



BİLECİK ŞEYH EDEBALI ÜNİVERSİTESİ

Fen Bilimleri Enstitüsü

Elektrik-Elektronik Mühendisliği Anabilim Dalı

**AKILLI SİSTEMLER KULLANILARAK GÜÇ
SİSTEMLERİNDE YÜK TAHMİNİ ANALİZİ VE
UYGULAMASI**

İdil IŞIKLI ESENER
Yüksek Lisans Tezi

Tez Danışmanı
Doç. Dr. Mehmet KURBAN

Tez İkinci Danışmanı
Yrd. Doç. Dr. Tolga YÜKSEL

BİLECİK, 2012



**BİLECİK ŞEYH EDEBALI ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ELEKTRİK-ELEKTRONİK
MÜHENDİSLİĞİ ANABİLİM DALI**

YÜKSEK LİSANS

**JÜRİ ONAY
FORMU**

Bilecik Şeyh Edebali Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nuntarih ve sayılı kararıyla oluşturulan jüri tarafındantarihinde tez savunma sınavı yapılan İdil IŞIKLI ESENER'in "Akıllı Sistemler Kullanılarak Güç Sistemlerinde Yük Tahmini Analizi ve Uygulaması" başlıklı tez çalışması Elektrik-Elektronik Mühendisliği Anabilim Dalında YÜKSEK LİSANS tezi olarak oy birliği/oy çokluğu ile kabul edilmiştir.

JÜRİ

ÜYE

(TEZ DANIŞMANI) : Doç. Dr. Mehmet KURBAN

ÜYE

(TEZ İKİNCİ DANIŞMANI) : Yrd. Doç. Dr. Tolga YÜKSEL

ÜYE : Yrd. Doç. Dr. Nazım İMAL

ÜYE : Yrd. Doç. Dr. Celal YAŞAR

ÜYE : Yrd. Doç. Dr. Ayhan GÜN

ONAY

Bilecik Şeyh Edebali Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun/...../.....tarih ve/..... sayılı kararı.

İMZA/MÜHÜR

ÖZET

Elektrik enerji sistem planlamasının ilk adımı olan yük tahmini, elektrik enerjisinin ekonomik üretim ve dağıtımının gerçekleştirilebilmesi, sistem işletme koşullarının iyileştirilebilmesi, sistem kontrolünün daha verimli sağlanabilmesi ve elektrik enerji fiyatlandırması konularında büyük önem taşımaktadır. Kısa dönem yük tahmini ekonomik işletme koşullarının sağlanmasına olanak tanımaktadır. Bu çalışmada, sıcaklık verisi kullanmadan Türkiye geneli için 24 saatlik yük tahmini amaçlanmıştır. Bu amaçla, Yapay Sinir Ağları (YSA), Dalgacık Dönüşümü (DD) ve YSA, DD ve Radyal Tabanlı Fonksiyon Sinir Ağları (RTFSA), Görgül Kip Ayrışımı (GKA) ve RTFSA olmak üzere dört yapı oluşturulmuştur. Yerel tatil günlerinin tahminindeki bozucu etkisini kaldırmak için bu günlerin verileri değiştirilerek normal gün karakteristiğine getirilmiş ve bu günlere ait tahmin sonuçları hata hesabına katılmamıştır. Daha doğru sonuçlar elde edebilmek adına regüleli yük tahmini önerilmiştir. Regülesiz ve regüleli tahmin hata yüzdeleri, ortalama günlük Ortalama Mutlak Yüzdelik Hata (MAPE), maksimum günlük MAPE olarak hesaplanmış ve dört yapı için hata karşılaştırılması yapılmıştır. Simülasyon çalışması, *MATLAB GUI* kullanılarak hazırlanan kullanıcı ara yüzü ile 2009-2010 yılları için gerçekleştirilmiştir.

Anahtar Kelimeler

Kısa dönem yük tahmini, yapay sinir ağları, radyal tabanlı fonksiyon sinir ağları, görgül kip ayrışımı

ABSTRACT

Load forecasting, the first step of power system planning, is of great importance in economic electric power generation and distribution, improvement of system operating conditions, effective system control and energy pricing. Short-term load forecasting enables the provision of economic operation conditions. In this study, Turkey's 24-hour-ahead load forecasting without temperature data is aimed. For this purpose, four structures, Artificial Neural Networks (ANN), Wavelet Transform (WT) and ANN, WT and Radial Basis Function Neural Network (RBF NN), Empirical Mode Decomposition (EMD) and RBF NN are constructed. Local holidays' load data is replaced with normal day's characteristics to remove the disturbing effects of those days on estimation, and estimation results of these days are not included in error computation. To have more accurate forecast, regulated load forecasting is proposed. Unregulated and regulated forecast error percentages are computed as average daily Mean Absolute Percentage Error (MAPE) and maximum MAPE. All MAPE values are compared between the proposed structures. Simulation is performed for years 2009-2010 via the user interface created using *MATLAB GUI*.

Key Words

Short-term load forecasting, artificial neural networks, radial basis function neural networks, wavelet transform, empirical mode decomposition

TEŞEKKÜR

Bu tezin hazırlanması sırasında akademik birikimi ve tecrübesi ile beni yönlendiren ve destekleyen değerli danışmanım Sayın Doç. Dr. Mehmet KURBAN'a, tez çalışması süresince değerli bilgilerinin ve zamanının yanı sıra manevi desteğini de benden esirgemeyen ikinci danışmanım Sayın Yrd. Doç. Dr. Tolga YÜKSEL'e, çalışmalarım süresince manevi desteği ve sabrı ile yanımda olan sevgili eşim Emre ESENER'e ve bugünlere gelmemde maddi ve manevi desteklerini benden esirgemeyen kıymetli aileme içtenlikle teşekkür ederim.

İdil IŞIKLI ESENER

İÇİNDEKİLER

	Sayfa No
TEZ ONAY SAYFASI	
ÖZET.....	iii
ABSTRACT.....	iv
TEŞEKKÜR.....	v
İÇİNDEKİLER.....	vi
ÇİZELGELER DİZİNİ.....	viii
ŞEKİLLER DİZİNİ.....	ix
KISALTMALAR DİZİNİ.....	xi
SİMGELER DİZİNİ.....	xiii
1. GİRİŞ.....	1
1.1. Genel bilgi.....	1
1.2. Literatür çalışması.....	2
1.2.1. Analitik yöntemler ile yapılan çalışmalar.....	3
1.2.2. Yapay zeka yöntemleri ile yapılan çalışmalar.....	4
1.2.3. Hibrid yöntemler ile yapılan çalışmalar.....	5
1.3. Tezin amacı.....	13
2. YÜK TAHMİNİ.....	15
2.1. Yük tahmini çeşitleri.....	15
2.2. Yük tahminini etkileyen faktörler.....	16
2.3. Yük tahmininde kullanılan yöntemler.....	17
2.3.1. Analitik yöntemler.....	18
2.3.1.1. Benzer gün yaklaşımı.....	18
2.3.1.2. En küçük kareler yöntemi.....	18
2.3.1.3. Regresyon analizi.....	19
2.3.1.4. Zaman serisi analizi.....	21
2.3.1.5. Dalgacık dönüşümü.....	22
2.3.2. Yapay zeka yöntemleri.....	23
2.3.2.1. Yapay sinir ağları.....	23
2.3.2.2. Bulanık mantık.....	23
2.3.2.3. Destek vektör makineleri.....	24
3. UYGULAMADA KULLANILAN YÖNTEMLER.....	26
3.1. Yapay sinir ağları.....	26
3.1.1. Yapay sinir ağı hücresi.....	26
3.1.1.1. Girişler.....	27
3.1.1.2. Ağırlıklar.....	27

3.1.1.3. Toplama fonksiyonu.....	28
3.1.1.4. Aktivasyon fonksiyonu.....	28
3.1.1.5. Çıkışlar.....	29
3.1.2. Yapay sinir ağlarının sınıflandırılması.....	29
3.1.3. Yapay sinir ağlarının öğrenmesi.....	31
3.1.3.1. Danışmanlı öğrenme.....	31
3.1.3.2. Danışmansız öğrenme.....	31
3.1.3.3. Takviyeli öğrenme.....	31
3.2. RTF sinir ağları.....	32
3.3. Dalgacık dönüşümü.....	33
3.3.1. Sürekli dalgacık dönüşümü.....	37
3.3.2. Ayrık dalgacık dönüşümü.....	37
3.4. Görgül kip ayrışımı.....	38
4. VERİ ANALİZİ.....	40
5. UYGULAMA.....	44
5.1. Veri ön işleme.....	44
5.2. Veri kümesi seçimi.....	45
5.3. Uygulama için oluşturulan yapılar	45
5.3.1. Yapı 1: Yapay sinir ağları.....	47
5.3.2. Yapı 2: Dalgacık dönüşümü ve yapay sinir ağları.....	48
5.3.3. Yapı 3: Dalgacık dönüşümü ve RTF sinir ağları.....	49
5.3.4. Yapı 4: Görgül kip ayrışımı ve RTF sinir ağları.....	49
5.4. Regülasyon hesabı.....	50
5.5. Hata hesabı.....	51
6. YZ-KDYT ARAYÜZÜ.....	52
7. UYGULAMA SONUÇLARI.....	55
8. SONUÇLAR VE ÖNERİLER.....	67
KAYNAKLAR.....	69
EKLER.....	74
Ek-1 YZ-KDYT arayüzü kodları.....	74
ÖZGEÇMİŞ.....	88

ÇİZELGELER DİZİNİ

	Sayfa No
Çizelge 1.1: Analitik yöntemler ile yapılan çalışmalar.....	9
Çizelge 1.2: Yapay zeka yöntemleri ile yapılan çalışmalar.....	10
Çizelge 1.3: Hibrid yöntemler ile yapılan çalışmalar.....	11
Çizelge 1.4: Tez çalışmasının literatürdeki yeri.....	13
Çizelge 3.1: Biyolojik sinir sistemi ile YSA'nın benzeşimi.....	27
Çizelge 3.2: YSA'da kullanılan eğitim fonksiyonları.....	32
Çizelge 6.1: Arayüzde kullanılan objeler, karşılıkları ve kullanım amaçları.....	53
Çizelge 7.1: Örnek günlerin günlük ortalama MAPE değerleri.....	56
Çizelge 7.2: 12 Mayıs 2009 gününün tahmini için kullanılan veri kümesi.....	57
Çizelge 7.3: 12 Mayıs 2009 günü saatlik tahmin sonuçları.....	59
Çizelge 7.4: 2009 yılı MAPE değerleri.....	60
Çizelge 7.5: 2009 yılı aylık MAPE değerleri.....	61
Çizelge 7.6: 2010 yılı MAPE değerleri.....	61
Çizelge 7.7: 2010 yılı aylık MAPE değerleri.....	62

ŞEKİLLER DİZİNİ

	Sayfa No
Şekil 1.1: Elektrik enerji sistem planlaması organizasyon şeması.....	1
Şekil 3.1: Yapay sinir ağı hücresi modeli.....	27
Şekil 3.2: Yaygın olarak kullanılan aktivasyon fonksiyonları.....	28
Şekil 3.3: İleri beslemeli ağ yapısı.....	29
Şekil 3.4: 3 katmanlı YSA yapısı.....	30
Şekil 3.5: Geri beslemeli ağ yapısı.....	30
Şekil 3.6: RTF sinir ağları yapısı.....	33
Şekil 3.7: Fourier dönüşümü.....	34
Şekil 3.8: Kısa zamanlı Fourier dönüşümü.....	34
Şekil 3.9: Dalgacık dönüşümü.....	35
Şekil 3.10: Bir işaretin Fourier dönüşümü.....	35
Şekil 3.11: Bir işaretin dalgacık dönüşümü.....	36
Şekil 3.12: Daubechies dalgacık ailesi.....	36
Şekil 3.13: Üçüncü seviye ayrık dalgacık dönüşümü.....	38
Şekil 3.14: Yük eğrisi için GKA örneklenmesi.....	39
Şekil 4.1: Haftanın günlerine göre yük tüketimi (5-11 Ocak 2009).....	40
Şekil 4.2: Haziran ayı tüm haftaları için günlük puant yük tüketimi değerleri.....	41
Şekil 4.3: 2009 yılı aylık yük tüketim değerleri.....	42
Şekil 4.4: 2010 yılı aylık yük tüketim değerleri.....	42
Şekil 4.5: Yıllık yük tüketim değerleri.....	43
Şekil 5.1: KDYT blok diyagramı.....	46
Şekil 5.2: Sinir ağları eğitim şeması.....	46
Şekil 5.3: YSA yapısı blok diyagramı.....	47
Şekil 5.4: Dalgacık dönüşümü ve YSA yapısı blok diyagramı.....	48

Şekil 5.5:	Dalgacık dönüşümü ve RTF sinir ağları yapısı blok diyagramı.....	49
Şekil 5.6:	Görgül kip ayrışımı ve RTF sinir ağları yapısı blok diyagramı.....	50
Şekil 6.1:	YZ-KDYT arayüzü.....	52
Şekil 7.1:	12 Mayıs 2009 günü regülesiz tahmin eğrileri ve gerçek yük eğrisi.....	56
Şekil 7.2:	12 Mayıs 2009 günü regüleli tahmin eğrileri ve gerçek yük eğrisi.....	56
Şekil 7.3:	YZ-KDYT arayüzü, YSA yapısı tahmin sonuçları (12.05.2009-Salı).....	57
Şekil 7.4:	YZ-KDYT arayüzü, dalgacık dönüşümü ve YSA yapısı tahmin sonuçları (12.05.2009-Salı).....	58
Şekil 7.5:	YZ-KDYT arayüzü, dalgacık dönüşümü ve RTF sinir ağları yapısı tahmin sonuçları (12.05.2009-Salı).....	58
Şekil 7.6:	YZ-KDYT arayüzü, GKA ve RTF sinir ağları yapısı tahmin sonuçları (12.05.2009-Salı).....	58
Şekil 7.7:	YSA yapısı, 2009-2010 yılları regülesiz tahmin MAPE değerleri.....	63
Şekil 7.8:	YSA yapısı, 2009-2010 yılları regüleli tahmin MAPE değerleri.....	63
Şekil 7.9:	Dalgacık Dönüşümü ve YSA yapısı, 2009-2010 yılları regülesiz tahmin MAPE değerleri.....	64
Şekil 7.10:	Dalgacık Dönüşümü ve YSA yapısı, 2009-2010 yılları regüleli tahmin MAPE değerleri.....	64
Şekil 7.11:	Dalgacık Dönüşümü ve RTF sinir ağları yapısı, 2009-2010 yılları regülesiz tahmin MAPE değerleri.....	65
Şekil 7.12:	Dalgacık Dönüşümü ve RTF sinir ağları yapısı, 2009-2010 yılları regüleli tahmin MAPE değerleri.....	65
Şekil 7.13:	GKA ve RTF sinir ağları yapısı, 2009-2010 yılları regülesiz tahmin MAPE değerleri.....	66
Şekil 7.14:	GKA ve RTF sinir ağları yapısı, 2009-2010 yılları regüleli tahmin MAPE değerleri.....	66

KISALTMALAR DİZİNİ

ADD	: Ayrık Dalgacık Dönüşümü (Discrete Wavelet Transform)
AR	: Otoregresif (Auto Regressive)
ARIMA	: Birleştirilmiş Otoregresif-Hareketli Ortalama (Auto Regressive Integrated Moving Average)
ARMA	: Otoregresif-Hareketli Ortalama (Auto Regressive-Moving Average)
BHHT	: Tahmin Gününün Bir Hafta Öncesi Haftalık Tüketimi
db2, db3, db4	: Daubechies 2, Daubechies 3, Daubechies 4 Dalgacıkları
DD	: Dalgacık Dönüşümü
EKKY	: En Küçük Kareler Yöntemi
GA	: Genetik Algoritma
GKA	: Görgül Kip Ayrışımı (Empirical Mode Decomposition)
GUI	: Grafiksel Kullanıcı Ara Yüzü (Graphical User Interface)
HT	: Haftalık Yük Tüketimi
IHHT	: Tahmin Gününün İki Hafta Öncesi Haftalık Tüketimi
KDYT	: Kısa Dönem Yük Tahmini
KKO	: Karınca Kolonisi Optimizasyonu (Ant Colony Optimization)
MA	: Hareketli Ortalama (Moving Average)
MAPE	: Ortalama Mutlak Yüzdelik Hata (Mean Absolute Percentage Error)
MLP	: Çok Katmanlı Perseptron (Multi Layer Perceptron)
ODYT	: Orta Dönem Yük Tahmini
ÖKF	: Öz Kip Fonksiyonları (Intrinsic Mode Functions)
PSO	: Parçacık Sürü Optimizasyonu (Particle Swarm Optimization)
RTF	: Radyal Tabanlı Fonksiyon (Radial Basis Function)

RTFSA	: Radyal Tabanlı Fonksiyon Sinir Ağları (Radial Basis Function)
RYT	: Regüleli Yük Tahmini
RZYT	: Regülesiz Yük Tahmini
SARIMA	: Mevsimsel Birleştirilmiş Otoregresif-Hareketli Ortalama (Seasonal Auto Regressive Integrated Moving Average)
SDD	: Sürekli Dalgacık Dönüşümü (Continuous Wavelet Transform)
ST	: Saatlik Yük Tüketimi
SVM	: Destek Vektör Makineleri (Support Vector Machines)
TEİAŞ	: Türkiye Elektrik İletim Anonim Şirketi
UDYT	: Uzun Dönem Yük Tahmini
YSA	: Yapay Sinir Ağları
YZ-KDYT	: Kısa Dönem Yük Tahmini için Yapay Zeka Modelleri
Δ HT	: Haftalık Yük Tüketimi Değişim Oranı

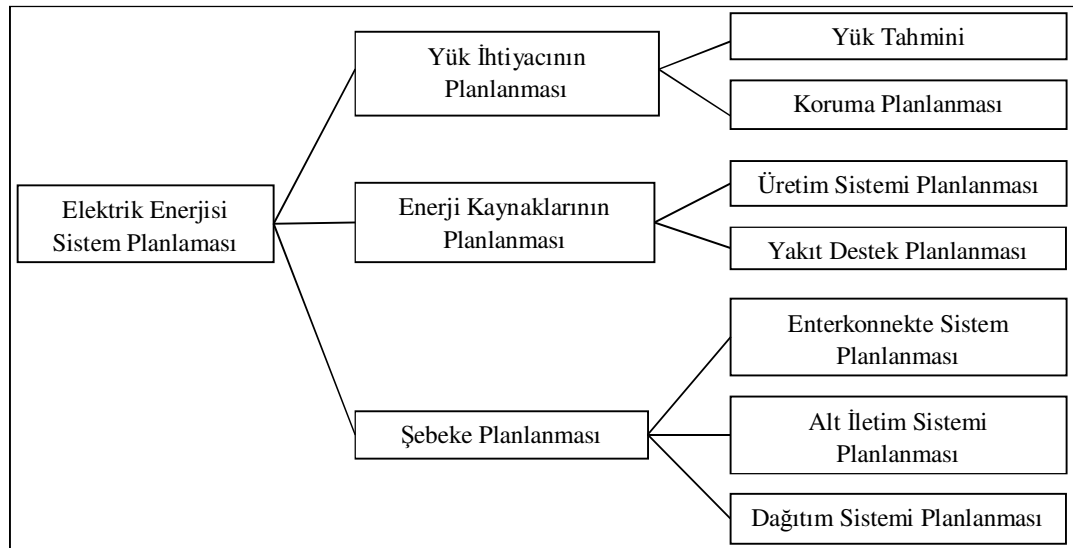
SİMGELER DİZİNİ

g	: Gün Endeksi
G	: Yerel tatil günleri dışında kalan günlerin sayısı
s	: Saat Endeksi

1. GİRİŞ

1.1. Genel Bilgi

Herhangi bir hareketi yapan ya da yapmaya hazır olan kabiliyet olarak tanımlanan enerji, varoluşumuzdan beri çevremizdeki yaşam döngüsünün merkezinde yer alan en önemli kavramlardan biri olmuştur. İlk çağlarda ısınma ve beslenme gibi amaçlar için enerjiye ihtiyaç duyulmuştur. Sanayi devrimi ile fabrikalaşma, günden güne değişen teknoloji ve kentsel gelişme sonucu enerjiye duyulan ihtiyaç artmaktadır. Elektrik enerjisi, üretim tesislerinden iletim hatları ile taşınıp dağıtım hatları ile tüketiciye ulaştırılan, günlük hayatın her anında ihtiyaç duyulan bir enerjidir. Enerji kaynaklarının kısıtlı olması, bu kaynakların bir gün biteceği gerçeği ve enerjinin depolanamaz olması talep dâhilinde elektrik enerjisi üretilmesi gerektiği gerçeğini ortaya koymaktadır. Artan elektrik enerjisi talebine karşı tüketiciye kaliteli, güvenilir ve ekonomik enerji sunmaktan taviz verilmemelidir. Bu nedenle, elektrik enerji sistemi planlaması enerji kapsamında üzerinde durulması gereken en önemli nokta haline gelmiştir. Elektrik enerji sistem planlaması organizasyon şeması Şekil 1.1’de gösterilmiştir (Yoldaş, 2006).



Şekil 1.1. Elektrik enerji sistem planlaması organizasyon şeması (Yoldaş, 2006).

Elektrik enerji sistemi, üretim, iletim ve dağıtım alt sistemlerinden oluşmaktadır. Sistem planlanması, bu üç alt sistemin planlanması anlamına gelmektedir. Elektrik enerji sistemi planlaması yük tahmini ile başlar. Yük tahmini, geçmişteki ve şimdiki koşulların incelenerek değişimlerin karakterlerinin çıkarılarak, gelecekteki durumun varsayımına dayanır. Enerjinin güvenilirliği açısından koruma planlaması yapılması gerekmektedir. Enerji sürekliliğini sağlamak amacı ile santrallere ekleme yapma gerekliliği, yeni enerji santrallerinin kurulması gerekliliği ve bunlara bağlı yatırım maliyetleri belirlenir. Üretim planlaması ile gelecekte talep edilecek elektrik enerjisinin ekonomik, kaliteli ve güvenilir şekilde sağlanması amaçlanır. Yakıt destek planlamasında, farklı yakıt türlerinin geçerlilikleri ve fiyatları tahmin edilir. Uzun dönemde yakıt sağlanmasına ilişkin olarak yapılan girişimler ve bunların sonuçları topluca değerlendirilir. Üretim sistemi planlamasının verilerine göre iletim sistemi planlaması yapılır. İletim sistemi planlamasının amacı, gelecekteki yük ve üretim senaryolarına, iletim hatlarının ekonomik olarak geçebileceği yerlere, iletim sistemi elemanlarının kapasitelerine ve güvenilirlik kistasına göre iletim sisteminin gelişimini belirlemektir. Enterkonnekte sistemlerde, güç alışverişinin gerçekleştirildiği sistemlerin her birinde ayrı ayrı ve birlikte iletim devresi eklemeleri zorunlu olduğundan, bu sistemlere ilişkin alınacak kararlar iletim sistemi planlamasını etkilemektedir. Alt iletim sistemi planlaması, ana dağıtım istasyonlarını dağıtım trafolarına bağlayan, gerilimi ana iletim sistemine oranla daha düşük kademede olan iletim elemanlarının ve dağıtım trafolarının yüksek gerilim tarafındaki planlama etkinliklerini kapsar (Yoldaş, 2006). Enterkonnekte sistem planlaması ve alt iletim sistem planlamasının verileri dikkate alınarak, enterkonnekte sistemden dağıtım trafoları ile alınan elektrik enerjisinin, tüketiciye ekonomik, yeterli ve nitelikli şekilde ulaştırılması dağıtım sistemi planlamasının amacıdır. Bu amaç doğrultusunda, dağıtım trafolarının yerleşimleri, güçleri, işletme karakteristikleri, gerilim kademeleri, besleme alanlarının büyüklükleri, yoğunlukları ve özellikleri dikkate alınmaktadır.

1.2. Literatür Çalışması

Elektrik yük tahmini uzun yıllardır üzerine çalışılan bir problemdir. Çeşitli bilim insanlarının konu üzerine geliştirdikleri değişik yöntemler ile geniş bir literatüre sahip

olmuştur. Kısa Dönem Yük Tahmini (KDYT) çalışmaları, genellikle yapay zeka yöntemleri kullanılarak gerçekleştirilmiştir. Analitik yöntemler ile yapılan KDYT çalışmaları da yapılmış ancak analitik yöntemler daha çok yapay zekâ yöntemleri ile birlikte kullanılmıştır.

1.2.1 Analitik yöntemler ile yapılan çalışmalar

Mu ve arkadaşları, Çin-Hainan bölgesi için benzer gün yöntemi ile 2008 yılına dair 7 günlük yük tahmini yapmışlardır (Mu vd., 2010). Çalışmada, benzer günler hava verisi, sıcaklık verisi, gün tipi, hafta tipi ve gün farkı temel alınarak belirlenmiştir. Hava verisi kapsamında havanın güneşli ya da bulutlu olması, sıcaklık verisi kapsamında günün ise maksimum ve minimum sıcaklıkları ele alınmıştır. Gün tipi değişkeni, tahmin edilecek günün normal gün ya da tatil günü olduğunu, hafta tipi değişkeni de haftanın hangi gününe denk geldiğini anlatmaktadır. Gün farkı değişkeni ise tahmin günü '0' kabul edilip diğer günlerin bu güne uzaklığı alınarak belirlenmiştir.

Nalbant ve arkadaşları, Türkiye-Kütahya bölgesi için bağımsız değişken kullanmadan 2000-2004 yılları aylık puant yük verileri giriş alınarak en küçük kareler yöntemi ile doğrusal, üstel ve kuadratik yaklaşım kullanılarak 2005-2009 yılları için aylık yük tahmini yapmışlar ve üç yaklaşımın tahmin sonuçlarını karşılaştırmışlardır (Nalbant vd., 2005). Elde edilen sonuçlardan yük değişiminin daha çok doğrusal bir karakter sergilediği görüldüğünden, doğrusal yaklaşımın en iyi sonuçları verdiği kanaatine varılmıştır.

Sümer ve arkadaşları, bağımsız değişken kullanmadan Birleştirilmiş Otoregresif-Hareketli Ortalama (Auto Regressive-Integrated Moving Average-ARIMA), Mevsimsel Birleştirilmiş Otoregresif-Hareketli Ortalama (Seasonal Auto Regressive Integrated Moving Average-SARIMA) ve regresyon yöntemleri ile aylık yük tahmininde bulunmuşlardır (Sümer vd., 2009). Çalışmada, 1997-2005 yılları aylık puant yük verileri kullanılarak 2006-2010 yılları aylık tahmin sonuçları elde edilmiştir.

Taylor ve McSharry, ARIMA modelleme, Otoregresif (Auto Regressive-AR) modelleme, temel bileşen analizi ve Holt-Winters üstel yumuşatma yöntemleri ile 24 saatlik yük tahmininde bulunmuşlardır (Taylor ve McSharry, 2008). Çalışmanın veri kümesi 3 Nisan 2005- 29 Ekim 2005 arası yarım saatlik ve saatlik yük verilerinden oluşmaktadır. Yapılar ilk 20 haftanın verileri ile eğitilmiş ve son 10 haftanın verileri tahmin sonuçları ile karşılaştırılmıştır. Veri kümesindeki özel günlerin verileri bir hafta öncesi ve bir hafta sonrası verilerinin ortalaması ile değiştirilmiştir.

1.2.2. Yapay zeka yöntemleri ile yapılan çalışmalar

Azadeh ve arkadaşları, Yapay Sinir Ağları (YSA) ile İran geneli 2003-2005 yılları için 24 saatlik yük tahmini yapmışlardır (Azadeh vd., 2009). Çalışmada mevsimsel YSA ve saatlik mevsimsel YSA olmak üzere iki yapı oluşturulmuştur. İki yapıda da giriş olarak gün tipi, tahmin gününün bir gün öncesi yük verisi, ortalama-maksimum-minimum gerçek sıcaklık verileri, ortalama-maksimum-minimum sıcaklık tahminleri kullanılmış ve her mevsim için farklı sinir ağları oluşturulmuştur. Mevsimsel YSA yapısı ile 24 saatlik, saatlik mevsimsel YSA yapısı ile saatlik tahmin sonuçları elde edilmiştir. Mevsimsel YSA yapısının daha düşük hata yüzdeleri verdiği görülmüştür.

Dongxiao ve Jie, Radyal Tabanlı Fonksiyon (RTF) sinir ağları ile 24 saatlik yük tahmini yapmışlardır (Dongxiao ve Jie, 2011). Çalışmada, giriş verileri, tahmin gününe ait maksimum ve minimum sıcaklık, yağış verileri, tahmin gününün öncesindeki son üç günün saatlik yük verileri olarak seçilmiştir. Sinir ağları eğitimi 16 Eylül 2009 – 16 Kasım 2009 arası yük ve hava verileri ile eğitilmiş, 17 Kasım 2009 gününe dair 24 saatlik tahmini yük talepleri elde edilmiştir.

Gontar ve Hatzigryiou, Crete-İtalya için RTF sinir ağları kullanarak 1999 yılı için 48 saatlik yük tahmininde bulunmuşlardır (Gontar ve Hatzigryiou, 2001). Çalışmada bağımsız değişken kullanılmamış ve tatil günleri için ayrı bir model oluşturulmamıştır.

