# Application of Artificial Intelligence-Genetic Algorithms to Select Stock Portfolios in the Asian Markets

Luu Thu Quang
Ho Chi Minh University of Banking, Vietnam

***Abstract*—The paper's main goal is to use a genetic algorithm to find the best stock portfolio that meets the criteria of high return and low risk, allowing investors to adjust the appropriate proportion for each share. Using the Python programming language based on the Jupyter Notebook engine, this paper introduces a model of six stock portfolios, each of 30 stocks selected with market capitalization and high liquidity criteria of six markets in the Asian region. The results show that the four portfolios created from the markets of Vietnam, Thailand, Philippines, and Singapore meet both the return and risk objectives. The Malaysian market only meets the risk target, but the portfolio's return is not close to the expected ratio. Meanwhile, the Indonesian market outperformed expectations in terms of profits, but high profits come with high risks, so this market carries a concerning level of risk when compared to the profit and loss of other markets. The suggested stock allocation levels for each portfolio are based on the above results. Finally, the author proposes several policy implications related to the management and operation of the market to limit unnecessary price fluctuations of the stock and affect the business model of companies.**

***Keywords—Artificial intelligence; genetic algorithms; optimal portfolio; Sharpe ratio; Asian stock market***

## I. Introduction

When learning about investment, optimizing the return on a portfolio is one of the most challenging issues in the financial sector. Since it is a process that is not simply a combination of individual stock codes with unexamined investment ratios, investors need to find ways to maximize the profitable investment and limit risk by understanding the market and setting themselves a desired level of return with an acceptable risk level. Dealing with this problem, recently, several artificial intelligence-based methods such as genetic algorithms have been applied to overcome this problem. Genetic algorithms (GAs) are optimal methods. They are widely used in many different fields such as medicine, engineering, electronics, not only the field of financial investment analysis to optimize value and solve problems through the fundamental processes of selection, breeding, and mutation between individuals in the population to create better latter-generation individuals, improving the adaptability of that individual in a more natural environment.

There have been many studies applying genetic algorithms to investments. However, the scope of the above studies is in the overseas market, where the market characteristics are quite different from the market in Southeast Asia. In Southeast Asia, studies of this method of migration algorithms have been applied and studied in many other fields such as health, biology, engineering, construction, information technology to solve different problems. The application of this method to the financial sector is still relatively small. Therefore, implementing this research in the Asian market is necessary and feasible, especially in the growing and constantly changing stock market.

An optimal portfolio is an asset allocation, or in other words, a group of different stock choices that provide the highest rate of return per unit of risk an investor is willing to accept. In addition, the optimal portfolio is considered to be the one with the maximum Sharpe ratio - which measures the amount of return generated per unit of risk when investing in an asset or type of other investments. According to Markowitz's methodology, the linear equations are used to determine the best portfolio, but this traditional method works best when the number of stocks and the size of the search space are both small. When there are a lot of stocks and large searchable space, special techniques, such as GAs, must be used. Compared to other optimization methods, GAs work by simulating evolution and natural selection on a computer starting with a random population. Additionally, we require an adaptive function to choose good individuals and reject bad ones in order to optimize. Genetic algorithms, in contrast to other optimization techniques, have the following advantages:

- GAs work with the variable's code, not directly on the variable.
- Most of the optimization techniques normally search from a single vertex, while GAs always operate on a set of vertices (optimal points), this is an advantage of GAs that increases the chances of reaching the comprehensive optimal, and avoids premature convergence at the local point.
- GAs evaluate the objective function to serve the search process, so it can be applied to any optimization problem (continuous or discrete).
- GA is a subclass of probability algorithms, and their fundamental operations are based on their capacity to incorporate randomness into the processing flow.

The paper will help answer the question: "Should genetic math be applied to the creation of an effective stock portfolio for southeast Asian markets?" To answer that question, the author will study, learn and apply the processes of acquisition to a portfolio of asset classes with a wide range of asset classes with a level of assets risks and ratios that vary to select and create a potential stock portfolio with high profitability criteria with the right level of risk.

## II. Theoretical Basis

Yang [1] compares three approaches from genetic algorithms, medium variances, and Bayes methods to improve portfolio efficiency with a sample of MSCI (Morgan Stanley Capital International) datasets of Total Return on Equity of Canada, France, Germany, Japan, the United Kingdom, and the United States include European stocks, bonds, and bonds based on the value of the U.S. dollar between January 1975 and December 2004, taking the risk-free rate of return as the T-bill interest rate of three months. The problem is that the portfolio has a minor standard deviation, and out-of-sample returns also have a significantly higher average value than the standard medium variance portfolio strategy and the Bayes approach. Portfolio weighting using genetic algorithms is less unbalanced and fluctuates much less over time than the standard average method and Bayes method. Furthermore, the author has stated that the algorithm is a model that combines historical information and future uncertainty to significantly improve the accuracy of the average estimate, helping to improve the model's performance by overcoming computational difficulties when approached by the Bayes method

Lin [2] also reports that adopting genetic algorithms would provide optimal value for creating a portfolio. The author asserted that the genetic algorithm could solve the corresponding optimization problems because existing traditional methods cannot be effectively addressed. Moreover, the study based on a team of 40 sample companies collected from the Taiwan market was used to demonstrate the proposed method with the probability of hybridization and random mutations.

