Modern Portfolio Theory Application and Python Demonstration of Volatility Minimization using Asset Pairs

Alper Ülkü,[*] Reggie C. Pantig,[†] and Tchinda T. Romuald[‡]

(Dated: August 2, 2022)

Efficient frontier is the optimal locus of return-risk pairs of all efficient portfolios introduced by Markowitz in 1952. In this paper, we perform volatility minimization and efficient frontier analysis of 2 assets with low covariance ("CANTE.IS" and "PETUN.IS"), from equity market of Turkey (BIST), we construct the minimum variance portfolio set targeted to be exercised by risk-averse investor. Instead of the well-known covariance matrix method, we utilized Monte Carlo simulations of 1000 portfolios with Dirichlet-conditioned randomized weights, we selected portfolio with the least volatility, maximum Sharpe ratio observing that, final minimum variance portfolio as compared to the individual assets performance in volatility, maximum draw down and stability.

Keywords: Volatility, Efficient Frontier, Portfolio Management, Modern Portfolio Theory, Mean Variance Model (MVM), Minimum Variance Portfolio, Portfolio Diversification, Sharpe Ratio

Contents

I. INTRODUCTION

Individuals prefer to direct their savings to various investment assets in order to protect them from the negative effects of uncertainties regarding the future. These investment tools can be risk-free investment tools such as treasury bills, government bonds, bank interests, as well as risky investment tools such as mutual funds, stocks and foreign exchange. Uncertainty about how the investor should use his money poses a significant problem. This situation brought along a detailed consideration of the selection and management of financial assets.

Markowitz's work [1] has made various contributions to portfolio management: The first of these contributions is that the portfolio with low risk from portfolios providing the same return and the portfolio with the highest return from portfolios with the same risk level should be preferred. The investor can create a portfolio combination according to his risk level on the effective frontier showing the highest expected return and lowest risk level. Second, with an appropriate diversification, the risk of a portfolio could be much less than the risk of the securities that make up the portfolio. For this reason, securities with high correlation with each other need to be avoided to be included in the portfolio. In his legendary seminal publication H.M. Markowitz stated the foundations of portfolio selection theory [1]. He showed that under certain given conditions, an investors portfolio choice can be reduced to balancing two dimensions, the expected return on the portfolio and its variance. Due to the possibility of reducing risk through diversification, the risk of the portfolio, measured as its variance, will depend not only on the individual variances of the return on different assets, but also on the pairwise covariances of all assets. For a

[*]Electronic address: alperulku1970@gmail.com
[†]Electronic address: reggie.pantig04@gmail.com
[‡]Electronic address: romualdtr@gmail.com

review of the portfolio theory till 1997, see Ref. [2].

Due to the fact that the modern portfolio theory has been made as a tool for decision [3, 4], many authors have considered improving or extending the theory in order to find more efficient portfolios. We mention a few of them in the following literary survey. Authors in Ref. [5] constructed, with Mean-Variance Model (MVM), a portfolio of securities with equal distribution. The relationship between the effective limit of return and the risk was analyzed geometrically and determined that the optimal portfolio is the tangency portfolio at which portfolio has optimally minimized risk and maximized expected return using MS Excel Solver Module. Number of starting assets were observed to be much less than expected and seemed insufficient to achieve an optimal portfolio.

In Ref. [6] made the performance evaluation of private pension funds between 2003 and 2007 using the Sharpe, Treynor and Jensen criteria and revealed the performance status of 10 pension mutual funds within the scope of the study according to each portfolio performance criterion. Work constructed no output portfolios but rather evaluated the Sharpe ratios of each pension fund and constructed a table which served to selection of a single fund with best Sharpe ratio.

In the paper in Ref. [7] provided a general framework that relied on solving the minimum variance problem but subject to the additional constraint that the norm of the portfolio weight vector smaller than a given threshold. Approach devised an additional constrained so that 2-norm of the weights matrix so be less that the this individual weight of each asset in an EW portfolio. This approach comes with useful and effective results, introduction of 2-norm as a constraint makes the portfolio much robust with less return yield yet better risk aversion.

The authors in Ref. [8] analyzed the long-term performances of the log-optimal portfolios based on the Stochastic Portfolio Theory and the portfolios were created according to Mean Variance Model were compared. It was determined that in 29 of 32 measurements, log-optimal portfolios performed better than portfolios created with Mean Variance Model. Using monthly returns between January 2000 and December 2002, portfolios on average-variance and log-optimal effective limits were determined for various return-risk and growth-risk values. While determining these portfolios, optimum asset weights were determined with the Sharpe ratio (average-variance) and growth (log-optimal) maximization. Given the mean-variance and log-optimal effective limits, maximum Sharpe ratio and maximum growth (log-optimal) portfolios were determined to yield that log-optimal boundary patterns, growth after the log-optimal point decreased sharply with increased risk. This showed that log-optimal portfolios were formed close to the points where risk is highest on the log-optimal effective boundary. Log-optimal approach proved better return yield but more risk than MVM portfolios. Disadvantage of this approach is while log-normal portfolio performance is depicted on a growth-risk plot, MVM portfolios is shown on the return-risk plot, thus making clear comparisons between two approaches was not possible.

A study in Ref. [9] measured the performance of 17 stocks and 13 portfolios were constructed Markowitz Mean Variance Model (MVM) and Black-Litterman Model using Sharpe, Treynor and Jensen portfolio performance criteria. In the study, daily adjusted prices of stocks belonging to a total of 17 companies traded continuously in the BIST 30 index during the 2003-2009 period were used. The results showed that the performance criteria were more compatible with the Black-Litterman Model.

