FinTech545 - hw4

Samuel Fuller

February 2024

1 Problem 1

Calculate and compare the expected value and σ of price at time t (P_t), given each of the 3 types of price returns assuming $r_t \sim N(0, \sigma^2)$ and price at $t-1$ (P_{t-1}).

- Classical Brownian Motion:

$$P_t = P_{t-1} + r_t \tag{1}$$
$$E[r_t] = \mu = 0 \tag{2}$$
$$E[P_t] = P_{t-1} + \mu = P_{t-1} \tag{3}$$

  With respect to variance, P_{t-1} is a constant with zero variance, thus all of the variance in P_t comes from r_t and therefore

$$\sigma_{P_t} = \sigma$$

- Arithmetic Return System:

$$P_t = P_{t-1}(1 + r_t) \tag{4}$$
$$E[r_t] = \mu = 0 \tag{5}$$
$$E[P_t] = P_{t-1}(1 + 0) = P_{t-1} \tag{6}$$

  With respect to variance:

$$\sigma^2_{a+bX} = b^2\sigma^2_X \tag{7}$$
$$\sigma^2_{P_t} = P^2_{t-1}\sigma^2_{1+r_t} \tag{8}$$
$$\sigma^2_{1+r_t} = \sigma^2_{r_t} \tag{9}$$
$$\sigma^2_{P_t} = P^2_{t-1}\sigma^2_{r_t} \tag{10}$$
$$\sigma_{P_t} = (P^2_{t-1}\sigma^2_{r_t})^{0.5} \tag{11}$$

- Geometric Brownian Motion:
  In this case, P_t is lognormally distributed (still assuming $r_t \sim N(0, \sigma^2)$):

$$ln(P_t) = ln(P_{t-1}e^{r_t}) = ln(P_{t-1}) + r_t \tag{12}$$
$$ln(P_t) \sim N(ln(P_{t-1}) + 0, \sigma^2) \tag{13}$$
$$\text{it follows that:} \tag{14}$$
$$P_t \sim LN(ln(P_{t-1} + 0, \sigma^2) \tag{15}$$

  We know that for the lognormal distribution, we can use the following moment generating function to solve for $E[P_t]$ and $\sigma^2_{P_t}$

$$E[P_t] = exp(\mu + \frac{1}{2}\sigma^2)$$
$$\sigma^2_{P_t} = exp(2(\mu + \sigma^2)) - exp(2\mu + \sigma^2)$$

  Therefore:

$$E[P_t] = exp(ln(P_{t-1}) + \frac{1}{2}\sigma^2)$$

$$\sigma^2_{P_t} = exp(2(ln(P_{t-1}) + \sigma^2)) - exp(2ln(P_{t-1}) + \sigma^2)$$

$$\sigma_{P_t} = \sqrt{\sigma^2_{P_t}}$$

$$\sigma_{P_t} = \sqrt{exp(2(ln(P_{t-1}) + \sigma^2)) - exp(2ln(P_{t-1}) + \sigma^2)}$$

Using $r_t \sim N(0, \sigma^2)$ and $P_{t-1} = 100$ where $\sigma = 0.01, \sigma^2 = 0.0001$:

Price Algorithm	$\mu_{observed}$	$\mu_{expected}$	$\sigma_{observed}$	$\sigma_{expected}$
Classical Brownian Motion	100.0000	100.0000	0.009995	0.01000
Arithmetic Return System	100.0006	100.0000	0.9995	1.0000
Log Return	100.0056	100.0050	0.9996	1.0001

Table 1: Price Algorithm Empirical Results vs Expected

Clearly, given the above values, our empirical results match our expected means and standard deviations for all three types of price returns.

2 Problem 2

VaR is the minimum amount (\$ or %) a portfolio is expected to lose on a day given that day is a $\alpha = \%$ "bad" day. After filtering the price data and implementing a "return_calculate" function in Python, we calculate arithmetic returns for the given price dataset, remove the mean, and then calculate VaR through various methods. For this problem, we will assume $\alpha = 5\%$. We also multiply the VaR return by the last reported META share price to get our absolute (adjusted for slightly positive average return) dollar VaR.

– **Normal dsitribution:**
To evaluate the VaR using the normal distribution, we calculate $-\Phi^{-1}(\alpha) = -1.645$. Taking the sample standard deviation from our mean-centred arithmetic returns σ_s computationally, $VaR = - * (-\Phi^{-1}(\alpha)) * \sigma_s$. We also are sure to account for our mean centering to calculate the absolute value on the profit/loss distribution (mean return = 0.278%)

$$VaR = 299 * (1.645 * 0.033 + 0.00278) = \$15.40$$

– **Normal distribution with EW variance:**
We then take the same formula as above but use the EW variance instead of σ_s (using our EW function from previous homework with $\lambda = 0.94$). We also account for the mean again to yield

$$VaR = -\text{share price} * (-\Phi^{-1}(\alpha)) * \sigma_E W$$

$$VaR = -299 * (1.645 * 0.015 + 0.00278) = \$6.30$$

– **MLE fitted T distribution:**
Using scipy we fit a T distribution to the meta returns data with the parameters determined by MLE. Once that T distribution is fit ($df = 2.87$), we then take the x value at α percentile of that distribution and then multiply by the sample standard deviation to yield the VaR return as a percentage. When multiplied by the close share price, our VaR is \$23.67. We also take care of the mean centering by outputting a zero mean T distribution.