Sefiane [3] applied genetic algorithms to optimize a portfolio of five stocks with data including portfolio-wide average returns and portfolio variances for five years from 2007 to 2011. By running the data on Matlab software, the author has produced the result of five proportions of investment in five stock codes and selected a hybrid by one point, two points through the stage of three hybrids. At the end of the research process, the author commented that this problem achieves exceptionally effective results by shortening the calculation time to get an investment solution for the whole portfolio. Dubinskas [4] also evaluate the genetic algorithm method for data of 18 stock codes made from Lithuanian businesses listed on the official list of NASDAQ OMX Baltic companies in 2013 planned portfolio to be built including four businesses representing different sectors. Research indicates that optimizing the portfolio by adopting an algorithmic approach generates four times more returns than portfolios built using a defined or random programming method. Along with high returns, the systematic risk assessment of the portfolio analyzed by genetic algorithms is higher than the portfolio optimized by defined and random programming. However, the authors commented that the portfolio's performance optimized by algorithmic genetics might exceed market performance. Moreover, the author also concludes that a research paper needs to examine several cycles to look for more reliable conclusions, emphasizing the application of genetic algorithms methods during a market downturn.

In addition, Cheong et al. [5] also implement genetic algorithms to support portfolio optimization based on investor information, which is to build portfolios by selecting institutionalized stocks or foreign investors invest more than other stocks. Through data obtained from the Korea Securities Computer Corporation (KOSCOM), specifically the top 90 Kospi 200 companies in terms of market capitalization from September 1, 2007, to May 30, 2014, and through the analysis process, the author commented that long-term information outperforms medium- and short-term information on the Korean stock market. The weight varies when using genetic algorithms how it affects portfolio performance. In particular, the study results add to the behaviors and patterns of investors that can be used to devise investment strategies. Investors build portfolios according to their risk-return level on the average variance framework, and relying on the portfolio strategies of other confident investors can enhance portfolio performance. However, the author also notes that institutional investors anticipate higher returns and lower risks on the Korean stock market, while foreign investors only consider portfolio risk.

Liu et al. [6] shows that genetic algorithms produce results that provide a more reliable portfolio in the same risk and without the need to introduce or mention each country's data. Liu et al [6] argues about the time series length of each stock on the construction of the portfolio and improve the algorithmic efficiency in fundamental stock analysis. With a fairly comprehensive sample of data, including the closing price of 2317 stocks on the Chinese market from February 20, 2016, to February 16, 2017, there are 250 trading days. The author agrees with the researchers above that genetic algorithms have optimized portfolio strategies. Sharpe ratios are relatively high, markedly improved along with an increase in the number of iterations, meaning that the ability to achieve excess returns increases gradually [7]. Furthermore, the author also compared two other methods, the random method and the K-mean algorithm. Their results indicate that with the increase in risk, the rate of profitability when using genetic algorithms increases, indicating the effectiveness of genetic algorithms.

In general, after consulting and researching research articles, the application of genetic algorithms to selectively and creating an optimal portfolio is agreed and encouraged by many authors because the effectiveness of the algorithm is persuasive when going through the analysis process to create an expected profit level. The standard deviation and deviation are in line with the expected level. Therefore, the expansion direction of future studies is to be able to conduct on a broader scale, a long time to test the appropriate level of an optimal category with the criteria of limiting risk, achieving the expected or highest level of profitability.

## III. Research Method

### A. Data

The data used in this study includes trading histories such as stock codes, trading months and closing prices provided at the DataStream system. The subjects selected in the study data are stocks listed on the stock exchanges of Vietnam, Thailand, Indonesia, Philippines, Malaysia, and Singapore with the following characteristics:

- Shares listed at least two years before the time of inclusion in the portfolio
- The stock must be traded during the study sample selection period

The data includes 180 stock codes, of which in Vietnam there are 30 company codes in VN30 and 30 companies with the largest capitalization in 5 countries, the month of trading started oil from 07/2018 to 04/2021

### B. Method

Genetic mathematics is a technique of computer science that predicts the behavior and reactions of humans and organisms. Migration algorithms can be applied in the financial sector by forecasting profits, stock prices, portfolio optimization, and determining trading rules and pricing options. Genetic mathematics includes transformation modalities, including selection, hybridization, and mutation. Identifying individuals in the current generation retained in the next generation is called the selection process. The random combination of two different individuals, using the things available in the previous generation passed on to the next generation, is hybridization. Randomly changing the structure of the individual, and transforming into new individuals is called the process of mutation [8].

