Revista de Investigación en Ciencias Contables y Administrativas
(Journal of Research in Accounting and Management Science)
Vol. 7, Num. 2, 2022

Forecasting exchange rate with machine learning and deep learning algorithms: the US Dollar (USD) to Mexican Peso (MXN) case

Alfredo Yasberth Pérez-Frías
Instituto Politécnico Nacional
yasberthperez@outlook.com

Ana Lorena Jiménez-Preciado
Instituto Politécnico Nacional
ajimenezp@ipn.mx

Salvador Cruz-Aké
Instituto Politécnico Nacional
salvador.ake22@gmail.com

Recibido el 03 de marzo del 2022; Aceptado el 17 de mayo del 2022; En línea el 31 de julio del 2022

Resumen: Se presenta un análisis del pronóstico el tipo de cambio peso mexicano/dólar estadounidense (MXN/USD) utilizando datos diarios. Desarrollamos tres modelos para el pronóstico: 1) Redes Neuronales Recurrentes con Memoria a Largo Corto Plazo (LSTM), correspondiente a una especificación de aprendizaje profundo, 2) un algoritmo *Random Forest* (RF) (aprendizaje automático), y 3) un Modelo Autoregresivo de media móvil integrada (ARIMA) como referencia. La principal contribución de este trabajo es mostrar que una Red Neuronal Recurrente del tipo LSTM predice con un mínimo margen de error el tipo de cambio MXN/USD incluso bajo alta volatilidad (incluyendo efectos del COVID-19). Estos resultados permiten a los agentes económicos, que quieren arbitrar y cubrir, tener un método de selección más robusto y con mayor poder predictivo.

Palabras Claves: Tipo de cambio, Machine Learning, Deep Learning, Forecasting.

Códigos JEL: C50, C53, C58, E47, E49.

Paper title in Native -speaker English language

Abstract: This analysis forecast the Mexican Peso / US Dollar (MXN/USD) exchange rate using daily data. We developed three models for the forecast: 1) a Recurrent Neural Network with Long Short-Term Memory (LSTM), corresponding to a deep learning specification, 2) a Random Forest (RF) algorithm (machine learning), and 3) an Autoregressive Integrated Moving Average (ARIMA) model as a benchmark. The main contribution of this work is showing that a Recurrent Neural Network of the LSTM type predicts with a minimum margin of error the MXN/USD exchange rate even under high volatility (including COVID-19 effects). These results allow economic agents, who want to arbitrage and hedge, to have a more robust selection method with greater predictive power.

Keywords: Exchange Rate, Machine Learning, Deep Learning, Forecasting.

JEL Codes: C50, C53, C58, E47, E49.

1. Introduction

The Foreign Exchange Market (Forex or FX) is an Over The Counter (OTC) market. Worldwide, this market is the largest in terms of volume transactions, according to the Bank for International Settlements (BIS, 2019). The daily trading volume of the foreign exchange market reaches up to 6.6 trillion USD dollars, representing the higher traded volume than the sum of all the stock markets in the world. In addition, the Forex market is a benchmark that circumscribes commercial relations between countries and people.

Different methods tried to model the movements of the price of currencies, among the conventional models, stand out: Autoregressive Integrated Moving Average (ARIMA), Autoregressive Conditional Heteroskedasticity (ARCH), and Generalized Autoregressive Conditional Heteroskedasticity (GARCH) models, to name a few. However, researchers are applying increasingly sophisticated and more developed technologies. The literature indicates that both developing and developed countries are changing the way of making forecasts in the financial area by using Machine Learning (ML) algorithms and Neural Networks from Deep Learning (DL) specifications, which are part of the Artificial Intelligence toolkit (AI).

In that sense, this research aims to apply a strategy with ML and DL algorithms to forecast the trend of the closing price of the Mexican Peso against the US dollar (MXN/USD). The hypothesis is that these techniques have a more robust accuracy in prediction than traditional time series forecasting models such as ARIMA. Recent research has revealed that ML and DL algorithms catch nonlinear trends related to conventional forecasting techniques.

This study used the Mexican Peso exchange rate against the US Dollar (MXN/USD) with daily data from January 1st, 2005, to November 31st, 2020. We divided the document's structure as follows: the second section describes the differences between ML and DL models. The third section presents the literature review regarding the forecasting techniques used on exchange rates and their applications with ML and DL tools. The fourth section corresponds to the methodological application, where we compare the forecasts and their prediction robustness.

This paper implemented the Random Forest (RF) from ML, a Long Short-Term Memory (LSTM) Neural Network from DL, and an ARIMA specification as a traditional benchmark. In that sense, the last part of the document displays the conclusions, highlighting the result of the LSTM Recurrent Neural Network as the model with the most significant predictive power, even in times of high volatility, allowing economic agents to use a more robust and helpful arbitrage and hedging tool.

