Structuring data analysis projects as R packages

Francisco Rodríguez-Sánchez

@frod_san

BEING ORGANIZED

is for people who are too lazy to look for their stuff.

Good & consistent project organisation

Facilitates

- Reproducibility
- Understanding by reviewers and collaborators (including yourself in a few months)
- Tool building and sharing

Rules for good project organisation

- All files in same directory

Noble 2009, Rodriguez-Sanchez et al 2016, Wilson et al 2017

Rules for good project organisation

- All files in same directory
- Raw data kept in separate folder

Noble 2009, Rodriguez-Sanchez et al 2016, Wilson et al 2017

Rules for good project organisation

- All files in same directory
- Raw data kept in separate folder
- Clean data derived through scripts

Noble 2009, Rodriguez-Sanchez et al 2016, Wilson et al 2017

Rules for good project organisation

- All files in same directory
- Raw data kept in separate folder
- Clean data derived through scripts
- Functions independent from analysis scripts

Noble 2009, Rodriguez-Sanchez et al 2016, Wilson et al 2017

Rules for good project organisation

- All files in same directory
- Raw data kept in separate folder
- Clean data derived through scripts
- Functions independent from analysis scripts
- Functions well documented & tested

Noble 2009, Rodriguez-Sanchez et al 2016, Wilson et al 2017

Rules for good project organisation

- All files in same directory
- Raw data kept in separate folder
- Clean data derived through scripts
- Functions independent from analysis scripts
- Functions well documented & tested
- Output disposable & separate from code

Noble 2009, Rodriguez-Sanchez et al 2016, Wilson et al 2017

Rules for good project organisation

- All files in same directory
- Raw data kept in separate folder
- Clean data derived through scripts
- Functions independent from analysis scripts
- Functions well documented & tested
- Output disposable & separate from code
- Makefile runs analyses in appropriate order

Noble 2009, Rodriguez-Sanchez et al 2016, Wilson et al 2017

Rules for good project organisation

- All files in same directory
- Raw data kept in separate folder
- Clean data derived through scripts
- Functions independent from analysis scripts
- Functions well documented & tested
- Output disposable & separate from code
- Makefile runs analyses in appropriate order
- README file with overall project description

Noble 2009, Rodriguez-Sanchez et al 2016, Wilson et al 2017

Rules for good project organisation

- All files in same directory
- Raw data kept in separate folder
- Clean data derived through scripts
- Functions independent from analysis scripts
- Functions well documented & tested
- Output disposable & separate from code
- Makefile runs analyses in appropriate order
- README file with overall project description
- Software dependencies under control

Noble 2009, Rodriguez-Sanchez et al 2016, Wilson et al 2017

R package structure

```
- data-raw/    # Original raw data
- data/        # Clean data (produced w/ script)
- R/           # Functions
- man/         # Function documentation (w/ Roxygen)
- tests/       # Tests (functions, Rmd)
- vignettes/   # Analyses, manuscript, reports (Rmd)
- makefile     # Master script that executes everything
- DESCRIPTION  # Metadata and dependencies
- README       # General info about the project
```

R packages can serve as **research compendia** (including code, data and outputs) for reproducible data analysis projects

