Maximizing Returns on B3 Stock Investments through Integer and Binary Programming Strategies

Raphael Boratto
Universidade Federal do ParanÃ¡
Av. Cel. Francisco H. dos Santos, 100 - Jardim das AmÃ©ricas, Curitiba - PR, 81530-000
raphaelboratto@ufpr.br

Alexandre C. Choueiri
Universidade Federal do ParanÃ¡
Av. Cel. Francisco H. dos Santos, 100 - Jardim das AmÃ©ricas, Curitiba - PR, 81530-000
alexandrechecoli@ufpr.br

ABSTRACT

Investment analysis in the stock market is essential for maximizing profits and minimizing risks. Thus, this article presents an integer programming model to optimize the financial returns of stocks traded on the Brazilian Stock Exchange (B3). The developed approach proposes an objective function focused on maximizing investments, based on the stock's appreciation over time, considering both extraordinary payments, such as dividends and interest on equity, as well as stock price appreciation. The model uses binary and integer variables to structure the mathematical formulation, adhering to constraints such as the maximum investment value, permitted combinations, and risk limits defined by the investor. Additionally, this article outlines the components necessary for its computational implementation, using an explanatory flowchart, alongside the Python programming language and the Gurobi library.

KEYWORDS. Operational Research, Integer Programming, Optimization, B3, Investments, Portfolio Selection

1. Introduction

Efficiency in financial asset management is crucial for achieving solid results and maximizing investments. This challenge is evident in financial investments, especially in the stock market, due to the numerous variables and possibilities compared to fixed income investments. Thus, maximizing returns on stock market investments requires a comprehensive and thorough study, aiming to optimize profits while minimizing risks. According to Fama e French [2015], the unpredictability and complexity of the variables affecting the stock market makes it unlikely that an analysis will guarantee a future outcome with exact precision. These adversities affecting the market range from company management to economic and political influences, as well as variables such as stock prices, volatility, liquidity, profitability of securities, dividends, market volume, among others (Nti et al. [2020]).

There are several ways for analyzing variables in stock markets, some of them include: i-) fundamental analysis, ii-) graphical analysis, iii-) quantum analysis, iv-) foreign exchange analysis and v-) and political analysis Nti et al. [2020]. Additionally, the application of mathematical methods represents an effective strategy for maximizing returns on stock investments, with a particular focus on the Markowitz model, as published by Markowitz [1991] and later refined by Rubinstein [2002]. This model is essential for optimizing resource allocation in an investment portfolio, considering expected returns and associated risks. The problem dealt through statistical models that seek an efficient frontier of investment, using a variance-covariance matrix, and, in some cases, through linear, integer, or quadratic programming models Rahmani et al. [2019].

In this context, operational research methods are highly valuable in the field of financial markets, as well as in various industry sectors. Consequently, the analysis of the best investment opportunities among the numerous listed companies, covering different sectors and services, depends on the strategy, profile, and method applied, as noted by Dantzig [2016]. Operational research offers a mathematical method with analytical and robust analysis to model all actions on a large scale and optimize stock operations more efficiently, considering an objective function and constraints, as detailed by Chen et al. [2021].

In this context, the Brazilian Stock Exchange (B3) plays a fundamental role in the financial landscape, listing 403 companies from various sectors and activities. Analyzing the best investment opportunities among the numerous listed companies depends on the strategy, profile, and method applied Nti et al. [2020].

In this paper, we propose a multi-period IP model that optimizes investment option in stock market, considering the dividend and profits of each stock, setting a minimum expected return value, and a maximum expected risk value. The model is implemented using Python and Gurobi, and an API is used to automatically collect stock inputs using an API.

The paper is structured as follows: Section 2 is dedicated to the Literature Review, addressing the main models and techniques used in portfolio optimization, with a focus on Modern Portfolio Theory. Section 3 details the mathematical programming model, as well as the workflow connecting all components of the software (API and Solver) In section 4 we present the computational results of the model using real data from the Brazilian stock market. Finally, in Section 5 we discuss some final remarks.

