

Package: urbnthemes (via r-universe)

September 4, 2024

Type Package

Title Additional theme and utilities for ``ggplot2`` in the Urban Institute style

Version 0.0.2

Description Align ``ggplot2`` output more closely with the Urban Institute Data Visualization style guide
<<https://urbaninstitute.github.io/r-at-urban/graphics-guide.html>>.

Depends R (>= 3.1.0)

Imports ggplot2 (>= 3.3.0), extrafont, ggrepel, grid, gridExtra, lifecycle, scales, conflicted, tibble, purrr, stringr, systemfonts

License GPL-3

URL <https://github.com/UrbanInstitute/urbnthemes>

BugReports <https://github.com/UrbanInstitute/urbnthemes/issues>

Encoding UTF-8

LazyData true

RoxygenNote 7.1.2

Suggests knitr, rmarkdown, testthat

VignetteBuilder knitr

Roxygen list(markdown = TRUE)

Repository <https://urbaninstitute.r-universe.dev>

RemoteUrl <https://github.com/UrbanInstitute/urbnthemes>

RemoteRef HEAD

RemoteSha f6a368d6018e128238e9f3cfcaf1f6c4acb62613

Contents

add_axis	2
fontawesome_install	3

fontawesome_test	3
geom_bar	4
geom_col	4
get_legend	5
lato_import	5
lato_install	6
lato_test	6
palette_urban	6
palette_urban_main	7
remove_axis	8
remove_legend	8
remove_ticks	9
scale_color_discrete	9
scale_color_gradientn	10
scale_color_ordinal	10
scale_colour_discrete	11
scale_colour_gradientn	11
scale_colour_ordinal	12
scale_fill_discrete	13
scale_fill_gradientn	13
scale_fill_ordinal	14
scatter_grid	14
set_urban_defaults	15
theme_urban_map	15
theme_urban_print	16
undo_urban_defaults	16
urban_color_pal	17
urban_geofacet	17
urban_logo_text	17
urban_note	18
urban_plot	18
urban_save	19
urban_source	20
urban_subtitle	20
urban_title	21
urban_y_title	21
view_palette	22

Index **23**

add_axis *The Urban Institute ggplot2 theme*

Description

Adds tick axes and axes labels

Usage

```
add_axis(axis = "y")
```

Arguments

axis Axis to drop. "x", "y", or "both"

fontawesome_install *Import and register Lato font*

Description

fontawesome_install tests to see if FontAwesome is imported and registered. If FontAwesome, isn't imported and registered, then fontawesome_install imports and registers FontAwesome with `library(extrafont)`.

Usage

```
fontawesome_install()
```

Details

Note: FontAwesome must be installed on your computer for fontawesome_install to work.

Test to see if FontAwesome is imported and registered with fontawesome_test().

fontawesome_test *Test for FontAwesome import and registration*

Description

fontawesome_test tests to see if FontAwesome is imported and registered. FontAwesome is used for glyphs with `library(waffle)` and can be installed from [Bob Rudis's GitHub](#).

Usage

```
fontawesome_test()
```

Details

Import and register FontAwesome in R with fontawesome_install().

geom_bar

geom_bar in the Urban Institute style

Description

Submit `?ggplot2::geom_line` to see the full documentation for `geom_bar()`

Usage

```
geom_bar(mapping = NULL, width = 0.7, ...)
```

Arguments

mapping	mapping from ggplot2
width	bar width
...	other arguments passed to <code>geom_bar()</code>

geom_col

geom_col in the Urban Institute style

Description

Submit `?ggplot2::geom_line` to see the full documentation for `geom_col()`

Usage

```
geom_col(mapping = NULL, width = 0.7, ...)
```

Arguments

mapping	mapping from ggplot2
width	column width
...	other arguments passed to <code>geom_col()</code>

`get_legend`*get_legend*

Description

Extract just the legend from a ggplot2 object.

Usage

```
get_legend(ggplot_object)
```

Arguments

`ggplot_object` a ggplot

Value

a ggplot legend

`lato_import`*Import and register Lato font*

Description

`lato_import()` tests to see if Lato is imported and registered. If Lato, isn't imported and registered, then `lato_import()` imports and registers Lato with `library(extrafont)`.