An example

https://github.com/Pakillo/Carex.bipolar

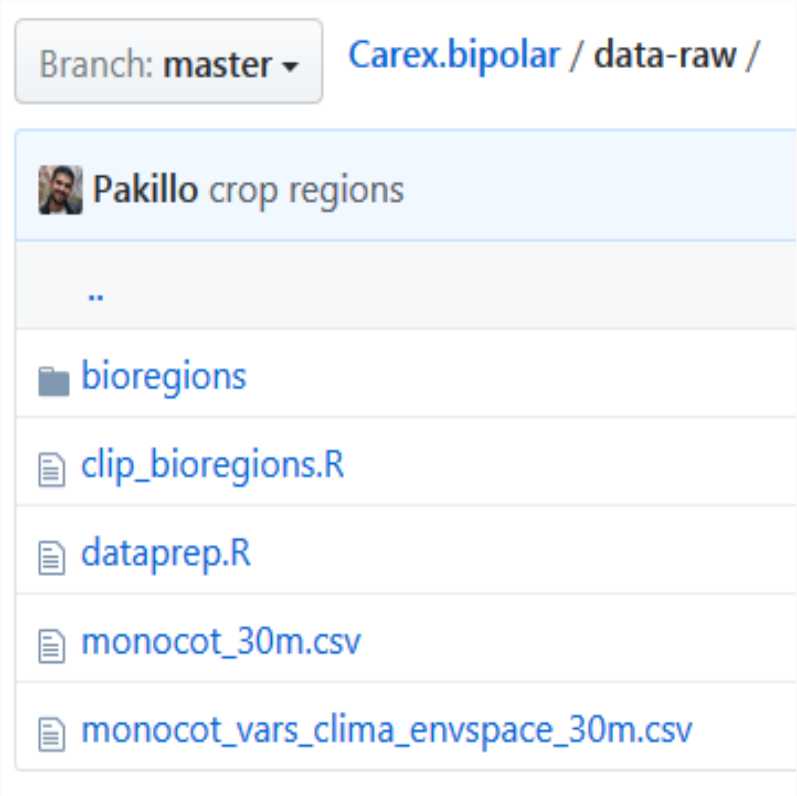
DOI 5281/zenodo.8967

build passing

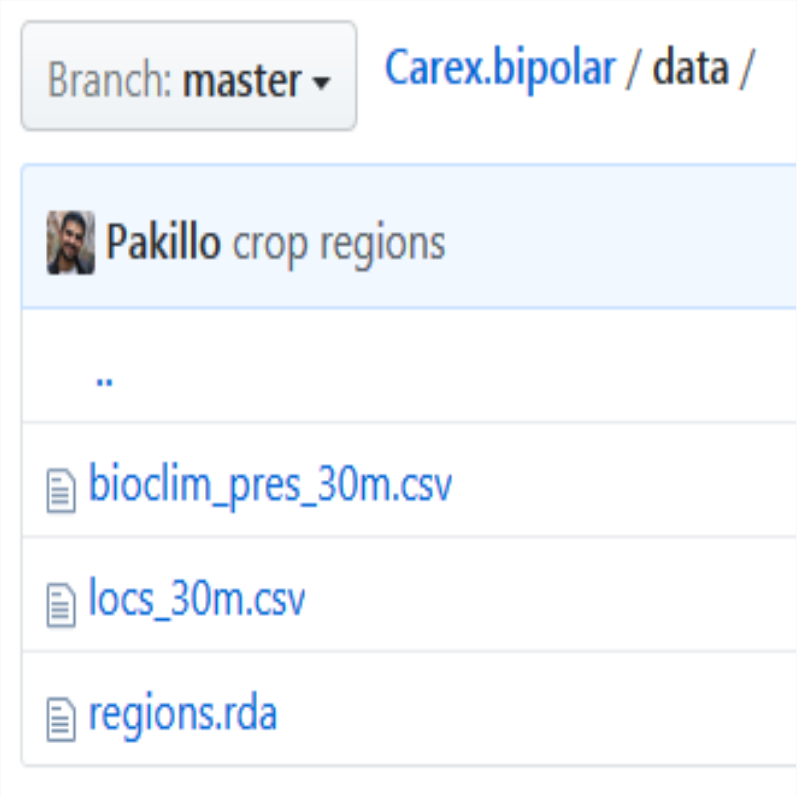
Research compendium (code and data) used for the species distribution modelling analyses in the following journal publication:

Villaverde T, González-Moreno P, Rodríguez-Sánchez F & Escudero M. (2017) Niche shifts after long-distance dispersal events in bipolar sedges (*Carex*, Cyperaceae). *American Journal of Botany*, in press.

