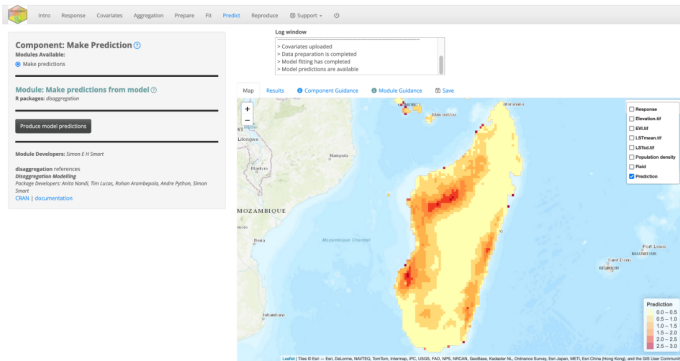


**Tim C.D. Lucas, Simon Smart**

 @StatsForBios.bsky.social

 @timcdlucas @statsforbios

 tim.lucas@le.ac.uk



shinyscholar (Simon Smart)

[www.github.com/simon-smart88/shinyscholar](https://www.github.com/simon-smart88/shinyscholar)

## Outline

The problem

The solution {shinyscholar}

Quick guide

NOT a full workshop tutorial

## The problem:

1000s of lines in one script.

No unit tests.

Unreproducible.

Hard to maintain and extend.

Not modular.

## Visualize Model Results

### Modules Available:

- ☐ BIOCLIM Envelope Plots  
☒ Map Prediction

## Module: Map Prediction

dismo : Species Distribution Modeling

### Set threshold

No threshold

Plot

### Select download file type

GRD

### Download displayed raster

Download

Module Developers: Jamie M. Kass, Gonzalo E. Pinilla-Buitrago, Bruno Vilela, Robert P. Anderson

dismo references

> Environmental data masked.  
> Random background points sampled (n = 302).  
> Occurrences partitioned by random k-fold (k = 2).  
> BIOCLIM ran successfully and output evaluation results.  
> BIOCLIM model prediction plotted.