2. Literature Review

Markowitz [1991] presents an approach to financial analysis, particularly by introducing the concept of efficient portfolio diversification. His studies provide a fundamental approach to understanding the best way to allocate financial assets, emphasizing diversification to minimize risk and maximize returns. The model aims to diversify the investment portfolio, focusing on a resolution based on financial theory and mathematics, exploring the monetary returns of each asset and their correlation. The risk is modeled trough through the variance-covariance matrix. Markowitz [1991] introduced the method of Modern Portfolio Theory, graphically presenting the efficient frontier, which demonstrates all possible combinations of risk and return for a set of assets. The model is a milestone in portfolio diversification and continues to be widely refined and integrated with other areas, such as optimization.

Siervo [2017] proposes an analysis to determine the best diversification in the investment portfolio, minimizing stock risks while considering a minimum expected return. His studies employed methods from Markowitz's Modern Portfolio Theory and linear programming. The authors use fictitious data from companies like PetrobrÃ¡s, EletrobrÃ¡s, Bradesco, Vale do Rio Doce and CEMIG, over 10 months period, in order to calculate the expected returns and covariances between the stocks. A mathematical model is then constructed, that expresses the minimization of the portfolio's variance, subject to various constraints, including the minimum expected return. The results were obtained using software like MATLAB and MS Excel, which indicated the optimal portfolio composition to minimize risk.

Ma et al. [2021] proposed an innovative portfolio optimization approach that integrates return forecasting using machine learning and deep learning techniques. The authors investigated the stock pre-selection process (before portfolio formation), evaluating the performance of Support Vector Machines and Random Forests, as well as 3 different neural networks forecasting models. Ma et al. [2021] analyzed a sample of data from the main stocks from "China Securities 100 Index", from 2007 to 2015, and applied Modern Portfolio Theory to define the percentage of each stock. By applying these techniques, they aimed to select high-quality stocks for portfolio construction. The results showed that combining return predictions with Modern Portfolio Theory yielded significantly better performances, compared to classical Markowitz models.

Chen et al. [2021] proposed a model that combines stock price prediction with machine learning and portfolio optimization based on the mean-variance model. The method uses XGBoost, a machine learning algorithm, to forecast stock prices. To optimize the hyperparameters of XGBoost, they employed an enhanced firefly algorithm, which dynamically divides the firefly group into elite and normal subgroups, incorporating chaotic and PSO-based search strategies. In the second stage, stocks with the highest potential return, as predicted, are selected, and the Markowitz model is then employed to determine the optimal investment allocation in each asset. Using historical data from the Shanghai Stock Exchange, experiments showed that the proposed method outperforms traditional methods.

Rahmani et al. [2019] conducted an analysis that applies meta-heuristic algorithms for portfolio optimization. In this study, they compared the results obtained through genetic algorithms (GA), ant colony optimization (ACO) and artificial bee colonies (ABC). Additionally, they calculated the sharpe, efficiency and risk indices for each portfolio. The ACO algorithm outperformed the GA and the ABC, proving itself effective in reducing risks and showing a more favorable risk-return relationship. The results concluded that research on the ACO is a suitable tool for investment managers in portfolio optimization.

3. Mathematical programming model and API

In this section we present the integer programming model the optimizes the portfolio of stocks, as well as the software architecture used to join the optimization engine with the data collection API.

3.1. Mathematical programming model

Considering the following notation:

Sets:

- $A = \{1, ..., n\}$: Set of different stocks.
- $T = \{1, ..., o\}$: Set of periods available to make the investment.
- $P = \{(j, k) \mid \forall j \in \{1, 2, \ldots, o\}, \forall k \in \{j, \ldots, o\}$: Where j is the start-period of investment and k is the end-period (the money is allocated at time j and removed at time k.

Parameters:

- L_{ijk}: Profit the stock i with investment start-start in j and end-time in k.
- D_{ijk}: Dividends of the in stock i with investment start-start in j and end-time in k.
- M_{ijk}: Sum of dividends and profits of the stock i with investment start-time in j and investment end-time in k, given by:

$$M_{ijk} = \left(1 + \frac{L_{ijk} + D_{ijk}}{100}\right)$$

- R_{ijk}: Risk in percentage associated with stock i, considering the investment starting at time j and ending at time k.
- n: Number of different stocks.
- V: Maximum amount invested at each period.
- E: Minimum expected return value.
- r: Maximum expected risk value in percentage.

