Unsupervised Learning of Bitcoin Transaction Data

AMSC 663/664 Project Proposal

Stefan Poikonen

Table of Contents:

Abstract: Bitcoin is largest cryptocurrency, surpassing $8 billion in market capitalization in 2014. There exists a sizable body of research regarding the economics, anonymity, and culture of Bitcoin. Our approach is to utilize unsupervised clustering algorithms to cluster transactions on the Bitcoin network. Reid and Harrigan describe a method to form transaction edge lists from the raw Blockchain. We then augment each transaction line with aggregated user data and "tag" data of both the sender and recipient. We then implement Principal Component Analysis (PCA) followed by a number of clustering algorithms including K-means, Fuzzy C-means (FCM), and Clustering Using Representatives (CURE) Algorithm. Finally we evaluate the effectiveness of the clustering algorithms upon the dataset.

1. Project Background/Introduction

Bitcoin is the largest decentralized virtual currency with a market capitalization surpassing $8 billion in early 2014 [Aiken, 2014]. A Bitcoin does not exist as a file or physical entity. Rather, a public ledger maintains a log of all past transactions in the BTC network. To spend a Bitcoin, a user applies his/her private key to act as a form of digital signature. The signed transaction is then sent to users on the Bitcoin network for verification. "Blocks" of transactions are verified by "miners' by solving cryptographically hard problems. Miners are then awarded Bitcoins for their work. This award is how new Bitcoins come into circulation, thus the minting of Bitcoin is also decentralized, in contrast with traditional currencies with a central bank [Nakamoto,2008].

The Bitcoin currency is still quite young, launched on January 1st, 2009. Consequently, research on Bitcoin (and other crypto-currencies) is in its infancy. Some research concerning

global transaction volume, exchange rates, and even deanonymization of large sets of users within the Bitcoin network has been published. [Biryukov, et al., 2013][Moore and Christin, 2013]. However, there exists little research in the way of categorizing Bitcoin transactions, utilizing available Blockchain data.

By comparison the body of research of machine learning techniques (and its application), supervised and unsupervised, for categorizing other major financial transaction types (credit card, stock exchanges, etc.) is significantly more developed.

2. Project Goal

Cluster each transaction within the Bitcoin network utilizing various unsupervised machine learning algorithms. Measure the effectiveness of these clusters in an objective manner. Evaluate the concentration of the Bitcoin network. Compile a list of transactions with a large distance from the nearest cluster centroid; these may be anomalous transactions.

3. Approach

First, I will download an updated raw Block chain dataset. As of this writing, the current Blockchain is 22-23 GBs. Reid and Harrigan describe methods for transforming the raw Blockchain data into a set of tables, one of these a table of transaction lines. These methods will be applied, producing slightly less than 50 million transaction lines. Each line includes source ID, destination ID, timestamp, and transaction amount.

Next I will download a table of historic BTC/USD conversion rates. I will then convert each transaction amount across the BTC network into USD equivalent. The value of a Bitcoin has grown massively since the beginning of the BTC network; converting to USD will be a normalizing factor.

We note that source ID and destination ID may serve as index variables by which we may aggregate user-level statistics Examples of user-level statistics to be computed are
:

- Total USD sent
- Total USD received
- Highest USD value transaction
- Total number of transactions as sender
- Total number of transactions as recipient
- Timestamp of first transaction
- Timestamp of most recent transaction
- “Average timestamp” of transactions, weighted by USD
- USD value of BTC still possessed by user
- Ratio of average value in USD of incoming transaction/outgoing transaction

Blockchain.info contains a database of tags for certain public addresses. Tracing back through a number of tables generated by the Reid and Harrigan methods, one can associate a user ID with these tags. These tags may be categorized (i.e. gambling, political, electronics vendor, YouTube channel, etc.) For each user, we calculate the number of transactions had with different categories of tags.

We then collate this data by transaction line. A collated transaction line will likely include the following data elements:

Transaction ID, Source ID, Destination ID, Timestamp, Amount in USD, Total USD sent by Source User, Total USD sent by Destination User, Highest USD value transaction by source user, Highest USD transaction by destination user,…,# of transactions by source with gambling tagged sites, # of transactions by destination user with gambling tagged sites, etc.

We then perform a principal component analysis via singular value decomposition. Doing so has two main benefits. Firstly, it reduces the dimensionality of the data. Secondly, it solves the problem of defining a distance metric between vectors containing different data types. (If we had training data and utilize supervised learning techniques, it would be possible to learn a linear metric as proposed by [Xing, et al.]. However, lacking a pre-categorized training set, we choose to use PCA.)

