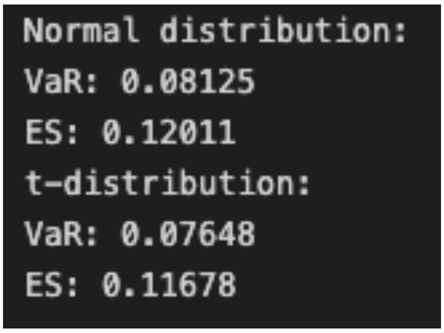
Problem 1

Use the data in problem1.csv. Fit a Normal Distribution and a Generalized T distribution to this data. Calculate the VaR and ES for both fitted distributions.

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

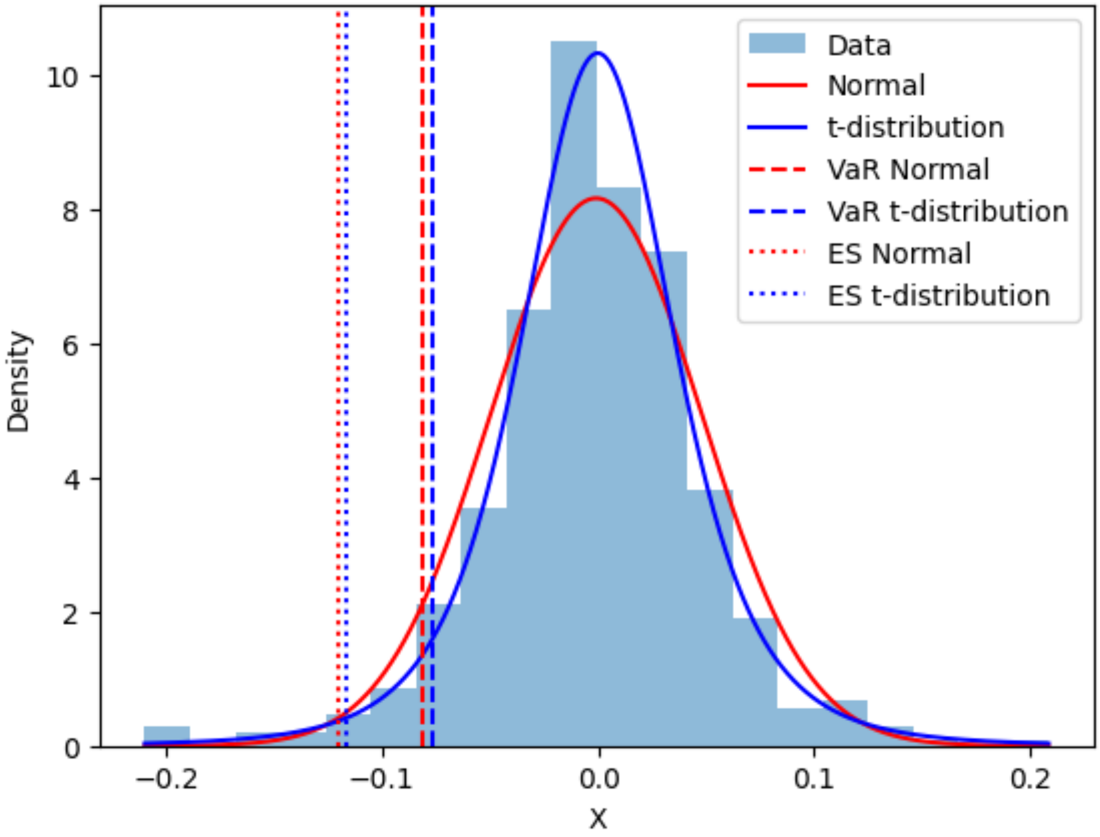
- First fit model with normal and t distribution in stats package, to calculate the VaR, and ES value,

Then, use function that calculate VaR and ES value in our library to get the values:



```
Normal distribution:
VaR: 0.08125
ES: 0.12011
t-distribution:
VaR: 0.07648
ES: 0.11678
```

We use matplotlib to plot the distribution , ES and VaR as well as the model fitting.



