

Machine Learning vs. Randomness: Challenges in Predicting Binary Options Movements

Gabriel M. Arantes^{1[0009-0002-7296-2607]}, Richard F. Pinto^{1[0009-0007-0176-3383]}, Bruno L. Dalmazo^{1[0000-0002-6996-7602]}, Eduardo N. Borges^{1[0000-0003-1595-7676]}, Giancarlo Lucca^{2[0000-0002-3776-0260]}, Viviane L. D. de Mattos^{1[0000-0002-3512-6290]}, Fabian C. Cardoso^{3[0000-0002-2842-0387]}, and Rafael A. Berri^{1[0000-0002-3812-4186]}

¹ Federal University of Rio Grande (FURG), Rio Grande, Brazil
 {gma26062004, richard_pinto, dalmazo, eduardoborges,
 vivianemattos, rafaelberri}@furg.br

² Catholic University of Pelotas (UCPel), Pelotas, Brazil
 giancarlo.lucca@ucpel.edu.br

³ University of Rio Verde (UniRV) , Rio Verde, Brazil
 fabian@unirv.edu.br

Abstract. Binary options trading is often marketed as a field where predictive models can generate consistent profits. However, the inherent randomness and stochastic nature of binary options make price movements highly unpredictable, posing significant challenges for any forecasting approach. This study demonstrates that machine learning algorithms struggle to outperform a simple baseline in predicting binary options movements. Using a dataset of EUR/USD currency pairs from 2021 to 2023, we tested multiple models, including Random Forest, Logistic Regression, Gradient Boosting, and k-Nearest Neighbors (kNN), both before and after hyperparameter optimization. Furthermore, several neural network architectures, including Multi-Layer Perceptrons (MLP) and a Long Short-Term Memory (LSTM) network, were evaluated under different training conditions. Despite these exhaustive efforts, none of the models surpassed the ZeroR baseline accuracy, highlighting the inherent randomness of binary options. These findings reinforce the notion that binary options lack predictable patterns, making them unsuitable for machine learning-based forecasting.

Keywords: Binary Options · Machine Learning · Neural Networks · Financial Market Prediction · Time Series Analysis.

1 Introduction

The financial market has become increasingly complex, requiring advanced tools for analysis and decision-making [23]. In this context, binary options have emerged as financial instruments that offer rapid returns but also carry significant risks due to their speculative nature. Unlike traditional financial assets, where price movements can often be explained by fundamental or technical analysis, binary

options operate under conditions that frequently resemble stochastic processes [22]. This raises the question of whether these instruments can be effectively predicted using data-driven approaches like machine learning.

While ML techniques have shown promise in other financial forecasting tasks [4, 18, 26], prior research suggests that markets exhibiting near-random characteristics may fundamentally limit their predictive power [22]. Given the popularity of binary options and the potential for significant financial losses, it is crucial to investigate whether advanced techniques can provide accurate predictions. This study aims to evaluate the effectiveness of various machine learning techniques in this domain by investigating the impact of feature selection, hyper-parameter optimization, and comparing different algorithms (including Random Forest, Logistic Regression, Gradient Boosting, kNN, and neural networks) to assess their ability to learn predictive patterns, using the ZeroR model as a minimum performance benchmark. Our results seek to contribute to the discussion on ML’s applicability in highly speculative markets, providing insights into whether these models can forecast binary options or if their limitations reinforce the notion of market randomness.

This paper is structured as follows: Section 2 presents the theoretical background. Section 3 discusses related works. Section 4 describes our methodology, followed by Section 5 with the experimental results. Finally, Section 6 concludes the paper.

2 Theoretical Foundations

This section briefly covers the core concepts applied in our experiments.

2.1 Binary Options, Technical Indicators, and ML Algorithms

Binary options allow traders to forecast an asset’s price direction relative to a predefined value at a specific expiration time [9]. Their speculative nature and resemblance to stochastic processes pose a significant prediction challenge [2, 21]. We use two widely-acknowledged technical indicators: the Simple Moving Average (SMA) [3], a trend-following indicator, and the Relative Strength Index (RSI) [25], a momentum oscillator.

