Politecnico di Torino

Engineering and Management

Y.y. 2024/2025

Graduation Session November 2025

Predicting Stock Market Crashes

A Comparative Analysis of Econometric and Machine Learning Models

Supervisor: Riccardo Calcagno

Candidate: Michele Della Mura

Abstract

Forecasting financial market crashes remains one of the most complex challenges in financial econometrics. While the Efficient Market Hypothesis (EMH) implies the futility of such attempts, the Adaptive Market Hypothesis (AMH) suggests that market efficiency is dynamic and context-dependent, allowing temporary windows of predictability.

This thesis examines the predictability of U.S. stock market crashes by comparing a traditional econometric model (Logistic Regression) with machine learning approaches (Random Forest and LSTM). The feature set integrates technical, macroeconomic, and volatility-based indicators, and the models are evaluated using metrics specifically chosen for imbalanced classification problems, where conventional accuracy would overstate performance and fail to capture the ability to detect rare but critical crash events.

Empirical results show that the LSTM achieves the best balance between precision and recall, yet Logistic Regression remains surprisingly competitive, at times matching or even outperforming more complex models. These findings emphasize that model parsimony retains value in turbulent environments and provide empirical support for the AMH view of financial markets as evolving, adaptive systems. Moreover, the integration of macroeconomic and volatility-based indicators reflects the logic of multifactor asset pricing models, which posit that systematic risk factors underpin market valuations. This perspective complements the AMH by grounding predictability in fundamental, exogenous drivers of returns.

Table of Contents

List of Tables IV

List of Figures V

1 Introduction 1
- 1.1 Background and Motivation: The Systemic Impact of Financial Crises 1
- 1.2 Problem Statement and Research Questions 2
- 1.3 Thesis Contribution and Structure 4

2 Literature Review 5
- 2.1 The Challenge to Prediction: The Efficient Market Hypothesis . . . 5
  - 2.1.1 Factor Models and Systematic Risk Premia 6
- 2.2 Explaining Instability: Speculative Bubbles and Limits to Arbitrage 7
- 2.3 The Adaptive Market Hypothesis 8
- 2.4 Machine Learning in Financial Forecasting 9
  - 2.4.1 Model Parsimony and the Risks of Overfitting 11

3 Data and Methodology 12
- 3.1 Data Acquisition and Temporal Scope 12
- 3.2 Foundational Market Data: The S&P 500 Index 13
- 3.3 Feature Engineering 13
  - 3.3.1 Internal Factors: Technical Indicators 14
  - 3.3.2 External Factors: Macroeconomic and Commodity Data . . 15
  - 3.3.3 Engineered Volatility Features 16
- 3.4 Data Integration and Cleaning 18
  - 3.4.1 Temporal Alignment of Asynchronous Data 18
  - 3.4.2 Computation of Logarithmic Returns 18
  - 3.4.3 Data Cleaning and Outlier Treatment 19
  - 3.4.4 Handling of Missing Data and Implications for Temporal Continuity 19
- 3.5 Descriptive Metrics and Structural Dependencies 20

4 Model Development and Evaluation 24
4.1 Modeling Framework for Imbalanced Data 24
4.1.1 Target Variable Definition 24
4.1.2 Temporal Train-Test Split and Feature Scaling 26
4.1.3 Addressing Class Imbalance with SMOTE 26
4.2 Predictive Models 26
4.2.1 Logistic Regression (LR) 27
4.2.2 Random Forest (RF) 27
4.2.3 Long Short-Term Memory (LSTM) Network 29
4.3 Empirical Results and Analysis 32
4.3.1 Analysis of Model Performance 33
4.3.2 Feature Importance Analysis 34

5 Conclusion 36
5.1 Summary of Findings 36
5.2 Concluding Remarks on the Predictability of Market Crashes 37
5.3 Avenues for Future Research 38

Appendices 39

A Data Preparation Script 40

B Model Development and Evaluation Script 44

Bibliography 47

List of Tables

3.1 Structure of the S&P 500 Daily Data 13
3.2 Descriptive Statistics of Dataset Variables 22

4.1 Model Performance Comparison on the Held-Out Test Set 32
4.2 Selected Coefficients of the Logistic Regression Model 35

List of Figures

2.1 Evolutionary cycle of the Adaptive Market Hypothesis (AMH). . . . 9

3.1 Correlation matrix of the final feature set. Darker colors indicate stronger positive or negative associations between variables. 23

4.1 S&P 500 Daily Logarithmic Returns with Identified Crash Events (2001–2023). The plot displays the daily log returns, with the 5th percentile crash threshold shown as a dashed line. It is important to note the apparent discontinuities in the time series (e.g., during the 2004–2006 period). These are not errors but a direct result of the strict data cleaning protocol detailed in Section 3.4.4, where any day with incomplete data across the entire feature set was removed to ensure analytical integrity. 25

4.2 Architecture of the Random Forest algorithm (Chen et al. (2019)). The original dataset is split into training and test sets. Multiple bootstrap samples are drawn from the training set to train individual decision trees. Their predictions are aggregated via a voting mechanism to produce the final classification result. 29

4.3 Internal architecture of an LSTM cell (Elkaseer et al.(2021)). The forget, input and output gates regulate the flow of information into and out of the cell state (C_t), enabling the network to remember or discard information as needed. 32

4.4 ROC Curves for the evaluated models on the held-out test set. The LSTM network demonstrates marginally the highest overall discriminatory power (AUC), closely followed by Logistic Regression, with Random Forest lagging. 34

Chapter 1

Introduction

1.1 Background and Motivation: The Systemic Impact of Financial Crises

The history of modern financial markets is punctuated by episodes of severe instability, crises and crashes that have exerted profound and lasting effects on the global economy and social welfare. Events such as the dot-com bubble collapse in 2000, the Global Financial Crisis (GFC) of 2008 and the sudden market downturn induced by the COVID-19 pandemic in 2020 serve as stark reminders of how market instability can erode wealth, paralyze credit systems, trigger widespread unemployment and undermine investor and consumer confidence. As extensively documented by Reinhart and Rogoff (2009), financial crises are not historical anomalies but recurring features of the economic landscape, often resulting in protracted periods of stagnation or even a "lost decade" of growth. The GFC, for instance, precipitated the most severe global recession since the Great Depression, leading to systemic bank bailouts, sovereign debt crises in Europe and a global reassessment of financial regulation. The increasing interconnectedness of global markets and the complexity of modern financial instruments have further amplified the speed and virulence with which shocks propagate through the system, a phenomenon known as financial contagion (Kaminsky & Reinhart, 1999). In today's environment, a localized shock can rapidly cascade across asset classes and borders, generating systemic risk. In this high-stakes context, the ability to forecast market instability and particularly stock market crashes, represents one of the most significant and enduring challenges for academics, regulators and investors alike. The advent of the Big Data era, coupled with the development of machine learning (ML) and advanced econometric methods, has opened new and promising frontiers for detecting potential early-warning signals of crises.

A fundamental theoretical challenge is posed by the Efficient Market Hypothesis (EMH) (Fama, 1970), which maintains that prices fully incorporate available information and follow a random walk, rendering systematic prediction virtually impossible. This skepticism has traditionally underpinned the dismissal of models claiming to forecast crises. In contrast, the Adaptive Market Hypothesis (AMH) (Lo, 2004) offers a more flexible view, suggesting that efficiency is not static but evolves as investors adapt to changing environments, creating temporary windows of predictability, particularly during stress regimes. Relatedly, the literature on speculative bubbles highlights how feedback loops, leverage cycles and limits to arbitrage can endogenously generate instability and eventual crashes, further motivating the search for reliable early-warning indicators. This perspective positions the present study within an active debate on whether crises are purely random or instead exhibit detectable precursors.

This theoretical tension is further enriched by insights from modern asset pricing theory. Building on the seminal contributions of Fama and French (1993), an extensive body of multi-factor models has shown that returns can be explained by exposure to systematic, exogenous risk factors. From this viewpoint, a certain degree of predictability, based on macroeconomic conditions, liquidity or market volatility, need not contradict efficiency. Rather, it reflects rational compensation for bearing fundamental risks (Chen, Roll, & Ross, 1986). Accordingly, this thesis positions itself at the intersection of these frameworks: testing whether predictive signals of instability are consistent with (i) the regime-dependent dynamics of the AMH, (ii) the instability-prone features of speculative bubbles and (iii) the priced risk factors central to contemporary asset pricing models.

1.2 Problem Statement and Research Questions

Despite decades of research, traditional econometric models have demonstrated limited efficacy in predicting extreme, non-linear events such as market crashes. This shortcoming stems from a fundamental mismatch between the core assumptions of these models and the empirical reality of financial market behavior. Consider a foundational model like linear regression. It assumes a linear relationship between the independent variables and the dependent variable and, when applied to time-series data, implicitly requires that the underlying relationships remain stable over time, a condition related to stationarity in the statistical properties of the process. However, financial time series data consistently violate these assumptions, exhibiting well-documented "stylized facts":

- **Fat-Tailed Distributions - Leptokurtosis:** Extreme price movements occur with a frequency far greater than predicted by a normal distribution.

This means crashes are not rare statistical outliers but inherent features of markets.

- **Volatility Clustering:** Turbulence in financial markets is not random; periods of high market turbulence tend to be followed by more turbulence and calm periods by more calm. This contradicts the assumption of constant variance.

- **Non-Linear Dependencies and Structural Breaks:** Financial markets exhibit dynamic and often unpredictable relationships among economic variables. These relationships are not fixed over time; rather, they are subject to abrupt changes known as *structural breaks*, which frequently occur during periods of financial distress. Such breaks reflect a fundamental shift in the underlying market dynamics, rendering previously stable correlations invalid.

  Two key mechanisms contribute to these shifts: (i) **Self-reinforcing feedback loops**, where initial market movements trigger reactions that amplify the original trend, for example, falling asset prices may incite panic selling, which accelerates the decline and intensifies market stress; and (ii) **Tipping points**, where markets appear stable until they reach a critical threshold, such as excessive leverage or deteriorating sentiment, beyond which the system transitions abruptly into crisis. These phenomena challenge the assumptions of linearity and stationarity embedded in traditional models, underscoring the need for predictive frameworks capable of capturing sudden regime shifts and complex, non-linear dynamics.

Consequently, a linear model is inherently ill-equipped to capture the essence of a market crash. It cannot simulate the sudden breakdown of normal correlations or the explosive, non-linear dynamics where panic becomes the dominant market force. In essence, the traditional toolkit for prediction is built upon a framework that misunderstands the very nature of financial stress, rendering it inadequate for predicting its most severe manifestations. This dissertation addresses this methodological gap by initially postulating that ML algorithms, which are inherently data-driven and designed to capture complex, non-linear patterns without strong a priori assumptions, can offer a significant improvement in predictive power. However, this study also explores a compelling counter-hypothesis: that during periods of extreme market stress, the signal-to-noise ratio may shift and the underlying dynamics might simplify into more structured, panic-driven behavior. In such regimes, more parsimonious and interpretable models may achieve good performance by capturing the dominant signal without overfitting to the noise characteristic of turbulent markets. To systematically investigate this tension, the thesis seeks to answer the following research questions (RQs):

RQ1: Do machine learning models exhibit superior predictive performance over a traditional econometric model in identifying pre-crash signals in the U.S. stock market?

RQ2: Which categories of predictors (technical, macroeconomic or volatility-based) are most salient in forecasting market downturns? Does the analysis of feature importance reveal a consistent hierarchy of predictors?

RQ3: Are the empirical findings consistent with the predictions of the Adaptive Market Hypothesis (AMH), which suggests dynamic and variable market efficiency, in contrast to the classical Efficient Market Hypothesis (EMH)? Furthermore, to what extent can the predictive power of macroeconomic and volatility factors be interpreted within the framework of modern factor models, which posit that systematic, exogenous risks drive asset valuations? Finally, can the varying success of different model complexities be understood through the evolutionary lens of competition and adaptation, as proposed by the AMH?

