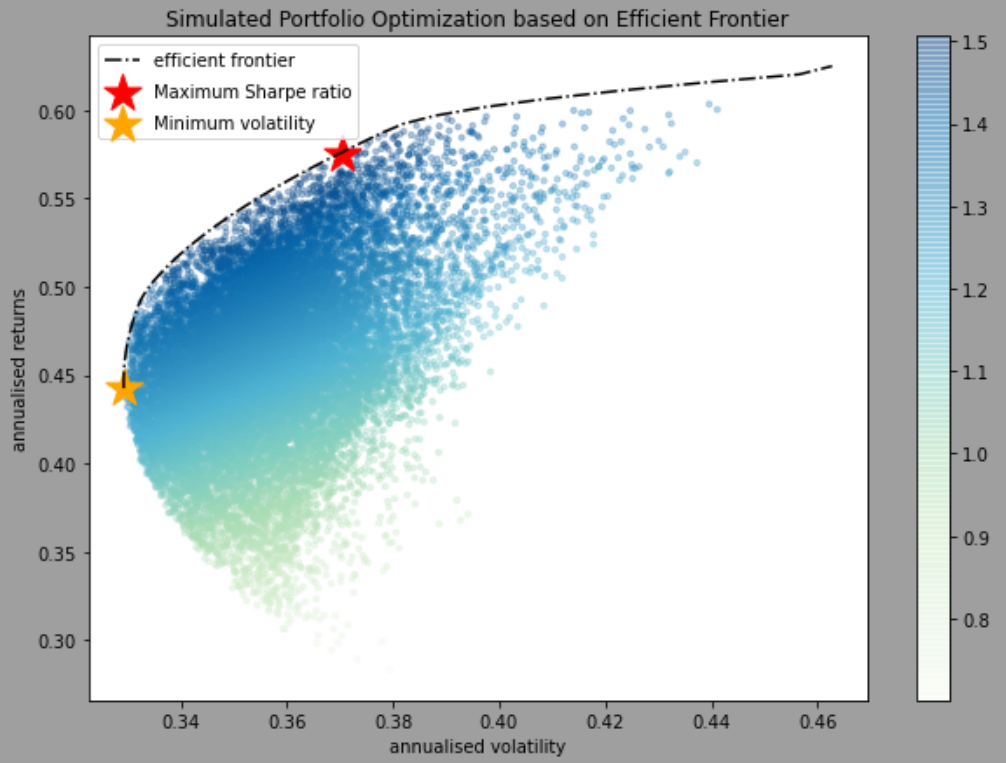
GB
730

How to Allocate Stocks to Construct a Portfolio?

Based on Markowitz Portfolio Selection Theory



GEN BUS 730: Prescriptive Modeling & Optimization
Instructor: Jordan Tong

Yuzhe Zhao
yuzhe.zhao@wisc.edu

Wisconsin School of Business
2020.12.14

Two Questions

Image you have some money to invest in stocks. When you look at thousands of stocks in the stock market, you naturally have two questions:

- Which stocks should I pick?
- How much should I allocate my money to them?

Markowitz Portfolio Selection Model

- It is a portfolio optimization model. Harry Markowitz raised it in his paper "Portfolio Selection," which was published in the Journal of Finance in 1952. He was later awarded a Nobel Prize for his work on modern portfolio theory.
- It assists in the selection of the most efficient portfolio by analyzing various possible portfolios of the given securities. It helps us to select the portfolio by,
  - Determination of a set of efficient portfolios.
  - Selection of the best portfolio out of the efficient set.
- https://www.math.ust.hk/~maykwok/courses/ma362/07F/markowitz_JF.pdf

Three problems were solved in this model

1. How to find the maximize Sharp ratio portfolio and minimum volatility portfolio for a set of stocks?
2. How to maximize the return (risk premium) for any level of volatility (standard deviation) of portfolio?
3. How to minimize the volatility (standard deviation) of the portfolio for any level of required return (risk premium)?

