

# Package ‘hrbrthemes’

August 19, 2018

**Type** Package

**Title** Additional Themes, Theme Components and Utilities for 'ggplot2'

**Version** 0.5.0.1

**Date** 2018-04-24

**Maintainer** ORPHANED

**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/>> is also included along with a copy of the 'IBM' 'Plex Sans' <<https://github.com/IBM/type/>> and 'Titillium Web' <<https://fonts.google.com/specimen/Titillium+Web>> 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, gdtools, purrr

**Depends** R (>= 3.2.0)

**Imports** ggplot2 (>= 2.2.1), grDevices, grid, scales, extrafont, knitr, rmarkdown, htmltools, tools, magrittr

**RoxygenNote** 6.0.1.9000

**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),  
 Jacob Barnett [ctb],  
 Thomas J. Leeper [ctb] (<<https://orcid.org/0000-0003-4097-6326>>)

**Repository** CRAN

**Date/Publication** 2018-08-19 08:51:32 UTC

**X-CRAN-Original-Maintainer** Bob Rudis <[bob@rud.is](mailto:bob@rud.is)>

**X-CRAN-Comment** Orphaned and corrected on 2018-08-19 as check problems were not corrected despite reminders.

## R topics documented:

flush_ticks . . . . .	2
font_an . . . . .	3
font_ps . . . . .	4
font_rc . . . . .	4
font_tw . . . . .	5
gg_check . . . . .	5
hrbrthemes . . . . .	6
hrbrthemes-exports . . . . .	7
import_plex_sans . . . . .	7
import_roboto_condensed . . . . .	7
import_titillium_web . . . . .	8
ipsum . . . . .	8
ipsum_pal . . . . .	10
ipsum_pdf . . . . .	10
modern_geom_defaults . . . . .	11
scale_colour_ipsum . . . . .	11
scale_x_percent . . . . .	12
theme_ipsum . . . . .	14
theme_ipsum_ps . . . . .	16
theme_ipsum_tw . . . . .	18
theme_modern_rc . . . . .	20
update_geom_font_defaults . . . . .	22
<b>Index</b>	<b>23</b>

---

flush_ticks	<i>Makes axis text labels flush on the ends</i>
-------------	---

---

### 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, coords, scales, themes, etc. It also builds the plot (but does not display it) so if the plot 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 <i>invisibly</i>
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	<i>Arial Narrow font name R variable aliases</i>
---------	--

---

**Description**

```
font_an == "Arial Narrow"
```

**Usage**

```
font_an
```

**Format**

length 1 character vector

---

font_ps	<i>PlexSans font name R variable aliases</i>
---------	--

---

**Description**

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

**Usage**

```
font_ps  
  
font_ps_light
```

**Format**

length 1 character vector

**Note**

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

---

font_rc	<i>Roboto Condensed font name R variable aliases</i>
---------	--

---

**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.

---

font_tw	<i>Titillium Web font name R variable aliases</i>
---------	---

---

**Description**

```
font_tw == "Titillium Web"
font_tw_light == "Titillium Web Bold"
font_tw_light == "Titillium Web Light"
```

**Usage**

```
font_tw
font_tw_bold
font_tw_light
```

**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.

---

gg_check	<i>Spell check ggplot2 plot labels</i>
----------	--

---

**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 <a href="#">hunspell::dictionary</a> . Defaults to <code>hunspell::dictionary("en_US")</code>
ignore	character vector with additional approved words added to the dictionary. Defaults to <code>hunspell::en_stats</code>

**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**

```
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

---

*Additional Themes and Theme Components for 'ggplot2'*


---

**Description**

A compilation of extra themes and theme components for 'ggplot2' with an emphasis on typography.

**Details**

The core theme: `theme_ipsum` ("ipsum" is Latin for "precise") uses Arial Narrow which should be installed on practically any modern system, so it's "free"-ish. This font is condensed, has solid default kerning pairs and geometric numbers. That's what I consider the "font trifecta" must-have for charts. An additional quality for fonts for charts is that they have a diversity of weights. Arial Narrow (the one on most systems, anyway) does not have said diversity but this quality is not (IMO) a "must have".

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.

**Author(s)**

Bob Rudis (bob@rud.is)

---

hrbrthemes-exports	<i>hrbrthemes exported operators</i>
--------------------	--------------------------------------

---

### 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

---

import_plex_sans	<i>Import IBM Plex Sans font for use in charts</i>
------------------	--

---

### Description

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

### Usage

```
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.

