UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
FACULDADE DE CIÊNCIAS ECONÔMICAS
DEPARTAMENTO DE ECONOMIA E RELAÇÕES INTERNACIONAIS

RODRIGO CARVALHO GARCÍA VIALE

ANALYSIS OF RETURN AND RISK OF CRYPTOCURRENCIES

Porto Alegre
2021

RODRIGO CARVALHO GARCÍA VIALE

ANALYSIS OF RETURN AND RISK OF CRYPTOCURRENCIES

Work presented in partial fulfillment of the requirements for the degree of Bachelor in Economics.

Advisor: Prof. Dr. Fernando Augusto Boeira Sabino da Silva

Porto Alegre

2021

CIP - Catalogação na Publicação

Viale, Rodrigo Carvalho García
Analysis of Return and Risk of Cryptocurrencies / Rodrigo Carvalho García Viale. -- 2021.
58 f.
Orientador: Fernando Augusto Boeira Sabino da Silva.

Trabalho de conclusão de curso (Graduação) -- Universidade Federal do Rio Grande do Sul, Faculdade de Ciências Econômicas, Curso de Ciências Econômicas, Porto Alegre, BR-RS, 2021.

1. GARCH. 2. Estimação de volatilidade. 3. Criptomoedas. I. Augusto Boeira Sabino da Silva, Fernando, orient. II. Título.

Elaborada pelo Sistema de Geração Automática de Ficha Catalográfica da UFRGS com os dados fornecidos pelo(a) autor(a).

RODRIGO CARVALHO GARCÍA VIALE

ANALYSIS OF RETURN AND RISK OF CRYPTOCURRENCIES

Work presented in partial fulfillment of the requirements for the degree of Bachelor in Economics.

Aprovada em: Porto Alegre, _____ de _____ de 2021.

BANCA EXAMINADORA:

Prof. Prof. Dr. Fernando Augusto Boeira Sabino da Silva
UFRGS

Prof. Dr. Cristiano Lima Hackmann
UFRGS

Prof. Dr. Nelson Seixas Dos Santos
UFRGS

"Risk comes from not knowing what you are doing."
Warren Buffet

(HAGSTORM Jr., p. 94-95, 1994)

ACKNOWLEDGMENTS

I first thank my parents, Any and Carlos, for all the support and opportunities they provided me with throughout this journey of studies that has started long ago and is yet to be finished.

I thank my family for the inspiration to aim big, and I know they wish me the best, as far away as they might be. I thank my friends, for their support and for all the good moments we shared together, especially those from Rosário and from Kingira, who I know I can count on for life. I thank the professors I had for the contributions they made towards my academic development.

Lastly, I thank my former co-worker and friend, Bernardo, for his assistance with the coding and for his support of my career.

RESUMO

Este estudo examina a aplicabilidade do GARCH (1, 1) com inovações assimétricas com distribuição t para séries que buscam representar o mercado de criptomoedas. Com um conjunto de dados de dois anos para dez ativos, dois índices teóricos são criados, um ponderando os ativos selecionados pelas suas capitalizações de mercado e outro por meio da Análise de Componentes Principais, e são submetidos a uma análise de risco-retorno, feita utilizando um modelo GARCH, com suas estimativas comparadas às de uma medida de volatilidade realizada. Os modelos GARCH apresentaram majoritariamente coeficientes significativos, porém evidenciaram a presença de autocorrelação conjunta nos resíduos, de acordo com os testes de Ljung-Box realizados.

Palavras-chave: Criptomoedas. Estimação de volatilidade. GARCH.

ABSTRACT

This study examines the applicability of GARCH (1, 1) with asymmetric t-distributed innovations to series that seek to represent the cryptocurrency market. With a two years data set of ten assets, two theoretical indices are created, one by weighting selected assets' by market capitalization and the other one through Principal Components Analysis, and are subjected to a risk-return analysis, made by using a GARCH model, with its estimates compared to the ones from a realized volatility measure. The GARCH models presented mostly significant coefficients, but showed, however, evidence for the presence of joint autocorrelation in the residuals, according to the Ljung-Box tests that were ran.

Keywords: Cryptocurrencies. Volatility estimation. GARCH.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

ADA	Cardano
ARCH	Autoregressive Conditional Heteroskedasticity
ALGO	Algorand
ANN	Artificial Neural Networks
ATOM	Cosmos
BTC	Bitcoin
BNB	Binance Coin
CAPM	Capital Asset Pricing Model
CME	Chicago Mercantile Exchange
DOGE	Dogecoin
ETH	Ethereum
GARCH	Generalized Autoregressive Conditional Heteroskedasticity
HAR-RV	Heterogeneous Autoregressive model of Realized Volatility
LTC	Litecoin
MATIC	Polygon
PCA	Principal Components Analysis
PCC	Principal Components Combining
XRP	XRP

CONTENTS

1 INTRODUCTION

The assertive allocation of investment portfolios can be a difficult task to carry out, as it invariably involves a plurality of risks. Market risk in particular can never be completely eliminated; however, much is discussed in terms of its control and minimization, mainly through diversification, which depends on the correct mapping of correlations between assets. In a securities selection process aimed at portfolio optimization, therefore, it becomes necessary having not only assertive estimates for the covariance matrix, but also precise volatility measures.

A very common and popularized method in the literature dealing with volatility estimation is the Generalized Autoregressive Conditional Heteroskedasticity (GARCH), introduced by Bollerslev (1986), which employs daily returns in order to get a measure for the conditional variance of an asset. GARCH-type models, however, can overpredict volatility, as documented by Nomikos and Pouliasis (2011) when dealing with petroleum futures' time series. These models, therefore, can present a slow response a volatility shock passes, taking time for the estimates to go down, which can incur in losses of forecasting performance as documented by Vortelinos (2017), in a study across multiple asset classes, further detailed in Section 2.

With the growing amount, availability, and manageability of data at higher frequencies (intraday) than the daily one, a richer sample covariance matrix can be estimated, leading ultimately to the construction of a matrix based on realized measures.

Few high-frequency studies have been performed, however, for cryptocurrencies. Since 2008, cryptocurrencies have gained notoriety, especially because of the appreciation of Bitcoin, which had an incredible peak in 2018 that was only reached and surpassed by the end of 2020 (see Figures 4.1 and 7.3 for cumulative log-returns). These alternative assets are traded worldwide in various *exchanges*, private platforms that bare similarities to traditional stock and futures exchanges, like connecting buyers and sellers, but also some peculiarities, like unlimited trading 24 hours a day every day of the year, which increases the amount of data per day, since negotiation is not limited to a specific time frame of each day. From 2018 many new digital currencies were introduced, each with a different purpose. These have also had noteworthy appreciation in general, with the total market cap of 50 most valued cryptocurrencies having reached over 1,9 trillion dollars, according to CoinMarketCap, a website that gathers and provides information on a number of cryptocurrencies. These higher returns when compared to traditional assets, however, are attained only by the increased volatility that is underlying to this market.

This study aims to verify the applicability of volatility estimators, namely the realized volatility and the GARCH, for cryptocurrencies' time series.

In Chapter 2, a literature review concerning this work is introduced, Chapter 3 presents the empirical methodology applied in this work, Chapter 4 briefly describes the data set, Chapter 5 presents the obtained results and finally, Chapter 6 concludes.

2 LITERATURE REVIEW

Measuring volatility is of great importance for building portfolios and controlling their risk. This measure, however, is not observable, but latent, and when considering daily returns, it is not capable of fully expressing the price variation that occurred during the day. Although unobservable, volatility is estimable, with several models with satisfactory predictive capacity, considering common characteristics of such series, including non-stationary mean, occurrence of volatility clusters, that is, high values tend to be followed by high values, which also ends up causing heteroscedasticity; divergent impacts between positive and negative innovations, causing asymmetry in the distribution, which deviates from normality also due to the occurrence of heavy tails. In order to deal with these characteristics of volatility, Engle (1982) introduced the Autoregressive Conditional Heteroskedasticity (ARCH) model, which allows for heteroskedasticity and captures the dependency found. In this model, the average-adjusted return (innovation) does not have serial correlation, but is dependent, and can be written as a function of previous values, so that the volatility in the present can be written as a function of innovations in the previous period. Not long after this model was proposed, Bollerslev (1986) presented a generalization of the ARCH, named Generalized Autoregressive Conditional Heteroskedasticity (GARCH), in which the present volatility is given not only by previous innovations, but also by the volatility in the previous period, a method which became popular in volatility prediction and is still used on a large scale today, through the GARCH (1,1), which combines simplicity with accuracy in predictions. This model is widely applied to several asset classes and is hardly surpassed, as Hansen (2005) demonstrated through a comparison in which 330 distinct ARCH-type models were used.

With the increasing digitization of financial markets in recent decades, it has become customary for stock exchanges to store all trades carried out (tick-by-tick) for listed assets. From the availability of this information, new possibilities for the computation of volatility arise since this becomes an increasingly observable phenomenon as the sampling frequency increases. In other words, as one moves from samples of daily returns to intraday returns, price fluctuations throughout the analyzed period are considered, making a discrete series (day by day, for example) increasingly closer to a continuous series. This gives rise to a new class of measures called realized measures, which seek to reflect the growing observability (realization) of the concept of volatility, the most popular being the Realized Volatility, to be described in Subsection 3.1. Therefore, new estimators and models are needed in order to make the best use of all the available information. Corsi (2009) proposes the Heterogeneous Autoregressive model of Realized Volatility (HAR-RV), which consists of a linear regression in which a volatility measure in the present is dependent on the same measure for the day before, its average in the last 5 days (i.e., during the week) and its average over the last 22 days (i.e., during the month). The main argument behind the windows used by the author is that market agents have different horizons, contributing differently to volatility. With this specification, agents with short trading horizons react to short- and long-term volatility, while market participants with longer-term investments do not necessarily consider abrupt fluctuations in shorter periods. As for the volatility measure used in the model, there is a wide range of estimators that employ different computations on the data in order to make the most efficient use of them, some of which deal with the microstructure noise problem present and resulting from the increase of the sampling frequency.

Studies on the application of measurements performed to time series are diverse when it comes to assets from developed markets. Liu, Patton and Sheppard (2015) compare more than 400 estimators for volatility using measurements for 31 assets, from 5 different classes and 2 different countries. They concluded that there is great difficulty in overcoming the simple Realized Variance sampled every 5 minutes, given the presence of microstructure noise when the sampling frequency is increased. Microstructure noise is a recurrent problem in studies with high frequency data and can be attributed mainly to bid-ask spreads. Due to its presence, by increasing the sampling frequency it is common for measurements taken to tend to infinity, but at the same time, more of the available information is being used. This, therefore, generates a trade-off relationship between the use of data in a more complete way, at the cost of greater noise in the series, and vice versa.