Analysis tools

notebook

Stock data were automatically collected

- It would automatically collect the past one year's stock data from Yahoo Finance to build the model.
- You can also choose option 2 and 3 in the code to customize your own data to calculate efficient frontier.

```
# Option 2: manually input end date, automatically calculate start date.
'''
end_date = input('Please enter today in YYYY/MM/DD format:')
def find_start_date(end_date):
    start_date = end_date.split('/')
    start_date = [str(int(start_date[0])-1),start_date[1],str(int(start_date[2])+1)]
    start_date = '/'.join(start_date)
    return start_date
start_date = find_start_date(end_date)
'''

# Option 3: manually input start data and end date.
#start_date = input('Please enter the start date in YYYY/MM/DD format:')
#end_date = input('Please enter the end date in YYYY/MM/DD format:')
```

How to find the maximize Sharp ratio portfolio and minimum volatility portfolio for a set of stocks?

1. Monte Carlo Simulation
    - Do a lot of random simulations to the weights of stocks, then calculate the annualized return and volatility based on the simulated weights and plot them on Charts.
2. Find out the maximum Sharpe ratio portfolio and minimum volatility portfolio. (Printed in the Jupyter Notebook.)

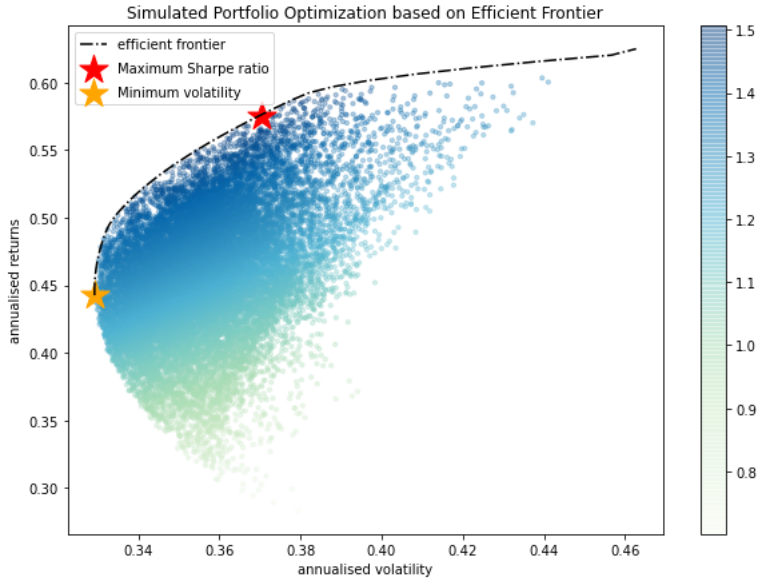


Chart 1

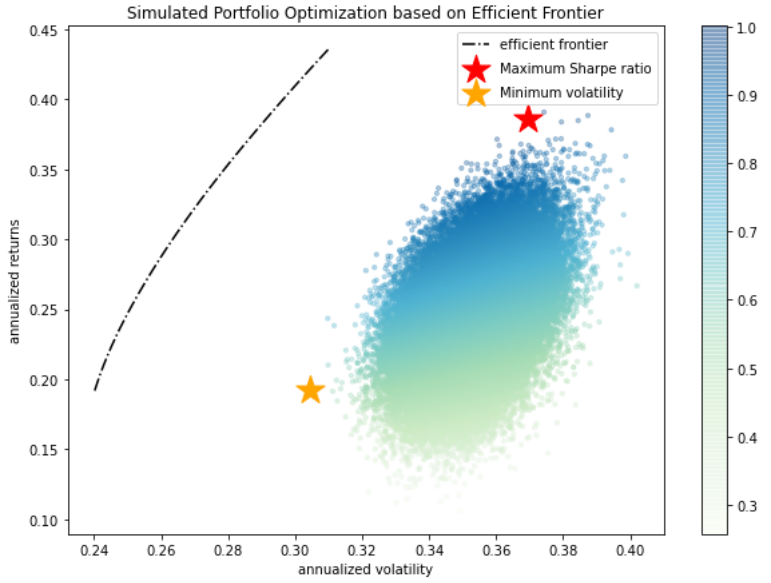


Chart 2

Chart 1: 25000 times simulations to calculate the weight for 4 stocks.

Chart 2: 100000 times simulations to calculate the weight for 25 stocks.

