Testing the Application of Support Vector Machine (SVM) to Technical Trading Rules

André R. Fonseca*, Michel C. R. Leles*, Mariana G. Moreira*, Adriano S. Vale-Cardoso*, Marcos V. L. Pereira*, Elton F. Sbruzzi†, Cairo L. Nascimento Jr.† (IEEE Member)
* Universidade Federal de São João del-Rei, Ouro Branco, Brazil.
† Instituto Tecnológico de Aeronáutica, São José dos Campos, Brazil.
Emails: andre.fonseca@live.com, {mleles,marianageny,adrianosvc, marcos.vinicius}@ufsj.edu.br, {elton, cairo}@ita.br

2021 IEEE International Systems Conference (SysCon) | 978-1-6654-4439-2/20/$31.00 ©2021 IEEE | DOI: 10.1109/SysCon48628.2021.9447068

***Abstract*—The stock price movements result from many factors that are often difficult to be detected and modelled. The investigation of price trends and the use of the information available to evaluate investments and identify trading opportunities can be promising. However, financial data are non-stationary, i.e., their statistical characteristics constantly change. Therefore, the financial market is a challenging environment for the application of Machine Learning techniques, since they can only make reliable predictions to data consistent with what they have seen before. This paper test the use of a Machine Learning technique known as Support Vector Machines (SVM) aiming at being a tool to support the decision making process of trading at the stock market. SVM aggregates some input signals and, based on a set of technical indicators and historical price changes, create buy/sell recommendations of a given security as outputs. The dataset comprises several Brazilian stocks time-series traded on both the Brazilian (B3) and American (NYSE) stock exchange. These time-series belong to various economic sectors and present different market dynamics. The computational simulations are based on a fictitious strategy that does not consider the trading costs and only long positions are allowed. Using two risk-adjusted performance metrics, the results show that strategies based on the SVM model achieve better performance than the Buy & Hold benchmark.**

***Index Terms*—Machine Learning, Financial Time Series, Stock Market, Support Vector Machines, Supervised Learning.**

I. Introduction

Price changes in the financial market is based on factors that are difficult to be identified and modelled. Even though, in the recent literature, some papers has proposed to analyse possible components that may influence this change. Bollen et al. [1] argue the analysis of external factors, by stating that both information and emotions play a significant role in decision making at similar level the financial market.

The well known Efficient Market Hypothesis (EMH) argues that prices are not based on past data analysis, but absorbing information instanter when this is produced [2]. However, the advances of computing in recent decades opened new avenues to improve the gains and decision making when trading in the stock market, serving as a mechanism that can counter EMH. Different algorithms have been used in activities related to the analysis and classification of information, and also in the execution of trading strategies [3]. In this way, practitioners can use statistical means with the resources and computational models of machine learning to observe and interpret market behaviours with a certain accuracy, creating efficient trading strategies.

Roughly, trading strategies (rules) consist on a list of functions or operators which take a price series as its input and computes an outcome (based on some parameters), that is converted into a market order (to buy or to sell a particular stock) [4–6]. Therefore, the same trading strategy can exhibit a collection of different results according to the provided parameters. Several trading strategies have been proposed in the literature, some of them based on classic indicators, such as moving averages and relative strength index [7]. Recently, different authors have combined those classic indicators to artificial intelligence methods [8], and have showed an overall trading performance superior to the previous ones presented in the literature. For example, A multi-criteria trading system has been proposed by Leles et al. [9].

Promising results are presented in works using Support Vector Machine - SVM - as a classification technique in financial series [10]. Kim [11] applies SVM to forecast the KOSPI Index, and compares its results to neural networks. This author observes that the incorrect selection of parameters can lead to over-fitting, and shows that the use of the SVM method surpasses neural networks in forecasting the future values of the index.

In the present work, it is proposed a signal-based stock trading strategy that uses SVM, aiming at being a tool to support the decision making process of trading at the stock market. The dataset comprises several Brazilian stocks time-series traded on both the Brazilian (B3) and American (NYSE) stock exchange. These time-series belong to various economic sectors and present different market dynamics. Using two risk-adjusted performance metrics, the results show that strategies based on the SVM model achieve better performance than the Buy & Hold benchmark.

II. Theoretical background

A. Financial Analysis

This is necessary to analyse the events and information available in order to build an investment strategy in the financial market. The types of analysis can be split into twofold: fundamental analysis and technical analysis. The fundamental analysis requires multidisciplinary knowledge, as it involves a careful assessment of a list of different information that includes: the target company, the sector in which the company operates, the sectors that may influence the company reputation, the news, the balance sheet and the economic scenario. The strategies developed based on this analysis aim the returns after longer periods. On the other hand, the technical analysis uses the interpretation and the observation of historical prices and volumes to forecast the short-term changes future prices, by the creation of different indicators, which are known as technical indicators.

