INDIAN INSTITUTE OF TECHNOLOGY, KANPUR

SURGE-2022

Project-Report

"INVESTMENT MANAGEMENT USING PYTHON"

Submitted by

Ayush Agrawal

SURGE No.: 2230262

Department Of Aerospace Engineering

IIT Kanpur

Under the Guidance of

Prof. SUVENDU SAMANTA

Department of Electrical Engineering,

IIT Kanpur

CERTIFICATE

This is to certify that the project entitled **"INVESTMENT MANAGEMENT USING PYTHON"** submitted by **Ayush Agrawal** (S220078) as a part of Summer Undergraduate Research and Graduate Excellence 2022 offered by the Indian Institute of Technology, Kanpur, is a Bonafede record of the work done by him under my guidance and supervision at the Indian Institute of Technology, Kanpur from 13th May, 2022 to 14th July, 2022.

Dr. Suvendu Samanta
Assistant Professor
Department of Electrical Engineering
Indian Institute of Technology Kanpur

Acknowledgments

I'd want to start by saying how appreciative I am to have had the chance to work with such an insightful professor like Prof. Suvendu Samanta at SURGE, IIT Kanpur.

I value your unwavering encouragement, sir. He sparked my interest in Markowitz's theory, which led to the discovery of exciting new opportunities. I credit a great deal of my project's success to the guidance and counsel he offered me. Despite his motivational qualities, he gave me the freedom to explore my own potential.

My parents encouraged and motivated me during my internship because they saw the value in me gaining experience in the field of research.

I appreciate all of your kind words and votes of confidence.

Ayush Agrawal

Abstract

Investment management has proven to be a time-consuming endeavour, and more efficient computing approaches are needed to increase our returns. There are now more sophisticated ways to invest thanks to advances in technology. Research is required when deciding where to put money, since it is necessary to estimate returns while also projecting hazards. We can use better and more efficient portfolio strategies if we build them using machine learning. I developed my own python module including several measurements and techniques for risk prediction, and utilised python for the method writing itself. At first, I learned how to set up and run jupyter lab, as well as how to apply basic concepts related to returns and derive appropriate calculations. After that, I gained an understanding of the meaning of and how to compute drawdown, volatility, skewness, and kurtosis of a dataset in Python. I gained knowledge of both Gaussian and Non-Gaussian instances, using new and innovative techniques like Historic VaR, Parametric VaR, modified Cornish-Fisher VaR, etc., to forecast the risk of an investment. I bundled all the techniques' calculations into one python package so they could be applied to any dataset. For my own study and education, I made use of the data set made available by the Ken French Data Library.

Keywords: *Drawdown, Cornish-Fisher VaR, Kurtosis*

Contents

Page no.

1. Introduction

When it comes to the stock market, individuals' personalities and risk tolerances vary widely. An investor who enjoys taking chances exists. Huge-risk investors like these are looking for high returns, therefore they're willing to go where the action is. Those who are willing to take risks and speculate heavily while making investments. There is a subset of investors that do not place any value on safety or security, known as "risk neutrality" investors, and who instead apply the same return to all possible levels of progress. Investors with these traits are cautious yet adaptable in their approach to the market. Potentially harmful investors, however, are not ruled out (risk averter). These investors are only willing to risk their capital on projects with modest returns. The average investor isn't looking for a challenge, and they wait until they're older to make investing choices that include risk. Regarding trading on the stock market Through either fundamental or technical research, investors choose stocks with the potential to generate the highest dividends. Many variables, including government policies, political concerns, micro- and macroeconomic phenomena, and others, impact investors' decisions about which stocks to purchase. As a result, stock selection may be harmed by these issues. When making an investment, every sane person would want to get the best possible return. The huge potential rewards from investing in stocks are matched by equally high degrees of uncertainty. Divided into two categories, systematic risk and unsystematic risk, stock market risk may be better understood. Market risk, also known as systematic risk or common risk, is a kind of risk that affects the market as a whole. Therefore, the general direction of the stock market will have an impact on the direction of a few specific stock prices. Investors have little say over this kind of systemic risk, and it cannot be mitigated through diversification. On the other hand, the nonsystematic risk, also known as the particular risk, is the possibility that the stock price of a firm would be negatively impacted by a single occurrence. In order to mitigate the systemic risk associated with stock investments, investors should construct diversified stock portfolios. There is a trade-off between the potential rewards of investing in the stock market and the inherent dangers of doing so. Therefore, it is incumbent upon investors to think sensibly and to pay close attention to the best means of maximizing return while minimizing risk. So that those who invest may reap the greatest possible rewards, it's important that their money be put to good use. This is where the investor's due diligence comes in. Investors risk losing money if they get an overwhelming amount of data that causes them to become confused. The investor is unable to get the data she or he needs to conduct a thorough analysis of the portfolio.