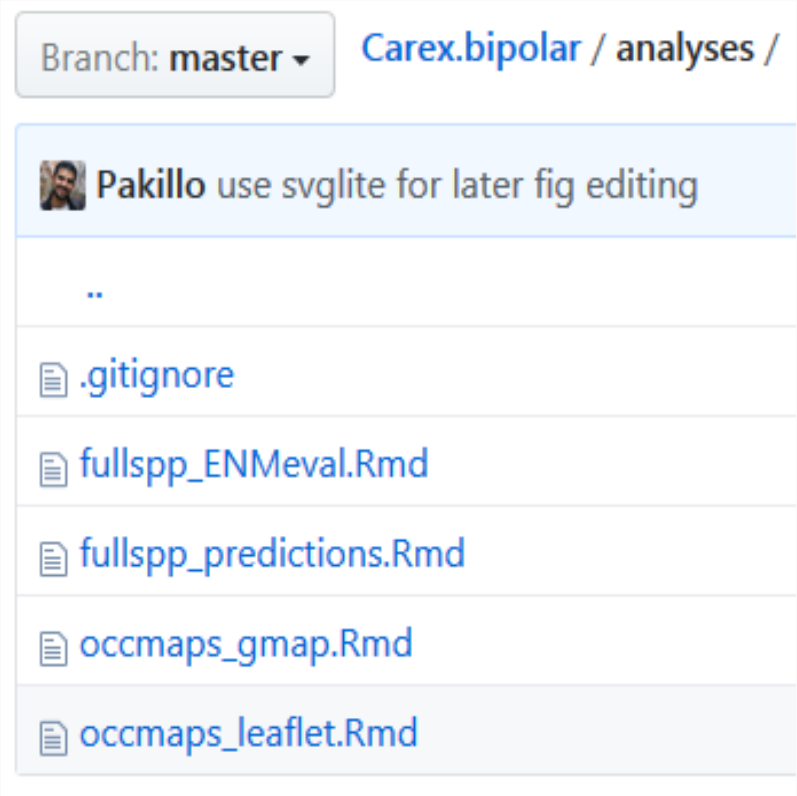
Raw data and munging scripts in `data-raw` folder



Clean data go to `data` folder



Rmarkdown documents in `analysis` or `vignettes` folder



Functions documented with Roxygen in R folder

```r
#' Combine future predictions from a Maxent model
#'
#' @param model A maxent model, as created by dismo.
#' @param scenario Character. Either "rcp45" or "rcp85".
#'
#' @return A rasterstack.
#' @export
#' @import raster
#' @import dismo
combine_pred <- function(model, scenario) {

  ## load future climate

  if (scenario == "rcp45") {
    ccsm <- read_futclim("ccsm4_rcp45_bio_2050")
    gfdl <- read_futclim("gfdl-cm3_rcp45_bio_2050")
    giss <- read_futclim("giss-e2_rcp45_bio_2050")
    hadgem <- read_futclim("hadgem2-es_rcp45_bio_2050")
    miroc <- read_futclim("miroc5_rcp45_bio_2050")
  }
```

Function documentation created automatically (`man` folder)

combine_pred {Carex.bipolar}

Combine future predictions from a Maxent model

Description

Combine future predictions from a Maxent model

Usage

```
combine_pred(model, scenario)
```

Arguments

`model`	A maxent model, as created by dismo.
`scenario`	Character. Either "rcp45" or "rcp85".

Value

A rasterstack.

Makefile runs analyses in right order

```
#### Fig 1: occurrence map ####
render("manuscript/figures/Fig1_occmap.Rmd")

#### Fig Present Suitability ####
render("manuscript/figures/Fig_suitability_present_code.Rmd")

#### Figs Future Suitability ####
render("manuscript/figures/Fig_suitability_2050_code.Rmd")
```

Advantages

Advantages of R package structure

- Reproducibility

Advantages of R package structure

- Reproducibility
- Consistent, standard, streamlined organisation

Advantages of R package structure

- Reproducibility
- Consistent, standard, streamlined organisation
- Promotes modular, well-documented and tested code

Advantages of R package structure

- Reproducibility
- Consistent, standard, streamlined organisation
- Promotes modular, well-documented and tested code
- Easy to share (zip, GitHub repo)

Advantages of R package structure

- Reproducibility
- Consistent, standard, streamlined organisation
- Promotes modular, well-documented and tested code
- Easy to share (zip, GitHub repo)
- Easy to install & run (Dependencies)

Advantages of R package structure

- Reproducibility
- Consistent, standard, streamlined organisation
- Promotes modular, well-documented and tested code
- Easy to share (zip, GitHub repo)
- Easy to install & run (Dependencies)
- Use R package development machinery:

Advantages of R package structure

- Reproducibility
- Consistent, standard, streamlined organisation
- Promotes modular, well-documented and tested code
- Easy to share (zip, GitHub repo)
- Easy to install & run (Dependencies)
- Use R package development machinery:
- R CMD CHECK

Advantages of R package structure

- Reproducibility
- Consistent, standard, streamlined organisation
- Promotes modular, well-documented and tested code
- Easy to share (zip, GitHub repo)
- Easy to install & run (Dependencies)
- Use R package development machinery:
- R CMD CHECK
- Continuous integration (Travis-CI)

Automatic checks with every commit

Travis CI About Us Blog Status Help Sign in with GitHub

Help make Open Source a better place and start building better software today!

