

Creating basic R workflows and literate programming

Ernest Guevarra

26 November 2024

Outline

- Project-oriented workflows in R
 - Why organise workflows into projects
 - How to organise workflows into projects
- Literate programming
 - Why use literate programming
 - RMarkdown for literate programming
- Practical session

Project-oriented workflows in R

Why organise workflows into projects

- key word is "*organise*";
- organisation ensures *coherence* and *order* which is helpful to the user whenever he/she goes back to the same work again later;
- organisation supports good documentation; and,
- organisation allows for collaboration.

How to organise workflows into projects

- a unique piece of "*work*" should be its own *project*;
- all resources and tools needed for a *project* should be within the same directory; and,
- group resources and tools within appropriately into their own directories with the project directory.

OxfordIHTM / Awareness-Driven-Behaviour (Private)

Code Issues 1 Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags Go to file Add file <> Code

grecomalijan	Delete Model Code.Rmd	2e57851 on Aug 8, 2021	2 commits
App	Add files via upload		last year
Model_Comparison.csv	Add files via upload		last year
PHSI.csv	Add files via upload		last year
PH_NPI+STA+LTA.R	Add files via upload		last year
PHcases.csv	Add files via upload		last year
PHdeaths.csv	Add files via upload		last year
README.md	Add files via upload		last year
Sensi.csv	Add files via upload		last year

OxfordIHTM / sc-vaccine-study-draft (Public)

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags Go to file Add file <> Code

gsrajahram	initial commit	3affb15 on Dec 9, 2022	1 commit
data	initial commit		last month
output	initial commit		last month
.gitignore	initial commit		last month
sc-vaccine-study-draft.Rproj	initial commit		last month
sc_vaccines_study_non_targets.R	initial commit		last month

Help people interested in this repository understand your project by adding a README. Add a README

Literate programming

Why use literate programming

- write code that satisfies both the machine/computer and the human reader/user;
- highlight and give as much importance to the documentation of the processes and the outputs that your code represents;
- focus on communication and understanding of the process and the output; and,
- single document that integrates both code and textual documentation.

R Markdown for literate programming

- R Markdown provides an authoring framework for literate programming;
- A single R Markdown file is used to both save and execute code, and generate high quality reports that can be shared with an audience; and,
- R Markdown documents are fully reproducible and support dozens of static and dynamic output formats.

R Markdown example - PDF report

2189 lines (1748 sloc) | 142 KB

```
1 ---  
2 title: Report on Coverage Assessment of Direct Nutrition Interventions in Liberia  
3 #author: "Valid International"  
4 date: `r format(Sys.Date(), "%d %B %Y")`  
5 fontsize: 12pt  
6 geometry: margin=2cm  
7 documentclass: article  
8 classoption: a4paper  
9 bibliography: bibliography.bib  
10 lot: TRUE  
11 lof: TRUE  
12 link-citations: TRUE  
13 links-as-notes: FALSE  
14 colorlinks: TRUE  
15 linkcolor: blue  
16 citecolor: blue  
17 urlcolor: blue  
18 ---  
19  
20 ```{r setup, include = FALSE}  
21 knitr::opts_chunk$set(echo = TRUE,  
22                 warning = FALSE,  
23                 message = FALSE)  
24  
25 if(!require(stringr)) install.packages("stringr")  
26 if(!require(raster)) install.packages("raster")  
27 if(!require(bbw)) install.packages("bbw")  
28 if(!require(tidyR)) install.packages("tidyR")  
29 if(!require(ggplot2)) install.packages("ggplot2")  
30 if(!require(knitr)) install.packages("knitr")  
31 if(!require(kableExtra)) install.packages("kableExtra")  
32 if(!require(RColorBrewer)) install.packages("RColorBrewer")  
33 if(!require(rgdal)) install.packages("rgdal")  
34 if(!require(cowplot)) install.packages("cowplot")  
35  
36 if(!require(remote)) install.packages("remote")  
37 if(!require(liberia)) install_github("validmeasures/liberia")  
38 if(!require(liberiaData)) install_github("validmeasures/liberiaData")  
39  
40 options(stringsAsFactors = FALSE)  
41 options(kableExtra.latex.load_packages = FALSE)  
42  
43 themeSettings <- theme_bw() +  
44 theme(
```



