animint2 Documentation and Bug Fix Project: An Application

There's been an explosion of AI tools this year. I hereby certify that I haven't used them for this application in any way, shape, or form.

Project Info

Project Title: `animint2` Documentation and Bug Fix Project

Project Short Title: `animint2` Documentation Project

URL: For the project that I'm hoping to work on? It's here.

Proposal Summary

The `animint2` Documentation and Bug Fix Project will make the `animint2` reference documentation more accessible, fix errors in the documentation, and reduce the number of bugs by at least one.

Broad Scope of the Project

My goals for this project are as follows:

1. To improve access to `animint2`'s features for scientists, students, and other users; and
2. To make `animint2` easier and more pleasant to use for scientists, students, and other users.

This will involve:

1. Improving documentation for `animint2` via a reference site accompanying the `animint2` Manual; and
2. Documenting and fixing bugs and issues in `animint2`.

`animint2` already has a number of useful features for interactive data visualization. This project will not add to those features, but it will make those features more publicly prominent and bug-free.

Contributor Biography

Hello! I am currently an unaffiliated researcher, which is a fancy way of saying that I do research without getting paid for it.[1] Until recently, I was a graduate student in psychology at the New School for Social Research, though I also have an undergraduate background in philosophy. I specialize in network psychometrics and substance use disorders. I also have a strong interest in computational psychopathology, and I'm hoping this project will let me focus on the computational side for a few months.

I'm familiar with R, though mostly as a tool for research and data analysis. I have no formal training in R and am entirely self-taught—I imagine that this kind of programming background is not uncommon among scientists. I'm sure this means that my knowledge of R has many holes, even as I've worked to close them up. For this project, though, I think it gives me a couple of advantages:

1. Many of the scientists who are likely to use `animint2` have similar programming backgrounds, which I can take into account when documenting `animint2`.
2. I'm self-sufficient and used to resolving programming problems without the assistance of other people.

Contact Information

Name: J. Chen

Postal Address: `not publicly available`.

Telephone: `not publicly available`.

Email: `not publicly available`.

Other Communication Channels: I'm fine communicating through whatever means my mentors want. I have Slack, Telegram, and Signal, but I have no problem with using another service.

Contributor Affiliation

No current affiliation. If my former affiliations matter: I worked and studied at the New School for Social Research, and I was an extern clinician at the Lower East Side Harm Reduction Center.

Schedule Conflicts

I'm not applying anywhere else, and I have no other conflicting commitments (e.g. I'm not travelling or anything). I have a weekly ASL class that I attend, which may affect my weekly video call availability.

[1] One paper's been in the works for a while and will hopefully be published this year!

Mentors

Evaluating Mentor: Toby Dylan Hocking (tdhock5@gmail.com)

Co-mentor: Faizan Khan (faizan.khan.iitbhu@gmail.com)

Have I been in contact? Yes. I first contacted them via email on either the 28th or the 29th of March, depending on the time zone.

Coding Plan and Methods

On Timing

Two of the most important questions

Google only allocates 175 hours (and 14 weeks) to each project, and I assume I'll make mistakes and run into problems that will take longer than expected to resolve. Therefore, I tried to keep each week's scope limited and realistic.

Each week

I plan on spending at least three days a week coding (which will exceed the 12.5 hours/week, but shhh.)

If things go exceedingly well and I finish up before the week is over, I'll start on tasks for next week. If things go poorly,

I'll also keep a notebook of what I've done each week.

The project is modular.

Coding Timeline

Community Bonding Period 1 (May 4 to May 7)

Community Bonding Period 2 (May 8 to May 14)

Community Bonding Period 3 (May 15 to May 21)

Community Bonding Period 4 (May 22 to May 28)

Week 1 (May 29 to June 4)

The goal this week is to reorganize files and get a draft of the reference website ready.

- 1st weekly meeting.
- Fork the `animint2` GitHub page. Begin reorganizing files in a way that `pkgdown` recognizes. `pkgdown` mostly wants a vignettes folder and a README that's a `.md` and not an `org`.
- Compile preliminary `pkgdown` website.

Week 2 (June 5 to June 11)

The goal this week is to appropriately reorganize the functions, add necessary links, and edit. Then I'll publish the website online.

- 2nd weekly meeting.
- Reorganize the `animint2` functions on the website.
  - By default, `pkgdown` organizes functions alphabetically, but it's possible to manually specify the order.
  - My initial thought is to break up the functions by exclusivity—`animint2`-exclusive functions versus those also available in `ggplot2`.
- Ensure that the reference website has appropriate links to the `animint2` Manual, as well as vice versa.

Week 3 (June 12 to June 18)

The goal this week is to begin diagnosing and clearing up errors in both the PDF and HTML documentation.