1.3 Thesis Contribution and Structure

This study aims to contribute to the existing literature on three distinct levels. Theoretically, by providing fresh empirical evidence in the enduring debate between the EMH and AMH, using modern computational tools to test how market predictability evolves, particularly during periods of stress and by grounding this analysis in the role of systematic risk factors, macroeconomic, volatility and liquidity, that underpin modern asset pricing models. Methodologically, by presenting a rigorous, comparative framework for the application of ML to rare event forecasting in finance and, crucially, by providing a compelling case study on the virtues of model parsimony in high-noise financial environments, challenging the narrative of ever-increasing model complexity. Practically, by offering insights that could inform the development of more effective and robust, data-driven early-warning systems for risk management and tactical asset allocation, where the inclusion of factor-based indicators enhances both interpretability and robustness. The remainder of this thesis is structured as follows. Chapter 2 reviews the relevant theoretical and empirical literature, establishing the intellectual context for the study. Chapter 3 details the data acquisition, feature engineering and preprocessing procedures employed. Chapter 4 describes the model development pipeline, including the handling of class imbalance and the implementation of predictive models, before presenting and analyzing the empirical results. Finally, Chapter 5 concludes the study, summarizing the findings, acknowledging limitations and suggesting promising avenues for future research.

Chapter 2

Literature Review

2.1 The Challenge to Prediction: The Efficient Market Hypothesis

Any attempt to forecast financial markets must first contend with the formidable intellectual edifice of the Efficient Market Hypothesis (EMH), formally articulated by Fama (1970). The EMH posits that asset prices, at any given time, fully reflect all available information. Competition among rational, profit-maximizing investors rapidly eliminates arbitrage opportunities, causing prices to follow a "random walk", that is, price changes are independent and unpredictable, as each new piece of information is instantaneously incorporated into market valuations. In its semi-strong form, the most relevant to this study, the hypothesis asserts that prices incorporate all publicly available information, including historical market data, macroeconomic announcements and firm-specific news. A direct implication is the futility of both technical and fundamental analysis for earning abnormal returns. Within this paradigm, predicting market crashes is, by definition, an impossible task, as any information signaling an impending downturn would be instantaneously incorporated into prices.

The EMH has been a cornerstone of modern finance for half a century, yet it has faced persistent challenges from empirical research documenting numerous market anomalies, observations inconsistent with its predictions. These include the value premium (cheap stocks tend to outperform expensive ones), the size effect (small companies often outperform large ones) and, perhaps most robustly, the momentum effect (stocks that have been rising tend to continue rising). Crucially, any test of the EMH is subject to the "joint hypothesis problem": one simultaneously tests for market efficiency and the validity of the asset pricing model employed. Therefore, an apparent anomaly could indicate either genuine market inefficiency or a misspecified risk model. The EMH thus serves as the essential null hypothesis

against which the predictive validity of the models in this thesis is evaluated.

Importantly, the EMH implies that systematic identification of pre-crash signals should be impossible, making the pursuit of forecasting models inherently a test of whether market anomalies contain exploitable information beyond the random walk paradigm.

2.1.1 Factor Models and Systematic Risk Premia

While the EMH provides an essential null hypothesis, its strictest form has been challenged by persistent evidence of predictability. This did not, however, lead to a wholesale rejection of efficiency but rather to the development of more sophisticated asset pricing models that could account for these anomalies within a rational framework. The most influential shift was the move from the single-factor Capital Asset Pricing Model (CAPM) to multi-factor models.

The cornerstone of this evolution is the three-factor model proposed by Fama and French (1993). They demonstrated that, in addition to market risk (beta), two other factors, firm size (SMB, "Small Minus Big") and book-to-market value (HML, "High Minus Low"), held significant explanatory power over the cross-section of stock returns. Their findings suggested that the higher returns associated with small-cap and value stocks were not market inefficiencies but rather rational compensation or risk premia, for bearing exposure to distinct systematic risks.

The factor-based framework was subsequently expanded to include other variables, such as momentum (the tendency for past winning stocks to continue performing well), as seen in the Carhart (1997) four-factor model. More pertinent to the forecasting of broad market movements, this logic has been extended to macroeconomic and financial variables. Fluctuations in factors such as inflation, industrial production, interest rate term spreads and, crucially, market volatility are now widely considered to represent fundamental, non-diversifiable risks that influence asset valuations (Chen, Roll, & Ross, 1986).

This body of work provides a critical theoretical foundation for the methodology employed in this thesis. The selection of macroeconomic, commodity and volatility-based indicators as predictive features is directly informed by factor-model literature. These variables are not chosen arbitrarily; they are treated as proxies for the underlying, systematic risks that rational investors consider when making valuation decisions. Therefore, finding that these factors have predictive power for market crashes can be interpreted not as a sign of pure irrationality, but as evidence that the build-up of systematic risk precedes major market downturns.

2.2 Explaining Instability: Speculative Bubbles and Limits to Arbitrage

While the Efficient Market Hypothesis provides a powerful null model, its assertion of instantaneous price correction is difficult to reconcile with the historical recurrence of massive asset price bubbles and their subsequent collapses. If markets are truly efficient, how can such profound and persistent mispricing occur? This apparent contradiction is addressed by modern financial theory, which moves beyond explanations of pure irrationality to show how bubbles can emerge from the rational decisions of investors operating under real-world constraints.

This perspective acknowledges that historical episodes of speculative bubbles and subsequent crashes indeed challenge the strong assumption of full market rationality. A bubble exists when the realized asset return over a given future period is more than two standard deviations away from its expected return(Siegel, 2003). While early theories emphasized investor psychology and behavioral biases, the contemporary view seeks to explain these phenomena through models featuring rational agents operating under market frictions.

The limits to arbitrage literature provides a compelling framework for understanding why mispricing can persist. Brunnermeier (2009) argues that even when sophisticated arbitrageurs identify mispricing, they may be unable or unwilling to correct it due to significant risks: fundamental risk (the possibility that fundamentals shift against the arbitrageur), noise trader risk (the risk that irrational traders push prices further from fundamentals, potentially triggering margin calls) and synchronization risk (the risk of acting in isolation). Consequently, as Brunnermeier and Nagel (2004) demonstrated empirically during the dot-com bubble, it can be rational for sophisticated investors to ride the bubble rather than bet against it, thereby fueling its growth and contributing to its eventual collapse. These models show that persistent mispricing need not reflect investor irrationality but can emerge as an equilibrium outcome in markets with real-world frictions. Moreover, the limits to arbitrage framework highlights why bubbles can amplify systemic risk: when leverage and liquidity constraints bind simultaneously, localized mispricing can escalate into large-scale market dislocations, providing fertile ground for severe crashes.

Beyond the limits to arbitrage, the literature on speculative bubbles has identified several mechanisms through which mispricing not only persists but grows to systemic proportions. Brunnermeier emphasize the role of leverage cycles: credit expansion enables investors to bid up asset prices, but when deleveraging occurs, forced fire sales can rapidly deflate bubbles, amplifying market downturns. Similarly, feedback trading, where rising prices attract further demand, creating self-reinforcing dynamics, can push valuations further away from fundamentals

until a critical tipping point is reached. These mechanisms underscore that bubbles are not merely statistical anomalies or psychological artifacts, but endogenous features of modern financial systems.

The bursting of a bubble typically transforms localized overvaluation into systemic instability. When highly leveraged intermediaries are exposed, price corrections spread across balance sheets, liquidity dries up and contagion mechanisms accelerate the crash. From this perspective, bubbles provide a natural explanation for why severe market downturns are recurrent despite the presumption of efficiency. Moreover, they offer fertile ground for predictive modeling: the gradual build-up of imbalances and the dynamics of leverage, feedback loops and liquidity constraints may generate detectable early-warning signals. This aligns with the Adaptive Market Hypothesis, which suggests that market efficiency is regime-dependent and that predictability is most likely to emerge during periods of structural stress.

2.3 The Adaptive Market Hypothesis

Seeking to reconcile the EMH with persistent evidence of behavioral biases and market anomalies, Lo (2004, 2005) introduced the Adaptive Market Hypothesis (AMH). Drawing on principles from evolutionary biology, the AMH conceptualizes financial markets as dynamic ecosystems in which investors exhibit bounded rationality, rely on heuristics and learn from experience. Within this framework, investment strategies behave analogously to biological species, competing for scarce profit opportunities, adapting to environmental pressures and undergoing selection based on performance. Successful strategies attract capital and proliferate, while ineffective ones are gradually eliminated.

The core premise of the AMH is that market efficiency is not a static or binary condition, but rather dynamic and context-dependent. The degree of efficiency varies with the intensity of competition, the structure of the market environment and the adaptive capacity of its participants. For example, in a mature and stable market dominated by institutional investors, informational efficiency may be high. However, following structural disruptions, such as technological innovation, regulatory shifts or macroeconomic shocks, new risks and opportunities may arise that existing strategies are ill-equipped to exploit. These regime changes give rise to transient periods of predictability, often amplified by behavioral phenomena such as herding or overreaction.

This cyclical process is illustrated in Figure 2.1, which depicts the evolutionary loop of the AMH. A quintessential example of this cycle is the Dot-com Bubble. It began with a profound **market regime change**: the advent of the commercial internet. This disruption created **"pockets of inefficiency"** as traditional valuation metrics failed, leading to widespread mispricing of technology stocks. **Adaptive**

strategies, such as momentum investing, emerged to exploit these inefficiencies, fueling the bubble's growth. Eventually, these inefficiencies were **arbitraged away** as companies failed to meet expectations and valuations collapsed, restoring a more efficient market state, until the next disruption, the bubble's burst, initiated a new cycle. This dynamic feedback loop, exemplified by this episode, underscores the non-linear and adaptive nature of financial markets and provides a rationale for the transient, regime-dependent predictability that machine learning models may capture.

Empirical research has provided mixed but insightful evidence supporting the AMH. For instance, studies have shown that the predictive power of technical indicators varies substantially across regimes, consistent with the hypothesis of evolving efficiency (Timmermann, 2008). These findings reinforce the value of interpreting model performance through an adaptive lens.

According to the AMH, market dynamics change with the environment. During stable periods, complex and high-dimensional interactions dominate, making ML models better suited to capturing patterns. However, in pre-crash regimes, market behavior may simplify as a few dominant forces, such as panic, liquidity shocks or flight-to-safety, overwhelm other signals. This can create a more linear structure that parsimonious models like Logistic Regression are able to capture effectively. The results of this thesis can thus be interpreted as empirical evidence of regime-dependent predictability, consistent with the AMH framework.

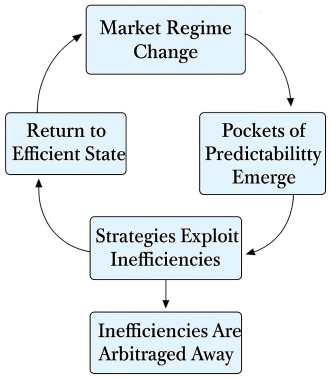

Figure 2.1: Evolutionary cycle of the Adaptive Market Hypothesis (AMH).

