

Herramientas para la elaboración de tareas y trabajos de curso

F. Tusell*

Curso 2.011-2.012

Índice

1. Introducción	2
2. Alternativas	3
2.1. R + editor o procesador de texto.	3
2.2. R en batch + editor o procesador de texto.	4
2.3. Emacs + ESS	6
2.4. Emacs + L ^A T _E X+ “corta y pega”	8
2.5. Emacs + L ^A T _E X+ Sweave	11
2.6. Uso de plantillas	14
2.7. ¿No hay herramientas “what you see is what you get”?	14
2.7.1. El modo Preview de Emacs, TeXmacs, y Lyx	14
2.7.2. R Commander	14
2.7.3. RStudio	16
A. Información sobre L^AT_EX.	17
B. Información sobre Emacs.	17
C. ¿Puedo instalar esto en mi casa?	18

*Actualización del día 3 de noviembre de 2011. La última versión de este documento, posiblemente más reciente, puede obtenerse en <http://www.et.bs.ehu.es/~etptupaf> o de MOODLE, <http://moodle3.ehu.es>.

1. Introducción

Los aspectos puramente materiales pueden absorber mucho tiempo y esfuerzo en la realización de tareas y trabajos de curso. Este documento describe algunas herramientas disponibles en el Laboratorio de Economía Cuantitativa (LEC), cuya utilización puede eliminar errores y hacer mucho más productivo el trabajo¹.

En esencia, la realización de trabajo aplicado en asignaturas como **Estadística: Modelos Lineales** o **Estadística: Análisis Multivariante** implica integrar ejecuciones de programas de ordenador con comentarios. En su forma más rudimentaria, esta integración puede ser una impresión con comentarios añadidos a mano, o mediante un programa editor. Lo que sigue detalla diversas alternativas y sus pros y contras.

Todo funciona sin más en el LEC, en que distintas herramientas han sido integradas. Si se quiere replicar en un ordenador personal y/o bajo otro sistema operativo, es preciso realizar dicha integración.

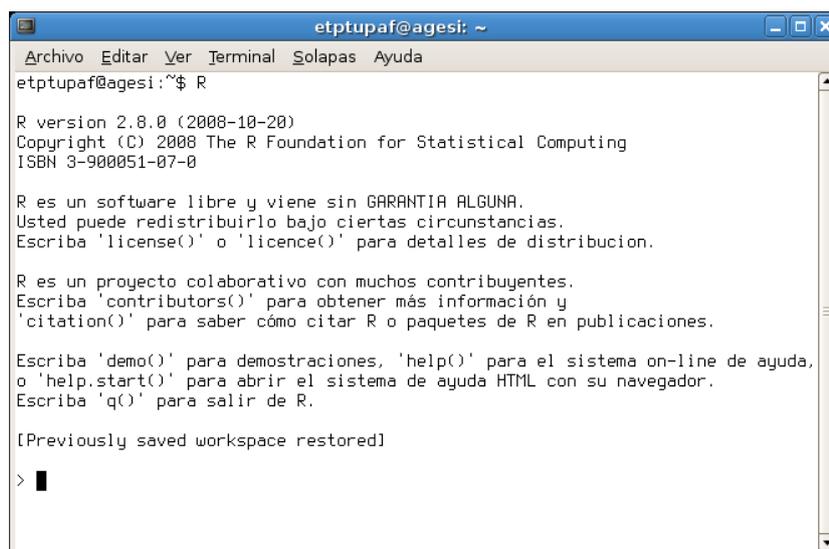
¹En el LEC, todo está montado sobre Linux. Algunas de las herramientas que se describen a continuación podrían instalarse con mayor o menor esfuerzo sobre otros sistemas operativos, en particular Windows XP o Vista. El LEC tiene por objetivo, entre otros, el obviar la necesidad de hacerlo y poner a disposición de los alumnos herramientas potentes, productivas y escalables, que sean reflejo del estado del arte y que potencien en lugar de limitar su trabajo.

2. Alternativas

2.1. R + editor o procesador de texto.

Modus operandi

1. Abrir una sesión interactiva tecleando en una ventana de mandatos R.



```
etptupaf@agesi: ~
Archivo Editar Ver Terminal Solapas Ayuda
etptupaf@agesi:~$ R

R version 2.8.0 (2008-10-20)
Copyright (C) 2008 The R Foundation for Statistical Computing
ISBN 3-900051-07-0

R es un software libre y viene sin GARANTIA ALGUNA.
Usted puede redistribuirlo bajo ciertas circunstancias.
Escriba 'license()' o 'licence()' para detalles de distribución.

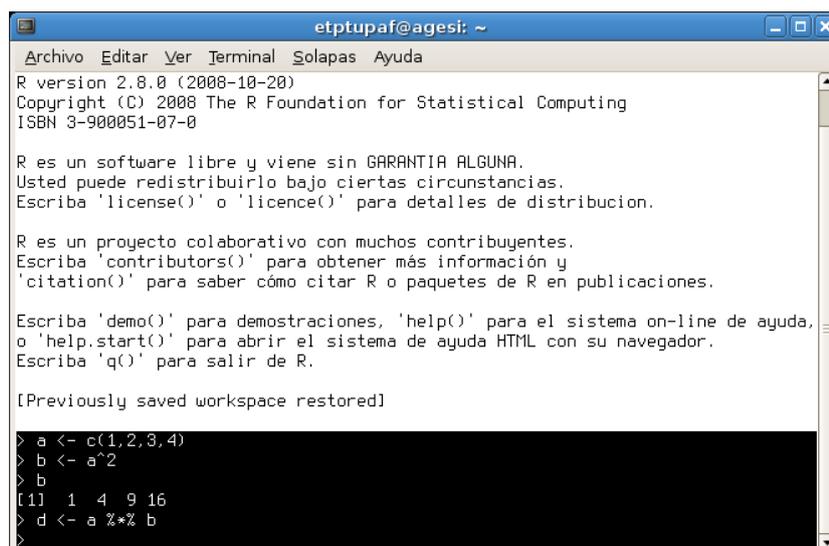
R es un proyecto colaborativo con muchos contribuyentes.
Escriba 'contributors()' para obtener más información y
'citation()' para saber cómo citar R o paquetes de R en publicaciones.

Escriba 'demo()' para demostraciones, 'help()' para el sistema on-line de ayuda,
o 'help.start()' para abrir el sistema de ayuda HTML con su navegador.
Escriba 'q()' para salir de R.

[Previously saved workspace restored]

> █
```

