HEC MONTREAL

Forecasting Stock Return Volatility Using Machine Learning Techniques and Macroeconomic Variables

By
Samaneh Maram

Supervisor
Prof. Tolga Cenesizoglu

Administrative Science (Finance)

A Thesis Submitted
in Partial Fulfillment of Requirements
for a Master of Science in Finance

August 2025

Résumé

En s'appuyant sur les conclusions de Paye (2012), cette thèse examine si l'intégration de variables macroéconomiques dans les modèles d'apprentissage automatique (ML) permet de produire des prévisions plus précises de la volatilité des rendements boursiers, comparativement à un modèle autorégressif (AR) de référence. Nous estimons la volatilité réalisée sur la période allant de 1927 à 2023, aux fréquences mensuelle et trimestrielle, à l'aide de la régression linéaire ordinaire (OLS), de la régression ridge, du lasso, de l'elastic net, de la forêt aléatoire, des arbres de régression à gradient boosting (GBRT) et d'un réseau de neurones à mémoire longue à court terme (LSTM). Les prévisions sont élaborées selon des fenêtres glissantes et récursives, couvrant huit périodes d'échantillonnage incluant des contextes de marché stables et de crise. Nous constatons des améliorations modestes de la précision prédictive hors échantillon pour les modèles linéaires (OLS et régressions régularisées) ainsi que pour le LSTM, lorsque les données sont mensuelles et le schéma de fenêtre récursive appliqué. Toutefois, ces gains ne sont pas statistiquement significatifs et, par conséquent, les modèles les plus performants tendent à égaler le modèle AR en termes de performance prédictive. En comparant l'ensemble des échantillons de prévision, nous observons que la qualité des prévisions s'améliore lorsque la fenêtre d'estimation s'élargit et lorsque l'on passe des données trimestrielles aux données mensuelles. Les tests par permutation réalisés sur les modèles LSTM et elastic net indiquent que la volatilité passée représente environ deux tiers du pouvoir prédictif, suivie par les variables d'écart de crédit, tandis que les mesures de l'activité économique réelle contribuent peu.

Mots clés: Prévision de la Volatilité, Ridge, Lasso, Elastic Net, Gradient Boosting, Mémoire Longue à Court Terme, Causalité de Granger, Modèle Autorégressif, Fenêtre Glissante, Fenêtre Récursive

Summary

Building on the findings of Paye (2012), this thesis investigates whether machine learning models augmented with macroeconomic variables yield more accurate forecasts of stock return volatility compared to the autoregressive (AR) benchmark model. We estimate ordinary least squares (OLS), ridge, lasso, elastic net, random forest, gradient boosted regression trees (GBRT), and a long short-term memory (LSTM) network on realized volatility spanning from 1927 to 2023 at both monthly and quarterly frequencies. Forecasts are generated under rolling and recursive windows across eight sample periods, which include tranquil and crisis market conditions. We find modest improvements in out-of-sample prediction accuracy in linear models (OLS and regularized regressions) and LSTM in monthly data and recursive window scheme. However, these gains are not statistically significant and therefore, the best performing models tend to match the AR benchmark in terms of predictive performance. Comparing all forecasting samples, we find that forecast quality improves when the estimation window expands and when data move from quarterly to monthly. Permutation tests on the LSTM and elastic net models show that past volatility accounts for roughly two-thirds of predictive power, followed by credit spread variables, while measures of real economic activity contribute little.

Keywords: Volatility Forecasting, Ridge, Lasso, Elastic Net, Gradient Boosting, Long Short-Term Memory, Granger Causality, Autoregressive Model, Rolling Window, Recursive Window

Contents

List of Tables

List of Figures

List of Abbreviations

Abbreviation	Full phrase
ALE	Accumulated Local Effects
ANNs	Artificial Neural Networks
AR	Autoregressive
ARCH	Autoregressive Conditional Heteroscedastic
ARMA	Autoregressive Moving Average
BAA	Moody's BAA-rated corporate bonds
CART	Classification and Regression Tree
CHF	Swiss Franc
CPI	Consumer Price Index
CRSP	Center for Research in Security Prices
COVID-19	Coronavirus Disease 2019
CW	Clark and West test
EGARCH	Exponential Generalized Autoregressive Conditional Heteroscedastic
EN	Elastic Net
FRED	Federal Reserve Economic Data
FX	Foreign Exchange
GARCH	Generalized Autoregressive Conditional Heteroscedastic
GARCH-NN	Generalized Autoregressive Conditional Heteroscedastic-Neural Network
GBRT	Gradient Boosted Regression Trees
GDP	Gross Domestic Product
GW	Giacomini-White Test
HAC	Heteroskedasticity and Autocorrelation Consistent
HAR	Heterogeneous Autoregressive
HARQ	Heterogeneous Autoregressive with Realized Quarticity
KNN	K-Nearest Neighbors
LASSO	Least Absolute Shrinkage & Selection Operator
LSTM	Long Short-Term Memory
ML	Machine Learning
NASDAQ	National Association of Securities Dealers Automated Quotations
NBER	National Bureau of Economic Research
NN	Neural Network
OLS	Ordinary Least Squares

Abbreviation	Full phrase
PPI	Producer Price Index
RF	Random Forest
RR	Ridge Regression
RV	Realized Volatility
SHAP	SHapley Additive exPlanations
SHAR	Scaled Heterogeneous Autoregressive
SVR	Support Vector Regression
UK	United Kingdom
US	United States
USD	United States Dollar
VIX	CBOE Volatility Index

Abbreviations of forecasting variables

Abbreviation	Full name
blev	Changes in bank leverage
cp	Commercial paper-to-Treasury spread
cay	Consumption-wealth ratio
gdp	Current GDP growth
dfr	Default return spread
dfy	Default spread
egdp	Expected GDP growth
exret	Expected return
ip	Growth in industrial production
ik	Investment-capital ratio
npy	Net payout yield
tms	Term spread
ipvol	Volatility of growth in industrial production
ppivol	Volatility of inflation growth

Acknowledgement

I would like to sincerely thank my research supervisor, Prof. Tolga Cenesizoglu for his guidance, patience, and generosity throughout this process. His thoughtful feedback and great perspective have been invaluable in helping me grow as a researcher and in bringing this thesis to completion. I am also grateful to my love, Reza, for his constant understanding and encouragement. His presence has given me the strength to keep going through the most challenging moments. This work is the result of not only my efforts but also the love, support, and kindness I have received from him.

Chapter 1. Introduction

Forecasting is crucial for making informed decisions in many different fields of study, including finance, economics, supply chain management, meteorology, and even public health and epidemiology. Accurate forecasts equip individuals and organizations to plan, mitigate risk, and allocate resources efficiently. Specifically, in the finance industry, the ability to forecast volatility in asset prices plays a significant role in risk management, portfolio allocation, pricing of derivatives, and broader policy making (Engle, 1982; Bollerslev, 1986; Ding et al., 1993; Christensen et al., 2023). As a result, volatility forecasting has attracted extensive attention from academic finance researchers and practitioners.

However, despite the ample attention volatility modeling has received, reliable volatility forecasting is challenging due to many factors, such as the complex behaviour of financial markets, incomplete and inconsistent understanding of volatility drivers, and frequent structural changes in market dynamics, especially during market downturns (Schwert, 1989; Glosten et al., 1993; Engle and Rangel, 2008).

Initial efforts at volatility modeling relied predominantly on linear econometric approaches, including the ARCH and GARCH models, proposed by Engle (1982) and Bollerslev (1986), respectively. Applied in different financial time series, these models successfully captured key features of volatility such as persistence and clustering behaviour, mean reversion, leverage effects, and heavy-tailed return distributions (Engle and Patton, 2001; Filipovic and Khalilzadeh, 2021).

Despite their effectiveness, ARCH and GARCH models struggle to incorporate a wide range of information, such as macroeconomic variables and firm- and market-specific features, limiting their predictive power.

Later studies proposed alternative volatility modeling techniques, such as AR and HAR models, that use realised volatility (RV) computed from high-frequency return data (Taylor, 1986; French et al., 1987; Schwert, 1989; Paye, 2012). Because RV is observable, it could be directly employed in linear forecasting models, producing more transparent predictions and avoiding the potential misspecification risks encountered with ARCH and GARCH models.

However, because these linear models can handle a large set of predictors, they are prone to biased and inconsistent parameter estimates, resulting in inferior out-of-sample forecasting performance (Paye, 2012; Filipovic and Khalilzadeh, 2021).

More recently, machine learning (ML) models have gained great attention in volatility forecasting due to their ability to handle complex and non-linear relationships and handle a broad set of predictive variables better than simpler models (Zhu et al., 2023). ML methods such as regularized regressions, tree-based (or ensemble) models, and neural networks, applied to volatility forecasting, have demonstrated superior out-of-sample predictive accuracy in most cases (Mittnik et al., 2015; Luong and Dokuchaev, 2018; Carr et al., 2019; Moon and Kim, 2019; Nybo, 2020; Filipovic and Khalilzadeh, 2021; Nõu et al., 2021; Petrozziello et al., 2022; Christensen et al., 2023; Zhu et al., 2023; Zhang et al., 2024; Niu et al., 2024; Rahimikia and Poon, 2024). Furthermore, several hybrid approaches, which combine linear econometric models, such as ARCH/GARCH, with ML algorithms, have also shown improved forecast accuracy (Donaldson and Kamstra, 1997; Nõu et al., 2021). A few studies, however, documented the underperformance of ML models compared to simpler benchmarks (Branco et al., 2022; Audrino and Chassot, 2022).

Despite broad evidence supporting the superior performance of ML models, there is considerable variation in findings due to differences in model specifications, data frequency, predictor sets, and estimation windows. For example, some studies have used highly granular market data such as bid-ask spreads and intraday volatility measures, which enhanced predictive accuracy through market microstructure information (Filipovic and Khalilzadeh, 2021; Zhang et al., 2024; Rahimikia and Poon, 2024). Some others have incorporated macroeconomic data alongside the market-based predictors and have reported that the predictive power of ML models relies predominantly on market-related variables (Filipovic and Khalilzadeh, 2021; Christensen et al., 2023), questioning the incremental value of macroeconomic variables in such forecasting frameworks.

These observations suggest that despite advancements in modeling techniques, financial market volatility forecasting remains challenging and inconclusive. Furthermore, although some studies have indicated the impact of economic events (notably market crashes) on volatility forecasting (Paye, 2012; Nybo, 2020; Rahimikia and Poon, 2024), to our knowledge,

there is limited research focusing solely on the predictive power of macroeconomic variables as a feature set in advanced machine learning techniques. Addressing these problems, this thesis investigates whether the application of machine learning techniques and the integration of exclusively financial and macroeconomic variables can significantly improve volatility forecasts, closing the existing gap in the literature. Specifically, our main research objectives are:

1. To test whether machine learning techniques, ranging from simpler regularized regression models to advanced ensemble and deep learning methods, significantly outperform the autoregressive benchmark in forecasting stock return volatility.
2. To assess the incremental predictive power of macroeconomic variables in forecasting volatility.

Our research has theoretical and practical pertinence. Theoretically, it contributes to the existing literature by assessing the predictive accuracy of advanced ML forecasting methods that employ macroeconomic predictors. Practically, our findings provide valuable insights for financial analysts, risk managers, and portfolio managers, helping them make optimal investment decisions. The originality of our work stems from its explicit focus on macroeconomic variables employed in machine learning models, which addresses a gap identified in existing volatility forecasting literature.

To conduct the analysis and answer the research questions, this thesis employs a diverse range of forecasting models, including regularized regressions (ridge, lasso, and elastic net), ensemble methods (random forest and gradient boosted regression trees (GBRT)), and a deep learning technique (long short-term memory (LSTM)). These models are tested using historical monthly and quarterly data on realized volatility and macroeconomic variables through rolling and recursive window approaches.

The set of macroeconomic variables used in our study is selected from Paye (2012). Paye (2012) investigated whether it was possible to enhance volatility forecasts on a monthly and quarterly basis by conditioning on macroeconomic variables. The author compared the forecasts generated from augmented ordinary least squares (OLS) with those generated from a simple autoregressive (AR) model. Similarly, the benchmark used in this thesis is the same

AR model from Paye (2012). Our study starts with replicating the findings of Paye (2012) and then extending his study by employing more advanced forecasting techniques beyond the traditional regression (OLS) model.

The remainder of this thesis is structured as follows: Chapter 2 reviews the relevant literature on volatility forecasting and outlines specific gaps motivating this research. Chapter 3 details the research design and methodology, including data collection and variable construction, forecasting models, evaluation methods, and specific details on forecast settings. Chapter 4 presents empirical results on the predictive accuracy of ML models against the AR benchmark. Moreover, this chapter discusses key findings in the context of existing literature, highlights practical implications, and suggests directions for future research. Finally, Chapter 5 presents the conclusion.

Chapter 2. Literature Review

2.1 Introduction

In finance, forecasting is fundamental for investors, portfolio managers, risk managers, policymakers, and other market participants in their decision-making process. Reliable and accurate forecasts can significantly enhance investment strategies and risk management practices, as well as support more informed fiscal and monetary policy decisions. Consequently, forecasting financial variables such as asset prices, returns, volatility, interest rates, exchange rates, and economic indicators has gained considerable interest from both academics and practitioners over recent decades.

Among financial variables, volatility is critical in risk management, portfolio allocation, derivatives pricing, and market regulation. However, volatility is challenging to predict for several reasons. First, there is an incomplete understanding of the true drivers of volatility, despite numerous studies attempting to identify these factors (Campbell, 1987; Schwert, 1989; Breen et al., 1989; Shanken, 1990; Glosten et al., 1993; Whitelaw, 1994; Graham and Harvey, 2001; Marquering and Verbeek, 2004; Ludvigson and Ng, 2007; Engle and Rangel, 2008; Engle et al., 2008; Campbell and Diebold, 2009; Lettau and Ludvigson, 2010; Paye, 2012).

Second, financial markets exhibit complex, nonlinear behavior and sudden structural shifts, making volatility hard to predict. Still, Engle (1982), Bollerslev (1986), Ding et al. (1993), and more recently Christensen et al. (2023) all reported that volatility is persistent. In other words, large fluctuations in prices tend to be followed by large fluctuations, a pattern known as volatility clustering. This slow decay in the autocorrelation of absolute returns suggests there is some predictability in market volatility.

Third, existing literature on volatility forecasting indicates inconsistent and various conclusions, due to differences in modeling techniques, volatility variable measurement, predictors, data frequencies, and forecasting horizons. This diversity makes it difficult to reach a consensus on whether volatility can be forecasted with high precision in financial markets.

In this chapter, we review prior studies, looking at their forecasting approaches, specific models used, data choices, and results. We then discuss how these findings align with our research objectives.

2.2 Evolution of volatility modeling

Methodologies in volatility modeling have evolved considerably over time. Initially, studies relied predominantly on linear econometric models, such as ARCH, GARCH, AR, and HAR models. However, more recently, an increasing number of studies employ nonlinear and advanced modeling techniques, including machine learning algorithms. In this section, we review the evolution of volatility forecasting models from traditional linear to advanced methods, discussing their effectiveness and limitations.

2.2.1 ARCH and GARCH frameworks

After decades of assuming constant variance for time series in econometric models, Engle (1982) introduced the autoregressive conditional heteroscedastic (ARCH) model, the first stochastic process to let volatility change over time. More specifically, Engle showed that while the unconditional variance (the long-run average volatility) remains fixed, the conditional variance evolves each period as a function of past squared forecast errors, so that volatility today reflects recent shocks (Engle, 1982).

To test his new model, Engle employed it to estimate the mean and variance in UK inflation, based on quarterly inflation data from 1958Q2 to 1977Q2. His analysis revealed that the ARCH effect played a significant role in modeling the volatility of UK inflation. In particular, there was a cluster of large variances in the mid-1970s, during which the UK inflation became hard to predict. However, the ARCH model captured this increased volatility more accurately than a homoscedastic model, which assumes that variance remains constant over time.

Overall, Engle (1982) demonstrated that variance is itself time-varying and should be explicitly modeled rather than treated as a constant noise level.

Later, Engle (1983) applied the ARCH technique to model the conditional variance of US inflation. His study provided support for the ARCH model by showing that volatility is time-varying rather than constant. His tests rejected homoskedasticity and demonstrated that modeling today's variance as a function of past squared errors allowed the ARCH model to track rises and falls in volatility. Building on this, Engle and Kraft (1983) extended the ARCH framework beyond one-period-ahead forecasts to multiperiod forecasting of US inflation

volatility, showing that iterating the model preserves volatility's persistence over multiple horizons.

Engle et al. (1987) thereafter extended the ARCH model to estimate time-varying risk premia in the term structure. To this end, the authors employed a new version of the ARCH, known as the ARCH-M model, which incorporated conditional variance directly into the mean equation. This model specification reflected the economic idea that risk-averse economic agents require higher returns for holding riskier assets. The authors applied the ARCH-M model to three fixed income products: 2-month Treasury bills, 6-month Treasury bills, and 20-year Aaa corporate bonds to determine whether time-varying risk premia exist for these assets. The empirical evidence demonstrated that risk premia are not constant over time; instead, they change with the degree of market uncertainty. In periods when investors perceived greater risk (higher conditional volatility), they demanded higher returns.

After several successful applications of the ARCH model, Bollerslev (1986) introduced the generalized ARCH (GARCH) model. This model included lagged values of conditional variance in addition to lagged squared errors from the original ARCH technique (Bollerslev, 1986). Bollerslev (1986) proved that incorporating past values of conditional variance into the traditional ARCH model accounts for the long memory of volatility and provides a more flexible lag structure. In the GARCH model, estimating more parameters can reduce the number of lags required for both squared errors and conditional variance, which leads to a more parsimonious model than the original ARCH model.

Baillie and Bollerslev (1989) put the GARCH model into practice by fitting it to daily exchange rate data and examining its pattern. Their study revealed that although daily exchange rate returns followed a random walk in the mean, their volatility was not constant over time. The authors implemented a GARCH(1,1) model, which uses one lag each of the conditional variance and past squared errors. The results indicated persistent, time-varying volatility and heavy-tailed behavior in the data. By successfully modeling the time-varying volatility of exchange rates using this framework, this study helped explain the risk dynamics of daily exchange rate movements.

Further studies introduced other extensions of ARCH and GARCH models, notably the exponential GARCH (EGARCH), which was developed by Nelson (1991). This new model used the log of the conditional variance as a function of past shocks. Nelson applied EGARCH to CRSP daily return series, and the results showed that the model not only captured volatility clustering but also exhibited the leverage effect, meaning that past negative shocks increased future volatility relative to positive shocks of the same magnitude (Nelson, 1991).

The ARCH and GARCH family models have become standard and useful tools for forecasting volatility, widely adopted by academics and risk managers. They also serve as the benchmark against which other modeling techniques are compared, in terms of prediction accuracy.

2.2.2 AR model

In the ARCH/GARCH models discussed above, volatility is inferred indirectly from return equations. In contrast, the autoregressive (AR) approach is based on observed volatility series directly. The mostly used variable in AR models in volatility forecasting studies, is realized volatility (RV), an ex-post measure of return variability over a fixed interval. It is constructed by summing squared high-frequency returns. For example, daily realized volatility is computed by summing the squared intraday returns for a given trading day (Andersen and Bollerslev, 2001; 2003). In this study, we adopted RV as the target variable at monthly and quarterly frequencies; accordingly, realized volatility was calculated as the sum of the squared daily returns within each month or quarter. Details of our volatility measurement are presented in the next chapter (methodology and research design). Some of the earliest studies employing realized volatility included Taylor (1986), French et al. (1987), Schwert (1989), and our reference study, Paye (2012).

The AR approach assumes that today's realized volatility is a linear function of its own past values (or lags). In one of the earliest applications of AR on realized volatility, Andersen and Bollerslev (1998) showed that an AR(1) model, using one lag of daily RV computed from five-minute returns, explained most of the predictable variation in the US equity market. Taylor and Xu (1997) similarly fitted an AR model to the realized volatility of UK equities, finding significant persistence in volatility up to ten trading days.

Subsequent studies have adopted AR models as benchmarks for more sophisticated methods. Andersen et al. (2001), using daily realized volatility constructed from high-frequency foreign exchange (FX) returns, found that a simple AR model performed competitively with more complex approaches in terms of forecasting accuracy. Ghysels et al. (2006) pushed this further by using an AR(5) on daily realized volatility of the S&P 500 index and showed that it often outperformed a GARCH(1,1) in one-day-ahead forecasts. Hansen and Lunde (2005) also demonstrated that straightforward AR fits on realized volatility were hard to beat in out-of-sample tests across a wide range of equity and FX series.

Overall, whereas ARCH/GARCH models predict today's volatility from yesterday's return shock, the AR approach forecasts today's volatility directly from past realized volatility. By using an observed volatility series, AR models generate more transparent forecasts, capture the strong persistence seen in the volatility of financial time series, and avoid potential misspecification in the return equation used in ARCH/GARCH. Since AR on realized volatility has proven to be a strong framework, and following our reference study (Paye, 2012), we also used AR(2) and AR(6) models in our study for quarterly and monthly data, respectively, as the benchmarks comparing their one-step-ahead forecasts to those generated from augmented and more advanced models.

2.2.3 HAR model

Building on the traditional autoregressive (AR) model for realized volatility, Corsi (2009) proposed the heterogeneous autoregressive HAR model. HAR employs multiple realized volatility components over different horizons as its inputs and accordingly, captures the long-term nature of market volatility in a parsimonious way, without using complex long-memory models. Specifically, the HAR model incorporates different realized volatility measures, including daily, weekly, and monthly, reflecting different time horizons of the market participants (Corsi, 2009).

Using the HAR model, Corsi (2009) generated out-of-sample forecasts for the realized volatility of three series: the USD/CHF exchange rate, the S&P 500 index, and US Treasury bonds, and showed that, at one-day, one-week, and two-week horizons, HAR reduced forecast errors significantly compared to short-memory benchmarks like simple AR or GARCH

models. Although it did not consistently beat more complex long-memory models, its performance was mostly similar to them, despite HAR's simpler structure.

Nevertheless, due to its simplicity and possible inefficiency at managing more complex data, later studies proposed extended versions of the standard HAR model. For example, Corsi and Reno (2012) added macroeconomic variables to account for the business cycle impacts on volatility, while Patton and Sheppard (2015) developed a jump-robust version of HAR to handle sudden price moves. Bollerslev et al. (2016) incorporated leverage effects and option implied volatility, Luong and Dokuchaev (2018) adjusted for high-frequency noise and microstructure bias, and Niu et al. (2024) combined HAR with machine learning techniques to learn nonlinear patterns flexibly.

Accordingly, the easy structure of HAR and its capability to add additional predictors (other than volatility lags), made it a popular forecasting tool or benchmark in the most recent studies. Notably, Christensen et al. (2023) enhanced HAR with macroeconomic and survey variables and reported gains in out-of-sample accuracy, Niu et al. (2023) mixed HAR inputs into a neural network and reported better short-term forecasts, and Rahimikia and Poon (2024) integrated jumps and liquidity measures into HAR to capture extreme moves and trading frictions. These studies will be discussed further in the following section.

In summary, the HAR model is a simple but powerful technique to forecast realized volatility by combining daily, weekly, and monthly measures. Its straightforward setup makes it easy to add new predictors, while still delivering strong accuracy.

2.2.4 Machine learning models

In addition to the time series frameworks such as ARCH/GARCH, AR, and HAR, a wide range of research has shifted toward implementing machine learning (ML) techniques for forecasting tasks. This shift is justified by the inherent complexity and nonlinearity of financial market data. While linear models capture persistent patterns using lagged volatility inputs, as discussed earlier, they might not be able to discover nonlinear dependencies in large and noisy datasets. Machine learning methods provide a data-driven approach that can find complex patterns without depending on any prior assumptions about the dataset being analyzed.

In simple words, machine learning techniques are a class of algorithms that can identify patterns in data and optimize prediction accuracy through experience. These algorithms minimize human intervention in detecting the patterns and relationships from the data. However, as noted by Filipovic and Khalilzadeh (2021), this minimal human intervention requires significant computational resources. Nevertheless, with the increasing use of high-frequency data, cloud computing, and efficient algorithms, ML approaches have become more feasible and practical for forecasting different variables in finance.

Specifically, prior studies have implemented ML models to predict volatility in various financial data including equity returns, comparing their forecasting performance to benchmark models such as ARCH/GARCH, AR, and HAR. However, the diversity of methodologies, predictors, volatility definitions (such as realized volatility, implied volatility, conditional variance), evaluation criteria, and time horizons has made it difficult to come up with consistent and unified conclusions across the literature. In this section, we review some of the most influential and interesting studies that have applied machine learning techniques, ranging from simplest ones like regularized linear models to neural networks for volatility forecasting.

2.2.4.1 Applications of multiple ML models

A vast majority of existing literature has applied multiple ML techniques (rather than a single model) to be able to compare their prediction performance. For instance, Christensen et al. (2023) applied a variety of ML models, including ridge regression, lasso, elastic net, random forest, gradient boosting, and neural networks, to forecast realized volatility for constituents of the Dow Jones Industrial Average. Not only did the authors employ several ML techniques, but their benchmark was also a set of extended versions of the HAR model, namely LevHAR, HAR-X, HARQ, SHAR, and LogHAR (Christensen et al., 2023).

