



Escribir un
Trabajo Fin de Estudios
con R Markdown

Pedro Luis Luque Calvo

Sevilla, Noviembre de 2017

Índice general

| | |
|---|-----------|
| Prólogo | III |
| Resumen | IV |
| Abstract | V |
| Índice de Figuras | VII |
| Índice de Cuadros | VIII |
| 1. Introducción | 1 |
| 1.1. ¿Por qué utilizar R Markdown para escribir un trabajo fin de estudios? | 1 |
| 1.2. La estructura de un trabajo fin de estudios | 3 |
| 1.3. Instalación de las herramientas necesarias para usar R Markdown | 4 |
| 1.3.1. Instalación de R | 5 |
| 1.3.2. Instalación de RStudio | 5 |
| 1.3.3. Instalación de Pandoc | 6 |
| 1.3.4. Instalación del sistema LaTeX | 6 |
| 2. Cómo utilizar la plantilla R Markdown para crear un trabajo fin de estudios con RStudio | 8 |
| 2.1. Pasos a seguir para escribir un proyecto completo de trabajo fin de estudios | 8 |
| 2.1.1. La portada | 15 |
| 2.2. Otras plantillas para realizar un trabajo fin de estudios | 17 |
| 3. Uso de R Markdown para escribir documentos | 18 |
| 3.1. Aprendiendo lo básico del lenguaje de marcas: Markdown | 18 |
| 3.1.1. Resaltado de texto | 18 |
| 3.1.2. Cabeceras o títulos de sección | 18 |
| 3.1.3. Listas | 19 |
| 3.1.3.1. Listas no numeradas | 19 |
| 3.1.3.2. Listas ordenadas o numeradas | 19 |
| 3.1.4. Producir saltos de línea manualmente | 21 |
| 3.1.5. Insertar enlaces web | 22 |
| 3.1.6. Insertar imágenes | 22 |
| 3.1.7. Incluir enlaces web e imágenes con identificadores | 23 |
| 3.1.7.1. Enlaces web | 23 |
| 3.1.7.2. Imágenes | 23 |
| 3.1.8. Notas al pie de página | 24 |
| 3.1.9. Resaltar bloques de texto | 25 |
| 3.1.10. Insertar código de lenguajes de programación | 25 |
| 3.1.10.1. Bloques de código de más de una línea | 25 |
| 3.1.10.2. Código de lenguajes dentro del texto | 26 |
| 3.1.11. Ecuaciones matemáticas con LaTeX | 26 |
| 3.1.11.1. Expresión matemática en el texto | 26 |
| 3.1.11.2. Ecuación matemática centrada | 26 |

| | |
|--|-----------|
| 3.1.11.3. Más ejemplos de expresiones matemáticas con LaTeX | 27 |
| 3.1.12. Incluir una línea horizontal | 28 |
| 3.1.13. Tablas | 29 |
| 3.1.14. Miscelánea | 29 |
| 3.2. R Markdown | 30 |
| 3.2.1. Cómo insertar código R en R Markdown | 30 |
| 3.2.1.1. Código R dentro del texto | 30 |
| 3.2.1.2. Ejecución de bloques de código R | 30 |
| 3.2.1.3. Cómo mostrar código R y las opciones del chunk utilizadas | 34 |
| 3.2.1.4. Cómo insertar gráficos generados con R | 34 |
| 3.2.1.5. Insertar 2 gráficos R frente a frente | 36 |
| 3.2.1.6. Insertar gráficos generados con ggplot2 | 37 |
| 3.2.2. Cómo incluir ficheros gráficos en R Markdown | 40 |
| 3.2.2.1. Incluir leyendas que describan un gráfico y cómo referenciarlos | 41 |
| 3.2.3. Cómo incluir información en forma de tablas en R Markdown | 42 |
| 3.2.3.1. Incluir leyendas que describan una tabla de información y cómo referenciarlas | 42 |
| 3.2.3.2. Tablas con kableExtra | 43 |
| 3.2.3.3. Tablas de más de una página con kableExtra | 46 |
| 3.2.3.4. Tablas en modo apaisado con kableExtra | 48 |
| 3.2.4. Referencias a fórmulas matemáticas LaTeX en R Markdown | 50 |
| 3.2.5. Cómo hacer referencias a secciones en R Markdown | 51 |
| 3.2.6. Incluir citas bibliográficas en R Markdown | 51 |
| 3.2.7. Incluir páginas web, aplicaciones Shiny y Widgets en R Markdown | 56 |
| 3.2.8. Usar código LaTeX dentro de R Markdown | 58 |
| 3.2.8.1. Tamaños de fuentes con ayuda de LaTeX | 58 |
| 3.2.8.2. Colores con ayuda de LaTeX | 59 |
| 3.2.8.3. Resaltar bloques de texto de un modo personalizado | 62 |
| 3.2.8.4. Uso del comando LaTeX: table | 65 |
| 3.2.8.5. Uso del comando LaTeX: figure | 66 |
| 3.2.8.6. Otros comandos LaTeX útiles | 66 |
| 4. Recomendaciones finales | 72 |
| 4.1. Qué se ha conseguido | 72 |
| 4.2. Qué otras posibilidades nos brinda R Markdown | 72 |
| 4.2.1. Crear una presentación para la defensa del trabajo fin de estudios con R Markdown | 73 |
| 4.2.2. Algunos consejos sobre la presentación del TFE ante un tribunal | 73 |
| 4.3. Líneas futuras de trabajo | 74 |
| A. Paquetes R utilizados | 75 |
| B. Código comentado | 76 |
| B.1. Fichero tfe_principal.Rmd | 76 |
| B.2. Fichero: capitulo01.Rmd | 81 |
| B.3. Fichero: latex_preambulo.tex | 82 |
| B.4. Fichero: latex_paginatitulo_modTFGE.tex | 91 |
| B.5. Ficheros: a_TODO.Rmd | 93 |
| Bibliografía | 94 |

Prólogo

Este trabajo surgió para ayudar a los alumnos en la tarea de escribir un trabajo fin de estudios, una tarea fundamental y que es obligatoria en el último curso de las titulaciones universitarias. Con este documento se pretende también facilitar la labor del tutor, al reducir el número de consultas que los alumnos hacen sobre cuestiones técnicas para la creación y organización del trabajo fin de estudios, pudiendo dedicar más tiempo a la revisión de las cuestiones científicas.

Además se ha pretendido dar a conocer las posibilidades que brinda R para crear informes a través de R Markdown. Muy impulsadas por el equipo de RStudio, no solamente con su entorno de desarrollo RStudio Desktop [9], sino que también por los paquetes R que han construido: “rmarkdown”, “knitr”, “ggplot2”, etc.

Se ha creado una plantilla R Markdown¹ que se puede utilizar para escribir un trabajo fin de estudios de cualquier titulación universitaria, no solamente para titulaciones del área de las Ciencias. Una vez instalados los programas gratuitos, solamente habría que escribir el texto y pulsar el botón de creación (“Knit”) en RStudio para obtener un documento pdf con altísima calidad. El conocimiento de R no sería necesario.

Esta plantilla también se puede emplear para hacer cualquier otro tipo de trabajos que tenga una estructura de capítulos. Señalar que este documento se ha creado con esta misma plantilla R Markdown.

Este trabajo está dedicado a mi familia.

Quiero agradecer también a toda la comunidad de usuarios de R y de LaTeX por haber hecho posible y mantener vivas estas fabulosas herramientas.

Información sobre el autor:

Pedro Luis Luque Calvo

- Profesor del Departamento de Estadística e Investigación Operativa de la Universidad de Sevilla
- Email: calvo@us.es
- Página web personal: <http://destio.us.es/calvo>.

En esta página se podrá encontrar información sobre material adicional: vídeos con tutoriales, actualizaciones futuras, . . . , y hacer consultas o sugerencias.

Para **citar este trabajo** en publicaciones, utilizar el siguiente formato:

Luque-Calvo, P.L. 2017. Escribir un trabajo fin de estudios con R Markdown.
Disponible en <http://destio.us.es/calvo>.

Para insertar esta referencia en un fichero bibliográfico BibTeX, añadir el siguiente código:

```
@Manual{Luque2017,  
  title = {Escribir un Trabajo Fin de Estudios con R Markdown},  
  author = {Pedro L. Luque-Calvo},  
  year = "2017",  
  howpublished = {Disponible en \url{http://destio.us.es/calvo}}  
}
```

¹Está disponible en: <http://destio.us.es/calvo/memoriatfe/MemoriaTFE.zip>

Resumen

En este trabajo se explica cómo escribir un trabajo fin de estudios con ayuda de R Markdown, a través de los programas gratuitos: R y RStudio.

Se ha dividido en cuatro capítulos. En el primer capítulo, de introducción, se motiva el uso de R Markdown para escribir un trabajo fin de estudios y se dan algunas pautas en cuanto a la estructura de este tipo de trabajos. También se dan las claves para instalar los programas informáticos necesarios.

En el segundo capítulo, se examina el contenido de la plantilla que acompaña a este documento y se explica cómo debe usarse para crear el trabajo fin de estudios.

El tercer capítulo, es el más extenso de todos y en él se hace un repaso a las características principales del lenguaje de marcas ligeras “Markdown” y a la variante “R Markdown”. Se dan muchos ejemplos para resolver situaciones habituales en la creación de un documento de este tipo: inclusión de gráficos, inclusión de tablas de información, uso de código R, y algunos consejos en el uso de LaTeX en documentos R Markdown.

En el último capítulo, además de obtener conclusiones sobre lo conseguido, enumera algunos usos adicionales de R Markdown que pueden ayudar también a otras tareas, como es la presentación o defensa del trabajo fin de estudios ante un tribunal.

Abstract

This document explains how to write a final Project by using R Markdown, available with the free software R and RStudio.

The document has been divided into four chapters. In Chapter 1, the use of R Markdown is motivated as an appropriate tool to write a final project. Guidelines dealing with the structure of this kind of work are given. The keys to install the necessary software are also included.

In Chapter 2, the content of the template file input of this document is analysed, and its use to create the final project is explained.

Chapter 3 is the most extensive of all. The main features of light mark-up language called “Markdown” and its variant “R Markdown” are reviewed in it. A lot of examples are given to solve common problems in the creation of this kind of documents such as inclusion of graphics, inclusion of tables with information, and use of R code. Some advice on the use of LaTeX in R Markdown documents is also given.

Finally, Chapter 4 is devoted to conclusions about the achievements we reached with this document, and R Markdown additional uses that can be helpful to write and defend a final Project before a supervisory committee.

Índice de figuras

| | |
|---|----|
| 2.1. En esta captura se muestra la aplicación RStudio al abrir el proyecto de la plantilla: MemoriaTFE.Rproj. | 11 |
| 2.2. En esta captura se muestra el fichero tfe_principal.Rmd al hacer click sobre este fichero en el Panel Files, en la parte inferior derecha. | 11 |
| 2.3. En esta captura se muestra la estructura del fichero tfe_principal.Rmd en la que aparece el código R Markdown para incluir los ficheros de los capítulos. | 12 |
| 2.4. En esta captura se muestra el fichero capitulo01.Rmd que se ha abierto con ayuda del Panel Files. Una vez abierto está listo para ser editado con código R Markdown. | 12 |
| 2.5. En esta captura se muestra el fichero apendice01.Rmd. | 13 |
| 2.6. En esta captura se muestra que al hacer click sobre el icono ‘Knit’ al tener abierto el fichero tfe_principal.Rmd se inicia el proceso de compilación del documento. | 13 |
| 2.7. En esta captura se muestra, si no ha habido errores, el fichero pdf generado con el TFE en su estado actual. | 14 |
| 2.8. En esta captura se muestra el pdf generado sobre el que se puede navegar para realizar consultas. | 14 |
| 2.9. Se muestra la apariencia de la portada elegida para el TFE. | 15 |
| 2.10. Se muestran los dos estilos de portadas del modelo LT: LT01 y LT02. . . | 16 |
| 2.11. Se muestran los dos estilos de portadas del modelo ST: ST01 y ST02. . | 16 |
| 2.12. Se muestran los tres estilos de portadas de los modelos OV: OV01, OV02 y OV03. | 17 |
| | |
| 3.1. Insertar gráficos en internet con markdown | 22 |
| 3.2. Insertar imágenes locales con markdown | 23 |
| 3.3. Leyenda que aparece bajo el gráfico incluido con identificadores | 24 |
| 3.4. Explicación figura. | 35 |
| 3.5. Una explicación del gráfico. | 36 |
| 3.6. Dos gráficos R cara a cara | 36 |
| 3.7. Gráfico creado con ggplot2 con una nube de puntos y el modelo de regresión lineal. | 39 |
| 3.8. En esta gráfico se muestra un mapa de carreteras alrededor de Sevilla generado con ggmap. | 40 |
| 3.9. Página oficial de R. | 41 |
| 3.10. Presentación de la Bibliografía con el fichero: acm-sig-proceedings-long-author-list.csl | 54 |
| 3.11. Presentación de la Bibliografía con el fichero: methods-in-ecology-and-evolution.csl | 55 |

| | |
|---|----|
| 3.12. Esquema del uso de Google Scholar para encontrar las entradas bibliográficas en formato BibTeX. | 56 |
| 3.13. Colores LaTeX en los paquetes <i>color</i> y <i>xcolor</i> , respectivamente | 61 |
| 3.14. Figura con información escrita con código LaTeX | 66 |

Índice de cuadros

| | |
|---|----|
| 3.2. Una tabla de las primeras 10 filas del dataset mtcars con knitr::kable. | 42 |
| 3.3. Filas Agrupadas | 45 |
| 3.4. Una tabla generada con knitr::kable, pero con el paquete LaTeX: longtable. | 46 |
| 3.5. Tabla Demo en modo (Landscape) [*] | 49 |
| 3.6. Tabla con información escrita con código LaTeX | 65 |

Capítulo 1

Introducción

1.1. ¿Por qué utilizar R Markdown para escribir un trabajo fin de estudios?

La realización del trabajo fin de estudios requiere la entrega de un documento escrito con estilo formal por parte de los alumnos. Suele ser la primera vez que se enfrentan a esta tarea lo que les produce muchas dudas sobre su redacción y sobre cuestiones de estilo al escribir el documento. Los tutores de trabajos fin de estudios, de forma reiterada, tienen que resolver cuestiones de estos tipos a sus alumnos.

En cuanto a la redacción del trabajo existen consejos básicos, como por ejemplo, el no escribir nunca el trabajo fin de estudios en primera persona del singular, resulta más elegante utilizar la tercera persona del singular. Siempre es un buen consejo el leer algún trabajo fin de grado de algún alumno que haya obtenido una buena nota (algunos están disponibles en la biblioteca del centro, en papel o en formato electrónico). En cuanto a cuestiones de estilo, algunas de las recomendaciones más habituales son:

- Los títulos de las secciones no deben finalizar con dos puntos.
- No usar ningún tipo de decoración personal de forma reiterada, como por ejemplo: textos subrayados y centrados. Ante la duda, ver si ese tipo de decoración aparece en libros ya publicados.
- Siempre se debe escribir alguna leyenda o pequeña descripción, debajo o encima, de los gráficos o tablas de información que se presenten, junto con algún tipo de numeración. Así, si se quiere mencionar en el texto a estos elementos, se debe utilizar la numeración con la que han sido etiquetados. También, es posible etiquetar las expresiones matemáticas o resultados matemáticos, para que puedan ser identificados en los razonamientos que se hagan en el texto.
- Cuando en el texto se citan autores por sus trabajos (artículos, libro, capítulo de libro, ...) deben incluirse estos en la bibliografía del trabajo fin de estudios, además de referenciarlos con los sistemas habituales: o bien un número que identifique en la bibliografía a qué trabajo se está citando, o bien utilizar el apellido del autor principal seguido del año del trabajo.

Los alumnos generalmente usan programas como Microsoft Word (con licencia comercial) o LibreOffice Writer (con licencia gratuita) para escribir sus trabajos fin de estudios. Estas

aplicaciones disponen de muchas facilidades para llevar a cabo esta tarea, la principal ventaja es que al escribir se está viendo siempre como está quedando el documento. Otras habilidades son: permite definir secciones de distintos niveles, insertar gráficos y tablas de información permitiendo modificar de forma visual el lugar que ocupan, incluir índices de contenido, insertar ecuaciones matemáticas, etc.

En facultades de Matemáticas se les suele pedir a los alumnos que hagan un esfuerzo y usen el sistema gratuito LaTeX¹ con ayuda de algún entorno-editor. La razón principal de esto es que para su futuro profesional, tanto en la enseñanza como en la investigación, será un recurso muy útil, ya que facilita mucho el trabajo con expresiones matemáticas y consigue una impresión de calidad muy alta en los documentos pdf que se crean. Además, muchas revistas científicas en el campo de las matemáticas en especial, pero también en otros campos científicos, requieren que se envíen en formato LaTeX. A pesar de la calidad profesional que se consigue al crear estos documentos, tiene una gran desventaja frente a programas como Word, y es que cuando se escribe no se está viendo como queda, se necesita de un proceso de compilación para construir el fichero pdf en el que podrá verse su aspecto final.

En las asignaturas del Departamento de Estadística e Investigación Operativa se está usando de forma habitual el sistema R (lenguaje y entorno de trabajo). Muchos de los trabajos fin de estudios que se realizan relacionados con estas asignaturas llevan el uso de código R, que genera resultados numéricos, pero también resultados gráficos y en formato de tablas. Para integrar estos elementos en los programas mencionados anteriormente, se requiere en la mayoría de las ocasiones de las acciones de copiado y pegado en el portapapeles, para insertarlos en el documento. Es un trabajo muy artesanal. En muchas ocasiones, ocurre que bien porque se han detectado errores o porque se quiere volver a recalcular con unos nuevos parámetros, se tiene que repetir todo el proceso.

Una solución a estos inconvenientes ha sido la que ha aportado el lenguaje de marcas ligeras para escribir documentos, “Markdown”², y en especial la variante “R Markdown”³, desarrollada por el equipo de RStudio. A partir de ficheros creados con este lenguaje sencillo de marcas ligeras (se usan con extensiones: “.md” y “.Rmd”) es posible crear documentos en ficheros con formatos diferentes, como por ejemplo: “html” (páginas web), “pdf”, “docx”, “odt”, “epub”, etc.

R Markdown es un lenguaje de marcas ligeras para escribir documentos, pero que además con la ayuda del sistema R, permite incluir elementos producidos por el lenguaje R, (código R, gráficos, tablas, etc) en el mismo documento. Es decir, evita que tengamos que recurrir al copiado y pegado al portapapeles para construir un documento de resultados. Es una herramienta básica para el trabajo habitual de un profesional de la Estadística o de una persona que en su trabajo requiera la inclusión de resultados estadísticos.

Como desventaja de R Markdown es la citada anteriormente con el sistema LaTeX y es que al escribir no vemos visualmente cómo queda el documento. Pero como ventajas se podría indicar la capacidad de realizar todas las tareas que se le exigen a un programa de edición de textos: formato de texto básico (negrita, itálica, creación de listas numeradas y no numeradas, etc), inserción de gráficos y tablas, inserción de referencias a páginas web u otros elementos del documento, uso de secciones y subsecciones para estructurar el

¹<https://www.latex-project.org>

²<https://es.wikipedia.org/wiki/Markdown>

³<http://rmarkdown.rstudio.com>

documento, inserción de código fuente y evaluación del mismo en distintos lenguajes de programación (R, Python, SQL, etc), insercción de bibliografía, etc. Al tratarse de ficheros de texto no binarios es fácilmente utilizable en sistemas como Git⁴, el cual permite hacer un versionado de la historia del documento y también al requerir poco espacio editarlo directamente de forma online (plataformas como GitHub⁵) lo que posibilita la creación de documentos de forma colaborativa entre varios autores en distintos lugares físicos. Permite crear documentos tipo:

- libros, artículos, etc. para su publicación en papel, habitualmente en formato pdf, aunque también en formato epub o incluso en formato html (libros disponibles en internet),
- páginas web en formato html (página personal, un blog, etc.),
- presentaciones tipo powerpoint, tanto en formato html (html5) como en formato pdf (Beamer).

Es una herramienta que ha surgido muy recientemente, y actualmente no es muy conocida. Desde los desarrolladores de RStudio se está dando un empuje muy fuerte para que su uso desde la aplicación RStudio resulte cada vez más fácil.

Todos estos precedentes me han hecho ver que sería conveniente crear un documento que facilite la tarea de escribir un trabajo fin de estudios en estilo formal con ayuda del lenguaje de marcas R Markdown. Además, he creado una plantilla de carpetas y ficheros (almacenada en un fichero comprimido) con todos los elementos necesarios, para que facilite esta tarea a la persona que se inicia en su uso.

1.2. La estructura de un trabajo fin de estudios

En la realización de un trabajo fin de estudios es necesario recoger en un documento escrito varios capítulos que expliquen las tareas llevadas a cabo.

- **En el capítulo 1 o capítulo de introducción**, se deben enumerar los objetivos que se hayan marcado en el trabajo o proyecto, así como la descripción de los motivos por los que es importante plantearse dichos objetivos. También, se puede incluir aquí una *introducción general* a la temática que se pretende tratar, viendo “el estado del arte” o puntos de vista desde la que se ha estudiado esta temática en el momento actual, incluyendo las referencias más reseñables.
- **En el capítulo 2 o capítulo de análisis inicial**, se deben presentar y describir los elementos disponibles.

En el caso de ser “un trabajo estadístico con datos”, la descripción constaría principalmente de las siguientes etapas:

1. importación de los datos,
2. proceso de limpieza de los datos,
3. tratamiento de las observaciones outliers,
4. realizar una estadística descriptiva unidimensional de las variables que se han recogido, y

⁴<https://git-scm.com>

⁵<https://github.com>

5. seleccionar las variables más importantes para los objetivos marcados, eliminando aquellas variables que no aporten información significativa al estudio.

En el caso de ser “un tema teórico”, se deben presentar las definiciones y resultados conocidos que son necesarios para entender el tema de estudio más detalladamente, citando el material consultado.

- **En el capítulo 3 o capítulo de metodologías**, se deben desarrollar cada una de las metodologías de interés en el proyecto que lleven a la resolución de los objetivos propuestos. Se debe utilizar material para ilustrar, gráficos y/o tablas de resultados, que ayuden a la comprensión de los resultados obtenidos.

En el caso de un “trabajo teórico” deben aparecer las demostraciones de los resultados enunciados.

En el caso de un “trabajo estadístico” con datos reales será el momento de aplicar cada una de las metodologías apropiadas para resolver el problema, señalando las hipótesis en las que se ha trabajado y presentando los resultados más reseñables.

- **En el capítulo 4 o capítulo de conclusiones**, se deben de comparar todos los resultados obtenidos desde diferentes puntos de vista para poder obtener conclusiones acerca de los objetivos marcados. Si es el caso, podrían enumerarse un listado de recomendaciones.

En este capítulo, también es habitual finalizarlo incluyendo futuras líneas de trabajo para mejorar o ampliar el proyecto realizado.

- Se pueden añadir diferentes **apéndices** que contengan material que pueda ser necesario para describir el trabajo realizado. Este material, podría ser:

- pequeños tutoriales para iniciarse en el uso de alguna herramienta,
- código fuente utilizado para realizar el trabajo,
- etc.

- **Un apartado para la Bibliografía**, tanto la básica como la consultada: libros, artículos, páginas web, etc. Es importantísimo, que durante la argumentación llevada a cabo al escribir el trabajo se citen las referencias bibliográficas utilizadas en cada momento.

1.3. Instalación de las herramientas necesarias para usar R Markdown

Para escribir un trabajo fin de estudios con R Markdown es necesario que se instalen en nuestro ordenador las siguientes herramientas gratuitas:

- el lenguaje R,
- la aplicación RStudio,
- la utilidad Pandoc y
- el sistema LaTeX, que será el encargado de producir el documento pdf.

A continuación se indicará dónde encontrar los ficheros necesarios para instalar cada una de estas herramientas e información sobre cómo realizar el proceso de instalación, lo cual será posible en los tres sistemas operativos más conocidos: Windows, Linux y Mac.

1.3.1. Instalación de R

La página oficial del “Proyecto R” es: <https://www.r-project.org>, y la red de archivos R “CRAN” tiene su web principal en: <https://cran.r-project.org>.

R se encuentra disponible para los sistemas operativos más habituales, y en el siguiente enlace puede encontrarse la información para descargarlo: <https://cran.r-project.org/bin/>.

- Instalación de **R** para Windows: <https://cran.r-project.org/bin/windows/base>,
 - El fichero de descarga para la versión ‘R-3.4.2’ es: <https://cran.r-project.org/bin/windows/base/R-3.4.2-win.exe>.
- Para Mac se necesita descargar el fichero en formato “pkg” desde CRAN⁶.
 - La versión de R-3.4.2 sería: <https://cran.r-project.org/bin/macosx/R-3.4.2.pkg>.
- Para Linux, se podrá instalar desde la terminal. Por ejemplo, si se trata de una distribución Debian/Ubuntu se haría ejecutando la siguiente instrucción:

```
$ sudo apt-get install r-base
```

1.3.2. Instalación de RStudio

RStudio es una poderosa y productiva interfaz de usuario para trabajar con R. RStudio necesita que R esté previamente instalado. Es libre y de código abierto. Y funciona sobre Windows, Mac y Linux.

La página oficial de RStudio es: <https://www.rstudio.com>.

- La versión oficial de **RStudio Desktop** se puede descargar desde: <https://www.rstudio.com/products/rstudio/download/>. Están disponibles los ficheros binarios de instalación para los sistemas operativos: Windows, Mac y Linux (además está disponible el código fuente).

Una vez descargado el fichero con la instalación de RStudio (la versión Windows sería un fichero del siguiente tipo: **RStudio-1.1.383.exe**⁷), se hace doble click sobre el fichero para iniciar la instalación.

- Versión **RStudio Preview y Daily**. Habitualmente el equipo de desarrollo de RStudio ofrece la última versión en la que están trabajando (versión preview) para que pueda ser probada por los usuarios que lo deseen y puedan conocer cuales son las novedades que van a introducir en la próxima versión oficial.
 - La página para descargar la versión previa de **RStudio** es: <https://www.rstudio.com/products/rstudio/download/preview/> (RStudio-1.1.392.exe⁸).
 - También existe una página en RStudio en la que se pueden descargar las versiones que compilan de forma diaria: <https://dailies.rstudio.com>. Hay que tener presente que esta versión incorpora las novedades en las que trabajan a diario, pero suelen presentar bastantes inestabilidades.

