RESEARCH ARTICLE　　　　　　　　　　　　　　　　　　　　**OPEN ACCESS**

CHOOSING THE MOST EFFICIENT MODEL FOR THE FORECASTING OF FINANCIAL RESOURCES

Andressa Contarato*[1], Pedro Ivo Rodrigues[1], Marco Aurélio Sanfins[1] and Reinaldo Castro Souza[2]

[1]Universidade Federal Fluminense, Statistics Department;
[2]Pontifícia Universidade Católica, Department of Electrical Engineering

ARTICLE INFO	ABSTRACT
	Within financial institutions, it is common practice to analyze the value of resources to obtain future results through prediction calculations. The calculation methodologies are the most diverse, from descriptive analysis and moving averages implemented in Excel spreadsheets to the most varied and powerful models in finance. These models are fundamental when it comes to creating diverse and complex systems that help simulate a range of investment scenarios and thus choose the best one among them. With the advent of Big Data and the significant increase of investments in new technologies, it has become more feasible to apply models derived from techniques called Machine Learning. These techniques have a high computational power to obtain the most accurate analysis and forecasts of financial market behavior. This paper aims to compare models based on the traditional methodological structure, using Time Series models, such as Autoregressive and Moving Average (ARIMA) and GARCH models. Furthermore, models with the Machine Learning approach, namely: Support Vector Regression (SVR) and Artificial Neural Network (ANN). These models were applied to some series of financial assets and some tokens. The results showed that for both types of assets, models with the machine learning approach performed better, but with different highlights for SVR and ANN respectively.

INTRODUCTION

The word Finance in the Aurélio dictionary has the following definition:

> "Doctrine or practice of using money, financial securities (...) usually refers to state institutions (...)."

Looking at this word in a more generalized way [4], it can be said that it encompasses the set of processes, markets, institutions, among others, with the purpose of transferring funds between people, companies and even the government. Finance is crucially involved in the everyday life of human beings. Each day new revenues, fund and investment resources are generated and surveyed by individuals and companies. Thus, it is generally understood that Finance is the art of managing funds. Financial analysis has been growing over time. It is always necessary to use sophisticated methodologies and complex algorithms to understand the market in order to predict future conditions that can affect anything from one person's life to an entire country. Models widely used in this field come from Statistics, more specifically, from the study area denoted as Time Series [8]. Currently, these models are traditionally used in the financial market,

such as the ARIMA, ARCH and GARCH models. The formulation of these types of models relies on analyzing observed data over time. For example, when analyzing a financial asset, the scope is to check how much past results influence future values and how much past errors may reflect on the future. Created by Box and Jenkins (1970), the models called ARIMA are from the class of linear models. The objective of linear models is to understand how past data can affect the present scenario. The models classified as GARCH and ARCH are the so-called non-linear models, whose purpose is to understand how much the errors (or residuals) can influence the errors of the present. Other methods, such as Support Vector Regression and Multilayer Perceptron (which is within the field of the so-called Artificial Neural Networks or ANN), are derived of methodologies from the area of Artificial Intelligence. Widely reported in several scientific articles, nowadays this area can have several applications with the computational advance and due to the large volumes of data, called Big Data. Within Artificial Intelligence, there is the Machine Learning branch, and within this one, Deep Learning. Machine Learning has been widely used in Finance, and just like Artificial Intelligence, its history is old. Machine Learning has become an essential technique in Finance, given the increasing amount of data one has, such as historical records by year, month, day, hours and

evenseconds. Today [3], it is seen that there are more cases of Machine Learning use in finance than ever before, with the increase in computational capacity and techniques, previously impossible to apply, suitable for this volume, causing Machine Learning to play a key role in many phases of the financial ecosystem, ranging from loan approval to risk assessment. There are many ways to use Learning techniques, such as in Portfolio Management and Fraud Detection. In view of all this, this paper discusses and compares these techniques on data related to financial resources and analyzes how good these models are in predicting future scenarios of these resources.

Objective: The motivation comes from the importance of studying and presenting efficient methods capable of providing improvements for the financial sector, or at least, other types of methodologies as effective as those already seen and applied. Traditional models and machine learning models will be analyzed for comparability, in order to verify their performance and accuracy. The scope is to identify which model best fits the financial resources data, looking at the RMSE and MAPE, to finally predict the time series. This will be done by analyzing the nature of the data used in the study, bibliographic survey of the models, their construction and application and, finally, the results analyzed with the possible choice of one or more models for each series addressed.

RESEARCH ELABORATIONS

The methodological analysis is divided into three parts: Basic Concepts of Finance, Time Series and Machine Learning. Regarding the topic of Basic Concepts of Finance, the definition of assets and tokens is shown. The Time Series section covers concepts such as: What are Time Series, Stationarity, Models and Testing. And finally, Machine Learning covers concepts such as: SVM, SVR, Artificial Neural Networks, Perceptron and Backpropagation.

