

Literate Statistical Programming with knitr

Problems, Problems

- Authors must undertake considerable effort to put data/results on the web
- Readers must download data/results individually and piece together which data go with which code sections, etc.
- Authors/readers must manually interact with websites
- There is no single document to integrate data analysis with textual representations; i.e. data, code, and text are not linked

Literate Statistical Programming

- Original idea comes from Don Knuth
- An article is a stream of text and code
- Analysis code is divided into text and code “chunks”
- Presentation code formats results (tables, figures, etc.)
- Article text explains what is going on
- Literate programs are weaved to produce human readable documents and tangled to produce machine readable documents

Literate Statistical Programming

- Literate programming is a general concept. We need
 - – A documentation language
 - – A programming language
- The original Sweave system developed by Friedrich Leisch used LaTeX and R
- knitr supports a variety of documentation languages

How Do I Make My Work Reproducible?

- Decide to do it (ideally from the start)
- Keep track of things, perhaps with a version control system to track snapshots/changes
- Use software whose operation can be coded
- Don't save output
- Save data in non-proprietary formats

What is knitr?

- An R package written by Yihui Xie (while he was a grad student at Iowa State)
- Available on CRAN
- Supports RMarkdown, LaTeX, and HTML as documentation languages
- Can export to PDF, HTML
- Built right into RStudio for your convenience

Requirements

- A recent version of R
- A text editor (the one that comes with RStudio is okay)
- Some support packages also available on CRAN
- Some knowledge of Markdown, LaTeX, or HTML
- We will use Markdown here

What is Markdown?

- A simplified version of “markup” languages
- No special editor required
- Simple, intuitive formatting elements
- Complete information available at <http://goo.gl/MUt9i5>

What is knitr Good For?

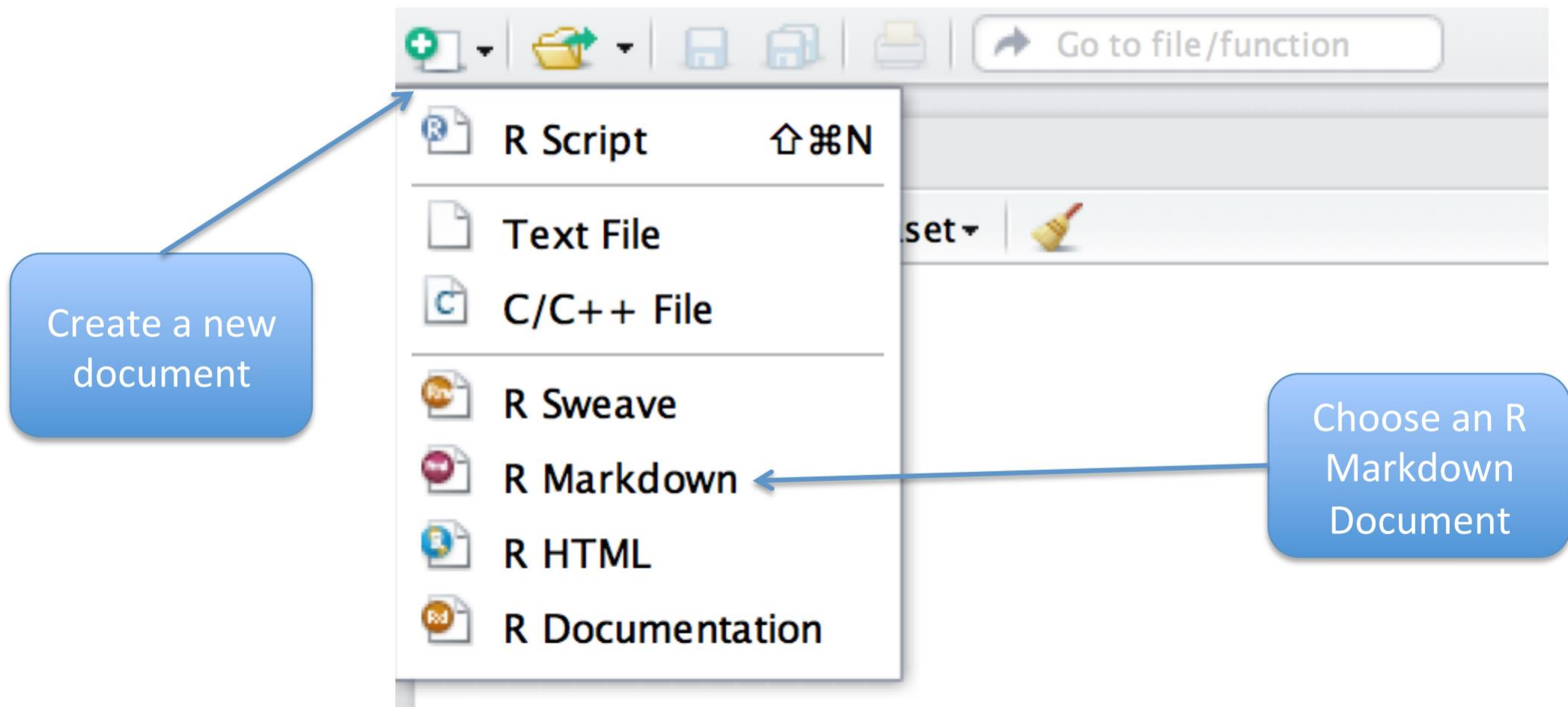
- Manuals
- Short/medium-length technical documents
- Tutorials
- Reports (esp. if generated periodically)
- Data preprocessing documents/summaries

