Division of Computing, Engineering and Mathematical Sciences
Department of Computing, University of Kent *in Canterbury*

Matplotlib
Critical Review

Written By

Lukasz R G Tomaszewski

Computer Science (Artificial Intelligence) MSc

Module CO886: Software Engineering

LaTeX Word Count: 983

April 2022

1 Introduction

The Matplotlib library [1] open source and is utilised by Python for visualising data, it sources data from an internal or an external source and plots it in the form of a graph, the user can specify the graph type, type of data chosen, what data goes on what axes, colour of the data points and more relevant features. This appeals to the data analysts, scientists and certain businesses that require graphical visualisation of masses of business data. Importable as a standard Python library, the Matplotlib library [1] is also hosted on GitHub, where over 1300 programmers contribute to the upkeep of the library dating back to 2014 when the library was first uploaded, they can freely change the code on a custom branch where they develop and test new features, debug and further experiment with the library without interrupting the current working script that millions of people are using.

2 Contribution & Modelling Tools

Though the GitHub repository has over 1300 programmers, it does openly allow the public to make a pull request to view, test and contribute to the development and debugging/ issues. It has specifics guidelines outlined here [2] for any developers wanting to assist. It tells and shows the community how to write and request through examples and via its documentation, shows bug triaging, how to write the documentation, the library policies and how to style the documentation.

The development of the documentation is mainly done by the original developer or groups there of however the community maintain it and therefore is constantly being looked at for mistakes and issues. On GitHub and the public pages, the only officially modelling tools are those of documentation and contribution criteria, though it appears standard practice that clear commit changes are the priority for the well maintained library. Furthermore the extensive documentation via its website is clear that the library serves a significant purpose to be fully functional and allow for future changes.

3 Testing, Build Tools and Integration

Via the multiple branches in the GitHub repository explored in section 4, the contributors and official developers can create, test and release new versions of the library at will, the contributors can pull a test version and make a change and push it where a developer can merge it with a final release version after testing. The development team release 2/3 approved versions each year [3], with concise release notes and detailed comments within the code scripts themselves allow for control over the types of changes happening to the library per commit and release.

The integration is that mainly of the Python open libraries, when installing Python or and IDE of Python, Matplotlib comes with is automatically which is source form the GitHub repository [1]. This means when the final version is released, it allows the entire community to have access to it at the same time and all bug fixes. This is another reason why the library is so well maintained via Python and C++, these are the main backbone languages of the Matplotlib library, while the core of the library remains intact, most features allow for faster run times, different data visualization techniques and bug fixes.

4 The Use of Git

As explained in section 1, Matplotlib utilises GitHub to host its up-to-date open source library to which through Git, it is sustained by over 1300 trusted contributors and have the millions more developing and testing new tools. With over 42,500 commits in 6,300 forks on 19 branches that carry the current, retired and testing versions of the library for the community of collaborators to pull from, 513,000 users on GitHub directly alone. One key feature in GitHub is the 'issues' tab which holds all the current and past issue the community have with the library, with multiple issues being resolved daily, it's clear that the library is not only well used but well maintained by its developers and community.

As mentioned in section 3, all changes are detailed not only of the public release notes but also within the edited code via comments and commit logs. GitHub records all of these and allows users to search through them to further understand how the library developed and what type of features and fixes have been added. This leads onto the quality of the code, the files pull and edited are pushed back onto GitHub waiting verification from the development team however the team can view, edit and change the push request before merging, however before being able to contribute, users must follow the documentation that outlines the requirements.

5 General Characteristics

The community mentioned in section 4 is big, so much so that it's governed by a code of conduct, implying that the community is interactive and the developers intend to maintain it by keep it safe and interactive for only the developers that want to be a part of the community. Even though the library is community driven, it appears that the library is funding independently from the community with grants and certain funds thought large companies as at the time of this report the development team are recruiting for a "Research Software Engineer Position via a grant from NASA." [2].

The use of GitHub allows for the community of developer and the contributors to all work together and build on this library continuously, fix bugs quickly and keep is constantly under proper maintenance. For the library the biggest advantage it has is its community and how regulated the library is, users are constantly using the library in different environments and under different conditions allowing the library to be rigorously tested and the community and development team are on hand to fix bugs/ issues quickly as well as respond to troubleshooting. The community supports the library though contributions and it appears to not need external funding to be maintained, the use the library gives users is sought so much that its community will appear to continuously update it.

6 References

[1] M. Contributors, *Matplotlib*, 2002. https://github.com/matplotlib/matplotlib.

[2] T. M. development team, *Matplotlib contributing*, 2021. https://matplotlib.org/devdocs/devel/contributing.html.

[3] ——, *Matplotlib release notes*, 2021. https://matplotlib.org/stable/users/release_notes.