Financial Resources: A Share is defined as the smallest portion of the capital stock of companies or even corporations. It can be considered as a property title granting the shareholders all the rights and duties of a partner, within the limit of the shares owned. According to item I, from article 2, of Law 6385/76, the share can be considered as a security, once the share is issued by companies registered at the Securities and Exchange Commission (called open companies). They can be publicly traded on the securities market. Today, shares are mostly book-entry, held in deposit accounts in the name of the holders, without the issue of a certificate. A share's attribution is that it must always be nominative, that is, bearer or endorsable shares cannot be issued and traded [9]. Another type of financial resource that is gaining strength are the so-called cryptocurrencies. With the advent of technology and the most varied forms of payment and receipt of money, cryptocurrencies were created. Being a means of exchange that makes massive use of blockchain technology and cryptography, it has the purpose of ensuring more confidence to the end user regarding virtual monetary transactions. The first currency produced with this purpose and characteristics was the so-called Bitcoin, created in 2009. Since then, other types of cryptocurrencies with similar objectives have been emerging massively [7]. The database is composed of 6 distinct types of stocks, namely: MSFT: asset of the Microsoft company; SBUX: asset of the Starbucks company; IBM: asset of the IBM company; AAPL: asset of the Apple company; GSPC: Asset; AMZN: asset of the Amazon company. And, 3 different types of cryptocurrencies, namely: BTC-USD: Bitcoin/US Dollar; ETH-USD: Ethereum to US Dollar; and LTC-USD: Litecoin US Dollar.

Stochastic Processes: Regarding the definition of time series, stochastic processes can be defined as a set of observations ordered by time [2]. The process considering T an arbitrary set designating time, is defined as: a stochastic process is a family $Z = Z_t$, $t \in T$, where, for each $t \in T$, Z_t is a random variable. Therefore, a stochastic process is a family of random variables (r.v.) defined in an equal probability space, namely (Ω, A, P), where Ω is the sample space, A

is defined as the algebra of events or event associated with the sample space, and P the probability associated with the event [5].

Stationarity: In a time series, the scenario where the number of parameters can be greater than the number of observations is usually observed. For a better use of time series models, certain assumptions have to be imposed in order to simplify the way of using them [8]. There are many types of assumptions, such as: stationary processes, Gaussian processes, and Markovian processes. A non-stationary series causes autocorrelations to decay to zero considering large values and also graphs such as the correlogram (to get an idea of the use of different model types) may show observations (on one side of the mean) too close together because they are biased. In this case almost no information is absorbed from the data. So, one of the assumptions to be used in a time series model is that of stationarity. The idea of a process being stationary is that it develops over time, making the determination of this time unimportant, i.e., the mean and the autocorrelation function do not change even if you change the time interval in the graph. In addition, there are two forms of stationarity referred to as: strict (or strong) and weak (or broad) stationarity discussed below.

Definition: Let Z_t, $t \in T$ be a stochastic process, Z_t is said to be strictly stationary if over time all its finite-dimensional distributions are the same. And, further, we can write as follows: $F(z_1, ..., z_n; t_1 + \tau, ..., t_n + \tau) = F(z_1, ..., z_n; t_1, ..., t_n)$. Whatever $t_1, ..., t_n, \tau$ of T.

For a better choice of a model, restrictions are made on first and second order moments. Another definition then arises. This one for a weakly stationary process:

Definition: A stochastic process $Z = Z_t$, $t \in T$ is said to be weakly stationary if and only if:

1. $E(Z_t) = \mu_t = \mu$, remain constant $\forall t \in T$
2. $E(Z_t^2) < \infty$, $\forall t \in T$; and
3. $\sigma(t_1, t_2) = \text{Cov}(Z(t_1), Z(t_2))$ function of $|t_1 - t_2|$.

Time Series: The monitoring of funds in the financial market can be daily, weekly, monthly, yearly, etc. Knowing what the behavior of an asset will be tomorrow or after a certain cycle is of great relevance to investors. This idea of monitoring observations over a certain period of time is what is called time series [2]. Techniques are developed over time such as:

1. In the 1930s and 1940s compound methods related to trend, seasonality and randomness were used, i.e. the model to be used was: $Z_t = f(T_t, S_t, A_t)$. Where t is time, Z is the series related to a function f with the arguments of seasonality (S), trend (T) and cycle (A). In 1960, automatic or black-box methods, i.e., exponential damping techniques.
2. Already in the 1970s, there was the creation of the Box and Jenkins Method, which will be addressed later in the determination of models.
3. In 1980, the use of Bayesian inference techniques was made for use in time series.
4. And, still in the 1980s, non-linear models were introduced by Engle (1982).
5. Currently, studies are being developed for a better understanding of data, derived from neural network technologies, fuzzy logic, and genetic algorithms.

