

**T.C.
HACETTEPE ÜNİVERSİTESİ
SAĞLIK BİLİMLERİ ENSTİTÜSÜ**

**BOX-JENKINS ZAMAN SERİSİ ANALİZ YÖNTEMİ ile İLERİ
BESLEMELİ YAPAY SİNİR AĞLARI TAHMİNLERİNİN
KARŞILAŞTIRMASI**

CANSEL BİÇEN

**Biyoistatistik Programı
YÜKSEK LİSANS TEZİ**

**ANKARA
2006**

**T.C.
HACETTEPE ÜNİVERSİTESİ
SAĞLIK BİLİMLERİ ENSTİTÜSÜ**

**BOX-JENKINS ZAMAN SERİSİ ANALİZ YÖNTEMİ ile İLERİ
BESLEMELİ YAPAY SİNİR AĞLARI TAHMİNLERİNİN
KARŞILAŞTIRMASI**

CANSEL BİÇEN

**Biyoistatistik Programı
YÜKSEK LİSANS TEZİ**

**TEZ DANIŞMANI
Prof. Dr. Reha ALPAR**

**ANKARA
2006**

Sağlık Bilimleri Enstitüsü Müdürlüğü'ne:

Bu çalışma jürimiz tarafından Biyoistatistik Programında Yüksek Lisans Tezi olarak kabul edilmiştir.

Jüri Başkanı :	Prof. Dr. Ergun Karaağaoğlu Hacettepe Üniversitesi	(imza)
Danışman :	Prof. Dr. Reha Alpar Hacettepe Üniversitesi	(imza)
Üye :	Doç. Dr. Atilla Halil Elhan Ankara Üniversitesi	(imza)
Üye :	Yrd.Doç. Dr. Kenan Köse Ankara Üniversitesi	(imza)
Üye :	Dr. Erdem Karabulut Hacettepe Üniversitesi	(imza)

ONAY:

Bu tez, Hacettepe Üniversitesi Eğitim-Öğretim ve Sınav Yönetmeliği'nin ilgili maddeleri uyarınca yukarıdaki jüri üyeleri tarafından uygun görülmüş ve Enstitü Yönetim Kurulu kararıyla kabul edilmiştir.

(İmza)
Prof. Dr. Hakan S. Orer
Enstitü Müdürü

TEŞEKKÜR

Çalışmalarımnda bilgi ve deneyimlerini paylaşarak yönlendiren tez danışmanım, Sayın Prof.Dr.Reha Alpar'a, bu tezin gerçekleşmesi için sonsuz destek ve katkılarını esirgemeyen Biyoistatistik Ana Bilim Dalı'nın başta çok değerli Bölüm Başkanı Sayın Prof. Dr. Ergun Karaağaoğlu olmak üzere, Sayın Doç. Dr. Osman Saraçbaşı, Öğretim Görevlileri Erdem Karabulut ve Pınar Geyik'e ve desteğini almaktan çok mutlu olduğum insan iyisi Menekşe'ye çok teşekkür ederim.

Tez izleme komitesinde görev alarak çalışmaya değerli katkılar sağlayan Sayın Doç.Dr. Atilla Halil Elhan ve Sayın Yrd.Doç. Dr. Kenan Köse'ye içtenlikle teşekkür ediyorum.

Sevgili aileme, bana inanan ve sabreden eşim ve çocuklarıma da teşekkür ediyorum.

ÖZET

Cansel, B. Box-Jenkins Zaman Serisi Analiz Yöntemi ile İleri Beslemeli Yapay Sinir Ağları Tahminlerinin Karşılaştırması. Hacettepe Üniversitesi Sağlık Bilimleri Enstitüsü Biyoistatistik Programı Yüksek Lisans Tezi, Ankara, 2006. Box Jenkins zaman serileri analizi yöntemi, kestirimlerde yaygın bir şekilde kullanılan kabul görmüş bir yöntemdir. Box-Jenkins yöntemi, doğrusal olmayan zaman serilerinde yetersizdir. Diğer taraftan yapay sinir ağları doğrusal ve doğrusal olmayan modellemede geleneksel istatistik tahmin yöntemlerine göre alternatif olarak kullanılmaktadır. Yapay sinir ağlarındaki en temel dezavantaj ise uygun olan model mimarisini belirlemedeki güçlüklerdir. Bu çalışmada, Yapay Sinir Ağları ve Box-Jenkins yöntemlerinin kestirim aracı olarak kullanılmasına ilişkin ayrıntılar açıklanmıştır. Geri yayılım öğrenme algoritmasının mantığı gösterilmiştir. Her iki yöntem ile, Ankara İli SO₂ ölçümlerine ilişkin 83 adet aylık veri kullanılarak kestirim yapılmıştır. Modellemede yer alan bazı önemli aşamalar yöntemler bazında açıklanmıştır. SO₂ ölçümlerine ilişkin veri örneği ile kurulan modeller, performans istatistikleri ile yorumlanmıştır.

Anahtar Kelimeler: Kestirim, Zaman Serileri, Yapay Sinir Ağları.

ABSTRACT

Cansel, B. Comparison between the Box-Jenkins and feed forward artificial neural network forecasts in time series analysis method. Hacettepe University Institute of Health Sciences, Ms Thesis in Biostatistics, Ankara, 2006. Box Jenkins time series analysis methodology is an acceptable prediction methodology that has been widely used. Box-Jenkins methodology is inefficient for nonlinear series. Whilst artificial neural network has been used as an alternative to the traditional statistical forecasting methods for linear and non-linear modelling. Most basic disadvantage of the artificial neural network is the difficulties in the identifying appropriate model architecture. This study presents in detail Box-Jenkins and artificial neural network methodologies as a forecasting tool. Logic of the Back Propagation algorithm is shown. 83 monthly SO₂ measurements data in Ankara municipality are forecasted by both methodologies. In modelling some important methodology based stages are described. Constructed models with SO₂ data are analysed by the performance statistics.

Key Words: Forecasting, Time Series, Artificial Neural Network

İÇİNDEKİLER

ONAY SAYFASI	iii
TEŞEKKÜR	iv
ÖZET	v
ABSTRACT	vi
1.GİRİŞ	1
2. ZAMAN SERİLERİ ANALİZİ	5
2.1. Zaman Serisi	5
2.2. Zaman Serilerinin Özellikleri	6
2.2.1. Trend (Eğilim)	6
2.2.2. Belirli Aralıklarla Yinelenen (Periyodik) ve Mevsimlik Dalgalanmalar	6
2.2.3. Çevrimsel (Devri) değişmeler	7
2.2.4. Düzensiz değişme	7
2.3. Zaman Serilerinin Gösterimi	8
2.4. Zaman Serileri Modelleri ile Kestirim	10
2.5. Performans Ölçütü	10
2.6. Box-Jenkins Modelleri	12
2.6.1. Box-Jenkins Modelinin Kurulumu	13
2.6.2. Otokovaryans ve Otokorelasyon Fonksiyonları	14
2.6.3. Otokovaryans ve Otokorelasyon Fonksiyonlarının Kestirimi	16
2.6.4. Örnek Otokorelasyonları Standart Hatası	16
2.6.5. Kısmi Otokorelasyonlar	18
2.6.6. Model Kurma	19
2.6.7. Otoregresif (Auto Regressive-AR) Modeller	20
2.6.7.1. Yule-Walker Eşitliği Cinsinden Otoregresif Parametreler	24
2.6.7.2. AR(1) Süreci	24
2.6.7.3. AR(2) Süreci	25
2.6.8. Hareketli Ortalama (Moving Average-MA) Modelleri	28
2.6.8.1. MA(1) Süreci	32
2.6.8.2. MA(2) Süreci	33
2.6.9. Karışık (ARMA) Modeller	34
2.6.9.1. ARMA (1,1) Süreci	37
2.6.10. Durağan Olmayan Doğrusal Stokastik Modeller	37
2.6.11. Model Belirleme	40
2.6.12. Q test istatistiği	42
2.6.13. Kestirim	44
2.6.14. Mevsimlik zaman serileri için Box-Jenkins modelleri	46
3. YAPAY SİNİR AĞLARI	48
3.1. Giriş	48
3.2. YSA'nın Temelleri	50
3.2.1. İşlem Birimi	51
3.2.2. Toplama (Birleştirme) Fonksiyonu	52
3.2.3. Aktivasyon Fonksiyonu	53
3.2.4. YSA'nın Yapısı	56
3.2.5. YSA'nın Eğitimi	57
3.2.6. Amaç Fonksiyonu	61

3.3.	Öğrenme Algoritmaları	61
3.3.1.	Geri yayılım algoritması	62
4.	GELECEĞİ KESTİRİM PROBLEMLERİ İÇİN YSA	65
4.1.	Giriş.....	65
4.2.	Konu ile İlgili Yapılmış Çalışmalar	66
4.3.	Geleceği Kestirimde YSA ile Modellemede Önemli Noktalar.....	68
4.3.1.	Ağ Yapısı	69
4.3.2.	Girdi nöronu sayısı.....	70
4.3.3.	Gizli katman ve gizli nöron sayısı.....	71
4.3.4.	Çıktı nöronu sayısı	73
4.3.5.	Nöronlar arası bağlantılar.....	75
4.3.6.	Aktivasyon fonksiyonu	75
4.3.7.	Öğrenme algoritması.....	76
4.3.8.	Veri normalleştirme	77
4.3.9.	Eğitim kümesi ve test kümesi	79
4.3.10.	YSA'nın performansının belirlenmesi.....	80
4.4.	YSA'nın Kestirimde Göreceli Performansı.....	81
4.5.	YSA'nın İstatistiksel Yöntemlere Göre Güçlü ve Zayıf Yanları.....	81
4.5.1.	YSA'nın güçlü olduğu yanları.....	81
4.5.2.	YSA'nın zayıf olduğu yanları.....	83
5.	BULGULAR.....	85
5.1.	Giriş.....	85
5.2.	Kestirimlerin üretilmesi	86
5.3.	Box-Jenkins Yöntemi ile SO ₂ verilerinin kestirimi	86
5.3.1.	Model tanımlama ve parametre tahmin aşaması	86
5.3.2.	Modelin geçerliliği	89
5.3.3.	Kestirim.....	91
5.4.	YSA ile Modelleme ve SO ₂ Verilerinin Kestirimi.....	92
5.4.1.	Kullanılan seri değerlerinin normalleştirilmesi.....	92
5.4.2.	Çalışmada kullanılan ağ yapısı	93
5.4.3.	Öğrenme algoritması.....	94
5.4.4.	Eğitim ve test kümesi	95
5.4.5.	Çalışmada kullanılan performans ölçütleri	95
5.4.6.	Kestirim tekniklerinin performans ölçütüne etkisi.....	96
5.4.7.	Kestirim dönemi ve sonuçlarının grafiksel karşılaştırması.....	97
6.	TARTIŞMA VE SONUÇ	100
	KAYNAKLAR	102

SİMGELER ve KISALTMALAR

AR	Otoregresif (<i>Auto regressive</i>)
MA	Hareketli Ortalama (<i>Moving Averages</i>)
ARMA	Otoregresif-Hareketli Ortalama (<i>Auto regressive Moving Average</i>)
ARIMA	Bütünleşik Otoregresif-Hareketli Ortalama (<i>Auto regressive integrated moving average</i>)
OH	Ortalama hata
OMH	Ortalama mutlak hata
HKT	Hata kareleri toplamı
HKO	Hata kareleri ortalaması
HKOK	Hata kareleri ortalaması karekökü
YH	Yüzde hata
OYH	Ortalama yüzde hata
OMYH	Ortalama mutlak yüzde hata
OKK	Otokorelasyon katsayıları
OKF	Otokorelasyon fonksiyonu
ÖOKK	Örnek otokorelasyon katsayıları
ÖOKF	Örnek otokorelasyon fonksiyonu
YSA	Yapay sinir ağı
ÇKA	Çok katmanlı algılayıcı
GYA	Geri yayılım algoritması
Y_t	Zaman serisinin tahmin modeli tarafından üretilen t zamanındaki değeri
F_t	Zaman serisinin tahmin modeli tarafından üretilen t zamanındaki değeri
y_t	Y_t seri değerinden sürecin ortalamasının çıkartılması ile elde edilen yeni seri değeri
ε_t	Tahmin modelinin t zamanındaki tahmin hatası
$v(\varepsilon)$	Tesadüfi değişkeninin varyansı
$E(e)$	Tesadüfi değişkeninin beklenen değeri
Kov	X ve Y tesadüfi değişkenleri arasındaki kovaryans

ŞEKİLLER

Sayfa

2.1. Tipik zaman serisi yapıları.....	8
2.2. AR (p) süreci otokorelasyon ve kısmi otokorelasyon fonksiyonu	23
2.3. AR(1) süreci otokorelasyon ve kısmi otokorelasyon fonksiyonu	25
2.4. AR(2) modeli otokorelasyon ve kısmi otokorelasyon fonksiyonları	29
2.5. MA(q) modeli otokorelasyon ve kısmi otokorelasyon fonksiyonu.....	31
2.6. MA(1) süreci otokorelasyon ve kısmi otokorelasyon fonksiyonu.....	33
2.7. MA(2) süreci otokorelasyon ve kısmi otokorelasyon fonksiyonları.....	35
2.8. ARMA (1,1) otokorelasyon ve kısmi otokorelasyon fonksiyonları	38
2.9 Box-Jenkins modelinin kurulumu.....	41
3.1. Temel insan sinir hücresi	49
3.2. Yapay sinir hücresi modeli.....	50
3.3. Basit bir YSA Yapısı	51
3.4. İşlem birimi	52
3.5. Doğrusal fonksiyon	54
3.6 Basamak fonksiyonu.....	54
3.7 Sigmoid fonksiyonu.....	55
3.8 Hiperbolik tanjant fonksiyonu	55
3.9. İleri Beslemeli Ağ Yapısı.....	56
3.10. Geri Beslemeli Ağ Yapısı.....	57
3.11. Öğreticili Öğrenme.....	58
4.1 Doğrusal kestiricisinin eğrisel bir yapıyı kestirimindeki performansı.....	82

5.1. SO ₂ verilerinin zaman serisi	87
5.2. SO ₂ verilerinin durağanlaştırılmış zaman serisi grafiği	87
5.3. SO ₂ verilerinin mevsimsel durağanlaştırılmış zaman serisi grafiği	88
5.4. Mevsimsel durağanlaştırılmış serinin otokorelasyonları	88
5.5. Mevsimsel durağanlaştırılmış serinin kısmi otokorelasyonları	89
5.6. SO ₂ verileri kestirim hataları	90
5.7. SO ₂ verileri kestirim hataları otokorelasyonları	90
5.8. SO ₂ verileri kestirim hataları kısmi otokorelasyonları	91
5.9. SO ₂ verileri kestirimi.....	92
5.10. SO ₂ verilerinin kestirimi için kurulan YSA yapısı.....	93
5.11. Yöntemlere göre kestirim sonuçları.....	97
5.12. Eğitim-test oranları değiştirildiğinde elde edilen kestirim sonuçları	99

TABLÖLAR

	Sayfa
2.1 Model belirlemede OKF ve KOKF kullanımı	42
3.1. Bazı Birleştirme Fonksiyonları.....	53
5.1. Zaman Serisi Uygulama Verisi	85
5.2 Uyum iyiliği performans sonuçları	96
5.3. Eğitim-test oranlarına göre uyum iyiliği sonuçları	98

1.GİRİŞ

Kestirim (öngörü) kavramı, bir değişkenin belirli varsayımlar altında gelecekte alabileceği değerlerin önceden yaklaşık olarak belirlenmesi olarak tanımlanır. Zaman serisi çözümlemesi ile öngörü, incelenen bir değişkenin şimdiki ve geçmiş dönemdeki gözlem değerlerini kullanarak ve birtakım varsayımlar altında öngörü değerlerinin hangi sınırlar arasında geçkeleşebileceğini ortaya koymak için yapılan uğraşlardır (1).

Tıbbi karar uygulamalarında amaç doğru tanıdır. Doğru tanıda yanlış kararlar alma riskinin getireceği olumsuzlukları ortadan kaldırılabilmek üzere, uzman sistemler, tıbbi görüntü analizi, konsültasyon, literatür vb. karar destek sistemlerinin yanı sıra öngörülerde, gerek zaman serisi analizinden gerekse yapay sinir ağlarından (YSA) sıklıkla faydalanılmaktadır. Öngörüler daima, karar vericilerin üstlendiği riskleri azaltarak karar vericiye destek olmaktadır. Tıbbi tanılara ilişkin hesapsal güçlükleri çözmek üzere Yapay Sinir Ağları yönteminden artan sayıda uygulamalar ile yararlanıldığı görülmüştür. Yapay Sinir Ağları elde edilen deneyimler ve teorik derinliği ile kayda değer ölçüde hatasız ve kesin başarı elde etmesi bakımından, diğer kalıplaşmış uygulamalara göre, hala genç bir disiplindir. Son yıllarda özellikle tıbbi fizik alanındaki tanı uygulamalarında anlamlı katkıları bulunmaktadır.

Tıbbi tanıların çok çabuk verilmesinin hastaya sağlayacağı yararı tartışmak bile gereksizdir. Örneğin, ağır kafa yaralanması yaşayan hastaların yarısı genellikle 36 saat içinde ölmektedir. Bunun yanı sıra nörolojik tanıda kapsamlı testler ortalama 24 saatte gerçekleştirilebilmektedir. Dolayısıyla, karar destek anlamında YSA teknolojisi ile geliştirilen uygulamalardan yararlanılması, nörolojik bileşkelerin öğrenilmesine yönelik testlerin daha hızlı, daha sık yapılabilmesi tıbbi tanıdaki başarı bakımından önemli bir kazanımdır (41). Herhangi bir değişkenin gelecekte alacağı değerlerin kestirimi, değişkenin geçmiş dönemlerdeki gözlem değerleri kullanılarak oluşturulan modeller yardımı ile sağlanır. Kestirim analizi model belirleme ve kestirim aşamasından oluşur. Kestirim modelinin verinin geçmiş değerleri ile en iyi

uyumu saęlayan ve geleceęi en iyi temsil eden nitelikte olması öngörülerdeki başarıyı arttırır.

Model kurma aşamasının başlangıcında ilgili probleme ait veriler saęlıklı bir şekilde toplanır ve toplanan verilere uygun model belirlenir. Belirlenen modelin matematiksel biçimi atanarak, var olan veriler yardımı ile modelin parametreleri belirlenmeye çalışılır. Modelin uyumu hata testleri ile saęlanabilir. Eğer kurulan model uygun bir model ise, gelecek için kestirimler yapılabilir.

Kestirim yöntemleri, nitel kestirim yöntemleri ve nicel kestirim yöntemleri olmak üzere iki şekilde sınıflandırılabilir. Her iki yöntemin çıkış noktası ilgili deęişkene ait gözlem deęerleridir. Geçmiş ve şimdiki dönem gözlem deęerlerinden, gelecek dönem gözlem deęerleri belirli kurallar çerçevesinde kestirilir (2).

Nicel kestirim yöntemleri ise istatistiksel yöntemlere dayanır. Nicel kestirim yöntemlerinde nasıl kestirimde bulunduęu net olarak bellidir ve işlemler matematikselidir. Geçmiş gözlem deęerleri kullanılarak sürecin oluşmasına katkıda bulunan ilişkiler belirlenir ve bu ilişkilerin geleceęe yansması belirlenmeye çalışılır. Nicel kestirimde bulunabilmek için iki temel yaklaşım kullanılmaktadır: neden-sonuç ilişkisine dayanan modeller ve zaman serileri analizine dayalı modeller (3). Neden-sonuç ilişkisine dayanan kestirim modelleri, regresyon yöntemi ve ekonometrik modellerdir. Regresyon yöntemi, bir ya da daha fazla deęişkenin üzerinde etkili olduęu bir başka deęişkenle aralarındaki ilişkinin matematiksel olarak ifade edilmesidir. Böyle bir yöntemde, etkilenen deęişkene bağımlı deęişken ve etkileyen deęişkene(lere) de bağımsız deęişken(ler) adı verilir. Bağımlı deęişken ile bağımsız deęişken(ler) arasındaki neden sonuç ilişkisi geçmiş gözlem deęerleri yardımı ile belirlenir. Daha sonra bağımsız deęişkenlerin gelecekteki çeşitli deęerleri için bağımlı deęişkenin alacaęı deęer kestirilmeye çalışılır.

Ekonometrik modeller, neden-sonuç ilişkisi gösteren iki ya da daha çok regresyon modelinden oluşan denklem sistemidir. Dolayısıyla, ekonometrik modellerde birden fazla bağımlı deęişken bulunur. Ekonometrik modellerde tüm

bağımlı ve bağımsız değişkenler arasındaki ilişkiler eşzamanlı olarak incelenir. Böylece, bağımlı ve bağımsız değişkenler arasındaki ilişkiler daha gerçekçi bir şekilde değerlendirilir (2).

Bir zaman serisi, bir değişkene ilişkin zamana göre sıralanmış gözlem değerleridir. Zaman serisi analizi, kestirimde bulunulacak değişkenin geçmiş zaman serisini kullanarak gelecek değerlerin kestirimi için model geliştirmede kullanılır. Model geliştirme, ilgili değişkene ait zaman serisinin analiz edilmesi, serinin ana eğiliminin ve özelliklerinin belirlenmesine dayanır. Serinin ana eğilimini ve özelliklerini yansıtacağı düşünülen bir model seçilir ve varolduğu seri değerleri kullanılarak modelin parametreleri yaklaşık olarak bulunur. Serinin gelecekte de aynı özellikleri koruyacağı ve aynı eğilimi göstereceği varsayılarak, belirlenen model yardımı ile gelecek dönem değerleri kestirilmeye çalışılır (2).

Zaman serileri analizi için yaygın olarak kullanılan bazı yöntemler vardır. Doğrusal zaman serilerinin analizinde oldukça başarılı sonuçlar veren Box-Jenkins modelleri bu tekniklerin en önemlilerindendir. Doğrusal ve durağan süreçlerde ya da durağan olmayan fakat bazı dönüşümlerle durağanlaştırılabilen serilerde Box-Jenkins yöntemi başarılı sonuçlar verebilir. Ancak, gerçek hayata ilişkin seriler genellikle doğrusal değildir. Bu nedenle doğrusal olmayan zaman serilerini modellemede farklı yöntemlere gereksinim duyulur.

1980'li yılların sonlarından başlamak üzere zaman serilerine ilişkin kestirimler için kullanılmakta olan yöntemlerden biri de Yapay Sinir Ağları (YSA) yöntemidir. YSA, girdi ve çıktı değişkenleri arasındaki herhangi bir ön bilgiye gereksinim duymadan doğrusal ve doğrusal olmayan modellemeyi sağlayabilmektedir. Bu nedenle YSA, kestirim aracı olarak diğer yöntemlere göre daha genel ve esnektir (4).

YSA'nın bir kestirim aracı olarak kullanılmasına ilişkin bir çok araştırmacı tarafından yapılmış çok sayıda çalışma olmasına rağmen, YSA'nın performansını etkileyen anahtar faktörlerin neler olduğu konusunda kesin bir yargı yoktur. Bu

konuda Zhang ve diğ.(6)'leri bir benzetim çalışması yaparak YSA'nın performansını etkileyebilecek ana faktörleri belirlemeye çalışmıştır. Yazarlar, sözü edilen faktörleri; girdi nöronu sayısı, gizli nöron sayısı ve eğitim kümesi büyüklüğü olarak incelemişlerdir. Belirtilen bu faktörlere ek olarak, eğitim algoritması, veri kümesinin düzenlenmesi, kestirim dönemi uzunluğu faktörlerinin de YSA performansı üzerinde etkili olduğu düşünülmektedir. Bu nedenle bütün bu faktörlerin etkisini araştırarak bir çalışmanın faydalı olacağına inanılmaktadır.

Bu çalışmada, YSA ile geliştirilen model ile zaman serileri analizinde sıklıkla kullanılan ve başarılı sonuçlar üreten Box-Jenkins modelleri yönteminin genel tanımlamaları yapılarak, gerçek veri uygulaması üzerinden elde edilen performans istatistikleri yardımı ile modeller karşılaştırılacaktır. YSA ile bulunan sonuçların etkinliği, literatüre uygun şekilde, Box-Jenkins modelleri sonuçları ile karşılaştırılarak araştırılmıştır. Box-Jenkins modellerinin geniş kabul görmesi, geliştirilen her yeni model için iyi bir karşılaştırma aracı olmalarını sağlamıştır. Ayrıca, Al-Saba ve Al-Amin (7) Box-Jenkins modellerinin; bileşenlerine ayırma (decomposition), basit üstel düzeltme ve Winters Yöntemi gibi diğer kestirim yöntemlerinden daha doğru sonuçlar verdiğini belirtmektedir.

Bu çalışma, amacı doğrultusunda şu şekilde planlanmıştır: Bölüm 2'de zaman serileri analizi için Box-Jenkins modelleri açıklanmıştır. Bölüm 3'te YSA hakkında ayrıntılı bilgi verilmiş, bilinen geri yayılım öğrenme algoritmasının mantığı gösterilmiştir. Bölüm 4'de, YSA'nın kestirim aracı olarak kullanılmasını içermektedir. YSA'nın kestirim aracı olarak kullanılmasına ilişkin literatür taraması ve önemli ayrıntılar bu bölümde açıklanmıştır. Bölüm 5'de, gerçek zaman serileri örneği olarak ele alınan; Refik Saydam Hıfzısıhha Başkanlığı'ndan alınan, Ankara İli 1999 Yılı-2005 (Kasım Sonu) itibarıyla, SO₂ ölçümleri ortalamasına ilişkin 83 adet aylık veri ile kestirim çalışmaları yapılmıştır. Sözü geçen veri üzerinden Box-Jenkins modellerine uygun olarak üretilen ARIMA modeli ve YSA teknikleri ile elde edilen model ve sonuçlarının performans istatistiklerine yer verilmiştir.

2. ZAMAN SERİLERİ ANALİZİ

2.1. Zaman Serisi

Zaman serisi, zaman sırasına konmuş gözlem değerleri kümesi olarak tanımlanabilir. Zaman serisinde ilgilenilen özellik bir değişkendir. Bu değişken zaman içerisinde çeşitli nedenlere bağlı olarak farklı değerler alır. Dolayısıyla zaman serisi, zaman sırasına konmuş değişken değer kümesi olarak ifade edilebilir (2). Zaman aralıkları her seride farklı farklıdır. Saatlik, günlük, haftalık, aylık, üç aylık, yıllık veya daha farklı zaman aralıklarına göre değer almış zaman serileri olabilir. Bir zaman serisinde, t , zaman devresini, Y_t ise değişkenin t dönemindeki gözlem değerini gösterir.

Zaman serisini oluşturan değerler kümesi sürekli bir küme ise zaman serisinin sürekli olduğu, eğer küme kesikli ise zaman serisinin kesikli olduğu söylenebilir. Kesikli bir zaman serisinden alınan gözlem değerleri Y_{t_1} , Y_{t_2} , ..., Y_{t_n} olarak gösterilebilir. $t_1, t_2, \dots, t_n$ gözlem zamanlarını göstermektedir. Gözlem zamanları arasındaki fark eşittir ve sabittir. Kesikli zaman serilerinde gözlem zamanları arasındaki fark eşit ve sabit olmayabilir. Kesikli zaman serileri iki şekilde elde edilebilir: (8)

1. Sürekli bir zaman serisinden örnekleme yapılarak,
2. Bir değişkenin değerini belli bir zaman diliminde gözleyerek.

Zaman serileri için bir başka sınıflandırma da seriyi oluşturan değişkenin ortalama ve varyansının zaman boyunca gösterdiği değişkenliğe göre yapılmaktadır. İlgili değişkenin ortalama ve varyansı zaman boyunca sabit ve çeşitli gecikmelerde kovaryansı t 'ye bağlı değil ise, zaman serisine durağan, eğer serinin ortalama ve varyansı zamana bağlı olarak değişiyorsa seriye durağan olmayan zaman serisi denir.

2.2. Zaman Serilerinin Özellikleri

Bir zaman serisinde düzenli ya da düzensiz çeşitli değişimler olabilir. Zaman serilerinde görülen düzenli değişimler genel ve sürekli nedenlerin etkisi sonucu meydana gelirken, düzensiz değişimler geçici ve rasgele nedenlerin etkisinden oluşur.

Ancak bir çok zaman serisi, tutarlı bir davranış yapısı gösterir. Bir zaman serisi kestirimi yapabilmek için bu davranış yapısının belirlenmesi ve geleceğe yansıtılması gerekir. Zaman serilerinde genel olarak görülen davranış yapıları aşağıdaki gibi özetlenebilir (13):

1. Uzun dönem eğilimi: Trend
2. Periyodik dalgalanmalar
3. Çevrimsel değişimler
4. Düzensiz dalgalanmalar

2.2.1. Trend (Eğilim)

Trend, bir zaman serisinde görülen sürekli artış ya da sürekli azalış eğilimidir. Trendin biçiminin bilinmesi, işletmelerin uzun döneme göre süresi sınırlanmış (uzun vadeli) plan ve programlarının en iyi biçimde düzenlenmesine olanak tanır. Trend, genel ve sürekli nedenlerin etkisinden dolayı ortaya çıkar.

2.2.2. Belirli Aralıklarla Yinelenen (Periyodik) ve Mevsimlik Dalgalanmalar

Eşit aralıklarla düzenli bir biçimde tekrarlanan dalgalanmalara periyodik değişme adı verilir. Birbirini izleyen iki en yüksek veya en düşük değişme arasında geçen süreye de değişimin periyodu denir. Periyodik değişme, değişimin tamamlanması için geçen sürenin uzunluğuna göre adlandırılır. Günlük, haftalık,

aylık, yıllık vb. periyodik deęişmeler olabilir. En önemli periyodik deęişme mevsimlik deęişmedir. Bu sebeple mevsimlik deęişme ifadesi de sıklıkla kullanılır. Mevsimlik deęişmeler bir yıldan kısa süreli ay ve üç aylık zaman serilerinde görülebilir. Bir yıldan daha uzun süreli zaman serilerinde mevsimlik deęişme görülmez (2).

2.2.3. Çevrimsel (Devri) deęişmeler

Bir yıldan uzun fakat deęişik sürelerle tekrarlanan dalgalanmalardır. Çevrimsel deęişmelerin süreleri çoğunlukla 3-15 yıl arasında deęişir. Genel ve sürekli nedenlerin ortaya çıkardığı çevrimsel deęişmeler az olarak bazı ekonomik zaman serilerinde görülür. Bir çok zaman serisinde çevrimsel (devri) deęişme görülmez.

2.2.4. Düzensiz deęişme

Ne zaman ne biçimde meydana geleceęi bilinmeyen dalgalanmalardır. Düzensiz deęişmelerin nedeni geçici ve raslantısal nedenlerlerdir. Düzensiz deęişmeler her zaman serisinde bulunabilir.

