ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

MACHINE LEARNING APPLICATIONS FOR TIME SERIES ANALYSIS

M.Sc. THESIS

Mert Can

Department of Mathematics Engineering

Mathematics Engineering Programme

JUNE 2024

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

MACHINE LEARNING APPLICATIONS FOR TIME SERIES ANALYSIS

M.Sc. THESIS

Mert Can
(509191237)

Department of Mathematics Engineering

Mathematics Engineering Programme

Thesis Advisor: Prof. Dr. Atabey KAYGUN

JUNE 2024

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

ZAMAN SERİLERİ ANALİZİ İÇİN MAKİNE ÖĞRENMESİ UYGULAMALARI

YÜKSEK LİSANS TEZİ

Mert Can
(509191237)

Matematik Mühendisliği Anabilim Dalı

Matematik Mühendisliği Programı

Tez Danışmanı: Prof. Dr. Atabey KAYGUN

HAZIRAN 2024

Mert Can, a M.Sc. student of ITU Graduate School student ID 509191237 successfully defended the thesis entitled “MACHINE LEARNING APPLICATIONS FOR TIME SERIES ANALYSIS”, which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor :	**Prof. Dr. Atabey KAYGUN** Istanbul Technical University	.....
Jury Members :	**Assoc. Prof. Dr. Gül İnan** Istanbul Technical University	.....
	Prof. Dr. Özgür MARTİN Mimar Sinan Fine Arts University	.....

Date of Submission : **24 May 2024**
Date of Defense : **24 June 2024**

To my father,

FOREWORD

The support and encouragement of numerous significant individuals in my life were instrumental in the completion of this thesis. Consequently, I would like to express my immense appreciation to them.

I dedicate this thesis to my loved father, Muharrem Can, who passed away during the COVID-19 pandemic. My father was a source of inspiration throughout my life, and his perseverance and determination served as a guide for me in the completion of this work. I recall him with affection and yearning. Also, I am profoundly appreciative of the unwavering support that my beloved mother, Hanife Can, provided throughout this process. I was able to achieve this juncture as a result of her patience, love, and sacrifices. I was most motivated to overcome the obstacles I encountered by my mother's faith in me.

I appreciate my advisor, Prof. Dr. Atabey Kaygun, for his essential mentorship, critical critique, and motivation. He consistently provided unwavering support and maintained complete trust in me, and his profound insights greatly enhanced the calibre of my work. Additionally, I would like to extend my heartfelt appreciation to Prof.Dr.Ayse Aslı Berkman Pierce for her unwavering support throughout my existence.

Lastly, I would like to extend my gratitude to all of my family and friends who have been there for me during this voyage. Their affection and encouragement provided me with the fortitude to finalise this thesis.

With my sincerest gratitude,

June 2024 Mert Can

TABLE OF CONTENTS

ABBREVIATIONS

ACF	: Autocorrelation Function
ADF	: Augmented Dickey-Fuller
AIC	: Akaike Information Criterion
AR	: Autoregressive
ARCH	: Autoregressive Conditional Heteroskedasticity
ARIMA	: Autoregressive Integrated Moving Average
ARMA	: Autoregressive Moving Average
BIC	: Bayesian Information Criterion
BIST	: Borsa Istanbul
BIST 30	: Borsa Istanbul 30 Index
BIST 100	: Borsa Istanbul 100 Index
BTC	: Bitcoin
EGARCH	: Exponential GARCH
ETH	: Ethereum
GARCH	: Generalized Autoregressive Conditional Heteroskedasticity
GJR-GARCH	: Glosten-Jagannathan-Runkle GARCH
KPSS	: Kwiatkowski-Phillips-Schmidt-Shin
MA	: Moving Average
MAE	: Mean Absolute Error
MAPE	: Mean Absolute Percentage Error
MSE	: Mean Squared Error
PACF	: Partial Autocorrelation Function
RMSE	: Root Mean Squared Error
SARIMA	: Seasonal ARIMA
USD	: United States Dollar
XU030.IS	: BIST 30 Index Ticker
XU100.IS	: BIST 100 Index Ticker

LIST OF TABLES

Page

LIST OF FIGURES

MACHINE LEARNING APPLICATIONS FOR TIME SERIES ANALYSIS

SUMMARY

In this studying, involves doing a range of tests using time series data sets collected from stock markets (BIST30, BIST100, Apple) and cryptocurrency marketplaces. Statistical analysis and artificial intelligence models are employed to investigate various data sets inside the studies, and the findings are subsequently analysed. The main objective of the study is to provide a valuable contribution to academic research and offer practical advantages to market investors. Consequently, the researcher has thoroughly examined the current models and studies in the literature and has chosen the most suitable artificial intelligence models (ARIMA, SARIMA, GARCH) for the thesis study. The paper extensively discusses and applies these concepts within its scope.

The study's findings indicate that no existing framework can accurately forecast the time series-dependent pricing of crypto assets traded on stock exchanges and crypto exchanges. These conclusions are based on the results gained from the experiments conducted. The primary factors contributing to this unpredictability can be ascribed to market price volatility and the fact that price variations generate outcomes regardless of time.

Subsequent investigations might prioritise the utilisation of additional data sources to enhance the existing time series data, hence enhancing the precision of prediction outcomes. Incorporating supplementary information such as macroeconomic indicators, sector-specific data, geopolitical events, and social media sentiment can augment the precision of prediction models. This thesis study offers essential insights into the predictability of financial time series. The present pricing and daily price changes alone are inadequate in providing credible predictions. This is because elements such as seasonality, seasonal variability, and periodic trends, which are stochastic in nature, make the prediction process more complex.

The thesis clearly demonstrates the constraints and difficulties encountered in financial market analysis as described in the literature, with the assistance of data derived from both literature-based research and experiments. The statistical methods used and the data gained in this study serve as an initial investigation, with the goal of establishing a methodological basis for future studies and opening up possibilities for further research in many areas. This study is expected to serve as a benchmark for future market researchers and academics conducting research in this subject.

ZAMAN SERİLERİ ANALİZİ İÇİN MAKİNE ÖĞRENMESİ UYGULAMALARI

ÖZET

Günümüzde yapay zeka teknolojilerinin hızla gelişmesi ve bireylerin finansal yatırım araçlarına yönelik farkındalığının artması, finansal piyasaların daha derinlemesine anlaşılmasını gerektirmektedir. Bu gelişmeler, geleneksel finans teorilerinin ötesinde, daha karmaşık ve çok boyutlu analizlerin yapılmasına olanak sağlamaktadır. Özellikle makine öğrenimi algoritmaları, büyük veri setlerindeki karmaşık desenleri tespit etme ve bu desenleri kullanarak geleceğe yönelik tahminler yapma konusunda önemli avantajlar sunmaktadır. Bu bağlamda, zaman serisi analizleri, finansal varlıkların davranışlarını modellemek ve gelecekteki hareketlerini tahmin etmek için kritik bir araç haline gelmiştir.

Yapay zeka ve makine öğrenimi teknikleriyle desteklenen bu analizler, yatırımcıların daha bilinçli kararlar almasına, risk yönetimi stratejilerinin geliştirilmesine ve piyasa etkinliğinin artırılmasına katkıda bulunmaktadır. Örneğin, derin öğrenme modelleri, geleneksel istatistiksel yöntemlerin yakalayamadığı doğrusal olmayan ilişkileri ve uzun vadeli bağımlılıkları tespit edebilmektedir. Bu, özellikle yüksek frekanslı ticaret ve algoritmik alım-satım stratejilerinin geliştirilmesinde büyük önem taşımaktadır. Ayrıca, doğal dil işleme teknikleri kullanılarak finansal haberler ve sosyal medya verileri analiz edilebilmekte, bu da piyasa duyarlılığının ölçülmesine ve yatırım kararlarının iyileştirilmesine olanak sağlamaktadır.

Finansal piyasaların karşılaştırmalı analizi, global ekonomik trendlerin anlaşılması ve uluslararası portföy çeşitlendirmesi stratejileri için kritik bilgiler sağlamaktadır. Bu analizler, farklı ülke borsaları arasındaki korelasyonları, sektörler arası etkileşimleri ve makroekonomik faktörlerin piyasalar üzerindeki etkilerini incelemektedir. Örneğin, gelişmiş ve gelişmekte olan piyasalar arasındaki ilişkilerin analizi, yatırımcılara küresel risk dağıtımı konusunda önemli içgörüler sağlamaktadır. Ayrıca, bu tür karşılaştırmalı analizler, finansal bulaşma etkilerinin ve sistemik risklerin belirlenmesinde de kritik rol oynamaktadır.

Kripto para piyasaları gibi yeni ortaya çıkan finansal araçların geleneksel piyasalarla olan etkileşimi, finansal sistemin evrimini anlamak açısından önem taşımaktadır. Bu yeni varlık sınıfı, geleneksel finansal teorileri ve risk yönetimi yaklaşımlarını yeniden değerlendirmeyi gerektirmektedir. Kripto paraların yüksek volatilitesi, geleneksel varlıklarla düşük korelasyonu ve 7/24 işlem görebilme özellikleri, hem fırsatlar hem de zorluklar sunmaktadır. Farklı varlık sınıflarının güvenilirlik ve öngörülebilirlik açısından karşılaştırılması, risk-getiri profillerinin belirlenmesinde kritik rol oynamaktadır. Örneğin, devlet tahvilleri genellikle daha öngörülebilir ve

düşük riskli olarak kabul edilirken, hisse senetleri ve kripto para birimleri daha yüksek volatilite ve rastgele davranış sergileyebilmektedir. Bu farklılıkların anlaşılması, yatırımcıların risk toleranslarına ve yatırım hedeflerine uygun portföyler oluşturmasına yardımcı olmakta, aynı zamanda düzenleyici otoritelerin piyasa istikrarını korumak için gerekli politikaları geliştirmelerine katkıda bulunmaktadır.

Bu tez çalışması, BIST30, BIST100, Apple hisse senetleri ile Bitcoin ve Ethereum gibi kripto para birimlerinin günlük kapanış fiyatlarını içeren çeşitli veri setlerini incelemektedir. Çalışmada, ARIMA, SARIMA ve GARCH modelleri uygulanmış, veri ön işleme aşamasında logaritmik dönüşümler ve durağanlaştırma teknikleri kullanılmıştır. Serilerin durağanlığı Augmented Dickey-Fuller (ADF) ve Kwiatkowski-Phillips-Schmidt-Shin (KPSS) testleri ile incelenmiştir.

ARIMA modelleme sonuçları, BIST30, BIST100 ve Apple için ARIMA(0,1,0) modelinin en uygun olduğunu göstermiş, bu da bu varlıkların basit rassal yürüyüş sürecine yakın davrandığını ortaya koymuştur. Bitcoin için ARIMA(2,1,0) ve Ethereum için ARIMA(1,1,2) modellerinin seçilmesi, kripto para birimlerinin daha karmaşık zaman serisi özelliklerine sahip olduğunu göstermiştir. GARCH modelleme sonuçları, tüm varlık sınıfları için asimetrik volatilite modellerinin (GJR-GARCH ve EGARCH) standart GARCH modellerinden daha iyi performans gösterdiğini ortaya koymuştur. Bu, finansal piyasalardaki asimetrik şokların önemini vurgulamaktadır. BIST30 ve BIST100 için GJR-GARCH(1,1), Bitcoin ve Ethereum için EGARCH(1,1), Apple için GJR-GARCH(1,2) modellerinin en uygun bulunması, her bir varlık sınıfının kendine özgü volatilite dinamiklerine sahip olduğunu göstermiştir.

Çalışmanın temel bulguları, finansal zaman serilerinin öngörülebilirliğinin oldukça sınırlı olduğunu ortaya koymaktadır. Bu sonuç, finansal piyasaların karmaşık ve dinamik yapısını yansıtmakta ve etkin piyasa hipotezi ile uyumlu görünmektedir. Mevcut fiyatlandırma veya günlük fiyat değişimleri, gelecekteki hareketleri tahmin etmek için tek başına yeterli olmamaktadır. Bu durum, piyasaların sürekli değişen doğasını ve çoklu faktörlerin etkileşimini göstermektedir. Mevsimsellik, sezonsal değişkenlik ve dönemsel trendler gibi faktörler, tahmin sürecini önemli ölçüde karmaşıklaştırmaktadır. Bu faktörler, finansal varlıkların fiyat hareketlerinde tekrarlanan ancak düzenli olmayan kalıplar oluşturabilmektedir. Örneğin, bazı hisse senetleri yılın belirli dönemlerinde daha iyi performans gösterebilirken, diğerleri ekonomik döngülere daha duyarlı olabilir. Bu tür karmaşık etkileşimler, basit doğrusal modellerin ötesinde, daha sofistike tahmin yöntemlerinin gerekliliğini vurgulamaktadır.

Özellikle kripto para birimlerinin, geleneksel finansal varlıklara göre daha karmaşık ve değişken dinamikler sergilediği gözlemlenmiştir. Bu gözlem, kripto para piyasalarının görece yeni olması, regülasyon eksikliği ve teknolojik gelişmelere olan yüksek duyarlılığı ile açıklanabilir. Kripto paraların fiyat hareketleri, sosyal medya trendleri, ünlü kişilerin açıklamaları ve global ekonomik olaylar gibi çok çeşitli faktörlerden etkilenebilmektedir. Bu durum, kripto para birimlerinin tahmin edilebilirliğini daha da zorlaştırmakta ve geleneksel finansal modellerin bu piyasalarda yetersiz kalabileceğini göstermektedir.

Gelecekteki araştırmalar için, makroekonomik göstergeler, sektörel veriler, jeopolitik olaylar ve sosyal medya duyarlılığı gibi ek faktörlerin modellere dahil edilmesi

önerilmektedir. Bu çok boyutlu yaklaşım, finansal varlıkların fiyat dinamiklerini etkileyen karmaşık ilişkileri daha iyi yakalayabilir. Örneğin, makroekonomik göstergeler (enflasyon oranları, faiz oranları gibi) ile hisse senedi fiyatları arasındaki ilişkinin incelenmesi, daha kapsamlı tahmin modelleri oluşturulmasına olanak sağlayabilir. Benzer şekilde, sosyal medya analizleri ve duygu analizi teknikleri, özellikle kripto para piyasalarındaki yatırımcı davranışlarını ve piyasa psikolojisini anlamak için değerli içgörüler sunabilir.

Daha kısa zaman aralıklarında (saatlik, 30 dakikalık veya 15 dakikalık) analizler yapılması, piyasa dinamiklerini daha iyi yakalayabilir ve tahmin performansını artırabilir. Bu yaklaşım, özellikle yüksek frekanslı ticaret ve gün içi fiyat hareketlerinin anlaşılması açısından kritik öneme sahiptir. Yüksek frekanslı veriler, piyasa mikroyapısı hakkında daha detaylı bilgiler sağlayarak, likidite dinamikleri, emir akışı ve piyasa derinliği gibi faktörlerin fiyat oluşumu üzerindeki etkilerini inceleme fırsatı sunar. Bu tür analizler, alım-satım stratejilerinin optimize edilmesine ve piyasa verimliliğinin artırılmasına katkıda bulunabilir.

Kısa vadeli fiyat hareketlerini etkileyen faktörlerin daha iyi anlaşılması, risk yönetimi ve portföy optimizasyonu açısından da önemlidir. Örneğin, ani fiyat sıçramaları veya düşüşleri, piyasa katılımcılarının davranışları ve haberlere olan tepkiler gibi kısa vadeli dinamikler, bu tür analizlerle daha net bir şekilde ortaya konabilir. Ancak, yüksek frekanslı verilerin analizi, büyük veri setleriyle çalışmayı gerektirdiğinden, ileri düzey veri işleme ve analiz tekniklerinin kullanılması zorunludur. Bu noktada, büyük veri teknolojileri, dağıtık hesaplama sistemleri ve yapay zeka algoritmaları gibi ileri teknolojilerin kullanımı önem kazanmaktadır.

Bu tez çalışması, finansal piyasaların karmaşık ve öngörülmesi zor doğasını ortaya koyarak literatüre önemli katkılar sunmaktadır. Çalışmanın en önemli katkılarından biri, farklı varlık sınıfları için en uygun modelleme tekniklerini belirlemesidir. Bu, yatırımcılar ve finansal analistler için piyasa dinamiklerini anlamada ve risk yönetiminde kullanılabilecek değerli içgörüler sağlamaktadır. Örneğin, hisse senetleri, tahviller, döviz kurları ve kripto para birimleri gibi farklı varlık sınıflarının her biri için optimal modellerin belirlenmesi, bu varlıkların kendine özgü karakteristiklerini ve risk faktörlerini daha iyi anlamamıza olanak tanır.

GARCH modelleri ailesinin farklı varlık sınıfları için farklı performans göstermesi, risk yönetimi stratejilerinin varlık sınıfına göre özelleştirilmesi gerektiğini vurgulamaktadır. Bu bulgu, "tek beden herkese uyar" yaklaşımının finansal modelleme ve risk yönetiminde yetersiz kalabileceğini göstermektedir. Örneğin, kripto para birimleri için EGARCH modellerinin daha uygun olması, bu varlık sınıfının asimetrik volatilite yapısını yansıtmaktadır. Bu tür bulgular, portföy yönetimi ve risk değerlendirmesi açısından önemli pratik uygulamalara sahiptir. Yatırımcılar ve fon yöneticileri, bu bilgileri kullanarak daha sofistike risk ölçüm teknikleri geliştirebilir ve varlık tahsis stratejilerini optimize edebilirler.

1. INTRODUCTION

A thorough understanding of data is crucial for anyone involved in computer science and for anyone who works with or manipulates raw data. The goal of data processing is to obtain novel insights or forecasts about future events by manipulating and analyzing actual information. Data analysts and data scientists primarily work with data, which serves as the essential cornerstone of their field.

The concept of "data" has undergone changes and developments throughout its history. The term was initially documented in 1946, referring to information that can be transmitted and stored for computer-related tasks. The term "data-processing" was first coined in 1954, while the term "data-base" (also known as "database"), which refers to a structured collection of data stored in a computer, came into being in 1962. The term "data-entry" was first used in 1970. Prior to its use in computer science, the term "data" originally denoted a fact that was provided or acknowledged in the 1640s. The term is taken from the Latin word "datum" which translates to "thing given" [1] [2].

Another vital factor to take into account is the way in which data is displayed. Data can be encoded using various characters, including letters, numbers, and symbols, or it can be represented in binary format at the machine level.

The ability to accurately comprehend and analyze data is highly valuable in the present era. Proficiency in analyzing unprocessed data and utilizing the resulting insights to make well-informed choices has become an essential competency. With the continuous advancement of technology and the exponential growth of data generation, the significance of efficient data comprehension and analysis will inevitably rise [3]. Therefore, individuals and organizations that are able to effectively utilize the potential of data will have a strong advantage in achieving success in the future.

1.1 A Brief History of Cryptocurrencies

The initial concepts for cryptocurrencies originated in the late 1980s. The primary objective was to create a decentralized form of currency that could be sent without the involvement of centralized institutions like banks, with a focus on ensuring transactions could not be traced. In 1995, David Chaum, an American cryptographer, introduced Digicash, a form of anonymous cryptographic electronic currency. This refers to an initial version of secure electronic payments that required specialized software for users to withdraw funds from a bank and mandated the usage of specific encrypted keys before transmitting the money to the intended recipient.