A performance analysis was conducted using Sharpe, Treynor and Jensen indices of 20 A-type mutual funds issued by brokerage houses and banks in the 2009-2012 period [10]. The results show that although brokerage funds tend to risk free assets, they do not perform as high as bank funds on a yield basis; on the other hand, it was revealed that bank funds, by taking a higher level of risk, provide their investors with more returns. Paper rigorously showed many aspect of usage of Sharpe, Treynor and Jensen indices over Turkish investment mutual funds, also mutual funds market progress quantitavely.

Working in the 2010-2012 period, 34 units operating in Turkey flexible and balanced performance of pension funds, based on the standard deviation [11]. Based on systematic risk with Sharpe, Modigliani, Sortino and Treynor ratios, work aimed to determine the pension mutual funds with the highest and lowest performance by evaluating them with T2 and Jensen performance criteria and an indication that these criteria were sufficiently accurate for evaluation of funds and their corresponding portfolio performance.

Vyniauskas, and others used the standard deviation, alpha, beta, Sharpe criterion and Treynor index to measure the performance of mutual funds, and as a result, they stated that the alpha metric gave the best results, and that no relationship could be established between the performance of mutual funds and other metrics [12]. This analysis is again do not show us anything about generating an efficient portfolio. It only evaluates the funds with respect to Jensen alpha criteria.

Investigation about the impact of the 2008 global financial crisis on the portfolio of stocks traded in the Borsa Istanbul (BIST) banking sector were done in Ref. [13]. The daily return data of the banks between 2005 and 2009 were optimized by Markowitz Mean Variance Model and tested with Sharpe ratio and Jensen alpha. Paper showed dramatic effect of 2008 crisis on via Sharpe and Jensen alpha.

Using Markowitz Mean Variance method [14], modeling was performed based on the monthly closing prices of the stocks included in the BIST Corporate Governance Index (XKURY) between 2009 and 2018, and optimum portfolios were created with the portfolio optimization application and the results achieved were calculated with the expected results. In addition, considering the Sharpe ratios and coefficients of variation, a comparison was made between the Mean Variance Model and the portfolios created with the 6-Best-Return-Stocks model based on the selection of 6 stocks with the highest return and equal weight.

Results supported the validity of the Mean Variance Model, and for us, the most useful among our literature survey is that it showed clearly many efficient portfolios were generated with the MVM, which risk-wisely outperformed their individual assets. We were motivated to select only two assets, returns of which are less correlated, for simplicity of the calculations, since CANTE.IS is among the most volatile assets and PETUN.IS is, in contrast, more stable in volatility.

We organized the paper as follows: In Sect. II, we review the basic statistical needed for portfolio variance, which is central to portfolio management. In Sect. III, we discuss portfolio management and its variance formula. Then in Sect. IV, we deploy the portfolio management theory using the Markowitz model. Finally, we form our conclusive remarks in Sec. V. Furthermore in this paper, we limit our study within two Turkish assets: CANTE.IS and PETUN.IS, and perform data analysis about its return-risk performance from July 30th, 2021 - July 30th, 2022.

II. BASIC CONSTRAINTS

In this paper, we will discuss the corresponding statistical quantities needed in relation to buid up the modern portfolio theory components the mean-variance model (MVM).

A. Mean

Mathematically, mean is the average of a given set of numbers. It is given by the expression

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i, \tag{1}$$

where n is the number of values, and x_i is the data set values. Suppose that we have the data set $x_i = [3, 2, 5, 7, 3, 1, 2, 3, 5, 7]$. Here, $n = 10$. If we find the mean, we then have $\bar{x} = 3.8$.

B. Variance

In the given data set x_i above, how far does each number is from the mean, and from every other number in the set? To answer this, we need the statistical concept called variance which also known as the degree of dispersion. The expression for variance is

$$s_x^2 = \frac{\sum(x_i - \bar{x}^2)^2}{n-1}. \tag{2}$$

It turns out that this is a sample variance. If this is a population variance, $n - 1 \to N$ where N is the population size. For the data set above, we can calculate the variance as $s_x^2 = 4.4$. Variance treats all deviations from the mean the same way, regardless of whether they are less than or greater than the mean. A variance of zero would indicate that each data point is the same.

C. Standard Deviation

Now that we know the variance, it is very easy to calculate the standard deviation, which is the measure of how dispersed a data set relative to the mean:

$$\sigma_x = \sqrt{s_x^2}. \tag{3}$$

In other words, to get the standard deviation, just get the square-root of the variance. For our example above, $\sigma_x = 2.10$. Given the data set, we can see that all values that are within 1 standard deviation except 7 and 1.

D. Covariance

So far, we only considered a single data set x_i. If we have, say another data set y_i, and we are exploring the relationship between these two data sets, a useful statistical measurement to incorporate first is the covariance. Covariance measures the direction of the relationship between two variables. The covariance can be calculated using

$$\text{Cov}(x, y) = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{n-1}. \tag{4}$$

Consider x_i given in the previous subsection, then let $y_i = [4, 2, 6, 5, 3, 7, 4, 2, 1, 5]$. The mean, variance, and standard deviation for the variable y are then $\bar{y} = 3.9$, $s_y^2 = 3.66$, and $\sigma_y = 1.91$. Given the mean and data sets alone, the covariance is $\text{Cov}(x, y) = 0.31$. Since the covariance is positive, the two data sets move in the same direction. If it is negative, we expect x and y to move in the opposite direction.