THE POWER OF
nutrition **VALID**

Report on Coverage Assessment of Direct Nutrition Interventions in Liberia

R Markdown example - HTML report

505 lines (440 sloc) | 9.3 KB

```
1  ---
2  title: Lifetables Seychelles 2021
3  author: Ministry of Health
4  date: 2022-06-16
5  output: html_document
6  ---
7
8  ``{r setup, include=FALSE}
9  knitr::opts_chunk$set(echo = FALSE)
10
11 ## Load packages and functions
12 suppressPackageStartupMessages(source("packages.R"))
13 for (f in list.files(here("R"), full.names = TRUE)) source (f)
14
15 ## Load required targets
16 tar_load(
17   c(
18     sc_lifetable, smooth_sc_lifetable,
19     demogdat, smooth_demogdat,
20     #lca_male, predict_lca_male,
21     #lca_female, predict_lca_female,
22     lca_total, predict_lca_total,
23     lca_smooth_total, predict_lca_smooth_total
24   )
25 )
26 ...
27
28 ## Lifetable
29
30 2021 life table samples, with forecast of mortality rates for next 30 years (based on 2011-2021 lifetable data, using lee-carter approach)
31
32 ### Lifetable by sex - 2021
33
34 ``{r lifetable-2021, fig.align = "center", fig.width = 12, fig.height = 8}
35 par(mfrow = c(1, 2))
36
37 plot(
38   sc_lifetable,
39   series = "male",
40   datatype = "rate",
41   transform = FALSE,
42   main = "Male death rates by age - 2021",
43   lty = 0,
44   ylim = c(0, max(sc_lifetable$rate$male, sc_lifetable$rate$female)),
```

Lifetables Seychelles 2021

Ministry of Health

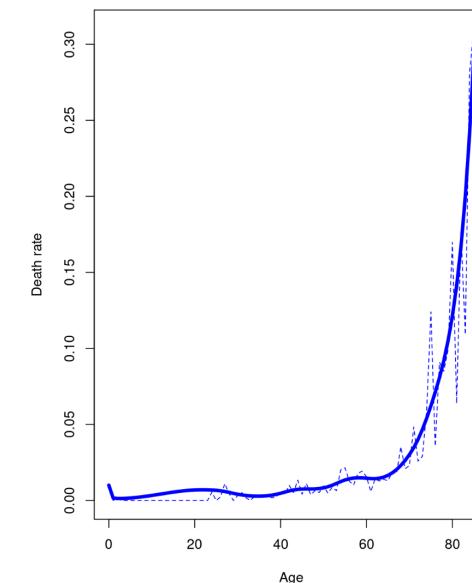
2022-06-16

Lifetable

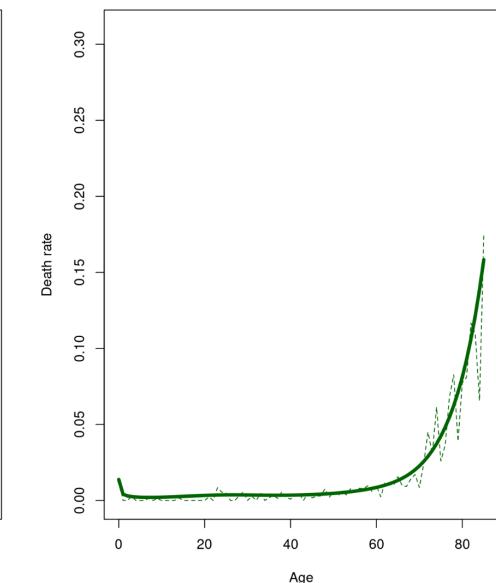
2021 life table samples, with forecast of mortality rates for next 30 years (based on 2011-2021 lifetable data, using lee-carter approach)