When it comes to data for the Brazilian market, however, the scarcity of studies in this field of research is noticeable. Boff (2017), through the analysis of two time series of Brazilian stocks, points out that more sophisticated measures when compared to the realized variance present better performance in predicting volatility. Furthermore, regarding the sampling frequency, the author concludes that the liquidity of the asset in question plays an important role in the decision, with more liquid stocks indicating the use of higher sampling frequencies, and less traded assets requiring more sampling spread out in more time. In a multivariate analysis for the selection of minimum variance portfolios of 30 assets traded on the Brazilian stock exchange, B3, Borges, Caldeira and Ziegelmann (2015) compare several performed measurements and intraday sampling frequencies, including an estimator that allows data sampling not synchronized. In this study, the sampling frequency of 5 minutes presents the lowest transaction costs, however, lower frequencies, such as 120 minutes and daily, present better results in terms of risk and return, respectively. The use of correlations from realized covariances (realized correlations) is not subject to the problem of overparameterization, like the GARCH-type models, in which the number of parameters increases as more assets are added, compromising the estimation necessary to carry out a multivariate analysis. Caldeira, Moura, Perlin and Santos (2017) employ different sampling frequencies and an algorithm for sampling unsynchronized data, along with estimators for the covariance matrix using intraday data, in addition to obtaining more dynamic versions of such correlations through multivariate GARCH structures. The sample consists of tick-by-tick data for the 30 most traded shares in the period analyzed (2009-2012) on B3. The authors conclude that portfolios built from realized covariance matrix generate better results in terms of risk and turnover (less financial volume traded in order to rebalance the portfolio) when compared to those built from covariance matrices based on low frequency data, in this case, daily. Portfolios built from conditional realized covariances – those that include a GARCH structure – also show better results in terms of risk when compared to their unconditional counterparts, but this comes at the cost of higher turnover, which can generate higher transaction costs and offset the lower risk. Furthermore, corroborating the study by De Pooter, Martens and Van Dijk (2008), the sampling frequency plays a fundamental role. The sampling frequency of 5 minutes generates the best results in terms of risk and turnover in the case of the Brazilian article, presenting robustness in a sub-sample of the same stocks with a lower level of liquidity.

When comparing the volatility predictions of the HAR-RV, the GARCH (1, 1), the Artificial Neural Networks (ANN), and the Principal Components Combining (PCC) method for 7 different asset classes, Vortelinos (2017) reveals that none of the models can surpass the HAR-RV, which is considered the best model in all the criteria used, closely followed by the model that employs the PCC. Results show the benefits that can be obtained when using high frequency data in detriment of daily data, as commonly used by financial market agents through the GARCH (1, 1).

The asset class of cryptocurrencies has been subject to few volatility measurement studies so far, although having over ten years of existence. Since 2008 these digital assets have become popular, among which the most relevant in the current scenario is the Bitcoin, which has the main purpose of providing means of payment worldwide, with fast transfers, and has its issuance limited in nature and decentralized. Given this popularization of crypto assets in recent years, as well as the rise in prices that accompanied it, and since several important players in the world financial market have already shown interest or are already investing in such assets, it is relevant to extend the analysis of high-frequency data usage to stocks, commodities, interest rates, and currencies that have been tested in developed markets to digital asset markets. The application of an analysis similar to cryptocurrencies, however, is hampered by the fact that these assets are not centrally traded, as in most cases when dealing with stocks, commodities, interest rates and currencies, which are traded in stock exchanges. Crypto assets, however, are negotiated in a decentralized manner, with their trading taking place on several different platforms, called exchanges. This characteristic of cryptocurrencies opens the possibility for different prices for time series on different exchanges, generating arbitrage opportunities. The selection of a series of prices for the assets to be analyzed must, therefore, consider the relevance of the exchange from which the data were extracted, in order to preserve the applicability and relevance of the tested methods. Market movements, however, continue to be relatively synchronous and uniform across all trading platforms known to the author. The series of these assets also show significantly greater volatility when compared to other assets, with large price movements occurring almost daily. This adds to the motivation of the present work to test the significance of price variations in cryptocurrencies: since volatility is higher, the inherent risk deserves special attention by all market agents and finding accurate methods of measuring variance becomes essential to the prevention of relevant financial losses, as well as testing rather the increased volatility is reflected on cryptocurrencies' returns. In addition, another factor that differentiates the assets in question from those traded traditional on the stock and mercantile exchanges is the trading hours, which are not limited, allowing trades to be carried out 24 hours a day, every day of the year. This affects even more the pricing of assets, as different agents may follow different trading hours, as several exchanges accept clients from other countries, creating a difference in time zones that can create imbalances in traded volumes, in addition to periods of volatility located in some time slots.

Catania and Sandholdt (2019) apply the methods already disseminated in the literature for developed markets that deal with volatility prediction through high-frequency data to the Bitcoin price series. The authors obtain tick-by-tick prices for Bitcoin from the listings of this asset on

Coinbase and Bitstamp, two exchanges that rank among the 10 largest cryptocurrency trading platforms in the world, according to CoinMarketCap, which provides several classifications both between exchanges and between crypto assets. The series covers the period from September 13, 2011, to March 18, 2018 for the Bitstamp price and, for Coinbase, from December 1, 2014 with the same end date as the other exchange. The period analyzed, therefore, includes the time when Bitcoin became popular, both in the media and in the investment world, due to the large price hike that occurred in early 2017. The study confirms the hypothesis of intraday seasonality, with notable occurrences of peaks and valleys in the average volume traded at different times of the day. Volatility also follows a similar pattern of intraday seasonality, with significant differences in the degree of price fluctuations depending on the time of day. These patterns are different for each of the trading platforms used in the study, and this difference can be explained by the location where each exchange is based, with Coinbase with headquarters in the United States, while Bitstamp is based in Europe. The authors note that seasonal patterns are present in terms of weekdays for both the average volume traded as well as for the realized volatility, with a clear weekend effect, which is more evident during what is called the “Hype” period, in which Bitcoin became extremely popular, leading to a sort of Dutch disease, defined as the period from 2017 onwards. Realized volatility also showed increasing intensity from Mondays onwards, reaching peaks on Thursdays and Fridays, a phenomenon that is notably not found in financial assets that are regularly traded on weekdays, such as stocks, commodities, interest rates and currencies. When analyzing the predictability of returns for Bitcoin, the result is negative for frequencies of 1 day or more. For sample frequencies of up to 6 hours, however, some predictability is found through a first-order autoregressive model of order one (i.e., an AR (1) model), although its statistical significance is limited. Furthermore, such predictability of returns varies over time, with periods of greater and lesser precision for the model used. When dealing with realized volatility, the study is able to find similarities to time series of regular financial assets, such as the long memory characteristic, that is, a slowly decreasing autocorrelation function, and the so-called leverage effect, which consists of asymmetry of contribution to realized volatility on the part of negative and positive returns, where lows usually present greater shocks in absolute values, having greater explanatory power on total realized volatility. In terms of predicting realized volatility, the authors conclude that estimates became more accurate after 2017, during the Hype period. These conclusions are reached from 5 specifications of HAR-RV models and their derivations, starting from the most basic, which consists of a linear regression of past performed volatility, to more robust specifications, which have the inclusion of components to identify jumps in intraday returns, and, even more, a model capable of identifying and considering the presence of the leverage effect. By evaluating the predictions from such models, the authors are able to identify significant benefits arising from the inclusion of a component for the leverage effect. Finally, the accuracy of estimates for future realized volatility is also dependent on the forecast horizon, as found in time series of regular financial assets. The article has great relevance in terms of laying the foundations for modeling volatility through the use of high-frequency data, applying methods that are already widespread when dealing with assets commonly acquired by the main investment funds. With the beginning of trading of future contracts for Bitcoin in world stock exchanges, such as the Chicago Mercantile Exchange (CME), it is expected that market agents will migrate part of their investments to crypto assets, making it particularly significant for risk management that precise volatility measures are developed in order to avoid excessive losses that may result from the high volatility of such assets. Given the availability of data from two different exchanges, investigating the possibility of arbitration between them would be a relevant extension of the study. In addition, it would also be interesting to replicate the analysis to other cryptocurrencies, since Bitcoin is just one among many, despite still being the most relevant in terms of average volumes traded. Furthermore, adding more crypto creates the possibility of carrying out a multivariate analysis, in order to explore the construction and performance of minimum variance portfolios.

3 METHODOLOGY

The methodology used in this study combines realized measures, which are presented in Section 3.1, applied to real series, i.e. the cryptocurrencies that make up the initial data set, but also to theoretical series, that are created through the assets' weighting techniques described in Section 3.2, constituting the final data set. At the same time, a GARCH volatility estimate is computed, according to what is depicted in Section 3.3. The entire analysis was conducted in the software R and the codes are accessible on GitHub (https://github.com/rcviale/tcc) and can also be found in Appendix 7.6.

3.1 REALIZED MEASURES

For the purposes of this study, individual asset log returns are computed from the sum of the 5-min log returns for each cryptocurrency i, according to equation (3.1). The sampling frequency choice follows the research from Liu et al. (2015), which concluded that it is very difficult to significantly beat a simple 5-minute realized variance (RV) when it comes to price variation estimators constructed from high-frequency data. This estimator for the time series variance consists of the sum of squared 5-minute returns for each day t and asset i, as in equation (4.2).

$$r_{i,t} = \sum_{j=1}^{m} r_{i,t-1+jn} \text{ and } R_i = \begin{bmatrix} r_{i,\,1} \\ r_{i,\,2} \\ \cdots \\ r_{i,\,T} \end{bmatrix}, \tag{3.1}$$

where t represents a day, n is a fraction of a trading session associated with the sampling frequency (since 5-min returns are being used and there are 1440 minutes in one day, $n = \frac{1}{(1440/5)} = \frac{1}{288}$), m represents the number of observations in on day $\left(m = \frac{1}{n} = 288\right)$ and T is the last observation day.

$$RV_{i,\,t} = \sum_{j=1}^{m} r_{i,\,t-1+jn}^{2}, \tag{3.2}$$

where n is a fraction of a trading session associated with the sampling frequency (since 5-min RV is being used and there are 1440 minutes in one day, $n = \frac{1}{(1440/5)} = \frac{1}{288}$) and m represents the number of observations in on day $\left(m \ = \frac{1}{n} = 288\right)$.

The volatility $RVOL_{i,\,t}$ provided by this estimator is then simply the square root of each $RV_{i,\,t}$.

$$RVOL_{i,\,t} = \sqrt{RV_{i,t}} \tag{3.3}$$

It is noteworthy that by using these estimators, although the analysis employs high-frequency data, this information is aggregated in daily series, enabling comparisons with models that make use of returns only in a daily scale, as the ones from the GARCH family.

The next section describes the computation of measures that seek representing the movements in the whole cryptocurrency market in a general manner.

3.2 SYNTHETIC MARKET ESTIMATES AND WEIGHTING METHODS

In order to create synthetic indices that are able to represent the overall cryptocurrency market, it becomes necessary to weight the assets in order to create this index. Two methods, therefore, are employed as weighting factors: the daily market capitalization for each asset and the weights derived from the application of the Principal Component Analysis (PCA).

3.2.1 Estimation based on market capitalization

From the daily market capitalization series, weights $w_{i,\,t}$ are estimated such that

$$w_{i,\,t}^{(MC)} = \frac{MC_{i,t}}{\sum_{j=1}^{K} M\,C_{j,t}}, \tag{3.4}$$

$$W_{MC,\,t} = \begin{bmatrix} w_{1,\,t}^{(MC)} \\ w_{2,\,t}^{(MC)} \\ \cdots \\ w_{K,\,t}^{(MC)} \end{bmatrix}, \tag{3.5}$$

where $MC_{i,\,t}$ is the market capitalization for asset i in period t, and the superscript (MC) denotes the weight derived from market capitalization.

