Predictive Analysis with Technical Indicators and Features Selection for Futures Contracts Trading

Andrey V. S. Souza[1(✉)], Richard F. Pinto[1], Bruno L. Dalmazo[1], Eduardo N. Borges[1], Giancarlo Lucca[2], Viviane L. D. de Mattos[1], and Rafael A. Berri[1]

[1] Federal University of Rio Grande (FURG), Rio Grande, Brazil
{andreyvinicius,richard_pinto,dalmazo,eduardoborges,vivianemattos,rafaelberri}@furg.br
[2] Catholic University of Pelotas (UCPel), Pelotas, Brazil
giancarlo.lucca@ucpel.edu.br

Abstract. This study investigates the impact of feature selection on the predictive performance of machine learning models in the Brazilian derivatives market. Precisely, by integrating technical analysis indicators, such as Simple and Exponential Moving Averages and Standard Deviation, derived from intraday price data of mini-Bovespa index futures traded on the Brazilian Stock Exchange, we construct a robust dataset of historical prices from the BovDBV2 database. We employ several feature selection methods to reduce dimensionality, minimize overfitting, and improve model generalization. A Random Forest classifier, evaluated using a 9-fold stratified cross-validation scheme, is employed to compare the performance of models trained on the full feature set versus those trained on reduced subsets. Our findings demonstrate that feature reduction significantly improves accuracy on test data while reducing computational costs. These results provide valuable insights for investors and traders, highlighting the importance of feature selection in developing predictive models capable of adapting to dynamic derivatives market conditions.

Keywords: Machine Learning · Derivatives market · Technical Analysis · Feature Selection

1 Introduction

The financial market relies on various types of analysis, forecasts, and models to assist in business decision-making around the world [3], playing a crucial role in the generation and interpretation of large volumes of data. Among the most traded assets in this environment are futures contracts, which are standardized financial agreements between two parties to buy or sell an underlying asset at a predetermined price on a specified future date [10].

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
O. Gervasi et al. (Eds.): ICCSA 2025, LNCS 15649, pp. 349–367, 2025.
https://doi.org/10.1007/978-3-031-96997-3_22

Stock exchanges are essential financial institutions for the trading of stocks, bonds, and other financial assets [8]. Globally, there are a variety of exchanges such as the New York Stock Exchange[1] NASDAQ[2] and the London Stock Exchange[3] which operate under distinct regulatory frameworks and serve different market segments [7]. In Brazil, the Brazilian Stock Exchange (B3)[4] plays a central role in the financial market, enabling companies to raise funds through the issuance and sale of stocks, bonds, and futures contracts, providing a structured and regulated environment. In this context, investors trade various financial instruments on B3, such as index contracts (for example, the Mini Index (WIN), based on the Ibovespa) and currency contracts (DOL and WDO, based on the dollar exchange rate), among others, promoting liquidity and facilitating the flow of capital between companies and investors.

Due to the economic importance of financial assets, particularly in the equity and derivatives markets [5], researchers and practitioners have extensively studied patterns and behaviors to improve market forecasting. Traditionally, technical analysis is one of the main approaches for market analysis, and it uses charts and historical price data to identify trends and potential reversals [6]. In recent years, the field of stock market forecasting has emerged at the intersection of finance and computing [14]. Machine Learning (ML), a subfield of artificial intelligence that develops algorithms capable of learning from data without explicit programming [16], has become central to predicting market behavior.

A critical step in building robust machine learning models is feature selection. This process involves identifying and retaining only the most relevant attributes to increase prediction accuracy while reducing computational complexity and training time. Since not all features contribute equally to predictive performance, techniques such as filtering, encapsulation, or embedding methods [24] are essential to minimize overfitting and improve model generalization.

The objective of this paper is to evaluate the effectiveness of feature selection methods in financial time series in order to improve the performance of predictive models. To achieve this, the study integrates technical analysis with advanced machine learning techniques, optimizing the accuracy and adaptability of the model to market dynamics.

This paper is organized as follows: Sect. 2 addresses Feature Selection Techniques and Technical Analysis Indicators; in Sect. 3, a literature review summarizes the main previous studies; Sect. 4 details the Methodology of this study, while Sect. 5 presents the findings of the study and, finally, Sect. 6 presents the conclusions and future work for the research.

[1] Access the NYSE website at www.nyse.com.
[2] Access the NASDAQ website at www.nasdaq.com.
[3] Access the LSEG website at www.londonstockexchange.com.
[4] Access the B3 website at www.b3.com.br.

2 Theoretical Concepts

In this section, we aim to contextualize our research by presenting the technical analysis indicators that are used as input to the models, followed by an exploration of the feature selection techniques employed.

2.1 Technical Analysis Indicators

Technical analysis indicators are quantitative tools derived from historical asset data that are used to predict future market movements. These indicators help traders identify trends, momentum, and potential reversal points, offering a systematic approach to interpreting market fluctuations.

Moving Average. The moving average is a fundamental statistical technique used to smooth the volatility inherent in time series data [9]. Different forms of Moving Averages, such as the Simple Moving Average (SMA) and the Exponential Moving Average (EMA), provide varying degrees of sensitivity to recent price changes. The use of moving averages in technical analysis helps traders to filter out the 'noise' in market data, making it easier to identify support and resistance levels and to confirm trend directions.

