Title: Estimating future Bitcoin profitability

Author: Trenton Potgieter

Date: 10/19/2014

Introduction:

Bitcoin is software that tracks and verifies transactions on a public ledger over a peer-to-peer network. Operations and data associated with Bitcoin are decentralized, meaning they are not performed or stored in one single location. Instead, the Bitcoin network consists of computers across the world that automatically store and relay Bitcoin data to each other. The computers' owners voluntarily choose to use and run the Bitcoin software. Anyone can use and run Bitcoin software. Bitcoins are produced through a process called mining. Mining is the competitive use of computational power to calculate a number that falls within a certain range. The first miner to discover a number “target” [1] that meets the criteria is rewarded with a set amount of brand new bitcoins (currently 25 bitcoins), the competition then repeats for the discovery of a new number [2]. In other words, when the number in the block header is equal or lower than the “target”, the block is accepted by the network and the process to create a new block begins.

Understanding just how difficult [3] it is to discover the given number below the given target is the key to determining profitability, or measure of success. More importantly is the ability to estimate future mining profitability by predicting what the network difficulty will be. This paper describes the process of forecasting what the network difficulty will be for a two year period and calculating how the profitability based on the current hashrate of the mining hardware.

Methods:

Data Collection

The data used for working with the current and previously recorded Network statistics (including Difficulty) is downloaded directly using the R programming language [4]. from the Bitcoin Network by using the Bitcoin Block Explorer website [5] which is an open source web tool that allows a user to view information about the blocks, addresses, and transactions created by Bitcoin.

<INSERT R CODE>

Exploratory Analysis

Each row within the data is a log containing the following:

- Time when the block was created (UTC)
- Decimal target
- Difficulty
- The average number of hashes it takes to solve a block at this difficulty

For the process of analysis, Time and Difficulty values are extracted. Any duplicated values are removed and the data is converted from UTC to a standard time format for easier sub setting into monthly periods.

<INSERT R CODE>

Time-Series Modeling

To model the predicted forecast of the network difficulty, the Arima model is used. Using the auto.arima() function in R uses a variation of the Hyndman and Khandakar [6] algorithm which combines unit root tests, minimization of the AICc and MLE to obtain an ARIMA model. The algorithm follows these steps: [7]

1 The number of differences **d** is determined using repeated KPSS tests.
2 The values of **p** and **q** are then chosen by minimizing the AICc after differencing the data **d** times. Rather than considering every possible combination of **p** and **q** , the algorithm uses a stepwise search to traverse the model space.
    a) The best model (with smallest AICc) is selected from the following four: ARIMA(2,d,2), ARIMA(0,d,0), ARIMA(1,d,0), ARIMA(0,d,1). If **d=0** then the constant c is included; if d≥1 then the constant c is set to zero. This is called the "current model".
    b) Variations on the current model are considered:

        vary **p** and/or **q** from the current model by ±1 ;
        include/exclude c from the current model.

        The best model considered so far (either the current model, or one of these variations) becomes the new current model

    c) Repeat Step 2(b) until no lower AICc can be found.

<INSERT R CODE>

Results:

By using the auto.arima() to select the best Arima model, we get a forecast (by month) of the Network Difficulty for the next 24 months, as well as the 80% and 95% prediction intervals for those predictions.

<INSERT R CODE + PLOT>

This data can then be used to estimate future profitability, based on the current hashrate of the mining hardware and the forecasted difficulty by using the following equation [3]:

BTC earned per day = Block Reward / (Difficulty * 2^32 / Hashrate / seconds in a day)

Where:

1) Block Reward is currently 25
2) Difficulty is the current or forecasted difficulty
3) 2 * 2^32 is a constant
4) Hashrate is the current hashrate from the mining equipment
5) Seconds in a day is 86400 (1 * 60 * 60 * 24)

Conclusions:

By using the difficulty forecast, current hashrate of the mining equipment and the profitability equation, the estimated amount of Bitcoins earned per month can be predicted.

<INSERT R CODE>

References:

1. Target page. URL: https://en.bitcoin.it/wiki/Target. Accessed 10/19.2014
2. How Bitcoin works page. URL: https://bitcoinhelp.net/know/how-bitcoin-works. Accessed 10/19/2014.
3. Difficulty page. URL: https://en.bitcoin.it/wiki/Difficulty. Accesses 10/19/2014.
4. R Core Team (2012). ”R: A language and environment for statistical computing.” URL: http://www.R-project.org
5. Bitcoin Block Explorer page. URL: http://blockexplorer.com. Accessed 10/14/2014.
6. Rob J. Hyndman and Yeasmin Khandakar. *Journal of Statistical Software* (2008), 27(3).
7. Forecasting: Principles and Practice online textbook. URL: https://www.otexts.org/fpp/8/7. Accessed 8/23/2014.