**Motivation:**

Bitcoin is a digital currency is that not regulated by any government and therefore does not follow traditional financial market patterns. Over the past year, Bitcoin has gained considerable media coverage and increased tenfold in value (weighted to the US dollar). Having known about Bitcoin since 2008, I wanted to explore what relation, if any, traditional media sources have to Bitcoin values. To do this, I initially proposed to do a sentiment analysis of all the articles published online in the British newspaper, The Guardian, that were returned by a content search for ‘Bitcoin’ for a six-month period (August 8, 2013-February 3, 2014) and plot article sentiment against market values by date. However, give the recent plummet of Bitcoin (February 25, 2014), I decided to expand the research period from 6-months to 1-year in order to capture Bitcoin’s incredible growth and its unexpected fall. In addition to increasing the data time period, I also decided to run The Guardian articles through two sentiment text analysis APIs to have richer data output for analysis, one which gives feedback as positive, negative, neutral from text that you feed it and the other which gives feedback as positive or negative with a sentiment score ranging from -1 to 1. This allows us to look at relative sentiment of The Guardian coverage of Bitcoin for certain Bitcoin values. In addition to simple sentiment analysis, I hoped to locate particular market changes in Bitcoin and create wordclouds from The Guardian to further contextualize the media coverage around significant Bitcoin market activity.

**Data Sources:**

The first dataset I used was the Bitcoin information available from [Bitcoin Charts](http://bitcoincharts.com/charts/mtgoxUSD#rg60zczsg2013-08-08zeg2014-02-03ztgWzm1g10zm2g25zv). This site allows for the following daily values for Bitcoin to be generated for a given time period in a chart format and in a table format:

* Timestamp (yyyy-mm-dd hh:mm:ss)
* Open
* High
* Low
* Close
* Volume (BTC)
* Volume (Currency)
* Weighted Price.

For my purposes, I set the time period from 2/25/2013-2/25/2014 and copied the raw data from the table format into a csv spreadsheet where I formatted the timestamp variable to only include yyyy-mm-dd because The Guardian API did not return hourly information. This analysis only uses the timestamp, highs, lows, and weighted price columns of Bitcoin data so all other columns were removed directly in Excel. This was preferable to modifying the data programmatically since there were no missing values to address and the columns were nicely formatted from their originating source.

The second dataset I used was [The Guardian Content Search API](http://explorer.content.guardianapis.com/#/). There is excellent documentation available at the above link on how to use this API. The Guardian Content Search API (gAPI) takes multiple arguments (keyword, #results per page, #pages, tags, dates, etc) and returns up to 50 results per page in a JSON string. For the time period of 2/25/2013 – 2/25/2014, there were 247 results that contained the keyword ‘bitcoin’. This number is available via API Explorer and was helpful for thinking about how to structure calls to the API. I called the API 5 times, once for each page containing 50 results to give me a total of 247 results. The variables returned in the JSON string are:

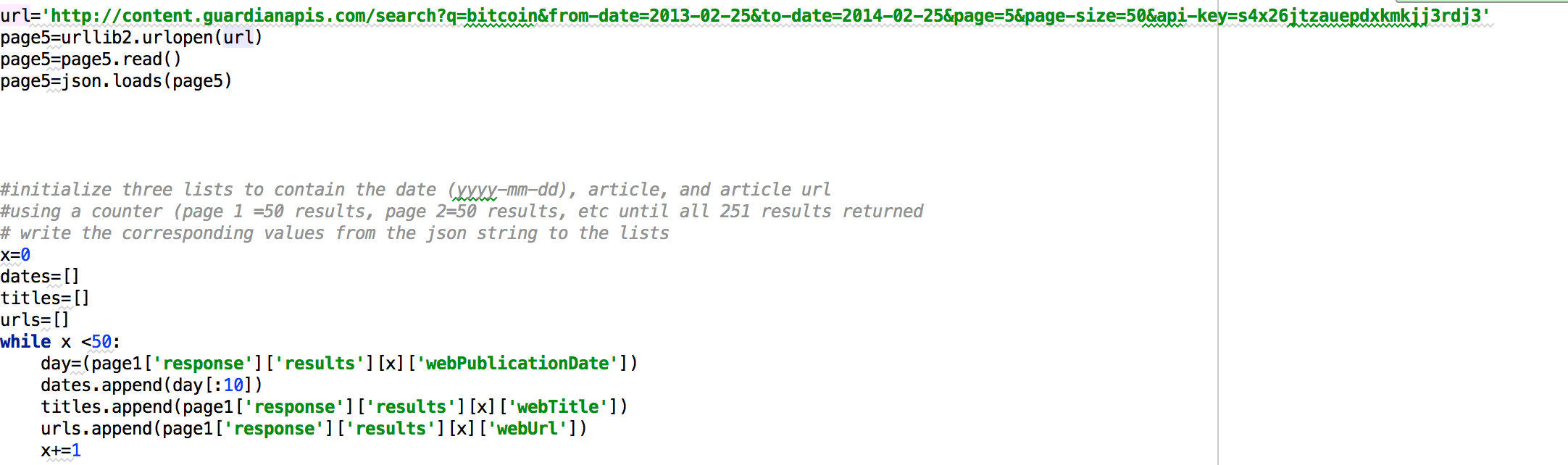
* Id
* sectionId
* sectionName
* webPublicationDate
* webTitle
* webUrl
* apiUrl.

