Time Series Analysis of Selected Cryptocurrencies

Adam Hall & Matt Schrader
MA5781 - Time Series Analysis
Michigan Technological University - Spring 2021

Abstract

The goals of this study are to identify the optimal model for four different cryptocurrencies and to compare models for established cryptocurrencies, like Bitcoin and Litecoin, to younger cryptocurrencies, like Ethereum and Chainlink. The dataset was collected from www.coinmarketcap.com on February 3rd, 2021 and includes historical price data for the four cryptocurrencies mentioned previously. Exploration of the data reveals that each of the cryptocurrencies can be modeled by an AR(1) + μ process, suggesting that both established and young cryptocurrencies follow a similar trend that can be classified as a random walk with drift.

Abstract 1

Background 3

Data Exploration and Model Fitting 3

- Data Visualization 3
- Series Decomposition 5
  - Trend Analysis 6
  - Seasonal Analysis 9
- Autoregressive, Integrated, Moving Average (ARIMA) Exploration 9
- Cointegration Analysis 14

Discussion 16

Summary and Conclusions 17

References 17

Appendix A: R Code 19

Background

Over the past five years, there has been a surge in popularity for cryptocurrencies like Bitcoin. A cryptocurrency ("crypto" or "coin") is a digital currency that can be used to buy goods and services, but uses an online ledger with strong cryptography to secure online transactions. One expert on the topic of cryptocurrency is Marcus Swanepoel, the CEO of cryptocurrency trading app Luno. He said cryptocurrency has the ability to "give power back to the people. To eliminate current social structures and systems that disenfranchise individuals. To provide the foundations of a system that's transparent yet secure. Where corruption is exposed and rampant inflation ended."[1]

While some experts think that cryptocurrencies may one day replace fiat money, much of the current interest in these unregulated currencies is to trade for profit. In an attempt to better understand these digital assets, this paper investigates the following research questions:

- Do established cryptocurrencies (such as Bitcoin and Litecoin) follow similar time patterns?
- Do newer cryptocurrencies (like Ethereum and Chainlink) follow these same patterns? I.e.: Does 100 days on the market for a newer cryptocurrency match 100 days on the market of an established coin?

Data Exploration and Model Fitting

Data Visualization

Opening price data was scraped for four cryptocurrencies: Bitcoin[2], Litecoin[3], Ethereum[4], and Chainlink[5]. For the purposes of this project, Bitcoin and Litecoin were considered "established" cryptocurrencies, and Ethereum and Chainlink considered "young" cryptocurrencies. The data exploration started with a visualization of the opening price data over time, shown in Figures 1 and 2. Due to the large value of Bitcoin compared to the other coins, it is difficult to discern any real trends in the time series data. The data in Figure 2 is shown on a logarithmic scale for the y axis in order to see the detail of each coin. This plot shows that each coin seems to generally follow similar patterns: a) each coin has been increasing in value over time, and b) each coin looks to have spikes and dips at generally the same time. Figure 2 right away suggests answers for the research questions - it appears that not only do established cryptocurrencies follow similar time patterns, but the younger coins follow these time patterns as well, so long as the comparison is made using the same starting date for each coin. Further analysis was performed to investigate these speculations.

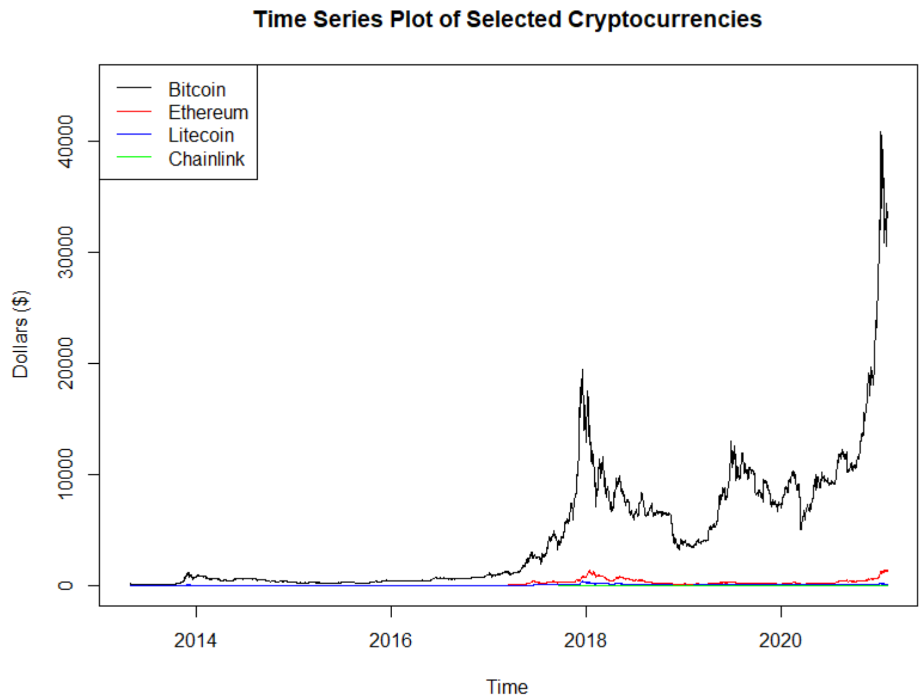

Figure 1: Cryptocurrency Opening Prices (Linear Scale)

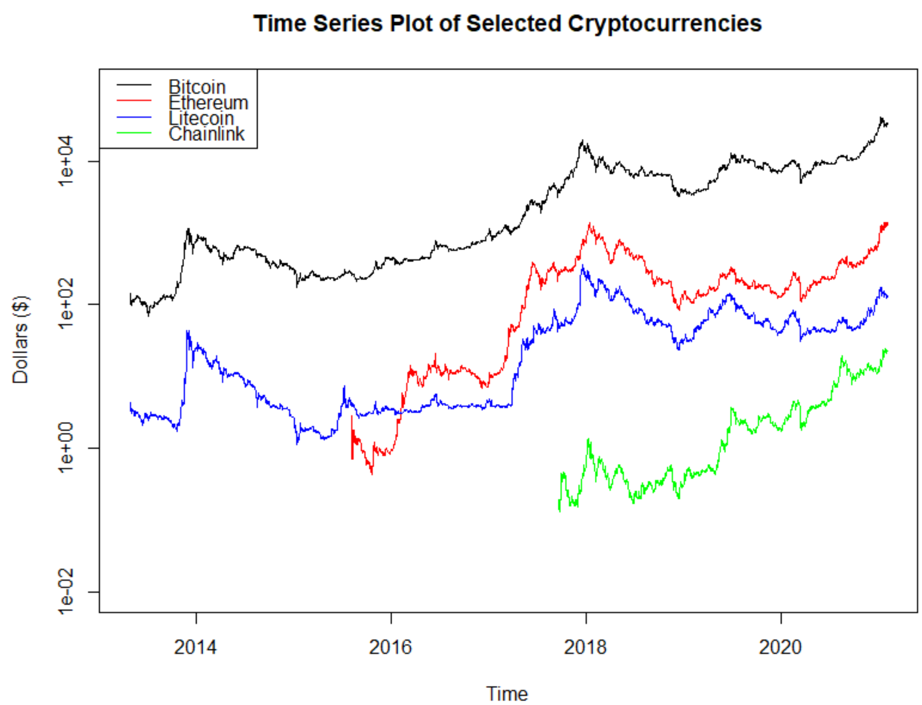

Figure 2: Cryptocurrency Opening Prices (Log Scale)

Series Decomposition

Series decomposition plots were generated in order to gain some insight into whether a linear or seasonal trend might exist for each of the four cryptocurrencies. These are shown in Figures 3 through 6.

