

Financial Stock Recommendation System

A Project Report Submitted in Partial Fulfillment of the Requirements for Award
of
the Degree of Bachelor of Technology in Information and Communication
Technology

Submitted by

Yash Sahitya
Roll No. 18BIT121

Under the Supervision and Guidance of
Dr. Jigarkumar Shah
Assistant Professor
Pandit Deendayal Energy University

Submitted to
Department of Information and Communication Technology
School of Technology
Pandit Deendayal Energy University (PDEU)
Gandhinagar, INDIA, 382007

Contents

Contents	i
1 Introduction	1
1.1 Problem statement	1
1.2 Motivation	1
1.3 Objective	1
1.4 Roadmap	2
2 Weekly progress from 20/08/21 to 27/08/21	3
2.1 Literature review	3
3 Weekly progress from 28/08/21 to 03/09/21	6
3.1 Dataset Description	6
3.1.1 Content	6
3.2 Acknowledgements	7
3.3 yfinance : Yahoo Finance Library	7
4 Weekly progress from 04/09/21 to 11/09/21	8
4.1 Portfolio Management	8
4.1.1 Modern Portfolio Theory	8
4.2 Summary	10
5 Weekly progress from 12/09/21 to 24/09/21	11
5.1 Markowitz Portfolio Optimization and Efficient Frontier	11
5.2 Efficient Frontier	12
5.2.1 SHARPE RATIO	13
5.2.2 Minimum Volatility	14
5.3 Capital Asset Pricing Model (CAPM)	15
5.3.1 The CAPM and the Efficient Frontier	15

6	Weekly progress from 25/09/21 to 1/10/21	17
6.0.1	Assumptions of Modern Portfolio Theory	17
6.0.2	Advantages of the Modern Portfolio Theory (MPT)	18
6.0.3	Disadvantages of the Modern Portfolio Theory (MPT)	18
6.0.4	Conclusion	19

Chapter 1

Introduction

1.1 Problem statement

Improvising the existing traditional stock recommendation system using portfolio optimization and ad hoc machine learning techniques while also validating the proposed results through backtesting.

1.2 Motivation

Recent advances in the Stock Recommendation Systems, trading technologies by using various Machine Learning and AI techniques to generate a reliable and accurate Stock Portfolio Recommendation system with minimum risk has been the main motivation behind selecting this particular topic. This project has encouraged our team to learn and understand financial terminologies, investment analysis and portfolio management, modern portfolio theory (MPT) while also requiring us to have in-depth domain knowledge and learn about the current AI techniques applied in the domain.

1.3 Objective

To build a ‘Stock Portfolio Recommendation’ web application and backtest the generated results to find out absolute returns of recommended stocks, Compound Annual Growth Rate (CAGR) and Portfolio Growth.

1.4 Roadmap

- To gain detailed domain knowledge required for better grasp and understanding for the project.
- Reviewing, analysing and inspecting the latest research work in the field to understand how to use and implement AI techniques in the project.
- Coding in Jupyter Notebook with reference to our proposed idea to conduct quantitative analysis and to test the AI algorithms for the same.
- Writing clean and reproducible code for production and deployment of the web application.
- To backtest the recommended portfolio i.e the generated results to verify and validate the proposed results.
- To study and examine if any further improvements can be made regarding the performance of the model by making changes in the required part of the pipeline to produce a more accurate and reliable ‘Stock Portfolio Recommendation System’.

