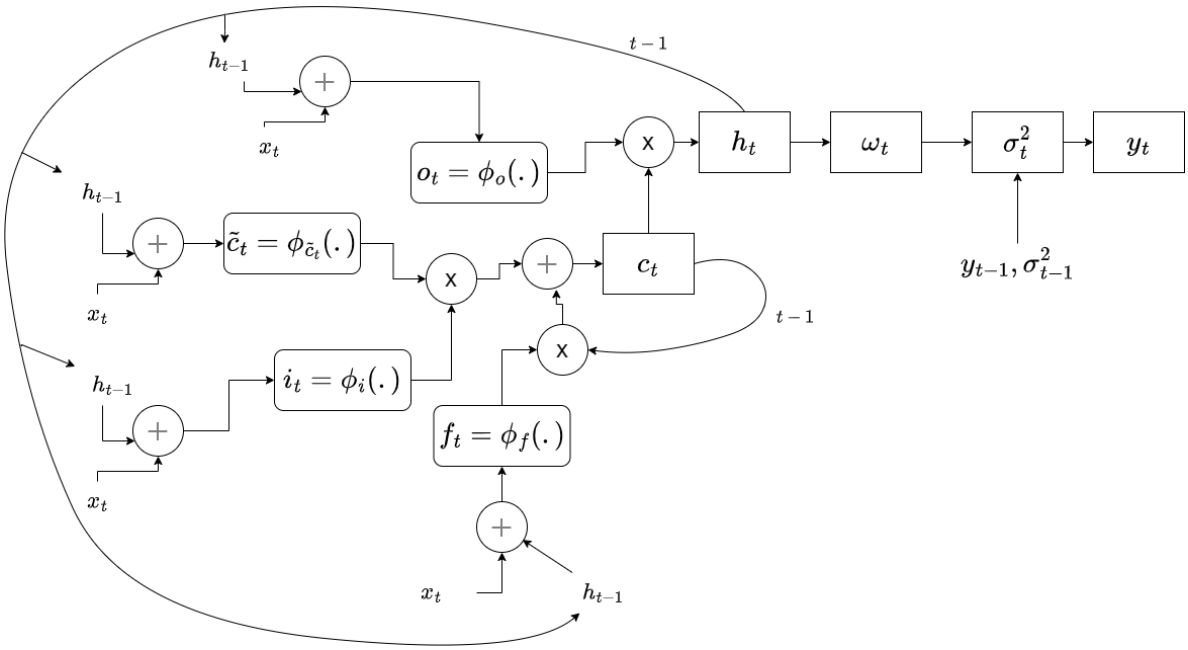
UNIVERSITY OF COPENHAGEN
DEPARTMENT OF ECONOMICS



Master Thesis

Moritz Bilstein (wmj863)

Recurrent heteroskedastic models for financial time series: Estimation, properties, and forecasting

Character Count: 81,432 (34 standard pages)

Supervisor: Rasmus Søndergaard Pedersen

Submitted on: 9 June 2023

Abstract

This thesis extends the literature on RECH models. Those models are hybrids of recurrent neural networks (RNNs) and GARCH-type models, proposed by Nguyen et al. (2022). The authors propose a general class of RECH models and provide an empirical application for one specific RECH model, the SRN-GARCH, which consists of the neuron of a simple recurrent network (SRN) and a GARCH-type model. They show superior in-sample and out-of-sample performances of RECH models in comparison to GARCH models for stock exchange indices.

The novel class of RECH models is embedded in the volatility modelling literature. For once, they extend the rich literature of GARCH-type models. here, they build upon previous efforts to incorporate neural networks in GARCH models. Related to this, several ensemble methods have previously been proposed which combine GARCH model outputs (recurrent) in neural networks. Furthermore, machine learning methods have been widely employed in volatility forecasting disregarding GARCH models. In that regard are RECH models a consequential attempt to combine the effectiveness and simplicity of GARCH-type models with the superb forecasting capabilities of (recurrent) neural networks.

A first extension is presented by proposing the incorporation of different types of neuron architectures. Those are the minimal gated unit (MGU) and long short-term memory (LSTM) neurons, where an LSTM-GARCH has recently been proposed by Liu et al. (2023) in a working paper. Those two neuron architectures are more complex than the SRN neuron. Additionally, the stochastic properties of RECH models are investigated. Stationarity conditions for the SRN-GARCH are derived and a prospect is given for the derivation of stationarity conditions of general RECH models. The stationarity conditions are applied to simulate an SRN-GARCH process and investigate parameter identification.

The performance of the RECH models is extensively evaluated in simulation studies and empirical analyses. The code for this can be found on `https://github.com/mbilsen/RECH_models`. GARCH processes of varying degrees of non-linearity are simulated and estimated using RECH models and benchmark GARCH-type models. This is done to analyse the capacity of RECH models to model non-linear financial time series.

The forecasting capabilities are analysed in two ways in an empirical analysis. The data considered spans from 2005 to October 2020 and contains the international Covid-19 outbreak and the following financial market volatility. The S&P 500 index and the assets listed in the Dow Jones Industrial Average index are examined. The conditional variance forecasts are used to predict the realised variance for the out-of-

sample period from January 2017. The forecasts are computed simulation-based and cover daily, weekly, and monthly forecasting horizons. From the prediction errors, model confidence sets (MCSs) are computed which lets us interpret the forecasting precision in a meaningful way. Furthermore, relative errors are taken into account to display the average error distribution relative to the GARCH(1,1). To put the forecasting precision into an economic context, a portfolio management application is conducted. Based on the conditional variance, an agent generates mean-variance efficient portfolios for each model which are evaluated based on the Sharpe ratio they produce.

The simulation studies are based on the simulation studies of Nguyen et al. (2022). A very important distinction is, that in this thesis multiple simulations instead of only one are performed to obtain robust results. The results of the simulation study are unexpected. Although the RECH models are designed to model non-linear time series, they cannot manage to outperform simple benchmark models with no or very limited capacities to model non-linearities. The RECH models do, however, produce realised variance forecasts that are more precise than those of benchmark models in the full sample. They especially excel at modelling the variance in the disruptive periods of March 2020 during the Covid-19 outbreak. In this period, for most assets, at least one of the RECH models generates significantly lower forecast errors than the other considered models on a confidence level of 5%. This result is robust to the choice of forecasting horizon and loss function. Here the SRN-GARCH performs best, followed by the LSTM-GARCH. The MGU-GARCH produces forecasts with losses varying much more than those of the other RECH models.

The portfolio application does not lead to the same results. The portfolios based on the conditional variance forecasts of GARCH-type models perform slightly better in the full sample as well as in a subsample of low volatility (Jan. 2020). For the month of March 2020, however, the RECH models produce portfolios with higher Sharpe ratios, with the MGU-GARCH performing best.

Two key conclusions can be drawn from this: RECH models perform best in high volatility periods and RECH models perform better at the modelling of intra-day return variance rather than inter-day return variance, which is the base of the portfolio application.

Contents

1 Introduction

Modelling and estimating volatility is an enormous part of financial econometrics. Return time series of commodities or stocks have in common that they exhibit certain stylized facts. Those are volatility clustering, the fact that drastic price changes are followed by drastic price changes, fat tails in the distribution, which means that extreme events or outliers occur more frequently than would be expected under a normal distribution, and the existence of an asymmetric effect of past positive and negative returns on future return volatility. Returns exhibit little to no correlation and are almost impossible to predict short-term (Chen et al. (2023), Gu et al. (2020)). Absolute or squared returns, however, show very high autocorrelation up to very high degrees. Hence, volatility is predictable to a certain extent. Given its significance in asset pricing, portfolio allocation, and risk management, volatility forecasting has garnered considerable attention.

The fundamental autoregressive conditional heteroskedasticity (ARCH) model for return volatility was proposed by Engle (1982) and has been extremely popular since, especially in its generalised form, the GARCH model, developed by Bollerslev (1986). Since then, a multitude of extensions has been developed to further improve the modelling and forecasting capacities of GARCH models, both linear and non-linear.

In the last years, the use of recurrent neural networks (RNNs) became increasingly popular for processing sequential data. A novel extension of GARCH models has been proposed by Nguyen et al. (2022) which is supposed to combine the features of GARCH models and the enormous performance capabilities of RNNs. Those recurrent conditional heteroskedasticity (RECH) models add a certain recurrent component to the conditional variance equation of a GARCH model. Those recurrent components are designed to be flexible and provide significant predictive improvements to GARCH models through the incorporation of non-linearity. Another claim is the conservation of interpretability of GARCH models in contrast to pure RNNs.

The authors propose a general model that allows different types of recurrent components to be used but only put one to use, the simple recurrent network (SRN) neuron. They use a Bayesian Markov chain Monte Carlo method to estimate their models. The models are evaluated using simulated data and stock index data.

This thesis aims to extend the literature on RECH models. To do this, functional forms that were originally proposed by the authors but not applied in the paper are considered and analysed, namely the minimal gated unit (MGU) and long short-term memory (LSTM), where the LSTM extension is not a novel extension but has

been implemented in the preprint Liu et al. (2023). Maximum likelihood estimation is considered as an alternative estimation method. All estimations shown in this thesis are estimated with it. This is supposed to extend the literature on RECH models by providing a simple estimation technique. Stationarity conditions are derived for the SRN-GARCH, which allow the simulation of stationary SRN-GARCH processes. This simulation study is conducted to see whether the SRN-GARCH can estimate the parameters of a stationary SRN-GARCH process and identify the true parameters correctly.

A considerable part of this study is the review of the RECHs modelling performance. This is done both in-sample and out-of-sample using different kinds of data. In-sample, with simulated data generated by GARCH-processes of varying degrees of non-linearity. Out-of-sample, with data on the Dow Jones assets as well as the S&P 500 index. The out-of-sample forecasting exercise is evaluated in multiple ways: Conditional variance forecasts of daily, weekly, and monthly horizon are compared to actual realised variance. Those losses are used to compute model confidence sets (MCSs), giving further intel on the forecasting precision. Additionally, to review the economic significance, the conditional variance forecasts of each model are used to construct mean-variance efficient portfolios.

2 Literature Review

There are two subfields in GARCH-neural network hybrid models. Firstly, there are models which synthesise a GARCH-type model with a neural network into one merged model. In those models, all parameters are estimated simultaneously. Initially, Donahoe and Dorsel (1997) proposed a GARCH extension which combined a GJR-GARCH with neural networks taking past returns as inputs. The authors demonstrate the model's versatility in modelling various kinds of non-linearities in volatility. They also report superiority in modelling four international stock market indices over the non-linear GARCH models GJR and EGARCH. The idea of inserting neural networks in the GARCH equations was picked up by Nguyen et al. (2022). They propose a class of recurrent conditional heteroskedasticity (RECH) models. An important distinction between the RECH models and the ANN-GARCH models is that the RECH models allow for the use of recurrent neural networks instead of feedforward neural networks. In this kind of application, this leads to two differences: the inclusion of past neural network outputs as neural network inputs and the use of a class of neuron architectures that are used in the recurrent neural network literature. Those can be the simple recurrent neuron, which differs from the neurons used in a regular neural network only through the inclusion of past outputs as inputs or more sophisticated architectures like the neuron of a long short-term memory (LSTM) network. The RECH models also explicitly allow for exogenous variables as inputs for the neural network component (also referred to as the recurrent component), whereas Donahoe and Dorsel (1997) only allow for past returns as inputs.

Secondly, there are hybrid models which build an ensemble of a neural network and a GARCH-type model. Those models estimate the parameters of the two-component models separately. As a method to combine volatility forecasts Donaldson and Kamstra (1996) use neural networks to combine the volatility forecasts of a moving average and a GARCH model. They find it to be superior to any linear combinations of the two forecasts and each forecast on its own. Several studies were published in *Expert Systems with Applications*: Roh (2007) use GARCH and EGARCH outputs as feed-forward neural network inputs to enhance the monthly volatility forecasting capabilities of a solitary feed-forward neural network and report that a neural-network-EGARCH-hybrid lowers the MAE by 29.43% over their sample of the KOSPI200. The authors of Hajizadeh et al. (2012) extend this and propose simulating multiple time series from GARCH estimates to use as neural network inputs. They report that this method improves the realized volatility forecasts even more for their S&P 500 sample for all considered performance measures.

Instead of feed-forward neural networks, Kim and Won (2018) used recurrent neural networks in the form of the LSTM, using Maknickienė and Maknickas (2012) as a reference who report superior performance of the LSTM compared to feed-forward neural networks in exchange rate predictions.

Neural networks have also been analysed as stand-alone variance forecasting models. The study of Brooks (1998) considers feed-forward neural networks to predict stock index volatility. The neural network places among the best forecasting models but is discarded due to the heavy computation time especially when performing rolling re-estimation. It is worth highlighting that the multiple-node neural network could not outperform an autoregressive model of order one with weekday dummy variables. Stronger results are found by Rodikov and Antulov-Fantulin (2022) who use recurrent neural networks in the form of the LSTM instead. In their unpublished study, the authors compare realized volatility forecasting accuracy based on three individual stocks, the S&P 500 index, and cryptocurrency exchange rates. The LSTM reports the best results, outperforming GARCH and GJR-GARCH as well as HAR models throughout the data samples.

The consideration of extending the RECH models demonstrated by Nguyen et al. (2022) has multiple factors. According to the unpublished study of Lipton et al. (2015), current applications of recurrent neural networks refrain from using the SRN neuron and use especially the LSTM. They also note that advances in the application of RNNs are achieved by novel architectures. Additionally, the LSTM-GARCH has been considered in the unpublished work of Liu et al. (2023).

General alternatives to GARCH-type models in realised volatility forecasting are the heterogeneous autoregressive models of the realised volatility, HAR-RV models, proposed by Corsi (2009). A HAR-RV model is a simple AR-type model with the idea to use past daily, weekly, and monthly realized volatility as regressors. There are also efforts to connect HAR-RV models with neural networks. The unpublished study of Reisenhofer et al. (2022) proposes a HARNet model which parallels the autoregressive structure of the HAR-RV model: Each layer exploits data of different time horizons. They initialise the neural network with HAR-RV parameter estimates so that before optimisation both models produce identical forecasts. The HARNets outperform HAR-RV models in terms of all considered loss functions for realised variance forecasts of the S&P 500, FTSE and DJL index. The work of Christensen et al. (2021) concludes that the capabilities of machine learning methods to model non-linear relationships are a major reason for their success in realised volatility forecasting. They do not include GARCH-type models nor recurrent neural networks but find that non-linear feed-forward neural networks and non-linear tree-based

machine learning methods outperform linear HAR models. They use the same assets used in this thesis: The assets comprising the Dow Jones index. Those advances in realised volatility forecasting through complex non-linearity motivate the further exploration of new non-linear neuron architectures.

On the other hand, Sharma et al. (2015) find that the simple GARCH model provides better forecasts than the more sophisticated GARCH-type models EGARCH, GJR-GARCH, TGARCH, AVGARCH, APARCH, and NGARCH for the one-step-ahead daily conditional variance of stock indices. The author reports results robust to market conditions and choice of performance evaluation. For certain data, the study by Hansen and Lunde (2005) comes to a similar conclusion. They evaluate 330 ARCH-type models to investigate whether the GARCH(1,1) can at all be outperformed by any other ARCH-type model. They find that for the DM/$ exchange rate the GARCH(1,1) is not outperformed in realized variance forecasting. For the individual stock returns of IBM, however, models that account for leverage effects provide superior forecasts. This encourages further study in terms of the sample size of individual stocks, as the irrelevance of non-linear capabilities of GARCH models for one certain exchange rate variance and one certain stock return variance does not categorically rule out the importance of non-linear models for all financial asset variances.

The further parts of this thesis are structured as follows: The theoretical framework is presented, containing advances on the RECH model properties. A section containing the empirical methods and their results is presented. The findings are discussed and a conclusion contains a summary as well as proposals for further research.

3 Theoretical Framework

This section features the theoretical framework of RECH models and advances in the properties of RECH processes.

3.1 GARCH models

The generalized autoregressive conditional heteroscedasticity (GARCH) process is commonly used for modelling the volatility of financial time series data. Specifically, a GARCH(1, 1) process is defined as:

$$y_t = \sigma_t \epsilon_t, \quad \epsilon_t \overset{\text{i.i.d.}}{\sim} \mathrm{N}(0,1), \tag{1}$$

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

With $\alpha, \beta \geq 0$, and $\omega > 0$. The parameter ω serves as a constant, and α and β are respectively ARCH and GARCH parameters. It is worth noting that other distributions for ϵ_t are possible, but for this thesis, a normal distribution is assumed.

Furthermore, the GJR-GARCH serves as a benchmark model for the RECH model performance. The GJR-GARCH, proposed by Glosten et al. (1993), allows for asymmetric effects. A GJR-GARCH(1,1,1) is given by:

$$y_t = \sigma_t \epsilon_t \tag{3}$$

$$\sigma_t^2 = \omega + \alpha y_{t-1}^2 + \rho \mathbb{1}_{\{y_{t-1}<0\}} y_{t-1}^2 + \beta \sigma_{t-1}^2, \tag{4}$$

with $\rho > 0$ as an additional parameter which additionally increases the conditional variance whenever a negative return is observed. This accounts for the empirically observed leverage effects.

3.2 RECH models

Recently, a new class of models called recurrent conditional heteroscedasticity (RECH) models, has been introduced by Nguyen et al. (2022) to allow for non-linear volatility dynamics. The RECH model is based on the GARCH model but with a recurrent component ω_t replacing the constant ω. This recurrent component is obtained through a set of functions that resemble the neuron of a recurrent neural network (RNN). Hence, σ_t^2 is the sum of two processes, the recurrent component and the GARCH-type component.

The authors suggest a general formulation of a RECH model, allowing the choice

of any GARCH-type model or any RNN for the process of h_t:

$$y_t = \sigma_t \epsilon_t, \qquad \epsilon_t \overset{\text{i.i.d.}}{\sim} \mathrm{N}(0,1), \quad t = 1, 2, \ldots, T, \tag{5}$$

$$\sigma_t^2 = g(\omega_t) + f(\sigma_{t-1}^2, \ldots, \sigma_{t-p}^2, y_{t-1}, \ldots, y_{t-q}), \quad t = 2, \ldots, T, \sigma_1^2 = \sigma_0^2, \tag{6}$$

$$\omega_t = \gamma' h_t + \gamma_0, \quad t = 2, \ldots, T, \tag{7}$$

$$h_t = \mathrm{RNN}(x_t, h_{t-1}), \quad t = 2, \ldots, T, h_1 = 0, \tag{8}$$

with h_t the recurrent state, the output of a set of functions resembling the neuron of an RNN, subject to the input variables x_t and h_{t-1}. The RNNs are governed by linear combinations of the input variables transformed by non-linear activation functions. The recurrent component ω_t is a linear combination of L hidden states with γ_0 a scalar and $\gamma = (\gamma_1, \ldots, \gamma_L)$. A function $f(.)$ of σ and y builds a GARCH-type structure. They also allow for another gate, a non-negative function $g(.)$ to transform ω_t. Through the general formulation of ω_t and $f(.)$ this model allows for all GARCH-type models and different choices of neuron architectures in h_t.