Jain ve Satish, Destek Vektör Makineleri (SVM) ile yaptıkları 24 saatlik yük tahmini çalışmasında bağımsız değişken olarak sıcaklık verisi ve gün tipi değişkenlerini kullanmışlardır (Jain ve Satish, 2008). Çalışmada iki yapı oluşturulmuştur. Birinci yapı verilen girişlere göre tahmin gününe dair 24 saatlik tahmin sonuçlarını vermektedir. İkinci yapıda ise, öncelikle tahmin gününün tepe yükleri, ikincil olarak da saatlik ortalama yükleri tahmin edilmiş ve bu sonuçlar birleştirilerek 24 saatlik tahmin sonuçları elde edilmiştir.

Khotanzad ve arkadaşları, Amerika'nın 10 bölgesi için YSA kullanarak 24 saatlik yük tahmini yapmışlardır (Khotanzad vd., 1997). Çalışmada, yük tahmini için saatlik, günlük ve haftalık olmak üzere üç modül oluşturulmuştur. Haftalık modül, tahmin gününün 24 saatlik sıcaklık verisi, bir hafta öncesindeki günün 24 saatlik yük ve sıcaklık verilerini kullanarak tahmin gününe dair 24 saatlik yük tahmin sonuçlarını vermektedir. Günlük modül, tahmin gününün 24 saatlik sıcaklık verisi, bir gün öncesinin 24 saatlik yük ve sıcaklık verilerini kullanarak tahmin gününe dair 24 saatlik yük tahmin sonuçlarını vermektedir. Saatlik modül ise gün endeksi, tahmin saatinin sıcaklık verisi, tahmin saatinin 24 ve 48 saat öncesi yük, sıcaklık ve bağıl nem verileri kullanılarak o saatin tahminini vermektedir. Bu üç modülün sonuçları her saat için yinelemeli en küçük kareler algoritması ile birleştirilmiş ve tahmin sonuçları elde edilmiştir. Yerel tatil günleri, tiplerine göre cumartesi ya da pazar günü gibi düşünülerek endekslenmiş ve bu günlere dair tahmin gerçekleştirilmemiştir. Çalışmada oluşturulan diğer bir yapıda ise günün 24 saati 01:00-09:00, 10:00-14:00 ve 19:00-22:00, 15:00-18:00, 23:00-24:00 olmak üzere dört kategoriye ayrılmıştır. Her bir saat için ayrı sinir ağları oluşturularak tahmin sonuçları elde edilmiştir. İkinci yapının daha düşük hata yüzdeleri verdiği görülmüştür.

1.2.3. Hibrid yöntemler ile yapılan çalışmalar

Başaran Filik ve Kurban, Türkiye geneli 2002 yılı Ocak, Şubat ve Mart ayları saatlik yük verilerini kullanarak aynı günler için AR modelleme ve YSA ile 24 saatlik yük tahmininde bulunmuşlardır (Başaran Filik ve Kurban, 2007). Çalışmada üç tahmin yapısı oluşturulmuştur. Tahmin, birinci yapıda yalnızca AR modelleme, ikinci yapıda da

sadece YSA ile gerçekleştirilmiştir. Üçüncü yapıda ise, AR modelleme ile elde edilen katsayılar YSA girişi olarak kullanılmış ve 24 saatlik tahmin sonuçları elde edilmiştir. Üçüncü yapıdan elde edilen tahmin sonuçlarının daha düşük hata yüzdeleri verdiği görülmüştür.

Ceylan ve Demirören, 2002 ve 2003 yılları için yapay sinir ağları ve regresyon analizini benzer gün yöntemi ile birlikte kullanarak Türkiye-Gölbaşı bölgesine ait 24 saatlik yük tahmini yapmışlardır (Ceylan ve Demirören, 2004). Benzer günlerin seçiminde yük tüketimleri, maksimum ve minimum sıcaklık değerleri, gün tipi endeksleri dikkate alınmıştır. Çalışmada, YSA ve regresyon ile iki farklı tahmin sonucu elde edilmiş, YSA tahmin sonuçlarının daha düşük hata yüzdeleri verdiği görülmüştür.

Chen ve arkadaşları, Çin-Guizhou bölgesi için en küçük kareler yöntemi, Destek Vektör Makineleri (Support Vector Machines-SVM) ve dalgacık dönüşümü kullanarak 72 saatlik yük tahmini yapmıştır (Chen vd., 2008). Çalışmada, 2005-2006 yılları yarım saatlik yük verileri ve 2003-2006 yılları günlük ortalama sıcaklık verileri kullanılmıştır. Yerel tatil günlerine ait veriler o günün bir gün öncesi ve bir gün sonrası yük tüketimlerinin ortalaması ile değiştirilmiştir. Giriş-çıkış vektörleri ayrık dalgacık dönüşümü ile belirlenmiş, tahmin sonuçları en küçük kareler yöntemi ve destek vektör makinelerini birlikte kullanılarak oluşturulan yapıdan elde edilmiştir. Ayrık dalgacık dönüşümü db3 dalgacığı ile üçüncü seviyeden gerçekleştirilmiştir.

Chen ve arkadaşları, benzer gün yöntemi, dalgacık dönüşümü ve YSA ile 24 saatlik yük tahmininde bulunmuşlardır (Chen vd., 2010). Çalışmada bağımsız değişken olarak rüzgâr-soğuğu sıcaklığı, nem, bulut yoğunluğu, yağış, rüzgâr hızı ve gün tipi endeksi kullanılmıştır. Dalgacık dönüşümü db4 dalgacığı ile gerçekleştirilmiştir. Dönüşüm ile elde edilen alçak ve yüksek frekans bileşenleri için ayrı sinir ağları oluşturulmuştur. Her iki sinir ağında da hava verileri ve gün tipi endeksi kullanılmıştır. İki ağın çıktıları birleştirilerek 24 saatlik tahmin sonuçları elde edilmiştir.

Gao ve Tsoukalas, sadece yük verisi kullanarak bulanık mantık, YSA ve dalgacık dönüşümü ile 24 saatlik yük tahmini yapmıştır (Gao ve Tsoukalas, 2001).

Çalışmada, sinir ağı ağırlıkları dalgacık dönüşümü ile belirlenmiştir. Özel günlerin tahmine etkisi için bulanık kural tabanı oluşturulmuştur. Sinir ağı eğitiminde tahmin günü öncesindeki son üç günün saatlik yük verileri kullanılmıştır.

He ve arkadaşları, yük, sıcaklık, nem ve yağış verilerini kullanarak bulanık mantık tabanlı YSA ile Çin-Hebei Bölgesi için 2002 yılına ait 24 saatlik yük tahmini yapmıştır (He vd., 2006). Çalışmada, bağımsız değişkenler maksimum koşulsal entropi teorisi ile belirlenmiştir. Belirlenen bağımsız değişkenler kullanılarak bulanık sınıflandırma ile tahmin edilecek gün için giriş verileri seçilmiştir. Seçilen veriler YSA eğitiminde kullanılmıştır.

Kim ve arkadaşları, 1990-1995 yıllarına ait saatlik yük verilerini kullanarak, YSA ve bulanık mantık ile Kore 1996-1997 yılları özel günleri için 24 saatlik yük tahmini yapmıştır (Kim vd., 2000). Çalışmada, YSA ile maksimum ve minimum yük değerlerine göre oranlanmış veriler kullanılarak 24 saatlik oranlanmış yük tahmini yapılmıştır. Bulanık mantık ile de maksimum ve minimum yük tahminleri yapılmış, bu tahminler ile YSA'dan elde edilmiş oranlanmış tahmin sonuçları ile birleştirilerek 24 saatlik yük tahmini sonuçları elde edilmiştir. Hafta içi günler, cumartesi günü, pazar günü, pazartesi günü ve Kore geleneksel tatil günleri için farklı sinir ağları oluşturulmuştur.

Lu ve Zhou tarafından yük verisi ve hava verisi kullanılarak Çin'in bir bölgesi için 24 saatlik yük tahmini yapılan çalışmada, RTF sinir ağı parametreleri PSO ile belirlenmiştir (Lu ve Zhou, 2009). Çalışmada, hava verisi olarak güneşli, bulutlu ve yağmurlu olma durumları dikkate alınmıştır. Sinir ağları eğitiminde tahmin gününün bir gün öncesi yük ve hava verileri kullanılmıştır.

Pandey ve arkadaşları, Kanada'nın bir bölgesi için dalgacık dönüşümü tabanlı RTF sinir ağları, dalgacık dönüşümü tabanlı zaman serileri analizi ve dalgacık dönüşümü tabanlı bulanık sinir ağlarını birlikte kullanarak 24 ve 168 saatlik yük tahmininde bulunmuşlardır (Pandey vd., 2010). Çalışmada bağımsız değişken olarak sıcaklık verisi kullanılmıştır. Ayrık dalgacık dönüşümü ile yük verisi alçak ve yüksek

frekans bileşenlerine ayrılmış, alçak frekans bileşenleri yeniden yapılandırılarak eğitim kümesinde kullanılmıştır. Dönüşüm db2 dalgacığı ile üçüncü seviyede gerçekleştirilmiştir.

Song ve arkadaşları Kore geleneksel tatil günlerinin cumartesi ve pazartesi günlerine denk gelmesi durumunda tahmin hatalarının azaltılmasına yönelik çalışmışlardır (Song vd., 2005). Çalışmada, Kore'nin 1996-1997 yılları özel günleri için 24 saatlik yük tahmini bulanık lineer regresyon yöntemi ile yapılmıştır.

Xu ve arkadaşları, yük ve sıcaklık verisi kullanarak Çin-Xi'an Bölgesi için SVM tabanlı gerçek zamanlı 24 saatlik yük tahmini yapmıştır (Xu vd., 2005). Çalışmada kullanılan eğitim veri kümesi tahmin edilecek saate dair geçmiş yük verileri (60 günlük), tahmin saati (1, 2, ... , 24), tahmin saatine dair ortalama sıcaklık tipi (sıcak $>20^{\circ}\text{C}$: 1, soğuk $\leq 20^{\circ}\text{C}$: 0) ve gün tipi (hafta içi: 1, hafta sonu: 0) değişkenlerinden oluşmaktadır. Bu veriler kullanılarak regresyon fonksiyonu SVM ile oluşturulmuş, tahmin sonuçları sıralı minimal optimizasyon algoritması denen bir algoritma ile elde edilmiştir.

Yalçınöz ve arkadaşları, bağımsız değişken olmaksızın 1991-1993 yılları yıllık ve 1994-2001 yılları aylık yük verilerini kullanarak YSA geri yayılım algoritması ve hareketli ortalamalar yöntemi ile Türkiye-Niğde bölgesine ait 2001-2004 yılları için aylık yük tahmini yapmışlardır (Yalçınöz vd., 2002). Yıllık yük verileri aylara göre oranlanarak aylık değerler elde edilmiş ve kullanılmıştır. Çalışmada, kış ve bahar aylarında YSA tahmin sonuçlarının, yaz ve güz aylarında da hareketli ortalamalar yöntemi ile elde edilen sonuçların daha düşük hata yüzdeleri verdiği görülmüştür.

Zhu ve arkadaşları, Hubei-Çin için yaptıkları 20 Haziran 2005 tarihine ait 24 saatlik yük tahmini çalışmasında Görgül Kip Ayrışımı (GKA) ve SVM'yi birlikte kullanmışlardır (Zhu vd., 2007). Çalışmada geçmiş yük verisi öz kip fonksiyonlarına (IMF) ayrıştırılmış ve sıcaklık verisinin de eklendiği her bir IMF için farklı SVM modelleri oluşturularak tahmin sonuçları elde edilmiştir. Çalışmada eğitim seti ile ilgili açık bilgi verilmemiştir.

Analitik yöntemler ile yapılan çalışmalar Çizelge 1.1’de verilmiştir.

Çizelge 1.1. Analitik yöntemler ile yapılan çalışmalar.

Yazar	Tahmin Bölgesi	Tarih	Tahmin Tipi	Veri Tipi	Kullanılan Yöntem
Mu ve arkadaşları	Hainan, Çin	2008	7 günlük	• Yük • Hava durumu • Sıcaklık • Gün tipi • Sosyolojik koşullar • Gün farkı	• Benzer Gün Yöntemi
Nalbant ve arkadaşları	Kütahya, Türkiye	2005-2009	Aylık	• Yük	• En Küçük Kareler Yöntemi
Sümer ve arkadaşları		2006-2010	Aylık	• Yük	• ARIMA modelleme • SARIMA modelleme • Regresyon Yöntemleri
Taylor ve McSharry		2005	24 saatlik	• Yük	• AR modelleme • ARIMA modelleme • Temel Bileşen Analizi • Holt-Winters Üstel Yumuşatma Yöntemi

Yapay zeka yöntemleri ile yapılan çalışmalar Çizelge 1.2’de verilmiştir.

Çizelge 1.2. Yapay zeka yöntemleri ile yapılan çalışmalar.

Yazar	Tahmin Bölgesi	Tarih	Tahmin Tipi	Veri Tipi	Kullanılan Yöntem
Azadeh ve arkadaşları	İran	2003-2005	24 saatlik	• Yük • Sıcaklık	• YSA
Dongxiao ve Jie		17 Kasım 2009	24 saatlik	• Yük • Sıcaklık • Yağış	• RTF Sinir Ağları
Gontar ve Hatziargyriou	Crete, İtalya	1999	48 saatlik	• Yük	• RTF Sinir Ağları
Jain ve Satish			24 saatlik	• Yük • Sıcaklık • Gün tipi	• SVM
Khotanzad ve arkadaşları	Amerika’nın 10 bölgesi		24 saatlik	• Yük • Sıcaklık • Nem • Gün tipi	• YSA

Hibrid yöntemler ile yapılan çalışmalar Çizelge 1.3'te, gerçekleştirilen tez çalışmasının literatürdeki yeri Çizelge 1.4'te verilmiştir.

Çizelge 1.3. Hibrid yöntemler ile yapılan çalışmalar.

Yazar	Tahmin Bölgesi	Tarih	Tahmin Tipi	Veri Tipi	Kullanılan Yöntem
Başaran Filik ve Kurban	Türkiye	Ocak, Şubat, Mart 2002	24 saatlik	• Yük	• AR modelleme • YSA
Ceylan ve Demirören	Gölbaşı, Türkiye	2002-2003	24 saatlik	• Yük • Sıcaklık • Gün tipi	• Benzer Gün Yöntemi • YSA • Regresyon Analizi
Chen ve arkadaşları	Guizhou, Çin		72 saatlik	• Yük • Sıcaklık	• SVM • Dalgacık Dönüşümü
Chen ve arkadaşları			24 saatlik	• Yük • Rüzgâr-soğuğu sıcaklığı • Nem • Bulut yoğunluğu • Yağış • Rüzgâr hızı • Gün tipi	• Benzer Gün Yöntemi • Dalgacık Dönüşümü • YSA
Gao ve Tsoukalas			24 saatlik	• Yük	• Bulanık Mantık • Dalgacık Dönüşümü • YSA
He ve arkadaşları	Hebei, Çin	2002	24 saatlik	• Yük • Sıcaklık • Nem • Yağış	• Maksimum Koşulsal Entropi Teorisi • Bulanık Mantık • YSA

Çizelge 1.3. Hibrid yöntemler ile yapılan çalışmalar (devam).

Yazar	Tahmin Bölgesi	Tarih	Tahmin Tipi	Veri Tipi	Kullanılan Yöntem
Kim ve arkadaşları	Kore (Özel günler)	1996-1997	24 saatlik	• Yük	• Yapay Sinir Ağları • Bulanık Mantık
Lu ve Zhou	Çin'in bir bölgesi		24 saatlik	• Yük • Hava verisi (güneşli, bulutlu, yağmurlu)	• RTF Sinir Ağları • PSO
Pandey ve arkadaşları	Kanada'nın bir bölgesi		24 saatlik 168 saatlik	• Yük • Sıcaklık	• Dalgacık Dönüşümü • RTF Sinir Ağları • Zaman Serileri Analizi • Bulanık Sinir Ağları
Song ve arkadaşları	Kore (Özel günler)	1996-1997	24 saatlik	• Yük	• Bulanık Lineer Regresyon Yöntemi
Xu ve arkadaşları	Xi'an, Çin		24 saatlik	• Yük • Sıcaklık • Gün tipi	• SVM • Minimal optimizasyon algoritması
Yalçınöz ve arkadaşları	Niğde, Türkiye	2001-2004	Aylık	• Yük	• YSA • Hareketli Ortalamalar Yöntemi
Zhu ve arkadaşları	Hubei, Çin	20 Haziran 2005	24 saatlik	• Yük • Sıcaklık	• Görgül Kip Ayırımı • SVM

Çizelge 1.4. Tez çalışmasının literatürdeki yeri.

	Tahmin Bölgesi	Tarih	Tahmin Tipi	Veri Tipi	Kullanılan Yöntem
Tez Çalışması	Türkiye	2009-2010	24 saatlik	• Yük	• YSA • Dalgacık Dönüşümü • RTF Sinir Ağları • Görgül Kip Ayırışımı

1.3. Tezin Amacı

Bu çalışmada, Türkiye Elektrik İletim Anonim Şirketi (TEİAŞ)'dan alınan Türkiye geneli 1 Aralık 2008-31 Aralık 2010 saatlik yük verileri kullanılarak Türkiye geneli için 24 saatlik yük tahmini yapılmıştır. Bölgesel sıcaklık verisi temini ve kullanımının aksine, ülke geneli gibi geniş ölçekli tahmin çalışmaları için etkin bir sıcaklık verisi elde etme ve kullanmanın zorluğundan dolayı çalışmada sıcaklık verisi kullanılmamıştır. Çalışmada tahmin, YSA, DD ve YSA, DD ve RTFSA, GKA ve RTFSA olmak üzere oluşturulan dört yapı ile gerçekleştirilmiştir. Yapılara ek olarak hata değerlerini düşürmek amacıyla tahmin için bir regülasyon hesabı tanımlanmıştır. Tahmin hata yüzdeleri MAPE cinsinden, ortalama günlük MAPE, maksimum günlük MAPE olarak hesaplanmış ve dört yapı için hata karşılaştırılması yapılmıştır. Simülasyon çalışması, *MATLAB GUI* kullanılarak hazırlanan kullanıcı arayüzü, YZ-KDYT ile 2009-2010 yılları için gerçekleştirilmiştir.

Buraya kadar tezin ana konusu hakkında bir giriş bilgisi ve konu ile ilgili literatür araştırmasına yer verilmiştir. Tez çalışmasının ikinci bölümünde yük tahmini konusu; yük tahmini çeşitleri, tahmini etkileyen faktörler ve tahmin çalışmalarında kullanılan yöntemler açısından irdelenmiştir. Bölüm 3'de ise tez çalışmasında kullanılan yöntemler hakkında teorik bilgiler yer almaktadır. Bölüm 4'te ise veri analizi yapılmıştır. Yük tüketim eğrisinin değişen koşullara göre değişimi incelenmiştir. Bölüm

5'te tez çalışmasında kullanılan yapılar ve Bölüm 6'da çalışma kapsamında ortaya çıkarılan YZ-KDYT arayüzü anlatılmıştır. Bölüm7'de ise çalışma kapsamında elde edilen uygulama sonuçlarına yer verilmiştir. Son olarak 8. Bölüm'de çalışmadan elde edilen sonuçlar tartışılmıştır.

2. YÜK TAHMİNİ

Elektrik enerjisi dağıtımında temel ilke tüketiciye sürekli, güvenli ve ekonomik enerji sunabilmektir. Artan yük taleplerine karşın bu ilkelerden taviz verilmemelidir. Bu nedenle, elektrik enerji sistemi planlaması enerji kapsamında üzerinde durulması gereken en önemli nokta haline gelmiştir. Elektrik enerji sistemi planlamasının ilk adımı olan yük tahmini, elektrik enerjisinin ekonomik üretim ve dağıtımının gerçekleştirilebilmesi, sistem işletme koşullarının iyileştirilebilmesi, sistem kontrolünün daha verimli sağlanabilmesi ve elektrik enerji fiyatlandırması konularında büyük önem taşımaktadır.

Gereğinden düşük yük tahminine dayalı yapılan planlama tüketiciye sunulan enerji arzının kısıtlanmasına, gereğinden fazla yük tahminine dayalı yapılan planlama da ekonomik olmayan işletme koşullarına neden olacaktır (Pandey vd., 2010). Bu nedenle, tüm planlama sürecini etkilediğinden, yük tahmininin etkili olması gerekmektedir. Etkili bir yük tahmini için esas doğru model ile doğru veriyi kullanmaktır. Dolayısıyla, problem çözümüne uygun modellerin ve tahmini etkileyen faktörlerin iyi belirlenmiş olması ve tahmin modellemesinde bu faktörlerin göz önünde bulundurulması gerekmektedir.

2.1. Yük Tahmini Çeşitleri

Yük tahmini, yapıldığı zaman aralığına göre kısa, orta ve uzun dönem yük tahmini olmak üzere üçe ayrılmaktadır:

- Kısa dönem yük tahmini (KDYT) bir saatten bir güne kadar yapılan tahmin,
- Orta dönem yük tahmini(ODYT) bir saatten bir yıla kadar yapılan tahmin,
- Uzun dönem yük tahmini (UDYT) de bir yıl ve daha fazlası için yapılan tahmin anlamına gelmektedir (Khotanzad vd., 1997).

KDYT ile güç santralleri arasında yük paylaşımı ve jeneratörlerin devreye girip çıkma zamanlarının belirlenmesi amaçlanır. ODYT, satış tarifleri, bakım periyotları ve yakıt kaynaklarını belirleyerek iletim ve dağıtım sistemleri ile kısa sürede devreye

alınabilecek santrallerin belirlenmesi amacı ile yapılır. UDYT ile yeni enerji santrali ihtiyacının belirlenmesi amaçlanır.

2.2. Yük Tahminini Etkileyen Faktörler

Yük talep karakteristiği tüketildiği bölge ve bağımsız değişken olarak adlandırılan haftanın günlerine, sosyolojik, meteorolojik, demografik ve ekonomik faktörler göre değişkenlik göstermektedir. Yük tahmini çalışmasının düşük hata yüzdelerinde gerçekleştirilebilmesi için bu faktörlerinde iyi değerlendirilmesi ve çalışmada göz önünde bulundurulması gerekmektedir.

Yük talep karakteristiği tüketildiği bölgeye göre değişmektedir. Sanayi bölgesinin yük karakteristiği, yerleşim bölgesinin yük karakteristiği ile farklılık göstermektedir. Bir şehir ya da bir coğrafi bölgenin yük karakteristiği ile o şehir ya da coğrafi bölgenin bulunduğu ülkenin genel yük karakteristiği de farklılık göstermektedir.

Yük talep karakteristiği haftanın günleri (gün tipi) için de farklılık göstermektedir. Hafta içi günlerinin yük karakteristiği hafta sonu günlerinin yük karakteristiğine göre oldukça farklıdır.

Aynı hafta sonu günlerinde olduğu gibi, yük karakteristiği dini bayramlar, milli bayramlar ve yılbaşı tatillerinde de değişmektedir. Hatta bu günler kendi aralarında da farklı yük karakteristikleri taşırlar. Bu nedenle, yerel tatil günleri ve normal günler için tek bir model oluşturmak oldukça zordur. Bunun yerine, literatürde yerel tatil günlerine ait verilerin değiştirilmesi ya da bu günlerin tahmin çalışmasına dahil edilmemesi önerilmiştir (Osman vd., 2009).

Yük talebi demografik koşullara göre farklılık gösterir. Nüfusun artması demek elektrik enerjisi tüketiminin de artması anlamına gelmektedir. Aynı şekilde nüfusun azalması elektrik enerjisinde azalma ihtimalini göstermektedir.

Yük talebi değişen ekonomik koşullara karşı da değişkenlik gösterir. Gayri safi milli hâsıla, ithalat ve ihracat oranları ülkenin ekonomik koşullarını sergilediği gibi, yük talebini de etkilemektedir.

Meteorolojik koşullar (sıcaklık, yağış, nem, rüzgâr, vb.) da yük talebini etkileyen faktörler arasındadır. Örneğin, hava sıcaklığın düşmesi ısınma amaçlı elektrik enerjisi tüketimini doğurur. Bu da havaların soğuması ile elektrik enerjisi tüketiminin artacağı düşünürebilir.

Demografik ve ekonomik koşullar genellikle UDYT çalışmalarında kullanılmıştır (Kıran vd., 2012). Yapılan KDYT çalışmalarında bağımsız değişken olarak genellikle sıcaklık verisi kullanılmıştır (Azadeh vd., 2009; Ceylan ve Demirören, 2004; Chen vd., 2008; Dongxiao ve Jie, 2011; Gontar ve Hatziargyriou, 2001; He vd., 2006; Jain ve Satish, 2008; Khotanzad vd., 1997; Lu ve Zhou, 2009; Mu vd., 2010; Pandey vd., 2010; Xu vd., 2005; Zhu vd., 2007). Bazı çalışmalarda sıcaklık verilerine ek olarak rüzgâr, nem, yağış gibi meteorolojik veriler de kullanılmıştır (Chen vd., 2010; Dongxiao ve Jie, 2011; He vd., 2006; Khotanzad vd., 1997; Lu ve Zhou, 2009). Bunun yanı sıra sadece yük verisinin kullanıldığı çalışmalar da yapılmıştır (Gao vd., 2001; Gontar ve Hatziargyriou, 2001; Nalbant vd., 2005; Sümer vd., 2009; Taylor vd., 2007; Yalçınöz vd., 2002). Yapılan çalışmalar genel olarak bölgesel yük tahmini üzerinde yoğunlaşmıştır (Ceylan ve Demirören, 2004; Chen vd., 2008; Gontar ve Hatziargyriou, 2001; He vd., 2006; Khotanzad vd., 1997; Lu ve Zhou, 2009; Nalbant vd., 2005; Pandey vd., 2010; Xu vd., 2005; Zhu vd., 2007).

2.3. Yük Tahmininde Kullanılan Yöntemler

Yük tahmininde kullanılan yöntemler genel olarak analitik yöntemler ve yapay zekâ yöntemleri olarak sınıflandırılmaktadır. Benzer gün yaklaşımı, en küçük kareler yöntemi (EKKY), regresyon analizi, zaman serisi analizi ve dalgacık dönüşümü analitik yöntemlerin başında gelmektedir. Analitik yöntemler, normal şartlarda iyi çalışırken, hava değişimlerine, sosyolojik ve ekonomik değişimlere ve tatil günlerindeki enerji talebi değişimlerine doyurucu sonuçlar verememektedir. Bu nedenle, yapay zeka

yöntemleri önem kazanmıştır. YSA, bulanık mantık, SVM, genetik algoritma (GA), parçacık sürü optimizasyonu (PSO), karınca kolonisi optimizasyonu (KKO) tabanlı yöntemler yük tahmini için kullanılan yapay zeka yöntemleri arasındadır. GA, KKO ve PSO sezgisel optimizasyon algoritmalarıdır ve tahmin çalışmalarında direk olarak kullanılmak yerine sinir ağları parametrelerinin belirlenmesinde kullanılmışlardır. Bunların yanı sıra birden fazla yapay zeka yönteminin, birden fazla analitik yöntemin ya da analitik yöntemler ve yapay zekâ yöntemlerinin birlikte kullanıldığı hibrid yöntemler de yük tahmini amacı ile kullanılmaktadır.

2.3.1. Analitik yöntemler

Yük tahmini çalışmalarında kullanılan analitik yöntemlerin başında benzer gün yaklaşımı, en küçük kareler yöntemi, regresyon analizi, zaman serisi analizi ve dalgacık dönüşümü gelmektedir.

2.3.1.1. Benzer gün yaklaşımı

Bu yaklaşım doğrudan bir tahmin yöntemi olmamakla birlikte tahmin çalışmasında kullanılacak veri kümesinin belirlenmesinde kullanılmaktadır. Yaklaşımın temeli veri kümesinden tahmini yapılacak gün ile benzer karakteristik gösteren günlerin bulunmasına dayanır (Mandal vd., 2006). Günlerin benzer karakteristik göstermesi meteorolojik koşullar (hava durumu), ekonomik koşullar, demografik koşullar, sosyolojik koşullar (normal gün, yerel tatil günü) ve gün tipi (pazartesi, salı, çarşamba, perşembe, cuma, cumartesi, pazar) özelliklerinin benzer olması anlamına gelmektedir.

2.3.1.2. En küçük kareler yöntemi

EKKY, E. 2.1’de verilen problemde gerçek değerler ile tahmin değerleri arasındaki farkların karelerinin toplamını minimize ederek bağımlı ve bağımsız değişkenler arasındaki ilişkiyi belirler (Soliman ve Al-Kandari, 2010).

$$\underline{Z} = H \cdot \underline{\theta} + \underline{v} \quad (E.2.1)$$

Denklemde Z , θ , H ve v terimleri sırasıyla $m \times 1$ boyutunda bağımlı değişkenler vektörü, $n \times 1$ boyutunda ilişki parametreleri vektörü, $m \times n$ boyutunda bağımsız değişkenler vektörü ve $m \times 1$ boyutunda hata vektörü ifadelerini temsil etmektedir.

Yöntemin amacı, hata vektörünü minimize edecek θ parametre vektörünü tahmin etmektir. Bu durumda, minimize edilecek maliyet fonksiyonu E. 2.2’de verilmiştir. Bir fonksiyonunun türevinin sıfır olduğu noktanın o fonksiyonun en küçük değerini verdiği kuralı temel alınarak eşitliğin türevi sıfıra eşitlenir ve hatayı minimum yapan parametre vektörü E. 2.3’de gösterildiği şekilde hesaplanır.

$$\underline{v} = \left\{ \sum_{i=1}^m |Z_i - H_i \cdot \theta|^2 \right\}^{\frac{1}{2}} = \frac{1}{2} \cdot \left\{ (\underline{Z} - H \cdot \underline{\theta})^T \cdot (\underline{Z} - H \cdot \underline{\theta}) \right\}^{\frac{1}{2}} \quad (\text{E. 2.2})$$

$$\underline{\hat{\theta}} = [H^T - H]^{-1} \cdot H^T \underline{Z} \quad (\text{E. 2.3})$$

En küçük kareler yöntemi, öncelikle maliyet fonksiyonunu doğrusallaştırmak şartı ile doğrusal olmayan problemlerin çözümüne de uyarlanabilmektedir.

