



**T.C.**

**HİTİT ÜNİVERSİTESİ**

**LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**

**BANKACILIK VE FİNANS ANABİLİM DALI**

**SERMAYE PİYASASI YATIRIMLARINDA YENİ YATIRIMCI  
DAVRANIŞI "ALGORİTMİK TRADE"**

**Yüksek Lisans Tezi**

**Aysema SERT**

**Çorum - 2022**



**SERMAYE PİYASASI YATIRIMLARINDA YENİ YATIRIMCI DAVRANIŞI**  
**"ALGORİTMİK TRADE"**

**Aysema SERT**

**Lisansüstü Eğitim Enstitüsü**  
**Bankacılık ve Finans Anabilim Dalı**

**Yüksek Lisans Tezi**

**TEZ DANIŞMANI**

**Doç.Dr. Eşref Savaş BAŞÇI**

**Çorum 2022**

Aysema SERT tarafından hazırlanan "Sermaye Piyasası Yatırımlarında Yeni Yatırımcı Davranışı "Algoritmik Trade"" adlı tez çalışması 30/09/2022 tarihinde aşağıdaki jüri üyeleri tarafından oy birliği ile Hitit Üniversitesi Lisansüstü Eğitim Enstitüsü Bankacılık ve Finans Anabilim Dalında Yüksek Lisans tezi olarak kabul edilmiştir.

Doç. Dr. İlker SAKINÇ

.....

Doç. Dr. Eşref Savaş BAŞCI

.....

Dr. Öğr. Üyesi Cihat SAVSAR

.....

Hitit Üniversitesi Lisansüstü Eğitim Enstitüsü Yönetim Kurulunun .../.../..... tarih ve ..... sayılı kararı ile Aysema SERT'in Bankacılık ve Finans Anabilim Dalında Yüksek Lisans derecesi alması onanmıştır.

Unvanı Adı SOYADI

Lisansüstü Eğitim Enstitüsü Müdürü

\* Jüri Başkanının adı yazılmalıdır.

\*\* Tez danışmanının adı yazılmalıdır.

## TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını beyan ederim.

(İmza)

Aysema SERT



# SERMAYE PİYASASI YATIRIMLARINDA YENİ YATIRIMCI DAVRANIŞI "ALGORİTMİK TRADE"

Aysema SERT

ORCID: 0000-0002-4437-7403

HİTİT ÜNİVERSİTESİ

LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

Yüksek Lisans Tezi

Ağustos 2022

## ÖZET

İletişim teknolojilerindeki ilerleme ve borsaların dijitalleşmesi, piyasalara erişim maliyetlerini azaltmıştır. Son yıllardaki önemli gelişmelerden bir tanesi de, yatırımcıların veri işleme hızını artıran, ticaret stratejilerinin uygulama maliyetini azaltan ve menkul kıymet ticaretini hızlandıran algoritmik ticarettir. Alım satım algoritmaları, tüm alım satımların büyük çoğunluğunu yürütebilme potansiyeline sahip olduğundan sermaye piyasalarda bir devrim yaşanmaktadır. Algoritmik ticaret uygulamalarının, piyasa verimsizliklerini ortadan kaldırarak sermaye piyasalarının daha etkili ve verimli çalışmasına yardımcı olacağı düşünülmektedir. Bu çalışmada, basit hareketli ortalama genetik algoritma kullanılmıştır. Algoritma kısa ve uzun vadeli (pozisyon) farklı senaryolarda tasarlanan BİST 30 endeks portföyüne yapılan bir yatırıma uygulanmıştır. Genetik algoritmanın uygulamasının sonucunda yatırımın farklı senaryolardaki getirileri ve performansları analiz edilmiştir. Araştırmanın sonuçlarına göre; BİST 30 endeks portföyüne yapılan farklı senaryolardaki yatırımlarda benzer veya farklı SMA genetik algoritmalar kullanılarak farklı oranlarda pozitif veya negatif getiriler elde edilebilmektedir. Elde edilen sonuçlar, BİST 30 endeks portföyünün satın alınması ve elde tutulmasına uygulanan algoritmik ticaret işlemlerinin normal ticaret işlemlerine göre daha iyi getiriler ve performanslar sağlayabileceğini göstermektedir.

**Anahtar Kavramlar:** Algoritmik ticaret, Basit hareketli ortalama, Genetik algoritma.

**Bilim Kodu:** 118219

# NEW INVESTOR BEHAVIOR IN CAPITAL MARKET “ALGORITHMIC TRADE”

Aysema SERT

ORCID: 0000-0002-4437-7403

HITIT UNIVERSITY

GRADUATE SCHOOL

Master of Scienc Thesis

August 2022

## ABSTRACT

Advances in communication technologies and digitalization of stock markets have reduced the costs of accessing the markets. One of the important developments in recent years is algorithmic trading, which increases the speed of data processing by investors, reduces the cost of implementing trading strategies and accelerates the trading of securities. Trading algorithms have the potential to handle the vast majority of all trades, which has revolutionized the capital markets. It is thought that algorithmic trading applications will help capital markets work more effectively and efficiently by eliminating market inefficiencies. In this study, a simple moving average genetic algorithm was used. The algorithm has been applied to an investment in the BIST 30 index portfolio designed in different short-term and long-term (position) scenarios. As a result of the application of the genetic algorithm, the returns and performances of the investment in different scenarios were analyzed. According to the results of the research; In investments made in the BIST 30 index portfolio in different scenarios, positive or negative returns at different rates can be obtained by using similar or different SMA genetic algorithms. The results show that algorithmic trading transactions applied to the buying and holding of the BIST 30 index portfolio can provide better returns and performances than normal trading transactions.

**Key Terms:** Algorithmic trading, Simple moving average, Genetic algorithm.

**Science Code:**118219

## TEŐEKKÜR

Bu alıőmanın gerekleőtirilmesinde üç yıl boyunca deęerli bilgilerini benimle paylaőan, kullandıęı her kelimenin hayatıma kattıęı önemini asla unutmayacaęım saygıdeęer danıőman hocam, Do. Dr. Eőref Savaő BAŐCI'ya, yine alıőmamda konu, kaynak ve yöntem aısından bana sürekli yardımda bulunarak yol gösteren kıymetli Arő. Gör. Fatih CİNGÖZ'e de sonsuz teőekkürlerimi sunarım.

alıőmam boyunca benden bir an olsun yardımlarını esirgemeyen ve alıőma süresince tüm zorlukları benimle göęüsleyen ve hayatımın her evresinde bana destek olan deęerli annem Ayőegöl SERT'e ve babam Hüseyin SERT'e sonsuz teőekkürlerimi sunarım.



Aysema SERT

# İÇİNDEKİLER

	Sayfa
ÖZET.....	iv
ABSTRACT.....	v
TEŞEKKÜR.....	vi
İÇİNDEKİLER.....	vii
TABLolar DİZİNİ.....	x
ŞEKİLLER DİZİNİ.....	xii
SİMGELER VE KISALTMALAR.....	xiii
GİRİŞ.....	1

## 1. BÖLÜM

### ALGORİTMA VE ALGORİTMİK KARAR VERME

1.1. Algoritmalar.....	9
1.2. Algoritma Sınıflandırılması.....	10
1.2.1. Yönüne göre sınıflandırma.....	10
1.2.2. Uygulandıkları yöntemlere göre sınıflandırma.....	10
1.3. Algoritma İle Bağlantılı Kavramlar.....	13
1.3.1. Büyük veri.....	13
1.3.2. Yapay sinir ağları.....	14
1.3.3. Makine öğrenmesi.....	16
1.3.4. Derin öğrenme.....	18
1.3.5. Yapay zekâ.....	19
1.3.6. Veri madenciliği.....	21
1.3.7. Finansal teknoloji.....	22
1.4. Algoritmik Karar Verme Sistemleri.....	22
1.4.1. Algoritmik karar verme sistemlerinin faydaları ve avantajları.....	23
1.4.2. Algoritmik karar verme sistemlerinin riskleri ve dezavantajları.....	24

## 2. BÖLÜM

### ALGORİTMİK TİCARET

	Sayfa
2.1. Geleneksel Menkul Kıymet Ticareti.....	25
2.2. Algoritmik Ticaret.....	25
2.2.1. Tanımı.....	26
2.2.2. Avantajları ve dezavantajları.....	30
2.3. Algoritmalarda Kullanılmakta Olan Ortak Prosedürler.....	35
2.4. Portföy Optimizasyonu.....	38
2.5. Algoritmalarda Kullanılan Finansal Veriler.....	39
2.6. Algoritmik Ticaret Stratejileri.....	40
2.6.1. Makro ve mikro düzey stratejiler.....	42
2.6.2. Hacim ağırlıklı ortalama fiyat stratejisi (VWAP).....	43
2.6.3. Zaman ağırlıklı ortalama fiyat (TWAP) algoritması.....	45
2.6.4. Hacim yüzdesi algoritması (POV).....	46
2.6.5. Uygulama eksikliği algoritması.....	46
2.6.6. Varış fiyatı algoritması.....	47
2.6.7. Sepet (basket) algoritmaları.....	47
2.6.8. Kara mızrak - likidite arayışı.....	48
2.6.9. Kara kutu (black-box) algoritmaları.....	48
2.6.10. Piyasayla paralel kalma algoritma stratejisi (PEG).....	48
2.6.11. Büyük sipariş (iceberg) gizleme algoritması stratejisi.....	49
2.6.12. Parite ticareti algoritması stratejisi.....	49
2.7. Algoritmik Analiz Aracı Testler.....	49
2.7.1. İşlem öncesi analiz.....	50
2.7.2. Gün içi analizi.....	51
2.7.3. İşlem sonrası analiz.....	51

	<b>Sayfa</b>
2.8. Algoritmik Ticarete Kullanılan Algoritmalar.....	51
2.8.1. Genetik algoritmalar.....	52
2.8.2. Sürü zekâsı algoritmaları.....	54
2.8.3. Apriori birliktelik kuralı algoritması.....	58
2.8.4. Karar ağacı algoritmaları.....	58
2.8.5. Naive-Bayes sınıflandırma algoritması.....	61

### **3. BÖLÜM**

#### **BİST 30 ENDEKSTE SMA GENETİK ALGORİTMA UYGULAMASI İLE ALGORİTMİK TİCARET UYGULAMASI**

3.1. Araştırmanın Yöntemi.....	64
3.2. Bulgular ve Tartışma.....	72
3.2.1. Verilerin öğrenmesinin analizi bulguları.....	72
3.2.2. Birinci senaryo bulguları.....	75
3.2.3. İkinci senaryo bulguları.....	76
3.2.4. Üçüncü senaryo bulguları.....	78
3.2.5. Dördüncü senaryo bulguları.....	80
3.2.6. Beşinci senaryo bulguları.....	82
3.2.7. Altıncı senaryo bulguları.....	84
3.2.8. Yedinci senaryo bulguları.....	86
3.2.9. Sekizinci senaryo bulguları.....	88
3.2.10. Dokuzuncu senaryo bulguları.....	90
<b>SONUÇ VE ÖNERİLER.....</b>	<b>93</b>
<b>KAYNAKÇA.....</b>	<b>98</b>

## TABLULAR DİZİNİ

<b>Tablo</b>	<b>Sayfa</b>
<b>Tablo 3.1.</b> Kısa ve Uzun Pozisyon Senaryoları .....	<b>62</b>
<b>Tablo 3.2.</b> Kısa 3 gün, uzun 34 günlük Senaryo İçin Algoritma Öğrenmesi .....	<b>69</b>
<b>Tablo 3.3.</b> Kısa 3 Gün, Uzun 8 Günlük Birinci Senaryo Uygulanmaksızın 3-21 Günlük Senaryonun Algoritma Öğrenmesi .....	<b>70</b>
<b>Tablo 3.4.</b> Kısa Pozisyon 3 Günlük SMA, Uzun Pozisyon 8 Günlük SMA Genetik Algoritma Sonuçları.....	<b>71</b>
<b>Tablo 3.5.</b> Kısa Pozisyon 3 Günlük SMA, Uzun Pozisyon 21 Günlük SMA Genetik Algoritma Sonuçları.....	<b>72</b>
<b>Tablo 3.6.</b> Kısa Pozisyon 3 Günlük SMA, Uzun Pozisyon 21 Günlük SMA Genetik Algoritma Sonuçları (Kısa pozisyon birinci senaryo olmaksızın) .....	<b>73</b>
<b>Tablo 3.7.</b> Kısa Pozisyon 3 Günlük SMA, Uzun Pozisyon 34 Günlük SMA Genetik Algoritma Sonuçları.....	<b>74</b>
<b>Tablo 3.8.</b> Kısa Pozisyon 3 Günlük SMA, Uzun Pozisyon 34 Günlük SMA Genetik Algoritma Sonuçları (Kısa pozisyon birinci senaryo olmadan) .....	<b>75</b>
<b>Tablo 3.9.</b> Kısa Pozisyon 8 Günlük SMA, Uzun Pozisyon 21 Günlük SMA Genetik Algoritma Sonuçları.....	<b>76</b>
<b>Tablo 3.10.</b> Kısa Pozisyon 8 Günlük SMA, Uzun Pozisyon 21 Günlük SMA Genetik Algoritma Sonuçları (Kısa pozisyon birinci senaryo olmadan) .....	<b>77</b>
<b>Tablo 3.11.</b> Kısa Pozisyon 8 Günlük SMA, Uzun Pozisyon 34 Günlük SMA Genetik Algoritma Sonuçları.....	<b>78</b>
<b>Tablo 3.12.</b> Kısa Pozisyon 8 Günlük SMA, Uzun Pozisyon 34 Günlük SMA Genetik Algoritma Sonuçları (Kısa pozisyon birinci senaryo olmadan) .....	<b>79</b>
<b>Tablo 3.13.</b> Kısa Pozisyon 8 Günlük SMA, Uzun Pozisyon 55 Günlük SMA Genetik Algoritma Sonuçları.....	<b>80</b>
<b>Tablo 3.14.</b> Kısa Pozisyon 8 Günlük SMA, Uzun Pozisyon 55 Günlük SMA Genetik Algoritma Sonuçları (Kısa pozisyon birinci senaryo olmaksızın).....	<b>81</b>
<b>Tablo 3.15.</b> Kısa Pozisyon 21 Günlük SMA, Uzun Pozisyon 34 Günlük SMA Genetik Algoritma Sonuçları.....	<b>82</b>

<b>Tablo 3.16.</b> Kısa Pozisyon 21 Günlük SMA, Uzun Pozisyon 34 Günlük SMA Genetik Algoritma Sonuçları (Kısa pozisyon birinci senaryo olmaksızın) .....	<b>83</b>
<b>Tablo 3.17.</b> Kısa Pozisyon 21 Günlük SMA, Uzun Pozisyon 55 Günlük SMA Genetik Algoritma Sonuçları.....	<b>84</b>
<b>Tablo 3.18.</b> Kısa Pozisyon 21 Günlük SMA, Uzun Pozisyon 55 Günlük SMA Genetik Algoritma Sonuçları (Kısa pozisyon birinci senaryo olmaksızın) .....	<b>85</b>
<b>Tablo 3.19.</b> Kısa Pozisyon 21 Günlük SMA, Uzun Pozisyon 89 Günlük SMA Genetik Algoritma Sonuçları.....	<b>86</b>
<b>Tablo 3.20.</b> Kısa Pozisyon 21 Günlük SMA, Uzun Pozisyon 89 Günlük SMA Genetik Algoritma Sonuçları (Kısa pozisyon birinci senaryo olmaksızın) .....	<b>87</b>
<b>Tablo 4.1.</b> Kısa Pozisyonlu Senaryo Bulgularının Özeti.....	<b>91</b>
<b>Tablo 4.2.</b> Kısa pozisyon (3-8) Birinci Senaryo Olmaksızın Senaryoların Bulgularının Özetleri	<b>92</b>

## ŞEKİLLER DİZİNİ

Şekil	Sayfa
Şekil 1.1. Makine Öğrenmesi Çalışma Modeli .....	25
Şekil 2.1. Algoritmik Ticaret, Yüksek Frekanslı Ticaret ve Nicel Portföy Yönetimi.....	35
Şekil 2.2. Basit Algoritmik Ticaret Algoritması.....	44
Şekil 2.3. Genetik Algoritmanın Ana Akış Şeması .....	59
Şekil 2.4. Sürü Zekasının Yetenekleri.....	61
Şekil 2.5. Karar Ağaçlarından Rastgele Orman Oluşturulmasının Şeması.....	66
Şekil 4.1. Genetik Algoritma Araştırma Modeli.....	67

## SİMGELER VE KISALTMALAR

### Kısaltmalar

BİST	Borsa İstanbul A.Ş.
CCTV	Close Circuit TeleVision (Kapalı Devre Televizyon)
FİNTECH	Finansal Teknoloji
FPGA	Field Programmable Gate Array (Alan Programlanabilir Kapı Dizisi)
GPS	Global Positioning System (Küresel Konumlama Sistemi)
HFT	High-Frequency Trading (Yüksek Frekanslı Ticaret )
LOM	Limited Order Model (Limit Emir Modeli)
POV	Percantage Of Volume (Hacim Yüzdesi Algoritması)
PSO	Parçacık Sürü Optimizasyonu
RFID	Radio Frequency Identification (Radyo Frekansı ile Tanımlama)
RSI	Relative Strength Index (İlişki Genişliği Endeksi)
SMA	Simple Moving Avarege (Basit Hareketli Ortalama)
SOR	Smart Order Router (Akıllı Emir Yönlendiricisi)
TWAP	Time Weighted Avarege Price (Zaman Ağırlıklı Ortalama Fiyat)
VWAP	Volume Weighted Avarege Price (Hacim Ağırlıklı Ortalama Fiyat Staretejisi )

## GİRİŞ

Finansal sektörde dijital dönüşümü sağlayan belirli yeteneklerin geliştirilmesi, rekabet edebilirliğin ve sürdürülebilirliğin sağlanması için en önemli problemlerden biri haline gelmiştir. Çağımızda, finansal kuruluşların sürdürülebilirliği, güncel iş çözümleri gerektiren çağdaş teknolojiler sebebiyle yöneticilerin ve profesyonellerin değişen seviyelerde yeteneklere ve altyapılara sahip olmalarını gerektirmektedir. Finansal teknoloji firmalarının finansal hizmetler sektöründeki pozisyonu küresel boyutlara ulaşmış, çeşitlenmiş ve hızla büyümektedir. Bu büyüyen dijital yenilikler çerçevesinde şirketlerin geleneksel finansal hizmet sağlayıcılarla münasabetleri artık yalnızca rekabetçi ilişkiler boyutunda değil, aynı zamanda işbirlikçi ve ortaklık ilişkileri düzeyinde de yürütülmektedir.

Teknolojik değişiklikler, finansal varlıkların işlem görme şeklinde devrim yaratmıştır. Emir girişinden aracı kuruma ve borsaya kadar ticaret sürecinin her adımı artık son derece otomatiktir ve araçların maruz kaldığı maliyetleri önemli ölçüde azaltmaktadır. Teknoloji, ticaret rekabetlerini ve maliyetlerini azaltarak, daha verimli risk paylaşımını sağlama, riskten korunmayı kolaylaştırma, likiditeyi iyileştirme ve fiyatları daha verimli hale getirme potansiyeline sahiptir. Bu kazanımlar sonuç olarak firmaların sermaye maliyetini azaltabilir. Algoritmik ticaret bu geniş kapsamlı teknolojik değişimin çarpıcı bir örneğidir. Pek çok piyasa katılımcısı artık, belirli ticaret kararlarını otomatik olarak almak, emirleri göndermek ve bu emirleri gönderildikten sonra yönetmek için yaygın olarak bilgisayar algoritmalarına dayalı algoritmik ticareti kullanmaktadır (Hendershott vd., 2011, s. 1).

Finansal piyasalardaki teknolojik gelişmeler arasında, algoritmik ticaret belki de günümüzün en önde gelen devrimi niteliğindedir. Algoritma, girdi değerlerini çıktı değerine dönüştürmek için hesaplamaları kullanan kesin bir adım planı olarak tanımlanabilir. Bireysel veya kurumsal yatırımcılar adına hisse senedi satın almaya veya satmaya tamamen özerk olarak karar veren bu hesaplama algoritmaları hisse senedi piyasalarındaki arz ve talebi etkilemekte, önemi giderek artmaktadır. Algoritmik ticarete yönelecek piyasa hacminin yüzdesi, son yıllarda varlık yöneticileri, yüksek frekanslı tacirler ve hedge fonlarının kullanımı ile büyük ölçüde artmıştır (Glantz ve Kissel, 2013, s. 33).

Gelişmiş yapay zekâ ve kuantum hesaplama ile birlikte algoritmik ticaretin finansal piyasaları tamamen değiştirmesi muhtemel olmakla birlikte; otomasyon devriminin ve algoritmik ticaretin finansal piyasaları nasıl etkilediği hakkında çok az şey bilinmektedir. Kuantum hesaplama ve yapay zekâ hala geleceğe yönelik olup, insan tacirlerin yerini büyük ölçüde bilgisayarlar almakta ancak etkileri hala nicel veriler kullanılarak ölçülememektedir. Algoritmik ticaretin etkilerini ölçmek, finansal piyasaların gelecekte nasıl davranacağına dair fikir verebilir (Verheggen, 2017, s. 7).

Bilgisayar teknolojisindeki gelişmeler, çeşitli diğer teknolojilerin tanıtılmasıyla birlikte fiyat hareketinin gerçek zamanlı analizini kolaylaştırdığından, tüm bunlar algoritmik ticaretin hayatta kalmak için mutlak bir zorunluluk haline gelmesiyle sonuçlanmıştır. Finansal piyasalarda hem alış hem de satış tarafı ve bireysel ve kurumsal yatırımcılar, rekabetçi kalabilmek için algoritmik ticareti kullanmaya başlamak zorunda kalmışlardır (Leshik ve Cralle, 2011, s. 4).

Bu tezin amacı, en son ticaret teknolojisini takip etmekten, yönetmekten ve uygulamaktan sorumlu finansal sektör katılımcılarının finansal piyasalarda uygulanmaya başlayan algoritmik ticaret uygulamalarını, stratejilerini incelemek ve BIST’te uygulanabilirliğini BİT 30 endeks portföyü üzerinde test etmektir.

Algoritmik ticaret uygulamaları sermaye piyasalarında kurumsal yatırımcılara, aracı kurumlara ve yazılım satıcılarına işlem maliyetlerini azaltabilecek, insan hatasını en aza indirebilecek, menkul kıymet ticaretinin verimliliğini ve üretkenliğini artırabilecek yenilikleri geliştirmeleri için fırsatlar sunmaktadır. Teknolojinin gelişmesine ekonomik ve düzenleyici baskıların da etkisi bulunmaktadır. Algoritmik karar verme sistemleri, borsaya kote bir hisse senedini yalnızca bir borsa aracılığıyla işlem görmeye devam etmesi yerine, fiyata hemen erişilebilir olduğunda yatırımcılar için en iyi fiyatı elde etmeyi sağlayan düzenleyici iyileştirmelerden biridir. Algoritmik ticaret, kurumsal yatırımcıların daha az işlem maliyeti, ihmal edilebilir piyasa etkisi ve bilgi sızıntısı ile büyük hisse bloklarının ticaretini yapabilmeleri için daha yaygın bir uygulama haline gelmeye başlamıştır. Sermaye piyasalarında maliyetleri düşürmek ve verimlilikleri artırmak için tasarlanan sürekli yenilikler, yatırım bankacılığı firmalarını ve kurumsal yatırım danışmanlarını alım satım işlemlerini yeniden düşünmeye zorlamıştır. Algoritmalar, hisse senedi alım satımlarını yürütmek için uygun maliyetli yöntemlerdir. Algoritmalar, finansal sektörde faaliyet gösteren işlem aracılığı yapan aracı kuruluşlarda personel sayısı azalmalarına yol açmıştır. Bu otomatik işlemler, daha düşük işlem maliyetleri isteyen müşterilerin talebini karşılayabilecek potansiyele sahiptir. Algoritmik ticarete yeni teknolojilerin büyümesi, finans uzmanları için yeni bir endüstri yaratmıştır. Bu endüstrinin büyümesine yardımcı olmak için uygun protokoller ve verimli süreç altyapısı gereklidir. Aracı kurumlar, yatırım bankaları ve yatırım yöneticileri için algoritmaların işe yaraması, algoritmaların tüm iş ve işlem süreçleri ile uyumlaştırılmasına bağlıdır (Kim, 2007, s. xix-xx).

Demirtaş ve Güngör (2004) çalışmalarında, genetik algoritmanın kullanılmasıyla oluşturulan portföyün riskinin en düşük düzeye indirebilmek için portföyde yer verilmesi gereken hisse senedi sayısını araştırmışlardır. Çalışmanın sonuçlarına göre en düşük seviyede riske sahip olan bir portföy oluşturabilmek üzere 19 adet hisse senedine gerek olduğu saptanmıştır.

Lai vd. (2006), araştırmalarında optimum portföyü oluşturabilmek üzere iki aşamalı bir model kullanmışlardır. İlk aşamada genetik algoritmayı iyi kalitedeki hisse senetlerini seçebilmek için kullanılmışlar, ikinci aşamada ise Markowitz’s modern portföy teorisini esas

olarak genetik algoritmayı kullanmış ve bu iyi kaliteli finansal varlıkların portföye optimum şekilde dağıtılmasını sağlamışlardır. Çalışmayı yürütürken Şangay Borsası'ndan 2 Ocak 2001-31 Aralık 2004 dönemindeki günlük kapanış fiyatlarını kullanmışlardır. Aylık, yıllık ve günlük veriler kullanılarak hesaplanmıştır. Çalışma için yüz adet hisse senedi rastgele seçim yöntemiyle belirlenmiştir. Çalışmanın sonucunda optimal bir portföyde 10, 20 ve 30 hisse senedi olması gerektiğini tespit etmişlerdir.

Zhou vd. (2006), Şangay piyasası endeks getirisinden daha iyi bir getiri elde edebilmek için genetik algoritmayı kullanarak bir çalışma yapmışlardır. Bu amaçla Şangay Borsası'ndan 2 Ocak 2002- 31 Aralık 2004 arasındaki günlük kapanış fiyatlarını kullanmışlardır. Günlük, aylık ve yıllık verileri kullanarak hesaplamalar yapılmıştır. Çalışma kapsamında yüz adet hisse senedi rastgele seçim yöntemiyle belirlenmiştir. Portföye hisse senedi seçiminin verimliliğini hesaplamak üzere genetik algoritma kullanılarak seçilen hisse senetlerinin eşit ağırlıklı portföy getirisini temel değerlendirme ölçütü olarak almışlardır. Çalışmanın sonucunda hisse senedi seçimi için kullanılan genetik algoritma uygulamasının portföy yönetiminde yöneticilere esneklik sağladığı ve yatırımcılara portföye hisse senedi seçiminde yardımcı olmak açısından oldukça faydalı bir yöntem olduğunu göstermişlerdir.

Keskintürk (2007), çalışmasında Markowitz ortalama varyans modelini kullanmış ve optimal getiriye ulaşmak için portföye hisse senedi seçebilmek amacıyla genetik algoritmayı kullanmıştır. Hisse senetlerini temsilen gerçel sayılar kullanılmıştır. Portföye dâhil edilen hisse senetleri dört farklı risk oranına göre seçilerek belirlenmiştir. Risk katsayısı sıfırken en yüksek getiriye sahip olan tek hisse senedi seçilmiştir.

Roudier (2007), araştırmasında genetik algoritmayı kullanarak optimum bir portföy oluşturmayı amaçlamıştır. Optimum portföy oluşturulmasında Dow Jones endeksine dahil hisse senetleri 3 1995-1999, 1999-2003, 2003-2007 dönemleri itibariyle kullanmıştır. Çalışmanın sonucunda genetik algoritmayı kullanarak oluşturulan portföyün getiri ve risk değerleri için Sharpe performans oranı ve Calmar performans oranı kullanarak yönteminin geçerliliğini yükseltmiş ve genetik algoritmanın bu amaçla kullanılabileceğini tespit etmiştir.

Chang ve Liu (2007) çalışmalarında portföy optimizasyonu problemini çözmek üzere genetik algoritma ve bulanık mantık yöntemlerini kullanmış ve sonuçları karşılaştırmışlardır. Çalışmada Şangay borsasının 1997-2000 dönemine ait gün sonu kapanış fiyatları kullanılmış ve her iki yöntemin de yaklaşık aynı sonuçları verdiği görülmüştür. Çalışmanın diğer önemli sonucu genetik algoritma uygulamasının en büyük avantajının zamansal olarak optimum çözüme çok daha erken ulaşması olduğudur.

Chang vd. (2010), Haziran 2003-Ocak 2009 dönemi için Tayvan Borsası 50 endeksindeki fiyat verilerini kullanarak optimum portföyü belirlemek üzere genetik algoritma yöntemini kullanmışlardır. Çalışmada optimal portföyün getirisi ile aynı dönemdeki faiz getirilerini

karşılaştırmışlardır. Araştırmanın sonuçlarına göre riski azaltmak ve yüksek getiri elde etmek üzere portföyün 20 hisse senedinden oluşturulması gerekliliğini tespit etmişlerdir.

Keskintürk vd. (2010), genetik algoritmayı kullanmış ve optimum portföyün seçilmesinde portföyün kaç adet hisse senedinden oluşturulması gerektiğini araştırmışlardır. Bu amaçla 05.01.1999 ile 25.07.2000 tarihleri arasındaki IMKB-30 da işlem gören hisse senetlerinin haftalık kapanış fiyatlarına göre getirilerini kullanılmışlardır. Çalışmanın sonuçlarına göre eşit ağırlıkta portföy seçimi yapıldığı zaman 3 ile 17 adet arasında hisse senedinin seçilmesinin uygun olduğu, serbest ağırlıkta oluşturulan portföylerde ise 6 ile 11 adet arasında hisse senedinin seçilmesi gerektiğini tespit etmişlerdir.

Eshlaghy vd. (2011), Tahran Borsasındaki 2001-2009 dönemi için 146 aktif hisse senedinin aylık verilerini kullanarak; optimum portföyü oluşturmak üzere genetik algoritma ve parçacık sürü optimizasyonu algoritması tekniklerinin sonuçlarını mukayese etmişlerdir. Çalışmanın sonuçlarına göre her iki algoritmanın da portföye hisse senedi seçilmesinde ve portföy optimizasyonunda verimli olarak çalıştığı saptanmıştır. Diğer taraftan parçacık sürü algoritmasına ve genetik algoritmaya göre yakınsama ve zamanlama açısından daha hızlı çalışmakta olduğunu tespit etmişlerdir.

Toloie vd. (2011) Tahran Borsası'nda işlem gören hisse senetlerinin optimizasyonunu sağlamak üzere genetik algoritmayı ve parçacık sürü algoritması tekniklerini kullanmışlardır. Çalışmanın sonucunda oluşturulan genetik algoritmanın hedeflenen sonuca parçacık sürü algoritması yöntemiyle kıyaslandığında daha kısa bir sürede sonuca vardığı tespit edilmiştir. Diğer taraftan genetik algoritmayı kullanılan yöntemlerin en düşük risk düzeyine ulaşmak üzere en az 10 hisse senedine gerek duyulduğunu tespit etmişlerdir.

Özdemir (2011) çalışmasında IMKB-100 endeksteki hisse senetlerinin 15.05.2008 - 26.06.2009 döneminde gün sonu kapanış fiyatlarını optimum portföy oluşturmak üzere kullanmıştır. Bu amaçla öncelikle genetik algoritmayı kullanmış, optimum portföyde olması gereken hisse senetlerinin seçimini yapmış, daha sonra kuadratik programlama dilini kullanarak genetik algoritma tarafından seçilen hisse senetlerinin yatırım tutarlarını hesaplamıştır. Genetik algoritmayı kullanarak hisse senetlerini seçmiş, portföydeki ağırlıklarını ise kuadratik programlamayı kullanarak tespit etmiştir. Genetik algoritmayla eşit ağırlıklı bir portföy oluşturabilmek için optimum portföyün 8 hisse senedi içermesi gerektiğini tespit etmiştir.

Guennoun ve Hamza (2012) Kasablanka Borsası'nda kottaki 48 adet hisse senedinin 24 aylık verilerini kullanarak genetik algoritma aracılığıyla portföy optimizasyonunu sağlamayı amaçlamışlardır. Çalışma sırasında girdi olarak beklenen getiri oranını ve riske maruz değeri esas almışlardır. Beklenen getiri oranının en yüksek, riske maruz değerinin en düşük olduğu durumun, portföy 15 adet hisse senedinden oluşturulduğu zaman gerçekleşeceğini tespit etmişlerdir.

Pandari vd. (2012) çalışmalarında Tahran Borsası'nda işlem gören 50 büyük şirketin hisse senetlerinden en yüksek getiriye sağlayan ve sistematik ve sistematik olmayan riskleri en aza indirgeyen en uygun portföyü oluşturabilmek üzere genetik algoritma yöntemini kullanmışlardır. Mart 2008' e kadarki 72 ay için aylık verileri kullanarak gerçekleştirdikleri çalışmalarında çok amaçlı genetik algoritma tabanlı modellerle Markowitz'in klasik modelini mukayese etmişler ve genetik algoritmayla oluşturulan portföyün Markowitz'in modelinden daha az getiri sağlamakta olduğunu bulmuşlardır.

Soam vd. (2012) çalışmalarında, genetik algoritma kullanarak Nasdaq 100 ve Dow Jones 30 endekslerde işlem gören hisse senetlerinin 2008 ekonomik krizinden hangi derecede etkilendiklerini incelemişlerdir. Bu araştırma için her iki endekse ait 2007, 2008 ve 2009 yıllarına ait hisse senetlerinin gün sonu kapanış fiyatlarını kullanmışlardır. Çalışmanın sonucunda her iki endeksteki hisse senetlerinin 2008 krizinden etkilendiğini tespit etmişlerdir.

Zeren ve Baygın (2015), BİST-30 endeksi kullanarak gerçekleştirdikleri çalışmalarında, amaç fonksiyonunu oluştururken; hedeflenen portföydeki getirinin ve riskin yüzde cinsinden değerini temsil eden "lambda" değerini kullanmışlardır. Çalışmanın sonucunda lambda değerinin 0,2 olması halinde optimal portföyün 18 adet hisse senedinden oluşturulması gerektiğini tespit etmişlerdir.

Çelenli vd. (2015) çalışmalarında klasik ve garanti yakınsamalı parçacık sürü optimizasyonu algoritması yöntemlerini İMKB 30 endeksini oluşturan hisse senetlerinden oluşturulacak portföy optimizasyonu için uygulamışlardır. Elde edilen sonuçlar matematiksel programlama ile sağlanan sonuçlarla karşılaştırılmıştır. Çalışma kapsamında İMKB 30 endeksine ait hisse senetlerinin günlük getiri oranları kullanılmış ve standart parçacık sürü optimizasyonu algoritması, garanti yakınsamalı parçacık sürü optimizasyonu algoritması ve matematiksel programlama yöntemleri kullanılarak üç ayrı portföy oluşturulmuştur. Optimizasyon yöntemleri uygulanarak oluşturulan portföyler, performans ölçütü olarak Sharpe Oranı ile mukayese edilmiştir. En yüksek Sharpe oranına sahip portföyün hem Parçacık Sürü Optimizasyonu algoritması yönteminden hem de garanti yakınsamalı parçacık sürü optimizasyonu algoritması yönteminin uygulanmasından elde edildiği görülmüştür. Her iki portföyün aynı Sharpe oranına sahip olmasına rağmen parçacık sürü optimizasyonu algoritması yöntemi, garanti yakınsamalı parçacık sürü optimizasyonu algoritması yönteminden daha düşük riske ve daha yüksek getiriye sahip portföyü oluşturma başarısını göstermiştir.

Özçalıcı ve Ayriçay (2016) çalışmalarında bilgi işlemsel (yapay) zekâ yöntemlerinin finansal piyasalarda uygulanmasını sağlayarak kullanıcının, karmaşık tahmin modeline daha az müdahale ettiği uzman bir sistemi tasarlamayı amaçlamışlardır. Çalışmalarında kullanılacak parametreleri belirleyebilmek üzere genetik algoritma tabanlı uzman bir sistem tasarlamışlardır. Teknik göstergeler ( $t$ ) gününe ilişkin fiyat ve hacim bilgileri kullanmak

suretiyle teknik hesaplanmıştır. Özelliklerin seçimi ve parametrelerin optimizasyonu genetik algoritma ile eşzamanlı olarak gerçekleştirilmiştir. Uzman sistem (t+1) gününe ilişkin fiyat tahminlerini gerçekleştirmek üzere kullanılmıştır. Çalışmada önerilen uzman sistemin, optimize edilmeyen modele göre daha iyi sonuçlar sağladığı saptanmıştır.

Sezer vd. (2017a) çalışmalarında genetik algoritma uygulanmış ve RSI (Relative Strength Index) optimizasyonu yapmış ve teknik analiz indikatörlerinden SMA (Basit hareketli Ortalama) ile borsanın yönünü tespit etmişlerdir. Akabinde optimize edilmiş değerlerle borsa yönüyle ileri beslemeli yapay sinir ağı beslenmiş ve "Al-Sat-Tut" önerisi veren bir sistem kurgulanmıştır. Genetik algoritmayla borsanın yukarı ve aşağı yönde olduğu durumlar için rastgele başlatılmış örneklem içinden en iyi RSI değerleri tespit edilmiş ve ileri beslemeli yapay sinir ağını beslemek üzere bir listede toplanmıştır. Daha sonra ileri beslemeli yapay sinir ağı endeksten elde edilen "Al-Sat-Tut" etiketleri ile elde edilen veriler gözetimli eğitime tabi tutulmuştur. Değerlendirmelerin yapılması için Dow 30 endeksi kullanılmıştır. Verilerin eğitimi için 1997-2006 arası veriler, test için ise 2007-2017 arası veriler kullanılmıştır. Gerçekleştirilen testler neticesinde geliştirilmiş olan sistemin Al-Tut stratejisiyle karşılaştırılabilir olduğu ya da Al&Tut stratejisinden daha yüksek performans sağladığı tespit edilmiştir.

Sezer vd. (2017b) çalışmalarında teknik analiz indikatörlerinden RSI, MACD (Moving Average Convergence Divergence) ve Williams %R değerleri ile çok katmanlı yapay sinir ağı beslenmiş ve "Al-Sat-Tut" önerileri veren bir sistem tasarlamışlardır. Araştırmada teknik indikatörler optimize edilmemiş ve standart değerleriyle kullanılmıştır. Daha sonrasında çok katmanlı yapay sinir ağı endeksinden sağlanan "Al-Sat-Tut" etiketlerine ilişkin veriler gözetimli eğitime tabi tutulmuştur. Değerlendirmeleri yapmak için Dow 30 endeksi kullanılmıştır. Verilerin eğitilmesi için 1997-2006 dönemi verileri, test için ise 2007-2017 dönemine ait veriler kullanılmıştır. Yapılan testlerin neticesinde geliştirilen sistemin Al&Tut stratejisi ile karşılaştırılabilir bir performans sergilediği; ancak çoğu durumda Al&Tut stratejisinin geliştirilen sistemden daha iyi sonuçlar verdiği tespit edilmiştir. Bunun sebebi, çok katmanlı yapay sinir ağının optimize edilmemiş değerlerle beslenmesi ve uzun test dönemleri için Al&Tut stratejisinin yenilmesinin güç olması ile izah edilmiştir.

Pabuçcu (2019) çalışmasında BIST 100 endeksinin negatif ve pozitif yönlü hareketlerinin tahmin edilmesini amaçlamıştır. Çalışmada yapay sinir ağının, destek vektör makinesinin ve Naive Bayes algoritmasının tahmin gücü performansları karşılaştırılmıştır. Yapılan analizler iki aşamalı olarak gerçekleştirilmiştir. İlk aşamada tahmin modellerinde girdi olarak kullanılan dokuz adet teknik gösterge, borsa endeksinde ait açılış, kapanış, en yüksek ve en düşük fiyatlar, kullanılarak hesaplanmış ve sürekli olan bu teknik göstergeler trendlerine göre sınıflandırılarak yeni bir veri seti oluşturulmuştur. İkinci aşamada ise trend belirleyici veri seti girdi olarak kullanılmış ve seçilen üç makine öğrenme algoritması kullanılarak tahminler yapılmıştır. BIST 100 veri seti 2009-2018 dönemini içeren günlük kapanış

fiyatlarından oluşmaktadır. Yapılan analizler sonucunda, destek vektör makineleri algoritmasının en iyi sınıflandırıcı yöntem olduğu bulgusuna ulaşılmıştır.

Mercangöz (2019) çalışmasında, BİST’de işlem gören ulaştırma sektörüne ait hisse senetleri için, Parçacık Sürü Optimizasyonu (PSO) algoritması kullanılarak minimum riski taşıyacak optimal portföyü oluşturmayı amaçlamıştır. Hisse senetlerinin kodları BEYAZ; CLEBİ, DOCO, GSDDE, PGSUS, RYSAS ve THYAO olarak belirlenmiş, bu hisse senetlerinin fiyat verileri kullanılarak son üç yıllık günlük fiyat hareketlerinden logaritmik getiriler hesab edilmiştir. Bu getiriler kullanılarak minimum riski sağlayacak optimum portföy PSO ve Doğrusal Olmayan GRG teknikleri ile tasarlanmıştır. Çalışmanın sonuçları PSO yönteminin genel olarak daha optimum sonuçlar sunduğunu göstermektedir.

Ünsal (2020) çalışmasında, veri madenciliği tekniklerini kullanarak günlük fiyat değişimine göre birbirleriyle en çok hareket eden hisse senetlerine ait ilişki kurallarının ortaya çıkarılmasını amaçlamıştır. Çalışmada BİST’te işlem gören 408 adet hisse senedine ait 2019 yılında gerçekleşen 249 işlem günü için gün sonu kapanış fiyatı, gün sonu fiyat değişimi ve günlük hacim verileri kullanılmıştır. Veriler üzerinde her bir işlem günü için K-Means yöntemi kullanılarak kümeleme yapılmış, arkasından elde edilen kümelere Birliktelik Kuralı yöntemlerinden Apriori algoritması uygulanarak yıl boyunca birbirleriyle en çok hareket eden hisse senetlerine ait ilişki kuralları belirlenmiştir

Başaran (2021) çalışmasında portföy optimizasyonu problemi için karesel programlama ile genetik algoritma yöntemleri portföy performans ölçütleri bakımından karşılaştırmıştır. Karşılaştırma amacıyla 2019 yılına ait BIST-30 endeksinde işlem gören hisse senetleri kullanılmıştır. Portföy performansını değerlendirmek üzere iki optimizasyon yönteminde de amaç fonksiyonu olarak “sharpe oranı” ile “treynor endeksi” performans ölçütleri kullanılmıştır. Yapılan analizlerin neticesinde genetik algoritma yönteminin her iki portföy performans ölçütüne göre de optimum sonuca ulaştığı görülmüş ve treynor endeksinin sharpe oranına göre daha yüksek performans ölçütü oranına sahip bir portföy oluşturabildiği saptanmıştır.

Ustalı vd. (2021) çalışmalarında, BİST 30 Endeksi’nde işlem gören şirketlerin hisse senetlerinin gelecek fiyatlarını tahmin etmeyi amaçlamışlardır. Bu amaçla ilk olarak BİST 30 Endeksi şirketlerinin 2010-2019 dönemindeki üçer aylık finansal tabloları alınmış, daha sonra bu tablolar aracılığıyla şirketlere ilişkin finansal oranlar hesaplanmıştır. Hisse senetlerinin aylık kapanış fiyatları temin edilmiş ve şirketlere ait finansal oranlarla denk olacak şekilde üçer aylık ortalamaları hesaplanmıştır. Yapay Sinir Ağları, Rastgele Orman algoritması ve XGBoost algoritması kullanılarak her bir şirketin hisse senetlerinin gelecek fiyatları tahmin edilmiştir. Çalışmanın sonuçlarına göre, XGBoost ve Rastgele Orman algoritmaları birbirine yakın sonuçlar vermiş olmalarına rağmen en iyi sonucu XGBoost algoritması vermiştir. Diğer taraftan her iki modelin de yapay sinir ağlarına göre daha yüksek performans sergilediği saptanmıştır.

