Received: 13 September 2018 | Revised: 2 April 2019 | Accepted: 13 June 2019
DOI: 10.1002/isaf.1455

RESEARCH ARTICLE

WILEY

Exploring the predictability of range-based volatility estimators using recurrent neural networks

Gábor Petneházi[1] | József Gáll[2]

[1] Doctoral School of Mathematical and Computational Sciences, University of Debrecen, Debrecen, Hungary

[2] Department of Applied Mathematics and Probability Theory, University of Debrecen, Debrecen, Hungary

Correspondence
Gábor Petneházi, Doctoral School of Mathematical and Computational Sciences, University of Debrecen. Debrecen, Hungary
Email: gabor.petnehazi@science.unideb.hu

Funding information
EFOP-3.6.3-VEKOP-16-2017-00002; Pallas Athéné Domus Scientiae Alapítvány

Summary

We investigate the predictability of several range-based stock volatility estimates and compare them with the standard close-to-close estimate, which is most commonly acknowledged as the volatility. The patterns of volatility changes are analysed using long short-term memory recurrent neural networks, which are a state-of-the-art method of sequence learning. We implement the analysis on all current constituents of the Dow Jones Industrial Average index and report averaged evaluation results. We find that the direction of changes in the values of range-based estimates are more predictable than that of the estimate from daily closing values only.

KEYWORDS

long short-term memory, range-based volatility, recurrent neural network, volatility forecasting

1 | MOTIVATION

The volatility of assets has an important role in several areas of finance. As a measure of riskiness, it is a key factor in, for example, portfolio management and option pricing. A good understanding of the nature and evolution of return volatilities is obviously valuable for financial practitioners.

Volatility quantifies the dispersion of returns. Unfortunately, this dispersion is not directly observable. Hence, we need to estimate it, with not having a reliable benchmark.

Several studies have tried to explore and understand the nature of this unknown volatility. One reasonable approach is sampling from the price process frequently, so that we do not lose too much data. Andersen, Bollerslev, Diebold, and Ebens, (2001) analysed the properties of stock market volatility using 5 min returns. They reported that daily variances significantly fluctuate through time, and their distributions are extremely right-skewed and leptokurtic, whereas logarithmic standard deviations approximate the normal distribution well. Engle and Patton, (2007) outlined several stylized facts of volatility that have emerged in previous studies:

- Persistence: large moves are usually followed by large moves, and small moves are usually followed by small moves in the price process.
- Mean reversion: usually, there is a normal level of volatility to which it returns after uplifts and falls.
- Asymmetric impact of innovations: positive and negative shocks have different impacts on volatility.
- Influence of exogenous variables: information outside the price series (e.g. announcements) could have an impact on volatility.

These features suggest that, if we could measure volatility, it should be somewhat forecastable. But we cannot measure it; the best we can do is come up with reasonable proxies.

One such proxy is the standard deviation of returns—returns, which are usually calculated from daily closing prices. It is obvious that by sampling the asset's price more frequently we could make better estimates of its unobservable true volatility. If, for example, we measured the daily price ranges (i.e. daily high minus daily low), we would already know a lot more about the unseen path of the prices.

Unlike high-frequency (say, minutely) data, daily open, high, low, and close values are freely available. Finding good estimators that use these four daily values only is therefore a challenging and important task.

In this paper, we are going to compare various range-based volatility estimators according to their predictability. We argue that those estimators whose changes are easier to predict can be more useful in practice. Forecasts can move historical volatilities a bit forward into the future, and knowing something about the future is valuable.

2 | VOLATILITY ESTIMATORS

It is usually assumed that stock prices follow a geometric Brownian motion (GBM). GBM satisfies $dS_t = \mu S_t\,dt + \sigma S_t\,dW_t$, where W_t is a Brownian motion or Wiener process, μ is the drift, and σ is the volatility.

The estimators that we investigate make estimates of the σ parameter while keeping the geometric Brownian assumption with slight differences (e.g. regarding the drift term). σ can change from one day to another, but it is assumed to remain unchanged during one particular day (Molnár, 2012).