2.3.1.3. Regresyon analizi

Regresyon analizi, bağımlı ve bağımsız değişkenler arasındaki ilişkiyi belirlemek için kullanılan bir analiz yöntemidir. Bağımlı ve bağımsız değişkenler arasındaki dağılım diyagramının çizilmesiyle iki değişken arasında bir bağlantının kurulup kurulamayacağına, eğer kuruluyorsa nasıl bir fonksiyonun ele alınacağına karar verilir (Oğurlu, 2011). Regresyon analizi üç alt başlık altında incelenmektedir (Palit ve Popovic, 2005):

- Basit regresyon,
- Çoklu regresyon,
- Doğrusal olmayan regresyon.

Basit regresyon analizi ile bir bağımlı ve bir bağımsız değişken arasındaki doğrusal ilişki E.2.4 esas alınarak belirlenir.

$$Y_i = a_0 + a_1 \cdot X_i + \varepsilon_i \quad i = 1, 2, \dots, n \quad (\text{E. 2.4})$$

Denklemden X_i bağımsız değişkeninin değerine göre Y_i bağımlı değişkeninin değerini ε_i hatası ile belirleyecek a_0 ve a_1 katsayıları tahmin edilmektedir. Analizin amacı, ε_i hatasını minimum yapacak şekilde a_0 ve a_1 katsayılarının belirlenmesidir. E. 2.5’de gösterilen maliyet fonksiyonu EKKY ile hesaplanabilir (Palit ve Popovic, 2005).

$$\varepsilon_i = \sum_{i=1}^n (Y_i - (a_0 + a_1 \cdot X_i))^2 \quad (\text{E. 2.5})$$

En küçük hata değerini bulabilmek için maliyet fonksiyonunun a_0 ve a_1 ’e göre kısmi türevlerinin sıfıra eşitlenmesi gerekmektedir. Kısmi türevlerin alınması işleminden sonra E. 2.6 ve E. 2.7’de gösterilen denklemler elde edilmektedir. Bu denklemler normal denklemleri olarak adlandırılmaktadır (Yoldaş, 2006). Elde edilen normal denklemleri çözülerek a_0 ve a_1 katsayıları E. 2.8 ve E. 2.9’da gösterildiği şekilde bulunmaktadır. Denklemlerde kullanılan $\bar{X}$ ve $\bar{Y}$ terimleri sırasıyla X ve Y değişkenlerinin aritmetik ortalamalarını ifade etmektedir.

$$\sum Y_i = n \cdot a_0 + a_1 \cdot \sum X_i \quad (\text{E. 2.6})$$

$$\sum Y_i \cdot X_i = a_0 \cdot \sum X_i + a_1 \cdot \sum X_i^2 \quad (\text{E. 2.7})$$

$$a_0 = \bar{Y} - a_1 \cdot \bar{X} \quad (\text{E. 2.8})$$

$$a_1 = \frac{\sum X_i \cdot Y_i - \bar{Y} \cdot \sum X_i}{\sum X_i^2 - \bar{X} \cdot \sum X_i} \quad (\text{E. 2.9})$$

Çoklu regresyon analizi ise bir bağımlı ve birden fazla bağımsız değişken arasındaki ilişkiyi belirlemede kullanılmaktadır. Bağımlı ve bağımsız değişkenler arasındaki doğrusal ilişki E. 2.10'da gösterildiği gibidir.

$$Y = a_0 + a_1 \cdot X_1 + a_2 \cdot X_2 + \dots + a_n \cdot X_n \quad (\text{E. 2.10})$$

İki bağımsız değişken içeren çoklu regresyon modelinde çözülmesi gereken normal denklemleri E. 2.11, E. 2.12 ve E. 2.13'de verilmiştir (Yalmaçlı, 2010).

$$\sum Y = 2 \cdot a_0 + a_1 \cdot \sum X_1 + a_2 \cdot \sum X_2 \quad (\text{E. 2.11})$$

$$\sum X_1 \cdot Y = a_0 \cdot \sum X_1 + a_1 \cdot \sum x_1^2 + a_2 \cdot \sum X_1 \cdot X_2 \quad (\text{E. 2.12})$$

$$\sum X_2 \cdot Y = a_0 \cdot \sum X_2 + a_2 \cdot \sum X_1 \cdot X_2 + a_2 \cdot \sum X_2^2 \quad (\text{E. 2.13})$$

X bağımsız değişkeni ile Y bağımlı değişkeni arasında doğrusal olmayan bir ilişki varsa, bu değişkenler arasındaki bağıntı genel olarak E. 2.14'de gösterildiği şekilde tanımlanmaktadır.

$$Y = a_0 + a_1 \cdot X + a_2 \cdot X^2 + a_3 \cdot X^3 + a_4 \cdot X^4 + \dots \quad (\text{E. 2.14})$$

Eşitliğin kısmi türevleri sıfıra eşitlenerek normal denklemleri elde edilmekte ve bu denklemlerin çözülmesi ile $a_0, a_1, a_2, \dots$ katsayıları hesaplanmaktadır.

2.3.1.4. Zaman serisi analizi

Zaman serileri, bir değişkenin belirli bir zaman dönemi içerisinde aldığı farklı değerlerin oluş zamanları esas alınarak sıralanmasıyla oluşan veri setleridir. Zaman serilerine ilişkin veriler zamanın belli anlarında rastlantısal değerler almaktadırlar, yani bu serilerin nasıl bir fonksiyonel yapıya bağlı olarak oluşacağı tam olarak

bilinememektedir. Ancak, ilgili serilere ilişkin çeşitli istatistiksel test ve analiz araçları kullanılarak fonksiyonel yapıya ait ipuçları sağlanabilmektedir (Oğurlu, 2011).

Zaman serisi analizi ile geçmiş verilerin gelecekte de benzer karakteristik göstereceği düşünülerek sayısal modeller oluşturulmakta ve bu modeller doğrultusunda geleceğe dair tahmin yapılmaktadır. Tek bir değişkene ait veri setiyle yapılan analizler tek değişkenli zaman serisi analizi olarak adlandırılmaktadır (Demirel, 2009). Tek değişkenli zaman serileri analizi genel olarak ilgili değişkenin gelecek değerlerinin tahmin edilmesi amacıyla kullanılmaktadır.

Zaman serisi analizinde genel olarak kullanılan analiz ve tahmin yöntemi Box-Jenkins tekniği olarak bilinmektedir. Bu teknik, kesikli doğrusal stokastik süreçlere dayanmaktadır. Box-Jenkins tahmin modelleri, AR modelleme, Hareketli Ortalama (Moving Average - MA), Otoregresif-Hareketli Ortalama (Auto Regressive-Moving Average – ARMA) ARIMA modelleme olarak sıralanmaktadır. Bu modellerden AR, MA ve ARMA durağan süreçlere uygulanırken ARIMA modelleri durağan olmayan süreçler için kullanılmaktadır (Oğurlu, 2011).

2.3.1.5. Dalgacık dönüşümü

Dalgacık dönüşümü temeli Fourier dönüşümüne dayanan bir sinyal analiz yöntemidir. Sürekli dalgacık dönüşümü (SDD), değişik boyutta bölgelerde sinyalin istenilen zaman aralığındaki frekans bileşenlerini belirlemek amacı ile Fourier ve kısa zaman Fourier dönüşümlerine alternatif olarak geliştirilmiştir. Ayrık dalgacık dönüşümü (ADD)'de ise sinyal istenilen seviyede alçak ve yüksek frekans bileşenlerine ayrılmaktadır. Dönüşüm ile ilgili ayrıntılı bilgi Bölüm 3'te verilmiştir.

2.3.2. Yapay zeka yöntemleri

2.3.2.1. Yapay sinir ağları

Yapay sinir ağları, insan beyninden esinlenerek geliştirilmiş, ağırlıklı bağlantılar aracılığıyla birbirine bağlanan ve her biri kendi belleğine sahip işlem elemanlarından oluşan paralel ve dağıtılmış bilgi işlem yapılarıdır (Pınarbaşı, 2009). YSA, bir sisteme ilişkin giriş ve çıkış verileri arasında ilişki kurarak önceden verilmeyen girişlere doğru çıkışlar üretebilme yeteneğine sahiptir. Giriş ve çıkış verileri arasında doğrusal bir ilişki olmadığı, matematiksel modellerin oluşturulmadığı durumlarda da YSA'nın tutarlı sonuçlar üretmesi, tahmin çalışmalarında oldukça önemli avantajlar sağlamaktadır. YSA ile ilgili ayrıntılı bilgi Bölüm 3'te verilmiştir.

2.3.2.2. Bulanık mantık

Bulanık mantık (Fuzzy Logic) kavramı 1965 yılında Lofty A. Zadeh tarafından geliştirilmiştir. Bulanık mantık ile klasik mantık arasındaki temel fark klasik mantığın önermelerin sadece '0 veya 1' ya da 'doğru veya yanlış' değerlerini kullanmasıdır. Verilerin tam ve net olması gerektiğinden, klasik mantık yöntemi ile karmaşık sistemlerin modellenmesi ve kontrol edilmesi oldukça zordur. Bulanık mantık yöntemi ise '0-1' aralığında her değeri kullanabilme özelliği ile karmaşık problemlerin çözümünü kolaylaştırmaktadır.

Bulanık kuramının merkez kavramı bulanık kümelerdir. Bulanık kümeler, kesin sınırlar göstermeyen, kişiden kişiye değişen yani matematiksel olarak kolayca ifade edilemeyen veri setleridir.

Bir bulanık küme kendi üyelik fonksiyonu ile temsil edilmektedir. Üyelik fonksiyonu 0 ile 1 arasındaki her değeri alabilmektedir. Böyle bir üyelik fonksiyonu ile verinin o fonksiyona aitliği konusunda, "kesinlikle ait" veya "kesinlikle ait değil" arasında istenilen incelikte ayarlama yapmak mümkün olmaktadır.

Bulanık mantıkta önemli bir diğer kavramda dilsel değişken kavramıdır. Dilsel değişken, sözcükler ve ifadelerle tanımlanabilen değişkenlerdir. Bir dilsel değişkenin değerleri de bulanık kümeler ile ifade edilebilmektedir.

Bulanık mantığın sağladığı en büyük fayda ise "insana özgü tecrübe ile öğrenme" olayının kolayca modellenmesi ve belirsiz kavramların bile matematiksel olarak ifade edilebilmesine olanak tanınmasıdır. Bu nedenle doğrusal olmayan sistemlere yaklaşım yapabilmek için özellikle uygundur. Bunun için kural tabanı oluşturulur. Kural tabanı incelenen konuya ilişkin insanlığın o güne kadar edindiği tüm gözlem, deneyim ve matematiksel bağıntıları yani tüm bilgiyi içermelidir. Kural tabanı ne kadar iyi ve geniş hazırlanırsa o kadar hassas ve doğru sonuçlar elde edilir.

Bulanık sistem yaklaşımları, yük tahmininde sıcaklık, gün tipi gibi yük değişim faktörleri içeren günlere ait yükün tahmin edilmesinde iyi sonuçlar verebilmektedir (Oğurlu, 2011).

2.3.2.3. Destek vektör makineleri (Support Vector Machines – SVM)

Temel olarak, istatistiksel öğrenme algoritmalarından biri olan ve Vapnik-Chervonenkis tarafından tanımlanan destek vektör makineleri, birçok gerçek problemde başarılı sonuçlar vermiştir. İlk başarılı uygulamaları 1990'lı yıllarda gerçekleştirilmiş olup, bu uygulamalardan sonra matematikçilerin ve yapay zekâ çalışan bilim insanlarının ilgi odağı olmuştur (Karagülle, 2008). Makine öğrenmesi alanındaki başarılı uygulamaları sonrasında, destek vektör makineleri uzaktan algılama alanında son yıllarda başarıyla kullanılan bir sınıflandırma yöntemidir.

Bu sınıflandırma yöntemi, veriyi daha yüksek bir boyuta dönüştürerek oluşturacağı bir hiper düzlem ile iki sınıfı birbirinden ayırma prensibini esas alır. Yüksek boyuta dönüşüm aşamasında değişik özelliklere sahip fonksiyonlar kullanılır. Bu fonksiyonlar kernel fonksiyonları olarak adlandırılmaktadır (Kavzoğlu ve Çölkesen, 2010). Kernel fonksiyonlarının kullanımı için bu fonksiyonların matematiksel ifadesinde bulunan bazı parametrelerin kullanıcı tarafından belirlenmesi gerekir (Kavzoğlu ve

Çölkesen, 2010). SVM son zamanlarda yüz tanıma, biyoloji, örüntü tanıma, veri madenciliği gibi alanlarda kullanılmaktadır.

3. UYGULAMADA KULLANILAN YÖNTEMLER

3.1. Yapay Sinir Ağları

YSA, insan beyninden esinlenerek geliştirilmiş, ağırlıklı bağlantılar aracılığıyla birbirine bağlanan ve her biri kendi belleğine sahip işlemci elemanlardan oluşan paralel ve dağıtılmış bilgi işlem yapılarıdır (Pınarbaşı, 2009). YSA, bir sisteme ilişkin giriş ve çıkış verileri arasında ilişki kurarak önceden verilmeyen girişlere doğru çıkışlar üretebilme yeteneğine sahiptir.

YSA'nın kullanıldığı uygulama alanları aşağıda verilmiştir (Sağıroğlu vd., 2003):

- Mühendislik: imalat sanayi, askeri proje uygulamaları, endüstriyel ürün tasarımı, bilgi yönetimi,
- Tıp: tıbbi görüntü işleme, tıbbi tanı koyma, organ morfasyonlarının belirlenmesi, biyomedikal uygulamalar (ses geliştirme uygulamalar, biyomedikal ürün yönetimi, protez tasarımı),
- Tarım: toplam ürün rekoltesinin tahmini,
- Hayvancılık: hayvan davranış modellerinin oluşturulması,
- Askeri: uzay ve havacılık sanayi, yüzey modelleme.

3.1.1. Yapay sinir ağı hücresi (Yapay nöron)

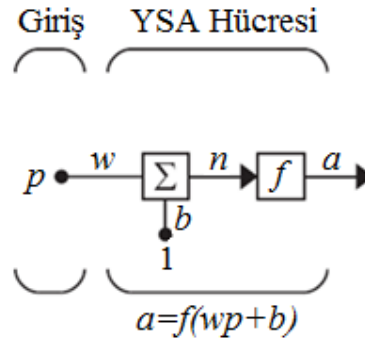
Yapay sinir ağları, biyolojik sinir ağlarına benzeştirilerek geliştirilmiş yapılardır. Temel bir yapay sinir ağı hücresi, biyolojik sinir hücresine göre çok daha basit bir yapıya sahiptir. Genel olarak, bir yapay sinir ağı hücresi girişler, ağırlıklar, toplama fonksiyonu, aktivasyon (transfer) fonksiyonu ve çıkışlardan oluşmaktadır.

Biyolojik sinir sistemi ile yapay sinir sistemi arasındaki benzeşimler Çizelge 3.1'de gösterilmiştir.

Çizelge 3.1. Biyolojik sinir sistemi ile YSA'nın benzeşimi (Sağıroğlu vd., 2003).

Biyolojik Sinir Sistemi	YSA
Nöron	İşlemci eleman
Dendrit	Toplama fonksiyonu
Hücre Gövdesi	Transfer fonksiyonu
Aksonlar	Yapay nöron çıkışı
Sinapslar	Ağırlıklar

Basit bir yapay sinir ağı hücresi modeli Şekil 3.1'de gösterilmiştir. Şekilde, p girişleri, w ağırlıkları, b kutuplama değerini, n aktivasyon potansiyeli ve a ise çıkışları ifade etmektedir.



Şekil 3.1. Yapay sinir ağı hücresi modeli (Beale vd., 2010).

3.1.1.1. Girişler

Girişler, kendinden önceki sinirlerden veya dış dünyadan aldıkları bilgiyi sinir ağına getirirler.

3.1.1.2. Ağırlıklar

Ağırlıklar, yapay sinir tarafından alınan girişlerin sinir üzerindeki etkisini belirleyen uygun katsayılardır. Her bir giriş kendisine ait bir ağırlığa sahiptir. Bir ağırlık değerinin büyük olması, o girişin yapay sinire güçlü bağlanması ya da önemli olması, küçük olması da zayıf bağlanması ya da önemli olmaması anlamına gelmektedir.

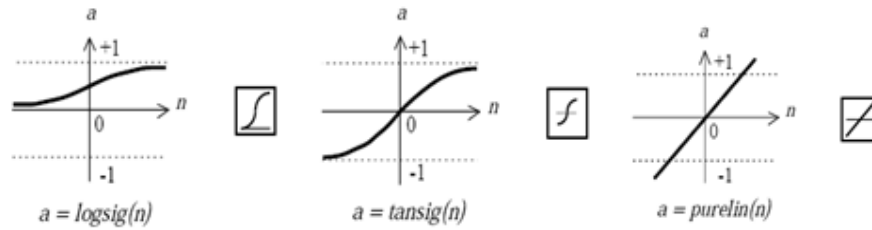
3.1.1.3. Toplama fonksiyonu

Toplama fonksiyonu ile her bir ağırlığın ait olduğu girişlerle çarpımının toplamalarını (aktivasyon potansiyeli) kutuplama değeri ile toplanarak aktivasyon fonksiyonuna gönderilir.

3.1.1.4. Aktivasyon fonksiyonu

Aktivasyon fonksiyonu, aktivasyon potansiyelini çıkış işaretini istenilen değerler arasında sınırlandıracak şekilde normalize eden bir fonksiyondur. Bu şekilde, zaman söz konusu olduğunda aktivasyon potansiyeli çıkışının değişmesine olanak sağlamaktadır.

Yaygın olarak kullanılan aktivasyon fonksiyonları, logaritmik sigmoid, tanjant sigmoid ve doğrusal fonksiyonlar olarak sıralanmaktadır. Bu fonksiyonların grafikleri Şekil 3.2’de gösterilmiştir.



Şekil 3.2. Yaygın olarak kullanılan aktivasyon fonksiyonları (Beale vd., 2010).

Logaritmik sigmoid fonksiyonu, lineer ve lineer olmayan davranış dengesi sergileyen kesin artan bir fonksiyon olarak tanımlanmaktadır (Haykin, 1998). Fonksiyonun formülü E. 3.1’de gösterilmiştir.

$$a = \text{logsig}(n) = \frac{1}{1 + e^{-n}} \quad (\text{E. 3.1})$$

Tanjant sigmoid fonksiyonu, doğrusal olmayışı ve türevi alınabilir özelliğinden dolayı doğrusal olmayan problemlerin YSA uygulamalarında en çok kullanılan aktivasyon fonksiyonlarından biridir. Tanjant sigmoid, giriş uzayının genişletilmesinde

etkili bir aktivasyon fonksiyonudur (Sağıroğlu vd., 2003). Fonksiyonun formülü E. 3.2’de gösterilmiştir.

$$a = \text{tansig}(n) = \frac{1 - e^{-2n}}{1 + e^{2n}} \quad (\text{E.3.2})$$

Doğrusal fonksiyon, giriş değerlerini çıkışa aynen aktaran fonksiyon olarak tanımlanmaktadır. Bu fonksiyon, YSA çıkış katmanında sıklıkla kullanılmaktadır.

$$a = \text{purelin}(n) = n \quad (\text{E.3.3})$$

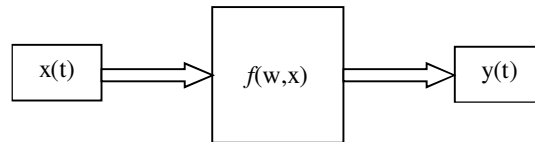
Sinir hücresi, aktivasyon fonksiyonunun eşik seviyesinin altında çıkış üretemez. Ayrıca, sinir ağına bir kutuplama değeri uygulanarak aktivasyon fonksiyonu artırılabilir. Kutuplama değeri, eşik değerinin negatif değerlisidir (Sağıroğlu vd., 2003).

3.1.1.5. Cıkışlar

Çıkışlar, aktivasyon fonksiyonunun sonucunun dış dünyaya veya diğer sinirlere gönderildiği yerdir. Bir sinirin bir tek çıkışı vardır. Sinirin bu çıkışı, kendinden sonra gelen herhangi bir sayıdaki diğer sinirlere giriş olabilir.

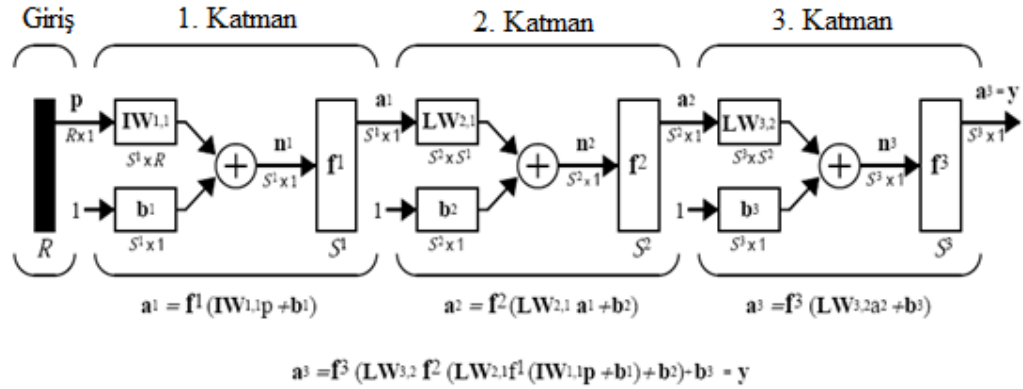
3.1.2. Yapay sinir ağlarının sınıflandırılması

Yapay sinir ağları yapılarına göre ileri beslemeli ve geri beslemeli sinir ağları olarak sınıflandırılmaktadır. İleri beslemeli sinir ağlarında sinyal iletimi, giriş katmanından çıkış katmanına doğru tek yönlü bağlantılar kurularak gerçekleştirilmektedir. Bu ağ yapısında, nöronların aynı katman içerisinde bağlantıları olmadığından gecikmeler yoktur. İleri beslemeli ağ yapısı Şekil 3.3’de gösterilmiştir.



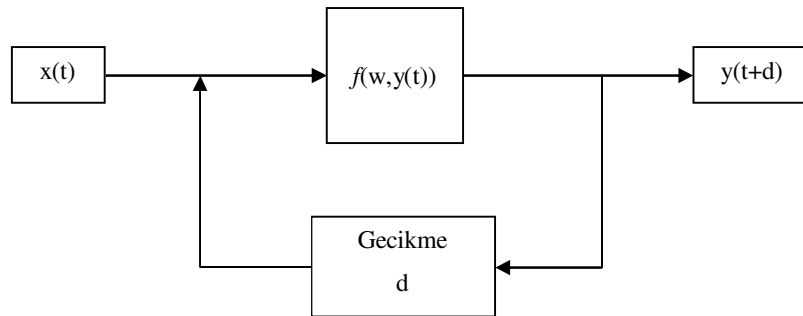
Şekil 3.3. İleri beslemeli ağ yapısı.

Çok katmanlı perseptronlar (Multi Layer Perceptrons-MLP), ileri beslemeli ağlara örnek olarak gösterilebilir. Bu yapılar bir giriş katmanı, bir veya daha fazla gizli katman ve bir çıkış katmanından oluşmaktadır. Her bir katmanda bir veya daha fazla nöron bulunmakta ve bu nöronlar üst katmanlardaki nöronlara bağlanmaktadır. Üç katmanlı bir YSA yapısının MATLAB gösterimi Şekil 3.4’de gösterilmiştir.



Şekil 3.4. 3 katmanlı YSA yapısı (Beale vd., 2010).

Geri beslemeli sinir ağları ise, çıkış ve gizli katman çıkışlarının giriş katmanı veya önceki gizli katmanlara geri beslendiği ağ yapılarıdır (Sağıroğlu vd., 2003). Böylece, bu ağ yapılarında hem ileri yönlü hem de geri yönlü iletim gerçekleşmiş olmaktadır. Geri beslemeli ağlarda, nöronların aynı katman içerisinde bağlantıları olduğundan gecikmeler meydana gelmektedir. Hopfield, Elman ve Jordan ağları bu yapıya örnek verilebilir. Geri beslemeli ağ yapısı Şekil 3.5’de gösterilmiştir.



Şekil 3.5. Geri beslemeli ağ yapısı.

3.1.3. Yapay sinir ağlarının öğrenmesi

Yapay sinir ağlarında nöron bağlantılarının ağırlık değerlerinin belirlenmesi işlemine “*ağın eğitilmesi*” denmektedir. Başlangıçta bu ağırlık değerleri rastgele olarak atanmakta ve YSA eğitildikçe bu ağırlıklar değişmektedir. Eğitimin amacı, ağa gösterilen örnekler için doğru çıktıları üretecek ağırlık değerlerini bulmaktır. Örnekler ağa defalarca gösterilerek en doğru ağırlık değerleri bulunmaya çalışılır. Ağın doğru ağırlık değerlerine ulaşması örneklerin temsil ettiği olay hakkında genellemeler yapabilme yeteneğine kavuşması demektir. Bu genelleştirme özelliğine kavuşması işlemine “*ağın öğrenmesi*” denmektedir. YSA öğrenme algoritmaları, göre danışmanlı, danışmansız ve takviyeli öğrenme olmak üzere üçe ayrılmaktadır (Sağıroğlu vd., 2003).

3.1.3.1. Danışmanlı öğrenme

Danışmanlı öğrenmede, ağa verilen girişlere karşı istenen çıkışlar verilmektedir. Ağ, bu çıkışları oluşturabilmek için kendi ağırlıklarını günceller. Ağırlıklar, ağ çıkışları ile istenen çıkışlar arasındaki hataya göre güncellenmektedir. Genelleştirilmiş delta kuralı ve geri yayılım algoritması danışmanlı öğrenme algoritmalarına örnek verilebilir (Sağıroğlu vd., 2003).

3.1.3.2. Danışmansız öğrenme

Danışmansız öğrenmede ağa öğrenme sırasında sadece örnek girdiler verilmektedir. Herhangi bir istenen çıkış bilgisinin verilmesine gerek yoktur. Girişte verilen bilgilere göre ağ, bu bilgilere göre her bir örneği kendi arasında sınıflandıracak şekilde kendi kurallarını oluşturmaktadır. Ağ, bağlantı ağırlıklarını aynı özellikte olan dokuları ayırabilecek şekilde düzenleyerek öğrenme işlemini tamamlamaktadır.

3.1.3.3. Takviyeli öğrenme

Takviyeli öğrenmede, ağın her eğitimi sonucunda elde ettiği sonucun girişe karşılık uygunluğunu değerlendiren bir ölçüt tanımlanmaktadır. Ağ, bu ölçütlere göre

ağırlıklarını güncellemektedir. Bu yaklaşım, uygunluk ölçütünün tanımlanması nedeniyle danışmanlı öğrenme, çıkış bilgisinin verilmemesi nedeniyle de danışmansız öğrenme yaklaşımlarına benzerlik göstermektedir.

YSA’da kullanılan eğitim fonksiyonları Çizelge 3.2’de gösterilmektedir.

Çizelge 3.2. YSA’da kullanılan eğitim fonksiyonları.

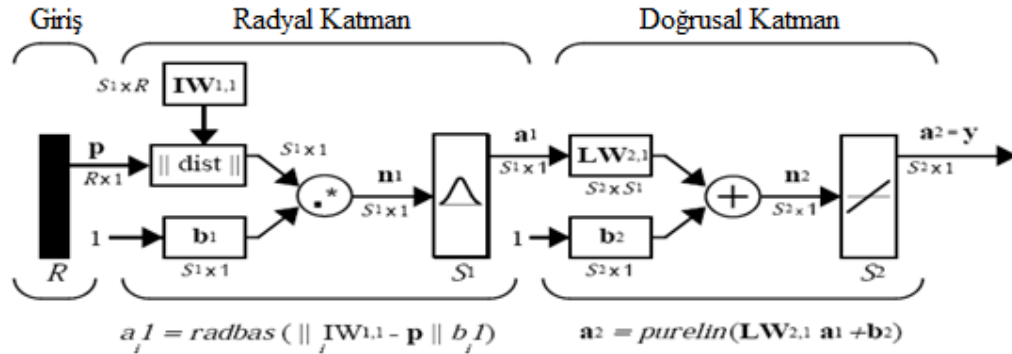
Fonksiyon	Algoritma
trainlm	Levenberg-Marquardt
trainbr	Bayesian Regularization
trainbfg	BFGS Quasi-Newton
trainrp	Resilient Backpropagation
trainscg	Scaled Conjugate Gradient
traincgb	Conjugate Gradient with Powell / Beale Restarts
traincgf	Fletcher-Powell Conjugate Gradient
traincgp	Polak-Ribière Conjugate Gradient
trainoss	One Step Secant
traingdx	Variable Learning Rate Gradient Descent
traingdm	Gradient Descent with Momentum
traingd	Gradient Descent

Tez çalışmasında kullanılan Levenberg-Marquardt fonksiyonu en hızlı eğitim fonksiyonudur (Beale vd., 2010). Doğrusal olmayan eğri uyurma problemlerinde iyi performans gösterdiğinden tahmin çalışmalarında kullanımı avantaj sağlamaktadır. Geniş ağlar için eğitim süresinin uzaması ve hafızada fazla yer kaplaması, bu fonksiyonun dezavantajları arasında sıralanmaktadır.