Our study employs several machine learning algorithms: k-Nearest Neighbors (kNN) [7], Random Forest [12], Gradient Boosting [8], Logistic Regression [24], Multi-Layer Perceptron (MLP) [14], and Long Short-Term Memory (LSTM) [13] networks. The MLP learns complex non-linear dependencies via backpropagation [20], with its weight update rule given by:

$$w_{ij}^{(t+1)} = w_{ij}^{(t)} - \eta \frac{\partial E}{\partial w_{ij}} \quad (1)$$

LSTMs are designed to capture long-term temporal patterns, using a memory cell (c_t) and gates (f_t, i_t) governed by the following key equations:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (2)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (3)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (4)$$

2.2 Feature Selection, Optimization, and Evaluation

Feature selection is vital for increasing model efficiency by removing unimportant variables [10]. We used the `SelectFromModel` method with Random Forest to identify the most relevant features based on impurity reduction [5]. Hyperparameter optimization refines model performance by tuning learning parameters. We employed Hyperband, an efficient resource-allocation algorithm that adaptively focuses on promising configurations [17]. To ensure reliability, we used k-fold cross-validation for model evaluation, which partitions data into subsets for training and validation to assess generalization [16]. Model performance was measured by accuracy, which quantifies correctly classified instances [19].

3 Related Works

The use of machine learning for financial forecasting has grown in recent years. In deep learning, a notable focus has been on Recurrent Neural Networks for stock price prediction, with studies indicating the potential of models like LSTM and GRU to capture temporal dependencies [6]. Other research has explored network-based features to model market correlation structures, using graph theory to improve prediction accuracy [4].

A significant portion of the literature involves comparing traditional machine learning models with deep learning approaches. For financial time series, deep learning models are often reported to perform better, particularly in capturing long-term patterns [11]. Concurrently, comprehensive reviews have highlighted the advantages of fuzzy logic and neural networks in handling market uncertainties [1], while early work demonstrated the feasibility of using MLPs, laying the groundwork for their adoption in financial applications [15].

Our research builds on these foundations by systematically applying and evaluating a wide range of these techniques specifically to the binary options market, focusing on the challenges posed by its apparent randomness.

4 Methodology

This section outlines the methodological framework, detailing the dataset, technical indicators, and machine learning techniques. The objective is to test the hypothesis that binary options are entirely random by applying multiple machine learning methods and determining if they can outperform a baseline ZeroR classifier.

4.1 Dataset Selection and Preprocessing

We utilized a historical Forex dataset from *HistData*, consisting of minute-by-minute Euro/Dollar (EUR/USD) data from 2021–2023. The dataset was divided into a training set (2021–2022, N = 742,240) and a testing set (2023, N = 322,572) to ensure models learn from historical patterns without information leakage.

4.2 Feature Engineering

We derived a set of technical indicators: Simple Moving Averages (SMA) with windows of 2, 3, 4, and 5 minutes (sma2, sma3, sma4, sma5), and Relative Strength Index (RSI) with the same windows (rsi2, rsi3, rsi4, rsi5). These were used as input variables to provide insights into price trends and market momentum.

4.3 Machine Learning Models and Evaluation

A variety of machine learning methods were applied: ZeroR (baseline), Random Forest, Logistic Regression, Gradient Boosting, and k-Nearest Neighbors (kNN). A Random Forest-based feature selection process was performed first. To ensure a robust comparison and improve stability, the input features for all models were normalized using the standardization technique (`StandardScaler`). Each model was evaluated using 5-fold cross-validation on the training data.

4.4 Hyperparameter Optimization and Neural Networks

The Hyperband algorithm was used to further investigate model performance by tuning the hyperparameters of the previously tested machine learning models. To further probe for non-linear patterns, several neural network architectures were explored, including Multi-Layer Perceptron (MLP) and Long Short-Term Memory (LSTM) networks. For all neural network experiments, input features were also normalized via `StandardScaler`.