2. Teclear las instrucciones que necesites, y “cortar y pegar” instrucciones y resultados en un editor o procesador de texto.



```
etptupaf@agesi: ~
Archivo Editar Ver Terminal Solapas Ayuda
R version 2.8.0 (2008-10-20)
Copyright (C) 2008 The R Foundation for Statistical Computing
ISBN 3-900051-07-0

R es un software libre y viene sin GARANTIA ALGUNA.
Usted puede redistribuirlo bajo ciertas circunstancias.
Escriba 'license()' o 'licence()' para detalles de distribución.

R es un proyecto colaborativo con muchos contribuyentes.
Escriba 'contributors()' para obtener más información y
'citation()' para saber cómo citar R o paquetes de R en publicaciones.

Escriba 'demo()' para demostraciones, 'help()' para el sistema on-line de ayuda,
o 'help.start()' para abrir el sistema de ayuda HTML con su navegador.
Escriba 'q()' para salir de R.

[Previously saved workspace restored]

> a <- c(1,2,3,4)
> b <- a^2
> b
[1] 1 4 9 16
> d <- a %*% b
>
```

3. Añadir los comentarios en el editor o procesador de texto.

Ventajas

Simple, rápido, cada cual puede emplear el editor o procesador de texto con el que esté más familiarizado.

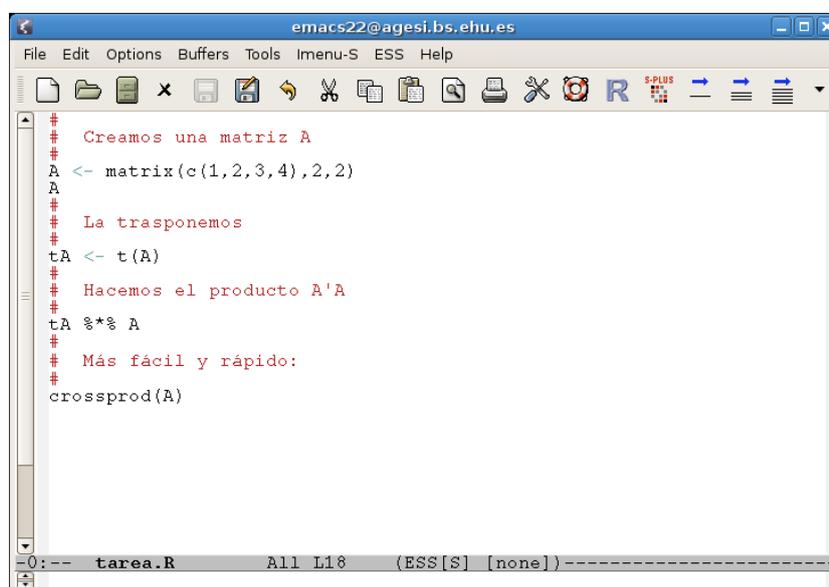
Inconvenientes

Si se detecta un error en un fragmento del código R, hay que volver a ejecutarlo, cortarlo y pegarlo. *Si unos resultados dependen de otros, hay que rehacer los cálculos erróneos y todos los posteriores, y cortarlos y pegarlos trozo a trozo.* La presentación es pobre.

2.2. R en batch + editor o procesador de texto.

Modus operandi

1. Mediante el editor que prefiramos, crear el código R en un fichero, que salvaremos como, por ejemplo, `tarea.R`. Podemos intercalar comentarios precediendo cada línea del símbolo `#`. Por ejemplo (usando Emacs como editor):



```
emacs22@agesi.bs.ehu.es
File Edit Options Buffers Tools Imenu-S ESS Help
#
# Creamos una matriz A
#
A <- matrix(c(1,2,3,4),2,2)
A
#
# La trasponemos
#
tA <- t(A)
#
# Hacemos el producto A'A
#
tA %*% A
#
# Más fácil y rápido:
#
crossprod(A)
-0:-- tarea.R All L18 (ESS[S] [none])
```