⁶<https://cran.r-project.org/bin/macosx/>

⁷<https://download1.rstudio.org/RStudio-1.1.383.exe>

⁸<https://s3.amazonaws.com/rstudio-dailybuilds/RStudio-1.1.392.exe>

- **Nota Importante:** La instalación de cualquier versión de RStudio sobreescribe la versión anterior que se tenga instalada, pero mantiene cualquier personalización que se haya hecho de RStudio. Si se quisiera volver a utilizar la versión anterior que se tenía instalada porque la nueva tiene algún tipo de error o mal comportamiento, lo único que se tendría que hacer es ejecutar de nuevo su fichero de instalación. De ahí que se recomiende siempre guardar el fichero de instalación de la versión anterior para que RStudio siga funcionando correctamente.

1.3.3. Instalación de Pandoc

RStudio instala convenientemente la **herramienta Pandoc**, por lo que no será necesario que instalemos nada más. Al instalar nuevas versiones de RStudio también suele instalar nuevas versiones de Pandoc.

Aunque no es recomendable, también sería posible instalar Pandoc de una forma independiente desde la página web: <http://pandoc.org>, lo que supondría obtener una versión más actualizada de Pandoc que la que instala RStudio.

Para su instalación se recomienda consultar la información existente en la página web: <http://pandoc.org/installing.html>, aunque la página de descarga se encuentra en: <https://github.com/jgm/pandoc/releases/>.

La siguiente instrucción R nos indicará la versión que está instalada en nuestro sistema:

```
rmarkdown::pandoc_version()  
## [1] '1.19.2.1'
```

1.3.4. Instalación del sistema LaTeX

Con LaTeX instalado se pueden crear documentos pdf con calidad profesional, y además se puede integrar código R para crear informes con ayuda de R Markdown.

- Las distribuciones recomendadas para su instalación⁹ en cada uno de los principales sistemas operativos son:

- **TeX Live:** es la mayor distribución de LaTeX (y TeX) para GNU/Linux, Mac OS X y Windows: <http://www.tug.org/texlive/>.
- **MiKTeX** está indicada para Windows: <http://www.miktex.org/>.

MiKTeX para Windows se puede descargar en: <https://miktex.org/download>.

La versión actual de MiKTeX es: [basic-miktex-2.9.6361-x64.exe \(191.13 MB\)](https://miktex.org/download/basic-miktex-2.9.6361-x64.exe).

Nota: en esta página también pueden encontrarse los diferentes modos de instalación de Miktex sobre otros sistemas operativos.

- **MacTeX** es específica de Mac y está basada en TeX Live: <http://www.tug.org/mactex/>.

⁹<https://en.wikibooks.org/wiki/LaTeX/Installation>

- Estas distribuciones del sistema LaTeX, no necesariamente incluyen un editor de ficheros LaTeX, pero existen algunos **editores gratuitos** como:
 - **TeXStudio:** <http://texstudio.sourceforge.net>. Tiene muchas utilidades, pero podría resaltarse la que permite visualmente acceder a la mayoría de los símbolos matemáticos. Además, está disponible para los sistemas operativos más conocidos.
 - El propio **RStudio** se puede usar como editor de LaTeX.

Capítulo 2

Cómo utilizar la plantilla R Markdown para crear un trabajo fin de estudios con RStudio

2.1. Pasos a seguir para escribir un proyecto completo de trabajo fin de estudios

A continuación se recogen los pasos a seguir para crear un **documento pdf** con el trabajo fin de estudios con la ayuda de RStudio y utilizando la plantilla R Markdown desarrollada junto a este documento¹ para ser **impreso a doble cara**:

1.- **Descomprimir el fichero zip:** “MemoriaTFE.zip”². Al descomprimirlo se creará una carpeta llamada “MemoriaTFE” que contiene un proyecto RStudio con el material necesario para crear el trabajo fin de estudios.

Nota: se puede cambiar el nombre de la carpeta “MemoriaTFE” para identificarla mejor en nuestro sistema y moverla a cualquier otra carpeta de nuestro ordenador (incluso se podría llevar en un pen-usb).

En esta carpeta se encuentran los ficheros R Markdown:

- **tfe_principal.Rmd:** fichero R Markdown principal sobre el que hay que realizar el proceso de “Knit”.
- **capitulo01.Rmd, capitulo02.Rmd, ...:** cada uno de los capítulos del TFE.
- **apendice01.Rmd, ...:** ficheros en los que se coloca material adicional, como por ejemplo, código comentado.
- **prologo.Rmd:** se hacen comentarios sobre el trabajo o su autor, o se introduce en su lectura. También se podrían incluir aquí los agradecimientos (es opcional).
- **resumen.Rmd:** para el resumen en español del TFE.
- **abstract.Rmd:** para el resumen en inglés del TFE.

¹Este documento ha sido escrito utilizando esta plantilla R Markdown.

²Está disponible en: <http://destio.us.es/calvo/memoriatfe/MemoriaTFE.zip>

Nota: En el apéndice B se ha incluido el código de algunos de estos ficheros.

Otros ficheros en este carpeta son:

- “**logo.png**”: es el fichero gráfico que contiene el logo que se utiliza en la portada. Si se quiere utilizar otro distinto, el modo más rápido de cambiarlo, es sobreescribirlo con el nuevo fichero gráfico.
- “**acm-sig-proceedings-long-author-list.csl**” (numerado-autor-año) y “**methods-in-ecology-and-evolution.csl**” (autor-año): dos ficheros que contienen la definición de cómo se muestra la bibliografía.
- “**XXX.sty**”: existen varios ficheros con extensión “.sty” que usa LaTeX para cambiar el aspecto de algunas portadas no incluidos en la instalación de LaTeX.
- “**a_TODO.Rmd**”: fichero en el que anotar las tareas que quedan por realizar en el proyecto. Al leer documentación de material escrito o de internet, se deben anotar en este fichero las cuestiones que sean interesantes incluir en el proyecto del TFE y dónde.
- “**a_NEWS.Rmd**”: fichero en el que anotar las tareas que ya se han realizado. Se aconseja ordenarlas por fecha de realización, colocando arriba las últimas que se hayan incluido.
- “**a_README.Rmd**”: fichero en el que describir el proyecto que se está realizando, además de incluir las consideraciones que se han tenido en cuenta al realizar el proyecto. Por ejemplo, se podrían indicar qué paquetes R se tienen que instalar y cómo instalarlos.

También existen varias subcarpetas:

- **Carpeta “bib”**: para colocar los ficheros de bibliografía BibTeX.
- **Carpeta “figurasR”**: aquí se guardarán los ficheros gráficos que cree R.
- **Carpeta “graficos”**: aquí se deben colocar los ficheros gráficos (formatos: png, jpg, gif, pdf) que se incluirán en el documento (capturas de pantalla, esquemas gráficos, gráficos obtenidos de internet, etc). Es recomendable usar nombres de ficheros que los identifiquen al capítulo en el que se incluyen.
- **Carpeta “latex”**: contiene ficheros para personalizar el documento pdf resultante, además de cargar los paquetes LaTeX necesarios.
- **Carpeta “modelos”**: contiene los ficheros Rmd originales: tfe_principal.Rmd, capitulo01.Rmd, apendice01.Rmd, etc, que pueden usarse para hacer copias de ellos en el directorio principal, por ejemplo para crear nuevos capítulos, apéndices, borradores de capítulos, etc.
- **Carpeta “portadas”**: contiene ficheros con distintos modelos de portadas o páginas iniciales del trabajo fin de estudios.

2.- **Utilizar RStudio para editar** cada uno de los capítulos, apéndices y resto de ficheros Rmd, que constituirán el trabajo fin de estudios, TFE.

Se recomienda consultar el siguiente enlace a un fichero pdf: <https://github.com/rstudio/cheatsheets/raw/master/rstudio-ide.pdf>, que contiene de forma muy resumida la descripción de las principales características de RStudio.

Para ver en cualquier momento cómo queda el documento actualmente se hará click sobre el botón “Knit” de RStudio para que se realice el proceso de compilado. Si se hace cuando se tiene abierto el fichero “tfe_principal.Rmd” creará el documento completo, pero si se hace cuando se tiene abierto un fichero de capítulo o apéndice, por ejemplo “capitulo02.Rmd”, creará el pdf asociado a este capítulo.

El proceso que realiza “Knit” (compilado del pdf) consume tiempo. En las primeras fases de creación del TFE se irá añadiendo información por capítulos. Así, por ejemplo, si estamos trabajando sobre el fichero “capitulo02.Rmd” se podrá compilar más rápidamente este capítulo y ver como está quedando, y asegurándonos que no existen errores. Cuando se tengan todos los apartados del TFE si será el momento de compilar el fichero “tfe_principal.Rmd” para ver que todos los elementos encajan correctamente en el documento final.

Nota: Ejecutar en la consola de R el siguiente comando (del paquete knitr [16]) le pedirá a R que genere un fichero llamado “tfe_principal.R” que contendrá el código R contenido en el TFE:

```
knitr::purl("tfe_principal.Rmd")
```

En el próximo capítulo, se detallarán las principales habilidades del lenguaje de marcado ligero “Markdown” y también de “R Markdown” a través de ejemplos, para conocer cuáles son los elementos que se podrán incluir y cómo utilizarlos en el documento de TFE que se elaborará. Algunos de estos elementos permitarán:

- Organizar el documento en secciones, subsecciones, etc.
- Incluir gráficos.
- Incluir tablas con información.
- Incluir código R.
- Incluir expresiones matemáticas.
- Incluir bibliografía.
- Hacer referencias entre los distintos elementos del documento: referencias a ecuaciones, referencias a gráficos, referencias a tablas de información, referencias a la bibliografía.

3.- Copia de seguridad del Trabajo. Si todos los ficheros que se creen se guardan en la carpeta de la plantilla, lo único que habría que hacer para tener una copia de seguridad del trabajo sería copiar la carpeta completa o mejor comprimirla en un único fichero.

Se podría enviar por email al tutor o a cualquier otra persona ese fichero comprimido para que pueda revisar y/o reproducir el trabajo, por supuesto teniendo instalado: R, RStudio y LaTeX. También está la posibilidad de enviar únicamente el fichero pdf para su revisión.

En las siguientes capturas, puede observarse cómo es el proceso de trabajo con R Markdown en RStudio para generar el documento pdf del TFE.

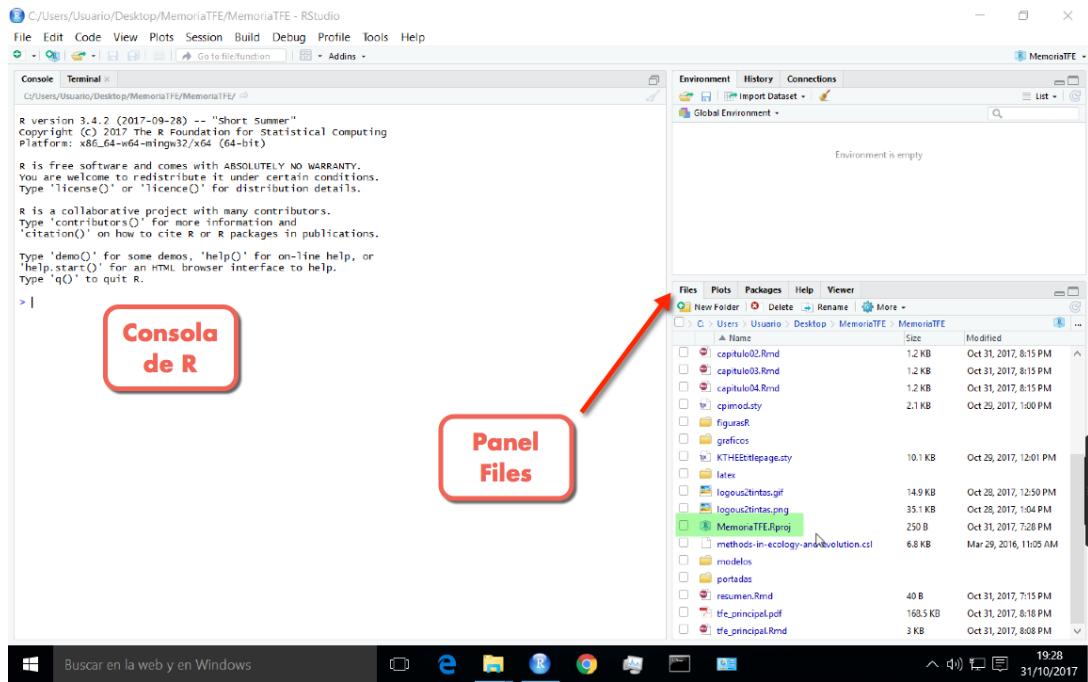


Figura 2.1: En esta captura se muestra la aplicación RStudio al abrir el proyecto de la plantilla: MemoriaTFE.Rproj.

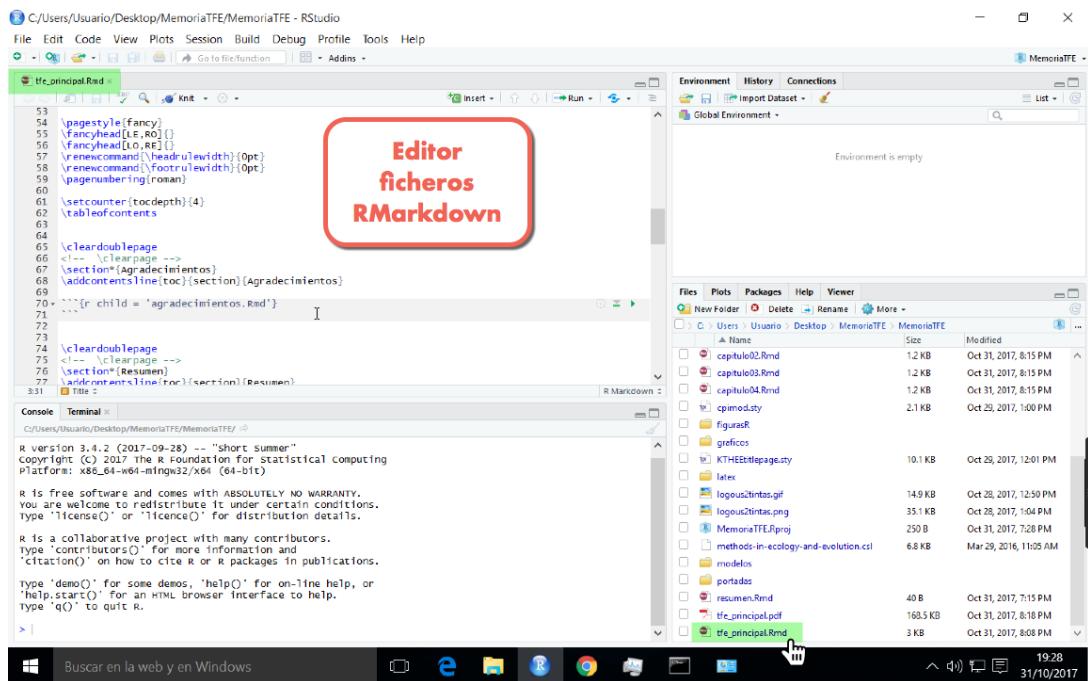


Figura 2.2: En esta captura se muestra el fichero tfe_principal.Rmd al hacer click sobre este fichero en el Panel Files, en la parte inferior derecha.

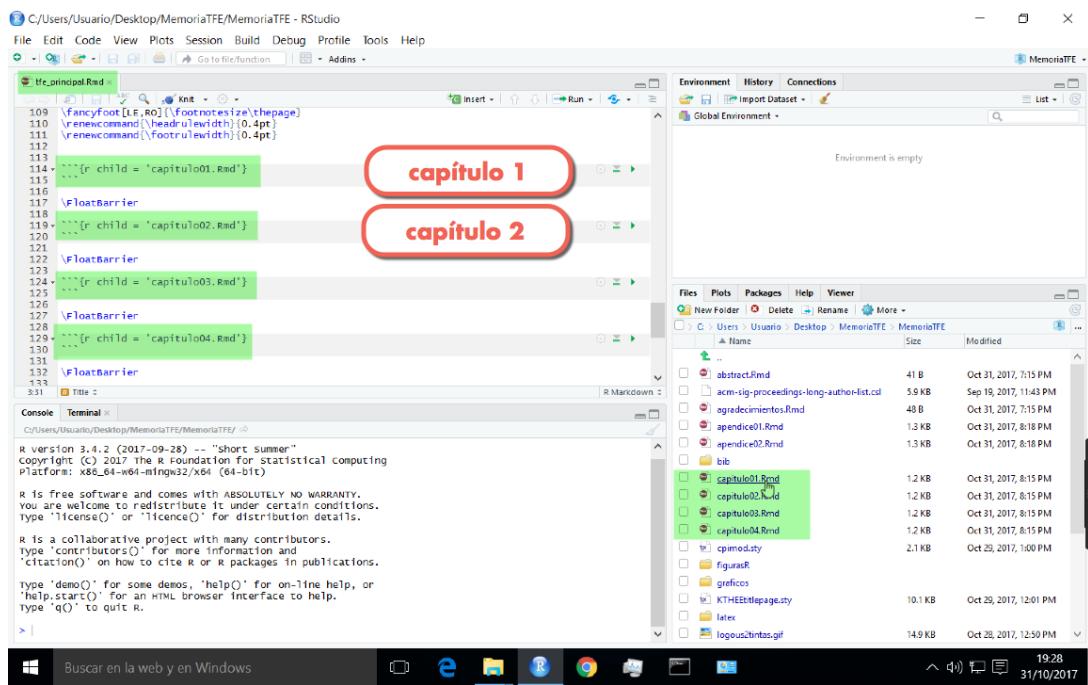


Figura 2.3: En esta captura se muestra la estructura del fichero tfe_principal.Rmd en la que aparece el código R Markdown para incluir los ficheros de los capítulos.

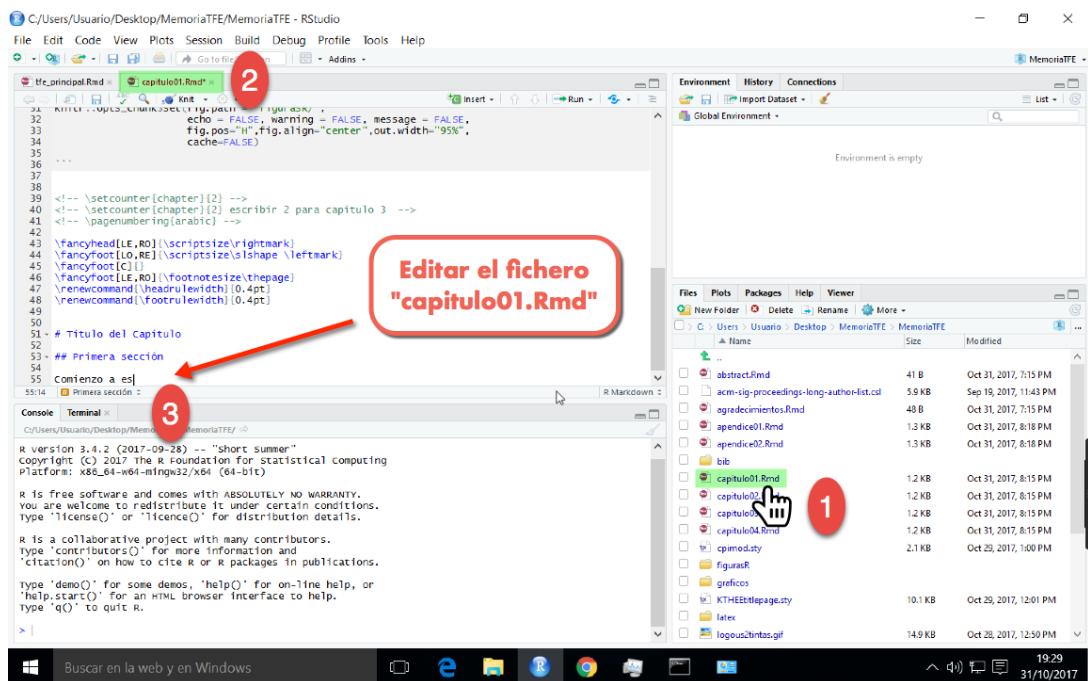


Figura 2.4: En esta captura se muestra el fichero capítulo01.Rmd que se ha abierto con ayuda del Panel Files. Una vez abierto está listo para ser editado con código R Markdown.

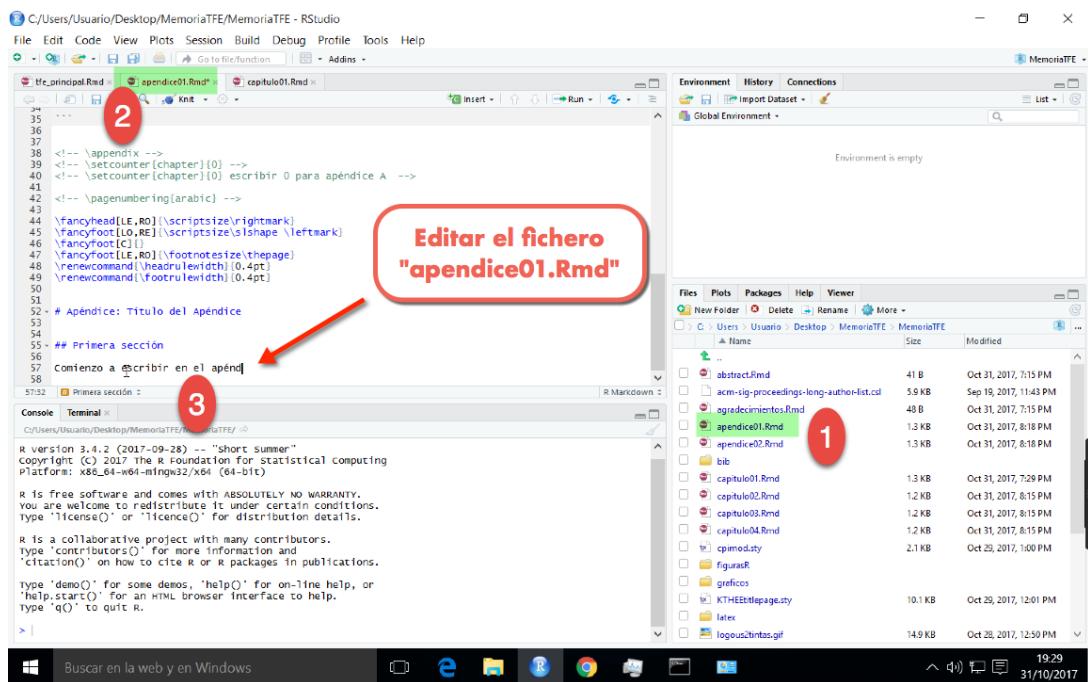


Figura 2.5: En esta captura se muestra el fichero apendice01.Rmd.

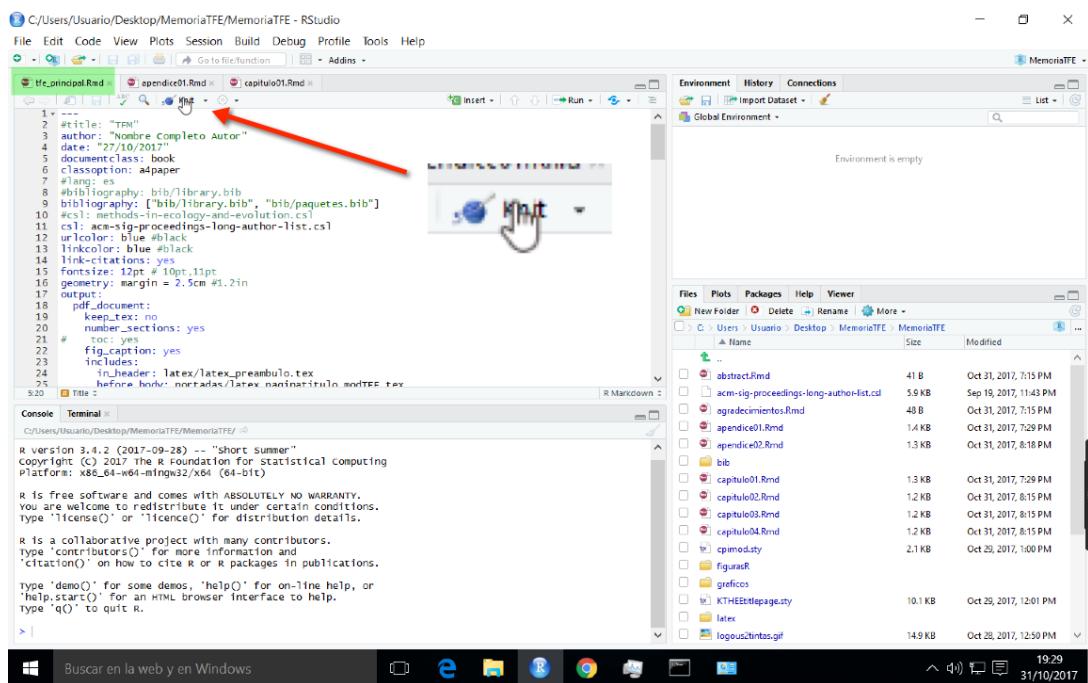


Figura 2.6: En esta captura se muestra que al hacer click sobre el icono 'Knit' al tener abierto el fichero tfe_principal.Rmd se inicia el proceso de compilación del documento.

2.1. PASOS A SEGUIR PARA ESCRIBIR UN PROYECTO COMPLETO DE TRABAJO FIN DE ESTUDIOS

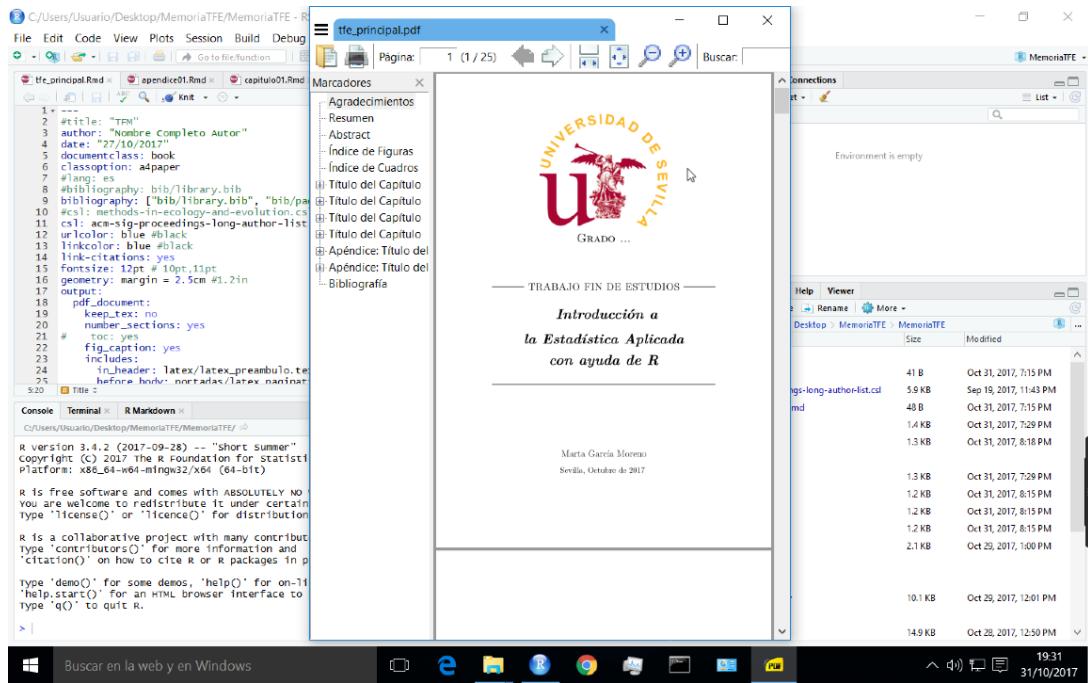


Figura 2.7: En esta captura se muestra, si no ha habido errores, el fichero pdf generado con el TFE en su estado actual.