There are several stock market investing options available, each representing a different sector of the economy. This facilitates investors' ability to diversify their portfolios, a crucial step in achieving optimal portfolio formation, the goal of which is to minimise loss by spreading capital over several, potentially profitable investments. The Markowitz portfolio theory is only one of many possible frameworks for constructing an optimal investment portfolio. The most efficient portfolio is the one that maximises expected return while minimizing risk. Choosing a target rate of anticipated return and then reducing the portfolio's risk, or choosing a target level of risk and then optimising the portfolio's expected return, are both valid strategies for identifying an optimal portfolio. Portfolio analysis is used by investors and stockholders to determine which stocks should be included in a portfolio in order to achieve the desired degree of risk and return. The goal of this research is to apply the Markowitz portfolio theory framework to the information provided in ata in order to identify the best stock investment strategy.

2. Method

Investment

Putting money into something with the expectation of a return is an investment. A current investment in diverse assets with the expectation of future revenue is what is meant by the term "investment."(Warini, 2009) The Portfolio Approach Optimal portfolio formation may be determined with the help of portfolio theory, which is a related theory of projected portfolio returns and tolerable portfolio risk levels. This portfolio theory demonstrates the relationship that should occur between the return and the risk of securities if investors form a portfolio that fits the portfolio theory, and it is related to capital market theory on the basis of the influence of investors' decisions on the price of securities. Assuming a formal framework exists to assess both risk and return, portfolio theory asserts that both must be taken into account when constructing a portfolio. For portfolio theory to work, it must first assume that the rate of return on future impacts can be determined, and only then can the risk associated with fluctuations in the distribution of returns be calculated. The rate of return expected on an investment or loan is higher when the risk involved is higher. A diversified portfolio is created to reduce exposure to risk without losing potential reward. A diversified portfolio reduces exposure to potential loss in a few key holdings. Portfolio diversification, also known as deployment, entails adding assets from across the board. Such as shares of stock, bonds, bank deposits, real estate, and so forth. Concentrating on a single asset class, like stocks, is another option for portfolio diversification. Due to the vast availability of securities, however, investors may have difficulty in constructing a suitable portfolio. When considering the total number of possible portfolios, consider the total number of possible pairings of securities. That portfolio is efficient if it maximises expected returns while maintaining a constant level of risk, or if it minimises risk while maintaining the same level of expected return. Investors' optimum portfolios are the ones they choose from among many effective ones that best suit their personalities. Assuming that all investors are risk-averse allows for the construction of an optimum portfolio. It's the same to try to maximise your return per unit of risk you take. An efficient portfolio, such as 178 JAN HORAS VERYADY PURBA AND BAMBANG PAMUNGKAS, is decent but not ideal. Optimal portfolios are the finest possible option. An efficient portfolio is one that is optimum. The most effective portfolio may not always be the best one. If you have access to risk-free assets, you may use them to calculate the best portfolio. There is no uncertainty in the rate of return anticipated from a risk-free investment. For traditional shares, the risk-free rate in a single index computation is often the BI rate or the interest rate of Bank Indonesia Certificates (SBI). The term "anticipated return" refers to the profit that investors anticipate making in the future. Speculators stand to get a return on their investment, but the precise amount of which is now unknown. There is a positive correlation between expected returns and the degree of risk taken. Generally speaking, the higher the perceived risk of an asset, the higher the projected return, and vice versa. The square of the standard deviation, known as the variation, is one way to quantify the degree of risk. This metric assesses the danger posed by the possibility that individual values may differ significantly from the mean. The standard deviation, or variance, of the value of individual investment returns in a portfolio is another way to quantify its exposure to risk.

