Journal of
Data Acquisition and Processing

ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.7765659

PERFORMANCE OF MACHINE LEARNING ALGORITHMS FAVOURING INVESTORS IN GENERATING PROFITS.

Mrs. Keerti Mahajan[#1], Dr. Sampada Galvani[*2]

[#]Research Student, Bharati Vidyapeeth Institute of Management,
Kolhapur

[1]kirtisheetal99@gmail.com

[*]Asso. Prof., Bharati Vidyapeeth Institute of Management,Kolhapur.
[2]sg4906@gmail.com

ABSTRACRT

Based on the research into computational learning theory and pattern recognition, a new subject of computer science was born: machine learning. Continued usage of the Capital Asset Pricing Model (CAPM) based on information from the US equities markets allows for accurate price projections. Training on time series data for the whole stock universe and external macroeconomic factors allows the applicable Machine Learning models to significantly outperform the CAPM on out-of-sample (OOS) test data. Scores for categorization models varied widely; among the machine and deep learning models examined, the long short-term memory (LSTM) model showed greater accuracy. In most directional evaluation indicators, the experimental results show that traditional ML algorithms perform better. Furthermore, all ML algorithms are vulnerable to fluctuations in transaction cost, which can negatively affect trading performance. Yet, the effects of both explicit and implicit transaction costs on market activity are distinct. This research is important because it allows us to determine which algorithm is most lucrative across various markets.

Keywords: Recognition, Machine, Learning, Accuracy, Risk,

INTRODUCTION

Based on the research into computational learning theory and pattern recognition, a new subject of computer science was born: machine learning. Researchers in the field of machine learning focus on developing and studying algorithms with the ability to "learn" from and make predictions based on previously collected data. These algorithms, rather than blindly executing a set of predetermined steps, take data as input and use it to make inferences or decisions. Machine learning has numerous parallels and overlaps with computational statistics, another discipline that emphasizes prediction. Mathematical optimization is a close relative that supplies the discipline with tools, concepts, and areas of study. Machine learning is used in place of explicit algorithms when such cannot be easily developed or written. Several common uses include spam filters, OCR, search engines, and computer vision. Despite the latter's stronger focus on exploratory data processing, data mining and machine learning are often conflated. "May be considered as two sides of the same subject," the authors write of pattern recognition and machine learning.

Predicting and projecting asset prices on global financial markets continues to be one of the most challenging and exciting topics in quantitative finance, both for academics and professionals. Researchers and investment organizations focus on computer science approaches, because of the exponential increase in computer power and data, several new professions are emerging, most notably data science, artificial intelligence (AI), and machine learning (ML). More than 2.5 quintillion bytes of data are created and collected daily by humans. It is anticipated that by 2020, More than 90% of all data ever produced by humans will have been created in the previous few years, which is more than 40 Zettabytes.

Since the inception of the capital markets, investors have looked for ways to gain an advantage over their competitors. One area that has always piqued investor attention is the capacity to accurately predict time series. Fast and effective decision-making are more crucial than ever because of the expansion of data sources available and the growing interconnection of investors. Using noisy, non-stationary data, approximating non-linear functions, and spotting latent patterns in datasets are all possible with machine learning techniques.

We choose to utilize an objective approach to find those horizons that best describe investors' buy or sell operations in order to allay potential concerns caused by the analyst's subjective judgements as well as to prevent dismissing a potentially significant predictor. We do this by utilizing an elastic net technique, a reliable feature selection method that was taken from the literature on machine learning. The elastic net's main advantages are its straightforward loss function, which is similar to a regression, and its robustness in preventing overfitting by employing an ideal convex mix of the Lasso and Ridge regularization techniques. Overfitting can happen when an algorithm learns the dynamics of the target variable and performs very well on the training dataset but poorly on other datasets in terms of predictability. For researchers, assessing the likelihood of overfitting is crucial because it could damage the model. We are aware that our approach attempts to mitigate some of the risks associated with overfitting. Large weights in the model are penalized by the Ridge and Lasso algorithms. They are able to alleviate worries about overfitting because they tend to lessen the model's complexity.

