Package ‘hrbrthemes’

October 13, 2022

<table>
<thead>
<tr>
<th>Type</th>
<th>Package</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Additional Themes, Theme Components and Utilities for ‘ggplot2’</td>
</tr>
<tr>
<td>Version</td>
<td>0.8.0</td>
</tr>
<tr>
<td>Date</td>
<td>2020-03-05</td>
</tr>
<tr>
<td>Maintainer</td>
<td>Bob Rudis <a href="mailto:bob@rud.is">bob@rud.is</a></td>
</tr>
</tbody>
</table>

**Description**  
A compilation of extra ‘ggplot2’ themes, scales and utilities, including a spell check function for plot label fields and an overall emphasis on typography. A copy of the ‘Google’ font ‘Roboto Condensed’ [https://github.com/google/roboto/](https://github.com/google/roboto/) is also included along with a copy of the ‘IBM’ ‘Plex Sans’ [https://github.com/IBM/type](https://github.com/IBM/type), ‘Titillium Web’ [https://fonts.google.com/specimen/TitilliumWeb](https://fonts.google.com/specimen/TitilliumWeb), and ‘Public Sans’ [https://github.com/uswds/public-sans/](https://github.com/uswds/public-sans/) fonts are also included to support their respective typography-oriented themes.

**URL**  
http://github.com/hrbrmstr/hrbrthemes

**BugReports**  
https://github.com/hrbrmstr/hrbrthemes/issues

**Copyright**  
file inst/COPYRIGHTS

**License**  
MIT + file LICENSE

**Encoding**  
UTF-8

**Suggests**  
testthat, dplyr, gridExtra, hunspell, stringi, gcookbook, clipr, vdiff, svglite

**Depends**  
R (>= 3.4.0)

**Imports**  
ggplot2 (>= 3.3.0), grDevices, grid, scales, extrafont, knitr, rmarkdown, htmltools, tools, magrittr, gdtools

**RoxygenNote**  
7.0.2

**VignetteBuilder**  
knitr

**NeedsCompilation**  
no

**Author**  
Bob Rudis [aut, cre] [https://orcid.org/0000-0001-5670-2640>,  
Patrick Kennedy [ctb],  
Philipp Reiner [ctb],  
Dan Wilson [ctb] (Secondary axis support),  
Xavier Adam [ctb],
Google [cph] (Roboto Condensed & Titillium Web Fonts),
IBM [cph] (Plex Sans Font),
Impallari Type [cph] (Public Sans Font),
Jacob Barnett [ctb],
Thomas J. Leeper [ctb] (<https://orcid.org/0000-0003-4097-6326>),
Joris Meys [ctb]

Repository CRAN

Date/Publication 2020-03-06 18:50:02 UTC

R topics documented:

flush_ticks ................................................................. 3
font_an ................................................................. 3
font_es ................................................................. 4
font_ps ................................................................. 4
font_pub ................................................................. 5
font_re ................................................................. 5
font_th ................................................................. 6
font_tw ................................................................. 6
ft_cols ................................................................. 7
ft_geom_defaults ......................................................... 7
ft_pal ................................................................. 8
gg_check ............................................................. 8
hrbrthemes-exports ..................................................... 9
import_econ_sans ....................................................... 9
import_plex_sans ...................................................... 10
import_public_sans .................................................... 10
import_robotoList ....................................................... 11
import_tinyhand ....................................................... 11
import_titillium_web .................................................. 12
ipsum ................................................................. 12
ipsum_pal ............................................................ 14
ipsum_pdf ............................................................ 14
modern_geom_defaults .................................................. 15
scale_colour_ft .......................................................... 15
scale_colour_ipsum ..................................................... 16
scale_x_percent .......................................................... 17
theme_ft_rc ............................................................ 20
theme_ipsum ............................................................ 24
theme_ipsum_es ........................................................ 27
theme_ipsum_ps ........................................................ 29
theme_ipsum_pub ........................................................ 31
theme_ipsum_tw ........................................................ 34
theme_tinyhand ........................................................ 36
update_geom_font_defaults ............................................. 38

Index 39
flush_ticks

Makes axis text labels flush on the ends

Description

A convenience function intended for basic, fixed-scale plots only (i.e. does not handle free scales in facets).

You need to pass in a ggplot2 object to this function. It can’t be +’d in a chain of geoms, co-ords, scales, themes, etc. It also builds the plot (but does not display it) so if the plt takes a while (i.e. has lots of data or transforms) this will also take a while.

Usage

flush_ticks(gg, flush = "XY", plot = TRUE, cat = TRUE)

Arguments

- gg: ggplot2 plot object
- flush: either "X" or "Y" or "XY" to flush individual or both axes. Default: both.
- plot: if FALSE then the ggplot object will be returned invisibly
- cat: if TRUE then display theme() statements and copy them to the clipboard

Value

ggplot2 object with theme() elements added

Note

Intended for basic, fixed-scale plots only (i.e. does not handle free scales in facets).

font_an

Arial Narrow font name R variable aliases

Description

font_an == "Arial Narrow"

Usage

font_an

Format

length 1 character vector
**font_es**  
*Econ Sans Condensed font name R variable aliases*

**Description**

- `font_es == "EconSansCndLig"`
- `font_es_bold == "EconSansCndBol"`
- `font_es_light == "EconSansCndLig"`

**Usage**

- `font_es`
- `font_es_bold`
- `font_es_light`

**Format**

- length 1 character vector

**Note**

- `font_es_bold` (a.k.a. "EconSansCndBol") is not available on Windows and will throw a warning if used in plots.
- `font_es_light` (a.k.a. "EconSansCndLig") is not available on Windows and will throw a warning if used in plots.

---

**font_ps**  
*PlexSans font name R variable aliases*

**Description**

- `font_ps == "IBMPlexSans"
- `font_ps_light == "IBMPlexSans-Light"

**Usage**

- `font_ps`
- `font_ps_light`

**Format**

- length 1 character vector
**font_pub**

**Note**

font_ps_light (a.k.a. "IBMPlexSans-Light") is not available on Windows and will throw a warning if used in plots.

---

<table>
<thead>
<tr>
<th>font_pub</th>
<th>Public Sans font name R variable aliases</th>
</tr>
</thead>
</table>

**Description**

font_pub == "Public Sans"
font_pub_bold == "Public Sans Bold"
font_pub_light == "Public Sans Light"
font_pub_thin == "Public Sans Thin"

**Usage**

font_pub
font_pub_bold
font_pub_light
font_pub_thin

**Format**

length 1 character vector

**Note**

font_pub_bold (a.k.a. "Public Sans Bold") is not available on Windows and will throw a warning if used in plots.

