

Package ‘afcharts’

January 13, 2025

Title Produce Charts Following UK Government Analysis Function Guidance

Version 0.4.1

Description Colour palettes and a 'ggplot2' theme to follow the UK Government Analysis Function best practice guidance for producing data visualisations, available at

<<https://analysisfunction.civilservice.gov.uk/policy-store/data-visualisation-charts/>>.

Includes continuous and discrete colour and fill scales, as well as a 'ggplot2' theme.

License MIT + file LICENSE

URL <https://github.com/best-practice-and-impact/afcharts>,
<https://best-practice-and-impact.github.io/afcharts/>

BugReports <https://github.com/best-practice-and-impact/afcharts/issues>

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Depends R (>= 3.5)

Imports ggplot2, scales, cli, rlang, dplyr

Suggests ggtext, knitr, rmarkdown, tibble, tidyr, glue, purrr, stringr, testthat (>= 2.1.0), plotly, gt, svglite (>= 2.1.2), ragg (>= 1.2.6), gapminder, diffviewer, vdiff

VignetteBuilder knitr

NeedsCompilation no

Author Crown Copyright [cph],
Government Analysis Function [fnd],
Alice Hannah [aut],
Olivia Box Power [cre, ctb]

Maintainer Olivia Box Power <Olivia.BoxPower@dhsc.gov.uk>

Repository CRAN

Date/Publication 2025-01-13 15:20:02 UTC

Contents

af_colour_palettes	2
af_colour_values	2
mm_to_inch	3
save_govuk	3
scale_colour_continuous_af	5
scale_colour_discrete_af	6
scale_fill_continuous_af	7
scale_fill_discrete_af	8
theme_af	9
use_afcharts	10

Index	11
--------------	-----------

af_colour_palettes	<i>Analysis Function colour palettes</i>
--------------------	--

Description

A list grouping colours into palettes.

Usage

```
af_colour_palettes
```

Format

A character list

Source

[Government Analysis Function Colours Guidance](#)

af_colour_values	<i>Analysis Function colour names and hex codes</i>
------------------	---

Description

A vector containing colour names and their corresponding hex code.

Usage

```
af_colour_values
```

Format

A character vector

Source

[Government Analysis Function Colours Guidance](#)

mm_to_inch	<i>Convert millimetres to inches</i>
------------	--------------------------------------

Description

Convert millimetres to inches

Usage

```
mm_to_inch(x)
```

Arguments

x Numeric value in millimetres

Value

A numerical value in inches

Examples

```
mm_to_inch(100)
```

save_govuk	<i>Save a plot at the correct dimensions for publishing on GOVUK</i>
------------	--

Description

This is a wrapper around `ggplot2::ggsave()` with plot dimensions set for publishing on GOVUK.

Usage

```
save_govuk(  
  filename,  
  plot = ggplot2::last_plot(),  
  device = c("svg", "png", "jpg"),  
  path = NULL,  
  ...  
)
```

Arguments

filename	File name
plot	The plot to save
device	File type to produce (svg, png or jpg). svg is preferred as it scales well without pixelating
path	Directory to save the plot in
...	Other params passed to ggplot::ggsave

Value

Character vector giving path to saved file

Examples

```
library(ggplot2)
library(dplyr)
library(gapminder)

# Images on GOVUK are shrunk. We therefore recommend using font size 20 pt
# when exporting charts for GOVUK, which will appear as approximately 12 pt on
# the website.
use_afcharts(base_size = 20)

grouped_bar_data <-
  gapminder |>
  filter(year %in% c(1967, 2007) &
         country %in% c("United Kingdom", "Ireland", "France", "Belgium"))

bar_chart <- ggplot(grouped_bar_data,
  aes(x = country, y = lifeExp, fill = as.factor(year))) +
  geom_bar(stat = "identity", position = "dodge") +
  scale_y_continuous(expand = c(0, 0)) +
  scale_fill_discrete_af() +
  labs(
    x = "Country",
    y = NULL,
    fill = NULL,
    title = "Living longer",
    subtitle = "Difference in life expectancy, 1967-2007",
    caption = "Source: Gapminder"
  )

file <- tempfile(fileext = ".svg")
save_govuk(file, bar_chart, device = "svg")
unlink(file)
```

`scale_colour_continuous_af`*Continuous colour scales for Analysis Function plots*

Description

Continuous colour scales for Analysis Function plots

Usage

```
scale_colour_continuous_af(  
  palette = "sequential",  
  palette_type = c("af"),  
  reverse = FALSE,  
  na_colour = "grey50",  
  guide = "colourbar",  
  ...  
)
```

Arguments

<code>palette</code>	Name of palette to use; e.g. "main", "sequential", "focus". Default value is "sequential".
<code>palette_type</code>	Currently only the Analysis Function palettes are supported. Defaults to "af".
<code>reverse</code>	Boolean value to indicate whether the palette should be reversed.
<code>na_colour</code>	Colour to set for missing values.
<code>guide</code>	A name or function used to create guide. Default is "colourbar".
<code>...</code>	Additional arguments passed to scale type.

