Evelina Ramoskaite

DSC 425 Time Forecasting

Final Project: Milestone 3

As a group, we are exploring the relationships between several cryptocurrencies and stocks over time. This includes looking at cryptocurrency prices over time as they relate to each other, cryptocurrency-related stocks, and common index funds.

For my portion of milestone 3, I decided to explore cryptos as they relate to chip manufacturing companies, specifically Bitcoin and AMD. High-end GPUs are commonly utilized for mining cryptocurrencies because they have faster processing times than CPUs. This has led some investors to speculate that their sales may be positively correlated with high crypto prices, making them an ideal candidate for a comparison. Chips for crypto mining made up a smaller chunk of AMD's sales relative to NVDA, so I expect that the relationship will be smaller, if a link between crypto prices and GPU sales exists.

The data we used was retrieved using the Yahoo Finance API with tidyquant, and includes the adjusted daily prices of stocks between 10/01/2016 and 09/20/2021. AMD is only traded on standard trading days, while Bitcoin is available for purchase around the clock and during holidays.

To start off my exploratory process, I examined the 5-number summaries and distribution histograms for AMD and Bitcoin. The distributions of both the AMD and Bitcoin Series were heavily sewed to the right, and there was a wide range between the minimum and maximum values for the series. When a log transformation was applied, the distributions got closer to normal, but had heavy kurtosis.

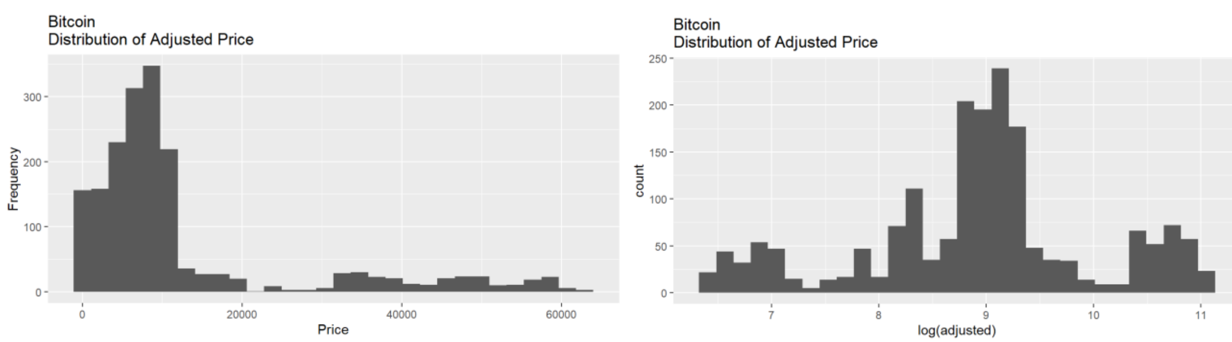

Figure 1 Bitcoin Price Distribution before and after a log transformation.

I then plotted the original series for each stock alongside the log transformed, differenced, and log return series. Taking the returns for the series still showed multiplicative behavior. The returns fluctuated a lot more in recent years, when the stock prices increased. Taking the log returns created the most uniform, stationary plot. The behavior was similar for both stocks.

