



**T.C.  
GAZİ ÜNİVERSİTESİ  
SOSYAL BİLİMLER ENSTİTÜSÜ**

**DOKTORA  
TEZİ**

**BIST 100 GETİRİ ENDEKSİNİN  
FARKLI YÖNTEMLERLE ÖNGÖRÜLEREK  
SONUÇLARININ KARŞILAŞTIRILMASI**

**ÖZLER ÖZGÜR**

**EKONOMETRİ ANABİLİM DALI**

**EKİM 2016**



**BIST 100 GETİRİ ENDEKSİNİN FARKLI YÖNTEMLERLE ÖNGÖRÜLEREK  
SONUÇLARININ KARŞILAŞTIRILMASI**

**Özler ÖZGÜR**

**DOKTORA TEZİ  
EKONOMETRİ ANABİLİM DALI**

**GAZİ ÜNİVERSİTESİ  
SOSYAL BİLİMLER ENSTİTÜSÜ**

**EKİM 2016**

Özler ÖZGÜR tarafından hazırlanan “BIST 100 Getiri Endeksinin Farklı Yöntemlerle Öngörülerek Sonuçlarının Karşılaştırılması” adlı tez çalışması aşağıdaki jüri tarafından OY BİRLİĞİ ile Gazi Üniversitesi Ekonometri Anabilim Dalında DOKTORA TEZİ olarak kabul edilmiştir.

**Danışman:** Prof. Dr. Aydın ÜNSAL

Ekonometri, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum

.....

**Başkan:** Prof. Dr. Hasan ŞAHİN

İktisat, Ankara Üniversitesi

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum

.....

**Üye:** Prof. Dr. Hamza GAMGAM

İstatistik, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum

.....

**Üye:** Prof. Dr. Yılmaz AKDİ

İstatistik, Ankara Üniversitesi

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum

.....

**Üye:** Doç. Dr. Şenol ALTAN

Ekonometri, Gazi Üniversitesi

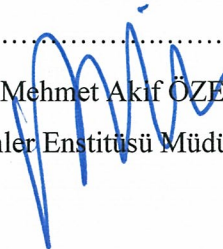
Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum

.....

Tez Savunma Tarihi: 14/10/2016

Jüri tarafından kabul edilen bu tezin Doktora Tezi olması için gerekli şartları yerine getirdiğini onaylıyorum.

.....  
Prof. Dr. Mehmet Akif ÖZER  
Sosyal Bilimler Enstitüsü Müdürü V.

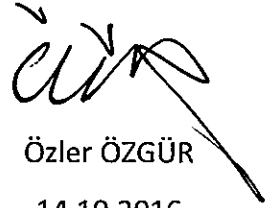


## ETİK BEYAN

Gazi Üniversitesi Sosyal Bilimler Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.



Özler ÖZGÜR

14.10.2016

BIST 100 GETİRİ ENDEKSİNİN FARKLI YÖNTEMLERLE ÖNGÖRÜLEREK  
SONUÇLARININ KARŞILAŞTIRILMASI  
(Doktora Tezi)

Özler ÖZGÜR

GAZİ ÜNİVERSİTESİ  
SOSYAL BİLİMLER ENSTİTÜSÜ  
Ekim 2016

ÖZET

Çalışmada, makine öğrenmesi kavramlarından saklı Markov modelleri ve yapay sinir ağları finansal zaman serilerinin bir sonraki dönem öngörüsü için kullanılmıştır. Çıkan öngörü sonuçları parametrik modeller olan ARIMA ve GARCH'tan çıkan öngörü sonuçları ile karşılaştırılmıştır. Çalışma dört bölümden oluşmaktadır. Birinci bölümde finansal zaman serilerine temel teşkil eden rassal süreçler konusu gözden geçirilmiştir. İkinci bölümde finansal zaman serileri analizinde kullanılan Box-Jenkins yöntemi ile ARCH/GARCH modellerinden bahsedilmiştir. Üçüncü bölümde makine öğrenmesi konusuna kısaca değinilmiş, saklı Markov modelleri ve yapay sinir ağları ile ilgili bilgiler verilmiştir. Son bölüm olan dördüncü bölümde ise ikinci ve üçüncü bölümdeki yöntemler BIST 100 günlük kapanış toplam getiri endeksi verilerine öngörü amaçlı uygulanmış, ortalama mutlak hata performans ölçütü kullanılarak çıkan sonuçlar karşılaştırılmıştır. ARCH/GARCH modellerinin çalışmadaki diğer modellere göre daha iyi öngörü sonuçları ürettiği görülmüştür.

Bilim Kodu : 1106  
Anahtar Kelimeler : Saklı Markov modelleri, yapay sinir ağları, Box-Jenkins yöntemi, ARIMA, ARCH/GARCH, beklenti büyütme algoritması, rassal süreçler  
Sayfa Adedi : 169  
Tez Danışmanı : Prof. Dr. Aydın ÜNSAL

FORECASTING BIST 100 RETURN INDEX WITH DIFFERENT METHODS  
AND COMPARING RESULTS

(Ph. D. Thesis)

Özler ÖZGÜR

GAZİ UNIVERSITY  
INSTITUTE OF SOCIAL SCIENCES

October 2016

ABSTRACT

In this study, some concepts of machine learning namely, hidden Markov models and artificial neural networks were used for financial time series' next period forecast. Forecast results were compared with those from parametric ARIMA and GARCH models. The study consists of four chapters. In the first chapter, the concept of stochastic processes which is principal for financial time series was reviewed. In the second chapter Box-Jenkins methodology and ARCH/GARCH models which are used in financial time series analysis were mentioned. In the third chapter, the concept of machine learning was referred and information about hidden Markov model and artificial neural networks was provided. In the fourth and the last chapter, methods which were discussed in the second and the third chapters were applied to daily closing values of BIST 100 total return index for forecast and model results were compared with mean absolute error performance criteria. It is concluded that, ARCH/GARCH models have produced better forecast results compared to the other models in the study.

ScienceCode : 1106

Key Words : Hidden Markov models, artificial neural networks, Box-Jenkins methodology, ARIMA, ARCH/GARCH, expectation-maximization algorithm, stochastic processes

Page : 169

Supervisor : Prof. Dr. Aydın ÜNSAL

## TEŐEKKÜR

Bu alıŐmayı titiz bir Őekilde ynlendiren tez danıŐman hocam Prof. Dr. Aydın ÜNSAL'a teŐekkr bir bor bilirim. Tez izleme komitesi hocalarım Prof. Dr. Hamza GAMGAM ve Prof. Dr. Yılmaz AKDİ'ye alıŐmaya yaptıkları deęerli katkı ve destekleri iin teŐekkr ederim. Zaman serisi analizi konusu ile beni tanıştıran ve doktora hayatım boyunca deęerli bilgi ve tecrbelerini paylaŐarak yolumu aydınlatan hocam Prof. Dr. Nezir KSE'ye teŐekkrlerimi sunarım.

Her zaman yanımda olan sevgili eŐime, neŐe kaynaklarım oęlum ve kızıma, desteklerini esirgemeyen babam, annem, kayınvalidem ve kayınpederime minnet ve Őkranlarımı sunarım.

alıŐtıęım kurumun yoęun iŐ temposunda doktora alıŐmalarım nedeniyle kısa vadede aksayabilen iŐlerin orta vadede telafi edileceęini bilip bana gvenen ve tekrar eden bu sreci defalarca yaŐadıęımız mevzuat birimi alan uzmanı iŐ arkadaşlarıma teŐekkr ederim.



## İÇİNDEKİLER

	Sayfa
ÖZET.....	iv
ABSTRACT .....	v
TEŞEKKÜR.....	vi
İÇİNDEKİLER .....	vii
ÇİZELGELERİN LİSTESİ .....	xi
ŞEKİLLERİN LİSTESİ.....	xii
SİMGELER VE KISALTMALAR.....	xiv
GİRİŞ.....	1
1 RASSAL SÜREÇLER .....	5
1.1 Bernoulli Süreci ve Bağımsız Rassal Değişkenlerin Toplamı.....	6
1.1.1 Bernoulli süreci.....	6
1.1.2 Başarı sayısı .....	7
1.1.3 Başarı zamanları (Çoklu başarı zamanları) .....	8
1.1.4 Bağımsız RD'lerin toplamı.....	10
1.2 Markov Zincirleri .....	12
1.2.1 Genel Bilgiler .....	12
1.2.2 Belli bir durumu ziyaret .....	15
1.2.3 Durumların sınıflandırılması .....	20
1.3 Limit Davranışları.....	24
1.3.1 R ve F'nin hesaplanması .....	24
1.3.2 Kalıcı durumlar ve limit olasılıkları .....	27
2 FİNANSAL ZAMAN SERİLERİ VE MODELLEMELER.....	31
2.1 Regresyon Analizi .....	31

2.2 Doğrusal Zaman Serisi Modelleri .....	32
2.2.1 ARIMA modelleri .....	34
2.2.1.1 Hareketli ortalama modeli .....	34
2.2.1.2 Otoregresif modeller .....	35
2.2.1.3 Otoregresif hareketli ortalama modelleri .....	36
2.2.1.4 Bütünleşik otoregresif hareketli ortalama modelleri.....	37
2.2.2 Modelleme .....	38
2.3 Oynaklık Modelleri .....	43
2.3.1 ARCH/GARCH modeli yapısı .....	45
2.3.2 ARCH/GARCH modelleme .....	50
2.3.3 ARCH/GARCH modellerinin zayıf olduğu noktalar .....	55
<b>3 MAKİNE ÖĞRENMESİ.....</b>	<b>57</b>
3.1 Saklı Markov Modelleri .....	58
3.1.1 Markov modelleri ve saklı Markov modelleri.....	58
3.1.1.1 Değerlendirme problemi (problem 1).....	62
3.1.1.2 Sır çözme (problem 2) .....	67
3.1.1.3 Öğrenme problemi (problem 3).....	69
3.1.1.3.1 Beklenti büyütme algoritması .....	70
3.1.1.3.2 Baum-Welch algoritması.....	78
3.2 Yapay Sinir Ağları.....	81
3.2.1 Yapay sinir ağları genel bilgiler .....	81
3.2.1.1 Biyolojik sinir ağı.....	82
3.2.1.2 Yapay sinir ağları tarihçe ve tanım .....	85
3.2.1.3 Biyolojik ve yapay sinir ağlarının karşılaştırılması .....	87
3.2.1.4 Modelin katmanları .....	88

	<b>Sayfa</b>
3.2.1.4.1 Giriş katmanı .....	88
3.2.1.4.2 Gizli katman.....	89
3.2.1.4.3 Çıktı katmanı.....	89
3.2.1.5 YSA modelinin karar amaçlı kullanımı .....	90
3.2.1.6 Modelin öğrenmesi .....	91
3.2.2 Bazı YSA modelleri .....	91
3.2.2.1 Algılayıcı.....	92
3.2.2.2 Adaline.....	93
3.2.2.3 Çok katmanlı algılayıcı .....	95
3.2.3 YSA Türleri .....	103
3.2.3.1 Tipine göre YSA türleri.....	103
3.2.3.1.1 İleri ve Geri beslemeli ağlar .....	103
3.2.3.2 Öğrenme yöntemlerine göre YSA türleri.....	103
3.2.3.2.1 Öğretmenli öğrenme .....	103
3.2.3.2.2 Öğretmensiz öğrenme.....	103
3.2.3.2.3 Destekleyici öğrenme.....	104
3.2.3.3 Mimarisine (topolojisine) göre ağlar .....	104
3.2.4 YSA Kullanım Özellikleri .....	104
3.2.4.1 Özellikleri ve avantajları .....	104
3.2.4.1.1 Doğrusal olmayan modelleme yapma yeteneği .....	104
3.2.4.1.2 Paralel çalışma.....	105
3.2.4.1.3 Az varsayım ile kullanılma .....	105
3.2.4.1.4 Kademeli bozulma .....	107
3.2.4.1.5 YSA modelleri ile ilgili dikkat edilmesi gereken hususlar .....	108
3.2.4.2 Dezavantajları.....	108

	<b>Sayfa</b>
3.2.4.2.1 Ağ kurulmasında kuralların net olmaması .....	108
3.2.4.2.2 Kara kutu özelliği .....	109
3.2.4.2.3 Ezberleme.....	109
3.2.4.2.4 Çözüme geç yakınsama .....	109
3.2.4.2.5 Global değere değil yerel değere yakınsayabilme .....	109
3.2.5 YSA ile zaman serileri öngörüsü .....	110
<b>4 MODELLERLE ÖNGÖRÜ .....</b>	<b>111</b>
4.1 Finans ve Oynak Bir Finansal Değişken: Hisse Senedi .....	111
4.2 Veri Özellikleri .....	112
4.3 ARMA modelleri .....	119
4.4 Oynaklık modelleri .....	125
4.5 SMM Yöntemi.....	132
4.6 YSA Modelleri .....	138
<b>5 SONUÇLAR .....</b>	<b>143</b>
<b>KAYNAKLAR .....</b>	<b>147</b>
<b>EKLER .....</b>	<b>153</b>
EK-1. Uygun model kalıntıları .....	154
EK-2. Uygun model kalıntıları histogramı.....	155
EK-3. Uygun model kalıntıları normal dağılım QQ çizgesi.....	156
EK-4. Uygulamada kullanılan hazır MATLAB kodları.....	157
<b>ÖZGEÇMİŞ .....</b>	<b>167</b>

## ÇİZELGELERİN LİSTESİ

<b>Çizelge</b>	<b>Sayfa</b>
Çizelge 1.1. R ve F'nin hesaplanması.....	24
Çizelge 3.1. YSA ile endüstride yapılan öngörü çalışmaları.....	107
Çizelge 4.1. Artış ve düşüslere göre dağılımlar .....	113
Çizelge 4.2. ADF testi sonuçları .....	115
Çizelge 4.3. B100gek ile ilgili genel bilgiler .....	118
Çizelge 4.4. Getiri verisi için karşılaştırılan modellerin AIC ve BIC değerleri .....	124
Çizelge 4.5. Model parametre tahmin bilgileri .....	124
Çizelge 4.6. ARCH etkisi ile ilgili bilgiler.....	127
Çizelge 4.7. Getiri verisi için karşılaştırılan modellerin AIC ve BIC değerleri .....	128
Çizelge 4.8. Ortalama denklemi parametre tahmin bilgileri.....	128
Çizelge 4.9. Oynaklık denklemi parametre tahmin bilgileri.....	128
Çizelge 4.10. Standartlaştırılmış kalıntılar için LB testi bilgileri .....	132
Çizelge 4.11. Sert değişim %3 tanımı ile kesiklileştirme tablosu .....	134
Çizelge 4.12. Geçiş matrisi .....	135
Çizelge 4.13. Başlangıç olasılıkları vektörü .....	135
Çizelge 4.14. Durum matrisi.....	135
Çizelge 4.15. Gözlem matrisi.....	136
Çizelge 4.16. Model seçimi bilgi kriterleri .....	136
Çizelge 4.17. Tahmin edilen durum matrisi .....	137
Çizelge 4.18. Tahmin edilen gözlem matrisi .....	137
Çizelge 4.19. Modellerin performans karşılaştırmaları.....	141

## ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1. Belli bir zaman serisinin 5 farklı $\omega$ için gerçekleştirilmesi .....	33
Şekil 2.2. Box-Jenkins yönteminin adımları .....	39
Şekil 3.1. Markov modelleri .....	59
Şekil 3.2. Saklı Markov modelleri .....	60
Şekil 3.3. Trellis şeması (Alpaydın, 2011) .....	63
Şekil 3.4. İleri algoritması .....	65
Şekil 3.5. İdeal üretim olasılığı .....	67
Şekil 3.6. Tam veri, tam olmayan veri transformasyonu .....	72
Şekil 3.7. BBA'nın çalışması .....	75
Şekil 3.8. Sinir sistemi .....	83
Şekil 3.9. Sinir hücresi .....	84
Şekil 3.10. Nöronun yapısı .....	88
Şekil 3.11. Modelin katmanları .....	90
Şekil 3.12. Sık kullanılan aktivasyon fonksiyonları .....	91
Şekil 3.13. Algılayıcı - nöronun yapısı .....	92
Şekil 3.14. Adaline - nöronun yapısı .....	94
Şekil 3.15. ÇKA - nöronun yapısı .....	96
Şekil 3.16. Tam bağlı, ileri yönlü, tek gizli katmanlı YSA .....	97
Şekil 3.17. Girdilerin işlenmesi ile veri akışı yönü .....	97
Şekil 3.18. Geri yayılım algoritması .....	98
Şekil 3.19. Öğrenme yöntemleri .....	104
Şekil 4.1. B100gek .....	113
Şekil 4.2. LB100gek .....	114

<b>Şekil</b>	<b>Sayfa</b>
Şekil 4.3. GB100gek verisi .....	116
Şekil 4.4. GB100gek histogram ve normal dağılım .....	117
Şekil 4.5. GB100gek QQ-Çizgesi .....	118
Şekil 4.6. MGB100gek .....	121
Şekil 4.7. MB100gek korelogram .....	121
Şekil 4.8. MB100gek kısmi korelogram .....	122
Şekil 4.9. MGB100gek korelogramı .....	123
Şekil 4.10. MGB100gek kısmi korelogramı .....	123
Şekil 4.11. ARMA öngörüsü .....	125
Şekil 4.12. MGB100gekhk korelogramı .....	126
Şekil 4.13. MGB100gekhk kısmi korelogramı .....	127
Şekil 4.14. Koşullu varyans .....	129
Şekil 4.15. GARCH standartlaştırılmış kalıntı kareler korelogram .....	130
Şekil 4.16. GARCH standartlaştırılmış kalıntı kareler kısmi korelogram .....	130
Şekil 4.17. GARCH standartlaştırılmış model kalıntıları histogram .....	131
Şekil 4.18. GARCH model kalıntıları QQ çizgesi .....	131
Şekil 4.19. ARCH/GARCH öngörüsü .....	132
Şekil 4.20. SMM öngörüsü .....	138
Şekil 4.21. YSA öngörüsü .....	141

## SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış bazı simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

### Simgeler

### Açıklama

$\gamma_X(h)$  Otokorelasyon fonksiyonu (okorf)

$\rho_X(h)$  Kısmi otokorelasyon fonksiyonu (kokorf)

$F(m)$  Karakteristik denklem

$B$  Gerileme operatörü

$\nabla^d$  Fark operatörü (d. fark)

$\tau$  Dickey-Fuller t test istatistiği

$\mu_t$  Koşullu beklenen değer

$\sigma_t^2$  Koşullu varyans

$SSR_0$  Toplam kareler toplamı

$SSR_1$  Kalıntı kareler toplamı

$L_T(\cdot)$  Log olabilirlik fonksiyonu

$I(\theta)$  Bilgi matrisi

$\pi$  Başlangıç olasılıkları vektörü

$A$  Durum matrisi

$B$  Gözlem matrisi

$net_j, o_j, t_j$  Fonksiyon çıktısı, çıktı, hedef

$b$  Eşik değeri

$\omega_{ij}$  Sinaptik ağırlık

$e_j, E$  Hata, Toplam hata

### Kısaltmalar

### Açıklama

**ARIMA** Bütünleşik otoregresif hareketli ortalamalar

**BBA** Beklenti büyütme algoritması

**ÇKA** Çok katmanlı algılayıcı



<b>Kısaltmalar</b>	<b>Açıklama</b>
<b>EKK</b>	En küçük kareler (yöntemi)
<b>FZS</b>	Finansal zaman serisi
<b>FFBP</b>	İleri beslemeli – geri yayımlı (ağlar)
<b>GARCH</b>	Genelleştirilmiş otoregresif koşullu değişen varyans
<b>IID</b>	Bağımsız ve aynı dağılımlı
<b>LB testi</b>	Ljung-Box testi
<b>MZ</b>	Markov zincirleri
<b>RD</b>	Rassal değişken
<b>SMM</b>	Saklı Markov modelleri
<b>YSA</b>	Yapay sinir ağları
<b>ZSA</b>	Zaman serisi analizi

## GİRİŞ

Rassallıkla çevrili hayatımızda gelecek bilgisi insanođlu için belki de en deđerli bilgidir. Geçmiş ve Őu anki bilgilere erişim belli bir kaynak karşılığında mümkün iken, gelecek bilgisine sonsuz kaynakla bile kesin olarak sahip olunamayabilir. İnsanođlu çođu zaman geçmişini ölçer ve saklar çünkü ancak geçmişini kullanarak gelecek hakkında fikir edinebilir.

Gelecekteki deđeri kestirme problemi (öngörme) diđer birçok bilim dalında olduđu gibi, finansal zaman serileri analizinde de önemli bir yer oluşturur. Bunun için zaman serileri modelleri tanımlanır, diđer bir deyişle, verilerin belli bir örüntüye göre hareket ettiđi bilgisi bulunmaya çalışılır.

Veri özelliklerinin diđer makro ekonomik verilerinkinden farklı olması çođu finansal olgunun, klasik ekonometrik modellerden farklı yapıdaki modeller ile ele alınmasını gerektirmiştir. Finansal verilerin özelliklerinden bazıları şöyle sıralanabilir:

- Finansal zaman serilerinin veri frekansı yüksektir. Günümüzde teknolojinin ilerlemesi ile finansal varlık deđerleri saniyeler içinde deđişmekte ve bunlar kayıt altına alınabilmektedir. Veri frekansının yüksek oluşu veriyi ele almayı zorlaştırırken, veriden normalde elde edilemeyecek örüntülerin ortaya çıkmasına da yardımcı olabilir.
- Finansal verilerdeki deđişimler bilinmeyen çok sayıda faktörden ortaya çıkmış olabilir, bu yüzden yapısal modellerde olduđu gibi bir neden sonuç ilişkisi ile birkaç açıklayıcı deđerşken içeren modeller oluşturmak mümkün olmayabilir.
- Finansal zaman serilerinin incelenmesini ve gelecek dönemlere ilişkin kestirim yapılmasını zorlaştıran diđer bir özelliđi durađan olmayabilmeleridir. Farklı kıymetler, durađanlık ile ilgili farklı özellikler göstermekte hatta farklı pazarlardaki aynı tür verilerin durađanlık özellikleri farklı olabilmektedir.
- Finansal zaman serisi verilerinin diđer bir özelliđi normal dađılmamaları, normal dađılıma göre kalın kuyruk özelliklerine sahip olmasıdır. Kalın kuyruk özelliđi oynaklığın yüksek olduğunun bir göstergesidir. Bu verilerdeki oynaklığın ise kaldıraç, kümelenme gibi modellemeyi zorlaştıran özellikleri bulunmaktadır.
- Finansal zaman serileri asimetrik dalgalanmalar gerçekleştirirler.

Finansal zaman serileri analizine yönelik modellerin bu özellikleri de kapsamı beklenir.

Rassal süreçler gün geçtikçe daha fazla hayatımıza girmektedir. Başka bir açıdan ise, zaten hayatımızın içinde olan rassallığın insanoğlu tarafından taklit edilebilirliği gün geçtikçe artmaktadır.

Rassal süreçler ve olasılık teorisi, kaçınılmaz hataların azaltılmasında, gözlenen sonuçların nedenlerinin varlığını saptamada ve şans değişimindeki temel olasılık yasalarının keşfi için kullanılmaktadır. Rassal süreçler ve olasılık teorisi genellemede ve öngöründe temel araçtır (Çınlar, 1975).

Finansal zaman serileri şimdiye kadar; klasik yöntem olarak da bilinen Box-Jenkins yöntemi (Box ve Jenkins, 1976), oynaklık modelleri (Engle 1982, Bollerslev 1986), birçok farklı modeli barındıran parametrik doğrusal olmayan yaklaşım, sürekli zaman difüzyon modelleri ve özellikle son yıllarda veri güdümlü metotlar ile ele alınmaya başlamışlardır.

Bilgisayarların erişilebilirliğinin kolaylaşması ve bilgisayar işlem gücünün artması bilimsel problemlerin çözümünde veri güdümlü metotların hesap yükü ile ilgili dezavantajlarını azaltmış, bu da veri güdümlü metotlarla yapılan çalışma sayısında görece artış olmasını sağlamıştır. Bu metotlardan bir kısmı finansal zaman serilerinin bir sonraki dönem öngörüsünde kullanılmıştır.

Son zamanlarda saklı Markov yöntemi ile finansal zaman serilerinin ele alınışı ve geleceğe ilişkin öngörüler ile ilgili çalışmalarda belirgin bir artış yaşanmıştır. Oysa saklı Markov modeli ilk olarak bilgisayar bilimi altında ses tanıma problemi ile başarısını kanıtlamış bir modeldir (Rabiner, 1989), biyolojik sıra analizi ve biyoinformatikte ise uzun yıllardır kullanılmaktadır. Saklı Markov modellerinde parametre tahminleri Baum-Welch algoritması ile gerçekleştirilir (Baum, Petrie, Soules ve Weiss, 1970). Baum-Welch algoritmasındaki çözüm yöntemi, beklenti büyütme algoritması (Dempster, Laird ve Rubin, 1977) olarak bilinen ve saklı verilerin olduğu kurulumlarda tahmin yapmayı sağlayan bir algoritmaya dayanmaktadır.

Yapay sinir ađları gemiři 1940'lı yıllara dayanan ve gnmze kadar poplaritesini yitirmemiř bir makine đrenmesi tekniđidir. Yapay sinir ađlarının bu alıřmada kullanılacak olan bir tr ok katmanlı algılayıcıların (KA) hata minimizasyon algoritması ise 1985 yılında Rumelhart, Hinton ve Williams tarafından tanıtılmıřtır. Yapay sinir ađları ile ilgili sayısız teorik ve pratik alıřma yapılmıřtır ve hala yapılmaya devam etmektedir.

Bu alıřmada literatrde makine đrenmesi yntemleri olarak adları geen veri gdml istatistiksel yntemlerden saklı Markov modelleri (SMM) ve yapay sinir ađları (YSA) yntemleri ile zaman serileri analizinde tartıřılmaz en nemli yerlere sahip olan ARIMA ve ARCH/GARCH modelleri incelenmiř ve bu modellerle BIST 100 gnlk getiri endeksi verilerinin bir sonraki dnem ngrs yapılmıřtır. Daha sonra tm modellerden ıkan sonular karřılařtırılmıřtır.

Modelleme alıřması BIST 100 kapanıř getiri endeksi deđerlerinin 02.01.1997-16.10.2015 tarihleri arasındaki 4688 adet verisi ile gerekleřtirilmifitir. Son 200 gnlk veri ngr performansını karřılařtırmak amacıyla test iin kullanılmıřtır. Performans lt olarak bu tip alıřmalarda sıka kullanılan ortalama mutlak hata (MAE) lt kullanılmıřtır.

Tm yntemlerin uygulama adımları MATLAB programı ile gerekleřtirilmifitir. Problem zmleri iin MATLAB ktphanesinden yararlanılmıř, iřlemlerin bazıları gerekleřtirmek iin MATLAB kodlaması yapılmıřtır. Uygulamaların tamamı MATLAB 2012a programı zerinde gerekleřtirilmifitir.



## 1 RASSAL SÜREÇLER

Rassal süreçler, içinde olasılık barındıran süreçleri modellemek için kullanılır. Bu konu olasılık teorisi temelleri üzerine oluşturulmuştur ve zaman fikri ile birlikte ele alınmaktadır. Rassal süreçler bilgilerinden yararlanmayan bilim dalı yok gibidir. Zaman serileri analizinde ise bu konu temel teşkil eder. Öyle ki her zaman serisi aynı zaman da bir rassal süreçtir. Bu bölümde rassal süreçler tüm yönleri ile değil fakat çalışmaya temel teşkil eden yönleri ile incelenmiştir.

Rassal süreç tanımı için iki yaklaşım bulunmaktadır. Bu yaklaşımların birinde rassal süreç, belli bir olasılık uzayında tanımlanmış olan rassal değişkenlerin çoğunlukla zaman ile dizilenmiş topluluğu olarak tanımlanır.

Bu yaklaşımın şu avantajları bulunmaktadır:

- İndeks kümesi keyfi olarak seçilebilir.
- Fonksiyon analizi bilgileri gerekliliği ortadan kalkmaktadır.

Bu yaklaşımın dezavantajları ise şunlardır:

- $X_t(\omega)$ 'nin her bir  $t$  için iyi tanımlanabildiği durumlarda kullanılabilir.
- Örnek patikasının sürekliliği ve monotonluğu doğrudan belirlenemez.

İkinci yaklaşımda genellikle  $T$  reel bir aralık olmak üzere, rassal süreçler fonksiyon değerli bir rassal değişken (RD) olarak tanımlanabilir:  $RD: \Omega \rightarrow (C[0,1], L^2[0,1])$  gibi bir fonksiyon uzayı). Bu yaklaşım, diğer yaklaşımın yukarıda bahsedilen dezavantajlarını ortadan kaldırmaktadır (Dembo, 2008).

İlk yaklaşım temel alındığında,  $t$  indeks kümesinin sayılabilir bir küme olması durumunda bu süreçlere *kesikli zaman rassal süreçler*, sayılamayan bir küme olması halinde ise *sürekli zaman rassal süreçler* denir. Bu yaklaşımda olasılıksal özellikler *sonlu boyutlu dağılımlar* ile ele alınır (Lakatos, Szeidl ve Telek, 2013).

## 1.1 Bernoulli Süreci ve Bağımsız Rassal Değişkenlerin Toplamı

Bernoulli süreci diğer rassal süreçlere temel teşkil eder. Ayrıca kavramsal açıdan temel olasılık teoremleri için gerekli tanımları barındırır.

### 1.1.1 Bernoulli süreci

Denemelerin *başarı* ve *başarısızlık* gibi iki sonuçlu olduğu bir deneyde sonsuz sayıda, birbirinden bağımsız ve özdeş denemelerin yapıldığı bir süreç tanımı Bernoulli sürecine karşılık gelmektedir (Çınlar, 1975). Buna göre;

$$\Omega = \{\omega: \omega = (\omega_1, \omega_2, \dots)\}, \omega_i \text{ başarı ve başarısızlık} \quad (1.1)$$

$$X_i(\omega = \text{başarı}) = 1 \text{ ve } X_i(\omega = \text{başarısızlık}) = 0 \text{ ve } E = \{0, 1\}$$

olmak üzere

$$0 \leq p \leq 1, q = 1 - p \quad (1.2)$$

ve

$$P(\text{başarı}) = p \text{ her bir denemedeki başarı olasılığı} \quad (1.3)$$

$$P(\text{başarısızlık}) = q \text{ ise başarısızlık olasılığı}$$

olarak tanımlanmıştır. Sürecin olasılığı  $P$ , yukarıdaki bilgilerle tam olarak belirlenebilmektedir. Şöyle ki:

$$\omega = \{\omega_1 = \text{başarı}, \omega_2 = \text{başarısızlık}, \omega_3 = \text{başarısızlık}, \omega_4 = \text{başarı}, \omega_5 = \text{başarı}\} \quad (1.4)$$

olayının gerçekleşme olasılığı

$$p \cdot q \cdot q \cdot p \cdot p = p^3 q^2 \quad (1.5)$$

olur. Süreçte her bir  $n$  için beklenen değer

$$E[X_n] = E[X_n^2] = E[X_n^3] = \dots = p \quad (1.6)$$

ve varyans

$$Var[X_n] = E[X_n^2] - E[X_n]^2 = pq \quad (1.7)$$

olarak elde edilir.

### 1.1.2 Başarı sayısı

Bir Bernoulli süreci  $\{X_n; n \in \mathbb{N}\}$  ile gösterilsin. Her bir  $\omega \in \Omega$  için

$$N_n(\omega) = \begin{cases} 0, & n = 0 \\ X_1(\omega) + \dots + X_n(\omega), & n = 1, 2, \dots \end{cases} \quad (1.8)$$

şeklinde tanımlanan  $N_n(\omega)$ , Bernoulli sürecinde  $n$  denemedeki başarı sayısını sayar. Süreç,

$$\{N_n; n \in \mathbb{N}\} \quad (1.9)$$

olarak gösterilebilir. Bu yeni süreç için

$$E = \mathbb{N} = \{0, 1, \dots\} \quad (1.10)$$

olmaktadır. Diğer bir deyişle zaman parametresi de durum uzayı da kesiklidir. Süreç için

$$E[N_n] = E[X_1 + \dots + X_n] = E[X_1] + \dots + E[X_n] = np \quad (1.11)$$

$$Var[X_1 + \dots + X_n] = Var[X_1] + \dots + Var[X_n] = npq$$

'dir ve sürecin dağılımı da her  $n \in \mathbb{N}$  için,



$$P(N_n = k) = C(n, k)p^k q^{n-k}, k = 0, \dots, n \quad (1.12)$$

olur.

$N_{m+n} - N_m$  süreci de aynı özellikleri göstereceğinden her  $n, m \in \mathbb{N}$  için,

$$P(N_{m+n} - N_m = k) = C(n, k)p^k q^{n-k}, k = 0, \dots, n \quad (1.13)$$

$m$ 'den bağımsızdır.

Bernoulli süreci için tanımdan dolayı,

$$\begin{aligned} P(N_{m+n} - N_m = k | N_0, \dots, N_m) &= P(N_{m+n} - N_m = k) \\ &= P(N_n = k) = C(n, k)p^k q^{n-k}, \text{ her } k = 0, \dots, n \end{aligned} \quad (1.14)$$

$m$  değerinden bağımsız olur.

Bağımsız artışlı süreç:

$n_0 = 0 < n_1 < n_2 < \dots < n_j$  tamsayılar olmak üzere  $N_{n_1} - N_{n_0}, N_{n_2} - N_{n_1}, \dots, N_{n_j} - N_{n_{j-1}}$  RD'leri birbirinden bağımsızdır. Bu özelliği sağlayan süreçlere *bağımsız artışlı süreç* (process of independent increments) denir.

$X_n$ 'lerin aynı dağılımlı olduğu bilindiğinde  $N_{m+n} - N_m$  RD'si  $m$  değerinden bağımsız olur. Bu süreçlere *durağan bağımsız artışlı süreç* denilmektedir.

### 1.1.3 Başarı zamanları (Çoklu başarı zamanları)

$\{X_n; n \in \mathbb{N}\}$  ve  $P(\text{başarı}) = p$  olarak tanımlanan bir Bernoulli süreci ve  $\omega \in \Omega$  iken  $X_1(\omega), X_2(\omega), X_3(\omega), \dots$  sürecin gerçekleşmesi olduğunda, bu süreç için  $T_1(\omega), T_2(\omega), T_3(\omega), \dots$  başarıların zamanlarını tutan RD'ler olarak tanımlandıklarında,

$$N_n(\omega) = \begin{cases} 0, & n = 0 \\ X_1(\omega) + \dots + X_n(\omega), & n = 1, 2, \dots \end{cases} \quad (1.15)$$

olmak üzere süreçte;

- $T_k(\omega) \leq n$  ancak ve ancak  $N_n(\omega) \geq k$
- $T_k(\omega) = n$  ancak ve ancak  $N_{n-1}(\omega) = k - 1, X_n(\omega) = 1$

ilişkisi bulunmaktadır. Ayrıca herhangi bir  $k \in \{1, 2, \dots\}$  için

$$P(T_{k+1} = n | T_0, \dots, T_k) = P(T_{k+1} = n | T_k) \quad (1.16)$$

'dır. Bu da  $T_{k+1}$  değerini tahminde  $T_k$  değerinden önceki  $T$  değerlerinin bir önemi olmadığı anlamına gelir. Yani belli bir zamandaki süreçteki değer verildiğinde o zamana göre geçmiş gelecekte bağımsızdır. Bu bilgilerle  $T_k$  RD'sinin olasılık fonksiyonu

$$P(T_{k+1} - T_k = m) = pq^{m-1} \quad (1.17)$$

olacaktır ve

$$E[T_{k+1} - T_k] = 1/p \quad (1.18)$$

ve

$$Var(T_{k+1} - T_k) = q/p^2 \quad (1.19)$$

olacaktır. Yukarıdaki bilgilerden görüldüğü üzere  $T_1, T_2 - T_1, T_3 - T_2, \dots$  bağımsız aynı dağılıma sahip (IID) geometrik RD'lerdir. Bu durumda aşağıdaki eşitlik yazılabilir:

$$P(T_k = t) = C(t - 1, k - 1)p^{k-1}(1 - p)^{t-k}p = C(t - 1, k - 1)p^k(1 - p)^{t-k} \quad (1.20)$$

Dolayısıyla

$$E[T_k] = \frac{k}{p} \quad (1.21)$$

ve

$$Var(T_k) = \frac{kq}{p^2} \quad (1.22)$$

şeklinde elde edilir.

#### 1.1.4 Bağımsız RD'lerin toplamı

$Y_1, Y_2, \dots$  serisinin IID RD'lerden oluştuğu varsayıldığında, bu seri için ve her  $n \in \mathbb{N}$  için

$$Z_n = \begin{cases} 0, & n = 0 \\ Y_1(\omega) + \dots + Y_n(\omega), & n \geq 1 \end{cases} \quad (1.23)$$

gibi bir süreç tanımlandığında, bu süreç için  $Z_{n_1} - Z_{n_0}, Z_{n_2} - Z_{n_1}, \dots, Z_{n_k} - Z_{n_{k-1}}$  değerleri tüm  $0 \leq n_0 < n_1 < n_2 < \dots < n_k$  değerleri için bağımsız ve  $Z_{n+m} - Z_n$  dağılımı da  $n$  değerinden bağımsız olur.

Aynı şekilde eğer  $Z_n$  süreci durağan artımlara sahipse  $n_0 = 0, n_1 = 1, n_2 = 2, \dots, n_k = k$  alındığında  $Z_1 - Z_0, Z_2 - Z_1, \dots$  artımları da IID olur.

$E[Y_n] = a$  ve  $Var(Y_n) = b^2$  olarak verildiğinde,

$$E[Z_n] = na$$

ve (1.24)

$$Var(Z_n) = nb^2$$

olacağı beklenen değer ve varyans özelliklerinden açıktır.

Fakat bu değerlerin elde edilmesi için bile öncelikle beklenen değerlerin " $a$ " olduğunun ve daha genel olarak dağılım fonksiyonunun keşfedilmesi gerekir.  $Z_n$  ele alındığında beklenen değer

tüm denemeler üzerinden ortalama değer olarak tahmin edilebilir. Varyans da  $Var\left(\frac{Z_n}{n}\right) = \frac{b^2}{n}$  olacaktır. Varyans eşitliği  $n$  değeri arttıkça tahminin daha iyi olacağını söylemektedir. Chebyshev eşitsizliği ile  $\varepsilon > 0$  değeri için belli bir limit değerine yaklaşacağı gösterilmektedir:

$$P\left(\left|\frac{Z_n}{n} - a\right| > \varepsilon\right) \leq \frac{b^2}{n\varepsilon^2} \quad (1.25)$$

Zayıf büyük sayılar kanunu (ZBSK),  $n \rightarrow \infty$  alındığında Chebyshev eşitsizliği kullanılarak

$$\lim_{n \rightarrow \infty} P\left(\left|\frac{Z_n}{n} - a\right| > \varepsilon\right) = 0 \quad (1.26)$$

elde edileceğini söylemektedir.

Güçlü büyük sayılar kanunu (GBSK) ise, neredeyse tüm  $\omega \in \Omega$  için

$$\lim_{n \rightarrow \infty} \left(\frac{Z_n(\omega)}{n}\right) = a \quad (1.27)$$

olacağını söylemektedir.

Aynı zamanda bu bilgileri kullanarak RD'nin bilinmeyen dağılım fonksiyonu ( $\varphi$ ) de tahmin edilebilir.  $Y_1(\omega), Y_2(\omega), \dots, Y_n(\omega)$  gözlemlerinin yapıldığı bir deneyde  $\varphi(t)$ 'nin tahmin edilmesi için

$$F_n(t, \omega) = \frac{1}{n} \sum_{k=1}^n I_{(-\infty, t]}(Y_k(\omega)) \quad (1.28)$$

şeklinde yazıldığında  $N$  sayısının yeterince büyük olduğu bir çalışmada  $\lim_{n \rightarrow \infty} F_n(t, \omega)$  değeri  $\varphi(t)$  değerine yakınsar.

Bu bilgi ile ve  $X_k = I_{(-\infty, t]}(Y_k(\omega))$  olmak üzere Bernoulli süreci için

$$p = P(X_k = 1) = P(Y_k \leq t) = \varphi(t) \quad (1.29)$$

yazılıp, böylece

$$\lim_{n \rightarrow \infty} F_n(t, \omega) = \lim_{n \rightarrow \infty} \frac{N_n(\omega)}{n} = p = \varphi(t) \quad (1.30)$$

elde edilebilir. Burada  $n$  ve  $\omega$  sabit alındığında  $t \rightarrow F_n(t, \omega)$  bir basamak fonksiyonu olur. Bu örneklemin yarattığı dağılıma *deneyisel dağılım* denilmektedir.

Merkezi limit teoremi ile bağımsız RD toplamlarının ortalama  $na$ , varyans  $nb^2$  ile şu eşitliği sağladığı söylenebilir:

$$\lim_{n \rightarrow \infty} P\left(\frac{Z_n - na}{bn^{1/2}} \leq t\right) = \int_{-\infty}^t (2\pi)^{-1/2} e^{-x^2/2} dx \quad -\infty < t < \infty \quad (1.31)$$

Bu teorem, ZBSK ve GBSK ile,  $b^2$  sonlu olduğu sürece  $Y_n$ 'in her bir dağılımı için geçerli olur.

## 1.2 Markov Zincirleri

Markov zincirleri (MZ) saklı Markov modelleri kavramının anlaşılmasında önemli rol oynar. Saklı Markov modelleri aslında Markov modellerinin sisteme saklı bilgisi eklenerek rassallığın bir kat daha artırıldığı sistemlerdir.

### 1.2.1 Genel Bilgiler

Markov süreçlerinin teorik alt yapısı son derece sağlam temellere dayandırılmıştır, aynı zamanda pratikte de birçok farklı tipteki problem çözümünde kullanılan bir çalışma alanıdır. Dolayısıyla günlük hayatta karşılaşılan olguları modellemede çok fazla kullanılan bir süreçtir.

$\Omega$  örnek uzayı,  $P$  de bunun üzerinde tanımlanmış bir olasılık ölçüsü ile,  $X = \{X_n; n \in \mathbb{N}\}$  rassal süreci, sayılabilir  $E$  durum uzayında tanımlanmış olsun öyle ki, her bir  $n \in \mathbb{N}$  ve  $\omega \in \Omega$  için  $X_n(\omega) \in E$  'dir. Bu yapıda  $X_n = j$ , sürecin  $n$  zamanında  $j$  durumunda olması halini gösterir.  $P(X_n = j)$  ise sürecin  $n$  zamanında  $j$  durumunda bulunma olasılığıdır.

Markov süreçleri durum uzayı kesikli değerlerden oluştuğunda *Markov zinciri* adını alır. Tez çalışmalarında sayılabilir bir durum uzayı kümesi kullanılacağından buradaki çalışma Markov zincirleri ile sınırlı kalmıştır.

$X = \{X_n; n \in \mathbb{N}\}$  rassal süreci  $P(X_{n+1} = j | X_0, X_1, \dots, X_n) = P(X_{n+1} = j | X_n)$  özelliğini taşıyorsa bu sürece *Markov zinciri* denilmektedir. Bu çalışmada da Markov zincirlerinin

$$P(X_{n+1} = j | X_n = i) = P(i, j), i, j \in E \quad (1.32)$$

şeklinde gösterilen özel hali ele alınmıştır. Eş. (1.32) *zaman homojen bir Markov zincirini* ifade eder.