Durmuşkaya ve Garayev (2017) çalışmalarında, portföy optimizasyonunda genetik algoritma kullanılmasının finansal kriz dönemleri bakımından farklılık gösterip göstermediğini tespit etmeyi amaçlamışlardır. Genetik algoritmanın kullanılmasının kriz dönemlerindeki etkinliğini ortaya koyabilmek üzere, çok amaçlı genetik algoritma yöntemi kullanılarak portföy optimizasyonu yapılmıştır. Çalışmanın sonuçlarına göre kriz döneminde (2008-2011) portföye en fazla dokuz hisse senedi dâhil edilmesi gerektiği, kriz öncesi dönemde ise en fazla getiri sağlayan portföyün yaklaşık iki ile üç hisse senedinden oluşması gerektiği tespit edilmiştir. Krizden sonraki dönemde ise en önemli özellik hisse senetlerinin aylık getirilerinde önemli bir artışın gözlenmesidir. Bütün dönemler itibariyle bakıldığında ise optimal portföyün yaklaşık olarak beş ya da altı hisseden oluştuğu portföyler olduğu tespit edilmiştir. Çalışmanın sonuçlarına göre çok amaçlı genetik algoritmanın kullanılmasının, kriz dönemlerine duyarlı olduğu ve portföy optimizasyonunda pozitif sonuçlar sağladığı saptanmıştır.

Bu çalışma üç bölümden oluşmakta olup; birinci bölümde algoritmanın tanımı, sınıflandırılması, avantajları ve dezavantajları incelenmiştir. İkinci bölümde algoritmik ticaret tanımı, sınıflandırılması, avantajları ve dezavantajları ile stratejileri üzerinde durulmuştur. Üçüncü bölümde ise BİST'te yapılan uygulama gerçekleştirilmiştir.

## 1. BÖLÜM

### ALGORİTMA VE ALGORİTMİK KARAR VERME

#### 1.1. Algoritmalar

Algoritma kelimesi; hesaplama ve problem çözme, sistem oluşturma ve kurallar dizisi gibi anlamlara karşılık gelmektedir. Algoritma, belli bir hesaplanabilir problemler setini çözebilmek üzere iyi tanımlanmış, bilgisayarda uygulanabilen talimatlar dizisidir. Algoritmalar, belli bir seri problem için çözüm bulunması sürecini kolaylaştırabilmek üzere oluşturulmaktadır. Algoritmalar, insanlar tarafından kodlanabilir ya da makine öğrenmesi aracılığıyla kendi kendilerine öğrenebilirler. Bu yönüyle algoritma, bazı hesaplamaların ve donanım kombinasyonunun adımsal sıralanmasıdır. Algoritmanın bilimsel bir yöntem olarak ortaya çıkışının Türk bilim adamı Harezmi'nin çalışmalarıyla başladığı kabul edilmektedir. Algoritma sözcüğü "cebir-algebra" kelimesinden türetilmiş olup batı kaynaklarına ise "algorismus" sözcüğü ile geçmiştir (Ausiello, 2013, s. 27). Algoritmalar; uzun bir süredir borsaya hükmetmekte, müzik bestelemekte, araba sürmekte yeni makaleler yazmakta, uzun matematiksel kanıtlar ortaya koymakta olup yaratıcı yazarlık konusunda da becerileri gelişmeye başlamıştır (Finn, 2020, s. 29).

Algoritma kelimesinin çok sayıda tanımı vardır. Bunlardan bazıları aşağıdaki gibidir (Leshik ve Cralle, 2011, s. 11):

- Algoritma, tanımlanmış bir görevi başarmak için bir dizi eylemi kesin olarak belirleyen bir dizi adımdan oluşan bir plandır. Temel algoritma deterministiktir ve her seferinde aynı girdilerden aynı sonuçları verir.
- Algoritma, bir girdi değeriyle başlayan ve bir çıktı değeri veren bir hesaplama prosedürü için adım adım kesin bir plandır.
- Algoritma girdi olarak değerleri alan ve çıktı olarak değerler üreten bir hesaplama prosedürüdür.

Algoritma, bir çeşit çıktı elde etmek için, bir girdiden hesaplama yapmak üzere kullanılan işlemsel programdır. Algoritmalar, çağımızın teknolojiyle donatılmasında ve üretilmesinde temel altyapıyı sağlamakta, her geçen gün üretimin hızlanmasına ivme kazandırmış ve kültürel hayata yeni pratikler getirmektedir. Bu pratiklerin sonucunda; son on yılda veri bilinci ve dijital benlik gibi kavramlar günlük hayata girmiştir. Algoritmik sınıflandırmalar dili, sosyal teoriyi ve sosyal uzmanlığı güçlü şekillerde dönüştürmekte bu sebeple çağın en önemli fenomenlerinden bir tanesi olarak gösterilmektedir. Algoritmalara temel teşkil eden veya ona işlev kazandıran bileşen ise verilerdir. İnsanlık, tarihin en başından buyana ürettiği bilgileri sınıflandırıp üzerine yeni bilgiler eklemeye çalışmış, doğayı ve bağlı oldukları toplulukları kontrol edebilmek üzere bu üretilen verileri bazen biriktirmiş, bazen manipüle etmiş ve verilere gerek duymuştur. Bu şekilde üretilen veriler günümüzde veri üretimi

araçlarının çeşitliliği ve dijital altyapılarla birlikte hem hacim olarak dev boyutlara ulaşmış hem de verileri işleme şemaları ve şekilleri daha karmaşık olmaya başlamıştır (Toker, 2021, ss. 204-205).

Algoritma, belli bir problemi çözecek yöntemin varolan ya da sonradan tanımlanan veri modeline dayandırılarak adım adım ortaya konulması ve bu modelin bilgisayar ortamında herhangi bir programlama diliyle kodlanmasıdır. Veri modeli, verilerin birbiriyle sırasal ve ilişkisel durumunun ortaya konmasıdır. Algoritma, aslında belirli bir işin veri yapılarıyla algoritmik ifadesini ortaya koymaktadır. Algoritma, belli bir işin ya da problemin sonucuna ulaşmak için arka arkaya uygulanacak adımları ve koşulları kesin bir şekilde ortaya koymakta, genelde tek bir işin görülmesi üzerine yoğunlaşmaktadır. Örnek olarak, bir kümenin elemanlarının sıralanması, bir hat üzerinde en kısa yolun tespit edilmesi, bir matrisin determinantının bulunması gibi algoritmalar tek bir amaca yönelik algoritmalarlardır.

## 1.2. Algoritma Sınıflandırılması

### 1.2.1. Yönüne göre sınıflandırma

Algoritmalar işlemlerin sıralanması yönüne göre seri algoritmalar, paralel algoritmalar ve yinelemeli algoritmalar olarak sınıflandırılmaktadır (Leshik ve Cralle, 2011, s. 22):

**a) Seri Algoritmalar:** Bu algoritmalar gerçekten bir bilgisayarda seri olarak yürütülen bir dizidir, algoritmalar dallanma sağlamak için muhtemelen bazı mantık kontrolleriyle birbirini takip edecek şekilde tasarlanır.

**b) Paralel Algoritmalar:** Bu algoritmalarda çok çekirdekli bilgisayar mimarisini kullanır ve aynı anda birden çok talimatı yürütebilir.

**c) Yinelemeli Algoritmalar:** Bu algoritmalar, "if", "bir süre yap", "için", "sonraki" gibi talimatlarla tekrarlayan yapılar kullanır ve genellikle test etmek ve karar vermek için bazı parametrelenebilir değerlerle, programlama dillerinde bulunanlara benzer yürütme akışını kontrol etmek için tasarlanır.

### 1.2.2. Uyguladıkları yönetime göre sınıflandırma

Algoritmalar, problemlerin çözümü için uyguladıkları yönetime göre sınıflandırılabilir. Algoritmaları sınıflandırmadaki temel amaç, problemlerin çözümünde başvurulabilecek değişik metotları ve alternatifleri tespit edebilmektir (Yılmaz vd., 2016, s. 7). Bu sınıflandırma aşağıdaki gibidir:

- Özyinelemeli Algoritmalar,

- Geri İzlemeli Algoritmalar,
- Böl ve Yönet Algoritmaları,
- Dinamik Programlama,
- Açgözlü Algoritmalar,
- Kaba Kuvvet Algoritmaları.

#### 1.2.2.1. Özyinelemeli Algoritmalar

Kendisini doğrudan ya da dolaylı bir şekilde geriçağırın algoritmalar, özyinelemeli algoritmalarlardır. Bu algoritmalarda, problemler daha küçük ve basit parçalara indirgenmektedir. Her bir küçük parça için oluşturulan çözümlerin birleştirilmesi ile ana problemin çözümüne ulaşılmaktadır (Yılmaz vd., 2016, s. 7). Özyinelemeli algoritmalarla ilgili temel fikir, bir problemi çözmek için, aynı problemin daha küçük bir örneği olan bir alt problemin çözülmesi ve bu küçük örneğin çözümünü kullanarak (kendiliğinden yenileyerek), başlangıçtaki problemin çözülmesidir (<https://tr.khanacademy.org/>). Özyinelemeli algoritmalar, önceden belirlenmiş bir koşul karşılanana kadar kendilerini tekrar tekrar "çağırarak" çalışırlar (Leshik ve Cralle, 2011, s. 22).

#### 1.2.2.2. Geri İzlemeli Algoritmalar

Geri izlemeli algoritmalar, genelde optimizasyon problemlerinin çözümünde kullanılan, problemi çözebilmek için tüm olasılıkları deneyen algoritmalarlardır. Geri izlemeli algoritmalarda çözüm kademeli bir biçimde oluşturulmaktadır. Algoritma çözüme ulaşmak için aşama aşama ilerlerken, muhtemel çözüm yollarının hepsini denemekte, bir sonraki adıma geçmeye çalışmaktadır. Algoritmanın denediği çözüm yolundan bir sonuç alınamaz ise, algoritma bir önceki adımda bulunan diğer olası çözüm yollarına geri dönmektedir (Yılmaz vd., 2016, s. 8).

#### 1.2.2.3. Böl ve Yönet Algoritmaları

Böl ve Yönet algoritması, problemi küçük parçalara ayırıp ayrı ayrı çözmek ve ardından birleştirmek mantığı ile çalışır. Böl ve yönet algoritmaları, problemlerin mümkün olduğu kadar en küçük alt problemlere bölüldüğü ve her bir alt problemin diğerlerinden bağımsız şekilde çözüldüğü algoritmalarlardır (Aygün ve Akçay, 2015, s. 2). Problemin genel çözümüne ulaşabilmek için alt problemlere ait çözümler belirli bir sıra ile bir araya getirilmektedir. Böl

ve yönet algoritmaları, genellikle üç ana aşamada çözüme ulaşmaya çalışmaktadır (Yılmaz vd., 2016, s. 8):

**a) Bölme:** Problemin daha küçük alt parçalara bölüdüğü aşamadır. Problem daha alt parçalara bölünemeyecek ekle gelene kadar, özyinelemeli bir bakış açısıyla bölme işlemleri gerçekleştirilir.

**b) Yönetme:** Problemin alt parçalarının, birbirinden bağımsız şekilde çözüldüğü aşamadır.

**c) Birleştirme:** Problemin alt parçalarına ait çözümlerin, özyinelemeli bir bakış açısıyla birleştirildiği son aşamadır.

#### 1.2.2.4. Dinamik Algoritmalar (Programlama)

Dinamik algoritmalar, karışık problemlerin küçük parçalar halinde çözüldüğü, elde edilen sonuçların bilgisayarın belleğinde bir veri deposunda saklandığı, genel çözüme ulaşılırken de veri yapılarında saklanan sonuçların kullanıldığı bir programlama yöntemidir. Bir problemin dinamik algoritma programlaması kullanılarak çözülebilmesi için problemin alt parçalara ayrılabilmesi ve genel çözümün bu alt parçalardan oluşturulabilmesine gerek vardır. Dinamik programlama yaygın bir şekilde optimizasyon problemlerinin çözümünde kullanılmaktadır (Yılmaz vd., 2016, s. 8):

Dinamik programlama, bir dizi karar verme işleminin optimize edildiği bir matematiksel işlemlerin bütünüdür. Esasen, dinamik programlama, problem çözümünde, problemin ya da problemin bir bölümünün parçalara bölünmesi ve bu parçaların çözülerek, bu çözümlerin depolanması biçiminde bir problem çözme yaklaşımı sunmaktadır. Bu çözümler, gerek duyulduğu zaman, yeniden çözmek yerine, yeniden canlandırılmak yoluyla problemin genel çözümüne eklenerek, nihai çözüme ulaşılmaktadır (Çetin, 2005, s. 140).

#### 1.2.2.5. Açgözlü Algoritmalar

Bir problem için mümkün olan en doğru çözüme ulaşmayı amaçlayan algoritmalara açgözlü algoritmalar denmektedir. Açgözlü algoritmalarda genelde optimum sonuca ulaşılırken, ulaşılan sonuç her zaman için en iyi çözüme karşılık gelmeyebilir. Açgözlü algoritmalar kullanılarak problem çözülmesinde temel yaklaşım, problemin küçük bir alt kümesi için çözüm oluşturmak ve ulaşılan bu çözümü problemin geneline yaygınlaştırmaktır. Algoritma içinde gerçekleştirilen bir seçim, o an için doğru olsa dahi daha sonraki seçimlerde negatif etkiler ortaya çıkarabilir (Yılmaz vd., 2016, s. 8):

Yapıcı sezgisel teknikler ile yerel arama tekniklerinin bileşiminden oluşan açgözlü algoritmalar esasen yapıcı bir sezgisel ile başlangıç çözümünün oluşturulmasıyla başlar. Arkasından önceden tanımlanmış durdurma kriteri sağlanana kadar tekrarlı bir biçimde

işletilen iki ayrı aşamadan meydana gelmektedir. Yıkım aşaması olarak adlandırılan ilk aşamada çözüme ait bazı bileşenler çıkarılır. Yeniden yapım aşaması olarak adlandırılan ikinci aşamada, çıkarılan bileşenler probleme özgü açgözlü tekniklerle tekrar çözüme ilave edilmektedir. İsteğe bağlı olarak çözüm kalitesini artırabilmek üzere başlangıç çözümü ve yeni oluşturulmuş çözüm sonrasında yerel arama uygulanır. Daha sonra yeni oluşan çözümün mevcut çözümle değiştirilip değiştirilmeyeceğine karar vermek üzere bir karar kriteri kullanılmaktadır (Ruiz ve Stützle, 2007, s. 2033).

#### 1.2.2.6. Kaba Kuvvet Algoritmaları

Bir problemin çözümünde, kabul edilebilir bir çözüme ulaşıncaya kadar tüm olasılıkları test eden algoritmalara kaba kuvvet algoritmaları denmektedir. Bu algoritmalar, genelde problemin tanımından hareket ederek en basit çözüm yolunu uygular ve kolaylıkla kodlanabilir. Ancak bu algoritmalarda çok sayıda işlem yapılması gerekir ve çözüm yolu optimum çözümden uzaklaşabilir. Problemdeki verinin hacmi büyüdükçe, kaba kuvvet algoritmasıyla çözüm ihtimali de azalmaktadır. Bir liste içinde eleman aramak, kaba kuvvet algoritmalarının kullanılmasıyla çözülebilecek problemlere örnek olarak gösterilebilir. Listedeki tüm elemanları sırasıyla kontrol edilerek, aranan elemanın listenin içinde olup olmadığına bakılabilir. Listedeki eleman sayısı arttıkça, kaba kuvvet algoritmasının çalışma süresi ve yaptığı karşılaştırmalar da artarak çözüme ulaşması gecikecektir (Yılmaz vd., 2016, s. 9).

### 1.3. Algoritma İle Bağlantılı Kavramlar

Algoritma ile ilişkili çok sayıda kavram bulunmaktadır. Bu kavramlardan en önemlileri olan büyük veri, yapay sinir ağları, makine öğrenmesi, derin öğrenme, yapay zekâ, veri madenciliği ve FİNTECH kavramları aşağıda incelenmiştir.

#### 1.3.1. Büyük veri

Günümüzde veri bir güç ve iktidar şekli olarak vasıflandırılmakta; çok geniş sayıda kullanıcının veri bilgisine ve kar amaçlı veri bilgilerine ve bu verilerle oluşturulan sezgisel algoritmalara veya veri işleme araçlarına sahip olan işletmeler, sezgileri, işlemleri ve kültürü etkileyebilme kabiliyetine sahip olmaktadır (Illiadis ve Russo, 2020, s. 2).

Büyük veri; çoklu, özel kaynaklara sahip, büyük hacimde, karmaşık ve büyümeye devam eden veri kümelerine verilen genel bir addır. Veri toplama ve veri depolama teknolojilerinin hem kalite hem de hacim olarak hızlı bir şekilde gelişmesiyle beraber “büyük veri”, fizik, biyoloji, biyomedikal ve finansal bilimler dâhil olmak üzere bütün bilim dallarında gelişmektedir.

Büyük veri, büyük boyutları ve doğası gereğince içerdiği komplekslik sebebiyle mevcut metodolojiler ya da veri madenciliği yazılım araçlarıyla yönetilemeyen veri kümelerini tanımlamak üzere kullanılmaya başlanmış yeni bir kavramdır (Sütçü ve AYTEKİN, 2019, s. 33). 1999'lu yıllarda veri analistleri tarafından büyük veri olarak tanımlanan toplam verinin boyutu 1 Gigabyte iken; içerisinde bulunan bu dönem "zettabyte" çağı olarak isimlendirilmektedir.

Büyük veri sistemi, her türlü verinin hammadde olduğu bir üretim süreciyle değer üretmektedir. Kredi kartı sensörleri, barkodlar, radyo frekansı ile tanımlama (RFID) ve GPS konum sistemleri uzun yıllardır büyük miktarda veri üretmekteydi. Benzer biçimde iş raporları, e-postalar, müşteri anketleri, internette bulunan metinler gibi yapılandırılmamış veriler analizlerin birer parçasıydı (Gentsch, 2018, s. 11). Büyük veride yeni olan unsur, mobil cihazlarda, sensörlerde, sosyal medya platformlarında, nesnelerin internetinde ve verilerin toplanmasında, işlenmesinde ve kullanılmasındaki yüksek hızdır. Veri sağlayan araçların ve yöntemlerin artması büyük veri kaynaklarını çeşitlendirmiş ve gözetim mekanizmaları olabildiği kadar derinlemesine veri sağlamaya başlamıştır (Zuboff, 2015, s.s. 78-79).

### **1.3.2. Yapay sinir ağları**

Yapay sinir ağları, insan beyninde yer alan nöronlardan esinlenerek tasarlanan matematiksel modellerdir. Yapay sinir ağlarında çok sayıda basit birim, merkezi bir kontrol ünitesine bağlı olmaksızın paralel bir şekilde işlemektedir. Birimler arası ağırlıklar aracılığıyla bilgi depolama işlevi gerçekleştirilmekte ve yeni bilgilerin öğrenilmesi bu ağırlıkların sinir ağlarında güncellenmesi ile temin edilmektedir. Yapay sinir ağlarının davranışı, nöron sayısı, katman sayısı ve katmanları arasındaki iletişim türünü içeren ağ mimarisıyla kategorize edilmektedir (Patterson ve Gibson, 2018, s. 339).

Yapay sinir ağları, insanlara benzer şekilde tecrübe ederek öğrenmekte ve bu öğretileri karar vermede kullanmakta; çalışma ilkeleri de insan beyinde bulunan sinir ağlarına benzemektedir. Yapay sinir ağları katmanlardan oluşmaktadır. Her katmanda yer alan nöronlar bir sonraki katmanda yer alan nöronlar ile bağlantı halindedir. Sistemin amacı, girdi katmanından gelen verilerdeki kalıpları tanıyarak gerçek çıktılara en yakın çıktılara ulaşmaktır. Yapay sinir ağlarıyla alakalı bu süreçler, geleneksel bilgi işleme tekniklerinden farklıdır. Geleneksel karar verme tekniklerinde problemin çözüm yöntemi bilinmekte ve algoritmik olarak tanımlanabilmektedir. Sinir ağlarında ise problemi modellemeye çalışmak yerine verilerden ve hatalardan devamlı öğrenerek, çıktılar yeni öğretilere göre güncellenir. Sistem yeni bilgi öğrendikçe ağırlıkları uyarlayarak performansını (bir hata kriterine göre hesaplanan beklenen çıktı ile gerçekleşen çıktı arasındaki toplam farkı) optimum seviyeye getirmeye çalışmaktadır. Yapay sinir ağının gücü çok sayıda işlemin paralel olarak

gerçekleştirilebilmesinden, birimler arasındaki çok sayıda bağın görev paylaşımından ve birimleri bağlayan ağırlıkların uyarlanabilen karakteristiğinden kaynaklanmaktadır (Tektaş ve Karataş, 2004, ss. 338-339).

Yapay sinir ağları tasarlanırken, kullanılacak değişkenler seçilerek, veriler düzenlendikten sonra kullanılacak ağ modelinin ve ağ konfigürasyonunun (katmanlar ve bu katmanlardaki nöron sayısının) saptanması gerekmektedir. Her katmanda belirli sayıda nöronun sonraki katmanlardaki nöronlarla bağlanmasıyla ağ topolojisi oluşturulmaktadır. Ağ ilk olarak uygun bir başlangıç topolojisiyle çalıştırılır. Başlangıç tipolojisi olarak genelde, ağda bir girdi, bir gizli ve bir çıktı katmanı oluşturmak; girdi katmanındaki birim sayısını girdi değişkenleri sayısı kadar, çıktı katmanındakileri çıktı değişken sayısı kadar almak, gizli katmandaki birim sayısını da girdi ve çıktı birimlerinin toplamının yarısına eşitlemek hüristik uygulaması yapılmaktadır. Aynı veri kümesi için değişik konfigürasyonlar deneyerek hata miktarı azaltılabilir. Ağın kabul edilebilir bir performans düzeyine ulaşmaması halinde gizli katmana yeni nöronlar eklenebilir ya da katman sayısı çoğaltılabilir. Ağlar genellikle ileri beslemeli olarak kurgulanır. İleri beslemeli ağlarda sinyaller girdiden çıktıya doğru tek yönde ilerlemektedir. Geri besleme bulunmamakta, başka bir ifade ile bir katmanın çıktısı aynı katmanı etkilememektedir (Malhotra ve Malhotra, 2003, s. 89).

Yapay sinir ağlarının tıpkı biyolojik sinir ağlarında olduğu şekilde sinir hücreleri bulunmaktadır. Bu yapay sinir hücrelerinin her bir tanesi beş temel unsura sahiptir. Bu unsurlar aşağıdaki gibidir (Öztemel, 2006, ss. 49-51):

**a) Girdiler:** Tasarlanan bir ağın öğrenmesi arzu edilen örnekler tarafından belirlenen ve dış dünyadan gelebileceği gibi ağın içinden başka hücrelerden gelen ya da kendisini ürettiği bilgilerdir.

**b) Ağırlıklar:** Ağırlıklar, bir yapay sinir hücresine gelen girdilerin önem düzeyini ve hücre üzerindeki etkisini gösteren sayısal değerlerdir. Ağırlıkların büyük veya küçük olması o girdinin önemini temsil etmemekle beraber ağırlık değerleri artı, eksi ve sıfır değerlerini alabilir. Artı değer etkinin pozitif olduğunu, eksi değer etkinin negatif olduğunu ve sıfır değeri etkisinin olmadığını göstermektedir.

**c) Toplama Fonksiyonu:** Bir hücreye gelen net girdiyi hesaplamaya yarayan fonksiyondur. Bu fonksiyon, her yapay sinir hücresi için farklı olabileceği gibi bazı sinir hücreleri gruplar halinde aynı fonksiyona da sahip olabilir.

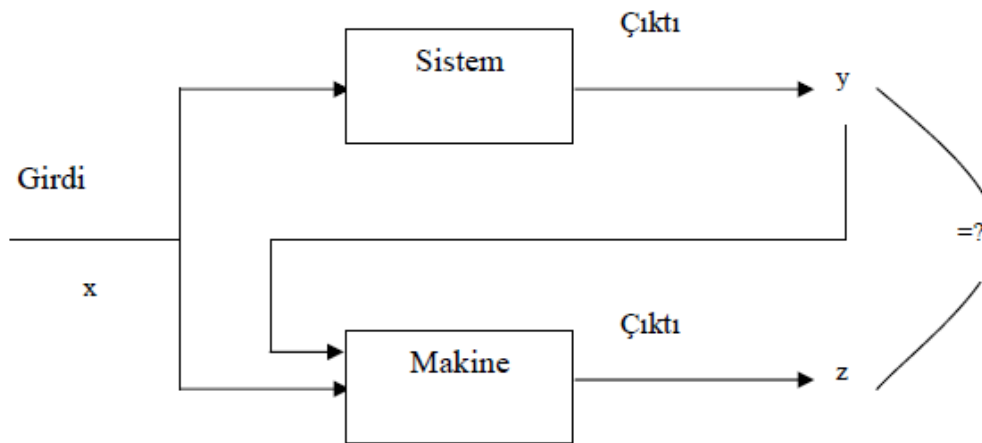
**d) Aktivasyon Fonksiyonu:** Aktivasyon fonksiyonu, hücreye giren girdiyi işleyerek bu girdiye karşılık gelmesi gereken çıktıyı üreten fonksiyondur. Bir sinir ağındaki yapay sinir hücrelerinin her biri farklı aktivasyon formülüne sahip olabileceği gibi bazıları aynı, diğer bazıları farklı aktivasyon formülüne sahip olabilir. Günümüzde yaygın bir şekilde Çok Katmanlı Algılayıcı modelinde sigmoid fonksiyonunun kullanımı tercih edilmektedir.

**e) Hücrenin Çıktısı:** Hücre çıktısı, aktivasyon fonksiyonu işletilerek üretilen çıktı değeridir. Bir sinir hücresinin birden fazla girdisi olmasına karşın yalnızca bir çıktısı olmaktadır. Diğer taraftan hücre ürettiği çıktıyı dış dünyaya sunabildiği gibi tekrar kendisine girdi olarak da kullanabilmektedir.

### 1.3.3. Makine öğrenmesi

Makine öğrenmesi, veri madenciliğinin teknik temelini sağlamakta, işlenmemiş ham verilerden anlaşılır bir şekilde ifade edilebilen ve çeşitli amaçlar için kullanılan veri tabanlarındaki bilgilerden bilgi sağlamak amacıyla kullanılır. Makine öğrenmesi, tahmine dayanan analitik bir modelleme şeklinde de ifade edilmektedir (Witten vd., 2011, s. 17). Makine öğrenmesi, verileri kullanarak tahminlerde bulunarak, karmaşık örüntüleri algılama ve ussal karar verebilme üzerine odaklanan sistemlere verilen genel addır. Günümüzde yapay zekâ uygulamalarında genellikle derin öğrenme yöntemleri kullanılmaktadır. Derin öğrenme, bilgisayarların yapısal olmayan ses, resim ve metin gibi veriler üzerinde insan performansına çok yakın sonuçlar vermesi mümkün olmaktadır (İstanbul Barosu, 2020, s. 4).

Makine öğrenmesi, insan zekâsını çevreleyen ortamdan öğrenerek; bu öğrenmeyi taklit etmek üzere tasarlanmış bilgisayarlı algoritmaların gelişen bir branşıdır. Bu algoritmalar öğrenebilen ve verilerden tahmin yapabilen türdedir. Bu algoritmalar sayesinde stabil program emirlerini takip etmekten ziyade örnek girişler sonucu veri tabanına dayalı tahminler yapabilen ve kararlar verebilen modeller üretilebilir (El Naqa ve Murphy, 2015, s. 3). Aşağıda Şekil 1-1'de makine öğrenmesinin çalışma modeli sunulmuştur:



**Şekil 1.1.** Makine Öğrenmesi Çalışma Modeli

**Kaynak:** Ünsal ve Kaya, 2020: 1194

Makine öğrenmesinin finansal sektördeki önemi çok büyüktür. Finansal sektörde, makine öğrenmesi geleceğe dönük tahminler yapılmasında, yatırımların planlanmasında, enflasyon ve faiz oranlarının tahmin edilmesinde ve piyasadaki dalgalanmaların önceden saptanmasında ya da tahmin edilmesinde, geçmiş verilerden hareketle geleceğe dönük bir öngörü sağlamaya yardımcı olmaktadır. Makine öğrenmesi, büyük veri setleri üzerinde çalışarak yüksek kalitede çıktılar alınabilen algoritmalara sahiptir. Finansal sektörde mevcut verilerin çok büyük hacimde olması da makine öğrenmesinin tercih edilmesini gerektirmekte, geçmiş verilerin kullanılmasıyla sağlıklı sonuçlar alınmasına katkı sağlamaktadır (Can, 2020, s. 10).

Makine öğrenmesi algoritmaları genelde denetlenebilir, denetlenemez ve yarı denetlenebilir şekilde sınıflandırılmaktadır (Bethapudi vd., 2018, ss. 4010-4011):

a) Denetimli makine öğrenme algoritması; gelecekteki olayları tahmin edebilmek üzere geçmişte öğrenilen örnekleri kullanarak yeni verilere uygulanması yöntemidir. Bu öğrenmede öğrenme algoritması çıktı değerleri ile ilgili tahminler yapmak üzere çıkarımlı bir fonksiyon üretmektedir.

b) Denetlenmeyen makine öğrenme algoritması; eğitmek için kullanılan bilgiler sınıflandırılmadığında ya da etiketlenemediğinde kullanılmaktadır. Denetimsiz öğrenme, sistemlerin etiketlenmemiş verilerden gizli bir yapı tanımlayan bir fonksiyonu nasıl ortaya çıkardığını araştırmaktadır. Sistem doğru çıktıyı üretmemekte fakat verileri araştırarak etiketlenmemiş verilerden gizli yapıları tanımlamak üzere veri kümelerinden çıkarsamalar yapabilmektedir

c) Yarı denetimli makine öğrenme algoritmaları ise eğitim için hem etiketli hem de etiketsiz verileri kullandıklarından denetimli ve denetimsiz öğrenme algoritmasının arasında bir yerde konumlanmaktadır. Bu algoritmalar, az miktarda etiketlenmiş veri ve büyük miktarda etiketlenmemiş veri kullanır. Bu yöntemi kullanan algoritma sistemleri öğrenme doğruluğunu ciddi oranda yükseltebilir.

Makine algoritması finansal sektörde kullanıldığında işletmenin maliyetlerini azaltır, iyi bir üretkenlik oluşturur ve gelişen deneyimler marifetiyle gelirin artırılmasına yardımcı olur, daha uyumlu bir çalışma ve güçlendirilmiş güvenlik sisteminin oluşturulmasını sağlar. Günümüzde finansman alanında kantitatif ve büyük verilerin olması finansal sektördeki şirketleri büyük çapta Ar-Ge yatırımları yapılmasına zorlamaktadır. Bu sebeple şirketler ve kurumlar açısından makine öğrenmesini gözardı etmenin maliyeti yüksektir. Finansal alanda uzman kişiler veya kuruluşlar, yatırım kararlarını alırken belli bir teoriyi takip etmeden karar verebilmektedirler. Makine öğrenmesi yardımıyla yatırım uzmanları teknik ve sağlam bir alt yapıya sahip makinelerin vereceği kararlardan faydalanabilirler. Makine öğrenmesi, borsalarda giderek daha da yaygınlaşmaktadır. Makine öğrenmesiyle geçmişteki fiyat bilgilerinden gelecekteki fiyat hareketleri ve eğilimler saptanabilir. Finansal alanda makine

öğrenmesi kullanımı hisse senetleri ve tahviller arasında daha iyi performans gösterenlerin seçilmesine ve portföy oluşturulmasına da yardımcı olmaktadır. Finansal sektörün çok sayıda evresinde, kredi tahsis süreci, kredi notlarının belirlenmesi, varlık yönetimi, risk değerlendirme, yatırım seçimi, fon yönetimi gibi alanlarda makine öğrenmesinden yararlanılmaktadır. Ayrıca yatırım portföyleri makine öğrenmesi teknikleriyle optimize edilebilmektedir (Yılmaz, 2019, s. 191).

#### **1.3.4. Derin öğrenme**

Makine öğrenmesinin bir alt dalı olan derin öğrenmede farklı çok katmanlı derin yapay sinir ağları mimarileri kullanılmaktadır. Derin öğrenme, yapay sinir ağlarına dayalı makine öğrenmesi çeşididir. Yapay bir sinir ağı, her ağın ya da nöronun davranışını etkileyebilen birbirine bağlı bir grup nörondan oluşmaktadır. Derin öğrenmede, birden çok algoritma katmanı insan beyninin katmanlı öğrenme sürecindeki nöronları taklit ederek yığılmaktadırlar. Algoritmaların her birisi, verilerden belli bir niteliği kaldıracak şekilde dizayn edilmektedir. Bu sözde temsil ya da soyutlama, takiben, verilerin başka bir yönünü yine kaldıran başka bir algoritmayı beslemektedir. Temsil-öğrenme algoritmalarının yığılması, derin öğrenme yaklaşımlarının düşük kaliteli, yapılandırılmamış veriler dâhil olmak üzere her türlü veriyle beslenmesine imkân sağlamaktadır. Algoritmaların, verilerin ilgili soyutlamalarını oluşturma yeteneği, sistemin bir bütün olarak bir analiz gerçekleştirmesine müsaade etmektedir. En önemlisi, bu özellik katmanları insanlar tarafından tasarlanmamakta, genel amaçlı bir öğrenme prosedürü kullanılarak verilerden öğrenilmektedir. Bunlara "gizli katmanlar" adı verilmektedir. Bir sinir ağı, temel olarak aşağıdaki üç ayrı işleve bölünmüş düğümler birleşimidir (Kandemir, 2021, s. 63):

- a) Giriş katmanı,
- b) Bir algoritma uygulayarak modeli belirleyen gizli katman ve
- c) Çıktı katmanı.

Derin öğrenmeyi kullanan hesaplama algoritmaları, model tahmin edilmesinde ve veri temsili öğrenmekte fayda sağlamaktadır. Derin öğrenme, uygulandığı amaca bağlı olarak hem denetimli hem de denetimsiz öğrenme şeklinde gerçekleştirilebilir. Derin öğrenme teknikleri kompleks yapıdadır ve genelde kara kutu şeklinde nitelendirilirler. Öngörülen bir çıktı oluşturabilmek için girdilerin nasıl yeniden birleştirildiği her zaman açık olmayıp; bu durumun risk yönetiminde kullanılmasına açık etkileri bulunmaktadır. Karar verme aşamasında kara kutunun varlığı kendine özgü zorluklarına sahip olup; kendi başına bir risk oluşturabilmektedir. Tahmine dayalı derin öğrenme, geleceği anlamakla ve tahmin etmekle alakalıdır, başka bir ifadeyle gelecekte ne olabileceğini tahmin etmek ve anlamak için istatistiksel modelleri ve tahmin yöntemlerini kullanmakla ilgilidir (Tammenga, 2020, s. 225).

İnternetin gelişimiyle beraber dijital ortamda her saniye milyonlarca veri büyük boyutlarda üretilmekte ve depolanmaktadır. Eğitilmek üzere kullanılan veri miktarı arttıkça, klasik makine öğrenmesi algoritmalarının büyük verilerle sonuca ulaşma performansı yeterli olamamaktadır. Makine öğrenmesi algoritmaları hem hızlı çalışmaları hem de hızlı biçimde düzeltme sağlasa da günümüzde internetin yaygın kullanımıyla beraber ulaşılabilen verinin boyutu artmıştır. Sonuca hızlıca ulaşmaktan çok ulaşılan sonucun performans değerinin daha iyi olması önem taşımaktadır. Derin öğrenme modelleri, bu büyük verileri etkili bir biçimde kullanarak verimli sonuçlar üretebilmektedir. Derin öğrenme, sınıflandırma özelliklerini ve görevlerini doğrudan veriden gerçekleştirmeyi öğrenerek işlemektedir. Derin öğrenme sistemleri, bir insan düzeyinde öğrenmeyi sağlamak ve hatta bu düzeyi aşmak için farklı ve çok sayıda sinir ağı mimarisi kullanılmak suretiyle eğitilebilmektedir (Şişmanoğlu vd., 2020, s. 437).

### **1.3.5. Yapay zekâ**

Yapay zekâ görüşünü ilk kez Turing 1950 yılında, insan tarzı zekâyı bütünüyle mekanikleştirme olasılığını ele alarak gündeme getirmiş ve “Makineler düşünebilir mi?” sorusunu zihinlere taşımıştır (Nilsson, 2019, s. 65). Yapay zekâ, çevresini algılayabilen ve vazifesini başarıyla gerçekleştirme olasılığını en üst seviyeye çıkarabilen eylemler gerçekleştiren cihazlar olarak tanımlanır. Yapay zekâ, "öğrenme" ve "problem çözme" gibi insana ait "bilişsel" fonksiyonları taklit edebilen bir bilgisayardır (Buckley vd., 2020, s. 7).

Yapay zekâ, insan gözetimini ve iş süreçleriyle etkileşimi artırabilen, çoğaltabilen ya da en sonunda onun yerini alabilen bir program ya da program paketini ifade eder (Smith, 2020, s. 11). Günümüzde yapay zekâ, büyük veri depolarındaki beklenmeyen değişken ilişkilerini saptamak, nedensellik için beklenen korelasyonları denemek ya da önceden tanımlanmış bir modelin ampirik ihtimalini saptamak için kullanılmaktadır. Yapay zekâ, normal olarak insan zekâsıyla ilgili nitelikleri sergileyen bilgisayar sistemlerinin teorisi ve gelişimini içermektedir. Yapay zekânın ilk amacı, insanlarda akıllı davranışa imkân veren bileşenleri daha iyi anlamak ve sonrasında insan-makine sistemlerini geliştirmektir. Yapay zekâ, derin öğrenme, makine öğrenmesi, görüntü tanımlaması, doğal dil işleme, kavramsal bilişim, zekâ yükseltme, kavramsal yükselme, makine ile arttırılmış zekâ ve arttırılmış zekâ gibi çok sayıda kavramla birlikte kullanılmaktadır. Yapay zekâ, bütün bu kavramları içermektedir (The Institute of Internal Auditors, 2017, s. 1).

Yapay zekâ, düşünmeyi insanların yerine yapabilecek bir makinenin hayal edildiği günlerden bugüne, bir hayal olmanın ötesine geçmiştir. 17. yy.'da Pascal'ın ve Leibniz'in mantıksal pozitivist çalışmalarıyla geliştirilen ve insanların yerine hesaplamalar gerçekleştirebilen makineler yardımıyla doğan yapay zekâ, günümüzde gündelik hayatın birçok sahasında kullanılan teknolojileri üretir duruma gelmiştir. İnternette arama yaparken, müzik dinlerken,

film izlerken, ulaşım araçlarını kullanırken, sağlık sektöründen finans sektörüne kadar birçok alanda hizmet verirken yapay zekâ teknolojilerinden faydalanılmaktadır. Bu kapsamda insanın zekâsını taklit eden, insanmış gibi düşünebilen makineler üretilmesi amacıyla, çok sayıda programlar yazılmıştır. Yapay zekânın sosyal bilimlerde farklı disiplinler tarafından sahiplenmesi ise yapay zekânın bir sektör haline gelmesi ve internetin yaygınlaşması ile beraber geniş kitleler tarafından kullanılmaya başlamasıyla başlamıştır (Kurzweil, 1985, s. 258).

Belirli bir amaca ulaşmak üzere tasarlanmış ve yapay zekânın esasını teşkil eden algoritmalar, kullanım sahalarına, tasarlandıkları yöntemlere ve işleme şekillerine göre farklı türlere ayrılmaktadır. Yapay zekâ algoritmaları ise dijital asistanları oluşturan algoritmalarlardır. Yapay zekâyı besleyen yapay öğrenme algoritmalarının temeli, insan beyninin bilgi işleme sisteminden uyarlanan yapay sinir ağlarıdır (Say, 2019: 98). Aslında, yapay zekâ, algoritma üretebilen ve kendi kendisine işleyen bir sistemi refere etmekte, fakat bu, insan eliyle üretildiği ve insan aklının bir ürünü olmaktan öteye geçemeyeceği gerçeğini değiştirmemektedir (Nilsson, 2019, s. 385).

Yapay zekâ, geniş bir disiplinlerarası bilim dalıdır. Yapay zekânın esasını oluşturan bilimsel disiplinler içinde mantık, istatistik, karar teorisi, nöroloji, psikoloji, dilbilimi, sibernetik ve bilgisayar mühendisliği gibi disiplinler yer almaktadır (Howard, 2019, s. 918).

Yapay zekâ uygulamaları, finans sektöründen, pazarlama sektörüne; üretimden ulaşımaya değin insanın olduğu her sahada yer almaktadır. Doğadaki varlıkların akıllıca davranışlarının yapay şekilde taklit edilmesinden hareket eden ve bu doğrultuda işini mükemmelce yapan, canlı sistemleri ve insan beynini model alan yapay zekâ uygulamaları; günlük hayatta farklı alanlarda ürünler vermekle birlikte, tahmin etme, veri sınıflandırma, veri kümeleme gibi amaçlar için de kullanılmaktadır (Riedl, 2019, s. 33). Karar verme uygulamalarında çok sayıda yapay zekâ uygulaması ve sistemleri kullanılmaktadır. Başlıca yapay zekâ uygulamaları arasında; uzman sistemler, bulanık mantık, genetik algoritmalar ve yapay sinir ağları sayılmaktadır (Demirhan, Kılıç ve İnan, 2010, s. 31).

Bilhassa, finansal hizmetler sektöründeki potansiyel uygulamalar için çok mühim olan yapay zekâ, portföy optimizasyonu, menkul kıymetlerin fiyat tahmini, al-sat-elde tut kararlarının verilmesi, siber suçların ortaya çıkarılması, kara para aklama, suçunun ve terör finansmanı suçunun ortaya çıkarılması, manipülasyonla ve dolandırıcılıkla mücadelede artan bir şekilde uygulanmaktadır. Yapay zekânın büyük ve yapılandırılmamış veri depolarındaki kalıpları hızlı bir biçimde tespit edebilme kabiliyeti, sadece finansal kararların verilme hızını ve isabetini artırabilmek için değil, aynı zamanda verilerin yoğun olduğu faaliyetleri otomatik hale getirmek ve geliştirebilmek için büyük bir potansiyele sahiptir. Bu şekilde işlem maliyetler düşürülürken riskler azaltılmakta ve getirinin en çoklanması mümkün olmaktadır (Kandemir, 2021, s.76).

### 1.3.6. Veri madenciliği

Ham verinin kendi başına değeri düşüktür. Veriler, bilgisayar sistemleri aracılığıyla belli bir amaca yönelik olarak işlenmek suretiyle bilgiye dönüştürülmektedir. İşletmelerde bu amaca yönelik organize edilen bilgi sistemleri, satışlar, faturalar, ödemeler gibi işlemlerin takip edilmesini mümkün kılarken, karar almaya yönelik gereksinimlere de girdi sağlamaktadır. Bilginin olağanüstü boyutlarda artmasıyla beraber işletme içinde ve dışında bilgilerin yanı sıra önceden tahmin edilemeyen sorunlara yönelik çözüm üreten, ileriye dönük tahmin sistemlerine gereksinim duyulmuştur. Bu çerçevede veri madenciliği teknikleri işletme ve finans alanında da yaygın bir biçimde kullanılmaya başlanmıştır. Veri madenciliği, bilgi teknolojilerinin doğal gelişim sürecinin sonucudur. Büyük veriler, farklı alanlardaki büyük ölçekli veri tabanları içinde değerli verileri barındıran bir veri madeni gibidir. Veri madenciliği ise bu verilerden önceden bilinmeyen anlamlı ve kullanışlı bilgileri üretme süreci olarak tanımlanmaktadır. Veri madenciliği, bu bilgi üretme sürecini bilgisayarı, makine öğrenmesini, veri tabanı (deposu) yönetimini, matematiksel algoritmaları ve istatistiksel teknikleri kullanarak yapmaktadır (Albayrak ve Yılmaz, 2009, s. 32).

Veri madenciliği genel olarak; büyük miktarda veri içinden, gizli kalmış, değerli, kullanılabilir bilgilerin açığa çıkarılmasını sağlar. Veri madenciliği uygulamalarında amaç, toplanmış olan verilerin bazı istatistiksel yöntemlerle incelenerek ilgili kurumda ve yönetim destek sistemlerinde kullanılmak üzere değerlendirmeye tabi tutulmasıdır. Analizlerin yapılmasından sonra elde edilen verilerin bir istatistikçi veya ekonometrist gözüyle incelenerek daha önceden düşünülmemiş kavramların ortaya çıkartılması, başarılı bir veri madenciliği süreci olarak değerlendirilmektedir (Vahaplar, 2003, s. 33).

