The comparison of LSTM in algorithmic investment strategies on BTCUSD and S&P500 index on different frequencies

Jakub Michańków, Paweł Sakowski and Robert Ślepaczuk

QFRG and DSLab Monthly Seminar

Cracow University of Economics
Quantitative Finance Research Group, Faculty of Economic Sciences, University of Warsaw

21st March 2022

Agenda

- Motivation, hypotheses and research questions
- Literature review
- Methodology
  - Terminology and Metrics
  - LSTM model
    - Architecture of LSTM model
    - Main features of LSTM model
    - Data preprocessing for LSTM model
    - Hyperparameters tuning
  - Most common drawbacks in papers analyzing algorithmic investment
  - Research Description
    - Loss function strategies (AIS)

Agenda II

- Data description
- Performance metrics
- Results
  - Base models
    - BTCUSD
    - S&P500 index
  - Sensitivity analysis for 1h data
    - Dropout
    - Sequence length
    - TrainSet length
    - Batch Size
  - Combined model on various frequencies and various assets
- Research Hypotheses Verification
- Research extensions

Motivation I

The main aims:

- exploration of **deep learning possibilities** in algorithmic investment strategies (AIS),
- analysis of **signals from LSTM** model in algorithmic investment strategies on **different frequencies** and **different asset classes** (equity indices and cryptocurrencies),
- designing the **proper architecture** (initial hyperparameters tuning) of the LSTM model and **testing the performance** of AIS with comparison to the traditional Buy&Hold model (B&H).

Motivation II

- None of the previous works covered the topic of **the performance of signals** from the **LSTM** model in AIS with simultaneous focus on **architecture** of LSTM tested on **various frequencies** and **various asset classes** with **rolling window** approach enhanced with additional **sensitivity analysis** at the end.
- Although each year researchers publish thousands of papers devoted to testing numerous alternative approaches employed in AIS, the results of these studies include numerous **drawbacks** and **mistakes** which in practice makes it **impossible** to use their findings in **real trading.**
- Therefore, the chase for the **efficient algorithmic investment strategies** still continues.

Hypotheses and research questions

First Hypothesis:

The signals from LSTM model employed in AIS are more efficient than Buy&Hold approach regardless of asset class tested.

Second Hypothesis:

The signals from LSTM model employed in AIS are more efficient than Buy&Hold approach regardless of data frequency tested.

Third Hypothesis:

The signals from LSTM model employed in AIS are more efficient in case of BTCUSD than in case of S&P500 index.

Fourth Hypothesis:

The robustness of tested models to various hyperparameters does not depend on asset class tested.

Software, Libraries, Hardware, Time

- The results for LSTM model were obtained using R 4.1.0 along with Python 3.7.10
- Deep learning libraries used for design, training and testing the network are Keras 2.4.0 and TensorFlow 2.5.0.
- The rest of the calculations, as well as graphs and tables were done using only R and RStudio environment.
- Computer specification: AMD Ryzen 7 3700X 3,6GHz, 16GB RAM, NVIDIA GeForce RTX 2060 Super with 270 tensor cores.
- One full training (number of rolling windows $\times$ 40 epochs) lasted around:
  - 20 minutes on 1d frequency,
  - 60 minutes on 1h frequency,
  - 180 minutes on 15m frequency for S&P500 data,
  - 80/240/720 minutes for BTC data

Literature review

- **Hochreiter & Schmidhuber (1997)** - the first introduction of LSTM. By introducing Constant Error Carousel (CEC) units, LSTM deals with the exploding and vanishing gradient problems. The initial version of the LSTM block included cells, input, and output gates.
- **Gers & Schmidhuber (1999)** introduced the forget gate (also called "keep gate") into LSTM architecture, enabling the LSTM to reset its own state.
- **Gers et al. (2000)** added peephole connections (connections from the cell to the gates) into the architecture. Additionally, the output activation function was omitted
- **Chung et al. (2014)** put forward a simplified variant called *Gated Recurrent Unit* (GRU).
- **Chen et al. (2015)** implemented LSTM model to predict the next day returns for China stocks.
- **Zhang et al. (2019)** presented AT-LSTM model which is combination of LSTM and Attention based model and provided results for three index datasets: Russell 2000, DJIA and NASDAQ.
- **Kijewski and Ślepaczuk (2020)** compared the performance of classical techniques with LSTM model for S&P500 index on daily frequency for the last 20 years and showed that LSTM model results are not robust to initial hyperparameters assumptions.

Terminology and Metrics

Methodology. Terminology and Metrics

- Main model used in this work is based on Long Short-Term Memory network which is a type deep neural networks (DNN).
- Custom loss function was created as the network performance metric and used during the training process.
- Strategy performance metrics - equity line and strategy specific performance metrics.
- Sensitivity analysis to show how changes in network hyperparameters and architecture affect the results.
- Combined strategies (frequencies and assets).

LSTM model

Methodology. LSTM model

LSTM networks are a type or recurrent neural networks (RNN) that can keep track of long term dependencies in data, allowing to partially solve vanishing gradient problem typical for classic RNNs. It's widely used to model sequential data such as text, speech and time series. LSTM units are composed of memory cells, with each cell having three types of gates (input gate, output gate and forget gate). These gates use *tanh* and *sigmoid* functions to regulate the flow of information through the cell, deciding how much and which information should be stored in long term state, passed on to another step, or discarded.

Architecture of LSTM model

Methodology. LSTM model. Architecture of LSTM model

- Our model consists of three LSTM layers with 512/256/128 neurons and one single neuron dense layer on the output. Each of LSTM layers is using *tanh* activation function (to retain negative values). L2 regularization (0.0005) and dropout (0.02) are also applied to each of these layers. The first two layers return sequences with the same shape as the input sequence (full sequence), the last LSTM layer returns only the last output.
- To train the model we used Adam optimizer - a stochastic gradient descent optimizer with momentum (estimating first-order and second-order moments). The learning rate of the optimizer was set to 0.0015 (after tuning).