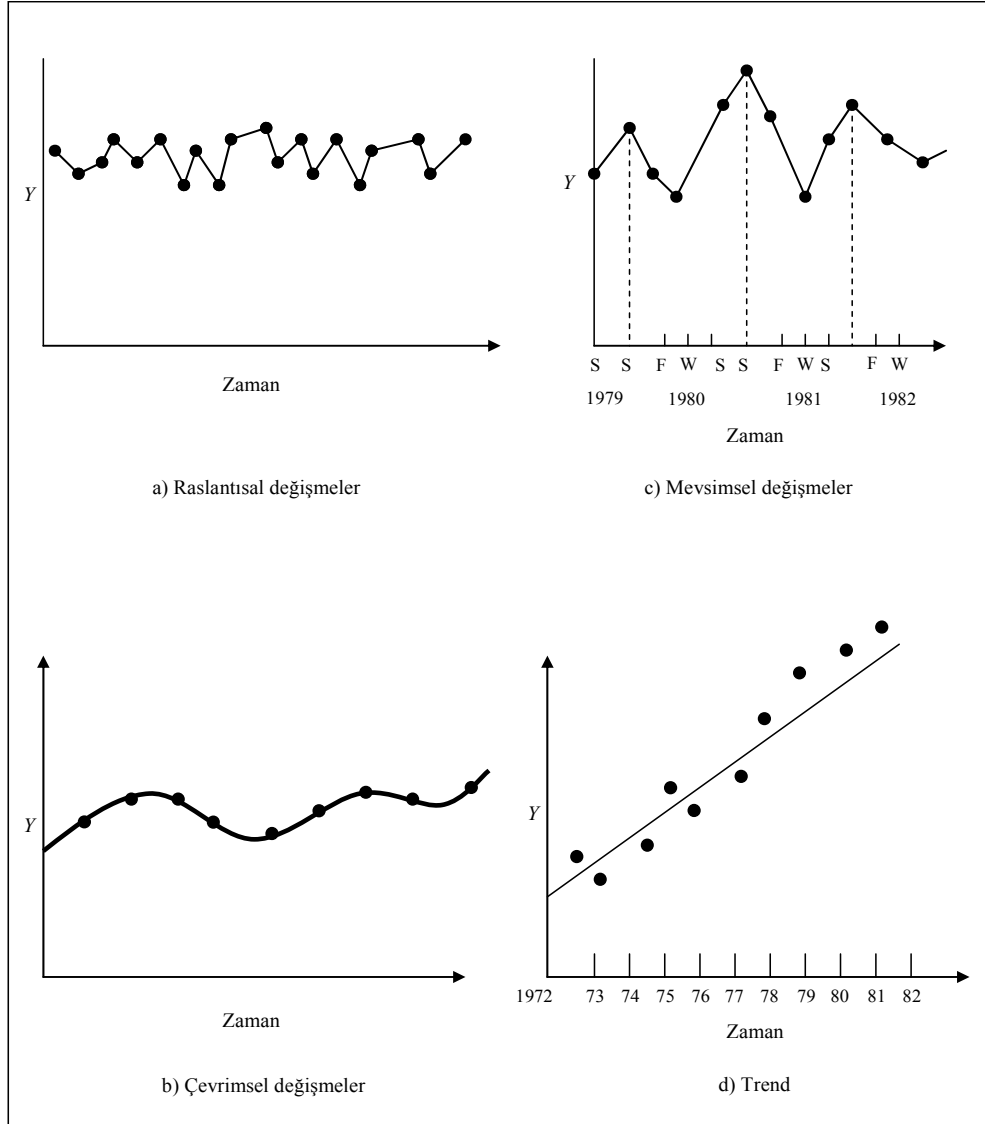
Şekil 2.1.'de, zaman serilerinin özellik yapılarına ilişkin grafikler gösterilmiştir. Şekil 2.1.a, zaman boyunca sabit bir düzeyde kalan bir süreci göstermektedir. Bu süreç sadece raslantısal nedenlerlerden dolayı deęişkenlik göstermektedir. Şekil 2.1.b'de sürecin çevrimsel ve raslantısal nedenlerden deęişmelerin etkisinde olduğu görülmektedir. Şekil 2.1.c'de süreç düzeyi mevsimsel bir deęişkenlik göstermektedir. Şekil 2.1.d'de ise süreç artan bir trend göstermektedir. Bir çok zaman serisi modeli bu yapılardan birini ya da bunların bileşimini temsil etmek üzere geliştirilirler (13).

Zaman serisinin temelindeki yapı tanımlandıktan sonra bu yapının davranışı matematiksel bir modelle ortaya konmalıdır. Hiçbir matematiksel modelin zaman serisindeki her bir deęeri tam dengi ile üretemeyeceęi açıktır. Gerçek zaman serisi

değerleri ile matematiksel formüllerle üretilen seri değerleri arasında bir fark olacaktır. Bundan dolayı zaman serisinin iki bileşenden oluşacağı söylenebilir (8)

$$\text{Zaman serisi} = \text{model değeri} + \text{hata} \quad [2.1]$$

$$Y_t = F_t + e_t \quad [2.2]$$



Şekil 2.1. Tipik zaman serisi yapıları (13)

2.3. Zaman Serilerinin Gösterimi

Zaman serilerini göstermek üzere kullanılan modellerin çoğu matematiksel modellerdir. Örneğin, gözlem değerleri bir olasılık dağılımından alınan rasgele

örneklerse ve dağılımın ortalaması zamana bağlı olarak değişmiyorsa, aşağıdaki sabit model bu durum için kullanılabilir.

$$Y_t = \mu + \varepsilon_t$$

Y_t , t anındaki gözlem değeri, μ , sürecin ortalaması, ε_t hata terimi ya da gürültü değişkeni olarak adlandırılır. Hatta raslantı değişkeninin, beklenen değerinin sıfır ve genellikle varyansının $V(\varepsilon_t) = \sigma_\varepsilon^2$ sabit olduğu kabul edilir. Anılan model Şekil 2.1.a'daki süreç için uygun bir modeldir.

Şekil 2.1.d'deki süreci temsil eden modeli göstermek için sürecin ortalamasının zaman bağlı olarak doğrusal değiştiği varsayılmış ve doğrusal bir trend modeli kullanılmıştır.

$$Y_t = b_0 + b_1 t + \varepsilon_t \quad [2.4]$$

b_0 ve b_1 sabit katsayılardır. b_1 bir periyottan diğer periyota gözlem değerindeki ortalama değişim miktarını göstermektedir. Eş.2.5, karesel bir trend modelini göstermektedir.

$$Y_t = b_0 + b_1 t + b_2 t^2 + \varepsilon_t \quad [2.5]$$

Periyodik değişimler, modele transdental terim eklenmesi ile gösterilebilirler:

$$Y_t = b_0 + b_1 \sin \frac{2\pi t}{12} + b_2 \cos \frac{2\pi t}{12} + \varepsilon_t \quad [2.6]$$

Yukarıda tanımlanan modeller aşağıdaki gibi genel bir biçimde yazılabilir :

$$Y_t = b_0 z_0(t) + b_1 z_1(t) + \dots + b_k z_k(t) + \varepsilon_t \quad [2.7]$$

b_i katsayıları, $z_i(t)$ t'ye bağlı matematiksel fonksiyonları ve ε_t de hata rasgele değişkenini ifade etmektedir. Bu gösterim, sürecin beklenen değerini t'nin matematiksel bir fonksiyonu olarak sunmaktadır (3).

2.4. Zaman Serileri Modelleri ile Kestirim

Zaman serileri modelleri ile kestirim, uygun bir modelin parametrelerinin kestirimi yardımı ile sağlanır. Örneğin ilgili modelin Eş. 2.4.'teki gibi olduğu bir durumda, β_1 ve β_2 parametrelerinin bir kestirimi olan b_1 ve b_2 kullanılarak geleceğe yönelik kestirim yapılabilir.

$$F_t = b_0 + b_1 t \quad [2.8]$$

T gözlem değerine sahip bir seride, T (T periyodunun sonu) zaman diliminden gelecek bir periyoda ($T + \tau$) yönelik bir kestirim yapılacaksa,

$$F_{T+\tau}(T) = b_0 + b_1 (T + \tau) \quad [2.9]$$

eşitliği istenen kestirim değerini verecektir.

2.5. Performans Ölçütü

Kestirim sisteminin etkinliğini değerlendirmek üzere birçok ölçüt kullanılabilir. Bunların en önemlisi, kestirim doğruluğudur. Kestirim yönteminin doğruluğu, kestirim hatalarının analiz edilmesi ile belirlenir. Kestirim hatası, gerçek gözlem değeri ile kestirim edilen değer arasındaki farktır. Herhangi bir t periyodu için kestirim hatası,

$$e_t = Y_t - F_t \quad [2.10]$$

eşitliği ile hesaplanır. Y_t , gerçek gözlem değerini ve F_t de modelin ürettiği değeri ifade etmektedir.

İncelenilen bir süreç ve kestirim metodu için kestirim hatasının, ortalaması $E(e)$ ve varyansı σ_e^2 olan normal raslantısal bir değişken olduğu varsayılır. Eğer kestirim yanlı değilse, $E(e)=0$ olur. Yanlı olmayan bir kestirim isteniyorsa, çok sayıda gözlem değeri ile çalışılması oldukça önemlidir (3).

Aşağıdaki eşitliklerde sıklıkla kullanılan performans ölçütlerine yer verilmiştir.

$$\text{Ortalama Hata (OH)} = \frac{1}{n} \sum_{i=1}^n e_i \quad [2.11]$$

$$\text{Ortalama Mutlak Hata (OMH)} = \frac{1}{n} \sum_{i=1}^n |e_i| \quad [2.12]$$

$$\text{Hata Kareleri Toplamı (HKT)} = \sum_{i=1}^n e_i^2 \quad [2.13]$$

$$\text{Hata Kareleri Ortalaması (HKO)} = \frac{1}{n} \sum_{i=1}^n e_i^2 \quad [2.14]$$

Yukarıdaki ölçütlerin dışında, daha açıklayıcı seçenek (alternatif) ölçütler de önerilmiştir (13). Bu seçeneklerin en önemlileri yüzde hata ölçütleridir. Bazı yüzde hata ölçütleri şöyledir :

$$\text{Yüzde Hata (YH)} = \frac{(Y_t - F_t)}{Y_t} * 100 \quad [2.15]$$

$$\text{Ortalama Yüzde Hata (OYH)} = \frac{1}{n} \sum_{i=1}^n \frac{(Y_t - F_t)}{Y_t} * 100 \quad [2.16]$$

$$\text{Ortalama Mutlak Yüzde Hata (OMYH)} = \frac{1}{n} \sum_{i=1}^n \left| \frac{(Y_t - F_t)}{Y_t} \right| * 100 \quad [2.17]$$

Makridakis (9), ortalama mutlak yüzde hata değerinin % 5 olduğunun bilinmesinin, hata kareleri ortalamasının değerinin 183 olduğunun bilinmesinden daha anlamlı olduğunu ifade etmektedir.

2.6. Box-Jenkins Modelleri

Zaman serilerinde bir çözümleme ve kestirim yöntemi olan Box-Jenkins yöntemi; kesikli, doğrusal stokastik süreçlere dayanır. Otoregresif (*Auto Regressive-AR*), Hareketli Ortalama (*Moving Average - MA*), Otoregresif-Hareketli Ortalama (*Autoregressive-Moving Average - ARMA*) ve Bütünleşik Otoregresif-Hareketli Ortalama (*Autoregressive Integrated Moving Average - ARIMA*), Box-Jenkins kestirim modelleridir. $AR(p)$, $MA(q)$ ve bunların birleşimi olan $ARMA(p,q)$ modelleri durağan süreçlere uygulanırken, $ARIMA(p,d,q)$ modelleri durağan olmayan süreçler için kullanılmaktadır.

Zaman serileri ile kestirimde bulunabilmek için bilinen ve yaygın kullanılan istatistik yöntemlerin en kapsamlısı Box-Jenkins modelleridir.

Box-Jenkins yaklaşımı ile bir zaman serisini modelleyebilmek için ilgili serinin durağan olması gerekir. Zaman serisinin ortalama ve varyansında sistematik bir değişme yok ise serinin durağan olduğu söylenebilir. Bir seri sabit bir büyüme örüntüsü ya da genelde bir trend gösteriyorsa ya da bir seviyeden bir başka seviyeye geri dönüyor veya ilerliyor ise, bu yapıdaki seriler durağan bir seriye dönüştürülmeden modellenemezler. Eğer sürecin özellikleri zaman orjininin değişmesinden etkilenmiyorsa bu tür süreçler tam durağan süreçler olarak adlandırılır. Bunun anlamı şudur : $t_1, t_2, \dots, t_m$ zamanlarında gerçekleşen m gözlem değeri ile $t_{1+k}, t_{2+k}, \dots, t_{m+k}$ zamanlarında gerçekleşen m gözlem değerinin ortak olasılık fonksiyonu aynı ise bu tür seriler tam durağan seri olarak adlandırılır (8).

Zaman serilerinin genelde durağan özelliğe sahip olmadığı bilinmektedir. Zaman serisinin ortalaması ve varyansında zamana bağlı bir trendin olup olmadığını belirlemede üzerinde uzlaşmış bir yöntem yoktur, bu nedenle verilerin grafiğini incelemenin en kolay yol olduğunu belirtmektedirler.

Zaman serisinin sabit bir ortalama etrafında dağılmaması veya stokastik sürecin özelliklerinin zamana bağlı olarak değişmesi nedeni ile durağan olmayan seriler ortaya çıkmaktadır. Böyle serilerin modellenebilmesi için durağanlaştırılmaları gerekmektedir. Durağanlaştırma genelde fark alma işlemi ile sağlanır. Fark alma işlemi seri durağanlaşmaya kadar yapılır. Uygulamada genelde 1 veya 2 defa fark alma işlemi gerçekleştirilir. Varyansta durağanlığı sağlamak için de serinin logaritması alınabilir (10).

2.6.1. Box-Jenkins Modelinin Kurulumu

Box-Jenkins tekniğinde amaç, zaman serisine en iyi uyan ve en az parametre içeren doğrusal stokastik süreç modelini elde etmektir. Box-Jenkins modelinin kurulması dört aşamada özetlenebilir:

1. Model tanımlama aşaması: geçmiş verilere göre model parametreleri belirlenir.
2. Model parametreleri kestirim aşaması: model parametrelerinin kestirimi yapılır. Hata kareleri toplamını en az yapan parametre değerleri elde edilir.
3. Model geçerlilik aşaması: parametreleri kestirilen modelin verilere uygun olup olmadığının testi yapılır. Seçenek (alternatif) modeller önerilir.
4. Model kestirim aşaması: geçerliliği sınanan model ile, zaman serisinin gelecek dönem değerleri kestirilir ve bu değerlere ilişkin güven aralıkları bulunur.

Zaman serisi için en uygun model geçici modeller arasından seçilmektedir. Geçici modellerin parametre kestirimleri ve uygunluk testleri uzun işlemler gerektirse de bilgisayar yardımı ile bu işlemler kolayca yapılabilmektedir.

Model belirleme aşamasında, zaman serisinin otokovaryans ve otokorelasyon fonksiyonlarına başvurulur. Otokovaryans ve otokorelasyon fonksiyonları, sürecin AR(p), MA(q), ARMA(p,q) ya da ARIMA(p,d,q) modellerinden hangisine uyduğunun tespitinde önemlidir (8).

2.6.2. Otokovaryans ve Otokorelasyon Fonksiyonları

Zaman serisini oluşturan sürecin kesin bir tanımının yapılabilmesi için sürecin olasılık dağılımının tanımlanması gerekir. Ancak genellikle bu mümkün olmamaktadır. Bu durumda sürecin özelliklerini saptayabilmek için otokovaryans fonksiyonu, otokorelasyon fonksiyonu ve kısmi otokorelasyon fonksiyonu gibi araçlardan yararlanılır.

Bir zaman serisinde k dönem uzaklıktaki gözlem değerleri arasındaki kovaryansa k gecikmeli otokovaryans, korelasyon ve korelasyon katsayısına da, otokorelasyon ve otokorelasyon katsayısı denir (2).

Zaman serisinin analizi sırasında otokovaryans fonksiyonu (OKF) örnek momentleri kullanılarak kestirim edilmektedir. İki rasgele değişken arasındaki kovaryansın genel gösterimi,

$$Kov(x, y) = E[(x - E(x))(y - E(y))] \quad [2.18]$$

şeklinde dir. Benzer şekilde stokastik sürecin aralarında k dönem bulunan, y_t, y_{t-k} gibi değerleri için otokovaryansı,

$$Y_k = Kov(y_t, y_{t-k}) = E[(y_t - E(y_t))(y_{t-k} - E(y_{t-k}))] = E[(y_t - \mu)(y_{t-k} - \mu)] \quad [2.19]$$

biçiminde gösterilir. Y_k , otokovaryans fonksiyonunu ifade etmektedir (10).

Otokorelasyonlar, farklı zamanlardaki gözlemler arasındaki doğrusal ilişkiyi gösterir. Genel kavram olarak iki veri seti arasındaki korelasyonun ölçülmesi işlemi, zaman serisi otokorelasyonunun ölçülmesi için temel oluşturur. Bir seride herhangi

bir k gecikme değeri için otokorelasyon değerinin hesaplanması işlemi, orijinal seri ile orijinal seri k zaman değeri kadar taşınarak elde edilen yeni seri arasındaki korelasyon değerinin hesaplanmasıdır.

Gecikme değeri k 'nın fonksiyonu olarak elde edilen otokorelasyon katsayıları (OKK), otokorelasyon fonksiyonu (OKF) olarak adlandırılmakta ve sürecin bir değerinin ilişkili olduğu daha önceki değerlerinin yayılımını ölçerek sürecin geçmiş dönemlerle doğrusal ilişkisinin gücünü göstermektedir (10).

Gecikme değeri k 'ya bağlı olarak otokorelasyon katsayısı aşağıdaki gibi tanımlanmaktadır :

$$\rho_k = \frac{E[(y_t - \mu)(y_{t-k} - \mu)]}{\sqrt{E[(y_t - \mu)^2] E[(y_{t-k} - \mu)^2]}} \quad [2.20]$$

Durağan bir süreç için varyans, $t-k$ ve t zamanları için eşit olduğundan, otokorelasyon katsayısı,

$$\rho_k = \frac{E[(y_t - \mu)(y_{t-k} - \mu)]}{\sigma_y^2} \quad [2.21]$$

şeklinde yazılabilir. Böylece, k gecikmesi için otokorelasyon,

$$\rho_k = \frac{\gamma_k}{\gamma_0} \quad [2.22]$$

eşitliği ile ifade edilir. Yukarıdaki eşitliğe göre, $\rho_0 = 1$ olmaktadır.

Otokovaryans ve otokorelasyonlarla ilgili olarak,

$$\gamma_k = \gamma_{-k} \quad [2.23]$$

ve

$$\rho_k = \rho_{-k} \quad [2.24]$$

eşitlikleri, otokovaryans ve otokorelasyonların simetrik olduğunu göstermektedir.

OKF zaman serisinin durağan olup olmadığının belirlenmesi amacı ile kullanılmaktadır. Gecikme uzunluğu k arttıkça, serinin örnek otokorelasyon fonksiyonu (ÖOKF) hızla azalıyor ve kısa gecikmelerde eksenini kesiyor ise serinin durağan olduğuna, serinin ÖOKF yavaşça azalıyor ve uzun gecikmelerde eksenini kesiyorsa serinin durağan olmadığına karar verilir (10).

2.6.3. Otokovaryans ve Otokorelasyon Fonksiyonlarının Kestirimi

k gecikmeli otokovaryans kestirimi c_k ile gösterilirse, c_k ,

$$c_k = \frac{1}{N} \sum_{t=1}^{N-k} (y_t - \bar{y})(y_{t+k} - \bar{y}) \quad k=0,1,2,\dots,k \quad [2.25]$$

eşitliği ile hesaplanır. Benzer şekilde otokorelasyon fonksiyonu ρ_k 'nin kestirimi r_k ile gösterildiğinde, r_k ,

$$r_k = \frac{c_k}{c_0} \quad [2.26]$$

eşitliği ile elde edilir. Burada c_0 sürecin varyansı olan σ_y^2 'nin kestirimidir.

Uygulamalarda otokorelasyonu virgülden iki basamak sonra yuvarlamak yeterlidir. Otokorelasyon fonksiyonunun kestiriminde güvenilir sonuçlar elde etmek için en az 50 gözleme ihtiyaç vardır ve ayrıca r_k otokorelasyonların kestiriminde en fazla $N/4$ gecikmeye kadar hesaplama yapılmalıdır (8).

2.6.4. Örnek Otokorelasyonları Standart Hatası

Bir zaman serisi için uygun modelin belirlenebilmesi otokorelasyon fonksiyonunun belirli bir gecikme değerinde kesilip kesilmediğinin bilinmesine bağlıdır. Örnek otokorelasyonu standart hatadan iki kat daha büyükse, %95 güven düzeyinde anlamlı olduğu söylenir. Standart hatanın ± 2 katı, örnek otokorelasyonu için güven aralığı olarak tanımlanır (8).

Otokorelasyon katsayılarının analizi yapılırken katsayıların istatistiksel anlamlılıklarının da sınanması gereklidir. Örnek otokorelasyon katsayıları için,

$$\begin{aligned} H_0 : \rho_k &= 0 \\ H_1 : \rho_k &\neq 0 \end{aligned} \quad [2.27]$$

hipotezi test edilir.

Durağan normal dağılmış bir süreç için örnek otokorelasyonun kestiriminin (r_k) varyansı için, teorik otokorelasyon fonksiyonunun kesildiği q değerinden büyük k gecikmeleri için Barlett'in yaklaşımı Eş.2.28'de verilmiştir (8).

$$Var[r_k] \cong \frac{1}{N} \left\{ 1 + 2 \sum_{j=1}^q r_j^2 \right\}, \quad k > q \quad [2.28]$$

Testin karar aşamasında izlenecek yaklaşım şöyledir :

1. % 95 güven aralığında geçerli olan yaklaşık 2 standart sapma için, hesaplanan ρ_k değerinin $|r_k| > \frac{2}{\sqrt{N}}$ olması durumunda, k .derece otokorelasyon katsayısının sıfırdan önemli derecede farklı olduğuna karar verilir.
2. $r_k < \frac{2}{\sqrt{N}}$ olduğunda ise, k .derece otokorelasyon katsayısının sıfırdan farklı olmadığına karar verilir.

Otokorelasyon katsayılarının istatistiksel açıdan farklı olup, olmadığının testinde t istatistiği de kullanılmaktadır. Yukarıdaki hipotezler için, hesaplanan t değeri 2 değeri ile karşılaştırılmaktadır. t istatistiği aşağıdaki eşitlikle hesaplanmaktadır.

$$t = \frac{r_k}{\frac{1}{\sqrt{N}} \sqrt{1 + 2 \sum_{j=1}^{k-1} r_j^2}} \quad [2.29]$$

Eş. 2.29'da k gecikme sayısını, N gözlem sayısını göstermektedir. $|t| < 2$ ise, otokorelasyonun istatistiksel olarak önemli olmadığına yani $\rho_k = 0$ olduğuna karar verilir (10).

Otokorelasyonlar için güven aralıkları, sıfırdan farklı örnek otokorelasyonlarının göz önünde bulundurulmaması ve anlamlı sayılması kararının verilmesinde kabaca yol gösterir (8).

2.6.5. Kısmi Otokorelasyonlar

Kısmi otokorelasyonlar, otokorelasyonlar gibi seri değerleri arasındaki ilişkiyi değerlendirmek amacı ile kullanılan bir başka istatistiksel ölçü kümesidir. Kısmi otokorelasyonlar, serinin hareketli ortalama ya da otoregresif süreçten hangisine uyduğunu anlamada otokorelasyonların tamamlayıcısıdır.

Bir zaman serisinde Y_t ile Y_{t-k} arasındaki korelasyonun büyük bir kısmının, bu değişkenlerin arasındaki korelasyonun Y_{t-1} , Y_{t-2} , ..., Y_{t-k+1} gecikmelerine sahip olması nedeni ile olduğu belirtilmektedir. Bu korelasyonları düzeltmek amacı ile hesaplanan kısmi otokorelasyon katsayıları, durağan bir değişkenin t ve t-k gibi iki farklı dönemde birbirleri ile olan ilişkisini, yani Y_t ve Y_{t-k} arasındaki ilişkiyi, bu zaman dönemleri arasında kalan diğer tüm dönemlerdeki t-1, t-2, gibi gecikmeleri dışlayarak veya sabit tutarak ortaya koymaktadır. Kısmi otokorelasyonlar da, otokorelasyonlar gibi Yule-Walker denklem sisteminin çözümünden elde edilirler ve literatürde ϕ_{kk} ile gösterilirler. Örnek kısmi otokorelasyonu varyansı Eş.2.30'da verilmiştir (10).

$$\text{var} \left[\hat{\phi}_{kk} \right] \cong \frac{1}{N} , \quad [2.30]$$

Bir AR(p) modelinde, $k < p$ olduğu durumda kısmi otokorelasyonlar sıfırdan farklı ve $k > p$ olduğu durumda da kısmi otokorelasyonlar sıfıra eşittir. Diğer yandan, bir MA sürecinde kısmi otokorelasyonlar bütün gecikme değerleri için sıfırdan farklıdır (8).

Yule-Walker denklem sisteminin çözülmesinden kısmi otokorelasyonların $k=1, 2$ ve 3 değerleri için Eş. 2.31 elde edilir.

$$\begin{aligned}\phi_{11} &= \rho_1 \\ \phi_{22} &= \frac{\begin{vmatrix} 1 & \rho_1 \\ \rho_1 & \rho_2 \end{vmatrix}}{\begin{vmatrix} 1 & \rho_1 \\ \rho_1 & 1 \end{vmatrix}} = \frac{\rho_2 - \rho_1^2}{1 - \rho_1^2} \\ \phi_{33} &= \frac{\begin{vmatrix} 1 & \rho_1 & \rho_2 \\ \rho_1 & 1 & \rho_3 \\ \rho_2 & \rho_3 & \rho_1 \end{vmatrix}}{\begin{vmatrix} 1 & \rho_1 & \rho_2 \\ \rho_1 & 1 & \rho_3 \\ \rho_2 & \rho_3 & 1 \end{vmatrix}} \quad [2.31]\end{aligned}$$

2.6.6. Model Kurma

Box ve Jenkins (8), durağan zaman serileri için üç genel stokastik model sınıfı önermektedir. Bunlar otoregresif (autoregressive-AR), hareketli ortalama (moving average-MA) ve bu iki sürecin karışımı otoregresif hareketli ortalama (autoregressive moving average-ARMA) stokastik süreç modelleridir.

AR modelleri ilk defa 1926 yılında Yule tarafından önerilmiş ve daha sonra 1931'de Walker tarafından geliştirilmiştir. MA modelleri ise ilk defa 1937 yılında Slutsky tarafından kullanılmıştır (13).

2.6.7. Otoregresif (Auto Regressive-AR) Modeller

AR (p) modelinde Y_t değeri, serinin p dönem geçmiş değerlerinin ağırlıklı toplamının ve rasgele hata teriminin doğrusal fonksiyonudur. AR (p) modelleri genel olarak aşağıdaki gibi gösterilir :

$$Y_t = \mu + \phi_1 Y_{t-1} + \dots + \phi_p Y_{t-p} + a_t \quad [2.32]$$

Serinin ortalaması μ , seri değerlerinden çıkarıldığında elde edilen yeni seri için AR(p) modeli, Eş. 2.33'teki gibidir.

$$y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + a_t \quad [2.33]$$

Eş.2.33'te, y_{t-1} , y_{t-2} , y_{t-p} geçmiş gözlem değerleri, ϕ_1 , ϕ_2 ,, ϕ_p geçmiş gözlem değerleri için katsayılar ve a_t de hata terimidir. Geri kaydırma işlemcisi B kullanılarak, yukarıdaki model Eş. 2.34'teki gibi ifade edilmektedir.

$$y_t = \frac{1}{\phi(B)} a_t \quad [2.34]$$

Burada $\phi(B)$,

$$\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p \quad [2.35]$$

eşitliğine sahiptir.

Otoregresif ifadenin temelindeki eşitlik regresyon eşitliğine benzemektedir. Bu eşitlikte y_t bağımlı değişken ve y_{t-1} , y_{t-2} , bağımsız değişken rolünü üstlenmektedir. Ayrıca y_{t-1} , y_{t-2} ,'ler, y_t ile aynı yapıda değişkenler olup, aralarında bir dönem zaman farkı vardır. ϕ_1 birinci dereceden otoregresif parametre, ϕ_2 ikinci dereceden otoregresif parametre ve ϕ_p p. dereceden otoregresif

parametredir. Model, $\mu, \phi_1, \phi_2, \dots, \phi_p, \sigma_a^2$ olmak üzere $p+2$ tane bilinmeyen parametre içermektedir. Uygulamada var olan verilerden bu parametrelerin kestirimleri elde edilir.

Otoregresif model, doğrusal filtreleme modelinin özel bir durumu olup, y_{t-1} Eş. 2.36'daki ifade ile yazılabilir.

$$y_{t-1} = \phi_1 y_{t-2} + \phi_2 y_{t-3} + \dots + \phi_p y_{t-p-1} + \alpha_{t-1} \quad [2.36]$$

Benzer şekilde, otoregresif model eşitliğinden $y_{t-1}, y_{t-2}, \dots$ sırasıyla modelden kaldırılıp yerlerine eşitleri konarak model yeniden yazıldığında sonsuz sayıda hata teriminden oluşan seri elde edilir. Dolayısıyla,

$$\phi(B)y_t = \alpha_t \quad [2.37]$$

eşitliği elde edilir. Buradan,

$$y_t = \psi(B)\alpha_t \quad [2.38]$$

yazılabilir. Bu eşitlikten de,

$$\psi(B) = \phi^{-1}(B) = \frac{1}{\phi(B)} \quad [2.39]$$

ifadesi elde edilir.

Böylece otoregresif süreç, girdisi a_t , çıktısı y_t olan, $\phi^{-1}(B)$ transfer fonksiyonuna sahip doğrusal filtreleme olarak düşünülebilir. Otoregresif süreç durağan veya durağan olmayan bir süreç olabilir. Sürecin durağan olabilmesi için Eş. 2.39'daki $\psi_1, \psi_2, \dots$, ağırlıklarının $\psi(B)$ yakınsak bir seri olacak şekilde seçilmesi gerekir (8).

AR(1) süreci için,

$$\psi(B) = \phi^{-1}(B) = (1 - \phi_1 B)^{-1} = \sum_{j=0}^{\infty} \phi_1^j B^j \quad [2.40]$$

eşitliği yazılabilir. AR(1) sürecinin durağan olabilmesi için $\phi(1)$ parametresinin $|\phi(1)| < 1$ ve dolayısıyla $1 - \phi(1)B = 0$ 'ın kökü $B = \phi(1)^{-1}$ koşulunu sağlaması gereklidir. Bu koşul $\phi(B) = 1 - \phi(1)B = 0$ kökünün birim çemberin dışında olması ile aynı anlamı ifade eder. $\phi(B) = 0$ eşitliği, sürecin karakteristik eşitliği olarak adlandırılır. Genel bir AR(p) süreci için karakteristik eşitlik aşağıdaki gibi yazılabilir.

$$\phi(B) = \prod_{i=1}^p (1 - G_i B) = (1 - G_1 B) (1 - G_2 B) \dots (1 - G_p B) \quad [2.41]$$

Burada $G_1^{-1}, G_2^{-1}, \dots, G_p^{-1}$ karakteristik eşitliğin kökleridir. Durağanlık koşulu şartları gereği karakteristik eşitliğin köklerinin birim çemberin dışında yer alması gerekir.

p. dereceden bir otoregresif model Eş. 2.33'te verilmiştir.

