ECONOMETRICS
EKONOMETRIA
Advances in Applied Data Analysis

Econometrics. Ekonometria.
Advances in Applied Data Analysis
Year XXXX, Vol. XX, No. X
ISSN 2449-9994
journals.ue.wroc.pl/eada

Enhancing Forecast Accuracy: The Impact of Data Transformation in Time Series Models

Autor's First Name Author's Last Name ## Fill in after the reviews (use the *Author* style)

Affiliation: ## Fill in after the reviews **(use the *Metadata* style)**

E-mail: ## Fill in after the reviews

ORCID: ## Fill in after the reviews

Quote as: ## Do not change

DOI ## Do not change

JEL: 1-3 C22, C53, C18

Abstract (use the *Abstract* style)

Aim: To formulate suggestion which preprocessing method is preferable for various forecasting method including machine learning approaches particularly for stock values forecasting

Methodology: Research study on actual stock values prediction on example on 10 average NYSE enterprises comparing five scenarios of data preparation

Results: The confirmation of theoretical assumptions and recommendations for proper design of benchmark studies and real forecasting models

Implications and recommendations: As stated in literature of subject data transformation for models based on stochastic process like ARIMA and GARCH the transformation to return ratios (as form of differentiation) is desired transformation. For machine learning approaches, especially recurrent neural networks Long Short-Term Memory Network and Gated Recurrent Unit the min max scaler data transformation. For exponential Smoothing and Brownian motion methods best result were achieved for no-transformed (raw) data. The suggestion for researchers preparing benchmark studies and real forecasting models is stated in final section of paper. The main goal of the article is to emphasize,on the example of stock values forecasting, that proper benchmark studies and real-life applications should be designed in the way that proper preprocessing stage should be used for given model and using the same preprocessing for different models may sometimes give the misleading results.

Originality/value: The topic of data preparation and transformation although commonly present in literature of subject in literature rarely is confirmed by research studies on real dataset. According to the author's knowledge, this type of analysis has not been carried out on real data so far .

Keywords: Forecasting, Stochastic process models, recurrent neural networks.

1. Introduction

Data preprocessing plays a crucial role in the accuracy and reliability of forecasting tasks, especially when using deep learning models. One of the essential steps is data transformation, which ensures that the input time series is appropriately prepared for modeling. Direct comparisons of forecasting methods using identical raw datasets can lead to misleading conclusions, as each model type has different requirements for input structure. Thus, comparing model outputs without accounting for proper reverse transformation of the data can distort benchmark results. In particular, raw data is typically unsuitable for deep learning models without preprocessing, as these models require normalized or integrated inputs to learn effectively.
Statistical forecasting models such as ARIMA and GARCH, on the other hand, inherently assume certain properties like stationarity, and thus necessitate integration in cases of non-stationarity—an almost ubiquitous condition in real-world time series like stock market prices or energy consumption. Consequently, the need for an appropriate transformation of input data is not only critical but also method-dependent. Therefore, the first and foundational step in any meaningful benchmark analysis should be the tailored preprocessing of data aligned with the requirements of the respective modeling approach.
It the paper five preprocessing strategies for time series forecasting: raw data, max-min scaled data, integrated data (using return ratios), integrated and scaled to percentage form, and finally, integrated followed by max-min scaling will be compared on real stock data time datasets.

2. Literature Review

When modeling stock price time series with ARIMA or GARCH family models, the use of raw prices is generally discouraged due to their strong nonstationarity and stochastic trends. Financial time series often exhibit unit roots, meaning that their statistical properties, such as mean and variance, change over time, violating the stationarity assumption required by ARIMA and GARCH processes (Box et al., 2016; Tsay, 2010). To address this issue, returns—typically defined as logarithmic differences of prices—are commonly employed. Returns tend to stabilize the variance, reduce the presence of autocorrelation in levels, and transform the series into a stationary process more suitable for linear and nonlinear time-series modeling (Hamilton, 1994). This transformation is critical, as ignoring nonstationarity may lead to spurious regression results and misleading forecasts (Enders, 2015).

Moreover, the differentiation between price levels and returns is especially important when volatility forecasting is the primary objective, as in GARCH-type models. GARCH models are designed to capture volatility clustering and conditional heteroskedasticity, features that are prominent in return series but much less visible in raw prices (Engle, 1982; Bollerslev, 1986). Using return ratios or log-differenced series not only ensures stationarity but also amplifies the presence of volatility patterns necessary for effective estimation of conditional variance dynamics. Therefore, transformations such as first differencing or return calculation are indispensable preprocessing steps for both ARIMA-based forecasts of mean dynamics and GARCH-family models focused on volatility structures in financial time series (Francq & Zakoïan, 2019).