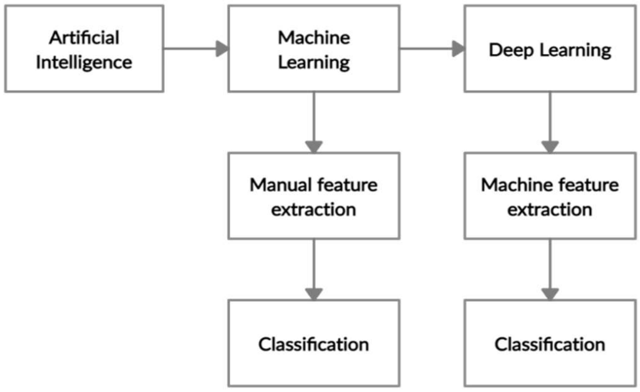
2. Machine Learning and Deep Learning algorithms

Machine Learning (ML) is, according to Mechelli & Vieira (2019), a discipline within the field of Artificial Intelligence (AI) that involves the development of algorithms, allowing machines to learn by themselves the existing trends and patterns in data. Sandoval (2018) also defines ML as a branch of AI responsible for generating algorithms that can learn by feeding the algorithm with a massive volume of data. A pioneer in computer games, Arthur Samuel, would start ML in the '50s by writing the first ML program. Therefore, the difference between AI and ML is that AI refers to the ability of machines to show "intelligent" behavior. At the same time, ML is a technique used to create and improve such behavior by training and constant feeding data.

According to Janiesch, Zschech, & Heinrich (2021), Deep Learning (DP) is a branch or subfield of ML-based on Deep Neural Networks (DNN) training, with a large set of data such as images or texts. As stated by the authors, DNN is the central technological architecture used in Deep Learning algorithms. However, it is necessary to understand the general idea of Artificial Neural Networks (ANN), which is also fundamental to AI. We can define an ANN as a set of mathematical algorithms that process information and find nonlinear relationships between the data set and whose basic processing unit is inspired by the fundamental cell of the human nervous system: the neuron (Sánchez, 2015).

In other words, the ANN's are information-processing models inspired by the human brain. Thus, ANN's mathematically mimic the human brain, connecting multiple "artificial" neural multilayers. The more hidden layers added, the deeper the network becomes. The difference between DL and ML techniques is their ability to extract features automatically. DL is a set of methods that allow a computer to automatically discover the high-level characteristics necessary for classifying data in its natural state using multiple layers of representation (Janiesch, Zschech, & Heinrich, 2021). Figure 1 simplifies the substantial difference between ML and DL.

Figure 1. Difference between Machine Learning and Deep Learning



Source*:* *Authors' illustration.*

2.1 Artificial Neural Networks

Bhagya & Dash (2020) describe the ANN as an information processing system that seeks to emulate the behavior of the biological nervous system. ANNs function as a massively parallel system that has proven efficiency for the intelligent resolution of multiple problems. The ANN translates algorithms into sequences of orders and operations and gradually adjusts according to use.

The ANNs represent the human brain's ability to think, remember, memorize, relate facts, and solve problems, which has prompted researchers to try to emulate its behavior through ANNs.These are artificial and simplified models that imitate the structure and functioning of the human brain's biological neural networks, a system capable of acquiring knowledge through experience.

According to Martín & Sanz (2006), there are three components that ANNs will try to emulate of biological nervous systems, consisting of 1) parallel processing, 2) distributed memory and 3) adaptability. Parallel processing guarantees faster processing, unlike the information provided by a sequential operation, performed instruction by instruction.

Distributed memory refers to the fact that neural systems have widely distributed information throughout the network. In this way, even if there is a neuronal deterioration, its impact on the system would be minimal. The distributed memory system can find the data accumulated in other neurons, considering that biological nerve networks. Finally, adaptability refers to the fact that the ANNs must modify the weight of the neural connections to generalize the objective problem through learning.

2.2 Recurrent Neural Network

According to Bonet et al. (2007), Recurrent Neural Networks (RNN) are designed to recognize patterns in data sequences such as texts, writing, images, or time series. In addition, RNNs are dynamic systems with an architectural configuration that allows the information to persist during training steps or periods. Bonet et al. (2007) indicates that RNRs can perform a wide variety of computational tasks such as:

- Sequence treatment.
- The continuation of a trajectory.
- Nonlinear prediction.
- The modeling of dynamic systems.

The RNN produces loops that allow holding information, meaning that the RNN has memory. The RNN achieves memory since it can use the previous outputs as inputs in the next layer. Consequently, the decision of time step $t-1$ affects the decision made in time step t. The multi-loop of the network is equivalent to having different neural networks, meaning that the information in $t-1$ influences the time t result.

This type of network has had a great boom, and this is due to the numerous applications. However, the RNN has begun to have the most applications in the Long Short-Term Memory (LSTM) type. Its structure allows the assimilation of short-term and long-term dependencies, which other Neural Networks cannot assimilate.

2.3 Random Forest

The Random Forest (RF) is a classification algorithm involving a combination of predictor trees. Each tree depends on the values of a random vector tested independently and with the same distribution. RF bases its operation on decision trees but, unlike these, improves classification precision by incorporating randomness in the construction of each classifier (Labañino, Valencia, & Toledano, 2019). In the same way, Medina & Ñique (2017) indicate that the RF algorithm is a combination of predictive trees (weak classifiers), which works with a collection of unrelated trees. According to Ao et al. (2019), some of the main advantages offered by the RF algorithm are the following:

- Deliver better precision and works efficiently with large databases.
- It can handle many variables without the need for deleting them.
- Estimate the importance of which variables are essential and provide methods for estimating missing data.
- The generated forests may be input for later use.
- It computes proximities between pairs that can be used in clustering, locating extreme values, or (scaling) giving interesting data views.
- The prototypes provide information about the relationship between the variables and the classification.