2. SYSTEM PRINCIPLE

2.1 Machine Learning

Essentially, the goal of machine learning is to derive insights from datasets (Kubat, 2017, p. 1). The majority of current stock market forecasting applications of machine learning make use of supervised learning. the whole process of using supervised learning for stock market forecasting. Initially, you'll need to choose a time period for which to analyze data and/or information. If you're facing a classification difficulty, you either know or can reasonably guess the target class. Pre-processing is the first step, and it entails preparing the associated data by, for example, eliminating any out-of-context or unnecessary information. Next, the underlying time-series data, such the closing price information, may be used to derive technical indications. Scaling and dimensionality reductions are used to the cleaned data, which may include technical indications, in order to get important variables and filter out irrelevant ones. Predictive accuracy is typically improved by using preprocessed data (Chen et al., 2019).

Authors suggested use deep learning architectures like CNN and RNN to further improve this method. Derivative characteristics, such as a set of technical indicators, may be obtained from the smoothed stock data. They found that a decision-tree-based ensemble machine learning algorithm called XGBoost performed much better than the standard non-ensemble learning method. This algorithm's reliability is measured by calculating its predictive accuracy, which may be done by calculating the receiver operating characteristic (ROC) curve and the area under the ROC curve. A random forest classifier was developed (Khaidem et al., 2016) using technical indicators generated from stock data. A number of characteristics, including accuracy, precision, recall, and specificity, are computed in order to assess how resilient the model is at predicting changes in stock price. In order to assess the model, ROC curves were also plotted and looked at. Prediction accuracy for the selected data was found to be between 85% and 90%, which is extremely good. An SVM, a kind of classic machine learning model, was fitted to feature data produced from price volatility, price momentum, sector volatility, and other factors, who discovered that the fitted model showed no signs of over-fitting.

Many different machine learning methods have been proposed for use in forecasting the stock market. Among these, the most often used are artificial neural networks (ANNs) (Nermend & Alsakaa, 2017;) and support vector machines (SVMs) and its derivatives (Ebrahimpour et al., 2011; Pan et al., 2017). There is a wealth of research covering fuzzy theory-based stock market forecasting models due to the widespread use of intelligent systems in fuzzy theory for handling data uncertainty. Other types of fuzzy models include the Takagi-Sugeno-Kang (TSK) type and fuzzy time-series (Cagcag Yolcu & Alpaslan, 2018; Wei et al., 2011) models (Pal & Kar, 2019).

2.2 Deep Learning

Used historical stock market data and pertinent news sentiments to build an AI framework predominantly made up of RNN, feed-forward neural network (FFNN), support-vector regression (SVR), and SVM for directional prediction of stock price movements. The authors discovered that SVM had the highest directional prediction accuracy. The authors recommended including additional significant factors like technical indicators or pertinent news (Moukalled et al. 2019).

Implemented intraday trend forecasting of the S&P 500 Index using deep learning algorithms using a data set consisting of technical indicators and titles of financial news articles. The RNN and CNN architectures were implemented by the authors to carry out conventional natural language processing (NLP) tasks. Input layer, recurrent layer, convolution layer, and output layer are the four phases that make up the model's architecture. The outcomes showed that RNN performs better for stock market forecasting when choosing context information and modeling complicated temporal characteristics. The outcomes also showed that when it came to extracting textual semantics, CNN performed better than RNN. The model was trained and tested on stock-trading simulations by the authors using reinforcement learning techniques (Vargas et al., 2017).

We compared the results of using reinforcement learning algorithms for asset allocation in US Equities with those of using more traditional portfolio management strategies, such as mean-variance, minimum-variance, risk-parity, and equally weighted, using a variety of learning architectures, including long short-term memory (LSTM) networks, convolutional neural networks (CNN), and recurrent neural networks (RNN) (Noguer I Alonso and Srivastava 2020). When tested with a simple reward function and a stock price time series, deep reinforcement learning was shown to be superior to more traditional methods. Among the models we looked at, CNN without turnover control had the highest returns after accounting for costs. In the future, researchers might examine how different incentive structures, reinforcement learning paradigms, and diversity play a role in this context. The role of exogenous variables and the use of technical indicators are also crucial factors to think about.

Used 10 technical indicators to anticipate stock market movements using a variety of machine and deep learning algorithms, including LSTM, a variation of RNN (Nabipour et al. 2020). In this research, LSTM performed much better than the other machine learning models analyzed. Shyam and Vinayak (2020) used the LSTM model with ReLU in the hidden layer and Sigmoid activation function in the output layer to analyze the open, close, high, and low price and volume of Google and Amazon. Together, dropout for overfitting prevention and RMS prop as optimizer were utilized by the authors. The findings showed that Amazon and Google had lowest losses of 0.00085 and 0.0018, respectively. Overall, Amazon has a 52.23 percent accuracy, while Google has an 89.44 percent accuracy.

