Exploratory Analysis of Salmon smelt

Matthew B. Jones[1,*] Bryce Mecum[1] S. Jeanette Clark[1]

12 September, 2021

Abstract

We review approaches to computational reproducibility that are helpful in writing a reproducible paper. By representing a paper as an RMarkdown document inside of an R package, we can powerfully interleave the text of the paper with the data cleaning, analysis, and visualization code that is used to create the tables and figures in the paper. Thus, the paper itself can be executed, and the end product produces the full paper in a way that documents all of the data inputs, data cleaning and integration, and analysis and visualization steps as a computational workflow. The whole package provides structured documentation about the requirements needed to execute the paper, including the R packages that need to be installed. And it can format the paper with advance mathematical equations, inline citations, figures and tables inline, and a reference section that in the style needed for specific journals.

Contents

[1] National Center for Ecological Analysis and Synthesis
[2] University of California Santa Barbara

[*] Correspondence: Matthew B. Jones <jones@nceas.ucsb.edu>

Keywords: open science; reproducible papers; transparency; provenance

Highlights: Transparency leads to reproducibility, and can be greatly facilitted through tooling that tracks specific data and workflows used for analysis and modeling in a paper.

1 Introduction

Writing reports and academic papers is a ton of work but a large amount of that work can be spent doing monotonous tasks such as:

- Updating figures and tables as we refine our analysis
- Editing our analysis and, in turn, editing our paper's text
- Managing bibliography sections and in-text citations/references

These monotonous tasks are also highly error-prone. With RMarkdown, we can close the loop, so to speak, between our analysis and our manuscript because the manuscript can become the analysis.

As an alternative to Microsoft Word, RMarkdown provides some advantages:

- Free to use
- Uses text so we can:
  - Use version control for
    - Tracking changes
    - Collaborating
  - Edit it with our favorite and most powerful text editors
  - Use the command line for automation

The rest of this document will show how we get some of the features we need such as:

- Attractive typesetting for mathematics
- Figures, tables, and captions
- In-text citations
- Bibliographies

2 Methods

Our analysis will be pretty simple. We'll use the `diamonds` dataset from the `ggplot2` (Wickham 2009) package and run a simple linear model. At the top of this document, we started with a code chunk with `echo=FALSE` set as a chunk option so that we can load the `ggplot2` package and `diamonds` dataset without outputting anything to the screen.

For our analysis, we'll create a really great plot which really shows the relationship between price and carat and shows how we include plots in our document.

2.1 Mathematics

Then we'll run a linear model of the form $y = mx + b$ on the relationship between price and carat and shows how we include tables in our document. Note how the previous equation was rendered inline. We can also put some more advanced math in our paper and it will be beautifully typeset as a block equation:

$$\sum_{i=1}^{N} log(i) + \frac{\omega}{x}$$

2.2 Citations

We can also use R itself to generate bibliographic entries for the packages we use so we can give proper credit when we use other peoples' packages in our analysis. Here we cite the `ggplot2` package:

```
> citation('ggplot2')

To cite ggplot2 in publications, please use:

  H. Wickham. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York, 2009.

A BibTeX entry for LaTeX users is

  @Book{ggplot,
    author = {Hadley Wickham},
    title = {ggplot2: Elegant Graphics for Data Analysis},
    publisher = {Springer-Verlag New York},
    year = {2009},
    isbn = {978-0-387-98140-6},
    url = {http://ggplot2.org},
  }
```

And then we just place that in our `.bibtex` file, and we can cite it inline to indicate that we used `ggplot2` for visualiations (Wickham 2009). Note how these inline citations are simply the bibliographic `key` for the bibtext entry, nested in square brackets with an `@` sign, like this: `[@ggplot]`. We can cite other papers in the flow of the text, for example by indicating that this guide follows (Marwick 2017). And when we cite all of the software that we used (R Core Team 2015)(RStudio Team 2015)(RMarkdown Team 2015), they will all be rendered in the References section.

2.3 Inline text

As we report on our methods, we can even include details inline from the text about the data and analysis. For example, we might report that the Delta smelt dataset contains 4309 samples from trawl surveys. We do that by embedding snippets of R code using backticks in the prose that get evaluated, and the results of those snippets are formatted in the text.