Usage

```
lato_import()
```

Details

Note: Lato must be installed on your computer for `lato_import()` to work. Lato is the Urban Institute's main font. To install, visit [Google fonts](#) and click "Download family". Unzip and open each of the .ttf files and click install.

Test to see if Lato is imported and registered with `lato_test()`.

lato_install	<i>Import and register Lato font</i>
--------------	--------------------------------------

Description**[Deprecated]**

lato_install() was deprecated because its name was ambiguous. Please use lato_import().

Usage

```
lato_install()
```

lato_test	<i>Test for Lato import and registration</i>
-----------	--

Description

lato_test tests to see if Lato is imported and registered. Lato is the Urban Institute's main font and can be installed from [Google fonts](#).

Usage

```
lato_test()
```

Details

Import and register Lato in R with lato_install().

palette_urban	<i>The Urban Institute ggplot2 theme</i>
---------------	--

Description

Palette data for the Urban Institute

Usage

```
palette_urban
```

Format

A list.

Details

Data used by the palettes in the uithemes package.

Source

Urban Institute Data Visualization Style Guide <http://urbaninstitute.github.io/graphics-styleguide/>

remove_axis	<i>The Urban Institute ggplot2 theme</i>
-------------	--

Description

Removes axes and axes labels

Usage

```
remove_axis(axis = "y", flip = FALSE)
```

Arguments

axis	Axis to drop. "x", "y", or "both"
flip	TRUE/FALSE is coord_flip() is used

remove_legend	<i>remove_legend</i>
---------------	----------------------

Description

Remove just the legend from a ggplot2 object.

Usage

```
remove_legend(ggplot_object)
```

Arguments

ggplot_object	a ggplot
---------------	----------

Value

a ggplot without a legend

remove_ticks	<i>The Urban Institute ggplot2 theme</i>
--------------	--

Description

Removes tick marks from all axes

Usage

```
remove_ticks(axis = "both")
```

Arguments

axis Axis to drop. "x", "y", or "both"

scale_color_discrete	<i>Discrete color scale that aligns with the Urban Institute style</i>
----------------------	--

Description

This function can only handle up to 8 categories/colors.

Usage

```
scale_color_discrete(...)
```

Arguments

... other arguments passed to `discrete_scale()`

Details

If you need more than 8 colors for exploratory purposes, use `ggplot2::scale_fill_discrete()`.

scale_color_gradientn *Continuous fill scale that aligns with the Urban Institute style*

Description

Continuous fill scale that aligns with the Urban Institute style

Usage

```
scale_color_gradientn(
  ...,
  colours = c("#CFE8F3", "#A2D4EC", "#73BFE2", "#46ABDB", "#1696D2", "#12719E",
             "#0A4C6A", "#062635"),
  colors = c("#CFE8F3", "#A2D4EC", "#73BFE2", "#46ABDB", "#1696D2", "#12719E",
             "#0A4C6A", "#062635"),
  values = NULL,
  space = "Lab",
  na.value = "grey50",
  guide = "colourbar"
)
```

Arguments

...	other arguments passed to discrete_scale()
colours	vector of colours
colors	vector of colours
values	if colours should not be evenly positioned along the gradient this vector gives the position (between 0 and 1) for each colour in the colours vector. See rescale for a convenience function to map an arbitrary range to between 0 and 1
space	colour space in which to calculate gradient. Must be "Lab" - other values are deprecated.
na.value	default color for NA values
guide	legend representation of scale

scale_color_ordinal *Discrete color scale for ordinal factors that aligns with the Urban Institute style*

Description

This function can only handle up to 8 categories/colors.

Usage

```
scale_color_ordinal(...)
```

Arguments

```
...          other arguments passed to discrete_scale()
```

Details

If you need more than 8 colors for exploratory purposes, use `ggplot2::scale_color_ordinal()`.

scale_colour_discrete *Discrete color scale that aligns with the Urban Institute style*

Description

This function can only handle up to 8 categories/colors.

