How Will Bitcoin Behave in the Near Future?

written by Hikmat Abdulazizov Hikmət Əbdüləzizov
Part 2 – Return and Volatility Prediction

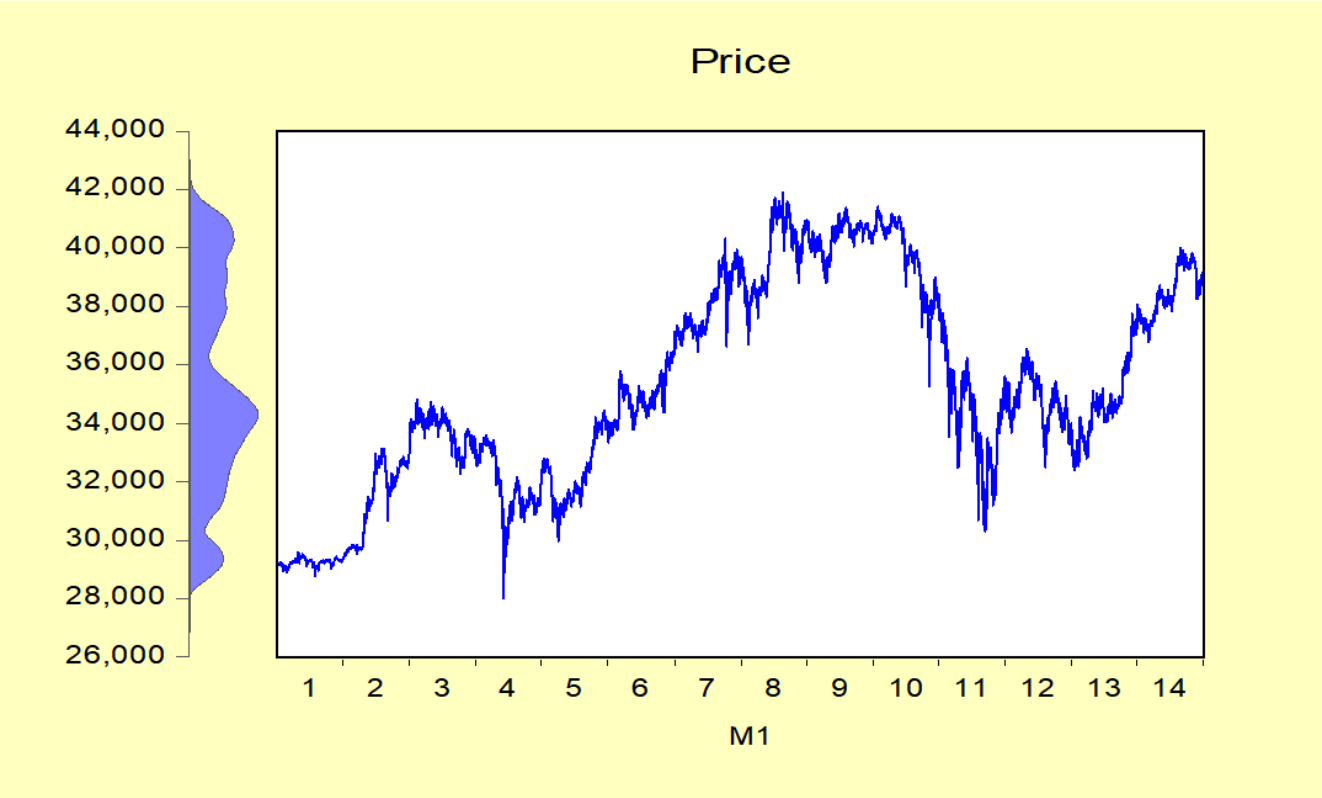
In Part 1 of this series on predicting Bitcoin returns, we introduced the characteristics of Bitcoin and gave data examples to conduct modelling on. Bitcoin is good for safe irreversible financial transactions. However, it is also invested in as an asset. Therefore, we need to know how to predict its price and returns. We will apply some statistical models and choose those that best match the data.