The forecasting variables used in this study included lagged values of realized volatility in different time horizons, incorporated in benchmarks and ML models, and a range of market-related and a few macro variables, specifically used in ML models. The findings revealed that even with minimal hyperparameter tuning by the researchers, ML models outperformed benchmark HAR models in out-of-sample forecasting, especially at longer horizons. Notably, ML models incorporating only lagged realized volatility still outperformed HAR, suggesting

that ML's ability to capture long-term dynamics provides a practical advantage over the HAR-family models. Moreover, using the accumulated local effects (ALE) technique to assess variable importance, the authors found consistent, still model-specific rankings, indicating that different ML models extract distinct insights from the same dataset.

Similarly, Filipovic and Khalilzadeh (2021) evaluated a range of ML algorithms, including elastic net, gradient boosted regression trees (GBRT), feedforward neural networks, and long short-term memory (LSTM), for forecasting future stock volatility. They used 46 market- and firm-specific characteristics like accounting variables and past returns, and eight macroeconomic predictors, such as interest rates, GDP growth rate, etc. Interestingly, the LSTM outperformed other models, particularly in market conditions with high volatility. Furthermore, the LSTM model with only volatility and return as predictors up to one year into the past, performed as good as an LSTM model with the full set of predictors and the same number of lags. The authors found that a small set of predictors, including current realized volatility, idiosyncratic volatility, bid-ask spread, and return, accounted for most of the models' predictive power.

In general, both Christensen et al. (2023) and Filipovic and Khalilzadeh (2021) found that parsimonious ML models, mainly using past volatility and market-based lagged predictors, perform as well as, or even better than, more complex models that incorporate a broad range of macroeconomic variables.

Consistent with this finding, Nõu et al. (2021) directly compared ML models to the econometric linear models, using only lagged prices as input features, for forecasting both returns and volatility. The authors evaluated random forest, support vector regression (SVR), and k-nearest neighbors (KNN) against ARMA (an AR model that incorporates moving averages of past volatilities) and GARCH on the NASDAQ Baltic Index. They also tested a hybrid GARCH-neural network (GARCH-NN) model to see whether combining models improves prediction performance. For return forecasting, SVR consistently outperformed ARMA. Regarding volatility forecasting, although GARCH models performed well in many cases, the hybrid GARCH-NN was able to outperform them in some settings, showing that combining traditional models with machine learning can improve predictions.

Hybrid models were originally introduced by Donaldson and Kamstra (1997) for forecasting daily volatility of the Canadian equity market. They proposed a GARCH-NN hybrid model where they first estimated volatility using a GARCH(1,1) model, and then applied a neural network to the residuals. The goal was to capture nonlinear dynamics that GARCH models could miss. Their study found that the hybrid model performed superior in terms of out-of-sample forecasting accuracy than a standard GARCH model.

Some studies employing multiple forecasting approaches have provided insights beyond the performance comparison of ML models and benchmarks. Specifically, Zhang et al. (2024) suggested that forecast horizon and data granularity significantly influence model performance in volatility forecasting. The authors studied the forecasting of intraday and daily volatility for the top 100 most liquid S&P 500 stocks. They tested various models, including ordinary least squares (OLS) regression, lasso, HAR, gradient boosting, multilayer perceptrons (MLP), and LSTM. Other than the lagged realized volatility measures, the predictors included market-wide volatility measures and microstructure features such as quote imbalances and stock-specific data. The results indicated that neural networks performed best for intraday volatility, especially when pooling data across stocks and incorporating market-wide volatility. However, for daily-basis data, simpler models such as OLS performed better.

Further research explored various aspects and analytical approaches in addition to testing several forecasting models. For example, Carr et al. (2019) attempted to forecast realized volatility using derivatives market data. Zhu et al. (2023) developed a panel data framework rather than relying solely on a time-series structure for their analysis. Finally, Niu et al. (2024) examined whether industry-specific realized volatility can predict aggregate future market volatility.

To be precise, Carr et al. (2019) extended the volatility forecasting literature by applying ML models to predict realized variance using option price data, specifically from out-of-the-money S&P 500 calls and puts. Instead of using past returns or volatility, their method looked at patterns in current option prices to understand what the market expects about future volatility. This reflects a forward-looking aspect that is not typically seen in econometric models. The study tested various models and found that ridge regression and shallow neural networks consistently outperformed both the VIX benchmark and simple linear models. This

suggests that ML techniques can effectively capture pricing signals embedded in option markets, under certain conditions.

Considering the role of cross-sectional information, Zhu et al. (2023) introduced a panel data-based machine learning framework for forecasting daily volatility across S&P 500 stocks. Instead of fitting separate models for each asset, they stacked features such as realized volatility, semi-variance, and jump components into a panel structure. ML models, including lasso, elastic net, random forest, and gradient boosting, outperformed traditional time series models like HAR, delivering better forecast stability and adaptability to different market conditions.

Niu et al. (2024) extended the gradual information diffusion hypothesis to volatility by assessing whether industry-specific realized volatility can predict aggregate market volatility. The idea behind this hypothesis, which was originally introduced by Hong et al. (1999), is that the information doesn't affect all parts of the market at once; it shows up in certain sectors first and then spreads more broadly. To test this in the context of volatility, Niu and colleagues used eight machine learning models, including support vector regression (SVR), neural networks, LightGBM, and AdaBoost, and found that LightGBM consistently had the best performance. Using SHAP, which is a variable importance analysis, the study identified health care, technology, and consumer services as early indicators at different forecast horizons. The results also revealed that adding sector-level information not only led to more accurate forecasts but also improved portfolio performance metrics such as Sharpe ratios and certainty equivalent returns.

While many of the studies discussed above have shown that ML models at least matched or achieved higher prediction accuracy than linear benchmark models, some literature indicated that various benchmarks consistently (or most of the time) delivered better forecasts. For instance, Branco et al. (2022) investigated whether ML models can outperform traditional linear models, specifically the HAR model, in forecasting one-day-ahead realized volatility of 10 major global stock indices. Using data from 2000 to 2021, the authors incorporated not only past RV values but also a broad set of predictors, including lagged returns and macroeconomic indicators. They evaluate both linear (OLS, lasso) and nonlinear (random forests, neural networks) models. Interestingly, their findings revealed that nonlinear ML

models did not statistically outperform the HAR or the linear ML models when enhanced with the same predictors. This suggests that linear models could be robust and competitive with complex ML models in realized volatility forecasting.

More interestingly, Audrino and Chassot (2022) evaluated the performance of the HAR model relative to ML techniques in forecasting realized volatility across a massive panel of 1,445 individual stocks. The ML models included lasso, random forest, gradient boosted tree, and feedforward neural networks. Even with extensive hyperparameter optimization, ML models consistently underperformed the HAR model with a rolling window estimation approach. Results showed that HAR, despite its simplicity and low computational expense, outperforms advanced ML models when both models use only realized volatility and VIX as predictors. The study underscores the importance of rolling window size and re-estimation frequency on model performance.

2.2.4.2 Applications of a single ML model

Among existing literature in volatility forecasting, numerous studies have evaluated the prediction accuracy of only one ML technique, such as a tree-based model or one of the neural networks. In an interesting setting, Luong and Dokuchaev (2018) proposed a two-stage ML approach using random forest. In the first stage, they predicted the direction of volatility using technical indicators and a purified implied volatility series. In the second, a random forest regression combined these predictions with traditional HAR inputs to forecast the magnitude of volatility. The results showed that their approach successfully predicted both the direction and magnitude of volatility, outperforming HAR models in predictive accuracy using high-frequency data.

Given the effectiveness of tree-based methods, Mittnik et al. (2015) demonstrated the impact of the boosting technique in stock market volatility forecasting. They developed a flexible approach by combining traditional ARCH models with the learning power of component-wise gradient boosting. Instead of estimating the model with fixed assumptions like in standard GARCH or exponential GARCH (EGARCH) models, they let the boosting algorithm learn how and to what extent different variables influence volatility, capturing non-linear relationships and important thresholds. Their model used a broad set of predictors, including

lagged returns, macroeconomic indicators (like interest rates), external variables such as oil prices, volatility indices, and exchange rates, as well as month/year effects and past volatility estimates. Compared to benchmark models like GARCH(1,1) and EGARCH, their approach produced more accurate out-of-sample forecasts and indicated how different economic and financial variables impact market volatility.

Further studies evaluated the performance of deep learning models (neural networks) in forecasting volatility. Among them, Nybo (2021) compared the effectiveness of GARCH models and artificial neural networks (ANNs) in forecasting stock market volatility across different sector categories with low, medium, and high volatility profiles. The results indicated that the volatility profile impacted the performance of forecasting models. In specific, ANNs outperformed GARCH models when applied to low-volatility sectors with smaller fluctuations, while GARCH models delivered stronger forecasts in medium and high-volatility sectors, probably due to their ability to capture persistent volatility clusters.

In another study, Petrozziello et al. (2022) used LSTM to forecast one-day-ahead realized volatility for US equities using only past returns and volatility. Compared with benchmarks such as GARCH(1,1), the LSTM model demonstrated superior accuracy, especially during the 2007-2008 financial crisis. However, the linear benchmarks performed comparably during calm market periods. This finding suggests that neural networks, in particular LSTM, can dynamically adjust to changing volatility dynamics.

More recently, Rahimikia and Poon (2024) advanced the research on neural networks by incorporating HAR variables, limit order book data, and news sentiment into LSTM models to 23 Nasdaq stocks over 15 years. With over 3.6 million model variations, the authors reported that LSTM outperformed standard HAR in most of out-of-sample forecasts, except during periods of extreme market stress, where HAR remained competitive. Moreover, SHAP analysis showed that mid-price, bid/ask levels, and pre-2018 sentiment indicators were the key drivers of forecast accuracy.

Findings of Nybo (2021), Petrozziello et al. (2022), and Rahimikia and Poon (2024) indicated that neural networks responded differently to various levels of market volatility. Petrozziello et al. (2022) showed strong LSTM performance during the 2007-2008 crisis, while Nybo

(2021) found that ANNs underperformed in high-volatility sectors. Similarly, Rahimikia and Poon (2024) reported that LSTM models were less effective than benchmarks during periods of extreme market stress.

Finally, Moon and Kim (2019) introduced "hybrid momentum" as a type of target variable for forecasting, using LSTM models. This hybrid measure combines both price momentum and volatility momentum, designed to reflect market behavior better. The authors tested various input feature sets, including moving averages, technical indicators, and market signals, to see how different data combinations affect model performance. Their results showed that using hybrid momentum as a forecasting target improved predictions for both index levels and market volatility. Moreover, adding more features helped improve price forecasts but had little impact on volatility predictions, suggesting that volatility is mostly driven by its recent patterns rather than additional variables. This highlights that although greater feature sets help in return forecasting, volatility may need a modeling approach with its own related and past values.

The next section reviews how macroeconomic variables have been integrated into the volatility forecasting literature. To conclude, we summarize the important empirical evidence discussed in this chapter to motivate our research objectives.

2.3 Macroeconomic predictors in volatility forecasting

Macroeconomic variables have been broadly used in return modeling literature, but their integration into the volatility forecasting has only gained interest more recently. As we discussed in the previous sections, traditionally, volatility models, namely ARCH/GARCH, AR, and HAR, have mainly relied on the past values of return or volatility. However, macroeconomic indicators provide forward-looking information and signals about the broader economy, possibly improving forecast accuracy.

One of the earliest and most impactful studies employing macroeconomic variables in volatility forecasting is by Paye (2012), which serves as the basis for our study. The author assessed whether aggregate US stock market volatility can be forecasted more accurately by using macroeconomic and financial variables, rather than relying solely on an autoregressive (AR) benchmark. Using monthly and quarterly realized volatility of the S&P 500, Paye first

documented the countercyclical and highly persistent behavior of volatility. He then assembled a wide set of macroeconomic predictors, which are all listed and discussed in detail in the next chapter of this thesis.

In-sample analysis indicated that several variables, notably the commercial paper-Treasury spread, default return spreads, and an investment-to-capital ratio, Granger-caused volatility. The author, then, tested out-of-sample forecasts using ordinary least squares (OLS) regression for multiple time horizons in three different settings: incorporating each macroeconomic variable separately, kitchen sink (which included all variables in OLS), and combining all forecasts from individual variables with mean, median, trimmed mean, or MSPE weighted schemes. He then compared all forecasts to those generated from AR benchmark models (Paye, 2012). The results of out-of-sample tests demonstrated that OLS models rarely beat parsimonious AR models. However, some small yet statistically significant gains emerged from combined forecasts, especially around the onset of NBER recessions. Paye concluded that macroeconomic variables indeed contain incremental information about future market risk (according to the Granger causality tests), however, their economic contribution to forecasting accuracy is limited.

Several studies, all discussed in more detail in the previous sections, have also incorporated macro variables as input features of their forecasting models. Corsi and Reno (2012) used macroeconomic indicators in an extended HAR framework to capture business cycle impacts on volatility. Filipovic and Khalilzadeh (2021) included macro variables among their 54 predictors when evaluating LSTM. Mittnik et al. (2015) incorporated multiple macro variables into their ensemble model. Similarly, Branco et al. (2022) added macro predictors to both linear and nonlinear models. And more recently, Christensen et al. (2023) included several macroeconomic variables as part of their extended feature set.

Although these studies have not consistently and explicitly indicated a clear advantage for macroeconomic variables over market-related or lagged variables in volatility forecasting, some others (Paye, 2012; Nybo, 2021; Petrozziello et al., 2022; and Rahimikia and Poon, 2024) have documented that macroeconomic events such as recessions and financial crises significantly impacted volatility forecast results. This suggests that macroeconomic variables may hold undiscovered predictive power.

2.4 Identified gaps and research motivation

Despite extensive research in financial volatility forecasting, discussed mainly in previous sections, there are significant gaps regarding application of machine learning techniques, particularly in the context of macroeconomic and financial predictors. In earlier studies, volatility modeling was predominantly based on linear econometric frameworks, including ARCH, GARCH, AR, and HAR models, which successfully captured the persistence and clustering nature of volatility (Engle, 1982; Bollerslev, 1986; Andersen and Bollerslev, 1998; Corsi, 2009). However, these models faced limitations in reflecting complex and nonlinear patterns frequently observed in financial markets (Mittnik et al., 2015; Luong and Dokuchaev, 2018).

Recent literature has increasingly demonstrated the potential of ML models to overcome these shortcomings. Studies employing multiple ML techniques, ranging from penalized regressions to ensemble and deep learning models, have found encouraging yet inconclusive results regarding their superior performance over linear traditional benchmarks (Filipovic and Khalilzadeh, 2021; Christensen et al., 2023; Zhang et al., 2024). Notably, some findings suggest that ML models consistently outperformed classical linear models, especially in high volatility market conditions (Filipovic and Khalilzadeh, 2021; Petrozziello et al., 2022), while others report comparable or even inferior performance relative to simpler benchmarks like the HAR model (Branco et al., 2022; Audrino and Chassot, 2022).

Moreover, although some previous studies showed that macroeconomic variables provided predictive value especially during economically turbulent periods (Paye, 2012; Nybo, 2021; Rahimikia and Poon, 2024), there is no consensus in the literature about their advantage relative to lagged return and volatility and market-based variables. Nevertheless, the finding that macroeconomic variables hold predictive power during recessions and financial crises underscores an unexplored potential of these variables in volatility forecasting.

Consequently, two clear gaps emerge from this synthesis: first, due to inconsistent evidence, there is still area for research in volatility forecasting and a need for a thorough comparison between ML models (ranging from simpler penalized regressions to advanced tree-based and deep learning methods) and strong benchmarks like AR model, second, the literature lacks an

explicit and detailed exploration of how macroeconomic variables exclusively, perform in volatility forecasting models (previous studies have used macro variables alongside many other variables such as lagged returns and market-based data e.g. bid-ask spread). Our research specifically addresses these two gaps by evaluating whether different ML techniques, which incorporate a set of macroeconomic variables, can significantly improve predictive accuracy compared to the standard AR model, extending the core findings of the seminal study of Paye (2012).

In the next chapter, we describe our research design and methodology thoroughly. We walk through how the data is prepared, describe each forecasting model, and explain the rolling and recursive steps and all relevant forecast settings in detail.

Chapter 3. Methodology and Research Design

In this chapter, we describe the methodology adopted to address our research question. In particular, this thesis investigates whether machine learning methods, ranging from simpler to more advanced models, using macroeconomic and financial predictors, can outperform the benchmark autoregressive (AR) model in forecasting US stock return volatility.

A variety of methods, widely applied in previous forecasting studies, are employed in our study to assess volatility prediction performance. Simple linear regression is included due to its simplicity and widespread use in previous studies, particularly in our reference study (Paye, 2012). Regularized regression models such as ridge, lasso, and elastic net were chosen because of their effectiveness in selecting relevant predictors through the introduction of penalties to regression coefficients. Ensemble methods, including random forest (RF) and gradient boosted regression trees (GBRT), were selected due to their capability to capture complex, nonlinear relationships among predictors and the target variable. Lastly, the long short-term memory (LSTM) model was adopted to explicitly account for long-term dependencies and volatility persistence inherent in financial time series data.

The chapter begins with a brief discussion of the replication process of the reference study (Paye, 2012). Then, a detailed description of all variables used in the analysis is provided. This is followed by a clear outline of the forecasting models and the evaluation criteria adopted to assess their predictive accuracy. The chapter concludes by explaining the forecasting setup, hyperparameter choices, and the variable importance analysis approach employed in this study.

3.1 Replication of the reference study

The initial step in our research was to replicate, as closely as possible, the findings of Paye (2012), which provided the foundation for our study. In his study, Paye examines whether the inclusion of macroeconomic and financial variables can enhance the forecasting of stock return volatility. Both in-sample and out-of-sample analyses were conducted using monthly and quarterly data. The main methodology relied on simple linear regression, applying lagged predictors within both rolling and recursive estimation windows.

For the replication process, we attempted to use the same datasets or, when unavailable, to reconstruct variables according to the definitions and procedures described by the author. The replicated tables and figures from Paye (2012) are provided in Appendix A for the purpose of consistency and comparison.

In the sections that follow, the full research design of this thesis is described in detail. Specifically, the data description, the OLS regression model, the evaluation metrics and statistical tests applied, main forecasting sample periods, and the estimation window procedures (rolling and recursive) are all closely aligned with Paye (2012). In contrast, the use of additional forecasting models, their corresponding hyperparameter tuning processes, the variable importance analysis, and the incorporation of extended sample periods represent extensions beyond the original study of Paye (2012).

3.2 Data description

This section describes the construction of the dependent variable; stock return volatility, and the macroeconomic and financial predictors employed in our forecasting models. Detailed information on data sources and the time horizon for which each variable is available is provided in Appendix B.

3.2.1 Target variable: stock return volatility

Construction of stock return volatility follows the same process as the reference study (Paye, 2012) and is explained step by step below. Specifically, this variable is measured as the natural logarithm of annualized realized volatility. First, realized volatility is calculated as the sum of the daily squared excess returns of the S&P 500 over the risk-free rate, at both monthly and quarterly frequencies. The formula used to compute realized volatility, adapted from Paye (2012), is:

$$RV(t) = \sum_{i=1}^{N_t} R_{i,t}^2, \qquad [1]$$

where $RV(t)$ denotes the realized volatility in month or quarter t, N_t indicates the number of trading days in in the corresponding period, $R_{i,t}$ denotes the excess return of S&P 500 over risk-free rate on trading day i of month or quarter t.

Paye (2012) notes that, following prior evidence from Andersen et al. (2001), taking the natural logarithm of realized volatility results in a distribution closer to Gaussian (Paye, 2012; Anderson et al., 2001), which improves estimation robustness. Consequently, the same transformation is applied as:

$$LVOL(t) \equiv Ln(\sqrt{mRV(t)}), \quad [2]$$

where $LVOL(t)$ denotes the log volatility in month or quarter t, and m is the annualization factor, set to four for quarterly data and twelve for monthly data.

The volatility variable is constructed from January 1927 to December 2023 at the monthly frequency (denoted as MLVOL), and from 1927Q1 to 2023Q4 at the quarterly frequency (denoted as QLVOL). Our study covers multiple forecasting periods, which are defined in the subsequent sections of this chapter. The quarterly stock return volatility series is presented in Figure 3.1, which is similar to Figure 1, Panel A of Paye (2012), but spans an extended period.

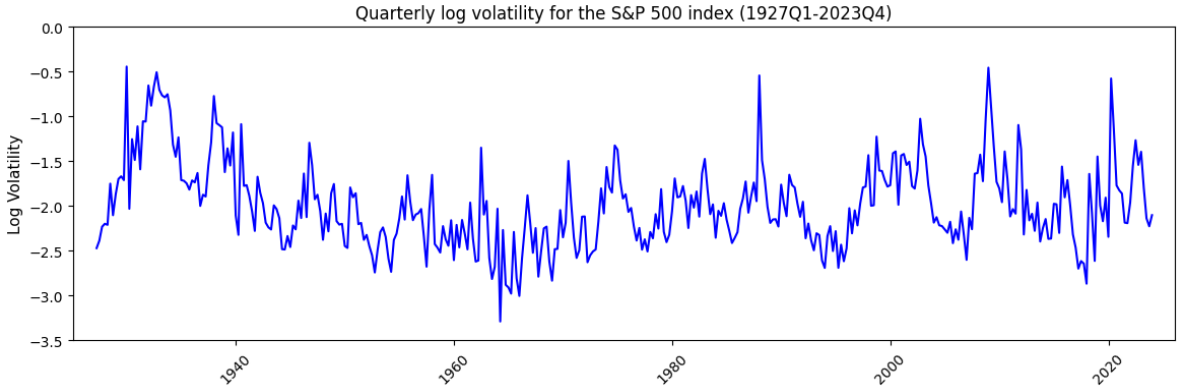

Figure 3.1 Quarterly volatility of the S&P 500 index, 1927-2023. This figure illustrates the time series plot of quarterly log realized volatility for the S&P 500 index from the first quarter of 1927 to the last quarter of 2023. The figure is inspired by and is an extended version of Paye (2012), Figure 1, Panel A.

As shown in Figure 3.1, sharp spikes in market volatility generally occur during periods of market stress and economic downturn, including the years 1929-1933, 1987, 2000-2002,

2008, and 2020.[1] This pattern suggests a close relationship between market volatility and the business cycle. To provide visual evidence of this, Figure 3.2, adapted from Paye (2012), illustrates the covariance between market volatility and the business cycle from 1947 to 2023.[2]

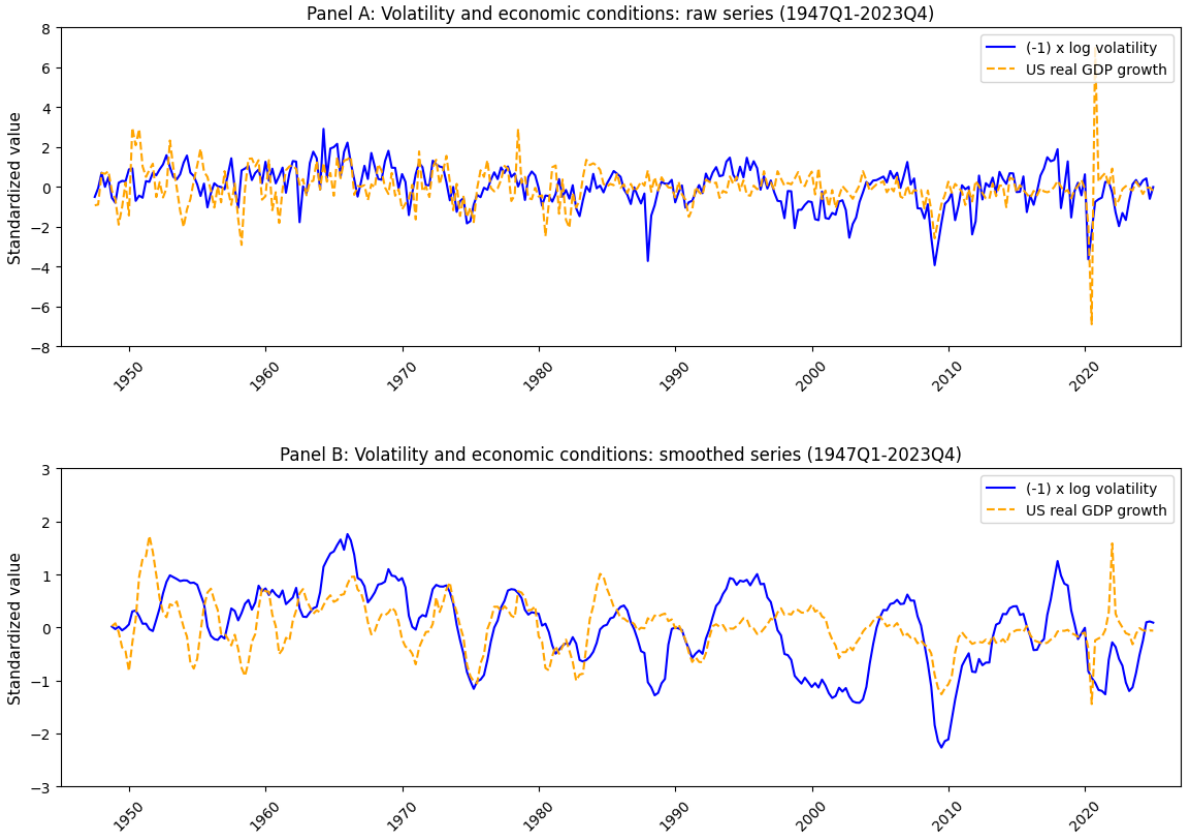

Figure 3.2 Relationship between market volatility and the business cycle, 1947-2023. This figureshows the covariation between quarterly volatility of the S&P 500 index and the US business cycle. The blue lines in both panels represent the opposite of the standardized values of quarterly log volatility, and the orange dashed lines represent the standardized values of US real gross domestic product (rGDP) from the first quarter of 1947 to the fourth quarter of 2023. Specifically, Panel A presents the raw data, and Panel B presents smoothed data computed as six-quarter moving averages.