Recall that the RF algorithm improves classification precision by incorporating randomness in the construction of each classifier. The algorithm introduces randomness in the training process, where each decision tree is slightly different. The combination predictions of each tree reduce the forecast variance, improving the performance in the data set. The following section exposes some ML and DL specification applications for Forex market analysis and forecast.

3. Currency modeling and forecasting review focused on ML and DL

Several authors studied the Forex market. This section focuses on the different modeling and forecasting techniques used to identify exchange rates' paths. For instance, we highlight the study of Lee & Wong (2007), who developed a neuro-fuzzy multivariate model for making foreign currency risk management decisions using multiple macroeconomic and market microstructure variables. Using the currency pair Australian dollar against the US dollar (AUD/USD), the system proposed by these authors combines and employs an artificial neural network and the tools of intuitive reasoning (fuzzy logic inference).

Dash (2020) develops a predictive hybrid model for the Forex market by using a Recurrent Legendre Polynomial Neural Network (RLPNN) with a learning strategy based on the Shuffled Frog Leaping Algorithm (SFLA). The recurring network helps to map the internal non-linearity associated with the input and output samples. In addition, they establish a learning strategy inspired by the memetic nature of a group of frogs in search of their food locations to estimate the parameters not revealed by the network. Their research showed better predictability compared to other models included in his study.

Contreras et al. (2018) presented an Elastic Network algorithm for FOREX, inspired by the behavior of dissolving macromolecules, to model the evolution of the foreign exchange market. The algorithm can reproduce the unstable nature of the Forex market, allowing the simulation to get out of balance. In addition, the algorithm enables the simulation of the evolution of the market up to 21 currency pairs. Their research analyzed different probability distributions such as Gaussian and Pseudo-Voigt, where the Pseudo-Voigt distributions fit better to model variations in trading prices.

On the other hand, Esmaiel, Shahrazi, & Rasekhi (2012) used the Iranian Rial/US Dollar currency pair from 2005 to 2010. They applied a trend fluctuation analysis technique to predict the weak form of the efficient market hypothesis. Their results showed that the currency pair is weakly inefficient over the period; however, the degree of inefficiency is not constant over time. The findings suggest that their model can make profitable risk-adjusted transactions using previous data.

Villada, Muñoz, & García (2012) used neural networks to obtain a price forecast for two leading shares traded in the Colombian stock market. They used two neural network structures: the daily price series and the price series plus the dollar index. In their work, the authors used a periodicity of six months for the different configurations of the neural networks. The results obtained by the authors show that the ANN performed accurately with low errors in its performance and over the sample testing.

For instance, Ni et al. (2019) forecast the Forex time series data based on DL. Their method is based on a deep recurrent neural network and a deep convolutional neural network. The nine main currencies used in the analysis obtained a low margin of error and better adaptability than the ANNs estimations. We can find another DP application in Forex in Jubert, Neves, & Horta (2018); their work proposes an algorithm capable of generating technical analysis rules to make investments with leverage.

A Support Vector Machine (SVM) model is used, which classifies the Forex market in three different stages. The study uses a Euro/US Dollar (EUR/USD) currency pair. The work presented a return on investment of 83% with accurate out-of-sample predictions. Likewise, Sánchez (2015) uses a classification with SVM and estimation with ANN to predict the movement of the Colombian Peso in the intraday spot USD/COP currency pair.

In the case of the Mexican foreign exchange market, we can name a few studies like Martínez & Tse (2018), which analyzes intraday prices in the spot and futures markets for the flexible exchange rate using the Mexican Peso against the US dollar (MXN/USD). The spot market leads the futures market in price discoveries. Among the main findings, it stands out that the volatility of the Mexican stock market Index of Prices and Quotations (IPC), the announcements of the Federal Open Market Committee (FOMC), and government interventions have a significant impact on the exposure of relative prices.

Rojas & Herman (2018) applied different ML methods such as Least Absolute Shrinkage and Selection Operator (LASSO), logistic regression, Ridge, SVM, Neural Network (NN), and Gradient Boosting Classifier (GBC) to find "buy/sell" signals for the MXN/USD currency. The authors found that the SVM model offers the most accurate forecasting to implement the binary trading strategy.

ML models have proven to be practical and applicable in the financial field. According to Sánchez (2015), the financial industry adapted and used these computational intelligence models to predict time series. The prediction becomes more complicated since data present noise, non-stationary characteristics, and complex dimensionality.

In that sense, econometrics has embraced models and techniques found in statistics and AI. The ANN models gained ground in time series analysis because they can classify data and represent unknown nonlinear relationships starting from the same data. Since we focused the paper's objective on machine learning methodologies, the following section displays its implementation to analyze the currency market.

