451 Financial Engineering: Programming Assignment 2

Prepared by Logan MacPhail

July 25, 2025

1. Problem Description

This programming assignment utilizes Monte Carlo simulation to observe the effects of taking long and/or short positions across a group of assets. The goal is to observe both the expected returns and the relative risk associated with a longs only position compared to taking both long and short positions. Three stocks, DraftKings Inc. (DKNG), FanDuel (FLUT), and MGM Sports (MGM), were used for this assignment.

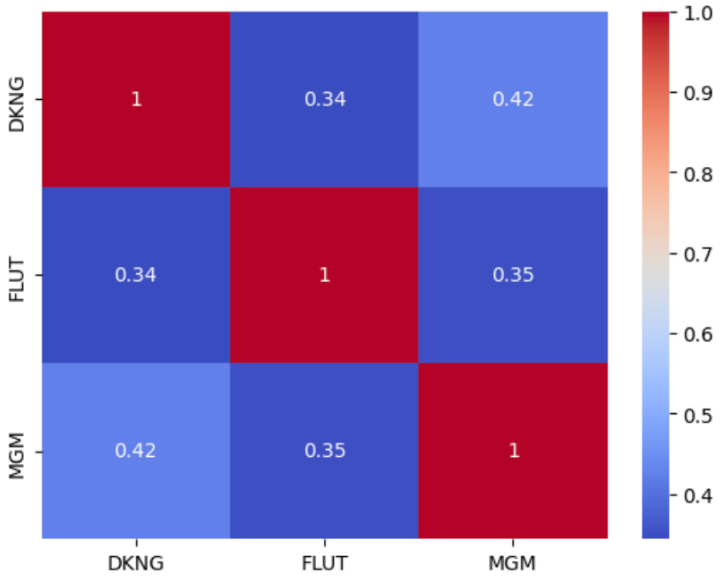
2. Data Preparation and Pipeline

The data used for this assignment comes from the *yfinance* module (yf) in Python, which is part of the Yahoo Finance package containing daily stock prices for most publicly traded stocks on the New York Stock Exchange. This assignment focuses on the daily log returns of the closing price of each stock for the last five years. The daily log returns were calculated by taking the natural log of the closing price divided by the one-lagged version of the closing price, representing the percentage change in the closing price from the previous day. Below is a summary of the daily log returns for each stock.

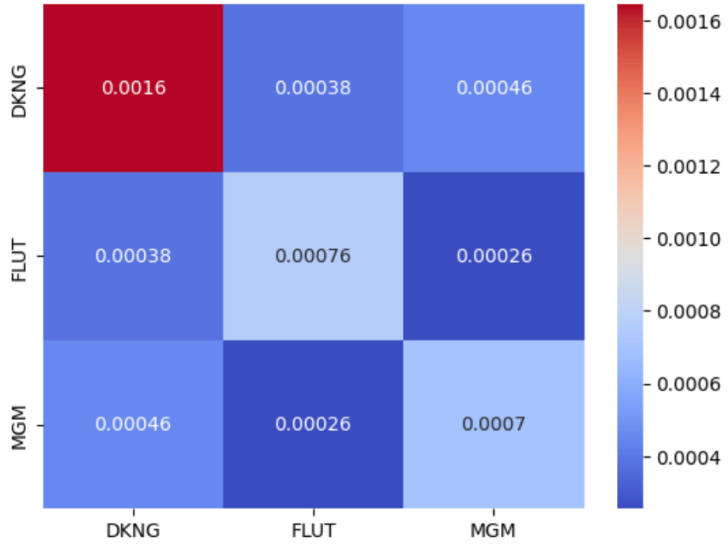
statistic	DKNG_LogReturn	FLUT_LogReturn	MGM_LogReturn
str	f64	f64	f64
count	1254.0	1254.0	1254.0
null_count	0.0	0.0	0.0
mean	0.000181	0.000643	0.000731
std	0.040567	0.027584	0.026433
min	-0.326061	-0.126518	-0.141777
25%	-0.023503	-0.014926	-0.013909
50%	0.0	0.000593	0.00054
75%	0.02272	0.016054	0.01492
max	0.159306	0.155277	0.160903

3. Research Design

After the data was imported and the daily log returns were calculated, a correlation matrix of the returns for the three stocks was created using the 'corr' function from the *pandas* library. Below is a look at the matrix.