2. Tras guardar el código, desde una ventana de mandatos teclearemos:

R CMD BATCH tarea.R

3. La salida se deposita en el fichero `tarea.Rout`, que podemos visualizar mediante, por ejemplo, Emacs. En `tarea.Rout` aparecen código, comentarios y resultados entreverados:

```

R version 2.8.0 (2008-10-20)
Copyright (C) 2008 The R Foundation for Statistical Computing
ISBN 3-900051-07-0

R es un software libre y viene sin GARANTIA ALGUNA.
Usted puede redistribuirlo bajo ciertas circunstancias.
Escriba 'license()' o 'licence()' para detalles de distribucion.

R es un proyecto colaborativo con muchos contribuyentes.
Escriba 'contributors()' para obtener más información y
'citation()' para saber cómo citar R o paquetes de R en publicaciones.

Escriba 'demo()' para demostraciones, 'help()' para el sistema on-line
de ayuda,
o 'help.start()' para abrir el sistema de ayuda HTML con su navegador.
Escriba 'q()' para salir de R.

> #
> # Creamos una matriz A
> #
> A <- matrix(c(1,2,3,4),2,2)
> A
      [,1] [,2]
[1,]    1    3
[2,]    2    4
> #
> # La trasponemos
> #
> tA <- t(A)
> #
> # Hacemos el producto A'A
> #
> tA %*% A
      [,1] [,2]
[1,]    5   11
[2,]   11   25
> #
> # Más fácil y rápido:
> #
> crossprod(A)
      [,1] [,2]
[1,]    5   11
[2,]   11   25
>
> proc.time()
  user  system elapsed
0.436   0.020   0.447
0:%% tarea.Rout All L10 (ESS Transcript [])

```

4. Podemos añadir comentarios adicionales, si lo deseamos, en el editor o procesador de texto.

Ventajas

Simple, rápido, cada cual puede emplear el editor o procesador de texto con el que esté más familiarizado. Si se detecta un error, basta reejecutar el

paso 2 para obtener los resultados corregidos, y los comentarios incluidos en el fichero fuente `tarea.R` permanecen en su sitio. No hay que volver a cortar y pegar. Permite ejecutar largos procesos desatendidos, por ejemplo durante la noche.

Inconvenientes

La presentación es pobre.

2.3. Emacs + ESS

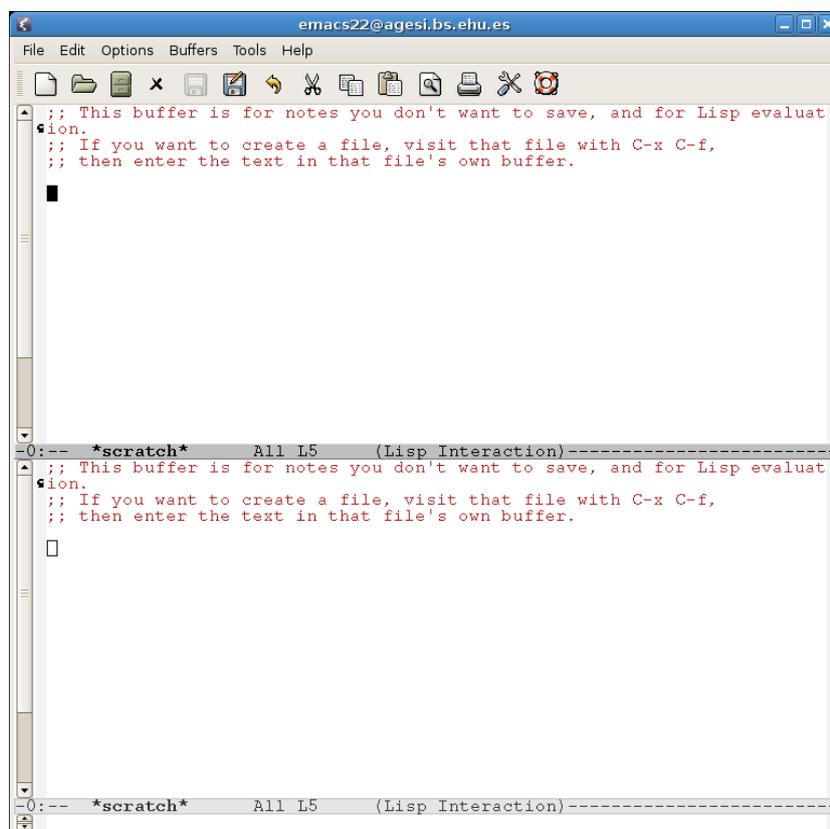
ESS (ver [The ESS Developers \(2010\)](#)) es un módulo añadido a Emacs para simplificar el trabajo con programas estadísticos: no sólo R, sino también otros que no utilizamos, como Stata, S-Plus o SAS.

Modus operandi

1. Invoca Emacs y utilizando el menú desplegable *File*→*Split Window* divide la ventana en dos partes. (Puedes emplear también el atajo de teclado `Ctrl-X 2` o `Ctrl-X 3`.) La apariencia será entonces la siguiente:
2. Sitúa el cursor en la mitad superior e inicia la edición de un fichero cuyo nombre inexcusablemente deberá acabar en R. Por ejemplo, `tarea.R` utilizado en 2.2. La apariencia será ahora:

Observa que han aparecido algunos nuevos iconos en la barra de instrumentos (han sido señalados rodeándolos con un trazo rojo). Estos iconos permiten (de izquierda a derecha) iniciar una sesión de R, iniciar una sesión de S-Plus², ejecutar una línea de código, ejecutar varias líneas de código, o ejecutar la totalidad del código en el “buffer” de edición.

²No instalado en el LEC.



3. Selecciona mediante el ratón la línea o líneas que desees procesar mediante R y teclea sobre los iconos “Ejecutar línea” o “Ejecutar región”. Si no hay una sesión de R iniciada, se iniciará en la ventana inferior y obtendrás un resultado como el de la figura a continuación.