Usage

```
scale_colour_discrete(...)
```

Arguments

```
...          other arguments passed to discrete_scale()
```

Details

If you need more than 8 colors for exploratory purposes, use `ggplot2::scale_color_discrete()`.

scale_colour_gradientn *Continuous fill scale that aligns with the Urban Institute style*

Description

Continuous fill scale that aligns with the Urban Institute style

Usage

```
scale_colour_gradientn(
  ...,
  colours = c("#CFE8F3", "#A2D4EC", "#73BFE2", "#46ABDB", "#1696D2", "#12719E",
             "#0A4C6A", "#062635"),
  colors = c("#CFE8F3", "#A2D4EC", "#73BFE2", "#46ABDB", "#1696D2", "#12719E",
             "#0A4C6A", "#062635"),
  values = NULL,
  space = "Lab",
  na.value = "grey50",
  guide = "colourbar"
)
```

Arguments

...	other arguments passed to <code>discrete_scale()</code>
colours	vector of colours
colors	vector of colours
values	if colours should not be evenly positioned along the gradient this vector gives the position (between 0 and 1) for each colour in the colours vector. See <code>rescale</code> for a convenience function to map an arbitrary range to between 0 and 1
space	colour space in which to calculate gradient. Must be "Lab" - other values are deprecated.
na.value	default color for NA values
guide	legend representation of scale

scale_colour_ordinal *Discrete color scale for ordinal factors that aligns with the Urban Institute style*

Description

This function can only handle up to 8 categories/colors.

Usage

```
scale_colour_ordinal(...)
```

Arguments

...	other arguments passed to <code>discrete_scale()</code>
-----	---

Details

If you need more than 8 colors for exploratory purposes, use `ggplot2::scale_colour_ordinal()`.

scale_fill_discrete *Discrete fill scale that aligns with the Urban Institute style*

Description

This function can only handle up to 8 categories/colors.

Usage

```
scale_fill_discrete(...)
```

Arguments

... other arguments passed to `discrete_scale()`

Details

If you need more than 8 colors for exploratory purposes, use `ggplot2::scale_fill_discrete()`.

scale_fill_gradientn *Continuous fill scale that aligns with the Urban Institute style*

Description

Continuous fill scale that aligns with the Urban Institute style

Usage

```
scale_fill_gradientn(  
  ...,  
  colours = c("#CFE8F3", "#A2D4EC", "#73BFE2", "#46ABDB", "#1696D2", "#12719E",  
             "#0A4C6A", "#062635"),  
  colors = c("#CFE8F3", "#A2D4EC", "#73BFE2", "#46ABDB", "#1696D2", "#12719E",  
            "#0A4C6A", "#062635"),  
  values = NULL,  
  space = "Lab",  
  na.value = "grey50",  
  guide = "colourbar"  
)
```

Arguments

...	other arguments passed to <code>discrete_scale()</code>
colours	vector of colours
colors	vector of colours
values	if colours should not be evenly positioned along the gradient this vector gives the position (between 0 and 1) for each colour in the colours vector. See <code>rescale</code> for a convenience function to map an arbitrary range to between 0 and 1
space	colour space in which to calculate gradient. Must be "Lab" - other values are deprecated.
na.value	default color for NA values
guide	legend representation of scale

<code>scale_fill_ordinal</code>	<i>Discrete fill scale for ordinal factors that aligns with the Urban Institute style</i>
---------------------------------	---

Description

This function can only handle up to 8 categories/colors.

Usage

```
scale_fill_ordinal(...)
```

Arguments

...	other arguments passed to <code>discrete_scale()</code>
-----	---

Details

If you need more than 8 colors for exploratory purposes, use `ggplot2::scale_fill_ordinal()`.

<code>scatter_grid</code>	<i>The Urban Institute ggplot2 theme</i>
---------------------------	--

Description

Adds vertical grid lines to plots for scatter plots. This is useful for scatter plots because `urbnthemes` only supplies horizontal grid lines.

Usage

```
scatter_grid()
```

set_urban_defaults	<i>The Urban Institute ggplot2 theme</i>
--------------------	--

Description

set_urban_defaults provides a [ggplot2](#) theme formatted according to the Urban Institute style guide, with sensible defaults.