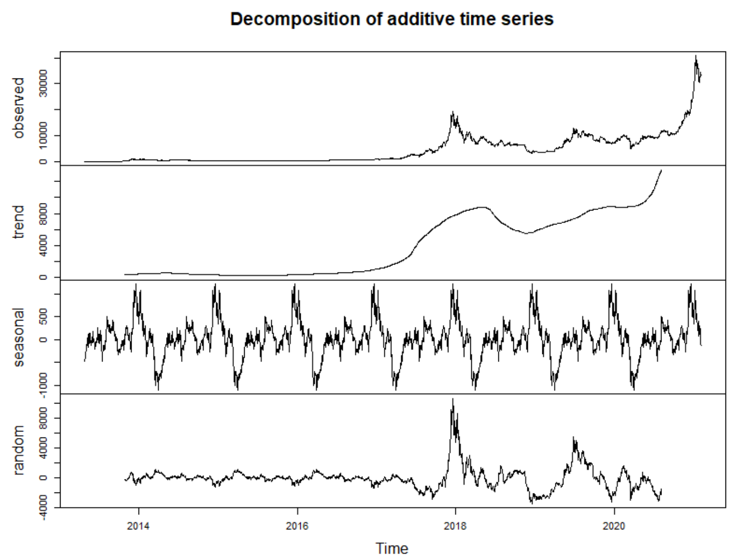

Figure 3: Time Series Decomposition of Bitcoin Data

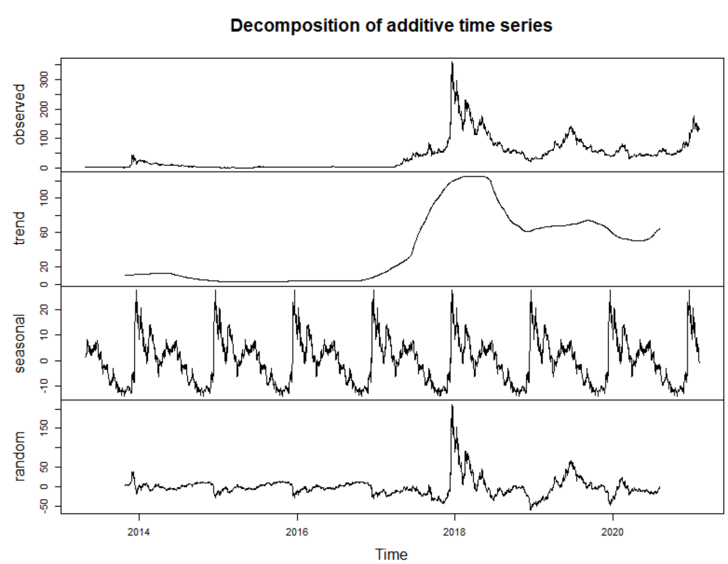

Figure 4: Time Series Decomposition of Litecoin Data

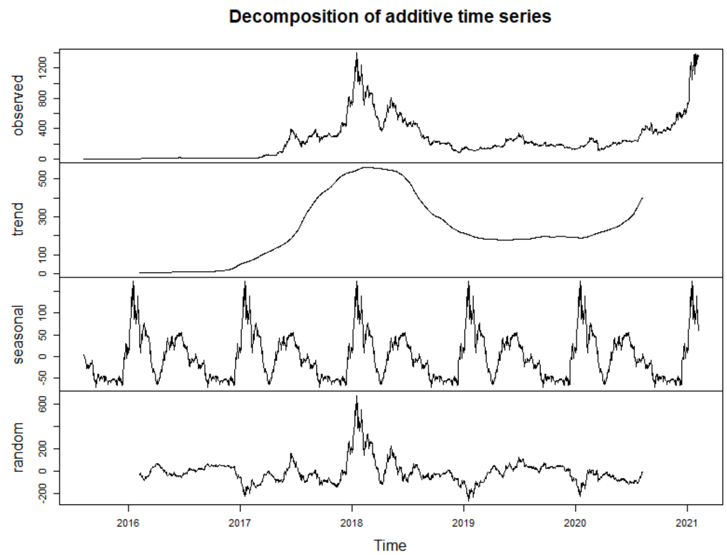

Figure 5: Time Series Decomposition of Ethereum Data

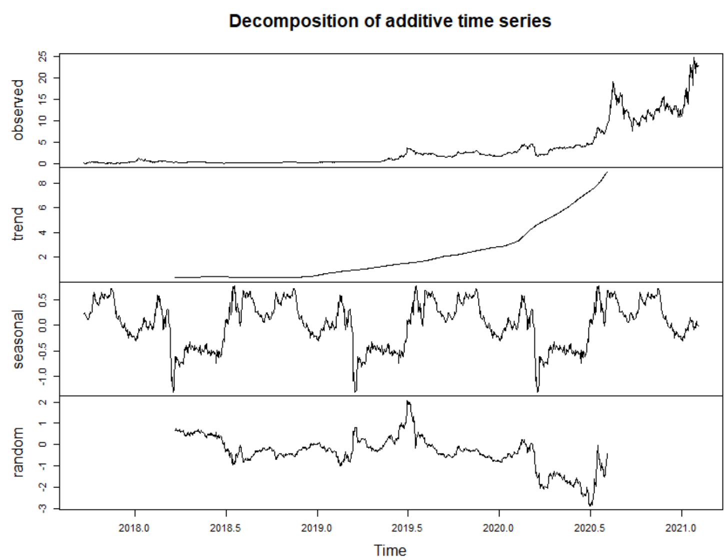

Figure 6: Time Series Decomposition of Chainlink Data

In these plots, there looks to be a possible exponential growth pattern, especially seen in the Chainlink data. This looks less so in the other three, mainly due to large humps in the middle of the series at the end of 2017 and beginning of 2018, where Chainlink was just starting. The seasonal trends in each are less apparent, but also analyzed to rule out seasonality just being masked by the change in value over time.

Trend Analysis

Exponential fits were performed for each of the cryptocurrencies. The model summaries of each of these fits are in Table 1.

Table 1: Exponential Models of Cryptocurrency Data

	Bitcoin			Litecoin		
Model	Est.	S.E.	P(>|t|)	Est.	S.E.	P(>|t|)
β0	-1300	10.86	$<2x10^{-16}$	-1024	16.53	$<2x10^{-16}$
β1	0.648	$5.386x10^{-3}$	$<2x10^{-16}$	0.509	$8.193x10^{-3}$	$<2x10^{-16}$
Model *p*-value	$<2x10^{-16}$			$<2x10^{-16}$		
	Ethereum			Chainlink		
Model	Est.	S.E.	P(>|t|)	Est.	S.E.	P(>|t|)
β0	-1932	36.16	$<2x10^{-16}$	-2644	32.61	$<2x10^{-16}$
β1	0.9594	$1.792x10^{-2}$	$<2x1^{0-16}$	1.309	$1.615x10^{-2}$	$<2x10^{-16}$
Model *p*-value	$<2x10^{-16}$			$<2x10^{-16}$		

The residuals plots of each of these fits are shown in Figure 7. These models do not appear to fit the data very well, and do not seem to remove the time dependence from the data.

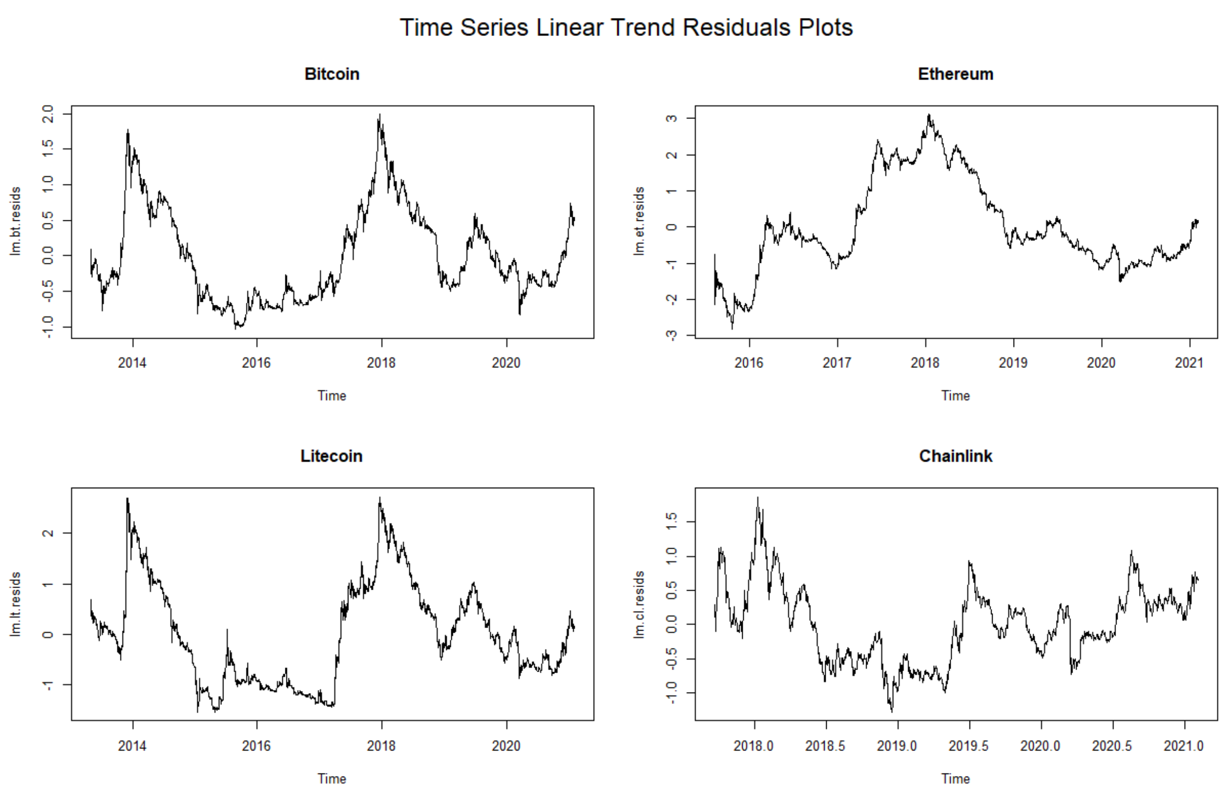

Figure 7: Residuals Plots from Exponential Model Fits

Autocorrelation function (ACF) plots were generated to analyze the time dependence of the residuals. The ACF plots are shown in Figure 8, and show at very slow decay across the lags. This kind of decay is indicative of an autoregressive process with a φ value close to one and a lack of stationarity in the residuals.