A subset of the general RECH models is considered in this thesis:

$$y_t = \sigma_t \epsilon_t \qquad \text{with } \epsilon_t \overset{\text{i.i.d.}}{\sim} \mathrm{N}(0,1), \quad t = 1, 2, \ldots, T \tag{9}$$

$$\sigma_t^2 = \omega_t + \alpha y_{t-1}^2 + \beta \sigma_{t-1}^2, \quad t = 2, \ldots, T, \sigma_1^2 = \sigma_0^2 \tag{10}$$

$$\omega_t = \gamma_0 + \gamma_1 h_t, \quad t = 2, \ldots, T \tag{11}$$

$$h_t = \mathrm{RNN}(x_t, h_{t-1}), \quad t = 2, \ldots, T, h_1 = 0. \tag{12}$$

In this case, h_t is only considered as a scalar, reducing it from L dimensions to one dimension. The only GARCH-type component considered is the GARCH(1,1). The input is chosen as $x_t = (\mathrm{sgn}(y_{t-1}) y_{t-1}^2, \sigma_{t-1}^2)$, so that y_{t-1} and σ_{t-1} enter the non-linear functions both in the quadratic equation, without y_{t-1} losing its sign. Four different functional forms for h_t are selected: the SRN, MGU, and LSTM. The activation functions ϕ considered, are the bounded rectified linear unit (ReLU) and the logistic function as a "sigmoid":

$$\phi_{\mathrm{ReLU}}(x) = \min[\max(x, 0), M], \qquad \phi_{\mathrm{sigmoid}}(x) = \frac{1}{1 + e^{-x}} \tag{13}$$

Those functions contribute to the non-negativity of the recurrent component ω_t, as both are greater than or equal to zero. Another activation function discussed is the hyperbolic tangent function, $\tanh(x) = \frac{e^{2x}-1}{e^{2x}+1}$, which is widely used in machine learning applications of RNNs but due to its mapping onto $[-1, 1]$ is dismissed here.

3.2.1 SRN-GARCH

The SRN-GARCH combines the GARCH model with a simple recurrent neuron of a simple recurrent network (SRN). In this form, the SRN was initially proposed by Jordan (1986) for processing natural language. The combination of an SRN neuron and a GARCH(1,1) gives the following model:

$$\sigma_t^2 = \omega_t + \alpha y_{t-1}^2 + \beta \sigma_{t-1}^2, \tag{14}$$

$$\omega_t = \gamma_0 + \gamma_1 h_t, \tag{15}$$

$$h_t = \text{SRN}(x_t, h_{t-1}). \tag{16}$$

The SRN is indeed simple, as it only consists of the non-linear transformation of a linear combination of the input variables:

$$\text{SRN}(x_t, h_{t-1}) = \phi(\mathbf{v}x_t' + wh_{t-1} + b), \tag{17}$$

$$\text{SRN}(x_t, h_{t-1}) = \phi(v_1 \,\text{sgn}(y_{t-1}) y_{t-1}^2 + v_2 \sigma_{t-1}^2 + wh_{t-1} + b). \tag{18}$$

Nguyen et al. (2022) use the bounded ReLU as an activation function. The upper bound M is irrelevant for estimation purposes but Nguyen et al. (2022) show that it guarantees finite conditional variance of y_t, conditional on $\sigma_0^2 < \infty$. The parameters of the SRN-GARCH are interpretable to the extent that b serves as a threshold parameter and v_1 as a leverage effect parameter. A visual representation of the SRN-GARCH is found in Figure (1), where the symbol $\bigoplus$ indicates a linear combination.

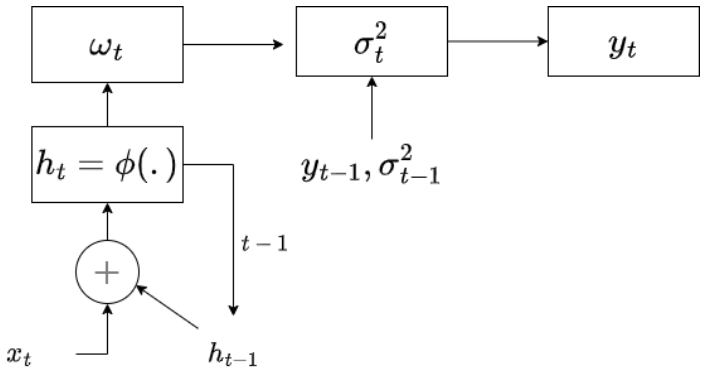


Figure 1: SRN-GARCH

3.2.2 MGU-GARCH

The minimal gated unit (MGU) was initially proposed by Zhou et al. (2016). An MGU-GARCH produces the following model:

$$\sigma_t^2 = \omega_t + \alpha y_{t-1}^2 + \beta \sigma_{t-1}^2, \tag{19}$$

$$\omega_t = \gamma_0 + \gamma_1 h_t, \tag{20}$$

$$h_t = \text{MGU}(x_t, h_{t-1}), \tag{21}$$

$$\text{MGU}(x_t, h_{t-1}) = f_t \hat{h}_t + (1 - f_t) h_{t-1}, \tag{22}$$

$$\hat{h}_t = \phi_{\hat{h}}(\mathbf{v}_1 x_t' + w_1 f_t h_{t-1} + b_{\hat{h}}), \tag{23}$$

$$f_t = \phi_f(\mathbf{v}_2 x_t' + w_2 h_{t-1} + b_f). \tag{24}$$

The vectors $\mathbf{v_i}$ contain the respective parameters as $\mathbf{v}_1 = (v_{11}, v_{12})$ and $\mathbf{v}_2 = (v_{21}, v_{22})$ While the activation function $\phi_{\hat{h}}$ can be chosen out of the pool of feasible activation functions, a meaningful activation function ϕ_f is the logistic function bound between 0 and 1, so that the MGU state h_t is a convex combination of the proposed MGU state $\hat{h}_t$ and the lagged MGU state, weighted by a scalar of "forgetting" f_t. A visual representation is given in Figure (2) with $\oplus$ indicating a linear combination, $\otimes$ a multiplication, and λ a convex combination. The parameter interpretation is not feasible, as the non-linearity of the MGU-GARCH is more complex compared to the SRN-GARCH.

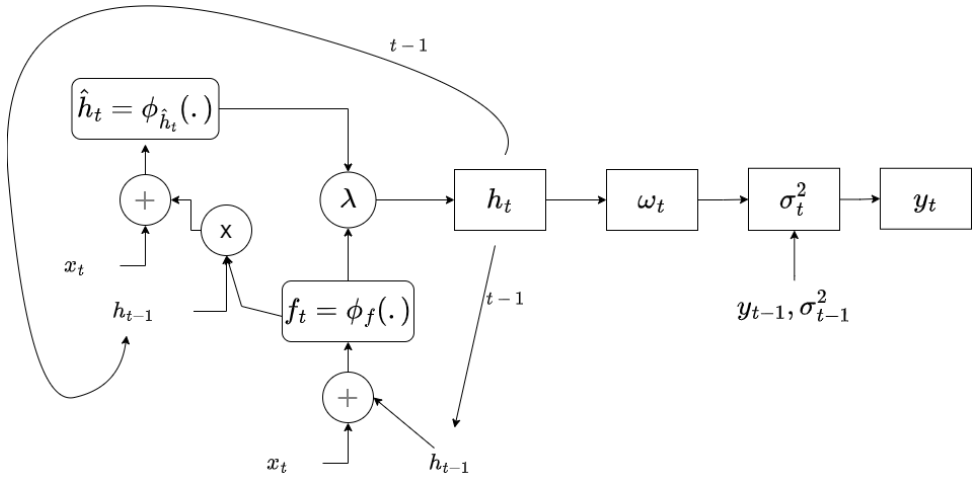


Figure 2: MGU-GARCH

3.2.3 LSTM-GARCH

The long short-term memory (LSTM) architecture was initially proposed by Hochreiter and Schmidhuber (1997). Combined with a GARCH model this yields the

following model:

$$\sigma_t^2 = \omega_t + \alpha y_{t-1}^2 + \beta \sigma_{t-1}^2, \tag{25}$$

$$\omega_t = \gamma_0 + \gamma_1 h_t, \tag{26}$$

$$h_t = \text{LSTM}(x_t, h_{t-1}), \tag{27}$$

$$\text{LSTM}(x_t, h_{t-1}) = o_t \phi_h(c_t), \tag{28}$$

$$c_t = f_t c_{t-1} + i_t \tilde{c}_t, \tag{29}$$

$$\tilde{c}_t = \phi_{\tilde{c}}(\mathbf{v}_1 x_t' + w_1 h_{t-1} + b_{\tilde{c}}), \tag{30}$$

$$o_t = \phi_o(\mathbf{v}_2 x_t' + w_2 h_{t-1} + b_o), \tag{31}$$

$$i_t = \phi_i(\mathbf{v}_3 x_t' + w_3 h_{t-1} + b_i), \tag{32}$$

$$f_t = \phi_f(\mathbf{v}_4 x_t' + w_4 h_{t-1} + b_f). \tag{33}$$

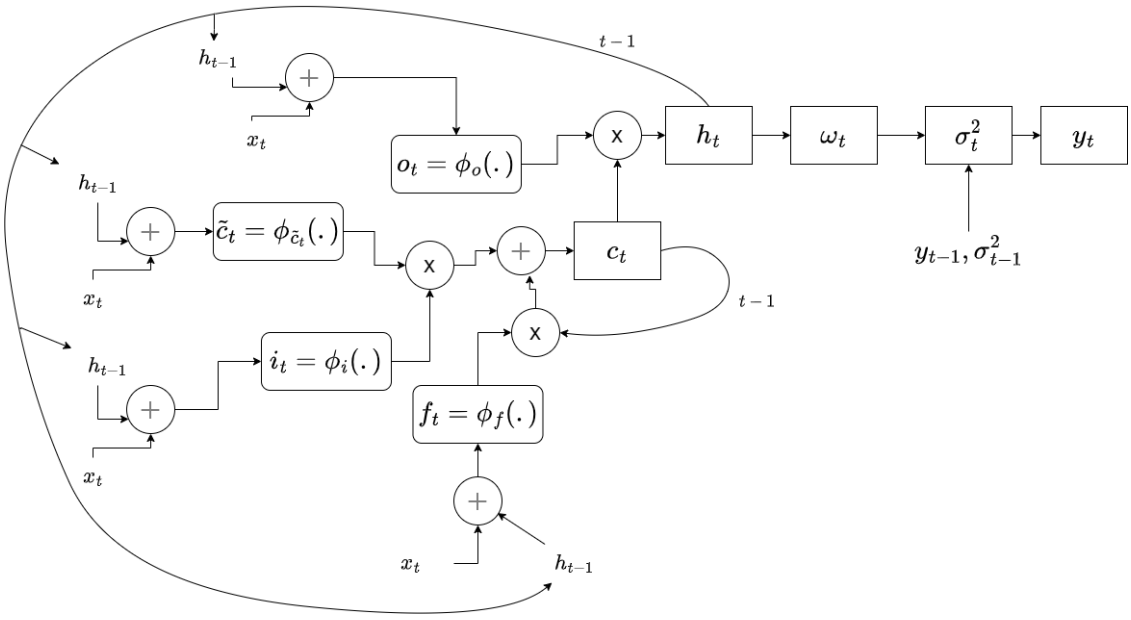


Figure 3: LSTM-GARCH

Within a neural network, c_t serves as a memory cell, that is updated with new information $\tilde{c}_t$. The input gate i_t determines how much new information is passed on and the forget gate f_t regulates how much information of the previous state c_{t-1} should be remembered. In the next step, an output gate o_t controls how much information of the memory cell is passed on to the state of the neuron h_t. Hochreiter and Schmidhuber (1997) use the logistic function for ϕ_o, ϕ_i, ϕ_f and the hyperbolic tangent for $\phi_{\tilde{c}}$ and ϕ_h. To have an identical output range for h_t throughout the different RECH models the ReLU activation function is chosen for $\phi_{\tilde{c}}$, making ϕ_h

obsolete. When choosing ReLU for ϕ_h the state of c_t would not change and with ϕ_h as the logistic function the output range of h_t would be pruned to $[0, 1]$. In Figure (3) a visual representation is given with $\bigoplus$ indicating a linear combination and $\bigotimes$ a multiplication.

3.3 Stationarity

Stationarity is a property of a stochastic process. If a process is stationary, its distribution does not change over time. Stationarity is defined as follows:

Definition 1 (Strict stationarity) *The process (X_t) is said to be strictly stationary if the vectors $(X'_1, \ldots, X'_k)'$ and $(X'_{1+h}, \ldots, X'_{k+h})'$ have the same joint distribution, for any $k \in \mathbb{N}$ and any $h \in \mathbb{Z}$.*

In the case where a process has finite second moments, $E(X_t^2) < \infty \forall t \in \mathbb{Z}$, one may consider the notion of second-order stationarity, which is defined as follows:

Definition 2 (Second-order stationarity) *The process (X_t) is said to be second-order stationary if*

1. $E(X_t^2) = \sigma^2 < \infty, \quad \forall t \in \mathbb{Z}$
2. $EX_t = m, \quad \forall t \in \mathbb{Z}$
3. $Cov(X_t, X_{t+h}) = \gamma_X(h), \quad \forall t \in \mathbb{Z}.$

The function $\gamma_X(.)$ is the autocovariance function.

Hence, (X_t) has a finite second-order moment, a constant mean and an autocovariance function independent from its (temporal) position within the process.

GARCH stationarity

Theorem 3.1 (Strict stationarity of the GARCH(1,1)) *The GARCH(1,1) is strictly stationary, if:*

$$-\infty \leq \gamma = E(\log[\alpha\epsilon_t^2 + \beta]) \tag{34}$$

If $\alpha + \beta < 1$, Condition (34) holds and the Jensen inequality can be applied:

$$E(\log[\alpha(\epsilon_t)]) \leq \log[E(\alpha(\epsilon_t)] = \log[\alpha + \beta] < 0. \tag{35}$$

If $\alpha + \beta < 1$, the process y_t is also second-order stationary.

SRN-GARCH stationarity Lee and Shin (2005) derive conditions for strict stationarity of a class of non-linear GARCH models:

$$y_t = \sigma_t \epsilon_t \qquad \text{with } \epsilon_t \overset{\text{i.i.d.}}{\sim} \mathrm{N}(0,1), \tag{36}$$

$$\sigma_t^2 = \sum_{i=1}^{p} c_i(\epsilon_{t-1}, \ldots, \epsilon_{t-r_1}, \sigma_{t-1}^2, \ldots, \sigma_{t-s_1}^2) f(\epsilon_{t-i}) \sigma_{t-i}^2, \tag{37}$$

$$+ \sum_{i=1}^{q} d_i(\epsilon_{t-1}, \ldots, \epsilon_{t-r_2}, \sigma_{t-1}^2, \ldots, \sigma_{t-s_2}^2) \sigma_{t-i}^2, \tag{38}$$

$$+ g(\epsilon_{t-1}, \ldots, \epsilon_{t-r_3}, \sigma_{t-1}^2, \ldots, \sigma_{t-s_3}^2), \tag{39}$$

with $p \geq 1, r, s \geq 0, r = max(r_1, r_2, r_3), s = max(s_1, s_2, s_3)$ positive integers and c_i, d_i, f and g are measurable functions defined on $R^{r_1+s_1}, R^{r_2+s_2}, R$ and $R^{r_3+s_3}$ respectively.

The RECH models fall into this class of non-linear GARCH models under the condition that past hidden states h_{t-1} do not enter the RNN equation systems. Rewriting the RECH models yields:

$$y_t = \sigma_t \epsilon_t \qquad \text{with } \epsilon_t \overset{\text{i.i.d.}}{\sim} \mathrm{N}(0,1), \tag{40}$$

$$\sigma_t^2 = g(\epsilon_{t-1}, \sigma_{t-1}) + \alpha \epsilon_{t-1}^2 \sigma_{t-1}^2 + \beta \sigma_{t-1}^2. \tag{41}$$

Here, we have $r_i = s_i = 1$ for $i = \{1, 2, 3\}$ and $p = q = 1$. $c_i(z) = \alpha z$, $d_i(z) = \beta z$ are just linear functions. For the SRN-GARCH this condition implies $w = 0$ in $h_t = \phi(v_1 \operatorname{sgn}(y_{t-1}) y_{t-1}^2 + v_2 \sigma_{t-1}^2 + w h_{t-1} + b)$. Otherwise, r_3 and s_3 in (39) would grow with time t.

The following contains the conditions derived by Lee and Shin (2005) and the implicated conditions for the SRN-GARCH:

1. $g(.)$ must be bounded continuous.

   This is fulfilled for all RECH models that use bounded continuous functions as activation functions.

2. $g(\epsilon_t, \sigma_t)$ must be a Feller chain.

   This condition is fulfilled, if $E[g(\epsilon_t, \sigma_t)|\epsilon_{t-1}, \sigma_{t-1}]$ is continuous.

For the SRN-GARCH we have $g(\epsilon_t, \sigma_t)$ as:

$$g(\epsilon_t, \sigma_t) = \gamma_0 + \gamma_1 \phi[v_1 \text{sgn}(y_t) y_t^2 + v_2 \sigma_t^2 + b] \tag{42}$$
$$= \gamma_0 + \gamma_1 \phi[\sigma_t^2 (v_1 \,\text{sgn}(\epsilon_t)\epsilon_t^2 + v2) + b], \text{ substituting } \sigma_t^2 : \tag{43}$$
$$= \gamma_0 + \gamma_1 \phi[(g(\epsilon_{t-1}, \sigma_{t-1}) + \alpha \epsilon_{t-1}^2 \sigma_{t-1}^2 + \beta \sigma_{t-1}^2)(v_1 \,\text{sgn}(\epsilon_t)\epsilon_t^2 + v2) + b] \tag{44}$$

$E[g(\epsilon_t, \sigma_t))|\epsilon_{t-1}, \sigma_{t-1}]$ is continuous with $g(\epsilon_{t-1}, \sigma_{t-1})$ continuous, $\epsilon_t \overset{\text{i.i.d.}}{\sim} \text{N}(0,1)$, and ϕ bounded continuous.

3. There are upper bounds to α and β: $\alpha \leq \tilde{\alpha}, \beta \leq \tilde{\beta}$ with $\tilde{\alpha} E[\epsilon_t^2] + \tilde{\beta} < 1 \Rightarrow \alpha + \beta < 1$, as $E[\epsilon_t^2] = 1$.

   This condition is fulfilled for $\alpha + \beta < 1$ in RECH models with an underlying normal distribution.

Under conditions (1) to (3) the RECH model is strictly stationary. Those conditions are only feasible for the SRN-GARCH but not for the MGU- and LSTM-GARCH, as their structure would be too heavily altered after implementing the conditions.

General RECH model stationarity The stationarity result of Lee and Shin (2005) for the class of non-linear GARCH models (36) - (39) does not apply to the general class of RECH models. General RECH models are not included because the time horizon of $g(.)$ in (39) is fixed by constants r_3 and s_3. With $h_{t-1}(\epsilon_{t-2}, \sigma_{t-2}, h_{t-2})$ being included in h_t, r_3 and $s_3 \to \infty$ for $t \to \infty$.

