Time Series Homework4

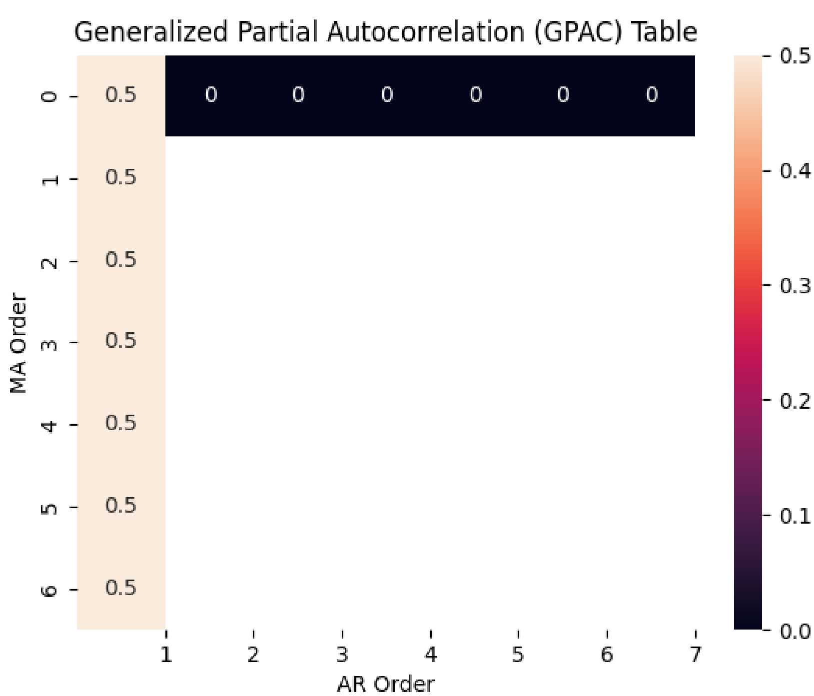
Rubrics:

The main purpose of this LAB is to implement the GPAC array covered in lecture using Python program and test the accuracy of your code using an ARMA(na,nb) model**. It is permitted to simulate the ARMA process using the statsmodels**. Everyone needs to write their own GPAC code that generates GPAC table for various numbers of rows and columns. Fix the numpy random seed to 123

Question 1 to 4 are about writing a function based on which subsequent questions will be answered. Hence, the completeness and the correctness of the answers to these questions can be seen from the solution to the subsequent questions.

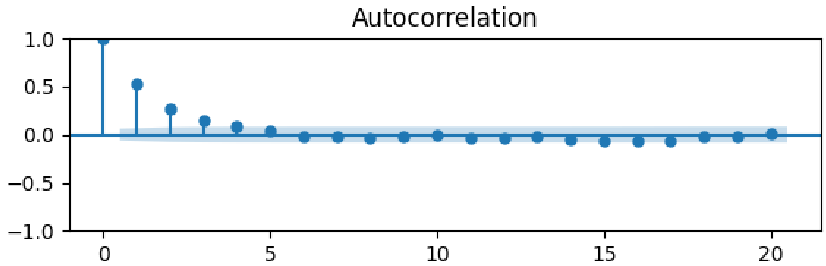
Question 5 based on the process given in Question 3

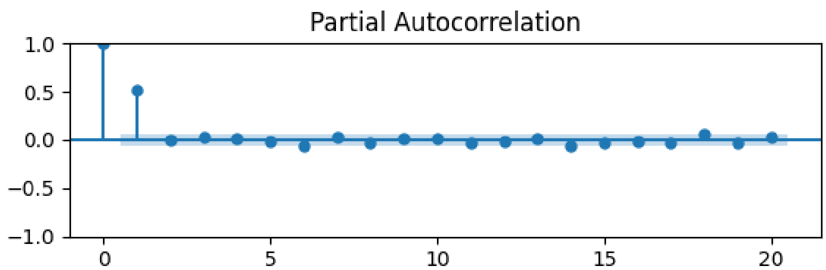
a. For ARMA(1,0) with AR coefficients [1,-0.5], and MA coefficients [1] (including the coefficients of y(t), and e(t) that is 1), the GPAC table is as follows:



Since the there is a column with constant values adjacent to a row filled with zeros, we can then conclude from the table that the order of MA is 0, while the order of the AR process is 1.

Question 6 – ACF and PACF plot based on the process mentioned in Question 3

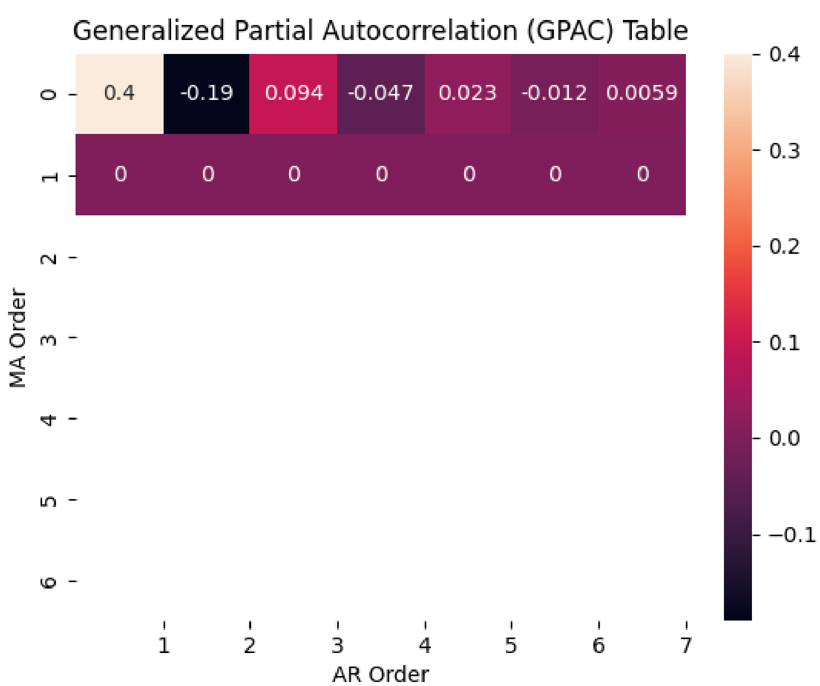




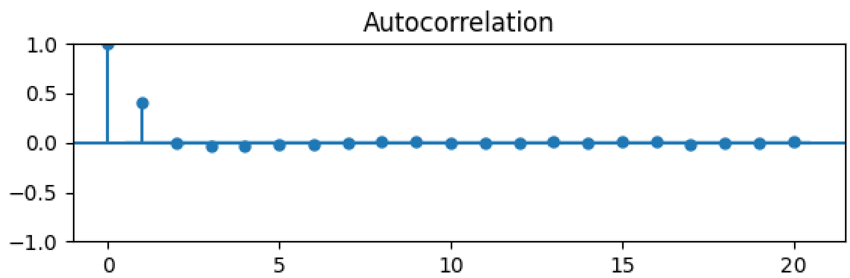
Question 7 – Repeat steps 3, 4, 5, and 6 for following 7 examples with only 10,000 samples

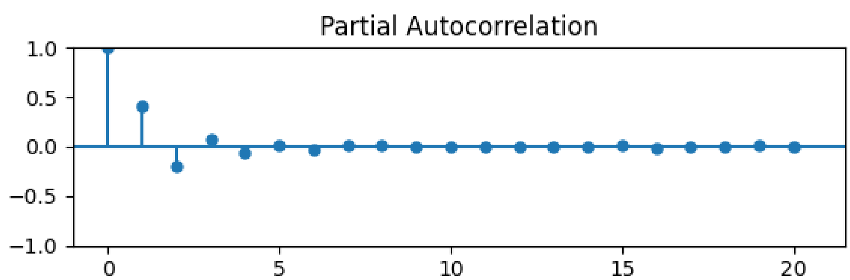
Example2:

Example 2: ARMA (0,1): y(t) = e(t) + 0.5e(t-1)



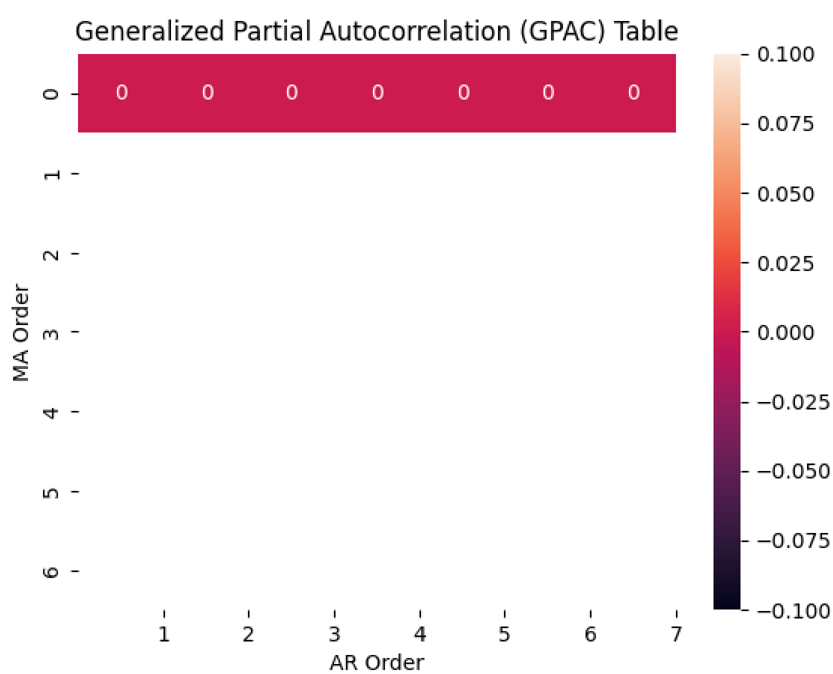
Since there is not a column with constant number, this is a pure MA process. Also, the order of the MA process is identified to be 1 as the row filled with zeros start from 1st order. Hence, the MA order nb=1.

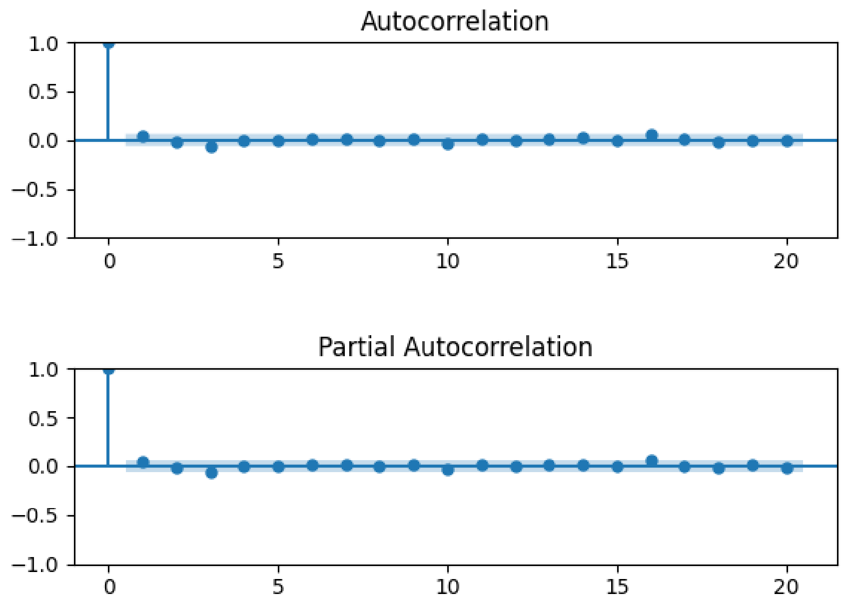




Example3:

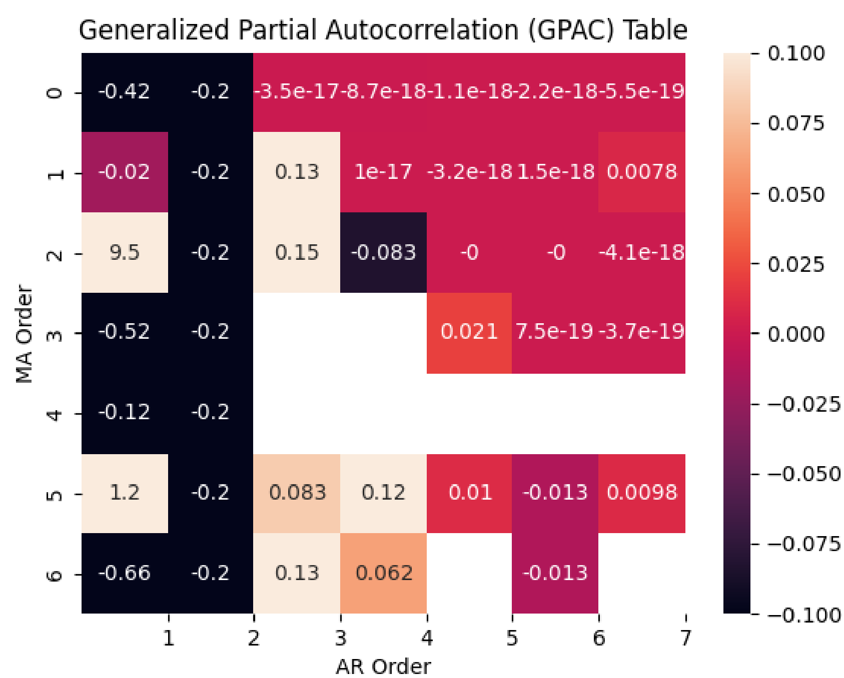
Example 3: ARMA (1,1): y(t) + 0.5y(t-1) = e(t) + 0.5e(t-1)



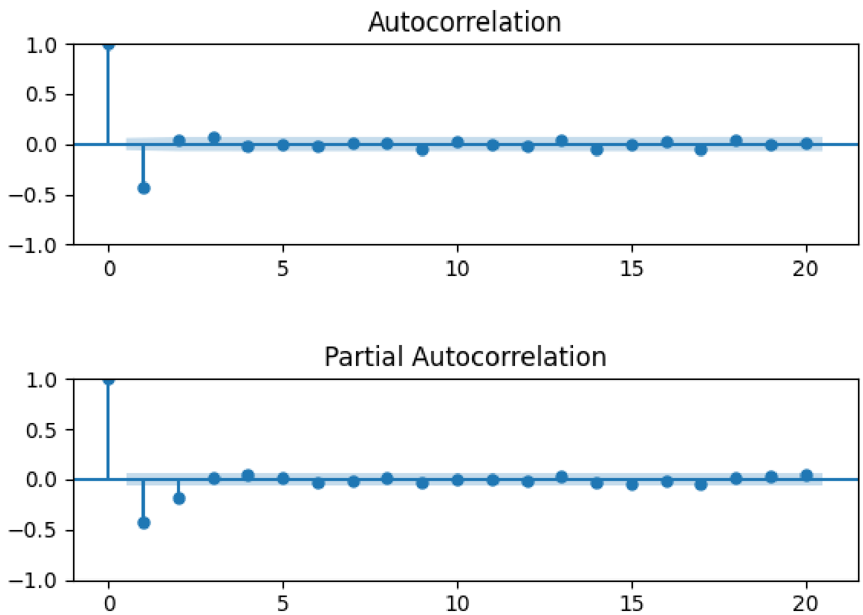


Example 4:

Example 4: ARMA (2,0): $y(t) + 0.5y(t-1) + 0.2y(t-2) = e(t)$

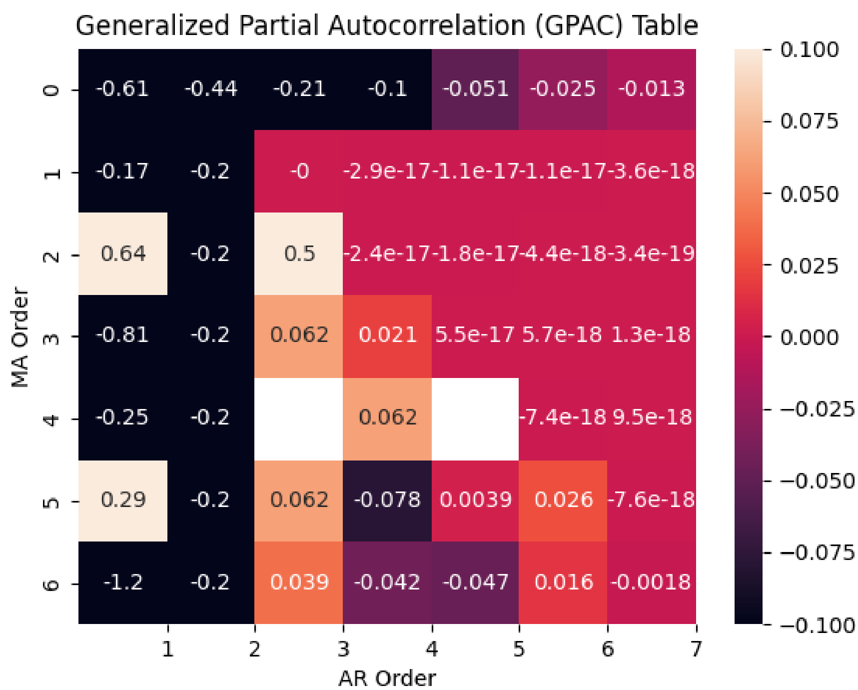


The above table shows the 2nd column i.e., na=2 having constant values adjacent to a row that is filled with zeros. Since values in the adjacent row are extremely close to zero such as an epsilon, they are still considered zeros. Hence, the order of the process is ARMA(na=2, nb=0). Here -0.2 starts from 0th row that represents the nb, which is the order of the MA process.

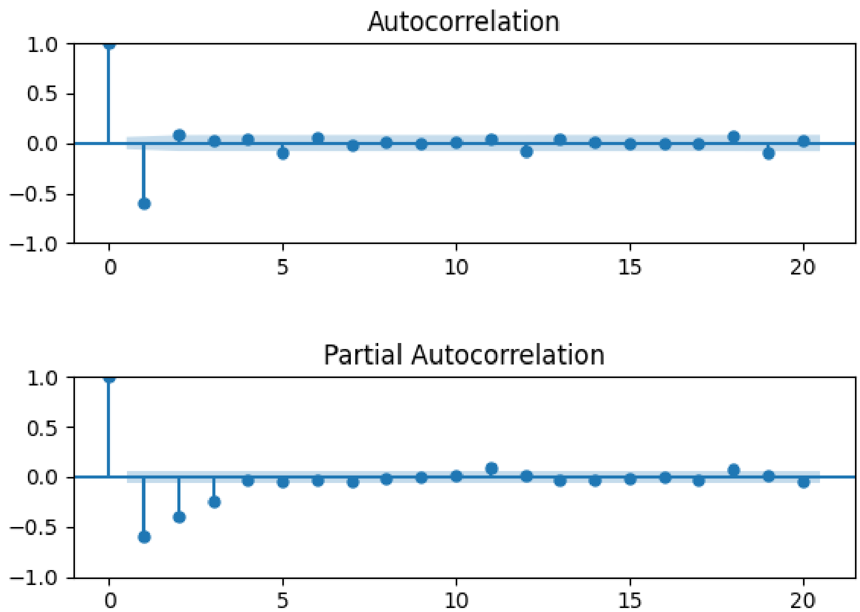


Example 5

Example 5: ARMA (2,1): y(t) + 0.5y(t-1) + 0.2y(t-2) = e(t) - 0.5e(t-1)

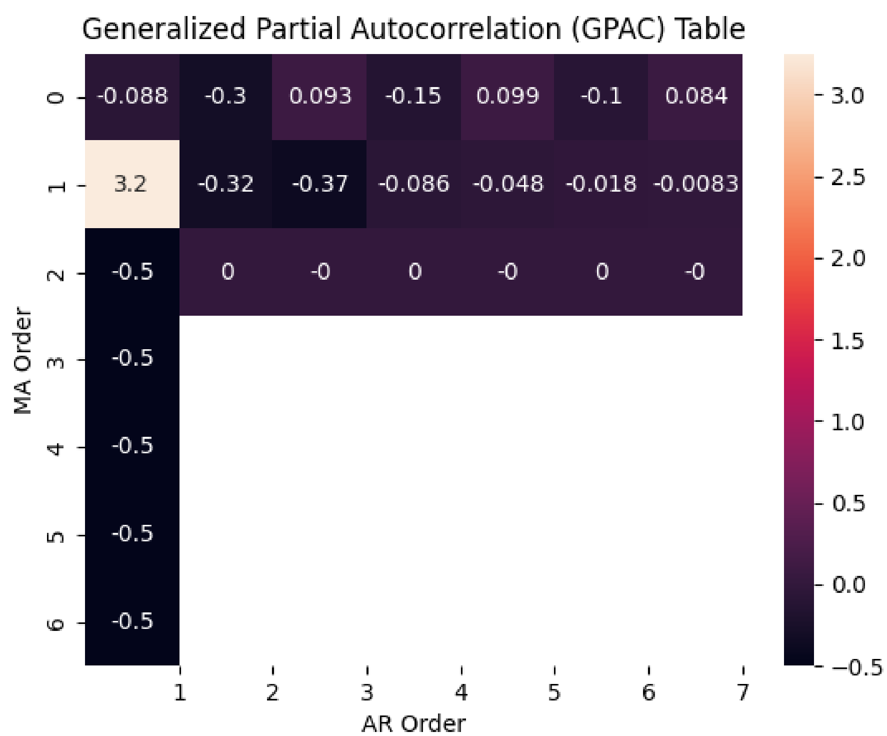


In the table above, -0.2 starts from 1st row and 2nd column that represents order of the MA, and AR process respectively. ARMA(na=2, nb=1). The field that starts the pattern is positioned adjacent to the row that is filled with zeros.

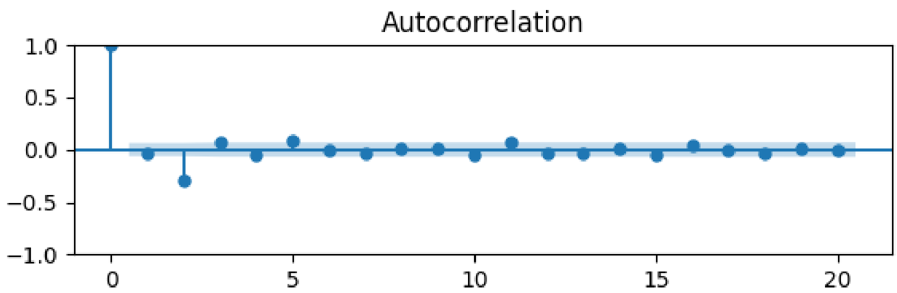


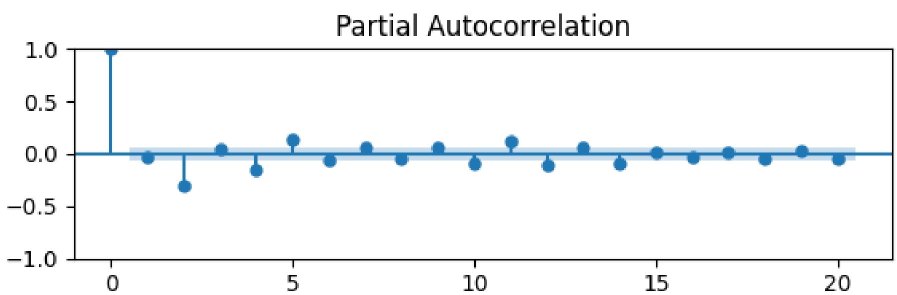
Example 6

Example 6: ARMA (1,2): y(t) + 0.5y(t-1) = e(t) + 0.5e(t-1) - 0.4e(t-2)



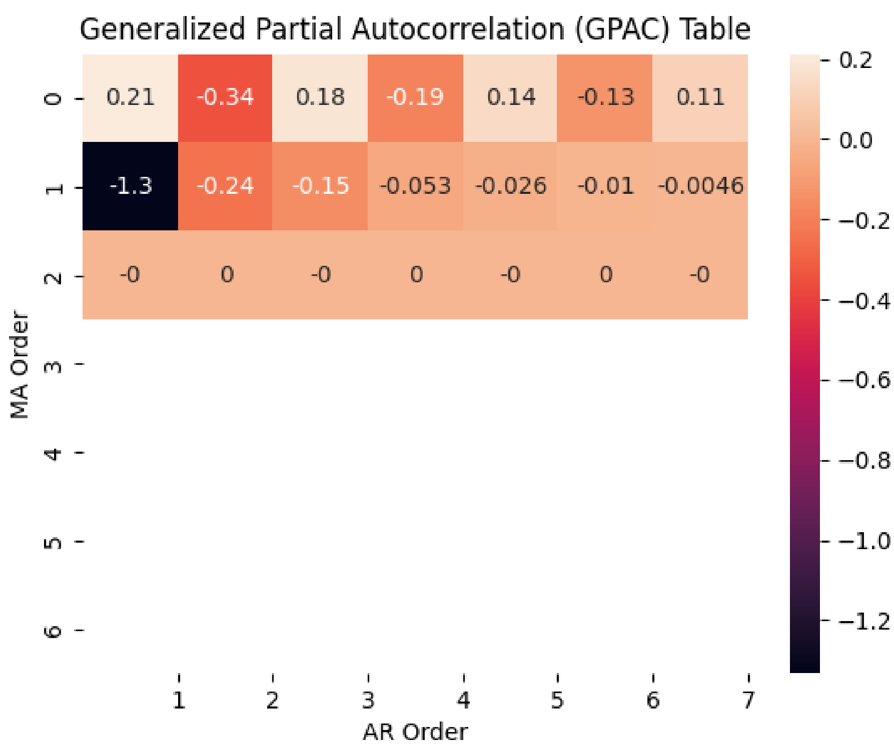
Here the column that exhibits the pattern is 1 i.e., the order of the AR process is 1, while the row where the pattern begins is 2, which represents the order of the MA process. Hence the estimated order of the ARMA is na=1, and nb=2



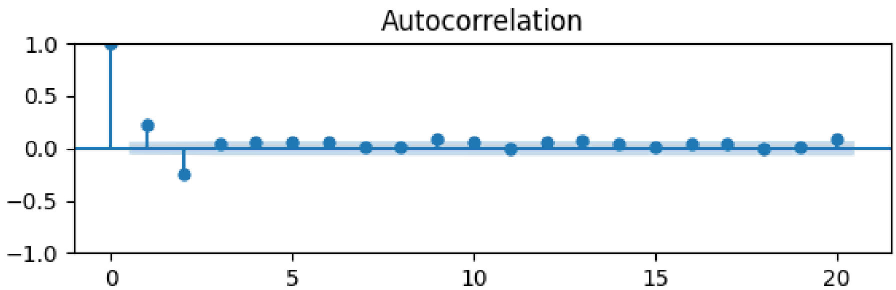


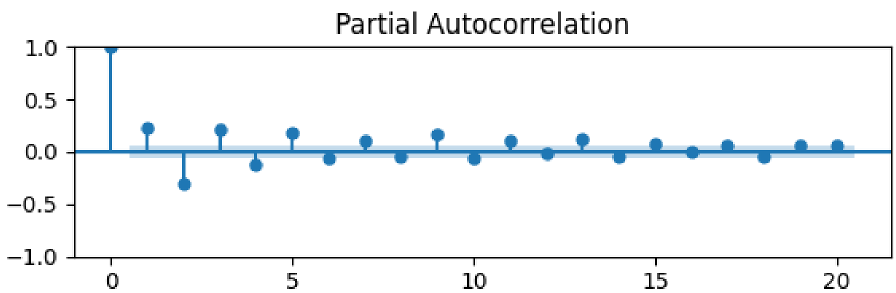
Example 7

Example 7: ARMA (0,2): y(t) = e(t) + 0.5e(t-1) - 0.4e(t-2)



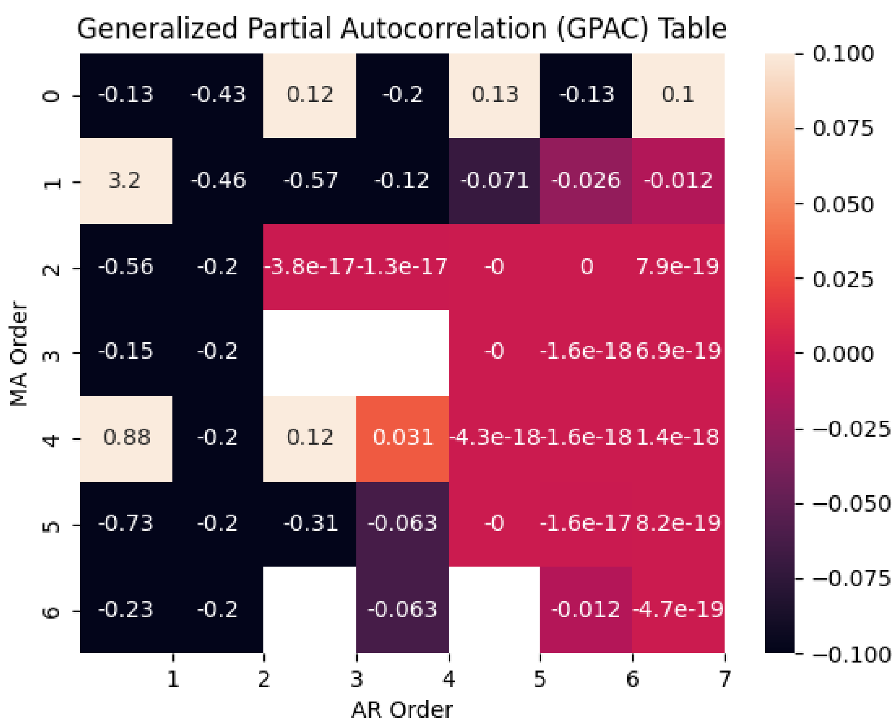
The above table shows a row of zeros with no single column showing a constant being repeated. This means what we have is a pure MA process, and the row at which the zeros begin to show represents the order of the MA process. Here, the order of the MA process i.e., nb=2. Hence ARMA(0,2)



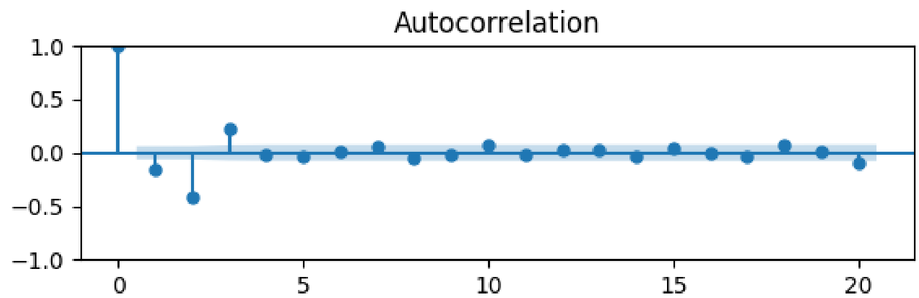


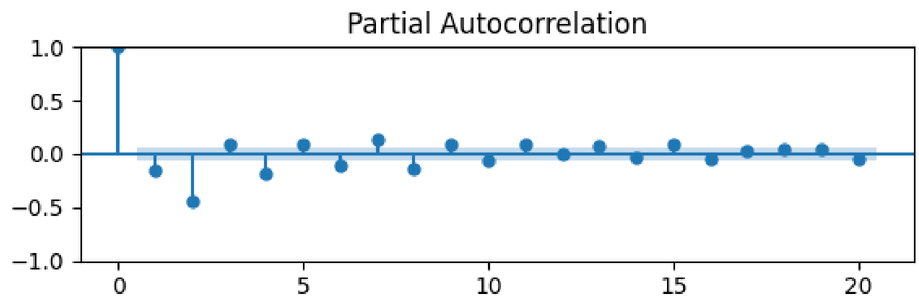
Example 8

Example 8: ARMA (2,2): y(t)+0.5y(t-1) +0.2y(t-2) = e(t)+0.5e(t-1) - 0.4e(t-2)



In the above table, there is a column (2) that has constant values beginning from the 2nd row. And the field that begins the pattern is positioned adjacent to the row that is filled with zeros. Hence, the order of the ARMA process is na=2, and nb=2.





Question 8 – Comments on ACF, and PACF

For Example1:

ARMA(1, 0) – a pure AR process.

Here, the PACF correctly shows the order of the AR process as 1, and the ACF exhibits tail-off pattern. Assuming tail-off pattern does not allow us to infer the order of any process, we can take it for granted that what we are dealing with is purely an AR process on the basis of both ACF, and PACF plots.

For Example2:

ARMA(0,1) – a pure MA process.

Although the ACF plot clearly shows a spike at the 1^{st} lag that represents the MA order, the PACF also shows some spikes at different lags and is not very clear to infer the order.

For Example3:

ARMA(1,1) process.

Here both the ACF, and the PACF does not exhibit any pattern and it was also the same case with GPAC. I believe the reason for this is the coefficients being the same for both the process has cancelled each other's (i.e., MA and AR) effect in calculating the autocovariance and as a result GPAC is unable to allow us identify the order of the process. I believe the cancellation makes the generated data a white noise, which is why both the ACF, and the PACF

For Example4:

ARMA(2,0) – a pure AR process.

Once again, the PACF lets us identify the correct order of AR process as 2 while the ACF shows a spike at 1^{st} lag.

For Example5:

ARMA(2,1) process.

Here, the ACF lets us identify the correct order of the MA process as 1, but the PACF exhibits tail-off pattern that does not allow us to identify the AR process.

For Example6:

ARMA(1,2) process.

Here, the ACF appears to show the correct order of the MA process, while the PACF shows a significant spike at lag 2 that is an incorrect order of AR process.

For Example7:

ARMA(0,2) – a pure MA process.

Here, the ACF shows significant spikes at both 1^{st} and 2^{nd} lag so the order of the MA process is 2. Observing the PACF plot, it appears to exhibit the tail-off pattern that is not a good representative of an AR process. Hence, we can conclude what we are dealing with is a pure MA process and the order is ARMA(0,2).

For Example8:

ARMA(2,2) process.

Here, the PACF shows the correct order of the AR process by producing significant spike at 2^{nd} lag. Although, the ACF did showed a significant spike at 2^{nd} lag, it still showed another spike at 3^{rd} lag, which is inaccurate.

General observations:

1. When the order of both the processes are equal, then one of the plots (either ACF or PACF) is often inaccurate.

2. When the process is pure MA, then PACF exhibits tail-off pattern while the ACF allows to identify the correct order of the MA process.
3. When the process is pure AR, then ACF exhibits tail-off pattern while the PACF allows to identify the correct order of the AR process.
4. When the order of both the process is equal, and share the same coefficients, both ACF, and the PACF displays an impulse function that only spikes at 0 lag, and insignificant for the other lags. It is my understanding the coefficients cancel opposing process's effects.
5. When both AR and MA is involved, and do not share the same coefficients, then one of the plots is inaccurate.

Hence, it is advisable to use GPAC along with ACF, and PACF regardless of the expectation of whether AR and MA are both involved in having generated the data.