In the financial market, a data series regarding to a particular stock is generated and made available at the end of a trading period. These data series are composed by the following information: the opening price, which represents the price at which the first deal was closed in the period; the closing price, which represents the price at which the last deal was closed in the period; the minimum price, representing the lowest price that the asset was traded in the period; the maximum price, which represents the highest price traded for the asset in the period; and the volume, as the amount of stocks that were traded in this period.

B. Time series and the dataset

Time series are datasets of observations of a quantitative variable ordered in time, and can detect standardized behaviours, including price changes. These sets can be representations of causal systems that usually create non-random patterns, and can be detected by analysing their graphs more efficiently with statistical tools.

This work analyses the time series of the following Brazilian companies: Petrobras, Itaú, Bradesco, Vale, Eletrobras, JBS, B3, CSN, CPFL and Engie. The period comprises July 29, 2015 to June 26, 2020. These companies belongs to the Forbes global list which includes the 20 largest Brazilian publicly traded companies, according to metrics such as assets, profit, sales and market value.

C. Technical Indicators

The primary tools for investors who operate in the financial market using technical analysis are technical indicators. The aim of these indicators is to improve the efficiency of the analysis of a financial time series, giving a specific view of the price changes to the investor [12, 13]. They are great tools to create variables that can be used in models based on machine learning techniques.

The mathematical model of a technical indicator can be separated by overlap or moment. Currently, there are too many indicators available, and an even greater number of combined indicators to help in the technical analysis process whether a particular trading strategy is employed. However, part of them are extensions of models that have already been consolidated, such as moving averages, Bollinger bands and SAR.

The indicators used in our model were created through the package TA-Lib [14] which contains approximately 70 of the most used technical indicators. The description and the configuration its parameters are the following [7]:

- Relative Strength Index (RSI): an oscillator used to measure the strength of a trend in reversal signals based on peaks and averages of n days over a period of time. The scale ranges between 0 and 100%;
- Simple Moving Average (SMA): the simplest indicator. It is an overlapping function with the advantage of eliminating noise from the time series. This is possible because the arithmetic mean is calculated within n days over a given period of time. The longer the period of an SMA, the smoother the curve;
- Stop and Reverse (Parabolic SAR): this indicator shows whether the market is in a downward or upward trend. It resembles moving averages, but has an acceleration coefficient. It changes its direction according to the trend captured, and its length depends on the price changes. The setting used in this work is 0.2 for the acceleration factor and 0.2 for the maximum point;
- Directional Movement Index (ADX): a momentum indicator that aims to show the strength of a trend and points out future price changes. The strength may be classified as positive or negative one, by ranging from 0 to 100%.

D. Support Vector Machine - SVM

SVM is a set of supervised machine learning methods proposed by Cortes and Vapnik [15]. These methods use a linear model to implement a class separation through a nonlinear mapping of the input data sets into a high-dimensional space. It is used in situations that require pattern recognition and classification by data analysis, being considered a learning model based on instances, which seeks to use the similarities between those instances observed in the training data through the features to make a prediction with unseen data model (test data). Hence, this is a memory-based learning method. It can also be extended to regression problems [16, 17] which is based on calculating a regression function.

SVM can be explained from the definition of the maximum margin classifier. The data used in the classification problem form a multidimensional space, as the example shown in Equation 1.

$$w_0 + (w_1 X_1) + \ldots + (w_n X_n) = w_0 + \sum_{i=1}^{N} w_i x_i = \vec{y} \quad (1)$$

where $\vec{x} = (X_1, \ldots, X_n)$ is an input data vector, $\vec{w}$ is the normal vector/weights of the hyperplane, $\vec{y} \in \{-1,1\}$ is a data set with two classes and N the number of samples used in the model [18, 19]. For this work, $\vec{x}$ represents the features that feed the model, being composed of technical indicators and return functions. Table I present in Section III-B describes such features. The vector y represents the trading signals, where -1 indicates selling and 1 indicates buying a particular stock. The coefficients w_i determine the slope and intersection of the hyperplane, and they are learnt by the SVM algorithm. A hyperplane has the dimension $n-1$ with regards to the space generated by the data set. The maximum margin classifier displaces the hyperplane in order to obtain the greatest distance between the extreme points of each class, forming the definition of maximum margin. Only the extreme points of each class are relevant for calculating the margin distance. These resulting data points are called support vectors.

