Erasmus University Rotterdam

Erasmus School Of Economics

A Machine Learning Approach to National Stock Index Volatility Prediction

Author:
Dominique de Nerée tot
Babberich
(568620)

Supervisor:
Dr. A. Pick
Second Assessor:
Dr. A. Naghi

A thesis submitted for the degree of

Msc Econometrics

May 20, 2022

The content of this thesis is the sole responsibility of the author and does not reflect the view of the supervisor, second assessor, Erasmus School of Economics or Erasmus University.

Abstract

This research proposes three machine learning methods as well as a novel hybrid method to carry out a comparative analysis on their ability to accurately model and forecast the volatility of the Amsterdam Exchange Index (AEX). Furthermore the machine learning methods are assessed in their ability to effectively select and use information in stock data and macroeconomic variables in order to increase forecast performances. The proposed methods are support vector regression, random forest, gradient boosted trees and the novel EGARCH-SVR model, since they possess promising characteristics to capture the complex structure of volatility. To evaluate the hypothesis that the proposed learning methods can accurately predict the AEX volatility they are benchmarked against traditional statistical time series models; the Generalized Autoregressive Conditional Heteroskedasticity (GARCH), exponential GARCH (EGARCH) and Glosten-Jaganathan-Runkle (GJR)-GARCH model, all with two distributional assumptions. The predictive performances are compared with two goodness-of-fit measures and significant differences in performance as well as robustness over different forecasting horizons are evaluated. Additional data includes variables on the largest components of the AEX as well as additional (macroeconomic) variables as possible drivers of volatility. The machine learning methods can significantly increase predictive performance compared to the traditional volatility models, especially in times of higher average volatility. The hybrid method disappoints and is highly affected by rapid fluctuation of the market. Overall, applying learning techniques to the out-of-sample forecasting of volatility show great potential in both prediction and extracting important information out of additional data.

Contents

1 Introduction

The goal of this research is to evaluate if three machine learning methods as well as a suggested novel hybrid method can accurately model and forecast the volatility of the Amsterdam Exchange Index (AEX) and can compete with or even outperform traditional volatility models. Furthermore, the machine learning methods are evaluated in their ability to effectively select and apply information in stock data of the largest components of the AEX as well as macroeconomic variables indicating the state of the Dutch and European economy. The proposed machine learning methods are support vector regression (SVR), random forest (RF) and gradient boosted trees (GB). All aforementioned methods possess qualities that are expected to be able to capture the nonlinearities and complex structure of financial return series data and have the ability to effectively select valuable information from high dimensional data to increase predictive accuracy. Additionally, a novel hybrid model is considered: the exponential generalized heteroskedasticity support vector regression (EGARCH-SVR) model. Combining a parametric volatility model with a nonparametric machine learning model is expected to lead to an increase in predictive performance by gaining benefits from both.

In order to evaluate the performances of these methods besides addressing only their relative predictive power, three traditional volatility models are selected to serve as a benchmark and substantiate the performance of the learning methods. These are three models of the generalized autoregressive conditional heteroskedasticity (GARCH) class: the GARCH model, the exponential GARCH (EGARCH) model and the Glosten-Jaganathan-Runkle (GJR)-GARCH model that are widely applied and often proved well performing in volatility forecasting (Hansen and Lunde, 2005; Monfared and Enke, 2014; McAleer, 2014).

These parametric models require a distributional assumption on the returns which is frequently assumed to be normal, despite the highly nonlinear structure and excess skewness and kurtosis often exhibited by a financial return series. In order to get a comprehensive representation of benchmark models this research is extended by including the student t distributional assumption. Furthermore, since the predictive performances of the applied methods can be influenced by selecting a particular forecasting window, the AEX volatility is predicted for multiple horizons and the robustness of the performances is assessed. The performance is compared with two well known performance measures and further justification of statistical differences in performance is based on the pairwise Diebold-Mariano test. Finally, through the variable importance measures of the tree based machine learning methods the informativeness of additional data on the stocks, exchange rates and other macroeconomic indicators is evaluated.

First of all, out of the benchmark models the EGARCH model performs the best for nearly all forecasting horizons. Interestingly, assuming a student t distribution of the returns does not lead to an increase in predictive performance, on the contrary; in almost all cases performance is slightly worse based on this assumption. The learning models often outperform the benchmark models, especially in the longer forecasting horizons containing a higher average volatility of the AEX. Amongst the learning methods the support vector regression and gradient boosting algorithms perform the best, whilst the performance of the hybrid model disappoints. Furthermore, it is found that the best performing machine learning models are more robust to the forecast horizon compared to the other considered models. Based on the Diebold-Mariano test superior performance of the machine learning methods can be concluded for some, however not for all forecasting horizons and not for both performance measures.

Finally, based on the variable importance measures of the tree based methods it is shown that additional data of stocks, exchange rates and some macroeconomic variables include some information regarding the future AEX volatility. However, it should be noted that volatility prediction remains a challenging task and especially in high volatile market periods all considered models have a hard time estimating the spiked structure of the series. Additionally, the need of using a proxy for the AEX volatility makes prediction complex and slightly flawed.

The volatility of a stock or index is an indicator of the fluctuation of the price around its mean and thus an indicator of the uncertainty of its profitability. Therefore, this variable takes up an important role in the estimation of financial risk models and many different types of models have been developed and used in the past decades (Dash et al., 2015). Amidst the statistical time series models the most widely applied are models of the autoregressive integrated moving average (ARIMA) and the GARCH class, where the latter is proven more suitable to capture the characteristics of persistence in volatility (Engle, 1982; Bollerslev, 1986; Hansen and Lunde, 2005).

In spite of their extensive use, the GARCH model in its simplest form is a symmetrical model and experiences some drawbacks in measuring persistence and inadequately responds to different shocks. In order to overcome these limitations multiple extensions are proposed, amongst them the EGARCH model that is designed to fit the difference in behavior of volatility within different market trends (Nelson, 1991). Another way to model the asymmetry in volatility is by employing the asymmetric power ARCH (APARCH) model or the GJR-GARCH model, that both incorporate a leverage effect parameter (Ding et al., 1993; Glosten et al., 1993). Numerous alternatives have been proposed in previous literature, however in this research the GARCH, EGARCH and GJR-GARCH are selected to serve as a benchmark based on often proven superior performance (Hansen and Lunde, 2005; Monfared and Enke, 2014; McAleer, 2014).

Limitations of parametric time series models of the GARCH class include the need of prespecification of the structure of the data. In the past the normal distribution is often imposed on the financial returns, however such a series usually exhibits a significantly negative skewness coefficient and excess kurtosis which leads to a non-symmetric, peaked and fat tailed distribution (Tsay, 2002). To fit these characteristics other distributional assumptions have been introduced such as the generalized error distribution or the student t distribution (Gong et al., 2019). In this research besides the normal distribution the student t distribution is adopted in order to potentially increase predictive performance.

Apart from the disadvantage of prespecification of the data structure, forecasting volatility using GARCH models can lead to unsatisfactory results as they are sometimes incapable to describe its complex and nonlinear structure. In order to increase prediction accuracy more advanced techniques are necessary when working with such complex patterns (Cheng and Wei, 2009). Thus, the interest in applying nonparametric and nonlinear models to the estimation of the volatility of a stock or index has grown tremendously.

The main focus of machine learning methods in financial market forecasting has been in the field of artificial neural networks (ANN). In the research of Donaldson and Kamstra (1997), they prove that applied ANNs can outperform the traditional GARCH class models in out-of-sample forecasting of the volatility for four large markets. Similarly, Miranda and Burgess (1997) apply standalone neural networks to the Ibex35 index to forecast the out-of-sample volatility and find that the neural networks indeed surpass traditional linear models by their flexibility and ability to deal with the complex structure of the financial time series. Hamid and Iqbal (2004) use neural networks in forecasting the S&P500 Index volatility, where they conclude that the neural networks provide better forecasts compared to the implied volatility forecasts, however not compared to the realized volatility forecasts.

Despite the appointed advantages of neural networks in financial forecasting the algorithm is extremely sensitive to the tuning of its parameters and tends to overfit in highly noisy and non-stationary data sets (Kara et al., 2011). In such situations support vector machines may be preferred, a powerful algorithm that adjusts its parameters relying on the structural risk minimisation principle which increases the ability of generalization (Boser et al., 1992; Cao and Tay, 2001). Cao and Tay (2003) prove in a simulation study that a support vector machine with adaptive parameters can outperform a back propagated neural network in the volatility forecasting of multiple bonds and futures. Furthermore, Radovic and Stankovic (2015) use the algorithm to forecast the volatility of the Belex15 index and compute its value at risk, where they find that the support vector machines outperform both feed forward neural network as well as markov regime switching models. Furthermore, Yuan (2013) applies support vector machines on the prediction of financial time series movement direction and states that the support vector algorithm might be able to find the global optimum where the neural network produces a local optimum solution.

Thus, the use of regression support vector machines in financial forecasting seem like a promising tool, since they are able to effectively select information from amongst others inputs of lagged returns and show better results than traditional methods and other machine learning methods such as neural networks (Cavalcante et al., 2016). Given these directions of previous research comparing neural networks and support vector machines in financial forecasting, in this research support vector machines are preferred over neural networks and applied to the AEX volatility forecasting.

Besides neural networks and support vector machines, tree based learning methods are applied to forecast volatility, such as random forest and boosted trees. The application of random forest is wide spread and the algorithm has been successfully applied to the forecasting of the euro to dollar exchange rate by Theofilatos et al. (2012), where it showed better predictive performances compared to multiple other learning techniques such as neural networks. Moreover, Luong and Dokuchaev (2018) show in their research that due to the great features the algorithm possesses, random forest can reduce the forecasting error when applied to nonlinear structured stock index data by better capturing volatility persistence or clustering throughout different time spans.

Boosted trees are less frequently applied to forecast volatility, since boosting methods are often more effected by highly noisy datasets such as financial returns (Dietterich, 2000). However Mittnik et al. (2015) use gradient boosted trees in their research to identify macroeconomic drivers of volatility and conclude that besides the usefulness of the boosted tree to identify variable importances and their ability to extract information out of additional input variables, they also produce better out-of-sample volatility forecasts on multiple horizons compared to the EGARCH model. Moreover, Christensen et al. (2021) assesses the capability of gradient boosted trees as well as other learning techniques in extracting information out of not only lagged returns but also additional macroeconomic indicators. They find that the machine learning algorithms can indeed obtain additional valuable information and produce better volatility forecasts compared to a heterogeneous autoregressive model. As both tree based methods appear to be useful in volatility forecasting they are both included in this research.

Besides the application of all aforementioned standalone learning methods efforts have been made to combine the advantages of the traditional time series models and the learning techniques to form hybrid models to increase predictive performance. Much of the research related to volatility forecasting is in the direction of combining neural networks with GARCH class models. Roh (2007) for instance suggests a combination of a neural network with a GARCH and EGARCH model and demonstrates the utility of the hybrid model in volatility forecasting, where it proves to be superior. The same is proven by Lu et al. (2016), who construct multiple hybrid models to estimate the volatility of the Chinese energy market and conclude that the hybrid ANN-EGARCH model performs the best. Other previous research has shown that combining asymmetric types of GARCH model with the more intelligent neural network structures increases the closeness of predictions to the actual volatility of oil prices (Kristjanpoller and Minutolo, 2016).

Another manner of combining methods is done by Monfared and Enke (2014), who apply an adaptive neural network in order to predict the errors made by the GJR-GARCH model in forecasting the volatility of multiple indices, whereafter a combination of the two provides more accurate out-of-sample volatility forecasts. Instead of neural networks, support vector machines have also been used in combination with GARCH models, where often the maximum likelihood method used to estimate the GARCH parameters is replaced by estimation with support vector machines, leading to better out-of-sample volatility forecasts than the standalone GARCH type models (Bezerra and Albuquerque, 2017; Peng et al., 2018).

However, according to the recent work of Sun and Yu (2020) the predictive performance can be increased even more when training the support vector machine on the prediction errors made by the GARCH class model. In their research the GARCH and GJR-GARCH models both with normal as well as student t distributional assumptions are combined with support vector regression to obtain a two stage volatility forecasting method and they find that the hybrid model indeed improves volatility predictions. Following this promising direction, in this research the EGARCH model is combined with the support vector regression in a similar manner to construct a novel hybrid EGARCH-SVR model to forecast the AEX volatility.

Hence, in previous literature often one of the following directions is taken; either a single machine learning method is applied to volatility forecasting and compared to a traditional volatility method, or multiple learning methods are applied and compared in relative performance and/ or compared to a single benchmark model. This research contributes to existing literature as multiple learning methods are applied and compared to a comprehensive set of benchmark models of the GARCH class to substantiate performances. In this way a better and more complete view on the power of machine learning methods in volatility forecasting is achieved. Moreover, this research introduces the novel EGARCH-SVR model extending the work of Sun and Yu (2020). Furthermore, in the discussed literature often only the lagged (squared) return or another proxy is used as input for the machine learning methods, whilst only few use additional macroeconomic variables and discuss their informativeness as is done in this research.

The remainder of this research is structured as follows. First, a methodology section is presented which consists of an introduction to volatility, followed by an introduction of statistical time series methods; the GARCH, EGARCH and GJR-GARCH models, the machine learning methods; support vector machines, random forest and gradient boosted trees and finally the hybrid EGARCH-SVR model. Furthermore in this section the metrics and tests used to evaluate the data and the performance of the models is presented. In the next section the data used for this research is described. The movement of the response variable is analyzed and the additional explanatory variables used for the machine learning methods are discussed. Next, the results are presented in tables and figures and discussed. The most technical assumptions and long lists and tables are presented in the Appendix. Finally, a conclusion is presented which covers a summary of this research, the approach, results and limitations and some recommendations for further research.

2 Methodology

This chapter first describes the basic definitions of volatility, followed by a description of the statistical methods used to serve as a benchmark, the machine learning methods and finally the hybrid method. It concludes with the tests used and the evaluation measurements implemented to validate and compare the results.

2.1 Volatility

The volatility of the AEX is the level of variation of the price over time and thus an important indication of the fluctuation of the Dutch stock market. A complication in forecasting volatility is that it is a latent variable, i.e. it can not be observed directly from the AEX data, which makes estimation and evaluation of the forecasts more complex (Andersen and Bollerslev, 1998). To approximate volatility different proxies can been used, however any volatility proxy is a more or less flawed estimator and will not lead to the same outcome in terms of model performances in every case (Patton, 2006).

The most commonly used volatility proxy is the squared close-to-close return (Vilder and Visser, 2007). Another widely used proxy for volatility is realized volatility, a measure based on the intraday data which is calculated as high minus low (Barndorff-Nielsen and Shephard, 2002; Andersen et al., 2003). In this research the squared daily close-to-close return is selected as proxy, where the logarithm of the daily returns is taken rather than the raw returns. This conveniently leads to the formula for the daily log return series r_t as in eq. (1) with p_t the closing price of the regarding index at time t. The squared log return series is simply r_t^2.

$$r_t = \log(p_t) - \log(p_{t-1}) \tag{1}$$

Now assume that the daily close-to-close log return series follow a stationary process as described in eq. (2), with μ the mean of r_t and the heteroskedastic error term ϵ_t, which has zero mean and time varying variance σ_t^2 conditional on I_{t-1}, the information set at time $t-1$. If the mean is assumed to be zero, the squared daily log return series is an unbiased estimator for the actual conditional variance. In this research, a constant mean is assumed and subtracted from the return series to obtain the residual vector.

$$r_t = \mu + \epsilon_t \tag{2}$$

$$\sigma_t^2 = \mathbf{E}[\epsilon_t^2 | I_{t-1}] \tag{3}$$

Both the statistical methods as well as the machine learning methods will be assessed in their performance of the out-of-sample forecasting of the volatility of the AEX. In the next sections these methods will be described separately. Note that hereafter the daily close-to-close log return series is sometimes referred to as the log return series or return series for simplicity.

2.2 Statistical Methods

In the following paragraphs first an introduction to the ARCH framework will be presented whereafter the three statistical models of the GARCH class serving as benchmarks for the machine learning methods are described.