---

<table>
<thead>
<tr>
<th>font_rc</th>
<th>Roboto Condensed font name R variable aliases</th>
</tr>
</thead>
</table>

**Description**

font_rc == "Roboto Condensed"
font_fc_light == "Roboto Condensed Light"
Usage

- font_rc
- font_rc_light

Format

- length 1 character vector

Note

- font_rc_light (a.k.a. "Roboto Condensed Light") is not available on Windows and will throw a warning if used in plots.

---

TinyHand Web font name R variable aliases

Usage

- font_th

Format

- length 1 character vector

---

Titillium Web font name R variable aliases

Usage

- font_tw
- font_tw_bold
- font_tw_light
ft_cols

Format
length 1 character vector

Note
font_tw_light (a.k.a. "Titillium Web Bold") is not available on Windows and will throw a warning if used in plots.
font_tw_light (a.k.a. "Titillium Web Light") is not available on Windows and will throw a warning if used in plots.

<table>
<thead>
<tr>
<th>ft_cols</th>
<th>FT color palette</th>
</tr>
</thead>
</table>

Description
FT color palette

Usage
ft_cols
ft_text_col

Format
An object of class list of length 9.

Note
don’t forget you can use scales::alpha() with these colors

<table>
<thead>
<tr>
<th>ft_geom_defaults</th>
<th>Change geom defaults from black to custom lights for the FT theme</th>
</tr>
</thead>
</table>

Description
Change geom defaults from black to custom lights for the FT theme

Usage
ft_geom_defaults()
ft_pal  

A bright qualitative color palette

Description

A bright qualitative color palette

Usage

ft_pal()

Examples

library(scales)
scales::show_col(ft_pal()(8))

---

gg_check  

Spell check ggplot2 plot labels

Description

Due to the way ggplot2 objects are created, this has to be used in a standalone context.

Usage

gg_check(gg, dict, ignore)

Arguments

- **gg**: ggplot2 object
- **dict**: a dictionary object or string which can be passed to *hunspell::dictionary*. Defaults to *hunspell::dictionary("en_US")*
- **ignore**: character vector with additional approved words added to the dictionary. Defaults to *hunspell::en_stats*

Details

Current functionality only looks for misspelled words in the labels of ggplot2 objects. When misspelled words are found, a message is printed with the words and the label that they are in. No messages will be printed if there are no misspelled words.

Value

the object that was passed in
Examples

```r
library(ggplot2)

df <- data.frame(x=c(20, 25, 30), y=c(4, 4, 4), txt=c("One", "Two", "Three"))

# not piping
ggplot(mtcars, aes(mpg, wt)) +
  geom_point() +
  labs(x="This is some txt", y="This is more text",
       title="This is a title",
       subtitle="This is a subtitle",
       caption="This is a caption") -> gg

gg_check(gg)
```

hrbrthemes-exports  hrbrthemes exported operators

Description

The following functions are imported and then re-exported from the hrbrthemes package to enable use of the magrittr pipe operator with no additional library calls

```r
import_econ_sans
```

Description

Roboto Condensed is a trademark of Google.

Usage

```r
import_econ_sans()
```

Details

There is an option `hrbrthemes.loadfonts` which – if set to `TRUE` – will call `extrafont::loadfonts()` to register non-core fonts with R PDF & PostScript devices. If you are running under Windows, the package calls the same function to register non-core fonts with the Windows graphics device.

Note

This will take care of ensuring PDF/PostScript usage. The location of the font directory is displayed after the base import is complete. It is highly recommended that you install them on your system the same way you would any other font you wish to use in other programs.
**Description**

IBM Plex Sans is a trademark of IBM and distributed under the SIL Open Font License, Version 1.1.

**Usage**

```r
import_plex_sans()
```

**Details**

There is an option `hrbrthemes.loadfonts` which – if set to `TRUE` – will call `extrafont::loadfonts()` to register non-core fonts with R PDF & PostScript devices. If you are running under Windows, the package calls the same function to register non-core fonts with the Windows graphics device.

**Note**

This will take care of ensuring PDF/PostScript usage. The location of the font directory is displayed after the base import is complete. It is highly recommended that you install them on your system the same way you would any other font you wish to use in other programs.

---

**Description**

Public Sans is Copyright 2015 Impallari Type and licensed under the SIL Open Font License, Version 1.1.

**Usage**

```r
import_public_sans()
```

**Details**

There is an option `hrbrthemes.loadfonts` which – if set to `TRUE` – will call `extrafont::loadfonts()` to register non-core fonts with R PDF & PostScript devices. If you are running under Windows, the package calls the same function to register non-core fonts with the Windows graphics device.

**Note**

This will take care of ensuring PDF/PostScript usage. The location of the font directory is displayed after the base import is complete. It is highly recommended that you install them on your system the same way you would any other font you wish to use in other programs.
**import_robo condensed**

*Import Roboto Condensed font for use in charts*

**Description**

Roboto Condensed is a trademark of Google.

**Usage**

import_robo condensed()

**Details**

There is an option hrbrthemes::loadfonts which – if set to TRUE – will call extrafont::loadfonts() to register non-core fonts with R PDF & PostScript devices. If you are running under Windows, the package calls the same function to register non-core fonts with the Windows graphics device.

**Note**

This will take care of ensuring PDF/PostScript usage. The location of the font directory is displayed after the base import is complete. It is highly recommended that you install them on your system the same way you would any other font you wish to use in other programs.

---

**import_tinyhand**

*Import Titillium Web font for use in charts*

**Description**

Titillium Web is a trademark of Google.

**Usage**

import_tinyhand()

**Details**

There is an option hrbrthemes::loadfonts which – if set to TRUE – will call extrafont::loadfonts() to register non-core fonts with R PDF & PostScript devices. If you are running under Windows, the package calls the same function to register non-core fonts with the Windows graphics device.

**Note**

This will take care of ensuring PDF/PostScript usage. The location of the font directory is displayed after the base import is complete. It is highly recommended that you install them on your system the same way you would any other font you wish to use in other programs.
import_titillium_web Import Titillium Web font for use in charts

Description

Titillium Web is a trademark of Google.

Usage

import_titillium_web()

Details

There is an option hrbrthemes::loadfonts which – if set to TRUE – will call extrafont::loadfonts() to register non-core fonts with R PDF & PostScript devices. If you are running under Windows, the package calls the same function to register non-core fonts with the Windows graphics device.