Usage

```
set_urban_defaults(  
  style = "print",  
  base_size = 8.5,  
  base_family = "Lato",  
  base_line_size = 0.5,  
  base_rect_size = 0.5,  
  scale = "continuous"  
)
```

Arguments

style	The default theme style for the R session. "print" or "map".
base_size	The base font size for the theme. All fonts are relative to this value.
base_family	The base font family for the theme.
base_line_size	The base line size for the theme. All line sizes are relative to this value.
base_rect_size	The base rect size for the theme. All rect sizes are relative to this value.
scale	For theme_urban_map(). Should the legend theme be continuous or discrete?

theme_urban_map	<i>A ggplot2 theme formatted in the Urban Institute style</i>
-----------------	---

Description

theme_urban provides a [ggplot2](#) theme formatted according to the Urban Institute style guide for maps, with sensible defaults.

Usage

```
theme_urban_map(  
  scale = "continuous",  
  base_size = 8.5,  
  base_family = "Lato",  
  base_line_size = 0.5,  
  base_rect_size = 0.5  
)
```

Arguments

scale "continuous" creates a vertical legend to the right of the map. "discrete" creates a horizontal legend above the map.

base_family, base_size
base font family and size

base_line_size, base_rect_size
base line and rectangle sizes

theme_urban_print *A [ggplot2](#) theme formatted in the Urban Institute style*

Description

theme_urban provides a [ggplot2](#) theme formatted according to the Urban Institute style guide for web, with sensible defaults.

Usage

```
theme_urban_print(
  base_size = 8.5,
  base_family = "Lato",
  base_line_size = 0.5,
  base_rect_size = 0.5
)
```

Arguments

base_family, base_size
base font family and size

base_line_size, base_rect_size
base line and rectangle sizes

undo_urban_defaults *The Urban Institute [ggplot2](#) theme*

Description

Resets all the [ggplot2](#) theme to its default settings

Usage

```
undo_urban_defaults()
```


urbn_color_pal *The Urban Institute [ggplot2](#) theme*

Description

Color palettes used in the Urban Institute.

Usage

```
urbn_color_pal(palette = "categorical")
```

Arguments

palette Palette name.

urbn_geofacet *Dataset used for creating geofaceted plots in the Urban Institute style.*

Description

Dataset used for creating geofaceted plots in the Urban Institute style.

Usage

```
urbn_geofacet
```

Format

Data frame with columns

row Row in geofacet

col Column in geofacet

code State abbreviation

name State name

urbn_logo_text *The Urban Institute [ggplot2](#) theme*

Description

Creates a grid object with the Urban Institute logo

Usage

```
urbn_logo_text()
```

urbn_note	<i>urbn_note</i>
-----------	------------------

Description

Add a notes section to the bottom of a plot created with `urbn_plot()`.

Usage

```
urbn_note(text, size = 8, width = 132, plural = FALSE)
```

Arguments

<code>text</code>	character string for a note
<code>size</code>	font size for the note
<code>width</code>	a number of characters to allow before a character return
<code>plural</code>	If 'TRUE', will change "Note:" to "Notes:"

Value

a grob formatted for a source in a ggplot

urbn_plot	<i>urbn_plot</i>
-----------	------------------

Description

Combine elements from `urbn_title`, `urbn_subtitle`, `urbn_y_title`, `get_legend`, `remove_legend`, `urbn_notes`, `urbn_source`, and `ggplot2` into one formatted plot.

Usage

```
urbn_plot(..., heights = 1)
```

Arguments

<code>...</code>	urbn plot objects or grobs
<code>heights</code>	relative heights of each object in the final plot

Value

one plot made from many grobs

`urbn_save`*Save ggplot2 plots in standard Urban Institute sizes*

Description

This is a function to save standardized images for Urban Institute publications. For more functionality like custom widths and heights, use `ggsave()` directly.

Usage

```
urbn_save(  
  filename,  
  plot = ggplot2::last_plot(),  
  size = "medium",  
  dpi = 300,  
  height = NULL  
)
```

Arguments

<code>filename</code>	File name to create on disk.
<code>plot</code>	Plot to save, defaults to last plot displayed.
<code>size</code>	Size of plot. Must be one of "small" (3.25 x 2 inches), "medium" (6.5 x 4 inches), or "large" (9 x 6.5 inches).
<code>dpi</code>	Plot resolution. Also accepts a string input: "retina" (320), "print" (300), or "screen" (72). Applies only to raster output types.