3.2. Radyal Tabanlı Fonksiyon Sinir Ağları

RTF sinir ağları, çok değişkenli modelleme ve yakınsama problemlerinde kullanılan YSA yapılarından (Sağıroğlu vd., 2003). Bu tür problemlerin çözümünde, daha az sayıda parametre kullanıldığından diğer sinir ağlarına göre daha iyi performans göstermektedirler. Ağ yapısı, Şekil 3.6’da görüldüğü gibi, radyal katman ve çıkış katmanı olmak üzere iki katmandan oluşmaktadır. RTF sinir ağlarında, ağırlık

vektörlerinin (nöronlar) yerleşimi kullanıcı tarafından tanımlanan bir yayılım katsayısı ile nöron sayısı da giriş vektörünün boyutuna göre belirlenmektedir. Radyal katmanda transfer fonksiyonu olarak *radbas* fonksiyonu ve çıkış katmanında transfer fonksiyonu olarak *purelin* fonksiyonu kullanılmaktadır.



Şekil 3.6. RTF sinir ağları yapısı (Beale vd., 2010).

RTF sinir ağlarında ağırlık fonksiyonu ağırlık ile giriş vektörünün çarpılması yerine ağırlık ve giriş vektörlerinin arasındaki farkın bulunmasıyla elde edilir. Net giriş fonksiyonu ise ağırlık fonksiyonu ile eşik değerinin toplanması değil de çarpılması ile elde edilir. Elde edilen net giriş fonksiyonu RTF sinir ağı transfer fonksiyonu ile işleme girer ve çıkış vektörleri elde edilir. RTF sinir ağı transfer fonksiyonu E. 3.4'de verilmiştir.

$$\text{radbas}(n) = e^{-n^2} \quad (\text{E. 3.4})$$

3.3. Dalgacık Dönüşümü

Sinyal işlemede kullanılan yöntemlerden en bilineni “Fourier Dönüşümü” dür. Fourier dönüşümü, sinyalin zaman düzleminden frekans düzlemine dönüştürülmesidir. Dönüşüm ile sinyal değişik frekanslardaki sinüzoidal serilere ayrılır. Bu sinüzoidler sadece genlik-frekans bilgisi içerir, zaman bilgisi içermezler. İncelenen sinyalin frekans içeriği zamanla değişmiyor (durağan) ise Fourier dönüşümü analiz için yeterlidir. Ancak, sinyalin frekans içeriği zamanla değişiyor (durağan olmayan) ise Fourier dönüşümü analiz için yetersiz kalmaktadır.



Şekil 3.7. Fourier dönüşümü (Türkmenoğlu, 2006).

Bu olumsuzlukları gidermek adına, 1946 yılında Denis Gabor tarafından “Kısa Zamanlı Fourier Dönüşümü” ortaya atılmıştır. Yöntem, sinyalin pencerelenmesi olarak bilinmektedir. Bir zaman aralığında sinyalin küçük bir kısmını çözümlemek için Fourier dönüşümü uygulanır. Kısa zamanlı Fourier dönüşümü, sinyalin zaman ve frekans düzlemi arasında kısa bir uzlaşma anını temsil eder. Sinyalin ne zaman ve hangi frekansta olduğunu gösteren bazı bilgiler sağlar, yani sinyali zaman ve frekansın iki boyutlu fonksiyonu olarak haritalar. Ancak, pencere büyüklüğü analiz süresince sabittir. Pencere ebatları küçülürse zaman çözünürlüğü artar, frekans bilgisi kaybolur. Pencere ebatları büyürse, frekans çözünürlüğü iyi, zaman çözünürlüğü düşük olur. Pencere geniş tutulduğunda düşük frekanslı bileşenler, dar tutulduğunda ise yüksek frekanslı bileşenler görülür. Pencere boyutu analiz süresince değiştirilebilirse daha doğru sonuçlara ulaşılabilecektir.



Şekil 3.8. Kısa zamanlı Fourier dönüşümü (Türkmenoğlu, 2006).

Dalgacık dönüşümü bu noktada devreye girmektedir. Dönüşüm, değişik boyutta bölgelerde çözümleme yöntemidir. Sinyal, ana dalgacığın ölçeklendirilmiş ve

kaydırılmış versiyonları olan kendi dalgacıklarına ayrılır. Dalgacık dönüşümü ile uzun zaman aralıklarında düşük frekans bilgileri, kısa zaman aralıklarında da yüksek frekans bilgileri daha doğru çözümlenebilmektedir.



Şekil 3.9. Dalgacık dönüşümü (Türkmenoğlu, 2006).

Dalgacıklar, bir ana dalgacıktan (Ψ), ölçekleme (s) ve kaydırma (τ) parametrelerinin değiştirilmesiyle oluşturulan, belirli bir zamanda ortalaması sıfır olan fonksiyonlardır. Bu fonksiyonların matematiksel ifadesi E. 3.5’de verilmiştir (Arı vd., 2008).

$$\psi_{s,\tau}(t) = \frac{1}{\sqrt{s}} \cdot \psi\left(\frac{t - \tau}{s}\right) \quad (\text{E. 3.5})$$

Düzgün ve sonlu uzunluğu olan sinüsler ile karşılaştırılınca, dalgacıkların düzensiz bir şekli ve sınırlı bir yapısı vardır. Bu düzensiz şekil ve sınırlı yapı, durağan olmayan dalgaları analiz etmek için daha uygundur. Herhangi bir işaretin Fourier ve dalgacık dönüşümleri sırasıyla Şekil 3.10 ve Şekil 3.11’de örneklenmiştir.



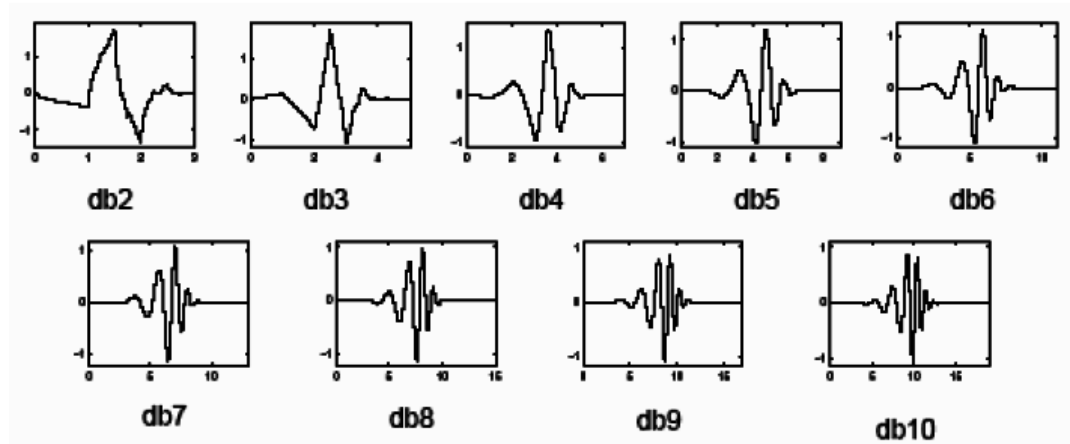
Şekil 3.10. Bir işaretin Fourier dönüşümü (Türkmenoğlu, 2006).



Şekil 3.11. Bir işaretin dalgacık dönüşümü (Türkmenoğlu, 2006).

Dalgacık dönüşümünde sıklıkla kullanılan dalgacık türleri aşağıda maddeler halinde verilmiş ve bu çalışmada kullanılan 'db2' dalgacığının yer aldığı Daubechies ailesi Şekil 3.12'de gösterilmiştir.

- Haar dalgacığı
- Daubechies dalgacıkları
- Coiflets dalgacıkları
- Morlet dalgacığı
- Meyer dalgacığı
- Symlets dalgacıkları
- Mexican Hat dalgacığı



Şekil 3.12. Daubechies dalgacık ailesi (Türkmenoğlu, 2006).

3.3.1. Sürekli dalgacık dönüşümü (SDD)

SDD, ana dalgacığın (Ψ) ölçeklendirilmiş (s) ve kaydırılmış (τ) dalgacık fonksiyonları ile analiz edilecek sinyalin ($f(t)$) katlanması ile sağlanmaktadır. Dönüşüm sonunda, işaretin zaman-frekans gösterimi, hepsi de farklı çözünürlüklü olan işaretler olacaktır (Yılmaz, 2005). Sürekli dalgacık dönüşümünün matematiksel ifadesi E. 3.6'da verilmiştir (Arı vd., 2008).

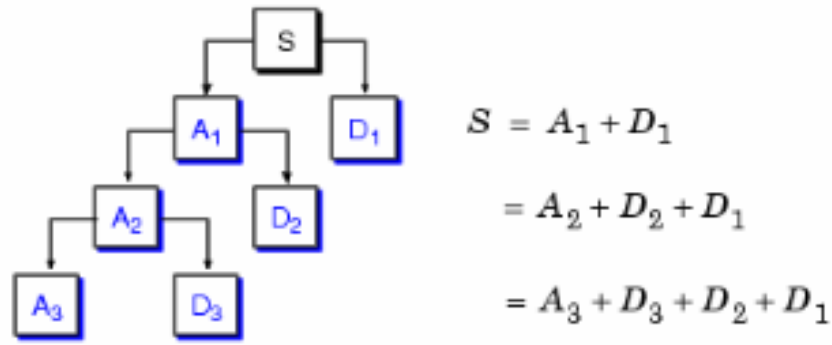
$$SDD_{s,\tau}(t) = \frac{1}{\sqrt{s}} \cdot \int_{-\infty}^{\infty} f(t) \cdot \Psi\left(\frac{t-\tau}{s}\right) \cdot dt \quad (E.3.6)$$

3.3.2. Ayrık dalgacık dönüşümü (ADD)

ADD, sinyali alçak frekans ve yüksek frekans bileşenlerine ayıran bir filtre bankasıdır. Dönüşüm, örnek sayısı her seviyede yarıya düşürülerek istenilen seviyeye kadar devam eder. Birinci seviye dönüşümde sinyal alçak frekans bileşenleri için yaklaşım (cA) ve yüksek frekans bileşenleri için detay (cD) olarak adlandırılan katsayılarına ayrılır. Sonraki seviyelerde yaklaşım katsayıları bileşenlerine ayrılarak dönüşüm devam eder. Yaklaşım ve detay katsayıları sırasıyla E. 3.7 ve E. 3.8 kullanılarak hesaplanır. ADD'nin daha iyi anlaşılabilmesi adına üçüncü seviye ADD Şekil 3.13'de gösterilmiştir.

$$cA_j = \sum_{n=-\infty}^{+\infty} f(n) \cdot \varphi_{j,k}(n) = \sum_{n=-\infty}^{+\infty} f(n) \cdot \frac{1}{\sqrt{2^j}} \cdot \varphi\left(\frac{n-k2^j}{2^j}\right) \quad (E.3.7)$$

$$cD_j = \sum_{n=-\infty}^{+\infty} f(n) \cdot \Psi_{j,k}(n) = \sum_{n=-\infty}^{+\infty} f(n) \cdot \frac{1}{\sqrt{2^j}} \cdot \Psi\left(\frac{n-k2^j}{2^j}\right) \quad (E.3.8)$$



Şekil 3.13. Üçüncü seviye ayrık dalgacık dönüşümü (Yılmaz, 2005).

3.4. Görgül Kip Ayırışımı (GKA)

GKA, sinyali alçak (trend) ve yüksek (detay) frekans bileşenlerine ayrıştırmada dalgacık dönüşümüne alternatif olarak geliştirilmiştir (Rilling vd., 2003). Ayırışım, durma kriterinin olması hariç dalgacık dönüşümü ile benzerlik göstermektedir. Durma kriteri olarak ayırışmış sinyale ait maksimum ve minimum noktaları kullanılmakta, ayırışmış sinyallerin maksimum ve minimum noktaları hesaplanamadığında dönüşümün durma kriteri sağlanmış olmaktadır. GKA, herhangi bir sabit analitik formülle ifade edilemeyen bir sinyal ayrıştırma yöntemidir. Ayırışım ile elde edilen yüksek frekans bileşenleri, sinyalin öz kip fonksiyonları (ÖKF) olarak adlandırılmaktadır.

GKA aşağıdaki adımlar izlenerek gerçekleştirilmektedir (Rilling vd., 2003).

- i. $x(t)$ sinyalindeki tüm yerel ekstremum (maksimum ve minimum) noktalar bulunur.
- ii. Bulunan üst ve alt ekstremum noktaları ara değerlendirme ile birleştirilerek bir üst zarf ($e_{\max}(t)$) ve bir alt zarf ($e_{\min}(t)$) elde edilir.
- iii. Alt ve üst zarfların ortalaması E. 3.9'da gösterildiği şekilde alınarak $m(t)$ ortalama sinyali elde edilir.

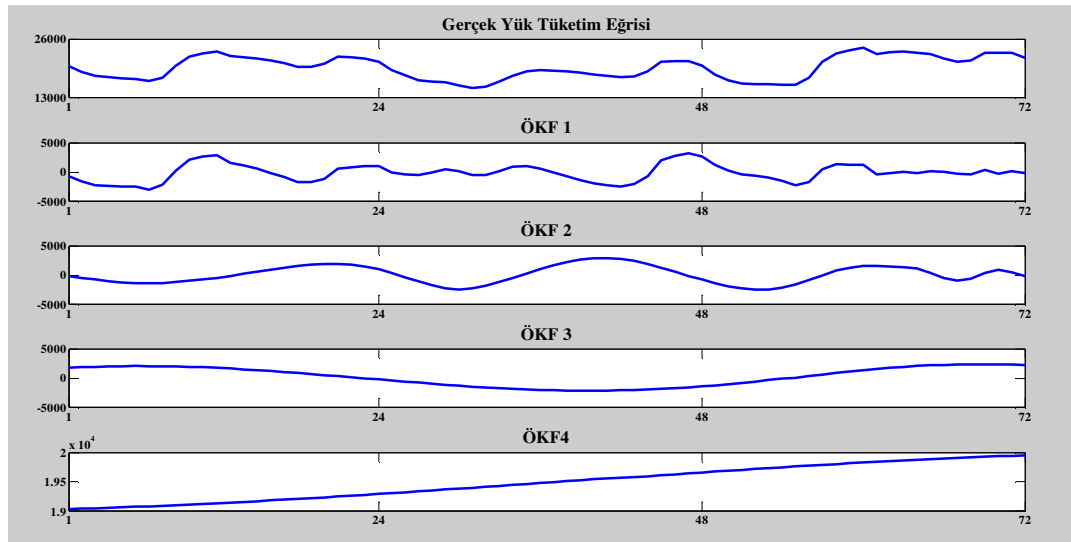
$$m(t) = \frac{e_{\min}(t) + e_{\max}(t)}{2} \quad (\text{E. 3.9})$$

iv. $d(t)$ detay sinyali E. 3.10'dan elde edilir.

$$d(t) = x(t) - m(t) \quad (\text{E. 3.10})$$

v. Durma kriteri sağlanana kadar $d(t)$ sinyaline ilk dört adım uygulanır.

GKA, 12 Mayıs 2009 gününe ait simülasyon veri kümesi için Şekil 3.14'de örneklenmiştir. Şekilde, bu günlere ait yük tüketim eğrisi ayrışımından elde edilen 4 ÖKF ile gösterilmiştir.

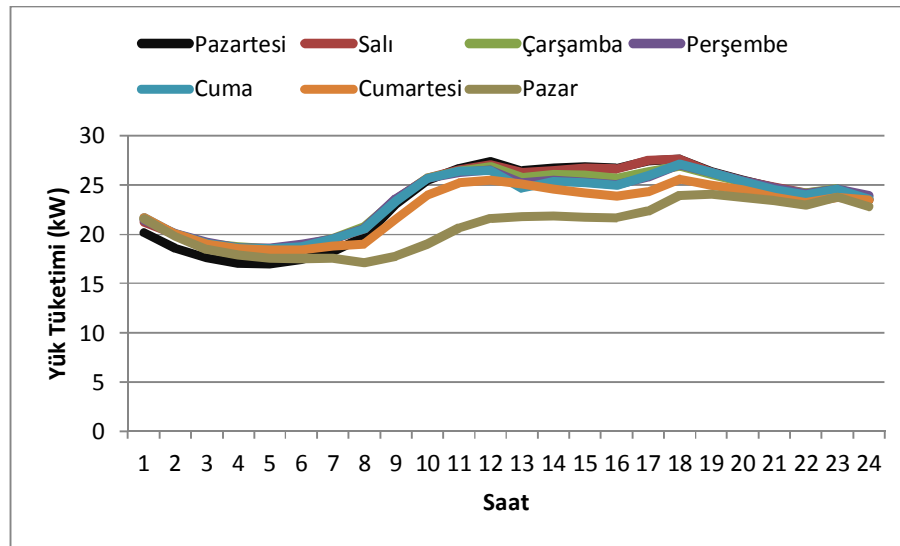


Şekil 3.14. Yük eğrisi için GKA örneklenmesi.

4. VERİ ANALİZİ

Yük yoğunluğu zamanla değişen bir karakteristik gösterir. Etkin bir yük tahmini için, yük karakteristiğinin zamanla bir başka deyimle günlük, haftalık, aylık ve yıllık olarak değişiminin iyi belirlenmiş olması gerekir.

Haftanın günlerine göre yük yoğunluğuna bakıldığında, hafta içi ve hafta sonu günlerinin farklı karakteristikte olduğu görülmektedir. Şekil 4.1'de 5-11 Ocak 2009 tarihleri saatlik yük verileri kullanılarak oluşturulmuş haftanın günlerine göre yük tüketimi grafiği gösterilmiştir.



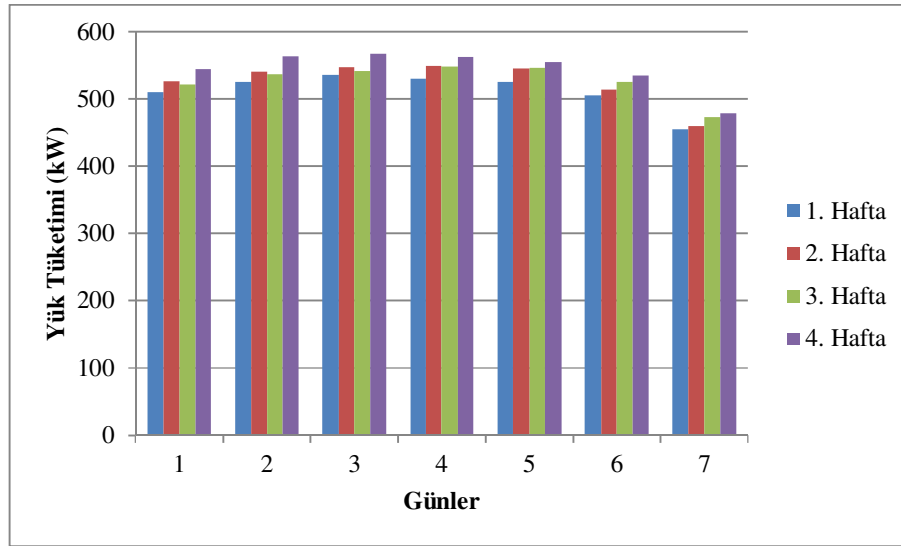
Şekil 4.1. Haftanın günlerine göre yük tüketimi (5-11 Ocak 2009).

Grafiğe göre, haftanın günlerini beş tipte incelemek gerekir:

- Pazartesi
- Salı, Çarşamba, Perşembe
- Cuma
- Cumartesi
- Pazar

Grafik, 5-11 Ocak 2009 tarihleri verileri örnek alınarak oluşturulmuştur, ancak yerel tatil günleri hariç diğer günler için de aynı sonuçlar geçerlidir. Yerel tatil günlerinde ise yük yoğunluğu bu sonuçlardan bağımsız bir karakter göstermektedir.

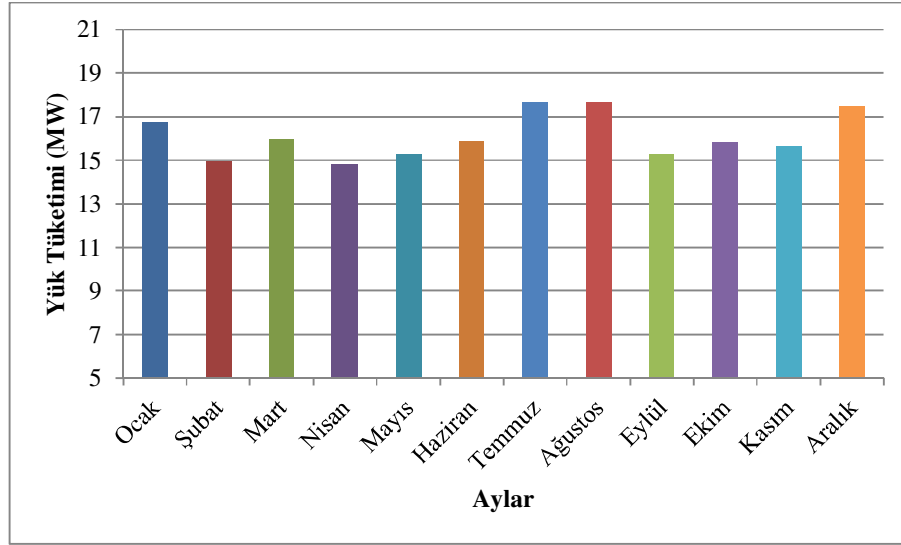
Yük talebinin mevsimsel olarak değiştiği bilinmektedir. Ancak, Şekil 4.2’de verilen 2009-Haziran ayı günlük puant yük tüketim değerlerine bakıldığında talep değişiminde haftalar arası da artış ya da azalış olduğu görülmektedir.



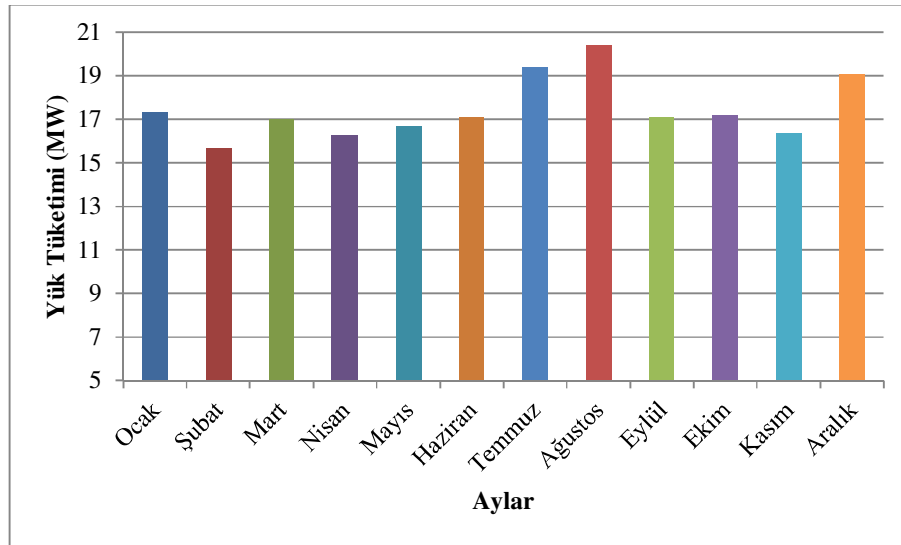
Şekil 4.2. Haziran ayı tüm haftaları için günlük puant yük tüketimi değerleri.

Şekil 4.3’te 2009 yılının ve Şekil 4.4’de 2010 yılının aylık toplam yük tüketimleri gösterilmiştir. Her iki yılın grafiğinden de görülmektedir ki yük tüketimi aydan aya değişmektedir.

Ayrıca, grafiklerden 2010 yılının aylık tüketimlerinin 2009 yılının aylık tüketimlerine göre daha fazla olduğu da açıkça görülmektedir. 2009 yılında yaşanan ekonomik kriz nedeniyle yük tüketiminde tasarrufa gidildiği ve 2009-2010 yılları arasında artan nüfus göz önüne alındığında, iki yıl arasındaki tüketim farklılıkları değişen ekonomik ve demografik koşullarının yük tüketimine yansıdığını kanıtlamaktadır.

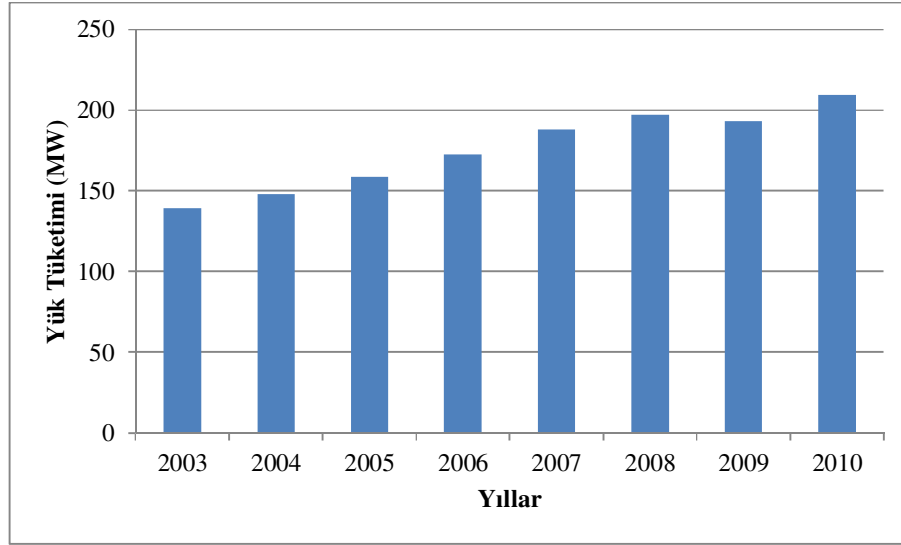


Şekil 4.3. 2009 yılı aylık yük tüketim değerleri.



Şekil 4.4. 2010 yılı aylık yük tüketim değerleri.

2002-2010 yılları arası yıllık elektrik enerjisi tüketim değişimleri Şekil 4.5’de grafiksel olarak ifade edilmiştir. Şekilden de görüldüğü üzere yük tüketimi her yıl artmaktadır. 2009 yılı yıllık yük tüketimindeki düşüş o yıl yaşanan ekonomik krizden kaynaklanmaktadır.



Şekil 4.5. Yıllık yük tüketim değerleri.

5. UYGULAMA

Bu çalışmada TEİAŞ'tan elde edilen Türkiye geneli 2009-2010 yılları saatlik yük verisi veriler kullanılarak 2009-2010 yılları Türkiye geneli 24 saatlik yük tahmini yapılmıştır.

Çalışmada öncelikle yerel tatil günlerinin bozucu etkisini ortadan kaldırmak amacı ile veri ön işleme yapılmıştır. İkincil olarak, geçmiş yük verilerinden bir veri kümesi oluşturulmuştur. Tahmin çalışması için oluşturulan yapılar bu veri kümesi ile simüle edilmiş ve tahmin hataları MAPE olarak hesaplanmıştır. Tahmin hata yüzdelerini düşürebilmek adına yük tahmininde regülasyona gidilmesi önerilmiş ve regüleli yük tahmini yüzdeleri de MAPE olarak hesaplanmıştır. MAPE değerleri, tüm yapılar için regüleli ve regülesiz olarak karşılaştırılmıştır.

Tüm bu adımlar aşağıda ayrıntıları ile açıklanmaktadır.

5.1. Veri Ön İşleme

Milli bayramlar, dini bayramlar ve yılbaşı tatilleri gibi yerel tatil günlerinde yük karakteristiği değişmekte ve bu değişiklik tahmini olumsuz yönde etkilemektedir. Bu nedenle, yerel tatil günlerinin verileri normal gün karakteristiğinde olacak şekilde düzeltilmiştir. Bu düzeltme, tatil günlerinin bir hafta öncesi ve bir hafta sonrası haftalık toplamaları temel alınarak yapılmıştır. Düzeltme formülasyonu E. 5.1'de verilmiştir:

$$(veri)'_{YT} = \frac{|\sum_{i=1}^7(veri)_{YT-i} + \sum_{i=1}^7(veri)_{YT+i}|}{2} \cdot (veri)_{YT} \quad (E. 5.1)$$

Eşitlikteki YT ve i terimleri sırasıyla yerel tatil günü ve gün sırasını ifade etmektedir. Tahmin çalışmasında değiştirilmiş yerel tatil gün verileri kullanılmış ancak bu günlere dair elde edilen tahmin sonuçları hata hesabına katılmamıştır.

5.2. Veri Kümesi Seçimi

Yük talebi, değişen meteorolojik, ekonomik, sosyolojik ve demografik durumlar karşısında değişkenlik gösterir. Bu nedenle, tahmin yapılırken bağımsız değişkenlerin (sıcaklık, nem, yağış gibi meteorolojik değişkenler, yerel tatil günlerini içeren sosyolojik değişkenler, gayri safi milli hâsıla, ithalat, ihracat gibi ekonomik değişkenler, nüfus) dikkate alınması gerekir.