These weights are then combined with the assets' returns in equation (3.1), computed previously to obtain an estimate of *market* returns $R_{m,\,t}^{(MC)}$ such as

$$r_{m,\,t}^{(MC)} = \sum_{i=1}^{K} (r_{i,\,t} w_{i,\,t}), \tag{3.6}$$

$$R_m^{(MC)} = \begin{bmatrix} r_{m,\,1}^{(MC)} \\ r_{m,\,2}^{(MC)} \\ \cdots \\ r_{m,\,T}^{(MC)} \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{K} (r_{i,\,1} w_{i,\,1}) \\ \sum_{i=1}^{K} (r_{i,\,2} w_{i,\,2}) \\ \cdots \\ \sum_{i=1}^{K} (r_{i,\,T} w_{i,\,T}) \end{bmatrix}. \tag{3.7}$$

Testing the market's volatility as a driver of returns is the main goal of this work, however, obtaining an estimate for the market's volatility is not as simple as weighting and summing up different series. This happens not only because the covariance between assets must be considered,[1] but also, there is no 5-minute market returns series (only daily), where the sum of squared returns could be taken, similarly as to what was done with the individual cryptocurrencies. To obtain an estimate for the market's 5-minute realized variance, therefore, firstly the 5-minute sample covariance matrix for each period must be estimated

[1] See Appendix 7.2 for the referred property of variance.

$$RCov_t = \begin{bmatrix} s^2_{11,\,t} & s^2_{12,\,t} & \cdots & s^2_{1K,\,t} \\ s^2_{21,\,t} & s^2_{22,\,t} & \cdots & s^2_{2K,\,t} \\ \cdots & \cdots & \cdots & \cdots \\ s^2_{K1,\,t} & s^2_{K2,\,t} & \cdots & s^2_{KK,\,t} \end{bmatrix}, \tag{3.8}$$

where $s^2_{ij,\,t}$ denotes the 5-minute sample covariance between assets i and j in day t. This matrix can then be simply weighted by the $W_{MC,\,t}$ vectors obtained from the assets' market capitalization in equation (3.5) in the following way.

$$RV^{(MC)}_{m,\,t} = W^T_{MC,\,t} \times RCov_t \times W_{MC,\,t}, \tag{3.9}$$

where superscript T denotes the transposed of the vector in question. The volatility $RVOL^{(MC)}_M$ estimate derived from market capitalization weighting is then obtained simply by taking square root of the obtained realized variance estimate, similarly to the procedure in equation (3.3).

$$RVOL^{(MC)}_M = \begin{bmatrix} \sqrt{RV^{(MC)}_{M,1}} \\ \sqrt{RV^{(MC)}_{M,2}} \\ \cdots \\ \sqrt{RV^{(MC)}_{M,T}} \end{bmatrix}. \tag{4.10}$$

3.2.2 Estimation based on principal component analysis

An alternative method used to obtain an estimate for the market's returns and volatility is through Principal Component Analysis (PCA), introduced by Pearson (1901). In a context of closely related variables such as this data set (see the series' correlation matrix in Appendix 7.3, Brooks (2019) noted that PCA can be particularly useful, because it can transform the K series into a new set of K uncorrelated variables, by taking linear combinations of the original data set. In other terms, the PCA can be written as

$$\begin{array}{ccccccccc} p_{1,t} & = & \alpha_{11,t}r_{1,t} & + & \alpha_{12,t}r_{2,t} & + & \dots & + & \alpha_{1K,t}r_{K,t} \\ p_{2,t} & = & \alpha_{21,t}r_{1,t} & + & \alpha_{22,t}r_{2,t} & + & \dots & + & \alpha_{2K,t}r_{K,t} \\ \dots & = & \dots & + & \dots & + & \dots & + & \dots \\ p_{K,t} & = & \alpha_{K1,t}r_{1,t} & + & \alpha_{K2,t}r_{2,t} & + & \dots & + & \alpha_{KK,t}r_{K,t} \end{array}, \tag{3.11}$$

where $p_{i,t}$ and $r_{i,t}$, represent, respectively, the i principal component and original variable (daily log-returns) in period t, and $\alpha_{ij,t}$ denotes the coefficient for variable j in the principal component i in day t. The sum of the squared coefficients α^2 that were estimated must be equal to one for each individual component, as in equation (3.12). James et al. (2014) noted that scaling of the variables is not necessary when they are measured in the same units, and since this is applicable to this study, as all variables are measured in the unit of log-returns, no scaling was performed.

$$\sum_{j=1}^{K} \alpha_{i,\,j}^2 = 1 \; \forall \; i = 1,\ 2,\ \dots,\ K. \tag{3.12}$$

This procedure is, therefore, carried out for the daily log-returns series for each day t in the analyzed period, and from the estimated principal components, the first one, which is able to explain the largest amount of the data set's variance, approximately 57.4% during the sampled period, (see Appendix 7.4 for additional information on PCA's application), is taken as the returns for an index to represent the cryptocurrencies market in the same fashion as the market cap weighted index constructed in Subsection 3.2.1. This first principal component being a weighted average of the analyzed assets is then set as an estimate for *market returns* derived from PCA, $R_M^{(PCA)}$, defined as

$$R_M^{(PCA)} = \begin{bmatrix} p_{1,1} \\ p_{1,2} \\ \dots \\ p_{1,T} \end{bmatrix}. \tag{3.13}$$

Computing the realized volatility of the market, however, requires a weighting vector for each day to apply the same matrix multiplication as in Subsection 3.2.1. The PCA described immediately before provides only weights for the entire sample period and not daily as the weighting set used in the previous subsection. These vectors are obtained, therefore, by applying the same method, PCA, for the 5-minute returns series. In other words, instead of applying PCA to the daily returns' series, the intraday ones are used, where there are $m = 288$ returns per day, and the method is computed for each separate day, obtaining a vector of containing asset weights for each day in the sample, as in equation (3.14).

$$W_{PCA,t} = \begin{bmatrix} w_{1,\,t}^{(PCA)} \\ w_{2,\,t}^{(PCA)} \\ \cdots \\ w_{K,\,t}^{(PCA)} \end{bmatrix}, \tag{3.14}$$

The market realized variance is then obtained in the same fashion as in the market capitalization case presented in equation (3.9), that is, by multiplying the transposed PCA weights vector by the 5-minute sample covariance matrix and then the PCA weights vector, as showed in equation (3.15). Lastly, the realized volatility $RVOL_M^{(PCA)}$ is again computed by taking the square root of the realized variance, as defined in equation (3.3), and applied in equation (3.10).

$$RV_{M,t}^{(PCA)} = W_{PCA,t}^{T} \times RCov_t \times W_{PCA,t}, \tag{3.15}$$

$$RVOL_M^{(PCA)} = \begin{bmatrix} \sqrt{RV_{M,1}^{(PCA)}} \\ \sqrt{RV_{M,2}^{(PCA)}} \\ \cdots \\ \sqrt{RV_{M,T}^{(PCA)}} \end{bmatrix}. \tag{3.16}$$

With these, there are two measures to represent the cryptocurrencies market derived from PCA and two derived from market capitalization of individual assets, with one in each case being for market returns and the other one for realized volatility.

In the next Section, a GARCH model is presented as an alternative volatility estimator.

3.3 GARCH MODEL

According to empirical studies from Hansen and Lunde (2005), where the authors compared the forecast potential of a total of 330 ARCH-type models, there is “no evidence that a GARCH(1, 1) is outperformed by more sophisticated models” in their analysis of exchange rates, by evaluating out-of-sample predictions, although this model is found inferior to the ones that can accommodate a leverage effect when analyzing IBM stock returns.

The GARCH(p, q) process models conditional variance σ_t^2, making it possible for the variance to be dependent on its own past lags. The letters *p* and *q* determine the order of the model, with *q* representing the number of past conditional variance lags directly influencing itself in period *t* and *p*, the number of past “innovations” that directly influence the conditional variance in time *t*, where an innovation is defined as the mean-corrected return, or in the case of this study, the mean-corrected log return. As it is not the objective nor the focus of this work, GARCH of higher orders will not be presented here, only the specific case of interest, the GARCH(1, 1), but a detailed explanation can be found directly in Bollerslev (1986), or in Tsay (2005).

The GARCH(1, 1), therefore, relies on past information to determine the present conditional volatility. Defining the innovations for asset *i* in period *t* as $a_{i,t} = r_{i,t} - \mu_{i,t}$, where $\mu_{i,t}$ is the mean of the returns, the model can be formalized as

$$a_{i,t} = \sigma_{i,t}\epsilon_{i,t} \tag{3.17}$$

$$\sigma_{i,t}^2 = \alpha_{0,i} + \alpha_{1,i}a_{i,t-1}^2 + \beta_{1,i}\sigma_{i,t-1}^2, \tag{3.18}$$

where $\epsilon_{i,t}$ is assumed, in this study, to be an i.i.d. random variable following a Student’s *t*-distribution, $\alpha_{0,i} > 0$, $\alpha_{1,i} \geq 0$ and $\beta_{1,i} \geq 0$. Finally, it is necessary that $\alpha_{0,i} + \beta_{1,i} < 1$, to ensure the unconditional variance of $a_{t,i}$ is finite while the conditional variance, $\sigma_{i,t}^2$, is allowed to change from period to period.

It can be seen that in this model a spike in $a_{i,t-1}^2$ or in $\sigma_{i,t-1}^2$ will impact $\sigma_{i,t}^2$ positively, as long as $\beta_{1,i}$ has a positive sign, or, in other words, large innovations or conditional variance in time *t - 1* tend to be followed by large innovations in time *t*, since an increase in $\sigma_{i,t}^2$ leads to a higher

$a_{i,t}$, making it possible for volatility clusters to occur as they do in financial markets, more noticeably in crisis periods, such as the global financial crisis (2007-2008) and the more recent 2020 stock market crash caused by the COVID-19 pandemic.

For its reasonable forecasting power, allied with its simplicity and parsimony two GARCH(1, 1) models, with *t*-distributed innovations to account for heavier tails, are estimated for the market returns estimates derived from the market capitalization and from the PCA described in Subsections 3.2.1 and 3.2.2, respectively. In other words, two cases are computed, one where the market capitalization weighted theoretical index is modeled, providing a volatility estimate σ_{MC} for $R_M^{(MC)}$, and in the second one, the first principal component series is subjected to a GARCH(1, 1) fitting, yielding volatility series σ_{PCA} for $R_M^{(PCA)}$.

4 DATA DESCRIPTION

The data used in this work consists of ten minute-by-minute cryptocurrency prices series obtained from Binance’s API and negotiated in this exchange, which is currently the largest one in the world in terms of daily trading volume of cryptocurrencies according to CoinMarketCap. It is also one of the world leaders in terms of number of “markets,” where a “market” is defined as the pair of currencies that are being traded.

