Data Science Project 1

Scraping TikTok

Boyd Kane (26723077)

I. Introduction

TikTok is a popular social media platform, with its success often attributed to the proprietary algorithm it uses to choose what content to display to which users. The content entirely consists of short, vertical-format videos (called TikToks) which are presented to the user sequentially.

The algorithm presents TikToks to a user, and learns what they like based on how long a user watches a particular TikTok and how they interact with it. Often certain audio clips can go viral: being used in many different TikToks by many different creators.

This project does not attempt to analyse the audio or video content of the TikToks, but rather analyses the numerical quantities associated with each TikTok such as comments, likes, and the number of followers the creator has. The goal is to explore trends and attempt to find instances where a particular audio, TikTok, or creator went viral.

It was found that TikTok has an integer overflow error for creators with over 2^{31} likes on their profile (at the time of writing, only @khaby.lame and @bellapoarch). This resulted in negative and malformatted likes appearing in their profile previews (see Figure 1). This error was corrected for in the analysis.

Many TikToks were found that went viral, accumulating millions of likes in less than a week. These viral TikToks were often followed by the creator gaining hundreds of thousands of followers.

Often the ratio of likes to comments on a TikTok were similar for a given creator, implying creators induce a consistent impulse on their viewers to like or comment, and that this impulse does not change as the TikToks becomes more popular.

Fig. 1. `khaby.lame`'s profile preview shows negative likes, indicative of an integer overflow in the TikTok source code.

II. Implementation

A scraper for TikTok was written using the python libraries `scrapy`, `selenium`, and `requests`. It made use of a single spider `tiktok_spider.py`, and a single `TikTok` item. A single pipeline was used to parse any poorly formatted values into numbers and to ensure the resulting data was clean.

This crawler was run every hour via a cron job that executed the script `tiktok_scraper.cron`. The data is saved to `tiktoks.jsonlines` and then copied via `scp` to the author's machine for analysis.

Data analysis was done with Hunter (2007), The pandas development team (2020), Harris et al. (2020), and Waskom (2021).

III. Crawl process

The scraper loads the home page of TikTok and scrolls until data from 100 unique TikToks have been recorded. This constitutes one run of the scraper. A new run of the scraper was initiated at the start of every hour for two weeks, with each run of the scraper taking about 50 minutes. If the previous run hasn't finished by the time the next run starts, the previous run is killed.

The scraper does not log in to the TikTok website, so each run the scraper is shown whatever content the TikTok algorithm would show to an anonymous user every time the scraper starts up. This reduces the variance between runs, but has the disadvantage that it is impossible to analyse the ability of the TikTok algorithm to tailor TikToks to a user's interests. This project can be seen as a broad analysis of the TikTok algorithm on all TikToks, as opposed to a narrow analysis on how the TikTok algorithm explores and exploits a given user's likes and dislikes.

TikTok has several mechanisms to deter automated scrapers. Selenium is required as TikTok only presents two TikToks when the home page is initially loaded, and if JavaScript is disabled then nearly the entire content of the website is deleted within a second and an error message is shown to the user. To avoid this and other defences, a non-default view of the website is navigated to via Selenium. This non-default view does not include some of the Captchas, and by using Selenium for full browser automation, TikToks can be loaded as the crawler scrapes the website.

IV. Data scraped

For each time the crawler was shown a TikTok, the following values were gathered directly from the website. Many more derivative values were calculated from these, but they will be discussed later.

- `scraped_at`: The datetime at which the TikTok was scraped.
- `url`: The URL of the TikTok.
- `audio`: The name of the audio associated with the TikTok
- `audio_url`: The full URL of the audio associated with the TikTok (the audio name given by `audio` need not be unique).
- `likes`: The number of likes on the TikTok.
- `comments`: The number of comments on this TikTok.
- `creator`: The username of the creator of this TikTok. Always starts with an `@`.
- `creator_url`: The full URL to the creator's TikTok homepage.
- `creator_followers`: The number of followers that the creator has at the time the TikTok was scraped.
- `creator_likes`: The total number of likes given to this creator over all their TikToks. This value suffers from an integer overflow for values over 2^31, but this overflow has been corrected in pre-processing.