2.4 Machine Learning in Financial Forecasting

The application of machine learning (ML) to financial forecasting has grown substantially, moving from academic exploration to mainstream quantitative finance. A landmark study by Gu, Kelly and Xiu (2020) demonstrated that ML methods,

particularly tree-based models and neural networks, systematically outperform traditional econometric models in predicting the cross-section of stock returns. For crash prediction specifically, a growing body of literature has reported promising results. The work of Patel et al. (2024), which serves as a key methodological reference for this thesis, surveyed logistic regression, support vector machines, random forests and LSTM models. Their analysis highlights that random forests offer strong performance in noisy environments and are valued for their robustness, though model effectiveness is highly context-dependent. This insight aligns with the broader literature on ensemble methods in financial forecasting, where tree-based models are often preferred for their ability to handle high-dimensional data and nonlinear relationships. However, as demonstrated in the empirical section of this thesis, model performance can vary significantly depending on the target definition, feature selection and market regime. In particular, while random forests showed competitive results, LSTM models outperformed them in capturing sequential patterns associated with crash dynamics, confirming the importance of temporal structure in volatility prediction. The findings of this thesis further suggest that, when properly optimized and trained on sequential data, LSTM models can outperform both Random Forest and Logistic Regression in detecting crash regimes. This emphasizes the critical role of model architecture and data structure in financial forecasting tasks. Other notable contributions include Fischer and Krauss (2018), who established LSTMs as powerful tools for capturing long-term dependencies in financial time series.

A significant challenge in this domain is the "black box" nature of many ML models. While they may exhibit high predictive accuracy, their decision-making processes can be opaque. This has spurred the growth of Explainable AI (XAI) in finance, which focuses on making machine learning models more transparent and interpretable. In a domain where decisions carry significant financial and regulatory consequences, XAI enables analysts to understand and validate how models arrive at their predictions. A widely used approach is feature importance analysis, which ranks input variables according to their influence on the model's output. Another powerful technique is SHAP (SHapley Additive exPlanations), which assigns each feature a numerical value representing its contribution to a specific prediction. SHAP not only identifies which variables are most influential, but also clarifies whether they increase or decrease the likelihood of a crash. This makes it particularly valuable for interpreting complex models such as Random Forests and LSTMs.

A systematic review by Sonkavde et al. (2023) concludes that a holistic approach integrating technical, macroeconomic and sentiment data yields the most robust models, a principle explicitly adopted in this research.

2.4.1 Model Parsimony and the Risks of Overfitting

While the literature often emphasizes the superior flexibility of ML methods, several scholars have cautioned against their misuse in finance. López de Prado (2018) warns that high-dimensional models are particularly vulnerable to the "curse of dimensionality" and the risk of overfitting in noisy and non-stationary environments like financial markets. This line of reasoning suggests a compelling hypothesis: that in such contexts, simple and parsimonious models may, in fact, outperform more complex algorithms by avoiding spurious correlations and capturing only the most robust relationships. The analysis of the following chapters shows this hypothesis directly, comparing a traditional model against modern algorithms. The competitive performance of the Logistic Regression benchmark, detailed in Section 4.3, provides strong empirical support for this view, demonstrating that parsimony retains significant value. Nonetheless, the superior results achieved by the LSTM model demonstrate that, when appropriately regularized and trained on temporally structured data, complex models can also capture subtle pre-crash dynamics that simpler models may overlook.

Chapter 3

Data and Methodology

This chapter details the empirical framework designed to test the central hypotheses of this thesis. It begins by defining the dataset, including sources, temporal scope and specific variables. Subsequently, it describes the preprocessing and feature engineering pipeline, covering the calculation of technical indicators, temporal alignment of macroeconomic data and computation of logarithmic returns. The chapter concludes by outlining data cleaning and outlier treatment procedures, ensuring the final dataset is robust and suitable for subsequent machine learning modeling.

3.1 Data Acquisition and Temporal Scope

The foundation of any robust quantitative analysis is a high-quality, comprehensive dataset. This study utilizes daily data spanning 22.5 years, from **January 1, 2001** to **June 26, 2023**. This specific temporal window was intentionally selected to encompass a wide array of market conditions and structural regimes, including the post-dot-com bubble adjustments, the 2007–2009 Global Financial Crisis, the 2020 COVID-19 shock and the recent period of monetary tightening and inflationary pressures. The start date of 2001 also marks a point where consistent, high-quality daily data for the chosen features became widely and reliably available.

The heterogeneity of this period is crucial for the research design. It allows for a rigorous assessment of whether predictive signals remain stable across time or exhibit regime-dependent characteristics. Furthermore, it provides a robust testing ground for evaluating whether adaptive models like machine learning algorithms can outperform static econometric baselines, serving as a direct empirical test of the Adaptive Market Hypothesis versus the classical Efficient Market Hypothesis.

The dataset comprises the **S&P 500 stock index** as the primary market benchmark and a curated set of internal and external factors as predictive features,

detailed in the subsequent sections.

3.2 Foundational Market Data: The S&P 500 Index

The core market data for this study is sourced from a proprietary dataset containing daily historical data for the S&P 500 index. This index was selected for its broad representation of the U.S. equity market and its high liquidity. The dataset includes the columns detailed in Table 3.1, which are essential for time-series analysis and technical indicator calculation.

Table 3.1: Structure of the S&P 500 Daily Data

Column Name	**Description**
Date	Trading date in YYYY-MM-DD format
Open	Opening price of the index
High	Highest price reached during the trading session
Low	Lowest price reached during the trading session
Close	Closing price of the index
Volume	Total number of shares traded

The raw data underwent initial cleaning to ensure numeric consistency (e.g., replacing commas with periods) and was filtered to the specified date range to maintain analytical focus and relevance.

3.3 Feature Engineering

The construction of a rich predictive feature set is guided by the complementary theoretical frameworks established in the literature review. To capture a holistic view of market dynamics, we engineered three categories of variables, each grounded in a distinct theoretical perspective:

- **Internal Factors (Technical Indicators):** Motivated by the Adaptive Market Hypothesis (AMH), this category includes indicators derived from the S&P 500's own price and volume data. They are designed to capture endogenous market dynamics, such as momentum, trend strength and money flow, which may arise from the behavioral patterns and adaptive learning of market participants (Lo, 2004).

- **External Factors (Macroeconomic and Commodity Data):** Grounded in multi-factor asset pricing models, these variables serve as proxies for the systematic, exogenous risks that influence fundamental valuations. They include indicators of economic policy uncertainty, financial stress and geopolitical risk, which represent the broader economic environment in which assets are priced (Chen, Roll, & Ross, 1986).

- **Engineered Volatility Features:** Also rooted in factor theory, these features are explicitly designed to capture "volatility risk." Elevated market volatility is not only a symptom of turbulence but is also considered a distinct priced risk factor for which investors demand compensation. These features therefore measure both historical (realized) and conditional (GARCH) volatility to quantify market fear and uncertainty.

3.3.1 Internal Factors: Technical Indicators

A set of five widely recognized technical indicators (TIs) was calculated from the S&P 500's price and volume data using the `ta` Python library. These indicators capture market momentum, volatility, trend strength and money flow dynamics:

- **Bollinger Bands Upper Bound (Boll_ub):** Measures price volatility by plotting two standard deviations above a 20-day simple moving average. Prices approaching or exceeding this band may signal overbought conditions, suggesting potential price reversion.

  > *Real-life Example:* In the weeks leading up to the COVID-19 crash of March 2020, the S&P 500 repeatedly touched or exceeded its upper Bollinger Band. This was a clear statistical warning of an overextended market, vulnerable to a sharp mean-reverting correction.

- **Relative Strength Index (RSI):** A momentum oscillator that measures the speed and change of price movements on a scale of 0 to 100. A value above 70 traditionally indicates overbought conditions, while a value below 30 indicates oversold conditions.

  > *Real-life Example:* During the peak of the Dot-com Bubble in early 2000, the RSI for the NASDAQ index sustained levels above 70 for prolonged periods, signaling extreme overbought momentum. The subsequent reversal and crash brought the RSI down to deeply oversold levels below 30.

- **Stochastic Oscillator (Stoch):** Compares a particular closing price to its price range over a specified period (14 days). Designed to capture momentum

relative to recent highs and lows, with values near 100 suggesting overbought conditions and values near 0 indicating oversold conditions.

> *Real-life Example:* In the volatile ranging market of 2015-2016, the Stochastic oscillator for the S&P 500 frequently cycled from overbought (readings above 80) to oversold (readings below 20), effectively identifying short-term reversal points within a broader sideways trend.

- **Average Directional Index (ADX):** Quantifies the *strength* of a trend regardless of direction. An ADX value above 40 indicates a strong trend, while a value below 20 suggests a weak or non-existent trend.

  > *Real-life Example:* The ADX rose strongly and sustained values well above 40 during the persistent bull market from 2017 to early 2020, confirming the strength of the upward trend. Conversely, it collapsed below 20 during the low-volatility, directionless market of mid-2019, indicating a lack of strong trend.

- **Money Flow Index (MFI):** A volume-weighted relative strength index that incorporates both price and volume data to measure buying and selling pressure. Its inclusion of volume makes it particularly adept at detecting divergences between price action and underlying market conviction.

3.3.2 External Factors: Macroeconomic and Commodity Data

The set of external factors comprises several macroeconomic indicators and international commodity prices, which collectively capture a broad spectrum of economic, financial and geopolitical dynamics known to influence equity markets:

- **Economic Policy Uncertainty (EPU):** This index measures the level of uncertainty regarding future government economic policies. It is constructed by counting the frequency of keywords related to economic uncertainty, policy debates and legislative disputes in major newspapers, combined with data on expiring tax code provisions and professional economic forecasts. High EPU indicates that businesses and investors lack clarity on future taxes, regulations or government spending, leading to delayed investment and increased market volatility. For example, during the U.S. debt ceiling debates, the EPU index typically spikes. This reflects market anxiety about potential government default or abrupt fiscal tightening, which often causes equity sell-offs and flight to safe-haven assets as investors price in heightened risk.

- **Geopolitical Risk (GPR):** This index quantifies the risk of events that disrupt international relations, such as wars, terrorist acts and military conflicts. It is built by systematically scanning global newspapers for articles related to geopolitical tensions and counting the frequency of relevant keywords. Elevated GPR increases global risk aversion, often causing investors to withdraw capital from risky assets and emerging markets, leading to higher market volatility. The initial phase of the Russia-Ukraine war in February 2022 caused a sharp spike in the GPR index. This was immediately followed by a surge in global oil prices, equity market volatility and capital outflows from European markets, demonstrating how geopolitical events transmit risk through financial systems.

- **Financial Stress Index (FSI):** The FSI is a composite indicator that measures strain in the financial system. It aggregates data across five key areas: credit spreads (cost of borrowing), equity market valuations, interbank lending rates, demand for safe assets like Treasuries and market volatility. A high FSI value indicates that the financial system is under stress, with frozen credit markets and falling asset prices, conditions that often precede or accompany market crises.

- **Shadow Short Rate (SSR):** The SSR is an estimated measure of what the central bank's policy rate would be if it could go below zero. When official rates hit zero (the "zero lower bound"), central banks use unconventional tools like quantitative easing. The SSR combines these tools into a single synthetic rate that better reflects the true stance of monetary policy. A falling SSR indicates expansionary policy (stimulus), while a rising SSR indicates policy tightening.

- **Volatility Index (VIX):** Commonly known as the "fear gauge," the VIX measures the market's expectation of 30-day volatility derived from S&P 500 index options. It reflects the price investors are willing to pay for option protection against market downside. A low VIX suggests investor complacency and stable markets, while a high VIX indicates fear, uncertainty and expectations of large price swings.

- **Gold and Oil Prices:** Daily closing prices for gold (GC=F) and crude oil (CL=F) futures were retrieved from Yahoo Finance using the `yfinance` API. Gold is traditionally viewed as a safe-haven asset, while oil prices serve as a key indicator of global economic health.

3.3.3 Engineered Volatility Features

In addition to technical and macroeconomic indicators, two volatility-based features were engineered to explicitly capture market turbulence, which is often a precursor

to crashes.