[1] These critical economic events are discussed further in the following sections.

[2] The figure extends Figure 1: Panel B and Panel C of Paye (2012), covering a longer historical period.

The business cycle in Figure 3.2 is measured as the standardized growth rate of US real gross domestic product (GDP). In the figure, the time series data for log realized volatility are inverted by multiplying the standardized series by -1. Panel A displays the raw series and Panel B shows the smoothed series, computed using a six-quarter moving average.

As more evident in Panel B, the covariation is strong from the mid 1960s through the early 1980s; however, it weakens in certain periods, such as after 1987, throughout the 1990s, and during the mid to late 2010s. This implies a time varying relationship between volatility and the business cycle. These observations support the inclusion of macroeconomic and financial variables in our study, as they may capture additional information not already captured in past volatility. In the next section, we describe all financial and macroeconomic variables used as predictors in our models. As our study extends the work of Paye (2012), we employ the same set of forecasting variables used in his analysis.

3.2.2 Macroeconomic and financial variables

This section outlines all forecasting variables and their definitions. The frequency of each variable (monthly or quarterly) is consistent with Paye (2012).

Changes in bank leverage (*blev*)

This variable is computed as the percentage change in the leverage ratio; defined as total assets divided by total equity for security brokers and dealers, following the approach of Adrian and Shin (2010). *blev* is constructed on a quarterly basis.

Commercial paper-to-Treasury spread (*cp*)

This variable is calculated as the difference between the three-month commercial paper rate and the three-month Treasury bill rate. *cp* is constructed at both monthly and quarterly frequencies.

Consumption-wealth ratio (*cay*)

cay, originally introduced by Lettau and Ludvigson (2001), is defined as the residual from a cointegrating relationship between aggregate consumption, wealth, and labor income (Lettau and Ludvigson, 2001; Paye, 2012). This variable is constructed on a quarterly basis.

Current GDP growth (*gdp)*

gdp is the annualized growth rate of real, seasonally adjusted US gross domestic product (GDP). This variable is on a quarterly basis.

Default return spread (*dfr*)

This variable is calculated as the return difference between the long-term corporate bond and the long-term government bond. *dfr* is constructed at both monthly and quarterly frequencies.

Default spread (*dfy*)

Default spread is calculated as the difference between the yields on BAA-rated corporate bonds and long-term US government bonds. *dfy* is constructed at both monthly and quarterly frequencies.

Expected GDP growth (*egdp*)

Construction of *egdp* follows Campbell and Diebold (2009) and is based on the Livingston survey from the Federal Reserve Bank of Philadelphia. The survey collects macroeconomic forecasts from economists on a biannual basis (June and December). According to Paye (2012), *egdp* is computed as the log difference between the median 12-month and 6-month nominal GDP forecasts, then adjusted by subtracting the corresponding log-differenced consumer price index (CPI) forecast to obtain real GDP growth. Since the Livingston Survey is only available in June and December each year, the *egdp* series remain constant in the first and third quarters (Paye, 2012). This variable is constructed at quarterly frequency.

Expected return (*exret*)

This variable is an in-sample estimate of expected excess returns on the S&P 500 index over the risk-free rate, using a simple regression model. These fitted values are not meant for actual forecasting but are used as an ex-post proxy for the unobserved, time-varying expected stock returns (Paye, 2012). Because of this, overfitting is less of a concern compared to out-of-sample forecasting.

The independent variables are selected from the same macroeconomic and financial predictors used in our main forecasting models, with their first lag employed as regressors. However,

some variables are not available over the entire sample period, so the set of predictors varies across sub-periods.

To avoid information leakage across time, the regressions are estimated separately for each of the eight sub-samples for both monthly and quarterly data. Following Paye (2012) and Campbell and Thompson (2008), any negative fitted values are replaced with zero (Paye, 2012; Campbell and Thompson, 2008). *exret* is estimated on both monthly and quarterly basis.

Growth in industrial production (*ip*)

ip represents the percentage change (growth rate) in industrial production. This variable is primarily constructed to derive the variable *ipvol*, however, it is also used on its own as a predictor in our monthly samples.

Investment-capital ratio (*ik*)

Originally proposed by Cochrane (1991), this variable is calculated as the ratio of aggregate investment to aggregate capital in the US economy. *ik* is available at a quarterly frequency.

Net payout (*npy*)

The variable *npy* is constructed following the approach of Paye (2012), using data on aggregate market capitalization, dividends, and net equity issuance from Boudoukh et al. (2007). The net payout for month t (npy_t), adapted from Paye (2012), is calculated as:

$$npy_t = \ln(0.1 + dy_t - ney_t), \quad [3]$$

where dy_t is the dividend yield at month t and ney_t is the net equity issuance yield at month t. The components are computed as follows:

$$dy_t = \frac{\text{aggregate dividends over months } t \text{ through month } t-11}{\text{market capitalization at month } t}, \quad [4]$$

$$ney_t = \frac{\text{aggregate net equity issues over months } t \text{ through month } t-11}{\text{market capitalization at month } t}, \quad [5]$$

net equity issuance at month t is calculated as:

$$\text{net equity issue} = (\Delta\text{shares outstanding}) \times \left(\frac{price_{start} + price_{end}}{2}\right), \quad [6]$$

npy is constructed at both monthly and quarterly frequencies. Quarterly values are obtained by taking the arithmetic average of the three corresponding monthly observations.

Term spread (*tms*)

This variable is calculated as the difference between the yield on long-term government bonds and the short-term Treasury bill rate. It is constructed at both monthly and quarterly frequencies.

Volatility of growth in industrial production (*ipvol*)

Following Paye (2012), the variable *ipvol* is used as a proxy for the conditional volatility of US industrial production growth. It is constructed based on the method of Engle et al. (2008), originally adapted from Schwert (1989).

To build this variable, we first calculate the percentage growth in industrial production (*ip*), which is explained above, and then, estimate the following autoregressive (AR) model (Schwert, 1989; Engle et al., 2008):

$$X_t = \sum_{j=1}^{k} \alpha_j D_{jt} + \sum_{i=1}^{k} \beta_i X_{t-i} + \varepsilon_t. \qquad [7]$$

In this model, X_t represents the growth rate in industrial production (the *ip* variable), D_{jt} denotes dummy variable for seasonality, and X_{t-i} refers to the lagged values of *ip*. The parameter k depends on the data frequency: $k = 4$ for quarterly data, and $k = 12$ for monthly data. The squared residuals ($\hat{\varepsilon}_t^2$) from this model are used as the volatility measure (*ipvol*). This variable is constructed on both monthly and quarterly basis.

Volatility of inflation growth (*ppivol*)

This variable is used as a proxy for conditional volatility of inflation growth (Paye, 2012). It is constructed using the percentage changes in the Producer Price Index (PPI) data, applying the same autoregressive method used for constructing *ipvol*. In this context, X_t in Equation [7] represents the growth rate in the PPI series. *ppivol* is built at monthly and quarterly frequencies.

The following table summarizes all the forecasting variables, and their abbreviations used in this study.

Abbreviation	Description
blev	Percentage changes in the bank leverage ratio
cp	3-month commercial paper rate minus 3-month T-bill rate
cay	Cointegration residual from aggregate consumption, wealth, and labor income
gdp	Growth rate of real US GDP
dfr	Return difference between long-term corporate bonds and government bonds
dfy	Difference between BAA-rated corporate and long-term US government bond yields
egdp	Forecasted GDP growth from the Livingston survey
exret	Expected excess returns on the S&P 500 index
ip	Percentage changes in industrial production
ik	Ratio of aggregate investment to aggregate capital in the US economy
npy	Net payout
tms	Long-term government bond yield minus short-term T-bill rate
ipvol	Volatility of US industrial production growth
ppivol	Volatility of inflation growth

Table 3.1 List of financial and macroeconomic variables of our study. These variables are used as predictors (or input features) in our forecasting models. All variables and their construction methods are directly adapted from Paye (2012).

3.2.3 Summary statistics and correlation analysis

Table 3.2 presents the descriptive statistics for the forecasting variables (predictors), using the same metrics as in Paye (2012). Panel A reports statistics for quarterly data from 1952Q2 to 2019Q4, and Panel B shows statistics for monthly data from February 1952 to December 2019. The sample period (1952-2019) is selected to ensure full data coverage for all variables.

For each variable, the table reports the mean, standard deviation, skewness, kurtosis, first and second order autocorrelations (ρ_1 and ρ_2) and results from the Phillips Perron unit root test (Phillips and Perron, 1998). Specifically, the final two columns display the test statistic (Z_t) and the corresponding MacKinnon p-value (MacKinnon, 1994).

The statistics show that the variables differ in their average values and variability. To improve comparability and reduce the effect of outliers or extreme values, variables are standardized in some forecasting models of our study when appropriate.

Symbol	Name	Mean	Satndard Deviation	Skewness	Kurtosis	ρ_1	ρ_2	Philips-Perron test Z_t	Philips-Perron test p-value
Panel A: Quarterly sampling frequency									
blev	Changes in bank leverage	-0.0025	0.0887	-0.23	1.37	-0.28	0.21	-4.44	0.00
cp	CP-to-Treasury spread	0.5476	0.4428	2.11	8.27	0.77	0.59	-4.31	0.00
cay	Consumption–wealth ratio	0.0044	0.0258	-0.52	-0.84	0.97	0.95	-1.67	0.45
gdp	GDP growth	3.1203	3.6103	-0.14	1.55	0.34	0.20	-8.51	0.00
dfr	Default return	0.0003	0.0078	0.30	11.37	-0.10	0.05	-8.57	0.00
dfy	Default yield	0.0097	0.0043	1.78	4.45	0.90	0.77	-4.65	0.00
egdp	Expected GDP growth	2.5164	1.3333	-0.65	3.23	0.86	0.72	-3.99	0.00
exret	Expected return	0.0190	0.0144	0.65	1.14	0.80	0.73	-3.94	0.00
ik	Investment-capital ratio	0.0361	0.0031	0.36	-0.23	0.97	0.90	-4.17	0.00
npy	Net payout yield	-2.1903	0.1940	-1.80	4.91	0.96	0.90	-2.00	0.29
tms	Term spread	0.0169	0.0137	-0.01	-0.58	0.91	0.78	-4.15	0.00
ipvol	Industrial production volatilit	0.0002	0.0005	4.13	19.82	0.15	0.24	-3.80	0.00
ppivol	Inflation volatility	0.0002	0.0010	13.12	193.87	0.05	0.14	-9.75	0.00
Panel B: Monthly sampling frequency									
cp	CP-to-Treasury spread	0.5498	0.4659	2.35	10.21	0.88	0.77	-4.16	0.00
dfr	Default return	0.0003	0.0141	-0.37	6.64	-0.08	-0.09	-8.25	0.00
dfy	Default yield	0.0097	0.0043	1.84	4.89	0.97	0.92	-3.57	0.01
exret	Expected return	0.0041	0.0032	0.56	-0.13	0.81	0.77	-3.63	0.01
ip	Growth in industrial producti	0.0022	0.0091	0.27	7.31	0.38	0.23	-8.81	0.00
npy	Net payout yield	-2.1898	0.1957	-1.75	4.88	0.98	0.97	-2.32	0.16
tms	Term spread	0.0169	0.0140	-0.12	-0.14	0.96	0.91	-4.45	0.00
ipvol	Industrial production volatilit	0.0001	0.0002	12.19	192.66	0.13	0.05	-14.46	0.00
ppivol	Inflation volatility	0.0001	0.0002	10.34	131.63	0.34	0.24	-5.72	0.00

Table 3.2 Descriptive statistics of macroeconomic variables, 1952-2019. The table, which is an extended version of Table 1 in Paye (2012), summarizes descriptive statistics for the forecasting variables analyzed in this study. We report the mean, standard deviation, skewness, and kurtosis, along with the first and second order autocorrelation coefficients (ρ_1 and ρ_2). The table also displays the Z_t test statistic from the Phillips Perron unit root test and the corresponding MacKinnon p-value (Phillips and Perron, 1998; MacKinnon, 1994). Panel A is for quarterly data from 1952Q2 to 2019Q4, and Panel B is for monthly data from February 1952 to December 2019.

Looking at the ρ_1 and ρ_2 statistics, most of the forecasting variables show high first and second order autocorrelations, often exceeding 0.80. This indicates that the variables are highly persistent over time, a common feature of non-stationary time series data. As discussed in Stambaugh (1999) and Paye (2012), such persistence can lead to biased coefficient estimates in forecasting models, especially when predictor variables are also correlated with model residuals. To verify whether this persistence reflects non-stationarity, we follow Paye (2012) and apply the Phillips Perron (PP) test, which tests for the presence of a unit root. A rejection of the null hypothesis indicates that the variable is stationary and suitable for regression analysis. As shown in the final column of Table 3.2, the PP test rejects the unit root hypothesis for most variables based on low p-values (except for *npy* and *cay*, which show weak evidence

of stationarity). Overall, this suggests that our forecasting variables are appropriate for forecasting models, with no strong need to use special methods for highly persistent and non-stationary data. Figure 3.3 presents the correlation heatmap for all variables.

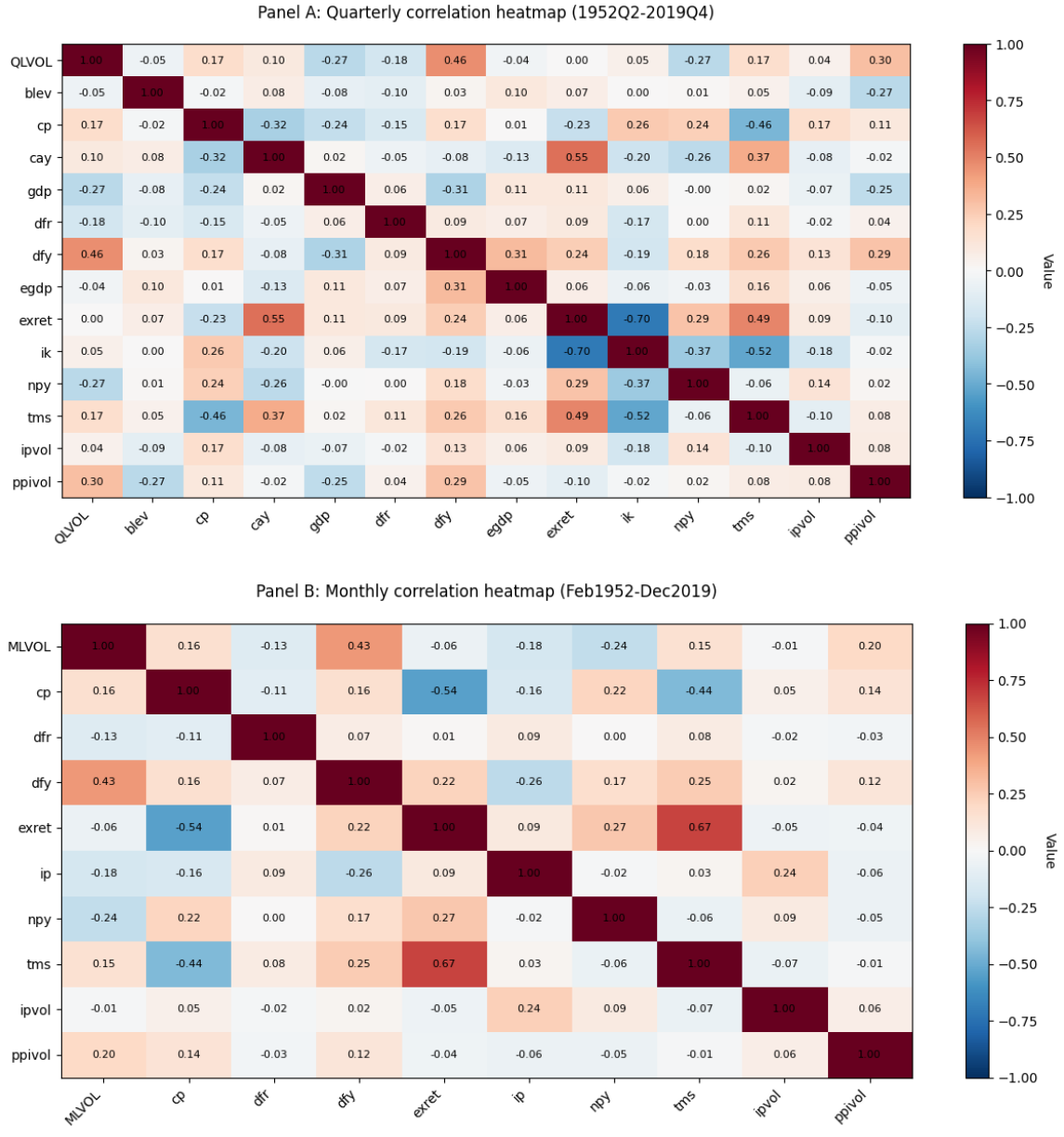

Figure 3.3 Correlation heatmap of all variables (log realized volatility and forecasting variables), 1952-2019. Panel A presents the correlation measures for quarterly data from 1952Q2 to 2019Q4, and Panel B shows the correlation values for monthly data from February 1952 to December 2019.

As seen in both heatmaps, most variables have low to moderate correlations with each other, suggesting that multicollinearity is not a major issue in our forecasting models. In particular, the quarterly heatmap in Panel A shows that most correlation values are below 0.60, which implies that the predictors contain a good variety of information without too much overlap. This supports their use in models that include multiple variables, as each one seems to reflect a different part of the economic and financial environment.

Similarly, the monthly heatmap in Panel B shows a similar pattern. Most variables are only moderately correlated, and while a few correlations are a bit stronger or weaker than those in the quarterly case, most stay under 0.60 as well. Taken together, the results from both panels suggest that the predictors are well-balanced and suitable for use in regression analysis.

3.3 Forecasting models

3.3.1 Autoregressive benchmark model

Following Paye (2012), we adapt a univariate autoregressive (AR) model as a benchmark to evaluate the out-of-sample predictive performance of our models. In particular, we compare the forecasts generated by forecasting models to those produced by this benchmark. The AR(P) model used in our study is defined as follows:

$$LVOL_t = \alpha + \sum_{i=1}^{P} \rho_i LVOL_{t-i} + \epsilon_t, \quad [8]$$

where $LVOL_t$ indicates the log realized volatility (our target variable) for quarter or month t, and P represents the number of lags used in the model, set to two for quarterly data and six for monthly data.

3.3.2 Ordinary least squares (OLS) regression

Traditional ordinary least squares (OLS) regression was applied for our forecasting purpose, in line with Paye (2012). The purpose of our study is to see how more advanced forecasting models perform compared to the benchmark and to the traditional multiple regression model in forecasting the log realized volatility of stock returns. This model, in part, was for our replication purpose for the common time periods with the reference paper. The OLS regression model used in the quarterly sampling, adapted from Paye (2012), is as follows:

$$LVOL_t = \alpha + \rho_1 LVOL_{t-1} + \rho_2 LVOL_{t-2} + \acute{\beta} X_{t-1} + \epsilon_t, \quad [9]$$

where $LVOL_{t-1}$ and $LVOL_{t-2}$ represent the first and second lags of the log volatility, respectively, and X_{t-1} denotes a vector containing the first lag of the variables used in our forecasting models. The OLS regression model for our monthly sampling, adapted from Paye (2012), is the same as the model used for the quarterly sampling, except that it includes six lags of log realized volatility:

$$LVOL_t = \alpha + \sum_{i=1}^{6} \rho_i LVOL_{t-i} + \acute{\beta} X_{t-1} + \epsilon_t. \quad [10]$$

3.3.3 Regularized models

In a multivariate OLS regression model, adding more predictors lowers the model's bias in the fitting sample but raises its variance on new data. This phenomenon, called overfitting, can reduce the signal-to-noise ratio because the model fits random noise instead of genuine information and patterns (Christensen et al., 2023). A common way to limit the overfitting problem, is to introduce a penalty term to the loss function (Hoerl and Kennard 1970; Tibshirani 1996; Zou and Hastie 2005) to minimize the residual sum of squares between the predicted values and the actual values (Maglaras et al., 2024). The penalized loss function is:

$$\tilde{\mathcal{L}}(\alpha, \beta, \theta) = \mathcal{L}(\alpha, \beta) + \phi(\beta; \theta), \quad [11]$$

where α indicates the intercept, β represents the vector of regression coefficients, $\phi(\beta; \theta)$ is the penalty term, and θ is the vector of hyperparameters, which determines how strongly the penalty is applied (Christensen et al., 2023). There are two main types of regularization terms, L_2 and L_1, which are explained in the following models.

Ridge regression (RR)

Ridge regression was originally introduced by Hoerl and Kennard (1970) as a method to enhance the generalisation of linear models. This is achieved by including an L_2 regularization (or penalty) term in the loss function, which shrinks less important parameters towards zero, but never exactly reaching zero. The penalty term is expressed as follows:

$$\phi(\beta; \lambda) = \lambda \sum_{i=1}^{J} \beta_i^2, \quad \lambda \geq 0 \quad [12]$$

where λ is the single hyperparameter, which controls the strength of shrinkage and can take any value from zero to infinity. A larger value of λ results in a stronger penalty and, therefore, produces a smaller coefficient (Maglaras et al., 2024). The hyperparameter optimization process in our study is based on a training set and a validation set for all models, which is explained in detail in the subsequent sections.

Ridge regression can be used with time series data by including lagged values of predictors as features (Maglaras et al., 2024). In our study, we applied ridge regression using the same structure as the OLS regression (Equation [9]). Specifically, for quarterly sampling, we created two lagged values of log realized volatility and one lagged value of each macroeconomic predictor. For monthly sampling, we used six lagged values of log realized volatility and one lagged value of each predictor. All these lagged variables are treated as features (input variables) in our forecasting models.

Ridge regression has been widely used in forecasting studies, including Christensen et al. (2023), Rahimikia and Poon (2024), Carr et al. (2019), Bianchi et al. (2020), and Gu et al. (2019). We employed this model in our study because it effectively reduces overfitting by shrinking coefficient estimates, especially when working with a moderate number of predictors. Although the predictors in our dataset are not highly correlated, ridge still helps stabilize the model and can improve out-of-sample forecasting performance by controlling the impact of less informative variables.

Least absolute shrinkage and selection operator (lasso)

Lasso was originally proposed by Tibshirani (1996) and is designed to select only the most relevant predictors by shrinking less important coefficients towards zero. It uses an L_1 regularization term, which, unlike ridge regression, can force some coefficients to become exactly zero (Zhu et al., 2023). This gives lasso the advantage of performing automatic variable selection by removing predictors that contribute little to the model (Niu et al., 2023). The penalty term for lasso is defined as:

$$\phi(\beta;\lambda) = \lambda \sum_{i=1}^{J} |\beta_i|, \quad \lambda \geq 0 \qquad [13]$$

where λ is the tuning hyperparameter and controls the intensity of shrinkage. Larger values of λ lead to stronger shrinkage and more coefficients being reduced to zero. In addition to

variable selection, lasso helps reduce overfitting by simplifying the model structure, especially when working with many predictors.

We include both ridge regression and lasso in our study because prior research shows no consistent evidence that one method always outperforms the other (Tibshirani, 1996, Fu, 1998). Like ridge regression, the lasso model in our study, uses lagged variables as features and applies the same regression structure described in Equation [9]. Hyperparameter tuning is based on a training set and a validation set. The lasso technique has been applied in forecasting and financial studies including Gu et al. (2019), Bianchi et al. (2020), Niu et al. (2023), Zhang et al. (2024), Zhu et al. (2023), Christensen et al. (2023), and Wu et al. (2021).

Elastic net (EN)

Elastic net was introduced by Zou and Hastie (2005) as a regularization method that combines the strengths of both ridge regression and lasso. It applies a mixed penalty: the L_2 regularization from ridge (which shrinks coefficients) and the L_1 regularization from lasso (which induces sparsity by setting some coefficients to zero) (Maglaras et al., 2024). The elastic net penalty function is defined as:

$$\phi(\beta; \lambda, \alpha) = \lambda(\alpha \sum_{i=1}^{J} \beta_i^2 + (1-\alpha) \sum_{i=1}^{J} |\beta_i|), \qquad \lambda \geq 0 \qquad [14]$$

where $\alpha \in [0, 1]$ is the second hyperparameter (in addition to λ), which determines the balance between the ridge and lasso components. When $\alpha = 1$, the model reduces to ridge regression, and when $\alpha = 0$, it becomes equivalent to lasso. Like ridge and lasso, both hyperparameters (λ and α) are tuned based on a training and a validation data split. Elastic net has been applied in previous forecasting studies such as Christensen et al. (2023), Niu et al. (2024), Zhu et al. (2023), Filipovic and Khalilzadeh (2021), Niu et al. (2023), Bianchi et al. (2020), and Gu et al. (2019). We apply elastic net in our study because it gives a flexible balance between ridge and lasso. EN can improve the forecasting accuracy of volatility in our study by shrinking less important variables and removing the least useful ones, depending on the data.