On the other hand, recurrent neural networks such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) models have become popular tools for forecasting stock time series due to their ability to capture nonlinear temporal dependencies. However, financial data, particularly stock prices and returns, typically exhibit large numerical ranges, volatility clustering, and heavy-tailed distributions, which can impede neural network training if left unscaled. Neural architectures are highly sensitive to input magnitude, and unscaled data often lead to exploding or vanishing gradients, slower convergence, and suboptimal weight updates (Goodfellow et al., 2016; Chollet & Allaire, 2018). Scaling techniques such as min–max normalization or z-score standardization improve the stability of gradient-based optimization, ensure numerical homogeneity across features, and enhance the model's ability

to learn temporal dependencies (Brownlee, 2019). Thus, data preprocessing through scaling is not only a best practice but a necessary condition for reliable training of LSTM and GRU models on stock time series.

Beyond numerical stability, scaling also improves predictive accuracy in financial forecasting tasks. Since neural networks internally rely on activation functions that operate efficiently within limited value ranges, scaled inputs help preserve dynamic patterns without distortion, allowing the models to better capture subtle variations in stock returns (Zhang et al., 2020). In empirical studies, normalized datasets consistently yield superior forecasting performance compared to raw data, particularly in volatile financial environments where large fluctuations can dominate unscaled sequences (Fischer & Krauss, 2018). Therefore, scaling should be regarded as a critical preprocessing step that directly affects the convergence, generalization ability, and forecasting accuracy of deep recurrent models in stock market applications.

3. Methodology

To confirm the recommendations from literature of the subject the following research simulation study has based on real stock time data series has been conducted:

Ten listed on the stock exchange enterprise have been selected due to following criteria:

1. **Market Cap Range:** These companies are large (typically $20B–$100B) but not among the top 50 mega-caps (e.g., Apple, Microsoft).
2. **Diverse Sectors:** Includes industrials, materials, finance, and real estate to avoid overconcentration.
3. **Established Presence:** All are well-known, stable firms with strong NYSE trading volumes.

Companies actually selected for research study are described in table 1:

Table 1. Enterprises, which stock values have been used in the research study.

Ticker	Company Name	Description
DHR	Danaher Corporation	A global science and technology innovator specializing in healthcare, environmental, and industrial solutions.
PLD	Prologis, Inc.	A leading real estate investment trust (REIT) focused on logistics and warehouse properties worldwide.
SHW	Sherwin-Williams Company	One of the largest producers of paints, coatings, and related products globally.
APD	Air Products and Chemicals, Inc.	A multinational industrial gases and chemicals company serving various industries.
SYY	Sysco Corporation	The world's largest food distribution company, serving restaurants, healthcare, and educational facilities.
AFL	Aflac Incorporated	A Fortune 500 insurer known for supplemental health and life insurance policies.
EFX	Equifax Inc.	A major consumer credit reporting agency providing data analytics and risk assessment services.
VMC	Vulcan Materials Company	The largest U.S. producer of construction aggregates (crushed stone, sand, and gravel).
EMR	Emerson Electric Co.	A diversified technology and engineering company specializing in automation and industrial solutions.
HWM	Howmet Aerospace Inc.	A manufacturer of engineered metal products for aerospace, defense, and transportation industries.

Source: own research based on Yahoo Finance repository

Fiva data preparation strategies have been compared:

- Raw data

- Max min scaled data
- Integrated (return ratios instead of values) data
- Integrated and scaled to percents (return ratios multiplied by 100%) data
- Integrated and then max-min-scaled data

For each enterprise the following methods ARIMA model(Box & Jenkins 1970), GARCH Model (Engle 1982; Bollerslev 1986), Brownian motion(Einstein 1905; Wiener 1923), Exponential Smoothing(Holt (1957; Winters 1960), Long Short-Term Memory neural network(Hochreiter, & Schmidhuber. 1997). and Gated Recurrent Unit neural network (Cho et al. 2014) have been used for forecasting seven session values. The learning period was established as and seven consequent stock values gave been used for estimating the quality of forecasting

To evaluate the quality of forecasting in each data preprocessing scenario

4. Results

Table 2 presents the results of simulation study for ARIMA Model. Tho lowest values or Root Mean Square Error, Mean Absolute Error and Mean Absolute Percentage Error are achieved for scenario in which data is transformed by calculating return ratios (which may be treated as a form of differentiation)

