

## 03-bis - L<sup>A</sup>T<sub>E</sub>Xbis

Ottavia M. Epifania  
University of Padova

ottavia.epifania@unipd.it

Corso RMarkdown

# Il perché di questa lezione

Quando ho preparato le slide per L<sup>A</sup>T<sub>E</sub>Xusando Sweave, tutto sembrava andare per il meglio

# Il perché di questa lezione

Quando ho preparato le slide per L<sup>A</sup>T<sub>E</sub>X usando Sweave, tutto sembrava andare per il meglio

Poi, gli aggiornamenti e:



- **Sweave** vi obbliga a usare L<sup>A</sup>T<sub>E</sub>X
- Non è facile farlo funzionare, specie se volete aggiungere dei pezzi di codice e i risultati delle vostre analisi
- Per usare **Sweave**, tanto vale usare direttamente L<sup>A</sup>T<sub>E</sub>X
- Vi lascio comunque le slide per conoscenza generale

I file si aprono esattamente come prima. L'unica cosa che modifichiamo è lo YAML

## Vantaggi

- Permette di usare L<sup>A</sup>T<sub>E</sub>X senza realmente saper usare L<sup>A</sup>T<sub>E</sub>X
- Ha tutta la flessibilità (e bellezza) dei file L<sup>A</sup>T<sub>E</sub>X
- Con un minimo sforzo si ottengono dei risultati di tutto rispetto

I file si aprono esattamente come prima. L'unica cosa che modifichiamo è lo YAML

## Vantaggi

- Permette di usare  $\text{\LaTeX}$  senza realmente saper usare  $\text{\LaTeX}$
- Ha tutta la flessibilità (e bellezza) dei file  $\text{\LaTeX}$
- Con un minimo sforzo si ottengono dei risultati di tutto rispetto

## Svantaggi

- Se vogliamo ottenere un HTML non è la soluzione che fa per noi
- Essendo un ibrido tra RMarkdown e  $\text{\LaTeX}$  bisogna fare attenzione a come comunicano
- Le soluzioni che vanno bene in  $\text{\LaTeX}$  non sempre vanno bene in RMarkdown

# Documenti di testo

# Lo YAML

```
---
```

```
title: "Il mio primo LaTeX"
author: "Ottavia M. Epifania"
date: ''
output:
  pdf_document:
    keep_tex: yes
    number_sections: yes
    toc: true
    toc_depth: 2
bibliography: biblio.bib
header-includes:
  - \usepackage{setspace} # per l'interlinea
  - \usepackage{multicol} # le colonne delle tavole
  - \usepackage{caption} # caption delle figure/tabelle
  - \usepackage[italian]{babel} # correttore dello spelling nella nostra lingua
  - \captionsetup{format=plain, font=small, labelfont=bf} # fancy caption
  - \usepackage{graphicx} # per inserire le figure
  - \usepackage{subcaption} # per inserire le caption alle sottofigure
```

Uno YAML come quello di cui sopra vi restituisce un risultato simile a quello che abbiamo visto ad adesso.

Le aggiunte che abbiamo messo ci permettono di utilizzare il file in maniera più “elastica”, ossia usando la sintassi e le potenzialità di L<sup>A</sup>T<sub>E</sub>Xma rimandando con la logica di RMarkdown

## Le figure

Ormai sappiamo a memoria come mettere le figure:

```
```{r, out.width="50%"}  
knitr::include_graphics(path = "percorso-all-a-figura")  
```
```

Però abbiamo visto il disagio che è mettere le cross-references con **bookdown**

Con **LATEX** invece è molto più semplice, anche se dobbiamo scrivere molto di più

Bisogna assicurarsi di aggiungere allo YAML:

```
> - \usepackage{graphicx}
```

# Codice

In Figura `\ref{fig:logo}` c'è il logo unipd:

```
\begin{figure}
\centering
\caption{Il logo unipd.}
\label{fig:logo}
```\{r out.width="50%"}
knitr::include_graphics("unipd.png")
```
\end{figure}
```

