Erasmus School of Economics

MASTER THESIS - MSC QUANTITATIVE FINANCE

Probabilistic forecasting of the equity premium: a tree-based Machine Learning approach

Supervisor:
DR. O. KLEEN

Author:
VINAY JAGGAN

Second assessor:
PROF. DR. C. ZHOU

Student ID:
472642

Internship company:
VB RISK ADVISORY

JULY 31, 2023

The views stated in this thesis are those of the author and not necessarily those of the supervisor, second assessor, Erasmus School of Economics or Erasmus University Rotterdam.

Abstract

This empirical study evaluates the predictive performance of tree-based machine learning models for providing probabilistic forecasts of the equity premium. These models can capture the complex non-linear relationships between the equity premium and their predictors, while also accounting for the associated uncertainty. The monthly U.S equity premium is predicted based on a set of macroeconomic and technical variables. To assess the predictive performance, we compare them against two time-series benchmark models using probabilistic forecasting measures. Our findings reveal that the machine learning models exhibit superior predictive capabilities compared to the traditional time-series models, resulting in better risk assessments. This advantage stems from their ability to effectively capture the time-varying dynamic of the equity premium.

KEYWORDS: PROBABILISTIC FORECASTING - EQUITY PREMIUM - MACHINE LEARNING

Contents

1 Introduction

This empirical study aims to make probabilistic forecasts of the equity premium using probabilistic machine learning (ML) models. The equity premium is defined as the difference between the expected return on stocks and the risk-free rate of return. Traditionally, point predictions have been the primary focus of research in forecasting the equity premium. However, point predictions do not provide information about the level of uncertainty around those outcomes, which can be crucial for investors (Diebold et al., 1998). We focus on the use of tree-based ML models because they are able to capture complex and non-linear relationships between a possibly large set of variables, and can handle interactions between variables. Additionally, they can provide interpretable insights into which variables are most important for predicting the equity premium.

The first model we consider is the Distributional Regression Forest (DRF) introduced by Schlosser et al. (2019). The DRF algorithm combines the strengths of Random Forest (RF) and Distributional Regression and can hence be seen as a probabilistic extension of the RF algorithm. The second tree-based ML model we consider is the Natural Gradient Boosting (NGB) algorithm introduced by Duan et al. (2020). The NGB algorithm modifies the Gradient Boosting algorithm introduced by Friedman (2001) to estimate the parameters of a conditional probability distribution $P(y|x)$ as functions of x. Furthermore, we explore the use of the XGBoostLSS model introduced by März (2019). The XGBoostLSS model is an extension of the widely used Gradient Boosting algorithm XGBoost introduced by Chen and Guestrin (2016), designed to handle probabilistic forecasting tasks. Furthermore, we include the Generalized AutoRegressive Conditional Heteroskedastic (GARCH) model introduced by Bollerslev (1986) as a benchmark model where we obtain density forecasts as in Hoogerheide et al. (2012). Finally, we add the GJR-GARCH model introduced by Glosten et al. (1993). This asymmetric GARCH model uses an indicator function to allow the model to react more to negative shocks in the returns which can improve the predictive performance of the model with respect to the standard GARCH model.

To evaluate the probabilistic forecasts of the models, we use proper scoring rules, such as the Logarithmic Score (LS) and Continuous Ranked Probability Score (CRPS) (Gneiting et al., 2007). Furthermore, we asses calibration plots and evaluate the forecast-implied Value at Risk to show the economic relevance for investors. Moreover, we implement the Model Confidence Set procedure introduced in Hansen et al. (2011) to test for significant differences in forecasting

performance.

Traditional time series models such as Auto-regressive and Moving Average models are based on the assumption of linearity and stationary dynamics, which may not always hold in financial time series data. By using DRF, NGB and XGBoostLSS, we aim to overcome these limitations and provide more accurate probabilistic predictions of the equity premium. Therefore, the main question of this research is formulated as follows: *"How do tree-based ML models perform in providing density forecasts of the equity premium, and how do these compare to the density forecasts obtained from traditional time-series models?"*

We make contributions to the literature on ML and forecasting in financial economics in several ways. Firstly, we aim to make probabilistic forecasts of the equity premium instead of point predictions and provide a framework to evaluate the forecasts. Secondly, we explore the applicability of probabilistic tree-based ML models on financial data. Furthermore, we apply a Walk-Forward testing procedure to account for the temporal dependencies in the time-serie data.

We obtain density forecasts of the U.S equity premium using the updated dataset of Neely et al. (2014). The dataset consists of 15 monthly macroeconomic and 14 technical monthly predictors of the U.S equity premium, spanning a period from January 1960 to December 2021. Here, the equity premium is defined as the return on the S&P 500 index minus the 1-month treasury bill rate. By utilizing an established set of predictive variables, we can focus the evaluation on the out-of-sample (OOS) predictive performance of the models.

The OOS results indicate a consistent superiority of the DRF model over the NGB and XGBLSS models. The main reason behind this performance difference lies in the inability of both Gradient Boosting methods, NGB and XGBLSS, to consistently predict the scale parameter of the distributions accurately. Additionally, the superiority of ML models over time-series models is evident in our findings. The ML models demonstrate a better ability to capture the time-varying aspect of the equity premium, making them more effective in this regard. Furthermore, we demonstrate that the utilization of the DRF model yields improved forecast-implied VaR measures. This is evident as the VaR-threshold is breached less frequently compared to other models, which holds significant economic importance for investors. Lastly, our analysis highlights that the superior model sets mostly consists of ML models, with the GARCH(T) model being the only time-series model included in these sets.

The remainder of the paper proceeds as follows. In Section 2, we present a brief literature review. A description of the data used is provided in Section 3. Next, our methods are introduced

in Section 4. Afterwards, the results are presented in Section 5. Finally, we conclude and discuss our results in Section 6.

2 Literature

This section provides an brief overview of the existing research on forecasting the equity premium and probabilistic forecasting in finance.

Mehra and Prescott (1985) formalized the equity premium puzzle, which refers to the excessively high historical outperformance of stocks over treasury bills that is difficult to explain. This puzzle has motivated extensive research aimed at understanding the determinants of the equity premium and developing predictive models. One of the main challenges in predicting the equity premium is the complexity of the market, which is influenced by a variety of factors such as economic indicators, interest rates, geopolitical events and investor sentiment. Moreover, the equity premium can be highly volatile, with fluctuations over short periods of time. Several papers identify variables that provide predictive power in forecasting the U.S equity premium (Fama and French, 1988), (Fama and French, 1989) and (Campbell and Thompson, 2007). In contrast, Goyal and Welch (2008) conduct an examination of several predictor variables of the equity premium and find low predictive power OOS. While the aforementioned papers use linear regression models to forecast the equity premium, more sophisticated statistical models are also studied in the academic literature (Kelly and Pruitt, 2015), Neely et al. (2014) and Rapach and Zhou (2013).

Recent advancements in ML techniques led to the increase of using ML models for predicting returns both in the cross-section and time-series context. Rapach and Zhou (2020) provides ML methods for both applications. This study focuses on the time-series context. Furthermore, Gu et al. (2020) provides a comparative analysis of ML methods for measuring equity risk premiums. The paper states that Neural Networks and tree-based ML models have a superior predictive performance compared to traditional statistical methods due to allowing non-linear predictor interactions. Conversely, Wolff and Neugebauer (2019) states that compared to sophisticated linear prediction models such as Penalized Least Squares or Principal Component Regressions, ML models do not improve in forecast accuracy. However, an investment strategy that uses ML predictions in a market timing strategy outperforms a passive buy-and-hold investment.

While point predictions have traditionally been used in the literature, we explore the use of probabilistic forecasts for predicting the equity premium. Probabilistic forecasts have gained

significant attention in the academic literature due to their ability to provide valuable insights into uncertainty quantification. Probabilistic forecasting approaches offer insights into various domains, including weather forecasting, election outcome prediction and financial risk management (Gneiting and Katzfuss, 2014). Several studies have presented density forecasts for financial returns. For example, Meligkotsidou et al. (2012) propose a quantile regression approach to predict the equity premium distribution and Beckmann and Schüssler (2014) introduces a Bayesian version of Dynamic Model Averaging for predicting the equity premium. Alternatively, Hoogerheide et al. (2012) uses GARCH models for obtaining density predictions of stock market returns. Density forecasts of financial returns allows for uncertainty quantification and enables investors to assess risk-return trade-offs (Diebold et al., 1998). We contribute to the existing body of literature by examining the potential of probabilistic ML models in forecasting the equity premium distribution. Moreover, we conduct a performance comparison of these models using time-series models instead of predictive regression models. This approach has the benefit of incorporating density forecasts. It is important to note that while there are quantile regression methods available for obtaining density forecasts, these approaches predict specific quantiles of the distribution rather than providing a direct prediction of the conditional distribution, as achieved with Distributional Regression approaches (Koenker et al., 2013).

3 Data

In this study we aim to predict the equity premium using both macroeconomic and technical variables. Specifically, the data consists of 29 monthly variables for the period January 1960 up to and including December 2021. Neely et al. (2014) states that incorporating macroeconomic variables and technical indicators improves equity risk premium forecasts as technical indicators are better at detecting declines near business-cycle peaks, while macroeconomic variables are more effective at capturing rises near cyclical troughs. Section 3.1 provides a more detailed description of the variables used, while Section 3.2 explains how the data is preprocessed.

3.1 Data description

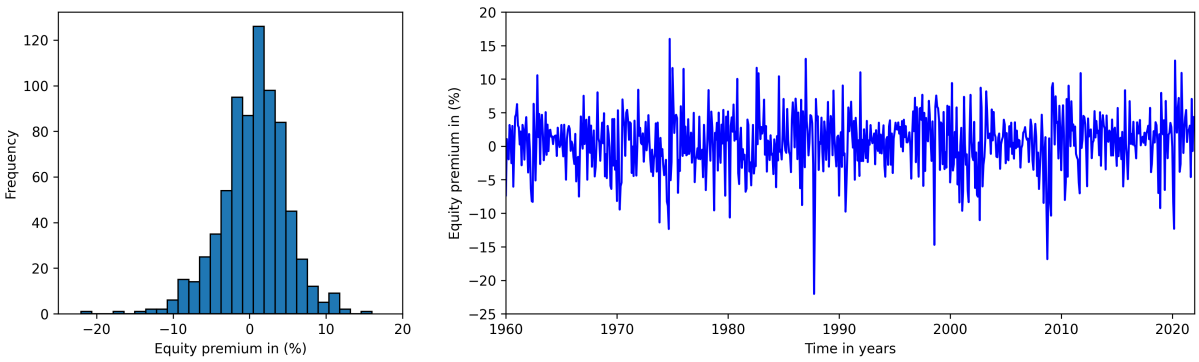
To obtain probabilistic forecasts, we have to make assumptions about the probability distribution used for the equity premium. Therefore, Section 3.1.1 investigates the properties of the variable. Then, in Section 3.1.2 and 3.1.3 we introduce the macroeconomic and technical variables used to predict the equity premium.

3.1.1 U.S Equity premium

The equity premium EP_t is defined as the return on a broad stock market index in excess of the risk-free rate from period $t-1$ to t. In this study we choose the S&P 500 including dividends as the broad stock market index. Furthermore, the risk-free rate is approximated by the return on a 1-month treasury bill.