The author uses a programming language called Python, which runs on the Jupyter Notebook tool to build a research process, a computer-language method, also known as computer-language code used to replace a string of characters or statements briefly. These codes are technically valuable in terms of information technology. The change in this code does not affect the values and results involved, nor does it explain any economic significance. The code used in the article is inherited from many different authors whose research papers agree with this research paper. The code is only intended to support the function and give the study results. The author has also changed the code to match the author's research direction. The study focuses on variables in profitability, risk, and allocation levels accordingly and delivers the most optimal portfolio outcome. The research process will be as follows:

The prerequisite for implementing the algorithm, the libraries are inserted into python language to help support the calculation of significant data sources such as numpy, pandas, functools. Code:

```
import numpy as np
import pandas as pd
from functools import reduce
```

Step 1, import the data of a country of 30 companies, specifically code as follows. The author takes the example of how Vietnam runs. Code:

```
files=['VIETNAM.csv']
dfs=[]
for file in files:
temp=pd.read_csv(file)
dfs.append(temp)
stocks = reduce(lambda left,right: pd.merge(left,right,on='Date'), dfs)
print(stocks.shape)
stocks.head()
```

In the Step 2, the profit is measured through the historical transaction values. The closing price equal to the average of 1 to 33 months, is takes the timeline of 04/2021 as the standard of calculating profitability. The formula used is the profit margin stated by the author in sector 2, section 2.1.3.2, specifically the total return of the stock plus the dividend paid and divided by the original price of the stock. Here, we assume the percentage of dividends paid is not in companies. Code:

```
def hist_return(months):
 idx=[]
 df=pd. DataFrame()
 for mon in months:35
 temp=(stocks.iloc[0,1:] - stocks.iloc[mon,1:])/(stocks.iloc[mon,1:])
 idx.append(str(mon)+'_mon_return')
 df=pd.concat([df, temp.to_frame(). T], ignore_index=True)
 df.index=idx
 return df
hist_stock_returns=hist_return([3,6,12,24,33])
hist_stock_returns
```

In Step 3, identify the gene or, in other words, encode the problem. In this step, the author will use a binary sequence of n-gene lengths (with n being the number of securities included in the category) as chromosomes to represent a category with n securities to study. Each gene represents the appearance or absence of a stock code, which these genes are regulated in alphabetical order. Code:

```
gene = np.random.rand()
Gene import time
def gen_mc_grid(rows, cols, n, N):
 np.random.seed(seed=int(time.time()))
 layouts = np.zeros((n, rows * cols), dtype=np.int32)
 # layouts_cr = np.zeros((n*, 2), dtype=np.float32)
 positionX = np.random.randint(0, cols, size=(N* n* 2))
 positionY = np.random.randint(0, rows, size=(N*n*2))
 ind_rows = 0
 ind_pos = 0
 while ind_rows < n:
 layouts[ind_rows, positionX[ind_pos] + positionY[ind_pos]*cols] =1
 if np.sum(layouts[ind_rows, :]) == N:
 ind_rows += 1
 ind_pos += 1
 if ind_pos >= N*n*2:
 print("Not enough positions")
 Break return layouts
def gen_mc_grid_with_NA_loc(rows, cols, n, N,NA_loc):
 np.random.seed(seed=int(time.time()))
 layouts = np.zeros((n, rows * cols), dtype=np.int32)
 layouts_NA= np.zeros((n, rows * cols), dtype=np.int32)
 for i in NA_loc:
 layouts_NA[:,i-1]=2
 positionX = np.random.randint(0, cols, size=(N* n* 2))
 positionY = np.random.randint(0, rows, size=(N*n*2))
 ind_rows = 0
 ind_pos = 0
```

```
 N_count=0
while ind_rows < n:
cur_state=layouts_NA[ind_rows,positionX[ind_pos]+positionY[ind_pos] * cols]
if cur_state!=1 and cur_state!=2:
layouts[ind_rows, positionX[ind_pos] + positionY[ind_pos] * cols]=1
layouts_NA[ind_rows, positionX[ind_pos] + positionY[ind_pos] * cols] = 1
 N_count+=1
 if np.sum(layouts[ind_rows, :]) == N:
 ind_rows += 1
 N_count=0
 ind_pos += 1
 if ind_pos >= N*n*2:
 print("Not enough positions")
 Break return layouts,layouts_NA
gen_mc_grid(9, 9, 180, 90)
gen_mc_grid_with_NA_loc(9, 9, 180, 90.range(10))
```

Step 4, gather the number of shares in the portfolio and randomly assign the capital allocation ratio, small parts of the total capital specified for each share, to set the investment allocation weighting. A stock portfolio will include fractional or decimal values of all stocks so that the sum of the values is not greater than 100%. Simply put, our portfolio has 30 stocks; the author will randomly create 30 variables corresponding to the proportion of investment in 30 stocks with a total capital of 1 unit. Code:

```
def chromosome(n):
 ch = np.random.rand(n)
 return ch/sum(ch)
child=chromosome(30)
print(child,sum(child))
```

The Step 5 is the first step in the genetic algorithm to initiate populations. The process is quite simple by creating a population of chromosomes, in which each chromosome is a long binary vector n-gene randomly generated. We need to calculate the probability of the securities number that allow research in the portfolio. Having a portfolio with many securities will not be highly effective in terms of objectives. That probability is measured as follows:

$$P = n/(pop\_size)$$

in which n is the number of securities included in the resulting portfolio, pop_size is the number of securities of the whole population. At that time, the initialization process was roughly described as in each gene of chromosomes in the population that generated a random number of realities in the passage [0, 1]. When $r < p$, the stock code appears in the category, i.e., the value of that gene is equal to 1. On the contrary, the gene is zero that stock code will not appear in the portfolio. Code:

```
n=30
pop_size=180
population = np.array([chromosome(n) for _ in range(pop_size)])
print(population.shape)
print(population)
```