Table 2. The simulation study results for ARIMA model

	STRATEGY	RMSE	MAE	MAPE
ARIMA	RAW	1.703619	1.466516	1.240773
	RETPERCENT	1.413970	1.168053	0.989860
	RETRATIOS	**1.325411**	**1.066128**	**0.904349**
	SCALER	1.515548	1.258586	1.074684
	SCALER RETRATIOS	1.987031	1.651763	1.396822

Source: Own calculations with *yfinance, numpy, pandas, statsmodel, keras, sci-kit learn* python packages

Table 3 presents the simulation study result for Brownian motion method. The best results are achieved, although, which may not be a surprise the overall results are worse than for other methods/ models.

Table 3. The simulation study results for Brownian motion method

	STRATEGY	RMSE	MAE	MAPE
BROWNIAN	RAW	**6.9271061**	**6.4963058**	**5.5047914**
	RETPERCENT	11.7456434	11.7447571	9.9846868
	RETRATIOS	11.7633319	11.7625180	9.9998469
	SCALER	11.7346792	11.7337449	9.9752872
	SCALER RETRATIOS	11.7423766	11.7414762	9.9818863

Source: Own calculations with *yfinance, numpy, pandas, statsmodel, keras, sci-kit learn* python packages

Table 4 describes the results for five data preparation scenarios For GARCH model. The lowest error coefficients are reached in the procedures with integration included. Both return ratios calculation and return ratios multiplied by one hundred may be stated as best method – the difference between them is unsignificant.

Table 4. The simulation study results for GARCH model

	STRATEGY	RMSE	MAE	MAPE
GARCH	RAW	1.987084	1.700064	1.437339

	RETPERCENT	**1.588790**	**1.342292**	**1.136150**
	RETRATIOS	**1.588994**	**1.342695**	**1.136499**
	SCALER	1.810632	1.347512	1.165044
	SCALER RETRATIOS	2.072017	1.720957	1.455080

Source: Own calculations with *yfinance, numpy, pandas, statsmodel, keras, sci-kit learn* python packages

Table 5 presents the results for Gated Recurrent Unit neural network Tho lowest values or Root Mean Square Error, Mean Absolute Error and Mean Absolute Percentage Error are achieved for scenario in which data min-max scaled with significant difference to other data preparation scenarios,

Table 5. The simulation study results for Gated Recurrent Unit model

	STRATEGY	RMSE	MAE	MAPE
GRU	RAW	2.201235	1.914494	1.622920
	RETPERCENT	3.120753	2.399227	2.026249
	RETRATIOS	2.408659	2.041601	1.725056
	SCALER	**1.493877**	**1.198476**	**1.024842**
	SCALER RETRATIOS	2.131069	1.778118	1.503224

Source: Own calculations with *yfinance, numpy, pandas, statsmodel, keras, sci-kit learn* python packages

For Holt Winters exponential smoothing the best strategy is leave raw data unchanged, however the error differences in table 6 are relatively smaller than the ones for other models/ methods

Table 6. The simulation study results for Holt Winters exponential smoothing

	STRATEGY	RMSE	MAE	MAPE
HOLT_ WINTERS	RAW	**1.657260**	**1.413981**	1.196456
	RETPERCENT	1.765113	1.535244	1.301922
	RETRATIOS	1.766970	1.537145	1.303544
	SCALER	1.724796	1.472040	**1.186088**
	SCALER RETRATIOS	1.856842	1.552035	1.312818

Source: Own calculations with *yfinance, numpy, pandas, statsmodel, keras, sci-kit learn* python packages

For the Long Short-Term Memory (LSTM) network, the second type of recurrent neural network analyzed, the results (presented in Table 7) indicate a slight superiority of the min-max scaler approach, although its advantage over the return ratios strategy is marginal.

Table 7. The simulation study results for Long Short-Term Memory model

	STRATEGY	RMSE	MAE	MAPE
LSTM	RAW	2.363669	2.013477	1.703297
	RETPERCENT	3.420169	2.909002	2.460272
	RETRATIOS	1.533231	1.286535	1.091071
	SCALER	**1.493877**	**1.198476**	**1.024842**
	SCALER RETRATIOS	1.978351	1.639916	1.386805

Source: Own calculations with *yfinance, numpy, pandas, statsmodel, keras, sci-kit learn* python packages

Author wants to emphasize at this point that the aim of the study was to compare data preparation strategies not models forecasting abilities themselves (which was examined in many studies in the literature of the subject)

The results above are sufficient to state some suggestions for proper design of benchmark and real forecasting studies (in fact to confirm the theoretical tips from the literature of the subject).