Note that TikTok does not show the fully accurate number of likes and comments. Instead, a summary like 12.3K or 45.6M is shown. These instances were parsed to complete numbers like 12300 or 45600000. This has the effect that the resolution of the values is dependant on the magnitude of those values; a change of 5000 is visible to the scraper if that change is from 5000 to 10000, but it not visible to the scraper if the change is from 1000000 to 1 005 000.

The resulting dataset contains 26 000 observations taken over the 15 day time period 2022-07-30 to 2022-08-14. There are 2000 unique TikToks, 1500 unique audio clips, and 1600 unique creators.

A. Defining Viral Content

There is no standardised academic definition for a 'viral' piece of content. For the purposes of this project, a viral piece of content is one which garners more interactions per unit time than a significant percent of other content on the platform. The interactions in question are platform specific but usually some combination of likes, followers, comments, shares, and saves.

Note that this definition intentionally does not account for how popular creators with millions of followers will more easily create viral content than smaller creators orders of magnitude fewer followers. This is because including the number of followers in the definition of a viral piece of content could cause a cyclical dependency issue: a viral piece of content could increase the number of followers the creator has, which could cause the

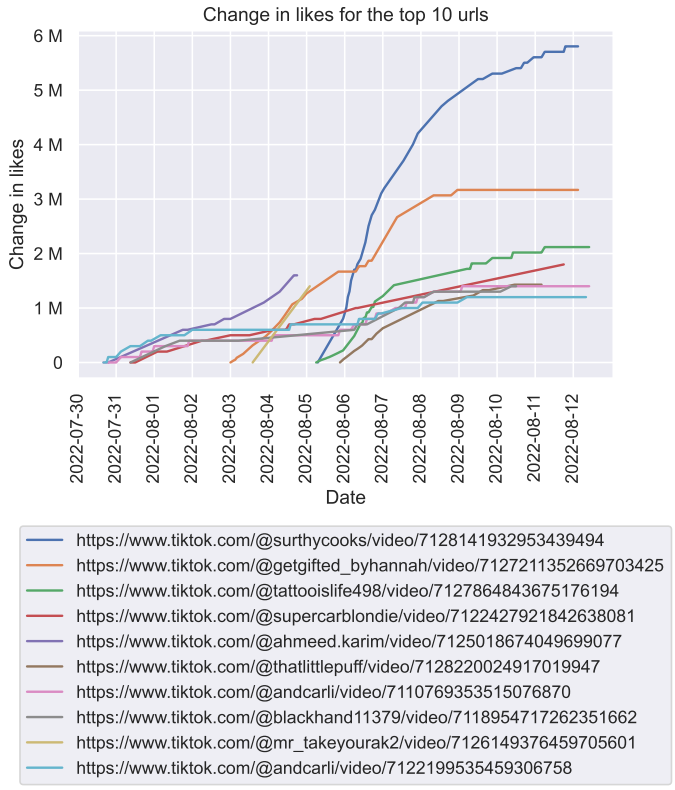

Fig. 2. Change in the number of likes for TikToks with the greatest increase in the number of likes.