2.2.1 ARCH Framework

Assume a return series that follow the stationary process as in eq. (2). In order to define the conditional variance the underlying processes of the shocks (ϵ_t) to the returns need to be specified (McAleer, 2014). A well known method that allows the conditional variance to vary over time is the autoregressive conditional heteroskedasticity (ARCH(p)) model by Engle (1982). The conditional variance is specified as a constant unconditional variance and a function of past errors and is specified as follows:

$$r_t = \mu + \epsilon_t, \ \epsilon_t = z_t \sigma_t \tag{4}$$

$$\sigma_t^2 = \omega + \sum_{i=1}^{p} \alpha_i \epsilon_{t-i}^2 \tag{5}$$

Where r_t the return series with its mean μ, ϵ_t the error term and σ_t^2 the conditional variance. The error term ϵ_t is conveniently described as z_t, which is assumed to be an I.I.D. process and $z_t \sim \mathcal{N}(0, 1)$, multiplied with σ_t. To impose that the conditional variance is always nonnegative the parametric constraints are $\omega > 0$ and $\alpha_i \geq 0$. The ARCH model is a symmetric model, which implies that a positive shock has the same influence on the conditional variance as a negative shock of the same size. Past research has proven that in practically all cases a more sophisticated form outperforms the simple ARCH model and therefore the ARCH model in its simplest form is not evaluated in this research (Hansen and Lunde, 2005).

2.2.2 GARCH

An extended form of the ARCH(p) model is the generalized autoregressive conditional heteroskedasticity (GARCH(p,q)) model by Bollerslev (1986), where the return series follows the same process as in eq. (4) and the conditional variance is described as:

$$\sigma_t^2 = \omega + \sum_{i=1}^{q} \alpha_i \epsilon_{t-i}^2 + \sum_{j=1}^{p} \beta_j \sigma_{t-j}^2 \tag{6}$$

The parametric restrictions for the GARCH model are: $\omega > 0, \alpha_i \geq 0 \ \forall i, \ \beta_j \geq 0 \ \forall j$, to guarantee that σ_t^2 is always positive and $\alpha_i + \beta_j < 1 \ \forall i, j$ to guarantee a covariance stationary process. When complying these conditions, the long term average of the unconditional variance of a GARCH(1,1) model is well defined as:

$$\sigma^2 = \frac{\omega}{1 - \alpha - \beta} \tag{7}$$

The GARCH model has a symmetric form similar to the ARCH model and therefore does not allow for different responses to positive or negative shocks, which is considered as a downside of the GARCH model.

2.2.3 EGARCH

The exponential generalized autoregressive conditional heteroskedasticity (EGARCH(p,q)) model is an extended form of the GARCH model that does allow for asymmetry and also for leverage effect (Nelson, 1991). The returns again follow the same process as in eq. (4) and the model specifies the conditional variance as:

$$\ln{(\sigma_t^2)} = \omega + \sum_{i=1}^{q} \left(\alpha_i \left| \frac{\epsilon_{t-i}}{\sigma_{t-i}} \right| + \gamma_i \frac{\epsilon_{t-i}}{\sigma_{t-i}} \right) + \sum_{j=1}^{p} \beta_j \ln{(\sigma_{t-j}^2)} \tag{8}$$

This exponential model utilizes the logarithms of the conditional variances σ_{t-j}^2, which implies that the conditional variances are always positive and therefore no restrictions on the parameters signs are necessary. However $|\beta_j| < 1 \ \forall j$ is necessary as stability condition and furthermore the EGARCH model only allows for asymmetry if $|\gamma_i| \neq 0$ and for leverage effect if $\gamma_i < 0$, and $\gamma_i < \alpha_i < -\gamma_i \ \forall i$ (McAleer and Hafner, 2014).

2.2.4 GJR-GARCH

Another asymmetric model of the GARCH class is the Glosten-Jaganathan-Runkle GARCH (GJR-GARCH(p,q)) model specified in eq. (9) with again the return process as in eq. (4), which allows for a different effect of a positive or a negative movement of the returns on the conditional variance through an indicator variable (Glosten et al., 1993).

$$\sigma_t^2 = \omega + \sum_{i=1}^{q} (\alpha_i + \gamma_i I_{t-i})\epsilon_{t-i}^2 + \sum_{i=1}^{p} \beta_j \sigma_{t-j}^2 \tag{9}$$

$$I_{t-i} = \begin{cases} 0, & \text{if } \epsilon_{t-i} \geq 0 \\ 1, & \text{if } \epsilon_{t-i} < 0 \end{cases}$$

The parameter restrictions of this model are $\omega > 0$, $\frac{\alpha_i+\gamma_i}{2} > 0 \ \forall i$, $\beta_j \geq 0 \ \forall j$, to guarantee that σ_t^2 is always positive. The indicator variable allows for asymmetry in the model and in order for leverage effect to exist in the GJR model $\gamma_i > 0$, which implies that a larger conditional variance is caused by a negative shock. If these restrictions in combination with $\frac{\alpha_i+\gamma_i}{2} + \beta < 1$ are respected, the long term average of unconditional variance of a GJR-GARCH(1,1) model is well defined as:

$$\sigma^2 = \frac{\omega}{1 - \frac{\alpha+\gamma}{2} - \beta} \tag{10}$$

2.2.5 Assumed Distributions and Estimation

The assumption in eq. (4) that z_t follows a gaussian process does not directly imply that the distribution of the returns follows a gaussian process. It is demonstrated in previous research that this is often not the case, due to the presence of skewness and excess kurtosis of the residual distribution (Tsay, 2002). When assuming the distribution of the returns to be normal there is a serious risk of over- or underestimating the actual returns and conditional variances (Bollerslev, 1986). A (partial) solution to this problem is the use of other underlying distributions such as the student t distribution that allows for fatter tails (Nugroho et al., 2021).

As showed in previous research, GARCH models with lags $p = 1$ and $q = 1$ in the variance equation often outperform specifications with more distant lags in forecasting volatility (Bollerslev et al., 1992; Hansen and Lunde, 2005). Therefore in this research all models are computed as such: GARCH(1,1), EGARCH(1,1) and GJR-GARCH(1,1). To estimate the described GARCH type models maximum likelihood method is used where the likelihood function and thus also the log likelihood function depend on the assumed distribution of the returns. For the complete density functions of the two assumed distributions please refer to Appendix A.1. The log likelihood functions are optimized using a numerical procedure in Python to estimate the parameters per each model, whereafter the fitted models are applied to predict the one-day-ahead out-of-sample conditional variances, following the forecasting scheme described in paragraph 2.5.1.

2.3 Machine Learning Methods

The following paragraphs are structured as follows: first a short introduction on the machine learning framework is presented, followed by a description of the three selected machine learning methods applied to forecast the volatility of the AEX.

2.3.1 Machine Learning Framework

Machine learning can be broadly viewed as the process of computer algorithms that can interact and learn from their environment with the goal of making better predictions through structural adaption. These learning methods are often applied in cases where building accurate predicting models is difficult and highly useful in cases of high dimensional data (Alpaydin, 2020). A great power of these machine learning methods is their ability to select only the descriptive variables from high dimensional data and model complex patterns accurately. For the purpose of examining the predictive performance of machine learning methods in financial time series a selection of promising methods is selected to estimate the AEX volatility.

Contrary to the GARCH type models that try to estimate volatility through the conditional variance σ_t^2, these machine learning methods try to predict the AEX volatility through the behaviour of ϵ_t in eq. (2) in squared form by predicting the demeaned squared log return series.

2.3.2 Support Vector Machines

Support vector machine (SVM) is a supervised learning method developed by Boser et al. (1992) for classification problems. The aim of the SVM algorithm is to construct a hyperplane that has the maximum margin between data points of different classes, in order to classify them. These separating hyperplanes can be viewed as decision boundaries that define the classification of the data points where linear equations are used to construct the support vectors of the model. The algorithm was later extended for the use on regression problems by Vapnik et al. (1997) as support vector regression (SVR) and can be applied to time series data.

Support vector regression holds some noteworthy advantages over various other machine learning methods that often adopt the so called empirical risk minimisation principle (minimising the squared error), as the SVR objective function is to minimize a loss function that is not affected when the difference between the prediction and the actual value is less then a certain predefined level η. The error term is controlled in the models constraints, where the quadratic error function makes place for the η-insensitive error function introduced by

Vapnik et al. (1997) which allows to set the error to a certain margin[1]. The error function now gives a weight of zero when the absolute difference between the prediction and the target is smaller than this η. Thus, this maximum error η is conveniently functional as tuning parameter to gain accuracy of the model.

Additionally, the SVR is expected to achieve relatively enormous gains compared to traditional models in capturing the nonlinear dynamics present in financial time series data by introducing kernel functions (Qu and Zhang, 2016). These kernel functions provide a method to transform the data in order to change a nonlinear decision plane to a linear equation within a higher dimension. Another advantage of applying SVR in volatility forecasting compared to the traditional methods is their ability to efficiently select information from additional data to increase predictive performance.

The SVR regularized error minimization problem for the AEX volatility is as in eq. (11), with $\hat{\epsilon}_t^2$ the predicted value of the AEX volatility at time t and ϵ_t^2 the true value at that time. The parameter w is the direction and C is the (positive) regularization parameter in order to balance complexity and the error made in training the model which assists in the prevention of overfitting.

$$\begin{aligned} C\sum_{t=1}^{T} E_\eta(\hat{\epsilon}_t^2 - \epsilon_t^2) + \frac{1}{2}||w||^2 \\ E_\eta(\hat{\epsilon}_t^2 - \epsilon_t^2) = \begin{cases} 0, & \text{if } |\hat{\epsilon}_t^2 - \epsilon_t^2| < \eta \\ |\hat{\epsilon}_t^2 - \epsilon_t^2| - \eta, & \text{otherwise} \end{cases} \end{aligned} \tag{11}$$

Often in real-world problems there is no function that satisfies all constraints imposed above. In order to be able to allow some data points to fall outside the margins two slack variables are introduced; $\xi_t \geq 0$ and $\xi_t^* \geq 0$, where if $\xi_t > 0$ ($\xi_t^* > 0$) for the corresponding data point holds that $\epsilon_t^2 > \hat{\epsilon}_t^2 + \eta$ ($\epsilon_t^2 < \hat{\epsilon}_t^2 - \eta$). This leads to the corresponding error function to be minimized and its constraints:

$$\begin{aligned} C\sum_{t=1}^{T} (\xi_t + \xi_t^*) + \frac{1}{2}||w||^2 \\ \text{s.t. } \epsilon_t^2 \leq \hat{\epsilon}_t^2 + \eta + \xi_t \\ \epsilon_t^2 \geq \hat{\epsilon}_t^2 - \eta - \xi_t^* \\ \xi_t \geq 0, \xi_t^* \geq 0 \end{aligned} \tag{12}$$

[1] Note that Vapnik et al. (1997) introduced this loss function as the ϵ-insensitive error function, however here it is renamed η to avoid confusion with the response variable ϵ_t^2

By using the Lagrange multipliers $\alpha_t \geq 0$, $\alpha_t^* \geq 0$ together with the Karush-Kuhn-Tucker optimality conditions this minimization problem can be solved through the dual problem as defined as in eq. (13).

$$\begin{aligned}\arg\max Q(\alpha,\alpha^*) = &-\frac{1}{2}\sum_{t=1}^{T}\sum_{k=1}^{T}(\alpha_t-\alpha_t^*)(\alpha_k-\alpha_k^*)\kappa(x_t,x_k)\\ &-\eta\sum_{t=1}^{T}(\alpha_t+\alpha_t^*)+\sum_{t=1}^{T}(\alpha_t-\alpha_t^*)\epsilon_t^2\\ \text{s.t. } &\sum_{i=1}^{N}(\alpha_t-\alpha_t^*)=0, 0\leq\alpha_t,\alpha_t^*\leq C\end{aligned} \tag{13}$$

Finally, the obtained model by solving the dual problem can be used to make predictions for out-of-sample data through:

$$\hat{\epsilon}_t^2 = \sum_{t=1}^{T}(\alpha_t-\alpha_t^*)\kappa(x,x_t)+b \tag{14}$$

Using the dual formulation of the optimization problem can be seen as disadvantageous, since the number of variables can exceed the number of variables in the original problem and therefore leads to a computationally more complex problem. However, this dual formulation introduces the kernel function $\kappa(x_t, x_k)$ that allows the model to be applied to high dimensional data sets and to even work efficiently when the dimensionality surpasses the number of observations. There are a large amount of kernels available and in Table 1 three popular and widely used kernels are presented; the polynomial kernel (of order d), the gaussian radial basis function kernel and the hyperbolic tangent (sigmoid) function kernel (Perez-Cruz et al., 2003; Santamaria-Bonfil et al., 2013; Qu and Zhang, 2016).

Table 1: Kernel functions

Polynomial	*RBF*	*Sigmoid*
$\kappa(x_t,x_k)=(1+x_t\cdot x_k)^d$	$\kappa(x_t,x_k)=\exp(-\gamma\|\|x_t-x_k\|\|^2)$	$\kappa(x_t,x_k)=\tanh(\kappa_1 x_t\cdot x_k-\kappa_2)$

The gaussian radial basis function kernel is used in this research and is probably the most extensively employed kernel due to the resemblance of the gaussian distribution (Peng et al., 2018). The similarities between data points are based on their Euclidian distances and the specific hyperparameter of this kernel is γ. Each different kernel relies on different assumptions about the generating process of the particular time series and it is difficult to conclude beforehand which kernel will perform the best. The radial basis kernel is often selected when applying SVR to time series data, however the tuning of the hyperparameters based on the particular data set plays a substantial role in its performance (Ruping, 2001).

2.3.3 Random Forest

The random forest algorithm was first introduced by Breiman (2001) as a supervised learning method building on an ensemble of decision trees both employable for classification as well as regression problems. These decision trees are constructed following the Classification And Regression Trees (CART) algorithm by Breiman et al. (1984) (for the step-wise algorithm refer to Appendix A.2). Next, the model makes a partition of the explanatory variable space to fit a noncomplex model on each of them. In order to build a regression tree the data set of N observations with p input variables (that is: (x_t, ϵ_t^2) for $t = 1, .., T$ and $x_t = (x_{t1}, .., x_{tp})$) is partitioned in M separate regions $R_1, ..., R_M$ with c_m the response variable as a constant per each region as follows:

$$f(x) = \sum_{m=1}^{M} c_m I(x \in R_m) \tag{15}$$

The criterion function minimized affects the prediction of the constant c_m, in this research this is the sum of squared residuals. This leads to the optimum for $\hat{c}_m$ as the average of the response variable ϵ_t^2 in the region R_m:

$$\hat{c}_m = \text{average}(\epsilon_t^2 | (x_T \in R_m)) \tag{16}$$

Generally, finding the best split based on this minimum sum of squares criterion is infeasible and therefore the CART algorithm builds from the top down and applies a greedy algorithm to find the best pairs of splitting points leading to the highest reduction of the selected criterion function. The process continues until all leafs consist of a single observation or is stopped when a certain stopping criteria is reached (Hastie et al., 2009).

Random forests employ the so called bagging technique in order to lower the variance when used for prediction. In a regression problem random forest will average the predicted values of multiple decision trees. Additionally, the algorithm deviates from the CART algorithm to grow its decision trees by selecting a fixed number of randomly chosen explanatory variables instead of selecting all explanatory variables to calculate the best split on. In this way the random forest algorithm can significantly raise the forecasting accuracy compared to the accuracy of a single decision tree, by reducing the variance due to the averaging of many maybe noisy but nearly unbiased models (Hastie et al., 2009). Algorithm 1 shows step-wise how the random forest in regression is constructed.

The output of this random forest algorithm for regression are B trees T_b and the random forest model will then be the average of these trees as below.

$$\hat{f}_{RF}^{B}(x) = \frac{1}{B} \sum_{b=1}^{B} T_b(x) \tag{17}$$

Algorithm 1: Random forest algorithm for regression trees.

```
Data: Data set $(x_t, \epsilon_t^2)$ with response variable $\epsilon_t^2$ for $t = 1, ..T$ and features $x_{ti}$ for $t = 1, ..T$,
  $i = 1, .., p$.
    1. Bootstrap B samples (with replacement) from the data to fit a tree.
    2. For a selected bootstrapped sample take a root node with the complete data set.
    3. At the current node, select $m$ explanatory variables at random from all explanatory variables in the
       sample.
    4. Find the best split points $s$ maximizing the splitting criterion for these $m$ variables.
    5. Find among the pairs of explanatory variables and best splits $(i, s)$ from step 4 the best pair
       maximizing the splitting criterion and split the current node on this split.
    6. Repeat the process from step 2 for all terminal nodes until the stopping criteria is reached or the
       tree is fully grown.
    7. Repeat the process for each bootstrapped sample.
```

2.3.4 Gradient Boosted Trees

Tree boosting is contrary to random forest a sequential approach to form a prediction model based on an ensemble of decision trees. It was originally described in order to solve classification problems in binary form by Freund and Schapire (1996). Later boosting was also described in the regression framework by Friedman (2001) where it is especially worthy to apply in the case of a large number of explanatory variables, with a high chance of multicollinearity problems as it restraints their influence by shrinking the coefficients to zero. The goal of the boosting method is sequentially add a simple algorithm to the ensemble taking into account the errors encountered in the previous trees. The final model will then be a combination of all these estimator through weighted majority voting. Gradient boosting relies on the gradient descent in order to locate the errors of the previous tree (Nabipour et al., 2020).