What is knitr NOT Good For?

- Very long research articles
- Complex time-consuming computations
- Documents that require precise formatting



My First knitr Document



My First knitr Document

```
1 My First knitr Document
```

```
2
```

```
3
```

```
4 This is some text (i.e. a "text chunk").
```

```
5
```

```
6 Here is a code chunk
```

```
7
```

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

```
8
```

```
set.seed(1)
```

```
9
```

```
x <- rnorm(100)
```

```
10
```

```
mean(x)
```

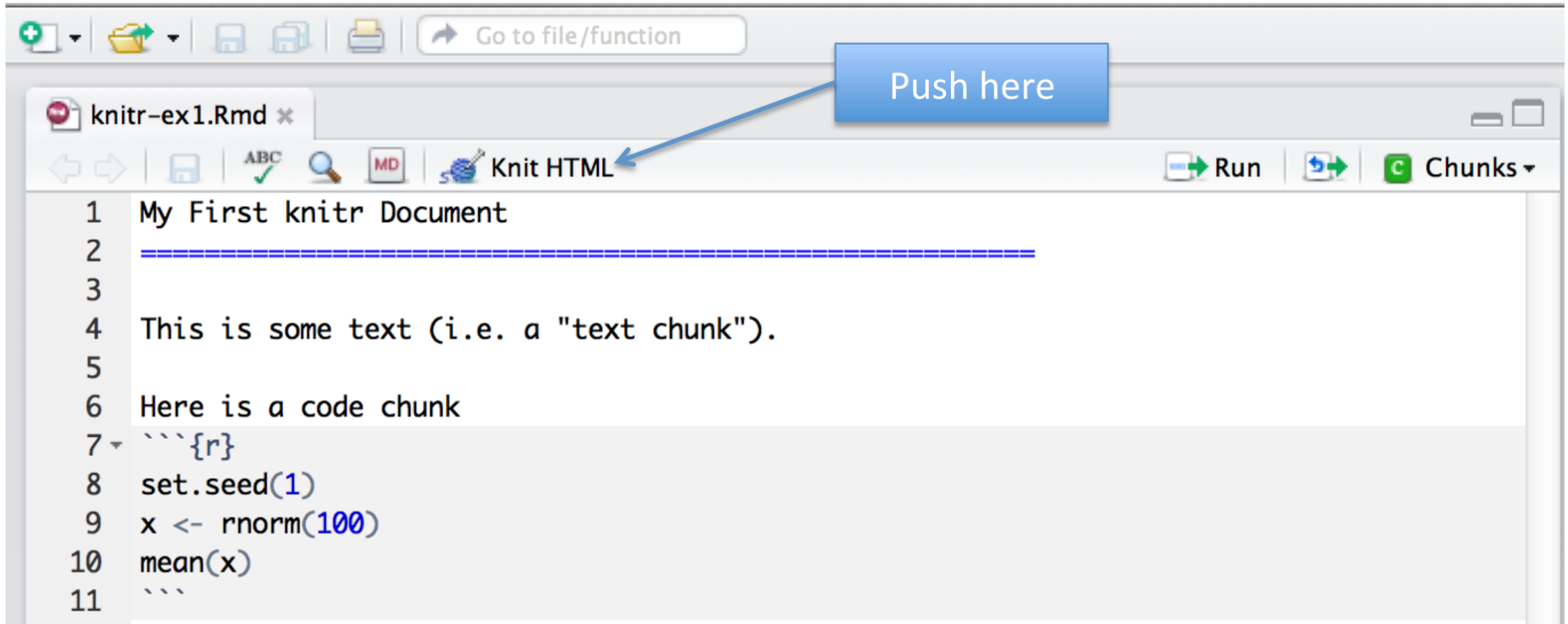
```
11
```

```
```\n
```

Start of code chunk

End of code chunk

Processing a knitr Document



HTML Output

My First knitr Document

This is some text (i.e. a “text chunk”).

Here is a code chunk