---

import_roboto_condensed	<i>Import Roboto Condensed font for use in charts</i>
-------------------------	---

---

### Description

Roboto Condensed is a trademark of Google.

### Usage

```
import_roboto_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_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**

<code>number_sections</code>	TRUE to number section headings
<code>fig_width</code>	Default width (in inches) for figures
<code>fig_height</code>	Default width (in inches) for figures
<code>fig_retina</code>	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 <code>keep_md</code> is specified (this is because <code>fig_retina</code> relies on outputting HTML directly into the markdown document).
<code>fig_caption</code>	TRUE to render figures with captions
<code>dev</code>	Graphics device to use for figure output (defaults to png)
<code>smart</code>	Produce typographically correct output, converting straight quotes to curly quotes, — to em-dashes, – to en-dashes, and ... to ellipses.
<code>self_contained</code>	Produce a standalone HTML file with no external dependencies, using data: URIs 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 it's size).
<code>highlight</code>	Syntax highlighting style. Supported styles include "default", "tango", "pygments", "kate", "monochrome", "espresso", "zenburn", "haddock", and "textmate". Pass NULL to prevent syntax highlighting.
<code>mathjax</code>	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.
<code>extra_dependencies, ...</code>	Additional function arguments to pass to the base R Markdown HTML output formatter
<code>css</code>	One or more css files to include
<code>includes</code>	Named list of additional content to include within the document (typically created using the <code>includes</code> function).
<code>keep_md</code>	Keep the markdown file generated by knitting.
<code>lib_dir</code>	Directory to copy dependent HTML libraries (e.g. jquery, bootstrap, etc.) into. By default this will be the name of the document with <code>_files</code> 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\_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

**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

**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 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 info.

**super** The super class to use for the constructed scale

---

scale_x_percent	<i>X &amp; Y scales with opinionated pre-sets for percent &amp; comma label formats</i>
-----------------	---

---

## Description

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

## Usage

```
scale_x_percent(name = waiver(), breaks = waiver(),
  minor_breaks = waiver(), labels = scales::percent, limits = NULL,
  expand = c(0.01, 0), oob = censor, na.value = NA_real_,
  trans = "identity", position = "bottom", sec.axis = waiver())
```

```
scale_y_percent(name = waiver(), breaks = waiver(),
  minor_breaks = waiver(), labels = scales::percent, limits = NULL,
  expand = c(0.01, 0), oob = censor, na.value = NA_real_,
  trans = "identity", position = "left", sec.axis = waiver())
```

```
scale_x_comma(name = waiver(), breaks = waiver(), minor_breaks = waiver(),
  labels = scales::comma, limits = NULL, expand = c(0.01, 0),
  oob = censor, na.value = NA_real_, trans = "identity",
  position = "bottom", sec.axis = waiver())
```

```
scale_y_comma(name = waiver(), breaks = waiver(), minor_breaks = waiver(),
  labels = scales::comma, limits = NULL, expand = c(0.01, 0),
  oob = censor, na.value = NA_real_, trans = "identity",
  position = "left", sec.axis = waiver())
```

**Arguments**

name	The name of the scale. Used as axis or legend title. If <code>waiver()</code> , the default, the name of the scale is taken from the first mapping used for that aesthetic. If <code>NULL</code> , the legend title will be omitted.
breaks	One of: <ul style="list-style-type: none"> <li>• <code>NULL</code> for no breaks</li> <li>• <code>waiver()</code> for the default breaks computed by the transformation object</li> <li>• A numeric vector of positions</li> <li>• A function that takes the limits as input and returns breaks as output</li> </ul>
minor_breaks	One of: <ul style="list-style-type: none"> <li>• <code>NULL</code> for no minor breaks</li> <li>• <code>waiver()</code> for the default breaks (one minor break between each major break)</li> <li>• A numeric vector of positions</li> <li>• A function that given the limits returns a vector of minor breaks.</li> </ul>
labels	One of: <ul style="list-style-type: none"> <li>• <code>NULL</code> for no labels</li> <li>• <code>waiver()</code> for the default labels computed by the transformation object</li> <li>• A character vector giving labels (must be same length as breaks)</li> <li>• A function that takes the breaks as input and returns labels as output</li> </ul>
limits	A numeric vector of length two providing limits of the scale. Use <code>NA</code> to refer to the existing minimum or maximum.
expand	same as in <code>ggplot2</code>
oob	Function that handles limits outside of the scale limits (out of bounds). The default replaces out of bounds values with <code>NA</code> .
na.value	If <code>na.translate = TRUE</code> , what value aesthetic value should missing be displayed as? Does not apply to position scales where <code>NA</code> 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

**Details**

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

---

theme_ipsum	<i>A precise &amp; pristine <a href="#">ggplot2</a> theme with opinionated defaults and an emphasis on typography</i>
-------------	---

---

## Description

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

## Usage