Due to this sequential approach boosted trees hold some advantages over other tree based methods; they can easily work with very high dimensional problems where they select relevant variables only and will ignore useless ones. Compared to random forests which loose most interpretability due to the combining of many trees, a boosted tree is straightforward in terms of interpretation. Additionally, a boosted tree is capable of capturing nonlinearities quite well and has proven to exhibit great properties in the estimation of such series (Mittnik et al., 2015). As financial time series data exhibit nonlinear properties this quality along with the other described advantages of boosted trees make them a promising tool in the estimation of the AEX volatility.

The gradient boosting algorithm again builds on multiple regression trees following the CART algorithm as in eq. (15). Then, a boosted tree is the summation of these single trees, where the following tree is estimated in a forward manner given the present model f_{m-1}. First, the model is initialized with a constant, depending on the loss function $L(\cdot)$ that is selected. The boosted tree then solves in a forward manner the following minimization problem:

$$\text{argmin} \sum_{t=1}^{T} L(\epsilon_t^2, f_{m-1}(x_t) + \sum_{m=1}^{M} c_m I(x \in R_m) \tag{18}$$

Where $L(\cdot)$ is again the loss function employed, which in this research is the squared loss: $\frac{1}{2}(\epsilon_t^2 - f(x_t))^2$. Next, in order to solve the minimization problem the gradient descent is applied. After computing the negative gradient values or pseudo residuals r_{tm}, a regression tree is fitted to them producing the regions R_{jm}. These negative gradient values simply become the regular residuals $\epsilon_t^2 - f_{m-1}(x_t)$ when applying squared loss. Finally, the multiplier c_{jm} can be computed and is used to update the model. This is repeated until the number of trees M is reached. Algorithm 2 shows the step-wise procedure with as output $f_M(x)$, the estimated gradient boosted tree.

```
Algorithm 2: Gradient regression tree boosting algorithm
Data: Data set $(x_t, \epsilon_t^2)$ with response variable $\epsilon_t^2$ for $t = 1, ..T$ and features $x_{ti}$ for $t = 1, ..T$,
  $i = 1, .., p$.
    1. Initialize for $f_0(x) = \text{argmin}_c \sum_{t=1}^{T} L(\epsilon_t^2, c)$ and set m=1.
    2. Compute $r_{tm} = -\left[\frac{\partial L(\epsilon_t^2, f(x_t))}{\partial f(x_t)}\right]_{f=f_{m-1}} \forall t = 1, .., T$.
    3. Fit a regression tree following the CART algorithm to $r_{tm}$ producing regions $R_{jm}$.
    4. Compute $c_{jm} = \text{argmin}_c \sum_{x_t \in R_{jm}} L(\epsilon_t^2, f_{m-1}(x_t) + c) \; \forall j = 1, ..J$.
    5. Update $f_m(x) = f_{m-1}(x) + \sum_{j=1}^{J} c_{jm} I(x \in R_{jm})$.
    6. Do m+=1 and repeat from step 2 until m=M.
```

2.4 Hybrid Method

The following paragraph contains a description of the hybrid method applied to the forecasting of the AEX volatility.

2.4.1 EGARCH-SVR

A novel method is introduced in order increase the predictive performances of the standalone GARCH as well as machine learning methods. Since financial time series is a complex series to predict, its different characteristics might not be captured by the standalone models. Combining the parametric EGARCH model and its convenient statistical information on the AEX volatility and the nonparametric SVR model to better capture the nonlinear features could increase predictive performance. In past research on volatility forecasting often the maximum likelihood method to estimate the GARCH parameters is replaced by support vector machines (Perez-Cruz et al., 2003; Bezerra and Albuquerque, 2017). However recently, Sun and Yu (2020) introduced a novel two stage combined method of GARCH and SVR as well as GJR-GARCH and SVR to forecast volatility.

In their approach first the traditional maximum likelihood method is applied to estimate the GARCH parameters whereafter the SVR model is trained to fit the errors made by the in-sample-estimations of the GARCH model and finally the combined model is applied to forecast the volatility. Algorithm 3 describes step-wise the construction of this model, where it differs from the research by Sun and Yu (2020) in terms of the volatility proxy as they select the five day moving average of the squared log return series, whereas in this research this is the demeaned squared log return series. Therefore, the SVR is trained on the data matrix containing: σ_{t-1}^2, the estimated in-sample conditional variances by the EGARCH model, M_{t-1}, the sequence of the true volatility (proxy) minus the estimated volatility by the EGARCH model and finally ϵ_{t-1}^2, the true value of the volatility (proxy), all at time $t-1$. Note that the forecasted value of $\hat{\epsilon}_t^2$ is bounded by zero below.

Algorithm 3: GARCH-SVR hybrid model algorithm

```
Data: Daily log return series $r_1, ..r_T$ for the AEX.
    1. Estimate the GARCH parameters with the ML method
    2. With the estimated parameters compute the in-sample conditional variances $\sigma_1^2, .., \sigma_T^2$
    3. Compute the sequence $M_1, .., M_T$, where $M_t = \epsilon_t^2 - \sigma_t^2$
    4. Train a SVR on the data matrix containing $\sigma_{t-1}^2, M_{t-1}, \epsilon_{t-1}^2$
    5. Compute the one-day-ahead forecast of $\hat{M}_{T+1}$ with the SVR model
    6. Compute one-day-ahead forecast of $\hat{\sigma}_{T+1}$ using the GARCH model
    7. Compute the forecasted value of $\hat{\epsilon}_{T+1}^2$ by computing $\hat{M}_{T+1} + \hat{\sigma}_{T+1}$
```

This method of model combination presents a promising extension to the volatility forecasting based on their simulation study on the GARCH-SVR and GJR-GARCH-SVR and therefore is adapted to fit the framework of this research. Furthermore, instead of employing the GARCH and the GJR-GARCH model the EGARCH model with normal distributional assumption is applied, leading to the novel EGARCH-SVR model.

2.5 Tests and Evaluation Measurements

In this section first the forecasting procedure is described followed by the performance measures and Diebold-Mariano test for forecast comparisons.

2.5.1 Forecasting Procedure and Performance Measures

In order to asses the predictive performance of the described methods, one-day-ahead out-of-sample forecasts are computed over different forecasting horizons, as the selection of a forecast horizon may lead to dissimilar results in terms of best performing model. In order to assess the robustness of the methods in volatility forecasting over different horizons and different levels of average volatility, seven windows are distinguished: one, two, three and six months ahead, one and two years ahead as well as the complete reserved test set of 784 observations, where a month is defined as 20 successive business days. The estimated parameters for the statistical methods are not re-estimated and the machine learning models are not re-fitted in this procedure.

Next, the predicted volatilities are evaluated in their fit to the actual values of the volatility of the reserved test set. A popular approach in order to do so is computing a certain loss function, however a single criterion that selects the best performing method is nonexistent which makes the evaluation of competing models more difficult (Bollerslev and Nelson, 1994). In most literature on the comparison of volatility models the focus is on the (root) mean squared error ((R)MSE) as loss function to be minimized (Ryll and Seidens, 2019). However, the RMSE is highly effected by large errors that can occur in volatility forecasting and therefore it is suggested to deviate to for example an absolute error loss function that is less sensitive to these extreme observations (Patton, 2006).

Since there is no conclusive evidence on what evaluation function is the most suitable to compare volatility models, for this research the following two goodness-of-fit metrics are used: the mean absolute error (MAE) and the RMSE. By not selecting a single evaluation function the insights in the performance of the forecasting models can improve (Brailsford and Faff, 1996). Initially the mean absolute percentage error (MAPE) measure was also considered, but it is omitted as it takes on extreme values when there are some true values that are very close to zero, which is common in the financial return series.

The lower the value of these function the closer the prediction is to the true volatility, where $\hat{y}_t$ is the predicted value of the model, ϵ_t^2 the true value of the AEX volatility and n the number of observations. Note that $\hat{y}_t$ takes the form of $\hat{\sigma}_t^2$ when considering the statistical methods and $\hat{\epsilon}_t^2$ when considering the machine learning methods.

$$MAE = \frac{1}{n}\sum_{t=1}^{n}\left|\hat{y}_t - \epsilon_t^2\right| \tag{19}$$

The MAE denotes the average of the absolute errors made by the model as in eq. (19) and is also known as the $L1$ loss of the model. The function is easily calculated and is useful in the case of outliers in the training data as it does not extra penalize higher errors caused by these outliers.

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}\left(\hat{y}_t - \epsilon_t^2\right)^2} \tag{20}$$

The RMSE as in eq. (20) denotes the square root of the squared errors and is also known as $L2$ loss of the model, which penalizes a model with high errors more by squaring them. This causes the RMSE to be more sensitive to outliers.

An additional measure that can be computed for the tree based methods is the relative importance of the explanatory variables. A tree following the CART methodology is grown through reducing a node's impurity in a recursive manner (for the step-wise CART algortihm refer to Appendix A.2). Since both random forest as well as gradient boosted trees are based on this methodology, the impurity of the explanatory variables is a side product of the algorithms splitting rule and therefore easy to compute and is determined by averaging the amount of decrease of a node's impurity over all trees. The Gini index measures this impurity in case of a classification problem (Breiman, 2001), whereas for a regression problem this is typically the sum of squares (Ishwaran, 2015). The way of averaging depends on the type of tree based method used. Through this measure it is possible to express the relative importances of the explanatory variables in the forecasting of the AEX volatility, indicating the informativeness of the additional data on stocks, exchange rates and macroeconomic variables.

2.5.2 Diebold-Mariano Test

The Diebold-Mariano (DM) test is a test designed for the purpose of comparing forecasts (Diebold and Mariano, 1995). The goal of this test is to determine if a forecast of a certain model is significantly better compared to the forecast of another model. One can be tempted to favor a model with a slightly smaller value of the loss functions as described in the previous paragraph, however this does not necessarily indicates that the particular model is significantly better. The DM test relies on the differences between loss functions and defines the loss differential $\delta_{ij,t}$ between model i and j at time t is as in eq. (21) where $L_{i,t}$ ($L_{j,t}$) the loss function at time t for model i (j).

$$\delta_{ij,t} = L_{i,t} - L_{j,t} \tag{21}$$

The test assumes that the loss differential $\delta_{ij,t}$ process is covariance stationary (Diebold, 2012). Under the null hypothesis (H_0) both competing models are of equal forecasting quality and thus have equal expected loss: $\mathbf{E}[\delta_{ij,t}] = 0$. Under the alternative hypothesis (H_A) the mean of the loss differential series is not equal to zero and therefore there is a significant difference between the two tested models regarding their forecasting performance. To test the null a asymptotic z-test is employed and the DM test statistic is as follows:

$$DM = \frac{\bar{\delta}_{ij}}{\hat{\sigma}_{\bar{\delta}_{ij}}} \tag{22}$$

With $\bar{\delta}_{ij,t} = \frac{1}{T}\sum_{t=1}^{T}\delta_{ij}$, the average of the loss differential over time t, $\hat{\sigma}_{\bar{\delta}_{ij}}$ a consistent estimator for the standard deviation of the average of the loss differential and T the out-of-sample number of observations. If the assumption of a covariance stationary process holds the DM test statistics simply follows the standard normal distribution under null. The null hypothesis is rejected if the p-value is below the selected significance level α in which case there is a significant difference between the two forecasts. A downside of the Diebold-Mariano test is that it might reject too often in small sample sizes. In that case an alternative to this test could be the Harvey Leybourne and Newbold test (Harvey and Newbold, 1997).

3 Data

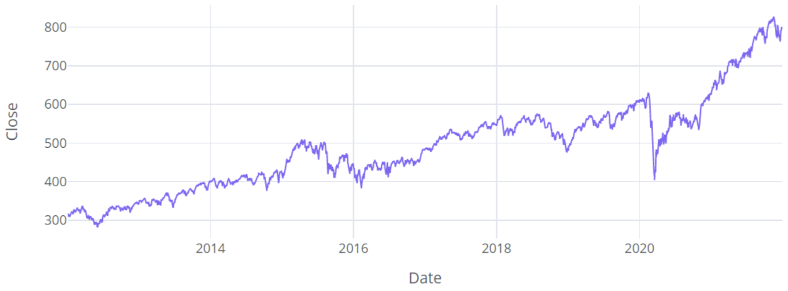
In this chapter the data will be presented and its characteristics will be described to provide insight in the dynamics of the AEX. Since the statistical models as described in section 2.2 and the machine learning models as described in section 2.3 do not employ entirely the same data set, this section is split into two parts. The first part describes the AEX data particularly on the index price, its daily close-to-close log return series and squared log return series. The second part describes all additional (macroeconomic) variables used for the machine learning methods.

3.1 AEX Historical Data

The Amsterdam Exchange Index provides a historical data set of the daily closing price of the index which is retrieved from the Yahoo! Finance[2] website where it is publicly available. The AEX is a good representation of the development of the Dutch stock market as it is a weighted index of the 25 most prominent stock market listed companies in the Netherlands. To asses the performance of the considered models in the prediction of the volatility of the AEX the data is analyzed from 2012-01-01 until 2021-12-31, which are 2555 observations.

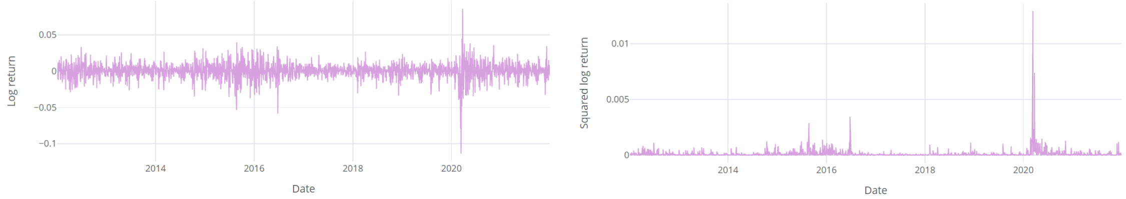
[2] Retrieved from: https://finance.yahoo.com/

Figure 1: AEX daily closing prices



The daily closing prices are presented in Figure 1 and exhibit a bullish trend with some smaller and some bigger drawdowns. An example of a drawdown is the peak visible at the end of 2015: the year of the refugee crisis in Europe. The largest peak is evidently observed early on in 2020 representing the economic disruption caused by the Covid-19 crisis. In general the drift is upwards sloping and these movements indicate that this financial time series is non-stationary.

Figure 2: AEX log return series and squared log return series



In the figure on the left the daily close-to-close log return series for the AEX is presented as calculated in eq. (1) and in the figure on the right the squared daily close-to-close squared log return series is presented. Both series are displayed from the period of January 2012 to December 2021.

The daily log return series and squared daily log return series in Figure 2 clearly show these drawdowns through higher peaks or in other words, bigger differences between the close-to-close prices. Furthermore, the phenomenon of volatility persistence or clustering, which is a common occurrence in financial time series, is visible in both figures as well. This persistence of the magnitude of the returns is caused by the fact that large changes in stock price are likely to follow large changes (Cont, 2004).

Table 2 shows the descriptive statistics for the closing price, log return and squared log return series. In order to check for stationarity of the series the augmented Dickey-Fuller test is applied and in order to check for normality the Jarque-Bera test is applied. For a complete description of these tests please refer to Appendix A.3 and A.4 respectively. As expected from the theory for the closing prices of the AEX the null hypothesis of non-stationarity is not rejected and thus the series is non-stationary, whilst for the log return series as well as the squared log return series the null is rejected and thus these series are stationary. Both return series exhibit excess kurtosis whilst the closing price series has a kurtosis of less than three. All three skewness coefficients deviate from zero, where the log return is negative as expected. From these values it is expected that the normality assumption is violated. The Jarque-Bera test for nomality confirms this, where for all series the test statistic exceeds the critical value at the 1% significance level. This indicates that for the modeling of the log return series a thicker tailed distribution is expected to be favored.