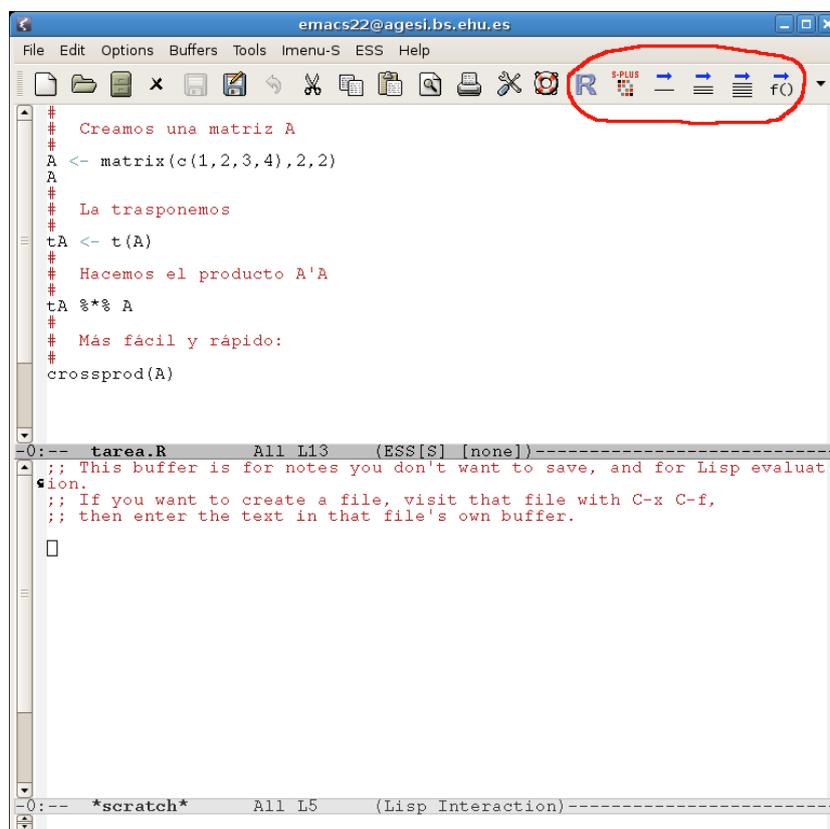
Ventajas

Puedes ver esto como una forma mejorada del método descrito en 2.2, con todas las ventajas de la interactividad y el “batch” reunidas. Te permite crear el fichero de salida de tu trabajo paso a paso, intercalando comentarios e instrucciones de una en una. Si alguna cosa resulta errónea, siempre puedes reejecutar el código desde donde sea preciso.

El fichero en la ventana inferior puede ser guardado con un nombre adecuado (no emplees el nombre por omisión *R*) e impreso.

Inconvenientes

La presentación es pobre.



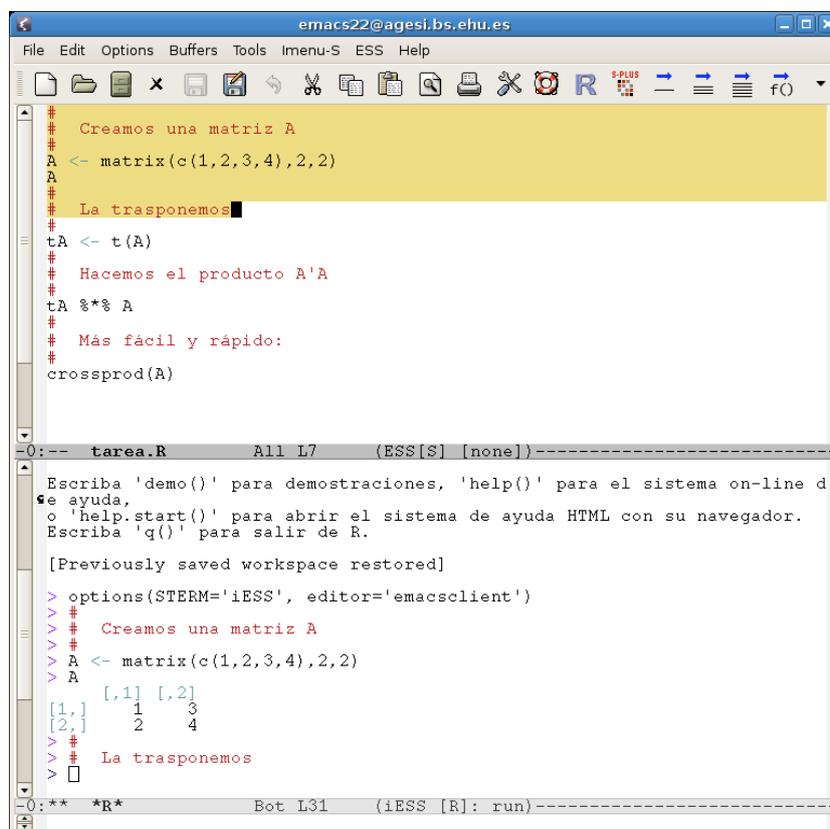
2.4. Emacs + L^AT_EX+ “corta y pega”

Este modo de trabajar puede conjugarse con el anterior, porque podemos tener un número ilimitado de ficheros abiertos en Emacs. En consecuencia, podríamos tener abierto un documento L^AT_EX en una ventana y código fuente y una sesión R en otras.

Lo que sigue presupone un conocimiento básico del formateador de texto L^AT_EX. Si no se está familiarizado con él, puede recurrirse a las fuentes de información relacionadas en el Apéndice A.