Eş.2.33, y_{t-k} ile çarpılıp beklenen değeri alınırsa, y_{t-k} 'nin $(t-k)$ 'inci zamana kadarki şokları içerdiği ve a_t ile korelasyonu olmadığı bilindiğinde Eş. 2.42 elde edilir:

$$Y_k = \phi_1 Y_{k-1} + \phi_2 Y_{k-2} + \dots + \phi_p Y_{k-p} \quad k > 0 \quad [2.42]$$

Eş. 2.42'deki ifade γ_0 ile bölünecek olursa otokorelasyon fonksiyonu 2.43'deki eşitliği sağlayacaktır.

$$\rho_k = \phi_1 \rho_{k-1} + \phi_2 \rho_{k-2} + \dots + \phi_p \rho_{k-p} \quad k > 0 \quad [2.43]$$

Bu da $\phi(B)\rho_k = 0$ olarak yazılabilir. Burada $\phi(B)$,

$$\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p \quad [2.44]$$

eşitliğine sahiptir.

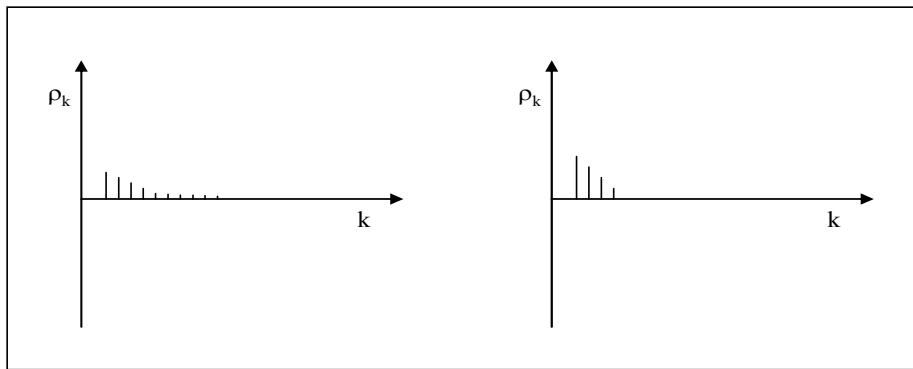
Genel çözümde aşağıdaki gibidir :

$$\rho_k = A_1 G_1^k + A_2 G_2^k + \dots + A_p G_p^k \quad [2.45]$$

Durağanlık için karakteristik eşitliğin köklerinin birim çemberin dışında yer alması gereklidir. Eğer kökler farklı ise iki farklı durumla karşılaşılabilir :

1. Kök G_i gerçektir ve k arttıkça sifıra doğru azalan bir $A_i G_i^k$ terimi bulunur.
2. Kök çiftlerinden biri komplekstik ve sinus fonksiyonunu takip eden bir terim eklenir.

Genel olarak bir durağan otoregresif sürecin otokorelasyon fonksiyonu üstel ve sinus fonksiyonunun bileşiminden oluşacaktır (8). Bir $AR(p)$ modeli için otokorelasyon fonksiyonu üstel olarak azalan ya da sinüs eğrisi şeklinde bir yapı gösterirken, kısmi otokorelasyon fonksiyonu da p gecikmeden sonra kesilir. Şekil 2.2, bu durumu göstermektedir.



Şekil 2.2. $AR(p)$ süreci otokorelasyon ve kısmi otokorelasyon fonksiyonu (10)

2.6.7.1. Yule-Walker Eşitliği Cinsinden Otoregresif Parametreler

Eş.2.43'te ρ_k ($k=1,2,\dots,p$) yerine konulursa, $\rho_1, \rho_2, \dots, \rho_p$ cinsinden $\phi_1, \phi_2, \dots, \phi_p$ için p tane doğrusal eşitlik elde edilir. Bu p tane eşitlik Yule-Walker eşitliği olarak bilinir :

$$\begin{array}{ccccccc} \rho_1 & = & \phi_1 & + & \phi_1 \rho_1 & + & \dots & + & \phi_1 \rho_{p-1} \\ \rho_2 & = & \phi_1 \rho_1 & + & \phi_2 & + & \dots & + & \phi_p \rho_{p-2} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ \rho_p & = & \phi_1 \rho_{p-1} & + & \phi_2 \rho_{p-2} & + & \dots & + & \phi_p \end{array} \quad [2.46]$$

Yule-Walker denklemlerinde parametrelerin kestirimi, r_k kestirimi otokorelasyonlar ile ρ_k teorik otokorelasyonlar yer değiştirerek elde edilir (8).

2.6.7.2. AR(1) Süreci

AR (1) modelinde y_t değeri, 1 dönem geçmiş gözlem değerinin ve rasgele hata teriminin doğrusal fonksiyonudur. Birinci dereceden otoregresif süreç aşağıdaki biçimdedir.

$$\begin{aligned} y_t &= \phi_1 y_{t-1} + a_t \\ &= a_1 + \phi_1 a_{t-1} + \phi_1^2 a_{t-2} + \dots \end{aligned} \quad [2.47]$$

Sürecin durağan olması için eşitliğin köklerinin birim çemberin dışında kalması gerekmektedir. Bu da ϕ_1 parametresinin $-1 < \phi_1 < 1$ şartını sağlamasını gerektirmektedir. Eş.2.43 kullanılarak,

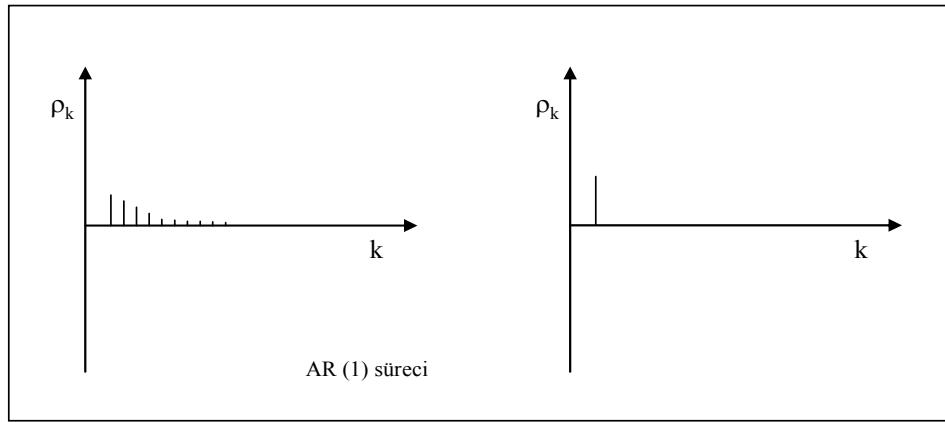
$$\rho_k = \phi_1 \rho_{k-1} \quad k > 0 \quad [2.48]$$

elde edilir.

$\rho = 1$ olduğundan,

$$\rho_k = \phi^k_1 \quad k \geq 0 \quad [2.49]$$

olur. Eş. 2.49, AR(1) sürecinde teorik otokorelasyon fonksiyonunun sıfıra doğru üstel olarak azaldığını göstermektedir. Şekil 2.3'te otokorelasyon fonksiyonunun üstel olarak azaldığı ve kısmi otokorelasyon fonksiyonunun $k=1$ gecikmeden sonra sıfır olduğu görülmektedir.



Şekil 2.3. AR(1) süreci otokorelasyon ve kısmi otokorelasyon fonksiyonu (10)

Sürecin varyansı aşağıdaki eşitlikle hesaplanır :

$$\sigma_y^2 = \frac{\sigma_a^2}{1 - \phi_1^2} \quad [2.50]$$

2.6.7.3. AR(2) Süreci

AR(2) modelinde y_t değeri, serinin 2 dönem geçmiş gözlem değerlerinin ağırlıklı toplamının ve rasgele hata teriminin doğrusal fonksiyonudur. İkinci dereceden otoregresif süreç aşağıdaki biçimdedir :

$$y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + a_t \quad [2.51]$$

Sürecin karakteristik eşitliği, Eş.2.52’de verilmiştir.

$$\phi(B)) 1 - \phi_1 B - \phi_2 B^2 = 0 \quad [2.52]$$

Sürecin durağan olması, karakteristik eşitliğin köklerinin birim çemberin dışında olmasına bağlıdır. Dolayısıyla ϕ_1 ve ϕ_2 ’nin aşağıdaki şartları sağlaması gerekmektedir.

$$\begin{aligned} \phi_1 + \phi_2 &< 1 \\ \phi_2 - \phi_1 &< 1 \\ -1 &< \phi_2 < 1 \end{aligned} \quad [2.53]$$

Eş.2.43, kullanılarak $\rho_k = \phi_1 \rho_{k-1} + \phi_2 \rho_{k-2}$ $k > 0$, elde edilir.

$$\rho_0 = 1, \quad [2.54]$$

$$\rho_1 = \frac{\phi_1}{1 - \phi_2}, \quad [2.55]$$

$$\begin{aligned} \rho_k &= A_1 G_1^k + A_2 G_2^k \\ &= \frac{G_1(1 - G_2^2)G_1^k - G_2(1 - G_1^2)G_2^k}{(G_1 - G_2)(1 + G_1 G_2)} \end{aligned} \quad [2.56]$$

G_1^{-1} ve G_2^{-1} karakteristik eşitliğin kökleridir.

ϕ_1 ve ϕ_2 parametreleri için Yule-Walker denklem sistemi çözülürse,

$$\phi_1 = \frac{\rho_1(1 - \rho_2)}{1 - \rho_1^2} \quad [2.57]$$

$$\phi_2 = \frac{\rho_2 - \rho_1^2}{1 - \rho_1^2}$$

elde edilir.

Eş.2.57 ϕ_1 ve ϕ_2 otoregresif parametrelerin kestirimini, r_1 ve r_2 kestirimi otokorelasyon katsayılarından elde etmek için kullanılabilir.

AR(2) süreci Yule-Walker denklem sistemi, otokorelasyonları otoregresif parametreler terimleri ile ifade edilmek için çözülebilir :

$$\rho_1 = \frac{\phi_1}{1 - \phi_2} \quad [2.58]$$

$$\rho_2 = \phi_2 + \frac{\phi_1^2}{1 - \phi_2}$$

AR(2) sürecinin durağan olabilmesi için otokorelasyonların Eş.2.59'daki şartları sağlaması gerekir :

$$-1 < \rho_1 < 1$$

$$-1 < \rho_2 < 1 \quad [2.59]$$

$$\rho_1^2 < \frac{1}{2} (1 + \rho_2)$$

Sürecin varyansı,

$$\begin{aligned} \sigma_y^2 &= \frac{\sigma_a^2}{1 - \rho_1\phi_1 - \rho_2\phi_{21}} \\ &= \left(\frac{1 - \phi_2}{1 + \phi_2} \right) \frac{\sigma_a^2}{\{(1 - \phi_2)^2 - \phi_1^2\}} \end{aligned} \quad [2.60]$$

eşitliği ile tanımlanır (8).

Şekil 2.4, AR(2) süreci otokorelasyon ve kısmi otokorelasyonları için örnekler göstermektedir.

2.6.8. Hareketli Ortalama (Moving Average-MA) Modelleri

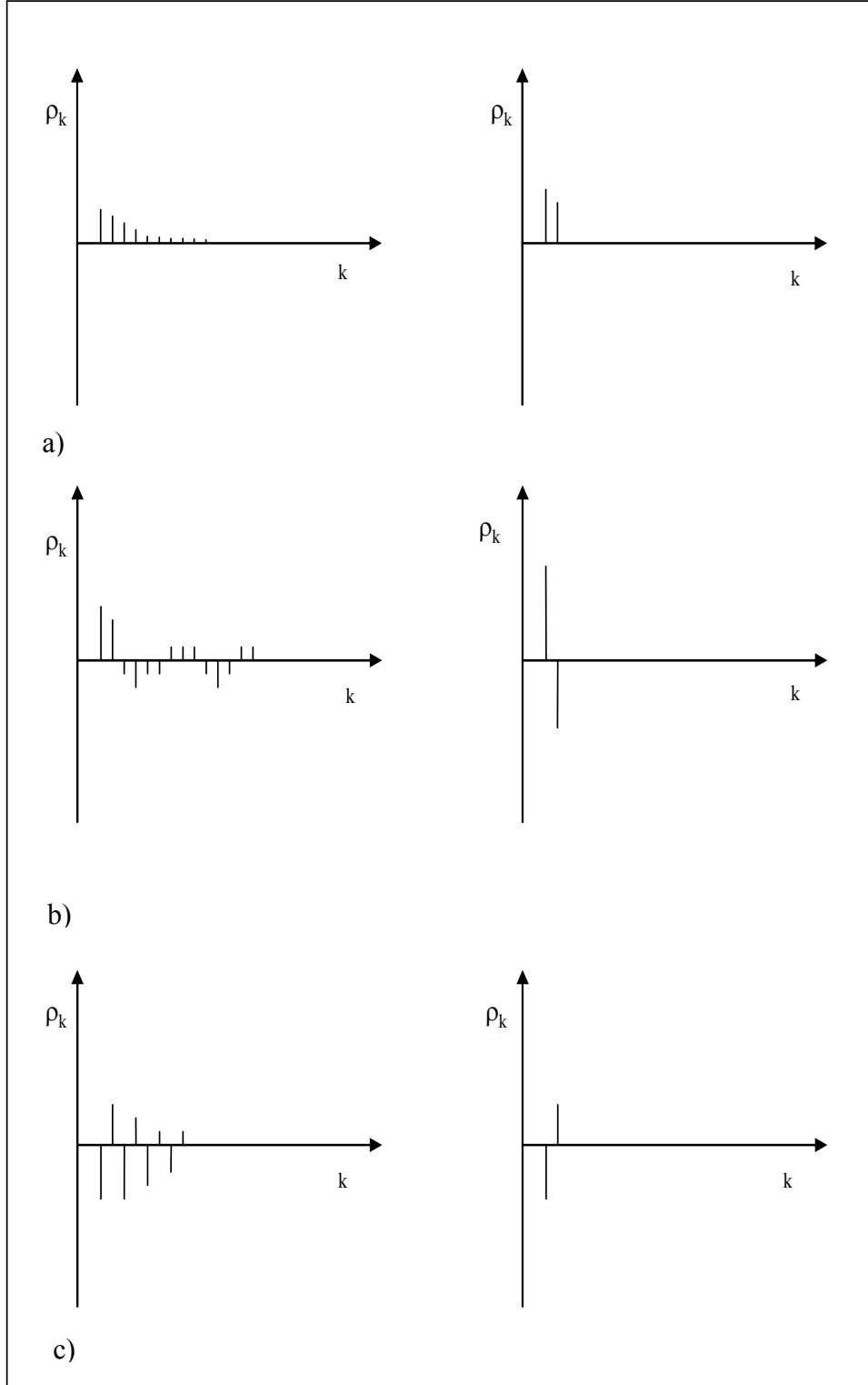
MA(q) modelinde y_t değeri, serinin geriye doğru q dönem geçmiş hata terimlerinin doğrusal fonksiyonudur. MA(q) modelleri genel olarak aşağıdaki gibi gösterilir.

$$y_t = a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \dots - \theta_q a_{t-q} \quad [2.61]$$

Burada $a_t, a_{t-1}, a_{t-2}, \dots, a_{t-q}$ hata terimlerini $\theta_1, \theta_2, \dots, \theta_q$ hata terimleri ile ilgili katsayıları göstermektedir. Hareketli ortalama işlemcisi ile model,

$$y_t = \theta(B)a_t \quad [2.62]$$

şeklinde gösterilir.



Şekil 2.4. AR(2) modeli otokorelasyon ve kısmi otokorelasyon fonksiyonları (10)

Hareketli ortalama işlemcisi,

$$\theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q \quad [2.63]$$

eşitliği ile tanımlanmaktadır.

Hareketli ortalama kavramı, seri değerlerinin geçmiş hata terimlerinin doğrusal bir fonksiyonu olduğunu belirtmektedir. Derecesi q olan bir hareketli ortalama modelinde, herhangi bir seri değeri, geriye doğru q dönem hata terimlerinin doğrusal bileşimine eşittir. $\theta_1, \theta_2, \dots, \theta_q$, sırasıyla 1, 2, $\dots$, q derece hareketli ortalama parametrelerini göstermektedir. Model $\mu, \theta_1, \theta_2, \dots, \theta_q, \sigma_a^2$ olmak üzere q+2 adet bilinmeyen parametre içermektedir. Bu parametrelerin kestirimleri var olan verilerden elde edilmektedir.

Genel olarak Eş.2.62'deki gibi gösterilen süreç için,

$$\pi(B) y_t = a_t \quad [2.64]$$

eşitliği yazılabilir.

$$\pi(B) = \theta^{-1}(B) = \frac{1}{\theta(B)} \quad [2.65]$$

olduğu açıktır.

Doğrusal bir sürecin çevrilebilir olması için $\pi(B)$ serisinin, $|B| \leq 1$ için yakınsak olması gereklidir. MA(q) süreci için çevrilebilirlik koşulu karakteristik eşitliğin kökleri birim çemberin dışında olmalıdır.

$$\theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q = 0 \quad [2.66]$$

$\psi(B) = \theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q$, serisi sonlu ise MA süreci durağandır.

Sürecin varyansı,

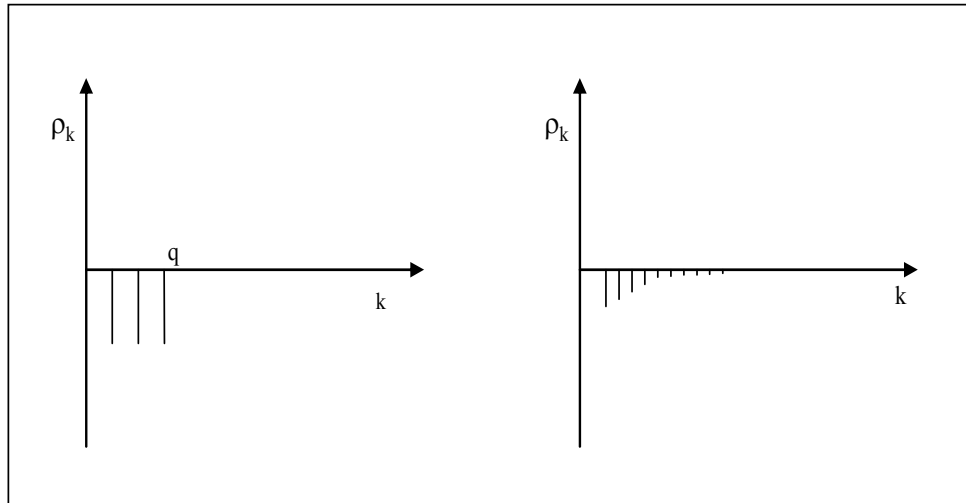
$$y_0 = (1 + \theta_1^2 + \theta_2^2 + \dots + \theta_q^2) \sigma_a^2 \quad [2.67]$$

ve otokorelasyon fonksiyonu,

$$\rho_k = \begin{cases} \frac{-\theta_k + \theta_1 \theta_{k+1} + \dots + \theta_{q+k} \theta_q}{1 + \theta_1^2 + \dots + \theta_q^2} & k = 1, 2, \dots, q \\ 0 & k > q \end{cases} \quad [2.68]$$

şeklinde tanımlanır.

MA(q) sürecinde otokorelasyon fonksiyonu q gecikmeden sonra kesilirken, kısmi otokorelasyon fonksiyonu üstel olarak azalır ya da sinüs eğrisi şeklinde bir yapı gösterir. Şekil 2.5, MA(q) süreci otokorelasyon ve kısmi otokorelasyon fonksiyonları için bir örnek göstermektedir.



Şekil 2.5. MA(q) modeli otokorelasyon ve kısmi otokorelasyon fonksiyonu (10)

2.6.8.1. MA(1) Süreci

MA(1) modelinde y_t değeri, serinin geriye doğru 1 dönem geçmiş hata teriminin doğrusal fonksiyonudur. MA(1) süreci, aşağıdaki gibi gösterilir :

$$\begin{aligned} y_t &= a_t - \theta_1 a_{t-1} \\ &= (1 - \theta_1 B)a_t \end{aligned} \quad [2.69]$$

Sürecin karakteristik eşitliği,

$$\theta(B) = 1 - \theta_1 B = 0. \quad [2.70]$$

Sürecin çevrilebilir olması için, karakteristik eşitliğin köklerinin birim çemberin dışında kalması gerekir. Dolayısıyla θ_1 parametresinin,

$$-1 < \theta_1 < 1 \quad [2.71]$$

şartını sağlaması gerekir. MA modellerinde tüm θ_i 'ler için süreç durağandır. MA(1) sürecinin varyansı,

$$\gamma_0 = (1 + \theta_1^2) \sigma_a^2 \quad [2.72]$$

ve otokorelasyon fonksiyonu,

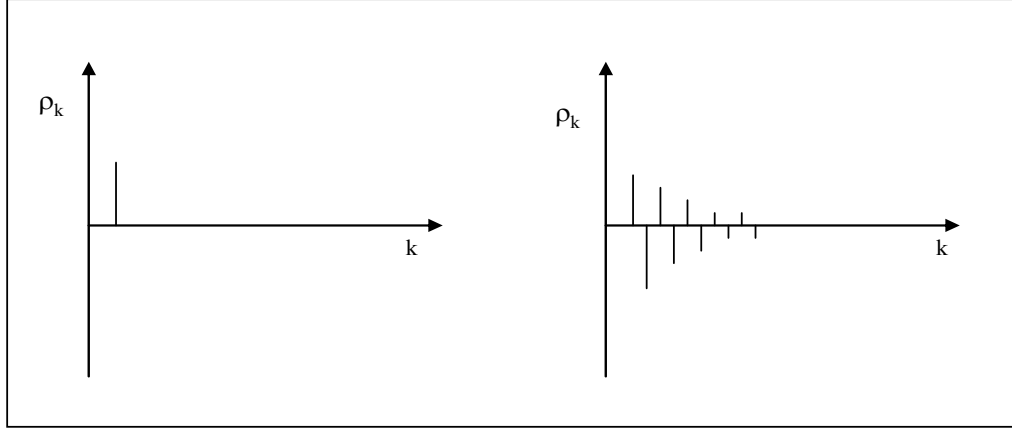
$$\rho_k = \begin{cases} \frac{-\theta_1}{1 + \theta_1^2} & k = 1 \\ 0 & k > 1 \end{cases} \quad [2.73]$$

eşitliği ile elde edilir.

Kısmi otokorelasyon fonksiyonu ise Eş.2.74'te verilmiştir.

$$\phi_{kk} = -\theta_1^k \frac{(1-\theta_1^2)}{(1-\theta_1^{2(k+1)})} \quad [2.74]$$

Şekil 2.6 MA(1) süreci otokorelasyon ve kısmi otokorelasyon fonksiyonunu göstermektedir.



Şekil 2.6. MA(1) süreci otokorelasyon ve kısmi otokorelasyon fonksiyonu (10)

2.6.8.2. MA(2) Süreci

MA(2) modelinde y_t değeri, serinin geriye doğru 2 dönem geçmiş hata teriminin doğrusal fonksiyonudur. MA(2) süreci, Eş.2.75'teki gibi gösterilir :

$$y_t = a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} \quad [2.75]$$

Sürecin karakteristik eşitliği,

$$\theta(B) = 1 - \theta_1 B - \theta_2 B^2 = 0 \quad [2.76]$$

denklemleri ile belirtilir.

Sürecin çevrilebilir olması için, karakteristik eşitliğin köklerinin birim çemberin dışında kalması gereklidir. Dolayısıyla θ_1 , θ_2 parametreleri aşağıdaki şartları sağlamalıdır :

$$\begin{aligned}\theta_2 + \theta_1 &< 1 \\ \theta_2 - \theta_1 &< 1 \\ -1 &< \theta_2 < 1\end{aligned}\tag{2.77}$$

Sürecin varyansı,

$$\gamma_0 = \sigma_a^2 (1 + \theta_1^2 + \theta_2^2)\tag{2.78}$$

eşitliği ile, otokorelasyon fonksiyonu

$$\begin{aligned}\rho_1 &= \frac{-\theta_1 (1 - \theta_2)}{1 + \theta_1^2 + \theta_2^2} \\ \rho_2 &= \frac{-\theta_2}{1 + \theta_1^2 + \theta_2^2} \\ \rho_k &= 0 \quad k > 2\end{aligned}\tag{2.79}$$

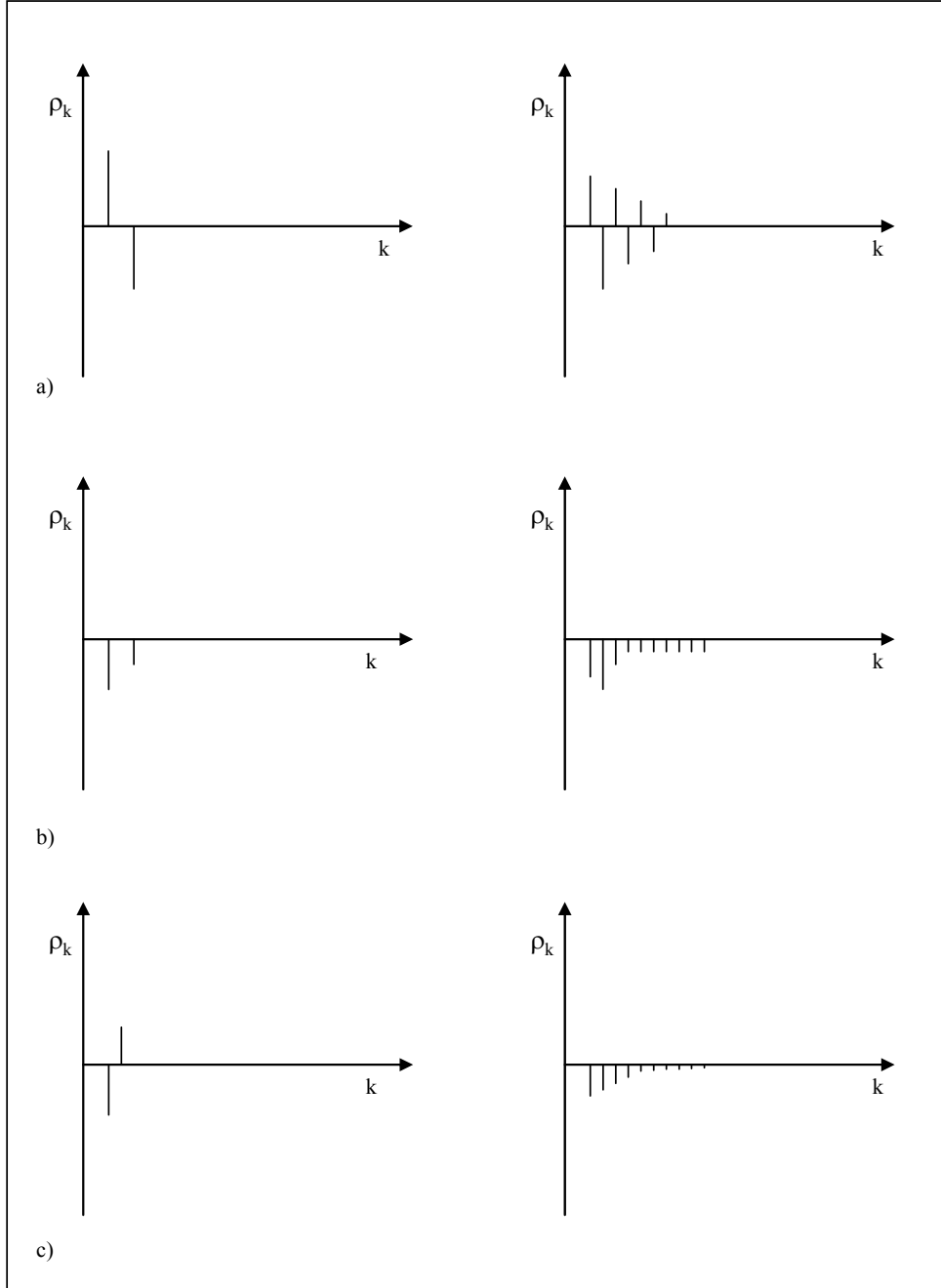
eşitlikleri ile elde edilir. Şekil 2.7, MA(2) süreci için örnekler göstermektedir.

2.6.9. Karışık (ARMA) Modeller

Birinci dereceden hareketli ortalama modeli $y_t = (1 - \theta B)a_t$ aşağıdaki şekilde yazılabilir :

$$\begin{aligned}\frac{1}{(1 - \theta B)} y_t &= a_t \\ (1 + \theta B + \theta^2 B^2 + \theta^3 B^3 + \dots) y_t &= a_t \\ y_t &= -\theta y_{t-1} - \theta^2 y_{t-2} - \theta^3 y_{t-3} - \dots + a_t\end{aligned}\tag{2.80}$$

Böylece, gerçek MA(1) süreci, çok parametrelili otoregresif model biçimine dönüştürülebilir. AR(1) modeli de MA modeline dönüştürülürse çok parametrelili bir model elde edilir. Pratikte ise az parametrelili model elde etmek için hem otoregresif hem de hareketli ortalama terimlerinin bazı durumlarda modelde bulunması gereklidir.



Şekil 2.7. MA(2) süreci otokorelasyon ve kısmi otokorelasyon fonksiyonları (10)

Bu modeller AR ve MA parametrelerini içerip ARMA modeli olarak adlandırılır ve aşağıdaki yapıdadır :

$$y_t = \frac{\theta(B)}{\phi(B)} a_t \quad [2.81]$$

ARMA modelinin derecesi p ve q cinsinden ifade edilir. Model $p+q+2$ $(\mu, \phi_1, \phi_2, \dots, \phi_p, \theta_1, \theta_2, \dots, \theta_q, \sigma_a^2)$ parametre içerir. Pratikte p ve q'nun değerleri 2'den büyük değildir.

Bir ARMA süreci aşağıdaki gibi gösterilir.

$$y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \dots - \theta_q a_{t-q} \quad [2.82]$$

Bu da,

$$(1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p) y_t = (1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q) a_t \quad [2.83]$$

ya da,

$$\phi(B) y_t = \theta(B) a_t \quad [2.84]$$

eşitliği ile gösterilir.

Hareketli ortalama terimlerinin AR(p) sürecinin durağanlık koşulunu etkilemediğinden $\phi(B) y_t = \theta(B) a_t$ bir durağan süreci göstermektedir. Eğer $\phi(B) = 0$ denkleminin kökleri birim çemberin dışında kalıyor ise, ARMA (p,q) süreci çevrilebilirdir (8).