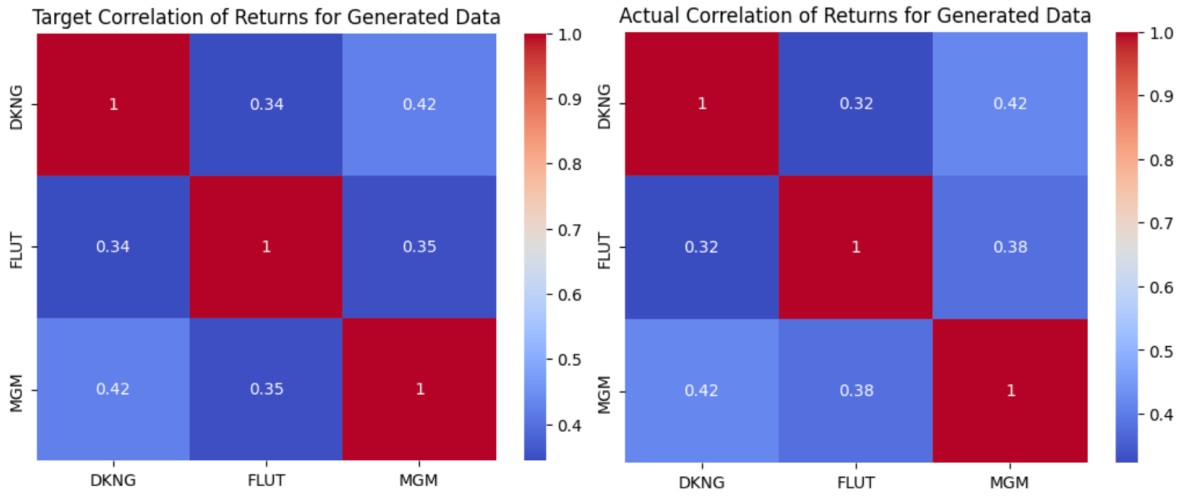


The three stocks are associated with companies in the sports gambling industry, so it makes sense that there would be some correlation between their stock prices, as they are within the same sector. A covariance matrix was also created, which is shown below.

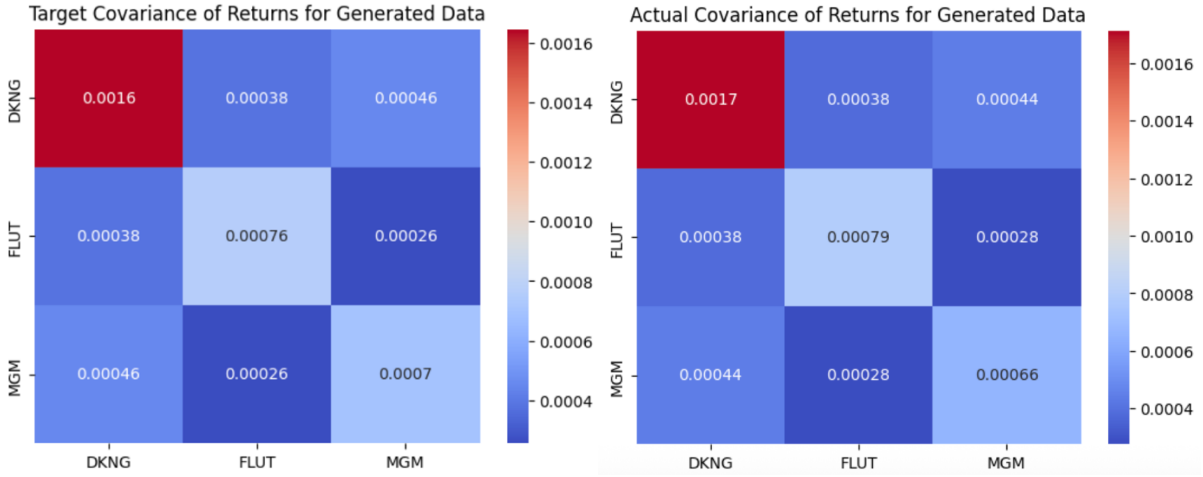


After the correlation and covariance matrices were created, the 'multivariate_normal' function from NumPy was used to generate 1000 random samples of means for each of the stocks, using the historical means and the covariance matrix. These samples would serve as the generated data to use for the Monte Carlo simulation. To test the results, the correlation and covariance matrices of the actual and generated data were compared.

Correlation Matrix Comparison



Covariance Matrix Comparison



The actual correlation and covariance of the generated returns were very similar to those of the historical data, indicating that the generated data provided an adequate representation of the historical data.