Pakillo / Carex.bipolar

build passing

Current Branches Build History Pull Requests More options

✓ master Pakillo	add two more articles to pkgdown	#7 passed 1c006ff	3 min 22 sec a day ago
✓ master Pakillo	added leaflet occurrence maps to appear as a	#6 passed 57f5374	5 min 23 sec a day ago
✓ master Pakillo	build site with pkgdown	#5 passed 6108a7a	17 min 35 sec a day ago
✗ master Pakillo	still trying to fix error with sf in travis (via rnat	#4 failed 2c922d4	16 min 58 sec 2 days ago
! master Pakillo	adding more sf dependencies to travis	#3 errored 5a60b49	13 min 59 sec 2 days ago
! master Pakillo	trying to fix error with rgdal on travis	#2 errored 076af29	14 min 15 sec 2 days ago
! master Pakillo	add travis	#1 errored 4bce6e8	18 min 54 sec 3 days ago

Advantages of R package structure

- Reproducibility
- Consistent, standard, streamlined organisation
- Promotes modular, well-documented and tested code
- Easy to share (zip, GitHub repo)
- Easy to install & run (dependencies)
- Use R package development machinery:
- R CMD CHECK
- Continuous integration (Travis-CI)
- Automatic code review with goodpractice

Advantages of R package structure

- Reproducibility
- Consistent, standard, streamlined organisation
- Promotes modular, well-documented and tested code
- Easy to share (zip, GitHub repo)
- Easy to install & run (dependencies)
- Use R package development machinery:
- R CMD CHECK
- Continuous integration (Travis-CI)
- Automatic code review with goodpractice
- Easily create project websites with pkgdown

Project websites with pkgdown

https://pakillo.github.io/Carex.bipolar/

Carex.bipolar | Reference | Articles

build passing

Research compendium (code and data) used for the species distribution modelling analyses in the following journal publication:

Villaverde T, González-Moreno P, Rodríguez-Sánchez F & Escudero M. (2017) Niche shifts after long-distance dispersal events in bipolar sedges (*Carex*, Cyperaceae). *American Journal of Botany*, in press.

Compendium DOI: 10.5281/zenodo.896787

CITATION: Francisco Rodriguez-Sanchez (2017) Research compendium for "Niche shifts after long-distance dispersal events in bipolar sedges (Carex, Cyperaceae)" (Version 0.1.0). Zenodo. http://doi.org/10.5281/zenodo.896787

Installation

In order to run the analyses you will need to install the package first:

```
devtools::install_github("Pakillo/Carex.bipolar")
```

Usage

There is a `makefile.R` that runs each analysis in the appropriate order.

A more user-friendly version of the functions and analyses can be browsed at https://pakillo.github.io/Carex.bipolar/index.html.

Links

Browse source code at https://github.com/Pakillo/Carex.bipolar

Report a bug at https://github.com/Pakillo/Carex.bipolar/issues

License

MIT + file LICENSE

Developers

Francisco Rodriguez-Sanchez
Author, maintainer

Dev status

DOI 5281/zenodo.896

All functions explained, browsable

Carex.bipolar | Reference | Articles

Reference version 0.1.1

All functions

`Carex.bipolar`	Carex.bipolar.
`combine_pred`	Combine future predictions from a Maxent model
`compare_suitab_futu_pres`	Compare future vs present suitability per species
`crop_bioregions`	Crop raster by bioregions
`ensemble_mean`	Aggregate future predictions: Calculate ensemble mean
`ensemble_sd`	Aggregate future predictions: Calculate ensemble standard deviation (SD)
`plot_ensemble`	Plot ensemble mean or standard deviation
`plot6maps`	Plotting function: six maps
`read_futclim`	Read future climate layers
`read_pres_suitab`	Read present suitability raster for all species
`read_presclim`	Load present climate
`regions`	Andes and Nearctic biogeographic regions.

