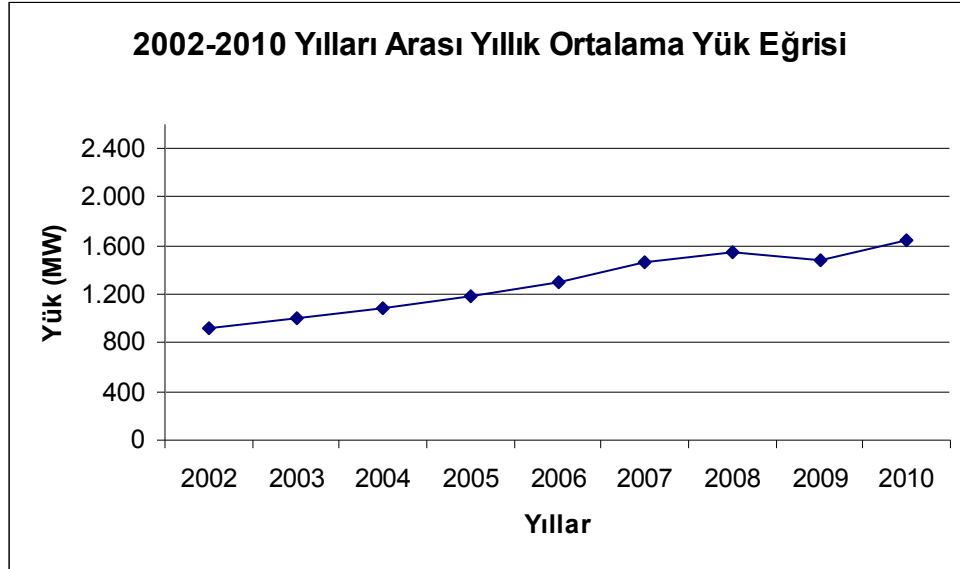
4.1. Yük Tahmin Verileri

4.1.1. Yük verileri

Adapazarı Türkiye Elektrik İletim Anonim Şirketinden (TEİAŞ) temin edilen 2002-2010 yılları arasında kaydedilen Kocaeli ili aylık puant yük değerleri Tablo 4.1.'de verilmiştir.

Tablo 4.1: Kocaeli ilinin aylık puant yük verileri

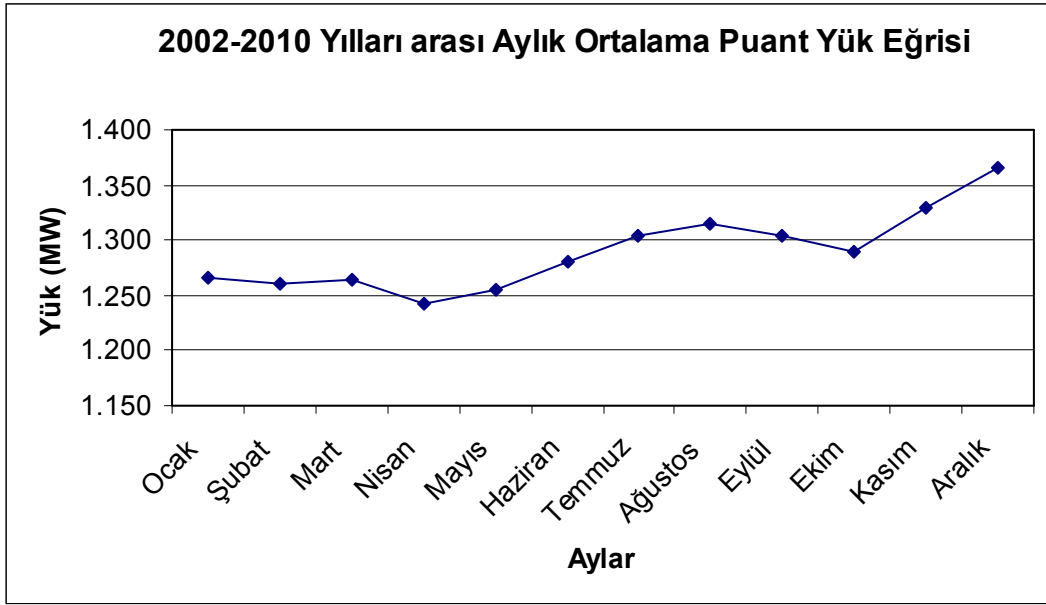
PUANT YÜK DEĞERLERİ (MW)												
AY/YIL	OCAK	ŞUBAT	MART	NİSAN	MAYIS	HAZİRAN	TEMMUZ	AĞUSTOS	EYLÜL	EKİM	KASIM	ARALIK
2002	1.011	876	921	930	848	878	932	906	907	912	939	1.017
2003	1.004	978	999	967	967	972	973	982	1.019	1.034	1.034	1.096
2004	1.087	1.066	1.061	1.042	1.080	1.058	1.097	1.073	1.076	1.069	1.148	1.172
2005	1.137	1.137	1.101	1.098	1.166	1.180	1.198	1.218	1.190	1.169	1.257	1.295
2006	1.273	1.267	1.220	1.257	1.255	1.290	1.281	1.305	1.290	1.288	1.352	1.433
2007	1.426	1.428	1.414	1.356	1.400	1.421	1.442	1.481	1.475	1.514	1.565	1.581
2008	1.603	1.618	1.554	1.520	1.562	1.606	1.580	1.594	1.557	1.474	1.469	1.340
2009	1.417	1.438	1.511	1.418	1.424	1.495	1.513	1.506	1.534	1.428	1.534	1.549
2010	1.437	1.536	1.595	1.600	1.598	1.628	1.728	1.772	1.684	1725	1667	1812



Şekil 4.1: 2002-2010 yılları arası yıllık ortalama puant yük eğrisi

Şekil 4.1.'deki eğriye göre yıllık ortalama puant yük değerlerinde yıllar itibariyle sürekli bir artış gözlenmektedir. Bunun nedeni olarak bölgenin sürekli göç alması ve

sanayinin hızla gelişmesi gösterilebilir. Ancak 2008 yılından itibaren yük miktarında bir düşüş gözlenmektedir. Bunun nedeni olarak özellikle 2008 yılının son aylarında etkisini hissettiren küresel krizin Kocaeli ilini önemli ölçüde etkilemesi gösterilebilir. Buna göre bu dönemde birçok kuruluş ya kapanmış ya da üretimini yavaşlatmıştır. Bu da elektrik enerjisi tüketimini doğrudan etkileyerek ciddi bir düşüşe neden olmuştur.



Şekil 4.2: 2002-2010 yılları arası aylık ortalama puant yük eğrisi

Şekil-4.2, 2002-2010 yılları arasındaki aylık ortalama puant yük miktarlarını göstermektedir. Buna göre bahar aylarına doğru yük miktarlarında bir azalma gözlenmekte, bu aylardan sonra ise bir artış görülmektedir. Ancak bu grafiğe göre eylül-ekim aylarına gelindiğinde bir düşüşle karşılaşmaktadır. Bunun nedeni, küresel krizin bu ayları ciddi şekilde etkilemesi ve ortalamayı azaltmasıdır.

4.1.2. Nüfus verileri

Bu çalışmada kullanılan Kocaeli ilinin aylık nüfus değerleri 2000-2007 yılları arasında yapılan nüfus sayım sonuçları dikkate alınarak, 4.1.'deki eşitlikteki doğal artış bağlantısının yardımı ile hesaplanmıştır[12].

$$P_n = P_0 e^{rn} \quad (4.1)$$

Bu bağıntıdaki terimlerden P_n iki ardışık sayımdan ikincisini, P_0 birincisini, n de iki sayım arasından geçen zaman birimi sayısını, r ise nüfus artış hızını göstermektedir. Türkiye İstatistik Kurumundan alınan 2000-2009 yılları itibariyle Kocaeli ilinin nüfus sayım bilgileri Tablo 4.2.'de verilmiştir.

Tablo 4.2: Kocaeli nüfus sayım bilgileri

NÜFUS BİLGİLERİ	
YILLAR	SAYIM
2000	1.206.085
2007	1.437.926
2008	1.490.358
2009	1.522.408

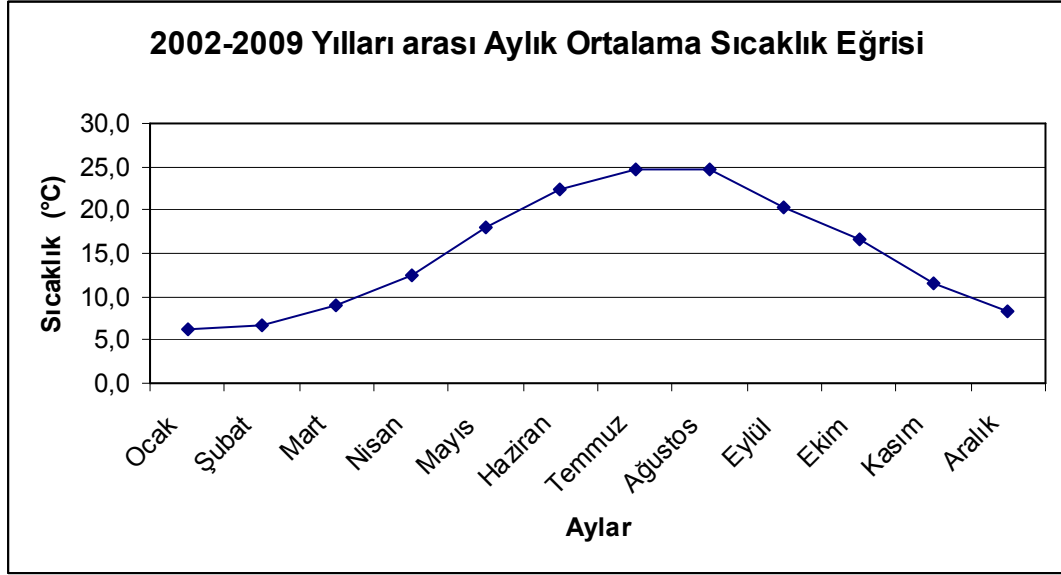
4.1.3. Sıcaklık verileri

Yükü etkileyen meteorolojik koşulların başında sıcaklık, nem, rüzgar hızı, yağış ve bulut yoğunluğu gelir. Sıcaklığa göre diğer faktörlerin yüke etkisi ihmal edilecek kadar azdır[12]. Bundan dolayı bu çalışmaya sadece sıcaklık verileri dahil edilecektir. Devlet Meteoroloji İşleri Genel Müdürlüğünden alınan Kocaeli ili 2002-2009 yılı aylık sıcaklık değerleri Tablo 4.3.'de verilmiştir.

