ISSN: 2093-5374 / E-ISSN: 2093-5382

Genetic Algorithm and LSTM Artificial Neural Network for Investment Portfolio Optimization

Gustavo A. Flores-Fernandez[1*], Miguel Jimenez-Carrion[2], Flabio Gutierrez[3], and Raul A. Sanchez-Ancajima[4]

[1*]Graduate from the Faculty of Industrial Engineering at the Universidad Nacional De Piura, Perú.
fgustavoalexis@gmail.com, https://orcid.org/0000-0002-4488-4325

[2]Professor, Facultad De Ingeniería Industrial, Universidad Nacional De Piura, Castilla-Piura, Perú.
mjimenezc@unp.edu.pe, https://orcid.org/0000-0001-9632-5085

[3]Professor, Universidad Nacional De Piura, Piura, Perú.
flabio@unp.edu.pe, https://orcid.org/0000-0003-0372-3864

[4]Professor, Universidad Nacional De Tumbes, Tumbes, Perú.
rsanchez@untumbes.edu.pe, https://orcid.org/0000-0003-3341-7382

Received: January 10, 2024; Revised: February 17, 2024; Accepted: April 08, 2024; Published: June 29, 2024

Abstract

The present research aimed to construct a genetic algorithm and artificial neural network to optimize investment portfolios, considering that in modern investment portfolio theory, optimization is a multi-objective problem involving maximizing return and minimizing volatility, also known as risk. This opens up the possibility of a highly combinatorial solution space, making it a computationally complex problem that cannot be solved by deterministic algorithms. To achieve the objective, 255 companies operating within the Peruvian national market and listed on the Lima Stock Exchange were evaluated. The research resulted in a mean squared error of 6.33%, a mean absolute error of 5.07%, and an accuracy of 92.35% related to the artificial neural network, indicating an acceptable generalization capacity for predicting positive trends in the stocks to be used as inputs for the genetic algorithm. Regarding the genetic algorithm, a quality function was successfully modeled, considering 5 factors related to the profitability and volatility of the stocks, as well as portfolio diversification. Ultimately, the best configuration of the genetic algorithm was found to have a fitness value of 0.772482, translating to a return of 1.00058% and volatility of 0.00612%. It is concluded that the genetic algorithm optimizes investment portfolios by achieving higher returns and lower volatility compared to other methods, with volatility specifically being a much lower percentage.

Keywords: Metaheuristics, Genetic Algorithms, Artificial Neural Networks, Optimization, Investment Portfolios.

1 Introduction

The productive units that drive local and national economies have as their main need the planning of their investments to better adapt to demand; this generally implies evaluating, over a planning horizon,

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), volume: 15, number: 2 (June), pp. 27-46. DOI: *10.58346/JOWUA.2024.I2.003*

*Corresponding author: Graduate from the Faculty of Industrial Engineering at the Universidad Nacional de Piura, Perú.

the future scaling of infrastructure or operations through an investment project evaluation (Baca, 2016; Podvalny et al., 2021; Tunga et al., 2021). Often, the evaluation of investment projects does not consider within its projected income statement, realistic scenarios and specific details inherent to the operations of the productive units, such as the acquisition of fixed assets in transitional periods of business scalability, intangible investments, etc. As mentioned by (Hart & Zingales, 2011; Trivedi et al., 2023), the need to invest in assets, often not accounted for in financial statements, highlights the lack of tools to improve investment efficiency, demonstrating the underutilization of excess liquidity in the productive unit, which would lead to the accumulated net flows reflecting business dynamism more adequately and avoiding negative returns without attempting to improve efficiency in external investments (Oleksandr et al., 2024; Mumtaj Begum et al., 2022).

The project manager facing an excess of liquidity or wealth in their projected cash flows must make a sound decision to improve project efficiency, which entails deciding whether to consider opportunity costs, among which could be keeping the money, avoiding or assuming inflationary changes, or making a risk-free investment in a financial institution (Udayakumar et al., 2023). Conversely, as indicated by (Ames, 2012; Fuw et al., 2011), investing in the capital market is an option for project managers willing to take risks, where they can find a store of value, potentially resulting in higher returns than investing in a financial institution. Faced with the challenges encountered by the project manager, the best option must be taken for investment, and this decision constitutes an NP-complete problem, as stated by (Didier et al., 2021; Srinadi et al., 2023), considering the option of generating a return in the capital market, given that there is risk, it is therefore crucial to efficiently select assets or evaluate the optimal investment portfolio, consequently projecting a higher return on investment and minimizing the assumable risk related to the volatility of those assets. Considering that each capital asset will have a percentage of investment from the excess returns of the projected flows, there would be a highly combinatorial process of options. For example, if 5 assets were evaluated, there would be 10,000,000,000 combinations, being the percentage representations. Additionally, if each combination were evaluated in an objective function with a time delay of 1 second, an exponential time would be required to evaluate all combinations for 10 or 20 assets. Therefore, this problem with two opposing objectives – maximization in the case of return and minimization in the case of risk – cannot be solved in all its combinations in a reasonable time using a deterministic algorithm, and considering its combinatorial nature, it is ultimately categorized as a "Non-polynomial" problem (Gutiérrez et al., 2020; Asadov, 2018).

While it is true that in the evaluation of investment portfolios there is a tendency to perform portfolio evaluation with a certain number of combinations to represent the Pareto frontier and use the Sharpe ratio to find the optimum, there is no deterministic solution that contemplates and iteratively evaluates a large number of efficient solutions, much less that contemplates the optimization of both objectives simultaneously to find feasible solutions (Veerasamy et al., 2023; Sravana et al., 2022; Srinivasareddy et al., 2021). Therefore, the use of artificial intelligence tools such as artificial neural networks and metaheuristics such as genetic algorithms becomes vitally important, as indicated by (Das et al., 2023; Li et al., 2023; Praveenchandar et al., 2024), who have demonstrated fundamental usefulness in the field of finance due to their predictive effectiveness; and the ability to make exhaustive iterations in search spaces, through the exploration and exploitation of feasible individuals, finding optimal solutions to complex problems by defining the correct hyperparameters, such as crossover probabilities, mutation, and number of generations, which serve to control the behavior and efficiency of the algorithm, allowing optimal individuals to be found in a reasonable time.

As background, authors such as (Qi Li et al., 2023) focus their research on presenting and developing an improved approach that combines Symbolic Genetic Algorithm (SGA) with Long Short-Term

Memory (LSTM) Neural Network to anticipate stock returns in the Chinese market. Results ranged from 15.26% to 22.35% annual profitability.

On the other hand, (Bo, 2023) aimed to optimize portfolio composition, reduce risks, and increase potential returns by analyzing interdependencies and correlations between financial assets using complex networks, and also using genetic algorithms as an optimization technique. As results, the author obtained a return of 0.2667 and a Sharpe ratio of 0.0685.

Vasiani et al., (2020) sought to optimize stock portfolios using the priority index method and genetic algorithms. Their methodology involves selecting stocks based on the priority index, considering parameters such as the price/earnings ratio (P/E), earnings per share (EPS), wealth creation, undervaluation, and price/earnings-to-growth ratio (PEG). Stocks in each sector are chosen based on a priority index score equal to or greater than the minimum score of the selected stocks. Results showed a maximum profitability of 14.08%, however, it should be noted that only the return is considered as the main metric.