Volatility is most often calculated simply as the standard deviation of returns:

$$\sigma = \sqrt{F}\sqrt{\frac{\sum_{t=1}^{N}\left[\ln\left(\frac{C_t}{C_{t-1}}\right) - \overline{\ln\left(\frac{C_t}{C_{t-1}}\right)}\right]^2}{N-1}} \tag{1}$$

where C_t is the closing price of day t, and N is the number of days used in the calculation. As volatility should measure the dispersion of the prices, standard deviation is a very reasonable proxy. However, when returns are calculated on a daily basis (as the difference of log closing prices), this simple and intuitive formula ignores all intraday price movements, which is a great loss of information.

The so-called range-based volatility estimators use daily open, high, low, and close values to make volatility estimates. Several such formulas have been proposed in the history of volatility estimation. Here, we are going to present some of the better known range-based volatility formulas. O_t, H_t, L_t, and C_t respectively stand for the open, high, low, and close price at time t. N, again, is the size of the time window in days for calculating the volatilities, and F is just for scaling the results to another time unit.

Parkinson, (1980) proposed the extreme-value method for variance estimation:

$$\sigma_P = \sqrt{F}\sqrt{\frac{\frac{1}{4\ln(2)}\sum_{t=1}^{N}\left[\ln\left(\frac{H_t}{L_t}\right)\right]^2}{N}} \tag{2}$$

He showed that by using high and low prices we may get an estimate that is far superior to the standard close-to-close formula.

Garman and Klass, (1980) published estimators using open, high, low, and close values:

$$\sigma_{GK} = \sqrt{F}\sqrt{\frac{\sum_{t=1}^{N} 0.5\left[\ln\left(\frac{H_t}{L_t}\right)\right]^2 - (2\ln - 1)\left[\ln\left(\frac{C_t}{O_t}\right)\right]^2}{N}} \tag{3}$$

Their results demonstrate much higher efficiency factors than that of the close-to-close estimator.

The Parkinson and Garman–Klass volatility estimators assume the asset prices follow a continuous Brownian motion with no drift. Rogers and Satchell, (1991) proposed a formula that allows for drifts:

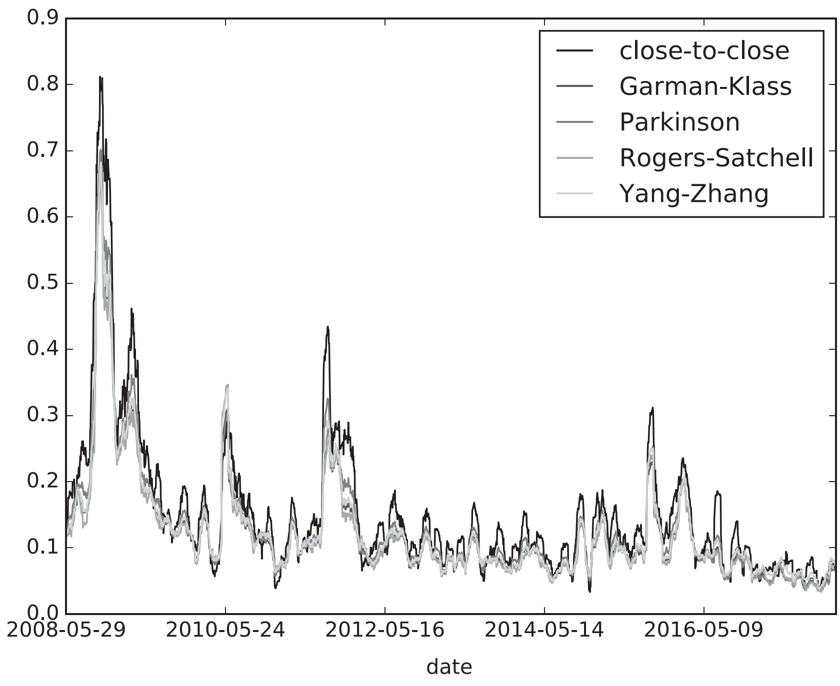

FIGURE 1 Volatility estimates for the Dow Jones industrial average in the observed period

$$\sigma_{\mathrm{RS}} = \sqrt{F}\sqrt{\frac{\sum_{t=1}^{N} \ln\left(\frac{H_t}{O_t}\right)\left[\ln\left(\frac{H_t}{O_t}\right) - \ln\left(\frac{C_t}{O_t}\right)\right] + \ln\left(\frac{L_t}{O_t}\right)\left[\ln\left(\frac{L_t}{O_t}\right) - \ln\left(\frac{C_t}{O_t}\right)\right]}{N}} \tag{4}$$

Rogers, Satchell, and Yoon, (1994) investigated the efficiency of volatility estimators through simulation and found that the Rogers–Satchell method is superior to the Garman–Klass if there is a time-varying drift in the data. However, when there is no drift, Garman–Klass outperforms Rogers–Satchell, so the former should be preferred when the expected returns are less volatile.