Modus operandi

1. Invoca Emacs y, si no tienes ya un fichero L^AT_EX a medio editar, crea uno nuevo.
2. Edita el texto de tu tarea. El código R y su salida pueden insertarse como texto verbatim, es decir, que no sea formateado. Tras cortar y pegar de una sesión en R (que puede estar corriendo en otro “buffer”



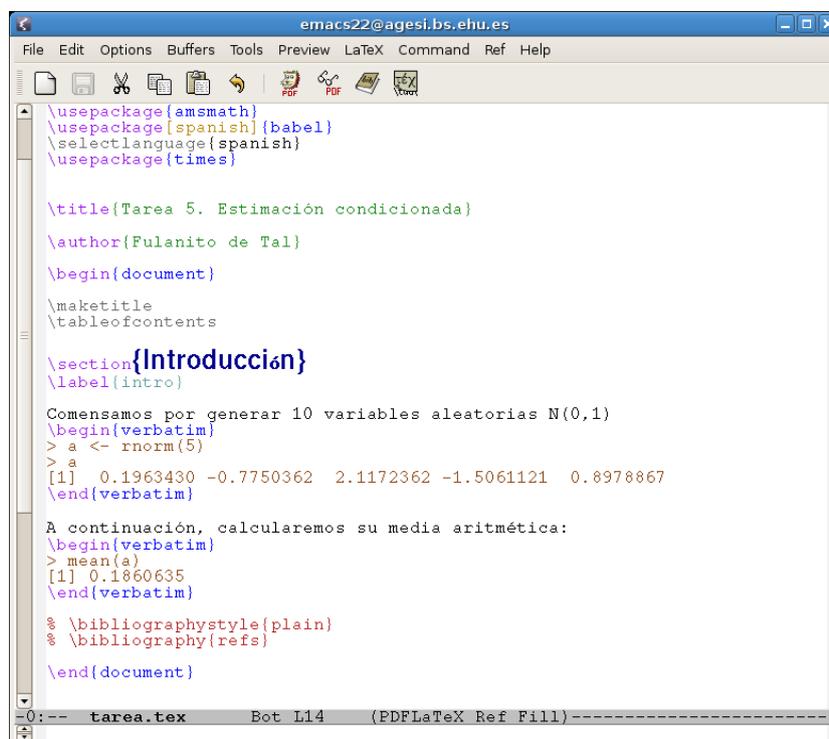
de Emacs, y servirse de ESS como se explicó en la sección 2.3), puedes obtener algo con una apariencia como la siguiente:

Observa que en la barra de tareas hay algunos iconos específicamente destinados a procesar ficheros \LaTeX . Uno de ellos (el pequeño león) procesa tu fichero fuente `tarea.tex` para generar `tarea.pdf`. Este último puede ser visualizado mediante el icono representando unas gafas³.

3. Si pinchas sobre el león observarás al cabo de un momento en la línea inferior de Emacs un mensaje tal como

LaTeX: successfully formatted {1} pages.

³Procesar en un solo paso la salida en PDF a partir de la fuente en `.tex` será lo que normalmente desees hacer con tus tareas. Para ello, has de seleccionar en Emacs *Command* → *TeX options* → *PDF mode*, y verás aparecer bajo los iconos mencionados la mención PDF en rojo. Si no haces esto, el formato de salida por omisión es el llamado `.dvi` (DeVice Independent), que puede también ser visualizado mediante el icono correspondiente, pero ha de ser transformado todavía en Postscript o un formato similar para ser impreso.



```

\usepackage{amsmath}
\usepackage[spanish]{babel}
\selectlanguage{spanish}
\usepackage{times}

\title{Tarea 5. Estimación condicionada}
\author{Fulanito de Tal}
\begin{document}
\maketitle
\tableofcontents

\section{Introducción}
\label{intro}

Comensamos por generar 10 variables aleatorias  $N(0,1)$ 
\begin{verbatim}
> a <- rnorm(5)
> a
[1] 0.1963430 -0.7750362 2.1172362 -1.5061121 0.8978867
\end{verbatim}

A continuación, calcularemos su media aritmética:
\begin{verbatim}
> mean(a)
[1] 0.1860635
\end{verbatim}

% \bibliographystyle{plain}
% \bibliography{refs}
\end{document}

```

(o un mensaje de error si algo ha ido mal).

Pinchando ahora sobre el icono de las gafas, deberías ver algo como lo que recoge la Figura 1 que es el “producto acabado” y habrá quedado en tu directorio de trabajo con el nombre `tarea.pdf`. Este es el fichero que entregarás a tu profesor, electrónicamente o sobre papel.

Ventajas

La presentación es incomparablemente mejor que la que puedes obtener mediante herramientas como Word, especialmente cuando hay notación matemática de por medio, para la que los procesadores de texto al uso están muy pobremente equipados.

Inconvenientes

Este modo de trabajar es todavía un “recorta y pega” de una sesión de R en un procesador de texto. Tiene por ello todos los inconvenientes de un recorta y pega: si algo es erróneo, tienes que volver a insertar todos los trozos de código manualmente en tu documento.

Figura 1: Ejemplo de tarea formateada.

Tarea XXX

F. Tusell

3 de noviembre de 2011

1. Introducción

Esta es la introducción.

1.1. Objetivo

El objetivo de esta tarea es bla bla bla

1.2. Resultados

Como se vio en la introducción, bla bla bla

El histograma de una distribución normal es campaniforme: ya que como se sabe la función de densidad es

$$f_X(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (1)$$

La definición (1) es válida para $-\infty < x < \infty$.