Data preprocessing of LSTM model

Methodology. Artificial Neural Network. Data preprocessing of LSTM model

- Logarithmic returns, based on one minute data for both BTC and S&P500, from 2013-04-01 to 2020-12-31
- For training set we used around 1000 observations (1371 for BTC and 948 for S&P500, after tuning). Validation set was set size to 33% of the training set. Test sets (and also rolling window) size was 90 for BTC and 65 for S&P500 (after turning).
- Input sequence size for LSTM network was set to 20 for BTC and 14 for S&P500, batch size was set to 80.
- The output of the model was a single number predicting the next return value.
- Based on the sign of the predicted return value we assigned -1, 0, 1 signals.

Hyperpaprameters tuning

Methodology. Artificial Neural Network. Hyperparameters tuning

During our research we conducted detailed hyperparameters tuning to ensure the best possible results from our model. The hyperparameters we tested were:

- number of layers (1-5) and neurons in each layer (5-512)
- dropout rate (0.001 - 0.2) and l2 kernel regularization (0.0001 - 0.01)
- type of optimizer (SGD, RMSProp and Adam variants), learning rate (0.001 - 0.1) and momentum values (0.1-0.9)
- training and testing window sizes, sequence length and batch size
- number of epochs (10-300) and callbacks (early stopping and model checkpoint)

Methodology. Artificial Neural Network. Hyperparameters tuning

Table 1. Values of hyperparameters selected after network tuning.

Hyperparameter	Selected Value
No. hidden layers	3
No. neurons	512/256/128
Activation function	tanh
Dropout rate	0.02
l2 regularizer	0.0005
Optimizer	Adam
Learning rate	0.00015
BTC train/test	1371/90
S&P train/test	948/65
Batch size	80
Sequence length	14/20

Source: Own study.

Training process of LSTM model

Methodology. Artificial Neural Network. Training process of LSTM model

- For training and prediction we used a walk forward validation/rolling window approach. Model was trained on around three years of data (equal to train set length) and then used to for predictions over the next 3 months (equal to test set length). After that the window was moved by three months ahead and the model was retrained. A single return value was predicted each time, based on the last 14/20 (sequence length) values.
- A single iteration was trained for 40 epochs. Model checkpoint callback function was used to store the best weights (parameters) of the model based on the lowest loss function value in specific epoch. There weights were then used for prediction.

Most common drawbacks in papers analysing AIS

Methodology.Most common drawbacks in papers analysing AIS

- only one in-sample and one out-of-sample period -> the results are heavily dependent on the selected period,
- tests of AIS are performed on only one basis instrument
- over-optimization,
- not proper Loss Function,
- forward looking bias in Buy/Sell signals,
- no any sensitivity analysis referring initially set parameters of the model,
- data snooping bias,
- survivorship bias,

Research Description

Methodology. Research Description

- Hyperparameters tuning,
- Buy/Sell signals definitions based on the next day forecasts,
- Tests for two types of strategies: Long/Short and Long only,
- New Loss function: MADE,
- Walk-forward optimization,
- Equity lines and performance metrics according to Ślepaczuk et al. (2018),
- Sensitivity analysis for various values of Dropout, Sequence length, TrainSet length, Batch size,
- The combination of signals across different frequencies (1d, 1h and 15m) and asset classes (equity - S&P500 index and cryptocurrency - BTCUSD)

Methodology. Research Description. Loss function

- We introduce our authorship loss function which better stress usefulness of forecasting ability of LSTM model in algorithmic investment strategies (AIS).
- RMSE, MSE, MAE, MAPE, %OP used in 99.9% of similar research are not proper error function for the evaluation of the forecasting ability of the given model **in AIS.**
- above mentioned error metrics evaluate the accuracy of forecasts which is often confused with the forecasting ability of the given model in AIS,

$$\mathrm{MADL} = \frac{1}{N}\sum_{i=1}^{N}(-1) \times \mathrm{sign}(R_i \times \hat{R}_i) \times \mathrm{abs}(R_i) \tag{1}$$

where: MADL is the Mean Absolute Directional Loss, R_i is the observed return on interval i, $\hat{R}_i$ is the predicted return on interval i, $\mathrm{sign}(X)$ is the function which gives the sign of X, $\mathrm{abs}(X)$ is the function which gives the absolute value of X and N is the number of forecasts.

Methodology. Research Description. Loss function

- This way, the value the function returns will be equal to the observed return on investment with the predicted direction, which allows the model to tell if the prediction will yield profit or loss and how much this profit or loss will be.
- MADL was designed specifically for working with AIS's.
- The function in our model is minimized, so that if it returns the negative values the strategy will make a profit, and if it returns a positive value the strategy will generate a loss.

Data description

Data description. Data description

- 1m data for S&P500 index and BTCUSD converted to 15m, 1h and 1d frequency,
- The time period for estimation and results presentation: 2013-04-01 - 2021-12-31
- The time period for results presentation: 2017-01-01 - 2021-12-31
- The hours of trading for:
  - S&P500 index: 3.30pm CET - 10.00pm CET, from Monday to Friday excluding official holidays,
  - BTCUSD: 24h per day, 7 days per week,

Performance metrics

Performance metrics

- aRC
- aSD
- MD
- MLD
- IR*
- IR**
- IR***; we regard this metric as the most important
- nObs
- nTrades

Base model

Results. MSE vs. MADL

Table 3. Performance metrics for investment strategies on BTC and S&P500 in the base case scenario with MSE and MADL loss functions.

