Replicating the Methodology of "Common Risk Factors in Cryptocurrency": A Python Approach

by Marc Gehring under the supervision of Professor Adrien d'Avernas

1. Introduction

The cryptocurrency market has experienced significant growth in recent years, attracting both academic and practical attention. In response, numerous studies have been conducted to investigate the factors driving the market and associated risks. Among these studies, "Common Risk Factors in Cryptocurrency" by Yukun Liu, Aleh Tsyvinski, and Xi Wu (2022) stands out as a comprehensive and influential work in the field.[1] In their paper, the authors find that three factors – cryptocurrency market, size, and momentum – are the primary drivers of cross-sectional expected cryptocurrency returns. They use a comprehensive list of price- and market-related return predictors, which have previously lead to important results for equities, and find that ten cryptocurrency characteristics can be used to form successful long-short investment strategies. The authors then show that these investment strategies are explained by the cryptocurrency three-factor model. The authors also investigated the underlying mechanisms of the cryptocurrency size and momentum effects.

This thesis aims to closely replicate the methodology used in "Common Risk Factors in Cryptocurrency" by implementing it in python code with the goal of verifying the robustness of the findings and making the methodology more accessible and replicable for future researchers.[2] Specifically, the thesis provides a step-by-step guide for replicating the results and highlights the process of conducting research in this field. Throughout the thesis, the original methodology is thoroughly examined and any limitation and areas for improvement are identified.

[1] We henceforth refer to this paper as the “original paper”.
[2] We confine our analysis to chapters II and III, excluding the principal component analysis, of the original paper.

The following section provides a comprehensive overview of the methodology, detailing the data sources, variables, and statistical method employed, and compares the findings of the original paper with the new results. Subsequently, the thesis concludes with a discussion of the implications of the result and a recommendation for future research.

2. Methodology and comparison

In this section, we outline our efforts to replicate the methodology from the original paper using Python. We detail the steps taken to produce each table and figure, then compare and contrast our results to those in the original paper.
In this thesis, we utilize a Jupyter Notebook as the platform for our main analysis, taking advantage of its user-friendly interface and seamless integration of code, text, and visualizations. The structure of the Notebook closely follows the sections of the original paper, starting with Data Retrieval and Preprocessing. In this section, the researcher can filter coins based on criteria such as market capitalization and modify any additional assumptions made. The next step involves the identification of long-short investment strategies that generate positive, statistically significant returns. Finally, a small number of explanatory factors for these strategies are investigated in the last section. The goal of each section is to match the relevant tables from the original paper. Since direct LaTeX tabular format rendering is not possible in a Jupyter Notebook, we created a PDF file to display the tables below each code block. It's important to note that this file is overwritten each time a new table is rendered and must be copied if the researcher intends to save it.

The study begins by setting the start date to January 1st, 2014, as prior to this time, Bitcoin was the only actively traded cryptocurrency. The end date is dynamically set to the present date. The researcher is expected to specify the storage directory for the data. The code checks for the presence of relevant data files, and skips the data generation section if these files already exist. The retrieval of the data involves obtaining the main dataset, for which the authors of the original paper used CoinMarketCap (coinmarketcap.com). CoinMarketCap might be considered the most popular crypto data platform, but it does not provide access to historical data as part of its free plan. In this

study, we propose the utilization of CoinGecko (coingecko.com) as an alternative data source. CoinGecko is the second largest crypto data platform and offers a free API for data gathering. However, it should be noted that the API has limitations, such as only allowing 50 calls per minute. The code for retrieving the data from CoinGecko is available in the "coingecko_data.py" file located in the "data_retrieval" folder. This folder also contains additional files that allow for the retrieval of data from other sources. We have decided to include additional data sources to enable researchers to explore additional explanatory factors. Given the API traffic limitation, it is advisable to download the data sets in smaller chunks to avoid having to restart the entire process in case of interruption. The code provides functionality for this purpose. Following the methodology described in the original paper, we transform the daily frequency of the data set to a weekly frequency by taking the last available data point each week, with a week being defined as starting on Monday and ending on Sunday. To account for potential artificial or erroneous returns, all returns greater than 10 (1,000%) were set to missing. This threshold can be considered generous, and other researchers may adjust it to their preference. The data points where the market capitalization was below $1 million were filtered out, in line with the exclusion principle proposed in the original paper. However, all values for Bitcoin, Ripple, and Ethereum are still retained since these cryptocurrencies were considered the most important ones in the early days of public cryptocurrency trading. Moreover, we impute all missing values by the mean of the respective coin's return series, but only considering data after the first non-missing value. We opted for this method for its simplicity and since it can effectively fill in missing data points. Mean imputation can also help to preserve the sample size and distribution of the data. This is important for maintaining the validity of the analysis (Schafer, 1997). For the market capitalization, we impute all missing values after the first non-missing value by the average of the last and next available data values, a method knows as linear interpolation. By using the average of the surrounding values, the imputed value is likely to be representative of the underlying trends in the data, rather than introducing an outlier. However, this method assumes a linear relationship between the missing value and the surrounding values, which may not always be accurate (Little and Rubin, 2002).[3] Next, the risk-free rate is retrieved as the 1-month constant maturity market yield on U.S. Treasury Securities from

[3] Unfortunately, the authors of the original paper do not provide an explanation for how they dealt with missing values. They neither include an explanation in their internet appendix.

the Federal Reserve economic data base (fred.stlouisfed.org/series/DGS1MO).[4] The code for this can be found in the "fred_data.py" file located in the "data_retrieval" folder. For missing values that occurred on weekends, we impute the average of the values of Friday and Monday. The remaining missing data points are imputed using linear interpolation by taking the average of the last and next available data values. After preprocessing the data, we compute the weekly value-weighted coin market return series and the coin market excess return series using the risk-free rate. We reindex the dataframes to the full date range after each operation to ensure consistency in the dimensionality of the data. Finally, the summary statistics for the daily and weekly return series are presented in Table I.

