Introduction to R Markdown

DIME Analytics

Presented by Luis Eduardo San Martin

The World Bank | WB Github

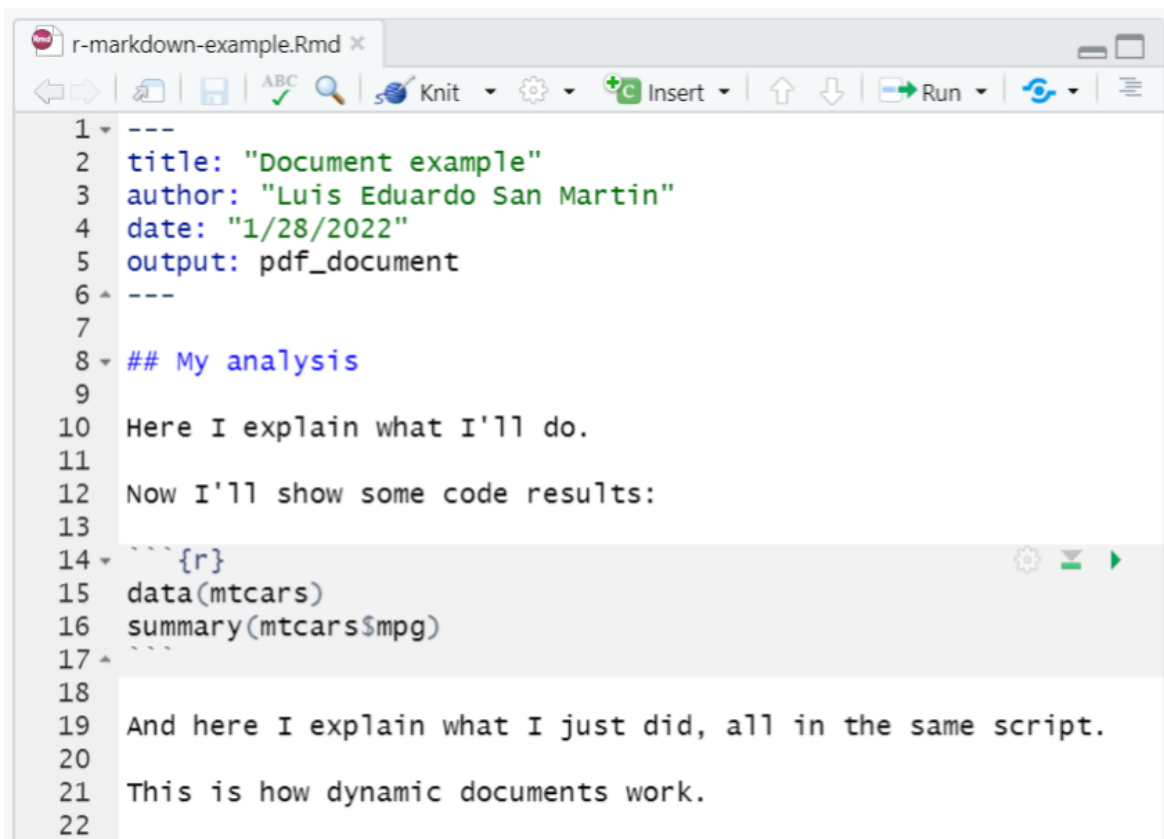February 2022

Table of contents

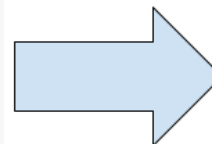# Dynamic documents

Dynamic documents

- Dynamic documents are documents that include both text and code outputs. They are generated by a script and are updated automatically every time the script runs

- In the context of this presentations, all code outputs will be R outputs

Dynamic documents

- Code and documentation is produced together

Why use dynamic documents?