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
panel A: BTC 1 d									
BTC	132.41	78.98	83.23	2.96	1.68	2.67	1.19	1461	NA
MSE Long/Short	182.18	78.89	83.23	2.08	2.31	5.05	4.42	1461	4
MADL Long/Short	190.47	78.88	48.59	0.85	2.41	9.47	21.23	1461	156
MSE Long only	159.07	77.46	83.23	2.85	2.05	3.92	2.19	1461	2
MADL Long only	200.49	57.83	50.79	1.40	3.47	13.69	19.56	1461	78
panel B: S&P500 1 d									
S&P500	13.39	20.51	33.92	0.58	0.65	0.26	0.06	1005	NA
MSE Long/Short	0.44	20.53	28.61	2.04	0.02	0.00	0.00	1005	1014
MADL Long/Short	6.13	20.48	25.09	2.56	0.30	0.07	0.00	1005	168
MSE only	7.90	14.18	16.31	1.47	0.56	0.27	0.01	1005	507
MADL Long only	11.20	12.35	13.08	1.74	0.91	0.78	0.05	1005	84

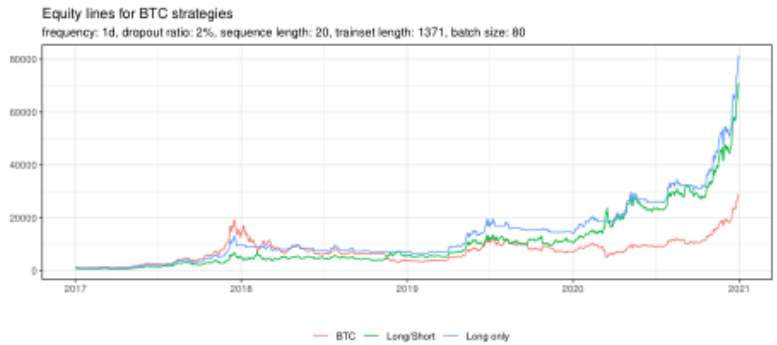
Note: BTC and S&P500 stand for the benchmark strategies, i.e., Buy&Hold applied for BTC and S&P500 prices, respectively. MSE Long/Short and MADL Long/Short stand for the investment strategy with long and short signals from models optimized with MSE and MADL loss functions, respectively. MSE Long only and MADL Long only stand for the investment strategy with long only signals from models optimized with MSE and MADL loss function. The table presents the results in the period between 1 January 2017 and 31 December 2020 for daily frequency. The hyperparameters of LSTM model for the the base case scenario were set as it was described in Table 1.

Results. Base model.BTCUSD

Results. Base model

BTCUSD

Results. BTCUSD. 1d. Long/Short and Long only



	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
BTC	132.41	78.98	83.23	2.96	1.68	2.67	1.19	1461	NA
Long/Short	190.47	78.88	48.59	0.85	2.41	9.47	21.23	1461	156
Long only	200.49	57.83	50.79	1.40	3.47	13.69	19.56	1461	78

Results. BTCUSD. 1h. Long/Short and Long only

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
BTC	134.66	87.01	83.94	2.96	1.55	2.48	1.13	35063	NA
Long/Short	90.10	87.01	84.39	2.02	1.04	1.11	0.49	35063	6616
Long only	155.40	61.40	57.14	1.50	2.53	6.88	7.12	35063	3310

Results. BTCUSD. 15m. Long/Short and Long only

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
BTC	134.93	89.35	84.01	2.96	1.51	2.43	1.11	140255	NA
Long/Short	-44.04	89.35	96.51	2.13	-0.49	-0.22	-0.05	140255	27100
Long only	40.39	62.73	73.88	2.69	0.64	0.35	0.05	140255	13550

Results. BTCUSD

BTCUSD Model Results

- 1d: the best results are for Long/Short (L/S) and Long only (LO) strategy which at the same time increased return and decrease risk metrics,
- 1h: LO much more efficient than B&H and LS,
- 15m: LO nor LS can not beat B&H,
- higher frequencies produce worse results
- LO is generally better than LS

Results. Base model. S&P500

Results. Base model

S&P500 index

Results. S&P500 index. 1d. Long/Short and Long only

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
S&P500	15.34	19.93	33.92	0.58	0.77	0.35	0.09	1119	NA
Long/Short	4.34	19.95	19.66	1.66	0.22	0.05	0.00	1119	224
Long only	11.10	12.07	10.16	1.20	0.92	1.00	0.09	1119	112

Results. S&P500 index. 1h. Long/Short and Long only

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
S&P500	12.50	18.09	34.77	0.64	0.69	0.25	0.05	7041	NA
Long/Short	-2.69	18.09	48.04	1.51	-0.15	-0.01	0.00	7041	1554
Long only	5.54	12.39	19.50	1.51	0.45	0.13	0.00	7041	777

Results. S&P500 index. 15m. Long/Short and Long only

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
S&P500	13.56	19.08	35.34	0.59	0.71	0.27	0.06	26155	NA
Long/Short	15.63	19.08	16.70	1.19	0.82	0.77	0.10	26155	6444
Long only	15.74	12.87	8.25	0.97	1.22	2.33	0.38	26155	3222

Results. S&P500 index.

S&P500 index. Model Results

- 1d: the best results for B&H and LO, and much worse for LS
- 1h: B&H -> LO -> LS
- 15m: LO -> LS -> B&H
- the best results for 15m, then for 1d and lastly for 1h,
- LO is much better than LS,

Results. Regressions for returns.