Veri madenciliğinde kullanılan modeller, "tahmin edici" ve "tanımlayıcı" olmak üzere iki ana başlıkta sınıflandırılmaktadır. Veri madenciliği modellerini işlevlerine göre "sınıflama ve regresyon modelleri", "kümeleme modelleri ve birliktelik kuralları" ve "ardışık zamanlı örüntüler" olmak üzere üç ana başlık altında da incelenmektedir. Sınıflama ve regresyon modelleri tahmin edici, kümeleme, birliktelik kuralları ve ardışık zamanlı örüntü modelleri tanımlayıcı modellerdir (Albayrak ve Yılmaz, 2009, s. 33).

Veri madenciliği genelde altı adımda gerçekleştirilmektedir (Daniel, 2005, s. 27):

- 1) Araştırma probleminin tanımlanması,
- 2) Verilerin tanınması aşaması,
- 3) Verilerin hazırlanması aşaması,
- 4) Modelleme aşaması,
- 5) Değerlendirme aşaması,
- 6) Uygulama aşaması.

### **1.3.7. Finansal teknoloji**

Kuruluşların büyüklüğü dikkate alındığı zaman, finans sektörü, bilhssa günümüzde FINTECH'ler, iş süreçlerinin dijitalleştirilmesi, otomize edilmesi ve yapay zekâ kullanılması ile alakalı yenilikçi teknolojilerin ana kullanıcısı pozisyonundadır (Mavlutova ve Volkova, 2019, s. 161).

Finansal Teknolojinin ilk heceleri birleştirilerek oluşturulan FİNTECH kavramıyla, yenilikçi finansal uygulamalar ve bu teknolojileri kullanan firmalar ifade edilmektedir. FINTECH'ler arasında, akıllı sözleşmeler, nesnelerin interneti, kriptoparalar ve blokzinciri, dijital ödeme sistemleri, robotların kullanıldığı hizmet alanları, fon fazlası olan kişilerle fon gereksinmesi olanlar arasında online olarak en uygun fiyattan fon seçme olanağı sağlayan P2P borçlanma, crowd finanslama, yapay zeka uygulamaları sayılmaktadır. Teknolojik gelişmelere paralel olarak Türkiye'de de yalnızca banka ve benzeri finansal kuruluşların tekelinde olmayan ve çağdaş, yenilikçi işletmelerin kullandığı bu tür geniş tabanlı uygulamaların tamamı FİNTECH içinde ele alınmaktadır (Bulazar ve Küçükçolak, 2021, s. 53).

FİNTECH şirketleri, dijital teknolojilerin ve büyük verilerin kullanımına dayalı finansal yenilikler üretmektedir. FİNTECH'ler hızlı, yenilikçi, müşteri odaklı, dijital teknolojileri kullanmaktadırlar. FİNTECH'in finansman alanındaki ivme sağlayan gücü, talep tarafından, karşılanmayan müşteri talebi ve tüketici kredileri iken arz tarafından veriye ulaşma, teknolojik avantajlı finansmana erişme, düzenleme eksikliği, rekabet eksikliği olarak ortaya çıkmaktadır. FİNTECH hizmetlerinin finansman alanındaki hızlıca büyümesi şüphesiz hem avantajlar hem dezavantajlar getiren ve finansal sistem için muhtemmel riskler taşıyan değişiklikleri birlikte getirmektedir. Diğer taraftan bu tür firmalar pazar gücünü daha yoğunlaştırabilmekte ve yeni sistematik risklere yol açabilmektedirler (Çetiner vd., 2021, s. 369).

### **1.4. Algoritmik Karar Verme Sistemleri**

Karar verme süreci, sürecin işleyiş mekanizmasına çıkarsamanın yapılaş biçimine bağlı olarak kullanılan verinin üzerinden işletilen çıkarsama adımları sayesinde bilgilerin ve gerçeklerin bir araya getirilmesi yoluyla bunların etkili, etkin ve üretken bir karara dayanak oluşturmasına yardımcı olmaktadır. Bir bilgisayarın zekice davranışlar sergileyebilmesi onun; zeki teknikler kullanmasına, çıkarsamalar yapabilmesine, tecrübelerden öğrenmesine ve anlayış geliştirmesine, bilgilerini ve deneyimlerini bulunduğu çevrenin yönelimine müdahil olmada kullanmasına bağlı olmaktadır. Diğer taraftan, insan davranışına benzer şekilde belirsizlik ve bulanıklık koşulları altında karar verebilen bilgisayar sistemleride bu tanımlamaya dâhil edilmektedir. Bazı durumlarda daha derin bilgilerine dayanarak çift anlamlı ya da zıtlıklar barındıran olaylarda böyle bir sistemin sofistike bir şekilde çalışarak değerlendirmede bulunması ve yargıya varması beklenir. Uzun yıllardır, bu niteliklere sahip

sistemler geliştirilmekte hem bilimsel amaçlarla hem de ticari amaçlarla kullanılmaktadırlar. Karar verme işlemini insanın uzmanlık alanına dayandıran, bu şekilde çıkarsama yaparak insana zor ve kompleks gerçek hayat problemlerinde yardımcı olabilen bilgisayar yazılımları ve sistemleri bulunmaktadır. Bununla birlikte, algoritmik modellere dayanan makine öğrenmesi teknikleri veya yapay zeka uygulamaları ve benzeri teknikleri kullanılarak sistemin çalıştığı çevre unsurlarıyla etkileşime girmesi, bu şekilde sistemin davranışlarının ve tepkilerinin değiştirilmesine yol açabilmesi sayesinde çevreye uyum sağlaması, bu şekilde çıkarsamanın esnek bir şekilde yapılabilmesi mümkün olmaktadır (Jantan vd., 2010, s. 33).

Algoritmik karar verme sistemleri, hem ticari kullanım ve pazar stratejileri özelinde, hem de kişisel verilerin korunması ve etkili kamu politikalarının oluşturulmasında önemli rol almaya başlayan faktörler olarak günlük hayatta da karşımıza çıkmaktadır (İstanbul Barosu, 2020, s. 5).

#### **1.4.1. Algoritmik karar verme sistemlerinin faydaları ve avantajları**

Algoritmalar farklı alanlarda faaliyet gösteren işletmelerin verimliliklerini arttırmak üzere karar alma mekanizmalarında da kullanılmaktadır. Bu çerçevede algoritmalar, belirlenmiş değişkenler ve kurulan matematiksel modeller kapsamında verimliliği ölçen karar destek mekanizmalarına dönüşmektedir. Bankacılık, sigortacılık, menkul kıymet portföy yönetimi gibi bireylerin gündelik hayatını doğrudan ilgilendiren pek çok farklı alanda; büyük verilerin değerlendirilmesi ve karar alma evrelerinde algoritmaların rol aldığı görülmektedir. Günden güne artan büyük verilerin yükü, bu veriden değer yaratılması ve veri destekli kararların kullanılması eğilimi arttıkça algoritma destekli karar destek ve tahmin sistemlerine duyulan güven artmaktadır. İnsanların karar verme süreçlerinin yapay sistemlere üstünlüğü insanların sorularına tartışarak cevap bulmaları ve kararlarını gerekçelendirmeleridir. Ancak algoritmik modeller, kısa sürede işleyebilecekleri verinin miktarı göz önüne alındığında insanlardan üstün gelmektedir (Burkart vd., 2020, s. 1).

Algoritmalar aracılığıyla alınan kararlar etkin, tarafsız ve güvenilir olarak kabul edilmektedir. Bu şekilde teknolojik yaklaşımlar algoritmaları, teknolojik, iktisadi ve toplumsal gelişmenin önemli bir faktörü olarak göstermekte ve bu teknolojileri kullanan veya geliştiren güç sistemlerini desteklemektedir (Mager, 2015, s. 1).

Algoritmaları benimsemek ve algoritmik düşünmek problem çözümünü kolaylaştıran bir metoloji sağlamaktadır. Günlük hayatta karşılaşılan problemlerin analiz edilerek çözüm aşamalarını basit bir biçimde belirlemek, problemin çözüm algoritmasını oluşturmak ve çözüme en kısa yoldan etkili bir biçimde ulaşılabilmesine imkân sağlamaktadır. Önceden geliştirilmiş çözüm algoritmalarını bilmek ve benimsemek daha önce karşılaşılmamış problemlerle karşılaşıldığında, çözüm üretebilmek için uyarlanabilir ya da yeni

algoritmaların oluşturulmasında yeni fikirler üretebilir. Bu çerçevede algoritmaları tanımak ve farkında olmak oldukça mühim kabul edilmektedir (Aytekin vd., 2018, s. 153).

#### **1.4.2. Algoritmik karar verme sistemlerinin riskleri ve dezavantajları**

Algoritmalar toplumsal eşitsizlikleri derinleştirmekle birlikte internet kullanıcılarını “yankı odaları”na da hapsetmektedirler. İnternet kullanıcıları, git gide artan bir biçimde kendilerine benzer görüşlere, beğenilere, ilgi alanlarına sahip haberlere, profillere, reklamlara maruz kaldıkları için sosyal farklılıklara dair farkındalıkları zayıflamaktadır. Kullanıcı türevli içeriğe dayalı çevrimiçi sosyal ağlar, kuramda toplumdaki farklı görüşleri içerebilecek iken platformların ticarileşmesi ve algoritmaların kullanıcıların dijital etkinliklerine göre içerikleri filtrelemesi neticesinde farklılıktan çok tekdüzelığe ve kutuplaşmaya neden olan ve insanların mevcut görüşlerini pekiştiren ve manipülasyona açık alanlar haline gelmektedirler. En yakın tarihli manipülasyon örneklerinden biri olarak, veri analizi şirketi Cambridge Analytica'nın Facebook kullanıcılarının kişisel verilerini hırsızlayarak hem 2016 Amerikan Seçimlerinde hem de İngiltere'deki Brexit Referandumunda sosyal medya kullanıcılarını profillemesi ve siyasi tercihlerini manipüle etmesi gösterilmektedir (Şener, 2021, s. 83).

## 2. BÖLÜM

### ALGORİTMİK TİCARET

#### 2.1. Geleneksel Menkul Kıymet Ticareti

Şirketler, faaliyetlerini yürütmek ve büyütme üzere sermayeye gereksinim duyarlar. Yatırımları finanse etmek için gereken finansmanı ya borç para alarak, sonra faiziyle geri ödeyerek ya da sermaye artırımına giderek yeni hisse senetlerinin sermaye piyasalarında satarak temin edebilirler. Şirketin bir parçası olarak hisse senetlerini satın alan yatırımcılar, kârın bir kısmını temettü şeklinde alırlar. Kıt kaynaklar olarak özsermaye ve borç, paranın zaman değeri açısından ek içsel değere sahiptir; fiyatları, şirketin performansına, mevcut piyasa koşullarına ve özellikle şirketin gelecekteki görünümüne, sektör, sermaye arz ve talebine ve bir bütün olarak ekonomi ile ilgili faktörlere bağlıdır. Örneğin, borçlanma sermayesi için uygulanan faiz oranları değişirse, getirileri daha yüksek/düşük getiri oranları sunan benzer ürünlerin/şirketlerin getirileriyle karşılaştırılacağından, bu mevcut borcun değerini etkilemektedir. Hisse senedi ve tahvillerin alımı ve satımı, gerçek değerleri hakkında farklı görüşlere sahip yatırımcılar arasında çeşitli şirket ve kurumların borç veya hisse senedi (ve diğer araç türleri) alıp satma eylemini ifade etmektedir. İkincil piyasa olarak borsalar aracılığıyla yapılan bu ikincil piyasa işlemleri aynı zamanda “fiyat keşfi” amacına da hizmet etmektedir. Alıcılar ve satıcılar, bir finansal aracın takası için bir fiyat üzerinde buluşurlar ve anlaşılırlar. Bu işlem kamuya açıklandığı zaman, diğer potansiyel alıcıları ve satıcıları menkul kıymetin en son piyasa değeri hakkında bilgilendirmiş olur. Son yirmi yıldaki ticaret sürecinin evrimi, inanılmaz bir yaratıcılık, şiddetli rekabet ve usta teknoloji hikâyesi oluşturmaktadır. Büyük ölçüde, bir şirketin gerçek değerini değerlendirmek için zaman içinde son derece karmaşık ve şaşırtıcı derecede verimli bir mekanizma yaratarak, kâr etmenin olumlu (ve bazen o kadar da olumlu olmayan) güçleri tarafından yönlendirilmeye devam etmektedir (Velu vd., 2020, s. 3).

#### 2.2. Algoritmik Ticaret

Algoritmik ticaret, finansal kurumlar, menkul kıymet piyasası düzenleyicileri ve farklı borsalar için önemli ölçüde daha büyük bir odak noktası haline gelmeye başlamıştır. Piyasadaki gelişmeler ve daha katı düzenlemeler, hisse senedi ticaretini daha karmaşık ve daha az karlı hale getirmiştir. Otomasyon ve yeni teknolojiler, son beş yılda ticaret oyununu önemli ölçüde değiştirmiştir. Finansal bilgilerin hızı, herkesin tahminde geride bırakmakta, finans mühendisliği yoluyla daha yüksek ağ oluşturma hızları, tüccarların ve piyasa katılımcılarının daha düşük komisyon talebini karşılama şeklini değiştirmekte ve algoritma (otomatik model) tabanlı ticaretin oluşturulmasını sağlamaktadır. Daha düşük işlem maliyetleri için artan rekabet, firmaları ticaret ve işleme altyapılarına önemli ölçüde yatırım yapmaya zorlamaktadır. Elektronik ve algoritmik ticaretin yaygınlaşması ile bir

komisyoncunun artık bazı elektronik yürütme yöntemlerini kullanmadan sipariş akışını gerçekleştirmesi neredeyse mümkün değildir. Borsada emir fişleri ile koşan geleneksel memurlar ve teklif fiyatlarını müzakere eden müzayededeki kişilerin yerini yakın zamanda, programları ve algoritmaları kullanarak emir akışını kolaylaştırmak için broker ekranlarına parametrelerini yazan tüccarların sesi alabilir. Geçmişte, teknolojiyi borsadaki ticaret sürecine uygulamak veya borsalar ve piyasa katılımcıları ile doğrudan etkileşim kurmak için fırsatlar sınırlıydı. Bu, programların tanıtımı, doğrudan pazar erişimi ve algoritmik ticaretle değişti. Otomatik ticaret akışı, insan denetimi olmadan bilgisayarlı ticaretin devralınması gibi çağrışımlar taşısa da, satın alma ve satma konusundaki gerçek kararlar bilgisayarlar tarafından değil, insanlar tarafından verilir. İnsanlar nihai ticaret kararlarını ve bunların uygulanmasının arkasındaki parametreleri verir, ancak bilgisayarlar sipariş akışını verimli bir şekilde yönlendiren algoritmaları hesaplayabilir ve çoğu durumda bilgisayarlar, program içindeki her bir hisse senedine yönelik işlemlerin dökümüne yardımcı olur (Kim, 2007, s. 1-2).

### **2.2.1. Tanımı**

Algoritmik ticaret, "menkul kıymetlere ve diğer sermaye piyasası araçlarına ilişkin belirli bir miktardaki alış veya satış emrini, algoritmanın parametreleri ve kısıtlamaları tarafından belirlenen hedeflere dayalı olarak emirlerin zamanlamasını ve emirlerin büyüklüğünü otomatik olarak oluşturan nicel bir modele yerleştirme" olarak tanımlanmaktadır (<http://www.towergroup.com>).

Nicel stratejiler, doğası gereği, ticaret fırsatları için piyasayı araştırmak için algoritmalar kullanır. Algoritmik ticaret, genellikle belirli bir dizi kurala göre birleştirilen bir dizi eşzamanlı işlemi içeren ticaret stratejilerini ifade etmektedir. Algoritmik ticaretin amacı, önceden ayarlanmış parametrelere dayalı olarak menkul kıymet siparişlerinin boyutunu ve zamanlamasını verimli bir şekilde kolaylaştırmaktır. Bu tür algoritmalara, iki karşılaştırılabilir menkul kıymetin yanlış fiyatlandırıldığı ancak temel benzerliklerine dayalı olarak aynı fiyat hedefinde birleşmesinin beklendiği bir çift alım satım algoritması örnek olarak verilebilir (Friedlander, 2005, ss. 6-8).

Algoritmik ticaret, en basit haliyle, önceden belirlenmiş bir dizi ticaret kuralı ve talimatını izleyerek bir finansal aracın alım satım işlemlerinin bilgisayar ortamında yürütülmesidir. Yatırımcılar, bir emri yürütme için bir aracı kuruma göndermek veya bir emri bir borsaya yönlendirmek yerine, emri yürütmek için algoritmaya girmeleri yeterlidir. Algoritmalar daha sonra, en iyi piyasa fiyatlarını elde etmek ve genel ticaret maliyetlerini azaltmak için gün içinde ve çeşitli ticaret merkezlerinde uygulanmak üzere daha büyük siparişleri daha küçük parçalara böler. Algoritmik ticaretin birincil amacı, yatırım kararının uygulanmasının fonun

yatırım hedefi ile tutarlı olmasını sağlamak ve emrin genel işlem maliyetlerini yönetmek ve uygun fiyatları elde etmektir (Kissell, 2021, s. 3).

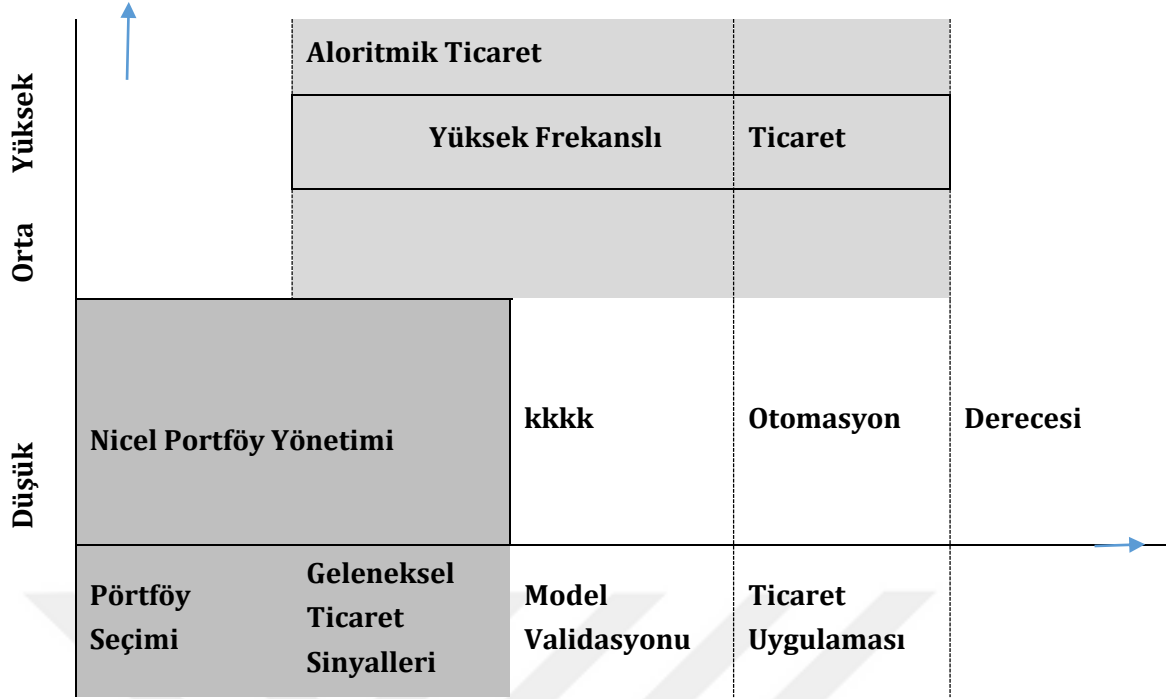
Son yirmi yılda teknolojinin gelişmesi, piyasaların işleyişini değiştirmiştir. Artık hisse senedi piyasaları, ticaret yapan bir değişim katındaki insanların egemenliğinde değildir. Bunun yerine, birçok firma elektronik verileri alan, analiz eden ve teklifler yayınlayan ve alım satımları başlatan bilgisayar algoritmalarını kullanır. Ticaret sürecini otomatikleştirmek için bilgisayar kullanan firmalara algoritmik yatırımcılar adı verilir (Brogaard, 2010, s. 1).

Algoritmik ticaret kavramı genellikle yüksek frekanslı ticaret, sistematik ticaret, elektronik ticaret ve düşük gecikmeli ticaret kavramlarıyla eş anlamlı olarak kullanılmaktadır. Bununla birlikte, terimler arasındaki farklılıkların ortaya konulması önemlidir. Bir ticaret sistemi genellikle oldukça metodolojik bir yaklaşımı ifade eder (Aldridge, 2013, s. 34). Bu kavramlar açıklanmıştır:

**a) Yüksek frekanslı ticaret (HFT: High-Frequency Trading)**, hisse senetlerinin bir bilgisayar tarafından hızla alınıp satıldığı bir yatırım stratejisi türüdür. Algoritması saniyeler veya milisaniyeler gibi çok kısa bir süre boyunca tutulur. HFT'ler, hisse senetlerini en hızlı şekilde devreden algoritmik tüccarların alt kümesidir (Brogaard, 2010, s. 1). Hızlı algoritmik yürütme kullanan tüm etkinlikleri içeren bir HFT tanımı ABD Emtia Vadeli İşlemler Ticaretinin Teknoloji Alt Komitesi tarafından yapılmıştır. Yüksek frekanslı ticaret, aşağıdakileri kullanan bir otomatik ticaret biçimidir (Aldridge, 2013, s. 13).

- İnsan yönlendirmesi olmaksızın her bir bireysel işlem için karar verme, sipariş başlatma, oluşturma, yönlendirme veya yürütme için algoritmalar;
- Yakınlık ve ortak konum hizmetleri de dâhil olmak üzere yanıt sürelerini en aza indirecek şekilde tasarlanmış düşük gecikmeli teknoloji;
- Emir girişi için piyasalara yüksek hızlı bağlantılar; ve
- Yüksek mesaj oranları (siparişler, teklifler veya iptaller).

TCMB'ye göre Algoritmik alım – satım işlemlerinin bir alt sınıfı olan HFT'ler, önceden programlanmış yatırım stratejileri doğrultusunda saniyenin milyonda biri hızlara ulaşabilen otomatik alım – satım işlemleri olarak tanımlanmaktadır. İleri teknolojiyle bağlantılı altyapılar ile çalışan HFT'ler, organize borsalar ve elektronik işlem platformları üzerinden, hisse senedi, döviz ve vadeli işlemler piyasalarında kullanılmaktadır. HFT işlemi, ticaret hacminin önemli bir bölümünü oluşturan menkul kıymetler ve türev piyasalarında yaygın hale gelmiştir. HFT'ler, organize borsalar ile elektronik işlem platformları üzerinden; döviz, hisse senedi ve vadeli işlemler piyasalarında kullanılmaktadır. HFT işlem payının ABD ve Avrupa'da sırasıyla %55 ve %40 olduğu düşünüldüğünde Türkiye'nin bu anlamda küresel benzerlerine yakınsaması sermaye likiditesini, yatırımcı hacmini ve piyasaları çok farklı ufuklara taşıyacaktır (Ersan ve Ekinci, 2016, ss. 233-234).



Şekil 2.1. Algoritmik Ticaret, Yüksek Frekanslı Ticaret ve Nicel Portföy Yönetimi

**Kaynak:** Gomber vd. 2011, s. 3

Şekil 2.1.'de algoritmik ticaret ile yüksek frekanslı ticaretin nicel portföy seçimi kararlarında kullanılmasının sınırları çizilmiştir.

**b) Sistematik ticaret,** bazı katı çerçeveleri takip etmekle birlikte tamamen otomatik olması gerekmemektedir. Bir yatırımcı, belirli göstergeler belirli bir model oluşturduğunda işlemleri manuel olarak yerleştirirse sistematik ticaret yapmış olarak kabul edilebilir. Sistematik terimi, sezgilerini veya takdirlerini kullanarak sistematik tahsisler uygulayan yatırımcıları diğer yatırımcılardan ayırmak için gündeme gelmiştir. Tüm yüksek frekanslı ve algoritmik yatırımcılar da sistemattiktir (Aldridge, 2013, s. 34).

**c) Elektronik ticaret,** yatırımcının emirleri, elektronik olarak mı yoksa telefonla mı vermeyi seçeceği işlem tercihlerini tanımlar. Yüksek frekanslı ticaret, algoritmik ticaret ve düşük gecikmeli ticaret zorunlu olarak elektrondur, ancak sistematik ticaret elektronik olmayan bileşenleri içerebilir. Ancak bunun tersi olmak zorunda değildir; birçok elektronik ticaret sistemi, yalnızca algoritmik olarak verilir ve verilmeyen siparişleri yönlendirir. Bununla birlikte, çoğu piyasa ve yatırımcı elektronik platformlara geçtiğinden, elektronik ticaret kavramı örtük ve geçersiz hale gelmektedir ve kullanımı azalmıştır (Aldridge, 2013, s. 34).

**d) Düşük gecikmeli ticaret,** yatırımcılar ve borsalar arasında hızlı bağlantı kullanan ticaret anlamına gelmektedir. Gecikme, yatırımcı ve borsa arasındaki zaman mesafesini ölçer. Çoğu gecikme ölçümü günümüzde mikrosaniye cinsinden kaydedilmektedir. Yüksek frekanslı ticaret sistemleri de genellikle düşük gecikmeli olur, ancak bunun tersi geçerli olmak zorunda

değildir. Düşük gecikmeli sistemler genellikle menkul kıymet ticaretlerinde daha iyi fiyatlar elde etmek için düşük frekanslı yatırımcılar tarafından kullanılır (Aldridge, 2013, s. 34).

Algoritmik ticaret, elektronik ticaretten daha karmaşıktır ve yüksek frekanslı portföy tahsis kararlarıyla birlikte emir yürütme süreçlerini kapsayan çeşitli algoritmaları kullanır. Yürütme algoritmaları, alım satım kararları başka bir yerde verildikten sonra alım satım uygulamasını optimize etmek için tasarlanmıştır. Algoritmik yürütme, emri borsaya yönlendirmenin en iyi yolu, emrin hemen yürütülmesi gerekmiyorsa, gönderilen bir emri yürütmek için en iyi zaman noktası ve emrin en iyi şekilde olması gereken en iyi boyut dizisi hakkında kararlar verir. HFT sinyalleri üreten algoritmalar, belirli bir menkul kıymette bir pozisyona girmek veya kapatmak için portföy tahsis kararları ve kararlar verir. Örneğin, algoritmik yürütme, 1 milyon IBM hissesini satın almak için alınan bir siparişin, fiyatta ani bir yükselişi önlemek için en iyi 100 hisselik artışlarla ele alınacağını belirleyebilir. Ancak yürütme algoritmasına verilen karar yüksek frekanslı olabilir veya olmayabilir. HFT sinyalleri oluşturmak için devreye alınan bir algoritma, IBM'in 1 milyon hissesini satın alma kararını doğuracaktır. Yüksek frekanslı sinyaller daha sonra siparişin optimal zamanlamasını ve yönlendirmesini belirleyecek olan yürütme algoritmasına iletilecektir (Aldridge, 2013, s. 11).

Algoritmik ticaretin yükselişi, doğrudan insan etkisindeki bir düşüşün finansal piyasalarda kendini gösterdiğini ifade etmektedir. Bu nedenle, ticaret algoritmalarının, kurallarında belirtilmedikçe, ticaret algoritmalarının önceden tanımlanmış kurallar dizisinden asla sapmadığı varsayıldığından, ticaret davranışında insan yatırımcılardan farklı olduğu düşünülebilir. Başka bir deyişle, bir ticaret algoritması her zaman programlanmış sınırları içinde hareket eder, ancak kendisine iletilen tüm bilgileri hesaba katar. Diğer taraftan, insan yatırımcıların sezgilerine ve çevrelerinde olup bitenlere göre hareket etme olasılıkları daha yüksektir ve belirli bilgilere diğerlerinden daha fazla değer verme eğilimi vardır (Verheggen, 2017, s. 8).

Algoritmalar, hem aracılık hem de tescilli olarak menkul kıymet ticareti yapmak için kullanılır. Kurumsal yatırımcılar, zaman içinde kademeli olarak büyük miktarlarda ticaret yapmak için algoritmik ticareti kullanırlar. Bu şekilde piyasa etkisini ve ticaret maliyetlerini en aza indirebilirler. Genellikle aracılık için kullanılan tescilli algoritmalara yüksek frekanslı tüccarlar (HFT) denir. HFT'ler, bir menkul kıymetin fiyatının etkin fiyattan ne zaman saptığını belirlemek ve sapmalara karşı ticaret yapmak için akışta bulunan bilgileri hızlı bir şekilde işlemek için algoritmalar kullanır. Algoritmik ticareti incelemek, finansal piyasalardaki teknolojik ilerlemelerin önemi ve bunların yatırımcıların ve tüccarların karşılaştığı katılım ve izlemedeki sürtüşmeleri nasıl etkilediğini genel olarak anlamamızı kolaylaştırır (Hendershott and Riordan, 2013, s. 1002).

Algoritmik ticaret yeni bir fenomen değil, ancak yoğunluğu, hacmi ve özellikle gerçekleştirilme hızı son on yılda hızlı gelişen yeni bir trenddir. Algoritmik ticaret stratejilerini uygulamak için verimli bir platformun mevcudiyeti, algoritmik ticaretin haber

akışındaki ve piyasa koşullarındaki değişikliklere milisaniyeler içinde yanıt verebilen bir alt kümesi olarak görülebilecek yüksek frekanslı ticaret (HFT) için bir ön koşuldur (Boehmer vd., 2015, s. 5).

Satış tarafında aracılık yapan aracı kurumlar, başlangıçta, firmalarının tescilli hesapları adına işlemleri yürütmek için algoritmik programlar geliştirmişlerdir. Başlangıçta şirket içinde tasarlanmış, ancak dış satıcılar, müşteri ticareti için doğrudan pazar erişimi/sipariş yönetim sistemleri sağlamak ve merkezi bir sipariş işleme ve takas sistemi sağlamaktadırlar. Satış tarafındaki oyuncular, işlem maliyetlerini daha da düşürürken daha verimli sipariş akışı sunmak için algoritmalarını emsallerinden daha rekabetçi olacak şekilde sürekli olarak yenileyerek özelleştirirler. Ayrıca, şirket içi algoritmalarını müşterilere ve daha küçük firmalara sunmaktadırlar. Satış tarafındaki aracı kurumlar tarafından müşterilere sunulan algoritmik stratejiler, müşterilerin kendi stilize versiyonlarını oluşturabilme yeteneğine sahip olmaları ile genellikle özelleştirilmiştir. Özelleştirilmiş algoritma seçeneklerindeki artışlar, portföy yöneticilerinin ticaret tarzlarına daha iyi hizmet edebilme potansiyeline sahiptir (Kim, 2007, s. 4).

## **2.2.2. Avantajları ve dezavantajları**

Algoritmik ticaretin çok sayıda avantajı olmakla birlikte bazı dezavantajları da bulunmaktadır.

### **2.2.2.1. Avantajları**

Algoritmik ticaret, bir stratejiyi “önceden düşünme” yeteneğiyle birlikte çok sayıda avantaj sunar. İnsan psikolojisine sahip yatırımcılar, genç yetenekli tüccarlarda “tükenmenin” ana kaynaklarından biri olan ticarete gerçek zamanlı duygusal katılımdan kurtulur. Ayrıca daha pahalı niteliklere ve eğitime sahip farklı bir çalışanın gerekliliği ile dengelenebilecek bir insan gücü tasarrufu sağlamaktadır (Leshik ve Cralle, 2011, s. 4).

Algoritmalar, karmaşık matematiği gerçek zamanlı olarak yürütebilir ve insan müdahalesi olmadan tanımlanan stratejiye dayalı olarak gerekli kararları alabilir ve ticaret işlemini bilgisayardan otomatik olarak Exchange'e gönderebilir. Artık insan “bant genişliği” ile sınırlı değildir. Koşullu kural katmanlarına sahip gelişmiş algoritmalar kullanarak aynı anda yüzlerce sayıda emiriletebilir ve işlem gerçekleştirilebilir. Bu yetenek, tek başına maliyet tasarrufu nedeniyle algoritmik ticaretin büyümesine güç sağlamak için yeterli olacaktır (Leshik ve Cralle, 2011, s. 4).

Artık neredeyse her büyük aracı kurum, kurumsal müşterilerine, siparişleri tek bir hisse senedi, hisse senedi çiftleri veya hisse senedi sepetleri halinde yürütmelerine yardımcı olmak için bir dizi algoritma sunmaktadır. Algoritmalar tipik olarak siparişlerin zamanlamasını,

fiyatını, miktarını ve yönlendirmesini belirler, farklı menkul kıymetler ve ticaret yerleri arasındaki piyasa koşullarını dinamik olarak izler, büyük siparişleri optimal olarak ve bazen rastgele daha küçük parçalara bölerek piyasa etkisini azaltır ve hacim ve fiyat aralığı boyunca ağırlıklı ortalama fiyat gibi kıyaslamaları yakından takip eder. İstenen bir pozisyonu takip ederken, bu algoritmalar genellikle hem limit emirleri hem de pazarlanabilir emirleri kullanarak aktif ve pasif stratejilerin bir karışımını kullanır. Bu nedenle bazen likidite talep eden, bazen de likidite sağlayan kişiler olarak işlev görürler (Hendershott, 2011, s. 2).

Algoritmik ticaret, daha küçük spreadler ve pazar parçalanması problemlerine bir çözüm haline gelmiştir. Algoritmik programlar, büyük pay blokları olan ana siparişleri verimli bir şekilde dilimleme yeteneğine sahip olup; her küçük siparişin ya da alt siparişin en iyi fiyatı almasını sağlar. Yeni ağ bilgi hizmeti oyuncularının ortaya çıkışı algoritmik piyasa ve piyasa yapıcılar arasında çeşitlilik oluşturmuş, ancak daha büyük menkul kıymet brokerları ve aracıları için ciddi bir tehdit oluşturmuştur. Ağ bilgi hizmeti oyuncuları her zaman olacaktır, ancak daha iyi olanların ittifaklar oluşturmaları ya da daha büyük katılımcılar tarafından kazanılması, rekabetçi olmayan piyasa yapıcılarının piyasadan dışlanmasına neden olabilecektir. Algoritmik ticaret, yatırımcıların yerini alamayacak; ancak onu tasarlayan ve kullanan yatırımcılar kadar etkili olabilecektir. Diğer yandan, algoritmik programları daha etkili kullanmayı öğrenen yatırımcılar, teorik olarak, teknolojik olarak gelişmiş yeni kaynakları kendi avantajlarına nasıl kullanacaklarını anlamayan çok sayıda yatırımcıdan öne çıkacaktır. Şu anda yatırımcılar için çok sayıda menkul kıymet alışverişi yöntemi alternatifi bulunmaktadır. Bu yöntemlerin bazıları daha fazla insan müdahalesi ve karmaşıklık gerektirirken; algoritmik olanları gibi diğerleri ise otomatikleştirilebilir ve daha az karmaşık olabilir (Kim, 2007, ss. 3-4).

Finansal kurumlar, algoritmik ticareti kullanarak büyük piyasa emir akışını daha küçük parçalara bölmek için teknolojiyi kullanmaktadırlar. Algoritmik ticaret süreci, bireysel ve kurumsal yatırımcılara ve aracı kurumlara, piyasayı hareket ettirmeden büyük siparişleri tamamlama yeteneği sağlamaktadır. Büyük siparişlerin daha küçük siparişlere bölünmesi, alım satım maliyetini en aza indirmek için fazla likidite alır ya da piyasada yetersiz likidite yaratır (Kim, 2007, ss. 10-11).

Yatırımcıların finansal piyasalara katılımı ve takibi ile ilgili sürtüşmeler alım satım ve varlık fiyat dinamikleri için önemlidir. Kusurlu izleme, yatırımcıların tüm karşı taraflarla hemen iletişime geçmesini engeller. Bu, aramayı uzatır ve yatırımcıların daha hızlı ticaret yapmak için daha büyük fiyat tavizleri sunmasına neden olarak likiditeyi azaltır. Arama sürecindeki belirsizlik likidite riskini artırmaktadır. Likiditenin hem düzeyi hem de belirsizliği fiyatları baskılamakta ve sermayenin yanlış tahsis edilmesine yol açmaktadır. Algoritmik ticaret şeklindeki teknolojik ilerleme, likidite için piyasada verimliliği artırabilen ve ticaretten elde edilen kazançları kolaylaştırabilen izleme sürtüşmelerini azaltır (Hendershott ve Riordan, 2013, s. 1001).

Algoritmik ticarete, faaliyet ve hız açısından alım satımların ve siparişlerin yayılmanın boyutuna bağımlılığı, sipariş verme stratejilerinin rastgele olmadığını, daha ziyade verimli bir talep ve arz stratejisinin bir parçası olduğunu göstermektedir. Daha iyi izleme, tüccarların piyasa koşullarındaki değişikliklere hızla tepki vermelerini sağlayarak, algoritmik ticaretin spreadler geniş olduğunda daha fazla likidite sağlamasına ve spreadler dar olduğunda daha fazla likidite talep etmesine yol açar. Bu, likidite sağlanmasındaki belirsizliği azaltarak likidite riskini azaltır (Hendershott and Riordan, 2013, s. 1003).

Sezgilere aykırı gerçekleşen spread sonucunu anlamaya yardımcı olmak için, geleneksel ve algoritmik ticaret için maliyet yapılarının nasıl farklılaştığını düşünmek önemlidir. İzleme maliyetleri, insanların piyasayı her zaman izleyememesi için yeterince yüksekse, insanlar tarafından verilen limit emirleri her zaman tüm kamu bilgilerini yansıtmaz ve eskiyebilir. Algoritmaların sabit geliştirme maliyetleri yüksek olsa da, ana avantajlarından biri, genel bilgileri izlemede ve siparişlerini veya fiyat tekliflerini ayarlama neredeyse hiçbir marjinal maliyetin olmamasıdır. Bu nedenle, algoritmik aktivitedeki bir artış, verimli fiyatta ticaret yerine teklif güncellemesi yoluyla daha fazla değişikliğin ortaya çıkmasına neden olur (Hendershott, 2011, s. 30).

Algoritmik ticaret, yatırımcılara aşağıdakiler gibi birçok avantaj sağlar (Kissel, 2021, s. 25):

**Düşük Komisyonlar:** Algoritmik ticaret, yatırımcılara yalnızca yürütme ve yürütme ile ilgili hizmetler (risk yönetimi ve sipariş yönetimi gibi) sağladığından komisyonlar genellikle geleneksel komisyon ücretlerinden daha düşüktür. Algoritmik komisyonlar, bazı fonlar araştırma erişimi için daha yüksek bir oran ödese de, genellikle komisyonculara araştırma faaliyetleri için tazminat ödemez (Kissel, 2021, s. 25). Algoritmaların büyümesi için önemli bir faktör, yatırım yöneticileriyle bir ilişki sürdürmek için satış tarafı ücretlerinin ve komisyonlarının azaltılması olmuştur. Satın alan taraf, tipik olarak, araştırmaya erişim ve bir ilişkiyi sürdürme karşılığında belirli bir satış tarafı firmasına işlerinin bir yüzdesini verecektir. Broker-satıcılar tarafından alınan komisyonların azaltılması, algoritmik penetrasyonu daha da artıracak ve onları daha karmaşık hale getirecektir. Bu, satın alan tarafın ödediği komisyonlarla sübvansede edilen araştırma karşılığında belirli bir aracı kurumla az verimli bir fiyatla iş yapmak yerine, alım satımları en verimli ve rekabetçi fiyatlı mekân aracılığıyla gerçekleştirerek elde edilir. İşlemleri ölçmenin verimliliği, mevcut veri setine bağlıdır (Kim, 2007, s. 51).

**Anonimlik:** Emirler sisteme girilir ve bilgisayar tarafından tüm yürütme yerlerinde otomatik olarak işlem görür. Alıcı taraf yatırımcısı, siparişi kendi firmasından yönetir ya da siparişin satış tarafı satış araçları tarafından yönetilmesini ister. Emirler, klasik emirlerin tabi olduğu kısıtlamalara tabi değildir.

**Kontrol:** Alım tarafı yatırımcıları, siparişler üzerinde tam kontrole sahiptir. Yatırımcılar, fonun yatırım amacına ve fiili piyasa koşullarına göre alım satım yerleri

(görüntülenen/karanlık), piyasa/limit fiyatları, pay miktarları, bekleme ve yenileme süreleri gibi emir verme kurallarını ve ayrıca alım satımı ne zaman hızlandıracak veya yavaşlatacaklarını belirler. Yatırımcılar emri iptal edebilir ya da işlem talimatlarını neredeyse anında değiştirebilir.

**Minimum Bilgi Sızıntısı:** Broker, yatırımcının emri veya alım satım niyetleri hakkında herhangi bir bilgi almadığı için bilgi sızıntısı en aza indirilir. Satın alan yatırımcı, sadece algoritma seçimi ve algoritmik parametrelerin özellikleri ile alım satım talimatlarını ve yatırım ihtiyaçlarını belirleyebilir.

**Şeffaflık:** Yatırımcılara, emrin nasıl uygulanacağı konusunda daha fazla şeffaflık sağlanır. Her algoritmanın altında yatan yürütme kuralları yatırımcılara önceden bildirildiğinden, algoritmalar tam olarak ne yapmaya programlandıklarını yapacağından, yatırımcılar algoritmanın pazardaki hisseleri nasıl yürüteceğini tam olarak bileceklerdir.

**Erişim:** Algoritmalar, farklı pazarlara ve karanlık havuza hızlı ve verimli erişim sağlayabilir. Ayrıca, yatırımcıya yüksek hızlı bağlantıların faydalarını sağlayan ortak konumlu, düşük gecikmeli bağlantılar sağlarlar.

Algoritmaların düşük karar verme gecikmesi ile birlikte kısa gidiş-dönüş süreleri, tam otomatik ticaret sistemleri için ideal ticaret koşulları sunarak, saniyenin altındaki aralıktaki yeni koşullara yanıt vermelerini sağlar (Prix vd., 2007, s. 718).

**Rekabet:** Algoritmik ticaretin evrimi, bağımsız satıcılar, sipariş yönetimi ve yürütme yönetimi yazılım firmaları, borsalar, üçüncü taraf sağlayıcılar ve şirket içi geliştirme ekipleri gibi çeşitli piyasa katılımcılarına ek olarak geleneksel satış tarafı komisyoncu-satıcıları arasında rekabet ortaya çıkarmıştır. Yatırımcılar, bu artan rekabetin faydalarını daha iyi uygulama hizmetleri ve daha düşük maliyetler şeklinde görmüşlerdir. Sağlayıcılar arasında seçim yapma ve geçiş yapma kolaylığı ve esnekliği göz önüne alındığında, yatırımcılar herhangi bir seçime bağlı kalmazlar. Buna karşılık, algoritmik sağlayıcıların tekliflerini ve verimliliklerini sürekli olarak iyileştirmede daha proaktif olmaları gerekmektedir.

**Azaltılmış İşlem Maliyetleri:** Bilgisayarlar, değişen piyasa koşullarına ve planlanmamış olaylara tepki vermek için daha donanımlı ve daha hızlıdır. Yatırım kararı ile alım satım talimatları arasında tutarlılığı sağlama konusunda daha yeteneklidirler, bu da daha düşük piyasa etkisi maliyeti, daha az zamanlama riski ve daha yüksek tamamlanan sipariş yüzdesi (düşük fırsat maliyeti) ile sonuçlanır (Kissel, 2021, s. 25). Algoritmalar, karmaşık matematiği gerçek zamanlı olarak yürütebilir ve insan müdahalesi olmadan tanımlanan stratejiye dayalı olarak gerekli kararları alabilir ve ticaret işlemi bilgisayardan otomatik olarak Exchange'e gönderebilir. Artık insan "bant genişliği" ile sınırlı değildir. Koşullu kural katmanlarına sahip gelişmiş algoritmalar kullanarak aynı anda yüzlerce sayıda emiriletebilir ve işlem gerçekleştirilebilir. Bu yetenek, tek başına maliyet tasarrufu nedeniyle algoritmik ticaretin büyümesine güç sağlamak için yeterli olacaktır (Leshik ve Cralle, 2011, s. 4).