Variables:

- x_{ijk}: Monetary investment amount in stock i, starting in period j and ending in period k, $x_{ijk} \in \mathbb{R}^+$
- $y_{ijk} = \begin{cases} 1 : \text{If} \quad x_{ijk} > 0 \\ 0 : \quad \text{otherwise} \end{cases}$

So the problem is then, the definition of how much money to be invested in which stocks, for how long. The time dimension is added to capture the premise that there exists compound interests, that is: if we invest in stock 1 only in time 1, and then again in time 2, is different than investing in time 1 and leaving the money there until time 2 (in the latter, compound interests would make it a more attractive investment).

3.2. Objective Function

The objective function is formulated as follows:

$$\max Z = \sum_{i=1}^{n} \sum_{j=1}^{o} \sum_{k \in T} x_{ijk} \cdot \mathrm{M}_{ijk} \tag{1}$$

The objective if pretty straightforward: maximize the amount invested at each stock i during the period (j, k), considering the sum of dividends and profits that this stock would generate for the period (M_{ijk}).

3.3. Constraints

The following constraints are imposed:

1. **Maximum Investment Limit Constraint:** This constraint aims for the variable x_{ijk} to be less than the constant V, which is the maximum investment value.

$$\sum_{i=1}^{n} \sum_{(j,k)\in P:(T\geq j, T\leq k)} x_{ijk} \leq V, \forall T = 1, ..., o \mid k \geq j \tag{2}$$

2. **Activation of y when x:** This constraint activates variable y_{ijk} whenever $x_{ijk} > 0$.

$$x_{ijk} \leq V \cdot y_{ijk}, \quad \forall i \in A, \forall j \in T, \forall k \in T \mid k \geq j \tag{3}$$

3. **Period investment constraint:** This constraint aims to prevent an investment in a stock from being repeated in the same periods. In other words, if an investment is made in a period, no further investment can be made in the same stock in the same period. As an example, consider Figure 3, where we have the y variables for 1 stock, considering a time horizon of 4 periods.

Figura 1: Possible investment combinations considering 4 periods

1	2	3	4
y_{111}			
y_{112}	y_{112}		
y_{113}	y_{113}	y_{113}	
y_{114}	y_{114}	y_{114}	y_{114}
	y_{122}		
	y_{123}	y_{123}	
	y_{124}	y_{124}	y_{124}
		y_{133}	
		y_{134}	y_{134}
			y_{144}

If we look at the variable matrix on the Figure it is easier to understand the constraint. Considering the first period alone: there are for variables that are affected by this period - $y_{111}, y_{112}, y_{113}, y_{114}$. It is clear that, if an investment is made on either one of them, none of the others could be also used. This constraint is then meant to avoid that situation. As an example, the constraints for the four periods are expressed as follows:

For the first period:

$$y_{111} + y_{112} + y_{113} + y_{114} \leq 1 \tag{4}$$

For the second period:

$$y_{112} + y_{113} + y_{114} + y_{122} + y_{123} + y_{124} \leq 1 \tag{5}$$

For the third period:

$$y_{113} + y_{114} + y_{123} + y_{124} + y_{133} + y_{134} \leq 1 \tag{6}$$

For the fourth period:

$$y_{114} + y_{124} + y_{134} + y_{144} \leq 1 \tag{7}$$

So the general constraint is:

$$\sum_{(j,k) \in P:(T \geq j, T \leq k)} y_{ijk} \leq 1, \quad \forall T = 1, ..., o \mid k \geq j \tag{8}$$

4. **Maximum Risk Constraint:** Finally, this constraint sets the maximum accepted risk.

$$\sum_{i=1}^{n} \sum_{(j,k) \in P:(T \geq j, T \leq k)} y_{ijk} \cdot \mathrm{R}_{ijk} \leq r \cdot 100, \quad \forall T = 1, ..., o \mid k \geq j \tag{9}$$

3.4. Software architecture

The resolution of the optimization model was developed using the Python programming language, with the application of the Gurobi library for solving mixed-integer programming (MIP) problems. Gurobi was chosen due to its efficiency and robustness in solving optimization problems involving continuous and binary variables. To obtain the financial data required for analysis, the Yahoo Finance API was integrated using the yfinance library (https://pypi.org/project/yfinance/), which allows the download of historical stock prices, dividends and other relevant data.

