

ETC5523 Communicating Wit...



Assignment 4: Turning analysis interactive

AUTHOR

Michael Lydeamore

PUBLISHED

16 October 2024

Objectives

Write an R package including:

- Data that you have analysed
- A shiny app to explore data
- Full documentation
- [roxygen2](#) website

Important

This is an **individual assignment**.

Output format

- Published website, using [pkgdown](#)
- An installable R package on GitHub

Instructions

In this assessment, you will be turning the analysis you have used for Assignments 1, 2 and 3 into an R package, including a Shiny app and a [roxygen2](#)-created website.

Important

You must join the GitHub classroom here: <https://classroom.github.com/a/VW9MaxFW>. If you cannot find your name in the list of students, please contact the lecturer directly before accepting the assignment.

Part A: The R Package (15 marks)

The template provided in the GitHub classroom is *blank*. You will need to set up the relevant structure for a package either manually or using the [usethis](#) package. The final repository must have:

- A README file with relevant information (2 mark)
- A license (1 mark)
- An installable R package using the [remotes](#) package (1 mark)

The R package must have:

- A shiny app inside the `inst` folder (1 mark)
- A function to launch the shiny app that is exported (1 mark)
- The data used by the shiny app, including any cleaning, contained in `data-raw` (1 mark)
- At least one vignette (3 marks)
- Completed DESCRIPTION file, including your name, package name and short description (2 marks)

Including extra functions that are useful to the package (and reduce repetition) may be granted extra marks, although they are not necessary.

All functions and data objects must be documented clearly using `roxygen2` (3 marks)

Part B: The Shiny App (11 marks)

You need to create a shiny app to help users explore the dataset you have been working on, and find similar conclusions to your analysis. The shiny app must:

- Use the dataset from your package (don't use `read.csv` inside the app) (1 mark)
- Present *some level* of interactivity to the user using at least one selector/button/input field (2 marks)
- Use these interactivities to change the displayed output(s) (2 marks)
- Include a description of what the fields mean (2 marks)
- Include a description of how to interpret the outputs (2 marks)
- Have some styling beyond the default white background / grey well panel (2 marks)

Remember, don't just include interactivity for the sake of it. It needs to *add* something to the message the users could otherwise receive from a static picture. The best interactive plots help users explore, and reach a guided conclusion.

Tip

You are free to use whatever shiny templates you like (such as `shinydashboard`), or just the default. Note that simply including `shinydashboard` will not get you full styling marks!

Part C: The `pkgdown` site (6 marks)

You should have a `pkgdown` site accessible from a public URL (1 mark).

- Link to the `pkgdown` site present in the README file (1 mark)
- Working function documentation (2 marks)
- A descriptive homepage for the `pkgdown` site (1 mark)
- Vignettes showing on the web (1 mark)

Overall grade: $15 + 11 + 6 = 32$ marks