$P(i, j)$  olasılıklarına geçiş olasılıkları denilmektedir.  $P(i, j) \forall i, j \in E$  ile oluşturulan matrise *geçiş olasılıkları matrisi*, kısaca *geçiş matrisi* denilmektedir. Geçiş matrisi, stokastik bir matristir. Geçiş matrisi şu şekilde gösterilir:

$$P = \begin{bmatrix} P(0,0) & P(0,1) & P(0,2) & \cdot \\ P(1,0) & P(1,1) & P(1,2) & \cdot \\ P(2,0) & P(2,1) & P(2,2) & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{bmatrix} \quad (1.33)$$

Geçiş matrisi bir Markov zinciri için *Markov matrisi* olarak da adlandırılmaktadır. Geçiş matrisi şu özellikleri taşır:

$$P(i, j) \geq 0 \quad \forall i, j \in E \quad (1.34)$$

$$\sum_{j \in E} P(i, j) = 1 \quad \forall i \in E$$

Geçiş matrisi bunlara ek olarak

$$\sum_{i \in E} P(i, j) = 1 \quad \forall j \in E \quad (1.35)$$

özelliğine sahip olduğunda ise *ikili Markov matrisi* adını alır. Böyle bir süreç için  $\Omega$ , tüm  $\omega = (\omega_0, \omega_1, \omega_2, \dots) \forall i \in \mathbb{N}$  için  $\omega_i \in E$  serilerinin bir kümesi olarak tanımlanmaktadır. Markov özelliği gösteren bir süreç için  $i, j, k \in E$  olmak üzere şu eşitlikler yazılabilir:

$$P(X_6 = j, X_7 = k | X_5 = i) = P(X_7 = k | X_5 = i, X_6 = j)P(X_6 = j | X_5 = i) \quad (1.36)$$

Aynı zamanda yine tanımdan

$$P(X_7 = k | X_5 = i, X_6 = j) = P(X_7 = k | X_6 = j) \quad (1.37)$$

olmaktadır. Zaman homojenliği varsayımı altında

$$P(X_7 = k | X_6 = j) = P(j, k) \text{ ve } P(X_6 = j | X_5 = i) = P(i, j) \quad (1.38)$$

şeklinde ele alınacağından

$$P(X_6 = j, X_7 = k | X_5 = i) = P(i, j)P(j, k) \quad (1.39)$$

olur. Bu eşitlik genelleştirildiğinde, her bir  $n, m \in \mathbb{N}$  ve  $m \geq 1$  için ve  $i_0, \dots, i_m \in E$  olmak üzere,

$$P(X_{n+1} = i_1, \dots, X_{n+m} = i_m | X_n = i_0) = P(i_0, i_1)P(i_1, i_2) \dots P(i_{m-1}, i_m) \quad (1.40)$$

$\pi, E$  üzerinde bir olasılık dağılımı olarak tanımlandığında

$$P(X_0 = i) = \pi(i) \quad \forall i \in E \quad (1.41)$$

şeklinde gösterilir. Bu tanımları da kullanarak genel eşitlik,

$$P(X_0 = i_0, X_1 = i_1, \dots, X_m = i_m) = \pi(i_0)P(i_0, i_1) \dots P(i_{m-1}, i_m) \quad (1.42)$$

haline dönüşmektedir. Bu bilgilerden yararlanarak

$$P(X_{n+1} = i, X_{n+2} = j, X_{n+3} = k | X_n = h) = P(h, i)P(i, j)P(j, k) \quad (1.43)$$

yazılabilir. Ayrıca

$$P(X_{n+3} = k | X_n = h) = \sum_{i \in E} P(h, i) \sum_{j \in E} P(i, j)P(j, k) \quad (1.44)$$

olacaktır. Yukarıdaki eşitlikte  $P(i, j)P(j, k)$  değeri  $\forall j \in E$  için toplandığında  $P$  matrisinin karesinin  $(i, k)$  girdisini verir. Bu bilgi ile

$$P(X_{n+3} = k | X_n = h) = \sum_{i \in E} P(h, i)P^2(i, k) \quad (1.45)$$

aynı kural tekrar uygulandığında

$$P(X_{n+3} = k | X_n = h) = P^3(h, k) \quad (1.46)$$

eşitlikleri yazılabilir. Bu eşitlik genelleştirilirse her bir  $m \in \mathbb{N}$  için,

$$P(X_{n+m} = j | X_n = i) = P^m(i, j) \quad \forall i, j \in E \text{ ve } n \in \mathbb{N} \text{ (} m = 0 \text{ ise } P^0 = I \text{)}, \quad (1.47)$$

olmaktadır. Bir zincirin  $i$  durumundan  $j$  durumuna  $m$  adımda hareket olasılığı geçiş matrisinin  $m$ . kuvvetinin  $(i, j)$  girdisidir. Matris bilgisi ile açıktır ki,

$$P^{m+n} = P^m P^n \quad (1.48)$$

Bu eşitlik açık halde şu şekilde ifade edilir:

$$P^{m+n}(i, j) = \sum_{k \in E} P^m(i, k)P^n(k, j) \quad i, j \in E \quad (1.49)$$

### 1.2.2 Belli bir durumu ziyaret

$P(A | X_0 = i)$ ,  $P_i(A)$  ve  $E[Y | X_0 = i]$ ,  $E_i[Y]$  ile gösterilmek üzere belli bir  $i \in E$  için  $P_i$ ,  $\Omega$  üzerinde tanımlanmış bir olasılık uzayı ve bir  $X$  sürecinin  $\omega$  gerçekleşmesi için  $j \in E$  olmak üzere her bir  $\omega \in \Omega$  için  $N_j(\omega)$ ,  $j$  durumuna toplam ziyaret sayısı olarak tanımlandığında,



$N_j(\omega)$  sonlu bir değere sahipse  $X$  sonunda  $j$  durumdan ayrılır ve geri dönmez. Bu, şu koşulları sağlayan bir  $n$  değişkeninin olduğu anlamına gelir:  $X_n(\omega) = j$  ve  $X_m(\omega) \neq j$   $m > n$ . Eğer  $N_j(\omega)$  sonsuz bir değere sahipse  $j$  durumu süreç boyunca ziyaret eder ve yukarıdaki kurulumda  $m > n$  değeri sağlayan bir  $n$  değeri bulunamaz. Ayrıca  $T_1(\omega), T_2(\omega), \dots$  süreci de  $n \geq 1$  için  $X_n(\omega) = j$ ,  $j$  durumuna gelişlerin sırasını gösterdiğinde, eğer böyle bir  $n$  değeri yoksa  $T_1(\omega) = T_2(\omega) - T_2(\omega) = \dots = \infty$  olur.  $n$ 'e kadar geçen zamanda  $j$  değerine sonlu olan bir  $m$  sayıda ulaşırsa, bu durumda  $T_1(\omega), \dots, T_m(\omega)$   $n$ 'e kadar değerler alacak, geri kalan  $T_{m+1}(\omega) - T_m(\omega) = T_{m+2}(\omega) - T_{m+1}(\omega) = \dots = \infty$  olacaktır.  $F(i, j)$ ,  $i$  durumundan başlayan  $X$  sürecinin, süreç boyunca  $j$  durumuna gelme olasılığını gösteren RD olmak üzere

$$F_k(i, j) = P_i(T_1 = k) \quad i \in E, k = 1, 2, 3, \dots \quad (1.50)$$

ise,

$$F_k(i, j) = \begin{cases} k = 1 \text{ için } P(i, j) \\ k \geq 2 \text{ için } \sum_{b \in E - \{j\}} P(i, b) F_{k-1}(b, j) \end{cases} \quad (1.51)$$

olur. Süreç boyunca  $i$  durumundan başlayıp  $j$  durumuna bir kere gelme olasılığı

$$F(i, j) = P_i(T_1 < \infty) = \sum_{k=1}^{\infty} F_k(i, j) \quad (1.52)$$

şeklinde tanımlandığında,

$$F(i, j) = P(i, j) + \sum_{b \in E - \{j\}} P(i, b) F(b, j) \quad i \in E \quad (1.53)$$

yazılabilir. Bu eşitlik doğrusal denklem sistemi yaratır. Fakat  $F(i, j)$ 'i bu denklem sistemini çözmeden bulmanın yolları da vardır.

$j$ 'e olan ziyaretlerin sayısı  $N_j(\omega)$  olmak üzere,  $\forall \omega$  için ancak ve ancak

$$T_1(\omega) < \infty, \dots, T_m(\omega) < \infty, T_{m+1}(\omega) = \infty \quad (1.54)$$

olduğunda

$$N_j(\omega) = m \quad (1.55)$$

olacaktır. Aşağıda gösterilen olaylar birbirinden bağımsızdır

$$\{T_1(\omega) < \infty\}, \{T_2(\omega) - T_1(\omega) < \infty\}, \dots, \{T_m(\omega) - T_{m-1}(\omega) < \infty\}, \{T_{m+1}(\omega) - T_m(\omega) = \infty\} \quad (1.56)$$

ve  $i$ 'den başlamak üzere olasılıkları da sıra ile

$$F(i, j), F(j, j), \dots, F(j, j), 1 - F(j, j) \quad (1.57)$$

olmaktadır. Böylelikle  $N_j, j$  durumuna toplam ziyaret sayısı olduğunda

$$P_j(N_j = m) = F(j, j)^{m-1}(1 - F(j, j)), m = 1, 2, \dots \text{ ve } i \neq j \text{ için} \quad (1.58)$$

olup böylelikle

$$P_i(N_j = m) = \begin{cases} 1 - F(i, j) & m = 0 \\ F(i, j)F(j, j)^{m-1}(1 - F(j, j)) & m = 1, 2, \dots \end{cases} \quad (1.59)$$

elde edilir. Bu bilgi ile tüm  $m$ 'ler için toplam alındığında  $N_j$ 'in sonlu değere sahip olduğu görülebilir.

$F(j, j) = 1$  ise tüm terimler sıfır olup, toplam sıfır olmaktadır.  $F(j, j) < 1$  ise bir geometrik dağılıma sahip olunacağından toplam bir olur. Böylece kısaca şu yazılabilir:

$$P_j(N_j < \infty) = \begin{cases} 1, & F(j, j) < 1 \\ 0, & F(j, j) = 1 \end{cases} \quad (1.60)$$

Bu tanımlamalara göre;

Başlangıç durumu  $j$  ve  $F(j, j) = 1$  ise, 1 olasılıkla  $N_j = \infty$ , böylece

$$E_j[N_j] = \infty \quad (1.61)$$

olmaktadır.

Başlangıç durumu  $j$  ve  $F(j, j) < 1$  ise  $N_j$ ,  $p = 1 - F(j, j)$  başarı olasılığı ile geometrik dağılıma sahip olup bu durumda

$$E_j[N_j] = 1/p = 1/(1 - F(j, j)) \quad (1.62)$$

olmaktadır. Bu ve  $i \neq j$  için benzer hesaplamalar şu bilgileri verir:

$$R(i, j) = E_i[N_j] \quad (1.63)$$

Eş. (1.63) ile verilen ifadede,  $i, j$  girdisi  $R(i, j)$  olan  $R$  matrisine  $X$  sürecinin *potansiyel matris*'i denilmektedir. Sonuç olarak

$$R(j, j) = 1/(1 - F(j, j)) \quad (1.64)$$

ve

$$R(i, j) = F(i, j)R(j, j) \text{ eğer } i \neq j \text{ ise} \quad (1.65)$$

olmaktadır.

Burada  $j$  durumuna sonlu ya da sonsuz sıklıkta ziyaret edilmesinin  $F(j, j)$  değerine bağlı olduğu görülür.  $F(j, j) = 1$  ise  $j$ 'e ziyaret sonsuz sıklıkta olur bu durumda  $j$  sonsuz kere ziyaret edilir böyle bir durum *kalıcı durum* olarak adlandırılır;  $F(j, j) < 1$  ise sonlu bir sayıda ziyaret olacaktır, daha sonra ziyaret olmayacaktır, böyle bir  $j$  durumuna *geçici durum* denilmektedir.

Durumlara olan ziyaret sayılarının beklenen değerleri, doğrudan, sonuç olarak çıkan bilgilerle hesaplanabilir. Pratikte ise önce  $R(i, j)$  değerinin hesabı yapıлып, sonra  $R(i, j)$  bilgisi kullanılarak,  $F(i, j)$  değeri hesabı yapılır.

Yani  $1_j(k) = 1$  eğer  $k = j$  ise ve  $1_j(k) = 0, k \neq j$  ise. Bu durumda  $\omega \in \Omega$  için  $1_j(X_n(\omega))$   $n$  zamanında  $X = j$  olduğunda 1,  $X \neq j$  olduğunda 0 olacaktır. Bu durumda  $1_j(X_n(\omega))$   $n$  zamanında  $j$  durumuna olan ziyareti göstermektedir. Böylelikle

$$N_j(\omega) = \sum_{n=0}^{\infty} 1_j(X_n(\omega)) \quad \omega \in \Omega \quad (1.66)$$

olmaktadır. *Monoton yakınsama teoremi* (monotone convergence theorem) ile şu yazılabilir:

$$\begin{aligned} R(i, j) &= E_i[\sum_{n=0}^{\infty} 1_j(X_n)] \\ &= \sum_{n=0}^{\infty} E_i[1_j(X_n)] \\ &= \sum_{n=0}^{\infty} P_i[X_n = j] \\ &= \sum_{n=0}^{\infty} P^n(i, j) \end{aligned} \quad (1.67)$$

Matris notasyonu ile bu eşitlik

$$R = I + P + P^2 + \dots \quad (1.68)$$

şeklinde yazılabilir. Burada  $P$  matrisleri önceden hesaplandığında  $R$  doğrudan bulunabilir. Ne var ki  $R$  matrisinin şu şekilde ifade edilebilen bir özelliği vardır:

$$RP = PR = P + P^2 + \dots = R - I \quad (1.69)$$

yani

$$R - RP = I \quad (1.70)$$

olacaktır bu da

$$R(I - P) = (I - P)R = I \quad (1.71)$$

olup, buradan da şu eşitlik elde edilir:

$$R = (I - P)^{-1} \quad (1.72)$$

### 1.2.3 Durumların sınıflandırılması

$j$  durumu şu özelliği sağladığında *kalıcı (recurrent) durumdur*:  $P_j(T_1 < \infty) = 1$ , diğer türlü bu durum *geçici durumdur*:  $P_j(T_1 = \infty) > 0$

Eğer kalıcı bir  $j$  durumu şu özelliği gösteriyorsa *etkisiz kalıcı durumdur*:  $E_j[T] = \infty$ , diğer türlü *pozitif kalıcı durumdur*.

Kalıcı bir durum olan  $j$  için,

$$P_j\{T = n\delta, n \geq 1 \text{ için}\} = 1 \quad (1.73)$$

özelliğini sağlayan  $n$ 'in en büyük ortak böleni  $\delta$  olduğunda,  $\delta = 1$  ise durum *aperiyodik*,  $\delta \geq 2$  ise durum  $\delta$  periyodu ile *periyodiktir*.

$j$  durumunun *kalıcı* olması halinde tanım gereği  $F(j, j) = 1$  olmaktadır. Diğer bir deyişle  $j$ 'den başlayan sistemin ancak ve ancak  $j$ 'e tekrar dönme olasılığının 1 olması durumunda  $j$  durumu kalıcı durumdur.  $F(j, j) = 1$  olması  $R(j, j) = \infty$  olmasını getireceğinden  $j$  durumuna dönüş sayısının beklenen değeri  $\infty$  olacaktır. Sonuç olarak  $j$  durumu ancak ve ancak  $R(j, j) = E_j[N_j] = \infty$  koşulları sağlandığında kalıcı olacaktır ki bu da  $P_j(N_j = \infty) = 1$  eşitliğine denktir. Böylelikle ayrıca  $P_j(T_1 < \infty) = 1$  olur.

$j$  durumu *geçici* ise  $F(j, j) < 1$  olacak böylelikle  $j$  durumuna dönmeme ile ilgili pozitif bir  $1 - F(j, j)$  olasılığından bahsedilebilir. Bu durumda  $R(j, j) = E_j[N_j] < \infty$  olacaktır.  $j$  durumuna dönüş sayısının beklenen değeri sonlu olup, bu da  $P_j(N_j < \infty) = 1$  eşitliğine denktir. Ayrıca  $P_j(T_1 = \infty) > 0$  olmaktadır.

$j$  durumu geçici olduğunda  $R(i, j) = F(i, j)R(j, j) \leq R(j, j) < \infty$  böylelikle  $R(i, j) < \infty$  olmaktadır.  $R(i, j)$ 'nin bütün  $n$ 'ler için  $P^n(i, j)$  toplamı olduğu bilinmektedir. Bu durumda  $R(i, j)$  ancak

$$n \rightarrow \infty \text{ iken } P^n(i, j) \rightarrow 0 \quad (1.74)$$

olduğunda sonlu olabilir.

$j$  durumu kalıcı olduğunda  $R(j, j) = \infty$  olduğundan  $P^n(j, j)$ 'in limitte 0 ya da pozitif olacağına kolaylıkla karar verilemez.

$j$  durumu etkisiz kalıcı durum ise,  $j$  durumuna iki geliş arasında geçen zamanın beklenen değeri sonsuz olacak bu durumda

$$\lim_{n \rightarrow \infty} P^n(j, j) = 0 \quad (1.75)$$

olacaktır. Bu bilgiler şu şekilde özetlenebilir:

- $j$  geçici ya da etkisiz kalıcı bir durum ise

- $\lim_{n \rightarrow \infty} P^n(i, j) = 0 \quad \forall i \in E$

olur.

- $j$  kalıcı pozitif aperiyodik bir durum ise

- $\pi(j) = \lim_{n \rightarrow \infty} P^n(j, j) > 0$

- $\lim_{n \rightarrow \infty} P^n(i, j) = F(i, j)\pi(j) \quad \forall i \in E$

$P^n(j, j) > 0$  ve  $n \geq 1$  için  $\delta$  en büyük ortak bölen olduğunda,  $\delta = 1$  ise  $j$  *aperiyodik*,  $\delta \geq 2$  olduğunda ise  $j$   $\delta$  ile *periyodik* olur.

$P^n(j, j) > 0$  olduğunu göstermek için  $P^n(j, j)$  değerini bulmak gerekmez. Mesela,

$$P^3(j, j) = \sum_i \sum_k P(j, i)P(i, k)P(k, j) > 0 \quad (1.76)$$

olduğunu göstermek için bulmak için  $P(j, i), P(i, k), P(k, j)$  değerlerinin pozitif olduğunu bilmek yeterlidir. Yani  $P^n(j, j) > 0$  olması için yeter şart  $i_1, \dots, i_{n-1} \in E$  olmak üzere  $P(j, i_1), P(i_1, i_2), \dots, P(i_{n-1}, j)$  değerlerinin pozitif olmasının sağlanabiliyor olmasıdır.

$P^n(i, j) > 0$  eşitsizliğini sağlayan bir  $n \geq 0$  değerinin olduğu biliniyorsa  $j$  durumu  $i$  durumundan *erişilebilir* denir ve  $i \rightarrow j$  şeklinde gösterilir. Eğer  $i \neq j$  ise ancak ve ancak  $F(i, j) > 0$  ise  $i \rightarrow j$  olmaktadır  $P^0(j, j) = 1 > 0$  olduğundan bir durumdan kendisine her zaman erişilebilir.  $i \rightarrow j$  olabilmesi için  $P(i, i_1) > 0, P(i_1, i_2) > 0, \dots, P(i_{n-1}, j) > 0$  olan bir sıra  $i_1, i_2, \dots, i_{n-1}$  durumunun sistemde bulunması gerekmektedir.

Bir durum kümesinde küme içindeki bir durumdan, küme dışındaki herhangi bir duruma erişilemiyorsa bu kümeye *kapalı küme* denir.

Kendi başına bir kapalı küme yaratan duruma *yutucu durum (absorbing state)* denir. Herhangi bir  $i$  yutucu durumu  $P(i, j) = 0 \forall j \neq i$  olur yani kendi dışında herhangi farklı bir durumdan kendine erişilemez. Bir  $j$  durumunun yutucu olma ölçütü  $P(j, j) = 1$  olmasıdır.

Bir kapalı kümenin herhangi bir alt kümesi kapalı küme oluşturmuyorsa bu kümeye *indirgenemez (irreducible)* denir. Bir MZ'nin indirgenemez olma ölçütü zincir içindeki tüm durumlardan diğer tüm durumlara erişim olmasıdır.

Bir Markov zincirinde tek bir kapalı küme varsa ve bu küme tüm durumları kapsıyorsa bu Markov zincirine *indirgenemez Markov zinciri* denir.

Buna göre,

- $j$  kalıcı bir durumsa:  $j \rightarrow k$  olduğunda  $k \rightarrow j$  olur ve yine aynı koşul ile  $F(k, j) = 1$  olacaktır.
- Kalıcı durum ancak kalıcı durumlara erişilebilir. Böylelikle bütün kalıcı durumların kümesi kapalıdır. Fakat bu kapalı küme içinde 1'den fazla kapalı küme olabilir.
- Her bir kalıcı durum  $j$ 'in dâhil olduğu bir *indirgenemez kapalı C* kümesi bulunmaktadır.

- Kapalı  $C$  kümesinde bulunan kalıcı durumlar eşsiz bir şekilde indirgenemez kapalı kümelere ayrılabilir: ( $C = C_1 \cup C_2 \cup \dots$ ). Böyle bir kapalı kümeler elde etme metodunda geriye kalan geçici durumlar da olacaktır. Böyle bir durumdan geçici durumlardan kalıcılara erişim olur tersi olmaz.
- Yukarıdaki gibi bir işlemde şu şekilde bir  $P = \dots$  matrisi elde edilebilir. Her bir  $C_1, C_2, \dots$  kapalı kümesi ayrı bir  $P_1, P_2$  alt matrisleri ile ifade edilirse bu  $P_1, P_2, \dots$  matrisleri  $P$ 'nin köşegenine gelecek biçimde yerleştirilebilir. Bu  $P$ 'lerin her biri kendi başına bir *indirgenemez* MZ'dir. Geçici durumlar ise en alta yerleştirilebilir.
- Eğer bir MZ indirgenemezse *ya* tüm durumlar geçici, *ya* tüm durumlar etkisiz kalıcı *ya da* tüm durumlar pozitif kalıcı durumdur. Bu sistemde bütün durumlar *ya* aperiyoiktir *ya da* eğer durumlardan biri  $\delta$  periyodu ile periyodik ise tüm durumlar da aynı periyodla periyodik olacaktır Eğer sistemde herhangi bir  $j$  durumu için  $P(j, j) > 0$  ise bu durumda tüm sistem aperiyoiktir olur.
- $C$  sonlu sayıda durumlu indirgenemez kapalı küme olduğunda,  $C$ 'deki hiç bir durum etkisiz kalıcı değildir. Yine böyle bir kurulumda geçici durum bulunmamaktadır.
- Durum uzayı  $E$ 'nin sonlu olduğu bir kurulumda hiç bir durum etkisiz kalıcı değildir ve bütün durumlar geçici olamaz.

Bir *indirgenemez* MZ için şu şekilde bir doğrusal denklem sistemi geçerli ise,

$$v(j) = \sum_{i \in E} v(i)P(i, j), \quad j \in E \quad (1.77)$$

bu durumda ancak ve ancak aşağıdaki özellikteki bir  $v$ 'nin çözüm olması halinde, bu MZ'deki tüm durumlar *kalıcı pozitif*dir:

$$\sum_{j \in E} v(j) = 1 \quad (1.78)$$

Yukarıdaki verilen eşitliklerin  $v$  için çözümü olması durumunda  $v(j) > 0 \forall j \in E$  olacaktır ve başka sonuç olmayacaktır.

$X, P$  geçiş matrisi ile indirgenemez bir MZ olduğunda,  $Q$  matrisi  $k \in E$  olmak üzere belli  $k$ . sıra ve  $k$ . satırın silinmesi ile oluşan bir matris olduğunda,  $Q$ 'nun tüm durumların kalıcı olması için tek çözüm



$$h(i) = \sum_{j \in E_0} Q(i, j)h(j), 0 \leq h(i) \leq 1, i \in E_0; \quad (1.79)$$

eşitliği için

$$h(i) = 0 \forall i \in E_0 \text{ ve } E_0 = E - \{k\} \quad (1.80)$$

olması halinde sağlanır.

### 1.3 Limit Davranışları

#### 1.3.1 R ve F'nin hesaplanması

$R(i, j)$ :  $i$ 'den başlayan sürecin  $j$ 'e ziyaret sayılarının beklenen değeri,

$F(i, j)$ :  $i$ 'den başlayan sürecin  $j$  durumunu 1 kere ziyaret etme olasılığı olmak üzere Çizelge

1.1.'deki kurallar geçerli olacaktır:

Çizelge 1.1. R ve F'nin hesaplanması

Kural	F = 0 ise R = 0	F = 1 ise R = ∞	0 < F < 1 ise 0 < R < ∞
i: kalıcı	j: kalıcı (aynı sınıf içinde)	F(i, j) = 1 (i = j için de geçerli)	R(i, j) = ∞ (i = j için de geçerli)
i: kalıcı	j: kalıcı (farklı sınıflarda)	F(i, j) = 0	R(i, j) = 0
i: geçici	j: kalıcı (i'den j'ye erişim varsa)	F(i, j) = 1	R(i, j) = ∞
i: geçici	j: kalıcı (i'den j'ye erişim yoksa)	F(i, j) = 0	R(i, j) = 0
i: kalıcı	j: geçici	F(i, j) = 0	R(i, j) = 0
i: geçici	j: geçici	F(j, j) = 1 - 1/R(j, j) ve F(i, j) = R(i, j)/R(j, j)	R(i, j) < ∞

$i$  ve  $j$  geçici durumlar ve  $D$  tüm geçici durumları gösteren küme olmak üzere,  $P$  ve  $R$  matrislerinden tüm kalıcı durumların silinmesi ile elde edilen matrisler  $Q$  ve  $S$  olarak tanımlandığında şu eşitlikler geçerlidir:

$$Q(i, j) = P(i, j) \text{ ve } S(i, j) = R(i, j) \quad i, j \in D \quad (1.81)$$

Bu şekilde  $Q$  ve  $S$  elde edildikten sonra kalıcı durumlar geçici durumlardan daha küçük numaralandırılarak  $P$  düzenlendiğinde;  $K$  kalıcı durumlar arasındaki geçiş olasılıklarını,  $Q$

geçici durumlar arasındaki geçiş olasılıklarını göstermek üzere  $P$  matrisi ile şu şekilde yazılabilir:

$$P = \begin{bmatrix} K & 0 \\ L & Q \end{bmatrix} \quad (1.82)$$

ve

$$P^m = \begin{bmatrix} K^m & 0 \\ L_m & Q^m \end{bmatrix} \quad (1.83)$$

Böylelikle

$$R = \sum_{m=0}^{\infty} P^m = \begin{bmatrix} K^m & 0 \\ L_m & Q^m \end{bmatrix} \quad (1.84)$$

ve bu gösterir ki

$$S = \sum_{m=0}^{\infty} Q^m = I + Q + Q^2 + \dots \quad (1.85)$$

ve

$$SQ = QS = Q + Q^2 + \dots = S - I \quad (1.86)$$

buradan

$$(I - Q)S = I \text{ ve } S(I - Q) = I \quad (1.87)$$

şeklinde iki doğrusal denklem sistemi elde edilir.

$D$  kümesi sonlu ise

$$S = (I - Q)^{-1} \quad (1.88)$$

'dir. Eğer  $D$  sonlu değilse,  $S(I - Q) = I$  birden fazla çözüme sahip olacaktır. Bu durumda da  $S$  en küçük değerli çözüm olur.

1.  $i, j$  kalıcı ise ve aynı sınıftalar ise  $F(i, j) = 1$ , aynı sınıfta değilse  $F(i, j) = 0$ .
2.  $i$  kalıcı  $j$  geçici ise  $F(i, j) = 0$
3.  $i, j$  geçici ise  $R(i, j) < \infty$  ve  $F(j, j) = 1 - 1/R(j, j)$ ;  $F(i, j) = R(i, j)/R(j, j)$
4.  $i$  geçici,  $j$  kalıcı ise, " $j$  ve  $k$ "  $\in C$  ve  $C$  indirgenemez kalıcı durumların kapalı kümesi olduğunda, herhangi bir geçici  $i$  durumu için,  $F(i, j) = F(i, k)$  eşitliği yazılabilir.

$C_1, C_2, \dots$  indirgenemez kalıcı sınıflar ve  $D$  geçici durumların bir kümesi olduğunda, bu sistemin matrisini kalıcı sınıf içi durumlar bir arada, geçici durumlar da en büyük indeks değerlerini alacak şekilde düzenlendiğinde, ayrıca her bir kalıcı sınıftan geri çıkılamayacağından kalıcı sınıfların her biri yutucu durum olarak adlandırılarak indekslendiğinde ve

$$b_j(i) = \sum_{k \in C_j} P(i, k) \quad i \in D \quad (1.89)$$

olarak tanımlandığında

$$\hat{P} = \begin{bmatrix} I & 0 \\ B & Q \end{bmatrix} \quad (1.90)$$

şeklinde matris elde edilir. Burada

$$B(i, j) = \sum_{k \in C_j} P(i, k) \quad i \in D, j = 1, 2, \dots \quad (1.91)$$

olmaktadır. Buradan da

$$\hat{P}^n = \begin{bmatrix} I & 0 \\ B_n & Q^n \end{bmatrix} \quad (1.92)$$

elde edilir ve

$$B_n = (I + Q + \dots + Q^{n-1})B \quad (1.93)$$

olacaktır. Burada  $B_n(i, j)$   $i'$ den başlayıp  $C_j$  kalıcı sınıfına  $n$  ya da daha önceki adımda girme olasılığı olmaktadır.

$$G = \lim_{n \rightarrow \infty} B_n = (\sum_{k=0}^{\infty} Q^k)B = SB \quad (1.94)$$

Yani  $G(i, j)$  her bir  $i$  geçici durum için  $C_j$  kalıcı sınıfına erişim olasılığıdır.

$S$  daha önce hesaplandığı gibi hesaplandığında

$$G = SB \quad (1.95)$$

$G$  matrisinin elde edilmesini sağlar. Her bir geçici durum  $i$  ve kalıcı sınıf  $C_j$  için

$$G(i, j) = F(i, k) \forall k \in C_j \quad (1.96)$$

olacaktır. Burada eğer sadece 1 kalıcı sınıf var ise ve sonlu sayıdaki geçici durum için

$$F(i, j) = 1 \forall j \in C \quad (1.97)$$

olur.

### 1.3.2 Kalıcı durumlar ve limit olasılıkları

$X$  sürecinin *indirgenemez* ve *aperiyodik* olması koşulu ile ancak ve ancak aşağıdaki doğrusal denklem sistemlerinin çözümü olan bir  $\pi$  var ise bütün durumlar *pozitif kalıcı*'dir. Eğer bir çözüm  $\pi$  var ise, bu çözüm kesin olarak pozitifdir ve  $\pi$  için başka çözüm bulunmamaktadır.

$$\pi(j) = \sum_{i \in E} \pi(i)P(i, j), j \in E \quad (1.98)$$

$$\sum_{j \in E} \pi(j) = 1 \quad (1.99)$$

Bu koşulları sağlayan bir sistem için

$$\pi(j) = \lim_{n \rightarrow \infty} P^n(i, j) \forall i, j \in E \quad (1.100)$$

olacaktır. Ayrıca *indirgenemez*  $X$  MZ'sinin *sonlu sayıda* durumu olması halinde *etkisiz kalıcı* ve *geçici durumu* bulunmamaktadır.

$X$  indirgenemez aperiodyk MZ'si sonlu sayıda duruma sahipse

$$\pi P = \pi \quad (1.101)$$

ve

$$\pi 1 = 1 \quad (1.102)$$

denklem sistemi eşsiz çözüme sahip olacaktır.  $\pi$  kesin olarak pozitif (strictly positive) olacaktır. Bu durumda çözüm

$$\pi(j) = \lim_{n \rightarrow \infty} P^n(i, j) \quad \forall i, j \in E \quad (1.103)$$

olur.

$\pi P = \pi$  denklem sistemini sağlayan olasılık dağılımı  $\pi$ ,  $X$ 'in *değişmez dağılımı* (invariant distribution) olarak adlandırılır (bazen de  $P$ 'nin *değişmez dağılımı* denilmektedir). Eğer bir *indirgenemez, aperiodyk MZ, değişmez dağılıma sahipse* bu durumda tüm durumları *kalıcı pozitifdir*, eğer *değişmez dağılıma sahip değilse* tüm durumları ya *geçicidir* ya *etkisiz kalıcıdır*.

Değişmez (invariant) terimi  $\pi = \pi P = \pi P^2 = \dots = \pi P^n \quad \forall n \in \mathbb{N}$  olmasından kaynaklanmaktadır. Bunu eşitlikle göstermek gerekirse  $\pi$ ,  $X$ 'in başlangıç dağılımı ise yani

$$P(X_0 = j) = \pi(j) \quad j \in E \quad (1.104)$$

ise o halde

$$P(X_n = j) = \sum_{i \in E} \pi(i) P^n(i, j) = \pi(j) \quad j \in E \quad \text{her } n \in \mathbb{N} \text{ için} \quad (1.105)$$

olmaktadır.

Yukarıdaki bilgilerle limit olasılıkları  $\lim_{n \rightarrow \infty} P^n(j, j)$  bilgisi kalıcı pozitif, aperiodyk  $j$  durumları için bulunabilir. Yani  $j$  durumunu içeren indirgenmez kapalı bir küme olduğu varsayıldığında, herhangi bir  $i$  ve kalıcı aperiodyk  $j$  durumu için

$$\lim_{n \rightarrow \infty} P^n(i, j) = F(i, j) \lim_{n \rightarrow \infty} P^n(j, j) \quad (1.106)$$

olduğu bilinmektedir. O halde  $F(i, j)$  hesaplandığında  $P^n(i, j)$  limiti tüm  $j$ 'ler için hesaplanabiliyor olacaktır.

Bir önceki çözümden belli bir  $X$  MZ için, belli bir  $j$  durumuna ziyaretler için geçen zamanın IID olduğu görülmektedir.

$j$  aperiodyk pozitif kalıcı durum ve  $m(j)$  de iki geri dönüş arası beklenen zaman olduğunda,

$$\pi(j) = \lim_{n \rightarrow \infty} P^n(i, j) = 1/m(j) \quad (1.107)$$

yazılabilir. Diğer bir deyişle  $j$  durumunda olmanın limit olasılığı  $\pi(j)$ ,  $j$ 'e ziyaret oranı ile aynı olmaktadır. Yani  $j$  durumuna ziyaretin ortalama zamanı  $m(j) = 7.2$  ise bu ortalama 7.2 adımda bir  $j$  durumuna ziyaret edildiği anlamına gelir bu da  $\pi(j)$  değerinin  $1/7.2$  olduğu anlamına gelir.

Daha kesin ifade etmek gerekirse  $j$  kalıcı pozitif ve aperiodyk durum olduğunda ve  $\pi(j) = \frac{1}{m(j)}$  olarak tanımlandığında, neredeyse tüm  $\omega \in \Omega$  için

$$\lim_{n \rightarrow \infty} \frac{\sum_{m=0}^n 1_{j(X_m(\omega))}}{n+1} = \pi(j) \quad (1.108)$$

olmaktadır (Çınlar, 1975).



## 2 FİNANSAL ZAMAN SERİLERİ VE MODELLEMELER

### 2.1 Regresyon Analizi

$n$  sayıda gözlemle elde edilen veri ikilileri şu şekilde gösterilsin:  $(x_i, y_i)$ ,  $i = 1, \dots, n$ . Burada  $X$  bir skalar ya da vektör iken,  $Y$  skalar bir değer olmaktadır (Takezawa, 2006). Bu verilerin arasındaki ilişki,

$$y_i = \mu(x_i) + \varepsilon_i \quad (2.1)$$

regresyon modeli ile gösterilebilir. Burada  $\varepsilon_i$  sıfır ortalamalı, ilişkisiz,  $\sigma^2$  varyanslı rassal değişkenlerdir,  $\mu(x_i)$  ise  $x_1, \dots, x_n$  tasarım noktalarında değer alan bilinmeyen bir fonksiyondur. Bu fonksiyon *regresyon fonksiyonu* ya da *regresyon eğrisi* olarak adlandırılır. Regresyon analizi, regresyon modelindeki  $\mu(t)$  fonksiyonu ile ilgili istatistiksel çıkarımları yapabilmek için gerekli olan yöntemlere karşılık gelmektedir. Model için uygun bir çıkarsama yönteminin belirlenmesi,  $\mu$  üzerine yapılacak varsayımlara bağlı olacaktır. Parametrik regresyon modeli,  $\mu$ 'nün sonlu sayıda parametre değerleri bilinmeyen fakat biçimi belli bir fonksiyon olduğunu varsayar. Yani  $\mathbb{R}^p$ 'nin bir alt uzayında tanımlı  $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3, \dots, \beta_p)^T \in B$  şeklinde bir parametre vektörü ile  $\mu(\cdot; \boldsymbol{\beta}) = \mu(\cdot)$  şeklinde bilinen bir fonksiyonun bulunduğu varsayılır. Böylelikle  $\mu$  hakkında çıkarsama yapmak  $\boldsymbol{\beta}$  hakkında çıkarsama yapmaya denktir (Eubank, 1999).

Parametrik modeller parametreye doğrusal ya da doğrusal olmayan bir usul ile bağlıdır. Örneğin  $\mu(t) = \beta_1 + \beta_2 \exp\{-\beta_3 t^{\beta_4}\}$  modelinde  $\mu(t)$ ,  $\beta_1$  ve  $\beta_2$  ile doğrusal ilişkili iken,  $\beta_3$  ve  $\beta_4$  ile doğrusal olmayan bir ilişkiye sahiptir. Doğrusal olmayan ilişkilerin doğrusal modellerle ele alınması çeşitli sakıncalara neden olabilir. Ne var ki katsayıların üs olduğu tip doğrusal olmayan ilişkiler güç dönüşümü (power transformation) ile doğrusal bir şekilde ele alınabilir (Cook ve Weisberg, 1994). Bununla birlikte doğrusal olmayan modeller ile yapılan öngörülerde doğrusal modellere göre belirgin bir iyileşme gözlemlenmediği çeşitli çalışmalarda belirtilmiştir.

Yukarıdaki gösterimler dışında *saf doğrusal modeller* (purely linear models) ise



$$\mu(t) = \sum_{j=1}^p \beta_j x_j(t) \quad (2.2)$$

şeklinde bilinen fonksiyonlardan oluşurlar (Eubank, 1999). Saf doğrusal modeller ve doğrusal modeller ile ilgili çıkarsamalar doğrusal regresyon analizi olarak bilinmektedir.

Doğrusal regresyon analizi tahmini için genellikle EKK yöntemi kullanılır. Model,  $y$  bağımlı değişken olmak üzere,

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon} \quad (2.3)$$

eşitliği ile ifade edilebilir ve bu durumda parametre tahmin edicileri

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} \quad (2.4)$$

olmaktadır. Burada hata terimi varsayımları sağlanmak üzere  $\hat{\boldsymbol{\beta}}$ 'lerin BLUE özelliği taşıdığı gösterilebilir (Akdi, 2012).

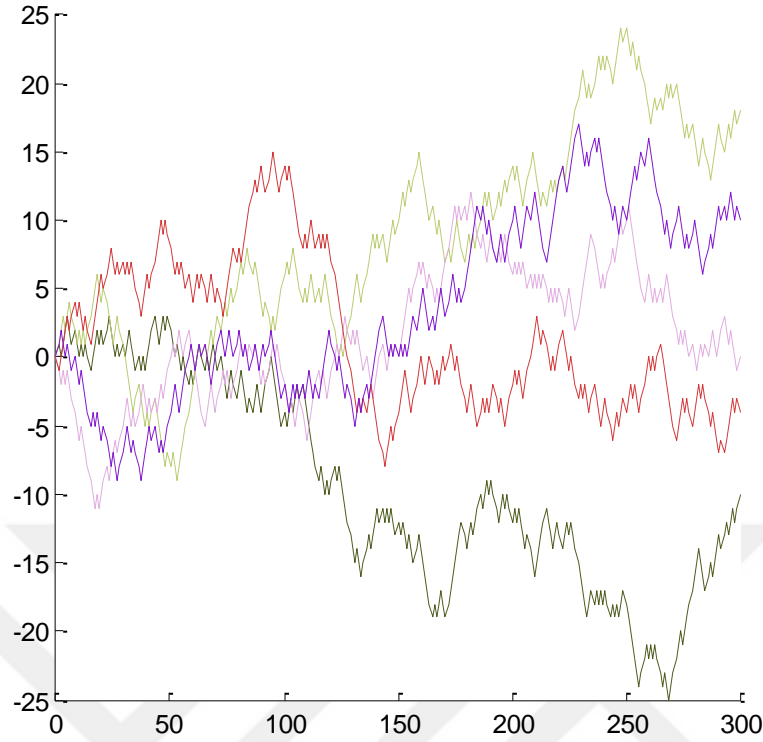
## 2.2 Doğrusal Zaman Serisi Modelleri

$(\Omega, \mathcal{F}, P)$  bir olasılık uzayı,  $T \in \mathbb{Z}$  de bir indis kümesi olmak üzere bir *zaman serisi*  $\Omega \times T$  çarpım uzayından reel sayılara giden bir fonksiyondur. Diğer bir deyişle zaman serisi,

$$\begin{aligned} X(\cdot): \Omega \times T &\rightarrow \mathbb{R} \\ &: (\omega, t) \rightarrow X(\omega, t) \end{aligned} \quad (2.5)$$

şeklinde tanımlanan bir fonksiyondur.  $X(\omega, t)$  yerine bazen  $X_t(\omega)$  ve bazen  $X_t$  gösterimi de kullanılmaktadır.

Bu tanıma göre bir zaman serisi (ZS) her sabit  $t$  için bir RD,  $\omega$  sabit tutulduğunda ise  $t$ 'nin reel değerli bir fonksiyonudur. İkinci durumda elde edilen reel değerli fonksiyona ZS'nin bir *gerçekleşmesi (realizasyonu)* adı verilir. Bu değerler ZS'nin *örneklem patikası*'nı oluşturur. ZS grafikleri buna karşılık gelmektedir (Akdi, 2012). Bir ZS'nin 5 farklı  $\omega$  için gerçekleşmesi Şekil 2.1.'de görülebilir.



Şekil 2.1. Belli bir zaman serisinin 5 farklı  $\omega$  için gerçektelemesi

Olasılıksal bir  $\{X_t\}$  ZS tüm  $t$ 'ler için;  $\mu$  sabit,  $\{Z_t\} \sim WN(0, \sigma^2)$  ve  $\psi_j \in \mathbb{R}$  olmak üzere ve  $\sum_{j=-\infty}^{\infty} |\psi_j| < \infty$  sabit sayılar dizisi iken

$$X_t = \mu + \sum_{j=-\infty}^{\infty} \psi_j Z_{t-j} \quad (2.6)$$

eşitliğini sağlıyorsa doğrusal kabul edilir. Aslında ikinci dereceden durağan bir süreç ya doğrusal bir süreçtir ya da deterministik bileşeni çıkarılarak doğrusal bir sürece dönüştürülebilir (Brockwell ve Davis, 2002).

Doğrusal zaman serileri, ortalama ve varyans denkleminde ve sadece ortalama denkleminde doğrusal olmak üzere iki başlık altında incelenebilir (Tsay, 2010). ARMA modellerini de içeren doğrusal zaman serileri sınıfı, durağan süreçleri çalışmak için uygun bir çerçeve sunmaktadır (Brockwell ve Davis, 2002). ARMA modelleri, durağan olmama da göz önüne alınırsa ARIMA modelleri, hem ortalama hem varyans denkleminde doğrusal modeller sınıfına girmektedir.

### 2.2.1 ARIMA modelleri

Doğrusal zaman serileri modellerinde, yapısal modellerden farklı olarak, bağımlı değişken kendi geçmiş değerlerinden ya da WN özellikteki RD'lerden etkilendiği varsayımı ile modellenir. Bu tür düşünce tarzı ekonomik teori olmadan da modellenmeye imkan verdiği gibi, model öngörü için de uygun bir yapıya sahip olacaktır. Doğrusal ilişkilerin yakalanması anlamında korelasyon doğrusal zaman serilerinin en önde gelen aracıdır. Bu korelasyon, otokorelasyon olarak adlandırılır (Tsay, 2010). Doğrusal zaman serilerinde bu tür bir doğrusal ilişkiyi araştıran iki yöntem, değişkenin değerini değişkenin geçmiş dönem değerleri ile açıklamaya çalışmak ile değişkenin değerini bağımsız aynı dağılıma sahip rassal değişkenlerle açıklamaktır. Bu yöntemlerle oluşturulan modellerden ilki AR ikincisi MA modelleridir. Bu iki yöntemin birleşimi de ayrı bir model olarak ele alınabilir: ARMA.

#### 2.2.1.1 Hareketli ortalama modeli

$e_t \sim WN(0, \sigma^2)$  ve  $q$  sonlu bir doğal sayı ve  $\beta_q \neq 0$  olmak üzere  $MA(q)$  serisi,

$$X_t - \mu = e_t + \sum_{j=1}^q \beta_j e_{t-j} \quad (2.7)$$

şeklinde verilir ve  $X_t \sim MA(q)$  şeklinde gösterilir.

$$E(X_t) = \mu \quad (2.8)$$

'dür çünkü serinin ortalaması sıfırdır.

$$Var(X_t) = Var(e_t + \sum_{j=1}^q \beta_j e_{t-j}) = \sigma^2 \sum_{j=0}^q \beta_j^2 \quad (2.9)$$

$$\begin{aligned} \gamma_X(h) &= Cov(X_t, X_{t+h}) = Cov\left(\sum_{j=0}^q \beta_j e_{t-j}, \sum_{i=0}^q \beta_i e_{t+h-i}\right) \\ &= \sum_{j=0}^q \sum_{i=0}^q \beta_j \beta_i Cov(e_{t-j}, e_{t+h-i}) \end{aligned} \quad (2.10)$$

'dir. Fakat,

$$\text{Cov}(e_{t-j}, e_{t+h-i}) = \begin{cases} \sigma^2, & j = i - h \\ 0, & d.d \end{cases} \quad (2.11)$$

olduğundan

$$\gamma_X(h) = \sigma^2 \sum_{j=0}^{q-|h|} \beta_j \beta_{j+|h|} \quad (2.12)$$

'dir. Çünkü  $j + h$ 'nin alabileceği en yüksek değer  $q$ 'dur. Buradan okovf'un simetrik olması gerektiğinden

$$\gamma_X(h) = \begin{cases} \sigma^2 \sum_{j=0}^{q-|h|} \beta_j \beta_{j+|h|}, & 0 \leq h \leq q \\ 0, & d.d \end{cases} \quad (2.13)$$

buradan da otokorelasyon fonksiyonu (okorf)

$$\rho_X(h) = \begin{cases} \frac{\sum_{j=0}^{q-|h|} \beta_j \beta_{j+|h|}}{\sum_{j=0}^q \beta_j^2}, & 0 \leq h \leq q \\ 0, & d.d \end{cases} \quad (2.14)$$

şeklinde elde edilir. Buradan anlaşılmaktadır ki, MA serileri her zaman durağandır (Akdi, 2012).

### 2.2.1.2 Otoregresif modeller

Otoregresif ZS'de serinin şimdiki değeri, geçmiş değerlerden ve beyaz gürültüden etkilenir. Genel olarak  $e_t \sim WN(0, \sigma^2)$  olmak üzere ve  $\mu$  serinin beklenen değeri olmak üzere,

$$X_t - \mu = \sum_{i=1}^p \alpha_i (X_{t-i} - \mu) + e_t \quad (2.15)$$

şeklinde verilir ve AR(p) ile gösterilir. AR modellerinde durağanlık, karakteristik denklemin köklerinden anlaşılabilir. Yukarıda verilen AR(p) modelinin karakteristik denklemi şu şekildedir:

$$F(m) = m^p - \sum_{i=1}^p \alpha_i m^{p-i} \quad (2.16)$$

AR(p) modelinin varyans ve otokovaryansları Yule-Walker (YW) denklemleri ile elde edilir. YW denklem sistemi ise şu şekilde oluşturulur:  $e_t \sim WN(0, \sigma^2)$  olmak üzere AR(p) modeli

$$X_t = \sum_{i=1}^p \alpha_i X_{t-i} + e_t \quad (2.17)$$

olacaktır ve bu serinin varyansı,

$$\begin{aligned} \gamma(0) &= \text{Var}(X_t) = \text{Cov}(X_t, X_t) = \text{Cov}(X_t, \sum_{i=1}^p \alpha_i X_{t-i} + e_t) \\ &= \sum_{i=1}^p \alpha_i \text{Cov}(X_t, X_{t-i}) + \text{Cov}(X_t, e_t) \\ &= \sum_{i=1}^p \alpha_i \gamma_X(i) + \sigma^2 \end{aligned} \quad (2.18)$$

şeklinde elde edilir. Ayrıca  $h > 0$  için otokovaryanslar,

$$\begin{aligned} \gamma_X(h) &= \text{Cov}(X_t, X_{t+h}) = \text{Cov}(X_t, \sum_{i=1}^p \alpha_i X_{t-i+h} + e_{t+h}) \\ &= \sum_{i=1}^p \alpha_i \text{Cov}(X_t, X_{t-i+h}) = \sum_{i=1}^p \alpha_i \gamma_X(h-i) \end{aligned} \quad (2.19)$$

olarak bulunur. Sonuç olarak YW denklemleri

$$\gamma_X(0) = \alpha_1 \gamma_X(1) + \alpha_2 \gamma_X(2) + \dots + \alpha_p \gamma_X(p) + \sigma^2 \quad (2.20)$$

$$\gamma_X(h) = \alpha_1 \gamma_X(h-1) + \alpha_2 \gamma_X(h-2) + \dots + \alpha_p \gamma_X(h-p) + \sigma^2$$

olarak elde edilir .