Straumann (2005) offers explicit conditions for establishing strict stationarity of stochastic recurrence equations (SRE). Those conditions were initially derived by Bougerol (1993). Model (9) - (12) can be expressed as SRE:

$$X_t = \phi_t(X_{t-1}), \tag{45}$$
$$X_t = \phi(\epsilon_{t-1}, X_{t-1}) \quad \epsilon_t \overset{\text{i.i.d.}}{\sim} \text{N}(0,1), \tag{46}$$

with $X_t = (\sigma_t^2, h_t)'$ and $\phi = (\phi_1, \phi_2)'$, where

$$\phi^{(1)}(\epsilon_{t-1}, X_{t-1}) = \gamma_0 + \gamma_1 g(\epsilon_{t-1}, \sigma_{t-1}^2, h_{t-1}) + \alpha \epsilon_{t-1}^2 \sigma_{t-1}^2 + \beta \sigma_{t-1}^2, \tag{47}$$
$$\phi^{(2)}(\epsilon_{t-1}, X_{t-1}) = g(\epsilon_{t-1}, \sigma_{t-1}^2, h_{t-1}). \tag{48}$$

Straumann (2005) derives sufficient conditions for strict stationarity. Because those are hard to verify they lie beyond the scope of this thesis.

3.4 Estimation

In contrast to the Bayesian estimation approach employed by Nguyen et al. (2022), a frequentist method with quasi-maximum likelihood estimation (QMLE) is adopted. According to Francq and Zakoian (2019), strict stationarity is required for the QMLE to be consistent and asymptotically normal. The estimator is derived as if the innovations ϵ_t were standard normal distributed. Violating this is not problematic, even if the innovations had a different distribution, the estimator would remain consistent.

The parameter set of true, unknown parameters is denoted by $\theta = (\omega, \alpha, \beta)'$ for a GARCH(1,1), $\theta = (\omega, \alpha, \beta, \gamma_0, \gamma_1, v_{11}, v_{12}, w, b)'$ for an SRN-GARCH and so on. Conditioning on initial values ϵ_0, σ_0^2, h_0, the conditional Gaussian quasi-likelihood is expressed as:

$$L_n(\theta) = L_n(\theta; y_1, \ldots, y_n) = \prod_{t=1}^{n} \frac{1}{\sqrt{2\pi\sigma_t^2(\theta)}} \exp\big(-\frac{y_t^2}{2\sigma_t^2(\theta)}\big), \tag{49}$$

with

$$\sigma_t^2(\theta) = \omega + \alpha y_{t-1}^2 + \beta\sigma_{t-1}(\theta)^2 \tag{50}$$

for the GARCH(1,1). For the RECH models, it is accordingly:

$$\sigma_t^2(\theta) = \gamma_0 + \gamma_1 h_t^{(i)}(\theta) + \alpha y_{t-1}^2 + \beta\sigma_{t-1}^2(\theta), \tag{51}$$

$$\tag{52}$$

with $i \in \{\text{SRN, MGU, LSTM}\}$. The QMLE is given by:

$$\hat{\theta}_n = \arg\max L_n(\theta), \tag{53}$$

which is equivalent to

$$\hat{\theta}_n = \arg\min \mathbb{I}_n(\theta), \tag{54}$$

with

$$\mathbb{I}_n(\theta) = n^{-1} \sum_{t=1}^{n} \ell_t(\theta), \quad \text{and} \quad \ell_t(\theta) = \frac{y_t^2}{\sigma_t^2(\theta)} + \log\sigma_t^2(\theta). \tag{55}$$

As Francq and Zakoian (2019) note, the initial values ϵ_0, σ_0^2 are unimportant asymptotically they can be chosen freely. The starting values of the optimisation algorithm

(denoted with $\sim$) are used to set $\epsilon_0 = \sigma_0^2 = \frac{\tilde{\omega}}{1-\tilde{\alpha}-\tilde{\beta}}$. For the GJR-GARCH $\frac{\tilde{\omega}}{1-\tilde{\alpha}-0.5\tilde{\rho}-\tilde{\beta}}$ is used. For the RECH models, which do not have the parameter ω, the arbitrary value $\frac{0.1}{1-\tilde{\alpha}-\tilde{\beta}}$ is used. As in Nguyen et al. (2022), we set $h_1 = 0$, initialising the recurrent component memoryless.

3.4.1 Optimisation

The application of the maximum likelihood estimation requires the optimisation of a target function. The implementation takes place using the scientific programming package `SciPy` [1] in Python. The choice of algorithm has a strong impact on the results. `SciPy` offers many different algorithms, with different qualities. The sequential least squares quadratic programming algorithm (SLSQP) is selected because it allows for constraints which are needed for estimating $\omega, \alpha, \beta, \gamma_0$ and γ_1. All those parameters must be greater than zero and the condition $\alpha + \beta < 1$ must hold. The implementation of the SLSQP can be found in Kraft (1988). Apart from the SLSQP, the Nelder-Mead algorithm was considered but can lead to estimates with $\hat{\alpha} + \hat{\beta} > 1$ which then lead to exploding forecasts.

3.5 Forecasting

This section presents the forecasting procedure and the forecast evaluation.

3.5.1 Variance Forecasting

The objective of variance forecasting is to create τ-day ahead forecasts of the variance of the log returns, given all observations up to a certain day t. The expected variance at time $t + \tau$ is given by:

$$E(y_{t+\tau}^2|\mathcal{I}_t) = E(\sigma_{t+\tau}^2|\mathcal{I}_t). \tag{56}$$

The expected value $E(\sigma_{t+\tau}^2|\mathcal{I}_t)$ is obtained through a Monte Carlo simulation. For this, H paths of y and σ^2 are simulated from $t + 1$ to $t + \tau$. The first value of σ_t^2, based on which to start the forecasting simulation, is obtained from the data up to period t. As starting values y_0, σ_0^2, and h_0, the same starting value as in the estimation are used, but the optimisation starting values are exchanged for the . For $\tau \geq 2$ a number of simulations H is chosen and starting at $i = 1$ the forecasting is conducted as follows:

[1] `https://docs.scipy.org/doc/scipy/index.html`

1. Draw a random variable $\epsilon_{t+1}^{(i)}$ from a standard normal distribution to compute

$$\hat{y}_{t+1}^{(i)} = \hat{\sigma}_{t+1}^{(i)} \epsilon_{t+1}^{(i)}. \tag{57}$$

2. Use $\hat{y}_{t+1}^{(i)}$ and $\hat{\sigma}_{t+1}^{(i)}$ to compute $h_{t+1}^{(i)}$.

3. Use $\hat{y}_{t+1}^{(i)}$, $\hat{\sigma}_{t+1}^{(i)}$, and $h_{t+1}^{(i)}$ to compute $\hat{\sigma}_{t+2}^{(i)}$.
   Use $\hat{\sigma}_{t+2}^{(i)}$ and resume from Step (1) for y_{t+2} until $t + \tau$ is reached.

4. When $t + \tau$ is reached, safe $\hat{\sigma}_{t+\tau}^{(i)}$ and continue with a new simulation run from Step (1) until $i = H$ is reached.

5. When $i = H$ is reached, hence when H paths have been simulated, the variance forecast of horizon τ is given by

$$\hat{\sigma}_{t+\tau} = \frac{1}{H} \sum_{i=1}^{H} \hat{\sigma}_{t+\tau}^{(i)}. \tag{58}$$

For $\tau = 1$ the forecast can be obtained analytically, both for the RECH models and for the GARCH models.

3.5.2 Realized Variance

Realized variance (RV) is a measure of risk or uncertainty of the returns of an asset. Realized variance is calculated by measuring the actual price changes of an asset over a particular time period, such as a day, week, month, or year. The daily RV is estimated using the RV estimator proposed by Andersen et al. (2003) in the notation of Brownlees et al. (2011):

$$\hat{\sigma}_{\text{RV } t}^2 = \sum_{j=2}^{I_t} (p_{t,j} - p_{t,j-1})^2 \tag{59}$$

with $p_{t,j} (j = 1, \ldots, J_t)$ the series of log mid-quote prices on day t, referred to as intra-day returns. An alternative to using RV is using squared daily returns as a volatility measure, but as Hansen and Lunde (2006) showed, this can lead to a biased ranking of volatility forecasting models. Hence, RV is used.

3.5.3 Forecasting Loss

The forecast loss of model i is captured by $L_{i,t}$ with $L_{i,t} = L(\hat{\sigma}^2_{i,t}, \hat{\sigma}^2_{\mathrm{RV}i,t})$. The loss functions used for $L_{i,t}$ are the absolute error, given by:

$$L^{\mathrm{AE}}_{i,t} = |\hat{\sigma}^2_{i,t} - \hat{\sigma}^2_{\mathrm{RV}i,t}|, \quad \mathrm{MAE}_i = \frac{1}{T}\sum_{t=1}^{T} L^{\mathrm{AE}}_{i,t}, \tag{60}$$

and the squared error, given by:

$$L^{\mathrm{SE}}_{i,t} = (\hat{\sigma}^2_{i,t} - \hat{\sigma}^2_{\mathrm{RV}i,t})^2, \quad \mathrm{MSE}_i = \frac{1}{T}\sum_{t=1}^{T} L^{\mathrm{SE}}_{i,t}. \tag{61}$$

The losses $L_{i,t}$ are used to compute MCSs for the realised variance forecasting. The mean losses are used for the evaluation of the in-sample fits and also for the realised variance forecasting.

3.5.4 Model Confidence Set

The model confidence set (MCS) is a model selection tool proposed by Hansen et al. (2011). The objective is to compute a subset of models that have a lower expected forecast loss than the set of total considered models. The MCS includes the best models with a certain level of confidence, corresponding to the idea of confidence intervals of parameter estimations.

The starting point is a set that contains all considered models, $\mathcal{M}^0$. The objective is to find the set $\mathcal{M}^*$ that only contains superior models. The superiority is determined by an iteration of significance tests which test for differences in forecasting loss. The set of superior models is defined as

$$\mathcal{M}^* = \{i \in \mathcal{M}^0 : \mu_{ij} = 0 \quad \text{for all models } i,\, j \in \mathcal{M}^0\}.$$

Hence, the set of superior models includes all models whose expected forecast loss relative to the other models is zero. To compare model performances, the relative performances are defined as $d_{ij,t} = L_{i,t} - L_{j,t}$ for all $i,\, j \in \mathcal{M}^0$. The expected value $\mu_{ij} = \mathrm{E}(d_{ij,t})$ is used for the hypothesis tests with the idea that when $\mu_{ij} < 0$ model i is superior to model j.

In an algorithmic computation, a set $\hat{\mathcal{M}}^{*1-\alpha}$ is obtained which includes $\mathcal{M}^*$ with probability greater than or equal to $1 - \alpha$. This is conducted as follows:

1. Set $\mathcal{M} = \mathcal{M}^0$,

2. Test $\mathcal{H}_{0,\mathcal{M}} : \mu_{i,j} = 0$ using significance level α,

3. if $\mathcal{H}_{0,\mathcal{M}}$ was not rejected, define $\hat{\mathcal{M}}^{*1-\alpha} = \mathcal{M}$; if it is rejected, the model with the highest t-statistic is removed from $\mathcal{M}$ and the algorithm continues from Step (2).

(2) is tested as a set of t-tests, which take the form

$$t_{ij} = \frac{\bar{d_{ij}}}{\sqrt{\widehat{\text{var}}(\bar{d_{ij}})}}, \tag{62}$$

where $\bar{d_{ij}}$ denotes the arithmetic mean of d_{ij}. The variances are obtained from bootstrap resampling. In Step (3) the model with the highest t-statistic is removed.

The models remaining in the set have a significantly lower forecasting error than the models in the original set at confidence level α. Models are only removed when they are found to perform significantly poorer. The crucial advantage of the MCSs is, that the degree of information in the data is accounted for. This means, that when there is not one clear best-performing model, multiple models are part of the MCS, while metrics like the MAE, MSE or other loss functions produce a model ranking where the differences in average losses are hard to interpret.

4 Methods

The following section contains the in-sample simulation studies and the realised variance forecasting application with multiple evaluation methods. The Python code is published on `https://github.com/mbilsen/RECH_models`.

4.1 Simulation Studies

The simulation studies are aimed to investigate two features: The parameter identification of the SRN-GARCH and the in-sample fit performance of the RECH models. The first simulation examines whether the simplest of the RECH models, the SRN-GARCH, is identifiable so that the model parameters are interpretable. For this, stationary SRN-GARCH processes are generated and based on this data, SRN-GARCH models are estimated.

To investigate the in-sample performances three more simulations are conducted. Those explore the ability of RECH models. Three questions are examined:

1. When the true DGP is a GARCH(1,1), can the RECH models fit the data just as well as a GARCH(1,1)?

2. When the true DGP exhibits leverage effects, can the RECH models fit them as well as a GARCH model with leverage effects, namely the GJR-GARCH?

3. When the true DGP is non-linear, can the RECH models fit the data better than linear GARCH models?

4.1.1 Simulation I: SRN-GARCH model identification

Identifiability is a fundamental criterion for conducting inference in an econometric model. An identifiable model is one for which it is theoretically possible to estimate the true values of its underlying parameters, given an infinite amount of data. It means that the true parameter values can be uniquely determined based on the data. To investigate the identifiability of the RECH model, a simulation study is conducted for the simplest model, the SRN-GARCH. The SRN-GARCH is chosen, because we have stationarity conditions to generate stationary data. A total of 1000 time series consisting of 1000 observations were simulated governed by an SRN-GARCH model, and based on each resulting series an SRN-GARCH model was estimated. If, on average, the parameter estimates are close to the parameters based on which the data was generated we have evidence pointing towards identifiability. If the opposite is found, and throughout the simulation study the estimates are far from the true values, we have evidence pointing towards non-identifiability.

	α	β	γ_0	γ_1	v_1	v_2	w	b
DGP:	0.1	0.8	0.07	0.2	-0.3	0.5	0	-0.5
St. Values:	0.1	0.1	0.1	0.1	0.1	0.1	0	0.1
Bounds:	(0.0001, 1)	(0.0001, 1)	(0.0001, 10)	(0.0001, 3)	(-50, 50)	(-50, 50)	(0, 0)	(-100, 100)

Table 1: True parameters (DGP), starting values, and the bounds of the optimization algorithm

The data-generating process (DGP) is an SRN-GARCH of the following form:

$$y_t = \sigma_t \epsilon_t, \quad \epsilon_t \overset{\text{i.i.d.}}{\sim} \mathrm{N}(0,1), \tag{63}$$

$$\sigma_t = \omega_t + 0.1 y_{t-1}^2 + 0.8 \sigma_{t-1}^2, \tag{64}$$

$$\omega_t = 0.07 + 0.2 h_t, \tag{65}$$

$$h_t = \phi(-0.3 \operatorname{sgn}(y_{t-1}) y_{t-1}^2 + 0.5 \sigma_{t-1}^2 - 0.5), \tag{66}$$

for $t = 0, \ldots, 1000$ with $y_0, \sigma_0^2 = 1$.

Table (1) contains information on the parameterization of the optimization algorithm.

Results: The histograms of the model fits are depicted in figures (4) and (5), reporting an MAE of 3.7 and an MSE of 66.6, averaged over all 1000 simulations. In ten models the MSE is above 500, but those outliers do not skew the average MSE, as it is about 60.1 without including the outliers.

To approximate the estimator distribution, the histograms of the parameter estimates are shown in figures (6) - (12). For the estimate of α, we see bunching around the true parameter of 0.1 and an average parameter estimate of 0.127 with a starting value of 0.1. The parameter β bunches at 0.0 and around 0.87 with a left skew. There is no bunching around the true value of 0.8, but the average estimate is close with 0.753. This is also the case for the estimations of γ_0, as the average estimate is 0.077 for a true value of 0.07. There is bunching at 0.0 with a right skew because 0.0 is the lower boundary. Although starting value of 0.1 is close to the true value, there is no bunching at 0.1. The parameter γ_0 exhibits bunching at 0.0, an average estimate of 0.391 with a true value of 0.2. For v_1 the estimates bunch at the lower boundary of -50 and pronounced around 0.0, reporting an average of -3.383 for the true value of -0.3. The parameter v_2 shows estimates that are very widely spread around the average of -13.121 with bunching at 0.0 and the lower boundary of -50. The average, however, is very far from the true value of 0.5. The last parameter, b, reports bunching at around 0.0 and around 10 and an average value of 6.012 for a true value of -0.5.

Overall, they reveal that the SRN-GARCH is at most partially identifiable with α being the only parameter with estimates distributed around the true parameter value. But this is likely found because the starting value of the optimization algorithm is identical to the true parameter value of α.

The results will likely be better when choosing the true parameters as starting values. This is not investigated further, as knowing the true parameters ex-ante is unrealistic in practice. The bounds found in Table (1) also explain the bunching of estimates at the edges of the ranges.

It is important to note that the lack of identification and the resulting misestimation do not lead to worse model fits. The average MAE of the 137 simulations with estimates of $\alpha \in (0.05, 0.15)$ and $\beta \in (0.75, 0.85)$ is about 3.7 too. This also holds up when considering other parameters.

4.1.2 Simulation II: Different Levels of Non-linearity

In the following three subsections different levels of non-linearity are investigated. A GARCH(1,1) model without non-linearity, a GARCH model with leverage effects and a GARCH model with a high degree of non-linearity.

In these simulations, the starting values might play a role too. For every parameter, the value 0.1 was chosen as a starting value, except for the β parameter which is 0.8 for all models.

GARCH DGP The GARCH(1,1) data is generated with the following model:

$$y_t = \sigma_t \epsilon_t, \quad \epsilon_t \overset{\text{i.i.d.}}{\sim} \mathrm{N}(0,1) \tag{67}$$

$$\sigma_t = 0.05 + 0.18 y_{t-1}^2 + 0.8 \sigma_{t-1}^2, \tag{68}$$

for $t = 0, \ldots, 1000$ with $y_0, \sigma_0^2 = \omega/(1 - \alpha - \beta) = 2.5$. The data is generated and estimated 100 times.

