Forecasting and Analyzing Financial Markets with Deep Learning Methods

Vladimir Pyrlik

HSE University

Yuriy Sosnin

HSE University

Abstract

Deep Learning is a powerful Machine Learning tool, especially suitable for for complex, nonlinear environments, like Natural Language Processing, Computer Vision, or Time Series forecasting. In recent years, there was a rise in applications of Deep learning to Financial Time Series. However, this stream of literature is inherently technical. Machine Learning specialists view financial data as one of time series domains, mostly uninterested in theoretical considerations or meaningful interpretation of results. We aim to shorten this gap.

We deploy LSTM, CNN and CNN-LSTM architectures in the task of forecasting daily return of Russian Exchange Index (IMOEX), and propose the use of 6 groups of macroeconomic variables and index constitutes to try enhance models' performance. We are unable to outperform zero naive predictor; however, we find that all of 6 groups (Underlying Stocks, Sectoral Indexes, Bonds, Commodities, Exchange rates and Foreign market indexes) improve models' forecast quality. In addition to this, we employ novel approach to Machine Learning interpretation, using SHAP values as measures of feature importance. We detect the effect of gold on Russian stock market during our considered period of late 2019, as well as several other interesting patterns.

Keywords: stock market prediction, financial time series, deep learning, feature attribution

Contents

1. Introduction

Financial time series forecasting has been of great interest for both researches and practitioners for a number of decades. Despite Efficient Market Hypothesis, claiming in its semi-strong form that all publicly available information about a stock is immediately incorporated in the stock's value (Fama, 1970), financial markets are still generally considered predictable. There are established stylized facts about predictability of financial markets, like existence of weekend and holiday effects, momentum and value effects, as well as importance of fundamental firm and macroeconomic variables. (Schwert, 2003; Fama, 1981; Campbell, 1987; Rapach et al., 2005). Still, financial market forecasting is not an easy task. Highly complex nonlinear relationships, noisy and temporally unstable patterns, and the market forces themselves, forcing away any possible arbitrage, all make financial forecasting notoriously difficult.

For a long time, simple linear ARIMA models have been the most popular, established and best-performing models for time series forecasting. They still remain a solid benchmark, and, combined with Machine Learning methods, continue to outperform other methods in time series forecasting competitions (Makridakis et al., 2020). However, linear models like ARIMA or GARCH by their design remain limited in their ability to model a field as complex as financial markets. At the same time, in recent years, a surge in the power of Machine Learning, and especially Deep Learning, has brought dramatic advances to many fields, like Computer Vision, Nature Language Processing or Speech Recognition. Significant effort is being made by ML specialists to successfully apply Neural Networks to time series problems, and, by extension, financial forecasting problem (Sezer et al., 2019).

The two main applications of Deep Learning in financial time series domain remain price and trend prediction. Despite being different problems (regression and classification respectively), they generally share the same methods and empirical procedures, so it makes sense to review them in tandem.

The basic Neural Network architecture is Multilayer Perceptron (MLP).

Broadly speaking, it is a combination of parametric matrix multiplications with non-linear activation function applied to each of them, stacked together (for detailed review of Neural Network architectures and training procedures, see, among others, Sezer et al. (2019)) Despite its long history of applications and proven modeling capability, it is rarely used as an end-to-end model, often being just the final building block of more sophisticated architectures.

Most of NN application fields have their own architectures, designed to effectively utilize domain-specific data structures. Recurrent Neural Networks, first introduced in the field of Natural Language Processing, are created for dealing with sequenced data. An RNN, instead of treating time series as a vector, rolls over its instances with a single cell. At each iteration, its weights get updated to accommodate new information for a particular point in time; in the end, a prediction is made based on those representations. There are several designs of RNN cells, and the most widely used is called LSTM, Long Short-Term Memory (Greff et al., 2017). Unlike default RNN cell, LSTM cell also receives its previous state as an additional input, thus being able to better control the flow of information across time. This property of memorizing both short- and long-term temporal dependencies makes it particularly efficient in time series with long and complex autocorrelations.

LSTM is extensively used as a go-to model for time series forecasting (Fischer and Krauss, 2018). There are many variations of the model architecture: Bi-Directional LSTM, Encoder-Decoder LSTM, Attention-LSTM (Chen and Ge, 2019; Istiake Sunny et al., 2020). Sometimes LSTM are also enhanced by additionally preprocessing the input series. Different data transformations are often applied to encode input series into some another dimension, reducing the noise or extracting more informative features. Some of the approaches include PCA (Ma et al., 2019; Gao and Chai, 2018) and Wavelet transforms (Liang et al., 2019; Yan and Ouyang, 2018). One more way is to use hybrid architectures, like CNN-LSTM.

Convolutional Neural Network (CNN), an architecture mainly designed for image processing, uses special Convolution layers to extract spatial information form multidimensional inputs. In the field of Computer Vision, 2-dimensional convolutions are used on images, extracting specific patterns of lower dimensionality, which are then effectively processed by a Perceptron. Similar approach works on time series, albeit 1-demensional, across time. CNNs are used both as end-to-end models and as encoding stage for RNNs (Gunduz et al., 2017; Mehtab and Sen, 2021; Hoseinzade and Haratizadeh, 2018; Lu et al., 2020; Livieris et al., 2020).

Besides RNNs and CNNs, there is also a growing number of advanced state-of-the-art architectures, like Transformers or Graph Neural Networks (Matsunaga et al., 2019; Lim et al., 2021).

Regarding the data used, most of researchers employ DL architectures in univariate prediction or classification task, with OHLCV (Open, High, Low, Close, Volume) data as inputs and a single y_{t+1} value or direction as outputs. It has to be noted that this methodology is somewhat questionable, as models use prices as both inputs and target variables, and not returns or some other stationary series. Essentially, their success depends upon a fortunate choice of time period, with prices not falling or rising dramatically during it. Even Deep Learning methods, in fact, can not deal with non-stationary data; a standard NN is not able to reasonably predict values in ranges it has not encountered during training.

Some authors use additional data to enhance model performance. The often choice is technical indicators: in many works they are included as additional features to increase dimensionality and supposedly provide the model with new information (Nelson et al., 2017; Singh and Srivastava, 2017; Zhang and Tan, 2018; Song et al., 2019, and many others). Sometimes, fundamental company characteristics like Book-to-Market, Dividend-Price Ratio, Earning-per-Share, etc. are included (Feng et al., 2018). There is also a stream of literature incorporating complex information like tweets or news articles into forecasting framework, using different specific NN architectures for processing that data (Huang and Liu, 2020; Kordonis et al., 2016; Li et al., 2021).

However, fundamental macroeconomic features are rarely used, despite wide support of those factors being causally connected with the stock market. One of the rare examples is Chen and Huang (2021). Authors test, whether information on gold and oil prices is suitable for predicting future stock movement using CNN. Several models with different input features are compared, models including gold and oil result in better classification for stocks in some industries.

Among macroeconomic variables widely considered to be affecting financial markets, are commodity prices, exchange rates, interest rates, foreign markets. Interest rate is the variable often considered the most important indicator of stock prices (Modigliani and Cohn, 1979). According to present value models, stock prices are discounted future cash flows, and interest rate is what they are being discounted by; higher interest rates make stock less attractive to rational investors, an vice versa. A rise in interest rate also means increases in debt maintenance costs and decreases in revenues of companies. At the same time, as CAPM suggests, higher interest rates makes holding bonds more attractive to investors than stocks, requiring higher returns on equity for the same level of risk.