### 2.2.1.3 Otoregresif hareketli ortalama modelleri

Zaman serisi modellemelerinde verilerin bazen MA serilerine de AR serilerine de uygun biçimde uydurulamadığı zamanlar olmaktadır. Böyle durumlarda matematiksel olarak AR ve MA serilerinin birleşimi olarak düşünülebilecek ARMA serileri kullanılmaktadır. Model gösterimi şu şekildedir:

$$\phi(B)(X_t - \mu) = \theta(B)e_t \quad (2.21)$$

Burada p serinin AR kısmının model derecesini q da serinin MA kısmının model derecesini göstermektedir. Eğer serinin ortalaması sıfır ise bir ARMA(p,q) serisi

$$\phi(B)X_t = \theta(B)e_t \quad (2.22)$$

şeklinde gösterilebilir. Daha açık olacak şekilde ortalaması  $\mu$  olan bir ARMA(p,q) serisi  $\phi_p \neq 0$  ve  $\theta_q \neq 0$  olmak üzere

$$X_t - \mu = \sum_{j=1}^p \phi_j(X_{t-j} - \mu) + e_t + \sum_{i=1}^q \theta_i e_{t-i} \quad (2.23)$$

şeklinde yazılır.

MA serileri her zaman durağandır. Karakteristik kökleri mutlak değerce 1'den küçük ise aynı zamanda *tersinirdir (invertible)*. Serinin tersinir olabilmesi demek MA serisinin AR serisine çevrilebiliyor olması demektir. Bir AR(p) serisinin durağanlığı karakteristik denklem köklerine bağlıdır ve bu seriler her zaman tersinirdirler. ARMA serilerinin durağan olması için AR kısmının durağan olması, tersinir olabilmesi için ise MA kısmının tersinir olması yeterlidir.

$X_t \sim \text{ARMA}(p,q)$  durağan ise bu seri  $\sum_{j=0}^{\infty} |\psi_j| < \infty$  olmak üzere  $X_t = \sum_{j=0}^{\infty} \psi_j e_{t-j}$  şeklinde yazılabilir ve dolayısıyla serinin otokovaryans fonksiyonu

$$\gamma(h) = \sigma^2 \sum_{j=0}^{\infty} \psi_j \psi_{j+h} \quad (2.24)$$

formunda olur. Bu eşitlikte  $\psi_j$  katsayısının belirlenmesi gerekmektedir. Bundan sonra diğer parametreler tahmin edilebilir.

#### 2.2.1.4 Bütünleşik otoregresif hareketli ortalama modelleri

Doğrusal zaman serisi modellerdeki varsayımlar serinin durağanlığı üzerine kurulduğundan, durağan olmayan zaman serilerinin durağanlaştırılarak ele alınması gerekmektedir.

Durağan olmayan seri durağanlaştırıldıktan sonra parametre tahminleri bilindik yöntemlerle yapılır. Durağan olmamayı da hesaba katan bu doğrusal modeller “ARIMA modelleri” başlığı altında incelenebilir. Bütünleşik otoregresif hareketli ortalama gösterimi fark operatörü kullanılarak kolaylıkla elde edilebilir.

### Fark operatörü:

Herhangi bir  $t$  anındaki  $X_t$  RD'nin  $t - 1$  anındaki  $X_{t-1}$  değerine dönüştüren transformasyon, *gerileme operatörü*'dür. Gerileme operatörü B şu şekilde gösterilebilir:

$$BX_t = X_{t-1} \quad (2.25)$$

Ayrıca  $k \geq 1$  için  $B^k$  operatörü ile  $B^k X_t = X_{t-k}$ 'dir.  $d$  pozitif bir tam sayı olmak üzere  $d$ . fark operatörü

$$\nabla^d X_t = (1 - B)^d X_t \quad (2.26)$$

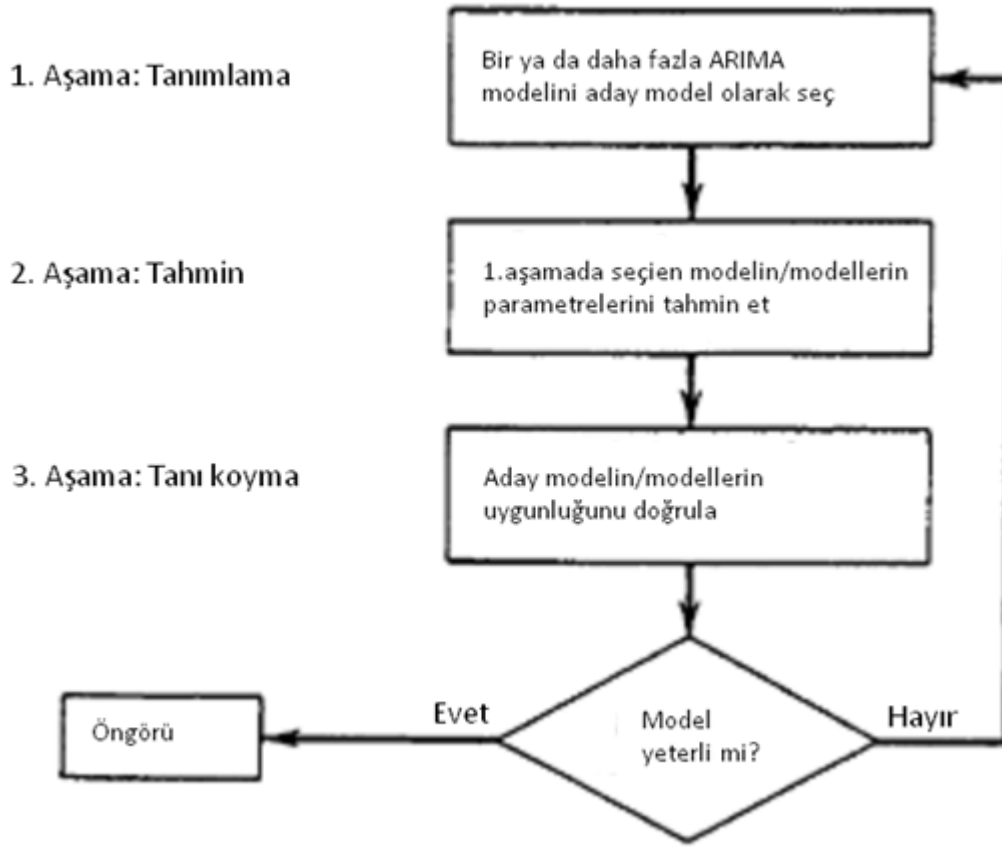
olarak verilir (Akdi, 2012).  $X_t$  durağan değilse fakat yukarıda gösterildiği gibi  $d$  kere fark aldıktan sonra durağan olabiliyorsa bu seriye  $d$ . dereceden *bütünleşik zaman serisi* denir ve  $X_t \sim I(d)$  ile gösterilebilir. AR(p) ve MA(q) özellikte  $d$ . dereceden bütünleşik bir zaman serisi ARIMA(p,d,q) ile gösterilir.

### **2.2.2 Modelleme**

Doğrusal zaman serilerinde modelleme çeşitli adımlar silsilesi ile yapılır. Modellemede genel olarak

- Modelin tanımlanması
- Tahmin
- Tanı koyma

adımları takip edilir. Yöntemin adımları Şekil 2.2.'de (Pankratz, 1983) özetlenmiştir.



Şekil 2.2. Box-Jenkins yönteminin adımları

Bu yöntem silsilesini ilk defa kurallaştıran George E. P. Box ve Gwilym M. Jenkins'dir, bu nedenle yöntem Box-Jenkins yöntemi olarak da adlandırılmaktadır (Pankratz, 1983). Doğrusal zaman serilerinde modellemenin 3 adımı çeşitli alt prosedürler içerir, bu adımlar aşağıda belirtilen şekilde ele alınabilir.

#### Modelin tanımlanması:

*Modelin tanımlanması* için kullanılan en önemli araçlar örneklem otokorelasyon ve kısmi otokorelasyon fonksiyonudur. *Model seçimini* sağlayan bir yöntem bilgi kriterleridir. Regresyon yöntemleri de model belirlemede ve model seçiminde kullanılmaktadır (Akdi, 2012).



Tahmin:

Zaman serileri analizinde *parametre tahmininde* çok çeşitli yöntemler kullanılmaktadır. EKK uygulaması kolay olduğundan ve ayrıca tahmin edicilerinin (t.e.'lerinin) özellikleri nedeniyle tercih edilmektedir.

AR modelleri bilindik doğrusal regresyon modellerine çok benzemektedir. Bu nedenle parametrelerin EKK yöntemi ile tahmin edilebileceği düşünülebilir. Ne var ki zaman serileri verileri EKK'nın varsayımlarının bir ya da birkaçını genelde sağlamaz.

ZS'lerde özellikle modeli oluşturan değişkenler arasındaki bağımlılık göz ardı edilemeyecek kadar fazladır. Bu da RD'lerin bağımsız oldukları varsayımının sağlanmasına engel olur. Bu durumda ZS'ler için bahsedilen birçok özellik yerine getirilemez. Mesela zayıf büyük sayılar yasası gereğince  $\bar{X}_n = \sum_{t=1}^n \frac{X_t}{n} \xrightarrow{P} \mu$  olduğunu söylenemez. Bu nedenle EKK yöntemi zaman serileri model parametrelerini tahmin etmek için her ne kadar uygun gözükme de, bazı ek koşulların sağlanması halinde EKK t.e.'lerin parametreleri birçok yönden oldukça iyi tahmin ettiği kanıtlanmıştır.

Durağan ZS'lerde  $n \rightarrow \infty$  için  $\bar{X}_n = \sum_{t=1}^n \frac{X_t}{n} \xrightarrow{P} \mu$  olduğu ve  $\hat{\gamma}(h) \xrightarrow{P} \gamma(h)$  olduğu gösterilebilmektedir. Bu da  $\hat{\rho}(h) \xrightarrow{P} \rho(h)$  sağlanması anlamına gelir. Durağan ZS'lerde t.e.'ler tutarlıdır.

$X_1, X_2, X_3, \dots, X_n$  bağımsız RD'lerin bir dizisi ve  $E(X_t) = \mu_t$  ve  $Var(X_t) = \sigma_t^2$  ise, *Liapounov koşulunun* sağlanması ile MLT'nin geçerli olduğu gösterilebilmektedir. Diğer bir deyişle,  $\delta > 0$  ve  $s_n^2 = Var(S_n) = \sum_{t=1}^n \sigma_t^2$  olmak üzere  $n \rightarrow \infty$  için

$$s_n^{-(2+\delta)} \sum_{t=1}^n E[[X_t - \mu_t]^{(2+\delta)}] \rightarrow 0 \quad (2.27)$$

ise MLT geçerlidir. Böylelikle  $n \rightarrow \infty$  için

$$\frac{S_n - E(S_n)}{\text{Var}(S_n)^{1/2}} \xrightarrow{D} N(0,1) \quad (2.28)$$

'dir. Ayrıca

1.  $X_t \sim IID(0, \sigma_x^2)$  ve  $e_t \sim IID(0, \sigma^2)$  ve  $X_t$  ve  $e_t$  bağımsız için ve
2.  $X_t \approx t$

durumlarında da MLT'nin geçerli olduğu gösterilebilir.

$$3. X_t = Y_{t-1}$$

durumunda ise  $Z_t = X_t e_t$  RD dizisinde okor sıfır olmasına rağmen bağımsız olmadığı görülmektedir. Bu durumda MLT uygulanamaz. Bu model için eğer seri durağan ise yani  $|\beta| < 1$  ise

- i.  $\delta > 0$  ve  $n \rightarrow \infty$  için  $s_n^{-(2+\delta)} \sum_{t=1}^n E|Z_t|^{(2+\delta)} \rightarrow 0$
- ii.  $\frac{V_n^2}{s_n^2} \xrightarrow{P} 1$

koşulları sağlandığında MLT geçerlidir. Yani

$$\frac{S_n - E(S_n)}{\text{Var}(S_n)^{1/2}} \xrightarrow{D} N(0,1) \quad (2.29)$$

dir. Burada

$$S_n = \sum_{t=1}^n Z_t \quad (2.30)$$

$$s_n^2 = \text{Var}(S_n) = \sum_{t=1}^n \text{Var}(Z_t) \quad (2.31)$$

$$V_n^2 = \sum_{t=1}^n E(Z_t | Z_1, Z_2, \dots, Z_{t-1}) \quad (2.32)$$

'dir. Eğer seri durağan değil ise t.e.'nin asimptotik dağılımları Dickey-Fuller dağılımı olmaktadır (Akdi, 2012).

Zaman serilerinin tahmini için EKK dışında birçok yöntem bulunmaktadır. Bunların başlıcaları Yule Walker, Burg algoritması, inovasyon algoritması, Hannan–Rissanen algoritması'dır. Bunlardan ilk ikisi saf AR modelleri için kullanılırken, diğer algoritmalar ARMA modelleri için kullanılmaktadır. Parametre tahminleri için bunların dışında maksimum olabilirlik yöntemleri de kullanılmaktadır (Brockwell ve Davis, 2002).

#### Tanı koyma:

Tahminler yapıldıktan sonra modelin her açıdan sorunsuz olduğunu sınamak amacı ile *tanı koyma* işlemi gerçekleştirilir. Bunun için hata terimleri arasında ilişki araştıran portmanteau testleri kullanılabilir. Bunun yanında model için varsayımların sağlanıp sağlanmadığı çeşitli testlerle sınanır.

*Modelin uyumun iyiliğini ölçmek* için istatistikte başka alanlarda kullanılan çok çeşitli yöntemler uygulanabilir. Bunlardan biri  $R^2$  istatistiğinin gözden geçirilmesi olabilir.

Uygun bir model olduğuna kanaat getirilen model *kestirim* amacı ile kullanılabilir. Kestirimler gelecek 1 ya da en fazla 2 dönem için güvenilirdir. Bu nedenle kestirimler yinelemeli bir şekilde yapılarak ilerlenir.

Box-Jenkins yönteminde bu adımlarla işlemler gerçekleştirilmeden önce *durağanlık araştırma* ve gerekiyorsa *durağanlaştırma yöntemleri* ile *mevsimsel etkilerden arındırma* yöntemleri de kullanılır.

Bu modeller detaylı bir şekilde incelenmişlerdir ve günümüzde de birçok ekonomik ve finansal problemin analizinde kullanılmaktadırlar.

Doğrusal zaman serileri modelleri, son derece sağlam bir alt yapıya sahiptir. Finansal zaman serileri başta olmak üzere çeşitli mühendislik problemlerinin ele alınmasında kullanılan araçlardır. Ayrıca doğrusal zaman serileri diğer finansal zaman serileri modelleri için de temel oluşturmaktadır.

Finansal zaman serilerinin asimetrik dalgalanma, kalın kuyruk gibi özelliklerini de ele alabilmek için, Box-Jenkins yöntemi ile kullanılan doğrusal zaman serileri modelleri, farklı bazı modellerle birlikte kullanılır. Böylelikle Box-Jenkins yönteminin etkinliği artırılmış olur.

Finansal zaman serilerinin özelliklerinden en önde geleni bu serilerin oynaklığa sahip olmasıdır. Bu özellik finansal zaman serilerinde “oynaklık modellemesi” konu başlığı altında incelenecek kadar detaylı bir çalışma alanı oluşturmuştur.

### 2.3 Oynaklık Modelleri

Kesit verileri ile yapılan çalışmalarda göze çarpan sabit olmayan varyans durumu zaman serilerinde de görülebilmektedir. Çalışmalarda sıkça kullanılan varlık değerlerinin logu fark alındığında bir beyaz gürültüye benzediği görülmektedir. Yani  $\{Y_t\}$  ile gösterilen finansal varlığın log serisi için muhtemelen bir ARIMA(0,1,0) modeli elde edilecektir.

$$\Delta Y_t = X_t = \varepsilon_t, \varepsilon_t \sim N(0,1) \quad \text{IID} \quad (2.33)$$

Bu eşitlik normal beyaz gürültü davranışı gösterir ve dolayısıyla rassal yürüyüş hipotezine uyar. Birinci dereceden yaklaşım için doğru bir seçenek olarak görülse de detaylı bir çözümleme verinin bazı özellikleri için bu yaklaşımın yeterli olmayacağını göstermektedir (Chan, 2002).

Günümüzde finansal kıymet değerleri her ne kadar anlık olarak hesaplanıp kaydedilebilse de modellemeler hala başlangıç, en yüksek, en düşük ve kapanış değerleri üzerinden yapılabilmektedir. Ayrıca kapanış değeri ile diğer günün başlangıç değeri arasındaki zaman farkı ve arz talep dinamiklerinin bu zaman aralığında değişebiliyor olması da modellemenin sürekli zaman boyutunda ele alınmasını engelleyebilmektedir. Bu durumda kesikli zaman uzayı varsayımı altında kurulan modeller tercih edilmektedir (Tsay, 2010).

Finansal zaman serileri ile ilgili *gelenekleştirilmiş gerçekler* şu şekilde sıralanabilir (Enders, 2015):

- Çoğu finansal zaman serisi genellikle yukarı yönlü bir trende sahiptir.

- Finansal zaman serileri *rassal yürüyüş* tipi bir hareket sergiler, rassal yürüyüş ise durağan değildir.
- Belli bir zamanda meydana gelen şokun etkisi hızlı bir şekilde kaybolmayabilir.
- Finansal zaman serileri asimetrik dalgalanmalar gerçekleştirirler.
- Dönem dönem daha fazla salınım yapan finansal zaman serilerinin, belli bir dönem geçtikten sonra yine eski salınım hareketlerine benzer hareketler yaptığı gözlemlenmektedir.

Bu tür özellikleri finansal zaman serilerinin durağanlık özelliklerini yitirmesine ya da modelin bazı parametrelerinin zamana bağlı olması ile sonuçlanabilir. Finansal zaman serilerinin bu gibi özellikleri, bilindik ekonometrik modellerin varsayımlarının sağlanmamasına neden olur.

Diğer taraftan *risk* ekonomide önemli etkilere sahip bir faktördür. Makroekonomide tahmin edilebilirlik kaynakların daha verimli kullanılmasını sağlamaktadır. Yatırım kararları ile risk arasında negatif bir korelasyon olduğu gösterilmiştir (Aizenman ve Marion, 1998; Wyk ve Lal, 2008). Dolayısıyla önde gelen tüm makroekonomik değişkenler bu bilgiden doğrudan etkilenmektedir.

Finansal zaman serilerinde risk ölçümlerinden biri oynaklıktır, diğer bir deyişle koşullu varyanstır (Lobato, Lobato ve Suarez, 2007). Oynaklık bu yüzden finans sektöründe çok kullanılan bir araçtır. Oynaklık bilgisinin finansta şu gibi kullanım alanları bulunmaktadır (Tsay, 2010):

- Bir finansal değer riskini ölçmek için kullanılabilir dolayısıyla fiyatını etkiler ve varlık tahsisinde önemli rol oynar.
- Oynaklık bilgisi belli bir değişkenin kestiriminde daha etkin bir çalışma yapılmasını sağlar.

Engle (1982) genel olarak ARCH süreci ile elde edilecek varyansların daha düşük olacağını belirtmiştir. Fakat yine de ARCH süreci ile elde edilecek güven aralıkları dönemselsel olarak ARMA sürecinin ürettiği güven aralıklarına göre daha büyük değer alabilmektedirler çünkü

volatilitenin yüksek olduğu dönemlerde güven aralıkları da daha geniş olacaktır (Ruppert, 2004).

Oynaklık modellerinin en çok bilineni ve deneysel çalışmalarda en fazla kullanılan ARCH/GARCH tipi modeller olarak bilinen ARCH, GARCH ve bunların türevi olan modellerdir (Chan, 2002). ARCH modeli 1982 yılında Engle, GARCH modeli ise 1986 yılında Bollerslev tarafından tanımlanmıştır. ARCH ve GARCH modelleri finansal zaman serilerinin yukarıda sayılan özelliklerinden bir kısmını ele alabilmektedir. Bu modellerin türevleri olan modellerle finansal zaman serilerinin yukarıda bahsedilen diğer özellikleri de ele alınabilmektedir (Jondeau, Poon ve Rockinger, 2000).

Finansal zaman serilerinde yapılan çalışmalar sonucunda uygun model bulunmasına rağmen, kalıntı verilerinin bağımsız olmadığı görülebilir. ARCH/GARCH, bu bağımlılık yapısını modelleyerek zaman serileri ile ilgili daha fazla bilgi edinilmesini sağlar.

Bir ARMA kalıntılarından oluşturulan veri setindeki verilerin ilişkisiz olmasına rağmen,  $\{a_t^2\}$  grafikleri incelenirse belli örüntülerin bulunduğu görülebilir. Bu durumda veri karesi okorf, kokorf ve getiri mutlak değer okorf grafiklerinde de örüntüler olacaktır. Bu örüntüler verilerin bağımsız olmadığı gösterir (Cryer ve Chan, 2008). Tüm bu nedenler verinin biraz daha irdelenmesi için gerekli motivasyonu oluşturmaktadır.

### 2.3.1 ARCH/GARCH modeli yapısı

#### ARCH(m) modeli:

ARCH/GARCH modelleri verinin ikinci momenti ile ilgili bir çalışma olduğundan, verinin öncelikle ortalama yapısının belirlenmesi gerekir. Bu nedenle ARCH çalışmalarında ortalama denklemi çoğu zaman bir ARMA modeli ile ele alınır. Buna göre bir ARCH(m) modelinde hata terimlerinde bağımlılık yapısı,  $F_{t-1}$ ,  $t - 1$  zamanına kadar olan bilgi kümesini gösteren  $\sigma$ -cebiri olmak üzere,

$$\mu_t = E(r_t | F_{t-1}) \quad (2.34)$$

ve

$$\sigma_t^2 = \text{Var}(r_t|F_{t-1}) = E[(r_t - \mu_t)^2|F_{t-1}] \quad (2.35)$$

olarak tanımlandığında,  $r_t$  bir ARMA modeli yapısına uygun olarak şu şekilde ifade edilebilir:  $k, p, q$  negatif olmayan doğal sayı ve  $x_{it}$  açıklayıcı değişkenler iken,

$$r_t = \mu_t + a_t \quad (2.36)$$

$$\mu_t = \sum_{i=1}^p \varphi_i y_{t-i} - \sum_{i=1}^q \theta_i a_{t-i} \quad (2.37)$$

$$y_t = r_t - \varphi_0 - \sum_{i=1}^k \beta_i x_{it} \quad (2.38)$$

ortalama denklemini göstermektedir. ARCH/GARCH modellerinde yukarıdaki eşitliklere ek olarak aşağıdaki ifade de oynaklık yapısını ele almaktadır:

$$\sigma_t^2 = \text{Var}(r_t|F_{t-1}) = \text{Var}(a_t|F_{t-1}) \quad (2.39)$$

ARCH için  $a_t = \sigma_t \varepsilon_t$  şeklinde ifade edilmektedir. Burada  $\varepsilon_t \sim N(0,1)$ , IID'dir, fakat  $t$  dağılımıyla da model uygun sonuçlar verir (Jondeau, Poon ve Rockinger, 2000).

Oynaklık modelleri  $\sigma_t^2$  'nın değişimi ile ilgilenmektedir. ARCH modeli oynaklık bilgisini inovasyon teriminin karesinin geçmiş değerleriyle ortaya koyar. Modelin yapısı şu şekilde ifade edilebilir:

$$\sigma_t^2(\theta) = \alpha_0 + \alpha_1 a_{t-1}^2 + \alpha_2 a_{t-2}^2 + \dots + \alpha_m a_{t-m}^2 \quad (2.40)$$

Bu model için

$$E(a_t) = E[E(a_t|F_{t-1})] = E[\sigma_t E(\varepsilon_t)] = 0 \quad (2.41)$$

ve

$$\text{Var}(a_t) = E(a_t^2) = E[E(a_t^2|F_{t-1})] = E(\alpha_0 + \alpha_1 a_{t-1}^2) \quad (2.42)$$

eşitlikleri geçerli olur. Modelden anlaşılacağı gibi yakın zamanlı geçmiş büyük şoklar güncel şokun da yüksek olması beklentisini yaratmaktadır (Chan, 2002).  $a_t$  durağan bir süreç olduğundan dolayı,

$$E(a_t) = 0 \quad (2.43)$$

ve

$$\text{Var}(a_t) = \text{Var}(a_{t-1}) = E(a_{t-1}^2) \quad (2.44)$$

'dir. Böylelikle,

$$\text{Var}(a_t) = \frac{\alpha_0}{1-\alpha_1} \quad (2.45)$$

olur. Burada  $a_t$ 'nin varyansı pozitif olacağından dolayı  $\sigma^2$ 'nin sonlu olması için  $0 \leq \alpha_1 < 1$  olması gerekir. ARCH(m) modeli için ise kısıt  $\sum_{i=1}^m \alpha_i < 1$  olur (Jondeau, Poon ve Rockinger, 2000).

Burada  $a_t$  serisel olarak ilişkisiz olmasına rağmen zaman-bağımsız değildir, çünkü  $\eta_t = a_t^2 - \sigma_t^2$  olarak tanımlandığında

$$a_t^2 = \omega + \sum_{i=1}^p \alpha_i a_{t-i}^2 + \eta_t \quad (2.46)$$

olacaktır. Bu da karesi alınmış inovasyon terimleri ile oluşturulan bir AR(p) modeline karşılık gelmektedir. Böyle bir tanımlamada  $\{\eta_t\}$  ilişkisiz ve  $E_{t-1}[\eta_t] = 0$ 'dir fakat IID değildir, bu nedenle de EKK t.e.'leri etkin olmaz, fakat bununla birlikte tutarlıdır. ARCH modelleri basıklık fazlalığına sahiptirler.  $\varepsilon_t = \frac{x_t - \mu_t(\theta)}{\sigma_t}$  şeklinde ifade edilen standart inovasyonların koşullu dağılımı normal olsa da  $a_t$ 'nin koşulsuz dağılımı kalın kuyruğa sahip olacaktır.

Kuyruk yapısının davranışları hakkında anlamlı sonuçlar çıkarmak için  $a_t$ 'nin 4. momentinin sonlu olması gereklidir.  $\varepsilon_t$  normal dağıldığı varsayımı altında,



$$\begin{aligned}
E(a_t^4 | F_{t-1}) &= 3[E(\sigma_t^2 | F_{t-1})]^2 \\
&= 3(\alpha_0^2 + 2\alpha_0\alpha_1 a_{t-1}^2 + \alpha_1^2 a_{t-1}^4)
\end{aligned} \tag{2.47}$$

eşitlikleri geçerlidir. Eğer  $m_4 = E(a_t^4)$  olmak üzere  $a_t$  4. dereceden durağansa

$$m_4 = 3\alpha_0^2 \left(1 + \frac{2\alpha_1}{1-\alpha_1}\right) + 3\alpha_1^2 m_4 \tag{2.48}$$

olarak bulunur ve sonuç olarak,

$$m_4 = \frac{3\alpha_0^2(1+\alpha_1)}{(1-\alpha_1)(1-3\alpha_1^2)} \tag{2.49}$$

olur. Bu sonuç 2 önemli anlama sahiptir:

a)  $a_t$ 'nin 4. momenti pozitif olacağından  $\alpha_1$  şu koşulu da sağlar:

$$1 - 3\alpha_1^2 > 0 \text{ diğer bir gösterimle } 0 \leq \alpha_1^2 < 1/3$$

b)  $a_t$ 'nin koşulsuz basıklığı aşağıdaki gibi olacaktır:

$$\begin{aligned}
\frac{E(a_t^4)}{[Var(a_t)]^2} &= \frac{3\alpha_0^2(1+\alpha_1)}{(1-\alpha_1)(1-3\alpha_1^2)} \frac{(1-\alpha_1)^2}{\alpha_0^2} \\
&= \frac{3(1-\alpha_1^2)}{1-3\alpha_1^2} > 3
\end{aligned}$$

Böylece  $a_t$ 'nin basıklık fazlalığı pozitiftir ve  $a_t$ 'nin kuyruk dağılımı normal dağılımından daha kalındır. Yani Gausgil ARCH(1) modeli, bir Gausgil WN'ye göre daha fazla dışa düşen üretmeye meyillidir.

Öngörü:

ARCH modeli kullanılarak yapılan öngörüler yinelemeli şekildedir. Tahmin edilen parametreler kullanılarak ve  $t$  başlangıç tarihi olmak üzere bir adım sonraki  $\sigma_{t+1}^2$  için öngörü

$$\sigma_t^2(1) = \hat{\omega} + \hat{\alpha}_1 \hat{a}_t^2 + \dots + \hat{\alpha}_p \hat{a}_{t+1-p}^2 = \hat{\omega} + \sum_{i=1}^p \hat{\alpha}_i \hat{a}_{t+1-i}^2 \quad (2.50)$$

şeklinde gösterilebilir.  $k$  adım sonraki öngörü ise

$$\sigma_t^2(k-i) = \hat{a}_{t+k-i}^2, k-i \leq 0 \quad (2.51)$$

olmak üzere,

$$\sigma_t^2(k) = \hat{\omega} + \hat{\alpha}_1 \sigma_t^2(k-1) + \dots + \hat{\alpha}_p \sigma_t^2(k-p) = \hat{\omega} + \sum_{i=1}^p \hat{\alpha}_i \sigma_t^2(k-i) \quad (2.52)$$

'dir.

#### GARCH(m,n) modeli:

GARCH modeli, oynaklık bilgisini inovasyon teriminin karesine ek olarak koşullu varyansın geçmiş değerleri de dikkate alır. GARCH(m,n) modelinde  $r_t$  için oynaklık denklemi,

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^m \alpha_i a_{t-i}^2 + \sum_{j=1}^n \beta_j \sigma_{t-j}^2 \quad (2.53)$$

şeklinde ele alınır. Yukarıdaki GARCH modelinde  $a_t$  doğrusal modelin inovasyonlarını göstermektedir. İnovasyon değişkeni ARCH'inki ile benzer özelliklere sahiptir.

#### Öngörü:

GARCH modellerinde öngörü ARCH modellerinde olduğu gibi yinelemeli yöntemlerle yapılır.

#### Modelin kısıtları:

Modellerde  $\alpha_0 > 0$ ,  $\alpha_i \geq 0$ ,  $\beta_j \geq 0 \forall i, j > 0$  ve ayrıca GARCH için  $\sum_{i=1}^{\max(m,s)} (\alpha_i + \beta_i) < 1$  koşulları sağlanmalıdır. Bunlara ek olarak  $\sum_{i=1}^p \alpha_i + \sum_{i=1}^q \beta_j < 1$  koşulu da  $a_t$ 'nin kovaryans durağan olmasını sağlar. Bu koşul  $a_t$ 'nin güçlü durağan olması için yeterli fakat gerekli olmayan bir koşuldur. GARCH (1,1) modeli

$$\sigma_t^2 = \alpha_0 + \beta_1 \sigma_{t-1}^2 + \alpha_1 a_{t-1}^2 \quad (2.54)$$

$$\sigma_t^2 = \alpha_0 + \sigma_{t-1}^2 (\alpha_1 \varepsilon_{t-1}^2 + \beta_1) = \alpha_0 (1 + \sum_{\tau=1}^{\infty} \prod_{k=1}^{\tau} (\alpha_1 \varepsilon_{t-k}^2 + \beta_1))$$

şeklinde yazıldığında,  $a_t$  sürecinin güçlü durağan olması için yukardaki eşitliğin sonlu bir limite yakınsaması gerekir. Bunun için

$$E(\log(\beta_1 + \alpha_1 e_{t-1}^2)) < 0 \quad (2.55)$$

olmalıdır. Buradan,

$$E(\log(\beta_1 + \alpha_1 e_{t-1}^2)) \leq \log(E(\beta_1 + \alpha_1 e_{t-1}^2)) = \log(\alpha_1 + \beta_1) \quad (2.56)$$

böylelikle  $\beta_1 + \alpha_1 = 1$  olsa dahi süreç hala güçlü durağandır.

### 2.3.2 ARCH/GARCH modelleme

#### ARCH etkisinin önemli olup olmadığının araştırılması:

$\{a_t^2\}$  ile süreçte koşullu değişen varyans olup olmadığı test edilebilir. Koşullu varyans ARCH etkisine karşılık gelmektedir. ARCH etkisinin olup olmadığını anlamak için kullanılabilecek testlerden bir tanesi Ljung-Box istatistiği  $Q(m)$ 'in  $a_t^2$  serisine uygulanmasıdır. Test için boş hipotez  $a_t^2$  serisinin okorfunun ilk  $m$  gecikmesi 0'a eşittir şeklindedir. Burada Ljung-Box istatistiğinin kritik değerlere sahip olması ARCH etkisinin varlığına işaret eder.

İkinci test bilindik F testinin  $a_t^2$  serisine uygulanmasıdır. Şöyle ki,

$$a_t^2 = \alpha_0 + \alpha_1 a_{t-1}^2 + \dots + \alpha_m a_{t-m}^2 + e_t, t = m + 1, \dots, T \quad (2.57)$$

ifadesi yazıldığında, burada  $e_t$  hata terimi iken  $m \in \mathbb{N}$  ve  $T$  de örnek sayısıdır. Boş hipotez  $H_0 = \alpha_1 = \dots = \alpha_m = 0$  şeklindedir. Ayrıca,

$$SSR_0 = \sum_{t=m+1}^T (a_t^2 - \bar{a})^2, \quad (2.58)$$

$$\bar{\omega} = \sum_{t=1}^T \frac{a_t^2}{T},$$

$$SSR_1 = \sum_{t=m+1}^T \hat{e}_t^2$$

olarak tanımlanmıştır. Böylelikle

$$F = \frac{\frac{SSR_0 - SSR_1}{m}}{\frac{SSR_1}{T - 2m - 1}} \quad (2.59)$$

'dir. Bu istatistik boş hipotez altında asimptotik olarak  $m$  serbestlik derecesi ile ki-kare dağılmaktadır. Karar kuralı  $F > \chi_m^2(\alpha)$  ise boş hipotezi reddetmek şeklindedir. Burada  $\chi_m^2(\alpha)$ ,  $\chi_m^2$ 'nin üst  $100(1 - \alpha)$ . yüzdeleri değeridir.

#### Varyans modeli sıra belirlenmesi:

ARCH etkisinin önemli olduğunun tespitinden sonra  $a_t^2$  serisinin koku ile ARCH sırası belirlenebilir. Bunun için aşağıdaki ARCH modeli kullanılması gerekir.

$$\sigma_t^2 = \alpha_0 + \alpha_1 a_{t-1}^2 + \alpha_2 a_{t-2}^2 + \dots + \alpha_m a_{t-m}^2 \quad (2.60)$$

Bu modelde gözlenen bir veri seti için  $a_t^2$ ,  $\sigma_t^2$  'nin sapmasız bir tahmin edicisidir.  $a_t^2, a_{t-1}^2, \dots, a_{t-m}^2$  ile doğrusal bir ilişkiye sahip olacağından  $m$ . dereceden kendinden bağımlı bir özellik göstereceği beklenir. Burada  $a_t^2$  genellikle  $\sigma_t^2$ 'nin etkin bir tahmin edicisi değildir fakat yine de sıra belirlemede bilgilendirici olacaktır.

GARCH yöntemi için sıra belirlenmesinde bu yol izlenmez. Parametre sayısının zaten az olacağı da göz önüne alınarak düşük sıralı modellerden başlayıp model seçim kriterleri ile en iyi model bulunabilir.

#### Tahmin:

ARCH ve GARCH parametre tahminlerinde çeşitli yöntemler kullanılmaktadır. En fazla kullanılan yöntem olabilirlik temelli tahmin yöntemleridir. GARCH(p,q) model parametrelerinin tahmini şu şekilde yapılmaktadır:

$$\{x_1, x_2, \dots, x_T\} \quad (2.61)$$

şeklinde bir seri tanımlandığında,

$$m = \max(p, q) \quad (2.62)$$

olmak üzere normallik varsayımı ile

$$f(a_1, \dots, a_T | \theta) = f(a_1, \dots, a_m | \theta) f(a_{m+1} | F_m) \dots f(a_{T-1} | F_{T-2}) f(a_T | F_{T-1}) \quad (2.63)$$

$$L_T(\theta | X_t, t = 1, \dots, T) = \prod_{t=m+1}^T \exp\left(-\frac{a_t^2}{2\sigma_t^2}\right) (2\pi\sigma_t^2)^{-\frac{1}{2}} f(a_1, \dots, a_m | \theta)$$

elde edilir. Burada

$$\theta = (\omega, \alpha_1, \dots, \alpha_p, \beta_1, \dots, \beta_q)' \quad (2.64)$$

olmaktadır. Olabilirlik fonksiyonunda  $f(a_1, \dots, a_m | \theta)$  teriminin etkisi az olacağından genelde düşürülür. Bu durumda koşullu-olabilirlik fonksiyonu

$$f(a_1, \dots, a_T | \theta, a_1, \dots, a_m) = \prod_{t=m+1}^T \frac{e^{-a_t^2/2\sigma_t^2}}{(2\pi\sigma_t^2)^{1/2}} \quad (2.65)$$

olur. Buradan,

$$l_t(\theta) = -\frac{\ln(2\pi)}{2} - \frac{\ln(\sigma_t^2)}{2} - \frac{a_t^2}{2\sigma_t^2} \quad (2.66)$$

$$L_T(\theta | X_t, t = 1, \dots, T) = \sum_{t=m+1}^T l_t(\theta) \quad (2.67)$$

elde edilir. Burada log-olabilirlik fonksiyonu

$$L_T(\theta | X_t, t = 1, \dots, T) = \sum_{t=m+1}^T \left( -\frac{\ln(2\pi)}{2} - \frac{\ln(\sigma_t^2)}{2} - \frac{a_t^2}{2\sigma_t^2} \right) \quad (2.68)$$

'dir ve  $\ln(2\pi)$  parametre içermediğinden

$$L_T(\theta|X_t, t = 1, \dots, T) = \sum_{t=m+1}^T \left( \frac{\ln(\sigma_t^2)}{2} + \frac{a_t^2}{2\sigma_t^2} \right) \quad (2.69)$$

olarak alınabilir. Log olabilirlik fonksiyonunun türevi eğimi (gradient) verir:

$$\frac{\partial l_t(\theta)}{\partial \theta} = -\frac{1}{2\sigma_t^2} \frac{\partial \sigma_t^2}{\partial \theta} + \frac{\varepsilon_t^2}{2\sigma_t^4} \frac{\partial \sigma_t^2}{\partial \theta} = \frac{1}{2\sigma_t^2} \frac{\partial \sigma_t^2}{\partial \theta} \left( \frac{\varepsilon_t^2}{\sigma_t^2} - 1 \right) \quad (2.70)$$

Burada

$$\frac{\partial \sigma_t^2}{\partial \alpha_0} = 1$$

$$\frac{\partial \sigma_t^2}{\partial \alpha_i} = a_{t-i}^2, i = 1, \dots, p \quad (2.71)$$

$$\frac{\partial \sigma_t^2}{\partial \beta_j} = \sigma_{t-j}^2, j = 1, \dots, q$$

'dir. İkinci dereceden türevler ile Hessian matrisi üretilir:

$$\frac{\partial^2 l_t(\theta)}{\partial \theta \partial \theta'} = -\frac{1}{2\sigma_t^2} \frac{\partial \sigma_t^2}{\partial \theta} \frac{\partial \sigma_t^2}{\partial \theta'} \frac{\varepsilon_t^2}{\sigma_t^2} + \left( \frac{\varepsilon_t^2}{\sigma_t^2} - 1 \right) \frac{\partial \sigma_t^2}{\partial \theta'} \frac{1}{2\sigma_t^2} \frac{\partial \sigma_t^2}{\partial \theta} \quad (2.72)$$

ve

$$E \left[ \frac{\partial^2 l_t(\theta)}{\partial \theta \partial \theta'} \right] = -E \left[ \frac{1}{2\sigma_t^2} \frac{\partial \sigma_t^2}{\partial \theta} \frac{\partial \sigma_t^2}{\partial \theta'} \right] \quad (2.73)$$

ve

$$\xi_t = (1, \xi_{t-1}, \dots, \xi_{t-p}, \sigma_{t-1}^2, \dots, \sigma_{t-q}^2)' \quad (2.74)$$

şeklinde tanımlandığında,

$$\sigma_t^2 = \xi_t' \theta \quad (2.75)$$

eşitliği yazılabileceğinden

$$E \left[ \frac{\partial^2 l_t(\theta)}{\partial \theta \partial \theta'} \right] = -E \left[ \frac{\xi_t \xi_t'}{2\sigma_t^2} \right] \quad (2.76)$$

elde edilir. En çok olabilirlik tahmin edicisinin kovaryans matrisi bilgi matrisinin tersine denk gelmektedir, bu da negatif Hessian matris beklenen değeridir.

$$I(\theta) = -E \left[ \frac{\partial^2 l_t(\theta)}{\partial \theta \partial \theta'} \right] = E \left[ \frac{\xi_t \xi_t'}{2\sigma_t^2} \right] \quad (2.77)$$

Bu ifade ise

$$\hat{I}(\theta) = \sum_{t=1}^{2T} \frac{\hat{\xi}_t' \hat{\xi}_t}{2\hat{\sigma}_t^2} \quad (2.78)$$

eşitliği ile tahmin edilebilir. Standartlaştırılmış inovasyonların sonlu 4. momentlerinin olması en çok olabilirlik tahmin edicisinin tutarlı olmasını sağlar, asimptotik dağılım ise şu şekilde olur:

$$T^{1/2}(\hat{\theta} - \theta) \Rightarrow N(0, I(\theta)^{-1}) \quad (2.79)$$

Ayrıca  $X_t$  dışsal açıklayıcı değişkenlerin bir kümesi ve  $\delta$  ortalama denkleminin bilinmeyen parametre kümesi olmak üzere,

$$x_t = X_t \delta + \varepsilon_t$$

$$\varepsilon_t = \sigma_t z_t \quad (2.80)$$

$$\sigma_t^2 = \omega + \sum_{i=1}^p \alpha_i \varepsilon_{t-i}^2 + \sum_{j=1}^q \beta_j \sigma_{t-j}^2$$

denklemleri için tahmin  $\delta$  ve  $\theta$  için eşzamanlı olarak yapılır. Fakat ikinci regresyon modelinin log-olabilirlik fonksiyonunun dereceden türevi şu şekildedir:

$$E \left[ \frac{\partial^2 l_t(\theta, \delta)}{\partial \theta \partial \theta'} \right] = 0 \quad (2.81)$$

Sonuç olarak bilgi matrisi blok köşegenel tiptedir ve parametreler ayrı ayrı hesaplanabilir.

Pratikte ise öncelikle inovasyon terimleri tahmin edilip ilk değerler verilir. Koşullu varyans hesaplanır. Sonrasında çeşitli yöntemlerle parametreler tahmin edilebilmektedir.

### Model doğrulaması:

Model doğrulaması hata terimleri üzerinden yapılır. Standartlaştırılmış hata terimleri  $\tilde{a}_t = \frac{a_t}{\sigma_t}$  serisinin IID olması ortalama modelinin sorunsuz olması için bir koşuldur. Ortalama denklemi için  $\tilde{a}_t$  ve varyans denklemi için  $\tilde{a}_t^2$  göz önüne alınarak aranabilir. Her iki test için de Ljung-Box istatistiği kullanılabilir. Burada Ljung-Box istatistiğinin kritik değerlere sahip olmaması modelin uygunluğuna işaret eder.  $\tilde{a}_t^2$ 'nin çarpıklık ve basıklık, QQ grafikleri dağılım varsayımlarının sağlanıp sağlanmadığını incelemek için kullanılabilir.

Sonuç olarak, ARCH/GARCH yöntemleri döviz parite, hisse senedi tahminleri gibi yüksek oynaklık içeren her türlü finansal varlıkların oynaklık modellemesinde günümüzde de kullanılmaktadır. Finansal zaman serileri analizinde ARCH/GARCH tipi modeller, finansal zaman serilerinde sıkça görülen şu özellikleri de hesaba katarak modelleme yapılmasına olanak sağlar:

- Kalın kuyruk özelliği
- Oynaklık kümelenmeleri
- Kaldıraç etkileri

### **2.3.3 ARCH/GARCH modellerinin zayıf olduğu noktalar**

- Pozitif ve negatif şokların seri üzerindeki etkilerinin aynı olacağı (simetri) varsayılır.
- ARCH modellerinin parametrelerle ilgili bazı kısıtları bulunmaktadır. Bu dezavantaj GARCH modeli ile bir nebze olsun giderilebilmiştir.
- Değişkenlikleri neden sonuç ilişkisine bağlamaz, sadece mekanik bir şekilde oynaklık modellemesi yapar.
- Büyük şoklara geç cevap vermesinden dolayı şoklardan sonraki dönemlerde genellikle oynaklığı olduğundan fazla tahmin eder.



ARCH-GARCH türevi modeller:

ARCH-GARCH modelleri tanımlandıktan bir süre sonra farklı türevleri geliştirilmeye başlanmıştır. Her bir türev model, ARCH-GARCH modellerinin eksik yanlarını azaltması amacı ile geliştirilmiştir. Literatürde geniş yer bulan ve pratikte çokça kullanılan bu modellerin bazıları şunlardır:

- IGARCH
- GARCH-M
- GJR-GARCH
- PARCH
- EGARCH
- TGARCH

### 3 MAKİNE ÖĞRENMESİ

Çalışmaya adını vermiş olan problem finansal bir problemdir. Bu bölümdeki çözüm yöntemleri ise daha çok bilgisayar bilimi altında incelenen makine öğrenmesi teknikleridir.

Makine öğrenmesi, performansı ve tahmin doğruluğunu artırmak için deneyimi kullanan hesaplama yöntemleri olarak tanımlanabilir. Burada deneyim öğreniciye erişilebilir olan geçmiş bilgiye karşılık gelmektedir. Bilgi genellikle analiz için toplanmış elektronik veri formundadır. Veri başarılı tahminler için kritik öneme sahiptir (Mohri, Rostamizadeh ve Talwalkar, 2012). Çoğunlukla istatistik teorisinin ve istatistiksel tekniklerinin kullanıldığı fakat aynı zamanda kontrol teorisi, felsefe, psikoloji ve nörobiyoloji bilgilerinden de yararlanılarak farklı yapıdaki problemleri optimize bir şekilde çözüme yolları sunan disiplinler arası bir bilim dalıdır (Mitchell, 1997).