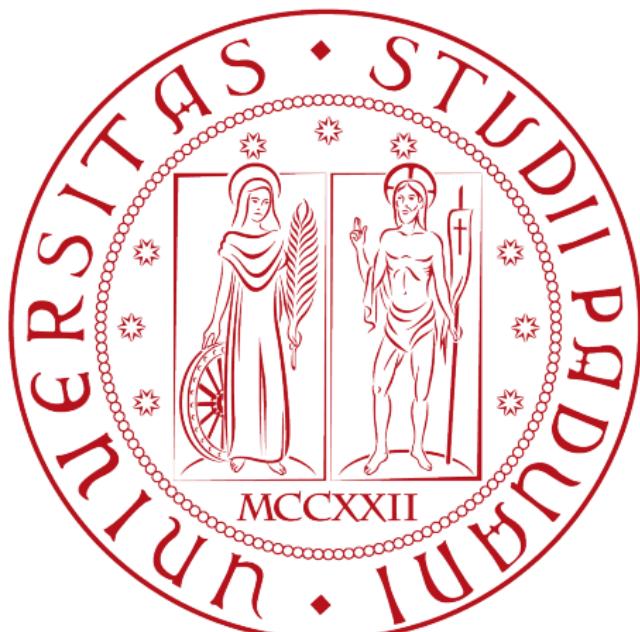
Etichetta che permette di richiamare la figura nel testo



# Risultato

In Figura 1 c'è il logo unipd:

Figura 1: Il logo unipd.



# Le sottofigure

Bisogna aggiunger allo YAML -\usepackage{subcaption}

In figura \ref{fig:doppia} si trova un esempio di sottofigure. In Figura \ref{sub:unipd1} c'è il logo unipd, in Figura \ref{sub:grafico} c'è un grafico.

```
\begin{figure}
\centering
\begin{subfigure}{0.3\textwidth}
``{r}
knitr::include_graphics("unipd.png")
```
\caption{Di nuovo unipd.}
\label{sub:unipd1}
\end{subfigure}
\begin{subfigure}{0.3\textwidth}
``{r out.width="80%"}
plot(cars)
```
\caption{Un grafico.}
\label{sub:grafico}
\end{subfigure}
\caption{Una figura doppia}
\label{fig:doppia}
\end{figure}
```

```

graph LR
    A["Caption prima sottofigura"] --> B["Label prima sottofigura"]
    C["Caption second sottofigura"] --> D["Label seconda sottofigura"]
    E["Caption della figura"] --> F["Label della figura"]
    G["Caption della figura"] --> F["Label della figura"]

```

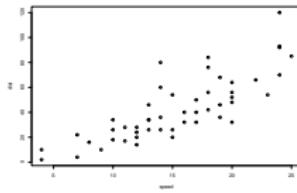
The diagram illustrates the mapping of LaTeX code elements to their corresponding labels and captions. Arrows point from specific code labels to their definitions:

- From \caption{Di nuovo unipd.} to Caption prima sottofigura
- From \label{sub:unipd1} to Label prima sottofigura
- From \caption{Un grafico.} to Caption second sottofigura
- From \label{sub:grafico} to Label seconda sottofigura
- From \caption{Una figura doppia} to Caption della figura
- From \label{fig:doppia} to Label della figura

In figura 2 si trova un esempio di sottofigure. In Figura 2a c'è il logo unipd, in Figura 2b c'è un grafico.



(a) Di nuovo unipd.



(b) Un grafico.

Figura 2: Una figura doppia

# Equazioni

Funziona il codice che abbiamo visto fino ad adesso (\$\$)

Per avere le cross-reference:

```
>
+ In Equazione \ref{eq:mean} è riportata la formula per calcolare la media
+
+ \begin{equation}\label{eq:mean}
+   \bar{X} = \frac{\sum_{i=1}^n x_i}{n}
+ \end{equation}
```

# Equazioni

Funziona il codice che abbiamo visto fino ad adesso (\$\$)

Per avere le cross-reference:

```
>
+ In Equazione \ref{eq:mean} è riportata la formula per calcolo
+
+ \begin{equation}\label{eq:mean}
+   \bar{X} = \frac{\sum_{i=1}^n x_i}{n}
+ \end{equation}
```

In Equazione 1 è riportata la formula per calcolare la media della variabile  $X$ :

$$\bar{X} = \frac{\sum_{i=1}^n x_i}{n} \quad (1)$$

## Tabelle

```
I risultati sono riportati in Tabella \ref{tab:tabella}
```
r, results='asis'}
print(xtable::xtable(rock[1:7, ], caption = "Tabella delle ro
                  label = "tab:tabella"),
      comment = FALSE,
      caption.placement = "top")
```
```

I risultati sono riportati in Tabella 1.

