

Package ‘albersdown’

April 1, 2026

Type Package

Title Minimalist Theme and Vignette Kit for 'pkgdown' and R Markdown

Version 1.0.0

Description Provides a minimalist 'ggplot2' theme, colour scales, and 'pkgdown' template built around a curated colour palette system inspired by Josef Albers' colour theory (Albers (1963, ISBN:978-0-300-17935-4) ``Interaction of Color"). Includes helpers to apply consistent theming to 'ggplot2' plots, 'gt' tables, and 'bslib' Bootstrap 5 sites, along with one-command setup functions for adopting the style across an R package.

URL <https://github.com/bbuchtsbaum/albersdown>,
<https://bbuchtsbaum.github.io/albersdown/>

BugReports <https://github.com/bbuchtsbaum/albersdown/issues>

Depends R (>= 4.1)

Imports grDevices, grid, stats, ggplot2

Suggests gt, bslib, usethis, yaml, cli, testthat, viridisLite, knitr, rmarkdown, pkgdown

Encoding UTF-8

RoxygenNote 7.3.3

VignetteBuilder knitr

Config/Needs/website albersdown

License MIT + file LICENSE

Config/testthat/edition 3

NeedsCompilation no

Author Bradley R. Buchsbaum [aut, cre]

Maintainer Bradley R. Buchsbaum <brad.buchsbaum@gmail.com>

Repository CRAN

Date/Publication 2026-04-01 08:50:10 UTC

Contents

albers_bs_theme	2
albers_palette	3
albers_palette_img	4
albers_presets	4
albers_ramp	5
albers_swatch	5
gt_albers	6
migrate_albersdown	7
scale_color_albers	8
scale_color_albers_distinct	8
scale_color_albers_diverging	9
scale_color_albers_diverging_n	10
scale_color_albers_highlight	10
scale_color_albers_img	11
scale_color_albers_img_diverging	12
scale_color_albers_img_red_tea	12
scale_fill_albers_distinct	13
scale_fill_albers_diverging	14
scale_fill_albers_diverging_5	14
scale_fill_albers_diverging_n	15
scale_fill_albers_highlight	16
scale_fill_albers_img_diverging	17
scale_linetype_albers	17
theme_albers	18
theme_albers_void	19
use_albersdown	19
use_albers_vignettes	21
Index	22

albers_bs_theme	<i>bs_theme for pkgdown (light/dark aware)</i>
-----------------	--

Description

Convenience wrapper exposing core variables; most consumers won't need this directly if they use `template: { package: albersdown }`.

Usage

```
albers_bs_theme(
  family = "red",
  preset = c("homage", "study", "structural", "adobe", "midnight"),
  accent = NULL,
  bg = NULL,
  fg = NULL
)
```

Arguments

family	Palette family name (default "red").
preset	Visual preset (default "homage"). See albers_presets() .
accent	Primary accent color (default A700 of the chosen family).
bg	Background color (default derived from preset).
fg	Foreground/text color (default derived from preset).

Value

A `bslib::bs_theme` object.

Examples

```
if (requireNamespace("bslib", quietly = TRUE)) {  
  albers_bs_theme()  
}
```

<code>albers_palette</code>	<i>Return four-tone Homage family by name</i>
-----------------------------	---

Description

Return four-tone Homage family by name

Usage

```
albers_palette(family = c("red", "lapis", "ochre", "teal", "green", "violet"))
```

Arguments

family	One of "red", "lapis", "ochre", "teal".
--------	---

Value

Named character vector of four hex colors (A900, A700, A500, A300).

Examples

```
albers_palette("red")  
albers_palette("lapis")
```

`albers_palette_img` *Image-derived Homage palettes (A900 -> A300)*

Description

Four-tone families distilled from the uploaded grid; order is darkest to lightest (A900, A700, A500, A300).

Usage

```
albers_palette_img(family = c("red", "lapis", "ochre", "teal", "green"))
```

Arguments

`family` One of "red", "lapis", "ochre", "teal", "green"

Value

Named character vector of four hex colors (A900, A700, A500, A300).

Examples

```
albers_palette_img("red")
```

`albers_presets` *List available Albers presets*

Description

Returns the names of the five built-in presets, each inspired by a different period or series in Josef Albers' work.

Usage

```
albers_presets()
```

Value

Character vector of preset names.

Examples

```
albers_presets()
```

albers_ramp	<i>Interpolate n colors along a palette family gradient</i>
-------------	---

Description

Uses `colorRampPalette` to interpolate between the four tones of a family (A900 → A300), producing an arbitrary number of evenly spaced colors.