Table 2: Summary statistics of the AEX data

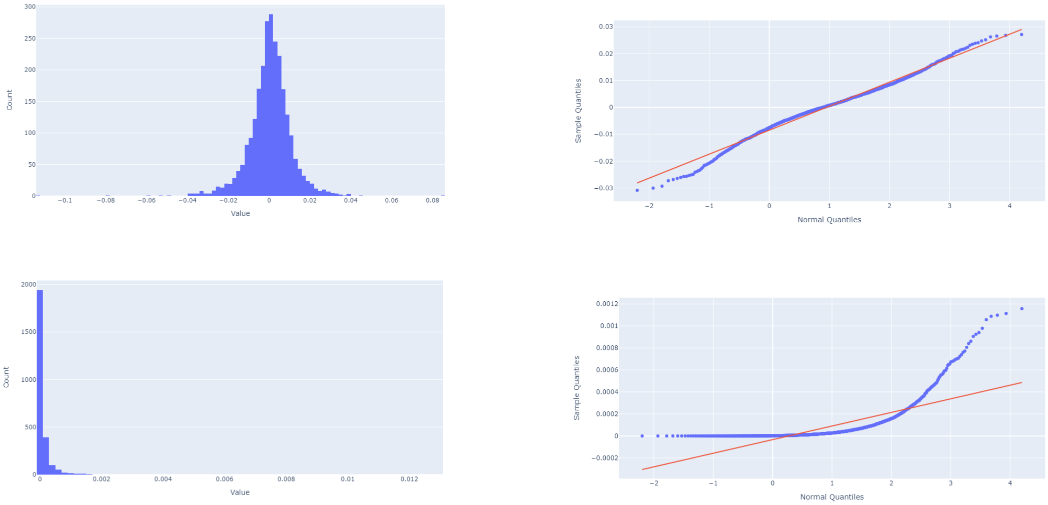
Series	Mean	Median	Std. deviation	Minimum	Maximum
Close	492.51	491.75	116.90	283.07	827.57
Log return	3.6100e-4	7.4045e-4	0.010620	-0.11376	0.085907
Squared log return	1.1300e-4	2.7444e-5	3.8600e-4	0.00	0.012941

Series	Skewness	Kurtosis	ADF	JB
Close	0.61856	0.20643	-0.09181 (p=0.95)	167.59 (p=0.0)
Log return	-0.76410	9.7981	-18.071 (p=2.6045e-30)	1.0469e4 (p=0.0)
Squared log return	19.639	556.57	-6.4478 (p=1.5499e-8)	3.3142e7 (p=0.0)

In the table the descriptive statistics for the AEX closing price, the log return series as well as the squared log return series are presented for the period of January 2012 to December 2021. Where JB is the test statistic and its p value in parenthesis of the Jarque-Bera test for normality and ADF the test statistic and its p-value in parenthesis of the augmented Dickey-Fuller test for unit roots.

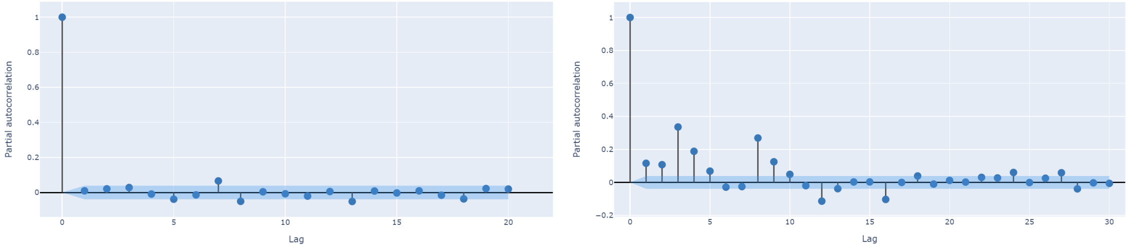
The histograms and quantile-quantile plots (QQ plot) in Figure 3 indicate the same results as the Jarque-Bera test results. Considering the log return series, the histogram shows that the series is centered around zero, however the tails are worth noticing. In the QQ plot the quantiles are compared to those of the normal distribution, where the deviation from the line in both tails shows that the data is heavier in the tails compared to the normal distribution. The squared log return series evidently does not follow a normal distribution. Furthermore, in Figure 4 the log return series partial autocorrelation shows almost no significant correlations besides one minor at the sixth lag. This indicates that the series is most likely serially uncorrelated. However, the partial autocorrelation of the squared log return series clearly shows a higher autocorrelation with past lags, indicating that the series is not serially independent.

Figure 3: Histograms and QQ plots of AEX daily log return series



In the top left figure the histogram of the log return series is presented with in the top right figure the corresponding QQ plot. In the bottom left figure the histogram of the squared log return series is presented with in the bottom right figure the corresponding QQ plot. For both QQ plots the x-axis are the theoretical quantiles of the normal distribution. Both figures are presented for the period of January 2012 to December 2021.

Figure 4: Partial autocorrelation of AEX daily log return series and daily squared log return series



In the figure on the left the partial autocorrelation for the log return series of the AEX is presented an in the figure on the right the partial autocorrelation for the squared log return series of the AEX is presented, both for the period of January 2012 to December 2021.

3.2 Additional Data

An important strength of the applied machine learning methods is their ability to deal with high dimensional data. This makes it feasible to include many explanatory variables in the data set whereof the methods select the ones most valuable in order to construct the best predictive model (Bishop, 2006). Therefore, the data set is extended with multiple variables of the most prominent stocks the AEX contained in the period from 2012 to 2021. The stocks are selected based on their size and share in the AEX and are the following thirteen stocks: Aegon, Royal Ahold Delhaize, Akzo Nobel, ArcelorMittal, ASML Holding, Royal DSM, Heineken, ING Groep, KPN, Randstad, Shell PLC, Unibail-Rodamco-Westfield and Wolters Kluwer. Since these stocks together are considerably the largest part of the AEX[3], it is expected that the machine learning models can extract valuable information out of their data to more accurately predict the AEX volatility.

From the data of all these stocks as well as from the data of the AEX multiple variables are incorporated. These include for example not only the high, low and close values for each stock but also the volume of the stocks, since empirical research has showed that volume is positively correlated to the return of the series (Chen et al., 2001; Wang and Huang, 2012). Moving averages are included as they can be important indicators for future volatility (Brailsford and Faff, 1996). Finally, besides these variables the data is supplemented with a number of macroeconomic variables as well. As proven in previous literature on equity return forecasts, some macroeconomic variables might be good indicators for the fluctuation of a national stock index (Welch and Goyal, 2008). The first set of variables included in this research concern the fluctuations in the supply and demand of the oil markets through Dutch prices of benzine, diesel and gas. The second set of variables are included to get an idea on the state of the Dutch and European economy; exchange rates, nominal effective exchange rate, gold price, composite indicator of systemic stress (CISS), short-term european paper (STEP), yield curve spot rates and yield curve forward rates.

Table 3 contains all variables as input for the machine learning methods. As from figure 4, the lags of the squared log return series exhibit significant correlation with the current value. The last significant correlation is present at lag 28 and therefore this series is incorporated up to this lag. For the additional explanatory variables the first lags are added to the input data matrix. In order to ensure a similar scale data normalization is applied for all additional variables described. Hereafter all variables are checked for stationarity with the augmented Dickey-Fuller test. If a series is non-stationary data transformation is applied, i.e. either the logarithm of the variable or the first difference of the variable is taken before adding it to the data matrix, if a variable remained non-stationary after these transformations it will be omitted from this research due to lack of interpretability. However the latter was not necessary, since after the data preparation all explanatory variables are stationary.

[3] Composition of AEX retrieved from euronext.com

Table 3: Complete list of additional explanatory variables

Variable	**Description**	**Source**
Stock data	Open, high, low, close, adjusted close and volume data for all stocks included in this research as well as the AEX.	Yahoo! Finance[1]
Log close-to-close return	Computed log close-to-close return series for all stocks included.	-
Squared log close-to-close return	Computed squared log close-to-close return series for all stocks included.	-
Simple moving averages	Simple moving averages of the log return series of the AEX with 3, 12 and 26 days as windows for short term averages and 50 and 200 days for long term averages.	-
Exponential moving averages	Exponential moving averages of the log return series of the AEX with 3, 12 and 26 days as windows for short term averages and 50 and 200 days for long term averages.	-
Oil prices	Contains average pump prices of motor fuels per day in euros for the Netherlands. The three pump prices considered are; benzine Euro95, diesel and LPG. Prices are including VAT and excise duty.	CBS Open Data Statline[4]
Exchange rates	Open, high, low, close and adjusted close data for three data sets of exchange rates; United States dollar to euro (USD/EUR), British pound sterling to euro (GBP/EUR) and the Japanese yen to euro (JPY/EUR).	Yahoo! Finance[1]
Nominal effective exchange rate	Computed Effective Exchange Rate (EER) by the ECB on the euro opposed to a group of 19 trading partners.	European Central Bank Statistical Data Warehouse[5]
Gold price	London Bullion Market Association Gold Price (previously London Gold Fix). Gold price set in USD two times a day, this data set contains the gold price at 15:00 London GMT. Corrected data set from US dollars to euros per troy ounce.	ICE Benchmark Administration Limited[6]
Composite indicator of systemic stress (CISS)	The CISS is computed for the entire euro area. It is an indicator that measures the current level of instability (i.e. stress and frictions) and is therefore a measure of systemic risk.	CBS Open Data Statline[3]
Short-term european paper (STEP)	The STEP is an aggregated value of total outstanding amounts of short term issued debt securities that possess the STEP label. Value is issued by the total economy (world) expressed in euros.	European Central Bank Statistical Data Warehouse[4]
Yield curve spot rate	The yield curve spot rate presents the yield to the remainder of the time to maturity for a zero coupon bond. The yield curve spot rate of the Euro area is included for 1-year, 2-year, 5-year and 10-year maturity.	European Central Bank Statistical Data Warehouse[4]
Yield curve forward rate	The yield curve forward rate presents the expected interest rate or a zero coupon bond. The yield curve forward rate of the Euro area is included for 1-year, 2-year, 5-year and 10-year maturity.	European Central Bank Statistical Data Warehouse[4]

In this table the complete list of explanatory variables for the machine learning methods is presented including a description per each variable. The periodicity of the variables is daily on business days. For the complete list of corresponding abbreviations, please refer to Appendix B.3.

[4] Retrieved from: https://opendata.cbs.nl/statline/portal.html?_la=nl&_catalog=CBS

[5] Retrieved from: https://sdw.ecb.europa.eu/

[6] Retrieved from: Federal Reserve Bank of St. Louis: https://fred.stlouisfed.org/

4 Results

The result chapter is structured as follows: the first section will be dedicated to the results of the GARCH models, whereafter the results of the learning models are described and compared to the GARCH benchmarks.

4.1 GARCH Models

The GARCH models are primarily used as a benchmark for the machine learning models to get an in depth examination of the true performances of the considered models in volatility forecasting. To fit the GARCH models, the preprocessed data set is split into two parts; a training and a testing set. The training set contains 70% of the observations which are 1763 observations leaving 30% or 756 observations for testing. The models predictive performance will be evaluated over different forecasting horizons as described in paragraph 2.5.1, where the estimation of the parameters will be done once on the training set.

Table 4: Log likelihood and information criteria of the GARCH models

	Normal			*Student t*		
	GARCH	**EGARCH**	**GJR-GARCH**	**GARCH**	**EGARCH**	**GJR-GARCH**
Log L	5740	5789	5779	5774	5817	5810
AIC	-11474	-11570	-11550	-11540	-11623	-11610
BIC	-11457	-11548	-11528	-11521	-11596	-11583

In the table the maximized log likelihood (Log L) function value is presented together with the information criteria (AIC and BIC) for all models. The results are based on the maximum of the likelihood function of the models in their fit to the training data.

In Table 4 the maximized log likelihood function value and information criteria values are presented. The results unanimously select the EGARCH model as best fitted model to the training set for both the normal and the student t distributional assumption by exhibiting the largest likelihood in combination with the lowest values of the information criteria. The GJR-GARCH model fits second best leaving the GARCH to be the worst fitted to the training data. This finding is in line with expectation that due to the asymmetrical behaviour of the financial return series the models that allow for asymmetry and leverage effect, which are the EGARCH and the GJR-GARCH model, are likely to fit better to the series compared to the symmetrical GARCH model. When comparing the two distributional assumptions of the returns, the use of the student t distribution increases the fit of the models to the return series. This was expected since financial returns often exhibit non-normalities.

In Table 5 the estimated parameters are presented, where not all parameters are significant. However, for the EGARCH model all parameters are significant for both distributional assumptions. This indicates that the EGARCH model probably fits well to the data, which is coinciding with the previous results based on the likelihood and the information criteria. Moreover, both the EGARCH model as well as the GJR-GARCH model allow for asymmetry and the presence of leverage effect in the series as the parameters satisfy the conditions described in section 2.2.

Table 5: Estimated parameters of the GARCH models

Normal	**GARCH**	**EGARCH**	**GJR-GARCH**
Parameter	Coefficient (p-value)	Coefficient (p-value)	Coefficient (p-value)
ω	2.292e-06 (0.500)	-0.423 (0.00)***	2.419e-06 (0.499)
α	9.978e-02 (0.00)***	0.135 (0.00)***	-6.420e-03 (0.189)
β	0.877 (0.00)***	0.967 (0.00)***	0.880 (0.00)***
γ	-	-0.159 (0.00)***	0.201 (0.00)***
η	-	-	-

Student t	**GARCH**	**EGARCH**	**GJR-GARCH**
Parameter	Coefficient (p-value)	Coefficient (p-value)	Coefficient (p-value)
ω	2.350e-06 (0.500)	-0.483 (0.00)***	2.970e-06 (0.499)
α	0.108 (0.00)***	0.143 (0.00)***	-1.337e02 (4.124e-02)
β	0.871 (0.00)***	0.961 (0.00)***	0.865 (0.00)***
γ	-	-0.180 (0.00)***	0.235 (0.00)***
η	6.490 (1.607e-03)***	8.070 (0.00)***	7.687 (0.00)***

In the tables the estimated parameters for the GARCH models are presented with their p-values in parenthesis. The p-values are based on the standard errors computed through the square root of the inverse of the Hessian, following the work of Gill and King (2004). One or three asterisks indicate significance at the 5% or 1% significance level respectively. All parameters are estimated in the training data set.

Table 6 shows the performance measures of the GARCH models in the one-day-ahead volatility forecasting of the AEX for all forecasting horizons. Overall the performance measures are lowest for the six months and one year forecasting horizons. This is directly related to the AEX developments as visible in Figures 1 and 2, where the start of the out-of-sample prediction period shows higher peaks in volatility caused by more fluctuation in the AEX. This is followed by a calmer market period leading to lower average errors made by all considered models wherafter the massive impact of the Covid-19 crisis clearly manifests in an increase in the performance measures.

Based on the MAE measure the EGARCH model with normal distribution is preferred for most long term horizons, except for the one year window where the GARCH model with normal distribution is preferred. In the shorter horizons the GARCH normal and EGARCH student t model appear. Based on the RMSE, for all forecasting windows except the one year window, the EGARCH model with normal distribution is preferred. It is clear that based on both measures, the GJR-GARCH model is never favored. In most cases the differences in performance measures of the models are small and statistical significance of superiority is not always supported by the Diebold-Mariano test.

In the short term forecast horizons a significantly better performance can not be concluded based on the MAE measure, however for the longer horizons of a year, 2 years and the full data set, both EGARCH models outperform both GARCH as well as both GJR-GARCH models. These findings indicate that the EGARCH models are probably less effected by a higher average volatility of the AEX that appears in the last two forecasting horizons. Conducting the Diebold-Mariano test based on the RMSE measure will reduce to the MAE measure as the test computes the loss differential forecast error per observation and therefore the results will be identical. For a more complete review of the performance of the models the Diebold-Mariano test is executed based on the MSE measure. Based on this measure it can not be concluded that any model outperforms the others significantly for all horizons. For the complete list of executed Diebold-Mariano tests please refer to Appendix B.2.