### Change Base Map

ESRI Topo

### Current Env Variable

bio01

### Current Model

bioclim

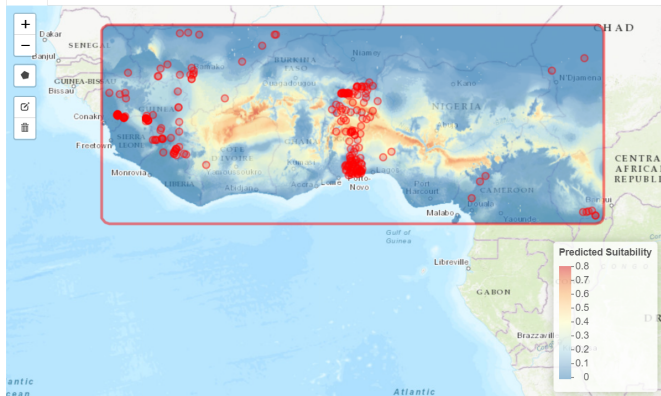
Map

Occs Tbl

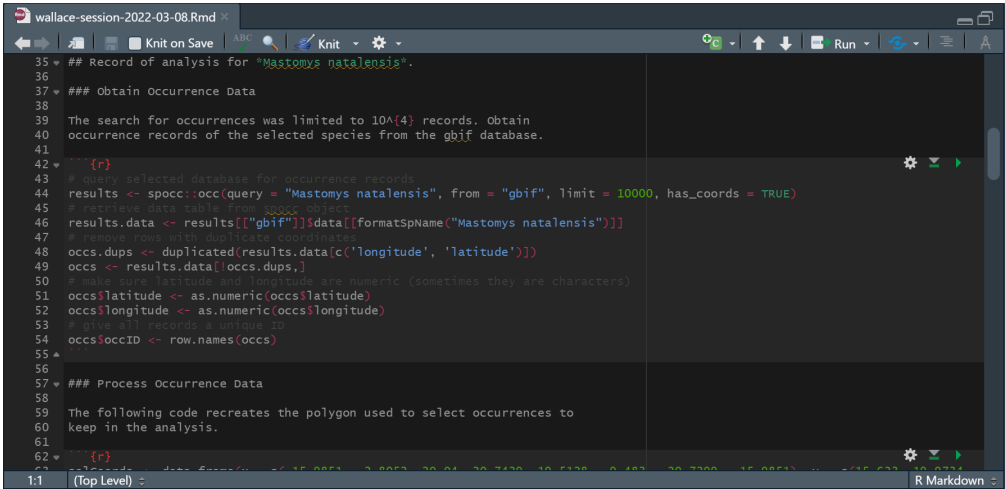
Results

Component Guidance

Module Guidance



# {Wallace}: reproducibility

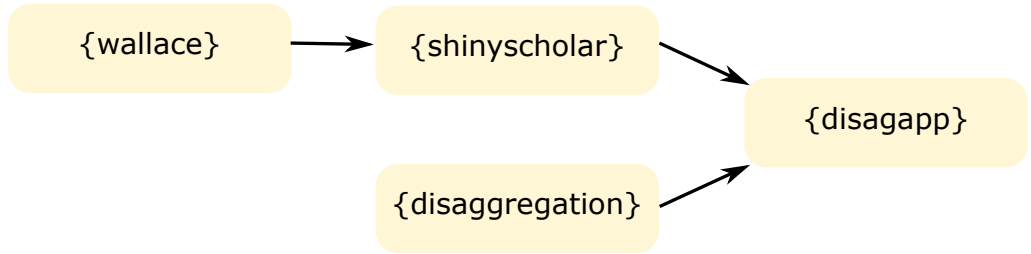


The screenshot shows an RStudio window titled "wallace-session-2022-03-08.Rmd". The interface includes a toolbar with icons for navigation, execution, and settings. The main editor displays R code with line numbers 35 through 62. The code is organized into sections with comments. The first section, starting at line 35, is a comment about the analysis for *Mastomys natalensis*. The second section, starting at line 37, is titled "Obtain Occurrence Data" and contains a comment about the search limit. The third section, starting at line 42, is a code block for querying the database, retrieving data, removing duplicates, and ensuring numeric coordinates. The fourth section, starting at line 57, is titled "Process Occurrence Data" and contains a comment about recreating the selection polygon. The bottom status bar shows "1:1 (Top Level)" and "R Markdown".


```
35 ## Record of analysis for Mastomys natalensis.
36
37 ### Obtain Occurrence Data
38
39 The search for occurrences was limited to 104 records. Obtain
40 occurrence records of the selected species from the gbif database.
41
42 {r}
43 # query selected database for occurrence records
44 results <- spocc::occ(query = "Mastomys natalensis", from = "gbif", limit = 10000, has_coords = TRUE)
45 # retrieve data table from spocc object
46 results.data <- results[["gbif"]]$data[[formatSpName("Mastomys natalensis")]]
47 # remove rows with duplicate coordinates
48 occs.dups <- duplicated(results.data[c('longitude', 'latitude')])
49 occs <- results.data[!occs.dups,]
50 # make sure latitude and longitude are numeric (sometimes they are characters)
51 occs$latitude <- as.numeric(occs$latitude)
52 occs$longitude <- as.numeric(occs$longitude)
53 # give all records a unique ID
54 occs$occID <- row.names(occs)
55 ...
56
57 ### Process Occurrence Data
58
59 The following code recreates the polygon used to select occurrences to
60 keep in the analysis.
61
62 {r}
```

1:1 (Top Level) R Markdown

{disagapp}: what we've done



# {shinyscholar} (Simon Smart)

[Intro](#) [Select data](#) [Plot data](#) [Reproduce](#) [Template](#) [Support](#)

## WORKFLOW

shinyscholar (v1.0.0) includes three components, or steps of a possible workflow. Each component includes two or more modules, which are possible analyses for that step.