Results: The losses of the model fits are reported in Table (2). For each model fit the MAE and the MSE are reported. Because the MSE of the GARCH data is very affected by a small number of simulations, columns are added that contain the losses of a subset of simulations where the five worst performances were eliminated. The first column of Table (2) shows that the MAEs of the SRN- and MGU-GARCH are even slightly lower than the MAE of the GARCH(1,1) with values of 5.52 and 5.51 respectively compared to 5.53. GJR-GARCH and LSTM-GARCH perform worse with MAEs of 7.64 and 13.75. In terms of MSE, which is found in the second column,

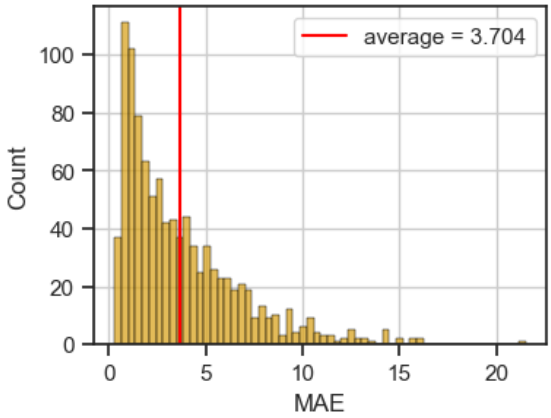


Figure 4: SRN absolute error histogram

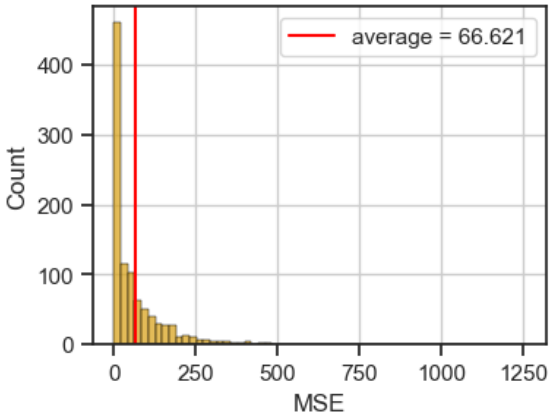


Figure 5: SRN squared error histogram

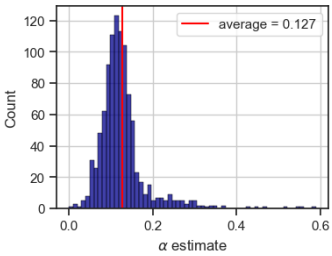


Figure 6: α estimates

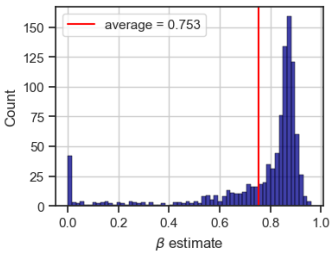


Figure 7: β estimates

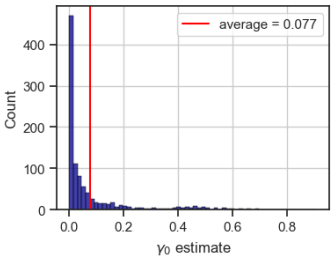


Figure 8: γ_0 estimates

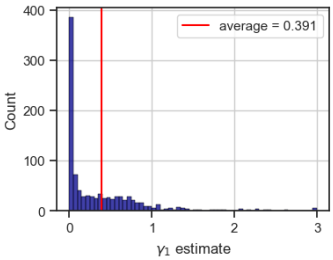


Figure 9: γ_1 estimates

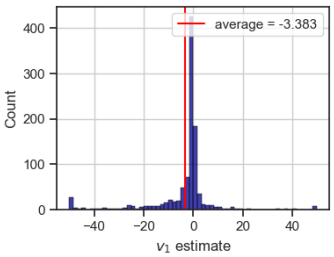


Figure 10: v_1 estimates

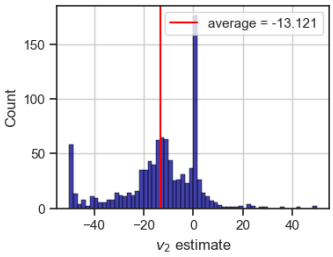


Figure 11: v_2 estimates

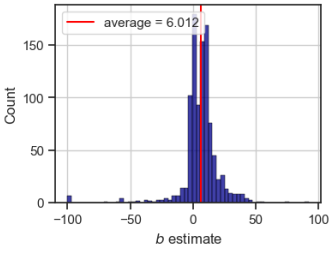


Figure 12: b estimates

the MGU-GARCH performs best with a value of 85.67 followed by the SRN-GARCH and GARCH(1,1) with values of 99.08 and 102.12 respectively. The GJR-GARCH

	full sample		95 best	
	MAE	MSE	MAE	MSE
GARCH	5.53	102.12	4.21	26.00
GJR	7.64	302.33	5.44	43.42
SRN	5.52	99.08	4.22	26.01
MGU	5.51	85.67	4.29	26.89
LSTM	13.75	6919.04	4.39	29.19

Table 2: GARCH simulation fits

reports an MSE of 302.33 and the LSTM-GARCH has a value of 6919.04 which is explained by extreme outliers. The third and fourth columns report the respective losses for each model subsampling the 95 best-fitted simulation runs. Within those, the MAEs and MSEs of the RECH models are very close to the GARCH(1,1) which reports the lowest losses and the GJR-GARCH performs worst.

GARCH with Leverage Effect The GARCH with leverage effect is generated as follows:

$$y_t = \sigma_t \epsilon_t, \quad \epsilon_t \overset{\text{i.i.d.}}{\sim} \mathrm{N}(0,1) \tag{69}$$

$$\sigma_t = 0.1 + 0.075 y_{t-1}^2 + 0.2 \mathbb{1}_{\{y_{t-1}<0\}} y_{t-1}^2 + 0.71 \sigma_{t-1}^2, \tag{70}$$

for $t = 0, \ldots, 1000$ with $y_0, \sigma_0^2 = \omega/(1-\alpha-0.5\rho-\beta) = 0.87$. The data is generated and estimated 100 times.

Results: The model fit losses for the data that has been generated by a GARCH with leverage effects can be found in Table (3). Both in MAE and MSE the GARCH(1,1) performs best, closely followed by the RECH models which perform very similarly. Although the data is generated by a GJR-GARCH, the GJR-GARCH provides the worst fit with an MAE that is 32% higher than the GARCH(1,1) and 69% higher in terms of MSE. In this simulation, no huge outliers were found in any of the model fits, so the results remain close to each other in both subsamples.

	full sample		95 best	
	MAE	MSE	MAE	MSE
GARCH	0.98	1.21	0.90	0.88
GJR	1.29	2.04	1.19	1.55
SRN	1.04	1.37	0.95	0.97
MGU	1.02	1.40	0.91	0.94
LSTM	1.03	1.40	0.92	0.96

Table 3: Leverage GARCH simulation fits

Non-linear GARCH To address the third characteristic simulation II in Nguyen et al. (2022) is repeated. The GARCH model with non-linear effects takes the form:

$$y_t = \sigma_t \epsilon_t, \quad \epsilon_t \overset{\text{i.i.d.}}{\sim} \mathrm{N}(0,1), \tag{71}$$

$$\sigma_t^2 = 0.05 + 0.10 y_{t-1}^2 + 0.21 \frac{y_{t-1}^2}{1+y_{t-1}^2} + 0.6\sigma_{t-1}^2 + 0.11 \frac{\sigma_{t-1}^2}{1+\sigma_{t-1}^2} + 0.2 \mathbb{1}_{\{y_{t-1}<0\}} y_{t-1}^2 + 0.1 \frac{\mathbb{1}_{\{y_{t-1}<0\}}}{1+e^{-y_{t-1}^2}}, \tag{72}$$

for $t = 0, \ldots, 1000$ with starting values $y_0, \sigma_0^2 = \omega/(1-\alpha_1-\beta_1) = 0.17$. The data is generated and estimated 100 times.

Results: In Table (4) the model losses for the non-linear GARCH process can be found. For this simulation, the losses of the GARCH(1,1) and the RECH models are extremely close to each other. The full-sample average MAE of the GARCH(1,1) is 1.32 and equal to the MAE of the SRN-GARCH. The LSTM-GARCH and MGU-GARCH report MAEs of 1.34 and 1.36 respectively. The GJR-GARCH demonstrates an MAE of 1.64. In terms of MSE, the SRN-GARCH reports the lowest value of 2.00 which is slightly less than the 2.03 of the GARCH(1,1) and the 2.04 of the LSTM-GARCH. MGU-GARCH and GJR-GARCH have values of 2.23 and 3.11 respectively. Within the sub-sample of the 95% best fits the MGU-GARCH performs as well as the other RECH models, which all report average MAEs and MSEs close to those of the GARCH(1,1).

4.2 Realised Volatility Forecasting

For each model and asset, the realised volatility forecasting is conducted as follows:

1. The model is estimated using the in-sample data.

	full sample		95 best	
	MAE	MSE	MAE	MSE
GARCH	1.32	2.03	1.23	1.59
GJR	1.64	3.11	1.53	2.47
SRN	1.32	2.00	1.24	1.59
MGU	1.36	2.23	1.25	1.65
LSTM	1.34	2.04	1.27	1.71

Table 4: Non-linear GARCH simulation fits

2. The forecasts ($\tau = 1, \tau = 5, \tau = 20$) for the first 20 days out-of-sample are computed based on 5000 simulations.

3. The first 20 observations of the in-sample data are dropped and the first 20 observations of the out-of-sample data are attached to the in-sample data and dropped from the out-of-sample data.

Hence, we have a monthly rolling re-estimation in accordance with the results of Brownlees et al. (2011). They heavily argue in favour of a rolling re-estimation. However, they recommend re-estimating once a week. Once a week is computationally not feasible due to the complexity of the optimisation problem of the RECH models. The first starting values for estimation of the GARCH-component parameters are chosen heuristically as 0.1 and 0.8. Other first starting values are arbitrarily set to 0.1. For every re-estimation the previous estimates are used as starting values.

4.2.1 Data description

To assess the model performances empirically, the return time series of the companies listed in the Dow Jones Industrial Average and the returns of the S&P 500 index are considered. The Dow Jones contains 30 US companies from a wide range of industries with a long operating history. Those companies reflect a great part of the US economy with a total market capitalization of \$34.51B[2] on September 29th 2020.

The total observation period starts on 03.01.2005 and ends on 30.09.2020 with 3964 trading days. For Visa Inc., the observation period starts with the stock market launch on 19.03.2008. The company Dow Inc. is dropped from the analysis because it is established its current form in 2019 which falls into the out-of-sample period. The return time series are demeaned before the estimation.

[2] `https://ycharts.com/`

The return time series are downloaded from yahoo finance[3] and split into an in-sample period from 2005-2017 and an out-of-sample period from 2017-2020. The model out-of-sample performance is evaluated using realised volatility. This is computed from quote data obtained from the LOBSTER database[4]. As this database does not provide index data, the realised volatility data of the S&P 500 is obtained from pitrading[5].

The sample data contains calm periods of low volatility as well as high volatility periods. The average return standard deviation of the observed assets in the year 2008 is 3.18% which is more than double the volatility of the period 2005-2007 (1.35%) and the period 2009-2017 (1.54%). Besides the financial crisis of 2007-2008, the sample contains more market-wide volatility surges, e.g. the European sovereign debt crisis in 2011 or the weeks after the 2010 flash crash.

Within the out-of-sample period, the average volatility of the considered assets rises from 1.29% in 2017-2019 to 3.10% in the observation period of 2020. This sharp rise in volatility is explained by the market reactions to the Covid-19 outbreak in March and April 2020.

4.2.2 Results

The variance forecasts are evaluated in several ways: A set of figures depicting the forecast series is created for a visual impression of the variance forecasts; using the MAE and the MSE for the whole out-of-sample loss as well as subsamples of low (Jan. 2020) and high (Mar. 2020) volatility the MCSs are computed, and the relative performances are illustrated in boxplots.

One-day-ahead forecasts of the S&P 500 are illustrated in Figures (13, 14, and 15), spotlighting different levels of market volatility and respective model forecasts. The five months time windows are chosen because, in each, the realized variance has a certain structure: In the subsample in 2017 (13) there is no high volatility cluster and the realized variance never exceeds a value of 1. The subsample in 2018 exhibits overall low volatility but has two clusters of higher volatility in February and late March to May. The five months sample of 2020 is interesting because it contains the period of the global Covid-19 outbreak followed by highest degrees of economic uncertainty. This is well observable in the realized variance of March 2020 on market level in the S&P 500 as well as all assets of the Dow Jones. The five months window contains the shock as well as its subsidence.

[3] `https://finance.yahoo.com/`
[4] `https://lobsterdata.com/index.php`
[5] `https://pitrading.com/`

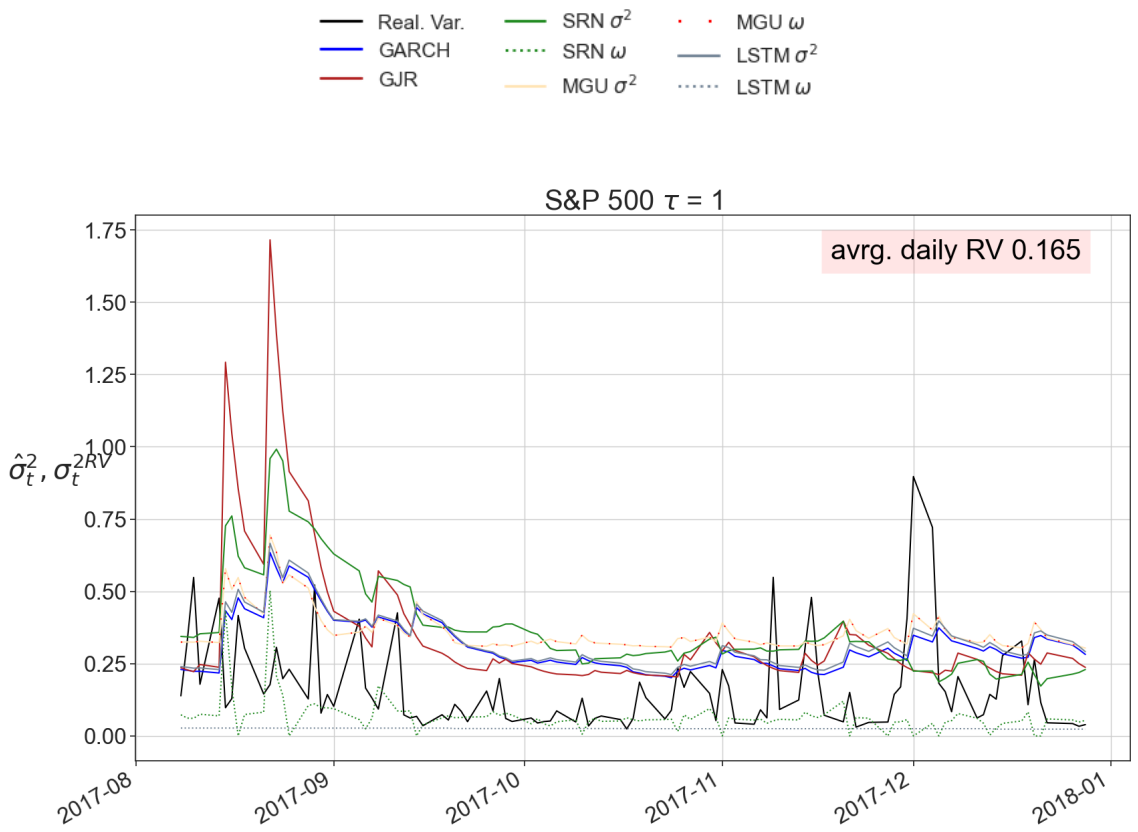


Figure 13: S&P 500 Variance forecasts 2017, one-day-ahead

The variance forecasts for the S&P 500 show how each RECH model can be very different in their composition. The recurrent component of the LSTM-GARCH is almost constant and very low, hence the forecasts of LSTM-GARCH and GARCH are almost identical. The recurrent component of the MGU-GARCH is very high and "dominates" the GARCH component in a way that the variance forecast consists almost entirely of the recurrent component. This is also reflected in the parameter estimates of the GARCH component in Table (12). Most of the SRN-GARCH forecasts are explained by the GARCH component and the recurrent component has mostly low values, but in high volatility periods can take on higher values as well.

Figure (16) highlights two aspects of the models. Firstly, they illustrate the relationship between the model input, the log returns, and model outputs as variance forecasts. Secondly, they show the model behaviours for forecast horizons of 5 and 20 trading days. The leverage effects of the GJR-GARCH are visible for a horizon of 1 in the low volatility period. When negative returns exceed -2 the conditional variance spikes the next day. The leverage effects are hardly visible for the RECH models. This is likely due to their more complex non-linear structure. The depictions

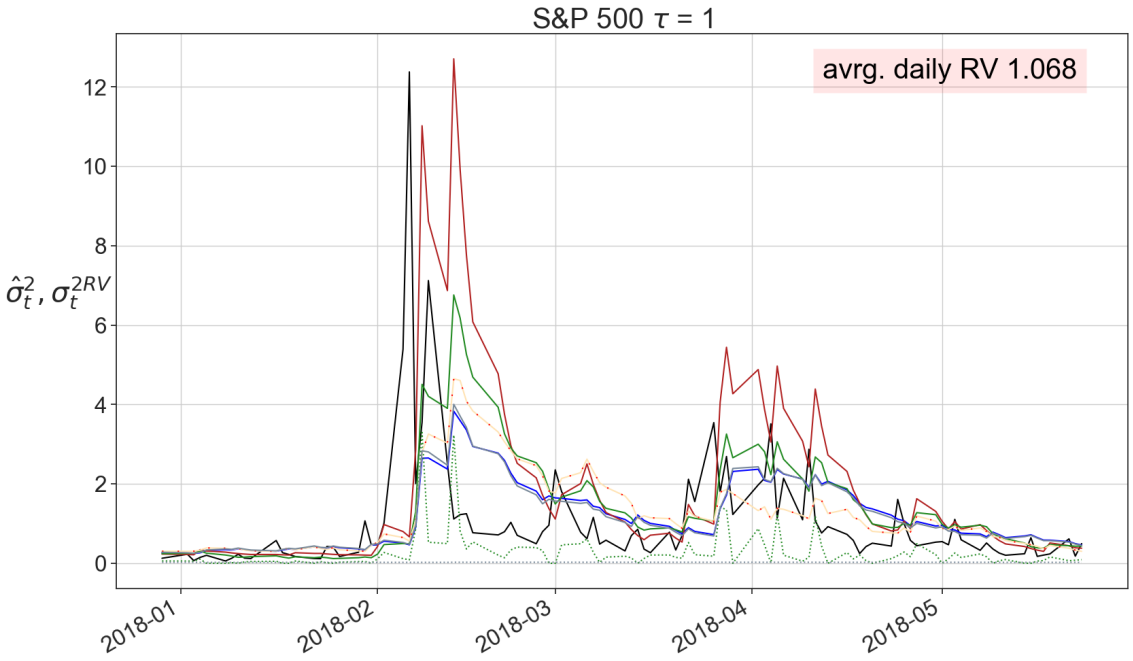


Figure 14: S&P 500 Variance forecasts 2018, one-day-ahead

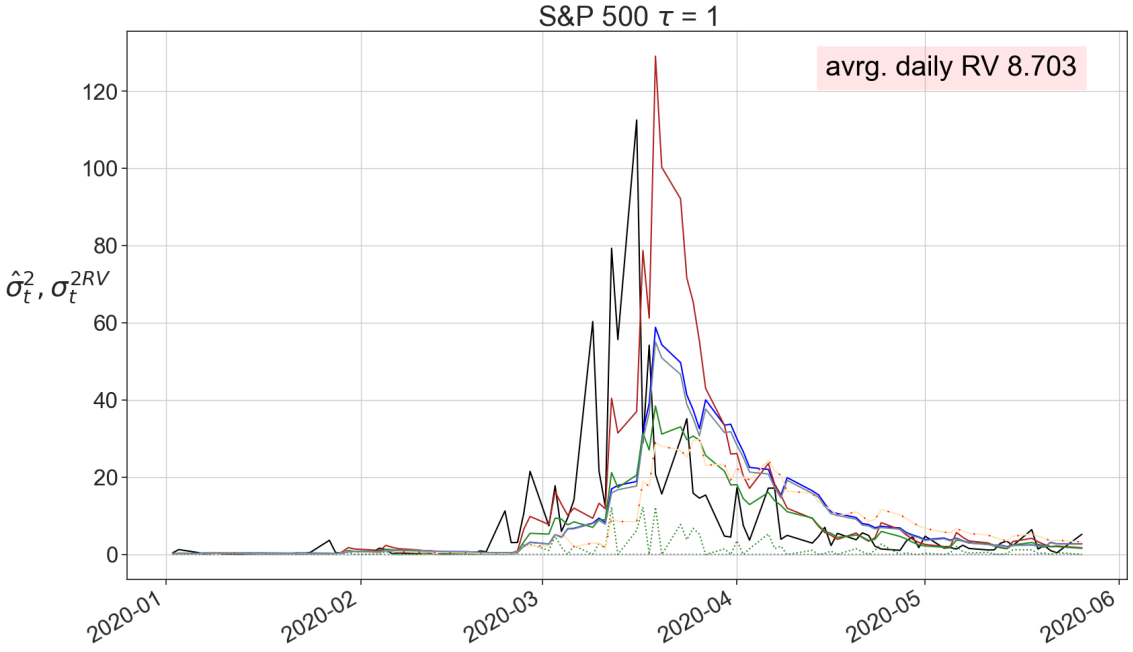


Figure 15: S&P 500 Variance forecasts 2020, one-day-ahead

of the variance forecasts of horizons 5 and 20 show that all models converge to a certain variance level.