Next in line was Bit Gold, a creation of Nick Szabo in 1998, which is frequently referred to as the immediate precursor to Bitcoin [4]. Participation in the activity necessitated the allocation of a specific quantity of computational resources to resolve cryptographic challenges, with those who successfully solved the riddles being granted a reward. Upon combining the concepts of Chaum and Szabo, it becomes evident that we have a construct that has resemblance to contemporary Bitcoin. Nevertheless, we encountered a few technical issues prior to our adoption of BitCoin. Szabo was unable to resolve the issue of double spending, which involves duplicating and replicating digital data, in BitGold without the presence of a central governing body. A decade later, an enigmatic individual or collective operating under the alias Satoshi Nakamoto released a document titled “Bitcoin - The Peer-to-Peer Electronic Cash System” marking the inception of Bitcoin and other digital currencies [5].

Let us take a historical look at the process since this article was published.

1.1.1 The beginning (2008-2010)

The period spanning from the birth of blockchain technology to the introduction of Bitcoin’s original version holds great significance. In 2008, Satoshi Nakamoto published a white paper titled “Bitcoin: A Peer-to-Peer Electronic Cash System", which outlined the underlying principles of blockchain technology.

Nakamoto's proposal laid the groundwork for Bitcoin's infrastructure, which offers a decentralized digital currency and payment system powered by blockchain technology. This era is also characterized by the creation of the "genesis block" on January 3, 2009, marking the official launch of the Bitcoin network. Nakamoto's white paper elucidates the operational principles of a distributed digital currency (Bitcoin) and how it ensures a decentralized framework. The "genesis block," representing the birth of Bitcoin, serves as the initial block of the Bitcoin blockchain, establishing the foundation for subsequent blocks.

1.1.2 The emergence of altcoins (2011-2013)

From 2011 through 2013, a wave of alternative cryptocurrencies (altcoins) emerged, each with unique features and use cases. Inspired by Bitcoin's success, these altcoins aimed to revolutionize the cryptocurrency landscape with their offerings, such as faster transaction confirmations, decentralized domain name systems, energy efficiency, and facilitating international payments [6]. Notable examples of these altcoins include Litecoin (LTC), Namecoin (NMC), Peercoin (PPC), and Ripple (XRP).

Litecoin (LTC), Charlie Lee's brainchild in 2011, swiftly rose to prominence as a viable alternative to Bitcoin. Lee's vision was to create a cryptocurrency that could expedite block production using a different cryptographic algorithm called Scrypt [7]. The aim was to offer a more efficient payment system than Bitcoin, achieved through faster transaction confirmations.

Namecoin (NMC), launched in 2011 using Bitcoin's codebase, is designed to swap internet domain names and DNS records using Bitcoin's blockchain technology. By supporting a decentralized domain name system, Namecoin aims to make the internet more open and censorship-resistant [8].

Peercoin (PPC), created in 2012 by Sunny King and Scott Nadal, combines "proof-of-stake" (PoS) and "proof-of-work" (PoW) mechanisms to reduce energy consumption and create a more sustainable blockchain network [9]. While its PoS algorithm secures the blockchain network, the PoW algorithm verifies blocks [10].

Ripple (XRP) is designed to facilitate fast and low-cost international payments, especially among financial institutions [11]. Unlike a decentralized blockchain, Ripple is built on its consensus protocol called the Ripple Protocol Consensus Algorithm (RPCA) [12]. Consequently, Ripple's blockchain technology differs from Bitcoin and other cryptocurrencies and is governed by a central entity. Ripple was released in 2012 and quickly gained acceptance among banks and financial institutions [13].

These altcoins were introduced to the market with the aim of providing features and use cases different from Bitcoin. While Litecoin focuses on faster transaction confirmations, Namecoin offers a decentralized solution for internet domain names. Peercoin prioritizes energy efficiency, while Ripple is designed to facilitate international payments. These alternative cryptocurrencies served as a laboratory for exploring various use cases of blockchain technology and enriched the cryptocurrency ecosystem [14].

1.1.3 Expansion and experimentation (2014-2016)

From 2014 to 2016, there was a rise in the adoption of blockchain technology, and its potential uses in several industries were investigated. Financial institutions, in particular, recognized the potential benefits of blockchain technology for payment systems and data security, leading them to invest in projects within this realm [15]. This growing interest in blockchain technology was further fueled by the release of Ethereum, which facilitated the development of smart contracts and decentralized applications (DApps) [16].

In 2015, Ethereum, a platform for innovative applications like decentralized finance and tokenization, was launched by Vitalik Buterin. This marked a significant shift in the utility of blockchain technology, extending its use beyond simple cryptocurrency transactions [17]. The Ethereum Virtual Machine empowered developers to create and deploy smart contracts and self-executing agreements, written directly into code on the Ethereum blockchain [18]. This breakthrough attracted developers and entrepreneurs, sparking a wave of exploration into the potential of blockchain technology [19].

The launch of Ethereum in 2015 paved the way for the widespread adoption of smart contracts and the development of DApps, revolutionizing the use of blockchain technology [20]. A standout example during this period is the development of Quorum, an Ethereum-based blockchain platform by JP Morgan Chase. Quorum is tailored to provide a permissioned blockchain solution for enterprises, with a focus on privacy, security, and performance [21]. Its creation underscores the growing interest of financial institutions in harnessing blockchain technology for their operations and services [22].

During the period of expansion and experimentation from 2014 to 2016, we have witnessed significant growth in the adoption and exploration of blockchain technology across various sectors. The release of Ethereum and the development of smart contracts and DApps opened up new avenues for innovation and experimentation, setting the stage for further advancements in the blockchain ecosystem [23].

1.1.4 Mainstream adoption and regulatory challenges (2017-2019)

From 2017 to 2019, we witnessed a significant surge in the adoption of cryptocurrencies and blockchain technology within the mainstream financial landscape. This heightened interest was primarily driven by the highly volatile price fluctuations observed in Bitcoin, which attracted extensive media coverage and sparked public fascination with digital assets [24]. As a result, the market capitalization of cryptocurrencies experienced exponential growth, with Bitcoin reaching an all-time high of nearly 20,000 dolar in December 2017 [25].

Concurrently, significant corporations recognized the disruptive potential of blockchain technology and began investing heavily in projects aimed at harnessing its capabilities across various sectors [26]. For instance, IBM launched its blockchain platform in 2017, providing a cloud-based solution for businesses to develop and deploy blockchain applications. Similarly, Microsoft introduced its Azure Blockchain Workbench, a tool designed to streamline the development and deployment of blockchain applications on the Azure cloud platform.

However, as the adoption of cryptocurrencies and blockchain technology gained momentum, regulatory authorities worldwide faced a daunting challenge. They had to develop comprehensive frameworks to govern the burgeoning ecosystem, a task that was not without its complexities. The regulatory response to the rapid rise of cryptocurrencies varied significantly across jurisdictions, reflecting the diverse attitudes and approaches adopted by different nations [27].

Notably, some countries took a cautious stance, implementing stringent regulatory measures or even imposing outright bans on cryptocurrency activities. This was done to mitigate perceived risks such as financial instability and illicit activities. The regulatory action taken by South Korea in 2017, targeting cryptocurrency exchanges and mandating stricter know-your-customer and anti-money laundering procedures, is a clear example of this. It underscores the challenges faced by authorities in balancing the innovative potential of cryptocurrencies with the need to ensure financial stability and protect investors [28].

From 2017 to 2019, they marked a significant milestone in the mainstream adoption of cryptocurrencies and blockchain technology, accompanied by heightened regulatory scrutiny. As the ecosystem continued to evolve, stakeholders grappled with the complex interplay between innovation, regulation, and the transformative potential of these emerging technologies.

1.1.5 Institutional investment and enterprise adoption (2020-2021)

The period from 2020 to 2021 witnessed a remarkable shift in the perception and adoption of cryptocurrencies and blockchain technology by large institutional investors and corporations. Notably, companies like MicroStrategy, Tesla, and Square demonstrated their strategic foresight by allocating a portion of their corporate reserves to Bitcoin [29] [30]. This bold move not only signalled their confidence in cryptocurrencies but also served as a catalyst for broader institutional adoption, making the audience aware of the industry's evolving trends [31].

Moreover, the period from 2020 to 2021 saw a surge in the popularity of blockchain-based projects, particularly those focused on decentralized finance and tokenization platforms. These groundbreaking initiatives provided compelling solutions for diverse financial services, encompassing lending, borrowing, asset management, and trading, without reliance on conventional intermediaries. The rapid and significant increase in the total value locked in DeFi protocols, exceeding 80 billion dollars by May 2021, highlights the growing interest and acceptance of these decentralized financial solutions, arousing curiosity among the public [32].

However, as institutional giants began to embrace cryptocurrencies and blockchain technology, regulatory agencies intensified their scrutiny of the space. This heightened regulatory oversight led to several countries implementing stricter controls and regulations surrounding cryptocurrencies and blockchain technology [33]. The Financial Action Task Force recently issued revised recommendations for virtual asset service providers with the aim of improving transparency and addressing the threats of money laundering and terrorism financing in the cryptocurrency industry [34].

Furthermore, regulatory authorities in various jurisdictions imposed more stringent compliance requirements on cryptocurrency exchanges and custodial services to strengthen investor protection and maintain market integrity [35]. These measures included enhanced know-your-customer and anti-money laundering procedures, as well as mandatory registration and licensing for cryptocurrency businesses [36].

The period from 2020 to 2021 has represented a significant milestone in the institutional investment and enterprise adoption of cryptocurrencies and blockchain technology. While this increased adoption brought legitimacy and growth to the ecosystem, it also attracted heightened regulatory scrutiny, emphasizing the need for a balance between innovation and compliance in the rapidly evolving landscape of digital assets.

1.1.6 Current developments and future outlook (2022-Present)

In the current period, blockchain technology and cryptocurrencies continue to evolve and develop rapidly. However, regulatory frameworks play a crucial role in shaping the future trajectory of this innovative technology. Governments and regulatory bodies worldwide are actively developing regulations and standards for cryptocurrencies and blockchain projects, encompassing various aspects such as taxation, identity verification, anti-money laundering, and know-your-customer requirements [37].

The regulatory landscape for cryptocurrencies is a tapestry of diverse approaches adopted by different countries. Some nations have chosen to ban or restrict cryptocurrencies, directly impacting their prices, adoption, and operation. For example, China escalated its crackdown on cryptocurrencies in 2021, prohibiting cryptocurrency mining and trading activities within its borders. In contrast, other countries have taken a more proactive stance towards cryptocurrency adoption. In September 2021, El Salvador made a significant milestone by formally acknowledging Bitcoin as a legitimate form of currency, making it the pioneering nation to take this step [38]. This ruling has profound ramifications for the widespread recognition and utilization of cryptocurrencies as a means of transaction.

The evolving regulatory landscape has profound implications for major cryptocurrency exchanges and institutional investors. Compliance with regulatory requirements has become imperative, compelling these entities to allocate resources and efforts to ensure adherence to the ever-changing regulatory frameworks [39]. This increased focus on compliance has led to the development of more robust KYC and AML procedures within the cryptocurrency industry and the emergence of specialized compliance solutions and services [40].

Moreover, one of the most significant developments in the post-2022 era has been the growing mainstream acceptance and integration of cryptocurrencies into traditional financial systems.

Major financial institutions, such as JPMorgan Chase, Goldman Sachs and BlackRock, have increasingly embraced cryptocurrencies as viable investment assets, offering cryptocurrency-related products and services to their clients [41] .

In addition, the emergence of stablecoins, which are cryptocurrencies tied to stable assets such as the US dollar, has made it easier to incorporate cryptocurrencies into current financial systems. This allows for smooth transactions and decreases the potential for price fluctuations[8].

The future of cryptocurrencies and blockchain technology holds promise, but also carries an element of uncertainty. The ongoing development and maturation of the technology, along with increasing institutional adoption and regulatory clarity, are projected to fuel further growth and innovation in the space [42].

Nevertheless, in order to guarantee the enduring viability and extensive acceptance of blockchain technology, it is imperative to address obstacles such as scalability, interoperability, and energy consumption [43].

Te current developments in the cryptocurrency and blockchain space highlight the complex interplay between technological innovation, regulatory frameworks, and institutional adoption. As technology evolves, stakeholders must navigate this dynamic landscape, balancing the need for innovation with the imperative of regulatory compliance and risk management.

Although we will delve further into the technical aspects in the following sections, it is important to reiterate that Bitcoin and other cryptocurrencies rely on blockchain technology, which eliminates the necessity for a centralized authority.

1.2 A Brief Introduction to the Blockchain

Blockchain technology is a game-changer in the world of data management, providing a revolutionary decentralised and distributed ledger system. This cutting-edge architecture guarantees a secure, transparent, and unchangeable data storage system. It achieves this by using interconnected blocks, with each block containing distinct data and the cryptographic hash of the previous block. Here are the key characteristics of blockchain technology:

1. Elimination of central control: In contrast to conventional centralised databases, blockchain technology distributes data across a network, ensuring equal participation and access for all.

2. Trust and transparency: The cryptographic links and distributed nature create a secure framework where every participant possesses a complete system copy, which improves overall transparency.

3. Immutability and data integrity: The cryptographic links between blocks serve as a safeguard against any tampering with past data, guaranteeing the trustworthiness and consistency of the stored information.

4. Distributed consensus: The system utilises consensus mechanisms to add blocks and validate data, ensuring that all participants must agree on any changes.

5. Resilience: The distributed nature of the system ensures that data access remains intact, even if individual nodes experience failures.

These features make blockchain a strong alternative to traditional centralised databases, providing improved reliability, transparency, and resilience. Blockchain technology heavily relies on cryptographic principles, particularly one-way hash functions. These functions ensure that the system's rules are securely encoded within itself, preventing any tampering or fraudulent activities. They also encapsulate, through a mathematical procedure, the guidelines for generating new currency units in cryptocurrency systems.

The essential components of blockchain are:

1.3 Blocks

The each block in a blockchain consists of three crucial elements:

1. Block Data: This includes transaction information and other pertinent data.

2. Nonce: A 32-bit integer that is generated when a block is created. It plays a crucial role in calculating the hash of the block header.

3. Hash: A 256-bit value that is closely connected to the nonce, acting as a distinct identifier for the block and guaranteeing the integrity of the data.

1.4 Nodes

Nodes play a crucial role in upholding the integrity and decentralised structure of a blockchain network. They perform a crucial role in maintaining the system's integrity by enforcing established protocols. Nodes serve as the central repository for tokens, smart contracts, and transaction records, establishing themselves as the primary reference for the blockchain ledger.

1.5 Miners

Miners have a crucial role in the blockchain by adding new blocks through the process of mining. This task requires solving intricate mathematical problems to find a nonce that produces an authorised hash. After a successful mining operation, the new block is added to the chain, ensuring the continued growth and integrity of the blockchain. Ultimately, blockchain technology signifies a noteworthy progression in the realm of data management and transaction systems. With its decentralised nature and cryptographic solid principles, this system provides a secure and transparent alternative to traditional centralised systems.

With the rapid advancement of technology, its potential to transform numerous industries extends far beyond its original use in cryptocurrencies.

2. THEORETICAL FOUNDATION OF CRYPTOCURRENCIES

2.1 Cryptographic Foundations

Blockchain's dense cryptographic infrastructure and fundamental mechanism based on signed keys and digital signatures make it more secure and reliable.

It is important to note that while Bitcoin has faced security challenges in the past, such as instances where bitcoins were stolen from user balances, these issues were primarily due to the weak security protocols of certain cryptocurrency exchanges. However, the blockchain system itself, including its ledger, has never been intentionally hacked, highlighting the inherent security of the technology.

Thus, historically, the main goal of Bitcoin hackers has been to hack Bitcoin wallets or exchange platforms rather than to hack Bitcoin's cryptographic system. Let us take Bitcoin's cryptographic foundations as an example and see how they are used to protect the integrity of the blockchain. Bitcoin's cryptographic components consist of the following:

(1) **Secure Hash Algorithm (SHA-256):** It is a cryptographic hash method employed to maintain the integrity of data. The function transforms data of arbitrary magnitude into a constant-size (256-bit) digest. Summaries facilitate the identification of even the most subtle alteration in the data. Bitcoin employs the SHA-256 algorithm to guarantee the integrity of transactions and blocks [44].

(2) **Elliptic Curve Digital Signature Algorithm (ECDSA):** ECDSA is an algorithm that creates and verifies digital signatures using elliptic curve cryptography. Bitcoin uses ECDSA for users to sign transactions and prove ownership. Each Bitcoin address is associated with an ECDSA private key, which provides control of the bitcoins in that address [45].

2.2 Cryptographic Hash Functions

Hash functions are versatile tools utilized in nearly all security applications. A hash function is a mathematical function that transforms a given numerical value into another compressed numerical value.

The hash function can accept inputs of varying lengths, but it always produces outputs of a predetermined fixed length. Hash values, also known as hash values, are the values that a hash function returns.

Hash functions are commonly employed for the purpose of data concealment. As an illustration, passwords provided during website registration are transformed into hash values and subsequently stored in a database. By utilizing hash functions, users can maintain authentication while preventing anyone with access to the database from knowing the actual password supplied by the user. This is why hash functions consistently produce the same output for a given input.

Another application of hash functions is to generate a secure digest of any data in order to safeguard them from tampering. The hash code can be stored for summarization purposes, as the result will consistently have the same length, regardless of the length of the input data. Furthermore, any alteration of the original data would result in a modification of the hash summary.

The following are the universal characteristics of all hash functions:

Fixed-Length Output (Hash Value): A hash function is required to transform input data of any length into a hash value of a specified length. This procedure is commonly known as data hashing. Typically, the hash result is far smaller than the input. Hash functions are occasionally referred to as compression functions due of this rationale. An n-bit hash function is a hash function that produces an output of n bits. Commonly used hash functions generate outputs ranging from 160 to 512 bits.

Pre-Image Resistance: A hash function must possess computational irreversibility, which serves as a safeguard against attacks targeting the retrieval of the input value based on its hash value. Put simply, if a hash function generates a hash value, it is highly probable that finding the input value that resulted in that hash value is challenging.

Second Pre-Image Resistor: When provided with an entry and its corresponding Hash, it should be difficult to locate another entry with the same hash value. In order for a hash function to be considered secure, it should be computationally difficult to identify an alternative input value y that returns the same hash $h(y)$ as the original input x. This attribute of the hash function provides defense against a potential intruder who possesses an input-hash combination.

Collision Resistance: Collision resistance is an essential characteristic of cryptographic hash functions that assures the integrity and security of the data being hashed. Collision resistance is crucial in ensuring the durability and reliability of the blockchain ledger within the framework of blockchain technology [6].

A hash function is deemed collision-resistant when it is practically impossible to identify two distinct inputs that yield the same hash value by computational means [46]. In simple terms, if presented with a hash function h, it should be exceedingly challenging to locate two different inputs x and y for which $h(x) = h(y)$. The attribute in question is of utmost importance due to the potential consequences of an attacker being able to readily discover collisions. Such a capability may enable them to manipulate or fabricate transactions on the blockchain, so jeopardizing its integrity [47]. Mathematically, the attribute of collision resistance can be precisely defined as follows [48]: For a given hash function $h : \{0,1\}^* \rightarrow \{0,1\}^n$, it is extremely difficult to discover two different inputs $x, y \in 0, 1^*$ such that $h(x) = h(y)$.