Thus, the action of assessing how much an asset will fall or rise is an important part of the investment in it. Monitoring an asset requires only a stochastic process of the asset. For this reason, it is important to know this process, so many studies are dedicated to accurate estimation [8]. Regarding the definition of Stochastic Processes previously seen, when $t \in T$, Z_t is a real random variable. Also, since $t \in T$, Z_t is called an r.v. defined over the sample space ω. One can see that Z_t is a function of two arguments, Z_t, ω, with $t \in T$ and $\omega \in \Omega$. One observation regarding the probability distribution Z_t, ω is that this probability density function (pdf) at time t_1 can be different from

the pdf at time t_2, for any two times t_1 and t_2. Even so, it is more usual the situation where the pdf Z_t, ω is the same, $\forall t \in T$. Contrarily, by fixing a $\omega \in \Omega$, we will obtain a function of t that can be called a realization or trajectory of the process, or a time series. It is interesting to note that if the set T is finite or at least enumerable, such as T = 1, 2, 3, ... or T = Z, the process is called a discrete parameter. On the other hand, if T is an interval, we will get a continuous parameter process.

White Noise: The concept of white noise follows by the following definition:

Considering T as the set of integers. A random process $\epsilon_t : t \in T$ is said to be a white noise with variance σ^2, where $\sigma^2 > 0$, if and only if:

1. ϵ_t is quadratically integrable variable, $\forall t \in T$.
2. $E(\epsilon_t) = \mu_\epsilon$, for $t \in T$.
3. $Var(\epsilon_t) = \sigma^2$, $\forall t \in T$.
4. $Cov(\epsilon_t, \epsilon_s) = 0$, $\forall t, s \in T, t \neq s$.

Notation: It can be said that $\epsilon_t \sim WN(0, \sigma^2)$, where W N stands for White Noise, such notation is taken from the book by Brockwell & Davis (2001) [2]. Statements with respect to covariance are validated because the random variables are quadratically integrable. If a more restricted notion of white noise is taken, one has the definition of i.i.d. noise. Considering T as the set of integers. A random process ϵ_t: $t \in T$ is said to be i.i.d. noise if the r.v. ϵ_t are independent, as a consequence uncorrelated. This sequence, defined above, is also called a purely random process.

1. The random variables ϵ_t, $t \in T$ are independent.
2. $\epsilon_t \sim$ i.i.d.$(0, \sigma_\epsilon^2)$.

Arima: Economic and financial series are mostly nonstationary, ideal models for these types of series are those that develop over time not having constant mean and variance. It is important to point out that many of these series, after being differentiated, become stationary (homogeneous non stationary series or series with unit roots). When considering an ARMA model, the condition for the series to be stationary is to verify that $\varphi < 1$. If $\varphi = 1$, a nonstationary process is obtained, and for $\varphi > 1$, the process explodes as t increases. However, here we will deal with series that do not explode but can be nonstationary and still have a deterministic trend (when they have an intercept). For non-stationary models to be stationary, $W_t = \Delta^d Z_t$ is considered for the d-th difference of the $Z_\square$ series, so it is possible to have W_t as an ARMA(p, q) model, i.e. [8]: $\phi(B) W_t = \theta(B) a_t$. (2)

Given that W_t is a difference of the original series, one comes to the conclusion that Z_t is an integral of W_t, rewriting (2) as: $\phi(B) \Delta Z_t = \theta(B) a_t$.(3)

The series $Z_\square$ is said to follow an ARIMA model (p, d, q), where p and q are the orders of $\phi(B)$ and $\theta(B)$ respectively. It is important to note that all the roots lie outside the unit circle. Then, rewriting equation (2) and considering $\varphi(B)$ as the autoregressive operator, we have the following formula: $\varphi(B) = \phi(B) \ \Delta^d Z_t = \phi(B)(1 - B)^d$ (4)

This model has to assume that the d-th difference can be represented by an ARMA model (which is stationary and invertible). Mostly, it is necessary to do only up to the second difference, d = 1 or d = 2, having the following characteristics [8]:

1. When you have stationarity right at the first difference, that is, d = 1, you can observe that the series oscillates around a level, on average, common over time and then changes to another temporary level. Cases like this happen in economic series.
2. Now, when one makes use of the second difference, that is d = 2, if it is common to find oscillations for some time, and then a change occurs. This happens in non-stationary series with respect to their slope.