From the visualization we can see the t-distribution descript more accurately on shape of the stock. Moreover, we could conclude that ES is always greater than VaR (when we add a negate on them). And then we can see that Normal distribution tend to be more conservative than t-distribution because the red lines are always on the left of the blue lines.

The reason for this could be the fact that the standard deviation of t-distribution is smaller, and the simulated data is more concentrated to the center.

Problem 2

In your main repository, create a Library for risk management. Create modules, classes, packages, etc as you see fit. Include all the functionality we have discussed so far in class. Make sure it includes

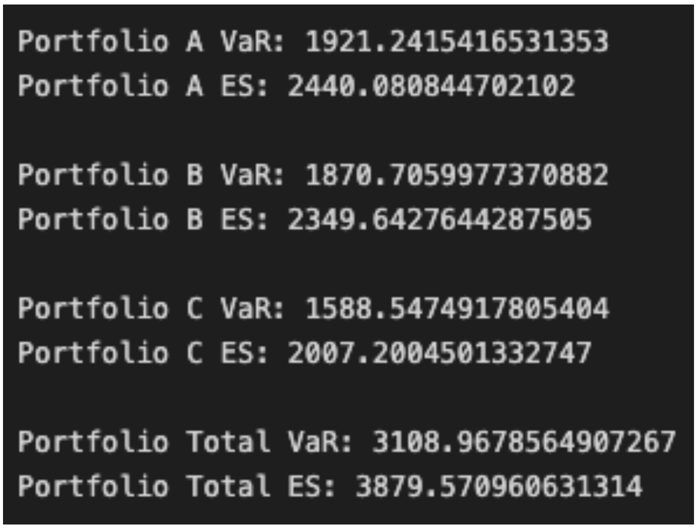
1. Covariance estimation techniques.
2. Non PSD fixes for correlation matrices
3. Simulation Methods
4. VaR calculation methods (all discussed)
5. ES calculation

All test case shown to be correct.

Problem 3

- we are simulating the returns and values of different portfolios using the t-distribution.

- First, we read in the portfolio and daily price data from two CSV files. We then calculate the daily returns for each stock and store them in a DataFrame called return_data.
-
- Next, we group the portfolio data by stock and calculate the total holdings for each stock. We then append a row to the portfolio DataFrame with the total holdings for all stocks, and set the portfolio column to "Total".
-
- We then loop through each stock in return_data and calculate the parameters for a t-distribution. We store these parameters in a dictionary called t_param. We also generate 10,000 simulated returns for each stock and store them in a list called sim_data.
-
- After all the simulated data has been generated, we create a DataFrame called sim_returns with the simulated returns. We also create a DataFrame called current_prices with the current prices of each stock.
-
- Finally, we loop through each portfolio and calculate the simulated returns and values for that portfolio. We calculate the VaR and ES for each portfolio using a risk management module called risk_mgmt. The results are:



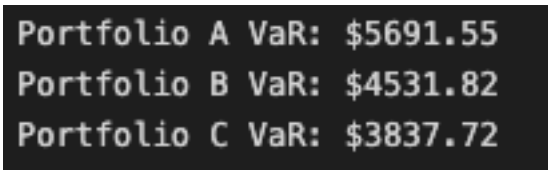
```
Portfolio A VaR: 1921.2415416531353
Portfolio A ES: 2440.080844702102

Portfolio B VaR: 1870.7059977370882
Portfolio B ES: 2349.6427644287505

Portfolio C VaR: 1588.5474917805404
Portfolio C ES: 2007.2004501332747

Portfolio Total VaR: 3108.9678564907267
Portfolio Total ES: 3879.570960631314
```

Compare the value we get from last week:



```
Portfolio A VaR: $5691.55
Portfolio B VaR: $4531.82
Portfolio C VaR: $3837.72
```

We can see the value is much smaller for our generalized T model of the portfolio. It allows the trader take more risk at a same price level.