In this context, $\{0,1\}^*$ denotes the collection of all conceivable binary sequences of varying lengths, while $\{0,1\}^n$ denotes the collection of binary sequences with a fixed length of n, where n represents the predetermined output size of the hash function.

The collision resistance property is achieved through the careful design of hash functions using complex mathematical algorithms. These algorithms take an input of arbitrary length and produce a fixed-size output, typically ranging from 128 to 512 bits, depending on the specific hash function [49].

Ultimately, collision resistance is an essential characteristic of hash functions that guarantees the safety and authenticity of blockchain technology. Hash functions like as SHA-256 ensure the integrity and reliability of the blockchain ledger by making it extremely difficult to identify collisions.

2.3 The SHA Hashing Algorithm

SHA, refers to a collection of hash algorithms that were created by the US National Security Agency (NSA). The initial hashing algorithm created in this sequence was SHA-0, which was developed in 1993. SHA-1 was introduced in 1995. The SHA-2 standard was created in 2004 in response to the discovery of security weaknesses in SHA-0 and SHA-1. The SHA-2 standard has various subsets, including SHA-224, SHA-256, SHA-384, and SHA-512. The Bitcoin network utilizes the SHA-256 hash algorithm. The number 256 in SHA-256 represents the number of bits. In its binary form, SHA-256 comprises of ones and zeros. However, reading it in binary can be challenging, therefore it is converted into hexadecimal to make it more accessible. Hexadecimal is a number system that has a base of 16 and includes the characters 0-9 and A-F.

The division of 256 bits by 8 yields a result of 32 bytes. Each computer character is represented by 4 bits, resulting in a hexadecimal value consisting of 64 characters. The SHA-256 standard algorithm consistently generates a 64-character output, irrespective of the input data's size, whether it be a single letter or a book spanning hundreds of pages.

For example, the hash code for the word "*bitcoin*" is

```
6b 88 c0 87 24 7a a2 f0 7e e1 c5 95 6b 8e 1a 9f 4c 7f
89 2a 70 e3 24 f1 bb 3d 16 1e 05 ca 10 7b
```

while the hash code for the word "*Bitcoin*" is

```
b4 05 6d f6 69 1f 8d c7 2e 56 30 2d da d3 45 d6 5f ea
d3 ea d9 29 96 09 a8 26 e2 34 4e b6 3a a4
```

The SHA-256 algorithm involves a series of specific steps. First, the input data is transformed into a specific format, and then logical operations are applied to these data blocks. These operations ensure that each part of the input influences the result independently.

Subsequently, the intermediate values obtained through these operations are transformed into a single fixed-length hash value.

The security of SHA-256 is contingent upon any slight alteration in the input data, which leads to an entirely distinct hash value. This characteristic indicates that any modification in the data will result in a corresponding modification in the hash value. Thus, within the Bitcoin network, SHA-256 is employed to generate the hash value of a block, which incorporates the hash code of the preceding block, establishing a strong connection between blocks. This guarantees the integrity of the blockchain, as any alteration in a block will change its hash value and thus impact the entire chain. This functionality enables the identification and preemptive mitigation of any alterations made to the blockchain.

The Bitcoin network utilizes the SHA-256 method to merge various block information, such as current user transactions, block number, 'nonce' value, and the hash code of the preceding block, into a unified hash code. SHA-256, or Secure Hash Algorithm 256-bit, is a cryptographic hash function that transforms a dataset, such as a message or file, into a hash result with a fixed length of 256 bits. The primary goal is to ensure the genuineness and security of the provided data by creating a unique representation.

The SHA-256 algorithm consists of a series of meticulous operations. At first, the incoming data is converted to a specific format, and then logical operations are applied to these data blocks. These processes ensure that each element of the input has a separate and distinct effect on the result.

Subsequently, the intermediate values obtained from these techniques are ultimately transformed into a solitary hash value with a predetermined length.

The security of SHA-256 relies on the feature that any slight modification in the input data will result in a completely different hash value. This feature signifies that any alteration made to the data will lead to a commensurate alteration in the hash value. Therefore, the Bitcoin network utilizes SHA-256 to produce the hash value of a block, which includes the hash code of the previous block, creating a robust link between blocks. The integrity of the blockchain is ensured by the fact that any modification made to a block will result in a change in its hash value, which in turn affects the entire chain. This feature allows for the detection and blocking of any alterations made to the blockchain.

The collision resistance of SHA-256 relies on the property that even a little alteration in the input data will produce a substantially distinct hash value. The phenomenon is referred to as the avalanche effect [50]. For instance, if a single bit in the input data is altered, approximately half of the output bits will, on average, be modified. This greatly impedes an attacker's ability to locate two distinct inputs that provide an identical hash value.

Moreover, the collision resistance of hash functions is frequently assessed through the utilization of the birthday attack [51]. The birthday attack is a stochastic technique used to estimate the minimum number of evaluations of a hash function needed to discover a collision with a specific probability. The birthday attack on a hash function with an output size of n bits necessitates around $2^{n/2}$ evaluations to discover a collision with a 50% likelihood. [52].

For SHA-256, the birthday assault would necessitate around 2^{128} assessments to discover a collision with a 50% likelihood, which is an exceedingly vast quantity. From a comparative standpoint, even if the collective computational power of all the computers worldwide were harnessed, it would require billions of years to discover a collision by the use of the birthday attack. [53].

2.4 Merkle Trees

A Merkle Tree is a data structure commonly employed in computer science apps. Merkel Trees are called the structures created to encode Blockchain data is the most efficient and secure. In other words, they are also called" Binary Hash Trees". [54].

Merkel Trees came into limelight recently as Bitcoin and other cryptocurrencies relies on the blockchain data which is a specific example of a Merkle tree. Merkle trees are synonymous with blockchain now, but their discovery was long before blockchains were created. It was discovered in 1987 by computer scientist Ralph Merkle, who suggested this system in his article "A Digital Signature Based on a Conventional Encryption Function" and was named after its creator [23]. Merkle was also the inventor of the cryptographic hashing system.

Merkle trees convert a large number of data into a string of characters that proves its accuracy without revealing the structure. This method securely stores all the data and allows access.

A Merkle tree is a structure of data that arranges data pairs by using a hashing function to construct tree leaves or nodes. Subsequently, the outcomes are consistently combined and subjected to hashing operations until a final hash, referred to as the Merkle root, is obtained. The Merkle root is the highest-level node in the tree, which includes all other nodes in the structure. Within this particular framework, the term “record" frequently denotes a “transaction". A Merkle tree transaction is a data packet that is linked to a leaf node through its hash. Blocks are the permanent transaction data entries that make up the leaves of the Merkle tree. The transactions that are now taking place form the individual components of a block, and they are linked together through the Merkle root. The Merkle tree’s leaf nodes store the hash of the block data, while the non-leaf nodes contain the cryptographic hash of their child nodes. Merkle trees effectively handle the entire record of every user’s transactions, guaranteeing the integrity of data and enabling quick verification of large datasets.

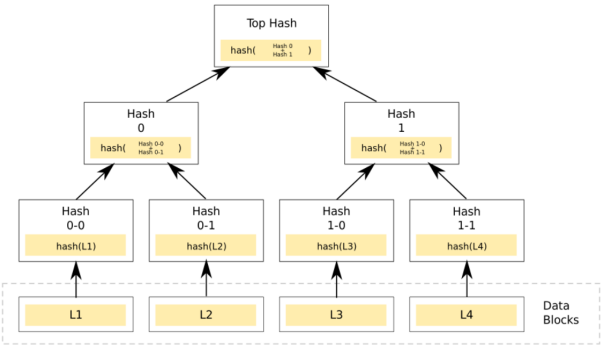

Figure 2.1 : Structure of Merkle Tree

To construct the tree depicted in Figure 2.1, we can begin with the four transaction data depicted by the boxes at the bottom of the diagram. Initially, hash references are generated for each individual transaction data (L1 to L4), and then these references are put together in pairs. Afterwards, hash references are formed for the pairs of hash references, namely Hash 0 (L12) and Hash 1 (L34). This process is iterated until we ultimately reach a solitary hash reference, known as the root of the Merkle tree (referred to as the "Top Hash").

The following are the primary characteristics of Merkle tree nodes:

Target Hash: The proof of work is generated by repeatedly hashing a single 80-byte block that contains a valid Merkle root derived from the transactions within the block. This Merkle root serves as a unique identifier for the block, enabling its precise location to be determined within the entire blockchain structure.

Block Header: The proof of work is generated through the continuous hashing of an 80-byte block that incorporates a valid Merkle root. This Merkle root is calculated based on all the transactions contained within the block, serving as a unique digital fingerprint. By including this Merkle root, the specific block can be efficiently identified and located within the overall blockchain structure, ensuring the integrity and traceability of the recorded transactions.

Block Time: The total time required to process the block starts when it is dispatched (off-block) and ends when it is received at its destination (on-block). The block time may vary depending on the routes taken within the communication network.

Hashed Time Lock Contract (HTLC): This is a sophisticated smart contract used in cryptocurrency channels to mitigate counterparty risk. It's a time-bound transaction enabler. In simpler terms, it allows a transaction only if the payment transaction acknowledgement is done by cryptographic proof within a specific timeframe. Otherwise, the transaction is not allowed. The HTLC is a complex but vital concept in cryptocurrency and blockchain technology, and grasping it will significantly enhance your understanding of these fields.

3. TIME SERIES

A time series is a sequential collection of data points that are measured at regular intervals over a period of time, and each data point represents the value that exists at a particular time t. Time series arise in several contexts such as engineering, economics and finance, environmental modelling, medicine, meteorology and hydrology.

In scientific, commercial, industrial, or economic fields, real-world data often show time series behavior, and predicting the future from these time series plays a vital role in decision-making processes based on past observations. In certain domains, such as financial time series, this forecasting task is often challenging because the data is noisy, non-stationary, and has a complex structure. Statistical and computational-based approaches are frequently used to capture time series patterns [55].

3.1 Time Series Analysis Objectives

The following points summarize the basic objectives of time-series analysis. Different goals may come to the fore in specific cases.

1. **Summary statistics:** Summarize the general characteristics of the data using statistical summaries such as mean, standard deviation, and variance.

2. **Graphs:** Visualize the change in time series using visualizations such as line charts, box plots, and histograms.

3. **Analysis and interpretation:** Create a model that explains the time dependence in the data. This model is usually a mathematical equation or a statistical method.

4. **Interpretation of the model:** Explain the trends, seasonalities, and other regularities in the data by looking at the coefficients of the created model.

5. **Forecasting:** Predict future values using the past of the data. This can be a single future value or a forecast of multiple values.

6. **Control:** Modify the control parameters in order to align the series more closely with a desired aim. Time series analysis can be employed to manage and regulate product quality in a manufacturing process, as an illustration.

7. **Correction of error terms:** In linear models, error terms can be correlated over time. In this case, the estimated variances must be corrected to take this into account.

3.2 Autocovariance

When we consistently examine a specific event at regular time intervals, it is probable that the obtained results will exhibit a correlation. Time-series analysis usually involves examining the autocovariance of a series of observations. Autocovariances measure the relationship between observations made at different time points and illuminate the temporal dependence structure of the data.

The autocorrelation function, denoted as ρ_t, is defined as

$$\rho_t = \text{corr}(Y_{t+\tau}, Y_\tau) = \frac{\gamma_t}{\gamma_0}. \tag{3.1}$$

The autocorrelation function identifies the second-order properties of the time series.

We forecast γ_t by c_t, and ρ_t by r_t, where

$$c_t = \frac{1}{n} \sum_{s=\max(1,1-t)}^{\min(n-t,n)} [Y_{s+t} - \overline{Y}][Y_s - \overline{Y}] \tag{3.2}$$

$$r_t = \frac{c_t}{c_0} \tag{3.3}$$

3.3 Stationarity

Time series analysis theory fundamentally rests upon the assumption of 'stationarity'. Real-world data frequently demonstrate non-stationary behavior, such as linear trends or seasonal effects. The stationarity assumptions outlined below are valid once any trends or seasonal components have been eliminated from the data [56].

The challenges posed by trends and seasonal effects will be addressed in subsequent discussions.

Suppose the series $(Y_t)_{t=0,\pm1,\pm2,\ldots}$ spans across time but is only observed at times $t = 1,\ldots,n$, resulting in the observed series $(Y_1,\ldots,Y_n)$. The theoretical properties pertain to the underlying process $(Y_t)_{t\in\mathbb{Z}}$. The process is considered weakly stationary or second-order stationary if it satisfies the condition for all integers t,τ.

$$\mathbb{E}(Y_t) = \mu \tag{3.4}$$

$$\operatorname{cov}(Y_{t+\tau}, Y_\tau) = \gamma_t \tag{3.5}$$

where μ is constant and γ_t does not depend on τ.

A process is considered strictly stationary or strongly stationary if the distributions of $(Y_{t_1},\ldots,Y_{t_k})$ and $(Y_{t_1+\tau},\ldots,Y_{t_k+\tau})$ are same for all sets of time points $t_1,\ldots,t_k$ and for all values of $\tau \in \mathbb{Z}$.

A strictly stationary process intrinsically fulfills the requirements for weak stationarity. However, the converse does not typically apply. Put simply, a process that is poorly stationary may not always be strictly stationary. In the case where the process adheres to a Gaussian distribution, the joint distribution of $(Y_{t_1},\ldots,Y_{t_k})$ is multivariate normal for all time points $t_1,\ldots,t_k$.

When encountering such situations, the concept of weak stationarity implies the concept of strong stationarity, thus creating a relationship between the two types of stationarity for Gaussian processes.

It should be noted that the variance of Y_t is equal to γ_0 , and due to stationarity, γ_{-t} is equal to γ_t. The autocovariance function is denoted by the sequence (γ_t).

3.4 Models of Stationary Processes

Consider a time series devoid of trends or seasonal effects, where any such patterns have been effectively eliminated from the data.

One common approach in constructing a linear model for such a time series exhibiting autocorrelation involves utilizing autoregressive (AR) or moving average (MA) models. These models capture the dependencies between observations at different time points, allowing for the incorporation of autocorrelation effects into the model.

The process (Y_t) is referred to as a linear process if it can be expressed in the following form.

$$Y_t = \mu + \sum_{r=-\infty}^{\infty} c_r \varepsilon_{t-r} \tag{3.6}$$

where μ is a common mean, $\{c_r\}$ is a sequence of fixed constants and $\{\varepsilon_t\}$ are independent random variables with mean 0 and common variance. We assume $\sum c_r^2 < \infty$ to ensure that the variance of Y_t is finite.

The process (Y_t) is referred a linear process if it can be represented in the following manner. The equation is defined as follows:

$$Y_t = \mu + \sum_{r=-\infty}^{\infty} c_r \varepsilon_{t-r} \tag{3.7}$$

In this equation, μ represents the common mean, $\{c_r\}$ is a sequence of fixed constants, and $\{\varepsilon_t\}$ are independent random variables with a mean of 0 and a common variance. To guarantee that the variance of Y_t is finite, we make the assumption that $\sum c_r^2 < \infty$.

The AR, MA, and ARMA processes we will discuss are all specific instances of fundamental linear processes [56].

3.4.1 Autoregressive processes

Suppose that the current value of the series is directly influenced by its prior value, with some degree of error. This can be described as

$$Y_t = \alpha Y_{t-1} + \varepsilon_t \tag{3.8}$$

where ε_t is a white noise time series. In other words, ε_t represents a sequence of uncorrelated random variables (which may be normally distributed, though not necessarily so) with a mean of 0 and a variance of σ^2. This model is referred to as an autoregressive (AR) model, given that Y is regressed on its own past values. Specifically, this is an AR model with a lag of 1.

In general, in an autoregressive model, a $p-th$ order autoregressive model, known as an AR(p) model, is defined as follows:

$$Y_t = \sum_{i=1}^{p} \alpha_i Y_{t-i} + \varepsilon_t. \tag{3.9}$$

If ε_t has variance σ^2, then from independence we have that

$$\text{Var}(Y_t) = \sigma^2 + \alpha^2\sigma^2 + \cdots + \alpha^{2(k-1)}\sigma^2 + \alpha^{2k}\text{Var}(Y_{t-k}) \tag{3.10}$$

The summation converges under the assumption of finite variance. However, it converges only when the absolute value of α is less than 1. Therefore,the condition $|\alpha| < 1$ is necessary for the AR(1) process to be stationary.

3.4.2 Moving average processes

An alternative approach is to presume that the present value of the series can be expressed as a weighted sum of previous white noise terms. For instance,

$$Y_t = \varepsilon_t + \beta\varepsilon_{t-1}. \tag{3.11}$$

The model is referred to as a moving average (MA) model because the variable Y is a weighted average of past values from the white noise series. In this case, the moving average is delayed by 1 unit of time. The white noise series can be understood as innovations or shocks, which refer to new and independent information that arises at each time interval. These innovations are then joined with other innovations to produce the observable series Y.

Within a broader framework, we can examine a moving average model with an order q, referred to as an MA(q) model. This model is described by

$$Y_t = \varepsilon_t + \sum_{j=1}^{q} \beta_j\varepsilon_{t-j}. \tag{3.12}$$

If ε_t has variance σ^2, then from independence we have that

$$\text{Var}(Y_t) = \sigma^2 + \sum_{j=1}^{q} \beta_j^2\sigma^2. \tag{3.13}$$

3.4.3 An example: MA(1)

We consider the MA(1) procedure: The formula for Y_t is $\varepsilon_t + \beta\varepsilon_{t-1}$. Given that ε_t has a variance of σ^2 and a mean of zero, we can compute the autocovariances as follows:

$$\gamma_0 = \text{Var}(Y_0) = (1+\beta^2)\sigma^2 \tag{3.14}$$

Thus,

$$\rho_0 = 1, \quad \rho_1 = \frac{\beta}{1+\beta^2}, \quad \rho_k = 0 \text{ for } k > 2 \tag{3.15}$$

are the autocorrelations.

Now evaluate the same procedure, substituting $1/\beta$ for β. It is evident from the preceding formulae that this transformation does not alter the autocorrelation function, making it impossible to discriminate between the two processes denoted by β and $1/\beta$.

It is typical to enforce the following requirement in order to resolve this identifiability issue: In the complex plane, all of the zeros of the function $\phi_\beta(z)$ are located outside of the unit circle.

3.4.4 The backshift operator

The definition of the backshift operator or lag operator L is:

$$LY_t = Y_{t-1} \tag{3.16}$$

$$L^2 Y_t = L(LY_t) = Y_{t-2} \tag{3.17}$$

The identity operator $IY_t = L^0 Y_t = Y_t$ is included.

The AR(p) process $Y_t = \sum_{i=1}^{p} \alpha_i Y_{t-i} + \varepsilon_t$ may be expressed as follows using this notation:

$$\left(I - \sum_{i=1}^{p} \alpha_i L^i \right) Y_t = \varepsilon_t \text{ or } \phi_\alpha(L) Y_t = \varepsilon \tag{3.18}$$

Remember that $Y_t = \varepsilon_t + \sum_{j=1}^{q} \beta_j \varepsilon_{t-j}$ represents an MA(q) process. Any complex number z can be represented by the moving average polynomial as follows:

$$\phi_\beta(z) = 1 + \beta_1 z + \cdots + \beta_q z^q \tag{3.19}$$

The MA(q) process may then be expressed in operator notation as follows:

$$Y_t = \left(I + \sum_{j=1}^{q} \beta_j L^j \right) \varepsilon_t \text{ or } Y_t = \phi_\beta(L) \varepsilon \tag{3.20}$$

There is no requirement to impose a stationarity constraint on the coefficients $\{\beta_j\}$ for an MA(q) process. But there is another problem that requires limitation on these coefficients.