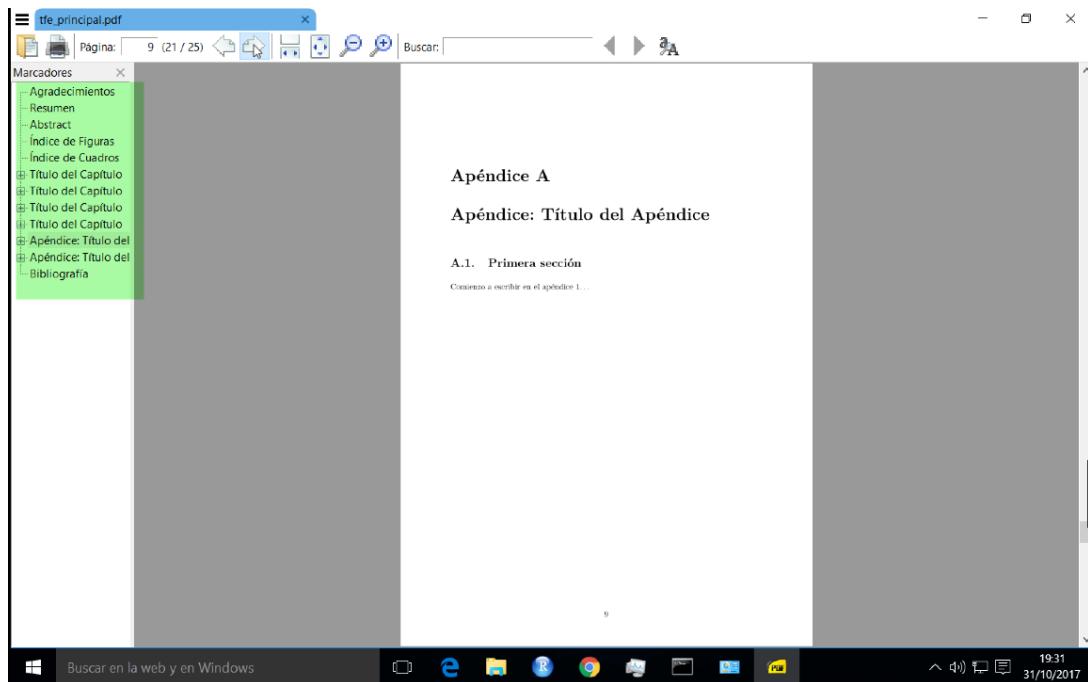


Figura 2.8: En esta captura se muestra el pdf generado sobre el que se puede navegar para realizar consultas.

2.1.1. La portada

La portada del trabajo fin de estudios es personalizable, y se han incluido algunos ficheros de ejemplo en la subcarpeta “portadas” que pueden modificarse/copiarse.

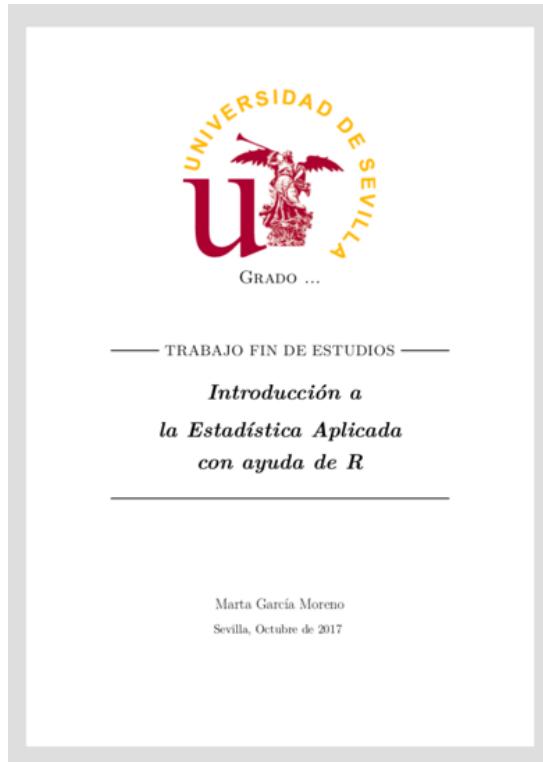


Figura 2.9: Se muestra la apariencia de la portada elegida para el TFE.

Para indicar cuál es la portada que se usará, se debe cambiar la opción “**before_body**”, en la cabecera YAML del fichero “`tfe_principal.Rmd`”, por el fichero de portada que se quiere utilizar.

En el siguiente ejemplo, actualmente se tiene seleccionada el fichero de portada: “`portadas/latex_paginatitulo_modTFE.tex`”.

```
includes:  
  in_header: latex/latex_preambulo.tex  
  before_body: portadas/latex_paginatitulo_modTFE.tex  
  after_body: latex/latex_anter_enddoc.tex
```

A continuación, se muestra el aspecto de las portadas que se han incluido en la plantilla “MemoriaTFE.zip”, organizadas en tres modelos diferentes.

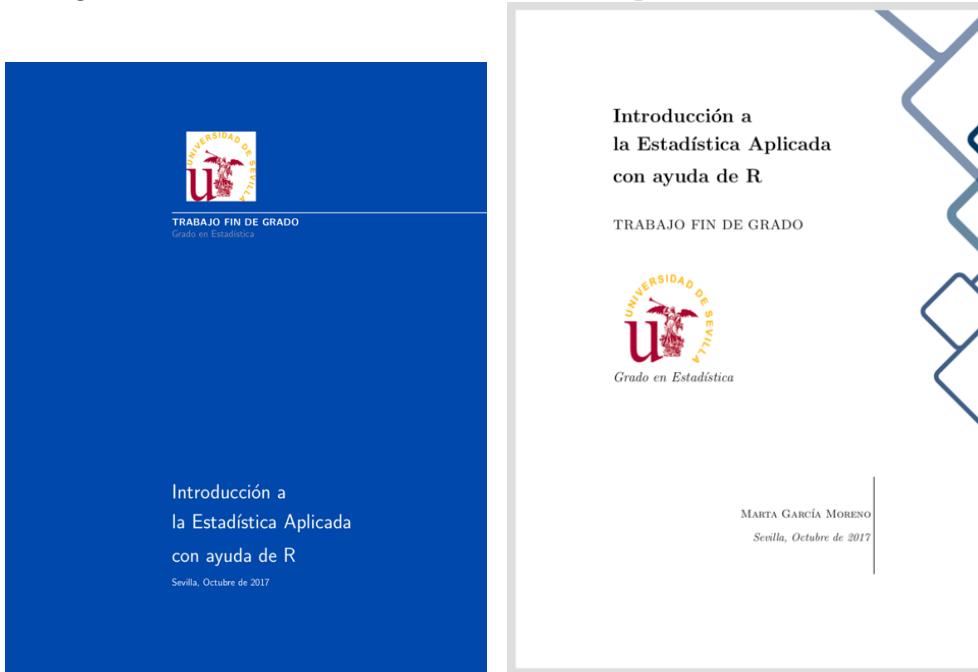
■ Los modelos LTXX:

Figura 2.10: Se muestran los dos estilos de portadas del modelo LT: LT01 y LT02.



■ Los modelos STXX:

Figura 2.11: Se muestran los dos estilos de portadas del modelo ST: ST01 y ST02.



- **Los modelos OVXX:** Se muestran los tres estilos de portadas del modelo OV: OV01, OV02 y OV03.

Figura 2.12: Se muestran los tres estilos de portadas de los modelos OV: OV01, OV02 y OV03.



En las siguientes referencias web puede encontrarse material adicional para poder particularizar la portada del documento:

- LaTeX Templates³
- Overleaf⁴
- ShareLaTeX⁵
- Stackexchange⁶

2.2. Otras plantillas para realizar un trabajo fin de estudios

En los siguientes enlaces pueden encontrarse otras plantillas a partir de las cuales es posible crear el documento a entregar como Trabajo Fin de Estudios:

- Plantilla en Overleaf para realizar Trabajo Fin de Estudios (Univ. Sevilla - Escuela de Informática)⁷: página del proyecto en Overleaf⁸.

Overleaf, <https://www.overleaf.com> es un sistema online para escribir documentos LaTeX y poder compartirlos. Existe la posibilidad de registrarse de forma gratuita en este servicio, el cual permite hacer muchas de las tareas habituales a la hora de escribir documentos.

³<https://www.latextemplates.com/cat/title-pages>

⁴<https://www.overleaf.com/gallery/tagged/title-page#.WfV-ydv3kXp>

⁵<https://es.sharelatex.com/blog/2013/08/09/thesis-series-pt5.html>

⁶<https://tex.stackexchange.com/questions/85904/showcase-of-beautiful-title-page-done-in-tex>

⁷<https://www.overleaf.com/11893526gjfbsdnvcckd#/45080032/>

⁸<https://www.overleaf.com/latex/templates/plantilla-latex-para-tfg/xqmjbwmgrjwr#.Wfbka9tDkmI>

Capítulo 3

Uso de R Markdown para escribir documentos

3.1. Aprendiendo lo básico del lenguaje de marcas: Markdown

En este primer apartado se enumeran las principales características del lenguaje de marcas ligeras “Markdown” con ayuda de ejemplos. Markdown es un lenguaje muy sencillo que permite elaborar documentos utilizando un editor de código de texto no binario, y obtener a partir de los mismos ficheros editados, distintos ficheros de salidas en formatos digitales tan conocidos como: html, pdf, docx, epub, etc.

3.1.1. Resaltado de texto

Para resaltar texto se pueden colocar marcas, bien entre asteriscos o guión bajo para conseguir *texto en itálica*, o bien entre dobles asteriscos o dobles guiones bajos para conseguir **texto en negrita**. A continuación puede verse un ejemplo de uso:

```
*italic*    **bold**    ** mal **  
  
_italic_    __bold__
```

Produce las siguientes salidas:

```
italic bold ** mal **  
italic bold
```

Atención: En “mal” no funciona el marcado por no comenzar el doble asterisco pegado al texto que se quiere resaltar.

3.1.2. Cabeceras o títulos de sección

Para conseguir un título de nivel 1 se usa una marca “#” y el texto del título. Para un título de nivel 2 se colocarían “##” seguidos del texto del título, y así sucesivamente

para los siguientes niveles de título. Los títulos de nivel 1 en esta plantilla corresponde a un nuevo capítulo, los títulos de nivel 2 serían las secciones, los títulos de nivel 3 serían las subsecciones, y así sucesivamente.

El siguiente código, produciría títulos de nivel 1, 2 y 3:

```
# Título nivel 1
## Título nivel 2
### Título nivel 3
```

3.1.3. Listas

3.1.3.1. Listas no numeradas

Para conseguir una lista no numerada, para cada elemento de la lista se usa una marca de nuevo elemento: '*' , '-' o '+', seguida del texto de ese elemento. De esta forma, se podrían añadir tantos elementos como se deseen. Para conseguir indentar una sublista dentro de algún elemento de una lista, en una nueva línea, **se añaden 4 espacios** y se vuelve a utilizar una marca de nuevo elemento. Para crear nuevos subniveles, seguiríamos la regla de añadir 4 espacios.

A continuación puede verse un ejemplo de uso:

```
* Item 1
* Item 2
  + Item 2a
  + Item 2b
```

Produce la siguiente salida:

- Item 1
- Item 2
 - Item 2a
 - Item 2b

3.1.3.2. Listas ordenadas o numeradas

Para conseguir listas numeradas se utiliza la misma metodología que para listas no numeradas, lo único que cambia es que las marcas para elementos nuevos son: números o letras.

A continuación puede verse un ejemplo de uso:

```
1. Item 1
  a. Item 1a
  b. Item 1b
2. Item 2
  a. Item 2a
  a. Item 2b
```

- 3. Item 3
 - + Item 3a
 - + Item 3b
- 4. Item 4

Produce la siguiente salida:

- 1. Item 1
 - a. Item 1a
 - b. Item 1b
- 2. Item 2
 - a. Item 2a
 - b. Item 2b
- 3. Item 3
 - Item 3a
 - Item 3b
- 4. Item 4

Nota: es importante tener en cuenta **la regla de los cuatro espacios** para continuar en un punto de la lista pero en distinto párrafo, y también para crear un subnivel nuevo dentro de una lista. En el siguiente ejemplo, puede verse cómo conseguir listas con subniveles correctamente.

- Lorem ipsum dolor sit amet,consectetuer adipiscing elit. Utpurus elit, vestibulum ut, pla-cerat ac, adipiscing vitae, fe-lis.

Curabitur dictum gravi-da mauris. Nam arcu libero,nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque.

Pe-llentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

* Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices.

Phasellus eu tellus sit amet tortor gravida placerat.

* Integer sapien est, iaculis in, pretium quis, viverra ac, nunc.

Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla.

- Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam.

Duis eget orci sit amet orci dignissim rutrum. Praesent eget sem vel leo ultrices bibendum. Aeneanfaucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla.

Produce la siguiente salida:

- Lorem ipsum dolor sit amet,consectetuer adipiscing elit. Ut purus elit, vestibulum ut, pla-cerat ac, adipiscing vitae, fe-lis.

Curabitur dictum gravi-da mauris. Nam arcu libero, nonummy eget, consectetur erid, vulputate a, magna. Donec vehicula augue eu neque.

Pe-llentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

- Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices.

Phasellus eu tellus sit amet tortor gravida placerat.

- Integer sapien est, iaculis in, pretium quis, viverra ac, nunc.

Praesent eget sem vel leo ultrices bibendum. Aeneanfaucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla.

- Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam.

Duis eget orci sit amet orci dignissim rutrum. Praesent eget sem vel leo ultrices bibendum. Aeneanfaucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla.

3.1.4. Producir saltos de línea manualmente

Para conseguir producir saltos de línea en una parte del texto, se utiliza la marca: “finalizar una línea con dos o más espacios”.

El siguiente código (después de “red,” existen dos espacios):

Roses are red,
Violets are blue.

Produce la siguiente salida:

Roses are red,
Violets are blue.

Nota: Para comenzar un nuevo párrafo, tan solo se tiene que dejar una línea en blanco de separación.

3.1.5. Insertar enlaces web

Para conseguir que aparezca un enlace a una página web tan solo habría que escribir una dirección http completa en el texto, como por ejemplo: `http://rmarkdown.rstudio.com`, producirá automáticamente un enlace como el siguiente: <http://rmarkdown.rstudio.com>.

También es posible crearlos, colocando la dirección web entre las marcas menor y mayor: `<url>` como se muestra en el siguiente código:

```
<http://rmarkdown.rstudio.com>
```

Produce la siguiente salida:

```
http://rmarkdown.rstudio.com
```

Si no se quiere que aparezca la dirección web, por ejemplo porque es una dirección muy larga, podría utilizarse las siguientes marcas `[arg1](arg2)`, donde “arg1” sería el texto que describe el enlace, y “arg2” sería la url específica. A continuación puede verse un ejemplo:

```
[frase enlace](http://rmarkdown.rstudio.com)
```

La salida siguiente es un enlace, que si se hace click sobre “frase enlace” se abrirá la página web asociada:

```
frase enlace
```

3.1.6. Insertar imágenes

Para conseguir incluir una imagen en nuestro documento se pueden utilizar las siguientes marcas: `![arg1](arg2)`, donde “arg1” es el texto de la leyenda que describe la imagen, y “arg2” puede ser la url de un gráfico que se encuentra en internet o el camino de un fichero gráfico local.

El código siguiente produce la imagen con el icono de RStudio (Figura 3.1) el cual se encuentra en internet:

```
![Insertar gráficos que estén en internet  
con markdown](https://www.rstudio.com/  
wp-content/uploads/2016/09/RStudio-Logo-Blue-Gray-250.png)
```



Figura 3.1: Insertar gráficos en internet con markdown

El siguiente código, inserta la imagen que se encuentra en el fichero “figuras/img.png” (Figura 3.2) que contiene una captura en la que se está usando el programa RStudio:

```
![Insertar imágenes locales con markdown](figuras/img.png)
```

```

39 # Leyendo datos
40
41 library(dplyr)
42 iris_tbl <- copy_to(sc, iris)
43 flights_tbl <- copy_to(sc, nycflights13::flights, "flights")
44 batting_tbl <- copy_to(sc, Lahman::Batting, "batting")
45
45:1 (Untitled) ▾ R Script ▾

```

History Spark

local

- batting
- flights
- iris

Figura 3.2: Insertar imágenes locales con markdown

3.1.7. Incluir enlaces web e imágenes con identificadores

3.1.7.1. Enlaces web

Para incluir un enlace web con un identificador se usan las siguientes marcas [arg1] [arg2] donde “arg1” es un texto explicativo y “arg2” es un identificador único. A continuación se puede ver un ejemplo de uso:

Un enlace [R Markdown] [id].

Se conseguirá el siguiente resultado:

Un enlace R Markdown.

Se define usando las marcas: [arg1]: arg2, donde “arg1” es el identificador, y “arg2” es la url con un texto entre dobles comillas con una descripción.

Esta definición se suele colocar en otra línea del documento, por ejemplo al final de la sección o del fichero Rmd.

A continuación, puede verse un ejemplo:

[id]: http://rmarkdown.rstudio.com "R Markdown"

3.1.7.2. Imágenes

Se puede insertar una imagen utilizando las siguientes marcas: ! [arg1] [arg2], donde “arg1” es un texto explicativo del gráfico y “arg2” es un identificador único. A continuación puede verse un ejemplo de su uso:

! [Leyenda que aparece bajo el gráfico incluido con identificadores] [id2]

En el documento aparece la siguiente figura:



Figura 3.3: Leyenda que aparece bajo el gráfico incluido con identificadores

La definición del identificador es la misma que en los enlaces web pero se coloca la ruta del fichero gráfico. A continuación puede verse un ejemplo:

[id2]: figuras/img.png "Leyenda que aparece bajo el gráfico
incluido con identificadores"

3.1.8. Notas al pie de página

Para conseguir que aparezca una nota al pie de página se utiliza el siguiente marcado `[^arg1]`, donde “arg1” será una etiqueta identificadora única, es decir no pueden estar repetidas.

En otra línea del documento, por ejemplo al final de la sección o del fichero Rmd, se tiene que definir el texto que aparecerá en la nota a pie de página. Se define usando las marcas: `[^arg1]: arg2`, donde “arg1” es la etiqueta identificadora, y “arg2” es el texto de la nota.

A continuación puede verse un ejemplo de uso:

Un nota a pie de página[^3.1]

[^3.1]: Aquí aparece el texto de la nota a pie de página:
[Introducción a R Markdown] (<http://rmarkdown.rstudio.com/lesson-1.html>).

Producirá una salida como la siguiente:

Un nota a pie de página^a

^aAquí aparece el texto de la nota a pie de página: [Introducción a R Markdown](#).

3.1.9. Resaltar bloques de texto

Para conseguir resaltar todo un bloque de texto, se coloca como marca el “signo mayor” o “>” delante de cada línea del bloque de texto.

A continuación puede verse un ejemplo de uso:

A friend once said:

```
> It's always better to give
> than to receive.
```

Produce la siguiente salida:

A friend once said:

It's always better to give than to receive.

3.1.10. Insertar código de lenguajes de programación

3.1.10.1. Bloques de código de más de una línea

Se pueden presentar bloques de código, habitualmente de algún lenguaje de programación, que se mostrarán tal como se escriben en una fuente de ancho fijo que no se evaluará.

Para conseguir esto se coloca el código entre 3 acentos agudos, como puede verse en el siguiente ejemplo:

```
```  
SELECT * FROM tablaalumnos
WHERE edad>10 and edad<20
ORDER BY apellidos
```
```

El resultado será el siguiente:

SELECT * FROM tablaalumnos
WHERE edad>10 and edad<20
ORDER BY apellidos

Para algunos lenguajes es posible resaltar las palabras claves del lenguaje, como puede verse en el siguiente ejemplo con “SQL” el que se ha añadido el texto “sql” a continuación de los primeros tres acentos graves:

```
```sql  
SELECT * FROM tablaalumnos
WHERE edad>10 and edad<20
ORDER BY apellidos
```
```

El resultado será el siguiente:

```
SELECT * FROM tablaalumnos
WHERE edad>10 and edad<20
ORDER BY apellidos
```

3.1.10.2. Código de lenguajes dentro del texto

Para mostrar código de algún lenguaje en el texto se debe escribir el código entre dos acentos graves, es decir, el marcado es: `arg1`, donde “arg1” es el código a mostrar. A continuación puede verse un ejemplo:

Definimos la función `add` para calcular la suma de dos números.

Se conseguirá resaltar el código escrito entre los dos acentos, como en el siguiente resultado:

Definimos la función add para calcular la suma de dos números.

3.1.11. Ecuaciones matemáticas con LaTeX

3.1.11.1. Expresión matemática en el texto

Para conseguir que se produzca una expresión matemática en el texto se usa el siguiente marcado: \$arg1\$, donde “arg1” es código LaTeX con expresiones matemáticas, como puede verse en el siguiente ejemplo:

Sea $f(x) = \frac{2^3}{3x}$ una función matemática.

Nota: visitar la web: <http://www.hostmath.com/> que permite utilizar un editor de ecuaciones LaTeX de modo online muy visual.

Se consigue un texto que contiene una expresión matemática, como se muestra en el siguiente resultado:

Sea $f(x) = \frac{2^3}{3x}$ una función matemática.

3.1.11.2. Ecuación matemática centrada

Si se escribe código LaTeX con expresiones matemáticas entre dos símbolos de doble dólar, como en el siguiente ejemplo:

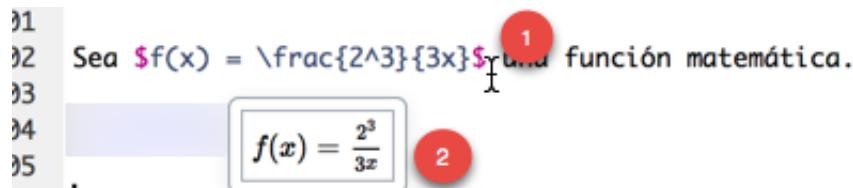
```
$$
f(x) = \frac{2^3}{3x}
$$
```

Se conseguirá una expresión matemática centrada en un párrafo aparte, como se ilustra en la siguiente imagen:

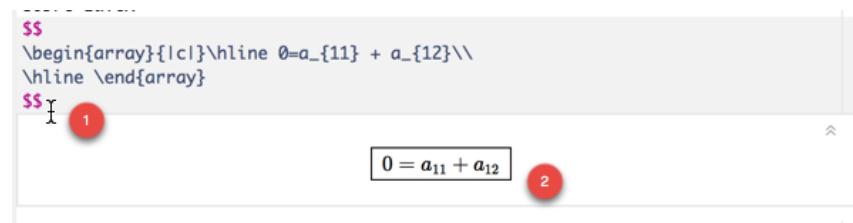
$$f(x) = \frac{2^3}{3x}$$

Nota: RStudio, en sus últimas versiones, muestra cómo queda la expresión matemática escrita en LaTeX,

- bien en una caja dinámica al acercar el ratón al símbolo del dólar “\$” o del doble dólar: “\$\$”,



- o bien la presenta dentro del fichero Rmd al terminar de escribir el doble dólar.



3.1.11.3. Más ejemplos de expresiones matemáticas con LaTeX

Los siguientes ejemplos se han obtenido de la página de wikipedia¹ sobre LaTeX

$$0 = a_{11} + a_{12}$$

$$x^{a+b} = x^a x^b$$

$$x_i = \sqrt[n]{\frac{a_i}{b_i}}$$

$$\begin{pmatrix} \alpha & \cdots & \beta^* \\ \vdots & \ddots & \vdots \\ \gamma^* & \cdots & \delta \end{pmatrix}$$

$$\sum_{0 \leq i \leq m, 0 < j < n} P(i, j)$$

$$\lim_{n \rightarrow \infty} \frac{n \cdot l}{2 \cdot r} = \pi$$

$$\int_{|x-x_0| < X_0} \Phi(x)$$

¹https://es.wikipedia.org/wiki/LaTeX#cite_ref-ref_duplicada_2_4-0

Estas expresiones matemáticas se consiguieron con el siguiente código en R Markdown:

```
$$
\begin{array}{|c|}\hline 0=a_{11} + a_{12}\\\hline \end{array}
$$

$$
x^{a+b}=x^ax^b
$$

$$
x_i=\sqrt[n]{\frac{a_i}{b_i}}
$$

$$
\begin{pmatrix} \alpha & \cdots & \beta^* \\ \vdots & \ddots & \vdots \\ \gamma^* & \cdots & \delta \end{pmatrix}
$$

$$
\sum_{0 \leq i \leq m} \sum_{0 < j < n} P(i, j)
$$

$$
\lim_{n \rightarrow \infty} \frac{n \cdot 1}{2 \cdot r} = \pi
$$

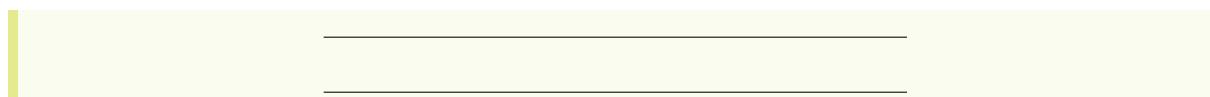
$$
\int_{\|x-x_0\| < X_0} \Phi(x)
$$
```

3.1.12. Incluir una línea horizontal

Si se escriben tres o más asteriscos o guiones:

```
*****
- - -
```

Se conseguirá el siguiente resultado:



3.1.13. Tablas

El siguiente código producirá una tabla con información, en la que se puede establecer una fila de cabecera. Las marcas utilizadas son guiones “-” para separar la fila de cabecera y la barra vertical “|” para separar columnas.

```
First Header | Second Header
----- | -----
Content Cell | Content Cell
Content Cell | Content Cell
```

La tabla que se obtiene a partir del código anterior es la siguiente:

| First Header | Second Header |
|--------------|---------------|
| Content Cell | Content Cell |
| Content Cell | Content Cell |

Nota: Se recomienda visitar la página de “Tables Generator” que permite de forma interactiva crear tablas en código Markdown: https://www.tablesgenerator.com/markdown_tables.