2.6.9.1. ARMA (1,1) Süreci

ARMA (1,1) süreci,

$$(1-\phi_1 B) y_t = (1-\theta_1 B) a_t \quad [2.85]$$

şeklinde tanımlanır. Eğer $-1 < \phi_1 < 1$ ise süreç durağan ve $-1 < \theta_1 < 1$ ise süreç çevrilebilir. Sürecin varyansı,

$$\gamma_0 = \frac{1 + \theta_1^2 - 2\phi_1\theta_1}{1 - \phi_1^2} \sigma_a^2 \quad [2.86]$$

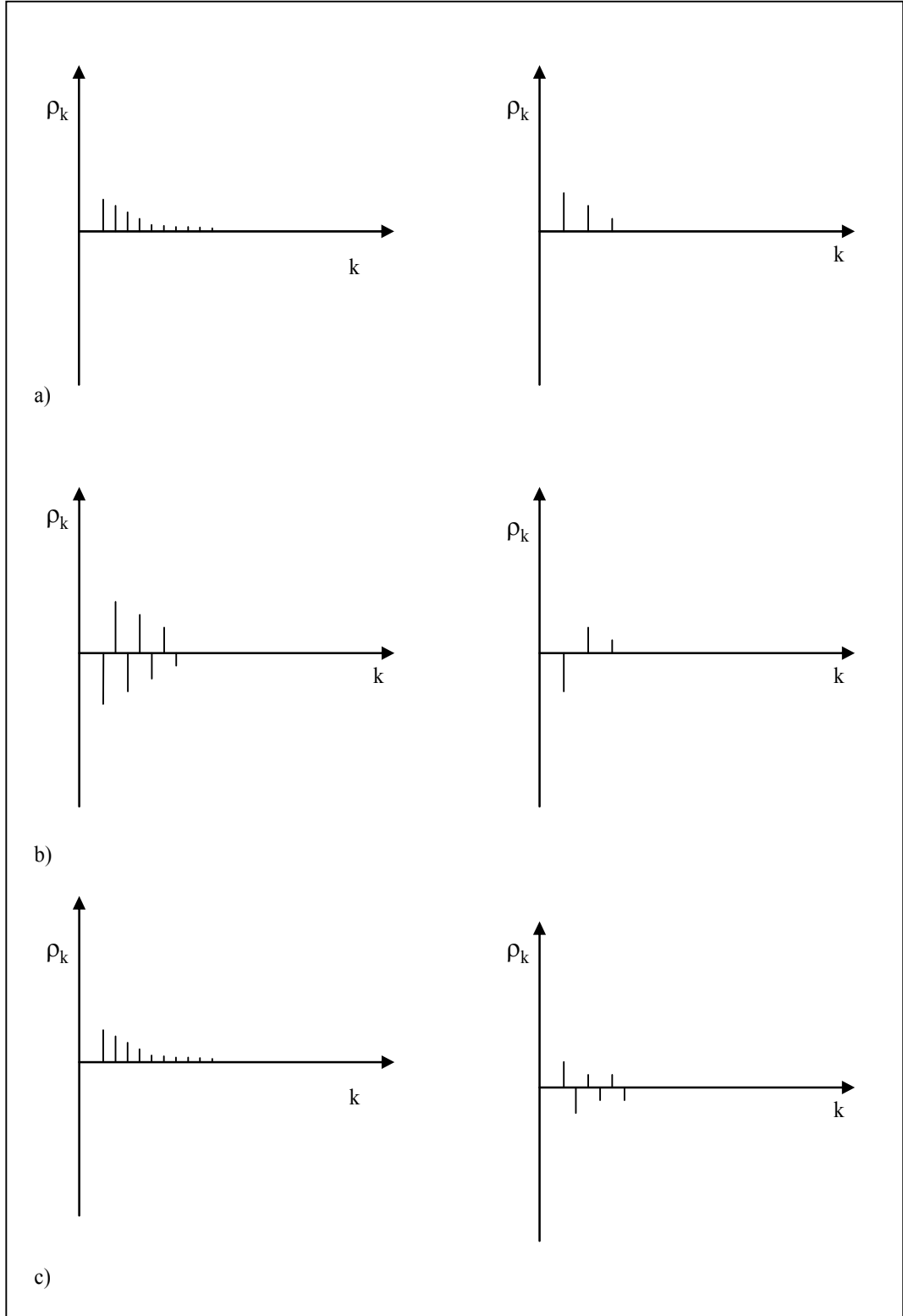
ve otokorelasyon fonksiyonu,

$$\begin{aligned} \rho_1 &= \frac{(1 - \phi_1 \theta_1)(\phi_1 - \theta_1)}{1 + \theta_1^2 - 2\phi_1\theta_1} \\ \rho_2 &= \phi_1 \rho_1 \\ \rho_k &= \phi_1 \rho_{k-1} \quad k > 2 \end{aligned} \quad [2.87]$$

eşitlikleri ile tanımlanmaktadır. Böylece, otokorelasyon fonksiyonu ρ_1 başlangıç noktasından itibaren θ_1 ve ϕ_1 değerlerine bağlı olarak üstel olarak azalır. Şekil 2.8, ARMA (1,1) sürecinin otokorelasyon ve kısmi otokorelasyon fonksiyonlarına örnekler göstermektedir.

2.6.10. Durağan Olmayan Doğrusal Stokastik Modeller

Zaman serisinin durağan olduğu durumlarda, yani sürecin ortalamasının, varyansının ve kovaryansının zamana bağlı olarak değişmediği durumlarda ARMA(p,q) veya ARMA(p,q)'nın özel hali olan AR(p) veya MA(q) modellerinden uygun olanı kullanılır.



Şekil 2.8. ARMA (1,1) otokorelasyon ve kısmi otokorelasyon fonksiyonları (10)

Ancak gerçekte zaman serilerinin ortalama ve varyansında zamana bağlı olarak bir değişim olmaktadır. Bu durum durağan olmayan durum olarak adlandırılır. Bu tip zaman serileri durağan hale dönüştürüldüğünde daha önce sözü edilen ARMA(p,q) modelleri kestirim için kullanılabilir. Zaman serisinin durağanlaştırılması ise fark almak suretiyle yapılır. Zaman serisinin doğrusal bir eğilimi var ise birinci fark serisi durağan olur. Eğer zaman serisinin eğrisel bir eğilimi var ise farkların farkı alındığında ikinci farklar serisi durağan olur. Bu durumda model, ARIMA(p,d,q) olarak ifade edilir. Burada d serinin durağanlaştırma (fark alma) parametresidir .

ARIMA(p,d,q) şeklinde gösterilen otoregresif bütünleşik hareketli ortalama modelinin genel ifadesi,

$$z_t = \phi_1 z_{t-1} + \phi_2 z_{t-2} + \dots + \phi_p z_{t-p} + a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \dots - \theta_q a_{t-q} \quad [2.88]$$

eşitliği ile verilir.

Eş. 2.88'deki model, ARMA(p,q) modelindeki y_t teriminin yerine z_t teriminin yazılmış halidir. Bu durum, durağan olmayan y_t sürecinin d derece farkı alınarak durağanlaştırılması sonucu z_t sürecinin elde edilmesinden kaynaklanmaktadır.

$$z_t = \Delta^d y_t \quad [2.89]$$

eşitliğinde,

z_t = fark alma sonucu oluşan seriyi

Δ = fark alma işlemcisini

d = fark alma derecesini

göstermektedir. Birinci farklar serisi durağan ise,

$$\Delta y_t = z_t = y_t - y_{t-1} = (1-B) y_t \quad [2.90]$$

eşitliği yazılabilir. Benzer şekilde serinin durağan hale gelmesi için d defa fark alınmışsa,

$$\Delta^d y_t = z_t = (I-B)^d y_t \quad [2.91]$$

ifadesi yazılabilir.

Bütünleşik ARMA modelleri olarak adlandırılan ARIMA modelleri, durağan olmayan serilerin durağan olana kadar kaç defa farklarının alındığını gösteren d derecesine, AR terim sayısı p ve MA terim sayısı q'nun ilave edilmesi ile belirlenmektedir (10).

ARMA(p,q) modellerinin özel durumları AR(p) ve MA(q) olabildiği gibi ARIMA(p,d,q) modellerinde de özel durum söz konusudur. Örneğin ARIMA(p,d,q) modeli hareketli ortalama parametresi içermiyorsa yani q=0 ise ARI(p,d), eğer ARIMA(p,d,q) modeli otoregresif model parametresi içermiyorsa yani p=0 ise IMA(d,q) modeli söz konusudur. Uygulamada sık kullanılan ARIMA modelleri, IMA(1,1) ve ARI(1,1) ya da ARIMA(1, 1,1) modelleridir (2).

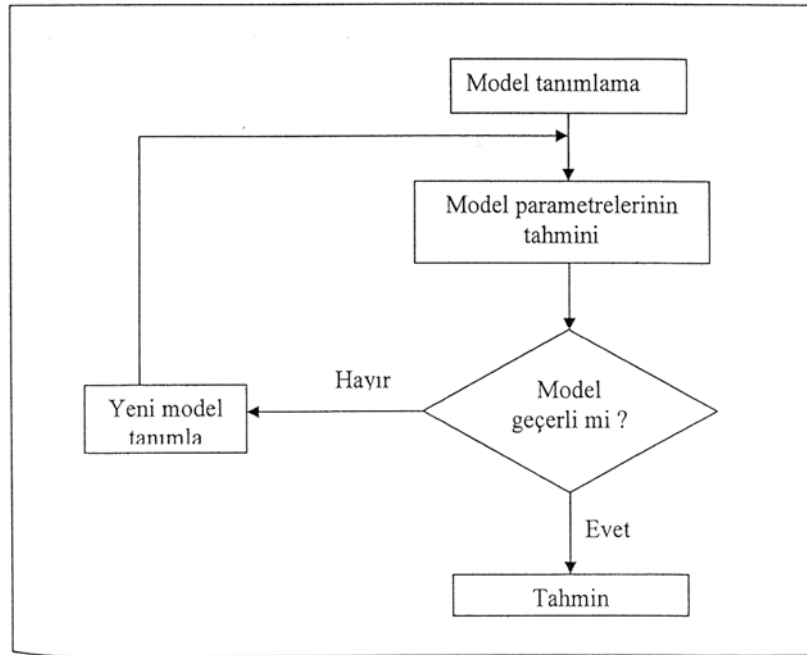
2.6.11. Model Belirleme

Model belirleme, zaman serisinin önceki bölümlerde bahsettiğimiz AR(p), MA(q), ARMA(p,q) ve ARIMA(p,d,q) süreçlerinden hangisine uyduğunun ve uygun sürecin derecelerinin ne olduğunun belirlenmesi işlemidir. En geniş hali ile ifade edilirse, ARIMA(p,d,q) sürecinde, p, q ve d değerlerinin ne olması gerektiğinin belirlenmesidir. Durağan olmayan zaman serileri için durağanlaştırma parametresini ifade eden d değerinin belirlenmesi kolaydır. Seri kaçınıcı fark alma işleminden sonra durağanlaşmış ise, d parametresinin değeri bu sayıya eşittir, p ve q parametrelerinin belirlenmesi ise otokorelasyon ve kısmi otokorelasyon fonksiyonları yardımı ile olur. Şekil 2.9, model belirleme aşamalarını göstermektedir. Otokorelasyon ve kısmi otokorelasyon fonksiyonları hakkında önceki bölümlerde ayrıntılı bilgi verilmiştir.

Tablo 2.1 OKF ve KOKF'nun model belirlemede nasıl yardımcı olduğunu özetlemektedir.

Model tanımlama işleminden sonra, seçilen geçici model için parametre kestirimi yapılması gerekir. En iyi parametre kestirimi, hata kareler toplamını en küçük yapan kestirimlerdir. Bu işlem için en çok olabilirlik (maksimum likelihood), Bayes yaklaşımı ya da en küçük kareler yaklaşımı kullanılabilir (2).

Model tanımlanıp parametre kestirimleri yapıldıktan sonra yapılması gereken, modelin geçerliliğinin sınanmasıdır. Eğer kurulan model geçerli bir model ise bu model ile geleceğe yönelik kestirim yapılabilir, aksi halde yeni bir model tanımlanmalı ve aynı işlemler bu yeni model için yapılmalıdır. Modelin geçerliliğinin sınanması, Q istatistiği adı ile bilinen bir test ile yapılabilir.



Şekil 2.9 Box-Jenkins modelinin kurulumu

Tablo 2.1 Model belirlemede OKF ve KOKF kullanımı

	OKF	KOKF
AR(p)	Üstel olarak ya da sinüs eğrisi şeklinde azalır.	p gecikme sonra istatistiksel olarak anlamlı değildir.
MA(q)	q gecikme sonra istatistiksel olarak anlamlı değildir.	Üstel olarak ya da sinüs eğrisi şeklinde azalır.
ARMA(p,q)	Üstel olarak ya da sinüs dalgalan şeklinde azalır. q-p gecikme sonra istatistiksel olarak anlamlı değildir.	Üstel olarak ya da sinüs dalgalan şeklinde azalır. p-q gecikme sonrası istatistiksel olarak anlamlı değildir.

2.6.12. Q test istatistiği

Q istatistiği 1970 yılında Box ve Pierce (11) tarafından geliştirilmiş, 1978 yılında Ljung ve Box (12) tarafından düzeltilmiştir. χ^2 dağılımına uyan Q istatistiği, parametre kestirimi yapılmış modelin verdiği artık değerlerini test ederek, modelin geçerliliği hakkında bilgi verir.

Box ve Jenkins (8) hataların ilk 20-25 otokorelasyonuna bir bütün olarak bakmayı sağlayan Q istatistiğinin kullanımını önermiştir. Ancak, Ljung ve Box (18), gözlem sayısının 100'den küçük olması durumunda χ^2 dağılımının zayıf bir istatistik olması nedeni ile düzeltilmiş Q istatistiğinin kullanılmasını önermektedir (10).

Q istatistiği, otokorelasyonların ilk m tanesinin sıfırdan farklı olup olmadığını belirlemektedir. Eğer tüm $\rho_k = 0$ ise süreç tamamen raslantısaldır. Box ve Pierce'nin önerdiği Q istatistiği,

$$Q = n \sum_{k=1}^m r_k^2, \quad \chi^2_{m-p-q} \quad [2.92]$$

şeklindedir.

Düzeltilmiş Ljung-Box istatistiği ise,

$$Q^o = n(n+2) \sum_{k=1}^m \frac{r_k^2}{n-k}, \quad \chi^2_{m-p-q} \quad [2.92]$$

biçimindedir. Yukarıdaki eşitliklerde,

n: gözlem sayısı,

m: test edilecek otokorelasyon katsayısı

p: AR modelinin derecesi

q: MA modelinin derecesi

olarak tanımlanmaktadır.

Q istatistiği, hata terimlerinin aralarında otokorelasyon olmadığına ilişkin aşağıdaki,

$$H_0 : \rho_1(\alpha) = \rho_2(\alpha) = \dots = \rho_k(\alpha) = 0 \quad [2.94]$$

hipotezinin kabul edilip edilmeyeceğine karar vermede kullanılır. Eğer hesaplanan Q değeri,

$$Q > \chi^2_{m-p-q, 1-\alpha} \quad [2.95]$$

ise, H_0 hipotezi red edilir ve modelin geçerli bir model olmadığına karar verilir. Hesaplanan Q değerinin,

$$Q \leq \chi^2_{m-p-q, 1-\alpha} \quad [2.96]$$

olması durumunda ise, H_0 hipotezi kabul edilir ve modelin geçerli bir model olduğuna karar verilir.

2.6.13. Kestirim

Modelin geçerliliği sılandıktan sonra, geçerli olduğu kabul edilen model gelecek dönemlere ait kestirim için kullanılabilir.

Bütün tek değişkenli zaman serisi modellerinde olduğu gibi ARIMA modellerinde de tek bir mantıktan hareket edilir. İlk olarak zaman serisi sürecinin beklenen değeri hesaplanır, daha sonra beklenen değer yardımı ile geleceğe ait değerler elde edilmeye çalışılır. Zaman serisinin bugünkü değeri Y_t olduğunda ve Y_{t+1} , Y_{t+2} , ..., Y_{t+L} değerleri kestirilmek istendiğinde, Y_{t+L} 'nin kestirimi $Y_t(L)$ biçiminde gösterilir.

ARIMA modeli ile kestirim yapılırken ilk olarak bir dönem sonraki kestirim değeri hesaplanmakta, bulunan kestirim değeri iki dönem sonraki kestirim değerinin hesaplanmasında kullanılmaktadır. Bu işlemler kestirilecek dönem sayısı bitinceye kadar devam etmektedir (10).

Bir çevrilebilir ve durağan ARIMA(p,0,q) stokastik süreci y_t için, $t=n+L$ olduğunda, süreç,

$$y_{n+L} = \phi_1 y_{n+L-1} + \dots + \phi_L y_{n+L-L} + a_{n+L} - \theta_1 a_{n+L-1} - \dots - \theta_q a_{n+L-q} \quad [2.97]$$

eşitliği ile gösterilebilir. Burada, n en son gözlem dönemini, L de kestirilecek dönemi göstermektedir. Eş. 2.97'den y_{n+L} 'nin beklenen değeri şu şekilde bulunur:

1. Şimdiki ve geçmiş hata terimleri a_{n+j} ($j < 0$) için gerçek hata terimleri,
2. Gelecek dönem hata terimlerinin a_{n+j} ($0 < j < L$) beklenen değeri için sıfır,
3. Şimdiki ve geçmiş gözlem değerleri y_{n+j} ($j \leq 0$) için gerçek gözlem değerleri,
4. Gelecek dönem gözlem değerleri y_{n+j} ($0 < j < L$) için y_{n+L} 'nin yaklaşık kestirimi kullanılır (2).

Durağan olmayan zaman serilerinde önce seri durağanlaştırılır ve daha sonra ARMA (p,q) sürecinde olduğu gibi kestirim yapılmaya çalışılır. Fark alınarak durağanlaştırılmış seri için yapılan kestirimlerden sonra orijinal seriye dönüşüm yapılır. Eş. 2.89, durağanlaştırılmış seriyi göstermektedir. Buradaki d parametresi kaç defa fark alma işleminin yapıldığını belirtmektedir. Örneğin, d=1 olduğunda Y_t 'nin L dönem kestirim i,

$$y_t(L) = y_t + z_t(1) + z_t(2) + \dots + z_t(L) \quad [2.98]$$

olarak hesaplanır.

Zaman serilerinde kestirim, nokta kestirimi olarak yapılabileceği gibi, aralık kestirimi olarak da yapılabilir. Aralık şeklinde kestirim için, standart hatasının hesaplanması gereklidir. Kestirim standart hatasının hesaplanması, ARIMA modelini hata terimleri cinsinden ifade ederek yapılabilir. Yani modeldeki geçmiş gözlem değerleri hata terimleri cinsinden yazılmalıdır. Bu durumda y_t süreci,

$$y_t = a_t + \psi_1 a_{t-1} + \psi_2 a_{t-2} + \dots \quad [2.99]$$

şeklinde yazılabilir. Burada $\psi_1, \psi_2, \dots$ ARIMA modeli parametreleri cinsinden katsayılarıdır.

Gelecek dönem kestirim hatası $a_n(L)$ 'nin beklenen değeri, kestirim hatasının ortalamasını verecektir ve sifıra eşittir. Varyansı ise,

$$Var[a_n(L)] = E[a_n^2(L)] = \sigma_a^2 \sum_{j=0}^{k-1} \psi_j^2 \quad [2-100]$$

formülü ile hesaplanabilir. Burada $\psi_0 = 1$ 'dir.

Hata varyansı da hesaplanınca, kestirim için güven aralığı aşağıdaki formülle bulunabilir.

$$y_n(L) \pm 1.96 \sqrt{Var[a_n(L)]} \quad [2.101]$$

2.6.14. Mevsimlik zaman serileri için Box-Jenkins modelleri

Daha önce de belirtildiği gibi mevsimlik değişimler eşit zaman araları ile tekrarlanan düzenli değişimlerdir. Mevsimlik değişimler bir yıldan kısa süreli zaman serilerinde görülür. Mevsimlik zaman serilerinde iki çeşit zaman aralığı bulunur. Bunlar, birbirini izleyen iki gözlem değeri arasındaki zaman aralığı ve birbirini izleyen aynı mevsim gözlem değerleri arasındaki zaman aralığıdır. Bir mevsimlik zaman serisinde iki tür değişme söz konusudur. Birbirini izleyen gözlem değerleri arasındaki değişim ve birbirini izleyen aynı mevsim gözlem değerleri arasındaki değişimdir. Dolayısıyla birbirini izleyen gözlem değerleri arasında ilişki olduğu gibi, birbirini izleyen aynı mevsim gözlem değerleri arasında da ilişki vardır.

Mevsimsel zaman serilerinde, birbirini izleyen gözlem değerleri arasındaki ilişkiyi gösteren bir ARIMA(p,d,q) modeli uygulanabildiği gibi, birbirini izleyen aynı mevsim gözlem değerleri arasındaki ilişkiyi gösteren ARIMA(P,D,Q)S modeli uygulanabilir. Mevsimsel ARIMA(P,D,Q)S modelindeki P, D ve Q parametreleri, ARIMA(p,d,q) modelinde olduğu gibi sırasıyla otoregresif süreç derecesi, fark derecesi ve hareketli ortalama süreç derecesi belirtmektedir, s ise birbirini izleyen aynı mevsim gözlem değerleri arasındaki zaman aralığıdır. Aylık zaman serilerinde s=12, üç aylık zaman serilerinde s=4'tür.

Bir mevsimlik zaman serisi için model belirlerken, yukarıda ifade edilen iki çeşit değişmeyi de yansıtacak model belirlenmelidir. Zaman serilerinde bulunabilecek değişkenlikler arasında çarpımsal ilişki olduğu varsayımından hareketle mevsimlik zaman serisi modeli kısaca, ARIMA(p,d,q)x(P,D,Q)s şeklinde gösterilir (2).

Örneğin bir ARIMA(1,1,1)(1,1,1)₄ modeli aşağıdaki gibi gösterilir (13):

$$(1-\phi_1B)(1-\Phi_1B^4)(1-B)(1-B^4)Y_t=(1-\theta_1B)(1-O_1B^4)a_t \quad [2.102]$$

Daha genel gösterimle çarpımsal mevsimlik zaman serisi kestirim modeli,

$$\phi_p(B)\Phi_p(B^s)\Delta^d\Delta^D Y_t = \theta_q O_Q(B^s)a_t \quad [2.103]$$

biçiminde gösterilir. Bu genel gösterim aşağıdaki açık ifadelerle daha kolay anlaşılacaktır.

$$\begin{aligned} \phi_p(B) &= 1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p \\ \Phi_p(B) &= 1 - \Phi_1 B^s - \Phi_2 B^{2s} - \dots - \Phi_p B^{ps} \\ \theta_p(B) &= 1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_p B^q \\ O_Q(B) &= 1 - O_1 B - O_2 B^{2s} - \dots - O_Q B^{Qs} \end{aligned} \quad [2.104]$$

$$BY_t = Y_{t-1}$$

$$B^2 Y_t = Y_{t-2}$$

.....

$$B^k Y_t = Y_{t-k}$$

$$\Delta Y_t = Y_t - Y_{t-k} \quad [2.105]$$

$$\Delta^2 Y_t = \Delta(\Delta Y_t) = \Delta(Y_t - Y_{t-1}) = Y_t - Y_{t-1} - (Y_{t-1} - Y_{t-2}) = Y_t - 2Y_{t-1} + Y_{t-2}$$

.....

$$\Delta_s Y_t = Y_t - Y_{t-s}$$

$$\Delta_s^2 Y_t = \Delta_s(Y_t - Y_{t-s}) = (Y_t - Y_{t-s}) - (Y_{t-s} - Y_{t-2s}) = Y_t - 2Y_{t-s} + Y_{t-2s}$$

Mevsimlik zaman serileri için uygun kestirim modellerinin belirlenmesi, parametrelerin kestirimi, uygunluk testleri ve gelecek dönemler için yapılması, mevsimlik olmayan modellerde olduğu gibi yapılır. Zaman serilerinde mevsimlik değişmelerin olup olmadığı otokorelasyon analizi ile ortaya konabilir. Örneğin zaman serisi üç aylık gözlem değerlerinden oluşuyorsa ve seri durağan ise, seride mevsimlik değişmeler söz konusu ise, her dördüncü gecikmedeki otokorelasyon sıfırdan önemli ölçüde farklı, bunları izleyen otokorelasyon değerleri ise küçük olacaktır. Seri durağan değilse, durağanlaştırma işleminden sonra otokorelasyon analizi yapılarak serinin mevsimsel olup olmadığına karar verilir (2).

3. YAPAY SİNİR AĞLARI

3.1. Giriş

İnsan beyninin çalışma ilkelerini öykünen yapay sinir ağları (YSA), örnek sınıflandırma ve örüntü tanımda önemli bir araçtır. YSA, deneme-yanılma yolu ile öğrenebilmekte ve daha sonra da genelleme yapabilmektedir. YSA; işletme bilimi, sanayi ve fen bilimleri gibi bir çok değişik alanda başarı ile kullanılmaktadır. YSA'nın kullanıldığı alanlardan biri de kestirim problemleridir. YSA'nın bir kestirim aracı olarak geniş kabul görmesi ancak kişisel bilgisayarların gelişmesi ve YSA yazılımlarının yaygınlaşması sonucu olmuştur.

Haykin (13) YSA'yı şöyle tanımlar:

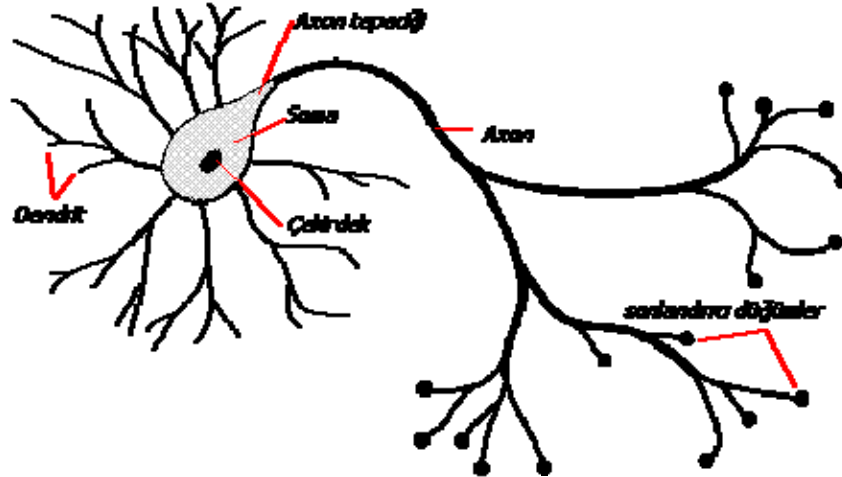
Yapay sinir ağı; deneyime dayalı bilgiyi depolamaya ve bu bilgiyi kullanıma sunmaya yönelik doğal bir eğilim içinde olan yoğun paralel dağıtılmış bir işlemcidir. YSA iki açıdan insan beynine benzemektedir: Bilgi ağ tarafından bir öğrenme süreci aracılığı ile elde edilmektedir ve sinir hücreleri arasında snaptik ağırlık olarak adlandırılan bağlar bilgiyi depolamakta kullanılmaktadır.

YSA, temel biyolojik sinir sistemlerinin taklididir. İnsan beyni, nöron adı verilen birbirleri ile bağlantılı işlem elemanlarından oluşmaktadır. Her bir nöron bir diğer nöronun çıktısını (toplam bilgisini) veya dışarıdan gelen bir uyarıyı girdi sinyali olarak alır, bir aktivasyon veya transfer fonksiyonu ile işler ve bir diğer nörona veya dışarıya verilmek üzere çıktı sinyalini üretir. Her bir nöron kendi görevini yavaş ve eksik yerine getirmesine rağmen, bütün ağ şaşırtıcı bir şekilde çok sayıda ki görevi eksiksiz ve oldukça etkin bir şekilde yapar (41). Bu bilgi işleme karakteristiği YSA'ya güçlü bir hesaplama aygıtı olma özelliği sağlar.

İnsan beyninin ve YSA'nın temel yapı taşı sinir hücreleridir. Temel insan sinir hücresi Şekil 3.1.'de gösterilmiştir. Sağlıklı bir insan beyni yaklaşık olarak birbirine

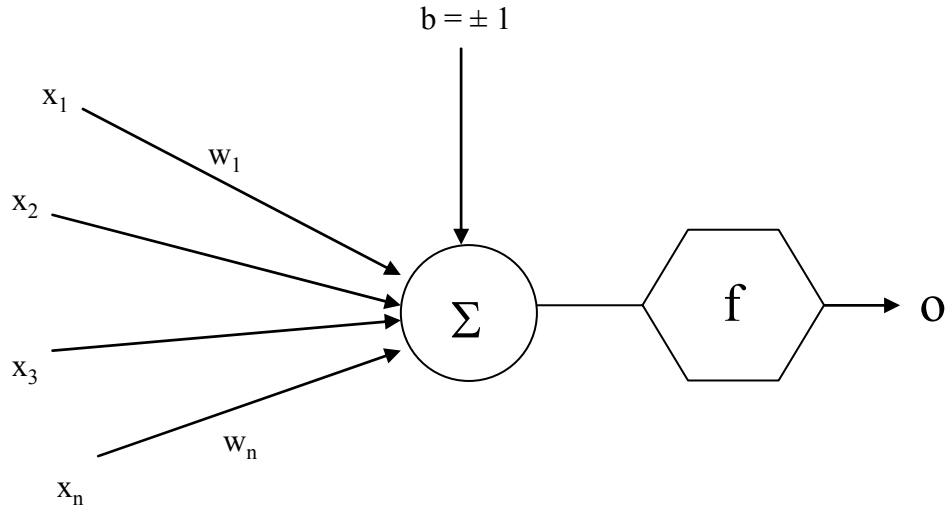
bağlı 10^{11} sinir hücresinden oluşarak, beynin paralel işlem yeteneğini sağlamaktadır (14).

İnsan sinir hücresi; çekirdek, gövde ve iki uzantıdan oluşmaktadır. Bu uzantılardan dentrit adı verilen kısadır ve binlerce dala dallanmıştır. Görevi giriş bilgilerini almaktır. Tek ve uzun olan uzantıya ise akson adı verilir. Görevi çıktı bilgilerini diğer sinir hücrelerine taşımaktır. Akson ve dentritin birleştiği yere sinaps adı verilir ve yaklaşık olarak 10^{14} sinaps var olduğu kestirilmektedir (14).



Şekil 3.1. Temel insan sinir hücresi (15)

Şekil 3.2.'de bir yapay sinir hücresi gösterilmektedir. Girdiler sinir hücresine girdikten sonra snaptik ağırlıkları ile çarpılırlar, daha sonra toplanırlar ve bir aktivasyon fonksiyonu tarafından işlenirler. Aktivasyon fonksiyonu, sinir hücresinin çıktısının sınırlarını belirler.

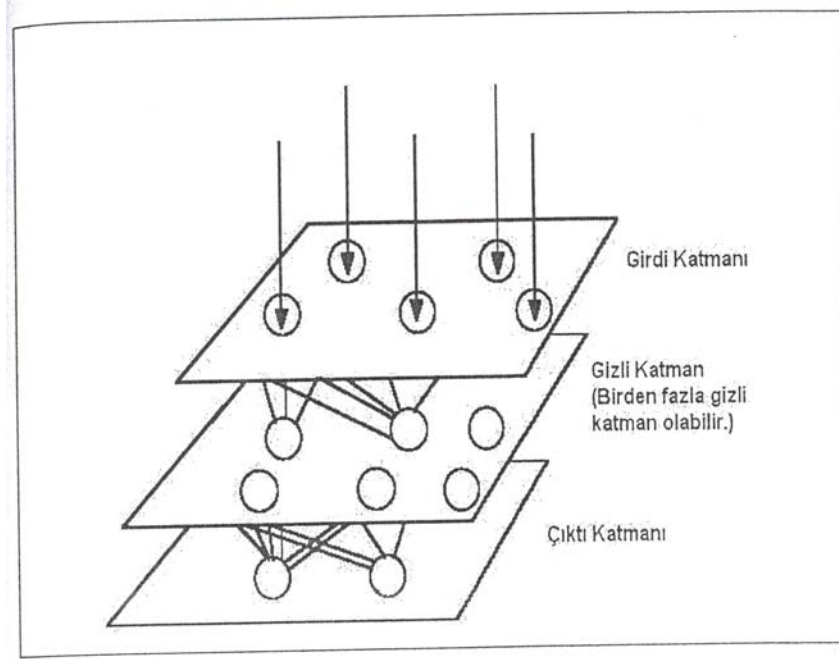


Şekil 3.2. Yapay sinir hücresi modeli (16)

YSA, yapay sinir hücrelerinin gruplanması ile oluşmuş sistemin adıdır (16). Bu gruplandırma, birbirine bağlı katmanlardan meydana gelir. Şekil 3.3, bir YSA yapısını göstermektedir. Bu yapıdaki ağlar, çok katmanlı algılayıcı (*Multi-layer perceptrons* - MLP) olarak adlandırılır. Şekil 3.3'te de görüleceği üzere YSA'lar, verileri alan nöronların bulunduğu girdi katmanından, ağın sonuçlarını veren nöronların oluşturduğu çıktı katmanından ve bu iki katman arasındaki bir veya daha fazla gizli katmandan oluşmaktadır (15). Bir çok ağda gizli katmanlardaki nöronlar sinyallerini bir önceki katmandaki nöronlardan alırlar. Herhangi bir nöron kendi fonksiyonunu yerine getirdikten sonra çıktısını bir sonraki katmanın nöronlarına gönderir.

