

# Introdução ao L<sup>A</sup>T<sub>E</sub>X

Bruna Gabriela Wendpap  
Djair Durand Ramalho Frade

Fernando de Pol Mayer

Luiz Ricardo Nakamura

Maria Cristina Martins

Thiago de Paula Oliveira

Thiago Gentil Ramires

Profa. responsável: Dra. Roseli Aparecida Leandro

Universidade de São Paulo (USP)

Escola Superior de Agricultura "Luiz de Queiroz" (ESALQ)

04 de Outubro, 2013



# Sumário

- 1 Introdução
- 2 Configuração
- 3 Opções



# Plano de aula

- 1 Introdução
- 2 Configuração
- 3 Opções



# Introdução

Teste



# Plano de aula

- 1 Introdução
- 2 Configuração
- 3 Opções



# Configuração

- O **knitr** pode ser utilizado em qualquer editor de texto, mas alguns facilitadores são
  - **Emacs** com ESS
  - **LyX**
  - **RStudio**
- A primeira coisa a fazer no **R** é instalar os pacotes necessário:

```
install.packages(c("knitr", "xtable"), dependencies = TRUE)
```



# Configuração

A ideia é fazer a seguinte sequência:

- 1 Criar um arquivo com a extensão `.Rnw`
- 2 Inserir o preâmbulo tradicional do  $\text{\LaTeX}$ , texto e código
- 3 Compilar o arquivo com a função `knit()` → vai gerar um arquivo `.tex`
- 4 Compilar o arquivo `.tex` no  $\text{\TeX}$ Maker (ou outros) → gera o arquivo `.pdf`



# Configuração

Expressões do **R** são inseridas normalmente dentro de um ambiente especial no arquivo `.Rnw`:

```
<<>>=
```

```
...
```

```
@
```

- Toda expressão do **R** que estiver dentro deste **chunk** será interpretada quando coompilada pelo **knitr**, gerando a saída, gráficos, etc.
- Para inserir resultados no meio do texto (*inline*) use `\Sexpr{}`





# Configuração

## Um exemplo mínimo:

```
\documentclass{article}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[brazil]{babel}
\usepackage[margin=2.5cm]{geometry}
```

```
\begin{document}
```

Definindo a variável aleatória  $X$  com distribuição Normal padrão, ou seja,  $X \sim \text{N}(0,1)$

```
<<>>=
```

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

```
(x <- rnorm(10))
```

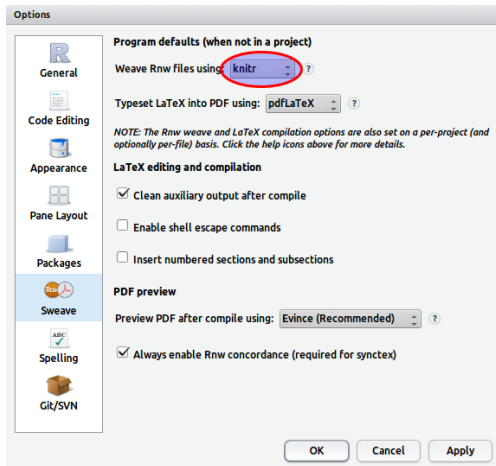
```
@
```

A média desta variável aleatória é  $\% \text{\%Sexpr{mean(x)}} \%$ . O primeiro valor é  $X_i = \% \text{\%Sexpr{x[1]}} \%$ .

```
\end{document}
```

# Configuração

No **RStudio**, altere as opções para deixar o **knitr** como padrão



# Configuração

Com isso, o botão **Compile PDF**:

- Compila o arquivo `.Rnw` com o **knitr**
- Compila o arquivo `.tex` resultante com o **PDF $\LaTeX$**
- Abre o PDF no editor padrão

Tudo isso é feito em um ambiente separado da sua área de trabalho original, para não haver confusão com objetos gerados



# Plano de aula

- 1 Introdução
- 2 Configuração
- 3 Opções**



# Opções

- **Opções específicas:** são opções definidas para cada *chunk* de código, especificadas entre `<<` e `>>`
- **Opções globais:** são opções definidas para todos os *chunks* do documento, especificadas pela função do **knitr**

```
opts_chunk$set()
```

Ainda assim, é possível especificar opções para *chunks* individuais!



# Opções

Algumas opções gerais (\* indica o padrão):

- `eval`: TRUE\*, FALSE, ou `c(1,3,4)` ou `-(4:6)`. Se o código deve ser interpretado ou não. Podem ser especificadas apenas linhas.
- `echo`: TRUE\*, FALSE, ou `c(1,3,4)` ou `-(4:6)`. Se deve mostrar ou não o código ou pedaços dele. O resultado sempre aparece. (Diferente de `eval`, porque sempre será interpretado).
- `include`: TRUE\*, FALSE. Se deve incluir ou não o código no documento final. Útil por exemplo para fazer cálculos auxiliares que não precisam ser mostrados no documento.



# Opções

Algumas considerações sobre as opções:

- Cada chunk pode ter um nome, por exemplo

```
<<bloco1>>=
```

@

que será útil posteriormente para dar nomes à figuras e extração de código.

- Evite usar espaços e pontos nos nomes dos chunks (use - e \_).
- As opções em <<>>= devem estar sempre na **mesma linha**
- **Todas** as opções devem ser expressões válidas do R
  - Caracteres entre aspas
  - Valores lógicos: TRUE ou FALSE



# Opções

## Exercício

- Insira mais 3 chunks no seu arquivo, cada um com pelo menos 3 linhas de código do **R**
- Teste as opções `eval`, `echo`, e `include` em cada uma delas





# Opções

Opções para decoração de código:

- `prompt`: TRUE ou FALSE\*. Se deve ou não inserir o *prompt* (>) do R no início de cada linha.
- `comment`: "##"\* ou NA. Se deve ou não comentar a saída dos comandos
- `highlight`: TRUE\* ou FALSE. Se deve colorir os códigos.
- `size`: "normalsize" ou qualquer tamanho de fonte do  $\text{\LaTeX}$  ("small", "footnotesize", ...)
- `tidy`: TRUE\* ou FALSE. Se o código deve ser formatado para um padrão geral pré-definido.



# Opções

## Exercício

- Altere estas opções em alguns chunks e veja o resultado
- Em especial escreva esse código dentro de dois chunks separados

```
rmnorm(10,  
        10, 5)
```

cada um com `tidy=TRUE` e `tidy=FALSE`.



# Opções globais

Para definir um padrão para todos os chunks globalmente, podemos especificar logo no início do documento as opções que queremos. Por exemplo:

```
<<setup, include=FALSE, cache=FALSE, tidy=FALSE>>=
# Minhas configurações globais para os chunks
opts_chunk$set(size = "small",
                prompt = FALSE,
                comment = NA,
                tidy = FALSE,
                cache = TRUE)
```

@



# Opções

O cache (TRUE\*/FALSE):

- Armazena os resultados dos chunks em disco
  - Cria um diretório cache
- Esse resultado será reaproveitado nas próximas compilações, portanto um chunk em cache não será interpretado novamente
- Muito útil para documentos muito longos ou com comandos demorados ou bases de dados grandes



## Teste

```
> rnorm(10)
[1] -0.18155 -0.25880  1.30993 -0.18987  1.39511  0.02522  0.93071
[8] -0.47489  2.77158 -1.38190
> runif(10, 1, 2)
[1] 1.524 1.180 1.952 1.353 1.192 1.050 1.667 1.935 1.810 1.486
```

## Mais