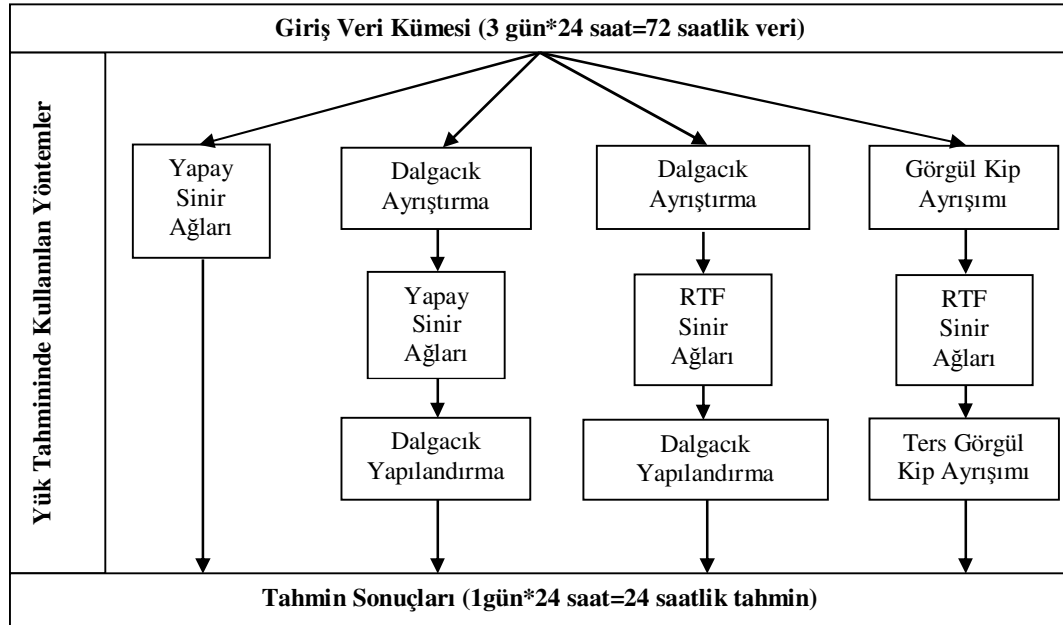
Sosyolojik koşullar olarak nitelendirilen yılbaşı tatilleri, milli ve dini bayram tatilleri gibi yerel tatil günlerine ait geçmiş yük tüketim verileri değiştirilerek normal gün karakteristiğine indirgenmiş ve bu günlere dair tahmin sonuçları hata hesabına katılmamıştır.

Literatürde KDYT çalışmalarında bağımsız değişken olarak sıcaklık verisi sıklıkla kullanılmaktadır. Ancak, bölgesel sıcaklık verisi temininin aksine ülke geneli çalışmalar için etkin bir sıcaklık verisi tanımlamak ve kullanmak oldukça zordur. Bu nedenle, çalışmada bağımsız değişken kullanılmamış, sadece geçmiş yük verisi kullanılmıştır.

5.3. Uygulama için Oluşturulan Yapılar

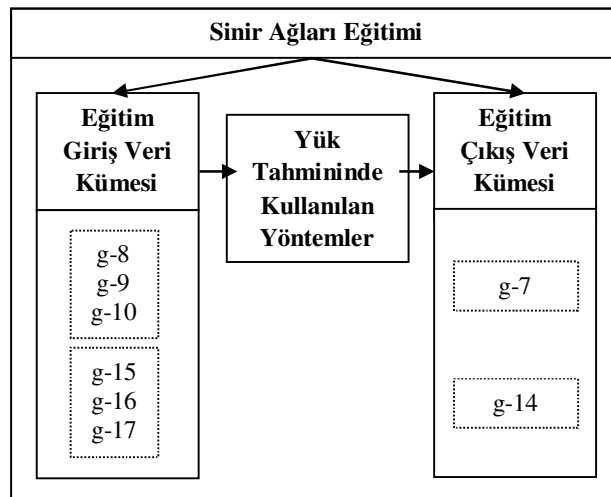
Tahmin çalışması için YSA, Dalgacık dönüşümü ve YSA, Dalgacık dönüşümü ve RTFSA, GKA ve RTFSA olmak üzere dört yapı oluşturulmuştur. Çalışmada MATLAB *neural networks toolbox*, *wavelet toolbox* ve *emd.m* (Rilling vd., 2003) alt programı kullanılmıştır.

Oluşturulan tüm yapılar için tahmin gününün öncesindeki 3 günün saatlik yük verileri giriş olarak kullanılmıştır. Giriş veri sayısı, farklı sayıda giriş verisi ile test edilmiş sinir ağlarından en az veri sayısı ile en düşük hata sonucunu veren giriş veri sayısının seçilmesi ile belirlenmiştir. Böylece, giriş için kullanılan günler ile tahmin günü meteorolojik, ekonomik ve demografik koşullar açısından yaklaşık olarak aynı karakteristiğe getirilmiştir. Çalışmanın blok diyagramı Şekil 5.1’de gösterilmiştir.



Şekil 5.1. KDYT blok diyagramı.

Veriler incelendiğinde bir günlük yük tüketiminin çoğunlukla öncesindeki iki haftanın yük tüketimi ile değiştiği öngörülebilir. Bu nedenle, sinir ağları eğitiminde giriş olarak tahmin günün öncesindeki 3 günün bir hafta öncesindeki ve iki hafta öncesindeki günlerin saatlik yük verileri, çıkış olarak da tahmin günün bir hafta öncesi ve iki hafta öncesindeki günlerin saatlik yük verileri kullanılmıştır. Sinir ağları eğitim şeması Şekil 5.2’de gösterilmiştir. Şekilde g terimi tahmin edilecek günü ifade etmektedir.

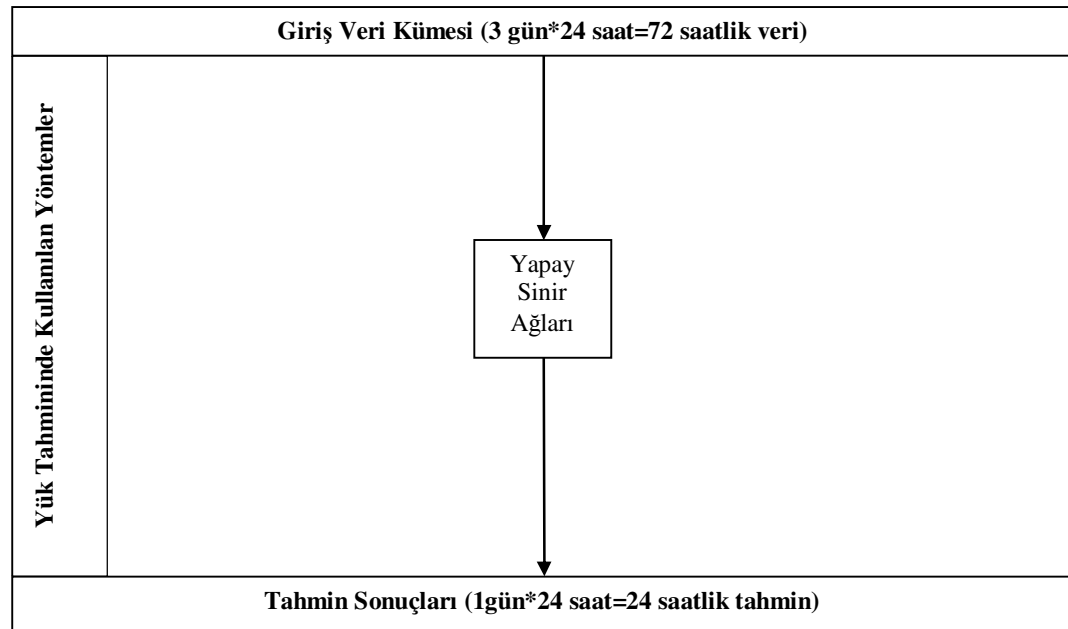


Şekil 5.2. Sinir ağları eğitim şeması.

YSA yapısı saatlik yük verisi ile eğitilmiştir. Dalgacık Dönüşümü ve YSA ile Dalgacık Dönüşümü ve RTF Sinir Ağları yapıları dalgacık ayrıştırma işleminden elde edilen katsayılar ile eğitilirken GKA ve RTF Sinir Ağları yapısı ise görgül kip ayrışımından elde edilen öz kip fonksiyonları ile eğitilmiştir. YSA eğitiminde nöronların ilk ağırlıkları rastgele belirlendiğinden, bu yapının her simülasyonda farklı sonuçlar vereceği unutulmamalıdır.

5.3.1. Yapı 1: Yapay sinir ağları (YSA)

Uygulama için tek gizli katmanlı, 0-hata amaçlı bir ileri beslemeli-geri yayımlı sinir ağı kullanılmıştır. Oluşturulan YSA yapısı ile Şekil 5.2’de belirtilen eğitim veri kümesi kullanılarak eğitim gerçekleştirilmiş ve giriş olarak verilen 72 saatlik yük verisine göre tahmin gününe dair 24 saatlik tahmin sonuçları elde edilmiştir. YSA yapısı blok diyagramı Şekil 5.3’de verilmiştir.



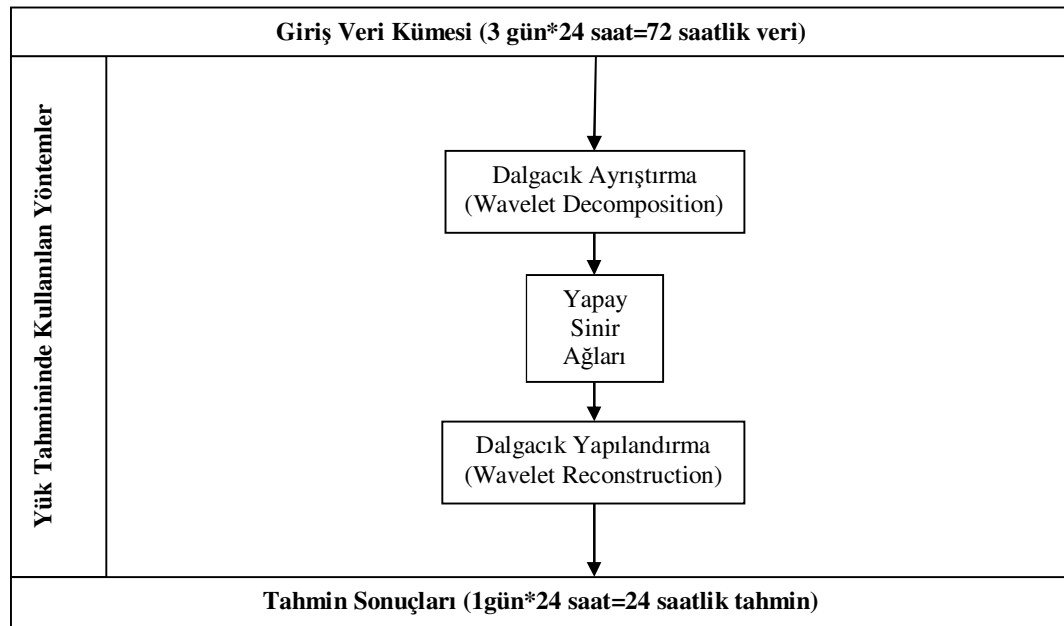
Şekil 5.3. YSA yapısı blok diyagramı.

Giriş ve çıkış katmanlarındaki nöron sayısı sırasıyla giriş ve çıkış sayısına eşittir. Gizli katman nöron sayısı deneme yanılma ile 10 olarak seçilmiştir. Sinir ağı “Levenberg-Marquardt” eğitim fonksiyonu ile eğitilmiştir. Gizli katman transfer

fonksiyonu “tansig” ve çıkış katmanı transfer fonksiyonu “purelin” olarak kullanılmıştır.

5.3.2. Yapı 2: Dalgacık Dönüşümü ve yapay sinir ağları (DD ve YSA)

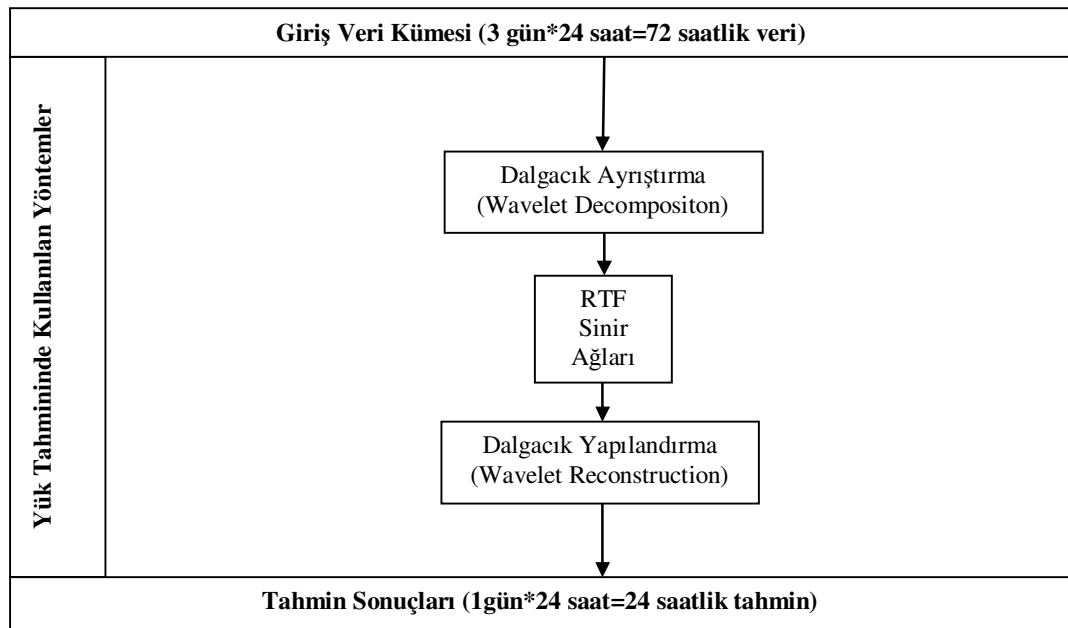
YSA yapısı fonksiyon yakınsama problemlerinde iyi performans göstermektedir (Soliman ve Al-Kandari, 2010). Ancak giriş verileri birbirine yakın değerlerde olduğunda yakınsama problemi zorlaşmaktadır. Dalgacık dönüşümü, zaman alanında verilen sinyali frekans bileşenlerine ayırarak giriş uzayını genişletir; böylece YSA daha iyi yakınsama yapabilir. Bu nedenle, daha iyi tahmin sonuçları elde edebilmek için aynı eğitim giriş veri kümesi “db2” dalgacığı ile beşinci seviyede alçak ve yüksek frekans bileşenlerine ayrılmış ve katsayıları elde edilmiştir. Oluşturulan YSA yapısı, bu katsayılarla eğitilmiş ve eğitim sonucunda elde edilen katsayılar “db2” dalgacığı ile yeniden yapılandırılarak test edilmiştir. Şekil 5.4’de belirtilen giriş veri kümesi kullanılarak yapı simüle edilmiş ve tahmin gününe dair 24 saatlik tahmin sonuçları elde edilmiştir.



Şekil 5.4. Dalgacık dönüşümü ve YSA yapısı blok diyagramı.

5.3.3. Yapı 3: Dalgacık dönüşümü ve RTF sinir ağları (DD ve RTFSA)

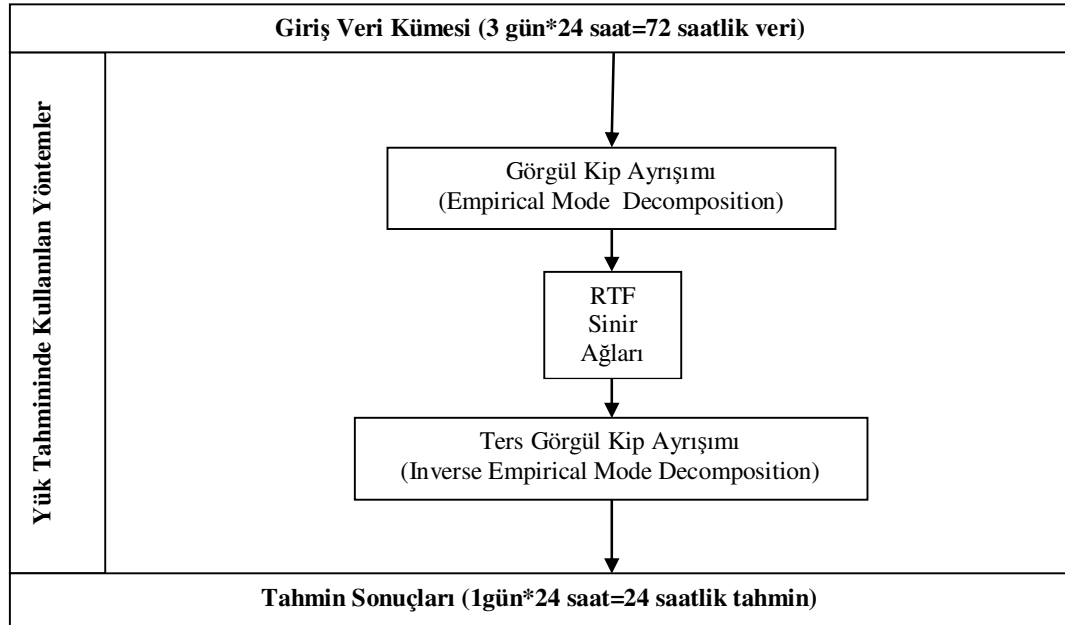
Durağan olmayan problemlerin çözümünde RTF sinir ağlarının YSA'ya göre daha doğru sonuçlar verdiği bilinmektedir (Soliman ve Al-Kandari, 2010). Bu nedenle, anlatılan dalgacık tekniği ile elde edilen alçak ve yüksek frekanslı katsayılar RTF sinir ağı eğitiminde kullanılmış ve eğitim sonuçları yine aynı dönüşümle yeniden yapılandırılmıştır. Aynı tahmin verileri kullanılarak 24 saatlik yük tahmini yapılmıştır. RTF sinir ağı yayılım sabiti 1 alınmıştır. Yapının blok diyagramı Şekil 5.5'te gösterilmiştir.



Şekil 5.5. Dalgacık dönüşümü ve RTF sinir ağları yapısı blok diyagramı.

5.3.4. Yapı 4: Görgül kip ayrışımı ve RTF sinir ağları (GKA ve RTFSA)

Bu yapıda RTF sinir ağları, görgül kip ayrışımı ile birlikte kullanılmıştır. İlk olarak, giriş verisi görgül kip ayrışımı ile öz kip fonksiyonlarına ayrıştırılır. Öz kip fonksiyonları ile eğitilen sinir ağına Şekil 5.6'da belirtilen giriş veri kümesi uygulanmış ve tahmin gününe dair 24 saatlik tahmin sonuçları elde edilmiştir.



Şekil 5.6. Görgül kip ayrışımı ve RTF sinir ağları yapısı blok diyagramı.

5.4. Regülasyon Hesabı

Literatürde yapılan çalışmalar yük talebinin mevsimsel olarak değiştiğini göstermektedir (Azadeh vd., 2009; Gontar ve Hatziargyriou, 2001; Osman vd., 2009; Pandey vd., 2010; Taylor ve McSharry, 2007; Chen vd., 2010). Bunun yanında, yük tüketimi haftalık olarak değişmektedir. Geçmiş yük verisi incelendiğinde, yük tüketiminin son iki haftanın haftalık yük tüketimlerinin belirlediği bir oranda değiştiği görülmektedir.

Bu nedenle çalışmada, daha düşük hata yüzdesine sahip tahmin sonuçları elde edebilmek adına yük tahmininde regülasyon önerilmiştir. Regüleli yük tahmini, regülesiz tahmin sonuçlarına haftalık yük tüketimi değişim oranı eklenerek gerçekleştirilmiştir.

Haftalık yük tüketimi (HT), haftalık yük tüketimi değişim oranı (ΔHT) ve regüleli yük tahmini (RYT) ifadeleri sırasıyla E. 5.2, E. 5.3 ve E. 5.4'de formülize edilmiştir:

$$HT = \sum_{g=1}^7 \sum_{s=1}^{24} ST_s^g \quad (\text{E. 5.2})$$

$$\Delta HT = \frac{BHHT}{IHHT} \quad (\text{E. 5.3})$$

$$RYT = RZYT + (\Delta HT - 1) \cdot RZYT \quad (\text{E. 5.4})$$

Eşitliklerdeki ST , g , s , $BHHT$, $IHHT$ ve $RZYT$ terimleri sırasıyla saatlik yük tüketimi, gün endeksi, saat endeksi, tahmin gününün bir hafta öncesi haftalık tüketimi, tahmin gününün iki hafta öncesi haftalık tüketimi ve regülesiz yük tahminini ifadelerini temsil etmektedir.

5.5. Hata Hesabı

Tahmin hataları MAPE olarak hesaplanmıştır. Yerel tatil günlerine ait tahmin sonuçları hata hesabına katılmamıştır. Oluşturulan her bir yapı için ortalama günlük MAPE ve günlük maksimum MAPE hesaplanmış ve karşılaştırılmıştır.

Günlük MAPE ve ortalama günlük MAPE sırasıyla E. 5.5 ve E. 5.6'da gösterildiği gibi hesaplanmıştır:

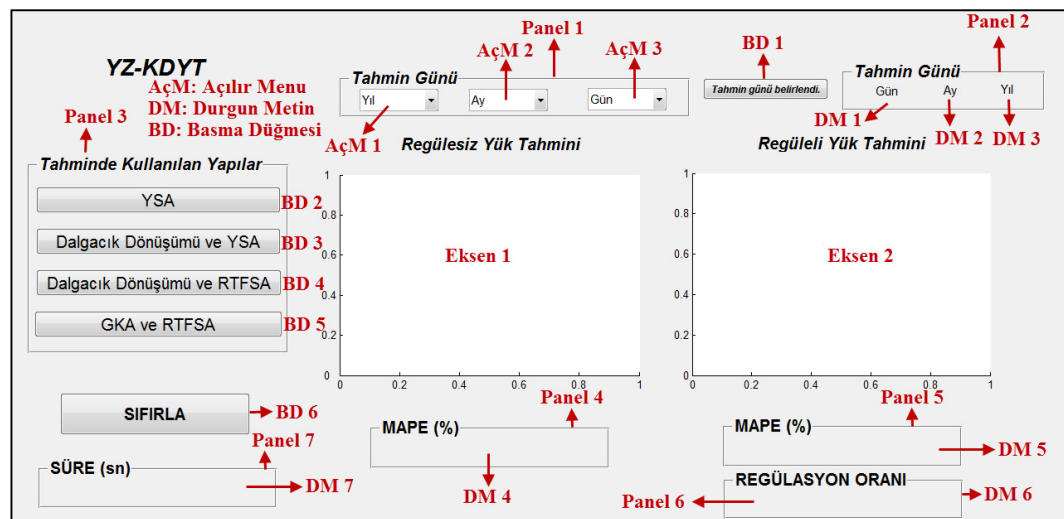
$$\text{Günlük MAPE} = \frac{1}{24} \cdot 100 \cdot \sum_{s=1}^{24} \frac{|(tahmin)_s - (gerçek)_s|}{(gerçek)_s} \quad (\text{E. 5.5})$$

$$\text{Ortalama Günlük MAPE} = \frac{1}{G} \cdot \sum_{i=1}^G (\text{Günlük MAPE}) \quad (\text{E. 5.6})$$

E. 5.6'daki G terimi normal gün (yerel tatil günleri dışındaki tüm günler) sayısını ifade etmektedir.

6. YZ-KDYT ARAYÜZÜ

Bu tez kapsamında, simülasyon çalışması için MATLAB *GUI* kullanılarak bir arayüz oluşturulmuştur. YZ-KDYT (Kısa Dönem Yük Tahmini için Yapay Zeka Modelleri) olarak adlandırılmış olan arayüz, kullanıcıya veri kümesi dahilinde istediği güne ait yük tüketimlerini oluşturulan yapılardan istediği ile tahmin edebilme olanağı tanımaktadır. Ayrıca bu arayüz, regüleli ve regülesiz tahmini yük eğrilerini gerçek yük eğrisi ile birlikte çizerek o güne ait MAPE değerlerini yazmaktadır. YZ-KDYT arayüzü Şekil 6.1’de gösterilmiştir.



Şekil 6.1. YZ-KDYT arayüzü.

Çalışmada kullanılan 1 Aralık 2008-31 Aralık 2010 tarihleri arası saatlik geçmiş yük tüketim verileri ara yüze yüklenmiştir. Bu verileri kullanarak yapılan çalışma, temel olarak arayüzde tasarlanan yedi panel, iki eksen ve altı basma düğmesi objeleri kullanılarak gerçekleştirilmiştir. Bu objeler, karşılıkları ve kullanım amaçları ile birlikte Çizelge 6.1’de verilmiştir.

Arayüz, yanlış girilen tarihlere karşı duyarlıdır. Örneğin, herhangi bir yıl için 31 Nisan günü seçilmek istendiğinde, ara yüz hata vererek Nisan ayının 30 günden oluştuğunu söylemektedir. Ayrıca, çalışmada yerel tatil günlerine yönelik tahminler hata

hesabına katılmamıştır. Kullanıcı, bu tatil günlerine denk gelen bir günü seçtiğinde, ara yüz o güne dair tahmin sonuçlarını göstermekte ancak tatil gününe dair sonuç olduğunu söyleyerek uyarıda bulunmaktadır.

Çizelge 6.1. Ara yüzde kullanılan objeler, karşılıkları ve kullanım amaçları.

Kullanılan Objeler		Karşılığı	Kullanım Amacı
Panel 1	Açılır Menü 1	Yıl	Tahmin edilecek gün yıl, ay, gün sıralaması ile girilir.
	Açılır Menü 2	Ay	
	Açılır Menü 3	Gün	
Basma Düğmesi 1		Onaylama	Girilen tarih onaylanır.
Panel 2	Durgun Metin 1	Gün	Onaylanan tarih gün, ay, yıl sıralaması ile yazdırılır.
	Durgun Metin 2	Ay	
	Durgun Metin 3	Yıl	
Panel 3	Basma Düğmesi 2	YSA	Tahmin çalışmasında kullanılması istenen yapı seçilir.
	Basma Düğmesi 3	Dalgacık Dönüşümü ve YSA	
	Basma Düğmesi 4	Dalgacık Dönüşümü ve RTF Sinir Ağları	
	Basma Düğmesi 5	GKA ve RTF Sinir Ağları	
Eksen 1		Regülesiz Yük Tahmini	Seçilen yapı ile elde edilen regülesiz tahmini yük talebi eğrisi tahmin gününe dair gerçek yük eğrisi ile çizdirilir.
Panel 4	Durgun Metin 4	MAPE	Regülesiz yük tahmini MAPE değeri yazdırılır.
Eksen 2		Regüleli Yük Tahmini	Seçilen yapı ile elde edilen regüleli tahmini yük talebi eğrisi tahmin gününe dair gerçek yük eğrisi ile çizdirilir.
Panel 5	Durgun Metin 5	MAPE	Regüleli yük tahmini MAPE değeri yazdırılır.
Basma Düğmesi 6		Sıfırlama	Yapılan işlemleri silerek ara yüzü yeni işlemler için hazır hale getirir.
Panel 6	Durgun Metin 6	Regülasyon Oranı	Haftalık yük tüketimi değişim oranını gösterir.
Panel 7	Durgun Metin 7	Süre	Yapılan işlemin gerçekleşme süresini saniye cinsinden gösterir.

Oluřturulan arayüz, önceden yüklenmiş verileri kullanarak çalışmaktadır. İstendiđi takdirde veri yükleme işlemini kullanıcıya bırakmak da mümkündür.

7. UYGULAMA SONUÇLARI

Bu çalışmada TEİAŞ'tan elde edilen Türkiye geneli 1 Aralık 2008-31 Aralık 2010 arası saatlik yük verisi veriler kullanılarak 2009-2010 yılları Türkiye geneli 24 saatlik yük tahmini yapılmıştır. Oluşturulan yapıların uygulanmasında MATLAB *neural networks toolbox*, *wavelet toolbox* ve *emd.m* (Rilling vd., 2003) alt programı kullanılmıştır. Simülasyon çalışması, MATLAB *GUI* ile oluşturulmuş YZ-KDYT arayüzü ile yapılmıştır. Oluşturulan yapılar aşağıda maddeler halinde verilmiştir.

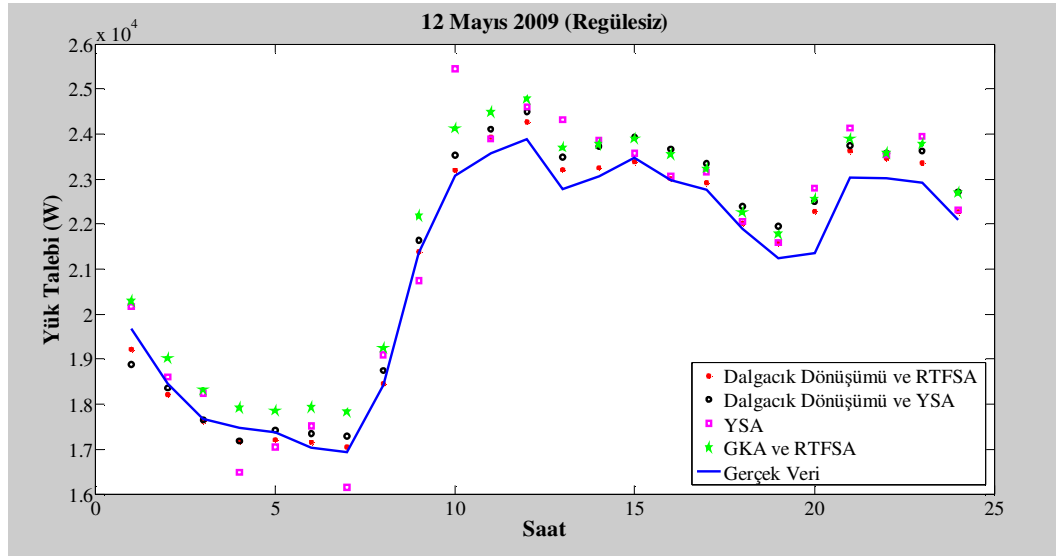
- Yapı 1: YSA
- Yapı 2: DD ve YSA
- Yapı 3: DD ve RTFSA
- Yapı 4: GKA ve RTFSA

Çalışmada kullanılan geçmiş yük verileri arayüze yüklenmiştir. Bu veriler arasından tahmini yapılmak istenen gün ve uygulanmak istenen yapı YZ-KDYT'de seçilmiş ve tahmin sonuçları elde edilmiştir. Arayüzde, regülesiz ve regüleli tahmin sonuçları için tahmin-gerçek yük eğrileri MAPE değerleri ile gösterilmiştir.