Yang and Zhang, (2000) published a formula that is unbiased, drift independent, and consistent in dealing with opening jumps; this latter feature is unique among the formulas examined:

$$\sigma_{\mathrm{YZ}} = \sqrt{F}\sqrt{\frac{\sum_{t=1}^{N}\left[\ln\left(\frac{O_t}{C_{t-1}}\right) - \overline{\ln\left(\frac{O_t}{C_{t-1}}\right)}\right]^2}{N-1} + \frac{k\sum_{t=1}^{N}\left[\ln\left(\frac{C_t}{O_t}\right) - \overline{\ln\left(\frac{C_t}{O_t}\right)}\right]^2}{N-1} + (1-k)V_{\mathrm{RS}}} \tag{5}$$

$$k = \frac{0.34}{1.34 + \frac{N+1}{N-1}}$$

$$V_{\mathrm{RS}} = \frac{\sum_{t=1}^{N} \ln\left(\frac{H_t}{O_t}\right)\left[\ln\left(\frac{H_t}{O_t}\right) - \ln\left(\frac{C_t}{O_t}\right)\right] + \ln\left(\frac{L_t}{O_t}\right)\left[\ln\left(\frac{L_t}{O_t}\right) - \ln\left(\frac{C_t}{O_t}\right)\right]}{N}$$

Range-based volatility estimation has quite a long history and evolution. Here, we have only mentioned the formulas that we are going to use in this work. Chou, Chou, and Liu, (2010) give a detailed review of the development of range-based volatility estimators (Figure 1).

All of these range-based estimators assume that the asset price follows a continuous GBM. This is a strict assumption. Shu and Zhang, (2006) analysed all four range-based estimators that we investigate in an attempt to measure the degree to which they can be useful in real markets that deviate from the GBM. They found that estimates from range-based models are quite close to integrated variances computed from intraday returns with much higher computational requirements. Using simulations, Shu and Zhang also confirmed the expectation that, when there is just a small drift and no opening jumps in the prices, all four estimators provide good estimates. When the drift is large, the Parkinson and Garman–Klass estimators overestimate the true variances, whereas the other two behave properly. Large opening jumps can only be handled by the Yang–Zhang estimator; all the other formulas give downward-biased estimates in the case of opening jumps.

3 | VOLATILITY FORECASTING

Although forecasting changes in stock returns is a very hard task, forecasting the size of changes (i.e. the volatility) seems more promising. It also has a high importance in financial practice, and it has already been the subject of many studies.

Poon and Granger, (2003) gave a detailed review on 93 papers that studied the forecasting power of volatility models, and Poon and Granger, (2005) provided a summary of the findings. They surveyed historical volatility, autoregressive conditionally heteroscedasticity (ARCH), stochastic volatility, and option-implied volatility models. Standard volatility forecasting models might be improved by incorporating price range data; for example, Molnár, (2016) proposed range generalized ARCH (GARCH) models and found that range-GARCH(1, 1) performs significantly better than the standard GARCH(1, 1).

For modelling the changes in volatility, artificial neural networks (ANNs) also seem a natural choice. Neural networks, even with a single hidden layer, are universal approximators (Hornik, 1991), and it makes them a strong competitor of traditional learning algorithms and time-series methods.

Malliaris and Salchenberger, (1996) forecasted implied volatilities using neural networks trained on past volatilities and other options market factors. Donaldson and Kamstra, (1996) used ANNs to combine different time-series forecasts of stock market volatilities, concluding that combining forecasts using ANNs generally outperforms traditional combining methods due to its flexibility. Roh, (2007) proposed new hybrid models combining neural networks and time-series models for improving volatility predictions in terms of deviation and direction accuracy.

Recurrent neural networks (RNNs) have also been applied to volatility forecasting. Xiong, Nichols, and Shen, (2015) used long short-term memory (LSTM) networks on Google Domestic Trends data to forecast S&P 500 volatilities. Their article is similar to ours, since we, too, apply LSTM RNNs to forecast range-based volatility. However, they used external data, whereas we only use historical stock prices, and they made predictions for the daily values of the volatility estimates, whereas we primarily aim for predicting the directions of daily changes.