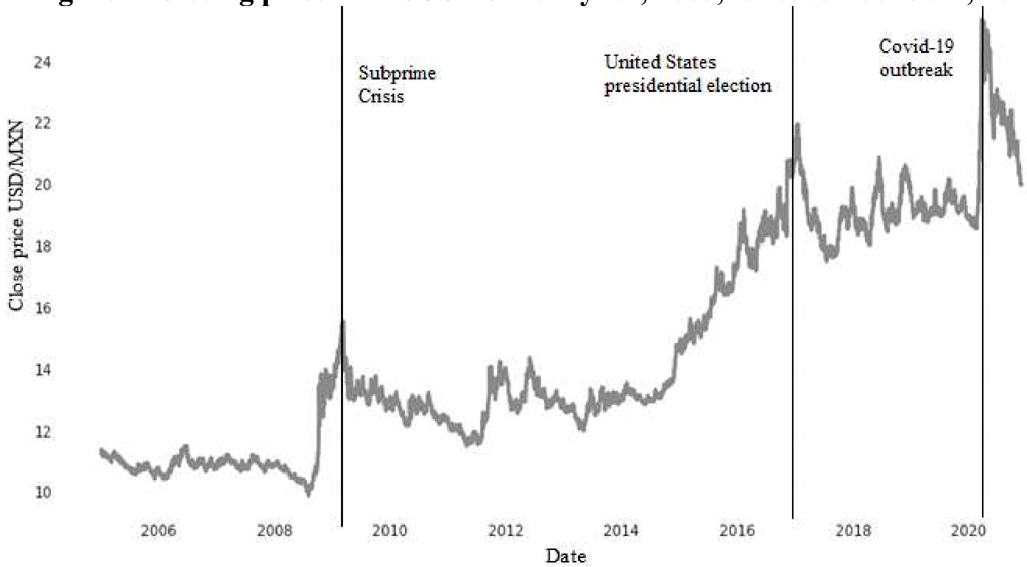
4. Currency modeling and forecasting review focused on ML and DL

First, we list the variables used for the model implementation:

- Variable: closing price MXN/USD (Mexican Peso against the US Dollar).
- Periodicity: from January 1st, 2005, to November 30th, 2020, with daily frequency.
- Figures and metrics outputs: Python 3.7.
- Database source: Yahoo Finance, 4145 observations.
- Models implemented: LSTM from deep learning, random forest from ML, and ARIMA as a benchmark.

Figure 2 shows the evolution and high volatility of the Mexican Peso due to economic, political, and health events. The first volatility peak corresponds to the subprime housing bubble crisis of 2008. The second episode of high volatility coincides with the president's Trump inauguration in 2017. The global COVID-19 pandemic triggered the third and most recent event.

Figure 2. Closing price MXN/USD January 1st, 2005, to November 30th, 2020.



Source*: Authors' illustration.*

Our paper compares the previously explained models using the Minimum Squared Error (MSE) and the Minimum Absolute Error (MAE) as the goodness of fit measurements, including high volatility events. In that sense, volatility plays a relevant role. We can understand volatility as a measure of risk derived from changes in the profitability of a financial asset. Volatility drivers are associated with the daily variations in currency prices that appear in financial markets due to multiple situations (economy, health, politics, financial reports of companies, a message written by a president on different platforms, and others).

4.1 Modeling with Recurrent Neural Networks: LSTM specification

This work presents Deep Learning, specifically Recurrent Neural Networks (RNN), with long short-term memory (LSTM) and short- and long-term memory. We structured the LSTM model with one and two layers, both hidden and dense, to allow the model to learn by layers and use a hierarchical process on the data. We developed nine LSTM models, modifying the following parameters:

- Number of neurons
- Hidden layers
- Dense layers

- Batch
- Epoch

The model starts with 50 neurons; subsequently, we duplicated the neurons and layers to obtain the most remarkable structure for forecasting the currency's price. We use evaluation metrics to select the best RNN: Mean Square Error (MSE), Mean Absolute Error (MAE), and R2 (R squared), as shown in the following equations:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 \quad (1)$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|Y_i - \hat{Y}_i\right| \quad (2)$$

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} \quad (3)$$

Where n is the number of observations considered, Y_i is the observed closing price of the currency, $\hat{Y}_i$ is the model's estimated price, SS_{res} is the sum of the squares of the residuals, and SS_{tot} is the sum of the total squares.

We started from 4,145 daily closing prices, corresponding to 15 trading years, between January 1st, 2005, and November 30th, 2020. We selected 3,316 data points for the training set (80% of the sample), and 829 were left out-of-sample to assess the model's forecasting capacity. According to Villada, Muñoz, & García (2012), this is the recommended ratio for modeling neural networks. Table 1 displays the LSTM results:

Revista de Investigación en Ciencias Contables y Administrativas
(Journal of Research in Accounting and Management Science)
Vol. 7, Num. 2, 2022

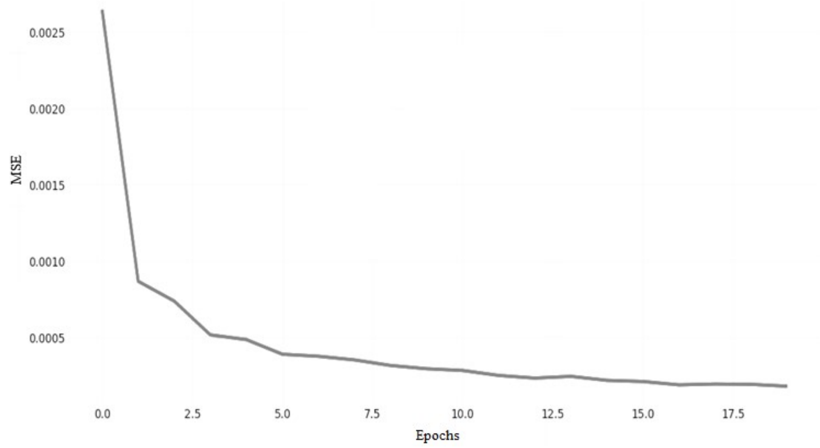
Table 1. Structure of the different LSTM models and results.