3.2. YSA'nın Temelleri

Çok katmanlı algılayıcılar (ÇKA), en bilinen ve en yaygın kullanılan YSA türüdür. Bir ÇKA aşağıdaki bileşenlerden oluşur:

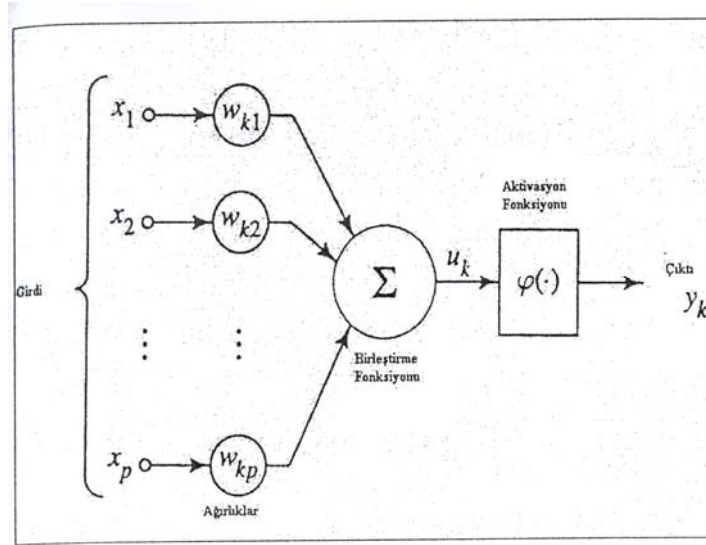


Şekil 3.3. Basit bir YSA Yapısı (15)

1. Girdiler,
2. İşlem birimleri kümesi,
3. Her bir birimin çıktısına eşit olan aktivasyon değeri,
4. Birimler arası bağlantılar,
5. Katmanlar,
6. Birleştirme fonksiyonu,
7. Aktivasyon fonksiyonu,
8. Öğrenme algoritması,
9. Öğrenme metodu,

3.2.1. İşlem Birimi

Nöron ya da düğüm diye de adlandırılan bir işlem birimi, tek başına ele alındıklarında çok basit işleve sahip işlemcilerdir. İşlem birimi (Şekil 3.4), dış kaynaklardan veya komşularından girdiyi alır ve diğer birimlere yayılacak sinyali hesaplamak üzere kullanır (16).



Şekil 3.4. İşlem birimi (13)

Yapay sinir sistemlerinde üç çeşit nöron vardır:

- Girdi nöronları: ağ dışından veriyi alır,
- Çıktı nöronları: ağına dışına veriyi gönderir,
- Gizli nöronlar: bir önceki katman nöronlarından girdiyi alır ve çıktısını sonraki katman nöronlarına gönderir.

Her bir nöron birden fazla ($x_1, x_2, \dots, x_n$) girdiye sahip olabilir ancak her bir nöronun sadece bir çıktısı vardır. Bir nöronun girdisi ağına dışından olabileceği gibi bir başka birimin çıktısı ya da kendi çıktısı olabilir. Bir nöronun girdileri ağırlıkları ile çarpılarak birleştirme fonksiyonuna girerler.

3.2.2. Toplama (Birleştirme) Fonksiyonu

Bir YSA'da, girdi birimi olmayan her bir nöron bağlantılar yardımı ile diğer nöronlardan gelen değerleri toplar ve net girdiyi üretir. Diğer nöronlardan gelen değerleri toplayan bu fonksiyona, toplama (birleştirme) fonksiyonu denir. Bir çok YSA'da, her bir nöronun bağlantılı olduğu diğer nöronun girdisine ek bir katkıda bulunduğu varsayılır. Herhangi bir j nöronunun toplam girdisi, diğer nöronlardan gelen değerlerin ağırlıklı toplamı ile eşik (bias) değerinin toplamına eşittir:

$$a_j = \sum_i w_{ji} x_i + \theta_j \quad [3.1]$$

Yukarıdaki fonksiyona toplam fonksiyonu denir. Toplam fonksiyonundan başka kullanılan birleştirme fonksiyonları da vardır. En çok kullanılan birleştirme fonksiyonları Tablo 3.1'de gösterilmiştir (17).

Tablo 3.1. Bazı Birleştirme Fonksiyonları

Toplam	$a_j = \sum_i w_{ji} x_i$	Çarpım	$a_j = \prod_i w_{ji} x_i$
Maksimum	$a_j = Maks(w_{ji} x_i)$	Minimum	$a_j = Min(w_{ji} x_i)$
Çoğunluk	$a_j = \sum_i Sgn(w_{ji} x_i)$		

x_i = i nöronunun çıktısı

w_{ji} = j ve i nöronları arasındaki bağlantının ağırlığı

a_j = j nöronunun net girdisi

3.2.3 Aktivasyon Fonksiyonu

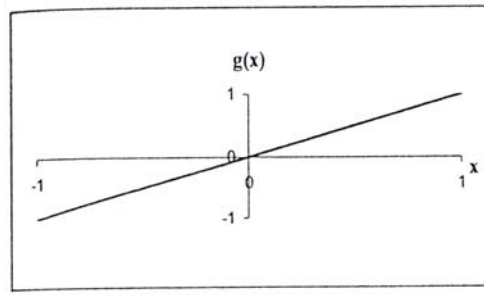
Nöron davranışını belirleyen önemli faktörlerden biri nöronun aktivasyon fonksiyonudur. Toplama fonksiyonunun sonuçları aktivasyon fonksiyonu diye bilinen bir süreç yardımı ile çıktıya dönüştürülür. Literatürde, sıkıştırma veya eşik fonksiyonu olarak da adlandırılmaktadır. Bunun nedeni, çıktı sinyallerini [0,1] veya [-1,1] aralığında sınırlandırmasıdır. İlgilenilen problem türüne ve ağ yapısına bağlı olarak değişik aktivasyon fonksiyonları kullanılabilir. Genellikle kullanılan aktivasyon fonksiyonlarına aşağıda değinilmiştir (4):

i. Doğrusal Fonksiyon

Fonksiyon Eş.3.2'de ve fonksiyona ait grafik Şekil 3.5'te gösterilmiştir.

$$g(x) = x$$

[3.2]

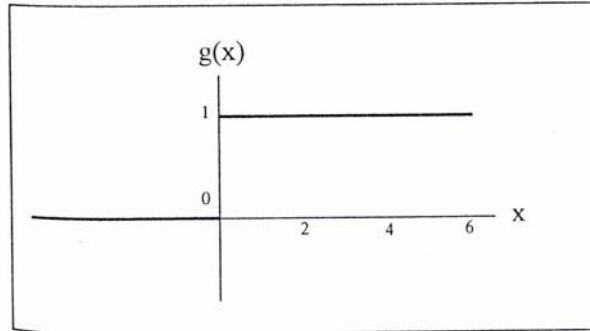


Şekil 3.5. Doğrusal fonksiyon

ii. Basamak Fonksiyonu

Fonksiyon Eş.3.3'te ve fonksiyona ait grafik Şekil 3.6'da gösterilmiştir.

$$g(x) = \begin{cases} 1 & \text{if } (x \geq \theta) \\ 0 & \text{if } (x < \theta) \end{cases} \quad [3.3]$$



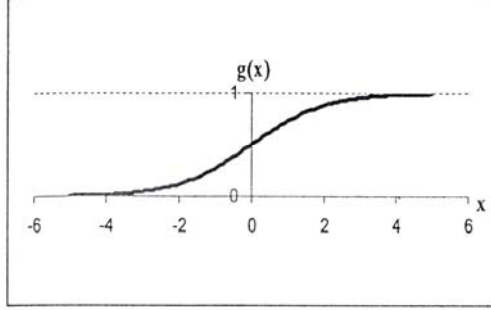
Şekil 3.6 Basamak fonksiyonu

Basamak fonksiyonu genellikle tek katmanlı ağlarda kullanılır.

iii. Sigmoid Fonksiyonu

Fonksiyon Eş.3.4'te ve fonksiyona ait grafik Şekil 3.7'de gösterilmiştir.

$$g(x) = \frac{1}{1 + e^{-x}} \quad [3.4]$$

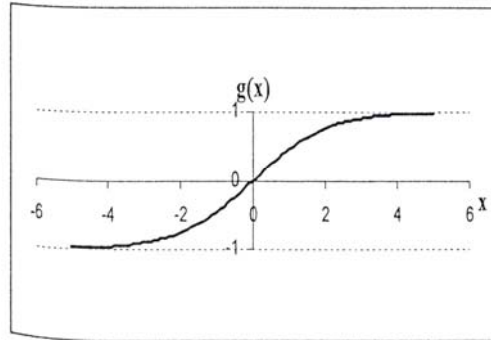


Şekil 3.7 Sigmoid fonksiyonu

Sigmoid fonksiyonu özellikle geri yayılım (backpropagation) tekniği ile eğitilen ağlarda avantajlıdır. Sigmoid fonksiyonunun çıktısı $[0,1]$ aralığındadır.

iv. Hiperbolik Tanjant Fonksiyonu

Fonksiyon Eş.3.5'te ve fonksiyona ait grafik Şekil 3.8'de gösterilmiştir.



Şekil 3.8 Hiperbolik tanjant fonksiyonu

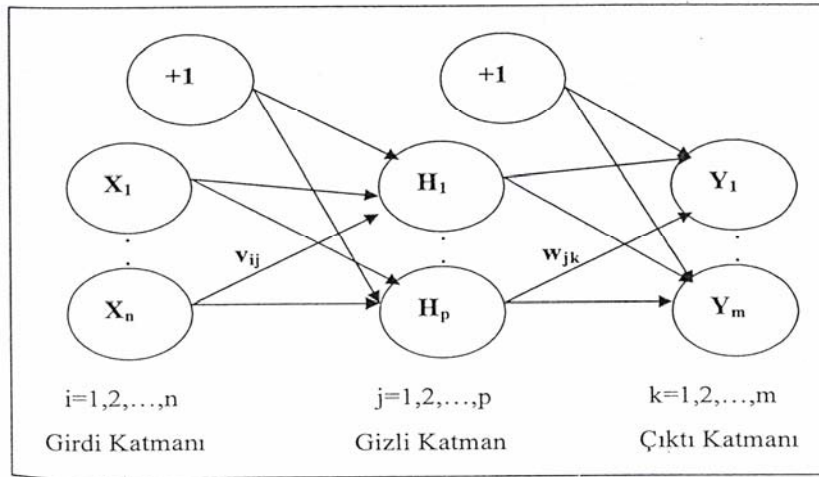
$$g(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad [3.5]$$

Hiperbolik tanjant fonksiyonu, çıktısının $[-1,1]$ aralığında olması gereken uygulamalarda kullanışlıdır.

3.2.4. YSA'nın Yapısı

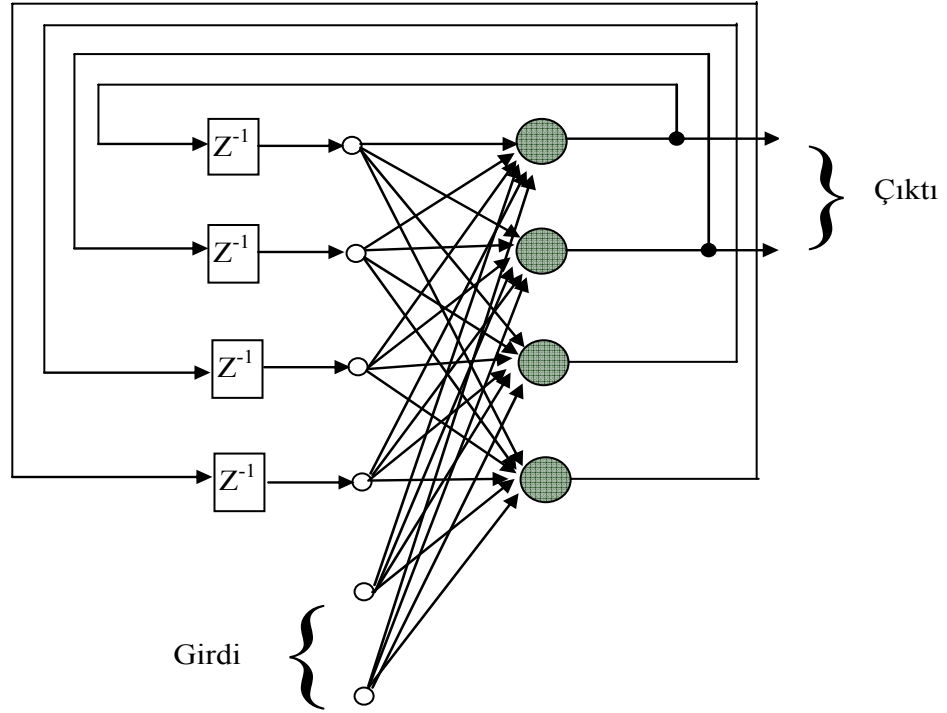
YSA'nın yapısı; katman sayısına veya katmanlar arasındaki bağlantıların durumuna göre tanımlanmaktadır. YSA'lar katmanlar arasındaki bağlantının yapısına göre iki sınıfta incelenir:

i. İleri Beslemeli Ağlar: Verilerin girdi birimlerinden çıktı birimlerine ileri doğru aktığı ağ yapısıdır. Bu ağ yapısında (Şekil 3.9) geri besleme yoktur. Şekil 3.9'da, $\{X_1, \dots, X_n\}$ girdi nöronlarını, $\{H_1, \dots, H_p\}$ gizli nöronları, $\{Y_1, \dots, Y_m\}$ çıktı nöronlarını, v_{ij} , i girdi nöronundan j gizli nöronuna olan bağlantının ağırlığını ve w_{jk} , j gizli nöronundan k çıktı nöronuna olan bağlantının ağırlığını göstermektedir. $+1$ olarak gösterilen birimler eşik (bias term) değerleridir.



Şekil 3.9. İleri Beslemeli Ağ Yapısı

ii. Geri Beslemeli Ağlar: Veri akışının sadece ileriye doğru değil geriye doğru da olabileceği ağ yapısıdır. Bu ağ yapısında (Şekil 3.10), ağ çıktısı aynı zamanda girdi olarak da kullanılabilir.



Şekil 3.10. Geri Beslemeli Ağ Yapısı (13)

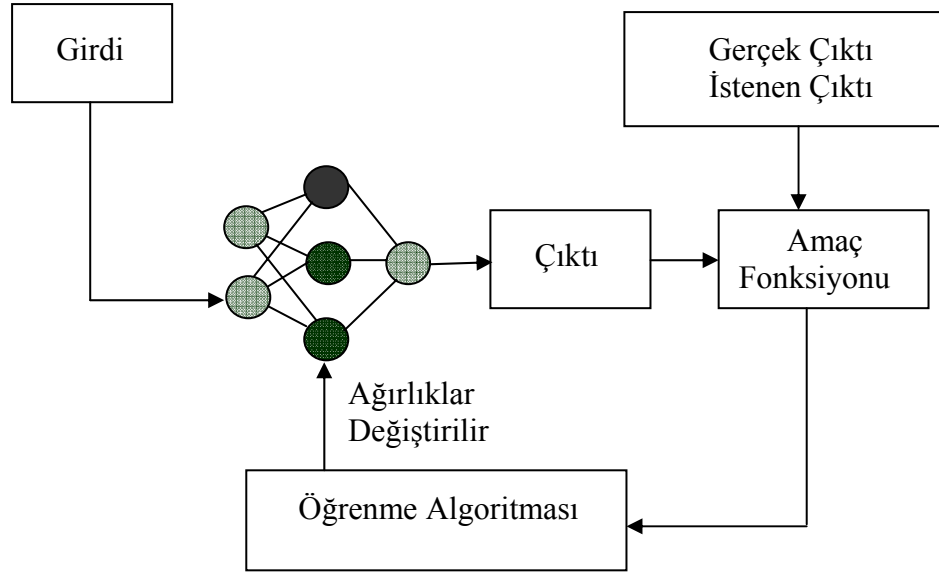
3.2.5. YSA'nın Eğitimi

YSA'nın en temel özelliği öğrenme yeteneğidir. YSA'da girdi ile çıktı arasındaki ilişkinin öğrenilmesi amacı ile bağlantıların ağırlıklarının hesaplanması işlemi ağı eğitimi olarak adlandırılır. Başlangıç ağırlıkları keyfi olarak seçilir ve öğrenme işlemi başlar. Problemin çözümünde en iyi ağırlık kümesini bulmaya yardım eden birçok öğrenme algoritması ileri sürülmüştür. Bu algoritmalar kabaca iki sınıfta incelenebilir: öğreticili öğrenme ve öğreticisiz öğrenme algoritmaları. Bir YSA'nın eğitiminde, öğrenme algoritmalarının yanı sıra, öğrenme oranı ve öğrenme kuralı da önemli rol oynar. Aşağıda kısaca bu konulara değinilmiştir.

i. Öğreticili Öğrenme:

Ağ eğitimi, çıktıların istenen değerleri ağa tanıtılabiliyorsa, bu tip öğrenme, öğreticili öğrenme adını alır. Girdi ve çıktı kümeleri ağa verilir. Ağ, girdiyi işleyerek kendi çıktısını üretir ve gerçek çıktı ile karşılaştırır. Öğrenme metodu

sayesinde, var olan hatayı en aza indirmek için bağlantılardaki ağırlıklar yeniden düzenlenir. Bu işlem kabul edilebilir bir hata seviyesine ulaşıncaya kadar devam eder. Şekil 3.11, öğreticili öğrenme algoritmasını göstermektedir. Algılayıcı, geri yayılım ağı ve boltzmann makinesi en yaygın kullanılan öğreticili ağlardır (15).



Şekil 3.11. Öğreticili Öğrenme

ii. Öğreticisiz Öğrenme:

Çıktıların istenen değerleri ağı tanıtlamıyorsa, bu tip öğrenme şekli öğreticisiz öğrenme olarak adlandırılır. Öğreticisiz öğrenme algoritmaları daha çok sistemin geçmişte karşılaştığı veri kümesinin içerdiği istatistiksel bilgilerin elde edilmesini amaçlar. Çok elemanlı veri kümeleri içerisinde deneme-yanılma yoluyla bilgi genelleştirilmesi yapılabilir (18).

Öğreticisiz öğrenmede girdiler aynı zamanda çıktı görevi görmektedir. Hem çıktı hem de girdi olan veriler arasındaki kural ve ilişkilerin araştırılması ve en uygununun bulunması ağı eğitilmesi anlamına gelmektedir. Adaptif Rezonans Teorisi (ART), Hopfield Ağı, Kohonen Ağı, Sayaçlı Yayılım Ağı en çok kullanılanlarıdır.

iii. Öğrenme Oranı:

Öğrenme oranı bağlantıların ağırlık değerlerindeki değişim miktarını belirler. Öğrenme oranı, genellikle 0 ile 1 arasında bir değerdir. Öğrenme oranının büyük değerleri değişim miktarını artırırken, küçük değerleri de değişim miktarını azaltacaktır. Öğrenme oranının çok büyük değerleri kararsızlığa, çok küçük değerleri de öğrenme sürecinin kabul edilemeyecek şekilde yavaşlamasına sebep olur. Öğrenme oranının 1'den büyük olduğu durumlarda ağ yerel minimumların arasında salınır. Öğrenme oranının çok düşük olması da genel minimumu bulmaya imkan vermeyebilir. Tang ve diğ. (20)'leri, az karmaşık veri yapısında yüksek öğrenme oranının iyi olduğunu ifade etmektedir.

iv. Momentum Katsayısı:

Ağın daha hızlı toparlanmasına yardım eden bir faktördür. Daha önceki değişimin bir kısmının o andaki değişime eklemek şeklinde ifade edilebilir. Bunu yapma amacı, öğrenme sırasında ağın salınımını engellemektir. 0 ile 1 arasında değerler alır (4). Tang ve arkadaşları (20) çok karmaşık veri yapısında, düşük öğrenme oranı ve yüksek momentum katsayısı kullanılmasının iyi olacağını ifade etmektedir.

v. Öğrenme Kuralları :

Öğrenme işlemini kolaylaştırmak amacı ile öğrenme kuralları (algoritmaları) geliştirilmiştir. Bir çok öğrenme kuralı kullanılmaktadır. Bunların bir çoğu, en eski ve en bilinen Hebb Algoritmasının bir çeşididir. Önemli öğrenme kurallarından bazıları aşağıda kısaca açıklanmıştır.

a. Hebb Kuralı:

Ponald Hebb tarafından geliştirilen bu kural, öğrenme kuralları arasında en çok bilinenidir ve ilk öğrenme kuralıdır. Tanımlama, 1949 yılında yayınlanan The

Organization of Behavior adlı kitabında verilmiştir (21). Bu kuralın temel tanımı şöyledir: eğer bir nöron diğer bir nörondan girdi aldığı anda, bu iki nöron yüksek aktivitede ise (matematiksel olarak aynı işarete sahip ise), nöronlar arasındaki ağırlık güçlendirilmelidir.

b. Hopfield Kuralı:

Hebb kuralına benzer ancak bir istisnası vardır. Hopfield kuralı, ağırlıkların güçlendirilmesi işleminde güçlendirme veya zayıflatma için bir büyüklük tanımlamaktadır. İstenen çıktı ve girdinin her ikisi de aktif veya pasif ise, bağlantı ağırlığı öğrenme oranı kadar artırılır, tersi durumda da öğrenme oranı kadar azaltılır.

c. Delta Kuralı:

En çok kullanılanlardan biridir. İstenen çıktı ile işlem biriminin gerçek çıktısı arasındaki farkın (delta) azaltılması amacı ile girdi bağlantılarının ağırlıklarının değiştirilmesi temeline dayanır. Bu kural snaptik ağırlıkları, ağırlık hata kareleri ortalamasını minimize etme yolu ile değiştirir. Bu kural Widrow-Hoff Öğrenme Kuralı veya En Küçük Kareler Öğrenme Kuralı olarak da bilinir.

d. Eğim Düşme (*Gradient Descend*) Kuralı:

Delta Kuralının benzeridir. Ağırlıkların düzenlenmesi işlemi, bir işlem birimi için arzu edilen çıktı ile gerçek çıktı arasındaki hatanın birinci türevine (gradient) göre orantılı bir şekilde gerçekleştirilir. Amaç, hata fonksiyonunu azaltarak, yerel minimumdan kurtulmak ve genel minimumu yakalamaktır.

e. Kohonen Öğrenme Kuralı:

Teuvo Kohonen tarafından biyolojik sistemlerdeki öğrenmeden esinlenerek geliştirilmiştir. Bu kurala göre, işlem birimi öğrenme fırsatı için yarışır veya ağırlıklarını günceller. En geniş çıktıya sahip işlem birimi kazanan olarak ilan edilir.

Bu işlem birimi komşularını heyecanlandığı gibi, rakiplerini de engelleme yeteneğine sahiptir. Sadece galip gelenin çıktısına izin verilir ve yine sadece galip gelenin ve komşularının bağlantı ağırlıklarının ayarlanmasına izin verilir. Komşu genişliği eğitim süresi boyunca değişebilir. Genellikle geniş bir komşu tanımlaması ile başlanır ve eğitim süreci işledikçe daraltılır (22, 30).

3.2.6. Amaç Fonksiyonu

Eğitilen ağın performansını ölçmek için bir amaç fonksiyonu (maliyet fonksiyonu) tanımlanmalıdır. Hata Kareleri Toplamı ve Hata Kareleri Ortalaması, hata terimine bağlı olarak tanımlandıklarından dolayı tipik olarak kullanılır. Finansal kestirim ve benzeri bazı problemlerde, hasıla, kar, fayda maksimizasyonu gibi farklı amaç fonksiyonları da uygun olabilir. Refenes (23), öğrenme algoritması (geri yayılım algoritması) ve diğer parametrelerin sabit olduğu bir YSA'da, farklı amaç fonksiyonlarının ağın kestirim performansını etkilediğini belirtmektedir. Bu durumda amaç fonksiyonunun en iyi değerini bulabilmenin bir olası yolu, tarama algoritmasını geri yayılım tipinden, genetik algoritmalar, tavlama benzetimi ya da benzeri diğer optimizasyon metotlarına dönüştürmektir (4).

3.3. Öğrenme Algoritmaları

YSA'nın en önemli özelliklerinden biri, ilgili probleme ait örneklerle veriler arasındaki ilişkiyi öğrenmesidir. YSA'nın, veri yapısındaki ilişkiyi öğrenmesi, probleme ait örnekler yardımı ile ağ ağırlıklarının en uygun değerlerinin belirlenmesine dayanır. Herhangi bir ağırlık için;

$$W_{yeni} = W_{eski} + \Delta W \quad [3.6]$$

denklemini, öğrenmenin matematiksel olarak nasıl gerçekleştiğini ifade etmektedir. Eş. 3.6'daki ΔW değeri belli bir kurala göre hesaplanarak var olan ağırlık değerlerinin değişim miktarını verir. ΔW 'yi belirlemek için tanımlanmış kurallara öğrenme algoritmaları denir.

Bu bölümde, YSA'nın eğitiminde kullanılan geri yayılım öğrenme algoritmasına değinilecektir.

3.3.1. Geri yayılım algoritması

Geri yayılım algoritması (GYA), YSA tarihindeki en önemli gelişmelerden biridir. Bu öğrenme algoritması, sürekli, türevi alınabilir aktivasyon fonksiyonuna sahip nöronlardan meydana gelen ÇKA'da uygulanabilmektedir. GYA'nın kullanıldığı ÇKA'lar geri yayılım ağları olarak da adlandırılmaktadırlar. Girdi ve bu girdilere karşılık gelen çıktılarından meydana gelen eğitim kümesi $\{x^{(k)}, t^{(k)}\}$, $k=1,2,...,p$, şeklinde ifade edilirse, GYA, verilen eğitim seti için en uygun çözümü üretecek ağırlıkları bulmayı sağlar. Ağırlıkların düzenlenmesi işlemi, basitçe gradyan inişe dayanmaktadır.

Şekil 3.9'a göre, y_k ağın ürettiği çıktıyı ve t_k gerçek çıktıyı temsil ettiğinde, ağın eğitimi,

$$E = \frac{1}{2} \sum_{k=1}^m (t_k - y_k)^2 \quad [3.7]$$

hata fonksiyonunun minimizasyonu ile sağlanır. E hata fonksiyonunun ağırlıklara göre türevi alınırsa, ağırlıklardaki değişim miktarı elde edilir.

Geri yayılım öğrenme sürecini başlatmadan önce aşağıdakilerin belirlenmiş olması gerekir (22):

1. Eğitim kümesi,
2. Öğrenme oranı,
3. Algoritmayı sonlandırma kriteri,
4. Ağırlıkları düzenleme metodolojisi,
5. Aktivasyon fonksiyonu,
6. Başlangıç ağırlıkları.

Eğitim kümesindeki girdiler, ağırlık girdi katmanındaki nöronların bir çıktı üretmesini sağlarlar. Bu çıktı bir sonraki katman nöronlarının girdisini oluşturur. Böylece çıktı katmanındaki nöronların ağırlık çıktısını üretmeleri sağlanır. Ağırlık ürettiği çıktı ile eğitim kümesindeki gerçek değerler karşılaştırılır. Aradaki fark hesaplanır. Hatayı minimize etmek için, algoritma, geriye doğru çıktı katman ağırlıklarının (son gizli katman ile çıktı katmanı arasındaki ağırlıklar) yeniden hesaplamaya çalışır. Daha sonra son gizli katmandaki ve girdi katmanına doğru bütün katmanlardaki ağırlıklar yeniden hesaplanır. Algoritmayı sonlandırma kriteri sağlanana kadar bu işlem sürer. Algoritma, ağırlıkların geriye doğru düzenlenmesinden dolayı, GYA adını almıştır (22).

Gizli katman ve çıktı katman aktivasyon fonksiyonu olarak sigmoid fonksiyonun kullanıldığı bir ÇKA için, GYA matematiksel olarak aşağıdaki gibi ifade edilebilir:

$$\Delta w[k,j] = -\eta \partial(E) / \partial w[k,j] \quad [3.8]$$

h_j ve y_k arasındaki bağlantının $j \rightarrow k$ olması durumunda,

$$\partial(E) / \partial w[k,j] = [\partial(E) / \partial [w^{hy}h]_k] [\partial([w^{hy}h]_k) / \partial w[k,j]] = -\delta_k h_j \quad [3.9]$$

$$\delta_k = -\partial(E) / \partial [w^{hy}h]_k = -[\partial(E) / \partial y_k] [\partial y_k / \partial [w^{hy}h]_k] \quad [3.10]$$

$$y_k = g([w^{hy}h]_k) \quad [3.11]$$

$$\delta_k = -[\partial(E) / \partial y_k] g'([w^{hy}h]_k) \quad [3.12]$$

$$\delta_k = -[y - t]_k g'([w^{hy}h]_k) \quad [3.13]$$

$$\Delta w[k,j] = -\eta \delta_k h_j \quad [3.14]$$

x_j ve h_j arasındaki bağlantının $i \rightarrow j$ olması durumunda,

$$\partial(E) / \partial v[j,i] = [\partial(E) / \partial [v^{xh}x]_j] [\partial([v^{xh}x]_j) / \partial v[j,i]] = -\delta_j x_i \quad [3.15]$$

$$\delta_j = -[\partial(E) / \partial [v^{xh}x]_j] = -[\partial(E) / \partial h_j] [\partial h_j / \partial [v^{xh}x]_j] \quad [3.16]$$

$$\partial(E) / \partial h_j = \sum_k [\partial(E) / \partial [w^{hy}h]_k] [\partial([w^{hy}h]_k) / \partial h_j] = -\sum_k \delta_k w[k,j] \quad [3.17]$$

$$\partial h_j / \partial [v^{xh}x]_j = g'([v^{xh}x]_j) \quad [3.18]$$

Böylece,

$$\delta_j = \sum_k \delta_k w[k,j] g'([v^{xh}x]_j) \quad [3.19]$$

$$\Delta v[j,i] = -\eta \delta_j x_i \quad [3.20]$$

elde edilmiş olur. Aşağıda algoritmanın bir özeti verilmiştir.