Makine öğrenmesi veri güdümlü (data driven) teknikleri içerir; yani modelleme, kurulan model üzerine çok az bir varsayımla etkili bir şekilde yapılabilir. Bu yüzden makine öğrenmesinde problem çözme teknikleri öncelikle veri odaklı bir üst sınıflandırmaya tabi tutulurlar:

- Gözetimli öğrenme
- Gözetimsiz öğrenme
- Pekiştirmeli öğrenme (daha çok robotik alanında kullanılır)

Bu yöntemler bir problemin çözümünde tek başlarına bir araç olarak kullanılacakları gibi birlikte kullanılabilirlerinde daha etkili çözümler elde edilir. Aslında makine öğrenmesinin gücü de buradan gelmektedir: Bilinen birçok istatistiksel tekniği bir arada kullanarak çözmesi zor olan sorunların çözümünü elde etmek. Yukarıda bahsedilen bu 3 öğrenme tipi birlikte kullanıldıklarında farklı isimlerle adlandırılmaktadırlar.

Kanımızca makine öğrenmesi adı ile genellenen bu istatistiksel tekniklere zamanla yenileri de eklenecektir. Çeşitli mühendislik dallarında yıllardır kullanılan durum uzay modellemelerinin de zamanla makine öğrenmesi kitaplarında bir konu olarak görülebileceğini düşünmekteyiz.

### 3.1 Saklı Markov Modelleri

#### 3.1.1 Markov modelleri ve saklı Markov modelleri

Saklı Markov modelleri (SMM), bilgisayar bilimi altında incelenmesine rağmen, bu modeller farklı bilim dallarında farklı problemlere uygulanabilmektedir. Bu modeller, bilgisayar bilimi dışında elektronik mühendisliği, genetik mühendisliği, sistem mühendisliği problemlerinin çözümünde kullanılmaktadır. Bilgisayar mühendisliğinde ses, karakter ve örüntü tanıma problemlerinde, eposta sunucularında zararlı e-postaların ayıklanması, medikal tanılama, müşteri sınıflandırma problemlerinde, kredi kartı hırsızlıklarının anlık olarak anlaşılmasında, genetik mühendisliğinde DNA ve protein yapılarını tanımlama amaçlı; elektronik mühendisliğinde ise sinyal işleme prosedüründe ve hatta hava tahmini problemlerinin çözümünde başarılı örnekleri bulunmaktadır.

Saklı Markov modelleri, makine öğrenmesi yöntemlerinden bir tanesidir. SMM'de durumlar gözlenemez varsayımı kabul edilmektedir. Fakat modele göre gözlenemez her bir durum oluştuğunda gözlenebilir bir gözlem ortaya çıkmaktadır. Bu gözlenebilir gözlemlerin dizi bilgisi ile gözlenemez durumların istatistiksel özellikleri hakkında çıkarımlar yapılabilir.

Saklı Markov modellerinin dayandığı Markov modelleri hem kullanım kolaylığı hem de uygulamalardaki başarısı ile bilim insanlarının farklı alanlardan birçok problemin çözümünde kullandıkları bir araç olmuştur. Markov modelleri gelecekte olacak bir olayın, sadece o olay yaşanmadan önceki durum tarafından belirlendiğini anlatan genel bir kavrama dayanır. Doğası gereği Markov modelleri sıra ya da zaman içeren bir yapıya sahiptir. Bu manada Markov modelleri ile zaman serileri modelleri arasındaki büyük benzerlik dikkat çeker. Aslında AR modeli Markov gösterimini kullanmaktadır. Buna rağmen iki model arasında istatistiksel farklar da bulunmaktadır (Anderson, 1976).

Basit Markov modelin olasılıksal ağ diyagramı Şekil 3.1.'te görülmektedir:



Şekil 3.1. Markov modelleri

Bir Markov modelinde, durum kümesi  $\mathcal{S} = \{S_1, S_2, \dots, S_N\}$ , başlangıç olasılıkları  $1 \leq i \leq N$  için  $\pi = [\pi_i] = [P(q_1 = S_i)]$  ve durum dizisi,  $\mathcal{Q} = \{q_1, q_2, \dots, q_T\}$  olmak üzere belli bir olayın olma olasılığı geçmiş dönem bilgilerini hesaba katarak şu şekilde gösterilir (Alpaydın, 2011):

$$P(q_{t+1} = S_j | q_t = S_i, q_{t-1} = S_k, \dots) \quad (3.1)$$

Olasılıksal ağ diyagramında görülen 1. derece Markov modeli için olasılık ise aşağıdaki gibi bulunur:

$$P(q_{t+1} = S_j | q_t = S_i) \quad (3.2)$$

Aynı örnek üzerinden bu dizinin ortaya çıkış olasılığı Markov modeli varsayımları altında,

$$P(O = Q | A, \pi) = P(q_{t+1} = S_j, \dots, q_1 = S_m) \quad (3.3)$$

$$= P(q_{t+1} = S_j | q_t = S_i) P(q_t = S_i | q_{t-1} = S_k) \dots P(q_2 = S_l | q_1 = S_m) P(q_1 = S_m)$$

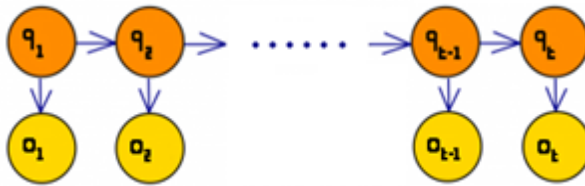
şeklinde gösterilir. Bu gösterimde  $A$  bir olaydan diğerine geçiş olasılıkları,  $\pi$  ise durumların başlangıç olasılıkları olarak tanımlanmıştır.  $\pi$  ve  $A$  bilindiğinde  $T$  uzunluğunda diziler üretilebilir. Elde birçok dizi varken  $\pi$  ve  $A$  kolaylıkla tahmin edilebilir.  $O$  ise gözlem dizisine denk gelmektedir.

Markov modelleri teoride ve pratikte sağlam matematiksel alt yapısı ve basitliği ile çok kullanılır bir model olarak yerini almıştır. Bazı problemlerin çözümünde ise Markov modelleri genişletilerek daha farklı soruların da cevapları bulunabilir.

Gözlemler her zaman olayların doğrudan kendisi üzerinden yapılamaz, zira buna neden olan fiziksel sebepler olabilir. En basitinden, gözlem sırasında ortaya çıkan ölçüm hataları bile kendi içinde bir sürece bağlı olarak hareket edebilirler. Bu gözlemlerden oluşturulan değerlerin tamamı hem gerçek olayın sonuç değerini, hem de ölçüm hatalarını oluşturan süreci barındıracaktır. Burada gözlenen ya da ölçülen değişken altta yatan başka bir süreç tarafından belli bir olasılıkla belirlenmektedir. Buna iyi bir örnek oynaklık modelleri konusundan verilebilir. Bu modellerde üstü kapalı olarak oynaklığın doğrudan gözlemlenemediği ve altta yatan başka bir süreç tarafından belirlendiği varsayılmaktadır. Hamilton'un (1989) Markov switching yaklaşımında, modeli farklı zamanlarda yöneten farklı süreçler olduğu ve bu süreç geçişlerinin doğrudan birinden diğerine gözlemlenemediği ve "durumlara bağlı ve olasılıksal bir biçimde" belirlendiği varsayımı vardır. SMM altta yatan bu süreci içermektedir.

SMM'nin farklı bilim dallarında başarılı örnekleri bulunmaktadır, özellikle konuşma ve yazı tanıma, gen ve protein tahmini gibi problemlerin çözümünde başarı ile kullanılmaktadır.

SMM'de durumlar gözlenemez fakat oluşan her durumda belli bir olasılıkla bir gözlem elde edilecektir (Alpaydın, 2011). Gözlemler simge olarak ifade edilir. SMM'nin olasılıksal ağ modeli Şekil 3.2.'te gösterilmiştir:



Şekil 3.2. Saklı Markov modelleri

1. derece SMM'nin elemanlarını Rabiner (1989) şu şekilde tanımlamıştır:

$N$ , modeldeki durumların sayısıdır. Durumlar gizli olarak varsayılmalarına rağmen uygulamalarda fiziksel öneme sahiptirler. Her bir durum  $\{1, 2, \dots, N\}$  kümesinin elemanları ile adlandırılır ve  $t$  zamanında  $q_t$  ile gösterilir.

$M$  her bir durum için ortaya çıkan farklı gözlemlerin sayısıdır. Gözlem sembolleri sistemin fiziksel çıktısına denk gelmektedir. Her bir sembol  $V = \{v_1, v_2, \dots, v_M\}$  kümesinden adlandırılır.

Durum geçiş olasılık dağılımı  $A = \{a_{ij}\}$  olarak adlandırılır.  $a_{ij} = P(q_{t+1} = j | q_t = i)$  olarak ele alınır. Genel saklı Markov modellerinde herhangi bir durumdan diğerine tek adımda geçiş mümkündür böylelikle tüm  $i, j$ 'ler için  $a_{ij} > 0$  olur. Gözlem sembollerinin olasılık dağılımı  $B = \{b_j(k)\}$  olarak adlandırılır. Burada

$$b_j(k) = P[o_t = v_k | q_t = j] \quad 1 \leq k \leq M \quad (3.4)$$

$j = 1, 2, \dots, N$  durumları için sembol dağılımlarına karşılık gelmektedir. Başlangıç durum dağılımı  $\pi = \{\pi_i\}$  ise

$$\pi_i = P(q_1 = i) \quad (3.5)$$

olarak ele alınır. Kısaca modelin parametre kümesi

$$\lambda = (A, B, \pi) \quad (3.6)$$

ile gösterilir. Bu parametrelerle gözlem kümesi  $O$ , bir olasılık ölçüsü olmaktadır ve  $P(O|\lambda)$  şeklinde ele alınmaktadır. Bu tanımlamalarla, bir gözlem dizisi için bir SMM sürecinin gerçekleşme olasılığı şu şekilde ifade edilir:

$$P(O, Q|\lambda) = P(q_1)P(O_1|q_1) \prod_{k=2}^t P(q_k|q_{k-1})P(O_k|q_k) \quad (3.7)$$

Buradan anlaşılmaktadır ki, Markov modelleri SMM'nin özel bir halidir öyle ki

$$M = N \text{ ve } b_j(O_m) = \begin{cases} 1, & j = m \\ 0 & d.d. \end{cases} \quad (3.8)$$

SMM'de  $N$  ve  $M$  diğer parametrelerce tanımlandığından, parametre kümesini gösteren  $\lambda$ ,  $\lambda = (A, B, \pi)$  olarak tanımlanmıştır.  $\lambda$  bilindiğinde  $T$  uzunluğunda istenilen sayıda dizi üretilebilir. Ne var ki SMM de parametre tahminleri Markov modellerindeki kadar kolay

değildir (Alpaydın, 2011). SMM'nin çözüm yöntemi olarak kullanıldığı 3 temel problem vardır.

### 3.1.1.1 Değerlendirme problemi (problem 1)

#### Üretim olasılığı:

Üretim olasılığı belli bir gözlem dizisi verildiğinde bu gözlem sırasının olasılığına karşılık gelir. Gözlem dizisinin oluşma olasılığı  $P(O|\lambda)$  ile gösterilir. Bu olasılığı elde etmek için şu şekilde bir yol izlenebilir:

$$\begin{aligned} P(O|Q, \lambda) &= \prod_{t=1}^T P(O_t|Q_t, \lambda) \\ &= b_{q_1}(O_1)b_{q_2}(O_2) \dots b_{q_T}(O_T) \end{aligned} \quad (3.9)$$

Ayrıca belli bir durum sırası olasılığı,

$$P(Q|\lambda) = \pi_{q_1} a_{q_1 q_2} a_{q_2 q_3} \dots a_{q_{T-1} q_T} \quad (3.10)$$

şeklinde tanımlanmaktadır.  $O$  ve  $Q$  sıralarının ortak dağılımları ise,

$$P(O, Q|\lambda) = P(O|Q, \lambda)P(Q|\lambda) \quad (3.11)$$

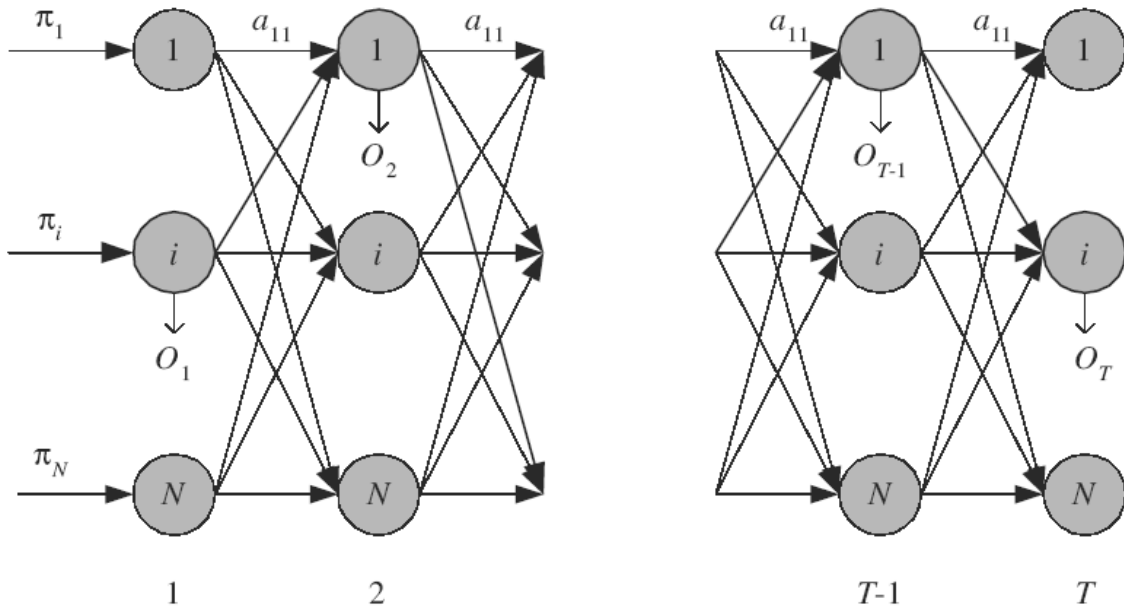
olarak elde edilir. Böylece gözlem olasılığı,

$$\begin{aligned} P(O|\lambda) &= \sum_Q P(O, Q|\lambda) = \\ &= \sum_Q \pi_{q_1} b_{q_1}(O_1) a_{q_1 q_2} b_{q_2}(O_2) a_{q_2 q_3} \dots a_{q_{T-1} q_T} b_{q_T}(O_T) \end{aligned} \quad (3.12)$$

şeklinde elde edilir. Zaman içinde açılan spesifik bir geçiş patikası Şekil 3.3.'te kalın geçişlerle verilmiştir. Buradan görüleceği üzere, eşitlikte her bir zamanda belli bir gözlem için altta yatan  $N$  farklı durum olabilir ve  $T$  zaman için

$$\sum_{k=1}^T [\sum_{i=1}^N \sum_{i=1}^N \sum_{i=1}^N \dots (T \text{ tane}) \dots \sum_{i=1}^N \sum_{i=1}^N ] \quad (3.13)$$

işlem ile istenilen sonuç elde edilir.



Şekil 3.3. Trellis şeması (Alpaydın, 2011)

Yukarıdaki değerin elde edilebilmesi için  $TN^T$  hesaplama gerekmektedir. Bu da 100 elemanlı bir sıraya sahip gözlem ile 10 farklı durum göz önünde bulundurulduğunda  $100 \times 10^{100}$  hesaplama demektir. Bu hesaplamanın günümüz koşullarında bile çok maliyetli olduğu açıktır. Buna ek olarak pratikte karşılaşılabilecek birçok problemde  $N$  ve  $T$  değerleri çok daha yüksek olacaktır. Örneğin bir finansal zaman serisinde  $T$  uzunluğu, 1000'ler ile ifade edilebilir.

Bu kadar yüksek sayıda bir hesaplama yapmak problem çözümünü imkânsız hale getirecektir. Bu zorluk *ileri algoritması* ile aşılabilmektedir. İleri ve geri algoritması birlikte kullanıldığında ise SMM'de birçok sorunun cevabı verimli bir şekilde bulunabilmektedir.

#### Üretim olasılığının çözümü:

Üretim olasılığı probleminin analitik çözümü yukarıda gösterildiği gibi oldukça maliyetlidir. Günlük hayattaki problemleri bu yolla çözmeye çalışmak ise bazı problemler için zaman kısıtı yüzünden mümkün değildir. Bunun yerine sayısal yuvarlama hataları dışında kesin sonucu bulmak için dinamik programlama kavramına dayalı bir yöntem olan *ileri algoritması* kullanılabilir. İleri algoritması yinelemeli bir denklem sistemini kullanarak normalde  $TN^T$  zamanda yapılabilecek değerlendirme problem çözümünü  $N^2T$  gibi kabul



edilebilir bir hesaplama yükü ile yapılabilmesini sağlar. İleri algoritması ile  $\sum_Q P(O, Q|\lambda)$  elde edilip  $P(O|\lambda) = \sum_Q P(O, Q|\lambda)$  tümleştirmesi yapılarak değerlendirme problemi çözülmüş olur. İleri algoritması için öncelikle yineleme yapılacak olan *ileri değişkeni* tanımlanır:

$$\alpha_t(i) = P(O_1 O_2 \dots O_t, q_t = S_i | \lambda) \quad (3.14)$$

Başlangıç şu şekilde oluşturulur:

$$\begin{aligned} \alpha_1(i) &= P(O_1, q_1 = S_i | \lambda) \\ \alpha_1(i) &= P(O_1 | q_1 = S_i, \lambda) P(q_1 = S_i | \lambda) \end{aligned} \quad (3.15)$$

$$\alpha_1(i) = \pi_i b_i(O_1), \quad t = 1, 1 \leq i \leq N$$

Sonrasında yineleme prosedürü ise şu şekilde oluşturulur:

$$\begin{aligned} \alpha_{t+1}(j) &= P(O_1 O_2 \dots O_{t+1}, q_{t+1} = S_j | \lambda) \\ &= P(O_1 O_2 \dots O_{t+1} | q_{t+1} = S_j, \lambda) P(q_{t+1} = S_j | \lambda) \\ &= P(O_1 O_2 \dots O_t | q_{t+1} = S_j, \lambda) P(O_{t+1} | q_{t+1} = S_j, \lambda) P(q_{t+1} = S_j | \lambda) \\ &= P(O_1 O_2 \dots O_t, q_{t+1} = S_j | \lambda) P(O_{t+1} | q_{t+1} = S_j, \lambda) \\ &= P(O_{t+1} | q_{t+1} = S_j, \lambda) \sum_{i=1}^N P(O_1 O_2 \dots O_t, q_t = S_i, q_{t+1} = S_j | \lambda) \\ &= P(O_{t+1} | q_{t+1} = S_j, \lambda) \sum_{i=1}^N P(O_1 O_2 \dots O_t, q_{t+1} = S_j | q_t = S_i, \lambda) P(q_t = S_i | \lambda) \\ &= P(O_{t+1} | q_{t+1} = S_j, \lambda) \sum_{i=1}^N P(O_1 O_2 \dots O_t | q_t = S_i, \lambda) P(q_{t+1} = S_j | q_t = S_i, \lambda) P(q_t = S_i | \lambda) \\ &= P(O_{t+1} | q_{t+1} = S_j, \lambda) \sum_{i=1}^N P(O_1 O_2 \dots O_t, q_t = S_i | \lambda) P(q_{t+1} = S_j | q_t = S_i, \lambda) \end{aligned} \quad (3.16)$$

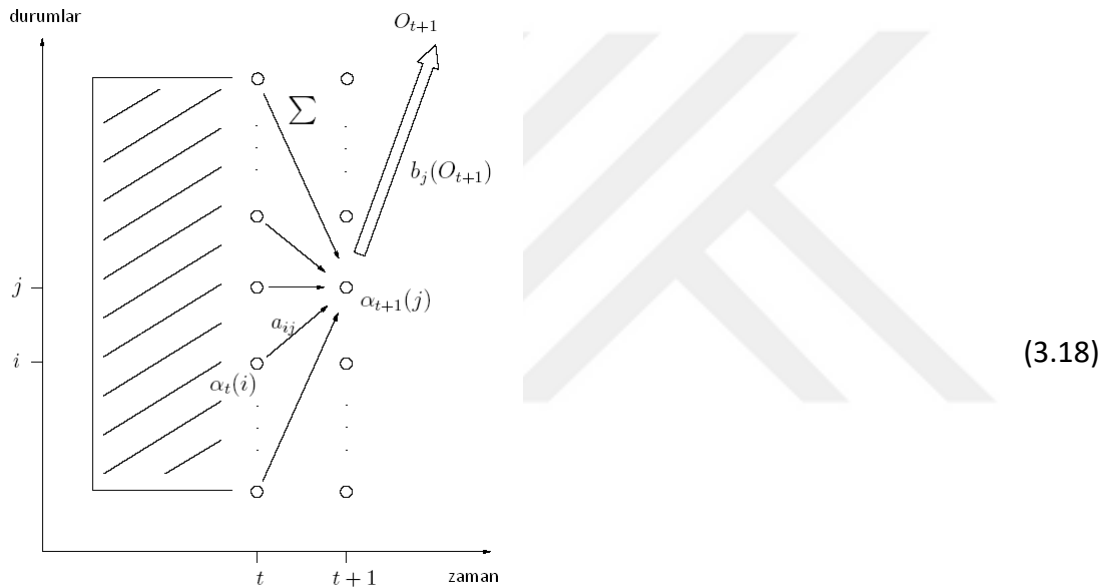
$$\alpha_{t+1}(j) = [\sum_{i=1}^N \alpha_t(i) a_{ij}] b_j(O_{t+1}), 1 \leq t \leq T-1, 1 \leq j \leq N$$

Sonlandırma ise aşağıdaki gibi ifade edilir:

$$P(O|\lambda) = \sum_{i=1}^N P(O, q_T = S_i | \lambda) \quad (3.17)$$

$$P(O|\lambda) = \sum_{i=1}^N \alpha_T(i)$$

İleri algoritması görseli Şekil 3.4.'te görülmektedir.



Şekil 3.4. İleri algoritması

*Geri algoritması* için de benzer yöntem izlenir, öncelikle *geri değişkeni* tanımlanır:

$$\beta_t(i) = P(O_{t+1} \dots O_T | q_t = S_i, \lambda) \quad (3.19)$$

Başlangıç,

$$\beta_T(i) = 1, 1 \leq i \leq N \quad (3.20)$$

Yineleme,

$$\begin{aligned}
\beta_t(i) &= P(O_{t+1} \dots O_T | q_t = S_i, \lambda) \\
&= \sum_{j=1}^N P(O_{t+1} \dots O_T, q_{t+1} = S_j | q_t = S_i, \lambda) \\
&= \sum_{j=1}^N P(O_{t+1} \dots O_T | q_{t+1} = S_j, q_t = S_i, \lambda) P(q_{t+1} = S_j | q_t = S_i, \lambda) \\
&= \sum_{j=1}^N P(O_{t+1} | q_{t+1} = S_j, q_t = S_i, \lambda) P(O_{t+2} \dots O_T | q_{t+1} = S_j, q_t = S_i, \lambda) P(q_{t+1} = S_j | q_t = S_i, \lambda) \\
&= \sum_{j=1}^N P(O_{t+1} | q_{t+1} = S_j, \lambda) P(O_{t+2} \dots O_T | q_{t+1} = S_j, \lambda) P(q_{t+1} = S_j | q_t = S_i, \lambda)
\end{aligned} \tag{3.21}$$

$$\beta_t(i) = \sum_{j=1}^N a_{ij} b_j(O_{t+1}) \beta_{t+1}(j), t = T-1, T-2, \dots, 1, 1 \leq i \leq N$$

şeklinde gösterilir.

#### İdeal üretim olasılığı:

Pratikte en uygun üretim olasılığının gerekliliği de söz konusu olabilmektedir. Bunun için yine ileri algoritmasına benzer bir yöntem ile ideal üretim olasılığı da kabul edilebilir zamanda elde edilecek şekilde bulunabilir. İdeal üretim olasılığı şu şekilde tanımlanır (Fink, 2008):

$$P^*(O|\lambda) = P(O, s^*|\lambda) = \max_s P(O, s|\lambda) \tag{3.22}$$

$\delta_t(i)$  değişkeni  $t$  zamanında ençoklanmış olasılığı tutan değişken olmak üzere,

$$\delta_t(i) = \max_{q_1, q_2, \dots, q_{t-1}} P[q_1 q_2 \dots q_t = i, O_1 O_2 \dots O_t | \lambda] \tag{3.23}$$

Başlangıç,

$$\delta_1(i) = \pi_i b_i(O_1) \tag{3.24}$$

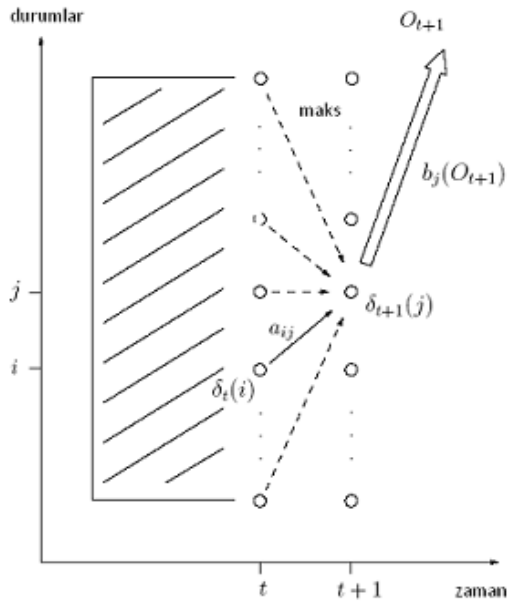
ve yineleme,

$$\delta_t(j) = \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij} b_j(O_t)], 2 \leq t \leq T, 1 \leq j \leq N \quad (3.25)$$

olarak alındığında aranan olasılık

$$P^*(O|\lambda) = P(O, s^*|\lambda) = \max_{1 \leq i \leq N} \delta_T(i) \quad (3.26)$$

olur (Şekil 3.5.).



Şekil 3.5. İdeal üretim olasılığı

### 3.1.1.2 Sır çözme (problem 2)

Belli bir gözlem dizisi için, belli bir optimallik kriterine göre en olası durum dizisinin bulunması problemdir. Optimallik kriteri diziyeye eklenen her bir gözlem için tek tek en olası durumun seçilmesi olabilir. Durumların tek tek en uygunlarının bulunması için gerekli kurulum şu şekilde olabilir (Rabiner ve Juang, 1993):

$$\gamma_t(i) = P(q_t = S_i | O, \lambda) = \frac{P(O | q_t = S_i, \lambda) P(q_t = S_i | \lambda)}{P(O | \lambda)} \quad (3.27)$$

$$= \frac{P(O_1 \dots O_t | q_t = S_i, \lambda) P(O_{t+1} \dots O_T | q_t = S_i, \lambda) P(q_t = S_i | \lambda)}{\sum_{j=1}^N P(O, q_t = S_j | \lambda)}$$

$$\gamma_t(i) = \frac{P(O_1 \dots O_t, q_t = S_i | \lambda) P(O_{t+1} \dots O_T | q_t = S_i, \lambda)}{\sum_{j=1}^N P(O | q_t = S_j, \lambda) P(q_t = S_j | \lambda)} = \frac{\alpha_t(i) \beta_t(i)}{\sum_{j=1}^N \alpha_t(j) \beta_t(j)}$$

$$\sum_{i=1}^N \gamma_t(i) = 1$$

$$q_t = \operatorname{argmax}_{1 \leq i \leq N} \gamma_t(i), 1 \leq t \leq T$$

Bu şekilde oluşturulacak bir durum dizisinde A matrisindeki herhangi bir elemanın  $a_{ij} = 0$  olması durumunda parametreler ile veri arasında uyumsuz bir durum oluşacaktır. Bunun için optimallik kriteri, gözlem dizisi verildiğinde durum dizisinin tümsel olarak uygunluğunun aranması şeklinde ele alınır. Yani  $P(Q|O, \lambda)$  değerini en çoklayan  $Q$  dizisinin bulunması amaçlanır:

$$Q^* = \operatorname{argmax}_Q P(Q|O, \lambda) \quad (3.28)$$

En çoklanacak ifade şu şekilde ele alınabilir:

$$P(Q|O, \lambda) = \frac{P(Q, O | \lambda)}{P(O | \lambda)} \quad (3.29)$$

Burada problemin yapısı gereği  $P(O | \lambda)$  ifadesi zaten sabit olacaktır. Bu nedenle  $P(Q|O, \lambda)$  ve  $P(Q, O | \lambda)$ 'yi en çoklayan sıra aynı olacaktır. Yani

$$Q^* = \operatorname{argmax}_Q P(Q, O | \lambda) = \operatorname{argmax}_Q P(Q|O, \lambda) \quad (3.30)$$

olur. Bu durumda

$$\delta_t(i) = \max_{q_1, q_2, \dots, q_{t-1}} P[q_1 q_2 \dots q_t = i, O_1 O_2 \dots O_t | \lambda] \quad (3.31)$$

$$\delta_{t+1}(j) = \max_i \delta_t(i) a_{ij} b_j(O_{t+1})$$

olacaktır. Algoritma aşağıdaki adımlara sahip olur.

Başlangıç:

$$\delta_1(i) = \pi_i b_i(O_1), 1 \leq i \leq N \quad (3.32)$$

$$\psi_1(i) = 0$$

Yineleme:

$$\delta_t(j) = \max_{1 \leq i \leq N} \{ \delta_{t-1}(i) a_{ij} \} b_j(O_t), 2 \leq t \leq T, 1 \leq j \leq N \quad (3.33)$$

$$\psi_t(j) = \operatorname{argmax}_{1 \leq i \leq N} \delta_{t-1}(i) a_{ij}, 2 \leq t \leq T, 1 \leq j \leq N$$

Sonlandırma:

$$P^* = \max_{1 \leq i \leq N} \delta_T(i) \quad (3.34)$$

$$q_T^* = \operatorname{argmax}_{1 \leq i \leq N} \delta_T(i)$$

$$q_t^* = \psi_{t+1}(q_{t+1}^*), t = T-1, T-2, \dots, 1 \quad (3.35)$$

$\psi_t(j)$  her adımda ortaya çıkacak durum değerini tutan değişkendir. Durum dizisi bilgisini barındırır. Bu algoritma Viterbi Algoritması olarak bilinir.

### 3.1.1.3 Öğrenme problemi (problem 3)

Öğrenme problemi ise model parametrelerinin nasıl tahmin edileceği üzerinedir. Bu problemde belli bir gözlem sırasının olasılığını maksimum yapacak model parametreleri,  $\lambda = (A, B, \pi)$ , tahmin edilmeye çalışılır. Bu problemin analitik bir çözümü bulunmamaktadır. Ancak  $\lambda = (A, B, \pi)$  değerlerini  $P(O|\lambda)$  -global olmasa bile- lokal olarak

tahmin edebilecek yinelemeli bir algoritma geliştirilmiştir. Baum ve diğerlerinin geliştirdiği bu algoritma *Baum-Welch (BW)* yöntemi adı ile bilinir (Rabiner ve Juang, 1993).

BW algoritması *beklenti büyütme algoritması (BBA)* olarak bilinen daha genel bir algoritma sınıfının SMM için kullanılan şeklidir.

### **3.1.1.3.1 Beklenti büyütme algoritması**

BBA bir parametre tahmin yaklaşımıdır. Her ne kadar algoritma olarak adlandırılrsa da, bu yaklaşımın birçok farklı algoritmayı içinde barındırabilecek bir çözüm süreci ve eniyileme (optimizasyon) yöntemi olarak görülmesi daha doğru olacaktır zira yöntemin adımları uygulamadan uygulamaya farklılaşabilmekte izlenecek kesin bir yol bulunmamaktadır. Bu çalışmada da literatüre uygun olarak adlandırma “algoritma” şeklinde yapılmıştır.

BBA, temel olarak yinelemeli bir eniyileme prosedürüdür. Yöntem ile ilgili ilk çalışmalar gen frekans tahmini problemi ile Ceppellini ve diğerleri (1955)'in yayınlarına kadar uzanmaktadır. Daha genel bir çalışma ise Hartley (1958) ile Baum ve diğerleri (1970) tarafından yapılmıştır. Yine de algoritma Dempster ve diğerleri (1977)'nin çalışmaları ile tanınmıştır (Do ve Batzoglou, 2008). Dempster ve diğerleri algoritmanın çeşitli özellikleri ile ilgili kanıtlar sunmuş ve BBA'nın özel bir hali için oldukça etkili bir çözüm yöntemi önermişlerdir. Algoritma ile ilgili diğer önemli çalışma ise J. Wu'nun 1983 tarihli makalesidir. J. Wu, BBA için genel yakınsama teorisinin olmadığını göstermiş ve BBA'nın daha genel bir hali için kanıt sunmuştur. J. Wu ayrıca algoritma ile daha etkin bir çalışma yapılması ile ilgili öneriler sunmuştur (Kobayashi, Mark ve Turin, 2012). Yöntemin gen biliminde, ekonometride, bilgisayar bilimi ve elektronik mühendisliğinde kullanılan başarılı örnekleri bulunmaktadır (Moon, 1996; Vashegi, 2007; McLackhan ve Krishnan, 2008).

BBA'nın bir avantajı aynı tür problemlerin çözümünde Newton-Raphson metodunu uygulamanın zor olduğu, Hessian matrisin zor elde edilebildiği durumlarda uygulanabilmesidir. Ayrıca BBA kötü başlangıç değerleri ile bile Newton-Raphson'a göre daha iyi sonuçlar vermektedirler. BBA, Newton-Raphson'a göre daha stabil bir algoritmadır (Gupta ve Chen, 2010). Bununla birlikte algoritma kuadratik yakınsama yapan Newton-Raphson algoritmasından farklı olarak doğrusal yakınsama yapmaktadır. BBA'yı hızlandıran

çeşitli yöntemler önerilmiştir, bunlardan ikisi Louis'nin 1982 tarihli çalışmasında temel aldığı Aitken hızlandırma yöntemi ve quasi-Newton yöntemleri ile birlikte kullanılan EM yöntemlerdir (Tanner, 1996; Jamshidian ve Jennrich, 1997).

BBA diğer bir çok yinelemeli algoritma ile benzer şekilde çalışmaktadır. Fakat bir algoritmanın BBA olması bazı koşullara bağlıdır. Bir kere BBA, beklenti adımı ile ençoklama adımı olmak üzere iki adımdan meydana gelmektedir. Ayrıca model değişkenlerinde üstü kapalı olarak  $\theta \rightarrow Y \rightarrow X$  bağımlılık yapısı yani Markovyen ilişki bulunmaktadır. Bu bağımlılık yapısı kullanılarak diferansiyel hesap kullanılmadan yineleme yapılabilmektedir. Bunlara ek olarak her BBA yapısında gizli ya da eksik veriler bulunmaktadır (Roche, 2012).

BBA eksik verili olan problemlerin çözümü için kullanılır. Bu da iki türlü olabilir: Ya modele giren veriler çeşitli nedenlerle eksiktir yani verilerin bir kısmı gözlenememiştir ya da problem öyle bir özelliğe sahiptir ki çözümü elde edilemediği halde bazı verilerin eksik olduğu varsayımı altında analitik olarak çözülebilir hale gelir. Örneğin karışım modelleri, SMM problemleri ikinci tür özelliğe sahiptir (Bilmes, 1998). Bu tür problemler örüntü tanıma problemleri olarak da bilinmektedir.

### BBA çalışma yöntemi

$P(\mathbf{x}|\boldsymbol{\theta})$  gibi bir dağılımdan seçilmiş bir veri kümesi  $\mathbf{X} = \{X_1, X_2, \dots, X_n\}$  şeklinde tanımlandığında yoğunluk fonksiyonu RD'lerin bağımsız olduğu varsayımı altında şu şekilde yazılabilir (Bilmes, 1998):

$$P(\mathbf{x}|\boldsymbol{\theta}) = \prod_{i=1}^N P(x_i|\boldsymbol{\theta}) = L(\boldsymbol{\theta}|\mathbf{X}) \quad (3.36)$$

Bu ifade olabirliğe karşılık gelmektedir. En çok olabirlik ise

$$\boldsymbol{\theta}^* = \max_{\boldsymbol{\theta}} L(\boldsymbol{\theta}|\mathbf{X}) \quad (3.37)$$

ile gösterilir.



BBA'nın problem tanımı yukarıda bahsedilen ikinci tür problemler için şu şekilde yapılabilir: İki tane örnek uzay bulunmaktadır. Bunlardan biri tam olmayan veri uzayıdır,  $\mathcal{Y}$  olarak adlandırılır. Diğeri ise tam veri uzayıdır,  $\mathcal{X}$  olarak adlandırılır. Gözlenen veri  $y$ ,  $\mathcal{Y}$  uzayından bir realizasyondur. Matematiksel olarak,

$$x \rightarrow y(x) \quad (3.38)$$

diğeri bir gösterimle

$$y = y(x) \quad (3.39)$$

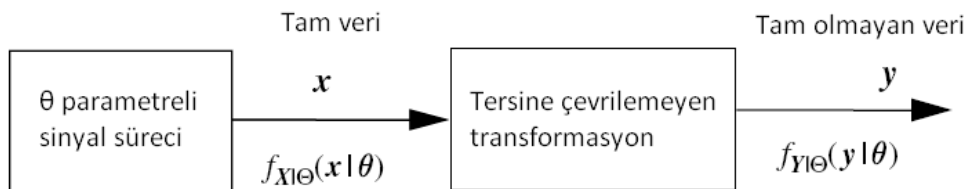
yazılabilmektedir (Dempster ve diğeri, 1977). Buna göre bir  $X$  gözlemlendiğinde buna ilişkin bir  $Y$ 'nin gözlenmekte olduğunu düşünülür. Başka bir ifade ile ise,

$$\mathcal{X}(Y) = \{X: f(X) = Y\} \quad (3.40)$$

yazılabilir (Collins, 1997). Modelde  $f(x|\varphi)$  ile gösterilen örneklem yoğunlukları olduğu varsayılır ve buna karşılık  $g(y|\varphi)$  yoğunlukları türetilir.  $f(\dots | \dots)$  ile ele alınan tam veri,

$$g(y|\varphi) = \int_{\mathcal{X}(y)} f(x|\varphi) dx \quad (3.41)$$

ile tam olmayan veriye ilişkilidir. Bu fonksiyonel ilişki Şekil 3.6. ile özetlemiştir (Vaseghi, 2007):



Şekil 3.6. Tam veri, tam olmayan veri transformasyonu

BBA,

$$L(\boldsymbol{\theta}) = \log g(\mathbf{Y}|\boldsymbol{\theta}) \quad (3.42)$$

değerini maksimize eden  $\boldsymbol{\theta}$ 'yı elde etmeyi sağlar. Burada  $\log f(\mathbf{X}|\boldsymbol{\theta})$  genellikle kolaylıkla tanımlanan analitik olarak çözülebilir bir maksimuma sahiptir fakat  $L(\boldsymbol{\theta})$ 'in maksimizasyonun analitik bir çözümü yoktur.  $L(\boldsymbol{\theta})$  genellikle birden fazla maksimuma sahiptir (Collins, 1997).

BB yinelemeli bir eniyileme algoritması olduğundan, algoritma yinelemeyi

$$\boldsymbol{\theta}_t \rightarrow \boldsymbol{\theta}_{t+1} \quad (3.43)$$

geçişini

$$L(\boldsymbol{\theta}_{t+1}) \geq L(\boldsymbol{\theta}_t) \quad (3.44)$$

ilişkisi ile ele alan bir seri parametre kurulumu tanımlar. Belli koşullar altında eşitlik sadece kritik noktalarda sağlanmaktadır.

BBA için yineleme prosedürü şu şekilde oluşturulmaktadır:

$$L(\theta) = \ln P(\mathbf{X}|\theta) \quad (3.45)$$

olmak üzere, nihai olabilirlik değerinin yinelemelerdeki değerlerden daha büyük olması beklenir, böylelikle

$$L(\theta) > L(\theta_n) \quad (3.46)$$

yazılabilir. Bu matematiksel ifadeler ile

$$L(\theta) - L(\theta_n) \geq \sum_Z P(Z|\mathbf{X}, \theta_n) \ln \left( \frac{P(\mathbf{X}|Z, \theta_n) P(Z|\theta_n)}{P(\mathbf{X}, \theta_n)} \right) \triangleq \Delta(\theta|\theta_n) \quad (3.47)$$

şeklinde bir tanımlama yapıldığında eşitsizlik

$$L(\theta) \geq L(\theta_n) + \Delta(\theta|\theta_n) \triangleq l(\theta|\theta_n) \quad (3.48)$$

şeklinde yazılabilir. Buradan da

$$\theta_{n+1} = \operatorname{argmax}_{\theta} l(\theta|\theta_n) \quad (3.49)$$

olacaktır.  $\theta'$ 'ye göre sabit olan terimler atıldığında

$$\begin{aligned} \theta_{n+1} &= \operatorname{argmax}_{\theta} \sum_Z \ln(Z, X|\theta) P(Z|X, \theta_n) = \operatorname{argmax}_{\theta} E_{Z|X, \theta_n} \ln(Z, X|\theta) = \\ &\operatorname{argmax}_{\theta} Q(\theta|\theta_n) \end{aligned} \quad (3.50)$$

elde edilir. Ayrıca bu tanımlamalara göre,

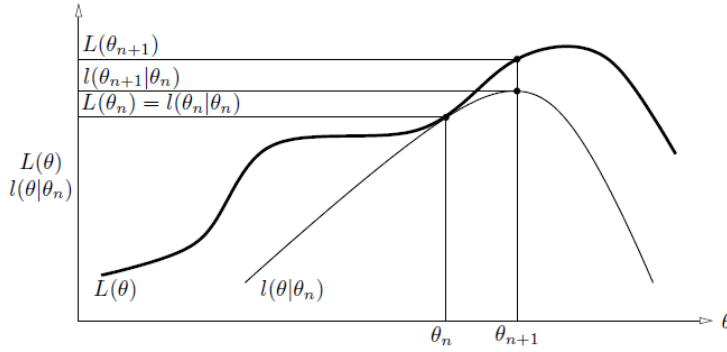
$$l(\theta_n|\theta_n) = L(\theta_n) + \Delta(\theta_n|\theta_n) = L(\theta_n) \quad (3.51)$$

olacaktır.  $\theta_{n+1}$ ,  $\Delta(\theta|\theta_n)$  değerini maksimize edecek şekilde seçildiğinden,

$$\Delta(\theta_{n+1}|\theta_n) \geq \Delta(\theta_n|\theta_n) = 0 \quad (3.52)$$

olacak ve böylelikle her bir yinelemede  $L(\theta)$ 'nin azalmayan bir yapıda olacaktır. Buradan şu sonuç çıkar:  $l(\theta|\theta_n)$  fonksiyonu  $L(\theta)$  fonksiyonu ile yukarıdan sınırlıdır. BBA  $\theta_{n+1}$ 'i  $l(\theta|\theta_n)$  fonksiyonunu maksimize edecek şekilde seçer ve her maksimizasyon  $L(\theta)$  değerini artıracaktır. Yani BBA alt sınırın maksimize edilmesi ile maksimizasyonu yakalamaktadır. Bu nedenle BBA için bir dolaylı maksimizasyon yöntemi denebilir (L. Wu, 2010).

BBA'nın yakınsamasını garantileyen bir yakınsama teoremi olmamasına rağmen beklendiği gibi bir kritik nokta elde edilmektedir. Fakat yakınsama sonunda global değil yerel bir değer elde edilebilir. Yinelemenin çalışma yöntemi Şekil 3.7.'de gösterilmiştir (Borman, 2009).



Şekil 3.7. BBA'nın çalışması

Elde edilen bu ifadelerden BBA'nın yineleme adımları tahmin ve ençoklama şu şekilde olur (McLachlan ve Krishnan, 2008):

Tahmin (beklenti) adımı:

$$\log L_c(\psi) = \log g_c(x; \psi) \quad (3.53)$$

olarak verildiğinde

$$Q(\psi; \psi^{(k)}) = E_{\psi^{(k)}}\{\log L_c(\psi) | y\} \quad (3.54)$$

değeri hesaplanır.

Ençoklama adımı:

Aşağıdaki ifade ile,

$$\psi_{t+1} = \underset{\psi}{\operatorname{argmax}} Q(\psi; \psi^{(k)}) \quad (3.55)$$

ençoklama yapılmaktadır. Bu anlamda BBA bir tepe tırmanma algoritmasına benzemektedir (Koller ve Friedman 2009).

BBA sezgisel olarak şu şekilde çalışır: Eğer tüm veri erişilebilir olsaydı  $\log f(\mathbf{X}|\boldsymbol{\theta}')$  fonksiyonunu maksimize etmek için  $\boldsymbol{\theta}'$  değeri tahmin edilirdi. Fakat tüm veri erişilebilir olmadığına, gözlenmiş veri ve  $\boldsymbol{\theta}'$ 'nin güncel değeri verilmişken  $\log f(\mathbf{X}|\boldsymbol{\theta}')$  'nin beklenen değeri maksimize edilmektedir (Ghahramani ve Jordan, 1994). BBA'ya benzer bir algoritma da Enbüyüklenme-Enküçükleme ya da Enküçükleme-Enbüyüklenme algoritmasıdır (Demidenko, 2013).

