Machine Learning for Financial Market Prediction

—

Engineering Data-Driven Approaches for Major Developments

Zur Erlangung des akademischen Grades eines
Doktors der Wirtschaftswissenschaften (Dr. rer. pol.)
von der KIT-Fakultät für Wirtschaftswissenschaften
des Karlsruher Instituts für Technologie (KIT)

genehmigte
DISSERTATION

von
M.Sc. Patrick Jaquart

Tag der mündlichen Prüfung: 26.10.2022
Referent: Prof. Dr. Christof Weinhardt
Korreferentin: Prof. Dr. Marliese Uhrig-Homburg

Karlsruhe, 2022

ACKNOWLEDGEMENTS

Several people positively influenced this dissertation, for who I am deeply grateful. First, I want to thank Prof. Dr. Christof Weinhardt for giving me the opportunity to pursue my PhD at his chair, his confidence in me, and his great support. Furthermore, I want to thank Prof. Dr. Marliese Uhrig-Homburg, Prof. Dr. Ali Sunyaev, and Prof. Dr. Orestis Terzidis for serving as my dissertation committee, as well as the challenging discussions and valuable comments.

Furthermore, I would like to thank my colleagues at the Karlsruhe Institute of Technology, many of whom have become great friends: Alex, Anja, Anke, Armin, Bent, Christian, David, Felix, Frederik, Leo, Marc, Marvin, Philipp, Sarah, Saskia, Tobi, and Wolfgang. Thank you for ensuring that I had a great dissertation period! Beyond that, I am grateful to David for helping me identify my research interests and for his valuable insights. Besides that, I want to thank all of my colleagues at Börse Stuttgart for their support. Special thanks go to Martin and Caro, who have successfully facilitated a constructive and pleasant working environment.

Also, I would like to thank all my friends for their support throughout the years and the lighthearted moments, which have greatly helped me reset and recharge.

Finally, I want to thank my family. I am incredibly grateful to my sister, mother, and father for their unconditional love and support.

CONTENTS

List of Figures **7**

List of Tables **9**

List of Abbreviations **11**

1. Introduction **1**
- 1.1. Motivation . . . 1
- 1.2. Research Agenda and Research Questions . . . 4
- 1.3. Thesis Structure . . . 6

2. Novel Object of Transaction: Bitcoin and Other Cryptocurrencies **9**
- 2.1. Machine Learning for Bitcoin Pricing – A Structured Literature Review 9
  - 2.1.1. Introduction . . . 9
  - 2.1.2. Methodology . . . 12
  - 2.1.3. Machine Learning and Bitcoin in Research . . . 14
  - 2.1.4. Discussion . . . 18
  - 2.1.5. Conclusion . . . 22
- 2.2. Short-Term Bitcoin Market Prediction via Machine Learning . . . 26
  - 2.2.1. Introduction . . . 26
  - 2.2.2. Related Work . . . 28
  - 2.2.3. Methodology . . . 32
  - 2.2.4. Results . . . 45
  - 2.2.5. Discussion . . . 49
  - 2.2.6. Conclusion . . . 50
- 2.3. Machine Learning for Cryptocurrency Market Prediction and Trading 52
  - 2.3.1. Introduction . . . 52
  - 2.3.2. Related Work . . . 53
  - 2.3.3. Methodology . . . 55
  - 2.3.4. Results . . . 66
  - 2.3.5. Discussion . . . 75
  - 2.3.6. Conclusion . . . 77

3. Changed Socio-Economic and Legal Environment: COVID-19 Pandemic **79**
3.1. Financial Market Prediction During the COVID-19 Pandemic Using Machine Learning . . . 79
3.1.1. Introduction . . . 80
3.1.2. Related Work . . . 81
3.1.3. Methodology . . . 82
3.1.4. Results . . . 87
3.1.5. Discussion . . . 90
3.1.6. Conclusion . . . 91

4. Changed Agent Behavior and Market Structure: The Shift from Active Towards Passive Investment **93**
4.1. The Impact of Active and Passive Investment on Fundamental Market Efficiency: A Simulation Study . . . 94
4.1.1. Introduction . . . 94
4.1.2. Related Work . . . 96
4.1.3. Methodology . . . 102
4.1.4. Results . . . 113
4.1.5. Discussion . . . 128
4.1.6. Conclusion . . . 130

5. Finale **133**
5.1. Contributions and Answers to Research Questions . . . 134
5.2. Limitations . . . 138
5.3. Future Research . . . 139

Appendices 143

A. Supplementary Material for Chapter 2.2 **145**
A.1. Supplemental Tables . . . 145
A.2. Supplemental Graphical Material . . . 151

B. Supplementary Material for Chapter 2.3 **155**
B.1. Supplemental Tables . . . 155

C. Supplementary Material for Chapter 4.1 **159**
C.1. Supplemental Tables . . . 160
C.2. Supplemental Graphical Material . . . 166
C.3. Supplemental Text . . . 167

Bibliography **169**

LIST OF FIGURES

1.1. Thesis structure 7

2.1. Bitcoin price overview 33
2.2. Neural network architecture 40
2.3. RNN structure 42
2.4. LSTM memory block 42
2.5. Study period composition and train-validation-test split 58
2.6. Creation of feature sequences and corresponding target labels for models with memory 60
2.7. Creation of tree-based and logistic regression feature sets and corresponding target labels 61
2.8. Prediction accuracy for long-short portfolio and different portfolio sizes 69
2.9. Accuracy of $k = 5$ long-short portfolio prediction by study period . . 71
2.10. Cumulative market returns and Bitcoin (BTC) returns for the full test sett date range 75

3.1. Permutation feature importance scores for the most relevant features of the full-feature random forest 90

4.1. Fundamental price deviation by fraction of active investors (relative to all active and passive investors) detailed over all random seeds in the standard setting 116
4.2. Relative fundamental price deviation by fraction of active investors (relative to all active and passive investors) detailed over all random seeds in the market setting with 40% random investment 118
4.3. Relative fundamental price deviation by fraction of active investors (relative to all active and passive investors) detailed over all random seeds in the market setting with 90% random investment 119
4.4. Fundamental price deviation by fraction of active investors (relative to all active and passive investors) detailed over all random seeds in the market settings with active investors' individual absolute target price forecasting errors (aTPE) of 0.225 and 0.9 122

4.5. Fundamental price deviation by fraction of active investors (relative to all active and passive investors) detailed over all random seeds in the market setting with individual investors' risk aversion of $\gamma = 2$ and $\gamma = 6$ 124
4.6. Fundamental price deviation by fraction of active investors (relative to all active and passive investors) detailed over all random seeds in the market setting with 0.5% transaction costs 126
4.7. Fundamental price deviation by fraction of active investors (relative to all active and passive investors) detailed over all random seeds in the market setting with individual investors' upper portfolio constraints of 0.33 and 1 128

A.1. Feature importance 1-minute memory models 151
A.2. Feature importance 5-minute memory models 151
A.3. Feature importance 15-minute memory models 151
A.4. Feature importance 60-minute memory models 152
A.5. Feature importance 1-minute no-memory models 152
A.6. Feature importance 5-minute no-memory models 153
A.7. Feature importance 15-minute no-memory models 153
A.8. Feature importance 60-minute no-memory models 154

C.1. Development of portfolio weights of active and passive agents for the composition 90-210-200, as well as the fundamental price deviation by simulation round. 166
C.2. Relative fundamental price deviation by fraction of active investors (relative to all active and passive investors) detailed over all random seeds in the market setting with 20 percent random investors. 167

LIST OF TABLES

2.1. Machine learning on Bitcoin pricing research corpus 13
2.2. Bitcoin Pricing Literature Overview 23
2.2. Bitcoin Pricing Literature Overview 24
2.2. Bitcoin Pricing Literature Overview 25
2.3. Feature overview 36
2.4. Parameter tuning space 38
2.5. Accuracy overview 46
2.6. Probabilities for true model accuracy of 50% 46
2.7. Trading results per model 48
2.8. Study periods and the respective date ranges for the training, validation, and test sets 58
2.9. Parameter tuning space and selected configuration per study period based on validation accuracy 61
2.10. Prediction accuracy for long-short portfolio and different portfolio sizes 68
2.11. Prediction accuracy for long and short legs of the long-short portfolio for different portfolio sizes 69
2.12. Prediction accuracy for long and short legs of $k = 5$ portfolio for individual study periods 71
2.13. Daily return, risk, and annualized risk-return metrics for all models and the market for the $k = 5$ long-short portfolio 73
2.14. Daily return, risk, and annualized risk-return metrics for all models and the market for the long leg of the $k = 5$ portfolio 73

3.1. Overview of the utilized feature types 84
3.2. Parameter tuning grid 86
3.3. Overview of the different model out-of-sample accuracy scores 87
3.4. Diebold-Mariano test p-values 88
3.5. Overview of the different models' precision, recall and F1-scores . . . 88
3.6. Probabilities for the different prediction models to exhibit a true model accuracy of 50% 88

4.1. Parameter overview of the standard market setting 110
4.2. Fundamental price deviation and trading volume for different investor compositions in the standard setting averaged over all random seeds . 115

4.3. Fundamental price deviation and trading volume for different investor compositions in the market settings with a share of 40% and 90% random investment averaged over all random seeds 117
4.4. Mean absolute changes of portfolio weights between simulation rounds by agent type and agent composition 119
4.5. Relative fundamental price deviation for different investor compositions in the market settings with active investors' individual absolute target price forecasting errors (aTPE) of 0.225 and 0.9 averaged over all random seeds 121
4.6. Relative fundamental price deviation for different investor compositions in the market settings with individual investors' risk aversion of $\gamma = 2$ and $\gamma = 6$ averaged over all random seeds 123
4.7. Fundamental price deviation and trading volume for different investor compositions in the market setting with 0.5% transaction cost averaged over all random seeds 125
4.8. Fundamental price deviation and trading volume for different investor compositions in the market setting with individual investors' upper portfolio constraints of $w_{max} = 0.33$ and $w_{max} = 1$ averaged over all random seeds 127

A.1. Trading class probability thresholds 145
A.2. Model accuracies per random seed 147
A.3. Diebod-Mariano test p-values 149

B.1. Overview of coins excluded from the study and the reason for their exclusion 155
B.2. Binomial test p-values 156
B.3. Binomial test p-values per study period 156
B.4. Daily return and risk metrics and annualized risk-return metrics for all models and the market (MKT) for the short leg of the k = 5 portfolio 157

C.1. Fundamental price deviation and trading volume for different investor compositions in the market setting with a share of 20 percent random investment averaged over all random seeds 160
C.2. Overview of the tests for the different stylized facts detailed over all investor compositions and random seeds 161

LIST OF ABBREVIATIONS

ANN	Artificial Neural Network
API	Application Programming Interface
aTPE	Absolute Target Price Forecasting Error
BTC	Bitcoin
CG	CoinGecko
CNN	Convolutional Neural Network
CNY	Chinese Yuan
CPU	Central Processing Unit
CVaR	Conditional Value at Risk
DL	Deep Learning
DSO	Distribution System Operator
DT	Decision Tree
E	Expected Value
ETF	Exchange-Traded Fund
EUR	Euro
FNN	Feedforward Neural Network
GARCH	Generalized Auto-Regressive Conditional Heteroskedasticity
GBC	Gradient Boosting Classifier
GPU	Graphics Processing Unit
GRU	Gated Recurrent Unit
IPO	Initial Public Offering
JPY	Japanese Yen
LR	Logistic Regression
LSTM	Long-Short Term Memory
MAD	Mean Absolute Deviation
MKT	Market
MLP	Multilayer Perceptron
MSE	Mean Squared Error
NAV	Net Asset Value
ReLU	Rectified Linear Unit
RF	Random Forest
RMSE	Root Mean Square Error
RNN	Recurrent Neural Network

SP	Study Period
T-Bill	United States Treasury Bill
TCN	Temporal Convolutional Network
TOU	Time of Use
TR	Trading Return
URL	Uniform Resource Locator
US	United States
USD	United States Dollar
UTC	Coordinated Universal Time
VaR	Value at Risk

CHAPTER 1

INTRODUCTION

1.1 Motivation

Financial markets are constantly reshaped by ever-changing socio-economic, technological, and regulatory realities. In this context, financial markets around the globe have changed significantly over the 21st century (Marszk and Lechman, 2021). These changes have affected all pivotal market elements, consisting of the socio-economic and legal environment, transaction object, market structure, agent behavior, and market outcome (Weinhardt and Gimpel, 2007). As financial markets are ubiquitous and constitute a major cornerstone of the global economic system, the transformation of financial markets is a highly relevant issue. Digitalization and the associated technological evolution have been an important factor in the transformation of financial markets in recent decades. The technological evolution has, for instance, directly affected asset values (Garleanu et al., 2012; Kogan and Papanikolaou, 2014), yielded new types of financial assets (Krückeberg and Scholz, 2019; Bianchi, 2020), and enabled new methods of data processing to estimate asset risk premia and predict asset prices (Fischer and Krauss, 2018; Gu et al., 2020).

With financial markets evolving continuously, financial research is also in constant progression. In financial research, asset pricing and financial market prediction constitute fundamental tasks. Despite the predominant notion of semi-strong form market efficiency in most financial markets (Fama, 1970), researchers have publicly identified hundreds of predictive signals for asset price movements over the last decades (Green et al., 2012; Feng et al., 2020), some of which have lost their predictive power over time (Zaremba et al., 2020). Over the last years, the number of potential predictive signals has drastically increased, since the digital age has yielded a large amount of new structured and unstructured financial data (Sagiroglu and Sinanc, 2013; Obaid and Pukthuanthong, 2021; Goldstein et al., 2021). This large amount of data render machine learning models renders complex models for data analysis necessary. Among asset pricing and financial market prediction, machine learning models have proven to incorporate a large number of features and flexibly learn high-dimensional functional relationships between features and targets (e.g., see Bianchi et al. (2020); Karolyi and Van Nieuwerburgh (2020); Gu et al. (2020, 2021)). However, open questions, for instance, regarding utilizing machine learning to analyze market predictability for specific time horizons (Goldstein et al., 2021), remain.

An example of how financial markets have changed in terms of the transaction object is the newly emerging class of digital assets. Digital assets comprise a heterogeneous set of digital products directly or indirectly connected to the blockchain (van der Merwe, 2021), which is considered an especially valuable financial technology innovation (Chen et al., 2019). The most relevant digital asset by market capitalization is the peer-to-peer electronic cash system Bitcoin (Nakamoto, 2008), which has first reached a market capitalization of more than a trillion USD in 2021. Since its inception in 2008 (Nakamoto, 2008), Bitcoin has inspired various other cryptocurrencies (Extance, 2015; Rauchs and Hileman, 2017) and gained massive research attention (Aysan et al., 2021). Despite its intended use as a means of payment, Bitcoin is also predominantly used as an investment asset (Glaser et al., 2014; Mattke et al., 2021). As Bitcoin is not backed by a central bank, has no physical utility, and does not promise future cash flows, it exhibits no fundamental value in the traditional sense. Consequently, researchers have developed unique economic frameworks to analyze the price formation of Bitcoin (Schilling and Uhlig, 2019; Bolt and Van Oordt, 2020; Biais et al., 2020). Regarding empirical Bitcoin pricing and market prediction, many of the predictive signals identified for stock markets, for instance, accounting-based signals, are not applicable. Furthermore, it remains unclear to what extent the Bitcoin pricing process is similar to the pricing processes of other financial assets and whether applicable signals, such as momentum-based signals (Jegadeesh, 1990; Jegadeesh and Titman, 1993), are relevant for the Bitcoin market. As academic research indicates that the Bitcoin market has become increasingly efficient over time (Köchling et al., 2019; Kristoufek and Vosvrda, 2019), it is crucial to fill research gaps regarding the predictability of the more mature Bitcoin market. At the same time, the flexibility of machine learning models may be especially valuable for examining the predictability of Bitcoin and also other cryptocurrencies, as there is limited evidence regarding the pricing process of these digital assets due to their recent nature.

Besides research gaps regarding the pricing of new asset types, in light of the financial market evolution, questions also arise regarding the pricing of traditional financial assets, for instance, stocks. It is essential to understand whether previously identified market predictive signals stay relevant and whether new features arise in the constantly evolving financial markets. This issue is especially relevant in times of significant change in the socio-economic and legal environment. The before-mentioned flexibility of machine learning models renders these models especially useful in these market conditions of increased uncertainty. An example of such a significant shift in the socio-economic and legal environment is the ongoing COVID-19 pandemic (Velavan and Meyer, 2020; Fauci et al., 2020), which has major implications for societies and economies around the globe. On an economic level, the COVID-19 pandemic comes with manifold supply and demand effects (Padhan and Prabheesh, 2021) and has drastically increased investor uncertainty (Zhang et al., 2020; Haroon and Rizvi, 2020). As the COVID-19 pandemic is unique in its nature, one can draw only limited parallels with other global disaster events (Borio, 2020). Therefore, from an asset pricing perspective, a crucial task is to examine the potential impact of COVID-19-related data on asset prices and financial markets.

Besides changes regarding transaction objects and socio-economic and legal environment, we also have witnessed large shifts regarding agent behavior and market structure in financial markets over the last decades. A major example is a shift from active investment toward passive investment (Blitz, 2014; Fichtner et al., 2017). For instance, the fraction of actively managed mutual funds and ETFs relative to the total fund market in the US has decreased from 81% in 2010 to 60% in 2020 (ICI, 2021). Additionally, active shares and fees of active mutual funds have fallen (Cremers and Petajisto, 2009; Stambaugh, 2014; ICI, 2020), resulting in active investment management becoming more similar to passive investment management. Due to higher expense ratios of active management, passive investment management tends to outperform active investment management after fees (Carhart, 1997; Fama and French, 2010). However, there is a need for active investment in financial markets since the security analysis of active investors connects market prices to fundamental asset values and keeps financial markets efficient (Blitz, 2014; Pedersen, 2018). Therefore, the shift from active to passive investment may reduce the information contained in individual asset prices (Sushko and Turner, 2018) and lead to higher systematic market risks (Anadu et al., 2019). As the shift from active to passive investment constitutes a major trend for global financial markets, it is essential to examine it further and analyze its implications.

Summarizing, the overarching objective of the thesis at hand is to develop a comprehensive understanding of relevant and current asset pricing challenges induced by significant changes regarding pivotal market elements through multiple in-depth quantitative analyses.

1.2 Research Agenda and Research Questions

This thesis sheds light on current asset pricing issues by raising five individual research questions. The emergence of the cryptocurrency Bitcoin motivates the search for new prediction models and predictive signals for the Bitcoin market. As the Bitcoin market is relatively young, there only is limited evidence regarding its price formation process. Hence, machine learning models, which enable the incorporation of many potential features and do not require specific ex-ante modeling of the functional form of how features enter into the target, might be well suited to address this challenge. For the Bitcoin market, especially the short-term predictability of the Bitcoin market remains under-researched Jaquart et al. (2020a). Hence, Research Question 1 addresses the performance of machine learning models to create short-term predictions of the Bitcoin market. Research Question 2 refers to which feature types are relevant for such prediction models.

Research Question 1 *What is the predictive power of machine learning models predicting short-term movements of the Bitcoin market?*

Research Question 2 *What are the most relevant features for predicting short-term movements of the Bitcoin market using different machine learning models?*

While Bitcoin is the most relevant cryptocurrency by market capitalization, it has inspired thousands of other cryptocurrencies (Extance, 2015; Rauchs and Hileman, 2017), some of which have also reached massive valuations. For instance, as of May 2022, more than 50 cryptocurrencies exhibit a market capitalization of more than one billion USD. Due to the recent inception of these cryptocurrencies, there still may exist certain inefficiencies in the market which could be discovered and economically exploited by utilizing complex machine learning models. Building on the answers to the previous research questions, Research Question 3 refers to the economic potential of applying well-suited and advanced machine learning methods to cryptocurrency market prediction.

Research Question 3 *What is the performance of machine learning models for generating statistical arbitrage in the cryptocurrency market?*

In shifting the focus from a new object of transaction in financial markets towards a change in the socio-economic environment of financial markets, the next research question addresses the impact of the COVID-19 pandemic on the pricing of stocks. As the pandemic has ample effects on businesses (Padhan and Prabheesh, 2021) and investor sentiment (Zhang et al., 2020), new pandemic-related risk factors or anomalies might have emerged in the stock market. As there is limited evidence regarding such new factors' nature, machine learning may again be well suited to examine potential factors due to their flexibility. Thus, Research Question 4 addresses the existence of potentially new pandemic-related stock market predictive signals.

Research Question 4 *What is the predictive power of machine learning models predicting S&P 500 stock price movements during the COVID-19 pandemic?*

Another development regarding pivotal market elements of financial markets, namely market structure and agent behavior, is the global shift from active towards passive investment (Fichtner et al., 2017). As the information collection and evaluation conducted by active investors keeps market prices connected to fundamental values (Pedersen, 2018), this shift may have considerable implications for the deviation of market prices from fundamental values - fundamental price efficiency. In addition to shifts between different types of investments, individual investment styles are also evolving. In line with the findings that machine learning models can improve financial market forecasts (Fischer and Krauss, 2018; Rasekhschaffe and Jones, 2019; Gu et al., 2020), a large part of professional active investors have started machine learning models for trading and portfolio management (BarclayHedge, 2018; Petropoulos et al., 2022). In light of this development, it is important to also evaluate the impact of differences in the accuracy of active investors's market forecasts on fundamental market efficiency. Therefore, Research Question 5 addresses the shift from active to passive investment and aims to shed light on its effect on fundamental price efficiency, while Research Question 6 refers to the implications of varying accuracy levels of active investors' market predictions.

Research Question 5 *How do different levels of active and passive investment affect fundamental price efficiency?*

Research Question 6 *How do different degrees of accuracy of active investors' market forecasts affect fundamental market efficiency?*

1.3 Thesis Structure

The structure of this thesis is illustrated in Figure 1.1. After this introduction, Chapter 2 examines the application of machine learning models for the market prediction of Bitcoin and other cryptocurrencies. Concretely, Chapter 2.1 provides a structured literature overview of studies and existing gaps in the research field of Bitcoin pricing. Chapter 2.2 incorporates these findings and presents a comprehensive study on short-term Bitcoin market prediction via various state-of-the-art machine learning methods, including a thorough model-agnostic feature importance analysis. Chapter 2.3 analyzes the potential of using advanced machine learning models specialized in time-series forecasting to generate statistical arbitrage in the cryptocurrency market.

After examining the pricing process of the new transaction objects in financial markets, Chapter 3 sheds light on the pricing of traditional assets in an altered socio-economic environment. It applies various machine learning models to analyze the predictability of daily stock price movements in the S&P 500 during the COVID-19 pandemic by utilizing a feature set comprising COVID-19-related features and traditional risk factors.

Chapter 4 studies the implications of a major trend regarding agent behavior and market structure in financial markets. It investigates the implications of the shift

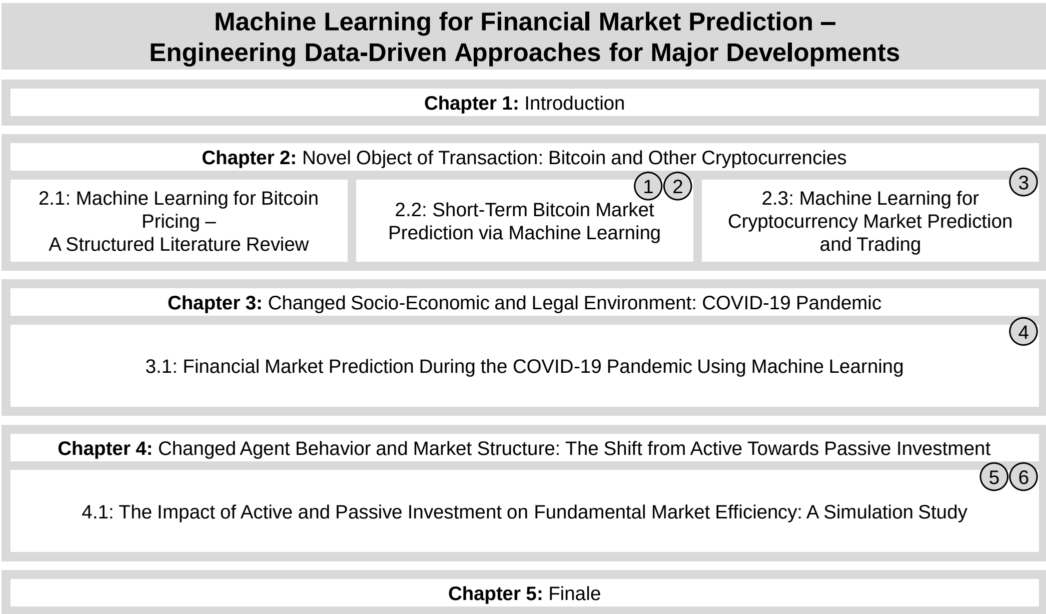


Figure 1.1.: Structure of this thesis. Circles map Research Questions 1 to 6 to Chapters 2 to 4.

from active to passive investment for fundamental price efficiency. The chapter introduces a simulated financial market in which active, passive, and random investors repeatedly issue orders to trade stocks. It thoroughly analyzes the impact of different investor compositions and other key market parameters on fundamental price efficiency.

Finally, Chapter 5 concludes this thesis by summarizing the contributions and discussing avenues for future research.

Chapters 2 to 4 are based on or contain published articles or working papers. I clearly indicate this at the beginning of the respective chapter in every case. I refer to the authors as "we" throughout these chapters because I collaborated with other researchers for these articles.

CHAPTER 2

NOVEL OBJECT OF TRANSACTION: BITCOIN AND OTHER CRYPTOCURRENCIES

2.1 Machine Learning for Bitcoin Pricing – A Structured Literature Review

This chapter sheds light on market prediction via machine learning regarding the novel transaction object Bitcoin and cryptocurrency. To provide an adequate understanding of the application of machine learning to Bitcoin market prediction and provide a foundation for the other studies of Chapter 2, the following section includes a structured literature review on the research field. The literature review analyses the existing body of literature and structures it according to four different concepts. It derives guidelines for future publications in the field to ensure a sufficient level of transparency and reproducibility. Furthermore, the review shows that research on Bitcoin market prediction via machine learning is highly diverse and that the results of several studies can only be compared to a limited extent.

This section, in large parts, comprises the published article: P. Jaquart, D. Dann, C. Martin, *Machine Learning for Bitcoin Pricing – A Structured Literature Review*, Proceedings of 15th International Conference on Business Information Systems Engineering, pages 174-188, 2020.

2.1.1 Introduction

Bitcoin is a digital peer-to-peer cash system introduced by Nakamoto in 2008 (Nakamoto, 2008). Its underlying technology blockchain is referred to as “trust machine” (The Economist, 2015) due to its three central properties: secure transfer of information through a cryptographic protocol, a distributed database, and a decentralized consensus mechanism (Beck et al., 2017). Benefiting from the stated innovative properties (cf., traditional currencies and payment systems), Bitcoin quickly gained relevance in academic literature as well as for the global financial system (Böhme et al., 2015). As of November 2019, Bitcoin has a market capitalization of over 155 billion USD, which corresponds to more than 66 of the whole cryptocurrency market (Coinmarketcap, 2020).

Researchers are investigating a variety of topics in connection with Bitcoin markets, for instance asset type, asset pricing, hedging, and market efficiency. Within the research branch of Bitcoin pricing, there are several different smaller streams of research. Some researchers (Bolt and Van Oordt, 2020; Pagnotta and Buraschi, 2018; Biais et al., 2020; Schilling and Uhlig, 2019) work on creating and validating theoretical economic models, while other researchers concentrate on empirical asset pricing. In this work, we analyze and structure the body of literature on empirical Bitcoin pricing via machine learning. Thereby, we use the term "bitcoin pricing" for the forecasting of target values based on the Bitcoin price (e.g., price, absolute price change, or return). For empirical Bitcoin pricing, market predictive features might consist of priced risk factors (which might be identified through reviewing theoretical economic models (Bolt and Van Oordt, 2020; Pagnotta and Buraschi, 2018; Biais et al., 2020; Schilling and Uhlig, 2019)) or other factors based on possible market inefficiencies. Empirical Bitcoin pricing is of explicit economic relevance, as accurate prediction models enable the employment of profitable trading strategies. Advances in computing technology, open-source implementation tools, and a skyrocketed Bitcoin price, boosted the scientific community's interest in the employment of machine learning methods for Bitcoin pricing. Searching google scholar for the terms "bitcoin" and "machine learning" yields over 8200 results.

Due to the novelty of the Bitcoin technology, the research on predictive features for the Bitcoin price is still in its early stages, and findings of several researchers indicate that Bitcoin might represent a new asset class (Glaser et al., 2014; Dyhrberg, 2016; Burniske and White, 2017). Therefore, classical market predictive signals from other asset classes (e.g., stocks (Green et al., 2012)) are only partly applicable to Bitcoin pricing. Most machine learning approaches demonstrate the ability to flexibly incorporate a large number of features (e.g., (Cybenko, 1989; Hornik et al., 1989; Hammer, 2000; Cortes and Vapnik, 1995; Breiman, 2001)). Together with the availability of large amounts of multidimensional data, this flexibility might render machine learning methods suitable for Bitcoin pricing. This flexibility is especially important, since the stream of research on Bitcoin pricing is still young and there exists limited guidance in the scientific literature about the nature of the Bitcoin price formation process. For the course of this work, we adopt the definition of machine learning from Gu et al. (2020), who apply machine learning to predict excess returns of stocks. They use the term machine learning "to describe (i) a diverse collection of high-dimensional models for statistical prediction, combined with (ii) so-called 'regularization' methods for model selection and mitigation of overfitting, and (iii) efficient algorithms for searching among a vast number of potential model specification" (Dyhrberg, 2016, pp. 2f). From the wide range of definitions, the chosen one stands out due to its broad scope, which allows us to consider a large variety of approaches (e.g., linear models). Since the spectrum of employed machine learning methods and models used is rather broad, analyzing and comparing the different approaches remains a challenging task.

Against this backdrop, we argue it is time to take a step back and evaluate the current status quo. With our literature review, we provide an overview of current research on Bitcoin pricing via machine learning. In so, we identify common methods, types of analysis, and findings. To our best knowledge, there is no comprehensive overview examining the diverse branch of research in the context of machine learning for Bitcoin pricing. Therefore, we seize the opportunity to take this step back, assess the current state of research in this field, and outline potentials for future research.

Doing so, our contribution is threefold. First, we provide researchers in this field an overview of already existing work, identify recurring patterns and remaining niches to be occupied. Second, we identify which methods appear promising for the Bitcoin pricing problem based on the evaluated body of literature. Third, we develop reporting guidelines for future research to enhance transparency and accelerate scientific progress.

The remainder of this work is structured as follows. Section 2 introduces the employed methodology for our structured literature review and provides summary statistics. In Section 3, we analyze the existing body of literature. Section 4 discusses prevalent shortcomings, theoretical and practical implications, and paves the way for future research. Eventually, Section 5 concludes this work.

2.1.2 Methodology

Our literature search follows the suggestions by Webster and Watson (2002) and Vom Brocke et al. (2009). We build our initial literature base by querying a broad set of interdisciplinary research databases (i.e., *ACM Digital Library, AIS eLibrary, Business Source Premier, emerald insight, IEEE, ProQuest, SAGE Journals, ScienceDirect/Scopus, Taylor & Francis Online, Web of Science*). We query those databases for matching our search term in title, abstract, or keywords (Vom Brocke et al., 2009). We adopt the machine learning definition by Gu et al. (2020), which is rather broad as it also includes linear models (e.g., linear regressions). By April 2019, this yields an initial set of 101 publications for further review. Analyzing each publication's title and abstract, we exclude 76 papers, which do not explicitly match the scope of our literature review. This may be due to (i) papers, employing methods not matching the machine learning definition of Gu et al. (2020), (ii) papers not focusing on the prediction of Bitcoin price/return (e.g., volatility), (iii) papers not being available in English, or (iv) papers not employing a prediction task (e.g., not using a time lag between predictive variables and target). Subsequent forward and backward search with the remaining relevant papers yields additional eight articles resulting in a total of 33 papers for in-depth review. Table 2.1 documents the number of identified articles for each database.

Next, we derive key concepts for paper categorization. Therefore, we initially screened a set of 10 papers consisting of the most recent peer-reviewed conference proceedings and journal papers, as we assumed that recent papers incorporate findings of previous works. Three researchers independently reviewed each of these papers and developed an initial set of concepts for classification. Throughout the entire paper screening process, we evaluated these initial concepts and adapted them as required. Subsequent discussion and synthesis of all identified concepts led to a

Table 2.1.: Machine learning on Bitcoin pricing research corpus

Data Base	*#Paper*
ACM Digital Library	3
AIS eLibrary	1
Business Source Premier	2
emerald insight	0
IEEE	10
ProQuest	0
SAGE Journals	0
ScienceDirect/Scopus	8
Taylor & Francis Online	0
Web of Science	1
Forward / Backward Search	8
Total	33

final set of four distinct concepts:

- *Method* (i.e., *multilayer perceptrons, recurrent neural networks, regression-based* models, *support vector machines, tree-based* models)
- *Features* (i.e., *technical, blockchain-based, sentiment- and interest-based, asset-based*)
- *Prediction Interval* (i.e., *second, minute, hour, day, week*)
- *Prediction Type* (i.e., *classification, regression*)

The subsequent paper classification process to one or more of the identified concepts followed a similar process — the classification was initially conducted independently and discrepancies were discussed afterward. The categorization guidelines for each researcher allowed a non-exclusive categorization (i.e., each article can be assigned to multiple categories). Table 2.2 summarizes all reviewed papers and specifies the assigned concepts. All reviewed papers have been published within the last 5 years: 2019 (5), 2018 (17), 2017 (2), 2016 (4), 2015 (4), and 2014 (1).

2.1.3 Machine Learning and Bitcoin in Research

Bitcoin, representing the most popular crypto asset (Coinmarketcap, 2020), has received a considerable amount of research attention since its inception in 2009. We use four different concepts to analyze and structure the literature, namely predictive *features*, *type* of prediction problem, prediction intervals, and machine learning methods. These identified concepts are rather broad and are applicable to a multitude of prediction tasks. However, some of the concept characteristics (e.g., blockchain-based features) are specific to the Bitcoin pricing problem. Since the models analyze different time horizons, have different parameter specifications, and are evaluated using different evaluation metrics, it remains infeasible to compare them *across* different papers. Yet, comparing different machine learning models *within* the same paper remains possible, since they, among other aspects, use the same data. However, even the comparison of models *within* the same paper remains only valid under the assumptions that (i) all models are equally optimal tuned, and (ii) the selected time window is representative of Bitcoin's price formation process. Table 2.2 provides an overview of the analysis of the different papers and concepts.

2.1.3.1 Machine Learning Methods

The analyzed body of literature leverages a multiplicity of different machine learning methods. We group the literature into five categories based on the introduced models. We differentiate *multilayer perceptrons, recurrent neural networks, regression-based* models, *support vector machines*, and *tree-based* models.

Multilayer perceptrons represent a type of feedforward neural network and consist of one input layer, one or more hidden layers, and one output layer (Cybenko, 1989; Hornik et al., 1989). In feedforward networks, information only flows into one direction. *multilayer perceptrons* with a non-linear and differentiable activation function can approximate any non-linear function, rendering them universal approximators (Hornik, 1991). Eight of the reviewed papers use *multilayer perceptrons.*

Recurrent neural networks drop the requirement for acyclic graphs from *multilayer perceptrons*, allowing for arbitrary feedback connections of the network (Rumelhart et al., 1988). Hammer (2000) shows that *recurrent neural networks* with a sufficient number of hidden nodes and non-linear activation function also satisfy the requirements of a universal approximator. Ten papers use *recurrent neural networks*, and they remain the best reported model in all papers that use benchmarked scenarios.

Regression-based models refer to models based on linear regressions (e.g., logistic regressions, lasso regressions, and vector autoregressions). Sixteen papers employ regression-based models, which often serve as a reference point for more sophisticated machine learning methods.

The underlying idea of *support vector machines* consists of minimizing generalization error through constructing a (set of) hyperplane(s) in a high-dimensional space (Cortes and Vapnik, 1995; Vapnik, 1995; Drucker et al., 1997). Six of the reviewed publications employ *support vector machines*.

Last, seven papers employ *tree-based models*. In these models, the outcomes are cuboid regions with axis-aligned edges (Breiman et al., 1984). A frequently used implementation of the methodology is the random forest, which constitutes an ensemble of imperfectly correlated trees to reduce the variance of forecasts (Breiman, 2001).

Eleven papers employ methods that are part of none of the five major categories (e.g., fuzzy-systems (Atsalakis and Valavanis, 2009)).

2.1.3.2 Market-predictive Features

Literature on Bitcoin pricing via machine learning uses a multiplicity of market predictive signals. While, for instance, technical features (e.g., historical returns) are used in the literature on pricing traditional financial assets (Jegadeesh, 1990; Jegadeesh and Titman, 1993), blockchain-based features (e.g., mining difficulty) are specifically related to cryptocurrencies — in particular Bitcoin. Unlike stocks, bonds or other financial assets, Bitcoins exhibit no fundamental value in a typical sense as they do not promise future cash flows, are not backed by a central bank, and cannot be utilized physically. Due to these different characteristics of Bitcoin, it is not possible to use the same feature categorizations as for other financial assets. Based on the reviewed literature, we categorize market predictive features into *technical*, *blockchain-based*, *sentiment-* and *interest-based*, and *asset-based*.

Technical features include past data of the Bitcoin market, for instance, historical prices or trading volumes. *Technical* features are the most frequently used features in the reviewed literature (27 models).

Blockchain-based features refer to data from the Bitcoin blockchain, for instance, mining difficulty or the number of transactions per block. Nine papers use *blockchain-based* features.

Sentiment- and interest-based features relate to social media sentiment and internet search volume, for instance, twitter sentiment or google trends data. Ten papers employ this type of feature.

Asset-based features relate to prices and returns of commodities and financial assets other than Bitcoin, for instance, oil or stock market prices. *Asset-based* features are used in nine papers.

Features not covered by one of the presented categories are categorized as *other* features. Among these, Demir et al. (2018) use economic policy uncertainty, Aysan et al. (2019) use geopolitical risks, Hotz-Behofsits et al. (2018) use GPU prices from Amazon's bestseller lists. Phaladisailoed and Numnonda (2018), as well as Mallqui and Fernandes (2019), use timestamps. Demir et al. (2018) and Aysan et al. (2019) conclude that Bitcoin may serve as a hedge against policy uncertainty and geopolitical risks, respectively.

2.1.3.3 Prediction Interval

The authors in the reviewed literature use different prediction intervals to price Bitcoin. The term "prediction interval" thereby denotes the frequency at which a model makes new predictions. The prediction intervals in the reviewed literature range from 5 seconds up to 1 week. Based on the prediction intervals, we group the models into five categories — *second, minute, hour, day*, and *week*.

Second includes models with prediction intervals of less than a minute (3 papers), *minute* between a minute and less than an hour (5 papers), *hour* between 1 hour and less than a day (3 papers), *day* between 1 day and less than 1 week (26 papers), and *week* includes models with prediction intervals of 1 week or longer (1 paper). Smuts Smuts (2019) tests multiple models with prediction intervals ranging from 1 hour to 1 week and finds that the model with the highest prediction accuracy for Bitcoin prices has a prediction accuracy of 1 week. Madan et al. (2015) directly compare prediction intervals of 10 seconds and 10 minutes and find a slightly higher prediction accuracy for the prediction interval of 10 minutes.

2.1.3.4 Prediction Types

There are several options to set up the prediction problem for Bitcoin pricing. First, we distinguish between prediction problems formulated as a *regression* or *classification* problem. Bitcoin prices and returns are numerical and continuous variables. Hence, it is possible to formulate a *regression* model, which tries to predict the exact values of these target variables. However, one can reduce the *regression* problem into a *classification* problem by creating classes based on the target variable. In this case, the prediction model attempts to predict class affiliations based. Second, we distinguish the literature based on whether *absolute* Bitcoin price levels or *relative* price changes are predicted. Traditional financial literature on other financial assets (e.g., on stocks (Green et al., 2012)) usually analyzes *relative* price changes.

The reviewed literature formulates the Bitcoin pricing problem 14 times as a *classification* problem in and 21 times as a *regression* problem. Some scholars create multiple models and set up the prediction problem as both a *classification* problem and a *regression* problem (Mallqui and Fernandes, 2019; Greaves and Au, 2015). For *classification* problems, nine of 14 cases formulate it as a binary *classification* problem, predicting the sign (i.e., positive or negative) of the Bitcoin price change. In contrast, three papers split the Bitcoin price change into three classes (i.e., positive, neutral, negative). Beyond that, Nakano et al. (2018) create four target classes based on the price change quantiles, and Huang et al. (2019) create 21 classes based on different Bitcoin return intervals. All papers that use *classification* models create target classes based on *relative* price changes, while 17 of the 21 papers that use *regression* models predict *absolute* Bitcoin price levels and only four of these papers predict *relative* price changes.

2.1.4 Discussion

Overall, the research on Bitcoin pricing via machine learning is not at a mature state yet. This may be due to the novelty of the protocol itself (Nakamoto, 2008), and that machine learning techniques require a substantial amount of data to learn relationships between features and target variables. An explicit limitation of the reviewed work is that none of the papers is published in a top-rated finance or information systems journal (VHV, 2019). Furthermore, a considerable amount of available literature barely meets academic standards in terms of transparent documentation of applied method and results. This includes, for instance, studies reporting unlikely R^2 values for four different methods within the range of .991 and .992 (Phaladisailoed and Numnonda, 2018). An R^2 of this magnitude is fairly unusual compared to the rest of the reviewed literature and might indicate setup problems (e.g., the use of unlagged features or a high similarity between features and target). In so, further shortcomings in the documentation render it impossible to reproduce and verify the empirical analyses at all. These include, not explicitly reporting the analyzed time range (Atsalakis et al., 2019; Rahman et al., 2019), data split (Wu et al., 2019), or machine learning setup (e.g., layer structure, activation function, loss function, learning function) (Khaldi et al., 2018; Lahmiri and Bekiros, 2019; Karakoyun and Cibikdiken, 2018). Furthermore, inconsistencies in the reporting prohibit reproducing the empirical tests. These inconsistencies can stem from reporting to optimize the number of units in a hidden layer of a multilayer perceptron within a specific range and using a number outside of that range in the final model (Almeida et al., 2015) or setting up a regression problem, but using the accuracy metric for model evaluation without further explanation (Pant et al., 2018).

Throughout the literature, the machine learning models are built and evaluated on rather short time periods and small data samples. A choice of longer prediction intervals, (e.g., weekly intervals (Smuts, 2019)) in combination with advanced machine learning models and a large number of features might result in an insufficient number of data points in the sample (Arnott et al., 2019). Furthermore, test splits of 3% or less, corresponding to 60 observations or less, limit the generalizability of the reported results (Atsalakis et al., 2019; Karakoyun and Cibikdiken, 2018).

2.1.4.1 Theoretical Implications

Researchers apply a wide variety of methods and underlying architectures with alternating success, such as *artificial neural networks*, *recurrent neural networks*, *regressions-based* models, *tree-based* methods, and *support vector machines*. Their main objectives are accurately predicting the Bitcoin price (absolute or relative) using *classification* and *regression* approaches. The models embody a broad spectrum of features, which relate to *technical*, *blockchain-based*, *sentiment- and interest-based*, and *assets-based* aspects. Most researchers use *technica*l features for their models. Only few authors (Poyser, 2019; Ciaian et al., 2016; Georgoula et al., 2015) use features from all four categories. Since 2017, scholars begin to consider features beyond these main categories (e.g., economic policy uncertainty (Demir et al., 2018)).

Researchers formulate *regression* and *classification* problems equally often until the end of 2017, while from 2018 onwards there is a slight shift towards a higher share of *regression* problems. Consequently, researchers in the field mostly (i.e., 60%) utilize *regression-based* methods in total.

The majority (i.e., 79%) of models are set up with *daily* prediction intervals. The relative share of these *daily* models further increased after 2017. However, varying time horizons and model specifications limited the comparability of methods *across* different papers. Importantly, this resonates with limited options to validate any trading strategies applied. To ensure a certain level of comparability (e.g., uniform time horizons), we focus on comparisons of different methods *within* the same paper. Nevertheless, as they are based on several assumptions (e.g., representative time windows and equally optimal tuning states of different models), these comparisons are limited. None of the authors have published their machine learning model, which would allow future researchers to train the model on new data and compare the performance to other methods. Additionally, there are no widely established guidelines or best practices in this research stream for reporting machine learning models.

Given these limitations, we find that recurrent neural networks, and in particular long-short term memory neural networks, perform well in the Bitcoin pricing problem compared to other methods (Phaladisailoed and Numnonda, 2018; Mallqui and Fernandes, 2019; Wu et al., 2019; Khaldi et al., 2018; Lahmiri and Bekiros, 2019;

Karakoyun and Cibikdiken, 2018; McNally et al., 2018). Interestingly, even though long short-term memory neural networks were published in 1997 already (Hochreiter and Schmidhuber, 1997), the first paper (McNally et al., 2018) taking these into account is from 2018.

2.1.4.2 Practical Implications

Based on the finding that complex network architectures such as recurrent neural networks yield promising results (Phaladisailoed and Numnonda, 2018; Mallqui and Fernandes, 2019; Wu et al., 2019; Khaldi et al., 2018; Lahmiri and Bekiros, 2019; Karakoyun and Cibikdiken, 2018; McNally et al., 2018), future research should evaluate further sophisticated network architectures for this particular problem. This may include assessing the effectiveness of ordinary convolutional neural networks LeCun et al. (1995), as well as dilated convolutional neural networks (Yu and Koltun, 2016). The latter has proven to provide promising results in forecasting S&P 500 stock market index already (Borovykh et al., 2018). However, more sophisticated models require more data (Arnott et al., 2019), which might be achieved, for instance, by considering shorter prediction intervals.

Beyond identifying appropriate modeling architecture, the process of model reporting demands for refinement and harmonization. Contrasting research from (bio)medical research (Luo et al., 2016) or psychology (Wilkinson, 1999), the analyzed research follows no established guidelines for uniformly reporting machine learning results. We recommend the following reporting standards for future research in the field of Bitcoin pricing via machine learning and machine learning projects in general. First, we propose that researchers are required to either publish their model and data to an open research repository (e.g., CORE CORE (2019), Open Research Library ANU Repository (2019)) reveal or to document the entire model configuration (i.e., hyperparameters) and data collection process in a structured manner. This may include a distinct table providing information about the number of a multilayer perceptron's hidden layers, number of units per layer, activation/loss functions, or optimizers. Second, we propose that researchers who publish new modeling approaches benchmark their models against other existing models from the field on their utilized dataset. Currently, there is no established benchmarking dataset. However, researchers commonly use benchmarking datasets (e.g., MNIST for handwritten digits) in other machine learning fields. Overall, the guidelines were developed due to shortcomings in the existing Bitcoin pricing literature and are therefore of particular importance in this specific field. However, they apply to empirical machine learning studies of various domains.

2.1.4.3 Limitations

There are three main limitations of the presented analysis. First, machine learning and Bitcoin pricing are two fast-evolving research disciplines. Therefore, our work reflects a quick blink in time of the literature in this field, and future analysis may yield different results. Moreover, the scope of our literature search is limited, as there exists no unique and widespread acceptance of the term "machine learning." Additionally, this review suffers from the low quality (insufficient documentation and data samples) from part of the Bitcoin pricing literature. Furthermore, we may speculate about the existence of more accurate machine learning models, which are exploited monetarily rather than contributed to the scientific body of literature.