Model	Neurons	Hidden layers	Dense layers	Batch	Epoch	MSE	MAE	R^2	Estimated price (01-12-2020)	Observed price (01-12-2020)
1	50	1	1	32	20	0.05	0.16	0.98	20.2262	20.1616
2	50	1	1	32	5	0.13	0.23	0.94	19.8952	20.1616
3	100	1	1	32	20	0.18	0.39	0.92	20.5199	20.1616
4	100	1	1	32	1	0.16	0.26	0.93	20.0262	20.1616
5	50	2	2	32	20	0.04	0.13	0.98	20.1765	20.1616
6	50	2	2	1	1	0.15	0.27	0.93	19.8291	20.1616
7	100	2	2	1	1	0.32	0.54	0.86	19.6453	20.1616
8	100	2	2	1	3	0.28	0.46	0.88	19.7406	20.1616

Source*: Own elaboration by the authors.*

According to table 1, the reader may see that two models (1 and 5) present the best R^2; however, among both models, the one that offers the lowest MSE is model 5. We want to point out that the lower the MSE value, the better the model's performance. We use the MSE as a condition for the LSTM Recurrent Neural Network training process to obtain the lowest error value in each epoch or iteration. The model generates the indicated number of iterations (epochs) and aims to decrease the MSE, as observed in figure 3.

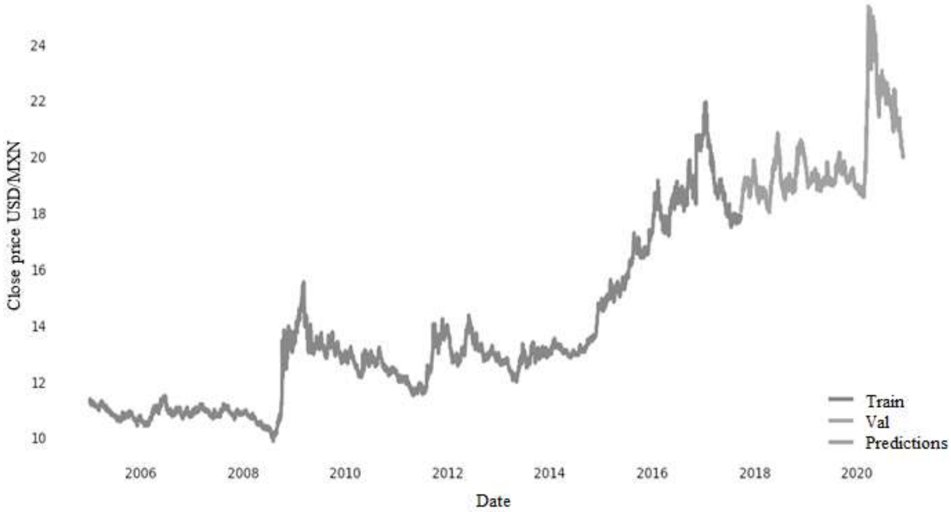
Figure 3. MSE decrease along with the epochs of the model.



Source*: Authors' output.*

Although both models contain the same number of neurons, the fifth model exhibits better accuracy because its two hidden and two dense layers reduce errors. Figure 4 compares the predicted versus actual closing prices for the Mexican Peso.

Figure 4. **LSTM training prices (Train), test prices (Val), and predicted (Predictions)**



Source*: Authors' output.*

Figure 4 shows how the LSTM Neural Network fits even under high volatility, in this case, the COVID-19 period. Since the information persists during the training epochs, the data obtained from the output in the previous layers become new information inputs for the next layer.

After showing the LSTM model's results, we present, in the following section, an ML algorithm, specifically Random Forest, to create a supervised learning model and analyze its efficiency and predictive power.

4.2 Modeling with Recurrent Neural Networks: LSTM specification

This section presents the results of the random forest algorithm, a supervised learning algorithm that uses ensemble learning, specifically, through the decision tree process. The metrics used to assess the model's efficiency are the same as those used on the LTMS model. We show its accuracy in Table 5.

Table 3. Structure of the random forest model and results

Model	N of trees	MSE	MAE	R^2	Estimated price (01-12-2020)	Observed price (01-12-2020)
1	10	0.5	0.35	0.78	20.1164	20.1616
2	50	0.53	0.36	0.77	20.1663	20.1616
3	100	0.52	0.35	0.77	20.1819	20.1616
4	1000	0.53	0.35	0.77	20.1457	20.1616
5	3000	0.53	0.35	0.77	20.1458	20.1616

Source: *Own elaboration by the authors.*

Table 3 suggests that the best model comprises ten trees, whit the lowest MSE, MAE, and goodness of fit confidence; this is interesting since although the number of trees was increasing gradually, the values of the assessment metrics did not improve. Hence, the number of trees is not the critical variable minimizing the errors. We found that the sample size causes the most significant sensitivity to the random forest setting.