3.1.14. Miscelánea

En Markdown también es posible escribir superíndices, subíndices y tachar el texto, colocando el texto entre las marcas: ^, ~ y ~~ , respectivamente. A continuación pueden verse algunos ejemplos de uso:

superíndice^2^

subíndice_2~

~~texto tachado~~

Se conseguirá el siguiente resultado:

superíndice²

subíndice₂

~~texto tachado~~

3.2. R Markdown

Con R Markdown se puede no solamente utilizar el lenguaje de marcas “Markdown”, sino que además se pueden incluir elementos en el documento que se generen con la ayuda del lenguaje R de forma automática, es decir, se podrán incluir todo tipo de gráficos, tablas con información resumida, y cualquier otro tipo de cálculo que esté relacionado con el trabajo fin de estudios.

En los siguientes enlaces puede encontrarse información muy útil sobre R Markdown:

- Resumen rápido sobre R Markdown (pdf)²
- Guía de referencia de R Markdown (pdf)³

3.2.1. Cómo insertar código R en R Markdown

3.2.1.1. Código R dentro del texto

Para insertar código R en el texto se usa las marcas: `r arg1`, donde “arg1” es una expresión en lenguaje R que se evalúa y devuelve un valor que mostrará.

Si se escribe el siguiente texto con código R insertado entre las marcas: `r arg1`

Se estudiaron `r nrow(cars)` coches.

Se evalúa la expresión R: `nrow(cars)` (se sabe que `nrow()` obtiene el número de filas del data.frame “cars”), y se coloca el resultado (que en este caso es “50”), como puede verse en la siguiente salida:

Se estudiaron 50 coches.

Nota: si se cambiase el número de filas del data.frame “cars” con código R antes de encontrarse esta expresión, sin cambiarla, aparecería el nuevo valor actualizado.

3.2.1.2. Ejecución de bloques de código R

Para indicarle a R Markdown que se va a escribir un bloque de código R (o “chunk”) se utilizan las siguientes marcas (tres acentos graves con r entre llaves y finaliza con tres acentos graves):

```
```{r etiqueta,opcioneschunks}
```

bloquecodigoR

```
```
```

²<https://github.com/rstudio/cheatsheets/raw/master/rmarkdown-2.0.pdf>

³<https://www.rstudio.com/wp-content/uploads/2015/03/rmarkdown-reference.pdf>

Donde:

- “bloquecodigoR”: el código R que se necesite ejecutar con todas las líneas que se necesiten. Incluso podría contener una llamada a la función R: `source("fichero.R")`, que ejecutaría todo el código R que se encuentre en “fichero.R”.
- “etiqueta” (opcional): es un identificador único para el bloque de código. RStudio permite navegar entre los bloques de código el cual se podrá reconocer si se ha utilizado una etiqueta que realmente lo identifique. El inconveniente es que no se pueden repetir etiquetas, ya que se produciría un error por el cual no se compilaría el documento.
- “opcioneschunks” (opcionales, con algunas opciones con valores predefinidos por defecto): es una lista de opciones que definen qué se va a hacer con el código R del chunk y con los resultados obtenidos, si se ha evaluado el código R. Por ejemplo, hay dos opciones que por defecto tienen los siguientes valores: `echo=TRUE, eval=TRUE` (si no se escriben “opcioneschunks”) que significan, respectivamente, que se va a mostrar el código R escrito en el documento que se generará, y además se evaluará el código R y si no se producen errores se mostrará la información que en el código R se indique.

Nota: Con la función R: `knitr::opts_chunk$set()`, se pueden cambiar cuáles son los valores por defecto de las opciones de todos los chunks del documento Rmd. A continuación, se muestra como cambiar el valor por defecto para `echo` a `FALSE` lo que haría que por defecto no se muestre el código R escrito en los bloques de código R (habitual cuando se realiza un informe para una empresa):

```
```{r include=FALSE}
knitr::opts_chunk$set(echo = FALSE, fig.path = 'figurasR/')
```

```

A continuación, vemos un ejemplo en el que el código R se mostrará y después de evaluarse se imprimirán los resultados:

```
```{r}
summary(cars$dist)
summary(cars$speed)
```

```

equivalente a:

```
```{r echo=TRUE, eval=TRUE}
summary(cars$dist)
summary(cars$speed)
```

```

Ambos producirán el mismo resultado, línea a línea, se iría mostrando el código R y el resultado de evaluar la línea:

```
summary(cars$dist)

##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
##      2.00   26.00  36.00    42.98  56.00 120.00
```

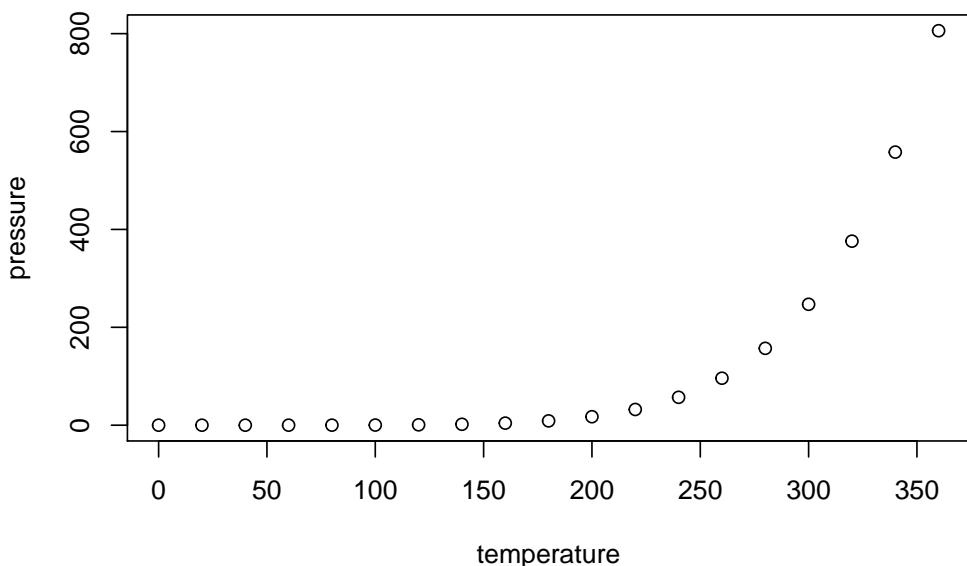
```
summary(cars$speed)
```

```
##   Min. 1st Qu. Median   Mean 3rd Qu.   Max.
##     4.0    12.0    15.0    15.4    19.0    25.0
```

En cambio, en el siguiente “chunk” o bloque de código R, atendiendo a las opciones del chunk no se mostrará el código (`echo=FALSE`), pero sí se evaluará e imprimirán los resultados que en este caso es un gráfico, además aparecen opciones adicionales que centra la figura y utiliza un 80 % del ancho del texto para el gráfico:

```
```{r echo=FALSE,fig.align='center',out.width="80%"}
plot(pressure)
```
```

Se produce la siguiente salida, en este caso gráfica:



Se pueden emplear porcentajes o valores en centímetros u otras unidades en la opción: `out.width="7cm"`. También existe la opción: “`out.height`”.

La habilidad especial de R Markdown está en incluir nuestro análisis de datos directamente en el documento, y cualquier cambio que se realice en el código R se actualizará en el documento.

```
```{r}
variable_t = 3 * 4
````
```

```
variable_t = 3 * 4
```

“`variable_t`” tiene un valor de: `r variable_t`.

“`variable_t`” tiene un valor de: 12.

Para obtener el valor de “variable_t” en el texto, se ha usado el acento grave: ` (backticks), como se comentó anteriormente.

Para realizar un cálculo rápido podría escribirse: $3 * 6 = `r 3*6`$, lo que produce: $3 * 6 = 18$.

A continuación, pueden verse los efectos del uso de las opciones “eval” y “echo” sobre los chunks de código R.

- **Ejemplo 1:**

El siguiente código:

```
```{r eval=FALSE, echo=TRUE}
variable_x1 = 3 * 4
variable_x1
```
```

Produce:

```
variable_x1 = 3 * 4
variable_x1
```

Es decir, muestra el código pero no se realiza ninguna evaluación.

- **Ejemplo 2:**

El siguiente código:

```
```{r eval=TRUE, echo=TRUE}
variable_x2 = 3 * 4
variable_x2
```
```

Produce:

```
variable_x2 = 3 * 4
variable_x2

## [1] 12
```

Es decir, muestra el código, lo evalúa y devuelve el valor del objeto `variable_x2`.

- **Ejemplo 3:**

El código:

```
```{r eval=TRUE, echo=FALSE}
variable_x3 = 3 * 4
variable_x3
```
```

En este caso no se mostraría el código, pero sí se evaluaría el código y se mostrarían los resultados. **Esta combinación es muy habitual en los informes de una empresa, en los que no se está interesado en el código que lo ha generado sino en los resultados obtenidos (generalmente, gráficos y tablas de información).**

```
## [1] 12
```

- **Ejemplo 4:**

El código:

```
```{r eval=FALSE, echo=FALSE}
variable_x3 = 3 * 4
variable_x3
````
```

En este caso no se mostraría el código y además no se ejecuta nada.

- **Ejemplo 5:**

El uso de `include=FALSE`, como en el siguiente código:

```
```{r include=FALSE}
variable_x3 = 3 * 12
variable_x3
````
```

En este caso no se mostraría el código, no se mostrarían los resultados, pero sí que se evaluaría el código R (“variable_x3” contendría “36”, el resultado de “3*12”):

3.2.1.3. Cómo mostrar código R y las opciones del chunk utilizadas

Si se quiere mostrar un chunk de código con las opciones utilizadas en un documento, se puede añadir una expresión r en línea que genere una cadena vacía (`r ' ') antes de la cabecera del chunk, e indentar el chunk de código con 4 espacios. Por ejemplo:

```
`r ' '
# chunk código literal
````
```

Después de compilar el documento, la expresión en línea desaparecerá y se verá lo siguiente:

```
```{r echo=FALSE}
# chunk código literal
````
```

### 3.2.1.4. Cómo insertar gráficos generados con R

Como ya se ha visto antes, pueden insertarse gráficos creados con R en el documento. Además, se podrán hacer referencias a ellos, de varias formas.

Una de ellas es utilizando el comando **LaTeX**: `\ref{}`. Se ha añadido una etiqueta al gráfico, llamada “irisgraph”, por lo se podrá hacer referencia a él escribiendo, por ejemplo: `ver la figura \ref{irisgraph}`, lo que producirá “ver la figura 3.4”.

```
```{r fig.pos="H",fig.cap = "Explicación figura. \\label{irisgraph}",
fig.height=4, fig.width=4}
plot(iris)
````
```

```
plot(iris)
```

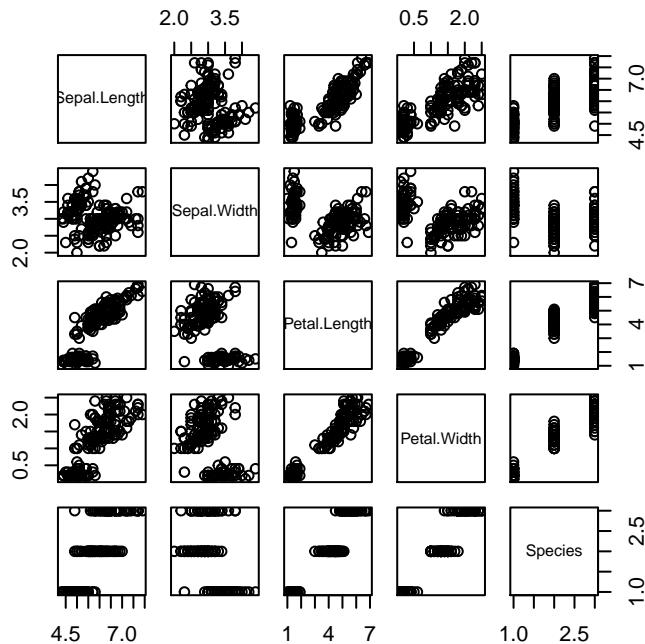


Figura 3.4: Explicación figura.

También se pueden hacer referencias a secciones, como por ejemplo: “en la sección 3.2.4” (en la sección `\ref{formulas}`). La etiqueta “formulas” se ha escrito sobre una subsección en el texto más adelante, de la siguiente forma:

```
Referencias a fórmulas matemáticas LaTeX en R Markdown {#formulas}
```

Si se escribe el siguiente chunk de código R:

```
```{r chunk-label, echo = FALSE, fig.cap = 'Una explicación del gráfico.'}
1 + 1 # suma igual a 2
rnorm(10) # 10 números aleatorios
plot(dist ~ speed, cars) # un diagrama dispersión
```
```

produce la siguiente salida:

```
[1] 2
[1] 0.76362283 2.17558935 -0.43343553 -0.32304168 -1.07257955
[6] -1.46054048 -0.10511143 -0.66711319 -0.09274044 -0.05704521
```

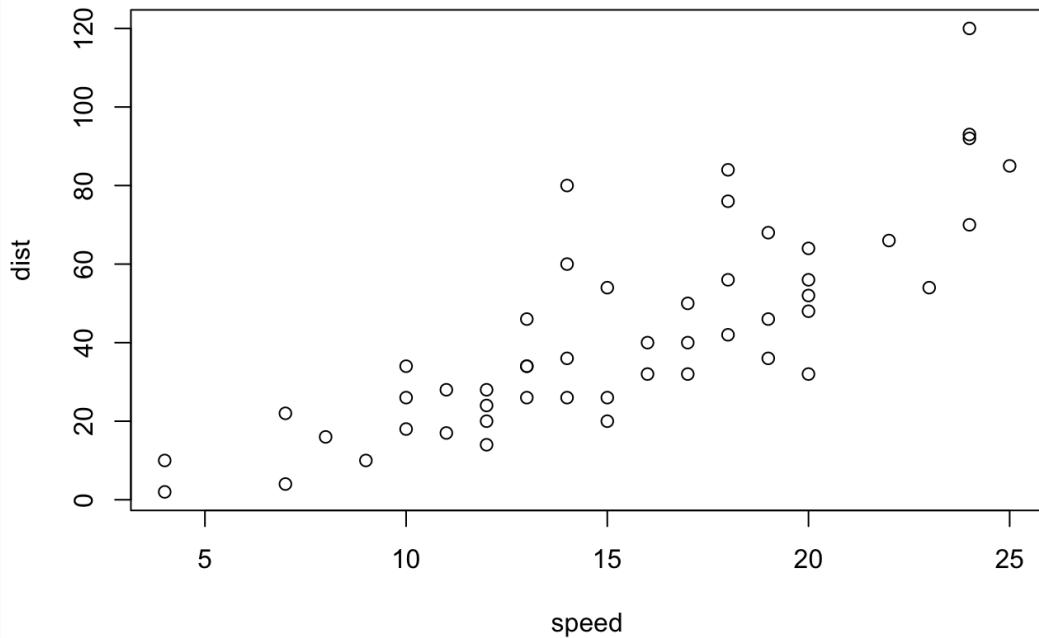


Figura 3.5: Una explicación del gráfico.

### 3.2.1.5. Insertar 2 gráficos R frente a frente

Se pueden insertar 2 gráficos frente a frente utilizando las siguientes opciones del chunk de código R que se muestran en el siguiente ejemplo:

```
```{r out.width='45%',fig.show='hold',fig.align='center',
fig.cap="Dos gráficos R cara a cara"}
par(mar = c(4, 4, 0.1, 0.1))
plot(pressure, pch = 19, type = "b")
plot(cars, pch = 19)
```

```

Produce la salida que se muestra a continuación:

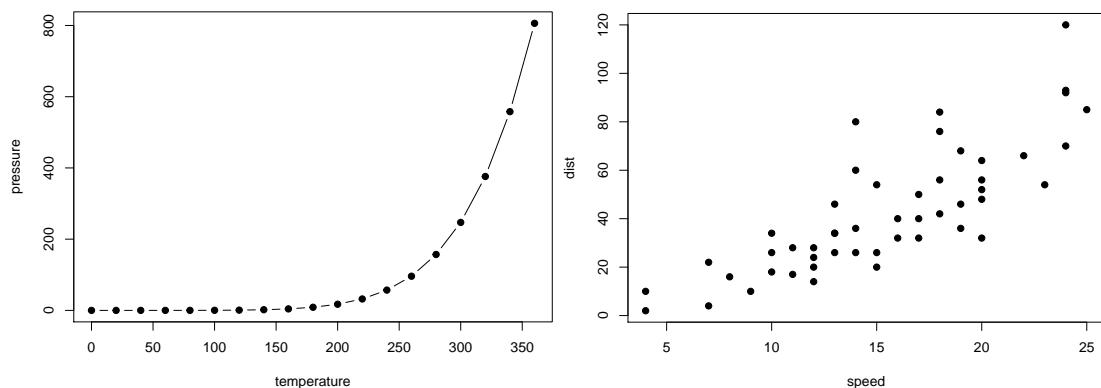


Figura 3.6: Dos gráficos R cara a cara

- `out.width`: permite especificar el ancho que ocupará el gráfico en porcentaje respecto al ancho del documento. En este caso se ha utilizado el ancho “45 %” para que tengan cabida los 2 gráficos generados con R.

### 3.2.1.6. Insertar gráficos generados con ggplot2

El paquete R “ggplot2” es uno de los más conocidos y utilizados en R, por su facilidad y potencia para construir gráficos estadísticos. Se puede encontrar ayuda de ggplot2 en los siguientes enlaces:

- Resumen rápido ggplot2 (pdf)<sup>4</sup>
- <http://www.cookbook-r.com/Graphs/>

Veamos un primer ejemplo. Antes de construir el gráfico con ggplot2, se preparan los datos necesarios a partir del dataset “mtcars” con ayuda del paquete R “dplyr” (es característico de este paquete el uso de “miniherramientas encadenadas”). Ambos paquetes han sido desarrollados por el equipo de RStudio [14]. Se usan las opciones de chunks por defecto.

```
library(dplyr)
d = mtcars %>% select(cyl,carb) # selecciona variables
head(d)

cyl carb
Mazda RX4 6 4
Mazda RX4 Wag 6 4
Datsun 710 4 1
Hornet 4 Drive 6 1
Hornet Sportabout 8 2
Valiant 6 1

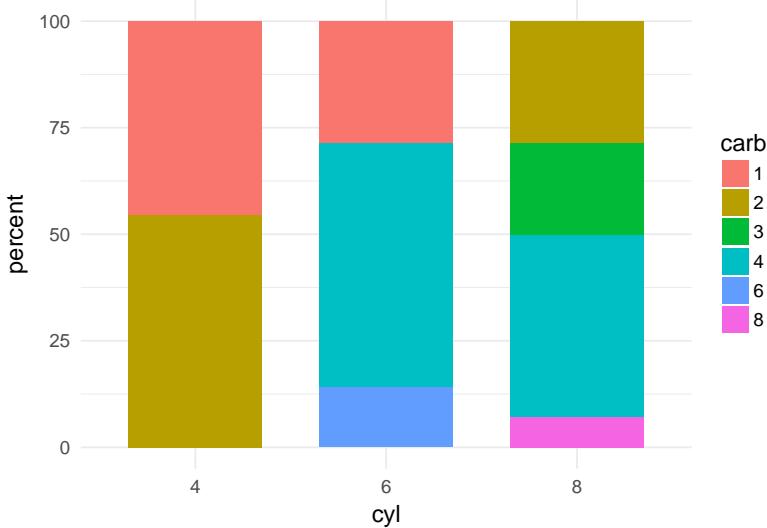
d2 <- d %>%
 group_by(cyl,carb) %>% # agrupa por variables
 summarise(count=n()) %>% # resume calculando nuevas variables
 mutate(perc=count/sum(count)) # calcula nuevas variables
head(d2)

A tibble: 6 x 4
Groups: cyl [3]
cyl carb count perc
<dbl> <dbl> <int> <dbl>
1 4 1 5 0.4545455
2 4 2 6 0.5454545
3 6 1 2 0.2857143
4 6 4 4 0.5714286
5 6 6 1 0.1428571
6 8 2 4 0.2857143
```

<sup>4</sup><https://github.com/rstudio/cheatsheets/raw/master/data-visualization-2.1.pdf>

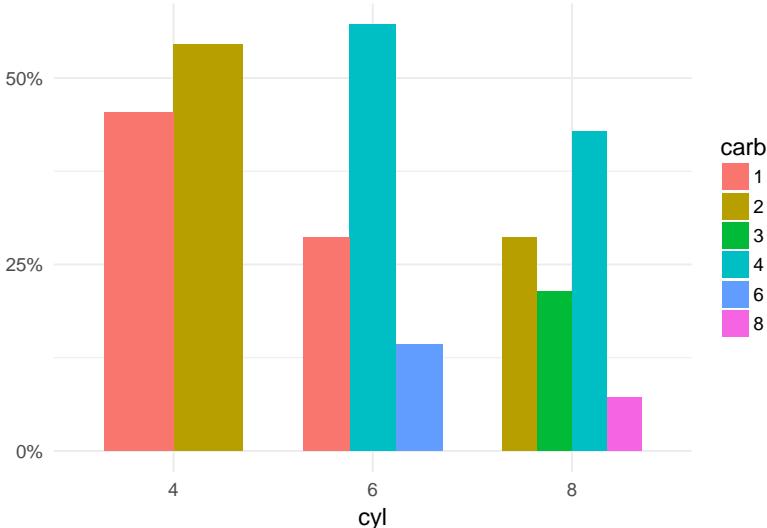
Se carga la librería “ggplot2” y se usa la función principal `ggplot` con la geometría `geom_bar` para construir un diagrama de barras apiladas. Para controlar el tamaño que ocupará en el documento se ha utilizado la opción del chunk: `out.height="7cm"`.

```
library(ggplot2)
ggplot(d2, aes(x = factor(cyl), y = perc*100, fill = factor(carb))) +
 geom_bar(stat="identity", width = 0.7) +
 labs(x = "cyl", y = "percent", fill = "carb") +
 theme_minimal(base_size = 14)
```



Ahora se construye un **diagrama de barras agrupadas** (`position="dodge"`), pero se construyen las etiquetas del eje de las Y con ayuda de `scales::percent()`:

```
brks <- c(0, 0.25, 0.5, 0.75, 1)
ggplot(d2, aes(x = factor(cyl), y = perc, fill = factor(carb))) +
 geom_bar(stat="identity", width = 0.7, position="dodge") +
 scale_y_continuous(breaks = brks, labels = scales::percent(brks)) +
 labs(x = "cyl", y = NULL, fill = "carb") +
 theme_minimal(base_size = 14)
```



Con el paquete R “ggplot2”, como con otros paquetes de R, se pueden crear gráficos con una calidad muy alta. En el siguiente ejemplo se muestra una nube de puntos en la que se ha incluido un modelo de regresión, y además se han distinguido los puntos con distintos colores según el valor de una variable (en este ejemplo: “cyl”).

```
library(ggplot2)
data(mtcars)
migraf <- ggplot(data = mtcars, aes(x = wt, y = mpg, col = cyl))
migraf <- migraf + geom_point(alpha = 1/2, size = 5)
migraf <- migraf + geom_smooth(method = "lm")
migraf
```

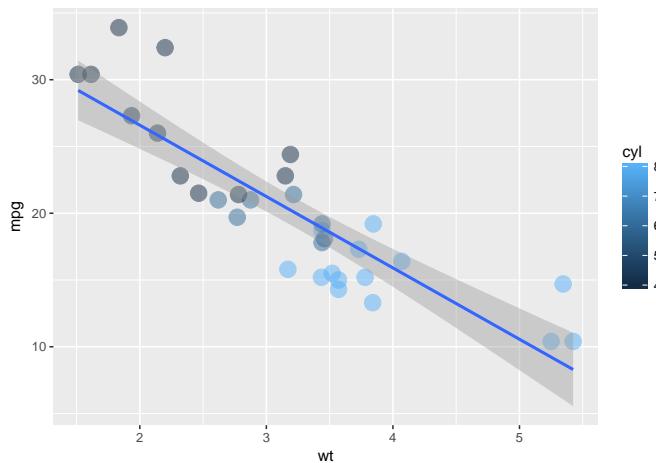


Figura 3.7: Gráfico creado con ggplot2 con una nube de puntos y el modelo de regresión lineal.

También se pueden incluir mapas con ayuda de R, como puede verse en el siguiente ejemplo:

```
```{r mapaSevilla,cache=TRUE,message=FALSE,out.width="90%",
fig.cap="En esta gráfico se muestra un mapa de carreteras
alrededor de Sevilla generado con ggmap."}
library(ggmap)
kcMap <- get_map(location = "Sevilla", zoom = 9, maptype = 'roadmap',
language = "es")
kcMap <- ggmap(kcMap)
misPuntos = as.data.frame((cbind(lon=runif(5,min=range(kcMap$data["lon"])[1],
max=range(kcMap$data["lon"])[2]),
lat=runif(5,min=range(kcMap$data["lat"])[1],
max=range(kcMap$data["lat"])[2]))))

kcMap <- kcMap + geom_point(data = misPuntos, aes(x = lon, y = lat, size = 5))
kcMap
```
```

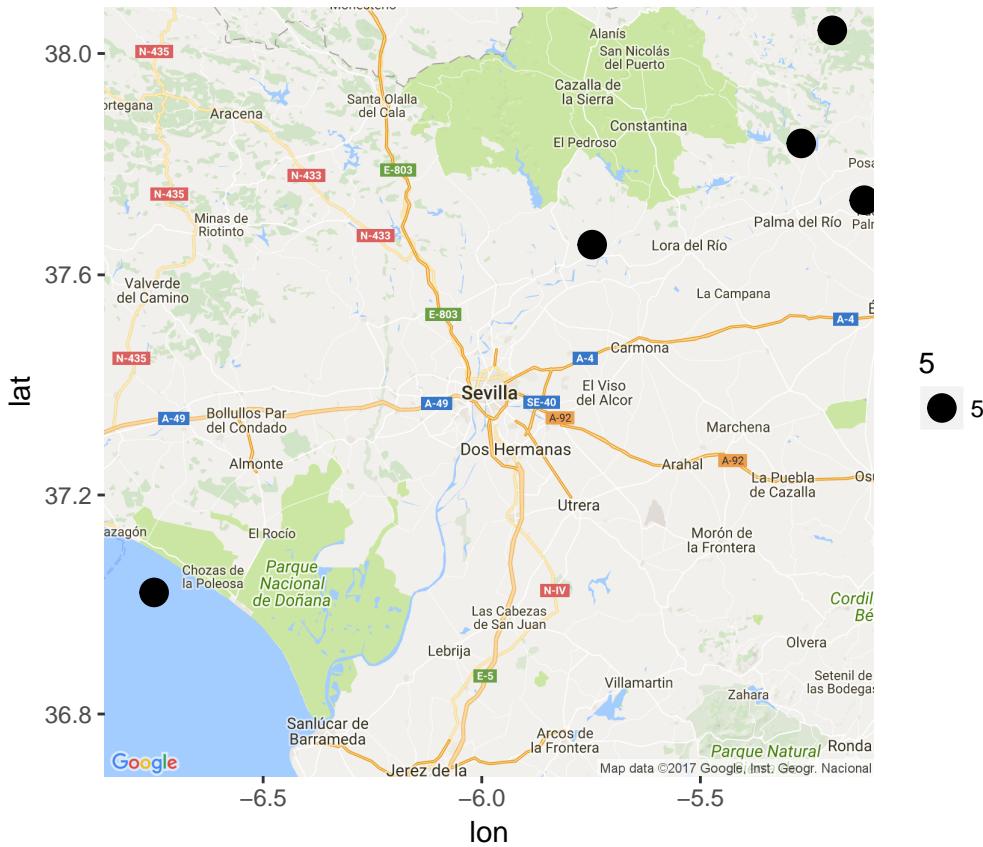


Figura 3.8: En esta gráfico se muestra un mapa de carreteras alrededor de Sevilla generado con ggmap.

**Nota:** Para la creación de este mapa, R tiene que descargar información desde internet, lo que le lleva un tiempo adicional. En esta ocasión se ha añadido una etiqueta identificativa “mapaSevilla” (tiene que ser única) en las opciones del chunk de código y se ha usado la opción “cache=TRUE”. Esta última opción, en el proceso de “Knit” o compilado del documento R Markdown para convertirlo en un fichero pdf, significa que R creará el gráfico del mapa una vez y lo guardará, de manera que mientras no haya cambios en ese chunk, no necesitará generararlo de nuevo en otros procesos de compilado, lo que hará que se realice en menos tiempo.

### 3.2.2. Cómo incluir ficheros gráficos en R Markdown

Para incluir un fichero gráfico en formato “jpg”, “png” o “gif” puede usarse la función R: `knitr::include_graphics()` de la librería “knitr”, [16].

A continuación, en el siguiente ejemplo, puede verse como se ha usado esta función, además de utilizar algunas opciones del chunk:

- `fig.cap`: permite definir la leyenda que describe la figura, además de incluir una etiqueta “figura01” para poder referenciar la figura.
- `fig.align`: permite definir la justificación del gráfico en el ancho del texto. En este caso, se ha usado “center”, que centrará el gráfico.
- `out.width`: permite definir el ancho que va a ocupar el gráfico. En este caso, el “80%” del ancho.

Las opciones de los chunks de código R para gráficos, se pueden usar tanto para gráficos creados con R como para ficheros gráficos.