We then utilize a number of unsupervised machine learning techniques to form clusters. These techniques will include: k-means clustering, c-means clustering with fuzzy logic. Time permitting, additional unsupervised clustering methods will be implemented and the parallelization will be incorporated. The CURE Clustering Algorithm (utilizing a sampling method) and the Nearest-neighbor Chain Algorithm in particular have been considered for later implementation.

Lastly, we consider those transactions whose distance away from the nearest centroid of a cluster is unusually large; these maybe anomalous transactions.

4. Scientific Computing Algorithms

1) Principle component analysis/Singular value decomposition
2) K-means clustering (utilizing heuristic initialization of clusters)
3) C-means Clustering with Fuzzy Logic (utilizing heuristic initialization of clusters)
4) CURE Clustering Algorithm (time permitting)
5) Nearest-neighbor Chain Algorithm (time permitting)
6) Approximate parallelized versions of the above algorithms (time permitting)

5. Implementation

Code will be implemented primarily C/C++ and run on a desktop with an Intel i5-3570K CPU and 16GB of DDR3 RAM. Some initial parsing of the block chain may be done in Python as convenient. If time permits, CUDA and/or OpenMP might be used for CPU and/or GPU parallelization respectively for key computationally intensive segments.

6. Validation Methods

The nature of the Bitcoin problem necessitated the use of unsupervised learning, precisely because there exists no external validation sets to compare categorization. Raskutti and Leckie document several evaluation criteria for unsupervised clusters. All revolve around the themes of compactness, separateness, and distance minimization.

One criterion [Raskutie and Leckie,1999] is as follows:

$$(\Sigma_{C_j}(\frac{max(D(R_i,R_k))}{min(D(C_{jc},C_{mc}))}))^{-1}$$
with $R_i, R_k \in C_j$ and C_{ac} represents the centroid of cluster a.

, where each term within the summation is the ratio of the maximum distance between points in the same cluster to the shortest distance from the current cluster's centroid to a neighboring cluster's centroid. Unfortunately, this is highly sensitive to outliers. The log transformations and normalization of data prior to PCA may alleviate outlier sensitivity. We may view this as utilizing the infinity norm as a measure of distance.

Another criterion computes a ratio. The numerator is the average distance between a data point to its cluster's centroid. The denominator represents the average distance between a data point and the mean.

$$\frac{Avg(D(R_i,Centroid(Cluster(R_i))))}{Avg(D(R_i,Centroid(AllData)))}\forall R_i \in R$$

Utilizing averages reduces the impact of outliers.

A slight deviation from the above formulation would take the mean squared distance in both denominator and numerator. The result of the below criterion is the unexplained square distance by the clustering:

$$\frac{Avg(D(R_i, Centroid(Cluster(R_i)))^2)}{Avg(D(R_i, Centroid(AllData))^2)} \forall R_i \in R$$

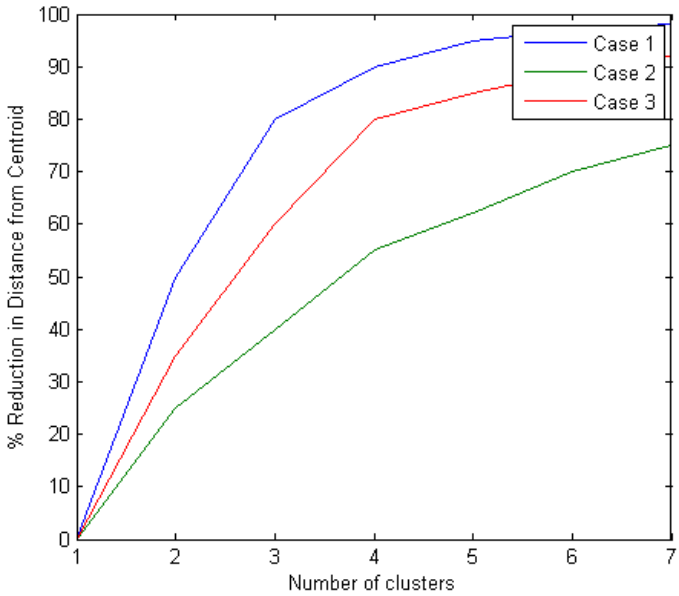
7. Test Problems for Verification

We can verify that the various clustering techniques have been properly implemented by running them on sample data sets from the UCI Machine Learning Data Repository and comparing our results to results of previous implementations of the same techniques on the same datasets.

