Victor Vahram Shahbazian
CMPS 161
March 12, 2017

Homework 2: Stream Graphs

Tools: Javascript, HTML, CSS, three.js, node.js

Data Source: https://www.quandl.com/collections/markets/bitcoin-data

Introduction:
Stream graphs are an interesting form of data visualization. Unlike most graphs that look boring and uninteresting, stream graphs are like the rock stars of data visualization. They don't stop by just being visually appealing, they also are a very good way of showing the increases and decreases of various topics. Unlike most graphs that require some knowledge of graphs in general, a person who is not scientifically trained can easily look at the graph and understand what is going on. This makes stream graphs a great way to allow the general public to understand the flow of data.

Objective:
The objective of this assignment is to create a circular stream graph using any kind data that is based over some amount of time. A challenge presented to us is on how to connect the beginning and end of the stream graph to make it look more uniformly cyclical. I deviated from the requirement a bit and decided to make the stream graph look a bit more interesting by bringing it to the realm of 3D. By doing this, I am able to create a graph that is cyclical in the form of a cylinder.

Data:
The data that I chose to use is the volume of Bitcoins given a certain currency. The currencies I was able to obtain data for were USD, PLN, JPY, GBP, EUR, CAD, AUD. The minimum number of days available for over all currencies is 883 days. I chose not to use the total available days, but instead to only use up to 90 days to make the stream graph easier to read. After inputting the data and creating the stream graph it became clear I had somewhat of a visual problem. Certain currencies such as USD and EUR had hundreds of times greater volume in Bitcoin that others, making the less used currencies almost invisible. To fix this I chose to instead calculate the percentage of change in volume from the current and previous day. Using a percentage instead of a static number made the stream graph look a lot more visually appealing as well as making it so that the currencies with much smaller volumes became much more apparent within the graph.

Implementation:
Using three.js I created a custom geometry and inputted vertices depending on the number of days I wanted to visualize. There are a total of seven custom geometries one for each of the

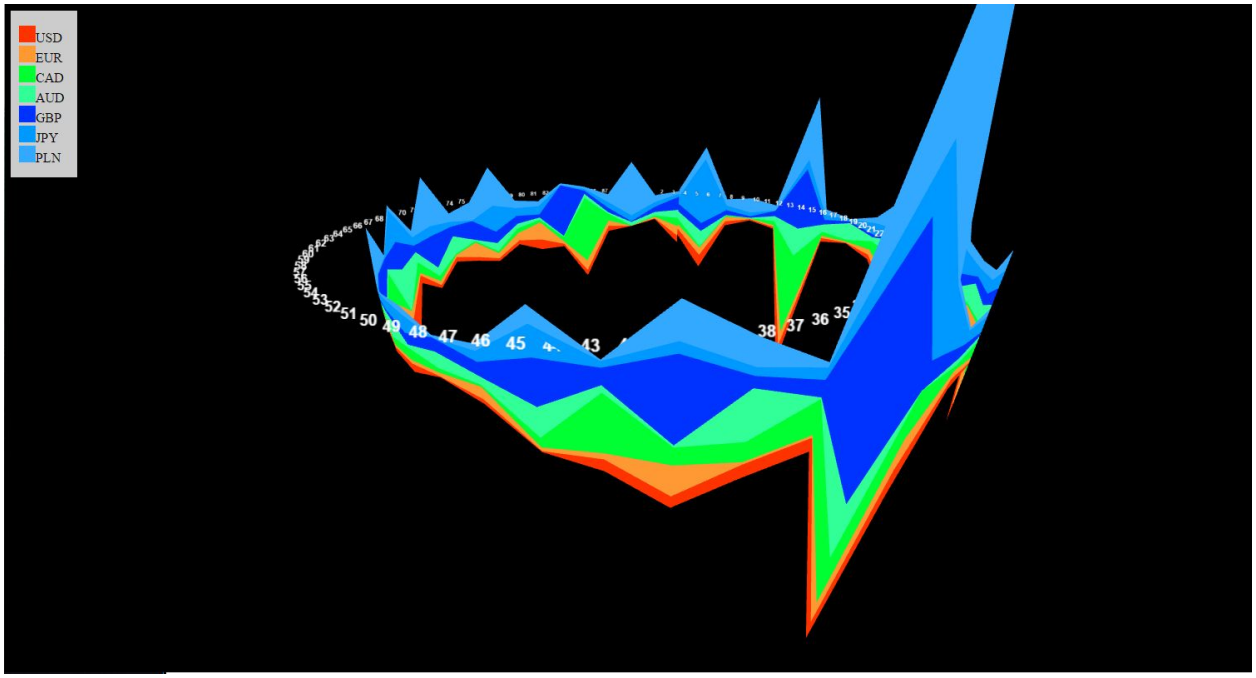
currency types. The bottom points for the first currency is the baseline(g0). The rest of the geometries used the stacked height of all the geometries that come before it, allowing the graph to be stacked as in a Stream Graph. The y component of the vertices being pushed into the graph is the value of the currency at a particular date. The x component is simply the number of days and can be manipulated to make the graph look much more visually appealing. An important note is that I'm using two triangles in the form of a quad to construct each day of each stream of each data point.

Baseline Calculation: g0 = -(1/(n+1)) * Summation[i=0 to n]{ (n-i+1)*fi } n = number of data sets

For each point per time, the data is represented as fi. The height of the stack at a given time is the baseline g0 plus the summation of data points up to the current data point.
fi = g0 + f1 + f2 + ... + fi

Bridging the Gap:
To connect the beginning and the end of the stream graph I've decided to use the accumulated average over all points, that have been stacked. I believe that this allows for the best representation of the graph as the larger the graph grows, the more accurate the connection will be. Given that cryptocurrency is somewhat of a new phenomenon it is difficult to fully predict how the value of the currency will change over time. There are many world factors that influence the value and those factors are more or less based on coincidence.

Data Analysis:
Observing the stream graph shows a lot of hills and valleys. Given that the increase and decrease in the volume of bitcoin per currency is all somewhat related, when one currency goes up, the rest go up as well. Same for when the volume of a currency goes down, all others go down as well. This is why I chose to take the percentage of difference from the previous to the next day, allowing to better visualize the changes between currencies. The currencies that were previously smothered by the currencies with higher volumes are now more apparent, making it easier to understand the changes between currencies. USD which was the most dominating currencies now shows much less change from day to day compared to GBP which shows massive fluctuations in volume.



Why Cylinder?:
I believe the cylinder does look visually appealing. I do not think there is much of a difference on whether the stream graph is cylindrical or circular. They both seem to be a good way of displaying cyclical data.

Problems:
The volume of some currency types is that it is so small that it is hardly visible on the stream graph. To fix this I took the percentage of change from day to day and used that as the data points. I'm also having trouble figuring out why the numbers aren't aligning with the graph. Everything is centered and the calculations seem correct but for some reason the center for the number is translated to the negative y axis.