A separate experiment was conducted where an MLP was trained on a smaller subset of the original training data to allow for an extended duration (up to 1,000 epochs) with an early stopping mechanism. For this purpose, dedicated training and validation sets were sampled from the 2021–2022 data, while the test set was constructed by combining the remaining 2021–2022 records with the entire original 2023 test set.

4.5 Final Model Evaluation

The final evaluation was conducted on the test dataset to assess generalization ability. Accuracy was the main performance metric, enabling direct comparison. The target variable was a binary classification problem: PUT (next price fall) or CALL (next price rise).

5 Results

This section presents the experimental results, covering feature selection, hyperparameter tuning, and neural network performance.

5.1 Feature Selection and Initial Performance

Feature selection using `SelectFromModel` with a Random Forest classifier retained only the four RSI features (`rsi2`, `rsi3`, `rsi4`, `rsi5`), discarding all SMA indicators. Table 1 shows that this had a negligible effect on accuracy, suggesting SMA features were not significant. Furthermore, before hyperparameter tuning (Table 2), only Logistic Regression matched the ZeroR baseline (0.5389), while others performed worse.

Table 1: Accuracy before and after feature selection.

Model	Before Feature Selection	After Feature Selection
Random Forest	0.5053	0.5062

Table 2: Initial model hyperparameters and accuracy.

Model	Initial Hyperparameters	Accuracy
Random Forest	100 trees	0.5060
Logistic Regression	max_iter=500	0.5389
Gradient Boosting	n_estimators=100, learning_rate=0.1	0.5386
kNN	n_neighbors=5	0.5055

5.2 Performance After Hyperparameter Tuning

After tuning with Hyperband, the optimized hyperparameters (Table 3) improved the accuracy of Random Forest, Gradient Boosting, and kNN. However, despite these improvements, no model surpassed the ZeroR baseline of 0.5389, reinforcing the difficulty of predicting binary options.

Table 3: Optimized hyperparameters and accuracy.

Model	Optimized Hyperparameters	Accuracy
Random Forest	n_estimators=161, max_depth=10, min_samples_split=9	0.5389
Logistic Regression	C=0.1699, max_iter=755, solver=liblinear	0.5389
Gradient Boosting	n_estimators=50, learning_rate=0.01, max_depth=3	0.5389
kNN	n_neighbors=9, metric=manhattan, weights=uniform	0.5075

5.3 Neural Network Performance

A MLP was tested with the 4 features selected. The architecture consisted of an input layer (4 neurons), two hidden layers (16 and 8 neurons with ReLU), and a sigmoid output layer, with 20% dropout. Trained for 30 epochs with the Adam optimizer, the MLP achieved an accuracy of 0.5389, identical to the ZeroR model.

Additionally, an LSTM network was evaluated. The model was configured with an input layer reshaped for sequence data, a single LSTM layer with 16 units, and a Dense sigmoid output layer. Trained for 20 epochs, this model also achieved a final test accuracy of 0.5389.

5.4 Extended Training Experiment on a Data Subset

To investigate if an extended training period could enable a model to discover underlying patterns, a final experiment was conducted with a distinct data configuration. A small subset of the 2021-2022 dataset was sampled to create a training set ($N = 3,000$) and a validation set ($N = 3,000$). The test set was then constructed by combining the remainder of the 2021-2022 records with the entire original 2023 test set, forming a test set of 1,058,807 samples. The ZeroR baseline accuracy for this specific data partition was calculated to be 0.5379.

For this experiment, a simpler MLP architecture was used, consisting of one hidden layer with 4 neurons and a sigmoid activation function, followed by a single sigmoid output neuron. The model was compiled with the RMSprop optimizer and trained for up to 1,000 epochs, employing an early stopping mechanism with a patience of 200 epochs on the validation loss.

The training was halted by the early stopping callback at epoch 210, with the model restoring the weights from epoch 10, where the best performance on the validation set was observed. The model's performance trajectory, shown in Figure 1, reveals a classic case of overfitting. Ultimately, despite the extended training, the MLP's accuracy on the final test set was 0.5379, a result identical to the new ZeroR baseline.