Lim et al., (2020) formulated the main objective of designing an optimal portfolio using a Genetic Algorithm (GA) that incorporates momentum and asset valuation strategies. Their methodology involves analyzing risk-adjusted returns in previous periods, using the momentum of these returns as momentum for stock selection. However, they acknowledge that historical movements alone are not sufficient to predict future changes or guarantee positive returns. Moreover, the authors did not consider the use of specialized artificial intelligence techniques such as LSTM networks.

Candia et al., (2020) addressed the problem of project portfolio selection for the awarding of public works through open merit competitions supervised by the National Roads Institute (INVIAS) in Colombia. Methodologically, they evaluated two alternative approaches: a meta-optimized genetic algorithm (GA) whose average fitness was 0.21748 with an execution time of 8.102 minutes, outperforming the meta-optimized adaptive probabilistic greedy search procedure (GRASP).

Rodríguez et al., (2020) proposed an alternative method using evolutionary algorithms, specifically a Canonical Genetic Algorithm, to design a currency investment portfolio called "currency portfolio". Descriptively, they selected six currencies in relation to the Mexican peso, Paraguayan guaraní, Uruguayan peso, Bolivian boliviano, US dollar, British pound, and euro. The authors obtained a return of 0.049202% and a volatility of 0.003543% as a result.

Maholi et al., (2019) aimed to predict future stock values using artificial neural networks (ANN) and then use a genetic algorithm (GA) to form optimal portfolios that maximize returns and minimize risk. Their methodology begins by applying ANN, a machine learning model represented by a dense neural network, which it should be emphasized are not specialized networks for time series evaluation. Results showed a mean squared error of 5.60% for a single time step, and in terms of profitability, a return of 1.42% was obtained considering a volatility of 0.15%.

Liagkouras, (2019) aimed to address the limitations of existing evolutionary algorithm techniques in solving large-scale combinatorial problems due to their extensive search space. Their methodology involves testing the performance of the proposed algorithm in the optimal allocation of limited resources to a series of competitive investment opportunities to optimize objectives. Experimental analysis resulted in a performance not exceeding 0.785 and a volatility of 0.0097.

2 Methodology

The steps for the development of the present research consisted of: 1) Obtaining information from all the companies listed on the Lima Stock Exchange, which were 255, whose data is openly accessible and was obtained through the institution's website via CSV files, 2) Performing an exploratory data analysis, which involved preprocessing by identifying outliers and data imputation through linear interpolation, 3) Designing a Long Short-Term Memory (LSTM) artificial neural network to predict the stock values trend of the companies, 4) Designing a genetic algorithm to optimize the investment portfolio based on the predictions of the LSTM neural network, 5) Developing the discussion of the results, and 6) Defining the conclusions based on the results. (Figure 1)

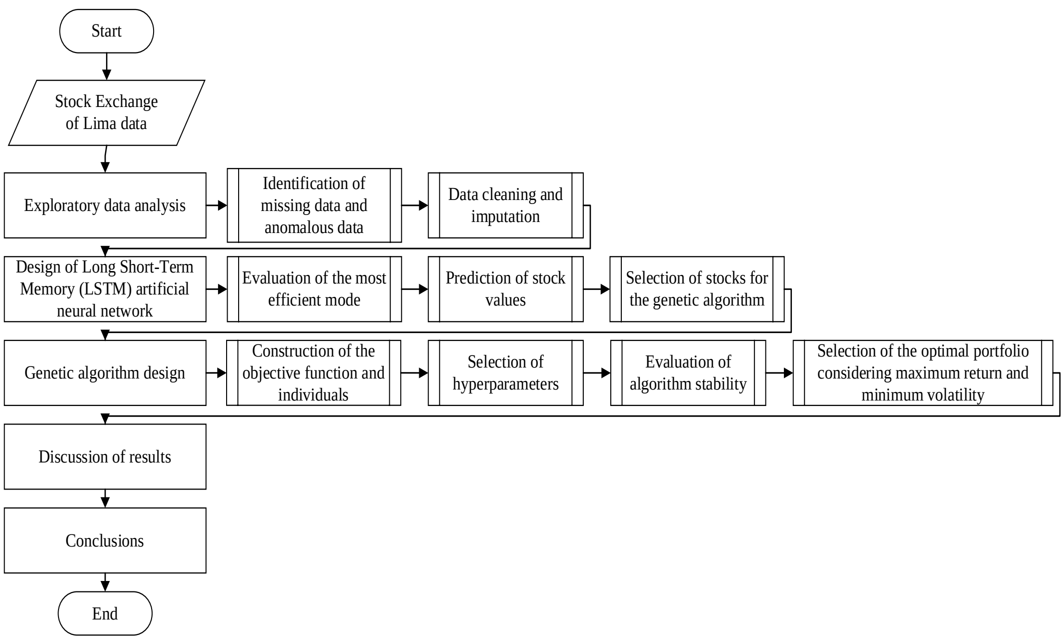


Figure 1: Flowchart of the Applied Methodology

3 Results

Exploratory Analysis for Lima Stock Exchange Data

To obtain inputs for the neural network model, a data preprocessing was developed to standardize the data and make it useful for the network to generate consistent data in its learning and general use process. Therefore, the process is guided by the procedure described by (Ruochen & Muchao, 2021), starting with handling missing data, which is a fundamental feature for them to be inputs for the neural network. For the data acquisition process, data was collected at a single point in time, and assets were gradually discarded based on the data they presented and filtered. Initially, the BVL portal listed records and data for 255 companies, of which only 85% (216) provided information on stocks. Exploring the CSV files revealed that only 57% of the total (145) companies had historical data, while the rest only provided the date when market behavior parameters should have been recorded. Next, the widest time range of the data was determined, i.e., the lower and upper bounds. It was found that companies showed data starting from January 2, 2012, representing 38.43% (98) of the companies, and 18% (46) had stock quote data from previous years, so the earlier date was set as the lower limit. Regarding the upper limit, since the research year had not yet ended, January 2, 2023, was considered to ensure symmetry in complete periods. Subsequently, each record was evaluated, revealing that 85 of the filtered companies had a maximum data range of 2764 records, of which only 13 of the companies had missing dates. Then, companies with this amount of records were evaluated, and it was found that asset parameter data was missing. The percentage of missing data was then evaluated to establish a tolerance that would not affect the model's behavior in design. According to Table 1, based on the 9 parameters that expose stock behavior, namely, opening price, closing price, highest price, lowest price, average price, volume traded, traded amount, previous date, adjusted closing (closing price of a stock adjusted to account for events affecting stock price), they were grouped into 10 ranges representing the amount of missing data. This was done to understand the percentage of missing data in the total of the 86 companies evaluated, and to maintain consistency with regard to the amount of missing data in accordance with authors like Dagnino, (2014) and Bennett, (2001), who recommend an acceptable tolerance margin of 10%. Considering that a higher percentage of absence can generate biases or loss of natural data distribution, this is represented in Table 1 where 10 ranges are presented, each representing a 10% interval of missing data. For example, in R1, companies with between 0 and 10% missing data are included. In this case, for the parameter "Volume traded", the 12% indicates that there are 12% of companies with between 0 and 10% missing data in that parameter, and 58% of companies have missing data between 0 to 10% in the "Previous date" parameter, and so on.