E. Correlation

While covariance gives us information about the directionality of the relation between x and y, the degree/strength of such movement in relation to each other is called correlation. It can be calculated using

$$\rho(x, y) = \frac{\text{Cov}(x, y)}{\sigma_x \sigma_y}. \tag{5}$$

We already know the covariance, and the standard deviations in x and y. Thus, using the above equation, we find the correlation approximately as $\rho(x, y) = 0.08$. To interpret such, the strength of positive directionality between x and y is quite weak.

III. MANAGING PORTFOLIO VARIANCE AND PORTFOLIO VARIANCE FORMULA

Investors have to keep volatility in mind as well when choosing an investment. For example, a pension fund may need to be extra careful with its money and will want to ensure they are not getting into any extremely volatile investments. There are also hedge funds, which short stocks and even trade volatility with options. Whether one is a risk-taker, or conservative the volatility (risk) of an investment is something you should care about. For some period of time, volatility is a measure of price variability. It typically described by the standard deviation of returns, in a particular context that depends on the definition used. In other words, volatility is the standard deviation of the change in the logarithmic price or a price index during a stated period of time [15]. For the example in the previous section, if x and y represent two different assets, then their standard deviations $\sigma_x = 2.1$ and $\sigma_y = 1.91$ reflects their volatility. It means that asset x is riskier than asset y. We see that individually, we tend to invest in y since gives less risk (but less reward). What could be the effect of, say, if one has a portfolio containing assets x and y? Furthermore, we must keep in mind that these example variables may represent an index, or ETF. This is where the usefulness of portfolio variance comes into play. Consider a portfolio that only contains two assets. The portfolio variance Port_{s^2} can be calculated as

$$\text{Port}_{s^2} = w_x^2\sigma_x^2 + w_y^2\sigma_y^2 + 2w_x^2w_y^2\text{Cov}(x,y), \tag{6}$$

but in terms of correlation, we can use Eq. (5) so we can have

$$\text{Port}_{s^2} = w_x^2\sigma_x^2 + w_y^2\sigma_y^2 + 2w_x^2w_y^2\sigma_x\sigma_y\rho(x,y). \tag{7}$$

Here, we have a new variable w_x and w_y which are the asset weights in the portfolio. To give an example how to compute this, suppose we bought 13 shares of asset x for \$10, while we bought 25 shares of asset y for \$12. The initial price of x is then $X_0 = \$130$, while that of y is $Y_0 = \$300$. The portfolio weights of assets x and y are then

$$\begin{aligned} w_x &= \frac{X_0}{X_0+Y_0}, \\ w_y &= \frac{Y_0}{X_0+Y_0}, \end{aligned} \tag{8}$$

respectively. In particular, $X_0 = 0.30$ and $Y_0 = 0.70$. Next, with our previous knowledge, it is easy to calculate the portfolio's standard deviation:

$$\text{Port}_\sigma = \sqrt{\text{Port}_{s^2}}. \tag{9}$$

As a final note, Eq. (7) shows that we need to minimize correlations in order to minimize portfolio variance. Owning uncorrelated assets illustrates the efficient benefits of diversification.

Having these information in mind, there are 3 things we can do to minimize variance.

- We can pick assets with lower standard deviations of returns, which might seem obvious, but if we really want to reduce the variance of our portfolio, the simplest thing to do is just pick assets that have relatively low volatilities;
- Invest a higher percentage of ones portfolio into less risky asset(s). From our previous example with assets x and y, we could further reduce our portfolio variance by investing a bigger portion of our funds into y since it had a lower standard deviation than x;
- Check for assets with a low covariance. If one look at the right side of our portfolio variance function, we can have the covariance of two assets as a function parameter. If we can reduce that, we can reduce the overall variance of our portfolio;

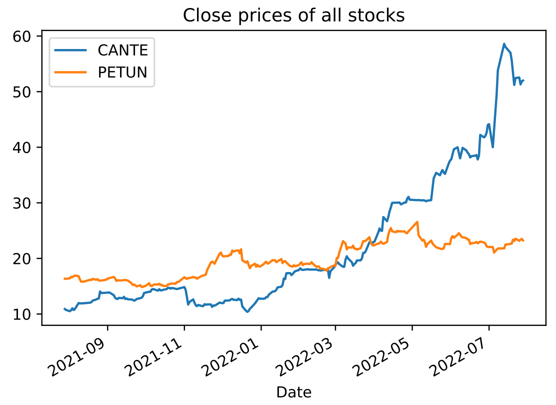

FIG. 1: CANTE.IS and PETUN.IS closing prices.

Statistics	CANTE.IS	PETUN.IS
mean	0.006307	0.001492
std.	0.030140	0.021854

TABLE I: Mean and standard deviation for CANTE.IS and PETUN.IS.

where the third case is due to an investor may still want to invest in a riskier asset like x. This is an important point and should be expanded upon further.

IV. DEPLOYING MODERN PORTFOLIO THEORY

In this section, we apply calculations to the stock market assets of Turkey: CANTE.IS and PETUN.IS. can be checked and downloaded from Yahoo Finance. After we import the necessary libraries, we extracted the data on price. See Fig. 1. plotting daily prices of two assets.

Next is the portfolio's daily returns, where we used the ratio of present day's return to yesterday's return. We found the following: We also considered the annual log returns: Based on the resulting standard deviations, CANTE.IS is riskier than PETUN.IS.