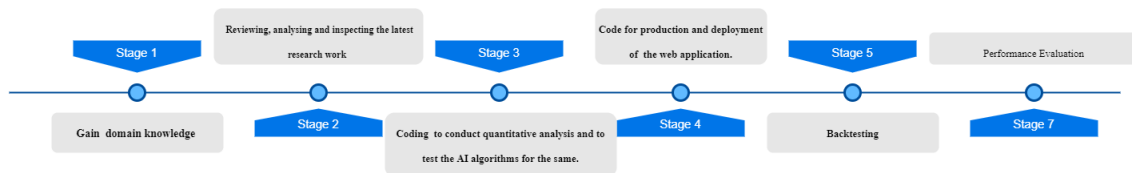


Figure 1.1: Roadmap

Chapter 2

Weekly progress from 20/08/21 to 27/08/21

2.1 Literature review

The field of stock trading investment has been pursued for a long time with many research papers and works for the same by using various kinds of models. This project only presents some relative researches. Huang [4] developed a model for stock selection by using SVR and genetic algorithms (GAs). This model applied SVR to predict future return of each stock whereas GA was employed to optimize model parameters and input features. Back then, the top-ranked stocks were equally weighted to build a portfolio. Empirical results showed that the investment performance of the proposed model performed better than the benchmarks. Krauss et al.[5] in 2017 analyzed the effectiveness of DNN, gradient-boosted trees, Random Forest, and several ensembles of these methods in the context of statistical arbitrage. Each model was trained on lagged returns of all stocks of the SP 500 after elimination of survivor bias. It was concluded from the experiments that the simple equal-weighted ensemble method could generate satisfying results. Lee and Yoo[6] compared three kinds of recurrent neural network for stock return prediction, i.e., recurrent neural network, gated recurrent unit and long short-term memory (LSTM) neural network. The experimental results presented showed that the LSTM neural network performed the best of these three models. Also, they built predictive threshold-based portfolios based on the predictive results of LSTM neural networks. This model was more data-driven than existing models in designing the portfolios.

Experimental results showed that this portfolio earned a promising return. Fischer and Krauss [3] deployed LSTM neural networks for predicting directional movements of the constituent stocks of the SP 500 from 1992 until 2015. They found that LSTM neural networks based portfolios surpassed the memory-free classifica-

tion based portfolio models, i.e. Random Forest, Deep Neural Network and Logistic regression. The common shortcoming of the models is that these models only apply simple methods to build their portfolio, such as equal weighted and threshold-based methods. These portfolio construction methods do not analyze the risk of each stock, which unbalances the portfolio's expected return and risk. Lin, Huang, Gen, and Tzeng[7] proposed a dynamic portfolio optimization model. This model used the Elman neural network to predict future stock return, and then applied a covariance matrix to measure the risk involved. Experimental results presented that this model outperformed vector autoregression model in dynamic portfolio selection models. Alizadeh, Rada, Jolai, and Fotoohi [1] developed a portfolio optimization model by using adaptive neuro-fuzzy inference system for portfolio return prediction and variance index for risk assessment. Experimental results showed that this portfolio optimization model performed better than the mean-variance model, neural network and Sugeno-Yasukawa method. This research informed us that combining artificial intelligence techniques with modern portfolio optimization could generate better performance than each single model for trading investment. Deng and Min [2] applied a linear regression model with ten variables to select stocks from US and global equities, and built a portfolio based on MV model with some practical constraints of risk tolerance, tracking error and turnover. They found that the risk adjusted return of the proposed model of the global equity universe performed better than the domestic equity universe, and the portfolio return increased with the systematic tracking error and risk tolerance. Paiva et al.[8] proposed a unique decision-making model for day trading investments on the stock market, which was developed using a fusion approach of SVM and MV model for portfolio selection. The proposed model was compared with two other models, i.e., SVM+ 1/N and Random+ MV. The experimental evaluation was based on assets from Ibovespa stock market, which showed the proposed model performed the best. Wang et al.[10] developed a portfolio model by using LSTM neural network for stock selection and MV for portfolio optimization. In this model, LSTM neural network first selects k stocks from the total stock set, then the chosen k stocks are used to build the MV portfolio model. They compared LSTM neural network with SVM, RF and ARIMA model in the stock selection process, then used MV for portfolio optimization. Experiments' results showed that their proposed model outperformed the others. Ta et al.[9] built portfolios by using LSTM neural network and three portfolio optimization techniques, i.e., equal weighted method, Monte Carlo simulation and MV model. Also, they applied linear regression and SVM as comparisons in the stock selection process. Experimental results showed that the LSTM neural network had higher predictive accuracy than linear regression and SVM, and its constructed portfolios outperformed the others. These models apply different methods for stock selection, then build portfolio optimization models with selected stocks for trading investment. These methods show us a promising direction to build portfolio models

