Creating a data package

Nicholas Horton (nhorton@amherst.edu)

November 9, 2020

Data package

The goals of this file is to describe how to create a data package using the data from the DiscGolf example I created.

1. Please ensure that you have the files in the DiscGolf_files folder.

2. Confirm that you've successfully installed a personal access token by running:

```
usethis::git_sitrep()
```

This should display information about your Personal access token value (if then you need to follow the instructions to add this to your `.REnviron` file).

3. Create a new project ("new package with devtools") called `DiscGolf`.

4. Run `usethis::use_git()` to set up Git connectiona and add other files. Be sure to confirm that you are seeing both a "Build" tab and a "Git" tab. If no "Git" tab appears please open "Tools" / "Project Options" / "Version Control" and ensure that "Git" is selected.

5. Edit the `DESCRIPTION` file with your information (see the DESCRIPTION file I've provided for an example) and save the file. Here are some lines that need to be changed (please keep the other lines).

```
Title: Flying objects officially approved for use in competition
Version: 0.1.0
Authors@R:
    person(given = "Nicholas",
           family = "Horton",
           role = c("aut", "cre"),
           email = "nhorton@amherst.edu",
           comment = c(ORCID = "0000-0003-3332-4311"))
Description: Everything you might want to know about Disc Golf flying objects.
```

6. Configure a license. Run:

```
options(usethis.full_name = "YOUR NAME")
```

```
usethis::use_mit_license()
```

7. Run `usethis::use_data_raw()` to create the `data-raw` folder.

8. Open the data-raw folder and DATASET.R then replace with contents of wrangling commands from the DATASET.R file I've provided. Save this modified file.

```
library(tidyverse)
DiscGolf <-
  readr::read_csv("pdga-approved-disc-golf-discs_2020-10-04T15-02-04.csv") %>%
  janitor::clean_names() %>%
  mutate(
    model = iconv(disc_model, "latin1", "ASCII", sub = ""),
```

```
    approved_date = lubridate::mdy(approved_date)
  ) %>%
  rename(
    diameter = diameter_cm,
    flexibility = flexibility_kg,
    height = height_cm,
    manufacturer = manufacturer_distributor,
    rim_depth = rim_depth_cm,
    rim_thickness = rim_thickness_cm,
    weight = max_weight_gr
  ) %>%
  select(
    approved_date, class, diameter, flexibility, height,
    manufacturer, model, rim_depth, rim_thickness, weight
  )
usethis::use_data(DiscGolf)
```

9. Use the "Session" menu to change "working directory" to "source file location", then run "source" on the DATASET.R file. This should run the commands in that file and add the newly created native R file into the package.

10. Check that the data folder now contains the data DiskGolf.rda.

11. Copy the DiscGolf.R file (documentation that I've provided) to the R folder

```
#' Discs for Disc Golf
#'
#' This package contains specifications for discs approved by
#' the Professional Disc Golf Association
#' @docType package
#' @name DiscGolf
#' @aliases DiscGolf DiscGolf-package
NULL


#' "Golf Discs"
#'
#' A data set containing golf discs approved by
#' the Professional Disc Golf Association
#' as of October, 2020
#'
#' @source \url{https://www.pdga.com/introduction}
#' @format A dataframe with 1260 elements
#' \describe{
#'   \item{approved_date}{Data the disc was approved}
#'   \item{class}{Class of disc}
#'   \item{diameter}{Diameter of disc (in cm)}
#'   \item{flexibility}{Flexibility of disc (in kg)}
#'   \item{height}{Height of disc (in cm)}
#'   \item{manufacturer}{Manufacturer/distributor}
#'   \item{model}{Disc model}
#'   \item{rim_depth}{Rim depth (in cm)}
#'   \item{rim_thickness}{Rim thickness (in cm)}
#'   \item{weight}{Weight (in gm)}
#' }
"DiscGolf"
```

12. Commit all of these files (see the "Git" tab).

13. Run `usethis::use_github()` to connect this project to GitHub (I used `https` option).

14. Check that the package passes a comprehensive set of tests by clicking on "check package" in the "Build" tab.

15. Click on the "install and restart" button in the "Build" tab to install the package.

16. Type "DiscGolf" at the console to confirm that things are working (you should see lots of discs!.

17. Run `usethis::use_readme_rmd()` to create a `README.Rmd` file. Edit this template file to describe the package.

18. Knit the file to create `README.md`.

19. Add `README.Rmd` and `README.md` to GitHub: commit and push.

20. Add a vignette stub: `usethis::use_vignette("DiscGolf")`. Edit the vignette and commit and push the files to GitHub.

21. Add a GitHub action to check the package: `usethis::use_github_action("check-standard")`. Add the .github files, commit, and push.

22. You can add a badge (click on the "action" tab from the GitHub web location of your package, then click on the specific action to display a settings ("...") box. Click this to find the code to create a status badge. This code should be added to `README.Rmd` near the top. Be sure to knit and then commit and push the updated `README.md` file.

23. Check that you can also install the package using `remotes::install_github("YOURGITHUBNAME/DiscGolf")`.

24. Show off the package! Open up the Settings tab on GitHub and make sure that the repository is public.

NEXT STEPS: Next week, you'll be doing the same with your own data package in a new repo that you'll create in your personal GitHub organization. You can and should start to pull the parts and pieces together that you will need.

Other resources

This examples builds on material that Kelly McConville (Reed College) first described to me. Helpful yet somewhat terse and dated examples (these technologies are changing very quickly!) can be found at https://www.erikhoward.net/blog/how-to-create-an-r-data-package and http://www.davekleinschmidt.com /r-packages.