In the ARIMA model it is possible to have three types of structure:

1. Difference equation form $Z_t = \varphi_1 Z_{t-1} + \varphi_2 Z_{t-2} + \ldots + \varphi_p Z_{t-p} + a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} \ldots - \theta_q a_{t-q}$, where $\varphi(B) = 1 - \varphi_1 B - \phi_2 B^2 - \ldots - \varphi_p + d + B^{p+d}$.
2. Random shock form (i.e., infinite moving averages) $Z_t = a_t + \psi_1 a_{t-1} + \psi_2 a_{t-2} + \ldots = \psi(B) a_t$ from which we have the following equation $\varphi(B) Z_t = \varphi(B) \psi(B) a_t$.
3. Inverted form (or infinite autoregressive model $Z_t = \left[1 - \sum_{j=1}^{\infty} \pi_j B^j\right] Z_t = a_t$) of the Z_t equation in random shocks, is obtained $\psi^{-1}(B) Z_t = a_t$ or $\pi(B) = a_t$, following $\varphi(B) Z_t = \theta(B) a_t = \theta(B) \pi(B) Z_t$.

GARCH

Even after the treatment of a series, the fitting of an ARMA, AR, MA or ARIMA model and its diagnosis, it is possible to find a case where the residuals are correlated when squared. So, for some situations, when applying the squared term to the residuals they are correlated. To solve this problem non-linear models were made, introduced by Engle (1982), as ARCH (autoregressive conditional heteroskedasticity) and GARCH (Generalized ARCH) [2]. When they were created, the purpose of autoregressive models with conditional heteroskedasticity was to estimate inflation variability. The return Xt is uncorrelated but the volatility (which is the conditional variance) depends on past returns. The idea of constructing this model is to consider the residuals of the return as Xt and the volatility dependent on past returns [8]. Thus, the ARCH model of order r is given by the following expression:

$$X_t = \sqrt{h_t \varepsilon_t} \qquad (5)$$

And

$$h_t = \alpha_0 + \alpha_1 X^2_{t-1} + \ldots + \alpha_r X^2_{t-r} \qquad (6),$$

where ε_t is a sequence of i.i.d v.a.'s with mean zero and variance constant and equal to 1. In practice it is common to assume that the residuals have Normal (0, 1) or T-student distribution with v degrees of freedom. Here, however, the following case will be considered:

1. $\varepsilon_t \sim$ Normal(0, 1).
2. $\varepsilon_t \sim$ GHsT(μ, δ, v, β).

The general form of the ARCH model is the GARCH (Generalized ARCH) model, introduced by Bollerslev (1986). A model of the form GARCH (r, s) is defined by [8]:

$$X_t = \sqrt{h_t \epsilon_t} \qquad (7)$$
and
$$h_t = \alpha_0 + \sum_{i=1}^{r} \alpha_i X^2_{t-i} + \sum_{j=1}^{s} \beta_j h_{t-j} \qquad (8)$$

Where: ϵ_t is i.i.dd $\alpha_0 > 0, \alpha_i \geq 0, \beta_j \geq 0, \sum_{i=1}^{q} < 1, q = max(r,s)$.

Machine Learning: Machine Learning (ML) [1] has been one of the most widely used techniques today. These are the characteristics of the information used: Volume, Speed and Veracity. Thus, with the advent of Big Data, ML has gone from formulas in books to something totally feasible. The term Machine Learning was first used by American computer scientist Arthur Samuel, who developed one of the first computer programs to play chess, and in 1959 he defined machine learning as a technology that allows programs the ability to learn without being programmed. As such, machine learning is a subset of artificial intelligence, the segment of computer science that focuses on creating computers that think the way humans think. ML can be divided into the following types of learning: Supervised Learning, Unsupervised Learning, Semi-Supervised Learning, and Reinforcement Learning. Machine Learning has become necessary to solve tasks that are too complex for humans [3]. Some tasks are so complex that it is impractical, if not impossible, for normal human beings to do. Thus, a large volume of data is provided to a machine learning algorithm to process and learn by exploring such data, in

search for a better model that achieves the goal [6]. Therefore, instead of writing a program by hand for each specific task, examples can be collected that guide the correct output for a given input. A machine learning algorithm takes these examples and produces a program that does the job. The idea is to look at previous successful inputs and outputs and to be able, in a way, to learn from the mistakes and successes. Once a good model is made, the program works for new data. If the data changes, the program can readapt itself as well. And with reduced computing costs, a large volume and parallel processing on GPU, you have perfect conditions for developing high-performance models in Machine Learning.