#### 2.2.2.2. Dezavantajları

Her alternatif yöntemin doğasına bağlı olarak dezavantajları vardır. Algoritmik ticaret günümüzde hisse senedi piyasalarına odaklanmaktadır, ancak küçük sermayeli hisse senedi ticareti gibi alanlarda henüz kullanılmamıştır. Gelecekte, algoritmik ticaret sabit gelir, vadeli işlemler, opsiyonlar ve döviz piyasalarını da içerebilir. Algoritmaların küçük sermayeli hisse senedi alım satımı ve birçok sabit gelir getiren menkul kıymetler gibi likit olmayan menkul kıymet alım satımında etkili bir biçimde kullanılıp kullanılmayacağı test edilmeye devam etmektedir. Geleneksel olarak ilişkilendirilen algoritmalar belirli bir varlık sınıfıyla, hisse senetleriyle, elektronik ticarete doğru hızlıca gelişen diğer pazarlara doğru çeşitlenmektedir. Türev araçlar gibi diğer varlık sınıflarının ticaretini yapmaya başlayanlar, başlangıçta teknoloji konusunda rahat ve bilgili olma eğilimindedir, bu nedenle bu sınıflardan bazılarının daha sistematik bir algoritmik yaklaşıma geçmek çok radikal görünmeyebilir. Algoritmik ticaret yakında vadeli işlemlerde, opsiyon ve döviz piyasalarında yerini alacaktır. Sabit getirili enstrümanların algoritmik ticarete geçen ya da sipariş akışını kolaylaştırmak için elektronik iletişim ağlarına dayanan son varlık sınıfı olması olasıdır. Bununla beraber, teknolojik olarak gelişmiş bir strateji olan algoritmik ticaret, küçük miktarlarda ya da devlet tahvilleri gibi sabit getirili çok likit piyasalara sunulmaktadır (Kim, 2007, ss. 3-4).

Algoritmik ticaret menkul kıymet ticaretindeki ihtiyaçların hepsini karşılama potansiyeline sahip olmakla birlikte bazı eksiklikleri ve sınırlamaları da mevcuttur. Bu dezavantajlarında bazıları aşağıdaki gibidir (Kissel, 2021, s. 27):

- Kullanıcılar, algoritmaya aşına oldukları için, siparişin özel gereklilikleri ve piyasa koşulları ne olursa olsun, kayıtsız hale gelebilir ve aynı algoritmaları kullanmaya devam edebilirler, bir yatırımcı körlüğü oluşabilir.
- Kullanıcıların, algoritmaları doğru kullandıklarından ve algoritmaların reklamı yapılan şeyi yaptığından emin olmak için algoritmaları sürekli olarak test etmesi ve değerlendirmesi gerekir. Kullanıcıların, piyasa ortamının türüne göre hangi algoritmaların en uygun olduğunu anlamak için araçlar, algoritmalar ve piyasa koşulları genelinde performansı ölçmesi ve izlemesi gerekir.
- Algoritmalar tam olarak belirtildiği gibi çalışır, bu da ticaret ortamı beklendiği gibi olduğunda güzeldir. Bununla birlikte, planlanmamış olayların meydana gelmesi durumunda, algoritma o özel durum için uygun şekilde eğitilmeyebilir ya da programlanmayabilir, bu da ortalamanın altında performansa ve daha yüksek maliyetlere neden olabilir.
- Kullanıcıların algoritma ve yatırım ihtiyaçları genelinde tutarlılığı sağlaması gerekir. Gerçek algoritmik ticaret kuralının olabildiğince şeffaf olmadığı ya da algoritmalara ne yapmaya çalıştıkları hakkında herhangi bir fikir vermeyen tanımlayıcı olmayan isimler verildiğinde tutarlılığı sağlamak git gider zorlaşmaktadır.

- Çok fazla algoritma mevcuttur. Hacim ağırlıklı ortalama fiyat olan VWAP, açıklayıcı bir algoritmik ad örneğidir ve araçlar arasında tutarlıdır. Ancak "Tarzan" gibi bir algoritma açıklayıcı değildir ve gün içinde nasıl işlem yapacağına dair fikir vermez. Yatırımcıların yüzlerce algoritma ve bu kod tabanlarında meydana gelen değişiklikleri takip etmesi gerekir. Örneğin, büyük bir aracı kurumlar 5-10 arasında farklı algoritmaya sahiptir.
- Algoritmaların büyümesi ve geleneksel uzmanların ve pazar belirleyici rollerinin azalması, açılışta daha zor bir fiyat keşif sürecine yol açmaktadır. Algoritmalar, uygun dilimleme stratejisini belirlemek için fiyat bilgilerini dâhil etme konusunda iyi bilgili olsalar da, bir menkul kıymet için adil piyasa fiyatını hızlı bir şekilde belirleme konusunda henüz çok bilgili değildir.

### 2.3. Algoritmalarda Kullanılmakta Olan Ortak Prosedürler

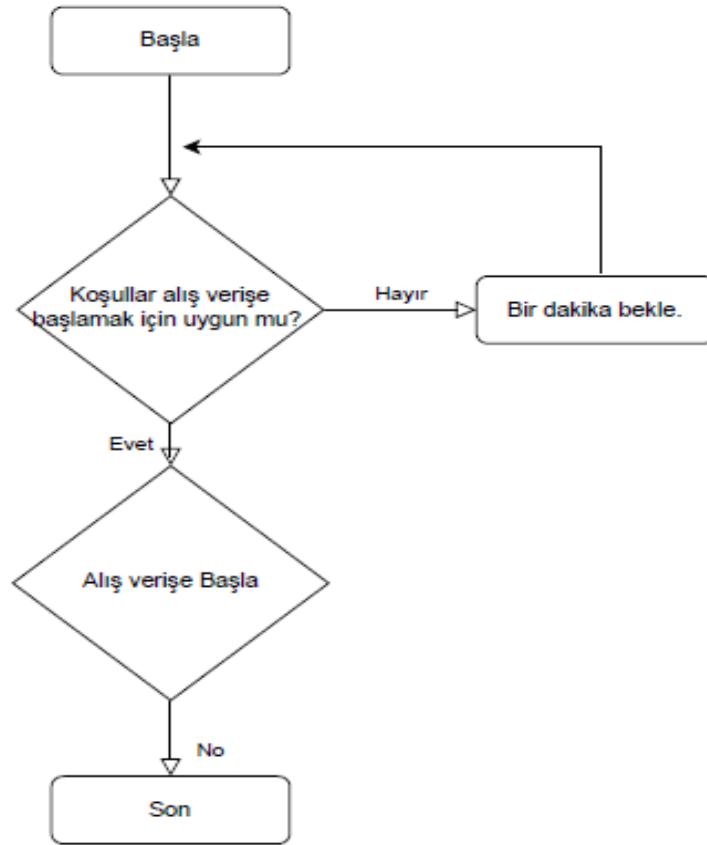
Kodlanmış algoritmaların ortak olarak kullandığı iki prosedür bulunmaktadır:

**a) Tamir fonksiyonu:** Algoritmalar tarafından üretilen çözümler talep edilen kısıtları her zaman karşılayamayabilmektedir. Başka bir ifadeyle toplam yatırım yapılan hisse senedi sayısı ( $K$ ) (Portföyde bulunmasına izin verilen toplam varlık sayısı) sayısına eşit olmayabilir veya bir hisseye ( $\epsilon$ ) değerinden daha az ağırlık verilmiş olabilir. Bu tür çözümler uygun olmayan çözümlerdir ve tamir prosedürü aracılığıyla uygun bir çözüme dönüştürülmeleri gerekir. Tamir prosedürünün ilk aşaması portföydeki toplam yatırım yapılan hisse senedi sayısını  $K'$ 'ya eşitlemektir. Eğer portföydeki yatırım yapılan hisse senedi sayısı  $K$  değerinden büyük ise,  $K'$ 'ya eşitlenene kadar en küçük ağırlığa sahip hisse senedinden başlanarak hisselerin ağırlıklarını ve o varlık için oluşturulan ikili değişkeni sıfır yapar. Eğer portföydeki yatırım yapılan hisse senedi sayısı  $K$  değerinden küçük ise,  $K'$ 'ya eşit olana kadar veri setinde yatırım yapılmayan rastgele bir hisse senedinin ikili değişkenini bir yapar. Bu şekilde, portföyde bulunması gereken toplam hisse senedi sayısı kısıtı sağlanmış olur. İkinci aşamada ise, yatırım yapılan hisse senetlerinin minimum ağırlıkları ( $\epsilon$ ) değerine eşitlenmektedir. İlk olarak, portföye dâhil edilen ( $K$ ) adet yatırım yapılan hisse senetlerinin ağırlıkları toplanır ( $CS$ ). Daha sonra portföyde bir hisse senedinin alabileceği maksimum ağırlık ( $FP$ ) hesaplanır. Daha sonra her hisse senedinin ağırlığı, o hisse senedinin ağırlığı ile  $FP/CS$  oranının çarpımının ve ( $\epsilon$ ) değerinin toplamına eşitlenir. Bu iki aşama sonucunda ilk başta uygun olmayan bir çözüm hem toplam yatırım yapılan hisse senedi sayısı kısıtını hem de minimum ağırlık kısıtını sağlayarak uygun bir çözüm haline dönüşmüş olur (Akyer, 2016, s. 39).

**b) Uygunluk hesaplama prosedürü:** Bu prosedür yardımıyla bir çözümün toplam getirisi, riski ve uygunluk değeri hesap edilir. İlk olarak portföydeki her hisse senedi ile o hisse senedinin ortalama getirisi çarpımları toplamı ile portföyün ortalama getirisi hesap edilir. Bundan sonra varyans-kovaryans matrisi kullanılarak portföyün riski hesap edilir. Varyans-kovaryans matrisi, veri setindeki hisse senetlerinin birbirleriyle ilişkisini gösteren matristir.

Bunun için yatırım yapılan her hisse senedinin ağırlığı teker teker portföydeki diğer hisse senetlerinin ağırlıkları ve o iki hisse senedinin varyans-kovaryans değeri ile çarpılır. Bu çarpımların toplamı toplam riski vermektedir. Nihayetinde uygunluk değeri hesaplanır. Uygunluk değeri hesaplanırken bir ağırlıklandırma kullanılır. Bu şekilde iki amaç fonksiyonu (maksimum getiri, minimum risk) tek bir amaç fonksiyonu olarak tanımlanır. Ağırlık değeri lambda ( $\lambda$ ) 0'dan başlayarak 1'e kadar istenilen aralıklarla ilerler.  $\lambda=0$  olduğunda portföy maksimumum getiriyi amaçlar,  $\lambda=1$  olduğunda ise amaç minimum risk olmaktadır (Akyer, 2016, s. 40).

Algoritmalar genelde basit olarak tasarlanır ve aşağıdaki Şekil 2.2.'deki gibi kurgulanır:



Şekil 2.2. Basit Algoritmik Ticaret Algoritması

**Kaynak:** Aldridge, 2013, s. 34

Bir algoritma tasarlandıktan sonra, bileşenlere bölünür ve bilgisayarların anlayacağı bir dilde kodlanır. Kodlamanın amacı, algoritmanın mantığını bilgisayarın "konuşmasına" doğru bir şekilde çevirmek ve bu süreçte, bilgisayarın kodu okuyup yorumlayacağı "çalışma süresi" sırasında mümkün olduğunca az gecikme olmasını temin etmektir. Doğrudan FPGA

yongalarına yazılan kod şu anda en hızlısıdır. Yine de, birçok yüksek frekanslı yatırımcı, standart FPGA olmayan mimariyi kullanır ve sistemlerini kodlamak için C++ ve Java gibi dillere güvenir. C++, insanlar tarafından kolayca anlaşılan en hızlı bilgisayar dili olmaya devam ederken, birçok sistem, en yavaş bileşenlerinin geçici çözümleriyle Java'da kodlanmıştır. Bu nedenle, örneğin, ünlü Nasdaq OMX sisteminin Java'da Java çöp toplama devre dışı bırakılarak kodlandığı ve daha yüksek hız için C++ benzeri doğrudan bellek erişimi ile değiştirildiği bildiriliyor (Aldridge, 2013, s. 34).

Bir algoritmanın gerçek ticaret mantığını özetleyen kod oldukça kısa olma eğilimindedir. Birçok başarılı durumda, ticaret mantığı 50 satır kadar az kod içerir. Bununla birlikte, gerçek satın alma ve satma kararlarına ek olarak, her HFT sistemi, 10.000'den fazla kod satırına kadar numaralandırılabilen destekleyici fiyat teklifi verisi alma işlevselliğinin yanı sıra, aynı zamanda 5.000 satır kod olarak çok fazla yer alabilecek ticari gönderme ve alındı makbuzu uygulamalarını içerir. Her HFT sisteminin belki de en uzun, ancak zorunlu bileşeni, toplam 50.000'den fazla kod satırı olabilen risk yönetimi kontrolleri ve dengeleridir (Aldridge, 2013, s. 35).

Algoritmalar; fiyatlara, haberlere, formasyonlara ve indikatörlere göre yazılabilmektedirler. HFT ve Ultra HFT (yanlış fiyatlama kaybolmadan veya piyasa sinyali fiyata dâhil edilmeden önce hızlı bir şekilde işlem yapmaktadır), işlemciler için, yalnızca küçük bir süreyle var olabilecek son derece küçük fiyat uyumsuzluklarından faydalanmak için bir yol oluşturup bilgisayar destekli kural tabanlı algoritmik işlem, emir vermek için otomatik kararlar veren programlar kullanır. Algoritmik ticaret, piyasa hareketlerinin sürekli izlenmesi gerekliliğini ortadan kaldırmış olup otomatik AL/SAT sistemiyle piyasa işlemlerini gerçekleştirmektedir. Algoritmik ticaret büyük boyutlu emirleri böler, bu ayrı emirleri farklı zamanlara yerleştirir ve teslim olduktan sonra işlem emirlerini yönetebilir. ABD hisse senedi piyasalarındaki işlem hacminin %50-%77 aralığında algoritmik ticarettten kaynaklandığı bildirilmektedir (Cvitanic ve Kirilenko, 2010, s. 33).

Algoritmalar piyasaya emir gönderme zamanlamasını da kontrol edebilmektedir. Algoritmalar, gerçek zamanlı yüksek hızlı veri akışlarını okumakta ve işlem sinyallerini tespit edip uygun fiyat seviyelerini belirleyerek sonrasında uygun bir fırsat bulduklarında işlem emirlerini göndermekte, ayrıca arbitraj fırsatlarını da tespit ettiklerinde takip eden eğilime, haber olaylarına ve spekülasyonlara dayalı işlemciler yerleştirebilmektedirler. Bu algoritmalar aynı zamanda piyasaya yüksek hızlı, genelde milisaniye ya da mikro saniye cinsinden gönderilecek küçük ölçekli işlem emirlerini yönetebilmektedir. Algoritmaların yönlendirdiği bu emirler, bir piyasa yapıcı rolünü çoğaltan yüksek hızlı algoritmalar tarafından yönetilmektedir (Daştan, 2021, s. 9).

Algoritmalar, alıcılar ve satıcılar arasında piyasa yapıcı olarak çalışmaktadırlar. Başka bir ifadeyle otomatik AL-SAT sinyallerini yakalayarak, geçmişe dönük performans testleri yapabilmektedirler (Daştan, 2021, s. 8).

## 2.4. Portföy Optimizasyonu

Portföy optimizasyonu problemi en basit şekliyle standart sayısal tekniklerle kolaylıkla çözüme ulaştırılabilen bir problemdir. Diğer taraftan, çeşitlendirmenin avantajlarından faydalanılması ve toplam riskin azaltılması için fazla sayıda hisse senedine küçük tutarlarda yatırım yaparak portföy oluşturmanın bazı zorluklar bulunmaktadır. Bu tür bir yatırım stratejisi, yüksek işlem maliyetleri ve fazla sayıda hisse senedini yönetmenin zorluğundan ötürü pratik olarak oldukça zordur. Bu zorluğun üstesinden gelebilmek üzere sermayenin hisse senetlerine dağıtılması üzerine çok sayıda kısıt getirilebilir. Portföydeki varlık sayısı sınırlanabilir ya da her varlığa yatırılan sermayenin oranlarına alt ve üst limitler konulabilir. Bu kısıtlar, problemin standart optimizasyon teknikleriyle çözülmesini zorlaştırmakta ve problemi NP-Zor haline getirmektedir (Akyer, 2016, s. 19).

Geleneksel portföy kuramı, getirisi yüksek olan menkul kıymetlerin portföye dahil edilmesi gerektiğini savunmaktadır. Aynı zamanda bu kuramda portföye ne kadar fazla menkul kıymet dâhil edilecek olursa portföyün riskinin düşeceği varsayılmaktadır. Her yatırımcı oluşturduğu portföyden getiri elde etmeyi arzı eder; ancak geleceğe dönük yapılan yatırımlar her zaman belirsizlikten dolayı risk içermektedir. Bu sebeple yatırımcı, portföyün riskinin neyle ölçüleceğini bilemediği için portföy riskinden kaçınmamaktadır. İkinci portföy teorisi ise risk ve getiri kavramlarını istatistiksel olarak bir modelle tanımlayan Markowitz'in (1952) Modern Portföy Teorisi'dir. Bu teori, yalnızca portföyün çeşitlendirilmesiyle portföyün riskinin azaltılamayacağını, portföye dahil edilen menkul kıymetler arasındaki ilişkinin (korelasyon) de riskin azaltılması için etkili olabileceği Markowitz tarafından geliştirilen Ortalama-Varyans Modeli ile modele dahil edilmiştir. Markowitz, portföy yönetimine modern anlamda önemli katkılar yapmıştır. Birincisi; portföy riski, portföyü oluşturan varlıkların riskinden daha düşük olabilmektedir. Portföydeki varlıklar arasındaki korelasyon katsayısının değeri azaldıkça portföy riski de azaltılabilmektedir. İkincisi; yatırımcılar aynı risk düzeyindeki portföy çeşitleri arasından daha yüksek getirili olanı tercih etme eğiliminde olacaktırlar. Modern portföy teorisinde, yalnızca portföydeki menkul kıymetleri sayısının artırılmasıyla riskin azaltılması amacına ulaşamayacağı, portföye katılan menkul kıymetlerin getirilerinin, riskin dağıtılmasında son derece önemli olduğu ortaya konulmuştur. Finansal varlık getirileri arasındaki ilişkilerin dikkate alınması ve tam negatif ilişki içinde bulunan varlıkların aynı portföyde yer alması, hedeflenen beklenen getiriden vazgeçmeden risk değerinin düşürebileceğini göstermiştir (Başaran, 2021, s. 19).

Modern Portföy kuramının öncüsü Markowitz'in genel yaklaşımı, bir yatırımcının sahip olduğu belli tutardaki bir parayı çeşitli menkul değerlere yatırmasıdır. Bu yaklaşım yatırımcının olası portföylerden seçeceği menkul değerlerden oluşan bir portföye dayanmaktadır. Bu yaklaşım, finansman literatüründe portföy seçimi problemi olarak da adlandırılmaktadır (Karan, 2001, s. 135).

Portföy yaklaşımına göre bir yatırımcı genelde tek bir finansal varlığa yatırım yapmaz; tasarruflarını farklı finansal varlıklar arasında paylaştırır. Yatırımcıların amacı tasarruflarını çeşitli finansal varlıklar arasında optimum bir şekilde paylaşmak, başka bir deyişle belli bir karlılık düzeyinde riski minimuma indirecek veya belirli bir risk seviyesinde getiriyi en yüksek seviyeye yükseltecek biçimde portföyünü oluşturmaktır. Bu sebeple yatırımcılar portföylerini menkul kıymetlerin beklenen getirileri ve riskleri ile ilgilenerek oluştururlar (Akgüç, 1998, s. 872).

Günümüz teknoloji ortamında portföy oluşturma da menkul kıymet seçiminde, artık bilgisayarların dahil olmasıyla birlikte yeni geliştirilen yöntemler farklı ufuklara açılım sağlamıştır. Bu amaçla çeşitli optimizasyon teknikleri kullanılmakta olup; Linear ve Non-Linear programlama, parçacık sürü algoritması, genetik algoritmaları ve yapay sinir ağları gibi yaklaşımlar bu yöntemlere örnek olarak verilebilir (Durmuşşkaya ve Garayev, 2017, s.175).

Hisse senetlerinin geçmiş fiyat bilgilerinin kullanılması yoluyla çok fazla sayıda teknik göstergenin hesaplanması olasıdır. Ancak çok fazla sayıda değişkene sahip olmak aynı zamanda yönetimi zor bir veri seti manasına gelmektedir. Bu sebeple, olası en az kayıpla veri setinin daha yönetilebilecek bir boyuta indirgenmesine gerek vardır. Başlangıç veri deposunda bir araya getirilen verilerin bazıları gereksiz olabilir ya da yanıltıcı öğeler içerebilir. Bu sebeple analizlerin ileri aşamalarına geçilmeden önce bu tür öğelerin, veri setinden ayıklanmasına gerek vardır. Bu tür temizlik işlemi yalnızca analizin daha hızlı gerçekleşmesini sağlamayacak aynı zamanda ulaşılan sonuçların daha anlaşılır olmasını imkân verir ve sonuçların kalitesini yükseltebilir (Jensen ve Shen, 2008, s. 62).

## 2.5. Algoritmalarda Kullanılan Finansal Veriler

Finansal veriler çok farklı şekillerde ve biçimlerde ortaya çıkabilir. Genel olarak finansal veriler dördü bir sınıflandırmaya tabi tutulabilir. Bunlar: Temel veriler, piyasa verileri, analitik veriler ve alternatif verilerdir (Prado de, 2018, ss. 23-25):

**a) Temel veriler**, yasal düzenlemelerde ve iş analizlerinde bulunabilecek bilgileri kapsar. Genellikle finansal aracı ihraç eden firmanın raporlama dönemi sonundaki finansal tablolarından oluşur. Çoğunlukla üç ayda bir raporlanan muhasebe verileridir. Bu verilerin özel bir yönü, gecikmeli olarak rapor edilmesidir. Bu bilgiler yalnızca kamuya açık hale geldikten sonra kullanma hazır olur ve her bir veri noktasının ne zaman yayınlandığını tam olarak doğrulanması gerekir.

**b) Piyasa verileri**, borsada gerçekleşen tüm işlemlere ait verileri kapsar. Bu veriler genelde günlük olarak borsa bültenlerinde yayınlanır. Bu veriler fiyat, zaman, volatilité, hacim, temettü, faiz, emirler ve iptallere ilişkin verilerdir. Piyasa verileri, bir borsada ya da ticaret

yerinde gerçekleşen tüm ticaret faaliyetlerini içerir. Her piyasa katılımcısı alım satım kayıtlarında karakteristik bir ayak izi bırakır ve yeterince sabırla veya algoritmalar gibi duygulara sahip olmayan bir rakiple bir sonraki hareketini tahmin etmenin bir yolu bulunur.

**c) Analitik veriler**, türevsel veri olarak da değerlendirilmektedir. Bu veriler temel ve alternatif piyasa hatta başka bir analitik veri koleksiyonuna sahip olabilecek orijinal kaynağa dayalı üretilmiş verilerdir. Analist önerileri, bağımsız denetim raporları, kredi derecelendirmesi, kurumsal yönetim ilkelerine uyum derecelendirmesi gibi verileri içerir. Analitik veriyi karakterize eden unsur, bilginin içeriği değil, orijinal bir kaynaktan hazır olmaması ve yatırımcılar için özel bir yolla işlenmiş olmasıdır. Yatırım bankaları ve araştırma firmaları, şirketlerin iş modelleri, faaliyetleri, rekabeti, görünümü ile ilgili derinlemesine analizlerden elde edilen değerli bilgileri satarlar. Bazı uzmanlaşmış firmalar, örneğin haber raporlarından ve sosyal medyadan elde edilen duygular gibi alternatif verilerden elde edilen istatistikleri satarlar. Analitik verilerin olumlu bir yönü, sinyalin yatırımcılar için ham bir kaynaktan alınmış olmasıdır. Olumsuz yönleri, analitik verilerin maliyetli olabileceği, bunların üretiminde kullanılan metodolojinin taraflı veya opak olabileceğidir.

**d) Alternatif veriler**, bireylerin sosyal medyadan, haberlerden, web aramalarından, iş süreçleri işlemlerinden, kurumsal verilerden, uydulardan, coğrafi konumdan, hava durumundan, CCTV (Kapalı Devre Televizyon = Close Circuit TeleVision) kameralar aracılığıyla alınan görüntünün belirli bir konuma iletildiği sistem), uydu görüntülerinden ve video yayınlarından elde edildikleri verilerdir. Alternatif veriler, çoğu zaman rakip yatırımcıların lojistik nedenlerle kullanmaya çalışmamış, yarıda bırakmış veya yanlış işlemiş oldukları verilerdir.

## 2.6. Algoritmik Ticaret Stratejileri

Algoritmik ticaret, bireysel ve kurumsal yatırımcılar için önemli derecede disiplin, sabır ve duygusal mesafe gerektirir. Alım-satımı bir algoritma gerçekleştirdiğinden, strateji yürütülürken stratejiye müdahale etmemeye karar verilmesi gerekir. Çalışmakta olan bir algoritmaya bu şekilde müdahale edilmemesi özellikle uzun süreli düşüş dönemlerinde son derece zor olabilir. Bununla birlikte, bir geriye dönük testte son derece karlı olduğu gösterilen birçok strateji basit bir şekilde başarısız olarak damgalanabilir.

Algoritmalar tarafından uygun bir şekilde ele alınabilen siparişlerin uygunluğunu değerlendirmede ticaret öncesi analiz gerekli hale gelmiştir. Algoritmik bir ticaret belirli bir sipariş için kabul edilebilir sayılırsa, alıcı ve satıcı yatırımcıların daha sonra makro ve mikro düzeydeki sorunları ele alması gerekir. Makro düzeydeki kararlar, istenen kıyaslama fiyatının ve uygulama hedeflerinin belirlenmesini içerir. Mikro düzeydeki kararlar, istenen herhangi bir sapma kuralının belirtilmesini içerir. Bu, değişen hisse senedi fiyatlarına, piyasa hareketine veya endeks veya sektör değerlerindeki bir değişikliğe ve değişen piyasa

koşullarına bağlı olarak algoritmanın nasıl sapması gerektiğini içerir. Mikro düzeydeki kararlar, piyasa veya limit emri, gösterim boyutu, bekleme süreleri, emir revizyonları ve değişiklikler veya iptaller gibi emir gönderme kurallarının belirlenmesini de içerir. Ticaret öncesi analiz, bu bilinçli kararları vermek için gerekli verileri sağlar. Hangi emirlerin bir algoritma veya manuel müdahale gerektiren bir emir aracılığıyla başarıyla uygulanabileceğini belirlemek için yatırımcılara likidite özetleri, maliyet ve risk tahminlerinin yanı sıra alım satım zorluğu ve istikrar önlemleri sağlar. Yürütmeyi daha da iyileştirmek için potansiyel risk azaltma ve riskten korunma fırsatları hakkında fikir verebilir. İşlem öncesi analiz ayrıca yatırımcılara kısa vadeli fiyat hareketleri ve piyasa koşulları hakkında görüş geliştirmeleri için gerekli verileri sağlar (Kim, 2007, s. 53).

Daha yaygın karşılaştırma ölçütleri, işlem öncesi, işlem içi veya işlem sonrası fiyatlar olarak kategorize edilebilir. İşlem öncesi kıyaslama fiyatları, uygulama açığı olarak da bilinir. Bunlar, ticaret başlamadan önce veya zamanda tanınan bilinen fiyatlardır. Bunlara bir önceki gecenin kapanış fiyatı, açılış fiyatı ve emir girişi anındaki fiyat dâhildir. Gün içi karşılaştırma ölçütleri, bir işlem seansı sırasında açılış, yüksek, düşük veya kapanış ortalamalarında oluşan fiyatlardan emirlerden ve fiyatlamaya şemalarından oluşur (Kim, 2007, s. 53).

Emir gönderme kuralları, paylaşım miktarlarını, sipariş gönderimleri arasındaki bekleme sürelerini, revizyonları ve iptalleri ifade eder. Daha yaygın fiyatlandırma kuralları, piyasa ve limit emirlerinin yanı sıra alım, satış veya orta nokta gibi bir referans fiyata sabitlenen ve referans fiyatla değişen değişken fiyatları içerir. Bu emir türleri, gerektiğinde agresif veya pasif bir şekilde yürüterek algoritmaların en uygun şekilde öngörülen stratejiyi kullanmasını sağlar. İşlem sonrası karşılaştırma ölçütleri, işlem bitiminden sonra oluşan fiyatları veya günün kapanış fiyatını içerir. İşlem sonrası analiz, maliyet ölçümü ve algoritma performans analizinden oluşur. Maliyet, gerçekleşen fiili uygulama fiyatı ile belirtilen kıyaslama fiyatı arasındaki fark olarak ölçülür. Bu, yatırımcıların gelecekteki maliyet tahminlerini iyileştirmek için alım satım maliyeti modelinin doğruluğunu eleştirmelerine olanak tanır ve yatırım yöneticilerine daha yüksek kaliteli fiyat bilgileri sağlar (Kim, 2007, s. 53).

Algoritma stratejisi oluşturma, genellikle Hacim Ağırlıklı Ortalama Fiyat Stratejisi (VWAP) gibi belirli bir kıyaslama ile bağlantılı, likidite arayan bir dizi yapılandırılmış veya yapılandırılmamış ticaret içerir. Yapılandırılmış bir yaklaşım, geçmiş verilere ya da strateji kıyaslamalarına dayalı izleme stratejilerini içerirken, yapılandırılmamış likidite genellikle gerçek zamanlı bilgi veya fiyat kıyaslamalarıyla ilişkilendirilir. Hangi yapının uygulanacağını belirlemek için aşağıda belirtilen ticaret öncesi bilgilerle gerek vardır (Kim, 2007, s. 64):

**a) Ticaret Ufku:** Kısa vadeli ufuklar daha az yapı gerektirir. *Yarım* saatlik bir VWAP ticareti ve benzer şekilde zamanlanmış bir sabitleme ve takdir stratejisi, önemli ölçüde farklı bir sonuç vermeyecektir.

**b) Bitirme ihtiyacı:** Bir emri tamamlama ihtiyacı ne kadar yüksek olursa, programın gerisinde kalmamak için o kadar fazla yapıya ihtiyaç duyulur. Buradaki işlem öncesi bilgi türü, belirli analitiklerden ziyade portföy yöneticisi talimatlarıyla ilgilidir.

**c) Öngörülebilirlik:** Öngörülebilirlik derecesi, ufkun ve programın uygulanma derecesini belirler. Bu değerlendirme, standart sapma ölçüleri gibi ortalamalara ek olarak dağılım tahminlerinin özelliklerinin kullanılmasını gerektirir.

**d) Fiyat duyarlılığı:** Fiyat duyarlılığı arttıkça, ticarete istekliliğin reklamını yapma ihtiyacı nedeniyle yapı daha az kullanışlı hale gelir. Kısa vadeli oynaklık geçmişi ve gerçek zamanlı sapma, boyut boyunca girdilerdir.

**e) Risk toleransı:** Kıyaslamaya karşı yürütme risklerini ifade eder. Daha fazla tolerans, yapılandırılmış bir ufuk ve programa daha az ihtiyaç duyar. Ticaret öncesi bilgiler, değişen ticaret ufukları için risk, maliyet ve alfa arasındaki en uygun dengeleri belirleyebilir.

## 2.6.1. Makro ve mikro düzey stratejiler

### 2.6.1.1. Makro Düzey Stratejiler

Makro düzeyde algoritmik ticaret stratejisi kararı, ya piyasa hacminin bir yüzdesi ile katılmayı belirterek ya da siparişin piyasaya girdiği zaman bir zaman dilimleme stratejisi ve açılış, kapanış ya da varış fiyatı gibi kıyaslama fiyatı ile AL-SAT yaparak ticaret emrinin zaman içinde nasıl dilimleneceğini belirlenerek oluşturulur. Makro strateji, fona, beklenen piyasa koşullarına dayalı olarak yatırım hedeflerine ulaşma olasılığını en yüksek düzeyde sağlamalıdır (Kissell, 2021, s. 20-48).

Makro seviye strateji karar kuralları, uygun bir optimal ticaret stratejisi (örneğin, sipariş dilimleme programı veya hacim yüzdesi- POV oranı) ve gerçek zamanlı olarak belirtmekten ve uygun olduğunda likidite ve fiyatlar gibi gerçek zamanlı piyasa koşullarından yararlanacak adaptasyon taktiklerinden oluşur. Yatırımcılar için makro düzeydeki ticaret özellikleri üç aşamalı bir süreçten oluşur (Kissell, 2021, s. 48).

a) Uygulama Karşılaştırması seçilmesi,

b) Optimal Yürütme Stratejisinin seçilmesi

c) Uyarılama Taktiklerinin belirtilmesi.

### 2.6.1.2. Mikro Düzey Stratejiler

Mikro algoritmik ticaret strateji, gün içinde fiyat hareketi, değişen hacimler ve/veya artan oynaklığa dayalı olarak makro stratejiden nasıl sapılacağına belirlenmesinin belirlenmesini

içerir. Örneğin, bir yöneticinin satın alma emri varsa ve gün içinde fiyat düşerse, yönetici daha iyi piyasa fiyatlarından yararlanmak için daha saldırgan bir şekilde işlem yapmaya karar verebilir veya yönetici, olumlu eğilimin olacağına inanıyorsa daha pasif bir şekilde işlem yapmayı seçebilir, buna seans boyunca devam eder. Mikro düzeydeki strateji karar kuralları, yatırımcının, algoritmanın değişen piyasa koşullarına ne zaman ve nasıl uyum sağlayacağını ve optimal olarak belirlenmiş makro stratejiden ne zaman sapmanın fonun en iyi yararına olacağını belirlemesinden oluşur. Değişim ücretlerinin yüksek olduğu durumlarda, her bir değişimde optimizasyon, geçiş ve mikro yönetim de önemli maliyet tasarruflarına yol açabilir. (Kissell, 2021, s. 20-48).

Broker ve/veya algoritma, gerçek piyasa işlemlerinin yatırımcı tarafından belirlenen makro ve mikro kararlarla tutarlı olmasını sağlamak için emir gönderme kurallarını tanımlar. Bunlar aşağıdaki gibidir (Kissell, 2021, s. 20-21):

**a) Limit Emir Modeli (LOM):** Limit emir modeli, yatırımcının belirlediği hedeflere ve her işlem anındaki gerçek piyasa koşullarına dayalı olarak limit emirleri ve piyasa emirlerinin en iyi karışımını belirler. Portföy yöneticisinin bu kararların makro ve mikro kararlarla tutarlı olması ve bunlara bağlı kalması esastır. Pasif stratejiler daha fazla miktarda limit emir kullanacak ve agresif stratejiler daha fazla miktarda piyasa emri kullanacaktır.

**b) Akıllı Sipariş Yönlendiricisi (SOR):** Akıllı sipariş yönlendiricisi, bir işlemin nereye yönlendirileceğini belirler. SOR, limit emri gerçekleştirme olasılığı en yüksek olan varış yerini belirleyecek ve piyasa emirleri için Ulusal En İyi Teklif olarak bilinen en iyi piyasa fiyatına sahip yeri belirleyecektir. SOR, borsa ve mekânlardan gelen gerçek zamanlı verileri izler ve ayrıca karanlık havuzdaki etkinliği değerlendirir. SOR ayrıca, yöneticinin alım satım niyetlerinin korunmasını ve değerli alım satım bilgilerinin piyasaya iletilmemesini sağlamak için alım satım kalitesini değerlendirmekle görevlidir.

### 2.6.2. Hacim ağırlıklı ortalama fiyat stratejisi (VWAP)

VWAP algoritması, hisse senedinin likiditesine göre belirli bir süre boyunca veya işlem seansı boyunca büyük siparişlerin dilimlenmesini boyutlandırmak için bir kıstas olarak gerçek zamanlı ve geçmiş hacim verilerini kullanır. Tüccar, zaman dilimindeki pazar hacmiyle doğru orantılı bir miktarda hisse ticareti yapmak için algoritma için ayrık zaman aralıklarının (bazen dalgalar olarak adlandırılır) sayısını belirtir. Genellikle ana zorluk, gün boyunca VWAP'yi takip eden ticaret yapmaktır. Birkaç saat üzerinde çalışılması gereken bu tür emirlerle, VWAP algoritmasının otomasyonu, anlamlı insan gücü maliyet tasarrufu sağlar. Bu nedenle VWAP stratejisi çoğunlukla daha uzun süreli emirlerde kullanılır (Leshik ve Cralle, 2011, s. 19).

VWAP, "alış tarafı" ile "satış tarafı" arasındaki blok alım satımları için sıklıkla bir kıstas olarak kullanılır. Büyük emirlerin çok sayıda küçük emire bölünmesi, piyasa etkisi maliyeti riskini azaltma şansını artırır. Ayrıca, emir boyutunun diğer piyasa katılımcılarına görünmez olmasına yardımcı olur. Gerçek zamanlı gerçek hacimle birlikte hacim profili hesaplaması ve tahmini, bu strateji tarafından verilen siparişlerin boyutunu ve sıklığını yönetir. Sıklık, bir piyasa rakibi tarafından "tanınmayacak" şekilde ve her bir sipariş dalgasının hacmi "ön çalışma" şansını azaltmak için kullanılır. VWAP'yi hesaplamak için temel formül aşağıdaki denklem 1'de olduğu gibidir (Leshik ve Cralle, 2011, s. 20):

$$P_{vwap} = \frac{\sum(P * V)}{\sum V} \quad \text{Denklem 1}$$

$P_{vwap}$  = hacim ağırlıklı ortalama fiyat

P=Her işlemin fiyatı ve

V = her ticaretin hacmidir.

Piyasanın biçimsel ve ergonomik özellikleri nedeniyle, fiyatı olumsuz etkilememek için seansın başında ve sonunda daha fazla hacimde işlem görmesi olağandışı değildir. Genel olarak VWAP olarak bilinen Hacim Ağırlıklı Ortalama Fiyat, algoritmik ticaret için birincil kriter olmaya devam etmektedir. Günlük VWAP, günlük hisse senedi işlemlerinin kayıtları üzerinden hesaplanabilir. VWAP, her işlem için işlem gören dolar tutarının (işlem yapılan fiyat çarpı hisse senedi) belirli bir gün için işlem gören toplam hisse sayısına bölünmesiyle tanımlanır. VWAP'ı hesaplama yöntemi basittir. Bir satın alma emrinin fiyatı VWAP'den düşükse, işlem iyi kabul edilir; fiyat daha yüksekse, zayıf olarak kabul edilir. Yatırımcıların performansı, belirli bir ticaret ufku boyunca hacim ağırlıklı ortalama fiyattan daha iyi fiyatlarla emir verme yetenekleriyle değerlendirilir. Hacim, işlemlerinin piyasa etkisini azaltmayı hedefleyen katılımcılar için önemli bir piyasa özelliğidir. Bu etki, bir emrin uygulama fiyatının bir kıyaslama ile karşılaştırılması yoluyla ölçülebilir. VWAP karşılaştırması, ödenen her işlem fiyatının hacmine göre ağırlıklandırılmış toplamıdır (Kim, 2007, s. 56).

VWAP stratejileri, emirlerin yürütülmesi sürecinde bir zaman boyutu sunar. Yatırımcı, işlemin gün içinde gerçekleştirilip gerçekleştirilmeyeceğini kontrol edemezse, VWAP stratejileri, siparişin gün boyunca siparişlerin etkisini seyreltmesine izin verir (Kim, 2007, s. 56).

Kurumsal yatırımcıların ticareti, günlük hacmi aşan emirlerin (yüksek frekanslı, büyük hacimli) tahtaya doldurulmasıyla gerçekleşir. Çok sayıda hissenin alınıp satılması gerektiğinde, likidite endişeleri fiyat hedeflerini etkileyebilir. Bu nedenle, bazı firmalar müşterilerin isteklerine yanıt vermek için çok günlük VWAP stratejileri sunmaktadır. Büyük siparişlerin piyasa etkisini daha da azaltmak için müşteriler, beklenen düşük hacimli

günlerde siparişlerinin hacmini sınırlayarak kendi hacim katılımlarını belirleyebilirler. Her emir birkaç günlük emirlere bölünür ve ardından ilgili günler için bir VWAP algoritmasına gönderilir (Kim, 2007, s. 56).

VWAP stratejileri, en iyi uygulamayı en üst düzeye çıkarmak ve en düşük işlem maliyetini sağlamak için kullanılır. Alım satım maliyetleri genellikle ortalama gerçekleşen işlem fiyatının bir referans veya kıyaslama fiyatı ile karşılaştırılmasıyla hesaplanır. Bir performans karşılaştırmasının seçimi, bir yatırımcının limit sınırlı piyasa emirleri, ticaret ufukları ve birincil piyasalar, üst piyasalar ve geçiş sistemleri gibi yerler gibi emir yerleştirme stratejilerine ilişkin kararlarını etkileyecektir. Bu kararların gerçekleşen işlem maliyetleri üzerinde önemli etkisi vardır (Kim, 2007, s. 58).

### **2.6.3. Zaman ağırlıklı ortalama fiyat (TWAP) algoritması**

Bu algoritmik ticaret stratejisi, emirleri kullanıcı tarafından belirlenen bir zaman çerçevesi üzerinde az çok eşit olarak böler. Genellikle emir, belirli sayıda ayrık zaman aralığına veya dalgalara eşit olarak dilimlenir. Her ne kadar uygun olsa da bu, yatırımcıyı diğer bir rakibin ticaretinde tam da bu tür bir öngörülebilirliği arayan ve bunu "önden yürüterek" bundan hızlı bir şekilde yararlanan piyasa katılımcılarının "sniffer" algoritmalarını içerir. Bu genellikle bir dalgayı dışarıda bırakarak veya "bulanık" bir zaman aralığı veya hatta dalga başına "bulanık" sayıda paylaşım kullanarak mücadele edilir. Daha karmaşık (veya daha fazla paranoyak) işlem algoritmaları, bir rastgele sayı üretici kullanarak bu stratejiyi uygular (Leshik ve Cralle, 2011, s. 21):

TWAP, Zaman Ağırlıklı Ortalama Fiyat anlamına gelir ve tüccarların belirli bir süre boyunca bir işlemi "zaman dilimine" ayırmalarına imkân tanır. Piyasa hacmi düştüğünde tipik olarak daha az hisse senedi ticareti yapan VWAP'ın aksine, TWAP, siparişte belirtilen süre boyunca aynı miktarda hisse senedi ticareti yapacaktır. Bu, hacime bağlı olmayan alım satım emirlerine çekici bir alternatiftir. Bu senaryo, öngörülemeyen hacimli likit olmayan hisse senetlerinde emirlerin karşılanmaması gibi engellerin üstesinden gelebilir. Örneğin, öğleden sonra saat 14:00'te, bir yatırımcı likit olmayan bir hisse senedindeki bir pozisyondan 16:00 kapanışına kadar çıkmak istesin, ancak o hisse senedindeki toplam hacmin %25'inden fazlasını belirli bir zaman çerçevesi içinde gerçekleştirmek istemesin. Bu durumda algoritma aşağıdaki şekilde yürütülür (Kim, 2007, s. 60):

- Yatırımcı, XYZ hisse senedinin 50.000 hissesi için satış emri verir.
- Hacim kısıtlaması %25 olarak ayarlanır.
- Fiyat koruması olarak bir limit fiyat belirlenir.
- Saat olarak 14:00 başlangıç saati ve 16:00 bitiş saati belirlenir.

#### 2.6.4. Hacim yüzdesi algoritması (POV)

Bu stratejideki ana amaç, pazarın geri kalanı tarafından “görülmecek” kadar mevcut hacmin yeterince düşük bir yüzdesinde hacime katılırken “radar altında kalmak”tır. Emir miktarı toplamına kadar işlem gerçekleştirme oranı, piyasada fiilen işlem gören hacimle orantılı tutulur. Bu, özellikle büyük miktarda hisse ticareti yaparken belirli bir miktarda koruma sağlar (Leshik ve Cralle, 2011, s. 21).

Bu algoritmalar, %20 gibi önceden belirlenmiş bir oranda piyasa hacmi ile katılır ve tüm emir tamamlanana kadar işlem yapmaya devam eder. Algoritmalar, daha yüksek likidite zamanlarında daha fazla hisse ve daha düşük likidite zamanlarında daha az hisse ticareti yapacak ve böylece piyasa koşullarına (en azından değişen hacim profillerine) tepki verecektir. Bu hacim stratejilerinin bir dezavantajı, zaman ufkunun sonuna kadar emrin tamamlanmasını garanti etmemeleridir. Örneğin, gün hacminin %20'sini oluşturan bir emir  $\frac{1}{4}$  %20 oranında bir POV ile işlem yapıyorsa, ancak gündeki gerçek hacim yalnızca normal hacmine sahipse, emir gün sonunda tamamlanmayacaktır. Potansiyel tamamlanmamış emirler için bir güvenlik olarak, bazı brokerler, dönemin sonuna kadar tamamlanmasını sağlamak için bir parametre önerirler. Bu parametre, minimum POV oranı olarak hizmet eder ve belirlenen bitiş saatinde siparişin tamamlanmasını sağlamak için gerçek zamanlı olarak ayarlanır (Kissell, 2021, s. 32).