Oil is another major contributor to fluctuations on financial markets. There are various theoretical connections between the variables (Degiannakis et al., 2018; Nandha and Faff, 2008). Oil plays an important role in cost structure of many firms, through primary product costs, energy costs and transportation costs, while also being the main output and determining profits of oil-producing firms. Oil-producers' stocks tend to move in the same direction as oil price as it affects their profits, while oil-consumers experience a rise or reduction in their costs which should drive their stocks' prices in the opposite direction (Basher and Sadorsky, 2006; Filis et al., 2011).

Gold is often considered a safe heaven in financial markets. Gold, seen as uncorrelated with other markets and safe in relation to inflation risks, is used as a hedge by investors (Baur and Lucey, 2010). There is a wide empirical support for the notion of gold prices being significant predictors of stock markets (Choudhry et al., 2015; Arfaoui and Ben Rejeb, 2017; Al-Ameer et al., 2018). An argument is also made about connection between gold volatility and stock markets: if volatility increases, hedging opportunities become less attractive, leading to unsafe investment conditions on the stock market (Baur, 2012; Gokmenoglu and Fazlollahi, 2015; Contuk et al., 2013).

Along with commodity prices, exchange rates are the other class of variables closely related to financial markets. Similar to the price of oil, exchange rates affect future cash flows of exporting and importing firms (Dornbusch and Fischer, 1980). Depreciation of domestic currency benefits exporters, increasing their profits, and hurts importers, increasing their costs, and, through this, affecting their stock prices.

Finally, world financial markets are interconnected. Information transfers between the markets, strengthening returns correlation. Investors seek hedging of national risks, and by doing so transfer some of it internationally. There is evidence of return and volatility spillovers between global markets, like London, New York and Tokyo exchanges, as well as regional exchanges in Europe, Asia and Latin America (Beirne et al., 2010; Kim and Rogers, 1995).

There are studies examining causal determinants of Russian stock market. Korhonen and Peresetsky (2016) consider the period between 1997 and 2012 and oil prices and eastern European stock markets performance as variables to affect stock market in Russia. According to their findings, oil prices stopped being significant after 2006, while correlation with other markets increased in that period. Robert D. Gay (2008) studies several emerging economies' markets, including Brazil, India, China and Russia in the period of 1999-2006. They report a non-significant result for both oil prices and exchange rates and all four countries. Lozinskaia and Saltykova (2019) study the impact of oil prices, exchange rates, foreign stock indexes and interest rates on Russian stock market from 2003 to 2018. The authors document a varying relationship, with variations caused by structural breaks.

This economic research, performed mostly by ARIMA-GARCH models to determine causality, poses a question of applicability of macroeconomic variables to Machine Learning stock market prediction. A number of authors attempt to check that applicability of other variables to their Deep Learning models' performance (Oriani and Coelho, 2016; Vargas et al., 2018). Usually it is done by iteratively excluding different variable groups from the set of inputs, re-training the model and re-evaluating it. If the model performs better with some variables included than without them, it can be said that they provide some important information about the target series.

Another, novel approach to analyze variable contribution would be to use a feature attribution method like SHAP (Lundberg and Lee, 2017). Developed under the paradigm of Explainable AI, SHAP tries to deal with the black-box nature of neural networks, and attribute prediction to input features, making for a human understandable explanation. Its concept is the following: for each input-prediction pair, it locally represents prediction as a linear combination of inputs, using Shapely values from coalition game theory. Shapely value is a player's contribution to the outcome of coalition game, evaluated with their mean marginal contribution across coalition permutations. SHAP does essentially the same for inputs of a prediction algorithm.

Attribution methods similar to SHAP has been adopted in some streams of empirical machine learning literature, and especially in medicine (Singh et al., 2020). When applied to image recognition problems, they highlight pixels on original image, which the model deemed important for the prediction. For example, images of brain scans are used to predict brain tumor or Alzheimer's disease, and the models are successfully interpreted by attribution techniques (Eitel et al., 2021; Pereira et al., 2018). Chen et al. (2019) use SHAP to interpret Gradient Boosting Machine predictions of Extubation Failure for intensive care unit patients; Kim and Kim (2022) use it in Random Forest explanations of heat-related mortality; Hu et al. (2018) predict mortality in critically ill influenza patients, and interpret it with SHAP.

In this work we assess performance of different DL models, namely LSTM, CNN and CNN-LSTM, in the task of forecasting IMOEX, a Russian benchmark stock market index. We propose the use of a number of macroeconomic variables, that are theoretically connected to stock markets, and assess their contribution to model performance. We fail to come up with a good forecasting model; however, we support the hypothesis of macroeconomic variables being useful in financial market prediction. Moreover, we gain some insights on Russian stock market with SHAP model interpretation, such as the importance of Gold as predictor of IMOEX in August to November 2019, as well as relative unimportance of Oil and Gas.

The rest of the paper is organized as follows. In the next section, we review data, propose variable transformations, then explain modelling technique. The following Results section is divided into 2 parts: in first, we report model training and evaluation results and measure effect of variables on forecasting quality; the second part is dedicated to SHAP feature attributions and their interpretation. The final section concludes.

2. Data and Methodology

Our modeling task is multivariate rolling-window one-step-ahead regression forecasting, i.e. predicting one future value based on historical series of predefined fixed length. The target variable of the study is Moscow Exchange Index (IMOEX), Russian benchmark capitalization-weighted composite index, calculated on 50 best-performing Russian issuers of most important sectors of Russian economy. As a benchmark index, it represents, broadly, the aggregated dynamics of Russian stock market, which makes forecasting and analyzing it especially interesting.

We use daily data on IMOEX as well as 6 groups of explanatory variables to perform the forecasts. The variables are:

- Moscow Exchange sectoral indexes, namely MOEXOG, MOEXEU, MOEXTL, MOEXMM, MOEXFN, MOEXC, MOEXCH
- Shares of top IMOEX constitutes by capitalization
- Euro and USD to Ruble exchange rates
- Commodity prices of Oil, Natural Gas, and Gold

- Foreign market indexes, London exchange index (FTSE), SnP-500 and NASDAQ index
- Yields of Russian government bonds, 1 week to 20 years maturity, as a proxy for interest rate

For each variable with standard OHLCV set, we create 4 features generally representing market dynamics, liquidity and volatility:

- Logarithmic Return

$$r_t = \log P_t^C - \log P_{t-1}^C \tag{1}$$

- HighLow, relative difference between High and Low daily prices

$$HighLow_t = \frac{P_t^H - P_t^L}{P_t^L} \tag{2}$$

- Volatility, daily realized volatility calculated on 5-minute interval data

$$RV_t = \sqrt{\sum_{i=1}^{T} r_i^2} \tag{3}$$

- Logarithmic Volume

$$LogVolume_t = \log\left(Volume_t + 1\right) \tag{4}$$

The chosen period is from 2018-07-04 to 2019-11-01. This specific period is relatively stable and homogeneous, without evident structural breaks or extreme news that would disrupt model training. Forecasting extreme events, though being a thriving research stream, is outside the scope of this study.