2.3 Reinforcement Learning

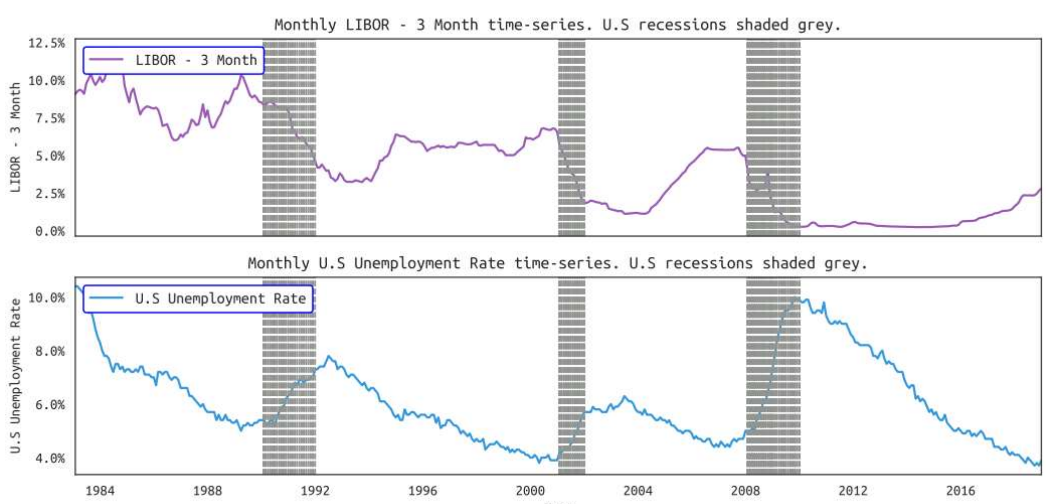
Employed reinforcement learning using conventional time-series stock price data and news headline feelings as input. Because there is a dearth of label data in the financial market, reinforcement learning may outperform supervised and unsupervised methods (Nan et al. 2020). This is further supported by the fact that, in reinforcement learning, the algorithm itself learns and generates a set of rules based on feature data. The authors examined two approaches, one using an agent given sentiment data and the other using an agent given no sentiment data. The findings reveal that the Sharpe ratio is much higher for a sentiment-informed AI than for one without such data. This study demonstrates the value of news emotion and reinforcement learning in forecasting stock trends on the stock market. Using actual stock data, we built a deep reinforcement learning model and compared it to a state-of-the-art supervised deep learning forecasting model. Given the intricacy of the stock market, the author feels the reinforcement learning model has huge potential to provide stock trading signals (Dang 2019). The results showed that Q-learning-based reinforcement learning model variations might provide a lucrative trading signal with as few as a few hundred data points.

Conducted a study of numerous studies on reinforcement learning and used prices, volumes, technical indicators, economic indicators, and correlation coefficients are all examples of output variables that use historical data as input variables (Meng and Khushi 2019). According to several research findings, when reinforcement learning is used appropriately, performance can significantly outperform baseline models, especially whether the success of a trade is measured by how well it makes money or how well it anticipates future trends. According to the results of earlier studies, reinforcement learning failed to perform well when there was a substantial price difference between the training and testing data. The authors provided a plausible justification, arguing that since frequent trading of illiquid assets resulted in high transaction costs and decreased profitability because liquidity plays a crucial role in profitability and reinforcing.

3. OPERATION MODE

We undertake a thorough empirical examination of 782 publicly listed U.S. stocks that have existed and endured throughout a 30-year period between 1983 and the beginning of 2019 and are currently available on the Wharton Research Data Services (WRDS) cloud [18]. The study compares the CAPM with the Machine Learning techniques provided in Table 1 to estimate the annual returns of this stock universe. All data was retrieved with the help of the WRDS. The WRDS cloud was scraped for information by means of the CRSP and S&P Global Market Intelligence Compustat databases. Data was gathered on asset prices on a monthly and annual basis, as well as on macroeconomic indicators and accounting financial statements [19]. A sampling of the macroeconomic time series features for the United States is shown in Figure 1, including monthly data on the GDP, bond rates, and consumer price indices.