Table 6: Performance measures of the GARCH models

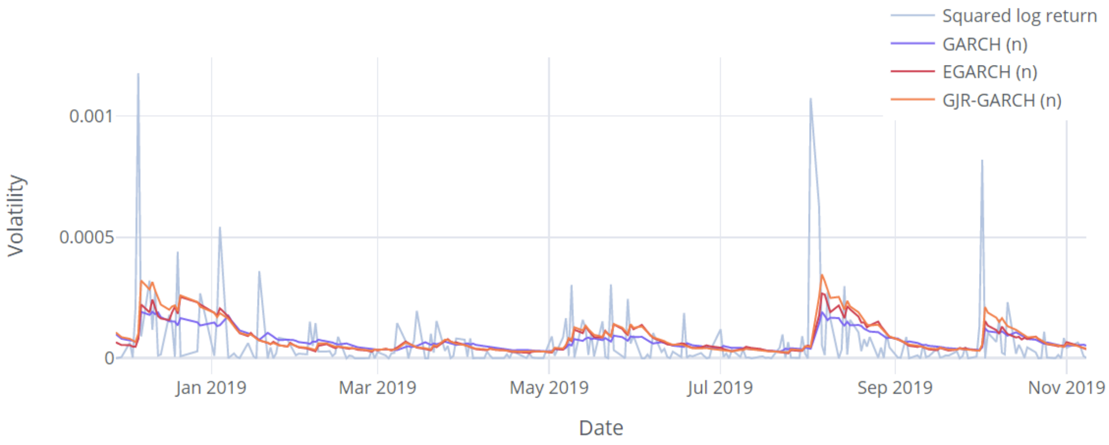
Distr	Model	Measure	1 month	2 months	3 months	6 months	1 year	2 years	Full testset
Normal	GARCH	MAE	1.468	1.335	1.062	0.785	0.747	2.078	1.639
		RMSE	2.328	2.155	1.830	1.368	1.394	7.409	5.910
	EGARCH	MAE	1.493	1.319	1.038	0.746	0.752	1.905	1.515
		RMSE	2.288	2.116	1.818	1.355	1.405	7.289	5.808
	GJR-GARCH	MAE	1.516	1.337	1.071	0.772	0.783	2.195	1.704
		RMSE	2.319	2.158	1.853	1.376	1.429	7.423	5.925
Student t	GARCH	MAE	1.484	1.352	1.073	0.791	0.752	2.098	1.654
		RMSE	2.329	2.158	1.832	1.370	1.396	7.412	5.913
	EGARCH	MAE	1.502	1.314	1.034	0.747	0.758	1.916	1.521
		RMSE	2.293	2.121	1.824	1.360	1.412	7.306	5.822
	GJR-GARCH	MAE	1.570	1.345	1.083	0.781	0.791	2.215	1.717
		RMSE	2.329	2.172	1.867	1.387	1.441	7.486	5.974
	Best over period	MAE	G (n)	EG (st)	EG (st)	EG (n)	G (n)	EG (n)	EG (n)
		RMSE	EG (n)	EG (n)	EG (n)	EG (n)	G (n)	EG (n)	EG (n)

In the table the computed performance measures are presented for all models and distributions, for all forecasting horizons. All values are multiplied by 1000. The best performing models per forecasting period based on the two performance measures are indicated in the bottom two rows of the table, where G, EG and GJR represents the GARCH, EGARCH and GJR-GARCH model respectively and (n) and (st) indicate the normal and student t distributional assumption respectively.

Furthermore looking at the distributional assumptions, it is found that an underlying student t distribution does not lead to a better out-of-sample forecasting performance, on the contrary; assuming a student t distribution almost always leads to a slight increase in performance measure compared to the normal assumption, with the exception of the EGARCH MAE measure at the two and three month forecasting horizon. Based on previous findings the models with student t distributional assumption where expected to perform better in the out-of-sample forecasting due to a better fit to the training data, however the opposite seems to be true based in the out-of-sample results. However, based on the Diebold-Mariano tests this difference in performance is not significant, with the exception of the GARCH model for all forecasting horizons besides one and six months.

In Figure 5 the one-day-ahead volatility forecasts of the GARCH models with normal distributional assumption are plotted along with the squared log return series as proxy for the AEX volatility. As visible the GARCH model tends to estimate the AEX volatility the most conservative whilst the GJR-GARCH model predicts the volatility spikes the highest. The EGARCH model seems to predict closely to the GJR-GARCH model in low volatile times and somewhere in between both other models in high volatile times. Clearly all three models struggle to accurately estimate the high peaks in the AEX volatility, which is for example apparent between August and September of 2019. Furthermore, the three models seem to follow the general trend of volatility reasonably, where the GARCH model performs the worst most likely caused by its symmetrical form.

Figure 5: Daily GARCH volatility forecasts



In the figure the daily volatility forecasts are plotted for the three GARCH class models with normal distributional assumption together with the daily squared log return series as proxy for the AEX volatility for the one year ahead forecasting horizon.

4.2 Machine Learning and Hybrid Models

In the next paragraphs first the tuning of the machine learning models as well as the hybrid model is described, followed by a discussion of the results and a comprehensive comparison.

4.2.1 Hyperparameter Settings

As past research has shown, the building of a well performing SVR model requires carefully selected hyperparameters (Chih Hung et al., 2009). In order to fine tune these parameters the SVR is fitted multiple times on five folds. Since random search is used this process is repeated 100 times, to ensure the combination of parameters found indeed approaches the optimal set. After preliminary testing the SVR model with rbf kernel appeared to often estimate negative values for the AEX volatility, most likely caused by the range of very low and almost zero values of the squared log return series. In order to resolve this issue the SVR model is trained to estimate the natural logarithm of the series whereafter the exponential of the predictions is taken. A drawback of this approach is the fact that the error function of the SVR is now minimized regarding the logarithmic values of the response variable, where it makes no distinction in penalizing errors of different magnitudes which can lead to lower values in terms of absolute error but higher values in terms of squared error.

The best set of parameters are presented in Table 7. The first parameter C is the strictly positive regularization parameter and determines the trade-off of the model complexity and the errors made on in testing. A decrease in C leads to an increase in the strength of regularization and lowers the probability of overfitting of the model on training data. The second parameter applicable is η, that determines the accuracy level of the function approximated. The value of η is highly dependent on the values of the response variable thus must reflect the data. If η falls outside of the range of values that the response variable can take, the SVR will lead to very poor predictive results and when η is set at zero, there is no error allowed. The final parameter applicable is γ, which is a kernel specific parameter that defines the way that the decision boundary is shaped. If γ is underestimated the kernel will lose its nonlinear power but if overestimated the extent of regularization will decrease and the resulting model will be sensitive to noise in the training data set.

The random forest is fitted following the same process and some preliminary testing is performed beforehand to decrease the ranges of the parameter bounds to lower computational costs in this way, since these are often high for tree based models. The optimal values for the parameters based on the training set are described in Table 7. The higher the number of estimators the more likely the random forest is to make accurate predictions since the more simple trees are combined the more the variance is reduced. However, the more trees that need to be grown the higher the computational costs will rise, whilst the true marginal gain of adding trees decreases (Friedman, 2001).

As the maximum number of variables considered at each split increases the single tree will be more powerful however will also be more correlated with the rest of the trees in the forest. Choosing the optimum value for this maximum variables parameter will attain a trade-off between the two in order to obtain the best model (Breiman, 2001). Also, if this number is high the random forest is more likely to overfit on the training data due to the level of complexity and performance on the testing set will be poor. To avoid this the parameter should be significantly lower compared to the number of explanatory variables in the model. The final parameter that is tuned is the maximum depth of the tree, which determines the flexibility of the single trees in the forest. A large or fully grown tree will presumably fit better to complicated noisy functions but tends to overfit on the training data, whereas a very small tree might miss the structures within the data.

Table 7: Hyperparameters of the machine learning and hybrid models

SVR			*RF*			*GB*		
Parameter	**Bounds**	**Optimal**	**Parameter**	**Bounds**	**Optimal**	**Parameter**	**Bounds**	**Optimal**
C	0.001 - 10	1.508	No. estimators	300 - 800	640	No. estimators	40 - 800	40
η	1e-02 - 0.6	0.482	Max variables	6 - 61	11	Max variables	4 - 61	9
γ	1e-06 - 1e-02	6.020e-04	Max depth	3 - 12	4	Learning rate (fixed)	-	0.1

EGARCH-SVR				
EGARCH parameters		*SVR parameters*		
Parameter	**Coefficient**	**Parameter**	**Bounds**	**Optimal**
ω	-0.423	C	0.1 - 6	4.239
α	0.135	η	1e-09 - 1e-06	4.751e-07
β	0.967	γ	1e-10 - 1e-06	6.441e-07
γ	-0.159	-	-	-

In the top table the bounds as well as the optimal values found by randomized search for the hyperparameters of the machine learning models are presented and in the bottom table the same is presented for the hybrid model.

The gradient boosted tree is fitted following the same process and the optimal values are described in Table 7. In order to select the learning rate some preliminary research was performed, since it is undesirable to optimize the learning rate as a hyperparameter through an optimizer. Smaller values for the learning rate lead to smaller improvement steps which require more iterations for training and a higher number of estimators, but can increase the robustness of the model. In most cases the performance of a gradient boosted tree increases when more estimators are added since the algorithm is fairly robust to overfitting. However, in cases of a very noisy data set adding more boosting stages can increase overfitting to the training data (Dietterich, 2000). This is most likely the reason that the number of boosting stages is low. The maximum number of variables is as for random forest the number of variables considered to find the best split.

The hybrid EGARCH-SVR model is trained using a two-stage approach. The EGARCH parameters are estimated with the maximum likelihood method normal distributional assumption and since the coefficients are only estimated once on the training set the parameters are identical to ones stated in Table 5. Next, the hyperparameters of the SVR are optimized with the same approach of randomized search as the machine learning models, however with different training data as described in section 2.4. In Table 10 the list of optimal hyperparameters are presented.

4.2.2 Comprehensive Model Comparisons

In Table 8 the performance measures of the volatility forecasts of the machine learning models as well as the hybrid model are presented for all forecasting windows. For all models both performance measures are relatively lowest at the six month and one year forecasting horizons, which is as expected since the average volatility of the AEX is lowest in these periods. In the one month, two years and full test set the performance measures are the highest, corresponding with the highest level of average volatility of the AEX.

Based on the MAE measure the SVR model has a lower error in the out-of-sample prediction of the AEX volatility for all horizons compared to the other models. Interestingly, the gradient boosted tree performs second best in the three horizons with the highest average volatility, followed by the random forest. This stronger performance of the tree based models in these particular forecasting windows is opposite to the expectation that they would suffer more from the extreme spikes in volatility causing the problematic phenomenon of covariate shift. This is a well known problem in machine learning where the response variable takes on values outside the bounds of the data in the training set, which is difficult for all models to predict but especially for the tree based models, since their predictions are limited as an average of previous observed values of the response variable, meaning that the forecasts are bounded by the range of the training data.

Apparently the EGARCH-SVR model suffers even more from the higher fluctuation of the market compared to the tree based models. It was expected that this hybrid model would be more robust in terms of volatility prediction in high volatile market times by taking the advantages of both the parametric EGARCH model as well as the nonparametric SVR model and would increase the predictive performance compared to the singular models. The poor performance of the hybrid model could have been caused by an expected poor performance of the parametric EGARCH model in high volatile market times, however this is contradicted since the standalone EGARCH model performs better. Between the tree based models, it was expected that the bagging model would outperform the boosting model due to the high noise in the financial return series that boosting algorithms are often more sensitive to. However according to the MAE, almost always the opposite is true and GB performs better than RF, with the exception of the two months forecasting horizon.

According the the RMSE measure, the GB consistently outperforms all other models in forecasting the AEX volatility. The RF is again inferior compared to the boosting model, where it performs second best from the two months until one year forecasting horizon. For the first horizon as well as the largest two horizons the RF seems to be highly affected by the higher average level of volatility of the series and is outperformed by the SVR. Interestingly, the SVR performs the worst out of all models for the calmer times in terms of volatility whilst outperforming the RF and EGARCH-SVR in the last two horizons with higher volatility. The hybrid model disappoints compared to the machine learning standards, as it only outperforms the standalone SVR model in the horizons up until a year.

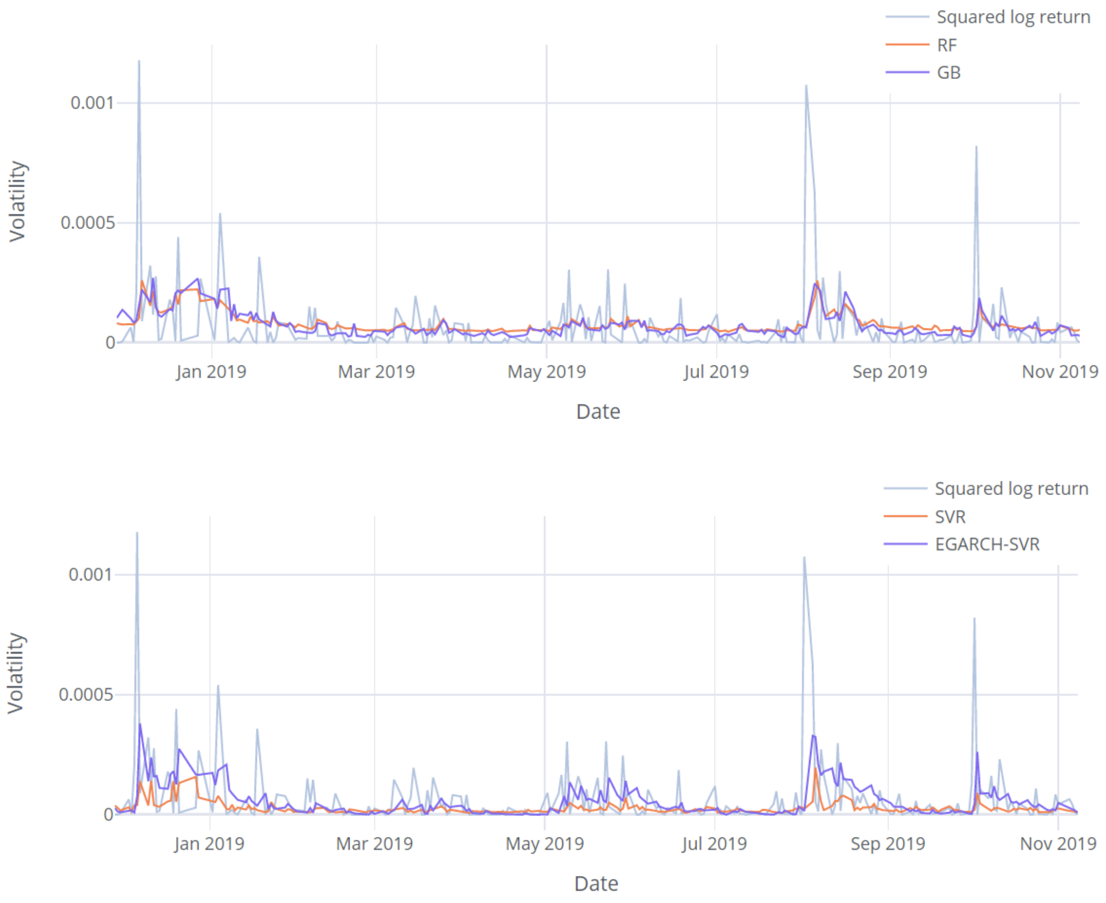
Table 8: Performance measures of the machine learning and hybrid models

Model	Measure	1 month	2 months	3 months	6 months	1 year	2 years	Full testset
SVR	MAE	1.440	1.143	0.870	0.654	0.611	1.766	1.379
	RMSE	2.493	2.243	1.903	1.432	1.442	7.334	5.862
RF	MAE	1.495	1.319	1.060	0.801	0.752	1.874	1.499
	RMSE	2.302	2.113	1.805	1.350	1.367	7.428	5.925
GB	MAE	1.467	1.351	1.050	0.778	0.718	1.869	1.494
	RMSE	2.206	2.057	1.754	1.315	1.340	7.240	5.783
EGARCH-SVR	MAE	1.554	1.323	0.984	0.728	0.681	1.964	1.517
	RMSE	2.364	2.174	1.860	1.409	1.418	7.626	6.076
Best over period	MAE	SVR	SVR	SVR	SVR	SVR	SVR	SVR
	RMSE	GB	GB	GB	GB	GB	GB	GB

In the table the computed performance measures are presented for all machine learning models as well as the hybrid model, for all forecasting horizons. All values are multiplied by 1000. The best performing models per forecasting period based on the two performance measures are indicated in the bottom two rows of the table.

In Figure 6 the one-day-ahead volatility forecasts are plotted together with the squared log return series as proxy for the AEX volatility. Comparing the tree based models, both have a hard time predicting the spiked structure of the squared log return series, which is especially visible in the underestimating the high peaks of volatility as well as overestimating of the days where the volatility is nearly zero. The boosting model has a more spiked structure and seems to fit the structure of the day-ahead volatility better compared to the RF. Both the SVR model as well as the EGARCH-SVR model seem to fit the nearly zero values of the day-ahead volatility better compared to the tree based models. Additionally, the SVR model is more conservative in estimating the volatility spikes compared to both tree based models and the EGARCH-SVR, where the latter seems to fit the spiked structure of the day-ahead volatility better.