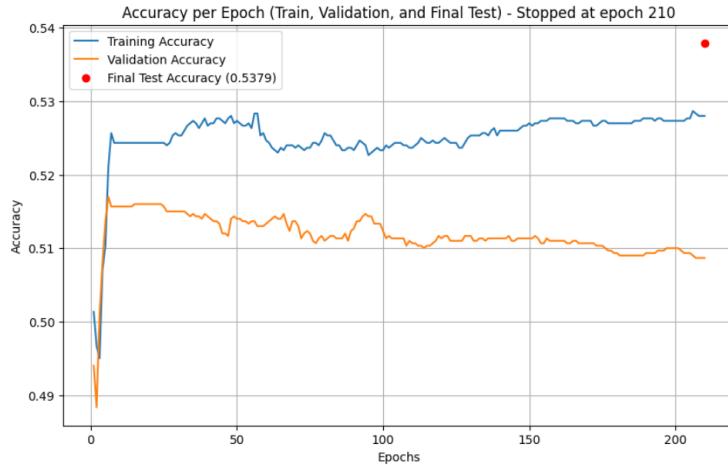


Fig. 1: Training and validation accuracy of the MLP over epochs. The divergence between the rising training accuracy (blue) and the decreasing validation accuracy (orange) is a clear sign of overfitting. The final test accuracy on the comprehensive unseen dataset (red dot) matched the ZeroR baseline exactly, confirming that the patterns memorized during training failed to generalize.

5.5 Analysis of Model Behavior via Confusion Matrices

While accuracy scores provide a high-level measure of performance, an analysis of the confusion matrices offers deeper insight into the models' predictive behavior prior to hyperparameter optimization. This analysis reveals why even the more complex models, in their default state, failed to outperform the simple ZeroR baseline. Figure 2 illustrates the spectrum of predictive behaviors observed across these initial experiments.

The first and most common pattern was the model's convergence to a strategy indistinguishable from the ZeroR baseline: predicting the majority class ('PUT') exclusively. This was observed in Logistic Regression, the 30-epoch MLP, and the LSTM network (Fig. 2(a)). A slight variation was seen in models like Gradient Boosting and the extended-training MLP, which made a minimal, yet insignificant, attempt to classify the minority class (Fig. 2(b)). The third pattern, from Random Forest and k-NN, involved a more genuine attempt to classify both classes but resulted in high error rates and an accuracy no better than the baseline (Fig. 2(c)).

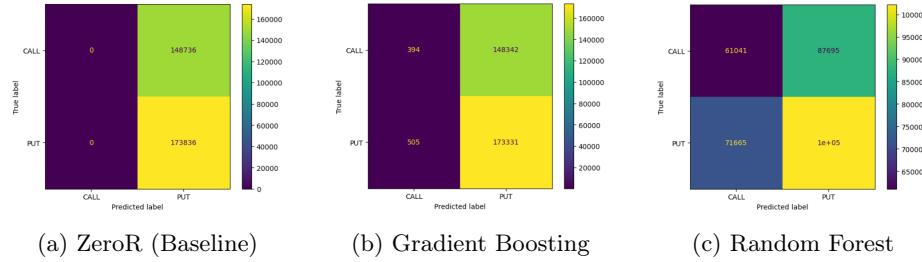


Fig. 2: Confusion matrices from models prior to hyperparameter optimization, illustrating three distinct predictive behaviors: (a) the baseline majority-class prediction, (b) a minimal deviation from the baseline, and (c) an unsuccessful attempt to classify both classes.

5.6 Final Analysis

To summarize all results, Figure 3 illustrates the accuracy of each primary model after hyperparameter tuning. Furthermore, the extended training experiment with the MLP, which used a different data partition, also concluded with the model achieving an accuracy identical to its corresponding ZeroR baseline.

As the results from all experiments demonstrate, none of the models outperformed their respective ZeroR baseline. The consistency in accuracy across different algorithms, configurations, and training approaches suggests that binary options exhibit high randomness, making them unsuitable for predictive modeling using these machine learning techniques.