Artificial Neural Network: Aiming at the use of high-performance models, we have the example of a simple model, but still widely used nowadays, called Perceptron. The Perceptron is a simple algorithm designed to perform binary classification; that is, it predicts whether the input belongs to a certain category of interest or not: dog or not dog, customer who can grant credit or customer who cannot grant credit. By definition, a Perceptron neural network is a linear classifier, that is, an algorithm that classifies the input by separating two categories with a straight line. In most problems solved, you start the input with a feature vector, called x, multiplied by weights w and add a bias b. For example, imagine a perceptron: y = w * x + b. A Perceptron produces a single output based on several real-valued inputs, forming a linear combination using the weights [1]. Considering a perceptron containing one layer as a way of example of understanding a neural network, that is, the algorithm does not include multiple layers, allowing neural networks to model a hierarchy of features. We can see that this construction prevents the neural network from being able to do non-linear classification, such as XOR function. Considering the Perceptron neural network, but now with multiple layers, called a Multilayer Perceptron (MLP). This network will consist of the following structure: an input layer to receive the signal, an output layer that holds the output over the input, in between these two layers, an arbitrary number of hidden layers that are the actual computational mechanism of the MLP. One of the advantages of this type of neural network is that it can approximate any continuous function. One of the most important components of this and other neural networks are the algorithms called Backpropagation. Without this resource it would be unfeasible to train neural networks, such as the MLP, with high performance as we see today [1]. Such an algorithm can have one of two stages of construction, namely: Forward Pass or Backward Pass. Thus, for any supervised learning problem, those weights that can provide an optimal estimate of a function that models the training data are selected, that is, a set of weights W that can minimize the output of J(W), where J(W) is the loss function. The main scope of the backpropagation algorithm is to optimize the weights so that the neural network can learn to correctly map inputs to outputs [6].

Support Vector Regression: The algorithm called Support Vector Regression, or $\in$-SVR, is intended to find a function f(x) that produces continuous outputs for the data under analysis and that such data deviates at most from its desired value. Another specificity is that this function should be as uniform and regular as possible. Considering, first of all, the use of linear functions f. Thus, regularity is reflected in seeking a function with a very small w. This can be obtained by minimizing the norm, denominated as $\|w\|$. We have below the following optimization problem [3] Minimize $Minimize_{\omega,b}\frac{1}{2}\left\|\omega\right\|^2$) with the following restrictions $y_i - \omega x_i \leq \epsilon_i$ or $\omega x_i + b - y_i \leq \epsilon_i$. Then we look for the linear function that approximates the training pairs (x_i, y_i) with a precision $\in$. Imagine a line with a region above and below the line that cuts the points on the Cartesian axis. This variation we can call ϵ and $-\epsilon$. Thus, we search for a linear function such that the training data are mostly within this region around f. Analogous to the case of Support Vector Machines (SVM's), a solution to this problem is based on SVM's with slack variables, allowing to deal with outliers and noise in the objects. The idea is to allow some examples to fall outside this stipulated $+\epsilon$ and $-\epsilon$ region. Thus:

$$Minimize_{w,b,\zeta,\underline{\xi}} \frac{1}{2}\|w\|^2 + C\left(\sum_{i=1}^{n}\zeta_i + \underline{\zeta_i}\right) \qquad (11)$$

with restrictions, we have: $y_i - wx_i \leq \epsilon_i$ or $wx_i + b - y_i \leq \epsilon_i + \zeta$ where ζ_i and $\underline{\zeta_i}, \geq 0$.

Where, ζi and $\underline{\zeta i}$ are the slack variables and C the constant that imposes the trade-off between the regularity of f and how much the deviations are tolerated. As in the case of SVMs, the dual problem equivalent to the previous one is elaborated by using a lagrangian, making the results of the partial derivations null and substituting the resulting expressions into the initial lagrangian equation. This dual problem is described in the form of an inner product between objects. Resorting to the use of kernels for the computation of the nonlinear regression. Using this implies mapping the objects to a feature space, where the most regular linear function with low training error is found.

Adjustment Measures

1. The next step is to actually analyze whether the model is fit to the data through test statistics denoted as RMSE and MAPE. Such tests are intended to analyze whether one model was more efficient than the other inference model. Where for the RMSE, the lower the value of this statistic, the better the fit of the model, and the lower, the less fit to the data the model is. This means that the higher the fit, the more suitable is the model for predicting future scenarios. The other measure used, MAPE, aims at comparability between models, where it indicates the predictive ability through the percentage of error in the model. Thus, the lower the MAPE value the better the model [2]. Below is a further breakdown of the measures adopted:

2. RMSE and Accuracy: The evaluation of model performance in the validation phase is subjective, and various metrics can be used to measure how good is the model's performance. As in the calibration phase, statistical indicators are used. One way to evaluate the model is by comparing the RMSE (Root Mean Squared Error); the lower the value, the better the model. The formula is explained by: $\mathbf{RMSE} = \sqrt{E(\epsilon^2)}$.

3. MAPE: The Mean absolute percentage error is a very effective measure in deciding whether the model has good accuracy with respect to the data prediction results. The percentage error is given by $\boldsymbol{p_t} = \frac{100\epsilon_t}{y_t}$. These percentage errors have the advantage that they are unitless and are therefore often used to compare prediction performances between data sets. Thus, one of the possible calculations of MAPE is given by: MAP E = mean($|\boldsymbol{p_t}|$).