The left panel of Figure 1 indicates that the U.S equity premium is negatively skewed and thus has a heavier left tail compared to a normal distribution. To formally test for normality we implement the Jarque-Bera test. The null hypothesis (H_0) of this test assumes normality. We obtain a test statistic value of 113.29 and a p-value of 0.00. Thus we reject normality at a significance level of 5% and we expect that a assuming a more heavy tailed distribution for the equity premium will improve the predictive performance of the models. Furthermore, the right panel of Figure 1 suggests that the time-series is stationary. To test this, we implement the Augmented Dickey-Fuller test. We obtain a test statistic of -26.41 with a corresponding p-value of 0.00 so we reject the H_0 of non-stationarity. We thus expect that the time-series exhibits predictable patterns and statistical properties that can be used to make accurate density forecasts.

Figure 1: Histogram and plot over time for the monthly U.S equity premium.



Notes: The left figure shows the histogram of the monthly U.S equity premium for the sample period January 1960 to December 2021. The right figure shows how the equity premium evolves over time.

3.1.2 Macroeconomic variables

We utilize 13 macroeconomic variables which are originally examined in Goyal and Welch (2008). The paper aims to contribute to the understanding of equity premium predictability by providing a comprehensive evaluation of various financial and macroeconomic variables, such as the dividend price ratio, dividend yield and the inflation rate. Furthermore, we added a volatility measure for the equity premium as in Neely et al. (2014). This volatility measure $RVOL_t$ is

defined as

$$RVOL_t = \sqrt{\frac{\pi}{2}}\sqrt{12}\hat{\sigma}_t, \tag{1}$$

where $\hat{\sigma}_t$ is equal to

$$\hat{\sigma}_t = \frac{1}{12}\sum_{i=1}^{12} |r_{t+1-i}| \,. \tag{2}$$

Moreover, the one-month lagged equity premium (EPL_t) is employed to improve the predictive performance of the models. Table 1 gives a brief description of the macroeconomic variables included in this study. Furthermore, Table 6 in Appendix A shows the descriptive statistics of the variables.

Table 1: Description of the macroeconomic variables

Variable	**Description**
Dividend-price ratio (log), DP	log of a twelve-month moving sum of dividends paid on the S&P 500 index minus the log of stock prices
Dividend yield (log), DY	log of a twelve-month moving sum of dividends minus the log of lagged stock prices.
Earnings-price ratio (log), EPR	log of a twelve-month moving sum of earnings on the S&P 500 index minus the log of stock prices.
Dividend-payout ratio (log), DE	log of a twelve-month moving sum of dividends minus the log of a twelve-month moving sum of earnings.
Book-to-market ratio, BM	book-to-market value ratio for the Dow Jones Industrial Average.
Net equity expansion, $NTIS$	ratio of a twelve-month moving sum of net equity issues by NYSE-listed stocks to the total end-of-year market capitalization of NYSE stocks.
Treasury bill rate, TBL	interest rate on a three-month Treasury bill.
Long-term yield, LTY	long-term government bond yield.
Long-term return, LTR	return on long-term government bonds.
Term spread, TMS	long-term yield minus the Treasury bill rate.
Default yield spread, DFY	difference between Moody's BAA- and AAA-rated corporate bond yields.
Default return spread, DFR	long-term corporate bond return minus the long-term government bond return.
Inflation, $INFL$	inflation rate calculated from the Consumer Price Index.
Equity risk premium volatility, $RVOL$	an volatility estimator of the equity risk premium
Lagged equity premium, EPL	the equity premium variable lagged for 1 month

Notes: The macroeconomic variables are calculated as in Goyal and Welch (2008). The price of the S&P 500 index is being referred to as stock prices. The original dataset can be found on the website of Amit Goyal

3.1.3 Technical indicators

We use 14 technical variables described in Neely et al. (2014). The variables are formulated by employing three technical strategies. The first strategy involves a moving-average (MA) rule, which generates buy or sell signals ($S_t = 1$ or $S_t = 0$) at the end of period t by comparing two MAs:

$$S_t = \begin{cases} 1 & \text{if } \mathrm{MA}_{s,t} \geq \mathrm{MA}_{l,t} \\ 0 & \text{if } \mathrm{MA}_{s,t} < \mathrm{MA}_{l,t} \end{cases} \tag{3}$$

where $\mathrm{MA}_{j,t}$ is defined as

$$\mathrm{MA}_{j,t} = \frac{1}{j}\sum_{i=0}^{j-1} P_{t-i} \quad \text{for} \quad j = s, l. \tag{4}$$

Here, P_t denotes the level of the S&P 500 index and s (l) is the length of the short (long) MA. The MA rule detects changes in stock price trends because the short MA will be more sensitive to recent price movements than the long MA. For example, when prices start to rise, the short MA increases faster than the long MA, eventually surpassing it and generating a buy signal. We analyze MA rules for $s = 1, 2, 3$ and $l = 9, 12$.

The second technical strategy is based on momentum. The binary variables are constructed as follows:

$$S_t = \begin{cases} 1 \text{ if } P_t \geq P_{t-m} \\ 0 \text{ if } P_t < P_{t-m} \end{cases}. \tag{5}$$

In essence, when the current stock price surpasses its level from m periods ago, it indicates positive momentum and relatively high expected excess returns, thereby generating a buy signal. We compute monthly signals for $m = 9, 12$.

Finally, we consider a technical strategy that incorporates on-balance volume (OBV). Initially, we compute the OBV_t as

$$OBV_t = \sum_{k=1}^{t} \mathbf{1}\{P_k - P_{k-1} \geq 0\} VOL_k, \tag{6}$$

where VOL_k is a measure of the trading volume during period k. Subsequently, we derive a trading signal based on the value of OBV_t as follows:

$$S_t = \begin{cases} 1 \text{ if } \mathrm{MA}^{OBV}_{s,t} \geq \mathrm{MA}^{OBV}_{l,t} \\ 0 \text{ if } \mathrm{MA}^{OBV}_{s,t} < \mathrm{MA}^{OBV}_{l,t} \end{cases}, \tag{7}$$

where $\mathrm{MA}^{OBV}_{j,t}$ is defined as

$$\mathrm{MA}^{OBV}_{j,t} = \frac{1}{j}\sum_{i=0}^{j-1} OBV_{t-i} \quad \text{for} \quad j = s, l. \tag{8}$$

Intuitively, relatively high recent volume together with recent increases in the price suggest a strong positive market trend and generate a buy signal. We compute monthly signals for $s = 1, 2, 3$ and $l = 9, 12$. Table 7 in Appendix A shows an overview of the technical indicators used.

3.2 Data preprocessing

Firstly, we assess the presence of missing values for the variables described in Section 3.1.2. The sample period, spanning from January 1960 to December 2021, is determined based on the availability of data for these variables. To facilitate the evaluation of our models, we divide the data into an initial training set and a test set. The split is made at January 2002, ensuring that the test set encompasses the last 20 years of data. Subsequently, the training set is further divided into a training set and a validation set, using January 1990 as the dividing point. This partition allows us to perform hyperparameter tuning effectively. After hyperparameter tuning, the validation set is merged back with the training set, enabling the ML models to be trained with the best hyperparameters.

The variables are then standardized to avoid estimation problems within our ML models (Kwak and Kim, 2017). Given the presence of a few severe outliers in the data, as depicted in Figure 1, we employ a robust scaler for standardization. We retain the outliers in the dataset as they hold economic significance. Variable x_i is scaled as follows:

$$x_{i,scaled} = \frac{x_i - x_{i,median}}{x_{i,95th} - x_{i,5th}} \quad \forall i = 1, \ldots, n, \tag{9}$$

where $x_{i,95th}$ and $x_{i,5th}$ respectively are the 95th percentile and the 5th percentile of x_i. Note that for the standardization, we use the median and percentiles of the training data as the test set is regarded as unknown.

4 Methodology

This section describes the models used to obtain probabilistic forecasts of the equity premium. We also explain the model validation and model testing procedures. Furthermore, we introduce the performance measures used to assess the predictive performance of the models. Lastly, we describe the statistical test used to asses if the difference in the numerical scores across models lead to significant difference in forecasting performance.

4.1 Tree-based machine learning models

Sections 4.1.1, 4.1.2 and 4.1.3 introduce the tree-based ML models and explain how the probabilistic forecasts are obtained from the models.

4.1.1 Distributional Regression Forest

The first ML model we consider is the Distributional Regression Forests (DRF) introduced by Schlosser et al. (2019). The DRF algorithm combines the strengths of Random Forest (RF) and Distributional Regression and can hence be seen as a probabilistic extension of the RF algorithm. The idea of RF is to learn an ensemble of Regression Trees, each on different training data obtained through resampling. Furthermore, in each node only a random subset of the variables $\boldsymbol{x}$ is considered for splitting to reduce the correlation among the different trees and to reduce the variance of the model. This way, the RF algorithm often has a higher predictive accuracy and is less prone to overfitting.

The key difference with the RF algorithm is that the DRF model is an ensemble of Distributional Regression Trees (DRTs) instead of Regression Trees. DRTs model the conditional distribution of the dependent variable at each leaf node of each tree, rather than predicting a single value by e.g taking the mean of the dependent variable. This allows the model to capture the uncertainty in the output. Unlike the Distributional Regression models such as the Generalized Additive Models for Location, Scale, and Shape (GAMLSS) introduced by Rigby and Stasinopoulos (2005), DRF doesn't require any assumptions about the functional form of the parameters. This allows for a more flexible modelling approach.

Consider the distributional model $\mathcal{D}(Y, \boldsymbol{\theta})$ for the dependent variable $Y \in \mathcal{Y}$ using the distributional family $\mathcal{D}$ with k-dimensional parameter vector $\boldsymbol{\theta} \in \boldsymbol{\Theta}$ and corresponding log-likelihood function $\ell(\boldsymbol{\theta}; Y)$. The goal is to recursively partition the covariate space $\mathcal{X}$ into B disjoint segments

$$\mathcal{X} = \bigcup_{b=1,\ldots,B} \mathcal{B}_b, \tag{10}$$

so that a homogeneous distributional model $\mathcal{D}(Y, \boldsymbol{\theta}^{(b)})$ can be fitted to the dependent Y in each segment with segment-specific parameters $\boldsymbol{\theta}^{(b)}$. To find the segments $\mathcal{B}_b$, an DRT is fitted as follows:

1. Start with the training sample $(y_1, \ldots, y_T)$ and define the current subsample as the entire sample.

2. For the current subsample estimate the parameter values $\hat{\boldsymbol{\theta}}$ using maximum likelihood and calculate the scores $s(\hat{\boldsymbol{\theta}}, y_t)$ for each observation y_t in the current subsample.

3. Test for significant associations/instabilities of $s(\hat{\boldsymbol{\theta}}, y_t)$ and $X_{j,t}$ for each partitioning variable x_j $(j = 1, \ldots, p)$.

4. Select the partitioning variable x_j^* with the strongest association/instability and select the breakpoint b^* that results in the highest improvement in the log-likelihood/highest discrepancy. Split the subsample into two subsamples with the chosen x_j^* and b^*.

5. Repeat steps 2-4 recursively, until the subsamples become too small or there is no statistical significant association/instability.

In step 2, the score function is defined as

$$s(\hat{\boldsymbol{\theta}}, y_t) = \frac{\partial \ell}{\partial \boldsymbol{\theta}}(\boldsymbol{\theta}, y_t) \tag{11}$$

and we test for associations/instabilities with the general class of permutation tests introduced by (Hothorn et al., 2006).

To obtain probabilistic forecasts of a DRT the test observation "falls down" the tree and ends up in a particular segment. In this way, the corresponding segment-specific $\hat{\boldsymbol{\theta}}_{MLE}$ can be extracted. Thus, instead of computing the parameter estimates for a new set of variables $\boldsymbol{x}$, the DRTs can utilize the estimates that were obtained during the training process.