Let us consider the special case when $\beta_0 = 1$. This may be expressed as follows:

$$\phi_\alpha(L)Y_t = \phi_\beta(L)\varepsilon_t \tag{3.21}$$

The prerequisites are as follows:

(i) The requirement for stationarity on $\{\alpha_1, \ldots, \alpha_p\}$

(ii) The condition of identifiability on $\{\beta_1, \ldots, \beta_q\}$

(iii) An extra requirement for identifiability is the absence of shared roots between $\phi_\alpha(z)$ and $\phi_\beta(z)$.

To prevent an ARMA(p,q) model that may be represented as a lower order model, such as an ARMA$(p-1,q-1)$ model, condition (iii) is required [57].

3.4.5 ARMA processes

An autoregressive moving average process, denoted as ARMA(p,q), can be represented by the following equation:

$$Y_t = \sum_{i=1}^{p} \alpha_i Y_{t-i} + \sum_{j=0}^{q} \beta_j \varepsilon_{t-j} \tag{3.22}$$

for $\beta_0 = 1$.

A more comprehensive definition of an ARMA process includes a non-zero mean μ, which is achieved by substituting Y_t with $Y_t - \mu$ and Y_{t-i} with $Y_{t-i} - \mu$ in the equation above.

Based on the definition, it is evident that an MA(q) process is second-order stationary for any given values of $\beta_1, \ldots, \beta_q$. However, the same cannot be said for AR(p) and ARMA(p,q) models, as they do not always yield second-order stationary time series.

As an example, we have previously discussed the requirement of $|\alpha| < 1$ for an AR(1) model to be stationary. This condition is essential for all AR processes of higher orders as well.

Definition of the autoregressive polynomial for any complex number z as:

$$\phi_\alpha(z) = 1 - \alpha_1 z - \cdots - \alpha_p z^p \tag{3.23}$$

In order for an autoregressive process of order p (AR(p)) to be stationary, all the zeros of the function $\phi_\alpha(z)$ must lie outside the unit circle in the complex plane. The condition imposed on the coefficients $\{\alpha_1, \ldots, \alpha_p\}$ ensures that the process is both well-defined and stationary [58].

3.4.6 Differencing

The difference operator ∇ is defined as the subtraction of the current value Y_t from the previous value Y_{t-1}, expressed as

$$\nabla Y_t = Y_t - Y_{t-1} \tag{3.24}$$

The disparities give rise to a new time series ∇Y (which is of length $n-1$ if the previous series had a length of n).

Similarly,

$$\nabla^2 Y_t = \nabla(\nabla Y_t) = Y_t - 2Y_{t-1} + Y_{t-2} \tag{3.25}$$

and so on.

If our initial time series is not stationary, we can analyze the first-order difference process ∇Y, or higher-order differences $\nabla^d Y$. If we determine that a differenced process exhibits stationarity, we can then pursue an ARMA model for that differenced process. Typically, when differencing is used, a value of $d = 1$ or at most $d = 2$ is usually adequate.

3.4.7 ARIMA processes

In the event that the original time series fails to exhibit stationarity, one approach is to examine the first order difference process ∇Y, or higher order differences such as $\nabla^2 Y$, and so forth. If it is determined that a differenced process displays stationarity, then an ARMA model can be fitted to that differenced process.

From a practical standpoint, if differencing is employed, it is usually sufficient to use $d = 1$, or in some cases $d = 2$. The process Y_t is defined as an autoregressive integrated moving average process, denoted as ARIMA(p,d,q), if the dth difference $\nabla^d Y$ follows an ARMA(p,q) process.

An ARIMA(p,d,q) model can be expressed as follows:

$$\phi_\alpha(L)\nabla^d Y = \phi_\beta(L)\varepsilon \tag{3.26}$$

$$\phi_\alpha(L)(I-L)^d Y = \phi_\beta(L)\varepsilon \tag{3.27}$$

3.4.8 Calculating Autocorrelation Function

The autocorrelation function (ACF) is calculated by dividing the k-th lag autocovariance, denoted as γ_k, by the zero-lag autocovariance, denoted as γ_0. This is expressed mathematically as $\rho_k = \gamma_k/\gamma_0$. It is important to mention that the autocorrelation function becomes zero when the absolute value of k is greater than q. The sudden termination of the autocorrelation function after lag q is a characteristic trait of the moving average (MA) process and can be employed to determine the order of an MA process.

3.4.9 Calculating the Partial Autocorrelation Function

In practice, the *Partial Autocorrelation Function* (PACF) is computed as follows. Consider the regression of Y_t on $Y_{t-1},\ldots,Y_{t-k}$, i.e., the model:

$$Y_t = \sum_{j=1}^{k} a_{j,k} Y_{t-j} + \varepsilon_t \tag{3.28}$$

where ε_t is independent of $Y_1,\ldots,Y_{t-1}$. Given the data $Y_1,\ldots,Y_n$, least squares estimates of $a_{1,k},\ldots,a_{k,k}$ are obtained by minimizing:

$$\sigma_k^2 = \frac{1}{n}\sum_{t=k+1}^{n}\left(Y_t - \sum_{j=1}^{k} a_{j,k} Y_{t-j}\right)^2 \tag{3.29}$$

These $a_{j,k}$ coefficients can be found recursively in k for $k = 0,1,2,\ldots$. For $k = 0$: $\sigma_0^2 = c_0$, $a_{0,0} = 0$, and $a_{1,1} = \rho(1)$. Then, given the $a_{j,k-1}$ values, the $a_{j,k}$ values are determined as follows:

$$a_{k,k} = \frac{\rho_k - \sum_{j=1}^{k-1} a_{j,k-1}\rho_{k-j}}{1 - \sum_{j=1}^{k-1} a_{j,k-1}\rho_j} \tag{3.30}$$

$$a_{j,k} = a_{j,k-1} - a_{k,k}a_{k-j,k-1} \tag{3.31}$$

where $j = 1,\ldots,k-1$. And subsequently:

$$\sigma_k^2 = \sigma_{k-1}^2(1 - a_{k,k}^2) \tag{3.32}$$

This method is commonly referred to as the Levinson-Durbin recursion. The $a_{k,k}$ value corresponds to the kth sample partial correlation coefficient. When considering a Gaussian process, we can interpret it in the following way:

$$a_{k,k} = \text{corr}(Y_t, Y_{t-k} \mid Y_{t-1}, \ldots, Y_{t-k+1}) \tag{3.33}$$

If the process Y_t follows an AR(p) model, then $a_{k,k} = 0$ for $k > p$. Thus, the PACF plot should display a significant decrease to almost zero following lag p, which can be used as a diagnostic tool to identify an AR(p) process.

3.4.10 Identifying an AR(p) Process:

In an AR(p) process, the ρ_k values decrease gradually as k increases, which can be difficult to identify in a plot of the acf. On the other hand, the diagnostic tool used to identify an AR(p) process relies on a different measure known as the partial autocorrelation function (PACF). The partial autocorrelation at lag k is calculated by examining the correlation between Y_t and Y_{t-k} while taking into account the influence of the intermediate variables $Y_{t-1}, \ldots, Y_{t-k+1}$.

We construct these partial autocorrelations by fitting autoregressive processes of increasing orders. At each stage, we define the partial autocorrelation coefficient as the estimate of the final autoregressive coefficient.

Put simply, a_k is the value that approximates α_k in an AR(k) process. If the underlying process follows an AR(p) model, then it is known that α_k is equal to zero for k greater than p. Thus, the PACF plot should display a clear cutoff point after the lag p.

One way to construct the partial autocorrelation function (PACF) is by using the sample analogues of the Yule-Walker equations for an autoregressive (AR) process with order p:

$$\rho_k = \sum_{i=1}^{p} \alpha_i \rho_{|k-i|}, \quad k = 1, \ldots, p \tag{3.34}$$

Replacing ρ_k with its sample value r_k, we obtain the sample analogue of these equations:

$$r_k = \sum_{i=1}^{p} a_{i,p} r_{|k-i|}, \quad k = 1, \ldots, p \tag{3.35}$$

In this context, the notation $a_{i,p}$ is employed to highlight the estimation of the autoregressive coefficients $\alpha_1, \ldots, \alpha_p$ while assuming that the underlying process follows an autoregressive pattern of order p. These equations can be solved, resulting in a set of unknowns that can be represented as $a_{1,p}, \ldots, a_{p,p}$. The pth partial autocorrelation coefficient is denoted as $a_{p,p}$.

3.4.11 Identifying an MA(q) process:

As previously mentioned, in an MA(q) time series, the autocorrelation function values beyond lag q are all zero, meaning that $\rho_k = 0$ for $k > q$. Hence, the plots of the autocorrelation function (ACF) should display a pronounced decline to almost zero following the qth coefficient. This tool functions as a diagnostic method for recognizing an MA(q) process.

3.4.12 Second order properties of MA(q)

The MA(q) process is defined as $Y_t = \sum_{j=0}^{q} \beta_j \varepsilon_{t-j}$, when $\beta_0 = 1$. It is clear that the expected value of Y_t is equal to zero for all values of t. Therefore, for values of k greater than zero, the autocovariance function is obtained in the following manner:

$$\begin{aligned} \gamma_k &= E(Y_t Y_{t-k}) \\ &= E\left[\left(\sum_{j=0}^{q} \beta_j \varepsilon_{t-j}\right)\left(\sum_{i=0}^{q} \beta_i \varepsilon_{t-k-i}\right)\right] \\ &= \sum_{j=0}^{q}\sum_{i=0}^{q} \beta_j \beta_i E(\varepsilon_{t-j}\varepsilon_{t-k-i}) \end{aligned} \tag{3.36}$$

Given that the sequence ε_t is white noise, it follows that $E(\varepsilon_{t-j}\varepsilon_{t-k-i}) = 0$ unless $j = i + k$. Hence, the only non-zero terms in the summation are of the form $\sigma^2 \beta_i \beta_{i+k}$ and we obtain:

$$\gamma_k = \begin{cases} \sigma^2 \sum_{i=0}^{q-|k|} \beta_i \beta_{i+|k|} & |k| \leq q \\ 0 & |k| > q \end{cases} \tag{3.37}$$

3.4.13 Second order properties of AR(p)

Let us examine the autoregressive process of order p, denoted as AR(p):

$$Y_t = \sum_{i=1}^{p} \alpha_i Y_{t-i} + \varepsilon_t \tag{3.38}$$

In this particular model, the expected value of Y at time t is equal to zero. As a result, when we multiply both sides of the equation above by Y_{t-k} and calculate the expected value, we get:

$$\gamma_k = \sum_{i=1}^{p} \alpha_i \gamma_{k-i}, \quad k > 0 \tag{3.39}$$

Expressing this in terms of the autocorrelations $\rho_k = \gamma_k / \gamma_0$, we have:

$$\rho_k = \sum_{i=1}^{p} \alpha_i \rho_{k-i}, \quad k > 0 \tag{3.40}$$

The Yule-Walker equations are the name given to these equations. The population autocorrelations, denoted as ρ_k, are computed by solving the Yule-Walker equations. All of these autocorrelations are frequently nonzero.

The Yule-Walker equations are of interest to us because they can be used to determine the ρ_k values, given that the α_i coefficients are known. Later, we will be interested in utilizing them to deduce the values of α_i that correspond to a certain collection of observed sample autocorrelation coefficients.

3.5 Non-linear Models

Various specialized models and methodologies have been developed to analyze financial time series, such as share prices, share price indices, spot interest rates, and currency exchange rates. These include ARCH (Autoregressive Conditionally Heteroscedastic) models and Stochastic Volatility models.

3.5.1 Stochastic volatility

An option to ARCH-type models is to incorporate unobserved components that affect the volatility σ_t^2.

The log-normal stochastic volatility model can be expressed as follows:

$$y_t = \exp(h_t/2) \tag{3.41}$$

$$h_{t+1} = \gamma_0 + \gamma_1 h_t + \eta_t \tag{3.42}$$

where $\varepsilon_t \sim \mathcal{N}(0,1)$ and $\eta_t \sim \mathcal{N}(0,\sigma_\eta^2)$ are independent for all t.

The process h_t exhibits strong stationarity if and only if the absolute value of γ_1 is less than 1. Additionally, if h_t is stationary, then y_t is also stationary. Monte-Carlo techniques are frequently used for estimation because the distribution of $\log \varepsilon_t^2$ is non-normal.

3.5.2 (G)ARCH models

The most basic ARCH model, referred to as ARCH(1), is defined as follows:

$$y_t = \sigma_t \varepsilon_t \text{ where } \sigma_t = \alpha_0 + \alpha_1 y_{t-1}^2$$

The variable ε_t follows a normal distribution with mean 0 and standard deviation 1. The sequence of ε_t values is independent. Here, the value of α_1 is set to be greater than zero in order to prevent negative variances. The conditional distribution of Y_t given $Y_{t-1} = y_{t-1}$ is $\mathcal{N}(0, \alpha_0 + \alpha_1 y_{t-1}^2)$.

(i) The GARCH (Generalized ARCH) model is an expanded version that incorporates moving average components to account for variations in variances. For instance, the GARCH(1,1) model:

$$y_t = \sigma_t \varepsilon_t \text{ where } \sigma_t^2 = \alpha_0 + \alpha_1 y_{t-1}^2 + \beta_1 \sigma_{t-1}^2 \tag{3.43}$$

(ii) The EGARCH (Exponential GARCH) model represents the logarithm of σ_t^2 as a function of the size and sign of ε_{t-1}.

Nelson (1991) proposed the exponential GARCH (EGARCH) model.

$$\log \sigma_t^2 = \omega + \sum_{i=1}^{p} g(Z_{t-i}) + \sum_{j=1}^{q} \beta_j \log \sigma_{t-j}^2 \tag{3.44}$$

Where,

$$g(Z_{t-i}) = \gamma_i Z_{t-i} + \alpha_i (|Z_{t-i}| - E(|Z_{t-i}|)) \tag{3.45}$$

Define $Z_{t-1} = \frac{\varepsilon_{t-1}}{\sigma_{t-1}}$ and the natural logarithm of the conditional variance equals to:

$$\log(\sigma_t^2) = \omega + \sum_{j=1}^{q} \beta_j \log \sigma_{t-j}^2 + \sum_{i=1}^{p} \alpha_i \left(\left| \frac{\varepsilon_{t-i}}{\sigma_{t-i}} \right| - E\left(\left| \frac{\varepsilon_{t-i}}{\sigma_{t-i}} \right| \right) \right) + \sum_{i=1}^{p} \gamma_i \frac{\varepsilon_{t-i}}{\sigma_{t-i}} \tag{3.46}$$

Alexander (2004) presented α represents the symmetric effect, β measures the lagged conditional variance and γ reflects the asymmetric performance.

$$E(|Z_{t-1}|) = \begin{cases} \sqrt{\frac{2}{\pi}}, & \text{when } Z_{t-1} \text{ is normal distribution} \\ \frac{2\Gamma(\frac{\nu+1}{2})}{\sqrt{\nu\pi}\Gamma(\frac{\nu}{2})}, & \text{when } Z_{t-1} \text{ is student-t distribution} \end{cases} \tag{3.47}$$

In their study, Wang, Fawson, Barrett, and McDonald (2001) show that $E(|Z_{t-1}|)$ is constant for all i when Z_t is normal distribution or is $\frac{2\Gamma(\frac{\nu+1}{2})}{\sqrt{\nu\pi}\Gamma(\frac{\nu}{2})}$ depended on different ν when Z_t is student-t distribution.

(iii) The GJR-GARCH model, proposed by Glosten, Jagannathan and Runkle (1993), is another asymmetric model that captures the leverage effect in financial time series. Define the sequence ε_t as $\varepsilon_t = z_t \sigma_t$, where z_t follows a normal distribution.

$$\varepsilon_t \sim \mathcal{N}(0, \sigma_t^2) \tag{3.48}$$

The GJR-GARCH model is written as:

$$\sigma_t^2 = \omega + \sum_{j=1}^{p} \beta_j \sigma_{t-j}^2 + \sum_{i=1}^{q} \alpha_i \varepsilon_{t-i}^2 + \sum_{k=1}^{r} \gamma_k \varepsilon_{t-k}^2 I_{t-k} \tag{3.49}$$

Where I_t is an indicator function defined as:

$$I_t = \begin{cases} 1, & \text{if } \varepsilon_t < 0 \\ 0, & \text{otherwise} \end{cases} \tag{3.50}$$

The asymmetry in the GJR-GARCH model is captured by the sign of the indicator term. When the residual (ε_t) is less than zero, the indicator term (I_t) is set to one; otherwise, it is set to zero. This enables the model to differentiate between positive and negative shocks, thus taking into consideration the leverage effect commonly observed in financial markets. Patrick, Stewart and Chris (2006) offer a comprehensive explanation of this model in their article.

4. STATISTICAL ANALYSIS AND MODEL FITTING

4.1 The Error Term's Distribution

In our study, this section mostly presents two distributions. The first distribution is a normal distribution, while the second one is a student-t distribution.

4.1.1 Normal distribution (Gaussian distribution)

The normal distribution, also known as the Gaussian distribution, is a fundamental continuous probability distribution in mathematics and statistics. It is characterized by two parameters: the mean (μ) and the standard deviation (σ). The probability density function (PDF) of the normal distribution is given by:

$$f(x|\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{4.1}$$

where x is the random variable, $\mu \in \mathbb{R}$ is the mean, and $\sigma > 0$ is the standard deviation. The distribution is symmetric about its mean and has a characteristic bell-shaped curve [59]. Key properties of the normal distribution include:

(i) The mean, median, and mode are all equal to μ.

(ii) Approximately 68%, 95%, and 99.7% of the data fall within one, two, and three standard deviations of the mean, respectively (the empirical rule).

(iii) It is the maximum entropy probability distribution for a random variable with specified mean and variance.

The standard normal distribution, where $\mu = 0$ and $\sigma = 1$, is particularly important in probability theory and statistics. Its cumulative distribution function is denoted by $\Phi(x)$ and cannot be expressed in terms of elementary functions.

4.1.2 Student's t-distribution

The Student's t-distribution, introduced by William Sealy Gosset under the pseudonym "Studen", is a continuous probability distribution that arises in the problem of estimating the mean of a normally distributed population when the sample size is small and the population standard deviation is unknown. The probability density function of the t-distribution with $\nu > 0$ degrees of freedom is given by:

$$f(t|\nu) = \frac{\Gamma(\frac{\nu+1}{2})}{\sqrt{\nu\pi}\Gamma(\frac{\nu}{2})}\left(1+\frac{t^2}{\nu}\right)^{-\frac{\nu+1}{2}} \tag{4.2}$$

where t is the random variable, ν is the degrees of freedom, and Γ is the gamma function. The Student t-distribution possesses several important characteristics:

The distribution is symmetrical about zero and has a bell-shaped curve, however it has fatter tails compared to the normal distribution.As $\nu \to \infty$, the t-distribution converges to the standard normal distribution. For $\nu > 1$, the mean exists and is equal to 0. For $\nu > 2$, the variance exists and is equal to $\frac{\nu}{\nu-2}$. It arises as the distribution of a ratio of a standard normal random variable and the square root of a scaled chi-squared random variable.