RESULTS AND FINDING

The following discusses the analysis and conclusions based on the data collected and the models application according to the methodology above. The procedure was similar in these two aspects, assets, and tokens. A database was collected for each financial resource, containing 6 columns (Opening, Highest Price, Lowest Price, Closing, Adjusted and Volume), using the adjusted value operated on the day under analysis. The first step of the analysis was to download the data on-line through the getSymbols function of the RStudio software. Next, a treatment of the data was done, such as the creation of the date column (previously as an index in the base) to facilitate the analyses and the removal of null values. Thus, the series diverged in their beginnings, being treated independently in the analyses and conclusions of the same methodological rules applied. After this data structuring, the following models were applied: ARIMA and GARCH; SVR; ANN: multilayer perceptron model trained with backpropagation algorithm, separately and for each financial resource. For all models applied, and for each asset analyzed, 80% was used as training data, to model the data, and 20% as test data (comparing the original value with the values predicted by each model and for each asset). The application of the time series methodology consisted in finding the linear model that best fitted the

data, with AIC and RMSE as a basis for comparing the models. After that, tests were applied to the residuals of this model in order to identify if it would be necessary to model these residuals, that is, if there were still explanatory factors in the residuals that could better predict/explain the model. The analyses of the series showed similar processes, mainly because they are financial series. For the training data, a graphical analysis was started by plotting the series graph to understand the data behavior. Then the unit root test was performed, and in all series, the null hypothesis (Ho: Non-stationary) did not have enough evidence to be rejected, applying the first difference to the data series.

Table 1. Result of the RMSE of the models obtained

Asset	Arima	Arima + Garch	ANN	SVR
MSFT	48,97	2,46	5,98	2,06
SBUX	34,41	1,15	2,33	1,01
IBM	19,87	3,96	1,92	1,94
AAPL	12,01	-	4,01	3,79
GSPC	178,12	69,29	30,91	31,30
AMZN	702,57	47,97	90,75	27,16

Source: Yahoo! Finance. Elaboration: author's own.

Table 2. Result of the MAPE of the models obtained

Asset	Arima	Arima + Garch	ANN	SVR
MSFT	77,38	-1,15	-0,62	-0,11
SBUX	57,81	-0,65	0,04	-0,37
IBM	11,53	-2,38	-0,02	-0,23
AAPL	8,41	-	-2,17	-0,38
GSPC	6,84	-1,70	-0,05	-0,21
AMZN	75,27	-2,83	-3,44	-0,31

Source: Yahoo! Finance. Elaboration: author's own.

Thus, the first difference was applied. Then, with the first difference data, the unit root test was applied again with the rejection of Ho for all the asset series, that is, the series was stationary with the first difference. The next step was to find the p and q parameters of the ARIMA model by doing a cross-validation, testing parameters. And, comparing the models according to the AIC and RMSE criteria. Finishing this step of finding the most suitable linear model, the squared residuals test was done to identify if there was a need to model with non-linear models.

And, in all cases there was such a need. In addition, the normality test of the residuals was performed where all rejected the null hypothesis Ho: They are Normal and thus, applying the GHskew distribution to the data. After this, the RMSE of the final model was returned in order to compare it with the others. In the next step, the neural network and support vector regression models were calculated. For the artificial neural network, the training parameters were set as follows: decay rate = 0.00001, number of units in the hidden layer = 10, and size of epochs = 10000. Finally, the network was tested with the data set to estimate its generalization ability. The metrics used to choose the best model were RMSE and MAPE, the smaller the model, the better.

Assets: Six stocks of companies listed on the US stock exchange were used, namely: MSFT, SBUX, IBM, AAPL, GSPC and AMZN. The time period analyzed in each series was from 01/01/2000 to 03/31/2020, with a total of 7390 observations, excepting those cases where there was a data problem, resulting in an NA (missing values) and being removed from the observation. Chart 1 below shows the series used in this study in order to understand their behavior, these are the original data extracted via software. Where it is possible to analyze that most of the asset series start very low, but have a tendency to increase over time, holding similar behaviors. Table 1.1 below shows the results of all models. The table shows that the ARIMA model obtained the worst values for all the analyzed assets, but it showed an improved when the GARCH model was incorporated. Even so, the model obtaining the best RMSE was the SVR. Table 1.2 shows the MAPE, as MAPE has an idea similar to RMSE, the smaller, the better the fit of the model used.MAPE results for all the models tested, where it can be seen that the ANN and SVR models have the lowest percentage of errors, indicating that they are the models with the best adjustment for forecasting. The series with no value in the GARCH part either did not need such modeling or it was not possible to apply it.