The data is reshaped to match usual input shape of Time Series DL models, which is a tensor [B, L, N], where B is batch size, L is sequence length and N is a number of variables. Therefore, one input of a model is a tensor of stacked matrices of the following form:

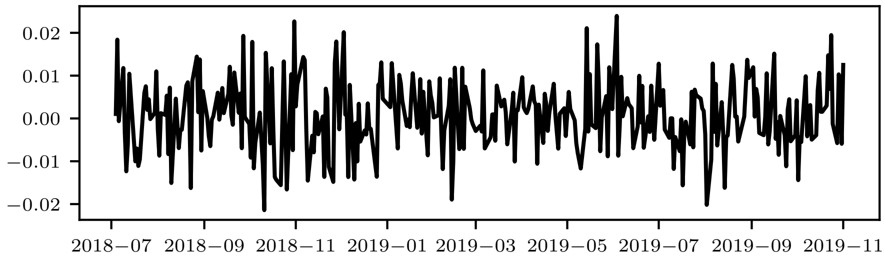

Figure 1: IMOEX Return

$$input = \begin{bmatrix} y_t & x_t^{(1)} & \dots & x_t^{(n)} \\ y_{t-1} & x_{t-1}^{(1)} & \dots & x_{t-1}^{(n)} \\ y_{t-2} & x_{t-2}^{(1)} & \dots & x_{t-2}^{(n)} \\ \vdots & \vdots & \ddots & \vdots \\ y_{t-T} & x_{t-T}^{(1)} & \dots & x_{t-T}^{(n)} \end{bmatrix} \tag{5}$$

The length of historic sequence considered at each step is a hyperparameter, and is chosen arbitrarily as 10, or 2 complete trading weeks.

Before feeding into a model, the data is normalized, i.e. each variable is demeaned and divided by its standard deviation. The scaling parameters are calculated only on training subset and applied to validation and test periods, to avoid information leakage.

Mean Squared Error is used as both Loss Function for training models and the main evaluation metric.

$$MSE = \frac{1}{N} \sum (y_t - \hat{y}_t)^2 \tag{6}$$

We employ Walk-Forward Cross Validation, a common validation scheme in time series forecasting, though underutilized in machine learning environment. The data is sequentially split into 12 folds, one for each subsequent week of testing data. Each fold consists of 265 training instances, followed by validation and test periods, 5 days (a trading week) each. For each fold, the training

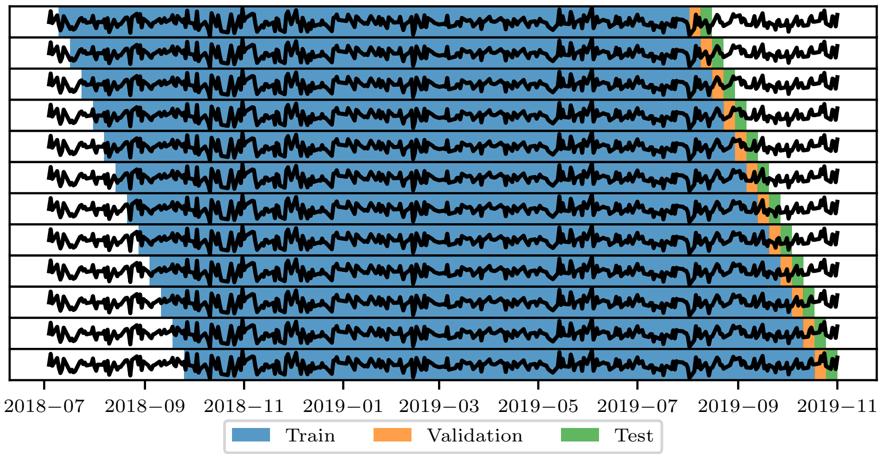

Figure 2: Walk-Forward Cross-Validation

period is used to train models, validation period is used to evaluate the set of hyperparameters and compare models of same architecture, and test period is used for final evaluation and comparison between different models.

We consider three model architectures and one naïve predictor:

- Naïve Zero

  A baseline predictor, always predicting 0 as each day's return. Since all data is normalized, it corresponds to AR(0), estimated on training period.

- LSTM

  A vanilla LSTM model without any enhancements. Two hyperparameters are tuned for this model: number of layers (1, 2 or 4) and layer hidden size (16, 32 or 64). The final prediction is made by one fully-connected layer.

- CNN

  Time series Convolutional Neural Network. Following X and Y, we construct it stacking 2 1D Convolution layers with ReLU activation functions, and Max Pooling layer, followed by another Convolution and Max Pooling pair. The hyperparameters for the model are filter sizes of Convolution layers (32-64-128, 64-128-256, 128-256-256, 256-512-1024), kernel sizes are fixed to 2.

- CNN-LSTM

  A hybrid architecture: CNN pre-processes inputs, extracting a new set of features, which are sequentially processed by LSTM. CNN part consists of 2 Convolution layers with ReLU activation functions, followed a Max Pooling layer. The hyperparameters are filter sizes (64-128, 128-256, 256-512, 512-1024), LSTM layers (1, 2, 4) and hidden size (16, 32 or 64).

All models are trained using Adaptive Momentum Stochastic Gradient Decent optimization algorithm (ADAM) with Learning Rate of 1e-3. All models are regularized by 10% Dropout and 1e-3 Weight Decay. Each of model specifications is trained 10 times, each time form a different weight initialization, to achieve better convergence. Best version of each model is chosen on each of validation periods.

After the models are trained, best sets of hyperparameters are selected for each of architectures based on validation metrics. To evaluate impacts of different variable groups on model performance, we exclude each of them, train the model again, and then compare the results.

Then, both global and local analysis of full models is performed using Gradient SHAP method (Lundberg and Lee, 2017). SHAP explains each of model's outputs by providing a set of additive attributions for each of the model's input features. Summed together, they reconstruct model's output for this particular instance. Resulting attributions roughly represent how the model made its decision, making its insight human-readable and interpretable. In our 3-dimensional case, attribution for each test value follows input's shape, or the following form.

$$attributions = \begin{bmatrix} \phi_t & \phi_t^{(1)} & \dots & \phi_t^{(n)} \\ \phi_{t-1} & \phi_{t-1}^{(1)} & \dots & \phi_{t-1}^{(n)} \\ \phi_{t-2} & \phi_{t-2}^{(1)} & \dots & \phi_{t-2}^{(n)} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{t-T} & \phi_{t-T}^{(1)} & \dots & \phi_{t-T}^{(n)} \end{bmatrix} \tag{7}$$

$$\hat{y}_{t+1} = \sum_{v=0}^{n} \sum_{l=-T}^{0} \phi_l^{(v)} \tag{8}$$

We aggregate absolute values of these attributions by different dimensions to get interpretations of relative importance across time lags, variables, variable groups, etc.

3. Empirical Results

3.1. Model Performance

Results of model evaluation are summarized in Table 1. As follows from comparison between the average MSE results and Zero Naïve predictor, we are unable to forecast Russian financial market any better than always predicting mean return over the previous year. Despite grid search over model hyperparameters, the best resulting models, CNN-LSTM(256-512, 32, 2) and CNN-LSTM(128-256-512) are only marginally close to the naïve case in terms of error. We can conclusively say, however, that this these architectures outperform basic LSTM.

All models exhibit very unstable training, with validation outcome solely depending on weight initialization. For example, loss figures for period 1, for best CNN-LSTM model, look totally different under different initial states Figure 7. This can indicate that the performance issue may be not attributed to model design, but rather to training procedure: optimizer, starting from different initial states, arrives at totally different local minimas, all fitting the training data perfectly, yet unable to generalize to validation and test samples. We believe, however, that with sufficiently larger set of hyperparameters to try, specifically

Table 1: Average Test MSE

Best Model	MSE
Naive Zero	**0.8793**
CNN(128-256-512)	**0.873**
Sectors	1.0061
Shares	0.951
Currencies	0.9916
Commodities	0.9182
Foreign indexes	0.9642
Bonds	1.0135
LSTM(16, 1)	**1.0643**
Sectors	1.3561
Shares	1.4998
Currencies	1.4968
Commodities	1.6165
Foreign indexes	1.7486
Bonds	1.1959
CNN-LSTM(256-512, 32, 2)	**0.8834**
Sectors	0.9428
Shares	1.1223
Currencies	1.1447
Commodities	1.045
Foreign indexes	0.9582
Bonds	1.0581

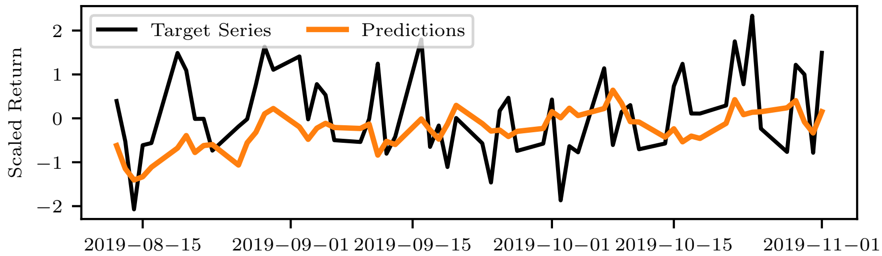

Figure 3: Best model predictions

tinkering over the training procedure, and perhaps employing more data, stable results can be achieved in principle.

