The package Rnews

Kurt Hornik Friedrich Leisch

February 5, 2006

1 Introduction

The LaTeX 2_ε package Rnews provides commands for formatting the R Newsletter.

2 Documentation

2.1 Marking Words and Phrases

The Rnews package provides roughly the same commands for marking words and phrases as does Texinfo (but note that the LaTeXspecial characters still need special treatment). These commands are

`\code`{*sample-code*} Indicate text that is a literal example of a piece of a program.

`\kbd`{*keyboard-characters*} Indicate keyboard input.

`\key`{*key-name*} Indicate the conventional name for a key on a keyboard.

`\samp`{*text*} Indicate text that is a literal example of a sequence of characters.

`\var`{*metasyntactic-variable*} Indicate a metasyntactic variable.

`\env`{*environment-variable*} Indicate an environment variable.

`\file`{*file-name*} Indicate the name of a file.

`\command`{*command-name*} Indicate a command name (such as '`ls`').

`\option`{*option-name*} Indicate a command line option.

`\dfn`{*term*} Indicate the introductory or defining use of a term.

`\acronym`{*acronym*} Use for abbreviattions written in all capital letters, such as '`NASA`'.

If this sounds rather confusing, please see the Texinfo documentation for more details.

`\strong` There is also a `\strong` command for emphasizing text more strongly than with `\emph`. For example, `\strong{Note:}` gives **Note:**.

`\pkg` Finally, use `\pkg` for indicating R packages.

2.2 Quotations and Examples

In addition to the standard LaTeX for quotations and examples (such as `quote`, `quotation`, `flushleft`, `center` and `flushright`), the Rnews package provides the following environments.

`example` Illustrate code, commands, and the like. The text is printed in a fixed-width font, and indented but not filled.

`smallexample` Similar to `example`, except that text is typeset in a smaller font.

These are patterned after the Texinfo environments with the same names. In particular, `{`, `}`, `\` retain their "usual" meanings and are not treated verbatim, which is not optimal for displaying R code or output. Hence, we also provide a `smallverbatim` environment which works like `verbatim` but uses a smaller font for typesetting.

2.3 Mathematics

`\P` `\E` `\VAR` `\COV` `\COR`

The commands `\P`, `\E`, `\VAR`, `\COV`, and `\COR` produce symbols for probability, expectation, variance, covariance and correlation. For example, Chebyshev's inequality

$$\mathbb{P}(|\xi - \mathbb{E}\,\xi| > \lambda) \le \frac{\mathrm{var}(\xi)}{\lambda^2}.$$

can be coded as

```
\P(|\xi-\E\xi|>\lambda) \le \frac{\VAR(\xi)}{\lambda^2}.
```

`\mathbb` The symbols

$$\mathbb{N} \quad \mathbb{Z} \quad \mathbb{Q} \quad \mathbb{R} \quad \mathbb{C}$$

for the positive integers, the integers, and the rational, real and complex numbers, respectively, can be obtained using `\mathbb` from package amsfonts as

```
\mathbb{N} \mathbb{Z} \mathbb{Q} \mathbb{R} \mathbb{C}
```

3 The Code

3.1 The Batch File

First comes the code for creating the batch file 'Rnews.ins' which in turn can be used for producing the package and driver files.

```
1 ⟨*install⟩
2 \begin{filecontents}{\filename.ins}
3 % Simply TeX or LaTeX this file to extract various files from the source
4 % file 'Rnews.dtx'.
5   \def\filedate{2001/01/05} \def\batchfile{Rnews.ins} \input
```

```
  docstrip.tex \preamble
\endpreamble
\generateFile{Rnews.drv}{t}{\from{Rnews.dtx}{driver}}
\generateFile{Rnews.sty}{t}{\from{Rnews.dtx}{package}}
\Msg{************************************************************}
\Msg{* For documentation, run LaTeX on Rnews.dtx or Rnews.drv. *}
\Msg{************************************************************}
\end{filecontents}
⟨/install⟩
```

3.2 The Driver

Next comes the documentation driver file for TEX, i.e., the file that will produce the documentation you are currently reading. It will be extracted from this file by the `docstrip` program. Since it is the first code in the file one can alternatively process this file directly with LATEX 2_ε to obtain the documentation.

```
⟨*driver⟩
\documentclass[fleqn]{ltxdoc}
\usepackage[driver]{\filename}
\renewcommand{\pkg}[1]{\textsf{#1}}
\begin{document}
  \DocInput{\filename.dtx}
\end{document}
⟨/driver⟩
```

3.3 The Code

Now comes the code for the package.