in practice. However, classic portfolio optimization models are often inappropriate for short term practical investment. Thus, it is important to explore a more efficient approach to combine return predictive results with portfolio optimization models. Hence in our project, we are trying to overcome the shortcomings of traditional portfolio optimization system with help of machine learning techniques including clustering followed by LSTM and Reinforcement Learning methods. Since machine learning and deep learning models have shown overwhelming superiority than time series models. In our project we also analyze technical parameters for generating profit yielding portfolio optimization system.

Chapter 3

Weekly progress from 28/08/21 to 03/09/21

3.1 Dataset Description

Stock market data is widely analyzed for educational, business and personal interests. The National Stock Exchange of India Limited (NSE) is the leading stock exchange of India, located in Mumbai. The NIFTY 50 index is National Stock Exchange of India's benchmark broad based stock market index for the Indian equity market. Apart from NIFTY 50 index, there are also other indices like NIFTY Next 50, Nifty Midcap 150 etc. Exploring these indices may help in taking investment decisions.

3.1.1 Content

The data is the price history and trading volumes of the fifty stocks in the index NIFTY 50 from NSE (National Stock Exchange) India. All datasets are at a day-level with pricing and trading values split across .csv files for each stock along with a metadata file with some macro-information about the stocks itself. The data spans from 1st January, 2000 to 30th April, 2021. This dataset has day level information on major NIFTY indices starting from 01 January 2000.

Each file represents an index and has the following columns

Since new stock market data is generated and made available every day, in order to have the latest and most useful information, the dataset will be updated once a month.

Date	date of observation
Open	open value of the index on that day
High	highest value of the index on that day
Low	lowest value of the index on that day
Close	closing value of the index on that day
Volume	volume of transaction
Turnover	turn over
P/E	price to earnings ratio
P/B	price to book value
Div Yield	dividend yield

3.2 Acknowledgements

NSE India: <https://www.nseindia.com/>

Thanks to NSE for providing all the data publicly.

Various machine learning techniques can be applied and explored to stock market data, especially for trading algorithms and learning time series models.

3.3 yfinance : Yahoo Finance Library

yfinance package not just helps to access the share-price details, it also provides myriad of other financial and non-financial data pertaining to all companies listed in the United States.

yfinance can also be used to obtain data related of bonds, stock indices (SP 500, Nasdaq, Dow 30 etc.), Commodities (Crude Oil , Gold, Silver etc.), Currencies (EUR/USD, GBP/USD, USD/JPY etc.), International Markets (FTSE 100, Nikkei 225 etc.) and even bitcoin and crypto-currency index.

Chapter 4

Weekly progress from 04/09/21 to 11/09/21

4.1 Portfolio Management

4.1.1 Modern Portfolio Theory

Modern portfolio theory (MPT) is a theory on how risk-averse investors can construct portfolios to maximize expected return based on a given level of market risk. Harry Markowitz pioneered this theory in his paper "Portfolio Selection," which was published in the Journal of Finance in 1952.

Modern portfolio theory argues that an investment's risk and return characteristics should not be viewed alone, but should be evaluated by how the investment affects the overall portfolio's risk and return. MPT shows that an investor can construct a portfolio of multiple assets that will maximize returns for a given level of risk. Likewise, given a desired level of expected return, an investor can construct a portfolio with the lowest possible risk. Based on statistical measures such as variance and correlation, an individual investment's performance is less important than how it impacts the entire portfolio. MPT assumes that investors are risk-averse, meaning they prefer a less risky portfolio to a riskier one for a given level of return. As a practical matter, risk aversion implies that most people should invest in multiple asset classes.