#### 2.6.5. Uygulama eksikliği algoritması

Optimum bir menkul kıymet ticareti stratejisi, ticaret maliyetlerinin ve uygulama eksikliğinin doğru bir şekilde ölçülmesiyle başlar. Uygulama açığı teorik bir portföy ile uygulanan portföy arasındaki getiri farkı olarak tanımlanır. Bir portföyde, bir portföy yöneticisi, fiili bir portföydeki yürütme fiyatlarıyla ilgili olarak geçerli fiyatlara bakar. Uygulama açığı, nihai, gerçekleşen işlem fiyatı ile işlem öncesi karar fiyatı arasındaki fiyat mesafesini ölçer. Uygulama açığı üç kategoriye ayırılabilir: “Kağıt portföy”, “gerçek portföy” ve “tavşan portföyü” (Kim, 2007, s. 54):

**a) Kağıt portföyü:** Kağıt portföyü ideal durumu temsil eder. Tüm menkul kıymetler, referans fiyatlarla işlem görür. İşlem maliyetleri, komisyonlar, alış-satış farkı, likidite etkisi, fırsat maliyetleri, piyasa eğilimleri ve kayma olmaz.

**b) Gerçek portföy:** Gerçek portföy gerçeği yansıtır; tüm menkul kıymetler reel piyasalarda işlem görmektedir. Piyasa etkisi, komisyonlar, teklif sorma marjı, likidite, fırsat maliyetleri ve kaymalar hesaba katılır.

**c) Tavşan portföyü:** “Tavşan” portföyü, beklenen ticaret maliyetlerini temsil eder; tüm menkul kıymetler beklenen piyasalarda işlem görür. Kağıt portföyünün alım satım maliyeti

yoktur. Gerçek portföyün yüksek ticaret maliyetleri vardır. Tavşan portföyü ikisi arasında bir yere düşer. Tavşan portföyü, yatırımcıların performansı ölçtüğü bir kriterdir.

Uygulama Eksikliği algoritması birçok yönden varış fiyatı algoritmasına benzemektedir. İlk olarak, anlamı farklı brokerler ve farklı satıcılar arasında farklılık gösterir, bu nedenle bu taraflarla tam özelliklerine göre konuşmak önemlidir. İkinci olarak, uygulama eksikliği algoritmasını optimizasyon yoluyla maliyeti en aza indirmeye çalışmaktadır. Bu, kullanıcı tarafından belirlenen bir riskten kaçınma düzeyinde maliyet ve risk arasındaki dengeyi dengeler. Algoritma ticaretinin ilk günlerinde, varış fiyatı ve uygulama eksikliği algoritmaları farklı brokerler arasında aynıydı. Böylece, uygulama eksikliğini varış fiyatından ayırt etmek için brokerlar, gerçek zamanlı uyarılama taktiklerini uygulama eksikliği mantığına dâhil etmeye başladılar. Bu kurallar, piyasa likidite kalıplarının ve piyasa fiyatlarının değiştiği zamanlarda ilk çözümün optimal olarak öngörülen stratejiden nasıl sapacağını belirler. Bu nedenle, varış fiyatı ve uygulama eksikliği algoritmasının sektör genelinde hala standart bir tanımı olmasa da, varış fiyatı algoritmasının sabit olduğu ve uygulama eksikliği algoritmasının piyasa hacimlerine ve piyasa fiyatlarına dayalı ikinci bir adaptasyon taktikleri içerdiği konusunda fikir birliği vardır (Kissell, 2021, s. 33).

#### **2.6.6. Varış fiyatı algoritması**

Genel olarak, varış fiyatı algoritması, maliyet ve risk arasındaki dengeyi sağlayan bir optimizasyondan belirlenen bir maliyet minimizasyon stratejisidir. Yatırımcılar riskten kaçınma veya alım satım aciliyet düzeylerini belirtir. Optimizasyon için elde edilen çözüm, ticaret programı veya ticaret yörüngesi olarak bilinir ve genellikle önden yüklenir. Bununla birlikte, bazı yatırımcılar, değişen likidite modellerinden yararlanmak için bu optimizasyonu statik bir program yerine bir POV oranına göre çözmektedir (Kissell, 2021, ss. 32-33).

#### **2.6.7. Sepet (basket) algoritmaları**

Portföy algoritmaları olarak da bilinen sepet algoritmaları, kullanıcı tarafından belirlenen bir riskten kaçınma düzeyine dayalı olarak maliyet ve toplam sepet riski arasındaki dengeyi yöneten algoritmalarıdır. Bu algoritmalar, işlem günü boyunca riski yönetecek ve kullanıcı özelliklerine göre değişen piyasa koşullarına uyum sağlayacaktır. Algoritmalar genellikle çoklu ticaret periyodu optimizasyon sürecine dayanır. Bu kararların sepetin genel performansını nasıl etkileyeceğine bağlı olarak gerçek zamanlı alım satım kararları verirler. Örneğin, bir sepet algoritması, mevcut likidite ve uygun fiyatlar ile karşı karşıya kalsa dahi, sepetin kalan riskini artıracaksa, bir emirde alım satımı hızlandırmamayı seçebilir. Ayrıca, sepet algoritması, likidite azlığı ve ters fiyat hareketi zamanlarında bile bir siparişte daha agresif olabilir, eğer böyle yapmak artık sepet riskinde önemli bir azalmaya yol açarsa. Tek

hisse ve sepet algoritmaları arasındaki en büyük fark, sepet algoritmasının maliyeti ve toplam sepet riskini (korelasyon ve kovaryans) yönetmesi, tek hisse algoritmasının ise stokun maliyetini ve bireysel riskini yönetmeye çalışmasıdır. Önemli sepet ticareti kısıtlamaları, nakit dengeleme, kendi kendini finanse etme, minimum ve maksimum katılım oranını içerir (Kissell, 2021, s. 33).

#### **2.6.8. Kara mızrak - likidite arayışı**

Tehditkâr olarak nitelendirilen bu algoritma, "Karanlık Havuzlar"da likidite bulmak için tasarlanmıştır. Ters seçim, karanlık bir havuz kullanıldığında ve emrin tam olarak (%100) gerçekleşmesi sağlandığındaki durumları ifade eder. Sonraki fiyat hareketi yatırımcının lehinedir (örneğin, alımlar ucuzlar ve satışlar yükselir), bu nedenle AL-SAT yapmak için beklemekten daha iyi durumda olmuş olur. Ve karanlık havuzda emirler gerçekleşmediğinde tam emirden daha az (<%100) sonraki fiyat hareketi emrinizden uzaktır (örneğin, alımlar daha pahalı hale gelir ve satışların değeri düşer). Karanlık havuzda bir miktar bilgi sızıntısı meydana geldiği veya yüksek frekanslı emirlerle etkileşimin toksik olduğu inancı, yani HFT'lerin yatırımcının aciliyeti veya hala ihtiyacı olan hisse sayısı gibi emir hakkında bilgi edinebilir. Buna karşılık, sızdırılan bilgilere dayanarak fiyatlarını ayarlarlar (Kissell, 2021, s. 43).

#### **2.6.9. Kara kutu (black-box) algoritmaları**

Kara kutu ticaret algoritmaları, genel olarak kar ve zarar algoritmaları ve/veya "robo" ticaret algoritmaları olarak adlandırılır. Bunlar, piyasa sinyallerine dayalı olarak yatırım kararları veren ve kararları piyasada uygulayan tüm algoritmaları içerir. Bazı belirli yönergeler ya da kurallar dâhilinde önceden belirlenmiş bir pozisyonu tasfiye etmekle görevlendirilen uygulama algoritmalarının tersine, kara kutu algoritmaları kârlı bir fırsat araştırması için piyasa olaylarını, fiyatları, ticaret miktarlarını vb. izler. Piyasada kar etme fırsatı oluştuğunda, algoritma anında hisseleri alıp satar. Birçok kara kutu algoritması, saniyelerden dakikalara değişen zaman ufkuna sahiptir ve bazı daha uzun zaman ufkuna, saatlerden günlere kadar uzanır. Bazı kara kutu alım satım algoritmaları, eşleme, otomatik piyasa oluşturma ve istatistiksel arbitraj içermektedir (Kissell, 2021, s. 34).

#### **2.6.10. Piyasayla paralel kalma algoritma stratejisi (PEG)**

Piyasayla paralel kalma algoritması, takip eden bir emire çok benzer şekilde, toplam emirin kesirini rastgele seçerek limit emirleri gönderir ve piyasayı sürekli takip eder, piyasanın geneli gibi davranmaya çalışır (Leshik ve Cralle, 2011, s. 21).

### 2.6.11. Büyük sipariş (iceberg) gizleme algoritması stratejisi

Burada, belirli bir hisse senedinde önemli bir pozisyon biriktirmeye çalışırken, "önden yürütmelerini" önlemek ve genel olarak piyasa etki maliyetini en aza indirmek için diğer piyasa katılımcılarından büyük bir emir saklamaya çalışılmaktadır. Bu, emri birçok küçük parçaya bölerek ve emir yerleşimini rastgele hale getirerek yapılır. Daha küçük emirlerin piyasa etkisi maliyetinden kaçınma şansını artırması beklenir. Iceberg'in daha uzun zaman dilimlerinde dağıtılabilen bir limit emri sürümü vardır. Bu büyük algoritmaların çoğu, büyük miktarlarda hisse değiştiren kurumlar tarafından kullanılmaktadır. Sadece "ön çalışma" olmadan güvenli bir şekilde ticaret yapmak, anında kârlı getiriler elde etmekten daha önceliklidir. Bu, hemen kârın gerçekleştirilmesi gereken ve ticaret hacminin küçük olduğu, çok nadiren 2500 ile ticaret başına 1000 hisse bölgesinde sınırlı sermayeli bireysel yatırımcının tam tersi bir stratejidir (Leshik ve Cralle, 2011, ss. 21-22).

### 2.6.12. Parite ticareti algoritması stratejisi

Parite ticareti algoritması, işleyişi piyasanın yönüne değil, işlem gören hisse senedinin korelasyonlu ve korelasyonsuz davranışına bağlı olduğundan, yapısı gereği piyasada tarafsızdır. Bir parite stratejisiyle işlem yapmak için, geriye bakma süresi boyunca yüksek korelasyona sahip iki hisse senedi bulunmalıdır. Algoritma, güvenlik zarfının dışına çıkmalarına dikkat edecek ve sapma, yatırımcının önceden belirlenmiş sınırını aştığında harekete geçecektir. İlerlemeyi izleyecek (bazen günler hatta haftalar gibi oldukça uzun bir süre boyunca) ve fiyatlar normal korelasyonlarına döndüğünde alım-satımı kapatacaktır (Leshik ve Cralle, 2011, s. 23).

## 2.7. Algoritmik Analiz Aracı Testler

Ticaret sistemlerini ve performans raporlarını ve öz sermaye eğrilerini değerlendirirken, sonuçların nasıl elde edildiğini ayırt etmek önemlidir. Sonuç üretmenin dört ana yolu vardır: (Dawey, 2014, ss. 49-51):

- a) Geçmişe dönük test,
- b) Örnek dışı test,
- c) İleriye dönük test,
- d) Gerçek zamanlı test.

**a) Geçmişe Dönük Test:** Geçmişe dönük test, en yaygın test yöntemidir. Aynı zamanda gerçekleştirmesi en kolay ve kötüye kullanılması en kolay olanıdır. Geliştirici yalnızca başlangıç tarihini ve bitiş tarihini (genellikle bugünün tarihi) girer, optimize edilecek tüm

parametreleri içerir ve ardından tüm hesaplamaları strateji motorunun yapmasına izin verir. Nihai sonuç, o zaman dilimi için gerçek canlı ticarete kullanılabilecek en iyi parametre seti olacaktır.

**b) Örnek Dışı Test:** Yalnızca en deneyimsiz geliştiriciler, ticaret sistemlerini tüm geçmiş veri kümesi üzerinde test eder ve optimize ederler. Sadece geçmiş verilerden analiz yapılması gerçek zamanlı alım satım sonuçlarının iyi olmaması veya en azından sürekli olarak iyi olmaması ihtimalini doğurur. Bunların çoğu, stratejinin performansını optimize edildiği aynı veriler üzerinde değerlendirmeye atfedilebilir.

**c) İleriye Dönük Analiz:** İleriye dönük analiz, geleneksel geriye dönük testlerden çok daha yavaş işler, ancak sonuçlar genellikle çabaya değerlidir. İleriye dönük analiz, ticaret yazılımı optimizasyonu ile birlikte elle yapılabilir. Günümüzde artık, birçok ticaret yazılımı paketi artık mevcut araçlarında ileriye dönük analiz içermektedir. İleriye dönük analiz, basit olarak pek çok örnek dışı dönemin toplamıdır. İleriye dönük analizin sonuçları, doğru yapıldığında, basit bir optimize edilmiş testten gerçeğe çok daha yakın olabilir.

**d) Gerçek Zamanlı Analiz:** Bazı çok başarılı tüccarlar, bu tür testlerdeki doğal çelişkiler ve sorunlar nedeniyle her türlü geriye dönük testten kaçınır. Bu insanlar, stratejileri gerçek zamanlı olarak, muhtemelen gerçek parayla bile test ederler. Böyle bir yöntemin bariz avantajı, kuralları geçmiş verilere uydurmanın ve geriye dönük önyargı kullanmanın mümkün olmamasıdır. Büyük bir dezavantaj, yalnızca piyasa hızında veri toplayabilmenizdir. Uzun yıllar boyunca gerçek zamanlı olarak test edene kadar uzun yıllar boyunca istatistik toplamak imkânsızdır. Çoğu insanın böyle bir testin tamamlanmasını bekleyecek sabrı yoktur. Diğer bir dezavantaj, strateji değiştirildiğinde saatin sıfıra dönmesi ve değerlendirmenin yeniden başlamasıdır. Bu, test süresini gerçeğe uzatabilir.

### 2.7.1. İşlem öncesi analiz

Ön ticaret analizi, yatırımcılara hem makro hem de mikro düzeyde bilinçli ticaret kararları vermeleri için gerekli verileri sağlar ve algoritmalara girdi görevi görür. Pretrade, yatırımcılara likidite özetleri, maliyet ve risk tahminleri ve hangi emirlerin algoritmik alım satım yoluyla başarıyla uygulanabileceğini ve hangi emirlerin manuel müdahale gerektirdiğini görmek için alım satım zorluğu göstergeleri sağlar. Ayrıca, algoritmik yürütme stratejilerini daha da geliştirmek için potansiyel risk azaltma ve riskten korunma fırsatları sağlar. İşlem öncesi verileri, cari fiyatlar ve kotasyonlar, likidite ve risk istatistikleri, momentum ve son işlem faaliyetinin bir hesabından oluşur. Bu aynı zamanda yatırımcılara kısa vadeli alfa modelleri geliştirmek için gerekli verileri sağlar (Kissell, 2021, s. 49).

### **2.7.2. Gün içi analizi**

Gün içi analizi, alım satım sırasında alım satım performansını izlemek için kullanılır. Bu sistemler genellikle gerçek zamanlı olarak yürütülen hisse sayısını, yürütülen hisseler için gerçekleşen maliyetleri, alım satım başladığından beri fiyat hareketini (bu, batık maliyet veya tasarruf anlamına gelir) ve beklenen piyasa etkisi maliyetini ve zamanlama riskini sağlar. kalan paylar için uygulama stratejisine ve beklenen piyasa koşullarına (ticaretin aşlangıcında beklenenlerden farklı olabilir) dayalı olarak. Daha gelişmiş gün içi analiz sistemlerinden bazıları, tüm hisseler için (strateji ve piyasa koşullarına dayalı olarak) öngörülen riske göre ayarlanmış işlem maliyeti olan z-skor tahminleri ve farklı algoritmalar ve stratejiler için öngörülen nihai işlem maliyetleriyle karşılaştırmalar sağlayacaktır. Gün içi analiz sistemleri, tüccarlar tarafından piyasa koşullarını değerlendirmek ve algoritmalarını revize etmek için kullanılır (Kissell, 2021, s. 49).

### **2.7.3. İşlem sonrası analiz**

İşlem sonrası analiz, maliyet ölçümü ve algoritma performans analizinden oluşan iki parçalı bir süreçtir. İlk olarak, maliyet, gerçekleşen fiili gerçekleştirme fiyatı ile belirtilen kıyaslama fiyatı arasındaki fark olarak ölçülür. Bu, yatırımcıların gelecekteki maliyet tahminlerini ve makro strateji kararlarını iyileştirmek için alım satım maliyeti modelinin doğruluğunu eleştirmesine ve yatırım kararlarını iyileştirmek için yöneticilere daha yüksek kaliteli fiyat bilgileri sağlamasına olanak tanır. İkinci olarak, algoritmanın optimal olarak öngörülen stratejiye bağlı kalma yeteneğini, adil ve makul fiyatlara ulaşma yeteneğini değerlendirmek ve algoritmanın optimal olarak belirlenmiş stratejiden uygun bir şekilde sapıp saptığını belirlemek için algoritmik performans analiz edilir. Yatırımcılar, brokerlerin ilan edildiği gibi teslimat yapmasını sağlamak için sürekli olarak ticaret sonrası analiz yapmalı ve ticaret öncesi maliyet tahminleriyle uyumlu olmayan uygulamaları sorgulamalıdır (Kissell, 2021, s. 50).

## **2.8. Algoritmik Ticarete Kullanılan Algoritmalar**

Algoritmik ticarete kullanılan çok sayıda algoritmadan genetik algoritma, sürü zekâsı algoritması, parçacık sürü optimizasyonu algoritması, karınca kolonisi algoritması, yapay arı kolonisi algoritması, rastgele orman algoritması, XGboost algoritması, naive-Bayes algoritması ve Chaid karar ağacı algoritması aşağıda incelenmiştir.

### 2.8.1. Genetik algoritmalar

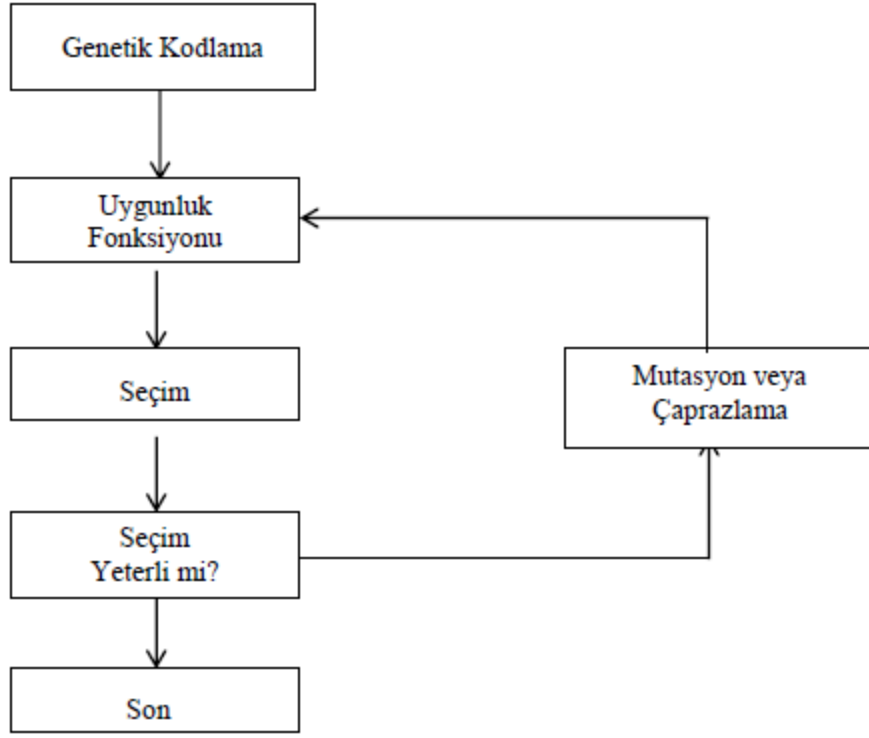
Genetik algoritmalar, stokastik bir arama yöntemi özelliği taşımaktadır. Darwin'in "en iyi olan yaşar" ilkesine dayalı olarak biyolojik sistemlerinin gelişme sürecini benzeten genetik algoritmalar, ilk kez Holland tarafından ileri sürülmüştür. Genetik algoritmalar, sezgisel bir yöntem olmaları sebebiyle verilen bir problem için optimum sonuca ulaşamayabilir, ancak bilindik yöntemlerle çözülemeyen veya çözüm süresi problemin büyüklüğü ile üstel olarak artan problemlerde optimuma yaklaşan sonuçlar sağlamaktadır (Akay vd., 2002, s. 130).

Genetik Algoritmalar, Darwin'in evrim kuramındaki seçme, mutasyon ve çaprazlama işlemlerinden esinlenilerek geliştirilmiştir. Genetik algoritmalarda iyi çözümler, olası daha iyi bir çözümü üretmek için seçilir ve değiştirilir (Chambers, 1998, s. 13). Genetik algoritmalar, doğal genetik kavramları kullanırlar. Genetik algoritmalarda popülasyon, her nesilde problemin muhtemel çözümlerinden oluşan bireylerin oluşturduğu topluluktur (Elmas, 2011: 388). Yapay sistemlerde, bireyler sıklıkla diziler ya da kromozomlar olarak adlandırılmaktadır. Kromozomlar, gen (özellikleri, karakterleri, kodlama) olarak nitelendirilen birimlerden oluşmaktadır. Her bir gen, kalıtımın bir ya da birkaç karakteristiğini kontrol etmektedir (Michalewicz, 1996, s. 28).

Holland, canlıların genetik dizilimini, optimizasyon problemlerine çözüm bulabilmek amacıyla kullanmıştır. Genetik algoritma tasarımının amacı, iyi genlere sahip nesillerin kaliteli bireyler üretmesi esasına dayanır. Kötü gene sahip bireyler (elemanlar) varlıklarını sürdüremezler. Algoritma ile üretilen bireyler iyi gene sahipse optimal çözüm olabilirken, kötü gene sahip bireyler ise hafızadan silinmesi gereken çözümlerdir. Genetik algoritmada elitizm oranı, mutasyon oranı, çaprazlama oranı, kromozom sayısı ve iterasyon sayısı algoritmaya dâhil edilecek olan parametrelerdir. Genetik algoritmanın genel yapısı adımlar halinde aşağıdaki gibidir (Vural, 2005, s. 14):

- a) Başlangıç için popülasyon oluşturulur;
- b) Elitizm yöntemi için en iyi kromozomları belirlenir ve eşleştirme havuzuna atılır,
- c) Çaprazlama ve mutasyon gerçekleştirilir.

Diğer taraftan bir genetik algoritmanın işleyiş mekanizması oldukça sadedir. Sadece dizilimlerin kopyalanmasını ve kısmi değişimlerini içerir (Vural, 2005, s. 14). Çok sayıda pratik problemde iyi sonuçlar verebilen basit bir genetik algoritmanın ana süreçlerinin akış şeması, Şekil 2.3'de gösterildiği şekildedir (Küpeli, 2013, s. 9):



**Şekil 2.3.** Genetik Algoritmanın Ana Akış Şeması

**Kaynak:** Küpeli, 2013, s. 9

Genetik algorithmada çözüm süreci, portföye dâhil edilecek herhangi bir finansal aracın kodlanarak programın çözebileceği şekle dönüştürülmesiyle başlamaktadır. Sonrasında çözümün başlatılabilmesi için getiriye maksimum kılacak ya da riski minimuma indirecek veya ikisini birleştirecek bir uygunluk fonksiyonu oluşturulur. Amaç fonksiyonu oluşturulduktan sonra seçim aşamasına geçilir. Seçim aşamasında en iyi sonuca götürebilecek finansal varlıkların seçimi yapılmaktadır. Seçimi yapmak için çeşitli yöntemler kullanılmaktadır. (Turnuva Yöntemi, Rulet çemberi gibi). Bir sonraki aşamada çözüm sürecindeki kısıtlar dikkate alınmaktadır. Çaprazlama sürecinde birbiriyle eşleşecek olan menkul değerler bir araya getirilir ve daha iyi sonuçları veren çözümler bir sonraki aşamaya geçer. Bu arada çaprazlamayı gerçekleştirebilmek için farklı yöntemler de kullanılabilir. (Tek noktalı, Çok noktalı vs.) Eğer çaprazlamayla istenilen neticeye ulaşılamamışsa, bu sırada araya mutasyon girmektedir. Mutasyon, yeni bir operatör görevini üstlenmektedir. Tüm bu aşamalar tamamlandıktan sonra eğer istenilen sonuca varılamamışsa bütün bu aşamalar en başa dönülerek yenilenir ve bu süreç en iyi çözümü buluncaya değin sürdürülür (Durmuşkaya ve Gurayev, 2017, s. 179).

Genetik algoritma, ilk defa Chang vd. (2000) tarafından portföy optimizasyonu problemine uyarlanmış, BT ve Yapay sinir ağları teknikleriyle karşılaştırılmıştır. Ulaşılan bulgulara göre,

geliştirdikleri her algoritmanın farklı veri setleri üzerinde birbirine üstünlükleri tespit edilmiş ve en etkin portföyün seçilmesi için tüm algoritmaların oluşturduğu çözüm havuzundan faydalanılabileceği sonucuna varılmıştır (Akyer, 2016, s. 42).

Yatırımcıya maksimum getiriyle birlikte minimum risk sağlayabilmek için amaç fonksiyonun maksimum olması gerekmektedir. Optimizasyon problemini çözebilmek için genetik algoritma yöntemi ile optimum portföyün oluşturulması amaçlanmıştır. Portföy optimizasyonu problemi için genetik algoritma yöntemi aşağıdaki adımlarla uygulanır (Başaran, 2021, s. 26).

**1. Adım:** Başlangıç çözümü için kromozomdaki genler, portföy optimizasyon problemi için hisse senetleri  $[0,1]$  aralığında rastgele ağırlıklandırılır.

**2. Adım:** Ağırlık oranlarıyla her bir kromozoma (portföy) ait amaç fonksiyonu değeri hesaplanır. Uygunluk değerleri maksimum olandan minimum olana doğru sıralanır. Sıralamanın sonucunda iyi olan kromozomlar elitizm yöntemiyle bir sonraki aşamalar için eşleşme havuzuna aktarılır. Bu adımın amacı bir sonraki adımda gerçekleşecek olan çaprazlama işlemi için yeni bireylerin üretilmesi amacıyla kaliteli ebeveynleri bir havuzda buluşturmaktır.

**3. Adım:** Eşleşme havuzundan rastgele 2 kromozom seçilir. Çaprazlama oranı ile  $(0,1)$  aralığında üretilen rastgele bir sayı karşılaştırılır ve çaprazlama işlemi olup olmayacağına karar verilerek, işlem uygulanır. Çaprazlamanın sonucunda elde edilen yeni bireyler, kromozomlar eski kromozomlar ile karşılaştırma yapılmaksızın yer değiştirirler.

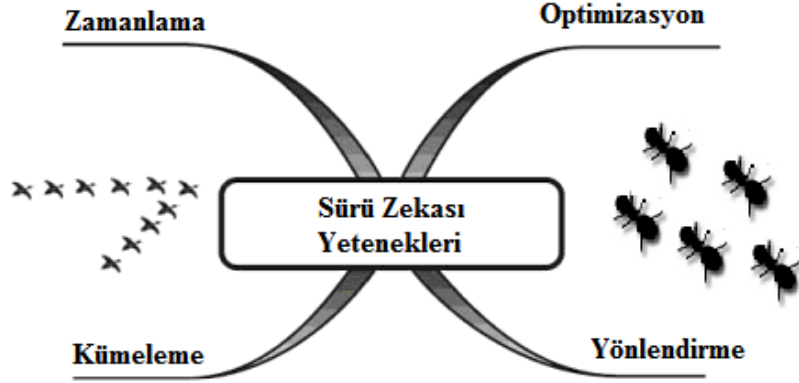
**4. Adım:** Rastgele bir kromozom seçilir. Önceden belirlenmiş mutasyon oranı ile  $(0,1)$  aralığında üretilen rastgele sayı karşılaştırılır ve mutasyon işlemi uygulanıp uygulanmayacağına karar verilir. Mutasyon işlemi uygulanan kromozom eski kromozomla yer değiştirir.

### 2.8.2. Sürü zekâsı algoritmaları

Genetik algoritmalar, tavlama benzetimi gibi doğal olaylara dayandırılarak geliştirilen algoritmalar, araştırmacıların başkaca doğa olaylarını da inceleyerek bunların modellenmeleri hususunda esin kaynağı olmuşlardır. Sürü halinde yaşayan canlıların sergiledikleri zekice davranışlar bu doğa olayları arasında yer almaktadır (Küçüksille ve Tokmak, 2011, s. 203).

Sürü zekâsı algoritmaları, kendi kendine organize olabilen ve merkezi olmayan sürü davranışları üzeri ne bina edilmiş yapay zekâ tekniklerinden bir tanesidir. Sürü zekâsı kavramı, Beni ve Wang (1989) tarafından ortaya atılmıştır. Sürü zekâsı algoritmalarına göre sürü birbiriyle etkileşim halinde olan basit tek bireylerin bir araya getirdiği topluluklardan oluşmaktadır. Genel olarak bireylere nasıl davranacağını telkin eden merkezci bir kontrol mekanizması bulunmamakta, ancak yerel etkileşimler sürü davranışlarını ortaya

koymaktadır. Karınca kolonisi, arı kolonisi, kuş sürüsü gibi doğal topluluklar sürü zekâsına örnek gösterilmektedir. Aşağıda Şekil 2.4.'de sürü zekasının yetenekleri gösterilmiştir (Irmak, 2019, s. 27).



Şekil 2.4. Sürü Zekasının Yetenekleri

**Kaynak:** Irmak, 2019, s. 27

#### 2.8.2.1. Parçacık Sürü Optimizasyonu (PSO)

Parçacık Sürü Optimizasyonu Kennedy ve Eberhart (1995) araştırmalarında, kuş ve benzeri hayvanların sosyal davranışlarını tanımak amacıyla önerilen, popülasyon esaslı sezgisel bir optimizasyon algoritması yöntemidir. PSO, kuş veya balık sürülerinin yiyecek arayışı çabalarından veya belirli bir yere giderlerken sürünün kendi içindeki iletişimlerinden psikolojik olarak esinlenilmiştir. Kuşların yiyecek arayışı çabaları ve davranışları bir probleme çözüm aranmasına benzetilmiştir. Yiyeceğin nerede olduğunu bilmeyen sürüdeki kuşlar eş zamanlı bir şekilde arama bölgesine dağılırlar, daha sonra tekrar eş zamanlı olarak bir araya gelirler ve yiyeceğin nerede olduğu konusunda paylaşımda bulunurlar. Bu paylaşım sonrasında hangi kuşun yiyeceğe daha yakın olduğu öğrenilir ve kuş sürüsü de yiyeceğe en yakın olan bu kuşu takip ederek yiyeceğe ulaşır (Çelenli vd., 2015, s. 27).

Bu algoritmaya göre sürüdeki her bir kuş bir çözümü temsil eder. O halde PSO'nun genel amacı, sürüdeki kuşlar veya çözümler arasındaki sosyal bilgi paylaşımının geliştirilmesidir. Sürüdeki her bir kuşa "parçacık" ve bu parçacıklardan oluşan topluluğa (kuş sürülerine) da sürü adı verilir. PSO'da yiyecek arama faaliyeti, sürüdeki kuşlar ya da parçacıklar tarafından gerçekleştirilir. Parçacık olarak adlandırılan her bir çözüm, her birim hareketinin sonunda bulunduğu koordinatları bir fonksiyona iletir. Bu fonksiyona "uygunluk fonksiyonu" adı verilir. Bu şekilde parçacığın uygunluk fonksiyonu veya uygunluk değeri ölçümlenmiş olur. Parçacıkların, uygunluk değerini, bu değerinde elde ettiği koordinatları, her birim hareketteki hızını ve yönünü hafızasında tutması gerekir. Parçacığın, uzayda yiyecek arayışı içinde geldiği en iyi (en yakın) uygunluk değeri ve bu değerinde elde edildiği pozisyonlar çözümü verir.

Parçacık, yalnızca kendisiyle olan en iyi koordinatları değil, bununla birlikte komşularının da sahip olduğu en iyi koordinatları optimal çözüme varmak için kullanır. Parçacığın her iterasyonunda, hangi yönde ve ne kadar hızla hareket edeceği, komşularının (diğer parçacıkların) en iyi koordinatları ve kendi kişisel en iyi koordinatlarının birleşimi belirlemektedir. Parçacık optimum çözüme yaklaşırken, bir sonraki adımını hem kendisinin hem de sürünün en iyi uygunluk değerine ve pozisyonuna göre ayarlamaktadır. Parçacıklar, ilk pozisyonlarını rastgele tayin ederler. Daha sonra adım adım hareket ederek, optimum çözüme ulaşabilmek için çözümü araştırırlar. Sürüdeki her parçacık, optimum çözüme ulaşmayı arzu eder. Optimum çözüme doğru yapılan her harekette, sürü içindeki en iyi uygunluk değerini veren parçacık dikkate alınır. Bu şekilde iterasyonlar ilerleyerek, sürü içindeki parçacıklardan en az birinin optimum çözüme ulaşması beklenir (Çelenli vd., 2015, s. 28).

Parçacık optimizasyon algoritması, evrensel (n) boyutlu uzayda en iyi çözümün nokta ya da yüzey olarak ifade edildiği problemler ile ilgilenmektedir (Irmak, 2019: 27). Genetik algortmada olduğu gibi, PSO'da da parçacıkların her iterasyonda hesaplanan bir hız ile pozisyonunu belirlediği popülasyon temelli bir arama yöntemidir. Bu sebeple, hız optimal uzaklığa ulaşılması için pozisyonunu ayarlama da önemli bir rol oynamaktadır. Genetik algortmalardan farklı olarak PSO'da popülasyon filtrelenerek evrimleşmek yerine korunarak ilerler. Genelde, her parçacığın davranışı grup belleği ile bireysel belleğin bir uzlaşmaya varmasıdır. PSO algoritmasının uygulamasının temel adımları aşağıdaki gibidir (Akyer, 2016, s. 46):

1. Adım: Başlangıç popülasyonunu rastgele oluşturulması,
2. Adım: Popülasyondaki her bireyin uygunluk değerinin (örneğin sharpe oranı) hesaplanması,
3. Adım: Her birey için bireysel en iyi pozisyonun belirlenmesi,
4. Adım: Popülasyondaki en iyi bireyin global en iyi pozisyonunun belirlenmesi
5. Adım: Her bireyin hızının güncellenmesi,
6. Adım: Her bireyin pozisyonunun güncellenmesi,
7. Adım: Durdurma ölçütü sağlanmadıysa adım 2'ye gidilmesi
8. Adım: Algoritmanın bitirilmesi ve sonuçların raporlanması.

Parçacık sürüsü eniyileştirmesi algoritması, parçacıkların çözüm uzayında hareket ederek en iyi sonuca ulaşmaları esasına dayanmaktadır. Her parçacık, pozisyon, hareket ve en iyi pozisyon bilgilerine sahiptir. Her parçacık çözüm uzayında hareket eder ve sonuca ne kadar yakın olduğu bilgisi sürüyle paylaşılır. Bu şekilde sürü içerisindeki parçacıklar tamamen rastgele değil, en iyi sonuca yaklaşacak biçimde eşgümlü olarak hareket etmeye başlarlar

(Poli vd., 2007, ss. 33-34). Bir parçacığın sonuca ne kadar yakın olduğu veya ne kadar iyi bir çözüm olduğu uygunluk fonksiyonuyla tanımlanır. Uygunluk fonksiyonu, problemin tanımına göre farklılıklar gösterir. Parçacıklar, hareket etme formüllerinde ise hem şimdiye kadar buldukları kendi en iyi pozisyon bilgilerini hem de sürüde bulunan en iyi pozisyon bilgisini kullanarak yeni pozisyonlarını belirlemektedirler (Bayram, 2014, s. 15).

#### 2.8.2.2. Karınca Kolonisi Optimizasyonu Algoritması

Karınca kolonisi optimizasyonu algoritması, matematiksel model algoritmalarını kullanarak zor kombinasyonel optimizasyon problemlerine yaklaşık çözümler bulmaya yarayan bir algoritmadır. Yapay karıncalar problem diyagramı üzerinde gerçek karıncaların diğer karıncalara yardımcı olması için yola bıraktıkları feromon kokusu gibi yapay feromonlar bırakmaktadır. Karıncalar bu bırakılan feromon kokusunun yoğunluk derecesine bakarak yönlerini tayin etmektedirler. Bununla birlikte karıncalar, zaman zaman diğer karıncaların aynı yolda gitmesini engellemek ayrıca hem yeni hem de daha kısa yolların bulunabilmesi için rastgele seçimler de yapmaktadırlar (Karaboğa, 2011, s. 33).

Karınca kolonisi algoritması bir optimizasyon yöntemi olup, veri tabanındaki değişkenlerin sınıflandırılması için de kullanılmaktadırlar. Karınca kolonisi algoritması, sürü zekâsı doğada var olan böcek ve diğer sosyal davranışlara sahip hayvanlardan ilham alınarak geliştirilmiş bir problem çözme yaklaşımıdır. Üzerinde en fazla çalışılan ve en başarılı optimizasyon tekniklerinden biri olan karınca kolonisi optimizasyonu, karıncalardan esinlenerek tasarlanmıştır. Karınca kolonisi optimizasyonu algoritması, bazı karınca türlerinin yiyecek kaynağı arama davranışlarından esinlenmiştir. Karıncalar diğer karıncaların takip edebileceği olası yerleri işaretlemek için yere feromon maddesi bırakmaktadır. Karınca kolonisi optimizasyonu ise benzer bir yöntemle optimizasyon problemlerini çözmeye çalışmaktadır (Dorigo vd., 2006, ss. 28-29). Karıncalar parametreler arasında gezecek ve gezinirken parametreler arasında geçiş kurallarına göre feromon (izler) bırakacaktır. Bu izlerin miktarı, iyi modeli oluşturacak kural listesini belirlemektedir. Bu şekilde en iyi başarıma oranına sahip model bulunacaktır.

#### 2.8.2.3. Yapay Arı Kolonisi Yaklaşımı

Yapay arı kolonisi algoritması, bal arısı sürülerinin kendilerine özgü zeki davranışlarından esinlenerek, arıların besin ararlarken kullandıkları yöntemlere bakılarak tasarlanmış bir optimizasyon algoritması türüdür. Sürü zekâsına dayalı olan bu algoritma, doğada sürüler halinde hareket eden arıların besin bulabilmek için sergiledikleri davranışları esas alarak optimizasyon problemlerini çözmek için kullanılmaktadır. Algoritmada, görevli arıların sayısı toplam yiyecek kaynağına eşittir. İşçi arıların sayısı gözcü arıların sayısına eşittir. Kaynakta

görevli arı kaynaktaki nektar (yiyecek) miktarı bitince kaşif arı olmaktadır. Arı kolonilerinin yiyecek kaynaklarının konumları çözülmek istenen problemin olası çözümlerine, nektar miktarı ise çözümün kalitesini temsil etmektedir. Yapay arı kolonisi algoritması en fazla nektara sahip kaynağın yerini bulmaya tespit etmeye, arama uzaydaki çözümlerden problemin minimumunu ya da maksimumunu veren noktayı (çözümü) bulmaya çalışmaktadır (Akay, 2009, s. 301).

### **2.8.3. Apriori birliktelik kuralı algoritması**

Birliktelik kuralları analizi, veri madenciliği uygulamalarında, olayların meydana gelmesinde birlikte hareket eden etkenlerin ortaya çıkarılmasında kullanılmaktadır. Birliktelik kuralları üretilirken, “destek” ve “güven” ölçütleri belirlenir. Bir kuralın destek ölçütü, o kuralın geçtiği olay sayısının veri setindeki tüm olayların sayısına oranlanmasıyla hesaplanır. Güven ölçütü ise kuralda birlikte hareket ettiği belirlenen unsurların birbirlerine oranlanmasıyla hesaplanır. Kurallar oluşturulurken araştırmacı tarafından verilen minimum destek ve güven ölçütü üzerinde yer alanlar dikkate alınmaktadır. A ve B gibi iki elemanın birlikte hareket etme kuralı  $A \Rightarrow B$  şeklinde ifade edilir. Oluşan kuralların kalitesini ve anlamlılığını gösteren en önemli parametre ise güven ölçütü olmaktadır. Güven ve destek ölçütlerinin dışında, conviction, leverage ve lift parametreleri de kullanılmaktadır (Ünsal, 2020, s. 109).

Birliktelik kuralı algoritması, oalyda sık geçen öğelerin tespit edilmesi ve sık geçen bu öğelerden güçlü birliktelik kurallarının üretilmesi olmak üzere iki aşamada uygulanmaktadır. Birliktelik kuralının ilk aşaması için kullanılan Apriori Algoritması, sık geçen öğeler madenciliğinde kullanılan en popüler ve klasik algoritmadır. Bu algoritmada nitelikler ve veriler, Boolean ilişki kuralları ile değerlendirmeye alınır (Özçakır ve Çamurcu, 2007, ss. 21-22). Bu algoritma özünde tekrarlayan bir niteliğe sahiptir. Algoritma uygulamasında, veri setinde yer alan tüm elemanların destek ölçütlerini hesaplanır ve belirlenen minimum destek ölçütünün altında kalan elemanlar elenir. Sonraki aşamada destek ölçütüne uygun elemanlar ile ikili, üçlü, vb. şekilde kümeler oluşturularak her bir kümenin destek ölçütü hesaplanır ve yine minimum destek ölçütüne uyanlar ile işlem sürdürülür. Nihayetinde oluşturulan kuralların güven ölçütleri hesaplanır; belirlenen minimum güven ölçütü değerinin üzerinde olan kurallar ortaya çıkartılmış olunur (Ünsal, 2020, s. 109).

### **2.8.4. Karar ağacı algoritmaları**

Veri madenciliğiyle verilerin sınıflandırılmasında en çok kullanılan yöntemlerden bir tanesi de karar ağacı algoritmalarıdır. Karar ağacının şemasında her bir düğüm bir niteliği temsil etmekte, dallar ve yapraklar ise ağaç yapısının elemanlarını temsil etmektedir. En üstteki eleman kök, en alttaki eleman yaprak ve bunların arasında kalan elemanlar ise dal şeklinde adlandırılır. Karar ağacı algoritmasında kurallar kökten yaprağa doğru inilerek (IF-THEN

rules) yazılır. Karar ağacı algoritmalarının temel amacı, verileri keşfetmek değil, verilerin üzerinde basit bir sınıflandırma modeli oluşturmaktır (Daş ve Türkoğlu, 2014, s. 382). Aşağıda karar ağacı algoritmalarını kullanan, CHAİD Karar Ağacı Algoritması, Aşırı Gradyan Artırma (XGBoost) algoritması ve Rastgele Orman Algoritması incelenmiştir.

#### 2.8.4.1. CHAİD Karar Ağacı Algoritması

CHAİD karar ağacı algoritması, Kaas (1980) tarafından en iyi bölmeyi hesaplayabilmek için istatistiki olarak anlamlı bir farklılığın olmadığı, hedef değişkene uyum sağlayan çiftlerde tahmin değişkeninin muhtemel kategori çiftini birleştirmesiyle oluşturulmuş bir algoritmadır. En uygun bölümleri seçmek için kullanılan entropy ya da gini dizileri yerine chi-square testi kullanılmaktadır. En iyi bölmeyi hesaplamak için tahmin değişkenleri hedef değişkene uyan bir çiftin içinde istatistik olarak anlamlı bir fark kalmayınca değin birleştirilmektedir. CHAİD algoritmasının diğer yöntemlerden en önemli farkı CHAİD algoritmasının çoklu ağaçlar türetmesidir (Türe vd., 2008, s. 3).

#### 2.8.4.2. XGboost Algoritması

XGBoost algoritmasında, karar ağaçları sıralı olarak oluşturulmaktadır. XGBoost algoritmasında ağırlıklar önemli bir rol oynamaktadır. Tüm bağımsız değişkenlere ağırlıklar, atanır ve daha sonra sonuçları tahmin eden karar ağacına beslenir. Karar ağacı tarafından yanlış tahmin edilen değişkenlerin ağırlıkları artırılır ve bu değişkenler daha sonra ikinci karar ağacına beslenir. Daha sonra, bu bireysel sınıflandırıcılar daha sonra güçlü ve daha kesin bir model vermek için toplanır (Can, 2020, s. 17).

