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.

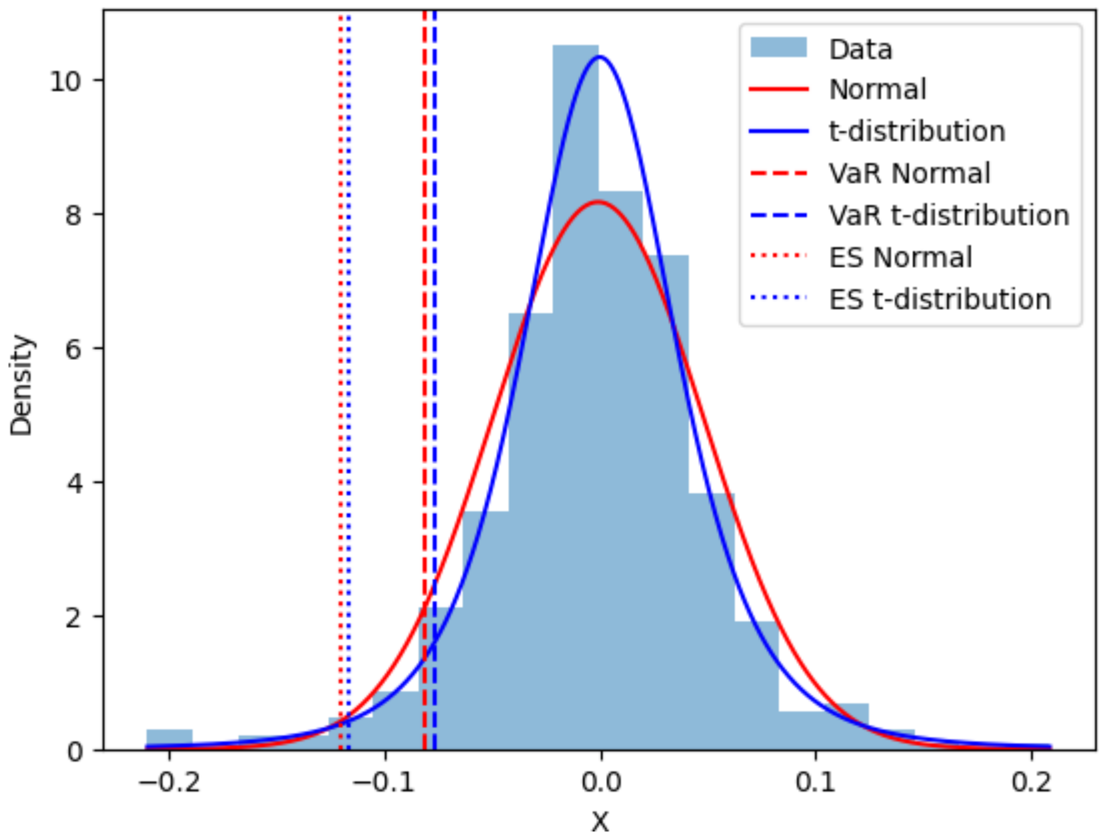
Overlay the graphs the distribution PDFs, VaR, and ES values. What do you notice? Explain the differences.

- 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 have developed a Python program to calculate the Value at Risk (VaR) and Expected Shortfall (ES) of multiple portfolios using a Generalized T model and a copula-based simulation method. The program uses data from two input files, Portfolio.csv and DailyPrices.csv, and is divided into several logical sections.
- First, we load the data into pandas dataframes and define the portfolios we want to analyze. Then, for each portfolio, we calculate its value, return, and holdings by calling a function from a separate module.
- Next, we fit a Generalized T distribution to each stock in the portfolio using its historical returns. We store the parameters and the cumulative distribution function (CDF) of each stock in pandas dataframes.
- We then use a principal component analysis (PCA) to generate correlated samples from the joint distribution of the stocks. We simulate 10,000 samples using a seed of 101 for reproducibility. We transform these samples back to the original space of stock returns using the inverse CDF of the Generalized T distribution.
- Finally, we calculate the simulated prices of each portfolio using the simulated returns and the portfolio holdings. We compute the VaR and ES of each portfolio using the simulated prices and the mean portfolio value, and print the results for each portfolio as following

```
Simulating with 22 PC Factors: 95.01 % total variance explained
For Portfolio A, the VaR is: 7940.895693258732
For Portfolio A, the ES is: 10754.943321283818

Simulating with 23 PC Factors: 95.47 % total variance explained
For Portfolio B, the VaR is: 6711.417545498523
For Portfolio B, the ES is: 9044.418040979726

Simulating with 23 PC Factors: 95.27 % total variance explained
For Portfolio C, the VaR is: 5600.537946650584
For Portfolio C, the ES is: 7773.806940833019

Simulating with 52 PC Factors: 95.03 % total variance explained
For Portfolio ALL, the VaR is: 19910.804313675384
For Portfolio ALL, the ES is: 27243.953659470422
```

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 larger for our generalized T model of the portfolio. It is a more conservative risk-management tool.