The expected return of the portfolio is calculated as a weighted sum of the individual assets' returns. If a portfolio contained four equally weighted assets with expected returns of 4, 6, 10, and 14%, the portfolio's expected return would be: $(4\% \times 25\%) + (6\% \times 25\%) + (10\% \times 25\%) + (14\% \times 25\%) = 8.5\%$

Calculating Portfolio Expected Return

To calculate the expected return of a portfolio, the investor needs to know the expected return of each of the securities in their portfolio as well as the overall weight of each security in the portfolio.

Let's say your portfolio contains three securities. The equation for its expected return is as follows:

$$\text{Expected Return} = W_A \times R_A + W_B \times R_B + W_C \times R_C$$

where:

W_A = Weight of security A

R_A = Expected return of security A

W_B = Weight of security B

R_B = Expected return of security B

W_C = Weight of security C

R_C = Expected return of security C

Python Function for Calculating Portfolio expected Return:

```
def portfolio_return(weights, returns):  
    """  
    Weights -> Returns  
    """  
    return weights.T @ returns
```

Calculating Portfolio Volatility

The method to compute the risk of a portfolio is written below and subsequently we explain and give the mathematical formulation for each of the step :

- Calculate the covariance matrix on the returns data
- Annualize the covariance by multiplying by 252
- Compute the portfolio variance by multiplying it with weight vectors
- Compute the square root of the variance calculated above to get the standard deviation. This standard deviation is called as the Volatility of the portfolio

The formula to compute the covariance and annualizing it is :

$$\text{covariance} = \text{returns.covO} * 252$$

since there are 252 trading days, we multiply the covariance by 252 in order to annualize it. The portfolio volatility , to be calculated in step 4 above, depends on the weights of the assets in the portfolio and is defined as :

$$\sigma_{portfolio} = \sqrt{(weights)^T \cdot Cov_{portfolio} \cdot (weights)}$$

where, Covportfolio is the covariance matrix of the portfolio.

weights is the vector of weights allocated to each asset in the portfolio.

Python Function for calculating Portfolio Volatility:

```
def portfolio_vol(weights, covmat):
    """
    Weights -> Vol
    """
    return (weights.T @ covmat @ weights)**0.5
```

4.2 Summary

1. *Expected return*

$E(R_p) = \sum_i w_i E(R_i)$ where R_p is the return on the portfolio, is the weighting of component asset i in the portfolio). R_i is the return on asset i and w_i (that is, the proportion of asset "i")

2. *Portfolio Return Variance*

$$\sigma_p^2 = \sum_i w_i^2 \sigma_i^2 + \sum_i \sum_{j \neq i} w_i w_j \sigma_i \sigma_j \rho_{ij}$$

where ρ_{ij} is the correlation coefficient between the returns on assets i and j . Alternatively the expression can be written as:

$$\sigma_p^2 = \sum_i \sum_j w_i w_j \sigma_i \sigma_j \rho_{ij}$$

3. *Portfolio Return Volatility*

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

Chapter 5

Weekly progress from 12/09/21 to 24/09/21

5.1 Markowitz Portfolio Optimization and Efficient Frontier

The portfolio optimization plays an important role in determining the portfolio strategies for investors. What investors hope to achieve through a portfolio optimization is to maximize portfolio return and minimize portfolio risk. Since the return changes based on risk investors have to balance the contradiction between risk and return for their investment. There is therefore no single optimal portfolio to satisfy all investors. The optimal portfolio is determined by the preferences of the investor's risk and return.

Our goal is to choose the weights for each asset in our portfolio such that we maximize the expected return given a level of risk. Mathematically, the objective function can be defined as shown below:

$$\begin{aligned} &\text{maximize } E[(weights)^T \cdot returns] \\ &\text{subject to } (weights)^T \cdot cov_{portfolio} \cdot weights = \sigma^2 \end{aligned}$$

Another way to look at the Efficient Frontier and the objective function is that we can minimize the risk given that the Expected return is at least greater than is given value. Mathematically this objective function can be written as:

$$\begin{aligned} &\text{minimize } (weights)^T \cdot cov_{portfolio} \cdot weights \\ &\text{subject to } (weights)^T \cdot \mu \geq \mu_{target} \\ &(weights)^T \cdot 1 = 1 \\ &weights_i \geq 0 \end{aligned}$$

the first line represents that the objective is to minimize the portfolio variance, i.e. in turn the portfolio volatility and therefore minimizing the risk. The subject

to constraints implies that the returns have to be greater than a particular target return, all the weights should sum to 1 and weights should not be negative