The variables of interest for this study were webPublicationDate, webTitle, and webUrl. These were extracted from the full JSON string using Python and are explained in the methods section.

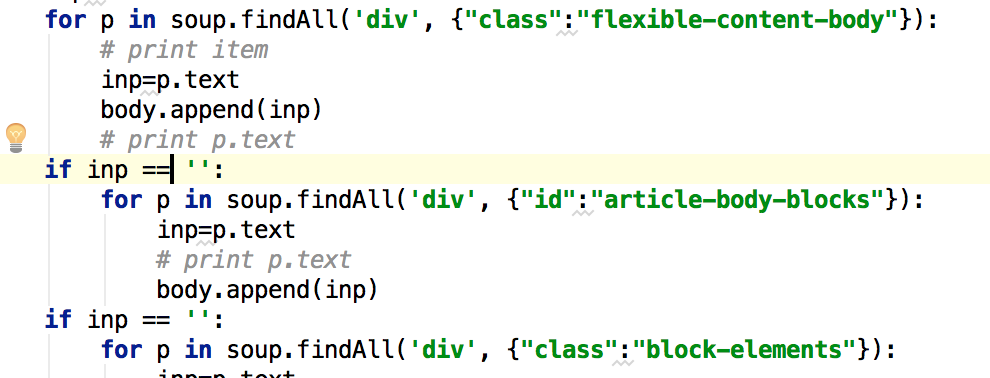
I used two different sentiment text analysis APIs to conduct sentiment analysis on the article title and the article content. The first of these is the [Datum Box API](http://www.datumbox.com), which takes an api\_key and the text to be analyzed as parameters (without html tags) and returns a JSON string containing the sentiment analysis which is one of positive, neutral, or negative. In order to use this API, it was necessary to scrape and parse the html from each Guardian article before sentiment analysis could take place. In contrast to this, the second sentiment text analysis employed[, Free Natural Language Processing Service (FNLPS) available from Mashape](https://www.mashape.com/loudelement/free-natural-language-processing-service#!documentation), takes a URL as the only parameter and returns a JSON string containing both a sentiment score ranging from -1 to 1 and a sentiment analysis which is either positive or negative.

**Data Manipulation Methods**

The Guardian data required the most manipulation. Since the API will only return a maximum of 50 results per call, I had to call the API 5 times using urllib2. I iterated through the returned JSON string to append the date, webTitle, and webUrl to lists as seen in the code snippet below.



After returning all the article dates, titles, and urls to lists, it was also necessary to get the article content in order to do a sentiment analysis programmatically. The first step in this was to use urllib2 again to scrape and parse the html of each article’s webpage. This was not necessarily easy as each article webpage is not uniform and it was necessary to view source code for multiple pages to get a sense for which HTML tags to search for and thus return text for. It seemed that the most plausible way to do this was to look for divs that contained the article content and return the text from these. The following code demonstrates how this was done. If the article did not contain any of these tags, ‘None’ was appended to the body list where the article content was stored in order to deal with null values or irretrievable article content without examining the source code for all 247 articles.



Once the article content was appended to the body-list, it was necessary to encode the text as utf-8 in order to pass it to the Datum Sentiment analysis API. This was realized after receiving multiple encoding errors from the Datum Box API and then maxing out the 1000 API calls/day. Following the encoding phase, the date list, the url list, the encoded-title list, and the encoded-body list were combined using ZIP and also saved to an output file so that they could be retrieved and analyzed by the second sentiment text API. Iterating through the zip file, the values for encoded-title and encoded-body were passed to the Datum Box Api which then returned a value of positive, neutral, or negative for each article title and body in our list, and these values were appended to separate lists called etitle and ebody. These sentiment lists were then combined with the date list and the url list via ZIP and also output to a csv file with utf-8 encoding.

The result of the above manipulation gives us two files, one that contains all article content by date and one that contains sentiment analysis by date.

To use the FNLPS, I created a new python script that opened the output file containing all the guardian urls and dates and passed the urls to the FNLPS API which returned both a sentiment score and a sentiment-text of positive or negative. This API failed multiple times so I added the a 5 second pause between each API call and that seemed to help, but it did not work all the time. It eventually worked flawlessly late at night when I presume the number of people calling the API was reduced. I appended each sentiment score and sentiment text to a list while iterating through all 247 urls returned from the Guardian. Then I once again used ZIP to combine dates, urls, sentiment scores, and sentiment text output to a single list. Iterating through the list, I wrote it to a csv file as seen in the below code.



Now that the Guardian data is formatted, it is necessary to combine with the Bitcoin Values but first, this file required some minor clean up. The first manipulation I did was to remove the columns that would not be necessary for this analysis, keeping only timestamp, high, low, and weighted value. As previously mentioned, I formatted the timestamp in Excel to be yyyy-mm-dd to match the returned date from The

Guardian API JSON.