Tabella 1: Tabella delle rocce

|   | area | peri    | shape | perm  |
|---|------|---------|-------|-------|
| 1 | 4990 | 2791.90 | 0.09  | 6.30  |
| 2 | 7002 | 3892.60 | 0.15  | 6.30  |
| 3 | 7558 | 3930.66 | 0.18  | 6.30  |
| 4 | 7352 | 3869.32 | 0.12  | 6.30  |
| 5 | 7943 | 3948.54 | 0.12  | 17.10 |
| 6 | 7979 | 4010.15 | 0.17  | 17.10 |
| 7 | 9333 | 4345.75 | 0.19  | 17.10 |

# Frontespizio

Sembra più difficile di quanto non sia:

```
> output:  
+   pdf_document:  
+     includes:  
+       before_body: titolo.sty  
+       keep_tex: yes  
+       number_sections: yes  
+       toc_depth: 2  
+ bibliography: biblio.bib  
+ geometry: margin=1in  
+ header-includes:  
+   - \usepackage{setspace}  
+   - \usepackage{multicol}  
+   - \usepackage{caption}  
+   - \usepackage[italian]{babel}  
+   - \captionsetup{format=plain, font=small, labelfont=bf}  
+   - \usepackage{graphicx}  
+   - \usepackage{subcaption}
```

In questo modo si forza RMarkdown a prendere un frontespizio esterno, creato a parte

Per creare il frontespizio è necessario avere un po' di conoscenze in L<sup>A</sup>T<sub>E</sub>X

Molto importante: se nel frontespizio usate dei pacchetti (e.g., se volete aggiungere una figura dovete avere il pacchetto `graphicx`) dovete caricarli nello YAML del documento principale

Ho creato `titolo.sty` per voi, dovete solo modificarlo!

Your turn!

- Inserite una figura con cross-reference nel testo
- Inserite il plot del vostro data set con cross-reference nel testo (il codice è [qui](#))
- Inserite una tabella con cross-reference nel testo

## ADAVANCED

- Inserite una figura con tre sottofigure (due foto e il grafico del vostro dataset) con rispettive cross-reference
- Cambiate la pagina del titolo modificando il file [titolo.sty](#)

# Presentazioni

# Lo YAML

```
---
```

```
title: "03-bis - \LaTeX bis"
author: |
  | Ottavia M. Epifania
  | University of Padova
date: 'Corso RMarkdown'
institute: "ottavia.epifania@unipd.it"
output:
  beamer_presentation:
    theme: CambridgeUS
    colortheme: beaver
    fonttheme: serif
    highlight: haddock
header_includes:
  - \AtBeginDocument{\title[\LaTeX bis]{03-bis - \LaTeX bis}}
  - \usepackage{graphicx}
  - \usepackage{setspace}
  - \usepackage{tabularx}
  - \usepackage[italian]{babel}
  - \usepackage{tikzsymbols}
  - \usepackage{subcaption}
  - \usepackage{tikz}
  - \usepackage{spot}
  - \usepackage{tabularx}
  - \usepackage[absolute,overlay]{textpos}
  - \usepackage{booktabs}
  - \newcommand\Factor{1.2}
  - \setbeamerfont{subtitle}{size=\large, series=\bfseries}
  - \definecolor{template}{RGB}{54, 114, 89}
  - \setbeamercolor{frametitle}{bg=white}
  - \setbeamertemplate{frametitle}[default][center]
  - \AtBeginSection{\frame{\sectionpage}}
  - \setbeamercolor{section name}{fg=white}
```

Aspetto della presentazione

Pacchetti e opzioni

Si può scegliere combinazione di `theme`, `colortheme`, `fonttheme` e `highlight`.

A [questa pagina](#) sono disponibili tutti i possibili temi per la vostra presentazione

`highlight` è la formattazione del vostro codice, a cui potete accedere cliccando sulla rotellina accanto a `knit` → Output Options

# Colonne

Mi raccomando: - `\usepackage{multicol}` deve essere nello YAML:

```
>
+ \begin{columns} % L'ambiente colonna
+ \begin{column}{0.50\textwidth}
+ Testo nella prima colonna
+ \end{column}
+
+ \begin{column}{0.50\textwidth}
+ Testo nella seconda colonna
+ \end{column}
+
+ \end{columns}
```

# Colonne

Mi raccomando: - `\usepackage{multicol}` deve essere nello YAML:

```
>
+ \begin{columns} % L'ambiente colonna
+ \begin{column}{0.50\textwidth}
+ Testo nella prima colonna
+ \end{column}
+
+ \begin{column}{0.50\textwidth}
+ Testo nella seconda colonna
+ \end{column}
+
+ \end{columns}
```

Testo nella prima colonna

Testo nella seconda colonna

PS: si possono mettere anche più di due colonne

# Testo incrementale

Avete visto che in alcune mie slide il testo appare in modo incrementale. Questo effetto si può ottenere in due modi, entrambi basati su L<sup>A</sup>T<sub>E</sub>X:

\pause

Questo è il comando più semplice, si mette davanti al contenuto che si vuole “bloccare”

# Testo incrementale

Avete visto che in alcune mie slide il testo appare in modo incrementale. Questo effetto si può ottenere in due modi, entrambi basati su L<sup>A</sup>T<sub>E</sub>X:

\pause

Questo è il comando più semplice, si mette davanti al contenuto che si vuole “bloccare”

Testo che appare subito

\pause

Testo che appare dopo la pausa

# Testo incrementale

Avete visto che in alcune mie slide il testo appare in modo incrementale. Questo effetto si può ottenere in due modi, entrambi basati su L<sup>A</sup>T<sub>E</sub>X:

\pause

Questo è il comando più semplice, si mette davanti al contenuto che si vuole “bloccare”

Testo che appare subito

\pause

Testo che appare dopo la pausa

Testo che appare subito

# Testo incrementale

Avete visto che in alcune mie slide il testo appare in modo incrementale. Questo effetto si può ottenere in due modi, entrambi basati su L<sup>A</sup>T<sub>E</sub>X:

\pause

Questo è il comando più semplice, si mette davanti al contenuto che si vuole “bloccare”

Testo che appare subito

\pause

Testo che appare dopo la pausa

Testo che appare subito

Testo che appare dopo la pausa

```
\onslide<i-n>
```

Dove  $i$  è la prima slide su cui il contenuto deve apparire,  $n$  è l'ultima slide su cui il contenuto appare ( $n$ ) può essere omesso.

```
\onslide<2-> Appare sulla seconda slide e rimane fino alla fine
```

```
\onslide<3-3> Appare alla terza slide e scompare alla terza
```

```
\onslide<4-> Appare sull'ultima slide
```

```
\onslide<i-n>
```

Dove  $i$  è la prima slide su cui il contenuto deve apparire,  $n$  è l'ultima slide su cui il contenuto appare ( $n$ ) può essere omesso.

```
\onslide<2-> Appare sulla seconda slide e rimane fino alla fine
```

```
\onslide<3-3> Appare alla terza slide e scompare alla terza
```

```
\onslide<4-> Appare sull'ultima slide
```

Appare sulla seconda slide e rimane fino alla fine

```
\onslide<i-n>
```

Dove  $i$  è la prima slide su cui il contenuto deve apparire,  $n$  è l'ultima slide su cui il contenuto appare ( $n$ ) può essere omesso.

```
\onslide<2-> Appare sulla seconda slide e rimane fino alla fine
```

```
\onslide<3-3> Appare alla terza slide e sparisce alla terza
```

```
\onslide<4-> Appare sull'ultima slide
```

Appare sulla seconda slide e rimane fino alla fine

Appare alla terza slide e sparisce alla terza

```
\onslide<i-n>
```

Dove  $i$  è la prima slide su cui il contenuto deve apparire,  $n$  è l'ultima slide su cui il contenuto appare ( $n$ ) può essere omesso.

```
\onslide<2-> Appare sulla seconda slide e rimane fino alla fine
```

```
\onslide<3-3> Appare alla terza slide e scompare alla terza
```

```
\onslide<4-> Appare sull'ultima slide
```

Appare sulla seconda slide e rimane fino alla fine

Appare sull'ultima slide

# Blocchi di testo

## Il testo

Viene racchiuso in blocchi che lo mettono in risalto

# Blocchi di testo

## Il testo

Viene racchiuso in blocchi che lo mettono in risalto

## Blocchi di esempio

Dove il titolo è in verde e a seconda del tema scelto anche la sfumatura esterna del blocco stesso

# Blocchi di testo

## Il testo

Viene racchiuso in blocchi che lo mettono in risalto

## Blocchi di esempio

Dove il titolo è in verde e a seconda del tema scelto anche la sfumatura esterna del blocco stesso

## Blocchi di Warning

Dove il titolo è in rosso e a seconda del tema scelto anche la sfumatura esterna del blocco stesso

```
> \begin{block}{Il testo}
+
+ Viene racchiuso in blocchi che lo mettono in risalto
+
+ \end{block}
+
+ \pause
+
+ \begin{exampleblock}{Blocchi di esempio}
+
+ Dove il titolo è in verde e a seconda del tema scelto anche la sfumatura
+
+ \end{exampleblock}
+
+ \pause
+
+ \begin{alertblock}{Blocchi di Warning}
+
+ Dove il titolo è in rosso e a seconda del tema scelto anche la sfumatura
+
+ \end{alertblock}
```

Your turn!

- Create una slide con due colonne, nella prima colonna testo, nella seconda colonna testo e una figura
- Far apparire la seconda colonna, il suo testo e la sua figura uno alla volta
- Create una slide con 3 blocchi

## ADVANCED

- Fate apparire i blocchi così:
  - il terzo blocco sulla prima slide e rimane fino alla fine
  - il secondo blocco compare e scompare alla seconda slide
  - il primo blocco appare e rimane sull'ultima slide