4 | RECURRENT NEURAL NETWORKS

RNNs are neural networks for sequential data: instead of relying on a single data point, RNNs take into account a whole sequence. An RNN uses the most recent observation together with the past to make a good decision:

$$h_t = \tanh(W_h x_t + U_h h_{t-1} + b_h) \tag{6}$$

This trait makes them a reasonable choice for modelling the behaviour of time series. Yet, plain RNN models (e.g. Elman, 1990) suffer from the vanishing gradient problem and are hard to train (Hochreiter, Bengio, Frasconi, & Schmidhuber, 2001). They are unable to model long-term dependencies in the data. Luckily, there are some more advanced RNN architectures that solve this problem at the expense of some model complexity. One such architecture is LSTM.

LSTM was invented by Hochreiter and Schmidhuber, (1997). An LSTM cell gives memory to the RNN, and the ability to read, write, and forget data. The cell uses separate gating units to operate these memory management abilities:

$$i_t = \mathrm{sigmoid}(W_i x_t + U_i h_{t-1} + b_i) \tag{7}$$

$$f_t = \mathrm{sigmoid}(W_f x_t + U_f h_{t-1} + b_f) \tag{8}$$

$$o_t = \mathrm{sigmoid}(W_o x_t + U_o h_{t-1} + b_o) \tag{9}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh(W_c x_t + U_c h_{t-1} + b_c) \tag{10}$$

$$h_t = o_t \odot \tanh(c_t) \tag{11}$$

$x_t \in \mathbb{R}^n$ is the input to the LSTM cell. $h_t \in \mathbb{R}^h$ denotes the output from the LSTM, which is usually called the hidden state (equation (11)). $c_t \in \mathbb{R}^h$ is the so-called cell state (equation (10)), which represents the memory. $i_t \in \mathbb{R}^h$, $f_t \in \mathbb{R}^h$, and $o_t \in \mathbb{R}^h$ are the input, forget, and output gates. The input gate (equation (7)) calculates what to keep in memory, the forget gate (equation (8)) calculates what to remember, and the output gate (equation (9)) calculates which part of the memory to use immediately. They do these things by applying some simple mathematical operations on the input data, the previous hidden state, and the corresponding learnable weights $W \in \mathbb{R}^{h \times n}$ and $U \in \mathbb{R}^{h \times h}$, which can easily be optimized using backpropagation through time (e.g. Greff, Srivastava, Koutník, Steunebrink, & Schmidhuber, 2017).

The LSTM formulas are rather formidable at first sight, but they form a system that is fairly intuitive, and which works, in many cases, amazingly well.

Some recent applications of RNNs to time-series forecasting include, for example, Che, Purushotham, Cho, Sontag, and Liu, (2016), cinar et al. (2017a), (2017b), Hsu, (2017), and Laptev, Yosinski, Li, and Smyl, (2017). LSTM networks are also used in some financial studies for modelling time-series data, like historical volatility. This study aims to contribute to this research area by comparing the predictability of range-based volatility estimates using LSTM networks.

5 | DATA

Our dataset was obtained from Yahoo Finance (https://finance.yahoo.com/). We have downloaded 10 years (from 1 January 2008 to 31 December 2017) of daily open, high, low, and close values for all current constituents of the Dow Jones Industrial Average index. (One out of the 30 stocks is missing the first few months' data since its initial public offering took place in March 2008.)

We used the previously presented formulas for quantifying volatility; namely, the close-to-close, the Garman–Klass, the Parkinson, the Rogers–Satchell, and the Yang–Zhang estimators. All volatility estimates were calculated using a window of 21 days.

We used very few data for training our neural networks. The exact values of the estimates have been dropped, and we only kept a single binary variable indicating the direction of daily changes: 1 for an upward movement and 0 everywhere else. After this simple binary experiment, we tried forecasting the values of volatility estimates as well.

We used the first 70% of the available data for training the LSTM models, and we made 1-day-ahead forecasts on the remaining 30%, which is roughly the last 3 years.