The flowchart presented in Figure 2 describes how the different parts of the software are connected, highlighting the steps from data collection, model creation, optimization and the generation of results. The flowchart is not solely based on the optimization model, but rather on the interactions between the system components, which include obtaining financial data for the analysis period, calculating dividends and stock appreciation, optimizing investment allocation Gurobi, and presenting the final solution.

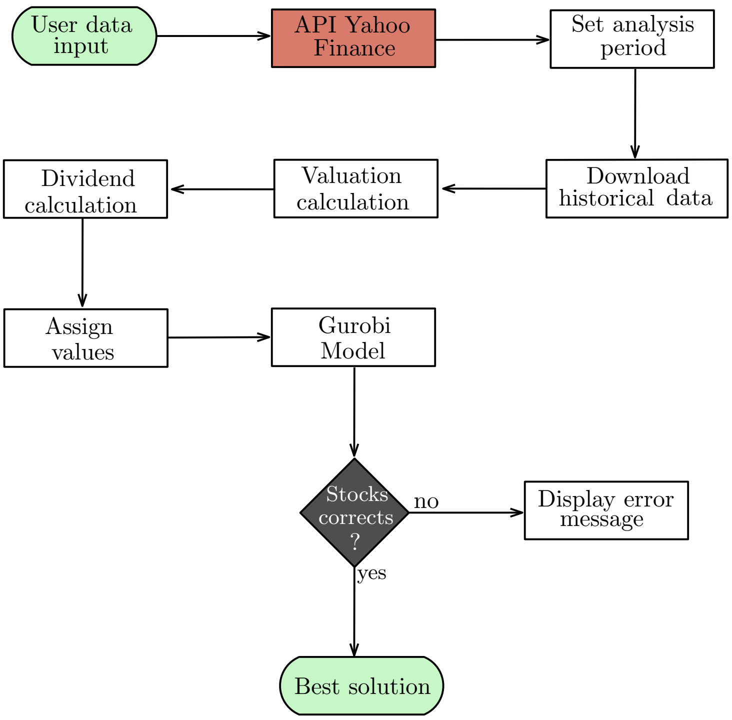
The first step is the user input, where information about the desired stocks for analysis is provided. Next, the Yahoo Finance API is called to obtain the necessary historical data, considering the defined analysis period. After obtaining the data, dividend and stock appreciation calculations are performed. These calculations are then used to assign values to the relevant variables through a dictionary. With data at hand, the Gurobi model is builded and optimized, taking into account the constraints and objectives defined by the user. If the stocks are incorrect, an error message is displayed. Otherwise, the system presents the best solution found, i.e., the optimal investment allocation according to the defined parameters.

4. Results

This section presents the collected data and the results obtained from the implementation of the optimization model, developed using the Python programming language and the Gurobi library. For model validation, tests were defined based on an analysis of four periods, corresponding to the four quarters of the year 2023.

The model was applied in five distinct instances, using different numbers of stocks (3, 10, 28, 40, and 66) to analyze the impact of the number of assets on the optimization results, both in terms of profitability and computational time, as demonstrated in Table 1. The choice of

Figura 2: Software architecture



different numbers of stocks aimed to assess the effect of diversification on the investment portfolio and the model's ability to adapt to varied allocation scenarios. The model also allows the user to define the maximum investment amount, represented as V in the mathematical model, without compromising computational performance. Additionally, the same risk level was considered for the stocks analyzed in all instances.

The simulation identified the best investment for the period under analysis, enabling not only a detailed evaluation of the stock behavior but also the time required to process the solutions, with the goal of analyzing the model's computational efficiency. The computational results of the five instances, corresponding to the four quarters of 2023, are described in Table 2. Each row corresponds to a different instance, showing the results of the stock allocation for each specific period. The 'Number of Stocks' column indicates how many stocks were analyzed in each instance, while the 'Time (s)' column shows the computational execution time of the code in seconds. The 'Result Q1', 'Result Q2', 'Result Q3', and 'Result Q4' columns display the selected stocks for each of the four periods, reflecting the optimal solution determined by the model. If a cell contains 'NaN', it means that no stock was selected for that period. The results obtained in Table 2 demonstrated the optimization model's ability to provide robust solutions to maximize returns while simultaneously respecting the defined constraints.