Step 6, the author will calculate the target function stated in sector 2 is the Sharpe ratio, taking this ratio as a measure of the performance of the entire portfolio, in the calculation formula, this ratio depends on the variable is a risk-free interest rate, the author will default this ratio to 2%.

$$eval(vi\ )=Sharpe\ ratio=\ (E(Rport\ )-rf)/\delta p$$

To be able to calculate the Sharpe ratio, calculate the following small components: Calculate the average profit, standard deviation and variance of past stock returns. Code:

```
# First of all, entering and re-universalizing the profit rate calculated in step 2
print(hist_stock_returns.info())
cols=hist_stock_returns.columns
hist_stock_returns[cols]=hist_stock_returns[cols].apply(pd.to_numeric,errors ='coerce')
print(hist_stock_returns.info())
# Calculate the variance of the historical profit, so that it can be easy to calculate, the author will put the variance of the stock itself by zero
cov_hist_return=hist_stock_returns.cov()
print(cov_hist_return)
for i in range(30):
 cov_hist_return.iloc[i][i]=0
cov_hist_return
# Calculating average historical returns
mean_hist_return=hist_stock_returns.mean()
mean_hist_return
# Calculate the standard deviation of historical profits
sd_hist_return=hist_stock_returns.std()
sd_hist_return
# Calculate the expected return rate of the category
def mean_portfolio_return(child):
 return np.sum(np.multiply(child,mean_hist_return))
mean_portfolio_return(population[0])
# Calculate the variance of the category
def var_portfolio_return(child):
 part_1 = np.sum(np.multiply(child,sd_hist_return)**2)
 temp_lst=[]
 for i in range(30):
 for j in range(30):
 temp=cov_hist_return.iloc[i][j] * child[i] * child[j] * child[j]
 temp_lst.append(temp)
 part_2=np.sum(temp_lst)
 return part_1+part_2
var_portfolio_return(population[0])
# Calculate the Sharpe ratio of the category
def fitness_fuction(child):
 return(mean_portfolio_return(child)-rf)/np.sqrt(var_portfolio_return(child))
fitness_fuction(population[31]
```

The step 7 is a pretty important step in the genetic algorithm. The author will filter the stocks with the highest Sharpe ratio calculated in Step 6. In this step the author defaulted to a selective probability of 0.3, and the author selected the first three stocks in the category to consider the target function of the research paper. Code:

```
def Select_elite_population(population, frac=0.3):
 population=sorted(population,key=lambda x:
fitness_fuction(x),reverse=True)
 percentage_elite_idx = int(np.floor(len(population)* frac))
 return population[:p ercentage_elite_idx]
print(wool(Select_elite_population(population, frac=0.3)))
Select_elite_population(population, frac=0.3)
[fitness_fuction(x) for x in population] [:3]
```

Step 8 is mutating the individuals in the category. It can be said that this is a function that improves the individual more and more healthily than the first life. Especially in this process, the author will randomly select two stocks in the list of 30 stocks to conduct the transformation, and the following computer code will perform the following mutations. Code:

```
def mutation(parent):
 child=parent.copy()
 n=np.random.choice(range(30),2)
 while (n[0]==n[1]):
 n=np.random.choice(range(30),2)
```

```
 child[n[0]],child[n[1]]=child[n[1]],child[n[0]]
 return child
mutation(population[1]),population[1]
```

Step 9, at this step the stock codes will be intersected through real-value investment proportions. In the hybrid task, the author uses the arithmetic hybrid method to bring the most optimal effect. Code:

```
def Arithmetic_crossover(parent1,parent2):
 alpha = np.random.rand()
 child1 = alpha * parent1 + (1-alpha) * parent2
 child2 = (1-alpha) * parent1 + alpha * parent2
 return child1,child2
Arithmetic_crossover(population[2],population[3])
```

Step 10 is the step that helps create a generation of stock codes that come with a more potential weighting perform mutation processes or cross-exchange based on probability. The author defaulted to the probability of mutation as 0.4 and hybridization as 0.6. In the final stages, the probability of mutation will be reduced to 0.1, and hybridization will increase to 0.9. The sum of these two probabilities is always equal to 1, and it should be noted that the probability of hybridization always needs to be greater than the probability of mutation because mutations need to be limited, not always mutations to individuals with good character and adaptability to the habitat or in other words to survive in the category. The input of this process will be the sum of the stocks with the appropriate proportion of investment allocation. The output will be the next generation with a more suitable gene level. Code:

```
def next_generation(pop_size,elite,crossover=Arithmetic_crossover):
 new_population=[]
 elite_range=range(wool(elite))
 while wool(new_population) < pop_size:
 if wool(new_population) > 2*pop_size/3:
 mutate_or_crossover = np.random.choice([0, 1], p=[0.9, 0.1])
 else:
 mutate_or_crossover = np.random.choice([0, 1], p=[0.4, 0.6])
 if mutate_or_crossover:
 indx=np.random.choice(elite_range)
 new_population.append(mutation(elite[indx]))
 else:
 p1_idx,p2_idx=np.random.choice(elite_range,2)
 c1,c2=crossover(elite[p1_idx],elite[p2_idx])
 chk=0
 for gene in range(30):
 if c1[gene]<0:
 chk+=1
 else:
 chk+=0
 if sum(range(chk),0)>0:
 p1_idx,p2_idx=np.random.choice(elite_range,2)
 c1,c2=crossover(elite[p1_idx],elite[p2_idx])
 new_population.extend([c1,c2])
 return new_population
elite=Select_elite_population(population)
next_generation(180,elite)[:3]
elite=Select_elite_population(population)
next_generation(180,elite,Arithmetic_crossover)[:3]
```