Dempster ve diğerleri (1977), BBA'nın kullanımında üstel dağılım ailesinin kullanım gücünü ortaya koymuşlardır. Bir parametre kümesi  $\boldsymbol{\theta} = \{\theta_1, \dots, \theta_n\}$  olarak gösterildiğinde,  $C_i(\boldsymbol{\theta}) = \theta_i$  doğal parametreler  $T$  yeterli istatistik olmak üzere tek değişkenli bir üstel dağılım ailesi,

$$f(x|\boldsymbol{\theta}) = \prod_x \exp[c(\boldsymbol{\theta})T(x_i)] \quad (3.56)$$

olarak ele alındığında,

$$\int P(X|\boldsymbol{\theta})dX = 1 \text{ ve } d(\boldsymbol{\theta}) = -\log \int \{\exp[C_i(\boldsymbol{\theta})T_i(X) + S(X)]\}I_A dX \quad (3.57)$$

kullanılarak

$$d'(\boldsymbol{\theta}) = -E[T(X)|\boldsymbol{\theta}] \quad (3.58)$$

olduğu gösterilebilmektedir. Ayrıca

$$L(X|\boldsymbol{\theta}) = \sum C_i(\boldsymbol{\theta})T_i(\boldsymbol{\theta}) + d(\boldsymbol{\theta}) + S(X) \quad (3.59)$$

olmak üzere yukarıdaki bilgilerle birleştirerek  $L'(X|\boldsymbol{\theta})$  bulunursa,

$$L'(X|\boldsymbol{\theta}) = T(X) + d'(\boldsymbol{\theta}) = T(X) - E[T(X)|\boldsymbol{\theta}] \quad (3.60)$$

elde edilir. Buradan  $L'(X|\boldsymbol{\theta})$  fonksiyonunu maksimize eden değer

$$L'(X|\boldsymbol{\theta}) = 0 \text{ noktasında } T(X) = E[T(X)|\boldsymbol{\theta}] \quad (3.61)$$

olmaktadır. Bu özellik ençoklama adımının kolaylıkla yapılmasını sağlamaktadır. BBA çok etkili bir algoritma olmasına rağmen, bazı eksik olduğu yönler de bulunmaktadır. BBA'nın özellikleri aşağıdaki şekilde sıranabilir.

#### Avantajlar:

- BBA matematiksel olduğu kadar pratikte de kolay uygulanabilir ve birçok problemin çözümünde başarılı sonuçlar üreten bir algoritmadır. Ayrıca BBA bilgisayar kaynaklarını çok tüketmeden çalışabilmektedir. Özellikle üstel aile dağılımlarında algoritma süreci daha da basit olacaktır.
- BBA sayısal kararlı bir algoritmadır. Her bir yinelemede olabilirliğin azalmayacağı kesindir ve bu matematiksel olarak kanıtlanmıştır.
- Genel koşullarda BBA global ekstremum sonucunu verecektir.
- BBA özellikle üstel dağılım ailesi ile çok pratik bir algoritma haline gelmektedir.

#### Dezavantajlar:

- BBA bir ençoklama yapacaktır fakat yapılan bu ençoklama global en büyük değerin bulunduğunu garanti etmez. Fakat algoritmanın global ekstremum sonucu elde etmesi için çeşitli yöntemler bulunmaktadır.
- BBA yavaş çalışabilen bir algoritmadır, yakınsama için çok sayıda yineleme gerekebilir. BBA, Newton yöntemi ile karşılaştırıldığında başlangıçta her bir BB yinelemesi 4 Newton yöntemi yinelemesine denk gelirken, kritik değere yaklaştıkça Newton yönteminden 5 kat daha yavaş kalır (Jiang, 2007; Schlattmann, 2009).
- BBA'da beklenti adımı bazı problemlerde analitik olarak çözülememektedir bu tür durumlarda Monte-Carlo yaklaşımı kullanılabilir (Demidenko, 2013).
- BB algoritması çok genel olup, uygulanması problemden probleme farklılık göstermektedir. Her problem yapısı için izlenecek belli bir yol bulunmamaktadır.
- BBA algoritması eksik verili problemlere uygulandığında eksik verilerin oranı arttıkça yakınsama yavaşlamaktadır. Ayrıca gerçek değere yaklaştıkça yakınsama oranı da azalacaktır (Fahrmeir ve Tutz, 1994) .

- Algoritma sürecinde standart hataların elde edilmesi standart bir prosedür bulunmamakta ve sadece yaklaşık değer bulma prosedürleri öne sürülmektedir. Pratikte algoritmanın çalışması için çok sayıda veri olması gerektiği belirtilmiştir (Frühwirth, 2006). Buna ek olarak algoritmanın genel yakınsama teorimi şu an bulunmamaktadır (Kneib ve Tutz, 2010).

### 3.1.1.3.2 Baum-Welch algoritması

Baum-Welch algoritması temelde daha genel bir yöntem olan BBA'ya dayanır. SMM için modeli eşsiz bir şekilde ifade eden tüm parametrelerin tahmin edilmesi amacı ile Baum ve diğerleri (1970) tarafından geliştirilmiştir. Baum-Welch algoritmasında BB'deki yöntem izlenir:

$$Q(\lambda, \lambda') = \sum_{qQ} \log P(O, q|\lambda) P(O, q|\lambda')$$

$$Q(\lambda, \lambda') = \sum_{qQ} \log \pi_{q_0} P(O, q|\lambda') + \sum_{qQ} (\sum_{t=1}^T \log a_{q_{t-1}q_t}) P(O, q|\lambda') + \sum_{qQ} (\sum_{t=1}^T \log b_{q_t}(O_t)) P(O, q|\lambda') \quad (3.62)$$

Yukarıdaki eşitlikte her bir terim  $\lambda$  değişkenine göre maksimize edilebilir. Bu maksimizasyonda aşağıdaki eşitliklerdeki kısıtların sağlanması gerekir:

$$\sum_{i=1}^N \pi_i = 1$$

$$\sum_{j=1}^N a_{ij} = 1, 1 \leq i \leq N \quad (3.63)$$

$$\sum_{k=1}^M b_j(k) = 1, 1 \leq j \leq N$$

Maksimize edilecek her terim  $\sum_{j=1}^N w_j \log y_j$  yapısına sahip olup  $\sum_{j=1}^N \log y_j = 1, y_j \geq 0$  kısıtları ile tek bir global maksimum noktasına sahip olur:

$$y_j = \frac{w_j}{\sum_{k=1}^N w_k}, j = 1, 2, \dots, N \quad (3.64)$$

Böylelikle aranan değerler,

$$\delta(o_k, v_k) = \begin{cases} 1, & o_k = v_k \\ 0, & d. d. \end{cases} \quad (3.65)$$

olmak üzere

$$\bar{\pi}_i = \frac{P(O, q_0=i|\lambda)}{P(O|\lambda)}$$

$$\bar{a}_{ij} = \frac{\sum_{t=1}^T P(O, q_{t-1}=i, q_t=j|\lambda)}{\sum_{t=1}^T P(O, q_{t-1}=i|\lambda)} \quad (3.66)$$

$$\bar{b}_j(k) = \frac{\sum_{t=1}^T P(O, q_t=i|\lambda)\delta(o_k, v_k)}{\sum_{t=1}^T P(O, q_t=i|\lambda)}$$

olarak elde edilir. Kolaylık sağlayacak yeni ara değişken tanımlanabilir:

$$\begin{aligned} \xi_t(i, j) &= P(q_t = S_i, q_{t+1} = S_j | O, \lambda) \\ &= \frac{P(O|q_t=S_i, q_{t+1}=S_j, \lambda)P(q_t=S_i, q_{t+1}=S_j|\lambda)}{P(O|\lambda)} \\ &= \frac{P(O|q_t = S_i, q_{t+1} = S_j, \lambda)P(q_{t+1} = S_j|q_t = S_i, \lambda)P(q_t=S_i|\lambda)}{P(O|\lambda)} \\ &= P(O_1 \dots O_t | q_t = S_i, \lambda)P(O_{t+1} | q_{t+1} = S_j, \lambda)P(O_{t+2} \dots O_T | q_{t+1} = S_j, \lambda)a_{ij}P(q_t = S_i|\lambda) \left(\frac{1}{P(O|\lambda)}\right) \\ &= \left(\frac{1}{P(O|\lambda)}\right)P(O_1 \dots O_t, q_t = S_i, \lambda)P(O_{t+1} | q_{t+1} = S_j, \lambda)P(O_{t+2} \dots O_T | q_{t+1} = S_j, \lambda)a_{ij} \\ &= \frac{\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{P(O|\lambda)} \\ \xi_t(i, j) &= \frac{\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{\sum_{i=1}^N \sum_{j=1}^N \alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)} \end{aligned} \quad (3.67)$$

Ayrıca şu eşitlik de tanımdan geçerli olacaktır:



$$\gamma_t(i) = \sum_{j=1}^N \xi_t(i, j) \quad (3.68)$$

Buradan da,

$\sum_{t=1}^{T-1} \gamma_t(i)$ :  $S_i$ 'den geçişlerin beklenen sayısı

$$\sum_{t=1}^{T-1} \xi_t(i, j): S_i \text{'den } S_j \text{'ye geçişlerin beklenen sayısı} \quad (3.69)$$

$\bar{\pi}_i$ :  $t = 1$  anında  $S_i$  durumunda beklenen sıklık (zamanların sayısı)

Böylelikle nihai olarak,  $i$  durumunda 1 zamanında olma frekansı:

$$\bar{\pi}_i = \gamma_1(i) \quad (3.70)$$

olarak tanımlanmakta A matrisini oluşturan elemanlar

$$\bar{a}_{ij} = \frac{S_i \text{'den } S_j \text{'ye geçişlerin beklenen sayısı}}{S_i \text{'den geçişlerin beklenen sayısı}} \quad (3.71)$$

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)}$$

B matrisini oluşturan elemanlar

$$\bar{b}_j(k) = \frac{v_k \text{ gözlem sembolü ve } j \text{ durumlu zamanların beklenen sayısı}}{j \text{ durumlu zamanların beklenen sayısı}} \quad (3.72)$$

$$\bar{b}_j(k) = \frac{\sum_{t=1}^T \gamma_t(j) 1_{(O_t=v_k)}}{\sum_{t=1}^T \gamma_t(j)}$$

olarak elde edilmektedir. Bu değişkenlerle sol taraf güncellemeye denk gelirken, sağ tarafta ise güncel veriler kullanılır.

## 3.2 Yapay Sinir Ağları

### 3.2.1 Yapay sinir ağları genel bilgiler

İnsanođlu yaptıđı icatların birçođunda dođanın örneklerinden ilham almıřtır. Yapay sinir ağları da bu motivasyon ile geliřtirilmiř bir modeldir.

İnsan beyni ve bilgisayarların çözdüđü problemlere bakıldıđında her ikisinin de farklı tür problemlerde daha verimli çalıřtıđı gözlenmektedir. Bilgisayar çok haneli sayıları kolayca çarpıp toplayabilirken, insan beyni bu işlemler için oldukça yavaş kalmakta ve çaba sarf etmektedir. Fakat insanlar için çok basit bir prosedür olan ses ya da görüntü tanıma işlemi bilgisayar için oldukça karmařık bir işlemdir. Bilgisayarın insan için çok basit olan konuşsal, görsel, işitsel işlemleri yaparken, performansının çok kötü olduđu bilinmektedir (Beale ve Jackson, 1990).

Bu farklılıkların nedeni beynin ve bilgisayarın çalışma yöntemlerinden kaynaklanmaktadır. Bilgisayar programları seri bir şekilde kendisine verilen görevleri yerine getirirken, beyin problemleri paralel bir yapı ile ele almaktadır (Beale ve Jackson, 1990). Bu yüzden konuşma, işitme ve görsel tanıma gibi paralel işleme gerektiren işlemlerde beyin daha başarılıdır.

Beyin ayrıca tecrübe ettiklerinden öğrenebilmekte ve daha sonra bu bilgiyi kullanabilme yeteneđine sahiptir. Bunların da ötesinde beyin aldıđı kararlarda sezi ve hislerini kullanabilmektedir (Carter, 2009).

Beyni oluřturan yapı taşlarına, nöronlara bakıldıđında, canlının verdiđi karmařık kararları verecek özellikli bir yapısı olmadıđı görölmektedir. Ayrıca nöronların herhangi bir elektronik ađgıta göre çok yavaş çalıřtıđı bilinmektedir, öyle ki nöronlar günümüzde kullanılan bir silikon kapısına göre  $10^6$  kat daha yavaş çalışmaktadırlar (Haykin, 1999).

Beynin işleme ve karar verme gücünün, beynin yapıtaşları olan nöronların dađıtık ve paralel çalışmasından geldiđine inanılmaktadır. Nöronların kendine diđer nöronlardan gelen sinyalleri birleřtirerek-birlikte kullanarak işlediđi, işledikten sonra da kendi ile bađlantılı diđer nöronlara gönderdiđi bilinmektedir. Sinyal gönderen nörona bađlı diđer nöronlar da

aynı prosedür ile bilgiyi işlemekte ve iletmektedir. Beyin nöronların bu şekilde bağlantıcı ve dağıtık bir yapı içinde çalışması ile karar vermekte ve yine bu şekilde öğrenmekte ve öğrendiklerini hatırlamaktadır. Böyle bir yapıda her nöron kendine gelen sinyalleri tek başına yani bağımsız olarak işlemekte ve sinyalin gönderilip gönderilmemesine sadece ilgili nöronda karar verilmektedir. Bu bağımsızlık, nöronların sistem içinde eş zamanlı olarak çalışmasına olanak sağlamaktadır (Gurney, 1997).

Beyin hücrelerinin paralel ve dağıtık olarak bu kadar verimli ve güçlü bir karar alma ve öğrenme sistemi oluşturması, bir modelleme aracının geliştirilmesi için esin kaynağı olmuştur. Bu çalışma alanının oluşmaya başladığı 1940'lı yıllarda bile devrelerle kolaylıkla taklit edilebilen nöronlar ve bu devrelerin uygun şekilde kurulması ile oluşturulan sistemlerin karar vermesi ve hatta öğrenmesi fikri bilim insanlarına çok çekici gelmiştir. Böylelikle yapay sinir ağları matematiksel bir çözüm aracı olarak ortaya atılmıştır (Anderson ve McNeil, 1992).

### **3.2.1.1 Biyolojik sinir ağı**

Beynin biyolojik yapısının incelenmesi ile ilgili çalışmalara binlerce yıl önce başlamıştır. Beynin öğrenme becerileri ile ilgili kayda değer çalışmaların ise 19. yy. sonlarında yapılmaya başlandığı bilinmektedir. Genel olarak özetlemek gerekirse (Carter, 2009):

- M.Ö. 1700'lerde Mısırlılar beynin şeklini gösteren belgeler bırakmışlardır.
- M.Ö. 450'lerde Eski Yunanlılar sezgisel işlemleri beyin ile ilişkilendirmişlerdir.
- 1543 yılında Andreas Vesalius beynin modern ve detaylı çizimlerini yayımlanmıştır.
- 1664 yılında Thomas Willis ilk beyin atlasını yayımlamıştır.
- 1791 yılında Luigi Galvani kurbağa bacağına yaptığı çalışmalarda sinirsel aktivitenin elektriksel olduğunu bulmuştur.
- 1849 yılında Hermann von Helmholtz sinirsel iletimin hızını bulmuştur.
- 1889 yılında Ramon Cajal nöronların bağımsız ve beynin temel birimi olduğunu ileri sürmüştür bu çalışması ile de Nobel ödülü kazanmıştır.
- 1906 yılında Ramon Cajal nöronların nasıl iletiştiğini tarif etmiştir.

Beynin işleme ve öğrenme yönteminin anlaşılmasını kolaylaştıran İspanyol sinir-bilimci Ramon Cajal'ın 1800'lü yılların sonlarındaki çalışmalarıdır. Ramon Cajal bu çalışmaları 3 cilt halinde yayınlamış (1889-1904) ve 1906 yılında Nobel ödülü almıştır. Bu çalışmalar beynin yapı taşlarının nöron birimleri olduğu, nöronların yapıları ve birbiri ile olan iletişimini anlatan ilk çalışmalar olmaları nedeniyle önemlidirler (Barquin, 2001).

Bir insanın öğrenmesi ve tecrübelerine göre karar alıp bunları uygulaması kısacası çevresi ile iletişim kurması beyni de kapsayan sinir sisteminin bütünün uyum içinde çalışması ile olur. Gözleri görmeyen, kulakları duymayan bir köpek öğrenemeyecektir de (Beale, 1991).

Sinir sistemi ise 3 kademeli yapının birleşimi ile oluşmuştur (Şekil 3.8.): alıcılar, sinir ağı, tepki sinirleri. Alıcılar çevreden aldıkları uyarıları elektriksel sinyallere dönüştürüp, sinirler üzerinden, sinir ağının merkez organı beyne iletir (Haykin, 1999). Örnek vermek gerekirse, burundaki alıcılar havadaki koku molekülleri ile reaksiyona geçerek ürettikleri koku bilgisini sinirler üzerinden beyne gönderir. Görme, duyma, koklama, dokunma alıcıları da benzer yöntemlerle çalışır. Tepki sinirleri ise sinir ağının ürettiği elektrik sinyallerini görülebilir cevaplara dönüştürürler. Bu şekilde sinir sistemi ile çevreden ve vücudun içinden gelen etkiler alıcı sinirlerle algılayıp sinir ağında karar verip eğer gerekiyorsa tepki sinirleri ile cevap vererek etkileşime geçer (Rogers, 2011).



Şekil 3.8. Sinir sistemi

Sinir ağı, merkez organ beyin ile beynin iletişimini sağlayan omurilik ve sinirlerden oluşur. YSA ise biyolojik sinir ağı alt sistemi içinde merkez karar verme organı 3 kısımdan oluşan beynin *serebral korteks* bölümüne benzemektedir. Beyin karar verme, yeni deneyimlerle öğrenme ve bu öğrendiklerini hatırlama görevlerini serebral korteks kısmında gerçekleşmektedir (Beale, 1991).

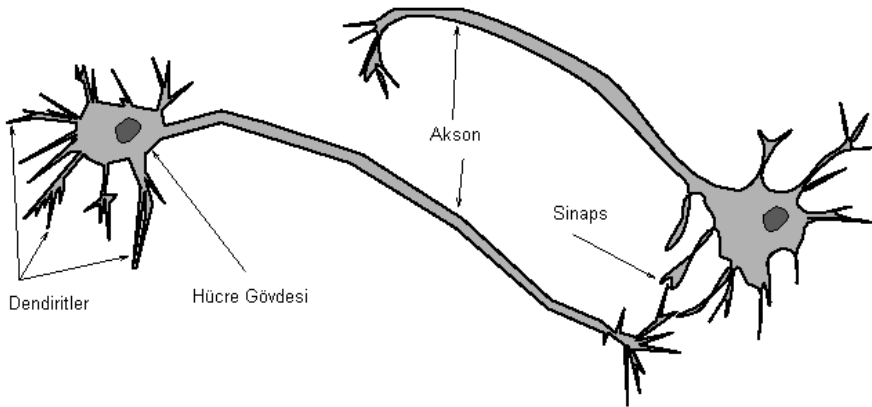
Beynin en temel birimleri, beyin hücreleri – nöronlardır. Beyinde  $10^{10}$  kadar nöron bulunmaktadır ve her bir nöron genel olarak  $10^2$  ile  $10^4$  arasında bağlantıya sahiptir fakat bir nöron diğer nöronlarla  $2 \times 10^5$  kadar bağlantı kurulduğu görülebilmektedir (Anderson ve McNeil, 1992). Beyin hücreleri tek bir karar verme işlemini silikon kapılarına göre  $10^{10}$  kat daha verimli yapmaktadır.

Nöronlar, genel olarak ele alındığında, 3 kısımdan meydana gelir (Gurney, 1997):

- **Gövde:** Sinir hücresi gövdesi
- **Dendritler:** Gövdeye gelen sinyal alıcı lifler. Bir hücrede çok sayıda dendrit bulunmaktadır.
- **Akson:** Gövdeden çıkan sinyal iletici lif. Her hücrede bir adet akson bulunmaktadır

Bu kısımlar arasındaki bilgi geçişi elektrik sinyalleri ile olur.

Bir sinir hücresi üzerinden geçen bir sinyalin bir sonraki sinir hücresine iletimi ise *sinaps* adı verilen boşluklarda olur (Şekil 3.9., [teco.edu/~albrecht/neuro/html/node7.html](http://teco.edu/~albrecht/neuro/html/node7.html)). Sinapslardan yapılan bu bilgi aktarımı hem elektriksel hem kimyasal niteliktedir (Rogers, 2011).



Şekil 3.9. Sinir hücresi

Bir beyin hücresi, bağlı bulunduğu diğer hücrelerden gelen sinyalleri kendi dendriti üzerinden alır, bu sinyalleri birleştirerek eğer belli bir eşik değeri aşıyorsa, yine elektriksel bir sinyal olarak kendine bağlı diğer hücelere aksonu üzerinden gönderir. Eğer eşik değeri

aşmıyorsa diğer hücrelere sinyal göndermez. Bu nedenle nöronların ya hep ya hiç karakteri olduğu söylenir (Gurney, 1997).

Beyin değişikliklere son derece uyumlu (adaptive) bir organdır. Bu nedenle beyin için *nöroplastiklik* tanımı kullanılır. Beyin çalışma sistemindeki bu esneklik, beynin hem çevresine kolaylıkla uyum sağlamasını, hem de beyin zedelenmelerinde zedelenen bölgenin yaptığı işlerin sağlıklı bölgelere kaydırılabilmesini sağlar (Rogers, 2011). Ayrıca beyinde çeşitli nedenlerden ölen hücrelerin görevlerini de diğer hücreler üstlenmektedir. Bu görev değişikliği yapılan kadar geçen zamanda beyin işlevlerinde performans kaybı olsa da duraksama olmaz.

### **3.2.1.2 Yapay sinir ağları tarihçe ve tanım**

Beynin öğrenmesi ve bununla ilgili nöronların işlevinin matematiksel bir model olarak ortaya konması ise McCulloch ve Pitts'in 1943 yılındaki çalışması ile olmuştur (Haykin, 1999). Bu çalışmada basit bir sinir ağı teorik olarak elektronik devrelerle modellenmiştir. İlerleyen yıllarda John Von Neumann basit bir nöronu vakumlu tüplerle taklit edilebileceğini bir öneri olarak ortaya atmıştır ve bununla ilgili çalışmalar yapmıştır (Anderson ve McNeil, 1992).

Daha sonra Rosenblatt ve nöro-biyolojist Cornell, 1958 yılında algılayıcı modelini geliştirmişlerdir. Algılayıcı tek katmanlı bir YSA'dır.

1960 yılında Bernard Widrow ve Marcian Ted Hoff Adaline ve Madaline modellerini tanıttılar. Adaline modelinin algılayıcıdan farkı öğrenme yöntemindedir.

Yapay sinir ağlarının sürekli olarak geliştiği ve beklentilerin arttığı bu yıllarda, bu modelleme sisteminin üstüne fazlasıyla ağır bir görev bindirilmiştir. Bu alanda çalışan bilim adamları YSA ile bilgisayarların insan gibi yürüyen, konuşan, gören, yazan, kendini yeniden üreten ve varlığının bilincinde olan makineler olacağını öne sürmekteydi (Olazaran, 1996). Özellikle o zamanki teknolojinin de yetersizliği nedeniyle bu beklentiler karşılanamamıştır. Bu beklentinin gerçek olma olasılığı ise bir kısım bilim insanlarında da tedirginlik yaratmıştır (Anderson ve McNeil, 1992).

Özellikle algılayıcı modeli çok büyük ümitler vadetmesine rağmen modelin çoğu veri setini doğru bir şekilde sınıflayamadığı görülmüştür. Şu var ki, algılayıcı modeli ancak doğrusal ayrılabilir verileri etkin bir biçimde sınıflayabilir.

1970 yılında YSA'ların, o zamanki modeller göz önüne alındığında, doğrusal modelleme yapamadığı Marvin Minsky ve Seymour Papert'in Algılayıcılar adlı kitabında gösterilmiştir. YSA'nın doğrusal modelleme yapamayacağının kanıtlanması YSA çalışmalarına ayrılan kaynakların azalmasına ve bu alandaki çalışmaların neredeyse durma noktasına gelmesine neden olmuştur. Bundan sonra konu üzerine az sayıda bilim insanı çalışmalara devam etmiştir. Kısıtlı kaynak ve az sayıda bilim insanının katkıları ile yapılan çalışmaların sonuçları ancak bir on yıl sonra meyvelerini vermeye başlamıştır. YSA'daki ilk önemli çalışmaların 1980'li yıllarda çıktığı görülür.

1982 yılında John Hopfield'in yayımladığı makale bu alana olan ilginin yeniden artmasını sağlamıştır. Hopfield bu çalışması için yaklaşımının beyni basitçe modellemekten ziyade çalışan yararlı makineler yapmak olduğunu vurgulamıştır (Anderson ve McNeil, 1992). Bu çalışmadan sonra bu alanda yapılacaklar için yeniden kaynak sağlanması ve YSA'nın yine eskisi gibi popüler olması YSA'nın yeniden doğması olarak nitelendirilir.

1970 yılından sonraki çalışmalar şu şekilde sıralanabilir (Öztemel, 2012):

- 1969-1972 doğrusal ilişkilendiricilerin geliştirilmesi
- 1972 korelasyon matris belleğinin geliştirilmesi
- 1974 Geriye yayılım modelinin geliştirilmesi (Çok katmanlı algılayıcının ilk çalışmalarının yapılması)
- 1978-1982 öğretmensiz öğrenmenin geliştirilmesi
- 1982 Hopfield ağlarının geliştirilmesi
- 1984 Boltzmann makinesinin geliştirilmesi
- 1985 Çok katmanlı algılayıcıların delta öğrenme kuralı ile öğrenmesi
- 1988 RBF modelinin geliştirilmesi
- 1988 PNN modelinin geliştirilmesi
- 1991 GRNN modelinin geliştirilmesi

1991'den günümüze YSA popülaritesi kaybetmemiş, akademide aktif bir çalışma alanı ve endüstride de birçok alanda kullanılan bir modelleme aracı olmuştur.

Tüm bu anlatılanlar ışığında YSA'nın basit bir tanımı şu şekilde yapılabilir: Yapay sinir ağları, yapısı ve çalışma prensipleri biyolojik sinir ağlarının yapısına ve çalışma prensiplerine göre temellendirilmiş çok yönlü bir modelleme aracıdır.

### 3.2.1.3 Biyolojik ve yapay sinir ağlarının karşılaştırılması

Yapay sinir ağları çözüme ihtiyaç duyulan sorunun cevabını biyolojik sinir ağlarının işleyişini taklit ederek bulmaktadır. Öyle ki, yapay sinir ağlarında da nöron tanımlanmış ve YSA da nöronların birbiri ile varsayımsal bağları ile oluşturulmaktadır. Biyolojik sinir hücresindeki elemanlar yapay sinir ağlarında matematiksel fonksiyon ve terimlere karşılık gelmektedir. Kabaca bir karşılaştırma şu şekilde olabilir:

Girdiler: BSA'da diğer hücrenin aksonlarından gelen sinyallerine karşılık gelmektedir. Bir yapay sinir hücresine gelen bilgilere karşılık gelmektedir. Yapay sinir hücresinde dış dünyadan olduğu kadar kendinden de bilgi gelebilir (Öztemel, 2012).

Ağırlıklar: Sinapslara karşılık gelmektedir. YSA'daki ağırlıklar ise bir nörona gelen reel değerlerin çarpım işlemi ile güncellendiği terimlerdir. Ağın öğrendiği bilgi ağırlıklarında saklanmaktadır. Ağırlıkların büyük-küçük ya da pozitif-negatif olması problem hakkında yorumlanabilir bir bilgi vermez.

Birleştirme: BSA'da dendritlere karşılık gelmektedir. Hücreye gelen girdiyi çeşitli matematiksel fonksiyonlar ile birleştirir.

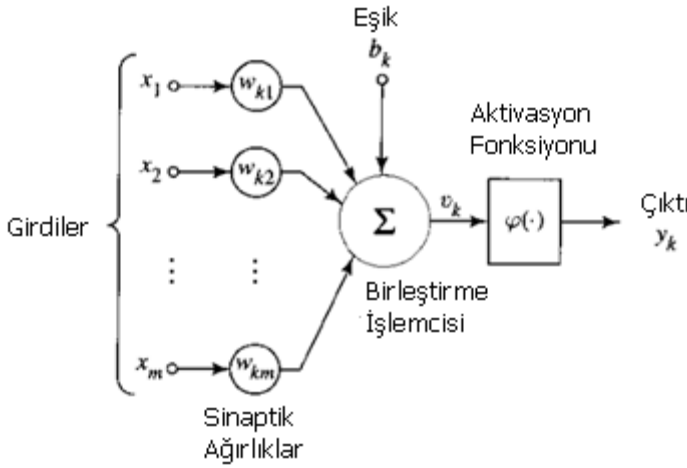
Aktivasyon fonksiyonu: BSA'da hücre gövdesine karşılık gelir. Birleştirilmiş girdi bu fonksiyona gönderilir.

Çıktılar: BSA'da aksonlara karşılık gelir. Aktivasyon fonksiyonundan çıkan değerdir. Bu değer nöronun çıktı değeri olmaktadır. Yeniden oluşumlu ağlarda hücre kendi çıktı değerini tekrar kullanmak üzere girdi olarak alabilir.



Eşik değeri: Eşik değeri matematiksel işlemlerde gerekli bir elemandır. Eşik değeri transfer fonksiyonunun çıktısını daha uygun çözümler vermesi için sisteme dâhil edilir. Zorunlu bir eleman değildir. Eşik değeri girdisi her zaman birdir ilgili ağırlığın ise değişimine izin verilir.

Bir nöron birden çok nöron ile bağlantılı olabilir, bu özelliğe sahip bir nöron yapısı Şekil 3.10.'da verilmiştir:



Şekil 3.10. Nöronun yapısı

### 3.2.1.4 Modelin katmanları

BSA'da katman kavramı bulunmamasına rağmen, YSA modelleri katmanlar ile bu katmanlar içinde birbirine bağlı nöronlar olarak ele alınmaktadır. YSA'da işlevleri bakımından farklı 3 katman tanımlanmıştır.

#### 3.2.1.4.1 Giriş katmanı

Giriş katmanı her modelde sadece bir tane bulunmaktadır. Bu katmanda bilgi değişmeden bir sonraki katmandaki nöronlara gönderilir. Giriş katmanındaki nöron sayısı modeldeki girdi sayısına karşılık gelir.

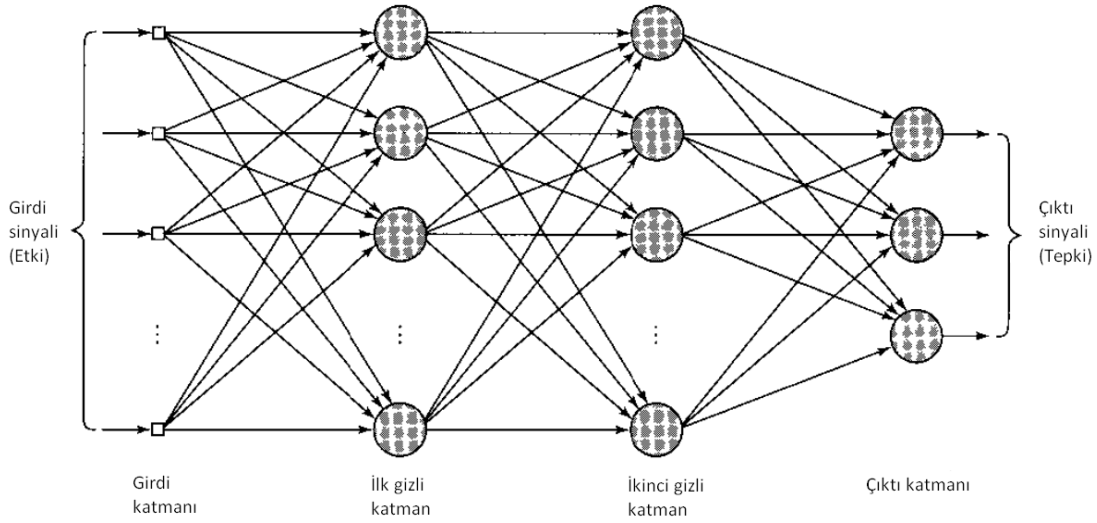
#### **3.2.1.4.2 Gizli katman**

Gizli katman bazı YSA modellerinde bulunmamaktadır, bazı modellerde ise problemin yapısına göre 1'den fazla seçilebilmektedir. Doğrusal olmayan modelleme yapmak için bu katmandan en az 1 tane olması gerekmektedir. 4 katmanlı bir YSA'nın karmaşıklık düzeyi ne olursa olsun tüm problemleri çözebildiği, katman sayısının 4'ten fazla olmasının ise modele herhangi avantaj getirmediği gösterilmiştir. İlk beklentinin aksine katman sayısı arttıkça hem modeldeki hesaplamalar zorlaşacak, hem model ezberlemeye daha meyilli olacaktır (Anderson ve McNeil, 1992). Bu katmanda bir önceki katmandan gelen veriler ağırlıklandırılarak birleştirilip aktivasyon fonksiyonundan geçirilir ve bir sonraki katmana gönderilir.

#### **3.2.1.4.3 Çıktı katmanı**

Çıktı katmanı her modelde sadece bir tane bulunmaktadır. Burada da ara katmandan gelen veri ağırlıklandırılarak kullanılır ve aktivasyon fonksiyonu çıktıları düğümlerin çıktısı olur.

YSA'da tüm katmanlarda probleme göre farklılaşmak üzere bir ya da daha fazla sayıda düğüm bulunmaktadır (Şekil 3.11.).








Şekil 3.11. Modelin katmanları

### 3.2.1.5 YSA modelinin karar amaçlı kullanımı

Veriler ya da veri seti modele girdi katmanından girer, sonuç ise çıktı seti olarak çıktı katmanından elde edilir. Gizli ve çıktı katmanda genellikle bir düğüme birden fazla düğümden bilgi gelmektedir. Bu katmanlardaki her bir düğüm bu değerleri birleştirici işlemden geçirir bu işlemi yaparken eğer varsa eşik de bu birleşime katılır. Birleştirme işlemi için yaygın olarak toplam fonksiyonu kullanılsa da teoride toplama fonksiyonu yerine çarpma fonksiyonu, işaret fonksiyonu, maks/min fonksiyonları ve hatta ve/veya operatörleri de kullanılabilir (Anderson ve McNeil, 1992). Diğer düğümlerden gelen ve birleştirilen bilgi aktivasyon fonksiyonuna gönderilir. Aktivasyon fonksiyonu sonraki hücreye gönderilecek bilginin nihai olarak işlendiği elemandır. Aktivasyon fonksiyonundan çıkan bilgi düğümün çıktısıdır. Bu da ya modelin çıktısıdır ya bir sonraki katmandaki düğümlere giden girdi değeridir. Yeniden oluşumlu ağlarda bir düğümden çıkan çıktı değeri, yine aynı düğüme girdi olarak gelebilir. Doğrusal olmayan modelleme yapabilmek için aktivasyon fonksiyonu da doğrusal olmayan bir fonksiyon olarak seçilmelidir.

Aktivasyon fonksiyonları çözümü yapılacak probleme göre farklılık gösterebilmektedir. Sigmoid en fazla kullanılan aktivasyon fonksiyonu olmak üzere hiperbolik tanjant, lineer, işaret ve sinüs fonksiyonlarının da kullanıldığı bilinmektedir (Anderson ve McNeil, 1992). Sık

kullanılan aktivasyon fonksiyonları Şekil 3.12.'de verilmiştir ([http://sebastianraschka.com/Articles/2015\\_singlelayer\\_neurons.html](http://sebastianraschka.com/Articles/2015_singlelayer_neurons.html)).

	Adım	$g(z) = \begin{cases} 1 & \text{if } z \geq 0 \\ -1 & \text{d.d.} \end{cases}$
		$g(z) = \begin{cases} 1 & \text{if } z \geq 0 \\ 0 & \text{d.d.} \end{cases}$
	Doğrusal	$g(z) = z$
	Lojistik (sigmoid)	$g(z) = 1 / (1 + \exp(-z))$
	Hiperbolik Tanjant	$g(z) = \frac{\exp(2z) - 1}{\exp(2z) + 1}$

Şekil 3.12. Sık kullanılan aktivasyon fonksiyonları

### 3.2.1.6 Modelin öğrenmesi

Modelin öğrenmesi ağırlıkların değiştirilmesi ile olur. Zaman içinde farklı problem çözümleri için farklı YSA modelleri geliştirilmiş bu YSA modelleri için de farklı öğrenme algoritmaları kullanılır olmuştur. Öğrenme yöntemlerinden bazıları şu şekildedir (Öztemel, 2012):

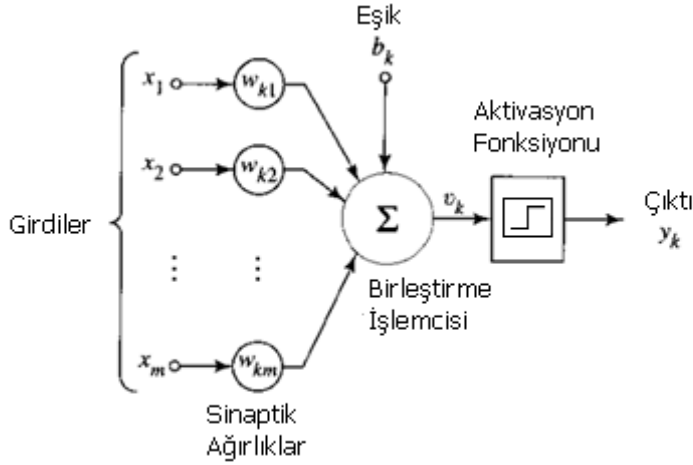
- Hebb kuralı
- Delta kuralı
- Hopfield kuralı
- Kohonen öğrenme kuralı
- Genelleştirilmiş delta kuralı

### 3.2.2 Bazı YSA modelleri

Çeşitli problemler için farklı YSA modelleri zaman içinde geliştirilmiştir. İlk YSA modelleri daha basit bir yapıya sahipken ilerleyen zamanlarda modellerin daha karmaşıklaştığı ve öğrenme algoritmalarının da buna paralel olarak değiştiği görülmektedir. Literatürde çok sayıda YSA modeli bulunmaktadır. En temel YSA modelleri aşağıdaki alt başlıklarda verilmiştir.

### 3.2.2.1 Algılayıcı

Algılayıcı (Şekil 3.13.), Rosenblatt ve diğerleri tarafından geliştirilmiş ve 1958'de tanıtılmıştır.



Şekil 3.13. Algılayıcı - nöronun yapısı

Algılayıcıda birden çok girdi olabilmekte fakat yalnız bir çıktı olmaktadır. Algılayıcının aktivasyon fonksiyonu birim adım fonksiyonudur.

Çok sayıda girdi olabileceği düşünüldüğünde ağırlıklar, girdi ve hedef çıktı değerleri sırası ile şu şekilde tanımlanabilir:

$$\mathbf{w}^T = [w_0, w_1, w_2, \dots, w_m], \mathbf{x}^T = [x_0, x_1, x_2, \dots, x_m] \text{ ve } t \quad (3.73)$$

Ağın diğer elemanları aşağıdaki gibidir:

Aktivasyon fonksiyonu: Birim adım fonksiyonu (*baf*)

$$\text{Girdi toplamı: } net = \mathbf{w}^T \mathbf{x}$$

(3.74)

$$\text{Nöron çıktısı: } o = baf(net)$$

Hata:  $e_i = t_i - o_i$

Güncelleme kuralı ise

$$w^{yeni} = w^{eski} + \Delta w$$

$$b^{yeni} = b^{eski} + \Delta b$$

(3.75)

$$\Delta w = \eta e_i x_i$$

$$\Delta b = \eta e_i$$

şeklinde. Tüm verilerin doğrusal ayrılabilir olduğu varsayımı altında algoritma şu şekildedir (Hamzaçebi, 2011):

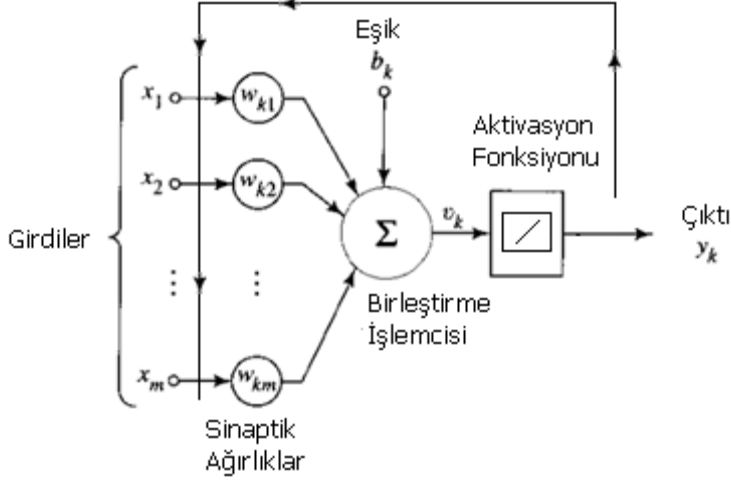
1. Bağlantılar için rastgele başlangıç değerleri ata,  $0 < \eta \leq 1$  olmak üzere  $\eta$  değerini belirle
2. Girdi kümesinden bir girdi vektörü seç
3. Nöronun çıktısını hesapla
4. Hedef çıktı ile nöron çıktısı arasındaki farkı kullanarak algılayıcı öğrenme kuralına göre ağırlık ve eşik değerlerini güncelle:
  - a.  $w^{yeni} = w^{eski} + \Delta w$
  - b.  $b^{yeni} = b^{eski} + \Delta b$
5. Verilerin tamamı doğru sınıfa etiketleninceye kadar 2 adımına git, tamamı doğru sınıflandırılmışsa dur.

Basit algılayıcılar ilk öğretmenli öğrenme algoritması olması açısından önemlidir. Basit algılayıcıların diğer bir önemli özelliği ise daha gelişmiş YSA algoritmalarına temel oluşturmasıdır.

### 3.2.2.2 Adaline

Adaline (Şekil 3.14.) yani, *Adaptive Linear Neuron* ya da sonraki kullanımı ile *Adaptive Linear Element*, Bernard Widrow ve Ted Hoff tarafından 1960 yılında ortaya atılmıştır (Gurney,

1997). Adaline tek katmanlı bir ağıdır. Endüstride ilk kullanılan yapay sinir ağı olmasından dolayı önemlidir (Anderson ve McNeil, 1992).



Şekil 3.14. Adaline - nöronun yapısı

Adaline'da birden çok girdi olabilmekte fakat tek çıktı olmaktadır (Fausett, 1993). Adaline algoritması, *delta kuralı* olarak da bilinir. Ağırlıklar, girdi sırası ve hedef çıktı değerleri basit algılayıcıdaki gibi tanımlanmıştır:

$$\mathbf{w}^T = [w_0, w_1, w_2, \dots, w_m], \mathbf{x}^T = [x_0, x_1, x_2, \dots, x_m], \mathbf{t} \quad (3.76)$$

Ağın diğer elemanları şu şekildedir:

Aktivasyon fonksiyonu: birim fonksiyonu

$$\text{Girdi toplamı: } net = \mathbf{w}^T \mathbf{x} \quad (3.77)$$

$$\text{Nöron çıktısı: } o_i = net$$

$$\text{Hata: } e_i = t_i - o_i$$

Güncelleme kuralı ise

$$w^{yeni} = w^{eski} + \Delta w$$

$$b^{yeni} = b^{eski} + \Delta b$$

(3.78)

$$\Delta w = \eta e_i x_i$$

$$\Delta b = \eta e_i$$

şeklindedir. Bu modelin basit algılayıcıdan farkı öğrenme sırasında ağırlıkların aktivasyon fonksiyonuna gitmeden, ağırlıklandırılmış toplama göre ayarlanmasıdır (Hamzaçebi, 2011).

#### Delta kuralı (en küçük kareler kuralı):

1. Bağlantılar için rastgele başlangıç değerleri ata,  $0 < \eta \leq 1$  olmak üzere  $\eta$  değerini belirle
2. Girdi kümesinden bir girdi vektörü seç
3. Nöronun çıktısını hesapla
4. Hedef çıktı ile nöron çıktısı arasındaki farkı kullanarak algılayıcı öğrenme kuralına göre ağırlık ve eşik değerlerini güncelle:
  - a.  $w^{yeni} = w^{eski} + \Delta w$
  - b.  $b^{yeni} = b^{eski} + \Delta b$
5. Bütün girdi çıktı eşleştirmeleri baştan belirlenmiş olan en küçük ortalama kareler değerinden çok ise adım 2'ye git, az ise nöron çıktıları birleştiriciden çıkan değerler ile iki kutuplu adım fonksiyonu (ikaf) kullanarak bul.

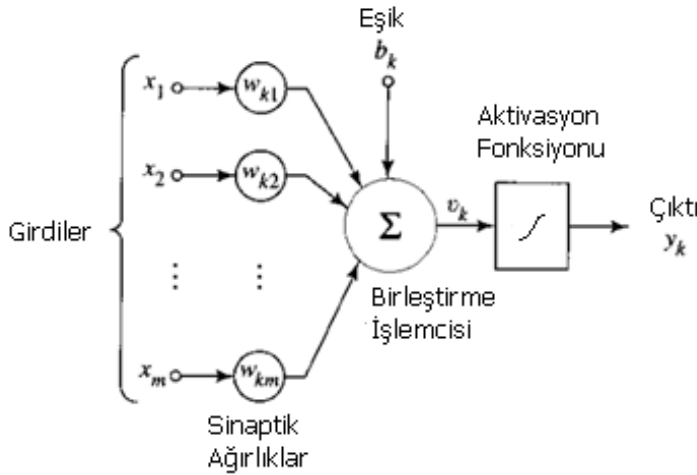
#### **3.2.2.3 Çok katmanlı algılayıcı**

Çok katmanlı algılayıcı modeli 1982 yılında geliştirilmiş, 1986 yılında da Rumelhart, Hinton, Williams tarafından geriye yayılım algoritması (genelleştirilmiş delta kuralı) ünlenmiştir. Tarihsel olarak bakıldığında aynı algoritma Parker ve LeCun tarafından bağımsız olarak 1985 yılında ve 1974 yılında Werbos'un doktora tezinde öne sürülmüştür.