Tablo 4.3: 2002-2009 yılı aylık ortalama sıcaklık değerleri

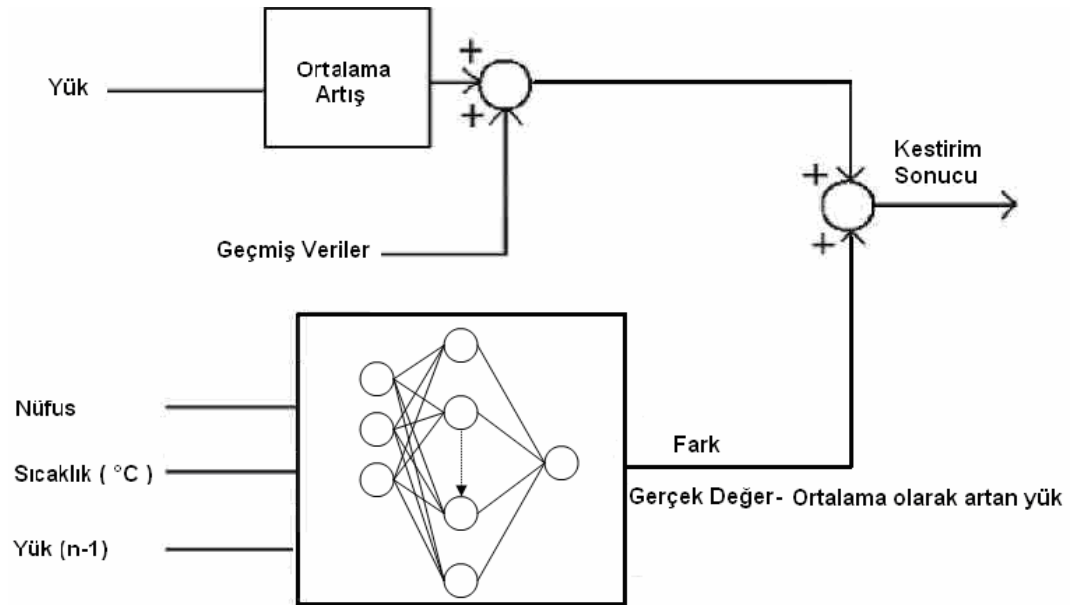
AYLIK ORTALAMA SICAKLIK DEĞERLERİ (°C)												
YILLAR	OCAK	ŞUBAT	MART	NİSAN	MAYIS	HAZİRAN	TEMMUZ	AĞUSTOS	EYLÜL	EKİM	KASIM	ARALIK
2002	4.8	9.9	10.2	11.6	17.1	22.7	26.1	23.5	20.9	16.7	12.9	6.4
2003	8.5	3.0	4.9	10.3	19.1	22.9	24.5	25.0	19.2	16.7	11.0	7.8
2004	5.6	5.8	9.5	13.0	16.8	21.4	24.3	23.5	20.7	17.4	12.1	9.0
2005	7.0	7.1	8.0	13.5	17.5	20.6	24.4	25.0	20.8	14.5	10.6	9.2
2006	4.1	6.0	9.7	13.1	17.5	21.9	23.3	26.3	20.2	16.7	10.4	7.3
2007	7.9	8.0	9.6	11.4	19.4	24.3	25.4	25.9	20.9	17.0	10.9	6.7
2008	4.1	6.1	12.0	15.7	17.6	22.8	24.3	25.7	20.5	16.5	13.3	9.2
2009	7.3	7.4	8.8	11.7	18.0	23.2	24.8	23.5	20.3	18.4	12.3	10.8

Aşağıdaki Şekil 4.3’de Kocaeli ilinin 2002-2009 yılları arasındaki ortalama aylık sıcaklık değişim eğrisi gösterilmektedir. Buna göre ortalama maksimum sıcaklık Temmuz ve Ağustos aylarında, minimum sıcaklık ise Ocak ayında gerçekleşmiştir.



Şekil 4.3: 2002-2009 yılları arası aylık ortalama sıcaklık eğrisi

4.2. YSA Modeli ve Uygulaması



Şekil 4.4: YSA mimarisi

Üç girdi değişkeni ve bir çıktı değişkeni içeren bu fonksiyonel yapıyı tahmin etmek için Şekil 4.4'deki gibi bir Geri Yayılmalı YSA mimarisi oluşturulmuştur. Şekil 4.4'te sunulan bu mimarinin girdi katmanında girdi değişkenlerin değerlerinin ağa sunulmasını sağlayan üç adet giriş ile çıktı katmanında bağımlı değişkene ait ağ çıktısının alındığı bir adet çıkış parametresi bulunmaktadır. Aynı şekilde bu model MATLAB programında da oluşturulmuştur. Modelin mimarisi oluşturulduktan sonra eğitim aşamasına geçilmiştir. Her ay için aylık 6 yılın (2002-2007) geçmiş yük, nüfus ve sıcaklık verileri eğitim sistemine girdi olarak verilmiştir. Yük tahmini için yukarıda yapılan modelde olduğu gibi gerçekleştirilen eğitim, her ay için ayrı ayrı yapılmış olup deneme yanılma yöntemi ile aktivasyon fonksiyonları ve eğitim algoritması belirlenmiştir. Bu uygulanan eğitim sonucunda en iyi sonuç veren parametreler ele alınmıştır. Ayrıca altı senelik gerçek yük verileri ile her yeni yük verisi eklendiğinde güncellenen ortalama yük miktarı da modele eklenmiştir. Meteorolojik veriler için 2002-2009 yılları arası aylık sıcaklık değerlerinin ortalaması kullanılmıştır. Aylık nüfus verileri, 2000-2009 yılları arasında yapılan nüfus sayım sonuçları dikkate alınarak 4.1'deki eşitliğe göre hesaplanmıştır. Hesaplanan bu değerler normalize edilerek tahmini olumsuz etkilemesi önlenmiştir. Bu şekilde bulunan geleceğe dönük sıcaklık ve nüfus verileri yük tahmini için oluşturulan yapay sinir ağı modeline sunulmuştur. Yapay sinir ağında nüfus, sıcaklık ve geçmiş yıllara ait yük verilerinin giriş parametresi olarak kullanılması ile çıkış olarak alınan 2002-2007 yılları arasındaki gerçek yük değeri ile ortalama yük değerleri arasındaki fark, hedef olarak belirtilmiştir. Eğitim sonrasında sonuçlar gerçek değerlerle karşılaştırılmıştır. Bu işlemlerden sonra eğitimin sınanması aşamasına geçilmiştir. Bunun için eğitimde kullanılmayan ancak elde mevcut olan 2008-2010 yılları arasındaki aylık yük verileri ağın doğruluğu için test edilmiştir. Böylece yük tahmini için yapılan modelin gerçeğe yakın sonuçlar verdiği saptanmıştır. Tablo 4.4'te eğitim için kullanılan yük verileri görülmektedir.

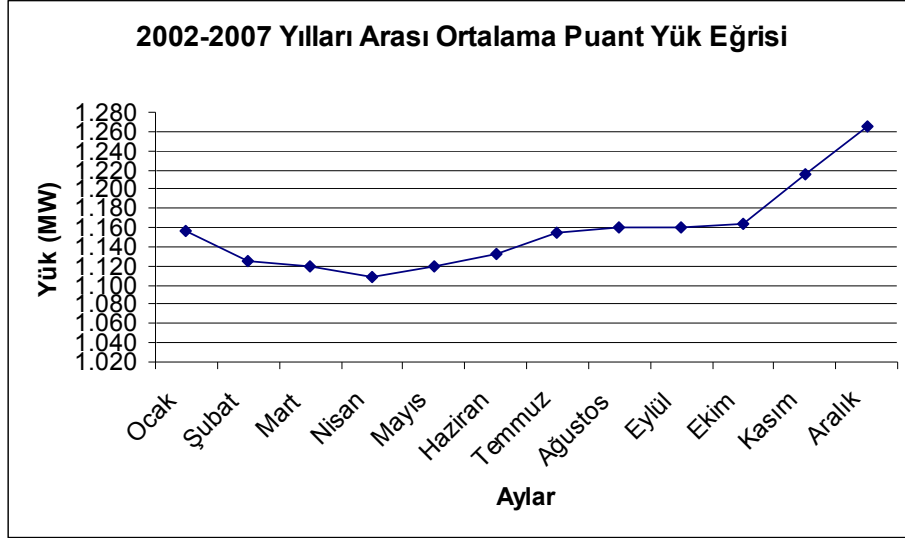
Tablo 4.4: 2002-2007 yılları arası aylık puant yük verileri

PUANT YÜK DEĞERLERİ (MW)												
AY/ YIL	OCAK	ŞUBAT	MART	NİSAN	MAYIS	HAZİRAN	TEMMUZ	AĞUSTOS	EYLÜL	EKİM	KASIM	ARALIK
2002	1.011	876	921	930	848	878	932	906	907	912	939	1.017
2003	1.004	978	999	967	967	972	973	982	1.019	1.034	1.034	1.096
2004	1.087	1.066	1.061	1.042	1.080	1.058	1.097	1.073	1.076	1.069	1.148	1.172
2005	1.137	1.137	1.101	1.098	1.166	1.180	1.198	1.218	1.190	1.169	1.257	1.295
2006	1.273	1.267	1.220	1.257	1.255	1.290	1.281	1.305	1.290	1.288	1.352	1.433
2007	1.426	1.428	1.414	1.356	1.400	1.421	1.442	1.481	1.475	1.514	1.565	1.581

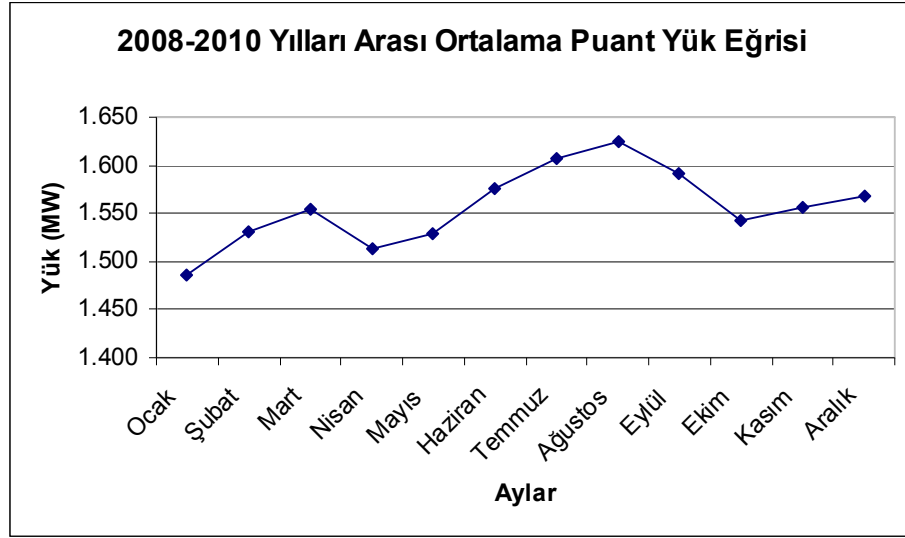
Tablo 4.5'te test için kullanılan yük verileri görülmektedir.

