In the name of God

ProjectII

Implement your code in a Jupyter notebook (ipynb file) so we can see your codes and results simultaneously.

1) Write a program to get the top five cryptocurrencies (by market cap); you can get the ticket name from the "crypto_currency" file, aggregate last year's closing price of these cryptocurrencies, scale them based on the mean and standard deviation, and plot altogether, then decide which one is profitable in this year based on cumulative return.

2) Generate a data frame consisting of the ten cryptocurrencies from "crypto_currency" file; based on the last 6-month daily returns, calculate how many days they had positive or negative returns; then calculate altcoins cumulative return when the bitcoin daily return is negative or positive.

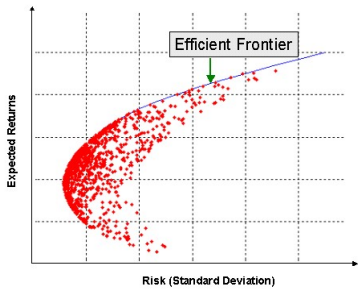
Cumulative return = $\prod(1 + returns_i)$

3) **Sharpe ratio**: Get BTC, ETH, XRP, last year's data; you want to create a portfolio from these; each of these tokens may contain 0 to 100% portfolio (for simplicity, assume each crypto share can range between 0 and multiplication of 5%), for each of possible composition, calculate portfolio return and volatility.($\sum (weights_i) = 1$)

Portfolio variance = $weights^T$ * covariance(portfolio) * weights

Portfolio return = $\sum (weights_i * returns_i)$

Sharpe ratio = (Portfolio return – risk free rate) / Portfolio std



Plot all possible Sharpe you get, assume the risk-free rate as zero, and ratios to a 2-D scatter. Save data to "portfolio.csv", and finally print out the best weights, which give us the highest ratio. (your plot must look something like on the left side)

4) **golden cross over:** get the bitcoin closing prices, calculate moving averages from 5 to 100 days; the objective is to find which pair of the two moving averages crossed together gives us the highest profit (for simplicity, consider a subset of 5 days to calculate moving average e.g.: the averages of 5, 10, 15, … 100)

Calculate how many times you have “buy” signal and “sell” signal.

Revise your solution when you have 5% transaction cost for the “buy” signal.

(moving average strategy: When the shorter-term moving average crosses above the longer-term moving average, it's a buy signal, as it indicates that the trend is shifting up. meanwhile, when the shorter-term moving average crosses below the longer-term moving average, it's a sell signal, as it indicates that the trend is shifting down)

5) Construct a function that checks whether the input number is prime or not (a prime number is a natural number greater than one that is not a product of two smaller natural numbers); write code from scratch and not use build-in functions.

6) **The best day for trading:** select five tokens from the "crypto_currency" file, set a 3-year time horizon, calculate the intraday return for each calendar date and calculate the best day for trading, do this exercise and find the best month for trading.

7) **rock, paper, scissor simulation**: define five rounds; in each round user chooses one shape (rock, paper, scissors), then your program should choose a random sample from one of that shapes; by applying the game rules, print out who wins this competition.

Repeat this game but this time, replace the user with another algorithm that chooses shape based on fixed proportional weights (1 for rock, 2 for paper, and 3 for scissors), then rerun this simulation 1000 times and print out the result. Which algorithm wins the game?

Good luck