Step 11 is a repeat of the entire process until the stocks in the portfolio do not change in terms of maximum profit and minimum risk or the number of fixed iterations selected by the author is 40 iterations. The author ordered that the expected return of the category would be greater than 20% and that the risk through variance calculations must be less than 5%. The loop will stop at the 40th round if that condition is met. If it does not meet, the study will perform through many subsequent loops so that 43 searches for the appropriate proportion for the 43. Stock codes suitable for the portfolio's great purpose achieve maximum profitability with a reasonably limited level of risk. Code:

```
n=30
pop_size=180
population = np.array([chromosome(n) for _ in range(pop_size)])
elite = Select_elite_population(population)
iteration=0
Expected_returns=0
Expected_risk=1
while (Expected_returns < 0.2 and Expected_risk > 0.05) or iteration <= 40:
 print('Iteration:',iteration)
 population = next_generation(180,elite)
 elite = Select_elite_population(population)
 Expected_returns=mean_portfolio_return(elite[0])
 Expected_risk=var_portfolio_return(elite[0])
 Print('Expected returns of {} with risk of {}\n'.format(Expected_returns,
Expected_risk)) )
 iteration+=1
print('Portfolio of stocks after all the iterations:\n')
[print(hist_stock_returns.columns[i],':',elite[0][i]) for i in list(range(30))]
# The ratios will in turn give results combined with the profit corresponding to
each stock code included in the portfolio
print('Portfolio of stocks after all the iterations:\n')
[print(hist_stock_returns.columns[i],':',elite[0][i]) for i in list(range(30))]
print('\nExpected returns of {} with risk of
{}\n'.format(Expected_returns, Expected_risk))
fitness_fuction(elite[29])
```

After this process, result will generate the weight distribution of the stocks included in the portfolio and the expected return and risk level of the entire category most suitable after undergoing the mutation and hybrid loops.

## IV. Results

The study used data from six different countries in Southeast Asia to create six portfolios with varying rates of return and risk. Our findings, as seen in Fig. 1, show that Vietnam, with the top VN30 stock portfolio, has produced the highest expected profit result among the six countries with an incredible rate of approximately three times the expected profit margin of 60.23%. Followed by Indonesia reaching 53.13% and the third place is Thailand with a profit rate exceeding the target of 42.76%. However, accompanied by high profitability is a risk that is also a worrying risk when bringing the highest profitability among the six countries considered in Southeast Asia. Vietnam ranked 2$^{nd}$ in terms of portfolio risk at 4.6%, however, significantly the risk of the portfolio of 30 companies in the Indonesian market accounted for 6.04%, reaching the highest risk level of 4.04% reaching the highest risk level. However, these findings failed to meet the author's initial risk expectations. So it can be commented that when comparing the top 30 companies in the stock market with the stock codes reaching the largest capitalization in each country, Vietnam's portfolio gives positive results and is worth investing more than the Indonesian portfolio with higher profits and lower risks. However, does the Vietnamese category produce the best results? The saying "The higher the profit, the greater the risk" is an immovable rule when it comes to investing; everyone has the price of that action, and paying the desired price for a high profit, the higher the risk.

When it comes to investing, any investor wishes to own a portfolio with stock codes that not only bring high returns, but risk must also be an acceptable variable. When it comes to risk, the study results indicate that all five categories meet the standard deviation expectations that the category brings except for one category that does not meet the expected risk requirements of 30 companies in the Indonesian market mentioned above. However, when it comes to profitability, there are only five categories that bring positive results and exceed the expectations of the author: Vietnam, Thailand, Indonesia, Philippines, and Singapore; Only the Malaysian portfolio reached 19.46% of the profit margin, but in return for the risk that the portfolio overcomes only accounts for nearly 1%, an extremely difficult to figure, the fact that the figure of 19.46% is still an adequate number and if we convert all the capital distributed into stock codes less than 1% into the Stocks with higher allocations can still raise the value of the profitability of the whole portfolio but also mean that the risk will increase.

In general, only four over six markets met both in terms of profitability and risk expectations of the author: Vietnam, Thailand, Philippines, and Singapore. On the other hand, when the author sets an expectation of a return of 20%, regardless of the risk of the whole category, five categories meet the requirements, in order of declining profits: Vietnam, Indonesia, Thailand, the Philippines and Singapore. When the author considers only the expected risk of the portfolio of 5%, which means setting the risk level more important than the expected profit level, five categories meet the requirements presented in incremental risk: Malaysia, Singapore, Thailand, Philippines and Vietnam. Therefore, it is difficult to expect a portfolio with high returns with a lower level of risk. Our findings are consistent with the risk and return trade-off.