FIGURE 1: EXAMPLE MACROECONOMIC TIME-SERIES DATA USED IN MACHINE LEARNING MODELS



Proprietary python software was built on top of the official WRDS python APIs to automate the extraction and translation of asset price data, as well as the training and testing of Machine Learning and CAPM models to provide reproducible outcomes. In accordance with, we apply the CAPM model expressed in equation 1 to a value-weighted (VW) U.S. the S&P 500 index instead of an equally weighted (EW) index to characterize the market portfolio. Ten-year U.S. government bonds were used to estimate the risk-free rate. yields on U.S. Treasuries derived from studies of CAPM's performance using real-world market data. In addition, a time horizon of three to eight years is recommended for estimating the asset's beta; this is the period of time during which we calculated historical returns. 4. The annualised average rate of return using monthly returns 5 data throughout the three-year time period was computed using basic arithmetic as opposed to the geometric mean, based on the literature. The next step in developing our machine learning models is to collect the data needed for the training and testing sets referenced. Machine learning researchers refer to problems with time series as "feature leakage," while financial and economics researchers use the term "look-ahead bias," therefore it's important to keep the training and test data in chronological order. Assuming we keep the temporal order of our time series data, the standard approach in the literature is to do an OOS (out-of-sample) assessment, in which a sub-sample at the end of the time series is held out for validation. Sequential evaluation will be used to define this kind of verification.

New varieties of blocked cross-validation have been developed by academics in recent years, and they have been used particularly to time-series problems in order to create robust models that do not over-fit datasets. Since Heston and Sadka have shown the seasonality of yearly returns on foreign equities, these innovative methodologies for evaluating time series predictions may be of significant interest to practitioners and industrial researchers. Since recent empirical research by Cerqueira et al. and Mozetic et al. did not demonstrate a major improvement of these novel cross-validation procedures for non-stationary time series data, a standard sequential evaluation methodology was applied in this situation. Near the conclusion of the time series, over 30% of the data was held back for testing; this represents about 6 years of unobserved data from 2012 to 2018. The final training and test data sets for the Machine Learning model includes around 200 variables linked to business financial performance and exogenous U.S. macroeconomic indices for the three years preceding to the forecast year.

4. ML ALGORITHMS

4.1 ML Algorithms and Teir Parameter Settings.

The goal of a ML algorithm is to accurately classify dataset D's class labels. In this paper, we will use twelve different ML models as classifiers to anticipate stock price swings: six classic ML models (LR, SVM, CART, RF, BN, and XGB) and six deep neural network models (MLP, DBN, SAE, RNN, LSTM, and GRU). Tables 1 and 2 display the primary model parameters and training parameters of these ML learning algorithms, respectively. Tables 1 and 2 contain feature and class labels formatted to be read by a variety of ML algorithms written in R. A matrix with m rows and n columns is represented by Matrix (m, n); a tensor is represented by Array (p, m, n), where each layer of the tensor is a Matrix (m, n) and the height of the tensor is p; and a vector is represented by c (h1, h2, h3,...), where the length of the vector is the number of hidden layers and the I Each WFA round in the experiment uses data from the previous 250 trading days as training samples, and n=44 indicates that each day's data consists of 44 features.

See Table 2 for the predetermined settings for DNN model parameters like activation function, learning rate, batch size, and epoch used by the corresponding R programs' algorithms.

4.2 WFA Method.

WFA is an effective form of rolling exercise. Instead of using historical data, we only use the most recent data to train the model, and then use that model to make predictions for data that isn't part of the training set (the testing dataset) in the future. After that, training for the next round is carried out using a new training set, which is the previous training set walked forward one step. When applied to real-time trading, WFA can strengthen the trading strategy's reliability and user faith. Predictions of future trends in stock prices are used as trading signals in this paper by employing ML algorithms and the WFA technique. The previous 250 days' worth of data (an entire year's worth) serves as the training set, while the upcoming 5 days' worth of data (a full week's worth) serves as the test set in each stage. Because there are 2,000 days of trading data per stock, it takes (2000-250)/5 = 350 training sessions to generate 1,750 predictions, or trading signals, for a daily trading strategy.

4.3 The Algorithm Design of Trading Signal.

Here, we use ML algorithms as classifiers to daily stock price data from SPICS and CSICS, and utilize the resulting trading signals to make profitable investments. Each ML algorithm is trained using the WFA technique.