2.5. Emacs + L^AT_EX + Sweave

Sería bueno que no tuviéramos que manualmente componer nuestro documento cortando e insertando de una sesión en R. Esto es lo que permite la conjunción de las tres herramientas en el encabezamiento.

La idea es la siguiente: se inserta el código R dentro de un documento con extensión `.Rnw`, y se procesa mediante un programa que ejecuta dicho código e inserta en el lugar los resultados, dando lugar a un documento `.tex`

ordinario. Este documento se procesa mediante L^AT_EX. Todo el proceso puede automatizarse y hacerse desde dentro de Emacs con unos pocos “clics”.

Modus operandi

1. Invoca Emacs y, si no tienes ya un fichero L^AT_EX a medio editar, crea uno nuevo. **Guárdalo con nombre acabado en .Rnw**; algo como `tarea.Rnw`.
2. Inserta, si no lo está ya en la plantilla que utilizas, la línea

```
\usepackage{/usr/local/lib/R/share/texmf/Sweave}
```

3. Edita el texto de tu tarea *insertando en ella el código R flanqueado de marcadores <<>= y @*. Así por ejemplo podrías editar un fichero `tarea.Rnw` de la siguiente apariencia:

```

\documentclass[a4paper,12pt]{article}
\usepackage[T1]{fontenc}
\usepackage[latin1]{inputenc}
\usepackage{graphicx}
\usepackage{url}
\usepackage{amsmath}
\usepackage[spanish]{babel}
\selectlanguage{spanish}
\usepackage{times}
\usepackage{/usr/local/lib/R/share/texmf/Sweave}

\title{Tarea 5. Estimación condicionada}

\author{Fulanito de Tal}

\begin{document}

\maketitle
\tableofcontents

\section{Introducción}
\label{intro}

Comensamos por generar 10 variables aleatorias N(0,1)
<<>=
a <- rnorm(5)
a
@

A continuación, calcularemos su media aritmética:
<<>=
mean(a)
% \bibliographystyle{plain}
% \bibliography{refs}

\end{document}

```

4. *No* puedes hacer ahora uso de los iconos representando el león y las gafas, porque lo que estás editando no es un documento `.tex` sino un precursor. En lugar de ello, tienes algunos atajos de teclado:

Cuadro 1: Proceso en Emacs de documentos `.Rnw`

ATAJO	FUNCIÓN
ESC-n s	Ejecutar Sweave sobre el documento <code>.Rnw</code>
ESC-n l	Ejecutar \LaTeX sobre el documento <code>.tex</code> generado a partir del <code>.Rnw</code>
ESC-n d	Visualizar el resultado en formato DVI
ESC-n p	Visualizar el resultado en Postscript
ESC-n P	Formatear directamente a PDF
ESC-n b	Ejecutar Bib \TeX para resolver referencias bibliográficas.

En el caso que nos ocupa, emplearías en sucesión los mandatos ESC-n s, ESC-n l y ESC-n P en sucesión. ¡El ciclo edición-formateo-visualización requiere sólo nueve golpes de tecla!

Ventajas

No hay que recortar y pegar nada, porque todo queda ya en su sitio. Adicionalmente, el código y el texto que lo comenta están en un mismo documento, vecinos uno del otro, lo que facilita saber qué se hizo al volver transcurrido un tiempo sobre el documento: un fichero `.Rnw` es código auto-documentado.

La presentación es la misma que si hubiéramos editado un documento \LaTeX artesanalmente.

Se puede hacer que el documento final muestre sólo el código R, el código R y su resultado⁴, o sólo el resultado⁵.

Inconvenientes

Se requiere un cierto trabajo para integrar todas las herramientas que intervienen: este trabajo ha sido hecho ya por tí en el LEC.

⁴Que es lo que tu profesor quiere ver.

⁵Que es probablemente lo que querrías si empleases esta herramienta en un entorno empresarial, en que tu lector no está interesado en cómo has procesado la información.

2.6. Uso de plantillas

Si desde Emacs visitas⁶ (File → Visit new file) un nuevo fichero de nombre `tarea.Rnw` o `tarea.tex`, verás que —aunque no exista nada de tal nombre en tu directorio de trabajo— se te ofrece una plantilla con los paquetes más habituales precargados, tu nombre señalado como autor, etc. Es la manera aconsejada de comenzar una nueva tarea.

Nota que para trabajar con una plantilla así cargada *hay que previamente guardarla en un fichero con otro nombre distinto*.

2.7. ¿No hay herramientas “what you see is what you get”?

2.7.1. El modo Preview de Emacs, TeXmacs, y Lyx

Sí, las tienes. Puedes utilizar editores especializados como `texmacs` o `lyx`. No obstante, la disponibilidad de pantallas grandes en el LEC hace que sea posible trabajar con el fichero fuente `.tex` o `.Rnw` en Emacs y al mismo tiempo visualizar el resultado. Se tienen así las ventajas reunidas del proceso integrado y la inmediatez en la visualización.

También desde dentro de Emacs tienes un modo de previsualización instantáneo. Si editas un documento L^AT_EX (pero no `.Rnw`) y pinchas el icono “Preview” (rodeado con trazo rojo en la figura a continuación), verás los resultados dentro de Emacs en su lugar.