As mentioned earlier, DRF is an ensemble of M DRTs which aims to prevent overfitting and provides more accurate predictions. Each DRT in a DRF predicts the conditional distribution of the dependent variable for a subset of the variables. Subsequently, DRF takes a weighted average of the predicted distributions from each DRT.

To obtain probabilistic forecasts of the DRF algorithm, we compute the weights $w_t^{forest}(\boldsymbol{x})$ as

$$w_t^{forest}(\boldsymbol{x}) = \frac{1}{M} \sum_{m=1}^{M} \sum_{b=1}^{B^m} \frac{\mathbf{1}((\boldsymbol{x}_t \in \mathcal{B}_b^m) \wedge (\boldsymbol{x}_t \in \mathcal{B}_b^m))}{|\mathcal{B}_b^m|}, \tag{12}$$

where $|\mathcal{B}_b^m|$ denotes the number of observations in the b-th segment of the m-th tree. A larger weight is given for those observations t from the training sample that occur in the same segment $\mathcal{B}_b^m$ as the new observations $\boldsymbol{x}$ for a large number of trees $m = 1, \ldots, M$. Suppose we forecast H periods OOS. We can compute the parameter estimates $\hat{\boldsymbol{\theta}}_k(\boldsymbol{x})$ for a new set of variables $\boldsymbol{x}$ by

$$\hat{\boldsymbol{\theta}}_k(\boldsymbol{x}) = \arg\max_{\boldsymbol{\theta} \in \Theta} \sum_{t=1}^{T} w_t^{forest}(\boldsymbol{x}) \cdot \ell(\boldsymbol{\theta}_t; y_t) \qquad k = T+1, \ldots, T+H. \tag{13}$$

4.1.2 Natural Gradient Boosting

The second tree-based ML model we examine is the Natural Gradient Boosting (NGB) algorithm introduced by Duan et al. (2020). The NGB model modifies the Gradient Boosting algorithm introduced by Friedman (2001) to estimate the parameters of a conditional probability distribution $P(y|\boldsymbol{x})$ as functions of $\boldsymbol{x}$. Gradient Boosting works by building a sequence of shallow decision trees that are combined to make a prediction. In boosting stage m ($m = 1, \ldots, M$), a decision tree is trained on the residuals of the previous trees, with the goal of minimizing a loss function.

A few differences between the Gradient Boosting algorithm and NGB are that a series of decision trees must be fit for each parameter instead of one. Furthermore, a parametric probability distribution $P_{\boldsymbol{\theta}}$ and a proper scoring rule $\mathcal{S}$ (see Section 4.4) must be specified to incorporate a probabilistic learning objective for the algorithm. Moreover, the Natural Gradient is used instead of the standard gradient because this makes the optimization problem invariant to reparametrization of the distribution parameters. Specifically, for iteration m and observation t ($t = 1, \ldots, T$), the algorithm calculates the Natural Gradient $\boldsymbol{g}_t^{(m)}$ of $\mathcal{S}$ with respect to the predicted k-dimensional parameter $\boldsymbol{\theta}_t^{(m-1)}$ as

$$\boldsymbol{g}_t^{(m)} = \mathcal{I}_{\mathcal{S}}\left(\boldsymbol{\theta}_t^{(m-1)}\right)^{-1} \nabla_{\theta}\mathcal{S}\left(\boldsymbol{\theta}_t^{(m-1)}, y_t\right), \tag{14}$$

where $\mathcal{I}_{\mathcal{S}}(\boldsymbol{\theta})$ is the Fisher Information. Subsequently, a k-dimensional set of regression trees for iteration m denoted as $\boldsymbol{f}^{(m)}$), are fit to predict the corresponding components of $\boldsymbol{g}_t^{(m)}$ and observation $\boldsymbol{x}_t$. Then the predicted parameters are updated as follows:

$$\boldsymbol{\theta}_t^{(m)} = \boldsymbol{\theta}_t^{(m-1)} - \eta\left(\rho^{(m)} \cdot \boldsymbol{f}^{(m)}(\boldsymbol{x}_t)\right), \tag{15}$$

where $\rho^{(m)}$ is a stage-specific scaling factor chosen to minimize the scoring rule loss along the direction of the projected gradient and η is a common learning rate.

The probabilistic forecasts of the NGB model are obtained by the parameters $\hat{\boldsymbol{\theta}}_t$ via an additive combination of the M tree outputs as

$$\hat{\boldsymbol{\theta}}_t = \hat{\boldsymbol{\theta}}^{(0)} - \eta \sum_{m=1}^{M} \rho^{(m)} \cdot f^{(m)}(\boldsymbol{x}_t), \qquad t = T+1, \ldots, T+H. \tag{16}$$

where $\hat{\boldsymbol{\theta}}^{(0)}$ is estimated using Maximum Likelihood over the entire training sample.

4.1.3 Extreme Gradient Boosting for Location Scale and Shape

The third ML model we examine is the Extreme Gradient Boosting algorithm for Location, Scale and Shape (XGBoostLSS) introduced by März (2019) which is an extension of the XGBoost

algorithm. XGBoostLSS builds upon XGBoost and incorporates principles of the GAMLSS framework to obtain density predictions. In GAMLSS, density estimation involves modeling the distributional parameters of a predictor variable. To understand how the XGBoostLSS algorithm estimates the distributional parameters, we provide a brief overview of how XGBoost works.

XGBoost is a gradient boosting algorithm introduced in Chen and Guestrin (2016). At each iteration m, the algorithm estimates the predictor variable by minimizing a regularized objective function:

$$\tilde{\mathcal{L}}^{(m)} = \sum_{t=1}^{T} \ell \left[y_t, \hat{y}_t^{(m-1)} + f_m \left(\mathbf{x}_t \right) \right] + \Omega \left(f_m \right). \tag{17}$$

The objective function consists of two components: a differentiable convex loss function ℓ that measures the discrepancy between the predicted values and the true values, and a regularization term $\Omega(\cdot)$ that penalizes the complexity of the model to prevent overfitting. To approximate $\ell[\cdot]$ and simplify Equation (17), a second-order Taylor expansion is employed. This approximation results in the following objective function:

$$\tilde{\mathcal{L}}^{(m)} = \sum_{t=1}^{T} \left[g_t f_m \left(\mathbf{x}_t \right) + \frac{1}{2} h_t f_m^2 \left(\mathbf{x}_t \right) \right] + \gamma K + \frac{1}{2} \lambda \sum_{j=1}^{K} w_j^2, \tag{18}$$

where $g_t = \partial_{\hat{y}^{(m-1)}} \ell \left[y_t, \hat{y}_t^{(m-1)} \right]$ and $h_t = \partial^2_{\hat{y}^{(m-1)}} \ell \left[y_t, \hat{y}_t^{(m-1)} \right]$ are respectively first and second order derivatives, w_j are leaf weights, γ is a parameter that controls the penalization for the number of terminal nodes K of the trees and λ is an L_2 regularization term on the leaf weights.

For the XGBoostLSS algorithm the distributional parameters θ_k $(k = 1, \ldots, K)$ are similarly to GAMLSS estimated using the first and second order partial derivatives of the log-likelihood function with respect to that specific parameter. If we specify the loss function in Equation (17) as an appropriate log-likelihood function, Maximum Likelihood can be formulated as empirical risk minimization so that the resulting XGBoostLSS model can be interpreted as a statistical model. XGBoostLSS updates the boosting ensemble by fitting regression trees to the negative gradients and Hessians of the log-likelihood function. These trees are added to the ensemble to improve the predictions iteratively. With each iteration, XGBoostLSS updates the estimations of the distributional parameters based on the learned ensemble. Finally, the density predictions are obtained with the predicted parameters as in Equation (16).

The key characteristic that sets XGBoostLSS apart from other boosting approaches is its use of Newton Boosting, also known as second-order gradient boosting. In Newton Boosting, the loss function $\ell[\cdot]$ is approximated as in Equation (18) which generally results in outperformance of Gradient Boosting because of the variability in the second-order terms (Sigrist, 2021).

4.2 Time-series models

To compare the density predictions of the ML models, we examine an extension of the Generalized AutoRegressive Conditional Heteroskedastic (GARCH) model introduced by Bollerslev (1986). Specifically, we add the AR(1)-GARCH(1,1) as a benchmark model which extends the GARCH(1,1) model by incorporating an autoregressive term. Adding an autoregressive term allows the model to capture the dependence between the equity premium at time t and its lagged value. The mean equation is as follows:

$$y_t = \mu + \phi y_{t-1} + \epsilon_t. \tag{19}$$

Moreover, the equation for the conditional variance is given by

$$\sigma_t^2 = \omega + \alpha \epsilon_{t-1}^2 + \beta \sigma_{t-1}^2, \tag{20}$$

where μ, ϕ, ω, α, and β are the parameters of the model that are estimated using Maximum Likelihood. Furthermore, y_t (y_{t-1}) represents the (lagged) equity premium. Moreover, ϵ_t (ϵ_{t-1}) is the (lagged) stochastic innovation term and σ_{t-1}^2 is the lagged value of the conditional variance.

Finally, we consider the AR(1)-GJR-GARCH(1,1) model which is an extension of the GJR-GARCH model introduced in Glosten et al. (1993). This model has a mean equation as defined in Equation (19) and has a similar conditional variance equation except for the indicator variable added to allow for asymmetric effects depending on the sign of ϵ_{t-1}:

$$\sigma_t^2 = \omega + (\alpha_1 \mathbf{1}_{\epsilon_{t-1} \geq 0} + \alpha_2 \mathbf{1}_{\epsilon_{t-1} \leq 0}) \epsilon_{t-1}^2 + \beta \sigma_{t-1}^2. \tag{21}$$

To obtain density predictions for the equity premium, we assume a probability distribution for the equity premium. For instance, by assuming a Normal distribution, the probability density function (PDF) for each y_t is given by:

$$f(y_t | \mathcal{I}_{t-1}) = \frac{1}{\sqrt{2\pi \hat{\sigma}_{t|t-1}^2}} \exp\left(-\frac{(y_t - \hat{\mu}_{t|t-1})^2}{2\hat{\sigma}_{t|t-1}^2} \right) \tag{22}$$

where $\mathcal{I}_{t-1}$ represents the information set up to time $t-1$ and contains the parameter forecasts $\hat{\theta}_{t|t-1}$. We iteratively calculate the conditional mean and variance for each time period, using the estimated parameters. We then substitute these values into the PDF to obtain the density forecasts. Section B in the appendix shows how we obtain density predictions from the models assuming a Student's t-distribution. Furthermore, Table 2 provides an overview of all models utilized in this study.

Table 2: Overview of all models and distributions

Model	**Normal distribution**	**Student's t-distribution**
ML models		
Distributional Regression Forest	DRF(N)	DRF(T)
Natural Gradient Boosting	NGB(N)	NGB(T)
XGBoost for location, scale and shape	XGBLSS(N)	XGBLSS(T)
time-series models		
AR(1)-GARCH(1,1)	GARCH(N)	GARCH(T)
AR(1)-GJR-GARCH(1,1)	GJR-GARCH(N)	GJR-GARCH(T)

Notes: "$Model(D)$" represents the model and distribution employed. Here, N (T) denotes the Normal distribution (Student's t-distribution).

4.3 Walk-Forward testing and hyperparameter tuning

When dealing with time-series data, the presence of temporal dependencies can render traditional Cross-Validation methods inadequate. Such methods may produce unrealistic performance estimates as they overlook the temporal aspect. By training the model on past data and evaluating its performance on future data, the assumption of independence between the past and future data is violated, leading to data leakage.