Usage

```
albers_ramp(family = "red", n = 9, reverse = FALSE)
```

Arguments

family	Palette family name.
n	Number of colors to return.
reverse	If TRUE, return colors from light to dark.

Value

Character vector of n hex colors.

Examples

```
albers_ramp("lapis", n = 5)
```

albers_swatch	<i>Visual swatch of Albers palette families and presets</i>
---------------	---

Description

Draws a tile plot showing the four tones of each palette family, optionally faceted by preset ground colors. Useful for quickly previewing the design system in a notebook or presentation.

Usage

```
albers_swatch(
  families = c("red", "lapis", "ochre", "teal", "green", "violet"),
  show_presets = FALSE
)
```

Arguments

families	Character vector of families to show. Defaults to all six.
show_presets	If TRUE, add a row of preset ground colors below the palette tones. Defaults to FALSE.

Value

A ggplot object.

Examples

```
if (requireNamespace("ggplot2", quietly = TRUE)) {
  albers_swatch()
}
```

 gt_albers

Quiet, legible gt style with subtle stripe from A300

Description

Quiet, legible gt style with subtle stripe from A300

Usage

```
gt_albers(
  x,
  family = "red",
  preset = c("homage", "study", "structural", "adobe", "midnight"),
  base_size = 14,
  width = 720,
  bg = NULL,
  fg = NULL
)
```

Arguments

x	A gt table
family	Palette family for subtle accents
preset	Visual preset (default "homage"). See albers_presets() .
base_size	Base font size in pixels (default 14).
width	Table width in pixels (default 720). Use NULL for auto.
bg	Override background color (default derived from preset).
fg	Override text color (default derived from preset).

Value

A styled gt table object.

Examples

```
if (requireNamespace("gt", quietly = TRUE)) {  
  tbl <- gt::gt(head(mtcars))  
  gt_albers(tbl)  
}
```

migrate_albersdown *One-command migration to latest albersdown*

Description

Convenience helper for existing packages that already use albersdown and need to replace older vignette/site wiring with the latest defaults while choosing an Albers accent family and preset.

Usage

```
migrate_albersdown(  
  path,  
  family = "red",  
  preset = c("homage", "study", "structural", "adobe", "midnight"),  
  dry_run = FALSE  
)
```

Arguments

path	Path to the package directory. Must be supplied explicitly; there is no default so that the function never writes to an unexpected location.
family	one of: "red", "lapis", "ochre", "teal", "green", "violet"
preset	Visual preset (default "homage"). See albers_presets() .
dry_run	if TRUE, report changes without writing files.

Value

TRUE invisibly.

Examples

```
if (interactive()) {  
  migrate_albersdown(path = ".", family = "teal", preset = "midnight", dry_run = TRUE)  
}
```

scale_color_albers *Scales that use the family's tones (discrete/continuous)*

Description

Scales that use the family's tones (discrete/continuous)

Usage

```
scale_color_albers(family = "red", discrete = TRUE, ...)
```

```
scale_fill_albers(family = "red", discrete = TRUE, ...)
```

Arguments

family	Palette family.
discrete	Whether to use a discrete palette; if FALSE, uses a gradient.
...	Passed to underlying ggplot2 scale.

Value

A ggplot2 scale object.

Examples

```
if (requireNamespace("ggplot2", quietly = TRUE)) {
  ggplot2::ggplot(iris, ggplot2::aes(Sepal.Length, Sepal.Width,
    color = Species)) + ggplot2::geom_point() + scale_color_albers()
}
```

scale_color_albers_distinct

Distinct, colorblind-friendly line palette across families

Description

Uses one high-contrast tone (default A700) from different families to maximize separation between lines. This departs from the single-family aesthetic but improves readability for multi-series lines.

Usage

```
scale_color_albers_distinct(n = NULL, tone = c("A700", "A900", "A500"), ...)
```


Arguments

n	Number of colors needed; defaults to length of available families (6).
tone	One of "A700", "A900", or "A500".
...	Passed to <code>ggplot2::scale_color_manual()</code> .

Value

A `ggplot2` scale object.