1. Başlangıç ağırlık değerlerini rasgele olarak ata.
2. Eğitim kümesindeki verilere göre ileriye doğru hesaplama yap ve kaydet.
3. Eğitim kümesinden her bir girdi-çıktı koleksiyonları için delta değerini hesapla.

$$\begin{aligned}\delta_k &= [y-t]_k g'([w^{hy}h]_k) = [y-t]_k g([w^{hy}h]_k) [1-g([w^{hy}h]_k)] \\ &= [y-t]_k y_k (1-y_k)\end{aligned}\quad [3.21]$$

4. Hatayı geriye doğru yay.

$$\delta_k = (\sum_i \delta_i w[k,j]) g'([v^{xh}x]_j) = (\sum_i \delta_i w[k,j]) h_j (1-h_j) \quad [3.22]$$

5. Ağırlıkları düzenle.

$$\Delta v[j,i] = \eta \delta_j x_i \quad [3.23]$$

6. Sonlandırma kriteri sağlanıncaya kadar adım 3-5'i tekrarla.

4. GELECEĞİ KESTİRİM PROBLEMLERİ İÇİN YSA

4.1. Giriş

Daha önceki bölümlerde de belirtildiği gibi YSA'nın başarı ile kullanıldığı alanlardan biri de geleceği kestirim problemleridir. Geleceği kestirim problemleri, neden-sonuç ilişkisine dayanan kestirim problemleri ve zaman serilerine dayalı kestirim problemleri olarak iki grupta incelenebilir. Neden-sonuç ilişkisine dayalı kestirim problemleri, bir veya daha fazla açıklayıcı değişkenin, bir açıklanan değişken üzerindeki etkisinin incelendiği problemler, zaman serilerine dayalı kestirim problemleri ise, zaman boyutunda var olan veriler ile gelecek dönemlere ait değerlerin kestirilmesi olarak bilinir. Neden-sonuç ilişkisine dayanan bir kestirim probleminde, YSA'nın girdileri genellikle bağımsız değişken veya değişkenlerdir, YSA çıktısı da bağımlı değişkendir. YSA tarafından kestirilen fonksiyonel ilişki şu şekilde yazılabilir:

$$y = f(x_1, x_2, \dots, x_p) \quad [4.1]$$

Yukarıdaki eşitlikte $x_1, x_2, \dots, x_p$, p adet bağımsız değişkeni, y ise bağımlı değişkeni ifade etmektedir. Bu şekilde bakıldığında YSA, doğrusal olmayan bir regresyon modeli gibi düşünülebilir.

Diğer yandan geleceğe ilişkin veya zaman serilerine dayalı kestirim probleminde, girdiler veri serisinin geçmiş gözlemlerinden oluşurken, çıktı gelecekteki kestirim değerini temsil etmektedir. Bu durumda YSA için ilişki şu şekilde yazılabilir:

$$y_{t+1} = f(y_t, y_{t-1}, \dots, y_{t-p}) \quad [4.2]$$

y_t , t zamanında gerçekleşen gözlem değerini ifade etmektedir. Böylece YSA'nın, zaman serileri kestirim probleminde doğrusal olmayan otoregresif modellere eşdeğer olduğu söylenebilir (4).

YSA'nın geleceği kestirimde kullanılmasına yönelik literatürde çok sayıda çalışma vardır. Bir çok farklı disiplinden araştırmacılar, bir kestirim aracı olarak YSA'nın kullanılabilirliğini araştırmışlardır. Yapılan çalışmaların bir çoğunda, geleneksel metotlar ile YSA teknikleri karşılaştırılmış ve hangi yöntemin daha iyi sonuç verdiği araştırılmıştır. YSA'nın bir kestirim aracı olarak kullanılmasına yönelik literatür taraması, yapılan çalışmaların uygulama alanları ve amacı hakkında bir bilgi verecektir.

4.2. Konu ile İlgili Yapılmış Çalışmalar

YSA'nın geleceği kestirime ilişkin ilk uygulamaları 1964 yılında olmuştur. Hu (24) tezinde Widrow'un uyarlanabilir doğrusal ağını (adaptive linear network) hava kestirimi için kullanmıştır. Ancak çok katmanlı ağları eğitecek algoritmanın olmayışından dolayı bu çalışma oldukça kısıtlıdır (4).

1986 yılında geri yayılım algoritmasının tanıtılmasından sonra (25), YSA'nın kestirimde kullanılmasında bir çok gelişme olmuştur. Werbos 1974 yılında (41) geri yayılım algoritmasını ilk formüle eden kişidir ve geri yayılım algoritması ile eğitilen YSA'nın regresyon analizi ve Box-Jenkins analizi gibi geleneksel yaklaşımlardan daha iyi sonuç verdiğini göstermiştir.

Sharda ve Patil (26), Tang ve Fishwick (27) çalışmalarında, YSA ve Box-Jenkins modellerinin sonuçlarını irdelemişlerdir.

Elena Montanes ve diğ. (42)'leri Nükleer Enerji ünitesinden alınan gerçek veriler ile Box-Jenkins ve YSA Modellerini karşılaştırmışlar ve Box-Jenkins tekniğinin veri serisindeki ani sistematik değişiklikleri öngöremediğini öne sürmüşlerdir.

Hanh H., Nguyen Christine ve W Chan (43) uzun dönemli zaman serisi analizinde çoklu YSA'ların tekli YSA modeline göre daha iyi sonuç verdiğini göstermiştir.

T.Taskaya., Temizel ve M.C.Casey (44) mevsimsel özellik gösteren ekonomik veriler için yeni bir dönüşüm algoritması önermiştir. Markus ve diğ.'leri gen açıklayıcı verilerin sınıflandırılmasında YSA'lerden faydalanmışlardır.

Sağlık alanında geniş ölçekli çalışmaların yer aldığı YSA uygulamaları genel olarak; tanı, beklenen yaşam, görüntü analizi, sınıflama konularında odaklanmıştır. D.Ripley yaşam analizi uygulamalarında YSA'ların geçerli bir istatistiksel yöntem olduğunu göstermiştir.

M. Swiercz ve diğ.'leri nörocerrahi uygulamalarına yönelik ICP Trendlerinin kestirimini YSA uygulaması ile gerçekleştirmişlerdir.

Sağlık alanına yönelik olarak YSA teknolojisinin kullanıldığı ticari uygulamalar da vardır. Bunlardan birisi PAPNET sistemidir. PAPNET Yöntemi ile, 300.000'den fazla hücresi bulunan smear örneği taranarak merkezi tarama ünitesine gönderilmekte ve veri olarak saklanmaktadır. Bu veriler PAPNET sistemi ile kısa sürede incelenmekte ve geleneksel yöntemler ile yanlış tanıya gidilmeksizin sağlam sonuçlar ile birlikte potansiyel kanser hücreleri de önceden ayırt edilebilmektedir (36).

Selami Serhatlioglu, Fırat Hardalaç, ve İnan Güler (38) beynin temporal bölgesi doppler sinyallerini tanı amaçlı YSA uygulaması ile incelemişlerdir.

Samuel, Eldar ve diğ. (39)'leri cerrahi laparoskopi yerine açık cholecystectomy uygulamasına karar vermek üzere YSA ile lojistik regresyon, ve doğrusal ayırma (*discriminant*) analizi tekniklerini karşılaştırmışlar ve YSA ile elde edilen sonuçların daha anlamlı ve faydalı olduğu sonucuna varmışlardır.

Cenk Şahin ve S.Noyan Oğulata (40) tiroid bezi bozukluklarının YSA ile tanısına yönelik bir çalışma gerçekleştirmiştir.

Al-Saba ve El-Amin (7), Suudi Arabistan için 1997-2006 tarihleri arasında en yüksek yük istemini kestirmeye çalışmıştır. Yazarlar çalışmalarında, YSA ile buldukları sonuçları, AR, ARMA ve ARIMA yöntemleri ile karşılaştırmıştır. Yazarlar YSA ile yapılan kestirim sonuçlarının daha iyi olduğunu ifade etmektedir.

Bir çok araştırmacı, YSA modelleri ile bilinen istatistiksel yöntemlerin karşılaştırması için çok kullanılan M-Competition verilerini kullanmıştır. M-Competition verileri işletme, ekonomi ve finansman verileridir. Bu şekilde yapılmış bazı çalışmalar : Tang ve diğ. (20)'leri ile Tang ve Fishwick (27) tarafından da ele alınmıştır.

YSA, çok başka alanlardaki kestirim problemlerinin çözümünde de kullanılmıştır. R.J.FRANK ve diğ.'leri ileri beslemeli YSA modellerini Lorenz Verisi, hava trafik talebi, hava trafik işletimi modellemesi ve Tree Ring Verisi üzerinde deneyerek modellerin genelleyebilme performansını incelemişlerdir.

Zhang (30), iki gizli katmanlı ağların, veri yapısını modellemede ve kestirimin doğruluk derecesini arttırmada tek gizli katmanlı ağlardan daha iyi olduğunu, bazı Santa Fe zaman serileri üzerinde yaptığı çalışmalar sonucunda ifade etmiştir.

Tang ve diğ. (20)'leri, öğrenme parametrelerinin YSA'nın öğrenmesindeki etkisini araştıran bir çalışma yapmıştır. Yazarlar, karmaşıklığı az olan verilerde yüksek öğrenme oranının iyi olduğunu ve çok karmaşık verilerde ise düşük öğrenme oranını ve yüksek momentum katsayısının kullanışlı olduğunu rapor etmiştir.

4.3. Geleceği Kestirimde YSA ile Modellemede Önemli Noktalar

YSA'nın istenen bir görevi yerine getirebilmesi için öncelikli olarak eğitilmesi gerekir. Eğitim işlemi girdi verileri, girdi değişkenleri vektörleridir. Girdi vektöründeki birbirleriyle benzer her bir eleman, ağın girdi katmanındaki girdi nöronlarını oluşturmaktadır. Böylece girdi nöronu sayısının, girdi vektörünün

boyutuna eşit olduğu söylenebilir. Neden-sonuç ilişkisine dayanan bir kestirim probleminde girdi nöron sayısının belirlenmesi çok kolaydır. Probleme ilişkin bağımsız değişken sayısı girdi nöronu sayısını verecektir. Ancak, bir zaman serisi kestirim probleminde uygun girdi nöronu sayısını belirlemek kolay değildir. Boyutu ne olursa olsun, zaman serileri kestirim probleminde girdi vektörü, sabit uzunluktaki serinin ileri doğru hareketli pencerelerinden oluşmalıdır. Eldeki toplam kullanılabilir veri genellikle eğitim ve test kümesi olmak üzere ikiye bölünür. Eğitim kümesi ağın ağırlıklarının belirlenmesinde kullanılırken, test kümesi ağın genelleştirme yeteneğinin ölçümünde kullanılır.

Her modellemede olduğu gibi YSA'nın performansında da modelleme önemli bir yer tuttuğundan dikkatli bir şekilde ele alınmalıdır. Önemli kritik kararlardan biri ağın yapısıdır. Ağ yapısından kasıt, katman sayısı, her katmandaki nöron sayısı ve nöronlar arasındaki bağlantıların sayısıdır. Diğer kritik kararlar ise gizli katman ve çıktı katmanındaki aktivasyon fonksiyonunun seçimi, eğitim algoritması, veri normalleştirme yöntemi, eğitim ve test kümelerinin belirlenmesi ve performans ölçütleridir.

4.3.1. Ağ Yapısı

Tipik bir YSA, nöronlardan oluşan katmanların bileşimidir. Bir ÇKA'larda, bütün girdi nöronları girdi katmanında, bütün çıktı nöronları çıktı katmanında ve gizli nöronlar da bir veya daha fazla katmana dağıtılmıştır. Kestirim için kurulan ÇKA'nın tasarımında aşağıdaki değişkenlerin belirlenmesi gerekir:

1. Girdi nöronu sayısı
2. Gizli katman ve gizli nöron sayısı
3. Çıktı nöronu sayısı

Yukarıdaki parametrelerin seçimi ilgilenilen probleme göre değişecektir. Optimal ağ mimarisinin belirlenmesinde, önerilen bazı yöntemler olmasına rağmen, bu yöntemler oldukça karmaşıktırlar ve yerine getirilmeleri zordur. Ayrıca, bu

yöntemlerder hiçbirisi gerçek kestirim problemleri için en uygun (optimal) çözümü garanti etmemektedir. Daha doğrusu bu parametrelerin belirlenmesinde kesin-açık yöntemler yoktur. Sezgisel yaklaşımlar ve kısıtlı deneylere dayalı benzetim çalışmaları yardımcı olabilir. Bu nedenle, bir YSA'nın tasarımı bir bilimden daha çok bir sanattır (4).

4.3.2. Girdi nöronu sayısı

Girdi nöronu sayısı, girdi vektöründeki değişken sayısına bağlıdır. Neden-sonuç ilişkisine dayalı kestirim problemlerinde girdi nöron sayısını belirlemek kolaydır. Zaman serisi kestirim problemlerinde, girdi nöronu sayısı gecikme sayısı ile ilişkilidir. Ancak bu sayıyı belirlemede önerilen herhangi bir kesin yol yoktur.

Tang ve Fishwick (27) tek değişkenli bir zaman serisi için, girdi nöronu sayısının basitçe Box-Jenkins AR(p) modelinin derecesine eşit olduğunu ileri sürmektedirler. Zhang vd'ne. (4) göre bu aşağıdaki iki nedenden dolayı doğru değildir:

1. Hareketli ortalama MA(q) modelinde, otoregresif (AR) terimler yoktur,
2. Box-Jenkins modelleri doğrusal modellerdir. Otoregresif terimler, geçmiş gözlem değerlerinin, kestirilmek istenen değer ile aralarındaki doğrusal ilişkiyi ifade eder. YSA ise doğrusal olmayan bir modeldir.

Yukarıdaki eleştirilere rağmen, AR(p) modelinin derecesinin bir başlangıç çözümü olarak iyi bir başlangıç olabileceği düşünülebilir. Bu düşünce, AR modellerinin MA modellerine ve MA modellerinin AR modellerine bazı şartları sağlamaları durumunda dönüştürülebilmesinden kaynaklanmaktadır.

Bir çok araştırmacı girdi nöronu sayısını belirlemek için deney tasarımı yaparken, bazıları sezgisel yaklaşımlarda bulunmuşlardır. Örneğin, Sharda ve Patil (28) ve Tang ve diğ. (20)'leri aylık veriler için 12 girdi nöronu ve üç aylık veriler için 4 girdi nöronu kullanmışlardır. Girdi nöronu sayısının belirlenmesine yönelik

literatürde tutarlı bir öneri yoktur. Bazıları fazla girdi nöronu kullanmanın faydalarından söz ederken (20), diğerleri tam tersini ifade etmektedir (29). Lachtermacher ve Fuller fazla girdi nöronunun, birer birer kestirimde kötü sonuçlar verdiğini, ancak çoklu kestirimde iyi sonuçlar verdiğini belirtmektedir.

Veri yapısındaki otokorelasyon bilgisini ortaya çıkarmada önemli olduğu için, girdi nöronu sayısının belirlenmesi belki de YSA'nın kurulmasındaki en kritik karardır (4). Kestirimi yönlendiren uzmanın veriyi iyi tanıması ve verinin zamana bağlı değişiminin niteliğini iyi bilmesi de nöron sayısını belirlemede önemli rol oynar.

4.3.3. Gizli katman ve gizli nöron sayısı

Gizli katman ve gizli nöronlar, YSA'nın başarisında büyük bir öneme sahiptir. Veri içerisindeki belirleyici özellikleri ortaya çıkartan ve girdi ile çıktı arasındaki doğrusal olmayan ilişkinin kurulmasına yardımcı olan, gizli katman ve bu katmanlardaki nöronlardır. Yapılan çalışmalar tek gizli katmanın, doğrusal olmayan karmaşık fonksiyon yaklaşımlarında istenilen herhangi bir doğruluk derecesinde başarılı sonuçlar verdiğini göstermiştir. Birden fazla gizli katman olabilir, ancak bir çok araştırmacı geleceği kestirim için kurdukları ağda tek gizli katman kullanmayı tercih etmişlerdir.

Gizli katman sayısının artırılması, hesaplama zamanını artırmakta ayrıca ağı öğrenme yerine ezberlemesine neden olabilmektedir. Ezberleme, kestirim modelinin serbestlik derecesi çok düşük olduğunda ortaya çıkabilir. Bir başka deyişle, gözlem sayısının model parametre sayısına göre miktarı düşük ise ağ, zaman serisindeki genel yapıyı öğrenme yerine, özel noktaları ezberler. Ağ ağırlıklarının sayısı, gizli katman ve nöron sayısına bağlı olarak değiştiği için eğitim kümesinin büyüklüğü ezberleme işleminin ortaya çıkıp çıkmayacağını belirler. Eğitim kümesine göre çok sayıdaki ağırlık değeri, ağın özel gözlem değerlerini ezberlemesine neden olacaktır (19).

İki gizli katmanlı ağlar, bazı problemler için yararlı olabilir. Zhang (30), iki gizli katmanlı ağların, veri yapısını modellemede ve kestirimin doğruluk derecesini arttırmada tek gizli katmanlı ağlardan daha iyi olduğunu, bazı Santa Fe zaman serileri üzerinde yaptığı çalışmalar sonucunda ifade etmektedir. Zhang (30), ikiden fazla gizli katmanın herhangi bir iyileştirme meydana getirmedeğini de söylemektedir. Zhang ve diğ. (4)'lerine göre, bir çok kestirim problemi için tek gizli katman yeterlidir. Ancak, bazı özel problemlerde, iki gizli katmanlı ağ yapısının tek gizli katmanlı ağ yapısından daha iyi sonuç vermesi olasıdır.

Kaastra ve Boyd (19) bütün YSA'ların öncelikle bir gizli katman ya da en fazla iki gizli katman kurularak çalıştırılmasını önermektedir. Eğer dört katmanlı bir ağ farklı gizli nöron sayısı ve rasgele başlangıç değerleri ile çalıştırıldığında iyi sonuç vermiyorsa, girdi değişkenlerinin gözden geçirilmesi ve yeniden düzenlenmesinde fayda vardır. Teori ve hemen hemen bütün deneysel çalışmalar ikiden fazla gizli katmanın ağın performansını iyileştirmedeğini ifade etmektedir.

Bir ağda gizli nöron sayısının belirlenmesi kritik bir karardır. Ancak önemine rağmen en iyi gizli nöron sayısının belirlenmesinde herhangi bir sihirli formül yoktur. Genelde az sayıda gizli nöron ile çalışma tercih edilir. Çünkü genelleştirme yetenekleri daha yüksektir. Çok sayıda gizli nöron, ağın genelleştirmeden daha çok ezberleme yeteneği kazanmasına sebep olur. Bu da istenilen bir durum değildir. Ancak, çok az sayıda gizli nöron da ağın öğrenmesi için yeterli değildir. Bu nedenle araştırmacılar gizli nöron sayısını belirlemede deneysel bir çalışmaya başvurabilirler. Ancak sistematik bir yöntem olarak, küçük bir başlangıç değeri ile başlanması ve ağ performansı iyileşene kadar gizli nöron sayısının artırılması önerilebilir (19).

Gizli nöron sayısını belirlemede deneme-yanılma yöntemi sıklıkla kullanılır. Ancak, yine de bazı deneysel çalışmalar gizli nöron sayısını sınırlayıcı öneriler sunmaktadır. Tek gizli katmanlı ağlarda gizli nöron sayısına ilişkin bazı öneriler şunlardır:

- a) n
- b) $n/2$
- c) $2n+1$
- d) $2n$
- e) $\sqrt{n * m}$
- f) $0,75 * n$

Yukarıdaki ifadelerde n girdi nöronu ve m çıktı nöronu sayısını göstermektedir. Bu yaklaşımlardan hiçbirisinin bütün problemlerde iyi sonuç verdiğini söylemek doğru olmaz (4). Tang ve Fishwick (27), gizli nöron sayısının kestirim performansında etkili olduğunu ancak bu etkinin çok da önemli olmadığını belirtmektedir. Zhang (5), ise girdi nöron sayısı ile gizli nöron sayısının eşit olduğu ağ yapılarında daha iyi kestirimde bulunulduğunu, bir çok araştırmancının bunu ortaya koyduğunu söylemektedir.

4.3.4. Çıktı nöronu sayısı

Çıktı nöronu sayısını hesaplamak kolaydır ve çalışılan probleme doğrudan bağlıdır. Bir zaman serisi kestirim probleminde, çıktı nöronu sayısı, kestirim döneminin uzunluğuna eşittir. İki şekilde kestirimde bulunulabilir:

1. Tek dönemlik kestirim,
2. Çok dönemlik kestirim.

Tek dönemlik kestirimde çıktı nöronu sayısı 1'e eşittir. Çok dönemli kestirim ise iki yolla yapılabilir. Birincisi, Box-Jenkins modelindeki gibi iteratif kestirimdir. Kestirim edilen dönem değeri, bir sonraki dönem için girdi olarak kullanılır. Bu durumda yine sadece tek çıktı nöronu yeterlidir. İkincisi ise, birden fazla dönemin aynı anda kestirildiği doğrudan yaklaşım yöntemi (*direct method*) olarak adlandırılan durumdur. Bu durumda çıktı nöronu sayısı kestirilmek istenen dönem sayısına eşittir. Zhang (30), her iki yöntemi de deneyerek çoklu dönem kestiriminde

bulunmaya çalışmıştır. Zhang (30)'in belirttiğine göre aynı anda birden fazla dönemin kestirildiği doğrudan yaklaşım yöntemi daha iyi sonuçlar vermiştir.

Zhang ve diğ. (4)'leri, doğrudan çoklu dönem kestiriminin aşağıdaki nedenden dolayı daha iyi sonuç vereceğini ifade etmektedir: YSA, doğrudan çoklu dönem kestiriminde bulunacak şekilde kurulabilir. Box-Jenkins modelinde olduğu gibi, iteratif yöntem sadece bir dönem kestirimde bulunmak üzere kullanılabilir. Bulunan kestirim değeri girdi olarak kullanılır ve bir sonraki dönem kestirilir. Kestirilecek dönem sayısı uzun bir dilim ise geçmiş gözlem değerleri kullanılmadan, sadece bulunan kestirim değerlerinin girdi olarak kullanıldığı bir durumla karşılaşılır. Bu da kestirimin doğruluğunu azaltır. Bu nokta, Box-Jenkins modellerinin neden kısa dönemli kestirim için daha uygun olduğunun da bir yanıtı niteliğindedir. Aşağıdaki eşitlikler bu durumu açık olarak gösterir:

$$\begin{aligned}
 F_{t+1} &= f(Y_t, Y_{t-1}, \dots, Y_{t-n}), \\
 F_{t+2} &= f(F_{t+1}, Y_t, Y_{t-1}, \dots, Y_{t-n+1}), \\
 F_{t+3} &= f(F_{t+2}, F_{t+1}, Y_t, Y_{t-1}, \dots, Y_{t-n+2}), \\
 &\dots\dots\dots \\
 F_{t+k} &= f(F_{t+k-1}, F_{t+k-2}, \dots, F_{t+1}, Y_t, Y_{t-1}, \dots, Y_{t-n+k-1}),
 \end{aligned} \tag{4.3}$$

Y_t , t dönemi gözlem değerini, F_t , t dönemi kestirim değerini, f ise YSA'nın atadığı fonksiyonu temsil etmektedir. Doğrudan çoklu dönem kestiriminde ise, YSA, bütün geçmiş dönem gözlem değerlerini kullanarak, k dönem gelecek değeri aynı anda talimin edebilir. Bu k adet çıktı nöronu kullanarak sağlanabilir. Bu durumda yukarıdaki eşitlikler aşağıdaki gibi düzenlenir:

$$\begin{aligned}
 F_{t+1} &= f_1(Y_t, Y_{t-1}, \dots, Y_{t-n}), \\
 F_{t+2} &= f_2(Y_t, Y_{t-1}, \dots, Y_{t-n+1}), \\
 F_{t+3} &= f_3(Y_t, Y_{t-1}, \dots, Y_{t-n}), \\
 &\dots\dots\dots \\
 F_{t+k} &= f_k(Y_t, Y_{t-1}, \dots, Y_{t-n}), \\
 f_1, f_2, \dots, f_k &\text{ ağ tarafından belirlenen fonksiyonları temsil etmektedir.}
 \end{aligned} \tag{4.4}$$

4.3.5. Nöronlar arası bağlantılar

Ağ yapısı, katman ve nöron sayısına göre şekillenmekle beraber, nöronlar arasındaki bağlantılar da ağ yapısını karakterize etmektedir. Nöronlar arasındaki bağlantılar, temel olarak ağın davranışını belirlemektedir. Bir çok kestirim probleminde, diğer uygulamalarda da olduğu gibi, girdi katmanındaki ve gizli katmandaki nöronlar, kendilerinden sonra gelen katmandaki nöronlar ile tam bir bağlantı içerisindeyler. Ancak, seyrek bağlantılı (bütün nöronlar arasında bağlantının olmadığı) ağ yapıları da vardır. Yine, farklı olarak, girdi katmanından çıktı katmanına direkt bağlantıların olduğu ağ yapıları da vardır (31).

4.3.6. Aktivasyon fonksiyonu

Aktivasyon fonksiyonu, transfer fonksiyonu olarak da adlandırılmaktadır. Bir nöronun veya ağın girdisi ve çıkışı arasındaki ilişkiyi belirlemektedir. Uygulamada, sınırlı, tekdüze artan ve türevi alınabilen aktivasyon fonksiyonları kullanılmaktadır. Daha önceki bölümlerde de değindiğimiz gibi bunlardan bazıları şunlardır:

1. Sigmoid (Lojistik) Fonksiyonu
2. Hiperbolik Tanjant (tanh) Fonksiyonu
3. Doğrusal Fonksiyon

Bunların arasında en yaygın kullanılanı, Sigmoid transfer fonksiyonudur.

Genel olarak, bir ağın, aynı ya da farklı katmanlarındaki nöronlar farklı aktivasyon fonksiyonunu kullanabilirler. Uygulamaların çoğunda ise, aynı katmandaki nöronların aynı aktivasyon fonksiyonunu kullandıkları görülmektedir.

Lojistik aktivasyon fonksiyonu, ikili hedef değerlerin olduğu sınıflandırma problemleri için çıktı katmanında sıklıkla kullanılmıştır. Ancak, hedef değerlerin

süreklilik gösterdiği kestirim problemi gibi problemlerde, çıktı katmanında doğrusal bir aktivasyon fonksiyonunun kullanılması tercih edilmelidir.

Burada şu noktanın unutulmaması gerekir: ağın ürettiği çıktığı değerleri, kullanılan fonksiyona göre $[0,1]$ veya $[-1,1]$ aralığında olacağı için, hedef değerler de kullanılan fonksiyonla uyumlu bir şekilde yukarıdaki aralıkları dikkate alarak normalleştirilmelidir.

4.3.7. Öğrenme algoritması

Bir YSA'nın eğitimi, ağ ağırlıklarının, ağ çıktısı ile istenilen değer arasındaki toplam ya da ortalama hata karelerinin enküçüklenmesi için sürekli olarak değiştirildiği, sınırsız doğrusal olmayan enküçüklenme problemidir. Ağ eğitimi için bir çok değişik optimizasyon yöntemi vardır. Ancak, genel bir doğrusal olmayan optimizasyon problemi için uygun bir zamanda genel optimumu garanti eden herhangi bir algoritma yoktur. Uygulamada bütün optimizasyon algoritmalarının sıkıntısı yerel optimuma takılmaktır. Genel çözümün mümkün olmadığı durumlarda, en iyi yerel optimumu veren algoritmaları kullanmak akıllıcadır.

En yaygın kullanılan öğrenme (eğitim) algoritması, gerçekte bir gradyan dik iniş algoritması olan geri yayılım algoritmasıdır. Gradyan iniş algoritması için YSA literatüründe öğrenme oranı olarak adlandırılan bir adım büyüklüğü tanımlanmalıdır. Öğrenme oranı, ağırlıkların değişim büyüklüğünü belirlediği için geri yayılım öğrenme algoritmasında çok önemlidir. Yavaş ilerlemesi ve verimsizliği dik iniş algoritmasının yetersiz yönleridir. Ayrıca, öğrenme oranındaki değişimlere duyarlıdır. Küçük öğrenme oranı, öğrenme sürecinin yavaş ilerlemesine sebep olurken, büyük öğrenme oranı da ağırlık uzayında ağ ağırlıklarının sarkaç gibi salınım göstermesine sebep olur. Orijinal gradyan iniş algoritmasını geliştirmenin bir yolu, algoritmaya bir momentum parametresinin eklenmesidir. Momentum parametresi, büyük öğrenme oranıyla eğitim sürecinin hızlanmasını sağlarken, salınma eğilimini de en aza indirmeye yardımcı olur.

Momentum terimini içeren standart geri yayılım tekniği, bir çok araştırmacı tarafından uyarlanmıştır. Öğrenme oranının ve momentum katsayısının eş zamanlı olarak seçiminde sadece birkaç sistematik yol olduğu için, bu öğrenme parametrelerinin en iyi değerleri genellikle deneme yolu ile seçilir. Öğrenme oranı ve momentum katsayısı, her ikisi de genellikle 0 ile 1 arasında herhangi bir değer alabilir. Ancak bu iki terimin en iyi kombinasyonunu bulmak imkansızdır. Araştırmacılar seçtikleri değerleri dikkate almıştır. Örneğin, Sharda ve Patil (28), üç öğrenme oranının (0,1 , 0,5 , 0,9) ve üç momentum katsayısının (0,1 , 0,5 , 0,9), dokuz kombinasyonunu denemiştir.

Tang ve Fishwick (27) öğrenme parametrelerinin YSA'nın performansında kritik bir rol oynadıklarını söylemektedir. Yazarlar, daha önceki çalışmalarda, YSA ile kötü sonuçlar alındığı rapor edilen bir çok zaman serisini, değişik öğrenme parametreleri kullanarak test etmiştir. Bu çalışma sonucunda, her bir zaman serisi eğer uygun öğrenme parametreleri uygulanırsa, iyi sonuçlar alınacağını belirtilmiştir. Tang ve diğ. (20)'leri, öğrenme parametrelerinin YSA'nın öğrenmesindeki etkisini araştıran bir çalışma yapmıştır. Yazarlar, karmaşıklığı az olan verilerde yüksek öğrenme oranının iyi olduğunu ve çok karmaşık verilerde ise düşük öğrenme oranını ve yüksek momentum katsayısının kullanışlı olduğunu rapor etmektedir.

Geleneksel geri yayılım algoritmasının zayıf yönleri doğrultusunda, geri yayılım algoritmasının bazı varyasyonları ve modifikasyonları önerilmiştir. Bunların arasında en etkili olanları ikinci derece (Levenberg-Marquardt) yöntemlerdir. Hızlı çalışmaları ve en iyi yerel minimumu bulmadaki başarıları, ikinci derece yöntemlerin YSA eğitiminde tercih edilmelerini sağlamıştır (4).

4.3.8. Veri normalleştirme

Daha önce de belirttiğimiz gibi, eğer çıktı nöronları için doğrusal olmayan bir aktivasyon fonksiyonu kullanılmış ise, hedef değerlerin ağ çıktısı ile aynı aralıkta olacak şekilde dönüşüm uygulanması gerekir. Lojistik fonksiyonu gibi doğrusal olmayan aktivasyon fonksiyonları, bir nöronun çıktısını $[0,1]$ veya $[-1,1]$ aralığına

sıkıştırma gibi bir rol üstlenirler. Eğer çıktı nöronlarında doğrusal bir aktivasyon fonksiyonu kullanılmış ise, hesaplama hatalarından kaçınmak için hem çıktıları hem de girdileri standartlaştırma avantajlı olacaktır. Veri normalleştirme (*data normalization*), eğitime süreci başlamadan uygulanır. Veri normalleştirme yaklaşımlarında, sıklıkla aşağıdaki formüller kullanılmaktadır:

- a) $[0,1]$ aralığına doğrusal dönüşüm: $x_{\text{norm}} = (x_0 - x_{\text{min}}) / (x_{\text{maks}} - x_{\text{min}})$,
- b) $[a,b]$ aralığına doğrusal dönüşüm: $x_{\text{norm}} = (b-a)(x_0 - x_{\text{min}}) / (x_{\text{maks}} - x_{\text{min}}) + a$,
- c) Basit normalizasyon: $x_{\text{norm}} = x_0 / x_{\text{maks}}$, (29)
- d) İstatistiksel normalizasyon: $x_{\text{norm}} = (x_0 - \bar{x}) / s$,

x_{norm} normalleştirilmiş, x_0 orijinal veriyi temsil etmektedir. x_{min} , x_{maks} , $\bar{x}$ ve s , sırasıyla satır veya kolon boyunca minimum, maksimum, ortalama ve standart sapmayı ifade etmektedir.