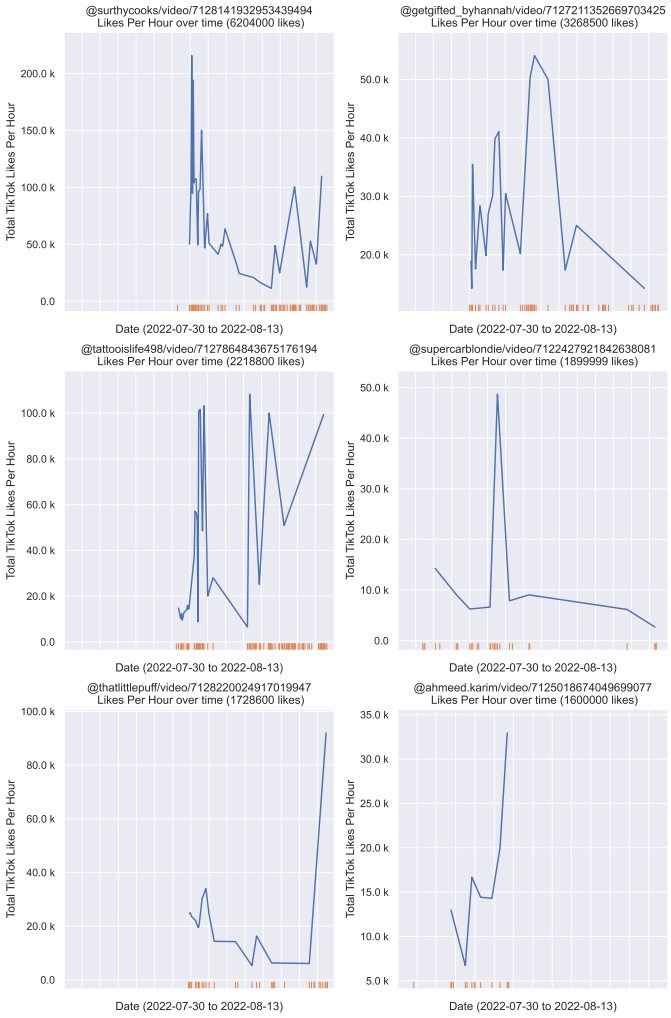

Fig. 3. Likes per hour for TikToks with the greatest increase in the number of likes. The orange rugplot shows every time that TikToks was shown to the crawler.

piece of content to no longer meet the definition of virality.

V. Results

A. Which TikToks saw the greatest increase in likes?

Figure 2 shows the change in the number of likes for 10 TikToks over the two weeks that data was gathered, with each TikTok represented by a different line. The TikToks shown were those which saw the greatest increase in likes.

This TikTok by @surthycooks (A home chef) accumulated five million likes in as many days, and at it's peak was gaining over 225 likes per hour (see Figure 3). The apparent jaggedness of Figure 3 is a by-product of how TikTok only shows likes, followers, and comments to three significant figures. This causes the change in likes as measured to be stagnant for a period of time (even if it is actually increasing) and then it will jump by several orders of magnitude. And so the likes per hour will appear very high when the number of likes crosses some decimal boundary (such as from 12.3 thousand to 12.4 thousand) but will remain zero for all the values in between (such as 12.31 thousand, 12.32 thousand, 12.33 thousand).

One can clearly see the steep increase in likes between 2022-08-06 and 2022-08-08, indicating that this TikTok was going viral during that time period.

After 2022-08-09, there is a decrease in the slope as the TikTok continues to gain views, but not at the rate it previously did.

The TikToks 2 (in orange) and 3 (in green) by @getgifted_byhannah and @tattooislife498 respectively each gained many likes, but not quite at the rate of the TikTok by @surthycooks.

Both TikToks went through phases of virality

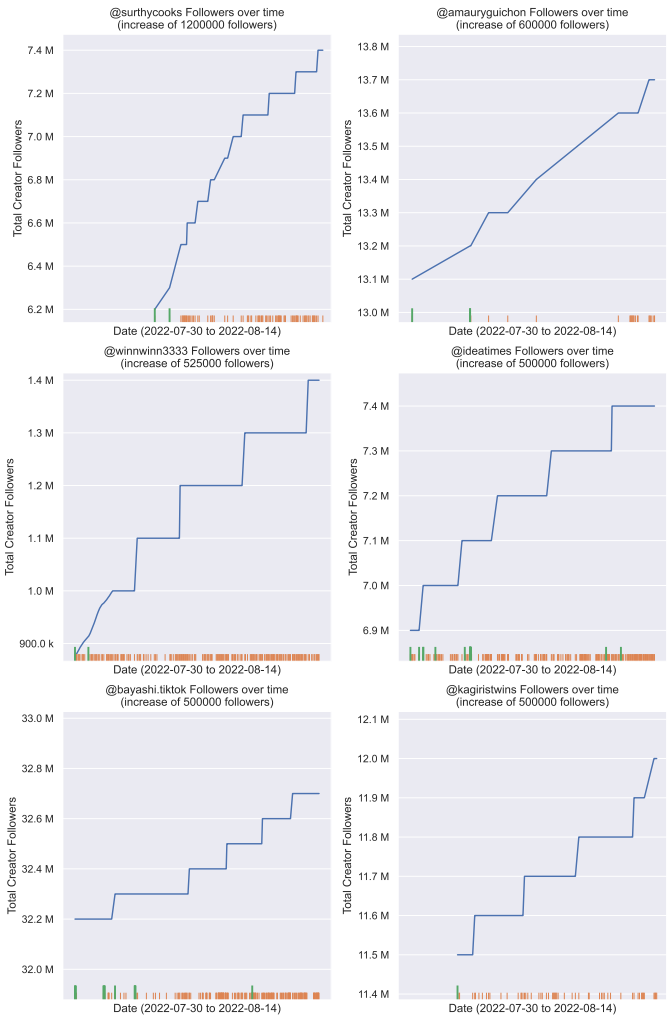

Fig. 4. The change in the top-6 creator's follower count over time. The green rugplot indicates the first time the scraper was shown a new TikTok, and the orange rugplot indicates every time the scraper was shown a tiktok by that creator.

