

R Markdown Tutorial

Rosemary Hartman and Sarah Perry

4/15/2020

R Markdown

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>.

Why use R Markdown?

- Break up your code into “chunks” that can be run individually.
- Save code and output together in one document.
- Share code and results with others.
- Easily create automated reports.

Structure of an R Markdown document

- Metadata
- Markdown
- Code chunks

Metadata

Rmd files include a metadata section, also known as a “YAML header” (typically located at the top of the file) that can specify (among other things) the title, author, and date of the document. Metadata adheres to the YAML format and is delimited by lines containing three dashes (---). If you create the document with R Studio, it creates most of it for you.

The metadata is also where you can specify some of the output parameters, including font size, font style, output format, whether to have a table-of-contents, etc.

The YAML header of this document looks like this:

```
---
title: "R Markdown Tutorial"
author: "Rosemary Hartman and Sarah Perry"
date: "4/15/2020"
output:
  pdf_document: default
  html_document:
    toc: true
    toc_float: true
    code_folding: show
---
```

Markdown

The “markdown” part of the document, is where you can put all your comments, but unlike commenting in an R script, you don’t have to begin every line with a hashtag and you can do fancier formatting.

We can look at R Studio’s R Markdown Cheat Sheet for help.

Lots more formatting help in this guide: <https://github.com/adam-p/markdown-here/wiki/Markdown-Here-Cheatsheet> Another good resource: <https://bookdown.org/yihui/rmarkdown-cookbook/>

Code Chunks

```
summary(cars)
```

```
##      speed      dist
##  Min.   : 4.0    Min.   :  2.00
##  1st Qu.:12.0    1st Qu.: 26.00
##  Median :15.0    Median : 36.00
##  Mean   :15.4    Mean   : 42.98
##  3rd Qu.:19.0    3rd Qu.: 56.00
##  Max.   :25.0    Max.   :120.00
```

The part in front of the code chunk between the curly braces is where you can specify your chunk options. Some important ones:

- `echo=True` - Print the code
- `echo=False` - Don’t print the code (but do evaluate it)
- `eval=False` - don’t evaluate the code (but do print it)

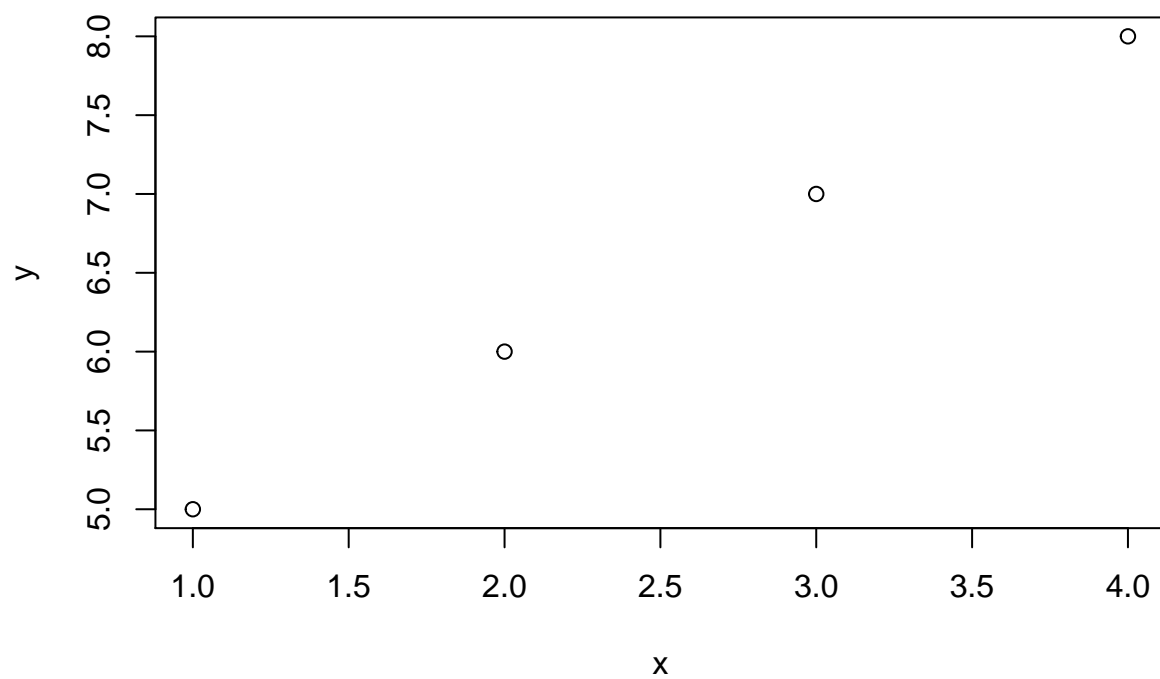
When you are creating reports, the chunk options are also where you can define figure size, font alignment, etc.