Standard Deviation. Standard deviation is a statistical metric that quantifies price volatility by measuring how much an asset's price deviates from its mean over a specific period. In technical analysis, it is calculated by first determining the moving average (mean) of closing prices and then calculating the squared differences between each price and that average. The average of these squared differences produces the variance, and the square root of the variance produces the standard deviation.

2.2 Feature Selection

Feature selection is the process of identifying and retaining the most relevant variables from a dataset to optimize machine learning models. By strategically eliminating redundant or uninformative features, this technique increases model accuracy, reduces training time, and improves interpretability.

Relief Method. The Relief method evaluates feature importance by comparing local differences between instances to determine how well an attribute distinguishes between different outcomes. This approach is valuable in both feature selection and financial time series analysis, where it can highlight subtle, context-dependent patterns without heavy computational demands.

Gain Ratio. Gain Ratio extends Information Gain by normalizing it with the intrinsic information of a feature, thereby reducing the bias toward attributes with many distinct values. This normalization makes it a practical choice for feature selection and financial data analysis where balanced split decisions are crucial [11].

Pearson Correlation. The Pearson Correlation Coefficient quantifies the linear relationship between two variables and is commonly used in feature selection to identify redundant attributes. Highly correlated features can be removed to reduce multicollinearity and improve model generalization.

Principal Component Analysis (PCA). PCA reduces dimensionality by projecting the data onto a set of uncorrelated principal components that capture the largest variance. This is achieved through an eigenvalue decomposition of the covariance matrix. PCA is particularly effective in financial time series for noise reduction and redundancy elimination.

Information Gain. Information Gain measures how much knowledge of an attribute reduces the uncertainty of a target variable [2]. This method is mainly used to classify attributes based on their ability to separate classes, favoring those that contribute most to predictive performance.

3 Related Work

The financial market generates vast volumes of data, which analysts and researchers increasingly analyze using advanced computational techniques [4]. Recent advancements in artificial intelligence, particularly Machine Learning (ML) and Deep Learning (DL), have drawn significant attention from investors seeking to predict asset prices and optimize trading strategies.

An important component of these models is the integration of technical analysis indicators as input features [18]. Moving Averages (MA) and Bollinger Bands (derived from standard deviation) are commonly used to encode trends and volatility. [17] demonstrated that combining MA crossover signals with Long Short Term Memory (LSTM) architectures reduced forecast errors by 22% in currency futures. Meanwhile, [25] highlighted the role of volatility indicators in improving risk-adjusted returns for stock index futures. These works emphasize that feature engineering, based on domain-specific knowledge, improves model interpretability and performance.

Feature selection plays a key role in optimizing these models; search techniques such as mutual information and Recursive Feature Elimination (RFE) have shown that selective subsets of features reduce overfitting in stock price prediction tasks. Li2017 [13] argued that feature selection is indispensable for balancing model complexity and generalizability, especially in non-stationary markets.

The research by [15] highlights a fundamental challenge: unlike static datasets, financial data exhibits rapid fluctuations and non-linear dynamics, which introduce noise and obscure genuine patterns. They emphasize that feature selection techniques such as PCA are essential to isolate informative indicators (e.g., momentum or volume trends) while filtering out redundant variables. Traditional predictive models often struggle under these conditions, as the inherent non-linearity of price movements can lead to problems such as overfitting and reduced generalization performance, as pointed out in the study by [19], which highlights that these models produce a memorization of the dataset rather than a prediction.

As highlighted in the reviewed literature, feature selection plays a critical role in improving the robustness and generalizability of predictive models in financial contexts. Despite its importance, many studies still apply selection techniques without comparative validation and in a generic manner. That said, this study seeks to fill three main gaps: (1) improve interpretability in models with multiple technical indicators; (2) leverage high-frequency intraday data more effectively; and (3) conduct a systematic comparative analysis of multiple feature selection methods under consistent experimental conditions. With this, this research aims to provide more reliable and efficient strategies for financial market forecasting.

4 Methodology

This chapter presents the methodology adopted in this study. The first Sect. 4.1 discusses what data will be used and where it will be obtained from. Section 4.2 presents information about the labels generated to train the model used to perform feature selection. While Sect. 4.3 will present how the features were generated. Finally, Sect. 4.4 presents the techniques used in this study.

Figure 1 shows an overview of how our methodology is organized. It separates each section by different colors and presents what will be covered in the section.

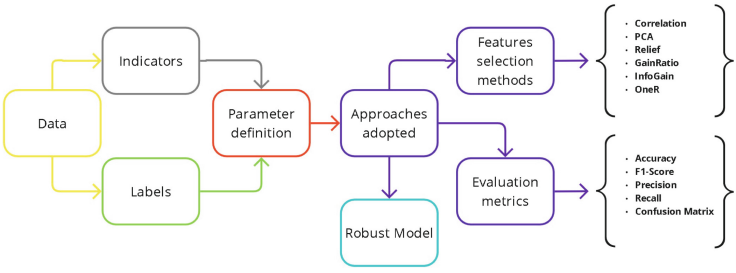


Fig. 1. Overview methodology.

4.1 Data

The data utilized in this study were obtained from the BovDbV2 [23] database, an updated version of the former BovDb [8]. The primary goal of this dataset is to provide publicly available, pre-processed data from the Brazilian Stock Exchange (B3) for research purposes. Unlike its predecessor, BovDbV2 offers a more comprehensive collection of data by including daily pricing information along with intraday trading details.