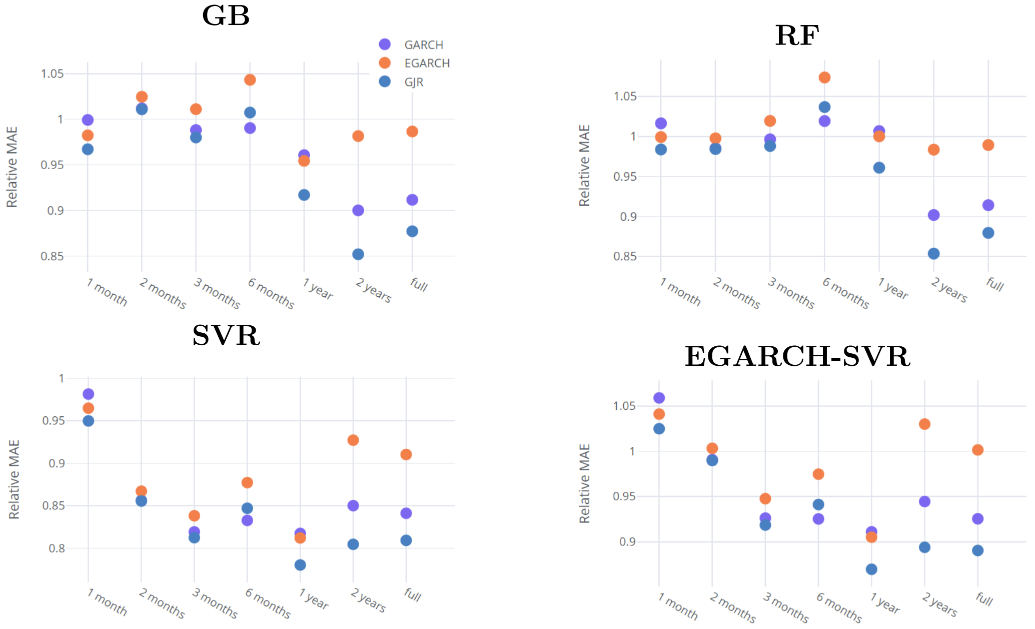
Figure 6: Daily volatility forecasts



In the top figure the daily volatility forecasts are plotted for both tree based models and in the bottom figure the daily volatility forecasts are plotted for the SVR model as well as the hybrid EGARCH-SVR. Both are plotted together with the daily squared log return series as proxy for the AEX volatility for the one year ahead forecasting horizon.

In Figure 7 the performances of the learning models based on the MAE measure are presented relative to the benchmark GARCH models with normal distributional assumption. The SVR consistently outperforms the benchmark models, indicating that the SVR model is indeed better at capturing the nonlinear complex structure of the series and provides more accurate volatility forecasts. The tree based models seem to be closer in terms of predictive performance with the benchmark models especially in the short(er) forecasting horizons, where they alternate in performance. For the larger horizons, both tree based models have a lower prediction error compared to all GARCH type models, where GB is the better of the two. This indicates that the tree based models are more robust to periods of higher average volatility and are better able to predict the volatility in uncertain times. Unfortunately, the hybrid EGARCH-SVR model does not perform consistently and is outperformed by the singular EGARCH model in multiple horizons, whereas it was expected that the combination of the EGARCH predictions with the nonlinear SVR model could decrease the errors made compared to the singular EGARCH model.

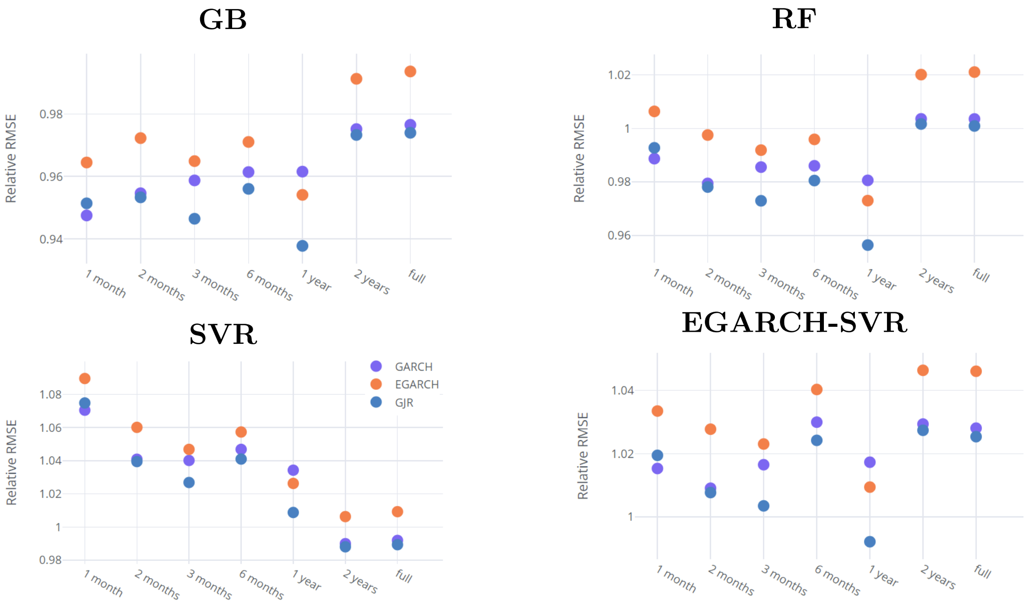
Figure 7: Relative performance of the learning models to the benchmark models based on MAE



In the figure the MAE values are presented for the three machine learning models as well as the hybrid model relative to the benchmark models with normal distributional assumption. A relative value below one indicates that the learning model performs better compared to the benchmark model. For completeness this figure based on GARCH models with student t distributional assumption is included in Appendix B.1.

In Figure 8 the same is presented based on the RMSE measure. Here the GB model consistently outperforms all traditional models. The RF model performs worse compared to the EGARCH model in times of high average volatility and SVR model performs poorly for most forecasting horizons. The latter could be explained by the fact that the SVR model is the only model discussed whose objective function is the minimizing of the logarithm of the response variable and thus does not distinguish errors of different magnitude which will likely cause a higher RMSE. However for the last two forecasting horizons, the SVR performs better and is only outperformed by the EGARCH benchmark model. The EGARCH-SVR again disappoints and performs worse compared to all benchmark models with a single exception.

Figure 8: Relative performance of the learning models to the benchmark models based on RMSE



In the figure the RMSE values are presented for the three machine learning models as well as the hybrid model relative to the benchmark models with normal distributional assumption. A relative value below one indicates that the learning model performs better compared to the benchmark model. For completeness this figure based on GARCH models with student t distributional assumption is included in Appendix B.1.

In order to assess the significant difference in predictive performance the Diebold-Mariano test is applied. Based on the MAE measure the learning models do not significantly outperform the benchmark models for the shortest two forecasting horizons. In the three month horizon, the SVR significantly outperforms both GARCH models and RF and in the six month forecasting horizon the GB as well. In the one year ahead forecasting window, both the standalone SVR as well as the hybrid EGARCH-SVR model significantly outperform all benchmark models, where the SVR is the superior model. As of the two year forecasting window, all machine learning models significantly outperform both GARCH as well as both GJR-GARCH models, however this is not the case for EGARCH models. For the full test set the SVR model is also significantly better than the EGARCH.

Again, computing the Diebold-Mariano test based on the RMSE measure will not lead to different results and the test is therefore conducted based on the MSE measure. Based on this no model can be significantly favored for any forecasting horizon with a single exception; for the one year horizon both tree based model significantly outperform both EGARCH and GJR-GARCH models and the GB outperforms the EGARCH-SVR. Please refer to Appendix B.2 for the complete list of executed Diebold-Mariano significance tests.

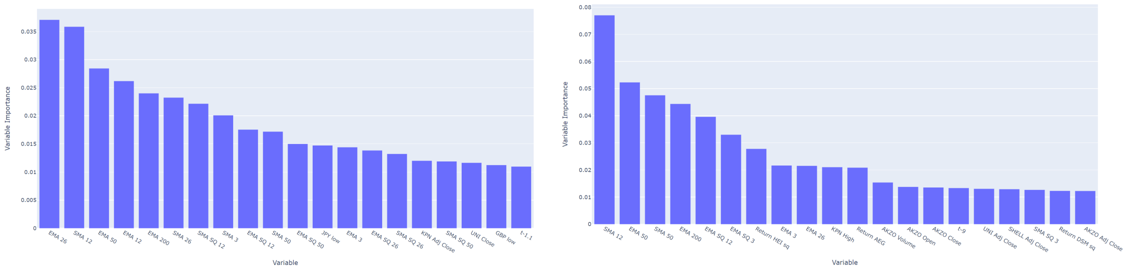
Another interesting feature of the tree based models is the possibility to examine the importance per explanatory variable through the trees variable importance measure. In Figure 9 the 20 most important variables for both tree based algorithms are presented. Both the simple moving averages as well as the exponential moving averages for especially the log return series but also the squared log return series of the AEX receive high importances. This is not a surprise since it is shown in previous literature that these moving averages can be good indicators of stock market movement and are therefore useful to include in the model (Brailsford and Faff, 1996). For random forest the first 11 most important variables are even all moving average variables, followed by variables regarding the Japanese yen and British pound sterling exchange rate as well as a variable regarding the Unibail and KPN stocks. The 20th most important variable is the first lag of the log return series.

The gradient boosting algorithm ranks the moving averages high as well, however includes more variables on stocks; variables regarding Heineken, KPN, AEGON, AKZO, Unibail, Shell and DSM appear. The gradient boosting algorithm gives a relatively high weight to the fourth (21st most important variable) and the ninth lag of the squared log return series, which coincides with the autocorrelation plot of the squared log return series where there is a clear spike in correlation visible at these lags. However, even more autocorrelation occurs at the third and eight lag, which are ranked significantly lower by both models.

An advantage of these tree based models that rank the explanatory variables based on importance is that in cases of many explanatory variables where the probability of multicollinearity rises, the models restrain the influence of these particular variables by shrinking their coefficients. Therefore some variables on the stocks are ranked low, since for all stocks included in this research multiple variables are incorporated such as high, low and close, which are likely highly correlated.

Interestingly, the macroeconomic variables on the Dutch oil market are not highly ranked by the tree based models. The CISS indicator is ranked far higher compared to all yield rates, the STEP indicator, the gold price and the effective exchange rate. This indicates that this indicator on systemic stress indeed has some predictive power for future values of volatility. Furthermore, most lagged values of the squared log return series are not ranked high. The gradient boosting algorithm applies even more regularization and shrinks the

Figure 9: Variable importance measures



In the figures the 20 most important explanatory variables based on variable importances are presented, with on the x-axis the (abbreviated) names of the explanatory variables and on the y-axis their relative importance. The left figure shows the ranking for the random forest and the right figure for the gradient boosted tree. For the complete list of explanatory variables and their abbreviations, please refer to Appendix B.3.

coefficient of 50 explanatory variables to zero, namely; some variables on the stocks, some distant lags of the squared log return series, some exchange rates, the gold price and yield curves spot and forward rates. This suggests that the gradient boosted tree tackles the multicollinearity more compared to the random forest and this coincides with its better performance.

5 Conclusion

The aim of this research was to investigate the power of three machine learning models as well as a novel hybrid model in the out-of-sample volatility forecasting of the AEX based on data of the period of January 2012 to December 2021. The proposed methods are support vector regression, random forest, gradient boosted tree and the hybrid method EGARCH-SVR. In order to assess not only their relative performance but also substantiate these findings the models are compared to the traditional statistical time series models of the GARCH class, whereof the GARCH, EGARCH and GJR-GARCH models are estimated with both the normal as well as student t distributional assumption. To check the robustness of all models regarding different levels of average volatility the out-of-sample predictions of volatility are made over multiple horizons. The models are compared with two well known performance measures and their statistical significance in terms of predictive performance is assessed. Finally, the tree based variable importance measures are discussed to evaluate if these machine learning models are indeed able to extract information out of additional explanatory variable based on stocks and macroeconomic indicators.

It is shown that amongst the traditional time series models, the EGARCH model with normally distributed returns outperforms the others which is most likely due to the allowance for leverage effect in the model. Surprisingly, the student t distributional assumption leads to a slight decrease in performance for almost all forecasts however this difference is not often significant. Regarding the forecasting horizons, it is evident that the GARCH type models are highly effected by an increase in average volatility of the AEX, leading their out-of-sample performances to decrease significantly. Furthermore, it is shown that compared to these benchmark models, the machine learning models indeed appear to better adapt to the noisiness and complex structure of the data due to their higher flexibility and non-parametric form. This leads them to often significantly outperform the benchmark models especially in the larger forecasting horizons, where the GARCH type models struggle more to grasp the spiked structure of the AEX volatility. However, the results for the hybrid EGARCH-SVR model are disappointing, as it is also highly influenced in more volatile periods of the market and is often outperformed by the standalone EGARCH models.

Furthermore, even though the aforementioned learning methods indeed increase predictive performance and often significantly outperform the benchmark models based on absolute errors, the same statistical significance can not be concluded based on squared errors. Also, the resulting best performing model differs for both measures. As a consequence, selecting a single best performing model amongst the machine learning methods is a difficult task. Overall taking both performance measures into account the gradient boosted tree is favored, however based on proven significance it is more reasonable to base a conclusion solely on the MAE measure, which would lead to selecting the support vector machine model followed by the gradient boosted tree as best performing models.

Finally, the variable importance measures of both tree based methods show that indeed the additional explanatory variables regarding the stocks the AEX consists of contain some information on the future value of the AEX volatility. Besides these variables, moving averages of the AEX volatility prove to be important indicators as well as some variables regarding exchange rates, whilst most macroeconomic variables are not selected as important drivers of the AEX volatility.

Thus, machine learning approaches indeed show good qualities to be applied to a highly nonlinear and complex time series as financial returns and are able to outperform more traditional statistical time series models. They show to be especially fitting in high volatile market times, where both the hybrid model as well as the traditional models perform less satisfactory.

5.1 Limitations and Further Research

Predicting volatility remains a difficult task as it is a non-observable variable and can not be precisely measured, but can only be extracted including a certain amount of error. The framework used in this research is based on the replacement of the unobserved variance with the demeaned squared log return series. This measure for daily volatility is a very specific one, possibly leading to poorer out-of-sample forecasts of the models. To gain a better understanding of the usefulness of machine learning and hybrid models in volatility forecasting more proxies could be applied, such as realized volatility. Furthermore, as more training data is often beneficial to both the traditional time series methods as well as the learning methods described it would be advisable to extend the historical data as much as possible. This could especially be useful for the tree based methods, that can not extrapolate from training data.

Also, since the variable importance measures of the tree based models shrunk the coefficients of some variables to (almost) zero, the models might improve when these variables are excluded. For future research it might be interesting to include many variables and apply the tree based methods to select the informative ones, and apply the models with the reduced number of variables to the AEX volatility forecasting. Another suggestion for further research is in the direction of the hybrid method, since the novel hybrid EGARCH-SVR model did not prove to perform effectively compared to either the standalone SVR model as well as the standalone EGARCH model. More research on the input data of the SVR part of the hybrid model should be conducted as well as the possibility of using other kernels.

References

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

Alpaydin, E. (2020). *Introduction to Machine Learning.* Cambridge, Massachusetts: The MIT press, 4 edition.

Andersen, T. and Bollerslev, T. (1998). Answering the skeptics: Yes, standard volatility models do provide accurate forecasts. *International Economic Review*, 39(4):885–905.

Andersen, T., Bollerslev, T., Diebold, F., and Labys, P. (2003). Modeling and forecasting realized volatility. *Econometrica*, 71(2):579–625.

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

Bezerra, P. and Albuquerque, P. (2017). Volatility forecasting via SVR-GARCH with mixture of gaussian kernels. *Computational Management Science*, 14(2):179–196.

Bishop, C. (2006). *Pattern Recognition and Machine Learning.* Springer, 1 edition.

Bollerslev, T. Engle, R. and Nelson, D. (1994). ARCH models. handbook of econometrics. *Handbook of Econometrics*, 4:2959–3038.

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

Bollerslev, T., Chou, R., and Kroner, K. (1992). ARCH modeling in finance: a review of the theory and empirical evidence. *Journal of Econometrics*, 52(1):5–59.

Boser, B., Guyon, I., and Vapnik, V. (1992). A training algorithm for optimal margin classifier. *Colt92: Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory*, pages 144–152.

Brailsford, T. and Faff, R. (1996). An evaluation of volatility forecasting techniques. *Journal of Banking and Finance*, 20(1):419–438.

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

Breiman, L., Friedman, J., Olshen, R., and Stone, C. (1984). *Classification and Regression Trees.* Chapman and Hall, 1 edition.