Table 1: Percentage of Companies According to the Amount of Missing Data

Data Range	Open	Close	High	Low	Average	Traded Quantity	Traded amount (S/)	Previous date	Corrected previous close
R_1	8%	8%	8%	8%	12%	12%	12%	58%	59%
R_2	6%	6%	6%	6%	4%	4%	4%	19%	19%
R_3	4%	4%	4%	4%	6%	6%	6%	9%	11%
R_4	4%	4%	4%	4%	1%	1%	1%	5%	4%
R_5	0%	0%	0%	0%	5%	5%	5%	6%	6%
R_6	6%	6%	6%	6%	5%	5%	5%	0%	0%
R_7	8%	8%	8%	8%	6%	6%	6%	0%	0%
R_8	8%	8%	8%	8%	11%	11%	11%	1%	1%
R_9	13%	13%	13%	13%	11%	11%	11%	1%	0%
R_{10}	45%	45%	45%	45%	42%	42%	42%	2%	2%

Given that the maximum allowable percentage of missing data is 10%, only companies in range 1 will be considered. In this range, only the adjusted closing parameter has a significant percentage of 59%, representing 50 companies in the first interval, whose missing data is below 10% specifically, with 24 companies having missing data and 26 having complete data records. Therefore, it is determined that the parameter to be used to generate the neural network projection will be the adjusted closing, defining the use of a multivariable LSTM recurrent neural network model.

Within the data, no outliers were found; however, 24 companies with missing data were identified. For this, imputation was evaluated using linear interpolation and also the K-means method, which is a machine learning algorithm that, for the specific case of imputation, generates a data point based on the k nearest neighbors to the missing data point, as recommended in the review by (Fang & Wang, 2020).

To evaluate which was the best imputation method, a comparison was made between the two methods used to determine which has the highest number of companies with a better R2 coefficient (Coefficient of determination). The results after imputation in relation to R2 are shown in Table 2.

Table 2: Comparison of Determination Coefficients of Data Imputation Methods

Ticker	Model KNN	Interpolation
CARTAVC1	0.0293	0.0219
BUENAVC1	0.2269	0.2241
BACKUAC1	0.4843	0.5402
AIHC1	0.6604	0.6868
FALABEC1	0.5235	0.5271
EXALMC1	0.0598	0.0601
ELCOMEI1	0.8295	0.8646
CREDITC1	0.462	0.4627
LAREDOC1	0.8067	0.858
HIDRA2C1	0.7759	0.7802
GLORIAI1	0.3341	0.341
PERUBAI1	0.1559	0.1594
PHTBC1	0.4889	0.474
POSITIC1	0.1931	0.1992
SNJACIC1	0.1185	0.1139
TUMANC1	0.4978	0.4959
TEF	0.8633	0.869
SPCCPI1	0.0015	0.0001
SIDERC1	0.4874	0.4878
EGEPIBC1	0.0015	0.0002
PODERC1	0.6065	0.7313
SAGAC1	0.4295	0.4173
HIDROSI1	0.6927	0.7642
GBVLAC1	0.6884	0.8218
Percentage of companies with a higher coefficient of determination	33%	67%

Taking into account that in the interpolation model, 67% of the companies show a higher coefficient of determination, this model was chosen as the imputation method.

Designing a Neural Network for Stock Prediction as Input for the Genetic Algorithm

To predict the closing price of the stock, a recurrent neural network was designed, which, due to its characteristics of using a previous state and being able to generate the next state, is suitable for time series models where they have shown their utility in multiple artificial intelligence tasks (Hewamalage et al., 2021). In the case of the Lima Stock Exchange, the tool used is the moving average through technical analysis, as confirmed by accessing the official portal of this entity. In the case of the present study, the design was proposed based on the type of recurrent neural networks called Long Short-Term Memory (LSTM), which have the ability to remember through memory loss gates that, through nonlinear functions, eliminate or preserve information, input gates that decide which information is accepted by the model, and output gates where it is decided which part of the LSTM memory contributes to the output (Siami-Namini et al., 2019).

The structure of an LSTM network for this case considers the corrected closing value of the stock valuation time series with its respective normalization processing as mentioned by (Coaquira Velásquez et al., 2023), as discussed in the exploratory analysis. Its output will be a measure of trend defined under the concept that the trend is bullish if its trading value is higher than in the previous period. However, this expression must be complemented with a margin measure to improve the accuracy of the threshold that can be accepted as a bullish trend value, based on the analyzed data. Therefore, it was determined that the expression would be complemented according to Equation 1:

$$P_{(t+1)} > P_{(t)} + [\sigma_{\%Market}]^{*} P_{(t)} \qquad \text{Eq. (1)}$$

Where:

$P_{(t)}$: Closing price of the stock at period (t)

$P_{(t+21)}$: Closing price of the stock at period (t+21)

σ_{Market}: Percentage representation of the standard deviation of the market

According to the previous expression, the measure of market standard deviation is used so that the trend has a surplus to absorb any variation that exists due to any atypical variable within the international market, considering that between 20% and 50% of price behavior is generated from market forces (Gitman & Joehnk, 2009). Therefore, inserting a measure into the equation that considers the fluctuations that the general behavior of stocks has been experiencing helps mitigate any condition that does not guarantee superior performance in a subsequent time frame, which is also defined after day 21 as short-term expected returns (Cáceres, 2018). Similarly, regarding the indicator used to determine this deviation measure, the Standard & Poor's 500 (S&P500) was used, which is defined as a genuine index by considering the capitalization of the largest 500 companies in the United States and capturing 80% of all capitalization.

Univariate-Multistep Algorithm

Based on the previously formulated approach shown in Algorithm 1, four LSTM network topologies were proposed for evaluation. The considered hyperparameters include the network input, which consists of batches of 42 days, and an output of 21 days, representing the closing value of the stock for each company as shown in Table 3, with a total of 138,200 records. Additionally, the number of neurons in each layer of the network for each topology, the number of epochs, and the learning rate were considered, as shown in the following Table 4.