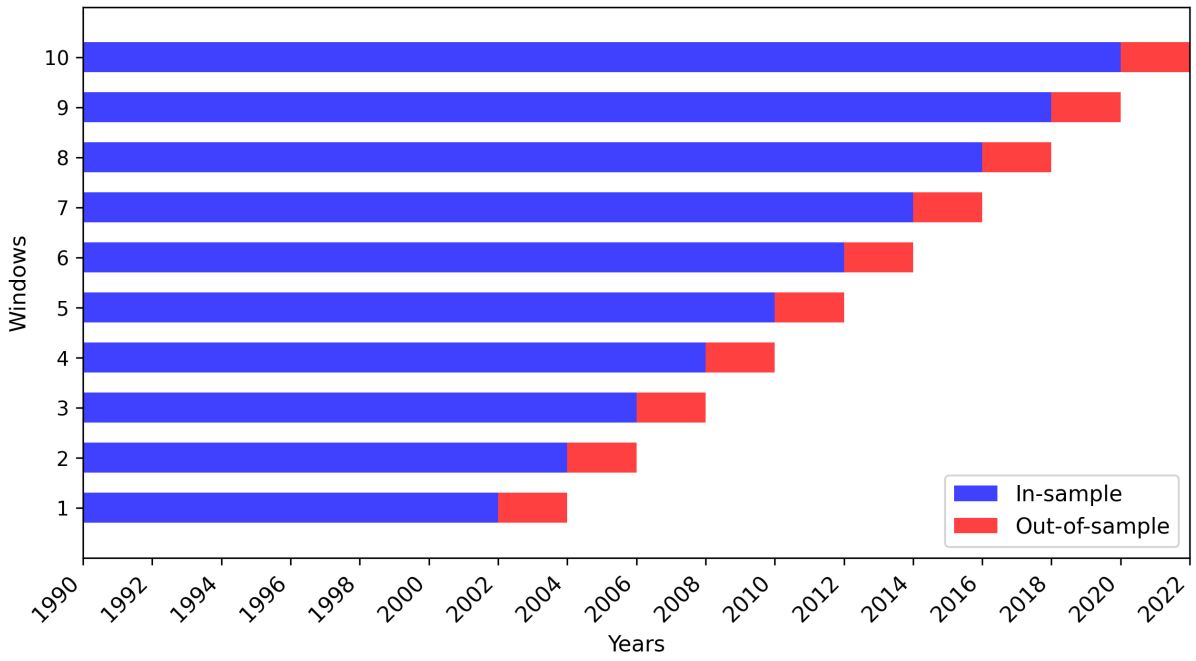
To address these limitations, we employ a Walk-Forward testing procedure as an alternative evaluation technique for our ML models. Walk-Forward testing is a sequential evaluation method that involves iteratively training the model on historical data up to a specific point, making predictions for a future time period and then retraining the model with the newly observed data. This process is repeated, progressively moving the training and testing windows forward in time. This allows the model to adapt and learn from changing patterns in the time-series data and provides a more accurate estimation of how the model would perform in practice.

Figure 2 shows a visualization of the Walk-Forward testing procedure. Note that we implement a fixed sliding window approach where the in-sample data at the start of each window shifts by the same size as the length of the in-sample period at the end.

Learning the optimal hyperparameters is performed through a grid search on the different hyperparameter combinations. However, we only use one fold for hyperparameter tuning because updating the hyperparameters in the walk-forward testing approach proves computationally expensive for certain models. We employ the period 1990:01-2001:12 for the hyperparameter

tuning process and the best hyperparameters selected for the models are provided in Tables 8 and 9 in Appendix C. Additionally, most parameters are set to default values as we have no theory about the optimal values for financial time-series data.

Figure 2: The Walk-Forward testing procedure for ML models.



Notes: The sliding window is of fixed size. The OOS period is from January 2002 to December 2021. Each fold consists of 42 years in-sample (IS) and two years OOS.

4.4 Probabilistic forecasts and performance measures

A probabilistic forecast takes the form of a predictive probability distribution over future quantities or events of interest. Gneiting et al. (2007) states that probabilistic forecasting aims to maximize the sharpness of the predictive distributions, subject to calibration, on the basis of the available information set. The calibration of a probabilistic forecast refers to the statistical compatibility of probabilistic forecasts and observations. Essentially, realizations should be indistinguishable from random draws from predictive distributions. On the other hand, the sharpness of a probabilistic forecast relates to how concentrated the predictive distribution is, and this aspect is specific to the forecasts themselves.

4.4.1 Calibration measures

To asses the calibration of the probabilistic forecasts we plot the Probability Integral Transformation (PIT) histograms as in Diebold et al. (1998). The PIT relates to the result that data

values that are modeled as being random variables from any given continuous distribution, can be converted to random variables having a standard Uniform distribution. This transformation is accurate when the distribution utilized in the model matches the true distribution of the random variables. Interpreting the shape of the PIT histogram provides insights into the dispersion characteristics of the predictive distributions. A U-shaped histogram indicates an underdispersed predictive distribution, implying that the the predicted distribution is narrower than the actual distribution of the data. Conversely, an inverse U-shaped histogram implies an overdispersed predictive distribution, implying that the predicted distribution is wider than the actual distribution of the data which can lead to overestimated standard errors. In order to formally assess whether the PIT values conform to a Uniform distribution, we utilize the one-sample Kolmogorov-Smirnov (KS) test as described in Massey (1951). Here, the null hypothesis is that the PIT values exhibit a Uniform distribution.

Furthermore we analyze the Quantile-Quantile plots (QQ-plots) to assess the goodness-of-fit by comparing the quantiles of the observed data against the quantiles of a theoretical distribution. Departures from the expected straight line indicate deviations from the assumed distribution, revealing potential model misspecification.

4.4.2 Proper scoring rules

Scoring rules assess the calibration and sharpness of probabilistic forecasts jointly, by assigning a numerical score $\mathrm{S}(F, y)$ to each pair (F, y), where $F \in \mathcal{F}$ is a probabilistic forecast and $y \in \mathbb{R}$ is the realized value. A scoring rule is considered proper if the forecaster maximizes the expected score for an observation drawn from the distribution F when they issue the probabilistic forecast F, rather than a different distribution $G \neq F$ (Gneiting and Katzfuss, 2014). The use of proper scoring rules is crucial and we refer the interested readers to Gneiting et al. (2007) for a case study that provides an example of the potential issues that result from the use of improper scoring rules. Generally, we take the scoring rules to be negatively oriented penalties that forecasters wish to minimize.

To compare the probabilistic forecasts of the various models we present two specific proper scoring rules in this study. The first proper scoring rule is the logarithmic score (LS) introduced by Good (1952). The LS is defined as follows:

$$\mathrm{LS}(f, y) = -\log f(y), \tag{23}$$

where $f(\cdot)$ is the probability density function (PDF) and y is the actual outcome. The LS

evaluates how well a density forecast matches the actual outcome. Another proper scoring rule we use as a performance measure is the Continuous Ranked Probability Score (CRPS) described in Gneiting et al. (2007). The CRPS measures the integrated squared difference between the forecasted CDF and a unit step function centered on the observed value. It is defined by the following equations:

$$\begin{aligned} \text{CRPS}(F, y) &= \int_{-\infty}^{\infty} (F(x) - \mathbf{1}\{y \leq x\})^2 \, \mathrm{d}x \\ &= \mathbb{E}_F|Y - y| - \frac{1}{2}\mathbb{E}_F\left|Y - Y'\right|, \end{aligned} \tag{24}$$

where Y and Y' are independent random variables with cumulative distribution function (CDF) F and finite first moment. A lower CRPS value indicate a better agreement between the forecasted CDF and the actual outcome. Note that the CRPS generalizes the Absolute Error for point forecasts, as indicated by the second line in Equation (24).

4.4.3 Forecast-implied Value at Risk

In order to demonstrate the economic significance of accurately predicting the equity premium distribution in comparison to relying on the conditional mean, we assess the forecast-implied Value at Risk measure. For a Normal distribution, the monthly VaR at a confidence level of $(1 - \alpha) * 100\%$ is given by

$$VaR^{\alpha}_{t|t-1} = \hat{\mu}_{t|t-1} + \hat{\sigma}_{t|t-1}\Phi^{-1}(\alpha), \tag{25}$$

where $\hat{\mu}_{t|t-1}$ is the predicted location parameter, $\hat{\sigma}_{t|t-1}$ is the predicted scale parameter and $\Phi^{-1}(\alpha)$ is the inverse CDF of the standard Normal distribution at confidence level α. Since we study returns we are interested in the left tail of the distribution.

Similarly, for a Student's t-distribution, the monthly VaR at a confidence level α with $\hat{\nu}$ degrees of freedom is given by:

$$VaR^{\alpha}_{t|t-1} = \hat{\mu}_{t|t-1} + \hat{\sigma}_{t|t-1}t^{-1}(\alpha, \hat{\nu}), \tag{26}$$

where $t^{-1}(\alpha, \hat{\nu})$ is the inverse CDF of the Student's t-distribution at confidence level of $1 - \alpha * 100\%$ with $\hat{\nu}$ degrees of freedom. We evaluate their accuracy by comparing them to the realized equity premium. The VaR violation ratio measures how often the realized equity premium exceeds the forecasted VaR and is calculated as:

$$\text{Violation Ratio} = \frac{1}{H-1}\sum_{t=T+1}^{T+H} \mathbf{1}\{EP_t < VaR^{\alpha}_{t|t-1}\}, \tag{27}$$

where H is the number of observations in the OOS period. A lower VaR violation ratio indicates that the model's forecasts align well with the observed data, suggesting reliable predictions. Conversely, a higher violation ratio suggests that the VaR estimates are to conservative, leading to more frequent breaches of the VaR threshold.

4.5 Model Confidence Set

To assess if the probabilistic forecasts from the various models exhibit statistically significant differences in forecast performance, we employ the Model Confidence Set (MCS) procedure described in Hansen et al. (2011). The MCS is a statistical test procedure used to sequentially determine the most accurate forecasting models among a group of competing models. We utilize the MCS procedure because it acknowledges the limitations of the data by selecting the best model when the data is highly informative, but includes multiple models when the data is less informative. It also enables valid statements about significance, unlike the commonly used approach of reporting p-values from multiple pairwise comparisons.

To illustrate the functioning of the MCS procedure, let $\mathcal{M}^0$ be the set that consists of all models where the forecasts of the models are evaluated using a performance measure S as in Equations (23) and (24). Furthermore, define the relative forecasting performance of the models on time t as

$$d_{ij,t} = S_{i,t} - S_{j,t} \quad \forall i, j \in \mathcal{M}^0, \quad t = T+1, \ldots, T+H \tag{28}$$

where $S_{i,t}$ and $S_{j,t}$ represent the performance of models i and j at time t, respectively. In addition, we denote $\bar{d}_{ij}$ as the average relative forecasting performance of the models. The objective of the MCS procedure is to determine the set $\mathcal{M}^0 = \{i \in \mathcal{M}^0 : \bar{d}_{ij} \leq 0, \quad \forall j \in \mathcal{M}\}$ through a sequence of statistical tests. The null hypothesis is that the predictive performance of all models in $\mathcal{M}$ is equal:

$$H_{0,\mathcal{M}} : \bar{d}_{ij} = 0 \quad \forall j \in \mathcal{M}, \quad \mathcal{M} \subset \mathcal{M}^0. \tag{29}$$

Moreover, the test statistic is calculated as

$$t_{ij} = \bar{d}_{ij} / \sqrt{\widehat{\mathrm{Var}}\left(d_{ij}\right)} \quad \forall i, j \in \mathcal{M}^0 \tag{30}$$

and the MCS test statistic is given by $T_{\mathcal{M}} = \max_{i,j \in \mathcal{M}} |t_{ij}|$. The MCS procedure iteratively tests the null hypothesis and eliminates the worst performing model from consideration each time the null hypothesis is rejected. This process continues until no further model can be excluded, resulting in a final set of models with superior forecasting ability for a confidence level of $1 - \alpha$.