Tablo 4.5: 2008-2010 yılları arası aylık puant yük verileri

PUANT YÜK DEĞERLERİ (MW)												
AY/YIL	OCAK	ŞUBAT	MART	NİSAN	MAYIS	HAZİRAN	TEMMUZ	AĞUSTOS	EYLÜL	EKİM	KASIM	ARALIK
2008	1.603	1.618	1.554	1.520	1.562	1.606	1.580	1.594	1.557	1.474	1.469	1.340
2009	1.417	1.438	1.511	1.418	1.424	1.495	1.513	1.506	1.534	1.428	1.534	1.549
2010	1.437	1.536	1.595	1.600	1.598	1.628	1.728	1.772	1.684	1725	1667	1812



Şekil 4.5: Eğitim yük eğrisi



Şekil 4.6: Test yük eğrisi

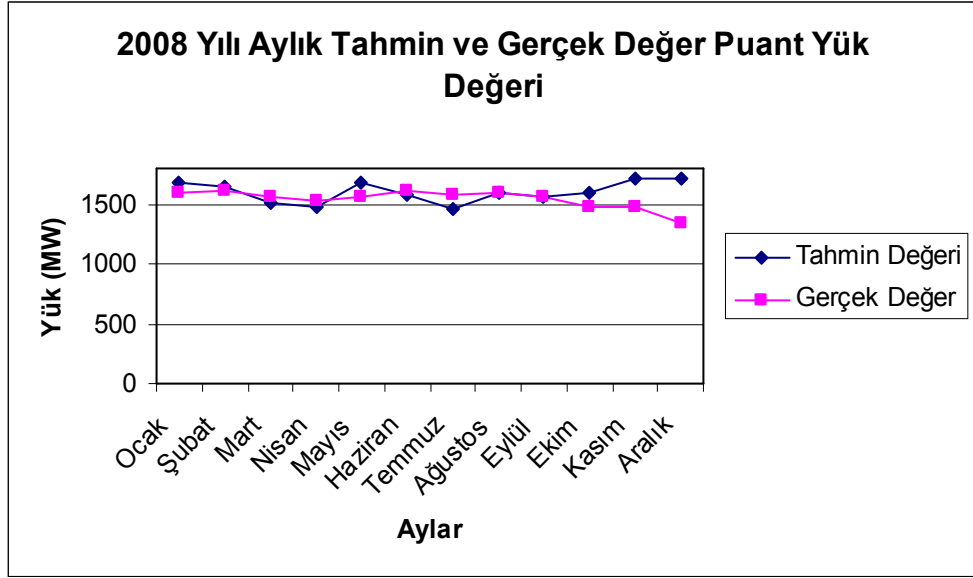
Şekil 4.5'te 2002-2007 yılları arası aylık ortalama yük değerlerini içeren eğitim yük eğrisi, Şekil 4.6'da 2008-2010 yılları arası aylık ortalama yük değerlerini içeren test yük eğrisi verilmiştir. Buna göre eğitim ve test verisi eğrilerinin ilk ve son aylarında farklılıklar görülmektedir. Eğitim yük eğrisinin ilk aylarında bir düşüş gözlemlenirken, son aylarında bir yükselme gözlenmektedir. Buna karşın test verisinin ilk aylarında bir yükselme görülürken, son aylarında ise bir düşüş kaydedilmiştir. Bu farklılığın test aşamasını olumsuz etkileyeceği bilinmeli, değerlendirme yapılırken bunların göz önüne alınması gerektiği unutulmamalıdır.

Tablo 4.6: Gerçek ve tahmin puant yük değerleri ile hata oranları

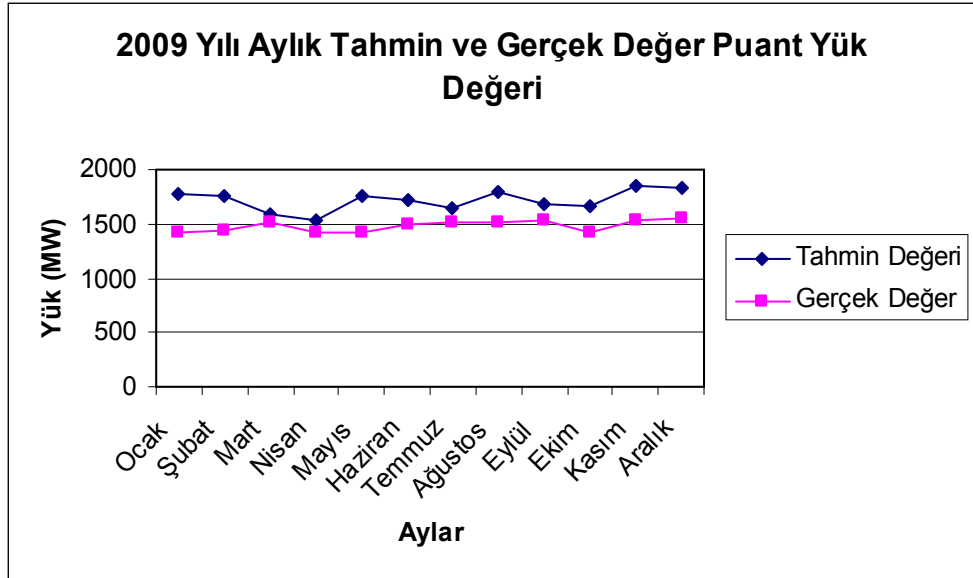
Aylar	2008 Gerçek (MW)	2008 Tahmin (MW)	Hata Oranı (%)	2009 Gerçek (MW)	2009 Tahmin (MW)	Hata Oranı (%)	2010 Gerçek (MW)	2010 Tahmin (MW)	Hata Oranı (%)
OCAK	1603	1677,9	5	1417	1779,6	26	1437	1877,3	31
ŞUBAT	1618	1640,2	1	1438	1765,7	23	1536	1890	23
MART	1554	1512,1	3	1511	1588,1	5	1595	1684,6	6
NİSAN	1520	1476,4	3	1418	1530	8	1600	1727,5	8
MAYIS	1562	1672,8	7	1424	1751,5	23	1598	1822,1	11
HAZİRAN	1606	1577,5	2	1495	1724,6	15	1628	1841,6	13
TEMMUZ	1580	1464,6	7	1513	1644,7	9	1728	1757,8	2
AĞUSTOS	1594	1599,8	9	1506	1797,5	19	1772	1890,9	7
EYLÜL	1557	1570	1	1534	1684	10	1684	1762,3	5
EKİM	1474	1588,7	8	1428	1665,1	17	1725	1756,8	1
KASIM	1469	1710	16	1534	1784	16	1667	1847	11
ARALIK	1340	1718,1	28	1549	1827,2	18	1812	1908,9	5
Yıllık Ort. Hata	%7			%16			%10		

Tablo 4.6, 2008-2010 yılları gerçek ve tahmin yük verileri ile eğitim sonucunda elde edilen mutlak hata oranlarını göstermektedir. Yapılan eğitim sonucunda bazı aylarda hata oranlarının %10'nun üzerinde olduğu görülmektedir. Bunun sebebi 2008 yılının son aylarında meydana gelen ve 2009 yılında da etkisini sürdürmeye devam eden krizin yük değerlerinde oluşturduğu olumsuz durumlardır. Tablodan da görüldüğü üzere oluşan kriz yük tüketimini önemli ölçüde azaltmıştır. Yine bu tabloya göre 2008 yılı için yıllık ortalama hata oranı %7 iken 2009 yılında bu oran %16'e çıkmış, 2010 yılında ise yük tüketiminin normal seviyelere ulaşması neticesinde bu oran %10'a düşmüştür. Yük tüketimindeki beklenmeyen ani düşüşler bazı aylarda hata oranlarını arttırdıysa bile genel itibariyle bu değerlerin kabul edilebilir olduğunu söyleyebiliriz. Bundan dolayı geleceğe yönelik yük tahmini için oluşturulan ağ modelinin kullanılmasının uygun olduğu değerlendirilmiştir.

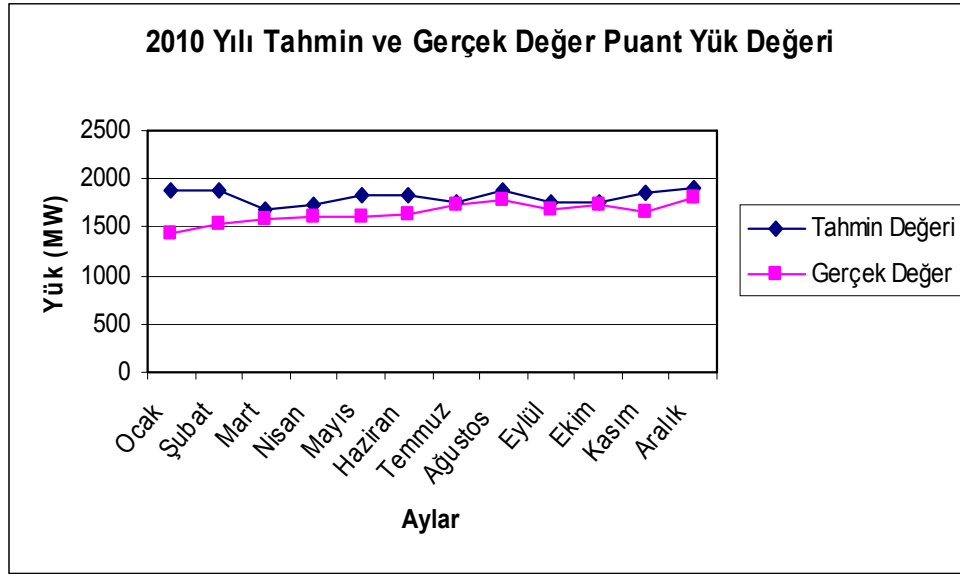
Şekil 4.7'te 2008 yılı, Şekil 4.8'da 2009 yılı ve Şekil 4.9'da 2010 yılı aylık tahmin ve gerçek puant yük değerleri değişim grafiği sunulmuştur.



