My Example Computed Manuscript
Created in Rmarkdown

Jeffrey M. Perkel ·

Abstract A mock computed manuscript created in RStudio using {Rmarkdown}. The {Bookdown} and {Rticles} packages were used to output the text in Springer Nature's desired manuscript format.

Keywords

1 Introduction

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 `http://rmarkdown.rstudio.com`.

Here we'll add some references from Zotero (Perkel 2020): (Fisch et al. 2015; Argelaguet et al. 2021; Lê Cao et al. 2021).

Markdown documents can include inline equations written in LaTeX, such as $F = ma$. Here is an equation on its own line:

$$a^2 + b^2 = c^2$$

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Jeffrey M. Perkel
Springer Nature, 1 New York Plaza, New York, NY
E-mail: `jeffrey.perkel@nature.com`

Table 1 Our fruit basket

Type	Count	Price	Total
apple	5	1.00	5.00
banana	12	0.50	6.00
cantaloupe	2	2.00	4.00
honeydew	2	3.00	6.00
orange	7	0.75	5.25

2 Results

2.1 Inline computation

One 'killer feature' of computed manuscripts is the ability to compute and insert values into the text rather than requiring authors to type them in. That circumvents the problems that can occur if an author makes a mistake in inputting numbers or forgets to update them, for instance, after new data have been folded into the calculations.

Suppose we go to the market and purchase some fruit, and save the data in a spreadsheet. (Table 1).

```r
# create a mock spreadsheet
fruit_basket <- tribble(
  ~type,~count,~unit_price,
  "apple",5,1,
  "banana",12,0.5,
  "cantaloupe",2,2,
  "honeydew",2,3,
  "orange",7,0.75) %>%
  mutate(total = count * unit_price)
knitr::kable(fruit_basket, format = 'latex', booktabs = TRUE,
             col.names = c("Type", "Count", "Price", "Total"),
             align = 'lccr', # left, center, center, right
             caption = "Our fruit basket")
```

Rather than hard-coding the numbers from our purchase directly in the text, we can use the programming language R to fill those numbers in for us. For instance, the text: "We purchased `r sum(fruit_basket$count)` pieces of fruit..." produces the output: "We purchased 28 pieces of fruit...." That way, if we make any additional purchases, we can update our manuscript automatically.

In the following text, computed values are indicated in **bold type**: We purchased **5** types of fruit at the grocery store. We purchased **28** pieces in all, including **12** bananas and **5** apples. The total cost was **$26.25**.
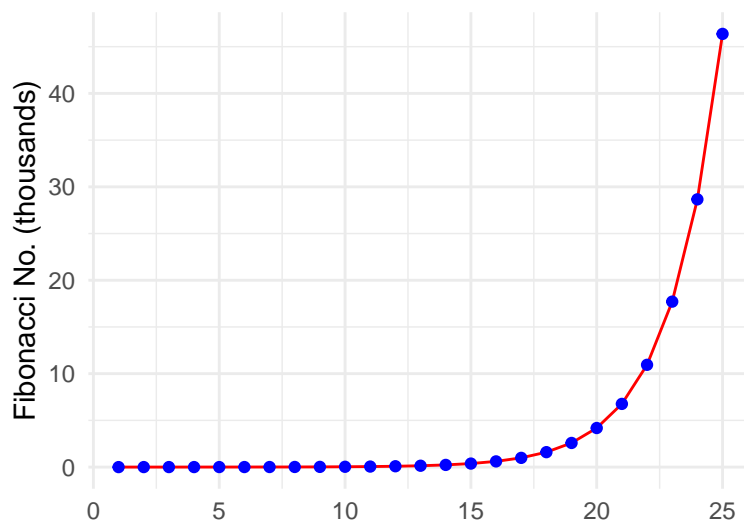
Fig. 1 The first 26 numbers of the Fibonacci sequence

2.2 The Fibonacci sequence

We calculated the first **25** numbers in the Fibonacci sequence. The data are shown in Figure 1. (See the code listing in Section 3.) Note that this figure number (as well as the table number above) is automatically generated.

3 Methods

The following code was used in section 2.2, to calculate and plot the Fibonacci sequence:

```
# calculate the first `n` Fibonacci numbers
fibonacci <- function(n) {
  stopifnot(n > 2)
  ar <- rep(0, n)
  ar[1] <- f1 <- 0
  ar[2] <- f2 <- 1
  for (i in 3:n) {
    f3 <- f1 + f2
    ar[i] <- f3
    f1 <- f2
    f2 <- f3
  }
  return (ar)
```

```
}

ncalc <- 25
f <- fibonacci(ncalc)
df <- data.frame(1:ncalc, f)
names(df) <- c("x","y")

div <- 1000

p <- ggplot(df) +
  geom_line(aes(x, y/div), color = "red") +
  geom_point(aes(x, y/div), color = "blue") +
  scale_y_continuous(limits = c(0, max(df$y)/div)) +
  xlab("") +
  ylab("Fibonacci No. (thousands)") +
  theme_minimal()
p
```

References

Argelaguet, Ricard, Anna S. E. Cuomo, Oliver Stegle, and John C. Marioni. 2021. "Computational Principles and Challenges in Single-Cell Data Integration." *Nature Biotechnology*, May, 1–14. https://doi.org/10.1038/s41587-021-00895-7.

Fisch, K. M., T. Meissner, L. Gioia, J.-C. Ducom, T. M. Carland, S. Loguercio, and A. I. Su. 2015. "Omics Pipe: A Community-Based Framework for Reproducible Multi-Omics Data Analysis." *Bioinformatics* 31 (11): 1724–28. https://doi.org/10.1093/bioinformatics/btv061.

Lê Cao, Kim-Anh, Al J. Abadi, Emily F. Davis-Marcisak, Lauren Hsu, Arshi Arora, Alexis Coullomb, Atul Deshpande, et al. 2021. "Community-Wide Hackathons to Identify Central Themes in Single-Cell Multi-Omics." *Genome Biology* 22 (1): 220. https://doi.org/10.1186/s13059-021-02433-9.

Perkel, Jeffrey M. 2020. "Streamline Your Writing and Collaborations with These Reference Managers." *Nature* 585 (7823): 149–50. https://doi.org/10.1038/d41586-020-02491-2.