Ağın öğrenmesinde veri normalleştirmenin önemini araştıran bazı çalışmalar yapılmıştır. Yazarlar, veri normalleştirmenin genelde faydalı olduğu kanısına varmıştır. Ancak örnek büyüklüğü arttığında bu faydanın azaldığını belirtmektedirler.

Çıktı hedeflerinin normalleştirilmesi, genellikle girdilerin normalleştirilmesinden bağımsızdır. Zaman serileri kestirim probleminde, girdilerin ve hedeflerin normalleştirilmesi birlikte yapılır. Girdilerin ve hedeflerin normalleştirme aralığının seçimi, çıktı nöronlarının aktivasyon fonksiyonuna bağlıdır. Eğer sigmoid fonksiyonu kullanılmış ise normalleştirme aralığı $[0,1]$, hiperbolik tanjant fonksiyonu kullanılmış ise normalleştirme aralığı $[-1,1]$ olarak seçilir.

Hedef değerleri normalleştirmenin bir sonucu olarak, gözlemlenen ağ çıktıları normalleştirme aralığı ile uyumlu olmalıdır. Ağdan alınan sonuçların yorumlanması, çıktıların orijinal aralığa dönüştürülmesinden sonra olabilir. Ağın ürettiği değerlerin doğruluğu orijinal veri kümesi temel alınarak hesaplanmalıdır. Performans ölçütü de, çıktıların orijinal aralığa dönüştürülmesinden sonra hesaplanmalıdır.

Verileri normalleştirmenin bir diğer yolu da, bütün verileri eğitim kümesindeki en büyük veriden daha büyük bir sayıya bölmektir. Lachtermacher ve Fuller (29), bu büyük sayıyı normalleştirme faktörü olarak adlandırmaktadır. Yazarlar, bu sayının eğitim kümesindeki en büyük sayının % 30- % 100 fazlası olabileceğini belirtmektedirler. Bu yaklaşım, eğitim kümesindeki verilerden daha büyük değerlerde kestirimin ortaya çıkabileceği durumda oluşabilecek problemleri ortadan kaldırmaya yöneliktir. Yazarlar, gelecek dönemlerdeki kestirim değerlerinin, normalleştirme faktöründen de büyük olması durumunda iki defa normalleştirme yapılabileceğini söylemektedir.

4.3.9. Eğitim kümesi ve test kümesi

Daha önce de değinildiği gibi, bir YSA kestiricisinin kurulumunda eğitim ve test örneklerine ihtiyaç vardır. Eğitim örnekleri, YSA modeli geliştirmek için kullanılırken, test örnekleri ise, geliştirilen modelin kestirim yeteneğinin değerlendirilmesinde kullanılır. Bazen, doğrulama kümesi olarak adlandırılan bir üçüncü küme kullanılır. Doğrulama kümesi, eğitim sürecinin durdurulacağı noktanın belirlenmesinde veya fazla eğitime probleminden kaçınmada kullanılabilir. Zhang ve diğ. (4)'lerine göre, eğitim ve test örneklerinin seçimi, YSA'nın performansını etkilemektedir.

İlk olarak, veri, eğitim ve test kümesi olmak üzere ikiye bölünür. Ancak bunu yapmanın genel bir yöntemi yoktur. Problem özelliği, veri tipi ve eldeki veri miktarı bu kararı vermede dikkate alınır. Hem eğitim kümesinin hem de test kümesinin bütünün özelliklerine sahip kümeler olması kritik bir noktadır. Bu nokta özellikle zaman serileri kestirim problemlerinde önemlidir. Eğitim ve test kümelerinin uygunsuz belirlenmesi, optimal YSA yapısının seçimini ve YSA'nın kestirim performansının değerlendirilmesini etkileyecektir. Literatürde eğitim ve test kümelerinin belirlenmesine yönelik az da olsa öneriler vardır. Bir çok araştırmacı % 90, % 10 veya % 80, % 20 ya da % 70, % 30 kuralını temel alan bir yöntem izlemiştir (4).

YSA kestiricisinin performansını etkileyen bir diğer yakın ilgili faktör de örnek büyüklüğüdür. Verilen bir problemin çözümü için ne kadar büyüklükte bir örnek gerektiği hakkında kesin bir kural yoktur. Ağın eğitiminde gerekli veri miktarı ağın yapısına, eğitim yöntemi, çalışılan problemin karmaşıklığına bağlıdır. Genel olarak, herhangi bir istatistiksel yaklaşımdaki gibi, örnek büyüklüğü, çözümün doğruluğu ile yakından ilgilidir. Örnek büyüklüğü ne kadar fazla olursa, sonuçların doğruluğu da o oranda artar.

Yeterli büyüklükte bir örnek kullanıldığında YSA, verideki herhangi bir karmaşık yapıyı modelleyebilir. Bu nedenle, büyük örneklerde bir YSA, doğrusal istatistiksel yöntemlerden daha fazla yararlı olabilir. YSA, doğrusal modellerin gerek duyduğundan daha fazla bir veriye gereksinim duymamaktadır. Box-Jenkins modelleri ise başarılı bir kestirim için en az 50 veriye gereksinim duymaktadır (4).

4.3.10. YSA'nın performansının belirlenmesi

Bir YSA kestiricisi için, modelleme zamanı veya eğitim zamanı gibi bir çok performans ölçütü olabilirken, en iyi ve en önemli performans ölçütü kestirimin doğruluğudur. Ancak, kestirim akademisyenleri ve pratisyenlerinin verilen bir problem için yaygın olarak kabul ettiği uygun bir doğruluk ölçütü yoktur. Doğruluk ölçüsü, gerçek değer/istenilen değer ile kestirilen değer arasındaki fark olarak tanımlanır. Bu fark kestirim hatası diye bilinir. Kestirim literatüründe bir çok doğruluk ölçüsü tanımlanmıştır ve her birinin kendine göre avantajları ve yetersizlikleri vardır (13). Sıklıkla kullanılan performans ölçüsü Bölüm 2.5 'te verilmiştir.

Her bir ölçünün kendine göre kısıtlamaları olduğundan dolayı, herhangi bir özel problem için birden fazla performans ölçüsü kullanılabilir.

4.4. YSA'nın Kestirimde Göreceli Performansı

YSA'nın bir kestirim aracı olarak performansı, yaygın kullanılan istatistiksel yöntemler ile karşılaştırılmıştır. Literatürde birbiri ile tutarsız bir çok çalışma vardır. Bunun ana sebebi, YSA'nın kestirim yeteneğini etkileyen, ağ yapısı, eğitim yöntemi, örnek veri gibi bir çok etken olabilir. Bazı durumlarda, YSA, doğrusal istatistiksel modellerden daha kötü sonuç verebilir. Bu, verinin doğrusal olmasından kaynaklanabilir. Doğrusal bir ilişkiye sahip veri yapısında, YSA'nın doğrusal modellerden daha iyi sonuç vermesi beklenemez. Diğer yandan, ideal ağ yapısının kullanılmadığı da söylenebilir. Tang ve diğ. (20)'leri, Tang ve Fishwick (27) çalışmalarında YSA kestiricisinin hangi koşullarda, Box-Jenkins modelleri gibi geleneksel zaman serisi kestirim yöntemlerinden daha iyi sonuç verdiği sorusunu cevaplamaya çalışmışlar ve aşağıdaki sonuçlara ulaşmışlardır:

1. YSA, kestirim dönemi uzunluğu artıkça daha iyi sonuç vermektedir.
2. Gözlem sayısının az olduğu zaman serilerinde, YSA daha iyi sonuç vermektedir. Sharda ve Patil (28), benzer sonuca ulaşmıştır.
3. Daha fazla girdi nöronu ile YSA daha iyi sonuç vermektedir.

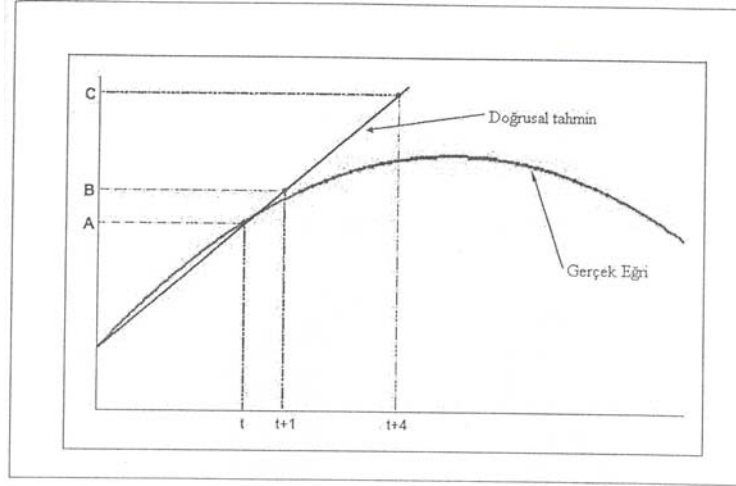
4.5. YSA'nın İstatistiksel Yöntemlere Göre Güçlü ve Zayıf Yanları

Daha önceki bölümlerde YSA'nın değişik alanlarda kestirim amaçlı olarak kullanılmasına ve diğer yöntemlerle performanslarının karşılaştırılmasına yer verildi. Bu bölümde, YSA'nın istatistik tekniklere kıyasla güçlü ve zayıf olduğu yanları özetlenecektir.

4.5.1. YSA'nın güçlü olduğu yanları

YSA'nın güçlü olduğu yanları dört maddede özetlenebilir:

1. Doğrusal olmayan ilişkiyi başarı ile modelleyebilir:



Şekil 4.1 Doğrusal kestiricisinin eğrisel bir yapıyı kestirimindeki performansı

YSA'nın doğrusal olmayan aktivasyon fonksiyonuna sahip olması, ilgili problemin yapısındaki ilişki doğrusal olmadığı durumlarda daha etkin bir model üretilmesini sağlar. Şekil 4.1 doğrusal olmayan bir ilişkinin doğrusal bir model ile nasıl kestirilmeye çalışıldığını göstermektedir. Şekil 4.1'e göre, 0 noktasından t noktasına kadar olan veriler $t+1$ anındaki gözlem değerinin kestirimi için kullanıldığında doğrusal model, B'yi kestirmede iyi bir kestirici olarak işlev görebilir, ancak daha ileriki noktalar için örneğin $t+4$ anında doğrusal model C değerini üretirken bu değer gerçek değerden oldukça farklıdır. Bu durum, YSA'nın uzun dönemli kestirimde doğrusal modellerden neden daha iyi sonuç verdiğini de açıklamaktadır (33).

2. Veri yapısındaki fonksiyonel ilişki için herhangi bir ön bilgiye gereksinim duymaz:

Bazı doğrusal olmayan modellerin de YSA kadar iyi sonuçlar verebileceği ileri sürülebilir. Ancak bu durum doğrusal olmayan yapı hakkında bir ön bilgiye sahip olduğunda geçerli olabilir. Örneğin doğrusal olmayan regresyon modellerinin doğrusal olmayan bir ilişkinin var olduğu bir problemde, YSA kadar iyi sonuç verebileceği iddia edilebilir. Teorik olarak bu doğrudur. Ancak uygulamada, araştırmacının doğrusal olmayan regresyon modelinin derecesi ve biçimi hakkında bir varsayımda bulunması gerekir. YSA için ise böyle bir durum söz konusu değildir.

3. Veri kaybı yoktur:

YSA, Box-Jenkins modellerinde olduğu gibi seriyi durağanlaştırmak için fark alma gibi bir işlem yapmadığından herhangi bir veri kaybına sebep olmaz.

4. Ağ yapısı esnektir:

YSA'nın dördüncü bir avantajı da ağ yapısının esnekliğidir. İstatistik tekniklerin (doğrusal regresyon, ikili probit model, otoregresif modeller vb.) büyük bir çoğunluğu, ağın aktivasyon fonksiyonunda yapılacak çok küçük değişikliklerle YSA için kullanılabilir hale getirilir. Bu da aynı ağ yapısının esnekliği gösterir.

4.5.2. YSA'nın zayıf olduğu yanları

YSA'nın zayıf olduğu yanları dört maddede kısaca açıklanabilir:

1. Kara kutu problemi:

Ağın ağırlıklarının yorumlanması zordur. Doğrusal regresyon modelinde bağımsız değişkenlerin katsayıları, bağımlı değişken üzerindeki etkilerini açık olarak gösterirler. Ancak YSA'da, girdi değişkenlerinin çıktı üzerindeki etkilerini analitik olarak belirlemek çok zordur.

2. Genel minimumu bulamama:

Bütün doğrusal olmayan kestirim metotlarında olduğu gibi YSA'da da genel minimumu bulamama riski vardır. Ancak, genel minimuma yakın yerel minimumlar da oldukça iyi sonuçlar verebilir.

3. Örnek sayısının büyüklüğü:

YSA'nın tam olarak genelleştirme yapabilmesi için yeterli miktarda örnekle ağ eğitilmelidir. Aksi durumlarda veri yapısındaki ilişkiyi ortaya çıkartamama ya da yanlış eğitime (*overfitting* -ezberleme) durumları ile karşılaşılabilir.

4. Modellemenin çok zaman alması:

Bir YSA'nın kurulumu ve eğitimi çok zaman alabilir. Bu da araştırmacılar için bezdirici bir durumdur.

5. BULGULAR

5.1. Giriş

YSA'nın zaman serileri kestirimindeki performansı, gerçek zaman serileri ile çalışılarak araştırılmıştır. Gerçek zaman serileri; Refik Saydam Hıfzısıhha Başkanlığı'nın, 1999 Yılı Ocak Ayı'ndan 2005 Yılı'nın Kasım sonuna kadarki, Ankara ili hava kalitesi ölçüm raporlarıdır. Hava kirliliğinin önemli göstergelerinden biri olan günlük SO₂ (Kükürtdioksit) ölçümlerinin aylık ortalaması alınarak elde edilen 83 adet aylık veri, YSA ve Box-Jenkins Yöntemleri ile kestirilmiştir.

Tablo 5.1. Zaman Serisi Uygulama Verisi*

Aylar	1999	2000	2001	2002	2003	2004	2005
Ocak	80	74	86	105	82	72	95
Şubat	50	69	57	102	47	75	62
Mart	57	60	49	74	59	64	51
Nisan	35	32	31	62	54	60	51
Mayıs	29	29	31	37	40	32	35
Haziran	25	26	23	38	35	28	24
Temmuz	26	33	25	45	37	22	27
Ağustos	29	23	27	46	42	33	32
Eylül	42	49	36	48	45	46	41
Ekim	66	64	47	60	59	40	63
Kasım	105	126	81	113	86	66	87
Aralık	129	106	53	99	87	92	

* Refik Saydam Hıfzısıhha Merkezi Başkanlığı, 1999-2005 (Kasım sonu) Ankara İli Hava Kalitesi SO₂ (Kükürtoksit) Ölçüm Sonuçları.

SO₂ hava kirliliğini oluşturan birincil kirleticiler grubundadır. Kömür ve fuel-oil'in doğal olarak yapısında bulunan kükürt bileşiklerinin yanması ile açığa çıkmaktadır. Dünya çapındaki temel kaynakları, endüstriyel işlemler, ısınma amaçlı kullanılan evsel yakıtlar ve termik santrallerdir. Çok az miktarı ise dizel yakıtlı taşıt araçlarından kaynaklanmaktadır. SO₂'nin yüksek konsantrasyonları öksürük ve bunun sonucunda akciğer fonksiyonlarında değişime neden olarak solunum sistemi tahribatına neden olmaktadır. Bu gaz ayrıca taş binaların ve diğer materyallerin de korozyonuna neden olur, bitkilere zarar verebilir ve asit yağmurlarının ve ikincil partiküllerin temel kaynağıdır.

Bu çalışmada, Ankara ili SO₂ ölçümlerinin kestirim değerleri, gerçek değerler ile karşılaştırılarak hangi kestirim yönteminin daha iyi sonuç verdiği istatistiksel performans değerleri ile açıklanmıştır. Ayrıca sonraki 13 aylık dönem için kestirimler elde edilmiştir.

Yapılan analizlerde aşağıdaki sorulara yanıt bulunmaya çalışılmıştır:

1. GYA ile eğitilen YSA'nın, zaman serileri kestiriminde Box-Jenkins modellerine göre performansı nasıldır?
2. En düşük OMYH değerini hangi model vermektedir?

5.2. Kestirimlerin üretilmesi

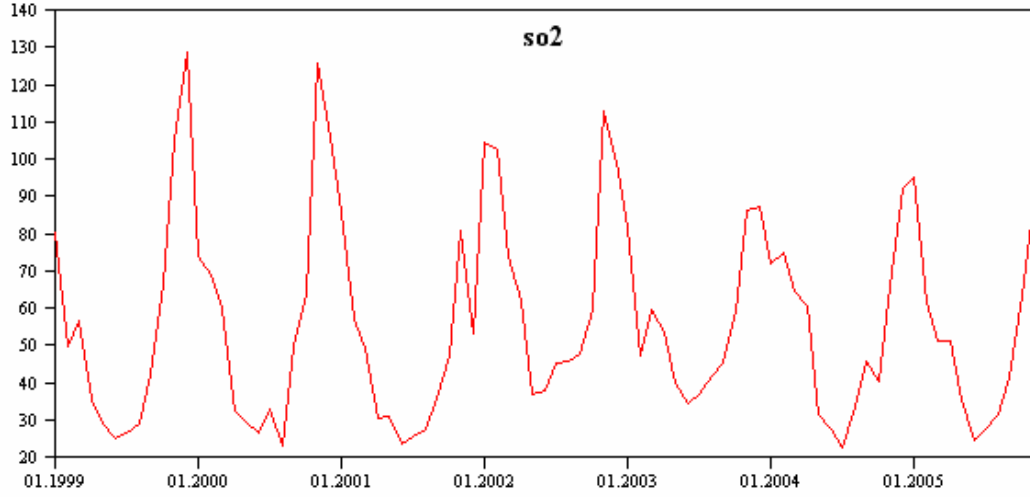
Verinin Box-Jenkins yöntemi ile kestirimi için SPSS- Decision Time (34) ve YSA teknikleri ile kestirimi için de MATLAB Neural Network ToolBox 7.0 (35) programları kullanılmıştır.

5.3. Box-Jenkins Yöntemi ile SO₂ verilerinin kestirimi

Box-Jenkins modelinin SO₂ verilerine göre kurulumu aşamalarına bu bölümde değinilecektir.

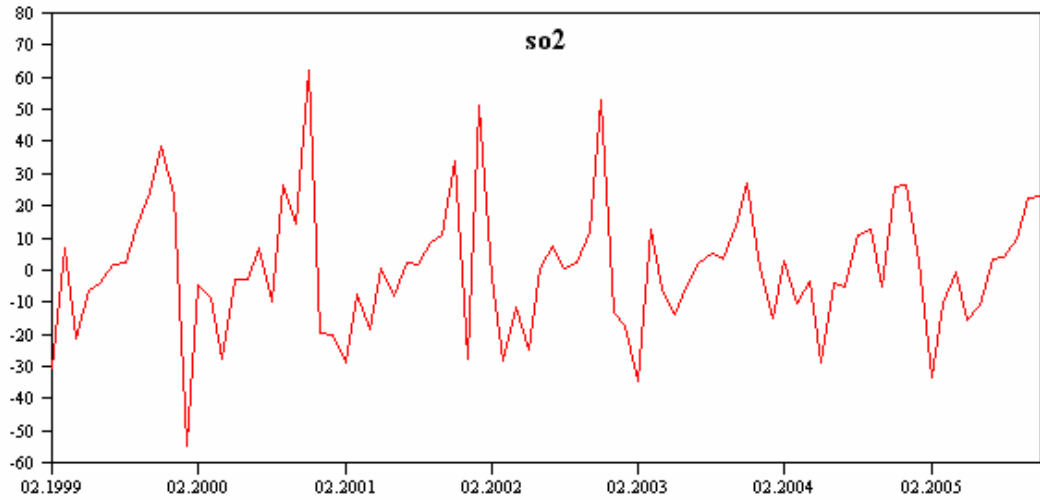
5.3.1. Model tanımlama ve parametre tahmin aşaması

Modellemeye geçmeden önce model belirleme aşamasında otokorelasyon ve kısmi otokorelasyon fonksiyonlarına başvurulur. Şekil 5.1.'de yer alan SO₂ verilerinin zamana bağlı grafiğinde ilk olarak mevsimsellik göze çarpmaktadır.



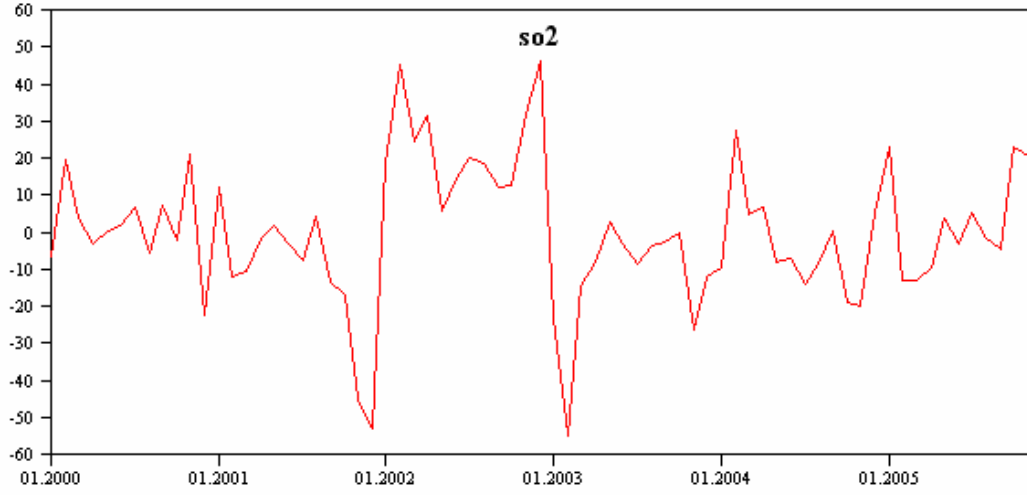
Şekil 5.1. SO₂ verilerinin zaman serisi

Kurulacak modelin belirlenmesi için serinin durağanlaştırılması gereklidir. Şekil 5.2.'de serinin durağanlaştırılmış grafiği yer almıştır.



Şekil 5.2. SO₂ verilerinin durağanlaştırılmış zaman serisi grafiği

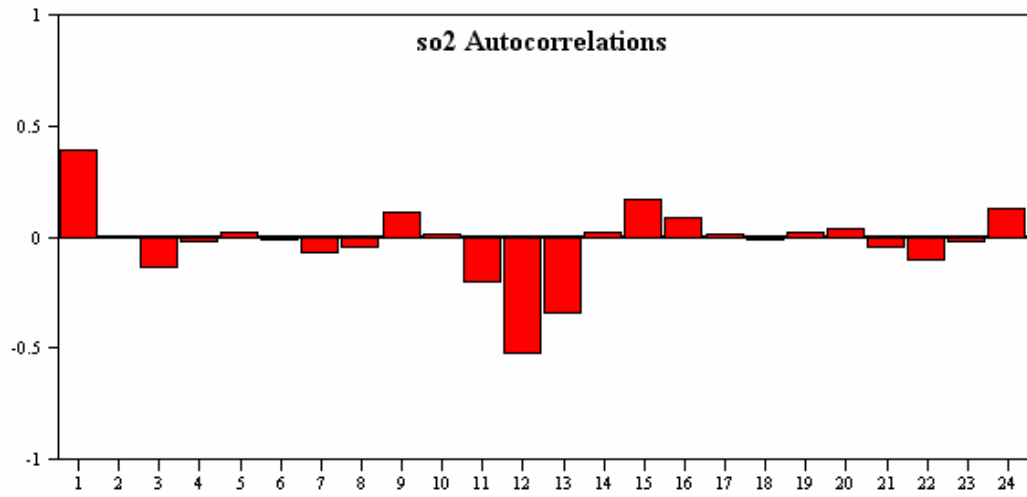
Durağanlaştırma işlemi sonrasında da mevsimselliğin devam ettiği gözlenmiş ve mevsimsel durağanlaştırma işlemi yapılmıştır. Şekil 5.3. mevsimsel durağanlaştıma uygulanmış zaman serisini göstermektedir.



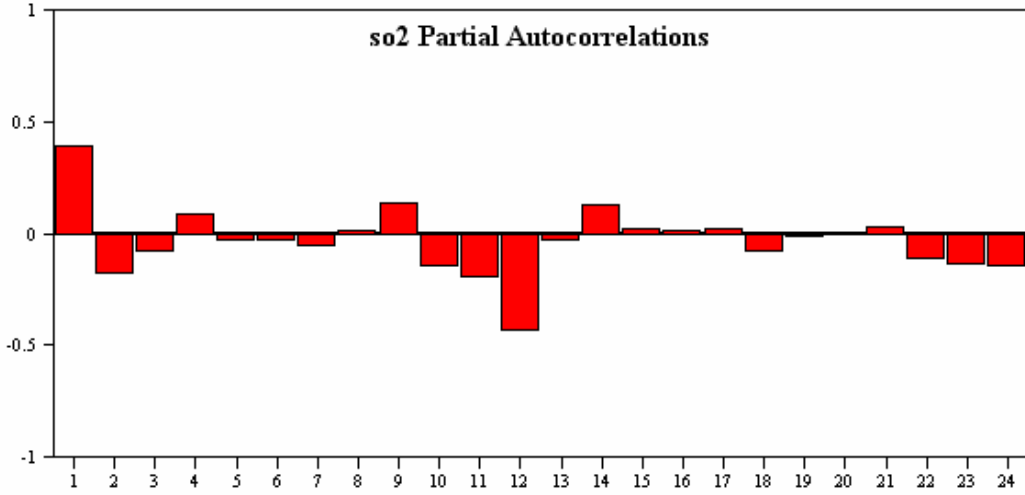
Şekil 5.3. SO₂ verilerinin mevsimsel durağanlaştırılmış zaman serisi grafiği

Mevsimsel durağanlaştırma 1. fark alma işlemi işlemi sonrasında elde edilmiştir. Bu nedenle $d=1$ 'dir.

Modelin $AR(p)$ ve $MA(q)$ parametreleri otokorelasyon ve kısmi otokorelasyon fonksiyonlarına başvurularda tahmin edilmiştir.



Şekil 5.4. Mevsimsel durağanlaştırılmış serinin otokorelasyonları



Şekil 5.5. Mevsimsel durağanlaştırılmış serinin kısmi otokorelasyonları

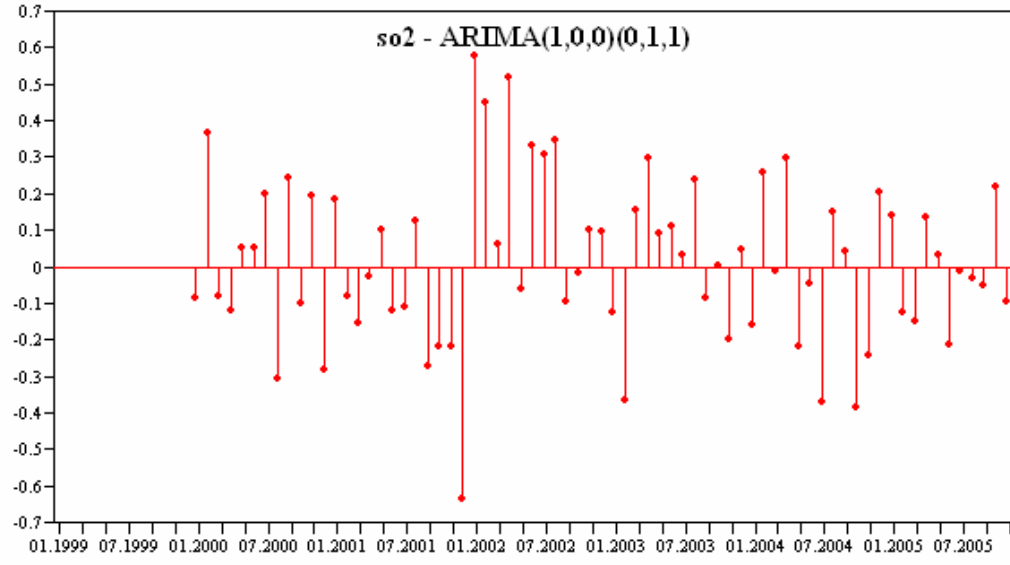
Şekil.5.4 ve şekil 5.5’te yer alan otokorelasyon ve kısmi otokorelasyon fonksiyonları ARMA(1,1) sürecinin modelde uygun olabileceğini göstermektedir.

SO₂ serisi, en küçük hata kareler ortalamasını veren ARIMA (1 0 0) (0 1 1) modeli ile açıklanmıştır.

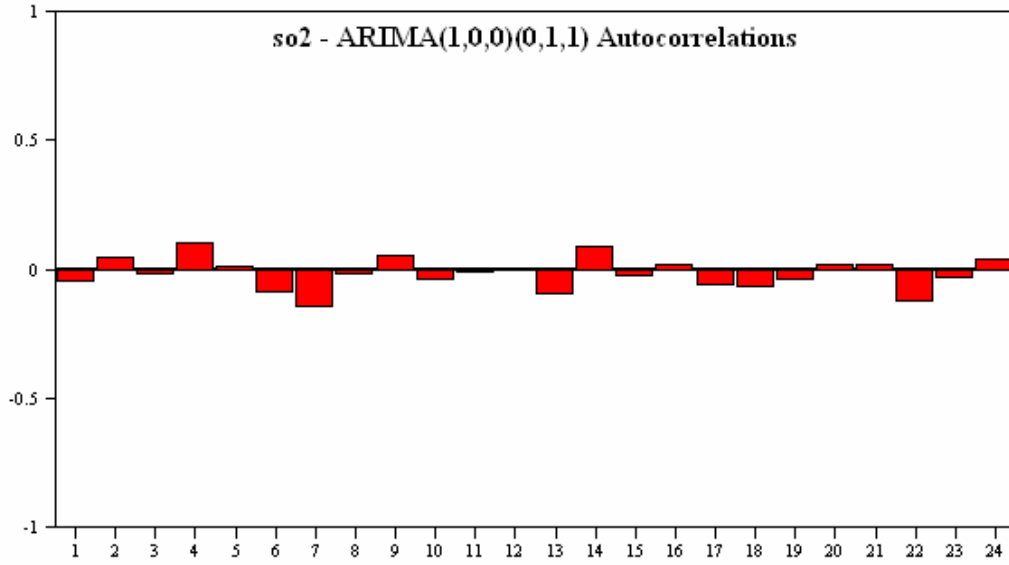
5.3.2. Modelin geçerliliği

Modelin geçerliliğinin sınanması Box-Ljung istatistiğinin testi ile yapılmıştır. Box-Ljung istatistiği 6.0476 ‘dır. Bu değer $p=0.05$ seviyesinde anlamlı değildir. Bu sonuç kestirim hatasının rassal olduğunu ve serinin model varsayımlarını karşıladığını gösterir.