Aşırı Gradyan Artırma (XGBoost) algoritması, karar ağaçları algoritmasında ve makine öğrenmesinde çok tercih edilen bir araç haline gelmiştir. Bu algoritma sınıflandırma, sıralama ve regresyon işlemlerinde yüksek performans sağlayan denetimli öğrenme alanında önemli bir araçtır. Denetimli öğrenme, bir dizi etiketli eğitim örneğinden tahmin edici bir model çıkarma görevini görmektedir. Bu tahmin edici modelle bir üretim hatasının meydana gelip gelmediğini belirleme veya belirli bir günde fiyat ya da hacmin tahmin edilmesi gibi problemlere çözüm üretilebilmektedir. Bu kapsamda XGBoost algoritması, doğru modeller üretebilmek için “artırma” adı verilen bir işlemi uygulayan, denetimli bir öğrenme algoritması özelliği taşımaktadır (Mitchell ve Frank, 2017, ss. 127-129).

XGBoost algoritmasındaki artırılmış ağaçlar, sınıflandırma ve regresyon ağaçlarına bölünmüştür. Bu algoritmanın esası, amaç fonksiyonun değerinin optimize edilmesine dayanmaktadır (Zheng vd., 2017, s. 1171). XGBoost algoritmasının en önemli niteliği, tüm senaryolarda ölçeklenebilir olmasıdır. Sistem, tek bir makinedeki mevcut popüler

çözümlerden 10 kattan daha hızlı çalışarak dağıtılmış ya da hafıza sınırlı ayarlarda milyarlarca örneğe ölçeklenir (Chen ve Guestrin, 2016, s. 785).

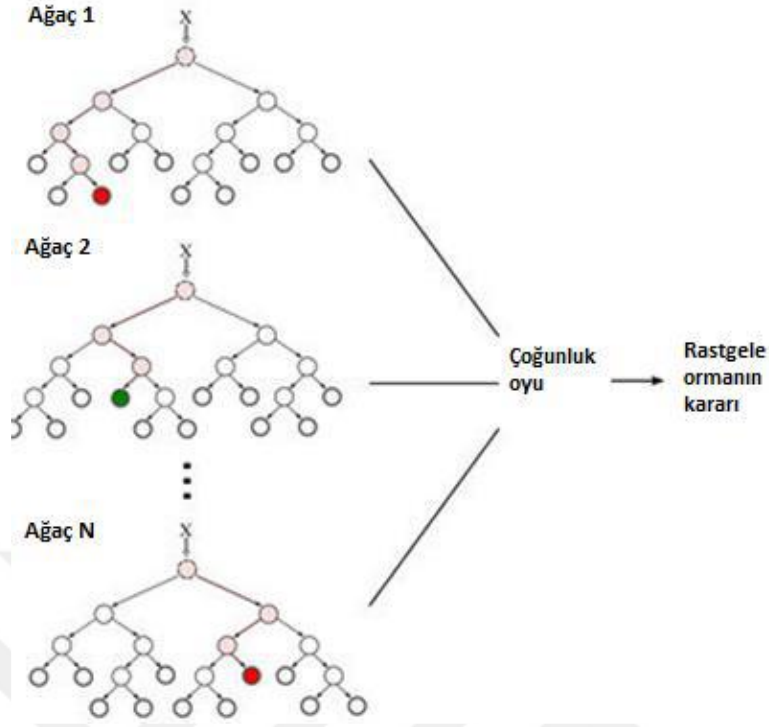
#### 2.8.4.3. Rastgele Orman Algoritması

Rastgele Orman Algoritması, Breiman (2001) tarafından geliştirilen karar ağaçları ve düğümlerden oluşturulan bir yöntemdir. Rastgele orman algoritması ile sınıflandırma ve regresyon analizleri gerçekleştirilebilmektedir. Bu algoritmaya göre düğümleri dallara ayırmak için düğümlerdeki rastgele değerlerden en iyi olanı seçilir ve oluşturulan karar ağaçlarına belirli ağırlıklar atanır. Bu ağırlıklar karar ağaçlarının iç hatalarına göre tespit edilmekte ve en düşük hataya sahip olan karar ağacına en yüksek, en yüksek hataya sahip karar ağacına en düşük ağırlık atanmaktadır. Atanan bu ağırlıklar sınıflandırma tahmininde oy verme işlemi için kullanılmaktadır. Daha sonra bu oylar toplanarak son karar verilmektedir (Ustalı vd., 2021, s. 7).

Karar ağaçlarının, tahminlerin varyansını artıran fazla uydurmaya meyilli olduğu bilinmektedir. Bu sorunu gidermek üzere, rastgele orman algoritması, daha düşük varyanslı topluluk tahminleri üretmek üzere tasarlanmıştır. Rastgele orman algoritması, verilerin önyüklenmiş alt kümeleri üzerinde bağımsız olarak bireysel tahmin edicileri eğitme anlamında, torbalama ile bazı benzerlikleri paylaşmaktadır. Torbalama ile temel fark, rastgele ormanların ikinci bir rastgelelik düzeyi içermesidir. Her bir düğüm bölünmesini optimize ederken, tahmin edicileri daha fazla ilişkilendirme amacıyla niteliklerin yalnızca rastgele bir alt örneği (değiştirilmeden) değerlendirilecektir (Prado, 2018, s. 98).

Rastgele orman algoritması, eğitim verisi ve test verisini karar ağacı modeliyle uygulayarak iyi sonuçlara ulaşmayı sağlamaktadır. Diğer sınıflandırma ve regresyon modellerine göre daha iyi sonuçlara ulaşabilen bir modeldir. Bunun sebebi ağaç sayısının istenilen şekilde belirlenmesi ve en değerli özneliği belirleme de kullanılabilir olmasıdır. Rastgele orman algoritmasında, değişken sayısı ve ağaç sayısı kullanıcıdan alınır. Parametreler alındıktan sonra veri kümesinin 2/3'lük kısmı eğitime tabi tutulur ve öğrenme amacıyla kullanılır. Geriye kalan 1/3'lük veri kısmı da test aşamasında kullanılır. Oluşturulan her düğüm için ayrı ayrı (m) değeri karışık bir biçimde seçilir ve seçilen değerler arasında en iyi sonucu veren dal tespit edilir. Bu uygulama için GINI dizini kullanılmaktadır. Aşağıdaki denklem 2'de (P<sub>j</sub>) her veri için kendisinden küçük ve büyük sayıların bölümünün karesini, (n) ise seçili olan veriyi göstermektedir (Can, 2020, s. 14):

$$\text{GINI (T)} = 1 - \sum_{j=1}^n (P_j)^2 \quad \text{Denklem 2}$$



**Şekil 2.5.** Karar Ağaçlarından Rastgele Orman Oluşturulmasının Şeması

**Kaynak:** Can, 2020, s. 14

Şekil 2.5'te rastgele orman algoritması görselleştirilmiştir. Rastgele orman algoritması dört adımda uygulanmaktadır. Bu adımlar aşağıdaki gibidir (Okumuş ve Aydemir, 2017, s. 3):

1. Adım: İlk adımda veri özelliklerine göre oluşturulmak istenen karar ağacı sayısı ( $n$ ) belirlenir.
2. Adım: Oluşturulan karar ağaçlarındaki her düğümde rastgele ( $m$ ) adet değişken seçilir ve Gini indeksi ile hesaplanarak en iyi dal tespit edilir.
3. Adım: Bir önceki adımda belirlenen en iyi dal iki alt dala ayrılır. Bu işlem Gini indeksi sıfır oluncaya kadar başka bir deyişle her düğümde bir sınıf kalıncaya değin sürdürülür.
4. Adım: ( $n$ ) tane karar ağacının ayrı ayrı yaptığı tahminler içerisinde en fazla oyu alan sınıf son karar tahmini olarak seçilir.

### 2.8.5. Naive-Bayes sınıflandırma algoritması

Naive Bayes algoritması, belirli bir veri kümesindeki değerlerin frekansını ve kombinasyonlarını sayarak bir olasılık kümesini hesaplayan basit bir olasılık sınıflandırıcıdır. Algoritma Bayes teoremini kullanarak sınıf değişkeninin değeri dikkate alındığı zaman tüm

değişkenlerin bağımsız olduğunu varsayar. Bu koşullu bağımsızlık varsayımı, gerçek dünya uygulamalarında nadiren geçerlidir, bu nedenle Naive olarak karakterize edilir, ancak algoritma çeşitli kontrollü sınıflandırma problemlerinde hızlı bir şekilde öğrenme eğilimindedir (Sarıtaş ve Yaşar, 2018, s. 90).

Bayes sınıflandırma tekniği, mevcut hali hazırda sınıflandırılmış verileri kullanarak yeni bir verinin mevcut sınıflardan herhangi birine girme olasılığını tespit eden bir yöntemdir. Veri madenciliğinde Bayes sınıflandırma tekniğine dayanan Naive Bayes algoritması, verilerin sınıflandırılmasını sağlayan en temel algoritmalarından bir tanesidir. Kümeleme yöntemleri değişik sonuçlar üreten modeller olarak tanımlanmaktadır. Bu yöntemlerin hepsinde test verisi formunda yeni bir örneğin, eğitim veri setindeki örneklerle göre analiz edilerek sınıflandırılması yapılmaktadır. Hisse senedi ticaretinde, temel olarak verilerin AL-SAT olarak iki durumu göz önünde bulundurularak çoklu kriterlere göre sınıflandırma yapılması esasına dayanarak çalışan bir algoritmadır (Silahtaroglu, 2008, s. 23).

Bu algoritma Bayes Teoremi'ni temel olarak alır. Naive Bayes algoritması, veri kümesi içindeki değerlerin kullanım sıklıklarını ve oluşabilecek kombinasyonları sayarak olasılık hesaplayan sınıflandırıcı bir algoritmadır. Bu algoritmanın büyük verilerle kullanılması daha etkilidir. Naive Bayes algoritmasında eğitim aşaması kullanılmamaktadır. Algoritma, verileri sınıflandırmak için bağımlı değişken ile bağımsız değişken arasındaki durumu inceleyerek çalışmaya başlar. Üyelik sınıfı bilinmeyen bir  $X=\{x_1, \dots, x_n\}$  veri setinde her bir  $(x_i)$  özelliğinin, C1 ve C2 gibi sınıflardan hangisine ait olduğunun olasılığını hesaplayan Bayes teoremi aşağıdaki Denklem 3'de ifade edilmiştir (Ünsal ve Kaya, 2020, s. 1194):

$$P(C_1|x_i) = \frac{P(x_i|C_1)P(C_1)}{P(x_i|C_1)P(C_1) + P(x_i|C_2)P(C_2)}$$

Denklem 3

C1 ve C2 olarak gösterilen iki ayrı hipotezin, başka bir deyişle iki ayrı sınıfın bulunduğu kabul edilmektedir.  $P(C_1 | x_i)$   $x_i$ 'nin C1 sınıfında olma olasılığını ifade etmektedir.  $P(x_i)$ ,  $x_i$  değerinin veritabanındaki bulunma sıklığı/sayıdır. Eğer  $(m)$  adet sınıf olduğu kabul edilirse bu durumda fonksiyon aşağıdaki Denklem 4'teki gibi olacaktır:

$$P(X_i) = \sum_{j=1}^m P(X_i|C_j)P(C_j)$$

Denklem 4

Bu modelde sınıf  $(m)$ , algoritmada tahmin üretilirken kullanılacak olan kriterleri temsil etmektedir. Sınıflandırma sırasında sınıflandırmaya tabi tutulacak veriler, test veri seti olarak kabul edilir. Sisteme daha önceden girilmiş veriler ise eğitim veri setini oluşturmaktadır. Eğitim veri setinde her bir örnek, yapılacak olan sınıflandırma sonucuna göre AL-SAT gibi iki duruma sahip olacaktır. Test verisinin sahip olduğu nitelikler ile eğitim veri setine göre hangi sınıfa gireceği Naive Bayes algoritması yardımı ile saptanmaktadır. Algoritmada, test veri

setinin başarılı olması durumu, test verisinin her bir özelliğinin, eğitim veri setinde (AL) sınıfına giren veri setlerinin özellikleri ile aynı olanlarının sayısının tüm veri seti sayısına bölünmesi ile olasılık hesabı yapılarak çarpımları sonucu elde edilen orana göre belirlenmektedir. Her bir kritere ait  $AL(p)$  ihtimali hesabı, test verisinde bulunan kritere ait özelliğinin/değerinin eğitim verisindeki tüm örnekler içinde bu özellikle eşit ve (AL) sınıfına giren örneklerin sayısının tüm (AL) örnek sayısına bölünmesi ile hesaplanır.  $SAT(q)$  ihtimali hesabı yapılırken de test verisinde bulunan kritere ait özelliğinin/değerinin eğitim verisindeki örnekler içinde bu özelliğe eşit olan ve (SAT) sınıfa giren örneklerin sayısı tüm (SAT) örnek sayısına bölünür. Son olarak test verisinin (AL) sınıfa girme olasılığı; tüm kriterlerin olumlu olasılıklarının birbirlerine ve (AL) örnek sayısının toplam örnek sayısına bölünmesiyle elde edilen değerlerin çarpılmasıyla hesaplanır. (SAT) sınıfa girme olasılığı hesab edilirken ise tüm kriterlerin olumsuz olasılıklarının birbirlerine ve (SAT) örnek sayısının toplam örnek sayısına bölünmesi sonucu ortaya çıkan değer ile çarpılması işlemi yapılır.  $AL(p)$  olasılığı ve  $SAT(q)$  olasılığı hesaplanarak eğer  $AL(p)$  olasılığı  $SAT(q)$  olasılığından daha büyük ise test verisi için (AL) sınıfına gireceği tahmini yapılmaktadır (Ünsal ve Kaya, 2020, s. 1195).

### 3. BÖLÜM

## BİST 30 ENDEKSTE SMA GENETİK ALGORİTMA UYGULAMASI İLE ALGORİTMİK TİCARET UYGULAMASI

### 3.1. Araştırmanın Yöntemi

Bu çalışmada BİST 30 endekste hisse senetlerinin, teknik analiz yöntemlerinden bir tanesi olan Basit Ağırlıklı Ortalama (SMA) yöntemini algoritmik olarak uygulayarak meydana gelen Al ve SAT sinyallerine göre oluşan kısa pozisyon ve uzun pozisyon yatırım stratejilerinin dönüşümlü getirilerinin değerlendirilmesi amaçlanmıştır. SMA, sermaye piyasası yatırımlarında en fazla kullanılan teknik analiz yöntemlerinden bir tanesi olması sebebiyle bu çalışmada SMA algoritması kullanılarak gerçekleştirilmiştir.

Basit hareketli Ortalama'da (Simple Moving Average-SMA) belirli bir tarih aralığında önceki kapanış fiyatlarının aritmetik ortalaması hesaplanmaktadır. N günlük SMA hesaplayabilmek için son N günlük kapanış fiyatlarının aritmetik ortalaması alınmaktadır. Farklı dönemsel değerlere sahip iki farklı SMA'nın kesişme noktalarından "Al-Sat" sinyalleri üretilebilmektedir (Irmak, 2019, s. 32).

En çok kullanılmakta olan hareketli ortalama çeşidi olan SMA, bir menkul kıymetin belirli bir zaman dilimi içindeki ortalama fiyatıdır. Kaç günlük hareketli ortalama hesaplanması arzu ediliyorsa, menkul kıymetin o kadar günlük değeri toplanır ve yine aynı sayıya bölünür. Örnek olarak, 40 günlük SMA hesaplanması isteniyorsa, menkul kıymetin 40 günlük kapanış değerleri toplanarak 40'a bölünür. SMA, basit ortalama olarak hesaplandığı için, ortalamanın içerisinde yer alan her noktanın eşit ağırlığı bulunduğu varsayılır. Bu yöntem uygulanırken günler ilerledikçe, en eski fiyat toplamdan düşer ve yeni gelen günün fiyatı toplama ilave edilerek yeni bir ortalama hesaplanır (Bahadır, 2008, s. 16).

Basit Hareketli Ortalama (SMA), bir hisse senedinin belli bir zaman dilimi içindeki kapanış fiyatlarının ortalamasını temsil eder. Kısa dönem ve uzun dönem SMA grafikleri beraber kullanılarak trend belirlenmesi gerçekleştirilebilir. Bu saptama kısa dönem ve uzun dönem grafiklerinin kesiştiği noktalardan yararlanılarak yapılmaktadır. SMA genelde 5, 10, 50, 100, 200 günlük zaman dilimleri için hesaplanmaktadır. Zaman diliminin kısa olması SMA eğrisinin, hisse senedinin fiyatına daha çabuk tepki vermesine neden olurken, zaman diliminin uzun oluşu daha yumuşak bir eğri oluşmasını mümkün kılar (Bayram, 2014, ss. 5-6):

$$SMA = \frac{\sum_{i=d-N}^d P_i}{N}$$

veya

$$SMA = \frac{P_1 + P_2 + \dots + P_N}{N}$$

Formülde gösterilen ( $\pi$ ) değeri hisse senedinin  $i$  günündeki kapanış fiyatını, ( $d$ ) o zamanki günü, ( $N$ ) ise SMA'nın kaç günlük hesaplandığını göstermektedir. Başka bir ifadeyle, o günden itibaren  $N$  gün öncesine kadar olan kapanış fiyatlarının ortalamaları hesaplanır.

SMA'daki hesaplanmış gün değeri ne kadar fazla olur ise, SMA çizgisi fiyat değişimlerine karşı o kadar az hassas olmaktadır. Bu nedenle ortalama gün sayısını arttırmak, uzun dönemdeki trendin gücünün ölçülebilmesi için uygulanabilecek en iyi metodlardan bir tanesidir. SMA için yapılan en önemli eleştiri, ortalama içerisindeki tüm günlerin eşit ağırlık taşımalarıdır. Genel kanaat ise, son günlerin ağırlığının daha yüksek olması gerektiğidir. Bu eleştiri, yeni hareketli ortalamaların hesaplanmasına neden olmuştur (Bahadır, 2008, s. 17).

Bu çalışma kapsamında algoritmik ticaret yaparken BİST 30'daki tüm hisseleri alıyormuş gibi basit ağırlıklı ortalama algoritması oluşturularak uygulanmıştır. Tarih aralığı olarak 31.05.2010 / 29.05.2020 seçilmiştir. Bunun sebebi 27 Temmuz 2020 tarihinde endeksten 00 atılmıştır. Bu sebeple 29.05.2020 tarihine kadar seçilmiştir. Çalışma için 2019 yılı içerisinde BİST 30 (XU030) (Başlangıç, 27.12.1996) endeksinde listede olan 30 adet hisse senedinin gün sonu kapanış fiyatları kullanılmıştır. Her bir hisse senedinin gün sonu kapanış fiyatlarından oluşturulan veri seti Genetik Algoritma ve karesel programlamada kullanılmıştır. Her iki yöntem için oluşturulan algoritmalar MATLAB (R2015a) programı kullanılarak çözümlenmeye ve analize tabi tutulmuştur.

Bu çalışma kapsamında, Basit Ağırlıklı Ortalama (SMA) Genetik Algoritma stratejisi uygulanırken crossoverda kısa basit hareketli ortalama bir de uzun basit hareketli ortalama esas alınarak ticaret yapılmıştır. Bu algoritmik ticaret stratejisinde kısa pozisyon uzun pozisyonu geçince uzun pozisyona geçilmiştir. Kısa pozisyon uzun pozisyonun altına inince ise açığa satış pozisyonuna geçilmiştir. Basit ağırlıklı ortalama algoritmik ticaret stratejisi uygulanırken plan dâhilinde ticaret yapılmış olup; birinci planda açığa satış yapılmakta, ikinci planda ise açığa satış yapılmamaktadır.

**Tablo 3.1.** Kısa ve Uzun Pozisyon Senaryoları

	<b>Kısa Pozisyon</b>	<b>Uzun Pozisyon</b>
1. Senaryo	3 gün	8 gün
2. Senaryo	3 gün	21 gün
3. Senaryo	3 gün	34 gün
4. Senaryo	8 gün	21 gün
5. Senaryo	8 gün	34 gün
6. Senaryo	8 gün	55 gün
7. Senaryo	21 gün	34 gün
8. senaryo	21 gün	55 gün
9. senaryo	21 gün	89 gün

Çalışmada SMA algortirması kapsamında kısa ve uzun pozisyon karşılaştırmalı uzun pozisyon alma veya açığa satma stratejileri kapsamında aşağıdaki Tablo 4.1.'de yer alan 9 senaryo uygulanmıştır.

Başlangıçtaki sinyalde eldeki hisse senedi miktarı ile belirlenen kısa ve uzun pozisyonlardaki alınan sinyallere göre pozisyonlarınb yeniden belirlenmesi sonucunda ilk yatırım tutarında artış olup olmadığı ve bu artışın tutarının tespit edilmesi amaçlanmıştır. Zira hisse senedi yatımcısı bakış açısıyla önemli olan hisse senedi miktarını/tutarını AL ve SAT işlemleri ile arttırabilmektedir. Bu nedenle elde edilen getirilerin alternatif yatırım araçları ile karşılaştırılmasına ihtiyaç duyulmamıştır.

Çalışma kapsamında, örneğin Tablo 4.1.'deki senaryo bir için SMA algoritması tekniğine göre, her gün için, o gün de dâhil olmak üzere (örneğin senaryo 1 için) kendisinden önceki 3 ve 8 günün SMA değeri hesaplanmış ve o günün SMA3 / SMA8 değerleri şeklinde işaretlenmiştir. SMA yöntemi kullanıldığında, önceki yöntemler gibi ertesi güne bakılmamış; bunun yerine, o günkü SMA değerleri ile kapanış değerleri mukayese edilmiştir. SMA yönteminde kullanılan yorum teknikleri kapsamında, SMA'in kapanış değerinden yüksek ya da alçak olduğu durumlar gözlemlenmiştir. Algoritmik ticaret kapsamında, kısa pozisyon (3 gün) uzun pozisyonu (8 günü) geçtiğinde uzun pozisyonu alınmıştır. Uzun pozisyon (8 gün) kısa pozisyonu (3 günü) geçtiğinde pozisyon kapatılarak açığa satış yapılmıştır. Diğer senaryolarda da aynı strateji uygulanmıştır.

Algoritmalar işledikleri girdi verilerine bağlı olarak zaman içerisinde kendilerini sürece uyumlu bir biçimde çalışabilecek forma getirebilmekte ve bu mekanizma ise algoritmaların öğrenme yeteneğini olarak tanımlanmaktadır. Algoritma öğrenmesi algoritmanın sürekli bir şekilde kendisini geliştirmesine imkân tanımaktadır. Bununla birlikte algoritmaların her zaman daha doğru sonuçlara ulaşabilme arzusu ise ister istemez rekabet karşıtı durumların da oluşmasına sebep olabilmektedir (Yıldız, 2020, s. 25).

Algoritmanın öğrenme işlemleri veri setinin eğitim verileri ve test verileri olarak bölünmesiyle başlatılmaktadır. Hazırlanan veri seti birden fazla durum karşısında farklı sonuçlar alabilmek ve gerçeğe yakın testler yapabilmek için farklı senaryolarda sınanmıştır. İlk durumda, veri setinde bulunan verinin yüzde yetmiş eğitim verisi, yüzde otuzu ise test verisi seti oluşturacak biçimde belirlenmiştir.

Bu çalışmada algoritmik ticaret kapsamında verilerin eğitilmesi için "Fibonacci" sayılarından faydalanılmıştır. Bunun sebebi de borsada teknik analizlerde çok kullanılmasıdır.

Fibonacci'nin "Fibonacci sayı dizisini" bir soruyu çözerken bulduğu bilinmektedir. Fibonacci sayı dizisi; her bir terimi kendinden önce gelen ilk iki terimin toplanmasıyla oluşturularak belirlenen "1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, ....." sayı dizisidir. Bu dizinin ilginç bir boyutu, 5. terimden sonra gelen ardışık terimlerin oranlarının altın orana çok yakın değerde olmaları, 12. terim olan 144'ten sonra gelen tüm ardışık terim oranlarının ise sürekli olarak

1,61803 olarak hesaplanmıştır. Fibonacci sayı dizisi ile altın oran özdeşleşmektedir (Baykut ve Kıvanç, 2004, s. 3).

Fibonacci sayı dizisinin terimleri  $F_1, F_2 \dots F_n$  şeklinde gösterildiğinde;  $F_1=1, F_2=1$  olmak üzere,  $F_3$ 'ten itibaren dizinin terimleri;  $F_n=F_{n-1}+F_{n-2}$  biçiminde olacaktır. Bu formül ile dizinin tüm terimleri hesaplanabilmektedir. Fibonacci sayı dizisinin asıl ilginç özelliklerden biride her bir terimin kendisinden önce gelen terime bölünmesiyle bulunan oran "altın oran" veya "ilahi oran" (1,618) olarak bilinen sayıya yaklaşmasıdır (Yaşar, 2016, s. 33).

Bu kapsamda çalışmada Fibonacci sayı dizisi kullanılarak veriler eğitime tabi tutulmuştur. 3 günlük kısa hareketli ağırlıklı ortalama fiyatlar seçildiğinde bu kapsamda veriler 8 günlük, 21 günlük ve 34 günlük olarak eğitilmiştir. Veriler ön incelemede 5 günlük ve 13 günlük eğitimlerde benzer sonuçları elde edilmesi sebebiyle 5 günlük ve 13 günlük veri eğitimi kullanılmamıştır. Oluşturulan SMA Algoritmik modelinin eğitimi sonucunda fiyat tahmini yüksek bir başarıyla gerçekleştirilmiştir. Son aşamada ise yine aynı veri seti kullanılarak eğitim verileri ve test verileri SMA algoritması için yeniden dizayn edilmiş ve tekrar eğitilmiştir. SMA algoritması eğitimlerin sonunda hangi özelliklerin fiyatların tahmini için kullanılması gerektiğini kendisi tespit ederek ve yüksek doğruluk tahminiyle başarılı bir biçimde öğrenme sürecini tamamlamıştır.

BİST 30'endekse başlangıçta 1.000.000 TL yatırım yapıldığı varsayılmıştır. Bu tutar endeksin ilk sinyal verdiği zamandaki fiyatına bölünerek 1.000.000 TL ile alınabilecek endeks miktarı belirlenmiştir. Alım satımlarda komisyon oranı %0,1 olarak uygulanmıştır. Her sinyalde bu başlangıç endeks miktarı ile hisse senedin fiyatı çarpılarak o andaki Over All hesaplanmıştır. Bu sinyallerdeki hisse senedi miktarı ile ilk sinyalde tespit edilen miktar getirileri hesaplanmış ve tüm sinyallerdeki getirilerin ortalaması en son bitiş tarihindeki toplam getiri ve toplam yatırım tutarına ulaşılmıştır.

Tablolarda başlangıç yatırımı, yatırımın ulaştığı son nokta, getirisi, yıllık getirisi, değişkenliği, getirinin pik yaptığı nokta, günlük getiri, Buy Hold (altüst stratejisi) getirisi, değişkenlik, Sharpe performans oranı, Sortino performans oranı, maksimum düşme, ortalama düşme, maksimum düşme süresi, ortalama düşme süresi gösterilmiştir. Aşağıda algoritmik ticarete basit ağırlıklı ortalama yöntemin kullanılması sonucunda BİST 30 endeks portföyünün performansını ölçebilmek için kullanılan 3 performans ölçüsü kısaca açıklanmıştır:

#### **a) Sharpe Performans Oranı**

Sharpe (1966) tarafından geliştirilen bu oran, riskin ölçüsü olarak standart sapma değişkeninin esas alan performans ölçüm yaklaşımıdır. Sharpe oranı portföyün getirisi ve riski arasındaki ilişkiyi ölçmek için geliştirilmiştir Risk ölçüsü olarak standart sapmanın alınması beklenen getiriden hem aşağı yönlü hem de yukarı yönlü sapmaların risk olarak kabul edildiği manasına gelmektedir. Bir portföye ait Sharpe oranı ne kadar yüksekse o portföyün performansının o kadar iyi olduğu neticesine varılmaktadır. Sharpe performans oranı, risksiz

faiz oranını aşan portföy getirisinin portföyün standart sapmasına bölünmesi yoluyla kolayca hesaplanan bir orandır. Sharpe oranı Denklem 5’de gösterilmiştir (Büberkökü, 2021, s. 342):

$$S_p = \frac{R_p - r_f}{\sigma_p} = \frac{\sum_{i=1}^n x_i(r_i - r_f)}{\sqrt{\sum_{i=1}^n \sum_{j=1}^n x_i x_j \sigma_{ij}}}$$

Denklem 5

Sp: Portföyün Sharpe performans oranı,

Rp: Portföyün beklenen getirisi,

ri: i .ninci hisse senedinin getirisi,

rf: Risksiz faiz oranı,

σp: Portföyün riski (Portföyün Standart Sapması)

Portföye ait Sharpe performans oranının yüksek olması o portföyün tercih edilme ihtimalini de yükseltmektedir.

#### **b) Sortino Performans Ölçütü**

Sortino Oranı, Sharpe oranının yeni bir varyasyonu olarak üretilmiştir. Sortino oranı hesaplanırken, Sharpe oranındaki standart sapma değişkeni yerine, kısmi standart sapmayı (semi-variance) kullanarak, portföye ait getirilerin varyansının yönünü dikkate alınmaktadır. Bu şekilde pozitif standart sapmalar ile negatif standart sapmalar ayırt edilebilmektedir (Karacabey ve Gökgöz, 2005, s. 33).

Sortino oranı, portföy getirisine ait toplam standart sapmadan çok, hisse senedi portföyünün negatif getirilerinin standart sapmasını (aşağı yönlü sapma) kullanarak zararlı oynaklığı toplam oynaklıktan ayırmakta ve bu açıdan Sharpe oranından farklılaşmaktadır. Sortino oranı, bir portföyün getirisinden risksiz faiz oranı düşüldükten sonra bu toplamı portföyün aşağı yönlü getiri sapmasına bölerek hesaplamaktadır. Sharpe oranı gibi, daha yüksek Sortino oranı da daha iyi bir performans anlamına gelmektedir. Rasyonel bir yatırımcı, iki özdeş yatırımı mukayese ederken daha yüksek Sortino oranına sahip olanı seçme eğiliminde olacaktır. Çünkü Sortino oranı, yatırımın aldığı kötü riskten birim başına daha fazla getiri elde edildiği manasına gelmektedir (Sortino ve Price, 1994, ss. 59-64).

Sortino oranında payda kısmında portföyün standart sapması yerine, risksiz faiz oranının altında kalan portföy getirilerinin standart sapmasının alınması Sharpe oranıyla arasındaki farklılıktır (Korkmaz ve Uygurtürk, 2008, s. 119). Sortino oranında sadece aşağı yönde sapmalar kullanılır. Bu biçimde getiri dağılımlarının asimetric olma problemine çözüm bulunmuş olmaktadır. Sortino oranı aşağıdaki Denklem 6’da gösterilmiştir (Teker vd, 2008, s. 94):

$$\text{Sortino Oranı} = \frac{r_p - r_{\text{expmin}}}{\sigma_d} = \frac{r_p - r_{\text{expmin}}}{\sigma - r_{\text{expmin}}} \quad \text{Denklem 6}$$

$r_p$ : Portföy getirisi,

$r_{\text{expmin}}$ : Yatırımcının kabul ettiği minimum getiri oranı,

$\sigma_d$ : Kısmi (semi-variance) standart sapma.

Eşitlikte,  $r_p$ , i portföyün ortalama getiri oranını,  $\sigma_{\text{expmin}}$  risksiz faiz oranının altında kalan portföy getirilerinin standart sapmasını (düşüş, aşağı yönlü standart sapma), yani kısmi standart sapmayı göstermektedir. Dolayısıyla  $\sigma_d$  sadece risksiz faiz oranının altında kalan portföy getirileri için hesaplanmaktadır. Sortino performans ölçütünde elde edilen değerlerin pozitif olması veya büyük olması portföy performansının iyi olduğu manasına gelmektedir (Gökgöz, 2006, s. 82).

### c) Calmar Oranı

Calmar oranı, portföyün en yüksek değerde satın alınarak en düşük değere ulaştığında satıldığı varsayımıyla yatırımcının yüzleşebileceği en büyük kaybı ölçen riske göre ayarlanmış bir performans ölçütüdür (Carles vd., 2019, s. 241). Young (1991) tarafından geliştirilen Calmar oranı risk ölçüsü olarak MDD değerini kullanmakta ve MDD başına düşen yıllık ortalama getiri oranını performans ölçüsü olarak ele almaktadır. Calmar oranı Denklem 7'de gösterilmiştir:

$$\text{Calmar Oranı} = \frac{r_{\text{mean}}}{MDD} \quad \text{Denklem 7}$$

$r_{\text{mean}}$ : Günlük veriler kullanılarak hesaplanan ortalama getiri oranını,

MDD: Günlük veriler kullanılarak hesaplanan MDD değerini

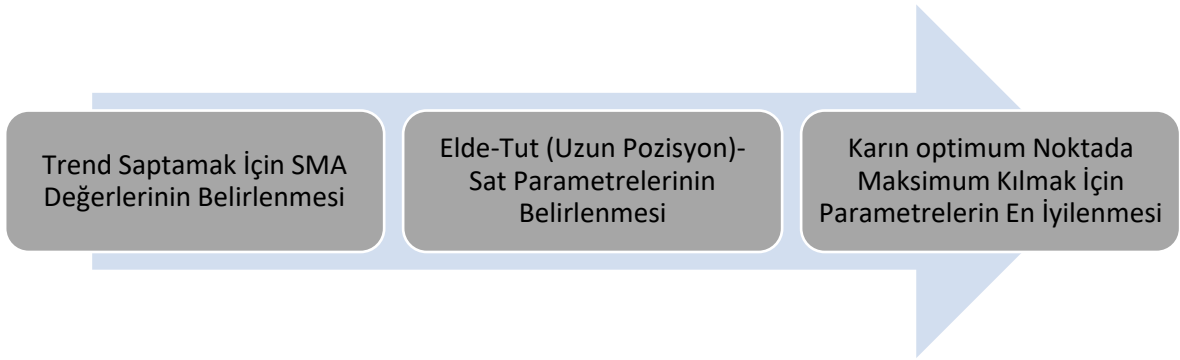
(MDD: Maximum drawdown)

Modelin uygulanmasında MDD Risk Ölçütü esas alınmıştır. Portföy risk ölçütlerinden bir tanesi de hisse senetlerinin belli dönemlerdeki önemli düşüşlerini veya en düşük noktasını dikkate alan drawdown risk ölçüsüdür. Drawdown risk ölçüsünün en çok kullanılan türü ise Maximum Drawdown (MDD) ölçüsüdür. MDD risk ölçüsü, belli bir zaman aralığında bir hisse senedi endeksinin veya portföyünün en düşük noktasını belirlemekte ve portföyün görece riskliliğini, oluşan en düşük fiyat noktalarına göre hesaplanmaktadır. Hisse senetlerinin göreceli olarak en düşük noktalarına göre riski ölçen MDD ölçüsü, volatilitenin (fiyat oynaklığının) artan bir fonksiyonudur ve buna göre yüksek volatilitelere yaşanan dönemlerde yüksek düşüş noktalarının yaşanmasını beklemek olağan bir durumdur (Vecer vd., 2006, ss. 1-3).

Ortalama varyans portföylerinde yer alan varyans ve standart sapma risk ölçülerine göre, beklenen getirilerin normal dağılımı gibi bir niteliğe dayalı olmaması, bilhassa piyasaların aşırı kötü dönemlerine göre ölçümlenmeler yapması ve bu ölçümlenmelere göre yatırımcıya portföy çeşitlendirmesi yapabilmesini sağlaması, MDD ölçüsünün önemli niteliklerindedir. Başka bir ifadeyle MDD, belirli bir dönemde menkul kıymetlerin fiyat serilerinde ulaşılan en yüksek nokta ile en dip nokta arasındaki kaybı göstermektedir (Ismail vd., 2004, s. 149).

MDD, portföy değerinin yeni bir zirve noktasına ulaşmasından önceki zirve noktasından dip noktasına doğru gerçekleşen maksimum fiyat düşüş oranı şeklinde ifade edilmektedir. MDD esasen portföy yönetimi açısından aşağı yönlü riskleri dikkate alan yeni bir risk ölçüsü sağlamaktadır. Bu risk ölçüsü standart sapma ve/veya varyans gibi risk ölçülerine göre bazı önemli avantajlara sahiptir. MDD standart sapma ve/veya varyans gibi yatırım süreci içinde gerçekleşen ortalama risk seviyesine değil, yatırım süreci içine gerçekleşen uç durumların yol açtığı aşağı yönlü risk seviyesine odaklanmaktadır. Bu çeşit bir yaklaşım, yatırımcıların karşılaşılabileceği uç finansal şoklardan en az zararlı çıkabilmelerini sağlamak bakımından önemli avantajlar sağlayabilmektedir (Büberkökü, 2021, ss. 335-340).

Bu çalışmada BİST 30 endeks pay alım-satım stratejisinde kârı optimize edebilmek üzere 3 aşamalı bir model oluşturulmuştur. İlk aşamada, SMA algoritmasını kullanarak trend tespiti yapılmıştır. İkinci aşamada al-sat sinyalleri oluşturabilmek için saptanan strateji parametreleri optimize edilmiştir. Son aşamada da korunmalı alım stratejisi kullanılarak kârı maksimize etmek amaçlanmıştır. Şekil 4.1’de üç aşamalı modeli şematik olarak gösterilmiştir:



**Şekil 3.1.** Genetik Algoritma Araştırma Modeli

#### **a) Birinci Aşama - Trendin Belirlenmesi**

Bu aşamada SMA verilerinin trendinin belirlenmesi amaçlanmıştır. Bu trendi saptayabilmek için SMA'nın zaman fiyat grafiğinde maksimum ve minimum noktalarının işaretlenmesi gerekmektedir. Maksimum ve minimum noktaların belirlenmesinin en kesin yöntemlerinden bir tanesi "pencere kaydırma" yöntemidir. Pencere kaydırma yöntemi; zaman fiyat grafiği üzerinde işaretlenmiş (3gün, 8gün vb.) günlük aralıklar ilk günden son güne doğru sırayla

seçilerek, bu aralıktaki maksimum ve minimum noktalarının yerel uç noktalar olarak seçilmesi şeklinde yürütülmektedir. Fakat bu yöntem var olan değerler üzerinde gerçekleştirilebildiğinden, (3 gün, 8 gün vb.) günlük aralıkların tamamının bilinmesi gerektiğinden test sırasında kullanılamaz. Çünkü test esnasında gelecek verilerin henüz bilinmediği varsayılmaktadır.

Gelecek veriler bilinemediği varsayılarak bu safhada pencere kaydırma yöntemiyle beraber SMA teknik analiz yönteminden faydalanılmıştır. SMA ile trendi saptamak için kısa dönem ve uzun dönem SMA değerleri hesaplanmış ve bu değerlerin grafiklerinin kesiştikleri noktalar trend değişim noktaları olarak kabul edilmiştir. Kısa dönemli SMA'nın uzun dönemli SMA grafiğini yukarı yönde kestiği noktada trendin yükselen trend, aksi durumda ise (uzun dönemli SMA çizgisinin kısa dönemli SMA çizgisini yukarı yönde kestiği durumda) trendin alçalan trend olduğu kabul edilmektedir. SMA teknik analiz algoritması ve pencere kaydırma yöntemi birlikte kullanılarak, ikisinin açıklarını gidermeye yönelik ortak bir uygulama yapılmıştır. Verilerin eğitilmesi sırasında pencere kaydırma yöntemi ile en büyük ve en küçük noktalar, SMA yöntemi ile de kısa pozisyon (dönem) ve uzun pozisyon (dönem) kesişim noktaları tespit edilmiştir. Bu iki yöntemin benzerliği uygunluk fonksiyonu olarak hesaplanmıştır. Benzerlik fonksiyonu hesaplanmasında, iki yöntemin oluşturduğu trend değişim noktalarının sayısının benzerliği ve bu noktalardaki tarihlerin birbirine yakınlığı kullanılmaktadır. Bu iki değerın çarpımından elde edilen sonuç, iki yöntem arasındaki benzerlik değerini vermektedir.

Üç aşamalı bu modelde en iyi SMA değerlerinin tespit edilmesi, trendin daha iyi tahmin edilmesine yardımcı olarak daha fazla trend değişim noktası oluşmasını mümkün kılar, bu da Korunmalı Alım stratejisinde alımların ve satımların yapılacağı noktaların daha iyi tahmin edilebilmesi anlamına gelir.

Bu aşamada optimizasyon algoritmaları yerine tüm değerlerin sınanması yoluyla sonuçlar bulunmaktadır. Zira çözüm uzayı bu aşama için küçük bir ölçüde olduğu için, kısa dönemde neticeye ulaşılabilmektedir. Diğer taraftan, optimizasyon algoritmaları her zaman en doğru sonucu ulaşmakta, ancak aramanın sonucunda bulabildiği uygunluğu en yüksek olan sonucu göstermektedir. Buna göre, en doğru sonucu bulabilmek için tüm değerlerin denenmesi yöntemi tercih edilmiştir.

## **b) İkinci Aşama**

Ticaret stratejisi parametrelerini hesaplayabilmek için temelde iki değişkene gereksinim duyulmaktadır. Bu iki değişken, "kaç günlük süre için hesaplanacağı" ve "eşik değerleri"dir. Bu çalışmada alım-satım için ve yükselen-alçalan trendler için ayrı ayrı toplamda dokuz senaryo için parametreler oluşturulmuş ve optimizasyonu gerçekleştirilmiştir. Bu aşamada optimizasyon algoritmalarında kullanılacak uygunluk fonksiyonu olarak gerçekleştirilen

işlemlerin sonucunda ulaşılan kâr kullanılmış ve uygunluk fonksiyonu en yüksek değerine kavuşturulmaya çalışılmıştır.

### **c) Üçüncü Aşama - Parametrelerin Optimize Edilmesi**

Son aşamada uygunluk fonksiyonu olarak yapılan ticaret işlemlerinden korunmalı alım stratejisi kullanılarak elde edilen kâr kullanılmış, korunmalı alım stratejisi ile yapılacak işlemlerin en çok getiriye sağlayacak olanların seçimi temin edilmiştir.

## **3.2. Bulgular ve Tartışma**

### **3.2.1. Verilerin öğrenmesinin analizi bulguları**

Algoritmanın uygulanmaya başlanması süreci veri setinin “eğitim verisi” ve “test verisi” olmak üzere ikiye ayrılması ile başlamaktadır. Literatürde net bir kural olmamakla beraber verinin %70’i rastgele seçilerek eğitim verisi olarak atanmaktadır. Eğitim verisi kullanmanın amacı kullanılan çeşitli makine öğrenmesi algoritmalarına ait kontrol parametrelerinin en iyi değerlerinin belirlenmesi, algoritmanın öğrenme sürecinin test edilmesidir. Optimal parametre değerleri seçimi yapılırken gerçek gözlem değeriyle modelin hesapladığı sonucun (tahmin çıktısı) arasındaki fark olarak adlandırılan toplam hata değerinin en aza indirgenmesi esas alınır. Buna göre, en küçük hatayı veren parametre değerleri seçilir. Eğitim aşamasından sonra geliştirilen modelin gerçek hayattaki başarısını ifade eden genelleme gücü test verisi yardımıyla ölçülür. Ancak, test verisinin yüksek başarı göstermesi yeterli değildir, aynı zamanda makine öğrenmesi modelinin eğitim ve test verilerinde gösterdiği performans değerlerinin de birbirine yakın olması beklenir. Başarılı bir makine öğrenmesi algoritmasından arzu edilen, eğitim verileri yardımıyla öğrendiği süreci o alanındaki tüm verilere genelleme yapabilmesidir. Bu, modelin daha önce hiç karşılaşmadığı veya ilk kez gördüğü veriler üzerinde başarılı tahminlerin yapılmasına imkân sağlar. Eğer eğitim veri setinde yüksek performans göstermiş fakat test veri setinde gösterdiği başarı oranı düşük ise bu durumda aşırı öğrenme (ezberleme) olarak nitelendirilir. Aşırı öğrenme bir makine öğrenmesi algoritmasının eğitim verisinin çok iyi öğrenmesi fakat modelin ilk kez karşılaştığı bir veriyi (test verisi) kullanarak elde ettiği tahminlerin başarısız veya düşük performans oranına sahip olması halidir. Herhangi bir algoritma modelinin performansı doğrudan doğruya seçilen kontrol parametrelerine bağlıdır. Doğru parametrenin seçilmesi modelin performansını yükseltmektedir. Her algoritma öğrenmesi modeli farklı sayıda kontrol parametrelerine sahiptir ve bunların eş zamanlı olarak en iyilerinin saptanması gerekir (Ustalı vd., 2021, s. 10).