First of all, we need to know which statistical process Bitcoin price and return follow. This analysis will be used for direct price prediction and input for volatility prediction as return process. Please note that in Part 2 here, we will focus more closely on estimating volatility rather than on price and return prediction because it is stylized fact that price and return of an asset are not as accurately predictable as their conditional volatility. Moreover, a conditional variance estimation gives us information about the distribution of the returns, which, on its own, will suffice for analysis. Once we estimate the conditional volatility one-step ahead, we will have a good measure of the lowest and highest returns, and we can build stop-buy, stop-sell and other trading strategies well.

Here In this paper, we will use only minutely data. In later parts, we will use other data types including daily data for volatility prediction. Keep in mind that the GARCH framework is widely used in risk analysis in banks as well. Let’s check the behavior of our data. We will mainly use R programming language and statistical package Eviews for our analyses.

Codes will be made available upon request of the author.

Graph 1. Values and Density of Bitcoin Prices



The Eviews output above shows 15-day minutely data and its kernel density from January 2021. The data has 20.154 observations, which is quite rich. The graph implies a random trend (that is, unit root). Let’s check:

Table 1. Eviews Output for Unit Root Test

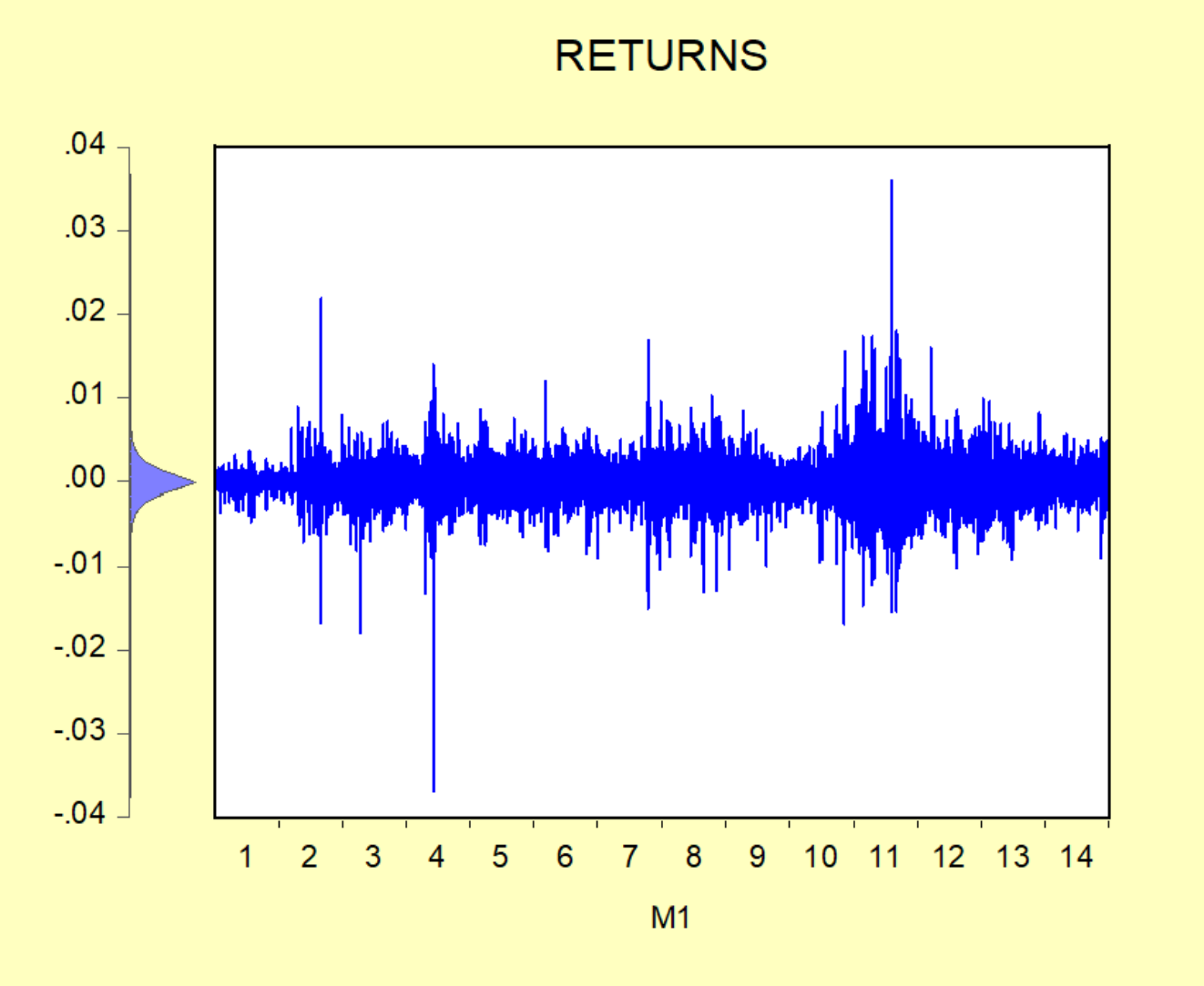
Null Hypothesis: PRICE has a unit root
Exogenous: Constant
Lag Length: 3 (Automatic - based on SIC, maxlag=45)