Rendered analysis reports

Carex.bipolar | Reference | Articles

Choosing best model with ENMeval

Francisco Rodriguez-Sanchez

2017-11-13

```
library(Carex.bipolar)
library(readr)
library(dplyr)
library(ENMeval)
library(rSDM)
```

Load data

```
locs <- as.data.frame(read_csv(file.path(root, "data/locs_30m.csv")))
bioclim <- as.data.frame(read_csv(file.path(root, "data/bioclim_pres_30m.csv")))
```

Select only occurrences of this species (defined in makefile):

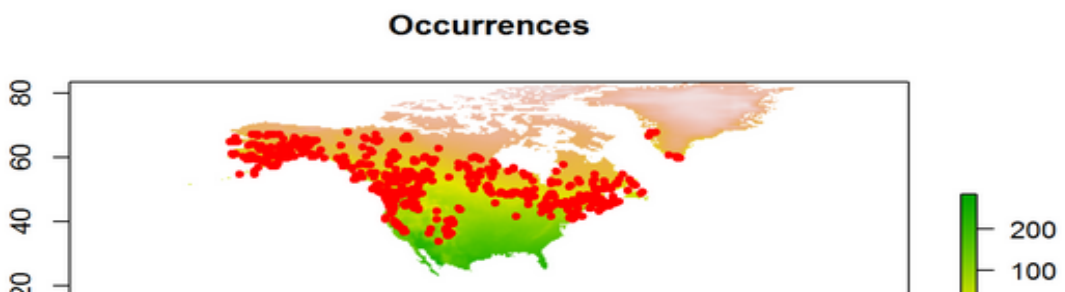
```
species <- "canescens"
if (species != "allspp") locs <- locs[locs$species == species, ]
```

Load present bioclim rasters

```
bioclim.pres <- read_presclim()
```

Map

```
rSDM::occmap(locs, ras = bioclim.pres[[1]], main = "Occurrences")  # Map
```



How

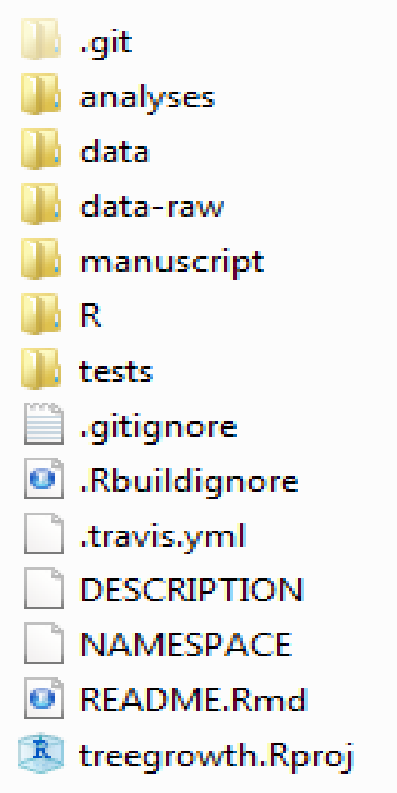
Several tools to create projects as packages

- rrtools
- workflowr
- C. Boettiger's template
- my template
- manuscriptPackage
- pRojects
- Rstudio Project Templates
- devtools, mason, prodigenr, etc

As easy as...

```
library(template)

new_project("treegrowth", github = TRUE,
  private.repo = FALSE, travis = TRUE)
```



Archive and be cited

Archive in permanent repository (Zenodo, Figshare),

get DOI and be cited

To read more

- Marwick B, Boettiger C, Mullen L. (2017) Packaging data analytical work reproducibly using R (and friends) *PeerJ Preprints* 5:e3192v1 https://doi.org/10.7287/peerj.preprints.3192v1
- Use of an R package to facilitate reproducible research https://github.com/ropensci/rrrpkg
- Rodríguez-Sánchez F, Pérez-Luque AJ, Bartomeus I, Varela S. (2016) Reproducible science: what, why, how. *Ecosistemas* 25(2): 83-92. https://doi.org/10.7818/ECOS.2016.25-2.11

End

Slides and materials available at
https://github.com/Pakillo/template