TABLE 1 Evaluation metrics for 1-day-ahead direction-of-change forecasts

	Accuracy		Precision		Recall		F_1	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
Close-to-close	0.51	0.02	0.52	0.07	0.31	0.25	0.33	0.17
Garman–Klass	**0.57**	0.03	**0.63**	0.05	0.30	0.10	0.40	0.10
Parkinson	**0.57**	0.02	**0.63**	0.04	**0.32**	0.09	**0.42**	0.08
Rogers–Satchell	0.55	0.03	0.61	0.04	0.26	0.10	0.35	0.10
Yang–Zhang	**0.57**	0.02	**0.63**	0.05	0.29	0.09	0.39	0.08

SD: standard deviation.

Bold entries indicate best mean value of the particular metric.

TABLE 2 Evaluation metrics for predictions with 0.45 probability threshold

	Accuracy		Precision		Recall		F_1	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
Close-to-close	0.50	0.02	0.50	0.01	**0.88**	0.16	**0.63**	0.06
Garman–Klass	**0.58**	0.03	0.59	0.03	0.49	0.09	0.53	0.06
Parkinson	**0.58**	0.02	**0.60**	0.03	0.46	0.09	0.51	0.06
Rogers–Satchell	0.57	0.02	0.58	0.03	0.45	0.10	0.50	0.07
Yang–Zhang	**0.58**	0.02	0.59	0.03	0.47	0.10	0.52	0.07

SD: standard deviation.

Bold entries indicate best mean value of the particular metric.

TABLE 3 Evaluation metrics for confident predictions (P > 0.5 or P < 0.4)

	Accuracy		Precision		Recall		F_1	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
Close-to-close	0.51	0.11	0.51	0.11	**0.93**	0.20	**0.66**	0.13
Garman–Klass	0.61	0.03	**0.63**	0.04	0.50	0.11	0.55	0.07
Parkinson	0.61	0.03	**0.63**	0.05	0.45	0.11	0.51	0.08
Rogers–Satchell	0.59	0.03	0.62	0.05	0.46	0.15	0.52	0.12
Yang–Zhang	**0.62**	0.03	**0.63**	0.04	0.48	0.12	0.54	0.09

SD: standard deviation.

Bold entries indicate best mean value of the particular metric.

6 | NEURAL NETWORK ARCHITECTURE

Our LSTM RNN was built in Keras (Chollet, 2015) with the TensorFlow (Abadi et al., 2015; tensorflow.org) backend.

We used a two-layer RNN with 10 hidden units in each layer. We chose the Adam optimizer (Kingma and Ba, 2014) to minimize the loss function; in this case, binary cross entropy. A dropout (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014) of 0.3 was applied on the nonrecurrent connections. The learning rate was set to 0.001.

The series of volatility directions were unrolled for 10 days, and the unrolled subsequences were fed to the algorithm in batches of 32. The experiments ran for 300 epochs.

No thorough hyperparameter optimization was conducted. We only aimed to find a reasonable setting that is appropriate for comparing the predictability of our volatility formulas examined.

7 | RESULTS: BASIC PROBLEM

We used roughly the last 3 years of our 10-year dataset for out-of-sample validation. Four evaluation metrics are reported: accuracy, precision, recall, and F_1 score. Those metrics were averaged over all 30 constituents of the Dow Jones Industrial Average stock market index. For each stock, we trained an individual RNN model.

Table 1 displays our results for this experiment. Whereas the close-to-close estimator's accuracy was barely higher than 50%, none of the range-based volatilities' averaged accuracies was below 55%. It seems to be a considerable difference.

Despite the promising accuracies, the F_1 score, being the harmonic mean of precision and recall, was consistently below 0.5 in each case.

Precision is the fraction of our predicted upward movements that really were increases in the volatility. Recall is the fraction of real upward movements that we have predicted to be so. Though each estimator's precision was higher than the overall accuracy, the recall was very poor. This means that our algorithm struggled in finding the upward movements. It simply follows from the fact that the RNN has chosen to predict downward movements in a higher proportion. Since the ratio of upward movements was usually close to 0.5, this behaviour of the algorithm does not invalidate its prediction ability. Yet, it is obviously not a preferred property, especially when we assume that identifying a rise in the volatility is more valuable than identifying a drop.

To solve the issue of low recall, we tried lowering the threshold of making an upward guess from the default of 0.5.