Analyzing the sets of used variables, we can conclude that, despite not being able to achieve results in general, the proposed macroeconomic features improve model performance. In Table 1 italicized entries represent models trained on the set of features *excluding* a particular group. Compared to the case with all variables included, reduced models all yield conclusively worse results.

3.2. Feature attribution

We proceed to explanation of model predictions using SHAP. For this section, only the best-performing CNN-LSTM model is considered.

As SHAP is able to explain each of predictions locally, this methodology is especially helpful for evaluating models in production. For instance, we can look at one of the days in the test period, 12-08-2019 (Figure 10). This waterfall plot is interpreted as follows. The variables are sorted by their absolute Shapley values. At this specific day, all Gold variables were the most important in explaining model's prediction, pushing in into negatives; so did Novatek's variables, Euro exchange rates, MOEX sectoral index for finance, and others, with less absolute values. At the same time, changes in bond yields, index' past dynamics and a number of stocks drove expected value upwards.

By averaging absolute values of local attributions over all test instances, we obtain global attributions, depicting relative importance of each feature for

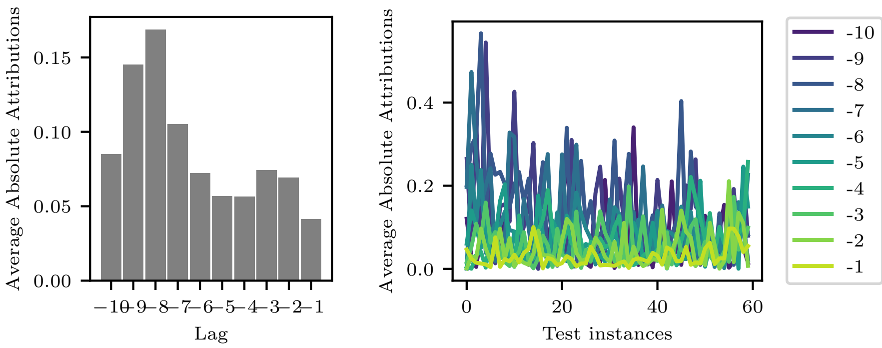

Figure 4: Lag importance

model's outputs across test period. As the input is a 3-dimensional tensor in our case, aggregations across different dimensions are analyzed.

Figure 4 represents absolute values of Shapely explanations, averaged across all dimensions but the lag one. The graph shows that immediate lags are less important, than about 7-9 days into the past - this advocates for the existence of long-term tendencies, rather than short-term reactions to some news. The figure on the right shows that, attributions vary from day to day in the test sequence, and overall dynamic changes across time.

Figure 5 presents SHAP variable attributions, averaged across Return-Volatility-HighLow-Volume sets for different instruments (detailed picture over all 101 features can be found in Appendix, Figure 11). Interestingly enough, the first most important feature is Gold, and, in particular, its relative High-Low indicator. Generally, it does not contradict econometric empirical literature, as well as common sense; however, there are no studies on Russian stock market, which use Gold as explanatory variable, to support or oppose this result.

Financial news of that period help reconstruct the story a bit better. 2019 was a splendid year for gold prices in general, and, in particular, so was September (Sieroń, 2020; Koh and Bafes, 2019). On September 3, Gold prices reached their (at that time) all-time peak of \$1546, having grown by 21.6% in the last three months. Such rise is explained by several factors, including expectations

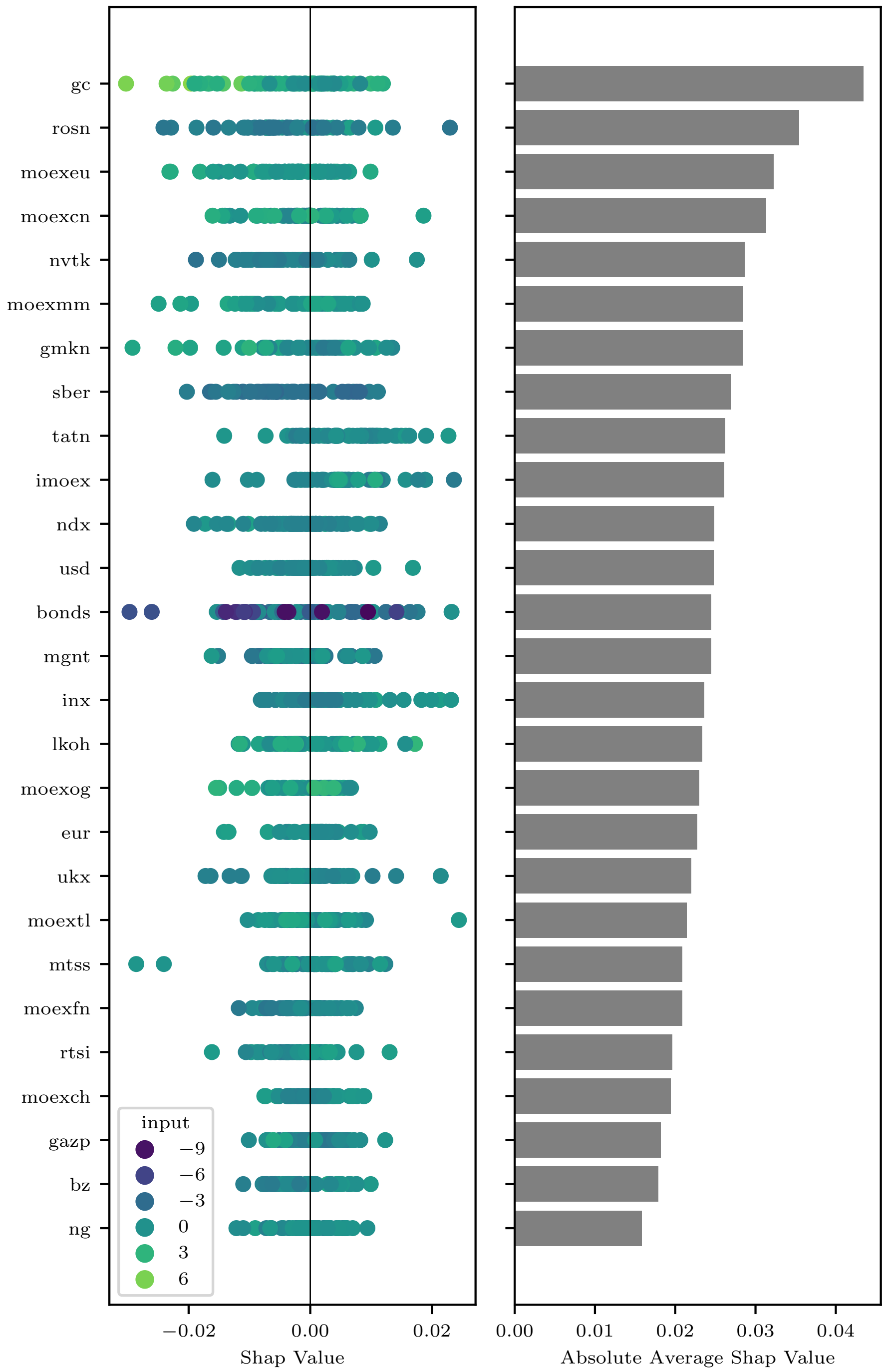

Figure 5: Variable Attributions

of reduction in the U.S. federal funds rate and rising demand by Central Banks. Moreover, the time just before the global pandemic, Spring and Summer of 2019 were marked by pres. Trump's trade war with China, further increasing demand for hedging opportunities like Gold.