```
set.seed(1)
x <- rnorm(100)
mean(x)
```

Code input

```
## [1] 0.1089
```

Numerical output

What knitr Produces: Markdown

RMarkdown Document

```
1 My First knitr Document
2 =====
3
4 This is some text (i.e. a "text chunk").
5
6 Here is a code chunk
7 ```{r}
8 set.seed(1)
9 x <- rnorm(100)
10 mean(x)
11 ```
```

Code is
echoed

Markdown Document (generated)

```
1 My First knitr Document
2 =====
3
4 This is some text (i.e. a "text chunk").
5
6 Here is a code chunk
7
8 ```r
9 set.seed(1)
10 x <- rnorm(100)
11 mean(x)
12 ```
13
14 ```
15 ## [1] 0.1089
16 ```
```

Result of
evaluating R
code

A Few Notes

- knitr will fill a new document with filler text; delete it
- Code chunks begin with ```` `{r}` and end with ``````
- All R code goes in between these markers
- Code chunks can have **names**, which is useful when we start making graphics

```
````{r firstchunk}  
R code goes here
````
```
- By default, code in a code chunk is echoed, as will the results of the computation (if there are results to print)

Processing of knitr Documents (what happens under the hood)

- You write the RMarkdown document (.Rmd)
- knitr produces a Markdown document (.md)
- knitr converts the Markdown document into HTML (by default)
- .Rmd → .md → .html
- You should NOT edit (or save) the .md or .html documents until you are finished

Another Example

```
# My First knitr Document  
Roger D. Peng
```

Level 1 heading

```
## Introduction
```

Level 2 heading

This is some text (i.e. a "text chunk"). Here is a code chunk.

```
```${r simulation,echo=FALSE}  
set.seed(1)
x <- rnorm(100)
mean(x)
```
```

Do not echo code




Output

My First knitr Document

Introduction

This is some text (i.e. a “text chunk”). Here is a code chunk.

```
## [1] 0.1089
```



Incorporating Graphics

```
## Introduction
```

Let's first simulate some data.

```
```{r simulatedata,echo=TRUE}  
x <- rnorm(100); y <- x + rnorm(100, sd = 0.5)
```
```

Here is a scatterplot of the data.

