DATA MINING TECHNIQUES FOR PORTFOLIO BUILDING

Iryna Lazarenko Ph.D. in Mathematics
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"
Kyiv, Ukraine
ORCID ID: 0000-0002-3384-1186
e-mail: irynalazar@gmail.com

Yevhen Krykun, Master Student
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"
Kyiv, Ukraine
ORCID ID: 0009-0001-5146-4273
e-mail: genia2002g@gmail.com

Data Mining has revolutionized portfolio building and stock forecasting by providing advanced methods to analyse complex financial data. Techniques like Deep Neural Networks (DNNs) and Recurrent Neural Networks (RNNs) enable accurate predictions by identifying patterns in historical stock prices and market trends. These models enhance decision-making by reducing forecasting errors and improving portfolio performance. Leveraging their ability to process large datasets and adapt to dynamic financial environments, deep learning plays a crucial role in optimizing investment strategies and mitigating risks.

According to Markowitz's theory, the expected return of a portfolio is determined using a formula that incorporates multiple factors and their interconnections:

Rp= iRiwi,#1

where Rp is the return on the portfolio, Ri - return on the asset, wi - share of the asset in the portfolio.

The first criterion is the expected return function and second criterion is to minimize the total portfolio risk as follows:

f1=Rp →max, f2= p→min, #(2) p=p2,

where p-the portfolio's expected risk, or standard deviation.

An optimization problem to find the optimal stock portfolio [1]:

W=α*wT*cov*w1-α*i=1nRi*wi →min {i=1nwi=1 wi≥0,01 i=1nRi*wi>0 wT*cov*w≥0 ,#3

This study will utilize two types of neural networks: DNN and RNN. The Deep Neural Network (DNN) architecture will comprise four layers: Dense(150), Dense(100), Dense(50), and Dense(1), with each layer employing the ReLU activation function to enhance non-linear feature learning. Similarly, the Recurrent Neural Network (RNN) architecture will consist of two SimpleRNN layers, each containing 40 neurons, followed by a Dense(1) output layer. Consistent with the DNN, all layers within the RNN will utilize the ReLU activation function to ensure effective data processing and feature extraction [2, 3].

Upon obtaining the results, a comparative analysis of the predictive accuracy between the two models will be conducted. The findings will be summarized and presented in Table 1.

After obtaining the forecasting results, optimal portfolios will be constructed and compared against the actual portfolio to evaluate performance and alignment with real-world outcomes. The results will be presented in Table 2 and Table 3 for detailed analysis and comparison:

Table 1 – DNNs/ RNNs Predictions

	AAPL		AMZN		GOOG		MSFT		NVDA		V	
	DNNs	**RNNs**	**DNNs**	**RNNs**	**DNNs**	**RNNs**	**DNNs**	**RNNs**	**DNNs**	**RNNs**	**DNNs**	**RNNs**
MSE	8.94	9.04	12.59	9.36	9.08	7.45	37.69	25.8	21.09	10.21	8.61	8.61
MAE	2.24	2.23	2.69	2.26	2.31	1.92	4.75	3.93	3.25	2.22	2.16	2.16
MAPE	1.16	1.16	1.65	1.39	1.52	1.27	1.22	1.00	3.8	2.54	0.83	0.83
R^2	0.98	0.98	0.97	0.98	0.97	0.98	0.98	0.99	0.98	0.99	0.97	0.97

Source: created by the author based on own calculations

Table 2 – Portfolio Results

	DNN	RNN	Actual
Income	0.260	0.186	0.191
Risk	1.016	1.076	1.429

Source: created by the author based on own calculations

Based on these indicators, weighting factors for each portfolio compared to the actual weights:

Table 3 – Portfolio Weights

	DNN	RNN	Actual
AAPL	0.010	0.082	0.148
AMZN	0.010	0.010	0.010
GOOG	0.010	0.010	0.010
MSFT	0.039	0.010	0.010
NVDA	0.641	0.393	0.370
V	0.290	0.495	0.452

Source: created by the author based on own calculations

The results indicate that the RNN model provides more accurate predictions for portfolio construction compared to the DNN model. Its predictions for income, risk, and portfolio weights are generally closer to the actual values, making it a more reliable choice for forecasting.

References:

1. Lazarenko I., Krykun Y. POTRFOLIO MANAGEMENT WITH TIME SERIES ANALYSIS METHODS. https://ev.fmm.kpi.ua/article/view/309267/300787

2. Neural Network Layers: All You Need Is Inside Comprehensive Overview. https://hackernoon.com/neural-network-layers-all-you-need-is-an-inside-comprehensive-overview

3. Rectifier (neural networks). https://en.wikipedia.org/wiki/Rectifier_(neural_networks)