The simulation result for Chart 2 is far away from the efficient frontier, because the times of simulation (100000 times) is too small to include those points nearby the efficient frontier.

Time horizon for stock data: 2019/12/16 – 2020/12/11

How to maximize the return (risk premium) for any level of volatility (standard deviation) of portfolio?

1. A quadratic problem is solved with Pyomo
   - Objective: Maximize the annualized return of the portfolio
   - Decision Variables: weights of stocks
   - Constraints:
     a) Sum of all decision variables should be 1
     b) Annualized return of the portfolio should be larger than the required minimum return
2. Example: Solution for 25 stocks in the StockList.xlsx Excel File. The required minimum return is 28%.

```
---------------------------------------------------------------------------
The Minimum Volatility Rate for a Given Rate of Return

Annualized Return: 28.0%
Annualized Volatility: 25.8%
Asset Allocation: ['AAPL 0.0%', 'MSFT 0.0%', 'AMZN 36.29%', 'FB 0.0%', 'GOOGL 0.0%', 'JNJ 8.8
1%', 'PG 0.0%', 'V 0.0%', 'NVDA 0.0%', 'HD 0.0%', 'MA 0.0%', 'JPM 0.0%', 'UNH 0.0%', 'VZ 45.7
4%', 'ADBE 0.0%', 'CRM 0.0%', 'PYPL 2.53%', 'NFLX 6.63%', 'DIS 0.0%', 'INTC 0.0%', 'MRK 0.
0%', 'T 0.0%']
* Results corrected to two decimal places
---------------------------------------------------------------------------
```

Time horizon for stock data: 2019/12/16 – 2020/12/11

How to minimize the volatility (standard deviation) for any level of return (risk premium) of portfolio?

1. A quadratic problem is solved with Pyomo
   - Objective: Minimize the annualized volatility of the portfolio
   - Decision Variables: weights of stocks
   - Constraints:
     a) Sum of all decision variables should be 1
     b) Annualized volatility of the portfolio should be smaller than the required maximum volatility
2. Example: Solution for 25 stocks in the StockList.xlsx Excel File. The required maximum volatility is 26%.

```
---------------------------------------------------------------------------
The Maximum Rate of Return for a Given Volatility Rate

Annualized Return: 28.76%
Annualized Volatility: 26.0%
Asset Allocation: ['AAPL 0.0%', 'AMZN 37.06%', 'FB 0.0%', 'GOOGL 0.0%', 'JNJ 8.57%', 'PG 0.
0%', 'NVDA 0.0%', 'HD 0.0%', 'VZ 44.64%', 'PYPL 3.06%', 'NFLX 6.66%', 'DIS 0.0%', 'MRK 0.0%']
* Results corrected to two decimal places
---------------------------------------------------------------------------
```

Time horizon for stock data: 2019/12/16 – 2020/12/11

References:

1. Markowitz, H.M. (1952) Portfolio Selection. Journal of Finance, 7, 77-91.
2. Bodie, Z., Kane, A., & Marcus, A. J. (2013). Chapter 6: Efficient Diversification. In Essentials of investments (9th Global Edition ed., pp. 164-166). New York, NY: McGraw-Hill Education.
3. Dash, S. (2020, November 08). Portfolio Optimization with Python using Efficient Frontier with Practical Examples. Retrieved December 11, 2020, from https://www.machinelearningplus.com/machine-learning/portfolio-optimization-python-example
4. Ricky, T. (2019, February 26). [Advanced] How to Find the Optimized Portfolio with Python. Retrieved December 10, 2020, from https://cloud.tencent.com/developer/article/1395644
5. Neves, F. (2018, October 14). Plotting Markowitz Efficient Frontier with Python. Retrieved December 12, 2020, from https://towardsdatascience.com/python-markowitz-optimization-b5e1623060f5
6. Clark, K. (2020, November 17). Where to Find a List of the Stocks in the S&P 500. Retrieved December 12, 2020, from https://www.investopedia.com/ask/answers/08/find-stocks-in-sp500.asp
7. Chen, J. (2020, September 11). Modern Portfolio Theory (MPT). Retrieved December 14, 2020, from https://www.investopedia.com/terms/m/modernportfoliotheory.asp