Table 1: Main parameter settings of traditional ML algorithms

	Input Features	Label	Main parameters
LR	Matrix(250,44)	Matrix(250,1)	A specification for the model link function is logit.
SVM	Matrix(250,44)	Matrix(250,1)	The kernel function used is Radial Basis kernel; Cost of constraints violation is 1.
CART	Matrix(250,44)	Matrix(250,1)	The maximum depth of any node of the final tree is 20; The splitting index can be Gini coefficient.
RF	Matrix(250,44)	Matrix(250,1)	The Number of trees is 500; Number of variables randomly sampled as candidates at each split is 7.
BN	Matrix(250,44)	Matrix(250,1)	the prior probabilities of class membership is the class proportions for the training set.
XGB	Matrix(250,44)	Matrix(250,1)	The maximum depth of a tree is 10; the max number of iterations is 15; the learning rate is 0.3.

Table 2: Main parameter settings of DNN algorithms

	Input Features	Label	Learning rate	Dimensions of hidden layers	Activation function	Batch size	Epoch
MLP	Matrix(250,44)	Matrix(250,1)	0.8	c(25,15,10,5)	sigmoid	100	3
DBN	Matrix(250,44)	Matrix(250,1)	0.8	c(25,15,10,5)	sigmoid	100	3
SAE	Matrix(250,44)	Matrix(250,1)	0.8	c(20,10,5)	sigmoid	100	3
RNN	Array(1,250,44)	Array(1,250,1)	0.01	c(10,5)	sigmoid	1	1
LSTM	Array(1,250,44)	Array(1,250,1)	0.01	c(10,5)	sigmoid	1	1
GRU	Array(1,250,44)	Array(1,250,1)	0.01	c(10,5)	sigmoid	1	1

5. THEORETICAL CALCULATION AND ANALYSIS

We utilized the Mean Squared Error as the performance metric (MSE). Using the MSE-determined real return r and the expected return y, the following definitions apply;

$$\mathrm{MSE}(\hat{y} - y) = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2.$$

The results of the model are summarised in Table 3. Based on the results, it's clear that machine learning approaches are much superior to the more conventional Capital Asset Pricing Model. This demonstrates the efficacy and versatility of machine learning methods in economic and financial market predictions. Both the neural network and the gradient boosting tree models behaved as expected [24]. Since the Deep FNN did better than the Shallow FNN, it follows that, it's possible that future research will examine convolutional neural network and recurrent neural network structures. As computing power, data availability, and scientific innovation all rise, new model structures and algorithms will be developed, allowing the performance of these models to improve over time.

TABLE 3: MODEL RESULTS

Optimized Model	Mean Squared Error (MSE)
CAPM	1.6001
NGBoost	0.3572
XGBoost	0.3280
Catboost	0.3125
LightGBM	0.3131
Shallow FNN	0.3628
Deep FNN	0.3531

6. THE IMPACT OF TRANSACTION COST ON PERFORMANCE OF ML ALGORITHMS

A stock trading strategy's profitability may be impacted by trading costs. What may be disregarded in long-term tactics due to low transaction costs is greatly exacerbated in day trading. However, a common assumption in research on algorithmic trading is that transaction costs are irrelevant. Although textbooks may portray a perfect market, transaction costs and other real-world frictions might skew it. Commissions, exchange fees, and taxes that are disclosed in advance are examples of transparent costs. Implicit costs consist of the bid-ask spread, delay or slippage, and the accompanying market effect that must be anticipated. This section examines the impact of both explicit and implicit costs on trading performance on a day-to-day basis.

6.1 Experimental Settings and Back testing Algorithm.

Here, we simplify things by calculating the visible transaction cost as a percentage of transaction volume; calculating the implicit transaction cost is far more involved because of the need to account for the unpredictability of the market environment and stock prices. As a result, we focus only on how slippage affects trading results.

The structure of transaction fees for American equities is similar to that of Chinese A-shares. The proportion of sales we use to estimate the transparent transaction cost is between 0.2% and 0.5% and between 0.2% and 0.5% in the literature. Calculating slippage is handled differently.

The slippage is set to 0.02 in Join Quant and Abuquant, two examples of quantitative trading simulation software. When buying and selling, both parties are subject to the visible transaction cost and the implicit transaction cost. It's important to remember that the implicit transaction cost is connected to market liquidity, market information, network condition, trading software, etc., while the transparent transaction cost differs with the various brokers.