Table 5. Results of regressions for returns: Long/Short and Long only strategies on BTC and S&P500 for three different frequencies

	Alpha	Std. Err.	t	pv	Beta	Std. Err.	t	pv
panel A: BTC 1 d vs.								
Long/Short	0.0028	0.0011	2.5805	0.0100 **	0.0613	0.0260	2.3596	0.0184 *
Long only	0.0018	0.0005	3.3295	0.0009 ***	0.5258	0.0130	40.3095	0.0000 ***
panel B: BTC 1 h vs.								
Long/Short	0.0000	0.0000	0.6506	0.5153	0.0401	0.0053	7.5261	0.0000 ***
Long only	0.0000	0.0000	1.4969	0.1344	0.5190	0.0027	194.5037	0.0000 ***
panel C: BTC 15 min vs.								
Long/Short	0.0000	0.0000	−1.2728	0.2031	−0.0126	0.0027	−4.7162	0.0000 ***
Long only	0.0000	0.0000	−0.3720	0.7099	0.4944	0.0013	370.3576	0.0000 ***
panel D: S&P500 1 d vs.								
Long/Short	0.0004	0.0004	0.9501	0.3423	−0.2705	0.0302	−8.9672	0.0000 ***
Long only	0.0002	0.0002	1.2270	0.2201	0.3598	0.0152	23.7180	0.0000 ***
panel E: S&P500 1 h vs.								
Long/Short	0.0000	0.0001	−0.2379	0.8120	−0.0551	0.0119	−4.6340	0.0000 ***
Long only	0.0000	0.0000	−0.0377	0.9699	0.4720	0.0060	79.3050	0.0000 ***
panel F: S&P500 15 min vs.								
Long/Short	0.0000	0.0000	1.6450	0.1000 *	−0.0874	0.0061	−14.2609	0.0000 ***
Long only	0.0000	0.0000	1.8525	0.0640 *	0.4537	0.0031	147.3734	0.0000 ***

Note: BTC and S&P500 stand for the benchmark strategies, i.e., Buy&Hold applied for BTC and S&P500 prices, respectively. Long/Short stands for the investment strategy with long and short signals. Long only stands for the investment strategy with long only signals. The table presents the results of regressions in the form of: $R_t = \alpha + \beta R_t^* + \varepsilon_t$, where R_t is the return for tested strategy in period t and R_t^* is the return in of BTC or S&P500 strategies. Regressions were calculated in the period between 1 January 2017 and 31 December 2020. The hyperparameters of LSTM model for the the base case scenario were set as it was described in Table 1. Asterisks *, ** and *** denote statistical significance at the 10%, 1% and 0.1%, respectively.

Sensitivity analysis for 1h data

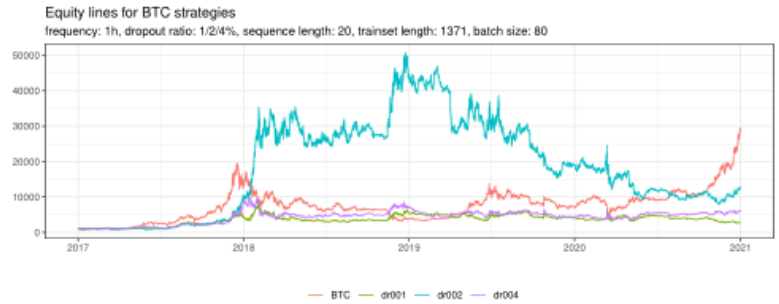
Dropout

Results. Sensitivity analysis for 1h data. Dropout

Results. Sensitivity analysis for 1h data. Dropout

BTCUSD. Long/Short

Results. Sensitivity analysis for 1h data. Dropout. BTCUSD. Long/Short



	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
BTC	134.66	87.01	83.94	2.96	1.55	2.48	1.13	35063	NA
dr001	29.13	87.01	71.82	2.91	0.33	0.14	0.01	35063	6434
dr002	90.10	87.01	84.39	2.02	1.04	1.11	0.49	35063	6616
dr004	58.26	86.97	62.96	2.96	0.67	0.62	0.12	35063	6534

Results. Sensitivity analysis for 1h data. Dropout

BTCUSD. Long Only

Results. Sensitivity analysis for 1h data. Dropout. BTCUSD. Long Only

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
BTC	134.66	87.01	83.94	2.96	1.55	2.48	1.13	35063	NA
dr001	111.01	61.28	51.13	1.41	1.81	3.93	3.10	35063	3219
dr002	155.40	61.40	57.14	1.50	2.53	6.88	7.12	35063	3310
dr004	134.11	62.38	56.15	2.41	2.15	5.13	2.86	35063	3263

Results. Sensitivity analysis for 1h data. Dropout

Results. Sensitivity analysis for 1h data. Dropout

S&P500 index. Long/Short

Results. Sensitivity analysis for 1h data. Dropout. S&P500 index. Long/Short

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
S&P500	12.50	18.09	34.77	0.64	0.69	0.25	0.05	7041	NA
dr001	-0.45	18.09	27.54	1.76	-0.02	0.00	0.00	7041	1574
dr002	-2.69	18.09	48.04	1.51	-0.15	-0.01	0.00	7041	1554
dr004	-1.43	18.09	45.17	1.51	-0.08	0.00	0.00	7041	1614

Results. Sensitivity analysis for 1h data. Dropout

S&P500 index. Long Only

Results. Sensitivity analysis for 1h data. Dropout. S&P500 index. Long Only

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
S&P500	12.50	18.09	34.77	0.64	0.69	0.25	0.05	7041	NA
dr001	6.78	12.19	14.10	0.96	0.56	0.27	0.02	7041	787
dr002	5.54	12.39	19.50	1.51	0.45	0.13	0.00	7041	777
dr004	6.32	11.65	26.61	1.03	0.54	0.13	0.01	7041	807

Results. Sensitivity analysis for 1h data. Dropout

BTCUSD. Long/Short

- the most efficient dropout was 2%, i.e. the one selected during hyperparameters tuning
- the results of the model are rather robust to slight changes in dropout rate,

BTCUSD. Long only

- the most efficient dropout was 2%, i.e. the one selected during hyperparameters tuning
- the results of the model are rather robust to slight changes in dropout rate,

S&P500 index. Long/Short

- the most efficient dropout was 1%, while 2% selected during hyperparameters tuning was the least efficient,
- the results of the model are quite robust to slight changes in dropout rate,

S&P500 index. Long only

- the most efficient dropout was 1%, but 2% selected during hyperparameters tuning gives almost the same results,
- the results of the model are quite robust to slight changes in dropout rate,