Our testing sample starts on Monday, 12th of August and ends on 1st of November, which corresponds exactly with that period. Therefore, the link between gold price and Russian stock market extracted by SHAP can be explained by Russian investors using gold as a hedge against increasing geopolitical risks; or, perhaps, gold fluctuations in this period just happened correlate with the general news sentiment. The presented feature attribution is by no means causal.

Another interesting observation is relative unimportance of the other considered commodities, Oil and Gas. Both of them ranked last among all other variables, despite both major Russian oil-producing companies' stocks ranking higher. While being somewhat surprising on the first glance, this result corresponds to empirical literature. Both works of Korhonen and Peresetsky (2016) and Robert D. Gay (2008) do not find oil prices driving Russian stock market as well for earlier periods, advocating for the argument for Russian economy being mature and diversified enough to not be affected by Oil prices.

Finally, it has to be noted, that those interpretations should be taken with a grain of salt. After all, as the forecasts themselves are not very accurate, so might be the attributions.

4. Conclusion

Deep learning methods are powerful ML methods applied in many prediction tasks, including one of financial time series forecasting. Despite the predictive power of DL and its growing popularity, studies including fundamental macroeconomic factors are rarely conducted. While economists reserve to simpler linear methods, ML specialists tend to favor technical indicators above macroeconomic variables. We tried to integrate the two approaches.

In this work we assess, whether including information on Exchange rates, Commodities, Sectors, Bonds Foreign markets and underlying Stocks can increase quality of predictions of IMOEX, a primary Russian stock market index. We considered three architectures: LSTM, CNN and CNN-LSTM. We did not come up with a sufficiently good forecasting model; best hyperparameter specifications ended up only marginally different from Zero Naive prediction in terms of MSE on test period, unable to forecast index returns reliably across different periods and different training runs. However, we still found out that including all of the considered variables is better for forecasts that leaving them out.

In addition to forecast quality analysis, we interpreted model predictions with a novel feature attribution method. We were able to extract a non-obvious relationship between IMOEX and gold prices in August-November of 2019, as well as other interpretable conclusions.

The main limitation, as well as a path for future research, is, of course, the models' predicting power. If the models can not forecast properly, then results regarding everything else have much less basis under them. However, we still have confidence in the fact that constructing and successfully training DL models is possible in proposed setup.

References

, a. Finam.ru - financial portal: economy and stock market news, forecasts and analysis. investment company finam. URL: https://www.finam.ru/.

, b. Investing.com - Stock Market Quotes & Financial News. URL: https://www.investing.com/.

Al-Ameer, M., Hammad, W., Ismail, A., Hamdan, A., 2018. The Relationship of Gold Price with the Stock Market: The Case of Frankfurt Stock Exchange. International Journal of Energy Economics and Policy 8, 357–371.

Arfaoui, M., Ben Rejeb, A., 2017. Oil, gold, US dollar and stock market interdependencies: a global analytical insight. European Journal of Management and Business Economics 26, 278–293. doi:10.1108/EJMBE-10-2017-016.

Basher, S.A., Sadorsky, P., 2006. Oil price risk and emerging stock markets. Global Finance Journal 17, 224–251. doi:10.1016/j.gfj.2006.04.001.

Baur, D.G., 2012. Asymmetric Volatility in the Gold Market. The Journal of Alternative Investments 14, 26–38. doi:10.3905/jai.2012.14.4.026.

Baur, D.G., Lucey, B.M., 2010. Is Gold a Hedge or a Safe Haven? An Analysis of Stocks, Bonds and Gold. Financial Review 45, 217–229. doi:10.1111/j.1540-6288.2010.00244.x.

Beirne, J., Caporale, G.M., Schulze-Ghattas, M., Spagnolo, N., 2010. Global and regional spillovers in emerging stock markets: A multivariate GARCH-in-mean analysis. Emerging Markets Review 11, 250–260. doi:10.1016/j.ememar.2010.05.002.

Campbell, J.Y., 1987. Stock returns and the term structure. Journal of Financial Economics 18, 373–399. doi:10.1016/0304-405X(87)90045-6.

Chen, S., Ge, L., 2019. Exploring the attention mechanism in LSTM-based Hong Kong stock price movement prediction. Quantitative Finance 19, 1507–1515. doi:10.1080/14697688.2019.1622287.

Chen, T., Xu, J., Ying, H., Chen, X., Feng, R., Fang, X., Gao, H., Wu, J., 2019. Prediction of Extubation Failure for Intensive Care Unit Patients Using Light Gradient Boosting Machine. IEEE Access 7, 150960–150968. doi:10.1109/ACCESS.2019.2946980. conference Name: IEEE Access.

Chen, Y.C., Huang, W.C., 2021. Constructing a stock-price forecast CNN model with gold and crude oil indicators. Applied Soft Computing 112, 107760. doi:10.1016/j.asoc.2021.107760.

Choudhry, T., Hassan, S.S., Shabi, S., 2015. Relationship between gold and stock markets during the global financial crisis: Evidence from nonlinear causality tests. International Review of Financial Analysis 41, 247–256. doi:10.1016/j.irfa.2015.03.011.

Contuk, F.Y., Burucu, H., Güngör, B., 2013. Effect of Gold Price Volatility on Stock Returns: example of Turkey. undefined .

Degiannakis, S., Filis, G., Arora, V., 2018. Oil Prices and Stock Markets: A Review of the Theory and Empirical Evidence. The Energy Journal 39. doi:10.5547/01956574.39.5.sdeg.

Dornbusch, R., Fischer, S., 1980. Exchange Rates and the Current Account. American Economic Review 70, 960–71.

Eitel, F., Schulz, M.A., Seiler, M., Walter, H., Ritter, K., 2021. Promises and pitfalls of deep neural networks in neuroimaging-based psychiatric research. Experimental Neurology 339, 113608. doi:10.1016/j.expneurol.2021.113608.

Fama, E.F., 1970. Efficient Capital Markets: A Review of Theory and Empirical Work. The Journal of Finance 25, 383–417. doi:10.2307/2325486.

Fama, E.F., 1981. Stock Returns, Real Activity, Inflation, and Money. The American Economic Review 71, 545–565.

Feng, G., He, J., Polson, N.G., 2018. Deep Learning for Predicting Asset Returns. Technical Report arXiv:1804.09314. arXiv. doi:10.48550/arXiv.1804.09314. arXiv:1804.09314 [cs, econ, stat] type: article.

Filis, G., Degiannakis, S., Floros, C., 2011. Dynamic correlation between stock market and oil prices: The case of oil-importing and oil-exporting countries. International Review of Financial Analysis 20, 152–164. doi:10.1016/j.irfa.2011.02.014.

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

Gao, T., Chai, Y., 2018. Improving Stock Closing Price Prediction Using Recurrent Neural Network and Technical Indicators. Neural Computation 30, 2833–2854. doi:10.1162/neco_a_01124.

