école
normale
supérieure
paris—saclay

Developing a deep learning framework with two-stage feature selection for multivariate financial time series forecasting

Time series

Félix Fourreau, Mathis Wauquiez

M2 Mathématiques Vision Apprentissage

February 3, 2025

Objectives and work done

Context:

The paper tackles the task of financial time series forecasting. One of the main contributions of this paper is a wrapper-based feature selection method.

Objective:

Replication of the paper's results, and add improvements to the methodology.

Methodology Overview

- Two stage feature selection
- RNN model
- Error correction

First stage feature selection

RreliefF :

$$W_F = \frac{p_{\text{diffP}|\text{diffF}} \cdot p_{\text{diffF}}}{p_{\text{diffP}}} - \frac{\left(1 - p_{\text{diffP}|\text{diffF}}\right) \cdot p_{\text{diffF}}}{1 - p_{\text{diffP}}}$$

- $p_{\text{diffF}} = p\big(\text{diffF} \mid \text{NIs}\big)$: distance relative to nearest neighbor
- $p_{\text{diffP}} = p\big(\text{diffP} \mid \text{NIs}\big)$: Probability of differences in prediction P among nearest instances (NIs).
- $p_{\text{diffP}|\text{diffF}} = p\big(\text{diffP} \mid \text{diffF, NIs}\big)$: Conditional probability of differences in prediction P, given differences in feature F.

Second Stage Feature Selection: Grey Wolf Optimizer

Context: Multi-objective optimization for selecting relevant features in a multivariate time series.

Key Equations:

$$D = |C \cdot X_p(t) - X(t)|$$
$$X(t+1) = X_p(t) - A \cdot D$$
$$A = 2a \cdot r_1 - a$$

Variable Explanations:

- $X(t)$, $X_p(t)$: Position of the wolf and prey at iteration t
- A, C: Random coefficients (decreasing between 2 and 0)

Second Stage Feature Selection: Cuckoo Search

Concept: Uses random exploration to enhance the optimization process.

Integration with GWO: The implementation perturbs the positions of the leader wolves to maintain diversity and improve convergence.

$$X_k = X_{\alpha,\beta,\delta} - A_k \left| C_k \cdot X_{\alpha,\beta,\delta} - X \right|$$

$$X_k = \text{Cuckoo}\left(X_k, X_{\alpha,\beta,\delta}\right)$$

$$X(t+1) = \frac{X_1 + X_2 + X_3}{3}$$

$$X(t+1) = \begin{cases} 1, & \text{Sigmoid}\left(X(t+1)\right) > \text{rand} \\ 0, & \text{otherwise} \end{cases}$$

X **Update with Cuckoo**: $X = X + \lambda \cdot \left(X_\alpha - X\right)$, where $\lambda \sim$ Levy Flight. The random parameter λ facilitates exploration by allowing variable step sizes, avoiding local optima.

Second Stage Feature Selection: Extreme Learning Machine (ELM)

Extreme Learning Machine : A single-hidden-layer feedforward neural network

Characteristics:

- Input weights (**W**) are randomly assigned.
- Biases (**B**) are randomly assigned.
- No iterative training; output weights (**Q**) are computed directly.

Output Weights Optimization:

$$\mathbf{Q} = \mathbf{H}^{\dagger}\mathbf{T}$$

- **H**: Hidden layer output matrix
- $\mathbf{H}^{\dagger}$: Moore-Penrose inverse of **H**
- **T**: Target output matrix

Integration with GWO+CS:

1. MSE: Mean Squared Error of ELM predictions
2. Number of Features: Model complexity
3. Pareto frontier

RNN model

- Use of a RNN for single-step forecasting
- Problem: their RNN outputs only the opening price at $t + 1 \longrightarrow$ **incompatible with auto-regressive forecasting**
- **Architecture:**

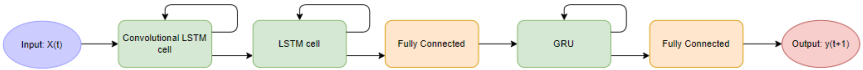


Figure: Architecture used in the paper

- **Problem:** What is a **Convolutional LSTM** in this context?