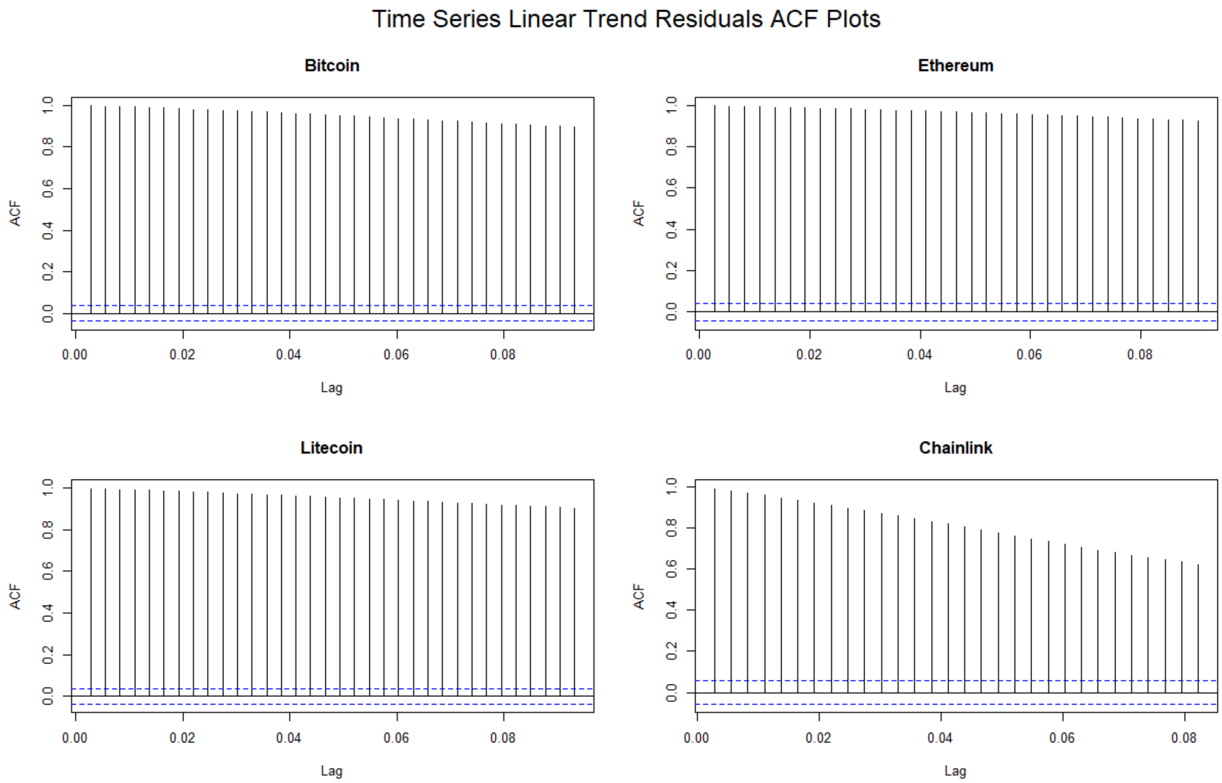

Figure 8: Exponential Model Residuals ACF Plots

The residuals were further tested for stationarity using Augmented Dickey-Fuller (ADF) tests and Phillips-Perron (PP) tests for unit roots. The results of these tests are shown in Table 2. The *p*-values for both tests for each model are well above the 0.05 significance level, indicating that each set of residuals is not stationary, or has a unit root.

Table 2: ADF and PP Unit Root Tests p-Values

Model	Bitcoin	Litecoin	Ethereum	Chainlink
ADF	0.5524	0.7640	0.5700	0.2079
PP	0.7547	0.9285	0.7294	0.3638

To analyze the quality of the fit of the models, normal quantile-quantile (QQ) plots were generated, shown in Figure 9. These plots show generally heavy tails or poor fits. The best fit is with Chainlink, though this also shows nonlinearity near the lower end of the plot.

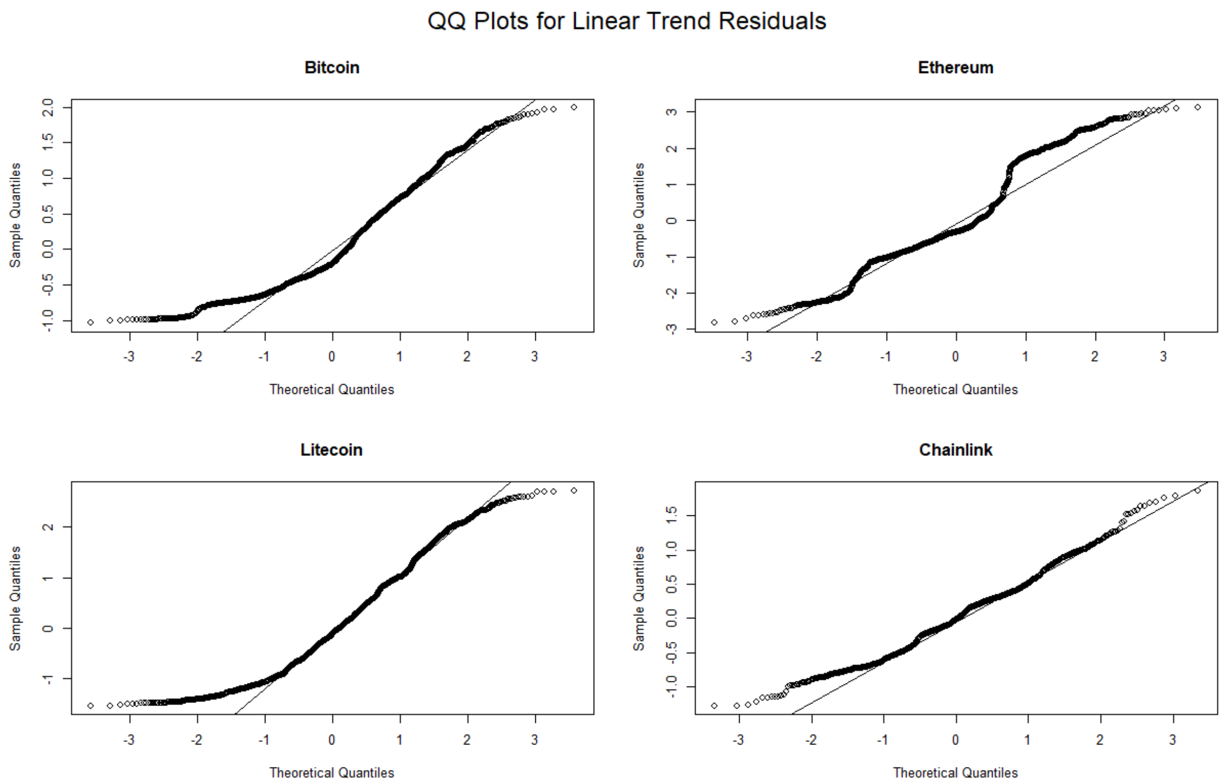

Figure 9: Exponential Model Residuals QQ Plots

Shapiro-Wilk tests were performed as well, with *p*-values of <2.2×10^{-16} for Bitcoin, Litecoin, and Ethereum, and 8.122×10^{-10} for Chainlink, each providing strong evidence against the null hypothesis of normality in the residuals for each model. Seasonality was next considered to explain the data.

Seasonal Analysis

Seasonality was visually assessed using the decomposition plots in Figures 3 through 6. The plot of the data had little, if any, match to the seasonality plots in the figures. It was therefore assumed that the data had no interesting seasonal trends, but that seasonality would be readdressed should it appear during further analysis.