It the current format is not LATEX 2_ε, we abort immediately. Otherwise, we provide ourselves and show the current version of the package on the screen and in the transscript file.

```
⟨*package⟩
\NeedsTeXFormat{LaTeX2e}[1995/12/01]
\ProvidesPackage{\filename}[\filedate\space\fileversion\space
                           Rnews package]
\typeout{Package: `\filename\space\fileversion \@spaces <\filedate>'}
\typeout{English documentation as of <\docdate>}
```

Next, we set up a more or less trivial option handler. We use option 'driver' for conditionalizing package code we do not want executed when typesetting the driver file.

```
\RequirePackage{ifthen}
\newboolean{Rnews@driver}
\DeclareOption{driver}{\setboolean{Rnews@driver}{true}}
\DeclareOption*{\PackageWarning{\filename}{Unknown option
    `\CurrentOption'}}
\ProcessOptions\relax
```

Now comes the real code.

```
\ifthenelse{\boolean{Rnews@driver}}{}{
```

First we load some utility packages.

```
\RequirePackage{multicol,graphicx,color,fancyhdr,hyperref}
```

3.3.1 Basic Structure

Issues of of *R News* are created from the standard LaTeX document class report. Individual articles correspond to chapters, and are contained in `article` environments. This makes it easy to have figures counted within articles and hence hyperlinked correctly.

Basic front matter information about the issue: volume, number, and date.

```
\newcommand{\volume}[1]{\def\Rnews@volume{#1}}
\newcommand{\volnumber}[1]{\def\Rnews@number{#1}}
\renewcommand{\date}[1]{\def\Rnews@date{#1}}
```

We do not want numbered sections.

```
\setcounter{secnumdepth}{-1}
```

`\author` `\title` `\subtitle`

An article has an author, a title, and optionally a subtitle. We use the obvious commands for specifying these.

```
\renewcommand{\author}[1]{\def\Rnews@author{#1}}
\renewcommand{\title}[1]{\def\Rnews@title{#1}}
\newcommand{\subtitle}[1]{\def\Rnews@subtitle{#1}}
```

Note that we put the title info in the TOC and the bookmarks when creating PDF. Thus titles should really only contain text.

`article`

Environment `article` clears the article header information its begin and restores single column mode at its end.

```
\newenvironment{article}{%
  \author{}\title{}\subtitle{}}{\end{multicols}}
```

`\maketitle`

The real work is done by a redefined version of `\maketitle`, which also switches to double column mode. Note that even though we do not want chapters (articles) numbered, we need to increment the chapter counter, so that figures get correct labelling.

```
\renewcommand{\maketitle}{
  \begin{multicols}{2}[\chapter{\Rnews@title}\refstepcounter{chapter}][3cm]
    \ifx\empty\Rnews@subtitle\else\noindent\textbf{\Rnews@subtitle}
    \par\nobreak\addvspace{\baselineskip}\fi
    \ifx\empty\Rnews@author\else\noindent\textit{\Rnews@author}
    \par\nobreak\addvspace{\baselineskip}\fi
    \@afterindentfalse\@nobreaktrue\@afterheading}
```

Now for some ugly redefinitions. We do not want articles to start a new page.

```
\renewcommand\chapter{\secdef\Rnews@chapter\@schapter}
\providecommand{\nohyphens}{%
  \hyphenpenalty=10000\exhyphenpenalty=10000\relax}
\newcommand{\Rnews@chapter}{%
  \renewcommand{\@seccntformat}[1]{}%
  \@startsection{chapter}{0}{0mm}{%
    -2\baselineskip \@plus -\baselineskip \@minus -.2ex}{\p@}{%
    \normalfont\Huge\bfseries\raggedright}}
```

TOC entries for articles (chapters) should really look like sections.

```
\renewcommand*\l@chapter{\@dottedtocline{0}{0pt}{1em}}
```

We want bibliographies as starred sections within articles. As the standard `thebibliography` environment uses `\chapter*`, we simply redefine the latter according to our needs.

```
\def\@schapter#1{\section*#1}
```

Package multicol, which is used for producing two-column output, only allows for starred (single-column) floats (figures and tables). Therefore, we provide a simple non-floating `figure` environment ourselves.

```
\renewenvironment{figure}[1][]{%
  \def\@captype{figure}
  \noindent
  \begin{minipage}{\columnwidth}}{%
  \end{minipage}\par\addvspace{\baselineskip}}
```

