
Problem 1. *Calculate and compare the expected value and standard deviation of price at time t, P_t, given each of the 3 types of price returns, assuming $r \sim \mathcal{N}(\mu, \sigma^2)$ and the price at $t-1$, P_{t-1}. Simulate each return equation using $r \sim \mathcal{N}(\mu, \sigma^2)$ and show the mean and standard deviation match your expectations.*

Solution.

Classical Brownian Motion
Given:

$$P_t = P_{t-1} + r_t$$

Where the return r_t at time t is normally distributed variable with mean, $\mu = 0$ and variance σ^2. And today's price, P_{t-1} is known.

We know that if X is a random variable and a is a constant then:

$$E[X + a] = E[X] + E[a] = E[X] + a$$

As the expected value of a constant a, $E[a] = a$

And that the expected value of a normally distributed random variable $X \sim \mathcal{N}(\mu, \sigma^2)$ is its mean, μ

$$\implies E[P_t] = E[P_{t-1} + r_t] = E[P_{t-1}] + E[r_t] = E[P_{t-1}] + 0 = E[P_{t-1}] = P_{t-1}$$

Thus, the expected price at today, at time t is the yesterday's price at time $t-1$, P_{t-1}. This makes sense because the expected return, r_t is 0.

Similarly,
We know that if X is a random variable and a is a constant then:

$$Var(X + a) = Var(X) + Var(a)$$

And that $SD(X) = \sqrt{Var(X)}$
And that the variance of a constant, a, is $\sigma^2 = 0$

$$\implies Var(P_t) = Var(P_{t-1} + r_t) = Var(P_{t-1}) + Var(r_t) = 0 + \sigma^2$$

$$\implies SD(P_t) = \sqrt{Var(P_t)} = \sqrt{\sigma^2} = \sigma$$

Arithmetic Return System
Given:

$$P_t = P_{t-1} * (1 + r_t)$$

Where the return r_t at time t is normally distributed variable with mean, $\mu = 0$ and variance σ^2. And today's price, P_{t-1} is known.

We know that if X is a random variable and a is a constant then:

$$E[X + a] = E[X] + E[a] = E[X] + a$$

And:

$$E[aX] = E[a] * E[X] = a * E[X]$$

And if X is a random variable and Y is a random variable then:

$$E[X * Y] = E[X] * E[Y]$$

And that the expected value of a normally distributed random variable $X \sim \mathcal{N}(\mu, \sigma^2)$ is its mean, μ

$$\implies E[P_t] = E[P_{t-1} * (1 + r_t) = E[P_{t-1}] * (E[1] + E[r_t] = P_{t-1} * (1 + 0) = P_{t-1} * 1 = P_{t-1}$$

Thus, the expected price at time t, P_t, under the arithmetic return model, is equal to the known price at time $t - 1$, P_{t-1}. This is due to the fact that the expected rate of return r_t is 0, leading to an expected value of the multiplier $1 + r_t$ being 1.

Now to find the standard deviation:
We know that if X is a random variable and a is a constant then:

$$Var(Xa) = a^2 * Var(X)$$

And that:

$$Var(X + a) = Var(X) + Var(a)$$

And that $SD(X) = \sqrt{Var(X)}$

$$\implies Var(P_t) = Var(P_{t-1} * (1 + r_t)) = {P_{t-1}}^2 * Var(1 + r_t) = {P_{t-1}}^2 * (Var(1) + Var(r_t))$$

$$\implies Var(P_t) = {P_{t-1}}^2 * (0 + \sigma^2) = {P_{t-1}}^2 * \sigma^2$$

$$\implies SD(P_t) = \sqrt{Var(P_t)} = \sqrt{{P_{t-1}}^2 * \sigma^2} = P_{t-1} * \sigma$$