Statistics	CANTE.IS	PETUN.IS
mean	1.59	0.38
std.	0.478	0.347

TABLE II: Annualized mean and standard deviation for CANTE.IS and PETUN.IS.

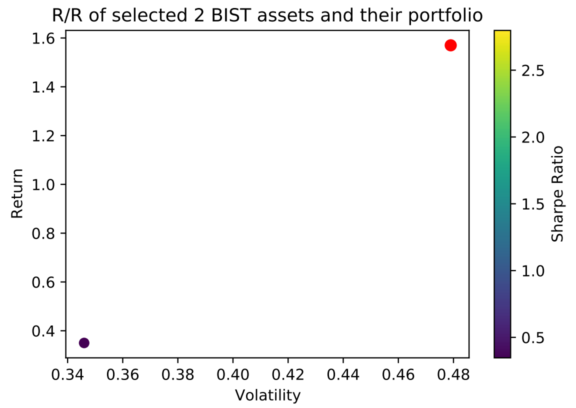

FIG. 2: In this plot, max Sharpe ratio is 2.845, max Sharpe ratio return is 1.59, and max Sharpe ratio volatility is 0.478.

Used often in modern portfolio theory is the Sharpe ratio, which is measures the performance of an investment compared to a risk-free asset, after adjusting for its risk. A portfolio with higher Sharpe ratio, the best risk-adjusted return there is. To calculate the Sharpe ratio, we have taken the risk free rate of 23% (1−year bank interest rate) and used the formula

$$S = \frac{R_p - R_f}{\sigma_p}, \tag{10}$$

where R_p is the annual return, R_f is the risk free rate, then σ is the volatility. We find that for CANTE.IS, $S = 2.845$ while for PETUN.IS, $S = 0.432$. In Fig. 2, we plot the return versus volatility.

As an alternative analysis, we also used pair plots. See Fig. 3. The pair plots also signify the same result that there is no pair of stocks with high negative correlation. We don't find any pair-plot with upper-left to lower-right pattern. The pairs with high positive correlation have scatter plot with lower-left to upper-right pattern, and other pairs don't form any pattern.

A. Markowitz Model

The Markowitz model assists in the selection of the most efficient portfolios by analyzing various possible portfolios

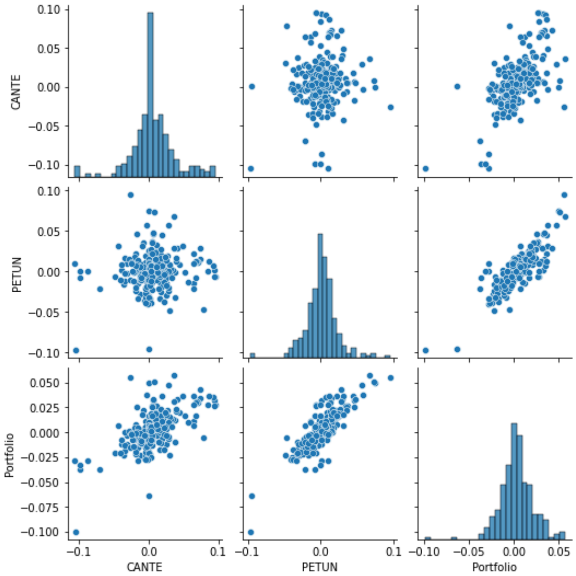

FIG. 3: Pair plots and histograms of individual and minimum variance portfolio returns

of the selected stocks. Here, the assets are modeled by their expected return, $E[R]$ and their risk, which is expressed as their standard deviation, σ. The investment decisions are expressed by investing 100% of our wealth in assets(here, stocks), where each particular investment represents a proportion of our total wealth. We invest w_i in $stock_i$ for every i, such that

$$\sum_{i=1}^{n} w_i = 1 \tag{11}$$

The expected return of the portfolio constructed would be

$$E[R_p] = \sum_{i=1}^{n} w_i E[R], \tag{12}$$

and the risk associated with the portfolio would be

$$\sigma^2(R_p) = \sum_{i=1}^{n} w_i^2 \sigma^2(R_i) + \sum_{i=1} \sum_{i \neq 1} w_i w_j \sigma(R_i) \sigma(R_j) \rho_{ij}, \tag{13}$$

where $E[R_i]$ is the annual expected return of ith stock, $\sigma(R_i)$ corresponds to annual standard deviation of ith stock and ρ_{ij} is the correlation between the logarithmic returns ith and the jth stock. $E[R_p]$ is the annual expected return of the portfolio and $\sigma(R_p)$ is the risk associated with the portfolio (Also the standard deviation of the portfolio). An efficient portfolio is one that maximizes return for a given level of risk. Our task is to select adequate weights w_i to get the efficient portfolio

We can implement this by means of letting $W_{1\times n}$ be a array containing the weights w_i such that $\sum_{i=1}^{n} w_i = 1$ and $E[R]_{n\times n}$ be another array containing annual expected returns of n stocks present in the portfolio and C be the covariance matrix of annual returns of stocks, then we have

$$E[R_p] = WE[R], \tag{14}$$

$$\sigma^2[R_p] = W^T C W. \tag{15}$$

For our portfolio under Markowitz model, we found the following: $R_p = 0.986, R_f = 0.308$, and $S = 2.434$.