Şekil 4.7: 2008 yılı tahmin ve gerçek puant yük eğrisi



Şekil 4.8: 2009 yılı tahmin ve gerçek değer puant yük eğrisi

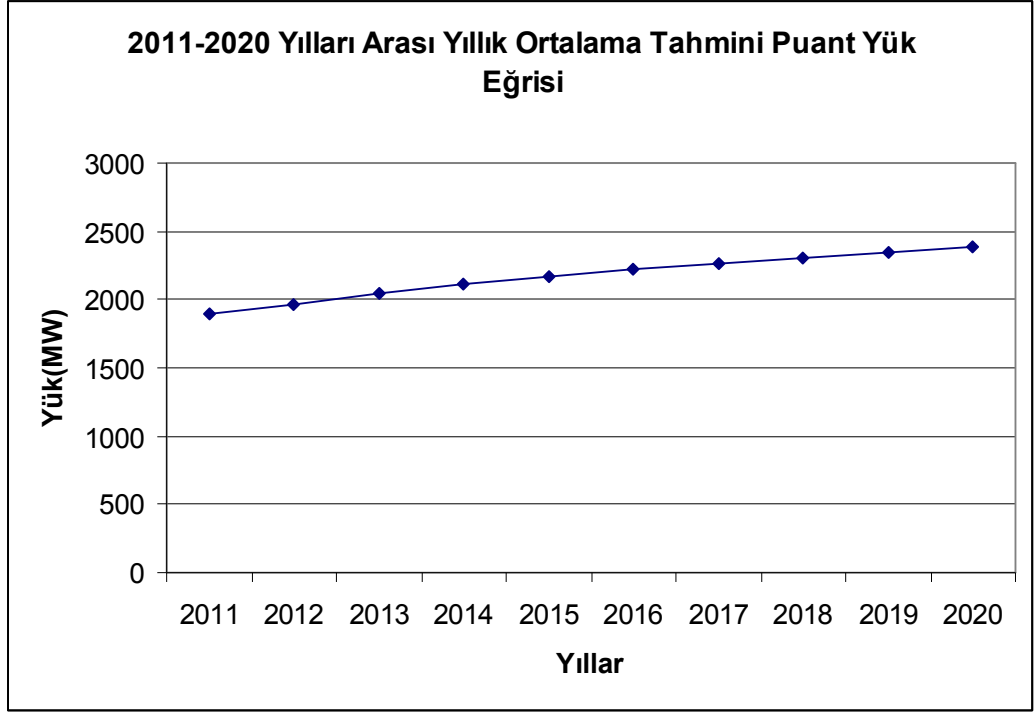


Şekil 4.9: 2010 yılı tahmin ve gerçek değer puant yük eğrisi

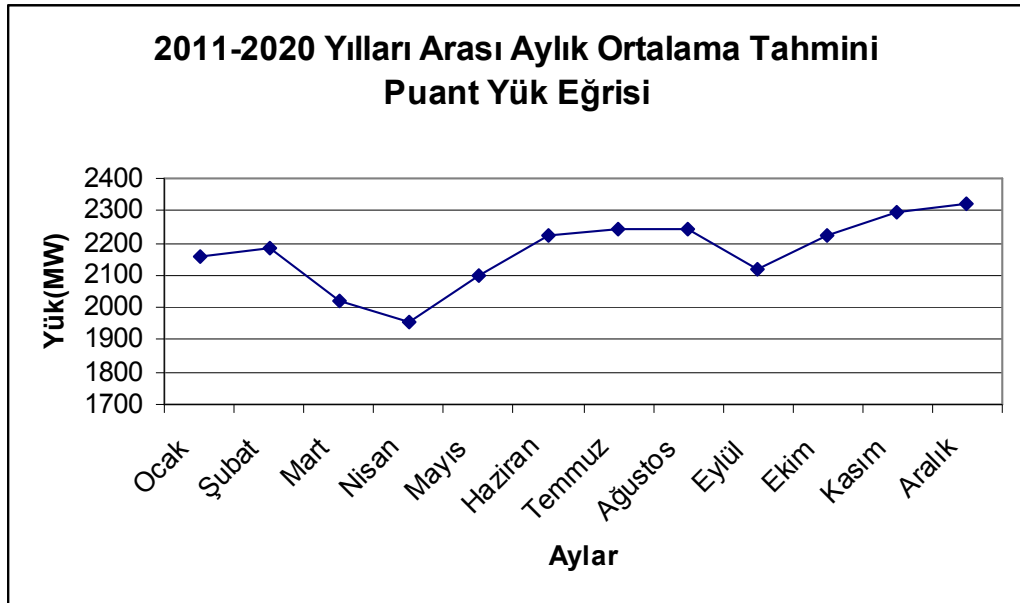
2020 yılına kadar yapılan puant yük tahmin değerleri Tablo 4.7’de gösterilmiştir.

Tablo 4.7: 2020 yılına kadar yapılan puant yük değerleri

PUANT YÜK DEĞERLERİ (MW)												
AY/ YIL	OCAK	ŞUBAT	MART	NİSAN	MAYIS	HAZİRAN	TEMMUZ	AĞUSTOS	EYLÜL	EKİM	KASIM	ARALIK
2011	1942,9	1959,5	1737,1	1741,3	1883,8	1932,1	1867,3	1990,2	1821,3	1977,1	1902,6	1968,2
2012	2001,3	2021,9	1808,2	1813,1	1939,3	2010,2	1982,7	2066,6	1880,6	2049,2	1952,6	2024,9
2013	2054,2	2078,5	1857,3	1858,3	1989,7	2081,3	2116,8	2129,1	1963,6	2108,6	1998,0	2092,2
2014	2102,6	2130,4	1902,8	1905,7	2036,0	2169,3	2193,1	2183,9	2074,7	2163,0	2039,7	2228,0
2015	2147,2	2178,2	1962,5	1946,0	2078,6	2223,2	2255,6	2233,9	2140,4	2213,2	2078,7	2352,6
2016	2188,6	2222,6	2076,7	1984,4	2118,3	2273,7	2311,8	2280,2	2186,4	2259,9	2118,1	2413,2
2017	2227,3	2264,1	2155,6	2021,4	2155,9	2321,6	2363,1	2323,4	2228,5	2303,4	2172,8	2461,8
2018	2263,5	2302,9	2196,1	2058,0	2196,2	2367,5	2410,5	2363,8	2267,7	2344,2	2266,4	2506,9
2019	2297,7	2339,5	2233,2	2094,6	2263,1	2411,1	2454,6	2401,9	2304,6	2382,6	2384,7	2549,3
2020	2329,9	2374,0	2267,6	2131,5	2337,1	2452,1	2496,0	2437,9	2339,4	2418,9	2487,6	2589,3



Şekil 4.10: 2011-2020 yılları arası yıllık ortalama tahmini puant yük eğrisi



Şekil 4.11: 2011-2020 yılları arası aylık ortalama tahmini puant yük eğrisi

BÖLÜM 5. SONUÇ VE ÖNERİLER

Bu çalışmada Kocaeli ilinin geçmiş yük, nüfus ve sıcaklık verileri kullanılarak geleceğe dönük puant yük tahmini yapılmıştır. Bu tahmin yapılırken öngörü modellemesi olarak Yapay Sinir Ağları (YSA) methodu kullanılmıştır. Çünkü YSA'lar özellikle lineer olmayan problem çözümlerinde geleneksel metodlara nazaran gösterdikleri performans ve eğitim için kullanılan sınırlı sayıdaki değerlerden hareketle genelleme yapabilmektedirler. Temel olarak, YSA detaylı bir şekilde incelenmiş ve uygun görülen bir YSA mimarisi tahmin için kullanılmıştır.

Yük tahmininde başarılı olmaları ve yaygın şekilde kullanılmaları nedeniyle bu tez araştırmasında Çok katmanlı Geri Yayılma Yapay Sinir Ağı mimarisi modellemede tercih edilmiştir. Eldeki mevcut veriler oldukça az olmasına rağmen, kullanılan geri yayılım algoritmasının hatayı tolere etme yeteneğinden dolayı hata oranları kabul edilebilir seviyelerdedir.

2008 yılının sonunda meydana gelen ve 2009 yılını da etkileyen kriz, test aşamasını olumsuz etkilemiştir. Ayrıca eğitime katılacak olan verilerdeki hataların, ekonomik verilerdeki değişkenliklerin ve ani iklim değişikliklerinin tahmini olumsuz yönde etkileyebileceği unutulmamalıdır.

Gerçeğe yakın sonuçlar elde edebilmek ve doğru bir sistem planlaması yapabilmek için öncelikle eğitim için kullanılacak verilerin uygunluğu tespit edilmelidir. Ayrıca tahmin modeline büyüme hızı ve kişi başına düşen milli gelir gibi sosyal ve ekonomik değişkenler eklenerek daha sağlıklı sonuçlar elde edilebilir. Şu unutulmamalıdır ki gerektiğinden daha düşük yük tahminine dayalı yapılan planlama çalışmaları, tüketiciye sunulan enerji talebinde yetersiz kalınmasına ve enerji kalitesinde düşmelere, gerektiğinden fazla yük tahminlerine dayalı yapılan planlama çalışmaları ise gereksiz yere yüksek kapasiteli tasarlanmış, düşük kapasiteyle çalışan ve ekonomik olmayan yatırımlara dolayısıyla mali kayıplara neden olacaktır.

KAYNAKLAR

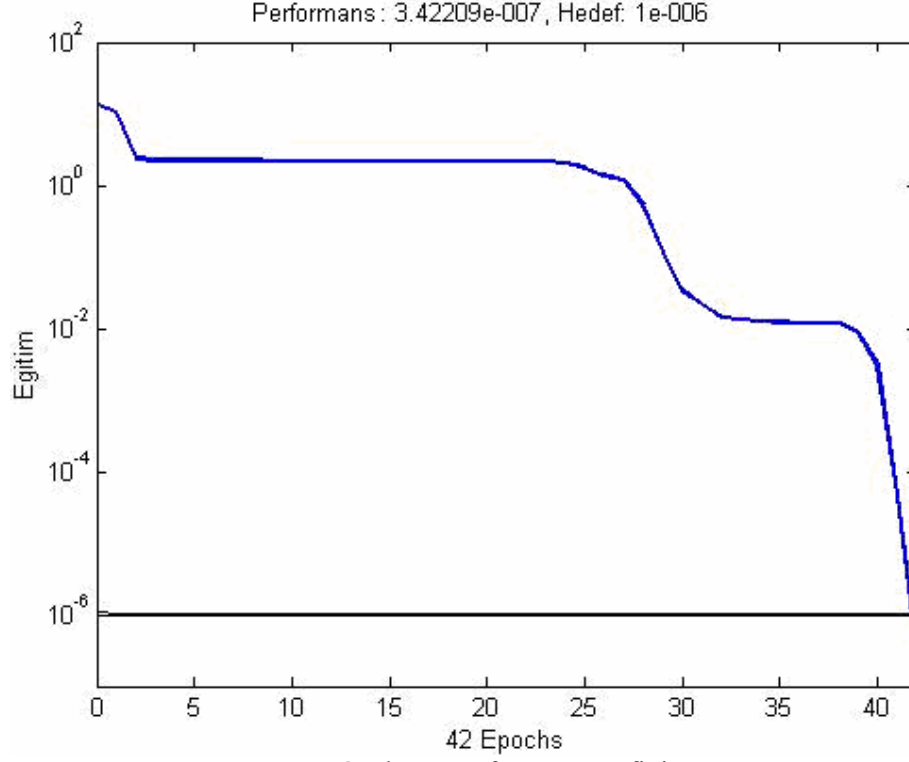
- [1] AKAR, H.A., “Dağıtım Sisteminin Özelleştirilmesi ve Yapay Sinir Ağları ile Yük Tahmini” , Yüksek Lisans Tezi, **Gazi Üniversitesi Fen Bilimleri Enstitüsü**, Ankara, (2005).
- [2] Çilliyüz, Y., “Yapay Sinir Ağları ile Çevre Koşulları Etkili Bölgesel Yük Kestirimi”, Yüksek Lisans Tezi, **Kocaeli Üniversitesi Fen Bilimleri Enstitüsü**, Kocaeli, (2006).
- [3] F. Aksel, “Regresyon Analizi ve Yapay Sinir Ağları Yöntemleri ile Uzun dönem Yük tahmini”, **I.T.Ü. Fen Bilimleri Enstitüsü**, Yüksek Lisans tezi, İstanbul, (2000).
- [4] <http://www.bilinen.net/yapaysiniraglari.htm>, **(Ziyaret.tarihi: 02 Nisan 2010)**.
- [5] <http://www.omereksi.com>, **(Ziyaret tarihi: 05 Nisan 2010)**.
- [6] Çakabey, O., Ergüner F., Patacı M., “Zaman Serileri Analizi” **Gebze Yüksek Teknoloji Enstitüsü Bilgisayar Mühendisliği Bölümü**, KOCAELİ, (2007)
- [7] Çetin Ö., “Yapay Sinir Ağları”, Bitirme Tezi, **Kocaeli Üniversitesi Elektrik Mühendisliği Bölümü**, Kocaeli, (2000).
- [8] <http://ekutup.dpt.gov.tr/ekonomi/tahmin/yurtoglh/ysa.pdf>, **(Ziyaret.tarihi : 10 Ocak 2010)**.
- [9] <http://yapay-sinir-aglari.uzerine.com>, **(Ziyaret.tarihi: 02 Ocak 2010)**.
- [10] Ertürk M., Anık M., “Yapay Sinir Ağları ile Sistem Tanıma”, Bitirme Tezi, **Kocaeli Üniversitesi Elektrik Mühendisliği Bölümü**, Kocaeli, (1998).
- [11] Hamzaçebi, C., Kutay, F., “Yapay Sinir Ağları ile Türkiye Elektrik Enerjisi Tüketiminin 2008 Yılına Kadar Tahmini”, **Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi**, 19, 227–233, (2004).
- [12] Nalbant, A., Aslan, Y., Yaşar, C., “Kütahya İlinin Yapay Sinir Ağları Kullanılarak Puant Yük Tahmini”, **Dumlupınar Üniversitesi Fen.Bil.Ens.Dergisi** 11, (2006).
- [13] http://www.emo.org.tr/ekler/a46fbfca3f1465a_ek.pdf, **(Ziyaret.tarihi : 9 Nisan 2010)**.
- [14] Park, D.C., El-Sharkawi, M.A., Marks II, R.J., Atlas, L.E., Damborg, M.J., “Electric Load Forecasting Using An Artificial Neural Network”, **IEEE Transactions on Power Systems**, 6, 442-449, (1991).