Examples

```
if (requireNamespace("ggplot2", quietly = TRUE)) {
  df <- data.frame(x = 1:6, y = 1:6, g = paste0("G", 1:6))
  ggplot2::ggplot(df, ggplot2::aes(x, y, color = g)) +
    ggplot2::geom_point() + scale_color_albers_distinct()
}
```

scale_color_albers_diverging

Diverging color scale (continuous)

Description

Diverging color scale (continuous)

Usage

```
scale_color_albers_diverging(
  low_family = "red",
  high_family = albers_complement(low_family),
  midpoint = 0,
  neutral = "#e5e7eb",
  ...
)
```

Arguments

low_family, high_family	Homage families for the two sides
midpoint	numeric midpoint for the diverging scale (default 0)
neutral	hex color for the midpoint (default matches CSS border)
...	passed to <code>ggplot2::scale_color_gradient2()</code>

Value

A `ggplot2` scale object.

scale_color_albers_diverging_n

Diverging color scale with multiple stops (continuous)

Description

Uses a 5-stop palette (low2, low1, neutral, high1, high2) for smoother transitions around the midpoint.

Usage

```
scale_color_albers_diverging_n(  
  low_family = "red",  
  high_family = albers_complement(low_family),  
  neutral = "#e5e7eb",  
  ...  
)
```

Arguments

low_family, high_family	Homage families for the two sides
neutral	hex color for the midpoint (default matches CSS border)
...	passed to ggplot2::scale_color_gradient2()

Value

A ggplot2 scale object.

scale_color_albers_highlight

Convenience scale: highlight vs other (color)

Description

Returns a manual color scale mapping a single highlighted group to a family tone (default A700) and all other points to a neutral gray.

Usage

```
scale_color_albers_highlight(
  family = "red",
  tone = c("A700", "A900", "A500", "A300"),
  other = "#9aa0a6",
  highlight = "highlight",
  other_name = "other",
  ...
)
```

Arguments

family	Palette family name.
tone	One of A900, A700, A500, A300 used for the highlight color.
other	Hex color used for non-highlight values.
highlight	Name of the value that should receive the highlight color.
other_name	Name of the value that should receive the neutral color.
...	Passed to <code>ggplot2::scale_color_manual()</code> .

Value

A `ggplot2` scale object.

scale_color_albers_img

Image-derived sequential color scale

Description

Image-derived sequential color scale

Usage

```
scale_color_albers_img(family = "red", discrete = TRUE, ...)
```

```
scale_fill_albers_img(family = "red", discrete = TRUE, ...)
```

Arguments

family	One of "red", "lapis", "ochre", "teal", "green"
discrete	Whether to use discrete palette; if FALSE uses a gradient.
...	Passed to underlying <code>ggplot2</code> scale.

Value

A `ggplot2` scale object.

scale_color_albers_img_diverging
Image-derived diverging color scale (gradientn)

Description

Image-derived diverging color scale (gradientn)

Usage

```
scale_color_albers_img_diverging(  
  low_family,  
  high_family,  
  neutral = "#E4E0D9",  
  ...  
)
```

Arguments

low_family	Family for the low side (left)
high_family	Family for the high side (right)
neutral	Hex color for the midpoint (default derived from image)
...	Passed to ggplot2::scale_color_gradientn

Value

A ggplot2 scale object.

scale_color_albers_img_red_teal
Pre-canned image-derived diverging pairs

Description

Pre-canned image-derived diverging pairs

Usage

```
scale_color_albers_img_red_teal(neutral = "#E4E0D9", ...)  
  
scale_color_albers_img_lapis_ochre(neutral = "#E4E0D9", ...)  
  
scale_color_albers_img_green_red(neutral = "#E4E0D9", ...)
```

```
scale_fill_albers_img_red_teal(neutral = "#E4E0D9", ...)
scale_fill_albers_img_lapis_ochre(neutral = "#E4E0D9", ...)
scale_fill_albers_img_green_red(neutral = "#E4E0D9", ...)
```

Arguments

neutral	Midpoint color (default from image); use "#e5e7eb" to match site CSS
...	Passed to the underlying gradientn scale

Value

A ggplot2 scale object.

scale_fill_albers_distinct

Distinct, colorblind-friendly fill palette across families

Description

Uses one high-contrast tone (default A700) from different families to maximize separation between filled regions. Fill counterpart of [scale_color_albers_distinct](#).

Usage

```
scale_fill_albers_distinct(n = NULL, tone = c("A700", "A900", "A500"), ...)
```

Arguments

n	Number of colors needed; defaults to length of available families (6).
tone	One of "A700", "A900", or "A500".
...	Passed to <code>ggplot2::scale_fill_manual()</code> .

