

Styling R `ggplot2` graphics with \LaTeX

Travis Stenberg

Abstract

The `ggplot2` package is widely used for R graphics. Example \LaTeX -style rendering of such graphics is presented, achieved via annotations with embedded \LaTeX markup. This allows R graphics to be integrated into \LaTeX documents with a harmonious visual style.

1 Technology stack

R is a statistical programming language. The material presented here was implemented using R 4.2.3, using the RStudio integrated development environment, on Windows 11.

In the same way that \LaTeX is enhanced by supporting packages, R is also. Key R packages for integrating \LaTeX into R graphics were `extrafont`, `fontcm`, `ggplot2` and `latex2exp`.

The Ghostscript interpreter was used to embed default \LaTeX fonts into PDF files. R, RStudio and Ghostscript are all free, and enjoy multiplatform support. Examples of these technologies applied to other \TeX -related issues appear elsewhere in *TUGboat* [1, 4, 5].

2 Ghostscript setup

Ghostscript typically needs manual configuration on Windows, summarized here. First, ensure it's in your operating system's path, e.g.:

```
C:\Program Files\gs\gs10.00.0\bin
```

Also add the environment variables `GS_CMD`, identifying the Ghostscript executable, e.g.:

```
C:\Program Files\gs\gs10.00.0\bin\
  gswin64c.exe
```

and `GS_FONTPATH`, designating any font folders to use, e.g.:

```
C:\Windows\Fonts\
C:\Users\Travis\AppData\Local\Microsoft\
  Windows\Fonts\
```

An R application-level environment variable for Ghostscript's executable should be set too. Find your local Rprofile document (usually somewhere under the R installation directory) and append the new setting, e.g.:

```
Sys.setenv(R_GSCMD = "C:/Program Files/gs/
  gs10.00.0/bin/gswin64c.exe")
```

3 Computer Modern fonts in R

The `ggplot2` package is designed especially for plots in R. A `ggplot2` object is instantiated, associated with data, and its display properties specified programmatically.

Font properties of `ggplot2` objects can be set in R to emulate \LaTeX . A Computer Modern default font is assumed to be present [2].

```
ggplot2::theme(
  text = ggplot2::element_text(
    family = "CM Roman", size = 10
  )
)
```

To style such plots with Computer Modern fonts, a one-time R call installs them:

```
extrafont::font_install("fontcm")
```

To load the fonts in subsequent R sessions, make a relevant `extrafont` call at least once per session:

```
extrafont::loadfonts(quiet = TRUE)
```

4 Styling `ggplot2`

\LaTeX -styled strings can be emulated in plots via the `latex2exp` package. Only a \LaTeX subset is supported, enumerable in R.

```
latex2exp::latex2exp_supported()
```

The package accommodates common features such as math mode (with escaped backslashes), or inline Unicode.

```
latex2exp::TeX("weight \\textbf{W}_1$")
latex2exp::TeX("high\U00ADtech")
```

5 PDF rendering

PDF rendering is invoked via `grDevices` (loaded by default in R). Fonts were embedded in the output file via `extrafont` and Ghostscript. Finally, superfluous bounding whitespace can be cropped via `knitr`.

Example R code to style `ggplot2` with \LaTeX , showing the results of Markov Chain Monte Carlo convergence testing for a mixture model, as per e.g., [3], is given below. The resulting PDF appears in Figure 1.

```

# Data setup.
library(latex2exp)
labels <- c(
  TeX("$\\eta_2$"), TeX("$\\eta_1$"),
  TeX("$\\sigma_2$"), TeX("$\\sigma_1$"),
  TeX("$\\mu_2$"), TeX("$\\mu_1$"))
df <- data.frame(
  x = c(3040, 3040, 3458,
        3392, 2758, 4176),
  y = sapply(labels, deparse))

# Fonts and PDF driver setup.
extrafont::loadfonts(quiet = TRUE)
file_name <- "plot_example.pdf"
pdf(file_name)

# Build plot.
library(ggplot2)
grid_line <- element_line(
  linewidth = 0.25, linetype = "dashed",
  color = "grey")
plot <- ggplot(df, aes(x = x, y = y)) +
  geom_point() +
  scale_y_discrete(labels = labels) +
  theme_bw() + theme(
  axis.text = element_text(
    color = "black"),
  panel.grid.major = grid_line,
  panel.grid.minor = grid_line,
  text = element_text(
    family = "CM Roman", size = 10)) +
  xlab(TeX(
    "bulk\\U00AD{}$\\textit{n}_{eff}$")) +
  ylab("Mixture parameter")

# Set plot size.
gridExtra::grid.arrange(
  grobs = lapply(list(plot),
    egg::set_panel_size,
    width = grid::unit(42, "mm"),
    height = grid::unit(42, "mm")))

# Close extraneous graphics devices.
while (!is.null(dev.list())) {
  device_num <- as.integer(dev.cur())
  if (device_num != 1) {
    dev.off(which = device_num)}
}

# Finalise plot.
extrafont::embed_fonts(file_name)
knitr::plot_crop(file_name)

```

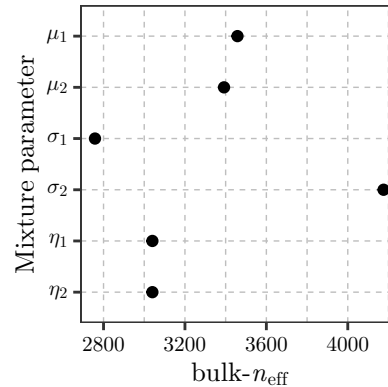


Figure 1: Example R `ggplot2` output. The y-axis tick mark labels and x-axis label were styled using \LaTeX .

Acknowledgements

This work was supported by the Australian Research Council Training Centre in Data Analytics for Resources and Environments (project ICI9010031) and funded by the Australian Government (including contributions by the Australian National Health and Medical Research Council Ideas Grant GNT1186572).

References

- [1] L. Scarso. Two applications of SWIGLIB: GraphicsMagick and Ghostscript. *TUGboat* 36(3):237–242, 2015. tug.org/TUGboat/tb36-3/tb114scarso.pdf
- [2] W. Schmidt. Font selection in \LaTeX : The most frequently asked questions. *TUGboat* 28(2):241–242, 2007. tug.org/TUGboat/tb28-2/tb89schmidt.pdf
- [3] A. Vehtari, A. Gelman, et al. Rank-normalization, folding, and localization: An improved \hat{R} for assessing convergence of mcmc (with discussion). *Bayesian Analysis* 16(2):667–718, June 2021. doi.org/10.1214/20-BA1221
- [4] B. Veytsman. Using knitr and \LaTeX for literate laboratory notes. *TUGboat* 43(2):130–133, 2022. tug.org/TUGboat/tb43-2/tb134veytsman-labnotes.pdf
- [5] U. Ziegenhagen. Dynamic reporting with R/Sweave and \LaTeX . *TUGboat* 31(2):189–192, 2010. tug.org/TUGboat/tb31-2/tb98ziegenhagen.pdf

◇ Travis Stenborg
 Sydney, Australia
 ORCID 0000-0002-2693-9628