The intraday data capture asset price fluctuations at regular time intervals during standard trading hours, enabling a more granular analysis of short-term price behavior (i.e., within the same day). For the purposes of this study, we focus on four main tables within the dataset. Figure 2 illustrates the relationships among these tables.

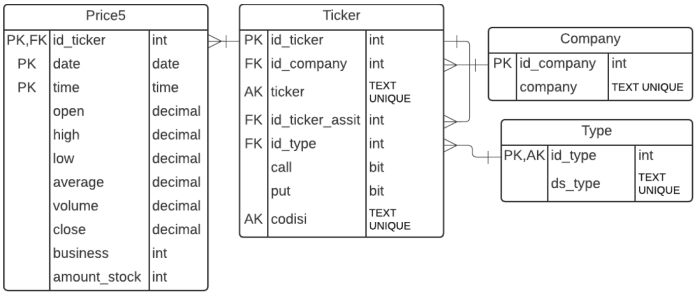


Fig. 2. Reduced Relational Model of the BovDBV2 Dataset.

The main tables are described as follows:

- **Company**: Stores information related to companies;
- **Ticker**: Contains data for individual securities;
- **Price5**: Records intraday trading data, including detailed price and volume information captured in 5-minute intervals throughout the trading day;
- **Type**: Classifies tickers into various categories, such as stocks, BDRs, financial instruments, and options.

This article specifically analyzes futures contract data extracted from the intraday table *Price5*. It focuses on different WiN tickers, which represent mini futures contracts based on the Ibovespa index. These contracts are popular among investors due to their lower cost and high liquidity, making them one of the most frequently traded instruments each month and providing a robust basis for examining short-term market dynamics.

4.2 Label Generation

In this study, the feature selection is oriented toward a classification problem, which requires the creation of labels to validate the selection technique. The

labels used are *uptrend* and *downtrend*. These labels are generated based on the identification of significant tops and bottoms within the intraday 5-minute time series data for each futures contract.

Tops and bottoms, also known as peaks and troughs, are fundamental concepts in technical analysis [21]. A top (or peak) is defined as a local maximum in a price series, an instance where the price reaches a high point before reversing direction. On the other hand, a bottom (or trough) is a local minimum where the price reaches a low before initiating an upward movement. These pivot points are crucial for recognizing trend reversal points [1] and understanding market dynamics [20].

The label generation process involves the following steps:

1. **Pivot Point Identification:** For each 5-minute time series, an algorithm is applied to detect local maxima (tops) and minima (bottoms). This detection is implemented using criteria such as comparing each data point with its neighboring values to determine whether it qualifies as a significant peak or trough;
2. **Label Assignment:** Once pivot points are identified, labels are assigned based on the most recent significant pivot. If the last major pivot is a bottom, followed by a sustained price increase, the time series is labeled as an *uptrend*. Conversely, if the last major pivot is a top, followed by a consistent decline in price, the label is set as a *downtrend*.

4.3 Parameter Definition

In our methodology, we improve the predictive performance of our model by deriving additional features from existing technical indicators. This process has been divided into three main approaches:

1. **Generation of Technical Analysis Indicators:** We computed primary indicators, Simple Moving Average (SMA), Exponential Moving Average (EMA) and Standard Deviation over various time windows. These indicators capture key aspects of market trends and volatility, forming the foundational data for further feature creation [9];
2. **Moving Average Differences:** To capture changes in trend dynamics, we calculated the differences between moving averages computed over different time periods. For example, the difference between the 7-period SMA and the 3-period SMA is given by

$$\texttt{SMA7-3} = SMA_7 - SMA_3 \tag{1}$$

   which provides insights into short- and medium-term momentum shifts;
3. **Feature Normalization:** Normalization scales features to a common range (typically [0, 1]) to improve model adaptability. All indicators were normalized (as well as open and close prices) by subtracting the feature value from "Low" divided by "High - Low", where "High" and "Low" are respectively the highest and lowest values of each 5-minute interval.

Using these methods, new features were generated for each indicator. The moving averages (SMA and EMA) generated 16 features, 8 in the windows used and 8 normalized. The standard deviation generated 16 features, 8 in the windows used with their respective values and 8 normalized. The difference between the moving averages generated 24 new features, 12 original and 12 normalized, totaling 56 features created from technical analysis indicators.

4.4 Approaches Adopted

This study focuses on feature selection to improve the predictive performance of futures market models for day trading applications and movements. Our methodology is based on the acquisition of a preprocessed dataset containing intraday data in 5-minute intervals contained in the Price5 table of the BovDBV2 database. The detailed price movements of each trading session are meticulously recorded, with columns such as `open`, `high`, `low`, `average`, `close`, `bid`, `ask`, `trade`, and `number of shares`.

For model training and evaluation, we adopted the Random Forest (RF) classifier, a robust ensemble learning technique that builds multiple decision trees and aggregates their predictions to achieve higher accuracy and control overfitting. Each tree is trained on a random subset of data and features, which makes the model particularly effective when dealing with high-dimensional datasets [12]. We selected this model due to its proven robustness, low computational complexity, and suitability for structured and tabular data, as emphasized by [22]. Furthermore, RF is considered a white-box model, allowing a straightforward interpretation of feature importance, a key aspect of this study given our focus on feature selection.