5. Conclusion

The optimization problem of stock investments is an area that has been studied for a long time in the financial market, with different approaches highlighting the growth related to computational optimization. With the advancement of computational methods and tools, the search for efficient solutions to maximize investment returns, respecting constraints such as risk and capital

Instance	Stocks
1	BBDC4, ITUB4, SANB4
2	BBDC4, ITUB4, SANB4, PETR3, VALE3, WEGE3, MGLU3 LREN3, ABEV3, CIEL3
3	BBDC4, ITUB4, SANB4, PETR3, VALE3, WEGE3, MGLU3 LREN3, ABEV3, CIEL3, UGPA3, KLBN11, BBAS3, RENT3 EMBR3, GGBR4, BRFS3, CSNA3, HYPE3, PSSA3, RADL3 B3SA3, TAEE11, IRBR3, QUAL3, SMAL11, TUPY3, CYRE3
4	BBDC4, ITUB4, SANB4, PETR3, VALE3, WEGE3, MGLU3 LREN3, ABEV3, CIEL3, UGPA3, KLBN11, BBAS3, RENT3 EMBR3, GGBR4, BRFS3, CSNA3, HYPE3, PSSA3, RADL3 B3SA3, TAEE11, IRBR3, QUAL3, SMAL11, TUPY3, CYRE3 CIEL3, TASA4, CRFB3, MRVE3, NGRD3, BEEF3, ELET3 FESA4, RENT3, VIVT3, OIBR3, TRPL4
5	BBDC4, ITUB4, SANB4, PETR3, VALE3, WEGE3, MGLU3 LREN3, ABEV3, CIEL3, UGPA3, KLBN11, BBAS3, RENT3 EMBR3, GGBR4, BRFS3, CSNA3, HYPE3, PSSA3, RADL3 B3SA3, TAEE11, IRBR3, QUAL3, SMAL11, TUPY3, CYRE3 CIEL3, TASA4, CRFB3, MRVE3, NGRD3, BEEF3, ELET3 FESA4, RENT3, VIVT3, OIBR3, TRPL4, AGRO3, MDIA3 LIGT3, BRAP4, EGIE3, CVCB3, SBSP3, NTCO3, CASH3 AMER3, CEAB3, CCRO3, AERI3, ASAI3, B3SA3, GGBR4 CSMG3, PLAS3, REDE3, POMO4, USIM5, EQTL3, LOGG3 BRPR3, CPLE6, VIVT3, BRKM3, SBFG3, NUTR3

Tabela 1: Description of all actions analyzed in each instance

Instance	Number of Actions	Time (s)	Result Q1	Result Q2	Result Q3	Result Q4
1	3	7.1	BBDC4	BBDC4	NaN	ITUB4
2	10	10.5	PETR3	ITUB4	ABEV3	LREN3
3	28	23.4	MGLU3	RADL3	QUAL3	TUPY3
4	40	37.3	GGBR4	RADL3	CSNA3	IRBR3
5	64	70.1	AMER3	LOGG3	MDIA3	TUPY3

Tabela 2: Computational results of each instance

limits, has intensified, especially through the use of exact and heuristic algorithms. The application of the optimization model in this study was performed using Gurobi, one of the most advanced libraries for solving operations research problems.

This work uses the integer programming model with computational application due to its quantitative efficiency in maximizing stock investment returns. The approach aims to identify the best solution for the defined parameters and analysis periods, presenting the results of different stock combinations over the four quarters of 2023. Thus, simulations were carried out with instances of 3, 10, 28, 40, and 66 stocks, resulting in the best feasible solution and the computational time necessary for execution.

The computational simulations of the described model presented effective results in identifying the best stock to invest in each period, respecting constraints on invested capital, combinations, and risk levels. The analysis of computational efficiency revealed that processing time grew proportionally as a larger number of stocks was analyzed. This increase in time is within the acceptable execution limit, given the model's complexity. The results suggest that the proposed model can be a valuable tool for investors seeking more efficient asset allocation strategies and quantitative stock analyses based on their profiles.