		t-Statistic	Prob.*
Augmented Dickey-Fuller test statistic		-1.864873	0.3493
Test critical values:	1% level	-3.430503	
	5% level	-2.861492	
	10% level	-2.566785	

*MacKinnon (1996) one-sided p-values.

The test above shows that our Bitcoin price data has a unit root. PP(unit root test) and KPSS(stationarity test) tests from both R and Eviews suggest the same result. Thus, it is useless to predict the price itself since unit root means that the price has a random trend that cannot be predicted. Let's move to the returns' structure.

Graph 2. Values and Density of Bitcoin Returns



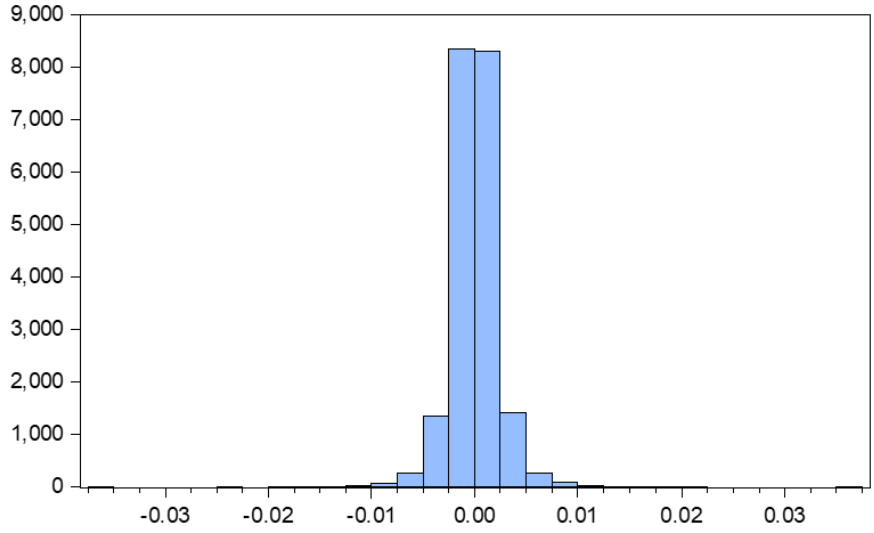
The returns' data and its kernel density look like a standard normal distribution with some spikes, but let us test its normality and stationarity. The graph also implies volatility clustering.

Table 2. Eviews Output for Unit Root Test and Normality

Null Hypothesis: RETURNS has a unit root
Exogenous: Constant
Lag Length: 2 (Automatic - based on SIC, maxlag=45)

		t-Statistic	Prob.*
Augmented Dickey-Fuller test statistic		-84.19061	0.0001
Test critical values:	1% level	-3.430503	
	5% level	-2.861492	
	10% level	-2.566785	

*MacKinnon (1996) one-sided p-values.



Series: RETURNS
Sample 1/01/2021 00:01 1/15/2021 00:00
Observations 20153

Mean	1.44e-05
Median	0.000000
Maximum	0.036024
Minimum	-0.036906
Std. Dev.	0.002228
Skewness	0.100415
Kurtosis	17.55369
Jarque-Bera	177892.3
Probability	0.000000

The tests suggests that returns are stationary but not normally distributed. It gives us a good idea as to when we estimate volatility since we can use different distributional assumptions for returns. If we use Student's t-distribution instead of normal, we would expect riskier values of up and downturns in data, as Bitcoin price behavior would suggest. R assigns the returns data ARMA(0,3) process (that is, MA(3)), but Eviews suggests the model for returns could be ARMA(4,4) process. I would go with MA(3) process as it is more parsimonious and depicts the randomness of the returns well. Instead of predicting returns themselves, we skip directly to volatility modelling as it is more interesting and useful.