3 Results

In the results section, you can interleave text and analysis. For example, we might start with a simple plot of a sample from the normal distribution. Building this does not require loading a dataset explicitly.

Figure 1 shows how we can have a caption and cross-reference for a plot in the text.

Alternatively, we could also load a dataset from a file that is in the package itself, in the `analysis/data/raw_data` directory. Here's how you might do that, but we're going to skip that step because embedding data in the package directly isn't the best practice unless it is very small, and is unlikely to be used in other papers and analyses.

```
# Note the path that we need to use to access our data files when rendering this document
my_data <- read.csv(here::here('analysis/data/raw_data/my_csv_file.csv'))
```

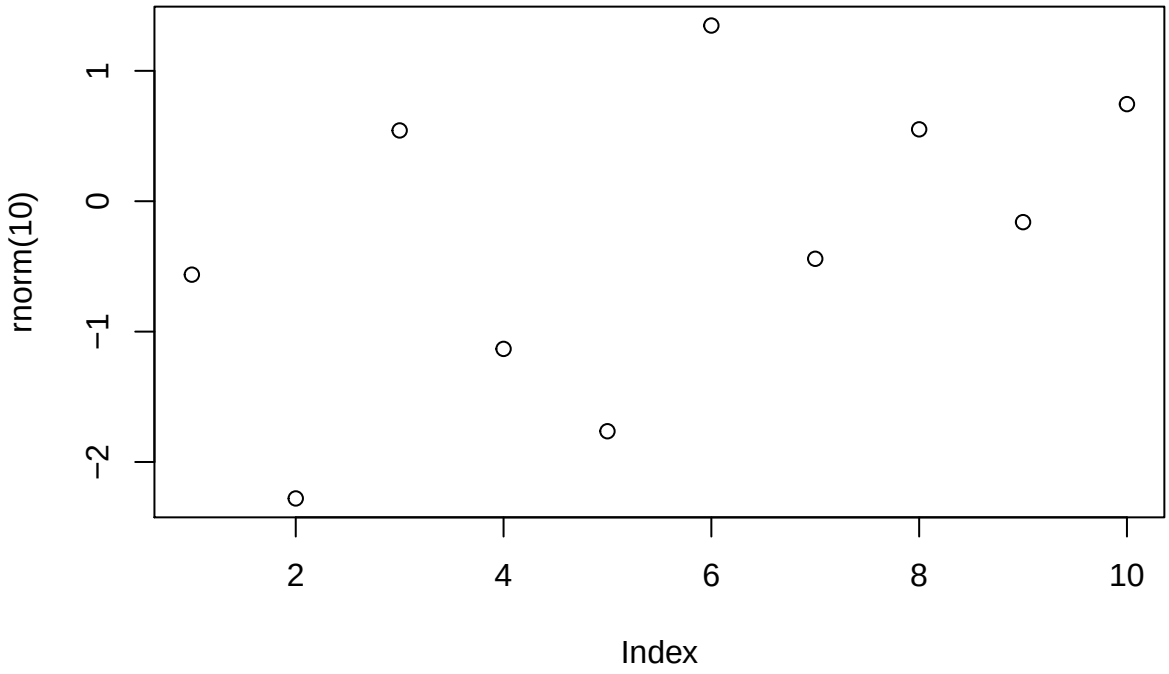

Figure 1: A plot of random numbers

Table 1: This is a broomed linear model summary table.

term	estimate	std.error	statistic	p.value
(Intercept)	0.37	0.09	4.14	0
Year	0.00	0.00	-4.10	0

A better approach is to load the data from an archival, versioned data repository, and cache it locally for speed. This allows anyone to run the script, which downloads and caches the data on the local system only on the first run. This means we are no longer dependent on local file paths, and the analysis is portable across many machines. Here's how we loaded the Delta Smelt dataset from EDI earlier in this document:

```
# We use the `contentid` package to ensure that data are referenced properly online, but also that they
# can be used locally with an unambiguous version
delta.file <- contentid::resolve("hash://sha1/3ccff226e8aefed9448386bbb09311239475301d", store = TRUE)
delta.df <- readr::read_csv(delta.file, show_col_types=FALSE)
```

Now that we have the data file loaded, we can build a linear model of Smelt count over time, and then plot it (Figure 2).