  Additionally, the computational mechanism of the convolutional LSTM (Shi et al., 2015) is similar to that of the LSTM. Different from the LSTM, the information flowing into the convolutional LSTM is a three-dimensional tensor that needs to be extracted by the convolutional layer before being processed by the LSTM.

RNN model

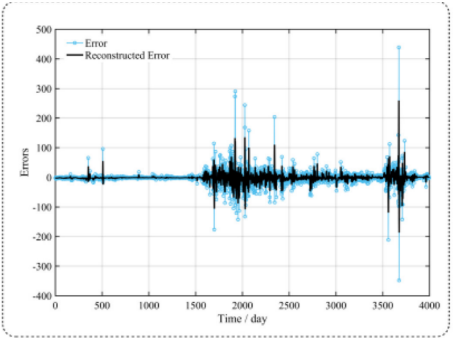
- In addition, the authors conducted a study on the impact of the learning rate and the choice of the optimizer (great).
- Their LR study ranged from 0 to 1 (not great), and showed a MSE of more than 10^6 for high LRs (expected).
- Their optimizer comparison used a LR too high (MSE of 10^7) and only 90 iterations.
- Their model has 10M parameters, for a time series of size 3800.

Error correction model

The paper uses an error correction model: they decompose the fitting errors into ”some intrinsic mode functions” (IMFs), eliminate the high-frequency IMFs, and then “ use a Lasso model to perform error forecasting ”.

- We do not know what the input of the Lasso model is.
- For a deep learning model with sufficient complexity and training data, trained with the $\mathcal{L}_2$ loss, $\mathcal{F}(\mathbf{X}[:t]) \approx \mathbb{E}[y[t+1]|\mathbf{X}[:t]]$ is already the best possible estimator.
- And if there are systematic errors or biases (due to practical architecture limitations or training data issues), the errors would need to have a linear relationship with the input features in order to improve the results.

Error correction model



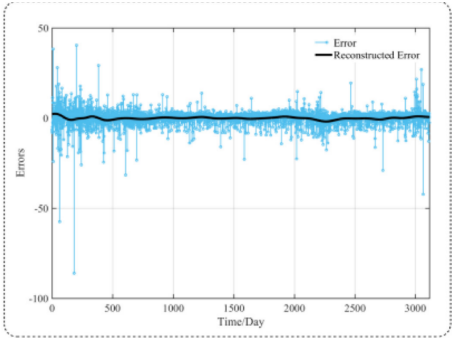
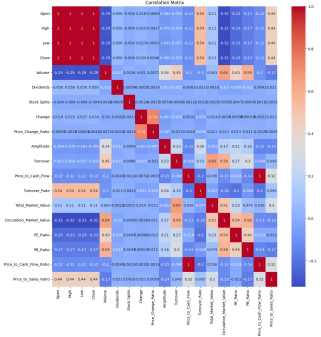


Figure: Fitting errors and corresponding reconstructions

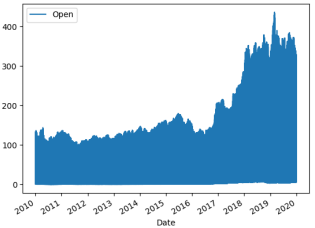
Data Used

F_1	Opening price
F_2	Highest price
F_3	Lowest price
F_4	Closing price
F_5	Change
F_6 (unit:%)	Price change ratio
F_7 (unit: 10^8)	Volume
F_8 (unit: 10^8)	Turnover
F_9 (unit:%)	Turnover rate
F_{10} (unit:%)	Amplitude
F_{11} (unit: 10^9)	Total market value
F_{12} (unit: 10^9)	Circulation market value
F_{13} (unit:%)	Price-earnings ratio
F_{14} (unit:%)	Price-to-book ratio
F_{15} (unit:%)	Price-to-cash-flow ratio
F_{16} (unit:%)	Price-to-sales ratio

(a) features from the paper



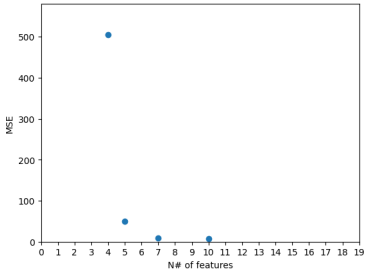
(b) Correlation Matrix



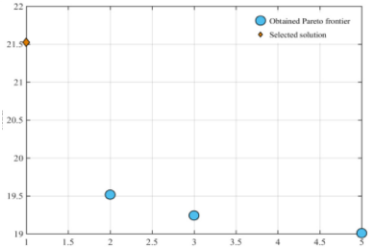
(c) Target: Opening Price

Figure: Three graphs displayed side by side.