2.1.4.4 Future Research

We encourage future researchers in the field to evaluate advanced machine learning models (e.g., dilated convolutional neural networks (Yu and Koltun, 2016)) for time series forecasting, which are not considered by contemporary research in this field. Furthermore, researchers should shed light on aspects of Bitcoin pricing that have not been sufficiently addressed in the existing literature (e.g., short-term Bitcoin market prediction with prediction horizons of less than a day and feature importance analysis on models utilizing a comprehensive feature set). Theoretical economic models for Bitcoin prices (Bolt and Van Oordt, 2020; Pagnotta and Buraschi, 2018; Biais et al., 2020; Schilling and Uhlig, 2019) might help to guide the search for further predictive features. To enable and accelerate scientific progress in the field, we propose that future researchers report all model configurations in a structured way and benchmark new models against other reported models.

2.1.5 Conclusion

Bitcoin has received a considerable amount of interest from researchers and investors since its inception in 2008. The research on Bitcoin pricing via machine learning constitutes a relevant and emerging topic. We review the existing body of literature of this research branch based on the guidelines of Webster and Watson (2002) and Vom Brocke et al. (2009). We structure and analyze the body of literature according to four different concepts, namely *method*, *feature*, *prediction interval*, and *prediction type*. A comparison of methods within the same paper indicates that *recurrent neural networks* might be well suited for the prediction problem. Most researchers use features from four categories, namely *technical*, *blockchain-based*, *sentiment*- and *interest-based*, and *asset-based*. Across the reviewed literature, we find a lack of transparency and comparability, limiting options to validate and reproduce model results and eventually applied trading strategies.

Based on these issues we propose that future researchers reveal their data collection process and all relevant model configurations in a structured way or benchmark their model against other published models.

Table 2.2.: Literature overview. Best method marked with bold cross (based on accuracy or lowest error). For papers using classification and regression: [A]: best method for the classification problem, [B]: best method for the regression problem. For papers in which an ensemble consisting of multiple methods achieves the best results: [C]: methods applied

	Method						**Features**					**Interval**	**Type**	
Source	Multilayer Perceptors	Recurrent Neural Networks	Regression-based	Super Network	Tree-based	Other	Technical	Blockchain-based	Sentiment-/interest-based	Asset-based	Other	S: Secondly, M: Minutely, H: Hourly, D: Daily, W: Weekly	Classification	Regression
(Almeida et al., 2015)	x						x					D	x	
(Amjad and Shah, 2017)			x		**x**	x	x					S	x	
(Atsalakis et al., 2019)	x					**x**	x					D		x
(Aysan et al., 2019)			x								x	D		x
(Cerda et al., 2019)		x					x		x			M, D		x
(Ciaian et al., 2016)			x				x	x	x	x		D		x
(Demir et al., 2018)			x				x				x	D		x
(Georgoula et al., 2015)			x				x	x	x	x		D		x
(Giudici and Abu-Hashish, 2019)			x				x			x		D		x
(Greaves and Au, 2015)	**x**[A]		**x**[B]	x			x	x				H	x	x
(Hegazy and Mumford, 2016)			**x**		x	x	x					M	x	

Table 2.2.: Literature overview. Best method marked with bold cross (based on accuracy or lowest error). For papers using classification and regression: ^{A}: best method for the classification problem, ^{B}: best method for the regression problem. For papers in which an ensemble consisting of multiple methods achieves the best results: ^{C}: methods applied

	Method						**Features**					**Interval**	**Type**	
Source	Multilayer Perceptors	Recurrent Neural Networks	Regression-based	Super Network	Tree-based	Other	Technical	Blockchain-based	Sentiment-/interest-based	Asset-based	Other	S: Secondly, M: Minutely, H: Hourly, D: Daily, W: Weekly	Classification	Regression
(Hotz-Behofsits et al., 2018)						x			x	x	x	D		x
(Huang et al., 2019)					**x**		x					D	x	
(Jain et al., 2018)			x						x			H		x
(Jang and Lee, 2017)	**x**		x	x			x	x		x		D		x
(Karakoyun and Cibikdiken, 2018)		**x**	x				x					D		x
(Khaldi et al., 2018)		**x**				x	x					D		x
(Kim et al., 2016)						x	x		x			D	x	
(Lahmiri and Bekiros, 2019)	x	**x**					x					D		x
(Madan et al., 2015)				x	**x**	x	x	x				S, M, D	x	
(Mallqui and Fernandes, 2019)	x	**x**^{A}		**x**^{B}	**x**^{A}	x	x	x		x	x	D	x	x
(McNally et al., 2018)		**x**	x				x	x				D		x
(Nakano et al., 2018)	x						x					M	x	
(Pant et al., 2018)		x					x		x			D		x

Table 2.2.: Literature overview. Best method marked with bold cross (based on accuracy or lowest error). For papers using classification and regression: [A]: best method for the classification problem, [B]: best method for the regression problem. For papers in which an ensemble consisting of multiple methods achieves the best results: [C]: methods applied

	Method						**Features**					**Interval**	**Type**	
Source	Multilayer Perceptors	Recurrent Neural Networks	Regression-based	Super Network	Tree-based	Other	Technical	Blockchain-based	Sentiment-/interest-based	Asset-based	Other	S: Secondly, M: Minutely, H: Hourly, D: Daily, W: Weekly	Classification	Regression
(Phaladisailoed and Numnonda, 2018)		**x**	x				x				x	D		x
(Poyser, 2019)						x	x	x	x	x		D		x
(Rahman et al., 2019)			x	x	x	**x**			x			D	x	
(Shah and Zhang, 2014)			x				x					S		x
(Sin and Wang, 2018)	x						x	x				D	x	
(Smuts, 2019)		x					x		x			H, D, W	x	
(Sun et al., 2018)				x	**x**					x		D	x	
(Tupinambás et al., 2018)						x				x		M	x	
(Wu et al., 2019)		**x**[C]	**x**[C]				x					D		x
$\sum$	8	10	16	6	7	11	27	9	10	9	5	S:3, M:5, H:3, D:26, W:1	14	21

2.2 Short-Term Bitcoin Market Prediction via Machine Learning

With a general understanding of the current state of research in the field of Bitcoin market prediction, this section addresses some of the previously identified research gaps by shedding light on the short-term predictability of the Bitcoin market. In the following study, various machine learning architectures are optimized and trained to predict short-term movements of the Bitcoin market, utilizing a comprehensive feature set consisting of technical, blockchain-based, sentiment-/interest-based, and asset-based features. The presented results show that recurrent neural networks and tree-based ensembles are especially well-suited for the examined prediction tasks. A feature importance analysis reveals that technical features constitute the most important feature type. An employed quantile-based long-short trading strategy based on the prediction generates monthly returns of up to 39% before transaction costs but does not compensate for incurred transaction costs due to the particularly short holding periods.

This section, in large parts, comprises the published articles:

- P. Jaquart, D. Dann, C. Weinhardt, *Short-Term Bitcoin Market Prediction via Machine Learning*, The Journal of Finance and Data Science 7, pages 45-66, 2021.
- P. Jaquart, D. Dann, C. Weinhardt, *Using Machine Learning to Predict Short-Term Movements of the Bitcoin Market*, 2020 International Workshop on Enterprise Applications, Markets and Services in the Finance Industry (FinanceCom 2020), pages 21-40, 2020.

2.2.1 Introduction

Bitcoin is a digital currency, introduced in 2008 by Nakamoto (2008). It is enabled by the blockchain technology and allows for peer-to-peer transactions secured by cryptography (Beck et al., 2017). In this study, we analyze the short-term predictability of the Bitcoin market. Therefore, we utilize a variety of machine learning methods and consider a comprehensive set of potential market-predictive features.

Empirical asset pricing is a major branch of financial research. Machine learning methods have been applied increasingly within this domain, due to the ability to flexibly select amongst a potentially large number of features and to learn complex, high-dimensional relationships between features and target (Gu et al., 2020). Although a considerable body of research has examined the pricing of equities and bonds, yielding in a substantial number of potentially market-predictive factors (Feng et al., 2020), less attention has been paid to the novel stream of cryptocurrency pricing. In particular, the short-term predictability of the Bitcoin market has not yet been analyzed comprehensively. Furthermore, most studies have solely considered technical features and have not analyzed the feature importance of the employed machine learning models (Jaquart et al., 2020a). Against this backdrop, we tackle this research gap by comparatively analyzing different machine learning models for predicting market movements of the most relevant cryptocurrency—Bitcoin. With a market capitalization of around 170 billion USD (September 2020), Bitcoin represents about 58% of the cryptocurrency market (Coinmarketcap, 2020). In this context, our overarching research questions are:

Research Question 1 *What is the predictive power of machine learning models predicting short-term movements of the Bitcoin market?*

Research Question 2 *What are the most relevant features for predicting short-term movements of the Bitcoin market using different machine learning models?*

We answer these research questions by comparing six well-established machine learning models trained on nine months of minutely Bitcoin-related data against each other and performing a permutation feature importance analysis. The results show that the trained models indeed significantly outperform random classification. Our study provides two main contributions. First, we contribute to the literature by systematically comparing the predictive capability of different prediction models (e.g., recurrent neural networks, gradient boosting classifiers), feature sets (e.g., technical, blockchain-based), and prediction horizons (1-60 minutes). Thereby, our study establishes a thorough benchmark for the predictive accuracy of short-term Bitcoin market prediction models. The general picture emerging from the analysis is that recurrent neural networks and gradient boosting classifiers appear well-suited for this prediction problem and technical features remain most-relevant. Also, an interesting side finding is that, for longer prediction horizons, prediction accuracy tends to increase and less recent features appear to be of particular importance. Second, despite the models' ability to create viable Bitcoin market predictions, our results do not violate the efficient market hypothesis, as the employed quantile-based long-short strategy yields returns that are not able to compensate for associated transaction costs.

2.2.2 Related Work

Financial market prediction is a prominent branch of financial research and has been studied extensively (Jaquart et al., 2020a). There is mixed evidence regarding the predictability and efficiency of financial markets (Fama, 1970; Lo, 2004). An established approach to analyze return-predictive signals is to conduct regression analysis on possible signals to explain asset returns (Fama and MacBeth, 1973; Fama and French, 2007). However, linear regressions are not able to flexibly incorporate a large number of features and impose strong assumptions on the functional form of how signals indicate market movements. In contrast, machine learning methods, which often do not impose those restrictions, have been increasingly applied for financial market prediction (Fischer and Krauss, 2018; Gu et al., 2020). Among those, neural network-based methods may be expected to be particularly well-suited, as they are already described to be the predominant method for predicting the dynamics of financial markets (Krollner et al., 2010).

2.2.2.1 Market Efficiency and Financial Market Prediction

Theory on Market Efficiency

Within efficient financial markets, prices reflect all available information and are not predictable in order to earn abnormal returns. To determine the degree of efficiency of a market, Fama (1970) defines a formal three-level framework—weak, semi-strong, and strong form market efficiency. In weak form efficient markets, prices reflect all information about past prices, whereby in semi-strong form efficient markets, prices reflect all publicly available information. In strong form efficient markets, prices additionally reflect all private information. While regulators aim to prevent investors from profiting from private information, it is generally agreed upon that major financial markets are semi-strong form efficient (Fama, 1997). On the other hand, Grossman and Stiglitz (1980) argue that market efficiency may not constitute a constant state of equilibrium over time. If information is costly and prices consistently reflect all available information, informed traders will stop to acquire information, which leads market prices to deviate from fundamental asset values. Besides, there is evidence in the scientific financial literature for a large number of potential market anomalies. Green et al. (2012), for instance, identify more than 330 different empirically-found return-predictive signals for the US stock market that have been published between 1970 and 2010. Similarly, Lo (2004) formulates the adaptive markets hypothesis, according to which markets may be temporarily inefficient. Thereby, the duration of the temporal inefficiency is influenced by the degree of competition within a market and limits to arbitrage (Shleifer and Vishny, 1995). Informed traders exploit these inefficiencies so that prices reflect all available information again. Summarizing, the question remains open, whether return-predictive signals constitute market anomalies or represent reasonably-priced risk factors. Also, some of the most prominent signals have disappeared after publication (Schwert, 2003), which indicates that part of the published return-predictive signals either have only existed in the sample period or have been erased due to traders adopting strategies for exploitation. Green et al. (2012) infer that a unified model of market efficiency or inefficiency should account for persistent empirically identified return-predictive signals.

Bitcoin Market Efficiency

Several findings in the financial literature (Glaser et al., 2014; Dyhrberg, 2016; Burniske and White, 2017; Hu et al., 2019) indicate that Bitcoin may constitute a new asset class. Therefore, findings regarding the weak form efficiency of other financial markets may not hold for the Bitcoin market. Several researchers examine the degree of market efficiency of the Bitcoin market using different time horizons. First, Urquhart (2016) investigates the time series of daily Bitcoin prices (August 2010 to July 2016). He finds that the Bitcoin market is not even weak form efficient. However, splitting the study period reveals that the Bitcoin market becomes increasingly efficient over time. Revisiting this data, Nadarajah and Chu (2017) find that a power transformation of the used Bitcoin returns satisfies the weak form efficient market hypothesis. Similarly, Bariviera (2017) examines daily Bitcoin prices (August 2011 to February 2017), using the Hurst exponent (Hurst, 1951) and shows that the Bitcoin market is not weak form efficient before 2014, but becomes weak form efficient after 2014. Vidal-Tomás and Ibañez approach the question of semi-strong form Bitcoin market efficiency from an event study perspective (Vidal-Tomás and Ibañez, 2018). With data on news related to monetary policy changes and Bitcoin (September 2011 to December 2017), they show that the Bitcoin market does not react to monetary policy changes but becomes increasingly efficient concerning Bitcoin-related events. Testing for the adaptive markets hypothesis, Khuntia and Pattanayak (2018) analyze daily Bitcoin prices (July 2010 to December 2017), finding evidence for an evolving degree of weak form market efficiency. They conclude that this finding constitutes evidence that the adaptive market hypothesis holds for the Bitcoin market.

Summarizing, there is mixed evidence among scholars regarding the efficiency of the Bitcoin market. However, most researchers find that the Bitcoin market has become more efficient over the years. An increasing degree of market efficiency seems intuitive, as the Bitcoin market has proliferated since its inception and, therefore, has become increasingly competitive.

2.2.2.2 Bitcoin Market Prediction via Machine Learning

Conducting a literature review, Jaquart et al. (2020a) analyze the literature on Bitcoin market prediction via machine learning published until April 2019. They examine the body of literature with regards to applied machine learning methods, return-predictive features, prediction horizons, and prediction types. The reviewed body of literature utilizes both classification and regression models approximately equally often, while regression models are used slightly more frequently. Due to the use of different time horizons, targets and feature variables, parameter specifications, and evaluation metrics, the comparison of prediction models across different papers often remains infeasible. On the other hand, comparisons within the same paper often avoid these shortcomings, and remain especially relevant. Based on the latter, Jaquart et al. (2020a) outline that especially recurrent neural networks yield promising results regarding Bitcoin market predictions (e.g., see Karakoyun and Cibikdiken (2018); McNally et al. (2018)). Furthermore, they group the utilized return-predictive features into four major categories—technical, blockchain-based, sentiment-/interest-based, and asset-based features. Technical features describe features that are related to historical Bitcoin market data (e.g., Bitcoin returns). Blockchain-based features denote features related to the Bitcoin blockchain (e.g., number of Bitcoin transactions). Sentiment-/interest-based features describe features that are related to sentiment and internet search volume of Bitcoin (e.g., Bitcoin Twitter sentiment). Asset-based features are features that are related to financial markets other than the Bitcoin market (e.g., gold returns, returns of the MSCI World index). More recently, Huang et al. (2019) utilize high-dimensional technical indicators to predict daily Bitcoin returns via tree-based prediction models (January 2012 to December 2017) and find that technical analysis can be useful in markets of assets with hard-to-value fundamentals. Chen et al. (2020) utilize various machine learning techniques to predict the direction of Bitcoin price movements. Using data between February 2017 to February 2019, they find that rather simple methods (e.g, logistic regressions) outperform more complex algorithms (e.g., recurrent neural networks). However, the use of a class split based on the directional price movement is likely to create an imbalanced training set, which may cause biased results (Kubat et al., 1997). Especially for financial time series, which are usually rather noisy (Gu et al., 2020), an imbalanced training set may cause classifiers to generally predict the majority class—regardless of input features. If the utilized test set has a similar target class imbalance, biased classifiers may achieve a high prediction accuracy without taking input feature values into account (Barandela et al., 2004). Using blockchain-based features and data from April 2013 to December 2019, Mudassir et al. (2020) compare feedforward neural networks, support vector machines and long-short term memory networks to predict Bitcoin price movements with prediction horizons ranging between 1 and 90 days. They find that the support vector machine model has the highest accuracy for the shorter prediction horizons, while the long-short term memory network performs best on the long term horizons. However, they also define target classes based on bi-directional price movements—potentially creating an imbalanced training set. Dutta et al. (2020) compare recurrent neural networks and feedforward networks to predict daily Bitcoin prices, using daily data from January 2010 to June 2019. They perform feature selection based on the variance inflation factor and find that recurrent neural networks tend to outperform feedforward networks on this task. However, in the chosen setting, the utilized feedforward networks do not receive similar temporal information as the recurrent neural networks and therefore results are expected to be biased towards a higher performance of the recurrent neural networks. In a first analysis, Jaquart et al. (2020b) analyze the short-term predictability of the Bitcoin market. Their results emphasize the potential of recurrent neural networks for predicting the short-term Bitcoin market. However, questions regarding the robustness of the results, theoretical foundations and descriptions of the research approach remain unanswered.

So far, only few researchers (e.g., Dutta et al. (2020); Poyser (2019)) utilize features of *all* established feature categories. Besides, the particular feature importance across different models has received little academic attention so far. The vast majority of researchers construct their models using daily prediction horizons and only few scholars (Madan et al., 2015; Smuts, 2019) benchmark different prediction horizons against each other (Jaquart et al., 2020a). Consequently, the Bitcoin market dynamics concerning prediction horizons of less than 1 hour are not fully understood yet.

2.2.3 Methodology

To tackle the previously-outlined research gap, we systematically evaluate different prediction models, features, and horizons. Thereby, we implement data gathering, preprocessing, and model building using the Python programming language and the libraries TensorFlow, scikit-learn, and XGBoost.

2.2.3.1 Data

We use data from Bloomberg, Twitter and Blockchain.com ranging from March 4, 2019 to December 10, 2019. Regarding Bloomberg, our data set includes minutely price data for Bitcoin, gold, oil and minutely levels for the total return variants of the indices MSCI World, S&P 500, and VIX. All prices and index levels are denoted in USD. Furthermore, the data set includes minutely exchange rates relative to the USD for the currencies euro (EUR/USD), Chinese yuan (CNY/USD), and Japanese yen (JPY/USD). Figure 2.1 shows the Bitcoin price development for the examined time period. From Blockchain.com, the data set includes minutely data for the number of Bitcoin transactions and growth of the mempool (i.e., storage of not-yet validated Bitcoin transactions). Last, the data set includes sentiment data of all English Twitter Tweets in the given period that include the hashtag Bitcoin ("#Bitcoin", case-insensitive).

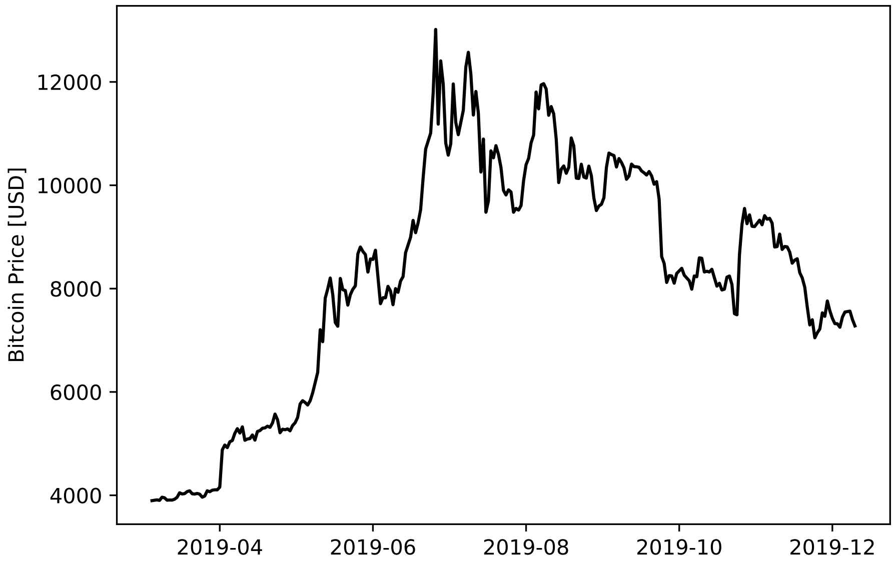


Figure 2.1.: Bitcoin price development between March 2019 and December 2019

2.2.3.2 Software and Hardware

Python 3.7 is used for all data processing and analysis, utilizing the packages pandas (McKinney, 2010a) and numpy (van der Walt et al., 2011). We rely on the NLTK library (Bird et al., 2009) for part of the Tweet preprocessing and use the Google Natural Language API (Google, 2020) for sentiment analysis. We use the keras library (Chollet, 2015) on top of the tensorflow backend (Abadi et al., 2016) to build our feedforward, LSTM, and GRU networks. We build gradient boosting classifiers with xgboost (Chen and Guestrin, 2016) and random forest as well as logistic regression models using scikit-learn (Pedregosa et al., 2011). All neural networks and the gradient boosting classifier utilize the GPU-based Nvidia CUDA parallel computing platform (GeForce GTX 1080), while the random forest and logistic regression models are trained on CPU (Intel Core i7-7700, 3.60GHz).

2.2.3.3 Features

We employ features from all four major feature categories identified by Jaquart et al. (2020a), as listed in Table 2.3. For all prediction models, we calculate minutely-updated feature values. Depending on whether the prediction model has a memory state, we further aggregate these feature values.
For the *technical* and *asset-based* features, returns are given by

$$r^a_{t,t-k} = \frac{p^a_t}{p^a_{t-k}} - 1, \tag{2.1}$$

where p^a_t is defined as the price of asset a at time t and k represents the number of periods over which the return is calculated. We obtain minutely-updated values for the selected *blockchain-based* features. *Sentiment-/interest-based* features are generated from the collected Tweets. We only keep Tweets that do not contain pictures or URLs, since the use of textual sentiment analysis is not able to capture all information contained in multimodal Tweets (Kumar and Garg, 2019). Following the suggestions of Symeonidis et al. (2018), we apply various preprocessing techniques to the collected Tweet texts. First, we remove usernames, non-English characters, and additional whitespace. Second, we replace contractions (e.g., replace "isn't" with "is not"). Last, we apply lemmatization to the Tweet texts to replace inflected word forms with respective word lemmas (e.g., replace "bought" with "buy"). Next, we make use of the Google Natural Language API (Google, 2020) to generate sentiment and estimates of strength of emotion for each Tweet. For every minutely instance, we calculate three different features: First, the number of Bitcoin Tweets published in last minute as a proxy for the overall interest in Bitcoin. Second, the sum of sentiment scores of all Tweets published in the previous minute. Third, sum of sentiment scores of all Tweets published in the previous minute, weighted by the strength of emotion per Tweet. Features 2 and 3 depend on the sentiment expressed towards Bitcoin but differ in the applied weighting scheme. While traditional prediction methods fail more often when predictors are highly correlated, machine learning models appear well-suited for these cases through the use of various variable selection methods (Gu et al., 2020).

Feature Set for Models with Memory Function

For the machine learning models with a memory function (i.e., LSTM and GRU), we create time series for all features listed in Table 2.3. To facilitate model training, all feature values are standardized based on the properties of the specific feature in the training set (Goodfellow et al., 2016).

Following feature standardization, we create time series from the 120 most recent, minutely feature values. For the employed technical and asset-based features, the time series consist of the latest 1-minute returns. However, for Bitcoin, we create an additional time series by also calculating the 1-week Bitcoin returns for each of the most recent 120 minutes according to Equation 2.1 to give the models information about the longer-term status of the Bitcoin market. Conclusively, the input for the memory models consists of 15 different time series, whereby each of the time series consists of 120 minutely time steps.

Feature Set for Models without Memory Function

The prediction models without memory function (i.e., feedforward networks, random forests, gradient boosting classifiers, and logistic regressions), require input in form of a one-dimensional vector with one observation per feature. Therefore, we create additional features by aggregating the 120-minute history of the feature classes to also give the employed no-memory models temporal information about the feature values. In line with Takeuchi and Lee (2013) and Krauss et al. (2017), we choose a more granular resolution for the most recent feature history. Specifically, we choose the following set of intervals, j, to aggregate the feature history: $j \in \{(0,1], (1,2], (2,3], (3,4], (4,5], (5,10], (10,20], (20,40], (40,60], (60,80], (80,100], (100,120]\}$, whereby these intervals describe the minutes before a prediction is made. For the aggregation process, blockchain-based features, as well as sentiment-/interest-based features are summed up across the respective intervals. For the aggregated technical and asset-based features, we calculate multi-period returns over the respective intervals (Equation 2.1). We build these intervals for all features used for the feature sequences of the memory models, except for the 1-week Bitcoin return, since this time series naturally exhibits low variation over 120 consecutive minutes. As for the memory models, we standardize all features based on the feature properties within the training set. Consequently, our feature set for the prediction models without memory function consists of $14 \times 12 + 1 = 169$ different features.

Table 2.3.: Overview of the utilized features

Technical	
Bitcoin returns	
Asset-based	
MSCI World returns	Crude Oil WTI returns
SP 500 returns	EUR/USD returns
VIX returns	CNY/USD returns
Gold returns	JPY/USD returns
Blockchain-based	
Number of Bitcoin Transactions	Mempool growth
Sentiment-/interest-based	
Twitter sentiment	Number of tweets
Twitter sentiment weighted with strength of emotion	

2.2.3.4 Targets

We formulate a binary classification problem for four different prediction horizons. For every observation, the target class c_m is formed based on the return over the next m minutes, with $m \in \{1, 5, 15, 60\}$.

We place observations, for which the return over the next m minutes is greater than or equal to the median m-minute return of all training set observations, in Class 1 and all other observations in Class 0. With regard to Equation 2.1, the target class at time t, y_t^m, is given by

$$y_t^m = \begin{cases} 1, & if\ r_{t+m,t}^{Bitcoin} \geq Median(r_{u+m,u}^{Bitcoin}) \forall u \in \{train\} \\ 0, & \text{otherwise} \end{cases}, \tag{2.2}$$

where $train$ denotes all time timestamps in the training set.

Creating classes directly from the training set ensures that the prediction models are trained on equally balanced proportions and are not subject to a bias towards one specific class. During prediction, a model returns the probability for an observation to belong to a specific class. The training set median return is 7.5984E-6 for the 1-minute prediction horizon, 2.8491E-5 for the 5 minute prediction horizon, 7.7942E-5 for the 15 minute prediction horizon and 2.7168E-4 for the 60 minute prediction horizon.

2.2.3.5 Generation of Training, Validation and Test Sets

We convert all timestamps to Coordinated Universal Time (UTC) and create a data set for each prediction problem by aggregating features and target variable. Most Bitcoin trading venues allow for continuous trading of Bitcoin but for the minutely Bloomberg Bitcoin price series, there is a gap in the time series on weekends, which we exclude from our analysis. Since the utilized asset-based features are related to assets that are mainly traded on weekdays, we consider this procedure to be reasonable. Due to the 7-day Bitcoin return feature, we require 7 days of history for every observation to calculate the complete feature set. Therefore, our final data sample spans a time range of 9 months, namely from March 11, 2019 to December 10, 2019. We use the first five-ninths of the data (approximately 5 months) to generate a training set. The subsequent one-ninth of the data (approximately 1 month) forms a validation set for hyperparameter tuning, including regularization techniques, such as early stopping (Finnoff et al., 1993; Prechelt, 2012). The remaining one-third of data (approximately 3 months) is used to test our models and obtain a representative out-of-sample prediction accuracy.

2.2.3.6 Prediction Models

With our set of models for Bitcoin market prediction, we benchmark neural networks with and without memory components, tree-based models, regression models, and ensemble models against each other. Apart from the ensemble models, all these models have already been applied to the domain of Bitcoin market predictions (Jaquart et al., 2020a). Besides the in the following described models, we make use of a logistic regression model (LR) as a benchmark. Table 2.4 gives an overview of the evaluated parameter grid for model tuning. Beyond the approach of Jaquart et al. (2020b), we train the stochastic prediction models (i.e., neural networks and random forests) on 10 different random seeds to reduce the impact of randomness on the results. We create the final class probabilities for these models by averaging the predicted class probabilities per seed over all utilized random seeds.

Table 2.4.: Overview of parameter tuning grid for all models.
* denotes selected parameters based on validation set accuracy

Model	Parameter Tuning Grid
GRU	Number of memory blocks: {64, 128, 256*, 512}
LSTM	Number of memory blocks: {64, 128, 256*, 512}
FNN	Hidden layer structure: {(512), (512-256), (512-256-128), (512-256-128-64), (512-256-128-64-32)*, (512-256-128-64-32-16)}
LR	-
GBC	Maximum tree depth: {1*, 2, 6, 10, 15, 20, None}
RF	Minimum fraction of instances per leaf: {1%, 5%, 10%, 20%*, 30%}

Neural Networks

The structure and intended behavior of artificial neural networks is inspired by the functionality of the human brain. In analogy to the structure of a brain, which consists of billions of highly interconnected neurons, artificial neural networks consist of various, highly connected nodes. Every node receives a certain amount of input from other nodes and, dependent on the received input, each note generates output, which is then passed to subsequent nodes. Hence, information is passed through the network of nodes and transformed by every node on its path. Formally, the network learns to approximate a function $f(x)$, which maps a given input x to a given output (category) y. All networks employed are trained with a batch size of 5000 and the established Adam optimizer (Kingma and Ba, 2015) to minimize the binary cross-entropy loss. To further improve the level of generalization, the networks contain individual dropout layers (Srivastava et al., 2014) and use the early-stopping method with a patience value of 10 (Finnoff et al., 1993; Prechelt, 2012).

Feedforward Neural Network

Feedforward neural networks (FNN) are a basic type of neural networks. FNNs represent a directed acyclic graph in which processed information flows exclusively in one direction, information is so to say 'fed forward'. FNNs consist of three types of layers: One input layer, capturing the input information, a variable number of hidden layers, and one output layer, determining the network's final classification. The final classification is dependent on the activation of nodes in preceding layers. The activation of each node in all layers is determined by a previously-assigned activation function. This (commonly non-linear) activation function controls the output of each node. Formally, the activation state of layer n is given by

$$a^{(n)} = g^{(n)}\left(W^{(n)^\top} a^{(n-1)} + b^{(n)}\right), \tag{2.3}$$

where $g^{(n)}$ is the activation function, $W^{(n)}$ is the weight matrix for the connections between layer $n-1$ and layer n, and $b^{(n)}$ is the bias for layer n.

Summarizing, information is passed from the input layer, processed and transformed in the different hidden layers, and finally classified in the output layer. It can be shown, that FNNs with at least one hidden layer and a differentiable and non-linear activation function are able to approximate any non-linear function, which renders them a universal approximator (Hornik, 1991). However, this does not imply that a FNN generalizes in such a way that it correctly classifies previously unseen data. Choosing an appropriate network architecture and hyperparameterization constitutes an essential step towards creating a generalized network for a given task. Figure 2.2 (bottom) describes the architecture of the applied FNNs.

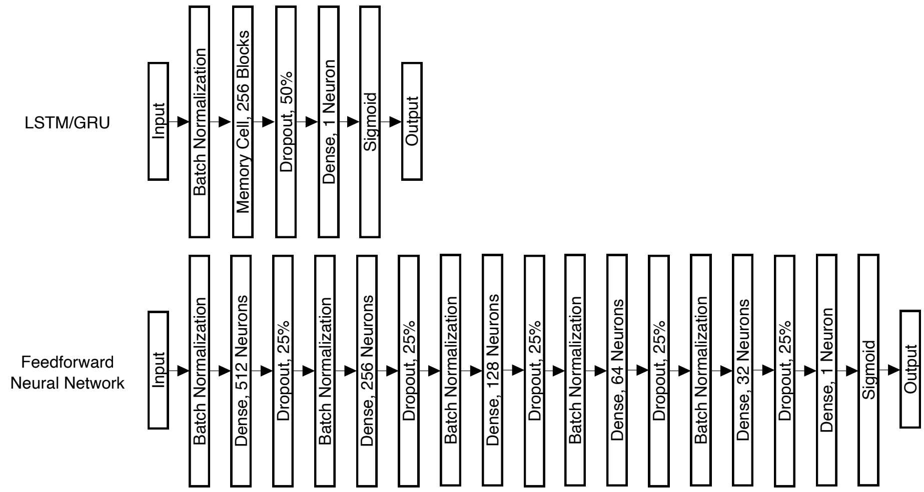


Figure 2.2.: Architecture of applied feedforward neural networks (bottom) and recurrent neural networks (top). Note: For the recurrent networks, the memory cell is either an LSTM cell or a GRU cell

LSTM and GRU

Long short-term memory (LSTM) and gated recurrent unit (GRU) networks belong to the category of gated recurrent neural networks (RNNs). RNNs drop FNN's condition of acyclic graphs. Relaxing this boundary allows for the existence of arbitrary feedback connections and an overall cyclic structure of the network. Hammer (2000) shows that RNNs with a sufficient number of hidden nodes and non-linear activation function also satisfy the requirements of a universal approximator. Figure 2.3 depicts a basic recurrent neural network in both a folded and unfolded representation. Hochreiter and Schmidhuber introduce the LSTM architecture in the late 1990's (Hochreiter and Schmidhuber, 1997) with a specific focus on long-term memorization of information in sequential data. LSTMs have been used for a variety of tasks in different domains. Among these are neural language processing and speech recognition (Graves and Schmidhuber, 2005; Graves et al., 2006, 2013; Graves and Jaitly, 2014), handwriting recognition and handwriting generation (Graves, 2013; Liwicki et al., 2007; Graves et al., 2008, 2009), music generation (Eck and Schmidhuber, 2002), analysis of financial data (Mäkinen et al., 2018), as well as more generic scenarios such as tasks that require to remember specific numbers along long sequences (Gers et al., 2002; Hochreiter and Schmidhuber, 1997). Their architecture replaces the nodes in the hidden layers with memory blocks. Each block usually consists of one memory cell and varying number of gates, which can manipulate the internal values of the cell. All blocks are connected via the cell states $c^{<t>}$, and the output of the memory blocks $a^{<t>}$. The gates enable the network to maintain the influence of inputs along longer time sequences. The original LSTM has two different gates: an input and an output gate. Each gate utilizes a sigmoid activation function. Gers (1999) extend the original LSTM with an additional forget gate, which allows the cell to reset itself. Formally, the output of the LSTM is given by:

$$\hat{y}^{<t>} = g^{<t>}(\mathrm{a}^{<t>}), \tag{2.4}$$

whereby $g^{<t>}$ is the network's activation function at t and

$$a^{<t>} = o^{<t>} \circ \tanh(c^{<t>}) \tag{2.5}$$

$$c^{<t>} = i^{<t>} \circ \widetilde{c}^{<t>} + f^{<t>} \circ c^{<t-1>} \tag{2.6}$$

$$\widetilde{c}^{<t>} = \tanh(W_c[a^{<t-1>}, x^{<t>}] + b_c) \tag{2.7}$$

$$i^{<t>} = \sigma(W_i[a^{<t-1>}, x^{<t>}] + b_i) \tag{2.8}$$

$$f^{<t>} = \sigma(W_f[a^{<t-1>}, x^{<t>}] + b_f) \tag{2.9}$$

$$o^{<t>} = \sigma(W_o[a^{<t-1>}, x^{<t>}] + b_o), \tag{2.10}$$

where $i^{<t>}$, $f^{<t>}$, $o^{<t>}$ denote the state of the input, forget, and output gate with their respective weight matrix W and bias b, $\widetilde{c}^{<t>}$ is the candidate for updating the current cell state $c^{<t>}$. Figure 2.4 depicts the composition of an LSTM memory block.

GRUs differ from the LSTMs insofar, as they use one unified gate unit to control the forget and the update gate simultaneously. Although the number of learnable parameters of GRUs is thereby smaller than that of LSTMs, their performance in various domains is comparable (Chung et al., 2014). Figure 2.2 (top) outlines the architecture of the applied RNNs.

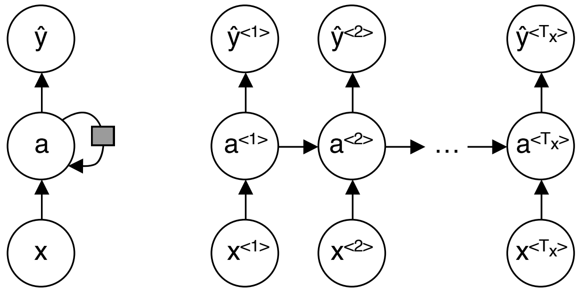


Figure 2.3.: RNN in a folded (left) and unfolded (right) state. Grey square indicates time step delay

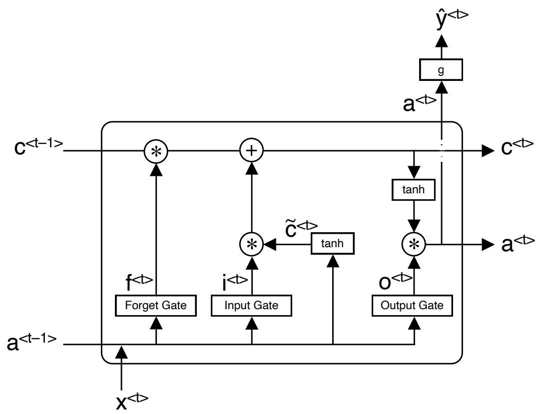


Figure 2.4.: LSTM memory block with input, forget, and output gate (based on Olah 2015)

Tree-based Models

Tree-based models use a decision tree to learn attribute-class relationships, which are fast to train and well interpretable. However, unpruned, single decision trees are prone to overfit to training data.

Random forest

Introduced by Ho (1995), random forests (RF) aim to overcome tree-based models' tendency to overfit by means of an ensemble method. Here, multiple decision trees are generated, and each of them is trained on different parts of the training data. The output of the final model for overall classification is the average output of all individual trees. The random forest applied is subject to the parameterization of 100 trees and a minimum number of instances per leaf of 20%. For all remaining parameters, we use the default values of the Python scikit-learn library (Pedregosa et al., 2011).

Gradient Boosting Classifier

Similar to random forests, gradient boosting classifiers (GBC) leverage the input of multiple decision trees (Friedman, 2001). In addition, boosting classifiers also train individual weights for all included models. In this way, the output classification of the better-adapted models is weighted more strongly in the model's final classification decision. In our analysis, we use the extreme gradient boosted (XGBoost) trees, parameterized with a binary logistic objective function to build a gradient boosting classifier, whereby individual trees have a max-depth of 1.

Ensemble Models

Similar to random forest and gradient boosting, ensemble models rely on the classification output of multiple models. However, in an ensemble, different model-types (e.g., neural networks and tree-based models) can be combined into a meta-model. The output of the meta-model constitutes the averaged predictive probability vector of all models included. In this way, method-specific misclassifications should be "overruled" by the other models in the ensemble. We apply an meta-model consisting of all individual models.

2.2.3.7 Evaluation

The prediction models are evaluated and analyzed regarding various aspects. First, we compare the models on a prediction level. Second, we analyze and compare feature importance for each model and prediction target. Third, we examine economic implications of our Bitcoin market predictions by employing a long-short portfolio strategy.

Forecast Evaluation

We compare the forecasts of our prediction models based on the predictive accuracy on the test set. Also, for our stochastic prediction models, we examine the impact of using multiple random seeds on model accuracy and stability. Furthermore, similar to Fischer and Krauss (2018), we compare the significance of the differences in model predictions based on Diebold-Mariano tests (Diebold and Mariano, 1994) with the mean absolute error loss function. Additionally, we estimate the probability that the models make predictions by chance. If the true accuracy of a binary classification model is 50%, the number of correctly classified targets follows the distribution

$$X \sim B(n = \#\text{test}, p = 0.5, q = 0.5), \tag{2.11}$$

where #test is the number of observations in the test sample (e.g., 94834 for the 1-minute horizon). Based on this binomial distribution, we calculate the probabilities that a prediction model has a true probability of 50%.

Feature Importance

The feature importance for all models is determined by the measure of permutation feature importance (Breiman, 2001). This ensures comparability between the resulting importance scores across all models. We randomly permute every feature vector with a random standard normally distributed vector and calculate the decrease in prediction accuracy, which we interpret as feature importance. A high decrease in prediction accuracy implies that the model strongly relies on the feature for its predictions. To decrease the impact of randomness on the results, we average the permutation feature importance across a set of 10 different random seeds, $s \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$. In the rare case that the random permutation increases the prediction accuracy, we set this feature's importance to zero. For our stochastic prediction models, we calculate the feature importance for every model seed (i.e., based on 10 feature importance seeds per model seed) and average the results over all model seeds. To enhance interpretability, we follow the recommendations of Gu et al. (2020) and normalize the feature importances in a way that all feature importance scores for a prediction model sum to one.

Trading Strategy

We analyze the economic implications of the Bitcoin market predictions by testing a straightforward trading strategy. To approximate an ex ante trading strategy, we calculate the 99%-quantiles of all the classes' probabilities from the training set predictions. If, for instance, the predicted probability in the test set for Class 1 is higher than the respective threshold probability, we take a long position. Vice versa, we take a short position, if a predicted class probability for Class 0 is above the respective threshold probability. Table A.1 lists these threshold probabilities per model and prediction horizon. At the end of the prediction horizon, the position is closed. We calculate the return of this strategy before and after transaction costs. Similar to Fischer et al. (2019), we assume round-trip transaction costs of 30 basis points (bps).

2.2.4 Results

2.2.4.1 Predictive Accuracy

We compare the model predictions based on the accuracy scores, which are presented in Table 2.5. We find that all tested models' predictive accuracy is above the 50% threshold. Furthermore, all models have a probability of less than 3.08E-09 for a true accuracy of 50% (see Table 2.6). Thereby, the use of multiple random seeds increases stability in the predictions of the stochastic prediction models. With regards to that, Table A.2 lists the predictive accuracies for the individual random seeds. Second, we find that the average prediction accuracy monotonically increases for longer prediction horizons. Third, we find that RNNs or GBCs constitute the best-performing methods across all prediction horizons. Specifically, the LSTM performs best on the 1-minute prediction horizon. As shown in Table A.3, Diebold-Mariano tests (Diebold and Mariano, 1994) reveal that LSTM predictions are more accurate than the predictions of all other models, apart from the GRU (α: 5%). On the 5-minute horizon, the GBC model shows the highest predictive accuracy, yielding significantly more accurate forecasts than all remaining models. The ensemble model is the most accurate method on the 15-minute prediction horizon, but does not produce significantly more accurate predictions than the GRU and GBC models. The LSTM model yields the most accurate predictions on the 60-minute horizon, which are significantly more accurate than the predictions of all other models. Summarizing, we find that the RNNs and tree-based ensembles, on average, provide more accurate predictions for the Bitcoin market compared to the other models. Both RNNs show comparable predictive accuracy. On the 1-minute and the 5-minute prediction horizon, the predictive accuracy of the GRU and the LSTM does not differ significantly, while the GRU yields more accurate predictions on the 15-minute horizon and the LSTM yields more accurate forecasts on the 60-minute horizon. Equivalently, the predictive accuracy of the tree-based models (i.e., GBC and RF) does only significantly differ on the 5-minute prediction horizon.

Table 2.5.: Predictive accuracy of the machine learning models for the different prediction horizons

	Accuracy			
Model	1-minute Predictions	5-minute Predictions	15-minute Predictions	60-minute Predictions
GRU	0.518411	0.524562	0.536490	0.556653
LSTM	0.519286	0.524931	0.531967	0.560067
FNN	0.509438	0.521988	0.520820	0.529587
LR	0.511272	0.517926	0.519595	0.538552
GBC	0.511093	0.529268	0.537282	0.557026
RF	0.511947	0.526662	0.534641	0.556356
E (All)	0.514626	0.526092	0.537863	0.557579

Table 2.6.: Probability for the prediction models to have a true accuracy of 50%, derived from the binomial distribution described in Equation 2.11

	Probabilities			
Model	1-Minute Predictions	5-Minute Predictions	15-Minute Predictions	60-Minute Predictions
GRU	4.18E-30	5.66E-52	6.19E-112	8.44E-265
LSTM	7.65E-33	1.75E-53	1.99E-86	2.28E-297
FNN	3.08E-09	4.64E-42	7.17E-38	7.02E-74
LR	1.92E-12	1.26E-28	8.99E-34	6.90E-124
GBC	4.18E-12	6.67E-73	9.47E-117	2.93E-268
RF	9.31E-14	7.30E-61	4.23E-101	4.75E-262
E (All)	1.05E-19	2.22E-58	2.40E-120	1.94E-273

2.2.4.2 Feature Importance

The predominant feature for both RNNs is the minutely Bitcoin return time series. The relative importance of this feature decreases for longer prediction horizons from about 80% (1-minute horizon) to less than 50% (60-minute horizon). It becomes apparent that, for longer time horizons, additional time series besides the minutely Bitcoin returns are increasingly relevant for both RNNs. Among those are the number of transactions per second, the number of Tweets, the weekly Bitcoin returns, and the weighted sentiment scores.

Subsequent analysis of the models without memory function provides further insights into the temporal distribution of feature importance. While the most important feature for the GBC and RF is consistently related to Bitcoin returns, for more extended prediction horizons, less recent Bitcoin returns become more important. On the 1-minute horizon, the most recent minutely return is most relevant, while on the 5-minute horizon, the Bitcoin returns from the period 10 to 5 minutes before the prediction point constitute the most important feature. Accordingly, for these models, the Bitcoin returns from 20 to 10 minutes before prediction are most important on the 15-minute horizon and the Bitcoin returns from 40 to 20 minutes before prediction constitute the most important feature on the 60-minute horizons.

Similar to the finding for the RNNs, for longer prediction horizons, the relative importance of the predominant feature drops for the GBC (60-minute horizon: 70%, 1-minute horizon: 30%) and the RF (60-minute horizon: 45%, 1-minute horizon: 30%). Besides technical features, mainly blockchain-based features (e.g., transactions per second, mempool size growth), as well as sentiment-/interest-based features (e.g., number of Tweets) remain important for the tree-based models. Compared to the GBC and RF, for the FNN and LR, feature importance is distributed along several features, which may be explained by the rather shallow parameterization of the tree-based models. We provide graphical representations of all feature importances in the Appendix A.2.