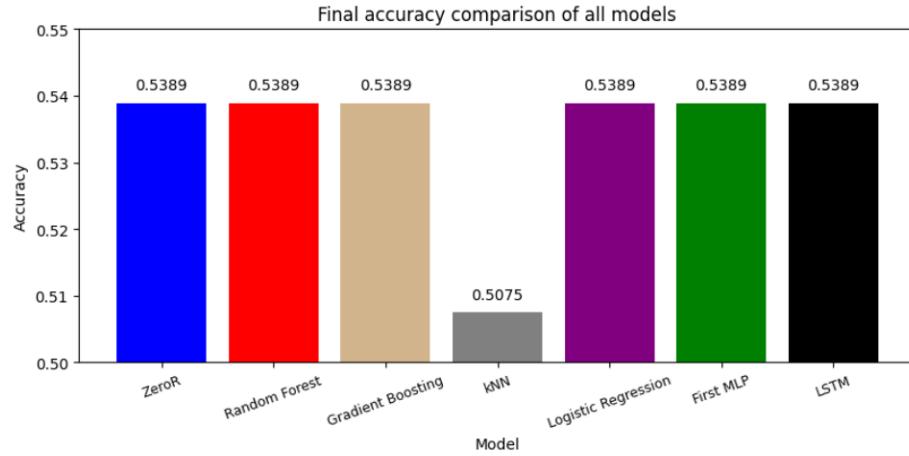


Fig. 3: Final accuracy comparison of all primary models against the ZeroR baseline.

6 Conclusion

This study conducted a comprehensive investigation into the applicability of machine learning for predicting binary options movements. A wide range of models was evaluated, from traditional algorithms such as Random Forest, Logistic Regression, Gradient Boosting, and k-NN, to more complex neural network architectures, including Multi-Layer Perceptrons and a Long Short-Term Memory (LSTM) network. The impact of feature selection, extensive hyperparameter optimization, and different training paradigms was systematically examined against a simple ZeroR baseline model.

Our findings are unequivocal: despite the application of diverse and sophisticated learning algorithms, none of the evaluated models were able to outperform their respective ZeroR baseline. A deeper analysis of the models' behavior revealed a critical insight: most algorithms, including the LSTM and MLP, converged to a simplistic strategy of predicting the majority class, as evidenced by their confusion matrices. This demonstrates a fundamental failure to learn any true predictive signal. Even models that attempted a more balanced classification, such as Random Forest, produced high error rates that resulted in even worse performance than the baseline.

The implications of these findings are significant. They reinforce the notion that binary options markets operate with a high degree of stochasticity, posing substantial challenges for machine learning-based forecasting. Our results suggest that simple technical indicators, such as SMA and RSI, are insufficient to capture any underlying predictability. Therefore, future research in this domain should pivot towards fundamentally different approaches. This could involve exploring more holistic feature sets, such as order book information, macroeconomic news, and sentiment data, or employing models specifically designed for noisy, non-stationary environments.

In conclusion, this study demonstrates the profound limitations of applying machine learning techniques to predict binary options movements. The consistent failure of diverse models to find a predictive edge serves as strong empirical evidence supporting the idea of market randomness in this specific, highly speculative context. These results should serve as a cautionary note for both practitioners and researchers, emphasizing that without a source of true predictive information, machine learning models are likely to perform no better than random chance in the specific context of binary options prediction.

Acknowledgments. The authors would like to thank FAPERGS (24/2551-0001396-2, 23/2551-0000773-8), CNPq (305805/2021-5) and FAPERGS/CNPq (23/ 2551-0000126-8).

References

1. Atsalakis, G.S., Valavanis, K.P.: Surveying stock market forecasting techniques – part ii: Soft computing methods. *Expert Systems with Applications* **36**(3), 5932–5941 (2009)

