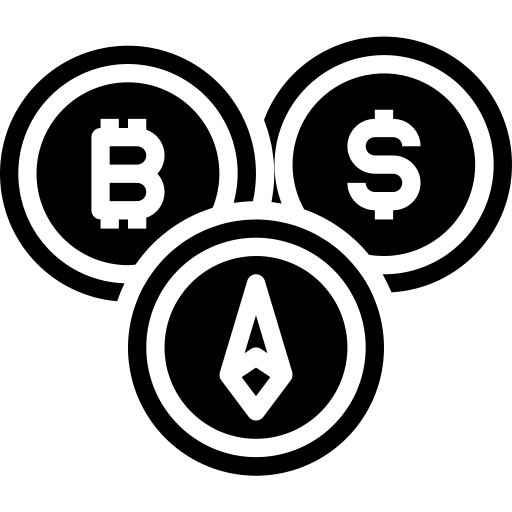
BARANZINI Guillaume 22-992-549

RENARD Emeric

**Programming project:**

**Crypto returns and fundamentals**





Evaluator : Pr. Peter Gruber

Year 2022 - 2023

SUMMARY

[I. PROGRAM DESCRIPTION 3](#_Toc123829598)

[A. Aim and definitions 3](#_Toc123829599)

[B. Mathematical theory 4](#_Toc123829600)

[II. USER GUIDE 5](#_Toc123829601)

[A. Requirements 5](#_Toc123829602)

[B. Change the main parameters 5](#_Toc123829603)

[C. Run the program 6](#_Toc123829604)

[III. RESULTS AND INTERPRETATION 7](#_Toc123829605)

[A. Regression coefficients 7](#_Toc123829606)

[B. Graphics 8](#_Toc123829607)

[BIBLIOGRAPHY 9](#_Toc123829608)

# I. PROGRAM DESCRIPTION

## A. Aim and definitions

The programming project behind the program is called ***Crypto returns and fundamentals***, which is also the name of the program file. The idea is to download historical data (market cap, price, and network size) using an API for at least five coins of layer-1 blockchains (excluding BTC) and study the relation between the market cap and the network size for individually each blockchain and for all blockchains.

To have a better understanding of the blockchain topics and the data, we recall some definitions you will find below:

* The **blockchain** is an information storage and transmission technology that keeps tracks of all the transactions made. It is a transparent and secured technology that works without a central review body.
* A **layer-1** means a blockchain that has its own network and does not depend on other ecosystems to process the transactions (verify and validate).
* A **crypto** **coin** is a cryptocurrency which operate on its own blockchain and function as digital money with the same attributes of traditional currencies. A **crypto token** is a digital unit of value that represent an asset or utility but do not have its own blockchain.
* The **market cap** for a cryptocurrency is the total value of all the coins that have been mined. It is calculated by multiplying the total number of mined coins by the price of one coin.
* The **mining** is the validation process for transactions waiting to be added to the blockchain database
* The **network size** (or active addresses) of a cryptocurrency is the number of unique addresses that were active in the network either as a sender or a receiver. Only addresses that were active in successful transactions are counted. Often, this information is expressed as Daily Active Addresses (DAA).
* An **API** or *Application Programming Interface* is a software that allows a software or a service to be “connected” to another software or service to exchange data and functionalities.

## B. Mathematical theory

**The simple linear regression model**

Globally, we want to understand if the network size (active addresses) has an influence on the blockchain’s market cap. The aim is to explain in the best way how the market cap varies according to the network size, and to another extent if we can predict the market cap from the network size.

For this purpose, we have used the simple linear regression model:

Where:

* is the variable to be explained (dependent variable), which is the **market cap**. It is expressed in a matrix form
* is the explanatory variable (independent variable), which is the **network size**. It is also expressed in a matrix form
* and are coefficients which respectively represents the intercept (intersection at the ordinate for ) and the slope of the regression.

The program give as results the estimations of and for each blockchain. From those estimations it computes a weighted average of and by weighting each and of each blockchain by the average network size for each blockchain.

**Logarithms**

We used logarithms to eliminate the effect of variable units on the coefficients. A change in units would not imply a change in the slope coefficients of the regression. In fact, we have different units in our data: market capitalization and price are expressed in USD while network size is expressed in number of active addresses per day. Using logarithms on market capitalization and network size allows us to measure them in the same "units" and therefore our model will have more stability.

# II. USER GUIDE

## A. Requirements

This program is written in *R language*. To be able to run it, you must download the “RStudio” software (from here: [https://posit.co/download/rstudio-desktop/#download](https://posit.co/download/rstudio-desktop/" \l "download)) and also “R” (from here: <https://cran.r-project.org>).

In R, you will find a lot of packages containing a set of functions that complete the basic functions. The program uses two packages (“httr” and “jsonite”) that you must install before, or the program will fail. You will find the code you have to run to install and launch the packages at the beginning of the program (lines 7 and 11).

## B. Change the main parameters

The program has been written with the idea to reduce to the minimum the necessity to change a lot of parameters. The main parameters are the cryptocurrencies that we analyze. In the program, we choose five cryptocurrencies: Ethereum (eth), Cardano (ada), Ripple (xrp), Tezos (xtz) and Tron (trx).