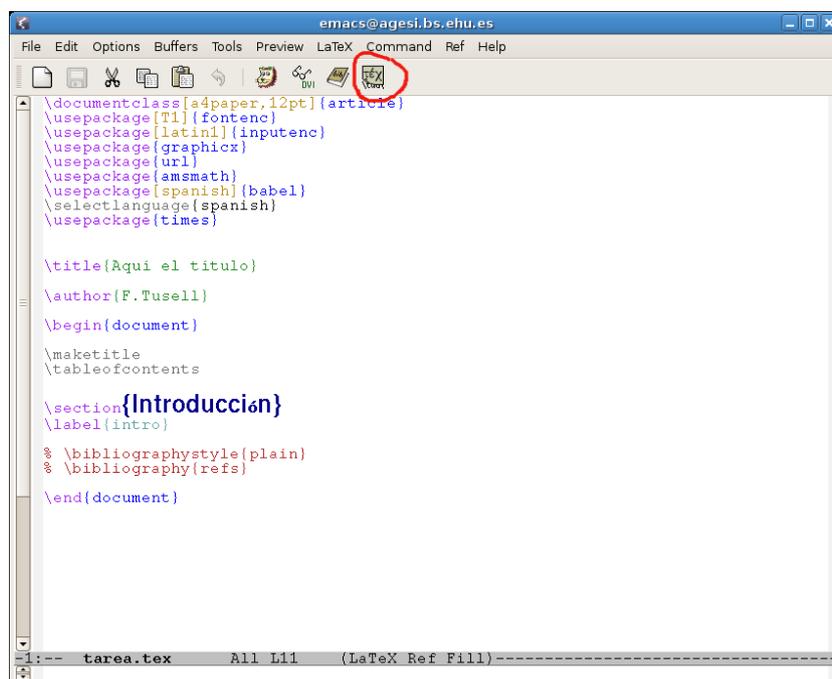
Dado que la mayor parte de tu trabajo requiere integrar código R, resultados y texto, el método descrito en la sección 2.5 es el que recomendamos. No te confundas: “what you see is what you get” significa la mayor parte del tiempo “what you see is *all* you get”. Y *all* no suele ser gran cosa: a medio plazo te limitará.

2.7.2. R Commander

Por las mismas razones que acabamos de exponer, tampoco te aconsejamos que uses ninguna de las capas que pretenden interponer un menú “point and click” entre el usuario y R: a medio plazo las encontrarás muy limitadoras. No obstante, si quieres utilizarlas, una de las más populares es R Commander.

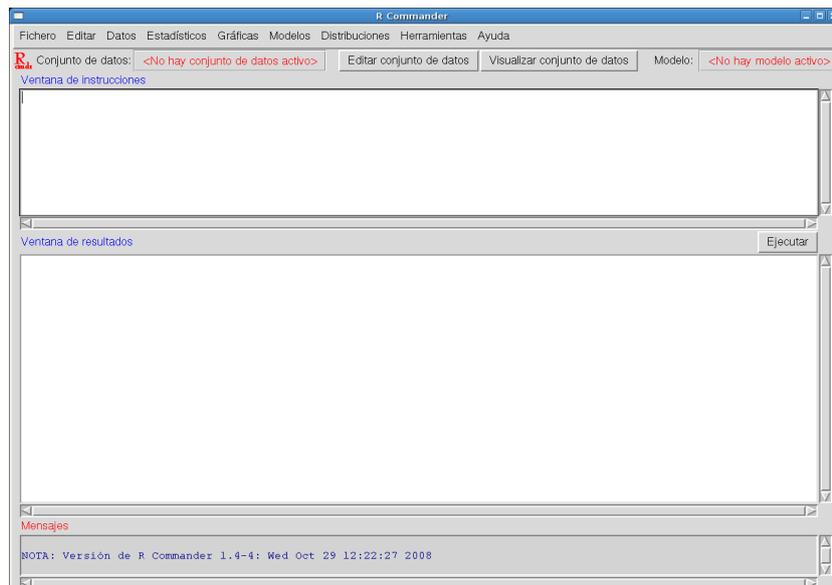
1. Inicia una sesión en R, y teclea:

⁶Esto funciona en el LEC, pero no es algo “standard”. Requiere la definición de las plantillas. Si quisieras replicar esta funcionalidad en una máquina de tu propiedad, mira <http://emacs-template.sourceforge.net/> o pide ayuda.



`library(Rcmdr)`

Se cargará el módulo `Rcmdr` y sus dependencias (todo está preinstalado en el LEC) y verás una ventana con la apariencia:



2. A partir de aquí, puedes hacer bastantes cosas sin más que desplegar menús y pinchar en las opciones que se te ofrecen.

Una de las ventajas de **Rcmdr** es que, como respuesta a las acciones seleccionadas mediante menús, se genera (y ejecuta) el código **R** pertinente. Ello permite aprender de manera fácil la manera de hacer directamente las cosas. Pueden además guardarse por separado los ficheros de instrucciones y de resultados: es una funcionalidad en cierto modo similar a la de Emacs + ESS.

2.7.3. RStudio

Si prefieres un entorno visual y estás dispuesto a prescindir de la ayuda que Emacs supone para la realización de tareas, puedes servirte de RStudio (mira en <http://rstudio.org/>). No está instalado en el LEC. Es probablemente lo más sencillo de instalar en una máquina Windows y ofrece buena funcionalidad.

