

# Tricks and Solutions

Yeny Katherine Muñoz Nuñez

Universidad Distrital Francisco José de Caldas

3 de junio de 2016

# Indice

- 1 Chunk Options
  - Option Aliases
  - Option Templates
  - Code in Appendix
  - Local R Options
- 2 Package Options
- 3 Typesetting
  - Output Width
  - Box Padding
  - Beamer
  - Suppress Long Output
- 4 Utilities
  - R Package Citation
- 5 Debugging
- 6 Multilingual Support
- 7 Bibliografía

# Chunk Options - Option Aliases

En el inicio de un documento:

```
set_alias(w = "fig.width", h = "fig.height")
```

Se puede hacer uso de 'w' y 'h':

```
<<fig-size, w=5, h=3>>=  
plot(1:10)  
@
```

Lo anterior es equivalente a:

```
<<fig-size, fig.width=5, fig.height=3>>=  
plot(1:10)  
@
```

# Chunk Options - Option Templates

Una plantilla es un conjunto de opciones:

```
opts_template$set(  
  fig.large = list(fig.width = 7, fig.height = 5),  
  fig.small = list(fig.width = 3.5, fig.height = 3)  
)
```

Después de creada la plantilla, podemos simplemente usar el nombre asignado:

```
<<fig-ex, opts.label='fig.large'>>=  
plot(1:10)  
@
```

Lo anterior es equivalente a:

```
<<fig-ex, fig.width=7, fig.height=7>>=  
plot(1:10)  
@
```

# Chunk Options - Code in Appendix

A veces no queremos mostrar los trozos de código en el cuerpo del informe:

```
<<A, echo=FALSE>>=
1+1
<<B, echo=FALSE>>=
2+2
<<C, echo=FALSE>>=
rnorm(10)
<<show-code, ref.label=c('A', 'B', 'C'), eval=FALSE>>=
@
```

Si se tienen muchos chunks en un documento, se puede hacer uso de:

```
<<show-code, ref.label=all_labels()>>=
@
```



# Chunk Options - Local R Options

R.options permite tomar una lista de R por un código chunk:

```
<<R.options = list(width=30, digits=2)>>=
```

```
seq(0, 10, length = 20)
```

```
## [1] 0.00 0.53 1.05 1.58
```

```
## [5] 2.11 2.63 3.16 3.68
```

```
## [9] 4.21 4.74 5.26 5.79
```

```
## [13] 6.32 6.84 7.37 7.89
```

```
## [17] 8.42 8.95 9.47 10.00
```

```
@
```

# Package Options

Para ver más información respecto a los chunk en un código fuente, se puede activar el modo detallado por medio del comando:

```
opts_knit$set(verbose = TRUE).
```

root.dir se puede utilizar para establecer el directorio de trabajo en el código chunk:

```
opts_knit$set(root.dir = "/home/foo/bar/")
```

Para los chunk que no están etiquetados:

```
opts_knit$set(unnamed.chunk.label = "fig")
```

# Typesetting - Output Width

Cuando vemos que el código fuente o la salida de texto es demasiado amplia, podemos utilizar una opción de menor anchura:

```
options(width = 55)
```

Sin embargo, existen casos donde la cadena de caracteres es demasiado larga en el código fuente:

```
# unable to wrap the source code
x <- "thisistoolongandRisunabletofindaplacetoinserthelinebreak"
# unable to wrap the output line
cat(x, "----")

## thisistoolongandRisunabletofindaplacetoinserthelinebreak ----
```

Es posible dividirla en pedazos más pequeños de forma manual y unirla nuevamente como se muestra a continuación:

```
x <- paste("this", "is", "too", "long", "and", "R", "is",
"unable", "to", "find", "a", "place", "to", "insert",
"the", "line", "break", sep = "")
```



# Typesetting - Box Padding

Si sentimos que el relleno por defecto de la caja es demasiado ajustado, se puede restablecer la longitud:

```
\setlength\fbboxsep{5mm}
```

```
## an intentional comment to to to to to to to to to to to to  
## reach the page margin  
rpois(40, 5)
```

```
## [1] 6 4 6 4 9 5 2 4 2 4 4 10 6 3 1 8 8  
## [18] 2 7 4 10 6 5 2 7 4 6 4 2 5 8 7 2 3  
## [35] 2 7 7 3 3 3
```