2.2.4.3 Trading Strategy

Table 2.7 lists the results of our quantile-based trading strategy before transaction costs. In comparison, a buy and hold strategy yields a return of -0.2958 during the test set period. Since the threshold class probabilities are calculated on predictions on the training set, the number of trades varies between methods and prediction horizons. Table presents the exact threshold class probabilities for the different prediction models and horizons. The results of the trading strategy yield three key insights. First, there is a rather large variance in trading results between the different prediction models. Higher predictive model accuracy does not necessarily translate into better trading results. We explain this by the fact that we do not set up our prediction problem to *optimize* trading performance, but rather to *predict* directional market movements. Additionally, based on our trading strategy, only a rather small proportion of observations is traded, which presumably increases variance. Trading returns based on the ensemble model are generally positive and near the average of the trading returns of the individual models, which indicates that combining predictions of individual prediction models may reduce the variance in trading results. Second, the average return per trade tends to increase with longer prediction horizons. Third, considering transaction costs of 30 bps per round-trip, trading performance becomes negative for all methods. These negative returns may be explained by the models' short-term prediction horizons. Based on the transaction costs, making 1000 trades would cause transaction costs of 300%.

Table 2.7.: Trading returns (*TR*) and number of trades (*#trades*) for the long-short 1%-quantile strategy before transaction costs over the 3 months of testing data

	1-minute Predictions		5-minute Predictions	
Model	TR	#trades	TR	#trades
GRU	-0.0933	557	-0.0816	586
LSTM	0.0281	584	-0.0869	701
FNN	0.0354	507	0.1089	583
LR	0.0599	664	0.3405	822
GBC	0.0637	1305	0.0097	1601
RF	0.0307	798	0.1881	574
E (All)	0.0203	582	0.1224	710

	15-minute Predictions		60-minute Predictions	
Model	TR	#trades	TR	#trades
GRU	0.0632	2164	0.2087	2806
LSTM	-0.2563	1268	1.1569	2582
FNN	0.0509	924	0.4699	661
LR	0.2059	881	-0.8732	929
GBC	0.4749	1971	0.3155	1602
RF	0.2909	385	0.5483	497
E (All)	0.2829	1421	0.4156	1244

2.2.5 Discussion

This study demonstrates that machine learning models are able to predict short-term movements of the Bitcoin market. Clearly, the forecasting accuracy of slightly over 50% indicates that the Bitcoin market predictability is somewhat limited. A limited Bitcoin market predictability is comparable to findings related to the market predictability of other financial assets, such as stocks (Fischer and Krauss, 2018; Gu et al., 2020). This may be due to multiple reasons, for instance, an immediate market reaction to the utilized features or a potentially large amount of information beyond these features that influence the Bitcoin market. Furthermore, our results are consistent with the findings that the Bitcoin market has become more efficient over the last years (Urquhart, 2016; Vidal-Tomás and Ibañez, 2018; Khuntia and Pattanayak, 2018).

Since trading results based on the market predictions are negative after transaction costs, our work does not represent a challenge to Bitcoin market efficiency. However, in this study, we analyze the predictability of the Bitcoin market movement and do not train our models to maximize trading performance. Nevertheless, our results indicate that empirical trading strategies should be implemented on the basis of models with more extended prediction horizons. This would correspond to longer holding periods, for which the relative impact of transaction costs is presumably lower. Complementary, our finding that predictive accuracy increases for longer prediction horizons paves the path for further research opportunities.

Next, we find that RNN and GBC models are particularly well-suited for predicting the short-term Bitcoin market. Yet, as the field of machine learning is evolving constantly, we may speculate about future specialized machine learning models performing even better on this task. The implemented RNN and GBC models clearly distinguish in the weighting of features, mainly relying on a set of few features. For these prediction models, technical features appear to be most influential, followed by blockchain-based and sentiment-/interest-based features. However, the exact source of predictive power for these features remains ambiguous. Possible sources of explanations may be theoretical equilibrium models. For instance, Biais et al. (2020) determine a quasi-fundamental value for Bitcoin based on, for instance, transactional costs and benefits. Some of the used features (e.g., transactions per second) may partially approximate these factors. Furthermore, Detzel et al. (2020) present an equilibrium model showing how technical indicators are able to affect the prices of assets with hard-to-value fundamental values. Subsequent studies may explore whether market anomalies, such as the momentum effect (Jegadeesh, 1990), exist within the Bitcoin market and test whether additional technical features, such as Bitcoin trading volume, exhibit predictive power. Besides, future research may examine whether behavioral financial market biases (e.g., the disposition effect (Shefrin and Statman, 1985)) are more pronounced for Bitcoin, as it does not exhibit a fundamental value in the traditional sense. Since financial markets are dynamic, future research could also evaluate whether the identified Bitcoin market mechanisms remain in place or to what extent the Bitcoin market structure has changed.

2.2.6 Conclusion

In our empirical analysis, we analyze the short-term predictability of the Bitcoin market, leveraging different machine learning models on four different prediction horizons. We find that all tested models make statistically viable predictions. The models are able to predict the binary market movement with accuracies ranging from 50.9% to 56.0% whereby predictive accuracy tends to increase for longer forecast horizons. We identify that especially recurrent neural networks, as well as tree-based ensembles, are well-suited for this prediction task. Comparing feature groups of technical, blockchain-based, sentiment-/interest-based, and asset-based features shows that, for most methods, technical features remain prevalently important. For longer prediction horizons, the relative importance appears to spread across multiple features (e.g., transactions per second, weighted sentiment), whereby less recent technical features become increasingly relevant. A quantile-based trading strategy based on the market predictions yields up to 116% return over 3 months before transaction costs. However, due to the particularly short holding periods and correspondingly frequent trading activities, these returns cannot compensate for incurring transaction costs.

2.3 Machine Learning for Cryptocurrency Market Prediction and Trading

The previous study investigated the short-term market predictability of Bitcoin via machine learning. The study highlighted the potential of recurrent neural networks and tree-based ensembles for Bitcoin market prediction and the importance of technical features. However, a potential trading strategy based on the short-term prediction could not compensate for incurred transaction costs due to the short holding-periods. In the following study, different machine learning models are set up to predict the relative daily performance of the 100 largest cryptocurrencies by market capitalization is developed and evaluated. Subsequently, a trading strategy based on these predictions is tested and evaluated. The results show that all employed machine learning models make statistically viable predictions. The employed trading strategy exhibits a higher risk-adjusted performance than the market benchmark after transaction costs.

This section, in large parts, comprises the unpublished article: P. Jaquart, S. Köpke, C. Weinhardt, *Machine Learning for Cryptocurrency Market Prediction and Trading*, Under Review, 2022.

2.3.1 Introduction

In 2008, Nakamoto Nakamoto (2008) has introduced the electronic peer-to-peer cash system Bitcoin to the world. Since then, Bitcoin has inspired numerous other cryptocurrencies with varying technical properties and use cases. Over the last decade, the cryptocurrency market has grown tremendously, whereby individual cryptocurrency prices have exhibited large volatility (Coinmarketcap, 2022). There exists mixed evidence with regards to the market efficiency of Bitcoin (Mnif and Jarboui, 2021; Noda, 2021; Vidal-Tomás, 2022) and other cryptocurrencies (Kristoufek and Vosvrda, 2019; Le Tran and Leirvik, 2020; Kakinaka and Umeno, 2022). These studies usually apply specific statistical tests that are in some form based on autoregressive approaches, whereby potential non-linear interactions, if considered, are modeled explicitly. Machine learning methods can flexibly learn the functional form between features and targets(Gu et al., 2020) and have been applied successfully to the domain of Bitcoin and cryptocurrency market prediction in the past (Huang et al., 2019; Fischer et al., 2019; Jaquart et al., 2021). Therefore, these methods may uncover and utilize high-dimensional non-linear feature interactions beyond the interactions modeled in specific market efficiency tests. In this study, we shed light on the potential of different machine learning models regarding market prediction and trading. Hence, the overarching research question of this study is:

Research Question 3 *What is the performance of machine learning models for generating statistical arbitrage in the cryptocurrency market?*

To answer this research question, we employ six machine learning classifiers to predict the relative daily performance of the 100 largest cryptocurrencies by market capitalization. Furthermore, we employ a long-short trading strategy based on the out-of-sample predictions of each model and evaluate the resulting trading outcomes. We analyze five heterogeneous study periods, with each spanning 800 days. This study has two main contributions:

First, we highlight the potential of machine learning for cryptocurrency market prediction, as all employed models make statistically viable predictions. In doing that, we find that recurrent neural networks and tree-based ensembles are particularly effective in classifying the relative daily performance of cryptocurrencies. Second, we demonstrate the potential for statistical arbitrage in the cryptocurrency market, as the employed long-short portfolio strategy outperforms the market benchmark on a risk-adjusted basis after transaction costs.

The remainder of this paper is structured as follows: Chapter 2.3.2 presents related work, Chapter 2.3.3 describes our methodological approaches, and Chapter 2.3.4 presents the results of our analyses. 2.3.5 discusses the implications of these results and Chapter 2.3.6 concludes this study.

2.3.2 Related Work

Fischer et al. (2019) examine the potential of machine learning predictions to generate statistical arbitrage in the cryptocurrency market utilizing a dataset from June to September 2018. They train a random forest classifier and a logistic regression model to predict the relative performance of the 40 largest cryptocurrencies over the next 120 minutes based on the temporal distribution of past returns over the last day. As an out-of-sample long-short trading strategy based on these model predictions yields daily returns of 7.1 bps per day, their findings indicate an impairment of cryptocurrency market efficiency. Fil and Kristoufek (2020) apply pairs trading to the cryptocurrency market, assuming a long-term stable state between different cryptocurrency pairs. They use data from January 2018 to September 2019 and compare 5-minute, hourly, and daily trading frequencies. Fil and Kristoufek (2020) find that pairs-trading may perform well for shorter frequencies in the cryptocurrency market, whereby these results are highly dependent on the selected market parameters (e.g., the magnitude of transaction cost).

Betancourt and Chen (2021) examine the potential of deep reinforcement learning for cryptocurrency trading based on a dataset ranging from August 2017 until November 2020. In the presented approach, agents repeatedly analyze the 20-day history of price, volume, and market capitalization of a specific cryptocurrency to make one-day trading decisions. Betancourt and Chen (2021) find that their approach is promising for cryptocurrency trading. McNally et al. (2018) compare an Elman recurrent neural network, a long short-term neural network, and an autoregressive integrated moving average approach to predict binary daily Bitcoin market movements. They utilize data from August 2013 to July 2016 and find that the long short-term neural network exhibits the highest predictive performance with a model accuracy of 52.78%. Dutta et al. (2020) compare different neural network approaches to predict daily Bitcoin prices based on a feature set consisting of various technical, blockchain-based, asset-based, and interest-based features. They find that a gated recurrent unit implementation with recurrent dropout yields the highest performance on their dataset, which ranges from January 2010 until June 2019.

Chen et al. (2020) employ and compare various linear statistical methods and machine learning approaches for 5-minute and daily Bitcoin market prediction on data from February 2017 to February 2019. They document a higher predictive performance of the employed statistical methods for the daily prediction horizon, while the machine learning methods exhibit a higher performance on the 5-minute horizon. Alessandretti et al. (2018) design different models based on gradient boosting classifiers and long short-term neural network approaches to predict the daily returns of 1681 cryptocurrencies. They utilize data from November 2015 until April 2018 and show that portfolio strategies based on these predictions outperform a baseline approach. Lahmiri and Bekiros (2019) compare a long short-term memory neural network and a generalized regression neural network approach to predict the prices of Bitcoin, Digital Cash, and Ripple. They utilize data sets with different temporal lower bounds that end in October 2018 and find that the employed long short-term memory neural network yields better forecasts than the generalized regression neural network.

2.3.3 Methodology

The study comprises four main steps and builds on the methodological approaches of Fischer et al. (2019) and Fischer and Krauss (2018). In the first step, we obtain the relevant data from various sources. We then generate features and targets from the raw price data, which we use to model coin returns. The next step is to split the complete data set into overlapping study periods with varying market constituents and non-overlapping test folds for backtesting. The final step is to train and tune the models used, individually for all study periods, and simulate trading based on the model predictions.

2.3.3.1 Data

For this study, we use daily closing price and market capitalization data denoted in U.S. Dollars (USD) over the period from February 8, 2018, to May 15, 2022, obtained through CoinGecko's (CG) API (CoinGecko, 2022).

Coin Market Capitalization Data

To avoid survivorship bias, the constituents of the investment universe are determined as the top 100 crypto assets by market capitalization on the first trading day of the training set. This ensures that no look-ahead bias is introduced through to the construction of the coin universe while providing a sufficient number of training instances for each coin. Stablecoins pegged to the USD or any other fiat currency are excluded from the list of eligible candidates, as their prices quoted in USD are static by design or entirely dependent on currency exchange rates. We exclude a list of ten other coins due to data issues such as missing data and erroneous values in the data sources (full exclusion list in Appendix B.1).

Daily market capitalization data for the largest 1,750 crypto assets (as of June 8, 2022) is obtained through the CG API, which provides market capitalization data denoted in USD calculated as the product of known available supply and the asset's price. For each study period, we rank all coins based on their market capitalization on the first trading day of the training set and the top 100 cryptocurrencies are used as the asset universe for creating the crypto asset portfolios.

Coin Price Data

We use aggregated market price data from the CoinGecko for the return calculations. CG provides aggregate prices based on the pairings (cryptocurrency vs. fiat currency or cryptocurrency vs. cryptocurrency) available on all monitored exchanges by applying a global volume-based weighting. Despite these prices being non-traded aggregate prices, Vidal-Tomás (2022) shows that such artificially compounded prices are a fair representation of the overall cryptocurrency market. The author finds that aggregating different exchange platforms to compute a singular price does not affect market efficiency for liquid cryptocurrencies. Since cryptocurrency exchanges are open around the clock, we create artificial closing prices from the market price at midnight (UTC). The CG API provides daily price data with a 00:00:00 UTC timestamp associated with the following day. Therefore, the retrieved time series of daily quotes are shifted by one day to calculate the previous day's returns. Thus, using p_t^c to denote the aggregate market price for coin c at the end of day t (measured at 00:00:00 UTC on day $t+1$), the m-period returns $r_t^{m,c}$ are calculated as follows:

$$r_t^{m,c} = \frac{p_t^c}{p_{t-m}^c} - 1, \tag{2.12}$$

where

$r_t^{m,c}$	Return for coin c on day t over the last m days
p_t^c	Aggregate closing price for coin c on day t.

For $m = 1$, we thus obtain the asset's daily returns, while for $m > 1$, $r_t^{m,c}$ represents the cumulative returns over the last m days.

Risk-free Rate of Return

The U.S. treasury's three-month Treasury Bill (T-bill) secondary market rate (Board of Governors of the Federal Reserve System, 2022) is used as the risk-free rate to calculate excess returns. The T-bill is a short-term debt obligation backed by the Treasury Department of the United States government with a maturity of three months. The annual interest rate is converted to daily returns by simple deannualization to calculate daily excess returns for calculating risk-adjusted return metrics such as the Sharpe ratio and the Sortino ratio. For most of the period under consideration, the risk-free rate, as measured by the T-bill rate cited above, is close to zero, ranging from 2.4×10^{-7} to 2.8×10^{-5} per day, with a mean value of 3.9×10^{-6}.

2.3.3.2 Software and Hardware

We use Python 3.9 for data acquisition, processing, and analysis throughout the study. The numpy (Harris et al., 2020) and pandas (McKinney, 2010b) software packages are used for data processing and feature creation. Deep learning models are built using Keras (Chollet, 2015) with the *TensorFlow* backend (Martín Abadi et al., 2015) and all other machine learning models are built and trained using the *scikit-learn* (Pedregosa et al., 2011) library. All models are trained on a CPU (Intel Core i5-8400, 2.8 GHz).

2.3.3.3 Data Split

The full study time frame is divided into five overlapping study periods (SPs), each comprising 800 trading days, i.e., prediction targets. The range of dates used for each SP includes the 90 days prior to the first trading day, as each prediction uses data from a 3-month look-back period as model inputs. Each study period consists of training (500 days), validation (150 days; for hyperparameter-tuning), and out-of-sample test sets (150 days) in chronological order, as illustrated in Figure 2.5. Table 2.8 shows the exact split of each study period into the three respective data folds.

The training and validation splits together constitute the formation period, during which the models are trained and the best hyperparameters for each model are selected based on the validation performance. The test split of each study period is used for out-of-sample testing and simulated trading. Study periods are shifted by the length of the test period to allow for five non-overlapping test sets for successive evaluation. Using multiple study periods allows for periodic retraining of models and thus captures the concept drift that occurs due to changing market phases.

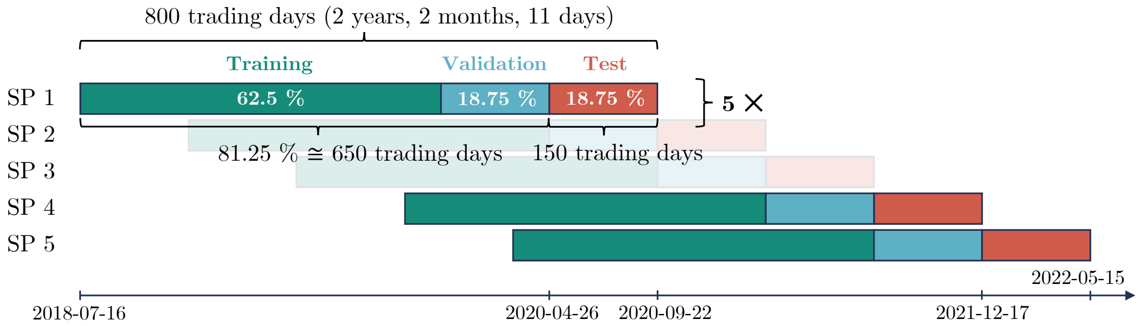


Figure 2.5.: Study period composition and train-validation-test split

Table 2.8.: Study periods and the respective date ranges for the training, validation, and test sets

SP No	Training Set	Validation Set	Test Set
1	2018-07-16 - 2019-11-27	2019-11-28 - 2020-04-25	2020-04-26 - 2020-09-22
2	2018-12-13 - 2020-04-25	2020-04-26 - 2020-09-22	2020-09-23 - 2021-02-19
3	2019-05-12 - 2020-09-22	2020-09-23 - 2021-02-19	2021-02-20 - 2021-07-19
4	2019-10-09 - 2021-02-19	2021-02-20 - 2021-07-19	2021-07-20 - 2021-12-16
5	2020-03-07 - 2021-07-19	2021-07-20 - 2021-12-16	2021-12-17 - 2022-05-15

2.3.3.4 Features

All models are trained on the binary classification problem of predicting whether a single coin will outperform the cross-sectional median of returns on the subsequent day, based on solely on price information of the previous 90 days. Thus, we derive the features for all models from the individual coin returns three months prior to trading.

Feature generation is performed separately for the two main types of classifiers used in this study, namely models with a memory function and models without memory. For the three deep-learning models with internal memory, the LSTM, the GRU, and the TCN, standardized daily return sequences of length 90 are created. The daily returns are standardized by subtracting the mean and dividing by the standard deviation of the respective training set. The tree-based classifiers and the

LR use lagged returns as model inputs due to their lack of memory. Tuples of input sequences and target labels are created successively by generating overlapping sequences of length 90 that are iteratively shifted forward by one day. The procedure for creating input sequences with corresponding target labels for the deep learning methods is exemplified in Figure 2.6.

Since the memory-free models (i.e., GBC, RF, and LR) are not inherently capable of using temporal input data, we create time-lagged features by aggregating returns over different intervals of increasing length. Based on Takeuchi and Lee (2013), and Krauss et al. (2017), we use multi-period returns with lags $m \in \{\{1, 2, ..., 20\} \cup \{30, 40, ..., 90\}\}$, increasing the resolution to 10 days after using daily increments for the first 20 days, resulting in a total of 27 features per sample. The multi-period returns are calculated using Equation 2.12. The creation of the return features and corresponding target labels for the tree-based methods and LR is illustrated in Figure 2.7. Per coin and study period (for all coins, respectively), both feature generation methods yield 500 (60,000) training samples, 150 (15,000) validation samples, and 150 (15,000) test samples.

2.3.3.5 Targets

The binary prediction problem is to forecast whether an individual coin will outperform the cross-sectional median on the day after portfolio formation. Hence, for each trading day, the daily returns for all coins are sorted in descending order, and the class label 1 is assigned to all coins above or equal to the cross-sectional median of returns and 0 otherwise. The target class for coin c at time t, y_t^c is thus given by

$$y_t^c = \begin{cases} 1, & \text{if } r_t^c \geq Median(r_t^c),\ \forall c \in C(t) \\ 0, & \text{otherwise,} \end{cases} \tag{2.13}$$

where

r_t^c	Daily return for coin c on day t
$C(t)$	Set of coins that are part of the coin universe on day t.

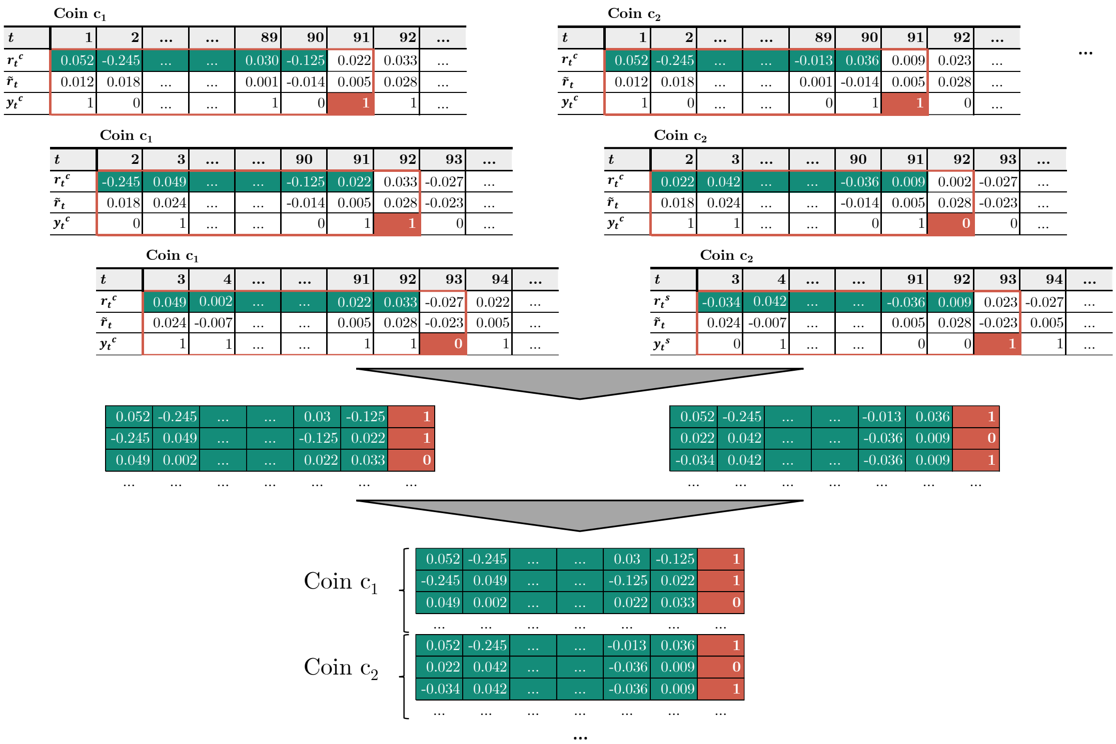


Figure 2.6.: Creation of feature sequences and corresponding target labels for models with memory

2.3.3.6 Models

We test and compare different types of predictive models, including recurrent neural networks, convolutional neural networks, tree-based ensemble methods, and the logistic regression (LR) model as a simple and efficiently computed benchmark. Due to the stochastic nature of their training process, we train all models except the logistic regression with ten different random seeds and create ensemble models by averaging the cross-sectional ranks resulting from the predicted probabilities. The hyperparameters are optimized separately for each study period using the classification accuracy of the respective validation fold for model selection. For the logistic regression, we use the default parameters without optimization. Table 2.9 shows the hyperparameter grid used for optimizing each model and the selected configuration per study period.

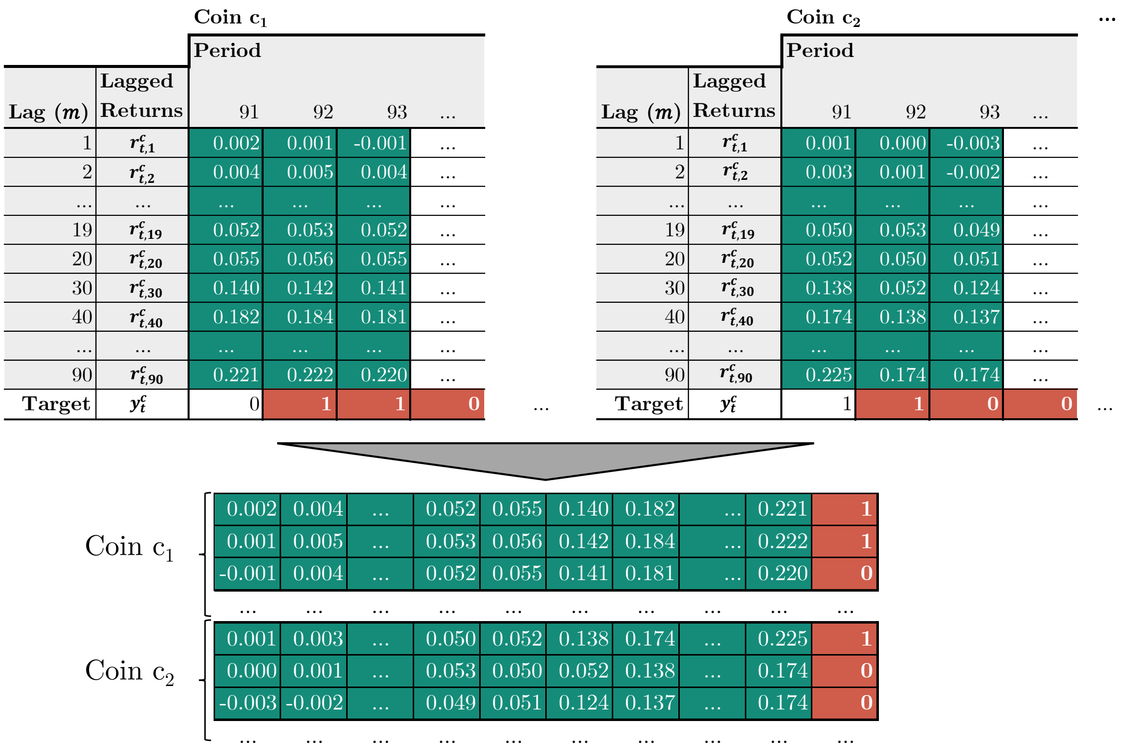


Figure 2.7.: Creation of tree-based and logistic regression feature sets and corresponding target labels

Table 2.9.: Parameter tuning space and selected configuration per study period based on validation accuracy

Model	**Parameter Grid**	**SP 1**	**SP 2**	**SP 3**	**SP 4**	**SP 5**
GRU	Number of memory cells: {5, 10, 15, 20}	10	10	15	20	5
LSTM	Number of memory cells: {5, 10, 15, 20}	15	20	20	10	10
TCN	Number of filters: {2, 4, 6}	6	6	6	4	4
GBC	Max. tree depth: {1, 2, 3, 5, 10}	2	1	2	2	2
RF	Max. tree depth: {1, 2, 3, 5, 10, 20, None} x max. number of features per split: {1, 3, 5, 7, 10, None}	(5, 3)	(5, 3)	(5, 1)	(3, 5)	(2, 3)
LR	-	-	-	-	-	-

Deep Neural Networks

We compare three different deep neural network architectures from two families: recurrent neural networks (RNN) and convolutional neural networks (CNN). RNNs maintain an internal state, or memory, and can therefore process input sequences of variable length. Temporal convolutional networks (TCN) use a number of variations of standard CNN architectures that allow them to also retain a long-term memory.

All deep neural network models are trained for a maximum of 25 epochs with a batch size of 1024 and a learning rate of 0.002 using the Adam (Kingma and Ba, 2015) optimizer for optimizing the binary cross-entropy loss. Early stopping with a patience of 4 epochs at a threshold of 1×10^{-4} with respect to the validation loss is used to mitigate overfitting and thus improve generalization.

Long Short-Term Memory

Long short-term memory (LSTM) neural networks are a prominent member of the class of recurrent neural networks commonly used for time series forecasting in various domains. They were first introduced by Hochreiter and Schmidhuber (1997) for learning long-term dependencies in long series of data and remedy the exploding vanishing and exploding gradient problems that vanilla RNNs suffer from. This is achieved by making use of several gating mechanisms (input gate, output gate, and forget gate). We use a simple architecture with a single LSTM layer containing a varying number of memory blocks of $nblocks \in \{5, 10, 15, 20\}$ and a dropout ratio of 0.1, followed by two dense layers. The first dense layer contains five neurons and uses the rectified linear unit (ReLU) as the activation function, while the final layer consists of a single neuron with a sigmoid activation function to produce a probability output in the range between zero and one.

Gated Recurrent Unit

The gated recurrent unit (GRU) is an RNN closely related to the LSTM architecture, but has fewer parameters as it lacks a dedicated output gate. This is achieved by using one gate (the update gate) that controls both the forgetting and the output simultaneously. We use the same architecture and learning parametrization as for LSTM and only replace the single LSTM layer with a single GRU layer.

Temporal Convolutional Network

Temporal convolutional networks are a fairly recent type of convolutional neural network with design characteristics that enable them to work well with long time series. The TCN architecture is characterized by (1) the use of one-dimensional causal convolutions (which ensure that the ordering of temporal data is preserved),

(2) the ability to transform a sequence of arbitrary length into an output sequence of the same length, and (3) the use of dilated convolutions to enable effective long-term memory (Bai et al., 2018).

We use a TCN architecture consisting of a single TCN layer with a kernel size of three and exponentially increasing dilation rates ($d = [1, 2, 4, 8, 16]$) for the dilated causal convolutions. The number of filters is used as a tuning parameter ($k_f \in \{2, 4, 6\}$). After the TCN layer, we use dropout of 0.25 followed by two dense layers. The first dense layers consists of five neurons with a ReLU activation and the output layer consists of a single unit and a sigmoid activation function.

Memory-Free Prediction Models

We use two different memory-free, tree-based ensemble methods for benchmarking the deep learning models used: *random forest classifiers* (RF) and *gradient boosting classifiers* (GBC). Both ensemble models use a collection of decision trees as base learners to mitigate overfitting, a common problem when using single decision trees. Logistic regression is used as a simple benchmark.

Random Forest

Random forests are based on randomized decision trees and work by creating and combining a diverse set of weak learners, i.e., decision trees, whose individual predictions are combined to an ensemble prediction by averaging their outputs. The used RF implementation of the *scikit-learn* library combines the individual classifiers by averaging their probability predictions. We implement the random forest with the default setting of 100 base learners and tune both the maximum tree depth and the maximum number of features used to split the decision trees.

Gradient Boosting Classifier

Gradient boosting classifiers rely on the sequential construction of shallow decision-trees based on the errors made in the previous iteration to reduce the bias of the combined estimator. They are a generalization of the AdaBoost algorithm by Freund and Schapire (1997) proposed in the seminal work of Friedman (2001). We use the scikit-learn (Pedregosa et al., 2011) implementation with the default value of 100 estimators and use the maximum depth of each trees as a tuning parameter.

Logistic Regression

The logistic regression model serves as a simple and efficiently trainable benchmark model against which the more complex models are compared. LR is the equivalent of simple linear regression for binary response variables, as is the case with classification problems, and models the probability of a binary event occurring based on a linear combination of a set of predictors. Even though there is no closed-form solution as in standard linear regression, the global optimum can be found efficiently by numerical methods due to the convexity of the loss function. Note that the LR model is the only model used for inference without creating an ensemble of individual models trained with different seeds, since it has a unique solution and is not subject to a stochastic optimization process.

Alternatively, the LR model can be represented by a simple single-layer neural network with one neuron and a sigmoid activation function when using binary cross-entropy as the loss function. We use the *scikit-learn* implementation of LR with Newton-CG as the solver for the optimization problem and a maximum number of iterations of 1000. For all other hyperparameters, we use the default values and do not perform any tuning.

2.3.3.7 Prediction and Portfolio Formation

At the end of each day t in the trading period and for each model type, we predict the probability of outperforming the market on day $t+1$, $\hat{\mathcal{P}}^{c}_{t+1|t}$, independently for each coin $c \in C(t+1)$, using only information available on day t. For all model types except the LR, we make the probability predictions for all 10 individual models (ensemble constituents) trained on different random seeds. To obtain the final class predictions per model type, coin, and day, these probability predictions are sorted in descending order for each model and day and ranks are assigned accordingly. The ten individual ranks of the constituent models per model type are then averaged to obtain a final ranking for all coins per model type and day.

We opt to use the individual ensemble model constituents' rank predictions instead of the predicted probabilities for obtaining the ensemble predictions. In doing so, we account for the fact that the probability predictions are only meaningful relative to the predicted probabilities of all other assets for the same day. By averaging the ranks instead of probabilities, we retain the information about predicted relative performance that is included in the prediction of each constituent model. Based on the averaged ranks for each day, the bottom half is assigned the label 0 and the top half is assigned the label 1. In other words, we use the averaged predicted cross-sectional ranks of coins to obtain a balanced set of predictions for the balanced set of true class labels.

The averaged prediction ranks per model type and trading day are used for portfolio formation by assigning the top k coins to the long leg and the bottom k coins to the short leg for the following trading day. As a result, the created long-short portfolios for each model contain $2k$ different coins from the available asset universe of 100 coins representing the market. The final class predictions and rankings are then used to calculate the predictive accuracy of the models for different portfolio sizes by restricting the calculation to the cryptoassets selected for each portfolio.

2.3.3.8 Backtesting

We base our long-short trading strategy on the portfolio selection rule described above, using the average ranks of the model predictions to form a balanced and dollar-neutral long-short portfolio of size $2k$ for different values of k. The portfolio positions are opened at the end of day t at the market closing prices and closed at the end of day $t + 1$ after a holding period of one full day. Each time the portfolio positions are opened and closed, they incur the assumed transaction costs of 15 bps of the transaction volume. After closing of each position at the end of the holding period, the resulting cash position is used to fund the next day's trades. For each model, we thus hold a long-short portfolio containing $2k$ coins at any given time, changing its composition once per day while incurring transaction costs for every trade. The financial performance of this daily trading strategy is calculated using the net asset value of investments, which includes all coin returns and incurred trading costs.

Transaction Costs

When in comes to arbitrage on cryptocurrency markets, the transaction costs for trading on various cryptocurrency exchanges must be taken into account. Transactions costs consist of commission fees, market impact and, in the case of short-selling, short-selling costs. For the calculation of daily returns, half-turn transaction costs of 15 bps are assumed, following similar work by Fischer et al. (2019). Additional short-selling costs are not taken into account as short-selling of cryptoassets is not possible for all considered coins as of the time of writing and an estimation of short-selling related costs not feasible.

2.3.4 Results

In this section, we present the results of the compared prediction methods in terms of predictive accuracy and financial performance achieved with the derived long-short trading strategy. An equally-weighted buy-and-hold market portfolio (MKT) is used as the benchmark for evaluating the portfolio performance. The choice to use an equally-weighted market portfolio as the benchmark is motivated by the fact that the trading strategy evaluated is based on daily equally-weighted long-short portfolios.

First, we analyze the overall results across all five study periods for different portfolio sizes ($k \in \{1, 2, 5, 10, 20, 50\}$), where each portfolio consists of $2k$ stocks. For predictive accuracy, we then take a more granular look at the results for each study period to see how the performance varies over time. We then focus our analysis on the $k = 5$ portfolio, which contains a sufficient number of assets to diversify risk, but not so many as to negate the effect of selecting coins with a relatively high degree certainty.

2.3.4.1 Model Accuracy

The employed models' ability to accurately predict whether a coin will outperform the cross-sectional median is the basis for forming a profitable long-short portfolio. Thus, we first evaluate and compare the models in terms of their predictive accuracy. Following the approach of Fischer and Krauss (2018), we evaluate the models' accuracy by calculating the probability of a random classifier achieving the same accuracy or higher. To do this, we model the number of correctly classified coins in the chosen portfolio of size k, X_k, as a binomial distribution under the assumption of a true classification accuracy of 50%, $X_k \sim B(n = 15,000 \cdot \frac{2k}{100},\ p = 0.5)$, and calculate for each model the p-values for achieving the number of correctly classified coins or better by chance alone.

Full Time Period

Using the method described above, the prediction accuracy for the entire time period examined (i.e., all test sets of the five study periods combined) is significantly higher than 50% for all surveyed prediction methods and all portfolio sizes (see Fig. 2.8). For $k = 50$, (i.e., taking all 100 daily predictions into account), the RF performs best with an accuracy of 54.2%, closely followed by the LSTM with 54.1%. The probabilities of a random classifier scoring at least as good is 3.419×10^{-119} and 3.203×10^{-110}, respectively, indicating a clear advantage of the two recurrent neural networks over a random classifier. The complete summary of p-values is given in B.2. Following behind at a comparable accuracy level are the TCN and GBC (both 53.6%), and the GRU (53.5%). All machine learning models thus considerably outperform the LR benchmark (52.9%).

Note that the prediction accuracy for $k = 50$ corresponds to the models' actual (i.e., unrestricted) test set performances. For all $k < 50$, the accuracy scores are those achieved on the corresponding subset of predictions. For different portfolio sizes, the accuracy is monotonically increasing for smaller values of k for all models (see Fig. 2.8).

For the restricted out-of-sample performances the pattern of model rankings remains similar, with all models improving in accuracy as k decreases. The LR model performs worst in terms of prediction accuracy regardless of the portfolio size, while the RF maintains its lead over all other models for $k = 20$ (57.2%) and $k = 10$ (58.3%). For $k = 5$, the GRU outscores the RF with a result of 59.5% compared to 59.3% of the RF and also outperforms the LSTM model (59.1%). For the $k = 2$ portfolio, the GRU (61.0%) still performs best, closely followed by the GBC (60.8%), and remains slightly better than the RF and LSTM models (both 60.6%). The TCN is more than one percentage point behind the GRU with only 59.5%, while the LR models is even further behind with 59.2%. When only one coin is selected for the long side and one for the short side of the portfolio (i.e., $k = 1$), the two recurrent neural networks have an advantage over all other models. The LSTM narrowly outperforms the GRU with an accuracy of 61.73% compared to 61.67% of the GRU. The tree-based ensemble models (GBC and RF) follow with 61.0% and the TCN achieves 60.8%, performing only slightly better than the LR (60.6%).

When considering the individual class predictions, a divergence between the performance of the long and short legs is evident for all $k < 50$. The accuracy within the predicted classes corresponds to the respective class precision (positive and negative predictive value). Note that for $k = 50$, the long and short leg performances are equal by construction. A slight deviation in the calculated accuracies is due to numerical limitations. For all models, the respective precision for the short positions is considerably higher than for the long positions (see Table 2.11). At the same time, the monotonicity of higher class precision for a lower value of k presists with the notable exception of the TCN model, which achieves a slightly lower long-only precision for $k = 10$ at 55.59% than for $k = 20$ at 55.65%. On the short side, the monotonicity holds up to the $k = 2$ and $k = 1$ portfolios, where precision decreases for the GRU, RF, and LR models.

Table 2.10.: Prediction accuracy for long-short portfolio and different portfolio sizes

Portfolio Size	**k = 1**	**k = 2**	**k = 5**	**k = 10**	**k = 20**	**k = 50**
GRU	0.617	0.610	0.595	0.579	0.561	0.535
LSTM	0.617	0.606	0.591	0.582	0.566	0.541
TCN	0.608	0.595	0.585	0.574	0.563	0.536
GBC	0.610	0.608	0.585	0.578	0.565	0.536
RF	0.610	0.606	0.593	0.583	0.572	0.542
LR	0.606	0.592	0.575	0.566	0.551	0.529

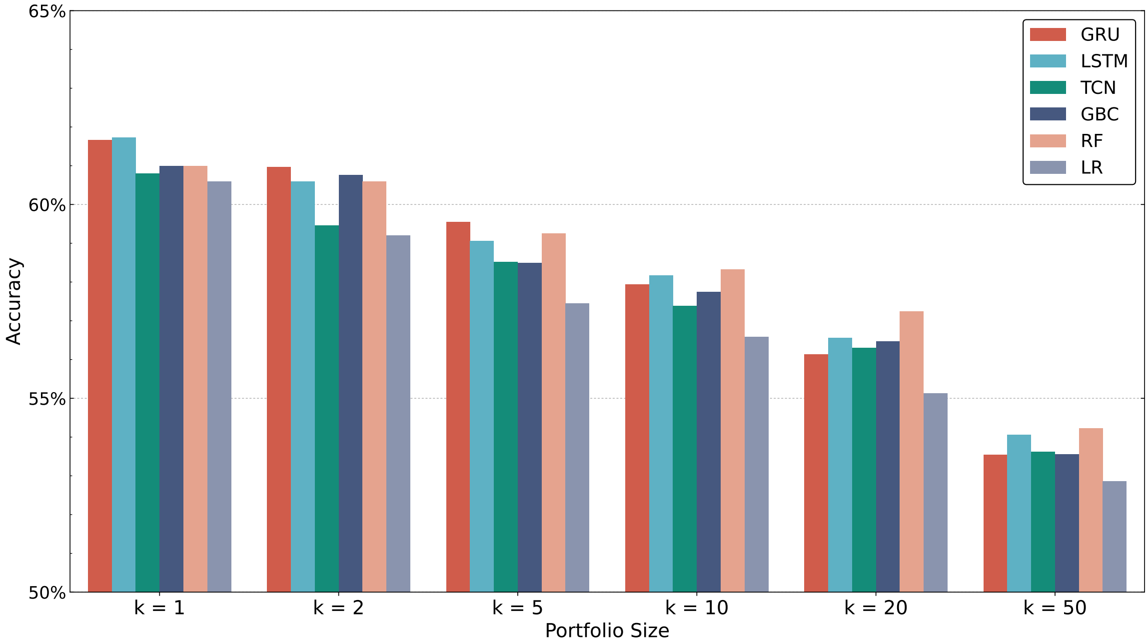


Figure 2.8.: Prediction accuracy for long-short portfolio and different portfolio sizes

Table 2.11.: Prediction accuracy for long and short legs of the long-short portfolio for different portfolio sizes

Portfolio Size	**k = 1**		**k = 2**		**k = 5**		**k = 10**		**k = 20**		**k = 50**	
Portfolio Type	**Long**	**Short**	**Long**	**Short**	**Long**	**Short**	**Long**	**Short**	**Long**	**Short**	**Long**	**Short**
GRU	0.601	0.632	0.583	0.636	0.569	0.622	0.555	0.604	0.550	0.573	0.535	0.535
LSTM	0.604	0.631	0.585	0.627	0.569	0.612	0.566	0.598	0.560	0.571	0.541	0.541
TCN	0.579	0.637	0.572	0.617	0.561	0.609	0.556	0.592	0.557	0.570	0.536	0.536
GBC	0.585	0.635	0.583	0.633	0.567	0.603	0.564	0.591	0.556	0.573	0.535	0.536
RF	0.597	0.623	0.578	0.634	0.570	0.615	0.569	0.598	0.564	0.581	0.542	0.543
LR	0.591	0.621	0.561	0.623	0.549	0.601	0.544	0.588	0.540	0.563	0.529	0.529

Individual Study Periods

In this section, we analyze the models' accuracy for the $k = 5$ portfolio over time. Figure 2.9 depicts how the achieved accuracies vary across study periods. The ranking between the models is not constant and does not follow a clear pattern, but all accuracies for each period are significantly larger than 50% based on the binomial test (see Appendix B.3 for an overview of all p-values).

The GRU performs best in the first study period with an accuracy of 60.8% (primarily due to its high short-leg precision of 62.8%, which exceeds the TCN's precision of 61.5%), followed by the GBC with 60.5%. Compared to the overall accuracies across all periods, the LR model's individual period performance exhibits the largest variability. The TCN, which performs worst among the three deep learning models (GRU, LSTM, and TCN) overall for $k = 5$, achieves the highest long-short accuracy in the second study period with 59.9%, compared to the 58.9% of the LSTM and the GRU's 58.3%. This is mainly due to its superior long-leg precision of 58.3% in that period (see Tab. 2.12).

In the third study period, the GRU again achieves the highest long-short accuracy again with 60.9%, as its short-only precision of 62.1% is considerably higher than that of all other models in this period (see 2.12). Interestingly, the LR model (60.1%) has the second best performance of all models, with a slight edge over the RF (60.0%). The TCN (58.7%) performs worse than the LSTM (59.2%), but manages to outperform the GBC, which performs worst with 57.6%. The fourth study period sees the TCN's long-short accuracy drop to only 55.1%, as its long-leg performance declines to a mere 48.5%. At the same time, it exhibits the highest short-leg accuracy of all models and catches up again with the other models in study period 5, where all models score within a relatively narrow range between 58.0% (LR) and 59.9% (both LSTM and RF). while showing the highest short-leg accuracy among all models, before it catches up again in study period 5 by achieving 59%, where all models score within a relatively narrow margin between 58.0% (LR) and 59.9% (both LSTM and RF).

In summary, the LR's relative disadvantage with regard to overall accuracy is due to its sub-par performance in period 1, while the overall best GRU model performs competitively across all study periods and outperforms all other models in periods 1 and 3. The same is true for the second and third best overall models, the RF and the LSTM, which also score competitively in all study periods and narrowly outperform all other models in study period 5.

2.3.4.2 Trading Results

This section presents the financial performance results of the employed long-short trading strategy as well as the long side of the strategy. The financial performance is analyzed along the dimensions of return, risk, and risk-return metrics for the $k = 5$ portfolio, a diversified portfolio with five long and five short positions. We restrict

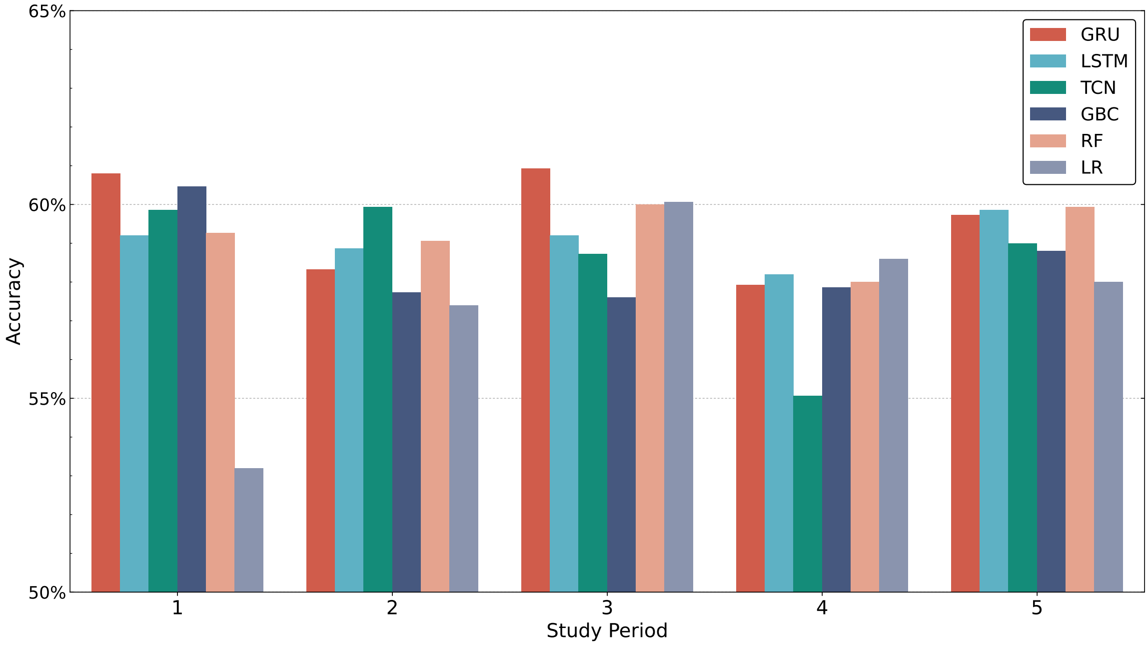


Figure 2.9.: Accuracy of $k = 5$ long-short portfolio prediction by study period

Table 2.12.: Prediction accuracy for long and short legs of $k = 5$ portfolio for individual study periods

Study Period	**1**		**2**		**3**		**4**		**5**	
Portfolio Type	**Long**	**Short**	**Long**	**Short**	**Long**	**Short**	**Long**	**Short**	**Long**	**Short**
GRU	0.588	0.628	0.557	0.609	0.597	0.621	0.551	0.608	0.551	0.644
LSTM	0.575	0.609	0.553	0.624	0.591	0.593	0.567	0.597	0.561	0.636
TCN	0.583	0.615	0.583	0.616	0.581	0.593	0.485	0.616	0.575	0.605
GBC	0.605	0.604	0.576	0.579	0.545	0.607	0.553	0.604	0.555	0.621
RF	0.576	0.609	0.577	0.604	0.584	0.616	0.564	0.596	0.551	0.648
LR	0.483	0.581	0.565	0.583	0.597	0.604	0.575	0.597	0.523	0.637

the financial analysis to this portfolio, which includes 10% of the market constituents, similar to Fischer et al. (2019), who include 15% of the respective market in their portfolio.