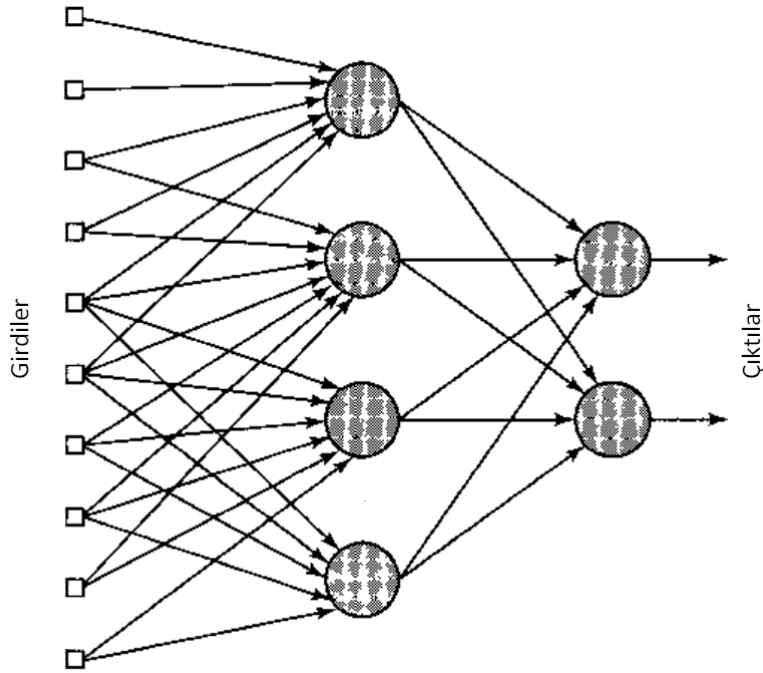
Çok katmanlı algılayıcı doğrusal olmayan modelleme yapabilmektedir bu yüzden çok tercih edilen bir model olmuştur. Ayrıca ÇKA evrensel bir t.e. olarak değerlendirilmektedir. ÇKA'nın şu karakteristikleri göze çarpmaktadır (Haykin, 1999):

- Modeldeki her bir nöron doğrusal olmayan ve türevlenebilir bir aktivasyon fonksiyonuna sahiptir. Bu fonksiyon genelde lojistik fonksiyonu olarak seçilir(Şekil 3.15.).
- Ağ bir ya da daha fazla gizli katman içerir. Ağın karmaşık ve doğrusal olmayan modelleme yapması bu katman ile olur.
- Ağdaki çözüm sinaptik bağlantıların sayılarının ya da ağırlıklarının değişimi ile gerçekleşir.



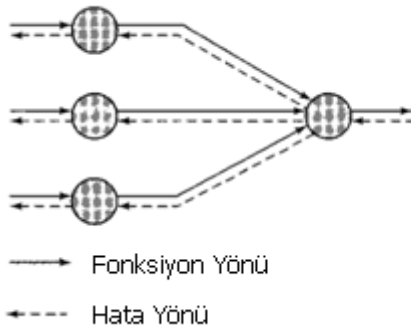
Şekil 3.15. ÇKA - nöronun yapısı

ÇKA tam bağı bir katmanlı bir ağıdır. Çok katmanlı algılayıcılarda en az 1 tane gizli katman olmaktadır (Şekil 3.16.). Doğrusal olmadan modelleme yapma yeteneği gizli katmandaki doğrusal olmayan nöronlar sayesinde yakalanabilmektedir (Samarasinghe, 2006).



Şekil 3.16. Tam bağlı, ileri yönlü, tek gizli katmanlı YSA

ÇKA'nın çalışması bundan önce anlatılan ağlarda olduğu gibidir. Model çıktısı girdilerin işlenmesi yönünde olur. Modelin öğrenmesi ise tam ters yöndeki veri akışı ile olmaktadır (Şekil 3.17.).

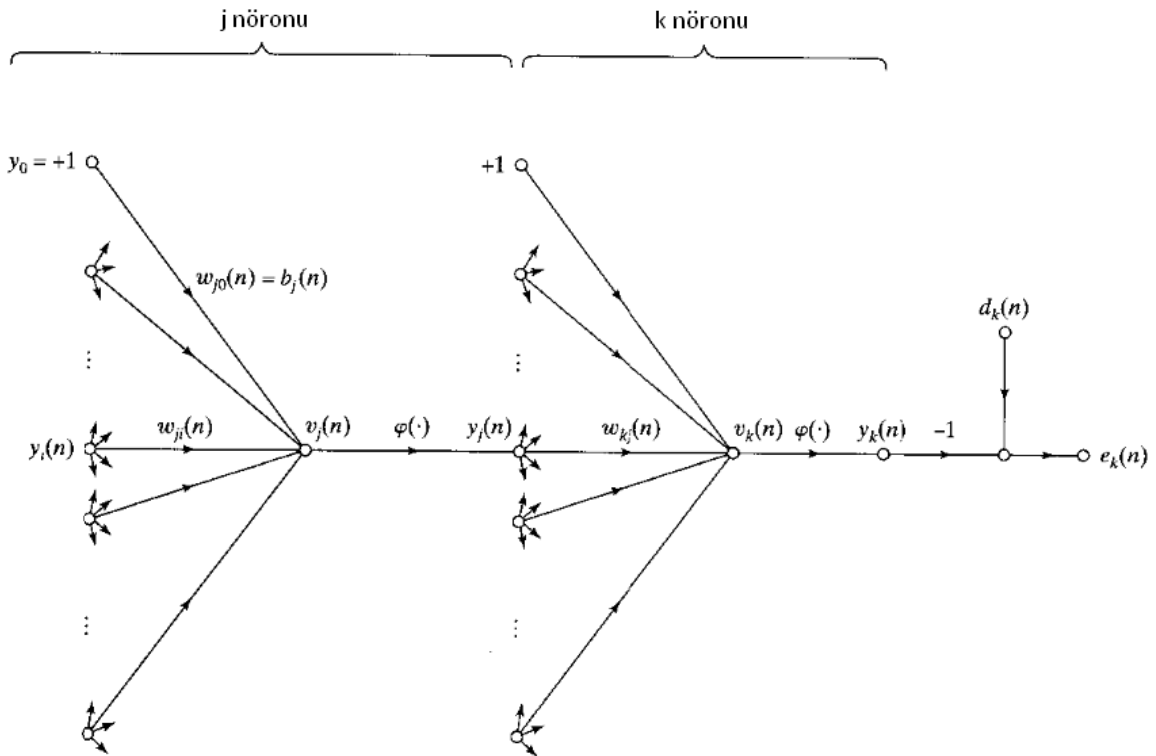


Şekil 3.17. Girdilerin işlenmesi ile veri akışı yönü

Girdi katmanındaki nöronlar veri üzerine bir işleme yapmadan verileri doğrudan gizli katmana gönderirler. Gizli katmandaki nöronlara gelen bu değerler birleştirici fonksiyondan, sonrasında da aktivasyon fonksiyonundan geçerler. Aynı şekilde gizli katmandaki nöronlarca işlenen bilgiler çıktı katmanına gönderilir. Eğitim işlemi yani geri yayılma ise tam ters yönde olur.

ÇKA öğrenmesi için kullanılan pek çok metot vardır: Geri yayılım algoritması (GYA), rassal arama, genetik algoritmalar, tavlama benzetimi başka sezgisel optimizasyon metotları bulunmaktadır (Hamzaçebi, 2011). Fakat bunların arasında en popüler öğrenme yöntemi GYA'dır (Şekil 3.18). GYA kullanım kolaylığı ile ön plana çıkmaktadır.

### Geri yayılım algoritması:



Şekil 3.18. Geri yayılım algoritması

Geri yayılım algoritması herhangi bir sayıda katmanda kullanılabilir olmasına rağmen algoritmanın nasıl çalıştığını göstermek için girdi katmanı ile birlikte 3 katmanlı bir ağ ile çalışmak yeterli olur (Şekil 3.18). Ayrıca ağ için aktivasyon fonksiyonlarının sigmoid olarak seçilmesi yaygın bir yaklaşımdır.

Geri yayılım algoritmasını ele alabilmek için öncelikle bazı ek tanımlamalar yapmakta yarar vardır:

$w_{ij}$ ,  $x_{ij}$  sırası ile  $i$  düğümünden  $j$  düğümüne olan ağırlıkları ve girdileri ve  $m$  de toplam girdi sayısını göstermek üzere,

$$net_j = \mathbf{w}^T \mathbf{x} = \sum_{i=0}^m \omega_{ij} x_{ij},$$

$$o_j = S(net_j) \quad (3.79)$$

$t_j = hedef\ çıktı\ değeri$

olarak tanımlansın. Bir ÇKA modelinde çıktı katmanındaki bir  $j$  nöronun  $n$ . yinelemedeki hata değeri için

$$e_j(n) = t_j(n) - o_j(n) \quad (3.80)$$

yazılabilir.  $C$  çıktı katmanındaki tüm nöronlar olmak üzere hata fonksiyonunu şu şekilde tanımlansın:

$$E(n) = \sum_{j \in C} \frac{e_j^2(n)}{2} \quad (3.81)$$

GYA'da da amaç çıktı hatasını minimize etmektedir. Toplam hata ise 1 ya da 1'den fazla örnek için hesaplanabilir. Batch öğrenme adı altında tüm örnekler sisteme gösterildikten sonra da hesaplanabilir. Hata minimizasyonu genellikle *eğim iniş* (gradient descent) metodu ile yapılır. Eğim iniş metodu her seferinde  $\omega_{ij}$ 'in  $\Delta\omega_{ij}(n)$  kadar düzelmesini sağlar. Hatanın azalması  $\omega_{ij}$  üzerinden olacağından her bir düğüm için

$$\frac{\partial E(n)}{\partial \omega_{ij}(n)} \quad (3.82)$$

değerinin bulunması gerekmektedir. Bu değer eğim iniş algoritmasındaki  $\Delta\omega_{ij}(n)$  değerinin oransal olarak elde edilmesini sağlar. Genel bir şekilde yazmak gerekirse delta kuralı şu şekilde ifade edilir:

$$\delta_k = o_k(1 - o_k)(t_k - o_k) \quad (3.83)$$

$$\delta_j = o_j(1 - o_j)\sum \omega_{jk}\delta_k$$

Yukarıdaki satırlarda verilen bilgileri ve türevin zincir kuralını kullanarak diferansiyel denklem şu şekilde yazılabilir:

$$\frac{\partial E(n)}{\partial \omega_{ij}(n)} = \frac{\partial E(n)}{\partial e_j(n)} \frac{\partial e_j(n)}{\partial o_j(n)} \frac{\partial o_j(n)}{\partial net_j(n)} \frac{\partial net_j(n)}{\partial \omega_{ij}(n)} \quad (3.84)$$

Burada

$$\frac{\partial E(n)}{\partial e_j(n)} = e_j(n) = t_j(n) - o_j(n)$$

$$\frac{\partial e_j(n)}{\partial o_j(n)} = -1$$

$$\frac{\partial o_j(n)}{\partial net_j(n)} = o_j(n)(1 - o_j(n))$$

$$\frac{\partial net_j(n)}{\partial \omega_{ij}(n)} = x_{ij}(n)$$

(3.85)

olmaktadır. Bu durumda

$$\frac{\partial E(n)}{\partial \omega_{ij}(n)} = -(t_j(n) - o_j(n))o_j(n)(1 - o_j(n))x_{ij}(n) \quad (3.86)$$

şeklinde bir ifade elde edilir.

Hata azaltma delta kuralına göre yapılmaktadır. Buna göre hata azaltma şu değere eşittir:

$$\Delta \omega_{ij}(n) = \eta \delta_j(n) x_{ij}(n) \quad (3.87)$$

Burada  $\eta$  değeri de geriye yayılım algoritmasındaki öğrenme oranına denk gelmektedir.

$\delta_j(n)$  yerel eğim olarak adlandırılır ve şu şekilde tanımlanmıştır:

$$\delta_j(n) = -\frac{\partial E(n)}{\partial net_j(n)} = e_j(n)o_j(n)(1 - o_j(n)) \quad (3.88)$$

Yerel eğim sinaptik ağırlıklardaki değişimin yönünü işaret eder. Son eşitlikten görüldüğü gibi bir  $j$  çıktı nöronu için bu değer ilgili nörondaki hata değeri  $e_j(n)$  ile o nöronun aktivasyon fonksiyonunun türevine karşılık gelmektedir. Burada iki farklı durum söz konusu olacaktır.

Eğer  $j$  nöronu çıktı katmanında olan bir nöron ise durum daha basittir çünkü bu durumda geri yayılma nöronun sadece kendi üzerinden olacaktır. Bu durumda yukarıdaki eşitlikler geçerli olur.

*Eğer  $j$  nöronu gizli katmanda ise  $j$  nöronunun hata değeri, bu nöronun doğrudan bağlantılı bulunduğu diğer tüm nöronların yinelemeli olarak belirlenmesi ile bulunabilir. Bunun için şu şekilde bir yol izlenebilir:*

$$\delta_j(n) = -\frac{\partial E(n)}{\partial o_j(n)} \frac{\partial o_j(n)}{\partial net_j(n)} \quad (3.89)$$

$\frac{\partial E(n)}{\partial o_j(n)}$  için şu şekilde bir yol izlenebilir:

$$E(n) = \sum_{k \in C} \frac{e_k^2(n)}{2} \quad (3.90)$$

olmak üzere,

$$\frac{\partial E(n)}{\partial o_j(n)} = \sum_{k \in C} e_k(n) \frac{\partial e_k(n)}{\partial o_j(n)} \quad (3.91)$$

elde edilir.

$$\frac{\partial E(n)}{\partial o_j(n)} = \sum_{k \in C} e_k(n) \frac{\partial e_k(n)}{\partial net_k(n)} \frac{\partial net_k(n)}{\partial o_j(n)} \quad (3.92)$$

$$e_k(n) = t_k(n) - o_k(n) \quad (3.93)$$

olduğunu bilinmektedir. Böylelikle

$$\frac{\partial e_k(n)}{\partial net_k(n)} = -o_k(n)(1 - o_k(n)) \quad (3.94)$$

ayrıca  $k$  nöronu için,  $m$  toplam girdi sayısı olmak üzere

$$net_k(n) = \sum_{j=0}^m \omega_{jk}(n)x_{jk}(n) \quad (3.95)$$

$o_j(n)$ 'e göre türev alındığında

$$\frac{\partial net_k(n)}{\partial o_j(n)} = \omega_{jk}(n) \quad (3.96)$$

olur.

$$\frac{\partial E(n)}{\partial o_j(n)} = -\sum_{k \in C} e_k(n)o_k(n)(1 - o_k(n))\omega_{jk}(n) = -\sum_{k \in C} \delta_k(n)w_{kj}(n) \quad (3.97)$$

Son olarak  $\delta_j(n)$  için geri yayılma formülü

$$\delta_j(n) = o_j(n)(1 - o_j(n)) \sum_{k \in C} \delta_k(n)w_{jk}(n) \quad (3.98)$$

Bu sonuçları kullanmak üzere ağırlıktaki değişim şu şekilde yapılmaktadır:

$$\Delta \omega_{ij}(n) = \eta \delta_j(n) o_i(n) \quad (3.99)$$

Zaman serileri analizinde daha çok ileri beslemeli geri yayımlı çok katmanlı algılayıcılar kullanıldığı bilinmektedir.

Bu çalışmada detaylandırılmayan diğer yapay sinir ağları şöyle sıralanabilir:

- Vektör kuantizasyon modelleri
- Kendi kendini organize eden model
- Adaptif rezonans teorisi modelleri
- Hopfield ağları
- Counter propogation ağı

- Neocognitron ađı
- Boltzman makinesi
- Olasılıksal ađlar
- Elman ađı
- Radyal tabanlı ađlar

### 3.2.3 YSA Türleri

YSA tipine, öğrenme yöntemine, mimarisine göre sınıflandırılabilir.

#### 3.2.3.1 Tipine göre YSA türleri

##### 3.2.3.1.1 İleri ve Geri beslemeli ađlar

Verilerin girdi birimlerinden çıktı birimlerine doğru sadece ileri doğru gönderildiđi ađlar ileri beslemeli ađlardır. Veri akışının geriye doğru da gidebildiđi ađlar geri beslemeli ađlardır.

#### 3.2.3.2 Öğrenme yöntemlerine göre YSA türleri

##### 3.2.3.2.1 Öğretmenli öğrenme

Sisteme girdilerle birlikte bunlara karşılık gelen çıktılar da verilebiliyorsa bu öğretmenli öğrenmedir. Sistemin ana stratejisi tanımlanan hata fonksiyonunun en az değeri veriyor olmasıdır.

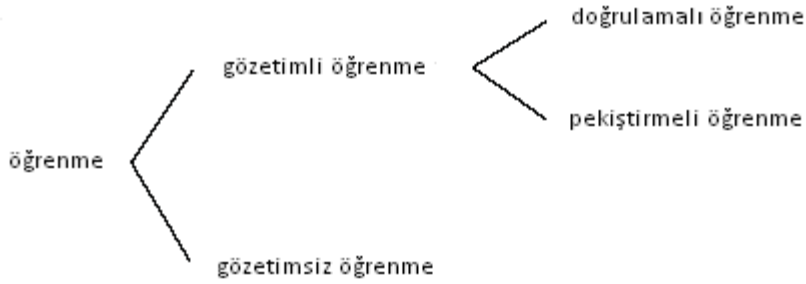
##### 3.2.3.2.2 Öğretmensiz öğrenme

Sisteme sadece girdiler verilir. Sistemin bu şekilde sonuç vermesi beklenir. Bazı tip problemlerde mesela kümeleme analizinde ya da boyut küçültme analizinde bu tür bir strateji izlenir. Bu tür öğrenmede amaç girdideki düzenlilikleri bulmaktır (Alpaydın, 2011).



### 3.2.3.2.3 Destekleyici öğrenme

Destekleyici öğrenmede sisteme girdiler verilir çıktılar verilmez fakat üretilen sonuçlara göre ağa olumlu ya da olumsuz sinyaller gönderilerek ağ ağırlıklarının düzenlenmesi sağlanır. Öğrenme yöntemlerini sınıflandıran iyi bir görsel Şekil 3.19.'te görülmektedir (Rojas, 1996):



Şekil 3.19. Öğrenme yöntemleri

### 3.2.3.3 Mimarisine (topolojisine) göre ağlar

YSA'lar verinin işlendiği katman sayılarına göre eğer bir adet katmana sahip ise tek katmanlı, çok katmanlı, yeniden oluşumlu ağlar olmak üzere üçe ayrılabilir (Haykin, 1999).

### 3.2.4 YSA Kullanım Özellikleri

#### 3.2.4.1 Özellikleri ve avantajları

YSA çeşitli özelliklerine bu başlık altında değinilmiştir.

##### 3.2.4.1.1 Doğrusal olmayan modelleme yapma yeteneği

Doğrusal olmayan modelleme yapma yeteneği YSA'nın en güçlü özelliklerinden bir tanesidir. YSA, bir benzeşim yapmak gerekirse, parametrik ve doğrusal olmayan regresyon modellerine karşılık gelmektedir. Öyle ki en az bir gizli katmanlı bir YSA teorik olarak her dereceden ve boyuttan fonksiyon parametrelerini, istenilen doğruluk düzeyinde tahmin edebilir. Bu yüzden YSA kestiricileri için evrensel tahmin ediciler denilmektedir. Bu özelliği

ile çok deęişkenli yöntemlerden çok deęişkenli doğrusal ve doğrusal olmayan regresyon modellerinin genişletilmiş yöntemleri olarak düşünülebilir (Samarasinghe, 2006).

Gerçek hayatta karşılaşılan çok sayıda problemin doğrusal olmadığı göz önüne alındığında bu YSA için son derece önemli bir özelliktir.

#### **3.2.4.1.2 Paralel çalışma**

YSA'nın sadece yazılımda değil donanımsal olarak da modellenebildiği bilinmektedir. Her bir nöron ise oldukça basit işlemlerle çalışmaktadır. Bu şekilde hem donanım hem yazılım maliyeti düşük ürünler üretmek mümkün olmaktadır.

#### **3.2.4.1.3 Az varsayım ile kullanılma**

YSA'nın diğer bir önemli özelliği modelleme için az sayıda varsayımın yeterli olmasıdır. Bu özellik de hem modellemeyi kolaylaştırmış hem de YSA'yı en tercih edilir modellerden biri haline getirmiştir.

Bu kadar özelliği bir arada sunan YSA modellerinin birçok alanda kullanılıyor olması bu nedenle çok olağandır. YSA örüntü tamamlama, sınıflandırma, optimizasyon, veri sıkıştırma, fonksiyon yakınlaştırma, ilişkilendirme, öngörü ve kontrol problemlerinde başarı ile kullanılmaktadır (Fyfe, 2000). YSA endüstride kullanılmaya uzun zaman önce başlamıştır.

Çizelge 3.1. YSA ile tahmin ve öngörü hizmeti veren bir şirketin internet sitesinden alınmıştır (<http://www.alyuda.com/products/forecaster/neural-network-applications.htm>, 10.12.2015). Tabloda görünen ana başlıklar altında sayısız uygulama hem akademide hem endüstride hayat bulmuştur ve hala bulmaktadır.



Çizelge 3.1. YSA ile endüstride yapılan öngörü çalışmaları

Finans	<ul style="list-style-type: none"> <li>Borsa öngörüsü</li> <li>Kredi araştırması</li> <li>Kredi derecelendirme</li> <li>İflas öngörüsü</li> <li>Mülkiyet değerlendirme</li> <li>Sahtecilik tespit</li> <li>Fiyat öngörülleri</li> <li>Ekonomik öngörüller</li> </ul>	Tıp	<ul style="list-style-type: none"> <li>Tıbbi tanı</li> <li>Algılama ve tıbbi olayların değerlendirilmesi</li> <li>Hastanın hastanede kalma süresi öngörülleri</li> <li>Tedavi maliyet öngörüsü</li> </ul>
Satış ve Pazarlama	<ul style="list-style-type: none"> <li>Satış öngörüsü</li> <li>Hedefli pazarlama</li> <li>Hizmet kullanımı öngörüsü</li> <li>Perakende kar marjı öngörüsü</li> </ul>	Yöneylem Araştırması	<ul style="list-style-type: none"> <li>Perakende stok optimizasyonu</li> <li>Planlama optimizasyonu</li> <li>Yönetimsel karar verme</li> <li>Nakit akışı öngörüsü</li> </ul>
Endüstriyel	<ul style="list-style-type: none"> <li>Süreç kontrolü</li> <li>Kalite kontrol</li> <li>Sıcaklık ve basınç öngörülleri</li> </ul>	İnsan Kaynakları Yönetimi	<ul style="list-style-type: none"> <li>Çalışan seçme ve işe alım</li> <li>Çalışan tutma</li> <li>Personel planlama</li> <li>Personel profili çıkarma</li> </ul>
Bilim	<ul style="list-style-type: none"> <li>Örüntü tanıma</li> <li>Tarif ve kimyasal formülasyon optimizasyonu</li> <li>Kimyasal bileşik tanıma</li> <li>Fiziksel sistem modelleme</li> <li>Ekosistem değerlendirme</li> <li>Polimer kimlik tanıma</li> <li>Gen tanıma</li> <li>Botanik sınıflandırma</li> <li>Sinyal işleme: sinir filtreleme</li> <li>Biyolojik sistemler analizi</li> <li>Yer seviyesi ozon öngörülleri</li> <li>Koku analizi ve tanıma</li> </ul>	Enerji	<ul style="list-style-type: none"> <li>Elektrik yük öngörüsü</li> <li>Enerji talep öngörüsü</li> <li>Kısa ve uzun vadeli yük öngörüsü</li> <li>Yakıt endeks fiyatları öngörüsü</li> <li>Güç kontrol sistemleri</li> <li>Baraj su seviyesi izleme</li> </ul>
		Veri Madenciliği	<ul style="list-style-type: none"> <li>Tahmin</li> <li>Sınıflandırma</li> <li>Değişim ve sapma tespiti</li> <li>Bilgi keşfi</li> <li>Tepki modelleme</li> <li>Zaman serileri analizi</li> </ul>
Eğitim	<ul style="list-style-type: none"> <li>Sinir ağları öğretim</li> <li>Sinir ağı araştırma</li> <li>Koleji başvuru eleme</li> <li>Öğrenci performans tahmin</li> </ul>	Diğer	<ul style="list-style-type: none"> <li>Spor bahisleri</li> <li>At ve köpek yarışı seçimleri</li> <li>Kantitatif hava öngörüsü</li> <li>Bilgisayar oyunları geliştirme</li> <li>Optimizasyon problemleri</li> <li>Araç yönlendirme</li> <li>Tarımsal üretim tahminleri</li> </ul>

### 3.2.4.1.4 Kademeli bozulma

Kademeli bozulma (graceful degredation) hatalara ve eksik verilere karşı dayanıklı olma demektir. Kademeli bozulma özelliği ile sistemin bir kısmı çalışmasını tamamen durdurursa bile sistem çıktılarında çok büyük bozulmalar görülmemekte ya da sistem kendini kısa süre içinde daha iyi çıktılar çıkacak şekilde uyarlayabilmektedir. YSA'da nöronların bir ya da bir kaçının sistemden çıkarılması dahi sonuçları kötüleştirirse de tamamen anlamsız sonuçların elde edilmesine neden olmamaktadır.

### 3.2.4.1.5 YSA modelleri ile ilgili dikkat edilmesi gereken hususlar

YSA kullanımında dikkat edilecek unsurlardan biri verilerin kullanılmadan önce bir normalizasyon işleminden geçirilmesi gerekliliğidir. Bu işlem genellikle modelin daha hızlı çalışmasını ve anlamlı sonuçların elde edilmesini sağlayacaktır. YSA bir çok farklı problemin çözümünde kullanılıp anlamlı sonuçların elde edilmesi sağlanabilir, yine de eğer bir problem için geliştirilmiş bir çözüm yöntemi varsa, çalışmada bu yöntemin kullanılması YSA'nın ise çözümü olmayan sorunların çözümünde kullanılması daha iyi bir strateji olarak tercih edilebilir. Ayrıca makine öğrenmesi yöntemlerinin genel bir özelliği olmakla birlikte, YSA'nın da aşırı öğrenebileceği gözden kaçırılmamalıdır.

### 3.2.4.2 Dezavantajları

Bütün bu avantajların yanında YSA'nın bazı *dezavantajları* da bulunmaktadır. Bunlara bu başlık altında değinilmiştir.

#### 3.2.4.2.1 Ağ kurulmasında kuralların net olmaması

YSA'ların en büyük sorunlarından biri ağ yapısının belirlenmesi için belli bazı ilkeler dışında kesin kuralların bulunmamasıdır. Örneğin,

- Gizli katmanın kaç tane olması gerektiği ve tüm diğer katmanlardaki düğüm sayıları için genel kurallar bulunmamaktadır.
- Girdi ve çıktı katmanlarındaki düğümler problemin yapısına göre belli sayılar alabilmesine rağmen kesin bir kural yoktur.
- Gizli katman ve bu katmandaki nöron sayıları ise sadece deneme yanılma yöntemi ile belirlenebilmektedir. Bunlara rağmen düğüm sayılarının belirlenmesinde genetik algoritma gibi tekniklerden yararlanıldığı bilinmektedir.

YSA'da, ağın eğitiminin ne zaman biteceğine de kesin bir kuralla karar verilemez. Beklenen sonucun önemine ya da kritikliğine göre ağın öğrenirken belli bir hata düzeyine inmesi ya da belli bir iterasyon sayısı eğitimi sonlandırabilir.

### 3.2.4.2.2 Kara kutu özelliği

Ağın en büyük dezavantajlarından biri ağın parametrelerinin yorumlanamamasıdır. Ağ bütünsel olarak çalışmakta ve sonuç üretmektedir. Tek bir nöron göz önüne alındığında bu nöronla ilgili parametrelerin hiç biri çözüm için bir şey ifade etmeyecektir. Bir bakıma problem çözümünde paralel çalışmanın verdiği avantaj modeli yorumlamada dezavantaja dönüşmektedir.

### 3.2.4.2.3 Ezberleme

Veri güdümlü öğrenmelerde genel bir sorun olarak modelin ezberleyebildiği bilinmektedir. Ezberlemek öğrenme verileri ile çok iyi performans gösteren bir ağın test verilerine geldiğinde performansının çok fazla düşmesinden anlaşılabilir. Çünkü öğrenme verileri modeli fazla eğitmiş ve sadece öğrenme verilerine uygun bir model ortaya çıkmıştır. Oysa aynı kümenin diğer bir kısmı test verileri için model çok da anlamlı olmayabilmektedir. Ezberleme olmaması için verinin sahip olduğu karaktere göre tecrübeli bir gözle modelin gizli katman sayısının uygun şekilde ayarlanması gerekmektedir.

### 3.2.4.2.4 Çözüme geç yakınsama

Yapay sinir ağları temel öğrenme algoritmaları çözüme geç yakınsayabilmektedir. Bu durumda çok sayıda iterasyon için hem zaman hem kaynak harcanması gerekmektedir. Ne var ki, günümüzde kullanılan yeni algoritmalar az sayıda iterasyon sayısı ve kısa zaman içinde çözüm üretebilmektedir.

### 3.2.4.2.5 Global değere değil yerel değere yakınsayabilme

Yapay sinir ağlarının diğer bir dezavantajı hatanın global minimuma erişememe ihtimalidir. YSA'lar bazı çalışmalarda yerel minimuma takılıp sonucu bu yerel minimum değeri olarak verirler. Global çözüme ulaşma şansı önsel bilgilerin ve çeşitli ek bazı parametrelerin kullanılması ile artırılabilir.

### 3.2.5 YSA ile zaman serileri öngörüsü

ZSA için en fazla kullanılan YSA model türü çok katmanlı algılayıcılardır. ÇKA, ZSA için kullanıldığında doğrusal olmayan ve  $\mathbb{R}^n - \mathbb{R}^m$  şeklinde tanımlanan bir fonksiyonu yaklaştırır.

Zaman serileri analizi YSA ile farklı şekillerde ele alınabilir (Öztemel, 2012). Örneğin, zaman serisinin günlük olduğunda ve aylık ortalama değerleri tahmin etmeye çalışılacağına, bunun için 30 günlük verileri sisteme girilir. Böyle bir sistemde çıktı ise aylık veriler olur. Fakat bu sistemde günlük tahminleri elde etmek mümkün olmaz. ZSA'nın YSA ile diğer bir şekilde ele alınışı şöyle olabilir: Otoregresif modellerde olduğu gibi bugünkü verinin daha önceki günlerdeki verilere göre belirleneceği varsayımı altında YSA'nın girdileri geçmiş değerler çıktısı ise şu anki değer olabilir. Böyle bir durumda günlük bir ya da birkaç gün sonrası için veri öngörüsü yapılabilir.

## 4 MODELLERLE ÖNGÖRÜ

### 4.1 Finans ve Oynak Bir Finansal Değişken: Hisse Senedi

Finans, makroekonomi bilim dalının önemli bir alt dalıdır. Finans en genel anlamda elinde varlık bulduran kişi ya da kurumların, bu varlıkları belirsizlikleri de göz önünde bulundurarak nasıl değerlendireceği ile ilgilendir. Bu çok geniş bir kavram olduğundan ve piyasa aktörüne göre bu strateji farklılık göstereceğinden, finans 3 ana başlık altında incelenir: Kişisel finans (kişisel yatırım), şirket finansı (finansal yönetim), halk finansı (maliye-para politikası). Özellikle hisse senetleri değerlerinin kestirimleri finansal çalışmaların oldukça aktif bir alanını oluşturmaktadır.

Hisse senetlerinin ikincil satışları borsalarda olur. Türkiye’de bugünkü anlamda borsa İMKB adı ile 1986 yılında kurulmuştur. Kriz dönemlerinin olumsuzluklarından diğer sektörlere göre daha çok etkilenmiştir. Bu nedenle dünya ekonomilerindeki örneklerine göre işlemlerin hacminin ekonomiye olan oranı daha küçük kalmıştır. Son dönemde adı değişen İMKB, yeni adı ile Borsa İstanbul (BIST), çeşitli tip araçların mübadele işlemlerinin tek çatı altında toplanmasını sağlayacaktır. Bu da ileride hedeflenen daha kurumsal bir pazarın ilk adımı olarak görülmektedir.

Finans sektöründe kullanılan birçok finansal araç bulunmaktadır. Bir ülkede finans sektörü ne kadar gelişmişse kullanılan finansal araç sayısı da o kadar çeşitlenmektedir. Finansal araçlardan hisse getiri oranları, faiz oranları, bonolar, opsiyonlar, forward sözleşme ve future sözleşmelerin hem kanuni imtiyaz ve yükümlülükleri, hem de değerlendirme yöntemleri birbirinden farklıdır. Bunlar işletme mühendisliği ve finans mühendisliği bilim dalları altında incelenen bilgilerdir.

Tezde yapılması amaçlanan, finansal değişkenlerin ele alındığı modellerin parametrelerini tahmin etmek ve bu modeller ile geleceğe ilişkin kestirimler yapmaktır. Çalışmadaki modellerin tamamında verilerin tam rassal olmadığı ve cari verinin daha önceki verilerden etkilendiği varsayımı bulunmaktadır.



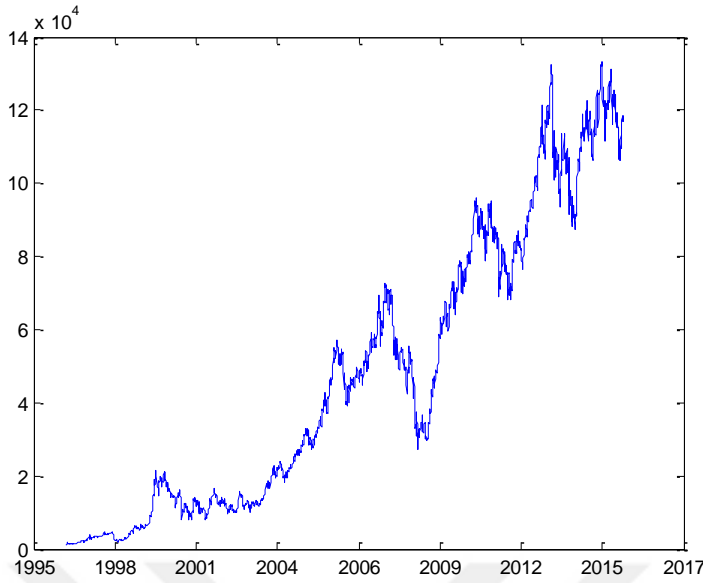
Finans sektörü ile ilgili en fazla çalışma yapılan değişkenlerden biri hisse senetlerinin ikincil değerleridir. Bu çalışmada hisse senetleri hakkında toplu bilgi veren hisse senedi endeksi verileri kullanılmıştır.

Uygulamada 02.01.1997-16.10.2015 tarihleri arası günlük BIST 100 getiri endeksi kapanış verileri kullanılmıştır. Çalışmada bu veri seti B100gek olarak adlandırılmış. Veri seti 4688 adet gözlemden oluşmaktadır. Veriler TCMB sayfasından temin edilmiştir.

Uygulama çalışmasında makine öğrenmesi modellemelerinde sıkça kullanılan performans ölçütlerinden biri olan ortalama mutlak hata (MAE) model öngörülerini karşılaştırmak amacı ile kullanılmıştır. Makine öğrenmesi çalışmalarında da çoğu zaman görüldüğü gibi, veri seti ikiye ayrılmış, modelleme ilk veri seti ile gerçekleştirilirken diğer veri seti öngörü için modellemeye katılmadan bekletilmiştir. Daha sonra ilk veri seti ile belirlenen modelleri kullanarak ikinci veri seti ile öngörüler yapılmıştır. Uygulama R2012a versiyonlu MATLAB programı ile gerçekleştirilmiştir. Uygulamanın bazı kısımları hazır kütüphaneler ile bazı kısımları ise MATLAB dili ile kodlayarak yapılmıştır.

## 4.2 Veri Özellikleri

B100gek veri seti tatil günleri hariç olarak oluşturulmuştur. Böylelikle 02.01.1997-16.10.2015 tarihlerini kapsayan aralıkta üretilmiş 4688 gözlem ile çalışma yapılmıştır. Verilerin grafiği Şekil 4.1.'de verilerin log işlemcisinden geçirildikten sonraki grafiği Şekil 4.2.'de görülmektedir. 10 tabanına göre logaritma işlemcisinden geçirilmiş veri LB100gek olarak adlandırılmıştır.



Şekil 4.1. B100ge

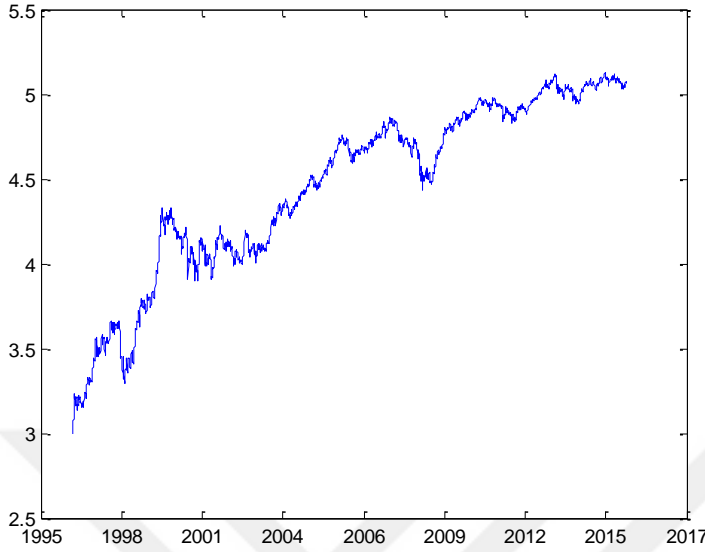
Şekil 4.1. ve Şekil 4.2.'deki veri grafiklerinden endeks değerinin 1998'ten 2000 yılına kadar ve 2008 yılı sonundan 2010 yılı başına kadar çok hızlı arttığı görülmektedir. Büyük düşüşler ise 1998'in hemen başında ve 2007 yıllarında olmuştur.

Şekillerde, özellikle ilk yıllarda oynaklığın yüksek olduğu göze çarpmaktadır. Grafikten BIST'te, o zamanki adı ile İMKB'de bilgisayarla işlem yapılmaya başladığı bu yıllarda farklı dinamiklerin etkili olduğu sonucu çıkarılabilir. Bu oynaklık bilgisi o yıllarda daha küçük ve mübadele şekli farklılaşan bir pazarın satıcı ve alıcılarının tecrübe edininceye ve pazardaki ürünlerin fiyatları dengeye ulaşmaya kadar geçen bir dönemin etkisi olabilir. Şekil 4.1. ve Şekil 4.2.'de seri boyunca volatilité kümelenmeleri de dikkat çekmektedir. Sezgisel olarak serideki yükselmelerin düşüşler kadar etkili olduğu görülmektedir, nitekim Çizelge 4.1.'de görüldüğü üzere dağılımda simetri göze çarpmaktadır.

Çizelge 4.1. Artış ve düşüşlere göre dağılımlar

	Artışlar	Sert artma	Yüksek artma	Artma	İniş	Yüksek iniş	Sert iniş
Sert deęişim aralığı							
% 2		746	682	1058	943	614	644
% 3		407	1021	1058	943	924	334

Buna rağmen çok güçlü olmamakla birlikte kaldıraç etkilerinin varlığı göze çarpmaktadır.



Şekil 4.2. LB100gek

Ayrıca grafiklerden, orijinal serinin durağan olmadığı ile ilgili önsel bir bilgi elde edilir fakat bunu yine de istatistiksel olarak göstermek gerekir. Durağanlık doğrusal zaman serileri modellemelerinde öngörü için oldukça önemli bir kriterdir ve durağan serilerle çalışılmak istenir.

Durağan olmama çeşitli nedenlerden kaynaklanmış olabilir. Borsa günlük endeks verileri için dikkat edilmesi gereken serinin otokorelasyon yapısıdır. Bu nedenle durağanlık Augmented Dickey Fuller testi (ADF testi) ile birim kök araştırarak gerçekleştirilmiştir. ADF testi, Dickey Fuller testinin (DF testi) geliştirilmiş bir halidir. Dickey ve Fuller'in (1979) önerdiği DF testi EKK t.e.'sinin dağılımına dayalı kullanımı standartlaşmış bir birim kök testidir (Akdi, 2012).

DF testi 3 farklı model tipi için gerçekleştirilebilir:

- Model I:  $Y_t = \rho Y_{t-1} + e_t$
- Model II:  $Y_t = a_0 + \rho Y_{t-1} + e_t$
- Model III:  $Y_t = a_0 + a_1 t + \rho Y_{t-1} + e_t$

Bu modeller için sıfır hipotezi  $\rho = 1$ , diğer bir deyişle birim kök vardır şeklindedir. Regresyondakine benzer bir hipotez, eşitliklerin iki tarafından  $Y_{t-1}$  ifadesinin çıkarılması ile elde edilir. Bu durumda sıfır hipotezi  $\delta = 0$  olmak üzere yine 3 tip model elde edilir:

- Model I:  $\nabla Y_t = \delta Y_{t-1} + e_t$
- Model II:  $\nabla Y_t = a_0 + \delta Y_{t-1} + e_t$
- Model III:  $\nabla Y_t = a_0 + a_1 t + \delta Y_{t-1} + e_t$

ADF testinin genel halinin kurulumu ise

$$\nabla Y_t = a_0 + a_1 t + \gamma Y_{t-1} + \delta_1 \nabla Y_{t-1} + \delta_2 \nabla Y_{t-2} + \dots + \delta_{p-1} \nabla Y_{t-p+1} + e_t \quad (4.1)$$

şeklindedir. Burada sıfır hipotezi  $\gamma = 0$  ve alternatif hipotez  $\gamma < 0$  olmaktadır. Yukarıda gösterilen tüm modellerde birim kök olma durumunda t.e.'ler asimptotik olarak normal dağılmayacağından t istatistiklerinin kritik değerleri standart t dağılımının kritik değerlerinden farklı olacaktır. Bu nedenle testler Dickey-Fuller dağılımı kritik değerleri kullanılarak gerçekleştirilir.

B100gek verisine ADF testi uygulandığında seride birim kök olduğu dolayısıyla serinin durağan olmadığı gösterilmiş olur. B100gek serinin ADF testi bilgileri Çizelge 4.2.'de görülebilir.

Çizelge 4.2. ADF testi sonuçları

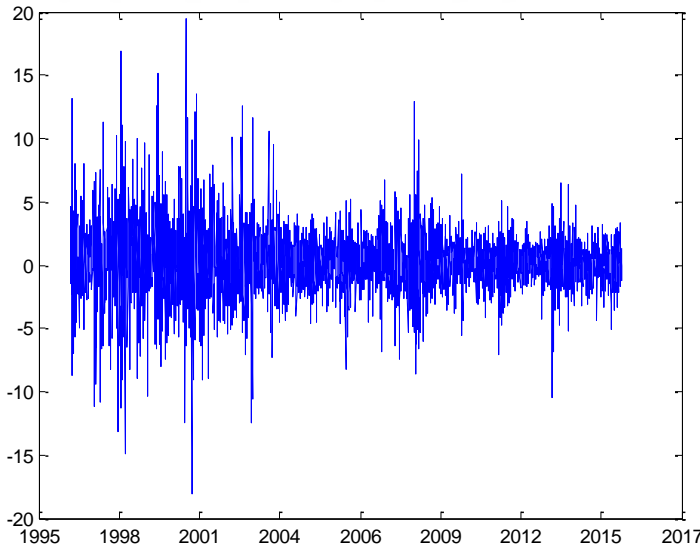
Veri Seti		Sabitli ve Trendsiz	Sabitli ve Trendsiz	Sabitli ve Trendli
B100gek	t-değeri	1,0274	-0,4177	-3,1444
	kritik değer	-1,9416	-2,8632	-3,4136
	p değeri	0,9205	0,9035	0,0964
GB100gek	t-değeri	-29,1224	-29,3836	-29,4990
	kritik değer	-1,9416	-2,8632	-3,4136
	p değeri	0,0010	0,0010	0,0010

Veriye istatistiksel açıdan istenen bir kaç özellik kazandıran bir dönüşüm getiri dönüşümüdür. Bu nedenle bu çalışmada

$$R_t = \frac{P_t - P_{t-1}}{P_{t-1}} \quad (4.2)$$

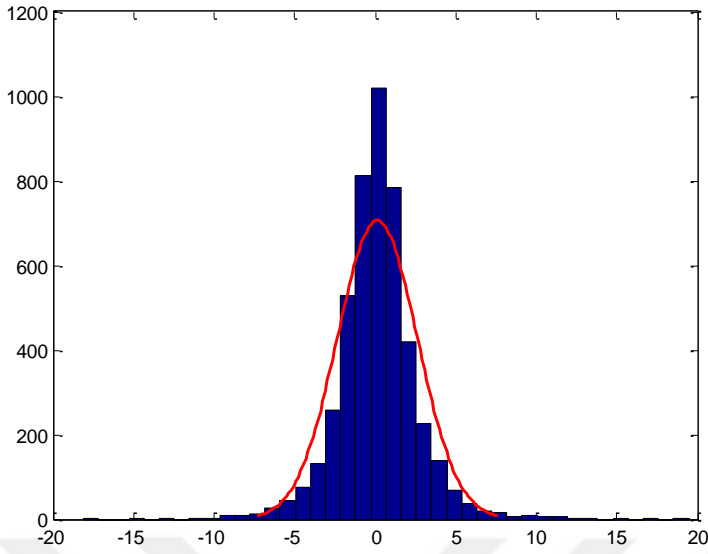
şekilde tanımlanan, tek periyod basit yüzde getirinin istenilen özelliklere sahip olacağı beklenmektedir. Tek periyod basit yüzde getiri verisi kısaca getiri verisi, ölçekten bağımsızdır ve istatistiksel özellikleri çalışmalara daha uygundur. Getiri verisinin ADF testi ile durağanlığı araştırıldığında serinin net bir şekilde durağan olduğu görülür, bu çalışmanın bilgileri Çizelge 4.2.'de görülebilir.

Getiri verisinin çalışmaya uygun özellikleri göz önünde bulundurularak çalışmalar getiri verisi ile yapılmıştır. Getiri verisinin zaman içindeki hareketi Şekil 4.3.'te verilmiştir. Getiri verilerinde seri boyunca oluşan volatilité kümeleri Şekil 4.1. ve Şekil 4.2.'deki bilgileri destekler nitelikte ve daha net bir biçimde görülebilmektedir.



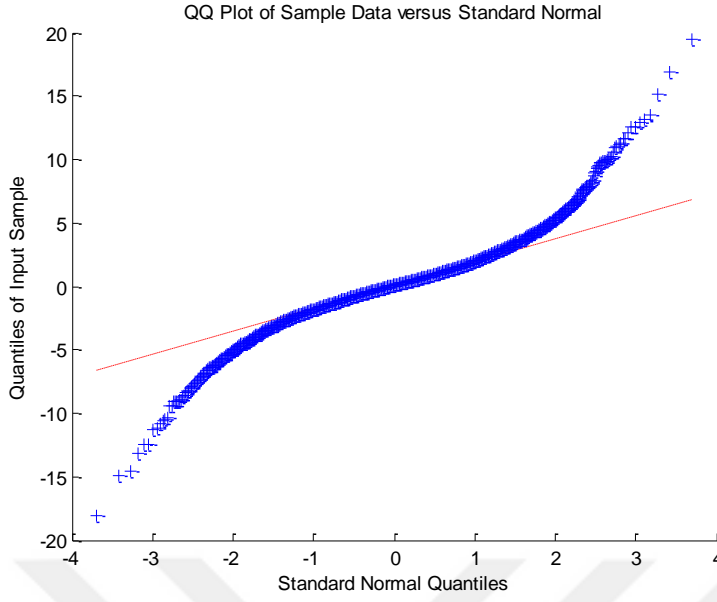
Şekil 4.3. GB100gek verisi

Ortalaması sıfıra yakın olan getiri verisinin dağılımı hakkındaki bulgular Şekil 4.4. ve Şekil 4.5.'te görülebilir. Çizelge 4.3.'den serinin normal dağılmadığı bilgisi istatistiksel olarak elde edilmektedir.