**Tablo 3.2.** Kısa 3 gün, uzun 34 günlük Senaryo İçin Algoritma Öğrenmesi

```
self.ma1 = self.I(SMA, price, 3)
self.ma2 = self.I(SMA, price, 34)

def next(self):
    if crossover(self.ma1, self.ma2):
        self.buy()
    elif crossover(self.ma2, self.ma1):
        self.sell()

bt = Backtest(df, SmaCross, commission=.001, cash = 1000000,
              exclusive_orders=True)
stats = bt.run()
stats
```

Start	2020-08-04 00:00:00
End	2022-07-29 00:00:00
Duration	724 days 00:00:00
Exposure Time [%]	92.827869
Equity Final [\$]	943885.0074
Equity Peak [\$]	1823291.163873
Return [%]	-5.611499
Buy & Hold Return [%]	126.029632
Return (Ann.) [%]	-2.938191
Volatility (Ann.) [%]	24.874959
Sharpe Ratio	0.0
Sortino Ratio	0.0
Calmar Ratio	0.0
Max. Drawdown [%]	-50.137453
Avg. Drawdown [%]	-5.888558
Max. Drawdown Duration	285 days 00:00:00
Avg. Drawdown Duration	39 days 00:00:00
# Trades	30
Win Rate [%]	33.333333

Tablo 4.2.'deki algoritmaya bakıldığında daha sonraki normal veriler üzerinden gerçekleştirilen algoritmalarındaki verilere göre aşırı olmadıkları ve normal bir öğrenme olduğu görülmektedir.

**Tablo 3.3.** Kısa 3 Gün, Uzun 8 Günlük Birinci Senaryo Uygulanmaksızın 3-21 Günlük Senaryonun Algoritma Öğrenmesi

```
self.ma1 = self.I(SMA, price, 3)
self.ma2 = self.I(SMA, price, 21)

def next(self):
    if crossover(self.ma1, self.ma2) :
        self.buy()
    elif crossover(self.ma1, self.ma2):
        if len(self.trades) > 0:
            self.trades[0].close()

t = Backtest(df, smacross_without_short, commission=.001, cash = 1000000,
            exclusive_orders=True)
tats = bt.run()
tats
```

```
start                2020-08-04 00:00:00
end                  2022-07-29 00:00:00
duration             724 days 00:00:00
exposure Time [%]   93.647541
equity Final [$]    2138628.589538
equity Peak [$]     2333993.228236
return [%]          113.862859
Buy & Hold Return [%] 126.029632
return (Ann.) [%]   48.074317
volatility (Ann.) [%] 39.475522
Sharpe Ratio        1.217826
Sortino Ratio        2.515247
Calmar Ratio         2.224583
Max. Drawdown [%]  -21.610482
Avg. Drawdown [%]  -3.682573
Max. Drawdown Duration 296 days 00:00:00
Avg. Drawdown Duration 24 days 00:00:00
Trades              15
Win Rate [%]        66.666667
```

Tablo 3'teki veriler incelendiğinde bu algoritma öğrenmesinin daha sonraki algoritma uygulamalarındakilere göre daha yüksek değerler olduğu ve bu öğrenmenin aşırı bir öğrenme olduğu görülmektedir.

### 3.2.2. Birinci senaryo bulguları

**Tablo 3.4.** Kısa Pozisyon 3 Günlük SMA, Uzun Pozisyon 8 Günlük SMA Genetik Algoritma Sonuçları

```
class SmaCross(Strategy):
    def init(self):
        price = self.data.Close
        self.ma1 = self.I(SMA, price, 3)
        self.ma2 = self.I(SMA, price, 8)

    def next(self):
        if crossover(self.ma1, self.ma2):
            self.buy()
        elif crossover(self.ma2, self.ma1):
            self.sell()

bt = Backtest(df, SmaCross, commission=.001, cash = 1000000,
             exclusive_orders=True)
stats = bt.run()
stats
```

Start	2010-05-31 00:00:00
End	2020-05-29 00:00:00
Duration	3651 days 00:00:00
Exposure Time [%]	99.602069
Equity Final [\$]	1567545.65618
Equity Peak [\$]	2097378.722953
Return [%]	56.754566
Buy & Hold Return [%]	80.295238
Return (Ann.) [%]	4.61077
Volatility (Ann.) [%]	24.121728
Sharpe Ratio	0.191146
Sortino Ratio	0.296237
Calmar Ratio	0.117673
Max. Drawdown [%]	-39.182764
Avg. Drawdown [%]	-6.328013
Max. Drawdown Duration	1801 days 00:00:00
Avg. Drawdown Duration	112 days 00:00:00
# Trades	337
Win Rate [%]	37.68546

Kısa pozisyon için 3 gün, uzun pozisyon için 8 gün esas alınarak yapılan genetik algoritma sonuçlarına göre, başta yatırılan 1.000.000 TL 2.097.378 TL'sına ulaşmış ancak en son pozisyonda 1.567.545 TL'sından kapatmış 567.545 TL kar bırakmıştır. Buna göre ortalama getiri oranı %56,75 olarak gerçekleşmiştir. Yıllık ortalama yıllık getiri % 4,6 olmuştur. Yıllık volatilité %24,12 olarak gerçekleşmiştir. BİST 30 endeks portföyü için performans ölçüsü olarak Sharpe oranı 0,19, Sortino oranı 0,29 ve Calmar oranı ise 0,11 olmuştur. Maksimum aşağı yönlü hareket -%39, maksimum aşağı yönlü hareket süresi 1081 gün ve ortalama aşağı yönlü hareket süresi ise 112 gün olarak gerçekleşmiş, toplamda 337 gün ticaret yapılmış ve kazanma yüzdesi %37,68 olmuştur. Kazanma yüzdesi örneğin, bir kişi toplam 100 kez işlem yaptıysa ve bu işlemlerden 37'sini kazandıysa, kazanma oranı %37'dir. Kazanma oranı

formülü her zaman kazanan işlem sayısının toplam işlem sayısına bölümüdür. Yatırımcılar kazanma oranlarını hesaplarlarken stratejilerinin işe yarayıp yaramadığını belirleyebilirler.

### 3.2.3. İkinci senaryo bulguları

**Tablo 3.5.** Kısa Pozisyon 3 Günlük SMA, Uzun Pozisyon 21 Günlük SMA Genetik Algoritma Sonuçları

```
class SmaCross(Strategy):
    def init(self):
        price = self.data.Close
        self.ma1 = self.I(SMA, price, 3)
        self.ma2 = self.I(SMA, price, 21)

    def next(self):
        if crossover(self.ma1, self.ma2):
            self.buy()
        elif crossover(self.ma2, self.ma1):
            self.sell()

bt = Backtest(df, SmaCross, commission=.001, cash = 1000000,
              exclusive_orders=True)
stats = bt.run()
stats
```

Start	2010-05-31 00:00:00
End	2020-05-29 00:00:00
Duration	3651 days 00:00:00
Exposure Time [%]	99.084759
Equity Final [\$]	1173026.222664
Equity Peak [\$]	1669610.4545
Return [%]	17.302622
Buy & Hold Return [%]	80.295238
Return (Ann.) [%]	1.613188
Volatility (Ann.) [%]	22.206019
Sharpe Ratio	0.072646
Sortino Ratio	0.110385
Calmar Ratio	0.029727
Max. Drawdown [%]	-54.266265
Avg. Drawdown [%]	-6.433902
Max. Drawdown Duration	2515 days 00:00:00
Avg. Drawdown Duration	112 days 00:00:00
# Trades	174
Win Rate [%]	32.183908

Kısa pozisyon için 3 gün, uzun pozisyon için 21 gün esas alınarak yapılan genetik algoritma sonuçlarına göre, başta yatırılan 1.000.000 TL 1.669.610 TL'sına ulaşmış ancak en son pozisyonda 1.173.026 TL'sından kapatmış 173.026 TL kar bırakmıştır. Buna göre ortalama getiri oranı %17,30 olarak gerçekleşmiştir. Yıllık ortalama yıllık getiri % 1,6 olmuştur. Yıllık

volatility %22,20 olarak gerçekleşmiştir. BİST 30 endeks portföyü için performans ölçüsü olarak Sharpe oranı 0,07, Sortino oranı 0,11 ve Calmar oranı ise 0,02 olmuştur. Maksimum aşağı yönlü hareket -%54,26, maksimum aşağı yönlü hareket süresi 2515 gün ve ortalama aşağı yönlü hareket süresi ise 112 gün olarak gerçekleşmiş, toplamda 174 gün ticaret yapılmış ve kazanma yüzdesi %32,18 olmuştur.

Ancak 3 gün kısa 8 gün uzun pozisyon senaryosu hiç uygulanmadan bu senaryo uygulanacak olursa sonuçlar aşağıdaki gibi değişmektedir.

**Tablo 3.6.** Kısa Pozisyon 3 Günlük SMA, Uzun Pozisyon 21 Günlük SMA Genetik Algoritma Sonuçları (Kısa pozisyon birinci senaryo olmaksızın)

```

price = self.data.Close
self.ma1 = self.I(SMA, price, 3)
self.ma2 = self.I(SMA, price, 21)

def next(self):
    if crossover(self.ma1, self.ma2) :
        self.buy()
    elif crossover(self.ma1, self.ma2):
        if len(self.trades) > 0:
            self.trades[0].close()

bt = Backtest(df, smacross_without_short, commission=.001, cash = 1000000,
              exclusive_orders=True)
stats = bt.run()
stats

```

Start	2010-05-31 00:00:00
End	2020-05-29 00:00:00
Duration	3651 days 00:00:00
Exposure Time [%]	98.925587
Equity Final [\$]	1575184.54868
Equity Peak [\$]	1953661.687852
Return [%]	57.518455
Buy & Hold Return [%]	80.295238
Return (Ann.) [%]	4.661779
Volatility (Ann.) [%]	24.279793
Sharpe Ratio	0.192002
Sortino Ratio	0.282951
Calmar Ratio	0.134126
Max. Drawdown [%]	-34.756611
Avg. Drawdown [%]	-4.418063
Max. Drawdown Duration	1462 days 00:00:00
Avg. Drawdown Duration	87 days 00:00:00
# Trades	87
Win Rate [%]	48.275862

Kısa pozisyon için 3 gün, uzun pozisyon için 21 gün esas alınarak yapılan genetic algoritma sonuçlarına göre, başta yatırılan 1.000.000 TL 1.9853.661 TL'sına ulaşmış ancak en son pozisyonda 1.575.184 TL'sından kapatmış 575.184 TL kar bırakmıştır. Buna göre ortalama getiri oranı %57,51 olarak gerçekleşmiştir. Yıllık ortalama yıllık getiri % 4,66 olmuştur. Yıllık volatiliti %24,27 olarak gerçekleşmiştir. BİST 30 endeks portföyü için performans ölçüsü olarak Sharpe oranı 0,19, Sortino oranı 0,2829 ve Calmar oranı ise 0,1341 olmuştur. Maksimum aşağı yönlü hareket -%34, maksimum aşağı yönlü hareket süresi 1462 gün ve ortalama aşağı yönlü hareket süresi ise 87 gün olarak gerçekleşmiş, toplamda 87 gün ticaret yapılmış ve kazanma yüzdesi %37,68 olmuştur. Kazanma yüzdesi ise %48,27 olmuştur.

Bu sonuçlara göre ilk (3 gün kısa, 8 gün uzun) senaryo uygulanmadan ikinci senaryo uygulandığında algoritmik ticaretteki getiri değerlerinin yükseldiği gözlenmektedir.

### 3.2.4. Üçüncü senaryo bulguları

**Tablo 3.7.** Kısa Pozisyon 3 Günlük SMA, Uzun Pozisyon 34 Günlük SMA Genetik Algoritma Sonuçları

```
class SmaCross(Strategy):
    def init(self):
        price = self.data.Close
        self.ma1 = self.I(SMA, price, 3)
        self.ma2 = self.I(SMA, price, 34)

    def next(self):
        if crossover(self.ma1, self.ma2):
            self.buy()
        elif crossover(self.ma2, self.ma1):
            self.sell()

bt = Backtest(df, SmaCross, commission=.001, cash = 1000000,
              exclusive_orders=True)
stats = bt.run()
stats
```

Start	2010-05-31 00:00:00
End	2020-05-29 00:00:00
Duration	3651 days 00:00:00
Exposure Time [%]	97.811381
Equity Final [\$]	2607408.905656
Equity Peak [\$]	2762132.951531
Return [%]	160.740891
Buy & Hold Return [%]	80.295238
Return (Ann.) [%]	10.087206
Volatility (Ann.) [%]	24.106815
Sharpe Ratio	0.418438
Sortino Ratio	0.702295
Calmar Ratio	0.338609
Max. Drawdown [%]	-29.790163
Avg. Drawdown [%]	-4.147976
Max. Drawdown Duration	1217 days 00:00:00
Avg. Drawdown Duration	57 days 00:00:00
# Trades	122
Win Rate [%]	38.52459

Kısa pozisyon için 3 gün, uzun pozisyon için 34 gün esas alınarak yapılan genetic algoritma sonuçlarına göre, başta yatırılan 1.000.000 TL 2.762.132 TL'sına ulaşmış ancak en son pozisyonda 2.607.408 TL'sından kapatmış 1.607.408 TL kar bırakmıştır. Buna göre ortalama getiri oranı %160,74 olarak gerçekleşmiştir. Yıllık ortalama yıllık getiri % 10,8 olmuştur. Yıllık volatiliti %24,10 olarak gerçekleşmiştir. BİST 30 endeks portföyü için performans ölçüsü olarak Sharpe oranı 0,41, Sortino oranı 0,70 ve Calmar oranı ise 0,33 olmuştur. Maksimum aşağı yönlü hareket -%29,79, maksimum aşağı yönlü hareket süresi 1217 gün ve ortalama aşağı yönlü hareket süresi ise 57 gün olarak gerçekleşmiş, toplamda 122 gün ticaret yapmış ve kazanma yüzdesi %38,52 olmuştur.

Ancak 3 gün kısa 8 gün uzun pozisyon senaryosu hiç uygulanmadan bu senaryo uygulanacak olursa sonuçlar aşağıdaki gibi değişmektedir.

**Tablo 3.8.** Kısa Pozisyon 3 Günlük SMA, Uzun Pozisyon 34 Günlük SMA Genetik Algoritma Sonuçları (Kısa pozisyon birinci senaryo olmadan)

```

self.ma1 = self.I(SMA, price, 3)
self.ma2 = self.I(SMA, price, 34)

def next(self):
    if crossover(self.ma1, self.ma2) :
        self.buy()
    elif crossover(self.ma1, self.ma2):
        if len(self.trades) > 0:
            self.trades[0].close()

bt = Backtest(df, smacross_without_short, commission=.001, cash = 1000000,
              exclusive_orders=True)
stats = bt.run()
stats

```

Start	2010-05-31 00:00:00
End	2020-05-29 00:00:00
Duration	3651 days 00:00:00
Exposure Time [%]	97.731795
Equity Final [\$]	1512077.687191
Equity Peak [\$]	1848236.433973
Return [%]	51.207769
Buy & Hold Return [%]	80.295238
Return (Ann.) [%]	4.233526
Volatility (Ann.) [%]	24.205371
Sharpe Ratio	0.1749
Sortino Ratio	0.256477
Calmar Ratio	0.118462
Max. Drawdown [%]	-35.737541
Avg. Drawdown [%]	-4.584729
Max. Drawdown Duration	1462 days 00:00:00
Avg. Drawdown Duration	93 days 00:00:00
# Trades	61
Win Rate [%]	47.540984

Kısa pozisyon için 3 gün, uzun pozisyon için 34 gün esas alınarak yapılan genetic algoritma sonuçlarına göre, başta yatırılan 1.000.000 TL 1.848.236 TL'sına ulaşmış ancak en son pozisyonda 1.512.077 TL'sından kapatmış 512.077 TL kar bırakmıştır. Buna göre ortalama getiri oranı %160,74 olarak gerçekleşmiştir. Yıllık ortalama yıllık getiri % 51,2 olmuştur. Yıllık volatilitiy %24,20 olarak gerçekleşmiştir. BİST 30 endeks portföyü için performans ölçüsü olarak Sharpe oranı 0,17, Sortino oranı 0,25 ve Calmar oranı ise 0,11 olmuştur. Maksimum aşağı yönlü hareket -%35,73, maksimum aşağı yönlü hareket süresi 1462 gün ve ortalama aşağı yönlü hareket süresi ise 93 gün olarak gerçekleşmiş, toplamda 61 gün ticaret yapılmış ve kazanma yüzdesi %47,54 olmuştur.

Bu sonuçlara göre birinci senaryonun uygulanmadığı durumda üçüncü senaryodaki bulgularda getiri oranlarında ve performansta düşme gerçekleştiği gözlenmektedir.

### 3.2.5. Dördüncü senaryo bulguları

**Tablo 3.9.** Kısa Pozisyon 8 Günlük SMA, Uzun Pozisyon 21 Günlük SMA Genetik Algoritma Sonuçları

```
def init(self):
    price = self.data.Close
    self.ma1 = self.I(SMA, price, 8)
    self.ma2 = self.I(SMA, price, 21
)

def next(self):
    if crossover(self.ma1, self.ma2):
        self.buy()
    elif crossover(self.ma2, self.ma1):
        self.sell()

bt = Backtest(df, SmaCross, commission=.001, cash = 1000000,
              exclusive_orders=True)
stats = bt.run()
stats
```

Start	2010-05-31 00:00:00
End	2020-05-29 00:00:00
Duration	3651 days 00:00:00
Exposure Time [%]	99.005173
Equity Final [\$]	898468.75768
Equity Peak [\$]	1420394.113219
Return [%]	-10.153124
Buy & Hold Return [%]	80.295238
Return (Ann.) [%]	-1.067873
Volatility (Ann.) [%]	21.50965
Sharpe Ratio	0.0
Sortino Ratio	0.0
Calmar Ratio	0.0
Max. Drawdown [%]	-59.196264
Avg. Drawdown [%]	-9.930606
Max. Drawdown Duration	1628 days 00:00:00
Avg. Drawdown Duration	190 days 00:00:00
# Trades	138
Win Rate [%]	39.855072

Kısa pozisyon için 8 gün, uzun pozisyon için 21 gün esas alınarak yapılan genetic algoritma sonuçlarına göre, başta yatırılan 1.000.000 TL 1.420.394 TL'sına ulaşmış ancak en son pozisyonda 898.468 TL'sından kapatmış 101.532 TL zarar bırakmıştır. Buna göre ortalama getiri oranı -%1,06 olarak gerçekleşmiştir. Yıllık ortalama yıllık getiri % 10,8 olmuştur. Yıllık volatiliti %21,50 olarak gerçekleşmiştir. BİST 30 endeks portföyü için performans ölçüsü olarak Sharpe oranı 0 Sortino oranı 0 ve Calmar oranı ise 0 olmuştur. Maksimum aşağı yönlü hareket -%59,19, maksimum aşağı yönlü hareket süresi 1628 gün ve ortalama aşağı yönlü hareket süresi ise 190 gün olarak gerçekleşmiş, toplamda 138 gün ticaret yapılmış ve yapılan işlemlerden kazanma yüzdesi %39,52 olmuştur.

Ancak 3 gün kısa 8 gün uzun pozisyon senaryosu hiç uygulanmadan bu senaryo uygulanacak olursa sonuçlar aşağıdaki gibi değişmektedir.

**Tablo 3.10.** Kısa Pozisyon 8 Günlük SMA, Uzun Pozisyon 21 Günlük SMA Genetik Algoritma Sonuçları (Kısa pozisyon birinci senaryo olmadan)

```

self.ma1 = self.I(SMA, price, 8)
self.ma2 = self.I(SMA, price, 21)

def next(self):
    if crossover(self.ma1, self.ma2) :
        self.buy()
    elif crossover(self.ma1, self.ma2):
        if len(self.trades) > 0:
            self.trades[0].close()

bt = Backtest(df, smacross_without_short, commission=.001, cash = 1000000,
              exclusive_orders=True)
stats = bt.run()
stats

```

Start	2010-05-31 00:00:00
End	2020-05-29 00:00:00
Duration	3651 days 00:00:00
Exposure Time [%]	98.766415
Equity Final [\$]	1567326.361711
Equity Peak [\$]	1906770.475
Return [%]	56.732636
Buy & Hold Return [%]	80.295238
Return (Ann.) [%]	4.609302
Volatility (Ann.) [%]	24.102864
Sharpe Ratio	0.191235
Sortino Ratio	0.281491
Calmar Ratio	0.13076
Max. Drawdown [%]	-35.25021
Avg. Drawdown [%]	-4.380167
Max. Drawdown Duration	1449 days 00:00:00
Avg. Drawdown Duration	87 days 00:00:00
# Trades	69
Win Rate [%]	44.927536
Loss Rate [%]	55.072464

Kısa pozisyon için 8 gün, uzun pozisyon için 21 gün esas alınarak yapılan genetic algoritma sonuçlarına göre, başta yatırılan 1.000.000 TL 1.906.770 TL'sına ulaşmış ancak en son pozisyonda 1.567.326 TL'sından kapatmış 567.326 TL kar bırakmıştır. Buna göre ortalama getiri oranı %56,73 olarak gerçekleşmiştir. Yıllık ortalama yıllık getiri % 4,6 olmuştur. Yıllık volatiliti %24,10 olarak gerçekleşmiştir. BİST 30 endeks portföyü için performans ölçüsü olarak Sharpe oranı 0,19 Sortino oranı 0,28 ve Calmar oranı ise 0,13 olmuştur. Maksimum aşağı yönlü hareket -%35,25, maksimum aşağı yönlü hareket süresi 1449 gün ve ortalama aşağı yönlü hareket süresi ise 87 gün olarak gerçekleşmiş, toplamda 69 gün ticaret yapılmış ve yapılan işlemlerden kazanma yüzdesi %44,92 olmuştur.

Buna göre birinci senaryonun uygulanmadığı durumda bu senaryonun getirilerinin ve performansının yükseldiği gözlenmektedir.

### 3.2.6. Beşinci senaryo bulguları

**Tablo 3.11.** Kısa Pozisyon 8 Günlük SMA, Uzun Pozisyon 34 Günlük SMA Genetik Algoritma Sonuçları

```

self.ma1 = self.I(SMA, price, 8)
self.ma2 = self.I(SMA, price, 34)

def next(self):
    if crossover(self.ma1, self.ma2):
        self.buy()
    elif crossover(self.ma2, self.ma1):
        self.sell()

bt = Backtest(df, SmaCross, commission=.001, cash = 1000000,
             exclusive_orders=True)
stats = bt.run()
stats

```

Start	2010-05-31 00:00:00
End	2020-05-29 00:00:00
Duration	3651 days 00:00:00
Exposure Time [%]	97.652209
Equity Final [\$]	1550497.406047
Equity Peak [\$]	2007376.09068
Return [%]	55.049741
Buy & Hold Return [%]	80.295238
Return (Ann.) [%]	4.496119
Volatility (Ann.) [%]	22.672419
Sharpe Ratio	0.198308
Sortino Ratio	0.304768
Calmar Ratio	0.104748
Max. Drawdown [%]	-42.922991
Avg. Drawdown [%]	-4.653231
Max. Drawdown Duration	786 days 00:00:00
Avg. Drawdown Duration	68 days 00:00:00
# Trades	82
Win Rate [%]	35.365854

Kısa pozisyon için 8 gün, uzun pozisyon için 34 gün esas alınarak yapılan genetic algoritma sonuçlarına göre, başta yatırılan 1.000.000 TL 2.007.0736 TL'sına ulaşmış ancak en son pozisyonda 1.550.497 TL'sından kapatmış 550.497 TL kar bırakmıştır. Buna göre ortalama getiri oranı %55,04 olarak gerçekleşmiştir. Yıllık ortalama yıllık getiri % 4,49 olmuştur. Yıllık volatiliti %22,67 olarak gerçekleşmiştir. BİST 30 endeks portföyü için performans ölçüsü olarak Sharpe oranı 0,19 Sortino oranı 0,30 ve Calmar oranı ise 0,10 olmuştur. Maksimum aşağı yönlü hareket -%42,92, maksimum aşağı yönlü hareket süresi 786 gün ve ortalama aşağı yönlü hareket süresi ise 68 gün olarak gerçekleşmiş, toplamda 82 gün ticaret yapılmış ve yapılan işlemlerden kazanma yüzdesi %35,36 olmuştur.

Ancak 3 gün kısa 8 gün uzun pozisyon senaryosu hiç uygulanmadan bu senaryo uygulanacak olursa sonuçlar aşağıdaki gibi değişmektedir.

**Tablo 3.12.** Kısa Pozisyon 8 Günlük SMA, Uzun Pozisyon 34 Günlük SMA Genetik Algoritma Sonuçları (Kısa pozisyon birinci senaryo olmadan)

```

self.ma1 = self.I(SMA, price, 8)
self.ma2 = self.I(SMA, price, 34)

def next(self):
    if crossover(self.ma1, self.ma2) :
        self.buy()
    elif crossover(self.ma1, self.ma2):
        if len(self.trades) > 0:
            self.trades[0].close()

bt = Backtest(df, smacross_without_short, commission=.001, cash = 1000000,
              exclusive_orders=True)
stats = bt.run()
stats

```

Start	2010-05-31 00:00:00
End	2020-05-29 00:00:00
Duration	3651 days 00:00:00
Exposure Time [%]	97.294071
Equity Final [\$]	1488354.343547
Equity Peak [\$]	1819295.265328
Return [%]	48.835434
Buy & Hold Return [%]	80.295238
Return (Ann.) [%]	4.068367
Volatility (Ann.) [%]	23.812626
Sharpe Ratio	0.170849
Sortino Ratio	0.249984
Calmar Ratio	0.118367
Max. Drawdown [%]	-34.370793
Avg. Drawdown [%]	-5.139699
Max. Drawdown Duration	1448 days 00:00:00
Avg. Drawdown Duration	90 days 00:00:00
# Trades	41
Win Rate [%]	48.780488

Kısa pozisyon için 8 gün, uzun pozisyon için 34 gün esas alınarak yapılan genetic algoritma sonuçlarına göre, başta yatırılan 1.000.000 TL 1.819.295 TL'sına ulaşmış ancak en son pozisyonda 1.488.354 TL'sından kapatmış 488.354 TL kar bırakmıştır. Buna göre ortalama getiri oranı %48,83 olarak gerçekleşmiştir. Yıllık ortalama yıllık getiri % 4,06 olmuştur. Yıllık volatiliti %23,81 olarak gerçekleşmiştir. BİST 30 endeks portföyü için performans ölçüsü olarak Sharpe oranı 0,17 Sortino oranı 0,24 ve Calmar oranı ise 0,11 olmuştur. Maksimum aşağı yönlü hareket -%34,37, maksimum aşağı yönlü hareket süresi 1448 gün ve ortalama aşağı yönlü hareket süresi ise 90 gün olarak gerçekleşmiş, toplamda 41 gün ticaret yapılmış ve yapılan işlemlerden kazanma yüzdesi %48,78 olmuştur.

Buna göre birinci senaryonun uygulanmadığı durumda bu senaryonun getirisi yükselmekle birlikte performansının düştüğü gözlenmektedir.

### 3.2.7. Altıncı senaryo bulguları

**Tablo 3.13.** Kısa Pozisyon 8 Günlük SMA, Uzun Pozisyon 55 Günlük SMA Genetik Algoritma Sonuçları

```

self.ma1 = self.I(SMA, price, 8)
self.ma2 = self.I(SMA, price, 55)

def next(self):
    if crossover(self.ma1, self.ma2):
        self.buy()
    elif crossover(self.ma2, self.ma1):
        self.sell()

bt = Backtest(df, SmaCross, commission=.001, cash = 1000000,
              exclusive_orders=True)
stats = bt.run()
stats

```

Start	2010-05-31 00:00:00
End	2020-05-29 00:00:00
Duration	3651 days 00:00:00
Exposure Time [%]	95.105452
Equity Final [\$]	1174307.00325
Equity Peak [\$]	1772195.407344
Return [%]	17.4307
Buy & Hold Return [%]	80.295238
Return (Ann.) [%]	1.624308
Volatility (Ann.) [%]	21.44721
Sharpe Ratio	0.075735
Sortino Ratio	0.11122
Calmar Ratio	0.034046
Max. Drawdown [%]	-47.709638
Avg. Drawdown [%]	-5.65609
Max. Drawdown Duration	2135 days 00:00:00
Avg. Drawdown Duration	110 days 00:00:00
# Trades	56
Win Rate [%]	39.285714

Kısa pozisyon için 8 gün, uzun pozisyon için 55 gün esas alınarak yapılan genetic algoritma sonuçlarına göre, başta yatırılan 1.000.000 TL 1.772.195 TL'sına ulaşmış ancak en son pozisyonda 1.174.307 TL'sından kapatmış 174.307 TL kar bırakmıştır. Buna göre ortalama getiri oranı %17,43 olarak gerçekleşmiştir. Yıllık ortalama yıllık getiri % 1,6 olmuştur. Yıllık volatiliti %21,44 olarak gerçekleşmiştir. BİST 30 endeks portföyü için performans ölçüsü olarak Sharpe oranı 0,07 Sortino oranı 0,11 ve Calmar oranı ise 0,03 olmuştur. Maksimum aşağı yönlü hareket -%47,70, maksimum aşağı yönlü hareket süresi 2135 gün ve ortalama aşağı yönlü hareket süresi ise 110 gün olarak gerçekleşmiş, toplamda 82 gün ticaret yapılmış ve yapılan işlemlerden kazanma yüzdesi %39,28 olmuştur.

Ancak 3 gün kısa 8 gün uzun pozisyon senaryosu hiç uygulanmadan bu senaryo uygulanacak olursa sonuçlar aşağıdaki gibi değişmektedir.

**Tablo 3.14.** Kısa Pozisyon 8 Günlük SMA, Uzun Pozisyon 55 Günlük SMA Genetik Algoritma Sonuçları (Kısa pozisyon birinci senaryo olmaksızın)

```

price = self.reset_data
self.ma1 = self.I(SMA, price, 8)
self.ma2 = self.I(SMA, price, 55)

def next(self):
    if crossover(self.ma1, self.ma2) :
        self.buy()
    elif crossover(self.ma1, self.ma2):
        if len(self.trades) > 0:
            self.trades[0].close()

bt = Backtest(df, smacross_without_short, commission=.001, cash = 1000000,
              exclusive_orders=True)
stats = bt.run()
stats

```

Start	2010-05-31 00:00:00
End	2020-05-29 00:00:00
Duration	3651 days 00:00:00
Exposure Time [%]	91.683247
Equity Final [\$]	1494186.735906
Equity Peak [\$]	1825128.440125
Return [%]	49.418674
Buy & Hold Return [%]	80.295238
Return (Ann.) [%]	4.10919
Volatility (Ann.) [%]	23.092545
Sharpe Ratio	0.177944
Sortino Ratio	0.26061
Calmar Ratio	0.119314
Max. Drawdown [%]	-34.440087
Avg. Drawdown [%]	-5.228317
Max. Drawdown Duration	1448 days 00:00:00
Avg. Drawdown Duration	87 days 00:00:00
# Trades	28
Win Rate [%]	46.428571

Kısa pozisyon için 8 gün, uzun pozisyon için 55 gün esas alınarak yapılan genetic algoritma sonuçlarına göre, başta yatırılan 1.000.000 TL 1.825.128 TL'sına ulaşmış ancak en son pozisyonda 1.494.186 TL'sından kapatmış 494.186 TL kar bırakmıştır. Buna göre ortalama getiri oranı %49,41 olarak gerçekleşmiştir. Yıllık ortalama yıllık getiri % 4,1 olmuştur. Yıllık volatiliti %23,09 olarak gerçekleşmiştir. BİST 30 endeks portföyü için performans ölçüsü olarak Sharpe oranı 0,17 Sortino oranı 0,26 ve Calmar oranı ise 0,11 olmuştur. Maksimum aşağı yönlü hareket -%34,44, maksimum aşağı yönlü hareket süresi 1448 gün ve ortalama aşağı yönlü hareket süresi ise 87 gün olarak gerçekleşmiş, toplamda 28 gün ticaret yapılmış ve yapılan işlemlerden kazanma yüzdesi %46,42 olmuştur.

Buna göre birinci senaryo hiç uygulanmadan bu senaryo uygulandığında getirinin, performansın ve kazanma yüzdesinin arttığı gözlenmektedir.

### 3.2.8. Yedinci senaryo bulguları

**Tablo 3.15.** Kısa Pozisyon 21 Günlük SMA, Uzun Pozisyon 34 Günlük SMA Genetik Algoritma Sonuçları

```
price = self.data.Close
self.ma1 = self.I(SMA, price, 21)
self.ma2 = self.I(SMA, price, 34)

def next(self):
    if crossover(self.ma1, self.ma2):
        self.buy()
    elif crossover(self.ma2, self.ma1):
        self.sell()

bt = Backtest(df, SmaCross, commission=.001, cash = 1000000,
              exclusive_orders=True)
stats = bt.run()
stats
```

Start	2010-05-31 00:00:00
End	2020-05-29 00:00:00
Duration	3651 days 00:00:00
Exposure Time [%]	97.373657
Equity Final [\$]	826583.158781
Equity Peak [\$]	1430290.825117
Return [%]	-17.341684
Buy & Hold Return [%]	80.295238
Return (Ann.) [%]	-1.891731
Volatility (Ann.) [%]	21.049454
Sharpe Ratio	0.0
Sortino Ratio	0.0
Calmar Ratio	0.0
Max. Drawdown [%]	-54.132622
Avg. Drawdown [%]	-5.283225
Max. Drawdown Duration	2682 days 00:00:00
Avg. Drawdown Duration	153 days 00:00:00
# Trades	80
Win Rate [%]	36.25

Kısa pozisyon için 21 gün, uzun pozisyon için 34 gün esas alınarak yapılan genetik algoritma sonuçlarına göre, başta yatırılan 1.000.000 TL 1.430.290 TL'sına ulaşmış ancak en son pozisyonda 826.583 TL'sından kapatmış 173.417 TL zarar bırakmıştır. Buna göre ortalama getiri oranı -%17,34 olarak gerçekleşmiştir. Yıllık ortalama yıllık getiri % -1,89 olmuştur. Yıllık volatiliti %21,04 olarak gerçekleşmiştir. BİST 30 endeks portföyü için performans ölçüsü olarak Sharpe oranı 0,0 Sortino oranı 0,0 ve Calmar oranı ise 0,0 olmuştur. Maksimum aşağı yönlü hareket -%54,13, maksimum aşağı yönlü hareket süresi 2682 gün ve ortalama aşağı yönlü hareket süresi ise 153 gün olarak gerçekleşmiş, toplamda 80 gün ticaret yapılmış ve yapılan işlemlerden kazanma yüzdesi %36,25 olmuştur.

Ancak 3 gün kısa 8 gün uzun pozisyon senaryosu hiç uygulanmadan bu senaryo uygulanacak olursa sonuçlar aşağıdaki gibi değişmektedir.

**Tablo 3.16.** Kısa Pozisyon 21 Günlük SMA, Uzun Pozisyon 34 Günlük SMA Genetik Algoritma Sonuçları (Kısa pozisyon birinci senaryo olmaksızın)

```
price = self.data.Close
self.ma1 = self.I(SMA, price, 21)
self.ma2 = self.I(SMA, price, 34)

def next(self):
    if crossover(self.ma1, self.ma2) :
        self.buy()
    elif crossover(self.ma1, self.ma2):
        if len(self.trades) > 0:
            self.trades[0].close()

bt = Backtest(df, smacross_without_short, commission=.001, cash = 1000000,
              exclusive_orders=True)
stats = bt.run()
stats
```

```
: Start                2010-05-31 00:00:00
End                    2020-05-29 00:00:00
Duration               3651 days 00:00:00
Exposure Time [%]     97.015519
Equity Final [$]      1453753.191398
Equity Peak [$]       1780472.863938
Return [%]            45.375319
Buy & Hold Return [%] 80.295238
Return (Ann.) [%]     3.823181
Volatility (Ann.) [%] 23.954341
Sharpe Ratio          0.159603
Sortino Ratio         0.232908
Calmar Ratio          0.10875
Max. Drawdown [%]    -35.155532
Avg. Drawdown [%]    -4.657376
Max. Drawdown Duration 1448 days 00:00:00
Avg. Drawdown Duration 94 days 00:00:00
# Trades              40
Win Rate [%]         55.0
```

Kısa pozisyon için 21 gün, uzun pozisyon için 34 gün esas alınarak yapılan genetic algoritma sonuçlarına göre, başta yatırılan 1.000.000 TL 1.780.472 TL'sına ulaşmış ancak en son pozisyonda 1.453.753 TL'sından kapatmış 453.753 TL kar bırakmıştır. Buna göre ortalama getiri oranı %45,37 olarak gerçekleşmiştir. Yıllık ortalama yıllık getiri % 3,8 olmuştur. Yıllık volatiliti %23,95 olarak gerçekleşmiştir. BİST 30 endeks portföyü için performans ölçüsü olarak Sharpe oranı 0,15 Sortino oranı 0,23 ve Calmar oranı ise 0,10 olmuştur. Maksimum aşağı yönlü hareket -%35,15, maksimum aşağı yönlü hareket süresi 1448 gün ve ortalama aşağı yönlü hareket süresi ise 94 gün olarak gerçekleşmiş, toplamda 40 gün ticaret yapılmış ve yapılan işlemlerden kazanma yüzdesi %55 olmuştur.

Buna göre ilk senaryo uygulanmadan sırasıyla bu senaryo uygulandığına getirinin, performansın ve kazanma yüzdesinin yükseldiği görülmektedir.

### 3.2.9. Sekizinci senaryo bulguları

**Tablo 3.17.** Kısa Pozisyon 21 Günlük SMA, Uzun Pozisyon 55 Günlük SMA Genetik Algoritma Sonuçları

```
price = self.data.Close
self.ma1 = self.I(SMA, price, 21)
self.ma2 = self.I(SMA, price, 55)

def next(self):
    if crossover(self.ma1, self.ma2):
        self.buy()
    elif crossover(self.ma2, self.ma1):
        self.sell()

bt = Backtest(df, SmaCross, commission=.001, cash = 1000000,
              exclusive_orders=True)
stats = bt.run()
stats
```

: Start	2010-05-31 00:00:00
End	2020-05-29 00:00:00
Duration	3651 days 00:00:00
Exposure Time [%]	94.787107
Equity Final [\$]	825053.697328
Equity Peak [\$]	1419204.124836
Return [%]	-17.49463
Buy & Hold Return [%]	80.295238
Return (Ann.) [%]	-1.90995
Volatility (Ann.) [%]	20.478598
Sharpe Ratio	0.0
Sortino Ratio	0.0
Calmar Ratio	0.0
Max. Drawdown [%]	-50.522611
Avg. Drawdown [%]	-8.751092
Max. Drawdown Duration	1004 days 00:00:00
Avg. Drawdown Duration	172 days 00:00:00
# Trades	48
Win Rate [%]	39.583333

Kısa pozisyon için 21 gün, uzun pozisyon için 55 gün esas alınarak yapılan genetik algoritma sonuçlarına göre, başta yatırılan 1.000.000 TL 1.419.204 TL'sına ulaşmış ancak en son pozisyonda 825.053 TL'sından kapatmış 174.947 TL zarar bırakmıştır. Buna göre ortalama getiri oranı -%17,49 olarak gerçekleşmiştir. Yıllık ortalama yıllık getiri % -1,90 olmuştur. Yıllık volatilitiy %20,47 olarak gerçekleşmiştir. BİST 30 endeks portföyü için performans ölçüsü olarak Sharpe oranı 0,0 Sortino oranı 0,0 ve Calmar oranı ise 0,0 olmuştur. Maksimum aşağı yönlü hareket -%50,52, maksimum aşağı yönlü hareket süresi 1004 gün ve ortalama aşağı yönlü hareket süresi ise 172 gün olarak gerçekleşmiş, toplamda 48 gün ticaret yapılmış ve yapılan işlemlerden kazanma yüzdesi %39,58 olmuştur.

Ancak 3 gün kısa 8 gün uzun pozisyon senaryosu hiç uygulanmadan bu senaryo uygulanacak olursa sonuçlar aşağıdaki gibi değişmektedir.

**Tablo 3.18.** Kısa Pozisyon 21 Günlük SMA, Uzun Pozisyon 55 Günlük SMA Genetik Algoritma Sonuçları (Kısa pozisyon birinci senaryo olmaksızın)

```

self.ma1 = self.I(SMA, price, 21)
self.ma2 = self.I(SMA, price, 55)

def next(self):
    if crossover(self.ma1, self.ma2) :
        self.buy()
    elif crossover(self.ma1, self.ma2):
        if len(self.trades) > 0:
            self.trades[0].close()

t = Backtest(df, smacross_without_short, commission=.001, cash = 1000000,
            exclusive_orders=True)
tats = bt.run()
tats

start                2010-05-31 00:00:00
end                  2020-05-29 00:00:00
duration              3651 days 00:00:00
exposure Time [%]    91.484282
equity Final [$]     1470779.051102
equity Peak [$]      1801709.795664
return [%]           47.077905
buy & Hold Return [%] 80.295238
return (Ann.) [%]    3.944476
volatility (Ann.) [%] 23.48061
Sharpe Ratio         0.167989
Sortino Ratio        0.245902
Calmar Ratio         0.110802
Max. Drawdown [%]   -35.599348
Avg. Drawdown [%]   -4.987421
Max. Drawdown Duration 1434 days 00:00:00
Avg. Drawdown Duration 80 days 00:00:00
# Trades             24
Win Rate [%]        54.166667

```

Kısa pozisyon için 21 gün, uzun pozisyon için 55 gün esas alınarak yapılan genetic algoritma sonuçlarına göre, başta yatırılan 1.000.000 TL 1.801.709 TL'sına ulaşmış ancak en son pozisyonda 1.470.779 TL'sından kapatmış 470.779 TL kar bırakmıştır. Buna göre ortalama getiri oranı %47,07 olarak gerçekleşmiştir. Yıllık ortalama yıllık getiri % 3,9 olmuştur. Yıllık volatility %23,48 olarak gerçekleşmiştir. BİST 30 endeks portföyü için performans ölçüsü olarak Sharpe oranı 0,16 Sortino oranı 0,24 ve Calmar oranı ise 0,11 olmuştur. Maksimum aşağı yönlü hareket -%35,59, maksimum aşağı yönlü hareket süresi 1434 gün ve ortalama aşağı yönlü hareket süresi ise 80 gün olarak gerçekleşmiş, toplamda 24 gün ticaret yapılmış ve yapılan işlemlerden kazanma yüzdesi %54,16 olmuştur.

Buna göre ilk senaryo uygulanmadan sırasıyla bu senaryo uygulandığına getirinin, performansın ve kazanma yüzdesinin yükseldiği görülmektedir.

### 3.2.10. Dokuzuncu senaryo bulguları

**Tablo 3.19.** Kısa Pozisyon 21 Günlük SMA, Uzun Pozisyon 89 Günlük SMA Genetik Algoritma Sonuçları

```

self.ma1 = self.I(SMA, price, 21)
self.ma2 = self.I(SMA, price, 89)

def next(self):
    if crossover(self.ma1, self.ma2):
        self.buy()
    elif crossover(self.ma2, self.ma1):
        self.sell()

bt = Backtest(df, SmaCross, commission=.001, cash = 1000000,
              exclusive_orders=True)
stats = bt.run()
stats

```

Start	2010-05-31 00:00:00
End	2020-05-29 00:00:00
Duration	3651 days 00:00:00
Exposure Time [%]	94.428969
Equity Final [\$]	422952.086164
Equity Peak [\$]	1122281.079438
Return [%]	-57.704791
Buy & Hold Return [%]	80.295238
Return (Ann.) [%]	-8.267122
Volatility (Ann.) [%]	19.130949
Sharpe Ratio	0.0
Sortino Ratio	0.0
Calmar Ratio	0.0
Max. Drawdown [%]	-64.40037
Avg. Drawdown [%]	-9.716569
Max. Drawdown Duration	3376 days 00:00:00
Avg. Drawdown Duration	431 days 00:00:00
# Trades	45
Win Rate [%]	28.888889

Kısa pozisyon için 21 gün, uzun pozisyon için 89 gün esas alınarak yapılan genetik algoritma sonuçlarına göre, başta yatırılan 1.000.000 TL 1.122.281 TL'sına ulaşmış ancak en son pozisyonda 422.952 TL'sından kapatmış 577.048 TL zarar bırakmıştır. Buna göre ortalama getiri oranı -%57,04 olarak gerçekleşmiştir. Yıllık ortalama yıllık getiri % -8,26 olmuştur. Yıllık volatiliti %19,13 olarak gerçekleşmiştir. BİST 30 endeks portföyü için performans ölçüsü olarak Sharpe oranı 0,0 Sortino oranı 0,0 ve Calmar oranı ise 0,0 olmuştur. Maksimum aşağı yönlü hareket -%64,40, maksimum aşağı yönlü hareket süresi 3376 gün ve ortalama aşağı yönlü hareket süresi ise 431 gün olarak gerçekleşmiş, toplamda 45 gün ticaret yapılmış ve yapılan işlemlerden kazanma yüzdesi %28,88 olmuştur.