Figure 5 shows the random forest forecast's accuracy for the MXN/USD exchange rate. We point out that the algorithm cannot adjust high volatility events. On the other hand, the model can predict "within a mean" values with low prediction error.

Figure 5. Random Forest training prices (Train), test prices (Val), and predicted (Predictions)

Source: *Authors' illustration.*

The following section presents an ARIMA model compared against the two previous models to establish which of them accurately predicts the closing price of the MXN/USD exchange rate.

4.3 ARIMA benchmark

For this section, we use the traditional Box & Jenkins (1970) Integrated Autoregressive Model of Moving Average (ARIMA), where the future value of MXN/USD is a linear combination of values and past errors. We stress out that the model will not capture any nonlinear feature. We used the exact specifications as in the LSTM and random forest models for the time series estimation.

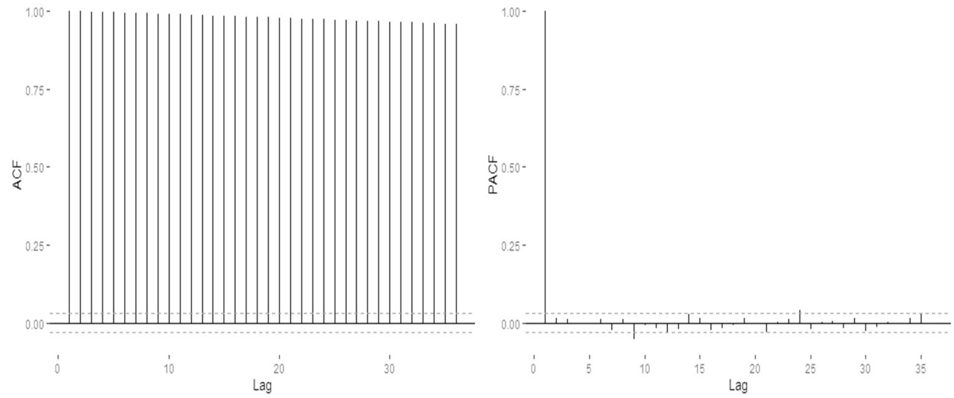
Figures 6 and 7 reveal non-stationarity for MXN/USD exchange rate in levels. Because of that, we applied the first difference to MXN/USD to achieve a weakly stationary series, satisfying the constant mean, variance, and covariance assumptions:

$$E(Y_i) = \mu \tag{4}$$

$$E(Y_t - \mu)(Y_t - \mu) = \sigma^2 \tag{5}$$

$$E(Y_{t1} - \mu)(Y_{t2} - \mu) = Y_{t1} - Y_{t2} \quad \forall t_1, t_2 \tag{6}$$

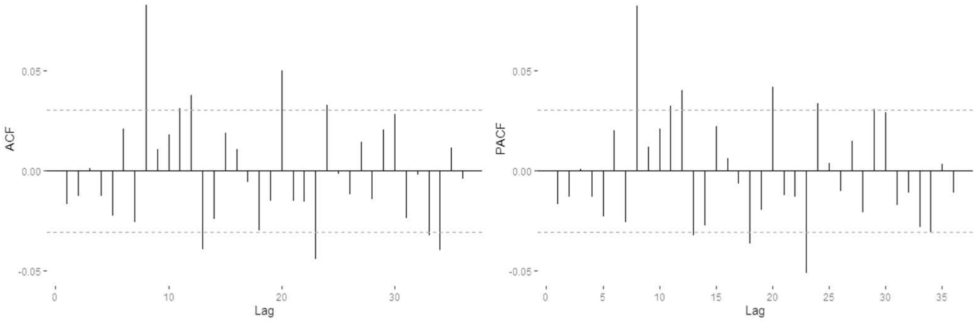
Figure 6. ACF and PACF for MXN/USD levels.



Source: *Authors' illustration.*

Revista de Investigación en Ciencias Contables y Administrativas
(Journal of Research in Accounting and Management Science)
Vol. 7, Num. 2, 2022

Figure 7. ACF and PACF for MXN/USD first difference.



Source: *Authors' illustration.*

In the MA component, we observed a memory structure in the eighth, thirteenth, and twelfth lag. We also found similar structures for the AR process. We developed and evaluated six ARIMA models to forecast the price of the Mexican currency. We show results for the unit-roots test, Ljung Box test (for residuals autocorrelation), MSE, MAE, and Akaike criterion. We started from the classical ARIMA (0,1,0) to less parsimonious models. We show the model's results in table 4.

Table 4. Structure of the ARIMA model and results[1].