Markowitz Method

Markowitz's approach to portfolio construction, often known as the Mean-Variant Model, prioritises maximising return (mean) expectations while minimising uncertainty/risk (variance). This suggests that the optimal portfolio selection strategy takes into account the investor's tolerance for risk while also taking into account the desired rate of return. All investors may benefit from the market data available when using this technique to construct a portfolio. Since Markowitz's theory is used to build the basis for the portfolio, the study spans just a single period, the investor calculates the value of his or her exposure and the risk of the portfolio, no loans or risk-free savings are included, and neither are transaction costs (Hartono, 2010 : 312)

Methodology

Systematically, the phase of data analysis is done as follows:

1. Variance of Shares:

$$\sigma_i^2 = \Sigma_{t=1}^{n} \frac{[Rit - E(Ri)]^2}{n}$$

Note:
σ_i^2 = Variance of return of share i
Rit = Return of share i in periode t
E(Ri) = Expected return share i
n = Time periode
So with the above variance formula, we can calculate the standard deviation, by:

$$\sigma_i = \sqrt{\sigma_i^2}$$

Note:
σ_i^2 = Variance of return of share i
σ_i = Standard deviation
2. Search for covariance by entering the standard deviation that has been obtained by the formula:
Note:

$$\mathrm{COV}_{ij} = \frac{\Sigma\ (R_{it} - E(R_i)(R_{jt} - E(R_j)}{n}$$

COV_{ij} = Covariance variables i and j
Rit = Return of stock i in period t
E(Ri) = Expected return stock i
Rjt = Return of stock j at period t
E(Rj) = Expected return of stock j
n = Time period
3. The correlation coefficient is calculated by dividing the stock covariance i and j with the result of standard deviation multiplication of shares i and j, by the formula:

$$r_{ij} = \frac{COV_{ij}}{\sigma_i \sigma_j}$$

Note:
r_{ij} = The correlation coefficient of variables i and j
$\mathrm{COV}_{\mathrm{ij}}$ = Covariance variables i and j
σ_i = Standard deviation i
σ_{j} = Standard deviation j
4. Make a simulation and assume the proportion of funds to the stocks that have been selected with percentage as a unit With the condition : $W_1 + W_2 = 1$ or 100%
5. Next look for a portfolio risk that consists of two securities, the formula used is :

$$\sigma_p = \sqrt{W_i^2 \cdot \sigma_i^2 + W_j^2 \sigma_j^2 + 2 \cdot W_i \cdot W_{j\cdot} (r_{ij}) \cdot \sigma_i \cdot \sigma_j}$$

Note :
σ_{p} = Standard deviation portfolio
$\mathrm{W_i}$ = Weight allocation on stock i
$\mathrm{W_j}$ = Weight allocation on stock j
$\mathrm{r_{ij}}$ = The correlation coefficient of variables ij
σ_{i} = Standard deviation i
$\sigma_{\mathbf{j}}$ = Standard deviation j

Efficient Frontier

We are well aware that the projected returns and risks associated with each asset in a portfolio will vary. Many other asset pairings exist, each with the potential to provide significant returns at a certain amount of risk.

Similarly, there may be a number of portfolios that, relative to a target projected return, have the lowest possible risk.

The efficient frontier is a two-dimensional graph, with "returns" shown along the vertical axis and "volatility" along the horizontal. The optimum portfolios that maximise anticipated return at a given risk level or minimise risk at a given expected return are shown.

Outside the efficient frontier, portfolios are sub-optimal because they either do not give enough return for the amount of risk or have a greater risk for the set rate of return.

Covariance

Covariance is a statistical measure of the degree to which two asset returns are correlated with one another.

When the covariance between two assets is positive, it indicates that their return rates are positively correlated. By combining assets with a negative correlation, portfolio risk and volatility may be mitigated.