(accumulating likes as a rate faster than many others) and phases of stagnation (such as the period at the start of 2022-08-06 for the TikTok by @getgifted_byhannah). The remaining seven TikToks did not appear to have moments of virality as pronounced as the first three. Rather they slowly accumulated likes over the period of two weeks.

B. Do viral TikToks cause an increase in followers?

Figure 4 shows the change in a creator's follower count over time.

Only the creators which experienced the greatest increase in followers are shown.

It is clear that @surthycooks experienced a massive increase in their follower count after the viral TikToks was released. Their followers grew by 1.2 million, double the growth of the next creator @amauryguichon (a chocolatier) who gained 600 thousand followers in the same time period.

The list of creators of the TikToks who gained the most likes (Figure 2) and the list of creators who gained the most followers (Figure 4) do not include the same creators, implying that gaining likes does not necessarily result in gaining followers.

However, Figure 5 shows the number of followers gained per hour against the number of likes per hour gained for every creator. The number of likes per hour is always greater than the number of followers per hour (as seen by how every point occupies the top-left triangle of the graph). This makes sense intuitively, as following a creator will almost always be associated with liking a TikTok made by that same creator. However it is rare for a user to choose to follow a creator without also liking one of their TikToks.

In Figure 5, one can see that the majority of creators have fewer than 25 thousand followers per hour, but the number of likes per hour often reached 100 thousand. This implies that many TikToks were going viral (receiving many likes per hour) but the viewers of those viral TikToks did not proceed to follow the relevant creator.

C. How to convert likes into followers?

Define the likes-to-followers ratio for a given creator as the number of likes that creator has divided by the number of followers that same creator has. Figure 6 then shows the frequency of different likes-to-followers ratios. If the likes-to-followers ratio was over 80, it was excluded due to a few outliers with a likes-to-followers ratio of over 500. The red lines indicate the 5-th and 95-th percentile of the trimmed data.

This histogram shows that 90% of creators experience between 4 and 38 likes for every follow they get on a TikTok, with the median of the data being one follower every 12 likes.

This would imply that if a creator can make TikToks which gain many likes, then some percentage of those likes will result in followers.

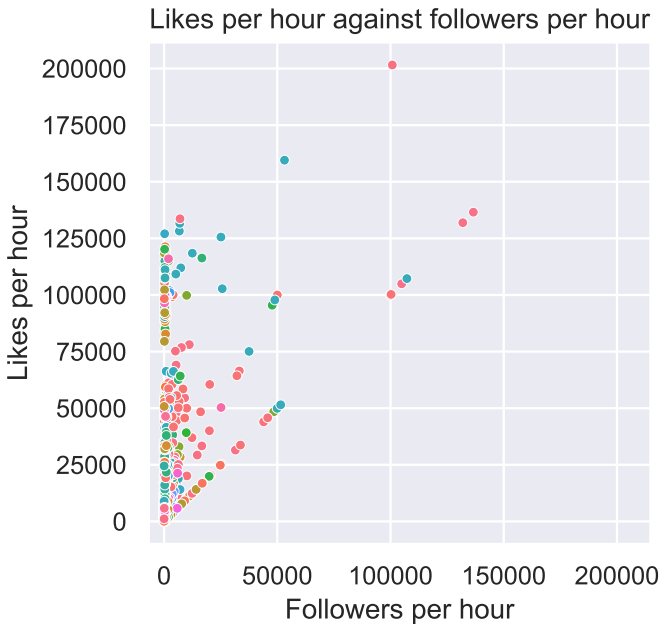

Fig. 5. Likes per hour against followers per hour. Each color shows a different creator.