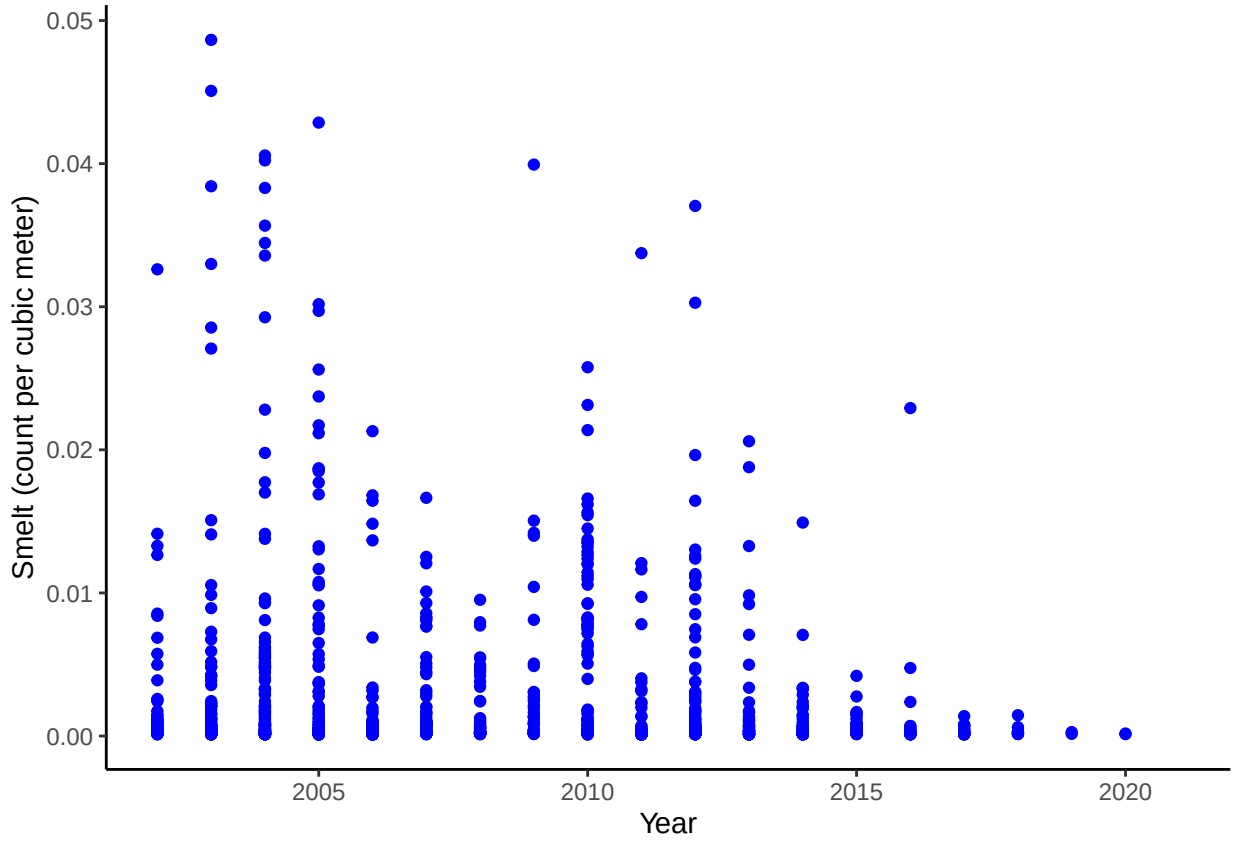

Figure 2: Smelt per cubic Meter by year across all surveys.

The plot we made was really great, but the model was even better. We were delighted to find that the slope parameter was -2×10^{-4}.

4 Discussion

This was just a quick demonstration of a reproducible paper that combined text, analysis, figures, tables, and citations into multiple output formats (HTML, PDF). Hopefully you found it useful.

A lot of people are using RMarkdown these days so there are tons of resources online but here are a few choice ones specifically about making papers:

- http://rmarkdown.rstudio.com/authoring_bibliographies_and_citations.html
- http://svmiller.com/blog/2016/02/svm-r-markdown-manuscript/
- http://www.petrkeil.com/?p=2401

5 Data Availability

Data are available from the EDI Data Repository:

Interagency Ecological Program (IEP), Lauren Damon, and Adam Chorazyczewski. 2021. Interagency Ecological Program San Francisco Estuary Spring Kodiak Trawl Survey 2002 - 2021. Environmental Data Initiative. https://pasta.lternet.edu/package/metadata/eml/edi/527/4.