The t-distribution is essential in statistical inference, specifically in hypothesis testing and constructing confidence intervals for situations with small sample sizes or unknown population standard deviation. It offers a more conservative approximation in comparison to the normal distribution, taking into consideration the extra uncertainty caused by estimating the population standard deviation from the sample [60].

4.2 Model Performance Metrics

Several metrics are utilized to assess the effectiveness of models. In this section, we will discuss some commonly used metrics, including their formulas and threshold values that indicate a successful model in our study.

4.2.1 Mean absolute error (MAE)

Mean Absolute Error (MAE) is a metric that computes the average of the absolute discrepancies between the projected values and the actual values. The formula is as follows: [61]

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{4.3}$$

In this context, y_i denotes the true values, $\hat{y}_i$ denotes the estimated values, and n represents the overall number of observations. A model's success is directly proportional to the decrease in the Mean Absolute Error (MAE) value. A model is deemed to have good performance if its Mean Absolute Error (MAE) value is less than 10% of the range of the target variable.

4.2.2 Mean squared error (MSE)

The Mean Squared Error (MSE) is a mathematical calculation that determines the average of the squared differences between the predicted values and the actual values. The formula can be expressed in the following way:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{4.4}$$

Achieving a lower Mean Squared Error (MSE) number indicates a higher level of success for the model. A model is deemed to have good performance if its Mean Squared Error (MSE) value is less than 5% of the variance of the target variable [62].

4.2.3 Root mean squared error (RMSE)

RMSE is obtained by taking the square root of the MSE. Its formula is as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{4.5}$$

RMSE has the same units as MSE, and the lower the value, the more successful the model is. A model with an RMSE value less than percentage of 10 of the range of the target variable is considered to have good performance [63].

4.2.4 R^2 score

Understanding the R^2 score is crucial for assessing how well the model can explain the variance in the dependent variable. The formula can be expressed as:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}i)^2}{\sum i = 1^n (y_i - \bar{y})^2} \tag{4.6}$$

Here, $\bar{y}$ denotes the average of the actual values. The R^2 score falls within the range of 0 to 1. Values near 1 suggest that the model effectively explains the data. A model that achieves an R^2 score higher than 0.7 is regarded as having excellent performance. [64].

4.2.5 Root mean squared logarithmic error (RMSLE)

RMSLE is a variant of RMSE that incorporates a logarithmic transformation. The formula is as follows:

$$RMSLE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\log(y_i + 1) - \log(\hat{y}_i + 1))^2} \tag{4.7}$$

RMSLE is particularly useful for keeping the errors of large values on the same scale as the errors of small values. A model with an RMSLE value less than 0.2 is considered to have good performance. In general, for a model to be considered successful, it is expected that the above metrics have low values (high values for R^2) and that the model captures the patterns in the data set well. Additionally, it is important that the model does not experience over-fitting or under-fitting problems.

4.2.6 The akaike information criterion (AIC)

AIC, or Akaike Information Criterion, is a quantitative measure applied for the purpose of selecting the most appropriate model [65]. The objective is to choose the most suitable model by striking a balance between the model's quality of fit and The AIC formula can be expressed as:

$$AIC = 2k - 2\ln(L) \tag{4.8}$$

Here, k denotes the amount of parameters in the model, while L represents the maximum likelihood value of the model. Models with lower AIC values are favored. A model that has an AIC value at least two units lower than the next best model is considered to have strong support [66].

4.2.7 The bayesian information criterion (BIC)

The Bayesian Information Criterion (BIC) is a commonly employed quantitative criterion for selecting models. Like AIC, BIC seeks to identify the most suitable model by taking into account both the model's goodness of fit and its complexity. Nevertheless, the Bayesian Information Criterion (BIC) exhibits a stronger inclination towards penalizing intricate model complexity compared to the Akaike Information Criterion (AIC), particularly in situations where the sample size is substantial. The BIC formula can be mathematically represented as:

$$\mathrm{BIC} = k\ln(n) - 2\ln(L) \tag{4.9}$$

In this scenario, k symbolizes the count of parameters in the model, n refers to the size of the sample, and L represents the maximum likelihood value of the model. Models with lower Bayesian Information Criterion (BIC) values are preferable, as they imply a superior trade-off between model fit and complexity. When doing a comparison of models using the Bayesian Information Criterion (BIC), a difference of 2 or more in BIC values is regarded as compelling evidence in support of the model with the lower BIC. A disparity of 6 or greater is regarded as compelling evidence, while a disparity of 10 or greater is regarded as highly compelling evidence [67]. A key benefit of BIC in comparison to AIC is its consistency. As the size of the sample increases, the likelihood of selecting the correct model (assuming it is one of the models being considered) approaches 1. On the other hand, the Akaike Information Criterion (AIC) lacks consistency and has a tendency to choose excessively intricate models as the sample size increases [68].

Nonetheless, it is crucial to acknowledge that both AIC and BIC are comparative metrics and should only be employed for model comparison inside a given dataset.

They do not offer a definitive assessment of the quality or suitability of the model. Furthermore, the selection between AIC and BIC could be contingent upon the particular circumstances and goals of the researcher. If the objective is to choose the most concise model that yet offers a satisfactory fit, the Bayesian Information Criterion (BIC) may be favored. If the objective is to reduce the model's prediction error, AIC may be a more suitable choice.

Practically, it is advisable to take into account both AIC and BIC when choosing models, in addition to other variables including the model's interpretability, theoretical explanation, and practical considerations.

4.3 The Augmented Dickey-Fuller (ADF)Test

Statistical researchers often rely on the Augmented Dickey-Fuller (ADF) test to assess the stationarity of time series. This test is used to determine if a time series contains a unit root. When a time series has a unit root, it is considered non-stationary, which can cause issues with spurious regression(Occurs when there are parallel local patterns). The ADF test is an expanded version of the Dickey-Fuller (DF) test. It is assumed that a time series follows a first-order autoregressive (AR(1)) process in the DF test. On the other hand, the ADF test considers higher-order autoregressive processes and incorporates lagged difference terms in the model to address autocorrelation in the error terms [69]. The structure of the ADF test's hypothesis is as follows:

- HO (Null hypothesis): The null hypothesis states that the time series, which includes a unit root, is non-stationary.
- H1 (Alternative Hypothesis): The time series is stationary, indicating no evidence of a unit root.

The ADF test employs the following regression equation:

$$\nabla y_t = \alpha + \beta t + \gamma y_{t-1} + \sum_{i=1}^{l-1} \delta_i \nabla y_{t-i} + \varepsilon_t \tag{4.10}$$

In this case, y_t refers to the time series under examination, ∇ represents the difference operator, t represents the time trend, l specifies the appropriate lag length, and ε_t

represents the error term. The coefficient γ is examined for statistical significance in order to establish whether a unit root is present. If the coefficient γ is not statistically significant (i.e., the p-value is large), the null hypothesis cannot be rejected, and it is concluded that the time series is non-stationary [70]. Take,for instance, a time series comprising the daily closing prices of a certain stock. The ADF test can be utilized to assess the stationarity of this time series. If the Augmented Dickey-Fuller (ADF) test does not reject the null hypothesis, which indicates the existence of a unit root, it implies that the stock prices are not stable. In such cases, it may be required to apply adjustments such differencing before doing ARIMA modeling [71].

4.4 The Kwiatkowski-Phillips-Schmidt-Shin (KPSS) Test

The KPSS test is specifically developed to examine whether a time series is stationary or non-stationary, in contrast to the Augmented Dickey-Fuller (ADF) test which focuses on detecting the presence of a unit root [72].

The KPSS test is based on the following model:

$$y_t = \beta t + r_t + \varepsilon_t \tag{4.11}$$

In the given equation, we have y_t as the time series being tested, t representing the deterministic trend, r_t as a random walk process, and ε_t as the stationary error term. The random walk process is defined as:

$$r_t = r_{t-1} + u_t \tag{4.12}$$

where u_t is an error term that is independently and identically distributed (i.i.d.), with a mean of zero and a constant variance. The structure of the hypothesis for the KPSS test is as follows:

- H0 (Null Hypothesis): The time series is stationary around a deterministic trend.
- H1 (Alternative Hypothesis): The time series is non-stationary due to the presence of a unit root.

The KPSS test statistic can be calculated using the following formula:

$$KPSS = \frac{\sum_{t=1}^{T} S_t^2}{T^2 \hat{\sigma}^2} \tag{4.13}$$

Let S_t represent the partial sum of the residuals obtained from regressing y_t on an intercept and a time trend. T denotes the sample size, while $\hat{\sigma}^2$ is an estimator of the long-run variance of the residuals [73]. When the KPSS test statistic surpasses the critical value at a selected significance level, we reject the null hypothesis of stationarity and conclude that the time series is non-stationary. If the test statistic is smaller than the critical value, the null hypothesis cannot be rejected, and it is assumed that the time series is stationary. For instance, let's examine a time series that consists of monthly inflation rates. The KPSS test can be used to assess the stationarity of this time series. When the KPSS test rejects the null hypothesis of stationarity, it means that the inflation rates are not stationary. In such cases, it may be necessary to apply suitable transformations or differences to the data before fitting an ARIMA model.

4.5 ARIMA Model Fitting

Three steps make up thetechnique for fitting ARIMA (Autoregressive Integrated Moving Average) models: estimation, verification, and identification.

4.5.1 Identification

Data preprocessing is a crucial step in achieving stationarity, as it lays the groundwork for accurate time series analysis. The selection and adjustment of the autoregressive (p) and moving average (q) components are key, and they can be modified as the model evolves. Time plots serve as a powerful tool for visually assessing data stationarity. They offer a quick and intuitive way to understand the data and determine if it originates from a stationary process.

- Take into account changing the data (for as by taking logs).
- Determine whether differencing is required to achieve series stationarity.

The autocorrelation function (ACF) should quickly decline to zero in order to achieve stationarity. If not, consider making one or two differences in the series (going beyond

the second change is uncommon). The initial identification of p and q is the next stage. We utilize PACF and the Autocorrelation Function for this, keeping in mind that:

- The ACF is 0 for a moving average order q (MA(q)) process beyond lag q.
- The PACF for an autoregressive process with order p (AR(p)) is zero after lag p.

4.5.2 Estimation: AR processes

For an AR(p) process defined as:

$$Y_t = \sum_{i=1}^{p} \alpha_i Y_{t-i} + \varepsilon_t$$

We have the Yule-Walker equations:

$$\rho_k = \sum_{i=1}^{p} \alpha_i \rho_{|i-k|}, \text{ for } k > 0$$

We fit the parameters $\alpha_1, \ldots, \alpha_p$ by solving:

$$r_k = \sum_{i=1}^{p} \alpha_i r_{|i-k|}, \text{ for } k = 1, \ldots, p$$

These p equations can be solved using the Levinson-Durbin recursion for the p unknowns $\alpha_1, \ldots, \alpha_p$.

The Levinson-Durbin recursion also provides the residual variance:

$$\hat{\sigma}_p^2 = \frac{1}{n} \sum_{t=p+1}^{n} \left(X_t - \sum_{j=1}^{p} \hat{\alpha}_j Y_{t-j} \right)^2$$

One way to guide the selection of the appropriate order p is by minimizing an information criterion such as the Akaike Information Criterion (AIC).

If $(Y_t)_t$ is a causal AR(p) process with independent and identically distributed white noise errors $(0, \sigma_\varepsilon^2)$, then the Yule-Walker estimator $\hat{\alpha}$ is considered to be the best choice when it comes to the normal distribution [74].

4.5.3 Estimation: ARMA processes

For an ARMA(p, q) process, assuming Gaussian white noise errors, we can use maximum likelihood estimation. We rely on the prediction error decomposition, where the joint density of $Y_1, \ldots, Y_n$ is expressed as:

$$f(Y_1, \ldots, Y_n) = f(Y_1) \prod_{t=2}^{n} f(Y_t | Y_1, \ldots, Y_{t-1})$$

Supposing the conditional distribution of Y_t given $Y_1, \ldots, Y_{t-1}$ is normal with mean $\hat{Y}_t$ and variance P_{t-1}, and $Y_1 \sim N(\hat{Y}_1, P_0)$, the log-likelihood is:

$$-2\log L = \sum_{t=1}^{n} \left\{ \log(2\pi) + \log P_{t-1} + \frac{(Y_t - \hat{Y}_t)^2}{P_{t-1}} \right\}$$

In this case, $\hat{Y}_t$ and P_{t-1} depend on the parameters $\alpha_1, \ldots, \alpha_p, \beta_1, \ldots, \beta_q$. To find the maximum likelihood estimators, one can numerically minimize $-2\log L$ with respect to these parameters. By utilizing the observed information matrix at the maximum likelihood estimates, we can obtain an estimate of the covariance matrix for the estimators. This allows us to calculate approximate standard errors for the parameters.

4.5.4 Confirmation

Confirming that the model accurately aligns with the data is the third step. There are two main approaches:

- Overfitting: Include additional parameters and determine their significance using likelihood ratios or t-tests.
- Residual analysis: For assessing consistency with white noise, you can calculate residuals from the fitted model and then plot their ACF, PACF, spectral density estimates, and so on.

5. DATA SETS

For this section, we will delve into the stock market and cryptocurrency market data that will be utilized in this thesis. We will also examine the structure of datasets, and will describe the features as rows and columns of the datasets are going to investigate.

5.1 Yahoo!Finance API

Yahoo Finance is a media organization that provides a wealth of information on financial news, stock prices, press releases, and financial reports. All data on this platform is freely available for access and download. One can obtain the data directly from the Yahoo!Finance portal, or utilize their Yahoo!Finance API to retrieve financial information published by Yahoo!Finance.

Yahoo previously had their own official API, however it was discontinued in 2017. Currently, unapproved application programming interfaces (APIs) and software libraries are being utilized to retrieve this data.

In this thesis, we will utilize the `yfinance` python package to work with the datasets. The library, created by Ran Aroussi, is a widely used open source tool designed to retrieve financial data from Yahoo Finance. The library offers a dependable approach for retrieving historical market data from Yahoo!Finance, with a level of detail as precise as 1 minute intervals. The library encompasses comprehensive market data on Cryptocurrencies, conventional currencies, equities, and bonds, as well as basic and options data, along with market analysis and news.

To obtain the datasets required for this thesis, it will suffice to construct the following python code. The `Ticker()` function enables us to retrieve a particular dataset based on its ticker symbol.

```
import yfinance as yf
BIST100 = yf.Ticker("XU100.IS")
```

The sample code we gave above gets the BIST 100 index data.

5.2 Data Sets

In this section, we will examine the structure of the four different data sets that we will study in this thesis. These data sets are listed below:

1. BIST 30 (XU030.IS)
2. BIST 100 (XU100.IS)
3. Bitcoin (BTC-USD)
4. Ethereum (ETH-USD)

All of the data sets are columnar data frames and share the same columns. These columns are listed below:

- **Open:** The price at which a financial instrument starts when the stock market opens on the given day.
- **High:** The maximum price at which a stock was traded during the day.
- **Low:** The minimum price at which a stock is traded during a specific day.
- **Close:** The price at which a financial instrument end when the stock market closed on the given day.
- **Dividend:** Since the Ticker function gives the profit share values as zero, we will not include this part in our study.
- **Stock Split:** Since our data is data for more than one company and cryptocurrencies do not have stock, ticker function showed these values to be zero. We will not include this section in our study.
- **Volume:** The number of stocks traded between open and close on a given day.

5.2.1 BIST data

In this section we are going to analyse the structure of the BIST 30 and BIST 100 datasets. The datasets we use in this thesis cover the time interval from 2000 to 2010.

The Istanbul Stock Exchange divides stocks into four distinct groups. These are listed below.

1. Class A Shares: These are the shares in active circulation with a value of 30 million TL or more.
2. Class B Shares: These are the stocks with an active circulating share value between TL 10 million and TL 30 million.
3. Class C Shares: Shares with an active circulating share value of less than TL 10 million.
4. Group D Stock: It is the stock traded in the Emerging Business Market, Free Trading Platform, Qualified Investor Transactions Market or the Custody Market.

The BIST 30 is a benchmark that evaluates the returns of the stocks belonging to the 30 firms with the greatest trading volume and market capitalization, which are listed on the Istanbul Stock Exchange. The stocks comprising the BIST 30 index are likewise encompassed within the BIST 50 and BIST 100 indices. The shares that are part of the BIST 30 index are meticulously selected from both group A and group B stocks. Group C and Group D are omitted from the index because they possess a greater degree of risk. Every corporation is restricted to a maximum allocation of 10 percent of the stocks that make up the BIST 30 index. The goal is to reduce the effect of price changes in stocks of companies with substantial market capitalization on the overall index.

The BIST 30 index shares undergo a re-evaluation process every 3 months, specifically in January, April, July, and September. Investors who prefer a long-term approach can benefit from the index, as it offers the potential for profits with relatively low risk. Additionally, experienced investors can capitalize on short-term buy-sell transactions to generate profits.

The BIST 100 index holds a significant position as the most popular and widely used index of the Istanbul Stock Exchange. It is closely monitored by all major investors. The companies included in this index are determined by reviewing the stocks traded in Istanbul Stock Exchange four times a year. Through the review process, companies eligible for inclusion in the BIST 100 index are chosen from those traded on the National Market, real estate investment partners traded on the Corporate Products Market, and venture capital investment partners.

Shares that are actively traded are organized based on their market value and daily average trading volume. The index includes the top 100 shares in both rankings. Stocks eligible for inclusion in the BIST 100 index are exclusively chosen from Group A and Group B shares. Group C and Group D groups are not part of the BIST 100 index, similar to how they were not included in the BIST 30 index. Similar to a quantitative analyst, the index shares undergo re-evaluation every 3 months, specifically in January-March, April-June, July-September, and October-December. Periodic changes are announced with a minimum of 10 days notice prior to the start of the relevant index period.

Data: The ticker code of BIST 30 and BIST 100 indices respectively are "XU030.IS" and "XU100.IS". We will use this code in all our python code and data analysis.

Date	Open	High	Low	Close	Volume	Dividends	Stock Splits
2000-01-04	15208.799805	17639.300781	15208.799805	17512.199219	54538700	0	0
2000-01-05	17512.199219	17802.099609	16237.700195	16932.000000	66720900	0	0
2000-01-06	16932.000000	17460.699219	16086.799805	16200.000000	66095000	0	0
2000-01-07	16200.000000	16305.599609	15623.500000	15837.400391	25444400	0	0
2000-01-11	15837.400391	16388.300781	15293.200195	16347.400391	53618400	0	0

Figure 5.1 : BIST100 Sample

In Figure 5.1, one can see a sample of the BIST 100 index dataset from January 4th, 2000 to January 11th, 2000. Also, in Figure 5.2 shows the four-year opening index of BIS100.