A. Información sobre L^AT_EX.

Hay mucha bibliografía. La referencia básica es [Lamport \(1994\)](#). Un excelente libro en español es [Cascales et al. \(2003\)](#). Obras de interés son [Grätzer \(2007\)](#), [Kopka and Daly \(1995\)](#), [Goossens et al. \(1994\)](#) y [Goossens et al. \(1997\)](#), centrada esta última en la integración de gráficos con L^AT_EX.

Otras obras parcialmente obsoletas, pero todavía de interés, incluyen [Hahn \(1991\)](#) y [Walsh \(1994\)](#). Por supuesto, siendo L^AT_EX una capa construida sobre T_EX, todo lo que hace referencia a éste es de interés. Pero no es preciso conocer T_EX en detalle para utilizar productivamente L^AT_EX. Con todo, si se desea se puede mirar [Abrahams et al. \(1994\)](#), [Schwartz \(1992\)](#) y la monumental obra que lo originó todo, [Knuth \(1986\)](#)

Hay multitud de información en la red. Buenos puntos de entrada son la Wikipedia, <http://es.wikipedia.org/wiki/LaTeX>, la página oficial del proyecto L^AT_EX, <http://www.latex-project.org/>, el manual introductorio (en castellano) http://es.wikibooks.org/wiki/Manual_de_LaTeX o la página del COMPREHENSIVE T_EX ARCHIVE NETWORK, <http://www.ctan.org/>, que contiene todos los materiales imaginables sobre T_EX y L^AT_EX.

B. Información sobre Emacs.

Sobre Emacs pueden consultarse [Stallman \(1997\)](#) y [Cameron and Rosenblatt \(1991\)](#). A un nivel más avanzado, sólo de interés si deseas extender Emacs o integrar otras herramientas con él, [Glickstein \(1997\)](#) y [Chassell \(1995\)](#).

En la red hay también abundante material y, como para el caso de L^AT_EX, la Wikipedia, <http://es.wikipedia.org/wiki/Emacs>, suministra un buen punto de entrada.

Mucha de la funcionalidad de Emacs para procesar documentos en T_EX y L^AT_EX proviene del módulo AUCT_EX, sobre el que puedes visitar <http://gnu.manticore.es/paginas-replica/auctex/index.html> o <http://en.wikipedia.org/wiki/AUCTeX>. La gestión de referencias bibliográficas reposa sobre el módulo RefT_EX, <http://www.gnu.org/software/auctex/reftex.html>.

Sobre el software ESS, otro de los módulos añadidos a Emacs que permite el procesamiento fácil de código R, puedes visitar <http://ess.r-project.org/>. Sobre Sweave puedes consultar [Leisch \(2002\)](#).

C. ¿Puedo instalar esto en mi casa?

Desde luego, todo lo que se ha empleado en las Secciones precedentes es software libre, y lo hay disponible sobre Unix/Linux, sobre Windows y sobre Mac⁷. Puedes replicar una instalación como la del LEC en tu máquina personal, aunque si lo haces sobre Windows la integración de todas las “piezas” puede ser un poco más tediosa.

Si, pese a ello, decides instalarlo en Windows, un buen punto de arranque puede ser la oferta de Vincent Goulet, <http://vgoulet.act.ulaval.ca/>, de la que tienes copia local (quizá no la más reciente) en el LEC.

⁷Esto último no lo he probado.

Referencias

- P. A. Abrahams, K. Berry, and K. A. Hargreaves. *TEX for the Impatient*. Addison-Wesley, 1994.
- J. S. Botella. *Iniciación a L^AT_EX₂ ϵ : un Sistema para Preparar Documentos*. Addison-Wesley Iberoamericana, Madrid, 1997.
- D. Cameron and B. Rosenblatt. *Learning GNU Emacs*. O'Reilly and Assoc., 1991.
- B. Cascales, P. Lucas, J. Mira, S. Sánchez-Pedreño, and A. Pallares. *El libro de L^AT_EX*. Pearson/Prentice Hall, 2003.
- R. J. Chassell. *Programming in Emacs Lisp*. Free Software Foundation, 1995.
- B. Glickstein. *Writing GNU Emacs Extensions*. O'Reilly and Assoc., 1997.
- M. Goossens, F. Mittelbach, and A. Samarin. *The L^AT_EX Companion*. Addison-Wesley, 1994.
- M. Goossens, S. Rahtz, and F. Mittelbach. *The L^AT_EX Graphics Companion*. Addison-Wesley, 1997.
- G. Grätzer. *More Math into L^AT_EX*. Springer, fourth edition, 2007.
- J. Hahn. *L^AT_EX for Everyone*. Personal T_EX, Inc., second edition, 1991.
- D. Knuth. *The T_EX Book*. Addison Wesley, Reading, Mass., 1986.
- H. Kopka and P. W. Daly. *A Guide to L^AT_EX₂ ϵ* . Addison-Wesley, second edition, 1995.
- L. Lamport. *A Document Preparation System. L^AT_EX User's Guide and Reference Manual*. Addison-Wesley, 1994.
- F. Leisch. Sweave, part I: Mixing R and L^AT_EX. *R News*, 2(3):28–31, December 2002. URL <http://CRAN.R-project.org/doc/Rnews/>.
- N. Schwartz. *Introduction to T_EX*. Addison-Wesley, 1992.
- R. M. Stallman. *GNU Emacs Manual*. Free Software Foundation, 1997.
- The ESS Developers. *ESS - Emacs Speaks Statistics*, version 5.15 edition, 2010. URL <http://stat.ethz.ch/ESS/>.
- N. Walsh. *Making T_EX Work*. O'Reilly and Assoc., 1994.