Engle (1982) developed the Nobel-winning Autoregressive Conditional Heteroskedasticity (ARCH) model that recognizes the difference between unconditional and conditional variance and lets the conditional variance change over time as a function of previous periods' error terms. This technique has the ability to capture the effect of volatility clustering, but it requires a model with a relatively long lag structure, which makes estimation difficult. Below is the formulation of the theory and stylized facts. Note that, MA(3) process meets the error criteria below.

R[t] = mu + e[t]

e[t] = s[t]*z[t].

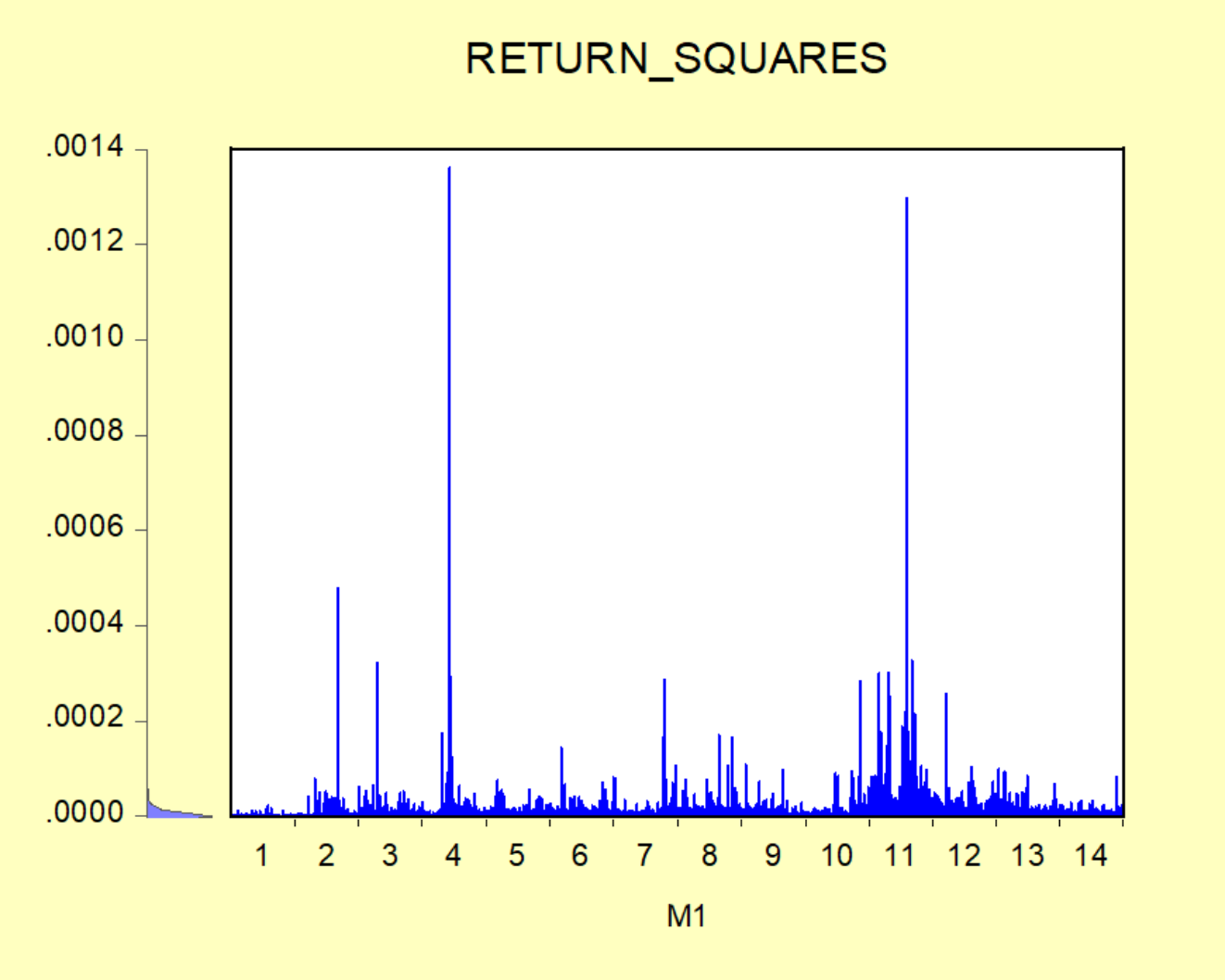
$$s_t^2 = \alpha_0 + \alpha_1 e_{t-1}^2 + \cdots + \alpha_1 e_{t-q}^2$$