These cryptocurrencies were the top ten ranked by market capitalization in CoinMarketCap that are classified as “coins” (as opposed to “tokens”), had at least a 2 year span of observations and are not stable coins, which are coins backed by a reserve asset. A short description about each selected asset is provided below.

a) **Bitcoin (BTC):** a peer-to-peer version of electronic cash that allows for online payments to be sent directly from one party to another without going through a financial institution (Nakamoto, 2008);
b) **Ethereum (ETH):** “What Ethereum intends to provide is a blockchain with a built-in fully fledged Turing-complete programming language that can be used to create “contracts" that can be used to encode arbitrary state transition functions (…)" (Buterin, 2013);
c) **Binance Coin (BNB):** Frankenfield (2021) pointed that Binance Coin was created as a provider of discount on Binance’s trading fees in 2017, it evolved and can now be used as payment of transaction fees on this exchange, travel bookings, entertainment, online services, and financial services;
d) **Litecoin (LTC):** “Litecoin is a peer-to-peer Internet currency that enables instant, near-zero cost payments to anyone in the world. Litecoin is an open source, global payment network that is fully decentralized without any central authorities.” (Lee, 2011);
e) **Cardano (ADA):** the Cardano Docs pointed that “Cardano is a decentralized third-generation proof-of-stake blockchain platform and home to the ada cryptocurrency. It is the first blockchain platform to evolve out of a scientific philosophy and a research-first driven approach”;

f) **XRP (XRP):** XRP is the native cryptocurrency of Ripple, a real-time gross settlement system which provides instant transactions that also accepts other currencies, according to CoinMarketCap;
g) **Cosmos (ATOM):** according to Cosmos' website, this cryptocurrency "(…) is a decentralized network of independent parallel blockchains" or, in other words, it "(…) is an ecosystem of blockchains that can scale and interoperate with each other." Blockchain is a technology used by many other cryptocurrencies and is the digital equivalent of a ledger;
h) **Polygon (MATIC):** CoinMarketCap states that it "is the first well-structured, easy-to-use platform for Ethereum scaling and infrastructure development. Its core component is Polygon SDK, a modular, flexible framework that supports building multiple types of applications."
i) **Algorand (ALGO):** this cryptocurrency is a self-sustaining, decentralized, blockchain-based network, and it was created with the purpose of providing shorter transaction time and lower fees than other blockchains, while avoiding mining;
j) **Dogecoin (DOGE):** CoinMarketCap states that this cryptocurrency "(…) is based on the popular "doge" Internet meme and features a Shiba Inu on its logo." It is an open-source digital currency that was forked (when a blockchain is separated in two different paths forward) from Litecoin in December 2013. According to CoinMarketCap, "Dogecoin's creators envisaged it as a fun, light-hearted cryptocurrency that would have greater appeal beyond the core Bitcoin audience, since it was based on a dog meme.";

This study's sample was restricted to the period between August 1, 2019, to July 31, 2021 ($T = 731$ days in total), because not all series were available from the source before these dates. Since cryptocurrencies trade 24 hours a day, every day of the year, all time references in this work are standardized to reflect the Coordinated Universal Time central time zone (UTC+0). The analyzed cryptocurrencies are listed in Table 4.1, along with each series' number of missing and available observations during this sample period. There is a total of 1,051,640 time stamps (minutes) during this time window, with 0.25% of it being of missing data. Additional information on the availability of the individual series, with each series' initial dates on Binance's API can be found in Appendix 7.6.

Table 4.1 - Series' Sample Sizes

Name	Acronym	NAs	N	% NAs
Bitcoin	BTC	2598	1050042	0.25
Ethereum	ETH	2599	1050041	0.25
Binance Coin	BNB	2599	1050041	0.25
Litecoin	LTC	2598	1050042	0.25
Cardano	ADA	2599	1050041	0.25
XRP	XRP	2598	1050042	0.25
Cosmos	ATOM	2599	1050041	0.25
Polygon	MATIC	2599	1050041	0.25
Algorand	ALGO	2599	1050041	0.25
Dogecoin	DOGE	2599	1050041	0.25

Source: Elaborated by the author.

From this data set, the last price from each 5-minute interval was taken, constructing ten series with this frequency, which then amount to 210,528 and $m = 288$ observations in total and per day, respectively, for each asset. This periodicity was chosen based on the empirical work from Liu, Patton, and Sheppard (2015), where the authors found it is very difficult to significantly beat 5-minute realized variance when it comes to price variation estimators constructed from high-frequency data, considering a span of five asset classes to reach this conclusion. Subsequently, to give proper treatment to the missing observations, Kalman's filter was used, which is able to preserve the correlation structure of the assets and was introduced by Kalman (1960).

These 5-minute closing prices series were then used to compute realized variances, $RV_{i,\,t}$, and realized volatilities, $RVOL_{i,\,t}$ for each asset, according to the formulas that presented in Section 3.1. Daily log-returns, $r_{i,\,t}$, consisting of the sum of all 5-minute log-returns in a day, were also taken. Summary statistics for these measures for each series can be found in Appendix 7.1. The assets' cumulative daily log-returns in the analyzed period are presented in Figure 4.1.

Figure 4.1 – Cryptocurrencies' Cumulative Log-Returns from August 1, 2019 to July 31, 2021

Source: Elaborated by the author.

In order to weight the assets and create a theoretical index to represent the "crypto market", the daily market capitalization of each cryptocurrency was obtained from CoinMarketCap, with the weighting procedures being described in Section 3.2.

5 RESULTS

This section presents the results from the estimated volatility measures, comparing the results from the Realized Volatility estimator and the GARCH(1, 1) model. The properties of the GARCH are verified in Subsection 5.1, while the comparison between the two estimators is made on Subsection 5.2. The Realized Volatility estimator is not further analyzed due to it being a non-parametric estimator, which makes it not liable of any verification on the assumptions area.

5.1 GARCH MODEL

The estimates for the first GARCH(1, 1) model, with an asymmetric *t*-distribution for the index constructed from the assets' market capitalization can be found in Table 5.1. It can be seen that the $\mu_{i,t}$ parameter, giving the mean for the series is highly significant, given its p-value is below the 1% level of significance. The intercept for the conditional variance, $\alpha_{0,i}$, however, is not significant at this level, but is at the conventional α = 0.05. The coefficient for the squared innovations, $\alpha_{1,i}$, is also significant at the 1% significance level, and its interpretation implicates that a unit increase in the squared innovation in *t - 1* causes the conditional variance in *t* to be increased by 0.08082. The last coefficient, $\beta_{1,i}$, is also significant at the 1% level, meaning a unit increase in the conditional variance in period *t - 1* would cause a 0.89517 increase in the conditional variance in period *t*. Finally, the shape parameter defines the degrees of freedom of the Student's *t* distribution, with the heaviness of its tails.

Table 5.1 – Market Capitalization GARCH Volatility Estimates

Coefficient	Estimate	Std. Error.	t statistic	p-value
$\mu_{i,t}$	0.00272	0.00102	2.66	0.01
$\alpha_{0,i}$	0.00006	0.00003	1.96	0.05
$\alpha_{1,i}$	0.08082	0.03102	2.60	0.01
$\beta_{1,i}$	0.89517	0.03068	29.18	0.00
Shape	3.24677	0.43559	7.45	0.00

Source: Elaborated by the author.

It is noteworthy, also, as demonstrated in Table 5.2, that when using robust standard errors all the estimates are also significant at the 5% level, with the exception of the $\alpha_{0,i}$ parameter. These standard errors are robust against violations of the distributional assumptions, by employing quasi maximum likelihood estimation (QMLE), introduced by White (1982).

Table 5.2 – Market Capitalization GARCH Volatility Estimates with Robust Standard Errors

Coefficient	Estimate	Std. Error.	t statistic	p-value
$\mu_{i,t}$	0.00272	0.00106	2.55	0.01
$\alpha_{0,i}$	0.00006	0.00003	1.82	0.07
$\alpha_{1,i}$	0.08082	0.03392	2.38	0.02
$\beta_{1,i}$	0.89517	0.02717	32.94	0.00
Shape	3.24677	0.46297	7.01	0.00

Source: Elaborated by the author.

Analyzing the standardized residuals for this model, in Table 5.3, the Ljung-Box test, introduced by Ljung and Box (1978), is presented for lags 1 to 20, significant evidence of joint autocorrelation is found, as all lags reject the null hypothesis of no joint autocorrelation at the 1% significance level, which compromises the inference. In Table 5.4, the same test is performed on the squared residuals for the same lags, where, no evidence of joint autocorrelation is indicated at the 5% level, except for the first lag.

Table 5.3 – Ljung-Box Test for the Standardized Residuals of the Market Capitalization GARCH

(continues)

Lag	Statistic	p-value
1	11.82	0.00
2	17.08	0.00
3	19.68	0.00
4	32.40	0.00
5	34.96	0.00
6	36.77	0.00
7	37.55	0.00
8	38.91	0.00
9	38.91	0.00

(conclusion)

Lag	Statistic	p-value
10	42.34	0.00
11	42.58	0.00
12	44.32	0.00
13	45.21	0.00
14	46.82	0.00
15	49.90	0.00
16	49.90	0.00
17	52.25	0.00
18	55.57	0.00
19	55.70	0.00
20	55.79	0.00

Source: Elaborated by the author.

Table 5.4 – Ljung-Box Test for the Standardized Squared Residuals of the Market Capitalization GARCH

Lag	Statistic	p-value
1	4.99	0.03
2	5.17	0.08
3	5.17	0.16
4	8.21	0.08
5	8.48	0.13
6	8.48	0.20
7	11.59	0.11
8	11.61	0.17
9	11.61	0.24
10	11.65	0.31
11	12.23	0.35
12	12.23	0.43
13	12.38	0.50
14	12.38	0.58
15	12.69	0.63
16	12.70	0.69
17	12.73	0.75
18	12.94	0.80
19	62.08	0.00
20	62.17	0.00

Source: Elaborated by the author.

The estimates for the second GARCH(1, 1) model, with aymmetric t-distribution, which derives from the cryptocurrency index constructed from the application of PCA, are presented in Table 5.5. In this model, $\mu_{i,t}$, the mean coefficient, is insignificant, as is the intercept for the conditional volatility, $\alpha_{0,i}$, even at the 10% significance level. The other coefficients, $\alpha_{1,i}$ and $\beta_{1,i}$, however, are highly significant, as is the number of degrees of freedom of the Student's t-distribution, even at the 1% significance level. The $\alpha_{1,i}$ value indicates a unit increase in the squared innovation in period t - 1, $a_{i,t-1}^2$, should increase the volatility in period t by 0.05125. As for the $\beta_{1,i}$ coefficient, a unit increase in the conditional variance in t - 1, $\sigma_{i,t-1}^2$, should cause an increase of approximately 0.90043 in the conditional variance in period t.

Table 5.5 – PCA GARCH Volatility Estimates

Coefficient	Estimate	Std. Error.	t statistic	p-value
$\mu_{i,t}$	0.00000	0.00406	0.00	1.00
$\alpha_{0,i}$	0.00003	0.00005	0.60	0.54
$\alpha_{1,i}$	0.05125	0.00440	11.65	0.00
$\beta_{1,i}$	0.90043	0.01249	72.11	0.00
Shape	4.02307	0.18921	21.26	0.00

Source: Elaborated by the author.

Using the same robust standard errors method as in the previous GARCH estimate, the results presented on Table 5.6 are obtained, with no difference in the significance of the parameters criteria.

Table 5.6 – PCA GARCH Volatility Estimates with Robust Standard Errors

Coefficient	Estimate	Std. Error.	t statistic	p-value
$\mu_{i,t}$	0.00000	0.00576	0.00	1.00
$\alpha_{0,i}$	0.00003	0.00007	0.38	0.70
$\alpha_{1,i}$	0.05125	0.00792	6.47	0.00
$\beta_{1,i}$	0.90043	0.01556	57.85	0.00
Shape	4.02307	0.14460	27.82	0.00

Source: Elaborated by the author.