Equations, figures and tables are counted within articles, but we do not show the article number. Unfortunately, for equations this means that we need to redefine `equation` as otherwise hyperref computes the equation hyperlinks (via `\make@stripped@name{\theequation}`) as `equation.\theequation`.

```
\renewcommand{\theequation}{\@arabic\c@equation}
\def\equation{%
  \let\refstepcounter\H@refstepcounter
  \H@equation
  \def\newname{\arabic{chapter}.\theequation}%
  \let\theHequation\newname%
  \hyper@makecurrent{equation}%
  \Hy@raisedlink{\hyper@anchorstart{\@currentHref}}%
  \let\refstepcounter\new@refstepcounter}%
\def\endequation{\Hy@raisedlink{\hyper@anchorend}\H@endequation}
\renewcommand{\thefigure}{\@arabic\c@figure}
\renewcommand{\thetable}{\@arabic\c@table}
```

`\tableofcontents` Need to provide our own version of `\tableofcontents` (no fiddling with the number of columns). Note that `\section*` is really the same as `\chapter*`).

```
\renewcommand{\contentsname}{Contents of this issue:}
\renewcommand\tableofcontents{%
  \section*{\contentsname
    \@mkboth{%
```

```
      \MakeUppercase\contentsname}{\MakeUppercase\contentsname}}%
  \@starttoc{toc}}
```

`\titlepage` The title page of each issue features logo et al at the top and the TOC. We start with the top.

```
\renewcommand{\titlepage}{%
  \noindent
  \rule{\textwidth}{1pt}\\[-.8\baselineskip]
  \rule{\textwidth}{.5pt}
  \begin{center}
    \includegraphics[height=2cm]{Rlogo}\hspace{7mm}
    \fontsize{2cm}{2cm}\selectfont
    News
  \end{center}
  The Newsletter of the R Project\hfill
  Volume \Rnews@volume/\Rnews@number, \Rnews@date\\[-.5\baselineskip]
  \rule{\textwidth}{.5pt}\\[-.8\baselineskip]
  \rule{\textwidth}{1pt}
  \vspace{1cm}
```

Now set up the header and footer information for the rest of the document.

```
  \fancyhf{}
  \fancyhead[L]{Vol.~\Rnews@volume/\Rnews@number, \Rnews@date}
  \fancyhead[R]{\thepage}
  \fancyfoot[L]{R News}
  \fancyfoot[R]{ISSN 1609-3631}
  \thispagestyle{empty}
```

And finally, put the TOC at the bottom in a framed box. Note the way `tocdepth` is adjusted before and after producing the TOC: thus, we can ensure that only articles show up in the printed TOC, but that in the PDF version, bookmarks are created for sections and subsections as well (provided that the non-starred forms are used).

```
  \begin{bottombox}
    \begin{multicols}{2}
      \setcounter{tocdepth}{0}
      \tableofcontents
      \setcounter{tocdepth}{2}
    \end{multicols}
  \end{bottombox}}
```

3.3.2 Layout, Fonts and Color

Layout. We set the basic layout parameters in a way that printouts should be fine for both A4 and Letter paper.

```
\setlength{\textheight}{250mm}
\setlength{\topmargin}{-10mm}
\setlength{\textwidth}{17cm}
\setlength{\oddsidemargin}{-6mm}
\setlength{\columnseprule}{.1pt}
```

```
\setlength{\columnsep}{20pt}
```

Fonts. We use the following fonts (all with T1 encoding):

rm	palatino
tt	almost european (computer modern working with T1) Reason for aett: uses less horizontal space than courier, which is better for example code
sf	almost european
math	palatino

```
\RequirePackage{ae,mathpple}
\RequirePackage[T1]{fontenc}
\renewcommand{\rmdefault}{ppl}
\renewcommand{\sfdefault}{aess}
\renewcommand{\ttdefault}{aett}
```

Colors. These are actually used for `\hypersetup` but we do not call this here, although we should.

FIXME

```
\definecolor{Red}{rgb}{0.7,0,0}
\definecolor{Blue}{rgb}{0,0,0.8}
\definecolor{hellgrau}{rgb}{0.55,0.55,0.55}
```

3.3.3 Miscellania

```
\newcommand{\R}{R}
\newcommand{\address}[1]{\addvspace{\baselineskip}\noindent\emph{#1}}
\newcommand{\email}[1]{\href{mailto:#1}{\normalfont\texttt{#1}}}
```

`bottombox` Used for creating the TOC and the back matter editorial information.

```
\newsavebox{\Rnews@box}
\newlength{\Rnews@len}
\newenvironment{bottombox}{%
  \begin{figure*}[b]
    \begin{center}
      \noindent
      \begin{lrbox}{\Rnews@box}
        \begin{minipage}{0.99\textwidth}}{%
        \end{minipage}
      \end{lrbox}
      \addtolength{\Rnews@len}{\fboxsep}
      \addtolength{\Rnews@len}{\fboxrule}
      \hspace*{-\Rnews@len}\fbox{\usebox{\Rnews@box}}
    \end{center}
  \end{figure*}}
```