1) Soft margin: If no data points are allowed in the margin areas, this type of linear classification is known as hard margin classification. This might lead to a narrow margin, in which the model will be extremely sensitive to noisy data points. On the other hand, the concept of soft margin is introduced to avoid the overfitting. In order to reach a larger margin, the soft margin allows some samples to be misclassified. Therefore, there is a trade-off between the margin width and errors in the training process.

2) Kernel function: In its basic design, SVM is a linear classifier that uses a hyperplane to divide the data set into smaller classes. However, many of the real problems cannot be separated by a linear [18] hyperplane. One alternative to deal with this issue is to modify the way the hyperplane works through a kernel. Cover's theorem states that if a data set is designed in a high-dimensional space, it is more likely to be separated linearly than in a small-space [20]. A kernel can be understood as a function that corresponds to the internal product of some expanded space, making this mapping implicitly using a technique known as kernel trick, which basically provides a proxy value for the internal product where the data were actually projected at a higher dimension.

E. Evaluation and performance metrics

The aim of the performance metrics is to evaluate and/or validate the model, since only the return generated should not be the key factor in the efficiency analysis of a trading strategy [21]. The results help to draw conclusions whether a satisfactory model is up to be used in real applications.

1) Classification Metrics: For problems involving data classification, it is useful to test the efficiency of the model for information based on the frequencies of the classifications.As for the data, it is important to consider a target class. For this work, there are two classes: -1, which indicates sale the stock, and 1, which indicates buy the stock. The metric TP refers to the number of true positives, where a sample is correctly predicted to belong to the target class, TN indicates the true negatives, where a sample belongs the other class is correctly predicted to belong to this other class, FP as false positives, where a sample belongs to the other class and it is incorrectly predicted to belong to the target class, and FN as false negatives, where a sample belongs to target class is incorrectly predicted to belong to another class.

- Accuracy relates the total of correct forecasts for all classes to the sum of correct and incorrect forecasts for all classes. It is given by the equation 2.

$$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

- Precision is the proportion of the correct classifications of the target class in relation to the samples that were predicted to belong to that class. It is given by the equation 3.

$$\text{precision} = \frac{TP}{TP + FP} \tag{3}$$

- Recall is the ratio of the correct classifications of the target class to all sample predictions that belong to the target class. It is given by Equation 4.

$$\text{recall} = \frac{TP}{TP + FN} \tag{4}$$

2) Sharpe ratio: Sharpe ratio Sharpe [22] measures the performance of an investment through its risk-return ratio compared to a risk-free securities. Examples of risk-free securities in the Brazilian financial market are those indexed to the CDI (bank deposit certificate) rate, which is being considered the benchmark for the calculation of the Sharpe ratio in this work. The higher the value of the ratio, the better the return of the analysed strategy. It can be calculated using the Equation 5 [21, 22].

$$SR = \frac{\mu - r_f}{\sigma} \tag{5}$$

where μ is the average expected return on an given stock, strategy or portfolio; r_f is the rate of a risk-free sectutity and σ is its standard deviation of the stock returns, which can be interpreted as the stock volatility in which the strategy was employed.

3) Maximum Drawdown: The maximum drawdown, or the maximum loss, is a measure of the decline in a historical peak for a variable. It is useful to analyse stocks. Its calculation can be done using the Equation 6, with P and L representing the previous peak of the return in the period and the current return, respectively.

$$MaxDD = \frac{P - L}{P} \cdot 100 \tag{6}$$

4) Annualised return: The annualised return is the return from any period represented in terms of an year as shown in Equation 7:

$$r_{(a.a)} = (1 + r_{cum})^{\frac{1}{k}} \tag{7}$$

where k is the time in years of the investment, r_{cum} is the return on investment accumulated over the entire period and $r_{(aa)}$ is the annualised return investment.

5) Calmar Ratio: Calmar ratio Young [23] is a ratio of the annualized real return of a strategy and the maximum drawdown. The higher the value of this ratio, the better the return on investment risk. Usually, this ratio employ a period of 36 months for calculation and monthly update. The purpose is to reduce the effect of natural short-term market volatility. Equation 8 shows the calculation of the Calmar ratio:

$$Calmar = \frac{r_{(a.a.)} - r_{f(a.a.)}}{MaxDD} \tag{8}$$

where $r_{(a.a.)}$ is the annualised rate of return on the stock under analysis; $r_{f(a.a.)}$ is the annualised rate for a risk-free investment and $MaxDD$ is the maximum drawdown for the period.

