Problem 1

Remember from last week we discussed that skewness and kurtosis functions in statistical packages are often biased. Is your function biased? Prove or disprove your hypothesis.

Analysis

In Python, I used functions kurtosis() and skew() from package scipy.stats to calculate the kurtosis and skewness. They are both biased in definition. However, I proved function kurtosis() is biased in small sample size at the significance of 5%, but when sample size going large, the bias will tend to weaken. I didn't prove that function skewness() is biased at the significance of 5% at sample sizes 1000, but not in size 100 and 10000.

I sample large amount of standardized random normal values, sample size is respectively 100, 1000, 10000. Then I calculated the kurtosis and skewness using function kurtosis() and skewness(). I sample the kurtosis and skewness by repeating the above steps for 1000 times. Using these 1000 samples as sample data, I did a T test to test whether the kurtosis and skewness calculated by kurtosis() and skewness() are significantly different from 0 which is both the kurtosis and skewness of normal distribution data.

The null hypothesis I set is function kurtosis() and skewness() is unbiased. And the significance level threshold is 5%.

		100	**1000**	**10000**
kurtosis	T statistic	-3.2466	-2.0181	-0.9668
	p-value	0.0012	0.0438	0.3338
	H0	Reject	Reject	
skewness	T statistic	1.2929	2.8996	-0.0330
	p-value	0.1963	0.0038	0.9736
	H0		Reject	

It can be seen that when sample size is comparatively small, function kurtosis() and skewness() are both tend to be biased.(in size 100, t statistic of skewness is comparatively large). They tend to converge to be unbiased as sample size grows. So, function kurtosis() and skewness() are both biased.

Problem 2

Problem 2 Fit the data in problem2.csv using OLS and calculate the error vector. Look at its distribution. How well does it fit the assumption of normally distributed errors?

Fit the data using MLE given the assumption of normality. Then fit the MLE using the assumption of a T distribution of the errors. Which is the best fit?

What are the fitted parameters of each and how do they compare? What does this tell us about the breaking of the normality assumption in regards to expected values in this case?

Analysis

Fit the data in problem2.csv using OLS, the results are as follow:

OLS Regression Results

Dep. Variable:	y	R-squared:	0.195
Model:	OLS	Adj. R-squared:	0.186
Method:	Least Squares	F-statistic:	23.68
Date:	Fri, 27 Jan 2023	Prob (F-statistic):	4.34e-06
Time:	18:45:25	Log-Likelihood:	-159.99
No. Observations:	100	AIC:	324.0
Df Residuals:	98	BIC:	329.2
Df Model:	1		
Covariance Type:	nonrobust		

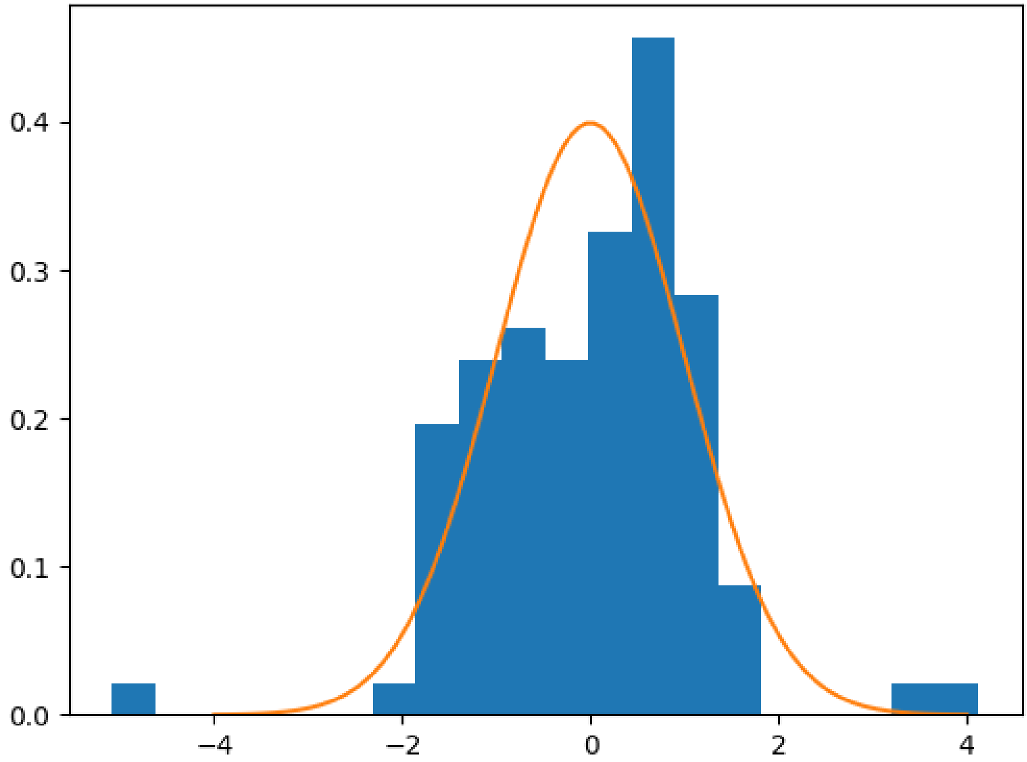
	coef	std err	t	P>|t|	[0.025	0.975]
const	0.1198	0.121	0.990	0.325	-0.120	0.360
x	0.6052	0.124	4.867	0.000	0.358	0.852

Omnibus:	14.146	Durbin-Watson:	1.885
Prob(Omnibus):	0.001	Jarque-Bera (JB):	43.673
Skew:	-0.267	Prob(JB):	3.28e-10
Kurtosis:	6.193	Cond. No.	1.03

Mean of error vector is -2.1094237467877975e-17
Variance of error vector is 1.4361484854062607
Skewness of error vector is -0.267266585528796

Kurtosis of error vector is 3.1931010009568785
It can be seen from the result above for normal distribution, skewness and kurtosis should be 0, there is a large digress from 0, where skewness of error vector is -0.2673 and kurtosis of error vector is 3.1931.

After plotting the distribution and the normal distribution curve in one picture,



It can be seen that there is huge difference between the error distribution and the normal distribution. The assumption of normally distributed errors is hardly held.

Fit the data using MLE given the assumption of normality and a T distribution of the errors, I used R-squared and information criteria AIC and BIC to measure fitness.

	OLS	**MLE-Normal**	**MLE_T**
Const	0.1198	0.1198	0.1426
X	0.6052	0.6052	0.5576
R_squared	0.195	0.1946	0.1934
AIC	324.0	324.0	314.9
BIC	329.2	329.2	320.2

It can be seen from the table that OLS and MLE-Normal have the same results, MLE-T is different. Compared with MLE-Normal, MLE-T has relatively the same R-squared but lower AIC and BIC value. We can say from this result that fitting the data using

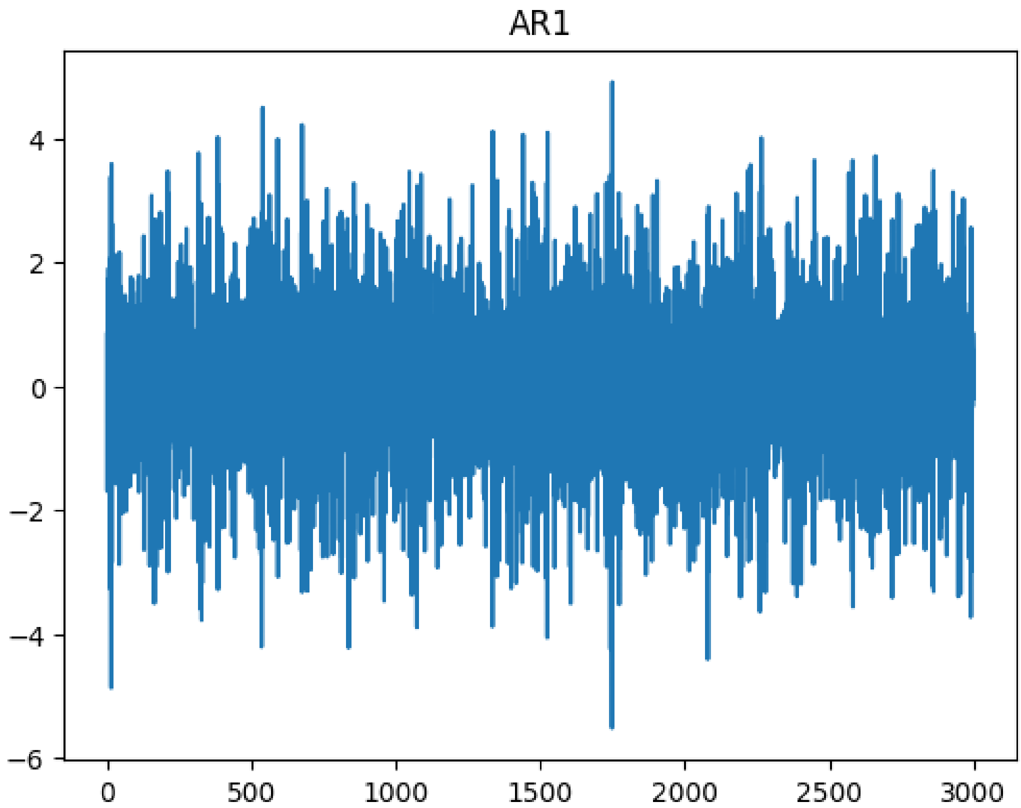
MLE given the assumption of a T distribution of the errors is best fit the data.
In this example, obviously, T distribution is better than normality assumption. OLS and MLE both use the assumption of normal distribution, which can be inappropriate in some cases.

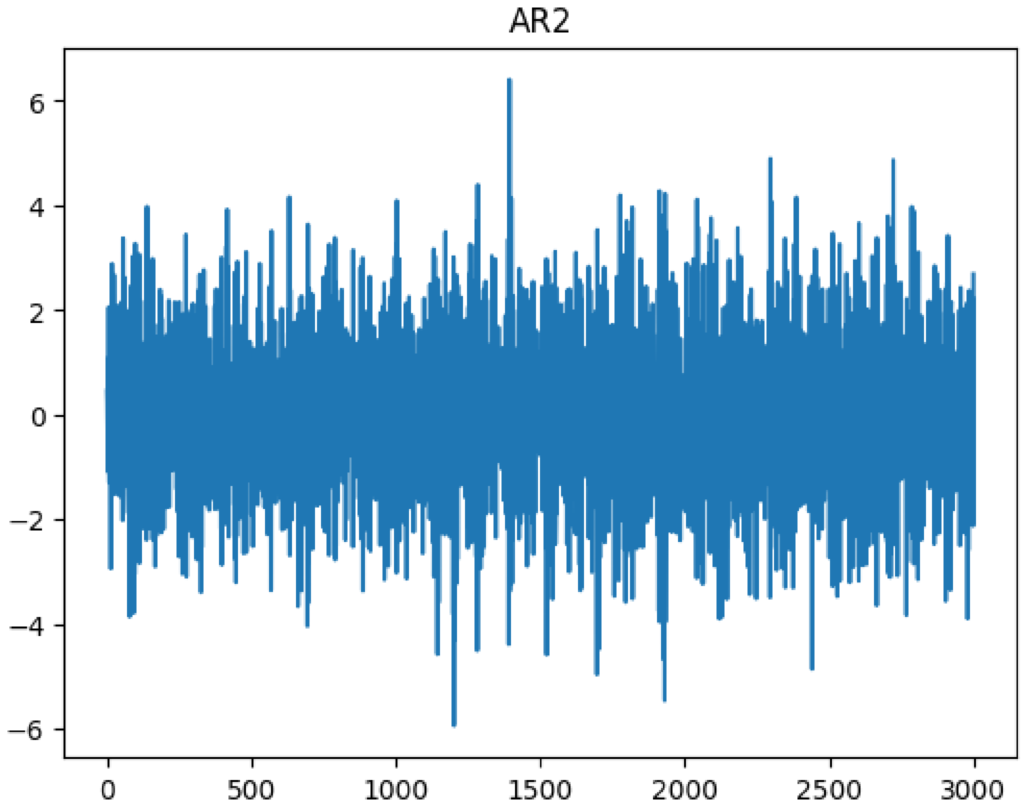
Problem 3

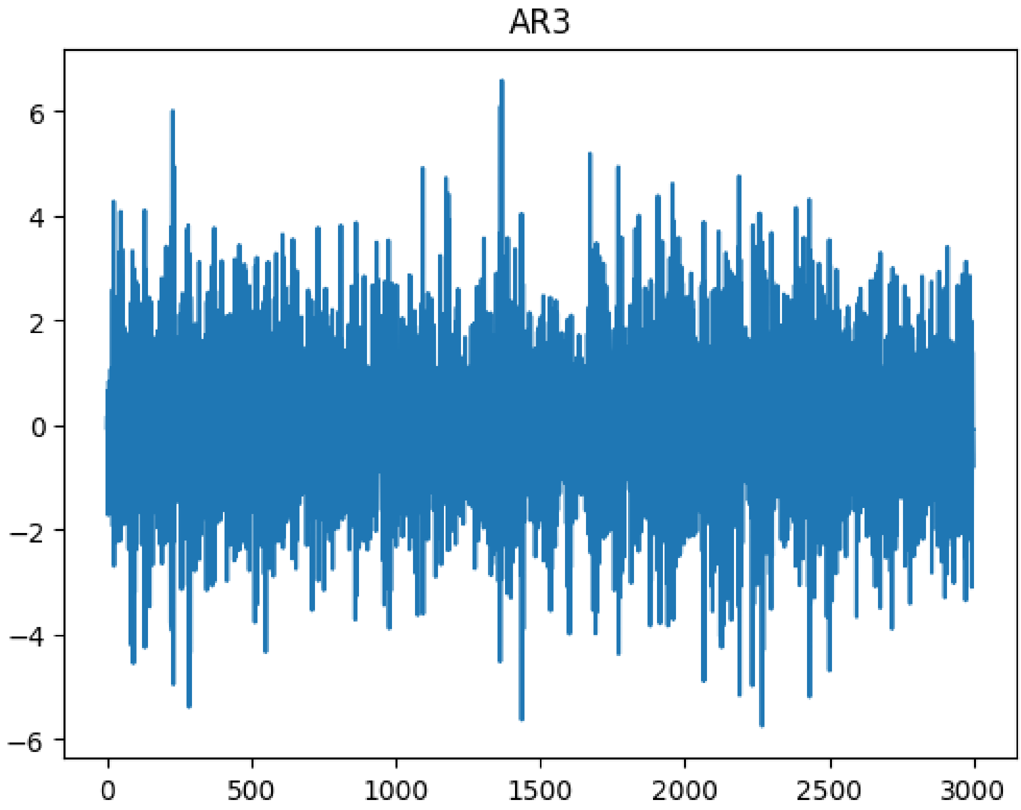
Simulate AR(1) through AR(3) and MA(1) through MA(3) processes. Compare their ACF and PACF graphs. How do the graphs help us to identify the type and order of each process?

Analysis

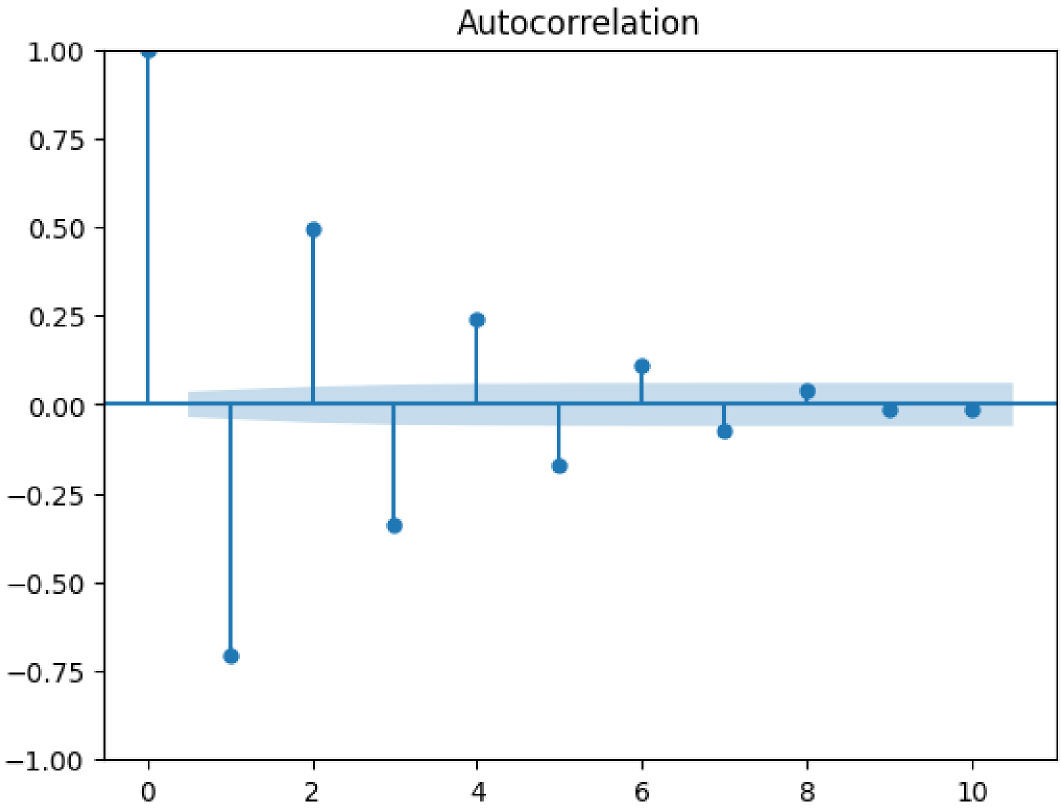
Simulate the AR1 AR2 AR3:



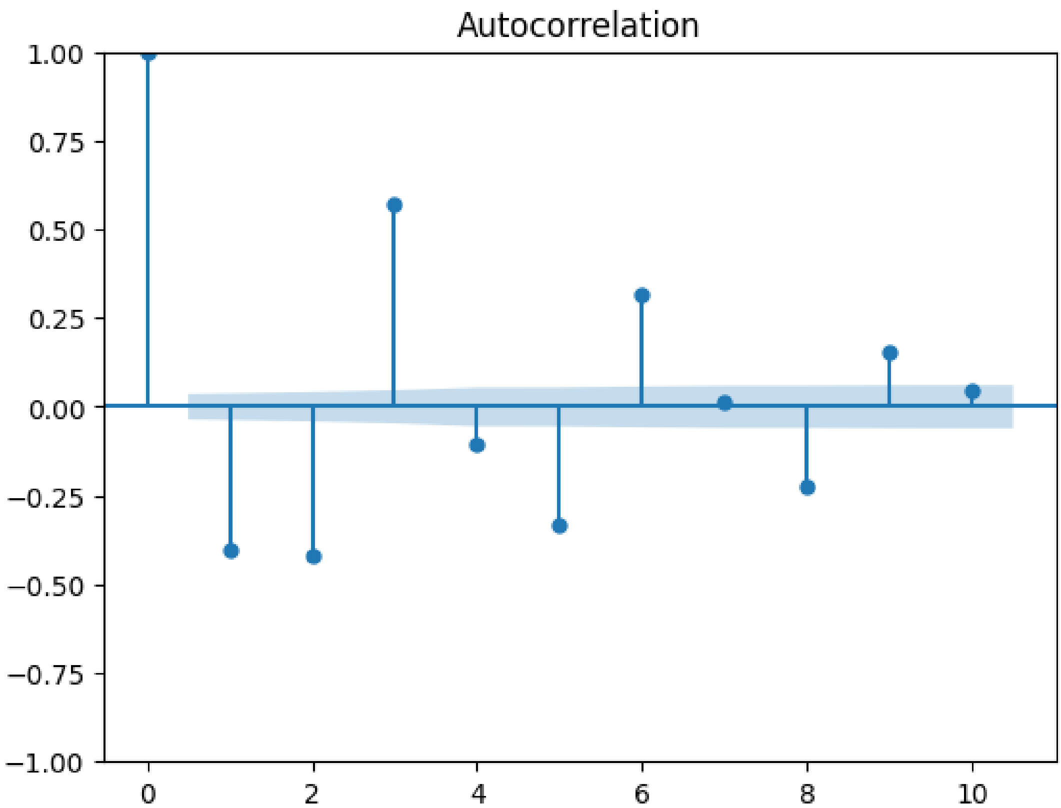

AR2
6
4
2
0
−2
−4
−6
0
500
1000
1500
2000
2500
3000


AR3
6
4
2
0
−2
−4
−6
0
500
1000
1500
2000
2500
3000

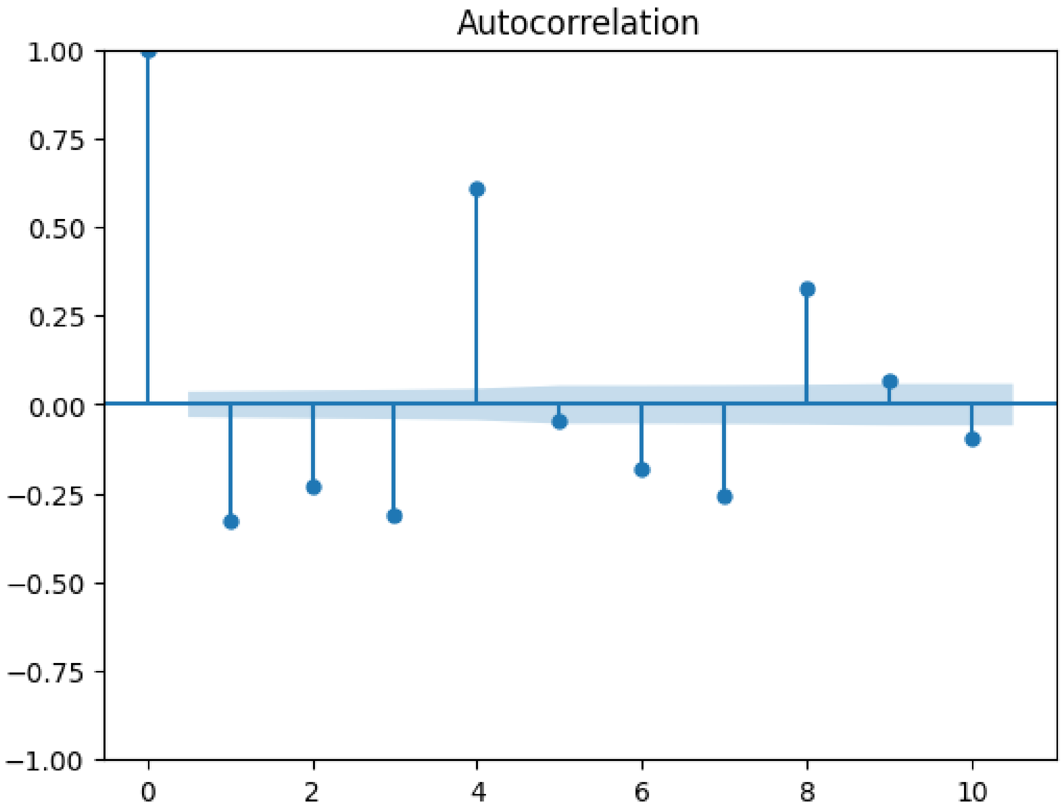
ACF of AR1:



ACF of AR2:

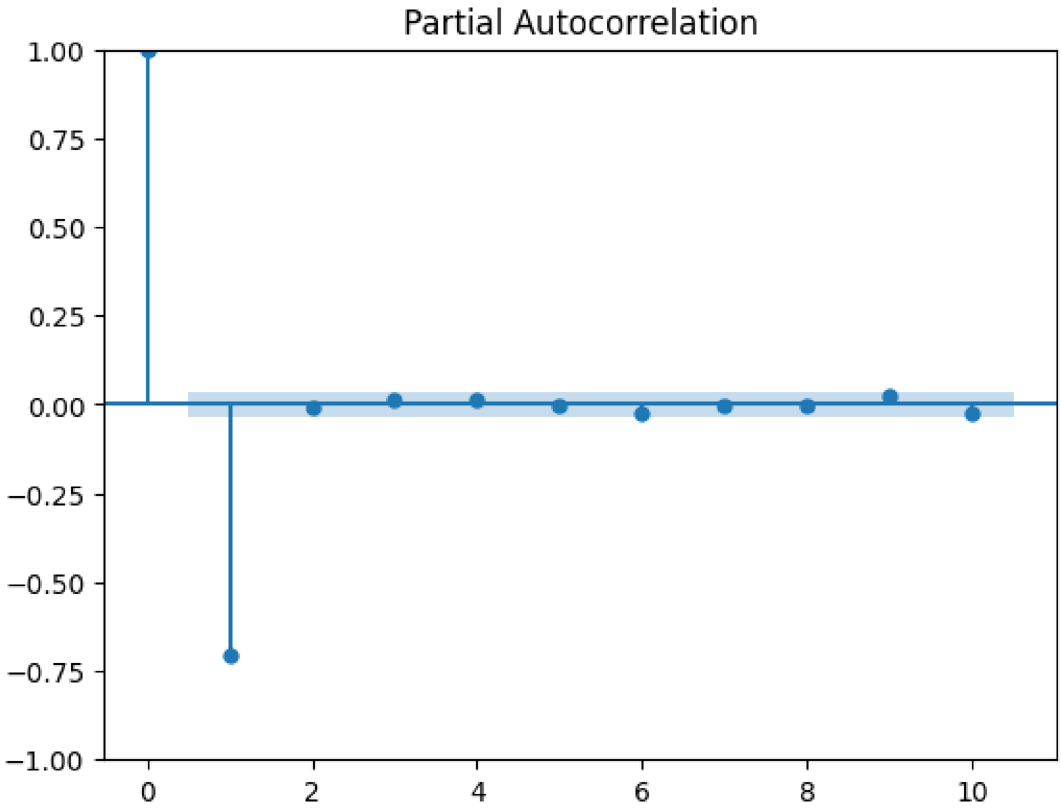


ACF of AR3:

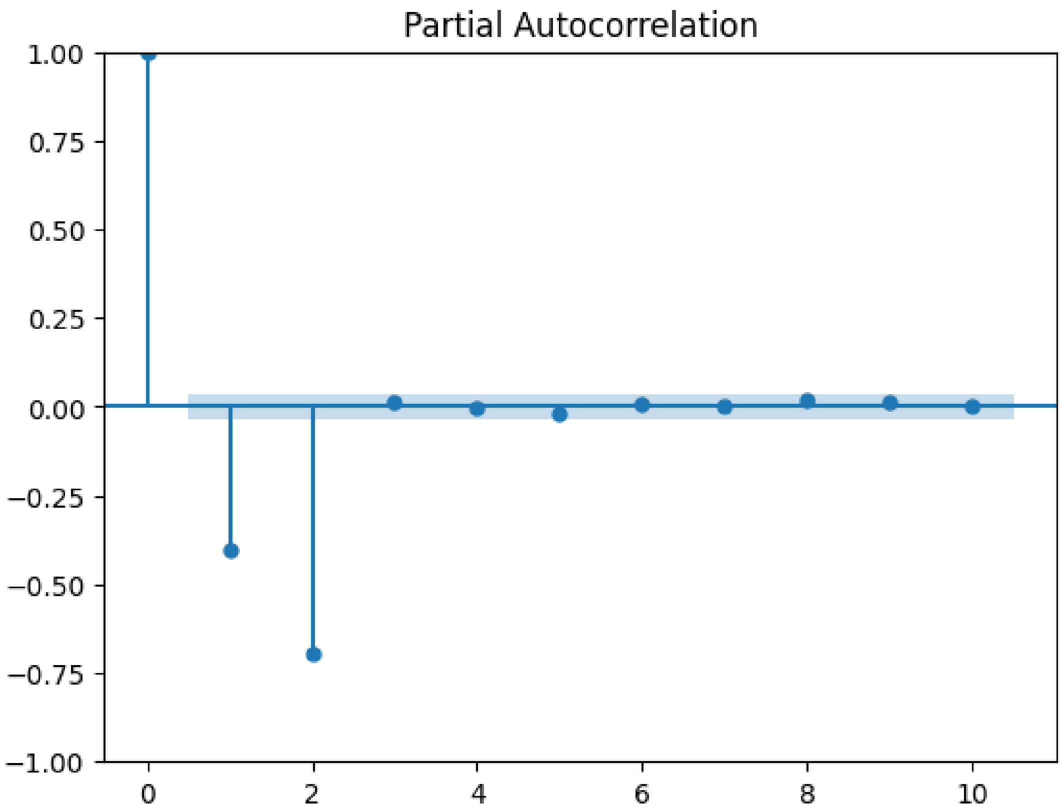


For AR process, it can be seen that the AR process shows a gradually decreasing trend in the ACF plots, because as an AR process, its current and past lag items have a good correlation.

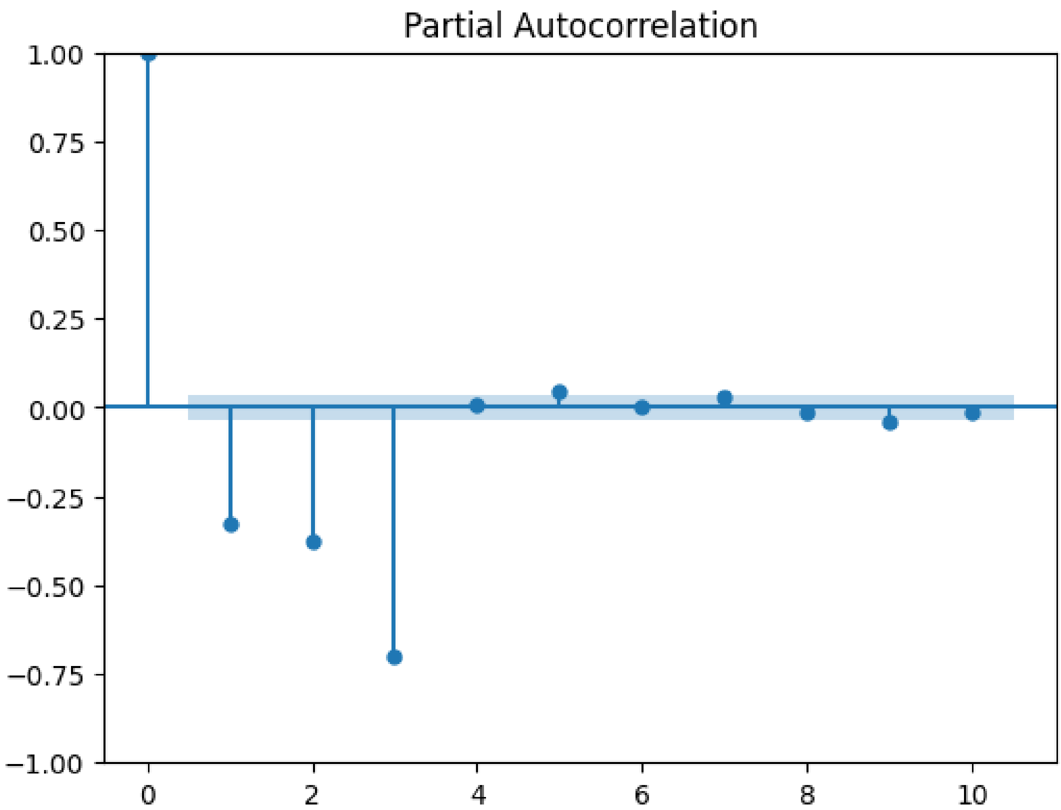
PACF of AR1:



PACF of AR2:

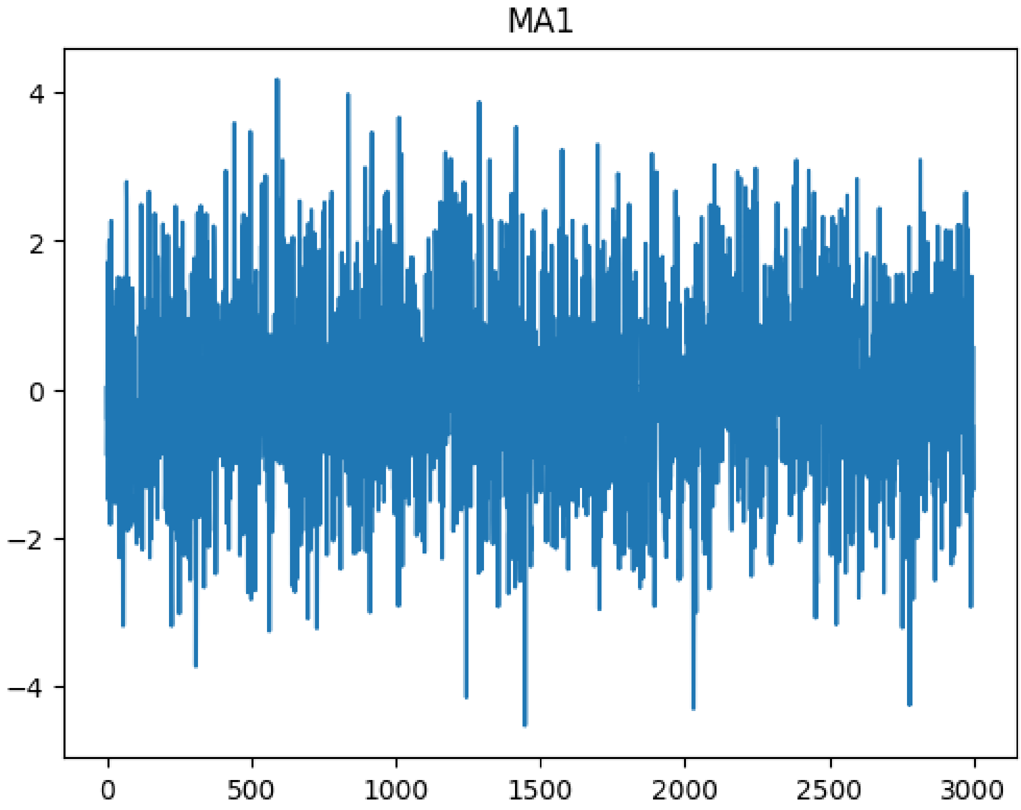


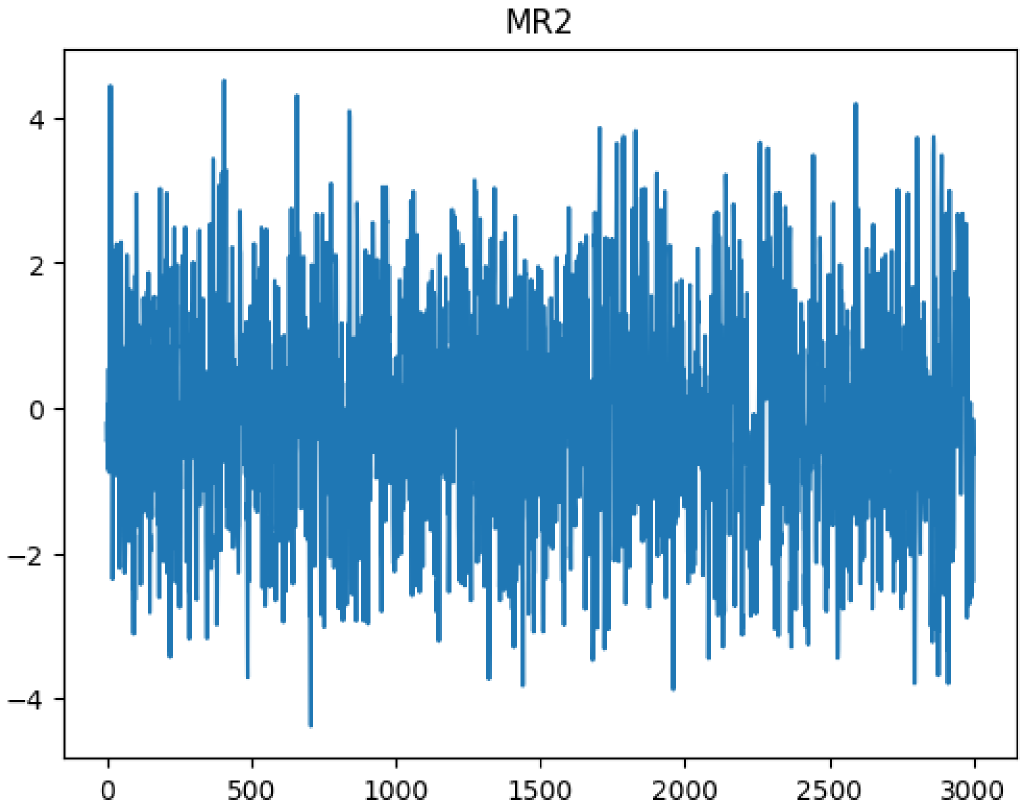
PACF of AR3:

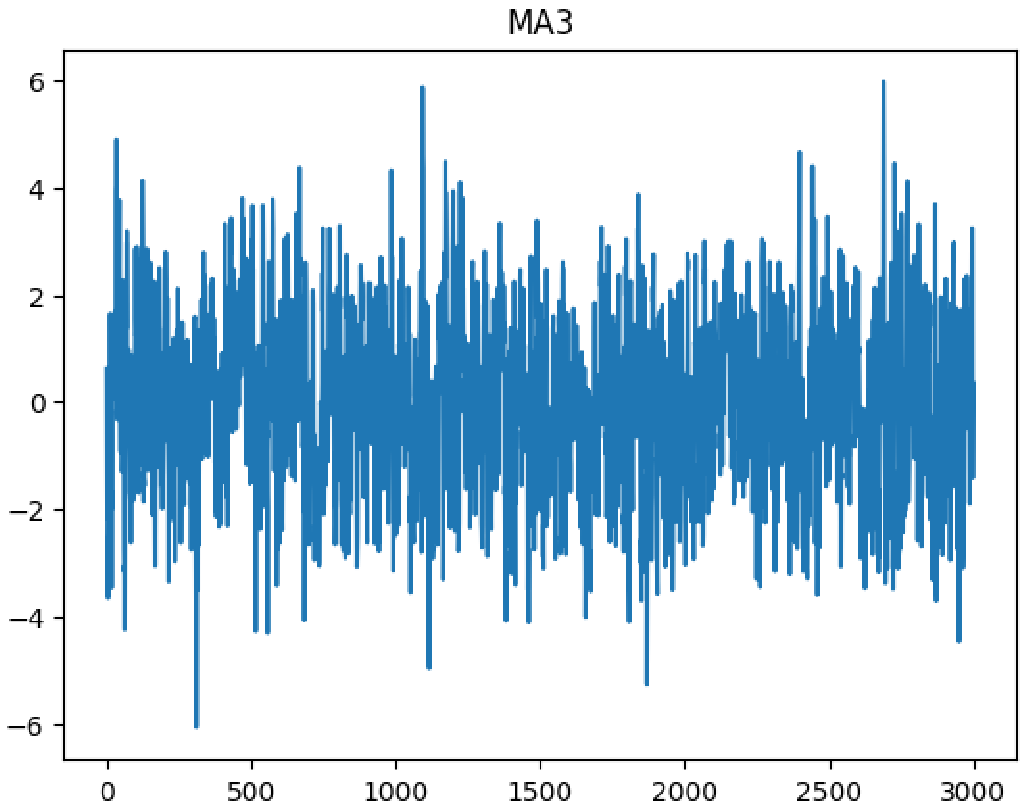


For AR process, it can be seen that PACF drops sharply after the order of the lags, since these lags close to the current term capture changes well.

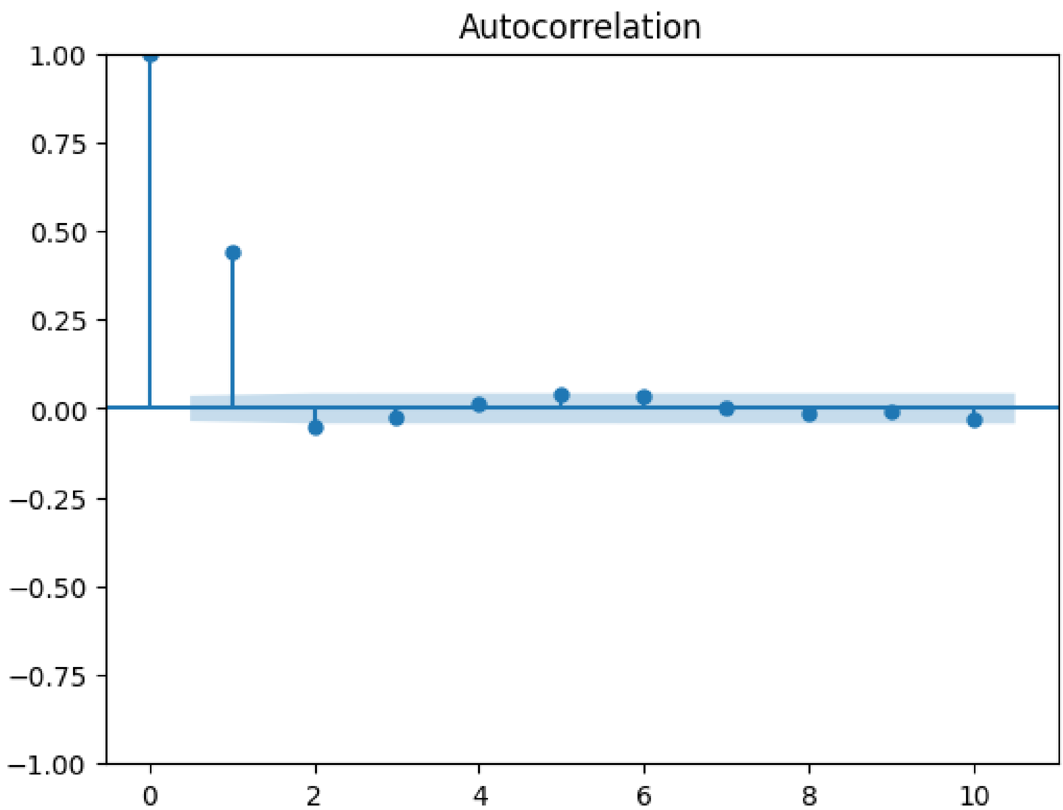
Simulate the MA1 MA2 MA3:


MA1
4
2
0
−2
−4
0
500
1000
1500
2000
2500
3000

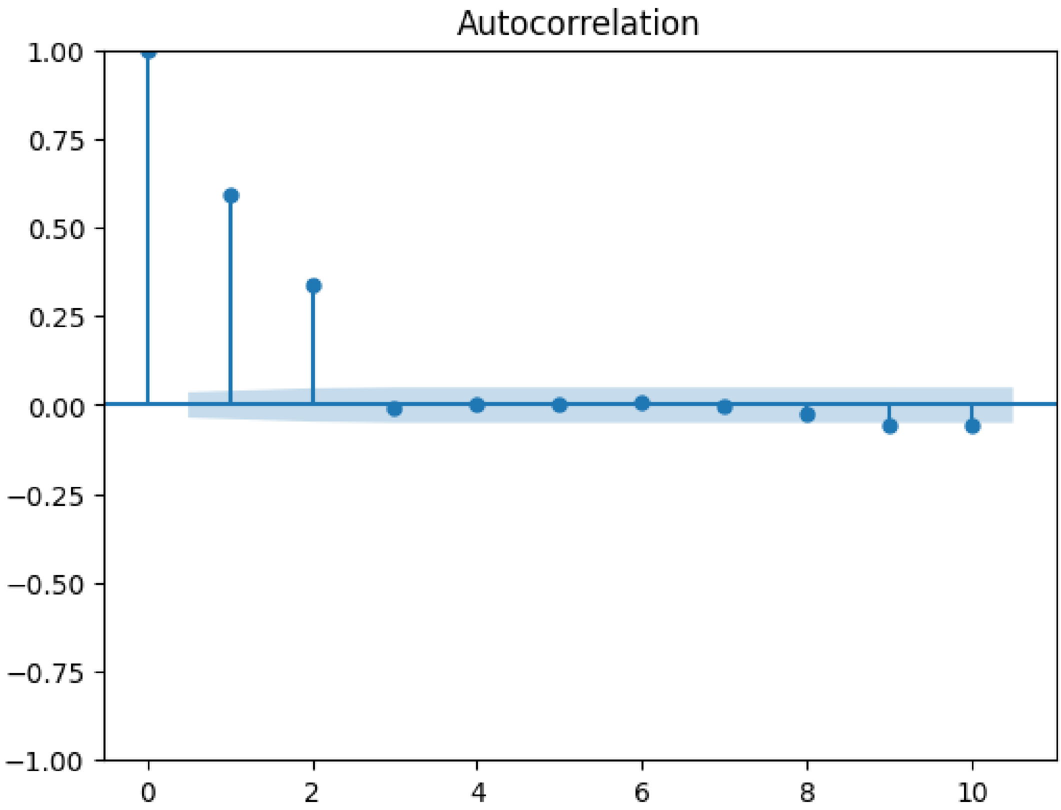

MR2
4
2
0
−2
−4
0
500
1000
1500
2000
2500
3000


MA3
6
4
2
0
−2
−4
−6
0
500
1000
1500
2000
2500
3000

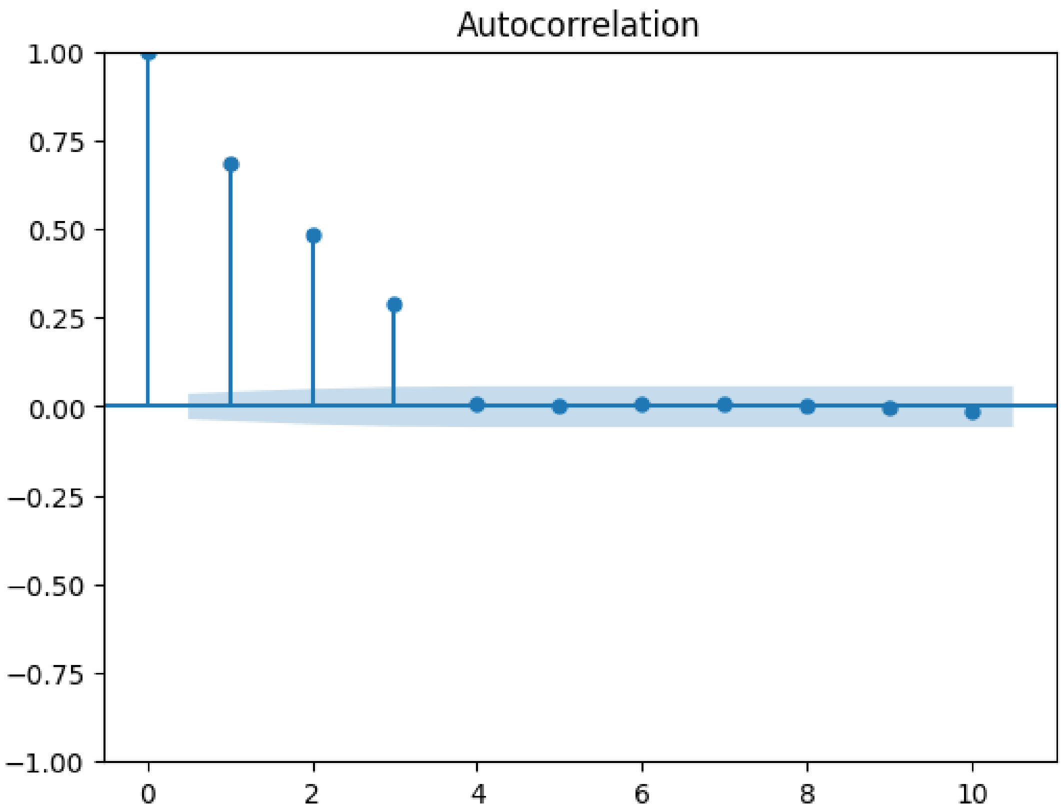
ACF of MA1:


Autocorrelation
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
2
4
6
8
10

ACF of MA2:

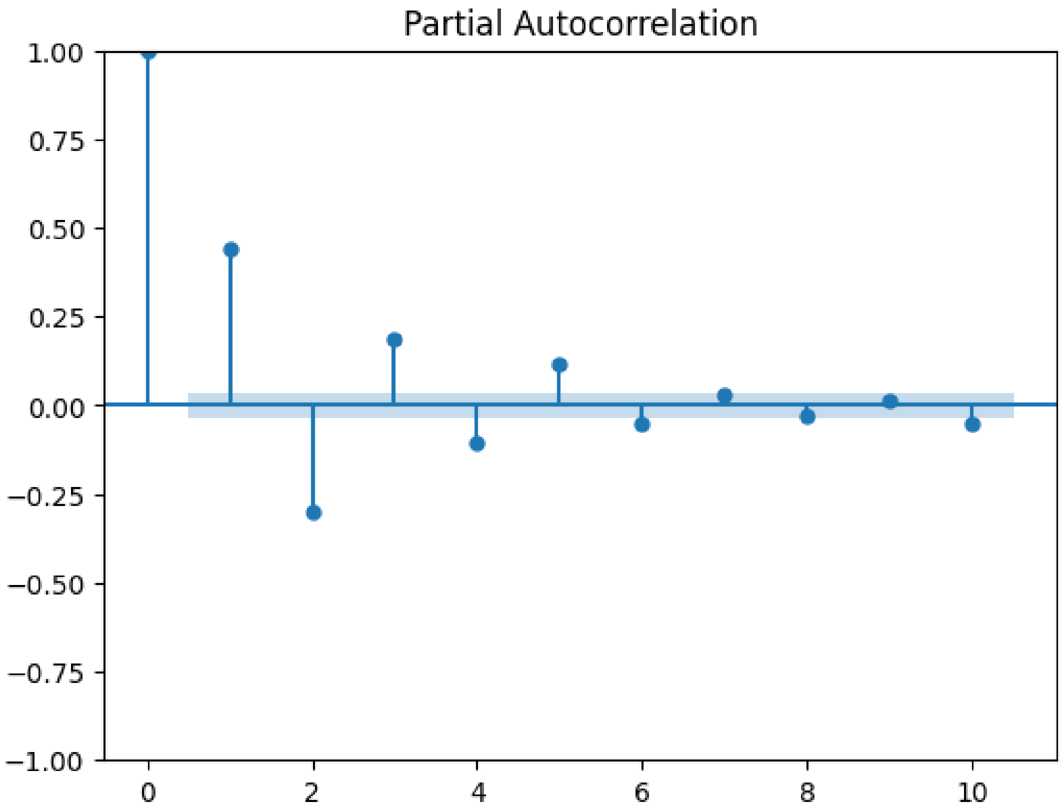


ACF of MA3

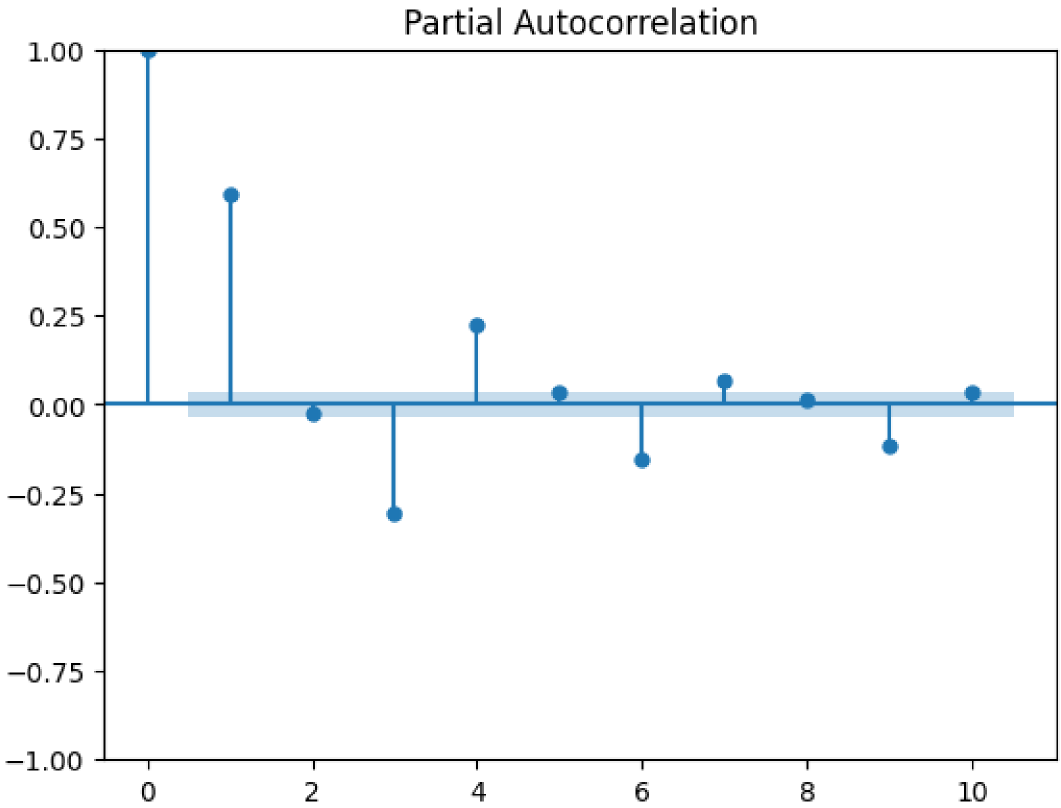


For MA process, it can be seen that ACF drops sharply after the order of the lags.

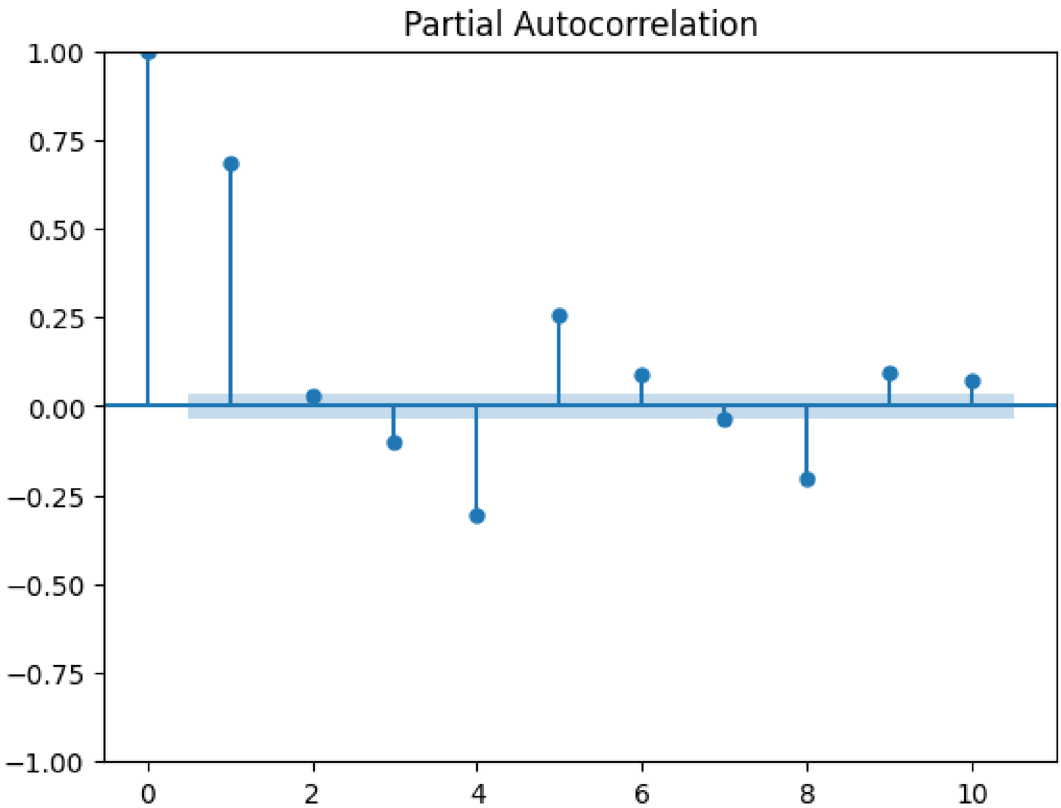
PACF of MA1:



PACF of MA2:



PACF of MA3:



For MA process, it can be seen that the MA process shows a gradually decreasing trend in the PACF plots.

These graphs help me tell the difference between AR process and MA process. I can optimize the features in the AR process through the PACF plot, because PACF is able to remove the variation already explained by the previous lag terms.

I can optimize the characteristics of the MA process through the ACF diagram, because the MA process has no seasonal and trend components, and we only get the relationship between the current item and the residual in the lag item in the ACF diagram.