Lifetable by sex - 2021

Male death rates by age - 2021



Female death rates by age - 2021



Male death rates by age - 2021

Female death rates by age - 2021

R Markdown example - Website

295 lines (156 sloc) | 18.5 KB

```
1 ---  
2 output: github_document  
3 ---  
4  
5 <!-- README.md is generated from README.Rmd. Please edit that file -->  
6  
7 ```{r, include = FALSE}  
8 knitr::opts_chunk$set(  
9   collapse = TRUE,  
10   comment = "#>"  
11 )  
12 ...  
13  
14 # Open Science and Reproducible Research in R: *An Oxford International Health and Tropical Medicine Lecture Series*  
15  
16 <!-- badges: start -->  
17 [![license for data](https://img.shields.io/badge/(for%20data)-CC0-blue.svg)](https://creativecommons.org/publicdomain/zero/1.0/)  
18 [![license for code](https://img.shields.io/badge/(for%20code)-GPL3.0-blue.svg)](https://opensource.org/licenses/gpl-3.0.html)  
19 [![license for slide deck](https://img.shields.io/badge/(for%20slide%20decks)-CC%20BY%204.0-blue.svg)](https://creativecommons.org/licenses/by/4.0/)  
20 [![pages-build-deployment](https://github.com/OxfordIHTM/open-reproducible-science/actions/workflows/pages/pages-build-deployment/badge.svg)]  
  (https://github.com/OxfordIHTM/open-reproducible-science/actions/workflows/pages/pages-build-deployment)  
21 [![DOI](https://zenodo.org/badge/451333046.svg)](https://zenodo.org/badge/latestdoi/451333046)  
22 <!-- badges: end -->  
23  
24 ## Part 1: All about R  
25  
26 [R](https://r-project.org) is a language and environment for statistical computing and graphics. It is a [GNU](https://en.wikipedia.org/wiki/GNU) project  
  which is similar to the [S language and environment](https://en.wikipedia.org/wiki/S_(programming_language)) which was developed at Bell Laboratories  
  (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues. R can be considered as a different implementation of S. There are some important  
  differences, but much code written for S runs unaltered under R.  
27  
28 [R](https://r-project.org) provides a wide variety of statistical (linear and nonlinear modelling, classical statistical tests, time-series analysis,  
  classification, clustering, etc.) and graphical techniques, and is highly extensible. The S language is often the vehicle of choice for research in  
  statistical methodology, and R provides an Open Source route to participation in that activity.  
29  
30 One of [R](https://r-project.org)'s strengths is the ease with which well-designed publication-quality plots can be produced, including mathematical symbols  
  and formulae where needed. Great care has been taken over the defaults for the minor design choices in graphics, but the user retains full control.  
31  
32 R is available as Free Software under the terms of the Free Software Foundation's GNU General Public License in source code form. It compiles and runs on a  
  wide variety of UNIX platforms and similar systems (including FreeBSD and Linux), Windows and MacOS.  
33  
34 R is unique in that it is not general-purpose. It does not compromise by trying to do a lot of things. It does a few things very well, mainly statistical  
  analysis and data visualization. While you can find data analysis and machine learning libraries for languages like [Python](https://www.python.org/), R has  
  many statistical functionalities built into its core. No third-party libraries are needed for much of the core data analysis you can do with the language.
```

Open Reproducible Science

Open Science and Reproducible Research in R Lecture Series

[View the Project on GitHub](#)
OxfordIHTM/open-reproducible-science

license (for data) CC0 license (for code) GPL3.0
license (for slide decks) CC BY 4.0 DOI 10.5281/zenodo.7515077 pages-build-deployment passing

Part 1: All about R

R is a language and environment for statistical computing and graphics. It is a GNU project which is similar to the S language and environment which was developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues. R can be considered as a different implementation of S. There are some important differences, but much code written for S runs unaltered under R.

R provides a wide variety of statistical (linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering, etc.) and graphical techniques, and is highly extensible. The S language is often the vehicle of choice for research in statistical methodology, and R provides an Open Source route to participation in that activity.

One of R's strengths is the ease with which well-designed publication-quality plots can be produced, including mathematical symbols and formulae where needed. Great care has been taken over the defaults for the minor design choices in graphics, but the user retains full control.

R is available as Free Software under the terms of the Free Software Foundation's GNU General Public License in source code form. It compiles and runs on a wide variety of UNIX platforms and similar systems (including FreeBSD and Linux), Windows and MacOS.

R is unique in that it is not general-purpose. It does not compromise by trying to do a lot of things. It does a few things very well, mainly statistical analysis and data visualization. While you can find data analysis and machine learning libraries for languages like Python, R has many statistical functionalities built into its core. No third-party libraries are needed for much of the core data analysis you can do with the language.