Following Hansen et al. (2011), we approximate the non-standard asymptotic distribution of the test statistic $T_{\mathcal{M}}$ using block bootstrapping with a block length of 20. Furthermore, we conduct 10000 bootstrap replications at each stage to ensure robustness and reliable outcomes.

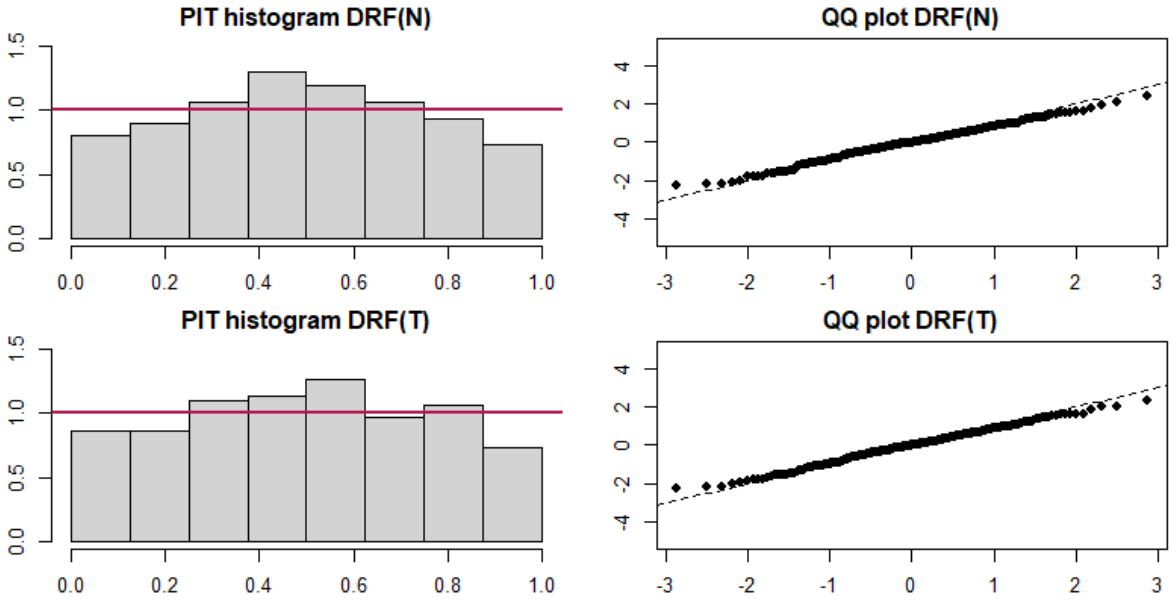
5 Results

Section 5.1 shows the results regarding the calibration of the models. Moreover, Section 5.2 displays the results for the relative predictive accuracy of the models. Furthermore, we show the final model sets obtained from the MCS procedure in Section 5.3. Finally, we illustrate the economic relevance of density forecasts using the forecast-implied VaR in Section 5.4.

5.1 Calibration assessment

Figure 3 shows the PIT histograms and QQ plots for the DRF models. We see that for both the Normal and the Student's t-distribution the PIT histograms are distributed Uniform. The KS-test results support this with KS test statistics of 0.06 and 0.05, along with corresponding p-values of 0.32 and 0.70. Therefore, we do not reject the H_0 of Uniformity. Additionally, the QQ plots demonstrate minimal deviations from the theoretical line of quantiles, indicating a strong alignment between the theoretical and empirical quantiles. These observations collectively suggest that there is no evidence of model misspecification.

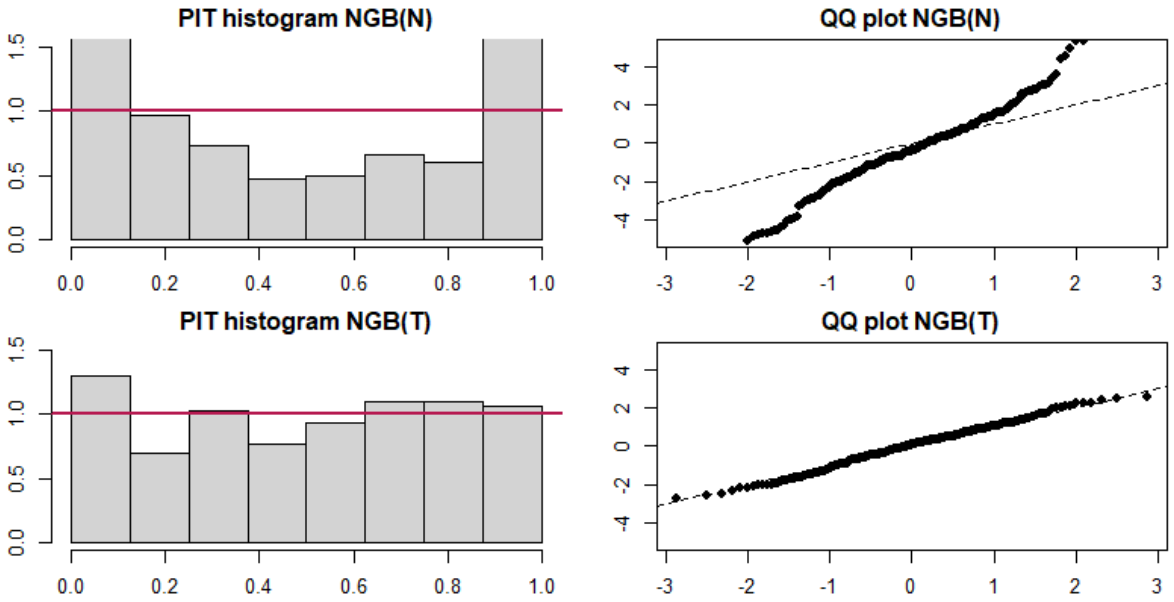
Figure 3: PIT histograms and QQ plots for the DRF models.



Notes: On the left side, the figure displays PIT histograms and on the right side it shows the QQ plots. These visualizations represent the PIT values derived from the DRF model predictions for the OOS period 2002:01-2021:12.

Secondly, Figure 4 shows the PIT histograms and QQ plots for the NGB models. We see for the Normal distribution an U-shaped PIT histogram which indicates an underdispersed predictive distribution. This suggests that the predicted distribution is narrower than the distribution of the realized equity premium. Moreover, the KS test statistic is equal to 0.20 with a p-value of 0.00, leading to the rejection of the H_0 of Uniformity. The large deviations from the QQ-line also indicate that the model is misspecfied. On the other hand, the PIT histogram for the Student's t-distribution appears to be distributed Uniform. This observation is supported by the KS-test, which reports a KS test statistic of 0.05 with a p-value of 0.58. In addition, the QQ plot for the NGB(T) model shows minimal deviations from the theoretical quantiles. This indicates that the more heavy-tailed Student's t-distribution is a more appropriate choice for modeling the equity premium compared to the Normal distribution.

Figure 4: PIT histograms and QQ plots for the NGB models.

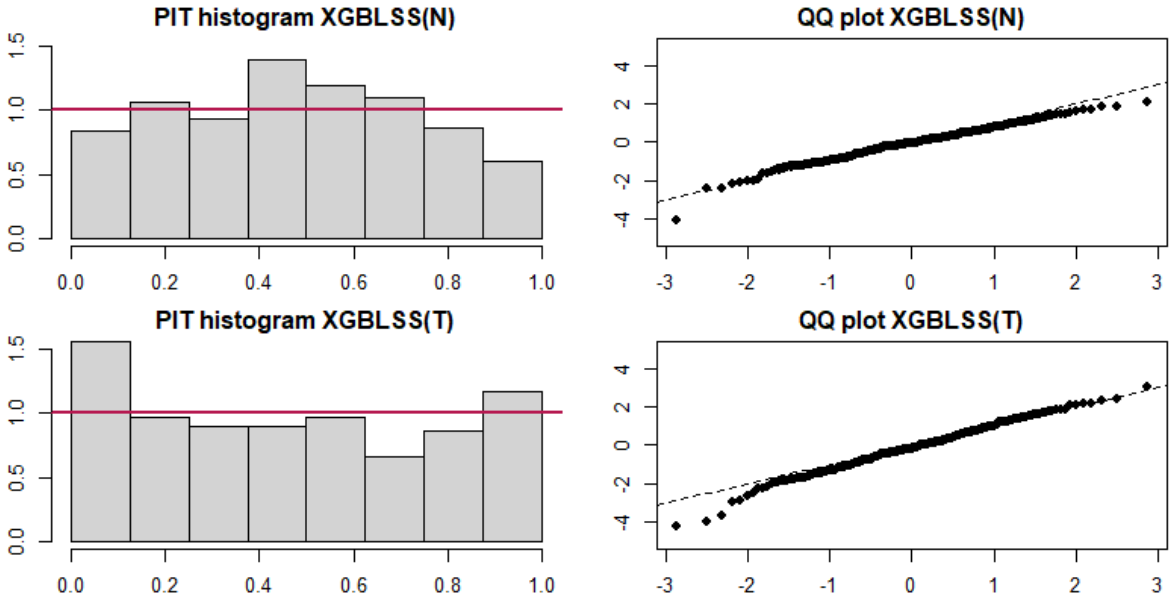


Notes: On the left side, the figure displays PIT histograms and on the right side it shows the QQ plots. These visualizations represent the PIT values derived from the NGB model predictions for the OOS period 2002:01-2021:12.

Figure 5 displays the PIT histograms and QQ plots for the XGBLSS models. The PIT histogram for the XGBLSS(N) model shows small peaks in the middle, such that it is unclear if the PIT values are Uniformly distributed. The KS-test gives KS test statistics of 0.07 with a corresponding p-value of 0.17. Therefore, we do not reject the H_0 of Uniformity. On the other hand, the PIT histogram for the Student's t-distribution exhibits peaks at its ends, implying potential overdispersion in the data. However, the KS test yields a test statistic of 0.08 with a corresponding p-value of 0.08, such that we don't reject the H_0 of Uniformity at a significance level of 5%.

Given these mixed results for the two distributional assumptions, we analyze the forecasting performance for the models incorporating both the Normal and Student's t-distribution in the remainder of this study. The PIT histograms and QQ plots for the two time-series models can be found in Appendix D.

Figure 5: PIT histograms and QQ plots for the XGBLSS models.



Notes: On the left side, the figure displays PIT histograms and on the right side it shows the QQ plots. These visualizations represent the PIT values derived from the XGBLSS model predictions for the OOS period 2002:01-2021:12.

5.2 Comparison probabilistic forecasts

First, in Sections 5.2.1 we compare the forecasts over time Then, in Section 5.2.2 we asses the average forecasting performance of the models.

5.2.1 Distribution parameter forecasts over time

In Figure 6, we present a visualization of the forecasted distributional parameters of the models over time. The plot shows the predicted conditional mean of the Student's t-distribution, accompanied by confidence intervals (CIs). For a confidence level α, a CI is obtained utilizing the conditional scale parameter of the Student's t-distribution. Note that a good density prediction does not necessarily require its conditional mean to exactly match the realized equity premium, as it focuses on capturing uncertainty and providing a well-calibrated probability distribution.