- **Rolling 30-Day Volatility:** To capture short-term market uncertainty, the 30-day rolling standard deviation of S&P 500 returns was computed as follows:

$$\sigma_{30d,t}^{\text{roll}} = \sqrt{\frac{1}{N-1}\sum_{i=t-N}^{t-1}(r_i - \bar{r})^2} \tag{3.1}$$

  where $N = 30$, r_i is the return at time i and $\bar{r}$ is the average return over the window. This feature, denoted `volatility_30d`, provides a smoothed, backward-looking estimate of recent price fluctuations. It quantifies the *realized* historical volatility, measuring how much returns have actually deviated from their mean over the previous month. This is particularly relevant for crash prediction, as periods of elevated realized volatility often precede major market dislocations.

  It is important to distinguish this measure from the VIX index. While `volatility_30d` is a *backward-looking* measure of *realized* volatility derived from historical returns, the VIX is a *forward-looking* measure of *implied* volatility, representing the market's expectation of future volatility derived from option prices. This fundamental difference in their construction, historical versus anticipatory, means they capture complementary aspects of market risk. Their divergence can be particularly informative; for instance, when the VIX is high relative to recent realized volatility, it often signals that investors are anticipating a significant change in market regime, a condition that frequently precedes crashes.

- **GARCH Conditional Volatility:** To capture the phenomenon of *volatility clustering*, where periods of high market turbulence are often followed by further turbulence, a GARCH model was employed. This model provides a dynamic, forward-looking estimate of latent market risk by incorporating the persistence of past volatility and the impact of recent market shocks.

  The GARCH specification is formally defined by the following system of equations:

$$\begin{aligned} r_t &= \mu + \epsilon_t, \quad \epsilon_t = \sigma_t z_t, \quad z_t \sim N(0,1) \\ \sigma_t^2 &= \omega + \alpha\epsilon_{t-1}^2 + \beta\sigma_{t-1}^2 \end{aligned} \tag{3.2}$$

  where:

  – σ_t^2 is the conditional variance (our measure of volatility at time t).
  – ω is the long-run average volatility.

- α measures the reaction to recent news, i.e., how much yesterday's squared shock (ϵ_{t-1}^2) impacts today's volatility.
- β measures the persistence of volatility, i.e., how much yesterday's forecasted variance (σ_{t-1}^2) influences today's forecast.

The sum ($\alpha + \beta$) quantifies the overall persistence of volatility shocks. A value close to 1 indicates that shocks are highly persistent and will decay slowly, creating the volatility clusters characteristic of financial markets. This makes the GARCH estimate particularly relevant for crash prediction, as it sensitively captures regimes where volatility is self-reinforcing and building towards a potential market dislocation.

3.4 Data Integration and Cleaning

Following the acquisition and engineering of all variables, the final stage of the pipeline involved merging all data sources into a unified analytical dataset and applying rigorous cleaning procedures to ensure statistical robustness.

3.4.1 Temporal Alignment of Asynchronous Data

A key challenge in multivariate financial dataset construction is the differing temporal granularity of variables. Market-based indicators are available only on trading days, while certain macroeconomic indicators are published daily, including weekends and holidays. To create a temporally consistent dataset, all external data sources were merged onto the S&P 500 trading-day calendar using a **backward merge** strategy. For any given trading day t, this method assigns the most recent available value of a macroeconomic indicator from a date t' where $t' \leq t$. This backward-looking approach is critical as it strictly **avoids forward-looking bias**, ensuring that the information set at any given time contains only data that would have been historically available to market participants.

3.4.2 Computation of Logarithmic Returns

To normalize scale across different variables and stabilize their statistical properties, all time-series (except the pre-calculated technical indicators) were converted into **logarithmic returns**. The log return is calculated as the difference in the natural logarithm of a variable between two consecutive periods:

$$r_t = \ln(P_t) - \ln(P_{t-1}) \tag{3.3}$$

This transformation is standard in financial econometrics due to its desirable time-additive properties and its approximation of percentage changes for small returns.

3.4.3 Data Cleaning and Outlier Treatment

The final preprocessing stage involved a multi-step cleaning process to prepare the data for modeling:

1. **Removal of Missing Values:** The initial merging and return calculation steps generated missing values (NaNs). All rows containing any NaN values were removed to ensure a complete-case analysis.

2. **Outlier Clipping:** Financial return series are characterized by heavy tails and extreme observations. To mitigate the influence of potentially erroneous outliers that could distort model training, a clipping procedure based on the **3-sigma rule** was applied to all return-based features. For each return series, any value falling outside three standard deviations from the mean was clipped to the respective upper or lower bound ($\mu \pm 3\sigma$). This method is preferred over outright removal as it retains the data point's temporal position while reducing its excessive leverage.

3. **Final NaN Removal:** A final check and removal of any residual missing values was performed to guarantee the integrity of the dataset fed into the models.

3.4.4 Handling of Missing Data and Implications for Temporal Continuity

A critical phase in constructing a multi-source dataset for financial time-series analysis is the management of missing values (NaNs), which can arise from the calculation of rolling indicators or the merging of series with asynchronous publication calendars. In this study, a conservative and rigorous strategy was adopted: any data row (i.e., a trading day) presenting even a single missing value across the entire feature set was completely removed using the `dropna()` function.

This decision, although it reduces the total number of observations, was deliberately chosen over imputation techniques such as forward-fill (`ffill`) or backward-fill (`bfill`) for several fundamental reasons:

1. **Integrity of the Predictive Signal:** Imputation, particularly in non-stationary financial series, risks introducing artificial and potentially misleading information. Filling a missing value for a volatile macroeconomic indicator with the previous day's value (`ffill`) assumes an inertia that may not exist, thereby diluting the precision of the predictive signal and introducing noise into the model.

2. **Prevention of Look-Ahead Bias:** While `ffill` is causally consistent, methods like `bfill` introduce a clear look-ahead bias by using future information to populate past data, a practice that is unacceptable in a forecasting context.
3. **Model Robustness:** Supplying the machine learning models with only complete and authentic observations ensures that learning occurs on genuine signals, not on statistical artifacts. This approach fosters the development of more robust and generalizable models.

A direct outcome of this methodological choice is the appearance of discontinuities in the final time series, as illustrated in the following chapter. Consequently, the analysis is conducted on a dataset that is temporally discontinuous yet informationally complete, placing greater emphasis on the quality and integrity of each observation than on preserving uninterrupted temporal coverage. This approach is fundamentally different from the subsequent application of the Synthetic Minority Over-sampling Technique (SMOTE) described in Section 4.1.3, which was applied *solely* to the minority crash class within the training set. Unlike imputation, SMOTE does not modify or replace existing records; instead, it synthesizes additional crash-day instances to mitigate severe class imbalance. The test set remained entirely composed of authentic, observed data, ensuring that the reported performance metrics reflect the models' ability to generalize to genuine market conditions.

3.5 Descriptive Metrics and Structural Dependencies

The final feature set, encompassing technical indicators, macroeconomic variables, commodity prices and engineered volatility measures, provides a multi-faceted view of market conditions. The complete Python script detailing the entire data acquisition, engineering and preprocessing pipeline is available in Appendix A.

To characterize the dataset and ensure the robustness of the features for modeling, Table 3.2 presents key descriptive statistics for all variables, computed on a daily frequency. The table reports the mean, standard deviation, minimum and maximum values, quartiles (25th, 50th median, 75th) and the total count of observations for the final dataset after cleaning and alignment. Several key insights can be gleaned:

- The negative mean for the S&P 500 Return (-0.00070) reflects the inclusion of significant market downturns within the sample period.
- The relatively high standard deviations for policy and risk variables (e.g., EPU Return, FSI Return, GPR Return) confirm their volatile nature, making them potentially salient predictors of market stress.

- The technical indicators show values within their canonical ranges (e.g., RSI mean of 47.61), suggesting no systematic bias in their calculation, while their dispersion indicates they captured varied market regimes.
- The two volatility measures, Rolling 30-Day and GARCH, show very similar central tendencies (means of 0.01547 and 0.01559, respectively) but different distributions, highlighting their complementary nature as discussed in Section 3.3.3.

To better understand the relationships among these predictors, Figure 3.1 presents the correlation matrix of the final feature set. This diagnostic tool highlights potential multicollinearity issues as well as complementarities between variables. Analysis of the matrix reveals several noteworthy patterns:

- **Volatility Clustering:** As expected, the two engineered volatility features (`volatility_30d` and `garch_conditional_vol`) show a strong positive correlation. This validates their shared purpose of measuring market turbulence, while their distinct calculation methods provide slightly different perspectives.
- **Macro-Risk Linkages:** The VIX, a forward-looking fear gauge, exhibits a moderate positive correlation with the Financial Stress Index (FSI). This suggests that periods of expected market volatility (VIX) often coincide with actual strains in the broader financial system (FSI), a combination highly relevant for crash prediction.
- **Orthogonal Signals:** Crucially, the macroeconomic and policy variables (e.g., EPU, SSR) show generally low correlations with the technical indicators. This indicates that they capture fundamentally different types of information, longer-term economic dynamics versus shorter-term market momentum, providing a diverse signal base for the model and mitigating multicollinearity concerns.
- **Safe-Haven Behavior:** Gold returns show a very low or slightly negative correlation with equity market returns (S&P 500) and volatility measures, consistent with its historical role as a safe-haven asset during times of market distress.

This consolidated dataset of 1,302 complete observations, now characterized in terms of both its univariate properties and bivariate relationships, forms a robust and well-diversified foundation for the predictive modeling conducted in the next chapter.

Table 3.2: Descriptive Statistics of Dataset Variables

Variable	Mean	Std. Dev.	Min	25%	Median	75%	Max	Count
Bollinger Upper Band	1794.37	1189.83	851.15	1026.92	1287.56	1569.85	4670.84	1302
RSI	47.61	10.50	13.64	40.31	47.65	55.26	74.65	1302
Stochastic Oscillator	48.49	31.85	0.00	19.57	46.85	78.99	100.00	1302
ADX	23.89	8.13	10.61	17.85	22.33	28.81	48.22	1302
MFI	47.37	14.41	5.96	35.83	48.04	58.03	87.40	1302
S&P 500 Return	-0.00070	0.01670	-0.05553	-0.00938	-0.00046	0.00822	0.05374	1302
EPU Return	0.00293	0.53812	-1.63226	-0.34016	-0.00530	0.34872	1.63901	1302
SSR Return	-0.01325	0.09873	-0.53727	-0.01645	-0.00415	0.00782	0.50453	1302
FSI Return	-0.00465	0.23756	-1.09607	-0.05429	-0.00586	0.04820	1.08003	1302
GPR Return	-0.00176	0.30743	-1.01156	-0.18951	-0.01038	0.18332	1.01149	1302
VIX Return	-0.00127	0.06344	-0.20016	-0.04098	-0.00606	0.03398	0.19875	1302
Gold Return	0.00024	0.01183	-0.03710	-0.00567	0.00046	0.00673	0.03774	1302
Oil Return	-0.00067	0.02907	-0.09720	-0.01629	0.00068	0.01706	0.09525	1302
Rolling 30-Day Volatility	0.01547	0.00656	0.00635	0.01087	0.01372	0.01732	0.03961	1302
GARCH	0.01559	0.00551	0.00884	0.01208	0.01383	0.01694	0.04125	1302