- [15] Haykin, S., *Neural Networks A Comprehensive Foundation*, Prentice Hall International, Inc. (1999).
- [16] Kermanshahi, B., Iwamiya, H., “Up to year 2020 Load Forecasting Using Neural Nets”, *Electrical Power and Energy Systems*, 24, 789–797, (2002).
- [17] Elmas Ç., *Yapay Sinir Ağları Kuram, Mimari, Eğitim, Uygulama*, sf 27-28, Seçkin Yayıncılık, Ankara, (2003).
- [18] Hsu, C.C., Chen, C.Y., “Regional Load Forecasting in Taiwan applications of Artificial Neural Network”, *Energy Conversion and Management*, 44, 1941-1949, (2003).
- [19] Singh, D., Singh, S. P., “A Self Selecting Neural Network For Short-Term Load Forecasting”, *Electric Power Components and Systems*, 29, 117-130, 2001.
- [20] Zheng, T., Girgis, A.A., Makram, E.B., “A hybrid wavelet-Kalman filter method for load forecasting”, *Electric Power Systems Research*, 54, 11–17,(2000).
- [21] Charytoniuk, W., Chen, M.S., “Very Short–Term Load Forecasting Using Artificial Neural Networks ”, *IEEE Transactions on Power Systems*, 15, 263–268, (2000).
- [22] Beccali, M., Cellura, M., Lo Brano, V., Marvuglia, A., “Forecasting Daily Urban Electric Load Profiles Using Artificial Neural Networks”, *Energy Conversion and Management*, 45, 2879–2900, (2004)..
- [23] Ghiassi, M., Zimbra, D.K., Saidane, H., “Medium Term System Load Forecasting with a Dynamic Artificial Neural Network Model”, *Electric Power Systems Research*, 76, 302–316, (2006).
- [24] Parker, D. B. “Optimal Algorithms For Adaptive Networks: Second Order Back Propagation, Second Order Direct Propagation, and Second Order Hebbian Learning”, *IEEE 1st International Conference on Neural Networks*, 2, 953–600, (1987).
- [25] Drezga, I., Rahman, S., “Short–Term Load Forecasting with Local ANN Predictors”, *IEEE Transactions on Power Systems*, 14, 844–850, (1999).
- [26] Peng, T.M., Hubele, N.F., Karady, G.G., “Advancement in the Application of Neural Networks for STLF”, *IEEE Trans. on Power Sys.*, No1, 250-257, (1992).
- [27] T., Sakihara H., Tamaki Y., Uezato K., ‘Next Day Peak Load Forecasting Using Neural Network With Adaptive Learning Algorithm Based On Similarity’, *Electric Machines and Power Systems.*, 613-624, (2000).
- [28] Baylar, A., Emiroğlu, M. E. ve Arslan, A. (1999), “Elektrik Enerjisinde Yük Tahmini Yöntemleri ve Türkiye’nin 2005–2020 Yılları Arasındaki Elektrik Enerjisi Talep Gelişimi ve Arz Planlaması”, Yüksek Lisans Tezi, *Gazi Üniversitesi Fen Bilimleri Enstitüsü*, Ankara, (2006).

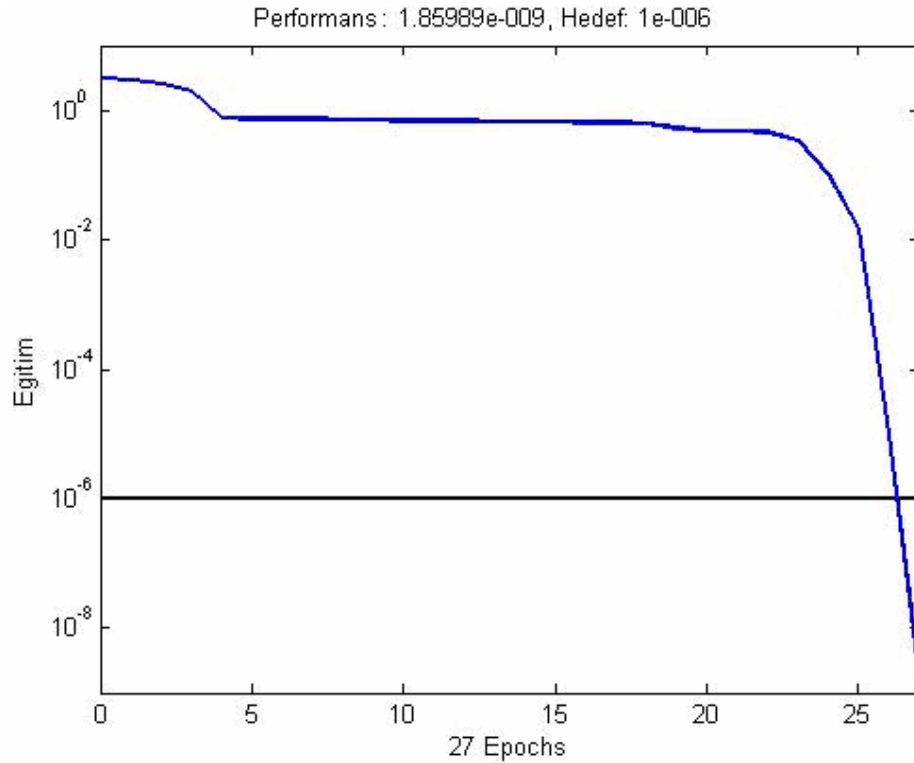
- [29] Chen, S., C.Yu., D., Moghaddamjo, A.R., “Weather Sensitive Short–Term Load Forecasting Using Nonfully Connected Artificial Neural Network”, *IEEE Transactions on Power Systems*, 7, 1098-1105, (1992).
- [30] Mohammed, O., Park, D., Merchant, R., Dinh, T., Tong, C., Azeem, A., “Practical Experiences with an Adaptive Neural Network Short–Term Load Forecasting System”, *IEEE Transactions on Power Systems*, 10, 254–265, (1995).
- [31] Cavallaro, F., “Electric Load Analysis Using an Artificial Neural Network”, *International Journal of Energy Research*, 29, 377–392, (2005).
- [32] Öztemel E., *Yapay Sinir Ağları*, Papatya Yayıncılık, İstanbul, (2003).
- [33] De Martino, B., Fusco, G., Mariani, E., Randino, R., Ricci, P., “A ;Medium and Short Term Load Forecasting Model for Electrical Industry”, *Power IndustryComputer Applications Conference*, 186–191 Cleveland–Ohio, May 15–18, (1979).