For the DRF(T) model, we see that predicted conditional mean often deviates from the realized equity premium. This deviation is even larger for financial crisis periods (marked grey) when the realized equity premium is more volatile. Moreover, we observe that the realized equity premium falls within the 30% and 50% CIs most of the time. In addition, during volatile periods, it often lies outside the 90% CI. The scale parameter remains relatively stable during non-crisis

periods, as suggested by the narrow CIs. Furthermore, in crisis periods the scale parameter increases, leading to wider CIs.

Similar to the DRF(T) model, the NGB(T) model predictions for the conditional mean of the equity premium show small deviations in non-crisis periods and larger deviations in crisis periods. Moreover, the scale parameter tends to be too small during non-crisis periods which is observed by the conservative CIs. Conversely, the model produces very large scale parameter predictions during financial crisis periods, resulting in wider CIs.

The XGBLSS(T) model also shows similar results for the conditional mean. Furthermore, we see that both Gradient boosting methods, XGBLSS(T) and NGB(T), show difficulties predicting the scale parameter. However, the CIs in crisis periods are smaller compared to the NGB(T) model.

Comparing the distributional parameters for the ML models, we observe that the scale parameter of the distribution for the DRF(T) model exhibits higher consistency in its predictions compared to Gradient Boosting methods. This is probably because the DRF model is less susceptible to overfitting, as it uses random data sampling and variable selection. Moreover, the Gradient Boosting methods fit the trees to the errors of the previous trees, potentially capturing noise in the equity premium data.

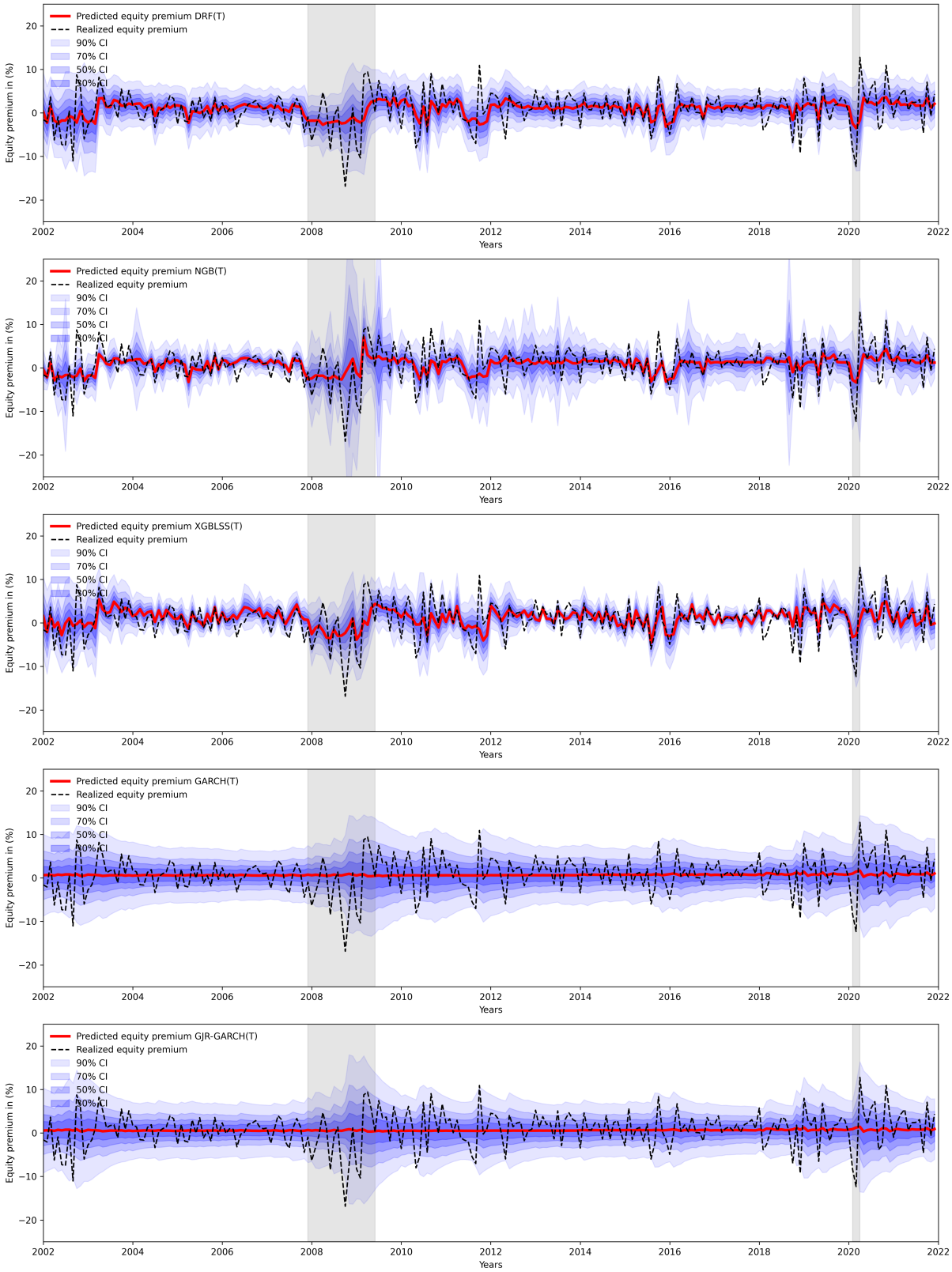
The time-series models GARCH(T) and GJR-GARCH(T) yield similar plots. In both cases, it is apparent that the models struggle to capture the time-varying component of the equity premium, as the predicted mean remains almost constant over time. This could be attributed to the AR(1) coefficient being close to zero. Moreover, larger scale parameters are observed during periods of higher volatility in the equity premium, resulting in wider confidence intervals.

Comparing the ML models with the time-series models, we see that the ML models give better predictions for the conditional mean. The predictions reflect better that the equity premium is varying over time and the CIs are smaller. Furthermore, it is evident that the scale parameter of the ML models shows greater variation over time, thereby providing a more accurate representation of the volatility of the equity premium. However, it should be noted that the GJR-GARCH(T) model predicts the volatility of the equity premium more accurately, as evidenced by the realized equity premium falling within the 90% CI for most observations.

Figure 9 in Appendix E shows the conditional mean with CIs over time for all models assuming a Normal distribution. We observe similar results as for the models included in Figure 6. The Figure also shows that predictions of the conditional means are generally inaccurate in crisis periods. Furthermore, we observe wider CIs in more volatile periods of the realized

equity premium. On the other hand, the NGB(N) model predictions show large deviations of the conditional mean and the realized equity premium both in crisis and non-crisis periods. We also observe smaller CIs compared to all ML models. This outcome is attributed to the model's tendency to predict small values for the scale parameter.

Figure 6: Predicted conditional mean of the equity premium over time



Notes: Each panel in the figure shows the predicted $\hat{\mu}_{t|t-1}$ (red line) for a model assuming a Student's t-distribution. The bands correspond to 90%, 70%, 50% and 30% CIs and obtained by $\hat{\mu}_{t|t-1} \pm t^{-1}(\alpha/2, \hat{\nu})\ \hat{\sigma}_{t|t-1}$ where $\hat{\sigma}_{t|t-1}$ is the scale parameter. Furthermore, crisis periods are marked grey.

5.2.2 Average forecasting performance

Table 3 shows the average LS and CRPS over time for the model forecasts. Note that a lower score corresponds to a better forecasting performance of the model on average. We observe for the DRF(N) and DRF(T) model a LS of 2.54 and 2.55 respectively. These scores are lower compared to the other ML and time-series models. This is in line with the results observed in Section 5.1 where we see that the models are well calibrated. Furthermore, we see for the NGB(N) model a LS of 4.10 which is higher then the LS for all other models including the benchmark GARCH(N). This is mainly due to the fact that the NGB(N) model provides incorrect density estimates in the crisis periods 2008-2010 as can be observed in Figure 9. Therefore, the actual outcome has a low probability and hence a high LS.

Comparing the models based on the CRPS we observe that the DRF(N) and DRF(T) models have a lower score. For example, The DRF(N) model has a CRPS of 1.86 which is smaller then the CRPS of the GARCH(N) model (2.26). Furthermore, we see that the NGB(N) model has a higher CRPS compared to other ML models. However, it outperforms the GARCH(N) model based on the CRPS (1.96¡2.26). Generally, we observe that the CRPS scores for the models assuming a Student's t-distribution have on average a lower CRPS score compared to the same models assuming a Normal distribution of the equity premium. This suggests that the model's predictions are more accurate when assuming a Student's t-distribution.

When comparing the time-series models we see for the GJR-GARCH models a slightly lower LS ($2.77 < 2.79$). On the other hand, we observe a lower CRPS score for the GARCH(T) model compared the GJR-GARCH(T) model ($1.94 < 2.24$).

Comparing the ML models with the time-series models we observe that on average both the LS and CRPS have lower values. For example, the LS and the CRPS of the XGBLSS(N) models are equal to 2.61 and 1.92 which are lower then 2.79 and 2.26 respectively for the GARCH(N) model. This indicates that the ML models outperform the time-series models in terms of forecasting performance.

Table 3: Average forecasting performance based on Scoring rules

Model	LS	CRPS
ML models		
DRF(N)	2.55	**1.86**
DRF(T)	**2.54**	1.87
NGB(N)	4.10	1.96
NGB(T)	2.68	1.94
XGBLSS(N)	2.61	1.92
XGBLSS(T)	2.62	1.87
time-series models		
GARCH(N)	2.79	2.26
GARCH(T)	2.79	1.94
GJR-GARCH(N)	2.77	2.25
GJR-GARCH(T)	2.77	2.24

Notes: The table shows the average Logarithmic Score (LS) and Continuous Ranked Probability Score (CRPS) over the OOS period (2002:01-2021:12). Lower scores are better and best scores are marked **bold**.

5.3 Final model sets

To asses if the probabilistic forecasts of the different models lead to statistically significant difference in forecasting performance, we implemented the MCS procedure. An overview of the final set of models obtained through different performance measures and confidence levels of $(1-\alpha)*100\%$ are presented in Table 4. For instance, when employing a 95% confidence level and using the LS as evaluation metric, the MCS procedure yields the following ranking for the final set of models: DRF(T), DRF(N) and XGBLSS(T). It is worth noting that there is no single superior model demonstrating better predictive ability. This outcome suggests that the data exhibits a low signal-to-noise ratio and provides limited information. Additionally, we observed that the CRPS measure resulted in a larger set of models, compared to the LS measure with the same α value. This finding suggests that CRPS is a more stringent metric, making it more challenging to reject the null hypothesis of equal predictive ability. Furthermore, our analysis revealed that the superior model sets primarily comprise ML models, with the GARCH(T) model being the only time-series model present in the sets.

Table 4: Sets of models obtained from the MCS procedure

	Final set models	
Scores	$\boldsymbol{\alpha} = 0.1$	$\boldsymbol{\alpha} = 0.05$
LS	{*DRF(T)*, *DRF(N)*, *XGBLSS(T)*}	{*DRF(T)*, *DRF(N)*,*XGBLSS(T)*}
CRPS	{*DRF(N)*,*DRF(T)*, *XGBLSS(T)* , *NGB(N)*, *XGBLSS(N)*}	{*DRF(N)*, *DRF(T)*, *XGBLSS(T)* , *NGB(N)*, *XGBLSS(N)*, *NGB(T)*, *GARCH(T)*}

Note. The superior model sets are given for a confidence level of $(1-\alpha)*100\%$. The test is implemented using the rugarch package in R.