Gokmenoglu, K.K., Fazlollahi, N., 2015. The Interactions among Gold, Oil, and Stock Market: Evidence from S&P500. Procedia Economics and Finance 25, 478–488. doi:10.1016/S2212-5671(15)00760-1.

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, 2222–2232. doi:10.1109/TNNLS.2016.2582924. conference Name: IEEE Transactions on Neural Networks and Learning Systems.

Gunduz, H., Yaslan, Y., Cataltepe, Z., 2017. Intraday prediction of Borsa Istanbul using convolutional neural networks and feature correlations. Knowledge-Based Systems 137, 138–148. doi:10.1016/j.knosys.2017.09.023.

Harris, C.R., Millman, K.J., van der Walt, S.J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N.J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M.H., Brett, M., Haldane, A., del Río, J.F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., Oliphant, T.E., 2020. Array programming with NumPy. Nature 585, 357–362. doi:10.1038/s41586-020-2649-2.

Hoseinzade, E., Haratizadeh, S., 2018. CNNPred: CNN-based stock market prediction using several data sources. Technical Report arXiv:1810.08923. arXiv. doi:10.48550/arXiv.1810.08923. arXiv:1810.08923 [cs, q-fin, stat] type: article.

Hu, H., Tang, L., Zhang, S., Wang, H., 2018. Predicting the direction of stock markets using optimized neural networks with Google Trends. Neurocomputing 285, 188–195. doi:10.1016/j.neucom.2018.01.038.

Huang, J.Y., Liu, J.H., 2020. Using social media mining technology to improve stock price forecast accuracy. Journal of Forecasting 39, 104–116. doi:10.1002/for.2616.

Hunter, J.D., 2007. Matplotlib: A 2d graphics environment. Computing in Science & Engineering 9, 90–95. doi:10.1109/MCSE.2007.55.

Istiake Sunny, M.A., Maswood, M.M.S., Alharbi, A.G., 2020. Deep Learning-Based Stock Price Prediction Using LSTM and Bi-Directional LSTM Model , 87–92doi:10.1109/NILES50944.2020.9257950.

Kim, S.W., Rogers, J.H., 1995. International stock price spillovers and market liberalization: Evidence from Korea, Japan, and the United States. Journal of Empirical Finance 2, 117–133. doi:10.1016/0927-5398(94)00013-7.

Kim, Y., Kim, Y., 2022. Explainable heat-related mortality with random forest and SHapley Additive exPlanations (SHAP) models. Sustainable Cities and Society 79, 103677. doi:10.1016/j.scs.2022.103677.

Koh, W.C., Bafes, J., 2019. Gold prices to shine in 2019 on persistent headwinds to global growth. URL: https://blogs.worldbank.org/developmenttalk/gold-prices-shine-2019-persistent-headwinds-global-growth.

Kokhlikyan, N., Miglani, V., Martin, M., Wang, E., Alsallakh, B., Reynolds, J., Melnikov, A., Kliushkina, N., Araya, C., Yan, S., Reblitz-Richardson, O., 2020. Captum: A unified and generic model interpretability library for PyTorch. arXiv:2009.07896 [cs, stat] ArXiv: 2009.07896.

Kordonis, J., Symeonidis, S., Arampatzis, A., 2016. Stock Price Forecasting via Sentiment Analysis on Twitter, in: Proceedings of the 20th Pan-Hellenic Conference on Informatics, Association for Computing Machinery, New York, NY, USA. pp. 1–6. doi:10.1145/3003733.3003787.

Korhonen, I., Peresetsky, A., 2016. What Influences Stock Market Behavior in Russia and Other Emerging Countries? Emerging Markets Finance and Trade 52, 1210–1225. doi:10.1080/1540496X.2015.1037200.

Li, Q., Tan, J., Wang, J., Chen, H., 2021. A Multimodal Event-Driven LSTM Model for Stock Prediction Using Online News. IEEE Transactions on Knowledge and Data Engineering 33, 3323–3337. doi:10.1109/TKDE.2020.2968894. conference Name: IEEE Transactions on Knowledge and Data Engineering.

Liang, X., Ge, Z., Sun, L., He, M., Chen, H., 2019. LSTM with Wavelet Transform Based Data Preprocessing for Stock Price Prediction. Mathematical Problems in Engineering 2019, e1340174. doi:10.1155/2019/1340174.

Lim, B., Arik, S.O., Loeff, N., Pfister, T., 2021. Temporal Fusion Transformers for interpretable multi-horizon time series forecasting. International Journal of Forecasting 37, 1748–1764. doi:10.1016/j.ijforecast.2021.03.012.

Livieris, I.E., Pintelas, E., Pintelas, P., 2020. A CNN-LSTM model for gold price time-series forecasting. Neural Computing and Applications 32, 17351–17360. doi:10.1007/s00521-020-04867-x.

Lozinskaia, A., Saltykova, A., 2019. Fundamental Factors Affecting The Moex Russia Index: Structural Break Detection In A Long-Term Time Series. Technical Report WP BRP 77/FE/2019. National Research University Higher School of Economics. Publication Title: HSE Working papers.

Lu, W., Li, J., Li, Y., Sun, A., Wang, J., 2020. A CNN-LSTM-Based Model to Forecast Stock Prices. Complexity 2020, e6622927. doi:10.1155/2020/6622927.

Lundberg, S.M., Lee, S.I., 2017. A Unified Approach to Interpreting Model Predictions, in: Advances in Neural Information Processing Systems, Curran Associates, Inc.

Ma, Y., Han, R., Fu, X., 2019. Stock prediction based on random forest and LSTM neural network, in: 2019 19th International Conference on Control, Automation and Systems (ICCAS), pp. 126–130. doi:10.23919/ICCAS47443.2019.8971687. iSSN: 2642-3901.

Makridakis, S., Spiliotis, E., Assimakopoulos, V., 2020. The M4 Competition: 100,000 time series and 61 forecasting methods. International Journal of Forecasting 36, 54–74. doi:10.1016/j.ijforecast.2019.04.014.

Matsunaga, D., Suzumura, T., Takahashi, T., 2019. Exploring Graph Neural Networks for Stock Market Predictions with Rolling Window Analysis. Technical Report arXiv:1909.10660. arXiv. doi:10.48550/arXiv.1909.10660. arXiv:1909.10660 [cs, q-fin] type: article.

McKinney, W., et al., 2010. Data structures for statistical computing in python 445, 51–56.

Mehtab, S., Sen, J., 2021. Stock Price Prediction Using Convolutional Neural Networks on a Multivariate Timeseries. Technical Report. doi:10.36227/techrxiv.15088734.v1. arXiv:2001.09769 [cs, q-fin].

Modigliani, F., Cohn, R.A., 1979. Inflation, Rational Valuation and the Market. Financial Analysts Journal 35, 24–44. doi:10.2469/faj.v35.n2.24.

Nandha, M., Faff, R., 2008. Does oil move equity prices? A global view. Energy Economics 30, 986–997. doi:10.1016/j.eneco.2007.09.003.

Nelson, D.M.Q., Pereira, A.C.M., de Oliveira, R.A., 2017. Stock market's price movement prediction with LSTM neural networks, in: 2017 International Joint Conference on Neural Networks (IJCNN), pp. 1419–1426. doi:10.1109/IJCNN.2017.7966019. iSSN: 2161-4407.

Oriani, F.B., Coelho, G.P., 2016. Evaluating the impact of technical indicators on stock forecasting, in: 2016 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 1–8. doi:10.1109/SSCI.2016.7850017.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S., 2019. PyTorch: An Imperative Style, High-Performance

Deep Learning Library, in: Wallach, H., Larochelle, H., Beygelzimer, A., Alché-Buc, F.d., Fox, E., Garnett, R. (Eds.), Advances in Neural Information Processing Systems 32. Curran Associates, Inc., pp. 8024–8035.