Autoregressive, Integrated, Moving Average (ARIMA) Exploration

Since the linear and seasonal trend models did not fit the data very well, as seen in Figure 9 with the poor QQ plot, but the ACF plot of the residuals from the linear trend indicated a probable unit-root for an ARIMA(1, 0, 0) model, ARIMA exploration was done on the first difference of the time series data. This data can be seen in Figure 10

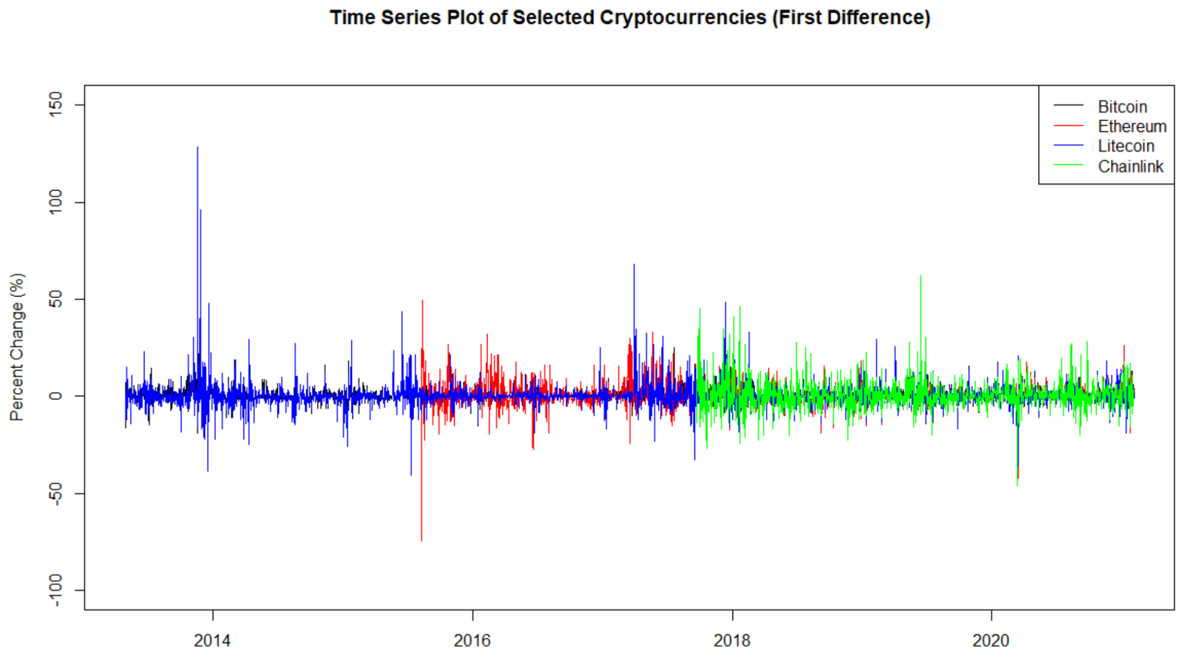

Figure 10: Time Series of First Differenced Data

The plot in Figure 10 suggests that taking the first difference of the data has caused the data to become a stationary process. Decomposition plots for each cryptocurrency are shown in Figures 11 through 14.

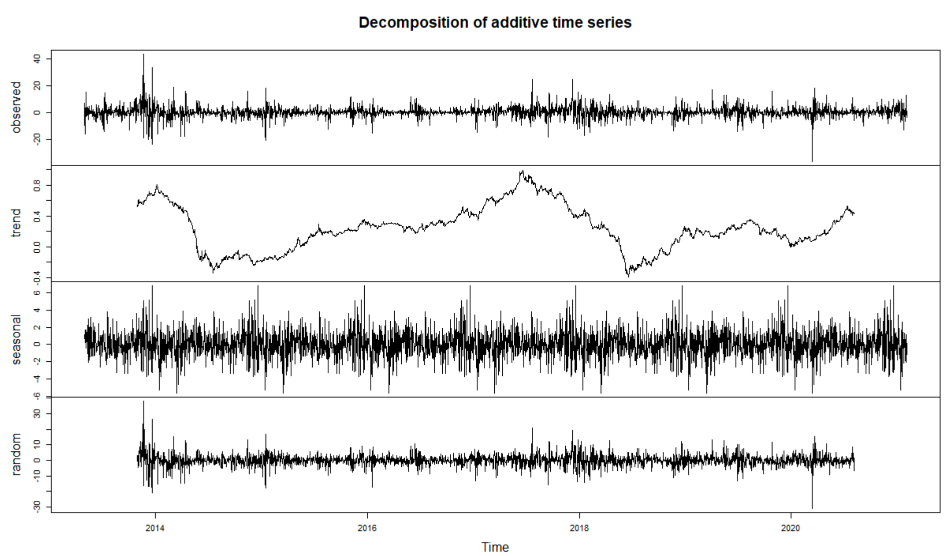

Figure 11: Decomposition Plot of Bitcoin First Difference

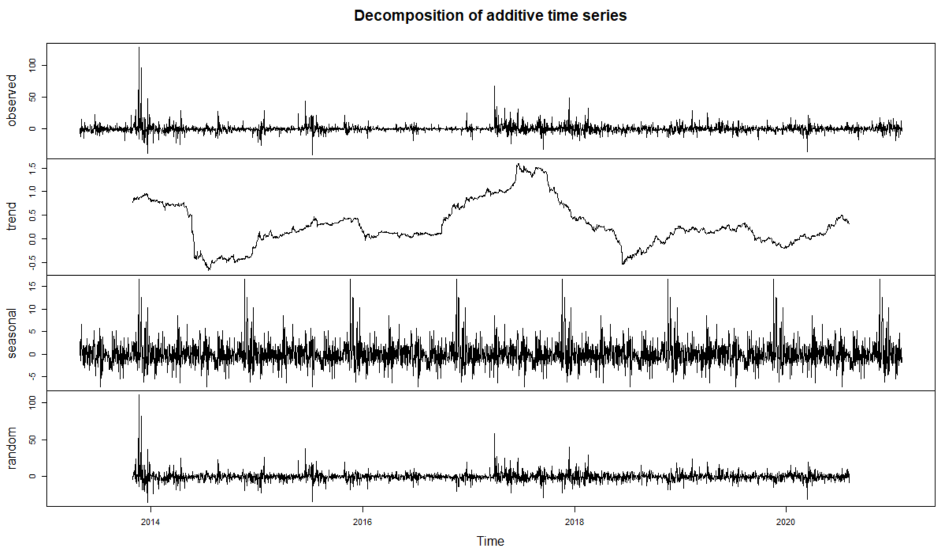

Figure 12: Decomposition Plot of Litecoin First Difference

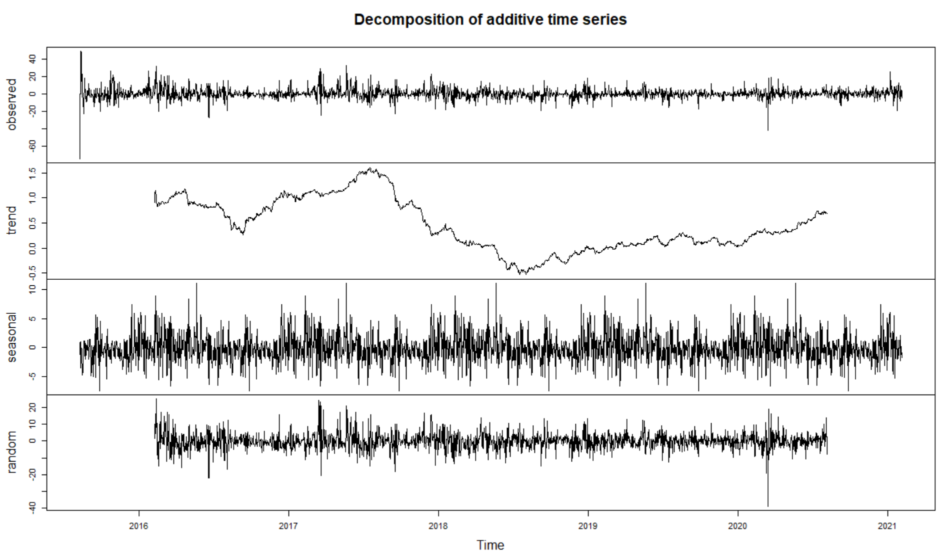

Figure 13: Decomposition Plot of Ethereum First Difference

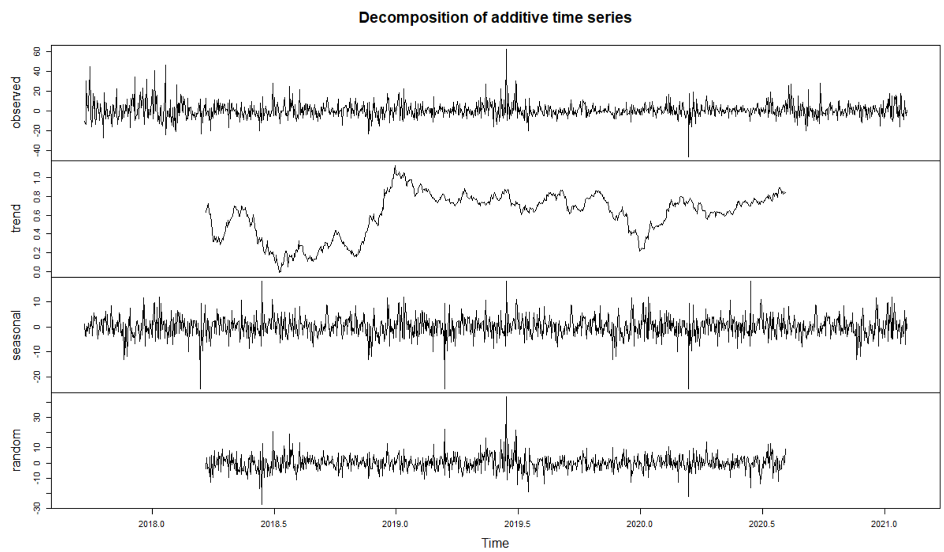

Figure 14: Decomposition Plot of Chainlink First Difference

The decomposition plots show that for each of the cryptocurrencies, the first differenced data does not have a trend or seasonal component to the data. The best fit of the data looks to be the random noise element. ACF, Partial Autocorrelation Function (PACF), and Extended Autocorrelation Function (EACF) plots were generated next, shown in Figures 15, 16, and 17, respectively. These plots each indicate an ARIMA(0, 0, 0) process for the first differenced data - suggesting these data sets are now random walk data sets.