3.3.4 Ensemble models

Ensemble models combine the predictions of multiple individual models to produce one stronger forecast. The idea is that by aggregating several weak or moderately accurate models, the final prediction is more reliable and robust than any single model (Zhou, 2012).

Ensemble methods can reduce overfitting (by averaging or sequential corrections) and capture complex non-linear relationships in the data that simple linear models might miss. In this study, we focus on two widely used ensemble methods: random forest (RF) and gradient boosted regression trees (GBRT), which are described in the following sections.

Random forest (RF)

Random forest (RF), introduced by Breiman (2001), is an ensemble learning method that builds multiple decision trees and combines their outputs to produce more accurate forecasts. Each tree in the forest is trained on a random sample of the training data, and a random subset of features is selected at each split. This approach reduces the correlation between trees and increases overall model stability. Compared to standard decision trees such as classification and regression tree (CART), random forest lowers generalization error by averaging across many trees, which helps reduce overfitting and variance (Breiman, 2001). This makes RF models particularly effective for forecasting non-linear patterns with moderate numbers of predictors.

In our implementation, bootstrapping with replacement is used to generate resampled training sets (Breiman, 2001) during hyperparameter tuning. The final forecast is obtained by averaging the predictions from all trees. Consistent with all models in this study, we use the same lagged variables (quarterly and monthly) as input features for the RF model. Additionally, instead of relying on the out-of-bag (OOB) error approach, we tune the model's hyperparameters using a separate validation set. The prediction function of the random forest regression model, following Breiman (2001), is given by:

$$\hat{f}_{RF}(x) = \frac{1}{B}\sum_{b=1}^{B} T(x; \Theta_b, D_b), \qquad [15]$$

where x indicates the input feature vector for which the forecast is made, B is the total number of trees in the forest, Θ_b represents the randomness used when building tree b, D_b is a bootstrap sample drawn from the training set, and $T(x; \Theta_b, D_b)$ is the prediction made by tree b.

Random forest technique for forecasting has been applied in several studies including Christensen et al. (2023), Luong and Dokuchaev (2018), Niu et al. (2024), Zhu et al. (2023), Carr et al. (2019), Niu et al. (2023), Krauss et al. (2017), Nõu et al. (2021), Bianchi et al. (2020), Gu et al. (2019) and Wu et al. (2021).

Gradient boosted regression trees (GBRT)

GBRT is a tree-based ensemble method proposed by Friedman (2001). Unlike bagging methods such as random forest, GBRT builds trees sequentially, with each new tree trained to correct the errors made by the ensemble so far (Friedman, 2001). This approach makes GBRT especially effective at refining predictions over time and often leads to better predictive accuracy than random forest technique (Caruana and Niculescu-Mizil, 2006).

The boosting process in GBRT constructs a series of shallow trees, each trained on the residuals left by the previous model. These residuals represent the negative gradients of the chosen loss function. At each stage, a new tree is added to reduce the remaining error. A learning rate parameter ν (also called the shrinkage factor) controls how much each tree contributes to the final model, helping to prevent overfitting while improving accuracy (Filipovic and Khalilzadeh, 2021).

The final prediction of the GBRT model after M boosting stages, following Friedman (2001), is:

$$\hat{F}_{GB}(x) = F_M(x) = F_0(x) + \sum_{m=1}^{M} \nu \gamma_m h_m(x), \quad [16]$$

where x indicates the input feature vector, $F_0(x)$ is the initial guess (typically the mean of the target variables) before any trees are added, M is the total number of boosting stages (or number of trees), ν $(0 < \nu \leq 1)$ is the learning rate controlling the contribution of each tree, γ_m is the weight for tree m, chosen to minimize the loss at that stage, $h_m(x)$ is the m-th weak learner (a shallow regression tree) fitted to the pseudo-residuals at stage m.

We apply GBRT in our study due to its strong performance in capturing non-linear relationships and reducing bias through iterative learning. The model uses the same lagged variables as features, consistent with previous models, and its hyperparameters are tuned using a training and validation set.

GBRT has been applied in a range of forecasting studies, including Christensen et al. (2023), Rossi (2018), Alessandretti et al. (2018), Niu et al. (2024), Wu et al. (2021), Zhu et al. (2023), Filipovic and Khalilzadeh (2021), Zhang et al. (2024), Krauss et al. (2017), Bianchi et al. (2020), and Gu et al. (2019).

3.3.5 Deep leaning model

Deep learning is a type of machine learning that uses neural networks with multiple layers to learn patterns in data. These models are useful for working with complex and time dependent data, such as financial time series (LeCun et al., 2015). Deep learning has become popular in both academic research and industry-level forecasting tasks because of its flexibility and strong performance. In this study, we employ one of the most widely used deep learning models for time series data: the long short-term memory (LSTM) network.

Long short-term memory (LSTM)

The models discussed so far use only a small number of lags: two lags for log realized volatility in quarterly data and six in monthly data, along with one-period lag for each predictor (consistent with Paye, 2012). This setup may miss important long-term patterns in the data. Since stock return volatility is known to be persistent, it is worth testing whether including more lags can improve forecasts of future volatility.

However, testing all our models with many lag combinations for each time period would require excessive time and computing power. Recurrent neural networks (RNNs) are specifically designed to model such time dependencies by retaining information from previous time steps through internal states. But regular RNNs usually cannot remember information from earlier time steps because their memory fades over time (known as vanishing gradient problem) (Filipovic and Khalilzadeh, 2021). LSTM, which is a specialized version of RNNs and was first introduced by Hochreiter and Schmidhuber (1997), solves this problem by using special gates that help decide what information to keep, forget, or pass on. Like RNNs, LSTMs

work by passing information through repeating network units, but their structure is more advanced and better at capturing long term relationships.

A unit in LSTM model includes a memory cell and three gates: the input gate, forget gate, and output gate (Rahimikia and Poon, 2024). Figure 3.4 in the following, which we created but is inspired by Filipovic and Khalilzadeh (2021), shows a schematic diagram of an LSTM unit.

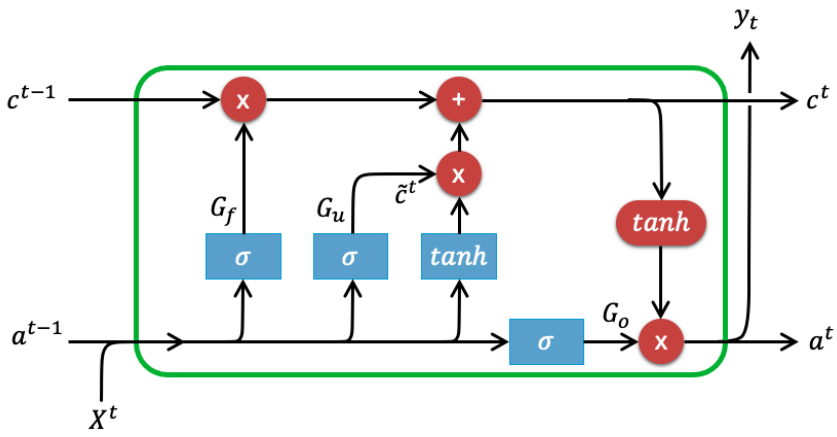

Figure 3.4 A long short-term memory (LSTM) unit. The figure illustrates the input, forget, and output gates, along with memory cell state and hidden state updates.

In this unit, the memory cell (represented by the horizontal line from c^{t-1} to c^t) stores information across time steps. The three gates control how information flows in and out of this memory cell (Rahimikia and Poon, 2024). The candidate value for the memory cell, follwoing Hochreiter and Schmidhuber (1997) and Rahimikia and Poon (2024)[3], is computed as:

$$\tilde{c}^t = tanh(w_c[a^{t-1}, X^t] + b_c), \qquad [17]$$

[3] All equations (17, 18, 19, 20, 21, 22) related to LSTM model are directly adapted from Rahimikia and Poon (2024).

where w_c and b_c are the weight and bias values in a memory cell, $[a^{t-1}, X^t]$ is the combination of the previous hidden state and current input, where a^{t-1} is the hidden state at $t-1$ and X^t is the input vector at t, and $tanh$ is the hyperbolic tangent activation function.

Following Rahimikia and Poon (2024), the gates are defined as:

$$G_u = \sigma(w_u[a^{t-1}, X^t] + b_u), \quad [18]$$

$$G_f = \sigma\big(w_f[a^{t-1}, X^t] + b_f\big), \quad [19]$$

$$G_o = \sigma(w_o[a^{t-1}, X^t] + b_o), \quad [20]$$

where σ is the sigmoid activation function. G_u, G_f, and G_o are the input, forget, and output gates respectively. Following Rahimikia and Poon (2024), the updated memory cell is computed as:

$$c^t = G_u \times \tilde{c}^t + G_f \times c^{t-1}, \quad [21]$$

where c^{t-1} indicates the previous memory state and $\tilde{c}^t$ is the new candidate value from Equation [17]. Finally, the hidden state output a^t, following Rahimikia and Poon (2024), is calculated as:

$$a^t = G_o \times \tanh(c^t), \quad [22]$$

which represents the part of the memory passed on to the next time step. All weights and biases are learned during the training process. LSTM helps avoid the vanishing gradient problem by using gates to control what information is remembered or forgotten. This allows the model to capture long term relationships in time series data (Rahimikia and Poon, 2024).

LSTM has been used in various studies aiming to improve forecasting accuracy through deep learning methods, including Moon and Kim (2019), McNally et al. (2018), Alessandretti et al. (2018), Filipovic and Khalilzadeh (2021), Rahimikia and Poon (2024), Zhang et al. (2024), Petrozziello et al. (2022), Bansal et al. (2022), and Nõu et al. (2021).

3.4 Evaluation criteria and statistical tests

This section describes the evaluation tools used to assess and compare the forecasting performance of our models. The main metric in this study is the mean squared prediction error (MSPE), which is used to tune hyperparameters during model training, to evaluate our

variable importance analysis, to calculate statistical tests, and to compute changes in out-of-sample R^2. The following subsections explain MSPE, R^2, and the statistical tests used in more detail.

3.4.1 Mean squared prediction error (MSPE)

Following Paye (2012), the mean squared prediction error (MSPE) is used to evaluate the accuracy of each model's one-step-ahead forecasts. It is defined as:

$$\hat{\sigma}_i^2 = P^{-1} \sum (LVOL_{t+1} - \widehat{LVOL}_{i,t+1})^2, \quad [23]$$

where P is the total number of out-of-sample forecasts within each sample period, $\widehat{LVOL}_{i,t+1}$ is the forecast of volatility from model i, and $LVOL_{t+1}$ is the actual observed value at time $t+1$.

3.4.2 R-squared (R^2)

To evaluate the economic relevance of each model's forecasts, we compute the change in out-of-sample R^2 of each forecasting model relative to the benchmark (univariate AR model). Following Campbell and Thompson (2008), Goyal and Welch (2008), Zhou et al. (2010), and Paye (2012), the out-of-sample R^2 for each model is calculated as:

$$R^2_{oos} = 1 - \frac{\hat{\sigma}_i^2}{\hat{\sigma}_0^2}, \quad [24]$$

where $\hat{\sigma}_i^2$ is the MSPE of model i and $\hat{\sigma}_0^2$ represents the MSPE of the simple historical average model. The change in out-of-sample R^2 between the forecasting model and the benchmark is calculated by:

$$\Delta R^2_{oos} = \left(1 - \frac{\hat{\sigma}_2^2}{\hat{\sigma}_0^2}\right) - \left(1 - \frac{\hat{\sigma}_1^2}{\hat{\sigma}_0^2}\right) = \frac{\hat{\sigma}_1^2 - \hat{\sigma}_2^2}{\hat{\sigma}_0^2}, \quad [25]$$

where $\hat{\sigma}_2^2$ is the MSPE of the forecasting model, and $\hat{\sigma}_1^2$ refers to the MSPE of the benchmark (the autoregressive model). This ΔR^2_{oos} is expressed as a percentage. A positive value indicates that the forecasting model performs better than the benchmark in terms of prediction accuracy.

3.4.3 Clark and West (CW) test

We apply the Clark and West (2007) test, following the approach used in Paye (2012), which is designed to compare a simple benchmark model (in our case, the univariate AR model) with a more complex, augmented model. Under the null, the augmented model's MSPE is no lower than the benchmark's, implying that the additional predictors do not improve out-of-sample forecasts (or as noted by Paye (2012), do not 'Granger-cause' volatility). While more complex models may include additional useful information, they can also produce more error in out-of-sample forecasts because they estimate more parameters. This extra error can make their forecast performance appear worse, even when they are more informative. The CW test adjusts for this issue by adding a correction term to the simple MSPE comparison. The test statistic, following Paye (2012), is calculated as:

$$CW = \hat{\sigma}_1^2 - \hat{\sigma}_2^2 + P^{-1}\sum(\widehat{LVOL}_{1,t+1} - \widehat{LVOL}_{2,t+1})^2, \quad [26]$$

where $\hat{\sigma}_1^2$ and $\hat{\sigma}_2^2$ are the MSPEs of the benchmark and forecasting models, respectively. The last term is the adjustment, which is the average of the squared differences between the two models' forecasts.

According to Paye (2012), if the CW statistic is significantly greater than zero, we can reject the null of no Granger-causality of the forecasting variables in volatility. This is a one-sided test, where the alternative hypothesis is that the forecasting model has a lower MSPE than the benchmark: $\hat{\sigma}_2^2 < \hat{\sigma}_1^2$.

3.4.4 Giacomini and White (GW) test

In addition to the CW test, we also apply the Giacomini and White (2006) test to evaluate the forecasting performance of our models, following the approach in Paye (2012). While the CW test compares different forecasting models, the GW test is more general and compares the performance of different forecasting methods. These methods may include differences in model estimation, forecast construction (e.g., rolling vs recursive windows), or data handling techniques. For example, in our study, we generate two forecasts using the same model; one based on a rolling window and another using a recursive window. Although the model is the same, the forecasting methods differ.

The GW test allows researchers to test forecasting performance using either unconditional comparisons (without extra information) or conditional ones (based on information available at time t) (Paye, 2012). The null hypothesis of the GW test is that, given the available information $\mathcal{G}_t$, the expected MSPE difference between the two models is zero:

$$H_0\colon\ E(\hat{\sigma}_1^2 - \hat{\sigma}_2^2|\mathcal{G}_t) = 0, \qquad [27]$$

where $\hat{\sigma}_1^2$ and $\hat{\sigma}_2^2$ are the MSPEs of the benchmark and forecasting models, respectively. The test statistic, following Paye (2012), is computed by:

$$GW = \frac{(\hat{\sigma}_1^2 - \hat{\sigma}_2^2)}{\hat{\sigma}_P / \sqrt{P}}, \qquad [28]$$

where $\hat{\sigma}_P$ is a heteroskedasticity and autocorrelation consistent (HAC) estimator of the asymptotic variance of the MSPE difference and P is the number of out-of-sample forecasts. Unlike the CW test, which is one-sided, the GW test is two-sided, meaning it checks whether either model is significantly better, not just whether one outperforms the other.

3.5 Forecasting settings

This section explains how the forecasting process is designed and implemented in this study. It describes the sample periods used for training and testing our models, the procedures for generating forecasts using different types of estimation windows (rolling and recursive), and the approach followed to tune model hyperparameters.

3.5.1 Sample periods

Our analysis uses several out-of-sample periods to test how well different models forecast volatility across various time periods. The main time periods include the original ones used in Paye (2012): 1947-2010, 1972-2010, 1982-2010, and 1972-2000. We expand these sample periods to include the most recent data, creating four new periods and serving as robustness check tests for our study: 1947-2023, 1947-2019, 1972-2023, and 1972-2019. Each period reflects a different economic environment, which can influence how macroeconomic variables affect the stock market and how well the models perform.

The longest period used in Paye (2012), 1947-2010, provides a broad historical view for testing volatility models. However, some forecasting variables are not available in the earlier

years of this period. To address this, Paye (2012) also examined the 1972-2010 period, which provides more complete data and includes additional variables. Two more periods, 1972-2000 and 1982-2010, are included in our analysis (consistent with Paye, 2012) to examine how the presence or absence of the volatile 1970s affects forecasting performance. This is important because, according to Goyal and Welch (2008), the oil shocks of 1973-1975 strongly impacted how well some economic variables predicted market behavior. Although both periods have the same number of observations, only the 1972-2000 timeframe includes the economic disruptions of the 1970s, such as stagflation, the collapse of the Bretton Woods system, and major changes in global oil markets (Paye, 2012).

To capture more recent events and long-term trends in financial markets, we include four extended forecasting periods in our analysis, to be used in our robustness check: 1947-2023, 1947-2019, 1972-2023, and 1972-2019. These timelines allow us to assess the impact of both pre and post pandemic environments on volatility forecasting. Table 3.3 lists all out-of-sample forecasting periods in our study, and Figure 3.5 indicates the major economic events that occurred within each sample period. To get a broader picture, Figure 3.6 presents the timeline of the most important economic shocks between 1970 and 2023.

Main sample periods	Sample periods for robustness check
1947-2010	1947-2023
1972-2010	1947-2019
1982-2010	1972-2023
1972-2000	1972-2019

Table 3.3 List of forecasting sample periods. Periods listed in the first column are directly adapted from Paye (2012) and represent our main sample periods. Those listed in the second column represent extended periods used for our robustness check.

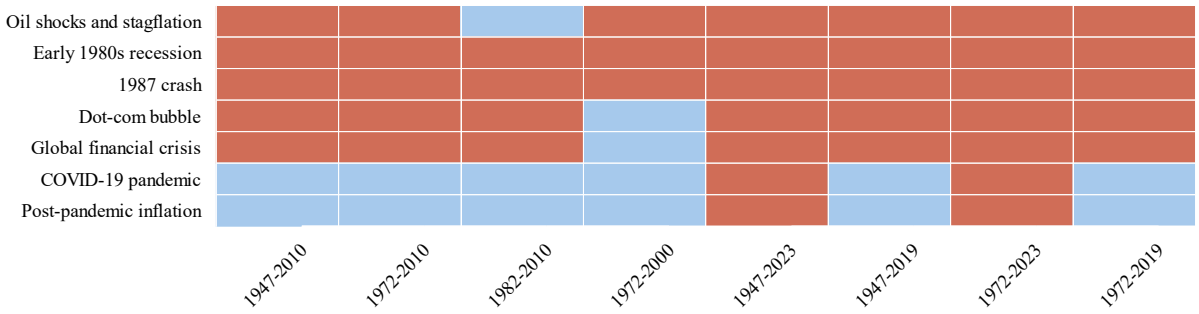

Figure 3.5 Coverage of major economic events across forecasting periods. The figure includes critical economic events such as the oil shocks of the 1970s, the recession of the 1980s, the crash of 1987, the dot-com bubble, the market crash of 2007-2008, the COVID-19 pandemic, and the inflationary period of the post-pandemic era. (red-shaded cells indicate which events are included in each out-of-sample window).

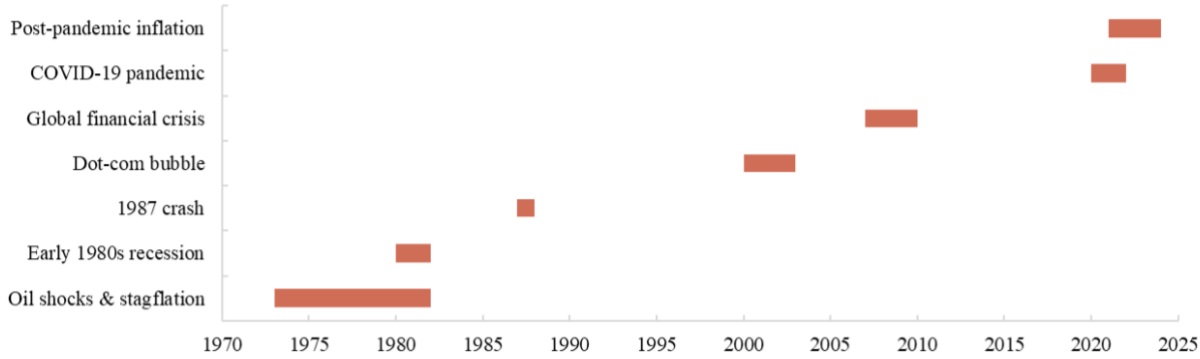

Figure 3.6 Economic events timeline (1970-2023). This figure presents the timeline of each critical economic event from 1970 to 2023. The events included in this figure are the same as those noted in Figure 3.5.

3.5.2 Estimation window strategies

Similar to Paye (2012), two common approaches are used in our study for model estimation over time: the rolling window and the recursive window. These methods allow the models to update their parameters dynamically as new data becomes available.

In the rolling window approach, we use a fixed-length sample of the most recent 20 years (equivalent to 80 quarters or 240 months) before each forecast date. This means the model is re-estimated using a moving window of the latest 20 years, which helps it adapt to possible structural changes or shifts in economic conditions. By contrast, the recursive window begins

with the initial 20-year sample and expands over time. As each new observation becomes available, it is added to the training data, allowing the model to learn from an increasingly larger dataset. For both methods, the models generate a one-step-ahead forecast for the log realized volatility, either the next quarter or next month, depending on the frequency of the data. Figure 3.7 represents a simplified schematic of the structure of the rolling window and the recursive window approaches.

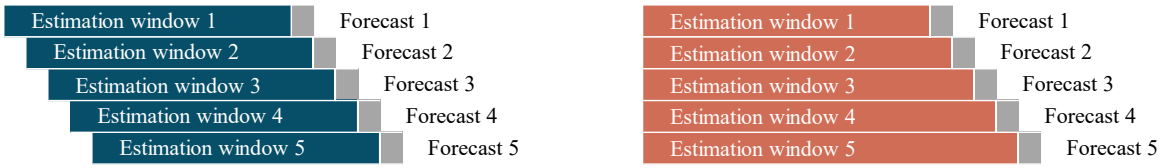

Figure 3.7 Illustration of estimation strategies. The estimation scheme on the left (in blue) indicates the rolling window approach, where the sample window moves forward by dropping the oldest observation and adding the newest. The scheme on the right (in red) represents the recursive window strategy, where the initial sample expands over time by adding new observations without dropping past data. The gray data points indicate the out-of-sample, one-period-ahead forecasts generated after each estimation window.

3.5.3 Hyperparameter tuning

Hyperparameters control a model's complexity, helping it generalize effectively by balancing bias and variance. However, selecting optimal hyperparameters is challenging, as existing guidance in literature is limited (Christensen et al., 2023). This section explains the hyperparameter tuning approach adopted in this research.

We optimize hyperparameters by splitting each estimation window into training and validation sets. Specifically, for each window (whether rolling or recursive, as described previously), 80% of data is used for training, and 20% for validation. To preserve the time order of our data, the validation set always come after the training set chronologically. Thus, standard k-fold cross-validation, which randomly splits data, is not appropriate for our time series analysis.

We first train the models on the training data with various hyperparameter combinations. Next, we calculate the mean squared prediction error (MSPE) on forecasts made for the validation

set, choosing hyperparameters with the lowest MSPE. Using these optimal hyperparameters, we generate one-step-ahead out-of-sample forecasts. The hyperparameter tuning process is repeated for each forecast in our study. We use the same training-validation split for all models to ensure consistency and fair comparison. Details about specific hyperparameters for each forecasting model are presented in Table 3.4.

Forecasting model	**Hyperparameters**	**Optimization search space**
Ridge and lasso	Lambda (λ)	$\lambda \in [10^{-4}, 10^{3}]$
Elastic net (EN)	Lambda (λ)	$\lambda \in [10^{-4}, 10^{3}]$
	Alpha (α)	$\alpha \in [10^{-4}, 1]$
Random forest (RF)	Max depth (d_{max})	$d_{max} \in \{3, 6, 9, 12, 15\}$
	Min samples split ($s_{\min-split}$)	$s_{\min-split} \in \{2, 5, 10, 20, 50\}$
	Min samples leaf ($s_{\min-leaf}$)	$s_{\min-leaf} \in \{1, 3, 5, 10, 20\}$
	Number of trees ($n_{\text{estimators}}$)	$n_{\text{estimators}} \in \{10, 50, 100\}$
	Max features (m)	$m \in \{\text{"sqrt"}, \text{"}log2\text{"}, 0.5\}$
GBRT	Learning rate (η)	$\eta \in \{0.01, 0.05, 0.1, 0.2, 0.3\}$
	Number of estimators ($n_{\text{estimators}}$)	$n_{\text{estimators}} \in \{50, 100, 200\}$
	Max depth (d_{max})	$d_{max} \in \{3, 5, 7, 9\}$
	Min child weight (w_{min})	$w_{min} \in \{1, 3, 5\}$
LSTM	LSTM units (u)	$u \in \{32, 64\}$
	LSTM layers (L)	$L \in \{1, 2\}$
	Batch size (b)	$b \in \{16, 32\}$
	Learning rate (η)	$\eta \in \{10^{-3}, 10^{-2}\}$
	Dropout rate (δ)	$\delta \in \{0.0, 0.2\}$
	Epochs (E)	$E \in \{50, 100\}$

Table 3.4 Hyperparameter search spaces for our forecasting models, showing the specific hyperparameters tuned for each model (ridge and lasso, elastic net, random forest, gradient boosted regression tree (GBRT), and long short-term memory (LSTM)), along with the ranges or sets of values explored during the optimization process.