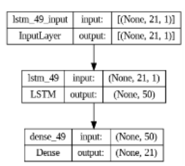
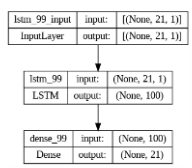
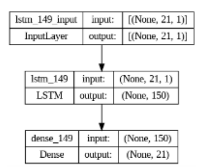
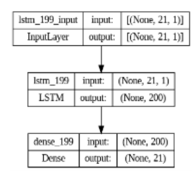
Table 3: Closing Values of Stocks for Each Company

Data Time	Mnemonic					
	CASAGRC1	CARTAVC1	. . .	SAGAC1	HIDROSI1	GBVLAC1
2/01/2012	S/ 15.15	S/ 23.00	. . .	S/ 4.25	S/ 0.60	S/ 13.50
3/01/2012	S/ 15.20	S/ 23.00	. . .	S/ 4.25	S/ 0.60	S/ 13.50
4/01/2012	S/ 15.30	S/ 23.00	. . .	S/ 4.25	S/ 0.60	S/ 13.50
.	.	.	. . .	.	.	.
.	.	.		.	.	.
.	.	.	. . .	.	.	.
29/12/2022	S/ 8.00	S/ 35.00	. . .	S/ 8.97	S/ 0.22	S/ 2.45
30/12/2022	S/ 7.85	S/ 35.10	. . .	S/ 8.97	S/ 0.22	S/ 2.60
2/01/2023	S/ 7.90	S/ 35.10	. . .	S/ 8.97	S/ 0.22	S/ 2.60

```
Input: Libraries
Input: Stocks from exploratory analysis
Input: Hyperparameters
Output: Algorithm performance metrics
Data frame ← Stock values
tr, vl, tst ← training, validation, and test sets
function DATA SCALING
    Scaled training set ← MinMax Scaler(training set)
    Scaled validation set ← MinMax Scaler(validation set)
    Scaled test set ← MinMax Scaler(test set)
end function
for companies in the data frame do
    function TRAIN MODEL
        Model ← Sequential
        Model ← add LSTM layer
        Model ← add Dense layer
    end function
    function PERFORMANCE METRICS
        RMSE, MAE, Accuracy ← Model performance
    end function
end for
Output: Model performance metrics
```

Algorithm 1: Univariate-Multistep LSTM Model

Table 4: Comparison of LSTM Network Models

Characteristics of model	Model 1	Model 2	Model 3	Model 4
Number of processing units	50	100	150	200
Epochs	200	300	400	500
Learning rate factor	0.0002	0.0003	0.0004	0.0005
Execution time optimized with GPU	35 minutes	31 minutes	33 minutes	58 minutes
Topology de la red				

Based on the three parameters considered for the network evaluation, we have the mean squared error, the mean absolute error, and the accuracy. After training the network, the following results were obtained, as shown in Table 5.

Table 5: Comparison of Model Metrics

Metrics	Model 1	Model 2	Model 3	Model 4
RMSE_train	3.85%	3.75%	3.69%	3.57%
RMSE_val	5.81%	5.47%	5.37%	5.16%
RMSE_test	7.25%	6.77%	6.76%	6.33%
MAE_train	3.09%	2.97%	2.89%	2.75%
MAE_val	4.71%	4.33%	4.19%	3.94%
MAE_test	6.21%	5.67%	5.56%	5.07%
Accuracy_train	96.58%	96.85%	96.90%	97.08%
Accuracy_val	96.81%	96.56%	97.00%	97.14%
Accuracy_test	90.46%	91.21%	91.98%	92.35%

As observed in the previous table, Model 4 achieves a lower mean squared error compared to the other models in its test set, where the network's generalization ability was tested with data exhibiting efficient behavior with data it has never seen before. Additionally, the mean absolute error was considered and the accuracy metric was compared with the other models, resulting in an accuracy of 92.35%. Therefore, Model 4 is chosen to evaluate the trend of stock prices of companies that will serve as input for the genetic algorithm.

Having identified the most efficient LSTM neural network architecture, we proceeded to use Equation 1, the application of which is illustrated as an example below for the case of the company with mnemonic CASAGRC1:

$$8.52 > 7.9 + 7.9 * (1.08\%)$$

Market standard deviation

Last closing price

Prediction of the closing price 21 days prior

$$8.52 > 7.98532 \text{ if it meets the constraint}$$

Upon conducting this procedure on the 50 companies, it was determined that only 24 capital assets meet the condition, considering companies with an upward trend over a period of 21 days and absorbing market variations, which will improve the performance of the investment portfolio. Therefore, these assets will serve as inputs for the genetic algorithm.

Modeling the Elements of the Genetic Algorithm

The genetic algorithm structure must include data standardization to allow processing, starting from the structure of individuals or potential solutions through chromosomal representation, as well as its fitness function that will evaluate the quality of each individual so that each of the candidates with better characteristics generates offspring and better solutions.

For this section, it is considered that there are 24 capital assets, so the sample space is denoted as follows:

$$\Omega = \{S \subseteq \{A_1, A_2, \dots, A_{24}\} \,/\, |S| \geq 1\} \qquad \text{Eq. (2)}$$

Where:

Ω: Sample space of all capital assets.
S: Subset of available assets.
|S|: Restriction ensuring that each portfolio has at least one asset.

In equation 2 for the genetic algorithm optimization, each individual represents a solution to the proposed problem, which means that each chromosome contains the capital percentages to be distributed among each of the assets. For this research, 24 capital assets have been identified for evaluation within the genetic algorithm; therefore, individuals will be represented by a vector with percentages.

Its representation is as follows equation 3, 4:

$$C = \left[w_1, w_{2,\dots}, w_n\right] \qquad \text{Eq. (3)}$$

Where:

n: Number of assets.
w_i: The weight or percentage representation of investment assigned to asset i This vector C must be subject to the following constraints:

$$0 \leq \omega_i \leq 1 \text{ para todo } i$$

$$\sum_{i=0}^{i=n} \omega_i = 1 \qquad \text{Eq. (4)}$$

Therefore, individuals will have the following chromosomal form, where each assigned weight will be a gene, as shown:

W_1	W_2	W_3	W_4	W_5	.	.	.	W_{24}

Therefore, the genotype will consist of percentage numbers, and the phenotype will have the same representation.

Objective Function Modeling

The objective function aims to incorporate multiple elements or factors representing the optimization goal, as existing theories suggest that there are systematic factors or others that help generate performance and volatility behavior with greater accuracy. For this particular study, the factors considered are return, volatility, Sharpe ratio, market risk premium, and portfolio diversification, considering that investors will have a predisposition or confidence in a portfolio that generates higher returns per unit of risk in equation 5.

The objective function is constructed taking into account the following elements:

$$F.\ O = Fitness = Max\ (\text{Return}) + Min\ (\text{Volatility}) = MAX \sum \lambda_i F_j \qquad \text{Eq. (5)}$$

Where:

λ_i : Weight assigned to each factor of the objective function.

F_j : Factor j considered within the objective function.

As mentioned earlier, access to information is limited to asset behavior data, so return and volatility factors will be present within the equation as relevant factors expressed as follows:

$\lambda_1 R_P$: Portfolio return.

$-\lambda_2 \sigma$: Portfolio volatility, which in this case is negative because it penalizes the objective function considering that the investor seeks to minimize it.