Where $\alpha_0 > 0, and\ \alpha_i \geq 0$ for i>0

To make this task easier, Bollerslev (1986) proposed a GARCH model that enables a reduction in the number of parameters by imposing nonlinear restrictions. This GARCH model can predict unconditional variance and requires fewer parameters. In a GARCH model, the most recent observations have greater impacts on the predicted volatility:

$$s_t^2 = \omega + \alpha_1 e_{t-1}^2 + \beta_1 s_{t-1}^2$$

Graph 3. Values and Density of Bitcoin Return Squares



The graph and the correlograms of the squared return series suggest autocorrelation, which we want for conditional variance estimation. Note that we only formulated simple GARCH(1,1) but will use as different as possible conditional volatility estimates and compare them, which is the main purpose of this paper. Moreover, we will choose different distributions for the error structure that we think better depicts Bitcoin price behavior. Please keep in mind that financial institutions use GARCH framework for value at risk analysis(VaR).

I consider you have already realized how many GARCH type models exist. So, I will choose interesting ones and summarize the results. First, I will look at the models' information criteria such as Akaike Information Criteria, Bayes Information Criteria, etc. The formula is below:

$$\mathrm{AIC} = 2k - 2\ln(\hat{L})$$

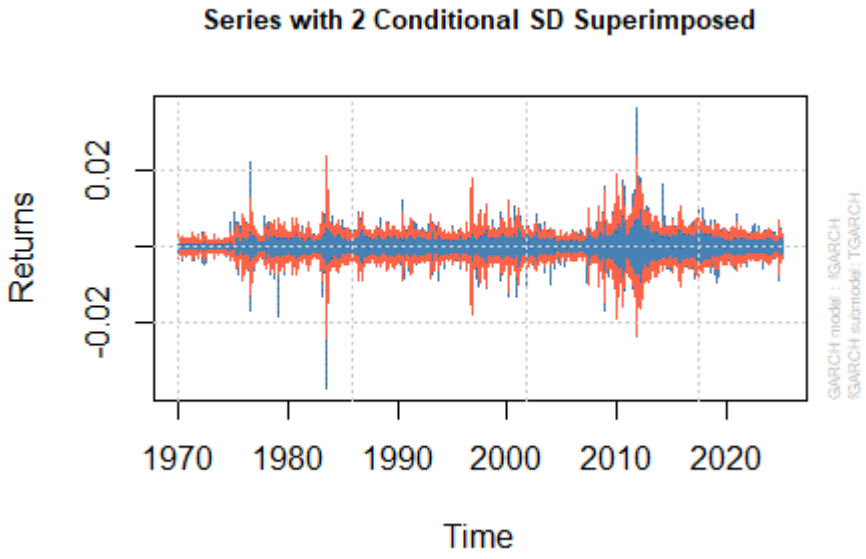
You can see that the function penalizes for overparametrization. Moreover, it has a negative relation with the log-likelihood function, which combined means that the smaller the value, the better the model. The criteria may mean information loss in the model. Let's see how the models behave:

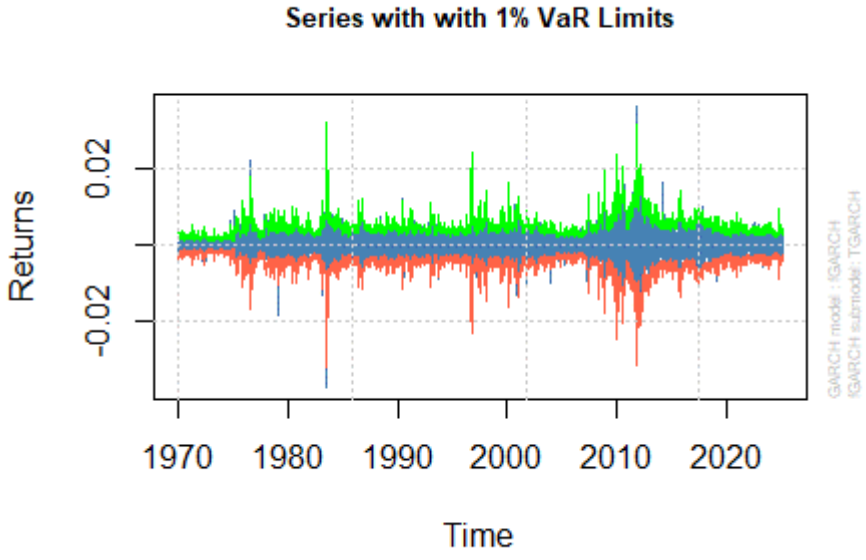
Table 3. Models with Different Distributions and Criteria