In ridge, lasso, elastic net, and for consistency, random forest and GBRT, all features (or all lagged variables) are standardized so penalties or splits aren't biased by variables with big numbers. To be consistent with Paye (2012), the features are not standardized in OLS. We also skip the standardization process for LSTM to avoid extra computational costs, as its internal normalisation already learns the right scale. During hyperparameter tuning, we standardize the training set itself by calculating its mean and standard deviation for each feature. These training-set parameters are then applied to standardize the validation set to avoid data leakage. For generating out-of-sample forecasts, we standardize the entire estimation window (rolling or recursive) and use its parameters to standardize the data for forecasting. The general formula for standardizing a feature X is:

$$X_{std} = \frac{(X-\mu)}{\sigma}, \quad [29]$$

where μ is the mean of X, computed from the training set for tuning, or the entire estimation window for forecasting, and σ is the standard deviation of X, with any zero values replaced by one to avoid division by zero.

3.5.4 Variable importance analysis

One of the main objectives of our study is to assess whether financial and macroeconomic variables have incremental predictive power in forecasting stock return volatility. Accordingly, we perform a variable importance analysis to measure the relative contribution of each predictor in forecasting volatility.

Previous studies have adopted different approaches to measure variable importance, including SHAP analysis (Rahimikia and Poon, 2024; Niu et al., 2024). In our study, we employ the permutation feature importance method initially introduced by Breiman (2001), which is model-agnostic and can be used with any predictive model[4]. According to Strobl et al. (2007), permutation importance can be distorted when predictors are highly correlated. Our

[4] scikit-learn developers, "Permutation importance," in Scikit-learn: Machine learning in Python (User Guide, version 1.3.0), accessed August 1, 2025, https://scikit-learn.org/stable/modules/permutation_importance.html.

correlation analysis (Figure 3.3), however, shows that no pair of forecasting variables in our study has an absolute correlation coefficient greater than 0.70.

This method measures the importance of each predictor by randomly permuting its values and therefore disrupting its relationship with the target variable, realized volatility. The rationale is that a significant increase in forecast error following permutation indicates high variable importance.

Specifically, we assess the feature importance using the mean squared prediction error (MSPE), which is the same evaluation metric used in hyperparameter tuning and statistical tests in our study. For each estimation window, the model is first fitted on the first 80% of the data (training set) with all features included. Permutation importance is then computed on the next 20% (validation set) by randomly permuting each feature and breaking its association with the target and measuring the changes in MSPE. For each feature, the increase in validation MSPE is averaged across multiple random permutations to get ΔMSPE. The variables are then ranked by their ΔMSPE as a measure of their contribution to forecasting volatility.

The variable importance analysis in our study will be applied only to the machine learning models with better predictive performance among others, to avoid unnecessary computational costs. Further details regarding the specific models selected and additional settings for the variable importance analysis will be provided in the results chapter.

Next chapter presents the empirical results of our study in detail and then discusses our observations in the context of the existing literature, highlighting where our findings confirm, extend, or challenge earlier work.

Chapter 4. Empirical Results, Analysis, and Discussion

In this thesis, we apply several machine learning methods augmented with macroeconomic variables used in Paye (2012) to forecast stock return volatility. Specifically, we test the predictive accuracy of our forecasting models in comparison to an autoregressive benchmark. This chapter presents the empirical findings of our study, highlighting the results of the statistical tests and evaluation metrics described in the previous chapter. Following the numerical results and their interpretation, we discuss our key findings in the context of existing literature and empirical evidence. Finally, the chapter concludes with the implications of our research, its limitations, and recommendations for future studies.

4.1 Quarterly out-of-sample forecasting performance (main analysis and robustness check)

The results of the one-period-ahead forecasts for quarterly sampling are presented in this section. Tables 4.1 and 4.2 show the results for two estimation methods: rolling window and recursive window, respectively. Each table has two parts: Panel A and Panel B. Panel A presents the forecasting results across four main time horizons (those covered in Paye, 2012), and Panel B demonstrates the results for extended time periods, serving as our robustness checks. For every forecasting model, the tables present the Clark and West (CW) test, which measures the adjusted difference in the mean squared prediction error (MSPE) of the forecasting model and the benchmark AR(2) model. The significance levels alongside the CW test values, shown as *, **, and ***, reflect the rejection of the null hypothesis of no Granger causality from the macroeconomic variables in the model at the conventional 0.90, 0.95, and 0.99 confidence levels, respectively.

The tables also report the changes in out-of-sample R^2 values of the forecasting models compared to the benchmark AR(2), presented as ΔR^2. This metric directly measures how well each model forecasts relative to the benchmark. Positive values for ΔR^2 indicate the forecasting model predicts volatility more accurately than the benchmark, while negative values indicate the benchmark performs better. GW test results are reported alongside ΔR^2 values as asterisks. This test assesses whether to reject the null hypothesis of equal predictive

accuracy between the forecasting model and the benchmark, regardless of direction, at the 0.90, 0.95, and 0.99 confidence levels.

4.1.1 Rolling window results

Panel A: Results of main time periods

	1947Q3–2010Q4 N=254		1972Q3–2010Q4 N=154		1982Q3–2010Q4 N=114		1972Q3–2000Q4 N=114	
Forecasting Model	***CW***	ΔR^2	***CW***	ΔR^2	***CW***	ΔR^2	***CW***	ΔR^2
OLS Regression	8.18	-23.05*	16.96	-21.96	-3.51	-25.87*	38.30***	-4.91
Ridge Regression (RR)	0.20	-14.35*	2.25	-16.10*	-0.67	-21.12*	4.81	-7.67**
Lasso	4.63	-6.77**	0.83	-12.19*	-7.83	-15.59**	10.46**	-3.76
Elastic Net (EN)	4.07	-7.47**	-0.32	-11.67*	-8.96	-16.19**	10.62**	-3.38
Random Forest (RF)	5.17	-9.19**	7.69*	-4.72	0.93	-11.06***	0.78	-10.65***
Gradient Boosted (GBRT)	17.72**	-8.46*	21.78**	-10.99**	24.42***	-4.75	27.27**	-9.33
Long Short-Term Memory (LSTM)	8.48**	-1.30	21.75**	-3.68	7.72	-4.67	31.70***	-0.57

Panel B: Results of extended time periods (as robustness check)

	1947Q3–2023Q4 N=306		1947Q3–2019Q4 N=290		1972Q3–2023Q4 N=206		1972Q3–2019Q4 N=190	
Forecasting Model	***CW***	ΔR^2	***CW***	ΔR^2	***CW***	ΔR^2	***CW***	ΔR^2
OLS Regression	2.95	-26.76*	9.96	-20.08*	2.5	-37.41**	18.60*	-17.06*
Ridge Regression (RR)	-4.08	-17.07*	3.89	-11.54*	-9.04	-29.10*	4.20	-15.68*
Lasso	-4.62	-14.37**	4.71	-6.73**	-7.64	-20.42*	0.75	-10.92**
Elastic Net (EN)	-4.24	-15.89*	5.75	-5.90**	-7.88	-21.17*	-0.21	-10.45**
Random Forest (RF)	4.37	-7.30***	3.53	-9.27***	6.14	-10.37***	6.08	-9.53**
Gradient Boosted (GBRT)	9.77*	-10.88***	22.13***	-6.63	26.03***	-8.66*	24.80***	-7.71
Long Short-Term Memory (LSTM)	9.83**	-1.66	8.94**	-0.82	18.16**	-3.33	22.12***	-2.32

Table 4.1 Out-of-sample forecasting results, quarterly data and rolling estimation. The table reports CW test statistics (Clark and West, 2007), calculated as adjusted differences in MSPE multiplied by 1,000, which are used to assess equal predictive accuracy relative to the AR(2) benchmark and, in this context, to test for Granger causality. Symbols ***, **, and * indicate rejection of the null hypothesis at the 1%, 5%, and 10% significance levels, respectively. The table also presents ΔR^2, measuring the improvement in out-of-sample R^2 over the AR(2) benchmark, with significance determined by the Giacomini-White (2006) test using the same notation. Forecasts are generated using a rolling estimation window of 80 quarters (or 20 years). Panel A reports results for the main sample periods, and Panel B shows the robustness check results for extended time periods.

As shown in Table 4.1, across all eight time periods, the CW test is positive and statistically significant predominantly for the gradient boosted regression tree (GBRT) and long short-term memory (LSTM) models, rejecting the null hypothesis of no Granger causality. This suggests that these two models can capture the informational content of the macroeconomic variables for volatility series. However, for the rest of the models across all time periods, the CW test results are mainly a mixture of negative or insignificant positive values.

Looking at the values of ΔR^2, the results are strongly negative for all linear models (OLS, ridge, lasso, and elastic net (EN)) across all time horizons, and also for the ensemble models (random forest (RF) and gradient boosted (GBRT)) in some time periods, which means that the AR(2) benchmark provides more accurate predictions. The LSTM has the lowest negative values of ΔR^2, all statistically insignificant, suggesting that for the quarterly sampling and rolling window, this model has performed better than other models in forecasting volatility, although still underperforming the benchmark.

The contradiction between CW values (positive) and the ΔR^2 values (negative), e.g., the results of GBRT for 1947Q3-2010Q4, implies that the predictive performance of the forecasting model is worse than the benchmark but not to the extent that we can conclude the predictors lack useful information, or, in the terms of Paye (2012), that they do not Granger cause volatility.

Comparing the results for different time periods employed, among the first four sample periods adapted originally from Paye (2012): 1947Q3-2010Q4, 1972Q3-2010Q4, 1982Q3-2010Q4, and 1972Q3-2000Q4, the worst results generally belong to 1982Q3-2010Q4, with more negative CW test values and stronger negative ΔR^2 values. This time period, unlike the others, excludes the turbulent environment of 1970s (oil shock). This suggests that economic shocks and market turbulence help models learn from significant volatility fluctuations. When the 1970s oil shock is excluded, there is less variation for the models to detect, leading to inferior forecasts relative to the AR(2) benchmark. This highlights the importance of including periods of economic turbulence when evaluating and comparing volatility forecasting models.

Analyzing the extended time periods covered only in our study and considered as robustness check: 1947Q3-2023Q4, 1947Q3-2019Q4, 1972Q3-2023Q4, and 1972Q3-2019Q4, which

include pre- and post-pandemic data, the results predominantly indicate negative or weaker positive CW test and stronger negative ΔR^2 values for the two periods including post-2019 data. This may suggest that the COVID-19 pandemic changed volatility patterns in ways that our forecasting variables could not capture effectively. In summary, in our analysis using quarterly data and rolling estimation approach, none of the forecasting models was able to forecast volatility better than the AR(2) benchmark across all time horizons. The LSTM model came closest to consistently matching the benchmark; however, it did not clearly outperform it.

4.1.2 Recursive window results

Panel A: Results of main time periods

	1947Q3–2010Q4 N=254		1972Q3–2010Q4 N=154		1982Q3–2010Q4 N=114		1972Q3–2000Q4 N=114	
Forecasting Model	***CW***	***ΔR²***	***CW***	***ΔR²***	***CW***	***ΔR²***	***CW***	***ΔR²***
OLS Regression	11.70***	-0.74	30.12***	-8.24	12.07**	-12.1	31.21***	-0.74
Ridge Regression (RR)	12.53***	0.33	13.05***	-9.03	10.23**	-11.34	7.68*	-1.72
Lasso	10.61***	0.41	13.80**	-9.69	13.41**	-11.29	9.70*	-1.74
Elastic Net (EN)	11.43***	0.17	10.08**	-10.08	12.09**	-11.52	2.57	-4.13
Random Forest (RF)	3.81	-5.79**	0.46	-7.42**	3.81	-7.86***	0.90	-10.11**
Gradient Boosted (GBRT)	6.64	-11.12***	17.55**	-9.39**	8.12	-13.18**	16.17*	-12.92*
Long Short-Term Memory (LSTM)	3.12	-3.82**	19.66**	-1.10	6.25	-4.36	24.86**	-0.69

Panel B: Results of extended time periods (as robustness check)

	1947Q3–2023Q4 N=306		1947Q3–2019Q4 N=290		1972Q3–2023Q4 N=206		1972Q3–2019Q4 N=190	
Forecasting Model	***CW***	***ΔR²***	***CW***	***ΔR²***	***CW***	***ΔR²***	***CW***	***ΔR²***
OLS Regression	8.93***	-0.47	12.83***	0.29	26.47***	-6.83	27.65***	-6.06
Ridge Regression (RR)	9.74***	-0.58	13.68***	1.22	14.16**	-7.87**	14.25***	-5.48
Lasso	8.84***	0.07	11.27***	1.02	6.59	-4.33**	12.78**	-7.49
Elastic Net (EN)	9.83***	-0.08	12.24***	0.90	8.94*	-6.02*	10.25***	-6.64
Random Forest (RF)	8.32*	-4.30*	8.16*	-3.47	-0.74	-9.84***	-12.68	-14.94***
Gradient Boosted (GBRT)	8.99	-10.46***	3.27	-13.82***	8.17	-12.34***	21.85***	-8.63*
Long Short-Term Memory (LSTM)	5.41*	-2.48	5.75*	-2.42	10.68	-3.65	20.20***	-0.10

Table 4.2 Out-of-sample forecasting results, quarterly data and recursive estimation. The table reports CW test statistics (Clark and West, 2007), calculated as adjusted differences in MSPE multiplied by 1,000, which are used to assess equal predictive accuracy relative to the AR(2) benchmark and, in this context, to

test for Granger causality. Symbols ***, **, and * indicate rejection of the null hypothesis at the 1%, 5%, and 10% significance levels, respectively. The table also presents ΔR^2, measuring the improvement in out-of-sample R^2 over the AR(2) benchmark, with significance determined by the Giacomini-White (2006) test using the same notation. Forecasts are generated using a recursive estimation approach, with an initial window length of 80 quarters (20 years). Panel A reports results for the main sample periods, and Panel B shows the robustness check results for extended time periods.

Looking at the results from quarterly sampling and the recursive window approach presented in Table 4.2, the results of the CW test are mainly positive and statistically significant for the linear models (OLS, ridge regression, lasso, and elastic net) across all time periods. This contrasts with the CW test results for the linear models using rolling window estimation shown in Table 4.1. For more complex models, including the tree-based models (random forest and gradient boosted regression tree) and the LSTM, CW test results indicate a mixture of strongly positive and weak values, with random forest performing the worst. Overall, it seems that, unlike the linear models, the GBRT and LSTM models deliver better CW test results under the rolling window approach. Random Forest results, however, indicate no major changes under the rolling and recursive estimation approaches.

Comparing the results across the first four time periods, following Paye (2012), the ΔR^2 values are closest to zero for 1947Q3-2010Q4 and 1972Q3-2000Q4 for the linear models. For the extended periods of our robustness check, which include data from before and after the COVID-19 pandemic, ΔR^2 results for the linear models are slightly better for 1947Q3-2023Q4 and 1947Q3-2019Q4 than for the two other periods starting from 1972Q3 and ending in 2023Q4 and 2019Q4, respectively.

In short, based on the results from quarterly sampling and the recursive window, all linear models (OLS and penalized linear models) and the long short-term memory (LSTM) performed better than the ensemble models (random forest and gradient boosted) across all time periods. While the LSTM performed roughly the same under both estimation windows in terms of prediction accuracy, the linear models clearly improved from the rolling to the recursive window. Despite these findings, none of the models across all time periods outperformed the benchmark AR(2) model.

4.2 Monthly out-of-sample forecasting performance (main analysis and robustness check)

Tables 4.3 and 4.4 in this section present the results of one-period-ahead forecasts for monthly data using two estimation methods: rolling window (Table 4.3) and recursive window (Table 4.4). Each table has two parts: Panel A and Panel B. Panel A presents the forecasting results across four main time horizons (those covered in Paye, 2012), and Panel B demonstrates the results for extended time periods, serving as our robustness checks. The tables report the same evaluation metrics as those used for the quarterly data in the previous section. ΔR^2 values indicate the out-of-sample prediction accuracy of the forecasting models relative to the benchmark AR(6) model.

4.2.1 Rolling window results

A general look at Table 4.3 below shows that, under the rolling window approach, the out-of-sample monthly forecasts perform better than the quarterly forecasts (as shown in Table 4.1), in terms of CW test statistics and ΔR^2 values. Specifically, the CW test results from Table 4.3 are predominantly positive and statistically significant in nearly every sample, implying strong Granger causality from financial and macroeconomic variables at higher frequency. In contrast, for the quarterly data using the same windowing approach, only the GBRT and LSTM models produced significantly positive CW test results.

Panel A: Results of main time periods

	1947.3–2010.12 N=759/766*		**1972.3–2010.12 N=459/466**		**1982.3–2010.12 N=339/346**		**1972.3–2000.12 N=346**	
Forecasting Model	***CW***	***ΔR²***	***CW***	***ΔR²***	***CW***	***ΔR²***	***CW***	***ΔR²***
OLS Regression	7.13**	-1.57	14.05***	0.18	11.43**	0.12	9.85***	-0.26
Ridge Regression (RR)	6.97***	-1.29	12.98***	0.70	12.48**	1.27	9.29***	-0.14
Lasso	4.67**	-1.12	8.56***	-0.33	8.19**	-0.21	5.23*	-1.78
Elastic Net (EN)	6.08***	-0.76	9.64***	0.33	9.47**	0.54	6.88**	-0.78
Random Forest (RF)	3.04	-5.74***	7.11**	-3.39**	10.35***	-2.28	8.71**	-3.11*
Gradient Boosted (GBRT)	5.44*	-10.41***	6.57*	-9.58***	12.13**	-7.09***	10.96**	-7.28**
Long Short-Term Memory (LSTM)	8.84***	-2.01	15.24***	0.55	14.48***	1.06	15.16***	0.43

*759/766 indicates that, due to NaN values (i.e., some missing data), our models produced fewer forecasts than expected. This applies to other time periods as well, as indicated in their respective columns.

Panel B: Results of extended time periods (as robustness check)

	1947.3–2023.12 N=911/922		1947.3–2019.12 N=867/874		1972.3–2023.12 N=611/622		1972.3–2019.12 N=567/574	
Forecasting Model	***CW***	ΔR^2	***CW***	ΔR^2	***CW***	ΔR^2	***CW***	ΔR^2
OLS Regression	7.32***	-1.80	8.83***	-0.72	11.72***	-1.31	15.28***	1.11
Ridge Regression (RR)	9.21***	-0.99	8.73***	-0.63	13.99***	0.23	13.91***	0.96
Lasso	4.16**	-2.04**	6.42***	-0.47	7.49***	-1.71	10.49***	0.54
Elastic Net (EN)	4.89**	-2.21**	7.60***	-0.38	9.00***	-1.49	11.52***	0.86
Random Forest (RF)	3.10	-8.30***	1.07	-7.50***	5.86*	-6.03***	8.94**	-3.92**
Gradient Boosted (GBRT)	9.64***	-7.98***	4.27	-12.12***	14.74***	-5.29**	9.40**	-7.61***
Long Short-Term Memory (LSTM)	8.48***	-2.86**	8.09***	-2.35**	11.31***	-1.75	12.84***	-0.60

Table 4.3 Out-of-sample forecasting results, monthly data and rolling estimation. The table reports CW test statistics (Clark and West, 2007), calculated as adjusted differences in MSPE multiplied by 1,000, which are used to assess equal predictive accuracy relative to the AR(6) benchmark and, in this context, to test for Granger causality. Symbols ***, **, and * indicate rejection of the null hypothesis at the 1%, 5%, and 10% significance levels, respectively. The table also presents ΔR^2, measuring the improvement in out-of-sample R^2 over the AR(6) benchmark, with significance determined by the Giacomini-White (2006) test using the same notation. Forecasts are generated using a rolling estimation window of 240 months (or 20 years). Panel A reports results for the main sample periods, and Panel B shows the robustness check results for extended time periods.

The results from the ΔR^2 metric suggest that the linear models (OLS and all penalized regressions) and the LSTM have performed better than the ensemble models (random forest and GBRT). Across all time periods, we even observe some positive but statistically insignificant ΔR^2 values for these best-performing models. Notably, the ΔR^2 values presented in this table show considerable improvements from their equivalent metrics under the rolling window for quarterly data (Table 4.1).

Among the original time periods adopted from Paye (2012), the results for the periods 1972Q3-2010Q4 and 1982Q3-2010Q4 are slightly better than those for 1947Q3-2010Q4 and 1972Q3-2000Q4, in terms of prediction accuracy (ΔR^2). Regarding the extended time periods and our robustness check, although the results are very close, the longest one (1947Q3-2023Q4) shows slightly worse performance than the rest. Moreover, the results for the two

periods that do not include pandemic data (ending at 2019Q4) are slightly better than those that include post-pandemic data (extending through 2023Q4).

In summary, at the monthly frequency and under the rolling window estimation approach, all models demonstrate strong CW test results. The ΔR^2 values for the linear models and LSTM are close to zero, positive or negative, but statistically insignificant. Overall, both metrics (CW test and ΔR^2) indicate that our models perform better on monthly data than on quarterly data using the same estimation methods.

4.2.2 Recursive window results

Panel A: Results of main time periods

	1947.3–2010.12 N=759/766		1972.3–2010.12 N=459/466		1982.3–2010.12 N=339/346		1972.3–2000.12 N=346	
Forecasting Model	***CW***	***ΔR²***	***CW***	***ΔR²***	***CW***	***ΔR²***	***CW***	***ΔR²***
OLS Regression	5.69***	0.19	8.90***	0.96	6.67**	0.68	9.10***	0.71
Ridge Regression (RR)	6.81***	0.43	8.96***	1.12	8.55***	1.05	9.33***	1.12
Lasso	4.21***	0.18	7.86***	1.04	4.99**	0.09	7.15***	0.46
Elastic Net (EN)	5.39***	0.12	8.29***	1.02	7.03**	0.58	7.64***	0.34
Random Forest (RF)	0.56	-5.01***	1.53	-4.32***	8.86**	-1.87	5.38**	-3.33**
Gradient Boosted (GBRT)	0.90	-4.95***	6.71**	-4.35***	11.70***	-3.16*	10.58***	-2.02
Long Short-Term Memory (LSTM)	5.64**	-2.98***	11.83***	0.07	6.45***	-0.86	16.17***	2.19

Panel B: Results of extended time periods (as robustness check)

	1947.3–2023.12 N=911/922		1947.3–2019.12 N=867/874		1972.3–2023.12 N=611/622		1972.3–2019.12 N=567/574	
Forecasting Model	***CW***	***ΔR²***	***CW***	***ΔR²***	***CW***	***ΔR²***	***CW***	***ΔR²***
OLS Regression	5.01***	0.40	6.02***	0.46	6.72***	0.01	8.62***	1.19
Ridge Regression (RR)	6.33***	0.48	7.01***	0.68	6.03***	-0.06	8.35***	1.14
Lasso	4.60***	0.29	4.72***	0.44	5.22***	0.12	7.04***	0.95
Elastic Net (EN)	5.39***	0.15	5.79***	0.40	5.34***	0.01	7.23***	0.87
Random Forest (RF)	3.46	-3.83***	2.18	-4.76***	2.62	-4.27***	1.01	-4.59***
Gradient Boosted (GBRT)	1.35	-5.62***	1.93	-5.32***	6.64**	-4.37***	7.92***	-3.77**
Long Short-Term Memory (LSTM)	9.07***	-1.60	8.17***	-1.88*	12.51***	1.33	11.67***	0.42

Table 4.4 Out-of-sample forecasting results, monthly data and recursive estimation. The table reports CW test statistics (Clark and West, 2007), calculated as adjusted differences in MSPE multiplied by 1,000, which are used to assess equal predictive accuracy relative to the AR(6) benchmark and, in this context, to test for Granger causality. Symbols ***, **, and * indicate rejection of the null hypothesis at the 1%, 5%,

and 10% significance levels, respectively. The table also presents ΔR^2, measuring the improvement in out-of-sample R^2 over the AR(6) benchmark, with significance determined by the Giacomini-White (2006) test using the same notation. Forecasts are generated using a recursive estimation approach, with an initial window length of 240 months (20 years). Panel A reports results for the main sample periods, and Panel B shows the robustness check results for extended time periods.

The results from Table 4.4 clearly show that all models perform best under the recursive window with monthly data. The CW test statistic is predominantly and strongly positive across all time periods, indicating Granger causality and confirming the informativeness of the predictors for forecasting volatility.