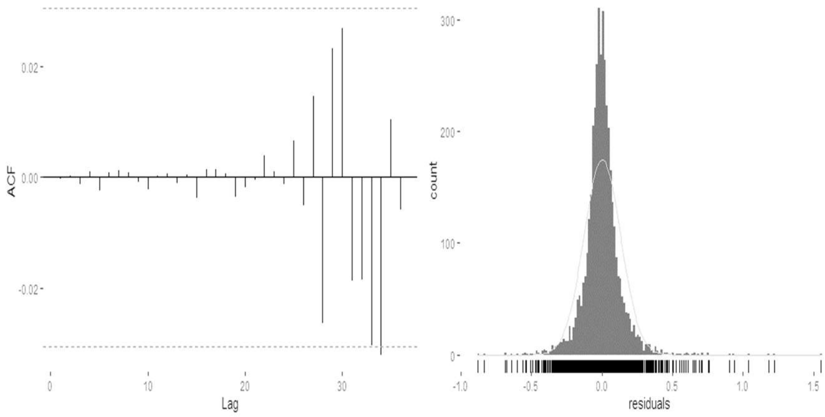
Model	ARIMA	Ljung-Box (p-value)	AIC	MSE	MAE	Estimated price (01-12-2020)	Observed price (01-12-2020)
1	(0,1,0)	0.000***	-5007.44	0.1321	0.0849	20.028	20.1616
2	(8,1,0)	0.000***	-5028.98	0.1315	0.0850	20.0288	20.1616
3	(8,1,8)	0.000***	-5044.58	0.1310	0.0847	20.0409	20.1616
4	(0,1,8)	0.000***	-5030.38	0.1315	0.0850	20.0406	20.1616
5	(23,1,0)	0.0552*	-5048.86	0.1307	0.0849	20.0263	20.1616
6	(24,1,0)	0.1402	-5051.63	0.1307	0.0848	20.0281	20.1616

Source: *Own elaboration by the authors.*

Table 4 suggests that model 6 (24,1,0) is the best, even being the less parsimonious. It did not exhibit correlation in the residuals and showed the lowest MSE and MAE; it also showed the best AIC criterion. Figure 8 shows the result of the ACF and the density function of the ARIMA (24,1,0).

[1] The Ljung-Box p-value is reported to test the null hypothesis of residuals independence (no autocorrelation). Symbols *, ** and *** expresses significance at 10%, 5% and 1% correspondingly.

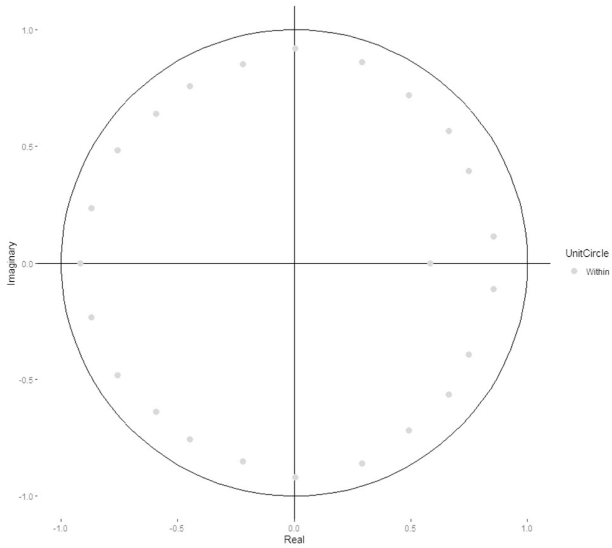
Figure 8. ACF and residual density.



Source: Authors' illustration**.**

We also present, in Figure 9, the inverse roots of the 24 lags used in the AR component. We point out that none of the lags is outside the unit circle, which guarantees a stable model over time.

Figure 9. Unit Circle stability test



Source: *Authors' illustration.*

Next, we present the forecast obtained by the model in figure 10. We take the last 100 data points from the time series of the MXN/USD currency with two confidence bands, the first of 80% and the second of 95%. The forecasted closing price obtained for December 1st, 2020, was 20.0281MXN; this is an 0.66% error concerning the actual value of that day.

Figure 10. Ten-day ARIMA (24, 1, 0) forecast (last 100 data sample).

Source: *Authors' illustration.*

5. Results and model comparations

This section presents the models' comparison between the three techniques' forecasts for the closing price of the MXN/USD. Table 5 shows the evaluation metrics.

Table 5. LSTM, random forest, and ARIMA metrics comparison.

Model	MSE	MAE	% Error
LSTM	0.04	0.13	0.07%
Random forest	0.50	0.35	0.22%
ARIMA (24,1,0)	0.13	0.08	0.66%

Source: *Own elaboration by the authors.*

According to table 5, the best prediction model for the MXN/USD exchange rate closing price is the Recurrent Neural Network of the LSTM type. Likewise, it is the model that presents the best evaluation metrics compared to random forest or ARIMA.

6. Conclusions

This paper aims to empirically find the best forecasting strategy between Machine Learning and Deep Learning models to verify their prediction power and, based on this, provide evidence for better decision-making regarding the path that variables can follow.

The proposed models go beyond conventional models such as an ARIMA. Our paper demonstrates that a Recurrent Neural Network of the LSTM type is the best model to forecast closing prices. The improved forecasting capabilities are due to its hidden and dense layer's structure, allowing a more comprehensive memory of the time series.

On the other hand, the model created with the random forest algorithm allowed identifying that the sample size is significant in Machine Learning models. Once we increase the data sample in the model, the algorithm can "learn" in a better way. The result of the random forest is like the ARIMA output since it exhibits underperformance under high volatility scenarios.