Performing the same tests as for the previous model, for joint autocorrelation of the residuals, the results are presented in Table 5.7 and Table 5.8, for the standardized residuals and for the standardized squared residuals, respectively. Similar results are found, with significant joint autocorrelation in all lags for the standardized residuals, while for the squared values, only the first lag presents this characteristic at the 5% significance level.

Table 5.7 – Ljung-Box Test for the Standardized Residuals of the PCA GARCH

Lag	Statistic	p-value
1	9.02	0.00
2	16.96	0.00
3	18.48	0.00
4	31.09	0.00
5	39.91	0.00
6	40.99	0.00
7	41.11	0.00
8	42.03	0.00
9	42.26	0.00
10	45.40	0.00
11	45.41	0.00
12	45.83	0.00
13	48.59	0.00
14	54.15	0.00
15	59.91	0.00
16	59.92	0.00
17	60.15	0.00
18	61.66	0.00
19	62.84	0.00
20	62.93	0.00

Source: Elaborated by the author.

Table 5.8 – Ljung-Box Test for the Standardized Squared Residuals of the PCA GARCH

(continues)

Lag	Statistic	p-value
1	4.41	0.04
2	5.27	0.07
3	5.28	0.15
4	12.09	0.02
5	14.63	0.01
6	14.67	0.02

(conclusion)

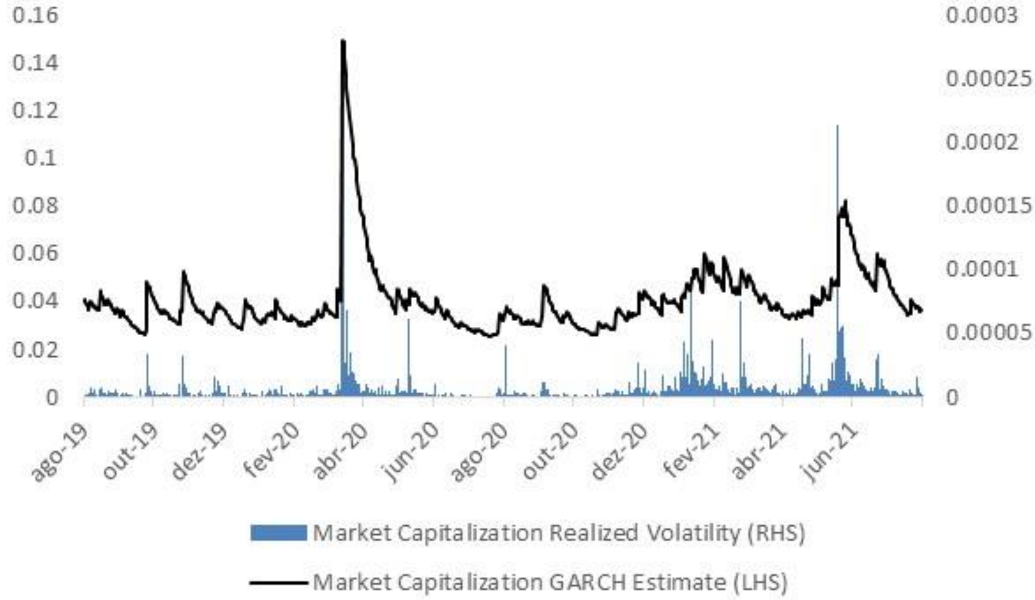
Lag	Statistic	p-value
7	17.77	0.01
8	17.77	0.02
9	17.79	0.04
10	18.06	0.05
11	18.13	0.08
12	18.13	0.11
13	18.28	0.15
14	18.50	0.19
15	18.92	0.22
16	19.02	0.27
17	19.03	0.33
18	19.03	0.39
19	19.03	0.45
20	19.22	0.51

Source: Elaborated by the author.

5.2 COMPARISON

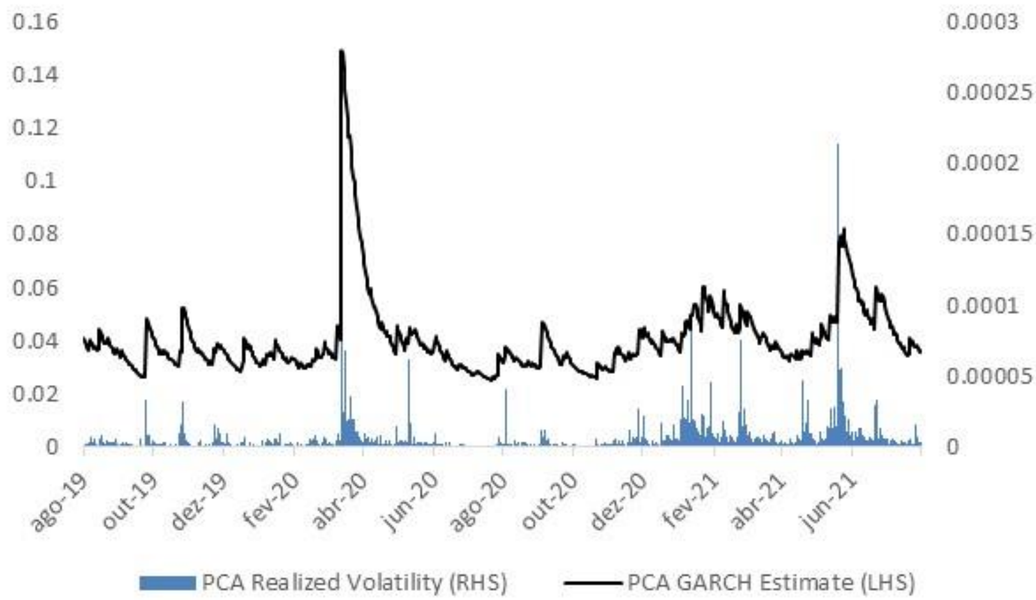
In Figures 5.1 and 5.2, a comparison between both volatility estimators can be found. The GARCH process is able to follow the realized volatility, demonstrating spikes in the series as the volatility increases. Another characteristic it is able to reproduce is the clustering of volatility, that is, when a spike occurs, it tends to be followed by another high value, which can be seen in the figures. This, however, also causes the volatility estimate to take longer to get lower, meaning that when using it as a forecasting tool it can overestimate the volatility for a few periods after a shock occurs, a known characteristic in this model. Still, as noted by Hansen and Lunde (2005), it is no ordinary task to beat a GARCH(1, 1) model, as it allies simplicity, parsimony and precision, while other more sophisticated models can be more complex while not improving forecasts in a satisfactory amount.

Figure 5.1 – Market Capitalization Realized Volatility and GARCH Conditional Volatility Estimates



Source: Elaborated by the author.

Figure 5.2 – PCA Realized Volatility and GARCH Conditional Volatility Estimates



Source: Elaborated by the author

Considering the results presented in this section, the GARCH(1, 1) with an asymmetric *t*-distribution is verified to be have significant coefficients when modeling the constructed indices of cryptocurrencies. It presented, however, autocorrelation in the residuals, which compromises any inference that might be done from such models. Still, it is a simple model, with known properties, widely explored in the literature.

6 CONCLUSION

This study's central purpose was to investigate the risk and return of two cryptocurrencies' indices through a GARCH(1, 1) fit, with an asymmetric *t*-distribution. With the rising popularity of this asset class and considering its seemingly natural higher variation when compared to traditional assets, such as stocks, commodities, and foreign exchange, it becomes necessary to estimate if this asset class is compatible with this model, as it is widely used in the literature. Furthermore, assertive volatility estimation and measurement is a relevant for the risk management of market agents such as asset management firms and banks, possibly preventing excessive exposures and losses. Volatility is a latent variable, not an observable one, therefore, estimators are necessary to obtain a measure of it. By correctly estimating volatility, investors might change their business decisions based on these computations, avoiding unrewarded variation as well as developing trading strategies based on this measure. For the rational risk averse investor, additional volatility should only be accepted when the higher returns are possible to compensate for it.

The investigation in this work was made by analyzing 10 different cryptocurrency 5-minute returns series during a 2-year period, obtaining estimates for the market returns and market volatility of this asset class derived from market capitalization weighting and PCA. Two GARCH(1, 1) with *t*-distributed innovations models were computed, and its estimates were analyzed and compared with the 5-minute realized volatility, checking for its capability of responding to volatility shocks and clusters in the cryptocurrency market. The models appeared to be well behaved, and decently described the volatility movements, but showed significant evidence of joint autocorrelation in the residuals.

This study aims to serve as an introductory application to the cryptocurrency market of traditional methods widely disseminated and used in the literature when analyzing asset pricing and the risk-return relation of regular assets. Further studies can better develop the ideas that were presented here in diverse research paths, such as:

1) expanding the data set to include more cryptocurrencies, and altering the period of time analyzed;

2) performing out-of-sample forecasts and evaluating these instead of the significance of models' coefficients;

3) using a robust covariance matrix to correct for autocorrelation and heteroskedasticity in the data set, such as the one introduced by Newey and West (1987), instead of sample covariance;

4) testing more sophisticated models from the GARCH framework, such as the EGARCH, presented by Nelson (1991), which is able to account for asymmetry often observed in financial markets, where negative shocks tend to be more intense than positive ones, presenting higher values for the returns' modules;

5) testing other volatility models, such as the HAR-RV, from Corsi (2009).

7 BIBLIOGRAPHY

BINANCE. Binance APIs. **Binance API endpoint for data queries.** 2021. Available in: https://api.binance.com/. Accessed on: August 1, 2021.

BOFF, Tainan de Bacco Freitas. **Comparação do Desempenho de Medidas Realizadas para a Previsão de Volatilidade de Ações da B3**. Porto Alegre, 2018. 48 p. Trabalho de Conclusão de Curso de Graduação (Bacharelado em Ciências Econômicas) – Faculdade de Ciências Econômicas, Universidade Federal do Rio Grande do Sul.

BOLLERSLEV, Tim. Generalized Autoregressive Conditional Heteroskedasticity. **Journal of Econometrics**, Amsterdã, v. 31, issue 3, p. 307–327, 1986. Available in: https://EconPapers.repec.org/RePEc:eee:econom:v:31:y:1986:i:3:p:307-327. Accessed on: November 3, 2021.

BORGES, Bruna K.; CALDEIRA, João F.; ZIEGELMANN, Flavio A. A. Selection of minimum variance portfolio using intraday data: An empirical comparison among different realized measures for BM&FBovespa data. **Brazilian Review of Econometrics**, Rio de Janeiro, v. 35, n. 1, p. 23-46, 2015.

BROOKS, Chris. 2019. **Introductory Econometrics for Finance**. Cambridge University Press, 4th ed., 2019. Available in: https://doi.org/10.1017/9781108524872. Accessed on: November 3, 2021.

HAGSTORM Jr., Robert G. **The Warren Buffett Way:** Investment Strategies of the World's Greatest Investor. John Wiley & Sons: New York, p.94-95, 1994

BUTERIN, Vitalik. **A Next Generation Smart Contract & Descentralized Application Platform**. [*s.l.*], 2013. Available in: https://translatewhitepaper.com/wp-content/uploads/2021/04/EthereumOrijinal-ETH-English.pdf. Accessed on: October 25, 2021.

CALDEIRA, João F.; MOURA, Guilherme V.; PERLIN, Marcelo S., SANTOS, André A. P. Portfolio management using realized covariances: evidence from Brazil. **Economia**, São Paulo, v. 18, n. 3, p. 328-343, 2017.