Sequence length

Results. Sensitivity analysis for 1h data. Sequence length

Results. Sensitivity analysis for 1h data. Sequence length

BTCUSD. Long/Short

Results. Sensitivity analysis for 1h data. Sequence length. BTCUSD. Long/Short

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
BTC	134.66	87.01	83.94	2.96	1.55	2.48	1.13	35063	NA
seq10	-60.61	87.01	98.67	3.94	-0.70	-0.43	-0.07	35063	12206
seq20	90.10	87.01	84.39	2.02	1.04	1.11	0.49	35063	6616
seq40	19.49	87.02	97.77	2.93	0.22	0.04	0.00	35063	4616

Results. Sensitivity analysis for 1h data. Sequence length

BTCUSD. Long Only

Results. Sensitivity analysis for 1h data. Sequence length. BTCUSD. Long Only

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
BTC	134.66	87.01	83.94	2.96	1.55	2.48	1.13	35063	NA
seq10	15.32	63.02	76.53	2.87	0.24	0.05	0.00	35063	6106
seq20	155.40	61.40	57.14	1.50	2.53	6.88	7.12	35063	3310
seq40	105.60	59.14	83.54	2.93	1.79	2.26	0.81	35063	2308

Results. Sensitivity analysis for 1h data. Sequence length

Results. Sensitivity analysis for 1h data. Sequence length

S&P500 index. Long/Short

Results. Sensitivity analysis for 1h data. Sequence length. S&P500 index. Long/Short

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
S&P500	12.50	18.09	34.77	0.64	0.69	0.25	0.05	7041	NA
seq07	6.02	18.09	20.34	1.97	0.33	0.10	0.00	7041	2922
seq14	-2.69	18.09	48.04	1.51	-0.15	-0.01	0.00	7041	1554
seq28	2.65	18.09	34.83	0.86	0.15	0.01	0.00	7041	912

Results. Sensitivity analysis for 1h data. Sequence length

S&P500 index. Long Only

Results. Sensitivity analysis for 1h data. Sequence length. S&P500 index. Long Only

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
S&P500	12.50	18.09	34.77	0.64	0.69	0.25	0.05	7041	NA
seq07	10.14	12.57	12.65	0.91	0.81	0.65	0.07	7041	1461
seq14	5.54	12.39	19.50	1.51	0.45	0.13	0.00	7041	777
seq28	8.57	11.01	19.53	0.90	0.78	0.34	0.03	7041	456

Results. Sensitivity analysis for 1h data. Sequence length

BTCUSD. Long/Short

- the most efficient sequence length was 20, i.e. the one selected during hyperparameters tuning
- the results of the model are not robust to slight changes in sequence length,

BTCUSD. Long only

- the most efficient sequence length was 20, i.e. the one selected during hyperparameters tuning
- the results of the model are not robust to slight changes in sequence length,

S&P500 index. Long/Short

- the most efficient sequence length was 7, while 14 selected during hyperparameters tuning was the least efficient,
- the results of the model are not robust to slight changes in dropout rate,

S&P500 index. Long only

- the most efficient sequence length was 7, while 14 selected during hyperparameters tuning was the least efficient,
- the results of the model are quite robust to slight changes in dropout rate,

TrainSet length

Results. Sensitivity analysis for 1h data. TrainSet length

Results. Sensitivity analysis for 1h data. TrainSet length

BTCUSD. Long/Short

Results. Sensitivity analysis for 1h data. TrainSet length. BTCUSD. Long/Short

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
BTC	134.66	87.01	83.94	2.96	1.55	2.48	1.13	35063	NA
tr0685	-40.19	87.02	94.91	2.98	-0.46	-0.20	-0.03	35063	6370
tr1371	90.10	87.01	84.39	2.02	1.04	1.11	0.49	35063	6616
tr2742	8.35	87.00	88.78	1.46	0.10	0.01	0.00	35063	6178

Results. Sensitivity analysis for 1h data. TrainSet length

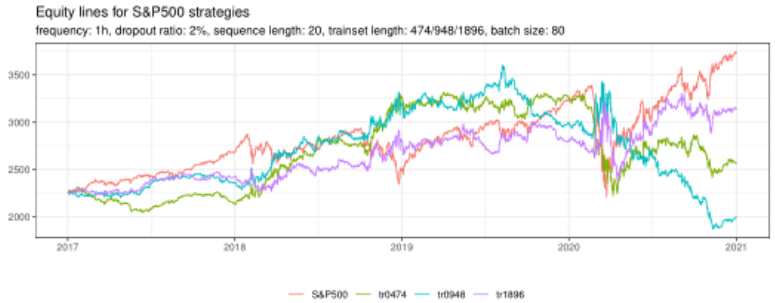
BTCUSD. Long Only

Results. Sensitivity analysis for 1h data. TrainSet length. BTCUSD. Long Only

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
BTC	134.66	87.01	83.94	2.96	1.55	2.48	1.13	35063	NA
tr0685	43.78	60.89	73.21	2.99	0.72	0.43	0.06	35063	3185
tr1371	155.40	61.40	57.14	1.50	2.53	6.88	7.12	35063	3310
tr2742	93.18	60.37	69.59	1.48	1.54	2.07	1.30	35063	3091

Results. Sensitivity analysis for 1h data. TrainSet length

S&P500 index. Long/Short



	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
S&P500	12.50	18.09	34.77	0.64	0.69	0.25	0.05	7041	NA
tr0474	3.03	18.09	32.91	1.89	0.17	0.02	0.00	7041	1516
tr0948	-2.69	18.09	48.04	1.51	-0.15	-0.01	0.00	7041	1554
tr1896	8.04	18.09	26.19	0.44	0.44	0.14	0.02	7041	1474

S&P500 index. Long Only

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
S&P500	12.50	18.09	34.77	0.64	0.69	0.25	0.05	7041	NA
tr0474	8.67	11.90	27.25	1.02	0.73	0.23	0.02	7041	758
tr0948	5.54	12.39	19.50	1.51	0.45	0.13	0.00	7041	777
tr1896	11.44	10.63	13.81	0.38	1.08	0.89	0.27	7041	737