First Difference ACF Plots

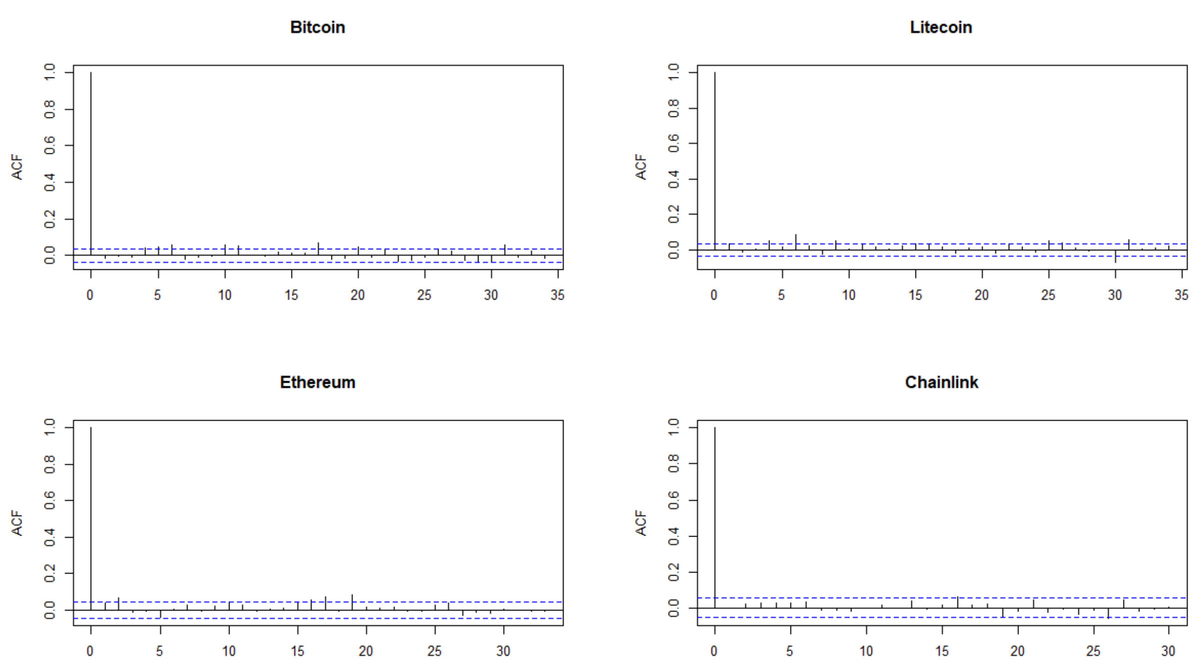

Figure 15: ACF Plots of Cryptocurrency First Differenced Data

First Difference PACF Plots

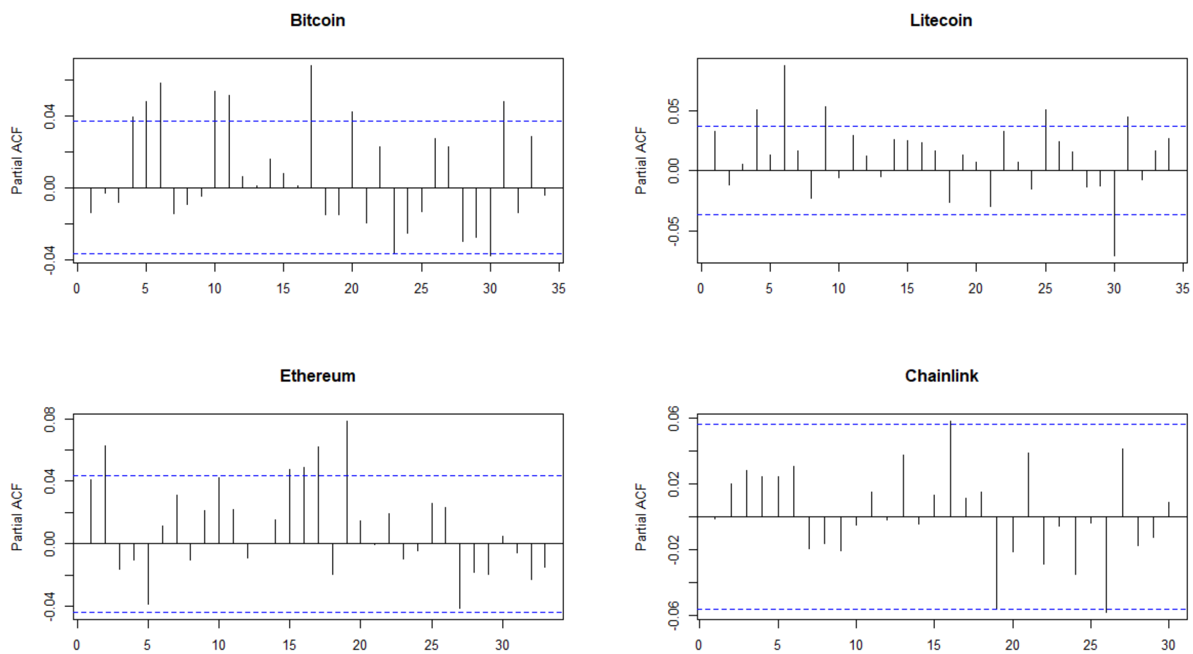

Figure 16: PACF Plots of Cryptocurrency First Differenced Data

First Difference EACF Plots

Bitcoin

AR/MA	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	o	o	o	x	x	x	o	o	o	x	x	o	o	o
1	x	o	o	o	o	x	o	o	o	o	x	o	o	o
2	x	o	o	o	o	x	o	o	o	o	x	o	o	o
3	x	x	x	o	o	x	o	o	o	o	o	o	o	o
4	x	o	x	x	o	o	o	o	o	o	o	o	o	o
5	x	o	x	x	x	o	o	o	o	o	o	o	o	o
6	x	x	x	x	x	o	o	o	o	o	o	o	o	o
7	x	o	o	x	x	x	x	o	o	o	o	o	o	o

Litecoin

AR/MA	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	o	o	o	x	o	x	o	o	x	o	o	o	o	o
1	x	o	o	x	o	x	x	o	x	o	o	o	o	o
2	x	o	o	o	o	x	o	o	o	o	o	o	o	o
3	x	x	x	o	o	x	o	x	o	o	o	o	o	o
4	x	x	x	x	o	x	o	x	o	o	o	o	o	o
5	x	x	x	x	x	x	o	o	o	o	o	o	o	o
6	x	x	x	x	x	x	o	o	o	x	o	o	o	o
7	x	o	x	x	x	x	x	x	o	x	o	o	o	o

Ethereum

AR/MA	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	o	x	o	o	o	o	o	o	o	o	o	o	o	o
1	x	x	o	o	o	o	o	o	o	o	o	o	o	o
2	x	x	o	o	o	o	o	o	o	o	o	o	o	o
3	x	o	x	o	o	o	o	o	o	o	o	o	o	o
4	x	o	x	o	o	o	o	o	o	o	o	o	o	o
5	x	x	o	x	o	o	o	o	o	o	o	o	o	o
6	x	x	x	x	x	o	o	o	o	o	o	o	o	o
7	x	x	x	x	x	x	x	o	o	o	o	o	o	o

Chainlink

AR/MA	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	o	o	o	o	o	o	o	o	o	o	o	o	o	o
1	o	o	o	o	o	o	o	o	o	o	o	o	o	o
2	x	x	o	o	o	o	o	o	o	o	o	o	o	o
3	x	x	x	o	o	o	o	o	o	o	o	o	o	o
4	x	x	x	x	o	o	o	o	o	o	o	o	o	o
5	x	x	x	x	x	o	o	o	o	o	o	o	o	o
6	x	x	x	x	x	x	o	o	o	o	o	o	o	o
7	x	o	x	x	o	o	x	o	o	o	o	o	o	o

Figure 17: EACF Plots of Cryptocurrency First Differenced Data

Next, R's auto-ARIMA function was used to search for possible ARIMA models, again using the Bayesian Information Criterion (BIC) as the selection criteria. The results of this function are in Table 3, along with fits to ARIMA(0, 0, 0) models for comparison. The results in Table 3 show agreement on ARIMA(0, 0, 0) models for each of the cryptocurrencies, except for Ethereum, where the auto-ARIMA function proposed an ARIMA(1, 0, 0) model. However, the results of the ARIMA(0, 0, 0) fit are fairly equivalent to this, and would give a more parsimonious model. The ARIMA(0, 0, 0) fits also produced mean values of the data, which are shown in Table 4, along with their 95% confidence intervals.