Table 2 displays our results using a threshold of 0.45. In this case, we force the RNN to predict more increases. The recall values increased, leading to F_1 scores >0.5, while not sacrificing the overall

TABLE 4 Average proportions of confidently predicted directions

Close-to-close	Garman–Klass	Parkinson	Rogers–Satchell	Yang–Zhang
0.28	0.66	0.67	0.54	0.64

TABLE 5 Correlations of different estimates of Dow Jones industrial average volatility

	σ	σ_{GK}	σ_P	σ_{RS}	σ_{YZ}
σ	1.000	0.973	0.988	0.956	0.973
σ_{GK}	0.973	1.000	0.996	0.997	0.999
σ_P	0.988	0.996	1.000	0.988	0.995
σ_{RS}	0.956	0.997	0.988	1.000	0.997
σ_{YZ}	0.973	0.999	0.995	0.997	1.000

TABLE 6 Evaluation metrics for value forecasts of the volatility estimates

	RMSE		SMAPE	
	Mean	SD	Mean	SD
Close-to-close	0.0185	0.0047	7.03	1.66
Garman–Klass	0.0141	0.0055	5.14	1.76
Parkinson	**0.0131**	0.0043	**5.01**	1.27
Rogers–Satchell	0.0155	0.0070	5.07	1.84
Yang–Zhang	0.0194	0.0046	5.28	0.99

RMSE: root mean-squared-error; SD: standard deviation; SMAPE: symmetric mean absolute percentage error.

Bold entries indicate best mean value of the particular metric.

accuracy. In fact, the accuracy increased a bit for all range-based estimators.

Finally, we freed the algorithm from having to make predictions at all times. It may be preferable to let the algorithm decide if it has the necessary confidence to make a prediction. We chose to flag prediction probabilities between 0.4 and 0.5 as unconfident and only kept the days with estimated probabilities outside this range. (All those thresholds were chosen arbitrarily.)

Table 3 presents the evaluation metrics for the confident predictions. By dropping (quite) some uncertain predictions, we have exceeded 60% accuracy with three out of four range-based estimators.

Table 4 displays the proportions of predictions that remained after excluding the uncertain ones. It seems that estimators with lower accuracies have more predictions close to the 0.45 binary decision threshold. Hence, weak forecasts make less guesses, which is preferable.

In this experiment, all four range-based estimators clearly outperformed the benchmark close-to-close estimator in terms of predictability. The range-based estimators generated similar results, though the Rogers–Satchell estimator performed slightly worse than the other three.

All of those volatility estimators move closely together, as expected, having above 0.95 correlations for the Dow Jones Industrial Average index (Table 5). We could observe similarly high correlation in case of the individual components as well. It is therefore quite remarkable that, whereas the close-to-close estimator seems essentially unpredictable, the directional changes of range-based estimators were so easily detected from so few data to near 60% accuracy that it calls for further research.

8 | RESULTS: ENHANCED PROBLEM

Our results so far suggest that range-based volatility estimators have a forecastability advantage over the close-to-close estimator. It is worth though exploring several further aspects.

We have only explored the changes of volatility, whereas one naturally wants to know the exact values too. Hence, we shall consider the regression problem as well.

Also, we used the volatility estimators as they appeared in the original articles, though some small modifications might make sense.

Some important corporate events (e.g. stock splits or dividends) might happen after-hours. The closing values should somehow reflect those events, so they should be adjusted. These adjustments need not be handcrafted; they are reported in publicly available financial data sources. Hence, it might be more appropriate to replace the close values by the adjusted close values in the volatility formulas.

Not all estimators estimate the exact same thing. Equations (1) and (5) estimate volatility over the whole day, whereas equations (2)–(4) estimate the volatility during the trading period only. For the sake of comparability, those latter volatility estimators should be modified (Molnár, 2012).

Here, we report the results of these modified experiments. Table 6 displays evaluation metrics for the forecasts of the exact values of the volatility estimates. Tables 7 and 8 show the results of the experiments with modified estimators for the direction and volatility forecasts respectively.

We used two metrics to evaluate the volatility value forecasts: root mean-squared-error, which was the loss function of our neural network for this problem; and symmetric mean absolute percentage error, which relates the absolute value of forecast errors to that of the actual and predicted values. It is a relative error measure expressed as a percentage ranging from 0 to 200.

The differences in forecastability are much less clear now. The value forecasts of the original estimators do not show any clear

TABLE 8 Evaluation metrics for value forecasts of the modified volatility estimates

	RMSE		SMAPE	
	Mean	SD	Mean	SD
Close-to-close	**0.0183**	0.0044	**6.80**	0.99
Garman–Klass	0.2786	0.1009	44.18	6.88
Parkinson	0.4190	0.1722	50.85	8.80
Rogers–Satchell	0.3815	0.1511	48.06	9.38
Yang–Zhang	0.0276	0.0070	11.88	3.64

RMSE: root mean-squared-error; SD: standard deviation; SMAPE: symmetric mean absolute percentage error.