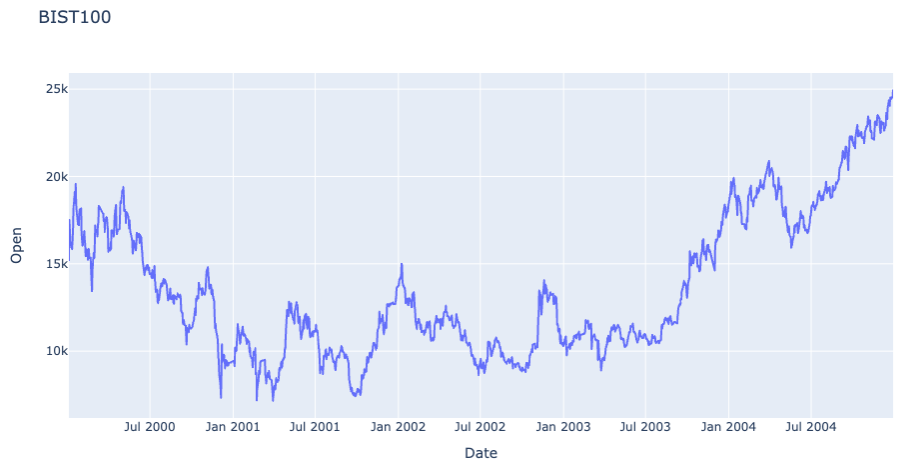

Figure 5.2 : BIST100 Date and Open

5.2.2 Crypto currency data

In this thesis, we are going to use the price time series data of ethereum, bitcoin from their beginning to the present. We retreived the data from Yahoo API. We will use the American dollar (USD) as the base currency. The value corresponding to each unit of cryptocurrency will be expressed in USD. In Figure 5.3 shows a small sample of BitCoin prices (in USD) from September 30th, 2021 to October 8th, 2021.

Date	Open	High	Low	Close	Volume	Dividends	Stock Splits
2021-09-30	41551.269531	44092.601562	41444.582031	43790.894531	31141681925	0	0
2021-10-01	43816.742188	48436.011719	43320.023438	48116.941406	42850641582	0	0
2021-10-02	48137.468750	48282.062500	47465.496094	47711.488281	30614346492	0	0
2021-10-03	47680.027344	49130.691406	47157.289062	48199.953125	26638115879	0	0
2021-10-04	48208.906250	49456.777344	47045.003906	49112.902344	33383173002	0	0
2021-10-05	49174.960938	51839.984375	49072.839844	51514.812500	35873904236	0	0
2021-10-06	51486.664062	55568.464844	50488.191406	55361.449219	49034730168	0	0
2021-10-07	55338.625000	55338.625000	53525.468750	53805.984375	36807860413	0	0
2021-10-08	53802.144531	55922.980469	53688.054688	53967.847656	34800873924	0	0

Figure 5.3 : Bitcoin-USD Sample

5.2.3 Apple stock market data

Yahoo Finance also provides historical pricing of AAPL shares on the NASDAQ exchange. Apple Inc. Yfinance provides the option to access data on a daily, weekly, or monthly basis until the date when shares are issued. We will be progressing our work using this source within the specified time frames.

6. EXPERIMENTS

The data sets to be used in all experiments throughout the research were downloaded from `yfinance` in accordance with the required date ranges.

Additionally, the experiments were conducted using *Jupyter Notebook* as the computational tool, specifically employing Python version 3.8.3 within the Anaconda distribution.

6.1 Computational Tools

What we were primarily looking for when searching the literature was that the library contains the ARIMA/SARIMA and GARCH models. In this direction, we conducted a systematic literature review. We decided to use `PyFlux`, `Sktime` and `Darts` among the software packages that contain the models we need.

6.1.1 PyFlux

PyFlux is a Python package that is open-source and specifically designed for tackling statistical challenges. The library offers a comprehensive selection of contemporary time series models, together with a versatile range of inference approaches (both frequentist and Bayesian) that can be utilized with these models.

Users have the ability to construct a comprehensive probability model in which four data and hidden variables (known as parameters) are considered as random variables using a shared probability distribution denoted as $p(y,z)$. The benefit of employing a probabilistic approach lies in its ability to provide a comprehensive representation of uncertainty, which holds significant importance in time series activities like forecasting. Alternatively, users can utilize the Maximum Likelihood estimate to enhance the speed of their operations within the unified API.

Models provided by PyFlux are as follows:

(i) ARIMA models

(ii) Dynamic Paired Comparison models

(iii) GARCH models

(iv) GAS models

(v) GAS State Space models

(vi) Gaussian State Space models

(vii) Non-Gaussian State Space models

(viii) VAR models

6.1.2 sktime

sktime is a powerful Python library for analyzing time series data. It brings together various functionalities found in multiple Python libraries. In addition, it brings its own distinct characteristics for making predictions. We can use it to train, fine-tune, and evaluate models for time series. It aligns well with scikit-learn. sktime offers a cohesive interface for various time series learning tasks, such as time series classification, regression, clustering, annotation, and forecasting.

Sktime aims to provide a unified API for various time series tasks, expanding on the familiar scikit-learn interface to accommodate temporal data. It strives to maintain a similar syntax and logic whenever feasible.

6.2 The Experimental Setups

6.2.1 The BIST setup

The BIST 30 data we use in our study is based on daily data between 2000 and 2010. Our data set consists of 2744 rows and 7 columns. Details about our data set can be found in Section 5.

After the data set check, there were no null values, so no null treatment was applied to our data set.

Table 6.1 : Descriptive Statistics of the BIST30

	Open	High	Low	Close	Volume	Div.	Splits
count	2744	2744	2744	2744	2744	2744	2744
mean	36721	37212	36175	36732	3.3×10^4	0	0
std	21018	21194	20814	21017	1.0×10^5	0	0
min	9073	9633	8701	9073	0	0	0
25%	16109	16374	15722	16109	0	0	0
50%	32925	33372	32508	32956	0	0	0
75%	53712	54363	52882	53715	0	0	0
max	91188	91485	90007	91249	8.7×10^5	0	0

In Figure 6.1 shows the line chart of the BIST30 index in the last specified date range of the stock market closing. Table 6.1 shows the statistical distributions of the BIST30 dataset. There are no null values in this data set.

Figure 6.1 : BIST30 Closed Price between 2000 and 2010

The BIST 30 data we use in our study is based on daily data between 2000 and 2010. Our data set consists of 2744 rows and 7 columns. Table 6.2 shows the statistical distributions of the BIST100 dataset. There are no null values in this data set.

6.2.2 The bitcoin setup

Table 6.3 presents various statistical summaries of the Bitcoin-USD data set. The table presents various statistical summaries of the Bitcoin-USD data set. The table provides a comprehensive set of statistical measures for various financial indicators, including opening, high, low, and closing prices, trading volume, dividends, and stock splits. The data includes various statistical measures for different aspects of the stock, such as

Table 6.2 : Summary Statistics of the Dataset

	Open	High	Low	Close	Volume	Div.	Splits
count	2744	2744	2744	2744	2744	2744	2744
mean	29100.88	29460.63	28701.49	29110.20	2.945×10^8	0	0
std	16705.10	16827.40	16562.10	16706.55	2.233×10^8	0	0
min	7159.70	7477.40	6796.90	7159.70	0	0	0
25%	12818.67	13112.75	12520.00	12818.67	1.250×10^8	0	0
50%	25719.90	26010.70	25442.65	25720.10	2.459×10^8	0	0
75%	42889.03	43434.10	42312.02	42863.77	3.986×10^8	0	0
max	71503	71776.90	70815.20	71543.30	1.312×10^9	0	0

mean, standard deviation, minimum, maximum, and quartiles. These measures cover opening, high, low, and closing prices, trading volume, dividends, and stock splits.

Figure 6.2 : Bitcoin Closed Price between 20018 and 2023

In Figure 6.2, the five-year closing price distribution of our data set is shown.

6.2.3 The ethereum setup

Table 6.4 displays a range of statistical summaries for the Ethereum-USD dataset, which are comparable to the Bitcoin-USD dataset seen in Table 3. The table presents an extensive range of statistical metrics for several financial indicators, such as opening, high, low, and closing prices, trading volume, dividends, and stock splits. The dataset comprises diverse statistical indicators for several stock attributes, including mean, standard deviation, minimum, maximum, and quartiles.

Table 6.3 : Summary Statistics of the Bitcoin-USD

	Open	High	Low	Close	Volume	Div.	Splits
count	1826	1826	1826	1826	1826	1826	1826
mean	20337.65	20838.15	19770.46	20337.36	2.660×10^{10}	0	0
std	16991.66	17435.46	16476.29	16986.40	1.982×10^{10}	0	0
min	3236.27	3275.38	3191.30	3236.76	2.923×10^{9}	0	0
25%	7686.93	7908.67	7517.31	7682.07	1.273×10^{10}	0	0
50%	10966.99	11301.31	10652.98	10960.59	2.432×10^{10}	0	0
75%	33789.31	34790.41	32267.99	33740.26	3.580×10^{10}	0	0
max	67549.73	68789.63	66382.06	67566.83	3.509×10^{11}	0	0

These metrics encompass the opening, high, low, and closing prices, trading volume, dividends, and stock splits. The summary statistics offer useful insights into the behavior and characteristics of the Ethereum-USD dataset. Through analyzing the mean, standard deviation, minimum, maximum, and quartiles, we may enhance our comprehension of the central tendency, dispersion, and range of the data. These statistics can assist investors and analysts in making well-informed decisions and evaluating the potential risks and opportunities linked to Ethereum-USD.

Table 6.4 : Summary Statistics of the Ethereum-USD

	Open	High	Low	Close	Volume	Div.	Splits
count	1826	1826	1826	1826	1826	1826	1826
mean	1146.97	1183.51	1105.15	1147.01	1.322×10^{10}	0	0
std	1200.83	1237.66	1158.30	1200.30	1.078×10^{10}	0	0
min	84.28	85.34	82.83	84.31	9.485×10^{8}	0	0
25%	209.03	213.73	203.59	208.92	5.157×10^{9}	0	0
50%	519.07	532.95	499.45	518.85	1.105×10^{10}	0	0
75%	1802.36	1840.00	1732.77	1803.34	1.836×10^{10}	0	0
max	4810.07	4891.70	4718.04	4812.09	8.448×10^{10}	0	0

6.2.4 The apple setup

Apple shares adhere to the same data structure and terminology, as you see in table 6.5. We retrieved Apple shares data from YahooFinance and conducted the pre-processing procedure.

Table 6.5 : Summary Statistics of the Apple

	Open	High	Low	Close	Volume	Div.	Splits
count	2515	2515	2515	2515	2515	2515	2515
mean	25.228	25.452	25.000	25.234	2.998×10^8	0.002	0.003
std	13.925	14.047	13.822	13.944	2.353×10^8	0.016	0.140
min	5.808	5.918	5.744	5.798	4.545×10^7	0.000	0.000
25%	13.905	14.017	13.755	13.871	1.248×10^8	0.000	0.000
50%	22.350	22.550	22.181	22.378	2.222×10^8	0.000	0.000
75%	35.426	35.723	35.039	35.355	4.068×10^8	0.000	0.000
max	70.718	71.410	69.989	70.424	1.881×10^9	0.193	7.000

6.3 ARIMA Models

The process began with thorough data preprocessing, which involved checking for missing values (null) and examining the statistical distribution of the data. We applied logarithmic transformations to the data to facilitate the modelling process, aiming to stabilize the variance and improve the model's performance.

Our study leveraged the power of visual exploratory analysis using the matplotlib library in Python. This approach, which involved creating various plots and charts, not only provided a clear picture of the data's behaviour but also unearthed valuable insights into the patterns, trends, and seasonality present in the data. These insights not only deepened our understanding of the data but also opened up intriguing possibilities for further analysis.

Before delving into the model selection, we conducted two crucial tests-the Augmented Dickey-Fuller (ADF) test and the Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test-to assess the stationarity of the time series. These tests played a pivotal role in determining the appropriate differencing order for the ARIMA model. We also examined the autocorrelation and partial autocorrelation plots, which provided key insights into the potential autoregressive (AR) and moving average (MA) terms for the model. These plots, by revealing the correlation between the time series and its lagged values, helped us identify the significant lags and the order of the AR and MA terms to be included in the ARIMA model.

Finally, we proceeded with ARIMA modelling, where we iteratively tested different combinations of parameters (p, d, q) to find the optimal model. The selection criterion was based on the Akaike Information Criterion (AIC), which balances the goodness of fit with the complexity of the model. We searched stepwise to minimize the AIC value and identify the best-performing ARIMA model. The chosen model was then used to make forecasts and assess its performance on the given time series data.

6.3.1 Constructing the ARIMA models

In this experiment, we will use five different data sets to try to determine the appropriate model for the ARIMA model, whose technical details we have given.

Table 6.6 : ADF and KPSS Test Results (without Log Transformation)

	BIST 30	BIST 100	Bitcoin	Ethereum	Apple Stock
ADF Statistic	-0.521	-0.481	-0.993	-1.015	-0.788
ADF p-value	0.888	0.896	0.756	0.748	0.823
KPSS Test Statistic	0.659	0.654	0.591	0.887	0.289
KPSS p-value	0.010	0.010	0.010	0.010	0.010
KPSS Lags Used	31.000	31.000	27.000	27.000	30.000
KPSS Critical Value (10%)	0.119	0.119	0.119	0.119	0.119
KPSS Critical Value (5%)	0.146	0.146	0.146	0.146	0.146
KPSS Critical Value (2.5%)	0.176	0.176	0.176	0.176	0.176
KPSS Critical Value (1%)	0.216	0.216	0.216	0.216	0.216

In the provided results, in Table 6.6, the ADF test statistics for all the time series are negative but quite close to zero, and the corresponding p-values are relatively high (ranging from 0.747834 to 0.895752). Based on the results, it appears that we are unable to reject the null hypothesis of a unit root at commonly used significance levels (such as 1%, 5%, or 10%) for any of the series. Based on the ADF test results, it seems that all the time series are non-stationary.

On the other hand, the provided results show that the KPSS test statistics for all the time series are high (ranging from 0.289036 to 0.886764), and the corresponding p-values are low (all equal to 0.010000). Moreover, the test statistics exceed the critical values at all conventional significance levels (1%, 2.5%, 5%, and 10%). These findings indicate that we can reject the null hypothesis of trend-stationarity for all the series, suggesting that they are non-stationary.

The ADF and KPSS test findings consistently indicate that the time series of BIST 30, BIST 100, Bitcoin, Ethereum, and Apple stock are non-stationary in their original form. Consequently, when we perform the tests using the logarithmic transformation of the feature utilized in the data sets, the following findings are found.

Table 6.7 : ADF and KPSS Test Results for Log Transformed Data

	BIST 30	BIST 100	Bitcoin	Ethereum	Apple Stock
ADF Statistic	-15.542	-15.221	-29.666	-12.962	-12.896
ADF p-value	0.000	0.000	0.000	0.000	0.000
KPSS Test Statistic	0.085	0.084	0.218	0.207	0.063
KPSS p-value	0.100	0.100	0.010	0.013	0.100
KPSS Lags Used	7	5	8	8	6
KPSS Critical Value (10%)	0.119	0.119	0.119	0.119	0.119
KPSS Critical Value (5%)	0.146	0.146	0.146	0.146	0.146
KPSS Critical Value (2.5%)	0.176	0.176	0.176	0.176	0.176
KPSS Critical Value (1%)	0.216	0.216	0.216	0.216	0.216
Differencing	1	1	1	1	1

The results of ADF and KPSS tests applied to transformed data show a significant improvement in the stationarity properties of all data sets in Table 6.7. Once differencing, ADF test statistics had high negative values for all series, and p-values were calculated at 0.000000. KPSS test results also generally support the stationarity of transformed data. These findings reveal that differencing eliminated the original series' trend and unit root problems, and the transformed data became suitable for model selection.

In this phase of the study, we made sure that the close prices were stationary, as this is a common feature in all of our 5 datasets. We utilised the auto arima function from the pmdarima library to identify the most suitable ARIMA model parameters. This function conducts a grid search across different combinations of p, d, and q values, taking into account a maximum of 10 lags for both the autoregressive and moving average components. It also automatically determines the suitable level of differencing.

The auto arima function also includes seasonal components with a period of 12, making it ideal for analysing monthly data. Once the optimal model was identified, we proceeded to extract the model order, the differenced series based on this order, and the model residuals for further analysis. By utilising this approach, we can effectively identify the optimal ARIMA model structure for every dataset.

In Table 6.8 , the selected models from all model grid searches are as follows.

Table 6.8 : Results of Grid Search and AIC Scores for Different Data Sets

Data Set	Result of Grid Search	AIC
BIST30	ARIMA(0,1,0)(0,0,0)[12]	-12238.825
BIST100	ARIMA(0,1,0)(0,0,0)[12]	-12477.416
Bitcoin	ARIMA(2,1,0)(0,0,0)[12]	-6661.455
Ethereum	ARIMA(1,1,2)(0,0,0)[12]	-5698.832
Apple	ARIMA(0,1,0)(0,0,0)[12]	-13570.951

End of this studying, ARIMA modelling experiments have uncovered fascinating patterns across various financial instruments. The low AIC values indicate that the models may not perfectly fit the data, but this doesn't always make them completely unsuitable. Interestingly, incorporating seasonal components (SARIMA) did not have a substantial impact on the model's performance. The BIST30, BIST100, and Apple stock series can be accurately characterised by ARIMA(0,1,0) models, which suggest that they follow simple random walk processes after first-order differencing. On the other hand, Bitcoin and Ethereum showcase more intricate dynamics. Bitcoin adheres to an ARIMA(2,1,0) model, while Ethereum follows an ARIMA(1,1,2) model. The difference in dynamics between cryptocurrency markets and traditional financial markets is evident from this disparity. Random walk models are commonly observed in line with the Efficient Market Hypothesis, suggesting limited predictability of price movements for the majority of assets in our study.

To support these results, the results of the Ljung-Box tests of the results of grid search are as follows:

Table 6.9 : Results of Ljung-Box for Results of Grid Search

Data	LB Statistics	LB p-value
BIST30	1.250	1.000
BIST100	1.102	1.000
Bitcoin	2.715	0.656
Ethereum	6.268	1.000
Apple	6.418	1.000

When we examine Table 6.9, the p-values of BIST30 and BIST100 data are equal to 1.0, indicating no autocorrelation. On the other hand, the p-values for Bitcoin, Ethereum, and Apple data are relatively lower but still above the 0.05 significance level. Therefore, there is no strong evidence for autocorrelation in these data sets.

In addition, the results of the Ljung-Box test suggest that there is no significant autocorrelation in the residuals of our model, which confirms that our models' underlying assumptions are valid. Nevertheless, the AIC values for all models indicate a lack of strong predictive ability, emphasising the intricate nature of financial time series and the potential requirement for more advanced modelling techniques in future studies.

The information provided sets exhibit either a lack of autocorrelation or a very weak autocorrelation. It is evident that these data sets exhibit a random walk pattern, where past values have no impact on future values.

Considering this analysis, future studies may want to include macroeconomic factors or market sentiment as external variables . In addition, conducting analyses with shorter time intervals, such as hourly or 30 or 15-minute intervals, may result in the creation of more appropriate models. These approaches have the potential to enhance our understanding of market dynamics and increase the precision of predictions.