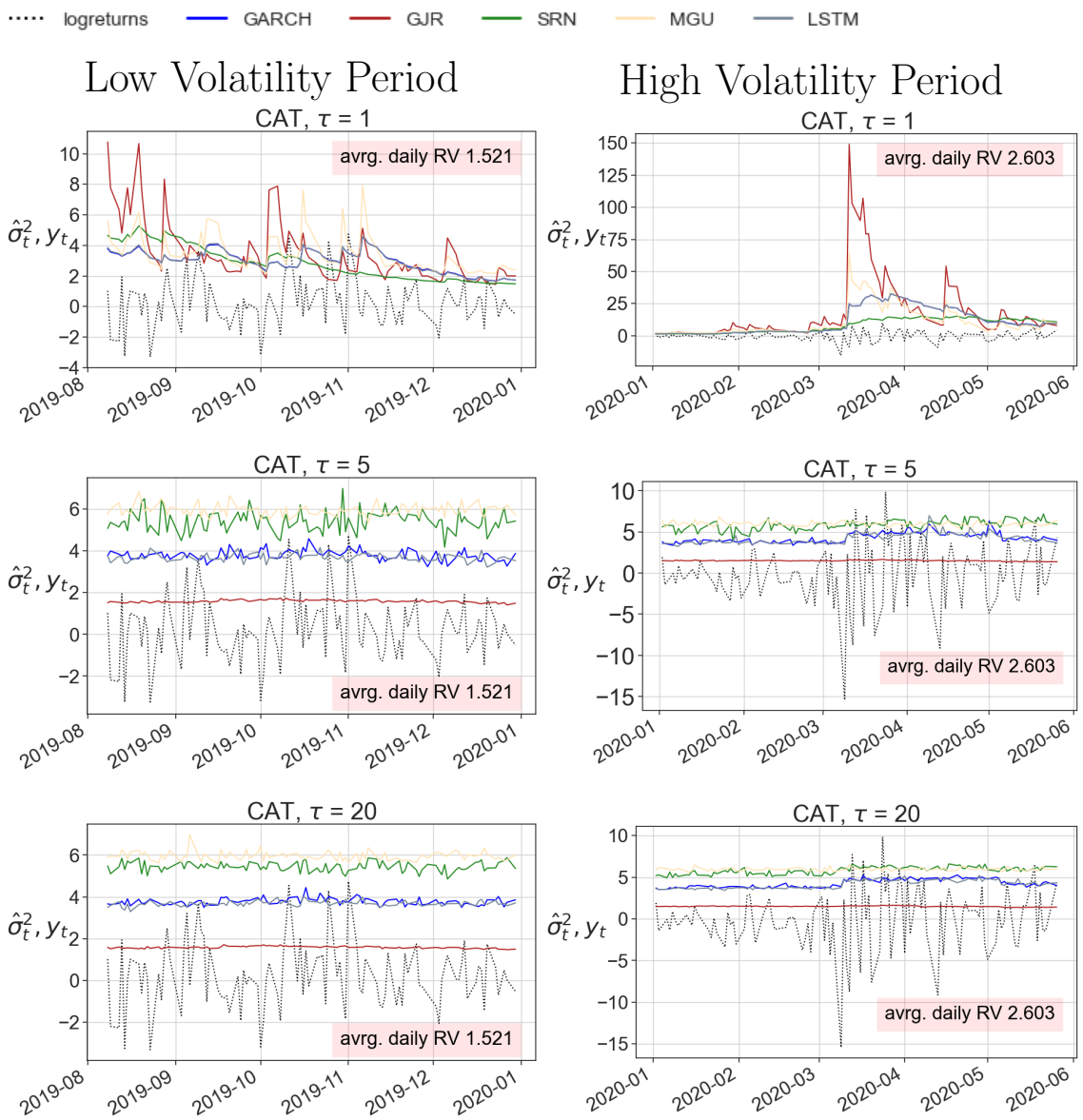


Figure 16: Conditional variance forecasts for Caterpillar Inc.

To evaluate the forecasting power of the RECH models, their predictive losses are put into relation to the losses of the GARCH(1,1) model. Those relative losses are summarized in boxplots in the figures (17 - 22) for a horizon of one day and in figures (24 - 29) and (30 - 35) for horizons of five and twenty days. For loss functions the MAE and MSE are used once again. To investigate systematic differences in forecasting power during low or high volatility periods, two subsamples are considered next to the full sample. They contain one month of low volatility, January 2020 with an average realized market variance of 0.56, and one month of high volatility,

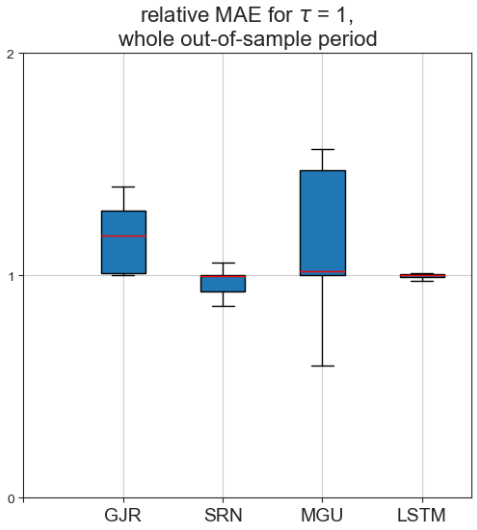


Figure 17: $\tau = 1$, metric: MAE

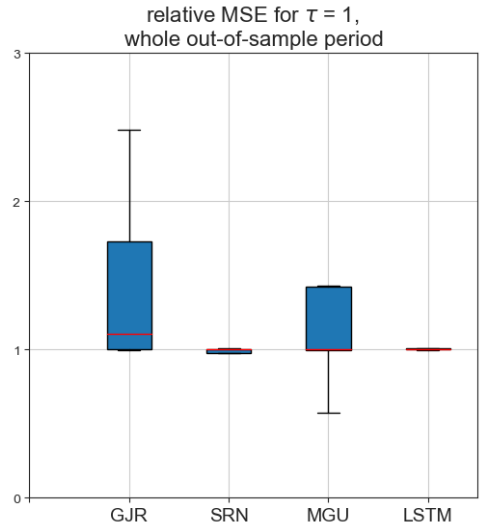


Figure 18: $\tau = 1$, metric: MSE

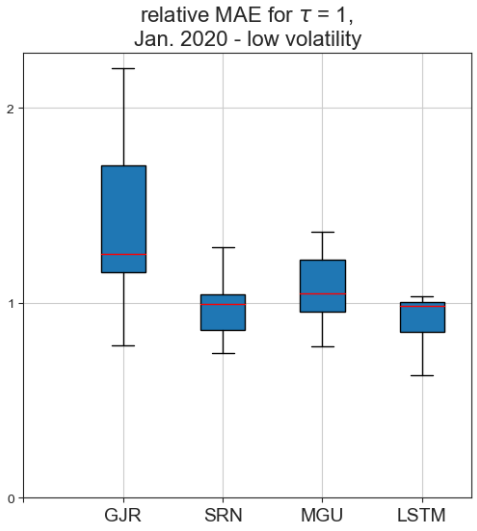


Figure 19: $\tau = 1$, metric: MAE

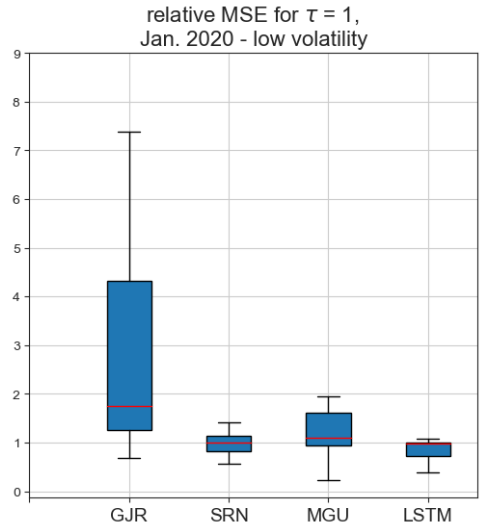


Figure 20: $\tau = 1$, metric: MSE

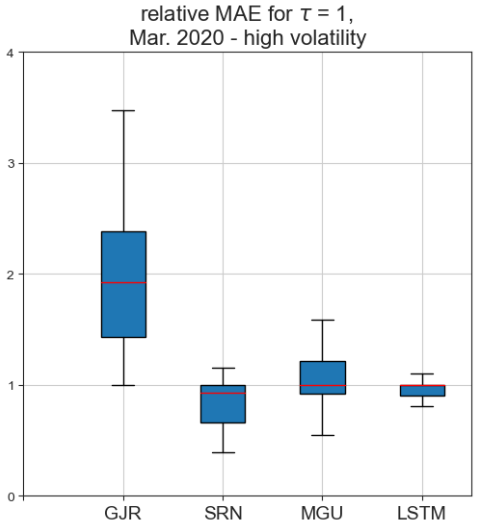


Figure 21: $\tau = 1$, metric: MAE

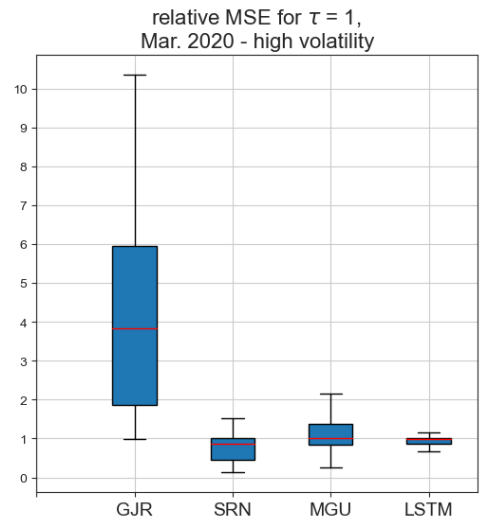


Figure 22: $\tau = 1$, metric: MSE

March 2020 with an average realized market variance of 28.92. The boxplot fences are computed as 1.5 times the interquartile range and for visual reasons, the outliers are not displayed.

For a horizon of one day, the losses are very similar in the low, high and whole out-of-sample period. SRN-GARCH and LSTM-GARCH losses are very close to the GARCH losses and tend to lie below slightly below. MGU-GARCH losses are spread more and lie rather above the GARCH losses for both subsamples. For the whole out-of-sample period, the whole interquartile range of MGU-GARCH losses lies above the GARCH losses. They indicate that the SRN-GARCH performs slightly better than the GARCH model in the selected high volatility period and in the whole out-of-sample period in terms of MAE. There are no quality differences in the low volatility period. The LSTM-GARCH outperforms the GARCH model in the low volatility period and slightly in the high volatility period. In the whole out-of-sample period there are no differences in precision.

The robust measures suppress the outliers which should not be dismissed: MGU-GARCH and LSTM-GARCH both have five very high outliers in their average loss, namely in the assets (CRM, CVX, KO, TRV, V) and (BA, GS, KO, MMM, VZ, WMT) respectively. For each of these series, the models predicted excessively high variances throughout some points in the series. The selected low volatility period does not include any of those outliers whereas the selected high volatility period does contain an outlier from the MGU-GARCH for CRM and from the LSTM-GARCH for WMT.

Being part of the MCS means that on a certain level of confidence, the models within the set have a significantly lower forecasting error than the models that are part of the original, full set of models considered. The MCSs are all computed on a significance level of $\alpha = 5\%$. When multiple models are selected, then the models within the set do not significantly outperform each other.

The MCSs are reported in Tables (5, 7, and 6). Throughout the horizon of one day within the whole out-of-sample period (Table 5) the SRN-GARCH is preferred, being part of the MCSs of 12 and 13 assets based on MAE and MSE respectively. The SRN-GARCH is closely followed by the LSTM-GARCH with 11 and 8 inclusions. Within the low volatility period, the SRN- and LSTM-GARCH are both included in the MCSs for 22 assets as a sum of their inclusions based on MAE and MSE. In the high volatility period, the SRN-GARCH is clearly preferred with 17 and 15 inclusions respectively.

For a forecasting horizon of 5 trading days in the whole out-of-sample period, the GJR-GARCH is preferred based on MAE. In the low volatility period, the GJR-

GARCH is preferred based on both MAE and MSE. In the high volatility period, the SRN-GARCH performs best with 8 and 13 inclusions.

Forecasting 20 trading days the SRN-GARCH outperforms the other models in the whole out-of-sample period with 14 inclusions in each MAE- and MSE-based MCSs, followed by the LSTM-GARCH with 11 and 8, respectively. They perform similarly in the low volatility period with 12 and 10 inclusions of the SRN-GARCH and 10 and 12 inclusions of the LSTM-GARCH. In the high volatility period, the SRN-GARCH outperforms all other models and is included 17 and 15 assets respectively.

Overall the SRN-GARCH performs best and is included in most MCSs which indicates that the SRN-GARCH has the overall best realised variance forecasting precision in the group of considered models for a set of assets.

4.3 Portfolio Application

To evaluate the economic value of the volatility forecasts a portfolio experiment is conducted, using the volatility forecasts as information for an agent to formulate her portfolio choice. This was used by Wang et al. (2016) to assess the economic gains of volatility forecasting precision. This is done for the full out-of-sample period and for two subsamples, one month of very low volatility (January 2020) and one of very high volatility (March 2020). The same mean-variance utility investor who builds portfolios out of one risky and one safe asset each is considered. Her utility from investing is given by

$$U_t(y_t) = E_t(w_t y_t + y_{t,f}) - \frac{1}{2}\nu \mathrm{Var}_t(w_t y_t + y_{t,f}), \tag{73}$$

with w_t, the portfolio weight of the risky asset, y_t, the risky asset's excess return, $y_{t,f}$ the return of the safe asset and ν, a risk aversion coefficient. Hence, the investor gains utility from the expected return of the portfolio at the expense of a penalty from return variation scaled by ν. She maximises her utility by choosing optimal portfolio weights for the upcoming period according to her beliefs as

$$w_t^* = \frac{1}{\nu}\Big(\frac{\hat{y}_{t+1}}{\hat{\sigma}_{t+1}^2}\Big), \tag{74}$$

which leads to a portfolio return at day $t+1$ of

$$Y_{t+1} = w_t^* y_{t+1} + y_{t+1,f}. \tag{75}$$

	MAE			MSE		
	1	5	20	1	5	20
AAPL	SRN	LSTM	SRN	GJR	MGU	GJR
AMGN	SRN	GJR	SRN	MGU	*****	MGU
AXP	MGU	MGU	MGU	*****	SRN	GJR
BA	MGU	GJR	MGU	MGU	GJR	MGU
CAT	SRN	GJR	SRN	SRN	GJR	SRN
CRM	SRN	LSTM	SRN	SRN	LSTM	SRN
CSCO	LSTM	LSTM	LSTM	GJR	MGU	GJR
CVX	LSTM	LSTM	LSTM	GJR	SRN	GJR
DIS	SRN	GJR	SRN	SRN	GJR	SRN
GS	LSTM	LSTM	LSTM	LSTM	LSTM	LSTM
HD	*	MGU	SRN	SRN	MGU	SRN
HON	LSTM	LSTM	LSTM	LSTM	LSTM	LSTM
IBM	SRN	GJR	SRN	SRN	MGU	SRN
INTC	SRN	GJR	SRN	GJR	SRN	GJR
JNJ	MGU	GJR	MGU	MGU	GARCH	MGU
JPM	MGU	GJR	MGU	MGU	GJR	MGU
KO	SRN	GJR	SRN	SRN	GARCH	****
MCD	LSTM	GJR	**	SRN	GARCH	SRN
MMM	LSTM	LSTM	LSTM	LSTM	LSTM	LSTM
MRK	LSTM	LSTM	LSTM	LSTM	LSTM	LSTM
MSFT	GARCH	GJR	GARCH	GJR	GARCH	GJR
NKE	MGU	GJR	MGU	SRN	MGU	SRN
PG	SRN	GJR	SRN	SRN	SRN	SRN
TRV	LSTM	LSTM	LSTM	LSTM	LSTM	LSTM
UNH	LSTM	GJR	LSTM	SRN	SRN	SRN
V	LSTM	SRN	LSTM	GJR	GARCH	GJR
VZ	GARCH	GJR	GARCH	GARCH	GARCH	*
WBA	SRN	GJR	**	SRN	***	SRN
WMT	LSTM	LSTM	LSTM	LSTM	LSTM	LSTM
S&P 500	SRN	GJR	SRN	SRN	GJR	SRN

*: SRN, GARCH; **: MGU, GARCH, SRN, LSTM
: SRN, GARCH, LSTM, GJR; *: SRN, LSTM
*****: All models selcted

Table 5: MCSs based on MAE and MSE for the whole out-of-sample period

	MAE			MSE		
	1	5	20	1	5	20
AAPL	SRN	GARCH	SRN	GJR	MGU	GJR
AMGN	SRN	SRN	SRN	MGU	MGU	MGU
AXP	SRN	LSTM	SRN	SRN	GJR	SRN
BA	SRN	SRN	SRN	SRN	SRN	SRN
CAT	SRN	GARCH	SRN	SRN	SRN	SRN
CRM	LSTM	LSTM	LSTM	LSTM	LSTM	LSTM
CSCO	LSTM	LSTM	LSTM	GJR	MGU	GJR
CVX	MGU	GARCH	MGU	MGU	GJR	MGU
DIS	SRN	GJR	SRN	SRN	LSTM	SRN
GS	LSTM	SRN	LSTM	LSTM	SRN	LSTM
HD	SRN	GARCH	SRN	SRN	GJR	SRN
HON	SRN	SRN	SRN	SRN	SRN	SRN
IBM	SRN	LSTM	SRN	SRN	GARCH	SRN
INTC	SRN	GJR	SRN	GJR	MGU	GJR
JNJ	MGU	MGU	MGU	MGU	MGU	MGU
JPM	MGU	GJR	MGU	MGU	SRN	MGU
KO	SRN	GJR	SRN	SRN	SRN	SRN
MCD	SRN	SRN	SRN	SRN	SRN	SRN
MMM	SRN	GARCH	SRN	SRN	SRN	SRN
MRK	LSTM	GARCH	LSTM	SRN	GARCH	SRN
MSFT	LSTM	GJR	LSTM	GJR	GARCH	GJR
NKE	SRN	LSTM	SRN	SRN	SRN	SRN
PG	SRN	SRN	SRN	SRN	SRN	SRN
TRV	MGU	GARCH	MGU	MGU	GARCH	MGU
UNH	LSTM	SRN	LSTM	LSTM	SRN	LSTM
V	MGU	GJR	MGU	MGU	SRN	MGU
VZ	SRN	SRN	SRN	SRN	LSTM	SRN
WBA	MGU	GJR	MGU	GJR	MGU	GJR
WMT	GARCH	GARCH	GARCH	GARCH	SRN	GARCH
S&P 500	SRN	GJR	SRN	SRN	MGU	SRN