See all options here: <https://yihui.org/knitr/options/>

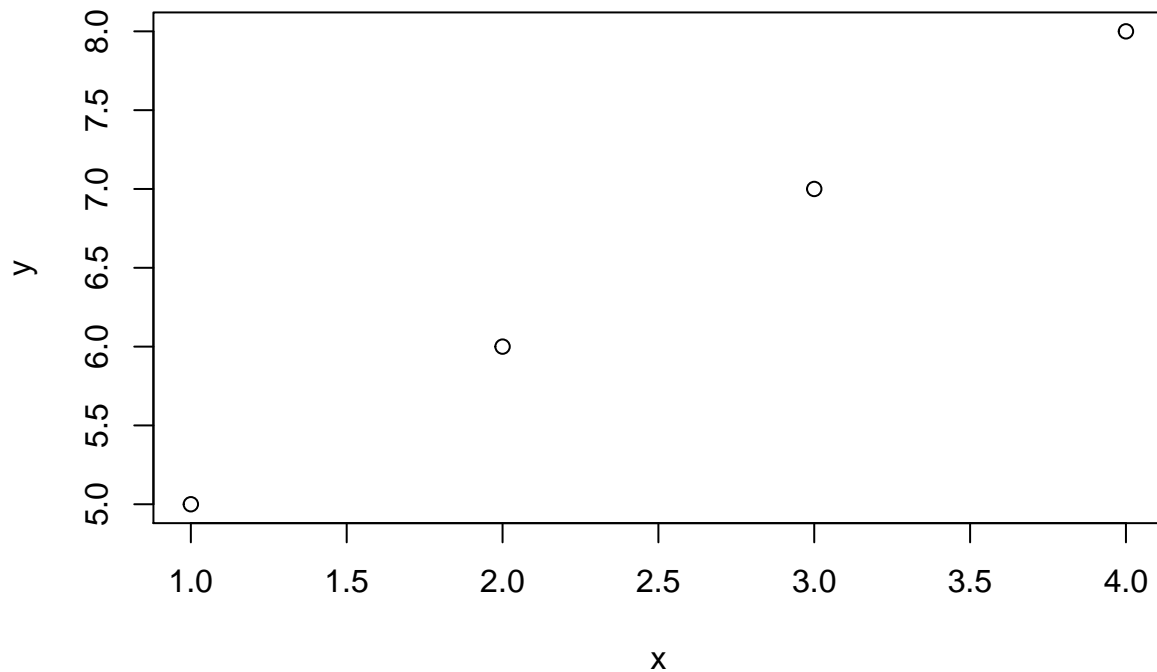
So, if I wanted to create a report to send to my friend for help with my code, I might write a chunk that says:

```
#I need help with my code!
x = c(1,2,3,4)
y = c(5,6,7,8)

plot(x,y)
```



However, if I wanted to just send the graph to my supervisor, I would do it:



Publishing

R Studio makes it really easy to turn your script into a real report. Just hit the **knit** button on the top of the document. The drop-down menu gives you the option to publish it as HTML, PDF, or Word. HTML works really well. PDF and Word isn't quite as nice.

If you want to publish as a PDF, you have to install LaTeX, which is a typesetting system (<https://www.latex-project.org/>). The package `tinytex` will install it for you.

```
#Load the tinytex R package  
install.packages('tinytex')  
  
#Now install LaTeX. This will take a while  
tinytex::install_tinytex()
```

Now you can knit to PDF

More advanced stuff

To create really nice automated reports, you can load data automatically and produce the graph right in the document.