Our primary test problem is clustering tens of millions of transaction lines in the Bitcoin network. There is no widely known previous implementation on this same exact dataset. Therefore, checking reduced distance and increased compactness (as described in part 6 above) verifies the success of the algorithms.

8. Expected Results

Certainly we should expect as the number of clusters, k, increases, variance and average distance from centroids of clusters will decrease, and compactness within clusters increases. At the limit, as k → n, variance and average distance from cluster centroid → 0. What is less certain, however, is how quickly unexplained variance tends towards zero as k increases. See the graphic below for reference:



For instance, case reduces average distance from centroids more quickly than case 2 or case 3, indicating a greater compactness of clusters. We may view this as receiver operator curve and measure the area under curve (AUC) is a metric determining efficacy of clustering technqiues.

9. Concluding Remarks

Though much has been written about the Bitcoin network, no widely circulated analyses has attempted clustering of transactions with data elements beyond price and timestamp. The lack of external training sets forces us to utilize unsupervised methods. Though, predictive validation is impossible, we may measure reduction in distance from the centroid with a number of metrics.

10. Timeline

Dates are approximate.

Phase 1 – October 1 to November 15: Data Download, Transformation, etc.

- Presentation/Proposal
- Blockchain download
- Transformation of Blockchain data to usable transaction line table
- Computation of user-level metrics
- Computation of tag-related metrics

Phase 2 – November 15 to November 28: Principal Component Analysis

- Normalize data. Determine whether to apply log transformation on certain columns of data.
- Implement Principal Component Analysis via Singular Value Decomposition

Phase 3a – November 28 – December 15: Implementation of Clustering Algorithms

- Implement K-means clustering

[Winter Break divides Phase 3]

Phase 3b – January 30 – March 30: Implementation of Clustering Algorithms

- Implement C-means clustering with Fuzzy Logic
- Implement CURE Clustering Algorithm (time permitting)
- Implement other clustering (time permitting)

Phase 4: April 1 – April 25: Analysis of Results

- Running the code of varying k
- The computation of cluster evaluation criterion from (6: Data Validation) above
- Analysis of clustering results
- Identification of potential anomalies

[At this point, if the project is ahead of schedule, I will attempt to parallelize key segments of code.]

Phase 5: April 25 – May 15: Final Paper and Presentation

- Finish final paper
- Make final presentation
- Consider routes of further research

11. Milestones

Milestones coincide with the completion of phases 1, 2, 3, 4 and 5 above listed in the above section.

12. Deliverables

- C++/Python code for transforming data to transaction line table
- C++ code for computing user-level metrics
- C++ code for computing tag-related metrics
- C++ code for normalizing data prior to PCA
- C++ code for computing K-means clusters
- C++ code for computing Fuzzy C-means clusters
- C++ code for other clustering (time permitting)
- Evaluation metrics from clustering with different numbers of clusters across different clustering algorithms
- First-Semester Progress Report
- Mid-Year Status Report
- Final Reports
- Weekly Reports

13. Bibliography

Aiken, Michael. "Future Funds: The Latest on Bitcoin and Cryptocurrency." *Diplomatic Courier*. Diplomatic Courier, 4 Sept. 2014. Web. 02 Oct. 2014.
Nakamoto, Satoshi. "Bitcoin: A peer-to-peer electronic cash system." (2008): 28.

Biryukov, Alex, Ivan Pustogarov, and R. Weinmann. "Trawling for tor hidden services: Detection, measurement, deanonymization." *Security and Privacy (SP), 2013 IEEE Symposium on*. IEEE, 2013.

Moore, Tyler, and Nicolas Christin. "Beware the middleman: Empirical analysis of Bitcoin-exchange risk." *Financial Cryptography and Data Security*. Springer Berlin Heidelberg, 2013. 25-33.

Raskutti, Bhavani, and Christopher Leckie. "An Evaluation of Criteria for Measuring the Quality of Clusters." *Telstra Research Laboratories* (1999): Web. <http://ww2.cs.mu.oz.au/~caleckie/ijcai99.pdf>.

Guha, Sudipto, Rajeev Rastogi, and Kyuseok Shim. "CURE: An Efficient Clustering Algorithm for Large Databases." (1998): Web. <http://www.cs.sfu.ca/CourseCentral/459/han/papers/guha98.pdf>.

Guha, Sudipto, Rajeev Rastogi, and Kyuseok Shim. "CURE: an efficient clustering algorithm for large databases." *ACM SIGMOD Record*. Vol. 27. No. 2. ACM, 1998.

Ding, Chris, and Xiaofeng He. "K-means clustering via principal component analysis." *Proceedings of the twenty-first international conference on Machine learning*. ACM, 2004.