Table 3: Auto-ARIMA Model Fits and Comparison

	ARIMA Model	BIC	ARIMA(0, 0, 0) BIC
Bitcoin	ARIMA(0, 0, 0)	16265.14	16265.14
Litecoin	ARIMA(0, 0, 0)	18982.49	18982.49
Ethereum	ARIMA(1, 0, 0)	13187.24	13183.22
Chainlink	ARIMA(0, 0, 0)	8604.94	8604.94

The results in Table 4 show that each of the first differenced data sets have a statistically significant mean at the 0.05 significance level. This indicates that each of the original data sets can then be classified as an AR(1) + μ process, or random walk with drift. The positive means fits well with the plots shown in Figure 2, as the positive mean indicates an overall upward drift, which was seen for each crypto in the plot. Further analysis is required to determine if the apparent similar changes in each crypto are actually shared among them.

Table 4: Data Fit Results with AR(1) Process

	μ	Std. Error	95% C.I.
Bitcoin	0.2849	0.0797	(0.1287, 0.4412)
Litecoin	0.3291	0.1288	(0.0768, 0.5815)
Ethereum	0.5307	0.1442	(0.2482, 0.8133)
Chainlink	0.6943	0.2267	(0.2500, 1.1386)

Cointegration Analysis

Cointegration is the concept that though two (or possibly more) series move independently, the average distance between them remains relatively constant[6]. If it is possible to form a linear combination of multiple non-stationary series that is stationary, then the multiple non-stationary series are said to be cointegrated. From looking at Figure 2, it seems reasonable that the log of Bitcoin and the log of Litecoin appear to exhibit cointegration.

After fitting an OLS model for log of Bitcoin against log of Litecoin, an ADF test was performed on the residuals to test for stationarity. The linear fit of the data can be seen in Figure 18.

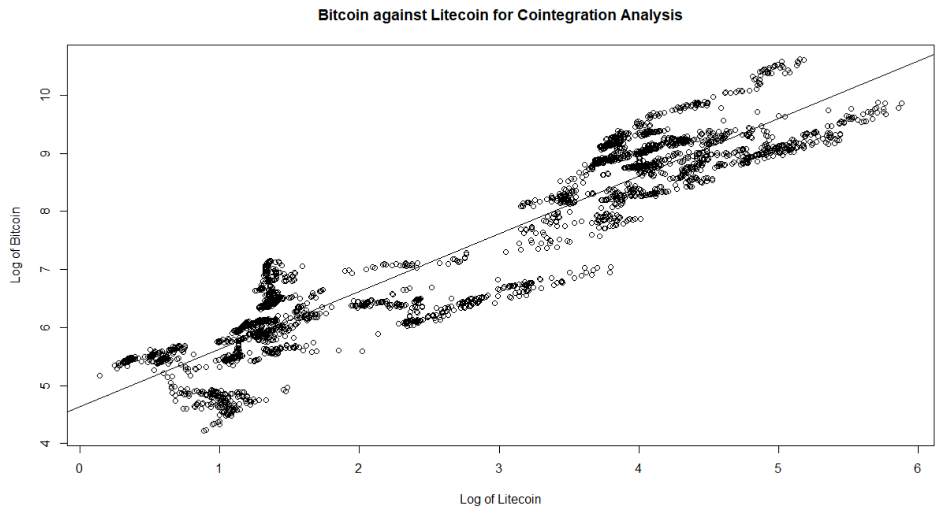

Figure 18: Log of Bitcoin against Log of Litecoin for Cointegration Analysis

There appears to be some correlation between the two data sets, as seen with the line through the data. However, the residuals, shown in Figure 19, were not made stationary using this technique.

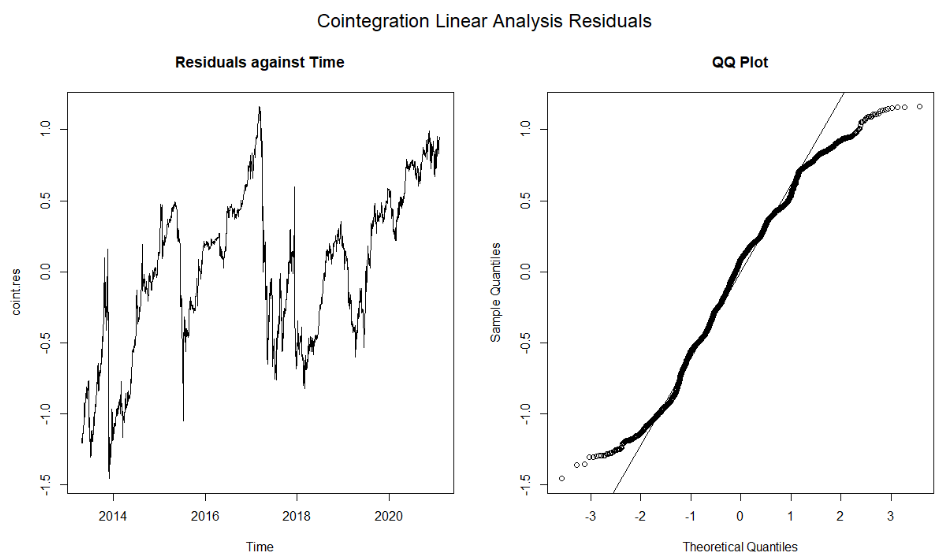

Figure 19: Residuals of Cointegration Analysis of Log of Bitcoin and Log of Litecoin

This implies that the data sets are not cointegrated. The p-value from the ADF test was 0.2181, which suggests that this new series has a unit root and thus agrees that the log of Bitcoin and the log of Litecoin are not cointegrated. Another analysis for cointegration was performed using the egcm function from R's egcm library, which performs the Engle-Granger cointegration procedure on a pair of time-series data sets. This was used on the Bitcoin and Litecoin data, and yielded the result that the two do not appear to be cointegrated. The results of this function are shown in Figure 20. The residuals plot looks to match that of Figure 19, and shows again the lack of cointegration between the two data sets.

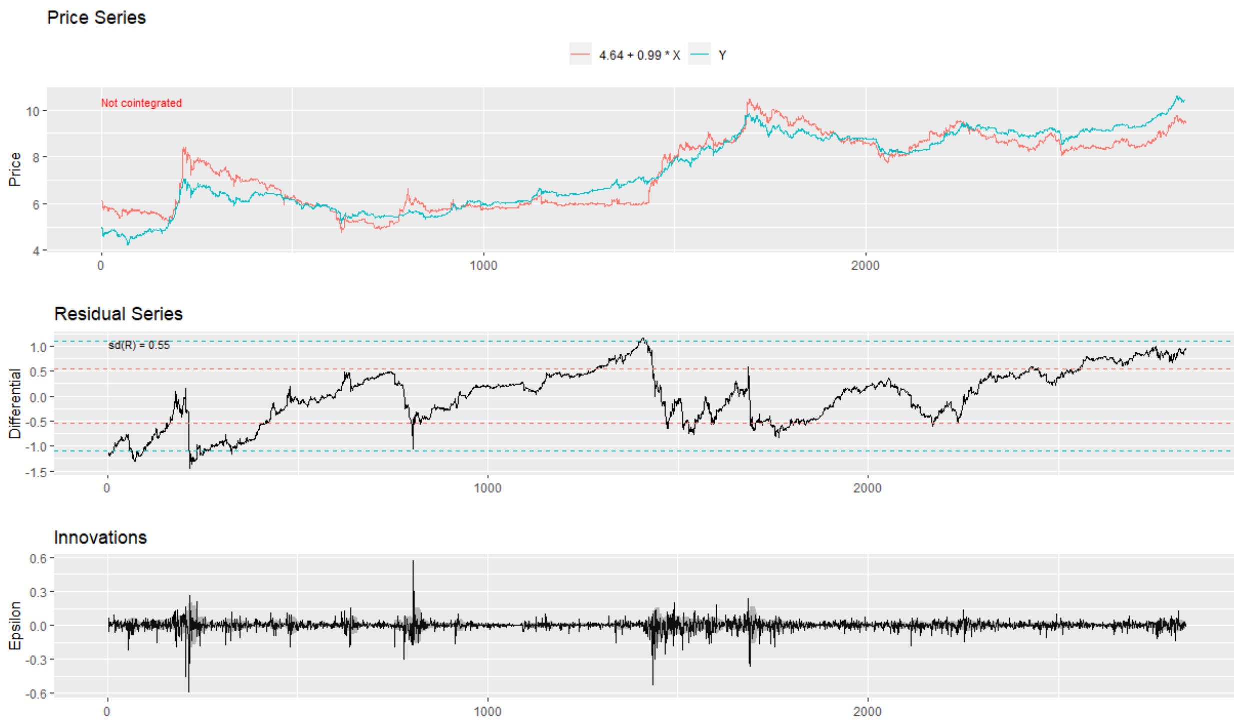

Figure 20: Engle-Granger Cointegration Model Results

Discussion

The data sets have been fit using random walk with drift models. The models look to fit the data well, with statistical significance shown for unit roots for each data set, and statistically significant means for each of the first differenced data sets. Tying these results back into the initial research questions:

- Do established cryptocurrencies (such as Bitcoin and Litecoin) follow similar time patterns?

