FinTech 545 Project Week05:

Leslie Dees

Problem1

Fit a normal distribution and a generalized T distribution to the data in problem1.csv

I used my own MLE fitting functions for both the normal distribution and t distribution. This required creating new methods that take in a single data input rather than an x and y. The optimized values are:

Distribution	Mean	Standard Dev	Degrees of Freedom
Normal	-0.000879	0.048865	N/A
T	-4.687518	0.036439	4.2511

Calculate VaR and ES for both fitted distributions.

The methods *calc_var_normal* and *calc_var_t* were created to calculate the Value at Risk from a prior assignment. These perform the inverse quantile function given a scale, location, and degree of freedom in the case of the t distribution. I obtained a ***Value at Risk*** of:

Normal Distribution: **0.081255**

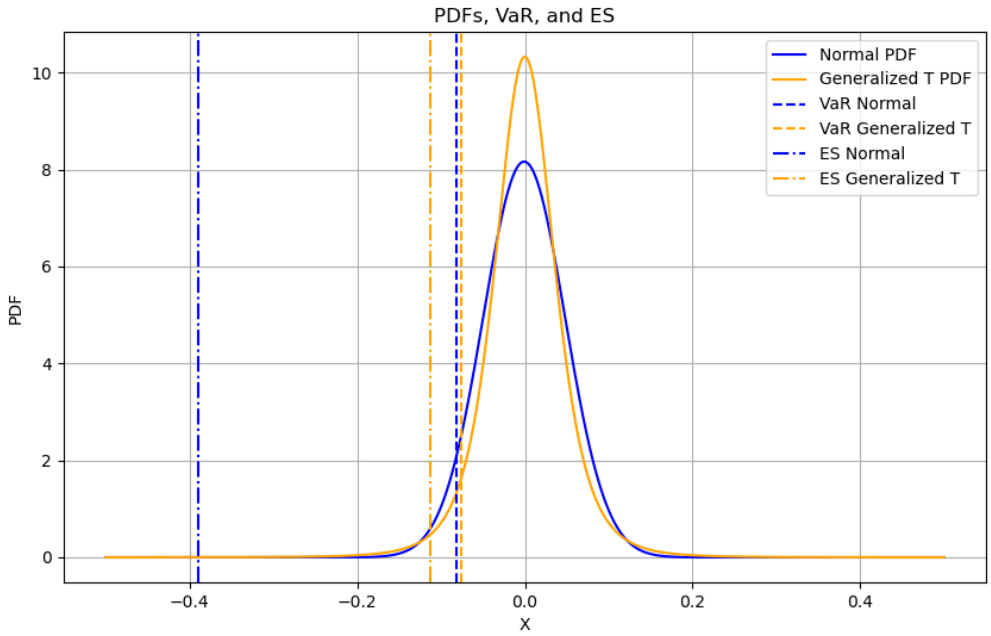
T Distribution: **0.076429**

And an ***Expected Shortfall*** of:

Normal Distribution: **0.389477**

T Distribution: **0.113171**

Overlay the graphs the distribution PDFs, VaR, and ES values. What do you notice? Explain the differences.



Firstly, all values were converted to negative for visualization on the graph's same side. This is for simplicity of viewership. What I can see from this graph is that the PDF for the Generalized T distribution is shorter with fatter tails than the Normal PDF. The T distribution's higher standard deviation results in a slightly higher absolute Value at Risk due to increased volatility. This also mirrors into the expected shortfall. The expected shortfall of the T distribution is drastically higher than for the Normal distribution. I can only explain this through the increased volatility explained by an increased standard deviation of the PDF. While I would expect the expected shortfall to be higher in the T distribution due to this factor, the extreme increase in distance seems somewhat off. I was unable to find any code errors though, so I can only explain this by the distribution shape.

Problem 2

I had already made a library in the form of fin_package.py, and so I just added updates to this library. The library was updated with inclusions for the separate functions that I included in other week assignments. I attempted to build a testing suite for values that I could validate, however due to a lack of confirmation of several values I was unable to completely build out the suite. There are still several functions in the notes that I have not implemented yet and plan to continue working back to include over the next several weeks in preparation for future homework assignments.

Problem 3

Using repository from #2

Fit a Generalized T model to each stock.

I created a dictionary to contain all stock T distributions that will be used later. For the sake of experimentation, I utilized 2 methods to calculate the T distributions. Firstly, was my function for MLE optimization to obtain T distribution parameters. Secondly, was the scipy.stats.t.fit() method. The Expected Shortfalls obtained were drastically different depending on this difference as explained later.

Calculate VaR and ES of each portfolio as well as total VaR and ES.

Week05 Value at Risk: Lambda = 0.97

Portfolio	**Value at Risk**	**t.fit() T Distribution Expected Shortfall**	**mle_t_distribution_one_input T Distribution Expected Shortfall**
A	**$48,640.93**	**$21,170.89**	**$6,778.23**
B	**$23,741.26**	**$13,502.40**	**$4,085.22**
C	**$82,029.31**	**$28,447.57**	**$9,691.68**
Whole	**$94,545.15**	**$21,114.37**	**$6,914.11**

Compared to the Value at Risk I obtained previously in Week04, the value at risk for every portfolio is calculated to be higher. The Values at Risk prior were obtained through the same exponentially weighted covariance matrix technique, with the only difference being the lambda values chosen. For this experiment I used a lambda of 0.97, as recommended. Below we can see the Week04 results, which contain lower values at risk. What this tells me is that the higher lambda value from Wee05 is weighting more recent data higher. Therefore, there must be more volatility in the market right now which Is why the Value at Risk has increased.

Week04 Value at Risk: Lambda = 0.97

Portfolio	Value at Risk ($)
A	$47,798.41
B	$23,249.99
C	$78,936.99
All	$91,962.09

As for the expected shortfall, here is where I decided to integrate the different T distribution techniques. For some reason, the means that were generated from my MLE method tend to be double what the means of the t.fit() method means are. Also, my MLE degrees of freedom is several factors higher than the t.fit() df. While I cannot pinpoint exactly why this is, I understand that it must be a difference in how the MLE method is fitting. Further experimentation into why this happens is needed in coming weeks.

We observe that the whole portfolio Expected shortfall tends to be close to Portfolio A's value, with B and C as outliers on either side. This is due to the Expected Shortfall being calculated as the mean of all expected shortfalls for each stock in the portfolio.