EKLER

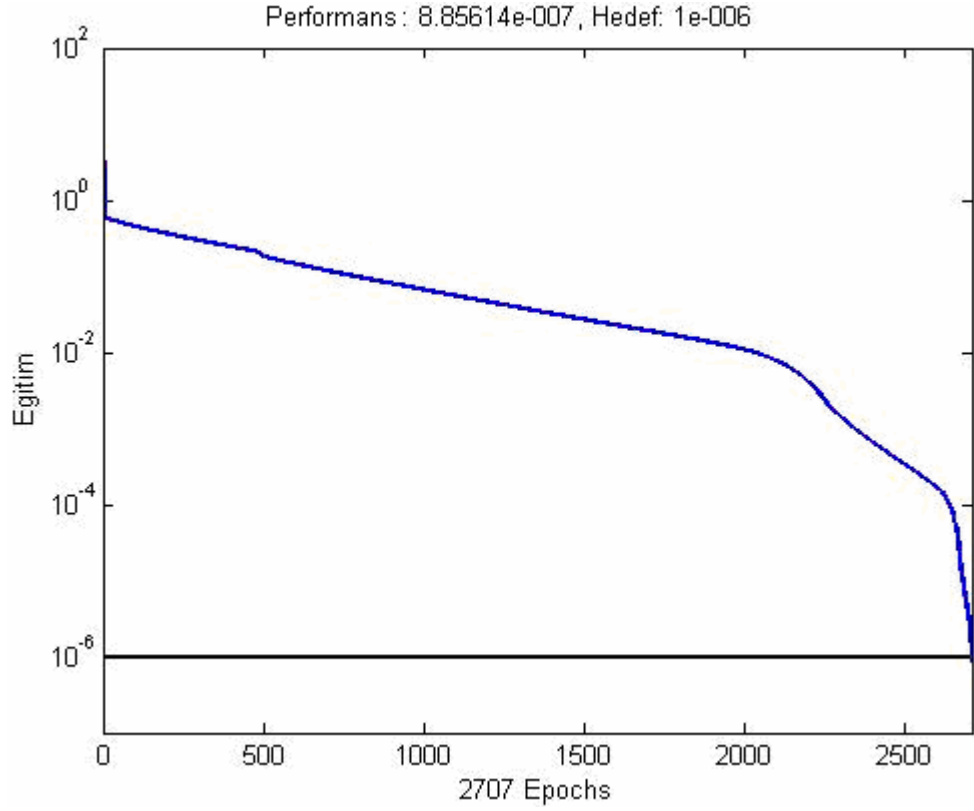
EK-A Ađın Eđitme Performansı



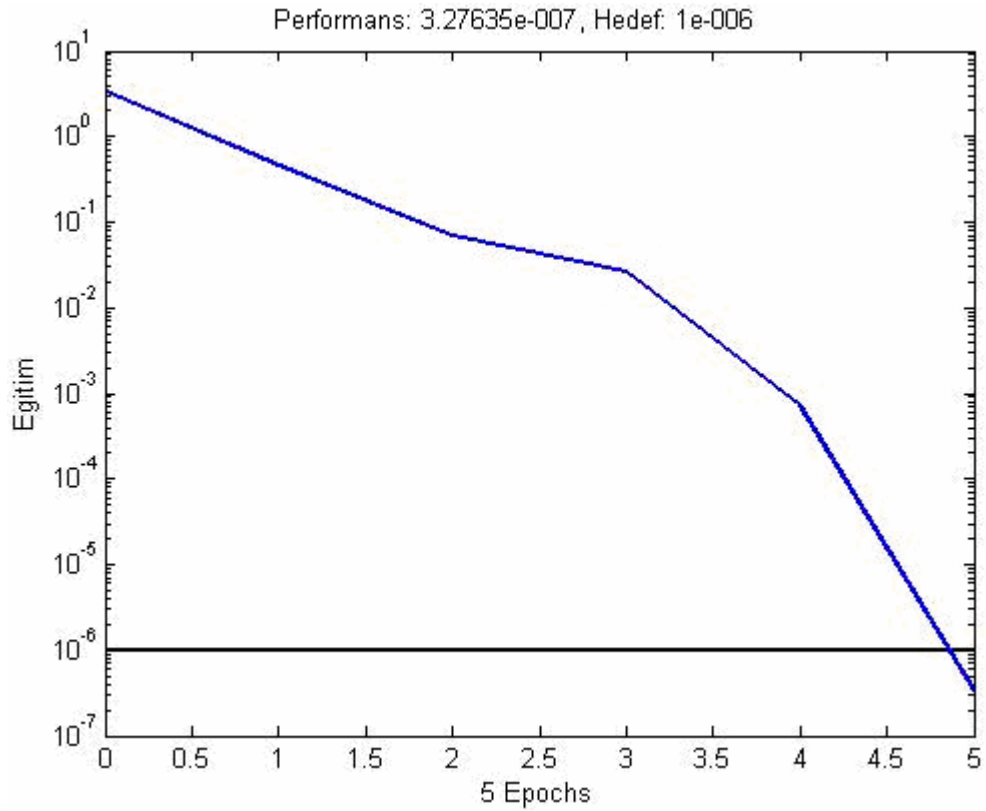
- Ocak ayı performans grafiđi



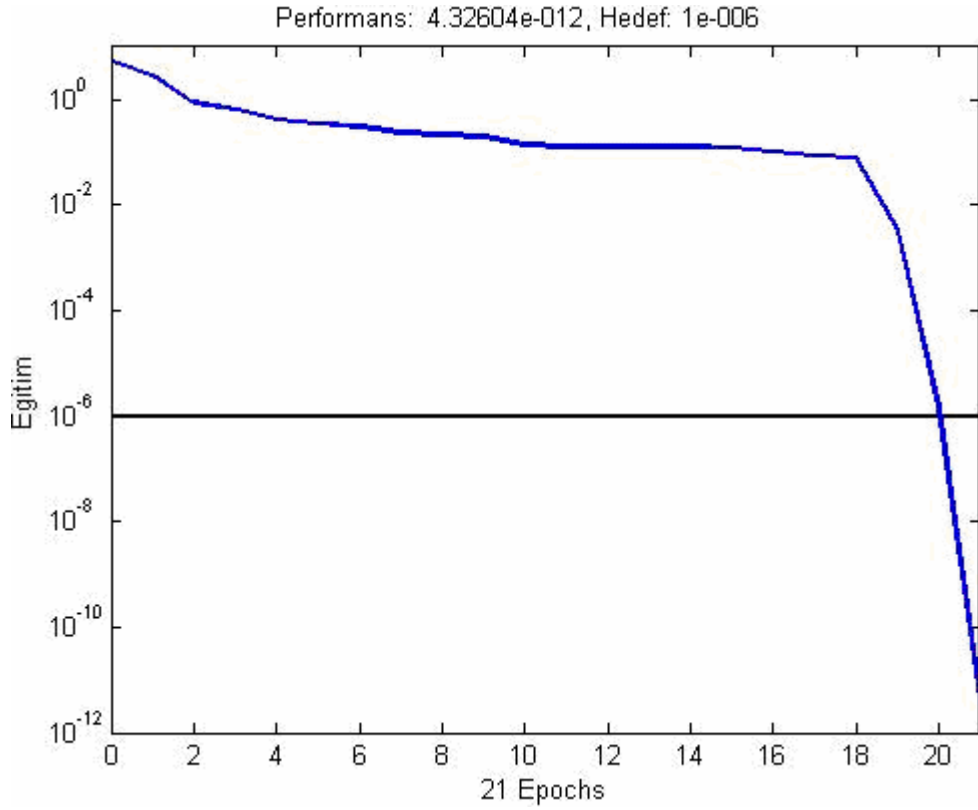
- Őubat ayı performans grafiđi



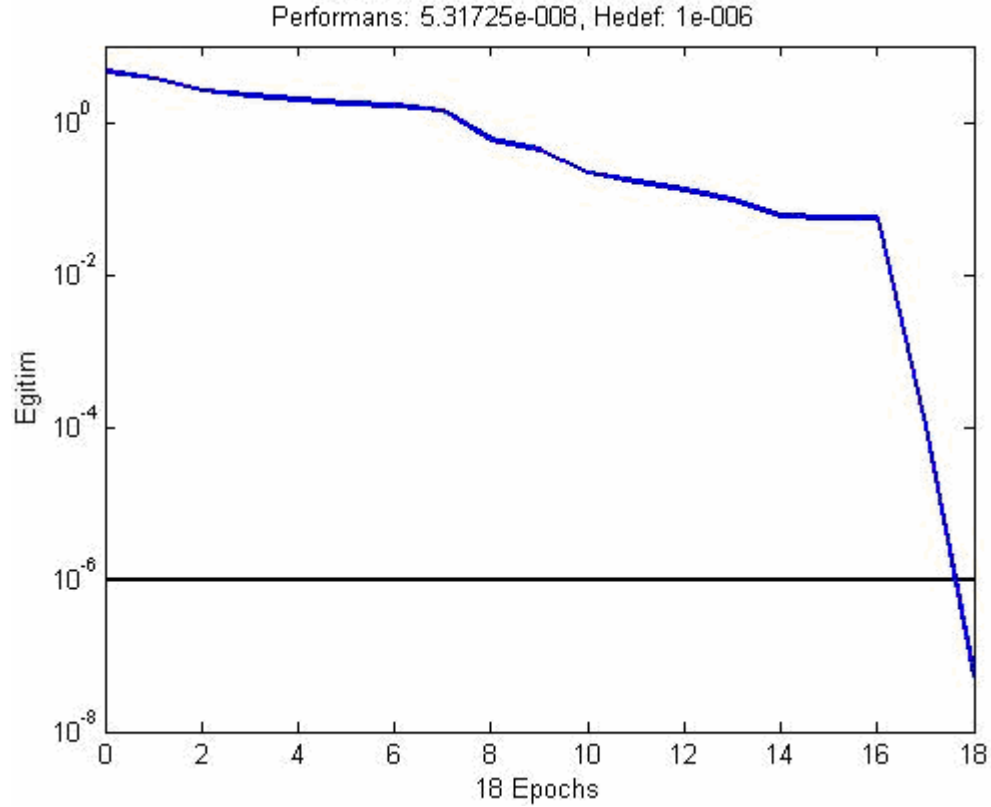
- Mart ayı performans grafiği



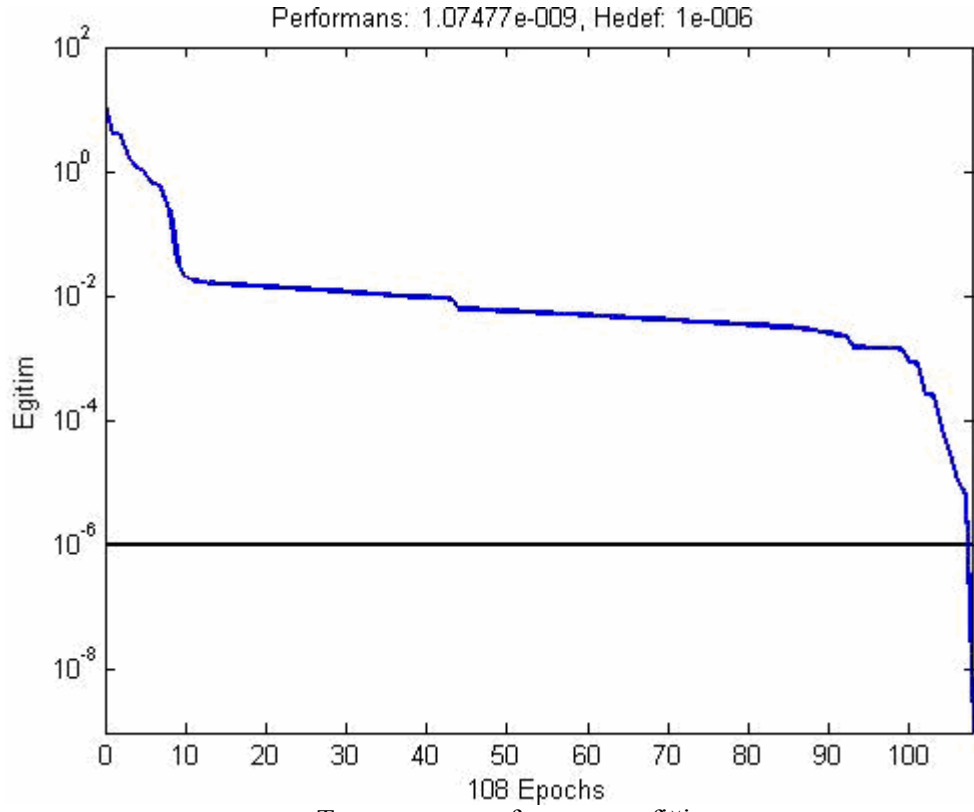
- Nisan ayı performans grafiği



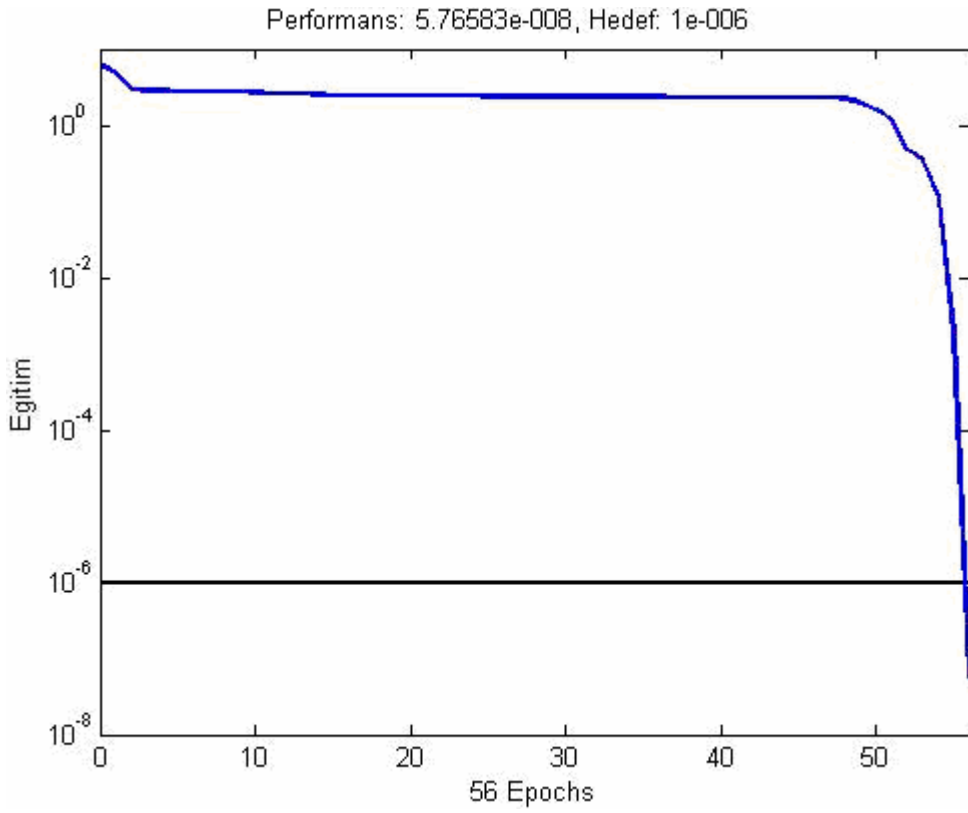
- Mayıs ayı performans grafiği



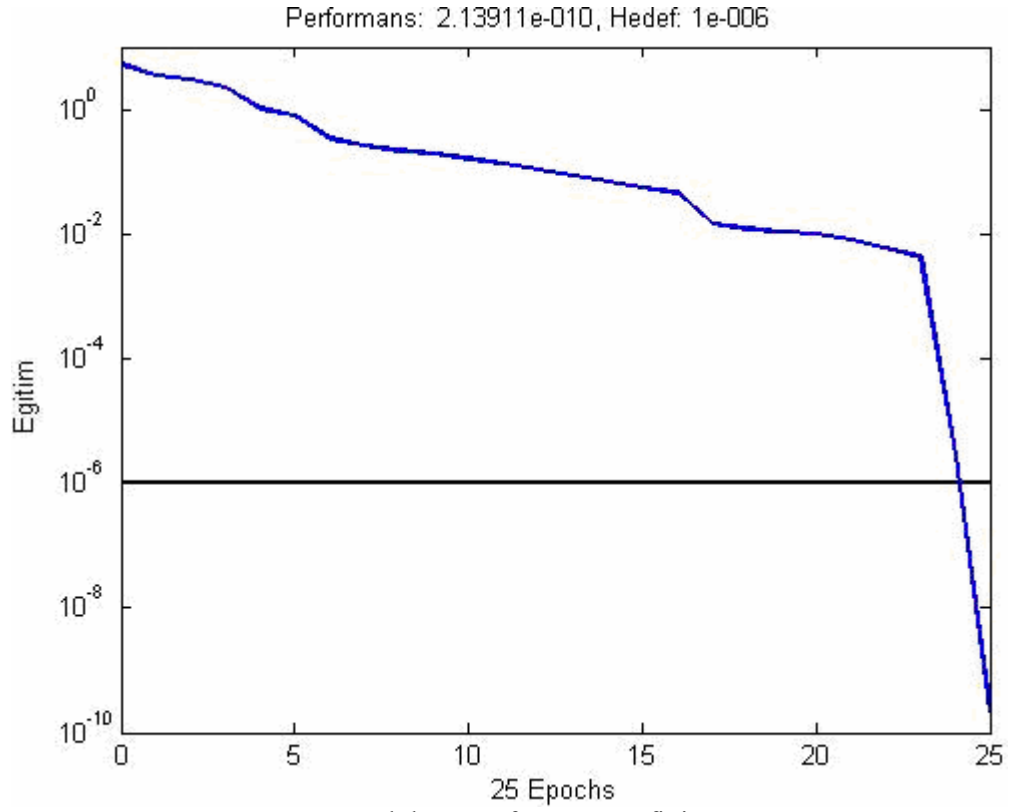
- Haziran ayı performans grafiği



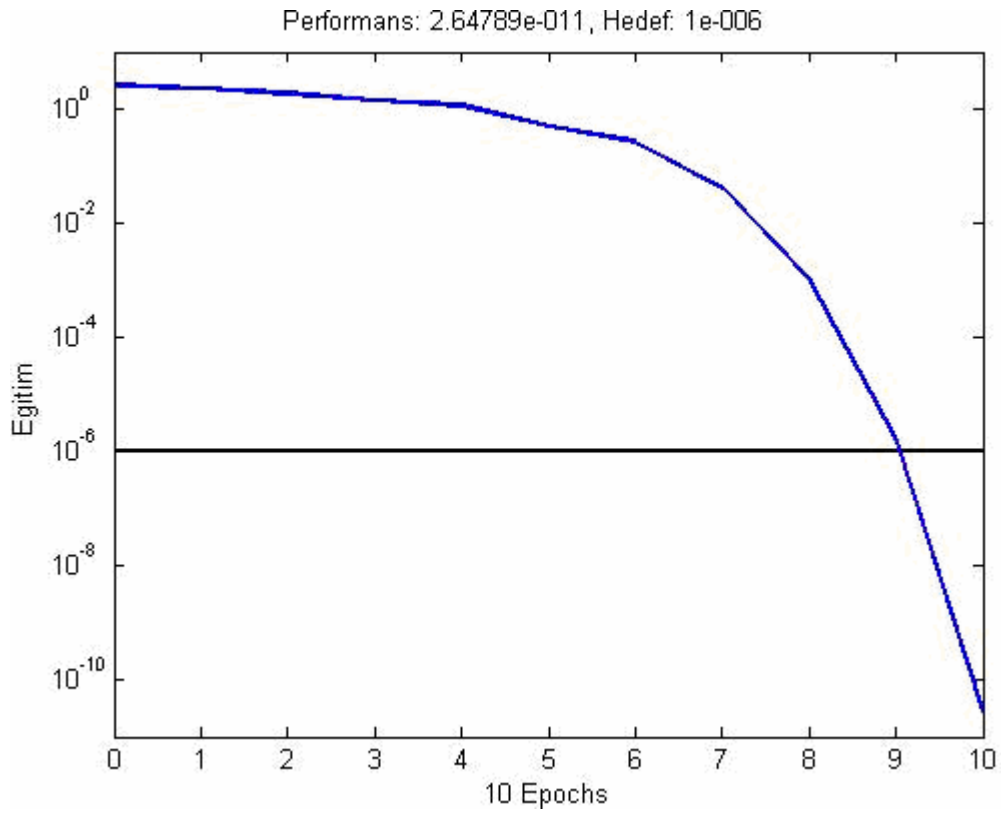
- Temmuz ayı performans grafiği



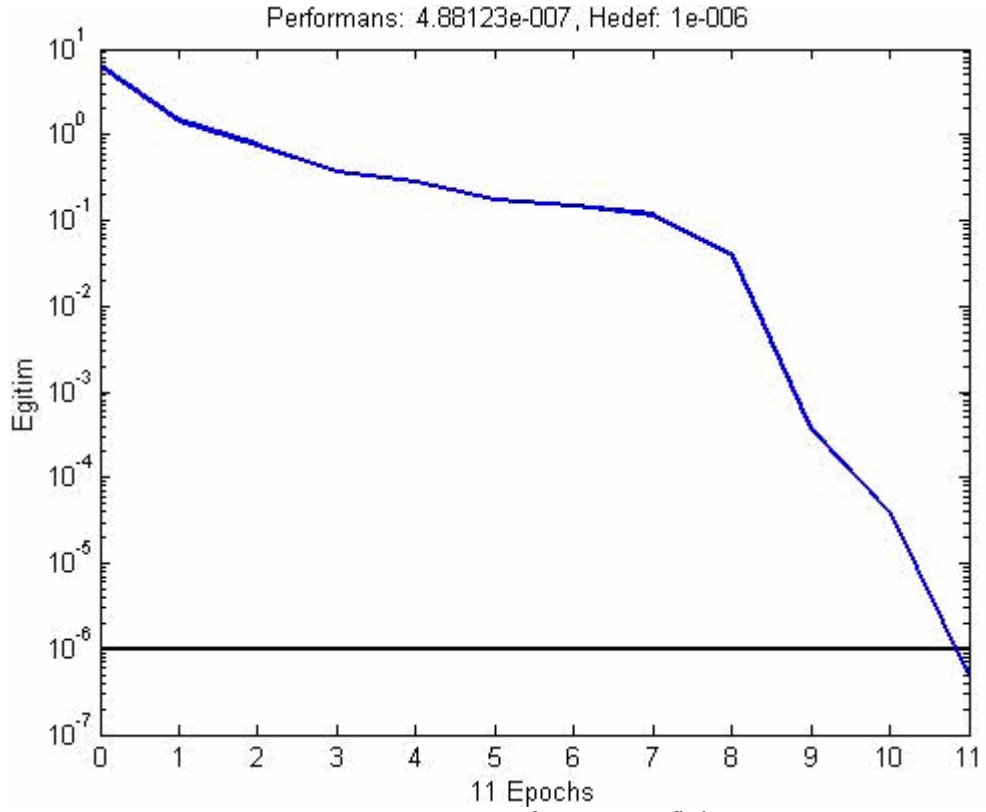
- Ağustos ayı performans grafiği



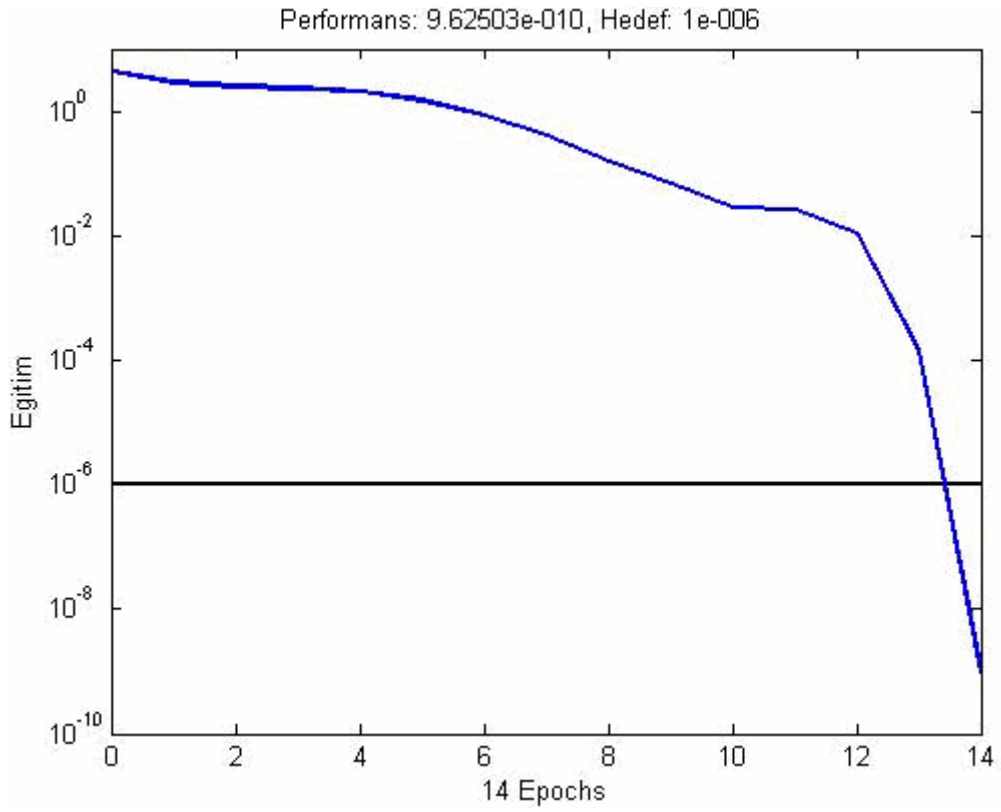
- Eylül ayı performans grafiği



- Ekim ayı performans grafiği



- Kasım ayı performans grafiği



- Aralık ayı performans grafiği

EK-B Program İşlem Kodları

```
giris=[Gecmis_Meteorolojik_Veriler;Geçmiş_Nüfus_Verileri;Gecmis_Yuk_Verileri]
hedef=[giris(end,2:end)-mean(giris(end,:)) Gecmis_Yuk_Verisi2007-
mean(giris(end,:))];
model=newff(minmax(giris),[9 1],{'logsig','purelin'});
model.trainParam.epochs=15000;
model.trainParam.min_grad=realmin;
model.trainParam.mu_max=realmax;
model.trainParam.goal=1*10^-5;