The.cov() method allows us to determine the covariance.

Correlation

The degree to which the prices of two assets move in tandem is quantified by a statistic known as correlation in the world of finance and investments. When utilised in high-level portfolio management, correlations are often expressed as the correlation coefficient, a number that must be between -1.0 and +1.0.

A useful way to conceptualise correlation is as a scaled form of covariance, with values only allowed to range from -1 to +1.

If Asset A and Asset B have a correlation of -1, then as Asset A rises, Asset B falls.

Specifically, if Asset A has a correlation of 1 with Asset B, then each time Asset A goes up, Asset B goes up as well, and vice versa.

No relationship exists between Assets A and B if their correlation is zero.

The.corr() method is used for the calculation.

Sharpe Ratio

The ratio is the average return earned in excess of the risk-free rate per unit of volatility or total risk. Volatility is a measure of the price fluctuations of an asset or portfolio.

The risk-free rate of return is the return on an investment with zero risk, meaning it’s the return investors could expect for taking no risk.

The optimal risky portfolio is the one with the highest Sharpe ratio. The formula for this ratio is:

$$Sharpe\ Ratio = \frac{R_p - R_f}{\sigma_p}$$

where:

R_p = return of portfolio

R_f = risk-free rate

σ_p = standard deviation of the portfolio

3. Result

Implementing various methods through python code

During the course of this project, we developed a python module to easily implement the theory and predict the best outcome.

Here we can present some of the code snippet for different cases and plotting the efficient frontier

Let's start by loading the returns and generating the expected returns vector and the covariance matrix

Assume we have some weights, and let's try and compute the returns and volatility of a portfolio, given a set of weights, returns, and a covariance matrix using Python.

```
def portfolio_return(weights, returns):

    """

    Computes the return on a portfolio from constituent returns and
weights

    weights are a numpy array or Nx1 matrix and returns are a numpy
array or Nx1 matrix

    """

    return weights.T @ returns
```

```
The volatility is just as easy in matrix form:
```

```
def portfolio_vol(weights, covmat):

    """

    Computes the vol of a portfolio from a covariance matrix and
constituent weights

    weights are a numpy array or N x 1 maxtrix and covmat is an N x N
matrix

    """

    return (weights.T @ covmat @ weights)**0.5
```

In this study, purposively selected 4 industries from Ken French 30 Industry Portfolios Value Weighted Monthly Returns from 1996 until 2000 which are Food, Beer, Smoke, Coal. To find the annualize returns of these we wrote below function:

```
def annualize_rets(r, periods_per_year):

    """

    Annualizes a set of returns

    We should infer the periods per year

    but that is currently left as an exercise

    to the reader :-)

    """

    compounded_growth = (1+r).prod()

    n_periods = r.shape[0]

    return compounded_growth**(periods_per_year/n_periods)-1

er = annualize_rets(ind["1996":"2000"], 12)

l = ["Food", "Beer", "Smoke", "Coal"]
```

Annualized returns of the above four industries are:

```
er[l]
```

Food 0.116799
Fin 0.223371
Games 0.068212
Coal 0.414689
dtype: float64

Covariance matrix:

```
cov.loc[l,l]
```

	Food	Fin	Games	Coal
Food	0.002609	0.002132	0.000846	0.000027
Fin	0.002132	0.003982	0.002416	0.002946
Games	0.000846	0.002416	0.003773	0.001888
Coal	0.000027	0.002946	0.001888	0.018641

Calculating the portfolio return and volatility:

```
def portfolio_return(weights, returns):

    """

    Computes the return on a portfolio from constituent returns and
weights

    weights are a numpy array or Nx1 matrix and returns are a numpy
array or Nx1 matrix

    """

    return weights.T @ returns




def portfolio_vol(weights, covmat):

    """

    Computes the vol of a portfolio from a covariance matrix and
constituent weights

    weights are a numpy array or N x 1 maxtrix and covmat is an N x N
matrix

    """

    return (weights.T @ covmat @ weights)**0.5




def plot_ef2(n_points, er, cov):

    """

    Plots the 2-asset efficient frontier

    """

    if er.shape[0] != 2 or er.shape[0] != 2:

        raise ValueError("plot_ef2 can only plot 2-asset frontiers")

    weights = [np.array([w, 1-w]) for w in np.linspace(0, 1, n_points)]

    rets = [portfolio_return(w, er) for w in weights]

    vols = [portfolio_vol(w, cov) for w in weights]

    ef = pd.DataFrame({

        "Returns": rets,
```

```
        "Volatility": vols
    })
    return ef.plot.line(x="Volatility", y="Returns", style=".-")
```