Note

This will take care of ensuring PDF/PostScript usage. The location of the font directory is displayed after the base import is complete. It is highly recommended that you install them on your system the same way you would any other font you wish to use in other programs.

ipsum ipsum R markdown template

Description

Template for creating an R markdown document with an emphasis on typography

Usage

ipsum(
  number_sections = FALSE,
  fig_width = 7,
  fig_height = 5,
  fig_retina = if (!fig_caption) 2,
  fig_caption = FALSE,
  dev = "png",
  smart = TRUE,
  self_contained = TRUE,
  highlight = "default",
  mathjax = "default",
  extra_dependencies = NULL,
  css = NULL,
  includes = NULL,
keep_md = FALSE,
lib_dir = NULL,
md_extensions = NULL,
pandoc_args = NULL,
toc = FALSE,
toc_depth = 2,
...)

Arguments

number_sections  TRUE to number section headings
fig_width        Default width (in inches) for figures
fig_height       Default height (in inches) for figures
fig_retina       Scaling to perform for retina displays (defaults to 2, which currently works for all widely used retina displays). Set to NULL to prevent retina scaling. Note that this will always be NULL when keep_md is specified (this is because fig_retina relies on outputting HTML directly into the markdown document).
fig_caption      TRUE to render figures with captions
dev              Graphics device to use for figure output (defaults to png)
smart            Produce typographically correct output, converting straight quotes to curly quotes, -- to em-dashes, --- to en-dashes, and . . . to ellipses.
self_contained   Produce a standalone HTML file with no external dependencies, using data: URLs to incorporate the contents of linked scripts, stylesheets, images, and videos. Note that even for self contained documents MathJax is still loaded externally (this is necessary because of its size).
mathjax           Include mathjax. The "default" option uses an https URL from a MathJax CDN. The "local" option uses a local version of MathJax (which is copied into the output directory). You can pass an alternate URL or pass NULL to exclude MathJax entirely.
extra_dependencies,...

Additional function arguments to pass to the base R Markdown HTML output formatter

css              One or more css files to include
includes         Named list of additional content to include within the document (typically created using the includes function).
keep_md          Keep the markdown file generated by knitting.
lib_dir          Directory to copy dependent HTML libraries (e.g.jquery, bootstrap, etc.) into. By default this will be the name of the document with _files appended to it.
md_extensions    Markdown extensions to be added or removed from the default definition or R Markdown. See the rmarkdown_format for additional details.
pandoc_args  Additional command line options to pass to pandoc
toc, toc_depth  TOC params

---

**ipsum_pal**  
A muted, qualitative color palette

**Description**
A muted, qualitative color palette

**Usage**
ipsum_pal()

**Examples**
library(scales)
scales::show_col(ipsum_pal()(9))

---

**ipsum_pdf**  
ipsum R markdown template for PDF output

**Description**
Template for creating an R markdown documents with an emphasis on typography

**Usage**
ipsum_pdf(...)

**Arguments**

...  Arguments to rmarkdown::pdf_document

**Value**
R Markdown output format to pass to render
modern_geom_defaults  

Change geom defaults from black to white for the modern theme

Description

Change geom defaults from black to white for the modern theme

Usage

modern_geom_defaults()

scale_colour_ft  

Discrete color & fill scales based on the FT palette

Description

See ft_pal().

Usage

scale_colour_ft(...)  
scale_color_ft(...)  
scale_fill_ft(...)

Arguments

...  
Arguments passed on to ggplot2::discrete_scale  
aesthetics  
The names of the aesthetics that this scale works with.  
scale_name  
The name of the scale that should be used for error messages associated with this scale.  
palette  
A palette function that when called with a single integer argument (the number of levels in the scale) returns the values that they should take (e.g., scales::hue_pal()).  
name  
The name of the scale. Used as the axis or legend title. If waiver(), the default, the name of the scale is taken from the first mapping used for that aesthetic. If NULL, the legend title will be omitted.  
breaks  
One of:
  • NULL for no breaks
  • waiver() for the default breaks (the scale limits)
  • A character vector of breaks
  • A function that takes the limits as input and returns breaks as output  
labels  
One of:
• NULL for no labels
• waiver() for the default labels computed by the transformation object
• A character vector giving labels (must be same length as breaks)
• A function that takes the breaks as input and returns labels as output

limits A character vector that defines possible values of the scale and their order.

na.translate Unlike continuous scales, discrete scales can easily show missing values, and do so by default. If you want to remove missing values from a discrete scale, specify na.translate = FALSE.

na.value If na.translate = TRUE, what value aesthetic value should missing be displayed as? Does not apply to position scales where NA is always placed at the far right.

drop Should unused factor levels be omitted from the scale? The default, TRUE, uses the levels that appear in the data; FALSE uses all the levels in the factor.

guide A function used to create a guide or its name. See guides() for more information.

super The super class to use for the constructed scale

describe • use when aesthetic is categorical

scale_colour_ipsum Discrete color & fill scales based on the ipsum palette

Description

See ipsum_pal().

Usage

scale_colour_ipsum(...)
scale_color_ipsum(...)
scale_fill_ipsum(...)

Arguments

... Arguments passed on to ggplot2::discrete_scale

aesthetics The names of the aesthetics that this scale works with.
scale_name The name of the scale that should be used for error messages associated with this scale.

palette A palette function that when called with a single integer argument (the number of levels in the scale) returns the values that they should take (e.g., scales::hue_pal()).

name The name of the scale. Used as the axis or legend title. If waiver(), the default, the name of the scale is taken from the first mapping used for that aesthetic. If NULL, the legend title will be omitted.
breaks One of:
  • NULL for no breaks
  • waiver() for the default breaks (the scale limits)
  • A character vector of breaks
  • A function that takes the limits as input and returns breaks as output

labels One of:
  • NULL for no labels
  • waiver() for the default labels computed by the transformation object
  • A character vector giving labels (must be same length as breaks)
  • A function that takes the breaks as input and returns labels as output

limits A character vector that defines possible values of the scale and their order.

na.translate Unlike continuous scales, discrete scales can easily show missing values, and do so by default. If you want to remove missing values from a discrete scale, specify na.translate = FALSE.

na.value If na.translate = TRUE, what value aesthetic value should missing be displayed as? Does not apply to position scales where NA is always placed at the far right.

drop Should unused factor levels be omitted from the scale? The default, TRUE, uses the levels that appear in the data; FALSE uses all the levels in the factor.

guide A function used to create a guide or its name. See guides() for more information.

super The super class to use for the constructed scale

---

scale_x_percent  

\textit{X} & \textit{Y} scales with opinionated pre-sets for percent & comma label formats

\textbf{Description}

The _comma ones set comma format for axis text and expand=c(0, 0) (you need to set limits).

