Week 4

---- Dong Pu

Question 1

- Question:

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 $r_t \sim N(0, \sigma^2)$. Simulate each return equation using $r_t \sim N(0, \sigma^2)$ and show the mean and standard deviation match your expectations.

Question 1 Answer

- Classical Brownian Motion

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

Expectation :

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

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

$$E[P_t] = P_{t-1}$$

Std :

$$Std[P_t] = Std[P_{t-1} + r_t]$$

$$Std[P_t] = Std[P_{t-1}] + Std[r_t]$$

$$Std[P_t] = \sigma$$

Question 1 Answer

- Arithmetic Return System

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

Expectation:

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

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

$$E[P_t] = P_{t-1}.$$

Std:

$$std[P_t] = std[P_{t-1} + P_{t-1} r_t]$$

$$std[P_t] = std[P_{t-1} r_t]$$

$$std[P_t] = P_{t-1}\sigma.$$

Question 1 Answer

- Log Return or Geometric Brownian Motion

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

Expectation:

$$E[\ln(P_t)] = E[\ln(P_{t-1} e^{r_t})]$$

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

$$E[\ln(P_t)] = \ln(P_{t-1}).$$

std:

$$std[\ln(P_t)] = std[\ln(P_{t-1} e^{r_t})]$$

$$std[\ln(P_t)] = std[\ln(P_{t-1})] + std[r_t]$$

$$std[\ln(P_t)] = \sigma.$$

Question 1 Answer

- If we assume $P_{t-1} = 50$, $\sigma = 0.5$:
- Classical Brownian Motion: $E[p_t] = 50$; $std[P_t] = 0.5$
- Arithmetic Return System: $E[p_t] = 50$; $std[P_t] = 25$
- Log Return or Geometric Brownian Motion: $E[\ln(p_{t)}] = \ln(50) = 3.91$; $std[\ln(P_t)] = 0.5$
- From python analysis:

```
50.00095988636931 0.4970290227834332
50.04799431846559 24.851451139171658
3.9129828917974585 0.4970290227834332
```

- We can see that the result from simulation matches our calculation.

Question 2

- Question:
- 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.
- 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 an Exponentially Weighted variance ($\lambda = 0.94$)
3. Using a MLE fitted T distribution.
4. Using a fitted AR(1) model.
5. Using a Historic Simulation.

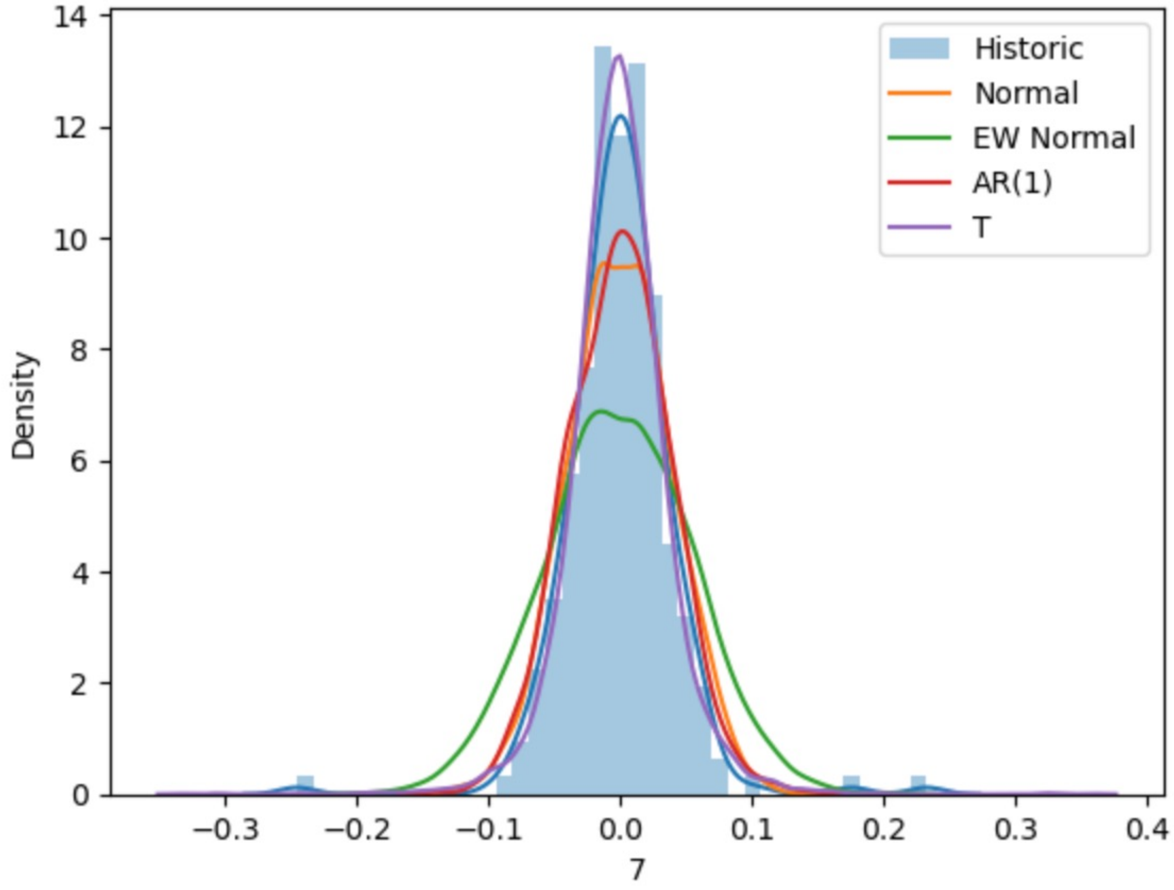
- Compare the 5 values.

Question 2 Answer

- Normal VaR = 0.06444215236842395
- Exponentially Weighted normal VaR = 0.094140771906898
- T Distribution VaR = 0.05722198493958679
- AR(1) fitted VaR = 0.0630520826909188
- Historic VaR = 0.0546200790823787

Question 2 Answer

We can see from the diagram at the right, the T distribution best describes the data given in the DailyPrices.csv.



Question 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 a $.
- 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?

Question 3 Answer

- I conducted Shapiro-Wilks test on the data to check what percentage of stocks follow normal distribution. I used it to determine if normal distribution can be assumed and use corresponding methods.
- The percentages of normally distributed returns for each portfolio are all around 30%. Normal distribution for these portfolios cannot be assumed. Although T distribution appears to be good for historic data, it may not be good for prediction future data. I choose to use historic VaR for these portfolios.

Question 3

Historic VaR:

```
Portfolio A: 8740.470609052107
Portfolio B: 6903.910270434921
Portfolio C: 5467.113886933308
Portfolio Total: 20904.80322360387
```

Since using a KDE to smooth the VaR estimation is highly recommended, I also tried to use KDE method as well. The model change did not affect much of the results.

Historic VaR smoothed by KDE:

```
Portfolio A: 8514.201940137194
Portfolio B: 7194.656898739166
Portfolio C: 6012.2212601721985
Portfolio Total: 21582.070704863523
```