Notice that I have `echo=FALSE` and `warning=FALSE` so that you just get the graph, not the code to produce the graph in the final report.

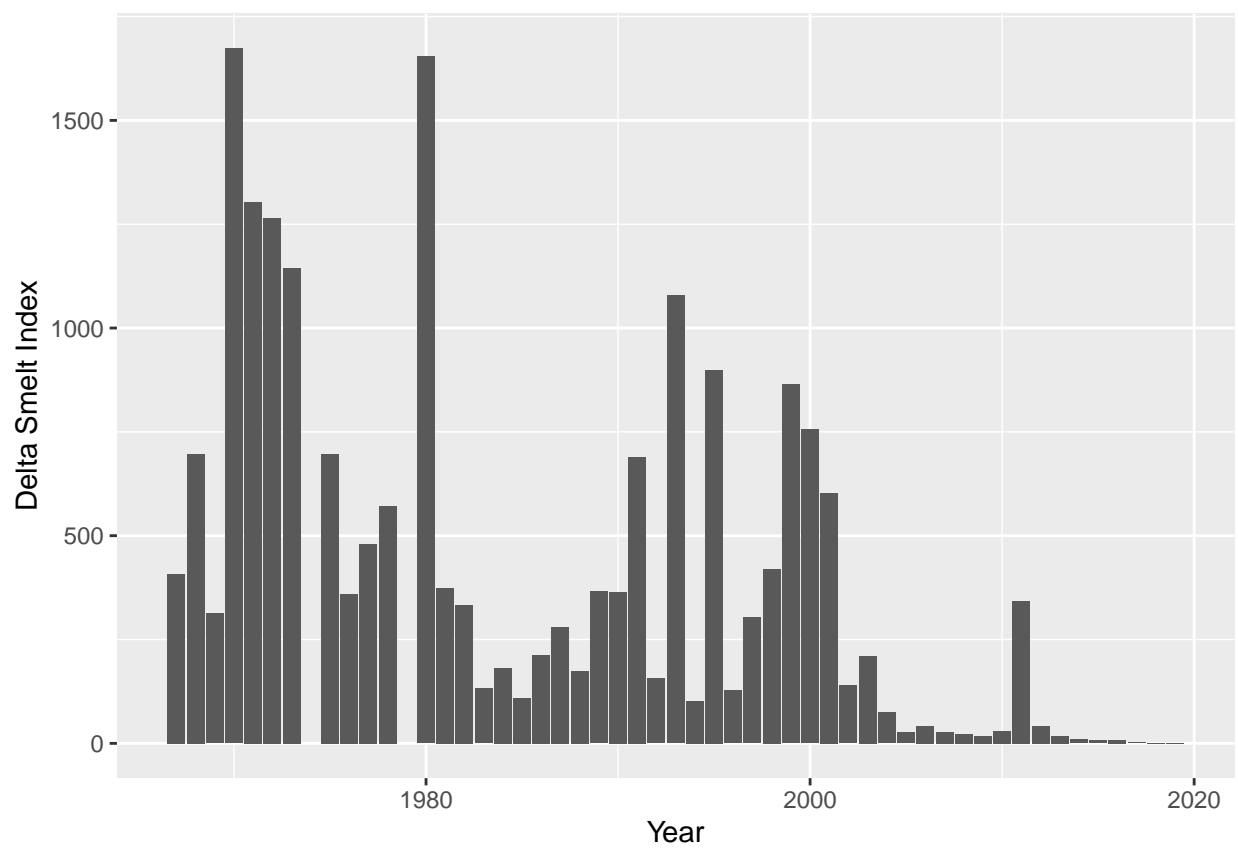
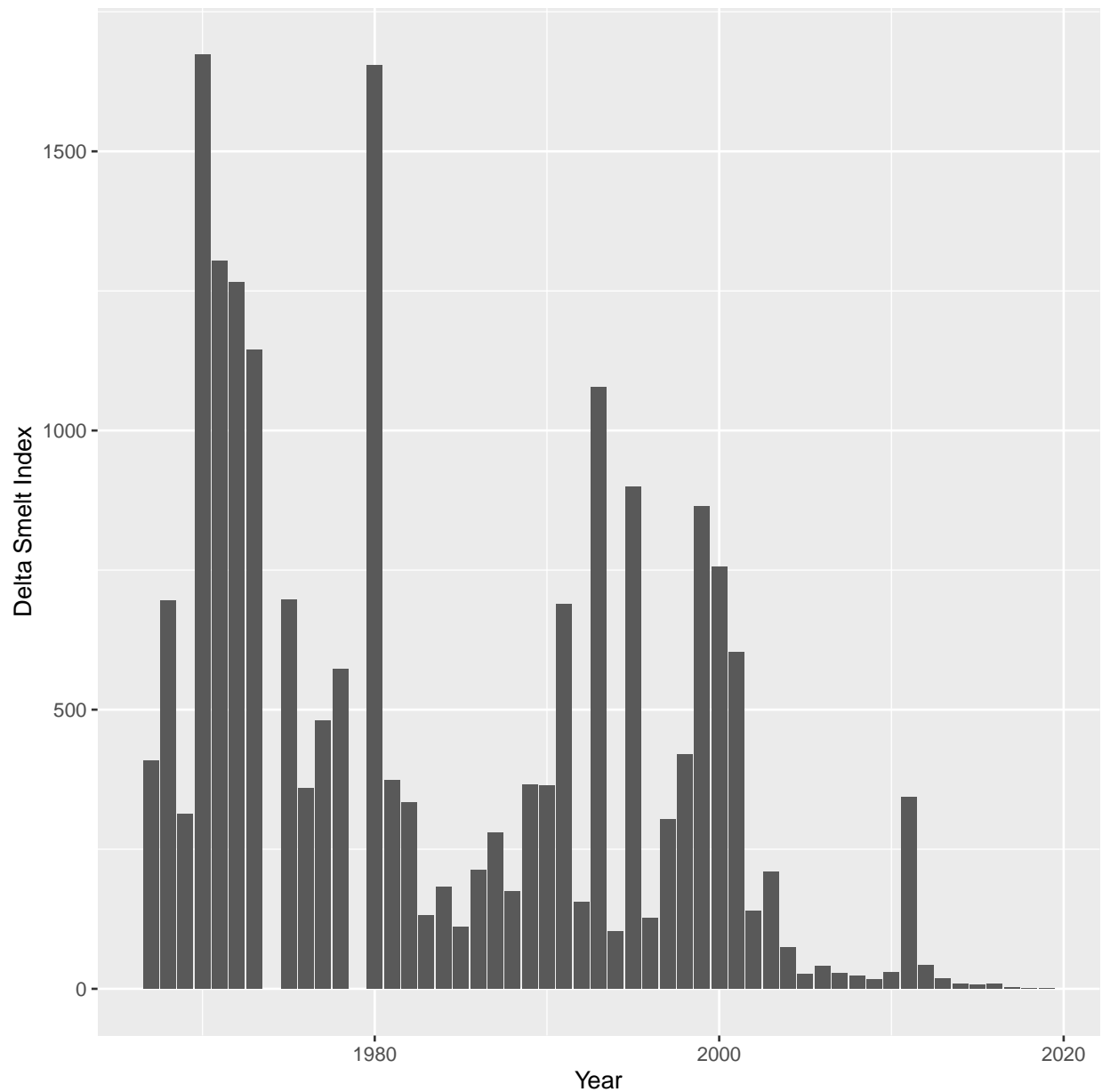