III. Proposed Approach

SVM can be used for classification or adapted for regression, and both lead to a model optimisation. SVM is a non-probabilistic method whether this is applied to the classification, because this method produces labels for the classes instead of probability of occurrence of a particular class. In this work, these classes refer to the upward or downward trends of the analysed stock, and the SVM model is responsible for finding characteristics in the data to perform the classification.

A. Data extraction, treatment and visualization

For financial applications, there are several sources and procedures for collecting and organising data. In this work, the programming language `Python` was used, which due to its versatility, besides having a large and active community, has a varied amount of tools for collecting and manipulating data from different sources. The data with the historical quotations of a given stock were collected from the Yahoo Finance.

B. Data Set Treatment

For the training and tests to be carried out by the model, it is necessary to reorganise the data according to the need. This includes the creation of features variables that will be used by the model, and are shown together with the periods in Table 1. As part of the features used by the model, the calculation of log - return, defined by Equation (9) as below:

$$r_{log} = ln\left(\frac{price_t}{price_{t-1}}\right) \tag{9}$$

One of its main advantages over the linear return is that the sum of several log - returns in a given period is the log - return for the period. This column will receive both positive and negative values between days. The periods of the indicators followed values that are commonly seen in works that implement and test the use and efficiency of technical indicators in trading strategies.

TABLE I: SVM Model: Features

Feature	Period (days)
RSI	15
$\text{SMA}_{\text{price}}$	6; 14; 20
$\text{SMA}_{\text{volume}}$	6; 14; 20
SAR	–
ADX	14
r_{log}	1; 5; 10; 20

Table I shows the features of our SVM model. The first column is the feature as detailed in Section II and the second column are the period in term of days used as parameters.

We create a column in the training data that will serve for the model to interpret the correspondences between the features in each sample and predict an output that would forecast future changes in the stock price. The reason is that our model tracks price trends of the asset under analysis. Therefore, the first column named y receives trading signals according to Equation 10 as below:

$$y = \begin{cases} -1, & \text{if price}_{t+1} < \text{price}_t \rightarrow \text{sell} \\ +1, & \text{if price}_{t+1} \geq \text{price}_t \rightarrow \text{buy} \end{cases} \tag{10}$$

The second column of signs called y_{pred} is filled with the signals provided by the model and follows the same interpretation as the column y. The data is adequate to be inserted in the model and generate the forecast column y_{pred} With these columns.

C. Application of the Model and Performance Metrics

Considering the procedure mentioned by Chang et al. [24] to build an SVM model, the data set with the columns of features and trading signals was divided between training and testing, these sets being represented by an array of real numbers. We used the $80-20\%$ split ratio, which is common in the development of machine learning models and the training set is input to the SVM classifier of the scikit-learn [25] library in order to be adjusted to these data and later receiving the test set, so that the column of prediction signals is generated.

The features have been standardized with all columns having an average 0 and standard deviation 1, and apply to training and test sets. For this pre-processing of the data, the pre-processing package from the scikit-learn library was used. This is necessary for the entire data set because SVM algorithms are sensitive to the scale of features. Standardization prevents features whose scales are greater than the others from having greater influence than the smaller ones.

Figure 1 shows the diagram of the method. Notice that we implement the walk forward method for updating the model in order to the following: (1) carry out the training of a whole historical series, (2) retrain each period to make the forecast throughout the series, (3) validate the proposed strategy and be able to capture new market conditions using the Henrique et al. [26] model.

1) Selection of parameters C and γ: We decide in favor of RBF kernel for the SVM model. This kernel takes into account two parameters: C and γ. The C parameter penalises classification errors along the hyperplane's decision edge. The higher its value, the more correctly the classification is made and the better the accuracy of the model becomes. The γ parameter is intrinsic to the kernel, and defines the influence of a sample on the classifier, and the higher its value, the closer the samples to be affected.

The method used to select the parameters is grid search [24]. At each step of the walk forward method, the SVM model is run for each element in a parameters matrix (C, γ), and optimize the training according the parameters. The default values for the parameters to be tested were input with exponential spacing, with the parameter $C = 2^{-5}, \ldots, 2^{15}$, and the parameter $\gamma = 2^{-15}, \ldots, 2^{3}$.

2) Strategy Execution and Return: The model generates the y_{pred} vector with the trading signals described by Equation 10. This vector is used to calculate the return of the strategy in the analysed period. The return is calculated by multiplying the predicted signals with the daily return where the buy or sell trade was supposed to take place, following Equation 11.