Results. Sensitivity analysis for 1h data. TrainSet length

BTCUSD. Long/Short

- the most efficient TrainSet length was 1371, i.e. the one selected during hyperparameters tuning
- the results of the model are not robust to slight changes in sequence length,

BTCUSD. Long only

- the most efficient TrainSet length was 1371, i.e. the one selected during hyperparameters tuning
- the results of the model are not robust to slight changes in sequence length,

S&P500 index. Long/Short

- the most efficient TrainSet length was 1896, while 948 selected during hyperparameters tuning was the least efficient,
- the results of the model are not robust to slight changes in dropout rate,

S&P500 index. Long only

- the most efficient TrainSet length was 1896, while 948 selected during hyperparameters tuning was the least efficient,
- the results of the model are not robust to slight changes in dropout rate,

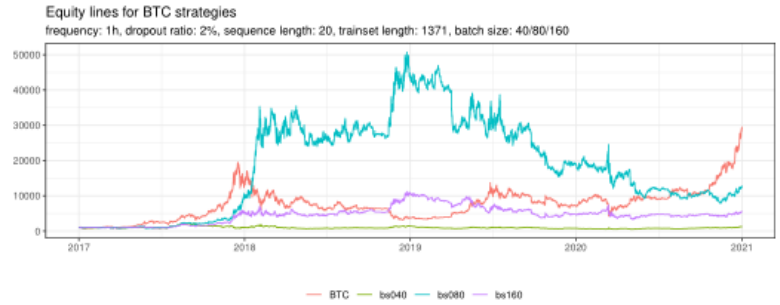
Batch Size

Results. Sensitivity analysis for 1h data. Batch Size

Results. Sensitivity analysis for 1h data. Batch Size

BTCUSD. Long/Short

Results. Sensitivity analysis for 1h data. Batch Size. BTCUSD. Long/Short



	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
BTC	134.66	87.01	83.94	2.96	1.55	2.48	1.13	35063	NA
bs040	6.67	87.02	76.87	3.33	0.08	0.01	0.00	35063	6160
bs080	90.10	87.01	84.39	2.02	1.04	1.11	0.49	35063	6616
bs160	54.61	87.02	71.63	2.02	0.63	0.48	0.13	35063	6436

Results. Sensitivity analysis for 1h data. Batch Size

BTCUSD. Long Only

Results. Sensitivity analysis for 1h data. Batch Size. BTCUSD. Long Only

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
BTC	134.66	87.01	83.94	2.96	1.55	2.48	1.13	35063	NA
bs040	95.33	57.81	50.23	1.35	1.65	3.13	2.21	35063	3081
bs080	155.40	61.40	57.14	1.50	2.53	6.88	7.12	35063	3310
bs160	128.70	62.61	46.02	1.37	2.06	5.75	5.41	35063	3218

Results. Sensitivity analysis for 1h data. Batch Size

Results. Sensitivity analysis for 1h data. Batch Size

S&P500 index. Long/Short

Results. Sensitivity analysis for 1h data. Batch Size. S&P500 index. Long/Short

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
S&P500	12.50	18.09	34.77	0.64	0.69	0.25	0.05	7041	NA
bs040	0.76	18.09	32.77	2.17	0.04	0.00	0.00	7041	1582
bs080	-2.69	18.09	48.04	1.51	-0.15	-0.01	0.00	7041	1554
bs160	-5.83	18.09	42.30	1.51	-0.32	-0.04	0.00	7041	1562

Results. Sensitivity analysis for 1h data. Batch Size

S&P500 index. Long Only

Results. Sensitivity analysis for 1h data. Batch Size. S&P500 index. Long Only

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
S&P500	12.50	18.09	34.77	0.64	0.69	0.25	0.05	7041	NA
bs040	7.53	10.77	12.64	1.20	0.70	0.42	0.03	7041	791
bs080	5.54	12.39	19.50	1.51	0.45	0.13	0.00	7041	777
bs160	3.95	11.43	21.45	0.90	0.35	0.06	0.00	7041	781

Results. Sensitivity analysis for 1h data. Batch Size

BTCUSD. Long/Short

- the most efficient Batch Size was 80, i.e. the one selected during hyperparameters tuning
- the results of the model are not robust to slight changes in sequence length,

BTCUSD. Long only

- the most efficient Batch Size was 80, i.e. the one selected during hyperparameters tuning
- the results of the model are quite robust to slight changes in sequence length,

S&P500 index. Long/Short

- the most efficient Batch Size was 40, while 80 selected during hyperparameters tuning was the least efficient,
- the results of the model are not robust to slight changes in dropout rate,

S&P500 index. Long only

- the most efficient Batch Size was 40, while 80 selected during hyperparameters tuning was the least efficient,
- the results of the model are not robust to slight changes in dropout rate,