model=train(model,giris,hedef);
test=sim(model,giris)
hedef
```

```
%Ocak
ort_kare_hata_ocak=mean((test-hedef).^2)
save yuk_tahmin_ocak.mat
load yuk_tahmin_ocak.mat
ocak=giris
ocak_tahmin_2020=[];
ocak(:,1)=[];
ocak=[ocak [7.9 1404844/10000 1426/100]'];
ocak_ort=[giris(3,:)*100 ocak(end)*100];
ocak_2008=sim(model,ocak)
ocak_tahmin=ocak_2008(end)+mean(ocak_ort)
ocak_tahmin_2020=[ocak_tahmin_2020 ocak_tahmin];
...
ocak_ort=[ocak_ort ocak_tahmin];
ocak(:,1)=[];
ocak=[ocak [6.2 1976998/10000 ocak_tahmin/100]'];
ocak_2020=sim(model,ocak)*100;
ocak_tahmin=ocak_2020(end)+mean(ocak_ort)
ocak_tahmin_2020=[ocak_tahmin_2020 ocak_tahmin];
```

```
%Subat
ort_kare_hata_subat=mean((test-hedef).^2)
save yuk_tahmin_subat.mat
load yuk_tahmin_subat.mat
subat=giris
subat_tahmin_2020=[];
subat(:,1)=[];
subat=[subat [8.0 1407794/10000 1428/100]'];
subat_ort=[giris(3,:)*100 subat(end)*100];
subat_2008=sim(model,subat)*100
subat_tahmin=subat_2008(end)+mean(subat_ort)
subat_tahmin_2020=[ subat_tahmin_2020 subat_tahmin];
```

```

.....
subat_ort=[subat_ort subat_tahmin];
subat(:,1)=[];
subat=[subat [6.6 1981743/10000 subat_tahmin/100]'];
subat_2020=sim(model,subat)*100;
subat_tahmin=subat_2020(end)+mean(subat_ort)
subat_tahmin_2020=[subat_tahmin_2020 subat_tahmin]