The results present a fictitious strategy that does not consider the trading costs involved in financial negotiations. Such costs can vary from broker to broker, and usually include brokerage fees, fees and taxes that also vary over time, besides specific exemptions for certain stock profit/sale amounts.

$$r = y_{pred} \cdot r_{daily} \tag{11}$$

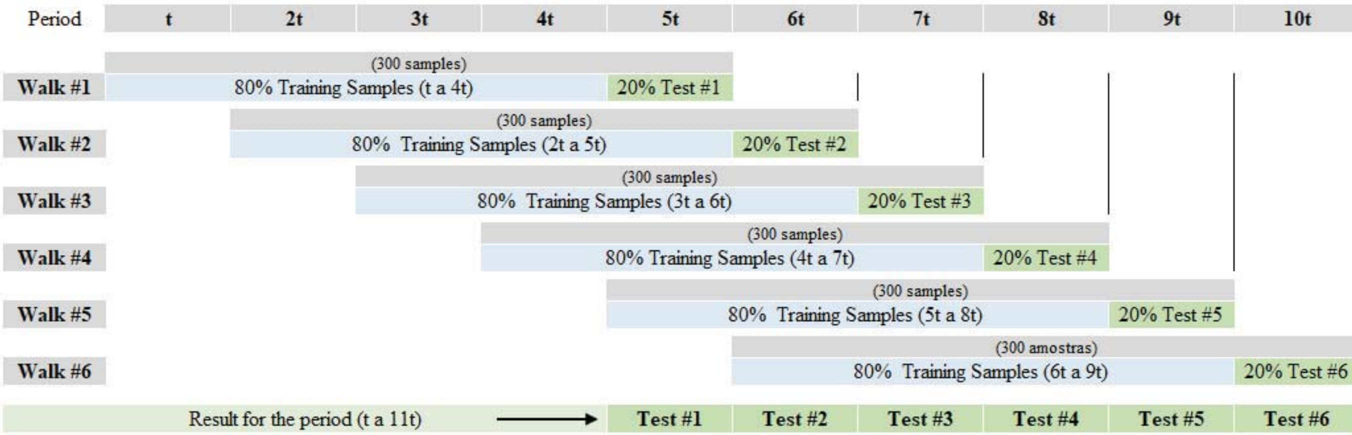

Fig. 1: Diagram of the walk forward update method.

The strategy does not consider short position, an operation in which the investor can sell stocks that he/she does not owned. Thus, only long positions are allowed in the experimental results presented in the sequel.

IV. Experimental Results and Discussion

Table II shows the results of annualised returns for both strategies. We assume that this an appropriate measure of performance and strategy validation.

TABLE II: Annualized Returns for SVM and Buy and Hold Strategies.

Stock	SVM Returns (%)	B&H Returns (%)
Petrobras	17.16	14.18
Itaú	13.85	4.22
Bradesco	26.24	2.98
Vale	24.01	17.00
Eletrobras	32.91	5.59
JBS	35.59	18.95
B3	53.61	31.65
CSN	6.91	0.44
CPFL	2.58	6.82
Engie	16.52	6.65

The results of the SVM strategy is superior to the results of the buy and hold strategy for 9 out of the 10 stocks. It is also worth mentioning that the period of analysis includes a sharp drop in stock exchanges in 2020 due to the pandemic situation.

Figure 2 shows the results of the two strategies for each series. Despite the lower performance of CPFL, the SVM strategy reduced the effect of the fall of 2020, as well as Bradesco, Vale and JBS. This behaviour in a period of high volatility can be understood as positive for the strategy, because of the losses reduction, even though CSN, Petrobras and Eletrobras drops were similar to the buy and hold strategy.

Table III shows the values of the Sharpe and Calmar ratios for both strategies. These ratios reflect how workable the return on investment is in view of the risk the investor is exposed. Similar to the returns present in Table II, the only stock with the worst indexes in the SVM strategy was CPFL. The others, such as B3 and JBS, presented attractive indexes when using the SVM strategy.

TABLE III: Calmar and Sharpe ratios of SVM and buy and hold strategies.

Stock	Calmar		Sharpe	
	SVM	B&H	SVM	B&H
Petrobras	0.280	0.223	0.356	0.261
Itaú	0.495	0.087	0.515	0.128
Bradesco	0.722	0.052	0.825	0.079
Vale	0.723	0.290	0.592	0.334
Eletrobras	0.581	0.092	0.695	0.097
JBS	1.214	0.354	0.852	0.032
B3	2.360	0.741	1.446	0.720
CSN	0.155	0.006	0.151	0.008
CPFL	0.074	0.170	0.123	0.245
Engie	0.580	0.207	0.777	0.249

Table III indicates that SVM strategy generate a lower risk of return to 9 for every 10 stocks analysed. In general, the risk of investing in that stocks by using the SVM strategy is inferior to the risk of the strategy buy and hold.