Established cryptocurrencies do appear to follow similar time patterns. This was first evidenced in Figure 2 with detail, where the time series plots had similar spikes and dips during the same time periods. Furthermore, the two established coins analyzed (Bitcoin and Litecoin) have similar models, being random walks with drift. However, further analysis showed that these do not appear to maintain a somewhat constant distance between them (i.e., are not cointegrated).

- Do newer cryptocurrencies (like Ethereum and Chainlink) follow these same patterns? I.e.: Does 100 days on the market for a newer cryptocurrency match 100 days on the market of an established coin?

The newer cryptocurrencies analyzed (Ethereum and Chainlink) do appear to follow the same style of patterns as their established counterparts. These both also fit well using random walks with drift. As seen in Figure 2, they also had similar spikes and dips as the established coins during the same times. It should be noted from Table 4 that the 95% confidence intervals for Bitcoin and Litecoin contain each other's means, indicating that their relative increase in value is not significantly different from each other at the 0.05 significance level. However, the Bitcoin confidence interval does not contain the means of either Ethereum or Chainlink, and the Litecoin confidence interval does not contain the mean of Chainlink. This indicates that both of the newer coins could show a higher relative increase in value compared to Bitcoin, and Chainlink could show a higher relative increase in value compared to Litecoin.

Possible questions for future research could include:

- Does the amount of drift change over time?
- If drift values do change:
  - How much change is there in the drift values?
  - Do different cryptocurrencies experience the same changes in drift values over time?
  - Could the stability of a selected cryptocurrency be predicted?

Summary and Conclusions

Times series analysis of four cryptocurrencies was performed. Trend and seasonal analysis was conducted and found inapplicable to the data. Various ARIMA models were fit, and a random walk with drift model was determined to be the best for each coin. It was determined that established and new coins generally follow the same pattern as time progresses, given that market forces affect them equally and they are compared at the same point in time. It was also determined that the newer coins analyzed do have relative increases in value that are statistically significantly different from their established counterparts.

References

1. Conklin, Audrey. “What are the benefits of cryptocurrency”, *Fox Business*, February 6, 2020, https://www.foxbusiness.com/money/what-are-the-benefits-of-cryptocurrency, accessed 02/22/2021
2. “Historical Data for Bitcoin.”, *CoinMarketCap*, 2021, https://coinmarketcap.com/currencies/bitcoin/historical-data/, accessed 02/03/2021
3. “Historical Data for Litecoin.”, *CoinMarketCap*, 2021, https://coinmarketcap.com/currencies/litecoin/historical-data/, accessed 02/03/2021
4. “Historical Data for Ethereum.”, *CoinMarketCap*, 2021, https://coinmarketcap.com/currencies/ethereum/historical-data/, accessed 02/03/2021

5. “Historical Data for Chainlink.”, *CoinMarketCap*, 2021, https://coinmarketcap.com/currencies/chainlink/historical-data/, accessed 02/03/2021
6. Goldkamp, Jeff. “Cointegration for Time Series Analysis.”, *Analytics Vidhya*, February 9, 2020, https://medium.com/analytics-vidhya/cointegration-for-time-series-analysis-1d0a758a20f1, accessed 02/23/2021