5.2 Efficient Frontier

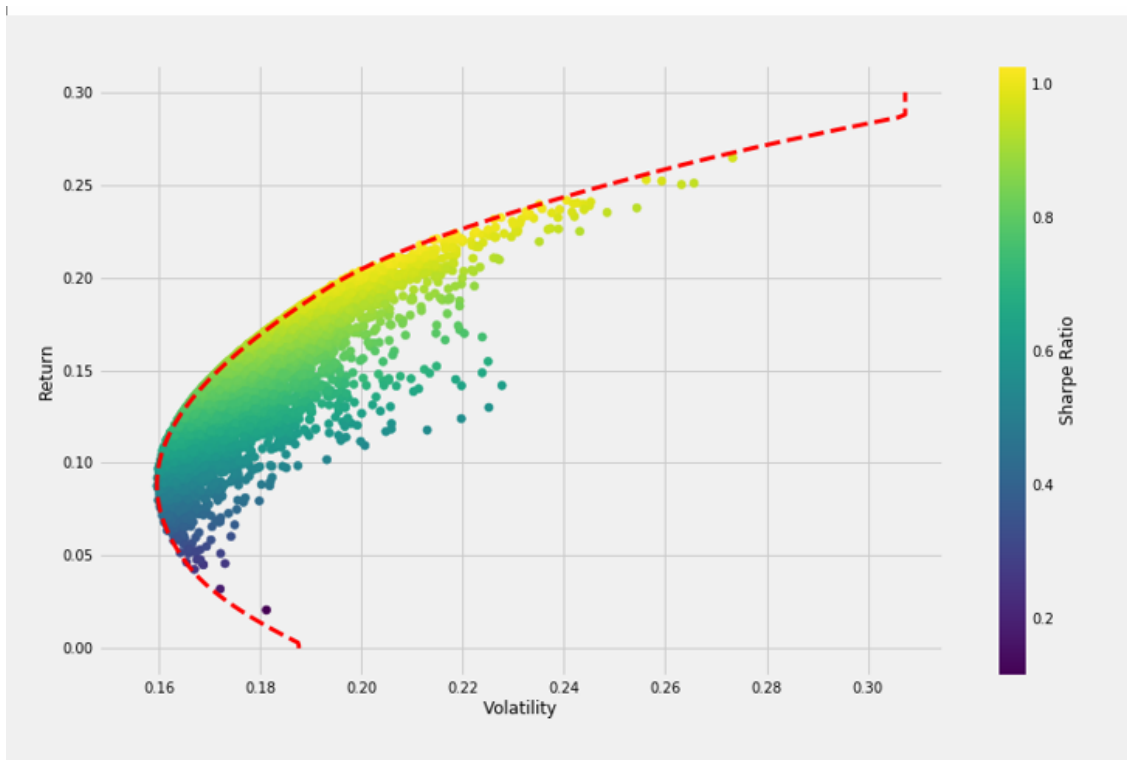


Figure 5.1: Efficient Frontier

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. Portfolios that cluster to the right of the efficient frontier are sub-optimal because they have a higher level of risk for the defined rate of return.

The efficient frontier rates portfolios (investments) on a scale of return (y-axis) versus risk (x-axis). Compound Annual Growth Rate (CAGR) of an investment is commonly used as the return component while standard deviation (annualized) depicts the risk metric.

The efficient frontier graphically represents portfolios that maximize returns for the risk assumed. Returns are dependent on the investment combinations that

make up the portfolio. The standard deviation of a security is synonymous with risk. The less synchronized the securities (lower covariance), the lower the standard deviation. If this mix of optimizing the return versus risk paradigm is successful then that portfolio should line up along the efficient frontier line.