B. Monte Carlo Simulation for random portfolios

To generate random portfolios, we ran a Monte Carlo simulation. We will use the results of simulation to draw an efficient frontier. We find that the return for maximum Sharpe ratio is 1.275, while the max Sharpe volatility is 0.375. Fig. 4 shows comparison of all portfolio combinations generated in Monte Carlo Simulation in terms of their risk and return. The red dot corresponds to the portfolio having the highest Sharpe ratio among the generated portfolios. Such a portfolio may not be

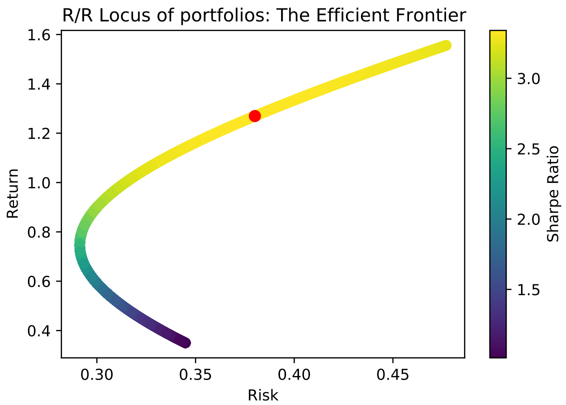

FIG. 4: The efficient frontier of R-R pairs of all portfolios possible with combination of CANTE.IS and PETUN.IS assets

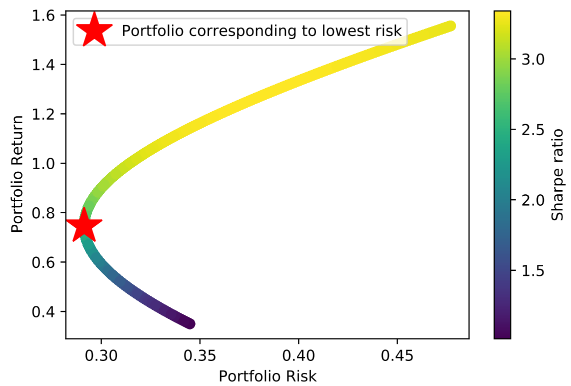

FIG. 5: MINVAR Portfolio R-R position

the one with highest Sharpe ratio as we are plotting random portfolios. It is just the portfolio with highest Sharpe ratio among all the randomly generated portfolios. We also generated optimized portfolios subject to various conditions. Such a hyperbolic plot is called 'Markowitz's Bullet'.

In the next section, we will show our results for the efficient frontier but as a prerequisite, we need to use optimization to find the maximum Sharpe ratio and then to find the portfolio that has the minimum risk for a given expected return. The reason for this is that investors want to have a portfolio with a fixed target return, or find a portfolio set-up that would provide return but with minimal risk. The result for the required weights to have an optimized maximum Sharpe ratio are $0.330, 0.670$.

C. The Efficient frontier

The efficient frontier is the set of optimal portfolios that offer the highest expected return for a defined level of risk or the lowest risk for a given level of expected return. Portfolios that lie below the efficient frontier are sub-optimal because they do not provide enough return for the level of risk.

The efficient frontier is different for different investors, depending upon the assets they are holding. There is nothing like a single optimal portfolio. The efficient frontier is the collection of optimal portfolios. The investors can choose any optimal portfolio depending upon the risk they can take. For our case for CANTE.IS and PETUN.IS, see Fig. 5. The weights of minimum volatility for CANTE.IS is 33.02 and 66.98 for PETUN.IS. Assuming that we have an amount of capital as 100000 TL, the portfolio consists of 66.98% PETUN.IS and 33.02% CANTE.IS. Or, 67000 TL for PETUN.IS, and 33000 TL for CANTE.IS. In terms of portfolio weights, the daily average return is 0.355 and the corresponding volatility is 1.854. Finally, we individually generated a simple tear for CANTE.IS and PETUN.IS, as well as the portfolio that holds them, to backtests the annual and cummulative returns, annual volatility, Sharpe ratio, Calmar ratio, Stability, Max drawdown, Omega and Sartino ratio, Skew, Kurtosis, Tail ratio, and daily value at risk. The plots in Figs. 6-8 shows the cumulative returns, Sharpe ratio, and maximum drawdown in a time series of 12 months.

V. CONCLUSION

The Sharpe ratio and maximum drawdown performance serves best for risk-aversed investor in selecting portfolios. In this paper, selected assets were firstly analysed on return and volatility, correlation, covariance for individual return-

FIG. 6: Pyfolio performance output for CANTE.IS

FIG. 7: Pyfolio performance output for PETUN.IS

	Backtest
Annual return	347.4%
Cummulative return	342.1%
annual volatility	47.9%
Sharpe ratio	3.38
Calmar ratio	10.84
Stability	0.82
Max drawdown	-32.0%
Omega ratio	1.93
Sortino ratio	5.61
Skew	-0.09
Kurtosis	3.09
Tail ratio	1.92
Daily value at risk	-5.4%

TABLE III: CANTE.IS

	Backtest
Annual return	36.7%
Cummulative return	36.4%
annual volatility	36.7%
Sharpe ratio	1.07
Calmar ratio	1.69
Stability	0.73
Max drawdown	-21.8%
Omega ratio	1.22
Sortino ratio	1.61
Skew	0.01
Kurtosis	4.42
Tail ratio	1.06
Daily value at risk	-4.2%

TABLE IV: PETUN.IS

risk of performance with these criteria for duration of last 1 year (from July 30th, 2021 until July 30th, 2022) and the minimum variance portfolio was achieved with combination, weights of which are results from the Monte Carlo simulation of randomized weights of selected assets. The efficient frontier hyperbola together with joint portfolio of minimized volatility were computed using our Python Code given in Appendix. In our forthcoming work, it seem promising and logical to enlarge the asset list so that optimizations can be made to cover for entire exchange. Also we may try to examine the low and high correlated asset pairs and examine their portfolio performance and effectiveness of volatility minimization with respect to that