6.4 (G)ARCH Models

In this phase of the study, we focused on conducting a comprehensive analysis of the time series data to assess its stationarity. We started by calculating returns and summarising the results. Furthermore, we performed various hypothesis tests, including the Augmented Dickey-Fuller (ADF) test, to evaluate the stationarity of the series. After conducting a thorough analysis of the results, it has been determined that returns, especially log returns, are highly advantageous for practical purposes, especially in the field of mathematical modelling. This preference stems from their ability to handle changing characteristics and enhance the stability of time series data. Furthermore, we performed an assessment of ARCH effects using the Box-Ljung test, which yielded support for utilising GARCH-type models for our analysis.

6.4.1 Return Calculations and Volatility Measures

In the experiment phase of this research endeavour, we depart from the approach adopted in the initial experiment, which used closing prices of financial instruments as input variables. Instead, we introduce three innovative features: returns, logarithmic returns, and volatility (squared returns). These novel measures allow for a more comprehensive analysis of the intricate dynamics underlying price fluctuations [75].

Returns: It is denoted by $R(t)$, quantify the relative change in an asset's price over a specified time interval. Mathematically, this is expressed as:

$$R(t) = \frac{P(t) - P(t-1)}{P(t-1)} \tag{6.1}$$

where $P(t)$ represents the asset's price at time t, and $P(t-1)$ denotes the price during the preceding period [76].

Logarithmic Returns: Logarithmic returns, represented by $r(t)$, involve taking the natural logarithm of the ratio of prices at consecutive time points [77]:

$$r(t) = \ln\left(\frac{P(t)}{P(t-1)}\right) \tag{6.2}$$

Volatility: Volatility, denoted as $\sigma^2(t)$, is a measure of the dispersion of returns. It is calculated as the squared value of the logarithmic returns:

$$\sigma^2(t) = r(t)^2 = \left[\ln\left(\frac{P(t)}{P(t-1)}\right)\right]^2 \tag{6.3}$$

It also gives more importance to larger price movements, allowing for a more accurate representation of the extent of fluctuations [78]. Volatility offers valuable insights into the risk level of a financial instrument. Higher volatility suggests increased uncertainty and the possibility of both gains and losses.

By incorporating returns, logarithmic returns, and volatility as input features, rather than solely relying on price data, our models are able to capture complex patterns and inherent characteristics in the evolution of asset prices. This significantly improves the overall predictive capabilities of the modelling process, making it more effective and reliable.

After adding new features and completing the data preprocessing steps, our current data structure is shown in figure 6.3. The time series graph of each data set for return are in figure 6.4, figure6.5, figure 6.6, figure6.7 and figure6.8

Date	Open	High	Low	Close	Volume	returns	log_returns	Sq_Returns
2010-01-04	6.444463	6.476771	6.412758	6.461976	493729600	1.556469	0.015445	2.422596
2010-01-05	6.479790	6.509683	6.439027	6.473147	601904800	0.172885	0.001727	0.029889
2010-01-06	6.473149	6.498815	6.363542	6.370185	552160000	-1.590602	-0.016034	2.530015
2010-01-07	6.393737	6.401286	6.312211	6.358409	477131200	-0.184868	-0.001850	0.034176
2010-01-08	6.349954	6.401286	6.312514	6.400681	447610800	0.664829	0.006626	0.441998

Figure 6.3 : Apple Data after new features are added

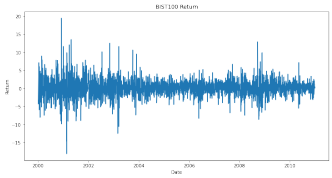

Figure 6.4 : BIST100 Return Graph

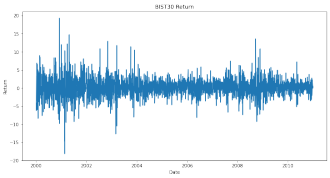

Figure 6.5 : BIST30 Return Graph

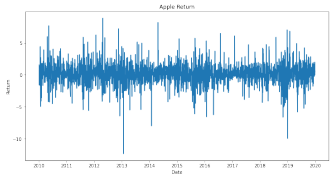

Figure 6.6 : Apple Return Graph

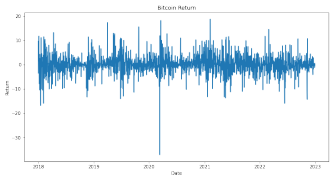

Figure 6.7 : Bitcoin Return Graph

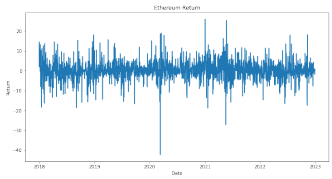

Figure 6.8 : Ethereum Return Graph

6.4.2 Summary Statistics of Returns

In the following paragraphs, we provide a thorough analysis of the log returns for our datasets. Through an analysis of the statistical distributions of their log returns, we seek to uncover valuable insights into the behaviour and characteristics of these varied assets. We include important statistical measures in our analysis, such as sample size, minimum and maximum values, mean, standard deviation, skewness, volatility and kurtosis. These metrics offer valuable insights into the characteristics of each asset's return distribution, helping to understand its central tendencies, asymmetry, and tail behaviour.

Table 6.10 : Descriptive Statistics of Different Data Sets for Log Returns

Statistics	**BIST100**	**BIST30**	**APPLE**	**Bitcoin**	**Ethereum**
Sample size	2743	2743	2515	1825	1825
Min	-0.181	-0.182	-0.124	-0.372	-0.423
Max	0.195	0.193	0.089	0.187	0.259
Mean	0.001	0.001	0.001	0.001	0.002
SD	0.025	0.026	0.016	0.038	0.050
Skewness	0.288	0.343	-0.178	-0.390	-0.308
Kurtosis	6.191	5.502	4.401	7.271	5.252

In Table 6.10, The skewness values for BIST100, BIST30, APPLE, Bitcoin, and Ethereum are as follows: 0.2881, 0.3433, -0.1780, -0.3897, and -0.3082, respectively. All of these values are non-zero, suggesting that all return distributions exhibit asymmetry.

The BIST100 and BIST30 demonstrate a positive skewness, indicating a distribution with a longer right tail. On the other hand, APPLE, Bitcoin, and Ethereum display a negative skewness, indicating a distribution with a longer left tail.

The kurtosis values for these financial instruments, 6.1914, 5.5020, 4.4012, 7.2710, and 5.2523 respectively, exceed 3, which is the kurtosis of a normal distribution. This finding is quite significant, as it suggests that all return distributions exhibit fat tails. This characteristic indicates that extreme events or outliers have a higher probability of occurring compared to what would be expected in a normal distribution. It is important to recognise the significance of this fact.

Bitcoin exhibits the highest kurtosis (7.2710), suggesting that it possesses the most pronounced tail thickness compared to other assets. This may suggest increased volatility and more frequent occurrences of extreme returns. Apple has the lowest kurtosis value of 4.4012, indicating that it still has fat tails, although not as pronounced as the other assets.

The values of standard deviation (SD), which offer valuable insights into the volatility of these assets, are quite revealing. Among these assets, Ethereum has the highest standard deviation (0.050012), making it the most volatile. On the other hand, APPLE has the lowest standard deviation (0.016235), suggesting relatively lower volatility. It's important not to overlook this significant difference in volatility.

Based on the statistics, it is evident that none of these financial instruments adhere to a perfectly normal distribution in their returns. Instead, each of them exhibits different levels of asymmetry and fat tails.

6.4.3 GARCH Modeling and Volatility Analysis of Returns

During this phase of the study, modelling procedures were conducted using the data from the 'log returns' column. The data was split into two sets: the train set and the test set. The data was rescaled and adjusted to enhance the performance of the model.

The GARCH model was applied to the training data and the calculated conditional volatility attribute was acquired for the training phase of the time series.

Next, the scaler was applied to the training data's conditional volatility arrays, resulting in a transformation. The converted data was then plotted and compared to the scaled realized volatility. Upon comparison, it was found that none of the coefficients exhibited statistical significance. This expansion of the model specifically accounts for the observation that negative shocks have a greater influence on volatility compared to positive shocks.

In addition, the study utilized a wide range of forecast performance indicators to thoroughly examine the accuracy of time series forecasts and evaluate the predictive ability of the GARCH models. The field commonly employs many metrics such as Mean Absolute Percentage Error (MAPE), Mean Squared Error (MSE), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE). The meticulous use of these techniques in this research provides a thorough assessment of the predictive capacities of GARCH models, therefore bolstering the dependability of the study's findings.

6.4.4 Constructing the (G)ARCH Models

To initiate the experiment, we generated data for the new features using the Python codes provided below for the features with mathematical definitions outlined in section 6.4.1.

```
df['returns'] = 100*df.Close.pct_change().dropna()
df['log_return'] = np.log(df.Close/df.Close.shift(1))
```

The code above calculates the simple returns and logarithmic returns. We proceeded with an analysis using this expanded dataset. The analysis of stationarity, as mentioned earlier, was conducted on the log returns to ensure that the necessary conditions for stationarity were satisfied prior to proceeding with the modelling process.

As seen in Table6.11, we obtained stationary evidence for the Log Return column with ADF test results. Notice that we provided the stationary status with the columns we created directly without any transformation.

Table 6.11 : Dickey-Fuller test results for the log returns columns of the 5 datasets

	BIST30	BIST100	Bitcoin	Ethereum	Apple
Test Statistic	-15.55	-15.23	-29.67	-12.97	-12.89
p-value	0.00	0.00	0.00	0.00	0.00
Critical Value (1%)	-3.43	-3.43	-3.43	-3.43	-3.43
Critical Value (5%)	-2.86	-2.86	-2.86	-2.86	-2.86
Critical Value (10%)	-2.57	-2.57	-2.57	-2.57	-2.57

In Table 6.11, based on the Dickey-Fuller test results, we can conclude that the returns series for all five datasets (BIST30, BIST100, Bitcoin, Ethereum, and Apple) are stationary.

This is evident from the test statistics, which are less than the critical values at the 1%, 5%, and 10% significance levels. Additionally, the p-values for all datasets are incredibly close to zero, indicating concrete evidence against the null hypothesis of a unit root (non-stationarity).

In the table 6.12 presents the Box-Ljung test, a tool used to determine if the rate of returns exhibits the ARCH effect. The null hypothesis states that the rate of returns does not have the ARCH effect, while the alternative hypothesis suggests otherwise. Refer to the publication by Forsberg and Bollerslev (2002).

Table 6.12 : Box-Ljung Test Results for Data Sets

Data Sets	Test Value	p-value
BIST100	18.590	0.029
BIST30	19.599	0.021
APPLE	16.125	0.064
Bitcoin	14.617	0.102
Ethereum	22.843	0.007

The Box-Ljung test results for the analyzed financial instruments (BIST100, BIST30, APPLE, Bitcoin, and Ethereum) indicate the presence of ARCH effects, albeit with varying degrees of strength:

1. BIST100, BIST30, and Ethereum demonstrate strong ARCH effects at the 5% significance level (p-values less than 0.05).

2. APPLE shows borderline ARCH effects at the 5% significance level (p-value = 0.0643), which are statistically significant at the 10% level.

3. Bitcoin, while having the highest p-value (0.1020), still shows some evidence of ARCH effects, although not statistically significant at the conventional 5% or 10% levels.

The results suggest that the return series of these financial assets display different patterns of volatility clustering and time-dependent variance. This discovery supports the use of GARCH-type models for predicting volatility in these assets.

ARCH effects indicate that previous levels of volatility can have an impact on future levels of volatility in these markets. The significant ARCH effects observed in BIST100, BIST30, and Ethereum suggest that these markets might be more responsive to previous shocks and necessitate more advanced volatility modelling techniques. However, the less compelling evidence for Bitcoin and the ambiguous case of APPLE suggest that although ARCH effects exist, they might be less prominent or more intricate.

Bist30 Model Selection: The results of the top 7 models are given in table 6.13.

Table 6.13 : Top 7 GARCH Model Performances for BIST 30 Log Return

Model	Distribution	Log-Likelihood	AIC	BIC
GJR-GARCH(1,1)	Student's t	-5009.021	10030.043	10064.204
GJR-GARCH(2,1)	Student's t	-5008.801	10031.602	10071.457
GJR-GARCH(1,2)	Student's t	-5009.021	10032.043	10071.897
GJR-GARCH(2,2)	Student's t	-5008.313	10032.627	10078.175
GARCH(1,1)	Student's t	-5015.976	10041.952	10070.420
GARCH(1,2)	Student's t	-5015.951	10043.903	10078.064
GARCH(2,1)	Student's t	-5015.976	10043.952	10078.113

Bist100 Model Selection: The results of the top 7 models are given in table 6.14.

Table 6.14 : Top 7 GARCH Model Performances for BIST 100 Log Return

Model	Distribution	Log-Likelihood	AIC	BIC
GJR-GARCH(1,1)	Student's t	-4884.207	9780.414	9814.574
GJR-GARCH(2,1)	Student's t	-4883.950	9781.900	9821.754
GJR-GARCH(1,2)	Student's t	-4884.201	9782.401	9822.255
GJR-GARCH(2,2)	Student's t	-4883.318	9782.636	9828.184
GARCH(1,1)	Student's t	-4892.059	9794.119	9822.586
GARCH(1,2)	Student's t	-4892.018	9796.037	9830.198
GARCH(2,1)	Student's t	-4892.059	9796.119	9830.280

Apple Model Selection: The results of the top 7 models are given in table 6.15.

Table 6.15 : Top 7 GARCH Model Performances for Apple Log Return

Model	Distribution	Log-Likelihood	AIC	BIC
GJR-GARCH(1,2)	Student's t	-3596.084	7206.168	7245.416
GJR-GARCH(1,1)	Student's t	-3597.425	7206.850	7240.491
GJR-GARCH(2,2)	Student's t	-3596.083	7208.167	7253.022
GJR-GARCH(2,1)	Student's t	-3597.425	7208.850	7248.098
EGARCH(2,1)	Student's t	-3600.583	7213.166	7246.808
EGARCH(1,2)	Student's t	-3600.612	7213.225	7246.866
EGARCH(2,2)	Student's t	-3600.583	7215.166	7254.415

Bitcoin Model Selection: The results of the top 7 models are given in table 6.16.

Table 6.16 : Top 7 GARCH Model Performances for Bitcoin Log Return

Model	Distribution	Log-Likelihood	AIC	BIC
EGARCH(1,1)	Student's t	-3826.778	7663.556	7689.987
EGARCH(1,2)	Student's t	-3826.676	7665.352	7697.069
EGARCH(2,1)	Student's t	-3826.731	7665.462	7697.180
EGARCH(2,2)	Student's t	-3826.610	7667.220	7704.224
GARCH(1,1)	Student's t	-3835.859	7681.718	7708.149
GJR-GARCH(1,1)	Student's t	-3835.393	7682.786	7714.503
GARCH(2,1)	Student's t	-3835.836	7683.671	7715.388

Ethereum Model Selection: The results of the top 7 models are given in table 6.17.

Table 6.17 : Top 7 GARCH Model Performances for Ethereum Log Return

Model	Distribution	Log-Likelihood	AIC	BIC
EGARCH(1,1)	Student's t	-4285.422	8580.845	8607.276
EGARCH(2,1)	Student's t	-4285.312	8582.625	8614.342
EGARCH(1,2)	Student's t	-4285.422	8582.845	8614.562
GARCH(1,1)	Student's t	-4287.125	8584.250	8610.681
GARCH(2,1)	Student's t	-4286.283	8584.567	8616.284
EGARCH(2,2)	Student's t	-4285.312	8584.625	8621.628
GJR-GARCH(1,1)	Student's t	-4287.045	8586.091	8617.808

The total results of all analyzes for GARCH models are as seen in tablo 6.18

Table 6.18 : Best GARCH Models for All Data Sets

Data	Best Model	Distribution	AIC	BIC
BIST30	GJR-GARCH(1,1)	Student's t	10030.043	10064.204
BIST100	GJR-GARCH(1,1)	Student's t	9780.414	9814.574
Bitcoin	EGARCH(1,1)	Student's t	7663.556	7689.987
Ethereum	EGARCH(1,1)	Student's t	8580.845	8607.276
Apple	GJR-GARCH(1,2)	Student's t	7206.168	7245.416

Through a thorough examination of different financial assets, one can uncover fascinating patterns in the modelling of volatility. The GJR-GARCH(1,1) model proved to be the most effective for the Turkish markets, specifically BIST30 and BIST100. It revealed notable asymmetric volatility effects. In the cryptocurrency sphere, Bitcoin and Ethereum exhibited a strong correlation with EGARCH(1,1) models, indicating the significance of leverage effects and asymmetric volatility structures in the crypto markets. The GJR-GARCH(1,2) model proved to be the most accurate in capturing the returns of Apple stock, further emphasising the existence of asymmetric effects in its volatility. Across all datasets, asymmetric volatility models (GJR-GARCH and EGARCH) consistently outperform standard GARCH models. This consistent finding highlights the prevalence of unequal reactions to positive and negative shocks in financial markets, emphasising the importance of advanced modelling approaches to accurately capture these complex volatility dynamics across various asset classes and markets.

As shown in table 6.19, the Root Mean Square Error (RMSE) performance results for various GARCH models (GARCH, EGARCH, and GJR-GARCH) using both normal and Student's t distributions are presented comparatively for different financial data sets (BIST100, BIST30, Apple, Bitcoin, and Ethereum).

Table 6.19 : The Result of RMSE About Performances for Models

Data Set	Model	RMSE
BIST100	GARCH (1,1)-Normal	4.343
	GARCH (1,1)-student-t	4.031
	EGARCH (1,1)-Normal	3.915
	EGARCH (1,1)-student-t	3.429
	GJR-GARCH (2,2)-Normal	4.372
	GJR-GARCH (2,2)-student-t	4.221
BIST30	GARCH (1,1)-Normal	4.440
	GARCH (1,1)-student-t	4.195
	EGARCH (1,1)-Normal	4.047
	EGARCH (1,1)-student-t	3.589
	GJR-GARCH (2,2)-Normal	4.543
	GJR-GARCH (2,2)-student-t	4.422
Apple	GARCH (1,1)-Normal	1.599
	GARCH (1,1)-student-t	1.599
	EGARCH (1,1)-Normal	1.652
	EGARCH (1,1)-student-t	1.446
	GJR-GARCH (2,2)-Normal	1.625
	GJR-GARCH (2,2)-student-t	1.626
Bitcoin	GARCH (1,1)-Normal	4.026
	GARCH (1,1)-student-t	4.107
	EGARCH (1,1)-Normal	4.351
	EGARCH (1,1)-student-t	5.357
	GJR-GARCH (2,2)-Normal	4.163
	GJR-GARCH (2,2)-student-t	3.664
Ethereum	GARCH (1,1)-Normal	5.153
	GARCH (1,1)-student-t	6.093
	EGARCH (1,1)-Normal	5.552
	EGARCH (1,1)-student-t	5.425
	GJR-GARCH (2,2)-Normal	5.130
	GJR-GARCH (2,2)-student-t	6.034

The table 6.20 above presents a comparison between the best models selected by AIC/BIC criteria and the models with the lowest RMSE for each asset. The analysis provides intriguing insights into the volatility dynamics of various financial assets.