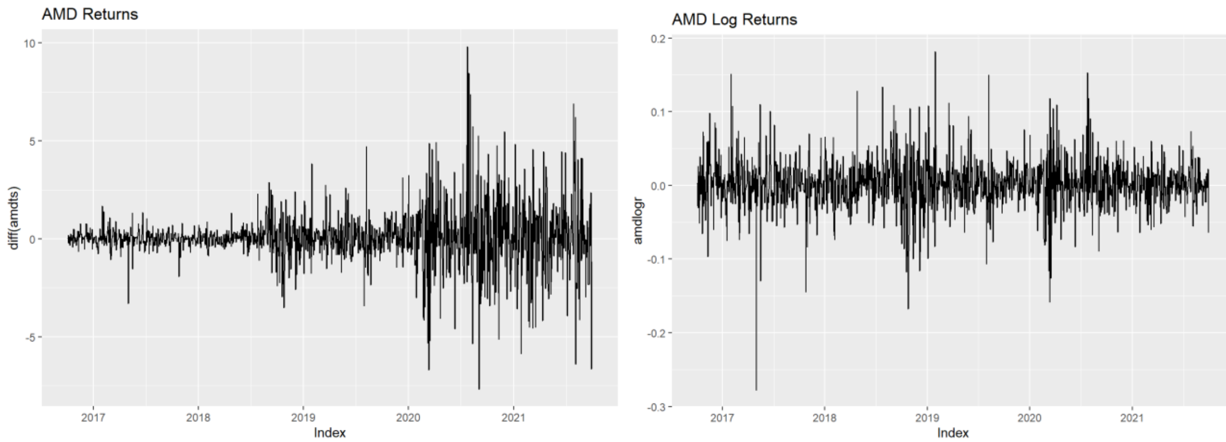

Figure 2 AMD simple returns (left), compared to the log returns.

After examining the distributions, I looked at the ACF and PACF for both the original and log-transformed series. The autocorrelation function allow us to see how values are correlated with each other at different lags, while the partial autocorrelation plots represent the autocorrelation of just the residuals. Both Graphs looked the same for the AMD Series.

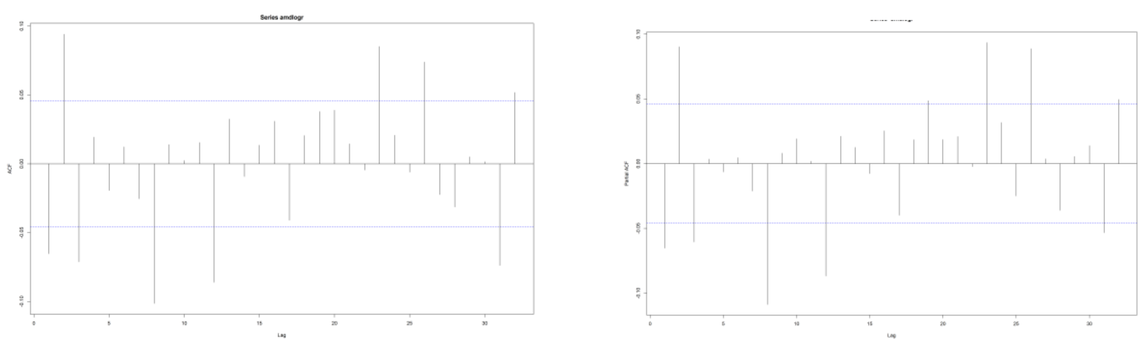

Figure 3 ACF (left) and PACF(right) of AMD

When I ran the Ljung Box test on the log returns, it failed to reject the null hypothesis (the model being stationary) at the 90% level for Bitcoin, and at the 99% confidence level for AMD, showing that this transformation helped to make the data more useable for an ARIMA analysis. The null hypothesis was rejected at the 99% confidence level for the original and log-transformed data.

My final step was creating a couple of preliminary ARIMA models. I used auto Arima, which determined that the best model for the Bitcoin series was a random walk with drift. The drift coefficient is significant at the 99% confidence level. Auto Arima also applied differencing.

```
## Series: log(btcts)
## ARIMA(0,1,0) with drift
##
```

```
## Coefficients:
##       drift
##      0.0023
## s.e.  0.0010
##
## sigma^2 estimated as 0.001752:  log likelihood=3194.47
## AIC=-6384.93   AICc=-6384.93   BIC=-6373.91
```

The auto ARIMA output for the AMD series was an AR(3) model with differencing. However, only the AR1 coefficient was significant.

```
## Series: log(amdts)
## ARIMA(3,1,0)

## z test of coefficients:
##
##      Estimate Std. Error z value  Pr(>|z|)
## ar1 -0.137856   0.029599 -4.6574 3.202e-06 ***
## ar2  0.058977   0.037038  1.5923    0.1113
## ar3 -0.021555   0.040894 -0.5271    0.5981
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The exploratory analysis opened a lot of ideas for the project for me. I plan to look into price/scarcity over time, if I can find quality data on the bitcoin supply. If substantially more people started mining the currency, then the price increase can be explained by scarcity and increased demand. This could potentially explain the drift I'm seeing. I also wonder if the model will have different/more accurate predictions if I focus only on 2020 onwards, which is when the behavior of the currency seemed to take a large shift.

I also plan on looking further into manual model building and potential seasonality. I ran into issues when dealing with missing values in the EACF plots, which is something I will work on.