6 Acknowledgements

7 References

Marwick, Ben. 2017. "Computational Reproducibility in Archaeological Research: Basic Principles and a Case Study of Their Implementation." *Journal of Archaeological Method and Theory* 24 (2): 424–50. https://doi.org/10.1007/s10816-015-9272-9.

R Core Team. 2015. "R: A Language and Environment for Statistical Computing." http://www.r-project.org.

RMarkdown Team. 2015. *Rmarkdown: R Markdown Document Conversion, r Package*. Boston, MA: RStudio, Inc. http://rmarkdown.rstudio.com/.

RStudio Team. 2015. *RStudio: Integrated Development Environment for r*. Boston, MA: RStudio, Inc. http://www.rstudio.com/.

Wickham, Hadley. 2009. *Ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. http://ggplot2.org.

7.0.1 Colophon

This report was generated on 2021-09-12 20:35:29 using the following computational environment and dependencies:

```
#> - Session info ---------------------------------------------------------------
#>  setting  value
#>  version  R version 4.0.3 (2020-10-10)
#>  os       macOS Mojave 10.14.6
#>  system   x86_64, darwin17.0
#>  ui       X11
#>  language (EN)
#>  collate  en_US.UTF-8
#>  ctype    en_US.UTF-8
#>  tz       America/New_York
#>  date     2021-09-12
#>
#> - Packages -------------------------------------------------------------------
#>  package     * version date       lib source
#>  askpass       1.1     2019-01-13 [1] CRAN (R 4.0.0)
#>  assertthat    0.2.1   2019-03-21 [1] CRAN (R 4.0.0)
#>  backports     1.2.1   2020-12-09 [1] CRAN (R 4.0.2)
#>  bit           4.0.4   2020-08-04 [1] CRAN (R 4.0.2)
#>  bit64         4.0.5   2020-08-30 [1] CRAN (R 4.0.2)
#>  bookdown      0.24    2021-09-02 [1] CRAN (R 4.0.2)
#>  broom       * 0.7.9   2021-07-27 [1] CRAN (R 4.0.2)
#>  cachem        1.0.6   2021-08-19 [1] CRAN (R 4.0.2)
#>  callr         3.7.0   2021-04-20 [1] CRAN (R 4.0.2)
#>  cli           3.0.1   2021-07-17 [1] CRAN (R 4.0.2)
#>  colorspace    2.0-2   2021-06-24 [1] CRAN (R 4.0.2)
#>  contentid   * 0.0.12  2021-09-12 [1] Github (cboettig/contentid@93cc31c)
#>  crayon        1.4.1   2021-02-08 [1] CRAN (R 4.0.2)
#>  curl          4.3.2   2021-06-23 [1] CRAN (R 4.0.2)
#>  DBI           1.1.1   2021-01-15 [1] CRAN (R 4.0.2)
#>  desc          1.3.0   2021-03-05 [1] CRAN (R 4.0.2)
#>  devtools      2.4.2   2021-06-07 [1] CRAN (R 4.0.2)
#>  digest        0.6.27  2020-10-24 [1] CRAN (R 4.0.2)
#>  dplyr         1.0.7   2021-06-18 [1] CRAN (R 4.0.2)
#>  ellipsis      0.3.2   2021-04-29 [1] CRAN (R 4.0.2)
#>  evaluate      0.14    2019-05-28 [1] CRAN (R 4.0.0)
#>  fansi         0.5.0   2021-05-25 [1] CRAN (R 4.0.2)
#>  farver        2.1.0   2021-02-28 [1] CRAN (R 4.0.2)
#>  fastmap       1.1.0   2021-01-25 [1] CRAN (R 4.0.2)
#>  fs            1.5.0   2020-07-31 [1] CRAN (R 4.0.2)