To ensure reliable performance estimates, we employed k-fold Cross-Validation. In each epoch of feature elimination, the dataset is randomly partitioned into k nearly equal folds. The Random Forest classifier is then trained on $k - 1$ folds and validated on the remaining fold. This process is repeated k times, with a different fold used for validation each time. The final accuracy is computed as the average accuracy over all folds. After cross-validation, the model is retrained on the entire training dataset to obtain the classifier that will be deployed in practice. After cross-validation, the model is retrained using the entire training dataset to obtain the classifier for deployment. Finally, an ensemble voting method is applied using the full test set to enhance prediction reliability.

5 Results

This chapter presents the results and corresponding discussions of our study, organized into 2 (two) subsections. The structure is organized as follows: the 5.1 section discusses how the loaded data from the BovDBV2 dataset was preprocessed, treated, and organized. The comparison of the results when training the model on the full feature set versus when trained using various feature selection

methods is presented in the 5.2 section, with analyses of the models and the most relevant features.

5.1 Data Preprocessing

Using the BovDbV2 database, we constructed another dataset containing the most relevant futures contracts for this study, along with their optimal intraday periods. From this dataset, we calculated the technical analysis indicators described in Subsect. 4.3. These indicators were then normalized to a range between 0 and 1, resulting in the creation of 56 new features along with the price data, forming a total of 66 features. Finally, the corresponding labels were produced using the approach detailed in Subsect. 4.2.

The selected futures contracts were WING24 and WINJ24 (Jan-Feb 2024), WINJ24 and WINM24 (Mar-Apr 2024), and WINM24 and WINQ24 (May-Jun 2024), chosen based on their high trading volumes and extensive trading activity, totaling approximately 38 to 40 active trading days per period. The data was subsequently partitioned into 3 (three) months for training and 3 (three) for testing, and these subsets were used in the subsequent analysis in the model.

5.2 Performance Evaluation: Full Feature Set vs. Feature Selection

In this section, we provide a comparative analysis of the model performance when trained with the full feature set (Case 1) versus various feature selection methods (Case 2). This approach aims to highlight how feature selection is applied to reduce dimensionality and filter noise, with the goal of improving both predictive performance and interpretability. For all experiments, we employed the Random Forest classifier, a robust ensemble model that aggregates multiple decision trees to achieve reliable accuracy and control overfitting. The model used 100 estimators and a fixed random state (42) to ensure reproducibility.

To validate the results, we adopted a 9-fold stratified cross-validation approach. Given the large number of labeled instances, 9 folds, as opposed to the more common 5 or 10, offered a balance between variance reduction and computational cost. The stratification ensured that class distributions remained consistent across folds, providing robust evaluation metrics. Furthermore, choosing an odd number of folds minimizes the potential for ties between classes during ensemble voting on the test data.

Case 1: Full Feature Set - Model Interpretation. For this case, the model was trained using the complete set of features without applying any feature selection. The data was partitioned into training and testing sets as described in Subsect. 5.1, and we employed stratified 9-fold Cross-Validation to ensure that each fold maintained representative class distributions. This stratification is crucial to reduce the variance in performance metrics such as accuracy and F1-score.

The Cross-Validation process produced an average accuracy of 0.7159 and a standard deviation of 0.0096. However, when evaluated on the test set, the final accuracy dropped to 0.5883. The classification report (Table 1) for the model is as follows:

Table 1. Classification report for training with all features

Class	Precision	Recall	F1-score	Support
Downtrend	0.62	0.35	0.45	3306
Uptrend	0.57	0.80	0.67	3592
Accuracy	–	–	0.59	6898
Macro avg	0.60	0.58	0.56	6898
Weighted avg	0.60	0.59	0.57	6898

The evaluation report indicates that the model's performance varies across classes. For the downtrend class, a precision of 0.62 shows that when the model predicts a downtrend, it is correct 62% of the time; however, a recall of 0.35 reveals that many true downtrend cases are missed, resulting in a lower F1 score of 0.45. In contrast, the uptrend class has a high recall of 0.80, meaning that the model successfully identifies most uptrend cases, although its precision of 0.57 suggests the presence of false positives, leading to an F1 score of 0.67. Table 2 reports the results of the confusion matrix.

Table 2. Confusion matrix for Full data training

	Predicted Downtrend	Predicted Uptrend
Real Downtrend	1169	2137
Real Uptrend	703	2889

The confusion matrix (Table 2) shows that the model has difficulty distinguishing "downtrend" from "uptrend." Specifically, 2,137 downtrend cases were misclassified as uptrends, while 1,169 true downtrends were missed, indicating lower sensitivity for this class. Similarly, the model misclassified 703 true uptrends as downtrends and failed to detect 2,889 cases of uptrends. These misclassifications align with the recall metrics, highlighting a bias toward predicting uptrends and potential challenges in detecting trend reversals.

Figure 3 is a bar chart representing the mean accuracy of stratified cross-validation and its corresponding result on the test set. To illustrate the variability of the results, each bar of the mean accuracy of stratified cross-validation includes standard deviation indicators, showing three deviations above and below the mean, ensuring that 99.7% of the results fall within this range.