Log Return or Geometric Brownian Motion
Given:

$$P_t = P_{t-1} * e^{r_t}$$

Where the return r_t at time t is normally distributed variable with mean, $\mu = 0$ and variance σ^2. And today's price, P_{t-1} is known.
We know that if X is a random variable and a is a constant then:

$$E[aX] = E[a] * E[X] = a * E[X]$$

And for a normal variable $X \sim \mathcal{N}(\mu, \sigma^2)$ the moment generating function for a normal distribution gives:

$$E[e^{tX}] = e^{\mu t + \frac{1}{2}\sigma^2 t^2}$$

$$\implies E[e^{r_t}] = e^{0*1+\frac{1}{2}\sigma^2 1^2} = e^{0+\frac{1}{2}\sigma^2} = e^{\frac{1}{2}\sigma^2}$$

$$\implies E[P_t] = E[P_{t-1} * e^{r_t}] = E[P_{t-1}] * E[e^{r_t}] = P_{t-1} * e^{\frac{1}{2}\sigma^2}$$

Now to find the standard deviation:
We know that if X is a random variable and a is a constant then:

$$Var(Xa) = a^2 * Var(X)$$

And that $SD(X) = \sqrt{Var(X)}$
And that the variance of a random variable Y distributed log-normally such that $Y = e^X$ where $X \sim \mathcal{N}(\mu, \sigma^2)$ is given:

$$Var(Y) = (e^{\sigma^2} - 1) * e^{2\mu+\sigma^2}$$

$$\implies Var(e^{r_t}) = (e^{\sigma^2} - 1) * e^{\sigma^2}$$

$$\implies Var(P_t) = Var(P_{t-1} * e^{r_t}) = {P_{t-1}}^2 * Var(e^{r_t}) = {P_{t-1}}^2 * (e^{\sigma^2} - 1) * e^{\sigma^2}$$

$$\implies SD(P_t) = \sqrt{Var(P_t)} = \sqrt{{P_{t-1}}^2 * (e^{\sigma^2} - 1) * e^{\sigma^2}} = P_{t-1} * \sqrt{(e^{\sigma^2} - 1) * e^{\sigma^2}}$$

I then simulated each return equation using $r \sim \mathcal{N}(\mu, \sigma^2)$ in Python. My parameters were: $P_0 = 100$ (my initial price), $\sigma = 0.02$ (standard deviation of r_t), $\mu = 0$ (mean of r_t) and $num_simulations = 100000$ (number of random normal simulations to preform. After simulating I computed a single step of each price path: Classical Brownian, Arithmetic, and Geometric, from initial price, P_0 to simulated P_1. I then found the mean and standard deviation of my new price. The results of my simulations were as follows:

Classical Brownian Mean: 100.0000678682307 - Classical Brownian Std: 0.019981638134243328
Arithmetic Mean: 100.00678682306906 - Arithmetic Std: 1.9981638134243331
Geometric Mean: 100.02675485653333 - Geometric Std: 1.9990707679466926

These results do match my expectations. The theoretical/analytical mean for one step of the Classical Brownian Motion Returns is $\mu = P_{t-1}$. In the case of my simulation $P_{t-1} = P_0$. As we can see the mean of $P_1 \approx 100 = P_0 = P_{t-1}$. For Classical Brownian Motion returns the single step standard deviation we derived was our original σ. As we can see the standard deviation of $P_1 \approx 0.02 = \sigma$. The same holds for the Arithmetic Return System and Geometric Brownian Motion Return system. In the Arithmetic case the analytical mean for one step was likewise P_{t-1}. As we can see the mean of $P_1 \approx 100 = P_0 = P_{t-1}$. For the Arithmetic standard deviation we can see that $SD(P_1) \approx 2 = 100*0.02 = P_0*\sigma$. This is in line with the analytical result $SD(P_t) = P_{t-1} * \sigma$. For the simulated Geometric Brownian mean we have the mean of $P_1 \approx 100.020002 = 100 * e^{\frac{1}{2}\sigma^2}$ again in line with the analytical result. Lastly for the standard deviation we have $SD(P_1) \approx 2.0006009668 = 100 * \sqrt{(e^{0.02^2} - 1)e^{0.02^2}} = P_0 * \sqrt{(e^{\sigma^2} - 1) * e^{\sigma^2}}$

$\square$

Problem 2. *Implement a function similar to the return_calculate() in this week's code. Allow the user to specify the method of return calculation. Use DailyPrices.csv. Calculate the arithmetic returns for all prices. You own 1 share of META. Remove the mean from the series so that the mean*$(META) = 0$ *Calculate VaR:*

1. *Using a normal distribution.*
2. *Using a normal distribution with Exponentially Weighted variance* ($\lambda = 0.94$)
3. *Using an MLE fitted T distribution*
4. *Using a fitted* $AR(1)$ *model*
5. *Using a Historic Simulation*

Compare the five values.