```
```{r scatterplot,fig.height=4}  
par(mar = c(5, 4, 1, 1), las = 1)
plot(x, y, main = "My Simulated Data")
```
```



Adjust figure height

What knitr Produces in HTML

```
<body>  
<h2>Introduction</h2>
```

```
<p>Let's first simulate some data.</p>
```

```
<pre><code class="r">x &lt;- rnorm(100)  
y &lt;- x + rnorm(100, sd = 0.5)  
</code></pre>
```

```
<p>Here is a scatterplot of the data.</p>
```

```
<pre><code class="r">par(mar = c(5, 4, 1, 1), las = 1)  
plot(x, y, main = "My Simulated Data")  
</code></pre>
```

```
<p><img src="data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAAFgAAAEgCAYAAABYRWE9AAAEJG  
lDQ1BJQ0MgUHJvZmlsZQAABGFiVd9v21QUPolVUqQWPyBYR4eKxa9VU1u5GxqtXgZJk6XtShal6dgqJ0Q6N4m  
pGwfb6baqT3uBNwb8AUDZAw9IPCENBmJ72fbAtElThyqqSUh76MQPISbtBVXhu3ZiJ1PEXPX6yznf0ec7517b  
RD1fabWaGVWILquunc8klZ0nFpSeTYrSs9RLA9Sr6U4tkcvNEi7BFff06+EdigjL7ZHu/k72I796i9zRiSJPw  
G4VHX0Z+AxRzNRrtksUvwf7+Gm3BtzzHPDTNgQCqwKXfZwSeNHJz10IT8JjtAq6xWtCLwGPLzYZi
```

Image is embedded
in HTML

Incorporating Graphics

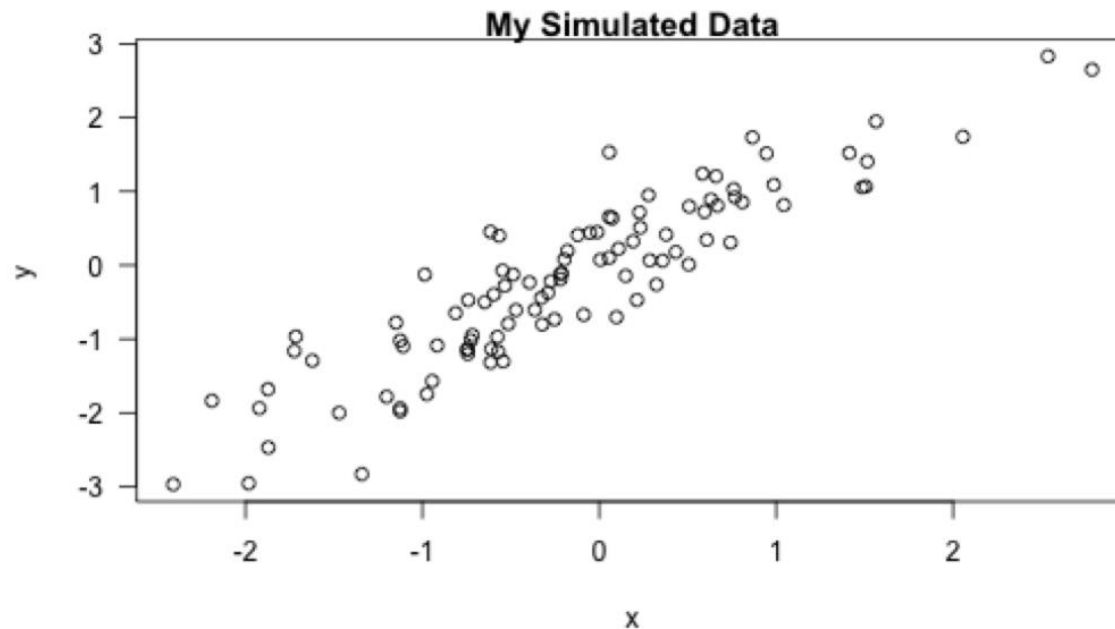
Introduction

Let's first simulate some data.

```
x <- rnorm(100)  
y <- x + rnorm(100, sd = 0.5)
```

Here is a scatterplot of the data.