The equally-weighted market portfolio (consisting of the 100 eligible coins that constitute the asset universe for each study period) serves as the natural benchmark for the tested trading strategies. Figure 2.10 shows the performance of the market index over all study periods as well as the performance of Bitcoin over the same time period and illustrates their high correlation.

Daily Returns

For portfolio size $k = 5$, the two recurrent neural network architectures (LSTM and GRU) achieve the highest daily returns (with 1.44% and 1.35%, respectively). Table 2.13 shows the daily return metrics for all models. Notably, the simplest model, the logistic regression, outperforms the tree-based models (RF and GBC) with a daily return of 1.18% (compared to 1.10% and 0.99%, respectively), and performs only slightly worse than the LSTM and GRU models. The TCN performs worst among all models, earning daily returns of only 0.31%, making it the only model to underperform the general market (0.33%).

In terms of risk as measured by the returns' standard deviation, all models exhibit a considerably higher volatility than the overall market (0.0473). The the LSTM's strategy returns exhibit the largest standard deviation (0.0850), closely followed by the the GRU (0.0825) and LR (0.0821). The TCN model (0.0783) as well as the RF (0.0768) and GBC (0.0762) have a slightly lower volatility. When only taking into account the downside risk of returns, measured by the downside standard deviation, the GRU exhibits the highest risk at 0.8347, followed by the TCN (0.8138) and the LSTM (0.7899). As is the case with the standard deviation, all three deep learning models have slightly higher downside deviations than the tree-based ensemble models (GBC: 0.7742, RF: 0.7696) and considerably higher values than the overall market (0.5485).

Quantifying the financial risk in terms of the value-at-risk at 1%, the GRU (-23.56%) and the GBC (-22.29%) perform slightly worse than the rest of the tested models, which range between -20.45% (LSTM) and -19.23% (LR).

Considering only the long leg of the portfolio results in similarly profitable daily returns for all models with the TCN model also lagging behind the other models (see Table 2.14). Compared to the full long-short portfolio, the long leg returns are slightly lower for the GRU and LSTM models, while the TCN and GBC have higher mean returns. For the RF, the long leg of the portfolio generates the same mean returns as when the short leg is included in the portfolio. This implies a positive contribution of the short leg of the long-short portfolio for the GRU and LSTM and a negative mean short-only returns for the TCN and GBC. Both the RF and the LR have negligible positive contributions on the short side.

With regard to the long leg's return distribution, we observe a lower standard deviation than for the long-short portfolio for all models, indicating a detrimental contribution of the short leg in terms of added risk. The same is true when only downside risk is considered. Here, the relative advantage of the long leg of the portfolio is even more pronounced. The lower downside risk for the long leg is most notable for the TCN, where the downside risk decreases from 0.8138 to 0.4477 when the short leg is not included.

Table 2.13.: Daily return, risk, and annualized risk-return metrics for all models and the market (MKT) for the $k = 5$ long-short portfolio

	GRU	**LSTM**	**TCN**	**GBC**	**RF**	**LR**	**MKT**
Mean Return	0.01348	0.01436	0.00308	0.00991	0.01098	0.01176	0.00330
Return Stdv.	0.08254	0.08504	0.07830	0.07615	0.07683	0.08212	0.04729
Downside Risk	0.83468	0.78989	0.81382	0.77421	0.76958	0.76237	0.54851
VaR 1%	-0.23565	-0.20450	-0.20177	-0.22286	-0.19656	-0.19230	-0.14187
VaR 5%	-0.10640	-0.10520	-0.11342	-0.09984	-0.09976	-0.10364	-0.07389
CVaR 1%	-0.32224	-0.30404	-0.28801	-0.28829	-0.28309	-0.28156	-0.19054
CVaR 5%	-0.18979	-0.17214	-0.17799	-0.17402	-0.16961	-0.16893	-0.11744
Ann. Volatility	1.57702	1.62474	1.49593	1.45486	1.46775	1.56897	0.90338
Sharpe Ratio	3.12061	3.22663	0.75165	2.48733	2.73153	2.73590	1.33105
Sortino Ratio	4.89560	5.51085	1.14722	3.88098	4.32569	4.67511	1.82023
Excess Sharpe	0.10596	0.11276	-0.00213	0.07226	0.08315	0.08729	-

Table 2.14.: Daily return, risk, and annualized risk-return metrics for all models and the market (MKT) for the long leg of the $k = 5$ portfolio

	GRU	**LSTM**	**TCN**	**GBC**	**RF**	**LR**	**MKT**
Mean Return	0.01131	0.01371	0.00529	0.01108	0.01097	0.01163	0.00330
Return Stdv.	0.07259	0.07445	0.05621	0.06962	0.06851	0.07398	0.04729
Downside Risk	0.62922	0.57016	0.44772	0.59317	0.59060	0.64008	0.54851
VaR 1%	-0.15797	-0.13844	-0.12434	-0.15598	-0.14938	-0.16315	-0.14187
VaR 5%	-0.08319	-0.08026	-0.05743	-0.07369	-0.07086	-0.08074	-0.07389
CVaR 1%	-0.23227	-0.19367	-0.17349	-0.22397	-0.22932	-0.23833	-0.19054
CVaR 5%	-0.13600	-0.12351	-0.09442	-0.12579	-0.12429	-0.13877	-0.11744
Ann. Volatility	1.38691	1.42229	1.07397	1.33012	1.30886	1.41336	0.90338
Sharpe Ratio	2.97663	3.51991	1.79749	3.04240	3.05875	3.00373	1.33105
Sortino Ratio	5.44768	7.29051	3.58004	5.66458	5.62834	5.50700	1.82023
Excess Sharpe	0.14801	0.18077	0.04223	0.15336	0.14865	0.14177	-

Risk-Return Characteristics

Considering the realized excess returns in relation to the incurred risk, the ranking is the same as for mean daily returns for the $k = 5$ long-short portfolio strategy, with the LSTM and GRU performing best with annualized Sharpe ratios of 3.23 and 3.12, respectively. Only the TCN (0.75) performs worse than the general market (1.33), while all other models perform considerably better. The substantial advantage of the RF over the LR in terms of accuracy (59.3% vs. 57.5%) notably does not translate into a more favorable Sharpe ratio, as the LR has a slight edge over with a ratio of 2.736 compared to the RF's 2.732.

The overall ranking in terms of risk-return performance remains unchanged when using the Sortino ratio, which only considers downside deviation for quantifying investment risk. The LSTM leads with a ratio of 5.51 ahead of the GRU with 4.90, while the TCN falls behind all other models and the general market (1.82) with a Sortino ratio of 1.15.

On the long-only side, all models except the GRU perform better in terms of the Sharpe ratio compared with the long-short portfolio performance, whereas the GRU drops from 3.12 to 2.98. The LSTM's long side of the portfolio strategy performs considerably better than all other models with a Sharpe ratio of 3.52. All other models, except the TCN, exhibit Sharpe ratios close to 3. The RF performs slightly better than GBC (3.04) at 3.06, while the GRU (2.98) underperforms the LR model (3.00). The TCN lags far behind with a ratio of only 1.80, but still manages to outperform the market (1.33).

As is the case for the full long-short portfolio, the LSTM achieves the highest long-only Sortino ratio with a value of 7.29, well ahead of the next best models, the GBC (5.66), RF (5.63), LR (5.51), and the GRU (5.45). However, all models have a higher Sortino ratio for the long side of the portfolio compared to the portfolio as a whole. Compared to the other models, the TCN model performs considerably worse with only 3.58, but has the largest relative difference compared with the long-short portfolio of more than 100%.

Figure 2.10.: Cumulative market returns and Bitcoin (BTC) returns for the full test set date range with starting value 1 at the beginning of the test set of study period 1 (2020/04/26)

2.3.5 Discussion

This study demonstrates the potential of machine learning for cryptocurrency market prediction, as all utilized models significantly outperform a random classifier. Our analysis indicates that the employed recurrent neural networks, the temporal convolutional network, and tree-based ensembles are particularly effective in correctly classifying the relative daily performance of cryptocurrencies. Comparing the long- and short-leg predictions indicates that short legs are more predictable, as we generally document a slightly higher accuracy for short-leg predictions.

In evaluating the economic implications of these predictions, we examine the performance of long-short portfolios that trade 10% of all model constituents. The higher overall portfolio risk of the resulting long-short portfolios may be driven by a lower degree of portfolio diversification compared to the market portfolio. However, as the portfolio returns of five of the six employed machine learning models yield positive returns of at least three times the market portfolio return, the long-short portfolios based on these models outperform the buy-and-hold market portfolio on a risk-adjusted basis. The temporal convolutional neural network is a noteworthy exception, as its high relative accuracy does not directly translate into a high prediction performance. This finding indicates that the temporal convolutional neural network is most confident in classifying observations with lower corresponding absolute returns than the other models. The GRU and LSTM models appear especially well-suited for the employed trading strategy, as the long-short portfolios based on these models' predictions yield the highest risk-adjusted performance. Overall, these results indicate a challenge to weak form cryptocurrency market efficiency (Fama, 1970), albeit the influence of certain limits to arbitrage cannot be entirely ruled out.

Furthermore, the risk-adjusted outperformance of the employed long-short strategies may have been reduced by an inflation of the buy-and-hold market benchmark. The potential inflation stems from the overall market upturn in the out-of-sample periods, as the buy-and-hold market strategy is exposed to the long-side of the cryptocurrency market. At the same time, the long-short portfolios exhibit zero net exposure. Also, due to the overall market upturn, the positive performance of the long-short portfolios is primarily driven by the long-side of the long-short portfolios, despite the slightly higher short-leg classification accuracy. An exception is study period five, characterized by an overall market downturn. In that specific period, the short-leg of the portfolio is responsible for the overall outperformance of the best-performing long-short portfolios.

The presented results are subject to several limitations and assumptions. First, we assume to be able to, on average, buy and sell cryptocurrencies at mid-price. Second, we assume to be able to short-sell the considered cryptocurrencies. Short-selling generally induces additional costs and is not consistently possible for all included cryptocurrencies. In this study, the risk-adjusted outperformance of the employed portfolio strategy compared to the market benchmark is predominantly driven by the long portfolio legs. Therefore, this limitation would be more pronounced in more neutral market environments. Third, as with every empirical study, this study is limited by its finite sample size. Finally, the external validity of the results may be limited by the use of cryptocurrency price data aggregated over multiple exchanges. Regarding the latter potential limitation, Vidal-Tomás (2022) finds that the use of aggregated cryptocurrency exhibits the same processes as data from individual exchanges. He infers that aggregated cryptocurrency data is appropriate to utilize in research.

2.3.6 Conclusion

In this study, we employ several machine learning models to predict the relative daily market movements of the 100 largest cryptocurrencies by market capitalization. We show that all employed models make statistically viable predictions, whereby the average accuracy values calculated on all cryptocurrencies range from 52.9% to 54.1% for the different models. Accuracy values range from 57.7% to 59.5% when calculated on the subset of predictions with the 10% highest model confidences per class per day. A long-short portfolio strategy based on the predictions of the employed LSTM and GRU ensemble models yields an annualized out-of-sample Sharpe ratio after transaction cost of 3.23 and 3.12, respectively. In comparison, the buy-and-hold benchmark market portfolio strategy only yields a Sharpe ratio of 1.33. These results indicate a challenge to weak form cryptocurrency market efficiency, albeit the influence of certain limits to arbitrage cannot be entirely ruled out.

CHAPTER 3

CHANGED SOCIO-ECONOMIC AND LEGAL ENVIRONMENT: COVID-19 PANDEMIC

3.1 Financial Market Prediction During the COVID-19 Pandemic Using Machine Learning

After developing an understanding of the application of machine learning to the market prediction of the novel transaction object cryptocurrency, this chapter examines financial market prediction in light of the COVID-19 pandemic, which represents a shift in the socio-economic environment of financial markets. The following study examines predictability of S&P 500 stock movements during the COVID-19 pandemic using Machine Learning. The presented results show that the forecasts of a random forest and a logistic regression are significantly more accurate when a feature set including COVID-19 related data is utilized compared to a benchmark feature set without COVID-19 related variables. A feature importance analysis of the most accurate model reveals that the predictive power is not concentrated on a single COVID-19-related feature type but spreads over multiple different features.

This Section, in large parts, comprises the unpublished article: P. Jaquart, F. Furtwängler, C.-P. Wachter, M. Kirchenbauer, C. Weinhardt, *Financial Market Prediction During the COVID-19 Pandemic Using Machine Learning*, Working Paper, 2022.

3.1.1 Introduction

The COVID-19 pandemic, induced by the SARS-COV-2 virus (Fauci et al., 2020), has major ongoing implications for societies (Mishra et al., 2020; Nicola et al., 2020) and economies (James K. Jackson, 2021; Ozili and Arun, 2020) around the globe. As with the whole economy, international financial markets have experienced various effects of the pandemic. Amongst these effects are more pronounced behavioral investment biases (Li et al., 2021), increased investor fear and panic (Haroon and Rizvi, 2020) and an overall increased market volatility (Salisu and Vo, 2020; Onali, 2020). For the first months of the pandemic, researchers have documented a robust negative relationship between COVID-19 case numbers and financial market performance. These first months of the pandemic can be characterized by rapidly rising global COVID-19 case numbers (Johns Hopkins University, 2022), increased panic amongst investors (Haroon and Rizvi, 2020), and mostly negative stock market developments around the globe. While these early pandemic months have been examined by various researchers that focus on specific pandemic aspects (e.g., see Ashraf (2021); Ciner (2021); Corbet et al. (2021)), there remain open questions regarding the utilization of COVID-19-related data for financial market prediction in a later, more stable pandemic stage. Against this backdrop, we aim to tackle some of these questions by examining the predictability of S&P 500 stock price movements with COVID-19-related data. As many potential pandemic-related variables exist, we apply flexible machine learning models to shed light on this prediction problem. To summarize, our overarching research question is:

Research Question 4 *What is the performance of machine learning models predicting stock price movements of the S&P 500 during the COVID-19 pandemic?*

We answer this research question by applying different random forest models to the task of predicting daily stock price movements of S&P 500 constituent stocks. We select this machine learning method as random forests can incorporate a large number of predictive variables and have been applied successfully to other financial market prediction tasks in the past. We further make use of logistic regression models to provide a predictive performance benchmark. Our analysis utilizes a comprehensive feature set consisting of various COVID-19-related data and control variables.

Subsequently, we analyze the feature importance of all utilized features. We further evaluate the predictive power of COVID-19-related variables by comparing the out-of-sample predictive performance of different models per method type, whereby one of these models relies on the full feature set to make predictions. In contrast, the other model only utilizes data unrelated to the COVID-19 pandemic.

3.1.2 Related Work

Due to the recent nature of the COVID-19 pandemic, there is limited evidence regarding its impact on financial markets. As of the time of writing this paper, most researchers have focused on the first months of the pandemic. For instance, Salisu and Vo (2020) examine the impact of google search volume related to health news on stock indices in the twenty countries with the most COVID-19 cases as of March 30, 2020. Using daily data from the beginning of January to the end of March 2020, they find that higher health news search volumes have a negative effect on stock returns. Wang et al. (2021) analyze the impact of the COVID-19 pandemic on stocks in the solar energy sector. They find that the intensity of the COVID-19 pandemic, approximated by the number of confirmed cases and government response stringency, negatively affects solar energy stocks from December 31, 2019, to June 4, 2020. Ashraf (2021) sheds light on the moderating effect of national culture on the impact of the COVID-19 pandemic on stock markets. He utilizes daily data from January 22th until April 17 in 2020 and presents evidence for a higher negative reaction of stock markets in countries with pronounced national-level uncertainty aversion. Pham et al. (2021) examine the impact of COVID-19-related information on stock returns at a US state level. Using data from January 22th to June 30, they find that next-day stock returns in a state are negatively related to the number of COVID-19 cases in that state. They show that this effect is smaller in states with more governmental support and with better medical resources. Salisu et al. (2020) investigate the effect of global fear on commodity returns up to five days in the future on daily data between March 11 to May 18, 2020. They show that commodity returns rise as fear related to COVID-19, approximated by the Global Fear Index, increases. Onali (2020) studies the effect of reported cases and deaths in different countries on the US stock market, using daily data from April 9, 2019, to April 9, 2020. Utilizing a GARCH model, he finds that only the COVID-19 cases in China are negatively related to US stock returns. Further, he applies a vector autoregression model, which also indicates that the reported deaths in France and Italy negatively impact stock returns and positively impact the volatility index VIX. Cepoi (2020) analyzes the relationship between news and stock market index returns in the US and major European countries. He applies a panel quantile regression framework to daily data from February 3, 2020, to April 17, 2020, and finds that stock market indices and different pandemic-related news exhibit asymmetric relationships. Corbet et al. (2021) investigate the impact of the COVID-19 pandemic on companies whose brand is associated with the word "Corona." They deploy a GARCH model to analyze a dataset ranging from March 11, 2019, to March 10, 2020, and present evidence for a negative knock-on effect of the pandemic on the stock performance of these companies. Subramaniam and Chakraborty (2021) construct a COVID-19 daily fear index based on Google search volume between March and August 2020. They find a robust negative relationship in-sample between fear and returns of major stock market indices of various countries up to five days in the future. Ciner (2021) examines US stock market predictability in the early stage of the COVID-19 pandemic. Applying a LASSO approach to a daily data set from January 2 to April 16, 2020, he finds that high yield and investment-grade corporate bonds exhibit significant predictive power for the US stock market. Narayan et al. (2021) analyze the impact of different governmental responses of G7 countries to the COVID-19 pandemic. Investigating daily data between July 1, 2019, and April 16, 2020, they find that lockdowns travel bans, and stimulus packages positively affect stock returns in the considered countries.

3.1.3 Methodology

3.1.3.1 Data

In this study, we utilize daily data ranging from the beginning of July 2020 until the end of December 2021. Our asset universe consists of all stocks in the S&P 500, whereby we use the index composition as of September 1, 2021 (the start date of our test set) to prevent a survivorship bias in our data. For company-specific data, we rely on the Compustat database (University of Pennsylvania, 2022), from which we acquire daily closing price, daily total return factor, and the S&P industry sector code. Furthermore, we download the values for the factors *small-minus-big*, *high-minus-low*, *market*, *momentum* and the daily risk-free rate from Kenneth French's data library (French, 2022). We obtain daily values for the number of PCR tests in the US and the number of COVID-19 cases, deaths, and recoveries in all countries from the COVID-19 data repository of the John Hopkins University (Johns Hopkins University, 2022). From the *Our World in Data* repository (Ritchie et al., 2020) by the Oxford University, we acquire the daily values for the number of hospitalized COVID-19 patients placed in an intensive care unit, vaccinations against COVID-19, the COVID-19 reproduction rate, and government stringency, for each country. We obtain the daily values of the infectious disease equity market volatility index based on the work of (Baker et al., 2019) from the website of the *Economic Policy Uncertainty* research team (Economic Policy Uncertainty, 2022). Additionally, we download the relative number of google search queries for the keyword 'coronavirus' from Google Trends and create a continuous trend time series by scaling the values relative to the training period. Finally, acquire information about announcements of governmental fiscal help related to the finance sector-related from the *worldbank* homepage (Worldbank, 2022). We only include NYSE trading days into our data set, remove days with missing observations and use the UTC timezone to aggregate all data. For sources that do not provide the data in UTC, we shift the respective daily timestamp by one day into the future to ensure that the feature set at a certain point in time does not consist of future data.

3.1.3.2 Feature Engineering

To receive our final feature set, we transform the downloaded data as follows: We calculate the daily total excess returns for each company by scaling the relative change of the daily closing prices with the daily total return factor and subtracting the daily risk-free return. We obtain the beta values for a specific day and company by regressing the daily values of the four Carhart-factors (Carhart, 1997), three of which are based on Fama and French (1993)), on the daily excess return of that company for the last 250 trading days prior to the selected day. This procedure is repeated for each day in our data set to receive daily beta values for every company in the data set. We construct additional features that represent the recent feature history to enable the different models to utilize information about the feature development over time. Concretely, we use the following set of intervals to aggregate the feature history for a specific feature: (0, 1], (1, 5], (5, 10], (10, 20], (20, 30], (30, 40]. Every interval in the set denotes the days prior to the prediction point over which a feature is aggregated. For return-based features, we calculate multi-period excess returns over these intervals. We construct a binary dummy variable for fiscal support announcements for each interval that indicates whether additional fiscal support was announced over that time window. We calculate the absolute and relative growth over the respective intervals for all other features that vary over time. Finally, we create one-hot encoded dummy variables for the weekday and the S&P industry sector. Table 3.1 gives a summary over the utilized feature types.

Table 3.1.: Overview of the utilized feature types

COVID-19-related Features	**Other Features**
Number of active COVID-19 cases	Weekday
Number of COVID-19 recoveries	S&P industry sector code
Number of COVID-19 related deaths	Beta small-minus-big
Number of officially conducted COVID-19 tests	Beta high-minus-low
People vaccinated against COVID-19	Beta market
COVID-19 reproduction rate	Beta momentum
Government response stringency	Daily excess return
Google search trends	
Infectious disease equity market volatility index	
Announcements of goverment fiscal help	
Number of intensive care unit COVID-19 patients	
COVID-19 reproduction rate	

3.1.3.3 Targets

We construct binary target variables based on the next-day excess returns of the different companies. Concretely, we use the median return of the training set to split the two different classes. Thereby, observations with a higher next-day excess return than the training set median are grouped into Class 0, and observations with a lower next-day excess return are grouped into Class 1. This procedure ensures a balanced training set, which is especially important for noisy prediction problems, such as financial market prediction tasks. An imbalanced training set might cause prediction models to learn to predict the majority class solely.

3.1.3.4 Training, Validation, and Test Set

The first 12 months of data, which incorporate 124,146 observations, constitute our set for model training. We use the subsequent two months of data, consisting of 20,461 observations, as our validation set for model tuning. The final four months of data, amounting to 40,811 observations, make up our test set used for an out-of-sample evaluation of model performance and feature importance.

3.1.3.5 Models

Random Forest

Random forests are an ensemble learning method consisting of multiple, not perfectly correlated decision trees (Breiman et al., 1984; Breiman, 2001). They can be used for classification and regression problems and have been applied successfully to various financial market prediction tasks (Krauss et al., 2017; Gu et al., 2020). Here, we use random forest classifiers consisting of 100 individual decision trees. Furthermore, we perform a grid search on the validation set to optimize the maximum individual decision tree depth and the number of features considered at each split in a tree. As our feature set utilizes a relatively high number of dummy variables, we also test a relatively high numbers of maximum features randomly considered per split. A higher number of features generally leads to more precise individual trees but a higher correlation amongst the different trees in the random forest. Based on the grid search on the validation set, we choose a parameterization with a tree depth of 5, whereby 40% of all features are considered per split for our final model that includes all features. For the model that is not trained on COVID-19-related features, the final model has an individual tree depth of two and considers the root of the total number of features at each split. The full parameter grid is depicted in Table 3.2. For the other parameters of our random forests, we use the default parameter values of the scikit-learn library (Defazio et al., 2014).

Table 3.2.: Parameter tuning grid

Model	Parameter Tuning Grid
Random Forest	Maximum depth: {1, 2, 3, 5, 10, 15, 20, None}
	Maximum Features per Split: {sqrt, log2, 0.2, 0.4, 0.6, 0.8, 1}
Logistic Regression	Penalty: {l2, l1, elasticnet, none}

Logistic Regression

We further utilize a logistic regression as a benchmark model, as logistic regression models have become a standard for classification tasks over the last decades (Hosmer Jr et al., 2013). Logistic regression models follow the same principles as linear regression models, with the difference of a binary target variable. For the applied logistic regression model, we tune the applied penalty term based on the validation set accuracy as detailed in Table 3.2. Multiple penalty specifications yield the same validation set accuracy for the logistic regression models trained on the complete feature set. In this case, we apply an l2-penalty, which represents the default value of the scikit-learn library (Pedregosa et al., 2011). For the logistic regression model only using the features not directly related to COVID-19 uses, the l1-penalty yields the highest validation set performance. We further train our logistic regression models with the 'saga' (Defazio et al., 2014) solver, as it trains fast on larger datasets and supports all considered penalty values in our tuning grid.

3.1.3.6 Evaluation

We evaluate our predictive models using the accuracy measure. Additionally, we utilize Diebold-Mariano tests (Diebold and Mariano, 1994) in combination with the mean absolute error loss function to compare the accuracy of different model forecasts pairwise, similar to Fischer and Krauss (2018). Since we train our models on a balanced training set, we can further calculate the probability for a random classifier with an accuracy of 50% to achieve a given model's test set accuracy based on the binomial distribution:

$$X \sim B(n = \#\text{test}, p = 0.5, q = 0.5), \tag{3.1}$$

where #test ($\widehat{=}40,811$) denotes the number of test set observations. Additionally, we calculate the precision and recall measures and the F1-score for every model and target class to examine potential differences in the predictability of the two classes. Also, we evaluate the feature importance of our random forest model using the measure of permutation feature importance. We consecutively permute all features randomly and calculate the overall decrease in model performance. We scale the overall decreases in model performance so that the feature importance scores sum up to one. If the decrease in model performance is higher for a given permuted feature, the respective model relies more on that feature for correct predictions.

3.1.4 Results

3.1.4.1 Model Performance

Table 3.3 presents the predictive accuracy of the different prediction models. The random forest utilizing COVID-19-related data achieves an out-of-sample predictive accuracy of 53.84%, while the logistic regression benchmark model trained on the same data set achieves a predictive accuracy of 52.74%. In comparison, the random forest and logistic regression model, trained on the data set without COVID-19-related data, only exhibit a test set accuracy of 50.87% and 50.56%, respectively.

Table 3.3.: Overview of the different model out-of-sample accuracy scores

	Random Forest		**Logistic Regression**	
	Full Feature Set	No Covid-Related Features	Full Feature Set	No Covid-Related Features
Accuracy	0.538409	0.508711	0.527358	0.505574

Table 3.4 presents the results of the Diebold-Mariano tests (Diebold and Mariano, 1994), which show that the forecasts by the random forest model trained on the full feature set are significantly more accurate than the forecasts of all other models at a significance level of 1%. Additionally, the logistic regression model trained on the full feature set yields significantly more accurate forecasts than the models only trained on data unrelated to COVID-19.

Table 3.4.: Diebold-Mariano test p-values to reject the null hypothesis towards the alternative hypothesis that the forecast of model i on the test sample is more accurate than the forecast of model j

i \ j		Random Forest		Logistic Regression	
		Full Feature Set	No Covid-Related Features	Full Feature Set	No Covid-Related Features
Random Forest	Full Feature Set	-	0.000000	0.001792	0.000000
	No Covid-Related Features	1.000000	-	1.000000	0.328625
Logistic Regression	Full Feature Set	0.998208	0.000000	-	0.000000
	No Covid-Related Features	1.000000	0.671375	1.000000	-

As shown in Table 3.6, we find that all models make statistically viable predictions. Our random forest and logistic regression model trained on the full features set have respective probabilities of 1.7875E-55 and 1.1295e-29 for a true model accuracy of 50%. For the random forest and the logistic regression trained on the feature set without COVID-19 data, these probabilities are 8.0668e-06 and 3.1262e-04. Table 3.5 presents precision, recall and F1-scores of the different models. It shows that the predictions of the full-feature random forest model are mostly balanced over both classes, as precision and recall are over 50% for Class 0 and Class 1. In comparison, the best performing logistic regression model is more likely to correctly classify observations that belong in Class 0, as it exhibits a recall score of 77.92% for Class 0 and 25.76% for Class 1.

Table 3.5.: Overview of the different models' precision, recall and F1-scores

		Random Forest		Logistic Regression	
		Full Feature Set	No Covid-Related Features	Full Feature Set	No Covid-Related Features
Class 0	**Precision**	0.555060	0.526326	0.529684	0.512789
	Recall	0.547812	0.513177	0.779276	0.907783
	F1-Score	0.551412	0.519668	0.630684	0.655371
Class 1	**Precision**	0.521000	0.490744	0.519914	0.426090
	Recall	0.528309	0.503913	0.256760	0.073541
	F1-Score	0.524629	0.497242	0.343755	0.125433

Table 3.6.: Probabilities for the different prediction models to exhibit a true model accuracy of 50%

	Random Forest		Logistic Regression	
	Full Feature Set	No Covid-Related Features	Full Feature Set	No Covid-Related Features
Probability	1.787494E-55	8.066806e-06	1.129511e-29	3.126180e-04

3.1.4.2 Feature Importance

Figure 3.1 presents the permutation feature importance scores of the 30 most relevant features of the full-feature random forest model. It shows that the model generally relies on numerous features to make predictions, as individual feature importance scores do not exceed 2%. The most important features are the newspaper-based infectious disease equity market volatility tracker and features that indicate the state of the pandemic, such as the changes in COVID-19-related active cases and deaths, the number of conducted COVID-19 tests and the number of COVID-19 patients in intensive care units. Regarding the regional distinction of features, we find that features indicating the state of the pandemic in the US *(11 features)* are most common among the 30 most important features, followed by features that indicate the state of the pandemic in the EU *(five features)* and China *(four features)*, as well as an aggregated world-wide scope *(four features)*. Furthermore, the remaining features represent different temporal characteristics of the infectious disease equity market volatility index, which is based on US newspapers.

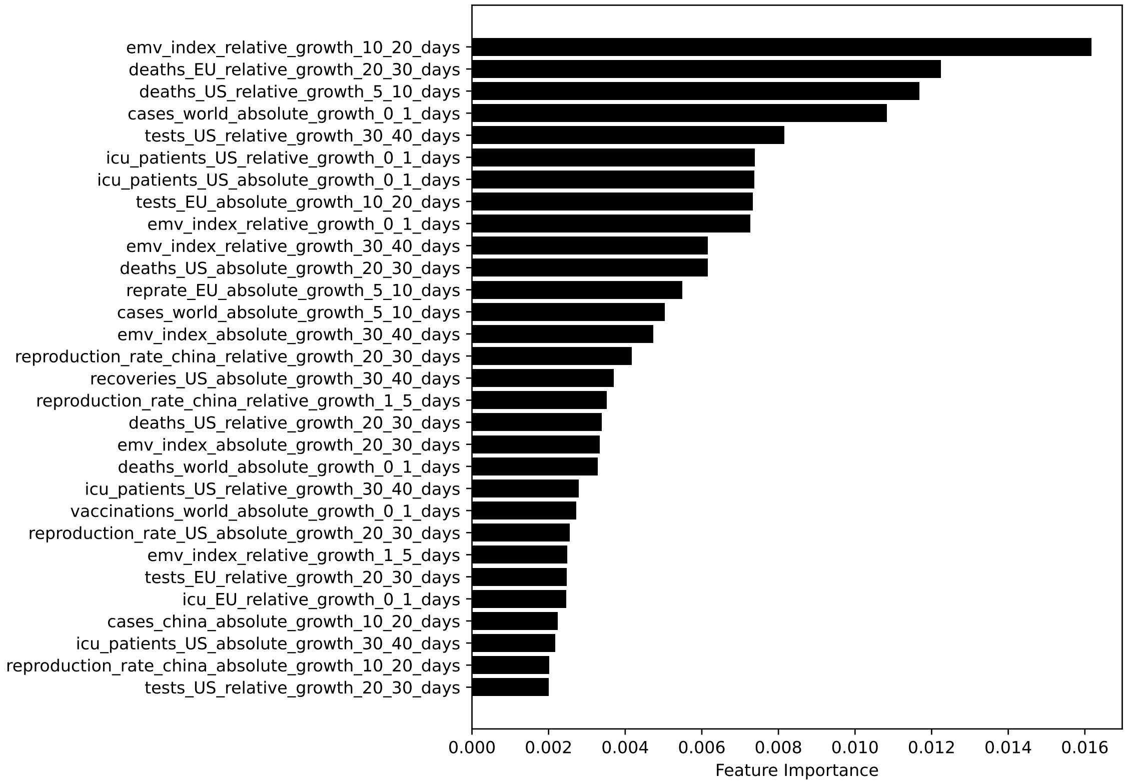


Figure 3.1.: Permutation feature importance scores for the most relevant features of the full-feature random forest

3.1.5 Discussion

Our study demonstrates that COVID-19-related data exhibits predictive power for next-day price movements of S&P 500 constituent stocks. While previous studies have primarily examined the first months of the pandemic, a stage in the COVID-19 pandemic that can be characterized by investor fear (Haroon and Rizvi, 2020) and high market volatility (Salisu and Vo, 2020; Onali, 2020), we shed light on a more extended, subsequent period. We document that the random forest model yields more accurate and balanced forecasts than the logistic regression model when training on the full feature set in our study. In combination with the finding that our grid search optimization results in medium-sized individual trees, this indicates that allowing for a limited number of high-dimensional feature interactions improves predictions for the problem at hand. The feature importance analysis reveals that the predictive power is not concentrated on a single COVID-19-related feature but is rather spread over multiple different feature types. The finding that US-based pandemic data tends to be most relevant for making predictions might be explained by the fact that S&P 500 companies generate most of their revenue domestically (Silverblatt, 2019). An explicit limitation of this study stems from the dynamic evolution of the COVID-19 pandemic, as different mutations of the SARS-Cov-2 virus may become prevalent over time, and medical research will develop new methods for treatment and vaccination. While we aim to reduce part of this limitation by examining a relatively long period, our study still provides a snapshot in time, and similar analyses should be repeated for future stages of the pandemic. Furthermore, while we include well-established benchmark predictor variables (Gu et al., 2020) such as the three Fama-French factors (Fama and MacBeth, 1973), we can not entirely rule out a potential omitted variable bias. In this study, we evaluate market predictability through a binary classification task to create an overall benchmark in the academic literature in demonstrating the overall feasibility of financial market prediction using COVID-19 data. Based on this finding, future researchers could examine classification models with more than two classes or regression models trained to predict the exact excess returns of the different stocks. However, one has to keep in mind that this would further increase the complexity of the prediction problem and that one should generally expect relatively low predictability of developed financial markets in light of financial market efficiency. Future research could also examine the use of other machine learning methods and apply additional COVID-19-related features to the problem. While we have aimed to create a comprehensive feature set, new features might become available in the future due to new COVID-19-related developments.

3.1.6 Conclusion

Our study examines the predictability of S&P 500 stock movements during the COVID-19 pandemic utilizing COVID-19-related features. We find that a random forest and a logistic regression model trained on a comprehensive feature set that includes various COVID-19-related features yields more accurate predictions than their counterparts trained only on the subset of features unrelated to the COVID-19 pandemic. Concretely, the full-data random forest model exhibits a predictive out-of-sample accuracy of 53.84%, whereby the respective logistic regression model exhibits a predictive out-of-sample accuracy of 52.74%. In comparison, the random forest and logistic regression models that are trained on data not directly related to COVID-19 exhibit a respective predictive accuracy of 50.87% and 50.56%. However, we still find that all employed models significantly outperform a random classifier. A feature importance analysis of the best model reveals that their predictive power is not concentrated on a single COVID-19-related feature type but spreads over multiple different features. Analyzing the regional differences, we find that features that reflect the pandemic development in the US tend to be more relevant than international developments, which may be explained by the fact that companies in the S&P 500 still generate a majority of their revenue domestically.

CHAPTER 4

CHANGED AGENT BEHAVIOR AND MARKET STRUCTURE: THE SHIFT FROM ACTIVE TOWARDS PASSIVE INVESTMENT

After examining the utilization of machine learning models for market prediction in light of developments in other central financial market elements, this chapter analyzes the implications of an important development regarding agent behavior and market structure, namely the shift from active to passive investment. Furthermore, it examines the implications of more accurate market forecasts of active investors caused, for instance, by employing machine learning methods for trading purposes. The following study presents a round-based simulated market framework in which active, passive, and random investors repeatedly optimize their portfolio weights and interact with each other through issuing orders. The presented results show that higher fractions of active investment within a market lead to increased fundamental market efficiency. The marginal increase in fundamental market efficiency per additional active investor is lower in markets with higher levels of active investment. Furthermore, the results indicate that large fractions of passive investors within a market may facilitate technical price bubbles, leading to market failure. Regarding changes in active investors' prediction accuracy, the results show that more accurate predictions of active investors increase fundamental market efficiency. This increase is more pronounced in markets with a higher level of active investment.

This Section, in large parts, comprises the unpublished article: P. Jaquart, M. Motz, L. Köhler, C. Weinhardt, *The Impact of Active and Passive Investment on Fundamental Market Efficiency: A Simulation Study*, Under Review, 2022.

4.1 The Impact of Active and Passive Investment on Fundamental Market Efficiency: A Simulation Study

4.1.1 Introduction

Passive investing has steadily grown relative to active investing over the last decades (Blitz, 2014; Anadu et al., 2019) and is expected to overtake active investing by 2026 in the US equity market (Seyffart, 2021). In the US domestic equity-fund market, passive vehicles have already overtaken active ones due to the large number of passive funds tracking the S&P 500 index (Seyffart, 2021). Passive investors select the portfolio weights of assets within the risky fraction of their portfolio based on asset market capitalization (Sharpe, 1991). Therefore, passive investors only need to decide about the fraction they want to invest in the risk-free asset and the risky market portfolio for each period (Pedersen, 2018). In contrast to passive investors, active investors usually trade based on the assessment of asset mispricings. As these assessments tend to change frequently, active investors generally trade more often than passive investors (Sharpe, 1991).

Against the backdrop of the rise of passive investment, it is crucial to understand the implications of this significant trend for financial markets and their participants. Therefore, financial researchers have begun to empirically examine different aspects of the shift from active to passive investment. These empirical analyses suggest that this shift may have reduced the information contained in individual asset prices (Sushko and Turner, 2018) and led to higher systematic market risks (Anadu et al., 2019). In this work, we want to shed more light on how different forms of investment affect financial markets. Concretely, we address the research question:

Research Question 5 *How do different levels of active and passive investment affect fundamental price efficiency?*

Besides shifts across investment types, specific investment forms are also evolving. Consistent with findings that machine learning models can improve financial market forecasting (Fischer and Krauss, 2018; Rasekhschaffe and Jones, 2019; Gu et al., 2020), a large proportion of professional active investors have adopted machine learning models for trading and portfolio management (BarclayHedge, 2018; Petropoulos et al., 2022). Given this trend, it is important to also assess the impact of differences in the accuracy of active investors' market forecasts on underlying market efficiency. Hence, our next research question states:

Research Question 6 *How do different degrees of accuracy of active investors' market forecasts affect fundamental market efficiency?*

To answer these research questions, we create a simulated financial market and closely examine the impact of different investor compositions and other market parameters, including the forecast accuracy of active investors, on fundamental market efficiency. Different stylized investors interact with each other in a simulated market through issuing orders, which are matched via common continuous double auctions. Thereby, market prices are solely set by the investor behavior without external interventions. While there exist different interpretations and levels of market efficiency in the literature (e.g., see (Fama, 1970, 1991)), in this paper, we focus on market efficiency in terms of the deviation between market prices and fundamental asset values. In this work, we denote this form of market efficiency with the term *fundamental market efficiency*.

Our paper has two main contributions. First, we show how different investor compositions affect fundamental market efficiency. In doing so, we find that larger fractions of active investment and lower fractions of passive investment within a market results in higher market efficiency. Thereby, the marginal increase in market efficiency is lower for an additional active investor if there is a high level of active investment within a market. We find that large portions of passive investment within a market may facilitate price bubbles and lead to market failure. Second, we evaluate the impact of different market parameters, for instance market frictions and individual target price forecasting errors, on fundamental market efficiency to increase the robustness of our results in the context of constantly evolving financial markets. We find that market frictions in the form of higher transaction costs and stricter portfolio constraints reduce fundamental market efficiency. Additionally, we find that a lower level of risk aversion of individual investors, tends to increase fundamental market efficiency. Lastly, we show that less volatile individual target price forecasts of active investors translate into higher fundamental efficiency of market prices.

4.1.2 Related Work

4.1.2.1 Implications of Active and Passive Investment

Anadu et al. (2019) empirically investigate the implications of the past shift from active to passive investment strategies for financial stability. They find that this shift may decrease liquidity transformation risks, but increase market volatility. Furthermore, Anadu et al. (2019) present evidence indicating that passive investment facilitates the co-movement of assets.

Pedersen (2018) challenges William Sharpe's equality, which states that"before costs, the return on the average actively managed dollar will equal the return on the average passively managed dollar" (Sharpe, 1991). Pedersen (2018) argues that the assumption of a never-changing market portfolio does not hold for real-world financial markets and presents ways how active traders can generate profits. For instance, adequately informed active traders could outperform non-informed active traders, influenced by cognitive biases. Additionally, only active investors can identify and participate in profitable IPOs. Furthermore, passive investors need to trade when the composition of their respective benchmark index changes, which can be anticipated and exploited by active investors. Pedersen (2018) concludes that there exists an equilibrium level of active and passive investment that keeps the market close to but not perfectly efficient.

Sushko and Turner (2018) examine the effect of the shift from active to passive investing on securities markets. They find that active mutual funds are subject to persistent outflows in stress periods, while the flows of passive mutual funds remain relatively stable. Additionally, the authors find a positive relationship between the weight of a company in the Bank of America Merrill Lynch Global Broad Market Corporate Index and the company's leverage factor. According to Sushko and Turner (2018), a further expansion of the passive fund management industry may have two main consequences: First, security-specific information could decrease in prices because passive fund managers do not use this information for their valuations. Second, security pricing can be influenced by an increasing number of passive managers selling or buying the entire set of index constituents by tracking an index.

These transactions could lead the prices of the assets in the index to be subject to higher co-movement.

Schwert (2003) shows that market anomalies published in the academic financial literature are integrated into trading strategies and hence disappear after publication. Therefore, he states that research on market anomalies makes financial markets more efficient.

Sullivan and Xiong (2012) analyze the effects of increased index trading. The authors find that the rising fraction of passively managed equity indices leads to an increased systematic risk of equity markets. The higher amount of passive investing amplifies the trading commonality of the index assets caused by interactions between market participants. This commonality increases systematic fluctuations in aggregate demand, which have a fundamental impact on markets and portfolio compositions. Moreover, Sullivan and Xiong (2012) state that more equity index trading leads to increasing stock return correlations. Additionally, the authors find that equity betas have risen and converged between 1997 and 2010.

Appel et al. (2019) empirically analyze the effect of passive investors on investment strategies and investment results of active investors. The authors find that if passive mutual funds own the shares of a company to a more considerable degree, activists are more likely to be represented on the company's board. Moreover, they show that a high fraction of passive traders owning a company's stocks is positively related to the number of proxy fights and settlements of a company. Finally, Appel et al. (2019) concludes that free-rider problems are weakened by the increasing number of passive institutional investors.

Qin and Singal (2015) examine the external effects of indexing on stock price's efficiency. By analyzing a sample of stocks, they find that a higher degree of indexing is related to less efficient stock prices. Concretely, Qin and Singal (2015) find a higher post-earnings-announcement drift and a more significant random walk deviation of stock prices for higher degrees of indexing. Qin and Singal (2015) suggest that price efficiency is decreased by indexing and passive trading, as passive investing lowers the appeal of information collection and arbitrage.

In his work, Blitz (2014) addresses the shortcomings of passive investment strategies. First, he outlines that passive investment constitutes free-riding, as passive investors rely on active investors to keep markets efficient. Blitz (2014) states that the relationship between fundamentals and asset prices would be destroyed if there were only passive investors. Moreover, he describes specific security characteristics which have been shown to cause a weak asset performance. Passive investors can not avoid investing in these securities when they are present in the respective passively replicated index.

Belasco et al. (2012) examine how passive investing affects corporate valuations by analyzing the relation between index fund money flow and company valuations. They find that valuations of index constituents are positively correlated with index fund money flow, whereas valuations of non-constituents are not. Moreover, they find that this impact does not directly divert since money flow also impacts valuations the following month after the flow. Belasco et al. (2012) conclude that mispricing caused by index fund investing could reduce stock market efficiency and manipulate how investors evaluate the performance of actively managed funds.

French (2008) analyzes the cost of active investing and finds that active investors pay 0.67% of the total market value of a stock for search costs. Moreover, he shows that society faces capitalized price discovery costs, which can amount to up to 10% of a stock's market capitalization. French (2008) finds that between the years 1980 to 2006, the average annual return of a typical active investor could have risen by 67 basis points if the investor switched from active to passive investing.

4.1.2.2 Financial Market Simulation

Ponta et al. (2011) simulate an artificial stock market that consists of zero-intelligence traders and analyze the effect of dividend and external cash flows on the market outcome. The traders randomly allocate a fraction of their wealth into different stocks, whereby asset prices are determined by aggregating demand and supply. Despite using zero-intelligence traders, the authors can reproduce several stylized facts (e.g., volatility clustering) in the resulting price series. The stylized facts reproduction does not depend on dividend payments and external cash flows.

Cocco et al. (2017) simulate an artificial agent-based cryptocurrency market in which heterogeneous agents trade bitcoins. The authors model several market characteristics of the bitcoin market (e.g., bitcoin mining, investor distribution). Cocco et al. (2017) examine whether their simulated market exhibits stylized facts known from real-world financial markets. They define two types of agents, namely momentum-based technical traders and random traders, whose orders are matched via limit order books. The authors find that the price series data of the artificial market shows three stylized facts of real financial time series data: Unit-root property, fat tails, and volatility clustering.

Bertella et al. (2014) use an agent-based artificial market to analyze the effect of (over-)confidence (Kahneman and Riepe, 1998) on market outcomes. In their model, agents can either be fundamental or technical traders. Fundamental traders estimate future asset values using the Gordon dividend growth model, whereas the technical traders use a moving average with different time horizons to estimate asset values. Both stylized agent types aim to maximize a utility function based on constant absolute risk aversion. Prices are calculated based on a market impact function based on Farmer and Joshi (2002) for varying levels of the agents' market confidence. The authors find that higher confidence levels are positively correlated with investment returns. However, market confidence also has negative effects, leading to increased price volatility.

