

Package ‘extrafont’

September 24, 2025

Type Package

Title Tools for Using Fonts

Version 0.20

Date 2025-09-24

Depends R (>= 2.15)

Imports extrafontdb, grDevices, utils, Rttf2pt1

Suggests fontcm, testthat, withr (>= 2.5.0)

Author Winston Chang [aut],
Frederic Bertrand [cre] (ORCID:
<<https://orcid.org/0000-0002-0837-8281>>)

Maintainer Frederic Bertrand <frederic.bertrand@lecnam.net>

Description Tools to using fonts other than the standard PostScript fonts.
This package makes it easy to use system TrueType fonts and with PDF or PostScript output files, and with bitmap output files in Windows. extrafont can also be used with fonts packaged specifically to be used with, such as the fontcm package, which has Computer Modern PostScript fonts with math symbols.

License GPL-2

LazyLoad yes

NeedsCompilation no

URL <https://github.com/fbertran/extrafont>

BugReports <https://github.com/fbertran/extrafont/issues>

RoxygenNote 7.3.3

Encoding UTF-8

Config/testthat/edition 3

Repository CRAN

Date/Publication 2025-09-24 10:50:02 UTC

Contents

choose_font	2
embed_fonts	3
fonts	4
fonttable	4
font_addpackage	4
font_import	5
font_install	5
loadfonts	6
ttf_import	6
Index	7

choose_font	<i>Choose an installed font from a list</i>
-------------	---

Description

For a sequence of font family names, return the first one installed on the system. This makes it easy for code to specify a preferred font-family, but fall back to other font families if that is not installed on the system. This function acts much like the CSS font-family property.

Usage

```
choose_font(fonts, quiet = TRUE)
```

Arguments

- fonts character List of font family names to try.
- quiet logical Do not print warning if the preferred font not found.

Value

character. First font in fonts that is installed on the system or "" if none of those are installed.

Examples

```
choose_font(c("GillSans", "Verdana", "sans"), quiet = TRUE)
choose_font(c("BemeboStd", "Garamond", "serif"), quiet = TRUE)
```

embed_fonts*Embeds fonts that are listed in the local Fontmap*

Description

Embeds fonts that are listed in the local Fontmap

Usage

```
embed_fonts(file, format, outfile = file, options = "")
```

Arguments

file	Name of input file.
format	File format. (see ?embedFonts)
outfile	Name of the output file (with fonts embedded). (Default is same as input file)
options	Other arguments passed to embedFonts.

See Also

[loadfonts](#)

Examples

```
## Not run:
loadfonts()
pdf('fonttest.pdf')
library(ggplot2)

p <- ggplot(mtcars, aes(x=wt, y=mpg)) + geom_point()

# Run only the code below that is appropriate for your system
# On Mac and Windows, Impact should be available
p + opts(axis.title.x=theme_text(size=16, family="Impact", colour="red"))

# On Linux, Purisa may be available
p + opts(axis.title.x=theme_text(size=16, family="Purisa", colour="red"))
dev.off()

embed_fonts('fonttest.pdf', outfile='fonttest-embed.pdf')

## End(Not run)
```

fonts	<i>Show the fonts that are registered in the font table (and available for embedding)</i>
-------	---

Description

Show the fonts that are registered in the font table (and available for embedding)

Usage

```
fonts()
```

fonttable	<i>Returns the full font table</i>
-----------	------------------------------------

Description

Returns the full font table

Usage

```
fonttable()
```

font_addpackage	<i>Add font from an installed package to fonts database</i>
-----------------	---

Description

This is usually not called directly, but via [font_install](#).

Usage

```
font_addpackage(pkg = NULL)
```

Arguments

pkg	The name of the font package, e.g., "fontcm".
-----	---

See Also

[font_install](#)

font_import	<i>Import system fonts</i>
-------------	----------------------------

Description

Presently only supports TrueType fonts.

Usage

```
font_import(paths = NULL, recursive = TRUE, prompt = TRUE, pattern = NULL)
```

Arguments

paths	A vector of directories to search in. (Default is to auto-detect based on OS)
recursive	Search recursively in directories? (Default TRUE)
prompt	Show confirmation prompt? (Default TRUE)
pattern	A regular expression that the filenames must match.

Examples

```
font_import()
```

font_install	<i>Install a font package and register it in the fonts database</i>
--------------	---

Description

If the font package specified by fontpkg is not already installed, it will be downloaded from CRAN. The font package will then be registered in the fonts database.

Usage

```
font_install(fontpkg = NULL, prompt = TRUE)
```

Arguments

fontpkg	The name of an R package containing a font, e.g., "fontcm".
prompt	Ask user to install font from CRAN if not already installed

Examples

```
## Not run:  
font_install('fontcm')  
  
## End(Not run)
```

loadfonts	<i>Reads the fonttable database and registers those fonts with R</i>
-----------	--

Description

This registers fonts so that they can be used with the pdf, postscript, or Windows bitmap output device. It must be run once in each R session.

Usage

```
loadfonts(device = c("all", "pdf", "postscript", "win"), quiet = FALSE)
```

Arguments

device	The output device. If "all", then it will load "pdf", "postscript", and "win" (if on Windows).
quiet	If FALSE, print a status message as each font is registered. If TRUE, don't print.

Value

A named list with up to three elements, one for each device for which fonts were loaded. Each device element is a named list, with an element for each family that was the function attempted to register with that device. The value is NULL if the function did not register the font family due to problems or because the font family was already registered. If value is the return value of windowsFonts for "win", [postscriptFonts](#) for "postscript", and [pdfFonts](#) for "pdf".

See Also

[embed_fonts](#), `#ifdef windows` [windowsFont](#), [windowsFonts](#), `#endif` [postscriptFonts](#), [pdfFonts](#), [Type1Font](#).

ttf_import	<i>Imports all TrueType fonts in a directory and all subdirectories</i>
------------	---

Description

Imports all TrueType fonts in a directory and all subdirectories

Usage

```
ttf_import(paths = NULL, recursive = TRUE, pattern = NULL)
```

Arguments

paths	A vector of directories to search in. (Default is to auto-detect based on OS)
recursive	Search recursively in directories? (Default TRUE)
pattern	A regular expression that the filenames must match.

Index

`choose_font`, [2](#)

`embed_fonts`, [3](#), [6](#)

`font_addpackage`, [4](#)

`font_import`, [5](#)

`font_install`, [4](#), [5](#)

`fonts`, [4](#)

`fonttable`, [4](#)

`loadfonts`, [3](#), [6](#)

`pdfFonts`, [6](#)

`postscriptFonts`, [6](#)

`ttf_import`, [6](#)

`Type1Font`, [6](#)

`windowsFont`, [6](#)

`windowsFonts`, [6](#)