- Increased research transparency. Documents are fully reproducible
- No more copying and pasting outputs from R to document editor
- Nice option for simple documents that don't require a lot of formatting
- Can include code snippets or code examples

Knitting R Markdown documents

Knitting R Markdown documents

- R markdown combines R code, rendered outputs and text.
- The code and outputs follow the syntax of R, while the text follows markdown's syntax
- Exports the documents to PDF, Word, HTML

Knitting R Markdown documents

Exercise 1

1. Go to our R trainings repository
2. In the files panel, download `Introduction to R Markdown...` > `r-markdown-template.Rmd`
3. Open this file in RStudio
4. Click on `Knit` > `Knit to PDF`

Knitting R Markdown documents

- Knitting a file to PDF means that RStudio created a PDF file from an R Markdown file
- You'll now see this PDF file in the folder where you saved `markdown-template.Rmd`

Name	Date modified	Type
Code	1/28/2022 2:51 PM	File folder
DataSets	8/24/2020 3:42 PM	File folder
Output	4/5/2021 4:37 PM	File folder
descriptive-statistics.R	1/13/2022 1:29 PM	R File
r-markdown-template.pdf	2/1/2022 10:51 PM	Adobe Acrobat Docu...
r-markdown-template.Rmd	2/1/2022 10:36 PM	RMD File

Markdown

Markdown

- As we indicated previously, the text part of R Markdown follows the syntax of Markdown
- Markdown is a lightweight markup language. It's similar to Latex or HTML, but simpler
- Markdown was designed to be easily readable while allowing to define text format

Markdown - Headers

- Headers in markdown are preceded by pound (`#`) symbols
- Additional pound symbols denote a lower level in the headers hierarchy

```
# This is a header
## Subheader 1
### Subheader 2
#### Subheader 3
```
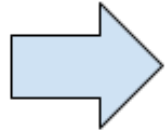
⟹

This is a header

Subheader 1

Subheader 2

Subheader 3

- Text not preceded by special symbols is a regular pragraph.
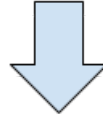
Markdown - Text emphasis

- Emphasized text is enclosed by special symbols.
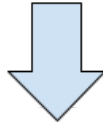
- Markdown allows to use ordered and unordered lists.

```
## Lists

Ordered lists:

1. Include a number and a dot before every item
2. Also remember to include a blank line before the beginning of the list
1. The actual number does not matter, the item will have the correct order
number

Unordered lists:

* You can use an asterisk
+ Or a plus symbol
- Or a minus symbol
```
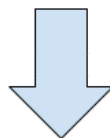
Lists

Ordered lists:

1. Include a number and a dot before every item
2. Also remember to include a blank line before the beginning of the list
3. The actual number does not matter, the item will have the correct order number

Unordered lists:

- You can use an asterisk
- Or a plus symbol
- Or a minus symbol

Markdown - Links

- Lastly, we can also include links as text in Markdown.
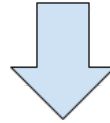
R Code

Including R code

- R code in R Markdown goes inside **fenced code blocks**, as the one below

```{r}
# Your R code goes here
```