6.2 Analysis of Impact of Transaction Cost on the Trading Performance of SPICS.

Cost per trade is a major factor in overall trading success. Transparent transaction costs in US stock trading can be assessed either as a flat rate per order or per month, or as a variable rate depending on the size and frequency of trades. Customers can sometimes negotiate with brokers to arrive at a mutually agreeable price for a transaction. Costs associated with completing a trade can vary considerably between brokers. Moreover, implicit transaction costs are difficult to predict and estimate. Since this makes calculations easier, we will treat this percentage of sales as the open transaction cost. Specifically, we focus on the effect slippage has on trading performance as an implicit transaction cost.

5. CONCLUSION

Machine learning has numerous parallels and overlaps with computational statistics, another discipline that emphasizes prediction. Daily, people generate and amass more than 2.5 quintillion bytes of data. Using noisy, non-stationary data, approximating non-linear functions, and spotting latent patterns in datasets are all possible with machine learning techniques. We are aware that our approach attempts to mitigate some of the risks associated with overfitting. The outcomes illustrated the effectiveness and strength of machine learning approaches in predicting annual returns. Unlike traditional finance theories like the CAPM, the Machine Learning algorithms may include over 200 time series factors into their predictions of return for each target U.S. equities. Future studies may be conducted from the following angles, all made possible by the quickening pace of ML development and the easy availability of financial big data: (3) taking into account the influence of more sophisticated implicit transaction cost, such opportunity cost and market impact cost, on stock trading performance via the use of ML algorithms to design dynamic optimum portfolios across various companies.

REFERENCES

1. Ahmad, m. O., dennehy, d., conboy, k., & oivo, m. (2018). Kanban in software engineering: a systematic mapping study. Journal of systems and software, 137, 96–113. Http://dx.doi.org/10.1016/j.jss.2017.11.045.

2. Ahmadi, e., jasemi, m., monplaisir, l., nabavi, m. A., mahmoodi, a., & amini jam, p. (2018). New efficient hybrid candlestick technical analysis model for stock market timing on the basis of the support vector machine and heuristic algorithms of imperialist competition and genetic. Expert systems with applications, 94, 21–31.

3. Ahmed, y. A., ahmad, m. N., ahmad, n., & zakaria, n. H. (2019). Social media for knowledge-sharing: a systematic literature review. Telematics and informatics, 37, 72–112. Http://dx.doi.org/10.1016/j.tele.2018.01.015.

4. Sedighi, m., jahangirnia, h., gharakhani, m., & fard, s. F. (2019). A novel hybrid model for stock price forecasting based on metaheuristics and support vector machine. Data, 4(2), 1–28. Http://dx.doi.org/10.3390/data4020075.

5. Selvamuthu, d., kumar, v., & mishra, a. (2019). Indian stock market prediction using artificial neural networks on tick data. Financial innovation, 5(1), http://dx.doi.org/10.1186/s40854-019-0131-7.

6. Shah, d., isah, h., & zulkernine, f. (2019). Stock market analysis: a review and taxonomy of prediction techniques. International journal of financial studies, 7(2), http://dx.doi.org/10.3390/ijfs7020026.

7. Kubat, M. (2017). An introduction to machine learning (2nd Ed.). Springer Publishing Company, Incorporated.

8. Chen, Y., Lin, W., & Wang, J. Z. (2019). A dual-attention-based stock price trend prediction model with dual features. IEEE Access, 7, 148047–148058.

9. Luckyson Khaidem, Predicting the direction of stock market prices using random forest, Applied Mathematical Finance Vol. 00, No. 00, Month 20XX, 1–20

10. Nermend, Y., & Alsakaa, K. (2017). Back-propagation artificial neural networks in stock market forecasting . An application to the warsaw stock exchange WIG20. The IEB International Journal Of Finance, 15, 88–99. Niaki

11. Ebrahimpour, R., Nikoo, H., Masoudnia, S., Yousefi, M. R., & Ghaemi, M. S. (2011). Mixture of mlp-experts for trend forecasting of time series: A case study of the tehran stock exchange. International Journal Of Forecasting, 27(3), 804–816.

12. Pan, Y., Xiao, Z., Wang, X., & Yang, D. (2017). A multiple support vector machine approach to stock index forecasting with mixed frequency sampling. Knowledge-Based Systems, 122, 90–102. http://dx.doi.org/10.1016/j.knosys.2017.01.033.