Table 6: MCSs based on MAE and MSE for the high volatility period, March 2020

	MAE			MSE		
	1	5	20	1	5	20
AAPL	SRN	garch	SRN	SRN	garch	SRN
AMGN	LSTM	GJR	LSTM	LSTM	GJR	LSTM
AXP	SRN	MGU	SRN	SRN	MGU	SRN
BA	LSTM	LSTM	LSTM	LSTM	GJR	LSTM
CAT	SRN	GJR	SRN	SRN	GJR	SRN
CRM	LSTM	LSTM	LSTM	LSTM	LSTM	LSTM
CSCO	LSTM	LSTM	LSTM	LSTM	LSTM	LSTM
CVX	MGU	MGU	MGU	MGU	MGU	MGU
DIS	SRN	GJR	SRN	SRN	GJR	SRN
GS	LSTM	LSTM	LSTM	LSTM	LSTM	LSTM
HD	MGU	MGU	MGU	MGU	MGU	MGU
HON	SRN	MGU	SRN	SRN	LSTM	SRN
IBM	SRN	GJR	SRN	SRN	GJR	SRN
INTC	SRN	GJR	SRN	SRN	GJR	SRN
JNJ	MGU	GJR	MGU	MGU	GJR	MGU
JPM	MGU	GJR	MGU	MGU	GJR	MGU
KO	SRN	GJR	SRN	SRN	GJR	SRN
MCD	LSTM	GJR	GJR LSTM	LSTM	GJR	LSTM
MMM	LSTM	LSTM	LSTM	LSTM	LSTM	LSTM
MRK	SRN	LSTM	SRN	SRN	LSTM	SRN
MSFT	LSTM	GJR	LSTM	LSTM	GJR	LSTM
NKE	SRN	GJR	SRN	SRN	GJR	SRN
PG	LSTM	GJR	LSTM	LSTM	GJR	LSTM
TRV	MGU	MGU	MGU	MGU	SRN	MGU
UNH	SRN	GJR	SRN	LSTM	GJR	LSTM
V	GJR	GJR	GJR	GJR	GJR	GJR
VZ	garch	GJR	garch	garch	GJR	garch
WBA	MGU	LSTM	MGU	MGU	LSTM	MGU
WMT	LSTM	LSTM	LSTM	LSTM	LSTM	LSTM
S&P 500	SRN	LSTM	SRN	LSTM	LSTM	LSTM

Table 7: MCSs based on MAE and MSE for the low volatility period, January 2020

For simplicity the in-sample mean of the historical daily returns is used to estimate y_{t+1}. Thus, we have $\hat{y}_t = \hat{y}$. For the variance $\hat{\sigma}^2_{t+1}$, however, the GARCH and RECH forecasts are used. For the risk-free rate, the 1-month treasury bill rate is used and converted into daily returns.

When at day t the model predicts high volatility of the risky asset for the day $t+1$, the investor anticipates high volatility and reduces the weight of the risky asset for day $t+1$ to ensure a low volatility of her portfolio. Each model leads to a unique portfolio that changes over time. They are evaluated by calculating the annualised Sharpe ratio:

$$\text{SR}_{\text{ann}} = \sqrt{252}\frac{\frac{1}{T}\sum_{t=1}^{T}(Y_t - y_{t,f})}{\frac{1}{T}\sum_{t=1}^{T}(Y_t - y_{t,f})^2} = \sqrt{252} \quad \frac{\hat{\mu}_P}{\hat{\sigma}^2_P}. \tag{76}$$

The portfolio weights are constructed by using the historical average return as a constant expected return. When a risky asset has an average return below the risk-free rate or even a negative return, the portfolio return can become negative, and with that, the Sharpe ratio.

To calculate with a constant risk-free rate, the average daily treasury bill rate is used. This is done for each considered sample period individually, which is 0.027% in the whole out-of-sample period, 0.031% in the selected low volatility period and 0.0027% in the selected high volatility period, as the treasury bill rates plummeted in that exact period of market distress. The data is taken from the FRED database[6]. For the degree of risk aversion, ν, a value of 3 is chosen.

4.3.1 Results

The annualised Sharpe ratios of the portfolios generated based on each model's one-day-ahead variance forecasts are displayed in Table (8) and visualised in Figure (23).

In the whole out-of-sample period, the GARCH(1,1) generates the portfolios with the highest Sharpe ratios (for 12 assets), followed by the LSTM-GARCH performing best for 9 assets. Within the selected low volatility period, the GJR-GARCH generates the portfolios with the highest Sharpe ratios (9), followed by the GARCH(1,1) (8), although the GJR had the highest losses in realized variance forecasting. In the high volatility period, the portfolios based on the MGU-GARCH have the highest number of highest Sharpe ratios with 9 portfolios, followed by the LSTM-GARCH with 8 portfolios.

[6] `https://fred.stlouisfed.org/series/DGS1`

	full out-of-sample					low volatility, Jan. 2020					high volatility, Mar. 2020				
	GARCH	GJR	SRN	MGU	LSTM	GARCH	GJR	SRN	MGU	LSTM	GARCH	GJR	SRN	MGU	LSTM
AAPL	2.05	2.06	2.07	2.25*	2.05	3.73	4.72*	4.18	4.03	3.69	1.14*	-0.03	0.12	0.13	1.11
AMGN	0.94	0.95*	0.91	0.78	0.9	-5.53	-5.22*	-5.57	-5.61	-5.4	1.47	0.88	0.97	1.93*	1.48
AXP	1.33*	1.07	1.19	1.25	1.33*	3.29*	3.11	2.84	3.1	3.28	-2.09*	-2.72	-2.4	-2.1	-2.1
BA	1.53*	1.45	1.37	1.48	1.17	-1.1*	-2.05	-1.17	-1.38	-1.11	-4.79	-4.46	-4.82	-2.79*	-4.9
CAT	1.17	1.43*	1.17	0.85	1.15	-5.47	-3.68*	-5.45	-4.8	-5.46	-3.3	-2.88	-2.38*	-2.62	-3.29
CRM	1.33*	1.11	1.21	0.37	1.16	6.15*	6.13	6.0	6.07	5.42	-2.72	-2.58	-2.64	-1.02*	-2.35
CSCO	0.5	0.62	0.44	0.47	0.76*	-1.89	-1.4	-1.3*	-1.77	-2.01	-1.52	-0.89	-0.29*	-1.2	-0.75
CVX	-0.1	-0.06*	-0.08	-0.55	-0.49	-8.47	-8.49	-8.91	-8.29	-8.11*	-2.42	-2.78	-2.78	-3.35	-2.39*
DIS	0.64	0.65*	0.61	0.63	0.64	-2.93	-3.1	-3.04	-2.79*	-2.92	-3.16	-3.26	-2.78*	-2.97	-3.15
GS	0.26	0.09	0.23	0.31	0.39*	2.8	3.7*	2.75	3.0	3.1	-2.52	-2.92	-2.62	-2.45	-2.31*
HD	1.41	1.56*	1.35	1.32	1.37	4.07	3.88	3.05	4.15*	3.95	1.46	0.39	0.53	1.51*	1.37
HON	1.24	1.02	1.17	1.46*	1.23	-1.07	-0.68	-1.09	3.64*	-1.07	-2.23	-2.83	-2.65	0.68*	-2.23
IBM	0.44	0.51*	0.39	0.27	0.44	4.67*	4.5	3.89	4.01	4.62	-1.99	-1.56*	-2.74	-2.36	-2.0
INTC	0.81	0.75	0.66	0.98*	0.74	4.13	3.72	3.75	4.47*	3.86	-1.06	-1.1	-0.9	-0.43*	-1.14
JNJ	0.7	0.61	0.67	0.58	0.74*	3.33*	2.73	2.87	3.03	3.32	1.93	0.29	0.99	1.22	2.89*
JPM	0.64*	0.5	0.53	0.45	0.61	-2.09	-1.86*	-2.17	-2.08	-2.12	-2.47	-3.08	-2.78	-2.04*	-2.48
KO	1.13*	0.87	1.04	1.01	1.13*	4.51*	4.43	4.35	4.46	4.5	0.51	-1.4	-0.29	1.04*	0.65
MCD	1.59*	1.49	1.54	1.52	1.4	7.13*	7.1	6.81	4.68	7.07	-0.12	0.51*	-0.03	0.51	-1.21
MMM	0.57*	0.15	0.4	0.43	-0.47	-4.85	-3.57*	-4.63	-4.91	-5.01	-0.97	-0.42*	-1.13	-0.79	-0.85
MRK	0.77	0.82	0.65	1.1*	0.65	-3.59	-3.8	-3.62	-1.28*	-4.98	2.19	1.68	0.36	-0.8	4.67*
MSFT	1.97	1.89	1.98	1.99*	1.9	4.91*	4.48	4.91	4.9	4.84	0.48	1.21*	0.47	0.4	0.79
NKE	1.11	1.01	1.04	1.11	1.18*	-3.49	-2.48*	-3.12	-4.2	-3.71	-2.88	-3.53	-1.84	-1.38*	-2.4
PG	1.2*	1.04	1.15	1.05	1.2	0.24	-0.04	-0.02	0.52*	0.26	2.43*	0.3	1.58	2.26	2.41
TRV	0.48	0.14	0.32	0.31	0.51*	-1.16*	-1.66	-1.4	-1.29	-1.49	-0.12	-1.4	-0.87	-0.0	0.6*
UNH	1.1	0.62	0.95	1.12*	1.07	-2.62	-1.63	-1.87	-1.33*	-2.75	0.19	-2.15	-1.17	-0.34	0.26*
V	1.72*	1.56	1.59	1.68	1.72*	4.09	4.83*	4.2	4.12	4.1	-0.8*	-2.4	-0.88	-0.88	-0.8
VZ	0.77*	0.67	0.75	0.74	0.77*	-1.53	-1.37	-1.72	-1.81	4.07*	2.5*	2.47	1.88	2.1	-0.41
WBA	-0.43	-0.4*	-0.52	-0.61	-0.51	-6.43	-6.64	-5.66*	-6.58	-6.36	1.27	0.36	1.24	0.45	1.5*
WMT	1.43*	1.34	1.42	1.29	0.89	-3.28	-2.71*	-3.11	-3.08	-2.94	2.58	-0.15	1.51	3.01	3.38*
S&P 500	1.24*	0.85	1.03	0.95	1.24*	0.98	1.14*	0.98	0.8	0.96	-0.19*	-1.83	-1.7	-3.07	-0.23

Table 8: Sharpe ratios of respective two-asset portfolios of the full sample

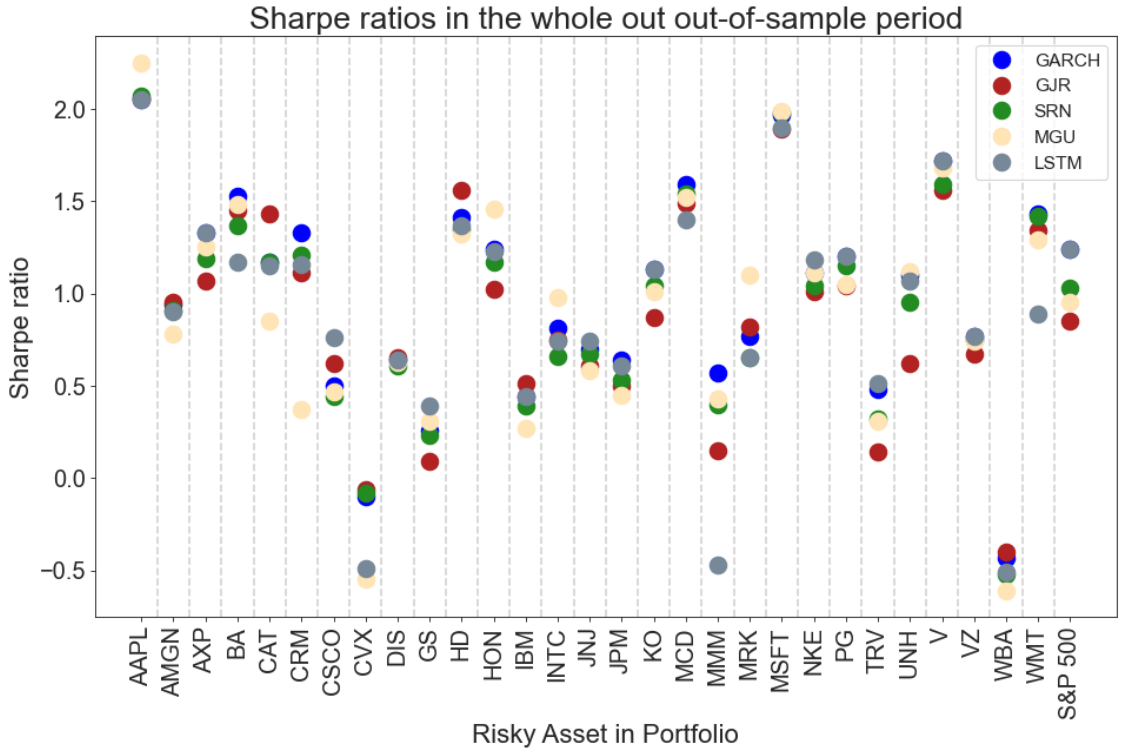


Figure 23: Sharpe ratios of respective two-asset portfolios of the full sample - graphically

5 Discussion

The MLE method and the necessary numerical optimisation might be a pitfall that leads to parameter mis-estimation. Especially given the high number of parameters of some of the RECH models, the optimisation algorithm can get stuck in a local optimum. The estimation of a simulated SRN-GARCH indicates that the model is not able to estimate its own parameters. Other approaches to estimate the simulated data can be considered, like using randomized starting values for the optimisation algorithm in each simulation run. The absence of identification prohibits parameter interpretation. Further research could investigate the model identification of the SRN-GARCH using MCMC estimation. This will reveal whether the source of mis-estimation is the use of MLE with an SLSQP optimisation algorithm or an inherent characteristic of the SRN-GARCH and with that possibly also other RECH models.

The stationarity conditions derived in (3.3) leave a bitter taste because the parameter w in the simple recurrent neuron is the one parameter that distinguishes it from a neuron used in feedforward neural networks. Requiring it to be equal to zero indicates impracticality of RECH models. Stationarity is an important feature to ensure that the MLE is consistent, so it should not be disregarded. The importance of including the lagged value of the recurrent unit is attributable to the w parame-

ter in the SRN-GARCH while it is more complex for the other RECH models. The figures (13 - 15) show that the outputs of the recurrent unit of the SRN-GARCH are much smaller than the variance forecasts. The parameter estimates $\hat{w}$ found in Table (11) show that the parameter w only plays a minor role in the formation of the predicted variance σ_t^2. Further research should include an assessment of the importance of a recurrent structure. Alternatives can be parsing a linear function through non-linear activation functions without including past states h, which would correspond to a feedforward neuron.

The results reported in the previous section allow us to draw conclusions about the empirical performance of considered RECH models. A very important insight is that the SRN-GARCH parameters cannot be reliably identified using MLE. Only the estimates of parameter α showed signs of identification. Here, the parameter identification was only investigated estimating the parameters using the same starting values in each run. No bunching around the starting values was observed, but repeating the analysis with random starting values might generate new insights. A possible outcome could be that estimates around the true parameter value will be observed whenever the starting value is close to the true value. Here, this was observed for α but not for γ_0. This might not affect the forecasting qualities of the SRN-GARCH and RECH models in general, but without parameter identification, the model parameters lose the capability to be interpreted. This especially plays a role for the SRN-GARCH, as some parameters could otherwise be interpreted, like a negative v_1 as a leverage effect or b as a threshold parameter.

Overall, the simulations of GARCH-type processes indicate that the RECH models cannot consistently outperform the GARCH(1,1) in the in-sample fitting of non-linear time series. For the non-linear data, the RECH models only report losses about as good as those as the GARCH(1.1) losses. This is unexpected as the RECH models are designed to be able to do that. The RECH models do manage, however, to provide good fits of linear processes, as the results of the simulation of the GARCH(1,1) data show. Due to outliers in the simulations, the MGU-GARCH performs slightly better in fitting the GARCH(1,1) data than the GARCH(1,1) itself. Eliminating the outliers, the GARCH(1,1) performs best with very little difference from the RECH models. The RECH models do not manage to incorporate the leverage effects but neither does the GJR-GARCH.

The realized variance forecasting results overall indicate that the SRN-GARCH and the LSTM-GARCH outperform the GARCH(1,1) and the GJR-GARCH models. The LSTM-GARCH however, provides forecasts that are rather slightly worse than those of the SRN-GARCH, while also producing heavy outliers for 1/6 of the assets

in the sample. The spread of relative MGU-GARCH losses is much wider than those of SRN-GARCH and LSTM-GARCH and tends to perform slightly worse while producing heavy outliers for 1/6 of the assets in the sample too.

To put this slight performance increase into an economic context, the variance forecasts are used in a portfolio application to assess a possible economic significance in the performance differences. The portfolios generated by the GARCH(1,1) generate better Sharpe ratios than the RECH models in the whole out-of-sample period. Only in the high volatility period, do the RECH models have an edge. In this subsample, the model with the highest number of highest Sharpe ratios is the MGU-GARCH, although the MGU-GARCH delivered worse MAE and MSE losses than the other RECH models and with that fewer inclusions in MCSs.

The performance of RECH models varying in forecasting applications is an interesting result. The SRN- and LSTM-GARCH tend to outperform both GARCH-type models in realized variance forecasting but not when applying those forecasts in portfolio management. The key difference here is, that the realized variance forecasting is evaluated how well intra-daily return variance is predicted, while the portfolio application is evaluated based on inter-daily return variance.

The Sharpe ratio results need to be treated with caution. To illustrate why, imagine a risky asset with negative returns out-of-sample but a positive historical return - one model produces unbiased forecasts and another model produces forecasts with an upward bias. In this scenario, this model would have a high Sharpe ratio (less negative), because, with the upwardly biased variance forecasts, the weights for the risky assets are downwardly biased, hence, less of the negative returns are contributing to the portfolio and more of the safe asset with a positive return. The Sharpe ratios portfolios of different models can also have reversed signs. In a scenario where a model predicts very low variance for a certain risky asset and its respective historical average return is positive, then its weight in the portfolio will be rather high - if the actual realized returns are negative the overall portfolio return may be negative. This can stand in contrast to a portfolio based on a model with higher predicted variance resulting in lower weights for the risky asset.