CARDANO DOCS. Cardano Docs. **Cardano's white paper**. Available in: https://docs.cardano.org/introduction. Accessed on: October 25, 2021.

CATANIA, Leopoldo; Sandholdt, Mads. Bitcoin at a High Frequency. **Journal of Risk and Financial Management**, Basiléia, v. 12, n. 36, 2019.

COINMARKETCAP. CoinMarketCap. Cryptocurrency data. [*s.l.*], 2021. Available in: https://coinmarketcap.com/. Accessed on: October 25, 2021.

CORSI, Fulvio. A simple approximate long-memory model of realized volatility. **Journal of Financial Econometrics**, Cary, v. 7, n. 2, p. 194-196, 2009.

COSMOS. Cosmos Network. **Cosmos' white paper**. [*s.l.*], [20--]. Available in: https://v1.cosmos.network/intro. Accessed on: October 25, 2021.

DE POOTER, Michiel; MARTENS, Martin; VAN DIJK, Dick. Predicting the daily covariance matrix for S&P 100 stocks using intraday data—but which frequency to use? **Econometric Reviews**, Nova Iorque, v. 27, n. 1-3, p. 199-229, 2008.

ENGLE, Robert F. Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation. **Econometrica**, Hoboken, v. 50, n. 4, p. 987-1007, 1982.

FRANKENFIELD, Jake. Binance Coin (BNB) Definition. **Investopedia**, [*s.l.*], 2021. Available on: https://www.investopedia.com/terms/b/binance-coin-bnb.asp. Accessed on: October 25, 2021.

GITHUB. **GitHub**, 2021. Author's GitHub repository with the used R codes. Available in: https://github.com/rcviale/tcc. Accessed on: November 16, 2021.

HANSEN, Peter R.; LUNDE, Asger. A forecast comparison of volatility models: does anything beat a GARCH(1, 1). **Journal of Applied Econometrics**, Hoboken, v. 20, n. 7, p. 873-889, 2005.

JAMES, Gareth et.al. **An Introduction to Statistical Learning:** With Applications in R. New York: Springer, 2014.

KALMAN, Rudolph E. A new Approach to Linear Filtering and Prediction Problems. **Journal of Basic Engineering**, Nova Iorque, v. 82, n. 1, p. 35-45, 1960.

LEE, Charlie. **Lite Coin White Paper**. [*s.l.*], 2011. Available in: https://whitepaper.io/document/683/litecoin-whitepaper. Accessed on October 25, 2021.

LIU, Lily Y.; PATTON, Andrew J.; SHEPPARD, Kevin. Does anything beat 5-minute RV? A comparison of realized measures across multiple asset classes. **Journal of Econometrics**, Amsterdã, v. 187, n. 1, p. 293-311, 2015.

LJUNG, Greta; BOX, George E. P. On a Measure of Lack of Fit in Time Series Models. **Biometrika**, v. 65, issue 2, p. 297-303, 1978. Available in: https://doi.org/10.1093/biomet/65.2.297. Accessed on: November 5, 2021.

NAKAMOTO, Satoshi. **Bitcoin:** A Peer-to-Peer Electronic Cash System. [*s.l.*], 2009. Available in: https://bitcoin.org/bitcoin.pdf. Accessed on October 25, 2021.

NELSON, Daniel B. "Conditional Heteroskedasticity in Asset Returns: A New Approach." **Econometrica**, [*s.l.*], v. 59, issue 2, p. 347–370, 1991. Available in: https://EconPapers.repec.org/RePEc:ecm:emetrp:v:59:y:1991:i:2:p:347-70. Accessed on: November 5, 2021.

NEWEY, Whitney K.; WEST, Kenneth D. A Simple, Positive Semi-definite, Heteroskedasticity and Autocorrelation Consistent Covariance Matrix. **Econometrica**, [*s.l.*], v. 55, issue 3, p. 703–

708, 1987. Available in: https://ideas.repec.org/a/ecm/emetrp/v55y1987i3p703-08.html. Accessed on: November 5, 2021.

NOMIKOS, Nikos K.; POULIASIS, Panos. 2011. Forecasting Petroleum Futures Markets Volatility: The Role of Regimes and Market Conditions. **Energy Economics**, [*s.l.*], v. 33, issue 2, p. 321–237, 2011. Available in: https://EconPapers.repec.org/RePEc:eee:eneeco:v:33:y:2011:i:2:p:321-337. Accessed on: November 5, 2021.

PEARSON, K. On Lines and Planes of Closest Fit to Systems of Points in Space. **Philosophical Magazine**, [*s.l.*], issue 2, p. 559–72, 1901.

TSAY, Ruey S. **Analysis of Financial Time Series**. 2. ed. Wiley Series in Probability and Statistics. Hoboken, NJ: Wiley-Interscience, 2005. Available in: http://gso.gbv.de/DB=2.1/CMD?ACT=SRCHA&SRT=YOP&IKT=1016&TRM=ppn+483463442&sourceid=fbw_bibsonomy. Accessed on: November 5, 2021.

VORTELINOS, Dimitrios I. Forecasting realized volatility: HAR against Principal Components Combining, neural networks and GARCH. **Research in International Business and Finance**, Amsterdã, v. 39, p. 824-839, 2017.

WHITE, Halbert. Maximum Likelihood Estimation of Misspecified Models. Econometrica, Hoboken, v. 50, n. 1, p.1-25, 1982.

8 APPENDIX

8.1 SERIES' SUMMARY STATISTICS

Table 7.1: $r_{i,t}$ summary statistics

Asset	Min.	Median	Mean	Max.	Std. Dev.	25%	75%
BTC	-0.5212	0.0020	0.0019	0.1734	0.0419	-0.0158	0.0191
ETH	-0.5809	0.0030	0.0034	0.2367	0.0544	-0.0197	0.0306
BNB	-0.5947	0.0018	0.0034	0.5256	0.0633	-0.0205	0.0298
LTC	-0.4722	0.0006	0.0005	0.2422	0.0568	-0.0248	0.0272
ADA	-0.5444	0.0045	0.0042	0.2726	0.0620	-0.0262	0.0311
XRP	-0.5564	0.0010	0.0012	0.4348	0.0684	-0.0208	0.0225
ATOM	-0.6220	0.0003	0.0017	0.2820	0.0707	-0.0329	0.0367
MATIC	-0.7494	0.0023	0.0062	0.4556	0.0914	-0.0356	0.0411
ALGO	-0.6976	0.0000	0.0005	0.3606	0.0735	-0.0381	0.0377
DOGE	-0.4924	-0.0001	0.0059	1.4953	0.0976	-0.0198	0.0159

Source: Elaborated by the author.

Table 7.2: $RV_{i,t}$ *summary statistics*

Asset	Min.	Median	Mean	Max.	Std. Dev.	25%	75%
BTC	0.0001	0.0010	0.0020	0.0870	0.0000	0.0049	0.0005
ETH	0.0002	0.0015	0.0031	0.1034	0.0000	0.0069	0.0009
BNB	0.0002	0.0017	0.0038	0.1427	0.0001	0.0091	0.0009
LTC	0.0002	0.0020	0.0040	0.1449	0.0001	0.0083	0.0011
ADA	0.0003	0.0023	0.0048	0.1828	0.0001	0.0099	0.0013
XRP	0.0001	0.0016	0.0055	0.1792	0.0002	0.0136	0.0008
ATOM	0.0005	0.0038	0.0063	0.2953	0.0002	0.0139	0.0022
MATIC	0.0005	0.0046	0.0102	0.4483	0.0006	0.0243	0.0025
ALGO	0.0005	0.0047	0.0073	0.2033	0.0001	0.0107	0.0027
DOGE	0.0002	0.0021	0.0097	0.7651	0.0015	0.0385	0.0012

Source: Elaborated by the author.

Table 7.3: $RVOL_{i,t}$ summary statistics

(continues)

Asset	Min.	Median	Mean	Max.	Std. Dev.	25%	75%
BTC	0.0077	0.0316	0.0374	0.2949	0.0007	0.0255	0.0225
ETH	0.0129	0.0392	0.0469	0.3215	0.0009	0.0301	0.0295

(conclusion)

Asset	Min.	Median	Mean	Max.	Std. Dev.	25%	75%
BNB	0.0150	0.0411	0.0506	0.3778	0.0012	0.0352	0.0303
LTC	0.0136	0.0443	0.0539	0.3806	0.0011	0.0337	0.0336
ADA	0.0186	0.0480	0.0587	0.4275	0.0013	0.0367	0.0367
XRP	0.0096	0.0394	0.0562	0.4234	0.0024	0.0486	0.0279
ATOM	0.0220	0.0613	0.0697	0.5434	0.0015	0.0384	0.0468
MATIC	0.0224	0.0680	0.0834	0.6695	0.0032	0.0569	0.0500
ALGO	0.0228	0.0683	0.0764	0.4509	0.0014	0.0380	0.0522
DOGE	0.0142	0.0453	0.0684	0.8747	0.0051	0.0711	0.0352

Source: Elaborated by the author.

8.2 SUM AND MULTIPLICATION OF VARIANCE PROPERTIES

To obtain the variance of a variable Z, that is constructed as a weighted average of two random correlated variables X and Y, the covariance must be considered. This can be achieved by computing:

$$Var(aX + bY) = a^2\, Var(X) + b^2\, Var(Y) + 2ab\, Cov(X,Y), \tag{7.1}$$

where a and b are constants. This case is extendable for a linear combination of K random variables $\{X_1, X_2, \dots, X_K\}$, where the variance can be written as:

$$\begin{aligned} Var(\sum_{i=1}^{K} a_i\, X_i) &= \sum_{i,j=1}^{K} a_i\, a_j Cov(X_i, X_j) \\ &= \sum_{i=1}^{K} a_i^2\, Var(X_i) + \sum_{i \neq j} a_i\, a_j Cov(X_i, X_j) \\ &= \sum_{i=1}^{K} a_i^2\, Var(x_i) + 2 \sum_{1 \leq i < j \leq K} a_i\, a_j Cov(X_i, X_j) \end{aligned} \tag{7.2}$$

8.3 SERIES' CORRELATION MATRIX

Table 7.4: Full sample 5-minute correlation matrix

	BTC	ETH	BNB	LTC	ADA	XRP	ATOM	MATIC	ALGO	DOGE
BTC	1.00	0.83	0.70	0.77	0.67	0.61	0.59	0.52	0.52	0.41
ETH	0.83	1.00	0.71	0.79	0.70	0.64	0.62	0.52	0.54	0.39
BNB	0.70	0.71	1.00	0.67	0.64	0.57	0.59	0.51	0.52	0.36
LTC	0.77	0.79	0.67	1.00	0.68	0.65	0.60	0.49	0.52	0.37
ADA	0.67	0.70	0.64	0.68	1.00	0.59	0.59	0.49	0.52	0.36
XRP	0.61	0.64	0.57	0.65	0.59	1.00	0.51	0.43	0.45	0.33
ATOM	0.59	0.62	0.59	0.60	0.59	0.51	1.00	0.46	0.51	0.33
MATIC	0.52	0.52	0.51	0.49	0.49	0.43	0.46	1.00	0.41	0.28
ALGO	0.52	0.54	0.52	0.52	0.52	0.45	0.51	0.41	1.00	0.30
DOGE	0.41	0.39	0.36	0.37	0.36	0.33	0.33	0.28	0.30	1.00

Source: Elaborated by the author.