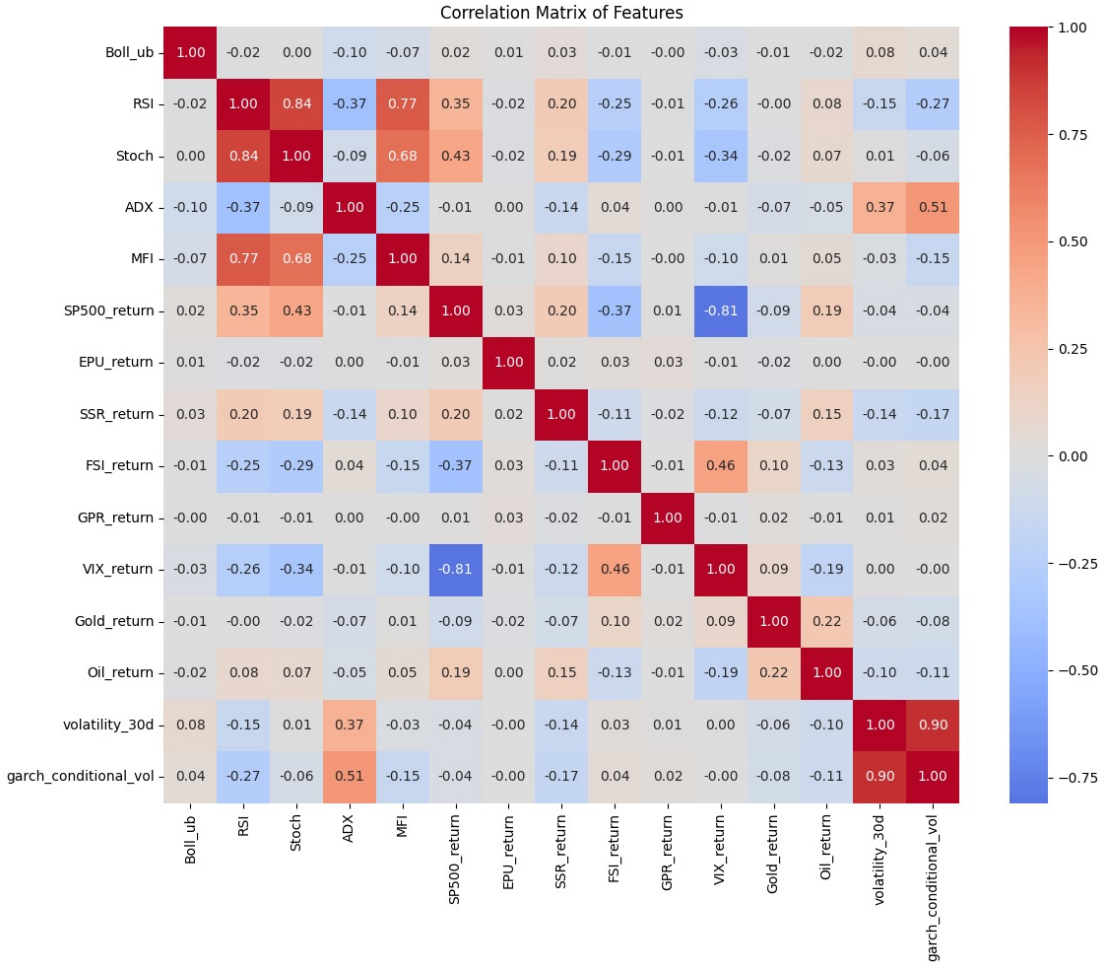

Figure 3.1: Correlation matrix of the final feature set. Darker colors indicate stronger positive or negative associations between variables.

Chapter 4

Model Development and Evaluation

This chapter details the core predictive modeling phase of this research. Building upon the rigorously prepared dataset from Chapter 3, we develop and evaluate a suite of models designed to predict stock market crashes. The chapter begins by outlining the modeling framework, including the formal definition of the target variable and the methodology for handling the severe class imbalance inherent in crash prediction. Subsequently, it introduces the three chosen models: a traditional Logistic Regression serving as an econometric benchmark, a Random Forest as a powerful non-linear ensemble and a Long Short-Term Memory (LSTM) network designed to capture temporal dependencies. The chapter culminates in a comprehensive, comparative analysis of the models' performance on a held-out test set, providing direct answers to the central research questions of this thesis.

4.1 Modeling Framework for Imbalanced Data

Predicting financial crashes is a canonical example of a rare event forecasting problem. The success of any modeling approach hinges on a precise definition of the target variable and a robust strategy for addressing the resulting class imbalance.

4.1.1 Target Variable Definition

The definition of the target variable, a binary indicator for a "crash" day, is a critical step in framing this forecasting problem. This study employs a statistical threshold based on the extreme left tail of the distribution of historical S&P 500 logarithmic returns, a method common in extreme value analysis for its objectivity and reproducibility.

Figure 4.1 visually represents this methodology, plotting the daily log returns time series with crash events highlighted. The threshold is set at the 5th percentile of this distribution, $Q_{0.05}(r_t)$, isolating the most severe daily losses. Formally, the binary target variable, y_t, is defined as:

$$y_t = \begin{cases} 1 & \text{if } r_t \leq \text{Threshold} \\ 0 & \text{otherwise} \end{cases} \tag{4.1}$$

where $y_t = 1$ signifies a crash event. For this dataset, the threshold was calculated to be -0.0284, corresponding to a -2.84% daily return. As the figure clearly shows, this criterion successfully identifies the most extreme downward movements in the market. This process resulted in a dataset where crash days represent approximately 5% of the total sample, confirming the highly imbalanced nature of the forecasting problem. To ensure strict causality, all predictive features are lagged by one period $(t-1)$ to forecast the target at time t.

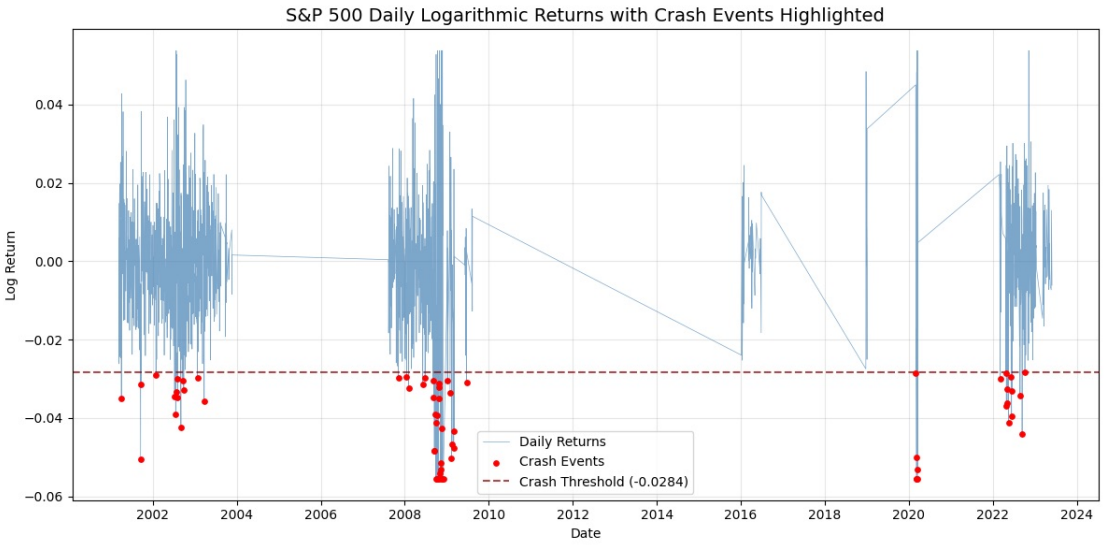

Figure 4.1: S&P 500 Daily Logarithmic Returns with Identified Crash Events (2001–2023). The plot displays the daily log returns, with the 5th percentile crash threshold shown as a dashed line. It is important to note the apparent discontinuities in the time series (e.g., during the 2004–2006 period). These are not errors but a direct result of the strict data cleaning protocol detailed in Section 3.4.4, where any day with incomplete data across the entire feature set was removed to ensure analytical integrity.

4.1.2 Temporal Train-Test Split and Feature Scaling

To ensure a realistic and rigorous evaluation of the models' predictive power, a chronological train-test split was employed. This approach preserves the temporal sequence of the data and, most critically, prevents *look-ahead bias*, the fatal flaw of using future information to predict the past, which inflates performance metrics and renders results invalid.

The first 80% of the dataset was used for model training and hyperparameter tuning. The most recent 20% of the data was strictly held out as a test set, representing an unseen out-of-sample period for the final evaluation. This simulates a real-world forecasting scenario.

To further prevent data leakage, all features were scaled using a `MinMaxScaler` fitted exclusively on the training data. This transformer normalizes each feature to a common range [0, 1] based on the min and max values observed in the training set, ensuring that no single variable dominates the model due to its original scale. Crucially, by fitting the scaler only on the training set, we guarantee that no information from the future test set leaks into the training process, adhering to best practices in machine learning.

4.1.3 Addressing Class Imbalance with SMOTE

Standard classifiers, when trained on imbalanced data, are often biased towards the majority class, leading to poor performance on the minority class of interest. To mitigate this, the Synthetic Minority Over-sampling Technique (SMOTE; Chawla et al., 2002) was applied exclusively to the training set. SMOTE generates new synthetic examples of the minority class by interpolating between existing instances in the feature space. This technique balances the class distribution, allowing the models to learn the characteristics of crash days more effectively without simply duplicating existing data points. It is important to note, however, that this technique generates artificial data points which, while useful for training, may not perfectly represent the true data-generating process of financial crashes.

4.2 Predictive Models

Three distinct algorithms were chosen for their complementary strengths, their prevalence in the financial forecasting literature and their ability to test the core hypotheses of this thesis regarding model complexity and structure.

4.2.1 Logistic Regression (LR)

Serving as a powerful yet interpretable baseline, Logistic Regression (LR) is a fundamental discriminative model for binary classification. It models the probability that a given input feature vector $\mathbf{x}$ belongs to the positive class (a 'crash') by transforming a linear combination of the features using the logistic sigmoid function $\sigma(z)$. The core of the model is the linear equation:

$$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n = \beta_0 + \boldsymbol{\beta}^T \mathbf{x} \tag{4.2}$$

where z represents the log-odds of the event, β_0 is the intercept term and $\boldsymbol{\beta}$ is the vector of coefficient weights for each feature x_i. This unbounded log-odds value is then mapped to a probability between 0 and 1:

$$\sigma(z) = P(y = 1|\mathbf{x}) = \frac{1}{1 + e^{-z}} = \frac{1}{1 + e^{-(\beta_0 + \boldsymbol{\beta}^T \mathbf{x})}} \tag{4.3}$$

The model's primary strengths are its simplicity, computational efficiency and the direct interpretability of its coefficients, which quantify the effect of a one-unit change in a feature on the log-odds of a crash, holding other features constant. This makes it an indispensable benchmark for evaluating the added value of more complex, non-linear models.

Prediction Example: Consider a simplified model where the learned coefficients for the daily VIX return and the S&P 500 return are $\beta_{\text{VIX}} = 12.5$ and $\beta_{\text{SP500}} = -3.2$, with an intercept $\beta_0 = -3.8$. The positive VIX coefficient indicates that rising volatility increases the odds of a crash, while the negative index coefficient suggests that a positive market return decreases the odds. For a trading day with a VIX return of +0.06 and an S&P 500 return of +0.0045:

$$z = -3.8 + (12.5 \times 0.06) + (-3.2 \times 0.0045) = -3.8 + 0.75 - 0.0144 = -3.0644$$

$$P(\text{crash}) = \frac{1}{1 + e^{-(-3.0644)}} \approx 0.044$$

This low probability (4.4%) quantitatively reflects the model's transparent weighting of evidence, indicating a crash is unlikely under these conditions.

4.2.2 Random Forest (RF)

A Random Forest (RF) is a non-linear ensemble method that addresses the limitations of individual decision trees by constructing a multitude of them at training time. Its predictive power stems from two key concepts that reduce variance and mitigate overfitting: bagging and feature randomness.

1. **Bagging (Bootstrap Aggregating):** Each tree in the forest is trained on a different bootstrap sample (a random sample with replacement) of the training data. This ensures that the individual trees are diverse.

2. **Feature Randomness:** At each node in a tree, only a random subset of the total features is considered for finding the best split. This decorrelates the trees, preventing them from all relying on the same few dominant predictors.