```

```

%Mart
ort_kare_hata_mart=mean((test-hedef).^2)
save yuk_tahmin_mart.mat
load yuk_tahmin_mart.mat
mart=giris
mart_tahmin_2020=[];
mart(:,1)=[]
mart=[mart [9.6 1410751/10000 1414/100]'];
mart_ort=[giris(3,:)*100 mart(end)*100];
mart_2008=sim(model,mart)*100
mart_tahmin=mart_2008(end)+mean(mart_ort)
mart_tahmin_2020=[mart_tahmin_2020 mart_tahmin];

```

```

.....
mart_ort=[mart_ort mart_tahmin];
mart(:,1)=[];
mart=[mart [9.1 1986499/10000 mart_tahmin/100]'];
mart_2020=sim(model,mart)*100;
mart_tahmin=mart_2020(end)+mean(mart_ort)
mart_tahmin_2020=[mart_tahmin_2020 mart_tahmin]

```

```

%Nisan
ort_kare_hata_nisan=mean((test-hedef).^2)
save yuk_tahmin_nisan.mat
load yuk_tahmin_nisan.mat
nisan=giris
nisan_tahmin_2020=[];
nisan(:,1)=[]
nisan=[nisan [11.4 1413713/10000 1356/100]']
nisan_ort=[giris(3,:)*100 nisan(end)*100];
nisan_2008=sim(model,nisan)*100;
nisan_tahmin=nisan_2008(end)+mean(nisan_ort)
nisan_tahmin_2020=[nisan_tahmin_2020 nisan_tahmin]

```

```

.....
nisan_ort=[nisan_ort nisan_tahmin];
nisan(:,1)=[];
nisan=[nisan [12.5 1991267/10000 nisan_tahmin/100]'];

```



```

nisan_2020=sim(model,nisan)*100;
nisan_tahmin=nisan_2020(end)+mean(nisan_ort)
nisan_tahmin_2020=[nisan_tahmin_2020 nisan_tahmin]

```

```

%Mayis
ort_kare_hata_mayis=mean((test-hedef).^2)
save yuk_tahmin_mayis.mat
load yuk_tahmin_mayis.mat
mayis=giris
mayis_tahmin_2020=[];
mayis(:,1)=[]
mayis=[mayis [19.4 1416682/10000 1400/100]];
mayis_ort=[giris(3,:)*100 mayis(end)*100];
mayis_2008=sim(model,mayis)*100
mayis_tahmin=mayis_2008(end)+mean(mayis_ort)
mayis_tahmin_2020=[mayis_tahmin_2020 mayis_tahmin];

```

⋮

```

mayis_ort=[mayis_ort mayis_tahmin];
mayis(:,1)=[];
mayis=[mayis [17.9 1996046/10000 mayis_tahmin/100]];
mayis_2020=sim(model,mayis)*100;
mayis_tahmin=mayis_2020(end)+mean(mayis_ort)
mayis_tahmin_2020=[mayis_tahmin_2020 mayis_tahmin]

```

```

%Haziran
ort_kare_hata_haziran=mean((test-hedef).^2)
save yuk_tahmin_haziran.mat
load yuk_tahmin_haziran.mat
haziran=giris
haziran_tahmin_2020=[];
haziran(:,1)=[]
haziran=[haziran [24.3 1419657/10000 1421/100]];
haziran_ort=[giris(3,:)*100 haziran(end)*100];
haziran_2008=sim(model,haziran)*100
haziran_tahmin=haziran_2008(end)+mean(haziran_ort)
haziran_tahmin_2020=[haziran_tahmin_2020 haziran_tahmin];

```

⋮

```

haziran_ort=[haziran_ort haziran_tahmin];
haziran(:,1)=[];
haziran=[haziran [22.5 2000836/10000 haziran_tahmin/100]];
haziran_2020=sim(model,haziran)*100;
haziran_tahmin=haziran_2020(end)+mean(haziran_ort)
haziran_tahmin_2020=[haziran_tahmin_2020 haziran_tahmin]

```

```

%Temmuz
ort_kare_hata_temmuz=mean((test-hedef).^2)
save yuk_tahmin_temmuz.mat
load yuk_tahmin_temmuz.mat
temmuz=giris
temmuz_tahmin_2020=[];
temmuz(:,1)=[]
temmuz=[temmuz [25.4 1422639/10000 1442/100]];
temmuz_ort=[giris(3,:)*100 temmuz(end)*100];
temmuz_2008=sim(model,temmuz)*100
temmuz_tahmin=temmuz_2008(end)+mean(temmuz_ort)
temmuz_tahmin_2020=[temmuz_tahmin_2020 temmuz_tahmin];
...
temmuz_ort=[temmuz_ort temmuz_tahmin];
temmuz(:,1)=[];
temmuz=[temmuz [24.6 2005638/10000 temmuz_tahmin/100]];
temmuz_2020=sim(model,temmuz)*100;
temmuz_tahmin=temmuz_2020(end)+mean(temmuz_ort)
temmuz_tahmin_2020=[temmuz_tahmin_2020 temmuz_tahmin]

%Agustos
ort_kare_hata_agustos=mean((test-hedef).^2)
save yuk_tahmin_agustos.mat
load yuk_tahmin_agustos.mat
agustos=giris
agustos_tahmin_2020=[];
agustos(:,1)=[]
agustos=[agustos [25.9 1425626/10000 1481/100]];
agustos_ort=[giris(3,:)*100 agustos(end)*100];
agustos_2008=sim(model,agustos)*100
agustos_tahmin=agustos_2008(end)+mean(agustos_ort)
agustos_tahmin_2020=[agustos_tahmin_2020 agustos_tahmin];
...
agustos_ort=[agustos_ort agustos_tahmin];
agustos(:,1)=[];
agustos=[agustos [24.8 2010452/10000 agustos_tahmin/100]];
agustos_2020=sim(model,agustos)*100;
agustos_tahmin=agustos_2020(end)+mean(agustos_ort)
agustos_tahmin_2020=[agustos_tahmin_2020 agustos_tahmin]

%Eylul
ort_kare_hata_eylul=mean((test-hedef).^2)
save yuk_tahmin_eylul.mat
load yuk_tahmin_eylul.mat
eylul=giris

```

```

eylul_tahmin_2020=[];
eylul(:,1)=[]
eylul=[eylul [20.9 1428620/10000 1475/100]'];
eylul_ort=[giris(3,:)*100 eylul(end)*100];
eylul_2008=sim(model,eylul)*100
eylul_tahmin=eylul_2008(end)+mean(eylul_ort)
eylul_tahmin_2020=[eylul_tahmin_2020 eylul_tahmin];

```

⋮

```

eylul_ort=[eylul_ort eylul_tahmin];
eylul(:,1)=[];
eylul=[eylul [20.4 2015277/10000 eylul_tahmin/100]'];
eylul_2020=sim(model,eylul)*100;
eylul_tahmin=eylul_2020(end)+mean(eylul_ort)
eylul_tahmin_2020=[eylul_tahmin_2020 eylul_tahmin]

```

%Ekim

```

ort_kare_hata_ekim=mean((test-hedef).^2)
save yuk_tahmin_ekim.mat
load yuk_tahmin_ekim.mat
ekim=giris
ekim_tahmin_2020=[];
ekim(:,1)=[]
ekim=[ekim [17.0 1431906/10000 1514/100]'];
ekim_ort=[giris(3,:)*100 ekim(end)*100];
ekim_2008=sim(model,ekim)*100
ekim_tahmin=ekim_2008(end)+mean(ekim_ort)
ekim_tahmin_2020=[ekim_tahmin_2020 ekim_tahmin];

```

⋮

```

ekim_ort=[ekim_ort ekim_tahmin];
ekim(:,1)=[];
ekim=[ekim [16.7 2020114/10000 ekim_tahmin/100]'];
ekim_2020=sim(model,ekim)*100;
ekim_tahmin=ekim_2020(end)+mean(ekim_ort)
ekim_tahmin_2020=[ekim_tahmin_2020 ekim_tahmin]

```

%Kasim

```

ort_kare_hata_kasim=mean((test-hedef).^2)
save yuk_tahmin_kasim.mat
load yuk_tahmin_kasim.mat
kasim=giris
kasim_tahmin_2020=[];
kasim(:,1)=[]
kasim=[kasim [10.9 1434913/10000 1565/100]'];
kasim_ort=[giris(3,:)*100 kasim(end)*100];
kasim_2008=sim(model,kasim)*100

```

```

kasim_tahmin=kasim_2008(end)+mean(kasim_ort)
kasim_tahmin_2020=[kasim_tahmin_2020 kasim_tahmin];
.....
kasim_ort=[kasim_ort kasim_tahmin];
kasim(:,1)=[];
kasim=[kasim [11.6 2024962/10000 kasim_tahmin/100]'];
kasim_2020=sim(model,kasim)*100;
kasim_tahmin=kasim_2020(end)+mean(kasim_ort)
kasim_tahmin_2020=[kasim_tahmin_2020 kasim_tahmin]

```

```

%Aralik
ort_kare_hata_aralik=mean((test-hedef).^2)
save yuk_tahmin_aralik.mat
load yuk_tahmin_aralik.mat
aralik=giris
aralik_tahmin_2020=[];
aralik(:,1)=[]
aralik=[aralik [6.7 1437926/10000 1581/100]'];
aralik_ort=[giris(3,:)*100 aralik(end)*100];
aralik_2008=sim(model,aralik)*100
aralik_tahmin=aralik_2008(end)+mean(aralik_ort)
aralik_tahmin_2020=[aralik_tahmin_2020 aralik_tahmin];
.....
aralik_ort=[aralik_ort aralik_tahmin];
aralik(:,1)=[];
aralik=[aralik [8.3 2029822/10000 aralik_tahmin/100]'];
aralik_2020=sim(model,aralik)*100;
aralik_tahmin=aralik_2020(end)+mean(aralik_ort)
aralik_tahmin_2020=[aralik_tahmin_2020 aralik_tahmin]

```

ÖZGEÇMİŞ

1980 yılında Dıgır'da doğdu. İlk ve orta öğrenimini Bursa'da tamamladı. 1998 yılında başladığı Dumlupınar Üniversitesi Elektrik-Elektronik Mühendisliği Bölümünden 2002 yılında mezun oldu. Aynı yıl Tarım Bakanlığı Yayın Dairesi Başkanlığı'nda göreve başladı. Çalışma hayatına 2004-2008 yılları arasında MSB Sivas İnşaat Emlak Başkanlığı'nda devam etti. 2008 yılında MSB İzmit İnşaat Emlak Başkanlığı'na tayin oldu. Yine aynı yıl Kocaeli Üniversitesi Fen Bilimleri Enstitüsü Elektrik Mühendisliği Anabilim Dalında Yüksek Lisans öğrenimine başladı. Halen MSB İzmit İnşaat Emlak Bölge Başkanlığı'nda elektrik mühendisi olarak çalışmaya devam etmektedir.