Value

A ggplot2 scale object.

scale_fill_albers_diverging
Diverging fill scale (continuous)

Description

Diverging fill scale (continuous)

Usage

```
scale_fill_albers_diverging(  
  low_family = "red",  
  high_family = albers_complement(low_family),  
  midpoint = 0,  
  neutral = "#e5e7eb",  
  ...  
)
```

Arguments

low_family, high_family	Homage families for the two sides
midpoint	numeric midpoint for the diverging scale (default 0)
neutral	hex color for the midpoint (default matches CSS border)
...	passed to <code>ggplot2::scale_color_gradient2()</code>

Value

A `ggplot2` scale object.

scale_fill_albers_diverging_5
5-class diverging (discrete)

Description

Useful for binned choropleths or sliced residuals. The middle class uses the neutral color.

Usage

```

scale_fill_albers_diverging_5(
  low_family = "red",
  high_family = albers_complement(low_family),
  neutral = "#e5e7eb",
  labels = ggplot2::waiver(),
  ...
)

scale_color_albers_diverging_5(
  low_family = "red",
  high_family = albers_complement(low_family),
  neutral = "#e5e7eb",
  labels = ggplot2::waiver(),
  ...
)

```

Arguments

low_family, high_family	Homage families for the two sides
neutral	hex color for the midpoint (default matches CSS border)
labels	Optional labels for the five classes (low2, low1, mid, high1, high2)
...	Passed to ggplot2::scale_fill_manual() or ggplot2::scale_color_manual().

Value

A ggplot2 scale object.

scale_fill_albers_diverging_n

Diverging fill scale with multiple stops (continuous)

Description

Diverging fill scale with multiple stops (continuous)

Usage

```

scale_fill_albers_diverging_n(
  low_family = "red",
  high_family = albers_complement(low_family),
  neutral = "#e5e7eb",
  ...
)

```

Arguments

low_family, high_family	Homage families for the two sides
neutral	hex color for the midpoint (default matches CSS border)
...	passed to <code>ggplot2::scale_color_gradient2()</code>

Value

A `ggplot2` scale object.

scale_fill_albers_highlight
Convenience scale: highlight vs other (fill)

Description

Convenience scale: highlight vs other (fill)

Usage

```
scale_fill_albers_highlight(
  family = "red",
  tone = c("A700", "A900", "A500", "A300"),
  other = "#9aa0a6",
  highlight = "highlight",
  other_name = "other",
  ...
)
```

Arguments

family	Palette family name.
tone	One of A900, A700, A500, A300 used for the highlight color.
other	Hex color used for non-highlight values.
highlight	Name of the value that should receive the highlight color.
other_name	Name of the value that should receive the neutral color.
...	Passed to <code>ggplot2::scale_color_manual()</code> .

Value

A `ggplot2` scale object.

scale_fill_albers_img_diverging
Image-derived diverging fill scale (gradientn)

Description

Image-derived diverging fill scale (gradientn)

Usage

```
scale_fill_albers_img_diverging(
  low_family,
  high_family,
  neutral = "#E4E0D9",
  ...
)
```

Arguments

low_family	Family for the low side (left)
high_family	Family for the high side (right)
neutral	Hex color for the midpoint (default derived from image)
...	Passed to <code>ggplot2::scale_fill_gradientn</code>

Value

A `ggplot2` scale object.

scale_linetype_albers *Discrete linetype scale to pair with Albers colors*

Description

Provides a sensible set of linetypes for multi-series line charts.

Usage

```
scale_linetype_albers(...)
```

Arguments

...	Passed to <code>ggplot2::scale_linetype_manual()</code> .
-----	---

Value

A `ggplot2` scale object.

`theme_albers`*Minimal, legible plot theme inspired by Josef Albers*

Description

Minimal, legible plot theme inspired by Josef Albers

Usage

```
theme_albers(  
  family = "red",  
  preset = c("homage", "study", "structural", "adobe", "midnight"),  
  base_size = 13,  
  base_family = "sans",  
  bg = NULL,  
  fg = NULL,  
  grid_color = NULL  
)
```

Arguments

<code>family</code>	Palette family used by companion scales.
<code>preset</code>	Visual preset: "homage" (gallery white), "study" (analytical white), "structural" (concrete), "adobe" (warm architectural grey), "midnight" (dark).
<code>base_size</code>	Base font size.
<code>base_family</code>	Base font family.
<code>bg</code>	Override background color (default derived from preset).
<code>fg</code>	Override foreground/text color (default derived from preset).
<code>grid_color</code>	Override grid line color (default derived from preset).

Value

A ggplot2 theme object.

Examples

```
if (requireNamespace("ggplot2", quietly = TRUE)) {  
  ggplot2::ggplot(mtcars, ggplot2::aes(wt, mpg)) +  
    ggplot2::geom_point() + theme_albers()  
}
```

theme_albers_void	<i>Stripped theme for maps, brain surfaces, and abstract compositions</i>
-------------------	---

Description

Extends [theme_albers](#) by removing axes, grid lines, ticks, and panel border – leaving only the plot background, titles, and legend. Useful for spatial visualizations where coordinate axes are meaningless.