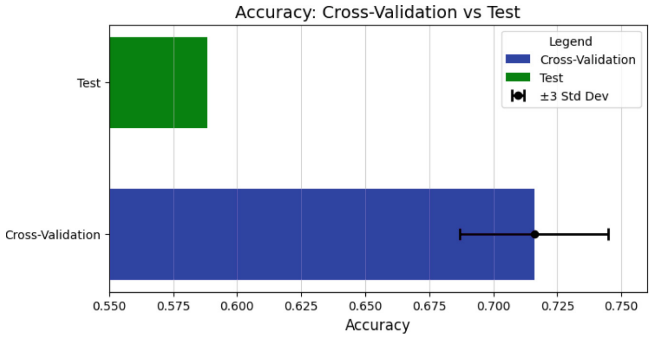


Fig. 3. Comparison between cross validation and test accuracy.

As we can see, there is a significant difference between the cross-validation accuracy and the test accuracy. The graph suggests that the model memorizes the training data and has difficulty generalizing to new data, indicating overfitting. The standard error in the cross-validation shows that the accuracy varied across different splits of the data, which may indicate that the model is learning specific quirks of the training data rather than general patterns.

Case 2: Feature Selection - Methods. In this step, we apply each feature selection method (shown in Sect. 3) to the training data. Each method returns a ranking of the features according to their estimated importance for the predictive task. The resulting ranking is then saved and used to train the Random Forest model with the same hyperparameters employed in the Full Feature Set experiment, using a 9-fold stratified cross-validation. An expected result is the identification of a reduced set of features that achieves an average accuracy comparable to that obtained with the model trained with all features, thus revealing the most significant indicators for the technical analysis of the derivatives market.

The set of Figs. 4 represents the model performance on training data using cross-validation. For each training iteration, a feature selection method is applied and evaluated by returning the average result of 9-fold cross-validation. The plots show the model performance as features are progressively removed. The Green Dot (GD) represents the local maximum, where accuracy reaches its highest value with the optimal number of k features, while the Red Dot (RD) marks the point where the model's average accuracy on the training data started to decline.

Once the initial model is trained using the ranked set of features, we implement an iterative elimination process. In each iteration, features are removed in inverse order of importance, as indicated by the selection methods. This process is performed to determine the impact of each subset of features on the accuracy of the model.

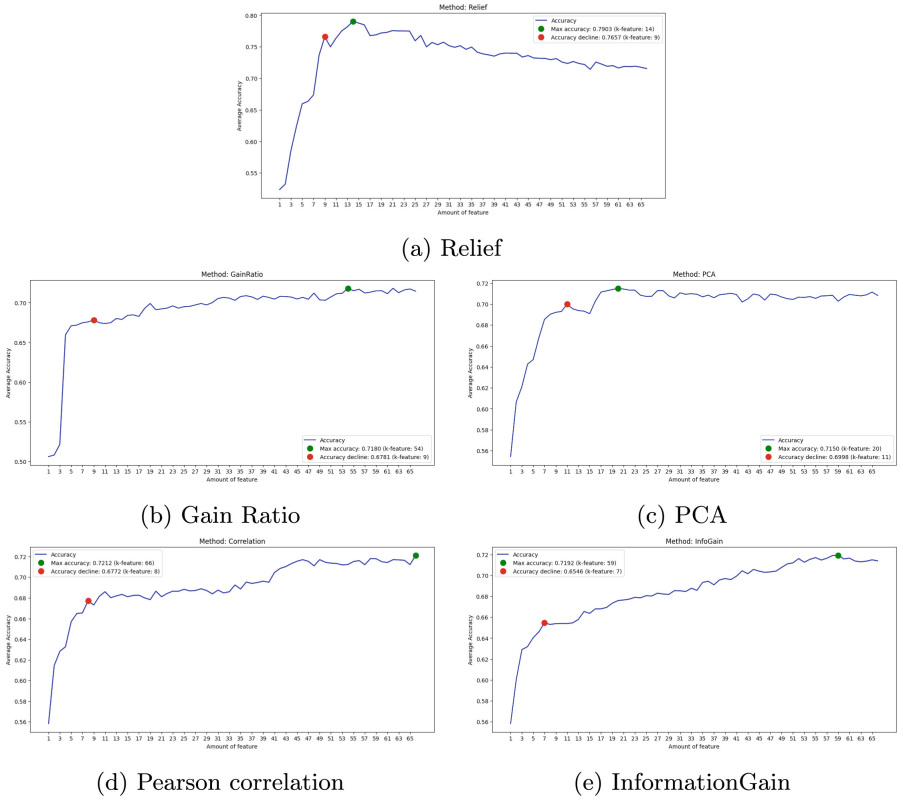


Fig. 4. Feature removal experiments.

Analysis of the Relief method in Fig. 4a shows that while it achieved an impressive average accuracy of 0.7903 in cross-validation with 14 features, demonstrating strong generalization in training, this did not translate as effectively to the test set, which had a final accuracy of 0.5657. In a more restricted setting with 9 features (red dot), validation accuracy dropped to 0.7657 and test accuracy reached 0.5626, indicating that excessive feature reduction can lead to the loss of essential predictive information.

For the next model, we will use the Gain Ratio feature ranking, following the same approach used for the Relief method. In this case, we obtained completely different results, as shown in the graph in Fig. 4b.

At this stage, the GD point indicates that, with 54 features, the cross-validation resulted in an accuracy of 0.7180. This reference point was then used to perform a stratified cross-validation. After training with the full data set, the model obtained an accuracy of 0.5752, which did not adapt as well to the test data. In contrast, the same methodology was applied to the RD point, which corresponds to the selection of 9 features. In this case, the initial validation presented an accuracy of 0.6781. When this model was evaluated with the test set,

the results demonstrated consistent performance, evidencing a balance between the classes.