A key finding of the concept was the benefit of diversification resulting from the curvature of the efficient frontier. The curvature is integral in revealing how diversification improves the portfolio's risk / reward profile. It also reveals that there is a diminishing marginal return to risk. Optimal portfolios that comprise the efficient frontier tend to have a higher degree of diversification than the sub-optimal ones, which are typically less diversified.

Python Function for Efficient Frontier:

```
def plot_ef(n_points , er , cov):
    """
    Plots the multi-asset efficient frontier
    """
    weights = optimal_weights(n_points , er , cov)
    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='.-' , legend=)
```

After obtaining the Efficient Frontier, the next step is to find the optimal weights. We can optimize the using multiple methods as written below:

5.2.1 SHARPE RATIO

The Sharpe-ratio is the average return earned in excess of the risk-free rate per unit of volatility or total risk. The formula used to calculate Sharpe-ratio is given below:

$$SharpeRatio = (R_p - R_f) / SD_p$$

where,

R_p is the return of portfolio

R_f is the risk free rate

SD_p is the standard deviation of the portfolio's returns

Python Function for Sharpe-ratio:

```
def sharpe_ratio(r, riskfree_rate, periods_per_year):  
    """  
    Computes the annualized sharpe ratio of a set of returns  
    """  
    # convert the annual riskfree rate to per period  
    rf_per_period = (1+riskfree_rate)**(1/periods_per_year)-1  
    excess_ret = r - rf_per_period  
    ann_ex_ret = annualize_rets(excess_ret, periods_per_year)  
    ann_vol = annualize_vol(r, periods_per_year)  
    return ann_ex_ret/ann_vol
```

5.2.2 Minimum Volatility

One of the most effective smart beta strategies for achieving a reduction of risk are known as 'minimum volatility' strategies, which are specifically designed with this goal in mind, seeking to deliver market-like returns with less volatility by targeting lower volatility stocks

Python Function for Minimum Volatility:

```
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 - portfolio_return(weights, er)  
                        }  
    weights = minimize(portfolio_vol, init_guess,  
                       args=(cov,), method='SLSQP',  
                       options={'disp': False},
```

```

        constraints=(weights_sum_to_1, return_is_target),
        bounds=bounds)

    return weights.x

def optimal_weights(n_points, er, cov):
    """
    Returns a list of weights that represent a grid of n_points on the e
    """
    target_rs = np.linspace(er.min(), er.max(), n_points)
    weights = [minimize_vol(target_return, er, cov) for target_return in
    return weights

```

5.3 Capital Asset Pricing Model (CAPM)

The Capital Asset Pricing Model (CAPM) describes the relationship between systematic risk and expected return for assets, particularly stocks. CAPM is widely used throughout finance for pricing risky securities and generating expected returns for assets given the risk of those assets and cost of capital.

$$ER_i = R_f + \beta_i(ER_m - R_f)$$

where

ER_i =expected return of investment

R_f =risk-free rate

i =beta of the investment

$(ER_m - R_f)$ =market risk premium

The goal of the CAPM formula is to evaluate whether a stock is fairly valued when its risk and the time value of money are compared to its expected return.

5.3.1 The CAPM and the Efficient Frontier

Using the CAPM to build a portfolio is supposed to help an investor manage their risk. If an investor were able to use the CAPM to perfectly optimize a portfolio's return relative to risk, it would exist on a curve called the efficient frontier, as shown on the following graph