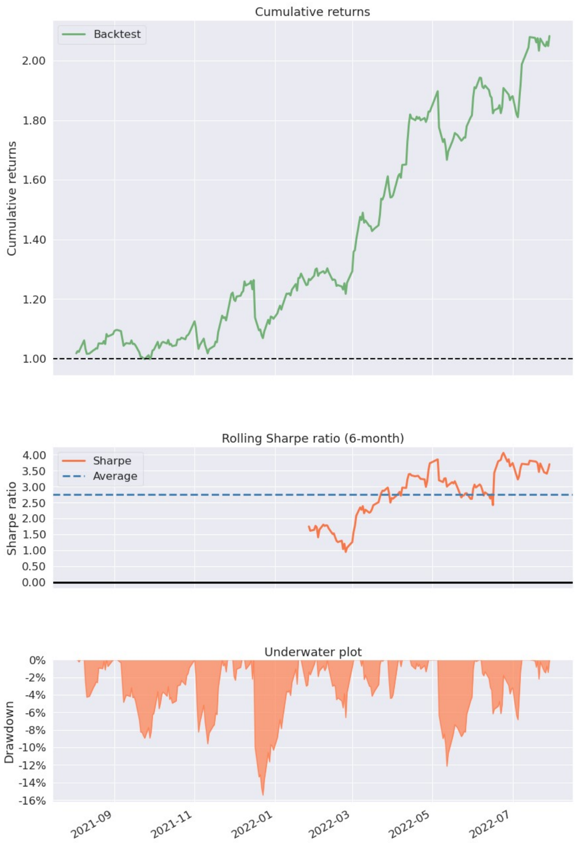

FIG. 8: Pyfolio performance output for MINVAR portfolio

	Backtest
Annual return	109.5%
Cummulative return	108.2%
annual volatility	29.3%
Sharpe ratio	2.67
Calmar ratio	7.09
Stability	0.91
Max drawdown	-15.4%
Omega ratio	1.60
Sortino ratio	4.18
Skew	-0.48
Kurtosis	4.02
Tail ratio	1.48
Daily value at risk	-3.4%

TABLE V: Portfolio

pair correlations. We may also add portfolios with maximum Sharpe ratios for more risky investments.

[1] H. M. Markowitz, Cowles Foundation Monograph **16** (1959).

[2] E. J. Elton and M. J. Gruber, Journal of banking & finance **21**, 1743 (1997).

[3] A. Rani, International Journal of Management Research and Reviews **2**, 1164 (2012).

[4] O. Iyiola, Y. Munirat, and C. Nwufo, Journal of Accounting and Taxation **4**, 19 (2012).

[5] Y. Topal and K. İlarslan, Afyon Kocatepe Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi **11**, 219 (2009).

[6] H. DAĞLI, B. Semra, and E. Bünyamin, Muhasebe ve Finansman Dergisi , 84 (2008).

[7] V. DeMiguel, L. Garlappi, F. J. Nogales, and R. Uppal, Management science **55**, 798 (2009).

[8] E. Ustaoğlu and E. Altay, Uygulamalı Sosyal Bilimler Dergisi **1**, 67 (2017).

[9] A. G. D. T. Çalışkan, (2011).

[10] H. Yolsal, Marmara Üniversitesi İktisadi ve İdari Bilimler Dergisi **32**, 343 (2012).

[11] H. AYAYDIN, Çukurova Üniversitesi Sosyal Bilimler Enstitüsü Dergisi **22**, 59 (2013).

[12] P. Vyšniauskas and A. V. Rutkauskas, Verslas: teorija ir praktika **15**, 398 (2014).

[13] U. Umut and A. Gökçe, Finansal Araştırmalar ve Çalışmalar Dergisi **7**, 209 (2008).

[14] G. P. Başar and Ö. Kuvat, Optimum Ekonomi ve Yönetim Bilimleri Dergisi **7**, 161 (2020).

[15] S. J. Taylor, *Asset price dynamics, volatility, and prediction* (Princeton University Press, 2011).

Appendix. Python codes used in this paper

Here, we present the codes used to generate the numeric and graphical outputs in this paper.

```
# Installing and Importing the libraries

# !pip3 install pandas
# !pip3 install numpy
# !pip3 install matplotlib
# !pip3 install seaborn
# !pip3 install scipy
# !pip3 install nsepy
# !pip3 install pyfolio

from datetime import date, timedelta
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from scipy.optimize import minimize
from nsepy import *
import pyfolio
import os
import pandas_datareader.data as web
from datetime import datetime
import time
import yfinance as yf
```

Listing 1: importing necessary libraries

```
# BIST10 = [ "XU100.IS",
#             "AKBNK.IS","AKSEN.IS","ALARK.IS", "ARCLK.IS", "ARDYZ.IS",
#             "HEKTS.IS","INDES.IS","SASA.IS" , "TTRAK.IS", "PKART.IS"
#           ]


# BIST10 = [ "ALKA.IS", "CANTE.IS", "ISGYO.IS","PETUN.IS","SILVR.IS","SAMAT.IS",
#             "DAGHL.IS", "ASUZU.IS", "LKMNH.IS", "ARMDA.IS", "TSGYO.IS"
#           ]


BIST10 = [ "CANTE.IS", "PETUN.IS"]


stock_list = BIST10
```