2. Biondo, A., et al.: Are financial markets really efficient? *Journal of Economic Interaction and Coordination* **8**, 1–16 (2013)
3. Brown, C.M.: *Technical Analysis for the Trading Professional*. McGraw-Hill Education (2005)
4. Castilho, D., Souza, T.T., Kang, S.M., Gama, J., de Carvalho, A.C.: Forecasting financial market structure from network features using machine learning. *Knowledge and Information Systems* pp. 1–27 (2021)
5. Chandrashekhar, G., Sahin, F.: A survey on feature selection methods. *Computers & Electrical Engineering* **40**(1), 16–28 (2014)
6. Chang, A., Smith, J., Lee, M.: Predicting economic trends and stock market prices with deep learning models. *Electronics* **13**(17), 3396 (2024)
7. Cover, T., Hart, P.: Nearest neighbor pattern classification. *IEEE Transactions on Information Theory* **13**(1), 21–27 (1967). <https://doi.org/10.1109/TIT.1967.1053964>
8. Friedman, J.H.: Stochastic gradient boosting. *Computational Statistics & Data Analysis* **38**(4), 367–378 (2002). <https://www.sciencedirect.com/science/article/pii/S0167947301000652>, non-linear Methods and Data Mining
9. Gandar, J.M., Zuber, R.A., Johnson, T.C., et al.: Testing rationality in the point spread function. *The Journal of Finance* **43**(4), 995–1008 (1988)
10. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. *Journal of Machine Learning Research* **3**, 1157–1182 (2003)
11. Hiransha, M., Gopalakrishnan, E., Menon, V., Soman, K.: An empirical study of machine learning algorithms for stock market forecasting. *arXiv preprint arXiv:1805.02015* (2018)
12. Ho, T.K.: Random decision forests. In: *Proceedings of 3rd international conference on document analysis and recognition*. vol. 1, pp. 278–282. IEEE (1995)
13. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural computation* **9**(8), 1735–1780 (1997)
14. Jain, A.K., Mao, J., Mohiuddin, K.M.: Artificial neural networks: A tutorial. *Computer* **29**(3), 31–44 (1996)
15. Kimoto, T., Asakawa, K., Yoda, M., Takeoka, M.: Stock market prediction system with modular neural networks. In: *Proceedings of the International Joint Conference on Neural Networks*. pp. 1–6. IEEE (1990)
16. Kohavi, R.: A study of cross-validation and bootstrap for accuracy estimation and model selection. In: *International Joint Conference on Artificial Intelligence*. vol. 14, pp. 1137–1145. Morgan Kaufmann (1995)
17. Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A., Talwalkar, A.: Hyperband: A novel bandit-based approach to hyperparameter optimization. In: *Proceedings of the 34th International Conference on Machine Learning (ICML)*. vol. 70, pp. 1067–1075. PMLR (2017), <https://proceedings.mlr.press/v70/li17a.html>
18. Obthong, M., Tantisantiwong, N., Jeamwatthanachai, W., Wills, G.: A survey on machine learning for stock price prediction: Algorithms and techniques. In: *Proceedings of 2nd International Conference on Finance, Economics, Management and IT Business* (2020)
19. Powers, D.M.W.: Evaluation: From precision, recall and f-measure to roc, informedness, markedness and correlation. *Journal of Machine Learning Technologies* **2**(1), 37–63 (2011)
20. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. *Nature* **323**, 533–536 (1986)

21. Securities, E., Authority, M.: Esma decision (eu) 2018/795 of 22 may 2018 on product intervention measures on binary options. Official Journal of the European Union (2018)
22. Silva, R., Matsushita, R., Matsushita, J.: Física de processos estocásticos aplicada a opções binárias no mercado financeiro. Revista Brasileira de Ensino de Física (2020), <https://www.scielo.br/j/rbef/a/QvvNPWh9tfVWzbcB6gGQc6Q/>
23. Systems, T.: A revolução dos dados no mercado financeiro (2020), <https://www.twist.systems/pt-br/blog/2020/10/21/dados-mercado-financeiro/>
24. Tolles, J., Meurer, W.J.: Logistic regression: Relating patient characteristics to outcomes. *JAMA* **316**(5), 533–534 (08 2016), <https://doi.org/10.1001/jama.2016.7653>
25. Wilder, J.W.: New concepts in technical trading systems. *Trend Research* (1978)
26. Zhang, Y., Zhou, X.: Neural network approaches to stock price prediction in volatile markets. *Journal of Finance and Data Science* **8**(3), 210–225 (2022)