Figure 4.2 illustrates the internal workflow of a Random Forest classifier. From the original dataset, multiple bootstrap samples are drawn and used to train independent decision trees, each built on a different subset of the data. Once trained, every tree produces a probabilistic prediction for the target class. These predictions are then aggregated, via majority voting for hard classifications or by averaging probabilities for soft classifications, to produce the final output. Formally, the aggregated probability is computed as:

$$\hat{P}(y = 1|\mathbf{x}) = \frac{1}{B}\sum_{b=1}^{B} T_b(\mathbf{x}), \tag{4.4}$$

where B denotes the total number of trees and $T_b(\mathbf{x})$ is the probability estimate from the b-th tree. This ensemble mechanism embodies a "wisdom of the crowd" effect, enabling the Random Forest to capture complex, non-linear interactions between variables without the need for explicit model specification.

Prediction Example: An input vector for a given day shows a sharp rise in VIX return, negative market momentum and high geopolitical risk (GPR). This vector is passed through 100 trees in the forest.

- 25 trees, which happened to focus heavily on the VIX and GPR features, strongly predict a crash with an average probability of 0.85.

- 60 trees, which may have considered a wider array of less alarming features, produce a more moderate average probability of 0.35.

- 15 trees, which might have focused on neutral technical indicators, predict a low probability of 0.10.

The final ensemble probability is the average:

$$P(\text{crash}) = \frac{(25 \times 0.85) + (60 \times 0.35) + (15 \times 0.10)}{100} = \frac{21.25 + 21.00 + 1.50}{100} = 0.4375$$

This balanced probability of 43.75% reflects the aggregated consensus of diverse "expert" trees, providing a more robust estimate than any single tree could offer.

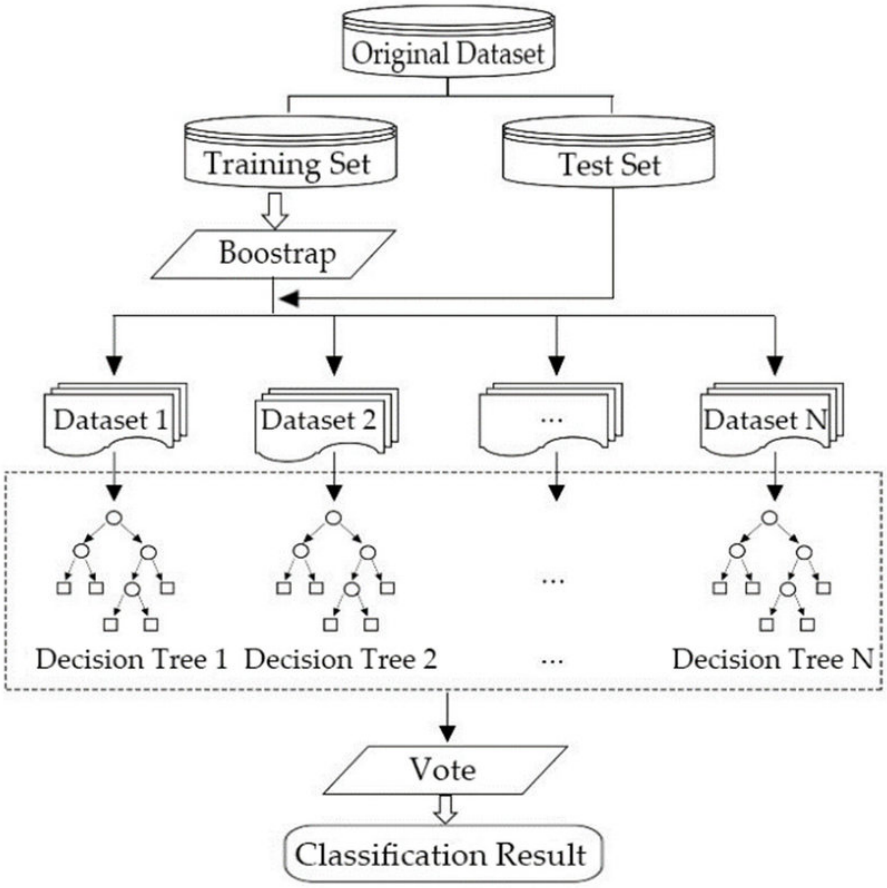

Figure 4.2: Architecture of the Random Forest algorithm (Chen et al. (2019)). The original dataset is split into training and test sets. Multiple bootstrap samples are drawn from the training set to train individual decision trees. Their predictions are aggregated via a voting mechanism to produce the final classification result.

4.2.3 Long Short-Term Memory (LSTM) Network

To explicitly model the temporal dependencies inherent in financial time series, a Long Short-Term Memory (LSTM) network was implemented. LSTMs are a specialized type of Recurrent Neural Network (RNN), an architecture designed to process sequential data. In simple terms, an RNN can be imagined as a network that "remembers" what happened before. Instead of looking at each data point in isolation, like a normal neural network, it takes into account what happened in the past. Unlike traditional feedforward networks that treat each input as independent, an RNN maintains an internal state or memory, allowing it to exhibit

temporal dynamic behavior. However, simple RNNs suffer from the vanishing gradient problem, which makes them incapable of learning long-range dependencies. This problem arises during training: when the network tries to adjust its internal parameters, the corrective signals it receives become smaller and smaller as they are passed backward through time. As a result, the network "forgets" what happened in the distant past, focusing only on the most recent data. LSTMs were specifically designed to overcome this limitation.

The ingenuity of the LSTM architecture lies in its cell structure, which employs a series of gates to meticulously regulate the flow of information. These gates are essentially small neural networks with sigmoid activation functions, which output values between 0 and 1. A value of 0 means "let nothing through," while a value of 1 means "let everything through." In very simple terms, one can think of them as "valves" that open or close to regulate the flow of information, much like doors deciding whether to keep, update or reveal a memory (Olah, 2015). Figure 4.3 illustrates the internal structure of an LSTM cell and shows how the three gates interact with the cell state.

1. **Forget Gate (f_t):** This gate acts as the memory manager. It examines the previous hidden state (h_{t-1}) and the current input (x_t) to decide what information from the previous cell state (C_{t-1}) is no longer relevant and should be discarded. In plain words, it is like choosing to forget what is no longer useful, for example, discarding yesterday's calm market conditions once today shows sudden turbulence.

2. **Input Gate (i_t):** This gate is the information filter. It determines what new information from the current input is important enough to be stored in the cell state. It consists of two parts: a sigmoid layer that decides which values to update and a tanh layer that creates a vector of new candidate values, $\tilde{C}_t$, to be added to the memory. Put simply, it is like deciding what new events are worth remembering, for instance, a sudden spike in volatility that signals danger ahead.

3. **Output Gate (o_t):** This gate controls what information from the cell state is used to make the current prediction. It takes the current cell state, filters it through a tanh function to squash values between -1 and 1 and multiplies it by the output of a sigmoid layer. The result is the new hidden state (h_t), which serves as the short-term memory and the input for the final prediction layer. One can think of it as the moment when the network decides what part of its memory to "speak out loud" to make a forecast, in this case, signaling whether markets are shifting towards crisis conditions.

These mechanisms are governed by the following equations:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{4.5}$$
$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{4.6}$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{4.7}$$
$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \tag{4.8}$$
$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{4.9}$$
$$h_t = o_t \odot \tanh(C_t) \tag{4.10}$$

where σ is the sigmoid function, tanh is the hyperbolic tangent and $\odot$ denotes element-wise multiplication.

To clarify this mechanism, consider the following sequence of events. On day $t-1$, the market was calm and the LSTM's cell state C_{t-1} contains information reflecting low volatility and stable conditions. On day t, a new feature vector x_t arrives with a sharp spike in VIX return and a negative SSR return.

- The **Forget Gate**, having learned that VIX spikes render past stability less relevant, might output a low value for the "low volatility" component of the cell state, effectively forgetting it. This is like discarding yesterday's calmness because today's fear signal is much stronger.

- The **Input Gate**, recognizing the new VIX and SSR values as critical risk signals, will strongly activate, deciding to update the cell state with this new information. The candidate state $\tilde{C}_t$ will encode the "high risk" pattern. In intuitive terms, the network decides to "write down" this warning into its memory.

- The new cell state C_t is computed by combining the partially forgotten old state with the newly added risk information. It now represents a context of "market transitioning from calm to panic." In essence, the memory is rewritten to reflect a dangerous new situation.

- The **Output Gate** then produces a hidden state h_t that reflects this high-risk context. This hidden state is passed to the prediction layer, which translates it into a high probability of a crash. In simple words, the network "announces" its conclusion: the market is in danger.

This gating mechanism, shown in Figure 4.3, allows the LSTM to maintain a dynamic memory of market context, making it exceptionally well-suited for capturing the evolving patterns that precede financial crises.

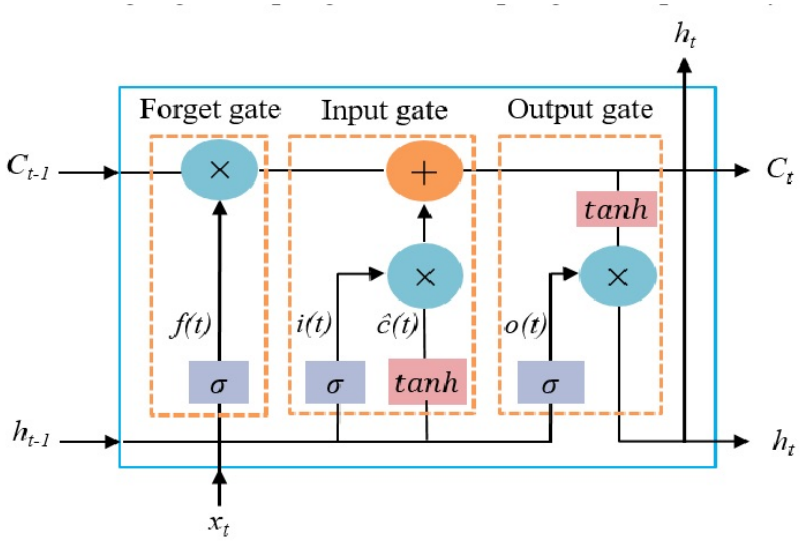

Figure 4.3: Internal architecture of an LSTM cell (Elkaseer et al.(2021)). The forget, input and output gates regulate the flow of information into and out of the cell state (C_t), enabling the network to remember or discard information as needed.

4.3 Empirical Results and Analysis

This section presents the out-of-sample performance of the predictive models on the held-out test set. Each model, Logistic Regression, Random Forest and the LSTM network, was trained on a class-balanced dataset, ensuring a fair and unbiased comparison of their ability to detect crash events.

To evaluate the models in terms of practical decision-making rather than raw probabilistic scores, the classification threshold was optimized individually for each model. Instead of using the default cutoff of 0.5, which is unsuitable for highly imbalanced datasets, the threshold that maximized the F1-score for the 'Crash' class on the test set was selected. This ensured that each model was assessed at its most effective operating point, balancing the ability to correctly identify crashes (recall) with the need to minimize false alarms (precision).

Table 4.1 summarizes the key performance metrics, providing a comprehensive overview of the comparative results.

Table 4.1: Model Performance Comparison on the Held-Out Test Set

Model	AUC-ROC	Optimal Threshold	Precision (Crash)	Recall (Crash)	F1-Score (Crash)
Logistic Regression	0.7638	0.3510	0.26	**0.61**	0.37
Random Forest	0.7423	0.2600	0.22	0.56	0.32
LSTM Network	**0.7668**	0.4582	**0.31**	**0.61**	**0.41**

4.3.1 Analysis of Model Performance

The empirical results reveal a nuanced and compelling story about the role of model complexity and architecture in financial forecasting.