Şekil 4.4. GB100gek histogram ve normal dağılım

Dağılımdaki simetriklik dikkat çekmektedir. Diğer yüksek frekanslı çoğu finansal veride olduğu gibi kalın kuyruk özelliği de görülmektedir. QQ çizgesinden de her iki kuyrukta da dışa düşenler olduğu göze çarpar. Veri ile ilgili oluşturulan betimleyici istatistik ve serisel ilişki bilgileri ise Çizelge 4.3.'te verilmiştir.



Şekil 4.5. GB100gek QQ-Çizgesi

Çizelge 4.3. B100gek ile ilgili genel bilgiler

B100gek 02.01.1997-16.10.2015	
Veri Adedi:	4688
Min.	-18,10935
Maks.	19,45100
Medyan	0,12289
Ortalama	0,13263
Var	6,18163
St. Dev	2,48629
Çarpıklık	0,19287
Basıklık	8,77823
GB100gek Kolmogorov-Smirnov Test	
İstatistik değeri	0,15813
Kritik değer	0,01980
p – değeri	0,0000
GB100gek Ljung-Box Q Testi	
İstatistik değeri	15,6496
Kritik değer	11,0705
p – değeri	0,0079

Daha önce belirtildiği gibi modeller ortalama mutlak hata (MAE) ölçütü ile karşılaştırılmıştır. Bu ölçütün matematiksel ifadesi şu şekildedir:

$$MAE = \left( \frac{\sum_{t=1}^n |y_t - \hat{y}_{t|t-1}|}{n} \right) \quad (4.3)$$

### 4.3 ARMA modelleri

ARMA ve ARCH/GARCH modelleri doğrusal zaman serilerini çözümlmek amacıyla kullanılırlar. Her iki model ile de günümüze kadar sayısız çalışma yapılmıştır ve hala yapılmaktadır. Yoğun bir matematiksel alt yapıya sahip ARMA modelleri kullanım kolaylığı ve zaman serilerini modelleme gücü nedeniyle çalışmalarda sıkça kullanılmaktadır. ARCH/GARCH modelleri ise birlikte kullanıldıklarında ARMA modellerine ek bazı yetenekler kazandırmakta ve modellerin etkinliğini artırmaktadır.

Doğrusal zaman serileri modelleri ekonometrik yöntemlerin önde gelen araçlarından biridir fakat bu modellerin sadece ekonometri alanında kullanıldığını düşünmemek gerekir. Bu modeller mühendislik dallarında yaygın bir biçimde kullanılmaktadırlar. Bir teknoloji şirketi, sahibi olduğu sinyal işleme yazılım aracının internet sayfasında şöyle demektedir:

Model bazlı yöntemler zaman serilerinin beyaz gürültü tarafından tetiklenen doğrusal modellerin tepkileri olduğunu varsayarlar. .... Doğrusal sistem  $H(z)$ , genellikle AR, MA ya da ARMA modelleri ile betimlenebilir. Zaman serileri için bu tür bir model üretebilmek amacı ile öncelikle model parametreleri tahmin edilir sonrasında zaman serisinin güç tayf yoğunluğu (PSD) hesaplanır ([http://zone.ni.com/reference/en-XX/help/372656C-1/lvasptconcepts/tsa\\_model-based/](http://zone.ni.com/reference/en-XX/help/372656C-1/lvasptconcepts/tsa_model-based/), 10.12.2015).

Literatürde ARMA ve ARCH/GARCH modelleri kullanılarak yapılan bir çok Türkçe çalışma bulunabilir. Son zamanlarda ARMA ve ARCH/GARCH modelleri kullanılarak gerçekleştirilen çalışmalardan biri işletmeleri etkileyen iç ve dış faktörleri modelleyen bir çalışmadır (Can, 2009). Çalışmada şirketin net satışları ve ham petrol fiyatları modellenmiştir. Ekonometri alanında Türkiye’de ARCH modeli kullanılarak yapılan ilk çalışmalardan biri Gökçe (1998)’in çalışmasıdır. Gökçe, GARCH modelleri ile döviz piyasası ve menkul kıymetler piyasası verilerini ele almış, bu yolla volatilité ile getiriler arasındaki pozitif bir ilişki olduğunu belirtmiştir.

Berument, Köse ve Şahin (2010) ise GARCH modellerinin bir türevi olan E-GARCH modelini kullanarak kısa ve uzun dönem oynaklık öngörülerini kaldıraç etkilerini incelemişlerdir.

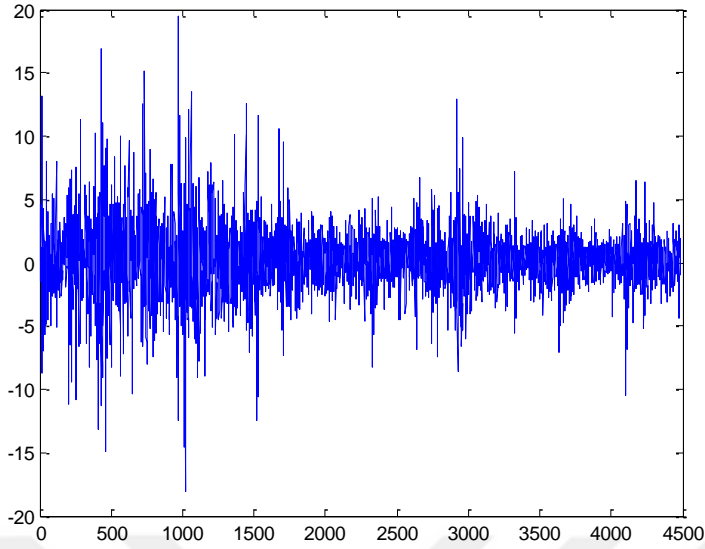


ARMA modellerinin temelleri 1926 yılında atılmaya başlamıştır. Günümüze kadar geçerliliğini sürdüren ve yeni tahmin yöntemleri ile desteklenen bu modeller çeşitli varsayımlara dayanır ve bu varsayımlar sağlandığı ölçüde istatistiksel açıdan istenilen sonuçları verirler. Varsayımlar sağlanmazsa da çeşitli düzeltme yöntemleri ya da ek bazı kuralları araştırıp doğrulamak suretiyle yine bu yöntemler kullanılabilir. Fakat her durumda belli bir prosedür izlenmeli ve varsayımların sağlanıp sağlanmadığı araştırılmalıdır.

ARMA modelleri durağanlık varsayımı altında çalışır. 1. fark verileri ya da benzer özellikler gösteren getiri verileri ARMA modelleri için sıkça kullanılan veri türleridir. Bu tezde de çalışmalar bir önceki alt bölümde durağan olduğu gösterilmiş olan getiri verileri ile yapılmıştır. Box-Jenkins yönteminin ilk adımı model belirlemedir.

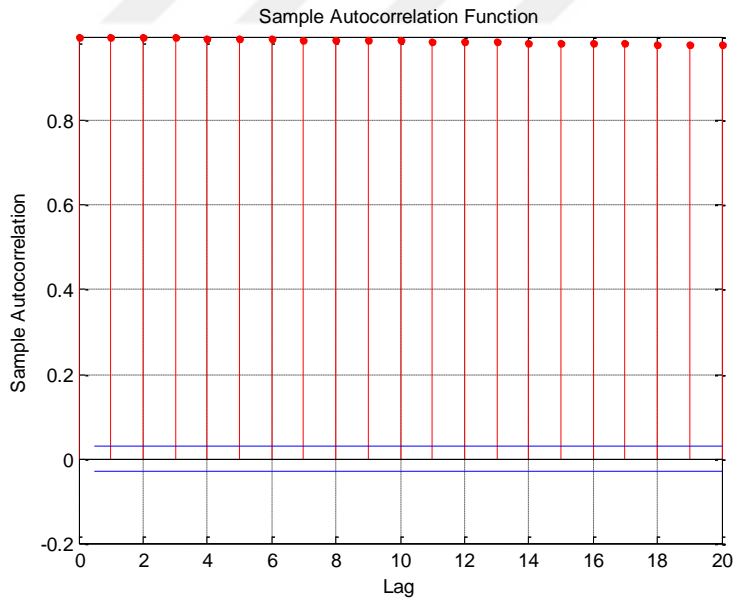
#### Model belirleme:

Model belirlemek için çeşitli araçlardan yararlanır. Bu çalışmada serinin korelasyon yapısına bakıldıktan sonra regresyon analizi bilgileri ve bilgi kriterlerinden yararlanılmıştır. Modelleme için modellemeye ilişkin orijinal veriden elde edilmiş getiri verisi MGB100gek kullanılmıştır (Şekil 4.6.).



Şekil 4.6. MGB100gek

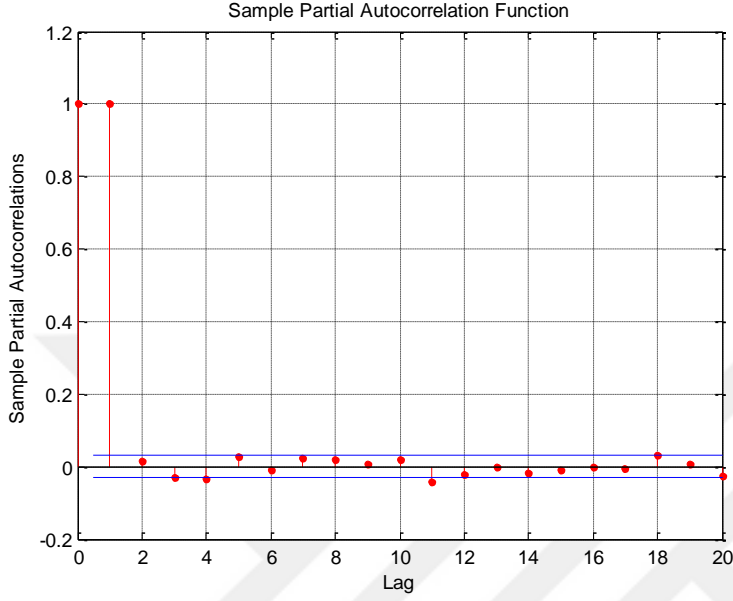
B100gek verisi için korelograma bakıldığında örneklem otokorelasyon değerlerinin çok yavaş bir şekilde azaldığı görülür (Şekil 4.7.).



Şekil 4.7. MB100gek korelogram

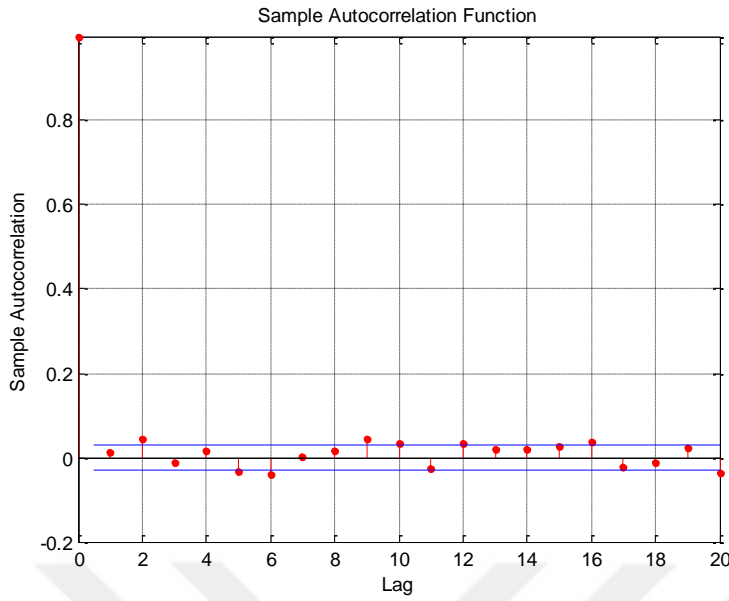
Bu beklenen bir durumdur çünkü veri özelliklerinde serinin zaten durağan olmadığı görülmüştür.

Kısmi korelograma bakıldığında ise 1. gecikme için sıfırdan farklı bir değer elde edilmekte diğer gecikmeler için ise sıfır ya da sıfıra çok yakın değerler göze çarpmaktadır, kısmi korelogramdan AR(1) modeline işaret edildiği sezgisel olarak söylenebilir (Şekil 4.8.).



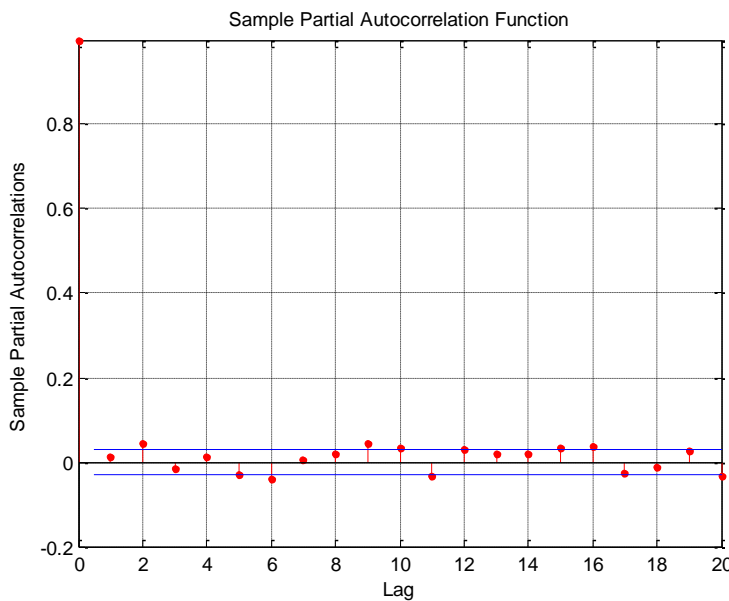
Şekil 4.8. MB100gek kısmi korelogram

Getiri verilerinin korelogram incelendiğinde 6. gecikmeye kadar sıfırdan farklı değerler göze çarpmaktadır (Şekil 4.9.).



Şekil 4.9. MGB100gek korelogramı

Getiri verilerinin kısmi korelogramında da 6. gecikmeye kadar sıfırdan farklı değerler bulunmaktadır (Şekil 4.10.). Getiri verisinin okorf ve kokorf'u çeşitli gecikmeler için sıfırdan farklı değerlere sahip olsa da bunların çok kuvvetli olmadığı söylenebilir. Bu aşamada çeşitli aday modeller elde edilmiş sonrasında bu modellerin parametreleri tahmin edilmiştir. Daha sonra her bir aday modelin AIC ve BIC değerleri elde edilmiştir, ilgili bilgiler Çizelge 4.4.'te görülmektedir.



Şekil 4.10. MGB100gek kısmi korelogramı

Akaike ve Schwarz bilgi kriterinin her ikisi de ARMA(1,2,5;1,2) modeli için en düşük değerleri almışlardır fakat kriterlere ilişkin olarak modeller arasında belirgin derecede bir farka rastlanmamıştır. Daha önceki çalışmalar dikkate alındığında ve bu çalışma sırasında verilerden elde edilen bilgiler doğrultusunda uygun model AR(1) olarak seçilmiştir. Model parametreleri tahminine ilişkin bilgiler Çizelge 4.5.'te verilmiştir.

Çizelge 4.4. Getiri verisi için karşılaştırılan modellerin AIC ve BIC değerleri

Model	AIC	BIC
AR(1)	21040	21047
ARMA(1,2,5;1)	21012	21038
ARMA(1,5;1,2)	21012	21038
ARMA(1,2,5;1,2)	21004	21036
ARMA(1,2,5,6;1,2)	21005	21043

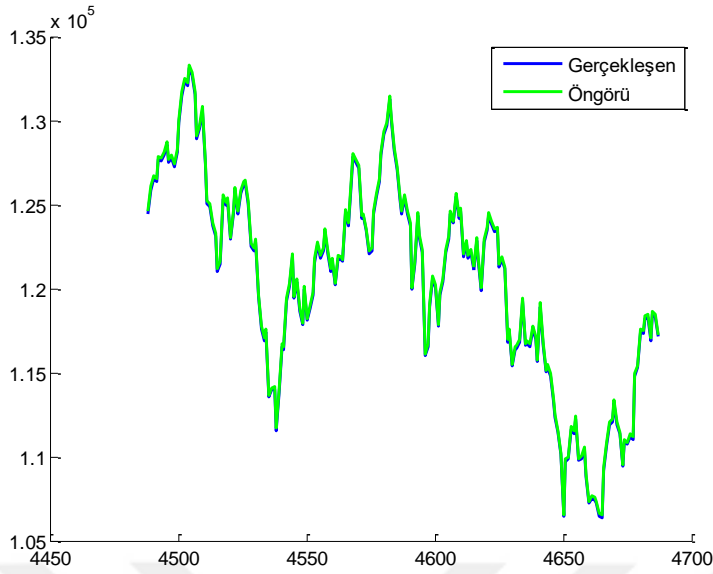
En uygun seçilen model parametreleri Çizelge 4.5'te verilmiştir.

Çizelge 4.5. Model parametre tahmin bilgileri

Parametreler	Değer	t - istatistiği	Standart hata
AR(1)	0,011414	1,31122	0,00870548
Sabit	0,137970	3,65093	0,03779040

Model kalıntıları ile ilgili bilgiler çalışmanın ekler bölümünde görülebilir.

En uygun modelle test verileri ile gerçekleştirilen öngörü için grafiği ise Şekil 4.11.'te verilmiştir.

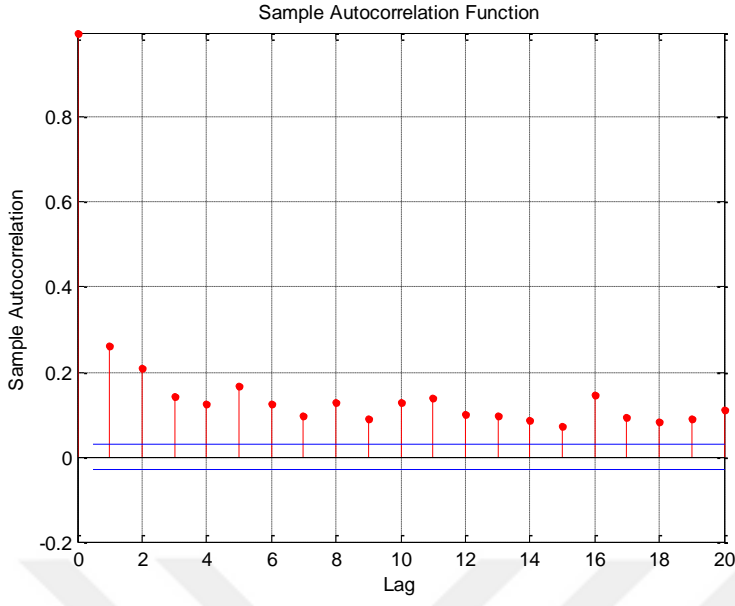


Şekil 4.11. ARMA öngörüsü

#### 4.4 Oynaklık modelleri

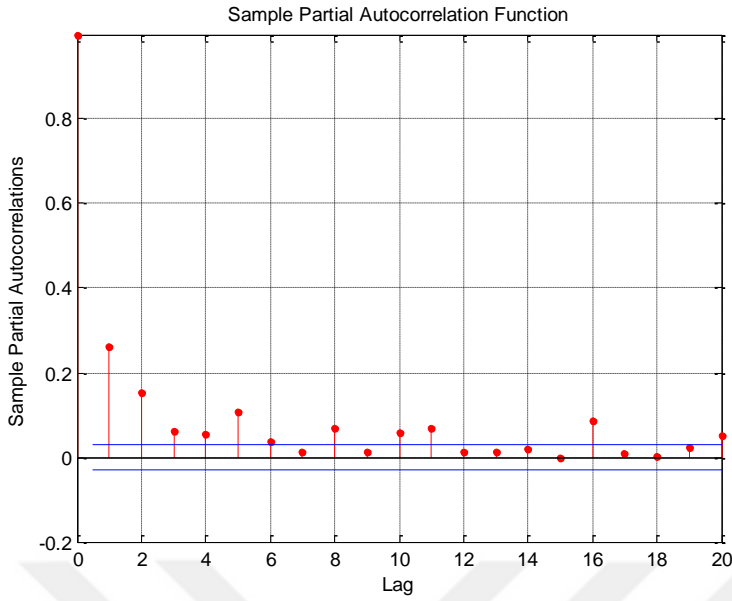
ARMA modellerinin varsayımlarından bir tanesi hata terimlerinin IID özellik göstermesidir. Oysa hata karelerinin serisel ilişkisi çeşitli veri setlerinde zaman zaman bu özelliğin sağlanmadığına işaret eder. Bu durumda veri setinin ek bir çalışma ile ikinci dereceden moment ilişkilerinin ele alınması doğru bir yaklaşım gibi düşünülebilir. Üstelik böyle bir çalışma veri setinden kaynaklı doğabilecek risklerden etkileneceklerin de ilgi alanına girmektedir. Bu durumda zaman ile değişebilen bu riski takip etmek ve hatta öngörmek önemli bir avantaj sağlar.

Uygun model için MGB100gekhk verisinin korelogramına (Şekil 4.12.) ve kısmi korelogramına (Şekil 4.13.) bakıldığında yukarıda bahsedilen ilişki göze çarpar. Bu ilişki verinin biraz daha incelenmesi için gerekli motivasyonu oluşturmaktadır.



Şekil 4.12. MGB100gekhk korelogramı

Uygun modelin kalıntı karelerinde serisel bir ilişkinin istatistiksel olarak gösterilmesi gerekir. ARCH etkisinin varlığını gösterebilmek için önde gelen testlerden biri ARCH-LM testi ve bir diğeri Ljung-Box testidir.



Şekil 4.13. MGB100gekhk kısmi korelogramı

Bu iki test bilgileri uygun model kalıntıları üzerinden uygulandığında iki testin de boş hipotezi reddettiği görülmektedir (Çizelge 4.6.). Buna göre seride ARCH etkisi bulunmaktadır.

Çizelge 4.6. ARCH etkisi ile ilgili bilgiler

MGB100gekhk Ljung-Box Q Testi bilgileri (5 gecikme için)	
İstatistik değeri	784
Kritik değer	11,0705
p – değeri	0
MGB100gekh ARCH-LM testi	
İstatistik değeri	674,3230
Kritik değer	96,2167
p – değeri	0

Bu bilgiler ARCH/GARCH değişkenlerini modele katabileceğimize işaret eder. Modele ARCH/GARCH terimlerini birlikte kattığımız için regresyon bilgileri kullanılarak deneme yanılma prosedürü ile çalışmaya devam edilmiştir. Uygun modelin elde edilmesi için bilgi kriterlerinden de yararlanılmıştır. Buna göre en uygun model seçim kararı GARCH(1,1) yönünde olmuştur. Bilgiler Çizelge 4.7’de görülmektedir.



Çizelge 4.7. Getiri verisi için karşılaştırılan modellerin AIC ve BIC değerleri

Model	AIC	BIC
ARCH(1)	20556	20569
GARCH(1;1)	19519	19538
GARCH(1;2)	19640	19660
GARCH(2;1)	19706	19725
GARCH(2;2)	19737	19756

Buna göre ortalama denklemi parametre tahmin bilgileri Çizelge 4.8. ve oynaklık denklemi parametre tahmin bilgileri Çizelge 4.9.'da verilmiştir.

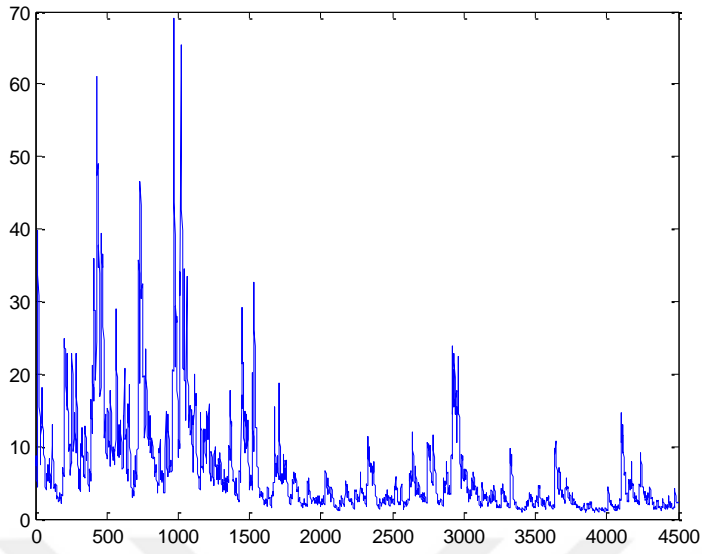
Çizelge 4.8. Ortalama denklemi parametre tahmin bilgileri

Parametreler	Tahmin değeri	t - istatistiği	Standart hata
AR(1)	0,0272857	1,73772	0,015702
Sabit	0,1518490	5,68038	0,026732

Çizelge 4.9. Oynaklık denklemi parametre tahmin bilgileri

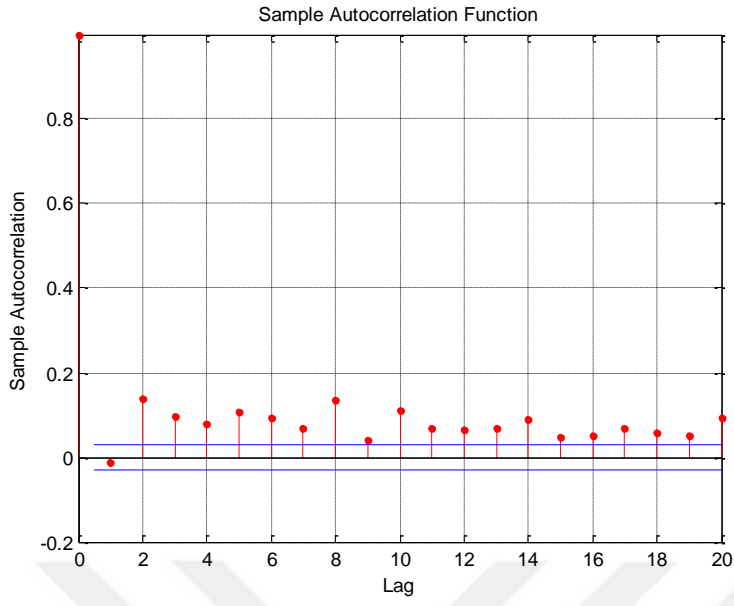
Parametreler	Tahmin değeri	t - istatistiği	Standart hata
ARCH(1)	0,1001780	17,7406	0,00564681
GARCH(1)	0,8957000	178,288	0,00502389
Sabit	0,0550021	6,79887	0,00808989

Oynaklık modeli tanımlandıktan sonra koşullu varyansın zaman içindeki değişimi elde edilmiştir (Şekil 4.14.).

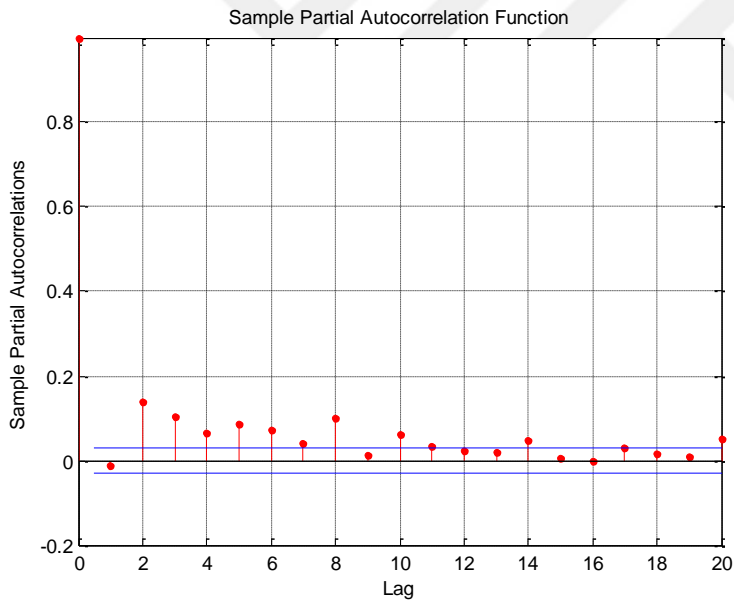


Şekil 4.14. Koşullu varyans

Modelden elde edilen standartlaştırılmış kalıntılar ile model doğrulama prosedürü gerçekleştirilmiştir. Standartlaştırılmış kalıntıların karelerinin korelogramlarına dayanarak daha rassal kalıntı terimleri elde edildiği söylenebilir (Şekil 4.15. ve Şekil 4.16.).

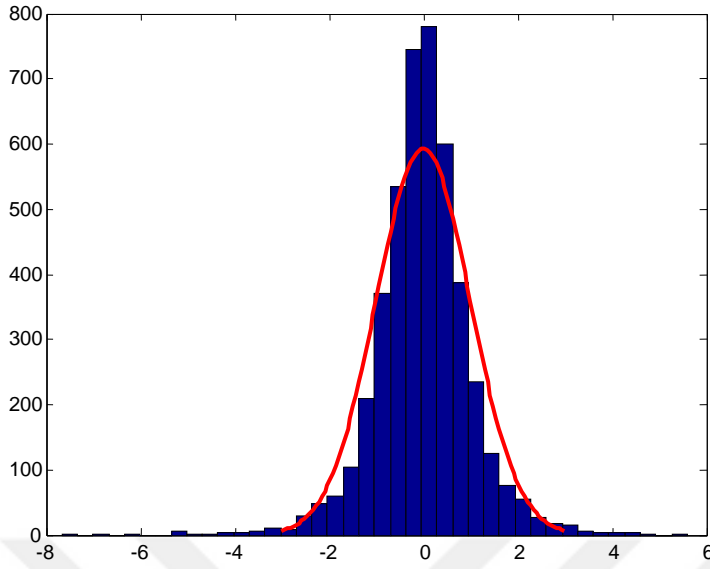


Şekil 4.15. GARCH standartlaştırılmış kalıntı kareler korelogram

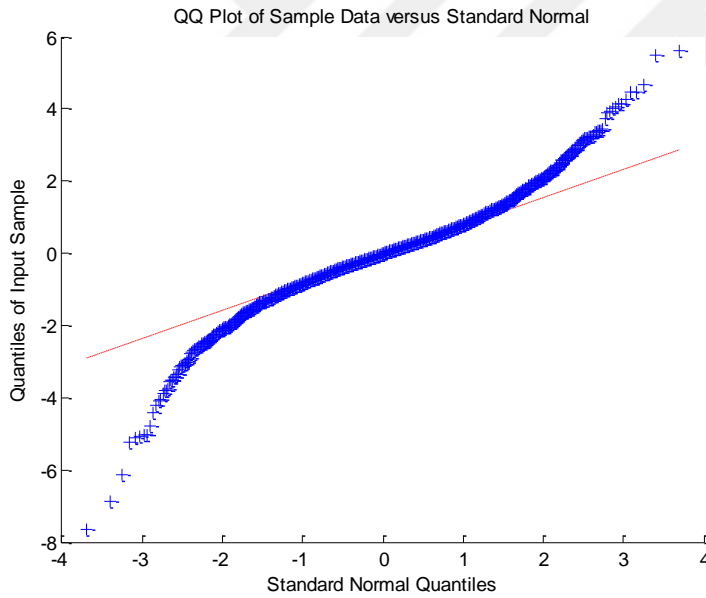


Şekil 4.16. GARCH standartlaştırılmış kalıntı kareler kısmi korelogram

Yine standartlaştırılmış kalıntılarının normal dağılıma yakınlaştığı da Şekil 4.17. ve Şekil 4.18.'den görülebilmektedir.



Şekil 4.17. GARCH standartlaştırılmış model kalıntıları histogram



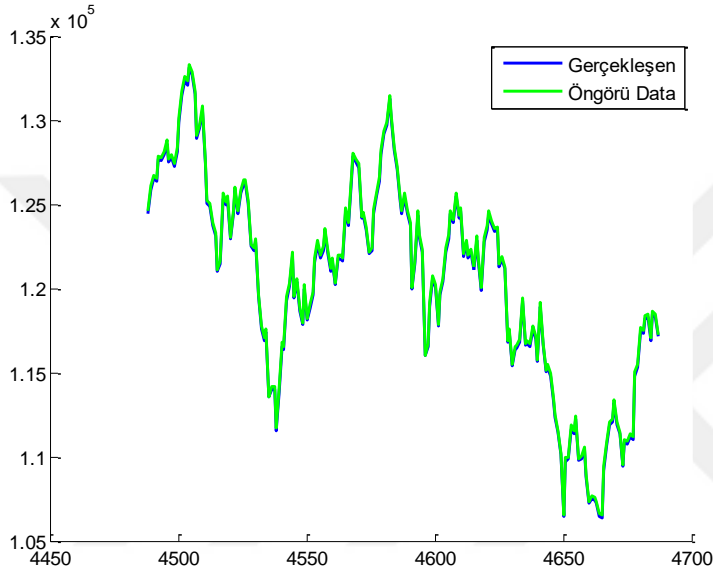
Şekil 4.18. GARCH model kalıntıları QQ çizgesi

Ljung-Box (LB) testlerine göre yüzde 1'lik anlamlılık seviyesinde hata terimlerinin ilişkisiz olduğu boş hipotezi reddetmek için yeterli bir kanıt görülememiştir (Çizelge 4.10.).

Çizelge 4.10. Standartlaştırılmış kalıntılar için LB testi bilgileri

Gecikmeler	Test istatistiği	Kritik değer	p-değeri
LB(5)	11,4723	11,0705	0,0428
LB(10)	19,5210	18,3070	0,0341

Son olarak model ile ilgili öngörü grafiği Şekil 4.19.'de verilmiştir.



Şekil 4.19. ARCH/GARCH öngörüsü

#### 4.5 SMM Yöntemi

Makine öğrenmesi tekniklerinin genel bir özelliği veri güdümlü olmalarıdır. Veri güdümlü yöntemler ise genellikle daha az varsayım altında uygun sonuçlar bulunmasını sağlarlar. SMM'de bu tekniklerden biridir.

SMM 1960'lerden bu yana konuşma tanıma problemi için en çok kullanılan yöntemdir (Bilmes, 2006). Zaman serilerin doğrudan SMM ile modellenmediği fakat gözlemlenmeyen bir süreç varsayımının olduğu diğer bir modelleme Hamilton tarafından 1989 yılında kullanılmıştır. Hamilton'un kullandığı yöntemde, altta rejimin rassal olarak farklılaştığı varsayımı vardır.

SMM son yıllarda çok çeşitli alanlarda ve farklı tür problemin çözümü için kullanılmaya başlamıştır. Özellikle finansal zaman serisinin modellenmesi ve öngörü amaçlı çalışmaların sayısı artmıştır. Arumugam ve Siva (2013) SMM yöntemi ile finansal verileri kümeleme sureti ile bir öngörü çalışması yapmışlardır. Saidane ve Lavergne (2006) finansal zaman serileri için SMM ile bir modelleme önerisinde bulunmuşlardır. Hassan ve Nath (2005) ve Hassan ve diğerleri (2007) çalışmalarında SMM ile finansal zaman serisi öngörüsünde bulunmuş başarılı sonuçlar elde ettiklerini belirtmişlerdir. Lajos (2010) doktora tezinde finansal olasılıkları kullanarak finansal zaman serileri için öngörü yapmıştır. Jabbour ve Maldonado (2007) dünyanın önde gelen borsa endeksleri için öngörüde yine finansal olasılıkları kullanarak gerçekleştirmiştir. Öngörülerini YSA kullanılarak elde edilenler ile karşılaştırmış, saklı Markov uzmanlarının YSA'ya göre çok daha iyi öngöründe bulduklarını belirtmişlerdir.

Her ne kadar SMM'nin az varsayımla çalıştığı söylene de modelin kurulumundaki şartların sağlanması gerekmektedir. Bu nedenle çalışmada ilk önce finansal zaman serilerine özelleştirilerek kullanılacak olan bu yöntemin verilerinin nasıl düzenleneceğine karar verilmesi gerekmektedir. Veriler sürekli uzaydadır. Oysa standart SMM kesikli uzayda tanımlanmış bir modeldir. Bunun için makine öğrenmesinde sıkça kullanılan bir yöntem *kesiklileştirilme işlemi* (vector quantization) verilere uygulanarak veriler kesiklileştirilmiştir.

Kesiklileştirme işleminde gruplarda oluşacak veri sayısının homojenliği ile kesiklileştirmenin yorumlamaya katacağı etki arasında ödünleşim yapılmıştır. Örneğin birçok kişi için %3'lük bir artış/azalış bilgisi %2'lik bir artış/azalış bilgisinden daha önemlidir. Bu nedenle sert değişim tanımı %3 olarak yapılmıştır. %1'liğe kadar olan değişiklikler ise BIST 100 endeks değerleri için olağan değişiklikler olarak düşünülebileceğinden, bu sınır da tüm tanımlarda bulunmaktadır (Çizelge 4.11.).

Çizelge 4.11. Sert deęişim %3 tanımı ile kesiklileştirme tablosu

Hadise	Karşılık gelen deęer	Gerçek deęer
Sert iniş	G1	$-\infty < G \leq -3$
Yüksek iniş	G2	$-3 < G \leq -1$
İniş	G3	$-1 < G \leq 0$
Artma	G4	$0 < G \leq 1$
Yüksek artma	G5	$1 < G \leq 3$
Sert artma	G6	$3 < G \leq \infty$

SMM’de model tanımlaması problemin yapısına ve veriye uygun olarak durum ve gözlem tiplerinin belirlenmesidir. Belirlenen bu durum ve gözlem tipleri problemin gerektirdiđi cevapları verebiliyor ve yorumlamanın uygun bir şekilde yapılmasını sağlıyor olmalıdır.

SMM çok deęişkenli bir yöntem olarak kullanılabilir. Bu çalışmada ise tüm yöntemlerde girdi bilgisi aynı veri ile sağlanacağından parametre olarak sadece endeks deęeri kullanılmıştır. Kesiklileştirilmiş veri kümesinde her bir eleman bir gözleme karşılık gelmek üzere 6 farklı simge kullanılması uygun bulunmuştur (Bkz. Çizelge 4.1.). Durum sayısı ise model seçimi sırasında 2, 4, 6 ve 10 olarak denenmiştir. Durum sayısı arttıkça daha detaylı bilgilere ulaşılırken çözüm süresi uzamakta, buna ek olarak çözümün global deęil de yerel bir deęerde oluşma eğilimi de artmaktadır.

Parametrelerin tahmini BW algoritması kullanılarak yapılmıştır. Finansal zaman serisi uygulamasında BW algoritması ile saniyeler içinde çözüm bulunabilmektedir. Ne var ki BW global maksimumu garanti etmemektedir. Bu yüzden başlangıç deęerleri elde edilecek sonuçlar için hayati önem taşımaktadır. Durum sayısı arttıkça global maksimumu tahmin edebilmek için iyi başlangıç deęerleri daha da önem kazanmaktadır. Başlangıç deęerlerini seçmek için sezgisel yöntem kullanılmıştır. Böylelikle başlangıç deęerleri tahminin önem kazandıđı 4, 6 ve 10 farklı durum seçimlerinde başarılı sonuçlar elde edilmiştir.

Çalışmalara gözlemler ile oluşturulan geçiş matrisi ve başlangıç matrisi elde edilere başlanmıştır. Modellemede yararlanılan geçiş matrisi Çizelge 4.12.’deki gibidir:

Çizelge 4.12. Geçiş matrisi

0,1520	0,2097	0,1003	0,1459	0,2097	0,1824
0,0937	0,1975	0,1907	0,2043	0,2235	0,0903
0,0495	0,2005	0,2230	0,2410	0,2207	0,0653
0,0551	0,2134	0,2074	0,2555	0,2124	0,0561
0,0601	0,1784	0,2171	0,2396	0,2232	0,0815
0,0941	0,1881	0,1683	0,1609	0,2178	0,1708

Başlangıç vektörü ise Markov modeli çerçevesinde Çizelge 4.13.'deki gibi alınmıştır.

Çizelge 4.13. Başlangıç olasılıkları vektörü

0,0733	0,1975	0,1979	0,2225	0,2187	0,0901
--------	--------	--------	--------	--------	--------

Durum matrisi ve gözlem matrisi:

Daha önceki çalışmalardan ve ekonomik şartların dönem uzunluklarını göz önünde bulundurarak başlangıç değerleri farklı durumlar için Çizelge 4.14.'deki gibi alınmıştır.

Çizelge 4.14. Durum matrisi

0,7	0,06	0,06	0,06	0,06	0,06
0,06	0,7	0,06	0,06	0,06	0,06
0,06	0,06	0,7	0,06	0,06	0,06
0,06	0,06	0,06	0,7	0,06	0,06
0,06	0,06	0,06	0,06	0,7	0,06
0,06	0,06	0,06	0,06	0,06	0,7

Araştırmanın model seçimi aşamasında çeşitli modelleri deneyebilmek için farklı durum sayıları ile denemeler yapılmıştır. SMM'nin başlangıç durum matrislerine özellikle 4 durumdan itibaren aşırı şekilde duyarlı olmasından dolayı sonuçlar global maksimum değil lokal maksimumda bulunduğu görülmüştür. Gözlem matrisinin oluşturulmasında bu nedenle yukarıda bahsi geçen sezgisel yöntem kullanılarak ilerlenmiştir. Gözlem matrisi Çizelge 4.15.'da gösterilmiştir.



Çizelge 4.15. Gözlem matrisi

0,107083	0,226436	0,155004	0,152945	0,237144	0,121387
0,107083	0,226436	0,155004	0,152945	0,237144	0,121387
0,107083	0,226436	0,155004	0,152945	0,237144	0,121387
0,01089	0,174622	0,25835	0,307596	0,228872	0,019671
0,01089	0,174622	0,25835	0,307596	0,228872	0,019671
0,01089	0,174622	0,25835	0,307596	0,228872	0,019671

Model parametreleri tahmin edildikten sonra farklı modeller arasında seçim olanağı doğar. Modeller arasında en uygun olanı ekonometride çok kullanılan Akaike ve Schwarz bilgi kriterleri kullanılarak seçilmiştir (Çizelge 4.16.). Akaike ve Schwarz bilgi kriterleri olabilirlik, parametre ve gözlem sayılarını da dikkate alarak modelin veriye ne kadar uygun olduğu konusunda bilgi vermektedir.

Çizelge 4.16. Model seçimi bilgi kriterleri

Model	AIC	BIC
2 Durumlu SMM	13039,72	13068,80
4 Durumlu SMM	13023,26	13123,58
6 Durumlu SMM	12998,26	13223,98
10 Durumlu SMM	13077,18	13704,18

AIC temel alınır 6 durumlu SMM'nin en uygun model olduğu görülür. Bu yüzden çalışmalara bu model ile devam edilmiştir. En uygun model durum parametre tahmin bilgileri Çizelge 4.17'de verilmiştir. Parametre tahminleri gözönüne alındığında piyasa durumunun sık değişmediği ve güncel durumda kalma eğiliminin yüksek olduğu söylenebilir. Ayrıca ayı piyasası, boğa piyasasından daha kalıcı olarak görülmektedir.

Çizelge 4.17. Tahmin edilen durum matrisi

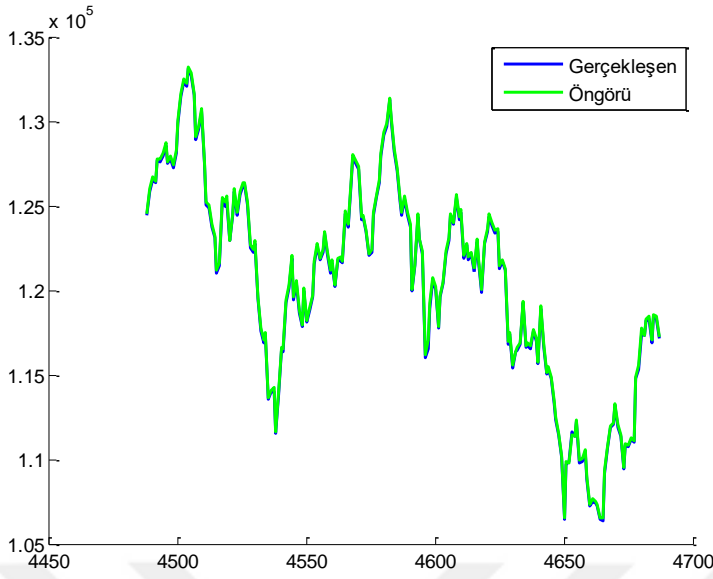
0,983942	0,008029	0,008029	8,21E-22	8,21E-22	8,21E-22
0,006754	0,89606	0,076805	0,006794	0,006794	0,006794
0,006754	0,076805	0,89606	0,006794	0,006794	0,006794
2,01E-17	0,006889	0,006889	0,841896	0,072163	0,072163
2,01E-17	0,006889	0,006889	0,072163	0,841896	0,072163
2,01E-17	0,006889	0,006889	0,072163	0,072163	0,841896

En uygun model gözlem matrisi tahmin bilgileri Çizelge 4.18.'de verilmiştir:

Çizelge 4.18. Tahmin edilen gözlem matrisi

0,194601	0,205685	0,102692	0,107852	0,147876	0,241293
0,107088	0,226436	0,154999	0,152944	0,237141	0,121392
0,107088	0,226436	0,154999	0,152944	0,237141	0,121392
0,01089	0,174623	0,25835	0,307594	0,228873	0,019671
0,01089	0,174623	0,25835	0,307594	0,228873	0,019671
0,01089	0,174623	0,25835	0,307594	0,228873	0,019671

Öngörü için BB'den tahmin edilen parametre değerleri kullanılarak gözlem olabirlikleri üzerinden benzer indeksler bulunmuş bu indeksler üzerinden ağırlıklandırma yöntemi ile gelecek gözlemler tahmin edilmiştir (Şekil 4.20.).