Se puede definir la misma clase en CSS de la siguiente forma:

```
div.chunk {  
  padding: 5mm;  
}
```

# Typesetting - Beamer

## Ejemplo de uso de knitr en diapositiva beamer:

```
\documentclass{beamer}
\begin{document}
\title{Using knitr in Beamer}
\author{Yihui Xie}

\maketitle

\begin{frame}
\frametitle{Introduction}
This is a normal slide.
\end{frame}

% need the option [fragile] for code output!
\begin{frame}[fragile]
\frametitle{Code chunks}
<<test, out.width='.6\\linewidth', fig.align='center'>>=
par(mar = c(4, 4, .1, .1))
x = rnorm(100)
hist(x, main='', col='lightblue', border='white')
rug(x)
@
\end{frame}

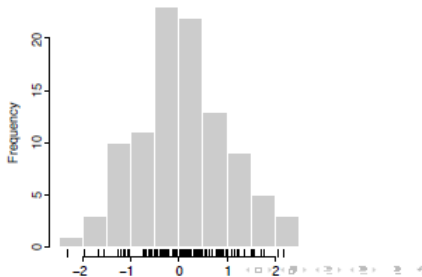
\end{document}
```

# Typesetting - Beamer

## Ejemplo de diapositiva con beamer:

### Code chunks

```
par(mar = c(4, 4, 0.1, 0.1))
x = rnorm(100)
hist(x, main = "", col = "lightblue", border = "white")
rug(x)
```



# Typesetting - Suppress Long Output

Omitir partes del libro, debido a que los resultados son muy largos:

```
# the default output hook
hook_output <- knit_hooks$get("output")
knit_hooks$set(output = function(x, options) {
  # print the first 5 lines by default
  if (is.null(n <- options$out.lines))
    n <- 5
  x <- unlist(stringr::str_split(x, "\n"))
  if (length(x) > n) {
    # truncate the output
    x <- c(head(x, n), "... \n")
  }
  # paste first n lines together
  x <- paste(x, collapse = "\n")
  hook_output(x, options)
})
```

La idea básica de que la regla definida anteriormente es:

```
library(MASS)
painters

##               Composition Drawing Colour Expression
## Da Udine         10         8      16           3
## Da Vinci         15        16       4          14
## Del Piombo        8         13      16           7
## Del Sarto        12        16       9           8
....
```

# Utilities - R Package Citation

Por defecto se recoge los paquetes cargados en la sesión actual de R y extrae su información de la cita:

```
write_bib(c("filehash", "RGtk2", "rms"))

@Manual{R-filehash,
  title = {filehash: Simple key-value database},
  author = {Roger D. Peng},
  year = {2014},
  note = {R package version 2.2-2},
  url = {http://CRAN.R-project.org/package=filehash},
}
@Manual{R-RGtk2,
  title = {RGtk2: R bindings for Gtk 2.8.0 and above},
  author = {Michael Lawrence and Duncan {Temple Lang}},
  year = {2014},
  note = {R package version 2.20.31},
  url = {http://CRAN.R-project.org/package=RGtk2},
}
@Manual{R-rms,
  title = {rms: Regression Modeling Strategies},
  author = {Frank E. {Harrell, Jr.}},
  year = {2015},
  note = {R package version 4.3-0},
  url = {http://CRAN.R-project.org/package=rms},
}
```

Si el archivo principal el chunk se escribe de la siguiente forma:

```
<<write-bib, cache=TRUE, version=packageVersion('foo')>>=
write_bib(c("foo", "other", "packages"), file = "paper.bib")
@
```

# Debugging

## Debugging

Cuando se produce un error que no se observa claramente en la pantalla, podemos utilizar herramientas de depuración comunes, tales como `traceback ()` (para ver el paso a paso que ha conducido al error), `debug ()` o `browser ()`.

# Multilingual Support

Si el documento de origen no se ha codificado con la codificación nativa del sistema actual, tendremos que especificar manualmente su codificación mediante el argumento de codificación en:

```
knit("yourfile.Rnw", encoding = "GB2312")
```

# Bibliografía



Yihui Xie (2013).

*Dynamic Documents with R and knitr.*

CRC Press 2016. 2 Ed.