Another factor considered is the investor's inclination to allocate resources to portfolios that maximize return per unit of invested risk, with the Sharpe ratio being this indicator shown in equation 6:

$$\lambda_3 \left(\frac{R_p - R_f}{\sigma}\right) \qquad \text{Eq. (6)}$$

For this factor, a specific constraint should be considered, only taking this factor into account if the portfolio return is greater than the risk-free rate:

If $R_p \geq R_f$ then: $\lambda_3 \left(\frac{R_p - R_f}{\sigma}\right)$

Otherwise: $\lambda_3 = 0$

Another relevant factor considered is the market risk premium, for which, lacking a known formula, a sigmoid function was considered. This function takes into account that as the difference between the portfolio's return and the market's return increases, the value of this function tends towards 1, and as it decreases, it approaches its asymptotic value of 0, as shown in the following figure 2:

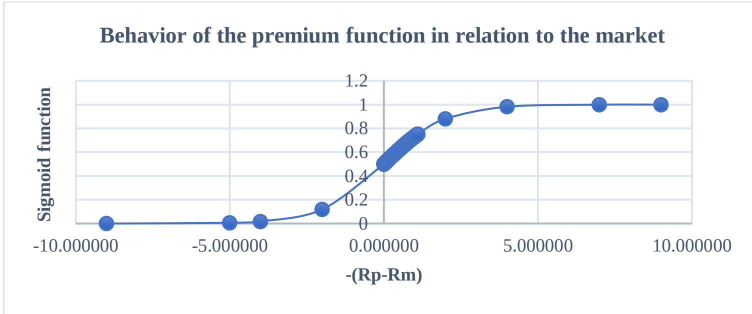


Figure 2: Representation of the Market Risk Premium Function

The behavior of this function was expressed through a simulation of a normal distribution based on the data, within a range for the market risk premium between -10 to 10. It had a mean of 0.55 and a standard deviation of 0.38. The Kolmogorov-Smirnov goodness-of-fit test resulted in a p-value of 85.57%. In the shaded area highlighted in the figure, there is a concentration of values where the portfolio's return is greater than the market's return. As the premium relative to the market becomes positive, the values tend to increase and approach the value of one. Conversely, if the difference is smaller, the values approach 0. Therefore, the function adheres to the requirements for the objective function and can be expressed as follows equation 7:

$$\lambda_4\left(\frac{1}{1+e^{-(Rp-Rm)}}\right) \qquad \text{Eq. (7)}$$

Finally, the last factor to consider is portfolio diversification, expressed as the average of the correlation of the assets comprising it, which is related to a penalty or reward measure to the objective function. In equation 8 shows the mathematical expression is as follows:

$$\lambda_5 \frac{2\rho_{ij}}{n(n-1)} \qquad \text{Eq. (8)}$$

Where:

ρ_{ij}: is the correlation between asset i and j that make up the portfolio.

$\frac{n(n-1)}{2}$: is the total number of combinations of assets comprising the portfolio.

Considering all these factors, the following objective function is formulated:

$$MAX\ (Objective\ Function)\ = \frac{\lambda_1 R_P - \lambda_2\sigma + \lambda_3\left(\frac{R_P - R_f}{\sigma}\right) + \lambda_4\left(\frac{1}{1+e^{-(Rp-Rm)}}\right)}{\lambda_5\left(\frac{2\rho_{ij}}{n(n-1)}\right)} \qquad \text{Eq. (9)}$$

Where:

λ_i: Weights assigned to each factor according to their level of representativeness or importance in relation to maximizing the objective function for i =1,2,3,4,5, the values for the weights assigned in a genetic algorithm selection process were established according to the researcher's criteria in equation 9. In relation to this, a Monte Carlo simulation was conducted using the Risk Simulator tool to verify the sensitivity of these factors with 10,000 simulations. For this particular case, the coefficient found will be used as a reference to weigh each of the lambda values. As can be observed, in coherence with the particularity of each associated factor, the lambdas related to volatility and diversification have negative values.

An empirical weighting of the lambdas was performed using a scale from 0 to 1 with a precision of two floating points. These values are proportional to their level of correlation, as shown in table 6:

Table 6: Behavior of Lambda Variables in Terms of their Correlation and Variation Associated with Fitness

	Correlation	**% Allocated**
Lambda 1	0.0096	1.0194%
Lambda 2	0.0076	0.8028%
Lambda 3	0.0159	1.6925%
Lambda 4	0.0016	0.1741%
Lambda 5	0.9069	96.3112%
Total	0.94	100.0000%

Considering these values, the equation 10 will have the following representation:

$$MAX(F.O) = \frac{0.0102 * R_P - 0.008 * \sigma + 0.017 * \left(\frac{R_p - R_f}{\sigma}\right) - 0.0017 * \left(\frac{1}{1 + e^{-(Rp-Rm)}}\right)}{0.963 * \left(\frac{2\rho_{ij}}{n(n-1)}\right)} \quad \text{Eq. (10)}$$

Hyperparameters of the Genetic Algorithm

Determining hyperparameters is a fundamental stage in the development, design, and evaluation of the genetic algorithm. For this specific case, certain configurations based on data type, problem environment, and researcher experience were used. For the initial population size, the algorithm is evaluated in three different search spaces of 1000, 2500, and 5000 individuals, and the number of generations ranges from 50, 100, and 500, adjusted depending on the algorithm's convergence. Additionally, the crossover probability is set to 80%, considering that the algorithm's floating-point numbers are individuals and cover a wide search space. Regarding the mutation probability, a specific 9% is chosen for this case. In terms of the selection method, the tournament mechanism is used, where a specific number of three individuals are chosen and evaluated. The crossover operator used is the CxBlend, known for generating new individuals that extensively explore search spaces by producing offspring with characteristics of both parents in a linear manner, fitting the problem type. The mutation operator used is Gaussian mutation with a mean of 0 and a standard deviation of 0.01, as it is required to generate individuals with a percentage mutation point.

Genetic Algorithm Evaluation

The model processed through algorithm 2 features the fitness function proposed in this study and was analyzed using an experimental design with a 2x3 factorial arrangement. The factors considered are the initial population size, referred to as TPo, and the number of generations, denoted as NG, each with 3 levels: 50, 100, and 500 for TPo, and 1000, 2500, and 5000 for NG, respectively. For each factorial combination, 4 repetitions were conducted, as depicted in table 7:

Input: Libraries.
Input: Market asset database.
Hyperparameter assignment: crossover probability, mutation probability $\leftarrow 0.8, 0.09$.
Assignment: Rf, Rm $\leftarrow 0.002803, 0.06775$.
Assignment: Initial population, Number of generations $\leftarrow 50, 2500$.
Assignment: $\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5 \leftarrow 0.0096, 0.0076, 0.0159, 0.0016, 0.9069$
Function calculate portfolio return.
Function calculate volatility.
Function calculate average correlation.
Function calculate objective function.
Function create individual.
Function calculate stability.
Genetic Algorithm Function
Problem Definition (Maximization).
Evaluate individuals

$$MAX(F.O) = \frac{\lambda_1 R_P - \lambda_2 \sigma + \lambda_3 \left(\frac{R_p - R_f}{\sigma}\right) + \lambda_4 \left(\frac{1}{1+e^{-(R_P - R_M)}}\right)}{\lambda_5 \left(\frac{2\rho_{ij}}{n(n-1)}\right)}$$

(1)

Perform CxBlend Crossover.

$$D_i = P_{1i} + \ 0.8 \cdot (P_{2i} - P_{1i}) \tag{2}$$

Perform Gaussian Mutation.

$$mutGaussian(x, \mu, \sigma, pb) = x + N(\ 0, .01^2) \cdot 0.09 \tag{3}$$

Perform Tournament Selection.

$$selTournament_3(population, tournsize) = Best_1 \cup Best_2 \cup Best_3 \tag{4}$$

Print the best fitness and the best individual.