5.4 Evaluation forecast-implied VaR

Table 5 shows the forecast-implied VaR violation ratios for various risk levels. A smaller value of α indicates a lower level of risk. The results show that for the DRF(N) model, the VaR violation ratio for $\alpha = 0.1$ is 7.9%. This indicates that OOS, the realized equity premium is smaller than the 90%-VaR estimate of the model 7.9% of the time. Generally, as we decrease α, we observe lower violation ratios due to more conservative VaR estimates and hence fewer breaches of the VaR threshold.

The DRF(N), DRF(T), and XGBLSS(N) models exhibit low violation ratios compared to the other models. For example, at a 95% confidence level, the violation ratios are 3.8%, 4.2%, and 3.8%, respectively. Conversely, the NGB(N) model displays a violation ratio of 23.8% for a confidence levels of 95%. From an investor's point of view, utilizing the DRF models holds greater appeal due to their ability to offer a more accurate estimate of potential losses and reduce the likelihood of unexpected substantial losses, as opposed to the NGB(N) model.

Table 5: VaR violation ratios for different levels of risk

Model	$\boldsymbol{\alpha} = 0.1$	$\boldsymbol{\alpha} = 0.05$	$\boldsymbol{\alpha} = 0.01$
ML models			
DRF(N)	7.9	3.8	0.0
DRF(T)	9.2	4.2	0.0
NGB(N)	28.7	23.8	15.8
NGB(T)	14.6	7.9	1.7
XGBLSS(N)	6.7	3.8	1.3
XGBLSS(T)	17.5	10.0	2.9
time-series models			
GARCH(N)	10.4	5.4	2.5
GARCH(T)	9.2	4.6	0.4
GJR-GARCH(N)	9.6	5.8	2.5
GJR-GARCH(T)	9.2	4.2	0.4

Note. The ratios are multiplied with a factor 100 such that the ratios are expressed in %. The violation ratio for a model is obtained by summing the number of VaR breaches and dividing by the total number of observations OOS (240). Furthermore, The VaR estimates are calculated for a confidence level of $(1 - \alpha) * 100\%$.

6 Conclusion & Discussion

This empirical study aims to make probabilistic forecasts of the equity premium using tree-based ML models and traditional time-series models. Specifically, we focus on the DRF, NGB, and XGBoostLSS models, which can provide probabilistic forecasts and capture complex relationships in the data. Furthermore, we assume a Normal distribution and Student's t-distribution for the equity premium.

Our central research question is: "How do tree-based ML models perform in providing density forecasts of the equity premium, and how do these compare to the density forecasts obtained from traditional time-series models?"

The main findings of this study reveal valuable insights into the performance of different forecasting models for the equity premium. We analyzed the average forecasting performance of the models using the LS and CRPS evaluation metrics. The study evaluates the performance

of various models and ranks them based on these scoring rules. The results show that the DRF model with Student's t-distribution (DRF(T)) achieves the best overall forecasting performance, with the lowest LS score among all models. Additionally, the DRF(N) model also performs well, obtaining the second-best LS score. Similarly, when considering CRPS, the DRF(N) model outperforms other ML models, including the benchmark GARCH(N) model. The Gradient Boosting models (XGBLSS(T) and NGB(T)) also show competitive performance in both LS and CRPS metrics. Overall, the ML models demonstrate superior forecasting capabilities compared to the time-series models.

Furthermore, the study analyzes the final model sets obtained through the MCS procedure. The MCS procedure aims to assess if the probabilistic forecasts of the different models lead to statistically significant differences in forecasting performance. Based on both proper scoring rules and confidence levels, the results reveal that the DRF(T) and DRF(N) models consistently appear in the final set of models, indicating their robust predictive ability. The XGBLSS(T) model also frequently appears in the final set, further emphasizing its competitive performance. However, it is noteworthy that there is no single superior model with significantly better predictive ability. This indicates that the data exhibits a low signal-to-noise ratio, leading to limited information for more precise model selection. These findings align with the conclusions drawn in earlier literature as presented by Goyal and Welch (2008). Despite this, the ML models, particularly the DRF(T) model, stand out as promising choices for forecasting equity premium distributions.

Lastly, the study examines the evaluation of forecast-implied VaR violation ratios for various risk levels. This analysis is economically relevant as it assesses how well the models estimate potential losses and manage risks. The results demonstrate that the DRF(N), DRF(T), and XGBLSS(N) models perform well, displaying lower VaR violation ratios compared to other models. This suggests that these models offer more accurate VaR estimates and reduce the likelihood of unexpected substantial losses. On the other hand, the NGB(N) model exhibits higher VaR violation ratios due to overly optimistic VaR predictions caused by small predicted scale parameters. These findings highlight the importance of precise scale parameter predictions for effective risk assessment.

The findings of this study have several practical implications for investors and financial practitioners. By using probabilistic tree-based ML models like DRF and XGBoostLSS, investors can obtain more accurate probabilistic forecasts of the equity premium, allowing them to make better-informed investment decisions.

While this study makes valuable contributions to the field of ML and forecasting in financial economics, there are several limitations and and interesting areas to consider in future research. Firstly, a random period of two years was chosen for the walk-forward OOS period at each fold. This period could be optimally selected to allow for hyperparameter tuning at each fold of the walk-forward procedure. Currently, when tuning the hyperparameters for the machine learning models, only one period is used. The values of the hyperparameters have a significant influence on determining the distributional scale parameter for the Gradient Boosting methods.

In future research, it would also be interesting to explore the application of skewed distributions. In this study, the use of the skewed Student's t-distribution for the DRF model was investigated, and yielded similar results to the DRF(T) model. It would be worthwhile to combine the skewed Student's t-distribution with the NGB and XGBLSS models. However, it should be noted that the current software package for the XGBLSS model does not support to add new distributions.

An additional potential direction for future research lies in exploring the variable importance. For tree-based ML models, variable importance plots can be generated to identify which variables play a crucial role in predicting the equity premium distribution. In this study, variable importance was not explored since utilizing variables with high importance can still lead to bad model forecasting performance.

References

Beckmann, J. and Schüssler, R. (2014). Forecasting equity premia using bayesian dynamic model averaging. *SSRN Electronic Journal*.

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

Campbell, J. Y. and Thompson, S. B. (2007). Predicting Excess Stock Returns Out of Sample: Can Anything Beat the Historical Average? *Review of Financial Studies*, 21(4):1509–1531.

Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, page 785–794. Association for Computing Machinery.

Diebold, F. X., Gunther, T. A., and Tay, A. S. (1998). Evaluating density forecasts with applications to financial risk management. *International Economic Review*, 39(4):863–883.

Duan, T., Anand, A., Ding, D. Y., Thai, K. K., Basu, S., Ng, A., and Alejandro, S. (2020). Ngboost: Natural gradient boosting for probabilistic prediction. *37th International Conference on Machine Learning*, pages 2690–2700.

Fama, E. F. and French, K. R. (1988). Dividend yields and expected stock returns. *Journal of Financial Economics*, 22(1):3–25.

Fama, E. F. and French, K. R. (1989). Business conditions and expected returns on stocks and bonds. *Journal of Financial Economics*, 25(1):23–49.

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

Glosten, L. R., Jagannathan, R., and 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.

Gneiting, T., Balabdaoui, F., and Raftery, A. E. (2007). Probabilistic forecasts, calibration and sharpness. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 69(2):243–268.

Gneiting, T. and Katzfuss, M. (2014). Probabilistic forecasting. *Annual Review of Statistics and Its Application*, pages 125–151.

Good, I. J. (1952). Rational decisions. *Journal of the Royal Statistical Society. Series B (Methodological)*, 14(1):107–114.

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

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

Hansen, P. R., Lunde, A., and Nason, J. M. (2011). Model confidence sets for forecasting models. *Econometrica*, 79(2):453–497.

Hoogerheide, L. F., Ardia, D., and Corré, N. (2012). Density prediction of stock index returns using garch models: Frequentist or bayesian estimation? *Economics Letters*, pages 322–325.

Hothorn, T., Hornik, K., van de Wiel, M. A., and Zeileis, A. (2006). A lego system for conditional inference. *American Statistician*, 60(3):257–263.

Kelly, B. and Pruitt, S. (2015). The three-pass regression filter: A new approach to forecasting using many predictors. *Journal of Econometrics*, 186(2):294–316.

Koenker, R., Leorato, S., and Peracchi, F. (2013). Distributional vs. Quantile Regression. *SSRN Electronic Journal*.

Kwak, S. and Kim, J. (2017). Statistical data preparation: Management of missing values and outliers. *Korean Journal of Anesthesiology*, 70:407.

Massey, F. J. (1951). The kolmogorov-smirnov test for goodness of fit. *Journal of the American Statistical Association*, 46(253):68–78.

Mehra, R. and Prescott, E. C. (1985). The equity premium: A puzzle. *Journal of Monetary Economics*, 15(2):145–161.

Meligkotsidou, L., Panopoulou, E., Vrontos, I., and Vrontos, S. (2012). A quantile regression approach to equity premium prediction. *Journal of Forecasting*, 33.

März, A. (2019). Xgboostlss–an extension of xgboost to probabilistic forecasting. *arXiv*.

Neely, C. J., Rapach, D. E., Tu, J., and Zhou, G. (2014). Forecasting the equity risk premium: The role of technical indicators. *Management Science*, 60(7):1772–1791.

Rapach, D. and Zhou, G. (2013). Forecasting stock returns. *Handbook of Economic Forecasting*, 2:327–383.

Rapach, D. E. and Zhou, G. (2020). Time-series and cross-sectional stock return forecasting: New machine learning methods. *Machine learning for asset management: New developments and financial applications*, pages 1–33.

Rigby, R. A. and Stasinopoulos, D. M. (2005). Generalized additive models for location, scale and shape. *Journal of the Royal Statistical Society*, 54(3):507–554.

Schlosser, L., Hothorn, T., Stauffer, R., and Zeileis, A. (2019). Distributional regression forests for probabilistic precipitation forecasting in complex terrain. *Annals of Applied Statistics*.

Sigrist, F. (2021). Gradient and newton boosting for classification and regression. *Expert Systems with Applications*, 167:114080.

Wolff, D. and Neugebauer, U. (2019). Tree-based machine learning approaches for equity market predictions. *Journal of Asset Management*, 20(4):273–288.

Appendix A Descriptive statistics data

Table 6: Descriptive statistics for the macroeconomic variables.

Variable	**Mean**	**Std**	**Min**	**25%**	**75%**	**Max**
DP^*	-3.62	0.40	-4.52	-3.95	-3.35	-2.75
DY^*	-3.62	0.40	-4.53	-3.94	-3.35	-2.75
EP^*	-2.87	0.43	-4.84	-3.11	-2.68	-1.90
DE^*	-0.75	0.30	-1.24	-0.92	-0.60	1.38
BM^*	0.48	0.26	0.12	0.28	0.64	1.21
$NTIS$	0.01	0.02	-0.06	0.00	0.02	0.05
TBL^{**}	4.40	3.20	0.01	1.89	6.08	16.30
LTY^{**}	6.15	2.85	0.62	4.15	7.95	14.82
LTR^*	0.61	2.91	-11.24	-1.05	2.28	15.23
TMS^{**}	1.75	1.43	-3.65	0.69	2.90	4.55
DFY^{**}	1.01	0.44	0.32	0.72	1.19	3.38
DFR^*	0.02	1.50	-9.76	-0.56	0.60	7.37
$INFL^*$	0.30	0.36	-1.92	0.07	0.51	1.81
$RVOL$	0.14	0.05	0.05	0.10	0.18	0.32
EPL^*	0.58	4.27	-22.03	-1.88	3.20	16.00

Note. * (**) indicates that the variable is measured in % (annual %). These are the descriptive statistics before the standardization of the variables is applied.