\textbf{Usage}

\begin{verbatim}
scale_x_percent(
  name = waiver(),
  breaks = waiver(),
  minor_breaks = waiver(),
  guide = waiver(),
  n.breaks = NULL,
  labels,
  limits = NULL,
  expand = c(0.01, 0),
  oob = censor,
  na.value = NA_real_,
)\end{verbatim}
trans = "identity",
position = "bottom",
sec.axis = waiver(),
accuracy = 1,
scale = 100,
prefix = "",
suffix = "%",
big.mark = " ",
decimal.mark = ".",
trim = TRUE, ...)

scale_y_percent(
  name = waiver(),
breaks = waiver(),
minor_breaks = waiver(),
guide = waiver(),
n.breaks = NULL,
labels,
limits = NULL,
expand = c(0.01, 0),
oob = censor,
na.value = NA_real_,
trans = "identity",
position = "left",
sec.axis = waiver(),
accuracy = 1,
scale = 100,
prefix = "",
suffix = "%",
big.mark = " ",
decimal.mark = ".",
trim = TRUE, ...)

scale_x_comma(
  name = waiver(),
breaks = waiver(),
minor_breaks = waiver(),
guide = waiver(),
n.breaks = NULL,
labels,
limits = NULL,
expand = c(0.01, 0),
oob = censor,
na.value = NA_real_,

Arguments

name The name of the scale. Used as axis or legend title. If waiver(), the default, the name of the scale is taken from the first mapping used for that aesthetic. If NULL, the legend title will be omitted.

breaks One of:
  • NULL for no breaks
  • waiver() for the default breaks computed by the transformation object
  • A numeric vector of positions
  • A function that takes the limits as input and returns breaks as output

minor_breaks One of:
- NULL for no minor breaks
- waiver() for the default breaks (one minor break between each major break)
- A numeric vector of positions
- A function that given the limits returns a vector of minor breaks.

**guide**

A function used to create a guide or its name. See `guides()` for more information.

**n.breaks**

An integer guiding the number of major breaks. The algorithm may choose a slightly different number to ensure nice break labels. Will only have an effect if `breaks = waiver()`. Use NULL to use the default number of breaks given by the transformation.

**labels**

Specifying overrides the default format (i.e. you really don’t want to do that). NULL means no labels.

**limits**

A numeric vector of length two providing limits of the scale. Use NA to refer to the existing minimum or maximum.

**expand**

same as in ggplot2

**oob**

Function that handles limits outside of the scale limits (out of bounds). The default replaces out of bounds values with NA.

**na.value**

If `na.translate = TRUE`, what value aesthetic value should missing be displayed as? Does not apply to position scales where NA is always placed at the far right.

**trans**

Either the name of a transformation object, or the object itself. Built-in transformations include "asn", "atanh", "boxcox", "exp", "identity", "log", "log10", "log1p", "log2", "logit", "probability", "probit", "reciprocal", "reverse" and "sqrt".

**position**

The position of the axis. "left" or "right" for vertical scales, "top" or "bottom" for horizontal scales

**sec.axis**

specify a secondary axis

**accuracy, scale, prefix, suffix, big.mark, decimal.mark, trim**

See [scales::comma_format()] or [scales::percent_format()]

**...**

passed on to [scales::comma_format()] or [scales::percent_format()]

**Details**

The _percent ones set percent format for axis text and expand=c(0,0) (you need to set limits).

---

**theme_ft_rc**

A precise & pristine ggplot2 theme with opinionated defaults and an emphasis on typography

**Description**

You should `import_roboato_condensed()` first and also install the fonts on your system before trying to use this theme.
Usage

```r
theme_ft_rc(
  base_family = "Roboto Condensed",
  base_size = 11.5,
  plot_title_family = base_family,
  plot_title_size = 18,
  plot_title_face = "bold",
  plot_title_margin = 10,
  subtitle_family = if (.Platform$OS.type == "windows") "Roboto Condensed" else "Roboto Condensed Light",
  subtitle_size = 13,
  subtitle_face = "plain",
  subtitle_margin = 15,
  strip_text_family = base_family,
  strip_text_size = 12,
  strip_text_face = "plain",
  caption_family = if (.Platform$OS.type == "windows") "Roboto Condensed" else "Roboto Condensed Light",
  caption_size = 9,
  caption_face = "plain",
  caption_margin = 10,
  axis_text_size = base_size,
  axis_title_family = base_family,
  axis_title_size = 9,
  axis_title_face = "plain",
  axis_title_just = "rt",
  plot_margin = margin(30, 30, 30, 30),
  grid = TRUE,
  axis = FALSE,
  ticks = FALSE
)
```

```r
theme_modern_rc(
  base_family = "Roboto Condensed",
  base_size = 11.5,
  plot_title_family = base_family,
  plot_title_size = 18,
  plot_title_face = "bold",
  plot_title_margin = 10,
  subtitle_family = if (.Platform$OS.type == "windows") "Roboto Condensed" else "Roboto Condensed Light",
  subtitle_size = 13,
  subtitle_face = "plain",
  subtitle_margin = 15,
  strip_text_family = base_family,
  strip_text_size = 12,
  strip_text_face = "plain",
  caption_family = if (.Platform$OS.type == "windows") "Roboto Condensed" else "Roboto Condensed Light",
  caption_size = 9,
  caption_face = "plain",
  caption_margin = 10,
  axis_text_size = base_size,
  axis_title_family = base_family,
  axis_title_size = 9,
  axis_title_face = "plain",
  axis_title_just = "rt",
  plot_margin = margin(30, 30, 30, 30),
  grid = TRUE,
  axis = FALSE,
  ticks = FALSE
)
```
theme_ft_rc

"Roboto Condensed Light",
caption_size = 9,
caption_face = "plain",
caption_margin = 10,
axis_text_size = base_size,
axis_title_family = base_family,
axis_title_size = 9,
axis_title_face = "plain",
axis_title_just = "rt",
plot_margin = margin(30, 30, 30, 30),
grid = TRUE,
axis = FALSE,
ticks = FALSE
)

theme_ipsum_rc(
  base_family = "Roboto Condensed",
  base_size = 11.5,
  plot_title_family = base_family,
  plot_title_size = 18,
  plot_title_face = "bold",
  plot_title_margin = 10,
  subtitle_family = if (.Platform$OS.type == "windows") "Roboto Condensed" else "Roboto Condensed Light",
  subtitle_size = 13,
  subtitle_face = "plain",
  subtitle_margin = 15,
  strip_text_family = base_family,
  strip_text_size = 12,
  strip_text_face = "plain",
  caption_family = if (.Platform$OS.type == "windows") "Roboto Condensed" else "Roboto Condensed Light",
  caption_size = 9,
  caption_face = "plain",
  caption_margin = 10,
  axis_text_size = base_size,
  axis_title_family = base_family,
  axis_title_size = 9,
  axis_title_face = "plain",
  axis_title_just = "rt",
  plot_margin = margin(30, 30, 30, 30),
  panel_spacing = grid::unit(2, "lines"),
  grid_col = "#cccccc",
  grid = TRUE,
  axis_col = "#cccccc",
  axis = FALSE,
  ticks = FALSE
)
Arguments