Ancak 3 gün kısa 8 gün uzun pozisyon senaryosu hiç uygulanmadan bu senaryo uygulanacak olursa sonuçlar aşağıdaki gibi değişmektedir.

**Tablo 3.20.** Kısa Pozisyon 21 Günlük SMA, Uzun Pozisyon 89 Günlük SMA Genetik Algoritma Sonuçları (Kısa pozisyon birinci senaryo olmaksızın)

```

self.ma1 = self.I(SMA, price, 21)
self.ma2 = self.I(SMA, price, 89)

def next(self):
    if crossover(self.ma1, self.ma2) :
        self.buy()
    elif crossover(self.ma1, self.ma2):
        if len(self.trades) > 0:
            self.trades[0].close()

bt = Backtest(df, smacross_without_short, commission=.001, cash = 1000000
              exclusive_orders=True)
stats = bt.run()
stats

```

Start	2010-05-31 00:00:00
End	2020-05-29 00:00:00
Duration	3651 days 00:00:00
Exposure Time [%]	91.325109
Equity Final [\$]	1414419.078227
Equity Peak [\$]	1716494.546977
Return [%]	41.441908
Buy & Hold Return [%]	80.295238
Return (Ann.) [%]	3.537996
Volatility (Ann.) [%]	23.039624
Sharpe Ratio	0.153561
Sortino Ratio	0.223403
Calmar Ratio	0.099031
Max. Drawdown [%]	-35.726191
Avg. Drawdown [%]	-5.24238
Max. Drawdown Duration	1462 days 00:00:00
Avg. Drawdown Duration	86 days 00:00:00
# Trades	22
Win Rate [%]	50.0

Kısa pozisyon için 21 gün, uzun pozisyon için 89 gün esas alınarak yapılan genetic algoritma sonuçlarına göre, başta yatırılan 1.000.000 TL 1.716.494 TL'sına ulaşmış ancak en son pozisyonda 1.414.419 TL'sından kapatmış 414.419 TL kar bırakmıştır. Buna göre ortalama getiri oranı %41,44 olarak gerçekleşmiştir. Yıllık ortalama yıllık getiri % -3,53 olmuştur. Yıllık volatiliti %23,03 olarak gerçekleşmiştir. BİST 30 endeks portföyü için performans ölçüsü olarak Sharpe oranı 0,15 Sortino oranı 0,22 ve Calmar oranı ise 0,09 olmuştur. Maksimum aşağı yönlü hareket -%35,72, maksimum aşağı yönlü hareket süresi 1462 gün ve ortalama aşağı yönlü hareket süresi ise 86 gün olarak gerçekleşmiş, toplamda 22 gün ticaret yapılmış ve yapılan işlemlerden kazanma yüzdesi %50 olmuştur.

Buna göre ilk senaryo uygulanmadan sırasıyla bu senaryo uygulandığına getirinin, performansın ve kazanma yüzdesinin yükseldiği görülmektedir.

## SONUÇ VE ÖNERİLER

BİST'te hisse senetlerine yatırım yapılırken kullanılması gereken ticaret stratejilerinin önemi her geçen gün daha da artmakta ve yatırım yaparken rastgele alınacak bir yatırım kararı yatırımcıları büyük zararlara uğratabilmektedir. Yatırımcıların ve onlara yatırım tavsiyelerinde bulunan yatırım danışmanlarının portföye hisse senedi seçimlerinde teknik analiz ve temel analiz çok önemlidir. Teknik analiz yapılırken en çok kullanılan yöntemlerden biri olan, diğer teknik analizlerin de esasen temelini oluşturan ve yatırım danışmanlarının ve yorumcuların da teknik analiz yorumlarında sık sık kullandıkları basit hareketli ortalamalar (SMA) birçok yatırımcı ve analist tarafından da kullanılmaktadır. Bu çalışmada teknik analiz yöntemi olarak uygulanan SMA genetik algoritmik ticaret, yatırım yapmak isteyen yatırımcılar için hisse senetlerinin geçmişte sergilediği performans açısından güzel bir tahminci olma özelliği taşımaktadır.

Portföyün optimizasyonunu sağlayacak ve getirisini maksimum kılacak şekilde portföye hisse senedi seçimi yapabilmek yalnızca yüksek getirili menkul kıymetlerin portföye dâhil edilmesi işleminden ibaret olmayıp; zamanlama açısından en erken kararı aldırabilecek modeller portföy optimizasyonu açısından oldukça önemlidir. Genetik algoritmalar, optimal sonuca yakın sonuçlar vermekle beraber; zamanlama açısından da hızlıca çözüme ulaştırabilmektedir. Son dönemlerdeki portföy optimizasyonu ile ilgili literatür çalışmalarında, en çok kullanılan yöntem genetik algoritmalar olarak karşımıza gelmektedir. Eğitilebilen algoritmalar finansman alanındaki problemlerin çözümünde sıklıkla kullanılmaktadır. Bu problemlerin başında trend tahmini, fiyat tahmini, risk analizi, teknik analiz ve öznitelik seçimi konuları gelmektedir.

BİST 30 endeks için benzer veya farklı SMA genetik algoritmalar kullanılarak farklı oranlarda pozitif ve negatif getiriler elde edilebilmektedir. Elde edilen sonuçlar, BIST 30 endeks portföyünün alınması ve elde tutulmasına ilişkin olarak algoritmik ticaret işlemlerinin normal ticaret işlemlerine göre daha iyi getiriler ve performanslar sağlayabileceğini göstermektedir.

BİST 30 için kısa pozisyon 3 gün ve uzun pozisyon 8 günlük uygulanan algoritmik ticaret senaryosunun getirisi ve performansı çok iyi olmakla birlikte, bu senaryo analizden çıkarıldıktan sonra diğer senaryoların da getirilerinin ve performanslarının iyileştiği görülmüştür. Bu ilk senaryonun getirisinin ve performansının yüksek olmakla birlikte algoritmanın geçmişten öğrenerek geleceği tahmin etmesi konusunda 3-8 günlük bir sürenin algoritmanın öğrenmesi için kısa bir süre olmasından kaynaklandığı düşünülmektedir. Dolayısıyla bu çalışmanın sonuçlarına göre SMA algoritmasının en optimum ve verimli öğrenme ve öğrendiklerini gelecek ticaret verilerine uygulaması açısından makul sürenin 3-21 günlük süre olduğu sonucuna ulaşılmıştır.

Çalışma kapsamında SMA Genetik Algoritması uygulanarak yapılan dokuz adet algoritmik ticaret uygulamasının sonuçlarının özetleri aşağıda Tablo 4.21 ve Tablo 4.22'de gösterilmiştir:

**Tablo 4.1.** Kısa Pozisyonlu Senaryo Bulgularının Özeti

	1	2	3	4	5	6	7	8	9
	3-8	3-21	3-34	8-21	8-34	8-55	21-34	21-55	21-89
Son Birikimli yatırım Tutarı	1.567.54 5	1.173.02 6	<b>2.607.408</b>	898.468	1.550.49 7	1.174.30 7	826.583	825.053	422.952
Birikimli yatırım tutarının Tavan Noktası	2.097.37 8	1.669.61 0	<b>2.762.132</b>	1.420.39 4	2.007.37 6	1.772.19 5	1.430.29 0	1.419.20 4	<b>1.122.281</b>
Getiri Oranı	0,5675	0,173	1,607	-0,1015	0,5504	0,1743	-0,1734	-0,01749	-0,577
Buy Hold Return	0,8029	0,8029	0,8029	0,8029	0,8029	0,8029	0,8029	0,8029	0,8029
Yıllık Getiri	0,0461	0,0161	0,1008	-0,0106	0,0449	0,0162	-0,0189	-0,019	-0,0826
Volatility (Oynaklık)	0,2412	0,222	0,241	0,215	0,2267	0,2144	0,2104	0,2047	0,1913
Sharpe Oranı	0,1911	0,0726	0,4184	0	0,1983	0,0757	0	0	0
Sortino Oranı	0,2962	0,1103	0,7022	0	0,3047	0,1112	0	0	0
Calmar Oranı	0,1176	0,0297	0,3386	0	0,1047	0,034	0	0	0
Max. Drawdown (%)	-0,3918	-0,5426	-0,2979	-0,5919	-0,4292	-0,477	-0,5413	-0,5052	-0,644
Ortalama Drawdown	-0,06328	-0,0643	-0,0414	-0,0993	-0,0465	-0,0565	-0,0528	-0,0875	-0,0971
Max. Drawdown Süresi	1801	2515	1217	1628	786	2135	2682	1004	3376
Ortalama Drawdown Süresi	112	112	57	190	68	110	153	172	431
Ticaret sayısı	337	174	122	138	82	56	80	48	45
Kazanma Yüzdesi	0,3768	0,3218	0,3852	0,3985	0,3536	0,3928	0,3625	0,3958	0,2888
			EN YÜKSEK GETİRİ						EN DÜŞÜK GETİRİ

Tablo 21 incelendiğinde aşağıdaki bulgular gözlenmektedir:

- BİST 30 endekste anılan dönemde en iyi getiri ve performans kısa pozisyonun 3 gün ve uzun pozisyonun 34 gün olarak alındığı SMA algoritması ile uygulanan ticaret sonuçları olmuştur. Aynı zamanda bu senaryo fiyat oynaklığının en yüksek olduğu senaryodur.
- BİST 30 endekste anılan dönemde en kötü getiri ve performans kısa pozisyonun 21 gün ve uzun pozisyonun 89 gün olarak alındığı SMA algoritması ile uygulanan ticaret sonuçları olmuştur. Aynı zamanda bu senaryo fiyat oynaklığının en düşük olduğu senaryodur.
- Kısa 8 uzun 21 günlük SMA algoritmasının ticarete uygulandığı senaryoda negatif getiri ve performans sözkonusu olmuştur.
- Kısa 21 uzun 34 günlük SMA algoritmasının ticarete uygulandığı senaryoda negatif getiri ve performans sözkonusu olmuştur.

e) Kısa 21 uzun 55 günlük SMA algorirmasının ticarete uygulandığı senaryoda negatif getiri ve performans sözkonusu olmuştur.

**Tablo 4.2.** Kısa pozisyon (3-8) Birinci Senaryo Olmaksızın Senaryoların Bulgularının Özetleri

	1	2	3	4	5	6	7	8	9
	3-8,	3-21,	3-34,	8-21,	8-34,	8-55,	21-34,	21-55,	21-89,
Son Birikimli yatırım Tutarı		1.575.184	1.512.077	<b>1.567.326</b>	1.488.354	1.494.186	1.453.753	1.470.779	1.467.020
Birikimli yatırım tutarının Tavan Noktası		<b>1.953.661</b>	1.848.236	1.906.770	1.819.295	1.825.128	1.780.472	1.801.709	1.834.404
Getiri Oranı		<b>0,5751</b>	0,512	0,5673	0,4883	0,4941	0,4537	0,4707	0,467
Buy Hold Return		0,8029	0,8029	0,8029	0,8029	0,8029	0,8029	0,8029	0,8029
Yıllık Getiri		<b>0,4661</b>	0,04233	0,046	0,0406	0,041	0,0382	0,0394	0,0391
Volatility		<b>0,2427</b>	0,242	0,241	0,2381	0,2309	0,2395	0,2348	0,2425
Sharpe Oranı		<b>0,192</b>	0,1749	0,1912	0,1708	0,1779	0,1596	0,1679	0,1615
Sortino Oranı		<b>0,2829</b>	0,2564	0,2814	0,2499	0,2606	0,2329	0,2459	0,2359
Calmar Oranı		<b>0,1341</b>	0,1184	0,1307	0,1183	0,1193	0,1087	0,1108	0,1089
Max. Drawndown (%)		-0,3475	-0,3573	-0,3525	-0,3437	-0,3444	-0,3515	-0,3559	-0,3595
Ortalama Drawndon		-0,0441	-0,04584	-0,0438	-0,0513	-0,0522	-0,0465	-0,0498	-0,0457
Max. Drawndown Süresi		1462	1462	1449	1448	1448	1448	1434	1472
Ortalama Drawndown Süresi		87	93	87	90	87	94	80	87
Ticaret sayısı		87	61	69	41	28	40	24	169
Kazanma Yüzdesi		0,4827	0,4754	0,4492	0,4878	0,4642	0,55	0,5416	0,5088
		EN YÜKSEK GETİRİ					EN DÜŞÜK GETİRİ		

Tablo 4.22 incelendiğinde aşağıdaki bulgular gözlenmektedir:

a) Birinci senaryo (Kısa 3 gün, uzun 8 gün) uygulanmaksızın doğrudan ikinci senaryo uygulandığında (Kısa 3 gün, uzun 21 gün) BİST 30 endekste anılan dönemde en iyi getiri ve performansa ulaşılmıştır. Aynı zamanda bu senaryo fiyat oynaklığının en yüksek olduğu senaryodur.

b) Birinci senaryo (Kısa 3 gün, uzun 8 gün) uygulanmaksızın sırasıyla diğer senaryolar uygulandığında BİST 30 endekste anılan dönemde en kötü getiri ve performans kısa

pozisyonun 21 gün ve uzun pozisyonun 34 gün olarak alındığı SMA algoritması ile uygulanan ticaret sonuçları olmuştur.

c) Birinci senaryo (kısa 3 gün, uzun 8 gün) uygulanmaksızın sırasıyla diğer senaryolar uygulandığında fiyat oynaklığının yaklaşık %23-24'ler seviyesinde tüm senaryolar için stabil kaldığı görülmektedir.

d) Birinci senaryo (kısa 3 gün, uzun 8 gün) uygulanmaksızın sırasıyla diğer senaryolar uygulandığında maksimum aşağı düşme oranının, ortalama aşağı düşme oranının, maksimum aşağı düşme süresinin ve ortalama aşağı düşme süresinin çok değişmediği stabil kaldıkları gözlenmiştir.

Bu bulgular kısa vadeli pozisyonlar için 3-8 günlük SMA algoritmasının en iyi sonucu verdiği ve uygulanması gerektiği; uzun vadeli pozisyonlar için ise 3-21 günlük SMA algoritmasının en iyi sonucu verdiği ve uygulanması gerektiğini göstermektedir.

Diğer taraftan BİST 30 portföyünde 3-8 günlük senaryo çıkarılarak algoritmik ticaret yapıldığında tüm verilerin standart sapmalarının azaldığı ve verilerin istikrar kazandığı gözlenmektedir. Dolayısıyla riski sevmeyen yatırımcılar için birinci senaryo (kısa 3 gün, uzun 8 gün) uygulanmaksızın BİST 30 endeks portföyünde algoritmik ticaret yapılması daha uygun olacaktır. Aynı zamanda riskli seven yatırımcılar için ise kısa 3 gün, uzun 8 günlük senaryo uygulanarak devam eden algoritmik ticaretin uygulanması daha iyi getiri ve performans sonuçlarına ulaşılmasıyla sonuçlanabilir.

Bunun için SMA algoritması kullanılmadan önce mutlaka geçmişe yönelik kaç günlük veri ile hangi yöntemin en iyi sonuç verdiğinin test edilmesi gerekmektedir. Çünkü her hisse senedinin SMA genetik ticaret algoritmasına vereceği tepkinin birbirinden farklı olması beklenir. Bir hisse günlük verilerde 3 günlük ile 8 günlükte en iyi sonucu verebilirken başka bir SMA yönteminde 3 günlük ile 21 günlükte daha iyi getiri sağlayabilmektedir. Bu sebeple SMA genetik algoritması kullanılarak yatırım yapılmadan önce hisse senedinin ne kadar süreli zaman periyotlarında ve hangi yöntemle en iyi sonucu verdiğinin mutlaka test edilmesi gerekmektedir.

Ancak Basit Hareketli Ortalama (SMA) yönteminin yatay seyreden piyasalarda pek etkili sonuçlar vermeyebileceği hatta zarar getirebileceği de dikkate alınmalıdır. Diğer taraftan hiçbir piyasa sürekli yatay, sürekli yükseliş veya sürekli düşüş yönünde hareket edemez. Yükselişler, düşüşler piyasanın işleyişinin kaçınılmaz sonuçlarıdır. Bu sebeple SMA algoritması yöntemi her hisse senedi yatırımcısının getirisini ençoklamak için kullanabileceği veya kullanmak isteyebileceği bir yöntemdir.

Sonuç olarak basit hareketli ortalama genetik algoritması ile portföy oluşturulmasının başarılı bir şekilde gerçekleştiğini söylemek mümkündür. Bu algoritmik ticaret modelinin en önemli avantajı, değişkenlerin seçiminin bilgisayar marifetiyle yapılması, başka bir ifadeyle

değişkenlerin tarafsız bir şekilde seçilmesinin gerçekleşmesidir. Çalışmadaki gibi önerilen bir algoritmik ticaret uygulamasının kullanılması yatırım danışmanı bilgisine olan gereksinimi ortadan kaldırmakta, menkul kıymet ticaretini hızlandırmakta, yatırımların getirilerini ve performanslarını yükseltmektedir.

Gelecekteki çalışmada, daha doğru tahminler yapılabilmesi için bazı yeni özelliklerin dikakte alınması gerekmektedir. Algoritmik ticaretin verimliliğini yükseltebilmek için bazı optimizasyon algoritmalarının kullanılması yerinde olacaktır. Algoritmik ticaret kullanımının artırılması ile piyasanın bazı gizli kalmış dinamikleri yakalanabilir ve isabetli tahminler mümkün kılınabilir, getiriler ve performans yükseltilebilir. Algoritmik ticaretin kendi performansının daha da yükseltilebilmesi için daha büyük veri setlerinin ve daha geniş zaman dilimlerini içeren büyük verilerin kullanılması daha faydalı sonuçlar vererek algoritmik ticaretin daha isabetli al, sat elde tutu kararları vermesine yardımcı olabilir. Gelecekteki çalışmalarda ayrıca bu çalışma BİST 100 endekse genişletilerek alt sektör endekslerine de yayılacak şekilde uygulanabilir.

## KAYNAKÇA

- Akay D., Çetinyokuş T. Ve Dağdeviren M. (2002). Portföy seçimi problemi için KDS/GA yaklaşımı. *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, 17(4), 125-138.
- Akay, B. (2009). *Nümerik optimizasyon problemlerinde yapay arı kolonisi (artificial bee colony) algoritmasının performans analizi*, (Tez No, 246078) [Doktora Tezi, Erciyes Üniversitesi Fen Bilimleri Enstitüsü]. Yök Ulusal Tez Merkezi.
- Akgüç, Ö. (1998). *Finansal yönetim*. Avcıol Yayınevi.
- Akşin, S. (1996). *Ana çizgileriyle türkiye'nin yakın tarihi*. İmaj Yayıncılık.
- Akyer H. (2016). *Optimal portföy yönetiminde sezgisel yaklaşımlar*, (Tez No, 446279) [Doktora Tezi, Pamukkale Üniversitesi]
- Albayrak A. S., Yılmaz K. (2009). Veri madenciliği: karar ağacı algoritmaları ve İMKB verileri üzerine bir uygulama. *Süleyman Demirel Üniversitesi, İktisadi Ve İdari Bilimler, Fakültesi Dergisi*, 14(1), 31-52.
- Aldridge I. (2013). *High-Frequency Trading A Practical Guide to Algorithmic Strategies and Trading Systems*. (2. Edition), Wiley.
- Algoritmaların sınıflandırılması. *Yazılım Delisi içinde*, <https://yazilimdelisi.com/algoritmalarin-siniflandirilmesi/>
- Algorithmic Trading, (2022, 15 Temmuz). Glossary of Terms içinde <http://www.towergroup.com/research/content/glossary.jsp?page¼1&glossaryId¼382>
- Ausiello, G. (2013). *Algorithms, an historical perspective. The power of algorithms: inspiration and examples in everyday life*. Springer.
- Aygün, S., ve Akçay, M. (2015). *Yüz Tanıma Teknolojilerinde Yüksek Başarım İçin Paralel Hesaplama [Konferans Sunumu]*. 4. Ulusal Yüksek Başarımli Hesaplama Konferansı.
- Aytekin A., Çakır F. S., Yücel Y. B., Kulaözü İ. (2018). Algoritmaların hayatımızdaki yeri ve önemi. *Avrasya Sosyal ve Ekonomi Araştırmaları Dergisi*, 5(7), 151-162,
- Baykut V., Kıvanç F. E. (2004). Fibonacci sayıları. *Pivolka Dergisi*, 3(13), 3-4.
- Bayram İ. (2014). *Evrimsel Algoritmalar ile Üç Aşamalı Bir Korunmalı Alım Opsiyonu Modeli Eniyilemesi*, (Tez No, 371111) [Yüksek Lisans Tezi TOBB Ekonomi ve Teknoloji Üniversitesi]
- Bethapudi, P., Murthy, G., Ashok, P., Prithvi, B., ve Kira, S. (2018). ATM Card fraud detection system using machine learning techniques. *International Journal of Research*, 5(12), 4010-4016.
- Boehmer E., Fong K., Wu J. (2015). *International evidence on algorithmic trading*. In AFA 2013 San Diego Meetings Paper.

- Bulazar, A. R., ve Küçükçolak, Ö. Ü. R. A. (2021). *Finans sektöründe fintek etkisi*. Working Paper Series, 2(1), 53-63.
- Burkart, N., Robert, S., ve Huber, M. F. (2021). Are you sure? prediction revision in automated decision-making. In *Expert Systems*, 38(1). <https://doi.org/10.1111/exsy.12577>
- Büberkökü Ö. (2021). Alternatif yöntemlere dayalı portföy optimizasyonu. *Erciyes Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 59(3), 333-358.
- Can Ö. Y. (2020). *Makine öğrenmesi teknikleri kullanılarak kredi risk analizi*. (Tez No: 629717) [Yüksek Lisans Tezi, T.C. İstanbul Aydın Üniversitesi]
- Carles, P. G.; Doncel, L. M., Sainz, J. (2019). Stability in mutual fund performance rankings: a new proposal. *International Review of Economics and Finance*, 61(1), 337-346.
- Cvitanic ve Kirilenko. (2010). *High frequency traders and asset prices*. SSRN, 6. Electronic copy available at: <https://ssrn.com/abstract=1569067>
- Chambers, L. D. (1998). *Practical handbook of genetic algorithms: complex coding systems*. CRC Press.
- Chang, C. L. ve Y. T. Liu (2007). Genetic algorithms for portfolio selection problems with minimum transaction lots. *European Journal of Operational Research*, 185(34), 547-566.
- Chang, J. F., Wang, T. C., ve Min, Y. T. (2010). *Using Genetic Algorithms To Construct A Low-Risk Fund Portfolio Based On The Taiwan 50 Index*. [Konferans Sunumu], In *Computational Aspects Of Social Networks (Cason)*, 2010 International Conference On, 284-289.
- Çelik M. S., Öztürk M. B. (2022). Sermaye piyasalarında algoritmik ve yüksek frekanslı işlem stratejileri. *Niğde Ömer Halisdemir Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 4(1), 77-85.
- Chen, Tai L.; Cheng, Ching H.; Teoh, Hia J. (2007), Fuzzy time-series based on fibonacci sequence for stock price forecasting, *Physica A: Statistical Mechanics and Its Applications*, 380(1), 377-390.
- Çelenli A. Z., Eğrioğlu E., Çorba B. Ş. (2015). İMKB 30 indeksini oluşturan hisse senetleri için parçacık sürü optimizasyonu yöntemlerine dayalı portföy optimizasyonu. *Doğuş Üniversitesi Dergisi*, 16 (1), 25-33.
- Çetin, E. (2005). *Dinamik programlama ile sınır tenörü optimizasyonu*. [Sempozyum Sunumu]. Doğu ve Güneydoğu Madenlerinin Değerlendirilmesi Sempozyumu, 139-143.
- Çetiner, E. M.; Gürel C. A. ve Türkekul, F. (2022). Küreselleşen dünyada finansal yönetim alanında dönüşümler fırsatlar ve tehditler. *International Social Mentality and Researcher Thinkers Journal*, 8(55), 363-372.
- Daniel T. L. (2005). *Discovering knowledge in data: An Introduction to Data Mining*. USA: John and Wiley Sons Incorporated.
- Daştan İ. (2021). *Yüksek frekanslı işlemlerin borsa istanbul pay piyasasındaki uç fiyat hareketleri çerçevesinde incelenmesi*, (Tez no, 662690) [Yüksek Lisans Tezi, Kadir Has Üniversitesi]

- Demirhan, A., Kılıç, Y. A., ve İnan, G. (2010). Tıpta yapay zekâ uygulamaları. *Yoğun Bakım Dergisi*, 9(1), 31-41.
- Demirtaş, Ö. ve Z. Güngör (2004). Portföy Yönetimi ve portföy yönelik uygulama. *Havacılık ve Uzay Teknolojileri Dergisi*, 1(4), 103-109.
- Devlet İstatistik Enstitüsü. (2002). *2000 Yılı Genel Nüfus Sayımı: Nüfusun Sosyal ve Ekonomik Nitelikleri-19 Çorum*. Ankara: Devlet İstatistik Enstitüsü.
- Dorigo, M., Birattari, M. ve Stutzle, T. (2006). Ant colony optimization. *Computational Intelligence Magazine, IEEE*, 1(4), 28-39.
- Durmuşkaya S., Garayev K. (2017). Genetik algortima ile portföy seçiminde kriz dönemi etkisi, bist-30'da bir uygulama. *İşletme Bilimi Dergisi (JOBS)*, 5(3), 173-187.
- Elmas, Ç. (2011). *Yapay zekâ uygulamaları* (2. Baskı). Seçkin Yayınevi.
- El Naqa, I., ve Murphy, M. (2015). What is machine learning? I. El Naqa, R. Li, & M. Murphy içinde, *Machine Learning in Radiation Oncology* (s. 3-11). Springer International Publishing. doi:10.1007/978-3-319-18305-3
- Endeksler, (2022 19 Ağustos). *Kap içinde*, <https://www.kap.org.tr/tr/Endeksler>
- Ersan, O. ve Ekinci, C. (2016). Algorithmic and high frequency trading in Borsa İstanbul. *Borsa İstanbul Review*, 16(4), 233-248.
- Eshlaghy, T.A., Abdolahi, A., Moghadasi, M. ve Maatofi, A. (2011). Using genetic and particle swarm algorithms to select and optimize portfolios of companies admitted to tehran stock exchange. *Research Journal Of International Studies*, 20(1), 95-105.
- Friedlander J. (2005). Algo wars. *Investment Dealers' Digest*, 30(6), 6-8.
- Finn, E. (2020). *Algoritmalar ne ister? - Hesaplama çağında hayal gücü*. Çev. S. Köse. Tellekt Yayıncılık.
- Gentsch, P. (2018). *AI in marketing, sales and service: how marketers without a data science degree can use ai, big data and bots*. Palgrave Macmillan.
- Glantz, M., ve Kissell, R. (2013). *Multi-asset risk modeling: techniques for a global economy in an electronic and algorithmic trading era*. Academic Press.
- Gomber P., Arndt B., Lutat M. Ve Uhle T. E. (2011). *High Frequency Trading Working Paper*.
- Gökgöz, E. (2006). *Riske maruz değer (Var) ve portföy optimizasyonu*. SPK Yayınları, Yayın No: 190.
- Guenoun, Z. ve F. Hamza (2012). Stock portfolio optimization using classification and genetic algorithm. *Applied Mathematical Sciences*, 6(94), 4673-4684.
- Hendershott T., Jones C. M., ve Menkveld A. J. (2011). Does algorithmic trading improve liquidity?, *The Journal Of Finance*, 66(1), 1-33.
- Hendershott T. and Riordan R. (2013). Algorithmic trading and the market for liquidity. *Journal Of Financial And Quantitative Analysis* 48(4), 1001-1024.

- Howard, J. (2019). Artificial intelligence: implications for the future of work. *American Journal of Industrial Medicine*, 62(11), 917-926.
- Illiadis A. ve Russo F. (2020). *Eleştirel veri çalışmaları*. Filiz Aydoğan (der.), Yeni Medya Kuramları 2. Der Yayınları.
- Irmak H. (2019). *Yapay zekâ kullanılarak borsa istanbul (BİST) için algoritmik işlem stratejilerinin geliştirilmesi*, (Tez No, 589935) [Yüksek Lisans Tezi, Hacettepe Üniversitesi]
- Ismail M. M., Atiya, A. F., Pratap, A., ve Abu-Mostafa, Y. S. (2004). On the maximum drawdown of a brownian motion. *Journal of Applied Probability*, 41(1), 147-161.
- İstanbul Barosu (2020). *Algoritmik karar verme ve veri koruması*. Yapay Zekâ Çalışma Grubu Raporu. <https://ssrn.com/abstract=3792207>
- Jantan, H., Hamdan, A.R. ve Othman, Z. A. (2010) *Intelligent techniques for decision support system in human resource management*. Decision Support Systems: Advances in, Ger Devlin (Ed.), InTech Europe Open Publishing.
- Jensen, R. ve Shen, Q. (2008). *Computational intelligence and feature selection rough and fuzzy approaches*. IEEE Press.
- Kandemir Ş. (2021). Bankacılık ve finansın denetiminde denetim teknolojisi (SupTech) ve yapay zekâ. *Çağ Üniversitesi Sosyal Bilimler Dergisi*, 18(1), 59-81.
- Karaboğa, D. (2011). *Yapay zekâ optimizasyon algoritmaları*. Nobel Yayın Dağıtım.
- Karacabey, A. A., Gökgöz, F. (2005). *Emeklilik fonlarının portföy analizi*. Siyasal Kitabevi.
- Karan, M. B. (2001). *Yatırım analizi ve portföy yönetimi*. Gazi Kitabevi.
- Keskintürk, T. (2007). Portföy seçiminde markowitz modeli için yeni bir genetik algoritma yaklaşımı, *Yönetim Dergisi*, 18(56), 78-90.
- Keskintürk, T., Demirci, E. ve Tolun, S. (2010). İyi Çeşitlendirilmiş portföy büyüklüğünün genetik algoritma tekniği kullanılarak incelenmesi. *Sosyal Bilimler Dergisi*, 2(1), 1-5.
- Kim, K.. (2007). *Electronic and algorithmic trading technology: the complete guide*. Elsevier Inc.
- Kissell R. (2021). *Algorithmic trading methods applications using advanced statistics, optimization, and machine learning techniques*. Elsevier.
- Korkmaz, T. ve Uygurtürk, H. (2008). Türkiye'deki emeklilik fonları ile yatırım fonlarının performans karşılaştırması ve fon yöneticilerinin zamanlama yetenekleri. *Kocaeli Üniversitesi Sosyal Bilimler Dergisi*, (15), 114-147.
- Kurzweil R. (1985). What is artificial intelligence anyway?. *American Scientist*, 73(1), 258-264.
- Küçüksille E. U., Tokmak M. (2011). Yapay arı kolonisi algoritması kullanarak otomatik ders çizelgeleme. *Süleyman Demirel Üniversitesi, Fen Bilimleri Enstitüsü Dergisi*, 15(3), 203-210.

- Küpeli H. (2013). *Çok Amaçlı Atama Probleminin Çözümü için Genetik Algoritma ile Bir Yaklaşım*. (Tez No, 332157) [Yüksek Lisans Tezi, Ege Üniversitesi, Fen Bilimleri Enstitüsü]
- Lai, K. K., Yu, L., Wang, S. ve Zhou, C. (2006). A double-stage genetic optimization algorithm for portfolio selection. *Neural Information Processing*, 4234(1), 928-937.
- Leshik E. A, Cralle J. (2011). *An introduction to algorithmic trading, basic to advanced strategies*. John Wiley Corp.
- Lin, C. M. ve Gen, M. (2007). An effective decision-based genetic algorithm approach to multiobjective portfolio optimization problem. *Applied Mathematical Sciences*, 1(5), 201-210.
- Mager, A. (2015). *Glocal search: search technology at the intersection of global capitalism and local socio-political cultures*. Vienna: Institute of Technology Assessment (ITA), Austrian Academy of Sciences. Retrieved from <http://www.astridmager.net/wp-content/uploads/2015>.
- Malhotra, R. ve Malhotra, D. K. (2003). Evaluating consumer loans using neural networks. *OMEGA*, 31(1), 83-96.
- Mavlutova, I., ve Volkova, T. (2019). *Digital transformation of financial sector and challenges for competencies development*. In 2019 7th International Conference on Modeling, Development and Strategic Management of Economic System (MDSMES 2019) (ss. 161-166).
- Mayıl G. (2011). *Hareketli ortalamalar yöntemine göre oluşan al/sat sinyal getirilerinin imkb'de değerlendirilmesi*, (Tez No, 296242 ) [Doktora Tezi, T.C. Sakarya Üniversitesi]
- Mercangöz, A. B. (2019). Parçacık sürü optimizasyonu ile portföy optimizasyonu: borsa istanbul ulaştırma sektörü hisseleri üzerine bir uygulama. *Journal of Yasar University*, 14 (Special Issue), 126-136.
- Michalewicz, Z. (1996). *Genetic algorithms + data structures = evolution programs*. Springer-Verlag.
- Mitchell, R.; Frank, E. (2017). Accelerating the XGBoost algorithm using gpu computing, *PeerJ Computer Science*, 3(1), 127-164.
- Nilsson J. N. (2019). *Yapay zekâ geçmişi ve geleceği*. Boğaziçi Üniversitesi Yayınevi.
- Okumuş, H.; Aydemir, Ö. (2017). Random Forest classification for brain computer interface applications, *In Signal Processing and Communications Applications Conference (SIU)*, 25(1), 1-4.
- Öz, C., ve Alan, Ü. (2020). *Yeni Medya 451*. Podcast Yayını (Bölüm 4).
- Özçalıcı M. ve Ayriçay Y. (2016). Bilgi işlemsel zekâ yöntemleri ile hisse senedi fiyat tahmini: BİST uygulaması. *Pamukkale Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 25(1), 274-298.
- Özçalıcı Ö. M. (2011). Genetik algoritma kullanılarak portföy seçimi. *İktisat İşletme ve Finans*, 26(299), 43-66.
- Özçakır, F. C., Çamurcu, A. Y., (2007). Birliktelik kuralı yöntemi için bir veri madenciliği yazılımı tasarımı ve uygulaması. *İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi*, 6(12), 21-37.
- Öztemel, E. (2006). *Yapay Sinir Ağları*. Papatya Yayıncılık.

- Özyinelemeli algoritmaların özellikleri. (2022, 10 Haziran). *Khan Academy İçinde*, <https://tr.khanacademy.org/computing/computer-science/algorithms/recursive-algorithms/a/properties-of-recursive-algorithms>
- Pabuççu H. (2019). Borsa endeksi hareketlerinin tahmini: trend belirleyici veri. *Selçuk Üniversitesi Sosyal Bilimler Meslek Yüksekokulu Dergisi*, 22(1), 246-256.
- Pandari, A. R., Azar, A. ve Shavazi, A. R. (2012). Genetic algorithms for portfolio selection problems with non-linear objectives. *African Journal Of Business Management*, 6(20), 6209-6216.
- Parpinelli, R. S., Lopes, H.S. ve Freitas, A. A. (2001). *An ant colony based system for data mining: applications to medical data*. Proc. 2001 Genetic and Evolutionary Computation Conference (GECCO-2001), USA, 791-797.
- Patterson J., Gibson A. (2018). *Foundations Of neural networks and deep learning, getting started with deep learning*. O'Reilly Media.
- Poli R., Kennedy J., Blackwell T. (2007). Particle swarm optimization. *Swarm Intell* 1(1):33-57. doi:10.1007/s11721-007-0002-0
- Prado M., Lopez de, M. (2018). *Advances in financial machine learning*. John Wiley & Sons, Inc.
- Prix J., Loistl O. ve Huetl M. (2007). Algorithmic Trading patterns in xetra orders. *The European Journal of Finance*, 13(8), 717-739, <https://doi.org/10.1080/13518470701705538>
- Riedl, M. O. (2019). Human-centered artificial intelligence and machine learning. *Human Behavior and Emerging Technologies*, 1(1), 33-36.
- Roudier F. (2006). *Portfolio optimization and genetic algorithms*. [Master's thesis, Zurich University, Department of Management, Technology and Economics] Swiss Federal Insqtitute of Technology (ETM).
- Ruiz R., Stutzle T., (2007). A simple and effective iterated greedy algorithm for the permutation flowshop scheduling problem. *European Journal of Operational Research*, 177(1), 2033-2049.
- Sarıtaş, M. M., Yaşar A. (2019). Performance Analysis of ann and naive bayes classification algorithm for data classification. *International Journal of Intelligent Systems and Applications in Engineering (IJISAE)*, 7(2), 88-91.
- Say C. (2019). *50 soruda yapay zekâ*. İstanbul: Bilim ve Gelecek Kitaplığı.
- Sezer, O. B., Özbayoğlu, A. M. ve Doğdu, E., (2017b). *An Artificial neural network-based stock trading system using technical analysis and big data framework*. Proceedings of the SouthEast Conference, 13-15 April, Kennesaw, USA,
- Silahtaroğlu, G. (2008). *Kavram ve algoritmalarıyla temel veri madenciliği*. Papatya Yayıncılık.
- Soam, V., Palafox, L., & Iba, H. (2012, June). *Multi-objective portfolio optimization and rebalancing using genetic algorithms with local search*. In Evolutionary Computation (CEC), 2012 IEEE Congress on (pp. 1-7). IEEE.

Sortino, F. A., Price, L. N. (1994). Performance measurement in a downside risk framework. *The Journal of Investing*, 3(3), 59-64.

Soysal, Y. N. (2000). *Changing citizenship in europe: postnational membership and the national state*. Redefining the Nation State and Citizen, Ed. Günay Göksu Özdoğan and Gül Tokay, (ss. 253-263). Eren Yayıncılık.

Sütçü, C. S. ve AYTEKİN, Ç. (2019). *Veri bilimi*. Paloma Yayınevi

Swinburne, R. (1991). *Tümevarımsal kanıtlar, din felsefesi metinleri*. Der. Andrew Rippin, Çev. Türkan Ulu, Say Yayınları.

Şener, G. (2021). *Algoritmalar Karşı aktivist taktikler. algoritmaların gölgesinde toplum ve iletişim*. Kitapta Bölüm: Ed. Kuş Oğuz, Alternatif Bilişim Derneği yayınları. <https://www.alternatifbilisim.org>

Şişmanoğlu G., Koçer F., Önde M. A., Şahingöz Ö. K. (2020). Derin öğrenme yöntemleri ile borsada fiyat tahmini. *Beü Fen Bilimleri Dergisi*, 9 (1), 434-445.

Teker, S., Karakum, E. ve Tav, O. (2008). Yatırım fonlarının risk odaklı performans değerlemesi. *Doğuş Üniversitesi Dergisi*, 9(1), 89-105.

The 21 Most Popular Trading Strategies Every Serious Trader Should Learn To Succeed (25 Eylül 2022). Quant strategies Academy içinde, [https://quantstrategies.academy/2020/05/19/the-21-most-popular-trading-strategies-every-serious-trader-should-learn-to-succeed/?gclid=EAIaIQobChMImpHD3P2l-gIVWo9oCR2apQGvEAAAYASAAEgJkz\\_D\\_BwE#Single\\_moving\\_average](https://quantstrategies.academy/2020/05/19/the-21-most-popular-trading-strategies-every-serious-trader-should-learn-to-succeed/?gclid=EAIaIQobChMImpHD3P2l-gIVWo9oCR2apQGvEAAAYASAAEgJkz_D_BwE#Single_moving_average)

Toker, K. A. (2021). Büyük veri ışığında algoritmaların dönüştürücü gücü: kültür makineleri. *Yeni Medya Elektronik Dergisi*, 5 (3), 204-218.

Toloie-Eshlaghy, A., Abdolahi, A., Moghadasi, M., ve Maatofi, A. (2011). Using genetic and particle swarm algorithms to select and optimize portfolios of companies admitted to tehran stock exchange. *Research Journal of International Studies-Issue*, 95(1), 85-96.

Türe, M., Tokatlı F. ve Kurt İ., (2008). *Using kaplan-meier analysis together with decision tree methods (C&RT, CHAID, QUEST, C4.5 and ID3) In Determining Recurrence-Free Survival of Breast Cancer Patients*, Expert Systems With Applications, Article in Pres.

Ustalı Koç, N., Tosun, N., Tosun, Ö. (2021). Makine öğrenmesi teknikleri ile hisse senedi fiyat tahmini, *Eskişehir Osmangazi Üniversitesi İİBF Dergisi*, 16(1), 1 – 16. Doi: 10.17153/oguiibf.636017

Ünsal Ö., Kaya M. E. (2020). Forex piyasaları için bayes tekniği ile otomatik al/sat sinyali üretilmesi. *Mühendislik Bilimleri ve Tasarım Dergisi*, 8(4), 1190 – 1201, e-ISSN: 1308-6693

Ünsal Ö. (2020). Veri madenciliği teknikleri ile hisse senetleri arasındaki fiyat etkileşimlerinin belirlenmesi. *Mühendislik Bilimleri ve Tasarım Dergisi*, 8(5), 106-112, 2020 e-ISSN: 1308-6693

Vahaplar A. (2003). *Bir coğrafi veri madenciliği uygulaması*, (Tez No, 134587), [Yüksek Lisans Tezi, Ege Üniversitesi]

Vecer, J., Novotny, P., and Pospisil, L. (2006). *Measuring severity of the market crashes by using contracts on maximum relative drawdown*. Columbia University, Department of Statistics. <http://www.stat.columbia.edu/~vecer/reldrawdown2006.pdf>

Velu R., Hardy M. ve Nehren D. (2020). *Algorithmic trading and quantitative strategies*. CRC Press.

Verheggen R. (2017). *The rise of algorithmic trading and its effects on return dispersion and market predictability*. [Master Thesis, Tilburg University, Finance]

Witten, I., Frank, E., ve Hall, M. (2011). *Data mining practical machine learning tools and techniques*. Morgan Kaufmann Publishers.

Yaşar F. (2016). *Fibonacci sayı türleri ve kuaterniyonlarda kullanımı*, [Yüksek Lisans Tezi, Gaziantep Üniversitesi]

Yıldız E. (2020). *Rekabet Politikası Çerçevesinde Gizli Algoritmik Anlaşmalar, Makine öğrenme ve yapay zekâ*, [Uzmanlık Tezi, Rekabet Kurumu]

Yılmaz, T. (2022). *Mesele: geo-code otomasyonlu bilgi sistemleri ve istatistik*. Bilgi Yönetimi. Erişim tarihi: 15 Ocak 2022. [www.bilgiyonetimi.org/cm/pages/mkl\\_gos.php?nt=67](http://www.bilgiyonetimi.org/cm/pages/mkl_gos.php?nt=67)

Yılmazel B., Gündüz S. Y., Uysal A. K. (2016). *Algoritmalar ve programlama*. Ed. Günel Serkan. T.C. Anadolu Üniversitesi Yayını No: 3445.

Yılmaz, V. (2019). Finansmanı öğrenen makineler. *4th International Symposium on Innovative Approaches in Social, Human and Administrative Sciences*, November 22-24, 2019, Samsun, Turkey SETSCI Conference Proceedings 4 (8), 187-192.

Zheng, H.; Yuan, J.; Chen, L. (2017). Short-term load forecasting using emd-lstm neural networks with a xgboost algorithm for feature importance evaluation. *Energies*, 10(8), 1168-1188.

Zeren, F. ve Baygın, M. (2015). Genetik algoritmalar ile optimal portföy seçimi: BİST-30 örneği. *Journal Of Business Research Türk*, 7(1), 309-324.

Zhou, C., Yu, L., Huang, T., Wang, S., ve Lai, K. K. (2006). Selecting valuable stock using genetic algorithm. *In Simulated Evolution and Learning*, 4247, 688-694.

Zuboff, S. (2015). Big other: surveillance capitalism and the prospects of an information civilization. *Journal of Information Technology*, 30(1), 75-89. <https://doi.org/10.1057/jit.2015.5>.