Variance Model	GARCH(1,1)	GARCH(1,1)	eGARCH(1,1)
Mean Model	ARFIMA(0,0,3)	ARFIMA(0,0,3)	ARFIMA(0,0,3)
Distribution	std	norm	std
Akaike	-99.181	-98.189	-97.159
Bayes	-99.146	-98.140	-97.155
Hannan-Quinn	-99.170	-98.270	-97.370
Variance Model	eGARCH(1,1)	tGARCH(1,1)	tGARCH(1,1)
Mean Model	ARFIMA(0,0,3)	ARFIMA(0,0,3)	ARFIMA(0,0,3)
Distribution	norm	std	norm
Akaike	-97.143	-102.123	-100.180
Bayes	-97.230	-102.164	-100.147
Hannan-Quinn	-97.454	-102.230	-100.273

We can see that the best model according to the data criteria is TGARCH with Student's t distribution. We will go through that model more closely.

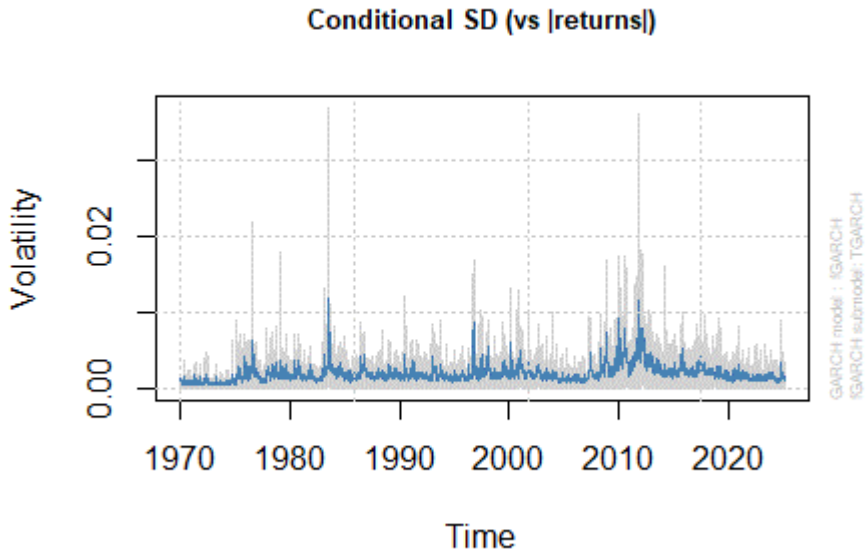
Graph 4. Series with Respective Limits





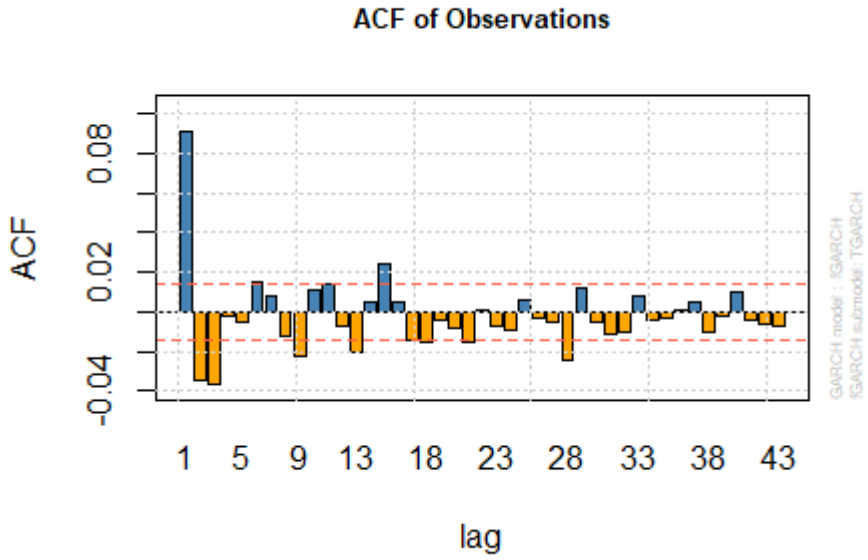
It is clear from above that our limits are estimated according to the data.