Including R code

- You can type the fenced code blocks directly or go to `Insert` > `R` in the script panel of RStudio to create a new block
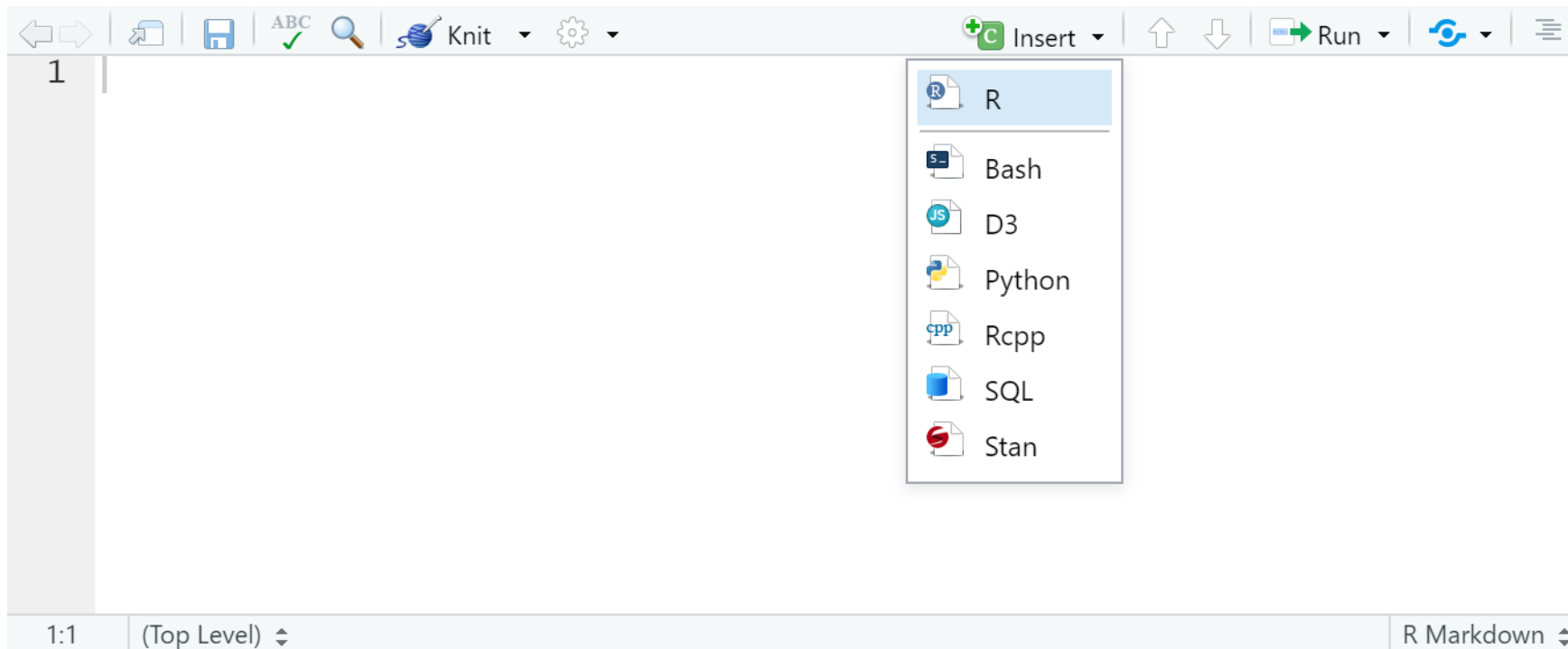
Including R code

Exercise 2

Include the summary of a variable in the document.

1. Create a header named `R Code` at the bottom of `markdown-template.Rmd`
2. Create a new fenced code block where you load the sample dataset `mtcars`
   - `mtcars` is a built-in dataset
   - load it with: `data(mtcars)`
3. Inside the same block, get the summary of the variable `mpg` with
   `summary(mtcars$mpg)`
4. Knit the document to PDF and check how this is rendered in the resulting file

Including R code

```
# R code
```
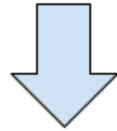
```r
```{r}
data(mtcars)
summary(mtcars$mpg)
```
```

R code

```r
data(mtcars)
summary(mtcars$mpg)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   10.40   15.43   19.20   20.09   22.80   33.90
```

Including R code

- Note that the output echoes both the code and the output
- What if we wanted to include the output but not the code?
- We use argument `echo = FALSE` in the fenced code block for that
- Code block arguments are separated by commas inside the curly brackets, as in:

```
```{r, echo = FALSE}
data(mtcars)
summary(mtcars$mpg)
```
```

Including R code

Exercise 3

Omit the code when knitting R code.

1. Add the option `echo = FALSE` to the fenced code block created in exercise 2
2. Knit the document into PDF and check the difference

Including R code

```
# R code
```

```{r, echo = FALSE}
data(mtcars)
summary(mtcars$mpg)
```
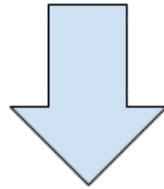
R code

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   10.40   15.43   19.20   20.09   22.80   33.90
```

Including R code

- To include R code but not the output, we can use the option `eval = FALSE`

```
```{r, eval = FALSE}
data(mtcars)
summary(mtcars$mpg)
```
```

Including R code