```
par(mar = c(5, 4, 1, 1), las = 1)  
plot(x, y, main = "My Simulated Data")
```



Making Tables with xtable

Introduction

```
```{r fitmodel}  
library(datasets)
data(airquality)
fit <- lm(Ozone ~ Wind + Temp + Solar.R, data = airquality)
```
```

Here is a table of regression coefficients.

```
```{r showtable,results="asis"}  
library(xtable)
xt <- xtable(summary(fit))
print(xt, type = "html")
```
```

Making Tables with xtable

Introduction

```
library(datasets)
data(airquality)
fit <- lm(Ozone ~ Wind + Temp + Solar.R, data = airquality)
```

Here is a table of regression coefficients.

```
library(xtable)
xt <- xtable(summary(fit))
print(xt, type = "html")
```

| | Estimate | Std. Error | t value | Pr(> t) |
|-------------|-----------------|-------------------|----------------|---------------------|
| (Intercept) | -64.3421 | 23.0547 | -2.79 | 0.0062 |
| Wind | -3.3336 | 0.6544 | -5.09 | 0.0000 |
| Temp | 1.6521 | 0.2535 | 6.52 | 0.0000 |
| Solar.R | 0.0598 | 0.0232 | 2.58 | 0.0112 |

Setting Global Options

- Sometimes we want to set options for every code chunk that are different from the defaults
- For example, we may want to suppress all code echoing and results output
- We have to write some code to set these global options

Setting Global Options

```
## Introduction
```

```
``{r setoptions,echo=FALSE}  
opts_chunk$set(echo = FALSE, results = "hide")  
``
```

Set default to NOT
echo code

First simulate data

```
``{r simulatedata,echo=TRUE}  
x <- rnorm(100); y <- x + rnorm(100, sd = 0.5)  
``
```

Override default

Here is a scatterplot of the data.

```
|``{r scatterplot,fig.height=4}  
par(mar = c(5, 4, 1, 1), las = 1)  
plot(x, y, main = "My Simulated Data")  
``
```

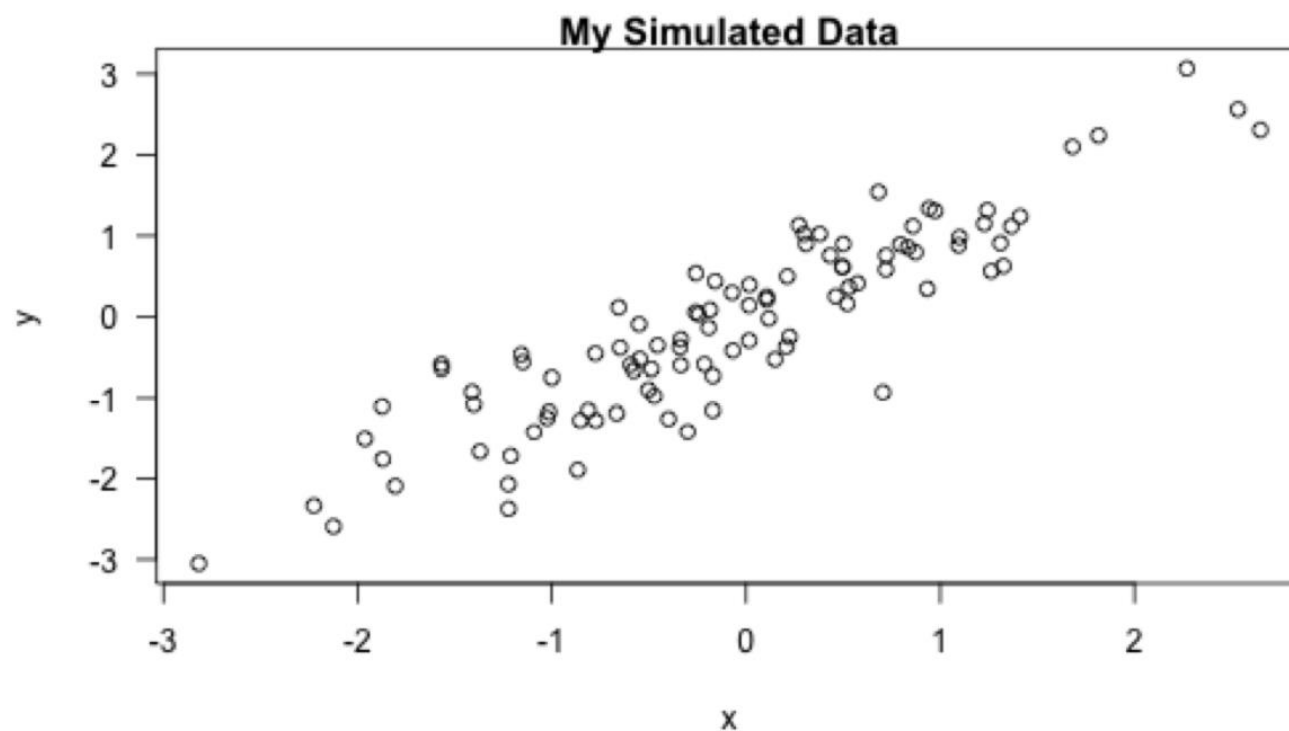
Don't echo code here

Introduction

First simulate data