HT correct for BTCUSD but should be improved for S&P500

Combined model on various frequencies and various assets

Results. Combined model on different frequencies and different assets

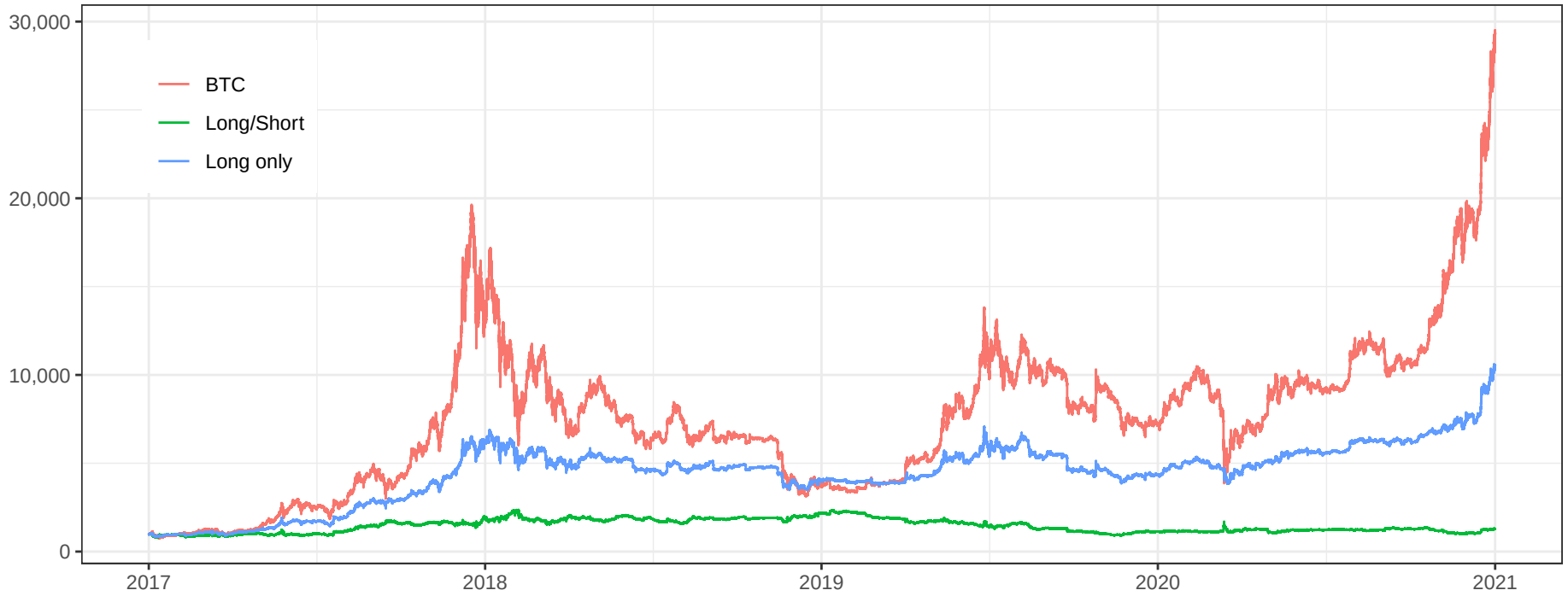
- in order to smooth our equity lines and use limited correlations between AIS on various frequencies and assets we decided to present Combined results

Assumptions

- two ways of combinations of signals across frequencies used: 1d, 1h and 15m:
  - approach #1: three signals {1, -1, 1} in the same interval are combined as {1/3}
  - approach #2: three signals {1, -1, 1} in the same interval are combined as {1}
- every other aspects of construction of Equity lines stays as it was before,

Results. Combined frequencies. BTC. approach #1

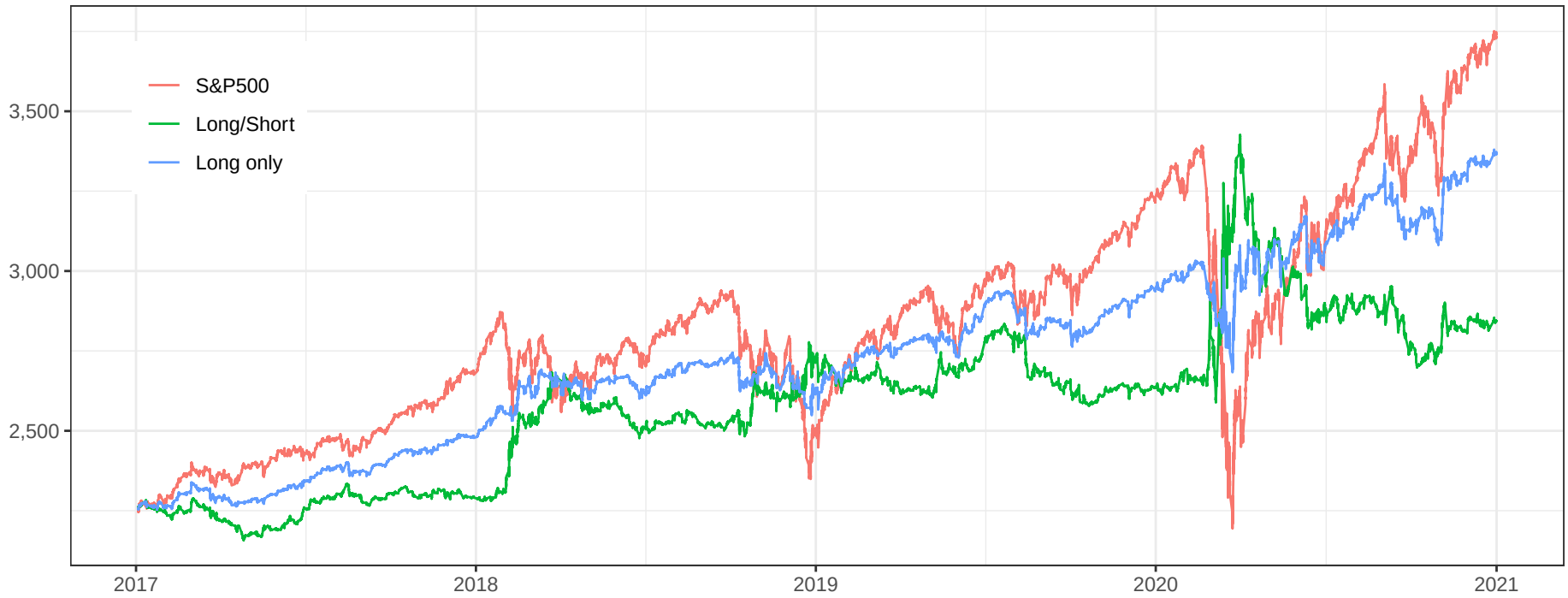
approach #1: three signals {1, -1, 1} in the same interval are combined as {1/3}