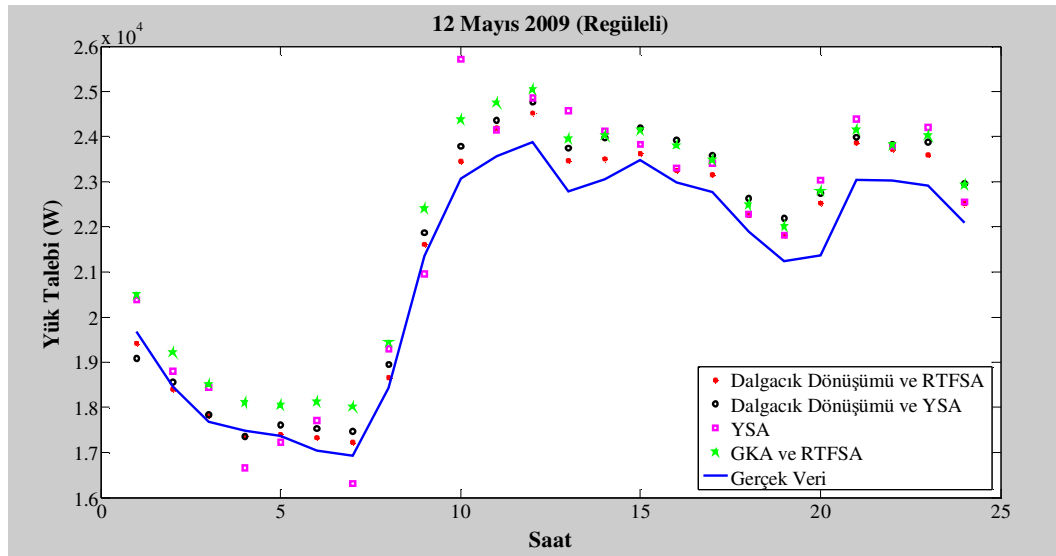
Tahmin hata yüzdeleri verileri değiştirilmiş yerel tatil günlerine ait tahmin sonuçları dikkate alınmadan ortalama günlük MAPE ve maksimum günlük MAPE olarak hesaplanmıştır. Çizelge 7.1'de örnek olarak seçilmiş günlerin tüm yapılara dair MAPE değerleri verilmiştir. Çizelgede koyu olarak belirtilmiş değerler, örnek günlere ait en düşük hata yüzdelerini veren yapıları göstermektedir. 12 Mayıs 2009 günü gerçek yük eğrisi ile bu gün için tüm yapılar ile elde edilen regülesiz ve regüleli tahmin eğrileri sırasıyla Şekil 7.1 ve Şekil 7.2'de gösterilmiştir.

Çizelge 7.1. Örnek günlerin günlük ortalama MAPE değerleri.

Ortalama Günlük MAPE değerleri	Regülesiz				Regüleli			
	YSA	DD ve YSA	DD ve RTFSA	GKA ve RTFSA	YSA	DD ve YSA	DD ve RTFSA	GKA ve RTFSA
12 Mayıs 2009	3.1356	3.3674	1.2423	3.4051	3.9121	3.9713	1.9122	4.5047
1 Temmuz 2009	3.2527	4.5922	0.8747	5.0412	2.6880	3.8910	1.0031	3.4613
11 Ekim 2010	3.6648	3.9231	1.8788	3.9716	3.8805	3.5754	3.1951	2.8139
27 Aralık 2010	3.4826	4.0092	1.4423	5.0228	2.8381	3.0562	2.2300	2.8884



Şekil 7.1. 12 Mayıs 2009 günü regülesiz tahmin eğrileri ve gerçek yük eğrisi.



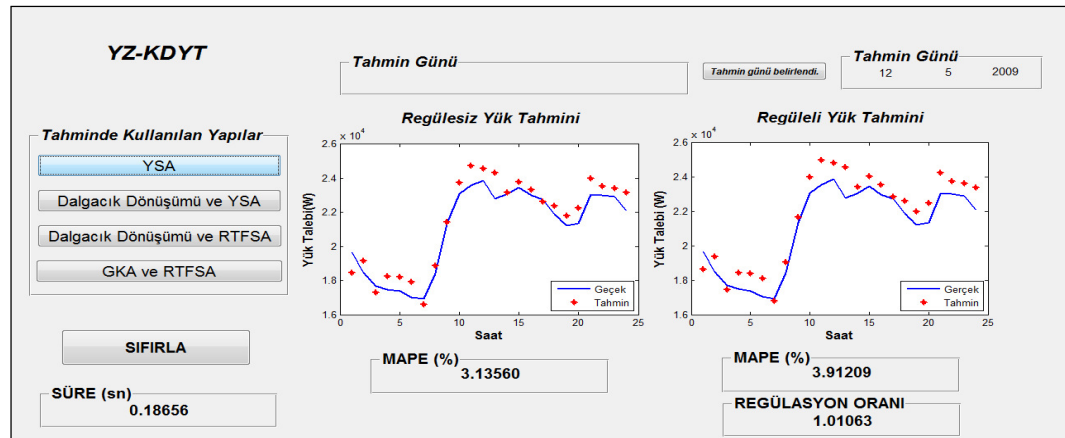
Şekil 7.2. 12 Mayıs 2009 günü regüleli tahmin eğrileri ve gerçek yük eğrisi.

Oluşturulan arayüz 12 Mayıs 2009 günü için örneklenmiştir. Söz konusu gün için ağ eğitiminde ve simülasyonda kullanılan veriler Çizelge 7.2’de verilmiştir.

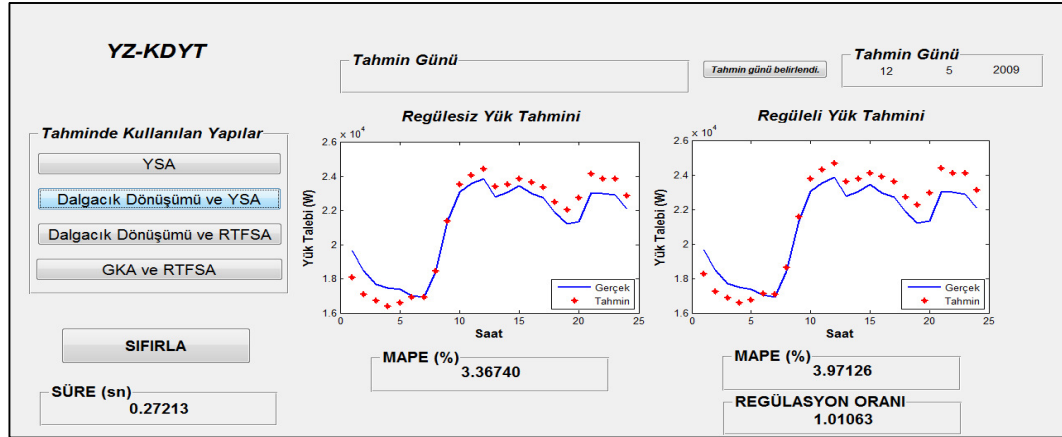
Çizelge 7.2. 12 Mayıs 2009 gününün tahmini için kullanılan veri kümesi.

Ağ Eğitimi				Simülasyon				
Giriş Veri Kümesi		Çıkış Veri Kümesi		Giriş Veri Kümesi		Hedef		
g-8	4 Mayıs 2009	g-7	5 Mayıs 2009	g-1	11 Mayıs 2009	g	12 Mayıs 2009	
g-9	3 Mayıs 2009			g-2	10 Mayıs 2009			
g-10	2 Mayıs 2009				g-3			
g-15	27 Nisan 2009	g-14	28 Nisan 2009	g-3	9 Mayıs 2009			
g-16	26 Nisan 2009							
g-17	25 Nisan 2009							

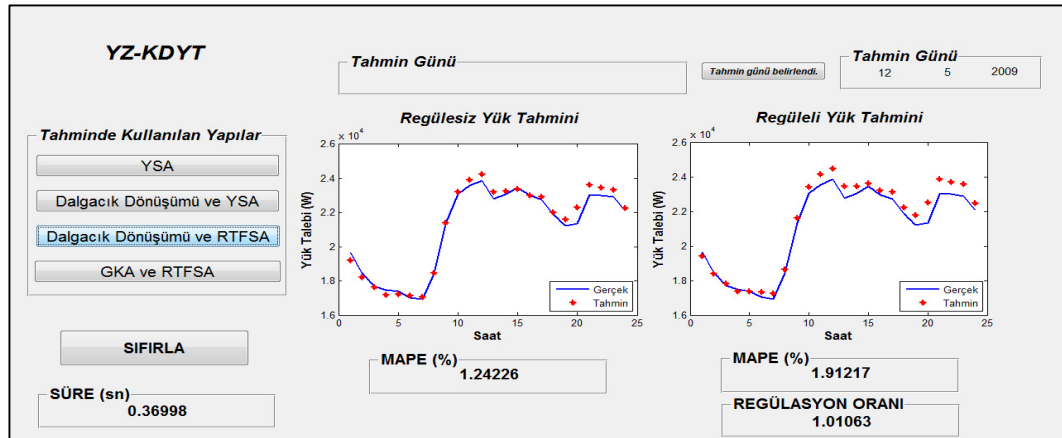
Bu veriler kullanılarak YSA, Dalgacık Dönüşümü ve YSA, Dalgacık Dönüşümü ve RTF Sınır Ağları, Görgül Kip Ayrışımı ve RTF Sınır Ağları yapıları ile arayüzden elde edilmiş regüleli ve regülesiz tahmin sonuçları sırasıyla Şekil 7.3, Şekil 7.4, Şekil 7.5 ve Şekil 7.6’da gösterilmiştir.



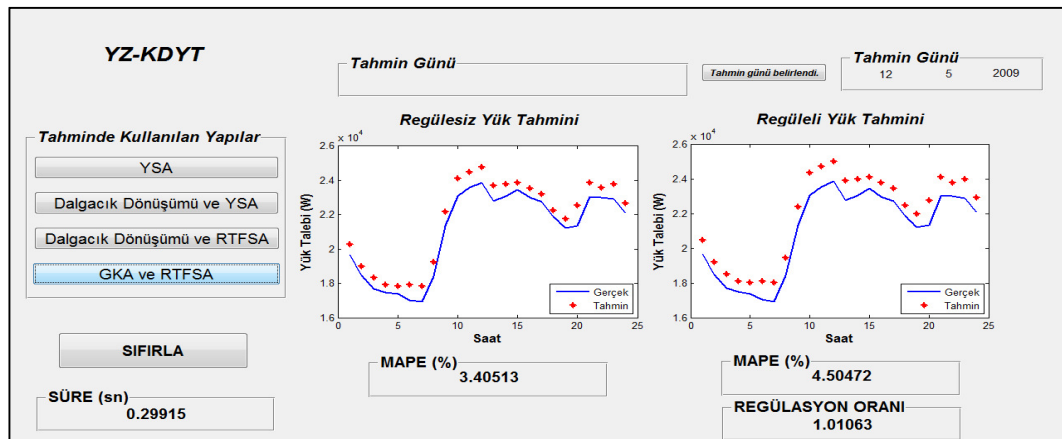
Şekil 7.3. YZ-KDYT arayüzü, YSA yapısı tahmin sonuçları (12.05.2009-Salı).



Şekil 7.4. YZ-KDYT arayüzü, dalgacık dönüşümü ve YSA yapısı tahmin sonuçları (12.05.2009-Salı).



Şekil 7.5. YZ-KDYT arayüzü, dalgacık dönüşümü ve RTF sinir ağı yapıları tahmin sonuçları (12.05.2009-Salı).



Şekil 7.6. YZ-KDYT arayüzü, GKA ve RTF sinir ağı yapıları tahmin sonuçları (12.05.2009-Salı).

Bu güne dair saatlik gerçek yük verileri ve oluşturulan yapılardan elde edilen saatlik tahmin sonuçları Çizelge 7.3'te her yapıya ait koyu yazılmış ortalama günlük MAPE değerleri ile verilmiştir. En düşük ortalama günlük hata yüzdeleri DD ve RFTSA yapısından elde edilmiş ve çizelgede altı çizili olarak belirtilmiştir. Çizelgedeki RZYT ve RYT terimleri sırasıyla regülesiz ve regüleli tahminleri ifade etmektedir.

Çizelge 7.3. 12 Mayıs 2009 günü saatlik tahmin sonuçları.

Saat	Gerçek Veri	YSA		DD ve YSA		DD ve RTFSA		GKA ve RTFSA	
		RZYT	RYT	RZYT	RYT	RZYT	RYT	RZYT	RYT
01:00	19677	20167	20381	18884	19084	19213	20381	20287	20503
02:00	18463	18601	18799	18364	18559	18212	18799	19015	19217
03:00	17682	18246	18440	17648	17836	17621	18440	18319	18514
04:00	17477	16488	16663	17179	17361	17170	16663	17915	18106
05:00	17371	17048	17229	17423	17608	17210	17229	17859	18049
06:00	17032	17517	17703	17351	17535	17152	17703	17936	18127
07:00	16927	16150	16321	17287	17471	17049	16321	17825	18015
08:00	18426	19097	19300	18747	18946	18456	19300	19241	19446
09:00	21346	20743	20963	21637	21867	21393	20963	22174	22410
10:00	23075	25444	25715	23533	23783	23202	25715	24116	24372
11:00	23567	23893	24147	24110	24366	23932	24147	24487	24747
12:00	23878	24598	24859	24496	24756	24263	24859	24779	25042
13:00	22783	24319	24577	23491	23740	23213	24577	23689	23941
14:00	23068	23860	24114	23726	23978	23252	24113	23772	24025
15:00	23473	23569	23820	23930	24184	23379	23820	23890	24144
16:00	22987	23059	23304	23671	23922	23006	23304	23548	23798
17:00	22761	23161	23407	23339	23588	22914	23407	23226	23473
18:00	21892	22051	22286	22391	22629	22023	22286	22252	22489
19:00	21238	21593	21822	21952	22186	21582	21822	21776	22008
20:00	21366	22793	23035	22497	22736	22283	23035	22554	22794
21:00	23041	24133	24390	23738	23991	23620	24390	23892	24146
22:00	23015	23537	23788	23573	23823	23468	237881	23576	23827
23:00	22911	23949	24204	23624	23875	23350	24204	23774	24027
24:00	22099	22316	22553	227140	22955	22276	22553	22693	22934
MAPE:		3.1356	3.9121	3.3674	3.9713	<u>1.2423</u>	<u>1.9122</u>	3.4051	4.5047

Örneklendirilen tahmin çalışması 2009-2010 yılları tüm günleri için gerçekleştirilmiş ve ortalama günlük MAPE ve maksimum günlük MAPE değerleri karşılaştırılmıştır.

Çizelge 7.4’de 2009 yılının tüm yapılar için regüleli-regülesiz ortalama günlük ve maksimum günlük MAPE değerleri, Çizelge 7.5’de 2009 yılı tüm aylarının ortalama MAPE değerleri verilmiştir. Çizelgelerde koyu olarak belirtilmiş değerler, 2009 yılına ait en düşük hata yüzdelerini veren yapıları göstermektedir.

Çizelge 7.4. 2009 yılı MAPE değerleri.

2009	YSA	DD ve YSA	DD ve RTFSA	GKA ve RTFSA
Ortalama Günlük MAPE (RZYT)	3.6735	3.7296	2.8955	3.5155
Maksimum Günlük MAPE (RZYT)	10.9531	15.7200	10.1784	12.4950
Ortalama Günlük MAPE (RYT)	3.3279	3.3459	2.9340	2.6686
Maksimum Günlük MAPE (RYT)	11.9370	20.4328	11.5731	12.7550

2009 yılı için en düşük ortalama günlük MAPE değerini regüleli tahmin çalışmasında Dalgacık dönüşümü ve RTF sinir ağıları, regülesiz tahmin çalışmasında ise GKA ve RTF sinir ağıları yapısı vermiştir. Regüleli yük tahmini hata yüzdeleri genel olarak regülesiz yük tahmini sonuçlarına göre daha düşük çıkarken Dalgacık dönüşümü ve RTF sinir ağıları ile GKA ve RTF sinir ağıları yapıları için bu durumun tam tersi geçerlidir.

Maksimum günlük MAPE değerlerine bakıldığında, her iki tahmin çalışmasında da en düşük hata yüzdesini yine Dalgacık dönüşümü ve RTF sinir ağıları yapısı vermiştir. Regülesiz yük tahmini hata yüzdeleri, regüleli yük tahmini sonuçlarına göre daha düşük çıkmıştır.

Çizelge 7.5 incelendiğinde, 2009 yılı için yapılan regülesiz ve regüleli yük tahmini çalışmalarında YSA yapısının kış aylarında, Dalgacık dönüşümü ve YSA yapısının sonbahar aylarında, Dalgacık dönüşümü ve RTF sinir ağıları yapısının kış

aylarında performanslarının maksimumunu gösterdikleri görülmektedir. GKA ve RTF sinir ağı yapıları ise regülesiz tahmin çalışmasında yaz aylarında, regüleli tahmin çalışmasında ise kış aylarında en iyi sonuçlarını vermiştir. Çizelgede, yapıların kendi performanslarının maksimumunu gösterdiği aylara ait MAPE değerleri koyu yazılmıştır.

Çizelge 7.5. 2009 yılı aylık MAPE değerleri.

2009	YSA		DD ve YSA		DD ve RTFSA		GKA ve RTFSA	
	RZYT	RYT	RZYT	RYT	RZYT	RYT	RZYT	RYT
Ocak	3.3379	2.9641	3.1957	3.0124	2.5274	2.6865	3.0507	2.1892
Şubat	3.6770	2.9847	3.5972	2.8712	2.3746	2.2567	3.7158	2.3689
Mart	4.7423	3.7310	4.9081	3.6235	4.1373	3.7404	4.8553	2.6435
Nisan	4.6394	4.0409	4.7506	4.1342	3.2540	3.3398	4.4484	3.2464
Mayıs	3.2373	3.4178	3.5705	3.5041	2.4336	2.6590	3.1234	3.3230
Haziran	3.0275	2.4207	3.5291	2.7798	2.1088	1.8790	3.3782	2.1365
Temmuz	3.4946	3.0233	3.1928	2.8913	2.7386	2.5633	3.4087	2.3759
Ağustos	3.0208	3.5495	2.9638	3.6405	2.9827	3.6107	1.9692	2.5624
Eylül	4.2172	3.2553	3.9803	3.3429	3.0668	2.8426	4.7994	2.8739
Ekim	2.5695	2.5704	2.8339	2.7387	1.7667	1.9189	2.6181	2.5126
Kasım	4.9775	4.8929	4.6926	4.6605	4.8569	5.0833	4.1567	3.7536
Aralık	3.7190	3.5579	3.9054	3.3679	3.1204	3.2538	3.2328	2.4379

Çizelge 7.6’de 2010 yılının tüm yapılar için regüleli-regülesiz ortalama günlük ve maksimum günlük MAPE değerleri, Çizelge 7.7’de 2010 yılı tüm aylarının ortalama MAPE değerleri verilmiştir. Çizelgelerde koyu olarak belirtilmiş değerler, 2010 yılına ait en düşük hata yüzdelelerini veren yapıları göstermektedir.

Çizelge 7.6. 2010 yılı MAPE değerleri.

2010	YSA	DD ve YSA	DD ve RTFSA	GKA ve RTFSA
Ortalama Günlük MAPE (RZYT)	3.8104	4.1859	2.9919	3.6346
Maksimum Günlük MAPE (RZYT)	9.5459	15.2228	9.7173	11.1660
Ortalama Günlük MAPE (RYT)	3.3983	3.5704	3.0351	2.6424
Maksimum Günlük MAPE (RYT)	9.1770	14.8118	13.9602	8.1421

2010 yılı için en düşük ortalama günlük MAPE değerini regülesiz tahmin çalışmasında Dalgacık dönüşümü ve RTF sinir ağı yapısı, regülesiz tahmin çalışmasında ise GKA ve RTF sinir ağı yapısı vermiştir. Regüleli yük tahmini hata yüzdeleri genel olarak regülesiz yük tahmini sonuçlarına göre daha düşük çıkarken Dalgacık dönüşümü ve RTF sinir ağı yapısı için bu durumun tam tersi geçerlidir.

Maksimum günlük MAPE değerlerine bakıldığında, regülesiz tahmin çalışmasında da en düşük hata yüzdesini YSA yapısı, regüleli tahmin çalışmasında ise GKA ve RTF sinir ağı yapısı vermiştir. Regüleli yük tahmini hata yüzdeleri genel olarak regülesiz yük tahmini sonuçlarına göre daha düşük çıkarken Dalgacık dönüşümü ve RTF sinir ağı yapısı için bu durumun tam tersi geçerlidir.

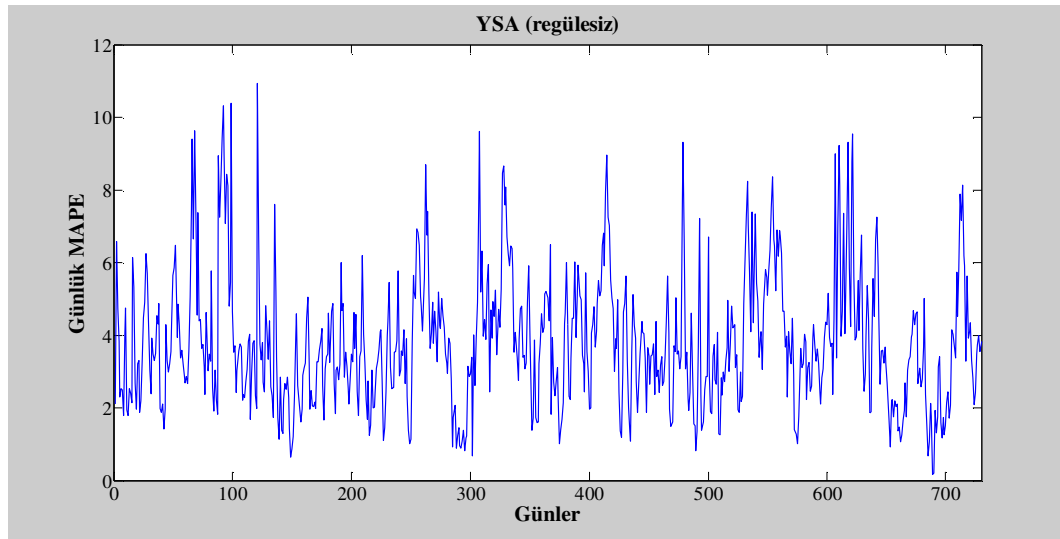
Çizelge 7.7. 2010 yılı aylık MAPE değerleri.

2010	YSA		DD ve YSA		DD ve RTFSA		GKA ve RTFSA	
	RZYT	RYT	RZYT	RYT	RZYT	RYT	RZYT	RYT
Ocak	3.5563	2.9942	3.1451	2.8677	2.6469	2.5106	3,5805	2,5499
Şubat	4.9644	4.5649	5.9383	5.0603	4.1183	4.3236	4,2568	3,1832
Mart	3.1914	3.6604	3.0524	3.6171	2.4461	3.1398	2,7169	3,1290
Nisan	3.5154	3.4259	4.3403	3.8259	2.9107	3.0911	2,9190	2,5385
Mayıs	2.8094	2.7004	2.8419	2.5046	2.3512	2.4327	2,5313	2,1733
Haziran	4.5172	4.0397	5.6180	4.3371	4.4881	4.5084	3,7150	2,6116
Temmuz	4.7278	3.6100	5.1472	3.9969	3.1205	2.9282	5,0266	2,5125
Ağustos	3.4385	2.8086	4.1199	3.1158	2.5455	2.3741	3,6463	2,3435
Eylül	5.1633	4.2784	5.1470	4.4400	3.3906	2.8519	5,6685	4,3653
Ekim	2.9916	2.6626	2.7729	2.2201	2.2527	2.3564	2,9064	2,1037
Kasım	2.9336	2.7209	3.4254	2.6496	2.4707	2.6338	3,0065	2,1881
Aralık	4.1035	3.3917	4.7372	4.2794	3.2016	3.2075	3,9640	2,3512

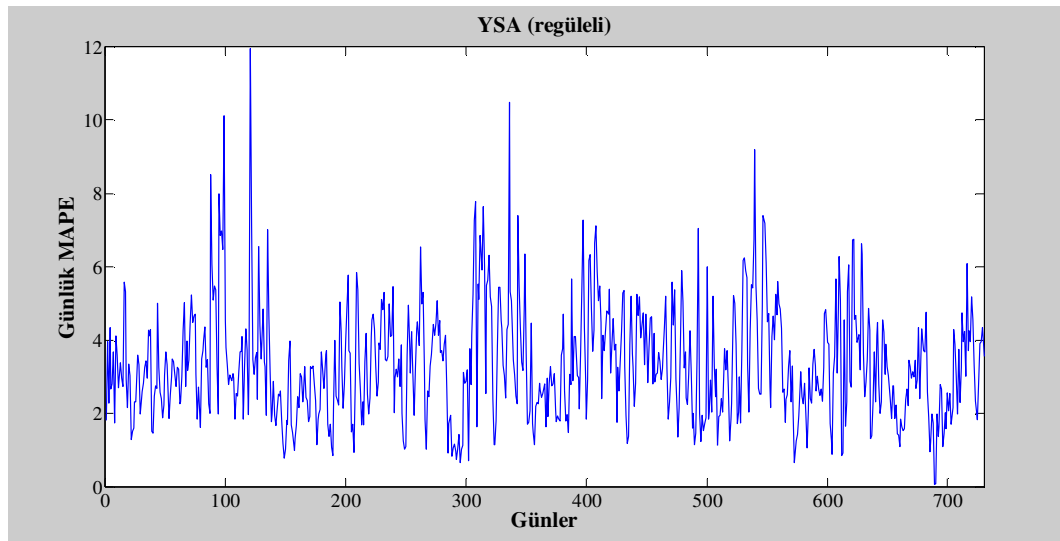
Çizelge 7.7 incelendiğinde, 2010 yılı için yapılan regülesiz ve regüleli yük tahmini çalışmalarında YSA yapısının sonbahar ve ilkbahar aylarında, Dalgacık dönüşümü ve YSA yapısının kış aylarında, Dalgacık dönüşümü ve RTF sinir ağı yapısının sonbahar ve ilkbahar aylarında kendi performanslarının maksimumunu

gösterdikleri görülmektedir. GKA ve RTF sinir ağı yapıları ise regülesiz tahmin çalışmasında ilkbahar aylarında, regüleli tahmin çalışmasında ise yaz aylarında en iyi sonuçlarını vermiştir. Çizelgede, yapıların kendi performanslarının maksimumunu gösterdiği aylara ait MAPE değerleri koyu yazılmıştır.

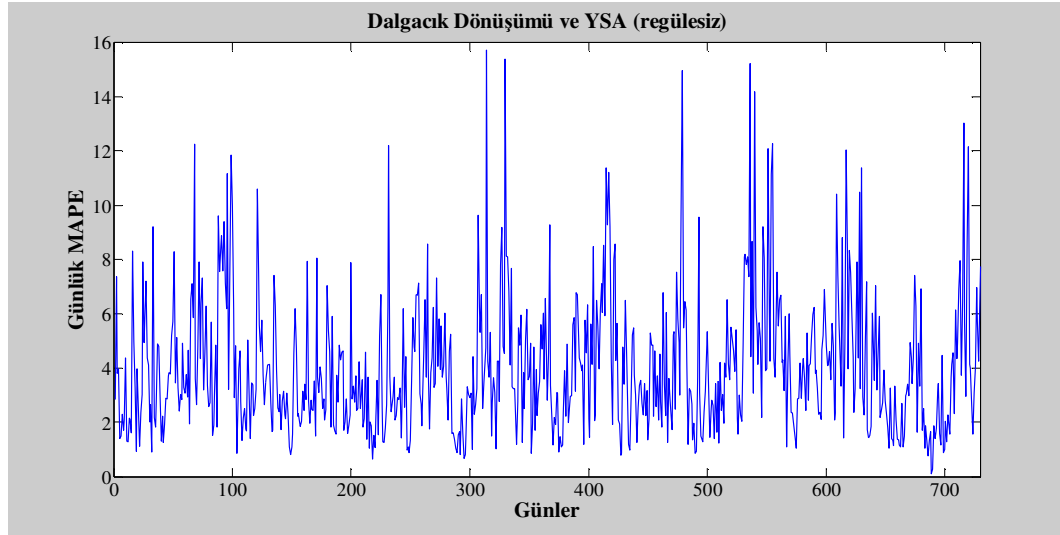
2009-2010 yılları tüm günleri (yerel tatil günleri dâhil) için elde edilen tahmin sonuçlarının hata yüzdelерinin grafikleri Şekil 7.7, Şekil 7.8, Şekil 7.9, Şekil 7.10, Şekil 7.11, Şekil 7.12, Şekil 7.13 ve Şekil 7.14’de gösterilmiştir.