In addition, the ΔR^2 values for almost all linear models are positive across all time periods but remain statistically insignificant. This supports our earlier finding from the quarterly results that moving from a rolling to a recursive window improves the predictive performance of linear models. The LSTM model performs similarly to its results under the monthly rolling window, with ΔR^2 values close to zero, either slightly positive or negative. However, the two ensemble models (random forest and GBRT) achieve their best results here compared to all previous setup results. Their ΔR^2 values mostly range from -3% to -5%, which is slightly better than in the monthly rolling window setting. This supports the idea that ensemble models, like linear models, also benefit from more data, both in terms of frequency and estimation window length. Comparing different time periods, we observe little variation in the results.

To summarize, although the linear models (OLS, ridge, lasso, and r net) and the LSTM model show the highest number of positive ΔR^2 values, none of them outperform the benchmark AR(6) model in a statistically significant way. This suggests that these models mostly match the benchmark AR models in terms of prediction accuracy. The fact that even these positive gains lack statistical significance highlights the difficulty of consistently outperforming a univariate autoregressive benchmark, even with monthly data. Random Forest and GBRT perform better in prediction and manage to narrow the gap in out-of-sample R^2 compared to the benchmark.

4.3 Overall model comparison

In the previous sections, we presented and interpreted the evaluation metrics and statistical test results of our forecasting models in detail. To provide a broader view of whether the complex and augmented forecasting models used in this study outperformed the univariate autoregressive benchmarks (AR(2) for quarterly and AR(6) for monthly data), we summarize their predictive performance (ΔR^2) in Figure 4.1.

Figure 4.1 illustrates the average ΔR^2 values for each model, calculated across all time horizons (the main four periods and the four extended periods for robustness check) within four estimation windowing: quarterly rolling, quarterly recursive, monthly rolling, and monthly recursive. As noted earlier, a positive ΔR^2 indicates superior forecast accuracy relative to the benchmark, while a negative ΔR^2 reflects underperformance.

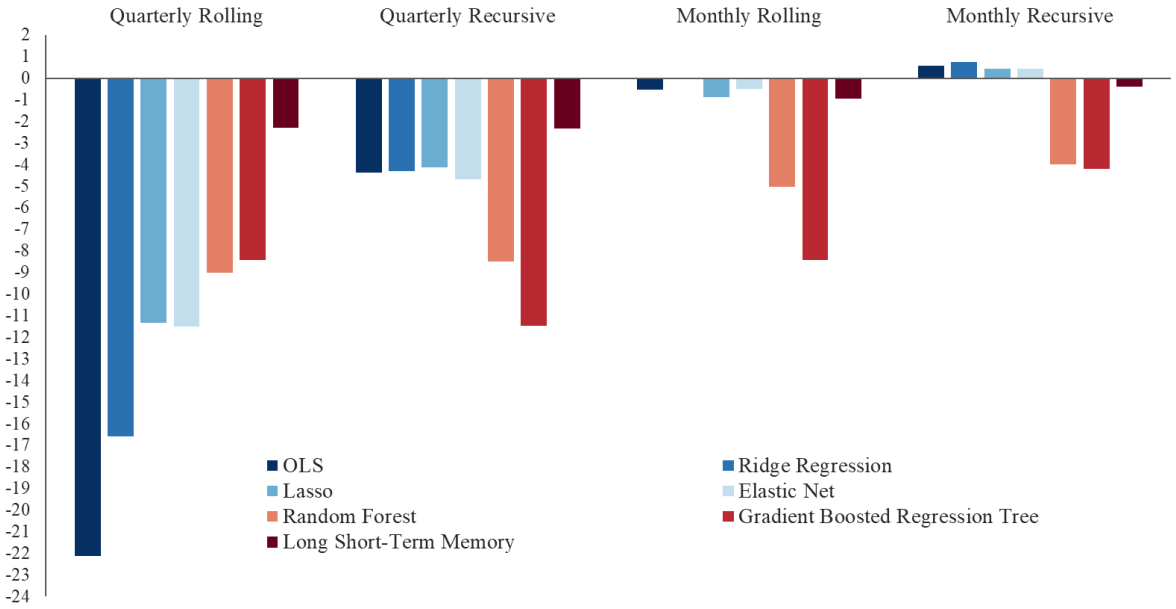

Figure 4.1 Average ΔR^2 by forecasting model, data frequency, and estimation window. The y-axis shows ΔR^2 values in percentage points. For each model, the average ΔR^2 was computed across all sample periods, separately for each data frequency and estimation window type.

The figure does not display statistical significance; it is intended solely to show the average change in ΔR^2 over time. The key takeaways from the figure are as follows:

- Quarterly rolling: all models show negative mean ΔR^2 values, with LSTM performing closest to zero (-2.29%) and tree-based models averaging around -9%. Linear models (OLS and penalized regressions) perform the worst, showing the strongest negative mean ΔR^2 values.

- Quarterly recursive: all regression models show substantial improvement over the rolling window results, while ensemble methods remain similar to the rolling window results (around -8% to -11%). LSTM again shows the smallest average gap (-2.33%), with no notable change from its rolling window performance.

- Monthly rolling: mean ΔR^2 values are clustered near zero for the linear models and LSTM, indicating notable improvement, particularly for the linear models, when shifting from quarterly to monthly data. The ensemble models show larger negative gaps compared to the others, although random forest shows some improvement over its quarterly performance.

- Monthly recursive: all linear models exhibit positive mean ΔR^2 values, though never statistically significant. LSTM shows a slightly negative mean ΔR^2, close to zero, while both ensemble models remain negative but improve compared to their rolling window results.

These averages reflect the detailed period-by-period results in Tables 4.1 to 4.4 and confirm three key findings. First, our forecasting models rarely outperform the AR benchmark. Second, expanding the estimation window (using recursive instead of rolling) improves the performance of linear models and, to a smaller extent, the other models, especially at the quarterly frequency. Third, increasing data frequency from quarterly to monthly significantly enhances predictive accuracy. Taken together, these results show that the best overall performance, though still not surpassing the benchmark, is achieved using monthly data with the recursive estimation window.

4.4 Results of variable importance analysis

Our results presented in the previous section highlight that the linear models and the LSTM overall performed better than the ensemble methods. These models nearly matched the benchmark AR model, specifically for monthly data. Consequently, in the last phase of our study, we conducted a variable importance analysis, namely permutation feature importance, to identify which variables contributed the most to the performance of these models.

To avoid unnecessary and extensive computational costs associated with worst performing models, including random forest and GBRT, we applied the permutation feature analysis only on the LSTM and the elastic net, the latter representing the linear models as it combines ridge and lasso.

Regarding the sample period for the variable importance analysis, we selected the 1972-2019 forecasting period because among eight time periods in our study, it is the only one that contains all available variables. This analysis was carried out on both monthly and quarterly data under rolling and recursive estimation windows. The results of permutation feature importance for quarterly and monthly data are presented below.

4.4.1 Variable importance analysis results of quarterly sampling

Figures 4.2 and 4.3 in the following indicate the average of the permutation importance measure of all lagged variables across all windows generated from the elastic net, under the rolling and recursive windows, respectively. These two figures illustrate the strong persistence of volatility, as the first two lags of volatility (QLVOL_L1 and QLVOL_L2) account for roughly two-thirds of the elastic net model's predictive power. After controlling for this persistence, the most important predictors, under both rolling and recursive windows, are the credit risk measures, the default spread (*dfy*) and the commercial paper-to-Treasury spread (*cp*), followed by economic uncertainty captured in producer price index volatility (*ppivol*). The net payout yield (*npy*) gives the model additional information about future volatility by signalling whether firms are net issuers of equity or returning cash through dividends and share buybacks. The investment-capital ratio (*ik*) indicates that shifts in corporate investment activity also contribute to the model's predictive accuracy.

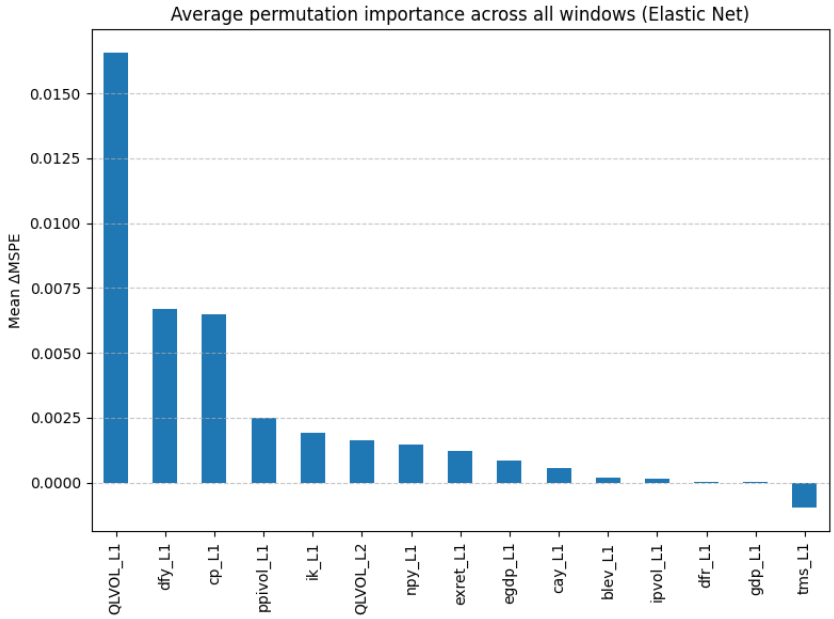

Figure 4.2 Average permutation importance across all rolling windows, quarterly data, elastic net model, 1972Q3-2019Q4. The y-axis values represent the mean increase in MSPE when each variable is randomly permuted, averaged over all windows. Higher values indicate greater importance of the variable in the forecasting task.

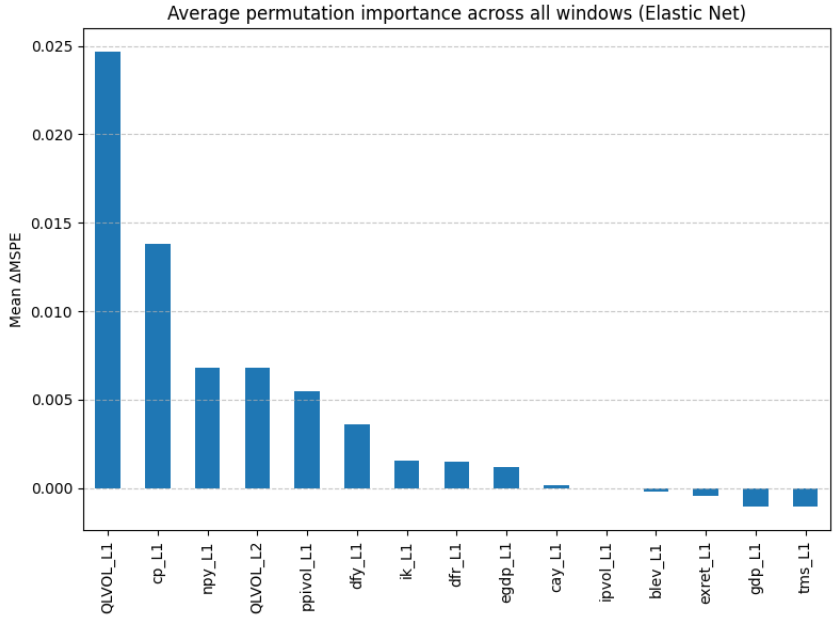

Figure 4.3 Average permutation importance across all recursive windows, quarterly data, elastic net model, 1972Q3-2019Q4. The y-axis values represent the mean increase in MSPE when each variable is randomly permuted, averaged over all windows. Higher values indicate greater importance of the variable in the forecasting task.

Figures 4.4 and 4.5 in the following demonstrate the average of the permutation importance measure across all windows generated from the LSTM, under the rolling and recursive windows, respectively. Both figures confirm that volatility is mainly self-driven, similar to the elastic net results, the first two lags of volatility together account for about two-thirds of the model's predictive power under either estimation window. Beyond these lags, the model relies on credit market indicators, largely the commercial paper-to-Treasury spread (*cp*) and, to a lesser extent, the default return spread (*dfr*). The GDP growth (*gdp*) also shows noticeable importance in both figures, indicating that recent economic activity changes help the LSTM in volatility forecasting. The survey-based expected GDP growth (*egdp*) adds a smaller but still positive contribution, suggesting that growth expectations provide an additional signal about future volatility.

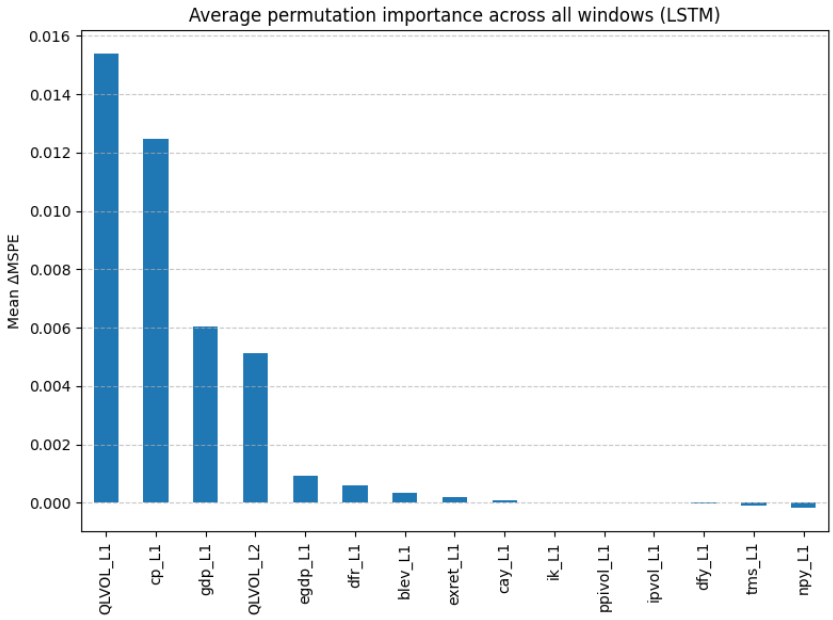

Figure 4.4 Average permutation importance across all rolling windows, quarterly data, LSTM model, 1972Q3-2019Q4. The y-axis values represent the mean increase in MSPE when each variable is randomly permuted, averaged over all windows. Higher values indicate greater importance of the variable in the forecasting task.

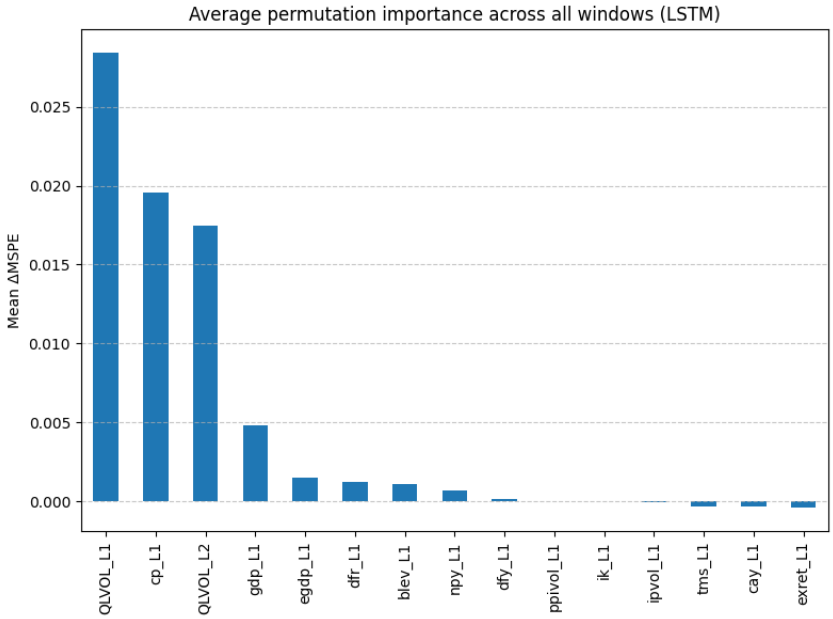

Figure 4.5 Average permutation importance across all recursive windows, quarterly data, LSTM model, 1972Q3-2019Q4. The y-axis values represent the mean increase in MSPE when each variable is randomly permuted, averaged over all windows. Higher values indicate greater importance of the variable in the forecasting task.

In summary, for quarterly data, both models (elastic net and LSTM) indicate that volatility is largely driven by its own past values. After this persistence, credit market variables (*cp*, *dfy*, and *dfr*) are the key macro drivers in both elastic net and LSTM results. The elastic net also uses two balance sheet measures, net payout (*npy*) and the investment-capital ratio (*ik*), whereas the LSTM receives additional signals from economic growth measures, specifically current GDP growth (*gdp*) and, to a lesser extent, expected GDP growth (*egdp*).

4.4.2 Variable importance analysis results of monthly sampling

Figures 4.6 and 4.7 indicate the average of the permutation importance measure across all windows generated from the elastic net, under the rolling and recursive windows, respectively.

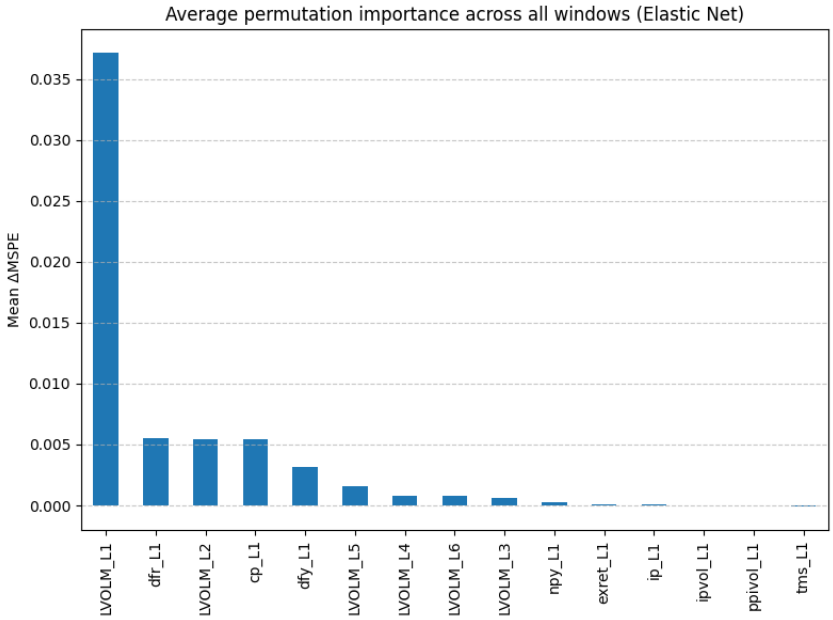

Figure 4.6 Average permutation importance across all rolling windows, monthly data, elastic net model, 1972.3-2019.12. The y-axis values represent the mean increase in MSPE when each variable is randomly permuted, averaged over all windows. Higher values indicate greater importance of the variable in the forecasting task.

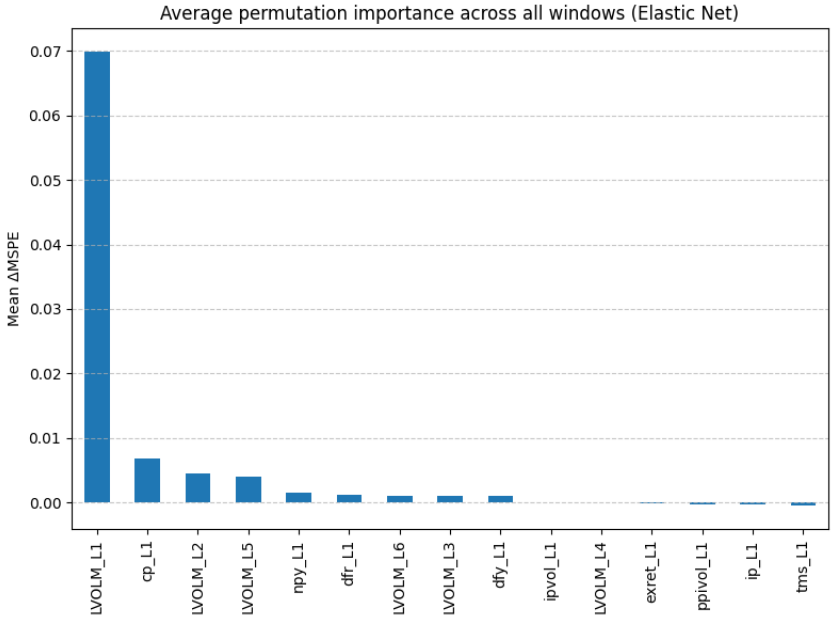

Figure 4.7 Average permutation importance across all recursive windows, monthly data, elastic net model, 1972.3-2019.12. The y-axis values represent the mean increase in MSPE when each variable is randomly permuted, averaged over all windows. Higher values indicate greater importance of the variable in the forecasting task.

As can be seen from Figures 4.6 and 4.7, with monthly data, the elastic net again shows that volatility is largely driven by its own past. In both rolling and recursive settings, the first lag of volatility dominates the permutation order, and the second lag adds a smaller increment. The credit spread variables provide most of the external signals: default return spread (*dfr*) and default spread (*dfy*) in the rolling window, and commercial paper-to-Treasury spread (*cp*) in both windows. Additional volatility lags and all other macro variables contribute progressively less and have permutation scores near zero.

Figures 4.8 and 4.9 indicate the average of the permutation importance measure across all windows generated from the LSTM, under the rolling and recursive windows, respectively.

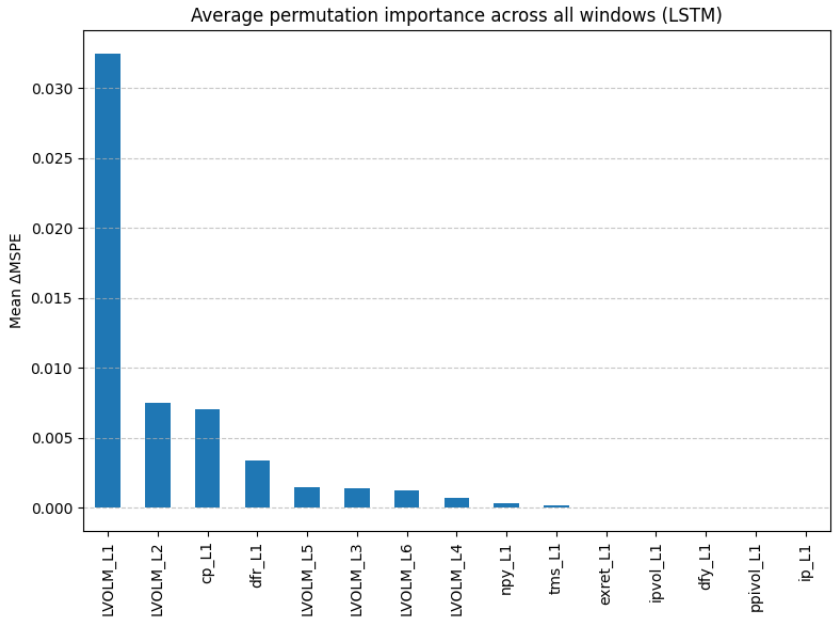

Figure 4.8 Average permutation importance across all rolling windows, monthly data, LSTM model, 1972.3-2019.12. The y-axis values represent the mean increase in MSPE when each variable is randomly permuted, averaged over all windows. Higher values indicate greater importance of the variable in the forecasting task, with the ranking reflecting relative predictive contribution.

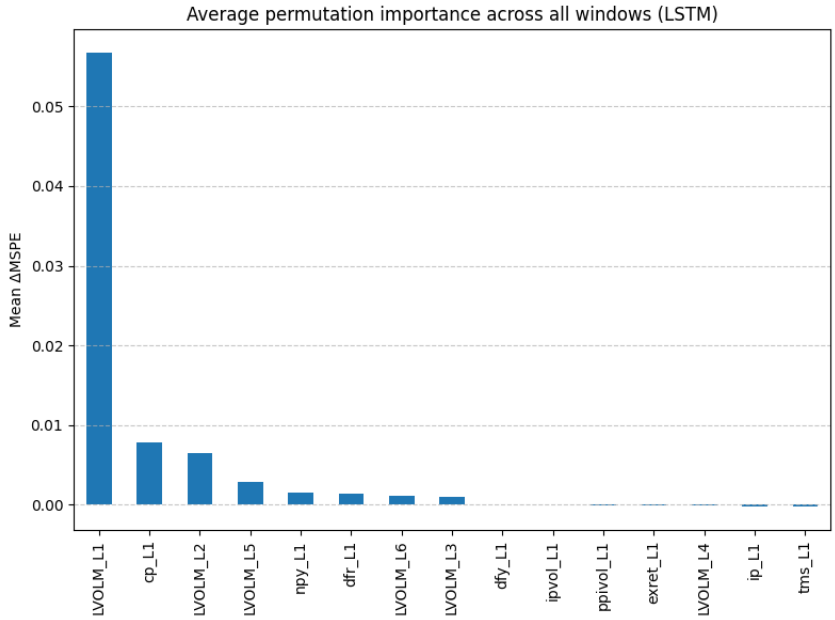

Figure 4.9 Average permutation importance across all recursive windows, monthly data, LSTM model, 1972.3-2019.12. The y-axis values represent the mean increase in MSPE when each variable is randomly permuted, averaged over all windows. Higher values indicate greater importance of the variable in the forecasting task, with the ranking reflecting relative predictive contribution.