Table IV shows that the SVM strategy presented falls inferior to buy and hold strategy for all the analysed stocks. Even though exhibiting lower returns and ratios in Tables II and III respectively, CPFL's fall by using SVM strategy is inferior to buy and hold strategy. These results suggest that the strategy is efficient in reducing the risks involved when investing in the analysed stocks.

Notice that some industries are more susceptible to certain scenarios of financial difficulty than others. The oil and gas and steel industry were the most affected one, with Petrobras and CSN exhibiting the biggest falls in the period for the buy and hold strategy.

The results of Tables II, III and IV demonstrates that the performance of SVM strategy is superior to buy and hold strategy. However, as shown in Table V, the accuracy of

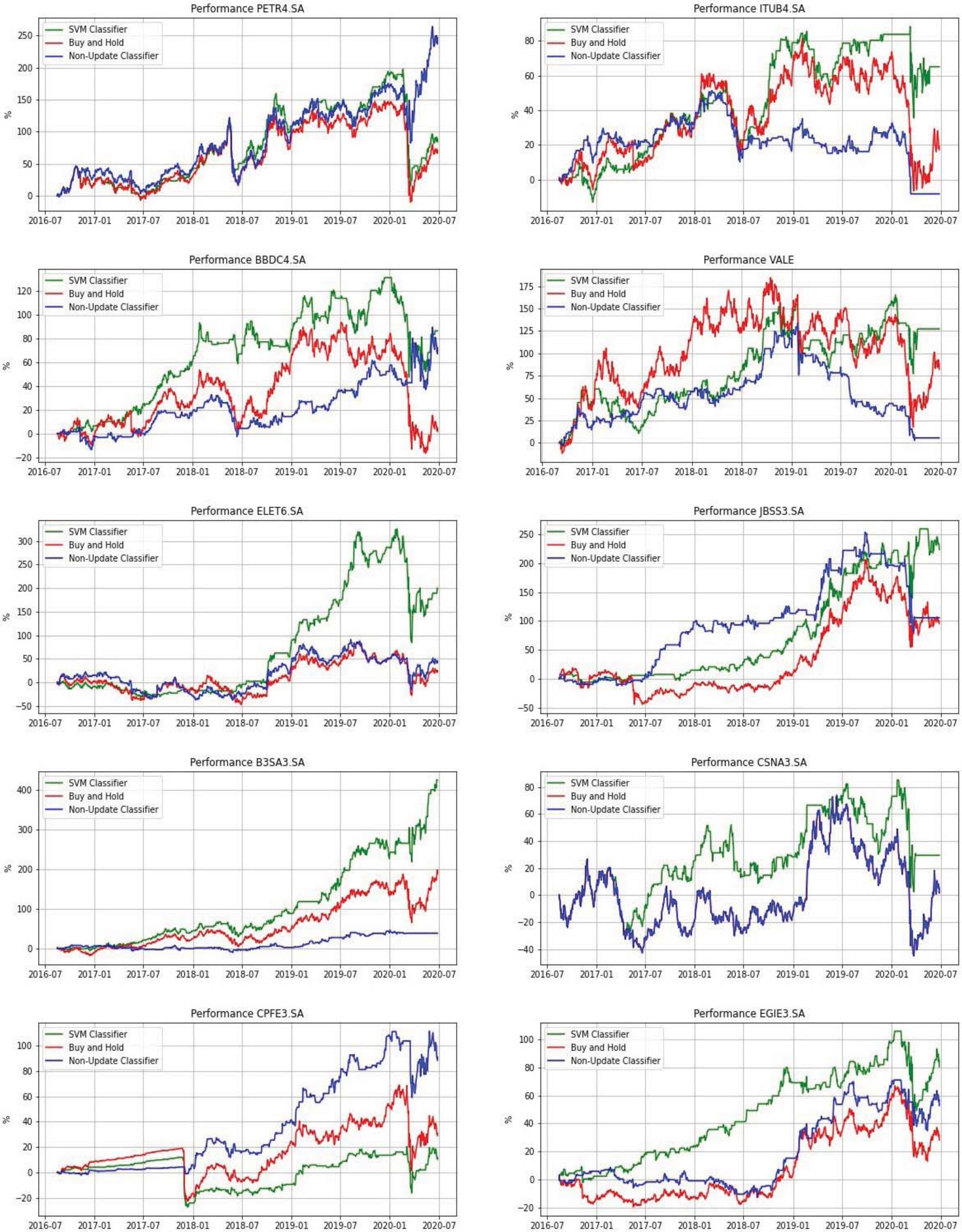

Fig. 2: Performance Comparison between SVM and Buy and Hold Strategies.

the SVM model is not attractive, reiterating the notion of instantaneous pricing in the financial market. This table also shows the values of precision and recall. The target class was considered to be the forecast of the days when the stock price would increase, that is, the forecast of the value of 1, according to Equation 10.