Şekil 4.20. SMM öngörüsü

#### 4.6 YSA Modelleri

YSA modelleri bir çok probleme uygulanabilen bu nedenle de sıkça kullanılan modellerdendir. İlk ortaya atıldığı 1940'lı yıllardan bu yana hala aktif bir çalışma alanı olarak dikkat çekmektedir. YSA'nın uygulamasının bulunmadığı alan yok gibidir. Bu geniş yelpaze içinde yüksek frekansı ile dikkat çeken borsa endeks verileri de bulunmaktadır.

YSA modellerinin bazıları istatistiksel yöntemlerle yakından ilişkilidir. YSA aslında bilinen istatistiksel yöntemlerin aynı matematiksel altyapı fakat farklı bir model olarak ele alınışı olarak görülebilir (Cheng ve Titterington, 1994). Yine de kurulum ve kullanım kolaylığı YSA modellerini popüler hale getirmektedir. Bu bağlamda hisse senetleri endeksleri tahmini de bir çok çalışmada ele alınmıştır. Faraway ve Chatfield (1998) havayolu verileri için yaptıkları çalışmada YSA ile elde ettikleri öngörülerini Box-Jenkins ve Holt-Winters'inkilerle karşılaştırmışlardır. Ayrıca kara kutu özelliğinin ortaya çıkardığı zorluğu aşabilecek bir yöntem de önermişlerdir. Çalışma, onların deyişleriyle bir istatistikçi gözü ile YSA uygulamasının nasıl olabileceğini anlatan bir rehberdir. Kaastra ve Boyd (1996) makalelerinde YSA'nın finansal zaman serilerine nasıl uygulanabileceğini açıklayan detaylı bir çalışma sunmuşlardır. Nygren (2004) tezinde birden fazla farklı hisse senedinin veri seti ile çalışmış kendinden önceki çalışmalara benzer sonuçlar elde edildiğinden bahsetmiştir.

YSA'da tek deęişken ve bu deęişkenin gemiş deęerleri kullanmaktan ziyade birden fazla ve nedensel baęı olan verilerin kullanılması gereklilięi eşitli başka kaynaklarda da önerilmektedir (Rojas, 1996). Khashei ve Bijari (2010), zaman serilerinin öngörüsünde YSA'nın tek başına kullanılmasının her veri seti için yeterince iyi sonuç vermediğini, YSA'nın ARIMA modelinin çıktılarını kullanabilen hibrit bir modelin öngöründe daha iyi sonuçlar verebileceğini gösteren bir alıřma yapmışlardır.

YSA bir makine öğrenmesi yöntemi olarak dięer makine öğrenmesi yöntemlerine benzer özellikler gösterir. YSA veri üzerinde ok az varsayımla alışır. Fakat YSA uygulamalarda kullanılırken önerilen önemli bir ön alıřma veri normalizasyonudur. Veri normalizasyonu zaman zaman dięer makine öğrenmesi tekniklerinde de kullanılan bir ön tekniktir ve bir ya da birden fazla yöntemle yapılabilir. Veri normalizasyonu özellikle girdiler arasındaki farkın yüksek olduęu uygulamalarda önerilir. Normalizasyon farklı modeller için çıktılarının daha az varyasyona sahip olmasını sağladığı gibi aynı zamanda genelde uygulamanın özüm bulma zamanını da kısaltır. YSA için kullanılan veri normalizasyonu yöntemleri řu řekilde sıralanabilir (Priddy ve Keller 2005):

- İstatistiksel ya da z-skoru normalizasyonu
- Min-maks. normalizasyonu
- Sigmoid ya da eşiksiz-en büyük normalizasyon
- Enerji normalizasyon
- Temel bileşenler normalizasyonu

Yukarıdaki maddeler dışında eksik yada dışa düşen veriler için genel bir prosedür belirlenmesi ve uygulanması da veri normalizasyonuna örneklerdir.

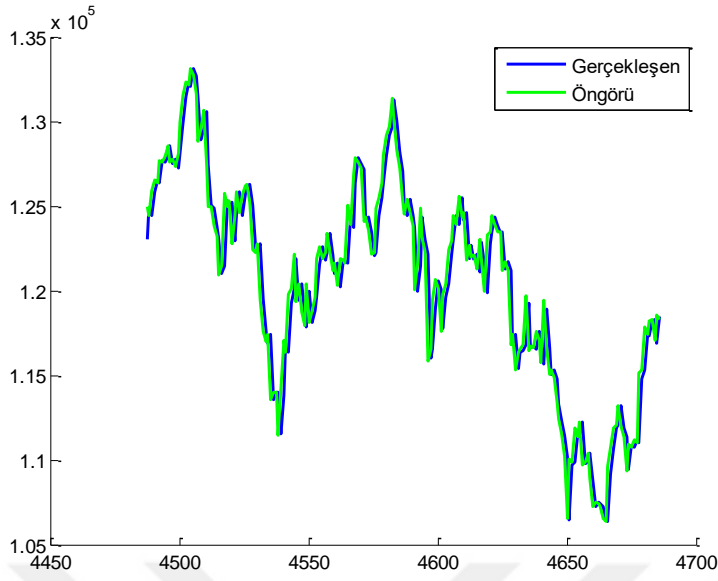
Uygulamada kullanılan veri getiri verileridir. Getiri verileri genel olarak uygun veri özelliklerine sahip olması ile bilinir yine de modelleme yapmadan önce min-maks normalizasyonu ile rassal olmayan verilerin uygulamadan çıkarılmasını sağlayan iki yöntem ile veri normalize edilmiştir. Uygulamada orijinal verilerle alışıldığında aynı mimari YSA yapıları için aynı sonuçların alındığı görülmüş fakat orijinal verilerle yapılan alışmaların

sonuçlanma süresinin normalize edilmiş getiri verilere göre 10 kat mertebesinde arttığı dikkat çekmiştir.

Zaman serilerinin modellenmesinde en çok tercih edilen YSA tipleri ileri beslemeli – geri yayımlı (FFBP) modellerdir. Bu tip YSA'lar öğrenme kuralı olarak delta kuralını temel alırlar. Geri yayımlı modeller diğer YSA modelleri ile karşılaştırıldığında nispeten anlaşılması ve kullanması kolay bir YSA türüdür.

Bu çalışmada FFBP literatürdeki birçok çalışmada olduğu gibi otoregresif yapıyı ele alacak şekilde modellenmiştir. Daha açık olmak gerekirse modelin bir AR yapısını modellemesi beklenmiştir.

FFBP'nin mimari yapısı diğer çalışmalardan bilgi edinilerek ve YSA'daki genel prosedür olan deneme yanılma süreci ile oluşturulmuştur. Bu amaçla 1. ve 2. katmanda toplam 60 nörona kadar çeşitli şekillerdeki mimari ile çalışmalar yapılmıştır. Yapılan bu denemelerde öngörü performansında pek bir farklılık olduğu gözlemlenmemiştir. Bu nedenle her bir katmanda 30 nöron ve aktivasyon fonksiyonlarının tamamı sigmoid alınarak mimari oluşturulmuştur. Uygulamada gecikme sayılarının da BIST 100 getiri verileri için sonuçlara çok fazla etkisi olmadığı saptanmıştır. Bu nedenle Ockham'ın usturası prensibine bağlı kalınarak basit yapıdaki bir YSA kullanılmasına karar verilmiştir. Böylelikle çalışmanın doğrusal zaman serileri bölümünde uygunluğu gösterilen otoregresif yapı YSA'da da kullanılmıştır. Öngörü grafiği Şekil 4.21.'te verilmiştir.



Şekil 4.21. YSA öngörüsü

Bu bölümde 4 adet model ile zaman serileri ele alınmış modelleme yapıldıktan sonra öngörüler gerçekleştirilmiştir. Modellerin öngörü performanslarının MAE performans ölçütüne göre karşılaştırma bilgileri Çizelge 4.19.'te verilmiştir.

Çizelge 4.19. Modellerin performans karşılaştırmaları

Model	MAE
ARMA	1,0339
GARCH	1,0184
SMM	1,0408
YSA	1,0493



## 5 SONUÇLAR

Yüksek frekansları ve kendine özgü özellikleri ile finansal zaman serileri, modellenmesi ve öngörüsü zor olan verilerdendir. Bu çalışmada finansal zaman serileri birden fazla modelle ele alınmıştır. 2. bölümde klasik parametrik modeller, 3. bölümde bu parametrik modellere göre daha az varsayımla kullanılan veri güdümlü makine öğrenmesi yöntemleri incelenmiş 4. bölümde ise modeller ile öngörü yapılmıştır. Daha sonra bu öngörüler bu tür çalışmalarda sıkça kullanılan ortalama mutlak hata ölçütüne göre karşılaştırılmıştır.

Ortalama mutlak hata ölçütüyle çıkan sonuçlara göre tüm modeller benzer bir öngörü performansı göstermişlerdir, yine de modeller içinde öngörü performansı en yüksek olan modelin GARCH olduğu söylenebilir. Çıkan sonuçlardan genelleme yapmak çok gerçekçi olmayabilir. Örneğin finansal zaman serilerinin tamamı için en iyi öngörü yapan model GARCH en kötüsü de yapay sinir ağları genellemesi yanlış bir çıkarım olacaktır. Hatta BIST 100 endeks verisi için en iyi öngörü yönteminin GARCH olduğunu söylemek için bile bu çalışma yeterli değildir. Endeks verisi farklı dönemlerde farklı karakterler sergilemektedir. Çalışmadan çıkan sonuç, BIST 100'ün 2014 sonundan çalışmanın yapıldığı tarih olan 2015 Ekim'e kadar olan dönemde çalışmaya dahil edilen modeller arasında iyi bir öngörü modelinin GARCH olduğudur.

Çalışmada kullanılan modellerin kurulumu ve öngörülerle ilgili ortaya çıkan sonuçlar neticesinde bir kaç izlenim edinilmiştir. Modeller ile ilgili teori, uygulama ile deneyimlenmiştir. Buna göre çalışmada kullanılan klasik parametrik modellerde teori ve teoriye ilişkin varsayımlar daha ön plandadır ve modelleme yaparken teori çeşitli araçlar ve testlerle sınanmaktadır. Testlerden çıkan bilgiler aday modellerin ortaya çıkmasına ve model seçimine yön vermektedir. Diğer bir deyişle modelin varsayımları modellemede kullanılmaktadır. Bu modellerde yöntemi kullanan araştırmacı modeli oluşturan teori ve varsayımlara da aşına olmak durumundadır. Makine öğrenmesi yöntemlerinde ise model ile ilgili istatistiksel teori kullanıcı için parametrik yöntemlerdekinde olduğu kadar ön planda değildir. Model verilerinin yönlendirdiği doğrultuda, eğer veri iyi ise iyi ya da anlamsız ise kötü, bir sonuç çıkaracaktır. Bu sonuçlar YSA'da olduğu gibi ilgilenilen problemin teorisi ile doğrudan bağlantılı olmayan değişkenler ile; bazen de SMM'de olduğu gibi, verinin süreci



dışında model içinde tanımlanan ve yorumu zenginleştiren ek değişkenlerle elde edilir. Diğer bazı makine öğrenmesi yöntemleri ile ise doğrudan öngörü yapmanın mümkün olmadığı bilinmektedir. Makine öğrenmesi yöntemlerinin varsayımlarının az oluşu modellerin daha basit bir teorik alt yapıya sahip olduğu anlamına gelmemektedir.

Makinelerin öğrenmesi hayali ile motivasyon kazanan yapay sinir ağları modelleri aynı zamanda yapay zeka ve yapay öğrenmenin iskeletine sahiptir. Bu nedenle YSA modelleri için söylenen bir çok özellik aslında diğer makine öğrenmesi yöntemleri için de geçerli olacaktır. YSA ile ilgili olarak kaynaklarda sıkça rastlanan, probleme özgü bir yöntemin var olması halinde bu yöntemin kullanılması uyarısı bu çalışmada da doğrulanmıştır. Hem AR hem GARCH yöntemleri YSA'ya göre daha hızlı sonuçlar üretmiş bunun yanında bir sonraki dönem veriyi GARCH modelleri MAE ölçütü göz önüne alındığında daha iyi öngörmüştür. Ortalama ve oynaklık parametrelerinin bilgisayar olmadan dahi tahmin edilebiliyor olması ve bu parametrelerin yorumlamaya elverişli olması da klasik parametrik yöntemlerin diğer avantajlarıdır.

Makine öğrenmesi esnek modeller içerir. Bu çalışmada bu esneklik, konuşma tanıma ve biyometri bilim dalının probleminde sıkça kullanılan bir modelin finansal zaman serileri içinde kullanılmasını sağlamıştır; finansal bir veri seti, kesikleştirme dışında ön çalışma yapmadan ele alınabilmiştir. Uygun modelin elde edilebilmesi için bilgi kriterlerinden yararlanılmıştır. Çıkan sonuçlar finansal zaman serileri öngörülerinde SMM'nin kullanılabilmesine işaret etmektedir. Fakat bir genelleme yapabilmek için farklı veri setlerinin SMM ile ele alınıp sonuçların gözden geçirilmesi daha doğru olur.

Makine öğrenmesi yöntemleri verinin belli varsayımları sağlayıp sağlamadığına bakmaksızın veriyi belli bir modele uydurulabilecektir. Makine öğrenmesi yöntemlerinde modelin ezberlemesi dışında model seçimi ile bir sıkıntı yaşanması beklenmez çünkü bu modeller zaten "verinin konuşmasını" sağlayacaktır. Bu yöntemlerde karşılaşılabilen modelin ezberlemesi probleminin ise belli önlemlerle önüne geçilebilir. Klasik parametrik yöntemlerde modeli doğru seçmek ise kritik öneme sahiptir diğer türlü model varsayımları sağlanmayacak bu da parametre tahminleri ile ilgili sıkıntılara yol açacaktır. Fakat klasik parametrik yöntemlerde model doğru seçildiğinde bu modelin makine öğrenmesi

yöntemlerine göre daha az varyansa sahip uydurulmuş veriler üretmesi beklenir çünkü model eğer önceden tanımlanabilmişse bu bilgi baştan modellemeye dahil edilmiş olur. Klasik yöntemlerde, sezgisel de olsa uygun model seçimi için birden fazla yöntem bir modelleme adımı olarak belirlenmiştir. Bununla birlikte birinci moment göz önüne alındığında ve bilhassa doğrusal özellik göstermeyen verilerle çalışıldığında makine öğrenmesi yöntemleri modelleme açısından daha başarılı olabilirler.

Yine makine öğrenmesi yöntemleri model ezberlemesi ile de ilişkili fakat daha kritik olmak üzere, modellemenin yapılabilmesi için yüksek sayıda ve süreci iyi yansıtan veriye ihtiyaç duyarlar. Diğer türlü çalışılan alandaki teori ile çelişen sonuçlar elde etmek dahi mümkün olacaktır.

Çıkan sonuçların benzer olduğu göz önüne alındığında finansal zaman serilerini ele alan klasik parametrik modeller ile burada incelenen makine öğrenmesi yöntemlerinin iyi birer rakip olabileceğini söylemek bir açıdan doğru ise de farklı bir açıdan eksik bir yaklaşımdır. Aslında bu modeller farklı veriler için farklı performanslar sergileyebilecek ve birbirlerinin yorum gücünü artıracak tümsel modellemenin bir parçası olarak da görülebilir. Örneğin veriyi doğrudan ele alabilen makine öğrenmesi yöntemleri klasik yöntemlerin model uygunluğu hakkında fikir vermek amacı ile kullanılabilir. Ya da SMM ve YSA'nın önemli bir eksikliği varyans analizi ilgili çıkarımlar ARMA ve ARCH/GARCH yöntemleri ile elde edilebilir.



## KAYNAKLAR

- Aizenman, J., Marion, N. (1998). Volatility and Investment: Interpreting Evidence from Developing Countries. *Economica*, 66, 157-159.
- Akdi, Y. (2012). *Zaman Serileri Analizi*. 3. Baskı. Gazi Kitapevi.
- Alpaydın, E. (2011). *Yapay Öğrenme*, Boğaziçi Üniversitesi Yayınevi.
- Anderson D. R., Sweeney J. D. and Williams T. A. (2011). *Statistics For Business And Economics*. Onbirinci baskı. South-Western, Cengage Learning.
- Barquin, P. J. A. (2001). Ramón y Cajal: a century after the publication of his masterpiece. *Endeavour*, 25(1):13-17.
- Baum, L. E., Petrie, T., Soules, G., & Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics*, 41(1), 164-171.
- Beale, R., Jackson, T. (1990). *Neural Computing: An Introduction*. Adam Hilger IOP Publishing Ltd., pp. 1-13.
- Berument, H., Köse, N. and Şahin, A. (2010). Seasonal Patterns of Inflation Uncertainty for the US Economy: An EGARCH Model Results. *The IUP Journal of Monetary Economics*, 8 (7 - 22).
- Bilmes, J. A. (1998). A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models. *International Computer Science Institute*, 4(510), 126.
- Bilmes, J. A. (2006). What HMMs Can Do. *IEICE Transactions on Information and Systems*, e89–d(3).
- Bollerslev, T. (1986). Generalized Autoregressive Conditional Heteroskedasticity. *Journal of Econometrics*, 31, 307-327.
- Borman, S. (2009). The Expectation Maximization Algorithm A short tutorial.
- Borman, S. (2004). The expectation maximization algorithm-a short tutorial. *Submitted for Publication*, 1-9.
- Box, G. E. P., Jenkins, G. M. (1976). *Time Series Analysis Forecasting and Control*. Holden-Day.
- Brockwell, P. J., Davis A. R. (2002). *Introduction Time Series and Forecasting* (Second Edition). Springer Texts in Statistics.
- Can, M. (2009). *İşletmelerde Zaman Serileri Analizi ile Tahmin*. İstanbul Üniversitesi Sosyal Bilimler Enstitüsü, İstanbul.

- Carter, R. (2009). *The Human Brain Book*. Dorling Kindersley Limited.
- Ceppellini, R., Siniscalco, M. and Smith, C.A. (1955). The Estimation of Gene Frequencies in a Random - Mating Population. *Annals of Human Genetics*, 20, 97–115.
- Chan, H. N. (2002). *Time Series Applications to Finance*. John Wiley and Sons.
- Cheng, B. and Titterington, D. M. (1994). Neural Networks A Review from a Statistical Perspective. *Statistical Science*, 9(1), 2-30.
- Collins, M. (1997). *The EM algorithm*.
- Cook, R. D., Weisberg, S. (1994). *An Introduction to Regression Graphics*. John Wiley & Sons, Inc.
- Cryer, J. D., Chan, K. S. (2008). *Time Series Analysis* (Second Edition). Springer Texts in Statistics.
- Çınlar, E. (1975). *Introduction to Stochastic Processes*. Prentice-Hall Inc.
- Data and Analysis Center for Software. (1992). *Artificial Neural Networks Technology*. Anderson, D., McNeil, G. 1-3,17-19, 22,23,32.
- Dembo, A. (2008). Basılmamış kitap/ders notu, 49.
- Demidenko, E. (2013). *Mixed Models* (Second Edition). John Wiley & Sons, Inc.
- Do, C., Batzoglou, S. (2008). What is the expectation maximization. *Nature Biotechnology* 26(8).
- Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1), 38.
- Engle, R. F. (1982). Autoregressive Conditional Heteroscedasticity with Estimates of The Variance of United Kingdom Inflation. *Econometrica*, 50(4), 987-1007.
- Enders, W. (2015). *Applied Econometric Time Series*. 4. Basım. John Wiley and Sons
- Eubank, R., L. (1999). *Nonparametric Regression and Spline Smoothing* (Second Edition), Marcel Dekker Inc.1-10.
- Fahrmeir, L., Tutz, G. (1994). *Multivariate Statistical Models based on Generalized Linear Models*. Springer-Verlag
- Faraway, J., Chatfield, C. (1998). Time Series Forecasting with Neural Networks: A Comparative Study Using Airline Data. *Journal of Applied Statistics*, 47(2), 231-250.
- Fausett, L., V. (1993). *Fundamentals of Neural Networks: Architectures, Algorithms and Applications*. Pearson Education.

- Fink, G. A. (2008). *Markov Models for Pattern Recognition*. Springer-Verlag Berlin Heidelberg.
- Frühwirth-Schnatter, S. (2006). *Finite Mixture and Markov Switching Models*. Springer Science+Business Media, LLC.
- Fyfe, C. (2000). *Artificial Neural Networks and Information Theory*.
- Ghahramani, Z., Jordan, M. I. (1994). Supervised learning from incomplete data via an EM approach. *Advances in neural information processing systems*, 6.
- Gökçe, A. (1998). *ARCH Modelleri Döviz ve Sermaye Piyasalarına bir Uygulama*. Gazi Üniversitesi Sosyal Bilimler Enstitüsü, Ankara.
- Gupta, M. R., Chen, Y. (2010). Theory and Use of the EM Algorithm. *Foundations and Trends in Signal Processing*, 4(3), 223–296.
- Gurney, K. (1997). *An Introduction to Neural Networks*. UCL Press.
- Hamilton, J. D. (1989). A New Approach to the Economic Analysis of Nonstationary Time Series and the Business Cycle. *Econometrica*, 57(2), 357-384.
- Hamzaçebi, C. (2011). *Yapay Sinir Ağları Tahmin Amaçlı Kullanımı*. Ekin Yayınevi.
- Hartley, H. O. (1958). Maximum Likelihood Estimation From Incomplete Data. *Biometrics*, 14(2), 174-194.
- Hassan, R., Nath, B. (2005). *StockMarket Forecasting Using Hidden Markov Model: A New Approach*. Intelligent Systems Design and Applications, 2005. ISDA '05. Proceedings. 5th International Conference.
- Hassan, R., Nath, B. and Kirley, M. (2007). A fusion model of HMM, ANN and GA for stock market forecasting. *Expert Systems with Applications*, 33(2007), 171–180.
- Haykin, S. (1999). *Neural Networks A Comprehensive Foundation*, Prentice-Hall Inc.
- Hyndman, R. J., Koehler A. B., Ord J. K. and Snyder R. D. (2008). *Forecasting with Exponential Smoothing*. Springer Series in Statistics.
- İnternet: National Instruments - *Using the Model-Based Methods (Advanced Signal Processing Toolkit)*. Web: [http://zone.ni.com/reference/en-XX/help/372656C-01/lvasptconcepts/tsa\\_model-based/](http://zone.ni.com/reference/en-XX/help/372656C-01/lvasptconcepts/tsa_model-based/) adresinden 10 Aralık 2015 tarihinde alınmıştır.
- İnternet: Alyuda Research Company - Forecaster. Web: <http://www.alyuda.com/products/forecaster/neural-network-applications.htm> adresinden 10 Aralık 2015 tarihinde alınmıştır.
- Jamshdian, M., Jennrich, R. I. (1997). Acceleration of the EM Algorithm by Using Quasi-Newton methods. *Journal of the Royal Statistical Society Series B (Methodological)*, Vol. 59, No. 3, 569-587.

- Jabbour, G., Maldonado, L. (2007). Prediction of financial time series with Time-Line Hidden Markov Experts and ANNs. *Wseas Transactions on Business and Economics*, 9(4).
- Jiang, J. (2007). *Linear and Generalized Linear Mixed Models and Their Applications*. Springer Science+Business Media, LLC.
- Jondeau, E., Poon, H. and Rockinger, M. (2000). *Financial Modeling Under Non-Gaussian Distributions*. Springer.
- Kaastra, I., Boyd, M. (1996). Designing a neural network for forecasting financial and economic time series. *Neurocomputing*, 10(1996), 215-236.
- Khashei, M., Bijari, M. (2010). An artificial neural network (p, d,q) model for timeseries forecasting. *Expert Systems with Applications*, 37(2010), 479–489.
- Kneib T., Tutz G. (2010). *Statistical Modelling and Regression Structures*. Springer-Verlag Berlin Heidelberg.
- Kobayashi, H., Mark, B. L. and Turin, W. (2012). *Probability, Random Processes, and Statistical Analysis*. Cambridge University Press.
- Koller, D., Friedman, N. (2009). *Probabilistic Graphical Models*. Massachusetts Institute of Technology.
- Lajos, J. (2011). Computer Modeling Using Hidden Markov Model Approach Applied To The Financial Markets.
- Lakatos, L., Szeidl, L. Telek M. (2013). *Introduction to Queueing Systems with Telecommunication Applications*, Springer Science & Business Media, 56.
- Lobato, J. M. H., Lobato, D. H. and Suarez, A. (2007). *Time Series Models for Measuring Market Risk Technical Report*.
- Louis, T. A. (1982). Finding The Observed Information Matrix When Using The EM Algorithm. *Journal of the Royal Statistical Society Series B (Methodological)*, 44(2), 226-233.
- McLachlan J. G., Krishnan T. (2008). *The EM Algorithm and Extensions* (Second Edition). John Wiley & Sons, Inc.
- Mitchell, T. M. (1997). *Machine Learning*. McGraw Hill.
- Mohri M., Rostamizadeh A. and Talwalkar A. (2012). *Foundations of Machine Learning*. MIT Press.
- Moon, T. K. (1996). The Expectation Maximization Algorithm. *The Expectation-Maximization Algorithm in Signal Processing, IEEE Signal Processing Magazine*, 13(6), 47–60.

- Nygren, K. (2004). *Stock Prediction – A Neural Network Approach*. Yüksek Lisans Tezi. Royal Institute of Technology, KTH.
- Office of Naval Research. (1976). *Panels and Time Series Analysis: MC and AR*. Anderson, T. A.
- Olazaran, M. (1996). *A Sociological Study of the Official History of the Perceptrons Controversy*. Sage Publications, Ltd., 621.
- Öztemel, E. (2012). *Yapay Sinir Ağları* 3. basım, Papatya Yayıncılık Eğitim A.Ş., 26,27,41,49, 97.
- Pankratz, A. (1983). *Forecasting with Univariate Box-Jenkins Models*. John Wiley and Sons.
- Priddy, K. L., Keller, P. E. (2005). *Artificial Neural Networks An Introduction*. SPIE Press.
- Rabiner, L., Juang, B. H. (1993). *Fundamentals of Speech Recognition*. Prentice-Hall International.
- Rabiner, L. R. (1989). A Tutorial On Hidden Markov Models And Selected Applications In Speech Recognition. *Proceedings of the IEEE*, 77(2). 257-286.
- Roche, A. (2012). EM Algorithm and Variants: An Informal Tutorial. *arXiv preprint arXiv*, 1105.1476.
- Rogers, K. (2011). The Brain and the Nervous System. *Britannica Educational Publishing*, 11-19.
- Rojas, R. (1996). *Neural Networks A Systematic Introduction*. Springer-Verlag Berlin Heidelberg.
- Ruppert, D. (2004). Springer Texts in Statistics, *Statistics and Finance*.
- Saidane, M., Lavergne, C. (2006). On Factorial HMMs for Time Series in Finance. *The Kyoto Economic Review*, 75(1), 63–90.
- Samarasinghe, S. (2006). Neural Network for Applied Sciences and Engineering. *Auerbach Publications*, 12,70.
- Schlattmann, P. (2009). *Medical Applications of Finite Mixture Models*. Springer-Verlag Berlin Heidelberg.
- Takezawa, K. (2006). *Introduction To Nonparametric Regression*. John Wiley and Sons.
- Tanner, M. A. (1996). *Tools for Statistical Inference* (Third Edition). New York: Springer-Verlag, Inc.
- Tsay, R. (2010). *Analysis of Financial Time Series* (Third Edition). John Wiley and Sons Ltd.
- Vaseghi, S. (2007). *Multimedia Signal Processing*. John Wiley and Sons Ltd.



Wu, J. (1983). On Convergence Properties of EM Algorithm. *The Annals of Statistics*, 11, 95-103.

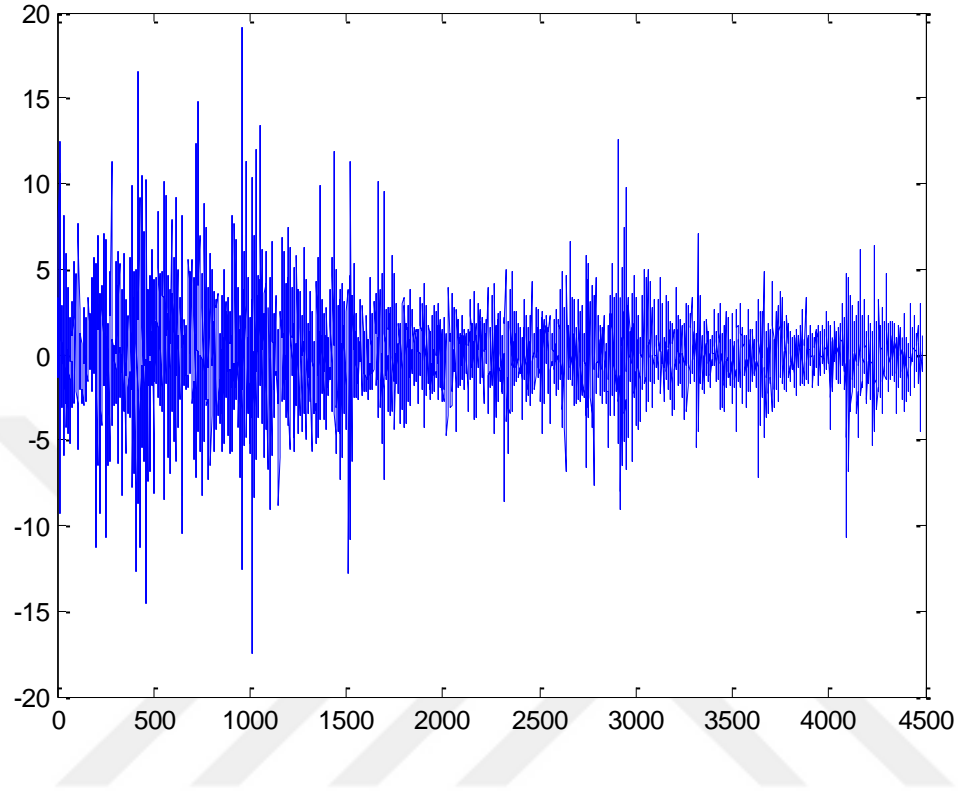
Wu, L. (2010). *Mixed Effects Models for Complex Data*. Chapman & Hall/CRC.

Wyk, J., Lal, A. K. (2008). Risk and FDI Flows to Developing Countries South African. *Journal of Economic and Management Sciences*, 11(4).

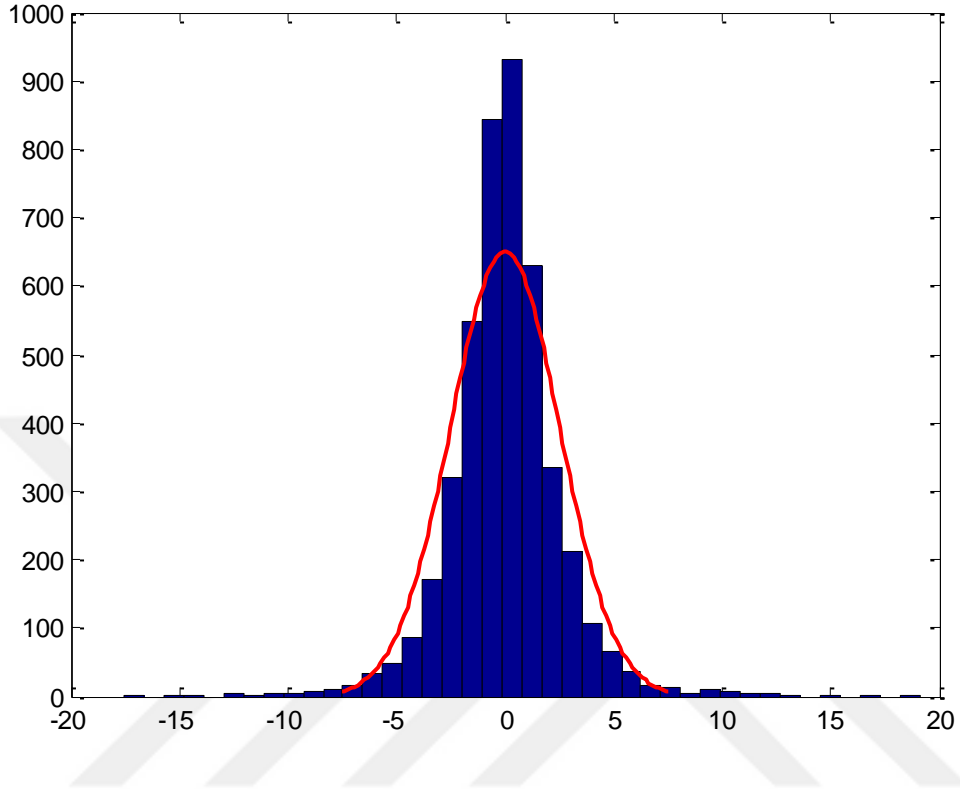




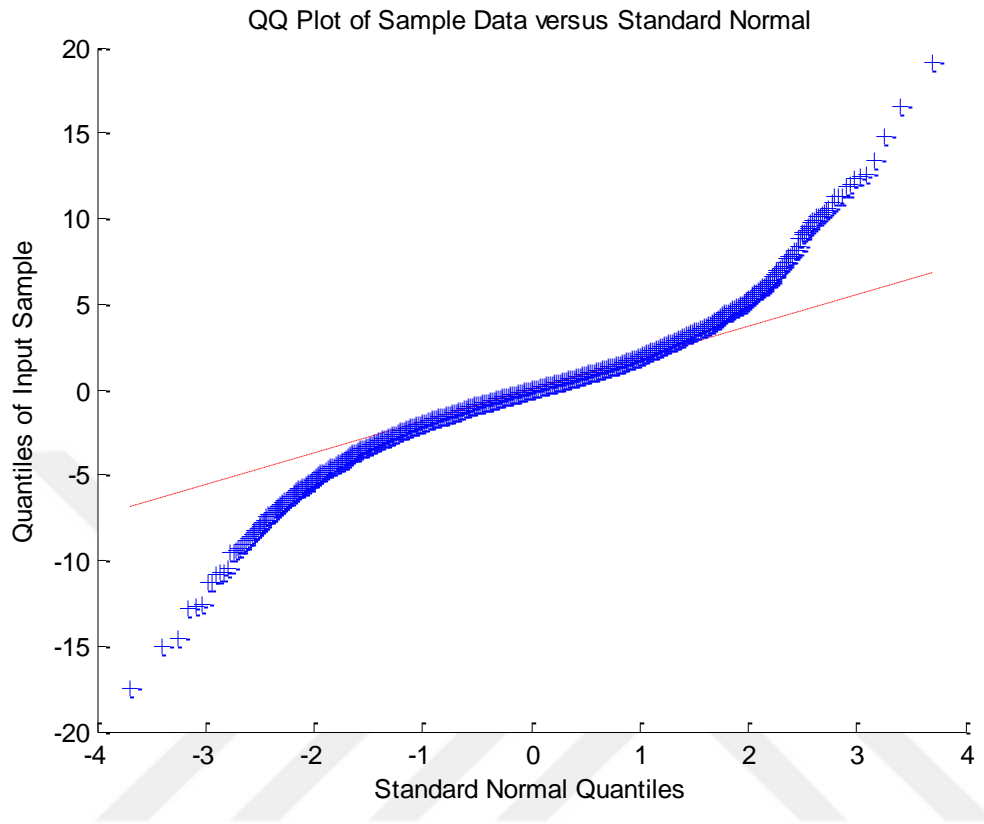
## EK-1. Uygun model kalıntıları



## EK-2. Uygun model kalıntıları histogramı



## EK-3. Uygun model kalıntıları normal dağılım QQ çizgesi



## EK-4. Uygulamada kullanılan hazır MATLAB kodları

```

function [guessTR,guessE,logliks] = hmmtrain(seqs,guessTR,guessE,varargin)
tol = 1e-6;
trtol = tol;
etol = tol;
maxiter = 500;
pseudoEcounts = false;
pseudoTRcounts = false;
verbose = true;
[numStates, checkTr] = size(guessTR);
if checkTr ~= numStates
error(message('stats:hmmtrain:BadTransitions'));
end
[checkE, numEmissions] = size(guessE);
if checkE ~= numStates
error(message('stats:hmmtrain:InputSizeMismatch'));
end
if (numStates ==0 || numEmissions == 0)
guessTR = [];
guessE = [];
return
end
baumwelch = true;
if nargin > 3
if rem(nargin,2)== 0
error(message('stats:hmmtrain:WrongNumberArgs', mfilename));
end
okargs =
{'symbols','tolerance','pseudoemissions','pseudotransitions','maxiterations','verbose','algorithm','trtol','etol'};
dfalts = {[[]] [[]] maxiter verbose " [] []};

```

```

[symbols,tol,pseudoE,pseudoTR,maxiter,verbose,alg,trtol,etol] = ...
internal.stats.parseArgs(okargs, dflts, varargin{:});
if ~isempty(symbols)
numSymbolNames = numel(symbols);
if ~isvector(symbols) || numSymbolNames ~= numEmissions
error(message('stats:hmmtrain:BadSymbols', numEmissions));
end
% deal with a single sequence first
if ~iscell(seqs) || ischar(seqs{1})
[~, seqs] = ismember(seqs,symbols);
if any(seqs(:)==0)
error(message('stats:hmmtrain:MissingSymbol'));
end
else % now deal with a cell array of sequences
numSeqs = numel(seqs);
newSeqs = cell(numSeqs,1);
for count = 1:numSeqs
[~, newSeqs{count}] = ismember(seqs{count},symbols);
if any(newSeqs{count}{:)==0)
error(message('stats:hmmtrain:MissingSymbol'));
end
end
seqs = newSeqs;
end
end
if ~isempty(pseudoE)
[rows, cols] = size(pseudoE);
if rows < numStates
error(message('stats:hmmtrain:BadPseudoEmissionsRows'));
end
if cols < numEmissions
error(message('stats:hmmtrain:BadPseudoEmissionsCols'));

```

```
end
numStates = rows;
numEmissions = cols;
pseudoEcounts = true;
end
if ~isempty(pseudoTR)
[rows, cols] = size(pseudoTR);
if rows ~= cols
error(message('stats:hmmtrain:BadPseudoTransitions'));
end
if rows < numStates
error(message('stats:hmmtrain:BadPseudoEmissionsSize'));
end
numStates = rows;
pseudoTRcounts = true;
end
if ischar(verbose)
verbose = any(strcmpi(verbose,{'on','true','yes'}));
end
if ~isempty(alg)
alg = internal.stats.getParamVal(alg,{'baumwelch','viterbi'},'Algorithm');
baumwelch = strcmpi(alg,'baumwelch');
end
end
if isempty(tol)
tol = 1e-6;
end
if isempty(trtol)
trtol = tol;
end
if isempty(etal)
etal = tol;
```



```
end
if isnumeric(seqs)
[numSeqs, seqLength] = size(seqs);
cellflag = false;
elseif iscell(seqs)
numSeqs = numel(seqs);
cellflag = true;
else
error(message('stats:hmmtrain:BadSequence'));
end
% initialize the counters
TR = zeros(size(guessTR));
if ~pseudoTRcounts
pseudoTR = TR;
end
E = zeros(numStates,numEmissions);
if ~pseudoEcounts
pseudoE = E;
end
converged = false;
loglik = 1;
logliks = zeros(1,maxiter);
for iteration = 1:maxiter
oldLL = loglik;
loglik = 0;
oldGuessE = guessE;
oldGuessTR = guessTR;
for count = 1:numSeqs
if cellflag
seq = seqs{count};
seqLength = length(seq);
else
```

```

seq = seqs(count,:);
end
if baumwelch % Baum-Welch training
% get the scaled forward and backward probabilities
[~,logPseq,fs,bs,scale] = hmmdecode(seq,guessTR,guessE); %%function [pStates,pSeq, fs,
bs, s] = hmmdecode(seq,tr,e,varargin){...}
loglik = loglik + logPseq;
logf = log(fs);
logb = log(bs);
logGE = log(guessE);
logGTR = log(guessTR);
% f and b start at 0 so offset seq by one
seq = [0 seq];
for k = 1:numStates
for l = 1:numStates
for i = 1:seqLength
TR(k,l) = TR(k,l) + exp( logf(k,i) + logGTR(k,l) + logGE(l,seq(i+1)) + logb(l,i+1))./scale(i+1);
end
end
end
for k = 1:numStates
for i = 1:numEmissions
pos = find(seq == i);
E(k,i) = E(k,i) + sum(exp(logf(k,pos)+logb(k,pos)));
end
end
else % Viterbi training
[estimatedStates,logPseq] = hmmviterbi(seq,guessTR,guessE);
loglik = loglik + logPseq;
% w = warning('off');
[iterTR, iterE] =
hmmestimate(seq,estimatedStates,'pseudoe','pseudoE','pseudoTR','pseudoTR');

```

```

%warning(w);
% deal with any possible NaN values
iterTR(isnan(iterTR)) = 0;
iterE(isnan(iterE)) = 0;
TR = TR + iterTR;
E = E + iterE;
end
end
totalEmissions = sum(E,2);
totalTransitions = sum(TR,2);
% avoid divide by zero warnings
guessE = E./(repmat(totalEmissions,1,numEmissions));
guessTR = TR./(repmat(totalTransitions,1,numStates));
% if any rows have zero transitions then assume that there are no
% transitions out of the state.
if any(totalTransitions == 0)
noTransitionRows = find(totalTransitions == 0);
guessTR(noTransitionRows,:) = 0;
guessTR(sub2ind(size(guessTR),noTransitionRows,noTransitionRows)) = 1;
end
% clean up any remaining Nans
guessTR(isnan(guessTR)) = 0;
guessE(isnan(guessE)) = 0;
if verbose
if iteration == 1
fprintf('%s\n',getString(message('stats:hmmtrain:RelativeChanges')));
fprintf(' Iteration  Log LikTransition Emmission\n');
else
fprintf(' %6d %12g %12g %12g\n', iteration, ...
(abs(loglik-oldLL)./(1+abs(oldLL))), ...
norm(guessTR - oldGuessTR,inf)./numStates, ...
norm(guessE - oldGuessE,inf)./numEmissions);

```

```

end
end
logliks(iteration) = loglik;
if (abs(loglik-oldLL)/(1+abs(oldLL))) < tol
if norm(guessTR - oldGuessTR,inf)/numStates < trtol
if norm(guessE - oldGuessE,inf)/numEmissions < etol
if verbose
fprintf('%s\n',getString(message('stats:hmmtrain:ConvergedAfterIterations',iteration)))
end
converged = true;
break
end
end
end
E = pseudoE;
TR = pseudoTR;
end
if ~converged
warning(message('stats:hmmtrain:NoConvergence', num2str( tol ), maxiter));
end
logliks(logliks ==0) = [];
function [tr,E] = hmestimate(seq,states,varargin)
pseudoEcounts = false;
pseudoTRcounts = false;
tr = [];
E = [];
seqLen = length(seq);
if length(states) ~= seqLen
error(message('stats:hmestimate:InputSizeMismatch'));
end
uniqueSymbols = unique(seq);
uniqueStates = unique(states);

```

```

if isempty(uniqueSymbols)
warning(message('stats:hmmestimate:EmptySequence'));
return
end
if isempty(uniqueStates)
warning(message('stats:hmmestimate:EmptyState'));
return
end
if isnumeric(seq)
numSymbols = max(uniqueSymbols);
numStates = max(uniqueStates);
else
numSymbols = length(uniqueSymbols);
numStates = length(uniqueStates);
end
if nargin > 2
okargs = {'symbols','statenames','pseudoemissions','pseudotransitions'};
[symbols,statenames,pseudoE,pseudoTR,setflag] = ...
internal.stats.parseArgs(okargs, {[ ] [ ] [ ]}, varargin{:});
if setflag.symbols
numSymbols = numel(symbols);
if length(symbols) ~= numSymbols
error(message('stats:hmmestimate:BadSymbols'));
end
[~, seq] = ismember(seq,symbols);
if any(seq == 0)
error(message('stats:hmmestimate:SymbolNotInSymbolNames'));
end
end
if setflag.statenames
numStates = numel(statenames);
if length(statenames) ~= numStates

```

```
error(message('stats:hmmestimate:BadStateNames'));
end
[~, states] = ismember(states, statenames);
if any(states == 0)
error(message('stats:hmmestimate:StateNotInStateNames'));
end
end
if setflag.pseudoemissions
[rows, cols] = size(pseudoE);
if rows < numStates
error(message('stats:hmmestimate:StatesPseudoMismatch'));
end
if cols < numSymbols
error(message('stats:hmmestimate:SeqPseudoMismatch'));
end
end
numStates = rows;
numSymbols = cols;
pseudoEcounts = true;
end
if setflag.pseudotransitions
[rows, cols] = size(pseudoTR);
if rows ~= cols
error(message('stats:hmmestimate:BadTransitions'));
end
if rows < numStates
error(message('stats:hmmestimate:StatesPseudoMismatch'));
end
end
numStates = rows;
pseudoTRcounts = true;
end
end
tr = zeros(numStates);
```

```

E = zeros(numStates, numSymbols);
% count up the transitions from the state path
for count = 1:seqLen-1
tr(states(count),states(count+1)) = tr(states(count),states(count+1)) + 1;
end
% and count up the emissions for each state
for count = 1:seqLen
E(states(count),seq(count)) = E(states(count),seq(count)) + 1;
end
% add pseudo counts if necessart
if pseudoEcounts
E = E + pseudoE;
end
if pseudoTRcounts
tr = tr + pseudoTR;
end
trRowSum = sum(tr,2);
ERowSum = sum(E,2);
% if we don't have any values then report zeros instead of NaNs.
trRowSum(trRowSum == 0) = -inf;
ERowSum(ERowSum == 0) = -inf;
% normalize to give frequency estimate.
tr = tr./repmat(trRowSum,1,numStates);
E = E./repmat(ERowSum,1,numSymbols);

```

## ÖZGEÇMİŞ

### Kişisel Bilgiler

Soyadı, adı : ÖZGÜR, Özler  
Uyruğu : T.C.  
Doğum yeri : Ankara  
Medeni hali : Evli  
e-mail : ismim.soyismim@gmail.com

### Eğitim

Derece	Eğitim Birimi	Mezuniyet Tarihi
Doktora	Gazi Üniversitesi Ekonometri Bölümü	2016
Yüksek lisans	Orta Doğu Teknik Üniversitesi Bilgisayar Mühendisliği Bölümü	2007
Lisans	Orta Doğu Teknik Üniversitesi Makina Mühendisliği Bölümü	2003

### İş Deneyimi

Yıl	Çalıştığı Yer	Görev
Nisan 2006 – Nisan 2007	Elba Bilgi Teknolojileri / T.C. SGK	Yazılımcı
Nisan 2007 - ...	T.C. Sosyal Güvenlik Kurumu	Bilişim Uzmanı





*GAZİ GELECEKTİR..*