Listing 2: Partial list of assets in BIST exchange of Turkey.

```
look_back = 365
current_date = date.today().isoformat()
days_before = (date.today() - timedelta(days=look_back)).isoformat()

days_after =  date.today().isoformat()

print("\nCurrent Date: ",current_date)
print(f"{look_back} days before current date: ",days_before)
print("today : ",days_after)

dateStart = days_before
START_DATE = days_before
END_DATE   = days_after
```

Listing 3: Backtest days calculation from lookback period

```
LENGTH = 64

start_time = datetime.now()

price_list = []
```

```
print(LENGTH*"*")
print("Starting Download ...")
print(LENGTH*"*")
for tick in stock_list:
        print(f"Downloading {tick}")
        yf_tick = yf.Ticker(tick)
        df = yf_tick.history(interval='1d', auto_adjust=True, start=START_DATE, end=END_DATE, back_adjust
    = True, rounding=True)
        df.dropna(how='all', inplace=True)
        price_list.append(df)
print(LENGTH*"*")
print('Download complete...Time elapsed (hh:mm:ss.ms) {}'.format(datetime.now() - start_time))
print(LENGTH*"*")

## Save datafiles to disk

for i,df in enumerate(price_list):
        df.to_csv(f"{stock_list[i]}.csv")
```

Listing 4: Downloading assets and writing to dataframes and disk

```
Stocks = stock_list
pf_data = pd.DataFrame()
li = pd.DataFrame() # my real portfolio dataframe
rets = pd.DataFrame()
names = []
count = len(Stocks)

# os.chdir(wd)
#for file in sorted2.Stock:
for file in Stocks:
    pf_data = pd.read_csv(f"{file}.csv", index_col='Date', parse_dates=True, keep_date_col = True,
    infer_datetime_format=True, dayfirst=True, decimal="." )
    li = pd.concat( [li,pf_data['Close']],axis=1) #, ignore_index=True)

    st_name = file.split('.',maxsplit = 1)
    names.append(st_name[0])

li.columns = names

li = li.rename_axis(index="Date")

pf_data = li.sort_values(by=['Date'], ascending=[True])
pf_data.to_csv('BIST10.csv')
pf_data

pf_data.info()

plt.figure()
pf_data.plot(subplots = True,figsize = (10,10))
plt.title("Close prices of all stocks")
plt.show()
```

Listing 5: Organizing dataframe

```
log_returns = np.log(pf_data/pf_data.shift(1))
log_returns

log_returns.describe()

np.round(log_returns.mean(),4)

yearly_rets = np.round(log_returns.mean() * 252,2) # Mean returns annualized for year
yearly_rets

vol = np.round(log_returns.std()*np.sqrt(252),3) # annualized version of std deviation
vol

risk_free_rate = 0.23

```

```
sharpe = (yearly_rets - risk_free_rate)/vol
sharpe

max_sr_vol = vol[sharpe.argmax()] # risk corresponding to maximum sharpe ratio
max_sr_ret = yearly_rets[sharpe.argmax()] # return corresponding to maximum sharpe ratio

plt.figure(figsize=(8,8))
plt.scatter(vol, yearly_rets, c=sharpe, cmap='viridis')
plt.colorbar(label='Sharpe Ratio')
plt.xlabel('Volatility')
plt.ylabel('Return')
plt.title(f"R/R of selected {len(BIST10)} BIST assets and their portfolio")
plt.scatter(max_sr_vol, max_sr_ret,c='red', s=50) # red dot
plt.show()

print(f"Max Sharpe Ratio = {sharpe.max()}")
print(f"Max Sharpe Ratio Return = {max_sr_ret}")
print(f"Max Sharpe Ratio Volatility = {max_sr_vol}")

log_returns.cov()

log_returns.corr()
```

Listing 6: Calculating annualized values of volatility and returns of portfolio

```
plt.figure(figsize=(8,8))
sns.heatmap(log_returns.corr(),linecolor='white',linewidths=1,annot=True)
plt.title("correlation heatmap of stocks")
plt.show()

sns.pairplot(pf_data,palette='coolwarm')
plt.show()
```

```
# A function for generating a numpy array containing random weights that add upto 1
def RandWeights(size):
    weight = np.random.dirichlet(np.ones(size))
    return weight

risk_free_rate = 0.235 # quite high in Turkey !

# A function to generate the avg return, risk and the sharpe ratio of the portfolio
# correponding to the weight array passed
def portfolio_stats(weight):

    # Convert to array in case list was passed instead.
    weight = np.array(weight)
    port_return = np.sum(log_returns.mean() * weight) * 250
    port_risk = np.sqrt(np.dot(weight.T, np.dot(log_returns.cov() * 250, weight)))
    sharpe = (port_return - risk_free_rate)/port_risk

    return {'return': port_return, 'risk': port_risk, 'sharpe': sharpe}

# Trying to generate random weights
length = len(log_returns.columns)
weight = RandWeights(length)
weight



# Generating Portfolio Statistics
pf_stats = portfolio_stats(weight)
pf_return = pf_stats['return']
pf_risk = pf_stats['risk']
pf_return
pf_risk
sharpe_ratio = pf_stats['sharpe']
sharpe_ratio
```