Cao, L. and Tay, F. (2001). Improved financial time series forecasting by combining support vector machines with self-organizing feature map. *Intelligent Data Analysis*, 5(4):339–354.

Cao, L. and Tay, F. (2003). Support vector machine with adaptive parameters in financial time series forecasting. *IEEE Transactions on Neural Networks*, 14(6):1506–1518.

Cavalcante, R., Brasileiro, R., Souza, V., Nobrega, J., and Oliveira, A. (2016). Computational intelligence and financial markets: A survey and future directions. *Expert Systems with Applications*, 55(10).

Chen, G., Rui, O., and Firth, M. (2001). The dynamic relation between stock returns, trading volume, and volatility. *The Financial Review*, 36(3):153–174.

Cheng, C. and Wei, L. (2009). Volatility model based on multi-stock index for taiex forecasting. *Expert Systems with Application*, 36(3):6187–6191.

Chih Hung, W., Gwo Hshiun, T., and Rong Ho, L. (2009). A novel hybrid genetic algorithm for kernel function and parameter optimization in support vector regression. *Expert Systems with Applications*, 36(1):4725–4735.

Christensen, K., Siggaard, M., and Veliyev, B. (2021). A machine learning approach to volatility forecasting. *Center for Research in Econometric Analysis of Time Series: Research paper.*

Cont, R. (2004). Volatility clustering in financial markets: Empirical facts and agent-based models. *In: Teyssière G., Kirman A.P. (eds) Long Memory in Economics*, pages 289–309.

Dash, R., Dash, P., and Bisoi, R. (2015). A differential harmony search based hybrid internal type2 fuzzy EGARCH model for stock market volatility prediction. *International Journal of Approximate Reasoning*, 59(3):81–104.

Dickey, D. and Fuller, W. (1979). Distribution of the estimators for autoregressive time series with a unit root. *Journal of the American Statistical Association*, 74(366):427–431.

Diebold, F. (2012). Comparing predictive accuracy, twenty years later: a personal perspective on the use and abuse of diebold-mariano tests. *National Bureau of Economic Research: Working Paper.*

Diebold, F. and Mariano, R. (1995). Comparing predictive accuracy. *Journal of Business and Economic Statistics*, 13:253–263.

Dietterich, T. (2000). An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Machine Learning*, 40(2):139–157.

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

Donaldson, R. and 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 heteroskedasticity with estimates of the variance of u.k. inflation. *Econometrica*, 50:987–10080.

Freund, Y. and Schapire, R. (1996). Experiments with a new boosting algorithm. *In: Proceedings of the Thirteenth International Conference on Machine Learning Theory*, Morgen Kaufmann, San Fransisco:148–156.

Friedman, J. (2001). Greedy function approximation: a gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232.

Gill, J. and King, G. (2004). What to do when your hessian is not invertible. *Sociological Methods and Research*, 33(1):54–87.

Glosten, L., Jagannathan, R., and Runkle, D. (1993). On the relationship between the expected value and the volatility of the nominal excess return on stocks. *Journal of Finance*, 48(5):1779–1802.

Gong, X., Liu, X., Xiong, X., and Zhuang, X. (2019). Forecasting stock volatility process using improved least square support vector machine approach. *Journal of Soft Computing*, 23(11):11867–11881.

Hamid, S. and Iqbal, Z. (2004). Using neural networks for forecasting volatility of s&p 500 index futures prices. *Journal of Business Research*, 57(10):1116–1125.

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

Harvey, D. Leybourne, S. and Newbold, P. (1997). Testing the equality of prediction mean squared errors. *International Journal of Forecasting*, 13(2):281–291.

Hastie, T., Ribshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning. Data Mining, Inference, and Prediction.* Springer, 2nd edition.

Ishwaran, H. (2015). The effect of splitting on random forests. *Machine Learning*, 99(1):75–118.

Jarque, C. and Bera, A. (1987). A test for normality of observations and regression residuals. *International Statistical Review*, 55(2):163–172.

Kara, Y., Boyacioglu, M., and Baykan, O. (2011). Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul stock exchange. *Expert Systems with Applications*, 38(5):5311–5319.

Kristjanpoller, W. and Minutolo, M. (2016). Forecasting volatility of oil price using an artificial neural network-GARCH model. *Expert Systems with Applications*, 65(1):233–241.

Lu, X., Que, D., and Cao, G. (2016). Volatility forecast based on the hybrid artificial neural network and GARCH-type models. *Procedia Computer Science*, 91(1):1044–1049.

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

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

McAleer, M. (2014). Asymmetry and leverage in conditional volatility models. *Econometrics*, 2(5):145–150.

McAleer, M. and Hafner, C. (2014). A one line derivation of egarch. *Econometrics*, 2(2):92–97.

Miranda, F. and Burgess, N. (1997). Modelling market volatilities: the neural network perspective. *International Journal of Phytoremediation*, 21(1):137–157.

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

Monfared, S. and Enke, D. (2014). Volatility forecasting using a hybrid GJR-GARCH neural network model. *Procedia Computer Science*, 36(1):246–253.

Nabipour, M., Nayyeri, P., Jabani, H., and Mosavi, A. (2020). Deep learning for stock market prediction. *Entropy (Basel)*, 22(8):840.

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

Nugroho, D., Priyono, A., and Susanto, B. (2021). Skew normal and skew student-t distributions on GARCH(1,1) model. *Media Statistika*, 14(1):21–32.

Patton, A. (2006). Volatility forecast comparison using imperfect volatility proxies. *University of Technology Quantitative Finance Research Centre: Research Paper*, 175:1–45.

Peng, Y., Albuquerque, P., Camboim de Sa, J., Padula, A., and Montenegro, M. (2018). The best of two worlds: Forecasting high frequency volatility for cryptocurrencies and traditional currencies with support vector regression. *Expert Systems with Applications*, 97(1):177–192.

Perez-Cruz, F., Afonso-Rodriguez, J., and Giner, J. (2003). Estimating GARCH models using SVM. *Quantitative Finance*, 3(1):163–172.

Qu, H. and Zhang, Y. (2016). A new kernel of support vector regression for forecasting high-frequency stock returns. *Mathematical Problems in Engineering, Research Paper*, 1:9.

Radovic, O. and Stankovic (2015). Tail risk assessment using support vector machine. *Journal of Engineering Science and Technology Review*, 8(1):61–64.

Roh, T. (2007). Forecasting the volatility of stock price index. *Expert Systems with Applications*, 33(1):916–922.

Ruping, S. (2001). SVM kernels for time series analysis. *Technical report, Universitat Dortmund*, (43).

Ryll, L. and Seidens, S. (2019). Evaluating the performance of machine learning algorithms in financial market forecasting: A comprehensive survey. *Computational Finance*, 1(1).

Santamaria-Bonfil, G., Vazquez-Rodarte, I., and Fausto-Solis, J. (2013). Volatility forecasting using support vector regression and a hybrid genetic algorithm. *Computational Economics*, 45(1):111–133.

Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics*, 6(2):461–464.

Sun, H. and Yu, B. (2020). Forecasting financial returns volatility: a GARCH-SVR model. *Computational Economics*, 55(1):451–471.

Thadewald, T. and Buning, H. (2004). *Jarque-Bera test and its competitors for testing normality - A power comparison.* School of Business and Economics, Free University of Berlin.

Theofilatos, K., Likothanassis, S., and Karathanasopoulos, A. (2012). Modeling and trading the EUR/USD exchange rate using machine learning techniques. *Engineering, Technology and Applied Science Research*, 2(5):269–272.

Tsay, R. (2002). *Analysis of Financial Time Series.* New York: Wiley, 3rd edition.

Vapnik, V. et al. (1997). Predicting time series with support vector machines. *International Conference on Artificial Neural Networks*, 1327:999–1004.

Vilder, R. and Visser, M. P. (2007). Proxies for daily volatility. *PSE Working Papers, HAL.*

Wang, T. and Huang, Z. (2012). The relationship between volatility and trading volume in the chinese stock market: A volatility decomposition perspective. *Annals of Economics and Finance*, 13(1):211–236.

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

Yuan, Y. (2013). Forecasting the movement direction of exchange rate with polynomial smooth support vector machine. *Mathematical and Computer Modelling*, 57(3-4):932–944.

A Additional Methodology

A.1 Distributions

Let $p(r_t)$ be the probability density function of the returns on time t, independent of the previous returns r_{t-1}. Then, the normal distribution and the student t distribution follow the probability density function of eq. (23) and eq. (24) respectively.

$$p(r_t) = (2\pi\sigma_t^2)^{-1/2} \exp\left(-\frac{(r_t - \mu_t)^2}{2\sigma_t^2}\right) \tag{23}$$

$$p(r_t) = \frac{\Gamma((\nu+1)/2)}{\Gamma(\nu/2)\sqrt{\pi\sigma_t^2(\nu-2)}} \left(1 + \frac{(r_t - \mu_t)^2}{(\nu-2)\sigma_t^2}\right)^{-\frac{\nu+1}{2}} , \nu > 0 \tag{24}$$

Where $\Gamma(\nu) = \int_0^{\inf} e^{-x} x^{\nu-1} dx$ is the gamma function, and ν parameter of the thickness of the tail.

A.2 CART Algorithm

Algorithm 4: CART algorithm for decision trees.

```
Data: Data set $(x_t, \epsilon_t^2)$ with dependent series $\epsilon_t^2$ for $t = 1, ..T$ and features $x_{ti}$ for $t = 1, ..T$,
  $i = 1, .., p$.
    1. Take a root node with the complete data set.
    2. Find the best split points $s$ that maximizes the splitting criterion for all features.
    3. Find among the pairs of features and best splits $(i, s)$ from step 2 the best pair that maximizes the
       splitting criterion.
    4. Split the current node on this split.
    5. Repeat the process from step 2, until the stopping criteria is reached or the tree is fully grown.
```

A.3 Augmented Dickey-Fuller Test

The augmented Dickey-Fuller (ADF) test is a unit-root test designed to test the stationairity of a time series (Dickey and Fuller, 1979). Let y_t the value of the series in time t and consider a simple autoregressive model (AR):

$$y_t = \rho y_{t-1} + \epsilon_t \tag{25}$$

Where the null hypothesis (H_0) in this test is the case that coefficient ρ is equal to one; there is a unit root present in the considered series and therefore the series is non-stationary. The alternative hypothesis (H_A) states that the tested series has no unit roots and therefore is stationary. Next, the first order difference of eq. (25) is taken:

$$\Delta y_t = \phi y_{t-1} + \epsilon_t \tag{26}$$

The null hypothesis (H_0) becomes $\phi = 0$ which is equal to testing $\rho = 1$ and states that there is a unit root present in the considered time series. Under the alternative hypothesis (H_A), $\phi < 1$ and the series follow a stationary process. The null hypothesis of non-stationairity is rejected if the p-value is lower then 0.05 at the $\alpha = 5\%$ significance level. This discussed p-value is based on MacKinnons p-value (MacKinnon, 1994).

A.4 Jarque-Bera Test

The Jarque-Bera test for normality is used to test if a data set is normally distributed by testing simultaneously if the skewness of the data is equal to zero and the kurtosis of the data is equal to three (Jarque and Bera, 1987). Under the null hypothesis (H_0) the data is normally distributed, under the alternative hypothesis (H_A) it is not normally distributed. The Jarque-Bera test statistic is defined as in eq. (27).

$$JB = N\left(\frac{(\sqrt{b_1})^2}{6} + \frac{(b_2 - 3)^2}{24}\right) \tag{27}$$

Where N the number observations, b_1 skewness coefficient and b_2 the kurtosis coefficient. The test statistic can be compared with a critical value from the χ^2 distribution, where the number of degrees of freedom is two. If the measured test statistic results in a higher value than the value of $\chi^2_{(2)}$, the null hypothesis is rejected and therefore the tested data does not follow a normal distribution. The critical value for multiple significance levels can be obtained from a χ^2 table. It should be mentioned that the Jarque-Bera test does not perform well in small samples and furthermore loses power when applied to non-symmetric distributions that do not have long tails. In such cases the Shapiro-Wilk test could be a good substitute (Thadewald and Buning, 2004).

A.5 Information Criteria

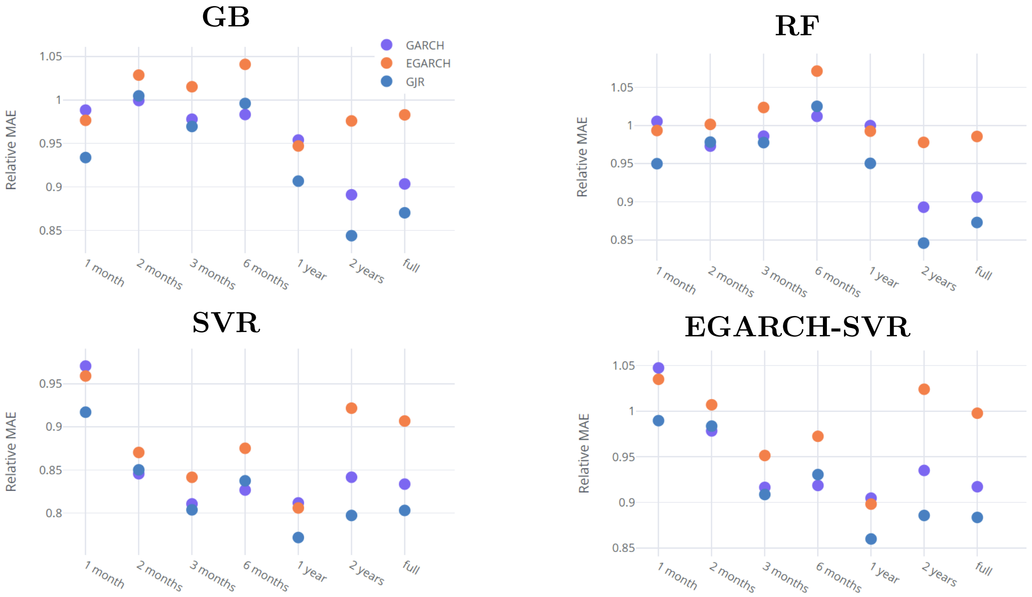
The first measures to compare relative prediction performance of the statistical time series models are the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC) (Akaike, 1974; Schwarz, 1978). These information criteria provide a method to select the model that is relatively the best fitted model on the training data. The AIC and the BIC criteria are designed to compare statistical models and therefore will be used in this research to determine which of the benchmark models fits best to the data and therefore is expected to produce the best out-of-sample forecasts as well. The AIC and the BIC are calculated as below, where L is the models maximum value of the likelihood function, n the number of observations and k the number of parameters to be estimated in the model and a lower value for these criteria is preferred. The BIC is a similar measure as the AIC, however it incorporates a penalty for an extra added parameter.

$$\begin{aligned} AIC &= 2k - 2\ln(L) \\ BIC &= \ln(n)k - 2\ln(L) \end{aligned} \tag{28}$$

B Results

B.1 Model Comparisons

Figure 10: Relative performance of the learning models to the benchmark models based on MAE



GB

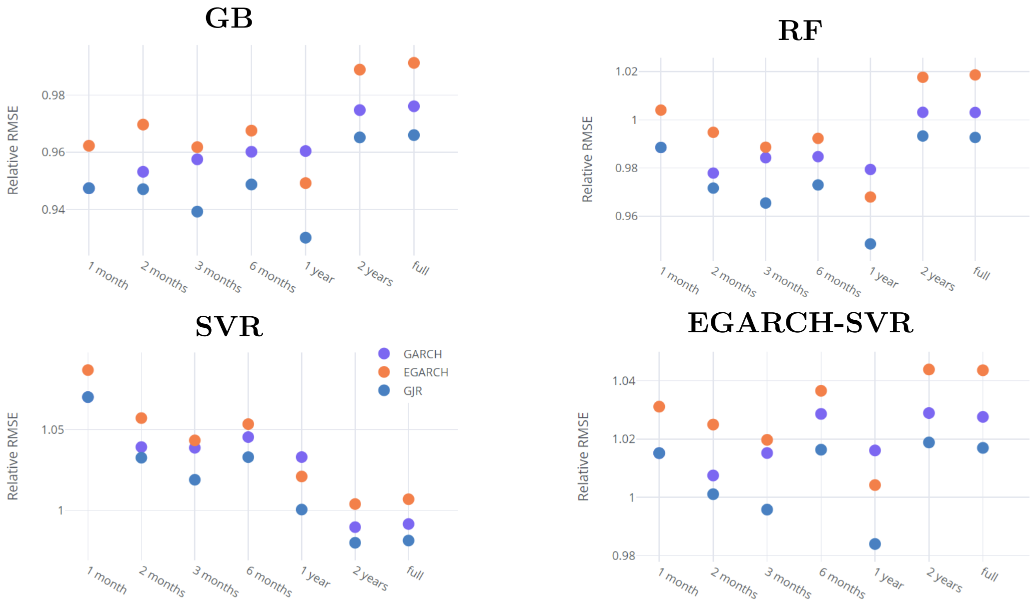
RF

SVR

EGARCH-SVR

In the figure the MAE values are presented for the three machine learning models as well as the hybrid model relative to the benchmark models with student t distributional assumption. A relative value below one indicates that the learning model performs better compared to the benchmark model.