For the above given data, we can calculate the return and volatility as below:

```
ew = np.repeat(0.25, 4)
portfolio_return(ew, er[l])
```

0.19511097196038385

```
portfolio_vol(ew, cov.loc[l,l])
```

0.055059195776437045

The 2-Asset Case

In the case of 2 assets, the problem is somewhat simplified, since the weight of the second asset is 1-the weight of the first asset.

Let's write a function that draws the efficient frontier for a simple 2 asset case.

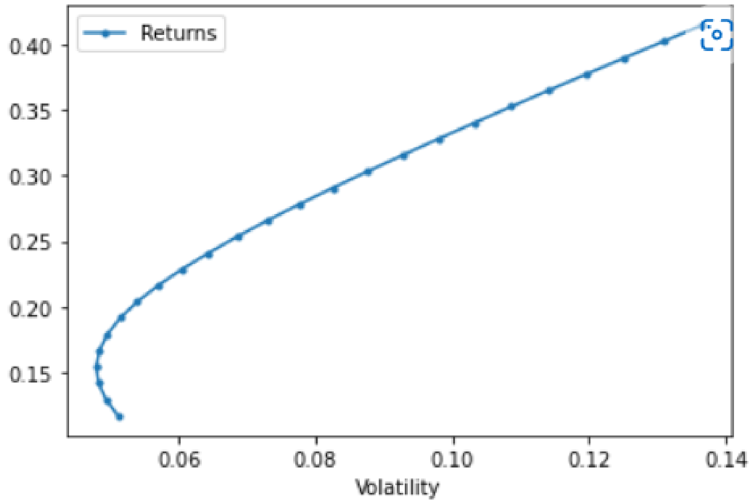
Start by generating a sequence of weights in a list of tuples. Python makes it easy to generate a list by using something called a *list comprehension* ... which you can think of as an efficient way to generate a list of values instead of writing a for loop.

```
n_points = 20
weights = [np.array([w, 1-w]) for w in np.linspace(0, 1, n_points)]
```

Let us consider two assets “Food” & “Coal” in the same data.

```
l = ["Food", "Coal"]
plot_ef2(25, er[l].values, cov.loc[l,l])
```

```
<AxesSubplot:xlabel='Volatility'>
```



For more than two assets, we need to find the optimal weights, minimizing the volatility for a given level of return. For this purpose, we can implement the below function:

```
from scipy.optimize import minimize

def minimize_vol(target_return, er, cov):

  """

  Returns the optimal weights that achieve the target return

  given a set of expected returns and a covariance matrix

  """

  n = er.shape[0]

  init_guess = np.repeat(1/n, n)

  bounds = ((0.0, 1.0),) * n # an N-tuple of 2-tuples!

  # construct the constraints

  weights_sum_to_1 = {'type': 'eq',

            'fun': lambda weights: np.sum(weights) - 1

  }

  return_is_target = {'type': 'eq',

            'args': (er,),
```

```
            'fun': lambda weights, er: target_return - erk.portfolio_return(weights,er)
    }
    weights = minimize(erk.portfolio_vol, init_guess,
                       args=(cov,), method='SLSQP',
                       options={'disp': False},
                       constraints=(weights_sum_to_1,return_is_target),
                       bounds=bounds)
    return weights.x
```

Now that we can find the weights to minimize the vol given a target return, we can plot the efficient frontier by dividing up the range from the highest to the lowest possible return into a grid, and finding the portfolio that targets the minimum volatility given a particular targeted rate of return.