We use the next block of code to visualize the return indices of the market excess return series and the return series of the major cryptocurrencies Bitcoin, Ripple, and Ethereum in Figure I. The indices are normalized to a value of 1 on the 1st of January, 2014. In the event of missing data, the last observed value is employed for imputation purposes.

In the following section of the Notebook, the computation of the five quintile excess return series as well as the long-short investment strategies for different cryptocurrency trading variables is performed. Part A focuses on computing these returns for size characteristics such as *MCAP* (log last-day market capitalization in the portfolio formation week), *PRC* (log last-day price in the portfolio formation week), *MAXDPRC* (maximum price of the portfolio formation week), and *AGE* (number of days listed since the time period began on January 1st, 2014). The quintile return series are computed by dividing all coins into quintiles based on each of the aforementioned characteristics on a weekly basis and calculating the value-weighted return of each quintile in the following week. Subsequently, the risk-free rate is subtracted from the individual quintile return series for each characteristic. The results of the size strategy returns are presented in Table II.

Additionally, we also calculate the quintile return series for the momentum characteristics, including *r 1,0* (past one-week return), *r 2,0* (past two-week return), *r 3,0* (past three-week return), *r 4,0* (past four-week return), *r 4,1* (past one-to-four-week return), *r 8,0* (past eight-week return), *r 16,0* (past 16-week return), *r 50,0* (past 50-week return), and *r 100,0* (past 100-week return). The results are shown in Table III.

[4] The authors of the original paper use the one-month Treasury bill rate.

For the volume characteristics we consider the predictors *VOL* (log average daily volume in the portfolio formation week), *PRCVOL* (log average daily volume times price in the portfolio formation week), and *VOLSCALED* (log average daily volume times price scaled by market capitalization in the portfolio formation week). The results of this analysis are depicted in Table IV.

Finally, we examine the quintile return series for the volatility characteristics. The predictors for this analysis include *BETA* (The regression coefficient β^i_{CMKT} in$R_i - R_f = \alpha^i + \beta^i_{CMKT} CMKT + \epsilon_i$ using daily returns of the previous 365 days before the formation week), *BETA2* (*BETA* squared), *IDIOVOL* (idiosyncratic volatility, measured as the standard deviation of the residual after estimating $R_i - R_f = \alpha^i + \beta^i_{CMKT} CMKT + \epsilon_i$ using daily returns of the previous 365 days before the formation week), *RETVOL* (standard deviation of daily returns in the portfolio formation week), *MAXRET* (maximum daily return of the portfolio formation week), *DELAY* (the improvement in R^2 in $R_i - R_f = \alpha^i + \beta^i_{CMKT} CMKT + \beta^i_{CMKT_{-1}} CMKT_{-1} + \beta^i_{CMKT_{-2}} CMKT_{-2} + \epsilon_i$ where $CMKT_{-1}$ and $CMKT_{-2}$ are the lagged one- and two-day coin market index excess returns, compared to using only current coin market excess returns using daily returns of the previous 365 days before the formation week), *STDPRCVOL* (log standard deviation of price volume in the portfolio formation week), and *DAMIHUD* (average absolute daily return divided by price volume in the portfolio formation week). The results of this analysis are presented in Table V.

The subsequent section of the analysis investigates whether a limited number of factors can explain the long-short investment strategies identified previously. To this end, a one-factor model is run using the cryptocurrency market excess return (*CMKT*), also referred to as the cryptocurrency Capital Asset Pricing Model (CAPM). The dependent variables in this analysis are the various long-short investment strategies minus the risk-free rate. The results of the one-factor model are presented in Table VI.

We finally set up three multi-factor models, adding the size and the momentum factors. The size factor, *CSMB*, is determined by comparing the returns of small and large portfolios. The *CMOM* is established by utilizing three-week momentum and forming portfolios by combining 2 sets of 3 portfolios. Each week, coins are divided into two groups based on their size, then further separated into three portfolios based on their past three-week performance. These portfolios consist of the bottom 30%, middle 40%,

and top 30% of coins ranked by their past returns. The momentum factor is then constructed as

$$CMOM = \frac{1}{2}(Small\ High + Big\ High) + \frac{1}{2}(Small\ Low + Big\ Low).$$

Model (1) incorporates the *CMKT* and *CSMB* factor, Model (2) includes the *CMKT* and momentum factor, and Model (3) includes all three factors, i.e. *CMKT*, *CSMB*, and *CMOM*. The results of these models are reported in Table VII, including the model statistics.

3. Conclusion

References

Little, R. J. A. and Rubin, D. B. (2002). *Statistical Analysis with Missing Data* (2nd ed.). New York, U.S.: John Wiley & Sons, Inc.

Liu, Y., Tsyvinski, A., and Wu, X. (2022). Common risk factors in cryptocurrency. *Journal of Finance*, 77, 1133-1177.

Schafer, J. L. (1997). *Analysis of Incomplete Multivariate Data* (1st ed.). New York, U.S.: Chapman and Hall/CRC.