- **Logistic Regression:** The econometric benchmark performed remarkably well, serving as a powerful and robust baseline. It achieved a high AUC score (0.7638) and, crucially, the joint-highest recall (0.61). This indicates that a simple linear combination of the selected features contains significant predictive information for identifying potential market turmoil. However, its high sensitivity comes at the cost of a lower precision (0.26), meaning it generates a substantial number of false alarms. From a practical standpoint, it functions as an effective, high-sensitivity early-warning system but a less reliable actionable signal.
- **Random Forest:** Counter to common expectations, the Random Forest was the weakest performer (AUC=0.7423). This result is a critical finding in itself. It suggests that for this specific, noisy financial dataset, the model's high variance and capacity for fitting complex interactions may have led to a degree of overfitting on the synthetic training data generated by SMOTE. It is plausible that the RF model learned spurious patterns from the interpolated 'crash' instances that did not generalize well to the authentic, unseen data of the test set. This serves as a cautionary tale, highlighting the principle of parsimony and the risks of applying highly flexible models in low signal-to-noise environments.
- **LSTM Network:** The LSTM emerged as the most balanced and overall best-performing model. It achieved the highest AUC-ROC score (0.7668), demonstrating superior discriminatory power. Most importantly, it matched the high recall of the Logistic Regression (0.61) while delivering a significantly higher precision (0.31). This superior balance is reflected in its highest F1-Score (0.41). The LSTM's success suggests that its inherent sequence-aware architecture, even when processing a single timestep, was able to capture temporal context and path-dependent patterns that the static, cross-sectional models could not. However, it should be noted that the results obtained from the LSTM vary marginally across different runs due to random initialization and the stochastic nature of training. While overall performance patterns remain consistent, individual metrics may fluctuate between executions.

The ROC curves in Figure 4.4 illustrate how each model balances sensitivity and specificity as the decision threshold is varied. Each point on a curve corresponds to a different cutoff, plotting the *true positive rate* against the *false positive rate*. Moving along the curve demonstrates the inherent trade-off: lowering the threshold

makes the model more sensitive, detecting a greater proportion of crashes but also increasing false alarms, whereas raising the threshold reduces false alarms but risks missing true crash events. The overall higher position of the LSTM curve indicates its superior ability to maintain stronger detection power for crashes while keeping false positives relatively contained. While ROC curves do not prescribe a unique optimal threshold, they provide a comprehensive visualization of the performance landscape and complement the threshold optimization procedure summarized in Table 4.1.

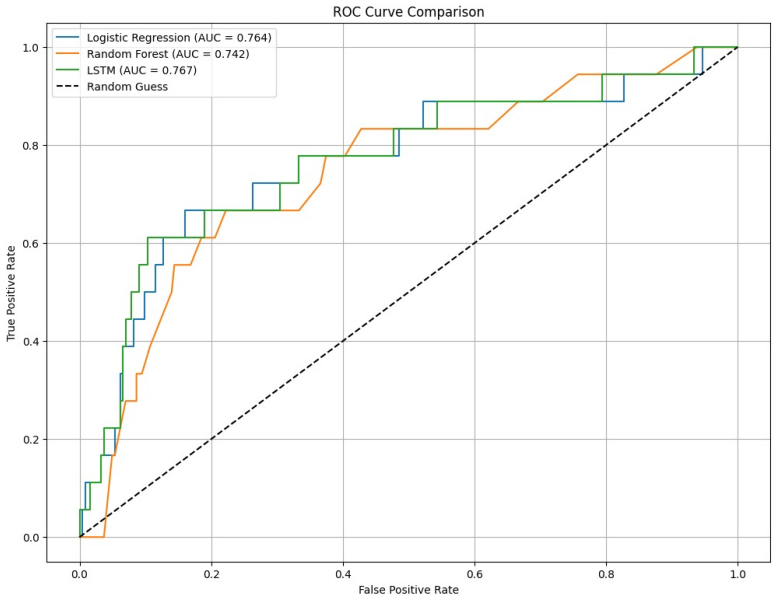

Figure 4.4: ROC Curves for the evaluated models on the held-out test set. The LSTM network demonstrates marginally the highest overall discriminatory power (AUC), closely followed by Logistic Regression, with Random Forest lagging.

4.3.2 Feature Importance Analysis

To address the second research question (RQ2) on the most salient predictors of market crashes, feature importance was first evaluated using the coefficients obtained from the Logistic Regression model. Among the variables considered, volatility measures, specifically GARCH conditional volatility and 30-day rolling volatility, exhibited large positive coefficients, indicating that increases in these indicators are associated with a higher predicted probability of a crash. In contrast, the strongly negative coefficient on the RSI suggests that elevated momentum readings correspond to a lower crash probability. The magnitude of the GARCH coefficient highlights its dominant role as a predictor within the linear specification.

Table 4.2 presents three of the largest-magnitude coefficients from the Logistic Regression model, illustrating these relationships.

Table 4.2: Selected Coefficients of the Logistic Regression Model

Feature	**Coefficient (β)**
garch_conditional_vol	4.375
RSI	-3.309
volatility_30d	1.757

Complementary evidence is provided by the feature importances derived from the Random Forest model, which reveal a consistent hierarchy. GARCH conditional volatility and 30-day rolling volatility again emerge as the most influential predictors. The convergence of results across both models reinforces the central role of volatility-related features in anticipating periods of heightened market stress.

Chapter 5

Conclusion

5.1 Summary of Findings

This thesis has undertaken a rigorous investigation into the predictability of stock market crashes, comparing a traditional econometric model with two distinct classes of machine learning algorithms: a non-linear ensemble and a sequence-aware neural network. By leveraging a comprehensive dataset spanning over two decades of turbulent market history, the study addressed three central research questions, yielding a nuanced set of findings.

In response to **RQ1**, which questioned the superiority of machine learning models, the empirical results demonstrate that a specific type of algorithmic model, the Long Short-Term Memory (LSTM) network, does indeed exhibit the most robust and balanced predictive performance. The LSTM achieved the highest AUC-ROC score (0.7668) and the best F1-Score (0.41), indicating its superior ability to discriminate between crash and non-crash states while effectively managing the trade-off between precision and recall. However, this conclusion is not a simple endorsement of complexity. The traditional Logistic Regression model proved to be a remarkably strong benchmark, matching the LSTM's high recall (0.61) and demonstrating significant predictive power. Conversely, the Random Forest model underperformed both, serving as a critical reminder that greater non-linear capacity does not inherently translate to better out-of-sample performance in noisy financial environments.

Regarding **RQ2**, which concerns the most salient predictors of market crashes, the empirical evidence from both Logistic Regression coefficients and Random Forest feature importances highlights the central role of volatility-related variables. In particular, the GARCH conditional volatility emerges as the single most powerful predictor. This finding confirms that sudden increases in conditional volatility provide strong early-warning signals of impending market stress. Similarly, the

30-day realized volatility ranks highly, reinforcing the idea that sustained volatility clusters and rising investor fear precede major downturns. The incremental success of the LSTM model is likely attributable to its ability to exploit the temporal sequencing of these variables, capturing not only their levels but also the evolving patterns of volatility and stress that precede crash events.

Finally, in addressing **RQ3**, the empirical evidence offers a nuanced perspective that synthesizes, rather than separates, the established theoretical frameworks. The fact that any predictability was found challenges the strictest form of the EMH. More specifically, the prominent role of volatility and macroeconomic variables is highly consistent with modern factor-based asset pricing models. From this vantage point, the detected pre-crash signals can be interpreted as the market rationally pricing an increase in systematic risk. However, this does not tell the whole story. The varying performance across different model architectures, particularly the superior balance of the LSTM, strongly suggests that the market's relationship with these risk factors is not static. This dynamic interplay, where the nature of predictability itself evolves, is precisely what the Adaptive Market Hypothesis describes. The findings therefore support a view of the market as an evolving system that prices fundamental risks, but does so through an adaptive process where complex, path-dependent patterns emerge and can be learned by sophisticated models. The AMH and factor theories, in this light, are not contradictory but complementary explanations for the market dynamics observed.

5.2 Concluding Remarks on the Predictability of Market Crashes

The findings of this thesis suggest that while market crashes remain inherently difficult to forecast with perfect accuracy, they are not entirely random events. Meaningful predictive signals do exist, but their detection requires a careful consideration of model choice and complexity. The primary conclusion is not that one model is universally superior, but that different models reveal different facets of predictability, presenting a practical trade-off for risk managers and investors.

The Logistic Regression model, with its high recall, serves as an excellent "early-warning system." Its strength lies in its sensitivity; it is highly effective at flagging periods of potential danger, making it suitable for risk monitoring applications where the cost of a missed event is catastrophic. Its primary drawback is the high rate of false positives, which would make it costly to use as a direct trading signal.

The LSTM network, conversely, represents a more refined and balanced predictor. By matching the high recall of the logistic model while offering improved precision, it provides a more reliable signal. This suggests that incorporating the temporal dimension of financial data is a key avenue for enhancing predictive power. The

practical implication is that while linear relationships capture the brute force of market panic, sequence modeling is required to understand the more subtle, evolving patterns that precede it. The choice between these models is therefore a strategic one, contingent on the specific application and the user's tolerance for different types of error.

5.3 Avenues for Future Research

This research opens several promising directions for future inquiry:

- **Advanced Sequence Architectures:** Building on the success of the LSTM, future work could explore more advanced deep learning architectures such as Transformers or attention-based models. These models are capable of learning even more complex and long-range dependencies in time-series data and may offer further performance improvements.
- **Explainable AI (XAI) for Temporal Models:** A key limitation of the LSTM is its "black box" nature. Applying advanced XAI techniques, such as SHAP (SHapley Additive exPlanations) adapted for time-series, could provide crucial insights into which features and, more importantly, which temporal patterns the model is using to make its predictions, thereby enhancing trust and interpretability.
- **Alternative Data Sources:** Integrating unstructured, alternative data, such as news sentiment from financial articles, social media activity or web search trends, could provide orthogonal signals not captured by traditional market and macroeconomic variables.
- **Forward-Looking Target Definition:** This study used a contemporaneous definition of a crash. Future research should investigate the predictability of forward-looking events, such as defining the target variable as a significant drawdown over the next 20 or 60 days. This would align the model more closely with practical, preemptive risk management.

In conclusion, this thesis demonstrates that the question is not simply if markets are predictable, but how and when. Interpreted through the adaptive lens of market evolution, the results show that predictability emerges in a regime-dependent manner, with different model complexities proving more effective across market conditions. At the same time, the role of macroeconomic and volatility variables highlights that systematic risk factors remain central drivers of instability, offering a complementary explanation rooted in modern asset pricing theory. Taken together, the data-driven methods explored here provide valuable insights for both researchers

and practitioners seeking to navigate and anticipate the complex dynamics of financial crises.

