Packages, Version Control, & Testing

2023-10-03

Creating an R package

For this portion of the portfolio, I have been working to improve an R package I have been working on for the last few years called `hull2spatial`. The package can be found in this GitHub repository: https://github.com/babichmorrowc/hull2spatial. The purpose of the package is to convert objects produced by the `alphahull` R package into objects that are compatible with the `sp` package. The `alphahull` package creates 2 types of shapes around a set of points: α-shapes and α-hulls. Both types of shapes depend on the parameter α: when α is high, the resulting shape is closer to a convex hull, while when α is low, the resulting shape is more concave. α-shapes are composed of straight lines, while α-hulls can include arcs. See this blog post for more detail on the motivation behind this package.

Adding a license

I hadn't previously added a license to `hull2spatial`, so I used the `usethis` package to add one. After doing some reading about license selection here, I selected the GPLv3license, which is a copyleft license that stipulates that everything using my code must remain open source. The `usethis` package has a built-in function `use_gpl_license()` that will automatically set up the appropriate licensing when run within the project containing the package.

Writing tests

I also wanted to add some tests to my package using the `testthat` package. The `usethis` package automatically creates testing files in the proper format. By running `usethis::use_test()` when the relevant file of functions is open, it automatically created the test file using the naming convention `test-<r-file-name>.R` and put it inside the `tests/testthat` folder. In the case of my package, the test files can all be found here. By going to Build > Test Package within RStudio, I can run all of the tests I have written and find out how many tests have passed. This is equivalent to running `devtools::test()` in the console.

I also set up GitHub Actions testing by running `usethis::use_github_action_check_standard()`, as mentioned in the course website here. I did receive the following warning message:

```
Warning message:
`use_github_action_check_standard()` was deprecated in usethis 2.2.0.
  Please use the `check-standard` argument of `use_github_action()` instead.
```

which indicates that as of the most recent version of the `usethis` package, it is best to run `use_github_action("check-standard")` instead. I made a small pull request to update this in the course website.

I noticed that the GitHub Actions workflows were failing, so I worked on a series of fixes, mostly having to do with the need to explicitly note the package from which a given function was taken, e.g. `alphahull::ahull`. I also found that by running `devtools::check_rhub()` in the RStudio Console, I could run CRAN checks on the package before pushing my changes (which is faster than waiting for the entire GitHub Actions checks to finish after pushing).

Package improvements

I had an Australian scientist named Fonti Kar reach out to me about how we might modify my R package to be used to convert alpha hulls for IUCN range assessments. The developer of the `alphahull` package, Beatriz Pateiro, has been working on a new feature to create alpha-hull objects following the IUCN Red List method of creation.

She can add documentation / tests / GitHub actions / landing page We could do a Hex sticker I will work on the maths What's the other package that does this?