base_family, base_size
  base font family and size
plot_title_family, plot_title_face, plot_title_size, plot_title_margin
  plot title family, face, size and margin
subtitle_family, subtitle_face, subtitle_size
  plot subtitle family, face and size
subtitle_margin
  plot subtitle margin bottom (single numeric value)
istrip_text_family, strip_text_face, strip_text_size
  facet label font family, face and size
caption_family, caption_face, caption_size, caption_margin
  plot caption family, face, size and margin
axis_text_size
  font size of axis text
axis_title_family, axis_title_face, axis_title_size
  axis title font family, face and size
axis_title_just
  axis title font justificationk one of [blmcrt]
plot_margin
  plot margin (specify with ggplot2::margin)
grid
  panel grid (TRUE, FALSE, or a combination of X, x, Y, y)
axis
  add x or y axes? TRUE, FALSE, "xy"
ticks
  ticks if TRUE add ticks
panel_spacing
  panel spacing (use unit())
grid_col
  grid color
axis_col
  axis color

Details

There is an option hrbrthemes.loadfonts which – if set to TRUE – will call extrafont::loadfonts() to register non-core fonts with R PDF & PostScript devices. If you are running under Windows, the package calls the same function to register non-core fonts with the Windows graphics device.

Why Roboto Condensed?

It’s free, has tolerable kerning pairs and multiple weights. It’s also different than Arial Narrow and the fonts most folks use in ggplot2 charts.

Examples

```r
## Not run:
library(ggplot2)
library(dplyr)

# seminal scatterplot
ggplot(mtcars, aes(mpg, wt)) +
```
theme_ipsum

A precise & pristine ggplot2 theme with opinionated defaults and an emphasis on typography

Description

Also has a "dark" / "modern" version for the new RStudio theme

Usage

```r
theme_ipsum(
  base_family = "Arial Narrow",
  base_size = 11.5,
  plot_title_family = base_family,
  plot_title_size = 18,
  plot_title_face = "bold",
  plot_title_margin = 10,
  subtitle_family = base_family,
  subtitle_size = 12,
  subtitle_face = "plain",
  subtitle_margin = 15,
  strip_text_family = base_family,
  strip_text_size = 12,
  strip_text_face = "plain",
)```

```r
glimpse(mpg)
df <- mpg[as.numeric(mpg$class) %>% factor(., levels=c("cyl", "fin", "kit", "sib", "sta", "sup", "tis"), ordered=TRUE)]
glimpse(df)
```
caption_family = base_family,
caption_size = 9,
caption_face = "italic",
caption_margin = 10,
axis_text_size = base_size,
axis_title_family = subtitle_family,
axis_title_size = 9,
axis_title_face = "plain",
axis_title_just = "rt",
plot_margin = margin(30, 30, 30, 30),
grid_col = "#cccccc",
grid = TRUE,
axis_col = "#cccccc",
axis = FALSE,
ticks = FALSE
)

Arguments

base_family, base_size
base font family and size
plot_title_family, plot_title_face, plot_title_size, plot_title_margin
plot title family, face, size and margin
subtitle_family, subtitle_face, subtitle_size
plot subtitle family, face and size
subtitle_margin
plot subtitle margin bottom (single numeric value)
strip_text_family, strip_text_face, strip_text_size
facet label font family, face and size
caption_family, caption_face, caption_size, caption_margin
plot caption family, face, size and margin
axis_text_size
font size of axis text
axis_title_family, axis_title_face, axis_title_size
axis title font family, face and size
axis_title_just
axis title font justification, one of [blmcrt]
plot_margin
plot margin (specify with ggplot2::margin())
grid_col, axis_col
grid & axis colors; both default to #cccccc
grid
panel grid (TRUE, FALSE, or a combination of X, x, Y, y)
axis
add x or y axes? TRUE, FALSE, "xy"
ticks
ticks if TRUE add ticks

Why Arial Narrow?

First and foremost, Arial Narrow is generally installed by default or readily available on any modern system, so it’s "free"-ish; plus, it is a condensed font with solid default kerning pairs and geometric numbers.
Building upon theme_ipsum

The function is setup in such a way that you can customize your own one by just wrapping the call and changing the parameters. See source for examples.

Gotchas

There are distinctions between font names and various devices. Names that work for display graphics devices and bitmap ones such as png may not work well for PostScript or PDF ones. You may need two versions of a font-based theme function for them to work in a particular situation. This situation usually only arises when using a newer font with many weights but somewhat irregular internal font name patterns.

There is an option hrbrthemes.loadfonts which – if set to TRUE – will call extrafont::loadfonts() to register non-core fonts with R PDF & PostScript devices. If you are running under Windows, the package calls the same function to register non-core fonts with the Windows graphics device.

Examples

## Not run:
library(ggplot2)
library(dplyr)

# seminal scatterplot
ggplot(mtcars, aes(mpg, wt)) + 
  geom_point() +
  labs(x="Fuel effiiency (mpg)", y="Weight (tons)",
       title="Seminal ggplot2 scatterplot example",
       subtitle="A plot that is only useful for demonstration purposes",
       caption="Brought to you by the letter 'g'") +
  theme_ipsum()

# seminal bar chart

update_geom_font_defaults()

count(mpg, class) %>%
  ggplot(aes(class, n)) +
  geom_col() +
  geom_text(aes(label=n), nudge_y=3) +
  labs(x="Fuel efficiency (mpg)", y="Weight (tons)",
       title="Seminal ggplot2 bar chart example",
       subtitle="A plot that is only useful for demonstration purposes",
       caption="Brought to you by the letter 'g'") +
  theme_ipsum(grid="Y") +
  theme(axis.text.y=element_blank())

## End(Not run)
theme_ipsum_es

A precise & pristine ggplot2 theme with opinionated defaults and an emphasis on typography

Description

You should `import_econ_sans()` first and also install the fonts on your system before trying to use this theme.

Usage

theme_ipsum_es(
  base_family = "EconSansCndReg",
  base_size = 11.5,
  plot_title_family = "EconSansCndBol",
  plot_title_size = 18,
  plot_title_face = "bold",
  plot_title_margin = 10,
  subtitle_family = if (.Platform$OS.type == "windows") "EconSansCndLig" else "EconSansCndLig",
  subtitle_size = 13,
  subtitle_face = "plain",
  subtitle_margin = 15,
  strip_text_family = base_family,
  strip_text_size = 12,
  strip_text_face = "plain",
  caption_family = if (.Platform$OS.type == "windows") "EconSansCndLig" else "EconSansCndLig",
  caption_size = 9,
  caption_face = "plain",
  caption_margin = 10,
  axis_text_size = base_size,
  axis_title_family = base_family,
  axis_title_size = 9,
  axis_title_face = "plain",
  axis_title_just = "rt",
  plot_margin = margin(30, 30, 30, 30),
  panel_spacing = grid::unit(2, "lines"),
  grid_col = "#cccccc",
  grid = TRUE,
  axis_col = "#cccccc",
  axis = FALSE,
  ticks = FALSE
)
Arguments

- `base_family`, `base_size`  
  base font family and size
- `plot_title_family`, `plot_title_face`, `plot_title_size`, `plot_title_margin`  
  plot title family, face, size and margin
- `subtitle_family`, `subtitle_face`, `subtitle_size`  
  plot subtitle family, face and size
- `subtitle_margin`  
  plot subtitle margin bottom (single numeric value)
- `strip_text_family`, `strip_text_face`, `strip_text_size`  
  facet label font family, face and size
- `caption_family`, `caption_face`, `caption_size`, `caption_margin`  
  plot caption family, face, size and margin
- `axis_text_size`  
  font size of axis text
- `axis_title_family`, `axis_title_face`, `axis_title_size`  
  axis title font family, face and size
- `axis_title_just`  
  axis title font justificationk one of [bcmlcr\r]
- `plot_margin`  
  plot margin (specify with `ggplot2::margin()`)  
  panel spacing (use `unit()`)
- `grid_col`  
  grid color
- `grid`  
  panel grid (TRUE, FALSE, or a combination of `X`, `x`, `Y`, `y`)
- `axis_col`  
  axis color
- `axis`  
  add x or y axes? TRUE, FALSE, "xy"
- `ticks`  
  ticks if TRUE add ticks

Details

There is an option `hrbrthemes.loadfonts` which – if set to `TRUE` – will call `extrafont::loadfonts()` to register non-core fonts with R PDF & PostScript devices. If you are running under Windows, the package calls the same function to register non-core fonts with the Windows graphics device.

Why Econ Sans Condensed?

It’s free, has tolerable kerning pairs and multiple weights. It’s also different than Arial Narrow and the fonts most folks use in ggplot2 charts.

Examples