8.4 SUMMARY STATISTICS FOR ASSETS' WEIGHTS COMPUTED THROUGH PCA

Table 7.5: Summary Statistics for Assets' Weights Computed Through PCA

Acronym	T	Min.	Median	Mean	Max.	Var.	Std. Dev.	25%	75%
BTC	731	0	0.0360	0.0416	0.2830	0.0012	0.0341	0.0149	0.0605
ETH	731	0	0.0652	0.0670	0.7511	0.0027	0.0524	0.0319	0.0931
BNB	731	0	0.0558	0.0633	0.8850	0.0034	0.0586	0.0304	0.0870
LTC	731	0	0.0749	0.0830	0.4446	0.0045	0.0669	0.0427	0.1047
ADA	731	0	0.0835	0.0919	0.4933	0.0049	0.0703	0.0519	0.1213
XRP	731	0	0.0553	0.0833	0.9844	0.0130	0.1140	0.0297	0.0890
ATOM	731	0	0.1045	0.1297	0.9912	0.0169	0.1300	0.0622	0.1617
MATIC	731	0	0.1191	0.1881	0.9992	0.0501	0.2239	0.0624	0.2019
ALGO	731	0	0.1039	0.1684	0.9982	0.0391	0.1978	0.0603	0.1852
DOGE	731	0	0.0333	0.0837	0.9988	0.0327	0.1810	0.0128	0.0639

Source: Elaborated by the author.

8.5 SERIES' DATA AVAILABILITY INFORMATION

Table 7.6: Series' Initial Dates and Total Available Observations

Name	Acronym	Initial Date	NAs	N	% NAs
Bitcoin	BTC	2017-09-01	8122	2051078	0.40
Ethereum	ETH	2017-09-01	8123	2051077	0.40
Binance Coin	BNB	2018-05-01	7703	1920457	0.40
Litecoin	LTC	2017-12-01	7598	1875922	0.41
Cardano	ADA	2018-06-01	5412	1705308	0.32
Ripple	XRP	2019-08-01	5411	1660669	0.33
Cosmos	ATOM	2018-01-01	3260	1183300	0.28
Polygon	MATIC	2019-07-01	3260	1181860	0.28
Algorand	ALGO	2019-05-01	2599	1096121	0.24
Dogecoin	DOGE	2019-05-01	2599	1050041	0.25

Source: Elaborated by the author.

Figure 8.3: Cryptocurrencies' cumulative log-returns from September 1, 2017, to July 31, 2021

Source: Elaborated by the author.