Systematically, the next method discussed in this section is PCA (Principal Component Analysis), a widely used technique for dimensionality reduction. In this paper, the sequence of component embedding follows the order determined by the PCA-based feature selection method, and the accuracy plots Fig. 4c reflect the results obtained through cross-validation.

In the GD configuration, when using 20 principal components, PCA achieved a maximum accuracy of 0.7150 during cross-validation. However, when evaluating the model on the test set, the final accuracy was 0.5925. In the RD configuration, when reducing the set to 11 components, an average accuracy of 0.6998 was observed in the cross-validation. However, in the test, the final accuracy dropped to 0.5800, indicating a significant worsening using even fewer features, with the model not being able to generalize well to the test data.

Figure 4d illustrates the evolution of the average precision as a function of the number of features selected by the Pearson Correlation criterion.

The highest precision (0.712) occurs with 66 features (GD), while precision starts declining at 8 features (RD), reaching 0.6772. In the test data, GD performed worse, with 0.5865 precision and class imbalance, indicating that the high feature count may have led to overfitting, limiting generalization. For RD, precision and F1-score are more balanced across classes, suggesting a more stable model. Its final accuracy of 0.6086, higher than GD's 0.5865, implies that reducing features, though slightly lowering validation performance, improves generalization on new data.

Analysis of the Information Gain (InfoGain) method, illustrated in Fig. 4e, highlights the trade-off between complexity and generalization. On the green dot (GD) with 59 features, the model achieved an average cross-validation accuracy of 0.72, but struggled on the test set, dropping to 0.5780. Although it performed reasonably well (0.61 for downtrend, 0.57 for uptrend), its low recall (0.33) for downtrend suggests overfitting, limiting generalization. In contrast, the red dot, with only 7 features, had lower cross-validation accuracy (0.6546) but improved test accuracy (0.6163). This configuration better balanced precision and recall, with F1 scores around 0.60, indicating a more robust model with reduced overfitting. These results emphasize the importance of finding the right balance between feature quantity and model generalization.

To further our investigation and enrich this study in order to develop an optimal model with minimal overfitting, we trained four additional models using distinct feature selection techniques with different classifications, showing their training for the training data contained in Fig. 5, which includes the performance with all other methods.

In Fig. 5, we can see a comparison of how all feature selection methods in Ensemble work for the 9-fold cross-validation measure. It is guaranteed that there is a pattern in the accuracy variation throughout the removal of features. Initially, the accuracy increases to a certain point, showing that a smaller number of features increases the accuracy of the models. Then the accuracy starts

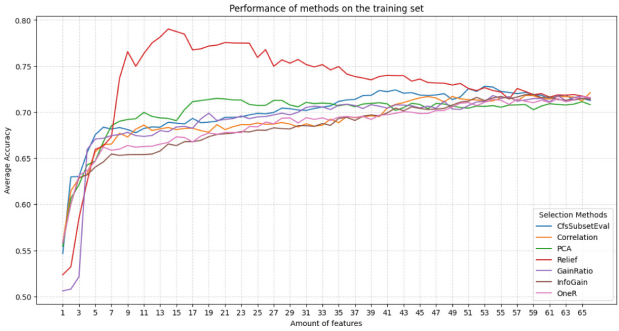


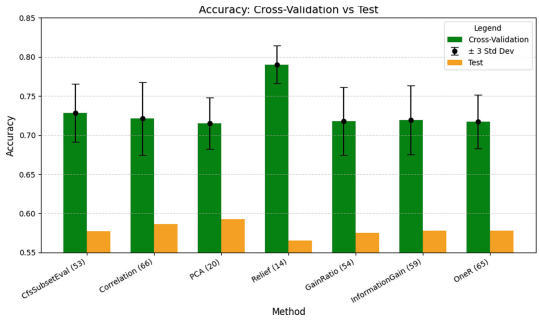
Fig. 5. Training performance with all methods.

to decrease. Some methods are able to combine a lot of features that maintain accuracy for a long period of epochs, without losing it significantly. Other methods, such as Relief, allow accuracy to drop significantly after its maximum peak, while others kill consistency as new features are removed.

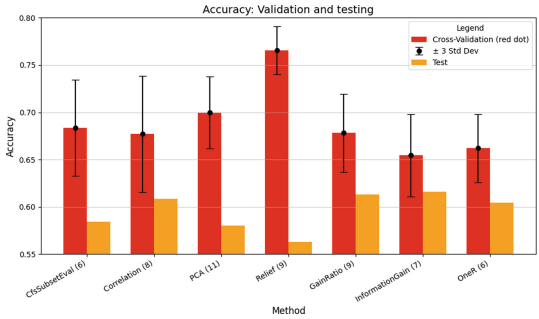
To further investigate the feature selection methods, we created a comparative bar chart (Fig. 6) that evaluates the performance of the methods at two critical points in the accuracy curve. These points were defined based on detailed analysis: the first, represented by green bars, corresponds to the optimality point (GD) of each method, while the second, shown in red bars, indicates the point at which accuracy begins to decline (RD).