**Components:**


- Select Data**
  - Query data via an API
  - Upload your own data
- Plot Data**
  - Histogram
  - Scatter plot
- Reproduce**
  - Download Session Code
  - Download Package References

[About](#) [Team](#) [How To Use](#) [Load Prior Session](#)

## What is shinyscholar?

Welcome to shinyscholar a template for creating flexible applications to academic standards. The template has very basic core functionality but demonstrates how applications can be created that are modular, reproducible and automatically cite package developers.

The application is written in R with the web app development package ([shiny](#)) and is based on the Wallace Eco Mod application ([CRAN website](#)) developed for modelling species distributions (Kass et. al 2018; Kass et al. 2022).



## Attributes of shinyscholar

- accessible:** lowers barriers to implementing complex modular ([shiny](#)) apps for scientific analysis
- open:** the code is free to use and modify (GPL 3.0) and can be viewed from inside the application
- expandable:** users can author and contribute modules that enable new methodological options
- flexible:** options for user uploads and downloads of results
- interactive:** includes an embedded zoomable ([leaflet](#)) map, sortable ([Df](#)) data tables, and visualizations of results
- instructional:** features guidance text that educates users about theoretical and analytical aspects of each step in the workflow
- reproducible:** users can download an ([rmarkdown](#)) .Rmd file that when run reproduces the analysis, and also save sessions and load them later
- robust:** modules and their underlying functions are tested using ([testthat](#)) and ([shinytest2](#))

## Contact us

Please [email us](#) with any other questions.

## Acknowledgments

shinyscholar was developed as part of a project to develop digital tools for modelling infectious diseases funded by Wellcome at the University of Leicester.

## References

- Kass J. M., Vilella B., Aiello-Lammens M. E., Muscarella R., Merow C., Anderson R. P. (2018). Wallace: A flexible platform for reproducible modeling of species niches and distributions built for community expansion. *Methods in Ecology and Evolution*, 9(4): 1151-1156. DOI: [10.1111/2041-210X.12945](#)
- Kass, J.M., Pinilla-Buitrago, G.E, Paz, A., Johnson, B.A., Grisales-Betancur, V., Meenan, S.J., Attali, D., Broennimann, O., Galante, P.J., Maitner, B.S., Owens, H.L., Varela, S., Aiello-Lammens, M.E., Merow, C., Blair, M.E., Anderson R.P. (2022). wallace 2: a shiny app for modeling species niches and distributions redesigned to facilitate expansion via module contributions. *Ecography*, 2023(3): e06547. DOI: [10.1111/ecog.06547](#).

<https://github.com/simon-smart88/shinyscholar>

<https://simonsmart.shinyapps.io/shinyscholar/>

Shinyscholar is a package and application for creating template applications with the same attributes as Wallace and Disagapp

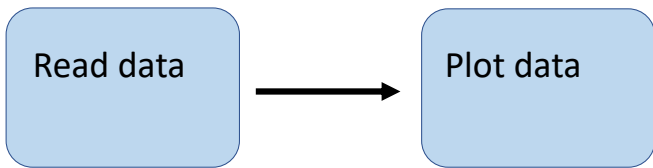


## Quick guide

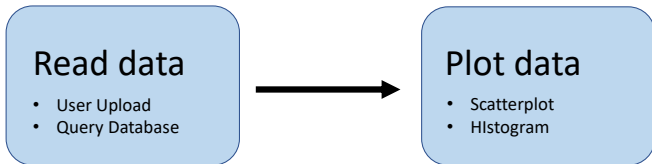
Five (six) Ps

Proper Planning Prevents (Piss) Poor Performance

## Components



## Modules



```
modules <- data.frame(  
  "component" = c("load", "load", "plot", "plot"),  
  "long_component" = c("Load data", "Load data", "Plot data", "Plot data"),  
  "module" = c("user", "database", "histogram", "scatter"),  
  "long_module" = c("Upload your own data", "Query a database to obtain data",  
    "Plot the data as a histogram", "Plot the data as a scatterplot"),  
  "map" = c(FALSE, FALSE, FALSE, FALSE),  
  "result" = c(FALSE, FALSE, TRUE, TRUE),  
  "rmd" = c(TRUE, TRUE, TRUE, TRUE),  
  "save" = c(TRUE, TRUE, TRUE, TRUE))  
  
common_objects = c("data", "histogram", "scatter")
```