8.6 R CODES

```
#### Import raw data matrix and unify all series #####
library(tidyverse)

ini_crypto <- readxl::read_excel(paste0('new_initial_dates.xlsx'))

# Function to unify all the series' closing prices in one tibble sorted by
time
source('R/unify.R')
all_data <- unify(files = as.matrix(ini_crypto[, 2]), directory = 'Data/', col
= 'open_time')
colnames(all_data) <- c("open_time", as.matrix(ini_crypto[, 2]))
all_data <- all_data %>% arrange(open_time) #FIXME
all_data %>% slice_tail(n = 10)

# Save RDS
readr::write_rds(all_data, file = 'Data/new_all_data.rds')

#### Raw data analysis and summary statistics ####
all_data <- readr::read_rds("Data/new_all_data.rds")

# Summary Timestamps Statistics
stats <- list(
  obs  = ~length(.x),
  min  = ~min(.x, na.rm = T),
  med  = ~median(.x, na.rm = T),
  mean = ~round(mean(.x, na.rm = T), digits = 2),
  max  = ~max(.x, na.rm = T),
  var  = ~round(var(.x, na.rm = T), digits = 2),
  sd   = ~round(sd(.x, na.rm = T), digits = 2),
  q1   = ~quantile(t(.x), na.rm = T)[2],
  q3   = ~quantile(t(.x), na.rm = T)[4]
)

difs <- all_data$open_time %>%
  diff() - 1 #%>% e
difs <- difs %>%
  as_tibble() %>%
  filter(value != 0) %>%
  summarise(across(everything(), stats, .names = "{.fn}")) %>%
  t()

hole_summ_table <- tibble(Metric = c("Ocurrences", "Minimum", "Median",
"Mean", "Maximum", "Variance", "Standard Deviation", "1st Quantile", "3rd
Quantile"),
                          Value  = as.vector(difs))

readr::write_rds(hole_summ_table, "Print/hole_summ_table.rds")

rm(stats, difs, hole_summ_table)

# Complete implicitly missing observations
cdata <- all_data %>%
  tidyr::separate(col = open_time, into = c('date', 'time'), sep = ' ') %>%
  tidyr::separate(col = date, into = c("year", "month", "day"), sep = "-") %>%
  tidyr::separate(col = time, into = c("hour", "minute", "second"), sep = ":")
%>%
  tidyr::complete(year, month, day, hour, minute, second) %>%
  mutate(datetime = lubridate::make_datetime(year = as.numeric(year),
                                             month = as.numeric(month),
```

```
                                              day = as.numeric(day),
                                              hour = as.numeric(hour),
                                              min = as.numeric(minute))) %>%
  select(-year, -month, -day, -hour, -minute, -second) %>%
  select(datetime, everything()) %>%
  arrange(datetime) %>%
  dplyr::filter(datetime >= as.Date('2017-09-01'), datetime < as.Date('2021-
08-01')) %>%
  arrange(datetime, BTC) %>%
  distinct(datetime, .keep_all = TRUE)

rm(all_data)

# Save RDS
readr::write_rds(cdata, file = "Data/cdata.rds")

# Fill data and save
cdata %>% fill(2:11, .direction = "down") %>%
  readr::write_rds("Data/fdata.rds")

# Take last observation of filled data before working sample starts
readr::read_rds("Data/fdata.rds") %>%
  filter(datetime >= as.Date("2019-07-31"), datetime < as.Date("2019-08-01"))
%>%
  slice_tail(n = 5) -> last

# Restrict data, fill and save
readr::read_rds("Data/cdata.rds") %>%
  filter(datetime >= as.Date("2019-08-01"), datetime < as.Date("2021-08-01"))
%>%
  fill(2:11, .direction = "down") %>%
  rbind(last) %>%
  arrange(datetime) %>%
  readr::write_rds("Data/rest_fdata.rds")

rm(list = ls())



# Load complete data
cdata <- readr::read_rds("Data/cdata.rds")
ini_crypto <- readxl::read_excel(paste0('new_initial_dates.xlsx'))

# Summary table of original sample
full_summ_table <- ini_crypto %>%
  mutate(nas      = as.vector(t(cdata %>% summarise_if(is.numeric,
~sum(is.na(.x)))) -
                                (t(cdata %>% summarise_if(is.numeric,
~which(is.na(.x) == FALSE)[1])) - 1)),
         nobs     = cdata %>% summarise_if(is.numeric, ~sum(is.na(.x) ==
FALSE)) %>% t() %>% as.vector(),
         perc_nas = round(nas / nobs * 100, digits = 2),
         Name     = c("Bitcoin", "Ethereum", "Binance Coin", "Litecoin",
"Cardano",
                      "Ripple", "Cosmos", "Polygon", "Algorand", "Dogecoin"),
         `start date` = lubridate::as_date(`start date`)) %>%
  rename(Acronym = coin, N = nobs, NAs = nas, `% NAs` = perc_nas,
         `Initial Date` = `start date`) %>%
  select(-market) %>%
  select(Name, everything())

readr::write_rds(full_summ_table, file = "Print/full_summ_table.rds")

rm(full_summ_table)
```

```
# Summary table of restricted completed sample
cdata <- cdata %>%
  filter(datetime >= as.Date("2019-08-01"), datetime < as.Date("2021-08-01"))

rest_summ_table <- ini_crypto %>%
  mutate(nas      = as.vector(t(cdata %>% summarise_if(is.numeric,
~sum(is.na(.x)))) -
                                (t(cdata %>% summarise_if(is.numeric,
~which(is.na(.x) == FALSE)[1])) - 1)),
         nobs     = cdata %>% summarise_if(is.numeric, ~sum(is.na(.x) ==
FALSE)) %>% t() %>% as.vector(),
         perc_nas = round(nas / nobs * 100, digits = 2),
         Name     = c("Bitcoin", "Ethereum", "Binance Coin", "Litecoin",
"Cardano",
                      "Ripple", "Cosmos", "Polygon", "Algorand", "Dogecoin"))
%>%
  rename(Acronym = coin, `Initial Date` = `start date`,
         N = nobs, NAs = nas, `% NAs` = perc_nas) %>%
  select(-c(market, `Initial Date`)) %>%
  select(Name, everything())

readr::write_rds(rest_summ_table, file = "Print/rest_summ_table.rds")

rm(list = ls())



#### Computation of Returns and Realized Volatility Matrices ####
# Functions to collapse series in chosen frequency, take log returns, take RV,
and collapse in day
source('R/collapse_time.R')
source('R/lrets.R')
source('R/rv.R')
source('R/collapse_date.R')

# Load and restrict data set
all_data <- readr::read_rds("Data/cdata.rds") %>%
  slice_tail(n = 731 * 1440)

# Convert series to different time frequency
all_data %>%
  collapse_time(datetime, 5, tail, 1) %>% # 5min, tail = closing
  slice_head(n = nrow(.) - 1) %>%
  readr::write_rds(file = "Data/cdata5.rds")

rm(collapse_time)


cdata <- readr::read_rds("Data/cdata5.rds")

# 5min series, replace NAs by Kalman filter values
cdata %>%
  select(-datetime) %>%
  as.matrix() %>%
  imputeTS::na_kalman() %>%
  as_tibble() %>%
  mutate(datetime = cdata$datetime) %>%
  select(datetime, everything()) %>%
  readr::write_rds("Data/kdata5.rds")



# Load specific data set
all_data <- readr::read_rds("Data/kdata5.rds")
```

```
# Cumulative log returns
cum_rets <- all_data %>%
  lrets() %>%
  slice_tail(n = nrow(.) - 1) %>%
  collapse_date(datetime, "day", sum, na.rm = TRUE) %>%
  modify_if(is.numeric, .f = ~cumsum(.x) * 100)

cum_rets[cum_rets == 0] <- NA

# Full sample cumulative log returns
cum_rets %>%
  reshape2::melt(id = "datetime") %>%
  rename(Asset = variable) %>%
  ggplot(aes(x = datetime, y = value, colour = Asset, group = Asset)) +
  geom_line() +
  labs(x = "Days",
       y = "Cumulative Log-Return") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5))

# Actual sample cumulative log returns
cum_rets %>%
  filter(datetime >= as.Date("2019-08-01")) %>%
  reshape2::melt(id = "datetime") %>%
  rename(Asset = variable) %>%
  ggplot(aes(x = datetime, y = value, colour = Asset, group = Asset)) +
  geom_line() +
  labs(x = "Days",
       y = "Cumulative Log-Return") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5))

# Tibble with daily returns
rets <- all_data %>%
  lrets() %>%
  slice_tail(n = nrow(.) - 1) %>%
  collapse_date(datetime, "day", sum, na.rm = TRUE) %>%
  rename(date = datetime)

rets[rets == 0] <- NA

# Save returns RDS
readr::write_rds(rets, file = 'Data/rets.rds')

# Read RDS
rets <- readr::read_rds(file = 'Data/rets.rds')

stats <- list(
  min  = ~min(.x, na.rm = T),
  med  = ~median(.x, na.rm = T),
  mean = ~mean(.x, na.rm = T),
  max  = ~max(.x, na.rm = T),
  var  = ~var(.x, na.rm = T),
  sd   = ~sd(.x, na.rm = T),
  q1   = ~quantile(t(.x), na.rm = T)[2],
  q3   = ~quantile(t(.x), na.rm = T)[4]
)

# Summary stats for returns
rets %>%
  select(-date) %>%
  summarise(across(everything(), stats, .names = "{.fn}_{.col}")) %>%
  pivot_longer(cols = everything(), values_to = "value") %>%
  separate(col = name, into = c("m", "cur"), sep = "_") %>%
  pivot_wider(names_from = m, values_from = value) %>%
```

```
  select(cur, min, med, mean, max, var, sd, q1, q3) %>%
  readr::write_rds(file = "Print/tablea13.rds")

rm(rets, lrets)



# Tibble with Realized Variances
rvs <- all_data %>%
  rv() %>%
  slice_tail(n = nrow(.) - 1) %>%
  collapse_date(datetime, 'day', sum, na.rm = TRUE) %>%
  rename(date = datetime)

rvs[rvs == 0] <- NA

# Save RVs RDS
readr::write_rds(rvs, file = 'Data/rvs.rds')

# Read RDS
rvs <- readr::read_rds(file = 'Data/rvs.rds')

rvs %>%
  select(-date) %>%
  summarise(across(everything(), stats, .names = "{.fn}_{.col}")) %>%
  pivot_longer(cols = everything(), values_to = "value") %>%
  separate(col = name, into = c("m", "cur"), sep = "_") %>%
  pivot_wider(names_from = m, values_from = value) %>%
  select(cur, min, med, mean, max, var, sd, q1, q3) %>%
  readr::write_rds(file = "Print/tablea11.rds")

# Tibble with Realized Volatilities
rvols <- rvs %>%
  modify_if(is.numeric, ~sqrt(.x))

# Save Rvols RDS
readr::write_rds(rvols, file = 'Data/rvols.rds')

# Read RDS
rvols <- readr::read_rds(file = 'Data/rvols.rds')

rvols %>%
  select(-date) %>%
  summarise(across(everything(), stats, .names = "{.fn}_{.col}")) %>%
  pivot_longer(cols = everything(), values_to = "value") %>%
  separate(col = name, into = c("m", "cur"), sep = "_") %>%
  pivot_wider(names_from = m, values_from = value) %>%
  select(cur, min, med, mean, max, var, sd, q1, q3) %>%
  readr::write_rds(file = "Print/tablea12.rds")

rm(list = ls())



#### Covariance Matrix and PCA Market Estimates #####
# Load RDS with specific time frequency
all_data <- readr::read_rds("Data/fdata5.rds")
source('R/lrets.R')

# Take log returns, compute PCA weights and 1st component series
covs_pca <- all_data %>%
  lrets() %>% # Take log rets
  slice_tail(n = nrow(.) - 1) %>% # Take out first row
  mutate(date = lubridate::as_date(datetime)) %>% # Only day column
  select(-datetime) %>%
```

```
  select(date, everything()) %>%
  nest(data = -date) %>% # Nest according to day
  mutate(covs    = map(.x = data, .f = ~ cov(.x, use =
"pairwise.complete.obs")),
         layout  = map_dbl(.x = covs, .f = ~ sqrt(nrow(.x)^2 -
sum(is.na(.x)))),
         pca     = map2(.x = data, .y = layout, .f = ~princomp(na.omit(.x[, 1
: .y]))),
         mkt_ret = map_dbl(.x = pca, .f = ~ mean(.x$scores[1:288])), # First
day of full sample is wrong (should be 287)
         weights = map2(.x = pca, .y = layout,
                        .f = ~ tibble(asset = colnames(all_data)[2 : (.y +
1)],
                                      weight = .x$loadings[1 : .y]^2))) %>%
  select(-data) %>%
  unnest(weights) %>%
  pivot_wider(names_from = asset, values_from = weight) %>%
  nest(data = -c(date, covs, mkt_ret, layout, pca)) %>%
  mutate(act_cov  = map2(.x = covs, .y = layout, .f = ~ as.matrix(.x[1:.y,
1:.y])),
         act_wts  = map2(.x = data, .y = layout, .f = ~ as.matrix(.x[1:.y])),
         mkt_rv   = map2_dbl(.x = act_cov, .y = act_wts, .f = ~ .y %*% .x %*%
t(.y)),
         mkt_rvol = sqrt(mkt_rv)) %>%
  select(-c(layout, data, act_cov))

# Select covariance matrices only and save RDS
covs_pca %>%
  select(date, covs) %>%
  readr::write_rds(file = "Data/covs.rds")

# Select the PCA market computations and save RDS
covs_pca %>%
  select(-c(covs, pca)) %>%
  readr::write_rds(file = "Data/pca_mkt.rds")

# Select the PCA weights and save RDS
all_data %>%
  lrets() %>% # Take log rets
  slice_tail(n = nrow(.) - 1) %>% # Take out first row
  mutate(date = lubridate::as_date(datetime)) %>% # Only day column
  select(-datetime) %>%
  select(date, everything()) %>%
  nest(data = -date) %>% # Nest according to day
  mutate(covs    = map(.x = data, .f = ~ cov(.x, use =
"pairwise.complete.obs")),
         layout  = map_dbl(.x = covs, .f = ~ sqrt(nrow(.x)^2 -
sum(is.na(.x)))),
         pca     = map2(.x = data, .y = layout, .f = ~princomp(na.omit(.x[, 1
: .y]))),
         weights = map2(.x = pca, .y = layout,
                        .f = ~ tibble(asset = colnames(all_data)[2 : (.y +
1)],
                                      weight = .x$loadings[1 : .y]^2))) %>%
  select(-data) %>%
  unnest(weights) %>%
  pivot_wider(names_from = asset, values_from = weight) %>%
  select(-c(covs, layout, pca)) %>%
  readr::write_rds(file = "Data/pca_wts.rds")

# Select PCA market Rvol
# covs_pca %>%
#   select(date, mkt_rvol) %>%
#   readr::write_rds("Data/pca_rvol.rds")
```

```
# pca_rvol <- readr::read_rds("Data/pca_rvol.rds")
rets <- readr::read_rds("Data/rets.rds")

# Daily PCA
dpca <- rets %>%
  na.omit() %>%
  select(-date) %>%
  princomp()

summary(dpca)

tibble(date = rets$date,
       mkt_ret = c(rep(NA, 699), as.matrix(dpca$scores)[, 1]),
       mkt_rvol = pca_rvol$mkt_rvol) %>%
  readr::write_rds("Data/dpca.rds")

# Difference in scalings
w1 <- dpca$loadings[1:10]^2
w2 <- rets %>%
  na.omit() %>%
  select(-date) %>%
  prcomp(scale. = TRUE)
w2 <- w2$rotation[1:10]^2

cbind(w1, w2) %>%
  as_tibble %>%
  mutate(dd = w1 - w2) %>%
  select(dd) %>%
  t.test

# Summary statistics for PC1
# covs_pca %>%
#   select(pca) %>%
#   slice_tail(n = 731) %>%
#   mutate(pc1 = map_dbl(.x = pca, .f =
~factoextra::get_eig(.x)$variance.percent[1])) %>%
#   summarise(across(pc1, stats))

# Full period covariance matrix
full_cor <- all_data %>%
  slice_tail(n = 210528) %>%
  lrets() %>%
  select(-datetime) %>%
  cor(use = "pairwise.complete.obs")

readr::write_rds(full_cor, "Print/full_cor.rds")

rm(list = ls())

#### Create Market Cap based Market Measures ####
# Load returns, realized volatilities and market cap based weights
rets <- readr::read_rds("Data/rets.rds")
rvols <- readr::read_rds("Data/rvols.rds")
weights <- readxl::read_excel(paste0('Data/weights.xlsx'))
covs <- readr::read_rds("Data/covs.rds")
pca_wts <- readr::read_rds("Data/pca_wts.rds")

# Pivot all longer
rets_long <- rets %>%
  pivot_longer(-date, values_to = "ret")
weights_long <- weights %>%
  pivot_longer(-date, values_to = "weights")
pcawts_long <- pca_wts %>%
  pivot_longer(-date, values_to = "weights")
```

```
rm(rets)

# Load mkt_est function
source("R/mkt_est.R")

### Market Cap based Market Estimates ###
# Market Cap based RVol
mkt_rvol <- covs %>%
  left_join(weights, by = "date") %>%
  mutate(layout = map_dbl(.x = covs, .f = ~ sqrt(nrow(.x)^2 -
sum(is.na(.x))))) %>%
  nest(weights = -c(date, covs, layout)) %>%
  mutate(act_covs = map2(.x = covs, .y = layout, .f = ~ .x[1:.y, 1:.y]),
         act_wts  = map2(.x = weights, .y = layout, .f = ~ .x[1:.y]),
         mkt_rvol = map2_dbl(.x = act_covs, .y = act_wts,
                             .f = ~ as.matrix(.y) %*% .x %*% t(.y))) %>%
  select(-c(layout, covs, weights, act_covs, act_wts))

# Save Market Cap based RVol RDS
readr::write_rds(mkt_rvol, "Data/mkt_rvol.rds")

# Create Market Cap based Market Returns
mkt_ret <- mkt_est(rets_long, weights_long, ret) %>%
  rename(mkt_ret = mkt_est)

# Save Market Cap based Returns RDS
readr::write_rds(mkt_ret, "Data/mkt_ret.rds")

# PCA RVol
covs %>%
  left_join(pca_wts, by = "date") %>%
  mutate(layout = map_dbl(.x = covs, .f = ~ sqrt(nrow(.x)^2 -
sum(is.na(.x))))) %>%
  nest(weights = -c(date, covs, layout)) %>%
  mutate(act_covs = map2(.x = covs, .y = layout, .f = ~ .x[1:.y, 1:.y]),
         act_wts  = map2(.x = weights, .y = layout, .f = ~ .x[1:.y]),
         mkt_rvol = map2_dbl(.x = act_covs, .y = act_wts,
                             .f = ~ as.matrix(.y) %*% .x %*% t(.y))) %>%
  select(-c(layout, covs, weights, act_covs, act_wts)) %>%
  readr::write_rds("Data/pca_rvol.rds")

rm(list = ls())

#### Create daily data tibbles ####
# Load Assets' Returns and Market estimates
mkt_ret <- readr::read_rds("Data/mkt_ret.rds")
mkt_rvol <- readr::read_rds("Data/mkt_rvol.rds")
pca_rvol <- readr::read_rds("Data/pca_rvol.rds")
dpca <- readr::read_rds("Data/dpca.rds")

# GARCH fit for Market estimates
gspec <- rugarch::ugarchspec(distribution.model = "sstd", mean.model =
list(armaOrder = c(0, 0)),
                             variance.model = list(model = "sGARCH",
garchOrder = c(1, 1)))
cap_garch <- rugarch::ugarchfit(gspec, mkt_ret$mkt_ret)
# pca_garch <- rugarch::ugarchfit(gspec, pca_mkt$mkt_ret, solver =
"hybrid")@fit$sigma
pca_garch <- rugarch::ugarchfit(gspec, na.omit(dpca$mkt_ret))

rm(gspec)

cap_garch

pca_garch
```

```
rm(list = ls())
```