Results - Pareto front



(a) Our pareto frontier for Dija Dataset



(b) Pareto frontier from the paper for DJIA Dataset.

Key insights:

- The hidden layer matrix was ill-conditioned, therefore we used ridge regression instead of OLS for determining the weights of the ELM. We also encountered a massive train - validation loss gap.
- We do not know what their ”optimal solution” selection method is.

Results - Train val loss

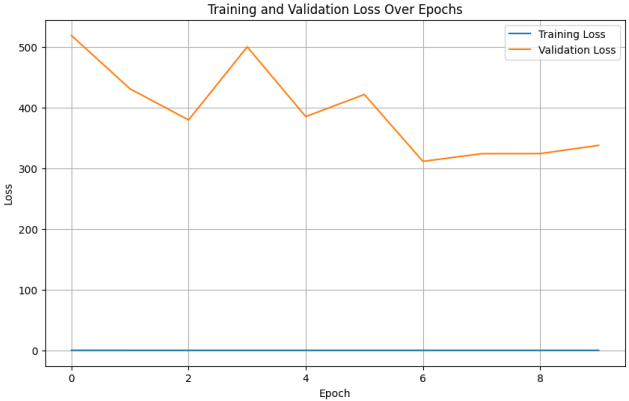


Figure: Training and Validation loss.

Results - Forecasting

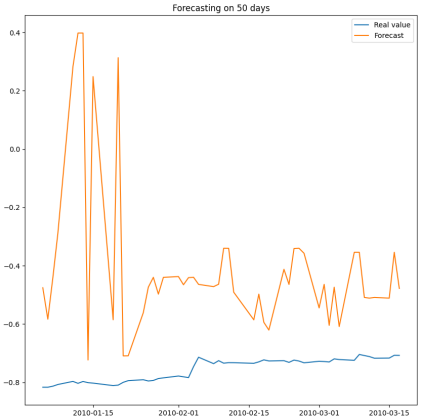


Figure: Forecasting on the validation set, using the previous values.

Implausible results

Dataset	Data partition	MAE	RMSE	MAPE	MdAPE	IA	U1	U2	DA	*R*
SZFI	Training set	9.0759	21.4930	0.2825	0.1509	1.0000	0.0032	0.3523	0.9010	0.9999
	Test set	3.2497	5.5473	0.0578	0.0244	0.9984	0.0005	0.2372	0.9137	0.9968
DJIA	Training set	2.6741	4.5812	0.0258	0.0157	1.0000	0.0002	0.0348	0.9862	1.0000
	Test set	3.2440	5.7899	0.0266	0.0137	1.0000	0.0002	0.0338	0.9829	1.0000

Figure: Paper's Forecasting accuracy evaluation based on training and test sets.

Implausible results

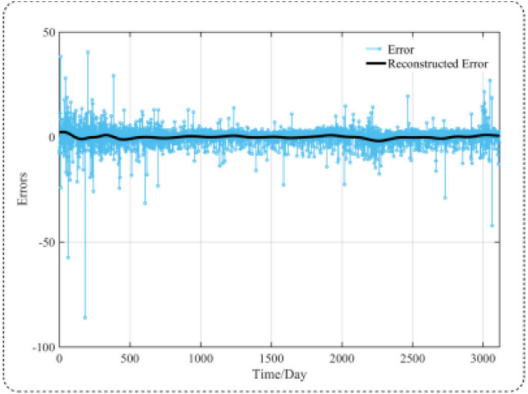


Figure: DJIA fitting errors and their reconstructed errors.

Conclusion

- Overfitting
- Unspecified design choices
- Implausible Results
- Incompatible with auto-regression
- Bad evaluation method