Project Description

Cuneyt Akcora

Data Description.

Our data files (shared on elearning/Ethereum datasets) contain two primary groups: token network edge files, and token price files. The **Ethereum project** is a blockchain platform, and our data comes from there. Although Ethereum started in 2015, most tokens have been created since 2016. As such, tokens have different starting dates, and their data starts from that initial date.

Token edge files have this row structure: fromNodeID\ttoNodeID\tunixTime\ttokenAmount\r\n

This row implies that fromNodeID sold tokenAmount of the token to toNodeID at time unixTime. fromNodeID and toNodeID are people who invest in the token in real life; each investor can also use multiple addresses. Two addresses can sell/buy tokens multiple times with multiple amounts. For this reason, the network is considered a weighted, directed multi(edge) graph. Each token has a maximum token count max_t; you can think of max_t as the total circulating token amount.

Clarification for supply. There are two things; first each token has a limited supply (i.e., token count, which can be found on coinmarketcap.com as circulating amount). Then each token may have sub-units. This is similar to dollar in the US. There are around 18 trillion dollars in the economy, and each dollar is divided into 100 cents (subunits). Similarly, there is a token supply, and then there is a subunit for each token. This idea comes from Bitcoin where subunits are called Satoshis, 1 Bitcoin =10^8 satoshis. Coin market cap gives the total supply, but not sub-units, which differ from token to token. Some tokens have 10^{18} sub-units. That means there can be numbers as big as $totalAmount * 10^{18}$.

Etherscan.io gives these sub-units as decimals, please see the Vechain here: It has 18 decimals, which means each Vechain token has 10^{18} subunits.
https://etherscan.io/token/0xd850942ef8811f2a866692a623011bde52a462c1
(https://etherscan.io/token/0xd850942ef8811f2a866692a623011bde52a462c1)

Price files have no extensions, but they are text based. If you open them with a text editor (use notepad++ or similar), you will see this row structure: Date\tOpen\tHigh\tLow\tClose\tVolume\tMarketCap\r

The price data is taken from https://coinmarketcap.com/ (https://coinmarketcap.com/). Open and close are the prices of the specific token at the given date. Volume and MarketCap give total bought/sold tokens and market valuation at the date.

Primary Token Selection.

In this project, each group will work with three tokens' data. These will be your primary tokens. To this end, sum the group members' UTD Ids and take modulo 20. Suppose that your id sum is 123456 whose modulo 20 gives 16. Order the tokens by edge file size on disk and choose the 16th, 17th and 18th biggest token. By this scheme, we will analyze one of the top 20 tokens.

If you have selected beautychain1 or beautychain2, please ignore them and use the next token in order. These tokens have failed recently.

Preprocessing step

Find your tokens and load their data. In each token, there may be outlier amounts which are bigger than the total amount of the token. Locate these extreme outliers, if exist, and filter them out. If there are many of these (>30), investigate how many users are included in these transactions.

Update See this news as an example of why we have these outliers: (https://cryptoslate.com/batchoverflow-exploit-creates-trillions-of-ethereum-tokens/ (https://cryptoslate.com/batchoverflow-exploit-creates-trillions-of-ethereum-tokens/))

Quality

Perhaps the most important aspect of this project is the presentation. Your report should explain each step in your solution, and provide good visuals. You may use the ggplot2library to draw plots. Data science is the art of finding and presenting actionable insights from data. This report may be a good part of your job application portfolio, so please do your best. Your output will be a doc/pdf or html file. RMD files will not be accepted, because we will have token data files access in the code. A viewer may not have these files.

Question 1 [Due 5/1/2019]

Find the distribution of how many times a pair users (i.e., address1 and address2) 1 - buys, 2 - sells a token with each other. Which distribution type fits these distributions best? Estimate population distribution parameters.

Question 2 [Due 5/1/2019, full project report due 5/8/2019]

This question is similar to the first question. You will find the most active buyers and sellers in each of your three token network, and track them in other tokens. Develop a regression model where “buys” of the top K buyers (by number of buys or amount of buys) are regressors, and token price is the outcome. Determine a K value to have the best regression results. This means that you will develop three regression models for three tokens, and K can be different for each model.

Bonus 10 points (i.e., 100+10) Currently, there is no global clasification for tokens. But we know that a token may be related to certain industry; storj is used to buy online space, so it is related to IT, technology, etc. We are currently developing a categorization for tokens based on their usage, utility etc. Develop your own classification and put at least 70 tokens in this classification.

Bonus points will be added after averages and letter grades are computed; this means that not doing this bonus part cannot reduce your final grade. If your exams are already satisfactory, you can skip the bonus part. Bonus points will be graded with utmost care; half-hearted attempts will not receive any bonus. If you do not want to put considerable effort, it will be a waste of your time. Bonus submission will be done separately. The format will be announced.

Grading

1. Describe Ethereum and ERC20 tokens (Always state the source when you copy paste from somewhere. Try to develop your writing. Copy pasting more than 2-3 sentences look bad.) The book Elements of Style is sold in the book store; it is a must read to develop technical writing.
2. Describe your primary tokens

3. Describe what you are trying to model
4. Remove values that are impossible
5. Visualize and explain your data
6. Find outliers, and remove them if they exist
7. Explain what distributions could fit this data and why
8. Explain what functions and packages you are using to fit your data.
9. Fit your data, explain your findings
10. Write a conclusion on your findings, and why they are important.
11. Keep the report below 8 pages- use any article format you choose. Anything above 8 pages will be returned without grading.