Listing 7: Function definitions for generating random weights and portfolio statistics

```python
def Monte_Carlo(iterations):
    portfolio_returns = []
    portfolio_risks = []
    for x in range (iterations):
        weight = RandWeights(length)
        pf_stats = portfolio_stats(weight)
        portfolio_returns.append(pf_stats['return'])
        portfolio_risks.append(pf_stats['risk'])

    portfolio_returns = np.array(portfolio_returns)
    portfolio_risks = np.array(portfolio_risks)
    return portfolio_returns, portfolio_risks

portfolio_returns, portfolio_risks = Monte_Carlo(10000)
sharpe = portfolio_returns / portfolio_risks
max_sr_ret = portfolio_returns[sharpe.argmax()] # return corresponding to maximum sharpe ratio
max_sr_vol = portfolio_risks[sharpe.argmax()] # risk corresponding to maximum sharpe ratio
max_sr_ret

plt.figure(figsize=(18,10))
plt.scatter(portfolio_risks, portfolio_returns, c=sharpe, cmap='viridis')
plt.colorbar(label='Sharpe Ratio')
plt.xlabel('Risk')
plt.ylabel('Return')
plt.title('R/R Locus of portfolios: The Efficient Frontier')
plt.scatter(max_sr_vol, max_sr_ret,c='red', s=50) # red dot
plt.show()
print(f"Max Sharpe Ratio = {sharpe.max()}")
print(f"Max Sharpe Ratio Return = {max_sr_ret}")
print(f"Max Sharpe Ratio Volatility = {max_sr_vol}")
```

Listing 8: Function Definition for Montecarlo simulation

```python
def OptimizingWithMinRisk():

    def fun(weight):
        pf_stats = portfolio_stats(weight)
        _risk = pf_stats['risk']
        return _risk


    res = minimize(
      fun,
      RandWeights(length),
      method = 'SLSQP',
      constraints=[
        {'type': 'eq', 'fun': lambda w: np.sum(w) - 1.},
      ],
      bounds=[(0., 1.) for i in range(length)]
    )

    return res

OptimizingWithMinRisk()

target_returns = np.linspace(portfolio_returns.min(), portfolio_returns.max(),20)

minimal_risks = []
for target_return in target_returns:
    optimal = OptimizingWithMinRisk()
    minimal_risks.append(optimal['fun'])

minimal_risks = np.array(minimal_risks)
print(minimal_risks)

plt.figure(figsize=(18,10))

plt.scatter(portfolio_risks, portfolio_returns,
            c = ( portfolio_returns / portfolio_risks),
            marker = 'o')

```

```
# Plotting the efficient frontier
# plt.scatter(minimal_risks,
#             target_returns,
#             c = (target_returns / minimal_risks),
#             marker = 'x')


#Plotting the optimal portfolio that has lowest risk
Optimal_weights_For_Lowest_Risk = OptimizingWithMinRisk().x

plt.plot(portfolio_stats(Optimal_weights_For_Lowest_Risk)['risk'],
         portfolio_stats(Optimal_weights_For_Lowest_Risk)['return'],
         'r*',
         markersize = 25.0, label = "Portfolio corresponding to lowest risk")


plt.xlabel('Portfolio Risk',fontsize = 20)
plt.ylabel('Portfolio Return', fontsize = 20)
plt.legend(prop={'size': 10})
plt.colorbar(label='Sharpe ratio')

w = np.round(Optimal_weights_For_Lowest_Risk,4)
w
len(w)

BEST_PF = pd.Series(w*100, log_returns.columns)
print("% weights of minimum volatility PF")
print (60*"-")

index = w>=0.001
BEST_PF[index].round(3)

w = np.round(Optimal_weights_For_Lowest_Risk,4)
BEST_PF = pd.Series(w*100, stock_list)
print("% weights of minimum volatility PF")
index = w>=0.001
Final_TEFAS_PF = pd.DataFrame(BEST_PF[index].round(3),  columns=['%'] )
Portfolio_Weights = w[index].round(3)
Portfolio_Assets = BEST_PF[index]
Portfolio_Amounts = Portfolio_Weights*100000
Final_TEFAS_PF['Amounts for Capital of 100000 TL'] = Portfolio_Amounts
print(portfolio_stats(Optimal_weights_For_Lowest_Risk))
Final_TEFAS_PF.sort_values(by=['%'], ascending=[False])

def getReturns(startTime, endTime, tickers):
    # pull price data from yahoo -- (list(tickers.keys())) = ['^GSPC','^RUT']
    prices = web.DataReader(BIST10, "yahoo", START_DATE, END_DATE)["Adj Close"]
    prices = prices.dropna()
    returns = prices.pct_change()
    return prices.pct_change()

def compareVariance(startTime, endTime, tickers, weights):
    returns = getReturns(startTime, endTime, tickers)
    tmp = weights * returns
    returns[f"Portfolio w/ weights {Portfolio_Weights}"] = tmp[tmp.columns[0]] + tmp[tmp.columns[1]]
    standardDev = returns.std()
    avgReturns = returns.mean()
    res = pd.concat([avgReturns * 100, standardDev*100], axis=1)
    res.columns = ["Daily Average Return %", "Standard Deviation of Returns %"]
    return res.round(3)

compareVariance( START_DATE, END_DATE, BIST10, Portfolio_Weights )

pyfolio.create_simple_tear_sheet(log_returns.CANTE.dropna())

pyfolio.create_simple_tear_sheet(log_returns.PETUN.dropna())

log_returns["Portfolio"] = (log_returns.CANTE * Portfolio_Weights[0]) + (log_returns.PETUN *
    Portfolio_Weights[1])
log_returns
```

```
pyfolio.create_simple_tear_sheet(log_returns.Portfolio.dropna())
```

Listing 9: Pyfolio portolio performance plots