Bold entries indicate best mean value of the particular metric.

TABLE 7 Evaluation metrics for one-day-ahead direction-of-change forecasts of the modified volatility estimates (0.45 threshold)

	Accuracy		Precision		Recall		F_1	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
Close-to-close	0.51	0.02	**0.50**	0.01	**0.83**	0.09	**0.63**	0.02
Garman–Klass	0.51	0.04	0.45	0.03	0.55	0.21	0.48	0.10
Parkinson	**0.56**	0.06	0.44	0.04	0.42	0.20	0.41	0.11
Rogers–Satchell	0.54	0.04	0.45	0.03	0.47	0.16	0.45	0.08
Yang–Zhang	0.50	0.02	**0.50**	0.02	0.82	0.11	0.62	0.04

SD: standard deviation.

Bold entries indicate best mean value of the particular metric.

differences. The values of those estimators that had to be modified in order to estimate 24 h volatility seem unpredictable, which is quite peculiar. The directional forecastability of the modified range-based volatility estimators is closer to that of the close-to-close estimator: now only two estimators could keep the lead.

9 | CONCLUSIONS AND FURTHER RESEARCH DIRECTIONS

We can conclude that the daily directions of changes in range-based volatility estimates can be forecasted to some degree, using LSTM RNNs and very little data—only historical patterns of up and down movements.

There is not much difference in the predictability of the range-based volatility estimators; however, they all seem to be easier to forecast than the baseline close-to-close estimator, which is most commonly used and acknowledged as financial volatility.

These differences were not present, though, when we were trying to forecast the values of volatility. Also, some reasonable changes to the original volatility estimator formulas led to weaker forecasting performance.

Our results are not highly optimized, and they stand for comparison purposes. We could have achieved higher accuracies, but we aimed to use small input data and simple model architectures. Future research might explore the degree of accuracy to which these volatility estimates can be forecasted.

We only used daily price range data and made predictions a day ahead. This is disadvantageous in two ways: we had to use a relatively small dataset for training our neural networks, and we had to make forecasts for relatively long periods. It would be worth exploring the forecastability of realized volatilities computed from intraday data.

We built small models from small data. It should be explored whether a general volatility forecasting network could learn from the price series history of different assets. Sirignano and Cont, (2018) found universal features of price formation that suggest it was possible.

ACKNOWLEDGEMENTS

This work was supported by the construction EFOP-3.6.3-VEKOP-16-2017-00002. The project was supported by the European Union, co-financed by the European Social Fund. This research was partially supported by the Pallas Athene Domus Educationis Foundation. The views expressed are those of the authors and do not necessarily reflect the official opinion of the Pallas Athene Domus Educationis Foundation.

ORCID

Gábor Petneházi https://orcid.org/0000-0001-6579-4583

REFERENCES

Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, Corrado G. S, Davis A, Dean J, Devin M, Ghemawat S, Goodfellow I, Harp A, Irving G, Isard M, Jia Y, Jozefowicz R, Kaiser L, Kudlur M, Levenberg J, Mané D, Monga R, Moore S, Murray D, Olah C, Schuster M, Shlens J, Steiner B, Sutskever I, Talwar K, Tucker P, Vanhoucke V, Vasudevan V, Viégas F, Vinyals O, Warden P, Wattenberg M, Wicke M, Yu Y, Zheng X. (2015). TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. arXiv:1603.04467.

Andersen, T. G., Bollerslev, T., Diebold, F. X., & Ebens, H. (2001). The distribution of realized stock return volatility. *Journal of Financial Economics*, *61*(1), 43–76.

Che Z, Purushotham S, Cho K, Sontag D, Liu Y. 2016. Recurrent neural networks for multivariate time series with missing values. arXiv:1606.01865.

Chollet F. 2015. Keras. https://github.com/keras-team/keras.

Chou, R. Y., Chou, H., & Liu, N. (2010). Range volatility models and their applications in finance. In C.-F. Lee, A. C. Lee, & J. Lee (Eds.), *Handbook of Quantitative Finance and Risk Management* (pp. 1273–1281). New York, NY: Springer.