One of the limitations of this work is the data availability since ML and DL models require a large amount of data. Also, the optimizer selection for the Neural Network can vary, which may lead to different results. The previous opens the possibility of exploring lines of research that allow obtaining robust results from low-frequency time series (for example, daily data, as is the case in this research) or exploring the optimizer's selection criteria; however, this can be extended in other papers.

The main contribution of this work is that we showed that a Recurrent Neural Network of the LSTM type predicts with a minimum margin of error the MXN/USD exchange rate even in times of high volatility. Our research allows economic agents to arbitrage or hedge high volatility variables because its greater predictive power.

References

Ao, Y., Li, H., Zhu, L., Ali, S., & Yang, Z. (2019). The linear random forest algorithm and its advantages in machine learning assisted logging regression modeling. *Journal of Petroleum Science and Engineering, 174*, 776-789. doi:10.1016/j.petrol.2018.11.067

Bhagya, R., & Dash, G. (2020). Comprehensive study on applications of artificial neural network in food process modeling. *Critical Reviews in Food Science and Nutrition*, 1-28. doi:doi.org/10.1080/10408398.2020.1858398

BIS. (2019). *Triennial Central Bank Survey. Foreign exchange turnover in April 2019.* Basel: Bank for International Settlements.

Bonet, I., Sain, S., Rodríguez, A., & Grau, R. G. (2007). Redes Neuronales Recurrentes para el análisis de secuencias. *Revista Cubana de Ciencias Informáticas, 1*(4), 48-57.

Box, G. E., & Jenkins, G. M. (1970). *Time Series Analysis Forecasting.* San Francisco: Holden-Day.

Contreras, A., Llanes, A., Pérez-Bernabeu, A., Navarro, S., Pérez-Sánchez, H., López-Espín, J., & Cecilia, J. (2018). ENMX: An elastic network model to predict the FOREX market evolution. *Simulation Modelling Practice and Theory, 86*, 1-10. doi:10.1016/j.simpat.2018.04.008

Dash, R. (2020). Performance analysis of an evolutionary recurrent Legendre Polynomial Neural Network in application to FOREX prediction. *Journal of King Saud University - Computer and Information Sciences, 32*(9), 1000-1011. doi:10.1016/j.jksuci.2017.12.010

Esmaiel, A., Shahrazi, M., & Rasekhi, S. (2012). An investigation of Forex market efficiency based on detrended fluctuation analysis: A case study for Iran. *Physica A: Statistical Mechanics and its Applications, 391*(11), 3170-3179. doi:10.1016/j.physa.2011.12.045

Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine learning and deep learning. *Electron Markets*, 1-11. doi:10.1007/s12525-021-00475-2

Jubert, B., Neves, R., & Horta, N. (2018). Combining Support Vector Machine with Genetic Algorithms to optimize investments in Forex markets with high leverage. *Applied Soft Computing, 64*, 596-613. doi:10.1016/j.asoc.2017.12.047

Labañino, S., Valencia, H., & Toledano, O. (2019). Algoritmo Random Forest para la detección de fallos en redes de computadoras. *Serie Científica de la Universidad de las Ciencias Informáticas, 12*(8), 27-41.

Lee, V., & Wong, H. T. (2007). A multivariate neuro-fuzzy system for foreign currency risk management decision making. *Neurocomputing, 70*(4-6), 942-951. doi:10.1016/j.neucom.2006.10.025

Martín, B., & Sanz, A. (2006). *Redes Neuronales y Sistemas Borrosos* (3ra ed.). Madrid: Alfaomega.

Martínez, V., & Tse, Y. (2018). Intraday price discovery analysis in the foreign exchange market of an emerging economy: Mexico. *Research in International Business and Finance, 45*, 271-284. doi:10.1016/j.ribaf.2017.07.159

Mechelli, A., & Vieira, S. (2019). *Machine Learning: Methods and Applications to Brain Disorders* (Primera ed.). España: Academic Press.

Medina, F., & Ñique, C. (2017). Bosques aleatorios como extensión de los árboles de clasificación con los programas R y Python. *Interfases*(10), 165-189. doi:10.26439/interfases2017.n10.1775

Ni, L., Li, Y., Wang, X., Zhang, J., Yu, J., & Qi, C. (2019). Forecasting of Forex Time Series Data Based on Deep Learning. *Procedia Computer Science, 147*, 647-652. doi:10.1016/j.procs.2019.01.189

Rojas, C., & Herman, M. (2018). *Foreign exchange forecasting via machine learning.* Obtenido de http://cs229.stanford.edu/proj2018/report/76.pdf

Sánchez, N. (2015). Máquinas de soporte vectorial y redes neuronales artificiales en la predicción del movimiento USD/COP spot intradiario. *Observatorio de Economía y Operaciones Numéricas, 9*, 113-172. doi:10.18601/17941113.n9.04

Sandoval, L. (2018). Algoritmos de aprendizaje automático para análisis y predicción de datos. *Revista Tecnológica*, 36-40.

Villada, F., Muñoz, N., & García, E. (2012). Aplicación de las Redes Neuronales al Pronóstico de Precios en el Mercado de Valores. *Información tecnológica, 23*(4), 11-20. doi:10.4067/S0718-07642012000400003