Week02 Assignment by Angela Liang

Problem 1

Given the dataset in problem1.csv:
a. calculate the first four moments values by using normalized formula in the "Week1 - UnivariateStats".
b. calculate the first four moments values again by using your chosen statistical package.
c. Is your statistical package functions biased? Prove or disprove your hypothesis respectively.
Explain your conclusion.

My approach for this question:

For the problem, I generate a standard normal distribution so the skewness and kurtosis are expected to be 0. In the code below, I first parse the dataset from csv file and use the data to generate results using normalized formula from the lecture, and use the standard package scipy to calculate the same. The results are exactly the same or very close. Then I use t-test to test at significance level of 95% to see if my package is biased or not.

Results:

Normalized Formula (Statistical Package)
Mean 1.049 (1.049)
Variance 5.4272 (5.4218)
Skew 0.8806 (0.8806)
Kurtosis 23.1222 (23.1222)
Mean diff = 0.0
Variance diff = 0.0054
Skewness diff = 0.0
Kurtosis diff = 0.0\

T-test results for each moment:
Mean: T-statistic = -0.0482, p-value = 0.9615
Variance: T-statistic = -22.368, p-value = 0.0
Skewness: T-statistic = 1.9568, p-value = 0.0505
Kurtosis: T-statistic = -30.1013, p-value = 0.0

Problem 1 Conclusion

Upon testing, I have the following results: We fail to reject null hypothesis and conclude that the difference is not significant. The package is unbiased for mean calculation. We reject null hypothesis and conclude that the difference is significant. The package is biased for variance calculation. We reject null hypothesis and conclude that the difference is significant. The package is biased for skewness calculation. We reject null hypothesis and conclude that the difference is significant. The package is biased for kurtosis calculation.

The results of the four t-tests show that only the mean calculation is unbiased and I conclude that the statistical package is biased for variance, skewness, and kurtosis calculation.

Problem 2

First, install necessary packages if needed using the code below

```
pip install statsmodels
```

a. MLE method and compare with OLS

Estimated Betas in MLE: [-0.08737946 0.77523121]
Estimated Sigma in MLE: 1.0037674485926553
Estimated Betas in OLS: [-0.08738446 0.7752741]
Estimated Sigma in OLS: 1.003756319417732\

I find that the beta and standard error calculated using OLS and MLE is the same. Slight difference is found might be because the optimization process in MLE might not converge exactly to the same solution. It could be due to any computational rounding in the process or the initial guess (assumtion) made in MLE method.

-

b. MLE for t-distribution of error

First, I use MLE method under normality assumption, and then use it for t-distribution error. Then, I compare goodness-of-fit using AIC and BIC, and whichever has the lower AIC/BIC is a better fit.

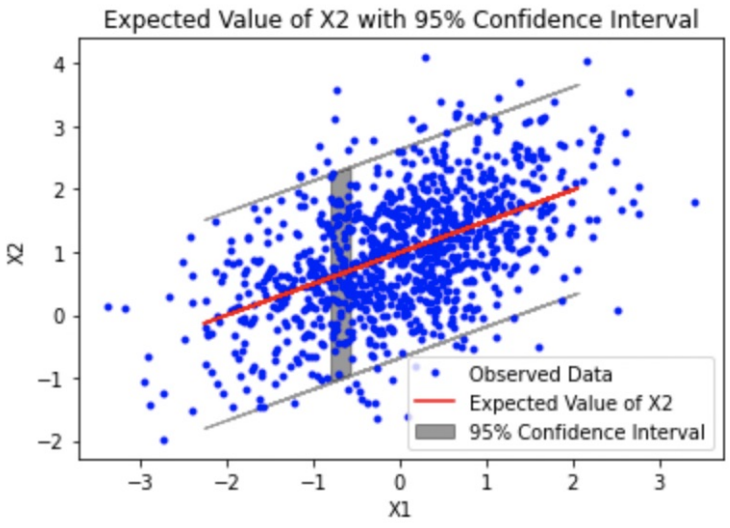
```
k_normal: 3
k_t: 4
AIC (Normal): -563.0751264822612
BIC (Normal): 1146.068007566218
AIC (t-distribution): -554.58680635997
BIC (t-distribution): 1149.072922272167
Normal distribution error model has a lower AIC.
Normal distribution error model has a lower BIC.
Thus, normal distribution is a better fit.
```

Conclusion

Based on the comparison of AIC and BIC, the model that has a lower AIC or BIC is a better fit. Thus, I conclude that normal distribution is a better fit.

c. Distribution of X2 and plots

I begin by parsing the dataset from two separate CSV files, problem2_x.csv and problem2_x1.csv, which contain the values of X1 and X2, respectively. Then I define a negative log-likelihood function for the multivariate normal distribution. This function takes parameters such as means (mu1, mu2), standard deviations (sigma1, sigma2), and correlation coefficient (rho) as inputs. MLE Optimization: Using the minimize function from SciPy, I optimize the negative log-likelihood function to estimate the parameters (mu1, mu2, sigma1, sigma2, rho) that maximize the likelihood of observing the data. After optimization, I extract the Maximum Likelihood Estimation (MLE) parameters to obtain the estimated means, standard deviations, and correlation coefficient. I plot the observed data points of X1 and X2 along with the expected value of X2 given each observed value of X1. This allows me to visualize the relationship between the two variables. Additionally, I compute the 95% confidence interval for X2 based on the estimated parameters. This provides insights into the uncertainty associated with the expected value of X2. See the visualization of the expected value of X2 along with its 95% confidence interval on a scatter plot below.



Extra credit

Assume $\varepsilon \sim N(0, \sigma^2 I_n)$. using MLE, derive estimators for β and σ^2.

$$Y = X\beta + \varepsilon$$

$$\frac{\partial}{\partial \hat{\beta}} (y - X\hat{\beta})^T (y - X\hat{\beta}) = \frac{\partial}{\partial \hat{\beta}} (y^T y - y^T X\hat{\beta} - \hat{\beta}^T X^T y + \hat{\beta}^T X^T X\hat{\beta})$$

$$= -y^T X - y^T X + 2\hat{\beta}^T X^T X$$

$$= -2y^T X + 2\hat{\beta}^T X^T X$$

$$= -2X^T y + 2X^T X\hat{\beta}$$

Provided that $X^T X$ is invertible:

$$\hat{\beta} = (X^T X)^{-1} X^T y$$

$$\frac{\partial L(\beta, \sigma^2 | Y)}{\partial \sigma^2} = -\frac{n}{2}\left(\frac{1}{\sigma^2}\right) + \frac{1}{2(\sigma^2)^2}(Y - X\beta)^T (Y - X\beta) = 0$$

solve for MLE:

$$\frac{n}{2}\left(\frac{1}{\hat{\sigma}^2}\right) = \frac{1}{2(\hat{\sigma}^2)^2}(Y - X\hat{\beta})^T (Y - X\hat{\beta})$$

$$n\hat{\sigma}^2 = (Y - X\hat{\beta})^T (Y - X\hat{\beta})$$

$$\hat{\sigma}^2 = \frac{(Y - X\hat{\beta})^T (Y - X\hat{\beta})}{n} = \frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{SSR}{n}.$$

Thus,

$$\begin{cases} \hat{\beta} = (X^T X)^{-1} X^T y \\ \hat{\sigma}^2 = \dfrac{(Y - X\hat{\beta})^T (Y - X\hat{\beta})}{n} \end{cases}$$

Problem 3

Fit the data in problem3.csv using AR(1) through AR(3) and MA(1) through MA(3), respectively. Which is the best of fit?

My approach:

- Parse data from CSV
- loop over each combination of p and q
- use python package tsa.arima.model to auto fit with predefined p and q
- print the top three models with lower AIC and BIC and compare and decide on the best model

NOTE: Since I'm not sure if this question means a combination of ARMA or just AR and MA separately, I'm doing a AR and MA separate calculation (just AR or just MA), and then also doing a calculation that contains all possible cases (nested loop with all combinations of p and q).

```
Results Table:
   p  q          AIC          BIC
0  1  0  1269.580752  1281.555146
1  2  0  1233.539326  1249.505184
2  3  0  1141.345974  1161.303296
3  0  1  1211.564179  1223.538573
4  0  2  1197.168232  1213.134090
5  0  3  1198.683206  1218.640528
Top three models with the lowest AIC:
   p  q          AIC          BIC
2  3  0  1141.345974  1161.303296
4  0  2  1197.168232  1213.134090
5  0  3  1198.683206  1218.640528

Top three models with the lowest BIC:
   p  q          AIC          BIC
2  3  0  1141.345974  1161.303296
4  0  2  1197.168232  1213.134090
5  0  3  1198.683206  1218.640528
```

Conclusion (if doing AR and MA separately)

By comparing the AIC and BIC results for all the possible combinations, I conclude that AR(3) is the best model because it has the lowest AIC and BIC.

```
Results Table:
    p  q          AIC          BIC
0   0  0  1294.515823  1302.498752
1   0  1  1211.564179  1223.538573
2   0  2  1197.168232  1213.134090
3   0  3  1198.683206  1218.640528
4   1  0  1269.580752  1281.555146
5   1  1  1202.197907  1218.163765
6   1  2  1199.120857  1219.078180
7   1  3  1185.123358  1209.072146
8   2  0  1233.539326  1249.505184
9   2  1  1185.508809  1205.466131
10  2  2  1174.021340  1197.970127
11  2  3  1156.199414  1184.139666
12  3  0  1141.345974  1161.303296
13  3  1  1143.158773  1167.107560
14  3  2  1144.911214  1172.851466
15  3  3  1146.001160  1177.932876
Top three models with the lowest AIC:
    p  q          AIC          BIC
12  3  0  1141.345974  1161.303296
13  3  1  1143.158773  1167.107560
14  3  2  1144.911214  1172.851466

Top three models with the lowest BIC:
    p  q          AIC          BIC
12  3  0  1141.345974  1161.303296
13  3  1  1143.158773  1167.107560
14  3  2  1144.911214  1172.851466
```

Conclusion (if doing ARMA)

Therefore, even if the question is asking for a combination of ARMA model, that is, ARIMA, the best model would still be ARIMA(3,0,0) as it has the lowest AIC and BIC