```r
## Not run:
library(ggplot2)
library(dplyr)

# seminal scatterplot
ggplot(mtcars, aes(mpg, wt)) +
```
theme_ipsum_ps

A precise & pristine ggplot2 theme with opinionated defaults and an emphasis on typography

Description

You should import_plex_sans() first and also install the fonts on your system before trying to use this theme.

Usage

theme_ipsum_ps(
  base_family = "IBMPlexSans",
  base_size = 11.5,
  plot_title_family = "IBMPlexSans-Bold",
  plot_title_size = 18,
  plot_title_face = "plain",
  plot_title_margin = 10,
  subtitle_family = if (.Platform$OS.type == "windows") "IBMPlexSans" else "IBMPlexSans-Light",
  subtitle_size = 13,
  subtitle_face = "plain",
  subtitle_margin = 15,
  strip_text_family = "IBMPlexSans-Medium",

  geom_point() +
  labs(x="Fuel efficiency (mpg)", y="Weight (tons)",
       title="Seminal ggplot2 scatterplot example",
       subtitle="A plot that is only useful for demonstration purposes",
       caption="Brought to you by the letter 'g'") +
  theme_ipsum_es()

# seminal bar chart

# note: may need to make this font_es on Windows
update_geom_font_defaults(family=font_es_light)

count(mpg, class) %>%
  ggplot(aes(class, n)) +
  geom_col() +
  geom_text(aes(label=n), nudge_y=3) +
  labs(x="Fuel efficiency (mpg)", y="Weight (tons)",
       title="Seminal ggplot2 bar chart example",
       subtitle="A plot that is only useful for demonstration purposes",
       caption="Brought to you by the letter 'g'") +
  theme_ipsum_es(grid="Y") +
  theme(axis.text.y=element_blank())

## End(Not run)
strip_text_size = 12,
strip_text_face = "plain",
caption_family = if (.Platform$OS.type == "windows") "IBMPlexSans" else
   "IBMPlexSans-Thin",
caption_size = 9,
caption_face = "plain",
caption_margin = 10,
axis_text_size = 9,
axis_title_family = base_family,
axis_title_size = 9,
axis_title_face = "plain",
plot_margin = margin(30, 30, 30, 30),
grid_col = "ccccccc",
grid = TRUE,
axis_col = "ccccccc",
axis = FALSE,
ticks = FALSE
)

Arguments

base_family, base_size
  base font family and size
plot_title_family, plot_title_face, plot_title_size, plot_title_margin
  plot title family, face, size and margin
subtitle_family, subtitle_face, subtitle_size
  plot subtitle family, face and size
subtitle_margin
  plot subtitle margin bottom (single numeric value)
strip_text_family, strip_text_face, strip_text_size
  facet label font family, face and size
caption_family, caption_face, caption_size, caption_margin
  plot caption family, face, size and margin
axis_text_size
  font size of axis text
axis_title_family, axis_title_face, axis_title_size
  axis title font family, face and size
axis_title_just
  axis title font justificationk one of [blmcrt]
plot_margin
  plot margin (specify with ggplot2::margin)
grid_col
  grid color
grid
  panel grid (TRUE, FALSE, or a combination of X, x, Y, y)
axis_col
  axis color
axis
  add x or y axes? TRUE, FALSE, "xy"
ticks
  ticks if TRUE add ticks
Details

There is an option `hrbrthemes.loadfonts` which – if set to `TRUE` – will call `extrafont::loadfonts()` to register non-core fonts with R PDF & PostScript devices. If you are running under Windows, the package calls the same function to register non-core fonts with the Windows graphics device.

Why IBM Plex Sans?

It’s free, has tolerable kerning pairs and multiple weights. It’s also different “not Helvetica”.

Examples

```r
## Not run:
library(ggplot2)
library(dplyr)

# seminal scatterplot
ggplot(mtcars, aes(mpg, wt)) +
  geom_point() +
  labs(x="Fuel efficiency (mpg)",
       y="Weight (tons)",
       title="Seminal ggplot2 scatterplot example",
       subtitle="A plot that is only useful for demonstration purposes",
       caption="Brought to you by the letter 'g'") +
  theme_ipsum_rc()