Solution.
First I implemented my own version of the *return_calculate*() function from class in Python. Then I wrote a function that mean centers data series to remove the mean from the META returns. After that I calculated VaR in each of the five ways. I let $\alpha = 0.05$. For the fitted Historic Simulation model I let $numberdraws = 1000$ and used a KDE to smooth the returns and address limitations in the data. The results were as follows:

Normal VaR at $\alpha = 0.05$: 0.05428693242254699
EWV VaR at $\alpha = 0.05$: 0.0301370702553927
MLE fitted t-Dist VaR at $\alpha = 0.05$: 0.04313471495037609
Fitted AR(1) VaR at $\alpha = 0.05$: 0.05383668898535073
Historical VaR with KDE at $\alpha = 0.05$: 12.497194279632197

The Historical VaR is clearly significantly larger than each of the other four methods which range from ≈ 0.03 to ≈ 0.05. This might be because Historical VaR provides a more conservative estimate of risk. It is a non-parametric approach, thus we make no assumptions about the distributions of returns. Plus, there is a recency bias so any recent market volatility is not as smoothed out as in a parametric approach. Lastly, we can consider fat-tails. Financial return distributions often exhibit fat tails and skewness, meaning they have a higher likelihood of extreme outcomes than a normal distribution would predict. Historical VaR captures these characteristics of the actual data, while parametric methods based on the assumption of normality might not. □

Problem 3. *Using* $Portfolio.csv$ *and DailyPrices.csv. Assume the expected return on all stocks is* 0. *This file contains the stock holdings of 3 portfolios. You own each of these portfolios. Using an exponentially weighted covariance with* $\lambda = 0.94$, *calculate the VaR of each portfolio as well as your total VaR (VaR of the total holdings). Express VaR as dollars. Discuss your methods and your results. Choose a different model for returns and calculate VaR again. Why did you choose that model? How did the model change affect the results?*

Solution.

For this problem I utilized my ewCovar() function from *Project_03* as well as built two functions to calculate exponentially weighted covariance VaR in dollars for the individual portfolios and total portfolio. I calculated the covariance matrix of returns using exponentially weighted averages, where more weighting is spread more across the historical window. This is achieved by applying a large decay factor ($\lambda = 0.94$) to the weights. This method can be beneficial in capturing greater market trends and volatilities that extend throughought the historical window. For each portfolio, I calculated the variance by considering both the holdings in the portfolio and the covariance between the assets. This takes into account not only the individual risk of each asset but also how they interact with each other. I adjusted the calculations to consider the current market value of holdings (number of shares times current stock price), converting the VaR from a relative measure of risk to an absolute dollar value. This reflects the potential dollar loss in the portfolio. VaR is computed as the product of the portfolio's standard deviation (square root of the variance) and the z-score associated with the desired confidence level. This gives the minimum potential loss under normal market conditions at that α (1 minus the confidence level). The individual portfolio VaRs and total portfolio VaR I calculated using exponentially weighted covariance were as follows:

Portfolios VaR: 'A': 2978.2024806765453, 'B': 1505.7058524471406, 'C': 3182.906724859518
Total VaR: 7329.606790318982

Note that the VaR of the individual portfolios combined is greater than the Total VaR of all three portfolios!

The different model I chose was Historical Simulation because I like the idea of comparing a more conservative "data-driven" approach with the more "market-sensitive" approach of exponentially weighted covariance. Again I applied kernel density estimation to smooth the data non-parametrically. VaR is determined by finding the point in the range of historical value changes where the cumulative distribution (obtained via KDE) reaches the desired confidence level (1 minus α). This represents the minimum expected loss under normal market conditions in that 5% worst case scenario. Again I made sure to convert the VaR to dollar amounts for comparison purposes. The results were as follows:

Portfolios (KDE): Portfolio: A 19992.736264, B 11825.958596, C 25314.706079
Total (KDE): 54108.84252659601

Not surprisingly, the Historical Simulation VaR model provided much more "conservative" (i.e., larger) estimates of the minimum expected loss on a 5% bad day. Again, the VaR of the individual portfolios combined is greater than the Total VaR of all three portfolios. □