Figure 1: Graph of FMWT Delta Smelt Index

You can modify the position, height, and width of the plot by using the `fig.align`, `fig.height`, and `fig.width` chunk options. Here, I have set `fig.align = 'center'`, `fig.width = 7`, `fig.height = 7`.



In addition, you can use the **Plotly** package to create interactive, publication-quality graphs, as shown below.

```
library(plotly)

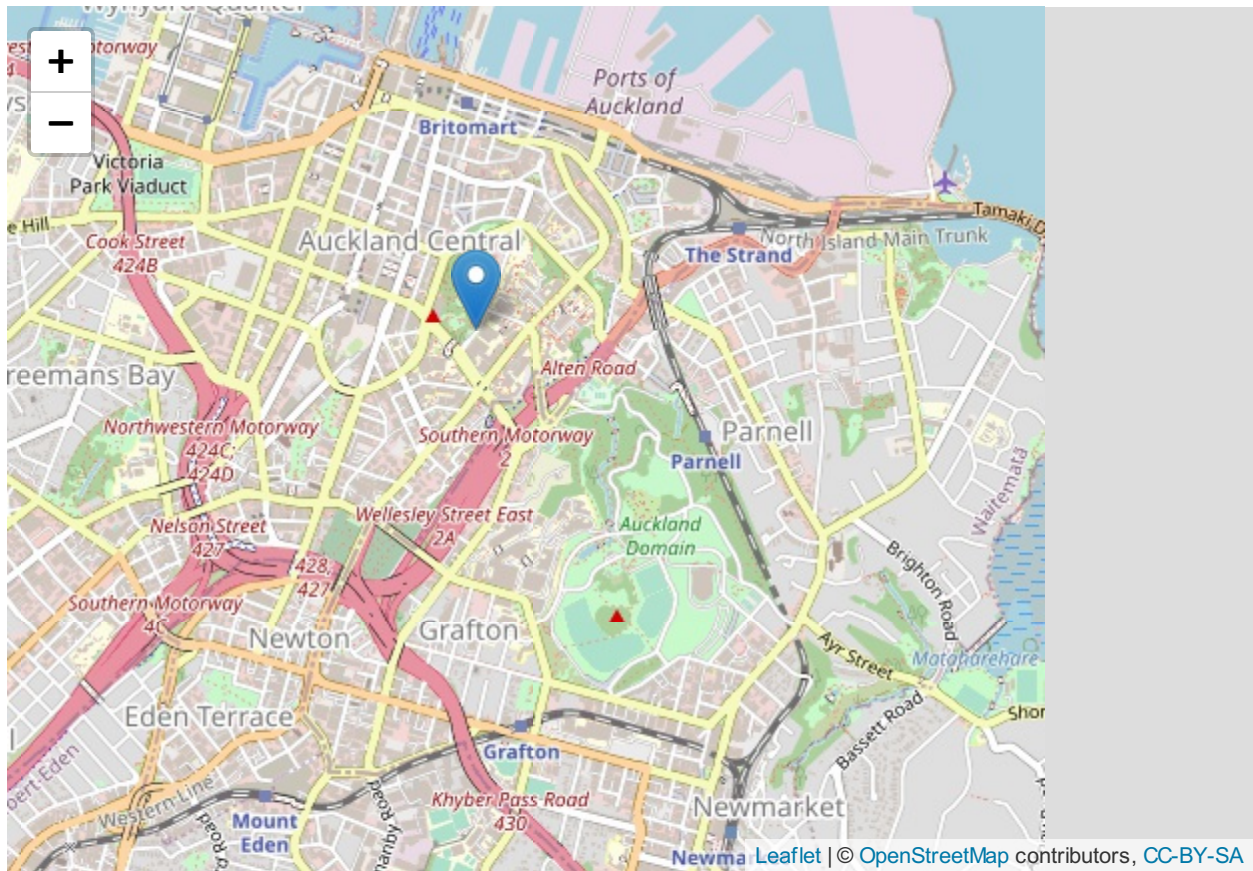
fig <- plot_ly(data = iris, x = ~Sepal.Length, y = ~Petal.Length)

fig
```

Leaflet, meanwhile, can be used to make interactive maps.

```
library(leaflet)

m <- leaflet() %>%
  addTiles() %>% # Add default OpenStreetMap map tiles
  addMarkers(lng=174.768, lat=-36.852, popup="The birthplace of R")
m # Print the map
```



For example, I made the IEP Seasonal Monitoring Report using RMarkdown and Bookdown. <https://interagencyecologicalprogram.github.io/Status-and-Trends/>

Other cool things you can do include

plotly

leaflet

GitHub pages