Table VI provides the number of trades for each series, the number of days with buy order and their respective percentage in terms of the analysis period of 960 days. In this context, trade is considered an entry or exit of a position, that is, a buy or sell of the stock. On average, a buy order based on SVM strategy was input approximately after 10 days .

TABLE IV: Maximum Drawdown of SVM and Buy and Hold Strategies.

Stock	SVM (%)	B&H (%)
Petrobras	61.23	63.55
Itaú	27.97	48.30
Bradesco	36.33	56.99
Vale	33.22	58.69
Eletrobras	56.68	60.91
JBS	29.32	53.50
B3	22.72	42.69
CSN	44.64	68.17
CPFL	34.83	40.05
Engie	28.47	32.13

TABLE V: SVM Model Metrics.

Stock	Accuracy (%)	Precision (%)	Recall (%)
Petrobras	53	53	73
Itaú	52	52	56
Bradesco	54	54	54
Vale	54	55	61
Eletrobras	55	55	60
JBS	52	51	49
B3	54	54	65
CSN	53	51	60
CPFL	52	52	50
Engie	54	55	49

Table VII displays the results of the two models for comparison, with and without updating (retraining) at each iteration of the walk forward procedure.

Financial data are non-stationary, i.e., their statistical characteristics constantly change. However, Machine Learning algorithms can only predict things consistent with what they have seen before. Therefore, the financial market is a challenging environment for the application of Machine Learning techniques [27].

In this regard, the methodology that updates its parameters periodically was able to better adapt to the market dynamics in 8 out of 10 time-series.

TABLE VI: Number of trades and days with stock in the portfolio.

Stock	Trades	Days	% of Total
Petrobras	190	585	60.9
Itaú	184	428	44.6
Bradesco	194	389	40.5
Vale	196	454	47.3
Eletrobras	166	446	46.5
JBS	178	360	37.5
B3	164	511	53.2
CSN	202	445	46.4
CPFL	212	356	37.1
Engie	230	319	33.2

TABLE VII: Annualized returns from SVM with and without updating in each iteration of the walk-forward test.

Stock	Update (%)	Non-Update (%)
Petrobras	17.16	36.98
Itaú	13.85	-2.17
Bradesco	26.24	14.12
Vale	24.01	1.43
Eletrobras	32.91	9.84
JBS	35.59	20.53
B3	53.61	8.67
CSN	6.91	0.44
CPFL	2.58	17.80
Engie	16.52	11.60

Figure 2 depicts the Buy & Hold time-series compared to the two SVM approaches used, with and without updating the SVM parameters at each iteration of the Walk Forward procedure. By analysis of this figure, one can observe the different market dynamics presented in these time-series.

V. Conclusion and Future Work

This study discussed the use of a Machine Learning technique known as Support Vector Machines (SVM) aiming at being a tool to support the decision making process of trading at the stock market. In general, the SVM strategy achieved superior results when compared to the Buy & Hold strategy, by means of two risk-adjusted performance metrics, for 10 Brazilian stocks with different market dynamics.

In addition, it was shown that setting the weights of the SVM algorithm periodically (at each iteration of the Walk Forward test) proved to be a more appropriate strategy than keeping the weights fixed. This result goes inline with the non-stationarity in financial time series.

Information plays a key role in modern finance. Market practitioners are exposed to an accelerating amount of new facts, data and statistics. For future works, the analysis of high frequency data can give the model greater robustness and versatility in order to create new trading strategies. Some indicators of Fundamental Analysis can also be taken into account, which might improve our analysis.

Acknowledgments

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan V GPU used for this research.

References

[1] J. Bollen, H. Mao, and X. Zeng, "Twitter mood predicts the stock market," *Journal of Computational Science*, vol. 2, no. 1, pp. 1–8, 2011.

[2] E. F. Fama, "Efficient capital markets: A review of theory and empirical work," *The Journal of Finance*, vol. 25, no. 2, pp. 383–417, 1970.