```
```{r figura01, fig.cap = "\\label{figura01}Página oficial de R.",
fig.align='center',out.width='80%'}
knitr::include_graphics("figuras/oficialR.png")
```
```

El resultado sería:

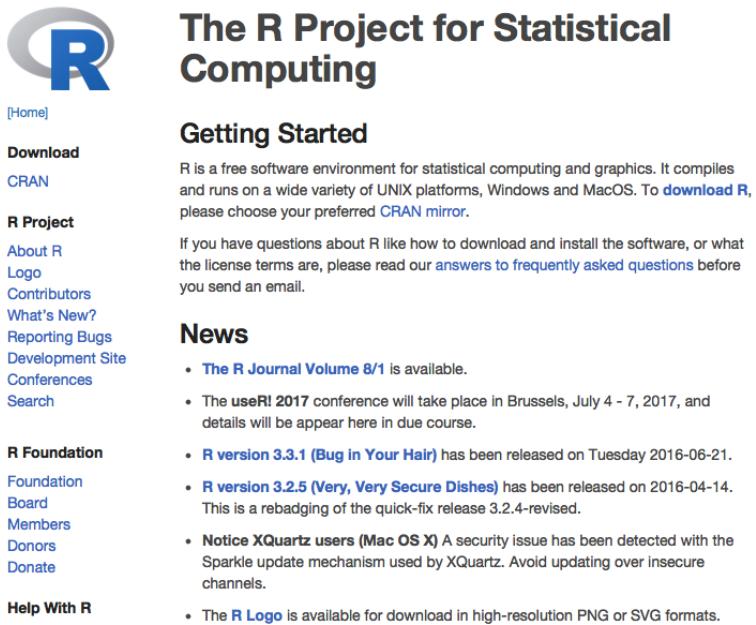


Figura 3.9: Página oficial de R.

### 3.2.2.1. Incluir leyendas que describan un gráfico y cómo referenciarlos

Para incluir una leyenda que describa el gráfico (generado por R o fichero gráfico) que se está mostrando, debe utilizarse la opción del chunk de código R: **fig.cap**, asociándole un texto entre dobles comillas. En el ejemplo anterior

```
```{r figura01, fig.cap = "\\label{figura01}Página oficial de R.",
fig.align='center',out.width='80%'}
knitr::include_graphics("figuras/oficialR.png")
```
```

Para hacer referencia a una figura, se debe añadir dentro del texto de la leyenda del gráfico: `\label{etiqueta}`. En el ejemplo anterior, se ha añadido `\label{figura01}`.

Hecho lo anterior, en cualquier parte del documento, se podrá hacer una referencia a ese gráfico al escribir: `ver la figura \ref{figura01}`, lo que produciría: “ver la figura 3.9”.

### 3.2.3. Cómo incluir información en forma de tablas en R Markdown

Para que los resultados se muestren con un aspecto más profesional en formato pdf, se suelen utilizar algunos paquetes R que tienen la facilidad de crear código LaTeX (o código html):

- Paquete **knitr**: este paquete dispone de la función “`kable()`” que facilita la creación de tablas de información.
- **kableExtra**<sup>5</sup>. Información más detallada para obtener salidas en pdf con `kableExtra`<sup>6</sup>.
- Otros paquetes que también crean tablas LaTeX desde R Markdown son: **xtable**, **huxtable**, **ztable**, etc.

Con el siguiente código se produce la tabla 3.2:

```
knitr:::kable(
 head(mtcars[, 1:8], 10), booktabs = TRUE,
 caption = "\\label{tabla01}Una tabla de las primeras 10 filas
 del dataset mtcars con knitr:::kable."
)
```

Cuadro 3.2: Una tabla de las primeras 10 filas del dataset mtcars con `knitr:::kable`.

|                   | mpg  | cyl | disp  | hp  | drat | wt    | qsec  | vs |
|-------------------|------|-----|-------|-----|------|-------|-------|----|
| Mazda RX4         | 21.0 | 6   | 160.0 | 110 | 3.90 | 2.620 | 16.46 | 0  |
| Mazda RX4 Wag     | 21.0 | 6   | 160.0 | 110 | 3.90 | 2.875 | 17.02 | 0  |
| Datsun 710        | 22.8 | 4   | 108.0 | 93  | 3.85 | 2.320 | 18.61 | 1  |
| Hornet 4 Drive    | 21.4 | 6   | 258.0 | 110 | 3.08 | 3.215 | 19.44 | 1  |
| Hornet Sportabout | 18.7 | 8   | 360.0 | 175 | 3.15 | 3.440 | 17.02 | 0  |
| Valiant           | 18.1 | 6   | 225.0 | 105 | 2.76 | 3.460 | 20.22 | 1  |
| Duster 360        | 14.3 | 8   | 360.0 | 245 | 3.21 | 3.570 | 15.84 | 0  |
| Merc 240D         | 24.4 | 4   | 146.7 | 62  | 3.69 | 3.190 | 20.00 | 1  |
| Merc 230          | 22.8 | 4   | 140.8 | 95  | 3.92 | 3.150 | 22.90 | 1  |
| Merc 280          | 19.2 | 6   | 167.6 | 123 | 3.92 | 3.440 | 18.30 | 1  |

#### 3.2.3.1. Incluir leyendas que describan una tabla de información y cómo referenciarlas

En el caso de tablas de información, la leyenda que describa el contenido de la tabla de información debe ir en al argumento “**caption**” de la función R: `kable`, como se muestra en el siguiente código:

<sup>5</sup><http://haozhu233.github.io/kableExtra/>

<sup>6</sup>[http://haozhu233.github.io/kableExtra/awesome\\_table\\_in\\_pdf.pdf](http://haozhu233.github.io/kableExtra/awesome_table_in_pdf.pdf)

```
knitr::kable(
 head(mtcars[, 1:8], 10), booktabs = TRUE,
 caption = "\\label{tabla01}Una tabla de las primeras 10 filas
 del dataset mtcars con knitr::kable."
)
```

La etiqueta se añade como `\label{etiqueta}` dentro del texto del argumento `caption`. En el ejemplo anterior ha sido: `\label{tabla01}`.

Hecho lo anterior, al igual que con los gráficos, en cualquier parte del documento, se podrá hacer una referencia a esa tabla de información al escribir: `ver la tabla \ref{tabla01}`, lo que produciría: “ver la tabla [3.2](#)”.

### 3.2.3.2. Tablas con kableExtra

El paquete “kableExtra” permite que la función `kable()` del paquete “knitr” tenga más versatilidad y opciones de uso.

Para usarlo, se tendrá que cargar el paquete “kableExtra” después del paquete “knitr”:

```
library(knitr)
#options(knitr.table.format = "latex",
kableExtra.latex.load_package = FALSE)
library(kableExtra)
dt es un data.frame que se usará para los ejemplos
dt <- mtcars[1:5, 1:6]
```

En este documento se está utilizando la salida “latex” que construye el fichero “pdf”, pero también podría usarse en documentos “html”, e incluso documentos “word” con el truco de crear un fichero gráfico de la tabla que se incluirá en el documento word (ver la documentación de kableExtra).

- **Tabla escalada con filas alternas coloreadas:**

El siguiente código produce la siguiente tabla (se recuerda que usando `echo=FALSE` se conseguiría que se mostrara únicamente la tabla y no el código):

```
kable(cbind(dt, dt, dt), format = "latex", booktabs = T) %>%
 kable_styling(latex_options = c("striped", "scale_down"))
```

|                   | mpg  | cyl | disp | hp  | drat | wt    | mpg  | cyl | disp | hp  | drat | wt    | mpg  | cyl | disp | hp  | drat | wt    |
|-------------------|------|-----|------|-----|------|-------|------|-----|------|-----|------|-------|------|-----|------|-----|------|-------|
| Mazda RX4         | 21.0 | 6   | 160  | 110 | 3.90 | 2.620 | 21.0 | 6   | 160  | 110 | 3.90 | 2.620 | 21.0 | 6   | 160  | 110 | 3.90 | 2.620 |
| Mazda RX4 Wag     | 21.0 | 6   | 160  | 110 | 3.90 | 2.875 | 21.0 | 6   | 160  | 110 | 3.90 | 2.875 | 21.0 | 6   | 160  | 110 | 3.90 | 2.875 |
| Datsun 710        | 22.8 | 4   | 108  | 93  | 3.85 | 2.320 | 22.8 | 4   | 108  | 93  | 3.85 | 2.320 | 22.8 | 4   | 108  | 93  | 3.85 | 2.320 |
| Hornet 4 Drive    | 21.4 | 6   | 258  | 110 | 3.08 | 3.215 | 21.4 | 6   | 258  | 110 | 3.08 | 3.215 | 21.4 | 6   | 258  | 110 | 3.08 | 3.215 |
| Hornet Sportabout | 18.7 | 8   | 360  | 175 | 3.15 | 3.440 | 18.7 | 8   | 360  | 175 | 3.15 | 3.440 | 18.7 | 8   | 360  | 175 | 3.15 | 3.440 |

### ■ Tabla flotante con el texto

El siguiente código con kable-kableExtra produce la siguiente salida:

```
kable(dt, format = "latex", booktabs = T) %>%
 kable_styling(position = "float_right")
```

  Lorem ipsum dolor sit amet,  
 consectetur adipiscing elit. Ut  
 purus elit, vestibulum ut, pla-  
 cerat ac, adipiscing vitae, fe-  
 lis. Curabitur dictum gravi-  
 da mauris. Nam arcu libero,  
 nonummy eget, consectetur  
 id, vulputate a, magna. Donec  
 vehicula augue eu neque. Pe-  
 llentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.  
 Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla  
 ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis  
 in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean  
 faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor  
 semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend,  
 sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

|                   | mpg  | cyl | disp | hp  | drat | wt    |
|-------------------|------|-----|------|-----|------|-------|
| Mazda RX4         | 21.0 | 6   | 160  | 110 | 3.90 | 2.620 |
| Mazda RX4 Wag     | 21.0 | 6   | 160  | 110 | 3.90 | 2.875 |
| Datsun 710        | 22.8 | 4   | 108  | 93  | 3.85 | 2.320 |
| Hornet 4 Drive    | 21.4 | 6   | 258  | 110 | 3.08 | 3.215 |
| Hornet Sportabout | 18.7 | 8   | 360  | 175 | 3.15 | 3.440 |

### ■ Tabla con tamaño de fuente menor

El siguiente código con kable-kableExtra produce la siguiente tabla:

```
kable(dt, format = "latex", booktabs = T) %>%
 kable_styling(font_size = 7)
```

|                   | mpg  | cyl | disp | hp  | drat | wt    |
|-------------------|------|-----|------|-----|------|-------|
| Mazda RX4         | 21.0 | 6   | 160  | 110 | 3.90 | 2.620 |
| Mazda RX4 Wag     | 21.0 | 6   | 160  | 110 | 3.90 | 2.875 |
| Datsun 710        | 22.8 | 4   | 108  | 93  | 3.85 | 2.320 |
| Hornet 4 Drive    | 21.4 | 6   | 258  | 110 | 3.08 | 3.215 |
| Hornet Sportabout | 18.7 | 8   | 360  | 175 | 3.15 | 3.440 |

- **Tablas con cabeceras agrupadas:**

El siguiente código con kable-kableExtra produce la siguiente tabla:

```
kable(dt, format = "latex", booktabs = T) %>%
 kable_styling(font_size = 11) %>%
 add_header_above(c(" ", "Group 4" = 4, "Group 5" = 2)) %>%
 add_header_above(c(" ", "Group 6" = 6), bold = T, italic = T)
```

| <i>Group 6</i>    |         |     |      |         |      |       |
|-------------------|---------|-----|------|---------|------|-------|
|                   | Group 4 |     |      | Group 5 |      |       |
|                   | mpg     | cyl | disp | hp      | drat | wt    |
| Mazda RX4         | 21.0    | 6   | 160  | 110     | 3.90 | 2.620 |
| Mazda RX4 Wag     | 21.0    | 6   | 160  | 110     | 3.90 | 2.875 |
| Datsun 710        | 22.8    | 4   | 108  | 93      | 3.85 | 2.320 |
| Hornet 4 Drive    | 21.4    | 6   | 258  | 110     | 3.08 | 3.215 |
| Hornet Sportabout | 18.7    | 8   | 360  | 175     | 3.15 | 3.440 |

- **Tablas con filas agrupadas:**

El siguiente código con kable-kableExtra produce como salida la tabla 3.3.

```
kable(mtcars[1:10, 1:6], format = "latex",
 caption = "\\label{tab:agrupadas}Filas Agrupadas", booktabs = T) %>%
 kable_styling()
 group_rows("Group 1", 4, 7) %>%
 group_rows("Group 2", 8, 10)
```

Cuadro 3.3: Filas Agrupadas

|                   | mpg  | cyl | disp  | hp  | drat | wt    |
|-------------------|------|-----|-------|-----|------|-------|
| Mazda RX4         | 21.0 | 6   | 160.0 | 110 | 3.90 | 2.620 |
| Mazda RX4 Wag     | 21.0 | 6   | 160.0 | 110 | 3.90 | 2.875 |
| Datsun 710        | 22.8 | 4   | 108.0 | 93  | 3.85 | 2.320 |
| <b>Group 1</b>    |      |     |       |     |      |       |
| Hornet 4 Drive    | 21.4 | 6   | 258.0 | 110 | 3.08 | 3.215 |
| Hornet Sportabout | 18.7 | 8   | 360.0 | 175 | 3.15 | 3.440 |
| Valiant           | 18.1 | 6   | 225.0 | 105 | 2.76 | 3.460 |
| Duster 360        | 14.3 | 8   | 360.0 | 245 | 3.21 | 3.570 |
| <b>Group 2</b>    |      |     |       |     |      |       |
| Merc 240D         | 24.4 | 4   | 146.7 | 62  | 3.69 | 3.190 |
| Merc 230          | 22.8 | 4   | 140.8 | 95  | 3.92 | 3.150 |
| Merc 280          | 19.2 | 6   | 167.6 | 123 | 3.92 | 3.440 |

## ■ Tablas con colores

El siguiente código con kable-kableExtra (al menos la versión 0.6 de kableExtra) produce la siguiente tabla:

```
iris[1:10,] %>%
 mutate_if(is.numeric, function(x) {
 cell_spec(x, "latex", bold = T,
 color = spec_color(x, end = 0.9),
 font_size = spec_font_size(x))
 }) %>%
 mutate(Species = cell_spec(Species, "latex",
 color = "white", bold = T,
 background = spec_color(1:10, end = 0.9,
 option = "A", direction = -1))) %>%
 kable("latex", escape = F, booktabs = T,
 linesep = "", align = "c")
```

| Sepal.Length | Sepal.Width | Petal.Length | Petal.Width | Species |
|--------------|-------------|--------------|-------------|---------|
| 5.1          | 3.5         | 1.4          | 0.2         | setosa  |
| 4.9          | 3           | 1.4          | 0.2         | setosa  |
| 4.7          | 3.2         | 1.3          | 0.2         | setosa  |
| 4.6          | 3.1         | 1.5          | 0.2         | setosa  |
| 5            | 3.6         | 1.4          | 0.2         | setosa  |
| 5.4          | 3.9         | 1.7          | 0.4         | setosa  |
| 4.6          | 3.4         | 1.4          | 0.3         | setosa  |
| 5            | 3.4         | 1.5          | 0.2         | setosa  |
| 4.4          | 2.9         | 1.4          | 0.2         | setosa  |
| 4.9          | 3.1         | 1.5          | 0.1         | setosa  |

### 3.2.3.3. Tablas de más de una página con kableExtra

Cuando la tabla es muy grande y no cabe en una única página puede utilizarse código con kable-kableExtra como el siguiente:

```
knitr::kable(
 iris[1:50,], longtable = TRUE, booktabs = TRUE,
 caption = 'Una tabla generada con knitr::kable,
 pero con el paquete LaTeX: longtable.'
) %>%
 kable_styling(latex_options = c("repeat_header"),
 repeat_header_text = "(continúa)")
```

Cuadro 3.4: Una tabla generada con knitr::kable, pero con el paquete LaTeX: longtable.

| Sepal.Length | Sepal.Width | Petal.Length | Petal.Width | Species |
|--------------|-------------|--------------|-------------|---------|
| 5.1          | 3.5         | 1.4          | 0.2         | setosa  |

Cuadro 3.4: Una tabla generada con knitr::kable (continúa)

| Sepal.Length | Sepal.Width | Petal.Length | Petal.Width | Species |
|--------------|-------------|--------------|-------------|---------|
| 4.9          | 3.0         | 1.4          | 0.2         | setosa  |
| 4.7          | 3.2         | 1.3          | 0.2         | setosa  |
| 4.6          | 3.1         | 1.5          | 0.2         | setosa  |
| 5.0          | 3.6         | 1.4          | 0.2         | setosa  |
| 5.4          | 3.9         | 1.7          | 0.4         | setosa  |
| 4.6          | 3.4         | 1.4          | 0.3         | setosa  |
| 5.0          | 3.4         | 1.5          | 0.2         | setosa  |
| 4.4          | 2.9         | 1.4          | 0.2         | setosa  |
| 4.9          | 3.1         | 1.5          | 0.1         | setosa  |
| 5.4          | 3.7         | 1.5          | 0.2         | setosa  |
| 4.8          | 3.4         | 1.6          | 0.2         | setosa  |
| 4.8          | 3.0         | 1.4          | 0.1         | setosa  |
| 4.3          | 3.0         | 1.1          | 0.1         | setosa  |
| 5.8          | 4.0         | 1.2          | 0.2         | setosa  |
| 5.7          | 4.4         | 1.5          | 0.4         | setosa  |
| 5.4          | 3.9         | 1.3          | 0.4         | setosa  |
| 5.1          | 3.5         | 1.4          | 0.3         | setosa  |
| 5.7          | 3.8         | 1.7          | 0.3         | setosa  |
| 5.1          | 3.8         | 1.5          | 0.3         | setosa  |
| 5.4          | 3.4         | 1.7          | 0.2         | setosa  |
| 5.1          | 3.7         | 1.5          | 0.4         | setosa  |
| 4.6          | 3.6         | 1.0          | 0.2         | setosa  |
| 5.1          | 3.3         | 1.7          | 0.5         | setosa  |
| 4.8          | 3.4         | 1.9          | 0.2         | setosa  |
| 5.0          | 3.0         | 1.6          | 0.2         | setosa  |
| 5.0          | 3.4         | 1.6          | 0.4         | setosa  |
| 5.2          | 3.5         | 1.5          | 0.2         | setosa  |
| 5.2          | 3.4         | 1.4          | 0.2         | setosa  |
| 4.7          | 3.2         | 1.6          | 0.2         | setosa  |
| 4.8          | 3.1         | 1.6          | 0.2         | setosa  |
| 5.4          | 3.4         | 1.5          | 0.4         | setosa  |
| 5.2          | 4.1         | 1.5          | 0.1         | setosa  |
| 5.5          | 4.2         | 1.4          | 0.2         | setosa  |
| 4.9          | 3.1         | 1.5          | 0.2         | setosa  |
| 5.0          | 3.2         | 1.2          | 0.2         | setosa  |
| 5.5          | 3.5         | 1.3          | 0.2         | setosa  |
| 4.9          | 3.6         | 1.4          | 0.1         | setosa  |
| 4.4          | 3.0         | 1.3          | 0.2         | setosa  |
| 5.1          | 3.4         | 1.5          | 0.2         | setosa  |
| 5.0          | 3.5         | 1.3          | 0.3         | setosa  |
| 4.5          | 2.3         | 1.3          | 0.3         | setosa  |

Cuadro 3.4: Una tabla generada con knitr::kabl (continúa)

| Sepal.Length | Sepal.Width | Petal.Length | Petal.Width | Species |
|--------------|-------------|--------------|-------------|---------|
| 4.4          | 3.2         | 1.3          | 0.2         | setosa  |
| 5.0          | 3.5         | 1.6          | 0.6         | setosa  |
| 5.1          | 3.8         | 1.9          | 0.4         | setosa  |
| 4.8          | 3.0         | 1.4          | 0.3         | setosa  |
| 5.1          | 3.8         | 1.6          | 0.2         | setosa  |
| 4.6          | 3.2         | 1.4          | 0.2         | setosa  |
| 5.3          | 3.7         | 1.5          | 0.2         | setosa  |
| 5.0          | 3.3         | 1.4          | 0.2         | setosa  |

Para hacer referencia a la tabla primera (se ha añadido `\label{tabla01}` en la función `kable()`), se puede escribir: ver la tabla `\ref{tabla01}` y quedaría como “ver la tabla 3.2”.

**Nota:** en el siguiente enlace en stackoverflow<sup>7</sup> se aclara como usarlo en otras situaciones.

### 3.2.3.4. Tablas en modo apaisado con kableExtra

Cuando una tabla con información necesita un ancho mayor que el ancho de un a4, puede indicarse que esa tabla va a imprimirse ocupando el espacio del largo del a4 (modo apaisado). Con el siguiente código kable-kableExtra se puede crear una tabla con información que se colocará en modo apaisado:

```
library(knitr)
library(kableExtra)
kable(cbind(rbind(dt,dt,dt),rbind(dt,dt,dt)), format = "latex",
 caption = "Tabla Demo en modo (Landscape) [note]", booktabs = T) %>%
 kable_styling(latex_options = c("hold_position")) %>%
 add_header_above(c(" ", "Group 1[note]" = 3, "Group 2[note]" = 3)) %>%
 add_footnote(c("This table is from mtcars",
 "Group 1 contains mpg, cyl and disp",
 "Group 2 contains hp, drat and wt"),
 notation = "symbol") %>%
 group_rows("Group 1", 4, 5) %>% landscape()
```

<sup>7</sup><https://stackoverflow.com/questions/38861041/knitr-rmarkdown-latex-how-to-cross-reference-figures-and-tables>

Cuadro 3.5: Tabla Demo en modo (Landscape)<sup>\*</sup>

|                    | Group 1 <sup>†</sup> |     |      | Group 2 <sup>‡</sup> |      |       | mpg  | cyl | disp | hp  | drat | wt    |
|--------------------|----------------------|-----|------|----------------------|------|-------|------|-----|------|-----|------|-------|
|                    | mpg                  | cyl | disp | hp                   | drat | wt    |      |     |      |     |      |       |
| Mazda RX4          | 21.0                 | 6   | 160  | 110                  | 3.90 | 2.620 | 21.0 | 6   | 160  | 110 | 3.90 | 2.620 |
| Mazda RX4 Wag      | 21.0                 | 6   | 160  | 110                  | 3.90 | 2.875 | 21.0 | 6   | 160  | 110 | 3.90 | 2.875 |
| Datsun 710         | 22.8                 | 4   | 108  | 93                   | 3.85 | 2.320 | 22.8 | 4   | 108  | 93  | 3.85 | 2.320 |
| <b>Group 1</b>     |                      |     |      |                      |      |       |      |     |      |     |      |       |
| Hornet 4 Drive     | 21.4                 | 6   | 258  | 110                  | 3.08 | 3.215 | 21.4 | 6   | 258  | 110 | 3.08 | 3.215 |
| Hornet Sportabout  | 18.7                 | 8   | 360  | 175                  | 3.15 | 3.440 | 18.7 | 8   | 360  | 175 | 3.15 | 3.440 |
| Mazda RX41         | 21.0                 | 6   | 160  | 110                  | 3.90 | 2.620 | 21.0 | 6   | 160  | 110 | 3.90 | 2.620 |
| Mazda RX4 Wag1     | 21.0                 | 6   | 160  | 110                  | 3.90 | 2.875 | 21.0 | 6   | 160  | 110 | 3.90 | 2.875 |
| Datsun 7101        | 22.8                 | 4   | 108  | 93                   | 3.85 | 2.320 | 22.8 | 4   | 108  | 93  | 3.85 | 2.320 |
| Hornet 4 Drive1    | 21.4                 | 6   | 258  | 110                  | 3.08 | 3.215 | 21.4 | 6   | 258  | 110 | 3.08 | 3.215 |
| Hornet Sportabout1 | 18.7                 | 8   | 360  | 175                  | 3.15 | 3.440 | 18.7 | 8   | 360  | 175 | 3.15 | 3.440 |
| Mazda RX42         | 21.0                 | 6   | 160  | 110                  | 3.90 | 2.620 | 21.0 | 6   | 160  | 110 | 3.90 | 2.620 |
| Mazda RX4 Wag2     | 21.0                 | 6   | 160  | 110                  | 3.90 | 2.875 | 21.0 | 6   | 160  | 110 | 3.90 | 2.875 |
| Datsun 7102        | 22.8                 | 4   | 108  | 93                   | 3.85 | 2.320 | 22.8 | 4   | 108  | 93  | 3.85 | 2.320 |
| Hornet 4 Drive2    | 21.4                 | 6   | 258  | 110                  | 3.08 | 3.215 | 21.4 | 6   | 258  | 110 | 3.08 | 3.215 |
| Hornet Sportabout2 | 18.7                 | 8   | 360  | 175                  | 3.15 | 3.440 | 18.7 | 8   | 360  | 175 | 3.15 | 3.440 |

<sup>\*</sup> This table is from mtcars<sup>†</sup> Group 1 contains mpg, cyl and disp<sup>‡</sup> Group 2 contains hp, drat and wt

### 3.2.4. Referencias a fórmulas matemáticas LaTeX en R Markdown

Como ya se ha dicho anteriormente, se pueden escribir ecuaciones dentro del texto escribiendo entre una pareja de símbolos del dólar \$ usando la sintaxis LaTeX, por ejemplo,  $\$f(k) = {n \choose k} p^k (1-p)^{n-k}\$$  (la salida será:  $f(k) = \binom{n}{k} p^k (1-p)^{n-k}$ ).

Las expresiones matemáticas que quieran presentarse centradas en un nueva línea pueden ser escritas utilizando un par de dobles símbolos del dólar \$\$, por ejemplo,