```
# R code
```

```
```{r, eval = FALSE}
data(mtcars)
summary(mtcars$mpg)
```
```

R code

```
data(mtcars)
summary(mtcars$mpg)
```

R Plots

Including R plots

- Adding R plots is similar to adding R code
- Include the code producing the plot in a fenced block
- The block option `echo = FALSE` is useful when we only want to include the plot but not the code producing it

Including R plots

Exercise 4

Include an R plot in your document.

1. Create a new header named `R Plots` at the bottom of `markdown-template.Rmd`
2. Create a new fenced code block with the option `echo` set to `FALSE`
3. Add the following code inside the new block:

```
plot(mtcars$wt,
     mtcars$mpg,
     main = "Plot example",
     xlab = "Car weight",
     ylab = "Miles per gallon")
```

Including R plots

```
# R Plots
```

```{r, echo = FALSE}
plot(mtcars$wt,
     mtcars$mpg,
     main = "Plot example",
     xlab = "Car weight",
     ylab = "Miles per gallon")
```

R Plots
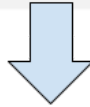
Plot example
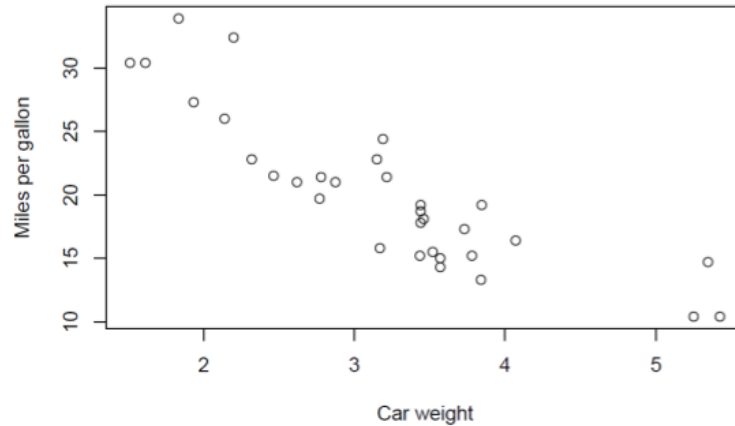
Inline code

Including code inline

- Inline code is enclosed by backtick followed by an r (`` `r ``) and a single backtick (`` ` ``)

- For example:

```
The mean of mpg is `r mean(mtcars$mpg)`.
```

- Will be rendered as:

```
The mean of mpg is 20.090625.
```

Including code inline

Exercise 5

1. Create a new header named `Inline code` at the bottom of `markdown-template.Rmd`
2. Add an unordered list with the items:
   - The number of elements in mtcars is:
   - The mean of weight is:
   - The standard deviation is:
3. Use the functions `nrow()`, `mean()`, and `sd()` in the corresponding list items so that the correct values are rendered. Remember you can select the weight variable like this: `mtcars$wt`

Including code inline

```
# Inline code

- The number of elements in mtcars is: `r nrow(mtcars)`
- The mean of weight is: `r mean(mtcars$wt)`
- The standard deviation is: `r sd(mtcars$wt)`
```
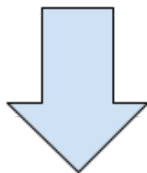
Inline code

- The number of elements in mtcars is: 32
- The mean of weight is: 3.21725
- The standard deviation is: 0.9784574

Annex

Annex - Opening a new R Markdown in R Studio

- Go to `File` > `New File` > `R Markdown`
- You can register the author name and the document title. This can be changed later if needed
- You can also define the default output format (HTML, PDF, Word). This can also be changed later
- Selecting `OK` will generate a template with document sections and code blocks that you can modify
- Selecting `Create Empty Document` will ignore the author, title, and output format registered and will result in a completely blank R Markdown document

Annex - Author, title, and output type

- The section enclosed in `---` at the beginning of the document can contain the author, title, and default output format
- You can add the author and document title with `author: NAME` and `title: TITLE`
- You can also change the default output format. Some options are:
  - `output: html_document`
  - `output: pdf_document`
  - `output: word_document`
  - `output: beamer_presentation`

Annex - Including images

- The Markdown syntax to include images is:

`![Image name](path/to/image)`

- For example:

`![R logo](img/r-markdown/r-logo.jpg)`

- Renders: