Ethereum price prediction with ARMA models

Bohdan Buhrii

January 2022

Contents

1 Motivation

Ethereum is the decentralized, open-source technology that powers much of the cryptocurrency world. Everything from decentralized finance applications and non-fungible tokens to enterprise blockchain solutions rely on Ethereum's technology. That has made Ethereum's native token, Ether, the second-largest cryptocurrency after Bitcoin.

The most direct option to have profit from the growing use of Ethereum is buying Ethereum cryptocurrency itself, but it can also be considered as store of value. Therefore there is a clear reason in desire to predict it's future prices.

For predictions we consider prices at the end of each week during last three years, in period from the January 1, 2019 till the beginning of 2022.

2 Data exploration and transformation

2.1 The first look

We start with plotting all the available data to have better understanding of time series we are working with (Fig.2.1). We also split data, reserving last 10 weeks of observation for the test set.

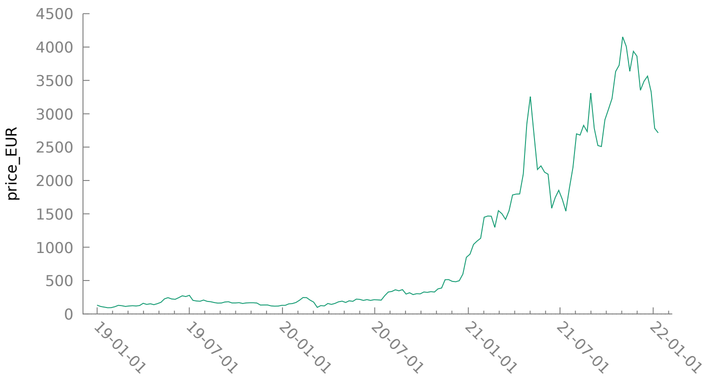

Figure 2.1: History of the Ethereum prices.

There is no obvious regularity in the data. Time series has no seasonal components, but we can easily notice a clear upgoing trend, with bigger jumps at the second half of the timeline. So we conclude process is not a stationary one.

2.2 Stationarity

The assumption that our time series is a realization of a stationary process is fundamental in time series analysis. Thus, in order to construct an ARMA model, we must first determine whether our time series can be considered a realization of a stationary process. If initial data has no stationarity, we can achieve it throught specific transformations.

Mean of the process is not constant and getting bigger and bigger over time. Therefore, taking the first difference seems to be reasonable (Fig.2.2).

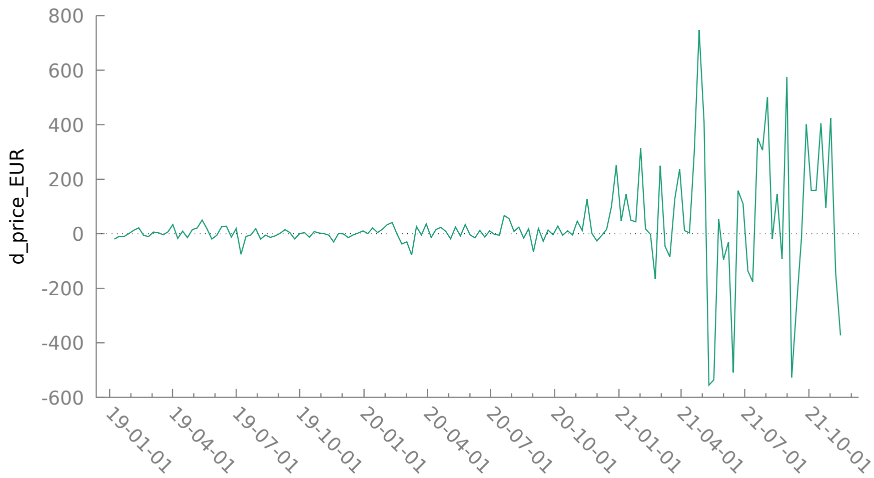

Figure 2.2: The first difference.

After the transformation we obtain the DGP with zero mean, as was expected, so we can proceed with the exploration.

Still, the process is not identically distributed because of high fluctuations in the second half of the timeline. To avoid this, we have to apply logarithmic function to the initial data. We will consider these data further in this work (Fig.2.3).

What we observe is reduction of fluctuations in variance of the process, which allows us to take first difference again, but now based on logarithmic data (Fig.2.4).

Figure 2.3: Logarithmized data.

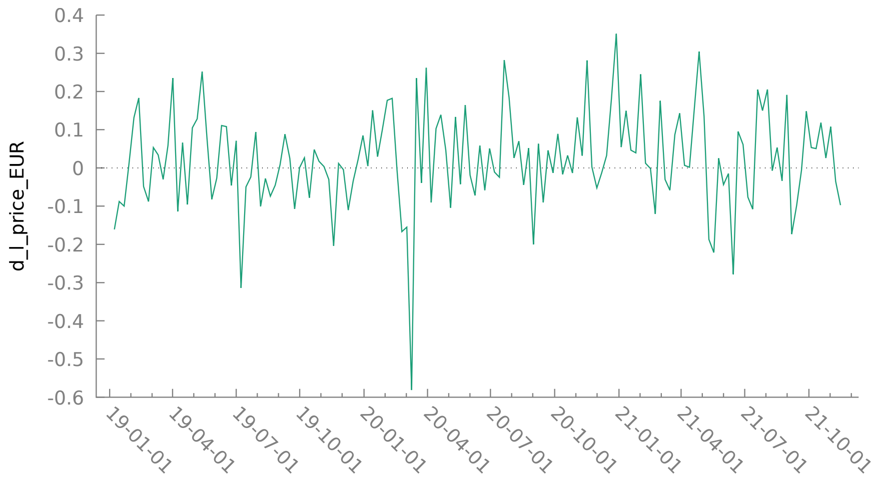

Figure 2.4: First difference of the logs.

2.3 Augmented Dickey-Fuller test

Even thought the process might look like stationary, our conclusions may not be objective. As a last check of stationarity of the time series we consider the Augmented Dickey-Fuller test. We assume that the time series has a unit root and then try to reject this hypothesis.

After running the test, the unit root null-hypothesis was rejected. As a result we got p-value near 1.5×10^{-12} which is definitely less than 0.5. So, indeed, the time series is stationary.

For the seek of comparison we also consider the Augmented Dickey-Fuller test for the time series on other steps of transformation (Table 1), which confirms our assumption about stationarity of other transformations, considered previously.

	p-value	is stationary?
raw data	1	No
logs	0.9834	No
first difference of logs	1.5×10^{-12}	Yes

Table 1: Results of Augmented Dickey-Fuller test.

3 Model identification

3.1 Autocorrelation

To identify a model, represented in the time series, we should first investigate, how values depend on each other. If we won't be able to spot any correlation, then there is no reason to proceed, because we won't be able to do any valuable prediction.

Therefore it is useful to build correlogram (Fig. 3.1) with autocorrelation and partial autocorrelation functions. It also will help to identify the order of our ARMA model.

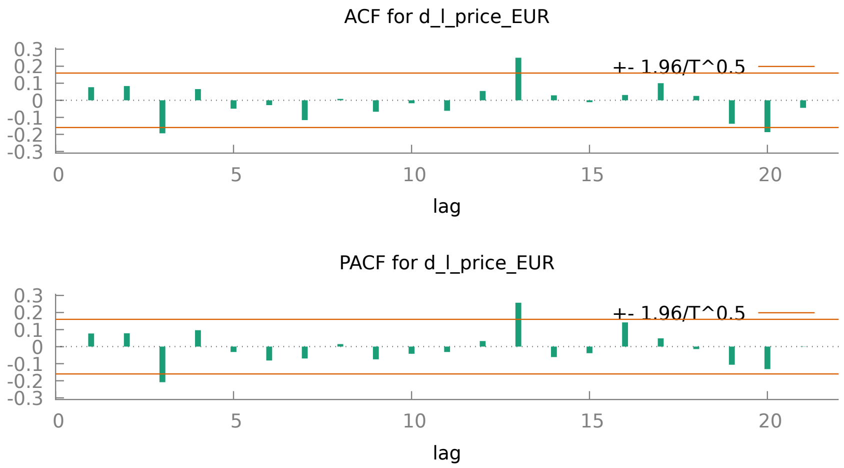

Figure 3.1: The correlogram.

Values at lags 3 and 13 are significantly different from zero. Luckily, we see the correlation, the process cannot be considered a realization of a white noise. The correlation is not significant, so it will be a bit tricky to predict values with high confidence, but we will try.

In advance we notice that MA model of order 3 might be a suitable realization of our process, but in general this case is difficult to interpret.

3.2 Automatic criteria

The mixed models can be particularly difficult to identify by using the correlogram and the partial correlogram. For this reason, it is preferable to use information-based criteria such as AIC, BIC or HQC. The AIC usually picks models which are over-parameterized. The BIC is a criterion which attempts to correct the overfitting nature of the AIC.

Among a set of models, we select the values of p and q for our fitted model to be those which minimize these criteria. We analyze models from ARMA(0,0) to ARMA(13,13). The most valuable results are shown it the Table 2.

AR	MA	AIC	BIC	HQC
0	0	-187.2606	**-181.2662**	-184.8251
0	1	-185.9914	-176.9997	-182.3381
1	0	-186.1382	-177.1466	-182.4849
0	3	-191.0878	-176.1017	**-184.9990**
3	0	-189.7032	-174.7172	-183.6144
5	2	**-191.4378**	-164.4629	-180.4779

Table 2: The automatic criteria test

The BIC criterion warns us that our process might be a realization of white noise. However, other criteria, AIC and HQC, propose different, MA(3) and ARMA(5, 2) models respectively, which we are going to estimate. After estimation of all the parameters, we should verify if the model is a good representation of our data.

4 Models evaluation

To check if the model fits data well, we have to calculate the residuals and then consider their properties. From theory we know that if the model chosen is correct and if the estimated parameters are close to the actual values, then the residuals should be a realization of a white noise.

4.1 ARIMA(0,1,3)

Firstly, let's evaluate ARIMA(0,1,3) model, which is MA(3) on first difference. We build the residual correlogram to identify, if there is any correlation between residuals (Fig. 4.1).

At first lags we see absolutely no correlation between residuals and only one value at lag 13 is significantly different from zero. So, at some point, it can be

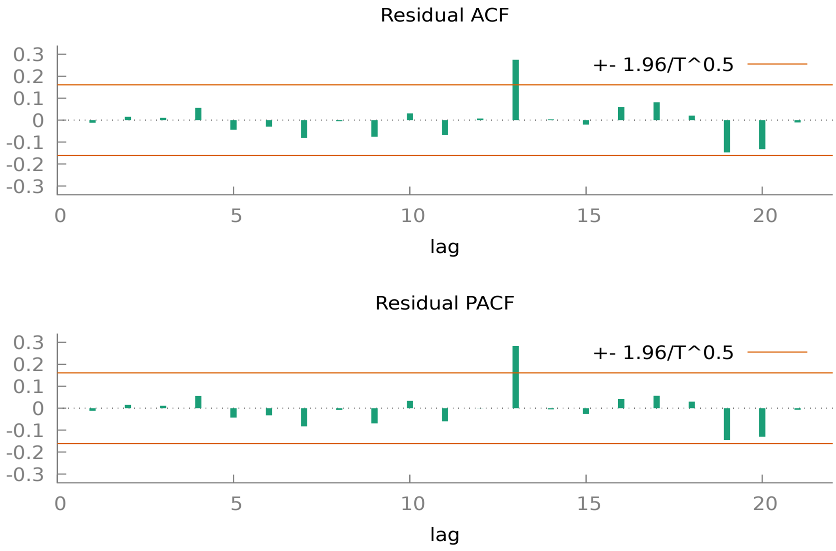

Figure 4.1: The residual correlogram for the ARIMA(0,1,3) model.

considered a realization of a white noise. The same conclusion may be done from Q-statistics plot (Fig. 4.2):

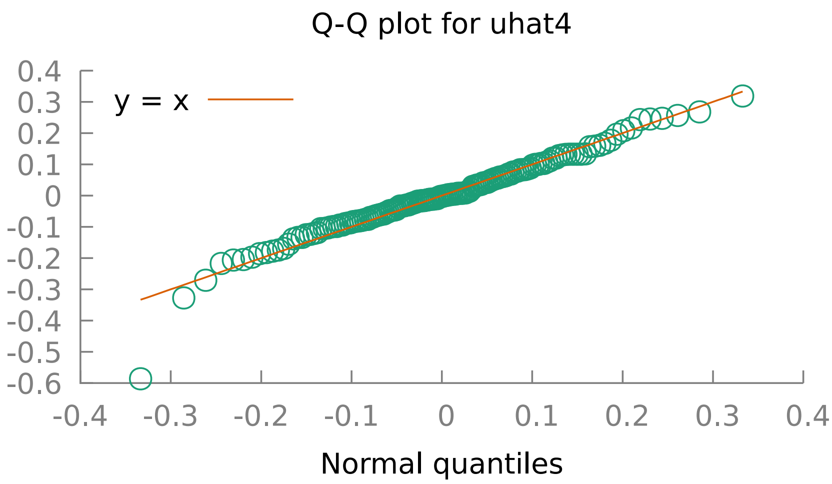

Figure 4.2: The residual Q-Q plot for the ARIMA(0,1,3) model.

It is not clear, what to do with that one anomaly at lag 13. Therefore we should check if residuals' distribution is a realization of Gaussian white noise. After testing it for normality we get results, shown on Fig. 4.3.

It is clear that the distribution is not a Gaussian one, so, theoretically, the ARIMA(0,1,3) model is not a good choice for our time series. We can also see this from the model forecast on Fig. 4.4.

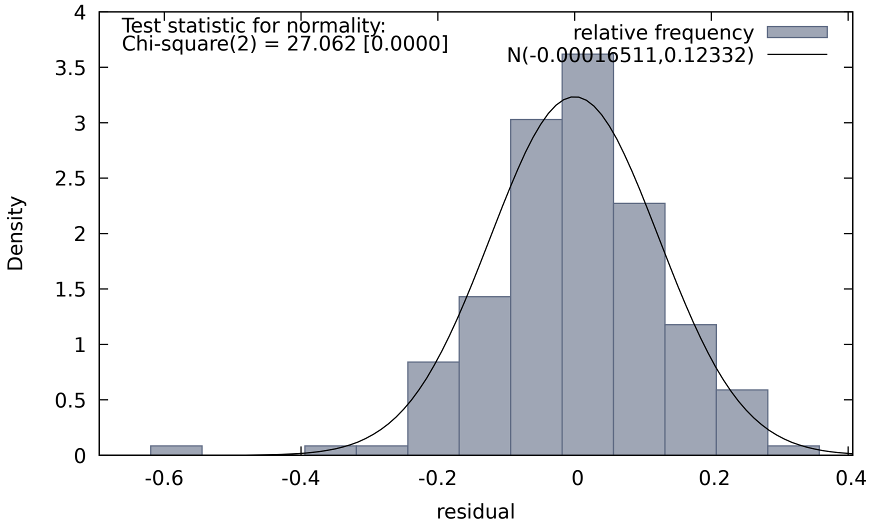

Figure 4.3: The residuals' normality test for the ARIMA(0,1,3) model.

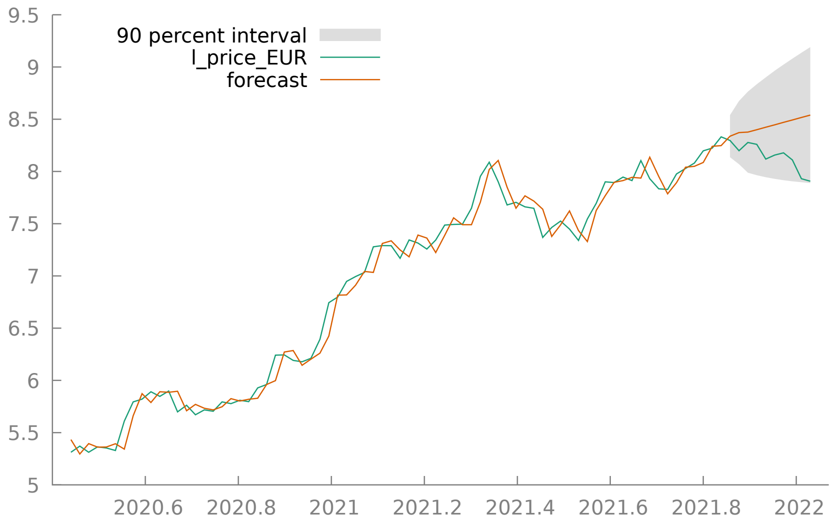

Figure 4.4: The forecast based on ARIMA(0,1,3) model.

4.2 ARIMA(5,1,2)

Secondly, let's evaluate ARIMA(5,1,2) model, which is ARMA(5,2) on first difference. As usual, we consider the residuals' correlogram on Fig. 4.5.

It has the same issue at lag 13, as the previous model, but with much smaller value. After testing it for normality we get analogical results (Fig. 4.6).

Unfortunately, it is also not a Gaussian distribution and model is not a good approximation of our data. Predictions of this model looks like on Fig. 4.7.

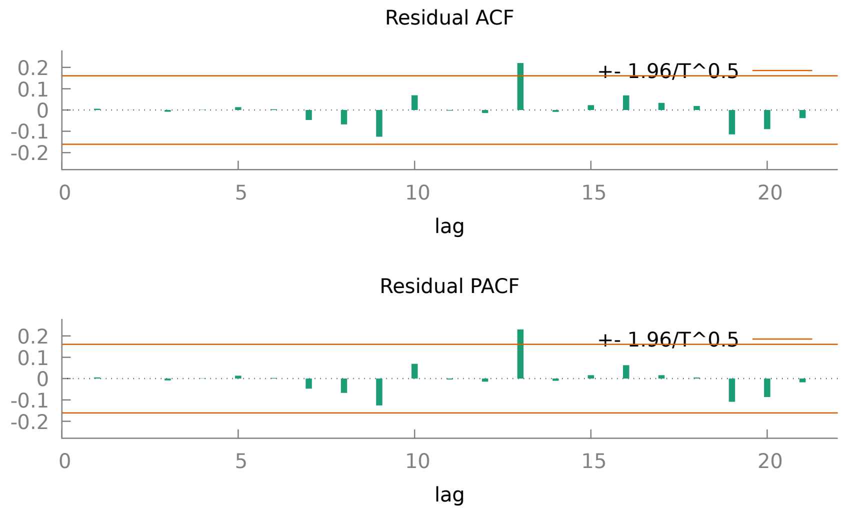

Figure 4.5: The residual correlogram for the ARIMA(5,1,2) model.

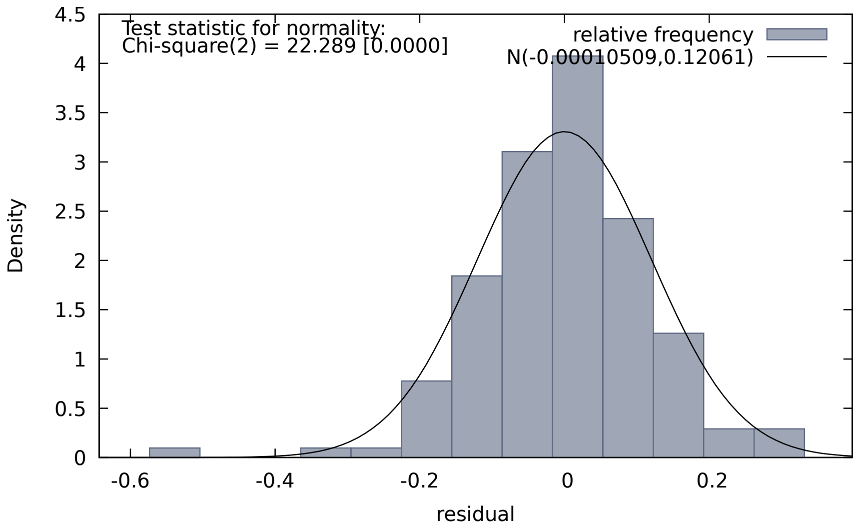

Figure 4.6: The residuals' normality test for the ARIMA(5,1,2) model.

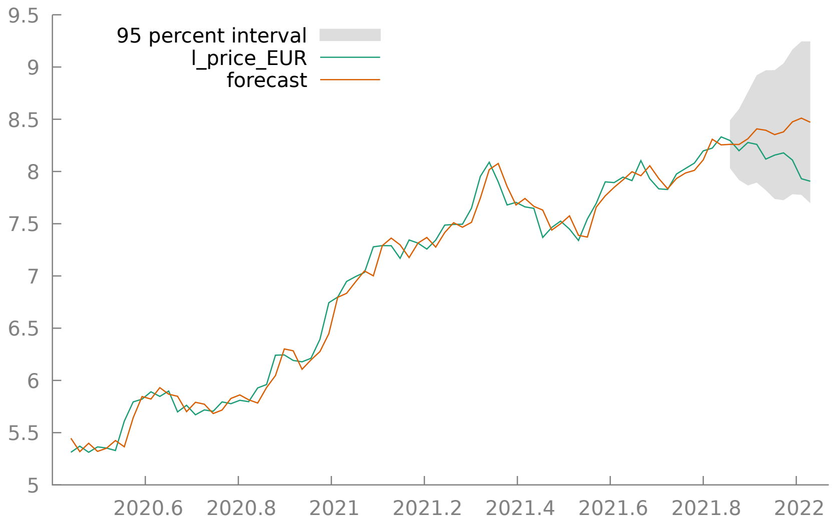

Figure 4.7: The forecast based on ARIMA(5,1,2) model.

Generally speaking, we obtain similar results from few other ARIMA models. This is a reason to draw a conclusion that it is hard to obtain good approximation for the chosen time series.

4.3 Comparison

To compare performance of these two models, we should consider root mean squared error. It also makes sense to compare predictions with white noise based one, to see if any of the models is efficient at all.

Model	Root Mean Squared Error
white noise	0.36843
ARIMA(0,1,3)	0.34833
ARIMA(5,1,2)	**0.3117**

Table 3: Models' error comparison.

We see that both models performs better than "random" predictions. The ARIMA(5,1,2) has lower RMSE than ARIMA(0,1,3), but it is also more complex, with bigger number of parameters, which means that there is a risk of overfitting.

Both models predict future not very well and this is most likely related to the almost random nature of the time series. But if we have to choose, than it is suitable to say that ARIMA(5,1,2) model represents the data distribution better.

5 Conclusion

As we can see, predicting future is definitely not an easy task. A reliable ARMA time series forecast requires that the future is not too different from the past. Therefore, poor performance might be related to the nature of the time series, which in our case was very similar to the stock data.

Even if we are able to build the appropriate model, usually there is no sense to consider predictions for long time periods as valuable. If one have to use such models in a real world, a good strategy is to retrain the model at the end of time interval, when new data is available, to maintain maximum possible accuracy.

The flexibility of ARMA models can be distinguished as one of the advantages, in a row with ability to forecast data without impact of side conditions. On other hand, it requires large number of observations for model identification, estimation and training. Also it cannot stand against structural changes of the data.

As a result we may sum up that ARMA model predictions not always may be considered as reliable, but they play important role for us to understand the data and the environment better.