Subsequently, the generated models were evaluated on a different test data set in order to measure their performance outside the validation environment. The final chart displays, for each feature selection technique, four bars: two correspond to the average accuracy obtained in the stratified cross-validation for the GD point and its respective performance on the test set, and the other two show the same indicators for the RD point. The orange bars reflect performance at the point of optimality, while the green bars highlight the moments when accuracy begins to decline. To highlight the variability in results, each validation average bar has standard deviation markers—indicating two deviations above and below the mean—ensuring that 99.7% of the results fall within this range. This graphical structure provides a clear and objective view of each method's performance at different stages of attribute reduction, also highlighting the consistency of each approach in maintaining good predictive performance under varying conditions.

Figure 6a shows the performance of each feature selection method at the point of highest accuracy (GD). At this stage, it can be seen that the green bars correspond to the average accuracy obtained through stratified cross-validation for the point where accuracy is maximum on the training data, while the orange bars indicate the result of these same models when evaluated on a separate test set. In this way, it is possible to compare how well each method adapts to the training data and how this adjustment remains or changes when exposed to data not used in the training stage. It can be seen that some methods perform



(a) Green dot accuracy in training and testing data



(b) Red dot accuracy in training and testing data

Fig. 6. Set of Figures of model performance. (Color figure online)

very well for training, but for unseen data (tests) the accuracy decreases (as for example in the Relief method), demonstrating low generalization of the model.

Figure 6b, on the other hand, shows the performance of the same feature selection techniques at a point where accuracy begins to decline (RD). The red bars indicate the average accuracy in cross-validation, and the orange bars again show the performance on the test set. This scenario illustrates how each method reacts to the use of a reduced number of features, showing whether the model maintains good generalization capacity or whether there is a sharp drop in performance outside the validation environment. The direct comparison between Figs. 6a and 6b allows us to identify which methods exhibit greater stability as features are removed, in addition to highlighting at what point each approach suffers the most significant loss of accuracy.

Among the methods evaluated, Information Gain stood out with the best test accuracy. Using seven features, the model achieved 61.63%, balancing precision and recall while minimizing the risk of overfitting. This highlights the importance of selecting a concise yet informative feature set to improve generalization.

The complexity of financial markets makes classification even more challenging, as it involves detecting subtle price changes. The dataset was structured to predict two distinct categories of price movements, which in itself makes the

Table 3. Ranking of features using Information Gain with the best k features.

Rank	Feature
1	EMA_5-3
2	EMA_5-3_normalized
3	EMA_7-3
4	EMA_7-3_normalized
5	SMA_9_normalized
6	EMA_9-3
7	EMA_7_normalized

task difficult. Furthermore, external factors directly influence financial trends, making the differentiation between periods of stability, small rallies or declines even more complex. The improved feature selection provided by Information Gain reinforces the relevance of technical indicators used in financial analysis, as demonstrated in Table 3. The analysis of the top 7 features shows that Exponential Moving Averages (EMAs) indicators, especially their short-term variations, are the most relevant for intraday trend prediction, reaffirming the importance of selecting an informative subset to reduce complexity and overfitting.

To understand the impact of this work and the proposed methods in relation to the literature, we performed a direct comparison with previous studies that also employed feature selection techniques to predict prices in the financial market.

Table 4 summarizes the main aspects of each study, including the dataset used, the model applied, the number of selection methods (FS - Feature Selection) and the accuracy achieved in the prediction tasks.

Table 4. Comparative summary of related studies.

Study	Dataset	Model	FS	Acc.
Peng [18]	SSE	RNN	3-Methods	50.0%
Nelson [17]	-	LSTM	1-Method	55,9%
Lu [15]	TSE	SVR	2-Method	57.5%
This study	B3	Radom Forest	7-Methods	**61.63%**

Comparative analysis with related studies allows us to observe differences in approaches that used complex computational models, which do not guarantee better performance if they are not accompanied by rigorous data processing and effective selection of the attributes used. It is therefore concluded that feature selection is not just a complementary step, but rather a strategic element in the construction of more interpretable, generalizable and efficient robust models for the financial market.

6 Conclusion

In this work, different feature selection techniques were explored to improve the performance of classification models applied to the derivatives market. We used data extracted from BovDBV2, which contains information on futures contracts traded on B3, and applied technical analysis indicators (such as moving averages and standard deviation) to generate the set of attributes. The results obtained by the Feature Selection techniques were ordered according to the importance attributed to each method and subsequently used to build classification models with the Random Forest algorithm, using cross-validation to ensure the robustness of the results.

The adopted approach demonstrated that, by employing feature selection, it is possible to improve the generalization capacity of the model compared to training with the full set of attributes, reducing noise and eliminating redundant variables. However, as feature reduction intensifies, a notable drop in model performance is observed, a fact that can be attributed to the removal of variables that contain essential information for the correct discrimination of classes. Therefore, it is essential to carefully balance the number of attributes to preserve the most relevant indicators - among which those identified by the Information Gain method stand out, which obtained the best accuracy of 61.63% with 7 characteristics for the Red Dot (RD) test data - demonstrating that these attributes support the robustness of the model. The article sought to fill the main gaps pointed out in the literature review, generating machine learning models that, even with certain limitations, when adopting good practices, significantly improve performance, thus providing strategies that can be adopted in the area of Stock Market Prediction.

Future work may explore the application of advanced attribute selection techniques, the integration of new indicators, as well as the evaluation of other classification algorithms, with the aim of further improving the accuracy and predictive capacity of the models in the context of the financial market.