Pereira, S., Meier, R., Alves, V., Reyes, M., Silva, C.A., 2018. Automatic brain tumor grading from MRI data using convolutional neural networks and quality assessment. volume 11038. doi:10.1007/978-3-030-02628-8. arXiv:1809.09468 [cs].

Rapach, D.E., Wohar, M.E., Rangvid, J., 2005. Macro variables and international stock return predictability. International Journal of Forecasting 21, 137–166. doi:10.1016/j.ijforecast.2004.05.004.

Robert D. Gay, J., 2008. Effect Of Macroeconomic Variables On Stock Market Returns For Four Emerging Economies: Brazil, Russia, India, And China. International Business & Economics Research Journal (IBER) 7. doi:10.19030/iber.v7i3.3229.

Schwert, G.W., 2003. Chapter 15 Anomalies and market efficiency, in: Handbook of the Economics of Finance. Elsevier. volume 1 of *Financial Markets and Asset Pricing*, pp. 939–974. doi:10.1016/S1574-0102(03)01024-0.

Sezer, O.B., Gudelek, M.U., Ozbayoglu, A.M., 2019. Financial Time Series Forecasting with Deep Learning : A Systematic Literature Review: 2005-2019. arXiv:1911.13288 [cs, q-fin, stat] ArXiv: 1911.13288.

Sieroń, A., 2020. The Gold Market in 2019. URL: https://finance.yahoo.com/news/gold-market-2019-091252036.html.

Singh, A., Sengupta, S., Lakshminarayanan, V., 2020. Explainable Deep Learning Models in Medical Image Analysis. Journal of Imaging 6, 52. doi:10.3390/jimaging6060052.

Singh, R., Srivastava, S., 2017. Stock prediction using deep learning. Multimedia Tools and Applications 76, 18569–18584. doi:10.1007/s11042-016-4159-7.

Song, Y., Lee, J.W., Lee, J., 2019. A study on novel filtering and relationship between input-features and target-vectors in a deep learning model for stock price prediction. Applied Intelligence 49, 897–911. doi:10.1007/s10489-018-1308-x.

Van Rossum, G., Drake, F.L., 2009. Python 3 Reference Manual. CreateSpace, Scotts Valley, CA.

Vargas, M.R., dos Anjos, C.E.M., Bichara, G.L.G., Evsukoff, A.G., 2018. Deep Leaming for Stock Market Prediction Using Technical Indicators and Financial News Articles, in: 2018 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. doi:10.1109/IJCNN.2018.8489208. iSSN: 2161-4407.

Waskom, M.L., 2021. Seaborn: Statistical sata visualization. Journal of Open Source Software 6, 3021. doi:10.21105/joss.03021.

Yan, H., Ouyang, H., 2018. Financial Time Series Prediction Based on Deep Learning. Wireless Personal Communications 102, 683–700. doi:10.1007/s11277-017-5086-2.

Zhang, X., Tan, Y., 2018. Deep Stock Ranker: A LSTM Neural Network Model for Stock Selection, in: Tan, Y., Shi, Y., Tang, Q. (Eds.), Data Mining and Big Data, Springer International Publishing, Cham. pp. 614–623. doi:10.1007/978-3-319-93803-5_58.

5. Appendix

5.1. Used Packages & Data

All empirical work is done in Python (Van Rossum and Drake, 2009). Pandas and NumPy are used for data processing (McKinney et al., 2010; Harris et al., 2020). Neural networks are implemented in PyTorch (Paszke et al., 2019). SHAP values are calculated with Captum (Kokhlikyan et al., 2020). Figures are drawn by Matplotlib and Seaborn (Hunter, 2007; Waskom, 2021).

All the data is openly available at finam.ru and investing.com (noa, b,a).