Add these:

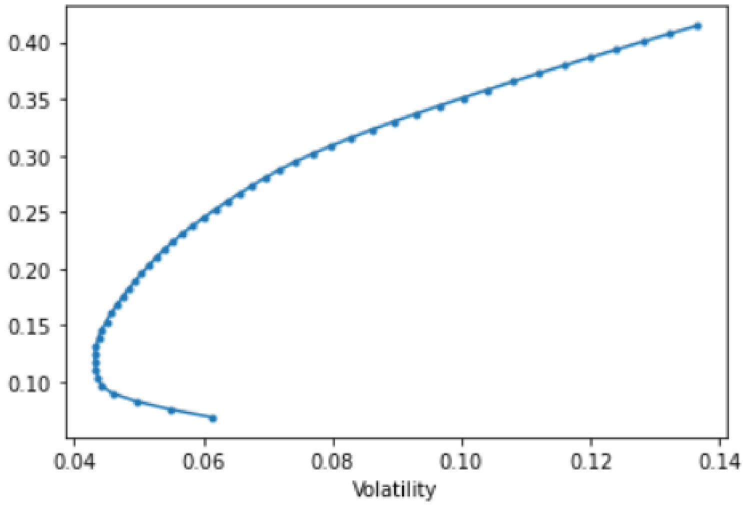
```
def optimal_weights(n_points, er, cov):
target_rs = np.linspace(er.min(), er.max(), n_points)
    weights = [minimize_vol(target_return, er, cov) for target_return in target_rs]
    return weights
def plot_ef(n_points, er, cov):
    """
    Plots the multi-asset efficient frontier
    """
    weights = optimal_weights(n_points, er, cov) # not yet implemented!
    rets = [portfolio_return(w, er) for w in weights]
    vols = [portfolio_vol(w, cov) for w in weights]
    ef = pd.DataFrame({
```

```
        "Returns": rets,
        "Volatility": vols
    })
    return ef.plot.line(x="Volatility", y="Returns", style='.-')
```

Implementing the above function for the 4 assets we discueed above:

```
l = ["Food", "Fin", "Games", "Coal"]
plot_ef(50, er[l], cov.loc[l,l])
```

We get our efficient frontier as below:



4. Conclusion and Further Research

When it comes to solving the portfolio optimization issue, the Markowitz modern portfolio theory provides a fairly simple model. Unfortunately, the Markowitz technique seldom delivers on its promise. The issue is that it is unusual to have complete knowledge of future expected returns and expected covariance. Inevitably, our estimations will have some degree of estimation error, and as we'll see, the technique is quite sensitive to these flaws, which tend to be exacerbated in the portfolio. While MPT has been around for a while, it has been superseded by post-modern portfolio theory, which adopts asymmetric measures of risk and other non-normal distributions to expand MPT. Some of these difficulties are alleviated, while others are not at all. For the last three decades, GP has been widely used to solve Portfolio Selection issues. Several key points are discussed briefly throughout the project. Multiple objectives, including the portfolio's return, risk, liquidity, and expenditure ratio, may be included into a GP model for PS. To that end, this project's study focused only on making projections based on historical data using a regression model. However, in practise, stock prices tend to fluctuate randomly and without pattern. It exhibits a statistical behaviour known as geometric Brownian motion. Therefore, the application of stochastic programming to the issue of portfolio optimization became possible. It's a more practical strategy. Some or all of the variables and/or constraints in stochastic programming are uncertain, but their distributions are known. By setting a threshold value 1, and then translating the restriction into its respective probability value being bigger than the threshold value, we are able to transform this into a deterministic programming issue.

5. References:

[1] Portfolio Management (investopedia.com)

[2] Kenneth R. French - Data Library (dartmouth.edu)

[3] Purba, Jan Horas & Pamungkas, Bambang. (2018). PORTFOLIO ANALYSIS WITH MARKOWITZ METHOD STUDY ON STOCKS LQ-45 IN INDONESIA. 5. 175-185.

[4] Portfolio Optimization with Python using Efficient Frontier with Practical Examples - Machine Learning Plus