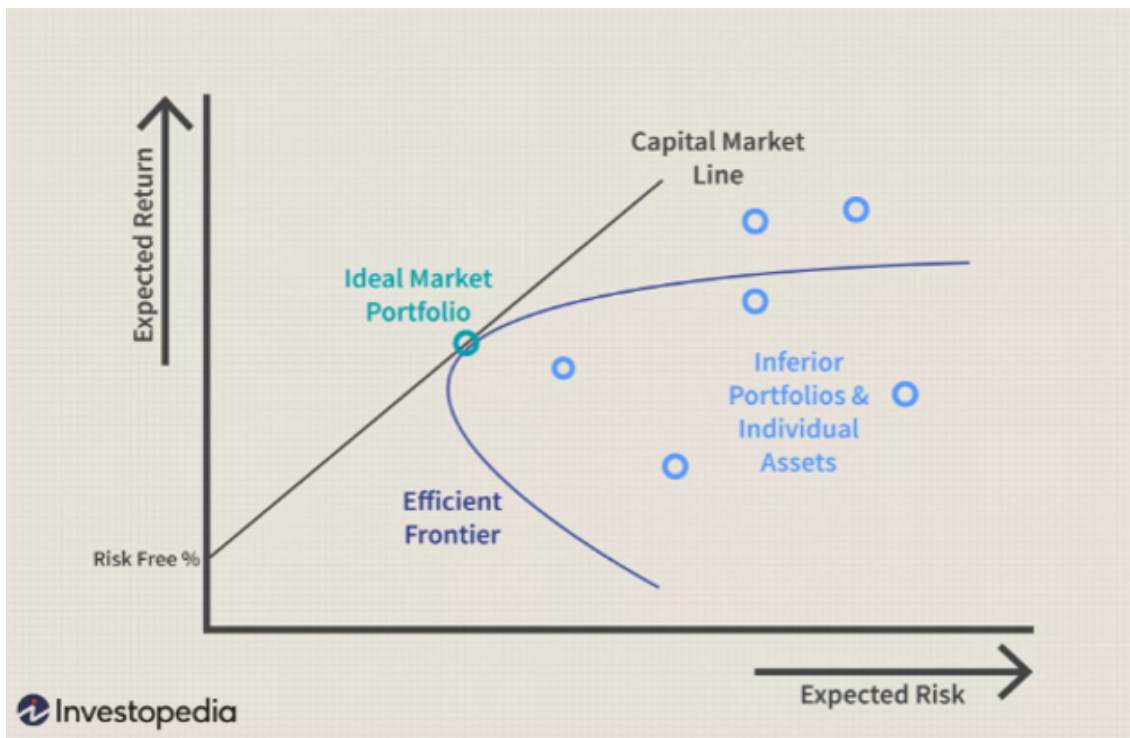


Figure 5.2: Efficient Frontier

The graph shows how greater expected returns (y-axis) require greater expected risk (x-axis). Modern Portfolio Theory suggests that starting with the risk-free rate, the expected return of a portfolio increases as the risk increases.

Chapter 6

Weekly progress from 25/09/21 to 1/10/21

6.0.1 Assumptions of Modern Portfolio Theory

Modern Portfolio theory has a certain assumption that is to be considered while making any decisions in order to arrive at the conclusion that risk, return, and diversification relationships hold true. The different assumptions of the modern portfolio theory are as follows:

- Returns from the assets are distributed normally.
- The investor making the investment is rational and will avoid all the unnecessary risk associated.
- Investors will give their best in order to maximize returns for all the unique situations provided.
- All investors are having access to the same information.
- The cost pertaining to taxes and trading is not considered while making decisions.
- All the investors are having the same views on the rate of return expected.
- The single investors along are not sizeable and capable enough to influence the prices prevailing in the market.
- Unlimited capital at the risk-free rate of return can be borrowed.

6.0.2 Advantages of the Modern Portfolio Theory (MPT)

There are several different advantages of the Modern portfolio theory providing the opportunity for the investors investing their money in the market. Some of the advantages are of the Modern portfolio theory as follows:

1. It helps in evaluating and managing risks and returns associated with the investments. With the help of analysis, the assets which are underperforming assets and the assets having an excessive risk with respect to returns can be scrutinized and then replaced with the new one.
2. The theory is an important tool for avoiding financial ruin because by following these theories, traders don't rely on only one investment for their financial stability; rather, they diversify their portfolio in order to get the maximum return with minimum risk.