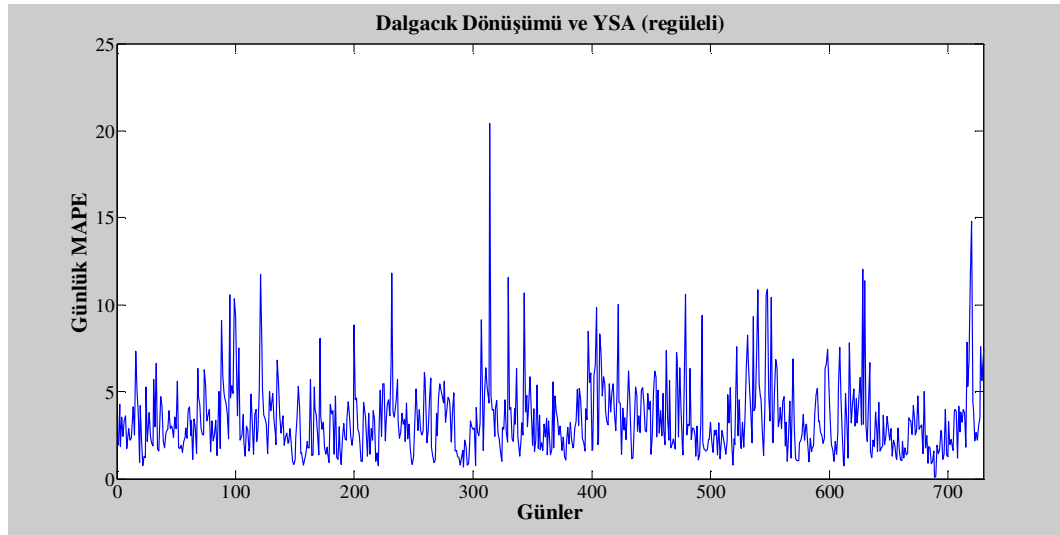
Şekil 7.7. YSA yapısı, 2009-2010 yılları regülesiz tahmin MAPE değerleri.



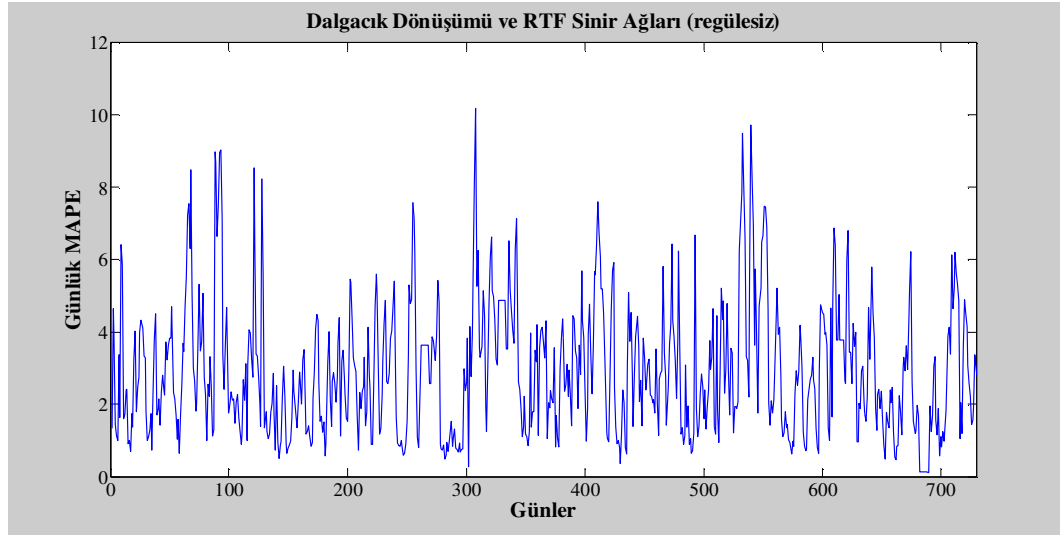
Şekil 7.8. YSA yapısı, 2009-2010 yılları regüleli tahmin MAPE değerleri.



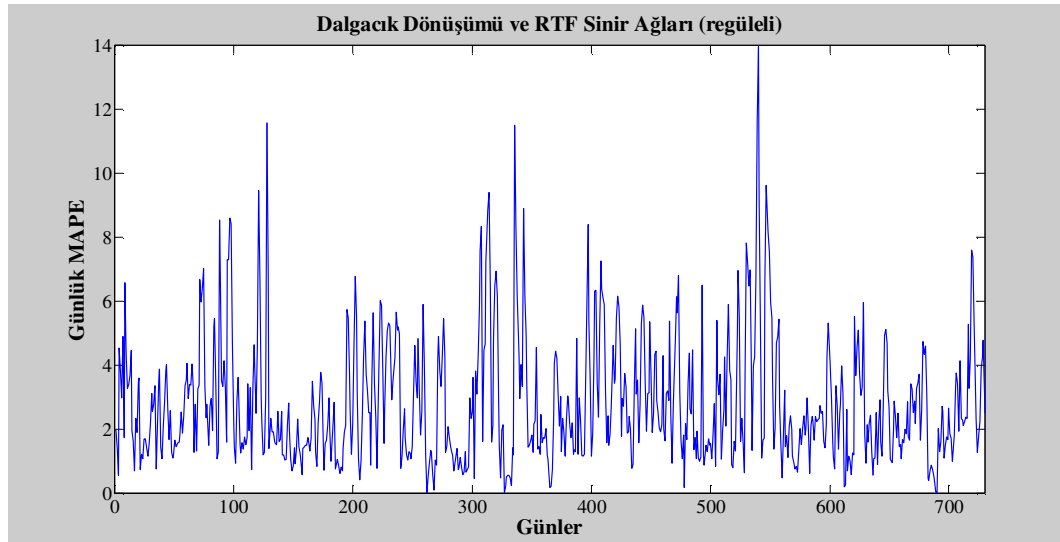
Şekil 7.9. Dalgacık Dönüşümü ve YSA yapısı, 2009-2010 yılları regülesiz tahmin MAPE değerleri.



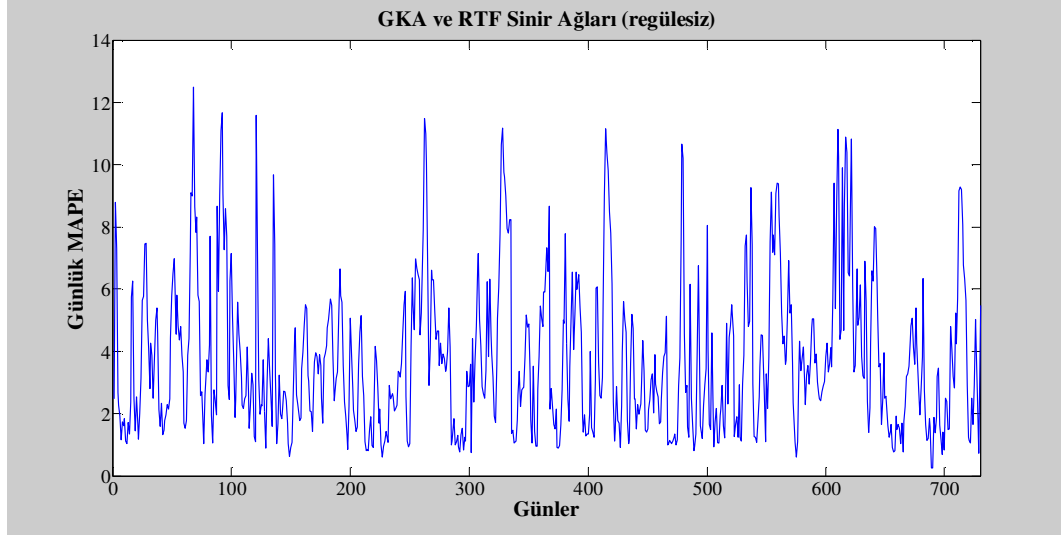
Şekil 7.10. Dalgacık Dönüşümü ve YSA yapısı, 2009-2010 yılları regüleli tahmin MAPE değerleri.



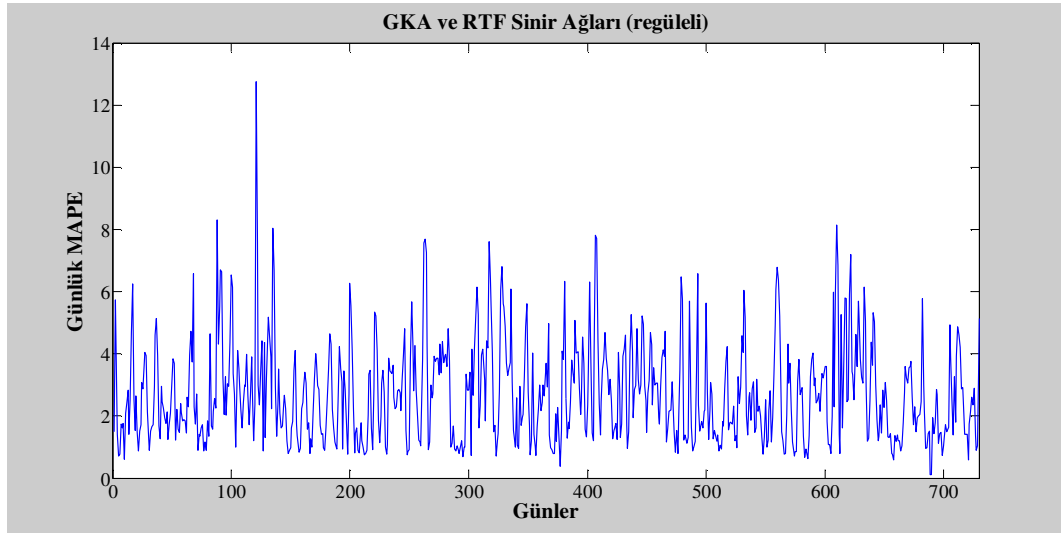
Şekil 7.11. Dalgacık Dönüşümü ve RTF sinir ağıları yapısı, 2009-2010 yılları regülesiz tahmin MAPE değerleri.



Şekil 7.12. Dalgacık Dönüşümü ve RTF sinir ağıları yapısı, 2009-2010 yılları regüleli tahmin MAPE değerleri.



Şekil 7.13. GKA ve RTF sinir ağları yapısı, 2009-2010 yılları regülesiz tahmin MAPE değerleri.



Şekil 7.14. GKA ve RTF sinir ağları yapısı, 2009-2010 yılları regüleli tahmin MAPE değerleri.

8. SONUÇLAR VE ÖNERİLER

Bu tez çalışmasında, Türkiye geneli için 24 saatlik yük tahmini amaçlanmış, sıcaklık ya da başka bir hava verisi kullanmadan kabul edilebilir hata yüzdelerinde gerçekleştirilebilen bir tahmin sistemi önerilmiştir.

Önerilen tahmin sistemi, YSA, Dalgacık Dönüşümü ve YSA, Dalgacık Dönüşümü ve RTF Sinir Ağları, Görgül Kip Ayrışımı ve RTF Sinir Ağları olmak üzere dört yapıdan oluşmaktadır. YSA ve RTF sinir ağları yapıları zaten iyi yaklaşımıcılar olarak bilinmektedir. Dalgacık dönüşümü ve görgül kip ayrışımı ise sinyali ayrıştırarak daha detaylı analiz yapabilme imkânı sağlamaktadır. Bu yapıların birlikte kullanılması ile daha iyi yakınsamalar yapılmıştır. Nöronların başlangıç ağırlıkları rastgele verildiğinden, YSA yapısının her simülasyon sonucunda farklı sonuçlar vermektedir.

En düşük ortalama günlük MAPE değerlerini Dalgacık dönüşümü ve RTF sinir ağları yapısı vermiştir. Ayrıca, hata yüzdelerine bakıldığında, diğer yapıların da kayda değer bir başarı gösterdiği görülmüştür.

Ek olarak, hata yüzdelerini azaltmak için, tahmin günü öncesindeki son iki haftanın saatlik yük verileri kullanılarak yapılan bir regülasyon hesabı tanımlanmıştır.

Ortalama günlük hata yüzdeleri, her iki yıl içinde regüleli yük tahmininde genel olarak daha düşük çıkmıştır. Dalgacık dönüşümü ve RTF sinir ağları yapısında bu durumun tersi geçerli olduğu görülmüştür. Regüleli yük tahmini çalışmasında, en düşük ortalama günlük hata yüzdelerini GKA ve RTF sinir ağları yapısı vermiştir.

Maksimum günlük hata yüzdeleri, 2009 yılı için genel olarak regülesiz yük tahmininde daha düşük çıkmıştır. 2010 yılı sonuçlarında ise, regüleli yük tahmininde maksimum günlük hata yüzdeleri daha düşük çıkmıştır. Dalgacık dönüşümü ve RTF sinir ağları yapısında bu durumun tersi olduğu görülmüştür.

Bu çalışma kapsamında, tatil günleri için yük tahmini yapılmamıştır. Gelecek çalışmalarda, bu günler için de bir model oluşturularak tahmin süreci daha da etkin bir hale getirilebilir.

KAYNAKLAR

- Arı, N., Özen, Ş. ve Çolak, Ö. H., “Sürekli Dalgacık Dönüşümü”, Dalgacık Teorisi (Wavelet) Matlab Uygulamaları İle, *Palme Yayıncılık*, 23-37 (2008).
- Azadeh, A., Ghadrei, S. F. ve Pourvalikhan Nokhandan, B., “Short term load forecasting by ANN”, *Hybrid Intelligent Models and Applications*, 39-43 (2009).
- Başaran Filik, Ü. ve Kurban M., “A New Approach for the Short-Term Load Forecasting with Autoregressive and Artificial Neural Network Models”, *International Journal of Computational Intelligence Research*, 3 (1): 66-71 (2007).
- Beale, M. H., Hagan, M. T. ve Demuth, H. B., “Neural Network Toolbox™ 7 User’s Guide”, *MathWorks*, 2010.
- Ceylan, G. ve Demirören, A., “Yapay sinir ağları ile Gölbaşı bölgesinin kısa dönem yük tahmini”, *ELECO 2004*, Bursa, 103-107 (2004).
- Chen , Q. S., Zhang, Xiong, X., S. H. ve Chen, X. W., “Short-term load forecasting with least square support vector machines and wavelet transform”, *Proceedings of the Seventh International Conference on Machine Learning and Cybernetics Kunming*, 1425-1429 (2008).
- Chen, Y., Luh, P. B., Guan, C., Zhao, Y., Michel, L. D., Coolbeth, M. A., Friedland, P. B. ve Rourke, S. J., “Short-Term Load Forecasting: Similar Day-Based Wavelet Neural Networks”, *IEEE Transactions on Power Systems*, 25 (1): 322-330 (2010).
- Demirel, Ö., “ANFIS ve ARMA modelleri ile elektrik enerjisi yük tahmini”, Yüksek Lisans Tezi, *Marmara Üniversitesi Fen Bilimleri Enstitüsü*, İstanbul, 2009.

KAYNAKLAR (devam ediyor)

- Dongxiao, N. ve Jie, T., “Improved RBF network applied to short-term load forecasting”, *ICSESS 2011*, 864-867 (2011).
- Gao, R. ve Tsoukalas, L. H., “Neural-Wavelet Methodology for Load Forecasting”, *Journal of Intelligent and Robotic Systems*, 31, 149-157 (2001).
- Gontar, Z. ve Hatziargyriou, N., “Short term load forecasting with radial basis function network”, *IEEE Porto Power Tech. Conference* (2001).
- Haykin, S., “Neural Networks: A Comprehensive Foundation”, *Prentice Hall*, 1998.
- He, Y. J., Zhu, Y. C., Duan, D. X. ve Sun, W., “Application of neural network model based on combination of fuzzy classification and input selection in short term load forecasting”, *Proceedings of the Fifth International Conference on Machine Learning and Cybernetics*, 3152-3156 (2006).
- Jain, A. ve Satish, B., “Integrated architecture for short-term load forecasting using support vector machines”, *NAPS’08*, 1-8 (2008).
- Karagülle, F., “Destek vektör makinelerini kullanarak yüz bulma”, Yüksek Lisans Tezi, *Trakya Üniversitesi Fen Bilimleri Enstitüsü*, Edirne, 2008.
- Kavzoğlu, T. ve Çölkesen, İ., “Destek Vektör Makineleri İle Uydu Görüntülerinin Sınıflandırılmasında Kernel Fonksiyonlarının Etkilerinin İncelenmesi”, *Harita Dergisi*, 144: 73-82 (2010).
- Khotanzad, A., Afkhami-Rohani, R., Lu, T. L., Abaye, A., Davis, M. ve Maratukulam, D. J., “ANNSTLF- A Neural-Network-Based Electric Load Forecasting System”, *IEEE Transactions on Neural Networks*, 8 (4): 835-846 (1997).

KAYNAKLAR (devam ediyor)

- Kim , K. H., Youn , H. S. ve Kang, Y. C., “ Short-Term Load Forecasting for Special Days İn Anomalous Load Conditions Using Neural Networks and Fuzzy İnference Method”, *IEEE Transactions on Power Systems*, 15 (2): 559-565 (2000).
- Lu , N. ve Zhou, J., “Particle swarm optimization-based RBF neural network load forecasting model”, *APPEEC 2009*, 27-31 (2009).
- Mandal, P., Senjyu, T., Urasaki, N. ve Funabashi, T., “A Neural Network Based Several-Hour-Ahead Electric Load Forecasting Using Similar Days Approach”, *Electrical Power and Energy Systems*, 28: 367-373 (2006).
- Mu, Q., Wu, Y., Pan, X., Huang, L. ve Li, X., “Short-term load forecasting using improved similar days method”, *APPEEC 2010*, 1-4 (2010).
- Nalbant, A., Aslan, Y. ve Yaşar, C., “Kütahya ili için elektrik puant yük tahmini”, *Elektrik Elektronik Bilgisayar Mühendisliği 11. Ulusal Kongresi*, 1: 211-214 (2005).
- Oğurlu, H., “Matematiksel modelleme kullanarak Türkiye’nin uzun dönem elektrik yük tahmini”, Yüksek Lisans Tezi, *Selçuk Üniversitesi Fen Bilimleri Enstitüsü*, Konya, 2011.
- Osman, Z. H., Awad, M. L. ve Mahmoud, T. K., “Neural network based approach for short-term load forecasting”, *Power Systems Conference and Exposition*, 1-8 (2009).
- Palit, A. K. ve Popovic, D., “Traditional Problem Definition”, Computational Intelligence in Time Series Forecasting, *Springer*, 17-75 (2005).

KAYNAKLAR (devam ediyor)

- Pandey, A. S., Singh, D. ve Sinha, S. K., “Intelligent Hybrid Wavelet Models For Short-Term Load Forecasting”, *IEEE Transactions on Power Systems*, 25 (3): 1266-1273 (2010).
- Pınarbaşı, M., “Elektrik enerji sistemlerinde talep tahmin yöntemleri ve yapay sinir ağları uygulaması”, Yüksek Lisans Tezi, *İstanbul Üniversitesi Fen Bilimleri Enstitüsü*, İstanbul, 2009.
- Rilling, G., Flandrin, P. ve Gonçalvès, P., “On empirical mode decomposition and its algorithms”, *NSIP-03* (2003).
- Sağıroğlu Ş., Beşdok, E. ve Erler, M., “Mühendislikte Yapay Zeka Uygulamaları-I Yapay Sinir Ağları”, *Ufuk Yayıncılık*, 2003.
- Soliman, S. A. ve Al-Kandari, A. M., “Static State Estimation”, Electrical Load Forecasting: Modeling and Model Construction, *Butterworth-Heinemann*, 45-78, 2010.
- Song, K. B., Back, Y. S., Hong, D. H. ve Jang, G., “Short-Term Load Forecasting For The Holidays Using Linear Regression Method”, *IEEE Transactions on Power Systems*, 20 (1): 96-101 (2005).
- Sümer, K. K., Gökteş, O. ve Hepsağ, A., “The Application Of Seasonal Latent Variable İn Forecasting Electricity Demand As An Alternative Method”, *Energy Policy*, 37: 1317-1322 (2009).
- Taylor, J. W. ve McSharry, P. E., “Short-Term Load Forecasting Methods: An Evaluation Based On European Data”, *IEEE Transactions on Power Systems*, 22 (4): 2213-2219 (2007).

KAYNAKLAR (devam ediyor)

- Türkmenoğlu, V., “Güç elektroniği devrelerinin bilgisayar destekli çözümlenmesinde dalgacık yaklaşımının incelenmesi”, Doktora Tezi, *Ondokuz Mayıs Üniversitesi Fen Bilimleri Enstitüsü*, Samsun, 2006.
- Xu, H., Wang, J. H. ve Zheng, S. Q., “Online daily load forecasting based on support vector machines”, *Proceedings of the Fourth International Conference on Machine Learning and Cybernetics Guangzhou*, 3985-3990 (2005).
- Yalçınöz, T., Herdem, S. ve Eminoğlu, U., “Yapay sinir ağları ile Niğde bölgesinin elektrik yük tahmini”, *ELECO 2002*, Bursa, 25-29 (2002).
- Yılmaçlı, M., “Türkiye’nin uzun dönem elektrik yük tahmini”, Yüksek Lisans Tezi, *Dumlupınar Üniversitesi Fen Bilimleri Enstitüsü*, Kütahya, 2010.
- Yılmaz, M., “Fırçasız doğru akım motorunun algılayıcısız kontrolünde dalgacık tekniğinin uygulanması”, Doktora Tezi, *İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü*, İstanbul, 2005.
- Yoldaş, U. C., “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.
- Zhu, Z., Sun, Y. ve Li, H., “Hybrid of EMD and SVMs for short-term load forecasting”, *ICCA 2007*, 1044-1047 (2007).

EKLER

Ek-1. YZ-KDYT Arayüzü Kodları

```

function varargout = YZ_KDYT (varargin)
warning off
% YZ_KDYT MATLAB code for YZ_KDYT.fig
% Begin initialization code - DO NOT EDIT
global str_sonuc
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
    'gui_Singleton', gui_Singleton, ...
    'gui_OpeningFcn', ...
    @YZ_KDYT_OpeningFcn, ...
    'gui_OutputFcn', ...
    @YZ_KDYT_OutputFcn, ...
    'gui_LayoutFcn', [], ...
    'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before YZ_KDYT is made visible.
function YZ_KDYT_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% --- Outputs from this function are returned to the command line.
function varargout = YZ_KDYT_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles) %onaylama
sira=0;
%yıl girişi
str3=findobj(gcbo,'Tag','popupmenu1');
yil=get(str3,'Value');
if yil==1
    errordlg('Yıl secilmedi!', 'Hata!', 'modal');
else
    if yil==2
        sira=32; %2009 yılı
        yil_gir=2009;
    else
        if yil==3
            sira=397; %2010 yılı
            yil_gir=2010;
        end
    end
end
str4=sira;
set(str3,'Value',str4)
%ay girişi
str2=findobj(gcbo,'Tag','popupmenu2');
ay=get(str2,'Value');
sira=get(str3,'Value');
if ay==1
    errordlg('Ay secilmedi!', 'Hata!', 'modal');
else

```

```
if ay==2  
    sira=sira; %ocak  
    ay_gir=01;  
else  
    if ay==3  
        sira=sira+31; %ubat  
        ay_gir=02;  
    else  
        if ay==4  
            sira=sira+59; %mart  
            ay_gir=03;  
        else  
            if ay==5  
                sira=sira+90; %nisan  
                ay_gir=04;  
            else  
                if ay==6  
                    sira=sira+120; %mayis  
                    ay_gir=05;  
                else  
                    if ay==7  
                        sira=sira+151; %haziran  
                        ay_gir=06;  
                    else  
                        if ay==8  
                            sira=sira+181; %temmuz  
                            ay_gir=07;  
                        else  
                            if ay==9  
                                sira=sira+212; %agustos  
                                ay_gir=08;  
                            else  
                                if ay==10  
                                    sira=sira+243; %eylul  
                                    ay_gir=09;  
                                else  
                                    if ay==11  
                                        sira=sira+273; %ekim  
                                        ay_gir=10;  
                                    else  
                                        if ay==12  
                                            sira=sira+304; %kasim  
                                            ay_gir=11;  
                                        else  
                                            if ay==13  
                                                sira=sira+334; %aralik  
                                                ay_gir=12;  
                                            end  
                                        end  
                                    end  
                                end  
                            end  
                        end  
                    end  
                end  
            end  
        end  
    end  
end  
  
str5=sira;  
set(str2,'Value',str5)  
%gün girişi  
str1=findobj(gcf,'Tag','popupmenu3');  
gun=get(str1,'Value');  
sira=get(str2,'Value');  
if gun==1  
    errordlg('Gun secilmedi!', 'Hata!', 'modal');  
else  
    if gun==2  
        sira=sira; %1. gun  
        gun_gir=1;  
        if ay==2
```

```

        warndlg('Yılbaşı tatili', 'Uyarı!', 'modal');
    end
else
    if gun==3;
        sira=sira+1; %2. gun
        gun_gir=2;
        if ay==2
            warndlg('Yılbaşı tatili', 'Uyarı!', 'modal');
        end
    else
        if gun==4
            sira=sira+2; %3. gun
            gun_gir=3;
            if ay==2
                warndlg('Yılbaşı tatili', 'Uyarı!', 'modal');
            end
        else
            if gun==5
                sira=sira+3; %4. gun
                gun_gir=4;
                if yıl==3&& ay==10
                    warndlg('Ramazan Bayramı Tatili', 'Uyarı!', 'modal');
                end
            else
                if gun==6
                    sira=sira+4; %5. gun
                    gun_gir=5;
                    if yıl==3&& ay==10
                        warndlg('Ramazan Bayramı Tatili', 'Uyarı!', 'modal');
                    end
                else
                    if gun==7
                        sira=sira+5; %6.gun
                        gun_gir=6;
                        if yıl==3&& ay==10
                            warndlg('Ramazan Bayramı Tatili', 'Uyarı!', 'modal');
                        end
                    else
                        if gun==8
                            sira=sira+6; %7.gun
                            gun_gir=7;
                            if yıl==3&& ay==10
                                warndlg('Ramazan Bayramı Tatili', 'Uyarı!', 'modal');
                            end
                        else
                            if gun==9
                                sira=sira+7; %8.gun
                                gun_gir=8;
                                if yıl==3&& ay==10
                                    warndlg('Ramazan Bayramı Tatili', 'Uyarı!', 'modal');
                                end
                            else
                                if gun==10
                                    sira=sira+8; %9.gun
                                    gun_gir=9;
                                    if yıl==3&& ay==10
                                        warndlg('Ramazan Bayramı Tatili', 'Uyarı!', 'modal');
                                    end
                                else
                                    if gun==11
                                        sira=sira+9; %10.gun
                                        gun_gir=10;
                                        if yıl==3&& ay==10
                                            warndlg('Ramazan Bayramı Tatili', 'Uyarı!', 'modal');
                                        end
                                    else
                                        if gun==12
                                            sira=sira+10; %11.gun
                                            gun_gir=11;
                                            if yıl==3&& ay==10
                                                warndlg('Ramazan Bayramı Tatili', 'Uyarı!', 'modal');
                                            end
                                        else

```

```

if gun==13
    sira=sira+11; %12.gun
    gun_gir=12;
    if yil==3&& ay==10
        warndlg('Ramazan Bayramı Tatili', 'Uyarı!', 'modal');
    end
else
    if gun==14
        sira=sira+12; %13.gun
        gun_gir=13;
        if yil==3&& ay==12
            warndlg('Kurban Bayramı Tatili', 'Uyarı!', 'modal');
        end
    else
        if gun==15
            sira=sira+13; %14.gun
            gun_gir=14;
            if yil==3&& ay==12
                warndlg('Kurban Bayramı Tatili', 'Uyarı!', 'modal');
            end
        else
            if gun==16
                sira=sira+14; %15.gun
                gun_gir=15;
                if yil==3&& ay==12
                    warndlg('Kurban Bayramı Tatili', 'Uyarı!', 'modal');
                end
            else
                if gun==17
                    sira=sira+15; %16.gun
                    gun_gir=16;
                    if yil==3&& ay==12
                        warndlg('Kurban Bayramı Tatili', 'Uyarı!', 'modal');
                    end
                else
                    if gun==18
                        sira=sira+16; %17.gun
                        gun_gir=17;
                        if yil==3&& ay==12
                            warndlg('Kurban Bayramı Tatili', 'Uyarı!', 'modal');
                        end
                    else
                        if gun==19
                            sira=sira+17; %18.gun
                            gun_gir=18;
                            if yil==3&& ay==12
                                warndlg('Kurban Bayramı Tatili', 'Uyarı!', 'modal');
                            end
                        else
                            if gun==20
                                sira=sira+18; %19.gun
                                gun_gir=19;
                                if ay==6
                                    warndlg('Atatürk ü Anma Gençlik ve Spor Bayramı', 'Uyarı!', 'modal');
                                end
                                if yil==2&& ay==10
                                    warndlg('Ramazan Bayramı Tatili', 'Uyarı!', 'modal');
                                end
                                if yil==3&& ay==12
                                    warndlg('Kurban Bayramı Tatili', 'Uyarı!', 'modal');
                                end
                            else
                                if gun==21
                                    sira=sira+19; %20.gun
                                    gun_gir=20;
                                    if yil==2&& ay==10
                                        warndlg('Ramazan Bayramı Tatili', 'Uyarı!', 'modal');
                                    end
                                    if yil==3&& ay==12
                                        warndlg('Kurban Bayramı Tatili', 'Uyarı!', 'modal');
                                    end
                                else
                                    if gun==22