Table 6.20 : Comparison of AIC/BIC Best Models and Lowest RMSE Models

Data Set	AIC/BIC Best Model	RMSE (AIC/BIC)	Lowest RMSE Model	Lowest RMSE
BIST100	GJR-GARCH(1,1)	4.221	EGARCH(1,1)	3.429
BIST30	GJR-GARCH(1,1)	4.422	EGARCH(1,1)	3.589
Apple	GJR-GARCH(1,2)	1.698	EGARCH(1,1)	1.446
Bitcoin	EGARCH(1,1)	5.357	GJR-GARCH(2,2)	3.664
Ethereum	EGARCH(1,1)	5.425	GARCH(1,1)	5.153

- **BIST100 and BIST30:** The GJR-GARCH(1,1) model is recommended as the best fit for both Turkish market indices based on the AIC/BIC criteria. On the other hand, the EGARCH(1,1) model produces lower RMSE values of 3.4291 for BIST100 and 3.5892 for BIST30. This difference implies that although GJR-GARCH(1,1) achieves a satisfactory trade-off between model fit and complexity (as indicated by AIC/BIC), EGARCH(1,1) more precisely captures the asymmetric impact on volatility, resulting in improved point forecasts as measured by RMSE.

- **Apple:** The AIC/BIC criteria favor the GJR-GARCH(1,2) model, which yields an RMSE of 1.6979. However, the EGARCH(1,1) model achieves a notably lower RMSE of 1.4459. It is evident that the stock returns of Apple display notable asymmetrical volatility effects, which can be more accurately represented by the EGARCH model. The reason why AIC/BIC is preferred for GJR-GARCH(1,2) is because it provides a better fit to the data without introducing unnecessary complexity.

- **Bitcoin:** Interestingly, while AIC/BIC suggest EGARCH(1,1) as the best model, the lowest RMSE is achieved by GJR-GARCH(2,2) (3.6644 compared to 5.3571 for EGARCH(1,1)). The significant disparity in RMSE values suggests that the volatility of Bitcoin may possess a more intricate framework that the GJR-GARCH(2,2) model is able to capture more effectively. The preference for EGARCH(1,1) in terms of AIC/BIC indicates a balance between the complexity of the model and its fit.

- **Ethereum:** For Ethereum, AIC/BIC criteria recommend the EGARCH(1,1) model, but the simple GARCH(1,1) model achieves a lower RMSE (5.1526 vs 5.4248). It appears that the volatility of Ethereum may have a more balanced structure than originally thought. The EGARCH model, although it captures potential asymmetric effects, may introduce unnecessary complexity in this situation.

Our research highlights the significance of taking into account various factors when choosing a model. RMSE is specifically designed to measure forecast accuracy, while AIC/BIC consider both model fit and complexity. The variations in these criteria emphasise the intricate nature of volatility in financial markets and the difficulties involved in modelling it.

In addition, our findings suggest that different assets may necessitate unique modelling methods. Market indices such as BIST100 and BIST30, along with established stocks like Apple, can be enhanced by utilising models that effectively capture asymmetric effects, such as EGARCH. On the other hand, cryptocurrencies like Bitcoin and Ethereum display a broader spectrum of behaviour, which could be attributed to their operation in newer and less predictable markets.

It would be advantageous to conduct out-of-sample tests in future research to assess the real predictive performance of these models. In order to ensure precise volatility forecasting and efficient risk management, it is essential to take into account the unique attributes of each asset class when choosing the suitable model. This approach would allow for a more customised and accurate analysis of different financial instruments under different market conditions.

6.5 Conclusions

This research mainly investigates the predictability of various datasets important to investors in financial markets by using the time series of these datasets at certain date intervals. The findings of the experiments conducted within the scope of the research are presented with statistical evidence, and the results are interpreted. Based on these results, it is understood that a prediction cannot be made only by observing current pricing or daily price changes but also that factors such as seasonality and trend are stochastic. These stochastic price changes prevent our studies from making forecasts. From an economic point of view, the results can help investors and finance professionals to make more informed decisions by providing information about the behaviour and predictability of these markets. To summarise, neither investment planning nor risk analysis is possible with time series alone, and the dataset should be enhanced with additional features.

REFERENCES

[1] **Peres, M.R.** (2021). The history of data storage, *Scientific American*, https://www.scientificamerican.com/article/the-history-of-data-storage/.

[2] **Etymology Online**. *data | Origin and meaning of data by Online Etymology Dictionary*, https://www.etymonline.com/word/data, accessed: 2024-05-21.

[3] **Jones, B.** (2020). *Data Literacy Fundamentals: Understanding the Power and Value of Data*, Data Literacy Press.

[4] **Narayanan, A.**, **Bonneau, J.**, **Felten, E.**, **Miller, A. and Goldfeder, S.** (2016). *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*, Princeton University Press.

[5] **Nakamoto, S.** (2008). *Bitcoin: A Peer-to-Peer Electronic Cash System*.

[6] **Narayanan, A.**, **Bonneau, J.**, **Felten, E.**, **Miller, A. and Goldfeder, S.** (2016). *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*, Princeton University Press.

[7] **Percival, C. and Josefsson, S.** (2016). The scrypt Password-Based Key Derivation Function, *RFC*, *7914*, 1–16, https://api.semanticscholar.org/CorpusID:31567403.

[8] **Ali, M.**, **Nelson, J.**, **Shea, R. and Freedman, M.J.** (2016). Blockstack: A Global Naming and Storage System Secured by Blockchains, *Proceedings of the 2016 USENIX Annual Technical Conference (USENIX ATC '16)*, USENIX Association, pp.181–194, https://www.usenix.org/conference/atc16/technical-sessions/presentation/ali.

[9] **King, S. and Nadal, S.** (2012). *PPCoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake*.

[10] **Tschorsch, F. and Scheuermann, B.** (2016). Bitcoin and Beyond: A Technical Survey on Decentralized Digital Currencies, *IEEE Communications Surveys & Tutorials*, *18*(3), 2084–2123.

[11] **Schwartz, D.**, **Youngs, N. and Britto, A.** (2014). *The Ripple Protocol Consensus Algorithm*.

[12] **Chase, B. and MacBrough, E.** (2018). Analysis of the XRP Ledger Consensus Protocol, *arXiv preprint arXiv:1802.07242*, https://doi.org/10.48550/arXiv.1802.07242, 25 pages, 6 figures, 3 algorithms.

[13] **Roth, N.** (2015). An Architectural Assessment of Bitcoin, *Procedia Computer Science*, *44*, 527–536.

[14] **Hileman, G. and Rauchs, M.** (2017). *Global Cryptocurrency Benchmarking Study*, Cambridge Centre for Alternative Finance.

[15] **Tapscott, D. and Tapscott, A.** (2016). *Blockchain Revolution: How the Technology Behind Bitcoin is Changing Money, Business, and the World*, Penguin.

[16] **Buterin, V.** (2014). *A Next-Generation Smart Contract and Decentralized Application Platform*, white Paper, 3(37).

[17] **Wood, G.** (2024). Ethereum: A Secure Decentralised Generalised Transaction Ledger, **Technical Report Paris Version 705168a**, Parity, mailto:gavin@parity.io, released on 2024-03-04.

[18] **Dannen, C.** (2017). *Introducing Ethereum and solidity*, volume 1, Springer.

[19] **Diedrich, H.** (2016). *Ethereum: Blockchains, Digital Assets, Smart Contracts, Decentralized Autonomous Organizations*, Wildfire Publishing.

[20] **Antonopoulos, A.M. and Wood, G.** (2018). *Mastering Ethereum: Building Smart Contracts and DApps*, O'Reilly Media.

[21] **J.P. Morgan** (2016). *Quorum: A Permissioned Implementation of Ethereum Supporting Data Privacy*, https://github.com/jpmorganchase/quorum.

[22] **Luu, L.**, **Chu, D.H.**, **Olickel, H.**, **Saxena, P. and Hobor, A.** (2016). Making Smart Contracts Smarter, *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, Association for Computing Machinery, New York, NY, USA, p.254–269, https://doi.org/10.1145/2976749.2978309.

[23] **Hofmann, F.**, **Wurster, S.**, **Ron, E. and Böhmecke-Schwafert, M.** (2017). The immutability concept of blockchains and benefits of early standardization, pp.1–8.

[24] **Corbet, S.**, **Lucey, B.**, **Urquhart, A. and Yarovaya, L.** (2019). Cryptocurrencies as a financial asset: A systematic analysis, *International Review of Financial Analysis*, *62*, 182–199, https://www.sciencedirect.com/science/article/pii/S1057521918305271.

[25] **Krafft, P.M.**, **Penna, N. and Pentland, A.** (2018). An Experimental Study of Cryptocurrency Market Dynamics.

[26] **Casino, F.**, **Dasaklis, T. and Patsakis, C.** (2018). A systematic literature review of blockchain-based applications: Current status, classification and open issues, *Telematics and Informatics*, *36*.

[27] **Hendrickson, J.R.**, **Hogan, T.L. and Luther, W.J.** (2016). The Political Economy of Bitcoin, *Economic Inquiry*, *54*(2), 925–939.

[28] **Shanaev, S.**, **Shuraeva, A.**, **Vasenin, M. and Kuznetsov, M.** (2019). Cryptocurrency value and 51

[29] **MicroStrategy** (2021). *MicroStrategy Announces Over $1 Billion in Total Bitcoin Purchases in 2020*, `https://www.microstrategy.com/en/company/press/microstrategy-announces-over-1B-in-total-bitcoin-purchases-`

[30] **Tesla** (2021). *Tesla's $1.5 Billion Bitcoin Purchase*, `https://www.sec.gov/Archives/edgar/data/1318605/000156459021004599/tsla-10k_20201231.htm`.

[31] **Hougan, M.**, **Kim, H. and Lerner, M.** (2021). *Bitwise Asset Management: Crypto Outlook 2021*, `https://static.bitwiseinvestments.com/Research/Bitwise-2021-Crypto-Outlook.pdf`.

[32] **DeFi Pulse** (2021). *Total Value Locked (USD) in DeFi*, `https://defipulse.com/`.

[33] **Blandin, A.**, **Cloots, A.**, **Hussain, H.**, **Rauchs, M.**, **Saleuddin, R.**, **Allen, J.**, **Zhang, B. and Cloud, K.** (2019). Global Cryptoasset Regulatory Landscape Study, *SSRN Electronic Journal*.

[34] **Financial Action Task Force (FATF)** (2021). *Updated Guidance for a Risk-Based Approach to Virtual Assets and Virtual Asset Service Providers*, `https://www.fatf-gafi.org/publications/fatfrecommendations/documents/guidance-rba-virtual-assets-2021.html`.

[35] **Cumming, D.J.**, **Johan, S. and Pant, A.** (2019). Regulation of the Crypto-Economy: Managing Risks, Challenges, and Regulatory Uncertainty, *Journal of Risk and Financial Management*, *12*(3), 126, `https://doi.org/10.3390/jrfm12030126`.

[36] **Feinstein, B.D. and Werbach, K.** (2021). The Impact of Cryptocurrency Regulation on Trading Markets, *Journal of Financial Regulation*, *7*(1), 48–99, `https://ssrn.com/abstract=3649475`.

[37] **Soderberg, G.**, **Bechara, M.**, **Bossu, W.**, **Che, N.X.**, **Davidovic, S.**, **Kiff, J.**, **Lukonga, I.**, **Griffoli, T.M.**, **Sun, T. and Yoshinaga, A.** (2022). Behind the Scenes of Central Bank Digital Currency: Emerging Trends, Insights, and Policy Lessons, **Technical Report2022/004**, International Monetary Fund (IMF).

[38] **Kharpal, A.** (2021). *El Salvador becomes first country to adopt Bitcoin as legal tender after passing law*, CNBC, https://www.cnbc.com/2021/06/09/el-salvador-proposes-law-to-make-bitcoin-legal-tender.html.

[39] **Cvetkova, I.** (2018). Cryptocurrencies legal regulation, *BRICS Law Journal*, *5*, 128–153.

[40] **Shanaev, S.**, **Sharma, S.**, **Ghimire, B. and Shuraeva, A.** (2019). Taming the Blockchain Beast? Regulatory Implications for the Cryptocurrency Market, *Research in International Business and Finance*, *51*, 101080.

[41] **da Gama Silva, P.V.J.**, **Klotzle, M.**, **Pinto, A.C.F. and Gomes, L.L.** (2019). Herding behavior and contagion in the cryptocurrency market, *Journal of Behavioral and Experimental Finance*, *22*(C), 41–50, https://EconPapers.repec.org/RePEc:eee:beexfi:v:22:y:2019:i:c:p:41-50.

[42] **Platanakis, E. and Urquhart, A.** (2020). Should investors include Bitcoin in their portfolios? A portfolio theory approach, *The British Accounting Review*, *52*(4), 100837, https://www.sciencedirect.com/science/article/pii/S0890838919300605.

[43] **Ante, L.** (2021). The Influence of Bitcoin on Energy Consumption and Carbon Emissions, *Joule*, *5*(4), 805–806.

[44] **Katz, J. and Lindell, Y.** (2015). *Introduction to Modern Cryptography*, Chapman & Hall/CRC Cryptography and Network Security Series, CRC Press, Taylor & Francis Group, Boca Raton, FL, 2 edition.

[45] **Johnson, D.**, **Menezes, A. and Vanstone, S.** (2001). The Elliptic Curve Digital Signature Algorithm (ECDSA), *IJIS*, *1*, 36–63.

[46] **Rogaway, P. and Shrimpton, T.** (2004). Cryptographic hash-function basics: Definitions, implications, and separations for preimage resistance, second-preimage resistance, and collision resistance, *International Workshop on Fast Software Encryption*, Springer, pp.371–388.

[47] **Wang, X.**, **Feng, D.**, **Lai, X. and Yu, H.** (2004). Collisions for hash functions MD4, MD5, HAVAL-128 and RIPEMD, *IACR Cryptol. ePrint Arch.*, *2004*, 199.

[48] **Stinson, D.R.** (2005). *Cryptography: Theory and practice*, CRC press.

[49] **Preneel, B.**, (2010). The first 30 years of cryptographic hash functions and the NIST SHA-3 competition, Cryptographers' track at the RSA conference, Springer, pp.1–14.

[50] **Feistel, H.** (1973). Cryptography and computer privacy, *Scientific American*, *228*(5), 15–23.

[51] **Yuval, G.** (1979). How to swindle Rabin, *Cryptologia*, *3*(3), 187–191.

[52] **Bellare, M. and Kohno, T.**, (2004). Hash function balance and its impact on birthday attacks, International Conference on the Theory and Applications of Cryptographic Techniques, Springer, pp.401–418.

[53] **Van Oorschot, P.C. and Wiener, M.J.** (1999). Parallel collision search with cryptanalytic applications, *Journal of Cryptology*, *12*(1), 1–28.

[54] **Merkle, R.** (1988). A Digital Signature Based on a Conventional Encryption Function, volume293 of *Lecture Notes in Computer Science*, Springer, Berlin, Heidelberg.

[55] **Cortez, P. and Donate, J.P.** (2014). Global and decomposition evolutionary support vector machine approaches for time series forecasting, *Neural Computing and Applications*, *25*(5), 1053–1062.

[56] **Cooray, T.M.J.A.** (2008). *Applied Time Series: Analysis and Forecasting*, Alpha Science International Limited, United Kingdom.

[57] **Chatfield, C.** (2000). *Time-Series Forecasting*, CRC Press, United States of America.

[58] **Brockwell, P.J. and Davis, R.A.** (1991). *Time Series: Theory and Methods*, Springer-Verlag, 2nd edition.

[59] **Casella, G. and Berger, R.L.** (2002). *Statistical Inference*, Duxbury, Pacific Grove, CA, 2nd edition.

[60] **Hogg, R.V.**, **McKean, J.W. and Craig, A.T.** (2018). *Introduction to Mathematical Statistics*, Pearson, Boston, MA, 8th edition.

[61] **Hyndman, R.J. and Koehler, A.B.** (2006). Another look at measures of forecast accuracy, *International Journal of Forecasting*, *22*(4), 679–688, `https://www.sciencedirect.com/science/article/pii/S0169207006000239`.

[62] **Franses, P.** (2016). A note on the Mean Absolute Scaled Error, *International Journal of Forecasting*, *32*, 20–22.

[63] **Chai, T. and Draxler, R.** (2014). Root mean square error (RMSE) or mean absolute error (MAE)?, *Geosci. Model Dev.*, *7*.

[64] **Chin, W. and Marcoulides, G.** (1998). The Partial Least Squares Approach to Structural Equation Modeling, *Modern Methods for Business Research*, *8*.

[65] **Akaike, H.** (1974). A new look at the statistical model identification, *IEEE Transactions on Automatic Control*, *19*(6), 716–723.

[66] **Burnham, K.P. and Anderson, D.R.** (2002). *Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach*, Springer, latest edition edition, includes supplementary material: `https://sn.pub/extras`.

[67] **Kass, R.E. and Raftery, A.E.** (1995). Bayes factors, *Journal of the american statistical association*, *90*(430), 773–795.

[68] **Konishi, S. and Kitagawa, G.** (2008). *Information criteria and statistical modeling*, Springer, Almanya.

[69] **Said, S.E. and Dickey, D.A.** (1984). Testing for unit roots in autoregressive-moving average models of unknown order, *Biometrika*, *71*(3), 599–607.

[70] **Cheung, Y.W. and Lai, K.S.** (1995). Lag order and critical values of the augmented Dickey-Fuller test, *Journal of Business & Economic Statistics*, *13*(3), 277–280.

[71] **Box, G.E.**, **Jenkins, G.M.**, **Reinsel, G.C. and Ljung, G.M.** (2015). *Time series analysis: Forecasting and control*, John Wiley & Sons.

[72] **Lee, J.** (1996). Testing for a unit root in time series with trend breaks, *Journal of Macroeconomics*, *18*(3), 503–519, https://ideas.repec.org/a/eee/jmacro/v18y1996i3p503-519.html.

[73] **Choi, I.** (1999). Testing the null of stationarity in the presence of structural breaks, *Applied Economics Letters*, *6*(7), 413–417.

[74] **Brockwell, P.J. and Davis, R.A.** (1991). *Time Series: Theory and Methods*, Springer Series in Statistics, Springer Science & Business Media, New York, 2 edition.

[75] **Campbell, J.Y.**, **Lo, A.W. and MacKinlay, A.** (1997). *The Econometrics of Financial Markets*, Princeton University Press, http://www.jstor.org/stable/j.ctt7skm5.

[76] **Tsay, R.S.** (2005). *Analysis of financial time series*, John Wiley & sons.

[77] **Brooks, C.** (2019). *Introductory Econometrics for Finance*, Cambridge University Press, illustrated, revised edition.

[78] **Poon, S.H. and Granger, C.W.** (2003). Forecasting Volatility in Financial Markets: A Review, *Journal of Economic Literature*, *41*(2), 478–539, https://www.aeaweb.org/articles?id=10.1257/002205103765762743.

CURRICULUM VITAE

Name Surname: Mert CAN

EDUCATION:

- **B.Sc.:** 2017, Mimar Sinan Fine Arts University, Faculty of Science and Letters, Mathematics
- **M.Sc.:** 2024, Istanbul Technical University, Science and Letter Faculty, Department of Mathematics Engineering, Mathematics Engineering Programme

PROFESSIONAL EXPERIENCE :

- 2021-2022 Instructor at Istanbul Data Science Academy
- 2023-... Data Scientist at ING