# seminal bar chart

# note: make this font_rc on Windows
update_geom_font_defaults(family=font_rc_light)

count(mpg, class) %>%
  ggplot(aes(class, n)) +
  geom_col() +
  geom_text(aes(label=n), nudge_y=3) +
  labs(x="Fuel efficiency (mpg)",
       y="Weight (tons)",
       title="Seminal ggplot2 bar chart example",
       subtitle="A plot that is only useful for demonstration purposes",
       caption="Brought to you by the letter 'g'") +
  theme_ipsum_rc(grid="Y") +
  theme(axis.text.y=element_blank())

## End(Not run)
```

Description

You should `import_public_sans()` first and also install the fonts on your system before trying to use this theme.
Usage

```r
theme_ipsum_pub(
  base_family = "Public Sans",
  base_size = 10.5,
  plot_title_family = if (.Platform$OS.type == "windows") "Public Sans" else "Public Sans Bold",
  plot_title_size = 18,
  plot_title_face = "bold",
  plot_title_margin = 10,
  subtitle_family = if (.Platform$OS.type == "windows") "Public Sans Thin" else "Public Sans Thin",
  subtitle_size = 13,
  subtitle_face = "plain",
  subtitle_margin = 15,
  strip_text_family = base_family,
  strip_text_size = 12,
  strip_text_face = "plain",
  caption_family = if (.Platform$OS.type == "windows") "Public Sans Thin" else "Public Sans Thin",
  caption_size = 9,
  caption_face = "plain",
  caption_margin = 10,
  axis_text_size = base_size,
  axis_title_family = base_family,
  axis_title_size = 9,
  axis_title_face = "plain",
  axis_title_just = "rt",
  plot_margin = margin(30, 30, 30, 30),
  grid_col = ".ccc",
  grid = TRUE,
  axis_col = ".ccc",
  axis = FALSE,
  ticks = FALSE
)
```

Arguments

- **base_family, base_size**
  - base font family and size
- **plot_title_family, plot_title_face, plot_title_size, plot_title_margin**
  - plot title family, face, size and margin
- **subtitle_family, subtitle_face, subtitle_size**
  - plot subtitle family, face and size
- **subtitle_margin**
  - plot subtitle margin bottom (single numeric value)
- **strip_text_family, strip_text_face, strip_text_size**
  - facet label font family, face and size
caption_family, caption_face, caption_size, caption_margin
  plot caption family, face, size and margin
axis_text_size  font size of axis text
axis_title_family, axis_title_face, axis_title_size
  axis title font family, face and size
axis_title_just
  axis title font justification (one of [blmcrt])
plot_margin  plot margin (specify with ggplot2::margin)
grid_col  grid color
grid  panel grid (TRUE, FALSE, or a combination of X, x, Y, y)
axis_col  axis color
axis  add x or y axes? TRUE, FALSE, "xy"
ticks  ticks if TRUE add ticks

Details

There is an option hrbrthemes.loadfonts which – if set to TRUE – will call extrafont::loadfonts() to register non-core fonts with R PDF & PostScript devices. If you are running under Windows, the package calls the same function to register non-core fonts with the Windows graphics device.

Why Public Sans?

See the design principles.

Examples

## Not run:
library(ggplot2)
library(dplyr)

# seminal scatterplot
ggplot(mtcars, aes(mpg, wt)) +
  geom_point() +
  labs(x="Fuel efficiency (mpg)", y="Weight (tons)",
       title="Seminal ggplot2 scatterplot example",
       subtitle="A plot that is only useful for demonstration purposes",
       caption="Brought to you by the letter 'g'") +
  theme_ipsum_pub()

# seminal bar chart
update_geom_font_defaults(family=font_pub)

count(mpg, class) %>%
  ggplot(aes(class, n)) +
  geom_col() +
  geom_text(aes(label=n), nudge_y=3) +
  labs(x="Fuel efficiency (mpg)", y="Weight (tons)",
       title="Seminal ggplot2 bar chart example",
       subtitle=""
theme_ipsum_tw

A precise & pristine ggplot2 theme with opinionated defaults and an emphasis on typography

Description

You should import_titillium_web() first and also install the fonts on your system before trying to use this theme.

Usage

theme_ipsum_tw(
  base_family = "Titillium Web",
  base_size = 10.5,
  plot_title_family = if (.Platform$OS.type == "windows") "Titillium Web" else "Titillium Web Bold",
  plot_title_size = 18,
  plot_title_face = "bold",
  plot_title_margin = 10,
  subtitle_family = if (.Platform$OS.type == "windows") "Titillium Web" else "Titillium Web Light",
  subtitle_size = 13,
  subtitle_face = "plain",
  subtitle_margin = 15,
  strip_text_family = base_family,
  strip_text_size = 12,
  strip_text_face = "plain",
  caption_family = if (.Platform$OS.type == "windows") "Titillium Web" else "Titillium Web Light",
  caption_size = 9,
  caption_face = "plain",
  caption_margin = 10,
  axis_text_size = base_size,
  axis_title_family = base_family,
  axis_title_size = 9,
  axis_title_face = "plain",
  axis_title_just = "rt",
  plot_margin = margin(30, 30, 30, 30),
  grid_col = "cccccc",
  grid = TRUE,
)
theme_ipsum_tw