```
$$ f(k) = {n \choose k} p^k (1-p)^{n-k} $$
```

producirá la siguiente salida:

$$f(k) = \binom{n}{k} p^k (1-p)^{n-k}$$

Pero para numerar y hacer referencias a ecuaciones, deben ponerse en los entornos de LaTeX “equation” y utilizar etiquetas, por ejemplo:

```
\begin{equation}
f(k) = {n \choose k} p^k (1-p)^{n-k}
\label{eq:binom}
\end{equation}
```

La ecuación se mostraría del siguiente modo:

$$f(k) = \binom{n}{k} p^k (1-p)^{n-k} \tag{3.1}$$

Para referirse a ella en cualquier parte del documento se escribiría: `ecuación \ref{eq:binom}`, por ejemplo, esto se vería en el texto como: “ecuación 3.1”.

**Nota:** las referencias a ecuaciones funcionan correctamente en salidas LaTeX/PDF, y trabaja parcialmente en salidas HTML en el sentido que no pueden hacerse referencias a ecuaciones en otras páginas. Actualmente esta característica no está disponible en salidas word o epub.

Consultar los siguientes enlaces para obtener más información sobre cómo escribir en modo matemático en R Markdown usando el lenguaje LaTeX:

- Uso de LaTeX en modo matemático (wikipedia)<sup>8</sup>.
- Uso avanzado de matemáticas en LaTeX (wikibooks)<sup>9</sup>.
- Alineando ecuaciones en LaTeX (sharelatex)<sup>10</sup>.
- Teoremas y demostraciones en LaTeX (sharelatex)<sup>11</sup>.

<sup>8</sup>[https://es.m.wikipedia.org/wiki/Ayuda:Uso\\_de\\_TeX](https://es.m.wikipedia.org/wiki/Ayuda:Uso_de_TeX)

<sup>9</sup>[https://en.m.wikibooks.org/wiki/LaTeX/Advanced\\_Mathematics](https://en.m.wikibooks.org/wiki/LaTeX/Advanced_Mathematics)

<sup>10</sup>[https://es.sharelatex.com/learn/Aligning\\_equations\\_with\\_amsmath](https://es.sharelatex.com/learn/Aligning_equations_with_amsmath)

<sup>11</sup>[https://es.sharelatex.com/learn/Theorems\\_and\\_proofs](https://es.sharelatex.com/learn/Theorems_and_proofs)

### 3.2.5. Cómo hacer referencias a secciones en R Markdown

En Markdown, los capítulos, las secciones y subsecciones se crean utilizando los símbolos: #, ##, ###. Para poder hacer referencias a ellas se necesita etiquetarlas. Se puede hacer

- **de forma explícita**, añadiendo detrás del nombre de la sección, entre llaves, un símbolo # seguido de una etiqueta, como por ejemplo: {#etiqueta}.
- **de forma automática**. Si no se ha hecho de forma explícita, R Markdown asocia una etiqueta a cada sección-subsección, construida a partir del título. Con este método es más probable el error.

Una vez etiquetadas las secciones-subsecciones, es posible hacer referencias a ellas de varias formas:

- Utilizando el comando LaTeX \ref{}. Por ejemplo: Ver la sección \ref{formulas} produce: Ver la sección [3.2.4](#).
- Escribir el título de la cabecera entre corchetes.
  - [Referencias a fórmulas matemáticas LaTeX en R Markdown] produce: [Referencias a fórmulas matemáticas LaTeX en R Markdown](#)
  - [LaTeX] [Referencias a fórmulas matemáticas LaTeX en R Markdown] produce: [LaTeX](#)

Cuando una etiqueta no puede encontrarse aparecerá “??”, además aparecerá un aviso en la consola de R al construir el documento.

### 3.2.6. Incluir citas bibliográficas en R Markdown

Se recomienda usar ficheros BibTeX para almacenar las referencias a todos los trabajos que queramos citar. El sistema LaTeX usa los ficheros BibTeX. Usarlo, permite seleccionar de una infinidad de estilos bibliográficos de revistas científicas existentes.

Un fichero BibTeX es un fichero de texto plano, con la extensión “.bib”, que consta de entradas bibliográficas como las siguientes:

```
@Manual{Luque2017,
 title = {Escribir un Trabajo Fin de Estudios con R Markdown},
 author = {Pedro L. Luque-Calvo},
 year = "2017",
 howpublished = {Disponible en \url{http://destio.us.es/calvo}}
}

@Manual{R-base,
 title = {R: A Language and Environment for Statistical
 Computing},
 author = {{R Core Team}},
 organization = {R Foundation for Statistical Computing},
 address = {Vienna, Austria},
 year = {2016},
 url = {https://www.R-project.org/},
}
```

```

@book{Wickham2017,
 Author = {Hadley Wickham and Garrett Grolemund},
 Edition = {First Edition},
 Publisher = {O'Reilly},
 Title = {R for Data Science: Import, Tidy, Transform, Visualize, and Model Data},
 Year = {2017}
}

@ARTICLE {Example1999,
 author = "Example, The",
 title = "This is an exmaple article not cited in the text",
 journal = "Journal of Examples",
 year = "1999",
 volume = "4",
 number = "2",
 pages = "1-9"
}

@Misc{Techopedia,
 title = {"Definition - What does Business Intelligence (BI) mean?"},
 author = {Techopedia},
 howpublished = {Disponible en \url{https://www.techopedia.com/definition/345/business-intelligence-bi}}
}

```

Una entrada bibliográfica comienza con `@type{`, donde “type” puede ser: “article” (para artículos científicos), “book” (para libros), “Manual” (para manuales), “Misc” (para otros documentos), etc. A continuación, sigue un identificador o clave único para la entrada bibliográfica, en el ejemplo anterior, la clave de la primera entrada es: `@R-base`.

Para citar a esta entrada bibliográfica en el texto, se usa la marca: `@identificador`, o `[@identificador]`. Por ejemplo,

- `@R-base` produce la salida: [8], y
- `[@R-base]` produce la salida: [8].

**Nota:** cuando el listado bibliográfico es numerado, no hay distinción entre las dos marcas, cuando no es numerado, la primera marca no pone paréntesis y la segunda marca escribe la referencia entre paréntesis, como se ve en la siguiente captura:

Para citar a esta entrada bibliográfica en el texto, se usa la marca: `@identificador`, o `[@identificador]`. Por ejemplo,

- `@R-base` produce la salida: R Core Team (2016), y
- `[@R-base]` produce la salida: (R Core Team 2016).

Existen diferentes campos en una entrada bibliográfica, tales como: “title”, “author”, “year”, etc. Puede consultarse la siguiente url <https://en.wikipedia.org/wiki/BibTeX> para ver todos los posibles campos en BibTeX (algunos son obligatorios incluirlos).

Hay una función R en el paquete “knitr”: `write_bib()` para generar entradas BibTeX automáticamente para paquetes R.

Por ejemplo:

```
```{r}
knitr:::write_bib(c("knitr","stringr"),file = "bib/paquetes.bib",width = 60)
```
```

Produce las entradas bibliográficas de los paquetes R: “knitr” y “stringr”:

```
@Manual{R-knitr,
title = {knitr: A General-Purpose Package for Dynamic Report
Generation in R},
author = {Yihui Xie},
year = {2017},
note = {R package version 1.17},
url = {https://CRAN.R-project.org/package=knitr},
}
@Manual{R-stringr,
title = {stringr: Simple, Consistent Wrappers for Common
String Operations},
author = {Hadley Wickham},
year = {2017},
note = {R package version 1.2.0},
url = {https://CRAN.R-project.org/package=stringr},
}
```

En la **cabecera YAML del documento principal R Markdown**, se usa el campo “bibliography” para incluir uno o más ficheros “.bib”. También puede especificarse un estilo biliográfico con el campo “csl”, como puede verse en el siguiente ejemplo:

```

bibliography: ["library.bib", "paquetes.bib"]
#csl: methods-in-ecology-and-evolution.csl
csl: acm-sig-proceedings-long-author-list.csl
#link-citations: true

```

Existen muchas posibilidades para la opción “csl”, y cada una define un modo particular para mostrar la bibliografía. Cada una de esas opciones está asociada a un fichero que debe encontrarse en la carpeta del fichero Rmd principal. En los siguientes enlaces puede encontrarse más información sobre ficheros “csl”:

- Ficheros de estilos csl en Github (más de 1000 ficheros)<sup>12</sup>
- <http://citationstyles.org>

---

<sup>12</sup><https://github.com/citation-style-language/styles>

Los dos ficheros que se han incluido en la plantilla “MemoriaTFE.zip” son:

- “**acm-sig-proceedings-long-author-list.csl**”: cada una de las referencias bibliográficas aparece numerada, pero el orden es por los apellidos de los autores y en caso de empate se ordena en orden descendente por año de publicación.

## Bibliografía

- [1] Allaire, J., Xie, Y., McPherson, J., Luraschi, J., Ushey, K., Atkins, A., Wickham, H., Cheng, J. and Chang, W. 2017. *Rmarkdown: Dynamic documents for r*.
- [2] Dahl, D.B. 2016. *Xtable: Export tables to latex or html*.
- [3] Example, T. 1999. This is an example article not cited in the text. *Journal of Examples*. 4, 2 (1999), 1–9.
- [4] Gravitar, I. sin límites 2013. *¿Qué es pentaho?* Disponible en <http://gravitar.biz/pentaho/>, version 1.6.0.
- [5] Luque-Calvo, P.L. 2017. *Escribir un trabajo fin de estudios con rmarkdown*. Disponible en <http://destio.us.es/calvo>.
- [6] R Core Team 2016. *R: A language and environment for statistical computing*. R Foundation for Statistical Computing.
- [7] Techopedia “Definition - what does business intelligence (bi) mean?” Disponible en <https://www.techopedia.com/definition/345/business-intelligence-bi>.
- [8] Wickham, H. 2017. *Stringr: Simple, consistent wrappers for common string operations*.
- [9] Wickham, H. and Chang, W. 2016. *Ggplot2: Create elegant data visualisations using the grammar of graphics*.
- [10] Wickham, H. and Grolemund, G. 2017. *R for data science: Import, tidy, transform, visualize, and model data*. O'Reilly.
- [11] Wickham, H., Francois, R., Henry, L. and Müller, K. 2017. *Dplyr: A grammar of data manipulation*.
- [12] Xie, Y. 2017. *Knitr: A general-purpose package for dynamic report generation in r*.

Figura 3.10: Presentación de la Bibliografía con el fichero: acm-sig-proceedings-long-author-list.csl

- “methods-in-ecology-and-evolution.csl”: Muestra por autor fecha.

## Bibliografía

- Allaire, J., Xie, Y., McPherson, J., Luraschi, J., Ushey, K., Atkins, A., Wickham, H., Cheng, J. & Chang, W. (2017). *Rmarkdown: Dynamic documents for r*.
- Dahl, D.B. (2016). *Xtable: Export tables to latex or html*.
- Example, T. (1999). This is an exmaple article not cited in the text. *Journal of Examples*, 4, 1–9.
- Gravitar, I. sin límites. (2013). *¿Qué es pentaho?* Disponible en <http://gravitar.biz/pentaho/>, version 1.6.0.
- Luque-Calvo, P.L. (2017). *Escribir un trabajo fin de estudios con rmarkdown*. Disponible en <http://destio.us.es/calvo>.
- R Core Team. (2016). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria.
- Techopedia. ‘Definition - what does business intelligence (bi) mean?’
- Wickham, H. (2017). *Stringr: Simple, consistent wrappers for common string operations*.
- Wickham, H. & Chang, W. (2016). *Ggplot2: Create elegant data visualisations using the grammar of graphics*.
- Wickham, H. & Gromelund, G. (2017). *R for data science: Import, tidy, transform, visualize, and model data*, First Editionn. O'Reilly.
- Wickham, H., Francois, R., Henry, L. & Müller, K. (2017). *Dplyr: A grammar of data manipulation*.
- Xie, Y. (2017). *Knitr: A general-purpose package for dynamic report generation in r*.

Figura 3.11: Presentación de la Bibliografía con el fichero: methods-in-ecology-and-evolution.csl

Todas las referencias bibliográficas que se citen en el documento serán incluidas en la sección de “Bibliografía” automáticamente. Pero si queremos obligar a que aparezcan referencias no citadas en el trabajo, se pueden incluir sus claves en la lista que aparece a continuación de la cabecera “yaml” del documento R Markdown principal, justo después del término: “nocite”, como se muestra a continuación:

```

nocite: |
 @Luque2017, @RStudio, @R-base,
 @R-knitr, @R-rmarkdown, @R-dplyr, @R-ggplot2, @R-xtable,
 @R-stringr, @Techopedia
...
```

Para más información sobre citación bibliográfica en R Markdown consultar la url:

- [http://rmarkdown.rstudio.com/authoring\\_bibliographies\\_and\\_citations.html](http://rmarkdown.rstudio.com/authoring_bibliographies_and_citations.html).

**Recomendación:** en el siguiente enlace se explica cómo usar Google Scholar<sup>13</sup> para crear-descargar entradas de citas bibliográficas (libros, artículos, ...) en estilo BibTeX, listas para incluir en los ficheros “.bib”:

<http://texblog.org/2014/04/22/using-google-scholar-to-download-bibtex-citations/>

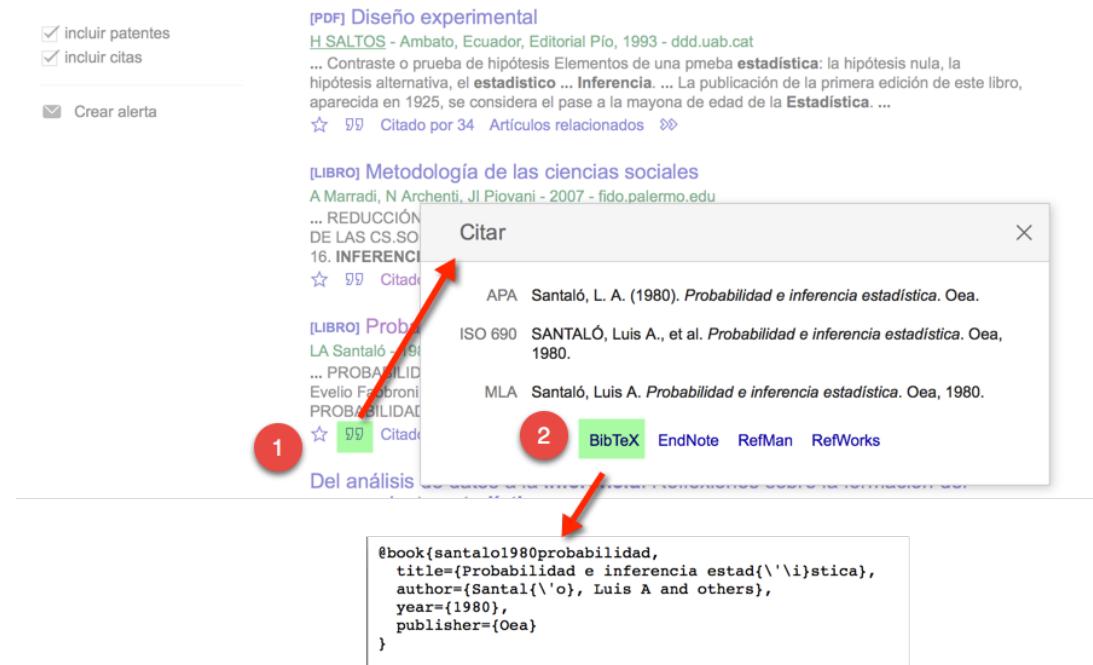


Figura 3.12: Esquema del uso de Google Scholar para encontrar las entradas bibliográficas en formato BibTeX.

### 3.2.7. Incluir páginas web, aplicaciones Shiny y Widgets en R Markdown

Para incluir una página web en el documento, siempre se podrá hacer una captura de pantalla y con el fichero gráfico, por ejemplo: “png”, incluirlo como se ha hecho en la sección 3.2.2.

Pero para incluir una página web, también puede usarse la función `include_url()` del paquete “knitr”:

```
```{r out.width='75%',fig.align='center'}
knitr::include_url("http://www.google.com")
```
```

produce la siguiente salida:

<sup>13</sup><https://scholar.google.es>



También se puede limitar la altura del “iframe” con el argumento `height = "600px"` en la función `knitr::include_url`.

Para incluir código R que produzca salidas “HTML5-Javascript” debemos tener instalados el paquete R “webshot”, e instalar desde R el programa “PhantomJS”. Puede hacerse del siguiente modo desde la consola de R:

```
install.packages("webshot")
webshot::install_phantomjs()
```

Una vez instalado, por ejemplo, el siguiente código R, capturaría una foto del aspecto de la tabla creada con el paquete DT (html5):

```
```{r out.width='95%',fig.align='center'}
DT::datatable(iris)
```
```

produciría la siguiente salida (captura en PDF):

| Search: <input type="text"/>                   |              |             |              |             |         |
|------------------------------------------------|--------------|-------------|--------------|-------------|---------|
| Show <input type="button" value="10"/> entries |              |             |              |             |         |
|                                                | Sepal.Length | Sepal.Width | Petal.Length | Petal.Width | Species |
| 1                                              | 5.1          | 3.5         | 1.4          | 0.2         | setosa  |
| 2                                              | 4.9          | 3           | 1.4          | 0.2         | setosa  |
| 3                                              | 4.7          | 3.2         | 1.3          | 0.2         | setosa  |
| 4                                              | 4.6          | 3.1         | 1.5          | 0.2         | setosa  |
| 5                                              | 5            | 3.6         | 1.4          | 0.2         | setosa  |
| 6                                              | 5.4          | 3.9         | 1.7          | 0.4         | setosa  |
| 7                                              | 4.6          | 3.4         | 1.4          | 0.3         | setosa  |
| 8                                              | 5            | 3.4         | 1.5          | 0.2         | setosa  |
| 9                                              | 4.4          | 2.9         | 1.4          | 0.2         | setosa  |
| 10                                             | 4.9          | 3.1         | 1.5          | 0.1         | setosa  |

Showing 1 to 10 of 150 entries      Previous 1 2 3 4 5 ... 15 Next

**Nota:** También se pueden incluir capturas de aplicaciones shiny con el comando `knitr::include_app`.

**Nota:** Existen muchos ejemplos de uso del paquete R knitr en: <https://github.com/yihui/knitr-examples>.

### 3.2.8. Usar código LaTeX dentro de R Markdown

Con LaTeX se pueden crear todo tipo de documentos. Para ilustrar esto pueden consultarse los siguientes enlaces

- Galería de documentos LaTeX (Overleaf): <https://www.overleaf.com/gallery/recent/page/4>

**Nota Importante:** hay que tener en cuenta que si se usan comandos LaTeX en un documento R Markdown, los resultados sí funcionarán en salidas “pdf”, pero por lo general **no** funcionarán con otras salidas: “html” o “word”.

Por ejemplo, el paquete de R “bookdown” [15] (muy recomendable): <http://www.bookdown.com>, sí permite crear documentos R Markdown que mantienen la posibilidad de obtener diferentes formatos de salida a partir del mismo fichero.

#### 3.2.8.1. Tamaños de fuentes con ayuda de LaTeX

Para especificar cambios de tamaños de fuente se pueden utilizar los siguientes comandos LATEX:

| Comando       | 10pt  | 11pt  | 12pt  |
|---------------|-------|-------|-------|
| \tiny         | 5     | 6     | 6     |
| \scriptsize   | 7     | 8     | 8     |
| \footnotesize | 8     | 9     | 10    |
| \small        | 9     | 10    | 10.95 |
| \normalsize   | 10    | 10.95 | 12    |
| \large        | 12    | 12    | 14.4  |
| \Large        | 14.4  | 14.4  | 17.28 |
| \LARGE        | 17.28 | 17.28 | 20.74 |
| \huge         | 20.74 | 20.74 | 24.88 |
| \Huge         | 24.88 | 24.88 | 24.88 |

**Nota:** se recomienda su uso únicamente cuando el texto que se muestra se sale del ancho del texto. Para ello, en primer lugar se tendrá que usar `\footnotesize`, por ejemplo, y para recuperar el tamaño normal usar el comando `\normalsize`.

### 3.2.8.2. Colores con ayuda de LaTeX

Se pueden colorear elementos de nuestro documento con los siguientes comandos LaTeX:

- `\textcolor{colorLaTeX}{texto}.`

El código:

```
\textcolor{red}{Resalto texto en color rojo,}
y el resto no.
```

```
\textcolor{RubineRed}{\rule{0.8\linewidth}{0.5mm}}
```

Produce:

**Resalto texto en color rojo**, y el resto no.

- `\colorbox{colorLaTeX}{texto}.`

El código:

```
Con este \colorbox{BurntOrange}{texto se ha resaltado}
y luego se puede continuar escribiendo normal.
```

Produce:

Con este **texto se ha resaltado** y luego se puede continuar escribiendo normal.

- **Definición de nuevos colores** con: `\definecolor` y `\colorlet`:

El código:

```
\colorbox{green}{\textcolor{orange}{Texto resaltado}}
en color "green" el fondo y el texto en color "orange".
```

```
\definecolor{mypink1}{rgb}{0.858, 0.188, 0.478}
\definecolor{mypink2}{RGB}{219, 48, 122}
\definecolor{mygray}{gray}{0.6}
\colorlet{LightRubineRed}{RubineRed!70!}
\colorlet{Mycolor1}{green!10!orange!90!}
\definecolor{Mycolor2}{HTML}{00F9DE}
```

```
\colorbox{mygray}{\textcolor{LightRubineRed}{Texto resaltado}}
en color "mygray" el fondo y el texto en color "LightRubineRed".
```

```
\colorbox{Mycolor1}{\textcolor{White}{Texto resaltado}}
en color "Mycolor1" el fondo y el texto en color "White".
```

```
\textcolor[rgb]{0.13,0.29,0.53}{Texto}
```

- Produce la siguiente salida:

**Texto resaltado** en color “green” el fondo y el texto en color “orange”.

- Se pueden mezclar los dos comandos:

**Texto resaltado** en color “mygray” el fondo y el texto en color “LightRubineRed”.

**Texto resaltado** en color “Mycolor1” el fondo y el texto en color “White”.

### Texto

- Se puede cambiar el color de una página con `\fcolorbox` y el entorno `minipage`:

El siguiente código:

```
\noindent\fcolorbox{red}{cyan}{
\begin{minipage}{0.30\textwidth}
\color{white}
Lo que se se ve en esta \textcolor{red}{minipágina}
tiene un color de fondo negro y
el texto está escrito en blanco.
\end{minipage}\hspace{0.5cm}
\begin{minipage}{0.60\textwidth}
\color{white}
Lo que se se ve en esta \textcolor{red}{minipágina}
tiene un color de fondo negro y
el texto está escrito en blanco.
\end{minipage}
```

Produce la siguiente salida:

Lo que se se ve en esta **mi-nipágina** tiene un color de fondo negro y el texto está escrito en blanco.

Lo que se se ve en esta **minipágina** tiene un color de fondo negro y el texto está escrito en blanco.

Otra variante en el siguiente código:

```
\noindent\colorbox{blue}{
\begin{minipage}{0.30\textwidth}
\color{white}
Lo que se se ve en esta \textcolor{red}{minipágina}
tiene un color de fondo negro y
el texto está escrito en blanco.
\end{minipage}}\hspace{0.5cm}
\colorbox{yellow}{\begin{minipage}{0.60\textwidth}
\color{white}
Lo que se se ve en esta \textcolor{red}{minipágina}
tiene un color de fondo negro y
el texto está escrito en blanco.
\end{minipage}}
```

Produce la siguiente salida:

Lo que se se ve en esta **mi-nipágina** tiene un color de fondo negro y el texto está escrito en blanco.

Lo que se se ve en esta **minipágina** tiene un color de fondo negro y el texto está escrito en blanco.

**Nota:** No se recomienda usar `\color{colorLaTeX} texto` ya que no funciona correctamente en R Markdown.

En la siguiente imagen (figura 3.13) se recogen algunos de los colores más habituales en LaTeX y por tanto en R Markdown:

white, black, red, green, blue, cyan, magenta, yellow



#### Colour names available with the dvipsnames option

|                |              |             |                |
|----------------|--------------|-------------|----------------|
| Apricot        | Emerald      | OliveGreen  | RubineRed      |
| Aquamarine     | ForestGreen  | Orange      | Salmon         |
| Bittersweet    | Fuchsia      | OrangeRed   | SeaGreen       |
| Black          | Goldenrod    | Orchid      | Sepia          |
| Blue           | Gray         | Peach       | YellowOrange   |
| BlueGreen      | Green        | Periwinkle  | SkyBlue        |
| BlueViolet     | GreenYellow  | PineGreen   | SpringGreen    |
| BrickRed       | JungleGreen  | Plum        | Tan            |
| Brown          | Lavender     | ProcessBlue | TealBlue       |
| BurntOrange    | LimeGreen    | Purple      | Thistle        |
| CadetBlue      | Magenta      | RawSienna   | Turquoise      |
| CarnationPink  | Mahogany     | Red         | Violet         |
| Cerulean       | Maroon       | RedOrange   | VioletRed      |
| CornflowerBlue | Melon        | RedViolet   | White          |
| Cyan           | MidnightBlue | Rhodamine   | WildStrawberry |
| Dandelion      | Mulberry     | RoyalBlue   | Yellow         |
| DarkOrchid     | NavyBlue     | RoyalPurple | YellowGreen    |

Figura 3.13: Colores LaTeX en los paquetes color y xcolor, respectivamente

Más información en:

- Usando colores en LaTeX (sharelatex)<sup>14</sup>
- Cajas en LaTeX (wikibooks)<sup>15</sup>

<sup>14</sup>[https://es.sharelatex.com/learn/Using\\_colours\\_in\\_LaTeX](https://es.sharelatex.com/learn/Using_colours_in_LaTeX)

<sup>15</sup><https://en.m.wikibooks.org/wiki/LaTeX/Boxes>

### 3.2.8.3. Resaltar bloques de texto de un modo personalizado

A continuación, a través de código de ejemplo, pueden verse distintas formas de crear bloques de texto personalizados con ayuda de macros LaTeX.