Algorithm 2: Genetic Algorithm

Table 7: Repetitions for each Configuration of the Genetic Algorithm

NG	Tpo	Repetitions			
		I	II	III	IV
50	1000	0.726487	0.752226	0.752226	0.752226
50	2500	0.772482	0.772482	0.772482	0.772482
50	5000	0.772482	0.772482	0.772482	0.772482
100	1000	0.752226	0.752226	0.752226	0.752226
100	2500	0.772482	0.772482	0.772482	0.772482
100	5000	0.772482	0.772482	0.772482	0.772482
500	1000	0.752226	0.752226	0.752226	0.752226
500	2500	0.772482	0.772482	0.772482	0.772482
500	5000	0.772482	0.772482	0.772482	0.772482

After generating the repetitions, an analysis of variance was conducted, yielding the following results shown in table 8:

Table 8: Analysis of Variance

Source of Variation	Gl	SC	CM	F	Sig.
NG	2	0.000037	0.000018	1	0.3811
TPo	2	0.004014	0.002007	109.0714	**0.0000
NG*TPo	4	0.000074	0.000018	1	0.4247
Error	27	0.000497	0.000018		
Total	35	0.004622			
CV	0.0024%				

As observed, a bilateral level of significance was found in the TPo factor, while the NG factor and the interaction between the NG x TPo factors did not show statistical significance. It can also be noted that there is a coefficient of variation of 0.0024%, which is a very low value, indicating stability and low dispersion around the mean of the values.

Regarding the evaluation of group pairs, the Duncan test was conducted to further examine the differences between groups within the TPo factor level. This is illustrated in table 9:

Table 9: Duncan's Test

Contrast	A	B	dof	alternative	p-unc	BF10
Tpo	1000	2500	22	two-sided	5.45E-10	1.087e+07
Tpo	1000	5000	22	two-sided	5.45E-10	1.087e+07
Tpo	2500	5000	22	two-sided		nan

According to the findings of the test, there is a significant difference between the levels 1000 and 2500, while no significant difference was found between the levels 2500 and 5000. Therefore, the instance of the genetic algorithm is set with an NG value of 50 and a TPo of 2500. The behavior regarding the stability of the algorithm can be observed in the figure 3.

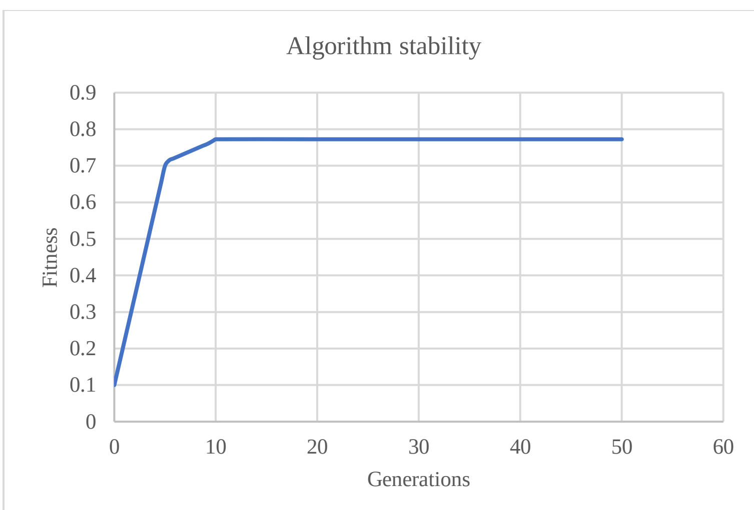


Figure 3: Stability of the Algorithm in Relation to its Fitness

In Figure 3, it can be observed that the fitness converges starting from the tenth generation. Considering that these variables oppose each other, with profitability being the element to maximize and volatility being the element to minimize, the Pareto frontier was generated to visualize the behavior of these variables across generations. This is shown in Figure 4.

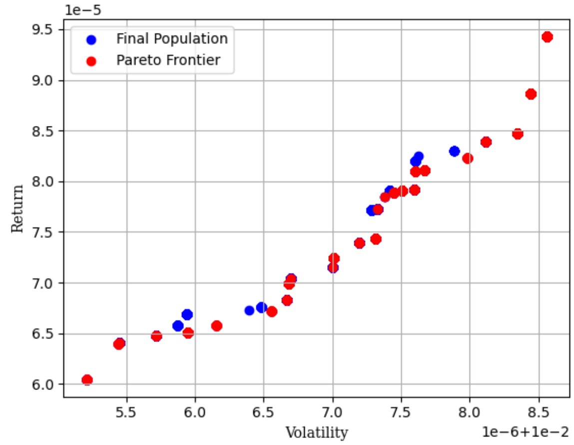


Figure 4: Pareto Frontier: Performance vs. Volatility

As (Markowitz, 1959) mentions, these two elements are opposed, and assuming higher profitability entails assuming higher risk or volatility. Therefore, the Pareto frontier previously shown evidences this behavior, which aligns with the nature of these variables, and no anomaly is observed in their linear relationship.

Finally, knowing the stability of the algorithm, we can represent the individual that generates optimality at this point as shown in the following table 10.

Table 10: Results of Percentages to Invest in Each Asset Belonging to the Portfolio

Percentage to invest	**Companies**
0.2948%	CASAGRC1
6.1901%	BUENAVC1
2.5055%	BBVAC1
2.2108%	ATCU
7.0744%	FALABEC1
4.1268%	ELCOMEI1
1.0317%	CVERDEC1
4.6426%	CREDITC1
0.1474%	CPACASC1
6.4849%	CORAREC1
4.2741%	LUISAI1
5.0847%	INVCENC1
6.9270%	INTERBC1
2.8740%	IFS
4.4952%	SCOTIAC1
2.2845%	PML
1.7686%	MINSURI1
3.2424%	MIRL
2.5792%	NEXAPEC1
5.1584%	PERUBAI1
6.6323%	POSITIC1
6.1164%	TELEFBC1
7.0007%	TEF
6.8534%	SAGAC1

As observed, the companies with the highest percentage to invest were represented by FALABEC1 and TEF, with approximately 7% each, while the company with the lowest percentage to invest was CPACASC1, with 0.1474%.

4 Discussion

For the discussion of the present research with various studies, it should be considered that each author within the background and the present research has developed their study in a particular environment, meaning that the results obtained regarding profitability or volatility depend specifically on the chosen and evaluated capital assets by the authors.

(Maholi et al., 2019), similarly to this research, employ a methodology where they first use an artificial neural network. In the author's case, they use a dense neural network architecture in the phase of selecting the most suitable capital assets to form the portfolios generated by the genetic algorithm, which will have as fitness function the return over risk, termed ERB, analogous to the Sharpe ratio, measuring excess return over a risk-free rate. The main criticism regarding this methodology is that dense neural networks do not discern regarding time series; that is, the inputs are not differentiated based on time, so the data adapts more to their internal structure instead of providing a response regarding the time variable. The precision reaches 98% for a unistep model, which is lower than this research where the unistep model has a precision of 99% in an LSTM network, while the MAE obtained by the researcher is 5.60%, whereas for the present research, it was 5.07% in a multistep model. Regarding the portfolio return, it reached 1.42%; however, the portfolio's volatility is much higher, reaching 1.86%, significantly higher than the 0.00612188% found in the optimal portfolio of this research.