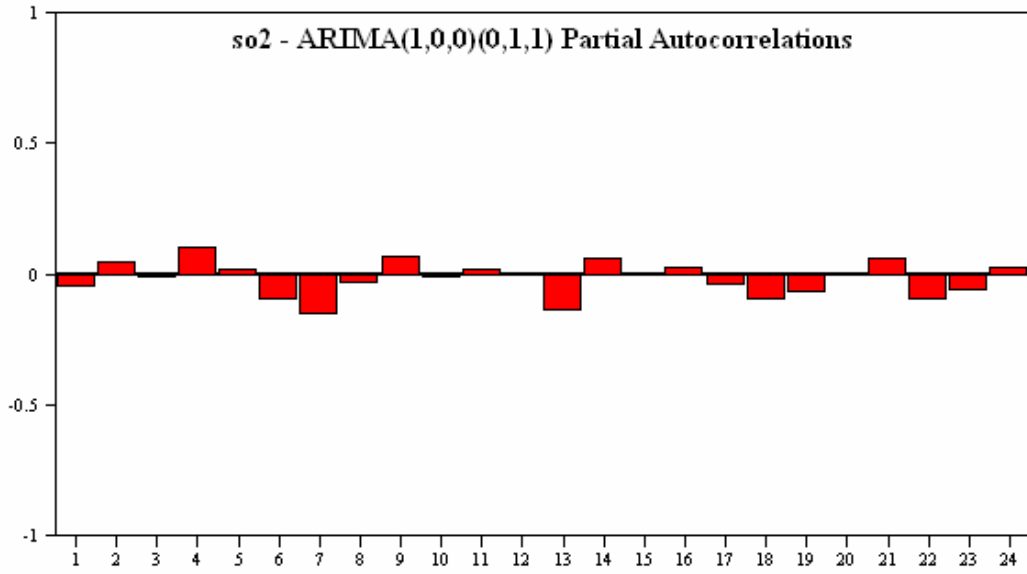
Şekil 5.6, Şekil 5.7 ve Şekil 5.8 kestirim hatasının rassallığını grafiksel olarak ta açıklamaktadır.



Şekil 5.6. SO₂ verileri kestirim hataları



Şekil 5.7. SO₂ verileri kestirim hataları otokorelasyonları



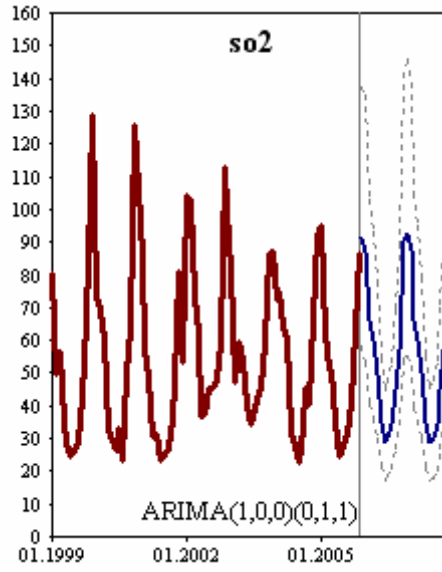
Şekil 5.8. SO₂ verileri kestirim hataları kısmi otokorelasyonları

5.3.3. Kestirim

Modelin geçerliliği sınıandıktan sonra geçerli olduğu kabul edilen ARIMA (1 0 0) (0 1 1) modeli kestirim için kullanılmıştır.

Model ile kestirilen ilk değer 2005 Yılı'nın 12. ayına aittir. Üst kontrol limiti 137 ve alt kontrol limiti 58 olmak üzere, bu tarihteki SO₂ verisi kestirim değeri 91'dir. Bu sonuçlar baz alındığında SO₂ verisinin gerçek değerinin, % 95 güvenilrlikle üst ve alt kontrol limitleri içerisinde gerçekleşeceği söylenebilir.

Şekil 5.9'da ARIMA (1 0 0) (0 1 1) modeli ile elde edilen 13 aylık kestirim değerleri grafiği yer almıştır.



Şekil 5.9. SO₂ verileri kestirimi

5.4. YSA ile Modelleme ve SO₂ Verilerinin Kestirimi

YSA ile modelleme için kurulacak ağ yapısının belirlenmesi ve veri ön işleme yapılması gereklidir.

5.4.1. Kullanılan seri değerlerinin normalleştirilmesi

Veri normalleştirme YSA'nın başarısında etkilidir. Kurulan YSA'nın yapısına uygun olacak şekilde verilerin normalleştirilmesi gereklidir. Ara katman ve çıktı katmanında kullanılan aktivasyon fonksiyonları, veri normalleştirmenin [0 1] ya da [-1 1] aralıklarından hangisine göre yapılacağını belirler.

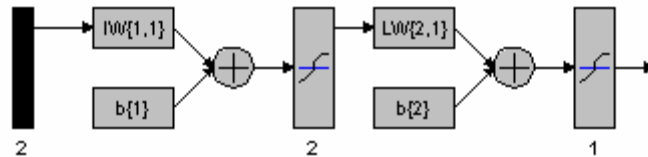
Bu çalışmada yer alan veriler doğrusal olmayan özelliktedir. Bu bakımdan veriler [-1 1] aralığına göre normalleştirilmiştir. [0 1] aralığına normalleştirme işlemi ise zaman (t) değerleri için, Lachtermacher ve Fuller (29)'in önerdiği basit normalizasyon ile gerçekleştirilmiştir. [-1 1] aralığına normalleştirme ise,

$$x_{\text{norm}} = (2 * (x_0 - x_{\min}) / (x_{\max} - x_{\min})) - 1 \text{ eşitliğine göre yapılmıştır.}$$

5.4.2. Çalışmada kullanılan ağ yapısı

Zaman serileri kestirimi için kurulacak YSA'nın olası ağ yapıları üzerinden uygulamalar yapılarak en iyi sonucu veren YSA yapısı seçilmiştir. Girdi nöronu ve gizli nöron sayısının belirlenmesinde literatürdeki uygulamalardan da gözlendiği üzere deneme-yanılma yöntemi ile uygun değerler belirlenmeye çalışılır. Çıktı katmanını nöron sayısı, eğer iteratif bir kestirim yapılacaksa 1, aynı anda birden fazla dönemin kestirimi yapılacaksa kestirilecek dönem sayısı kadar olabilir. Box-Jenkins yöntemi de iteratif kestirimde bulunmaktadır. Her iki tekniğin karşılaştırılması bakımından, YSA uygulamasında da iteratif kestirim yöntemi seçilerek, çıktı nöronu sayısı 1 olarak alınmıştır.

Girdi nöronu zaman serisi olarak ele alınan veriler aylık SO₂ verileridir. Bir çok çalışmada girdi nöronu sayısının, ARIMA(p,d,q) modellerindeki p veya q derecelerine eşit ya da yakınında bir değer olabileceği belirtilmektedir. Bu çalışmada girdi nöronu sayısı 1 olarak belirlenmiştir. Gizli nöron sayısının belirlenmesinde herhangi bir kesin kural olmamasından dolayı seri üzerinde ön deneme çalışmaları yapılmıştır. Yapılan ön deneme çalışmalarında gizli nöron sayısının, girdi nöronu sayısına eşit veya bir eksiği ya da bir veya iki fazlası denenmiştir. Her bir durumda ağın ürettiği sonuçlar kaydedilmiştir. Ağın performansında en küçük değeri veren gizli nöron sayısı ilk katmanda 2, ikinci katmanda 1 olarak alınmıştır. YSA modeli yapısı; 1 girdi nöronu, iki katman, ilk katmanda 2, ikinci katmanda 1 gizli nöron, nodlarda sigmoid aktivasyon fonksiyonu ve Levenberg-Marquardt (trainlm) öğrenme algoritması ile oluşmuştur. Şekil 5.10, kullanılan ağa ilişkin genel YSA yapısını göstermektedir.



Şekil 5.10. SO₂ verilerinin kestirimi için kurulan YSA yapısı

Şekil 5.1'de gösterilen ağıın ürettiği çıktının matematiksel ifadesi Eş. 5.2'de verilmiştir.

$$Y_t = \theta_0 + \sum_{j=1}^m v_j \left(\int \sum_{i=1}^k w_{ij} Y_{t-1} + \theta_{0j} \right) + e_i \quad [5.2]$$

Eş. 5.2'de, Y_t , t dönemindeki seri değerini, $\{Y_{t-i}, i=1,2,...,k\}$ k dönem geçmiş gözlem değerini, $\{w_{ij}, i=1,2,...,k; j=1,2,...,m\}$ girdi katman nöronlarından gizli katman nöronlarına olan bağlantıların ağırlıklarını, θ_0 ve θ_{0j} eşik (bias) terimlerinin bağlantılarına ait ağırlıkları, $\{v_j, j=1,2,...,m\}$ gizli katman nöronlarından çıktı katmanı nöronuna olan bağlantıların ağırlıklarını, f kullanılan aktivasyon fonksiyonunu ve e_i 'de hata terimini ifade etmektedir.

5.4.3. Öğrenme algoritması

YSA'nın performansında büyük etkiye sahip faktörlerden biri de öğrenme (eğitim) algoritmasıdır. Eğitim algoritması ağ ağırlıklarının hangi kurala göre düzenleneceğini belirleyen algoritmadır. Bu çalışmada GYA çalıştırılmıştır.

MATLAB programı bir çok GYA'nı içermektedir (35). Hangi öğrenme yönteminin kullanılması gerektiğini belirlemek için Genelleştirilmiş Delta (traingdm), Levenberg-Marquardt (trainlm) ve Conjugate Gradient (traincgf) öğrenme yöntemlerinin karşılaştırıldığı ön deneme çalışmalarında, öğrenme yöntemi olarak Levenberg-Marquardt (trainlm) algoritmasının kullanıldığı ağda daha iyi sonuçlar elde edildiği görülmüştür. Bu sebeple GYA olarak, Levenberg-Marquardt (trainlm) öğrenme yöntemi seçilmiştir.

5.4.4. Eğitim ve test kümesi

Kurulan YSA'nın eğitimi için kullanılan veriler eğitim kümesi olarak adlandırılmaktadır. Test kümesi ise eğitim kümesinin dışında, ağın eğitim sırasında görmediği verilerdir. Test kümesinin amacı, ağın öğrenmesinin yeterli ya da doğru olup olmadığının sınanmasıdır. Literatürde eğitim ve test kümesinin büyüklüğünün ne olması gerektiği çoğunlukla % 90, % 10 veya % 80, % 20 ya da % 70, % 30 oranlarına göre belirlenmektedir (4).

Bazı çalışmalarda da veri kümesi eğitim, doğrulama ve test kümesi olmak üzere üç parçaya bölünmüştür. Eğitim kümesi ağın öğrenmesinde kullanılırken, doğrulama kümesi en iyi ağ modelinin seçilmesinde ve test kümesi de bulunan sonuçların genelleştirilmesinde kullanılmaktadır.

Veri kümesinin, eğitim-test olarak ayırmada % 84, % 16 oranı uygulanmıştır. Bu oranın seçilme nedeni test için ayrılan 13 adet aylık verinin 2005 yılını temsil edecek sayıda olmasını sağlamak, mevsimsel etkileri kapsayacak periyotları içermesini sağlamak ve ayrıca 2004 yılının bir bölümünü de kapsamasını sağlamaktır. Dolayısıyla eğitim için ilk 70 adet veri, test için de son 13 adet veri ayrılmıştır.

5.4.5. Çalışmada kullanılan performans ölçütleri

SO₂ zaman serilerinin kestiriminde YSA modellerinin Box-Jenkins modellerinden daha iyi sonuç verip vermediğinin karşılaştırmaları, YSA'da eğitim ve test kümesi, Box-Jenkins tekniğinde 71 adet kestirim değeri üzerinden yapılmıştır. Bu çalışmada: ortalama hata (OH), ortalama mutlak hata (OMH), hata kareleri toplamı (HKT), hata kareleri ortalaması (HKO), ortalama yüzde hata (OYH), ortalama mutlak yüzde hata (OMYH) performans ölçütleri hesaplanmıştır.

5.4.6. Kestirim tekniklerinin performans ölçütüne etkisi

Kullanılan kestirim tekniklerinin birbirlerinden farklı olup olmadığını anlamak için uyum iyiliği performans sonuçları Tablo 5.2’de yer almıştır.

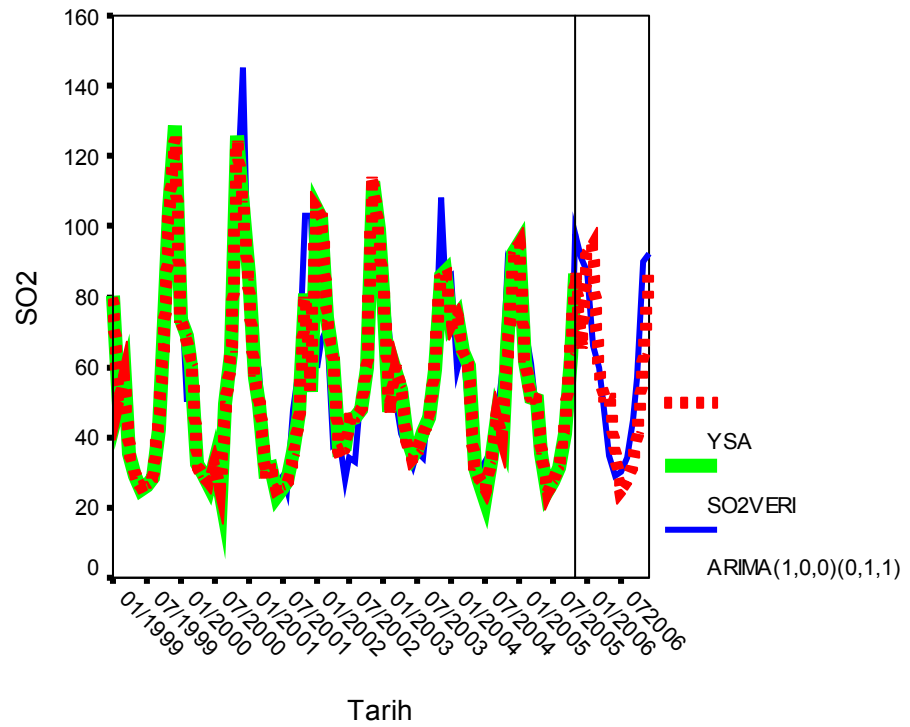
Tablo 5.2 Uyum iyiliği performans sonuçları

Tanımlayıcı İstatistikler	SO ₂ VERİ	YSA			ARIMA (1 0 0) (0 1 1)
		EĞİTİM	TEST	EĞİTİM+TEST	
Gözlem Sayısı	83	70	13	83	71
Min.	22	22	24	24	25
Max.	129	129	95	125	145
Ortanca	49	49	51	50	52
OH (Ortalama Hata)		-0.02	0.005	0.016	0.876
OMH (Ortalama Mutlak Hata)		0.67	0.005	0.56	10.4
HKT (Hata Kareler Toplamı)		52.3	0.0043	52.3	14744.7
HKO (Hata Kareler Ortalaması)		0.75	0.00033	0.63	207.7
OYH (Ortalama % Hata)		-0.31	0.006	-0.240	-4.0
OMYH (Ortalama Mutlak % Hata)		1.5	0.6	1.3	18.0

Tablo 5.2’de 0,05 anlamlılık düzeyinde kullanılan ARIMA ve YSA yöntemi uygulamasında en düşük OMYH değerini YSA uygulamasının verdiği görülmektedir. Ayrıca sadece test verileri üzerinden eğitim sonrası elde edilen kestirim değerlerinin, sıfıra yakın bir hata ile elde edilmiş olması da önemli bir başarıdır. Benzer şekilde YSA’na ait diğer tanımlayıcı istatistikler de ARIMA (1 0 0) (0 1 1) modeline göre daha başarılıdır. Bu da, 1 girdi nöronu, iki katman, ilk katmanda 2, ikinci katmanda 1 gizli nöron, nodlarda sigmoid aktivasyon fonksiyonu ve Levenberg-Marquardt (trainlm) öğrenme algoritması ile oluşan YSA modelinin SO₂ gibi meteorolojik verilerin kestiriminde kullanılabilir olduğu düşüncesini kuvvetlendirmektedir. Sonucumuzu destekler nitelikte, Gardner ve Dorling, Boznar et al. (37), YSA’larda aktivasyon fonksiyonu olarak kullanılan Sigmoid fonksiyonunun meteorolojik verilerde en iyi optimizasyonu sağladığını belirtmişlerdir.

5.4.7. Kestirim dönemi ve sonuçlarının grafiksel karşılaştırması

Şekil-5.11, YSA ve ARIMA yöntemleri ile elde edilen 13 Aylık SO₂ kestirimleri yer almıştır.



Şekil 5.11. Yöntemlere göre kestirim sonuçları

Şekil-5.2 incelendiğinde, verilerin mevsimsellik gösterdiği ve her iki yöntemden elde edilen kestirimlerde, mevsimselliğin öngörülere yansıdığı anlaşılmaktadır. Nelson ve diğ. (32)'leri YSA'nın, mevsimsel zaman serilerinde, mevsimselliği öğrenmeye uygun olmadığını, daha doğru kestirim yapmak için önceden mevsimselliğin giderilmesinin yararlı olacağını söylemektedir. Ancak, bu çalışmada mevsimselliğin YSA Modeli ile öğrenilmiş olduğu anlaşılmaktadır. Bu konuyu destekler nitelikte olan görüşlerinde, Sharda ve Patil (28) zaman serisinin mevsimselliğinin YSA'nın performansını etkilemediğini ve YSA'nın mevsimselliği de tam olarak içine alabilmeye yetenekli olduğunu belirtmektedir.

Bu aşamada, SO₂ serisi için belirlenmiş YSA yapısı üzerinden 83 adet verinin eğitim ve test oranları modelin performansı ve mevsimselliği yansıtabildiğini gözlemlemek üzere değiştirilmiştir. Yeni oranlarda eğitim verisi olarak, bitimi yıl ortasına raslayan ilk 66 veri alınmış ve geriye kalan 17 adet veri test amaçlı kullanılmıştır.

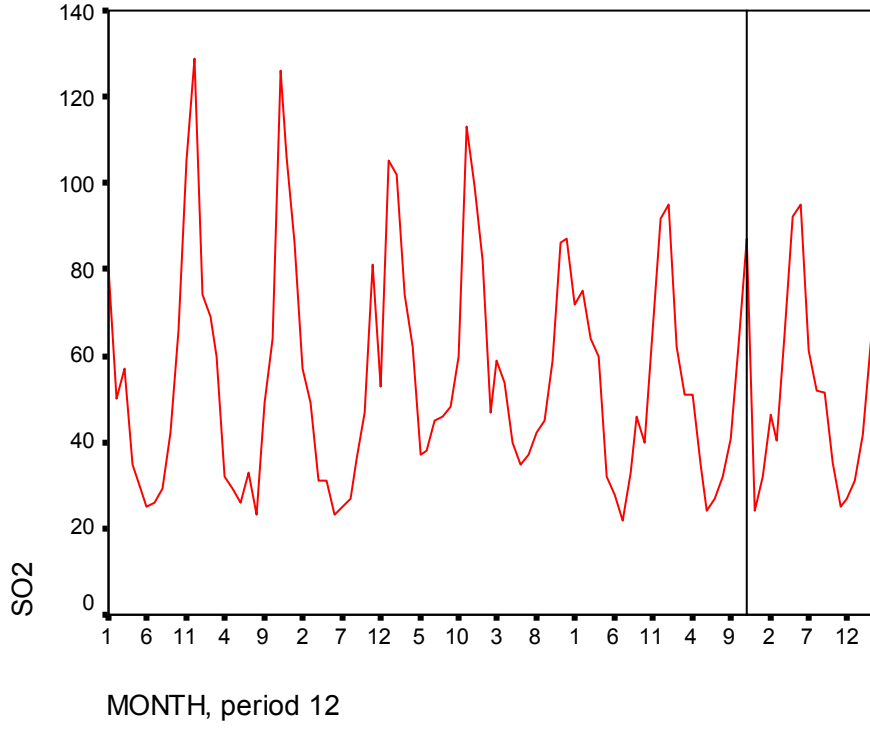
Tablo 5.3, SO₂ serisi için belirlenmiş YSA modelinin, 70-13 ve 66-17 eğitim ve test oranlarına göre kestirim performanslarını göstermektedir.

Tablo 5.3. Eğitim-test oranlarına göre uyum iyiliği sonuçları

Tanımlayıcı İstatistikler	SO ₂ VERİ	YSA		
		EĞİTİM	TEST	EĞİTİM+TEST
Gözlem Sayısı	83	70	13	83
OMYH (Ortalama Mutlak % Hata)		1.5	0.6	1.3
HKO (Hata Kareler Ortalaması)		0.75	0.00033	52.3
Gözlem Sayısı	83	66	17	83
OMYH (Ortalama Mutlak % Hata)		1.4	1.5	1.4
HKO (Hata Kareler Ortalaması)		0.63	0.44	0.59

Test kümesinin amacı, ağın öğrenmesinin yeterli ya da doğru olup olmadığının sınanmasıdır. Tablo 5.3 'te yer alan test verilerinin OMYH ve HKO sonuçları, 13 adet test verisinin, 17 adet test verisine göre daha iyi performans sergilediğini göstermiştir. 13 adet test verisi 2005 yılını kapsayacak şekilde seçilmiştir ve bütünün özelliklerine sahip kümeleri temsil edecek niteliktedir. Bu durumun OMYH ve HKO sonuçlarının daha düşük çıkmasında etkisi olduğu söylenebilir.

SO₂ serisi için belirlenmiş YSA yapısının kestirimlerde mevsimselliği içine alıp alamayacağını anlamak üzere, serinin son 17 verisi yıl ortasından bölünerek test verisi olarak alınmıştır. Şekil 5.3'te yer alan kestirim grafiğinde mevsimsellik gözlenmiştir.



Şekil 5.12. Eğitim-test oranları değiştirildiğinde elde edilen kestirim sonuçları

6. TARTIŞMA VE SONUÇ

Geleceğe ilişkin karar problemlerini zorlaştıran belirsizliktir. Karar vericiler için belirsizliğin giderilmesi mümkün olmasa da azaltılması sağlanabilir. Bu da iyi yapılmış kestirimlerle gerçekleştirilebilir. İyi bir kestirim, minimum hatayı veren tutarlı sonuçlar üreten kestirimdir. Kestirim uğraşısı içerisinde olan uzmanlar tek kestirim tekniğine bağlı kalmak yerine, kestirim modellerini destekleyecek farklı kestirim yöntemlerini de denemelidirler.

Doğrusal zaman serileri kestiriminde başarılı bir yöntem olduğu bilinen Box-Jenkins modelleri, aynı başarıyı doğrusal olmayan zaman serilerinde gösterememektedir. Öte yandan gerçek hayata ilişkin zaman serileri çoğu zaman doğrusal değildir. Bu nedenle, zaman serileri ile kestirimde Box-Jenkins modellerinden daha iyi sonuç verebilecek yeni yöntemler arayışı süregelmektedir. Alternatif yöntemlerden biri de YSA yöntemidir.

YSA, sınıflandırma, kümeleme, örüntü tanıma vb. bir çok alanda kullanılan bir araçtır. Başarı ile kullanıldığı alanlardan biri de geleceği kestirim problemleridir. 1980'li yıllardan beri, bir kestirim aracı olarak kullanılmaktadır. Zaman serileri kestiriminde başarılı sonuçlar verip vermediğini araştırmak için bir çok çalışma yapılmıştır. Bu çalışmaların bir kısmı YSA tekniklerinin, geleneksel yöntemlerden daha iyi sonuç verdiğini söylerken bir kısmı da herhangi bir farklılık olmadığını ileri sürmektedir.

Bu çalışmada, YSA'nın bir kestirim aracı olarak kestirim uygulamasında geniş kabul gören Box-Jenkins modellerinden farklılığının olup olmadığı araştırılarak SO₂ zaman serisi verileri ile bir uygulama gerçekleştirilmiştir. Kestirim modeli ile çalışan uzmanların, YSA'da geliştirilen her yeni model için karşılaştırma aracı olması bakımından diğer teknikler ile de model geliştirmeleri faydalıdır.

Çalışmada ele alınan SO_2 ölçümüne dayalı zaman serilerinde YSA, karşılaştırma aracı olarak kullanılan Box-Jenkins modellerinden daha iyi sonuç vermektedir.

Gerçek zamanlı veri örneği olarak ele alınan SO_2 verilerinde mevsimsellik gözlenmiştir. YSA çalışmasının başında mevsimselliğin sözkonusu olduğu verilerde YSA'nın optimizasyonunda yaşanabilecek güçlüklerle ilişkin literatürde yer alan çelişkili yorumlar ortadan kaldırılarak, YSA'ların mevsimselliği öğrenmede başarılı olabildiği gözlenmiştir.

Örneğe yönelik olarak YSA'nın zaman serileri ile geleceği kestirimde, doğrusal olmayan zaman serilerinde Box-Jenkins modellerinden daha başarılı olduğu söylenebilir.

KAYNAKLAR

1. Kayım, H., İstatistiksel ön kestirim yöntemleri, H.Ü. İ.L.B.F. Yayın No:11, (Ankara), 1-10, 1986.
2. Kutay, F., Zaman serilerinde kestirim teknikleri ve Box-Jenkins modelleri, G.Ü.,(Ankara), 1-99, 1989.
3. Montgomery, D.C., Johnson, L.A. ve Gardiner, J.S., Forecasting and time series analysis, Mc-Graw Hill, (New York), 5-290, 1990.
4. Zhang, G., Patuwo, B.E. ve Hu, M.Y., Forecasting with artificial neural networks:The state of the art, International Journal of Forecasting, 14:35-6 2, 1998.
5. Zhang, G., Patuwo, B.E., Hu, M.Y., A simulation study of artificial neural networks for nonlinear time-series forecasting, Computers&Operations Research, 28: 381-396, 2001.
6. Zhang, G., Time series forecasting using a hybrid ARIMA and neural network model, Neurocomputing, 50: 159-175, 2003.
7. Al-Saba, T. ve El-Amin, L, Artificial neural networks as applied to Long-term demand forecasting, Artificial Intelligence in Engineering, 13:189-197, 1999.
8. Box, G.P. ve Jenkins, G.M., Time series analysis forecasting and control, Holden-Day, (San Francisco), 1-170, 1976.
9. Makridakis, S., Wheelwright, S.C., McGee, Forecasting methods and applications, John Wiley&Sons, (New York), 3A16, 1983.
10. Akgül, I., Zaman serilerinin analizi ve ARIMA modelleri, Der Yayınları, (İstanbul), 8-148, 2003.
11. Box, G.P., Pierce, D.A., Distribution of residual autocorrelations in autoregressive-integrated moving average time series models, Journal of American Statistical Association, 65332:1509-1526, 1970.
12. Ljung, G.M., Box, G.P., On a measure of lack of fit in time series model, Biometrika, 65:297-303, 1978.
13. Haykin, S., Neural networks: A comprehensive foundation, Macmillan College, (New York), 1-23 5, 1994.
14. Pandya, A. ve Macy, R., Pattern recognition with neural networks in C++, CRCPres, (USA), 3-50, 1996.
15. Anderson, D. ve Mc Neill, G., Artificial neural networks, Technology, Kaman Sciences Corporation, (New York), 2-38, 1992.
16. Kubat, M., Bratko, I. ve Michalski, R.S., A Review of machine learning methods, Machine Learning and Data Mining: Methods and Applications, UK, John Wiley & Sons, 3-70., 1997.
17. Öztemel, E., Yapay sinir ağları, Papatya Yayıncılık, (İstanbul), 13-114, 2003.

18. Efe, M.Ö. Yapay sinir ağları ve uygulamaları, Boğaziçi Üniversitesi Yayınevi, (İstanbul), 1-25, 2000.
19. Kaastra, I. ve Boyd, M., Designing A neural network for forecasting financial and econometric time series, *neurocomputing*, 10:215-236, 1996.
20. Tang, Z., Almeida, C. ve Fishwick, P.A., Time series forecasting using neural networks vs Box-Jenkins Methodology, *Simulation*, 57:303-310, 1991.
21. Hebb, D.O., The organization of behavior: a neuropsychological theory, John Wiley&Sons, (New York), 1949
22. Kartalopoulos, S.V., Understanding neural network and fuzzy logic, IEEE Press., (New York), 1-82, 1996.
23. Refenes, A.N., Neural networks in the capital markets, John Wiley, (Chichester), 23-46, 1995.
24. Hu, M.J.C., Application of the adaline system to weather forecasting, Master Thesis, Technical Report, 6775-1, Stanford Electronic Laboratories, (Stanford, CA), 1964.
25. Rumelhart, D.E., Hinton ve G.E., Williams, R.J., Learning internal representation by Back-propagating errors, The PDP Research Group, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Rumelhart, D.E. ve McClelland, J.L., MIT Press, MA, 1986.
26. Sharda, R. ve Patil, R.B., Neural networks as forecasting experts: An Empirical Test, *Proceedings of the International Joint Conference on Neural Networks*, (Washington D.C.), 2:491-494, 1990.
27. Tang, Z. ve Fishwick, P.A., Feedforward neural nets as models for time series forecasting, *ORSA Journal on Computing*, 5:374-385, 1993.
28. Sharda, R. ve Patil, R.B., Connectionist approach to time series prediction: An Empirical Test, *Journal of Intelligent Manufacturing*, 3:317-323, 1992.
29. Lachtermacher, G., Fuller, J.D., Backpropagation in time series forecasting, *Journal of Forecasting*, 14:381-393, 1995.
30. Zhang, X., Time series analysis and prediction by neural networks, *Optimization Methods and Software*, 4:151-170, 1994.
31. Duliba, K.A., Contrasting neural nets with regression in predicting performance in the transportation industry, *Proceedings of the Annual IEEE International Conference on System Sciences*, 25:163-170, 1991.
32. Nelson, M., Hill, T., Remus, B. ve O'Connor, M., Can neural networks be applied to time series forecasting and learn seasonal patterns: An Empirical Investigation, *Proceedings of the Twenty Seventh Annual Hawaii International Conference on System Sciences*, 649-655, 1994.
33. Gonzalez, S., Neural networks for macroeconomic forecasting: A Complementary Approach to Linear Regression Models, Working Paper 2000-07, Department of Finance, (Canada), 26-33, 2000.

34. SPSS-Decision Time for Windows, Release 1.1, SPSS Inc.
35. MATLAB Neural Network Toolbox, Version 7.0, Math Works Inc.
36. Bert K.,Stan G., Neural Networks, Best practice in Europe, Proceedings of stiching Neural Network Conference., 183-187., 1997.
37. M.W. Gardner ve S.R. Dorling, Neural Network Modelling and Prediction Hourly in NO_x and NO₂ concentrations in urban air in London, Athmospheric Environment 1997.
38. Selami Serhatlıoğlu, Fırat Hardalaç, ve İnan Güler 167 ‘Journal of Medical Systems’.
39. Samuel ,Eldar ve diğ. World Journal of Surgery 2002.
40. S.Noyan OğulataYöneylem Araştırması Kongresi 2004.
41. J.G. Taylor, The promise of neural networks., Springer-Verlag., 1-131., 2005
42. Elena Montanes, Jose R. Quevedo, Maria M.Prieto ve Cesar O Menendez, Forecasting time series combining machine learning and Box-Jenkins time series., Artificial Intelligence Center, (Viesques)., E-33271, 2002.
43. Hanh H. Nguyen ve Christine W. Chan, Multiple neural networks for a long time series forecast., Neural Comp. and Appl.13:90-98., 2004.
44. T.Taskaya-Temizel ve M.C.Casey, Configuration of neural networks for the analysis of seasonal time series., Fingrid Project, (UK)., 2005.