The parameter estimates of the LSTM-GARCH are another critical observation. The parameters v_{11}, v_{12}, w_1, and b_c do not change from the starting values for all assets. It should be further investigated whether MLE or the SLSQP algorithm are responsible. The fact the estimates do not differ from the starting values at all suggests that there is a numerical reason in the optimisation algorithm. As the SLSQP algorithm is based on Newton's method it could mean that there is not enough non-linear information in the data to obtain a sufficient slope of the linear

approximation of the partial derivates with respect to the parameters in question.

The results for the S&P 500 are in accordance with the results of Nguyen et al. (2022) who find that the SRN-GARCH outperforms GARCH models without recurrent components as extensions. They find that GARCH models with an SRN extension generally outperform their GARCH counterparts without extensions forecasting realized variance measures for national indices. For the individual assets, however, the results are more heterogenous than what Nguyen et al. (2022) found. Here, the RECH models are not preferable throughout all assets. As Nguyen et al. (2022) found the RECH models to generally outperform regular GARCH models in realized variance forecasting, subsequent research should include how the models estimated with MCMC perform in portfolio applications.

The chosen benchmark models, the GARCH(1,1) and the GJR-GARCH do not necessarily include the best possible GARCH-type models. Some literature suggests that models like the EGARCH are preferable to the GJR-GARCH, see Brooks (1998), or preferable to the simple GARCH, see Pagan and Schwert (1990). They produce variance forecasts for the Dow Jones index and evaluate it based on squared returns. Others like Franses and Van Dijk (1996) find that both the GJR-GARCH and the EGARCH are subpar of the QGARCH in forecasting weekly stock market volatility for European indices. Hence, it is not entirely clear which GARCH-type model serves as the best benchmark for RECH models. Although other GARCH models might have further challenged the RECH models, they were not investigated to contain the scope of this thesis.

6 Conclusion

This thesis analysed a wide range of aspects of the theory and application of the novel class of RECH models proposed by Nguyen et al. (2022). First of all, a different estimation approach than the authors initially used was examined - the frequentist maximum likelihood estimation (MLE) instead of a Bayesian Markov chain Monte Carlo (MCMC) method. Secondly, two further functional forms, the MGU-GARCH and LSTM-GARCH, are considered and compared to the SRN-GARCH that has already been applied by Nguyen et al. (2022). The LSTM-GARCH however, has been analyzed in the unpublished paper Liu et al. (2023) including realized volatility as an input variable in the recurrent component.

Advances in the theoretical characteristics of the RECH models are made by deriving stationarity conditions for the SRN-GARCH and investigating model identification properties for the SRN-GARCH. The RECH models are compared to one another and to the baseline models, the GARCH(1,1) and the GJR-GARCH which allows for a simple, yet important form of non-linearity in asset return time series, leverage effects. The models are evaluated in three ways: Using synthetically generated data, the models abilities to fit time series with different levels of non-linearity are evaluated; a realized variance (RV) forecasting study is conducted and evaluated using relative errors and model confidence sets (MCSs); a portfolio application is performed where an agent uses the models' variance forecasts for portfolio management which is evaluated based on the resulting annualised Sharpe ratios.

An accompanying software package containing the estimation methods is published on `https://github.com/mbilsen/RECH_models`, allowing other researchers to build upon the MLE method and simulation based multi-period forecasting.

This thesis is a first step into the further academic exploration of RECH models. Future research should consider the parameter identification of RECH models when employing a Bayesian MCMC method. A milestone would be the derivation of stationarity conditions for general RECH processes. A key characteristic of RECH models was left out, which is the incorporation of exogenous variables in the recurrent component. Here, several different variables could be included, like trade volume, exchange rates, or realised variance. The latter was considered in the working paper Liu et al. (2023). The set of benchmark GARCH-type models was limited in this thesis. Further research could include more non-linear GARCH models like the EGARCH.

The found lack of parameter identification prevents parameter interpretation of the considered SRN-GARCH and indicates similar results for more complex RECH models. It also prevents parameter interpretation, which is one of the key features

of simple RECH models like the SRN-GARCH. Furthermore, the models' ability to fit time series of varying degrees of non-linearity is tested in simulation studies. In contrast to the results of the simulation studies of Nguyen et al. (2022), the RECH models cannot outperform the in-sample fits of a GARCH(1,1) or a GJR-GARCH. A notable difference in the simulation approaches is that in this study multiple time series are generated instead of just one. This is done to obtain robust results.

In the realized variance forecasting application using the Dow Jones assets and the S&P 500 index, the SRN-GARCH overall performs best, closely followed by the LSTM-GARCH. There is an exception for the horizon of 5 days which favours the GJR-GARCH in the whole out-of-sample period as well as in the low volatility subsample. The RECH models cannot outperform the GARCH(1,1) and GJR-GARCH throughout all assets considered. The greatest discrepancy in forecasting accuracy is found in the high volatility period of March 2020 where the SRN-GARCH produces significantly better forecasts than the other considered models for 17 or 15 assets, evaluated based on the MAE and MSE respectively. The added complexity of the LSTM neuron did not lead to better realised variance forecasts compared to the SRN-GARCH.

Those findings do not hold up when applying the variance forecasts in portfolio management. When using the model forecasts to create mean-variance-optimised portfolios the RECH models all perform worse than the GARCH(1,1) and the GJR-GARCH for the whole out-of-sample period. This implies that the application of RECH models is better for intra-day return variance than inter-day return variance. The range of data considered reveals an advantage of RECH models modelling the volatility of the Covid-19 outbreak. In this sub-sample, the SRN- and LSTM-GARCH perform best in realised variance forecasting and the MGU-GARCH in the portfolio application.

RECH models establish a way of connecting RNNs and GARCH-type models. This thesis reveals how this connection is computationally and econometrically challenging but rewards the user with an edge in conditional variance forecasting compared to simple benchmark GARCH-type models.

References

Andersen, T. G., Bollerslev, T., Diebold, F. X. and Labys, P. (2003), 'Modeling and forecasting realized volatility', *Econometrica* **71**(2), 579–625.

Bollerslev, T. (1986), 'Generalized autoregressive conditional heteroskedasticity', *Journal of econometrics* **31**(3), 307–327.

Bougerol, P. (1993), 'Kalman filtering with random coefficients and contractions', *SIAM Journal on Control and Optimization* **31**(4), 942–959.

Brooks, C. (1998), 'Predicting stock index volatility: can market volume help?', *Journal of Forecasting* **17**(1), 59–80.

Brownlees, C. T., Engle, R. F. and Kelly, B. T. (2011), 'A practical guide to volatility forecasting through calm and storm', *Available at SSRN 1502915* .

Chen, L., Pelger, M. and Zhu, J. (2023), 'Deep learning in asset pricing', *Management Science* .

Christensen, K., Siggaard, M. and Veliyev, B. (2021), 'A machine learning approach to volatility forecasting', *Available at SSRN* .

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

Donahoe, J. W. and Dorsel, V. P. (1997), *Neural network models of cognition: Biobehavioral foundations*, Elsevier.

Donaldson, R. G. and Kamstra, M. (1996), 'Forecast combining with neural networks', *Journal of Forecasting* **15**(1), 49–61.

Engle, R. F. (1982), 'Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation', *Econometrica: Journal of the econometric society* pp. 987–1007.

Francq, C. and Zakoian, J.-M. (2019), *GARCH models: structure, statistical inference and financial applications*, John Wiley & Sons.

Franses, P. H. and Van Dijk, D. (1996), 'Forecasting stock market volatility using (non-linear) garch models', *Journal of forecasting* **15**(3), 229–235.

Glosten, L. R., Jagannathan, R. and Runkle, D. E. (1993), 'On the relation between the expected value and the volatility of the nominal excess return on stocks', *The journal of finance* **48**(5), 1779–1801.

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

Hajizadeh, E., Seifi, A., Zarandi, M. F. and Turksen, I. (2012), 'A hybrid modeling approach for forecasting the volatility of s&p 500 index return', *Expert Systems with Applications* **39**(1), 431–436.

Hansen, P. R. 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.

Hansen, P. R. and Lunde, A. (2006), 'Consistent ranking of volatility models', *Journal of Econometrics* **131**(1-2), 97–121.

Hansen, P. R., Lunde, A. and Nason, J. M. (2011), 'The model confidence set', *Econometrica* **79**(2), 453–497.

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

Jordan, M. I. (1986), 'Serial order: a parallel distributed processing approach. technical report, june 1985-march 1986'.
URL: *https://www.osti.gov/biblio/6910294*

Kim, H. Y. and Won, C. H. (2018), 'Forecasting the volatility of stock price index: A hybrid model integrating lstm with multiple garch-type models', *Expert Systems with Applications* **103**, 25–37.

Kraft, D. (1988), 'A software package for sequential quadratic programming', *Forschungsbericht- Deutsche Forschungs- und Versuchsanstalt fur Luft- und Raumfahrt* .

Lee, O. and Shin, D. (2005), 'On stationarity and β-mixing property of certain nonlinear garch (p, q) models', *Statistics & probability letters* **73**(1), 25–35.

Lipton, Z. C., Berkowitz, J. and Elkan, C. (2015), 'A critical review of recurrent neural networks for sequence learning', *arXiv preprint arXiv:1506.00019* .

Liu, C., Wang, C., Tran, M.-N. and Kohn, R. (2023), 'Realized recurrent conditional heteroskedasticity model for volatility modelling', *arXiv preprint arXiv:2302.08002* .

Maknickienė, N. and Maknickas, A. (2012), 'Application of neural network for forecasting of exchange rates and forex trading', pp. 10–11.

Nguyen, T.-N., Tran, M.-N. and Kohn, R. (2022), 'Recurrent conditional heteroskedasticity', *Journal of Applied Econometrics* **37**(5), 1031–1054.

Pagan, A. R. and Schwert, G. W. (1990), 'Alternative models for conditional stock volatility', *Journal of econometrics* **45**(1-2), 267–290.

Reisenhofer, R., Bayer, X. and Hautsch, N. (2022), 'Harnet: A convolutional neural network for realized volatility forecasting', *arXiv preprint arXiv:2205.07719* .

Rodikov, G. and Antulov-Fantulin, N. (2022), 'Can lstm outperform volatility-econometric models?', *arXiv preprint arXiv:2202.11581* .

Roh, T. H. (2007), 'Forecasting the volatility of stock price index', *Expert Systems with Applications* **33**(4), 916–922.

Sharma, P. et al. (2015), 'Forecasting stock index volatility with garch models: international evidence', *Studies in Economics and Finance* .

Straumann, D. (2005), 'Estimation in conditionally heteroscedastic time series models', *Lecture Notes in Statistics* **181**.

Wang, Y., Ma, F., Wei, Y. and Wu, C. (2016), 'Forecasting realized volatility in a changing world: A dynamic model averaging approach', *Journal of Banking & Finance* **64**, 136–149.

Zhou, G.-B., Wu, J., Zhang, C.-L. and Zhou, Z.-H. (2016), 'Minimal gated unit for recurrent neural networks', *International Journal of Automation and Computing* **13**(3), 226–234.

A Relative Errors for Horizons $\tau = 5$ and $\tau = 20$

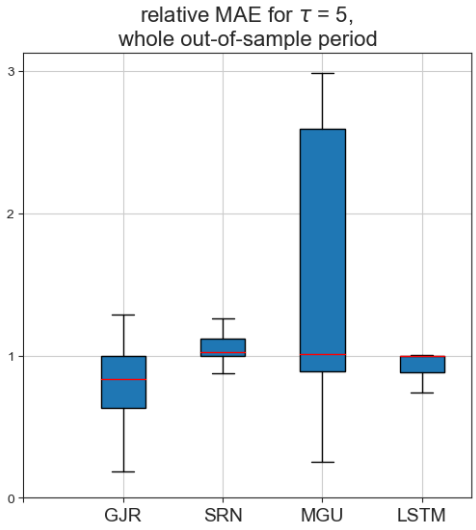


Figure 24: $\tau = 5$, metric: MAE

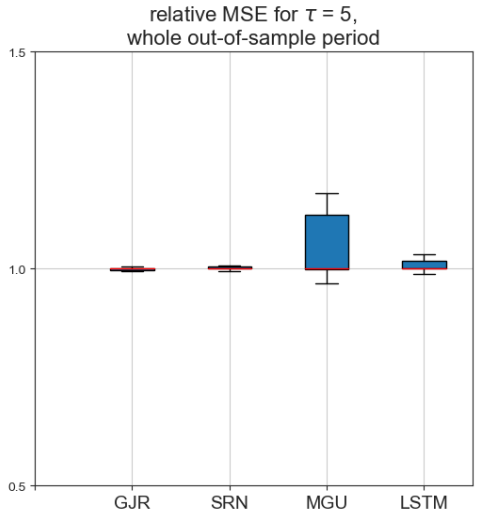


Figure 25: $\tau = 5$, metric: MSE

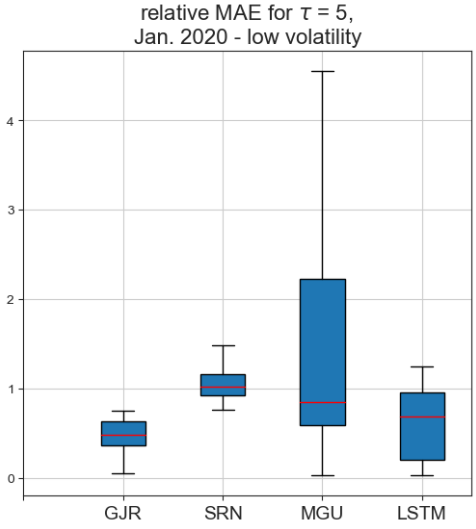


Figure 26: $\tau = 5$, metric: MAE

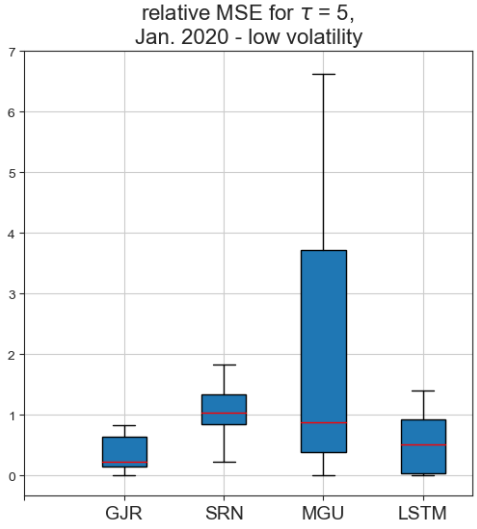


Figure 27: $\tau = 5$, metric: MSE

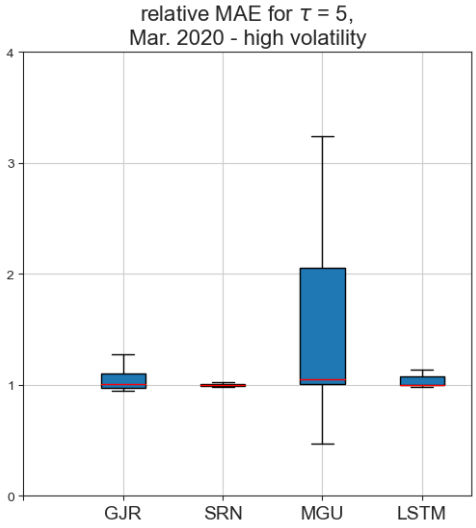


Figure 28: $\tau = 5$, metric: MAE

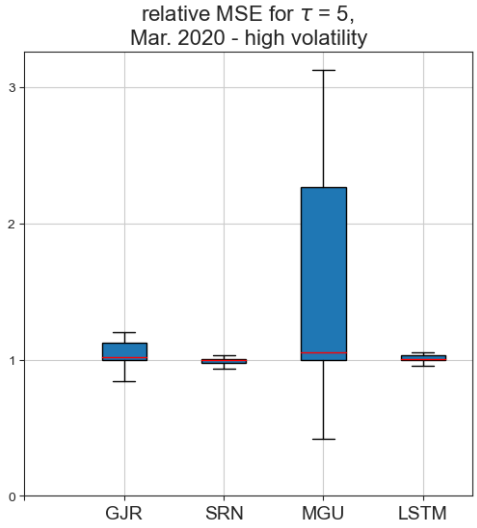


Figure 29: $\tau = 5$, metric: MSE

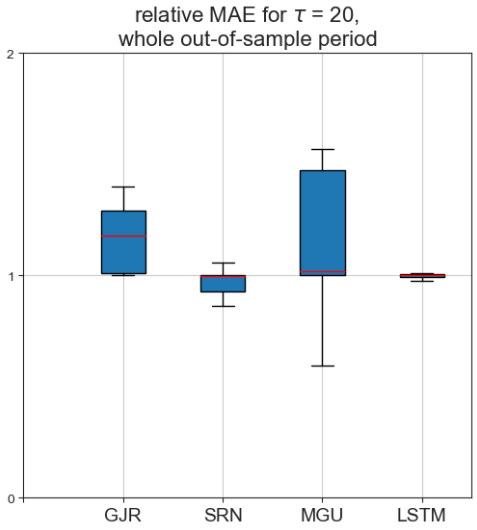


Figure 30: $\tau = 20$, metric: MAE

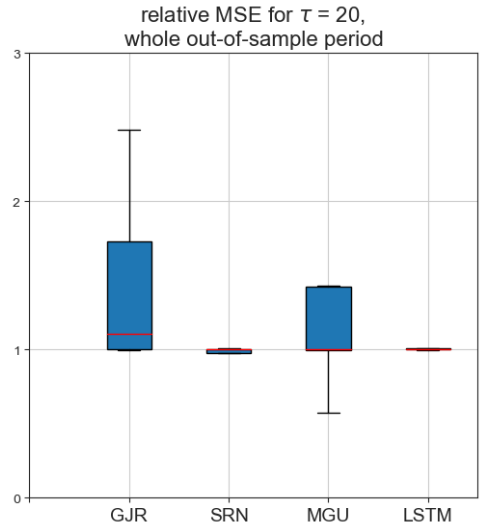


Figure 31: $\tau = 20$, metric: MSE

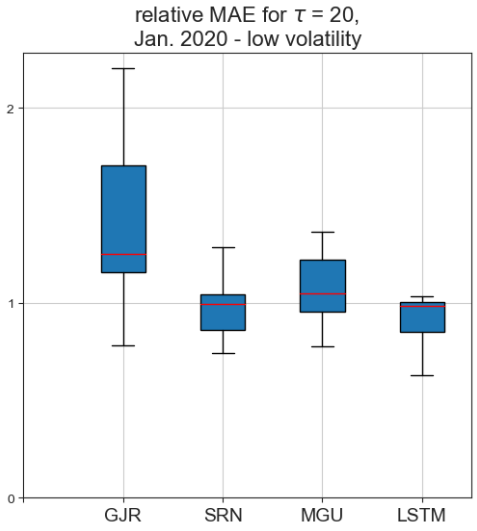


Figure 32: $\tau = 20$, metric: MAE

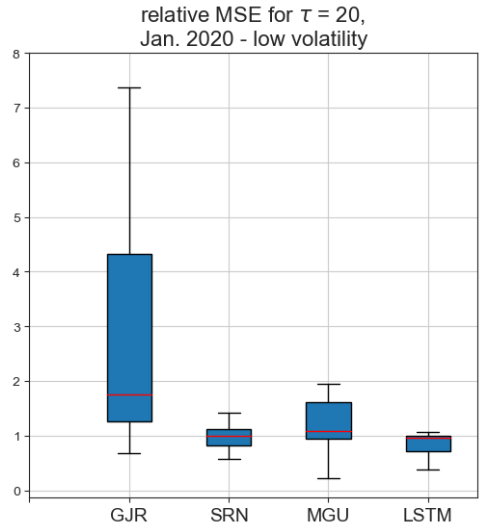


Figure 33: $\tau = 20$, metric: MSE

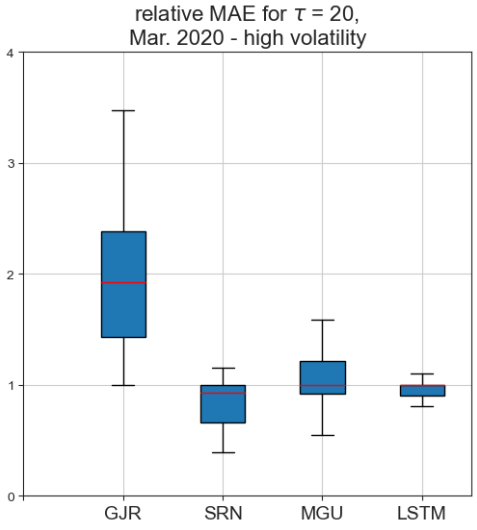


Figure 34: $\tau = 20$, metric: MAE

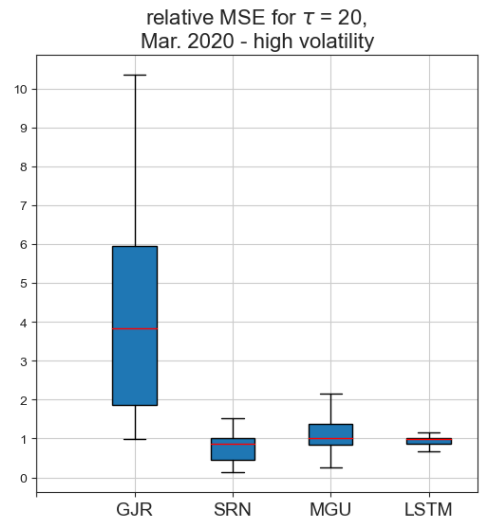


Figure 35: $\tau = 20$, metric: MSE

B Parameter Tables

This section features the parameter estimates of the last model estimation, the almost full sample, from 2005 to September 2020.

