11069 - Data Visualisation

Bitcoin Data Visualisation

14th October 2019

u3179196 Kim Hughes

Project Challenges

The data visualisation project was particularly challenging due to the plethora of visualisation tools available and approaches to creating an effective and successful visualization. After assessing the tools available I decided on using 2 tools - D3.js and Matplotlib. The other tools I assessed were Tableau, Plotly and Chart.js.

I settled on D3 and Matplotlib for numerous reason. D3 appeared to be the best data visualization library available, had a huge community, good documentation, runs on JavaScipt and uses HTML, CSS and SVG. Given my major in web design this seemed a good fit. Matplotlib seemed a good option from the perspective of demonstrating knowledge of more than a single tool and it being python based which would serve to expose me to further programming tools and skills. Additionally Matplotlib (like D3) had excellent documentation, a large community and produced plots in SVG format for displaying on web pages.

In retrospect I underestimated the learning curve in gaining a working knowledge of these tools (in particular D3) - but, of course, as an exercise in exposure to new technologies the choices were satisfying if not often frustrating.

Beyond steep learning curves, the challenge of choosing a dataset and approaches to plotting/ graphing them was also challenging. The dataset chosen was Bitcoin as an emerging force in digital assets and capital markets. Being interested in the subject generally helped settle the choice and the question became one of how to narrow the visualisation goal sufficiently considering the options available, time restraints and the complexity of the tools.

I settled on creating visualisations highlighting bitcoins strength as a viable option in the capital asset space and its dominance in the wider cryptomarket space. Additionally I wanted to show some elements of dynamism and interactivity in the D3 plots to demonstrate (an inkling of) the power of the tool and to push the limits of my capacities. This was via mouseover displays, tooltips on sparkline markers and selection of date ranges with (animated) plot updates.

Creating the visualisations

The starting point was identifying the goal (the foundation of a purposeful visualisation). In this case it was to prompt an action from the audience to further research bitcoin and in particular to position it as the high priority in wealth stewardship and portfolio diversification.

Understanding the data itself and verifying it best served to support the goal was more straightforward as I have been researching digital assets for sometime. The more significant issue in this respect was focusing on what the target audience should know in order to put the audience first. The audience in this case is those aware of impending problems in fiat currency markets and strategies to manage the risks together with a rudimentary awareness cryptocurrencies.

Choosing the right chart(s) was difficult. Presenting a dashboard or story succession approach (being cognisant of "too many often equals too much") seemed best and the final choices were based on the goal being satisfied (leading the audience to answers or directions for investigations) and my limited capacities in utilisation of the tools potential powers .

I endeavored to use labels giving the audience context and minimise explanatory elements by using short annotations or story points ensuring they added value. Further was a focus on eliminating superfluous chart features, unnecessary headers or labels and artsy details.

Ultimately the design choices were driven by the over-arching goal of letting the data speak and aligning the visualisations story with the needs of the audience.

From a technical perspective the Matplotlib charts were created from scratch whereas the D3 charts were adapted from examples on the D3 Example:bl.ockeplorer.org galleries. I needed to resolve an issue (not being aware of it initially) of multiple D3.js versions. I tried to modify the code for compatibility with a single version (the latest v5) but this proved too challenging given the time constraints and, fortunately, I found a way to run multiple d3.js versions. The D3 example charts used as (the substantial) foundations are acknowledged in the script.js comments.

Final reflections

Overall I am satisfied with the outcome but it leaves significant scope for improvements. Mostly these could not be achieved in the time-frame due my very limited capacity to fully utilise the power of these two tools. In particular I would point out issues with the x-axis of the 'Bitcoin Price with 300MA' chart (dates needed to shift to right on step) and the y-axis on both 'Bitcoin Dominance' and 'Bitcoin Hashrate' charts (how to show large numbers in shortened format and not as scientific notation).

 Further I was unable to fix the problem with the options element whereby it only worked one-way (down the list but not up the list). To go up the list requires a browser window refresh. Finally I would like to have added to each chart the sources of the dateset(s) involved.

Having full command of either or both tools would have significantly altered the approach to implementing the visualisation but, again, time and learning curves constrained my options and limited the outcomes possible. I resigned myself to consider the project as a trade-off between a visualisation project per se and an exercise in exposure to two powerful technologies to a lessor degree rather than one charting choice to a greater degree.