[3] M. Chlistalla, "High-frequency trading: Better than its reputation?" 2011.

[4] A. W. Lo, H. Mamaysky, and J. Wang, "Foundations of technical analysis: Computational algorithms, statistical inference, and empirical implementation," *The Journal of Finance*, vol. 55, no. 4, pp. 1705–1765, 2000.

[5] M. C. R. Leles, E. F. Sbruzzi, C. L. Nascimento Júnior, and J. M. P. Oliveira, "Trading Switching Setup Based on Reinforcement Learning Applied to a Multiagent System Simulation of Financial Markets," in *2019 Annual IEEE International Systems Conference*. IEEE, 2019, pp. 1–8.

[6] M. C. R. Leles, E. F. Sbruzzi, C. L. Nascimento, and J. M. P. Oliveira, "A MatLab Computational Framework for Multiagent System Simulation of Financial Markets," in *2019 IEEE International Systems Conference (SysCon)*. IEEE, 2019, pp. 1–8.

[7] R. W. Colby, *The Encyclopedia of Technical Market Indicators*, 2nd ed. McGraw-Hill, 2003.

[8] R. C. Cavalcante, R. C. Brasileiro, V. L. Souza, J. P. Nobrega, and A. L. Oliveira, "Computational intelligence and financial markets: A survey and future directions," *Expert Systems with Applications*, vol. 55, pp. 194–211, 2016.

[9] M. C. R. Leles, L. A. Mozelli, E. F. Sbruzzi, C. L. Nascimento Júnior, and H. N. Guimarães, "A multicriteria trading system based on singular spectrum analysis trading rules," *IEEE Systems Journal*, 2019.

[10] W. Huang, Y. Nakamori, and S.-Y. Wang, "Forecasting stock market movement direction with support vector machine," *Computers & operations research*, vol. 32, no. 10, pp. 2513–2522, 2005.

[11] K.-j. Kim, "Financial time series forecasting using support vector machines," *Neurocomputing*, vol. 55, no. 1-2, pp. 307–319, 2003.

[12] M. C. R. Leles, L. A. Mozelli, and H. N. Guimarães, "New trend-following indicator: Using SSA to design trading rules," *Fluctuation and Noise Letters*, vol. 16, no. 2, p. 1750016, 2017.

[13] M. C. R. Leles, L. A. Mozelli, C. L. Nascimento Júnior, E. F. Sbruzzi, and H. N. Guimarães, "Study on Singular Spectrum Analysis as a new technical oscillator for trading rules design," *Fluctuations and Noise Letters*, vol. 17, no. 4, p. 1850034, 2018.

[14] "TA-Lib: Python wrapper for TA-Lib," (accessed November 13, 2020). [Online]. Available: `https://mrjbq7.github.io/ta-lib/`

[15] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[16] H. Drucker, C. J. Burges, L. Kaufman, A. J. Smola, and V. Vapnik, "Support vector regression machines," in *Advances in Neural Information Processing Systems*, 1997, pp. 155–161.

[17] A. J. Smola and B. Schölkopf, "A ttutorial on support vector regression," *Statistics and Computing*, vol. 14, no. 3, pp. 199–222, 2004.

[18] J. Platt, "Sequential minimal optimization: A fast algorithm for training support vector machines," *Advances in Kernel Methods-Support Vector Learning*, 1998.

[19] S. Madge and S. Bhatt, "Predicting stock price direction using support vector machines," *Independent Work Report Spring*, 2015.

[20] T. M. Cover, "Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition," *IEEE Transactions on Electronic Computers*, no. 3, pp. 326–334, 1965.

[21] M. L. Prado, *Advances in Financial Machine Learning*. John Wiley & Sons, 2018.

[22] W. F. Sharpe, "Mutual fund performance," *The Journal of Business*, vol. 39, no. 1, pp. 119–138, 1966.

[23] T. W. Young, "Calmar ratio: A smoother tool," *Futures*, vol. 20, no. 1, p. 40, 1991.

[24] C.-C. Chang, C.-W. Hsu, and C.-J. Lin, "A practical guide to support vector classification," 2003.

[25] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, and V. Dubourg, "Scikit-learn: Machine learning in python," *The Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[26] B. M. Henrique, V. A. Sobreiro, and H. Kimura, "Stock price prediction using support vector regression on daily and up to the minute prices," *The Journal of Finance and Data Science*, vol. 4, no. 3, pp. 183–201, 2018.

[27] M. L. De Prado, *Advances in financial machine learning*. John Wiley & Sons, 2018.