To be able to provide detailed statistics of six categories in six Southeast Asian markets, the author uses the Sharpe ratio after the optimal portfolio allocation results have produced the same results with the same data on closing prices with the timeline from 07/2018 to 04/2021 of the 30 companies with the largest market capitalization in each country has produced a result. Sharpe's ratio declined, led by Malaysia with 19.81, Singapore with 15.07, Vietnam with 12.67, Thailand with 9.33, Indonesia with 8.46 and Philippines with 6.9 (see Fig. 2). The Sharpe ratio is a ratio that represents the premium value of the portfolio that an investor will receive per unit of risk. The smaller this ratio, the smaller the risk premium, and vice versa. The larger the ratio, the greater the risk premium.

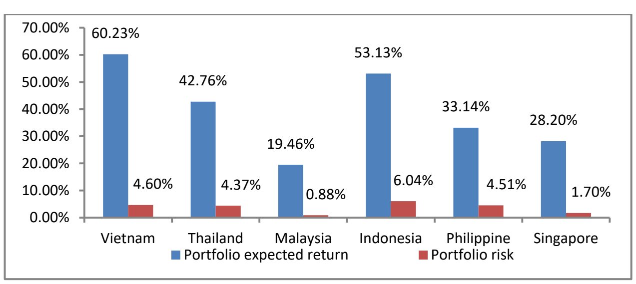


Fig. 1. Portfolio expected return and portfolio risk of Asian market

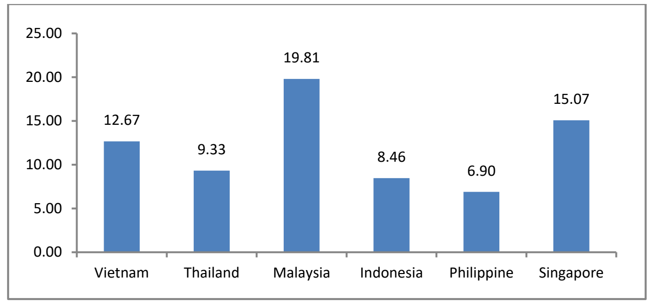


Fig. 2. Sharpe ratio of Asian market

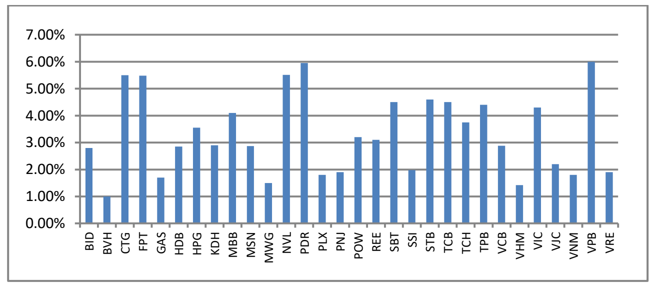


Fig. 3. Optimal portfolio of Vietnam market

The results showed that although Vietnam has the highest profitability and Malaysia offers relatively low profits investing in the Malaysian market still preserves much more profitability than other highly profitable categories. On the other hand, the Philippines market, as analyzed in the above subsection on the level of profitability and risk that the market brings, with the profitability brought although higher than 20% of the expected profit rate of the author. However, the risk of this market is very high, reaching 4.51%, overwhelmed the risk in the Thai market (4.37%). Therefore, when calculating the Sharpe ratio, the Philippines is a market where the adjusted profit above the risk level is much less competitive than the Thai market.

For Vietnamese stocks, After 40 iterations, the model has found an optimal solution for profitability and risk limitation, resulting in the allocation of investment capital into stocks in the VN30 basket. Surprisingly, in the stocks with the historical data levels used as research, there were five stocks with an allocation of over 5% respectively: VPB, PDR, NVL, CTG and FPT with weights of 6.01%, 5.95%, 5.51%, 5.50% and 5.48% respectively. The top five stocks in the allocation table belong to the real estate, technology and banking sectors. However, there are three stocks whose portfolios give relatively low allocation results: BVH, VHM and MWG, 1%, 1.42% and 1.50%, respectively (refer Fig. 3).

For Thai stocks, as can be seen from Fig. 4, the allocation of over 5%, four stocks invested quite a lot in the category: COM7, KTC, CPF and DELTA, respectively. The allocation levels are 7.03%, 6.93%, 6.29% and 6.2%, respectively. In particular, according to the research, we can see that com7 stock code is a company based in Thailand with the business of operating information technology products. This is a company in the form of distributing branded technology products. KTC is a company that serves credit card services and provides personal or business loans. CPF is an agricultural and food corporation specializing in producing the world's largest feed and shrimp. DELTA is an exporting company and a manufacturer of electronics and a flexible power supply. However, in the portfolio, a stock with a fairly low allocation of less than 1% is BBL. This is an operating bank headquartered in Bangkok, Thailand. In general, capital flows are evenly distributed, and no assets are excluded from the stock. This is the leading group of stocks in terms of market capitalization and high liquidity, so the proportion of allocation is evenly arranged is also an inevitable factor.