`boxedverbatim` Used for creating a boxed (small) verbatim environment, e.g., when showing output that is too wide for displaying in two-column mode. The code is taken from package moreverb. Note that we need to use `verbatim` rather than `alltt`.

```
\RequirePackage{verbatim}
\def\boxedverbatim{%
  \def\verbatim@processline{%
    {\setbox0=\hbox{\the\verbatim@line}%
    \hsize=\wd0 \the\verbatim@line\par}}%
  \@minipagetrue
  \@tempswatrue
  \setbox0=\vbox
  \bgroup\small\verbatim
}
\def\endboxedverbatim{%
  \endverbatim
  \unskip\setbox0=\lastbox
  \egroup
  \fbox{\box0}
}
```

Finally, we turn on fancy page style.

```
\pagestyle{fancy}
} % \ifthenelse{\boolean{Rnews@driver}}
```

3.3.4 Marking Words and Phrases

Simple font selection is not good enough. For example, `\texttt{--}` gives '--', i.e., an endash in typewriter font. Hence, we need to turn off ligatures, which currently only happens for commands `\code` and `\samp` and the ones derived from them. Hyphenation is another issue; it should really be turned off inside `\samp`. And most importantly, LaTeX special characters are a nightmare. E.g., one needs `\~{}` to produce a tilde in a file name marked by `\file`. Perhaps a few years ago, most users would have agreed that this may be unfortunate but should not be changed to ensure consistency. But with the advent of the WWW and the need for getting '~' and '#' into URLs, commands which only treat the escape and grouping characters specially have gained acceptance (in fact, this is also what alltt does, and hence environments based on it such as our `smallexample`). Hence, in the long run we should implement the same for `\code`, `\kbd`, `\samp`, `\var`, and `\file`. (The other Texinfo-style commands do not need this.)

```
%\newcommand\code{\bgroup\@noligs\@codex}
\newcommand\code{\bgroup\@codex}
\def\@codex#1{{\normalfont\ttfamily\hyphenchar\font=-1 #1}\egroup}
\newcommand{\kbd}[1]{{\normalfont\texttt{#1}}}
\newcommand{\key}[1]{{\normalfont\texttt{\uppercase{#1}}}}
\newcommand\samp{`\bgroup\@noligs\@sampx}
\def\@sampx#1{{\normalfont\texttt{#1}}\egroup'}
\newcommand{\var}[1]{{\normalfont\textsl{#1}}}
\let\env=\code
\newcommand{\file}[1]{{`\normalfont\textsf{#1}'}}
\let\command=\code
\let\option=\samp
```

```
\newcommand{\dfn}[1]{{\normalfont\textsl{#1}}}
\newcommand{\acronym}[1]{{\normalfont\textsc{\lowercase{#1}}}}
\newcommand{\strong}[1]{{\normalfont\fontseries{b}\selectfont #1}}
\let\pkg=\strong
```

3.3.5 Quotations and Examples

```
\RequirePackage{alltt}
\newenvironment{example}{\begin{alltt}}{\end{alltt}}
\newenvironment{smallexample}{\begin{alltt}\small}{\end{alltt}}
\newenvironment{display}{\list{}{}\item\relax}{\endlist}
\newenvironment{smallverbatim}{\small\verbatim}{\endverbatim}
```

3.3.6 Mathematics

`\operatorname` The implementation of `\operatorname` is similar to the mechanism LaTeX 2$_\varepsilon$ uses for functions like sin and cos, and simpler than the one of $\mathcal{AMS}$-LaTeX. We use `\providecommand` for the definition in order to keep the one of the amstex if this package has already been loaded.

```
\providecommand{\operatorname}[1]{%
  \mathop{\operator@font#1}\nolimits}
```

`\P` `\E` `\VAR` `\COV` `\COR` Next, we provide commands for probability, expectation, variance, covariance and correlation which are obviously useful in probability theory and statistics. (Of course, originally `\P` gives ¶.)

```
\renewcommand{\P}{%
  \mathop{\operator@font I\hspace{-1.5pt}P\hspace{.13pt}}}
\newcommand{\E}{%
  \mathop{\operator@font I\hspace{-1.5pt}E\hspace{.13pt}}}
\newcommand{\VAR}{\operatorname{var}}
\newcommand{\COV}{\operatorname{cov}}
\newcommand{\COR}{\operatorname{cor}}
```

Finally, we load package amsfonts so that `\mathbb` is available for producing the symbols for positive integers etc.

```
\RequirePackage{amsfonts}
```

This ends the implementation of the Rnews package.

```
⟨/package⟩
```