6.0.3 Disadvantages of the Modern Portfolio Theory (MPT)

Along with the different advantages, there exist the limitations and drawbacks also of the Modern portfolio theory, which includes the following:

1. In the case of the modern portfolio theory, the past performance of the company under consideration is taken. The performance of the past never provides a guarantee for the result that could arise in the future. Considering only the past performances sometimes leads to overpassing the newer circumstances, which might not be there when historical data were considered but could play an important role in making the decision.
2. This theory assumes that there is a normal distribution of the return on an asset within a class of assets, which is proved to be wrong for individual equities as the correlations of asset class may change over the period of time.
3. In this theory, there is an assumption that securities of any of the sizes can be bought and sold, which doesn't hold true as some of the securities have minimum order sizes, which cannot be dealt with in the fraction.
4. Modern Portfolio Theory even though is accepted widely all over the world and also applied by different investment institution, but at the same time it has also been criticized by different persons particularly by representatives of the behavioral economics who challenges the assumptions of the Modern portfolio theory on the parameters of investor rationality and the expectations for the return.

5. Variance-based – The most serious criticism of modern portfolio theory is that it assesses portfolios based on variance and not the downside risks. As a result, two portfolios with the same level of variance and returns are usually considered equally desirable under the MPT. Frequent small losses could be the reason for the variance for one portfolio whereas the other could be showing similar variance because of two or three extraordinary declines. Most investors are likely to prefer frequent small losses as they are easier to endure. To overcome this shortfall of MPT, investors prefer to use the Post-Modern Portfolio Theory (PMPT), which attempts to improve on modern portfolio theory by minimizing downside risk instead of variance.

6.0.4 Conclusion

The main idea or the purpose of the Modern portfolio theory says that the risk is undertaken and return expected linked directly, which means that in order to achieve the greater rate of expected returns, an investor must have to take a higher level of risk. Also, the theory says that the overall risk of the portfolio having securities can be reduced through the means of diversification. In case two different portfolios are given to the investor having the same level of expected return, then the rational decision would be to choose the portfolio having lower total risk. Modern Portfolio Theory, even though it is accepted widely all over the world and also applied by different investment institutions, but at the same time, it has also been criticized by many people. Due to the above mentioned drawbacks of MPT we are currently learning about machine learning and reinforcement learning techniques to maximize the returns and minimize the risks.

Bibliography

- [1] M. Alizadeh, R. Rada, F. Jolai, and E. Fotoohi. An adaptive neuro-fuzzy system for stock portfolio analysis. *International Journal of Intelligent Systems*, 26(2):99–114, 2011.
- [2] S. Deng and X. Min. Applied optimization in global efficient portfolio construction using earning forecasts. *The Journal of Investing*, 22(4):104–114, 2013.
- [3] T. Fischer and C. Krauss. Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, 270(2):654–669, 2018.
- [4] C.-F. Huang. A hybrid stock selection model using genetic algorithms and support vector regression. *Applied Soft Computing*, 12(2):807–818, 2012.
- [5] C. Krauss, X. A. Do, and N. Huck. Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the SP 500. *European Journal of Operational Research*, 259(2):689–702, 2017.
- [6] S. I. Lee and S. J. Yoo. Threshold-based portfolio: the role of the threshold and its applications. *The Journal of Supercomputing*, 76(10):8040–8057, 2020.
- [7] C.-M. Lin, J.-J. Huang, M. Gen, and G.-H. Tzeng. Recurrent neural network for dynamic portfolio selection. *Applied Mathematics and Computation*, 175(2):1139–1146, 2006.
- [8] F. D. Paiva, R. T. N. Cardoso, G. P. Hanaoka, and W. M. Duarte. Decision-making for financial trading: A fusion approach of machine learning and portfolio selection. *Expert Systems with Applications*, 115:635–655, 2019.
- [9] V.-D. Ta, C.-M. Liu, and D. A. Tadesse. Portfolio optimization-based stock prediction using long-short term memory network in quantitative trading. *Applied Sciences*, 10(2):437, 2020.

- [10] W. Wang, W. Li, N. Zhang, and K. Liu. Portfolio formation with preselection using deep learning from long-term financial data. *Expert Systems with Applications*, 143:113042, 2020.