Figure 4.8 confirms the dominance of the first and second lag of volatility in the predictive power of LSTM for monthly data under the rolling window estimation approach. The commercial paper-to-Treasury spread (*cp*) and the default return spread (*dfr*) represent the next important variables in this sample. Additional lags of volatility follow *cp* and *dfr* in importance and have a smaller contribution in this setting. In the recursive setting as shown in Figure 4.9, the LSTM depends mainly on last month's volatility. The most important external feature is the commercial paper-to-Treasury spread (*cp*), with the second and fifth volatility lags (LVOLM_L2 and LVOLM_L5) ranking behind it. The net payout yield (*npy*) and the default return spread (*dfr*) provide only small incremental gains.

4.5 Discussion

After presenting our empirical results, this section places the findings in the context of the volatility forecasting literature. Our results provide evidence on how different modeling choices, data frequency, estimation window, time horizon, and the use of macroeconomic

predictors impact forecast performance relative to a univariate autoregressive (AR) benchmark.

Our main observation is that every forecasting model we tested (OLS, ridge, lasso, random forest, GBRT, and LSTM) either equalled or fell short of the AR benchmark in out-of-sample accuracy. "Equal" here means small, statistically insignificant changes in out-of-sample R^2, whether positive or negative. This reinforces earlier evidence that simple AR models are hard to beat in volatility forecasting (Paye, 2012; Andersen et al., 2001; Ghysels et al., 2006; Hansen and Lunde, 2005).

This persistence of the AR model's strength suggests that information embedded in past volatility alone captures much of the predictable variation, so added model complexity does not always lead to better forecasts. This outcome mirrors the conclusions of Branco et al. (2022) and Audrino and Chassot (2022), who found little benefit from advanced machine learning (ML) methods over simpler linear benchmarks.

Turning to economic shocks, our quarterly results show mixed effects. Including the 1970s oil shock, in the sample period of 1972Q3-2010Q4, improved performance of our forecasting models, compared with a subsample that begins in 1982Q3, which excludes this event. As a result, higher volatility periods may provide richer signals that complex models can exploit, consistent with findings of Paye (2012) and with the superior LSTM forecasts reported by Petrozziello et al. (2022) during the 2007-2008 crisis. Nevertheless, when we extend the sample periods to cover the COVID-19 event, forecast accuracy weakens slightly: models perform a bit better when the pandemic years are excluded (ending in 2019). This echoes Rahimikia and Poon (2024), who also found that neural networks underperformed during extreme market stress.

Looking at monthly data, all models improve. CW statistics are mostly positive, and statistically significant and ΔR^2 values move closer to zero, confirming that higher-frequency data contain richer short-term signals, as also documented by Christensen et al. (2023). Several studies that reported ML gains also relied on high-frequency data (daily or intraday), such as Donaldson and Kamstra (1997); Zhu et al. (2023); and Zhang et al. (2024).

In the monthly setting, linear models (OLS and penalized regressions) and the LSTM consistently outperform ensemble trees (random forest and GBRT), especially under recursive estimation. This finding highlights neural networks' ability to capture nonlinear dynamics in volatility (Petrozziello et al. 2022; Rahimikia and Poon 2024).

The switch from rolling to recursive windows further improves forecasting accuracy, mostly pronounced in linear models. This indicates that expanding the estimation window helps the models adapt to changing economic conditions, a conclusion also noted by Audrino and Chassot (2022) in their study of window length impact on model performance.

Concerning macroeconomic predictors, CW tests show that these variables, most of the time, have informational value and Granger-cause the realized volatility, supporting Paye (2012). However, the prediction accuracy test results, measured by changes in out-of-sample R^2 and GW test statistics, imply limited incremental value of these variables in forecasting volatility. This evidence is supported by findings of previous studies, such as Christensen et al. (2023); Filipovic and Khalilzadeh (2021); Nõu et al. (2021); Petrozziello et al. (2022); and Moon and Kim (2019), where ML models using only past price or volatility data performed as well as, or better than, versions that added an extended set of macro and other external predictors.

Permutation feature importance results are consistent with Paye (2012). Across all eight samples on which we conducted this analysis (quarterly vs monthly, rolling vs recursive, and elastic net vs LSTM), the first one to two lags of realized volatility accounted for near two-thirds of the model performance, regardless of frequency, estimation windowing, or algorithm. After this persistence, the variables related to credit risk, including commercial paper-to-Treasury spread (*cp*), default return spread (*dfr*), and default spread (*dfy*), frequently dominated other macroeconomic predictors. Elastic net and LSTM both rely on these spreads, with only minor to modest extra help from PPI volatility (*ppivol*), net payout yield (*npy*), the investment-capital ratio (*ik*), and at the quarterly horizon, real current and expected GDP growth (*gdp* and *egdp*). In short, credit conditions provide the most important external signal for future volatility, confirming Paye's original insight with advanced machine learning evidence.

In summary, we find only modest gains from complex ML techniques with macroeconomic predictors over autoregressive benchmarks. However, some conditions, such as higher-frequency data, turbulent periods, and recursive estimation windowing, enhance forecast quality and allow advanced ML models to add value.

4.6 Implications of findings

Our findings have several practical and academic implications for volatility forecasting. First, the persistent strength of the autoregressive (AR) benchmark shows that simple, parsimonious models still hold a significant predictive power. Therefore, investment professionals, risk managers, and policymakers should think more carefully before adopting ML frameworks to avoid, as much as possible, the unnecessary computational costs and extra time needed for complex settings of these models.

Second, the small incremental value of macroeconomic variables in predictive accuracy of ML models, despite their statistical relevance under the Clark and West (CW) test, suggests that volatility forecasts mostly benefit from market-based information and the recent history of volatility itself (as also confirmed in our variable importance analysis results). Therefore, focusing on direct market signals may be more efficient in practice because such data respond faster to changing conditions, whereas macro indicators and variables are much slower to update.

Third, the improved results observed from higher-frequency data (monthly vs quarterly) and during some of the turbulent periods indicate that the forecasting model effectiveness depends largely on data granularity and the market condition. As a result, practitioners should consider higher frequency data and use estimation approaches, such as recursive window, that can adapt as information accumulates and economic regimes shift.

Finally, because our variable importance analysis shows that credit spread variables dominate all other macro predictors and that higher-frequency (monthly) data perform better, practitioners should monitor real time movements in credit conditions as these predictors offer the greatest incremental value when forecasting stock return volatility.

4.7 Limitations and recommendations for future studies

Although our study compares eight forecasting models (including the benchmark AR) across two data frequencies (quarterly and monthly), two estimation schemes (rolling and recursive), and eight sample periods (four main periods and four extended periods for robustness check), several limitations remain that open avenues for further research.

Our first constraint is data frequency. We use realized volatility aggregated at a monthly and quarterly basis. However, high-frequency sampling, such as weekly, daily, or even intraday, contains many more observations and therefore, richer short-term signals. Moreover, some previous studies showed that realized variance measures converge to the underlying quadratic variation as the sampling interval gets smaller (Andersen et al. 2003; Barndorff Nielsen and Shephard 2002). Future studies can examine whether our models improve when trained on higher-frequency data.

The next limitation of our study is predictor choice. To stay consistent with Paye (2012), we use the same set of macroeconomic variables. Choosing different sets of features that have been previously proven useful in forecasting realized volatility, such as option implied metrics, order book measures, or news and social media sentiment, may reveal whether other types of input features help ML methods outperform a strong linear benchmark such as an AR model.

Model architecture is another constraint of this study. Although we include lagged volatility (up to two lags for quarterly and six lags for monthly sampling) in addition to the macro predictors in our forecasting models, to account for temporal dependency of time series data, our ensemble methods assume observations are independent and identically distributed, which possibly ignores volatility clustering. Block bootstrap resampling or other bagging approaches could preserve time dependency more effectively. While Christensen et al. (2023) noted that they have not observed a difference in results from standard bootstrap and block bootstrap, additional tests in future research would be valuable.

Model validation during hyperparameter tuning is also a constraint, as we rely on a fixed split between training and validation sets. Although in our study, the validation period follows the

training period to respect time order, a rolling or time series cross-validation approach in future studies may produce more robust hyperparameter choices.

Finally, due to the computational cost, we evaluate only one deep learning model, the LSTM, which adds another limitation to this study. Future research could explore other neural networks such as gated recurrent units, temporal convolutional networks, transformers, or attention-based hybrids to see whether they can capture volatility dynamics more effectively.

Chapter 5. Conclusion

The main objective of this thesis was to investigate the effectiveness of machine learning (ML) models, augmented with macroeconomic variables, in forecasting US stock return volatility. Specifically, the central questions addressed were: 1) whether machine learning algorithms can consistently improve one-step-ahead forecasts relative to the autoregressive (AR) benchmark model, and 2) whether the financial and macroeconomic variables introduced by Paye (2012) provide incremental information over the volatility's past values. These questions were motivated by the mixed evidence in existing literature and by the theoretical discussion of macroeconomic variables as forward-looking signals of financial risk.

To answer them, our analysis began by replicating Paye (2012), which provided a baseline for all subsequent extensions we added in our study. Building on that foundation, six machine learning models, including ridge, lasso, elastic net, random forest, gradient boosted regression trees (or GBRT), and long short-term memory (LSTM) networks, were estimated alongside ordinary least squares (OLS) and the AR model. Each model was employed on eight different sample periods to assess the impact of calm and turbulent market environments, and the data were on both a quarterly and a monthly basis. Forecasts were produced under rolling and recursive (or expanding) windows. The predictive accuracy was gauged by changes in out-of-sample R^2, Giacomini-White (GW) test, and Clark and West (CW) test statistics.

At the last step, a variable importance analysis was performed to assess the contribution of each predictor in forecasting performance of our models. Following Breiman (2001), each predictor was randomly permuted and the resulting increase in mean squared prediction error (MSPE) was recorded. This analysis only focused on two best performing algorithms, elastic net and LSTM.

This study provided several important conclusions. First, none of our machine learning predictive models delivered statistically significant gains over an AR(2) at the quarterly horizon and AR(6) at the monthly frequency. However, the results from linear models (OLS and penalized regressions) and the LSTM suggested that these models nearly matched the AR model's predictive accuracy in most cases, as indicated by slightly positive or negative changes in R^2. The tree-based models consistently indicated the worst predictive performance.

Second, data frequency and estimation window length influenced our models' performance. In particular, moving from quarterly to monthly data and from rolling to recursive estimation window improved the out-of-sample R^2 values. Third, the models performed differently across various volatility regimes. Including the 1970s oil-shock interval enhanced the forecasting performance of our models, whereas the inclusion of COVID-19 data slightly decreased their prediction accuracy, suggesting that high-volatility periods can both improve and distort predictive relationships. Fourth, Clark and West (CW) test statistics revealed that the macroeconomic predictors in our models mostly Granger-caused the realized volatility, however, their incremental economic value (as measured by ΔR^2) was not significant. Fifth, the permutation analysis revealed that after accounting for the persistence of volatility, credit market spread variables (commercial paper-to-Treasury spread, default spread, and default return spread) were the most relevant macroeconomic variables among all.

Accordingly, this thesis offers several empirical findings to the literature. First, extending Paye's study through 2019 and 2023 and employing more complex machine learning algorithms reveals that simple AR models remain difficult to outperform in forecasting volatility. Second, when incorporating macroeconomic variables into machine learning models, we find no statistically significant out-of-sample improvement beyond lagged volatility across our samples and horizons. Finally, permutation tests show that credit spreads are the most informative macro variables; however, their incremental predictive power is not statistically significant.

Bibliography

Adrian, T., & Shin, H. S. (2010). Liquidity and leverage. *Journal of Financial Intermediation, 19*(3), 418-437.

Alessandretti, A., ElBahrawy, A., Aiello, L. M., & Baronchelli, A. (2018). Anticipating cryptocurrency prices using machine learning. *Complexity, 2018*, 8983590.

Andersen, T. G., & Bollerslev, T. (1998). Deutsche mark-dollar volatility: Intraday activity patterns, macroeconomic announcements, and longer run dependencies. *Journal of Finance, 53*(1), 219-265.

Andersen, T. G., Bollerslev, T., Diebold, F. X., & Labys, P. (2001). The distribution of realized exchange rate volatility. *Journal of the American Statistical Association, 96*(453), 42-55.

Andersen, T. G., Bollerslev, T., Diebold, F. X., & Labys, P. (2003). Modeling and forecasting realized volatility. *Econometrica, 71*(2), 579-625.

Audrino, F., & Chassot, J. (2022). HARd to beat: The overlooked impact of rolling windows in the era of machine learning. *Working paper*.

Baillie, R. T., & Bollerslev, T. (1989). The message in daily exchange rates: A conditional variance tale. *Journal of Business & Economic Statistics, 7*(4), 297-305.

Bansal, M., Goyal, A., & Choudhary, A. (2022). Stock market prediction with high accuracy using machine learning techniques. *Procedia Computer Science, 215*, 247-265.

Barndorff-Nielsen, O. E., & Shephard, N. (2002). Econometric analysis of realized volatility and its use in estimating stochastic volatility models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology), 64*(2), 253-280.

Bianchi, D., Büchner, M., & Tamoni, A. (2020). Bond risk premiums with machine learning. *Working paper*.

Bianchi, D., Büchner, M., & Tamoni, A. (2021). Bond risk premiums with machine learning. *Review of Financial Studies, 34*(2), 1046-1089.

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

Bollerslev, T., Patton, A. J., & Quaedvlieg, R. (2016). Exploiting the errors: A simple approach for improved volatility forecasting. *Journal of Econometrics, 192*(1), 1-18.

Boudoukh, J., Michaely, R., Richardson, M., & Roberts, M. R. (2007). On the importance of measuring payout yield: Implications for empirical asset pricing. *Journal of Finance, 62*(2), 877-915.

Branco, R. R., Rubesam, A., & Zevallos, M. (2022). Forecasting realized volatility: Does anything beat linear models? *Working paper*.

Breen, W., Glosten, L. R., & Jagannathan, R. (1989). Economic significance of predictable variations in stock index returns. *Journal of Finance, 44*(5), 1177-1189.

Breiman, L. (2001). Random forests. *Machine Learning, 45*(1), 5-32.

Campbell, J. Y. (1987). Stock returns and the term structure. *Journal of Financial Economics, 18*(2), 373-399.

Campbell, J. Y., & Thompson, S. B. (2008). Predicting excess stock returns out of sample: Can anything beat the historical average? *Review of Financial Studies, 21*(4), 1509-1531.

Campbell, S. D., & Diebold, F. X. (2009). Stock returns and expected business conditions: Half a century of direct evidence. *Journal of Business & Economic Statistics, 27*(2), 266-278.

Carr, P., Wu, L., & Zhang, Z. (2019). Using machine learning to predict realized variance. *Working paper* (arXiv:1909.10035).

Caruana, R., & Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd International Conference on Machine Learning* (pp. 161-168). ACM.

Christensen, K., Siggaard, M. V., & Veliyev, B. (2023). A machine learning approach to volatility forecasting. *Journal of Financial Econometrics, 21*(5), 1680-1727.

Clark, T. E., & West, K. D. (2007). Approximately normal tests for equal predictive accuracy in nested models. *Journal of Econometrics, 138*(1), 291-311.

Cochrane, J. H. (1991). Production-based asset pricing and the link between stock returns and investment. *Journal of Finance, 46*(1), 207-234.

Corsi, F. (2009). A simple approximate long-memory model of realized volatility. *Journal of Financial Econometrics, 7*(2), 174-196.

Corsi, F., & Renò, R. (2012). Discrete-time volatility forecasting with persistent leverage effect and the link with continuous-time volatility modeling. *Journal of Business & Economic Statistics, 30*(3), 368-380.

Ding, Z., Granger, C. W. J., & Engle, R. F. (1993). A long memory property of stock market returns and a new model. *Journal of Empirical Finance, 1*(1), 83-106.

Donaldson, R. G., & Kamstra, M. (1997). An artificial neural network-GARCH model for international stock return volatility. *Journal of Empirical Finance, 4*(1), 17-46.

Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation. *Econometrica, 50*(4), 987-1007.

Engle, R. F. (1983). Estimates of the variance of US inflation based upon the ARCH model. *Journal of Money, Credit and Banking, 15*(3), 286-301.

Engle, R. F., & Kraft, D. F. (1983). Multiperiod forecast error variances of inflation estimated from ARCH models. In A. Zellner (Ed.), *Applied time series analysis of economic data* (pp. 293-302). Bureau of the Census.

Engle, R. F., & Rangel, J. G. (2008). The Spline-GARCH model for low-frequency volatility and its global macroeconomic causes. *Review of Financial Studies, 21*(3), 1187-1222.

Engle, R. F., Ghysels, E., & Sohn, B. (2008). On the economic sources of stock market volatility (Working paper).

Engle, R. F., Lilien, D. M., & Robins, R. P. (1987). Estimating time varying risk premia in the term structure: The ARCH-M model. *Econometrica, 55*(2), 391-407.

Filipovic, D., & Khalilzadeh, A. (2021). Machine learning for predicting stock return volatility. Swiss Finance Institute Research Paper No. 21-95.

French, K. R., Schwert, G. W., & Stambaugh, R. F. (1987). Expected stock returns and volatility. *Journal of Financial Economics, 19*(1), 3-29.

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

Fu, W. J. (1998). Penalized regressions: The bridge versus the lasso. *Journal of Computational and Graphical Statistics, 7*(3), 397-416.

Ghysels, A., Santa-Clara, P., & Valkanov, R. (2006). Predicting volatility: Getting the most out of return data sampled at different frequencies. *Journal of Econometrics, 131*(1-2), 59-95.

Giacomini, R., & White, H. (2006). Tests of conditional predictive ability. *Econometrica, 74*(6), 1545-1578.

Glosten, L. R., Jagannathan, R., & Runkle, D. E. (1993). On the relation between the expected value and the volatility of the nominal excess return on stocks. *Journal of Finance, 48*(5), 1779-1801.

Goyal, A., & Welch, I. (2008). A comprehensive look at the empirical performance of equity premium prediction. *Review of Financial Studies, 21*(4), 1455-1508.

Graham, J. R., & Harvey, C. R. (2001). Expectations of equity risk premia, volatility and asymmetry from a corporate finance perspective (NBER Working Paper No. 8678). National Bureau of Economic Research.

Gu, S., Kelly, B., & Xiu, D. (2019). Empirical asset pricing via machine learning. *Working paper*.

Gu, S., Kelly, B., & Xiu, D. (2020). Empirical asset pricing via machine learning. *Review of Financial Studies, 33*(5), 2223-2273.

Hansen, P. R., & Lunde, A. (2005). A forecast comparison of volatility models: Does anything beat a GARCH(1,1)? *Journal of Applied Econometrics, 20*(7), 873-889.

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

Hoerl, A. E., & Kennard, R. W. (1970a). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics, 12*(1), 55-67.

Hoerl, A. E., & Kennard, R. W. (1970b). Ridge regression: Applications to nonorthogonal problems. *Technometrics, 12*(1), 69-82.

Hong, H., & Stein, J. C. (1999). A unified theory of underreaction, momentum trading, and overreaction in asset markets. *Journal of Finance, 54*(6), 2143-2184.

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

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436-444.

Lettau, M., & Ludvigson, S. C. (2001). Consumption, aggregate wealth, and expected stock returns. *Journal of Finance, 56*(3), 815-849.

Lettau, M., & Ludvigson, S. C. (2010). Measuring and modeling variation in the risk-return trade-off. In Y. Aït-Sahalia & L. P. Hansen (Eds.), *Handbook of Financial Econometrics* (Vol. 1, pp. 617-690). Elsevier.

Ludvigson, S. C., & Ng, S. (2007). The empirical risk-return relation: A factor analysis approach. *Journal of Financial Economics, 83*(1), 171-222.

Luong, C., & Dokuchaev, N. (2018). Forecasting of realised volatility with the random forests algorithm. *Journal of Risk and Financial Management, 11*(4), 61.

MacKinnon, J. G. (1994). Approximate asymptotic distribution functions for unit-root and cointegration tests. *Journal of Business & Economic Statistics, 12*(2), 167-176.

Maglaras, L. A., Das, S., Tripathy, N., & Patnaik, S. (Eds.). (2024). *Machine learning approaches in financial analytics*. Springer.

Marquering, W., & Verbeek, M. (2004). The economic value of predicting stock index returns and volatility. *Journal of Financial and Quantitative Analysis, 39*(2), 407-429.

McNally, P., Roche, J., & Caton, S. (2018). Predicting the price of Bitcoin using machine learning. In *Proceedings of the 26th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP 2018)* (pp. 339-343).

Mittnik, S., Robinzonov, N., & Spindler, M. (2015). Stock market volatility: Identifying major drivers and the nature of their impact. *Journal of Banking & Finance, 58*, 1-14.

Moon, K.-S., & Kim, H. (2019). Performance of deep learning in prediction of stock market volatility. *Economic Computation and Economic Cybernetics Studies and Research, 53*(2), 77-92.

Nelson, D. B. (1991). Conditional heteroskedasticity in asset returns: A new approach. *Econometrica, 59*(2), 347-370.

Niu, Z., Demirer, R., Suleman, M. T., Zhang, H., & Zhu, X. (2023). Forecasting realized volatility with machine learning: Panel data perspective. *Journal of Empirical Finance, 73*, 251-271.

Niu, Z., Demirer, R., Suleman, M. T., Zhang, H., & Zhu, X. (2024). Do industries predict stock market volatility? Evidence from machine learning models. *Journal of International Financial Markets, Institutions and Money, 90*, 101903.

Nõu, A., Lapitskaya, D., Eratalay, M. H., & Sharma, R. (2021). Predicting stock return and volatility with machine learning and econometric models: A comparative case study of the Baltic stock market. *Working paper*.

Nybo, C. (2021). Sector volatility prediction performance using GARCH models and artificial neural networks. *Working paper*.

Patton, A. J., & Sheppard, K. (2015). Good volatility, bad volatility: Signed jumps and the persistence of volatility. *Review of Economics and Statistics, 97*(3), 683-697.

Paye, B. S. (2012). Déjà vol': Predictive regressions for aggregate stock market volatility using macroeconomic variables. *Journal of Financial Economics, 106*(3), 527-546.

Petrozziello, A., Troiano, L., Serra, A., Jordanov, I., Storti, G., Tagliaferri, R., & La Rocca, M. (2022). Deep learning for volatility forecasting in asset management. *Soft Computing, 26*, 8553-8574.

Phillips, P. C. B., & Perron, P. (1988). Testing for a unit root in time series regression. *Biometrika, 75*(2), 335-346.

Rahimikia, E., & Poon, S.-H. (2024). Machine learning for realised volatility forecasting. *SSRN Electronic Journal*. *Working paper*.

Rapach, D. E., Strauss, J. K., & Zhou, G. (2010). Out-of-sample equity premium prediction: Combination forecasts and links to the business cycle. *Journal of Financial Economics, 97*(2), 202-221.

Rossi, A. G. (2018). Predicting stock market returns with machine learning. *Working paper*.

Schwert, G. W. (1989). Why does stock market volatility change over time? *Journal of Finance, 44*(5), 1115-1153.

Shanken, J. (1990). Intertemporal asset pricing: An empirical investigation. *Journal of Econometrics, 45*(1-2), 99-120.

Stambaugh, R. F. (1999). Predictive regressions. *Journal of Financial Economics, 54*(3), 375-421.

Strobl, C., Boulesteix, A.-L., Zeileis, A., & Hothorn, T. (2007). Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC Bioinformatics*, 8, 25.

Taylor, S. J. (1986). *Modelling financial time series*. John Wiley & Sons.

Taylor, S. J., & Xu, X. (1997). The incremental volatility information in one million foreign exchange quotations. *Journal of Empirical Finance, 4*(4), 317-340.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological), 58*(1), 267-288.

Whitelaw, R. F. (1994). Time variations and covariations in the expectation and volatility of stock market returns. *Journal of Finance, 49*(2), 515-541.

Wu, W., Chen, J., Yang, Z., & Tindall, M. L. (2021). A cross-sectional machine learning approach for hedge fund return prediction and selection. *Management Science, 67*(7), 4577-4601.

Zhang, C., Zhang, Y., Cucuringu, M., & Qian, Z. (2024). Volatility forecasting with machine learning and intraday commonality. *Journal of Financial Econometrics, 22*(2), 492-530.

Zhou, Z.-H. (2012). *Ensemble methods: Foundations and algorithms*. Chapman & Hall/CRC.

Zhu, H., Bai, L., He, L., & Liu, Z. (2023). Forecasting realized volatility with machine learning: Panel data perspective. *Journal of Empirical Finance, 73*, 251-271.

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology), 67*(2), 301-320.

Appendix A. Paye (2012) Replication Results

The tables in this appendix present the results of the replication process from Paye (2012).