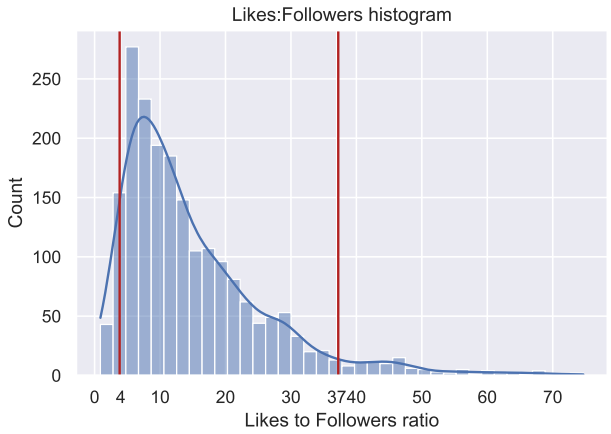

Fig. 6. Frequency of various likes-to-followers ratios for all creators. Red lines indicate the 5-th and 95-th percentile.

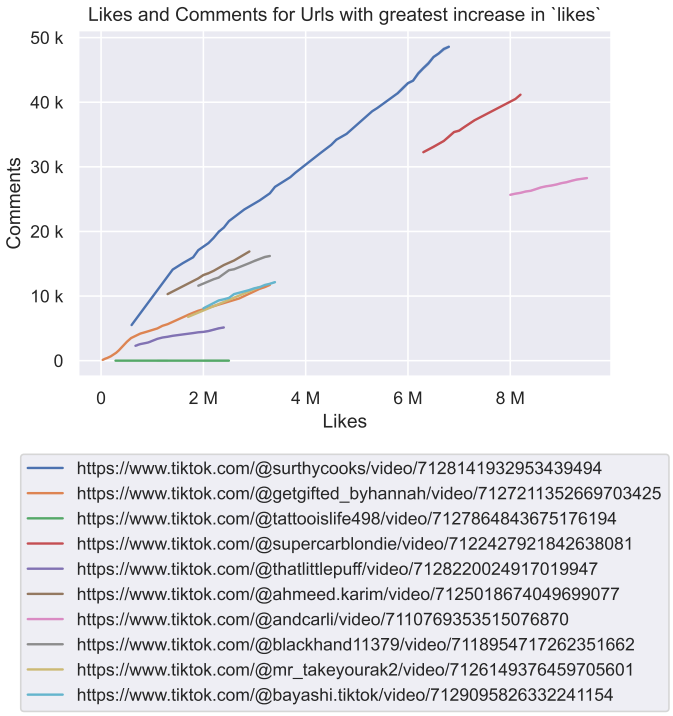

Fig. 7. Likes vs comments on TikToks with the greatest increase in likes.

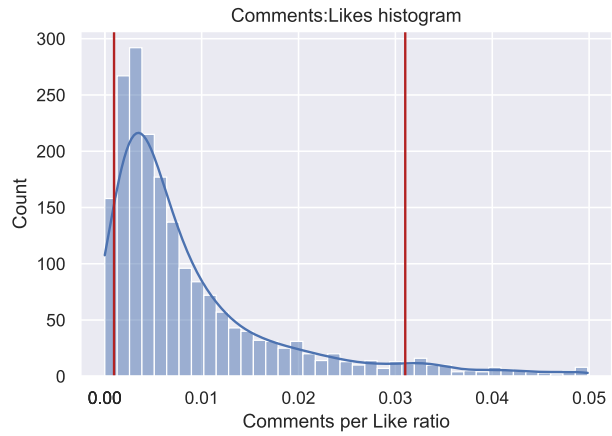

Fig. 8. Comments to likes ratio, with the red lines indicating the 5-th and 95-th percentiles.

D. Likes, Comments, Followers, and the connections between them

Figure 7 shows a line plot of the number of likes vs the number of comments on the top 10 TikToks with the greatest increase in likes. From this plot we can see that the ratio of likes to comments for a given TikTok remains approximately constant regardless of how much attention it gathers.

This likes to comments ratio can be calculated explicitly and plotted as a histogram, as shown in Figure 8. This plot shows that 95% of TikToks will have fewer than 3 comments for every 100 likes they get.

Figure 9 shows every TikTok made by every creator, plotted as a two dimensional heatmap with the comments per like against the log of the number of followers of the creator.