To analyze other cryptocurrencies, you must change one or more values in the vector x you will find at the beginning of the data extraction part (line 18):

Une image contenant texte

Description générée automatiquement

For example, if you prefer Solana instead of Ethereum, you must change “eth” by “sol” (always use the acronym of the cryptocurrency).

## C. Run the program

The program is divided in 3 big parts which are the following: create the database, the mathematical application and print the regressions’ graphs.

**Create the data base**

To be able to run the regression, we need to find all the necessary data on the cryptocurrencies that we want to study. This is the job of this part of the code.

First, you will find a loop that automatically imports data from an API (found on coinmetrics.io) and assign values for each parameters of each crypto to a variable. The data that we have from the API isn’t in the good format to work in R, so we must convert it before to groups the results. In this stage, the results are the five different dataframes containing the variables we want to study. Below, an example of what you could have (first five lines):

Une image contenant table

Description générée automatiquement

***Important: you will find those results in the environment (top right) in RStudio under the acronyms of the cryptos.***

**Mathematical application**

In this part, you will find all the results to appreciate the relation between the market cap and the network size for the blockchains you choose. It uses the “lm” function to compute all the parameters of the regression, but we only need the alpha (or ) and but also which estimates how well the fitted model can explain . If you want to have all the other parameters, they are stocked in ‘summary’.

The code also estimates an alpha and a beta for all blockchain using the “weighted.mean()” function. The results are stored in the ‘results’ dataframe. Below, an example of what you could have (first line):



***Important: you will also find the ‘summary’ and the ‘results’ dataframe in the environment (top right) in RStudio. To have all the information in ‘results’ you must run all the code in the mathematical application part.***

**Regressions’ graphs**

The code will create five graphs, one for each cryptocurrency, with the points corresponding to all the market cap values and the regression line. The graphs let you visualize if the points fit well the regression line, which is theoretically the good points when we predict the market cap with the network size.

***Important: you will find the graphs in the “Plots” window (bottom right) in RStudio. To have a better view you can click on “zoom” to enlarge them.***

# III. RESULTS AND INTERPRETATION

## A. Regression coefficients

Below, the results for the five cryptocurrencies we choose:

Une image contenant table

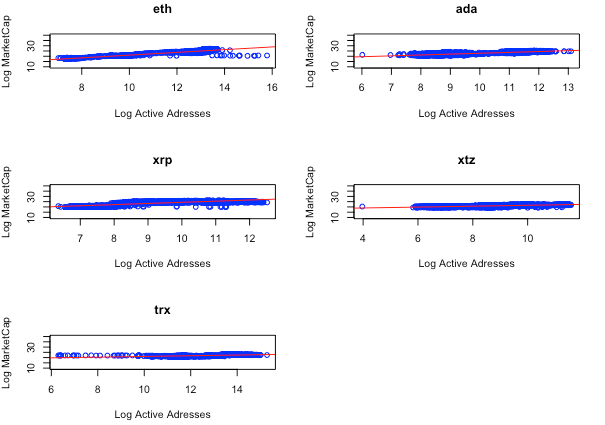
Description générée automatiquement

Alpha is the intercept with the ordinate and beta is a coefficient that multiplicate the value of x. For Ethereum, we have the following model:

For this example, when y (market cap) is moving up by one, *x* (number of daily active addresses) is moving up by 1.299. for Ethereum is equal to 0.86 which means that this model can explain more than 85% of the observed market cap variability. If we took Cardano (“ada”) this is 75% of the observed market cap variability that can be explained by this model.

We can see that for some cryptos the is lower. We think is due to the fact that there is less observations for them.

## B. Graphics



The graphs show that in trend the observations follow the regression line. The model seems to fit, the network size seems to have an impact on the market cap.

# BIBLIOGRAPHY

1. Active Addr Cnt - Data Encyclopedia. (s. d.-a). <https://docs.coinmetrics.io/asset-metrics/addresses/adractcnt>

2. Active Addr Cnt - Data Encyclopedia. (s. d.-b). <https://docs.coinmetrics.io/asset-metrics/addresses/adractcnt>

3. Blockchain. (2022, 5 décembre). CoinMarketCap Alexandria. <https://coinmarketcap.com/alexandria/glossary/blockchain>

4. CapMrktCurUSD. (s. d.). <https://docs.coinmetrics.io/info/metrics/CapMrktCurUSD>

5. Coin Metrics API v4 Reference. (s. d.). <https://docs.coinmetrics.io/api/v4>

6. Coin Metrics Python API Client. (s. d.). https://coinmetrics.github.io/api-client-python/site/index.html Interface de programmation d’application (API) | CNIL. (s. d.). Zerion Blog. <https://www.cnil.fr/fr/definition/interface-de-programmation-dapplication-api>

7. Just a moment. . . (s. d.). <https://www.datacamp.com/tutorial/linear-regression-R>

8. PriceUSD. (s. d.). <https://docs.coinmetrics.io/info/metrics/PriceUSD>

9. Statistics Globe. (2021, 2 novembre). Access & Collect Data with APIs in R (Example) | Ft. Kirby White | JSON File, Key & Create Shiny App [Vidéo]. YouTube. <https://www.youtube.com/watch?v=7XDDI0BnDAw>

10. usdc source: R/usdc-historical.R. (s. d.). https://rdrr.io/cran/usdc/src/R/usdc-historical.R