Benhammada et al. (2017) implement an artificial stock market based on continuous double auctions to identify sources of bubbles and crashes in financial markets. The modeled agents can be grouped into fundamental, noise, technical, and hybrid traders. Noise traders issue orders randomly, which increases market liquidity. Fundamental traders issue orders based on a function to calculate the fundamental value, while technical traders issue their orders based on the direction of the forecasted price and the market liquidity. Hybrid traders change their stylized trading behavior between technical and fundamental trading based on the market state. Benhammada et al. (2017) find that prices deviate from fundamental values if technical traders dominate the artificial market. However, these markets lack characteristics of financial price bubbles in the real world. The price bubbles become more realistic when hybrid traders are predominant within the market. The authors find no evidence for the existence of market bubbles in the case that fundamental traders dominate the market.

Goykhman (2017) implements a sentiment-driven artificial financial market and examines how the wealth of different agents develops over time. In the market, agents do not maximize a utility function but issue orders depending on three time series processes: buy/sell imbalance, jump volatility, and trading intensity. These orders are matched via limit order books. The author finds that non-trivial volatility sentiment processes result in large stock returns different from the log-normal distribution. The use of non-trivial buy/sell sentiments leads to predictable price trends. Furthermore, Goykhman (2017) finds that the results do not depend on the initial wealth distribution amongst the agents, as the wealth is distributed very quickly with a power-law Pareto tail amongst the agents in any case.

Katahira et al. (2019) create an agent-based artificial asset market and analyze stylized market facts. They model agents as technical traders and calculate asset prices for each period from the aggregated excess demand, assuming sufficient market liquidity. They find several stylized facts in the time series data of resulting market price returns (e.g., heavy tails and conditional heavy tails) but cannot reproduce the gain/loss asymmetry.

Khashanah and Alsulaiman (2017) construct an agent-based artificial market in which agents can trade a risk-free asset and a risky asset. They aim to identify the causes of market instability depending on the information flow between the agents. Agents are randomly selected to adopt four different strategies: random, fundamental, momentum/technical, and adaptive trading (using neural networks to predict asset prices). Thereby all but the technical traders optimize a utility function based on Markowitz (1952). Asset prices are calculated based on the bid and ask prices, but the underlying market mechanism does not utilize order books. The authors simulate jump events that affect the market and test whether agents can react appropriately to these jumps. They find that the outputs of the scenarios depend on the market state regarding information awareness. In states of systematic ignorance, mean volatility and the volatility index are lowest. The volatility index and fear index increase for a larger number of hubs or hermits in a network.

Moiseev and Akhmadeev (2017) implement an agent-based artificial stock market and examine resulting wealth distribution and price movements. In their simulation, agents randomly issue orders, and the turnover maximization criterion determines prices. The authors find that wealth distribution becomes increasingly positively skewed over time. The behavior patterns of the agents influence the speed of inequality growth. Thereby, the inequality of the wealth distribution grows the fastest in a setting in which most agents are issuing bid orders, followed by a setting in which most agents issue sell orders. The wealth inequality grows the slowest when there is a balance between buy and sell orders in the market.

Ponta and Cincotti (2018) simulate an agent-based artificial stock market to examine the influence of agents' networks on the structure of the market. Agents decide about issuing orders based on their vision of the market trend and the average sentiment towards all assets. The resulting supply and demand curves are matched to obtain market prices. The authors find an intrinsic structural resilience of the stock market. Moreover, the inclusion of the network between agents leads to a higher number of stylized facts reproduced in the artificial market.

Wu et al. (2018) implement an agent-based stock market in order to investigate the stock price dynamics in agent networks. Agents are fundamental or technical traders and aim to maximize their individual utility function, which is based on constant absolute risk aversion. All agents are connected via networks, which allows them to collect information from their neighbors. Orders are matched via continuous double auctions. The authors find that small-world networks lead to a decreasing kurtosis of returns. The return kurtosis is lower for a higher reconnection probability between nodes. Wu et al. (2018) find that changing the network structures does not affect the standard deviation of returns. Finally, the authors conclude that the level of information efficiency has a manifold impact on the market outcomes for diverse network structures.

Vanfossan et al. (2020) construct an artificial stock market to evaluate the success of different trading strategies. They model agents as investors and mutual funds, whereby investors have a lower buying power but are more frequent than mutual funds. The agents are connected via media networks and social media. Based on the network and their strategy, investors issue buy or sell orders, which are matched through continuous double auctions. Finally, the authors calculate the mean returns for each strategy for a 50-day period. They find that the strategy based on relative asset strength constitutes the most successful strategy, whereby the strategy based on market index acceleration is the least successful.

Mathieu and Brandouy (2010) introduce an API for artificial stock markets that allows for a broad spectrum of configurations. Agents randomly place orders without utilizing information from the market or other agents. These orders are matched via order books. Mathieu and Brandouy (2010) highlight that the asset returns of the artificial market are similar to the ones of a real-world data sample and that stylized facts can be reproduced successfully.

4.1.3 Methodology

We introduce a simulated financial market where different heterogeneous agents buy and sell stocks based on their respective utility functions, following (Lebaron, 2001). Relative to other financial market simulation frameworks our market model exhibits a rather high degree of complexity, as we create a limit-order market in which heterogeneous agents conduct individual utility-based portfolio optimization and trade multiple assets. We choose this relatively high model complexity, to increase the external validity of our results. However, it still constitutes, like economic models in general, a significant simplification of the real world. In this shapter, we present the market framework and market parameters. We systematically modify central market aspects (e.g., agent composition, agent target price forecasting errors) to increase the robustness of our results with regards to different market situations and to analyze the impact of these central market aspects on market outcomes. To further increase the robustness of our results, we run every simulation setting on eight different random seeds, whereby each individual simulation consists of 100 simulation rounds. One simulation round is considered a quarter of a year in the real world. Agents adjust their portfolio holdings once per simulation round based on their stylized behavior and utility functions.

4.1.3.1 Assets

The agents' portfolios consist of four different risky assets and risk-free cash holdings. We choose a number of four risky assets to still enable a relative weighting between the different risky assets, while ensuring a feasible computational complexity of the portfolio optimization problem. Holding cash is equivalent to investing into a risk-free asset with zero return, which is a commonly used economic models (Clarkson et al., 1996; Ovtchinnikov and McConnell, 2009; Adam-Müller and Panaretou, 2009). Stocks are traded on the simulated market and have an underlying fundamental value and a publicly observable market value. At the time of inception, every artificial stock is assigned a random real-world equivalent. For every random seed, real-world equivalent stocks are sampled randomly amongst the constituents of the S&P 500 index as of the start of 1996. We calculate these stocks' quarterly total-return time series using data from the CRSP/Compustat merged database and exclude stocks that have not been traded publicly constantly between 1986 and 2020. We further exclude other stocks that do not have complete and uninterrupted time series during that time, which are necessary for our simulation setting where we need a real-world total return for each round. We acknowledge that this exclusion induces a survivorship bias in the remaining return series. However, we argue that this survivorship bias should not significantly affect the critical results of our analysis of the individual stocks, as the survivorship bias would only affect random seeds, in which a company with an incomplete return series would be sampled into one of the four risky stocks of a market. Even in these cases, the return forecasting process of active agents is the same for different return levels. We use the first 10 years of the resulting time series, namely the data from the beginning of 1986 until the end of 1995, for parameter initialization (e.g., variance-covariance matrix) and use the remaining data from the beginning of 1996 to the end of 2020 to run the simulation. Specifically, at the end of each simulation round, the fundamental value for each artificial stock is updated based on the total quarterly return of the respective real-world stock. We denote the fundamental value of stock i at time t with p_{it}^{true} and the total number of risky assets with I. While this fundamental price is not directly observable for market participants, every stock also has a publicly observable market price, p_{it}^{market}, that gets updated continuously throughout the simulation rounds and is exclusively determined by the trading activity in the simulated market. A market capitalization-weighted combination of all risky assets constitutes the market portfolio at a given point in time. At the beginning of the simulation, we initialize all artificial stocks with a fundamental value and a market value of 100 USD.

4.1.3.2 Market Mechanism

We implement continuous double auctions to aggregate supply and demand in our simulated market. We choose continuous double auctions over a simpler market clearing mechanism, as continuous double auctions are used within most real-world stock markets and therefore make the market setting more realistic (Lebaron, 2001).

Over each simulation round, the traders gradually place their buy and sell orders for the different stocks. Orders are matched (partially) with suitable orders in the respective order book, if possible. Parts of orders that can not be executed directly are added to the respective order book. At the beginning of each simulation round, all order books are cleared.

4.1.3.3 Agents

In our simulated market setting, we distinguish between three different stylized agent types: active investors, passive investors, and random investors. We choose a total number of 500 agents for each simulated market setting, as we find that these parameter value leads to a stable convergence of market prices while ensuring computational feasibility. In every simulation round, each agent determines their target portfolio weights once and issues limit buy or sell orders based on these target portfolio weights. Active investors and passive investors both maximize the following commonly used utility function (Bodie et al., 2018) based on the modern portfolio theory (Markowitz, 1952):

$$\hat{U} = \hat{r} - 0.5\gamma\ \hat{\sigma}^2, \tag{4.1}$$

where $\hat{U}$ is the estimated utility, $\hat{r}$ is the estimated portfolio return, $\hat{\sigma}^2$ is the estimated portfolio return variance, γ is the risk aversion, which is randomly initialized with a value between 2 and 6 at the beginning of the simulation, and 0.5 is a scaling convention. While active investors and passive investors have the same utility function, they differ in the way they generate their return estimates, as we describe in detail in section 4.1.3.4 and section 4.1.3.5. For a given simulation round, the optimization and trading activities of all agents occur sequentially with a randomized agent order. Specifically, the trading process for a simulation round is divided into J segments, where J represents the total number of agents. At a given point in time t, the jth randomly selected trader determines their target portfolio weights based on their stylized agent behavior and issues limit buy or sell orders to obtain their target portfolio composition. For a buy order, the limit buy price for stock i at time t, p_{it}^b, is computed by:

$$p_{it}^{b} = b_{it}\ n_{it}, \tag{4.2}$$

where b_{it} represents the highest bid price for stock i at time t and n_{it} represents a random draw from the Gaussian distribution $N(1.005, 0.005)$. Conversely, the sell price for asset i at time t, p_{it}^{s}, is computed by:

$$p_{it}^{s} = a_{it}/n_{it}, \tag{4.3}$$

where a_{it} denotes the lowest ask price for asset i at time t and n_{it} represents a random draw from the Gaussian distribution $N(1.005, 0.005)$. This means that agents who want to buy an asset marginally overbid the current best bid price on average and agents who seek to sell an asset tend to marginally underbid the current best ask price in the market, which is similar to the mechanism designs in Raberto et al. (2003), Raberto and Cincotti (2005), and Ponta et al. (2011). Therefore, buy (sell) orders with prices higher (lower) than the best ask (bid) price for an asset are equivalent to market orders, given a sufficient market depth.

4.1.3.4 Active Investors

Our active investors estimate the fundamental values for the individual stocks and optimize their individual portfolio weights based on these estimates. Mathematically, at time t, agent j estimates the end-of-round fundamental value of asset i, $p_{it}^{true\ eor}$ with:

$$\hat{p}_{ijt}^{eor} = p_{it}^{true\ eor} + n_{it}^{fc}, \tag{4.4}$$

where n_{it}^{fc} denotes the forecasting error, determined by a random draw from the Gaussian distribution $N(0, \sigma_{it}^{fc})$. Since there is limited evidence about target price forecasting errors of financial analysts in academic research (Bonini et al., 2010) and forecasting errors can vary strongly for different markets and countries (Bilinski et al., 2013), we run the simulations with different values for σ_{it}^{fc}. Bilinski et al. (2013) evaluate the 12-month target price forecasting errors of financial analysts, $aTPE$, in different countries. We use the average analyst target price accuracy identified in Bilinski et al. (2013) to calculate the starting value for σ_{it}^{fc} and scale it to a 3-month horizon under the assumption of unbiased and normally distributed forecast errors and a Brownian motion forecast error development over time (see Appendix C.3). This yields a starting value of 0.45 for $aTPE$ and equivalently $0.2820\,p_{it}^{market}$ for σ_{it}^{fc}. We consider this our baseline setting, but as it is based on a number of assumptions, we run our simulation with different values for the individual target price forecasting errors to ensure the validity of our results in different settings. This results in a final value set for $aTPE$ of $aTPE \in \{0.2250, 0.4500, 0.9000\}$ and equivalently a final value set for σ_{it}^{fc} of $\sigma_{it}^{fc} \in \{0.1410\,p_{it}^{market}, 0.2820\,p_{it}^{market}, 0.5640\,p_{it}^{market}\}$.

Utilizing their price estimate $\hat{p}_{ijt}^{eor}$ for risky asset i for the end of the round, $\hat{p}_{ij}^{eor}$, agent j estimates the end-of-round return for asset i at time t, $\hat{r}_{ijt}^{eor}$, as follows:

$$\hat{r}_{ijt}^{eor} = \frac{\hat{p}_{ijt}^{eor}}{p_{it}^{market}} - 1, \tag{4.5}$$

where p_{it} denotes the market price of asset i at time t.

Based on Equation 4.5 agent j generates return estimates for all risky assets, which, combined with the risk-free return of zero, yield return estimate vector $\hat{\mathbf{r}}_{jt} = \{\hat{r}_{ijt}^{eor}, i = 1, ..., I+1\}$. At time t, the fundamental agent j selects their desired portfolio weights $\hat{\mathbf{w}}_{jt} = \{\hat{w}_{ijt}, i = 1, ..., I+1\}$ by maximizing their utility based on Equation 4.1. Thereby, the estimated portfolio return, $\hat{r}_{jt}$, is given by

$$\hat{r}_{jt} = \hat{\mathbf{w}}_{jt}^{\intercal}\ \hat{\mathbf{r}}_{jt}, \tag{4.6}$$

with

$$\hat{w}_{ijt} >= w_{min}\ \forall\ i, j, t \tag{4.7}$$

and

$$\hat{w}_{ijt} <= w_{max}\ \forall\ i, j, t, \tag{4.8}$$

where w_{min} denotes the minimum individual portfolio weight, and w_{max} denotes the maximum individual portfolio weight. For these portfolio optimization tasks, we utilize the *cvxpy* package(Diamond and Boyd, 2016) in combination with the *ecos* solver(Domahidi et al., 2013). Furthermore, the estimated portfolio variance of agent j at time t, $\hat{\sigma}_{jt}^2$, is given by

$$\hat{\sigma}_{jt}^2 = \hat{\mathbf{w}}_{jt}^{\intercal}\ \hat{\mathbf{V}}_t\ \hat{\mathbf{w}}_{jt}, \tag{4.9}$$

where $\hat{\sigma}^2_{jt}$ denotes the estimated portfolio variance of agent j, $\hat{\mathbf{V}}_t$ denotes the estimated variance-covariance matrix $((I+1)\times(I+1))$ of all asset returns. All agents estimate the variance-covariance matrix at a specific time from the 40 most recent asset returns. The variance of the risk-free asset and its covariance with other assets is zero per definition. We focus on active agent's forecasts of prices and, related to that, returns, as returns are the critical input parameter in mean-variance portfolio optimisation (Best and Grauer, 1991). We choose the simplification of a joint variance-covariance matrix, as, similar to individual target price forecasting errors, modeling individual errors would be subjected to several assumptions. As this assumptions already constitute Chopra and Ziemba (1993) compare the importance of different input parameters on mean-variance portfolio optimization and show that returns have about 11 times as much influence on the portfolio selection as portfolio variances. Further, we restrict the individual portfolio weights to be non-negative with an upper bound of $w_{max} \in \{0.33, 0.5, 1\}$, where $w_{max} = 0.5$ constitutes our base case. We define these portfolio weight restrictions to increase the robustness of portfolio optimization. In reality, short-selling usually comes with high costs and certain types of investors are restricted to participate in short-selling. Imposing short sale constraints is equivalent to shrinking larger elements of the covariance matrix towards zero (Jagannathan and Ma, 2003). Jagannathan and Ma (2003) argue that the most extreme covariance estimates are likely to be caused by downward-biased or upward-biased estimation errors. Therefore, this shrinking may reduce the overall estimation error. Furthermore, mean-variance optimization tends to lead to extreme portfolio weight results and the introduction of an upper weight bound can ensure a certain level of portfolio diversification and lower the risk of extreme events (Eichhorn et al., 1998; Grauer and Shen, 2000; Jagannathan and Ma, 2003). Finally, the difference between the desired portfolio weights $\hat{\mathbf{w}}_{jt}$ and actual portfolio weights $\mathbf{w}_{jt}$ determines the order vector $\Delta\mathbf{w}_{jt}$ for agent j at time t:

$$\Delta\mathbf{w}_{jt} = \hat{\mathbf{w}}_{jt} - \mathbf{w}_{jt}. \tag{4.10}$$

After determining their order vector, agent j issues orders with the corresponding order quantities, where we round the exact number of shares to integer values.

4.1.3.5 Passive Investors

The passive investors do not estimate stock values of individual securities but instead combine the market with the risk-free asset to form their overall portfolio. All risky assets are weighted based on their market capitalization within the market portfolio. Thus, passive investors face a more simple portfolio optimization problem than active investors, as they only determine which fraction of their wealth they invest in the stock market and do not actively select the individual portfolio weights of the risky assets. Specifically, at time t, passive agent j selects the portfolio weights $\hat{\mathbf{w}}_{jt}^{passive} = \{\hat{w}_{jt}^{market}, \hat{w}_{jt}^{cash}\}$ that maximize the utility function specified in Equation 4.1. The estimated portfolio return, $\hat{r}_{jt}$, is given by:

$$\hat{r}_{jt} = \hat{\mathbf{w}}_{jt}^{passive\ \intercal}\ \hat{\mathbf{r}}_{jt}^{passive}, \tag{4.11}$$

with

$$\hat{w}_{jt}^{market} \geq w_{min}\ \forall\ j, t \tag{4.12}$$

and

$$\hat{w}_{jt}^{cash} \geq w_{min}\ \forall\ j, t, \tag{4.13}$$

where w_{min} denotes the minimum portfolio weight and $\hat{\mathbf{r}}_{jt}^{passive} = \{\hat{r}_t^{market}, 0\}$ is a vector of length 2 that includes the expected market return of passive investors, the average return of the market portfolio over the last 40 observations, and the expected return of cash, zero. The estimated portfolio variance $\hat{\sigma}_{jt}^2$ is given by:

$$\hat{\sigma}_{jt}^2 = \hat{\mathbf{w}}_{jt}^{\intercal\ passive}\ \hat{\mathbf{V}}_t^{passive}\ \hat{\mathbf{w}}_{jt}^{passive}, \tag{4.14}$$

where $\hat{\mathbf{V}}_t^{passive}$ denotes the estimated return variance-covariance matrix (2×2) at time t, whereby the market return variance is estimated over the last 40 simulation rounds and the estimated covariance between the market portfolio return and cash, as well as the variance of the risk-free return, is zero. Corresponding to Equation 4.10, the passive agent j issues orders with trading quantities based on the difference between their desired portfolio weights and their actual portfolio weights. Again, the exact number of shares is rounded to integer values.

4.1.3.6 Random Investors

Our third stylized investor group consists of agents that have a completely randomized behavior. These random behaviors can stabilize the trading system of a simulated market and are often described in the academic literature as a *thermal bath* to evaluate other stylized trading behavior (Raberto et al., 2003; Cincotti et al., 2003; Cocco et al., 2017). Researchers have been able to reproduce various stylized facts of real-world financial markets in simulated markets consisting of only random investors (Mathieu and Brandouy, 2010; Ponta et al., 2011). In our study, random investors may denote a multitude of different investor types who have in common that their aggregated trading behavior is not related to fundamental asset values and, hence, does not have a systematic directed impact on asset prices. Same as the other types of investors, random investors rebalance their portfolio once per simulation round. For the random agents, we loosely follow the agent design of Ponta et al. (2011). At the time t, the desired risky portfolio weight for random agent j is drawn from a uniform distribution between zero and one.

The buy and sell prices of these orders are given by Equations 4.2 and 4.3. The desired market weight of random agent j at time t is given by:

$$\hat{w}_{jt}^{market} = u_{jt}, \tag{4.15}$$

where u_{jt} denotes a random draw from the continuous uniform distribution $U(0, 1)$. Correspondingly, the desired weight of the portfolio cash fraction of random agent j at time t is given by:

$$\hat{w}_{jt}^{cash} = 1 - \hat{w}_{jt}^{market}. \tag{4.16}$$

Furthermore, the desired weights of each individual risky asset within the risky portion of the portfolio are each drawn from the continuous uniform distribution $U(0, 1)$. These draws are normalized to sum up to 1 and subsequently scaled by the desired risky portfolio portion $\hat{w}_{jt}^{market}$. Combined with the desired cash weight, this gives the desired portfolio weight vector of random agent j at time t, $\hat{\mathbf{w}}_{jt}^{random}$. Parallel to the other stylized investors, random agent j issues orders based on the difference between actual and desired portfolio weights (see Equation 4.10), rounding the exact number of shares to integer values.

4.1.3.7 Parameter Overview and Sensitivity Analysis

Table 4.1 gives an overview over our parameter choices in the standard market setting. However, as mentioned above, we aim to understand the impact of major market parameters on market results and hence repeatedly run the simulation on all random seeds with systematically altered simulation parameters. Concretely, we repeat the simulation with the following, varying parameter specifications:

- Fraction of random agents relative to all market participants of *40%* and *90%*
- Individual active investors' absolute target price forecasting errors (aTPE) of *0.225* and *0.9*
- Individual risk aversion factors fixed at *2* and *6*
- Transaction costs of *0.5%*
- Active investors' upper portfolio weight constraints of *0.33* and *1*.

Table 4.1.: Parameter overview of the standard market setting

Parameter	Notation	Value
Total # of Risky Assets	I	4
Risk-Free Rate	r_{rf}	0
Initial Asset Price		100$
Gamma Lower Bound	γ_{min}	2
Gamma Upper Bound	γ_{max}	6
Initial Wealth of Agents		100,000$
Portfolio Constraint Lower Bound	w_{min}	0
Portfolio Constraint Upper Bound	w_{max}	0.5
Absolute Target Price Forecasting Error	$aTPE$	0.45
Transaction Cost		0
# Simulation Rounds		100
Total # of Agents	J	500
Fraction of Random Agents		0.8

4.1.3.8 Evaluation

To assess our simulated market's quality and create comparability to a real-world stock market, we use different established measures, i.e., price deviation, trading volume, market depth, and quoted spread. These measures are calculated for each market configuration by aggregating the results over all assets, simulation rounds, and random seeds. Additionally, we test whether selected stylized facts of real-world financial time series are reproduced in our baseline simulated market.

4.1.3.9 Fundamental Market Efficiency

We measure fundamental market efficiency by calculating the mean absolute deviation between the market price of the simulation, p_{it}^{market} from the fundamental market prices, p_{it}^{true}, at time t for each asset i:

$$PriceDevAbs_t = \frac{1}{I}\sum_{i=1}^{I}\left|p_{it}^{market} - p_{it}^{true}\right|. \tag{4.17}$$

To receive relative price deviation as a percentage value, we divide the absolute value in the above formula by p_{it}^{true} and multiply by 100:

$$PriceDevRel_t = 100 * \frac{1}{I}\sum_{i=1}^{I}\frac{\left|p_{it}^{market} - p_{it}^{true}\right|}{p_{it}^{true}}. \tag{4.18}$$

4.1.3.10 Trading Volume

Another measure we use is the trading volume, which describes the number of shares traded in each round multiplied by their corresponding prices. Trading volume generally indicates market activity and constitutes a basis for many liquidity measures of financial markets (Sarr and Lybek, 2002).

4.1.3.11 Stylized Facts of Real-World Financial Markets

To further evaluate our simulation model, we investigate if our time series of returns follow the same statistical patterns found in many financial time series, so-called stylized facts of financial markets. We check whether the resulting market return series of our simulated market elicits these stylized facts to evaluate the external validity of our results. Cont (2001) list 11 different stylized facts that have been observed and studied repeatedly over the last decades. While these statistical properties are typical for real-world data, Cont (2001) also describe that it is very challenging to create a synthetic market model that can reproduce all stylized facts. Furthermore, since stylized facts generalize and simplify, they are a more qualitative measure by nature. This qualitative character of many stylized facts impedes the comparability of different market scenarios based on these properties. Therefore, we focus on the following selected stylized facts, as they allow for a certain level of quantitative analysis. Since the stylized facts are usually found in return time series with a higher data frequency (Cont, 2001), we further split our simulation rounds into 90 equal proportions, which then results in a resolution of daily returns. We test for the existence of each stylized fact on every random seed and every trader composition in the baseline market setting.

- Heavy Tails: Real-world financial return time series tend to be heavy tailed and non-Gaussian (Cont, 2001; Bradley and Taqqu, 2003). These series often appear to exhibit a power-law tail with a tail index between 2 and 5 (Cont, 2001; Katahira et al., 2019). To test whether a return series follows a normal distribution, we calculate its excess kurtosis and conduct a Kolmogorov-Smirnov test (Kolmogorov, 1933; Smirnov, 1948; Massey, 1951) to show that it is non-Gaussian and heavy tailed. Following Katahira et al. (2019), we also calculate the alphas (i.e., tail-indices) of the heavy tail power-law distributions and calculate the log-likelihood ratio (Wilks, 1938) between a power-law and an exponential distribution given our observed return distribution sample. In total, we calculate the excess kurtosis and p-values of the Kolmogorov-Smirnov test, the power-law alpha, as well as log-likelihood ratios including corresponding p-values for a goodness of fit comparison between a power-law and an exponential distribution.

- Conditional Heavy Tails: Real-world returns also often show so called conditional heavy tails, these are given when a return series is corrected for volatility clustering and still shows heavy tails (Cont, 2001). However, the excess kurtosis of the corrected returns is smaller than the excess kurtosis for the unconditional return distribution. Therefore, we calculate the excess kurtosis of the residuals gained from a GARCH (Bollerslev, 1986) model trained to account for volatility clustering. Again, we run the Kolmogorov-Smirnov test to calculate each corresponding p-value. We consider the stylized fact as fulfilled if the calculated excess kurtosis is positive but lower than the kurtosis calculated for the unconditional heavy tails and if the Kolmogorov-Smirnov test shows statistical significance at the 95% confidence level.

- Gain Loss Asymmetry: Gain loss asymmetry is the difference in the upward and downward movement of returns. In real-world financial markets, prices tend to fall faster than they rise Cont (2001). To test for the fact in a quantifiable way we mainly follow the procedure of Jensen et al. (2003). We set a positive and a negative return level of 10% and minus 10% respectively and then count the time steps needed (i.e., the investment horizons) until the asset reaches this return level. This is done for each asset and at each time step. We then calculate the two density functions of the negative and positive investment horizons and determine their maxima to compare whether, in general, the positive or the negative return level is reached more quickly. We consider the stylized fact as fulfilled if the maximum of the density function for the negative return level lies before the positive return level.

4.1.4 Results

In this section, we present the results of the different simulation runs described in Section 4.1.3. Concretely, we analyze the effect of different market settings and parameters on our key metric fundamental market efficiency and on market activity. Furthermore, we evaluate our market model based on the quantifiable stylized facts described in Section 4.1.3.11.

4.1.4.1 Standard Market Setting

Figure 4.1 shows the relative fundamental price deviation for varying fractions of active and passive investors over all random seeds for our standard scenario (i.e., 80% random investors). We find a lower relative fundamental price deviation for a higher fraction of active investors. Specifically, market prices on average deviate by 55.17% from fundamental prices in the market setting without active investors and 100 (i.e., 20%) passive investors and by 25.86% in the market setting with 100 (i.e., 20%) active investors and zero passive investors. On average, the substitution of a passive investor with an active investor increases fundamental market efficiency, as the fitted cubic function is monotonously decreasing between $x = 0$ and $x = 1$. However, the marginal increase in fundamental market efficiency for this substitution is lower for higher levels of active investment in the market, as the second-order derivative of the function is strictly positive in the area under consideration. Furthermore, as detailed in Table 4.2, the mean trading volume increases for a larger fraction of active investors, nearly doubling from the setting without active investors and the setting with 100 (i.e., 20%) active investors.

Table 4.2.: Fundamental price deviation and trading volume for different investor compositions in the standard setting averaged over all random seeds

# of Agents Active/Passive/Random	Mean Price Deviation Relative [%]	Mean Trading Volume [$]
0/100/400	55.17	2,561,184
5/95/400	49.25	2,551,623
10/90/400	49.04	2,868,562
15/85/400	46.14	2,803,487
20/80/400	40.87	2,880,618
25/75/400	40.71	2,969,273
30/70/400	39.42	3,199,982
35/65/400	39.18	3,379,545
40/60/400	35.35	3,293,083
45/55/400	35.00	3,549,615
50/50/400	33.53	3,548,018
55/45/400	32.35	3,734,339
60/40/400	29.71	3,832,248
65/35/400	31.40	4,014,977
70/30/400	29.38	4,020,009
75/25/400	27.86	4,094,883
80/20/400	27.28	4,187,787
85/15/400	26.15	4,206,388
90/10/400	25.25	4,196,554
95/5/400	26.26	4,591,515
100/0/400	25.86	4,528,378

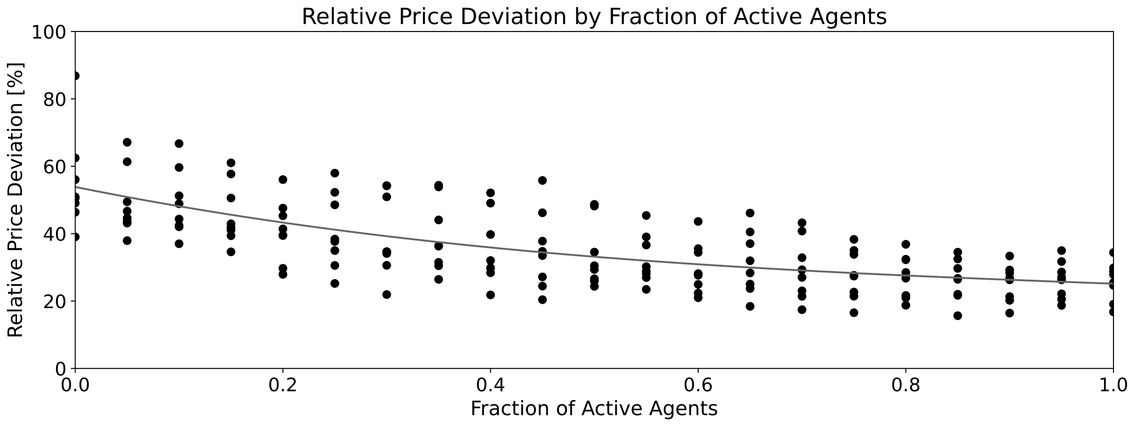


Figure 4.1.: Fundamental price deviation by fraction of active investors (relative to all active and passive investors) detailed over all random seeds in the standard setting

4.1.4.2 Sensitivity to Different Levels of Random Investment

Table 4.3 shows the relative fundamental price deviation and mean trading volume for different fractions of random agents of the total number of market participants (i.e., 40% and 90%). For each scenario, we run the simulation on multiple random seeds, repeatedly altering the combinations of active and passive investors by five percentage points. We find that a lower portion of random agents, which corresponds to a higher combined fraction of passive and active investors, leads to a higher discrepancy between the market composition without active investors and compositions with a high share of active investors. Concretely, in the scenario with 40% random agents, the mean relative fundamental price deviation starts at 74.42% with no active investors in the market. It decreases to 19.10% for the setting with 300 (i.e., 60%) active investors. In the scenario with a share of 90% random agents, the mean relative fundamental price deviation starts at only 47.64% in the setting where all non-random investors are passive investors. However, it decreases to just 31.53 for the market composition in which all remaining investors are active investors.

Table C.1 and Figure C.2 present the corresponding results for a market composition with only 20% random investors. These results allow us to further examine the effects of passive investment on fundamental market efficiency. For instance, the average relative price deviation from fundamental prices is 65.24% in the market with 100 active, 300 passive, and 100 random investors and amounts to 25.86%

in the market with 100 active, zero passive, and 400 random investors (see Table 4.2). Combined with the in Section 4.1.4.1 presented findings, this indicates that a large fraction of passive investors, contrary to a large fraction of active and random investors, impairs fundamental market efficiency. While it is clear that the trading behavior of active investors links market prices to fundamental values, it is noteworthy that this link is distinctly stronger in market environments with a lower amount of passive investment. This finding may be driven by the similar trading patterns and generally lower trading activity within the group of passive investors. Regarding the latter, Table 4.4 shows that the average absolute weight change of all risky assets per simulation round is distinctly higher for active investors than for passive investors.

Table 4.3.: Fundamental price deviation and trading volume for different investor compositions in the market settings with a share of 40% and 90% random investment averaged over all random seeds

# of Agents Active/Passive/Random (40% Random Investment)	Mean Price Deviation Relative [%]	Mean Trading Volume [$]	# of Agents Active/Passive/Random (90% Random Investment)	Mean Price Deviation Relative [%]	Mean Trading Volume [$]
0/300/200	74.42	1,104,871	0/50/450	47.64	3,033,663
15/285/200	128.99	1,512,533	2/47/450	45.72	2,968,380
30/270/200	57.40	1,670,072	5/45/450	44.38	3,061,623
45/255/200	52.29	1,999,648	7/42/450	44.89	3,396,441
60/240/200	57.53	2,284,938	10/40/450	43.70	3,297,070
75/225/200	100.57	2,783,213	12/37/450	40.82	3,131,807
90/210/200	106.12	3,221,095	15/35/450	40.76	3,298,982
105/195/200	51.26	3,381,901	17/32/450	40.31	3,305,112
120/180/200	92.91	4,002,829	20/30/450	38.28	3,387,004
135/165/200	47.03	4,106,699	22/27/450	38.33	3,358,055
150/150/200	56.59	4,567,460	25/25/450	37.95	3,366,321
165/135/200	47.00	4,928,309	27/22/450	37.49	3,628,239
180/120/200	36.70	5,233,076	30/20/450	36.73	3,596,709
195/105/200	42.26	5,785,707	32/17/450	35.87	3,608,822
210/90/200	32.63	6,077,184	35/15/450	34.47	3,615,557
225/75/200	30.53	6,516,738	37/12/450	34.39	3,700,074
240/60/200	26.17	6,878,131	40/10/450	33.11	3,688,612
255/45/200	22.99	7,259,559	42/7/450	33.58	3,764,347
270/30/200	22.04	7,782,640	45/5/450	32.75	3,727,598
285/15/200	20.02	8,183,754	47/2/450	33.96	3,973,665
300/0/200	19.10	8,663,756	50/0/450	31.53	3,882,438

Additionally, large fractions of passive investment may lead to price bubbles and market failure. Table 4.3 shows that some compositions that have a high share of passive investment exhibit a particularly high price deviation from fundamental values. This is driven by specific simulation runs, in which market values deviate strongly from fundamental values. Figure 4.2, which details the results over the different random seeds, shows that these large price bubbles do not occur in all market settings with a high fraction of passive investment. However, these settings enable price bubbles, as we do not observe bubbles of similar magnitude in settings with a generally lower level of passive investment (see Figure 4.3). Figure C.1 illustrates the mechanism of a price bubble at the example of the simulation run for random seed 1 and a market composition of 90 active, 210 passive, and 200 random investors. After a series of high market returns, passive investors gradually increase their risky portfolio fractions. Simultaneously, active investors realize that the market becomes overpriced and sell as many units of the risky assets as they can, given their portfolio constraints. However, due to the large share of passive investors, in these scenarios active investors cannot correct for the overpricing and the link between market prices and fundamental values breaks down.

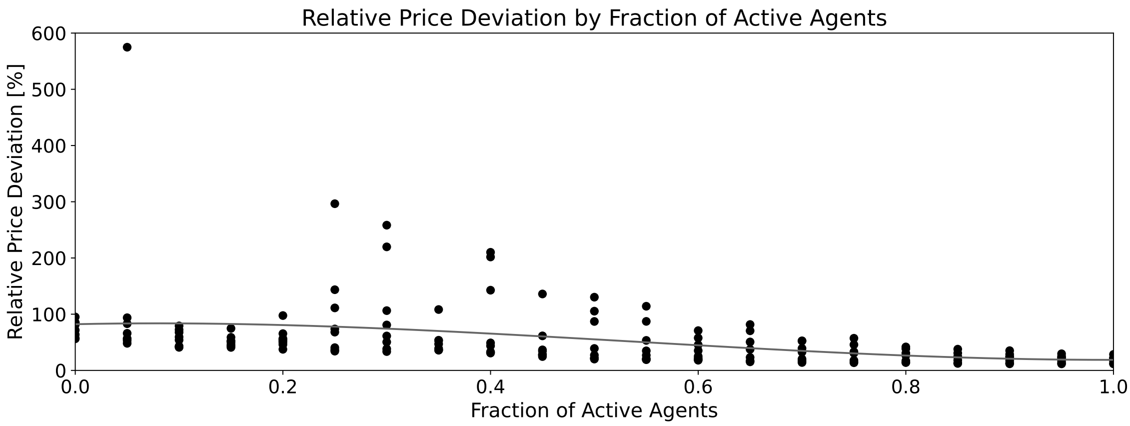


Figure 4.2.: Relative fundamental price deviation by fraction of active investors (relative to all active and passive investors) detailed over all random seeds in the market setting with 40% random investment

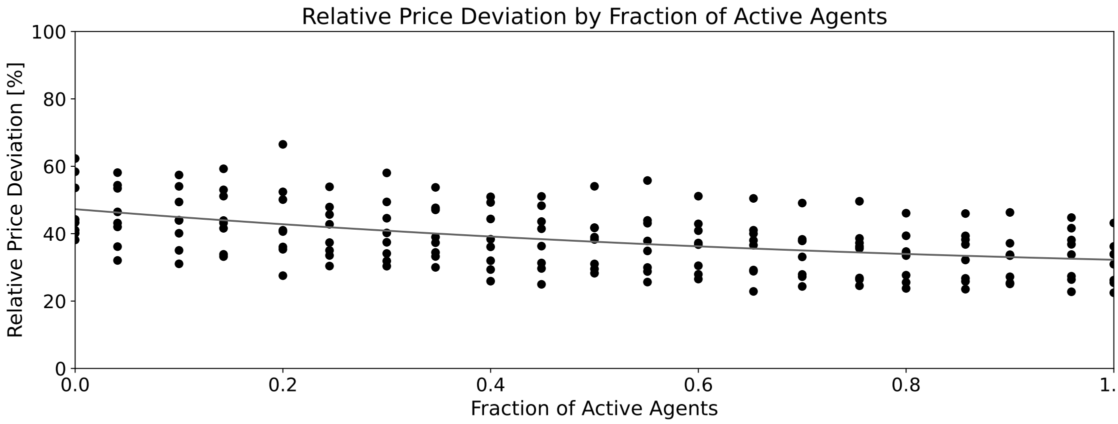


Figure 4.3.: Relative fundamental price deviation by fraction of active investors (relative to all active and passive investors) detailed over all random seeds in the market setting with 90% random investment

Table 4.4.: Mean absolute changes of portfolio weights between simulation rounds by agent type and agent composition

# of Agents Active/Passive/Random (40% Random Investment)	Mean Absolute Changes of Portfolio Weights by Agent Type		
	Active	Passive	Random
0/300/200	-	0.03	0.33
15/285/200	0.47	0.04	0.33
30/270/200	0.52	0.04	0.34
45/255/200	0.57	0.04	0.34
60/240/200	0.59	0.05	0.34
75/225/200	0.62	0.05	0.34
90/210/200	0.62	0.06	0.34
105/195/200	0.65	0.06	0.34
120/180/200	0.66	0.07	0.34
135/165/200	0.67	0.07	0.35
150/150/200	0.67	0.07	0.35
165/135/200	0.68	0.08	0.35
180/120/200	0.69	0.08	0.35
195/105/200	0.69	0.08	0.35
210/90/200	0.7	0.08	0.35
225/75/200	0.7	0.08	0.35
240/60/200	0.71	0.08	0.35
255/45/200	0.72	0.08	0.35
270/30/200	0.73	0.08	0.35
285/15/200	0.73	0.08	0.35
300/0/200	0.73	-	0.35

# of Agents Active/Passive/Random (90% Random Investment)	Mean Absolute Changes of Portfolio Weights by Agent Type		
	Active	Passive	Random
0/50/450	-	0.07	0.34
2/47/450	0.42	0.07	0.34
5/45/450	0.44	0.07	0.34
7/42/450	0.44	0.08	0.34
10/40/450	0.46	0.07	0.34
12/37/450	0.47	0.08	0.34
15/35/450	0.47	0.08	0.34
17/32/450	0.48	0.08	0.34
20/30/450	0.49	0.08	0.34
22/27/450	0.5	0.08	0.34
25/25/450	0.5	0.08	0.34
27/22/450	0.52	0.08	0.34
30/20/450	0.52	0.09	0.34
32/17/450	0.53	0.08	0.34
35/15/450	0.53	0.09	0.35
37/12/450	0.53	0.09	0.35
40/10/450	0.54	0.09	0.35
42/7/450	0.54	0.08	0.35
45/5/450	0.55	0.09	0.35
47/2/450	0.55	0.09	0.35
50/0/450	0.56	-	0.35

4.1.4.3 Sensitivity to Different Individual Target Price Forecasting Errors of Active Investors

Table 4.5 und Figure 4.4 present the effect of varying individual target price forecasting errors of active investors on fundamental market efficiency. Generally, for market compositions with active investors, we find that lower individual errors lead to a higher level of fundamental market efficiency and vice versa. In our setting, a bisection of individual target price forecasting errors compared to the standard market setting reduces the mean fundamental price deviation by up to 4.56 percentage points, which occurs for the composition of 100 active, zero passive, and 400 random investors. Doubling the individual target price forecasting errors results in an average increase in fundamental price deviation of up to 8.00 percentage points, occurring for the market composition of 20 active, 80 passive, and 400 random investors. Furthermore, we find that a lower (higher) individual active forecasting error results in a lower (higher) overall trading volume. For lower (higher) individual forecasting errors, the estimated fundamental prices of active investors become more (less) similar and it is less (more) likely for active investors to trade with each other and more (less) likely to compete for buying or selling the same stocks.

Table 4.5.: Relative fundamental price deviation for different investor compositions in the market settings with active investors' individual absolute target price forecasting errors (aTPE) of 0.225 and 0.9 averaged over all random seeds

# of Agents Active/Passive/Random	Mean Price Deviation Relative [%]		Mean Trading Volume [$]	
	$aTPE = 0.225$	$aTPE = 0.9$	$aTPE = 0.225$	$aTPE = 0.9$
0/100/400	55.17	55.17	2,561,184	2,561,184
5/95/400	48.84	52.19	2,681,653	2,742,627
10/90/400	44.84	49.65	2,662,429	2,838,584
15/85/400	45.02	48.27	2,940,474	3,045,832
20/80/400	41.49	48.87	2,775,608	3,016,024
25/75/400	37.56	43.57	2,965,971	3,160,534
30/70/400	35.65	44.18	2,930,773	3,290,539
35/65/400	35.29	39.08	3,228,643	3,244,884
40/60/400	32.30	43.01	3,206,435	3,571,412
45/55/400	32.79	40.74	3,337,158	3,685,447
50/50/400	31.04	38.23	3,401,701	3,710,353
55/45/400	28.78	38.51	3,440,410	3,906,162
60/40/400	25.66	36.34	3,580,685	3,911,939
65/35/400	25.51	34.47	3,527,615	4,109,141
70/30/400	26.12	34.57	3,706,781	4,197,060
75/25/400	24.13	33.27	3,899,526	4,295,706
80/20/400	23.92	32.83	3,941,963	4,418,396
85/15/400	23.69	32.18	3,967,384	4,506,681
90/10/400	22.67	32.02	4,120,524	4,635,829
95/5/400	22.93	30.35	4,199,213	4,747,823
100/0/400	21.30	31.66	4,291,552	4,832,332

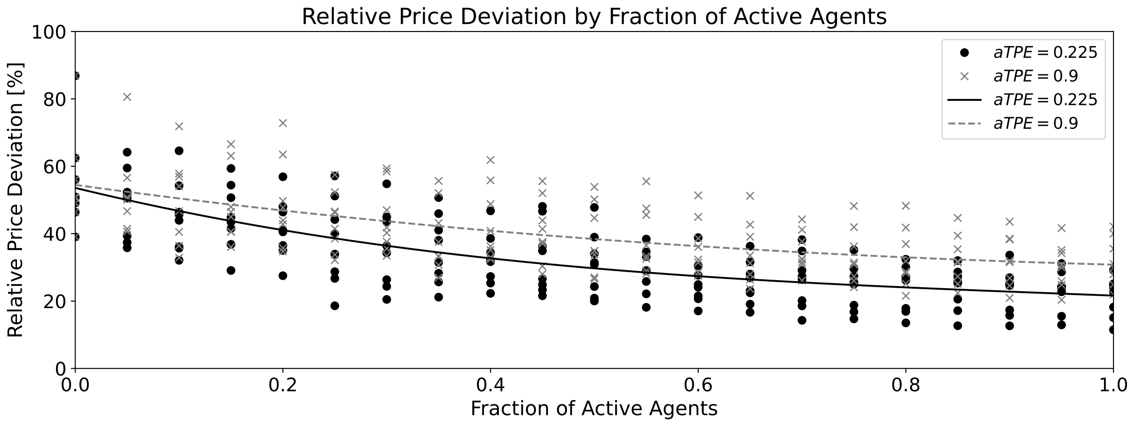


Figure 4.4.: Fundamental price deviation by fraction of active investors (relative to all active and passive investors) detailed over all random seeds in the market settings with active investors' individual absolute target price forecasting errors (aTPE) of 0.225 and 0.9

4.1.4.4 Sensitivity to Changes in Individual Risk Aversion

To analyze the influence of individual risk aversion on market outcomes, we formulate two additional scenarios in which the risk aversion factor γ is fixed at values of 2 and 6, respectively. Table 4.6 and Figure 4.5 show that changes in individual risk aversion only have a minor impact in markets with a higher share of active investment. In market settings with a low share of active investment, we observe that lower individual risk aversion increases mean fundamental price deviation. This effect might be explained by the fact that lower risk aversion leads to more extreme portfolio weights, which, in turn, may drive prices faster away from fundamental values in markets that only have a weak link between fundamentals and market prices (i.e., low share of active investors).