To combine my CSV files (Datum Sentiment output, FNLPS Output, Bitcoin Values, Body Output), I used the Pandas Python library and the csv\_read and merge functions. I first merged the datum sentiment output with the Bitcoin values using the date(timestamp) column to join them. I wrote this merge to a csv file. I then read this csv file in again and merged it with the FNLPS output and wrote that to a file. Writing these merges to file allowed me to inspect that duplicate values were not present and the data was complete and in the desired format. The file that I now have contains all the Bitcoin values, the Datum Output, and the FNLPS Output arranged by date. One last manipulation that I wanted to perform was to group the articles by date in order to see which dates produced the most Guardian content and thus enabling a comparison of prolific publication to Bitcoin values. To group by date, I used a function of the Pandas library and then wrote this to file and subsequently read it and merged it with the previous file mentioned giving me a final output file with the following columns:

* Timestamp (date)
* Url
* Sentiment Score
* Sentiment Text
* Title (Datum)
* Article (Datum)
* High
* Low
* Weighted Value
* Number Articles Published

At this point, there is no further data manipulation necessary via Python. The remaining manipulations will take place within Excel.

In order to chart the sentiment of each article against Bitcoin values by time, it was necessary to turn ‘positive’, ‘negative’, and ‘neutral’ into numeric values of 1, -1, and 0. I did this using the find and replace function of excel per the columns containing sentiment outputs. I then computed a column of weighted or cumulative values per sentiment output by summing the sentiment values by day. These new columns accurately depict rises and falls as an increase in positive articles or negative articles. I also created a daily change column that measures the difference between the high and the low value of Bitcoin in case that I wanted to explore the relationship between reporting and daily changes in value.

**Analysis and Visualization**

With the combined data of Bitcoin value, FNLPS output, and Datum output, I was able to look at the The Guardian publications in relationship to the changing Bitcoin values. An initial analysis was to look at how many articles were published by day in relationship to Bitcoin values. To do this, I plotted the weighted valued of Bitcoin by day using an Excel line chart with the left-hand axis being value in USD against the number of articles published using an area graph with the right-hand axis being total number of articles published. Charting these Bitcoin values against the number of published articles revealed that there was an increase in publishing while Bitcoin increased in value that was sustained during Bitcoin’s rapid price increase from November 2013-December 2013. Figure 1 provides a visualization of what this looks like where the Bitcoin weighted values are plotted using a line chart with the left-axis representing US dollar values and the number of Guardian publications is shown using an area graph with the right-axis representing number of articles. Closer inspection of this graph in the attached Excel document reveals a correlation between the number of articles published and Bitcoin values over time. The November-December 2013 period and late February 2014 represent the strongest of these correlations.

Figure 1

In addition to looking at how much The Guardian was publishing about Bitcoin in relation to the weighted price of one Bitcoin, I also looked at how positive or negative the articles were. To do this, positive scores were quantified as 1, negative scores as -1, and neutral scores as 0. Then I calculated a cumulative score for the Datum Sentiment Score and the FNLPS Score that shows the fluctuation in sentiment over time. This cumulative score was plotted against the Bitcoin value. Figure 2 shows this relationship. From looking at the cumulative scores for both Datum and FNLPS Figure 2, it becomes clear that there was more positive reporting while the value of Bitcoin was on the increase and more negative reporting as the value of Bitcoin fell for the period of November-December 2013. The FNLPS cumulative score correlated better with Bitcoin value than the Datum score which could in part be attributed to the fact that FNLPS parsed the content directly from The Guardian pages itself and scored articles as only positive or negative.

Figure 2

Due to the fact that there is a considerable peak in Bitcoin value and the sentiment analyses correlate with it, a word cloud was generated to look at article content during the rise (Image 1) and fall(Image 2) of Bitcoin in November-December of 2013. A word cloud was also generated for the entire year (Image 3) of content to look for any particular words of interest that appeared more frequently in one interesting period of Bitcoin activity. In order to get the words for each time period, I returned to the output file of the merge between Bitcoin values and Guardian Articles that contained the parsed webpage content. This content was selected and put into the word cloud generator available via the [Tagul](http://tagul.com/) website. This online word cloud generator allowed for custom shape output and for custom tag modifications. Common words were removed as were any words that were likely a product of website jargon like ‘photograph’ which would have been commonly returned from parsing the HTML. This was a challenging aspect of the word cloud analysis. The text input that I used to generate the clouds came from stripping HTML and therefore might contain some erroneous words. To minimize this, I used the aforementioned word remover solution but it is not perfect. Additionally, Guardian articles that were returned for Bitcoin are not necessarily exclusively reporting on Bitcoin.

**

Image 1: 11/18/2013 - 12/04/2013 Image 2: 12/05/2013 - 12/18/2013

**

Image 3: 2/25/2013- 2/25/2014

Overall, the combination of the Guardian dataset and the Bitcoin dataset demonstrate that there is a relationship between an Bitcoin market activity, frequency of media publication, and an increase in the positive sentiment of these publications.