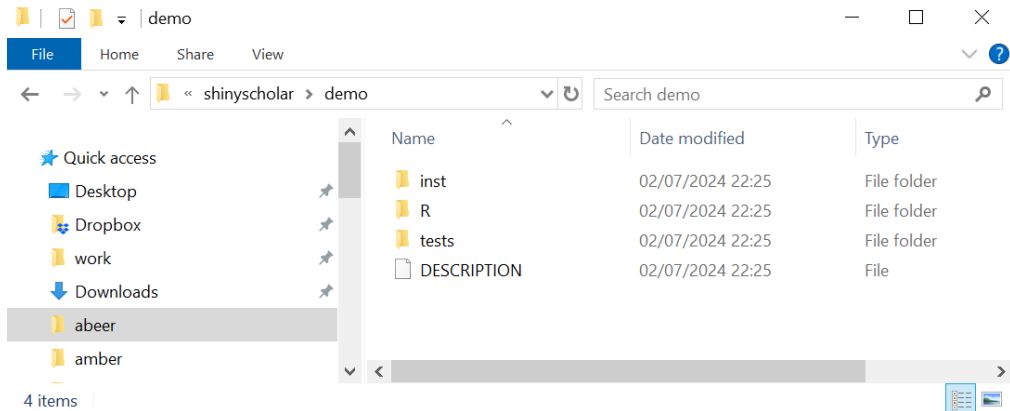
```
modules <- data.frame(  
  "component" = c("load", "load", "plot", "plot"),  
  "long_component" = c("Load data", "Load data", "Plot data", "Plot data"),  
  "module" = c("user", "database", "histogram", "scatter"),  
  "long_module" = c("Upload your own data", "Query a database to obtain data",  
    "Plot the data as a histogram", "Plot the data as a scatterplot"),  
  "map" = c(FALSE, FALSE, FALSE, FALSE),  
  "result" = c(FALSE, FALSE, TRUE, TRUE),  
  "rmd" = c(TRUE, TRUE, TRUE, TRUE),  
  "save" = c(TRUE, TRUE, TRUE, TRUE))  
  
common_objects = c("data", "histogram", "scatter")
```

```
modules <- data.frame(  
  "component" = c("load", "load", "plot", "plot"),  
  "long_component" = c("Load data", "Load data", "Plot data", "Plot data"),  
  "module" = c("user", "database", "histogram", "scatter"),  
  "long_module" = c("Upload your own data", "Query a database to obtain data",  
    "Plot the data as a histogram", "Plot the data as a scatterplot"),  
  "map" = c(FALSE, FALSE, FALSE, FALSE),  
  "result" = c(FALSE, FALSE, TRUE, TRUE),  
  "rmd" = c(TRUE, TRUE, TRUE, TRUE),  
  "save" = c(TRUE, TRUE, TRUE, TRUE))  
  
common_objects = c("data", "histogram", "scatter")
```

{shinyscholar}.

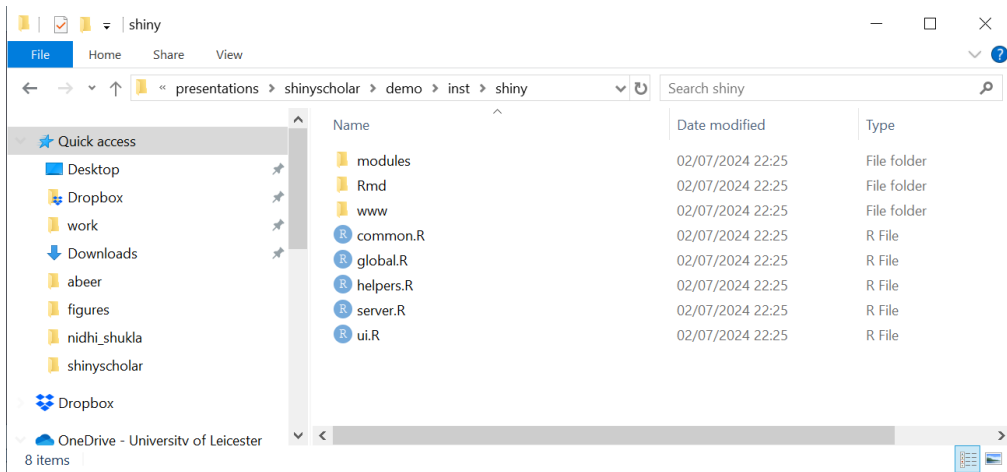
[illegible]