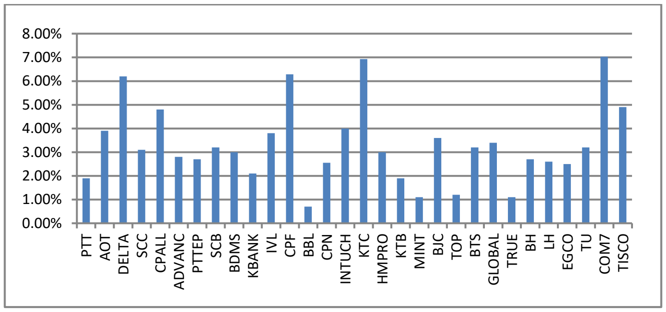


Fig. 4. Optimal portfolio of Thailand market

This is a reasonably particular category for Malaysian stocks because of the profitability and risk that the portfolio generates. Compared to the profitability that the portfolio brings is the lowest of the six velvet research markets. This is also the category with a respectable risk of less than 1%, Five times smaller than the author expected. Moreover, it can be said that although it does not meet the original target set by the author in terms of profitability, with a risk result of only 0.88%, generating a profit of 19.46% of the whole portfolio is a very positive result. The study results have shown the results of the distribution of the proportion of capital investment evenly in stocks. Above the allocation of 5% of the proportion of investment in the portfolio, the result of 4 companies: PMET, TPGC, TLMM and WPHB, with capital allocated of 6.86%, 6.22%, 5.94% and 5.39%, respectively. Most of the above companies are in the form of manufacturing products in aluminum, medical, other services in telecommunications and transportation. However, two stocks are allocated investment levels of less than 1% in the portfolio: PETR (0.88%) and DSOM (0.48%) as evident from Fig. 5.

The results of the research paper in the Indonesian market showed that with the segment of the allocation density of over 5%, there are five stock codes such as EMTK, TBIG, SMMA, TPIA and MEGA with the following proportions as follows 7.39%, 6.8%, 6.78%, 5.62% and 5.28% respectively (see Fig. 6) TBIG and EMTK are two companies specializing in telecommunication technology and the second-largest media company in Indonesia. SMMA specializes in the comprehensive, integrated financial sector, including banking, insurance, asset management and security. TPIA is the leading petrochemical production company using world-class advanced factories. Finally, MEGA is a domestic bank in Indonesia. There are three stock codes: SMGR, BRIS, and ASII, with an allocation of 0.99%, 0.93%, and 0.66%, respectively. SMGR is a state-owned company specializing in domestic cement production, BRIS is a state-owned Islamic bank of Indonesia, ASII is a large corporation specializing in supplying automobile and motorcycle products bearing the brand of Toyota, BMW, Isuzu, Peugeot are the largest in Southeast Asia.

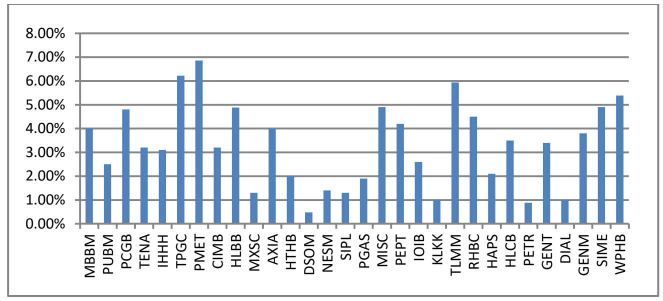


Fig. 5. Optimal portfolio of Malaysia market

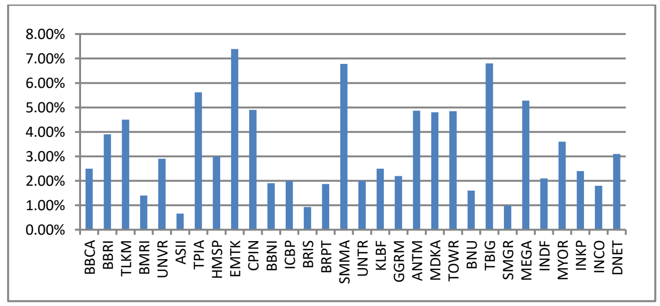


Fig. 6. Optimal portfolio of Indonesia market

For the Philippine market, the author used data from the 30 companies with the highest market capitalization to review and research to deliver fairly consistent attribution results. The results show that there are quite a few companies allocated a proportion of investment above 5%, such as EMP – the largest spirits company in the world with a staggering 7.24%, followed by stock codes such as ACEN, UBP, AGI, HVN, LTG, FGEN, TEL with allocations of 6.85%, 6.31%, 6.26%, 6.19%, 6%, 5.97% and 5.44% respectively as seen in Fig. 7. Among them are companies in electricity production, solar energy, real estate, multi-sectors, serving telecommunication services. However, there are still five companies with an investment capital allocation of less than 1% in the list, including the following stock codes AEV, JGS, SECB, SMC, and ALI. These are primarily multidisciplinary companies in consumption, air transport, and banking.