Asset	ω	α	β
AAPL	0.16	0.11	0.84
AMGN	0.11	0.09	0.87
AXP	0.04	0.11	0.89
BA	0.08	0.09	0.89
CAT	0.08	0.09	0.90
CRM	0.05	0.07	0.93
CSCO	0.23	0.11	0.83
CVX	0.05	0.10	0.88
DIS	0.08	0.10	0.88
GS	0.04	0.10	0.90
HD	0.07	0.13	0.84
HON	0.04	0.10	0.88
IBM	0.15	0.13	0.80
INTC	0.06	0.05	0.93
JNJ	0.04	0.11	0.86
JPM	0.09	0.13	0.85
KO	0.03	0.07	0.90
MCD	0.06	0.09	0.86
MMM	0.02	0.05	0.94
MRK	0.16	0.14	0.78
MSFT	0.16	0.13	0.82
NKE	0.23	0.12	0.81
PG	0.07	0.14	0.81
TRV	0.07	0.13	0.85
UNH	0.03	0.07	0.92
V	0.11	0.14	0.81
VZ	0.09	0.13	0.81
WBA	0.37	0.14	0.76
WMT	0.10	0.13	0.82
S&P 500	0.03	0.16	0.82

Table 9: GARCH parameter estimates

Asset	ω	α	ρ	β
AAPL	0.29	0.02	0.37	0.73
AMGN	0.34	0.02	0.00	0.69
AXP	0.11	0.01	0.37	0.80
BA	0.34	0.08	0.14	0.68
CAT	0.30	0.08	0.00	0.71
CRM	0.33	0.00	0.33	0.79
CSCO	0.67	0.00	0.04	0.63
CVX	0.12	0.00	0.42	0.77
DIS	0.29	0.08	0.00	0.67
GS	0.32	0.10	0.00	0.68
HD	0.20	0.04	0.50	0.70
HON	0.10	0.00	0.34	0.78
IBM	0.32	0.07	0.00	0.62
INTC	0.58	0.08	0.09	0.60
JNJ	0.16	0.03	0.00	0.66
JPM	0.21	0.05	0.00	0.72
KO	0.21	0.03	0.05	0.62
MCD	0.20	0.04	0.00	0.64
MMM	0.22	0.00	0.20	0.72
MRK	0.22	0.03	0.00	0.70
MSFT	0.36	0.09	0.00	0.64
NKE	0.34	0.01	0.00	0.70
PG	0.13	0.07	0.13	0.67
TRV	0.19	0.09	0.00	0.67
UNH	0.19	0.03	0.13	0.76
V	0.25	0.03	0.47	0.67
VZ	0.23	0.11	0.00	0.59
WBA	1.26	0.13	0.00	0.35
WMT	0.36	0.15	0.00	0.49
S&P 500	0.04	0.00	0.43	0.78

Table 10: GJR-GARCH parameter estimates

Asset	α	β	γ_0	γ_1	v_1	v_2	w	b
AAPL	0.05	0.87	0.00	0.08	-6.64	-3.35	-0.22	5.85
AMGN	0.04	0.89	0.06	0.07	-3.27	-0.60	-0.68	-0.01
AXP	0.02	0.91	0.00	0.71	-0.19	-0.02	0.00	0.10
BA	0.06	0.86	0.15	0.12	-0.56	0.58	-0.00	-1.80
CAT	0.01	0.93	0.07	0.11	-1.05	0.30	0.13	-1.70
CRM	0.03	0.91	0.00	0.10	-3.76	-0.17	-0.00	0.55
CSCO	0.01	0.86	0.21	0.12	-6.71	3.32	-0.85	-24.97
CVX	0.02	0.89	0.00	1.97	-0.07	0.00	-0.00	0.03
DIS	0.03	0.86	0.14	0.16	-0.93	0.74	-0.03	-2.85
GS	0.08	0.85	0.15	1.54	-0.03	0.03	0.27	-0.13
HD	0.08	0.77	0.05	0.66	-0.14	0.10	0.29	0.03
HON	0.00	0.88	0.06	0.35	-0.57	0.10	-0.03	-0.14
IBM	0.06	0.91	0.00	0.02	-90.45	-109.77	2.11	74.54
INTC	0.01	0.91	0.13	0.09	-1.09	1.10	-0.00	-4.68
JNJ	0.02	0.87	0.04	0.35	-0.32	0.02	0.23	-0.01
JPM	0.06	0.76	0.10	2.18	-0.08	0.06	-0.21	-0.00
KO	0.04	0.91	0.02	0.02	-15.86	0.42	-1.92	-4.50
MCD	0.02	0.88	0.06	0.23	-0.46	0.15	-0.04	-0.13
MMM	0.01	0.95	0.00	0.07	-0.81	-0.00	0.01	0.40
MRK	0.03	0.86	0.00	0.49	-0.28	-0.00	0.00	0.27
MSFT	0.14	0.81	0.17	0.11	0.00	-0.26	-1.50	0.33
NKE	0.01	0.90	0.00	0.20	-1.07	-0.44	-0.01	1.27
PG	0.05	0.83	0.06	0.59	-0.23	0.04	-0.01	-0.02
TRV	0.06	0.85	0.06	0.72	-0.14	0.02	-0.03	-0.00
UNH	0.02	0.88	0.05	0.61	-0.17	0.07	0.18	-0.01
V	0.15	0.81	0.00	0.01	-37.38	-15.97	16.19	7.29
VZ	0.06	0.83	0.09	0.31	-0.26	0.12	0.02	-0.14
WBA	0.14	0.68	0.00	0.12	-0.89	-0.08	-1.06	8.74
WMT	0.29	0.37	0.33	0.33	-0.02	0.04	0.96	-0.03
S&P 500	0.06	0.71	0.00	0.77	-0.29	0.17	0.09	0.04

Table 11: SRN-GARCH parameter estimates

Asset	α	β	γ_0	γ_1	v_{11}	v_{12}	v_{21}	v_{22}	w_1	w_2	b_h	b_z
AAPL	0.10	0.80	0.25	0.08	-3.70	5.69	0.20	0.58	-1.43	3.58	-7.19	-4.98
AMGN	0.03	0.86	0.11	2.83	-0.25	-0.21	0.13	1.47	-118.19	-5.62	0.26	-3.41
AXP	0.10	0.80	0.01	0.17	-0.22	0.31	0.79	-0.78	1.42	-0.94	-0.05	-1.35
BA	0.11	0.54	0.56	0.88	-0.06	0.49	0.35	-0.08	-0.22	19.44	-0.68	-2.34
CAT	0.25	0.63	0.00	0.03	-11.59	26.12	-5.03	13.69	0.84	15.33	-40.12	100.09
CRM	0.00	1.00	0.00	0.00	-12.30	18.84	-12.13	7.25	4.64	-8.98	-14.97	30.42
CSCO	0.00	0.91	0.01	0.13	-1.24	-0.34	-0.43	-0.53	-3.13	0.28	3.90	2.71
CVX	0.00	0.99	0.00	0.00	-1.30	-0.42	-1.45	0.52	1.42	12.90	-0.34	-81.36
DIS	0.13	0.26	0.76	0.89	-0.05	1.19	1.12	-3.53	-0.67	9.93	-1.38	5.86
GS	0.11	0.82	0.13	0.26	0.00	0.23	2.86	0.00	-0.02	-13.72	-0.46	-40.85
HD	0.15	0.79	0.11	0.02	0.00	0.36	9.81	-6.99	0.80	-5.01	-0.71	9.80
HON	0.96	0.01	0.00	0.82	-8.15	9.68	6.14	18.97	-6.98	-10.10	-1.55	-53.14
IBM	0.04	0.83	0.07	0.20	-9.75	-2.16	0.11	2.19	0.00	1.43	-0.00	-6.53
INTC	0.11	0.59	0.52	0.51	-0.04	1.57	0.83	-2.29	-1.53	2.20	-2.65	1.59
JNJ	0.06	0.59	0.12	0.43	-1.24	0.58	0.00	1.26	-0.07	-0.01	-0.33	-3.33
JPM	0.10	0.80	0.11	0.23	-0.03	0.93	0.64	-1.26	-1.59	2.51	-1.17	2.54
KO	0.00	1.00	0.02	0.04	0.08	0.29	0.73	1.84	-1.06	-11.94	-1.60	-7.71
MCD	0.38	0.09	0.17	0.04	-0.28	8.59	3.74	-9.82	0.88	-2.25	-2.27	30.88
MMM	0.04	0.95	0.02	0.56	-0.23	0.46	7.05	-23.49	0.65	-0.56	-0.24	-3.54
MRK	0.10	0.80	0.19	0.29	0.04	1.15	0.39	0.83	-2.56	1.28	-1.70	5.84
MSFT	0.13	0.82	0.15	0.00	-0.52	-0.39	5.53	-0.15	1.43	4.28	0.19	-1.45
NKE	0.06	0.79	0.00	0.67	-1.21	-1.46	0.34	1.31	3.14	-1.75	2.12	-0.94
PG	0.14	0.72	0.13	0.15	0.01	0.40	0.20	1.67	0.23	-3.39	-1.70	-4.14
TRV	0.00	1.00	0.18	0.41	-0.29	0.44	1.33	-0.12	0.67	-5.36	-1.06	13.69
UNH	0.11	0.84	0.09	0.11	-1.18	2.03	7.19	-4.30	-0.72	14.04	-2.51	7.76
V	0.00	1.00	0.12	0.00	-0.30	-0.17	0.01	4.07	1.55	-5.02	0.48	-1.66
VZ	0.09	0.80	0.11	0.86	-1.11	0.54	4.67	-0.11	-5.89	-3.46	-1.32	5.62
WBA	0.00	0.93	0.07	0.17	-1.01	-0.02	2.19	16.27	-0.97	24.78	0.15	-20.00
WMT	0.18	0.47	0.19	0.95	-0.02	0.59	8.72	-6.51	-0.55	0.15	-0.20	-1.63
S&P 500	0.17	0.26	0.15	1.18	-0.16	0.56	22.70	-0.00	-0.05	46.55	-0.14	30.22

Table 12: MGU-GARCH parameter estimates

Asset	α	β	γ_0	γ_1	v_{11}	v_{12}	v_{21}	v_{22}	v_{31}	v_{32}	v_{41}	v_{42}	w_1	w_2	w_3	w_4	b_c	b_o	b_i	b_f
AAPL	0.11	0.84	0.16	0.00	0.1	0.1	-9.96	-9.96	9.96	-9.96	9.96	-9.96	0.1	-9.96	-9.96	-9.96	0.1	-1.71	3.83	-10.09
AMGN	0.10	0.86	0.12	0.00	0.1	0.1	-1.77	-8.54	9.13	14.50	20.46	-1.21	0.1	-6.43	11.06	6.05	0.1	13.37	13.60	11.72
AXP	0.10	0.89	0.05	0.00	0.1	0.1	-10.00	-10.00	10.00	-10.00	10.00	-10.00	0.1	-10.00	-10.00	-10.00	0.1	-30.67	-1.07	-49.86
BA	0.09	0.89	0.00	0.01	0.1	0.1	-6.18	-4.39	-2.63	1.49	14.90	-6.11	0.1	-11.22	-6.80	-11.64	0.1	178.29	13.34	-6.32
CAT	0.08	0.90	0.07	0.14	0.1	0.1	-2.66	-3.31	1.81	-2.71	2.82	-2.95	0.1	-2.98	-2.66	-2.91	0.1	-26.79	-23.45	-26.78
CRM	0.02	0.93	0.05	0.09	0.1	0.1	-16.77	2.00	-92.71	-11.43	-0.02	-4.32	0.1	98.10	-36.91	-4.93	0.1	-62.26	-174.26	-6.70
CSCO	0.10	0.80	0.00	0.03	0.1	0.1	3.73	19.85	-145.94	-46.00	-45.21	2.49	0.1	4.35	-74.57	-4.82	0.1	20.82	-17.22	-32.01
CVX	0.10	0.80	0.00	0.00	0.1	0.1	5.20	-32.22	2.18	4.36	4.73	8.68	0.1	-14.39	6.82	-17.39	0.1	162.40	17.06	138.65
DIS	0.10	0.88	0.07	0.45	0.1	0.1	-7.14	-5.04	0.20	-8.99	8.36	-9.73	0.1	-6.73	8.72	-6.86	0.1	78.59	-26.44	5.31
GS	0.00	1.00	0.13	0.00	0.1	0.1	8.75	5.93	20.22	2.95	42.23	0.08	0.1	-9.64	8.42	4.71	0.1	76.10	18.72	14.29
HD	0.00	0.99	0.09	0.38	0.1	0.1	0.75	7.08	-40.53	-89.46	21.02	-2.63	0.1	-4.83	-27.54	-26.27	0.1	-4.24	-65.80	-21.13
HON	0.10	0.88	0.04	2.47	0.1	0.1	-2.34	-2.48	2.30	-2.16	2.84	-2.42	0.1	-2.38	-2.15	-2.43	0.1	-22.06	-4.56	-22.06
IBM	0.13	0.80	0.15	0.09	0.1	0.1	-0.96	-0.96	1.14	-0.96	1.14	-0.96	0.1	-0.96	-0.96	-0.96	0.1	-10.37	-10.37	-10.37
INTC	0.04	0.94	0.05	5.70	0.1	0.1	-24.50	-0.04	233.77	-49.74	0.58	-7.83	0.1	-9.72	-6.52	-8.34	0.1	139.00	-1204.01	1.87
JNJ	0.08	0.86	0.05	53.44	0.1	0.1	-16.55	-0.26	0.50	-0.00	1.25	-1.08	0.1	-0.01	-0.00	-1.08	0.1	-138.13	-0.00	-3.38
JPM	0.12	0.86	0.08	2.31	0.1	0.1	0.23	-3.06	6.76	-1.69	1.94	-1.70	0.1	0.49	-1.75	-1.69	0.1	-66.90	-1.59	-16.41
KO	0.06	0.91	0.03	0.06	0.1	0.1	-1.09	-1.09	1.27	-1.09	1.27	-1.09	0.1	-1.09	-1.09	-1.09	0.1	-11.73	-1.33	-5.70
MCD	0.06	0.89	0.05	0.09	0.1	0.1	-6.75	-5.91	5.89	10.39	-5.50	0.53	0.1	0.07	41.20	18.57	0.1	-4.91	4.91	-81.23
MMM	0.00	0.99	0.00	0.00	0.1	0.1	10.00	-10.00	-10.00	-10.00	-10.00	-10.00	0.1	-9.49	10.00	-0.14	0.1	-99.99	-99.99	-100.00
MRK	0.00	0.01	0.00	1.12	0.1	0.1	-6.30	-5.27	10.92	0.55	7.97	-11.99	0.1	-12.28	-7.16	-10.00	0.1	5.63	4.10	-3.78
MSFT	0.05	0.92	0.07	0.15	0.1	0.1	-13.69	28.32	-44.21	-41.50	-7.00	-3.40	0.1	-71.68	-8.95	-0.91	0.1	-455.39	-8.87	-93.50
NKE	0.04	0.84	0.26	0.23	0.1	0.1	-17.53	41.06	-19.73	-0.07	5.02	-5.60	0.1	-24.66	-6.89	-4.70	0.1	-392.20	-122.77	-45.72
PG	0.14	0.81	0.07	0.04	0.1	0.1	-1.12	-1.11	0.81	-1.11	1.29	-1.11	0.1	-1.12	-1.11	-1.11	0.1	-9.52	-0.93	-3.42
TRV	0.43	0.27	0.60	0.01	0.1	0.1	1.24	-2.43	-4.10	2.13	-0.66	-0.24	0.1	-2.65	3.66	-0.85	0.1	98.85	16.65	18.46
UNH	0.07	0.92	0.04	0.59	0.1	0.1	-1.58	-1.93	0.67	-1.59	1.80	-1.73	0.1	-1.78	-1.56	-1.72	0.1	-16.13	-16.11	-16.12
V	0.14	0.82	0.11	0.03	0.1	0.1	3.63	-3.75	-10.00	3.87	2.33	-1.75	0.1	-2.33	-0.99	-1.67	0.1	-12.67	0.99	-8.35
VZ	0.01	0.98	0.00	0.01	0.1	0.1	9.57	8.97	-2.98	-4.06	9.07	11.37	0.1	-6.14	6.66	-1.82	0.1	41.97	17.29	15.86
WBA	0.09	0.83	0.15	0.05	0.1	0.1	1.68	-5.58	28.69	-23.80	4.82	-8.58	0.1	-445.57	-37.42	-9.01	0.1	28.10	7.59	-52.62
WMT	0.00	0.99	0.00	3.00	0.1	0.1	-3.60	-1.89	10.00	3.23	-2.84	-10.00	0.1	-8.16	10.00	-10.00	0.1	-67.63	-99.99	-20.66
S&P 500	0.15	0.83	0.03	0.05	0.1	0.1	-1.20	-1.20	1.38	-1.20	1.37	-1.20	0.1	-1.20	-1.20	-1.20	0.1	-12.76	-1.44	-4.03

Table 13: LSTM-GARCH parameter estimates