Appendix A: R Code

```
library(TSA)
library(tseries)
library(forecast)
library(egcm)

# Import data
ETH = read.csv("D:/Shared drives/MA5781/Ethereum.csv")
ethereum.ts = ts(data=rev(ETH[,3]), frequency=365, start=c(2015, 220))

LTC = read.csv("D:/Shared drives/MA5781/Litecoin.csv")
litecoin.ts = ts(data=rev(LTC[,3]), frequency=365, start=c(2013, 119))

BTC = read.csv("D:/Shared drives/MA5781/Bitcoin.csv")
bitcoin.ts = ts(data=rev(BTC[,3]), frequency=365, start=c(2013, 119))

LINK = read.csv("D:/Shared drives/MA5781/Chainlink.csv")
chainlink.ts = ts(data=rev(LINK[,3]), frequency=365, start=c(2017, 264))

# Time series plot - linear scale
par(mfrow=c(1,1))
plot(bitcoin.ts, type='l', ylim=c(0.01, 45000), main="Time Series Plot of Selected
Cryptocurrencies", ylab="Dollars ($)", xlab="Time")
lines(ethereum.ts, col='red')
lines(litecoin.ts, col='blue')
lines(chainlink.ts, col='green')
legend('topleft', col=c('black','red','blue','green'),
legend=c('Bitcoin','Ethereum','Litecoin','Chainlink'), lty=1)

# Time series plot - log scale
par(mfrow=c(1,1))
plot(bitcoin.ts, type='l', log="y", ylim=c(0.01, 100000), main="Time Series Plot of
Selected Cryptocurrencies", ylab="Dollars ($)", xlab="Time")
lines(ethereum.ts, col='red')
lines(litecoin.ts, col='blue')
lines(chainlink.ts, col='green')
legend('topleft', col=c('black','red','blue','green'),
legend=c('Bitcoin','Ethereum','Litecoin','Chainlink'), lty=1)

# Decomposition plots to see what kinds of trends may exist
decomp.bt = decompose(bitcoin.ts)
plot(decomp.bt)
decomp.lt = decompose(litecoin.ts)
plot(decomp.lt)
decomp.et = decompose(ethereum.ts)
plot(decomp.et)
decomp.cl = decompose(chainlink.ts)
plot(decomp.cl)

# Try removing possible log-linear trend
lm.bt = lm(log(bitcoin.ts)~time(bitcoin.ts))
lm.et = lm(log(ethereum.ts)~time(ethereum.ts))
lm.lt = lm(log(litecoin.ts)~time(litecoin.ts))
```

```
lm.cl = lm(log(chainlink.ts)~time(chainlink.ts))

summary(lm.bt)
summary(lm.et)
summary(lm.lt)
summary(lm.cl)

lm.bt.resids = ts(resid(lm.bt), frequency=365, start=c(2013, 119))
lm.et.resids = ts(resid(lm.et), frequency=365, start=c(2015, 220))
lm.lt.resids = ts(resid(lm.lt), frequency=365, start=c(2013, 119))
lm.cl.resids = ts(resid(lm.cl), frequency=365, start=c(2017, 264))

# Residuals Plots from Linear Trend Removal
par(mfrow=c(2,2), oma=c(0, 0, 2, 0))
plot(lm.bt.resids, main="Bitcoin")
plot(lm.et.resids, main="Ethereum")
plot(lm.lt.resids, main="Litecoin")
plot(lm.cl.resids, main="Chainlink")
mtext("Time Series Linear Trend Residuals Plots", outer = TRUE, cex = 1.5)

# ACF Plots of Residuals from Linear Trend Removal
par(mfrow=c(2,2), oma=c(0, 0, 2, 0))
acf(lm.bt.resids, main="Bitcoin")
acf(lm.et.resids, main="Ethereum")
acf(lm.lt.resids, main="Litecoin")
acf(lm.cl.resids, main="Chainlink")
mtext("Time Series Linear Trend Residuals ACF Plots", outer = TRUE, cex = 1.5)

# Runs Tests on Linear Trend Residuals - These just give errors
runs(lm.bt.resids)
runs(lm.et.resids)
runs(lm.lt.resids)
runs(lm.cl.resids)

# Augmented Dickey-Fuller Tests on Linear Trend Residuals
adf.test(lm.bt.resids)
adf.test(lm.et.resids)
adf.test(lm.lt.resids)
adf.test(lm.cl.resids)

# PP Tests on Linear Trend Residuals
pp.test(lm.bt.resids)
pp.test(lm.et.resids)
pp.test(lm.lt.resids)
pp.test(lm.cl.resids)

# QQ Plots for Linear Trend Residuals
par(mfrow=c(2,2), oma=c(0, 0, 2, 0))
qqnorm(lm.bt.resids, main="Bitcoin"); qqline(lm.bt.resids)
qqnorm(lm.et.resids, main="Ethereum"); qqline(lm.et.resids)
qqnorm(lm.lt.resids, main="Litecoin"); qqline(lm.lt.resids)
qqnorm(lm.cl.resids, main="Chainlink"); qqline(lm.cl.resids)
mtext("QQ Plots for Linear Trend Residuals", outer = TRUE, cex = 1.5)

# Shapiro-Wilk Test on Linear Trend Residuals
```

```
shapiro.test(lm.bt.resids)
shapiro.test(lm.et.resids)
shapiro.test(lm.lt.resids)
shapiro.test(lm.cl.resids)

# Try removing possible log-seasonal trend
lm.s.bt = lm(log(bitcoin.ts)~season(bitcoin.ts)+time(bitcoin.ts))
lm.s.et = lm(log(ethereum.ts)~season(ethereum.ts)+time(ethereum.ts))
lm.s.lt = lm(log(litecoin.ts)~season(litecoin.ts)+time(litecoin.ts))
lm.s.cl = lm(log(chainlink.ts)~season(chainlink.ts)+time(chainlink.ts))

lm.s.bt.resids = ts(resid(lm.bt), frequency=365, start=c(2013, 119))
lm.s.et.resids = ts(resid(lm.et), frequency=365, start=c(2015, 220))
lm.s.lt.resids = ts(resid(lm.lt), frequency=365, start=c(2013, 119))
lm.s.cl.resids = ts(resid(lm.cl), frequency=365, start=c(2017, 264))

# QQ Plots for Seasonal Trend Residuals
par(mfrow=c(2,2), oma=c(0, 0, 2, 0))
qqnorm(lm.s.bt.resids, main="Bitcoin"); qqline(lm.bt.resids)
qqnorm(lm.s.et.resids, main="Ethereum"); qqline(lm.et.resids)
qqnorm(lm.s.lt.resids, main="Litecoin"); qqline(lm.lt.resids)
qqnorm(lm.s.cl.resids, main="Chainlink"); qqline(lm.cl.resids)
mtext("QQ Plots for Seasonal Trend Residuals", outer = TRUE, cex = 1.5)

# Shapiro-Wilk Test on Seasonal Trend Residuals
shapiro.test(lm.s.bt.resids)
shapiro.test(lm.s.et.resids)
shapiro.test(lm.s.lt.resids)
shapiro.test(lm.s.cl.resids)

# The linear trend ACF plots suggest an AR(1) with phi=1 (RW) - take first difference
dbitcoin.ts = na.omit(100*(bitcoin.ts-zlag(bitcoin.ts))/zlag(bitcoin.ts))
dlitecoin.ts = na.omit(100*(litecoin.ts-zlag(litecoin.ts))/zlag(litecoin.ts))
dethereum.ts = na.omit(100*(ethereum.ts-zlag(ethereum.ts))/zlag(ethereum.ts))
dchainlink.ts = na.omit(100*(chainlink.ts-zlag(chainlink.ts))/zlag(chainlink.ts))

# Time series plot of first difference
par(mfrow=c(1,1))
plot(dbitcoin.ts, type='l', ylim=c(-100, 150), main="Time Series Plot of Selected
Cryptocurrencies (First Difference)", ylab="Percent Change (%)", xlab="Time")
lines(dethereum.ts, col='red')
lines(dlitecoin.ts, col='blue')
lines(dchainlink.ts, col='green')
legend('topright', col=c('black','red','blue','green'),
legend=c('Bitcoin','Ethereum','Litecoin','Chainlink'), lty=1)

# Decomposition plots of first difference
ddecomp.bt = decompose(dbitcoin.ts)
plot(ddecomp.bt)
ddecomp.lt = decompose(dlitecoin.ts)
plot(ddecomp.lt)
ddecomp.et = decompose(dethereum.ts)
plot(ddecomp.et)
ddecomp.cl = decompose(dchainlink.ts)
plot(ddecomp.cl)
```

```
# ACF Plots
dbtcacf = acf(dbitcoin.ts, plot=FALSE, drop.lag.0=FALSE)
dbtcacf$lag = dbtcacf$lag*365
dltcacf = acf(dlitecoin.ts, plot=FALSE, drop.lag.0=FALSE)
dltcacf$lag = dltcacf$lag*365
detcacf = acf(dethereum.ts, plot=FALSE, drop.lag.0=FALSE)
detcacf$lag = detcacf$lag*365
dclcacf = acf(dchainlink.ts, plot=FALSE, drop.lag.0=FALSE)
dclcacf$lag = dclcacf$lag*365
par(mfrow=c(2,2), oma=c(0, 0, 2, 0))
plot(dbtcacf, main="Bitcoin")
plot(dltcacf, main="Litecoin")
plot(detcacf, main="Ethereum")
plot(dclcacf, main="Chainlink")
mtext("First Difference ACF Plots", outer = TRUE, cex = 1.5)

# PACF Plots
pdbtcacf = pacf(dbitcoin.ts, plot=FALSE, drop.lag.0=FALSE)
pdbtcacf$lag = pdbtcacf$lag*365
pdltcacf = pacf(dlitecoin.ts, plot=FALSE, drop.lag.0=FALSE)
pdltcacf$lag = pdltcacf$lag*365
pdetcacf = pacf(dethereum.ts, plot=FALSE, drop.lag.0=FALSE)
pdetcacf$lag = pdetcacf$lag*365
pdclcacf = pacf(dchainlink.ts, plot=FALSE, drop.lag.0=FALSE)
pdclcacf$lag = pdclcacf$lag*365
par(mfrow=c(2,2), oma=c(0, 0, 2, 0))
plot(pdbtcacf, main="Bitcoin")
plot(pdltcacf, main="Litecoin")
plot(pdetcacf, main="Ethereum")
plot(pdclcacf, main="Chainlink")
mtext("First Difference PACF Plots", outer = TRUE, cex = 1.5)

# EACF Plots
eacf(dbitcoin.ts)
eacf(dlitecoin.ts)
eacf(dethereum.ts)
eacf(dchainlink.ts)

# Auto-ARIMA for suggestions
auto.arima(dbitcoin.ts, ic='bic') # Suggests Arima(0, 0, 0) # BIC = 16265.14
auto.arima(dlitecoin.ts, ic='bic') # Suggests Arima(0, 0, 0) # BIC = 18983.92
auto.arima(dethereum.ts, ic='bic') # Suggests Arima(1, 0, 0) # BIC = 13187.24
auto.arima(dchainlink.ts, ic='bic') # Suggests Arima(0, 0, 0) # BIC = 8604.94

# The first difference plots suggest RW - let's fit a model DYt = mu + et
bt_fd_mle = Arima(dbitcoin.ts, order=c(0, 0, 0), method="ML"); bt_fd_mle # BIC =
16265.14
lt_fd_mle = Arima(dlitecoin.ts, order=c(0, 0, 0), method="ML"); lt_fd_mle # BIC =
18983.92
et_fd_mle = Arima(dethereum.ts, order=c(0, 0, 0), method="ML"); et_fd_mle # BIC =
13183.22
cl_fd_mle = Arima(dchainlink.ts, order=c(0, 0, 0), method="ML"); cl_fd_mle # BIC =
8604.94
bt_fd_mle$coef+c(-1,1)*qnorm(1-0.025)*as.vector(sqrt(bt_fd_mle$var.coef))
```

```
lt_fd_mle$coef+c(-1,1)*qnorm(1-0.025)*as.vector(sqrt(lt_fd_mle$var.coef))
et_fd_mle$coef+c(-1,1)*qnorm(1-0.025)*as.vector(sqrt(et_fd_mle$var.coef))
cl_fd_mle$coef+c(-1,1)*qnorm(1-0.025)*as.vector(sqrt(cl_fd_mle$var.coef))

# Cointegration Analysis
par(mfrow=c(1,1))
coint.mod = lm(log(bitcoin.ts)~log(litecoin.ts))
plot(log(bitcoin.ts)~log(litecoin.ts), main="Bitcoin against Litecoin for
Cointegration Analysis", ylab="Log of Bitcoin", xlab="Log of Litecoin")
lines(abline(coint.mod$coefficients))
coint.res =  ts(resid(coint.mod), frequency=365, start=c(2013, 119))

par(mfrow=c(1,2), oma=c(0, 0, 2, 0))
plot(coint.res, main="Residuals against Time")
qqnorm(coint.res, main="QQ Plot"); qqline(coint.res)
mtext("Cointegration Linear Analysis Residuals", outer = TRUE, cex = 1.5)

# ADF test of cointegration residuals
adf.test(coint.res)

# EGCM function for Cointegration
egcm.results = egcm(log(litecoin.ts), log(bitcoin.ts))
plot(egcm.results)
```