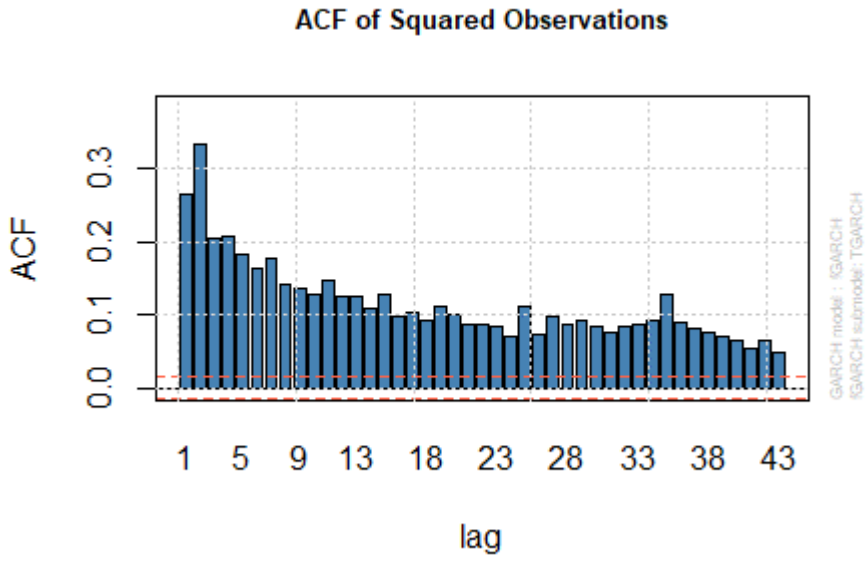
Graph 5. Estimated Volatilities



The chart above shows the estimated volatilities and their return counterparts. Looking at the graph, we can think that there is no particular problem because it accords with a typical volatility structure.

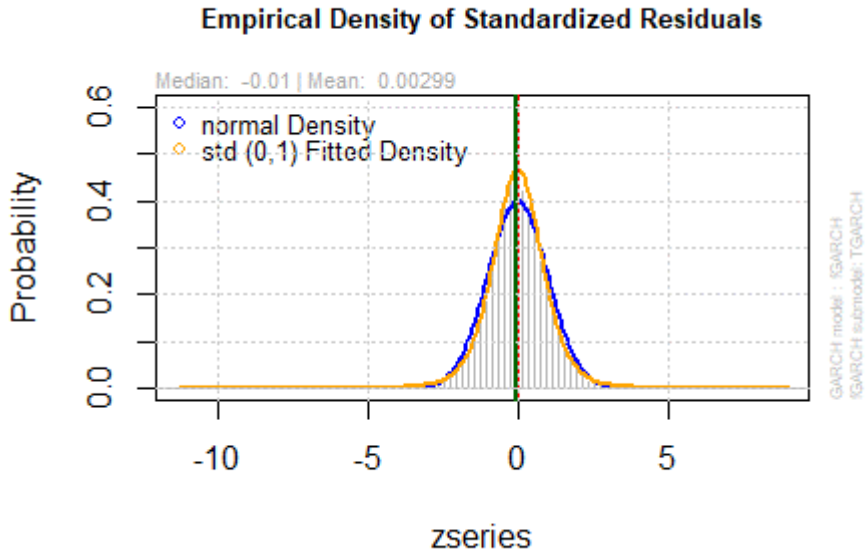
Graph 6. Autocorrelation of Series and Squared Series





Looking at the graphs above, we can conclude that the series do not have strong autocorrelation, but the quadratic series exhibits strong memory properties. This is important because we estimate the autocorrelation structure in the variances.