- 3rd weekly meeting.
- There are a number of functions in the `animint2` reference that just don't work. Test all the functions present in `animint2` and examine their code. Compile a list of functions that either just output a message or exhibit buggy behavior.
  - For example, `geom_bin2d` is present, but when actually using it, `animint2` outputs: "`bin2d` is not supported in `animint`. Try using `geom_tile()` and binning the data yourself." This is confusing for the user, since `geom_bin2d` is listed as one of `animint2`'s functions.
  - Discuss this with my mentors. The output is intentional, so is there a reason `geom_bin2d` (and the like) are present?

Week 4 (June 19 to June 25)

The goal this week is to continue diagnosing and clearing up errors in the documentation.

- 4th weekly meeting.
- If necessary, continue compiling the list of problematic functions.
- Either remove those functions from documentation, note (in the documentation) why they function they way they do, or mark them as bugs to be placed in GitHub Issues. The goal is not to fix them—I wouldn't have time for that.

Week 5 (June 26 to July 2)

The goal this week is to document bugs on GitHub Issues, which will make it easier to track and maintain.

- 5th weekly meeting.
- If I haven't done so already, add all the bugs noted in the earlier weeks and enter them as GitHub Issues.
- Add all observed bugs from my website to `animint2`'s issues.
- Close or add bugs from the `animint` repository to `animint2`'s. I'll examine them to see if they're relevant anymore—at least one is.

Week 6 (July 3 to July 9)

The goal this week is to diagnose and repair Chapter 7 of the `animint` Manual, which scrolls up by itself.

- 6th weekly meeting.
- Figure out why Chapter 7 exhibits the auto-scrolling bug. (My guess is that it has something to do with the JavaScript.)
- Repair the auto-scrolling error. Submit a pull request to the `animint2-book` repository.

Week 7 (July 10 to July 16)

It looks like Google designates this a non-coding period. Midterm evaluations are due.

- 7th weekly meeting.
- Finish midterm evaluations this by the 14th.

Week 8 (July 17 to July 23)

The goal this week is to add a vignette about common errors and workarounds when using `animint2`, as well a vignette about any major bugs to be aware of.

- 8th weekly meeting.
- Compile a list of those errors and workarounds (e.g. "in lieu of `geom_col()`, use `geom_bar(stat = "identity")`"). Write a vignette addressing them and then add that to `vignette/`.
- Compile a list of major bugs (e.g. "it's not possible to layout multiple charts"). Write a vignette listing them, suggest workarounds, and then add that to `vignette/`.

Week 9 (July 24 to July 30)

The goal this week is to catch up on any missing work. This concludes the larger documentation aspect of the project.

- 9th weekly meeting.

- Catch up on work, clean up messy code, edit documentation, and the like.

Week 10 (July 31 to August 6)

The goal this week is to figure out why using `showSelect` in interactive scatterplots throws up an erroneous error, and then to fix it.

- 10th weekly meeting.
- `animint()` incorrectly informs me that "`showSelected` only works with `position=identity`, problem: geom1_point_foobar."
- Convert `showSelect` into a failing test, then repair the function so that the test passes.

Week 11 (August 7 to August 13)

The goal this week is to figure out why `animint2` fails to correctly lay out multiple graphs. I think this problem is likely more complex than some other bugs, so I'll give myself two weeks to figure it out.

- 11th weekly meeting.
- Figure out why the graphs aren't laid out correctly.
  - In my experience, `animint2` doesn't throw up any error messages—it just fails to correctly output the graphs.
  - I wonder if the problem is not just with `animint2` but with how it interacts with other packages (or with Markdown, HTML, or LaTeX).
  - `ggplot2` outputs multiple graphs correctly, but the problem was present when `animint2` depended on `ggplot2`—before `animint2` was a fork. It follows that there must be a problem with `animint2`-exclusive code, and not with `ggplot2`'s. Maybe that's a place to start.

Week 12 (August 14 to August 20)

Having diagnosed the problem, the goal this week is to fix the problem with multiple graph layouts.

- 12th weekly meeting.
- Convert `animint2` into a failing test, then repair the package so that the test passes.
  - Without knowing why the package fails to lay out the graphs, it's hard to explain what exactly I'll be doing.
  - Another trouble here is that I can't just isolate a function to test.
  - Toby Dylan Hocking suggests borrowing code from `flexdashboard`.

Week 13 (August 21 to August 28)

The goal this week is to catch up on unfinished work and submit mentor evaluations. This concludes the smaller bug fix aspect of the project and concludes the project altogether.

- 13th weekly meeting.
- Submit final mentor evaluation on the 28th.
- Catch up on work, clean up messy code, edit documentation, and the like.

Management of Coding Project

Commitments will have

Tests

See the `animint2` examination website for the tests that I took. Thanks to Toby Dylan Hocking for advice regarding changes to the easy test.

Anything Else?

Have a great day! :>