Cinar, Y. G., Mirisaee, H., Goswami, P., Gaussier, E., Aït-Bachir, A., & Strijov, V. (2017a). Position-based content attention for time series forecasting with sequence-to-sequence RNNs. In D. Liu, S. Xie, Y. Li, D. Zhao, & E. S. El-Alfy (Eds.), *Neural Information Processing. ICONIP 2017*. *Lecture Notes in Computer Science*. (pp. 533–544). Cham, Switzerland: Springer.

Cinar YG, Mirisaee H, Goswami P, Gaussier E, Aït-Bachir A, Strijov V. 2017b. Time series forecasting using RNNs: an extended attention mechanism to model periods and handle missing values. arXiv:1703.10089.

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

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, *14*(2), 179–211.

Engle, R. F., & Patton, A. J. (2007). In J. Knight, & S. Satchell (Eds.), *What good is a volatility model? In Forecasting Volatility in the Financial Markets* (3rd ed.) (pp. 47–63). Oxford, UK: Elsevier.

Garman, M. B., & Klass, M. J. (1980). On the estimation of security price volatilities from historical data. *Journal of Business*, *53*(1), 67–78.

Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., & Schmidhuber, J. (2017). LSTM: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, *28*(10), 2222–2232.

Hochreiter, S., Bengio, Y., Frasconi, P., & Schmidhuber, J. (2001). Gradient flow in recurrent nets: the difficulty of learning long-term dependencies. In S. C. Kremer, & J. F. Kolen (Eds.), *A Field Guide to Dynamical Recurrent Neural Networks* (pp. 237–244). Piscataway, NJ: IEEE Press.

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

Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks*, *4*(2), 251–257.

Hsu D. 2017. Time series forecasting based on augmented long short-term memory. arXiv:1707.00666.

Kingma DP Ba J. 2014. Adam: A Method for Stochastic Optimization. arXiv:1412.6980.

Laptev N, Yosinski J, Li LE, Smyl S. 2017. Time-series extreme event forecasting with neural networks at Uber. Presented at the ICML 2017 Time Series Workshop.

Malliaris, M., & Salchenberger, L. (1996). Using neural networks to forecast the S&P 100 implied volatility. *Neurocomputing*, *10*(2), 183–195.

Molnár, P. (2012). Properties of range-based volatility estimators. *International Review of Financial Analysis*, *23*, 20–29.

Molnár, P. (2016). High–low range in GARCH models of stock return volatility. *Applied Economics*, *48*(51), 4977–4991.

Parkinson, M. (1980). The extreme value method for estimating the variance of the rate of return. *Journal of Business*, *53*(1), 61–65.

Poon, S.-H., & Granger, C. (2005). Practical issues in forecasting volatility. *Financial Analysts Journal*, *61*(1), 45–56.

Poon, S.-H., & Granger, C. W. (2003). Forecasting volatility in financial markets: A review. *Journal of Economic Literature*, *41*(2), 478–539.

Rogers, L. C. G., & Satchell, S. E. (1991). Estimating variance from high, low and closing prices. *The Annals of Applied Probability*, *1*(4), 504–512.

Rogers, L. C. G., Satchell, S. E., & Yoon, Y. (1994). Estimating the volatility of stock prices: A comparison of methods that use high and low prices. *Applied Financial Economics*, *4*(3), 241–247.

Roh, T. H. (2007). Forecasting the volatility of stock price index. *Expert Systems with Applications*, *33*(4), 916–922.

Shu, J., & Zhang, J. E. (2006). Testing range estimators of historical volatility. *Journal of Futures Markets*, *26*(3), 297–313.

Sirignano J, Cont R. 2018. Universal features of price formation in financial markets: perspectives from deep learning. arXiv:1803.06917.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, *15*(1), 1929–1958.

Xiong R, Nichols EP, Shen Y. 2015. Deep learning stock volatility with Google Domestic Trends. arXiv:1512.04916.

Yang, D., & Zhang, Q. (2000). Drift-independent volatility estimation based on high, low, open, and close prices. *The Journal of Business*, *73*(3), 477–492.

How to cite this article: Petneházi G, Gáll J. Exploring the predictability of range-based volatility estimators using recurrent neural networks. *Intell Sys Acc Fin Mgmt*. 2019;1–8. https://doi.org/10.1002/isaf.1455