#>  generics      0.1.0   2020-10-31 [1] CRAN (R 4.0.2)
#>  ggplot2     * 3.3.5   2021-06-25 [1] CRAN (R 4.0.2)
#>  glue          1.4.2   2020-08-27 [1] CRAN (R 4.0.2)
#>  gtable        0.3.0   2019-03-25 [1] CRAN (R 4.0.0)
#>  highr         0.9     2021-04-16 [1] CRAN (R 4.0.2)
#>  hms           1.1.0   2021-05-17 [1] CRAN (R 4.0.2)
#>  htmltools     0.5.2   2021-08-25 [1] CRAN (R 4.0.2)
#>  httr          1.4.2   2020-07-20 [1] CRAN (R 4.0.2)
#>  knitr       * 1.34    2021-09-09 [1] CRAN (R 4.0.2)
#>  labeling      0.4.2   2020-10-20 [1] CRAN (R 4.0.2)
```

```
#>  lifecycle     1.0.0   2021-02-15 [1] CRAN (R 4.0.2)
#>  magrittr      2.0.1   2020-11-17 [1] CRAN (R 4.0.2)
#>  memoise       2.0.0   2021-01-26 [1] CRAN (R 4.0.2)
#>  munsell       0.5.0   2018-06-12 [1] CRAN (R 4.0.0)
#>  openssl       1.4.5   2021-09-02 [1] CRAN (R 4.0.2)
#>  pillar        1.6.2   2021-07-29 [1] CRAN (R 4.0.2)
#>  pkgbuild      1.2.0   2020-12-15 [1] CRAN (R 4.0.2)
#>  pkgconfig     2.0.3   2019-09-22 [1] CRAN (R 4.0.0)
#>  pkgload       1.2.2   2021-09-11 [1] CRAN (R 4.0.3)
#>  prettyunits   1.1.1   2020-01-24 [1] CRAN (R 4.0.0)
#>  processx      3.5.2   2021-04-30 [1] CRAN (R 4.0.2)
#>  ps            1.6.0   2021-02-28 [1] CRAN (R 4.0.2)
#>  purrr         0.3.4   2020-04-17 [1] CRAN (R 4.0.0)
#>  R6            2.5.1   2021-08-19 [1] CRAN (R 4.0.2)
#>  readr         2.0.1   2021-08-10 [1] CRAN (R 4.0.2)
#>  remotes       2.4.0   2021-06-02 [1] CRAN (R 4.0.2)
#>  rlang         0.4.11  2021-04-30 [1] CRAN (R 4.0.2)
#>  rmarkdown     2.10    2021-08-06 [1] CRAN (R 4.0.2)
#>  rprojroot     2.0.2   2020-11-15 [1] CRAN (R 4.0.2)
#>  rstudioapi    0.13    2020-11-12 [1] CRAN (R 4.0.2)
#>  scales        1.1.1   2020-05-11 [1] CRAN (R 4.0.2)
#>  sessioninfo   1.1.1   2018-11-05 [1] CRAN (R 4.0.0)
#>  stringi       1.7.4   2021-08-25 [1] CRAN (R 4.0.2)
#>  stringr       1.4.0   2019-02-10 [1] CRAN (R 4.0.0)
#>  testthat      3.0.4   2021-07-01 [1] CRAN (R 4.0.2)
#>  tibble        3.1.4   2021-08-25 [1] CRAN (R 4.0.2)
#>  tidyr         1.1.3   2021-03-03 [1] CRAN (R 4.0.2)
#>  tidyselect    1.1.1   2021-04-30 [1] CRAN (R 4.0.2)
#>  tzdb          0.1.2   2021-07-20 [1] CRAN (R 4.0.2)
#>  usethis       2.0.1   2021-02-10 [1] CRAN (R 4.0.2)
#>  utf8          1.2.2   2021-07-24 [1] CRAN (R 4.0.2)
#>  vctrs         0.3.8   2021-04-29 [1] CRAN (R 4.0.2)
#>  vroom         1.5.4   2021-08-05 [1] CRAN (R 4.0.2)
#>  withr         2.4.2   2021-04-18 [1] CRAN (R 4.0.2)
#>  xfun          0.25    2021-08-06 [1] CRAN (R 4.0.2)
#>  yaml          2.2.1   2020-02-01 [1] CRAN (R 4.0.0)
#>
#> [1] /Library/Frameworks/R.framework/Versions/4.0/Resources/library
```

The current Git commit details are:

```
#> Local:    master /Users/jones/development/repropaper2021
#> Head:     [c9fd2cf] 2021-09-12: Initial commit
```