13. Cagcag Yolcu, O., & Alpaslan, F. (2018). Prediction of TAIEX based on hybrid fuzzy time series model with single optimization process. Applied Soft Computing, 66, 18–33. http://dx.doi.org/10.1016/j.asoc.2018.02.007

14. Wei, L. Y., Chen, T. L., & Ho, T. H. (2011). A hybrid model based on adaptive-network based fuzzy inference system to forecast Taiwan stock market. Expert Systems with Applications, 38(11), 13625–13631. http://dx.doi.org/10.1016/j.eswa.2011.04.127.

15. Pal, S. S., & Kar, S. (2019). Time series forecasting for stock market prediction through data discretization by fuzzistics and rule generation by rough set theory. Mathematics And Computers in Simulation, 162, 18–30.

16. Nabipour, M., Nayyeri, P., Jabani, H., Shahab, S., & Mosavi, A. (2020). Predicting stock market trends using machine learning and deep learning algorithms via continuous and binary data: A comparative analysis. IEEE Access, 8, 150199–150212.

17. Shyam, R., & Vinayak, P. (2020). Stock prediction overview and a simple LSTM based prediction model. International Research Journal of Engineering and Technology (IRJET), 7(4), 5935–5940. https://www.irjet.net/archives/V7/i4/ IRJET-V7I41119.pdf

18. Zhong and Enke Financial Innovation, Predicting the daily return direction of the stock market using hybrid machine learning algorithms, Springer, (2019) 5:24

19. Moukalled, M., El-Hajj, W., & Jaber, M. (2019, September). Automated stock price prediction using machine learning. In Proceedings of the second financial narrative processing workshop (FNP 2019), September 30, Turku Finland (No. 165, pp. 16–24). Linköping University Electronic Press.

20. Vargas, M. R., De Lima, B. S., & Evsukoff, A. G. (2017, June). Deep learning for stock market prediction from financial news articles. In 2017 IEEE international conference on computational intelligence and virtual environments for measurement systems and applications (CIVEMSA) (pp. 60–65). IEEE.

21. Noguer i Alonso, M., & Srivastava, S. (2020). Deep reinforcement learning for asset allocation in US equities. arXiv preprint arXiv:2010.04404.

22. Nan, A., Perumal, A., & Zaiane, O. R. (2020). Sentiment and knowledge based algorithmic trading with deep reinforcement learning. Cornell University. arXiv preprint arXiv:2001.09403.

23. Dang, Q. V. (2019, December). Reinforcement learning in stock trading. In International conference on computer science, applied mathematics and applications (pp. 311–322). Springer.

24. Meng, T. L., & Khushi, M. (2019). Reinforcement Learning in Financial Markets. Data, 4(3), 110.

25. Shen, j., & shafiq, m. O. (2020). Short-term stock market price trend prediction using a comprehensive deep learning system. Journal of big data, 7(1), http: //dx.doi.org/10.1186/s40537-020-00333-6.

26. Shi, l., teng, z., wang, l., zhang, y., & binder, a. (2019). Deepclue: visual interpretation of text-based deep stock prediction. Ieee transactions on knowledge and data engineering, 31, 1094–1108. Http://dx.doi.org/10.1109/tkde.2018.2854193.

27. Shynkevich, y., mcginnity, t. M., coleman, s. A., & belatreche, a. (2016). Forecasting movements of health-care stock prices based on different categories of news articles using multiple kernel learning. Decision support systems, 85, 74–83. Http://dx.doi. Org/10.1016/j.dss.2016.03.001.

28. Singh, r., & srivastava, s. (2017). Stock prediction using deep learning. Multimedia tools and applications, 76(18), 18569–18584

29. Snyder, h. (2019). Literature review as a research methodology: an overview and guidelines. Journal of business research, 104, 333–339

30. Zhong, x., & enke, d. (2017). Forecasting daily stock market return using dimensionality reduction. Expert systems with applications, 67, 126–139. Http://dx.doi.org/10. 1016/j.eswa.2016.09.027.

31. Zhou, p. Y., chan, k. C., & ou, c. X. (2018). Corporate communication network and stock price movements: insights from data mining. Ieee transactions on computational social systems, 5(2), 391–402. Http://dx.doi.org/10.1109/tcss.2018. 2812703.