5. Discussion and Conclusions

The following conclusions may be drawn from research study

- Data preparation for GARCH and ARIMA requires the integration step
- The best strategy for LSTM and GRU is to min-max scale raw data
- Brownian motion and exponential smoothing don't require transformation step

It is worth to emphasize that benchmark analysis comparing forecast measures with the same data preparation step are not entirely factually correct because every method needs a different preparation, and only the result after reverse transformation should be compared. Similarly for real forecasting procedures the data preparation and transformation are crucial for the final quality.

References

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

Box, G. E. P., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2016). Time series analysis: Forecasting and control (5th ed.). Wiley.

Brownlee, J. (2019). *Deep learning for time series forecasting: Predict the future with MLPs, CNNs and LSTMs in Python*. Machine Learning Mastery.Enders, W. (2015). Applied econometric time series (4th ed.). Wiley.

Cho, K., van Merriënboer, B., Bahdanau, D., & Bengio, Y. (2014). On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation* (pp. 103–111). Association for Computational Linguistics. https://doi.org/10.3115/v1/W14-4012

Chollet, F., & Allaire, J. J. (2018). *Deep learning with R*. Manning.

Einstein, A. (1905). On the movement of small particles suspended in stationary liquids required by the molecular-kinetic theory of heat. *Annalen der Physik*, 17, 549–560.

Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of UK inflation. Econometrica, 50(4), 987–1007.

Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research, 270*(2), 654–669. https://doi.org/10.1016/j.ejor.2017.11.054

Francq, C., & Zakoïan, J.-M. (2019). GARCH models: Structure, statistical inference and financial applications (2nd ed.). Wiley.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.

Hamilton, J. D. (1994). Time series analysis. Princeton University Press.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Holt, C. C. (1957). Forecasting trends and seasonals by exponentially weighted moving averages. *O.N.R. Memorandum*.

Tsay, R. S. (2010). Analysis of financial time series (3rd ed.). Wiley.

Zhang, D., Chen, S., & He, Z. (2020). A deep learning framework for financial time series using stacked autoencoders and long–short term memory. *PLOS ONE, 15*(1), e0226997. https://doi.org/10.1371/journal.pone.0226997

Wiener, N. (1923). Differential space. *Journal of Mathematics and Physics*, 2(1), 131–174.

Winters, P. R. (1960). Forecasting sales by exponentially weighted moving averages. *Management Science*, 6(3), 324–342.

Article title in Polish ## If you don’t know Polish, Journal Editors will assist you with this section

Streszczenie

Cel: Sformułowanie sugestii, która metoda wstępnego przetwarzania danych jest preferowana dla różnych metod prognozowania, w tym podejść uczenia maszynowego, szczególnie w prognozowaniu wartości akcji.

Metodyka: Badanie dotyczące predykcji rzeczywistych wartości akcji na przykładzie 10 średniej wielkości przedsiębiorstw notowanych na NYSE, porównujące pięć scenariuszy przygotowania danych.

Wyniki: Potwierdzenie założeń teoretycznych oraz rekomendacje dotyczące właściwego projektowania studiów porównawczych (benchmark) i rzeczywistych modeli prognostycznych.

Implikacje i rekomendacje: Zgodnie z literaturą przedmiotu, dla modeli opartych na procesach stochastycznych, takich jak ARIMA i GARCH, pożądaną transformacją jest przejście na stopy zwrotu. W przypadku podejść uczenia maszynowego, w szczególności rekurencyjnych sieci neuronowych typu Long Short Term Memory (LSTM) i Gated Recurrent Unit (GRU), zalecana jest transformacja danych przy użyciu skalowania min-max. Dla metod wygładzania wykładniczego i ruchu Browna najlepsze wyniki osiągnięto dla danych nieprzekształconych (surowych). Sugestie dla badaczy przygotowujących studia porównawcze i rzeczywiste modele prognostyczne zostały przedstawione w końcowej części artykułu.

Oryginalność/wartość: Temat przygotowania i transformacji danych, choć powszechnie obecny w literaturze przedmiotu, rzadko jest potwierdzany przez badania na rzeczywistych zbiorach danych. Według wiedzy autorów, tego typu analiza nie była dotąd przeprowadzona na danych rzeczywistych.

Słowa kluczowe: Prognozowanie, modele procesów stochastycznych, rekurencyjne sieci neuronowe.