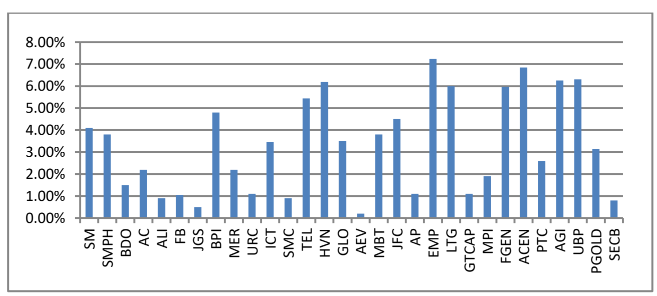


Fig. 7. Optimal portfolio of Philippine market

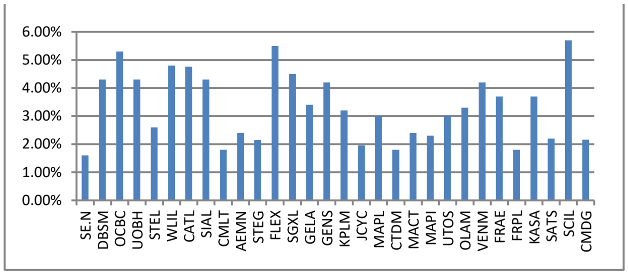


Fig. 8. Optimal portfolio of Singapore market

In the Singapore Stock Exchange, the author used 30 companies listed on the stock exchange with a high market capitalization, and more than half of the portfolio was stock codes in the top straits time index, which is considered a benchmark index for the Singapore market calculated by Singapore Press Holdings, Singapore Exchange and FTSE Group. Moreover, the study results worked when there was no stock code with an investment allocation ratio of less than 1% in this category, which explains that any stock code that appears in the portfolio is highly appreciated and has great capital investment potential. Although the highest allocation in the portfolio has not reached 6% compared to other countries, the even distribution between stocks is very high and homogeneous. The following image shows three stock codes above the 5% allocation: SCIL, FLEX and OCBC, respectively. The proportion is 5.7%, 5.5% and 5.3%, respectively. In particular, SCIL is an urban development and energy company, and FLEX specializes in manufacturing the third-largest multinational electronics in the world. OCBC is a stock code in the banking sector (see Fig. 8).

## V. Conclusion

The content of the study is to address and examine the theoretical foundations of portfolio management based on Markowitz's portfolio theory and theories of genetic algorithms such as definitions and cycles contained in the algorithm to establish the portfolios of six different markets in Southeast Asia. The author uses the Python programming language to interpret and assist in producing results and following the steps contained in the genetic algorithm. Based on theory and related studies, the author has obtained very positive results on allocating stocks included in each category with the corresponding profitability and risk level.

The result consists of four categories that meet the author's expectations: the categories in the markets of Vietnam, Thailand, Philippines, and Singapore. The two categories alone are pretty positive in terms of results, namely Malaysia and Indonesia. Although Malaysia has not met the profitability, the risk ratio of less than 1% is a very potential figure. An immovable rule in investing is that the higher the return on their assets, the higher the risk-taking bravery they need. Therefore, for the portfolio of 30 stocks from the Indonesian market with outstanding portfolio results, there is still a level of risk that needs to be accepted, and there must be a trade-off.

Considering the Sharpe ratio of the categories, topped by the 30 companies with the largest market capitalization in Malaysia with 19.81% of the profitability that can be achieved per unit of risk. Although the profitability of this market is not satisfactory to the author, this is still a category worth seriously considering and invests. Ranked second is Singapore, with a Sharpe rate of 15.07%. Thirdly, the Vietnamese market with a not-so-modest figure of 12.67. Ranked fourth in Thailand, the country of golden pagodas reaching 9.33% of the risk compensation level of the whole category. The fourth market among the six countries in Southeast Asia is Indonesia, with a research rate of 8.46% and the Philippine market of 6.9%. After the above statistical step, the significance level is realized when both categories are not satisfactory in terms of profitability, and the expected risk is Malaysia and Indonesia. The Sharpe rate in the Malaysian market is higher than in Indonesia.

## References

[1] Yang, X, "Improving portfolio efficiency: A genetic algorithm approach". Computational Economics, vol 28, 2006.

[2] Lin, C.M., & Gen, M. "An effective decision-based genetic algorithm approach to multiobjective portfolio optimization problem". Applied Mathematical Sciences, Vol. 1, no. 5, pp. 201-210, 2007.

[3] Sefiane, S., Benbouziane, M. "Portfolio Selection Using Genetic Algorithm". Journal of Applied Finance & Banking, Vol. 2, No. 4, pp. 143-154, 2012.

[4] Dubinskas, P., & Urbšienė, L. "Investment portfolio optimization by applying a genetic algorithm-based approach". Ekonomika, Vol. 9, no.6, pp.66-78, 2017.

[5] Cheong, D., Kim, Y.M., Byun, H. W., Oh, K. J., & Kim, T. Y. "Using genetic algorithm to support clustering-based portfolio optimization by investor information". Applied Soft Computing, Vol. 61, pp. 593-602, 2017.

[6] Liu, C., Gan, W., & Chen, Y. "Research on Portfolio Optimization Based on Affinity Propagation and Genetic Algorithm". In 2017 14th Web Information Systems and Applications Conference (WISA) (pp. 122-126). IEEE.

[7] Adam, H. Portfolio Management. Book, 2021.

[8] Ahn, W, Hee S. L, Hosun, R, Kyong J. "Asset Allocation Model for a Robo-Advisor Using the Financial Market Instability Index and Genetic Algorithms". Sustainability, Vol. 12, no. 3, pp. 849, 2020.