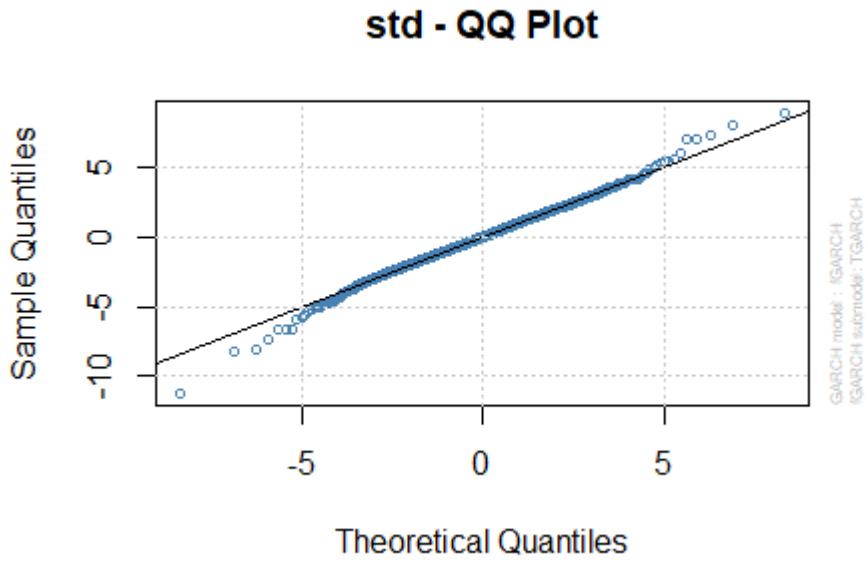
Graph 7. Normality Test



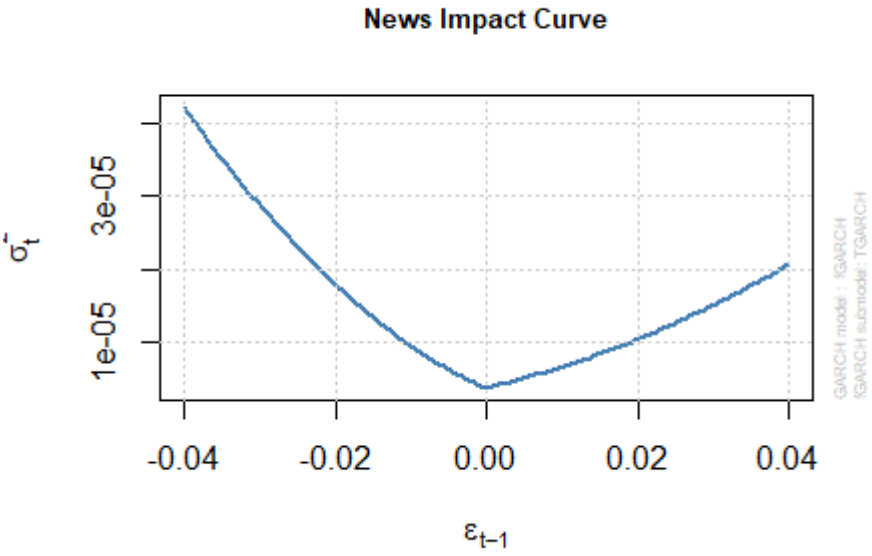
As we assume a Student's t distribution for the errors, we do not expect a perfect fit. The chart above shows that there is no concern with volatility modeling. Below is a QQ-plot, i.e., normality check in another way. But some of the tail values we

expect differ.

Graph 8. QQ Normality Test



Graph 9.News Impact Curve



Above is the News Impact Curve chart, which shows how volatility (risk) reacts to shocks (returns). As can be seen, negative values affect the risk more, and this is quite intuitive under the TGARCH mechanism.

In conclusion, we have found that TGARCH model was the best

for modelling Bitcoin data. This means that our data has asymmetry in structure, which is well captured by TGARCH model. The News Impact Curve also strongly supports our result.

References:

Bollerslev, Tim. 1986. Generalized autoregressive conditional heteroskedasticity. Journal of Econometrics 31: 307–27

Engle, Robert F. 1982. Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation. Econometrica: Journal of the Econometric Society 50: 987–1007.

You can read the first part of the article here.