# {shinyscholar} (Simon Smart)





# {shinyscholar} (Simon Smart)



{shinyscholar}.

```
# We created a pkg called {demo}  
library(demo)
```

```
# It has a shiny app called demo  
run_demo()
```

# {shinyscholar} (Simon Smart)

The screenshot shows the shinyscholar web application running in a browser at 127.0.0.1:7112. The browser's address bar and navigation icons are visible at the top. The application's header includes a logo and navigation links: Intro, Load data, Plot data, Reproduce, Support, and a power icon. Below the header, there are tabs for About, Team, How To Use, and Load Prior Session. A 'Skip' dialog box is centered on the screen, with the title 'Skip' and the text 'Welcome to demo! This tour will show you various features of the application to help get you started'. The dialog also features a progress bar with 15 dots, the first of which is filled blue, and 'Prev' and 'Next' buttons. The background content is dimmed but shows a 'WORKFLOW' section with three components: 1. Load data (Upload your own data, Query a database to obtain data), 2. Plot data (Plot the data as a histogram, Plot the data as a scatterplot), and 3. Reproduce (Download Session Code, Download Package References). A large shinyscholar logo is visible on the right side of the page.

127.0.0.1:7112

shinyscholar

Intro Load data Plot data Reproduce Support

About Team How To Use Load Prior Session

## WORKFLOW

*demo* (v0.0.1) includes two components, or steps of a possible workflow. Each component includes one or more modules, which are possible analyses for that step.

**Components:**

- Load data**
  - Upload your own data
  - Query a database to obtain data
- Plot data**
  - Plot the data as a histogram
  - Plot the data as a scatterplot
- Reproduce**
  - Download Session Code
  - Download Package References

**Skip**

Welcome to demo! This tour will show you various features of the application to help get you started

Progress bar: 15 dots, 1st dot filled blue

Prev Next

shinyscholar

# {shinyscholar} (Simon Smart)



Intro

[Load data](#)

Plot data

Reproduce

⊗ Support ▼



## Component: Load data



### Modules Available:

☐ Database

☒ User

### Log window

\*\*\*WELCOME TO DEMO\*\*\*

-----  
Please find messages for the user in this log window.  
-----

[Intro](#)[Load data](#)[Plot data](#)[Reproduce](#)[Support](#)

## Component: Reproduce

### Modules Available:

- ☒ Session Code
- ☐ Reference Packages

## Module: Download Session Code ?

**R packages:** *rmarkdown*, *knitr*

### Select download file type

Rmd



Download Session Code

## Module: *Session Code*

**BACKGROUND** Via the *Session Code* module, the user can download files that document the analyses run in a given *shinyscholar* session (including executable code that can reproduce them). This functionality supports reproducible science.

### IMPLEMENTATION

Here, the user can download documented code that corresponds to the analyses run in the current session of *shinyscholar*. Multiple formats are available for download (.Rmd [R Markdown], .pdf, .html, or .doc). The .Rmd format is an executable R script file that will reproduce the analysis when run in an R session; it is composed of plain text and R code “chunks”. Extended functionality for R Markdown files exists in RStudio. Simply open the .Rmd in RStudio, click on “Run” in the upper-right corner, and

# Modules

Each module calls *one* function in the top R package.