The final step before executing the Monte Carlo simulation was to create random weights that represented the level of investment in each stock. Two different sets of weights had to be made. One set of weights was for a portfolio that only allowed long positions. Those weights ranged from 0 to 1, with the sum of the weights equaling 1. The other set of weights was for a portfolio that allowed both short and long positions. Those weights ranged from -1 to 1, with numbers below 0 indicating short positions. The sum of these weights also added up to 1.

4. Models

Once the data and weights were generated, a Monte Carlo simulation was used to determine the expected returns and risk for each portfolio. Expected returns were calculated by matrix multiplying each set of weights by the transpose of the generated means, then averaging those results. The standard deviation, or "Risk," of the returns was calculated by matrix multiplying each set of weights by the covariance matrix of the generated data and the transpose of the weights, then taking the square root of that value. Each calculation was performed for each portfolio and stored in a resulting data frame.

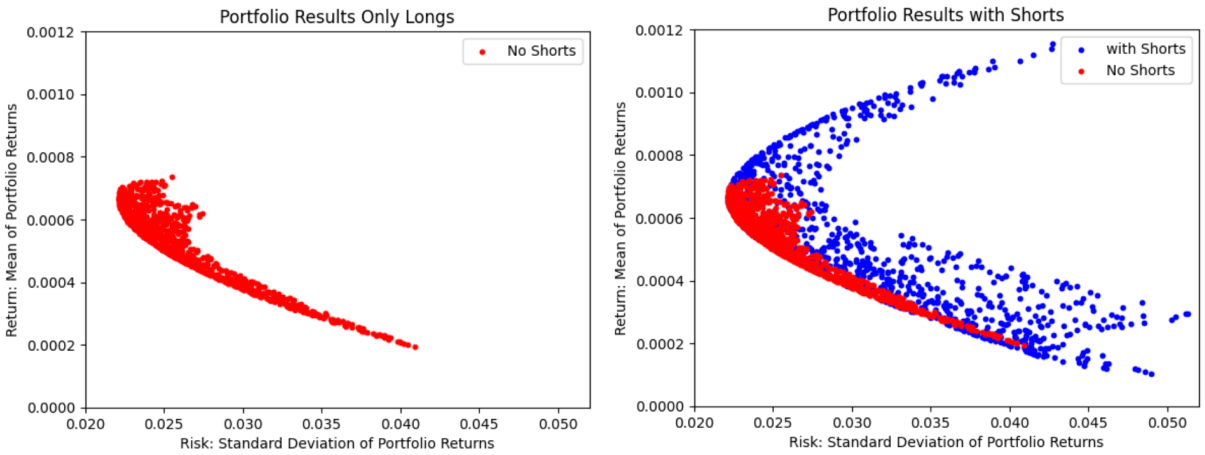
5. Results

Below is a summary of the expected returns and risk for the longs only portfolio and the shorts and longs portfolio. The longs and shorts is on the left, and the longs only is on the right.

statistic	expected_Return	sd_Return
str	f64	f64
count	1000.0	1000.0
null_count	0.0	0.0
mean	0.000509	0.031767
std	0.000229	0.006761
min	0.000104	0.022154
25%	0.000329	0.026083
50%	0.000466	0.030587
75%	0.000662	0.03688
max	0.001153	0.051308

statistic	expected_Return	sd_Return
str	f64	f64
count	1000.0	1000.0
null_count	0.0	0.0
mean	0.000514	0.026632
std	0.000124	0.004009
min	0.000194	0.02214
25%	0.000421	0.023726
50%	0.000534	0.025428
75%	0.000608	0.028381
max	0.000735	0.04091

The mean of both portfolios is fairly similar, with the shorts slightly ahead; however, the longs and shorts portfolio offers better returns and a higher upper end of its distribution compared to the longs only portfolio. The standard deviation of returns for the longs and shorts portfolio is higher than that of the longs only portfolio, indicating greater risk in the longs and shorts portfolio. This is better shown in the graphs below.



The longs only portfolio returns are more compact and tighter in the graph, but they also don't reach as high as the longs and shorts portfolio, indicating there are instances where the longs and shorts portfolio yields a higher return than the longs only portfolio. It is interesting to see the relative boomerang shape in the portfolio results. This suggests that there is a lot of variance in the returns that are on the lower end of the range for these particular stocks.

6. Conclusion

As expected, a portfolio that only takes long positions has much less risk but also greater upside, as shown by both the summary and graph of the longs only portfolio. A portfolio that involves both short and long positions has higher potential upside but also significantly more risk compared to the longs only portfolio. For these three specific stocks, each type of portfolio would make sense depending on the level of risk an individual investor is comfortable with.