```

```

sira=sira+20; %21.gun
gun_gir=21;
if yil==2&& ay==10
warndlg('Ramazan Bayramı Tatili', 'Uyarı!', 'modal');
end
if yil==2&& ay==12
warndlg('Kurban Bayramı Tatili', 'Uyarı!', 'modal');
end
if yil==3&& ay==12
warndlg('Kurban Bayramı Tatili', 'Uyarı!', 'modal');
end
else
if gun==23
sira=sira+21; %22.gun
gun_gir=22;
if yil==2&& ay==10
warndlg('Ramazan Bayramı Tatili', 'Uyarı!', 'modal');
end
if yil==2&& ay==12
warndlg('Kurban Bayramı Tatili', 'Uyarı!', 'modal');
end
else
if gun==24
sira=sira+22; %23.gun
gun_gir=23;
if ay==5
warndlg('Ulusal Egemenlik ve Çocuk Bayramı', 'Uyarı!', 'modal');
end
if yil==2&& ay==10
warndlg('Ramazan Bayramı Tatili', 'Uyarı!', 'modal');
end
if yil==2&& ay==12
warndlg('Kurban Bayramı Tatili', 'Uyarı!', 'modal');
end
else
if gun==25
sira=sira+23; %24.gun
gun_gir=24;
if yil==2&& ay==10
warndlg('Ramazan Bayramı Tatili', 'Uyarı!', 'modal');
end
if yil==2&& ay==12
warndlg('Kurban Bayramı Tatili', 'Uyarı!', 'modal');
end
else
if gun==26
sira=sira+24; %25.gun
gun_gir=25;
if yil==2&& ay==10
warndlg('Ramazan Bayramı Tatili', 'Uyarı!', 'modal');
end
if yil==2&& ay==12
warndlg('Kurban Bayramı Tatili', 'Uyarı!', 'modal');
end
else
if gun==27
sira=sira+25; %26.gun
gun_gir=26;
if yil==2&& ay==10
warndlg('Ramazan Bayramı Tatili', 'Uyarı!', 'modal');
end
if yil==2&& ay==12
warndlg('Kurban Bayramı Tatili', 'Uyarı!', 'modal');
end
else
if gun==28;
sira=sira+26; %27.gun
gun_gir=27;
if yil==2&& ay==10
warndlg('Ramazan Bayramı Tatili', 'Uyarı!', 'modal');
end
if yil==2&& ay==12
warndlg('Kurban Bayramı Tatili', 'Uyarı!', 'modal');
end

```



```
end  
end  
end  
end  
end  
end  
end  
end  
end  
end  
end  
end  
end  
end  
end  
end  
end  
  
str_gun=findobj(gcf,'Tag','text1');  
str_ay=findobj(gcf,'Tag','text2');  
str_yil=findobj(gcf,'Tag','text3');  
str_yil_yaz=num2str(yil_gir,'%10.0f\n');  
str_ay_yaz=num2str(ay_gir,'%10.0f\n');  
str_gun_yaz=num2str(gun_gir,'%10.0f\n');  
set(str_gun,'String',str_gun_yaz);  
set(str_ay,'String',str_ay_yaz);  
set(str_yil,'String',str_yil_yaz);  
  
str6=sira;  
set(str1,'Value',str6)  
  
% --- Executes on button press in pushbutton2.  
function pushbutton2_Callback(hObject, eventdata, handles) %YSA  
data=importdata('D:\MIL\ÇALIŞMALAR\TEZ\Teas Verileri_05.2006\gui_veri.xlsx','Sayfa1');  
str1=findobj(gcf,'Tag','popupmenu3');  
gun=get(str1,'Value');  
MAPE_nn=0;  
MAPE_nn_reguleli=0;  
for gun=gun  
    %birinci hafta  
    ocak=data.Sayfa1(1:24,gun-17);  
    ocak=[ocak data.Sayfa1(1:24,gun-16)];  
    ocak=[ocak data.Sayfa1(1:24,gun-15)];  
    veri_giris1=reshape(ocak,24*3,1);  
    veri_cikis1=data.Sayfa1(1:24,gun-14);  
  
    %ikinci hafta  
    ocak=data.Sayfa1(1:24,gun-10);  
    ocak=[ocak data.Sayfa1(1:24,gun-9)];  
    ocak=[ocak data.Sayfa1(1:24,gun-8)];  
    veri_giris2=reshape(ocak,24*3,1);  
    veri_cikis2=data.Sayfa1(1:24,gun-7);  
  
    ca_veri_giris_training_nn=[veri_giris1 veri_giris2];  
    ca_veri_cikis_training_nn=[veri_cikis1 veri_cikis2];  
    tic  
    %Egitilen newtwork (YSA)  
    net_nn=newff(ca_veri_giris_training_nn,ca_veri_cikis_training_nn,10);  
  
    % Test için önceden verilmeyen veriler!!!!  
    ocak=data.Sayfa1(1:24,gun-3);  
    ocak=[ocak data.Sayfa1(1:24,gun-2)];  
    ocak=[ocak data.Sayfa1(1:24,gun-1)];  
    veri_giris3=reshape(ocak,24*3,1);  
    veri_cikis3=data.Sayfa1(1:24,gun);  
  
    %Test (YSA)  
    Y_test_haftasi_nn = sim(net_nn,veri_giris3);
```

```

tt=1:24;
plot(handles.axes1,tt,veri_cikis3,tt,Y_test_haftasi_nn,'r*','LineWidth',2)
xlabel(handles.axes1,'Saat','fontsize',12,'fontweight','b')
ylabel(handles.axes1,'Yük Talebi(W)','fontsize',12,'fontweight','b')
legend(handles.axes1,'Geçek','Tahmin','fontsize',12,'fontweight','b','Location','SouthEast')

MAPE_nn=sum(abs((Y_test_haftasi_nn-veri_cikis3)./veri_cikis3))*100/24;
str=findobj(gcf,'Tag','text4');
str_sonuc=num2str(MAPE_nn,'%10.5f\n');
set(str,'String',str_sonuc);

%Regülasyon hesabi
training_hafta2_toplam=0;
for i=gun-14:gun-8
    for j=1:24
        training_hafta2_toplam=training_hafta2_toplam+data.Sayfa1(j,i);
    end
end

test_hafta_toplam=0;
for i=gun-7:gun-1
    for j=1:24
        test_hafta_toplam=test_hafta_toplam+data.Sayfa1(j,i);
    end
end
oran=test_hafta_toplam/training_hafta2_toplam;
str_reg=findobj(gcf,'Tag','text6');
str_oran=num2str(orán,'%10.5f\n');
set(str_reg,'String',str_oran);

Y_test_haftasi_nn_reguleli=Y_test_haftasi_nn*(test_hafta_toplam/training_hafta2_toplam);
plot(handles.axes2,tt,veri_cikis3,tt,Y_test_haftasi_nn_reguleli,'r*','LineWidth',2)
xlabel(handles.axes2,'Saat','fontsize',12,'fontweight','b')
ylabel(handles.axes2,'Yük Talebi (W)','fontsize',12,'fontweight','b')
legend(handles.axes2,'Gerçek','Tahmin','fontsize',12,'fontweight','b','Location','SouthEast')

MAPE_nn_reguleli=sum(abs((Y_test_haftasi_nn_reguleli-veri_cikis3)./veri_cikis3))*100/24;
str_reguleli=findobj(gcf,'Tag','text5');
str_sonuc_reguleli=num2str(MAPE_nn_reguleli,'%10.5f\n');
set(str_reguleli,'String',str_sonuc_reguleli);
sure=toc;
str_toc=findobj(gcf,'Tag','text7');
str_sure=num2str(sure,'%10.5f\n');
set(str_toc,'String',str_sure);
end
% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles) %DD ve YSA
data=importdata('D:\DİL\ÇALIŞMALAR\TEZ\Teas Verileri_05.2006\gui_veri.xlsx','Sayfa1');
str1=findobj(gcf,'Tag','popupmenu3');
gun=get(str1,'Value');
MAPE_ff=0;
MAPE_ff_reguleli=0;
for gun=gun
    gun
    %birinci hafta
    ocak=data.Sayfa1(1:24,gun-17);
    ocak=[ocak data.Sayfa1(1:24,gun-16)];
    ocak=[ocak data.Sayfa1(1:24,gun-15)];
    veri_giris1=reshape(ocak,24*3,1);
    veri_cikis1=data.Sayfa1(1:24,gun-14);

    %ikinci hafta
    ocak=data.Sayfa1(1:24,gun-10);
    ocak=[ocak data.Sayfa1(1:24,gun-9)];
    ocak=[ocak data.Sayfa1(1:24,gun-8)];
    veri_giris2=reshape(ocak,24*3,1);
    veri_cikis2=data.Sayfa1(1:24,gun-7);

    [ca_veri_giris_training11 cd1] = wavedec(veri_giris1,5,'db2');
    [ca_veri_cikis_training12 cd2] = wavedec(veri_cikis1,5,'db2');

    [ca_veri_giris_training21 cd1] = wavedec(veri_giris2,5,'db2');

```

```

[ca_veri_cikis_training22 cd2] = wavedec(veri_cikis2,5,'db2');

ca_veri_giris_training=[ca_veri_giris_training11 ca_veri_giris_training21];
ca_veri_cikis_training=[ca_veri_cikis_training12 ca_veri_cikis_training22];
tic
%Egitilen newtwork wavelet (DD ve YSA)
net_ff=newff(ca_veri_giris_training,ca_veri_cikis_training,10);

Y_birinci_hafta_ff = sim(net_ff,ca_veri_giris_training(:,1));
Y_ikinci_hafta_ff = sim(net_ff,ca_veri_giris_training(:,2));

veri_cikis1_reconstructed_ff=waverec(Y_birinci_hafta_ff,cd2,'db2');
veri_cikis2_reconstructed_ff=waverec(Y_ikinci_hafta_ff,cd2,'db2');

% Test için önceden verilmeyen veriler!!!!
ocak=data.Sayfa1(1:24,gun-3);
ocak=[ocak data.Sayfa1(1:24,gun-2)];
ocak=[ocak data.Sayfa1(1:24,gun-1)];
veri_giris3=reshape(ocak,24*3,1);
veri_cikis3=data.Sayfa1(1:24,gun);

[ca_veri_giris_training31 cd1] = wavedec(veri_giris3,5,'db2');

Y_test_haftasi_ff = sim(net_ff,ca_veri_giris_training31);
veri_cikis3_reconstructed_ff=waverec(Y_test_haftasi_ff,cd2,'db2');

tt=1:24;
plot(handles.axes1,tt,veri_cikis3,tt,veri_cikis3_reconstructed_ff,'r*','LineWidth',2)
xlabel(handles.axes1,'Saat','fontsize',12,'fontweight','b')
ylabel(handles.axes1,'Yük Talebi (W)','fontsize',12,'fontweight','b')
legend(handles.axes1,'Gerçek','Tahmin','fontsize',12,'fontweight','b','Location','SouthEast')

MAPE_ff=sum(abs((veri_cikis3_reconstructed_ff-veri_cikis3)./veri_cikis3))*100/24;
str=findobj(gcf,'Tag','text4');
str_sonuc=num2str(MAPE_ff,'%10.5f\n');
set(str,'String',str_sonuc);

%Regülasyon hesabi
training_hafta2_toplam=0;
for i=gun-14:gun-8
    for j=1:24
        training_hafta2_toplam=training_hafta2_toplam+data.Sayfa1(j,i);
    end
end

test_hafta_toplam=0;
for i=gun-7:gun-1
    for j=1:24
        test_hafta_toplam=test_hafta_toplam+data.Sayfa1(j,i);
    end
end
oran=test_hafta_toplam/training_hafta2_toplam;
str_reg=findobj(gcf,'Tag','text6');
str_oran=num2str(orán,'%10.5f\n');
set(str_reg,'String',str_oran);

veri_cikis3_reconstructed_ff_reguleli=veri_cikis3_reconstructed_ff*(test_hafta_toplam/training_hafta2_toplam);
plot(handles.axes2,tt,veri_cikis3,tt,veri_cikis3_reconstructed_ff_reguleli,'r*','LineWidth',2)
xlabel(handles.axes2,'Saat','fontsize',12,'fontweight','b')
ylabel(handles.axes2,'Yük Talebi (W)','fontsize',12,'fontweight','b')
legend(handles.axes2,'Gerçek','Tahmin','fontsize',12,'fontweight','b','Location','SouthEast')

MAPE_ff_reguleli=sum(abs((veri_cikis3_reconstructed_ff_reguleli-veri_cikis3)./veri_cikis3))*100/24;
str_reguleli=findobj(gcf,'Tag','text5');
str_sonuc_reguleli=num2str(MAPE_ff_reguleli,'%10.5f\n');
set(str_reguleli,'String',str_sonuc_reguleli);
sure=toc;
str_toc=findobj(gcf,'Tag','text7');
str_sure=num2str(sure,'%10.5f\n');
set(str_toc,'String',str_sure);
end

```

```

% --- Executes on button press in pushbutton4.
function pushbutton4_Callback(hObject, eventdata, handles) %DD ve RTFSA
data=importdata('D:\DİLİÇALİŞMALAR\TEZ\Teas Verileri_05.2006\gui_veri.xlsx','Sayfa1');
str1=findobj(gcf,'Tag','popupmenu3');
gun=get(str1,'Value');
MAPE_rbf=0;
MAPE_rbf_reguleli=0;
for gun=gun
gun
%birinci hafta
ocak=data.Sayfa1(1:24,gun-17);
ocak=[ocak data.Sayfa1(1:24,gun-16)];
ocak=[ocak data.Sayfa1(1:24,gun-15)];
veri_giris1=reshape(ocak,24*3,1);
veri_cikis1=data.Sayfa1(1:24,gun-14);

%ikinci hafta
ocak=data.Sayfa1(1:24,gun-10);
ocak=[ocak data.Sayfa1(1:24,gun-9)];
ocak=[ocak data.Sayfa1(1:24,gun-8)];
veri_giris2=reshape(ocak,24*3,1);
veri_cikis2=data.Sayfa1(1:24,gun-7);

[ca_veri_giris_training11 cd1] = wavedec(veri_giris1,5,'db2');
[ca_veri_cikis_training12 cd2] = wavedec(veri_cikis1,5,'db2');

[ca_veri_giris_training21 cd1] = wavedec(veri_giris2,5,'db2');
[ca_veri_cikis_training22 cd2] = wavedec(veri_cikis2,5,'db2');

ca_veri_giris_training=[ca_veri_giris_training11 ca_veri_giris_training21];
ca_veri_cikis_training=[ca_veri_cikis_training12 ca_veri_cikis_training22];
tic
%Egitilen newtwork radyal tabanlı wavelet (DD ve RTFSA)
net_rbf = newrb(ca_veri_giris_training,ca_veri_cikis_training,0,1);

% Test için önceden verilmeyen veriler!!!
ocak=data.Sayfa1(1:24,gun-3);
ocak=[ocak data.Sayfa1(1:24,gun-2)];
ocak=[ocak data.Sayfa1(1:24,gun-1)];
veri_giris3=reshape(ocak,24*3,1);
veri_cikis3=data.Sayfa1(1:24,gun);

[ca_veri_giris_training31 cd1] = wavedec(veri_giris3,5,'db2');

Y_test_haftasi_rbf = sim(net_rbf,ca_veri_giris_training31);
veri_cikis3_reconstructed_rbf=waverec(Y_test_haftasi_rbf,cd2,'db2');

tt=1:24;
plot(handles.axes1,tt,veri_cikis3,tt,veri_cikis3_reconstructed_rbf,'r*','LineWidth',2)
xlabel(handles.axes1,'Saat','fontsize',12,'fontweight','b')
ylabel(handles.axes1,'Yük Talebi (W)','fontsize',12,'fontweight','b')
legend(handles.axes1,'Gerçek','Tahmin','fontsize',12,'fontweight','b','Location','SouthEast')

MAPE_rbf=sum(abs((veri_cikis3_reconstructed_rbf-veri_cikis3)./veri_cikis3))*100/24;
str=findobj(gcf,'Tag','text4');
str_sonuc=num2str(MAPE_rbf,'%10.5f\n');
set(str,'String',str_sonuc);

%Regülasyon hesabi
training_hafta2_toplam=0;
for i=gun-14:gun-8
for j=1:24
training_hafta2_toplam=training_hafta2_toplam+data.Sayfa1(j,i);

end
end

test_hafta_toplam=0;
for i=gun-7:gun-1
for j=1:24
test_hafta_toplam=test_hafta_toplam+data.Sayfa1(j,i);
end
end
end

```

```

oran=test_hafta_toplam/training_hafta2_toplam;
str_reg=findobj(gcf,'Tag','text6');
str_oran=num2str(oran,'%10.5f\n');
set(str_reg,'String',str_oran);

veri_cikis3_reconstructed_rbf_reguleli=veri_cikis3_reconstructed_rbf*(test_hafta_toplam/training_hafta2_toplam);
plot(handles.axes2,tt,veri_cikis3,tt,veri_cikis3_reconstructed_rbf_reguleli,'*','LineWidth',2)
xlabel(handles.axes2,'Saat','fontsize',12,'fontweight','b')
ylabel(handles.axes2,'Yük Talebi (W)','fontsize',12,'fontweight','b')
legend(handles.axes2,'Gerçek','Tahmin','fontsize',12,'fontweight','b','Location','SouthEast')

MAPE_rbf_reguleli=sum(abs((veri_cikis3_reconstructed_rbf_reguleli-veri_cikis3)./veri_cikis3))*100/24;
str_reguleli=findobj(gcf,'Tag','text5');
str_sonuc_reguleli=num2str(MAPE_rbf_reguleli,'%10.5f\n');
set(str_reguleli,'String',str_sonuc_reguleli);
sure=toc;
str_toc=findobj(gcf,'Tag','text7');
str_sure=num2str(sure,'%10.5f\n');
set(str_toc,'String',str_sure);
end

% --- Executes on button press in pushbutton5.
function pushbutton5_Callback(hObject, eventdata, handles) %GKA ve RTFSA
data=importdata('D:\ADIL\ÇALIŞMALAR\TEZ\Teas Verileri_05.2006\gui_veri.xlsx','Sayfa1');
str1=findobj(gcf,'Tag','popupmenu3');
gun=get(str1,'Value');
MAPE_emd=0;
MAPE_emd_reguleli=0;
rbf_giris1=[];
rbf_cikis1=[];
rbf_giris2=[];
rbf_cikis2=[];
imf1_topla=[];
imf2_topla=[];
rbf_giris_test=[];
imf_test_dagit=[];
imf_test_topla=[];

for gun=gun
%birinci hafta
ocak=data.Sayfa1(1:24,gun-17);
ocak=[ocak data.Sayfa1(1:24,gun-16)];
ocak=[ocak data.Sayfa1(1:24,gun-15)];
veri_giris1=reshape(ocak,24*3,1);
veri_cikis1=data.Sayfa1(1:24,gun-14);

imf_giris1=emd(veri_giris1);
a=size(imf_giris1);
a1=a(:,1);
for i=1:a1
    gir1=imf_giris1(i,:);
    rbf_giris1=[rbf_giris1 gir1];
end
rbf_giris1=rbf_giris1';
rbf_giris1=[rbf_giris1; zeros(72,1)];

imf_cikis1=emd(veri_cikis1);
b=size(imf_cikis1);
b1=b(:,1);
for i=1:b1
    cik1=imf_cikis1(i,:);
    rbf_cikis1=[rbf_cikis1 cik1];
end
rbf_cikis1=rbf_cikis1';

%ikinci hafta
ocak=data.Sayfa1(1:24,gun-10);
ocak=[ocak data.Sayfa1(1:24,gun-9)];
ocak=[ocak data.Sayfa1(1:24,gun-8)];
veri_giris2=reshape(ocak,24*3,1);

```

```

veri_cikis2=data.Sayfa1(1:24,gun-7);

imf_giris2=emd(veri_giris2);
c=size(imf_giris2);
c1=c(:,1);
for i=1:c1
    gir2=imf_giris2(i,:);
    rbf_giris2=[rbf_giris2 gir2];
end
rbf_giris2=rbf_giris2';
rbf_giris2=[rbf_giris2; zeros(72,1)];

imf_cikis2=emd(veri_cikis2);
d=size(imf_cikis2);
d1=d(:,1);
for i=1:d1
    cik2=imf_cikis2(i,:);
    rbf_cikis2=[rbf_cikis2 cik2];
end
rbf_cikis2=rbf_cikis2';

if length(rbf_giris1)<length(rbf_giris2)
    sonuc_gir=length(rbf_giris2);
    fark_gir=sonuc_gir-length(rbf_giris1);
    sifir_gir=zeros(fark_gir,1);
    rbf_giris1=[rbf_giris1;sifir_gir];
else
    sonuc_gir=length(rbf_giris1);
    fark_gir=sonuc_gir-length(rbf_giris2);
    sifir_gir=zeros(fark_gir,1);
    rbf_giris2=[rbf_giris2;sifir_gir];
end

if length(rbf_cikis1)<length(rbf_cikis2)
    sonuc_cik=length(rbf_cikis2);
    fark_cik=sonuc_cik-length(rbf_cikis1);
    sifir_cik=zeros(fark_cik,1);
    rbf_cikis1=[rbf_cikis1;sifir_cik];
else
    sonuc_cik=length(rbf_cikis1);
    fark_cik=sonuc_cik-length(rbf_cikis2);
    sifir_cik=zeros(fark_cik,1);
    rbf_cikis2=[rbf_cikis2;sifir_cik];
end

nn_veri_giris_training=[rbf_giris1 rbf_giris2];
nn_veri_cikis_training=[rbf_cikis1 rbf_cikis2];
tic
%Egitilen newtwork radyal tabanlı emd (GKA ve RTFSA)

net_rbf_emd = newrb(nn_veri_giris_training,nn_veri_cikis_training,0,1);

imf1_rbf = sim(net_rbf_emd,nn_veri_giris_training(:,1));
imf2_rbf = sim(net_rbf_emd,nn_veri_giris_training(:,2));

e=length(imf1_rbf);
n=e/24;
m=e/n;
t=[(n-1)*m]+1;

imf1_reconstructed=0;

for j=1:24:t
    imf1_rbf_dagit=imf1_rbf(j:j+23,1);
    imf1_topla=[imf1_topla imf1_rbf_dagit];
end
for j=1:n
    imf1_reconstructed=imf1_reconstructed+imf1_topla(:,j);
end

f=length(imf2_rbf);
n1=f/24;

```

```

m1=f/n1;
t1=((n1-1)*m1)+1;
for j=1:24:t1
    imf2_rbf_dagit=imf1_rbf(j:j+23,1);
    imf2_topla=[imf2_topla imf2_rbf_dagit];
end
imf2_reconstructed=0;
for j=1:n1
    imf2_reconstructed=imf2_reconstructed+imf2_topla(:,j);
end

% Test için önceden verilmeyen veriler!!!!
ocak=data.Sayfa1(1:24,gun-3);
ocak=[ocak data.Sayfa1(1:24,gun-2)];
ocak=[ocak data.Sayfa1(1:24,gun-1)];
veri_giris3=reshape(ocak,24*3,1);
veri_cikis3=data.Sayfa1(1:24,gun);

imf_giris_test=emd(veri_giris3);
test=size(imf_giris_test);
test1=test(:,1);
for i=1:test1
    gir_test=imf_giris_test(i,:);
    rbf_giris_test=[rbf_giris_test gir_test];
end

rbf_giris_test=rbf_giris_test';

if length(rbf_giris1)<length(rbf_giris_test)
    sonuc_test=length(rbf_giris_test);
    fark_test=sonuc_test-length(rbf_giris1);
    sifir_test=zeros(fark_test,1);
    rbf_giris1=[rbf_giris1;sifir_test];
    rbf_giris2=[rbf_giris2;sifir_test];
else
    sonuc_test=length(rbf_giris1);
    fark_test=sonuc_test-length(rbf_giris_test);
    sifir_test=zeros(fark_test,1);
    rbf_giris_test=[rbf_giris_test;sifir_test];
end

%Test (EMD&RBF)
imf_test_rbf = sim(net_rbf_emd,rbf_giris_test);

x=length(imf2_rbf);
x1=x/24;
y=x/x1;
w=((x1-1)*y)+1;
for j=1:24:w
    imf_test_dagit=imf1_rbf(j:j+23,1);
    imf_test_topla=[imf_test_topla imf_test_dagit];
end
imf_test_reconstructed=0;
for j=1:x1
    imf_test_reconstructed=imf_test_reconstructed+imf_test_topla(:,j);
end

tt=1:24;
plot(handles.axes1,tt,veri_cikis3,tt,imf_test_reconstructed,'*','LineWidth',2)
xlabel(handles.axes1,'Saat','fontsize',12,'fontweight','b')
ylabel(handles.axes1,'Yük Talebi (W)','fontsize',12,'fontweight','b')
legend(handles.axes1,'Gerçek','Tahmin','fontsize',12,'fontweight','b','Location','SouthEast')

MAPE_emd=sum(abs((imf_test_reconstructed-veri_cikis3)./veri_cikis3))*100/24;
str=findobj(gcf,'Tag','text4');
str_sonuc=num2str(MAPE_emd,'%10.5f\n');
set(str,'String',str_sonuc);

%Regülasyon hesabi
training_hafta2_toplam=0;
for i=gun-14:gun-8
    for j=1:24
        training_hafta2_toplam=training_hafta2_toplam+data.Sayfa1(j,i);
    end
end

```

```

    end
end

test_hafta_toplam=0;
for i=gun-7:gun-1
    for j=1:24
        test_hafta_toplam=test_hafta_toplam+data.Sayfa1(j,i);
    end
end
oran=test_hafta_toplam/training_hafta2_toplam;
str_reg=findobj(gcf,'Tag','text6');
str_oran=num2str(oran,'%10.5f\n');
set(str_reg,'String',str_oran);

imf_test_reconstructed_reguleli=imf_test_reconstructed*(test_hafta_toplam/training_hafta2_toplam);
plot(handles.axes2,tt,veri_cikis3,tt,imf_test_reconstructed_reguleli,'r*','LineWidth',2)
xlabel(handles.axes2,'Saat','fontsize',12,'fontweight','b')
ylabel(handles.axes2,'Yük Talebi (W)','fontsize',12,'fontweight','b')
legend(handles.axes2,'Gerçek','Tahmin','fontsize',12,'fontweight','b','Location','SouthEast')

MAPE_emd_reguleli=sum(abs((imf_test_reconstructed_reguleli-veri_cikis3)/veri_cikis3))*100/24;
str_reguleli=findobj(gcf,'Tag','text5');
str_sonuc_reguleli=num2str(MAPE_emd_reguleli,'%10.5f\n');
set(str_reguleli,'String',str_sonuc_reguleli);
sure=toc;
str_toc=findobj(gcf,'Tag','text7');
str_sure=num2str(sure,'%10.5f\n');
set(str_toc,'String',str_sure);
end

% --- Executes on selection change in popupmenu1.

function popupmenu1_Callback(hObject, eventdata, handles) %Yıl girişi

% --- Executes during object creation, after setting all properties.
function popupmenu1_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on selection change in popupmenu2.
function popupmenu2_Callback(hObject, eventdata, handles) %Ay girişi

% --- Executes during object creation, after setting all properties.
function popupmenu2_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on selection change in popupmenu3.
function popupmenu3_Callback(hObject, eventdata, handles) %Gün girişi

% --- Executes during object creation, after setting all properties.
function popupmenu3_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pushbutton6.
function pushbutton6_Callback(hObject, eventdata, handles) %Sıfırlama
close all
YZ_KDYT

```

ÖZGEÇMİŞ

Kişisel Bilgiler



ADI SOYADI : İdil IŞIKLI ESENER
DOĞUM YERİ VE TARİHİ : Ankara / 01.11.1986

Eğitim Durumu

Lisans Öğrenimi : Eskişehir Osmangazi Üniversitesi Elektrik-Elektronik
Mühendisliği Anabilim Dalı (2005-2010)

Bildiği Yabancı Diller : İngilizce

Bilimsel Faaliyetleri :

Işıklı İ., Açıklıkalp E., Yamık H. ve Kurban M., 2011. Biyodizelin Dizel Santrallerde Kullanım Analizi, *6th International Advanced Technologies Symposium (IATS'11)*, 16-18 Mayıs 2011, Elazığ, Turkey.

Bozdemir M., **Işıklı İ.** ve Kurban M., 2011. Hidromobillerde Kullanılan Motorlar ve Sürücü Devreleri, *6th International Advanced Technologies Symposium (IATS'11)*, 16-18 Mayıs 2011, Elazığ, Turkey.

Işıklı İ., Yüksel T. ve Kurban M., 2012. Short-Term Load Forecasting Without Weather Data: A Comparison Study, *The 12th International Conference on Probabilistic Methods Applied to Power Systems (PMAPS'12)*, 10-14 Haziran 2012, İstanbul, Turkey.

Işıklı Esener İ., Yüksel T. ve Kurban M., 2012. Artificial Intelligence Based Hybrid Structures for Short- Term Load Forecasting Without Temperature Data, *IEEE International Energy Conference & Exhibition (ENERGYCON'12)*, 9-12 Eylül 2012, Florence, Italy (gönderildi).

Işıklı Esener İ., Yüksel T. ve Kurban M., 2012. Short-term load forecasting without meteorological data using AI based structures, *Expert Systems with Applications* (gönderildi).

İş Deneyimi

Stajlar : Eskişehir 1'inci Hava İkmal Bakım Merkezi Komutanlığı (2010)
Eskişehir Osmangazi Elektrik Dağıtım A.Ş. (2009)
Çalıştığı Kurumlar : Bilecik Şeyh Edebali Üniversitesi Mühendislik Fakültesi
(Şubat 2011-bugüne)

İletişim

Adres: Bilecik Şeyh Edebali Üniversitesi Gülümbe Kampüsü Mühendislik Fakültesi
Elektrik-Elektronik Mühendisliği

Tel: 0228 216 01 01 - 1465

E-Posta Adresi: idil.isikli@bilecik.edu.tr

Tarih: Haziran, 2012

İmza :