Table 7: Frequencies for the technical indicators (in %).

Variable	Value 1	Value 0
$MA_{1,9}$	70.03	29.97
$MA_{1,12}$	72.18	27.82
$MA_{2,9}$	70.03	29.97
$MA_{2,12}$	70.03	29.97
$MA_{3,9}$	70.03	29.97
$MA_{3,12}$	70.03	29.97
$MOM_{1,9}$	72.04	27.96
$MOM_{1,12}$	73.92	26.08
$VOL_{1,9}$	70.43	29.57
$VOL_{1,12}$	72.72	27.28
$VOL_{2,9}$	69.09	30.91
$VOL_{2,12}$	72.31	27.69
$VOL_{3,9}$	71.24	28.76
$VOL_{3,12}$	72.04	27.96

Notes: The table shows the frequencies of the Moving Average (MA), Momentum (MOM) and Volume (VOL) based variables. These frequencies are based on the sample 1960:01-2021:12. The variables are constructed as in Neely et al. (2014).

Appendix B Density forecasts Student's t-distribution

Density forecasts for the location-scale t-distribution are given by:

$$f(y_t|\mathcal{I}_{t-1}) = \frac{\Gamma\left(\frac{\hat{\nu}+1}{2}\right)}{\Gamma\left(\frac{v_{t|t-1}}{2}\right)\sqrt{\pi\hat{\nu}\hat{\sigma}^2_{t|t-1}}}\left(1+\frac{(y_t-\hat{\mu}_{t|t-1})^2}{\hat{\nu}\hat{\sigma}^2_{t|t-1}}\right)^{-\frac{\hat{\nu}+1}{2}} \tag{31}$$

where $\hat{\mu}_{t|t-1}$, $\hat{\sigma}^2_{t|t-1}$ and $\hat{\nu}$ are the location, scale and shape parameters. A small value of $\hat{\nu}$ gives a distribution with heavy tails. The conditional mean and variance are respectively denoted as $\hat{\mu}_{t|t-1}$ and $\frac{\hat{\nu}}{\hat{\nu}-2}\hat{\sigma}^2_{t|t-1}$ for $\hat{\nu} > 2$.

Appendix C Hyperparameter tuning tables

Table 8: Hyperparameter grids for DRF model

	number of estimators	**column sample fraction**	**minimal split**
DRF(N)	[100, 150, 250, **500**]	[0.10, **0.33**, 0.5]	[**10**, 20, 50, 100]
DRF(T)	[100, 150, 250, **500**]	[0.10, **0.33**, 0.5]	[10, **20**, 50, 100]

Notes: This table shows the parameter grids for the DRF models in the OOS period 1990:01-2001:12. The optimal values for the hyperparameters are marked **bold**.

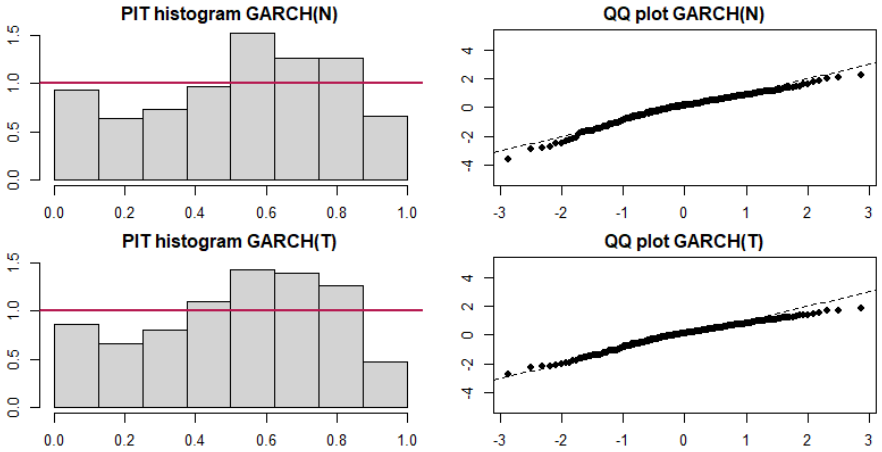
Table 9: Hyperparameter grids for Gradient Boosting models

	learning rate	**mini batch fraction**	**max depth**
NGB(N)	Unif(0.05,1) : **0.14**	[0.5, **0.75**, 1]	[**2**, 3, 4]
NGB(T)	Unif(0.05,1) : **0.13**	[0.5, 0.75, **1**]	[**2**, 3, 4]
XGBLSS(N)	Unif(0.05,1) : **0.95**	Unif(0.5,1) : **0.83**	[**1**,...,10]
XGBLSS(T)	Unif(0.05,1) : **0.21**	Unif(0.5,1) : **0.78**	[1,..., **9**, 10]

Notes: This table shows the parameter grids for the NGB and XGBLSS models in the OOS period 1990:01-2001:12. The optimal values for the hyperparameters are marked **bold**.

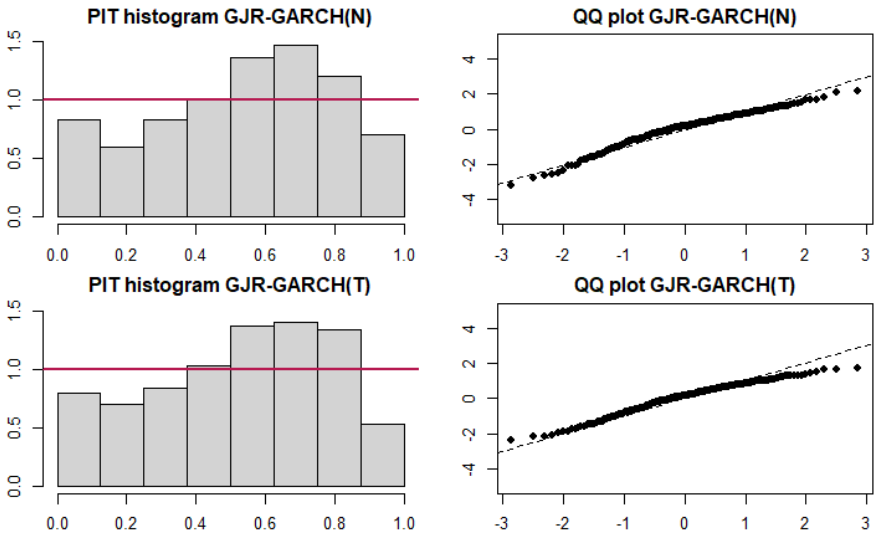
Appendix D PIT histograms and QQ plots time-series models

Figure 7: PIT histograms and QQ plots for the AR(1)-GARCH(1,1) model.



Notes: GARCH(N): KS test-statistic is equal to 0.097 with p-value 0.02 such that the H_0 of uniformity is rejected. GARCH(T): KS test-statistic is equal to 0.090 with p-value 0.04 such that the H_0 of uniformity is rejected at a significance level of 5%.

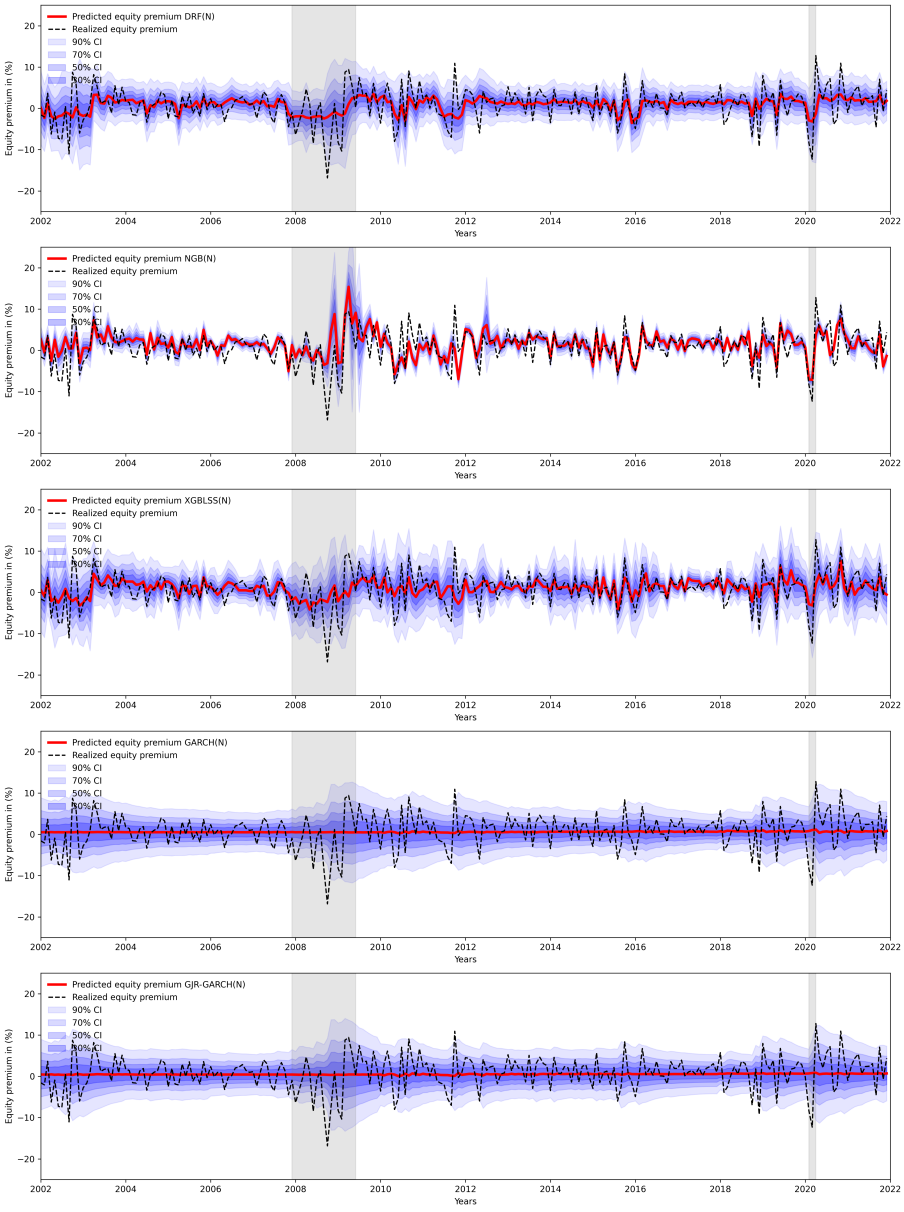
Figure 8: PIT histograms and QQ plots for the AR(1)-GJR-GARCH(1,1) model.



Notes: GARCH(N): KS test-statistic is equal to 0.11 with p-value 0.00 such that the H_0 of uniformity is rejected. GARCH(T): KS test-statistic is equal to 0.10 with p-value 0.02 such that the H_0 of uniformity is rejected at a significance level of 5%.

Appendix E Conditional mean over time for Normal distribution

Figure 9: Predicted conditional mean of the equity premium assuming a Normal distribution with corresponding confidence intervals



Notes: Each panel in the figure shows the predicted $\hat{\mu}_{t|t-1}$ (red line) for a model assuming a Normal distribution. The dashed line corresponds to the realized equity premium. The bands correspond to 90%, 70%, 50% and 30% CIs and are obtained by $\hat{\mu}_{t|t-1} \pm \Phi^{-1}(\alpha/2)\ \hat{\sigma}_{t|t-1}$. Furthermore, crisis periods are marked grey.