Acknowledgments. The authors would like to thank FAPERGS (24/2551-0001396-2, 23/2551-0000773-8), CNPq (305805/2021-5) and FAPERGS/CNPq (23/2551-0000-126-8).

References

1. Adegboye, A., Kampouridis, M., Otero, F.: Improving trend reversal estimation in forex markets under a directional changes paradigm with classification algorithms. Int. J. Intell. Syst. **36**(12), 7609–7640 (2021)
2. Azhagusundari, B., Thanamani, A.S., et al.: Feature selection based on information gain. Int. J. Innovative Technol. Exploring Eng. (IJITEE) **2**(2), 18–21 (2013)
3. Bodie, Z., Kane, A., Marcus, A.: Fundamentos de investimentos. AMGH Editora (2014)

4. Cavalcante, R.C., Brasileiro, R.C., Souza, V.L., Nobrega, J.P., Oliveira, A.L.: Computational intelligence and financial markets: a survey and future directions. Expert Syst. Appl. **55**, 194–211 (2016)
5. Chiang, I., Hughen, W.K., Sagi, J.S.: Estimating oil risk factors using information from equity and derivatives markets. J. Financ. **70**(2), 769–804 (2015)
6. Christensen, H.B., Nikolaev, V.V.: Does fair value accounting for non-financial assets pass the market test? Rev. Acc. Stud. **18**, 734–775 (2013)
7. Clayton, M.J., Jorgensen, B.N., Kavajecz, K.A.: On the presence and market-structure of exchanges around the world. J. Financ. Markets **9**(1), 27–48 (2006)
8. Corrêa Cardoso, F., et al.: BovDB: a data set of stock prices of all companies in B3 from 1995 to 2020. J. Inf. Data Manag. **13**(1) (2022). https://doi.org/10.5753/jidm.2022.2345
9. Hoque, M.E., Billah, M., Kapar, B., Naeem, M.A.: Quantifying the volatility spillover dynamics between financial stress and us financial sectors: evidence from QVAR connectedness. Int. Rev. Financ. Anal. **95**, 103434 (2024)
10. Hull, J.V., Dokovna, L.B., Jacokes, Z.J., Torgerson, C.M., Irimia, A., Van Horn, J.D.: Resting-state functional connectivity in autism spectrum disorders: a review. Front. Psych. **7**, 205 (2017)
11. Karegowda, A.G., Manjunath, A., Jayaram, M.: Comparative study of attribute selection using gain ratio and correlation based feature selection. Int. J. Inf. Technol. Knowl. Manag. **2**(2), 271–277 (2010)
12. Kumar, M., Thenmozhi, M.: Forecasting stock index movement: a comparison of support vector machines and random forest. In: Indian Institute of Capital Markets 9th Capital Markets Conference Paper (2006)
13. Li, J., et al.: Feature selection: a data perspective. ACM Comput. Surv. (CSUR) **50**(6), 1–45 (2017)
14. Long, J., Chen, Z., He, W., Wu, T., Ren, J.: An integrated framework of deep learning and knowledge graph for prediction of stock price trend: an application in Chinese stock exchange market. Appl. Soft Comput. **91**, 106205 (2020)
15. Lu, C.J., Lee, T.S., Chiu, C.C.: Financial time series forecasting using independent component analysis and support vector regression. Decis. Support Syst. **47**(2), 115–125 (2009)
16. Mitchell, J.C., Mitchell, M., Stern, U.: Automated analysis of cryptographic protocols using MUR/SPL PHI. In: Proceedings of 1997 IEEE Symposium on Security and Privacy (Cat. No. 97CB36097), pp. 141–151. IEEE (1997)
17. Nelson, D.M., Pereira, A.C., De Oliveira, R.A.: Stock market's price movement prediction with LSTM neural networks. In: 2017 International joint conference on neural networks (IJCNN), pp. 1419–1426. IEEE (2017)
18. Peng, Y., Albuquerque, P., Kimura, H., Saavedra, C.: Feature selection and deep neural networks for stock price direction forecasting using technical analysis indicators. Mach. Learn. Appl. **5**, 100060 (2021)
19. Power, A., Burda, Y., Edwards, H., Babuschkin, I., Misra, V.: Grokking: Generalization beyond overfitting on small algorithmic datasets. arXiv preprint arXiv:2201.02177 (2022)
20. Pring, M.J.: Study guide for technical analysis explained, vol. 5. McGraw-Hill New York (2002)
21. Sabatier, P.A.: Top-down and bottom-up approaches to implementation research: a critical analysis and suggested synthesis. J. Publ. Policy **6**(1), 21–48 (1986)
22. Soto-Osorio, D., Sidorov, G., Chanona-Hernández, L., López-Ramírez, B.C.: Identification of scientific texts generated by large language models using machine learning. Computers **13**(12) (2024). https://doi.org/10.3390/computers13120346

23. Souza, A.S., Lucca, G., Borges, E.N., Cardoso, F.C., Dalmazo, B.L., Berri, R.: Dataset for Intraday Analysis of B3 stock prices (2024). https://doi.org/10.7910/DVN/TMB4IG
24. Venkatesh, B., Anuradha, J.: A review of feature selection and its methods. Cybern. Inf. Technol **19**(1), 3–26 (2019)
25. Wang, J., Ma, F., Bouri, E., Zhong, J.: Volatility of clean energy and natural gas, uncertainty indices, and global economic conditions. Energy Econ. **108**, 105904 (2022)