(Rodríguez et al., 2020) in their research use another type of assets, which are the currencies of various countries, to minimize risk due to their stability. These were coded in terms of their return using natural logarithms. For this case, genetic operators are used analogously, such as cxblend and Gaussian mutation. However, the objective function is related to the conventional one used in many sources, which is associated with return for each unit of assumed risk. For this research, the weighting results differ. For quantified results, a return of 0.049292% was obtained, much lower than the proposed research, while the risk or volatility was 0.0354%, much higher than the volatility found in the present research.

(Bo Liu, 2023) in his research generates a network approach to optimize the investment portfolio, where the network entropy is used as a measure to determine the relevance degree of each action that makes up the portfolio to be optimized by a genetic algorithm. This is analogous to the preprocessing of this research, where a neural network is used to select the actions with the best performance in their next values to pass to the optimization of the genetic algorithm. As mentioned, the main metric of the research is the network entropy representing the investment portfolio, whereas in the present research, a method more related to the systematic market behavior was used. The results found with this present research differ in terms of return, as the result was higher, since for (Bo Liu, 2023), only a return of 0.2667% was found. On the other hand, (Chun-Hao et al., 2019) in their research, in contrast to this research, consider multiple aspects in the objective function, where they take into account that the portfolio is balanced and satisfaction related to the portfolio, as well as investment metrics, finally having a fitness of 4 elements in the numerator and one element in the denominator. For this research, another differing aspect is the number of elements comprising the portfolio, where only 5 capital assets were evaluated, mainly based on return on investment, resulting in a maximum return of 0.6%, lower than the return obtained in the results found in this research.

Vasiani, (2020) in their research considers a priority index for each business sector to be evaluated, so a preprocessing is considered that leads to the assets with the highest score for each sector, and then passes to the optimization of the investment portfolio. This process differs from that of the present research, considering that it does not take time series but rather the internal behavior values of the company regarding its transactions, such as buying, selling assets, or their devaluation. Another notable aspect is the objective function, where the return per unit of risk was used; for this case, it had an average of 3% higher than that of this research. However, there is no reference to the investment portfolio regarding the volatility of the assets, so this return could be related to a very high investment risk.

5 Conclusions

An artificial neural network was developed to predict trends in financial assets, configured with 150 processing units, trained for 400 epochs, and fine-tuned with a learning rate of 0.0004. The results demonstrated a significant accuracy of 92.35%, positioning it as a reliable tool for prediction in this context. Furthermore, error metrics were evaluated, such as the mean squared error with a value of 6.33% and a mean absolute error with a value of 5.07%, indicating a good generalization capacity of the model.

Regarding the genetic algorithm, the objective function was designed considering multiple key aspects such as profitability, volatility, Sharpe ratio, market risk premium, and portfolio diversification. This modeling is crucial for understanding and optimizing asset allocation within the investment portfolio, maximizing expected returns while simultaneously minimizing associated risk.

The chromosomal representation of individuals in the genetic algorithm was based on a vector of capital percentages, allowing effective manipulation of asset allocation within the portfolio. Additionally, algorithm hyperparameters such as initial population size, number of generations, crossover probability, and mutation probability were adjusted to achieve optimal algorithm performance.

After numerous iterations and evaluations, it was determined that the best configuration of the genetic algorithm was one comprising 50 generations and an initial population size of 2500 individuals. This configuration achieved a fitness of 0.772482, a return of 1.00058%, and a volatility of 0.00612%, suggesting a well-balanced portfolio with high returns and low risk.

The methodology developed in this research has demonstrated favorable results, thus it can be applied in any Stock Exchange, provided there are equity assets available.

References

[1] Asadov, B. (2018). The Current State of Artificial Intelligence (AI) and Implications for Computer Technologies. *International Journal of Communication and Computer Technologies (IJCCTS)*, *6*(2), 15-18.

[2] Bennett, D. A. (2001). How can I deal with missing data in my study? *Australian and New Zealand journal of public health*, *25*(5), 464-469. https://doi.org/https://doi.org/10.1111/j.1467-842X.2001.tb00294.x

[3] Cáceres Chian, V. A. E. (2018). Predicción de precios de acciones de bolsa de valores utilizando Support Vector Regression.

[4] Candia-García, C. D., López-Castro, L. F., & Jaimes-Suárez, S. A. (2020). Selección óptima del portafolio de proyectos utilizando metaheurísticas de población y trayectoria meta-optimizadas. *Revista EIA*, *17*(34), 271-288. https://doi.org/10.24050/reia.v17i34.1399

[5] Coaquira Velásquez, M. A., Tudela Mamani, J. W., & Jiménez Carrión, M. (2023). Regional comparative evaluation: Synthetic regional development index (RDI) for Peru. *Desarrollo y Sociedad*, (94), 109-157. https://doi.org/10.13043/DYS.94.4.
[6] Dagnino, J. (2014). Datos faltantes (missing values). *Rev Chil Anest*, *43*, 332-4. https://revistachilenadeanestesia.cl/datos-faltantes-missing-values/
[7] Das, A., Chaudhuri, T., Roy, S. S., Biswas, S., & Guha, B. (2023). Selection of appropriate portfolio optimization strategy. *Theoretical and Applied Computational Intelligence*, *1*(1), 58-81. https://doi.org/10.31181/taci1120237
[8] Didier, T., Levine, R., Montanes, R. L., & Schmukler, S. L. (2021). Capital market financing and firm growth. *Journal of International Money and Finance*, *118*, 102459. https://doi.org/10.1016/j.jimonfin.2021.102459
[9] Fang, C., & Wang, C. (2020). Time series data imputation: A survey on deep learning approaches. http://arxiv.org/abs/2011.11347
[10] Fuw, Y. Y., Zhen, W. L., & Su, H. C. (2011). Mobile Banking Payment System. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, *2*(3), 85-95.
[11] Gitman, L. J., & Joehnk, M. D. (2009). Fundamentos de inversiones. Pearson Educación.
[12] Hart, O. D., & Zingales, L. (2011). *Inefficient provision of liquidity* (No. w17299). National Bureau of Economic Research. http://www.nber.org/papers/w17299
[13] Hewamalage, H., Bergmeir, C., & Bandara, K. (2021). Recurrent neural networks for time series forecasting: Current status and future directions. *International Journal of Forecasting*, *37*(1), 388-427. https://doi.org/10.1016/j.ijforecast.2020.06.008
[14] Li, C., Shen, L., & Qian, G. (2023). Online Hybrid Neural Network for Stock Price Prediction: A Case Study of High-Frequency Stock Trading in the Chinese Market. *Econometrics*, *11*(2), 13. https://doi.org/10.3390/econometrics11020013
[15] Li, Q., Kamaruddin, N., & Al-Jaifi, H. A. (2023). Can Long-short Term Memory Neural Network with Symbolic Genetic Algorithm Predict Stock Price Change Basing on Fundamental Indicators. https://doi.org/10.21203/rs.3.rs-3306087/v1
[16] Liagkouras, K. (2019). A new three-dimensional encoding multiobjective evolutionary algorithm with application to the portfolio optimization problem. *Knowledge-Based Systems*, *163*, 186-203. https://doi.org/10.1016/j.knosys.2018.08.025
[17] Lim, S., Kim, M. J., & Ahn, C. W. (2020). A genetic algorithm (GA) approach to the portfolio design based on market movements and asset valuations. *IEEE Access*, *8*, 140234-140249. https://doi.org/10.1109/ACCESS.2020.3013097
[18] Liu, B. (2023). Portfolio Optimization Based on Complex Networks and Genetic Algorithms. *Academic Journal of Computing & Information Science*, *6*(10), 85-92. https://doi.org/10.25236/ajcis.2023.061013
[19] López, E. R., Mejía, J. F. G., & Garduño, Y. M. (2020). Diseño de un portafolio de inversión con algoritmos evolutivos. *Tecnología, Diseño e Innovación*, *6*(1), 51-70.
[20] Lu, R., & Lu, M. (2021). Stock trend prediction algorithm based on deep recurrent neural network. *Wireless Communications and Mobile Computing*, *2021*, 1-10. https://doi.org/10.1155/2021/5694975
[21] Markowitz, H. (1952). Portfolio Selection. *Journal of Economic Literature*, *7*(1), 77–91. http://links.jstor.org/sici?sici=0022-1082%28195203%297%3A1%3C77%3APS%3E2.0.CO%3B2-1
[22] Mumtaj Begum, H. (2022). Scientometric Analysis of the Research Paper Output on Artificial Intelligence: A Study. *Indian Journal of Information Sources and Services*, *12*(1), 52–58.
[23] Oleksandr, K., Viktoriya, G., Nataliia, A., Liliya, F., Oleh, O., Maksym, M. (2024). Enhancing Economic Security through Digital Transformation in Investment Processes: Theoretical Perspectives and Methodological Approaches Integrating Environmental Sustainability. *Natural and Engineering Sciences*, *9*(1), 26-45.