Usage

```
theme_albers_void(
  family = "red",
  preset = c("homage", "study", "structural", "adobe", "midnight"),
  base_size = 13,
  base_family = "sans",
  bg = NULL,
  fg = NULL
)
```

Arguments

family	Palette family used by companion scales.
preset	Visual preset: "homage" (gallery white), "study" (analytical white), "structural" (concrete), "adobe" (warm architectural grey), "midnight" (dark).
base_size	Base font size.
base_family	Base font family.
bg	Override background color (default derived from preset).
fg	Override foreground/text color (default derived from preset).

Value

A ggplot2 theme object.

use_albersdown	<i>One-shot setup for existing packages</i>
----------------	---

Description

Turn-key retrofit to adopt the albersdown theme in an existing package. Copies local assets for CRAN-safe vignettes, ensures pkgdown template, optionally patches all vignettes, writes a README note, and prints a doctor report.

Usage

```

use_albersdown(
  path,
  family = "red",
  preset = c("homage", "study", "structural", "adobe", "midnight"),
  apply_to = c("all", "new"),
  dry_run = FALSE,
  fallback_extra = c("auto", "always", "never"),
  force_replace = TRUE
)

```

Arguments

path	Path to the package directory. Must be supplied explicitly; there is no default so that the function never writes to an unexpected location.
family	one of: "red", "lapis", "ochre", "teal", "green", "violet"
preset	Visual preset (default "homage"). See albers_presets() .
apply_to	"all" to patch every <i>.Rmd/.qmd</i> in <i>vignettes/</i> , or "new" to only add the template and assets
dry_run	if TRUE, show changes without writing
fallback_extra	Controls copying site-wide fallbacks into <i>pkgdown/</i> : <ul style="list-style-type: none"> "auto": copy <i>pkgdown/extra.css</i> and <i>pkgdown/extra.js</i> only when the consuming package does not use <code>template: { package: albersdown }</code> in <i>_pkgdown.yml</i>. "always": always copy to <i>pkgdown/</i> (useful as a safety net or for custom setups). "never": never copy site-wide fallbacks.
force_replace	if TRUE (default), overwrite existing albersdown assets and replace existing vignette CSS/header hooks so albersdown becomes the active theme.

Value

TRUE invisibly.

Examples

```

if (interactive()) {
  use_albersdown(path = ".", dry_run = TRUE)
}

```

use_albers_vignettes *Configure current package to use albersdown (back-compat wrapper)*

Description

This wrapper preserves the old name and forwards to use_albersdown().

Usage

```
use_albers_vignettes()
```

Value

TRUE invisibly.

Examples

```
if (interactive()) {  
  use_albers_vignettes()  
}
```

Index

albers_bs_theme, 2
albers_palette, 3
albers_palette_img, 4
albers_presets, 4
albers_presets(), 3, 6, 7, 20
albers_ramp, 5
albers_swatch, 5

colorRampPalette, 5

gt_albers, 6

migrate_albersdown, 7

scale_color_albers, 8
scale_color_albers_distinct, 8, 13
scale_color_albers_diverging, 9
scale_color_albers_diverging_5
 (scale_fill_albers_diverging_5),
 14
scale_color_albers_diverging_n, 10
scale_color_albers_highlight, 10
scale_color_albers_img, 11
scale_color_albers_img_diverging, 12
scale_color_albers_img_green_red
 (scale_color_albers_img_red_tea),
 12
scale_color_albers_img_lapis_ochre
 (scale_color_albers_img_red_tea),
 12
scale_color_albers_img_red_tea, 12
scale_fill_albers(scale_color_albers),
 8
scale_fill_albers_distinct, 13
scale_fill_albers_diverging, 14
scale_fill_albers_diverging_5, 14
scale_fill_albers_diverging_n, 15
scale_fill_albers_highlight, 16
scale_fill_albers_img
 (scale_color_albers_img), 11
scale_fill_albers_img_diverging, 17
scale_fill_albers_img_green_red
 (scale_color_albers_img_red_tea),
 12
scale_fill_albers_img_lapis_ochre
 (scale_color_albers_img_red_tea),
 12
scale_fill_albers_img_red_tea
 (scale_color_albers_img_red_tea),
 12
scale_linetype_albers, 17

theme_albers, 18, 19
theme_albers_void, 19

use_albers_vignettes, 21
use_albersdown, 19