```
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", 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 [blmcr]
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 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()

# 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_ps	<i>A precise &amp; pristine <a href="#">ggplot2</a> theme with opinionated defaults and an emphasis on typography</i>
----------------	---

---

## 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$OS.type == "windows") "IBMPlexSans" else "IBMPlexSans-Light",
  subtitle_size = 13, subtitle_face = "plain", subtitle_margin = 15,
  strip_text_family = "IBMPlexSans-Medium", strip_text_size = 12,
  strip_text_face = "plain", caption_family = if (.Platform$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", 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 justificationk one of [blmcr]t]
plot_margin	plot margin (specify with <code>ggplot2::margin</code> )
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

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

# seminal scatterplot
ggplot(mtcars, aes(mpg, wt)) +
  geom_point() +
  labs(x="Fuel effiency (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 effiency (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)

```

---

theme_ipsum_tw	<i>A precise &amp; pristine <a href="#">ggplot2</a> theme with opinionated defaults and an emphasis on typography</i>
----------------	---

---

## 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,
  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 [blmcr]t]
plot_margin	plot margin (specify with <code>ggplot2::margin</code> )
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 effiency (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 effiency (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_modern_rc	<i>A precise &amp; pristine <a href="#">ggplot2</a> theme with opinionated defaults and an emphasis on typography</i>
-----------------	---

---

## Description

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

## Usage

```

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_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), 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 [blmcr]t  
 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  
 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**

```
## 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)

```

---

```
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

## \*Topic **datasets**

- font\_an, [3](#)
- font\_ps, [4](#)
- font\_rc, [4](#)
- font\_tw, [5](#)
- %>% (hrbrthemes-exports), [7](#)
- flush\_ticks, [2](#)
- font\_an, [3](#)
- font\_ps, [4](#)
- font\_ps\_light (font\_ps), [4](#)
- font\_rc, [4](#)
- font\_rc\_light (font\_rc), [4](#)
- font\_tw, [5](#)
- font\_tw\_bold (font\_tw), [5](#)
- font\_tw\_light (font\_tw), [5](#)
- gg\_check, [5](#)
- ggplot2, [14](#), [16](#), [18](#), [20](#)
- ggplot2::margin, [14](#), [17](#), [19](#), [21](#)
- guides(), [12](#)
- hrbrthemes, [6](#)
- hrbrthemes-exports, [7](#)
- hrbrthemes-package (hrbrthemes), [6](#)
- hunspell::dictionary, [5](#)
- import\_plex\_sans, [7](#)
- import\_roboto\_condensed, [7](#)
- import\_titillium\_web, [8](#)
- includes, [9](#)
- ipsum, [8](#)
- ipsum\_pal, [10](#)
- ipsum\_pdf, [10](#)
- modern\_geom\_defaults, [11](#)
- render, [10](#)
- rmarkdown\_format, [10](#)
- scale\_color\_ipsum (scale\_colour\_ipsum), [11](#)
- scale\_colour\_ipsum, [11](#)
- scale\_fill\_ipsum (scale\_colour\_ipsum), [11](#)
- scale\_x\_comma (scale\_x\_percent), [12](#)
- scale\_x\_percent, [12](#)
- scale\_y\_comma (scale\_x\_percent), [12](#)
- scale\_y\_percent (scale\_x\_percent), [12](#)
- theme\_ipsum, [14](#)
- theme\_ipsum\_ps, [16](#)
- theme\_ipsum\_rc (theme\_modern\_rc), [20](#)
- theme\_ipsum\_tw, [18](#)
- theme\_modern\_rc, [20](#)
- update\_geom\_font\_defaults, [22](#)