```r
axis_col = "#cccccc",
axis = FALSE,
ticks = FALSE
)
```

**Arguments**

- `base_family, base_size`: base font family and size
- `plot_title_family, plot_title_face, plot_title_size, plot_title_margin`: plot title family, face, size and margin
- `subtitle_family, subtitle_face, subtitle_size`: plot subtitle family, face and size
- `subtitle_margin`: plot subtitle margin bottom (single numeric value)
- `strip_text_family, strip_text_face, strip_text_size`: facet label font family, face and size
- `caption_family, caption_face, caption_size, caption_margin`: plot caption family, face, size and margin
- `axis_text_size`: font size of axis text
- `axis_title_family, axis_title_face, axis_title_size`: axis title font family, face and size
- `axis_title_just`: axis title font justification, one of `blmcrt`
- `plot_margin`: plot margin (specify with `ggplot2::margin`)
- `grid_col`: grid color
- `grid`: panel grid (TRUE, FALSE, or a combination of X, x, Y, y)
- `axis_col`: axis color
- `axis`: add x or y axes? TRUE, FALSE, "xy"
- `ticks`: ticks if TRUE add ticks

**Details**

There is an option `hrbrthemes.loadfonts` which – if set to TRUE – will call `extrafont::loadfonts()` to register non-core fonts with R PDF & PostScript devices. If you are running under Windows, the package calls the same function to register non-core fonts with the Windows graphics device.

**Why Titillium Web?**

It’s free, has tolerable kerning pairs and multiple weights. It’s also different than Arial Narrow and the fonts most folks use in ggplot2 charts.
Examples

## Not run:
library(ggplot2)
library(dplyr)

# seminal scatterplot
ggplot(mtcars, aes(mpg, wt)) +
  geom_point() +
  labs(x="Fuel efficiency (mpg)", y="Weight (tons)",
       title="Seminal ggplot2 scatterplot example",
       subtitle="A plot that is only useful for demonstration purposes",
       caption="Brought to you by the letter 'g'") +
  theme_ipsum_rc()

# seminal bar chart
# note: make this font_rc on Windows
update_geom_font_defaults(family=font_rc_light)

count(mpg, class) %>%
  ggplot(aes(class, n)) +
  geom_col() +
  geom_text(aes(label=n), nudge_y=3) +
  labs(x="Fuel efficiency (mpg)", y="Weight (tons)",
       title="Seminal ggplot2 bar chart example",
       subtitle="A plot that is only useful for demonstration purposes",
       caption="Brought to you by the letter 'g'") +
  theme_ipsum_tw(grid="Y") +
  theme(axis.text.y=element_blank())

## End(Not run)

theme_tinyhand

Something you should never use.

Description

You should `import_tinyhand()` first and also install the fonts on your system before trying to use this theme.

Usage

theme_tinyhand(
  base_family = font_th,
  base_size = 10.5,
  plot_title_family = font_th,
  plot_title_size = 18,
  plot_title_face = "bold",
  plot_title_margin = 10,
```r
subtitle_family = \texttt{font\_th},
subtitle_size = 13,
subtitle_face = \texttt{"plain"},
subtitle_margin = 15,
strip_text_family = \texttt{base\_family},
strip_text_size = 12,
strip_text_face = \texttt{"plain"},
caption_family = \texttt{font\_th},
caption_size = 9,
caption_face = \texttt{"plain"},
caption_margin = 10,
axis_text_size = \texttt{base\_size},
axis_title_family = \texttt{base\_family},
axis_title_size = 9,
axis_title_face = \texttt{"plain"},
axis_title_just = \texttt{"rt"},
plot_margin = \texttt{margin(30, 30, 30, 30)},
grid_col = \texttt{"#cccccc"},
grid = \texttt{TRUE},
axis_col = \texttt{"#cccccc"},
axis = \texttt{FALSE},
ticks = \texttt{FALSE}
```

Arguments

- `base_family`, `base_size`
  - base font family and size
- `plot_title_family`, `plot_title_face`, `plot_title_size`, `plot_title_margin`
  - plot title family, face, size and margin
- `subtitle_family`, `subtitle_face`, `subtitle_size`
  - plot subtitle family, face and size
- `subtitle_margin`
  - plot subtitle margin bottom (single numeric value)
- `strip_text_family`, `strip_text_face`, `strip_text_size`
  - facet label font family, face and size
- `caption_family`, `caption_face`, `caption_size`, `caption_margin`
  - plot caption family, face, size and margin
- `axis_text_size`
  - font size of axis text
- `axis_title_family`, `axis_title_face`, `axis_title_size`
  - axis title font family, face and size
- `axis_title_just`
  - axis title font justificationk one of [blmcrt]
- `plot_margin`
  - plot margin (specify with `ggplot2::margin`)
- `grid_col`
  - grid color
- `grid`
  - panel grid (TRUE, FALSE, or a combination of X, x, Y, y)
update_geom_font_defaults

axis_col  axis color
axis      add x or y axes? TRUE, FALSE, "xy"
ticks     ticks if TRUE add ticks

Details

There is an option hrbrthemes.loadfonts which – if set to TRUE – will call extrafont::loadfonts() to register non-core fonts with R PDF & PostScript devices. If you are running under Windows, the package calls the same function to register non-core fonts with the Windows graphics device.

Why Titillium Web?

It’s free, has tolerable kerning pairs and multiple weights. It’s also different than Arial Narrow and the fonts most folks use in ggplot2 charts.

update_geom_font_defaults

Update matching font defaults for text geoms

Description

Updates [ggplot2::geom_label] and [ggplot2::geom_text] font defaults

Usage

update_geom_font_defaults(
  family = "Arial Narrow",
  face = "plain",
  size = 3.5,
  color = "#2b2b2b"
)

Arguments

family, face, size, color
  font family name, face, size and color
Index

* datasets
  - font_an, 3
  - font_es, 4
  - font_ps, 4
  - font_rc, 5
  - font_th, 6
  - font_tw, 6
  - ft_cols, 7

%>% (hrbrthemes-exports), 9

- flush_ticks, 3
- font_an, 3
- font_es, 4
- font_es_bold (font_es), 4
- font_es_light (font_es), 4
- font_ps, 4
- font_ps_light (font_ps), 4
- font_pub, 5
- font_pub_bold (font_pub), 5
- font_pub_light (font_pub), 5
- font_pub_thin (font_pub), 5
- font_rc, 5
- font_rc_light (font_rc), 5
- font_th, 6
- font_tw, 6
- font_tw_bold (font_tw), 6
- font_tw_light (font_tw), 6
- ft_cols, 7
- ft_geom_defaults, 7
- ft_pal, 8
- ft_pal(), 15
- ft_text_col (ft_cols), 7

- ggplot2, 8
- ggplot2:::discrete_scale, 15, 16
- ggplot2:::margin, 23, 28, 30, 33, 35, 37
- guides(), 16, 17, 20
- hrbrthemes-exports, 9
- hunspell::dictionary, 8
- import_econ_sans, 9
- import_econ_sans(), 27
- import_roboto_condensed(), 11
- import_roboto_condensed(), 20
- import_tinyhand, 11
- import_tinyhand(), 36
- import_titillium_web, 12
- import_titillium_web(), 34
- includes, 13
- ipsum, 12
- ipsum_pal, 14
- ipsum_pal(), 16
- ipsum_pdf, 14
- modern_geom_defaults, 15
- render, 14
- rmarkdown_format, 13
- scale_color_ft (scale_colour_ft), 15
- scale_color_ipsum (scale_colour_ipsum), 16
- scale_colour_ft, 15
- scale_colour_ipsum, 16
- scale_fill_ft (scale_colour_ft), 15
- scale_fill_ipsum (scale_colour_ipsum), 16
- scale_x_comma (scale_x_percent), 17
- scale_x_percent, 17
- scale_y_comma (scale_y_percent), 17
- scale_y_percent (scale_y_percent), 17
- scales::alpha(), 7
- scales::hue_pal(), 15, 16
theme_ft_rc, 20
theme_ipsum, 24
theme_ipsum_es, 27
theme_ipsum_ps, 29
theme_ipsum_pub, 31
theme_ipsum_rc (theme_ft_rc), 20
theme_ipsum_tw, 34
theme_modern_rc (theme_ft_rc), 20
theme_tinyhand, 36
update_geom_font_defaults, 38