This project is maintained by OxfordIHTM

Hosted on GitHub Pages — Theme by orderedlist

R Markdown example - slide deck

```
273 lines (173 sloc) | 8.7 KB
1  ---
2  title: "Getting the right tools for the job"
3  subtitle: "R, RStudio, git, and GitHub"
4  author:
5    - "Ernest Guevarra"
6  date: '2023-01-12'
7  output:
8    xaringan::moon_reader:
9      css: xaringan-themer.css
10     nature:
11       slideNumberFormat: "%current%"
12       highlightStyle: github
13       highlightLines: true
14       ratio: 16:9
15       countIncrementalSlides: true
16   ...
17
18 ``{r setup, include=FALSE}
19 options(htmltools.dir.version = FALSE)
20 knitr::opts_chunk$set(
21   fig.width=9, fig.height=3.5, fig.retina=3,
22   out.width = "100%",
23   cache = FALSE,
24   echo = TRUE,
25   message = FALSE,
26   warning = FALSE,
27   hline = TRUE
28 )
29
30 if (!require(remotes)) install.packages("remotes")
31 if (!require(fontawesome)) remotes::install_github("rstudio/fontawesome")
32 ...
33
34 ``{r xaringan-themer, include=FALSE, warning=FALSE}
35 library(xaringanthemr)
36 style_mono_light(
37   base_color = "#002147",
38   title_slide_background_image = "",
39   title_slide_background_size = "cover",
40   header_font_google = google_font("Fira Sans"),
41   text_font_google = google_font("Fira Sans Condensed"),
42   text_font_size = "1.2em",
43   link_color = "#214700",
44   header_h1_font_size = "50px",
```



Basics of R Markdown

- An R Markdown document is written in markdown (an easy-to-write plain text format) and contains chunks of embedded R code, like the document below.

```
---
```

```
output: html_document
```

```
--
```

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see .

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
```{r}
```

```
summary(cars)
```

```
```
```

You can also embed plots, for example:

```
```{r, echo=FALSE}
```

```
plot(cars)
```

```
```
```

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

Basic Markdown syntax

Markdown syntax

- *italics* or italics
- **bold** or bold
- > block quote
- # Header 1
- ## Header 2
- ### Header 3
- ##### Header 4

Resulting output

- *italics*
- **bold**
- | block quote
- # Header 1
- ## Header 2
- ### Header 3
- #### Header 4

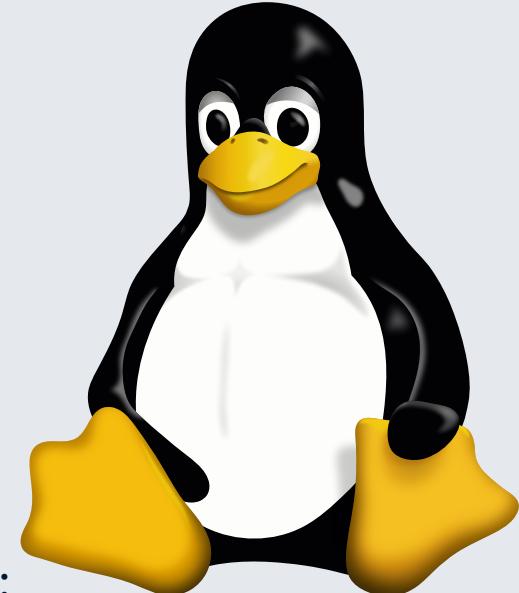
Basic Markdown syntax

Markdown syntax

- [link to Open Reproducible Science website](<https://oxford-ihtm.io/open-reproducible-science/>)
- adding an image:

Resulting output

- link to Open Reproducible Science website



- adding an image:

Questions?

Practical session

We'll work through *Exercise 5 - Extending R with packages* in Practical R for Epidemiologists (<https://practical-r.org/exercise5.html>) as a GitHub Classroom assignment

Thank you!

Slides can be viewed at <https://oxford-ihtm.io/open-reproducible-science/session5.html>

PDF version of slides can be downloaded at <https://oxford-ihtm.io/open-reproducible-science/pdf/session5-literate-programming.pdf>

R scripts for slides available [here](#)