Tokens: For cryptocurrencies, the same time interval was used (from 01/01/2000 to 03/31/2020), except for series that did not hold such a large time evolution, the NA value was removed for the analyses. Three time series of different cryptocurrencies listed on the American stock exchange were used, namely: BTC-USD, ETH-USD and LTC-USD. Table 1.3 below reports the number of observations for each series.
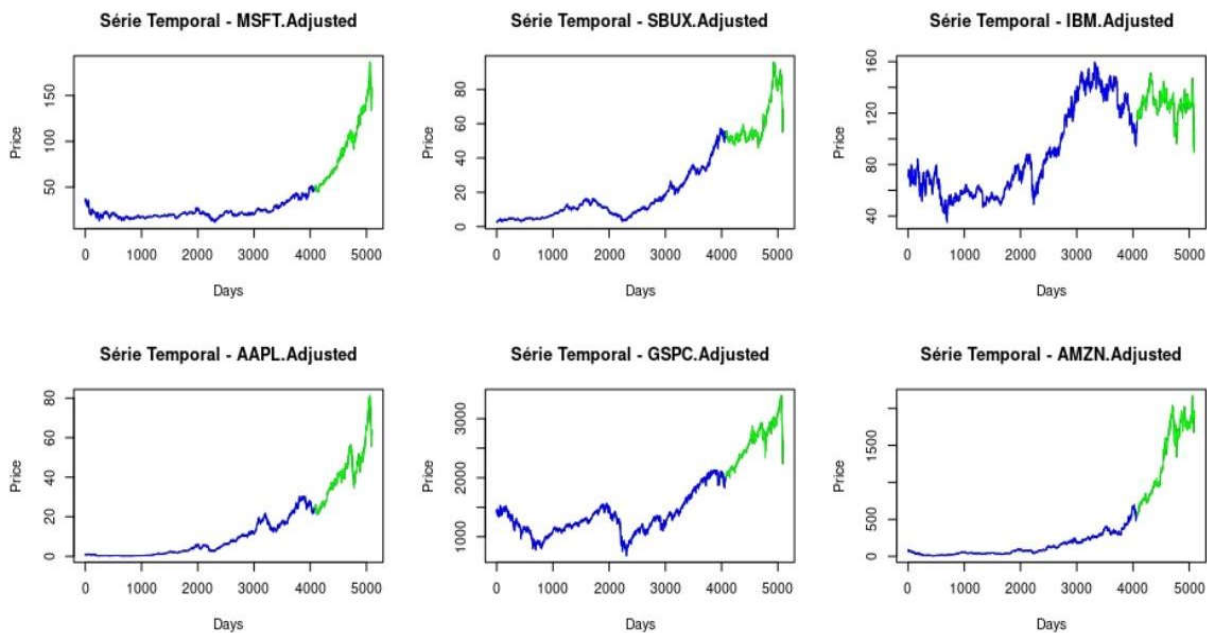

Figure 1. Original time series of financial assets, showing the training series in blue and the test series in green. Source: Yahoo! Finance. Elaboration: author's own
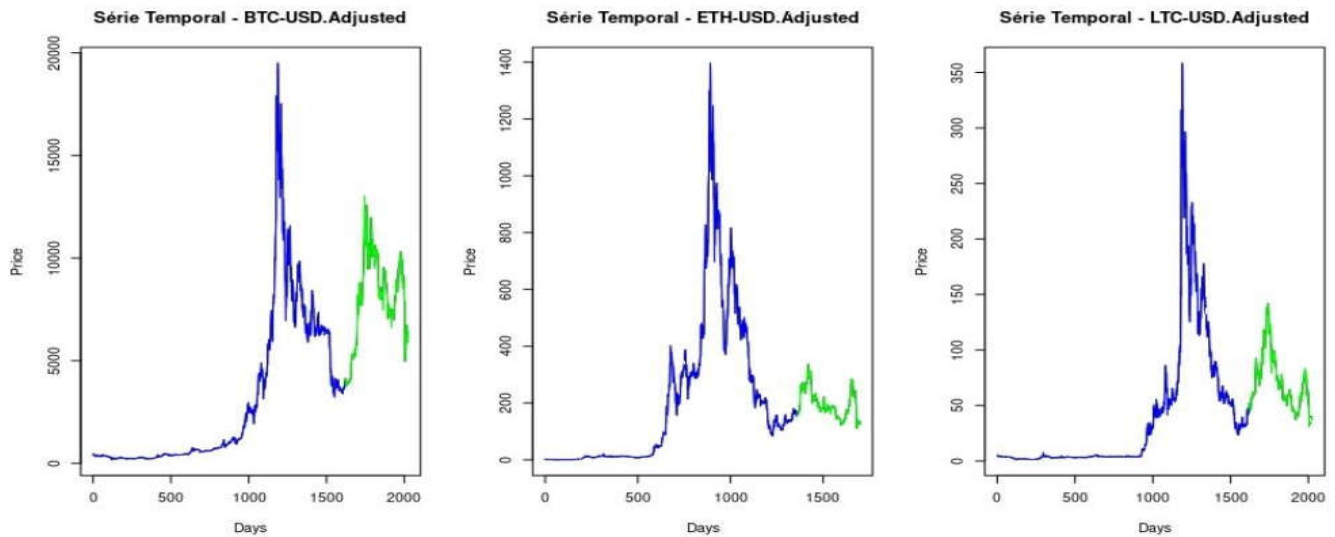
Figure 2. Original cryptocurrency series, where blue is the training series and green is the test series. Source: Yahoo! Finance. Elaboration: author's own

Table 3. Number of Observations

Cryptocurrency	Initial Date	Final Date	Total Observations	Forecast
BTC-USD	2014-09-17	2020-03-31	2023	404
ETH-USD	2015-08-07	2020-03-31	1699	339
LTC-USD	2014-09-17	2020-03-31	2023	404

Source: Yahoo! Finance. Elaboration: author's own.