```
\begin{mdframed}[style=exampledefault,frametitle={Inhomogeneous linear}]
Some Text
\end{mdframed}
```

Inhomogeneous linear

Some Text

```
\bedefinition
Se **define** Lo que se ve en esta \textcolor{red}{minipágina}
tiene un color de fondo negro y
el texto está escrito en blanco.
\eedefinition
```

**Definición 3.2.1** Se **define** .... Lo que se ve en esta **minipágina** tiene un color de fondo negro y el texto está escrito en blanco.

```
\betheorem
Dado ... entonces se verifica ... Lo que se ve en esta
\textcolor{red}{minipágina} tiene un color de fondo negro y
el texto está escrito en blanco.
\etheorem
```

**Teorema 3.2.1** Dado ... entonces se verifica ... Lo que se ve en esta **minipágina** tiene un color de fondo negro y el texto está escrito en blanco.

```
\beremark
\lipsum[1]
\eremark
```



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

```
\beexercise
\lipsum[1]
\eeexercise
```

**Ejercicio 3.1** Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

---

```
\besolutionExe
\lipsum[1]
\eesolutionExe
```

(S)

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

---

```
\beexample
\lipsum[1]
\eeexample
```

■ **Ejemplo 3.1** Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

---

```
\besolutionExa
\lipsum[1]
\eesolutionExa
```

**S**

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

---

```
\becorollary
\lipsum[1]
\ecorollary
```

**Corolario 3.2.2** Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

---

```
\begin{proof}
La demostración sería ...
\end{proof}
```

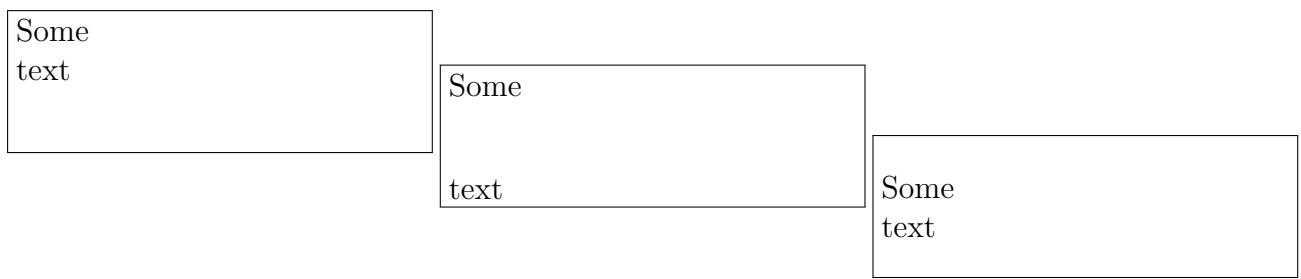
*Demostración.* La demostración sería ... ■

---

El siguiente código:

```
\noindent
\fbox{\parbox[b]{4em}[t]{0.33\textwidth}{Some \\ text} }
\fbox{\parbox[c]{4em}[s]{0.33\textwidth}{Some \vfill text} }
\fbox{\parbox[t]{4em}[c]{0.33\textwidth}{Some \\ text} }
```

Produce:



### 3.2.8.4. Uso del comando LaTeX: table

El siguiente código L<sup>A</sup>T<sub>E</sub>X en un documento R Markdown:

```
\begin{table}[H]
\centering
\caption{Tabla con información escrita con código LaTeX}
\label{table-paramvalues}
\begin{tabular}{ p{4cm} p{4cm} p{4cm} }
\hline \\ [1.5ex]
colname & colname & colname \\ [1ex]
\hline \\ [1.5ex]
Info & info & info \\ [1ex]
Info & info & info \\ [1ex]
Info & info & info \\ [1ex]
\hline
\end{tabular}
\end{table}
```

Produce la tabla 3.6:

Cuadro 3.6: Tabla con información escrita con código LaTeX

| colname | colname | colname |
|---------|---------|---------|
| Info    | info    | info    |
| Info    | info    | info    |
| Info    | info    | info    |

Algunas veces, es preferible usar tablas L<sup>A</sup>T<sub>E</sub>X en lugar de R Markdown, ya que se tienen más posibilidades de personalización.

### 3.2.8.5. Uso del comando LaTeX: figure

Puede incluirse el comando LaTeX: “figure” e incluir figuras (ver la figura 3.14).

```
\begin{figure}[H]
\centering
\caption{Figura con información escrita con código LaTeX}
\label{figurelabel}
\includegraphics[width=0.9\textwidth]{figuras/figure1}
\end{figure}
```

Produce el siguiente gráfico:

Figura 3.14: Figura con información escrita con código LaTeX



Se podría emplear también el siguiente código “markdown” para incluir gráficos (incluyendo etiquetas para referenciarlos), pero el inconveniente sería que no se puede controlar del todo la posición en la que iría el gráfico. Por ello algunas veces se utilizará la versión LaTeX.

```
![Esta es la leyenda de una figura para esta creación
verde. \label{figurelabel}](figuras/figure1.png)
```

### 3.2.8.6. Otros comandos LaTeX útiles

- El comando `\clearpage` (o `\newpage`) produce un salto de página en el lugar del documento donde se escriba.
- El comando `\cleardoublepage` produce un salto de página en el punto en el que se escriba pero obliga a que la próxima página sea impar.
- Si se quiere centrar el texto, colocarlo entre los comandos `\bcenter` y `\ecenter`, en líneas separadas.
- El comando `\pageref` inserta el número de página dónde se encuentra el objeto referenciado. Por ejemplo,  
la ecuación que aparece en la página `\pageref{eq:binom}`,  
produce la siguiente salida:  
la ecuación que aparece en la página 50.

- El paquete LaTeX “float” (cargado siempre en esta plantilla) permite obligar con la opción “H” que se pueda elegir que las figuras y/o las tablas se coloquen justo cuando se define la figura/tabla.

Colocando en el chunk de código R: `fig.pos="H"` se puede obligar a colocar la figura inmediatamente a continuación, pero no lo permite para las tablas.

Una solución, según se necesite delante de la figura o tabla, es usar el comando LaTeX:

```
* `\\floatplacement{figure}{H}` o
* `\\floatplacement{table}{H}` .
```

También se podrían usar con otras opciones: “b” = abajo, “t” = arriba, “h” = aquí, “p” = todos en una página de gráficos, “!” = obliga.

Como por ejemplo: `\\floatplacement{figure}{!bthp}`.

- Es posible incluir un documento pdf externo dentro de nuestro documento con ayuda del comando LaTeX `\\includepdf` del paquete: “pdffpages”<sup>16</sup>. En el siguiente ejemplo se insertan en nuestro documento las páginas 8 y 14 del fichero local “xaringanej01\_defecto.pdf” con un escalado de 75 % (de la 13 a 14, habría que escribir: `pages=13-14`):

```
\\includepdf[scale=0.75, pages={8,14},
pagecommand=\\section*{Ejemplo de
inserción de un pdf en un documento},
offset=00 -30]{xaringanej01_defecto.pdf}
```

---

<sup>16</sup>Más información en: <http://texdoc.net/texmf-dist/doc/latex/pdffpages/pdffpages.pdf>

## Ejemplo de inserción de un pdf en un documento

# Hello World

Install the **xaringan** package from [Github](#):

```
devtools::install_github("yihui/xaringan")
```

You are recommended to use the [RStudio IDE](#), but you do not have to.

- Create a new R Markdown document from the menu **File** → **New File** → **R Markdown** → **From Template** → **Ninja Presentation**<sup>1</sup>
- Click the **Knit** button to compile it;
- or use the [RStudio Addin](#)<sup>2</sup> "Infinite Moon Reader" to live preview the slides (every time you update and save the Rmd document, the slides will be automatically reloaded in RStudio Viewer).

[1] 中文用户请看[这份教程](#)

[2] See [#2](#) if you do not see the template or addin in RStudio.

## Ejemplo de inserción de un pdf en un documento

# xaringan

Provides an R Markdown output format `xaringan::moon_reader` as a wrapper for `remark.js`, and you can use it in the YAML metadata, e.g.

```

```

```
title: "A Cool Presentation"
output:
 xaringan::moon_reader
 yolo: true
 nature:
 autoplay: 30000

```

See the help page `?xaringan::moon_reader` for all possible options that you can use.

En el siguiente ejemplo, se usa la variante `\includepdfmerge` (se utiliza más de un fichero pdf), con el argumento “nup” para crear una matriz 2 por 6 para colocar los documentos:

```
```latex
\includepdfmerge[nup=2x6,scale=0.95]{
presentadotodo.pdf,3-4,8-9,15-16,26-27,
xaringanej01_defecto.pdf,15-18}
```
```

En las 12 páginas que se muestran en la rejilla de la página siguiente, se muestran ejemplos del aspecto de transparencias generadas con ayuda de R Markdown desde RStudio (ver sección ?? en página ??):

- Fila 1: presentación Beamer (pdf).
- Fila 2: presentación ioslides (html).
- Fila 3: presentación slidy (html).
- Fila 4: presentación revealjs (html).
- Fila 5 y 6: presentación xaringan (html).

El siguiente código muestra otro ejemplo de uso de `\includepdf` en el que se utiliza “nup” y también las opciones: “landscape” y “turn=false” (para que la página pdf se muestre en vertical), aunque no se muestra el resultado.

```
```latex
\includepdf[nup=2x2,landscape,turn=false,pages=15-18,
scale=0.80]{xaringanej01_defecto.pdf}
```
```

## Slide with Bullets

- ▶ Bullet 1
- ▶ Bullet 2
- ▶ Bullet 3

## Slide with R Output

```
summary(cars)

speed dist
Min. : 4.0 Min. : 2.00
1st Qu.:12.0 1st Qu.: 26.00
Median :15.0 Median : 36.00
Mean :15.4 Mean : 42.98
3rd Qu.:19.0 3rd Qu.: 56.00
Max. :25.0 Max. :120.00
```

## Slide with Bullets

- Bullet 1
- Bullet 2
- Bullet 3

## Slide with R Output

```
summary(cars)

speed dist
Min. : 4.0 Min. : 2.00
1st Qu.:12.0 1st Qu.: 26.00
Median :15.0 Median : 36.00
Mean :15.4 Mean : 42.98
3rd Qu.:19.0 3rd Qu.: 56.00
Max. :25.0 Max. :120.00
```

3/5

4/5

## Slide with Bullets

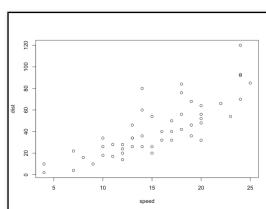
- Bullet 1
- Bullet 2
- Bullet 3

## Slide with R Output

```
summary(cars)
```

|    |              |                |
|----|--------------|----------------|
| ## | speed        | dist           |
| ## | Min.   : 4.0 | Min.   :  2.00 |
| ## | 1st Qu.:12.0 | 1st Qu.: 26.00 |
| ## | Median :15.0 | Median : 36.00 |
| ## | Mean   :15.4 | Mean   : 42.98 |
| ## | 3rd Qu.:19.0 | 3rd Qu.: 56.00 |
| ## | Max.   :25.0 | Max.   :120.00 |

## Slide with Plot



## Slide with R Code and Output

```
summary(cars)
```

|    |              |                |
|----|--------------|----------------|
| ## | speed        | dist           |
| ## | Min.   : 4.0 | Min.   :  2.00 |
| ## | 1st Qu.:12.0 | 1st Qu.: 26.00 |
| ## | Median :15.0 | Median : 36.00 |
| ## | Mean   :15.4 | Mean   : 42.98 |
| ## | 3rd Qu.:19.0 | 3rd Qu.: 56.00 |
| ## | Max.   :25.0 | Max.   :120.00 |

◀ ▶

## remark.js vs xaringan

Some differences between using remark.js (left) and using xaringan (right):

1. Start with a boilerplate HTML file;
2. Plain Markdown;
3. Write JavaScript to autoplay slides;
4. Manually configure MathJax;
5. Highlight code with `;
6. Edit Markdown source and refresh browser to see updated slides;

[\*] Not really. See next page.

## Math Expressions

You can write LaTeX math expressions inside a pair of dollar signs, e.g.  $\$alpha + beta$  renders  $\alpha + \beta$ . You can use the display style with double dollar signs:

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

Limitations:

1. The source code of a LaTeX math expression must be in one line, unless it is inside a pair of double dollar signs, in which case the starting \$ must appear in the very beginning of a line, followed immediately by a non-space character, and the ending \$ must be at the end of a line, led by a non-space character;
2. There should not be spaces after the opening \$ or before the closing \$;
3. Math does not work on the title slide (see #61 for a workaround).

12 / 31

13 / 31

## R Code

```
a boring regression
fit = lm(dist ~ 1 + speed, data = cars)
coef(summary(fit))

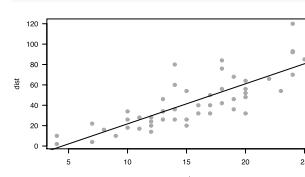
Estimate Std. Error t value Pr(>|t|)
(Intercept) -17.579095 6.7584402 -2.601058 1.231882e-02
speed 3.932409 0.4155128 9.463998 1.289836e-12

dojutsu = c("地獄天星", "天照", "加具土命", "神威", "须佐能乎", "無限月读")
grep('火', dojutsu, value = TRUE)

[1] "地獄天星" "天照"
```

## R Plots

```
par(mar = c(4, 1, 1, 1))
plot(cars, pch = 19, col = 'darkgray', las = 1)
abline(fit, lwd = 2)
```



14 / 31

15 / 31

# Capítulo 4

## Recomendaciones finales

### 4.1. Qué se ha conseguido

Se ha conseguido crear, utilizando software gratuito, un documento con el contenido de un trabajo fin de estudios en formato pdf de aspecto muy profesional, y teniendo muchísimas formas de personalizarlo.

### 4.2. Qué otras posibilidades nos brinda R Markdown

Con R Markdown se pueden crear documentos de otros tipos, como por ejemplo:

- Crear una presentación para la defensa del Trabajo Fin de Estudios.
- Crear notebooks en formatos: “html\_notebook” y “github\_document”, documentos html que guardan tanto el texto como las salidas de los chunks de código generadas. Permiten esconder/mostrar el código R utilizado, y también la recuperación del documento Rmd que lo generó.
- Crear libros web para compartir en internet con ayuda del paquete R “bookdown”<sup>1</sup>.
- Crear una página web sobre el Trabajo Fin de Estudios con R Markdown, como por ejemplo:
  - con el paquete R “blogdown”<sup>2</sup>.
  - con R Markdown websites<sup>3</sup>.
- Crear una aplicación web o cuadro de mandos que muestre los resultados de forma dinámica con ayuda del paquete “Shiny”: <http://shiny.rstudio.com>.
- Ver más posibilidades en la galería de R Markdown: <http://rmarkdown.rstudio.com/gallery.html>.

---

<sup>1</sup><https://bookdown.org>

<sup>2</sup><https://blogdown.org> y <https://bookdown.org/yihui/blogdown/>

<sup>3</sup>[http://rmarkdown.rstudio.com/rmarkdown\\_websites.html](http://rmarkdown.rstudio.com/rmarkdown_websites.html)

### 4.2.1. Crear una presentación para la defensa del trabajo fin de estudios con R Markdown

Puede encontrarse más información sobre este aspecto en <http://rmarkdown.rstudio.com/formats.html>.

Existen diferentes formatos para crear presentaciones, entre los más conocidos:

- **ioslides**<sup>4</sup>: presentación en formato HTML con “ioslides”. Ejemplo html<sup>5</sup>.
- **Slidy**<sup>6</sup>: presentación en formato HTML con “W3C Slidy”. Ejemplo html<sup>7</sup>.
- **Beamer**<sup>8</sup>: presentación en formato PDF con “el paquete LaTeX: Beamer”. Ejemplo 1 pdf<sup>9</sup>, Ejemplo 2 pdf<sup>10</sup>.
- **Paquete revealjs**<sup>11</sup>: presentación en formato HTML con “reveal.js”. Ejemplo html<sup>12</sup>.
- **Presentación con el paquete rmdshower**<sup>13</sup>: presentación en formato HTML con “shower”. Ejemplo html<sup>14</sup>.
- **Presentación con el paquete xaringan**<sup>15</sup>: presentación en formato HTML con “remark.js”. Ejemplo html<sup>16</sup>.

### 4.2.2. Algunos consejos sobre la presentación del TFE ante un tribunal

A continuación, se recogen algunas cuestiones prácticas a tener en cuenta para la preparación de la presentación del Trabajo Fin de Estudios ante un tribunal de evaluación:

- El número de diapositivas no debe ser muy grande, como mucho una por minuto del tiempo disponible.
- No hacer diapositivas con demasiado texto, es mejor resaltar la idea en una frase e ilustrarla con gráficos o tablas.
- En la primera transparencia aparece siempre el título y el nombre del autor. A esta, debe seguirle una transparencia con un esquema de la estructura (capítulos) del trabajo fin de estudios o de la presentación.
- Inmediatamente después, se deben presentar cuáles han sido los objetivos o la motivación del trabajo.

---

<sup>4</sup>[http://rmarkdown.rstudio.com/ioslides\\_presentation\\_format.html](http://rmarkdown.rstudio.com/ioslides_presentation_format.html)

<sup>5</sup><https://rpubs.com/cheyu/ioslideDemo>

<sup>6</sup>[http://rmarkdown.rstudio.com/slidy\\_presentation\\_format.html](http://rmarkdown.rstudio.com/slidy_presentation_format.html)

<sup>7</sup><https://rpubs.com/sdplus/vulcan74>

<sup>8</sup>[http://rmarkdown.rstudio.com/beamer\\_presentation\\_format.html](http://rmarkdown.rstudio.com/beamer_presentation_format.html)

<sup>9</sup><http://svmiller.com/rmarkdown-example.pdf>

<sup>10</sup>[https://mdozmorov.github.io/BIOS567/assets/presentation\\_Presentation/2\\_Reports\\_presentations.pdf](https://mdozmorov.github.io/BIOS567/assets/presentation_Presentation/2_Reports_presentations.pdf)

<sup>11</sup>[http://rmarkdown.rstudio.com/revealjs\\_presentation\\_format.html](http://rmarkdown.rstudio.com/revealjs_presentation_format.html)

<sup>12</sup><https://rpubs.com/znmrb/rwebappplate>

<sup>13</sup><https://github.com/MangoTheCat/rmdshower>

<sup>14</sup><https://shwr.me>

<sup>15</sup><https://github.com/yihui/xaringan>

<sup>16</sup><https://slides.yihui.name/xaringan/#1>

- Es un buen consejo, que en la exposición oral, **antes de terminar las explicaciones sobre una transparencia se utilice alguna frase que enlace con la siguiente transparencia.**
- No es adecuado leer continuamente el texto que aparece en las transparencias, generalmente no causa buena impresión.
- Hay que finalizar la presentación con las conclusiones, recomendaciones y/o líneas futuras de trabajo. Acabar agradeciendo al tribunal su atención, para darles paso al turno de preguntas. Siempre se debe mantener un actitud de respeto y atención al tribunal.

### 4.3. Líneas futuras de trabajo

Algunas de las líneas futuras de trabajo que actualmente tengo previstas están relacionadas con las siguientes cuestiones:

- Mejorar la documentación con los comentarios recibidos.
- Convertir esta plantilla en un paquete R, incluyendo una “Template” para R Markdown y algunos “RStudio Addins”.

# Apéndice A

## Paquetes R utilizados

A continuación se muestran todos los paquetes R utilizados, con el fin de informar bajo qué condiciones se generó este documento y facilitar así sus posibles nuevas reproducciones/mejoras.

```
#sessionInfo()
toLatex(sessionInfo(), locale = FALSE)
```

- R version 3.4.2 (2017-09-28), x86\_64-apple-darwin15.6.0
- Running under: macOS High Sierra 10.13.1
- Matrix products: default
- BLAS:  
*/Library/Frameworks/R.framework/Versions/3.4/Resources/lib/libRblas.0.dylib*
- LAPACK:  
*/Library/Frameworks/R.framework/Versions/3.4/Resources/lib/libRlapack.dylib*
- Base packages: base, datasets, graphics, grDevices, methods, stats, utils
- Other packages: bindrcpp 0.2, dplyr 0.7.4, ggmap 2.6.1, ggplot2 2.2.1, kableExtra 0.6.1, knitr 1.17
- Loaded via a namespace (and not attached): assertthat 0.2.0, backports 1.1.1, bindr 0.1, codetools 0.2-15, colorspace 1.3-2, compiler 3.4.2, digest 0.6.12, DT 0.2, evaluate 0.10.1, geosphere 1.5-7, glue 1.2.0, grid 3.4.2, gtable 0.2.0, hms 0.4.0, htmltools 0.3.6, htmlwidgets 0.9, httr 1.3.1, jpeg 0.1-8, jsonlite 1.5, labeling 0.3, lattice 0.20-35, lazyeval 0.2.1, magrittr 1.5, mapproj 1.2-5, maps 3.2.0, munsell 0.4.3, pkgconfig 2.0.1, plyr 1.8.4, png 0.1-7, proto 1.0.0, R6 2.2.2, Rcpp 0.12.14, readr 1.1.1, reshape2 1.4.2, RgoogleMaps 1.4.1, rjson 0.2.15, rlang 0.1.4, rmarkdown 1.8, rprojroot 1.2, rvest 0.3.2, scales 0.5.0, sp 1.2-5, stringi 1.1.6, stringr 1.2.0, tibble 1.3.4, tools 3.4.2, viridisLite 0.2.0, webshot 0.4.2, xml2 1.1.1, yaml 2.1.14

# Apéndice B

## Código comentado

En este apéndice se muestran y describen algunos de los ficheros más interesantes que contiene la plantilla para escribir un trabajo fin de estudios (TFE). Además, aparecen algunos consejos de cómo conseguir modificar/cambiar algunas características:

- Posibilidad de cambiar la estructura del TFE de una división en capítulos (nivel 1), a una división en secciones (`tfe_principal.Rmd`).
- Impresión a doble página con inicio de capítulo colocado correctamente (`the_principal.Rmd`).
- Cambiar el tamaño de fuente y los márgenes del documento (`tfe_principal.Rmd`).
- Enlaces de color azul a color negro (`tfe_principal.Rmd`).
- Modificar lo que aparece en la cabecera y pie del documento (`tfe_principal.Rmd`).
- Definición con LaTeX de una **caja de color para resaltar** ciertos elementos producidos con Markdown (`latex_preambulo.tex`).
- Modificar el interlineado vertical (`latex_preambulo.tex`).
- Cómo cambiar el fichero de portada que se va a utilizar (`tfe_principal.Rmd`) y cómo crear una variante de estos ficheros (`latex_paginatitulo_modXXXX.tex` y `latex_paginatitulo_mod_XXXX.tex`).

### B.1. Fichero `tfe_principal.Rmd`

El siguiente código es el contenido del fichero principal para construir el documento pdf que contendrá el trabajo fin de estudios. En los chunks de código R se han separado los 3 acentos graves para que no sean interpretados en este apéndice (en el fichero van juntos):

```

```

```
documentclass: book
classoption: a4paper
#classoption: a4paper,oneside
#bibliography: bib/library.bib
bibliography: ["bib/library.bib", "bib/paquetes.bib"]
csl: acm-sig-proceedings-long-author-list.csl
```

```
#csl: methods-in-ecology-and-evolution.csl
fontsize: 12pt # 10pt,11pt
geometry: margin = 2.5cm
link-citations: yes
```

- La opción **documentclass** tiene por defecto el valor “book” la cual organiza el TFE en varios capítulos (nivel 1). Es posible cambiarlo por el valor “article”, en este caso el TFE se organiza en secciones (representan al nivel 1).
- La opción **classoption** añade características a **documentclass**. El valor “a4paper” le indica que el documento tiene el tamaño de papel A4. Por defecto, cuando se usa “book”, las páginas de inicio de capítulo se colocan en páginas impares, insertando páginas en blanco si es necesario. Con esto se consigue que cuando el documento se imprime a doble cara, la página de inicio de capítulo queda a la derecha, como es habitual en los libros.



Si usamos el valor adicional “oneside” (no es el valor por defecto) creará el documento sin forzar a que los inicios de capítulo vayan en página impar.

- También es posible cambiar el tamaño de fuente en **fontsize**, por defecto es “12pt”, pero puede reducirse a “11pt” o “10pt”.
- Con la opción **geometry** se pueden cambiar los márgenes del documento. En este caso se ha fijado a un margen de “2.5cm”.
- Si se comentan las siguientes dos líneas de la cabecera YAML o se eliminan, se consigue que el documento pdf que se crea, tenga todos los enlaces (de índices y a enlaces internos o externos) en negro, y se resalten únicamente cuando se acerca el cursor sobre ellos. Esto está indicado cuando se quiere imprimir el documento en blanco y negro.

```
linkcolor: blue
urlcolor: blue
output:
 pdf_document:
 fig_caption: yes
 includes:
 after_body: latex/latex_anter_enddoc.tex
```

- Al cambiar el fichero que aparece en la opción `before_body` permite seleccionar otra portada diferente para el TFE. En la carpeta “portadas” se pueden encontrar páginas de inicio para algunas titulaciones y también otras variantes de la portada general. En estos ficheros hay que cambiar algunos datos, como: autor, fecha, título, etc.