– **AR(1) model**
Fitting the AR(1) model to our returns distribution and then evaluating the standard deviation of the residuals to estimate the σ of the next return (given the fitted AR parameters), we come to the value \$16.94 for our VaR. This is the value of our specific sample but there is a distribution for the residuals standard deviation dependent on the model so other trials may yield varying results

– **Historic simulation:**
Sampling 10000 times with replacement from the mean-centred historical META returns as determined by our return_calculate function and multiplying those returns by the most recent share price, we get the bootstrapped distribution of returns for META. Then evaluating the 5th percentile of those samples using the np.percentile function, we achieve the minimum dollar loss on the worst 5% of days which equals \$12.57

In summary:

Method	Calculated VaR (\$)
Using a normal distribution.	15.40
Using a normal distribution with an Exponentially Weighted variance ($\lambda = 0.94$)	8.13
Using a MLE fitted T distribution.	23.67
Using a fitted AR(1) model.	16.94
Using a Historic Simulation.	12.56

In theory, the exponential weighted VaR being lower could indicate less volatility in recent markets relative to the unweighted variance (which is true). The difference between the historic simulation result and the normally distributed results indicate the distribution of returns is somewhere in between a normal and a T distribution given neither was accurate with respect to our bootstrapped sample. The T distribution likely overestimates the true value indicating the tails may be slightly too fat for the true return distribution.

3 Problem 3

I approached this problem using the Delta Normal approach with varying λ. The high-level steps of the algorithm for evaluating porfolio VaR I implemented are below (where $\alpha = 0.05$), please see the algorithms in my repo:

- Create most recent price dataframe with tickers and create new dataset of each ticker an its most recent close price
- Join most recent close data frame to portfolio data frame (so that we have quantity and price in the portfolio dataset)
- Iterate through portfolio and calculate holding values for each holding, sum these holding values in the loop as well
- Divide each holding's holding value by the total portfolio to solve for each delta
- Calculate the exponentially weighted covariance matrix, Σ ($\lambda = 0.94$) using the original price data (transformed into arithmetic returns data) for the securities in the portfolio
- Solve the following equation for p_sig where $\nabla R = \hat{\delta}$:

$$\text{p sig} = \sqrt{\nabla R^T \Sigma \nabla R}$$

- finally, solve the following for $VaR(\alpha)$

$$\text{VaR}(\alpha) = -PV \times \Phi^{-1}(\alpha) \times \sqrt{\nabla R^T \Sigma \nabla R}$$

The algorithm was run on all of the portfolios (A,B,C) as well as the overall portfolio

Portfolio	Calculated VaR ($)	Portfolio Value ($)
A	12,073.95	1,089,316
B	4,150.22	573,542
C	20,783.20	1,387,410
Total	98,416.73	3,051,268

Table 2: Portfolio Value at Risk (VaR) and Values, $\lambda = 0.94$

The second approach remained the same but changed lambda to zero.

Portfolio	Calculated VaR ($)	Portfolio Value ($)
A	18,604.92	1,089,316
B	6,039.5	574,542
C	37,059.99	1,387,410
Total	156,777.50	3,051,268

Table 3: Portfolio Value at Risk (VaR) and Values, $\lambda \sim 0.0$

When we set $\lambda = 0$, we effectively uniformly weight observations as opposed to $\lambda = 0.94$ which induces bias toward more recent observations. Given our price data begins in September 2022 and ends in September 2023, if we look at the VIX as a general measure of volatility (which is fair given the diversification of our portfolios):

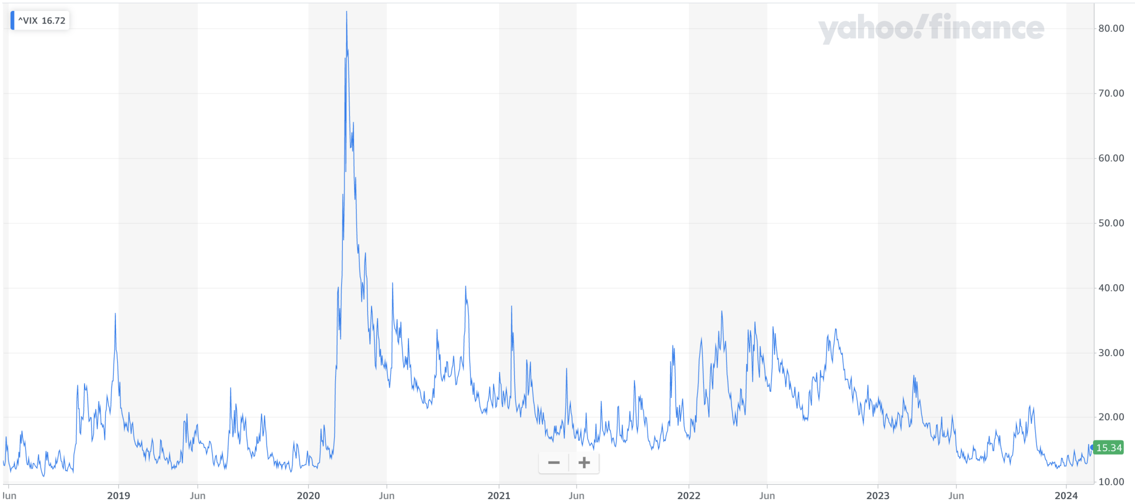

Figure 1: VIX from 2019 - Present

The data shows a generally higher volatility environment toward the beginning of the dataset relative to the end. Thus, we would expect VaR to be higher for the dataset which equally weights the observations relative to the $\lambda = 0.94$ dataset which biases the later observations. This is observed empirically in our findings. This cause and effect highlight the relationship between λ and the weighting of the dataset as well as market volatility.