Table 4. Result of the RMSE of the models obtained

Cryptocurrency	Arima	Arima + Garch	ANN	SVR
BTC-USD	4783,57	648,16	357,76	357,45
ETH-USD	65,54	20,66	10,42	10,88
LTC-USD	37,84	5,88	3,74	3,82

Source: Yahoo! Finance. Elaboration: author's own.

Table 5. Result of the MAPE of the models obtained

Cryptocurrency	Arima	Arima + Garch	ANN	SVR
BTC-USD	43,61	-7,00	-0,35	-0,34
ETH-USD	21,97	-8,20	0,64	1,81
LTC-USD	30,71	-7,05	0	1,38

Source: Yahoo! Finance. Elaboration: author's own.

Chart 2 below shows the series used in this study for the purpose of understanding their behavior, and this is the original data extracted via software. It is possible to analyze that most of the cryptocurrency series start very low, for having been recently created, with slight peaks at the beginning in some, and distinct peaks over time for the series overall, showing more random behaviors from each other. The procedure was analogous to the financial asset series. The metric used to choose the best model was the RMSE, the lower the RMSE the better the model. Table 1.5 below shows the results for all the models. As MAPE has an idea similar to RMSE, the smaller, the better the fit of the model used. Thus, table 1.5 shows that the best models are SVR and ANN for the asset series.The series with no value in the GARCH part either did not need such modeling or it was not possible to apply it.The table showed that the ARIMA model obtained the worst values, for all analyzed assets, but when the GARCH model was incorporated there was an improvement. Even so, the results showed that both ANN obtained great results. The same happens when we look at table 1.5 below, where the models that stand out the most, that is, that had the lowest error, are the SVR models.

CONCLUSION

The results are part of the test set, aimed at validating the chosen models. From the RMSE values of all the models performed above, that the lower the RMSE, the better the model fits.

The ANN and SVR models were the best for predicting the closing data of the assets. However, the ARIMA + GARCH model resulted in good models as well. It was observed that the prediction accuracy level of SVR and ANN models compared to that of ARIMA model is quite significant. It can be argued that these two models achieved good forecasting performance by judging the forecast error of all the models, which were quite low. It was also observed that even the function that makes the ARIMA model automatic, does not make it better than the other models. As for cryptocurrencies, it was shown that there are models with higher accuracy (looking at the RMSE) than others.

But all these models showed good results regarding the data fit, especially the Time Series models after fitting with non-linear models. However, if one were to choose one of these, the model that would fit best for all the data examples provided would be the ANN. Some models showed an overfitting of the data. In other words, that is a model that learned a lot from the training data but becomes ineffective with other types of data. A future work would be to analyze the causes of this overfitting and to be able to better fit the models. When you are using time series models, a common method is to increase the run of the training data, when using neural networks, another way out is regularization (which boils down to adding a cost to the loss function).

REFERENCES

[1] DEEP LEARNING BOOK. Data Science Academy, Rio de Janeiro. Disponível em: <http://deeplearningbook.com.br/>. Acesso em: 01 de mai. de 2021.

[2] BROCKWELL, Peter J. et al. Introduction to time series and forecasting. springer, 2016.

[3] FACELI, Katti et al. Inteligência Artificial: Uma abordagem de aprendizado de máquina. 2011.

[4] CONCEITOS BASICOS EM FINANCAS. Puc Goias, Goias. Disponível em: <http://professor.pucgoias.edu.br/SiteDocente/admin/ arquivosUpload/3843/material/AULA%203% 20GEST%C3%83O%20FINANCEIRA%20_2015.pdf>. Acesso em: 01 de mai. de 2021.

[5] JAMES, Barry R. Probabilidade: umcurso em nível intermediário. IMPA,1996.

[6] LECUN, Yann; BENGIO, Yoshua; HINTON, Geoffrey. Deep learning. nature, v. 521, n. 7553, p. 436-444, 2015.2015.

[7] MARTIN, Felix. Dinheiro: Uma biografia não autorizada: Da cunhagem à criptomoeda. Portfolio-Penguin, 2016.

[8] MORETTIN, Pedro Alberto; TOLOI, Clélia Maria de Castro. Análise de séries temporais. [S.l: s.n.], 2004.

****