	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
BTC	134.93	89.35	84.01	2.96	1.51	2.43	1.11	140255	NA
Long/Short	7.70	54.02	62.07	2.90	0.14	0.02	0.00	140255	16902
Long only	81.89	52.65	49.55	1.47	1.56	2.57	1.43	140255	4048

Results. Combined frequencies. S&P500. approach #1

approach #1: three signals {1, -1, 1} in the same interval are combined as {1/3}



	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
S&P500	13.56	19.08	35.34	0.59	0.71	0.27	0.06	26155	NA
Long/Short	5.99	14.06	21.27	0.76	0.43	0.12	0.01	26155	4262
Long only	10.61	10.88	11.74	0.42	0.98	0.88	0.22	26155	1134

Results. Combined RB3M/6M & W10%90%/W20%80%

Results. Combined freq. & assets. RB3M & W10%90%

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
BTC/SP500	31.72	21.57	35.88	1.51	1.47	1.30	0.27	140255	NA
Long/Short	7.22	13.89	18.58	1.21	0.52	0.20	0.01	140255	NA
Long only	19.62	11.75	12.43	0.55	1.67	2.64	0.94	140255	NA

Results. Combined freq. & assets. RB3M & W20%80%

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
BTC/SP500	47.49	28.12	39.07	1.52	1.69	2.05	0.64	140255	NA
Long/Short	7.80	15.84	18.60	1.16	0.49	0.21	0.01	140255	NA
Long only	27.41	15.01	14.57	1.20	1.83	3.43	0.79	140255	NA

Results. Combined freq. & assets. RB6M & W10%90%

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
BTC/SP500	35.51	22.96	35.88	1.51	1.55	1.53	0.36	140255	NA
Long/Short	7.37	14.05	19.00	1.21	0.52	0.20	0.01	140255	NA
Long only	21.37	12.19	12.43	0.55	1.75	3.01	1.18	140255	NA

Results. Combined freq. & assets. RB6M & W20%80%

	aRC	aSD	MD	MLD	IR*	IR**	IR***	nObs	nTrades
BTC/SP500	53.76	30.17	41.24	1.52	1.78	2.32	0.82	140255	NA
Long/Short	8.05	16.21	17.70	1.16	0.50	0.23	0.02	140255	NA
Long only	30.27	15.85	14.07	1.20	1.91	4.11	1.04	140255	NA

Results. Combined model on different frequencies and different assets

Combined frequencies. BTC & S&P500 approach #1

- LO -> B&H -> LS
- lower volatility, more smooth equity lines,

Combined freq. & assets

- W20%80% always better than W10%90%,
- RB6m always better than RB3m,
- post factum combined results suggest rare rebalancing and higher weight of BTCUSD in our optimal portfolio,

Research Hypotheses Verification

First Hypothesis:

The signals from LSTM model employed in AIS are more efficient than Buy&Hold approach regardless of asset class tested. –> this hypothesis holds only for BTCUSD (1d_LS, 1d_LO, and 1h_LO) and S&P500 (1d_LO, 15m_LS, and 15m_LO) so we reject RH1

Second Hypothesis:

The signals from LSTM model employed in AIS are more efficient than Buy&Hold approach regardless of data frequency tested. –> the best results are for daily data in case of BTCUSD and 15m data in case of S&P500 so we reject RH2

Third Hypothesis:

The signals from LSTM model employed in AIS are more efficient in case of BTCUSD than in case of S&P500 index. –> for various frequencies we obtain different results so we reject RH3

Fourth Hypothesis:

The robustness of tested models to various hyperparameters does not depend on asset class tested. –> the results were not robust for BTCUSD nor for S&P500 but in significantly different way

Conclusions

- the efficiency of LSTM in AIS strictly depend on HT and the construction of the model and estimation process
- proper Loss Function is crucial in model estimation process
- the results are dependent on asset classes tested and frequencies used
- final results are not robust to initial assumptions

Research extensions

- more extensive SA
- various Loss Functions
- larger set of HF data possibilities
- to repeat the whole research with transaction costs included in the estimation process
- more careful hyperparameters tuning process, especially in case of S&P500

References I

- Cheng S. Huang C., Zhang R., Zhang W., 2018, Multi Factor Stock Selection Model Based on LSTM. International Journal of Economics and Finance, 10, 3pp. 6-42.
- Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555.
- Fischer, T., & Krauss, C., 2018, Deep learning with long short-term memory networks for financial market predictions. European Journal of Operational Research, 270(2), pp. 654–669.
- Gers, F. A., Schmidhuber, J., & Cummins, F. (2000). Learning to forget: Continual prediction with LSTM. Neural computation, 12(10), 2451-2471.
- Goodfellow I., Bengio Y., Courville A., 2016, Deep Learning, The MIT Press.
- Hochreiter S., Schmidhuber J., 1997, Long Short-term Memory, Neural Computation, vol. 9(8), pp. 1735-1780.
- Ślepaczuk R., Sakowski P., Zakrzewski G., 2018, Investment strategies beating the market. What can we squeeze from the market?, eFinanse Vol.14, no. 4, pp. 36-55, https://efinanse.com/current-issue/?number=59&id=421

Publication

Final publication

Michańków J., Sakowski P., Ślepaczuk R., 2022, The comparison of LSTM in algorithmic investment strategies on BTC and SP500 index, Sensors 22, 917, https://www.mdpi.com/1424-8220/22/3/917, https://doi.org/10.3390/s22030917

Thank you for your attention!

Jakub Michańków
jakub.michankow@phd.uek.krakow.pl
Cracow University of Economics
Doctoral School

Paweł Sakowski
sakowski@wne.uw.edu.pl
University of Warsaw, Faculty of Economic Sciences
Department of Quantitative Finance,
Quantitative Finance Research Group

Robert Ślepaczuk
rslepaczuk@wne.uw.edu.pl
University of Warsaw, Faculty of Economic Sciences
Department of Quantitative Finance
Quantitative Finance Research Group