Examples

```
## Not run:  
library(tidyverse)  
library(urbnthemes)  
  
set_urban_defaults(style = "print")  
  
cars %>%  
  ggplot(aes(speed, dist)) +  
  geom_point()  
  
urbn_save("test.png")  
  
## End(Not run)
```

urbn_source	<i>urbn_source</i>
-------------	--------------------

Description

Add a source section to the bottom of a plot created with `urbn_plot()`.

Usage

```
urbn_source(text, size = 8, width = 132, plural = FALSE)
```

Arguments

<code>text</code>	character string for a source
<code>size</code>	font size for the source
<code>width</code>	a number of characters to allow before a character return
<code>plural</code>	If 'TRUE', will change "Source:" to "Sources:"

Value

a grob formatted for a source in a ggplot

urbn_subtitle	<i>urbn_subtitle</i>
---------------	----------------------

Description

Add a subtitle to a plot created with `urbn_plot()`.

Usage

```
urbn_subtitle(string, size = 9.5)
```

Arguments

<code>string</code>	character string for a subtitle
<code>size</code>	font size for the subtitle

Value

a grob formatted for a subtitle in a ggplot

urbn_title	<i>urbn_title</i>
------------	-------------------

Description

Add a title to a plot created with `urbn_plot()`.

Usage

```
urbn_title(string, size = 12, width = 68)
```

Arguments

string	character string for a title
size	font size for the title
width	a number of characters to allow before a character return

Value

a grob formatted for a source in a ggplot

urbn_y_title	<i>urbn_y_title</i>
--------------	---------------------

Description

Add a horizontal y axis title to a plot created with `urbn_plot()`.

Usage

```
urbn_y_title(string, size = 8.5)
```

Arguments

string	character string for a y-axis title
size	font size for the y-axis title

Value

a grob formatted for a y-axis title in a ggplot

view_palette	<i>The Urban Institute ggplot2 theme</i>
--------------	--

Description

view_palette displays the colors and hexadecimal codes for palette_urban_* vectors.

Usage

```
view_palette(palette = palette_urban_main)
```

Arguments

palette A palette_urban_* vector from library(urbnthemes). Options are palette_urban_diverging, palette_urban_politics, palette_urban_quintile, palette_urban_cyan, palette_urban_gray, palette_urban_yellow, palette_urban_magenta, palette_urban_green, palette_urban_spacegray, and palette_urban_red.

Examples

```
view_palette()  
view_palette(palette_urban_cyan)
```

Index

- * **datasets**
 - palette_urban, 6
 - palette_urban_main, 7
 - urban_geofacet, 17
- * **urban palettes**
 - palette_urban_main, 7
- add_axis, 2
- fontawesome_install, 3
- fontawesome_test, 3

- geom_bar, 4
- geom_col, 4
- get_legend, 5
- ggplot2, 2, 6, 8, 9, 14–17, 22

- lato_import, 5
- lato_install, 6
- lato_test, 6

- palette_urban, 6
- palette_urban_cyan (palette_urban_main), 7
- palette_urban_diverging
 - (palette_urban_main), 7
- palette_urban_gray (palette_urban_main), 7
- palette_urban_green (palette_urban_main), 7
- palette_urban_magenta
 - (palette_urban_main), 7
- palette_urban_main, 7
- palette_urban_politics
 - (palette_urban_main), 7
- palette_urban_quintile
 - (palette_urban_main), 7
- palette_urban_red (palette_urban_main), 7
- palette_urban_spacegray
 - (palette_urban_main), 7
- palette_urban_yellow
 - (palette_urban_main), 7

- remove_axis, 8
- remove_legend, 8
- remove_ticks, 9

- scale_color_discrete, 9
- scale_color_gradientn, 10
- scale_color_ordinal, 10
- scale_colour_discrete, 11
- scale_colour_gradientn, 11
- scale_colour_ordinal, 12
- scale_fill_discrete, 13
- scale_fill_gradientn, 13
- scale_fill_ordinal, 14
- scatter_grid, 14
- set_urban_defaults, 15

- theme_urban_map, 15
- theme_urban_print, 16

- undo_urban_defaults, 16
- urban_color_pal, 17
- urban_geofacet, 17
- urban_logo_text, 17
- urban_note, 18
- urban_plot, 18
- urban_save, 19
- urban_source, 20
- urban_subtitle, 20
- urban_title, 21
- urban_y_title, 21

- view_palette, 22