However, it is important to emphasize that this optimization model requires additional considerations in its application. The definition of parameters must be updated and aligned with market realities, especially the percentage of risk level and historical stock data, to ensure the best solution. Although the model has proven efficient in results and processing time, its application in real-world conditions depends on the reliability of the data used.

Finally, this article aims to assist investors in their stock market investment analyses, contributing to academic development in the application of optimization models focused on the financial market. In this way, the model allows investors to define parameters independently, using tools such as Gurobi and Python to optimize investment allocation.

6. Appendix

Example of period investment constraint

For example, suppose we are analyzing a single stock ($T = 1$) over four periods ($o = 4$). The central idea is to establish all possible combinations of periods in which an investment can start and end, adhering to the rule that the ending period (k) must be greater than or equal to the starting period (j).

For this, we consider the following sets:

- **T**: Represents the number of stocks under analysis. In this example, we have $T = 1$, meaning we are analyzing a single stock.
- **o**: Represents the number of periods. In this case, we have $o = 4$, meaning we are considering four time periods.

Now, we will detail the valid combinations for j (investment start) and k (investment end) for each value of R (the restriction associated with the start and end of the investment):

For $\boldsymbol{o = 4 \Rightarrow T = \{1, 2, 3, 4\} \Rightarrow j = \{1, 2, 3, 4\}}$

The valid combinations for each value of T are described below:

$\boldsymbol{T = 1}$**:**

$$j = 1 \quad \text{and} \quad k \in \{1, 2, 3, 4\}$$

Explanation: $1 \geq j$, so $j = 1$, and $1 \leq k$, so $k \in \{1, 2, 3, 4\}$

The valid combinations are:

$$(1, 1), (1, 2), (1, 3), (1, 4)$$

$T = 2$:

$$j \in \{1, 2\} \quad \text{and} \quad k \in \{2, 3, 4\}$$

Explanation: $2 \geq j$, so $j \in \{1, 2\}$, and $2 \leq k$, so $k \in \{2, 3, 4\}$
The valid combinations are:

$$(1, 2), (1, 3), (1, 4), (2, 2), (2, 3), (2, 4)$$

$T = 3$:

$$j \in \{1, 2, 3\} \quad \text{and} \quad k \in \{3, 4\}$$

Explanation: $3 \geq j$, so $j \in \{1, 2, 3\}$, and $3 \leq k$, so $k \in \{3, 4\}$
The valid combinations are:

$$(1, 3), (1, 4), (2, 3), (2, 4), (3, 3), (3, 4)$$

$T = 4$:

$$j \in \{1, 2, 3, 4\} \quad \text{and} \quad k = 4$$

Explanation: $4 \geq j$, so $j \in \{1, 2, 3, 4\}$, and $4 \leq k$, so $k = 4$
The valid combinations are:

$$(1, 4), (2, 4), (3, 4), (4, 4)$$

Referências

Chen, W., Zhang, H., Mehlawat, M. K., e Jia, L. (2021). Mean–variance portfolio optimization using machine learning-based stock price prediction. *Applied Soft Computing*, 100:106943.

Dantzig, G. B. (2016). Linear programming and extensions. In *Linear programming and extensions*. Princeton university press.

Fama, E. F. e French, K. R. (2015). A five-factor asset pricing model. *Journal of financial economics*, 116(1):1–22.

Ma, Y., Han, R., e Wang, W. (2021). Portfolio optimization with return prediction using deep learning and machine learning. *Expert Systems with Applications*, 165:113973.

Markowitz, H. M. (1991). Foundations of portfolio theory. *The journal of finance*, 46(2):469–477.

Nti, I. K., Adekoya, A. F., e Weyori, B. A. (2020). A systematic review of fundamental and technical analysis of stock market predictions. *Artificial Intelligence Review*, 53(4):3007–3057.

Rahmani, M., Khalili Eraqi, M., e Nikoomaram, H. (2019). Portfolio optimization by means of meta heuristic algorithms. *Advances in mathematical finance and applications*, 4(4):83–97.

Rubinstein, M. (2002). Markowitz's"portfolio selection": A fifty-year retrospective. *The Journal of finance*, 57(3):1041–1045.

Siervo, J. S. D. (2017). Aplicação de programação linear na seleção de carteiras de investimento.