Symbol	Name	Mean	Satndard Deviation	Skewness	Kurtosis	ρ_1	ρ_2	Philips and Perron test	
								Z_t	p-value
Panel A: Quarterly sampling frequency									
blev	Changes in bank leverage	0.0072	0.1344	-0.68	4.88	-0.19	0.13	-18.95	0.00
		-0.0017	0.0948	-0.24	0.83	-0.29	0.21	-3.95	0.00
cay	Consumption–wealth ratio	0.0001	0.0193	0.09	2.52	0.92	0.86	-2.75	0.07
		0.0026	0.0186	-0.63	-0.06	0.91	0.88	-2.46	0.12
cp	CP-to-Treasury spread	0.6461	0.4920	2.25	10.55	0.60	0.45	-8.04	0.00
		0.6133	0.4396	2.17	8.60	0.73	0.52	-4.54	0.00
dfr	Default return	-0.0002	0.0995	0.03	15.80	-0.02	0.06	-16.99	0.00
		0.0002	0.0076	0.51	13.95	-0.07	0.07	-8.36	0.00
dfy	Default yield	0.0158	0.0072	1.41	7.02	0.85	0.73	-4.23	0.00
		0.0097	0.0045	1.72	3.82	0.91	0.78	-4.17	0.00
egdp	Expected GDP growth	2.5364	1.4331	-0.66	5.37	0.86	0.72	-3.87	0.00
		2.5466	1.4231	-0.67	2.52	0.87	0.73	-3.41	0.01
exret	Expected return	0.0199	0.0201	0.90	3.78	0.78	0.67	-5.40	0.00
		0.0199	0.0197	0.97	1.19	0.46	0.45	-4.84	0.00
gdp	GDP growth	3.0439	3.8156	-0.38	4.34	0.37	0.19	-9.99	0.00
		3.2289	3.8165	-0.20	1.16	0.35	0.20	-7.89	0.00
ik	Investment-capital ratio	0.0358	0.0036	0.27	2.43	0.96	0.89	-2.57	0.10
		0.0362	0.0032	0.24	-0.53	0.97	0.90	-3.88	0.00
ipvol	Industrial production volatility	0.0045	0.0046	2.25	8.86	0.26	0.11	-12.71	0.00
		0.0000	0.0001	4.05	17.37	0.19	0.10	-7.05	0.00
npy	Net payout yield	-2.1916	0.2064	-1.63	7.23	0.94	0.87	-2.59	0.10
		-2.1947	0.2073	-1.63	3.76	0.97	0.90	-1.86	0.35
ppivol	Inflation volatility	0.0036	0.0046	4.36	33.26	0.42	0.28	-10.39	0.00
		0.0000	0.0001	12.01	162.93	0.20	0.14	-8.67	0.00
tms	Term spread	0.0160	0.0143	-0.11	3.00	0.83	0.69	-4.61	0.00
		0.0162	0.0139	0.07	-0.60	0.90	0.77	-3.90	0.00
Panel B: Monthly sampling frequency									
cp	CP-to-Treasury spread	0.6147	0.4646	2.42	13.61	0.86	0.74	-7.40	0.00
		0.6160	0.4658	2.41	10.56	0.86	0.74	-4.33	0.00
dfr	Default return	-0.0091	0.2236	1.64	37.78	-0.12	-0.03	-30.61	0.00
		0.0002	0.0135	-0.27	7.70	-0.07	-0.06	-9.05	0.00
dfy	Default yield	0.0157	0.0072	1.30	6.10	0.93	0.87	-4.92	0.00
		0.0097	0.0046	1.79	4.30	0.97	0.93	-3.20	0.02
exret	Expected return	0.0052	0.0040	1.19	5.87	0.89	0.82	-6.52	0.00
		0.0028	0.0036	2.31	9.31	0.52	0.48	-4.96	0.00
ip	Growth in industrial production	0.0024	0.0095	0.27	9.43	0.39	0.24	-18.35	0.00
		0.0024	0.0095	0.21	6.58	0.39	0.24	-8.30	0.00
ipvol	Industrial production volatility	0.0062	0.0064	3.31	23.01	0.24	0.14	-23.20	0.00
		0.0001	0.0003	11.31	164.57	0.12	0.05	-13.48	0.00
npy	Net payout yield	2.1905	0.2069	-1.67	7.34	0.98	0.96	-2.75	0.07
		-2.1941	0.2090	-1.59	3.82	0.98	0.97	-2.08	0.25
ppivol	Inflation volatility	0.0044	0.0056	3.79	24.67	0.37	0.39	-24.17	0.00
		0.0001	0.0002	9.63	112.08	0.35	0.25	-5.30	0.00
tms	Term spread	0.0162	0.0142	-0.05	2.85	0.95	0.90	-4.19	0.00
		0.0162	0.0143	-0.05	-0.16	0.96	0.90	-4.24	0.00

Table A.1 Descriptive statistics. This table replicates Table 1 in Paye (2012). Values in black are those reported in Paye (2012), and values in green are the replication results.

		1927Q2–2010Q4		1952Q2–2010Q4		1927Q2–1951Q4		1952Q2–1985Q4		1986Q1–2010Q4	
Symbol	Name	β	ΔR^2	β	ΔR^2	β	ΔR^2	β	ΔR^2	β	ΔR^2
blev	Changes in bank leverage	-	-	-0.03	0.09	-	-	-0.14**	1.98	0.08	0.60
		-	-	-0.05	0.29	-	-	-0.11	1.16	0.05	0.18
cay	Consumption-wealth ratio	-	-	-0.05	0.26	-	-	-0.09	0.75	-0.09	0.77
		-	-	0.07	0.43	-	-	0.01	0.01	0.04	0.13
cp	CP-tp-Treasury spread	0.12***	1.36	0.11**	1.11	0.23**	2.77	0.31***	7.78	0.07	0.45
		0.12***	1.30	0.10**	0.99	0.25***	2.84	0.29***	7.00	0.08	0.55
dfr	Default return	-0.08**	0.58	-0.13***	1.54	0.04	0.20	-0.08*	1.45	-0.14**	1.59
		-0.05	0.28	-0.14***	1.89	0.14**	1.99	-0.14*	1.95	-0.15*	1.83
dfy	Default yield	0.17***	1.22	0.07	0.26	0.33***	3.46	0.11	0.79	0.04	0.08
		0.14***	1.00	0.04	0.12	0.23**	1.83	0.10	0.73	0.03	0.05
egdp	Expected GDP growth	-	-	-0.02	0.06	-	-	0.00	0.00	-0.03	0.10
		-	-	-0.06	0.38	-	-	-0.04	0.19	-0.07	0.47
exret	Expected return	0.00	0.00	-0.08**	0.56	0.00	0.00	-0.08	0.57	-0.13**	1.65
		0.00	0.00	-0.03	0.11	0.04	0.15	-0.03	0.06	-0.06	0.37
gdp	GDP growth	-	-	0.00	0.00	-	-	0.00	0.00	-0.02	0.04
		-	-	-0.01	0.02	-	-	-0.01	0.02	-0.05	0.19
ik	Investment-capital ratio	-	-	-0.12***	1.42	-	-	0.12**	1.48	0.15**	2.27
		-	-	0.11**	1.24	-	-	0.07	0.49	0.16**	2.46
ipvol	Industrial production volatility	0.01	0.01	-0.02	0.05	-0.03	0.08	-0.04	0.14	0.09	0.76
		0.02	0.03	-0.02	0.05	-0.01	0.01	-0.08	0.59	0.18**	2.77
npy	Net payout	-0.03	0.12	-0.11***	1.08	-0.06	0.34	0.00	0.00	-0.08	0.68
		-0.04	0.12	-0.09*	0.75	-0.08	0.58	0.03	0.07	-0.07	0.44
ppivol	Inflation volatility	0.04	0.12	0.09	0.64	-0.03	0.09	0.18***	2.87	0.02	0.03
		0.01	0.02	0.06	0.34	-0.07	0.45	0.10	1.03	0.03	0.10
tms	Term spread	-0.03	0.08	-0.01	0.02	-0.05	0.21	-0.06	0.31	-0.08***	0.56
		-0.02	0.06	-0.02	0.04	-0.03	0.10	-0.07	0.52	-0.07	0.49
		F	ΔR^2	F	ΔR^2	F	ΔR^2	F	ΔR^2	F	ΔR^2
sink	Kitchen sink	5.29***	3.32	4.21***	6.76	2.34**	5.35	8.16***	11.56	7.63***	7.00
		50.63***	3.16	17.21***	7.48	21.04***	7.96	5.91***	12.11	8.85***	9.88

Table A.2 In-sample regressions for quarterly data. This table replicates Table 2 in Paye (2012). Values in black are those reported in Paye (2012), and values in green are the replication results.

		1927.2–2010.12		1952.2–2010.12		1927.2–1951.12		1952.1–1985.12		1986.1–2010.12	
Symbol	Name	β	ΔR^2	β	ΔR^2	β	ΔR^2	β	ΔR^2	β	ΔR^2
cp	CP-tp-Treasury spread	0.07***	0.47	0.06**	0.29	0.17***	1.40	0.16***	2.01	0.03	0.05
		0.08***	0.54	0.06***	0.39	0.19***	1.63	0.16***	2.14	0.04	0.11
dfr	Default return	-0.05***	0.20	-0.06**	0.35	-0.02	0.06	-0.01	0.00	-0.11***	1.22
		-0.07***	0.44	-0.09***	0.78	-0.03	0.10	-0.06*	0.31	-0.14***	1.58
dfy	Default yield	0.10***	0.40	0.04	0.07	0.20***	1.27	0.03	0.05	0.09	0.32
		0.08***	0.32	0.02	0.03	0.18***	1.14	0.04	0.10	0.05	0.11
exret	Expected return	-0.01	0.01	-0.04*	0.17	-0.01	0.01	-0.06**	0.29	-0.02	0.04
		-0.03*	0.10	-0.04	0.14	-0.06	0.32	-0.04	0.18	-0.01	0.01
ip	Growth in industrial production	-0.01	0.02	-0.01	0.01	-0.03	0.10	0.01	0.01	-0.08	0.53
		-0.03*	0.11	-0.02	0.05	-0.06*	0.35	-0.01	0.02	-0.06*	0.34
ipvol	Industrial production volatility	0.00	0.00	0.00	0.00	-0.02	0.04	-0.03	0.07	0.10	0.96
		-0.01	0.01	0.01	0.01	-0.03	0.09	-0.01	0.01	0.11***	1.05
npy	Net payout	-0.02	0.03	-0.06**	0.29	-0.03	0.09	0.00	0.00	-0.05	0.22
		-0.02	0.04	-0.04*	0.18	-0.04	0.16	0.00	0.00	-0.03	0.12
ppivol	Inflation volatility	0.03	0.11	0.05*	0.26	0.01	0.02	0.07**	0.47	0.04	0.15
		0.00	0.00	0.05**	0.26	-0.03	0.08	0.02	0.03	0.08**	0.56
tms	Term spread	-0.02	0.02	-0.01	0.01	-0.01	0.01	-0.02	0.05	-0.05	0.27
		-0.02	0.04	-0.02	0.05	-0.01	0.01	-0.04	0.19	-0.05	0.23
		F	ΔR^2	F	ΔR^2	F	ΔR^2	F	ΔR^2	F	ΔR^2
sink	Kitchen sink	4.06***	1.22	6.29***	1.61	2.97***	2.26	4.12***	2.67	2.08**	3.07
		144.03***	1.52	84.71***	1.84	38.71***	3.95	34.69***	2.45	37.36***	3.64

Table A.3 In-sample regressions for monthly data. This table replicates Table 3 in Paye (2012). Values in black are those reported in Paye (2012), and values in green are the replication results.

		1947Q3–2010Q4, N=254		1972Q3–2010Q4, N=154		1982Q3–2010Q4, N=114		1972Q3–2000Q4, N=114	
Symbol	Name	*CW*	ΔR^2	*CW*	ΔR^2	*CW*	ΔR^2	*CW*	ΔR^2
blev	Changes in bank leverage	-	-	2.43*	0.13	-1.39	-1.52*	2.55	-0.01
		-	-	0.42	-0.46	-1.85	-1.27**	1.66	0.13
cay	Consumption-wealth ratio	-	-	3.09	-0.70	-0.95	-1.22	4.72**	1.96
		-	-	-3.22	-2.17**	-2.04	-1.26**	-3.50	-2.72**
cp	CP-tp-Treasury spread	7.77**	-0.71	15.29**	1.99	1.99	-2.45	21.86***	4.37
		10.16***	0.60	17.25***	2.55	5.53*	-0.44	22.85***	4.46
dfr	Default return	2.93**	0.37	3.98*	0.32	3.97	0.05	2.63	0.14
		2.91*	-0.17	4.88*	0.52	3.89	-0.46	3.08*	0.43
dfy	Default yield	-0.26	-1.21	-0.95	-1.54	0.25	-0.24	-0.27	-0.45
		-0.77	-1.94	-1.06	-2.47	-1.93	-3.25	0.17	-0.73
egdp	Expected GDP growth	-	-	-1.68	-1.30	1.03	0.36	-1.34	-1.29
		-	-	-0.46	-1.22	-0.16	-0.54	-0.30	-1.57
exret	Expected return	-0.33	-1.22	3.32*	-0.15	1.78	-0.23	2.27	0.62
		-0.50	-1.14	-0.66	-1.14	-0.37	-1.10	-2.89	-2.76**
gdp	GDP growth	-	-	-0.51	-0.64	-0.86	-0.73	0.34	-0.26
		-	-	0.68	-0.23	0.08	-0.41	1.38	-0.01
ik	Investment-capital ratio	-	-	6.07**	1.30	5.69*	0.74	6.22**	2.02
		-	-	6.03**	1.01	7.45**	1.04	4.93**	1.35
ipvol	Industrial production volatility	-1.25	-0.83	-2.04	-1.41	-1.43	-0.90	-0.59	-0.51
		0.51	-0.21	1.00	-0.25	0.67	-0.44	0.13	-0.25
npy	Net payout	-1.59	-1.54*	-1.91	-2.12	-0.15	-0.79	-1.82	-2.12
		-2.32	-2.10**	-2.61	-2.28*	-2.38	-2.21	-3.23	-3.08**
ppivol	Inflation volatility	4.65**	0.16	6.83*	-0.06	-2.79	-2.97***	10.32**	0.88
		-1.77	-4.96	-2.80	-8.22	-5.54	-9.91	2.31	-0.57
tms	Term spread	2.64	-0.66	1.43	-1.22	-1.77	-1.46	3.05	-0.46
		2.49	-0.61	1.72	-0.92	0.16	-1.18	2.58	-0.40
sink	Kitchen sink	5.71	-7.89**	5.58	-15.79**	-2.34	-6.90**	19.69**	-7.94
		6.62	-17.29**	21.70*	-21.79**	-2.92	-22.97*	35.99**	-11.48
Combined forecasts									
Mean		1.82**	0.62	2.72***	1.25*	0.41	0.10	3.84***	2.16**
		1.29*	0.34	1.63*	0.65	0.27	-0.06	2.24**	1.23*
Median		0.77	0.26	0.98**	0.48	0.04	-0.03	1.31***	0.75**
		0.82*	0.36	0.21	0.07	-0.49	-0.30	0.40	0.19
Trim-mean		1.17*	0.41	1.75**	0.81	0.12	-0.02	2.46***	1.39**
		0.76	0.23	0.92	0.38	0.03	-0.08	1.11*	0.58
MSPE		2.05**	0.73	3.66***	1.69*	0.91	0.34	5.08***	2.84**
		0.88	-0.19	2.26**	0.72	0.38	0.59	3.10**	1.38

Table A.4 Out-of-sample forecasting results for quarterly data and rolling estimation. This table replicates Table 4 in Paye (2012). Values in black are those reported in Paye (2012), and values in green are the replication results.

		1947Q3–2010Q4, N=254		1972Q3–2010Q4, N=154		1982Q3–2010Q4, N=114		1972Q3–2000Q4, N=114	
Symbol	Name	*CW*	ΔR^2	*CW*	ΔR^2	*CW*	ΔR^2	*CW*	ΔR^2
blev	Changes in bank leverage	-	-	1.03	-0.66	-2.62	-1.90**	2.58	-0.30
		-	-	0.67	-0.14	-1.68	-1.03**	0.79	-0.24
cay	Consumption-wealth ratio	-	-	0.66	-0.06	-0.69	-0.60	0.38	-0.32
		-	-	-0.51	-0.53	0.53	0.12	-1.15	-1.13
cp	CP-tp-Treasury spread	5.90**	0.13	11.35**	0.60	-0.01	-2.39	16.87**	2.56
		5.86**	0.11	10.41**	0.75	-0.56	-4.27**	15.97**	3.18
dfr	Default return	1.92*	0.48	6.77**	1.21	6.81*	0.76	3.28*	0.38
		-0.37	-0.77	8.60**	1.70	9.89*	1.00	3.34*	0.72
dfy	Default yield	6.02**	0.46	0.60	0.04	0.71	0.04	0.29	-0.10
		5.19***	0.84	0.20	-0.22	0.32	-0.38	0.28	-0.29
egdp	Expected GDP growth	-	-	-0.26	-0.14	0.15	0.06	-0.32	-0.20
		-	-	0.49	0.13	2.70*	0.89	0.37	0.05
exret	Expected return	-0.17	-0.20	2.01**	0.49	1.69	0.26	1.22	0.36
		-0.19	-0.29	-0.69	-0.45	-1.23	-1.04*	-1.15	-0.84*
gdp	GDP growth	-	-	-0.55	-0.30	-0.27	-0.12	-0.56	-0.39
		-	-	-0.53	-0.29**	0.46	-0.02	-0.70	-0.47**
ik	Investment-capital ratio	-	-	5.82***	1.56	5.13**	0.95	7.50***	3.11**
		-	-	4.19**	1.30	4.56*	0.27	3.99**	1.75*
ipvol	Industrial production volatility	-0.47	-0.21***	-0.49	-0.32	-0.70	-0.40	0.62	0.27
		-0.02	-0.01	-0.92	-0.97	-1.73	-0.89	1.68*	0.41
npy	Net payout	-0.04	-0.16	3.19	0.22	5.45*	0.94	2.56	0.30
		-0.03	-0.19	1.60	0.06	1.81	0.06	0.00	-0.69
ppivol	Inflation volatility	0.42	-0.01	7.39**	-0.50	0.10	-2.15*	8.84*	-0.05
		-0.42	-0.25**	5.14*	-7.72	2.60	-9.02	2.36	-0.29
tms	Term spread	-0.70	-0.41*	0.43	-0.71	-2.33	-1.20*	1.26	-0.75
		-0.71	-0.41*	0.10	-0.59	-1.94	-1.43	0.81	-0.49
sink	Kitchen sink	11.91***	0.77	21.08***	1.25	12.59***	0.59	23.06**	0.90
		9.15***	-1.14	30.20***	-6.58	15.35**	-6.73	32.11***	-3.16
Combined forecasts									
Mean		1.61***	0.59**	2.92***	1.20**	1.03**	0.36	3.42***	1.72*
		1.12**	0.43**	2.21***	0.95***	1.21*	0.41	2.04***	1.12**
Median		-0.03	-0.03	0.66*	0.27	0.10	0.01	0.94**	0.48*
		0.03	0.00	0.50**	0.22*	0.09	0.01	0.48*	0.26
Trim-mean		0.66**	0.24*	1.79***	0.74*	0.44	0.13	2.21***	1.12*
		0.45*	0.16	1.20***	0.54***	0.50	0.17	1.11**	0.61**
MSPE		1.86***	0.69***	3.77***	1.53**	1.56**	0.57*	4.55***	2.27*
		0.63	-0.03	2.68***	0.95**	1.34*	1.02***	2.68***	1.18*

Table A.5 Out-of-sample forecasting results for quarterly data and recursive estimation. This table replicates Table 5 in Paye (2012). Values in black are those reported in Paye (2012), and values in green are the replication results.

Appendix B. Data Sources

This appendix provides the sources of data used for the construction of our variables.

Changes in bank leverage (*blev*): (1952-2023)

To calculate this variable, the relevant quarterly data series, FL664090005.Q (total financial assets) and FL664190005.Q (total liabilities), were obtained from Table F.130 on the website of the board of governors of the federal reserve system[5]. Total equity was computed as the difference between total financial assets and total liabilities. The leverage ratio was then calculated as total assets divided by total equity, and finally, to get this variable, the percentage changes in the leverage ratio was calculated.

Commercial paper-to-Treasury spread (*cp*): (1927-2023)

To construct *cp*, defined as the spread between the 3-month commercial paper rate and the 3-month T-bill rate, several data series from federal reserve economic data (FRED)[6] were combined due to the lack of unified complete historical data:

1. Commercial Paper Rates:

- 1927-01 to 1971-03: commercial paper rates for New York, NY (M13002US35620M156NNBR).
- 1971-04 to 1997-08: 3-month prime commercial paper, average dollar offering rate, discount basis.
- 1997-09 to 2023-12: 90-day AA nonfinancial commercial paper rate.

2. Treasury Bill Rates:

- 1927-01 to 1933-12: yields on short-term U.S securities, including three-month Treasury notes and bills.

[5] https://www.federalreserve.gov/

[6] https://fred.stlouisfed.org/

- 1934-01 to 2023-12: 3-month Treasury bill secondary market rate, discount basis.

The *cp* variable was constructed at both monthly and quarterly frequencies. The monthly series, derived directly from the sources, represent average daily rates. For the quarterly series, we simply took the average of monthly *cp* values per quarter.

Consumption-wealth ratio (*cay*): (1952-2023)

Consistent with Paye (2012), *cay* data are obtained from Amit Goyal's website, using the version updated through 2024.[7]

Current GDP growth (*gdp)*: (1952-2023)

gdp data are sourced from federal reserve economic data (FRED): series A191RL1Q225SBEA: real gross domestic product.

Default return spread (*dfr*): (1927-2023)

dfr data are obtained from Amit Goyal's website, using the version updated through 2024.

Default spread (*dfy*): (1927-2023)

dfy data are also obtained from Amit Goyal's website, using the version updated through 2024.

Expected GDP growth (*egdp*): (1952-2023)

Data required to construct this variable are obtained from the "surveys and data" section of the federal reserve bank of Philadelphia's website.[8]

Expected return (*exret*): (1927-2023)

Forecasting variables for in-sample fitted values of *exret* for different sample periods in our study:

[7] https://sites.google.com/view/agoyal145

[8] https://www.philadelphiafed.org/surveys-and-data/real-time-data-research/livingston-survey

Quarterly:

1947-2010: 'cp_lag', 'dfr_lag', 'dfy_lag', 'npy_lag', 'ppi_lag', 'tms_lag'

1972-2010: 'cp_lag', 'dfr_lag', 'dfy_lag', 'npy_lag', 'ppi_lag', 'tms_lag', 'cay_lag', 'egdp_lag', 'ik_lag'

1982-2010: 'cp_lag', 'dfr_lag', 'dfy_lag', 'npy_lag', 'ppi_lag', 'tms_lag', 'cay_lag', 'egdp_lag', 'ik_lag'

1972-2000: 'cp_lag', 'dfr_lag', 'dfy_lag', 'npy_lag', 'ppi_lag', 'tms_lag', 'cay_lag', 'egdp_lag', 'ik_lag'

1947-2023: 'cp_lag', 'dfr_lag', 'dfy_lag', 'ppi_lag', 'tms_lag'

1947-2019: 'cp_lag', 'dfr_lag', 'dfy_lag', 'npy_lag', 'ppi_lag', 'tms_lag'

1972-2019: 'cp_lag', 'dfr_lag', 'dfy_lag', 'npy_lag', 'ppi_lag', 'tms_lag', 'cay_lag', 'egdp_lag', 'ik_lag'

1972-2023: 'cp_lag', 'dfr_lag', 'dfy_lag', 'ppi_lag', 'tms_lag', 'cay_lag', 'egdp_lag', 'ik_lag'

Monthly:

1947-2010: 'cp_lag', 'dfr_lag', 'dfy_lag', 'ppi_lag', 'tms_lag', 'npy_lag'

1972-2010: 'cp_lag', 'dfr_lag', 'dfy_lag', 'ppi_lag', 'tms_lag', 'npy_lag'

1982-2010: 'cp_lag', 'dfr_lag', 'dfy_lag', 'ppi_lag', 'tms_lag', 'npy_lag'

1972-2000: 'cp_lag', 'dfr_lag', 'dfy_lag', 'ppi_lag', 'tms_lag', 'npy_lag'

1947-2023: 'cp_lag', 'dfr_lag', 'dfy_lag', 'ppi_lag', 'tms_lag'

1947-2019: 'cp_lag', 'dfr_lag', 'dfy_lag', 'ppi_lag', 'tms_lag', 'npy_lag'

1972-2023: 'cp_lag', 'dfr_lag', 'dfy_lag', 'ppi_lag', 'tms_lag'

1972-2019: 'cp_lag', 'dfr_lag', 'dfy_lag', 'ppi_lag', 'tms_lag', 'npy_lag'

Growth in industrial production (*ip*): (1927-2023)

Industrial production data are obtained from the federal reserve economic data (FRED): industrial production: total index (INDPRO) series.

Investment-capital ratio (*ik*): (1952-2023)

ik data are obtained from Amit Goyal's website, using the version updated through 2024.

Net payout (*npy*): (1927-2019)

The data for constructing *npy* are sourced from the website of Michael R Roberts[9].

Term spread (*tms*): (1927-2023)

tms data are sourced from Amit Goyal's website, using the version updated through 2024.

Volatility of growth in industrial production (*ipvol*): (1927-2023)

The data are the same used in constructing *ip*.

Volatility of inflation growth (ppivol): (1927-2023)

This variable is constructed from the series of producer price index by commodity: all commodities (PPIACO), from federal reserve economic data (FRED).

[9] https://finance.wharton.upenn.edu/~mrrobert/research.html