[24] Podvalny, S., Zolotukhina, I., Sotnikova, O., & Prokshits, E. (2021). Enhancing the Investment Attractiveness and Effectiveness of the Special Economic Zone. *Archives for Technical Sciences*, *1*(24), 1–8.

[25] Praveenchandar, J., Venkatesh, K., Mohanraj, B., Prasad, M., Udayakumar, R. (2024). Prediction of Air Pollution Utilizing an Adaptive Network Fuzzy Inference System with the Aid of Genetic Algorithm. *Natural and Engineering Sciences, 9*(1), 46-56.

[26] Reina, D. G., Córdoba, A. T., & del Nozal, Á. R. (2020). *Algoritmos Genéticos con Python: Un enfoque practico para resolver problemas de ingeniería*. Marcombo.

[27] Santillán, J. C. A. (2012). Alternativas de diversificación internacional para portafolios de acciones de la Bolsa de Valores de Lima. *Contabilidad y Negocios, 7*(13), 13-32. https://www.redalyc.org/articulo.oa?id=281623577003

[28] Siami-Namini, S., Tavakoli, N., & Namin, A. S. (2019). The performance of LSTM and BiLSTM in forecasting time series. *In IEEE International conference on big data (Big Data),* 3285-3292. https://doi.org/978-1-7281-0858-2/19/$31.00

[29] Solin, M. M., Alamsyah, A., Rikumahu, B., & Saputra, M. A. A. (2019). Forecasting portfolio optimization using artificial neural network and genetic algorithm. *In IEEE 7th International Conference on Information and Communication Technology (ICoICT),* 1-7. https://doi.org/10.1109/ICoICT.2019.8835344

[30] Sravana, J., Indrani, K. S., Saranya, M., Kiran, P. S., Reshma, C., & Vijay, V. (2022). Realısatıon of Performance Optımısed 32-Bıt Vedıc Multıplıer. *Journal of VLSI Circuits and Systems, 4*(2), 14-21.

[31] Srinadi, N. L. P., Hermawan, D., & Jaya, A. A. N. A. (2023). Advancement of banking and financial services employing artificial intelligence and the internet of things. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications, 14*(1), 106-117.

[32] Srinivasareddy, S., Narayana, Y. V., & Krishna, D. (2021). Sector beam synthesis in linear antenna arrays using social group optimization algorithm. *National Journal of Antennas and Propagation (NJAP), 3*(2), 6-9.

[33] Trivedi, J., Devi, M. S., & Solanki, B. (2023). Step Towards Intelligent Transportation System with Vehicle Classification and Recognition Using Speeded-up Robust Features. *Archives for Technical Sciences, 1*(28), 39-56.

[34] Tunga, S. K. (2021). Lotka's Law and Author Productivity in the Economic Literature: A Citation Study. *Indian Journal of Information Sources and Services, 11*(2), 1–8.

[35] Udayakumar, R., Joshi, A., Boomiga, S. S., & Sugumar, R. (2023). Deep Fraud Net: A Deep Learning Approach for Cyber Security and Financial Fraud Detection and Classification. *Journal of Internet Services and Information Security*, *13*(3), 138-157.

[36] Vasiani, V. D., Handari, B. D., & Hertono, G. F. (2020). Stock portfolio optimization using priority index and genetic algorithm. *In Journal of physics: conference series*, 1442(1), 012031. IOP Publishing. https://doi.org/10.1088/1742-6596/1442/1/012031

[37] Veerasamy, K., & Fredrik, E. T. (2023). Intelligent Farming based on Uncertainty Expert System with Butterfly Optimization Algorithm for Crop Recommendation. *Journal of Internet Services and Information Security, 13*(3), 158-169.

Authors Biography

Gustavo Alexis Flores-Fernandez is an industrial engineer graduate from the Universidad Nacional de Piura and holds a master's degree in economics from the Universidad San Antonio Abad del Cusco. He has served as a methodological advisor for academic research projects. His areas of interest include algorithmic optimization models, the use of agile methodologies for process optimization, and technological innovation projects.

Miguel Jimenez-Carrion, Doctor in Industrial Engineering from the National University of Piura, Peru, 2013. He was the winner of a mentoring contest to advise a participant with her thesis in Economic Engineering at the National University of the Altiplano of Peru. He is currently a Full Professor of the Operations Research Department of the Faculty of Industrial Engineering, and a RENACYT researcher; External Evaluator for CONCYTEC and Innovate Perú. His research interests include artificial neural networks, genetic algorithms, fuzzy logic, fuzzy controllers, shape recognition, mathematical modeling, and optimization.

Flabio Gutierrez, obtained my Bachelor's degree in Mathematics in 1991 at the National University of Trujillo (Peru), a Master's degree in Computer Science in 1997 at the University of Cantabria (Spain), I did doctoral studies in computer science at the Polytechnic University of Valencia (Spain) in 2013. In 2017 I obtained a PhD in Mathematics from the National University of Trujillo (Peru). I am currently a professor in the Department of Mathematics at the National University of Piura (Peru). My research area is artificial intelligence.

Raul A. Sanchez-Ancajima, Obtained the degree of Licenciado in Mathematics from the Faculty of Sciences of the National University of Piura, Peru (2004). He obtained a Master's degree in science with a major in Applied Mathematics from the National University of Piura, Peru (2011). Doctor in Mathematics from the National University of Trujillo, Peru (2021), with an internship at USP in Brazil. Senior Lecturer at the National University of Tumbes, Peru. My research interests are focused on Artificial Intelligence, Optimization Methods and Fractional Calculus.