Figure 11: Relative performance of the learning models to the benchmark models based on RMSE



In the figure the RMSE values are presented for the three machine learning models as well as the hybrid model relative to the benchmark models with student t distributional assumption. A relative value below one indicates that the learning model performs better compared to the benchmark model.

B.2 Diebold-Mariano

In the tables the Diebold-Mariano test statistics are presented for all considered models for all forecasting horizons. A positive value indicates that the model in the column performs better that its row counterpart. Performance is significant at the 5% and 1% level for *, ** respectively.

	GARCH (st)		EGARCH (n)		EGARCH (st)		GJR (n)		GJR (st)		SVR		RF		GB		EGARCH-SVR	
DM - 1 month	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*
GARCH (n)	-1.07	-0.33	-0.35	0.67	-0.44	0.61	-0.76	0.26	-1.28	-0.01	0.16	-1.28	-0.23	0.32	0.01	0.72	-0.65	-0.44
GARCH (st)			-0.14	0.77	-0.26	0.69	-0.56	0.30	-1.18	0.00	0.24	-1.18	-0.08	0.34	0.13	0.75	-0.52	-0.44
EGARCH (n)					-0.69	-0.83	-0.42	-0.65	-1.29	-0.99	0.27	-1.21	0.01	-0.28	0.26	0.69	-0.51	-1.04
EGARCH (st)							-0.29	-0.60	-1.27	-0.96	0.31	-1.19	0.12	-0018	0.35	0.71	-0.46	-0.98
GJR-GARCH (n)									-2.24*	-0.60	0.39	-1.13	0.31	0.24	0.43	0.72	-0.34	-0.53
GJR-GARCH (st)											0.64	-0.99	0.91	0.40	0.89	0.84	0.15	-0.43
SVR													-0.30	1.17	-0.13	1.15	-0.69	0.86
RF															0.34	0.94	-0.51	-0.72
GB																	-0.52	-1.05

	GARCH (st)		EGARCH (n)		EGARCH (st)		GJR (n)		GJR (st)		SVR		RF		GB		EGARCH-SVR	
DM - 2 months	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*
GARCH (n)	-2.12*	-0.47	0.28	0.69	0.31	0.54	-0.02	-0.05	-0.11	-0.26	1.57	-0.98	0.39	0.92	-0.21	0.89	0.14	-0.28
GARCH (st)			0.60	0.82	0.60	0.64	0.22	0.01	0.09	-0.22	1.63	-0.88	0.72	1.07	0.01	0.96	0.35	-0.24
EGARCH (n)					0.42	-0.74	-0.52	-1.66	-0.54	-1.68	1.22	-0.96	0.07	0.20	-0.56	0.84	-0.06	-1.07
EGARCH (st)							-0.80	-1.53	-0.79	-1.75	1.15	-0.90	-0.04	0.36	-0.59	0.91	-0.13	-1.01
GJR-GARCH (n)									-0.47	-0.86	1.21	-0.66	0.31	1.21	-0.18	1.17	0.19	-0.33
GJR-GARCH (st)											1.19	-0.51	0.36	1.29	-0.07	1.38	0.29	-0.05
SVR													-1.43	1.13	-1.41	1.03	-1.44	0.55
RF															-0.72	0.73	-0.11	-1.21
GB																	0.30	-1.22

	GARCH (st)		EGARCH (n)		EGARCH (st)		GJR-GARCH (n)		GJR-GARCH (st)		SVR		RF		GB		EGARCH-SVR	
DM - 3 months	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*
GARCH (n)	-2.00*	-0.38	0.55	0.26	0.56	0.11	-0.16	-0.45	-0.31	-0.58	2.21*	-0.97	0.12	0.77	0.25	0.88	1.26	-0.53
GARCH (st)			0.86	0.33	0.83	0.17	0.05	-0.44	-0.15	-0.58	2.24*	-0.88	0.46	0.91	0.49	0.96	1.47	-0.51
EGARCH (n)					0.45	-0.70	-1.22	-1.40	-1.19	-1.30	1.60	-0.79	-0.54	0.54	-0.28	1.03	1.08	-0.93
EGARCH (st)							-1.71	-1.41	-1.58	-1.36	1.50	-0.70	-0.57	0.65	-0.35	1.15	1.02	-0.82
GJR-GARCH (n)									-0.90	-0.93	1.71	-0.43	0.24	1.29	0.38	1.52	1.61	-0.15
GJR-GARCH (st)											1.73	-0.28	0.40	1.31	0.53	1.75	1.73	0.17
SVR													-2.19*	1.06	-1.79	1.02	-1.24	0.42
RF															0.24	0.82	1.38	-1.28
GB																	1.04	-1.32

	GARCH (st)		EGARCH (n)		EGARCH (st)		GJR-GARCH (n)		GJR-GARCH (st)		SVR		RF		GB		EGARCH-SVR	
DM - 6 months	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*
GARCH (n)	-1.94	-0.45	1.78	0.53	1.45	0.29	0.45	-0.23	0.13	-0.44	2.68**	-1.27	-0.79	0.78	0.27	0.92	1.54	-1.21
GARCH (st)			2.11*	0.62	1.74	0.37	0.70	-0.19	0.32	-0.43	2.70**	-1.16	-0.48	0.92	0.49	1.00	1.70	-1.17
EGARCH (n)					-0.32	-0.85	-1.56	-0.98	-1.61	-1.04	1.69	-1.22	-2.62**	0.27	-1.33	0.78	0.59	-1.87
EGARCH (st)							-1.75	-0.90	-1.83	-1.02	1.64	-1.08	-2.24*	0.48	-1.15	0.89	0.64	-1.70
GJR-GARCH (n)									-1.34	-1.08	1.86	-0.75	-0.98	1.09	-0.18	1.35	1.25	-1.01
GJR-GARCH (st)											1.91	-0.57	-0.60	1.22	0.09	1.60	1.43	-0.62
SVR													-2.98**	1.31	-2.28**	1.21	-1.64	0.40
RF															1.15	0.85	2.06*	-1.91
GB																	1.30	-1.55

	GARCH (st)		EGARCH (n)		EGARCH (st)		GJR-GARCH (n)		GJR-GARCH (st)		SVR		RF		GB		EGARCH-SVR	
DM - 1 year	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*
GARCH (n)	-2.78**	-0.66	-0.32	-0.65	-0.58	-0.84	-1.63	-1.26	-1.75	-1.34	4.30**	-1.50	-0.38	1.48	1.61	1.61	2.68**	-0.79
GARCH (st)			0.02	-0.59	-0.31	-0.81	-1.49	-1.30	-1.64	-1.37	4.33**	-1.36	0.01	1.67	1.91	1.76	2.94**	-0.76
EGARCH (n)					-1.32	-1.13	-2.50*	-1.41	-2.46*	-1.46	3.81**	-0.88	-0.01	2.28*	1.79	2.17*	3.35**	-0.50
EGARCH (st)							-2.61**	-1.38	-2.64**	-1.50	3.75**	-0.63	0.29	2.46*	1.93	2.46*	3.62**	-0.24
GJR-GARCH (n)									-1.79	-1.48	3.94**	-0.23	1.30	2.67**	2.65**	3.12**	4.20**	0.44
GJR-GARCH (st)											3.96**	-0.01	1.51	2.56*	2.77**	3.22**	4.38**	0.83
SVR													-4.52**	1.74	-3.24**	1.73	-2.18*	0.48
RF															2.74**	1.24	3.02**	-1.71
GB																	1.51	-2.11*

	GARCH (st)		EGARCH (n)		EGARCH (st)		GJR-GARCH (n)		GJR-GARCH (st)		SVR		RF		GB		EGARCH-SVR	
DM - 2 years	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*
GARCH (n)	-3.29**	-0.17	2.97**	1.34	2.52*	0.89	-1.48	-0.04	-1.47	-0.19	2.96**	0.25	2.43*	-0.16	2.70**	1.40	1.39	-0.91
GARCH (st)			3.29**	1.49	2.84**	1.00	-1.29	-0.03	-1.31	-0.19	3.05**	0.26	2.58**	-0.13	2.89**	1.41	1.68	-0.92
EGARCH (n)					-1.12	-0.58	-3.36**	-0.44	-3.33**	-0.55	1.48	-0.19	0.51	-0.78	0.70	0.57	-1.02	-1.56
EGARCH (st)							-3.31**	-0.41	-3.34**	-0.53	1.53	-0.12	0.66	-0.62	0.88	0.65	-0.90	-1.57
GJR-GARCH (n)									-1.17	-0.92	2.86**	0.21	2.54*	-0.03	2.97**	0.55	2.62**	-0.61
GJR-GARCH (st)											2.88**	0.33	2.58**	0.10	3.00**	0.63	2.76**	-0.39
SVR													-2.04*	-0.33	-1.38	0.41	-1.57	-0.76
RF															0.10	1.42	-0.99	-0.69
GB																	-1.18	-1.57

	GARCH (st)		EGARCH (n)		EGARCH (st)		GJR-GARCH (n)		GJR-GARCH (st)		SVR		RF		GB		EGARCH-SVR	
DM - full set	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*	*MAE*	*MSE*
GARCH (n)	-3.72**	-0.18	3.47**	1.42	2.96**	0.96	-1.28	-0.06	-1.32	-0.21	3.90**	0.21	2.68**	-0.16	2.97**	1.37	2.41*	-0.89
GARCH (st)			3.84**	1.59	3.33**	1.08	-1.05	-0.05	-1.12	-0.21	3.99**	0.22	2.86**	-0.13	3.18**	1.38	2.76**	-0.90
EGARCH (n)					-0.91	-0.58	-3.51**	-0.50	-3.47**	-0.60	2.27*	-0.29	0.42	-0.81	0.62	0.42	-0.07	-1.59
EGARCH (st)							-3.46**	-0.47	-3.48**	-0.58	2.26*	-0.22	0.53	-0.65	0.75	0.51	0.09	-1.60
GJR-GARCH (n)									-1.23	-0.93	3.44**	0.19	2.58**	-0.01	3.01**	0.55	3.37**	-0.58
GJR-GARCH (st)											3.43**	0.32	2.62**	0.11	3.05**	0.63	3.50**	-0.36
SVR													-3.50**	-0.29	-2.39*	0.44	-1.76	-0.73
RF															0.16	1.40	-0.32	-0.66
GB																	-0.45	-1.53

B.3 Explanatory Variables

Table 9: List of explanatory variables of included stocks and their abbreviations as used in text

Stock	*Variable Name*	*Abbreviation*	***Stock***	*Variable Name*	*Abbreviation*
Aegon	High	AEG High	Koninklijke Ahold Delhaize N.V.	High	AHOLD High
	Low	AEG Low		Low	AHOLD Low
	Close	AEG Close		Close	AHOLD Close
	Adjusted Close	AEG Adj Close		Adjusted Close	AHOLD Adj Close
	Volume	AEG Volume		Volume	AHOLD Volume
	Log return	Return AEG		Log return	Return AHOLD
	Squared log return	Return AEG sq		Squared log return	Return AHOLD sq
Akzo Nobel	High	AKZO High	ArcelorMittal SA	High	ARC High
	Low	AKZO Low		Low	ARC Low
	Close	AKZO Close		Close	ARC Close
	Adjusted Close	AKZO Adj Close		Adjusted Close	ARC Adj Close
	Volume	AKZO Volume		Volume	ARC Volume
	Log return	Return AKZO		Log return	Return ARC
	Squared log return	Return AKZO sq		Squared log return	Return ARC sq
ASML Holding	High	ASML High	Royal DSM N.V.	High	DSM High
	Low	ASML Low		Low	DSM Low
	Close	ASML Close		Close	DSM Close
	Adjusted Close	ASML Adj Close		Adjusted Close	DSM Adj Close
	Volume	ASML Volume		Volume	DSM Volume
	Log return	Return ASML		Log return	Return DSM
	Squared log return	Return ASML sq		Squared log return	Return DSM sq
Heineken N.V.	High	HEI High	ING Groep N.V.	High	ING High
	Low	HEI Low		Low	ING Low
	Close	HEI Close		Close	ING Close
	Adjusted Close	HEI Adj Close		Adjusted Close	ING Adj Close
	Volume	HEI Volume		Volume	ING Volume
	Log return	Return HEI		Log return	Return ING
	Squared log return	Return HEI sq		Squared log return	Return ING sq
Koninklijke KPN N.V.	High	KPN High	Randstad N.V.	High	RDS High
	Low	KPN Low		Low	RDS Low
	Close	KPN Close		Close	RDS Close
	Adjusted Close	KPN Adj Close		Adjusted Close	RDS Adj Close
	Volume	KPN Volume		Volume	RDS Volume
	Log return	Return KPN		Log return	Return RANDST
	Squared log return	Return KPN sq		Squared log return	Return RANDST sq
Shell PLC	High	SHELL High	Unibail-Rodamco-Westfield SE	High	UNI High
	Low	SHELL Low		Low	UNI Low
	Close	SHELL Close		Close	UNI Close
	Adjusted Close	SHELL Adj Close		Adjusted Close	UNI Adj Close
	Volume	SHELL Volume		Volume	UNI Volume
	Log return	Return SHELL		Log return	Return UNIBAIL
	Squared log return	Return SHELL sq		Squared log return	Return UNIBAIL sq
Wolters Kluwer	High	WOLT High	AEX	Lagged log return (t-1)	t-1.1
	Low	WOLT		Lagged squared log return (t-28 .. t-1)	t-28 .. t-1
	Close	WOLT Close			
	Adjusted Close	WOLT Adj Close			
	Volume	WOLT Volume			
	Log return	Return WOLTERS			
	Squared log return	Return WOLTERS sq			

Table 10: List of additional explanatory variables and their abbreviations as used in text

Simple Moving Average	*Variable*	*Abbreviation*	***Exponential Moving Average***	*Variable*	*Abbreviation*
Log Return	3 days	SMA 3	Log Return	3 days	EMA 3
	12 days	SMA 12		12 days	EMA 12
	26 days	SMA 26		26 days	EMA 26
	50 days	SMA 50		50 days	EMA 50
	200 days	SMA 200		200 days	EMA 200
Squared Log Return	3 days	SMA 3 SQ	Squared Log Return	3 days	EMA 3 SQ
	12 days	SMA 12 SQ		12 days	EMA 12 SQ
	26 days	SMA 26 SQ		26 days	EMA 26 SQ
	50 days	SMA 50 SQ		50 days	EMA 50 SQ
	200 days	SMA 200 SQ		200 days	EMA 200 SQ

Exchange Rate	*Variable*	*Abbreviation*	***Exchange Rate***	*Variable*	*Abbreviation*
United States Dollar to Euro	Open	USD open	British Pound Sterling to Euro	Open	GBP open
	High	USD high		High	GBP high
	Low	USD low		Low	GBP low
	Close	USD close		Close	GBP close
	Adjusted Close	USD adj close		Adjusted Close	GBP adj close
Japanese Yen to Euro	Open	JPY open			
	High	JPY high			
	Low	JPY low			
	Close	JPY close			
	Adjusted Close	JPY adj close			

Yield Curve	*Variable*	*Abbreviation*	***Yield Curve***	*Variable*	*Abbreviation*
Yield Curve Spot Rate	1-year maturity	Yield Spot 1 yr	Yield Curve Forward Rate	1-year maturity	Yield Forward 1 yr
	2-year maturity	Yield Spot 2 yr		2-year maturity	Yield Forward 2 yr
	5-year maturity	Yield Spot 5 yr		5-year maturity	Yield Forward 5 yr
	10-year maturity	Yield Spot 10 yrs		10-year maturity	Yield Forward 10 yrs

Oil Prices	*Variable*	*Abbreviation*
Dutch Oil prices	Benzine Euro95	Benzine
	Diesel	Diesel
	LPG	LPG gas

Additional Indicators	*Variable*	*Abbreviation*
	Nominal Effective Exchange Rate	EER-19
	Gold Price	Gold price
	Composite Indicator of Systemic Stress	logCISS
	Short Term European Paper	logSTEP