The double log scale makes it easier to see the full distribution of the data, although it hides how creators with more followers tended to attract

Fig. 9. Heatmap of comments per like against followers for every TikTok by every creator.

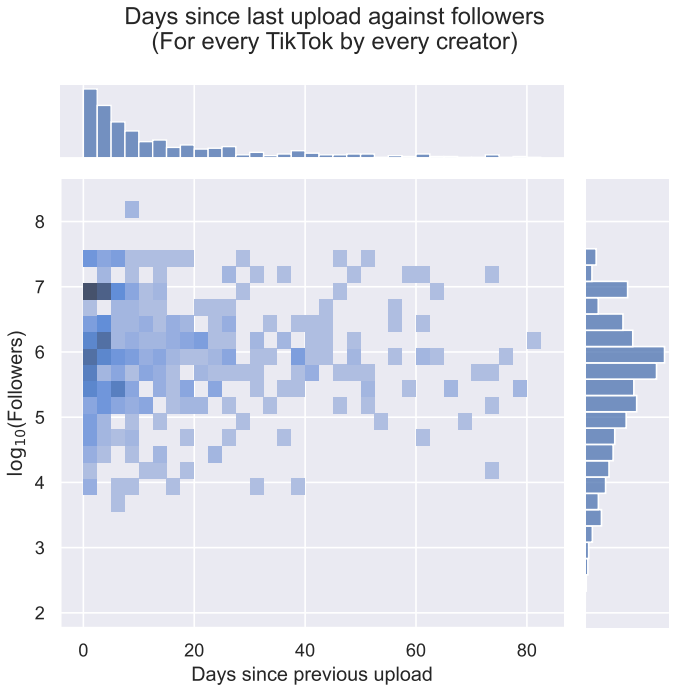

Fig. 10. Days since last upload against follower count for all creators.

fewer comments per like than creators with fewer followers, as can be seen by the slight negative slope and the larger variance in comments-per-like for TikToks with fewer than 10^5 followers.

It is possible that this negative relationship between followers and comments-per-like is due to the TikTok algorithm favouring likes over comments, but it could also be that it is easier for users to like a TikTok or follow it's creator than it is to leave a comment.

E. Does posting more often correlate with more followers?

Figure 10 shows the days since a creator last uploaded a TikTok against their follower count for every TikTok.

From this we can see that there is no strong relation between upload frequency and follower count. It appears that the majority of creators post within 10 days of posting their previous TikTok. However there is a high density of creators with just under 10^7 followers who post one video every five days.

The only definite conclusion that can be drawn about days since previous upload and follower count is that for creators who post every few days, there is a positive relationship between posting frequency and number of followers. This relationship does *not* extend to all creators, and there are creators with a high follower count who post on the order of weeks, not days.

VI. A Special Note

While working on this project, I discovered a bug in seaborn. This bug caused margins to increase when multiple rugplots were added to the same ax, even if `expand_margins` is False. To find the root cause was quite complicated, and but the bug came from how seaborn detects the correct colors to for the rugplot. In `seaborn/utils.py`, in method `_default_color`, the following line resolves to `ax.plot([], [], **kws)`:

```
scout, = method([], [], **kws)
```

By default, matplotlib has the parameters `scalex` and `scaley` of `ax.plot` set to `True`. Matplotlib would see that the rug was already on the `ax` from the previous call to `sns.rugplot`, and so it would rescale the x and y axes. This caused the content of the plot to take up less and less space, with larger and larger margins as more rugplots were added.

I have fixed this issue (see this pull request) and the change will be available in seaborn 0.12, but unfortunately was not available by the deadline of this project.

VII. References

Harris, Charles R., K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, et al. 2020. "Array Programming with NumPy." *Nature* 585 (7825): 357–62. https://doi.org/10.1038/s41586-020-2649-2.

Hunter, J. D. 2007. "Matplotlib: A 2d Graphics Environment." *Computing in Science & Engineering* 9 (3): 90–95. https://doi.org/10.1109/MCSE.2007.55.

The pandas development team. 2020. *Pandas-Dev/Pandas: Pandas* (version latest). Zenodo. https://doi.org/10.5281/zenodo.3509134.

Waskom, Michael L. 2021. "Seaborn: Statistical Data Visualization." *Journal of Open Source Software* 6 (60): 3021. https://doi.org/10.21105/joss.03021.