# {shinyscholar} (Simon Smart)

The screenshot shows a Windows File Explorer window with the following details:

- Address Bar:** documents > presentations > shinyscholar > demo > R
- Search Bar:** Search R
- Left Pane (Navigation):**
  - Quick access
    - Desktop
    - Dropbox
    - work
    - Downloads
    - abeer
    - figures
    - nidhi\_shukla
    - shinyscholar
  - Dropbox
  - OneDrive - Universitv of Leicester
- 6 items** are listed in the main pane:

Name	Date modified	Type
load_database.R	02/07/2024 22:25	R File
load_user.R	02/07/2024 22:25	R File
plot_histogram.R	02/07/2024 22:25	R File
plot_scatter.R	02/07/2024 22:25	R File
run_demo.R	02/07/2024 22:25	R File
run_module.R	02/07/2024 22:25	R File

To fully edit load database module.

Add code to `load_database.R` in package.

Add code to `load_database.R` in shiny app.

Edit `load_database.md`

Edit `load_database.Rmd`

Add packages to `.yaml`



Add code to load\_database.R in package

```
load_user <- function(x){  
  return(NULL)  
}
```

Add code to load\_database.R in package

```
load_user <- function(x){  
  
  data <- data.frame(x = rnorm(100),  
                     y = rnorm(100))  
  
  return(data)  
}
```

## Why put code in a package?

Testable

Transportable

Used in reproducible Rmarkdown

## Add code to load\_user.R in shiny app

```
load_user_module_server <- function(id, common, parent_session) {  
  moduleServer(id, function(input, output, session) {  
  
    observeEvent(input$run, {  
      # WARNING ####  
  
      # FUNCTION CALL ####  
  
      # LOAD INTO COMMON ####  
    })  
  })  
}
```

## Add code to load\_user.R in shiny app

```
load_user_module_server <- function(id, common, parent_session) {  
  moduleServer(id, function(input, output, session) {  
  
    observeEvent(input$run, {  
      # WARNING ####  
  
      # FUNCTION CALL ####  
  
      data <- load_user()  
      common$data <- data  
  
      # LOAD INTO COMMON ####  
    })  
  })  
}
```

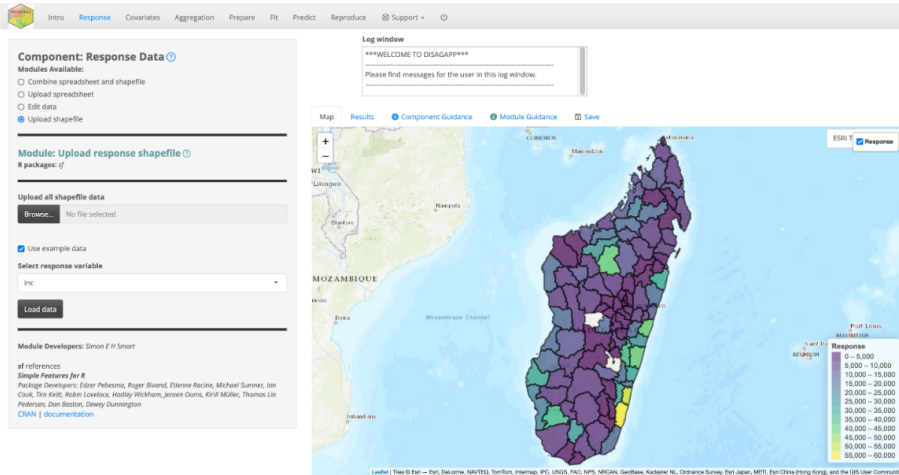
{shinyscholar} is itself a {shinyscholar} package/app, so can be used as a guide.

Vignette and README.

Paper (coming soon)

Email us! Happy to help.

# {disagapp} (Simon Smart)



There are various modules for uploading the response data depending on the current state

Please ask me some questions

 @StatsForBios.bsky.social

 @timcdlucas @statsforbios

 tim.lucas@le.ac.uk

 ss1545@le.ac.uk

[www.github.com/simon-smart88/shinyscholar](https://www.github.com/simon-smart88/shinyscholar)