Value

ggplot2 continuous colour scale

Examples

```
library(ggplot2)  
  
ggplot(mtcars, aes(x = mpg, y = wt, colour = cyl)) +  
  geom_point() +  
  scale_colour_continuous_af()
```

`scale_colour_discrete_af`*Discrete colour scales for Analysis Function plots*

Description

Discrete colour scales for Analysis Function plots

Usage

```
scale_colour_discrete_af(  
  palette = "main",  
  palette_type = c("af"),  
  reverse = FALSE,  
  ...  
)
```

Arguments

<code>palette</code>	Name of palette to use; e.g. "main", "sequential", "focus." Default value is "main".
<code>palette_type</code>	Currently only the Analysis Function palettes are supported. Defaults to "af".
<code>reverse</code>	Boolean value to indicate whether the palette should be reversed.
<code>...</code>	Additional arguments passed to scale type.

Value

ggplot2 discrete colour scale

Examples

```
library(ggplot2)  
library(dplyr)  
  
economics_long %>%  
  filter(variable %in% c("psavert", "uempmed")) %>%  
  ggplot(aes(x = date, y = value, colour = variable)) +  
  geom_line(linewidth = 1) +  
  scale_colour_discrete_af()
```

`scale_fill_continuous_af`*Continuous colour fill scales for Analysis Function plots*

Description

Continuous colour fill scales for Analysis Function plots

Usage

```
scale_fill_continuous_af(  
  palette = "sequential",  
  palette_type = c("af"),  
  reverse = FALSE,  
  na_colour = "grey50",  
  guide = "colourbar",  
  ...  
)
```

Arguments

<code>palette</code>	Name of palette to use; e.g. "main", "sequential", "focus." Default value is "sequential".
<code>palette_type</code>	Currently only the Analysis Function palettes are supported. Defaults to "af".
<code>reverse</code>	Boolean value to indicate whether the palette should be reversed.
<code>na_colour</code>	Colour to set for missing values.
<code>guide</code>	A name or function used to create guide. Default is "colourbar".
<code>...</code>	Additional arguments passed to scale type.

Value

ggplot2 continuous fill scale

Examples

```
library(ggplot2)  
  
ggplot(faithfuld, aes(x = waiting, y = eruptions, fill = density)) +  
  geom_raster() +  
  scale_fill_continuous_af()
```

`scale_fill_discrete_af`*Discrete colour fill scales for Analysis Function plots*

Description

Discrete colour fill scales for Analysis Function plots

Usage

```
scale_fill_discrete_af(  
  palette = "main",  
  palette_type = c("af"),  
  reverse = FALSE,  
  ...  
)
```

Arguments

<code>palette</code>	Name of palette to use; e.g. "main", "sequential", "focus." Default value is "main."
<code>palette_type</code>	Currently only the Analysis Function palettes are supported. Defaults to "af".
<code>reverse</code>	Boolean value to indicate whether the palette should be reversed.
<code>...</code>	Additional arguments passed to scale type.

Value

ggplot2 discrete fill scale

Examples

```
library(ggplot2)  
  
d <- subset(mpg, manufacturer == "ford")  
  
ggplot(d, aes(x = class, fill = class)) +  
  geom_bar() +  
  scale_fill_discrete_af()
```

`theme_af`*Analysis Function theme for ggplot2 charts.*

Description

ggplot2 theme for Analysis Function plots.

Usage

```
theme_af(  
  base_size = 14,  
  base_line_size = base_size/24,  
  base_rect_size = base_size/24,  
  grid = c("y", "x", "xy", "none"),  
  axis = c("x", "y", "xy", "none"),  
  ticks = c("xy", "x", "y", "none"),  
  legend = c("right", "left", "top", "bottom", "none")  
)
```

Arguments

<code>base_size</code>	base font size, given in pts.
<code>base_line_size</code>	base size for line elements.
<code>base_rect_size</code>	base size for rect elements.
<code>grid, axis, ticks</code>	'x', 'y', 'xy' or 'none' to determine for which axes the attribute should be drawn. Grid defaults to 'y', axis to 'x', and ticks to 'xy'.
<code>legend</code>	'right', 'left', 'top', 'bottom', or 'none' to determine the position of the legend. Defaults to 'right'.

Value

ggplot2 plot theme

Examples

```
library(ggplot2)  
  
p <- ggplot(mpg, aes(x = class)) + geom_bar()  
  
p  
p + theme_af()
```

use_afcharts	<i>Use afcharts defaults.</i>
--------------	-------------------------------

Description

Set afcharts theme, colour palette and geom aesthetic defaults for ggplot2 charts.

Usage

```
use_afcharts(default_colour = afcharts::af_colour_values["dark-blue"], ...)
```

Arguments

`default_colour` Default colour/fill for geoms. Default value is 'blue' from `af_colour_values`.
`...` Arguments passed to `theme_af()`.

Value

NULL. Function is used for side effects of setting ggplot2 plot theme, colour palette and geom aesthetic defaults.

Examples

```
library(ggplot2)

d <- subset(mpg, manufacturer == "ford")

ggplot(d, aes(x = model)) + geom_bar()
ggplot(d, aes(x = model, fill = class)) + geom_bar()

use_afcharts()

ggplot(d, aes(x = model)) + geom_bar()
ggplot(d, aes(x = model, fill = class, colour = class)) + geom_bar()
```

Index

* datasets

af_colour_palettes, 2

af_colour_values, 2

af_colour_palettes, 2

af_colour_values, 2

ggplot2::ggsave(), 3

mm_to_inch, 3

save_govuk, 3

scale_colour_continuous_af, 5

scale_colour_discrete_af, 6

scale_fill_continuous_af, 7

scale_fill_discrete_af, 8

theme_af, 9

use_afcharts, 10