Project02

#Problem1

A.

Parse the csv file, then turn it into the pandas dataframe. Select the tickers of SPY, AAPL and EQIX, use function pct_change() to calculate arithmatic returns, then minus the mean to get the arithmatic returns with zero mean. Display the last five rows of returns, and calculate the standard deviation using arithmatic returns dataframe with zero mean.

the last 5 rows of arithmetic returns:

	SPY	AAPL	EQIX
499	-0.011492	-0.014678	-0.006966
500	-0.012377	-0.014699	-0.008064
501	-0.004603	-0.008493	0.006512
502	-0.003422	-0.027671	0.000497
503	0.011538	-0.003445	0.015745

standard deviation:

SPY	0.008077
AAPL	0.013483
EQIX	0.015361

B.

Method is similar to question A, after getting the dataframe, using shift() function to get the lag(1) data, then calculate log return by using np.log(data1/data1.shift(1)). Remove the mean and calculate the standard deviation like the previous part did.

the last 5 rows of log returns:

	SPY	AAPL	EQIX
499	-0.011515	-0.014675	-0.006867
500	-0.012410	-0.014696	-0.007972
501	-0.004577	-0.008427	0.006602
502	-0.003392	-0.027930	0.000613
503	0.011494	-0.003356	0.015725

standard deviation:

SPY	0.008078
AAPL	0.013446
EQIX	0.015270

#Problem2

A.

After parsing the file and get the dataframe, create a dictionary to include the holdings, calculate the portfolio PV based on the sum of each stock's current price * share of

holdings.
PV = 251862.4969482422

B. a

Calculate the exponential weighted matrix with a lambda equals to 0.97, then multiply by the delta(weight), use np.sqrt(delta.T * ewma_cov.values * delta) to generate the sigma of the portfolio, finally calculate VAR and ES based on definitions.
About the individual stocks, it’s similar to the calculation of whole portfolio, the difference is select the variance from the overall ewma_cov_matrix, then use it as the individual stocks’ sigma.
results:
VAR_normal: 3856.3183014814754
(ES_normal): 4835.978727805545
SPY - VaR: 825.80, ES: 1035.59
AAPL - VaR: 944.78, ES: 1184.79
EQIX - VaR: 2931.34, ES: 3676.02

B. b

The t-distribution parameters (degrees of freedom, mean, and standard deviation) will first be fitted stock by stock using historical return data. The fitted t-distributions are then used to convert the historical returns to uniformly distributed CDF values, which are then inverted to obtain the standard normal scores using the inverse normal transformation. Copula correlation matrices are calculated and multivariate Monte Carlo simulations are performed, and finally VaR and ES are calculated for simulated returns, as well as for individual stocks.
(VAR_copula): 4163.3752453893
(ES_t_copula): 5814.3950203784425
SPY - VaR: 721.84, ES: 1010.53
AAPL - VaR: 962.12, ES: 1448.55
EQIX - VaR: 3237.74, ES: 4639.18

B. c

Directly use the dataframe to calculate the loss of each stock, then use the base definition of ES and VAR to calculate them.
(VAR_historical): 4314.934288727724
(ES_historical): 5799.2873041066305
SPY - VaR: 815.27, ES: 1022.97
AAPL - VaR: 997.25, ES: 1367.92
EQIX - VaR: 3501.99, ES: 4581.80

C.

The Delta-Normal method based on the assumption of a normal distribution calculates the lowest values of VAR and ES, showing its underestimation of the risk of extreme losses, while the use of the T-distribution in combination with Copula and historical simulation yields higher and similar results for VAR and ES, which better capture the thick-tailed nature of the asset returns and the extreme risks in the actual market, and therefore more closely match the true level of risk.

#Problem3

A.

After using all values from the Black-Scholes-Merton model and the call option price, we can generate the implied volatility by solving the equation. I choose the two-part method to calculate the iv by recursive function.

(iv): 0.33508046865463254

B.

Simply take the values of parameters into the formula of delta, vega and theta, we can get the results:

delta: 0.6659296311325037

vega: 5.640705582800387

theta: -5.544562353997324

C.

S + put_price: 32.25929736084998

call_price + X * np.exp(-r * T): 32.25929779053547

these two almost the same, which means call-put parity.

D.d

First, the sensitivity of the portfolio to stock price movements and time decay is calculated using the Delta and Theta of the current portfolio(delta_portfolio=1.3318; theta_portfolio=-8.1632). Based on the portfolio value (PV), the annual stock volatility, σ = 25%, is converted to the volatility over the holding period (20 trading days). The quantile (z-score) of the standard normal distribution is used to calculate the portfolio value at a given confidence level (5%) of the VAR; use the probability density function (PDF) of the standard normal distribution to calculate the portfolio ES.

(VAR_delta_normal): 6.048343783871945

(ES_delta_normal): 6.141716465986345

D.e

The change in stock price over the holding period (20 trading days) is first simulated and the future value of the option portfolio is recalculated based on the simulated future price of the stock; then the total value of the portfolio at the end of the holding period is calculated and the loss is defined as the difference between the current value of the portfolio and the future value; the simulated losses are then ranked, and the 5% quartile of the losses is taken as the VAR of the portfolio, and the portion of losses that exceeds the VAR is taken as the ES.

(VaR_MC): 5.655837735508804

(ES_MC): 7.650209572833033

E.

The difference between the two methods can be visualized by plotting Portfolio Value vs Stock Price:

When near the current stock price (S=31): The two curves are tangent because the slope of the straight line of delta normal is equal to the delta value of the Monte Carlo curve at the current point (first-order derivative matching). Here the approximation error is minimized and the delta normal method is locally valid.

When away from the current stock price (S significantly deviates from 31):

Delta Normal: remains linear and does not reflect the nonlinear characteristics of the option, because it assumes that portfolio value changes are approximated only by delta, ignoring the effects of gamma, vega, and so on. But it still provides a fast, reasonable approximation in the local range.

Monte Carlo: curve shows convexity (call option dominates when S rises) or concavity (put option dominates when S falls), reflecting nonlinear risk. This is because it dynamically calculating option values through BSM modeling accurately reflects the Gamma effect and delta changes with stock price. In conclusion, it provides relatively high accuracy results on a global scale, especially for nonlinear combinatorial sums, but is computationally expensive and time-consuming.