```
x <- rnorm(100)
y <- x + rnorm(100, sd = 0.5)
```

Here is a scatterplot of the data.



<https://www.rstudio.com/resources/cheatsheets/>

R Markdown : : CHEAT SHEET

What is R Markdown?

And files - An R Markdown document (R Markdown) is a blend of your research, its analysis, the code that generated the results, and the results themselves. It is a single file that can be used to generate a report, a presentation, or a website.

Reproducible Research - At the core of a workflow is the idea of reproducible research. This means that the code that generated the results can be used to generate the results again, and the results will be the same.

Dynamic Documents - You can create a document that updates its content automatically. This is useful for creating reports that update as new data is added, or for creating presentations that update as you add new slides.

Workflow

1. Open a new R Markdown document.
2. Write your R code and Markdown text.
3. Preview the document to see how it will look when rendered.
4. Publish the document to a website or save it as a PDF or HTML file.

render

Use `render()` to render the document to a specific format. The default format is HTML.

```
render(format = "html", output_file = "report.html")
```

Options for `format`:

- `html`: HTML document
- `pdf`: PDF document
- `presentation`: Beamer presentation
- `website`: Static website

Embed code with knitr syntax

Use `<pre>` and `</pre>` to embed R code in a document. Use `<code>` and `</code>` to embed inline code.

IMPORTANT CHUNK OPTIONS

| Option | Description |
|------------------------|---|
| <code>code</code> | Whether to execute the code in the chunk (default: <code>TRUE</code>) |
| <code>comment</code> | Whether to include the comments in the output (default: <code>TRUE</code>) |
| <code>echo</code> | Whether to echo the code in the output (default: <code>TRUE</code>) |
| <code>highlight</code> | Whether to highlight the code in the output (default: <code>TRUE</code>) |
| <code>message</code> | Whether to display messages in the output (default: <code>TRUE</code>) |
| <code>warning</code> | Whether to display warnings in the output (default: <code>TRUE</code>) |

GLOBAL OPTIONS

Set global options for the document using `knitr::opts_chunk$set()`.

```
knitr::opts_chunk$set(echo = TRUE, message = FALSE, warning = FALSE)
```

Pandoc's Markdown

Set render options with YAML

Use the following YAML code to set render options for the document.

```
---
title: "My Document"
author: "John Doe"
date: "2023-01-01"
output: "html_document"
---
```

Table Suggestions

Use the following code to create a table with suggestions.

```
knitr::kable(
  data.frame(
    name = c("John", "Jane", "Bob", "Alice"),
    age = c(30, 25, 35, 28)
  )
)
```

Citations and Bibliography

Use the following code to create a citation and bibliography.

```
knitr::cite(
  "knitr",
  "rmarkdown",
  "pandoc"
)
```



Rmd examples

Let's work on some Rmd examples.