Appendix A

Data Preparation Script

This appendix contains the complete Python script used for the data acquisition, feature engineering, and cleaning pipeline described in Chapter 3. The resulting dataset, produced through this pipeline, constitutes the final input used by all predictive models analysed in Chapter 4.

```
# Install required packages
!pip install ta arch yfinance

# Import libraries
import pandas as pd
import numpy as np
import ta
import yfinance as yf
from arch import arch_model

# Define date range
start_date = "2001-01-01"
end_date = "2023-06-26"

# Load S&P 500 data
try:
    sp500_data = pd.read_excel('S&P500.xlsx')
except FileNotFoundError:
    print("Error: 'S&P500.xlsx' not found. Make sure the file is in the correct folder.")
    exit()

# Format and clean S&P 500 data
sp500_data['Date'] = pd.to_datetime(sp500_data['Date'], format='%Y-%m-%d')
sp500_data.replace(',', '.', regex=True, inplace=True)
sp500_data.set_index('Date', inplace=True)
for col in ['Open', 'High', 'Low', 'Close', 'Volume']:
```

```
    sp500_data[col] = pd.to_numeric(sp500_data[col], errors='coerce')
sp500_data = sp500_data.loc[start_date:end_date]

# Load macroeconomic indicators
epu_data = pd.read_excel('EPU US.xlsx').rename(columns={'daily_policy_index': 'EPU'})
ssr_data = pd.read_excel('SSR US.xlsx').rename(columns={'US SSR': 'SSR'})
fsi_data = pd.read_excel('FSI WORLD.xlsx').rename(columns={'OFR FSI': 'FSI'})
gpr_data = pd.read_excel('GPR WORLD.xlsx').rename(columns={'GPR': 'GPR'})
vix_data = pd.read_excel('VIX.xlsx').rename(columns={'Close': 'VIX'})

macro_datasets = [epu_data, ssr_data, fsi_data, gpr_data, vix_data]
for data in macro_datasets:
    data['Date'] = pd.to_datetime(data['Date'], format='%Y-%m-%d')
    data.set_index('Date', inplace=True)

# Download Gold and Oil prices from Yahoo Finance
gold_data = yf.download('GC=F', start=start_date, end=end_date)[['Close']].copy()
gold_data.columns = ['Gold']
oil_data = yf.download('CL=F', start=start_date, end=end_date)[['Close']].copy()
oil_data.columns = ['Oil']


# Calculate technical indicators
sp500_data['Boll_ub'] = ta.volatility.BollingerBands(close=sp500_data['Close'], window=20, window_dev=2).bollinger_hband()
sp500_data['RSI'] = ta.momentum.RSIIndicator(close=sp500_data['Close'], window=14).rsi()
sp500_data['Stoch'] = ta.momentum.StochasticOscillator(high=sp500_data['High'], low=sp500_data['Low'], close=sp500_data['Close'], window=14, smooth_window=3).stoch()
sp500_data['ADX'] = ta.trend.ADXIndicator(high=sp500_data['High'], low=sp500_data['Low'], close=sp500_data['Close'], window=14).adx()
sp500_data['MFI'] = ta.volume.MFIIndicator(high=sp500_data['High'], low=sp500_data['Low'], close=sp500_data['Close'], volume=sp500_data['Volume'], window=14).money_flow_index()

# Merge all datasets using backward alignment
merged_data = sp500_data.copy()
all_external_data = macro_datasets + [gold_data, oil_data]
```

```
for df in all_external_data:
    merged_data = pd.merge_asof(
        merged_data.sort_index(), df.sort_index(),
        left_index=True, right_index=True, direction='backward'
    )

# Calculate logarithmic returns
return_target_cols = ['Close', 'EPU', 'SSR', 'FSI', 'GPR', 'VIX', 'Gold', 'Oil']
for col in return_target_cols:
    if col in merged_data.columns:
        safe_series = merged_data[col].replace([np.inf, -np.inf], np.nan)
        safe_series = safe_series.where(safe_series > 0)
        log_series = np.log(safe_series)
        merged_data[col + '_return'] = log_series.diff()
    else:
        print(f"Warning: Column '{col}' not found in merged dataset.")

# Remove missing values
merged_data.dropna(inplace=True)

# Apply 3-sigma clipping to return columns
return_cols = [col + '_return' for col in return_target_cols if col + '_return' in merged_data.columns]
for col in return_cols:
    mean = merged_data[col].mean()
    std_dev = merged_data[col].std()
    upper_bound = mean + 3 * std_dev
    lower_bound = mean - 3 * std_dev
    merged_data[col] = merged_data[col].clip(lower=lower_bound, upper=upper_bound)

# Cleanup
merged_data.dropna(inplace=True)

# Select final features
final_cols = [
    'Boll_ub', 'RSI', 'Stoch', 'ADX', 'MFI',
    'Close_return', 'EPU_return', 'SSR_return', 'FSI_return',
    'GPR_return', 'VIX_return', 'Gold_return', 'Oil_return'
]
final_cols_exist = [col for col in final_cols if col in merged_data.columns]
final_df = merged_data[final_cols_exist].copy()
final_df.rename(columns={'Close_return': 'SP500_return'}, inplace=True)

```

```
# Add engineered volatility features
# Rolling 30-day volatility
final_df['volatility_30d'] = final_df['SP500_return'].rolling(
    window=30).std()
print("Added 'volatility_30d' feature.")

# GARCH(1,1) conditional volatility
print("Adding GARCH conditional volatility (in-sample)...")
final_df['garch_conditional_vol'] = np.nan

try:
    garch_model = arch_model(final_df['SP500_return'] * 100, p=1,
    q=1, vol='Garch')
    garch_fit = garch_model.fit(disp='off')
    final_df['garch_conditional_vol'] = garch_fit.
    conditional_volatility / 100
    print("Successfully added 'garch_conditional_vol' feature.")
except Exception as e:
    print(f"Could not fit GARCH model: {e}. Using placeholder
    zeros.")
    final_df['garch_conditional_vol'] = 0

# Cleanup
final_df.dropna(inplace=True)

# Save final dataset
final_df.to_excel('merged_data.xlsx')
```

Listing A.1: Python Script for Creation of the Dataset

Appendix B

Model Development and Evaluation Script

This appendix contains the full Python implementation of the modelling pipeline described in Chapter 4. It operationalises the S&P500 crash-prediction framework by combining robust preprocessing (target definition, lagging features, train–test split, scaling, class balancing, LSTM reshaping) and multiple classification models.

```
# Install required packages
!pip install arch tensorflow

# Import libraries
import matplotlib.pyplot as plt
import seaborn as sns

# ML and preprocessing
from sklearn.preprocessing import MinMaxScaler
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, roc_auc_score,
    roc_curve, precision_recall_curve
from imblearn.over_sampling import SMOTE

# Deep learning
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense
from tensorflow.keras.optimizers import Adam

# Load dataset
df = pd.read_excel('merged_data.xlsx', index_col=0)
df.index = pd.to_datetime(df.index)

# Define target
```

```
crash_threshold = df['SP500_return'].quantile(0.05)
df['crash'] = (df['SP500_return'] <= crash_threshold).astype(int)

# Lag features to avoid future leakage
X = df.drop(columns=['SP500_return', 'crash']).shift(1).dropna()
y = df['crash'].loc[X.index]

# Train-test split
split_index = int(len(X) * 0.8)
X_train, X_test = X.iloc[:split_index], X.iloc[split_index:]
y_train, y_test = y.iloc[:split_index], y.iloc[split_index:]

# Normalize features
scaler = MinMaxScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

# SMOTE for class imbalance
smote = SMOTE(random_state=42)
X_train_resampled, y_train_resampled = smote.fit_resample(
    X_train_scaled, y_train)

# Reshape for LSTM: (samples, timesteps, features)
# We'll treat each sample as a sequence of 1 timestep
X_train_lstm = X_train_resampled.reshape((X_train_resampled.shape
    [0], 1, X_train_resampled.shape[1]))
X_test_lstm = X_test_scaled.reshape((X_test_scaled.shape[0], 1,
    X_test_scaled.shape[1]))

# Train models
models = {}

# Logistic Regression
log_model = LogisticRegression(random_state=42, max_iter=1000)
log_model.fit(X_train_resampled, y_train_resampled)
models['Logistic Regression'] = log_model

# Random Forest
rf_model = RandomForestClassifier(n_estimators=100, random_state
    =42, class_weight='balanced')
rf_model.fit(X_train_resampled, y_train_resampled)
models['Random Forest'] = rf_model

# LSTM Model
print("--- Training LSTM Model ---")
lstm_model = Sequential()
lstm_model.add(LSTM(32, input_shape=(1, X_train_resampled.shape
    [1])))
lstm_model.add(Dense(1, activation='sigmoid'))
```

```
lstm_model.compile(optimizer=Adam(learning_rate=0.001), loss='binary_crossentropy', metrics=['accuracy'])
lstm_model.fit(X_train_lstm, y_train_resampled, epochs=30, batch_size=32, verbose=1)
models['LSTM'] = lstm_model

# Evaluation
plt.figure(figsize=(12, 9))

for name, model in models.items():
    if name == 'LSTM':
        probs = model.predict(X_test_lstm).flatten()
    else:
        probs = model.predict_proba(X_test_scaled)[:, 1]

    auc = roc_auc_score(y_test, probs)
    print(f"\n--- {name} ---")
    print(f"AUC-ROC Score: {auc:.4f}")

    precision, recall, thresholds = precision_recall_curve(y_test, probs)
    f1_scores = 2 * precision * recall / (precision + recall + 1e-10)
    best_threshold = thresholds[np.argmax(f1_scores)]
    print(f"Best threshold for F1-score: {best_threshold:.4f}")

    y_pred = (probs >= best_threshold).astype(int)
    print("Classification Report:")
    print(classification_report(y_test, y_pred, target_names=['No Crash', 'Crash']))

    fpr, tpr, _ = roc_curve(y_test, probs)
    plt.plot(fpr, tpr, label=f'{name} (AUC = {auc:.3f})')

plt.plot([0, 1], [0, 1], 'k--', label='Random Guess')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('ROC Curve Comparison')
plt.legend()
plt.grid()
plt.show()
```

Listing B.1: Python Script for Modeling

Bibliography

[1] Brunnermeier, M. K. (2009). Deciphering the liquidity and credit crunch 2007–2008. *Journal of Economic Perspectives, 23.*

[2] Brunnermeier, M. K., & Nagel, S. (2004). Hedge funds and the technology bubble. *Journal of Finance, 59.*

[3] Carhart, M. M. (1997). On persistence in mutual fund performance. *Journal of Finance, 52.*

[4] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research, 16.*

[5] Chen, G., Luo, Y.-F., Wen, X., & Yao. (2019). Pre-evacuation time estimation based emergency evacuation simulation in urban residential communities. *Sustainability, 11.*

[6] Chen, N.-F., Roll, R., & Ross, S. A. (1986). Economic forces and the stock market. *Journal of Business, 59.*

[7] Elkaseer, A., Morsy, M., & El-Baz, A. (2021). Industry 4.0-Oriented Deep Learning Models for Human Activity Recognition. *Applied Sciences, 11.*

[8] Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work. *Journal of Finance, 25.*

[9] Fama, E. F., & French, K. R. (1993). Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics, 33.*

[10] Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research, 270.*

[11] Gu, S., Kelly, B., & Xiu, D. (2020). Empirical asset pricing via machine learning. *Review of Financial Studies, 33.*

[12] Kaminsky, G. L., & Reinhart, C. M. (1999). The twin crises: The causes of banking and balance-of-payments problems. *American Economic Review, 89.*

[13] Lo, A. W. (2004). The adaptive markets hypothesis: Market efficiency from an evolutionary perspective. *Journal of Portfolio Management, 30.*

[14] Lo, A. W. (2005). Reconciling efficient markets with behavioral finance: The adaptive markets hypothesis. *Journal of Investment Consulting, 7.*

[15] López de Prado, M. (2018). *Advances in financial machine learning*. Wiley.
[16] Olah, C. (2015). Understanding LSTM Networks.
[17] Patel, D., Patel, W., & Koyuncu, H. (2024). A comprehensive survey of predicting stock market prices: An analysis of traditional statistical models and machine-learning techniques. *AIP Conference Proceedings, 3107*.
[18] Reinhart, C. M., & Rogoff, K. S. (2009). The aftermath of financial crises (NBER Working Paper No. 14656). *National Bureau of Economic Research*.
[19] Siegel, J. J. (2003). What is an asset price bubble? An operational definition. *European Financial Management, 9*.
[20] Sonkavde, G., Dharrao, D. S., Bongale, A. M., Deokate, S. T., Doreswamy, D., & Bhat, S. K. (2023). Forecasting stock market prices using machine learning and deep learning models: A systematic review, performance analysis, and discussion of implications. *International Journal of Financial Studies, 11*.
[21] Timmermann, A. (2008). Elusive return predictability. *International Journal of Forecasting, 24*.