```

before_body: portadas/latex_paginatitulo_modTFE.tex
in_header: latex/latex_preambulo.tex
keep_tex: yes
number_sections: yes

nocite: |
@Luque2017, @RStudio, @R-base,
@R-knitr, @R-rmarkdown, @R-dplyr, @R-ggplot2, @R-xtable,
@R-stringr, @Techopedia, @Pentaho2, @R-kableExtra, @Wickham2017, @Example1999
...
<!-- , @Example1999, @Example2000 -->
` ``{r global_options, include=FALSE}
knitr::opts_chunk$set(fig.path = 'figurasR/',
echo = FALSE, warning = FALSE, message = FALSE,
fig.pos="H", fig.align="center", out.width="95%",
cache=TRUE)

knitr::write_bib(c("knitr", "rmarkdown", "dplyr", "ggplot2", "xtable",
"stringr", "shiny", "flexdashboard", "htmlwidgets",
"bookdown", "kableExtra"),
file="bib/paquetes.bib", width = 60)
options(knitr.table.format = "latex",
kableExtra.latex.load_package = FALSE)
` ``

<!-- Indentar el texto al inicio de cada nuevo párrafo -->
\setlength{\parindent}{1em}

\pagestyle{fancy}
\fancyhead[LE,RO]{}
\fancyhead[LO,RE]{}
\renewcommand{\headrulewidth}{0pt}
\renewcommand{\footrulewidth}{0pt}
\pagenumbering{roman}

\setcounter{tocdepth}{4}
\subpdfbookmark{Índice General}{indice}
\tableofcontents

```

```
\cleardoublepage
<!-- \clearpage -->
\section*{Prólogo}
\addcontentsline{toc}{section}{Prólogo}

` ``{r child = 'Prólogo.Rmd'}
` `

\cleardoublepage
<!-- \clearpage -->

\section*{Resumen}
\addcontentsline{toc}{section}{Resumen}

` ``{r child = 'resumen.Rmd'}
` `

\clearpage
\section*{Abstract}
\addcontentsline{toc}{section}{Abstract}

` ``{r child = 'abstract.Rmd'}
` `

\cleardoublepage
<!-- \clearpage -->
\listoffigures
\addcontentsline{toc}{section}{Índice de Figuras}

\cleardoublepage
<!-- \clearpage -->
\listoftables
\addcontentsline{toc}{section}{Índice de Cuadros}

\cleardoublepage
\pagenumbering{arabic}
```

- En las siguientes líneas es posible especificar lo que aparece en la cabecera y en el pie del documento.

```
\fancyhead[LE,RO]{\scriptsize\rightmark}
\fancyfoot[L0,RE]{\scriptsize\slshape \leftmark}
\fancyfoot[C]{}
```

```
\fancyfoot[LE,RO]{\footnotesize\thepage}
\renewcommand{\headrulewidth}{0.4pt}
\renewcommand{\footrulewidth}{0.4pt}

` ``{r child = 'capitulo01.Rmd'}
` ``

\FloatBarrier

` ``{r child = 'capitulo02.Rmd'}
` ``

\FloatBarrier

` ``{r child = 'capitulo03.Rmd'}
` ``

\FloatBarrier

` ``{r child = 'capitulo04.Rmd'}
` ``

\FloatBarrier

\appendix

` ``{r child = 'apendice01.Rmd'}
` ``

` ``{r child = 'apendice02.Rmd'}
` ``

\FloatBarrier
\cleardoublepage

\fancyhead[CO,CE]{Bibliografía}

Bibliografía
```

- **Nota:** es posible dividir el TFE en varias partes, y cada una de ellas compuesta por sus propios capítulos, incluso con su propia portada. Esto puede ser interesante, cuando en un único documento pdf, se crea una parte del trabajo para un lector analista y otra parte para un lector “sponsor” o empresario.

Utilizar el fichero “modelos/tfe\_principal2partes.Rmd” para ver un ejemplo.

## B.2. Fichero: capitulo01.Rmd

```

documentclass: book
classoption: a4paper
#bibliography: bib/library.bib
bibliography: ["bib/library.bib", "bib/paquetes.bib"]
#csl: methods-in-ecology-and-evolution.csl
csl: acm-sig-proceedings-long-author-list.csl
urlcolor: blue #black
linkcolor: blue #black
link-citations: yes
fontsize: 12pt # 10pt,11pt
geometry: margin = 2.5cm #1.2in
output:
 pdf_document:
 keep_tex: no
 number_sections: yes
toc: yes
 fig_caption: yes
 includes:
 in_header: latex/latex_preambulo.tex
 after_body: latex/latex_antes_enddoc.tex

` ``{r include=FALSE}
knitr::opts_chunk$set(fig.path = 'figurasR/',
 echo = TRUE, warning = FALSE, message = FALSE,
 fig.pos="H",fig.align="center",out.width="95%",
 cache=FALSE)
` ``
```

- El siguiente comando LaTeX \setcounter{chapter}{2} (quitando los comentarios) permite establecer el número del capítulo correctamente si se compila de forma individual el capítulo.

```
<!-- \setcounter{chapter}{2} -->
<!-- \setcounter{chapter}{2} escribir 2 para capítulo 3 -->
<!-- \pagenumbering{arabic} -->

\fancyhead[LE,RO]{\scriptsize\rightmark}
\fancyfoot[L0,RE]{\scriptsize\slshape \leftmark}
```

```
\fancyfoot[C]{}
\fancyfoot[LE,RO]{\footnotesize\thepage}
\renewcommand{\headrulewidth}{0.4pt}
\renewcommand{\footrulewidth}{0.4pt}

Título del Capítulo

Primera sección
```

### B.3. Fichero: `latex_preambulo.tex`

Este fichero LaTeX se puede personalizar, aunque hay que tener conocimientos básicos de LaTeX.

En este fichero:

- Se cargan paquetes LaTeX necesarios.
- Se definen ciertas características de la cabecera y el pie de página del documento.
- Se definen abreviaturas para usar determinadas características de LaTeX: centrar texto, ...
- Se definen entornos LaTeX específicos para matemáticas.
- Se definen contadores para numerar determinados aspectos.
- Se definen colores.

```
\usepackage[spanish,es-nodecimaldot,es-noshorthands]{babel}
\usepackage{float}
\usepackage{placeins}
\usepackage{fancyhdr}

\usepackage{chngcntr}
\usepackage{microtype}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc} % Use 8-bit encoding that has 256 glyphs

%\usepackage[usenames,dvipsnames]{color}
\usepackage[dvipsnames,table]{xcolor}
\usepackage{pdfpages}
\usepackage{natbib}

% Para portada: latex_paginatitulo_mod_ST02.tex (inicio)
\usepackage{tikz}
\usepackage{epigraph}
```

```

\input{portadas/latex_paginatitulo_mod_ST02_add.sty}
% Para portada: latex_paginatitulo_mod_ST02.tex (fin)

% Para portada: latex_paginatitulo_mod_OV01.tex (inicio)
\usepackage{cpimod}
% Para portada: latex_paginatitulo_mod_OV01.tex (fin)

% Para portada: latex_paginatitulo_mod_OV03.tex (inicio)
\usepackage{KTHEEtitlepage}
% Para portada: latex_paginatitulo_mod_OV03.tex (fin)

\renewcommand{\contentsname}{Índice}
\renewcommand{\listfigurename}{Índice de figuras}
\renewcommand{\listtablename}{Índice de tablas}
%\onehalfspacing
%\counterwithin{figure}{chapter}
%\counterwithin{table}{chapter}
\usepackage{amsthm}

% \newtheorem{theorem}{Teorema}
% \newtheorem{etheorem}[theorem]{Teorema}
% \newtheorem{eproposition}[theorem]{Proposición}
% \newtheorem{ecorollary}[theorem]{Corolario}
% \newtheorem{elemma}[theorem]{Lema}
%
% \theoremstyle{definition}
% \newtheorem{edefinition}[theorem]{Definición}
%
% \theoremstyle{example}
% \newtheorem{eexample}[theorem]{Ejemplo}
% \newtheorem{eexercise}[theorem]{Ejercicio}
% \newtheorem{eproblem}[theorem]{Problema}
%
% \theoremstyle{plain}
% \newtheorem{eremark}[theorem]{Nota}
%
% \newcommand{\betheorem}{\begin{etheorem}}
% \newcommand{\eetheorem}{\end{etheorem}}
% \newcommand{\beproposition}{\begin{eproposition}}
% \newcommand{\eeproposition}{\end{eproposition}}
%
% \newcommand{\bedefinition}{\begin{edefinition}}
% \newcommand{\eedefinition}{\end{edefinition}}
%
% \newcommand{\beexample}{\begin{eexample}}
% \newcommand{\eeexample}{\end{eexample}}
% \newcommand{\beexercise}{\begin{eexercise}}

```

```
% \newcommand{\eeexercise}{\end{eexercise}}
% \newcommand{\beproblem}{\begin{eproblem}}
% \newcommand{\eeproblem}{\end{eproblem}}
%
% \newcommand{\beremark}{\begin{eremark}}
% \newcommand{\eeremark}{\end{eremark}}
```

```
% \def\proofname{Demostraci\'on}
% \def\abstractname{Resumen}
% %\def\chaptername{Tema}
% \def\appendixname{Ap\'endice}
% \def\listfigurename{\textit{\'Indice de Figuras}}
% \def\listtablename{\textit{\'Indice de Tablas}}
% \def\indexname{\textit{\'Indice de Materias}}
% \def\figurename{Figura}
% \def\tablename{Tabla}
% \def\partname{Parte}
% \def\pagename{P\'agina}
% \gdef\contentsname{\textit{'Indice}} \gdef\bibname{Bibliograf\'ia}
% \gdef\refname{Referencias bibliogr\'aficas}
% \def\today{\number\day\de\space\ifcase\month\or
% enero\or febrero\or marzo\or abril\or mayo\or junio\or julio\or agosto\or
% septiembre\or octubre\or noviembre\or diciembre\fi \space de~\number\year}
```

```
% \newtheorem{ejemplo}{Ejemplo}[section]
% \newtheorem{ejem}{Ejemplo}[section]
% \newtheorem{props}{Proposici\'on}[section]
% \newtheorem{prop}{Proposici\'on}[section]
% \newtheorem{mdef}{Definici\'on}[section]
% \newtheorem{defn}{Definici\'on}[section]
```

```
\newcommand{\bcols}{}%
\newcommand{\ecols}{}%
\newcommand{\bcol}[1]{\begin{minipage}{#1\linewidth}}%
\newcommand{\ecol}{\end{minipage}}%
% \newcommand{\beexample}{\begin{example} \ \\ \ }
% \newcommand{\eeexample}{\end{example}}%
\newcommand{\balertblock}[1]{\begin{alertblock}{#1}}%
\newcommand{\ealertblock}{\end{alertblock}}%
\newcommand{\bitemize}{\begin{itemize}}%
\newcommand{\eitemize}{\end{itemize}}%
\newcommand{\benumerate}{\begin{enumerate}}%
\newcommand{\eenumerate}{\end{enumerate}}%
\newcommand{\saltopagina}{\newpage}
```

```

\newcommand{\bcenter}{\begin{center}}
\newcommand{\ecenter}{\end{center}}
% \newcommand{\bbcenter}{\begin{center}}
% \newcommand{\eecenter}{\end{center}}
```

*%De: https://texblog.org/2007/11/07/headerfooter-in-latex-with-fancyhdr/*

```

% E: Even page
% O: Odd page
% L: Left field
% C: Center field
% R: Right field
% H: Header
% F: Footer
%\fancyhead[CO,CE]{Resultados}
```

*%OPCIÓN 1*

```

% \fancyhead[LE,RO]{\slshape \rightmark}
% \fancyhead[LO,RE]{\slshape \leftmark}
% \fancyfoot[C]{\thepage}
% \renewcommand{\headrulewidth}{0.4pt}
% \renewcommand{\footrulewidth}{0pt}
```

*%OPCIÓN 2*

```

% \fancyhead[LE,RO]{\slshape \rightmark}
% \fancyfoot[LO,RE]{\slshape \leftmark}
% \fancyfoot[LE,RO]{\thepage}
% \renewcommand{\headrulewidth}{0.4pt}
% \renewcommand{\footrulewidth}{0.4pt}
```

*%%%%%%%%%*

```

\usepackage{calc,amsfonts}
% Elimina la cabecera de páginas impares vacías al finalizar los capítulos
\usepackage{emptypage}
```

```

\makeatletter
```

```

\definecolor{ocre}{RGB}{25,25,243} % Define el color naranja usado
% para resaltar algunas salidas
```

```

\usepackage{calc}
```

```

\usepackage{lipsum}
```

```
\usepackage{tikz} % Requerido para dibujar formas personalizadas

\usepackage{amsmath,amsthm,amssymb,amsfonts}

% Define el estilo texto theorem para cada tipo definido anteriormente
\newcounter{dummy}
\numberwithin{dummy}{section}
\theoremstyle{ocrenumbox}
\newtheorem{theoremeT}[dummy]{Teorema} % (Pedro: Theorem)
\newtheorem{problem}{Problema}[chapter] % (Pedro: Problem)
\newtheorem{exerciseT}{Ejercicio}[chapter] % (Pedro: Exercise)
\theoremstyle{blacknumex}
\newtheorem{exampleT}{Ejemplo}[chapter] % (Pedro: Example)
\theoremstyle{blacknumbox}
\newtheorem{vocabulary}{Vocabulario}[chapter] % (Pedro: Vocabulary)
\newtheorem{definitionT}{Definición}[section] % (Pedro: Definition)
\newtheorem{corollaryT}[dummy]{Corolario} % (Pedro: Corollary)
\theoremstyle{ocrenum}
\newtheorem{proposition}{Proposición}[dummy] % (Pedro: Proposition)

\usepackage[framemethod=default]{mdframed}

\newcommand{\intoo}[2]{\mathopen{[]}\#1\,,\#2\mathclose{[]}}
\newcommand{\ud}{\mathop{\mathrm{d}}\nolimits}
\newcommand{\intff}[2]{\mathopen{[]}\#1\,,\#2\mathclose{[]}}
\newtheorem{notation}{Notation}[chapter]

\mdfdefinestyle{exampledefault}{%
rightline=true,innerleftmargin=10,innerrightmargin=10,
frametitlerule=true,frametitlerulecolor=green,
frametitlebackgroundcolor=yellow,
frametitlerulewidth=2pt}

% Theorem box
\newmdenv[skipabove=7pt,
skipbelow=7pt,
backgroundcolor=black!5,
linecolor=ocre,
innerleftmargin=5pt,
innerrightmargin=5pt,
```

```

innertopmargin=10pt, %5pt
leftmargin=0cm,
rightmargin=0cm,
innerbottommargin=5pt]{tBox}

% Exercise box
\newmdenv[skipabove=7pt,
skipbelow=7pt,
rightline=false,
leftline=true,
topline=false,
bottomline=false,
backgroundcolor=ocre!10,
linecolor=ocre,
innerleftmargin=5pt,
innerrightmargin=5pt,
innertopmargin=10pt, %5pt
innerbottommargin=5pt,
leftmargin=0cm,
rightmargin=0cm,
linewidth=4pt]{eBox}

% Definition box
\newmdenv[skipabove=7pt,
skipbelow=7pt,
rightline=false,
leftline=true,
topline=false,
bottomline=false,
linecolor=ocre,
innerleftmargin=5pt,
innerrightmargin=5pt,
innertopmargin=10pt, %0pt
leftmargin=0cm,
rightmargin=0cm,
linewidth=4pt,
innerbottommargin=0pt]{dBox}

% Corollary box
\newmdenv[skipabove=7pt,
skipbelow=7pt,
rightline=false,
leftline=true,
topline=false,
bottomline=false,
linecolor=gray,
backgroundcolor=black!5,

```

```

innerleftmargin=5pt,
innerrightmargin=5pt,
innertopmargin=10pt,%5pt
leftmargin=0cm,
rightmargin=0cm,
linewidth=4pt,
innerbottommargin=5pt]{cBox}

% Crea un entorno para cada tipo de theorem y le asigna un estilo
% con ayuda de las cajas coloreadas anteriores
\newenvironment{theorem}{\begin{tBox}\begin{theoremeT}}{\end{theoremeT}}
\end{tBox}}
\newenvironment{exercise}{\begin{eBox}\begin{exercisET}}{\hfill{
\color{ocre}\tiny\ensuremath{\blacksquare}}\end{exercisET}\end{eBox}}
\newenvironment{definition}{\begin{dBox}\begin{definitionT}}{
\end{definitionT}\end{dBox}}
\newenvironment{example}{\begin{exampleT}}{\hfill{\tiny
\ensuremath{\blacksquare}}\end{exampleT}}
\newenvironment{corollary}{\begin{cBox}\begin{corollaryT}}{
\end{corollaryT}\end{cBox}>

% ENVIRONMENT remark
\newenvironment{remark}{\par\vspace{10pt}\small
% Espacio blanco vertical sobre la nota y tamaño de fuente menor
\begin{list}{}{
\leftmargin=35pt % Indentación sobre la izquierda
\rightmargin=25pt}\item\ignorespaces % Indentación sobre la derecha
\makebox[-2.5pt]{\begin{tikzpicture}[overlay]
\node[draw=ocre!60,line width=1pt,circle,fill=ocre!25,
font=\sffamily\bfseries,inner sep=2pt,outer sep=0pt]
at (-15pt,0pt){\textcolor{ocre}{N}};}
\end{tikzpicture}} % R naranja en un círculo (Pedro)
\advance\baselineskip -1pt}\end{list}\vskip5pt}
% Espaciado de línea más estrecho y espacio en blanco después del comentario

\newenvironment{solutionExe}{\par\vspace{10pt}\small
\begin{list}{}{
\leftmargin=35pt
\rightmargin=25pt}\item\ignorespaces
\makebox[-2.5pt]{\begin{tikzpicture}[overlay]
\node[draw=ocre!60,line width=1pt,circle,fill=ocre!25,
font=\sffamily\bfseries,inner sep=2pt,outer sep=0pt]
at (-15pt,0pt){\textcolor{ocre}{S}};}
\end{tikzpicture}}
\advance\baselineskip -1pt}\end{list}\vskip5pt}

```

```

\newenvironment{solutionExa}{\par\vspace{10pt}\small
\begin{list}{}{
\leftmargin=35pt
\rightmargin=25pt}\item\ignorespaces
\makebox[-2.5pt]{\begin{tikzpicture}[overlay]
\node[draw=ocre!60,line width=1pt,circle,fill=ocre!55,
font=\sffamily\bfseries,inner sep=2pt,outer sep=0pt]
at (-15pt,0pt){\textcolor{ocre}{S}};%
\end{tikzpicture}}
\advance\baselineskip -1pt}{\end{list}\vskip5pt}

\usepackage{float}

\usepackage{tcolorbox}

\usetikzlibrary{trees}

\theoremstyle{ocrenum}
\newtheorem{solutionT}[dummy]{Solución} % (Pedro: Corollary)
\newenvironment{solution}{\begin{cBox}\begin{solutionT}}{\end{solutionT}\end{cBox}}

\newcommand{\tcolorboxsolucion}[2]{%
\begin{tcolorbox}[colback=green!5!white,colframe=green!75!black,title=#1]
#2
\tcblower % pone una linea discontinua
\end{tcolorbox}
}% final definición comando

\newtcbbox{\mybox}[1][green]{on line,
arc=0pt,outer arc=0pt,colback=#1!10!white,colframe=#1!50!black,
boxsep=0pt,left=1pt,right=1pt,top=2pt,bottom=2pt,
boxrule=0pt,bottomrule=1pt,toprule=1pt}

\mdfdefinestyle{exampledefault}{%
rightline=true,innerleftmargin=10,innerrightmargin=10,
frametitlerule=true,frametitlerulecolor=green,
frametitlebackgroundcolor=yellow,
frametitlerulewidth=2pt}

```

```
\newcommand{\betheorem}{\begin{theorem}}
\newcommand{\eetheorem}{\end{theorem}}
\newcommand{\bedefinition}{\begin{definition}}
\newcommand{\eedefinition}{\end{definition}}

\newcommand{\beremark}{\begin{remark}}
\newcommand{\eeremark}{\end{remark}}
\newcommand{\beexercise}{\begin{exercise}}
\newcommand{\eeexercise}{\end{exercise}}
\newcommand{\beexample}{\begin{example}}
\newcommand{\eeexample}{\end{example}}
\newcommand{\becorollary}{\begin{corollary}}
\newcommand{\eecorollary}{\end{corollary}}

\newcommand{\besolutionExe}{\begin{solutionExe}}
\newcommand{\eesolutionExe}{\end{solutionExe}}
\newcommand{\besolutionExa}{\begin{solutionExa}}
\newcommand{\eesolutionExa}{\end{solutionExa}}
```

- **Truco:** Si incluimos el siguiente código LaTeX al final del fichero “latex\_preambulo.tex”:

```
\renewcommand{\chaptername}{Entregable}
\addto\captionsspanish{\renewcommand{\chaptername}{Entregable}}
```

En lugar de aparecer “Capítulo” al inicio de cada capítulo, aparecería “Entregable”.

- En las siguientes líneas se define con código LaTeX una caja de color con una línea gruesa a la izquierda que se utilizará para resaltar las salidas que se obtienen de aplicar markdown. El texto que va a la caja de color va entre los dos siguientes comandos: \bmarkdownsal y \emarkdownsal (ver página ??).

```
% Caja Salida Markdown
\newmdenv[skipabove=7pt,
skipbelow=7pt,
rightline=false,
leftline=true,
topline=false,
bottomline=false,
backgroundcolor=GreenYellow!10,
linecolor=GreenYellow!80,
innerleftmargin=5pt,
innerrightmargin=5pt,
innertopmargin=10pt, %5pt
innerbottommargin=5pt,
leftmargin=0cm,
rightmargin=0cm,
```

```

\linewidth=4pt]{mBox}

%% R Markdown
\newenvironment{markdownal}{\begin{mBox}}{\end{mBox}}
\newcommand{\bmarkdownal}{\begin{markdownal}}
\newcommand{\emarkdownal}{\end{markdownal}}

\usepackage{array}
\usepackage{multirow}
\usepackage{wrapfig}
\usepackage{colortbl}
\usepackage{pdflscape}
\usepackage{tabu}
\usepackage{threeparttable}

```

- Si descomentamos la siguiente instrucción LaTeX, se conseguiría en todo el documento que el espacio vertical entre líneas o interlineado fuese de una línea y media respecto al interlineado normal. Al estar comentada, por defecto se utilizará el espacio interlineal normal o recomendado.

```
%\renewcommand{\baselinestretch}{1.5}
```

## B.4. Fichero: latex\_paginatitulo\_modTFGE.tex

En este fichero LaTeX se personaliza la portada del trabajo fin de estudios, cambiando el título, autor, etc.

Como se comentó anteriormente en este trabajo, para cada titulación existe un fichero:

- “latex\_paginatitulo\_modXXXX.tex”, donde XXXX puede ser: TFGE, TFGM, TFDGFM, TFDGME, TFMUM, TFMDS, TFE.
- También existen otras variantes “latex\_paginatitulo\_mod\_XXXX.tex”, donde XXXX puede ser: LT01, LT02, ST01, ST02, OV01, OV02, OV03.

Se encuentran en la subcarpeta: “portadas”.

Se recomienda hacer una copia del fichero en la carpeta “portadas” que contenga la portada que se quiera utilizar, personalizando los datos necesarios. No hay que olvidar, modificar la cabecera YAML del documento R Markdown principal “tfe\_principal.Rmd”, para que el campo: “before\_body:” apunte al nuevo fichero de portada.

```

% nada
\begin{titlepage}

\newcommand{\HRule}{\rule{\linewidth}{0.5mm}}
% Define un nuevo comando para definir el grosor de la linea horizontal

```

```

\center % Center todo lo escrito en esta página

\begin{minipage}{14cm}
% Sección LOGO
\center

\includegraphics[width=8cm,height=8cm]{logo}\[0.5cm]
% Incluye logo del departamento/universidad
% esto requiere el paquete LaTeX graphicx

%-----
% CABECERA
%-----
\textsc{\LARGE Grado en Estadística}\[2.5cm]

%-----
% TÍTULO
%-----
\rule[1.7mm]{2cm}{0.5mm}
\hfill
\textsc{\Large TRABAJO FIN DE GRADO}
\hfill
\rule[1.7mm]{2cm}{0.5mm}
\[0.75cm]

%\bfseries
{\Huge
\textbf{\textit{
Introducción a \[0.2cm]
la Estadística Aplicada \[0.5cm]
con ayuda de R
}}}\[0.75cm]

\HRule \[4cm]

{\Large
Marta García Moreno} \[0.5cm]

{\large
Sevilla, Octubre de 2017
}

\end{minipage}

```

```
\vfill % Rellena el resto de la página con espacio en blanco

\cleardoublepage
% \newpage{ }
\thispagestyle{empty}
\end{titlepage}

\raggedbottom
```

## B.5. Ficheros: a\_TODO.Rmd

Este fichero R Markdown tiene la misma estructura que los ficheros: a\_NEWS.Rmd y a\_README.Rmd, lo que los diferencia es el contenido que deben tener.

El fichero “a\_TODO.Rmd” contiene el siguiente código:

```

title: "TODO"
author: ''
date: "03 de diciembre de 2017"
output:
 html_document:
 toc: yes
 pdf_document: default
 word_document: default

TODO o tareas por realizar
```

# Bibliografía

- [1] Allaire, J., Cheng, J., Xie, Y., McPherson, J., Chang, W., Allen, J., Wickham, H., Atkins, A., Hyndman, R. and Arslan, R. 2017. *Rmarkdown: Dynamic documents for r*.
- [2] Dahl, D.B. 2016. *Xtable: Export tables to latex or html*.
- [3] Example, T. 1999. This is an exmaple article not cited in the text. *Journal of Examples*. 4, 2 (1999), 1–9.
- [4] Goossens, M., Mittelbach, F. and Samarin, A. 1993. *The latex companion*. Addison-Wesley.
- [5] Gravitar, I. sin límites 2013. ¿Qué es pentaho? Disponible en <http://gravitar.biz/pentaho/>, version 1.6.0.
- [6] Knuth, D. Knuth: Computers and typesetting.
- [7] Luque-Calvo, P.L. 2017. *Escribir un trabajo fin de estudios con r markdown*. Disponible en <http://destio.us.es/calvo>.
- [8] R Core Team 2016. *R: A language and environment for statistical computing*. R Foundation for Statistical Computing.
- [9] RStudio Team 2015. *RStudio: Integrated development environment for r*. RStudio, Inc.
- [10] Techopedia “Definition - what does business intelligence (bi) mean?” Disponible en <https://www.techopedia.com/definition/345/business-intelligence-bi>.
- [11] Wickham, H. 2017. *Stringr: Simple, consistent wrappers for common string operations*.
- [12] Wickham, H. and Chang, W. 2016. *Ggplot2: Create elegant data visualisations using the grammar of graphics*.
- [13] Wickham, H. and Grolemund, G. 2017. *R for data science: Import, tidy, transform, visualize, and model data*. O'Reilly.
- [14] Wickham, H., Francois, R., Henry, L. and Müller, K. 2017. *Dplyr: A grammar of data manipulation*.
- [15] Xie, Y. 2017. *Bookdown: Authoring books and technical documents with r markdown*.
- [16] Xie, Y. 2017. *Knitr: A general-purpose package for dynamic report generation in r*.
- [17] Zhu, H. 2017. *KableExtra: Construct complex table with 'kable' and pipe syntax*.