Table 4.6.: Relative fundamental price deviation for different investor compositions in the market settings with individual investors' risk aversion of $\gamma = 2$ and $\gamma = 6$ averaged over all random seeds

# of Agents Active/Passive/Random	Mean Price Deviation Relative [%]		Mean Trading Volume [$]	
	$\gamma = 2$	$\gamma = 6$	$\gamma = 2$	$\gamma = 6$
0/100/400	61.12	46.17	2,675,948	2,554,114
5/95/400	56.60	48.74	2,699,050	2,593,278
10/90/400	51.49	44.07	2,725,954	2,770,256
15/85/400	48.91	43.45	2,892,430	2,844,814
20/80/400	46.99	41.01	3,042,139	2,794,737
25/75/400	43.60	37.94	3,111,944	3,011,106
30/70/400	40.31	37.40	3,113,275	3,076,410
35/65/400	39.31	35.77	3,168,636	3,119,941
40/60/400	35.21	36.07	3,297,038	3,321,216
45/55/400	36.65	35.24	3,519,634	3,466,496
50/50/400	33.80	33.18	3,453,610	3,362,015
55/45/400	33.03	30.62	3,788,077	3,694,802
60/40/400	31.73	30.74	3,888,638	3,819,000
65/35/400	30.10	28.49	3,917,745	3,800,704
70/30/400	28.92	28.25	3,960,200	3,913,296
75/25/400	28.32	27.71	4,187,611	3,883,864
80/20/400	28.17	26.23	4,216,768	4,138,441
85/15/400	27.46	26.38	4,300,281	4,246,452
90/10/400	27.13	25.69	4,743,153	4,278,943
95/5/400	26.06	25.01	4,450,700	4,438,806
100/0/400	25.21	24.56	4,519,973	4,615,691

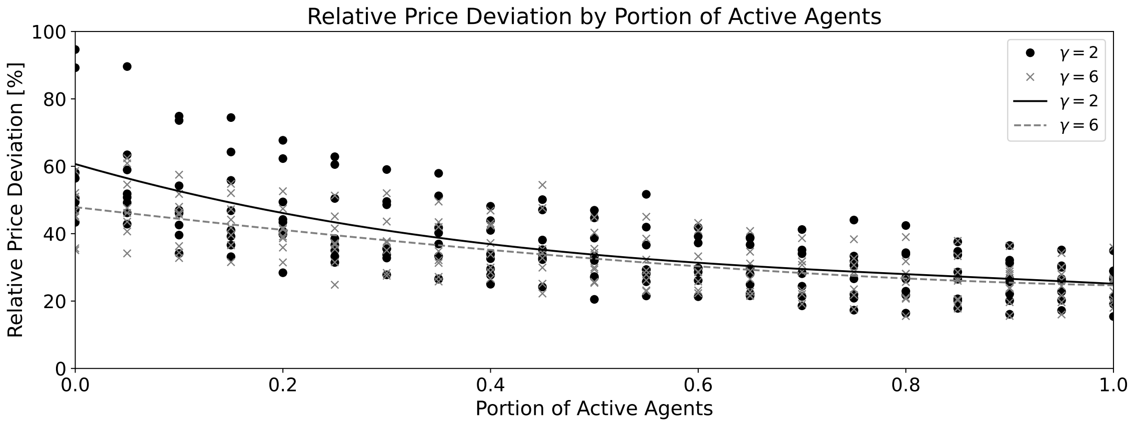


Figure 4.5.: Fundamental price deviation by fraction of active investors (relative to all active and passive investors) detailed over all random seeds in the market setting with individual investors' risk aversion of $\gamma = 2$ and $\gamma = 6$

4.1.4.5 Sensitivity to Different Transaction Costs

Table 4.7 presents the results of a simulation scenario of a market that features relative trading costs of 0.5%. Introducing transaction costs reduces fundamental market efficiency in the market settings with a higher share of active investors. Specifically, the mean fundamental market deviation increases for all tested agent compositions with more than 50 active agents. Furthermore, it only decreases to 28.21% for the composition with 100 active, zero passive, and 400 random investors, compared to 25.86% in the market setting without transaction costs (see Table 4.2). Moreover, an increase in transaction costs leads to lower trading volumes, as investors incorporate the expected trading costs into their utility functions and prefer smaller portfolio changes.

Table 4.7.: Fundamental price deviation and trading volume for different investor compositions in the market setting with 0.5% transaction cost averaged over all random seeds

# of Agents Active/Passive/Random	Mean Price Deviation Relative [%]	Mean Trading Volume [$]
0/100/400	51.20	2,485,096
5/95/400	49.78	2,603,984
10/90/400	44.84	2,397,038
15/85/400	41.50	2,625,995
20/80/400	40.66	2,650,902
25/75/400	39.27	2,699,538
30/70/400	38.24	2,725,631
35/65/400	36.84	2,883,039
40/60/400	38.50	3,112,331
45/55/400	34.40	3,028,697
50/50/400	33.88	3,074,151
55/45/400	34.20	3,078,258
60/40/400	33.74	3,217,903
65/35/400	32.14	3,388,530
70/30/400	31.93	3,438,023
75/25/400	30.86	3,501,592
80/20/400	29.39	3,704,058
85/15/400	29.28	3,718,231
90/10/400	29.07	3,874,499
95/5/400	29.67	3,804,029
100/0/400	28.21	3,950,970

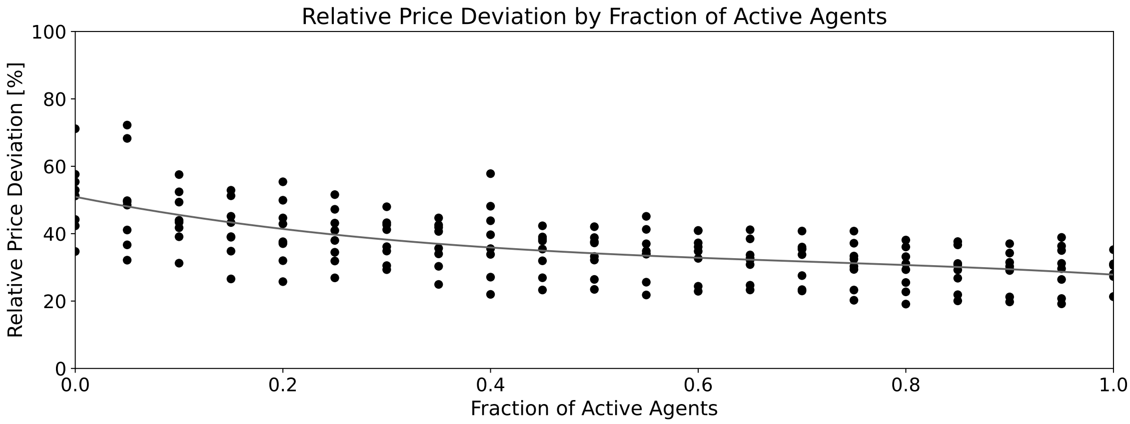


Figure 4.6.: Fundamental price deviation by fraction of active investors (relative to all active and passive investors) detailed over all random seeds in the market setting with 0.5% transaction costs

4.1.4.6 Sensitivity to Portfolio Weight Restraints

Table 4.8 and Figure 4.7 present the results for different upper bounds for the individual portfolio weights of the different assets. We find that less restrictive individual portfolio weights (i.e., $w_{max} = 1$) lead to higher fundamental market efficiency and higher trading activity on average. In contrast, the opposite is true for more restrictive individual portfolio weights (i.e., $w_{max} = 0.33$). In our market setting with 100 active, zero passive, and 400 random investors, the less restrictive upper bound of $w_{max} = 1$ results in a mean fundamental price deviation of 22.03%. For the same market composition, the more restrictive bound of $w_{max} = 0.33$ leads to a mean fundamental price deviation of 30.29%.

Table 4.8.: Fundamental price deviation and trading volume for different investor compositions in the market setting with individual investors' upper portfolio constraints of $w_{max} = 0.33$ and $w_{max} = 1$ averaged over all random seeds

# of Agents Active/Passive/Random	Mean Price Deviation Relative [%]		Mean Trading Volume [$]	
	w_{max}=0.33	w_{max}=1	w_{max}=0.33	w_{max}=1
0/100/400	55.17	55.17	2,561,184	2,561,184
5/95/400	49.19	51.68	2,655,221	2,879,598
10/90/400	50.48	47.32	2,841,004	2,810,632
15/85/400	47.39	43.18	2,867,019	2,979,468
20/80/400	49.01	42.50	3,039,779	3,099,328
25/75/400	46.58	39.10	3,040,331	3,191,590
30/70/400	45.06	37.18	2,893,290	3,340,096
35/65/400	42.13	34.00	3,115,205	3,447,845
40/60/400	42.40	33.87	3,177,080	3,458,834
45/55/400	40.06	32.25	3,275,576	3,771,100
50/50/400	39.14	32.15	3,369,085	3,937,050
55/45/400	38.29	28.73	3,508,492	4,025,272
60/40/400	36.72	27.64	3,422,609	4,231,287
65/35/400	36.09	27.12	3,517,104	4,408,642
70/30/400	34.68	27.20	3,619,060	4,682,958
75/25/400	35.49	24.30	3,553,326	4,810,109
80/20/400	33.79	24.99	3,912,030	4,799,979
85/15/400	32.38	21.49	3,798,371	2,982,970
90/10/400	31.81	22.80	3,909,569	5,153,857
95/5/400	31.68	23.13	3,963,279	5,276,650
100/0/400	30.29	22.03	4,096,592	5,507,077

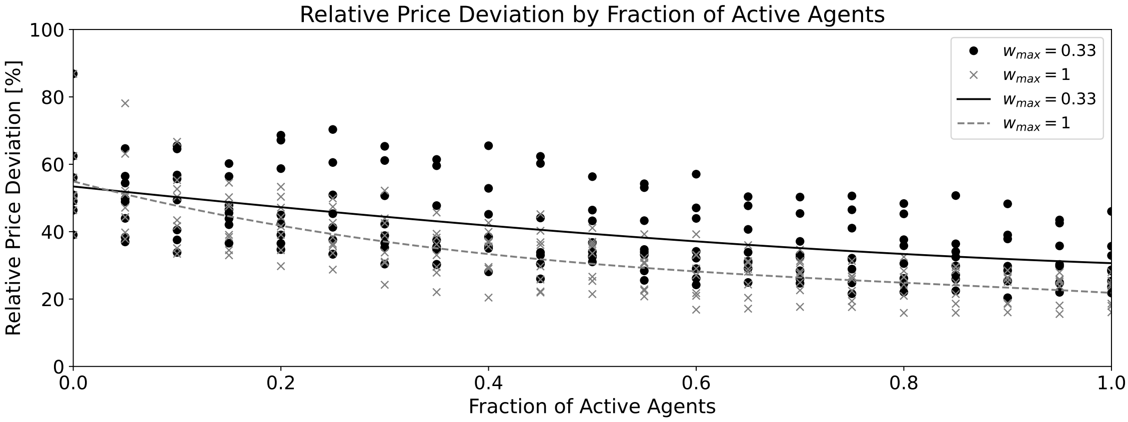


Figure 4.7.: Fundamental price deviation by fraction of active investors (relative to all active and passive investors) detailed over all random seeds in the market setting with individual investors' upper portfolio constraints of 0.33 and 1

4.1.4.7 Replication of Stylized Facts of Real-World Financial Markets

Table C.2 presents the results of our quantitative tests for the replication of stylized facts of financial return time series detailed over all compositions and random seeds in our standard market setting. We find that all return series exhibit an excess kurtosis of at least 2 and the Kolmogorov-Smirnov tests (Kolmogorov, 1933; Smirnov, 1948; Massey, 1951) show that all series are found to be non-Gaussian at a 95% confidence level. The alpha of a fitted power-law distribution is in 71% of all simulated market outcomes in the range between 2 and 5. Furthermore, the log-likelihood tests show that in 81% of market outcomes, the observed distribution rather follows a power-law distribution than an exponential distribution. While these tests are only in 17% of these cases significant at the 95% confidence level, they never significantly indicate that any of the observed return series rather follow an exponential distribution. Additionally, we find that the created return times series exhibit conditional heavy tails in 41% of all market outcomes. The stylized fact gain loss asymmetry is fulfilled in 64% of all simulated market outcomes.

4.1.5 Discussion

This simulation study sheds light on the impact of two essential investor types, namely active and passive investors, on fundamental market efficiency. It becomes apparent that high levels of passive investment within a market may foster price bubbles and impair fundamental market efficiency. This finding has implications for financial regulators, who should incorporate the level of passive investment and existing market frictions when assessing risks to financial market stability. In the light of an increase in passive investment due to the emergence of ETFs and robo-advisory, this issue has become increasingly relevant. However, while it is possible to approximate the level of passive investment through the volume of passively managed funds, it is challenging to differentiate between various types of active investors without conducting concrete interviews.

Additionally, the increase of fundamental market inefficiency for a higher fraction of passive investment is influenced by active investors' trading frictions and portfolio restrictions. Thereby, markets with less restricted active investors may incorporate a higher fraction of passive investors without creating large price bubbles, as less restricted active investors can trade more and correct a higher level of mispricing. This finding emphasizes the need for a careful situational assessment by regulators when imposing measures that constrain trading.

The study further shows that the trading behavior of active investors, who individually have imprecise but unbiased target price forecasts, links the market prices of assets to their fundamental values. The marginal increase in fundamental market efficiency for an additional fundamental investor entering the market decreases for higher levels of existing fundamental investment. However, due to the imprecision of individual forecasts, market prices generally may not exactly reflect fundamental asset values at a certain point in time.

Regarding the second research question addressed in this study, our results show that more accurate target price forecasts, as, for instance, caused by the proliferation of machine learning in the asset management industry, lead to higher fundamental efficiency of market prices. This increase in fundamental market efficiency is more pronounced in markets with a higher level of active investment.

Generally, we find that our results are in line with related work (e.g., (Sullivan and Xiong, 2012; Qin and Singal, 2015; Sushko and Turner, 2018; Anadu et al., 2019)), which finds that a certain share of active investors is necessary for functioning and efficient financial markets. Our simulation setting allows to provide further details on the underlying mechanisms of this issue and evaluates the impact of different market environments.

An explicit limitation of this study is its simulation setting. While we generally select simulation parameters carefully based on empirical evidence, it is unavoidable to make certain assumptions during parameter selection. Furthermore, the simulation setting significantly simplifies real-world financial markets. For instance, while mean-variance utility functions are still commonly used in finance, they may overly simplify the utility functions of investors. Hence, we aim to draw general conclusions about financial markets' functioning rather than narrow down on the exact magnitude of specific numeric results. We aim to reduce this limitation by testing whether the return times series generated by our simulated market exhibit quantifiable stylized facts that are often observed in real-world financial markets. We find that all generated return series exhibit excess kurtoses and further tests suggest that most of the generated time series exhibit heavy tails and a gain-loss asymmetry. However, we only identify the stylized fact of conditional heavy tails in 42% of all generated return series. Overall, these tests show that our simulated market exhibits some but not all properties of real-world financial markets.

As this study focuses on the impact of fundamental and passive investment on fundamental market efficiency, future research could shed further light on the decision of investors to invest based on fundamental analysis or to invest passively. As our study shows, markets become increasingly fundamentally efficient with higher fractions of active investors. However, in markets with higher shares of active investment, the competition to trade mispriced assets also increases, leading to a lower utility of individual active investors. Hence, researchers could investigate the existence of a possible equilibrium level of active and passive investment in a financial market and examine further influencing factors, such as the use of other utility functions that have been shown to resemble the real world more realistically (e.g., prospect theory (Kahneman and Tversky, 1979)).

4.1.6 Conclusion

In this simulation study, we analyze the effects of different levels of active and passive investment on fundamental market efficiency. We examine various market settings to understand the impact of other market parameters on our results. For the selected market settings, we generally find that larger fractions of active investment within a market increase overall fundamental market efficiency. The marginal positive effect on fundamental market efficiency per additional fundamental investor decreases for larger fractions of fundamental investment. In addition, a less restrictive trading environment for active investors in the form of relaxed portfolio constraints or lower trading costs facilitates trading and increases fundamental market efficiency. Furthermore, more accurate price forecasts of individual active investors, caused, for instance, by the proliferation of machine learning approaches within the asset management industry, tend to increase fundamental market efficiency. We also find that large fractions of passive investment within a financial market may lead to market failure and facilitate technical price bubbles.

CHAPTER 5

FINALE

This thesis presents four quantitative studies on financial market prediction in light of three major financial market developments. First, the market predictability of the novel object of transaction cryptocurrency was examined from multiple perspectives. A structured literature review established the state of research in the field of Bitcoin market prediction via machine learning. Based on the identified research gaps, a quantitative study applied various machine learning models to shed light on the short-term predictability of the Bitcoin market. The next study analyzed the potential for statistical arbitrage in the cryptocurrency market by applying machine learning to predict market movements of the 50 largest cryptocurrencies by market capitalization. Next, stock market predictability was investigated in light of the COVID-19 pandemic, which constitutes a significant change in the socio-economic and legal environment. A machine learning study examined the potential of financial market prediction utilizing COVID-19-related data. Last, the implications of the shift from active to passive investment, which constitutes a change in agent behavior and market structure in financial markets, were analyzed. A simulation study analyzed the relationship between the market share of active investors, who aim to predict stock prices, and passive investors on fundamental market efficiency. This final chapter summarizes the contributions of this thesis by answering the raised research questions, highlighting existing limitations of the presented studies, and elaborating on avenues for future research.

5.1 Contributions and Answers to Research Questions

The thesis at hand provides an understanding of financial market prediction in light of three major financial market developments. The concrete developments under consideration cover all pivotal market elements of financial markets (Weinhardt and Gimpel, 2007). This chapter summarizes the answers to the raised research question and distill the main contributions of this thesis.

Research Question 1 *What is the predictive power of machine learning models predicting short-term movements of the Bitcoin market?*

The first research question focuses on the predictability of the short-term Bitcoin market and is addressed in the empirical study presented in Chapter 2.2. In the study, various state-of-the-art machine learning models are trained on a comprehensive feature set to predict binary movements of the Bitcoin market on the grounds of four different prediction horizons ranging from 1 minute to 60 minutes. As the models are trained on a balanced training set, the accuracy measure is utilized for model evaluation.

The results show that the out-of-sample model accuracies range from 50.9% to 56.0%. While all utilized models statistically outperform a random classifier at a significance level of 1%, the employed recurrent neural network ensembles (LSTM, GRU), the tree-based ensemble models (gradient boosting classifier, random forest), and an ensemble model consisting of all considered machine learning models, exhibit higher predictive performance than the other employed methods. A comparison of the four considered prediction horizons reveals that the average and the highest model accuracy grow monotonously with increasing prediction horizons. In examining the economic implications of these predictions, a straightforward quantile-based trading strategy based on the respective market predictions yields up to 116% return over three months before transaction costs. While the trading performance based on individual model forecasts remains relatively volatile, the average trading performance over three months ranges from 3.97% (for the 1-minute holding periods) to 51.92% (for the 60-minute holding periods), again monotonously increasing with more extended periods. However, the achieved trading returns cannot compensate for incurring transaction costs due to the particularly short holding periods and correspondingly frequent trading activities.

Research Question 2 *What are the most relevant features for predicting short-term movements of the Bitcoin market using different machine learning models?*

In order to answer Research Question 2, the machine learning models described in the empirical study presented in Chapter 2.2 are trained on a comprehensive feature set consisting of various technical, blockchain-based, sentiment-/interest-based, and asset-based features.

A permutation feature importance analysis reveals that the best-performing models, recurrent neural network ensembles and tree-based ensembles, prevalently rely on technical features (i.e., Bitcoin returns) for making predictions. Regarding these models, for longer prediction horizons, feature series besides the time series consisting of Bitcoin returns become increasingly important, such as time series consisting of Bitcoin transactions per second, weighted sentiment, and the number of tweets. An examination of the tree-based ensemble models reveals that less recent returns constitute the most important feature for longer prediction horizons. Concretely, the last-minute Bitcoin returns are most relevant regarding the 1-minute prediction horizon and the Bitcoin returns between 10 to 5 minutes before the prediction point are most relevant regarding the 5-minute prediction horizon. The Bitcoin returns between 10 to 20 minutes before the prediction time constitute the most relevant feature for the 15-minute prediction horizon. The Bitcoin returns between 40 to 20 minutes before the prediction time constitute the most relevant feature for the 60-minute prediction horizon.

Research Question 3 *What is the performance of machine learning models for generating statistical arbitrage in the cryptocurrency market?*

Research Question 3 concerns the performance of statistical arbitrage strategies based on machine learning-based cryptocurrency market predictions and is addressed in the study presented in Chapter 2.3. The study employs six different machine learning classifiers to predict the relative daily performance of the 100 largest cryptocurrencies by market capitalization. Subsequently, it evaluates the trading outcomes of a long-short portfolio strategy based on the out-of-sample predictions of each model.

The results demonstrate the potential of machine learning for cryptocurrency market prediction, as all utilized models significantly outperform a random classifier. The average accuracy values calculated on all cryptocurrencies range from 52.9% to 54.1% for the different models. These accuracy values lie between 57.7% and 59.5% when calculated on the subset of predictions with the 10% highest model confidences per class per day. A long-short portfolio strategy based on the predictions of the employed LSTM and GRU ensemble models yields the highest annualized out-of-sample Sharpe ratio after transaction costs of 3.23 and 3.12, respectively. In comparison, the buy-and-hold benchmark market portfolio strategy only yields a Sharpe ratio of 1.33. These results indicate a challenge to weak form cryptocurrency market efficiency, albeit the influence of certain limits to arbitrage cannot be entirely ruled out.

Research Question 4 *What is the predictive power of machine learning models predicting S&P 500 stock price movements during the COVID-19 pandemic?*

The following research question focuses on stock market prediction in the socioeconomic environment altered by the Covid 19 pandemic. In the empirical study presented in Chapter 3.1, random forest and benchmark logistic regression models are trained to predict next-day market movements of S&P 500 constituents. Two versions for each model type are compared, whereby one version is trained on COVID-19-related data and control variables, while the other is only trained on the control variables.

The study has two main results. First, all models make statistically viable predictions at a confidence level of 1%. Thereby, the random forest (53.84% accuracy) and the logistic regression model (52.74% accuracy), trained on the complete feature set, exhibit a higher predictive accuracy than their respective counterparts trained on data not directly related to COVID-19 50.87% and 50.56% accuracy. This finding suggests that COVID-19-related data exhibit predictive power in the period under consideration. Second, the random forest performs better and makes more balanced predictions than the logistic regression model. In combination with the finding that the best performing random forest architecture grows medium-sized trees, non-linear feature interactions with increased complexity may improve model forecasts. A permutation feature importance analysis reveals that the best performing random forest relies on numerous features for making predictions, as individual feature importance scores do not exceed 2%. The most important features are the newspaper-based infectious disease equity market volatility tracker and features that indicate the state of the pandemic, such as the changes in COVID-19-related cases, deaths, and the number of conducted COVID-19 PCR tests. Regarding the regional distribution of the latter, features that indicate the state of the pandemic in the US constitute the largest share of the most important features.

Research Question 5 *How do different levels of active and passive investment affect fundamental price efficiency?*

The next research question, addressed in Chapter 4.1, focuses on the impact of different investor compositions on the deviation from observed market prices from fundamental values. The presented study introduces a financial market simulation framework and examines the impact of changes in the central market parameters on market outcomes.

The study reveals that larger fractions of active investment within a market increase overall fundamental market efficiency. The marginal positive effect on fundamental market efficiency per additional fundamental investor decreases for larger fractions of fundamental investment within a market. The study further shows that high levels of passive investment within a market may facilitate technical price bubbles and impair fundamental market efficiency. The study further examines the sensitivity of these results towards central market parameters and finds that a less restrictive trading environment for active investors in the form of relaxed portfolio constraints or lower trading costs tends to facilitate trading and increase fundamental market efficiency. Furthermore, a lower risk aversion of all investors tends to increase fundamental market efficiency.

Research Question 6 *How do different degrees of accuracy of active investors' market forecasts affect fundamental market efficiency?*

The last research question addresses the impact of different accuracy degrees in individual active investors' market forecasts. In order to answer the final research question, the individual target price errors of active investors in the simulation framework presented in Chapter 4.1 are systematically altered, and the resulting level of fundamental market efficiency is analyzed.

The results show that more accurate target price forecasts, as, for instance, caused by the proliferation of machine learning in the asset management industry, lead to higher fundamental efficiency of market prices. Thereby, the increase in fundamental market efficiency depends on the level of active investment and is more pronounced in markets with a higher level of active investment.

5.2 Limitations

The presented results are subject to several individual limitations. First, all the presented studies depend on the time frame under consideration. This limitation applies especially to the empirical machine learning studies presented in Chapters 2 and 3, where the utilized datasets are naturally limited due to the recent and dynamic nature of the examined asset pricing challenges. While stable and significant for the respective periods, the results presented in these studies may not be valid for future market environments. Financial markets are constantly evolving, and market anomalies tend to disappear after publication (Zaremba et al., 2020). The time dependence of the described results might be enhanced by the dynamic and recent nature of the examined financial market developments, which naturally restricts the amount of relevant data. An evaluation of whether the presented results are valid in future market environments may reduce uncertainty regarding the temporal stability of the results.

Second, to varying degrees, all of the presented studies exhibit limited external validity. This limitation applies especially to the financial market simulation framework presented in Chapter 4.1. As with nearly all economic models, the presented framework simplifies the real world. Although the selected market parameter values are based on empirical findings to the extent possible, they are subject to certain assumptions. Additionally, the presented simulation setting is static, as investors generally optimize their respective mean-variance utility functions and do not adapt over time. With regards to the empirical studies presented in Chapters 2 and 3, external validity is reduced due to certain limitations concerning the possibility of cryptocurrency short positions and the assumption of consistent and fast data availability. A field study to evaluate the identified mechanisms of Chapter 4.1 may help increase external validity.

Third, all studies are quantitative in nature. Qualitative studies, for instance, focusing on investor intention and behavior in the examined market environments, may constitute a viable complement for the presented studies.

Last, the applied machine learning methods in Chapters 2 and 3 exhibit limitations regarding transparency and interpretability. While examining feature importances and model architectures aims to shed light on the applied models, the level of model transparency remains modest due to the nature of the applied machine learning algorithms. Current developments in the field of explainable artificial intelligence may help increase machine learning models' transparency.

5.3 Future Research

This chapter presents three possible pathways for future research based on the studies presented in this thesis. These include (1) explainable artificial intelligence for financial market prediction, (2) deep reinforcement learning for market simulation, and (3) behavioral finance approaches to examine cryptocurrency investing.

Explainable artificial intelligence The studies presented in Chapters 2.2, 2.3 and 3.1 highlight the potential of complex machine learning architectures (e.g., recurrent neural network ensembles) in the considered market prediction contexts. While these more complex models exhibit higher predictive power than traditional linear models, they also come with limitations regarding model interpretability. In an attempt to solve this trade-off between predictive power and model interpretability (Montavon et al., 2018; Došilović et al., 2018), the research field of explainable artificial intelligence has emerged over the last years (Arrieta et al., 2020; Berente et al., 2021). Regarding explainable artificial intelligence, model-agnostic and model-specific approaches can be differentiated (Samek et al., 2019). In this work, the model agnostic approach of permutation feature importance (Breiman, 2001; Fisher et al., 2019) is utilized, which only allows for modest model interpretability (Gu et al., 2020). Generally, explainable artificial intelligence approaches may enable an improved model evaluation (Lapuschkin et al., 2019), facilitate an increase in model robustness (Arrieta et al., 2020), and help to develop a better understanding of prediction problems and corresponding models (Siddiqui et al., 2019). With regards to financial market prediction, these advantages may be particularly important, as economically significant decisions can be based on these predictions. Against this backdrop, future research may examine emerging and profound explainable artificial intelligence approaches for financial market prediction to improve model quality and enhance the knowledge about underlying market mechanisms.

Behavioral finance The studies presented in Chapters 2.2 and 2.3 demonstrate that technical signals constitute crucial features for the market prediction of Bitcoin and other cryptocurrencies. Based on this finding, future research could conduct experimental studies to determine the source of the predictive power of technical features within these markets, as experimental finance approaches have successfully generated insights into various asset pricing issues (Noussair and Tucker, 2013). A potential source for the predictive power of technical features may be behavioral investment biases. These biases could include overconfidence (Daniel et al., 1998), anchoring (Tversky and Kahneman, 1974; Campbell and Sharpe, 2009), and the disposition effect (Shefrin and Statman, 1985). For instance, investor overconfidence may be induced by a lack of fundamental Bitcoin value in the traditional sense, as individual investors might perceive a reduced informational edge of professional asset managers in the Bitcoin market compared to other financial markets. Furthermore, researchers could examine whether cryptocurrency investment induces excitement or whether cryptocurrency investors constitute a subset of investors that is more excitable generally, as excitement has shown to induce an increase in trading activity and fuel market bubbles (Andrade et al., 2016).

Deep reinforcement learning The study presented in Chapter 4.1 introduces a financial market simulation framework that incorporates different agents. These agents exhibit stylized behavior that is defined by their nature. Future research may examine more dynamic simulation frameworks incorporating multiple self-learning agents that flexibly learn their behavior. A promising design approach for self-learning agents may be deep reinforcement learning. Deep reinforcement learning combines the fields of traditional reinforcement learning (Kaelbling et al., 1996) and deep learning (Bengio et al., 2009; Schmidhuber, 2015). Deep reinforcement learning has shown the ability to solve various complex decision tasks (François-Lavet et al., 2018). It has been applied successfully to a magnitude of different domains over recent years, such as physics (Degrave et al., 2022), robotics (Ibarz et al., 2021), communications (Luong et al., 2019), and finance (Hirchoua et al., 2021). With deep reinforcement learning, single-agent and multi-agent models can be differentiated.

While the more simple single-agent models have initially overshadowed multi-agent models, the latter has risen rapidly in popularity in recent times (Gronauer and Diepold, 2022). While early works have applied deep reinforcement learning within the field of agent-based financial market simulation, these works are limited by either an overly simplistic agent design (Spooner et al., 2018; Raman and Leidner, 2019) or by utilizing only a single deep reinforcement learning agent (Maeda et al., 2020). In this context, employing multi-agent deep reinforcement learning to sophisticated agent-based financial market simulation frameworks represents a natural next research step.

Appendices

APPENDIX A

SUPPLEMENTARY MATERIAL FOR CHAPTER 2.2

A.1 Supplemental Tables

Table A.1.: Overview of the class probability thresholds for trading. A buy (sell) trade is initiated if the model's predicted probability that a observation belongs to Class 1 (Class 0) exceeds the listed probability threshold

	1-Minute Predictions		5-Minute Predictions	
Model	Buy	Sell	Buy	Sell
GRU	0.6021	0.6029	0.6103	0.5874
LSTM	0.6040	0.6108	0.5909	0.5755
FNN	0.5379	0.5498	0.5788	0.5730
LR	0.5804	0.5971	0.6354	0.6112
GBC	0.5727	0.5888	0.5973	0.5814
RF	0.5095	0.5113	0.5110	0.5111

	15-Minute Predictions		60-Minute Predictions	
Model	Buy	Sell	Buy	Sell
GRU	0.6136	0.5932	0.5999	0.5972
LSTM	0.6065	0.5899	0.6071	0.6043
FNN	0.5601	0.5608	0.5872	0.5858
LR	0.6633	0.6292	0.6962	0.6604
GBC	0.6216	0.5996	0.6645	0.6210
RF	0.5141	0.5147	0.5173	0.5182

	Accuracy 1-Minute Predictions				
Model	Seed 0	Seed 1	Seed 2	Seed 3	Seed 4
GRU	0.5164	0.5159	0.5166	0.5193	0.5153
LSTM	0.5136	0.5169	0.5183	0.5205	0.5171
FNN	0.5166	0.5088	0.5151	0.5124	0.5049
RF	0.5170	0.5142	0.5182	0.5118	0.5110

	Accuracy 1-Minute Predictions				
Model	Seed 5	Seed 6	Seed 7	Seed 8	Seed 9
GRU	0.5194	0.5204	0.5179	0.5174	0.5190
LSTM	0.5156	0.5172	0.5192	0.5194	0.5184
FNN	0.5033	0.5142	0.5103	0.4943	0.5094
RN	0.5134	0.5109	0.5113	0.5123	0.5102

	Accuracy 5-Minute Predictions				
Model	Seed 0	Seed 1	Seed 2	Seed 3	Seed 4
GRU	0.5228	0.5219	0.5260	0.5242	0.5267
LSTM	0.5241	0.5202	0.5223	0.5218	0.5228
FNN	0.5192	0.5190	0.5210	0.5207	0.5090
RF	0.5292	0.5231	0.5269	0.5258	0.5261

	Accuracy 5-Minute Predictions				
Model	Seed 5	Seed 6	Seed 7	Seed 8	Seed 9
GRU	0.5221	0.5186	0.5212	0.5243	0.5244
LSTM	0.5218	0.5209	0.5210	0.5225	0.5235
FNN	0.5167	0.5203	0.5183	0.5140	0.5172
RN	0.5265	0.5265	0.5242	0.5270	0.5259

Table A.2.: Overview of the predictive accuracies of all stochastic prediction models for the individual random seeds

	Accuracy 15-Minute Predictions				
Model	Seed 0	Seed 1	Seed 2	Seed 3	Seed 4
GRU	0.5321	0.5349	0.5325	0.5385	0.5367
LSTM	0.5295	0.5313	0.5279	0.5284	0.5324
FNN	0.5189	0.5292	0.5146	0.5224	0.5040
RF	0.5361	0.5322	0.5341	0.5332	0.5350

	Accuracy 15-Minute Predictions				
Model	Seed 5	Seed 6	Seed 7	Seed 8	Seed 9
GRU	0.5326	0.5344	0.5332	0.5313	0.5366
LSTM	0.5305	0.5309	0.5240	0.5338	0.5260
FNN	0.5111	0.5272	0.5194	0.5173	0.5172
RN	0.5357	0.5323	0.5342	0.5332	0.5325

	Accuracy 60-Minute Predictions				
Model	Seed 0	Seed 1	Seed 2	Seed 3	Seed 4
GRU	0.5555	0.5403	0.5471	0.5569	0.5545
LSTM	0.5548	0.5502	0.5549	0.5505	0.5485
FNN	0.5466	0.5164	0.5474	0.5448	0.5085
RF	0.5528	0.5563	0.5535	0.5531	0.5598

	Accuracy 60-Minute Predictions				
Model	Seed 5	Seed 6	Seed 7	Seed 8	Seed 9
GRU	0.5560	0.5579	0.5517	0.5374	0.5515
LSTM	0.5523	0.5526	0.5602	0.5453	0.5563
FNN	0.5318	0.5475	0.5115	0.5092	0.5061
RN	0.5565	0.5536	0.5549	0.5567	0.5540

Table A.3.: Diebold-Mariano test p-values to reject the null hypothesis towards the alternative hypothesis that the forecast of model i on the test sample is more accurate than the forecast of model j

1-Minute Predictions

i \ j	GRU	LSTM	FNN	LR	GBC	RF	E (All)
GRU	-	0.8542	0.0000	0.0000	0.0000	0.0000	0.0001
LSTM	0.1458	-	0.0000	0.0000	0.0000	0.0000	0.0000
FNN	1.0000	1.0000	-	0.8890	0.8409	0.9456	0.9999
LR	1.0000	1.0000	0.1110	-	0.4610	0.6478	0.9809
GBC	1.0000	1.0000	0.1591	0.5390	-	0.8054	0.9994
RF	1.0000	1.0000	0.0544	0.3522	0.1946	-	0.9935
E (All)	0.9999	1.0000	0.0001	0.0191	0.0006	0.0065	-

5-Minute Predictions

i \ j	GRU	LSTM	FNN	LR	GBC	RF	E (All)
GRU	-	0.6373	0.0464	0.0000	0.9978	0.9110	0.9128
LSTM	0.3627	-	0.0246	0.0000	0.9945	0.8682	0.8425
FNN	0.9536	0.9754	-	0.0021	1.0000	0.9974	0.9995
LR	1.0000	1.0000	0.9979	-	1.0000	1.0000	1.0000
GBC	0.0022	0.0055	0.0000	0.0000	-	0.0412	0.0141
RF	0.0890	0.1318	0.0026	0.0000	0.9588	-	0.3476
E (All)	0.0872	0.1575	0.0005	0.0000	0.9859	0.6524	-

15-Minute Predictions

i \ j	GRU	LSTM	FNN	LR	GBC	RF	E (All)
GRU	-	0.0000	0.0000	0.0000	0.6756	0.1154	0.8878
LSTM	1.0000	-	0.0000	0.0000	0.9986	0.9599	1.0000
FNN	1.0000	1.0000	-	0.2029	1.0000	1.0000	1.0000
LR	1.0000	1.0000	0.7971	-	1.0000	1.0000	1.0000
GBC	0.3244	0.0014	0.0000	0.0000	-	0.0518	0.6491
RF	0.8846	0.0401	0.0000	0.0000	0.9482	-	0.9866
E (All)	0.1122	0.0000	0.0000	0.0000	0.3509	0.0134	-

60-Minute Predictions							
i \ j	GRU	LSTM	FNN	LR	GBC	RF	E (All)
GRU	-	0.9999	0.0000	0.0000	0.5792	0.4194	0.7790
LSTM	0.0000	-	0.0000	0.0000	0.0500	0.0052	0.0213
FNN	1.0000	1.0000	-	1.0000	1.0000	1.0000	1.0000
LR	1.0000	1.0000	0.0000	-	1.0000	1.0000	1.0000
GBC	0.4208	0.9500	0.0000	0.0000	-	0.3433	0.6387
RF	0.5806	0.9948	0.0000	0.0000	0.6567	-	0.8145
E (All)	0.2210	0.9787	0.0000	0.0000	0.3613	0.1855	-

A.2 Supplemental Graphical Material

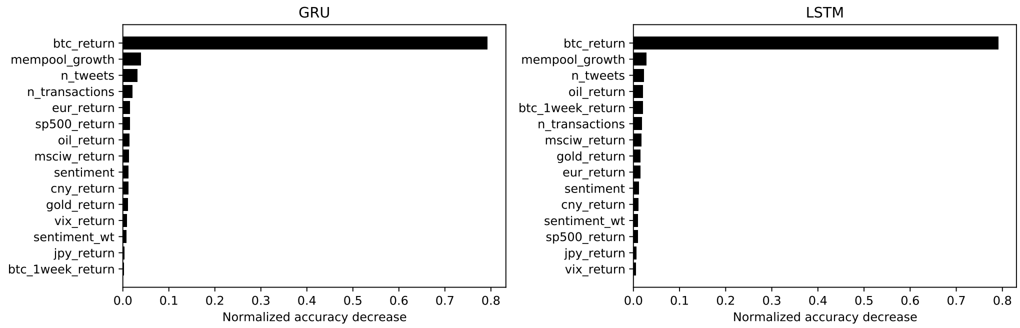


Figure A.1.: Feature importance of the models with memory function on the 1-minute prediction horizon

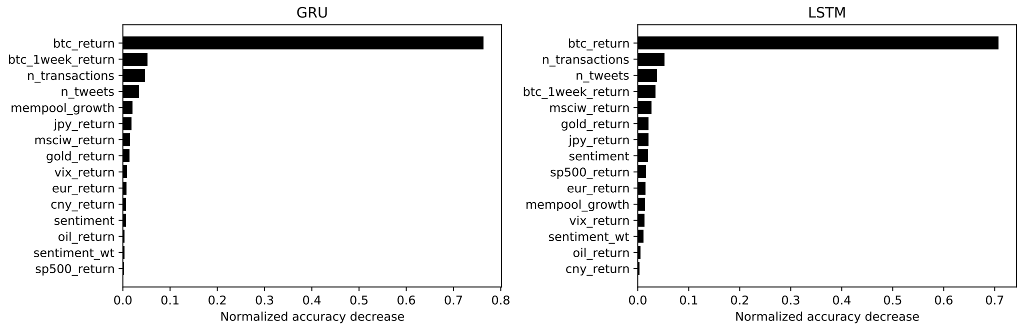


Figure A.2.: Feature importance of the models with memory function on the 5-minute prediction horizon

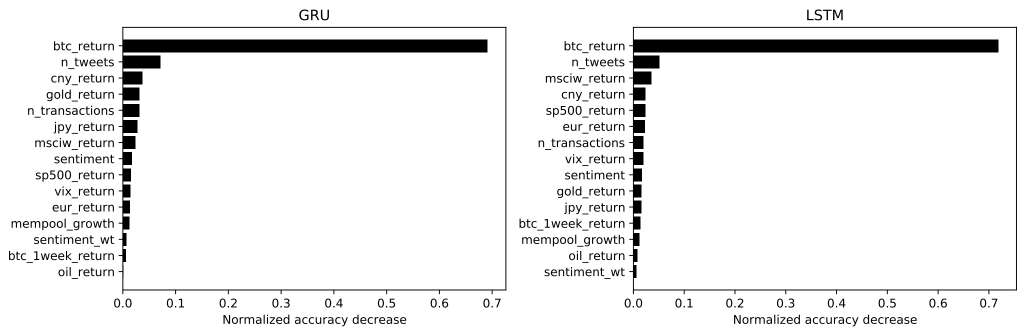


Figure A.3.: Feature importance of the models with memory function on the 15-minute prediction horizon

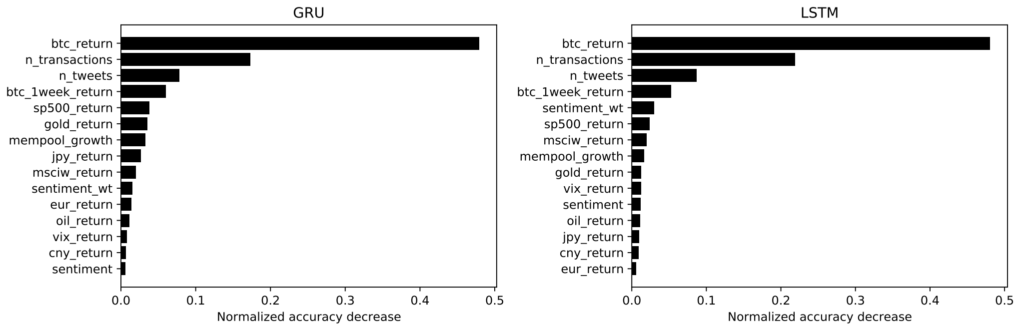


Figure A.4.: Feature importance of the models with memory function on the 60-minute prediction horizon

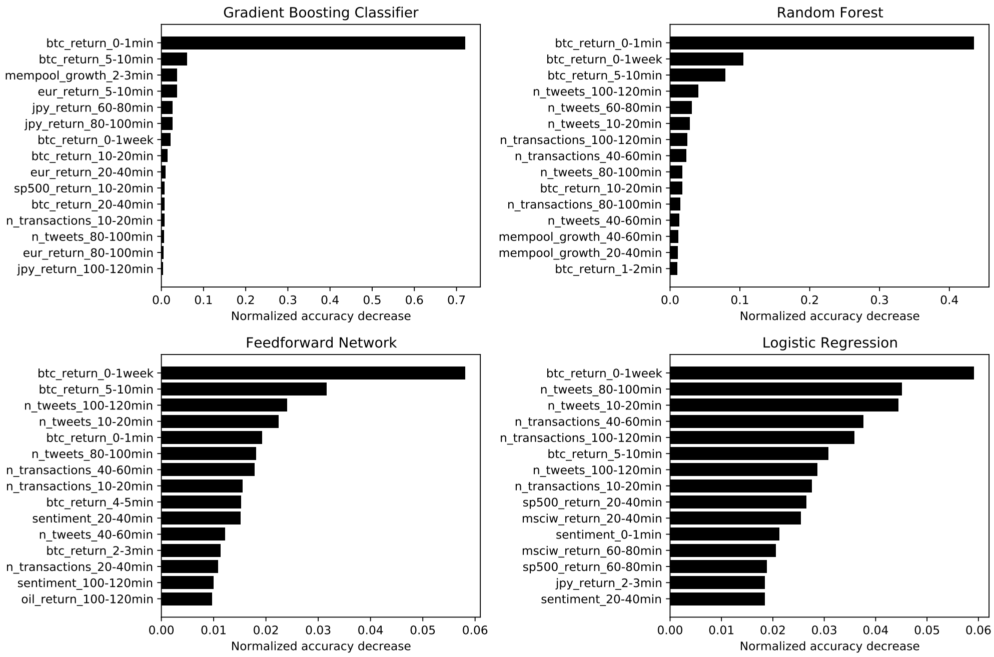


Figure A.5.: Feature importance of the models without memory function on the 1-minute prediction horizon

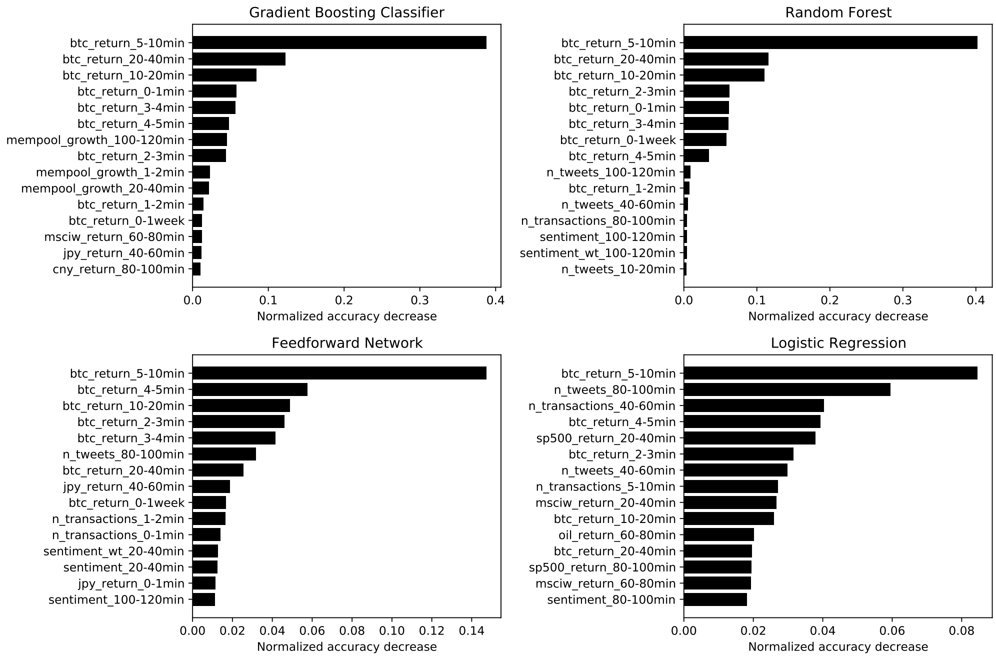


Figure A.6.: Feature importance of the models without memory function on the 5-minute prediction horizon

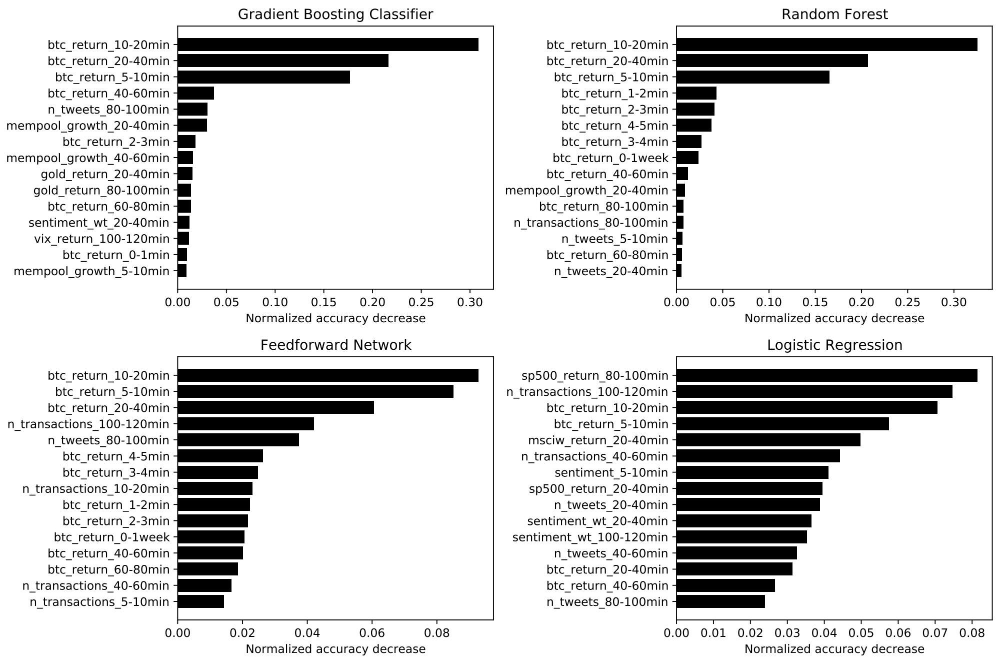


Figure A.7.: Feature importance of the models without memory function on the 15-minute prediction horizon

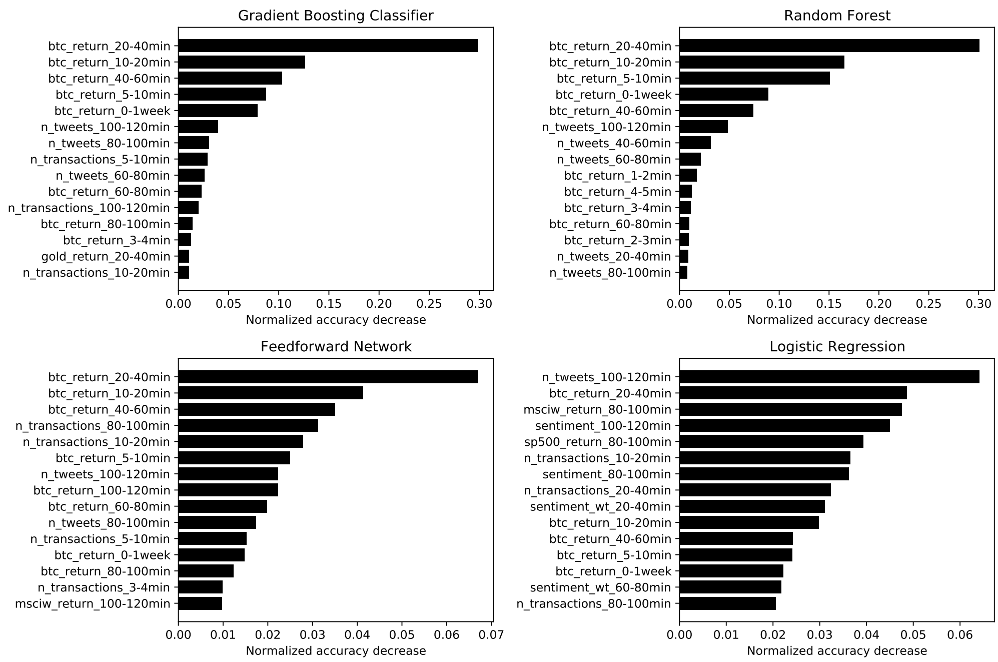


Figure A.8.: Feature importance of the models without memory function on the 60-minute prediction horizon

APPENDIX B

SUPPLEMENTARY MATERIAL FOR CHAPTER 2.3

B.1 Supplemental Tables

No	Coin Symbol	Exclusion Reason
1	BLUNA	Data issues
2	KNCL	Data issues
3	CDAI	Data issues
4	LN	Data issues
5	SOLVE	Data issues
6	VERI	Data issues
7	VEE	Data issues
8	JASMY	Data issues
9	MSOL	Data issues
10	MAID	Data issues
11	BUSD	Stablecoin
12	HUSD	Stablecoin
13	SAI	Stablecoin
14	DAI	Stablecoin
15	TUSD	Stablecoin
16	USDC	Stablecoin
17	USDT	Stablecoin
18	UST	Stablecoin
19	FRAX	Stablecoin
20	MIM	Stablecoin
21	MUSD	Stablecoin
22	SUSD	Stablecoin
23	USDN	Stablecoin

Table B.1.: Overview of coins excluded from the study and the reason for their exclusion

Table B.2.: Complete p-values for the binomial test of achieving the respective model's accuracies for the combined test sets of all study periods with the null hypothesis of each model having atrue predictive performance of 0.5 or below

Portfolio Size	**k = 1**	**k = 2**	**k = 5**	**k = 10**	**k = 20**	**k = 50**
GRU	7.012528e-20	1.048364e-33	7.864622e-62	7.823276e-85	1.137218e-100	2.136124e-84
LSTM	4.334874e-20	1.334199e-31	4.308048e-56	8.736139e-90	9.042803e-115	3.202961e-110
TCN	2.829318e-17	1.483387e-25	9.661208e-50	8.459317e-74	3.218419e-106	3.765699e-88
GBC	7.385857e-18	1.505160e-32	1.925738e-49	5.630044e-81	1.079329e-111	1.047139e-84
RF	7.385857e-18	1.334199e-31	2.365775e-58	5.677780e-93	1.566451e-139	3.419382e-119
LR	1.057233e-16	3.115607e-24	1.693509e-38	5.755924e-59	4.059272e-71	6.528285e-56

Table B.3.: Complete p-values per study period for the binomial test of realizing the respective prediction accuracy with the null hypothesis of each model having a true predictive performance of 0.5 or below

	Portfolio Size	**1**	**2**	**5**	**10**	**20**	**50**
Model	**Study Period**						
GRU	1	1.966113e-09	8.143005e-10	2.829318e-17	8.593063e-21	4.129621e-27	5.916977e-23
	2	2.053107e-04	5.951555e-06	5.829624e-11	6.540142e-14	9.855201e-25	1.468607e-25
	3	5.147597e-05	9.767993e-08	1.813239e-17	1.306882e-18	1.149283e-16	1.406433e-11
	4	5.147597e-05	1.355567e-09	4.344051e-10	1.987192e-16	2.808584e-21	9.363543e-19
	5	2.294356e-03	1.518496e-07	2.403512e-14	1.998324e-23	1.325015e-19	3.519168e-15
LSTM	1	1.273132e-06	2.496326e-08	5.388406e-13	3.441800e-18	3.811074e-26	8.770927e-26
	2	2.053107e-04	1.793673e-05	3.445781e-12	2.301733e-17	3.811074e-26	1.414983e-36
	3	5.147597e-05	9.767993e-08	5.388406e-13	9.309605e-20	2.131927e-18	2.449832e-13
	4	5.147597e-05	8.143005e-10	1.610661e-10	1.466401e-16	2.353418e-27	2.909156e-25
	5	1.914557e-05	2.344697e-07	1.074886e-14	4.629964e-26	3.368743e-25	1.966950e-20
TCN	1	7.103290e-07	3.679503e-09	1.074886e-14	6.873145e-25	1.140469e-34	1.860338e-27
	2	3.160513e-05	2.759758e-06	7.158810e-15	6.864123e-15	4.259964e-29	1.040745e-28
	3	3.908212e-07	1.518496e-07	7.102294e-12	3.353233e-13	4.073274e-15	3.161564e-10
	4	2.156425e-02	1.233186e-02	4.772219e-05	9.130275e-10	3.453358e-12	1.125556e-11
	5	3.280082e-03	3.960555e-08	1.653800e-12	6.080850e-21	7.581627e-28	2.945825e-22
GBC	1	7.103290e-07	9.016950e-14	2.511002e-16	9.616070e-27	3.234463e-37	1.994906e-29
	2	3.171100e-04	7.123364e-05	1.143767e-09	1.955044e-13	1.286722e-24	1.546419e-26
	3	2.294356e-03	1.246133e-06	2.940061e-09	1.215689e-14	1.074106e-18	2.375169e-14
	4	2.250362e-06	2.241060e-09	6.014537e-10	1.987192e-16	5.379156e-19	3.092007e-15
	5	7.274396e-04	2.759758e-06	4.953697e-12	6.806688e-19	2.980659e-22	1.369435e-10
RF	1	5.147597e-05	8.290170e-07	3.687780e-13	6.579187e-24	4.078689e-39	4.881953e-36
	2	1.311896e-04	9.716225e-09	1.141086e-12	8.954819e-18	3.219094e-28	1.468607e-25
	3	1.584634e-03	2.759758e-06	4.754797e-15	4.162588e-23	7.330139e-30	9.167393e-21
	4	1.144208e-05	1.355567e-09	4.344051e-10	3.144490e-17	1.664621e-26	8.046203e-25
	5	1.914557e-05	3.596098e-07	7.158810e-15	2.532673e-19	5.769668e-25	2.281399e-21
LR	1	7.274396e-04	1.233186e-02	7.072222e-03	3.693249e-05	9.511710e-05	3.971647e-06
	2	1.080569e-03	1.246133e-06	7.378645e-09	1.618262e-11	1.714117e-21	9.816374e-24
	3	7.103290e-07	2.496326e-08	3.149451e-15	1.816706e-19	2.852275e-24	3.783289e-11
	4	6.745841e-06	3.679503e-09	1.448182e-11	5.845011e-17	8.535440e-19	1.837793e-15
	5	3.280082e-03	1.249970e-05	4.344051e-10	3.634647e-16	7.494437e-15	2.085115e-10

Table B.4.: Daily return and risk metrics and annualized risk-return metrics for all models and the market (MKT) for the short leg of the k = 5 portfolio

	GRU	**LSTM**	**TCN**	**GBC**	**RF**	**LR**	**MKT**
Mean Return	0.00217	0.00065	-0.00221	-0.00117	0.00002	0.00013	0.00330
Return Standard Deviation	0.07646	0.07612	0.07676	0.07285	0.07236	0.07025	0.04729
Downside Risk	0.90256	0.92223	0.93714	0.88138	0.85539	0.81862	0.54851
VaR 1%	-0.22028	-0.22320	-0.22297	-0.21930	-0.20235	-0.19868	-0.14187
VaR 5%	-0.13972	-0.13221	-0.13441	-0.12916	-0.12400	-0.11725	-0.07389
CVaR 1%	-0.28448	-0.30545	-0.31089	-0.27844	-0.26553	-0.24951	-0.19054
CVaR 5%	-0.19742	-0.19863	-0.19963	-0.18967	-0.18092	-0.17207	-0.11744
Annual. Volatility	1.46079	1.45425	1.46656	1.39176	1.38247	1.34212	0.90338
Sharpe Ratio	0.54182	0.16135	-0.55060	-0.30860	0.00309	0.03408	1.33105
Sortino Ratio	0.72814	0.21126	-0.71546	-0.40462	0.00414	0.04639	1.82023
Excess Sharpe	-0.00993	-0.02357	-0.04824	-0.04013	-0.02977	-0.02910	-

APPENDIX C

SUPPLEMENTARY MATERIAL FOR CHAPTER 4.1

C.1 Supplemental Tables

Table C.1.: Fundamental price deviation and trading volume for different investor compositions in the market setting with a share of 20 percent random investment averaged over all random seeds

# of Agents Active/Passive/Random	Mean Price Deviation Relative [%]	Mean Trading Volume [$]
0/400/100	90.04	504,301
20/380/100	58.71	825,915
40/360/100	55.25	1,277,081
60/340/100	49.75	1,725,518
80/320/100	63.38	2,285,153
100/300/100	65.24	2,764,043
120/280/100	58.32	3,203,483
140/260/100	125.33	4,217,106
160/240/100	63.66	4,349,119
180/220/100	43.63	4,824,851
200/200/100	68.77	5,399,436
220/180/100	55.59	5,819,052
240/160/100	51.43	6,322,966
260/140/100	57.73	6,944,992
280/120/100	47.17	7,420,431
300/100/100	37.20	7,846,136
320/80/100	31.83	8,306,960
340/60/100	28.10	8,912,337
360/40/100	25.07	9,499,031
380/20/100	22.71	10,114,443
400/0/100	20.64	10,737,114

Table C.2.: Overview of the tests for the different stylized facts detailed over all investor compositions and random seeds. Bold values denote that the stylized fact is fulfilled

Composition/ Seed	Excess Kurtosis/ KS Test Gaussian p-Value	Power-Law Alpha/ Log-Likelihood-Ratio Power-Law-Exponential, p-Value	Excess Kurtosis Residuals/ KS Test Gaussian, p-Value	Position Maximum Positive/Negative
0-100-400 /0	**3.26 /< 0.001**	**4.77 /16.10**, 0.122	**2.960 /< 0.001**	95.12 / 100.88
0-100-400 /1	**3.27 /< 0.001**	**4.56 /26.60**, 0.0585	**2.734 /< 0.001**	**142.63 / 139.91**
0-100-400 /2	**3.76 /< 0.001**	**4.41 /49.6, < 0.001**	**2.980 /< 0.001**	**124.88 / 124.21**
0-100-400 /3	**3.17 /< 0.001**	5.02 /**0.152**, 0.977	**2.826 /< 0.001**	109.82 / 111.49
0-100-400 /4	**3.29 /< 0.001**	**4.58 /18.30**, 0.0572	**3.189 /< 0.001**	**151.68 / 149.10**
0-100-400 /5	**3.09 /< 0.001**	5.64 /**3.62**, 0.328	3.292 /< 0.001	**116.44 / 116.31**
0-100-400 /6	**3.38 /< 0.001**	**4.52 /19.50**, 0.129	**3.152 /< 0.001**	**135.21 / 134.11**
0-100-400 /7	**3.14 /< 0.001**	**4.76 /12.20**, 0.173	**2.972 /< 0.001**	**85.91 / 76.59**
5-95-400 /0	**3.42 /< 0.001**	**4.76 /26.0, 0.00938**	**3.173 /< 0.001**	142.46 / 143.00
5-95-400 /1	**3.37 /< 0.001**	**4.52 /34.1, 0.0114**	**2.000 /< 0.001**	**109.70 / 109.69**
5-95-400 /2	**3.04 /< 0.001**	**4.63 /6.45**, 0.504	3.395 /< 0.001	**151.79 / 150.93**
5-95-400 /3	**3.01 /< 0.001**	**4.80 /16.20**, 0.0774	**2.860 /< 0.001**	**96.08 / 96.06**
5-95-400 /4	**3.19 /< 0.001**	**4.59 /26.1, 0.0285**	4.098 /< 0.001	**108.20 / 89.22**
5-95-400 /5	**3.35 /< 0.001**	**4.78 /22.8, 0.0173**	**3.124 /< 0.001**	**116.92 / 114.83**
5-95-400 /6	**3.25 /< 0.001**	**4.88 /15.00**, 0.177	**2.832 /< 0.001**	**146.35 / 144.79**
5-95-400 /7	**3.17 /< 0.001**	**4.68 /25.0, 0.0289**	**2.407 /< 0.001**	**115.48 / 115.42**
10-90-400 /0	**3.59 /< 0.001**	**4.40 /20.90**, 0.203	3.784 /< 0.001	**115.71 / 115.60**
10-90-400 /1	**3.91 /< 0.001**	**4.66 /12.30**, 0.225	6.474 /< 0.001	**118.72 / 115.99**
10-90-400 /2	**3.95 /< 0.001**	**4.49 /12.80**, 0.32	6.463 /< 0.001	**131.55 / 130.78**
10-90-400 /3	**3.41 /< 0.001**	**4.77 /15.30**, 0.0577	**3.350 /< 0.001**	**99.56 / 98.04**
10-90-400 /4	**3.80 /< 0.001**	**4.44 /21.5, 0.0441**	3.870 /< 0.001	**120.09 / 118.21**
10-90-400 /5	**3.76 /< 0.001**	**4.69 /19.40**, 0.0659	4.185 /< 0.001	**145.43 / 142.83**
10-90-400 /6	**3.24 /< 0.001**	**4.86 /18.30**, 0.0824	**3.146 /< 0.001**	**133.10 / 105.53**
10-90-400 /7	**3.15 /< 0.001**	**4.59** /-3.52, 0.737	**2.529 /< 0.001**	**135.06 / 133.85**
15-85-400 /0	**3.24 /< 0.001**	**4.94 /5.71**, 0.517	**2.663 /< 0.001**	**120.43 / 119.73**
15-85-400 /1	**3.05 /< 0.001**	5.22 /**6.33**, 0.203	**3.032 /< 0.001**	104.30 / 104.70
15-85-400 /2	**2.83 /< 0.001**	5.17 /**1.85**, 0.685	3.051 /< 0.001	**143.45 / 135.80**
15-85-400 /3	**3.25 /< 0.001**	**4.78 /8.16**, 0.26	3.608 /< 0.001	**121.33 / 118.91**
15-85-400 /4	**2.93 /< 0.001**	**4.61** /-5.0, 0.618	3.444 /< 0.001	**93.81 / 90.89**
15-85-400 /5	**2.79 /< 0.001**	**4.80** /-1.89, 0.833	3.297 /< 0.001	**125.38 / 125.04**

Composition/ Seed	Excess Kurtosis/ KS Test Gaussian p-Value	Power-Law Alpha/ Log-Likelihood-Ratio Power-Law - Exponential, p-Value	Excess Kurtosis Residuals / KS Test Gaussian, p-Value	Position Maximum Positive/Negative
15-85-400 /6	**3.13 /< 0.001**	**4.57** /**12.60**, 0.273	**2.359 /< 0.001**	**114.97 / 114.18**
15-85-400 /7	**3.63 /< 0.001**	**4.83 /16.7, 0.0384**	**2.912 /< 0.001**	**147.23 / 142.60**
20-80-400 /0	**3.04 /< 0.001**	**4.79 /18.6, 0.0353**	**2.811 /< 0.001**	**112.57 / 110.44**
20-80-400 /1	**2.99 /< 0.001**	**4.79** /**6.78**, 0.425	3.479 /< 0.001	130.05 / 134.22
20-80-400 /2	**3.56 /< 0.001**	**4.92** /**6.54**, 0.261	**3.141 /< 0.001**	**95.36 / 92.36**
20-80-400 /3	**2.52 /< 0.001**	5.22 /**4.48**, 0.492	**1.908 /< 0.001**	**106.32 / 106.00**
20-80-400 /4	**3.30 /< 0.001**	**4.55 /26.8, 0.0195**	4.308 /< 0.001	**102.04 / 100.60**
20-80-400 /5	**2.81 /< 0.001**	**4.89** /**6.64**, 0.363	**2.659 /< 0.001**	**127.47 / 126.26**
20-80-400 /6	**2.78 /< 0.001**	**4.87** /-3.57, 0.708	**2.340 /< 0.001**	118.50 / 118.58
20-80-400 /7	**3.37 /< 0.001**	**4.62** /**21.10**, 0.0675	**3.226 /< 0.001**	134.79 / 138.75
25-75-400 /0	**2.70 /< 0.001**	5.37 /**1.65**, 0.716	**2.696 /< 0.001**	**130.23 / 129.39**
25-75-400 /1	**3.20 /< 0.001**	**4.62** /-0.963, 0.918	**2.850 /< 0.001**	97.88 / 99.77
25-75-400 /2	**2.90 /< 0.001**	**4.85** /**7.61**, 0.35	2.917 /< 0.001	**125.87 / 124.38**
25-75-400 /3	**3.83 /< 0.001**	**4.75 /32.6, 0.00861**	3.965 /< 0.001	127.03 / 129.06
25-75-400 /4	**3.09 /< 0.001**	5.33 /**3.84**, 0.367	**3.053 /< 0.001**	**130.89 / 99.85**
25-75-400 /5	**3.75 /< 0.001**	5.02 /**8.28**, 0.182	4.038 /< 0.001	**127.41 / 124.51**
25-75-400 /6	**2.95 /< 0.001**	**4.93** /**4.87**, 0.468	**2.898 /< 0.001**	**160.88 / 111.29**
25-75-400 /7	**3.59 /< 0.001**	**4.71** /**13.70**, 0.167	**3.449 /< 0.001**	**110.94 / 82.17**
30-70-400 /0	**2.86 /< 0.001**	**4.81** /-0.409, 0.951	2.872 /< 0.001	**127.01 / 124.41**
30-70-400 /1	**3.60 /< 0.001**	**4.65** /**18.70**, 0.124	**2.697 /< 0.001**	**101.03 / 82.75**
30-70-400 /2	**3.16 /< 0.001**	5.16 /**9.68**, 0.215	**2.494 /< 0.001**	112.42 / 113.16
30-70-400 /3	**3.77 /< 0.001**	**4.70** /**8.45**, 0.35	4.222 /< 0.001	110.62 / 111.68
30-70-400 /4	**3.42 /< 0.001**	5.47 /**0.319**, 0.918	4.084 /< 0.001	**115.57 / 112.32**
30-70-400 /5	**2.87 /< 0.001**	**4.92** /**8.82**, 0.251	**2.512 /< 0.001**	**123.42 / 122.13**
30-70-400 /6	**3.39 /< 0.001**	**4.83 /21.2, 0.0275**	3.448 /< 0.001	**108.03 / 88.50**
30-70-400 /7	**3.01 /< 0.001**	**4.83** /-1.65, 0.811	3.401 /< 0.001	**128.18 / 125.99**
35-65-400 /0	**2.82 /< 0.001**	**4.90 /24.6, 0.016**	**2.665 /< 0.001**	**94.97 / 91.14**
35-65-400 /1	**2.74 /< 0.001**	5.00 /-5.39, 0.366	2.986 /< 0.001	**128.00 / 126.79**
35-65-400 /2	**2.92 /< 0.001**	**4.64** /**5.56**, 0.625	2.991 /< 0.001	**105.70 / 105.37**
35-65-400 /3	**3.04 /< 0.001**	**4.51 /23.8, 0.043**	**2.924 /< 0.001**	**131.61 / 129.89**
35-65-400 /4	**3.06 /< 0.001**	**4.91** /**6.63**, 0.317	3.419 /< 0.001	**93.18 / 90.61**
35-65-400 /5	**2.91 /< 0.001**	**4.80 /17.2, 0.0472**	**2.608 /< 0.001**	**132.01 / 129.10**
35-65-400 /6	**2.72 /< 0.001**	**4.89** /**3.05**, 0.671	**2.306 /< 0.001**	**104.82 / 90.17**
35-65-400 /7	**3.13 /< 0.001**	**4.51** /**5.91**, 0.542	**2.972 /< 0.001**	98.13 / 100.11
40-60-400 /0	**2.78 /< 0.001**	**4.67** /**8.65**, 0.434	**2.675 /< 0.001**	106.26 / 106.61

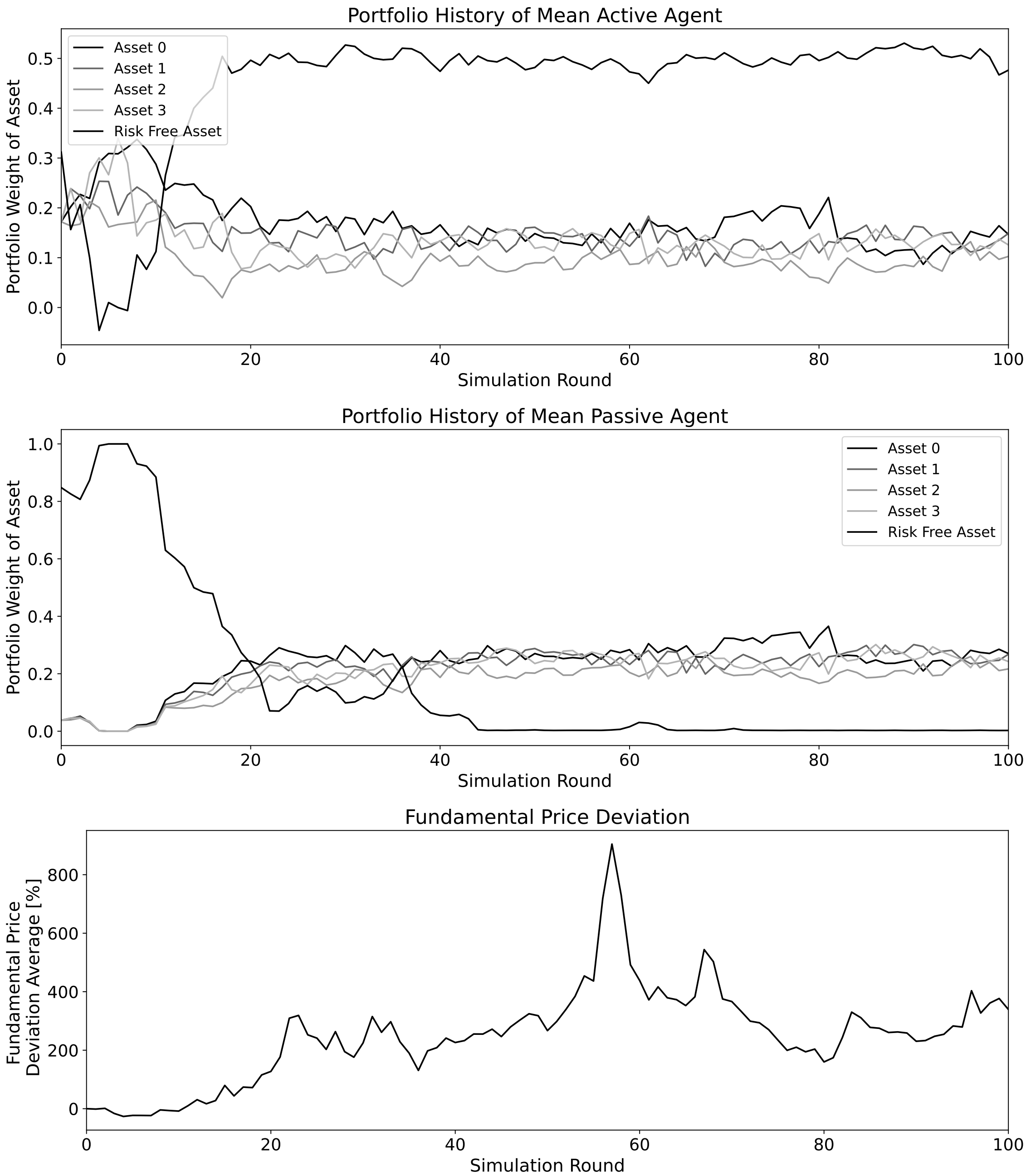
Composition/ Seed	Excess Kurtosis/ KS Test Gaussian p-Value	Power-Law Alpha/ Log-Likelihood-Ratio Power-Law - Exponential, p-Value	Excess Kurtosis Residuals / KS Test Gaussian, p-Value	Position Maximum Positive/Negative
40-60-400 /1	**2.78 /< 0.001**	**5.00** /-3.91, 0.525	**2.765 /< 0.001**	**108.74 / 104.81**
40-60-400 /2	**2.89 /< 0.001**	5.11 /**2.19**, 0.703	2.909 /< 0.001	**126.60 / 111.38**
40-60-400 /3	**2.74 /< 0.001**	5.07 /**2.65**, 0.693	**2.489 /< 0.001**	125.56 / 126.59
40-60-400 /4	**3.56 /< 0.001**	**4.69 /12.20**, 0.213	**3.265 /< 0.001**	**100.15 / 99.87**
40-60-400 /5	**3.13 /< 0.001**	5.54 /**2.51**, 0.473	4.815 /< 0.001	88.96 / 92.60
40-60-400 /6	**3.48 /< 0.001**	5.03 **/19.2, 0.0157**	4.647 /< 0.001	**111.30 / 109.89**
40-60-400 /7	**3.01 /< 0.001**	**4.61 /0.597**, 0.949	3.040 /< 0.001	103.05 / 110.28
45-55-400 /0	**3.63 /< 0.001**	5.85 /**2.35**, 0.308	4.644 /< 0.001	**129.42 / 128.17**
45-55-400 /1	**3.23 /< 0.001**	**4.63 /14.50**, 0.167	4.366 /< 0.001	**102.29 / 100.71**
45-55-400 /2	**3.74 /< 0.001**	**4.88 /24.6, 0.0128**	**3.403 /< 0.001**	108.21 / 110.33
45-55-400 /3	**3.28 /< 0.001**	**4.79 /9.03**, 0.259	4.784 /< 0.001	**121.48 / 117.95**
45-55-400 /4	**2.47 /< 0.001**	**4.58 /3.93**, 0.715	2.720 /< 0.001	**100.16 / 95.21**
45-55-400 /5	**3.04 /< 0.001**	**4.78 /13.70**, 0.0858	3.175 /< 0.001	**116.51 / 115.33**
45-55-400 /6	**3.10 /< 0.001**	**4.52 /28.7, 0.0454**	3.385 /< 0.001	**97.79 / 93.62**
45-55-400 /7	**3.44 /< 0.001**	**4.55 /5.67**, 0.587	**3.117 /< 0.001**	**96.50 / 92.80**
50-50-400 /0	**2.60 /< 0.001**	**4.54 /6.40**, 0.59	3.338 /< 0.001	**146.78 / 105.45**
50-50-400 /1	**3.19 /< 0.001**	**4.70 /10.00**, 0.365	**3.100 /< 0.001**	**93.40 / 86.22**
50-50-400 /2	**2.96 /< 0.001**	**4.87 /10.40**, 0.225	**2.803 /< 0.001**	**127.45 / 126.86**
50-50-400 /3	**3.02 /< 0.001**	**4.92 /14.70**, 0.133	3.082 /< 0.001	**112.16 / 111.35**
50-50-400 /4	**3.38 /< 0.001**	**4.39 /11.50**, 0.423	3.810 /< 0.001	**108.81 / 106.69**
50-50-400 /5	**3.22 /< 0.001**	**4.85 /7.71**, 0.383	4.450 /< 0.001	**94.44 / 92.84**
50-50-400 /6	**3.09 /< 0.001**	5.13 /**6.23**, 0.289	**2.796 /< 0.001**	**119.24 / 117.96**
50-50-400 /7	**3.70 /< 0.001**	**4.52 /19.10**, 0.14	**3.067 /< 0.001**	103.47 / 108.77
55-45-400 /0	**2.80 /< 0.001**	**4.84 /5.04**, 0.541	3.083 /< 0.001	102.58 / 103.25
55-45-400 /1	**2.70 /< 0.001**	**4.97 /1.55**, 0.828	2.948 /< 0.001	**93.26 / 92.00**
55-45-400 /2	**2.67 /< 0.001**	**4.68 /11.90**, 0.226	**2.402 /< 0.001**	**114.04 / 113.68**
55-45-400 /3	**3.22 /< 0.001**	**4.85 /8.70**, 0.259	3.523 /< 0.001	**122.22 / 117.13**
55-45-400 /4	**3.03 /< 0.001**	5.30 /-0.659, 0.88	3.216 /< 0.001	**110.78 / 89.72**
55-45-400 /5	**3.75 /< 0.001**	**4.60 /41.3, 0.00235**	4.188 /< 0.001	**116.40 / 110.94**
55-45-400 /6	**3.39 /< 0.001**	5.16 /**3.90**, 0.448	3.525 /< 0.001	**120.52 / 120.22**
55-45-400 /7	**2.59 /< 0.001**	**4.89 /2.41**, 0.783	**2.555 /< 0.001**	107.23 / 111.68
60-40-400 /0	**2.79 /< 0.001**	**4.61 /8.30**, 0.507	**2.453 /< 0.001**	**123.33 / 119.91**
60-40-400 /1	**3.22 /< 0.001**	**4.36 /36.6, 0.0186**	3.848 /< 0.001	**104.83 / 91.07**
60-40-400 /2	**2.89 /< 0.001**	5.03 /**16.60**, 0.0529	3.391 /< 0.001	132.97 / 133.52
60-40-400 /3	**3.13 /< 0.001**	5.02 /**4.35**, 0.569	3.388 /< 0.001	**100.77 / 99.46**

Composition/ Seed	Excess Kurtosis/ KS Test Gaussian p-Value	Power-Law Alpha/ Log-Likelihood-Ratio Power-Law - Exponential, p-Value	Excess Kurtosis Residuals / KS Test Gaussian, p-Value	Position Maximum Positive/Negative
60-40-400 /4	**3.35 /< 0.001**	6.25 /**1.88**, 0.383	**3.198 /< 0.001**	**117.72 / 95.23**
60-40-400 /5	**3.47 /< 0.001**	5.32 /**6.23**, 0.252	**3.342 /< 0.001**	**97.88 / 97.51**
60-40-400 /6	**2.64 /< 0.001**	**4.86** /**17.90**, 0.0562	2.837 /< 0.001	**97.12 / 95.92**
60-40-400 /7	**3.09 /< 0.001**	**4.46** /**3.24**, 0.763	3.332 /< 0.001	**137.11 / 111.61**
65-35-400 /0	**3.08 /< 0.001**	**4.70** /**3.90**, 0.702	4.448 /< 0.001	**133.22 / 130.82**
65-35-400 /1	**3.18 /< 0.001**	5.03 /-0.265, 0.962	3.323 /< 0.001	**108.57 / 101.03**
65-35-400 /2	**2.34 /< 0.001**	**4.93** /-1.4, 0.861	2.842 /< 0.001	**90.22 / 86.79**
65-35-400 /3	**3.28 /< 0.001**	**4.92** /**12.60**, 0.176	3.677 /< 0.001	**138.51 / 104.83**
65-35-400 /4	**3.79 /< 0.001**	**4.37** /**31.80**, 0.0508	**2.641 /< 0.001**	**99.05 / 98.04**
65-35-400 /5	**3.38 /< 0.001**	**4.69** /**18.80**, 0.0659	4.698 /< 0.001	**102.74 / 101.88**
65-35-400 /6	**2.94 /< 0.001**	5.10 /**5.21**, 0.452	3.553 /< 0.001	**101.39 / 98.94**
65-35-400 /7	**3.25 /< 0.001**	**4.75** /-4.03, 0.617	**2.317 /< 0.001**	**116.04 / 112.30**
70-30-400 /0	**2.32 /< 0.001**	**4.96** /-2.42, 0.778	2.476 /< 0.001	**141.95 / 141.89**
70-30-400 /1	**2.83 /< 0.001**	**4.79** /-2.11, 0.789	**2.664 /< 0.001**	**113.35 / 108.81**
70-30-400 /2	**2.92 /< 0.001**	**4.97** /**5.80**, 0.443	**2.707 /< 0.001**	**122.15 / 117.78**
70-30-400 /3	**3.21 /< 0.001**	5.13 /**0.19**, 0.967	**2.590 /< 0.001**	112.82 / 112.85
70-30-400 /4	**3.07 /< 0.001**	5.65 /**0.113**, 0.975	**2.700 /< 0.001**	**102.00 / 98.81**
70-30-400 /5	**2.99 /< 0.001**	**4.94** /**14.00**, 0.0975	3.019 /< 0.001	**128.85 / 128.76**
70-30-400 /6	**3.02 /< 0.001**	5.70 /**1.62**, 0.671	3.281 /< 0.001	**119.23 / 118.43**
70-30-400 /7	**3.70 /< 0.001**	**4.39** /**20.00**, 0.101	**3.018 /< 0.001**	112.90 / 114.95
75-25-400 /0	**2.62 /< 0.001**	5.33 /**2.12**, 0.643	2.691 /< 0.001	**140.09 / 138.12**
75-25-400 /1	**3.26 /< 0.001**	**4.32** /**6.94**, 0.63	**2.685 /< 0.001**	**97.43 / 92.45**
75-25-400 /2	**2.32 /< 0.001**	5.20 /**0.414**, 0.941	**1.998 /< 0.001**	**116.94 / 116.56**
75-25-400 /3	**2.73 /< 0.001**	**4.69 /24.4, 0.0337**	3.067 /< 0.001	108.87 / 109.08
75-25-400 /4	**2.92 /< 0.001**	5.09 /**9.77**, 0.213	**2.889 /< 0.001**	**97.00 / 95.08**
75-25-400 /5	**2.60 /< 0.001**	**4.53** /-16.8, 0.0785	3.327 /< 0.001	**119.62 / 118.16**
75-25-400 /6	**2.77 /< 0.001**	5.22 /**4.76**, 0.436	**2.705 /< 0.001**	**122.99 / 121.05**
75-25-400 /7	**3.47 /< 0.001**	**4.33** /-1.57, 0.908	**2.509 /< 0.001**	**126.92 / 103.65**
80-20-400 /0	**2.69 /< 0.001**	**4.95** /**1.82**, 0.807	3.056 /< 0.001	**138.04 / 134.54**
80-20-400 /1	**3.33 /< 0.001**	5.13 /**7.66**, 0.315	4.201 /< 0.001	**101.33 / 94.90**
80-20-400 /2	**2.23 /< 0.001**	5.05 /-7.18, 0.271	**2.207 /< 0.001**	**123.56 / 121.86**
80-20-400 /3	**3.04 /< 0.001**	**4.62** /**11.40**, 0.21	3.179 /< 0.001	115.33 / 116.58
80-20-400 /4	**2.75 /< 0.001**	**4.85** /-5.03, 0.444	**2.352 /< 0.001**	**127.97 / 100.58**
80-20-400 /5	**3.86 /< 0.001**	**4.84** /**10.20**, 0.364	**2.557 /< 0.001**	**135.51 / 100.08**
80-20-400 /6	**3.04 /< 0.001**	**4.75** /-2.02, 0.825	**2.423 /< 0.001**	**126.66 / 124.69**

Composition/ Seed	Excess Kurtosis/ KS Test Gaussian p-Value	Power-Law Alpha/ Log-Likelihood-Ratio Power-Law - Exponential, p-Value	Excess Kurtosis Residuals / KS Test Gaussian, p-Value	Position Maximum Positive/Negative
80-20-400 /7	**3.00 /< 0.001**	**4.61** /-0.266, 0.978	3.435 /< 0.001	111.29 / 111.39
85-15-400 /0	**3.14 /< 0.001**	5.04 /**9.04**, 0.16	**3.053 /< 0.001**	**128.12 / 124.96**
85-15-400 /1	**2.98 /< 0.001**	**4.89** /**4.64**, 0.626	4.785 /< 0.001	104.41 / 105.91
85-15-400 /2	**2.25 /< 0.001**	5.40 /-4.72, 0.408	**2.141 /< 0.001**	**123.23 / 123.12**
85-15-400 /3	**2.81 /< 0.001**	5.25 /**8.55**, 0.134	**2.492 /< 0.001**	112.63 / 115.18
85-15-400 /4	**2.84 /< 0.001**	**4.73** /**1.48**, 0.881	**2.421 /< 0.001**	**102.04 / 97.56**
85-15-400 /5	**2.89 /< 0.001**	**4.92** /**6.24**, 0.382	3.636 /< 0.001	**142.36 / 104.45**
85-15-400 /6	**2.45 /< 0.001**	**4.55** /-3.75, 0.772	**2.094 /< 0.001**	**128.08 / 105.50**
85-15-400 /7	**3.24 /< 0.001**	**4.59** /**13.60**, 0.182	3.251 /< 0.001	**126.15 / 104.07**
90-10-400 /0	**2.66 /< 0.001**	**4.51** /-4.95, 0.673	2.757 /< 0.001	130.26 / 130.54
90-10-400 /1	**3.32 /< 0.001**	5.08 /**6.29**, 0.383	5.077 /< 0.001	**102.70 / 97.67**
90-10-400 /2	**2.43 /< 0.001**	5.80 /-1.32, 0.647	**2.098 /< 0.001**	**151.52 / 104.97**
90-10-400 /3	**3.03 /< 0.001**	**4.96** /**13.70**, 0.057	**2.655 /< 0.001**	**121.08 / 117.51**
90-10-400 /4	**3.28 /< 0.001**	**4.69** /**10.40**, 0.273	**2.559 /< 0.001**	**132.99 / 121.79**
90-10-400 /5	**3.02 /< 0.001**	5.29 /**5.47**, 0.236	3.588 /< 0.001	104.25 / 106.55
90-10-400 /6	**3.58 /< 0.001**	5.10 /**9.46**, 0.106	**2.715 /< 0.001**	**109.47 / 106.96**
90-10-400 /7	**2.82 /< 0.001**	**4.76** /-16.9, **0.0139**	3.016 /< 0.001	**133.07 / 132.36**
95-5-400 /0	**2.54 /< 0.001**	**4.73** /-5.87, 0.534	2.605 /< 0.001	**149.77 / 140.66**
95-5-400 /1	**2.46 /< 0.001**	5.06 /-7.24, 0.266	**1.992 /< 0.001**	**104.22 / 101.29**
95-5-400 /2	**2.88 /< 0.001**	5.04 /**15.1, 0.0413**	**2.434 /< 0.001**	**138.42 / 137.68**
95-5-400 /3	**2.89 /< 0.001**	5.16 /**9.63**, 0.0869	3.703 /< 0.001	113.32 / 116.54
95-5-400 /4	**3.11 /< 0.001**	**4.70** /-1.52, 0.86	3.364 /< 0.001	**108.03 / 105.12**
95-5-400 /5	**4.51 /< 0.001**	**4.86** /**16.10**, 0.122	7.940 /< 0.001	**139.31 / 104.05**
95-5-400 /6	**2.71 /< 0.001**	5.55 /**3.67**, 0.344	**2.304 /< 0.001**	107.34 / 108.53
95-5-400 /7	**2.80 /< 0.001**	5.81 /-0.501, 0.875	2.808 /< 0.001	**131.88 / 114.75**
100-0-400 /0	**2.95 /< 0.001**	5.38 /**2.86**, 0.61	**2.136 /< 0.001**	**136.74 / 135.70**
100-0-400 /1	**2.95 /< 0.001**	**4.67** /-9.46, 0.251	**2.929 /< 0.001**	**103.47 / 92.23**
100-0-400 /2	**2.41 /< 0.001**	5.05 /**0.483**, 0.944	2.714 /< 0.001	**142.56 / 141.30**
100-0-400 /3	**2.69 /< 0.001**	**4.80** /**1.89**, 0.824	2.715 /< 0.001	**131.85 / 98.83**
100-0-400 /4	**2.65 /< 0.001**	**4.84** /-6.35, 0.425	**2.293 /< 0.001**	**126.17 / 102.34**
100-0-400 /5	**3.18 /< 0.001**	**4.92** /**11.30**, 0.151	**3.170 /< 0.001**	**140.87 / 102.43**
100-0-400 /6	**3.08 /< 0.001**	5.17 /**8.90**, 0.167	3.406 /< 0.001	**115.29 / 114.49**
100-0-400 /7	**3.22 /< 0.001**	**4.70** /**4.13**, 0.62	**2.976 /< 0.001**	**119.23 / 119.19**

C.2 Supplemental Graphical Material

Figure C.1.: Development of portfolio weights of active and passive agents for the composition 90-210-200, as well as the fundamental price deviation by simulation round.



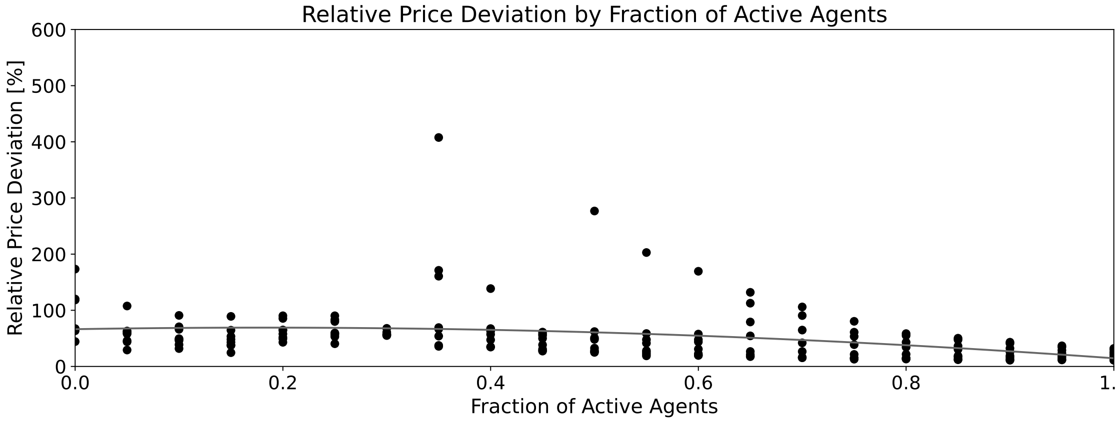


Figure C.2.: Relative fundamental price deviation by fraction of active investors (relative to all active and passive investors) detailed over all random seeds in the market setting with 20 percent random investors.

C.3 Supplemental Text

C.3.1 Derivation of the standard deviation of 3-month target price forecasting errors of active investors

Geary (1935) shows that the ratio of mean absolute deviation MAD and standard deviation σ is equal to $\sqrt{\frac{2}{\pi}}$ under a normal distribution. Equivalently, the standard deviation of a normal distribution can be described as:

$$\sigma = \frac{MAD}{\sqrt{\frac{2}{\pi}}}, \tag{C.1}$$

with

$$MAD = \frac{1}{n}\sum_{i=1}^{n} \left|X_i - \bar{X}\right|, \tag{C.2}$$

where X_i denotes the ith value in a set of n values and $\bar{X}$ denotes the measure of central tendency of that set. Following Bilinski et al. (2013), we define the absolute target price forecasting error $aTPE$ as follows:

$$aTPE = \frac{|TP - P_{12}|}{P_s}, \tag{C.3}$$

where TP denotes the target price, P_{12} the actual stock price after 12 months and P_s the stock price at the time of forecasting. By setting TP as the average of all J agents' forecasts for a given single stock the equation for the $aTPE$ changes to the following:

$$aTPE = \frac{1}{P_s} * \frac{1}{J} \sum_{j=1}^{J} |TP_j - P_{12}| \,. \tag{C.4}$$

As the individual forecasts in our setting are unbiased, the actual stock price after 12 months (P_{12}) is equivalent to the measure of central tendency of the individual forecasts. Therefore, inserting Equation C.2 in Equation C.4 gives:

$$aTPE = \frac{1}{P_s} * MAD. \tag{C.5}$$

$aTPE$ is an exogenous factor in our model, therefore, Equation C.5 is rearranged to

$$MAD = aTPE * P_s. \tag{C.6}$$

Inserting Equation C.6 into Equation C.1 gives:

$$\sigma = \frac{aTPE * P_s}{\sqrt{\frac{2}{\pi}}}. \tag{C.7}$$

Assuming a Brownian Motion allows us to calculate a 3-month equivalent of this 1-year standard deviation by dividing it by $\sqrt{4}$, which finally results in:

$$\sigma = \frac{aTPE * P_s}{\sqrt{\frac{2}{\pi}} * \sqrt{4}}. \tag{C.8}$$

BIBLIOGRAPHY

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., and Isard, M. (2016). Tensorflow: A system for large-scale machine learning. In *Proceedings of 12th USENIX Symposium on Operating Systems Design and Implementation*, pages 265–283.

Adam-Müller, A. F. A. and Panaretou, A. (2009). Risk management with options and futures under liquidity risk. *Journal of Futures Markets: Futures, Options, and Other Derivative Products*, 29(4):297–318.

Alessandretti, L., ElBahrawy, A., Aiello, L. M., and Baronchelli, A. (2018). Anticipating cryptocurrency prices using machine learning. *Complexity*, 2018.

Almeida, J., Tata, S., Moser, A., and Smit, V. (2015). Bitcoin prediciton using ANN. *Neural networks*, 7:1–12.

Amjad, M. and Shah, D. (2017). Trading bitcoin and online time series prediction. In *NIPS 2016 Proceedings*, pages 1–15.

Anadu, K., Kruttli, M. S., McCabe, P. E., Osambela, E., and Shin, C. (2019). The shift from active to passive investing: Potential risks to financial stability? *SSRN*.

Andrade, E. B., Odean, T., and Lin, S. (2016). Bubbling with excitement: an experiment. *Review of Finance*, 20(2):447–466.

Appel, I. R., Gormley, T. A., and Keim, D. B. (2019). Standing on the Shoulders of Giants: The Effect of Passive Investors on Activism. *The Review of Financial Studies*, 32(7):2720–2774.

Arnott, R., Harvey, C. R., and Markowitz, H. (2019). A backtesting protocol in the era of machine learning. *The Journal of Financial Data Science*, 1(1):64–74.

Arrieta, A. B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-López, S., Molina, D., Benjamins, R., and Others (2020). Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information fusion*, 58:82–115.

Ashraf, B. N. (2021). Stock markets' reaction to Covid-19: Moderating role of national culture. *Finance Research Letters*, 41(August 2020):101857.

Atsalakis, G. S., Atsalaki, I. G., Pasiouras, F., and Zopounidis, C. (2019). Bitcoin price forecasting with neuro-fuzzy techniques. *European Journal of Operational Research*, 276(2):770–780.

Atsalakis, G. S. and Valavanis, K. P. (2009). Forecasting stock market short-term trends using a neuro-fuzzy based methodology. *Expert systems with Applications*, 36(7):10696–10707.

Aysan, A. F., Demir, E., Gozgor, G., and Lau, C. K. M. (2019). Effects of the geopolitical risks on Bitcoin returns and volatility. *Research in International Business and Finance*, 47:511–518.

Aysan, A. F., Demirtaş, H. B., and Saraç, M. (2021). The Ascent of Bitcoin: Bibliometric Analysis of Bitcoin Research. *Journal of Risk and Financial Management*, 14(9):427.

Bai, S., Kolter, J. Z., and Koltun, V. (2018). An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. https://arxiv.org/pdf/1803.01271. Accessed 2022-07-21.

Baker, S. R., Bloom, N., Davis, S. J., and Kost, K. J. (2019). Policy news and stock market volatility. Technical report, National Bureau of Economic Research.

Barandela, R., Valdovinos, R. M., Sánchez, J. S., and Ferri, F. J. (2004). The Imbalanced Training Sample Problem: Under or over Sampling? In *Lecture Notes in Computer Science*, pages 806–814. Springer.

BarclayHedge (2018). BarclayHedge Survey: Majority of Hedge Fund Pros Use AI/Machine Learning in Investment Strategies.

https://www.barclayhedge.com/insider/barclayhedge-survey-majority-of-hedge-fund-pros-use-ai-machine-learning-in-investment-strategies. Accessed 2022-07-21.

Bariviera, A. F. (2017). The inefficiency of Bitcoin revisited: A dynamic approach. *Economics Letters*, 161:1–4.

Beck, R., Avital, M., Rossi, M., and Thatcher, J. B. (2017). Blockchain Technology in Business and Information Systems Research. *Business & Information Systems Engineering*, 59(6):381–384.

Belasco, E., Finke, M., and Nanigian, D. (2012). The impact of passive investing on corporate valuations. *Managerial Finance*, 38(11):1067–1084.

Bengio, Y. et al. (2009). Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, 2(1):1–127.

Benhammada, S., Amblard, F., and Chikhi, S. (2017). An asynchronous double auction market to study the formation of financial bubbles and crashes. *New Generation Computing*, 35(2):129–156.

Berente, N., Gu, B., Recker, J., and Santhanam, R. (2021). Managing artificial intelligence. *MIS quarterly*, 45(3):1433–1450.

Bertella, M. A., Pires, F. R., Feng, L., and Stanley, H. E. (2014). Confidence and the stock market: An agent-based approach. *PloS one 9(1)*, 9(1):1–9.

Best, M. J. and Grauer, R. R. (1991). On the sensitivity of mean-variance-efficient portfolios to changes in asset means: some analytical and computational results. *The Review of Financial Studies*, 4(2):315–342.

Betancourt, C. and Chen, W.-H. (2021). Reinforcement learning with self-attention networks for cryptocurrency trading. *Applied Sciences*, 11(16):7377.

Biais, B., Bisiere, C., Bouvard, M., Casamatta, C., and Menkveld, A. J. (2020). Equilibrium Bitcoin Pricing. *Working Paper*.

Bianchi, D. (2020). Cryptocurrencies as an asset class? An empirical assessment. *The Journal of Alternative Investments*, 23(2):162–179.

Bianchi, D., Büchner, M., and Tamoni, A. (2020). Bond Risk Premiums with Machine Learning. *The Review of Financial Studies*, 34(2):1046–1089.

Bilinski, P., Lyssimachou, D., and Walker, M. (2013). Target price accuracy: International evidence. *The accounting review*, 88(3):825–851.

Bird, S., Klein, E., and Loper, E. (2009). *Natural language processing with Python: analyzing text with the natural language toolkit*. O'Reilly Media, Inc.

Blitz, D. (2014). Invited Editorial Comment: The Dark Side of Passive Investing. *The Journal of Portfolio Management*, 41(1):1–4.

Board of Governors of the Federal Reserve System (2022). *3-Month Treasury Bill Secondary Market Rate, Discount Basis [DTB3]*.

Bodie, Z., Kane, A., and Marcus, A. J. (2018). *Investments*. McGraw Hill Education.

Böhme, R., Christin, N., Edelman, B., and Moore, T. (2015). Bitcoin: Economics, technology, and governance. *Journal of Economic Perspectives*, 29(2):213–238.

Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics*, 31(3):307–327.

Bolt, W. and Van Oordt, M. R. C. (2020). On the value of virtual currencies. *Journal of Money, Credit and Banking*, 52(4):835–862.

Bonini, S., Zanetti, L., Bianchini, R., and Salvi, A. (2010). Target price accuracy in equity research. *Journal of Business Finance & Accounting*, 37(9-10):1177–1217.

Borio, C. (2020). The Covid-19 economic crisis: Dangerously unique. *Business Economics*, 55(4):181–190.

Borovykh, A., Bohte, S., and Oosterlee, C. W. (2018). Conditional time series forecasting with convolutional neural networks. *Journal of Computational Finance*, Forthcoming.

Bradley, B. O. and Taqqu, M. S. (2003). *Financial Risk and Heavy Tails*. Woodhead Publishing Limited.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Breiman, L., Friedman, J. H., Olshen, R., and Stone, C. J. (1984). *Classification and Regression Trees*. Routledge.

Burniske, C. and White, A. (2017). Bitcoin: Ringing the bell for a new asset class. https://research.ark-invest.com/hubfs/$1_{D}ownload_{F}iles_{A}RK - Invest/White_{P}apers/Bitcoin - Ringing - The - Bell - For - A - New - Asset - Class.pdf.Accessed2022 - 07 - 21$.

Campbell, S. D. and Sharpe, S. A. (2009). Anchoring bias in consensus forecasts and its effect on market prices. *Journal of Financial and Quantitative Analysis*, 44(2):369–390.

Carhart, M. M. (1997). On persistence in mutual fund performance. *The Journal of Finance*, 52(1):57–82.

Cepoi, C.-o. (2020). Asymmetric dependence between stock market returns and news during COVID-19 financial turmoil. *Finance Research Letters*, 36.

Cerda, G. C., Reutter, J., and Maza, D. L. (2019). Bitcoin Price Prediction Through Opinion Mining. In *Proceedings of 2019 World Wide Web Conference*, pages 755–762.

Chen, M. A., Wu, Q., and Yang, B. (2019). How Valuable Is FinTech Innovation? *The Review of Financial Studies*, 32(5):2062–2106.

Chen, T. and Guestrin, C. (2016). XGBoost. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794. ACM.

Chen, Z., Li, C., and Sun, W. (2020). Bitcoin price prediction using machine learning: An approach to sample dimension engineering. *Journal of Computational and Applied Mathematics*, 365:Article 112395.

Chollet, F. (2015). keras. https://keras.io/. Accessed 2022-07-21.

Chopra, V. K. and Ziemba, W. T. (1993). The Effect of Errors in Means, Variances, and Covariances on Optimal Portfolio Choice. *The Journal of Portfolio Management*, 19(2):6–11.

Chung, J., Gülçehre, Ç., Cho, K., and Bengio, Y. (2014). Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *Working Paper*.

Ciaian, P., Rajcaniova, M., and Kancs, D. (2016). The economics of BitCoin price formation. *Applied Economics*, 48(19):1799–1815.

Cincotti, S., Focardi, S. M., Marchesi, M., and Raberto, M. (2003). Who wins? Study of long-run trader survival in an artificial stock market. *Physica A: Statistical Mechanics and its Applications*, 324(1-2):227–233.

Ciner, C. (2021). Stock return predictability in the time of COVID-19. *Finance Research Letters*, 38(May 2020):101705.

Clarkson, P., Guedes, J., and Thompson, R. (1996). On the diversification, observability, and measurement of estimation risk. *Journal of Financial and Quantitative analysis*, 31(1):69–84.

Cocco, L., Concas, G., and Marchesi, M. (2017). Using an artificial financial market for studying a cryptocurrency market. *Journal of Economic Interaction and Coordination*, 12(2):345–365.

CoinGecko (2022). Methodology. https://www.coingecko.com/en/methodology. Accessed 2022-07-21.

Coinmarketcap (2020). Coinmarketcap. https://coinmarketcap.com/. Accessed 2020-12-21.

Coinmarketcap (2022). Coinmarketcap. https://coinmarketcap.com/. Accessed 2022-07-21.

Cont, R. (2001). Empirical properties of asset returns: Stylized facts and statistical issues. *Quantitative Finance*, 1(2):223–236.

Corbet, S., Hou, Y., Hu, Y., Lucey, B., and Oxley, L. (2021). Aye Corona! The contagion effects of being named Corona during the COVID-19 pandemic. *Finance Research Letters*, 38(May 2020):101591.

CORE (2019). CORE. https://core.ac.uk/. Accessed 2022-03-21.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.

Cremers, K. J. M. and Petajisto, A. (2009). How Active Is Your Fund Manager? A New Measure That Predicts Performance. *The Review of Financial Studies*, 22(9).

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314.

Daniel, K., Hirshleifer, D., and Subrahmanyam, A. (1998). Investor psychology and security market under-and overreactions. *the Journal of Finance*, 53(6):1839–1885.

Defazio, A., Bach, F., and Lacoste-Julien, S. (2014). SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. *Advances in neural information processing systems*, 27.

Degrave, J., Felici, F., Buchli, J., Neunert, M., Tracey, B., Carpanese, F., Ewalds, T., Hafner, R., Abdolmaleki, A., de Las Casas, D., et al. (2022). Magnetic control of tokamak plasmas through deep reinforcement learning. *Nature*, 602(7897):414–419.

Demir, E., Gozgor, G., Lau, C. K. M., and Vigne, S. A. (2018). Does economic policy uncertainty predict the Bitcoin returns? An empirical investigation. *Finance Research Letters*, 26:145–149.

Detzel, A., Liu, H., Strauss, J., Zhou, G., and Zhu, Y. (2020). Learning and predictability via technical analysis: Evidence from bitcoin and stocks with hard-to-value fundamentals. *Financial Management*, 50:107–137.

Diamond, S. and Boyd, S. (2016). CVXPY: A Python-embedded modeling language for convex optimization. *The Journal of Machine Learning Research*, 17(1):2909–2913.

Diebold, F. and Mariano, R. (1994). Comparing Predictive Accuracy. *Journal of Business & Economic Statistics*, 20(1):134–144.

Domahidi, A., Chu, E., and Boyd, S. (2013). ECOS: An SOCP solver for embedded systems. In *2013 European Control Conference (ECC)*, pages 3071–3076. IEEE.

Došilović, F. K., Brčić, M., and Hlupić, N. (2018). Explainable artificial intelligence: A survey. In *2018 41st International convention on information and communication technology, electronics and microelectronics (MIPRO)*, pages 210–215. IEEE.

Drucker, H., Burges, C. J. C., Kaufman, L., Smola, A. J., and Vapnik, V. (1997). Support vector regression machines. In *Advances in neural information processing systems*, volume 9, pages 155–161.

Dutta, A., Kumar, S., and Basu, M. (2020). A Gated Recurrent Unit Approach to Bitcoin Price Prediction. *Journal of Risk and Financial Management*, 13(2):Article 23.

Dyhrberg, A. H. (2016). Bitcoin, gold and the dollar – A GARCH volatility analysis. *Finance Research Letters*, 16:85–92.

Eck, D. and Schmidhuber, J. (2002). Finding temporal structure in music: blues improvisation with LSTM recurrent networks. In *Proceedings of the 12th IEEE Workshop on Neural Networks for Signal Processing*, pages 747–756. IEEE.

Economic Policy Uncertainty (2022). Daily Infectious Disease Equity Market Volatility Tracker. $https://www.policyuncertainty.com/infectious_EMV.html$. Accessed 2022-05-22.

Eichhorn, D., Gupta, F., and Stubbs, E. (1998). Using constraints to improve the robustness of asset allocation. *Journal of Portfolio Management*, 24(3):41.

Extance, A. (2015). Bitcoin and beyond. *Nature*, 526(7571):21.

Fama, E. F. (1970). Efficient Capital Markets : A Review of Theory and Empirical Work. *The Journal of Finance*, 25(2):383–417.

Fama, E. F. (1991). Efficient Capital Markets: II. *The Journal of Finance*, 46(5):1575.

Fama, E. F. (1997). Market Efficiency, Long-Term Returns, and Behavioral Finance. *Journal of Financial Economics*, 49(3):283–306.

Fama, E. F. and French, K. R. (1993). Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics*, 33(1):3–56.

Fama, E. F. and French, K. R. (2007). Dissecting Anomalies. *The Journal of Finance*, 63(4):1653–1678.

Fama, E. F. and French, K. R. (2010). Luck versus Skill in the Cross-Section of Mutual Fund Returns. *Journal of Finance*, 65(5):1915–1947.

Fama, E. F. and MacBeth, J. D. (1973). Risk, return, and equilibrium: Empirical tests. *Journal of political economy*, 81(3):607–636.

Farmer, J. D. and Joshi, S. (2002). The price dynamics of common trading strategies. *Journal of Economic Behavior & Organization*, 49(2):149–171.

Fauci, A. S., Lane, H. C., and Redfield, R. R. (2020). Covid-19—Navigating the Uncharted. *New England Journal of Medicine*, 382(13).

Feng, G., Giglio, S., and Xiu, D. (2020). Taming the Factor Zoo: A Test of New Factors. *The Journal of Finance*, 75(3):1327–1370.

Fichtner, J., Heemskerk, E. M., and Garcia-Bernardo, J. (2017). Hidden power of the Big Three? Passive index funds, re-concentration of corporate ownership, and new financial risk. *Business and Politics*, 19(2):298–326.

Fil, M. and Kristoufek, L. (2020). Pairs trading in cryptocurrency markets. *IEEE Access*, 8:172644–172651.

Finnoff, W., Hergert, F., and Zimmermann, H. G. (1993). Improving model selection by nonconvergent methods. *Neural Networks*, 6(6):771–783.

Fischer, T. and Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, 270(2):654–669.

Fischer, T., Krauss, C., and Deinert, A. (2019). Statistical Arbitrage in Cryptocurrency Markets. *Journal of Risk and Financial Management*, 12(1):31.

Fisher, A., Rudin, C., and Dominici, F. (2019). All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. *J. Mach. Learn. Res.*, 20(177):1–81.

François-Lavet, V., Henderson, P., Islam, R., Bellemare, M. G., Pineau, J., et al. (2018). An introduction to deep reinforcement learning. *Foundations and Trends in Machine Learning*, 11(3-4):219–354.

French, K. R. (2008). Presidential address: The cost of active investing. *The Journal of Finance*, 63(4):1537–1573.

French, K. R. (2022). Kenneth R. French Data Library. https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html. Accessed 2022-07-21.

Freund, Y. and Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of statistics*, 29(5):1189–1232.

Garleanu, N., Panageas, S., and Yu, J. (2012). Technological growth and asset pricing. *The Journal of Finance*, 67(4):1265–1292.

Geary, R. C. (1935). The Ratio of the Mean Deviation to the Standard Deviation as a Test of Normality Author. *Biometrika*, 27(3/4):310–332.

Georgoula, I., Pournarakis, D., Bilanakos, C., Sotiropoulos, D., and Giaglis, G. M. (2015). Using Time-Series and Sentiment Analysis to Detect the Determinants of Bitcoin Prices. *Working Paper.*

Gers, F. (1999). Learning to forget: continual prediction with LSTM. In *Proceedings of 9th International Conference on Artificial Neural*, 10, pages 850–855. IEEE.

Gers, F. A., Schraudolph, N. N., and Schmidhuber, J. (2002). Learning precise timing with LSTM recurrent networks. *Journal of Machine Learning Research*, 3(1):115–143.

Giudici, P. and Abu-Hashish, I. (2019). What determines bitcoin exchange prices? A network VAR approach. *Finance Research Letters*, 28:309–318.

Glaser, F., Zimmermann, K., Haferkorn, M., Weber, M. C., and Siering, M. (2014). Bitcoin-asset or currency? Revealing users' hidden intentions. In *Proceedings of 22nd European Conference on Information Systems*, pages 1–15.

Goldstein, I., Spatt, C. S., and Ye, M. (2021). Big Data in Finance. *The Review of Financial Studies*, 34(7):3213–3225.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*. MIT press.

Google (2020). Google Cloud Natural Language API. https://cloud.google.com/natural-language/docs. Accessed 2022-07-21.

Goykhman, M. (2017). Wealth dynamics in a sentiment-driven market. *Physica A: Statistical Mechanics and Its Applications*, 488:132–148.

Grauer, R. R. and Shen, F. C. (2000). Do constraints improve portfolio performance? *Journal of banking & finance*, 24(8):1253–1274.

Graves, A. (2013). Generating sequences with recurrent neural networks. *Working Paper*.

Graves, A., Fernández, S., Gomez, F., and Schmidhuber, J. (2006). Connectionist temporal classification. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376. ACM Press.

Graves, A. and Jaitly, N. (2014). Towards end-to-end speech recognition with recurrent neural networks. In *Proceedings of International Conference on Machine Learning*, pages 1764–1772.

Graves, A., Liwicki, M., Bunke, H., Schmidhuber, J., and Fernández, S. (2008). Unconstrained on-line handwriting recognition with recurrent neural networks. In *Proceedings of Advances in Neural Information Processing Systems Conference*, pages 577–584.

Graves, A., Liwicki, M., Fernandez, S., Bertolami, R., Bunke, H., and Schmidhuber, J. (2009). A Novel Connectionist System for Unconstrained Handwriting Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(5):855–868.

Graves, A., Mohamed, A.-r., and Hinton, G. (2013). Speech recognition with deep recurrent neural networks. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6645–6649. IEEE.

Graves, A. and Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 18(5-6):602–610.

Greaves, A. and Au, B. (2015). Using the bitcoin transaction graph to predict the price of bitcoin. *Working Paper*.

Green, J., Hand, J. R. M., and Zhang, F. (2012). The Supraview of Return Predictive Signals. *Review of Accounting Studies*, 18(3):692–730.

Gronauer, S. and Diepold, K. (2022). Multi-agent deep reinforcement learning: a survey. *Artificial Intelligence Review*, 55(2):895–943.

Grossman, S. J. and Stiglitz, J. E. (1980). On the impossibility of informationally efficient markets. *The American Economic Review*, 70(3):393–408.

Gu, S., Kelly, B., and Xiu, D. (2020). Empirical Asset Pricing via Machine Learning. *The Review of Financial Studies*, 33(5):2223–2273.

Gu, S., Kelly, B., and Xiu, D. (2021). Autoencoder asset pricing models. *Journal of Econometrics*, 222(1):429–450.

Hammer, B. (2000). On the approximation capability of recurrent neural networks. *Neurocomputing*, 31(1-4):107–123.

Haroon, O. and Rizvi, S. A. R. (2020). COVID-19: Media coverage and financial markets behavior—A sectoral inquiry. *Journal of Behavioral and Experimental Finance*, 27:100343.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., Del Río, J. F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., and Oliphant, T. E. (2020). Array programming with numpy. *Nature*, 585(7825):357–362.

Hegazy, K. and Mumford, S. (2016). Comparative automated bitcoin trading strategies. *Working Paper*.

Hirchoua, B., Ouhbi, B., and Frikh, B. (2021). Deep reinforcement learning based trading agents: Risk curiosity driven learning for financial rules-based policy. *Expert Systems with Applications*, 170:114553.

Ho, T. (1995). Random decision forests. In *Proceedings of 3rd International Conference on Document Analysis and Recognition*, volume 1, pages 278–282. IEEE Comput. Soc. Press.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251–257.

Hornik, K., Stinchcombe, M., and White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366.

Hosmer Jr, D. W., Lemeshow, S., and Sturdivant, R. X. (2013). *Applied logistic regression*, volume 398. John Wiley & Sons.

Hotz-Behofsits, C., Huber, F., and Zörner, T. O. (2018). Predicting crypto-currencies using sparse non-Gaussian state space models. *Journal of Forecasting*, 37(6):627–640.

Hu, A. S., Parlour, C. A., and Rajan, U. (2019). Cryptocurrencies: Stylized facts on a new investible instrument. *Financial Management*, 48(4):1049–1068.

Huang, J.-Z., Huang, W., and Ni, J. (2019). Predicting bitcoin returns using high-dimensional technical indicators. *The Journal of Finance and Data Science*, 5(3):140–155.

Hurst, H. E. (1951). Long-term storage capacity of reservoirs. *Transactions of the American Society of Civil Engineers*, 116(1):770–799.

Ibarz, J., Tan, J., Finn, C., Kalakrishnan, M., Pastor, P., and Levine, S. (2021). How to train your robot with deep reinforcement learning: lessons we have learned. *The International Journal of Robotics Research*, 40(4-5):698–721.

ICI (2020). Trends in the Expenses and Fees of Funds. https://www.ici.org/system/files/private/2021-04/per27-03.pdf. Accessed 2022-07-21.

ICI (2021). Investment Company Fact Book 2021. https://www.ici.org/system/files/2021-05/2021_factbook.pdf. Accessed 2022-07-21.

Jagannathan, R. and Ma, T. (2003). Risk reduction in large portfolios: Why imposing the wrong constraints helps. *The Journal of Finance*, 58(4):1651–1683.

Jain, A., Tripathi, S., Dwivedi, H. D., and Saxena, P. (2018). Forecasting Price of Cryptocurrencies using Tweets Sentiment Analysis. In *Proceedings of 2018 International Conference on Contemporary Computing*, pages 2–4. IEEE.

James K. Jackson (2021). Global Economic Effects of Covid 19. Technical report, Congressional Research Service.

Jang, H. and Lee, J. (2017). An Empirical Study on Modeling and Prediction of Bitcoin Prices with Bayesian Neural Networks Based on Blockchain Information. *IEEE Access*, 6:5427–5437.

Jaquart, P., Dann, D., and Martin, C. (2020a). Machine Learning for Bitcoin Pricing — A Structured Literature Review. In *Proceedings of 15th International Conference on Business Information Systems Engineering*, pages 174–188. GITO Verlag.

Jaquart, P., Dann, D., and Weinhardt, C. (2020b). Using Machine Learning to Predict Short-Term Movements of the Bitcoin Market. In *Proceedings of 10th International Workshop on Enterprise Applications, Markets and Services in the Finance Industry*, pages 21–40. Springer.

Jaquart, P., Dann, D., and Weinhardt, C. (2021). Short-term bitcoin market prediction via machine learning. *The Journal of Finance and data science*, 7:45–66.

Jegadeesh, N. (1990). Evidence of Predictable Behavior of Security Returns. *The Journal of Finance*, 45(3):881–898.

Jegadeesh, N. and Titman, S. (1993). Returns to buying winners and selling losers: Implications for stock market efficiency. *The Journal of Finance*, 48(1):65–91.

Jensen, M. H., Johansen, A., and Simonsen, I. (2003). Inverse statistics in economics: The gain-loss asymmetry. *Physica A: Statistical Mechanics and its Applications*, 324(1-2):338–343.

Johns Hopkins University (2022). COVID-19 Data Repository by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University. https://github.com/CSSEGISandData/COVID-19.

Kaelbling, L. P., Littman, M. L., and Moore, A. W. (1996). Reinforcement learning: A survey. *Journal of artificial intelligence research*, 4:237–285.

Kahneman, D. and Riepe, M. W. (1998). Aspects of investor psychology. *Journal of portfolio management*, 24(4):52—-65.

Kahneman, D. and Tversky, A. (1979). Prospect Theory: An Analysis of Decision under. *Source: Econometrica*, 47(2):263–292.

Kakinaka, S. and Umeno, K. (2022). Cryptocurrency market efficiency in short-and long-term horizons during covid-19: An asymmetric multifractal analysis approach. *Finance Research Letters*, 46:102319.

Karakoyun, E. S. and Cibikdiken, A. O. (2018). Comparison of ARIMA Time Series Model and LSTM Deep Learning Algorithm for Bitcoin Price Forecasting. In *Proceedings of 2018 Multidisciplinary Academic Conference*, pages 171–180.

Karolyi, G. A. and Van Nieuwerburgh, S. (2020). New methods for the cross-section of returns. *The Review of Financial Studies*, 33(5):1879–1890.

Katahira, K., Chen, Y., Hashimoto, G., and Okuda, H. (2019). Development of an agent-based speculation game for higher reproducibility of financial stylized facts. *Physica A: Statistical Mechanics and its Applications*, 524:503–518.

Khaldi, R., El Afia, A., Chiheb, R., and Faizi, R. (2018). Forecasting of Bitcoin Daily Returns with EEMD-ELMAN based Model. In *Proceedings of 2018 International Conference on Learning and Optimization Algorithms: Theory and Applications*, pages 1–6.

Khashanah, K. and Alsulaiman, T. (2017). Connectivity, information jumps, and market stability: an agent-based approach. *Complexity*, 2017(6752086):1–16.

Khuntia, S. and Pattanayak, J. (2018). Adaptive market hypothesis and evolving predictability of bitcoin. *Economics Letters*, 167:26–28.

Kim, Y. B., Kim, J. G., Kim, W., Im, J. H., Kim, T. H., Kang, S. J., and Kim, C. H. (2016). Predicting fluctuations in cryptocurrency transactions based on user comments and replies. *PloS one*, 11(8):1–17.

Kingma, D. P. and Ba, J. (2015). Adam: A Method for Stochastic Optimization. *Proceedings of 3rd International Conference on Learning Representations*.

Köchling, G., Müller, J., and Posch, P. N. (2019). Does the introduction of futures improve the efficiency of Bitcoin? *Finance Research Letters*, 30:367–370.

Kogan, L. and Papanikolaou, D. (2014). Growth opportunities, technology shocks, and asset prices. *The Journal of Finance*, 69(2):675–718.

Kolmogorov, A. (1933). Sulla determinazione empirica di una lgge di distribuzione. *Inst. Ital. Attuari, Giorn.*, 4:83–91.

Krauss, C., Do, X. A., and Huck, N. (2017). Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500. *European Journal of Operational Research*, 259(2):689–702.

Kristoufek, L. and Vosvrda, M. (2019). Cryptocurrencies market efficiency ranking: Not so straightforward. *Physica A: Statistical Mechanics and its Applications*, 531:120853.

Krollner, B., Vanstone, B., and Finnie, G. (2010). Financial time series forecasting with machine learning techniques: A survey. In *Proceedings of European Symposium* *on Artificial Neural Networks: Computational and Machine Learning*, pages 1–7. Springer.

Krückeberg, S. and Scholz, P. (2019). Cryptocurrencies as an asset class. In *Cryptofinance and mechanisms of exchange*, pages 1–28. Springer.

Kubat, M., Matwin, S., and Others (1997). Addressing the curse of imbalanced training sets: one-sided selection. In *Proceedings of the 14th International Conference on MachineLearning*, pages 179–186. Citeseer, Morgan Kaufmann.

Kumar, A. and Garg, G. (2019). Sentiment analysis of multimodal twitter data. *Multimedia Tools and Applications*, 78(17):24103–24119.

Lahmiri, S. and Bekiros, S. (2019). Cryptocurrency forecasting with deep learning chaotic neural networks. *Chaos, Solitons & Fractals*, 118:35–40.

Lapuschkin, S., Wäldchen, S., Binder, A., Montavon, G., Samek, W., and Müller, K.-R. (2019). Unmasking Clever Hans predictors and assessing what machines really learn. *Nature communications*, 10(1):1–8.

Le Tran, V. and Leirvik, T. (2020). Efficiency in the markets of crypto-currencies. *Finance Research Letters*, 35:101382.

Lebaron, B. (2001). A builder's guide to agent-based financial markets. *Quantitative Finance*, 1(2):254–261.

LeCun, Y., Bengio, Y., and others (1995). Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1–14.

Li, W., Chien, F., Kamran, H. W., Aldeehani, T. M., Sadiq, M., Nguyen, V. C., and Taghizadeh-Hesary, F. (2021). The nexus between COVID-19 fear and stock market volatility. *Economic Research-Ekonomska Istrazivanja*, 0(0):1–22.

Liwicki, M., Graves, A., Fernàndez, S., Bunke, H., and Schmidhuber, J. (2007). A novel approach to on-line handwriting recognition based on bidirectional long short-term memory networks. In *Proceedings of 9th International Conference on Document Analysis and Recognition, ICDAR 2007*.

Lo, A. W. (2004). The Adaptive Markets Hypothesis. *The Journal of Portfolio Management*, 30(5):15–29.

Luo, W., Phung, D., Tran, T., Gupta, S., Rana, S., Karmakar, C., Shilton, A., Yearwood, J., Dimitrova, N., Ho, T. B., and others (2016). Guidelines for developing and reporting machine learning predictive models in biomedical research: a multidisciplinary view. *Journal of medical Internet research*, 18(12):e323.

Luong, N. C., Hoang, D. T., Gong, S., Niyato, D., Wang, P., Liang, Y.-C., and Kim, D. I. (2019). Applications of deep reinforcement learning in communications and networking: A survey. *IEEE Communications Surveys & Tutorials*, 21(4):3133–3174.

Madan, I., Saluja, S., and Zhao, A. (2015). Automated bitcoin trading via machine learning algorithms. *Working Paper*.

Maeda, I., DeGraw, D., Kitano, M., Matsushima, H., Sakaji, H., Izumi, K., and Kato, A. (2020). Deep reinforcement learning in agent based financial market simulation. *Journal of Risk and Financial Management*, 13(4):71.

Mäkinen, M., Iosifidis, A., Gabbouj, M., and Kanniainen, J. (2018). Predicting Jump Arrivals in Stock Prices Using Neural Networks with Limit Order Book Data. *Working Paper*.

Mallqui, D. C. and Fernandes, R. A. (2019). Predicting the direction, maximum, minimum and closing prices of daily Bitcoin exchange rate using machine learning techniques. *Applied Soft Computing Journal*, 75:596–606.

Markowitz, H. (1952). Portfolio Selection. *The Journal of Finance*, 7(1):77–91.

Marszk, A. and Lechman, E. (2021). *The Digitalization of Financial Markets*. Routledge.

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, and Others (2015). Tensorflow: Large-scale machine learning on heterogeneous systems. https://www.tensorflow.org/. Accessed 2022-07-21.

Massey, F. J. (1951). The Kolmogorov-Smirnov Test for Goodness of Fit. *Journal of the American Statistical Association*, 46(253):68–78.

Mathieu, P. and Brandouy, O. (2010). A generic architecture for realistic simulations of complex financial dynamics. In *Advances in Practical Applications of Agents and Multiagent Systems*, pages 185–197. Springer.

Mattke, J., Maier, C., Reis, L., and Weitzel, T. (2021). Bitcoin investment: a mixed methods study of investment motivations. *European Journal of Information Systems*, 30(3):261–285.

McKinney, W. (2010a). Data Structures for Statistical Computing in Python. In *Proceedings of 9th Python in Science Conference*, volume 445, pages 56–61.

McKinney, W. (2010b). Data structures for statistical computing in python. In *Proceedings of the 9th Python in Science Conference*, Proceedings of the Python in Science Conference, pages 56–61. SciPy.

McNally, S., Roche, J., and Caton, S. (2018). Predicting the Price of Bitcoin Using Machine Learning. In *Proceedings of 2018 Euromicro International Conference on Parallel, Distributed, and Network-Based Processing*, pages 339–343.

Mishra, N. P., Das, S. S., Yadav, S., Khan, W., Afzal, M., Alarifi, A., Kenawy, E. R., Ansari, M. T., Hasnain, M. S., and Nayak, A. K. (2020). Global impacts of pre- and post-COVID-19 pandemic: Focus on socio-economic consequences. *Sensors International*, 1(July).

Mnif, E. and Jarboui, A. (2021). Covid-19, bitcoin market efficiency, herd behaviour. *Review of Behavioral Finance*, 13(1).

Moiseev, N. A. and Akhmadeev, B. A. (2017). Agent-based simulation of wealth, capital and asset distribution on stock markets. *Journal of Interdisciplinary Economics*, 29(2):176–196.

Montavon, G., Samek, W., and Müller, K.-R. (2018). Methods for interpreting and understanding deep neural networks. *Digital signal processing*, 73:1–15.

Mudassir, M., Bennbaia, S., Unal, D., and Hammoudeh, M. (2020). Time-series forecasting of Bitcoin prices using high-dimensional features: a machine learning approach. *Neural Computing and Applications*, pages 1–15.

Nadarajah, S. and Chu, J. (2017). On the inefficiency of Bitcoin. *Economics Letters*, 150:6–9.

Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. *Working Paper*.

Nakano, M., Takahashi, A., and Takahashi, S. (2018). Bitcoin technical trading with artificial neural network. *Physica A*, 510:587–609.

Narayan, P. K., Phan, D. H. B., and Liu, G. (2021). COVID-19 lockdowns, stimulus packages, travel bans, and stock returns. *Finance Research Letters*, 38(June 2020):101732.

Nicola, M., Alsafi, Z., Sohrabi, C., Kerwan, A., Al-Jabir, A., Iosifidis, C., Agha, M., and Agha, R. (2020). The socio-economic implications of the coronavirus pandemic (COVID-19): A review. *International Journal of Surgery*, 78(March):185–193.

Noda, A. (2021). On the evolution of cryptocurrency market efficiency. *Applied Economics Letters*, 28(6):433–439.

Noussair, C. N. and Tucker, S. (2013). Experimental research on asset pricing. *Journal of Economic Surveys*, 27(3):554–569.

Obaid, K. and Pukthuanthong, K. (2021). A picture is worth a thousand words: Measuring investor sentiment by combining machine learning and photos from news. *Journal of Financial Economics*, 144(1):273–297.

Olah, C. (2015). Understanding lstm networks. https://colah.github.io/posts/2015-08-Understanding-LSTMs/. Accessed: 2020-09-01.

Onali, E. (2020). COVID-19 and Stock Market Volatility. *SSRN Electronic Journal*, pages 1–24.

Ovtchinnikov, A. V. and McConnell, J. J. (2009). Capital market imperfections and the sensitivity of investment to stock prices. *Journal of Financial and Quantitative Analysis*, 44(3):551–578.

Ozili, P. K. and Arun, T. G. (2020). Spillover of COVID-19: impact on the Global Economy. *SSRN*.

Padhan, R. and Prabheesh, K. P. (2021). The economics of COVID-19 pandemic: A survey. *Economic Analysis and Policy*, 70:220–237.

Pagnotta, E. and Buraschi, A. (2018). An equilibrium valuation of bitcoin and decentralized network assets. *Working Paper*.

Pant, D. R., Neupane, P., Poudel, A., Pokhrel, A. K., and Lama, B. K. (2018). Recurrent Neural Network Based Bitcoin Price Prediction by Twitter Sentiment Analysis. In *Proceedings of 2018 IEEE International Conference on Computing, Communication and Security*, pages 128–132. IEEE.

Pedersen, L. H. (2018). Sharpening the arithmetic of active management. *Financial Analysts Journal*, 74(1):21–36.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., and Others (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Petropoulos, F., Apiletti, D., Assimakopoulos, V., Babai, M. Z., Barrow, D. K., Taieb, S. B., Bergmeir, C., Bessa, R. J., Bijak, J., Boylan, J. E., et al. (2022). Forecasting: theory and practice. *International Journal of Forecasting*.

Phaladisailoed, T. and Numnonda, T. (2018). Machine learning models comparison for bitcoin price prediction. In *Proceedings of 2018 International Conference on Information Technology and Electrical Engineering: Smart Technology for Better Society*, pages 506–511. IEEE.

Pham, A. V., Adrian, C., Garg, M., Phang, S. Y., and Truong, C. (2021). State-level COVID-19 outbreak and stock returns. *Finance Research Letters*, 43(September 2020):102002.

Ponta, L. and Cincotti, S. (2018). Traders' networks of interactions and structural properties of financial markets: an agent-based approach. *Complexity*, 2018(9072948):1–9.

Ponta, L., Raberto, M., and Cincotti, S. (2011). A multi-assets artificial stock market with zero-intelligence traders. *EPL (Europhysics Letters)*, 93(2):28002.

Poyser, O. (2019). Exploring the dynamics of Bitcoin's price: a Bayesian structural time series approach. *Eurasian Economic Review*, 9(1):29–60.

Prechelt, L. (2012). Early Stopping — But When? In *Neural Networks: Tricks of the trade*, pages 53–67. Springer.

Qin, N. and Singal, V. (2015). Indexing and Stock Price Efficiency. *Financial Management*, 44(4):875–904.

Raberto, M. and Cincotti, S. (2005). Modeling and simulation of a double auction artificial financial market. *Physica A: Statistical Mechanics and its applications*, 355(1):34–45.

Raberto, M., Cincotti, S., Focardi, S. M., and Marchesi, M. (2003). Traders' long-run wealth in an artificial financial market. *Computational Economics*, 22(2-3):255–272.

Rahman, S., Hemel, J. N., Junayed Ahmed Anta, S., Muhee, H. A., and Uddin, J. (2019). Sentiment analysis using R: An approach to correlate cryptocurrency price fluctuations with change in user sentiment using machine learning. In *Proceedings of 2018 Joint International Conference on Informatics, Electronics and Vision and International Conference on Imaging, Vision and Pattern Recognition*, pages 492–497. IEEE.

Raman, N. and Leidner, J. L. (2019). Financial market data simulation using deep intelligence agents. In *International Conference on Practical Applications of Agents and Multi-Agent Systems*, pages 200–211. Springer.

Rasekhschaffe, K. C. and Jones, R. C. (2019). Machine Learning for Stock Selection. *Financial Analysts Journal*, 75(3):70–88.

Rauchs, M. and Hileman, G. (2017). *Global Cryptocurrency Benchmarking Study*. Number 201704-gcbs in Cambridge Centre for Alternative Finance Reports. Cambridge Centre for Alternative Finance, Cambridge Judge Business School, University of Cambridge.

Repository, O. R. (2019). Open Research Library - Australian National University. https://openresearch-repository.anu.edu.au/. Accessed 2022-02-04.

Ritchie, H., Mathieu, E., Rodés-Guirao, L., Appel, C., Giattino, C., Ortiz-Ospina, E., Hasell, J., Macdonald, B., Beltekian, D., and Roser, M. (2020). Coronavirus pandemic (covid-19). https://ourworldindata.org/coronavirus. Accessed 2022-05-21.

Rumelhart, D. E., Hinton, G. E., Williams, R. J., and others (1988). Learning representations by back-propagating errors. *Nature*, 323:533–536.

Sagiroglu, S. and Sinanc, D. (2013). Big data: A review. In *2013 International Conference on Collaboration Technologies and Systems (CTS)*, pages 42–47.

Salisu, A. A., Akanni, L., and Raheem, I. (2020). The COVID-19 global fear index and the predictability of commodity price returns. *Journal of Behavioral and Experimental Finance*, 27:100383.

Salisu, A. A. and Vo, X. V. (2020). Predicting stock returns in the presence of COVID-19 pandemic: The role of health news. *International Review of Financial Analysis*, 71(June):101546.

Samek, W., Montavon, G., Vedaldi, A., Hansen, L. K., and Müller, K.-R. (2019). *Explainable AI: interpreting, explaining and visualizing deep learning*, volume 11700. Springer Nature.

Sarr, A. and Lybek, T. (2002). Measuring Liquidity in Financial Markets. *IMF Working Papers*, 2002(232).

Schilling, L. and Uhlig, H. (2019). Some simple bitcoin economics. *Journal of Monetary Economics*, 106:16–26.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, 61:85–117.

Schwert, G. W. (2003). Anomalies and market efficiency. In *Handbook of the Economics of Finance*, Handbook of the Economics of Finance, pages 939–974. Elsevier.

Seyffart, J. (2021). Passive likely overtakes active by 2026, earlier if bear market. https://www.bloomberg.com/professional/blog/passive-likely-overtakes-active-by-2026-earlier-if-bear-market/. Accessed 2022-07-21.

Shah, D. and Zhang, K. (2014). Bayesian regression and Bitcoin. In *Proceedings of 2014 Annual Allerton Conference on Communication, Control, and Computing*, pages 409–414. IEEE.

Sharpe, W. F. (1991). The arithmetic of active management. *Financial Analysts Journal*, 47(1):7–9.

Shefrin, H. and Statman, M. (1985). The Disposition to Sell Winners Too Early and Ride Losers Too Long: Theory and Evidence. *The Journal of Finance*, 40(3):777–790.

Shleifer, A. and Vishny, R. (1995). The Limits of Arbitrage. *The Journal of Finance*, 52(1):35–55.

Siddiqui, S. A., Mercier, D., Munir, M., Dengel, A., and Ahmed, S. (2019). Tsviz: Demystification of deep learning models for time-series analysis. *IEEE Access*, 7:67027–67040.

Silverblatt, H. (2019). SP 500® 2018: Global Sales. *Working Paper*. Accessed 2022-07-21.

Sin, E. and Wang, L. (2018). Bitcoin price prediction using ensembles of neural networks. In *Proceedings of 2018 International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery*, pages 666–671.

Smirnov, N. (1948). Table for estimating the goodness of fit of empirical distributions. *The annals of mathematical statistics*, 19(2):279–281.

Smuts, N. (2019). What Drives Cryptocurrency Prices?: An Investigation of Google Trends and Telegram Sentiment. *ACM SIGMETRICS Performance Evaluation Review*, 46(3):131–134.

Spooner, T., Fearnley, J., Savani, R., and Koukorinis, A. (2018). Market making via reinforcement learning. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pages 434–442.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958.

Stambaugh, R. F. (2014). Presidential Address: Investment Noise and Trends. *The Journal of Finance*, 69(4):1415–1453.

Subramaniam, S. and Chakraborty, M. (2021). COVID-19 fear index: does it matter for stock market returns? *Review of Behavioral Finance*, 13(1):40–50.

Sullivan, R. N. and Xiong, J. X. (2012). How index trading increases market vulnerability. *Financial Analysts Journal*, 68(2):70–84.

Sun, X., Liu, M., and Sima, Z. (2018). A novel cryptocurrency price trend forecasting model based on LightGBM. *Finance Research Letters*, 32(101084).

Sushko, V. and Turner, G. (2018). The implications of passive investing for securities markets. *BIS Quarterly Review*, March:113–131.

Symeonidis, S., Effrosynidis, D., and Arampatzis, A. (2018). A comparative evaluation of pre-processing techniques and their interactions for twitter sentiment analysis. *Expert Systems with Applications*, 110(1):298–310.

Takeuchi, L. and Lee, Y.-Y. A. (2013). Applying deep learning to enhance momentum trading strategies in stocks. *Working Paper*.

The Economist (2015). The trust machine. https://www.economist.com/leaders/2015/10/31/the-trust-machine. Accessed 2022-07-21.

Tupinambás, T. M., Leão, R. A., and Lemos, A. P. (2018). Cryptocurrencies transactions advisor using a genetic mamdani-type fuzzy rules based system. In *Proceedings of 2018 IEEE International Conference on Fuzzy Systems*, pages 1–7. IEEE.

Tversky, A. and Kahneman, D. (1974). Judgment under uncertainty: Heuristics and biases: Biases in judgments reveal some heuristics of thinking under uncertainty. *Science*, 185(4157):1124–1131.

University of Pennsylvania (2022). Wharton Research Data Services. https://wrds-www.wharton.upenn.edu/. Accessed 2022-07-21.

Urquhart, A. (2016). The Inefficiency of Bitcoin. *Economics Letters*, 148:80–82.

van der Merwe, A. (2021). A Taxonomy of Cryptocurrencies and Other Digital Assets. *Review of Business*, 41(1).

van der Walt, S., Colbert, S. C., and Varoquaux, G. (2011). The NumPy Array: A Structure for Efficient Numerical Computation. *Computing in Science & Engineering*, 13(2):22–30.

Vanfossan, S., Dagli, C. H., and Kwasa, B. (2020). An Agent-Based Approach to Artificial Stock Market Modeling. *Procedia Computer Science*, 168:161–169.

Vapnik, V. (1995). *The nature of statistical learning theory*. Springer science & business media.

Velavan, T. P. and Meyer, C. G. (2020). The COVID-19 epidemic. *Tropical medicine & international health*, 25(3):278.

VHV (2019). VHB. https://vhbonline.org/en/service/jourqual/vhb-jourqual-3/complete-list-of-the-journals/. Accessed 2020-01-12.

Vidal-Tomás, D. (2022). Which cryptocurrency data sources should scholars use? *International Review of Financial Analysis*, 81:102061.

Vidal-Tomás, D. and Ibañez, A. (2018). Semi-strong efficiency of Bitcoin. *Finance Research Letters*, 27(1):259–265.

Vom Brocke, J., Simons, A., Niehaves, B., Riemer, K., Plattfaut, R., Cleven, A., and others (2009). Reconstructing the giant: on the importance of rigour in documenting the literature search process. In *ECIS 2009 Proceedings*, pages 2206–2217.

Wang, Q. J., Chen, D., and Chang, C. P. (2021). The impact of COVID-19 on stock prices of solar enterprises: A comprehensive evidence based on the government response and confirmed cases. *International Journal of Green Energy*, 18(5):443–456.

Webster, J. and Watson, R. T. (2002). Analyzing the past to prepare for the future: Writing a literature review. *MIS quarterly*, pages xiii–xxiii.

Weinhardt, C. and Gimpel, H. (2007). Market Engineering: An Interdisciplinary Research Challenge. *Dagstuhl Seminar Proceedings*, pages 1–15.

Wilkinson, L. (1999). Statistical methods in psychology journals: Guidelines and explanations. *American psychologist*, 54(8):594.

Wilks, S. (1938). The Large-Sample Distribution of the Likelihood Ratio for Testing Composite Hypotheses. *The Annals of Mathematical Statistics*, 9(1):60–62.

Worldbank (2022). COVID-19 Finance Sector Related Policy Responses. https://datacatalog.worldbank.org/search/dataset/0037999/COVID-19-Finance-Sector-Related-Policy-Responses. Accessed 2022-05-21.

Wu, C. H., Lu, C. C., Ma, Y. F., and Lu, R. S. (2019). A new forecasting framework for bitcoin price with LSTM. In *2019 Proceedings of IEEE International Conference on Data Mining Workshops*, pages 168–175.

Wu, S., He, J., and Li, S. (2018). Effects of fundamentals acquisition and strategy switch on stock price dynamics. *Physica A: Statistical Mechanics and its Applications*, 491:799–809.

Yu, F. and Koltun, V. (2016). Multi-scale context aggregation by dilated convolutions. In *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*.

Zaremba, A., Umutlu, M., and Maydybura, A. (2020). Where have the profits gone? Market efficiency and the disappearing equity anomalies in country and industry returns. *Journal of Banking & Finance*, 121:105966.

Zhang, D., Hu, M., and Ji, Q. (2020). Financial markets under the global pandemic of COVID-19. *Finance Research Letters*, 36:101528.