5.2. Training procedure

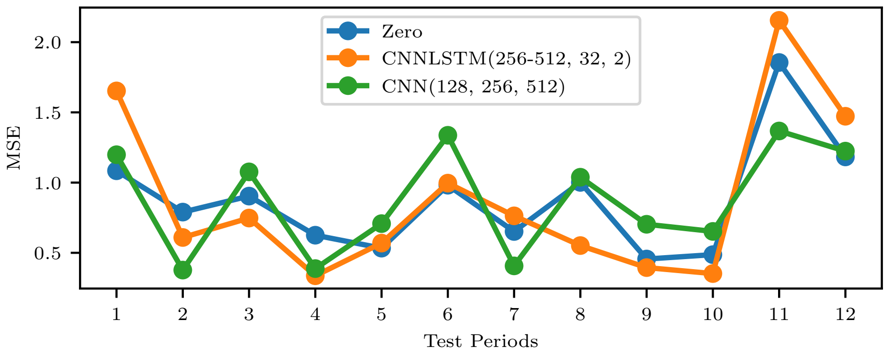

Figure 6: MSE on test periods

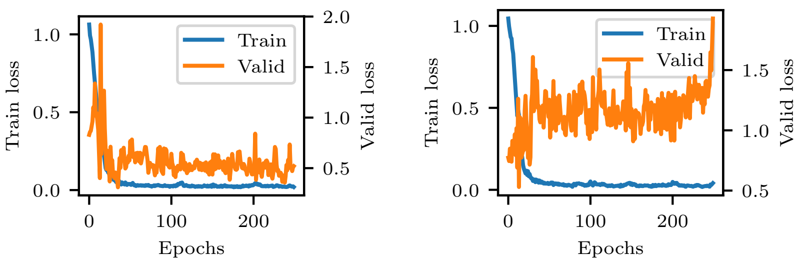

Figure 7: Loss plots for 2 sets of initial weights

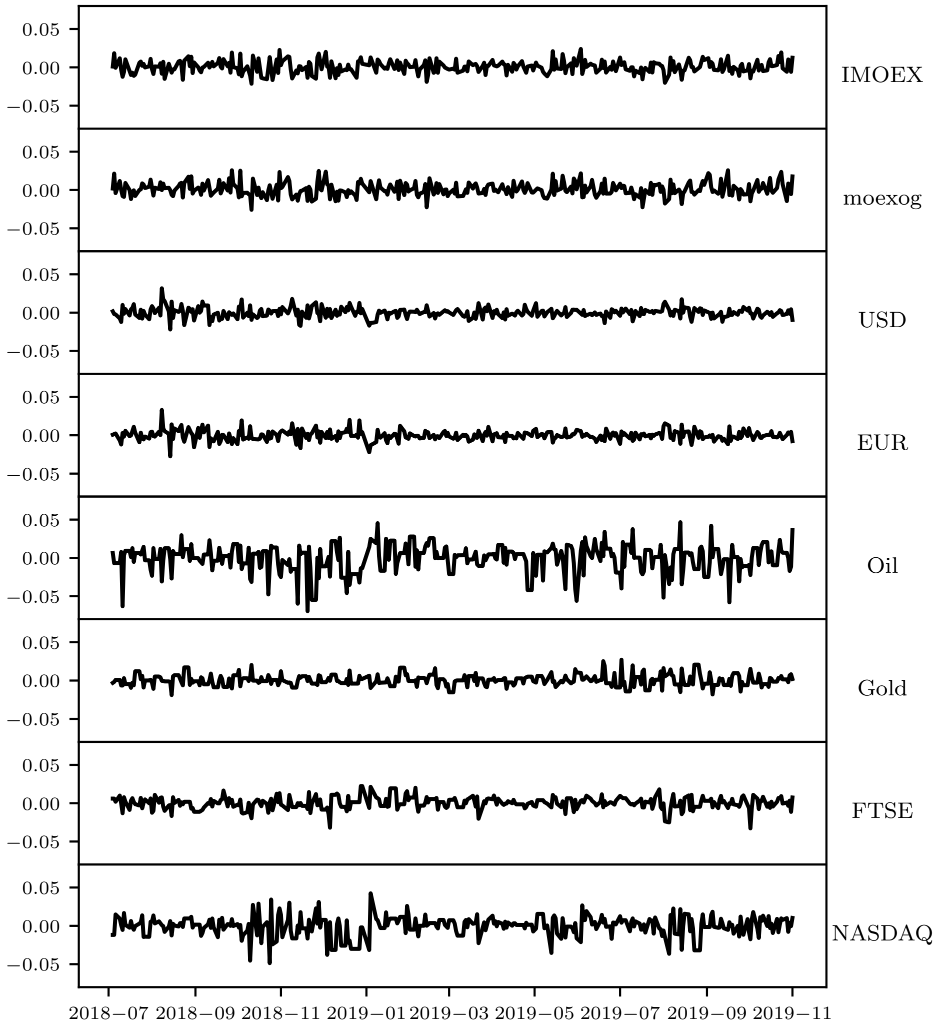

Figure 8: Selected Returns across considered period

5.4. SHAP values

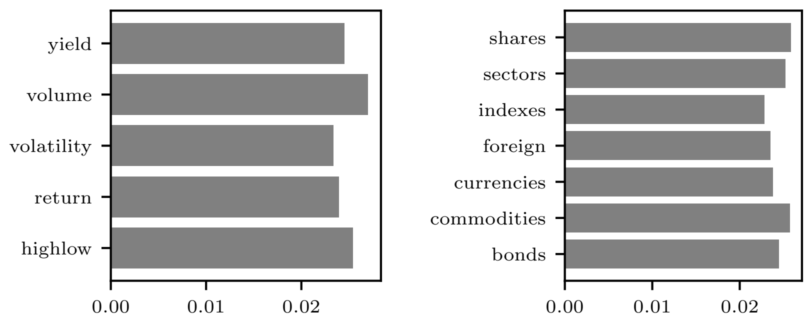

Figure 9: Average Abs SHAP values over variable types

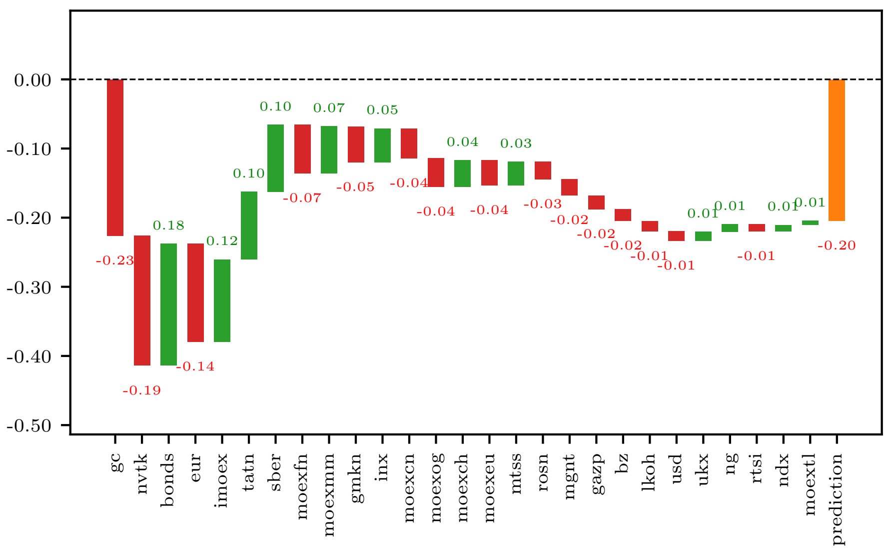

Figure 10: Local explanations for 12-06-2019

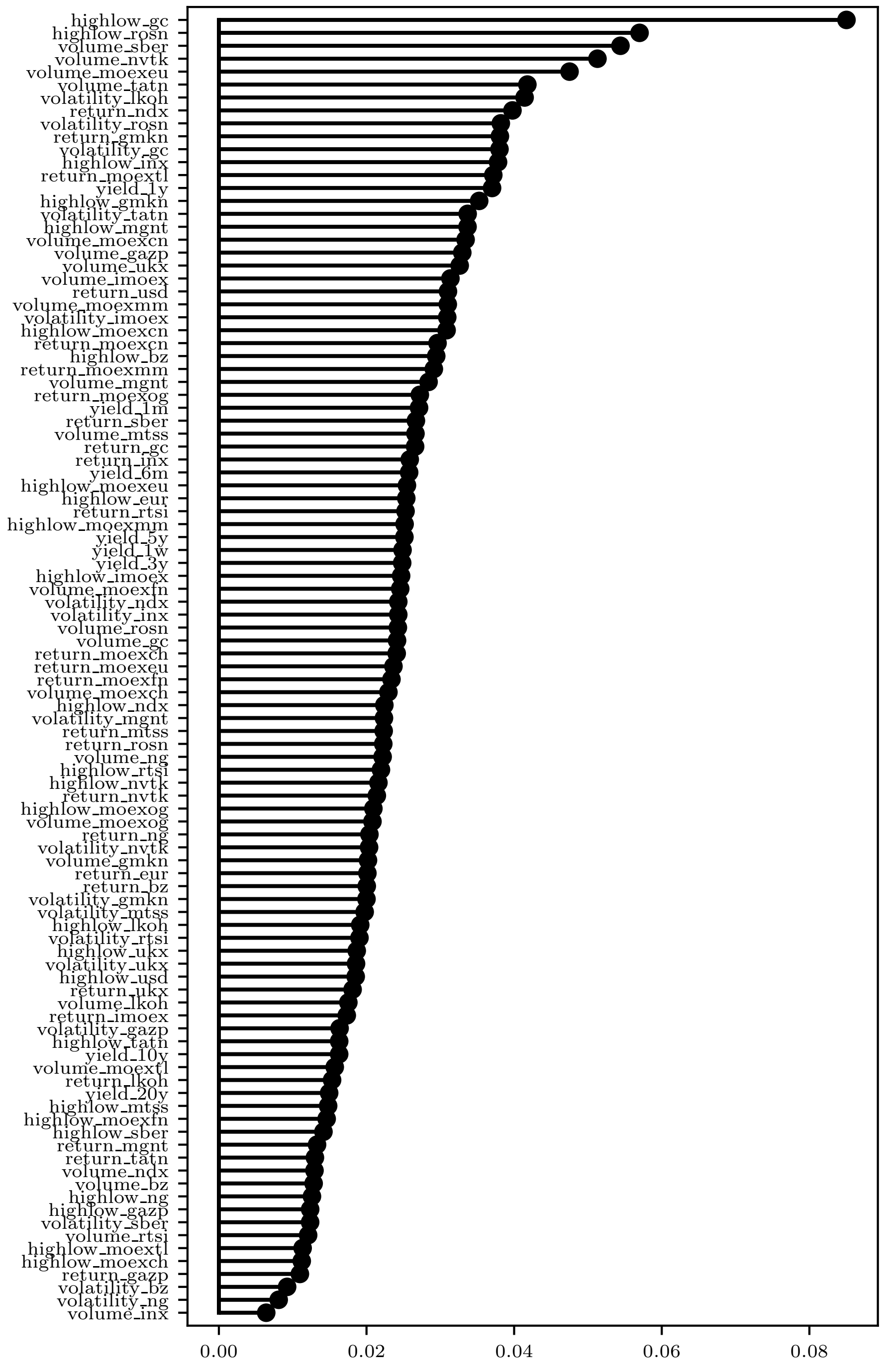

Figure 11: Variable attributions