

# Package ‘gameofthrones’

December 31, 2018

**Type** Package

**Title** Palettes Inspired in the TV Show “Game of Thrones”

**Version** 1.0.0

**Maintainer** Alejandro Jimenez Rico <aljrico@gmail.com>

**Description** Implementation of the characteristic palettes from the TV show ‘Game of Thrones’.

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** TRUE

**Depends** R (>= 2.10)

**Suggests** hexbin (>= 1.27.0), testthat

**URL** <https://github.com/aljrico/gameofthrones>

**BugReports** <https://github.com/aljrico/gameofthrones/issues>

**Imports** ggplot2 (>= 1.0.1), gridExtra, MASS

**RoxygenNote** 6.1.1

**NeedsCompilation** no

**Author** Alejandro Jimenez Rico [aut, cre]

**Repository** CRAN

**Date/Publication** 2018-12-30 23:30:06 UTC

## R topics documented:

got.map . . . . .	2
gotMap . . . . .	2
scale_color_got . . . . .	4

<b>Index</b>	<b>6</b>
--------------	----------

got.map

*'Game of Thrones' colour map*

---

### Description

A dataset containing the colour palettes from the TV show 'Game of Thrones'.

### Usage

```
got.map
```

### Format

A data frame containing all the colours used in the palette:

- V1: Red value
- V2: Green value
- V3: Blue value
- option: Refers to the houses of Westeros, or other colourmaps. It is intended to be a general option for choosing the specific colour palette.

---

gotMap

*Game of Thrones Colour Map.*

---

### Description

This function creates a vector of n equally spaced colors along the 'got colour map'.

### Usage

```
gotMap(n = 256, alpha = 1, begin = 0, end = 1, direction = 1,  
      option = "targaryen")
```

```
got(n, alpha = 1, begin = 0, end = 1, direction = 1,  
   option = "targaryen")
```

```
got_pal(alpha = 1, begin = 0, end = 1, direction = 1,  
        option = "targaryen")
```

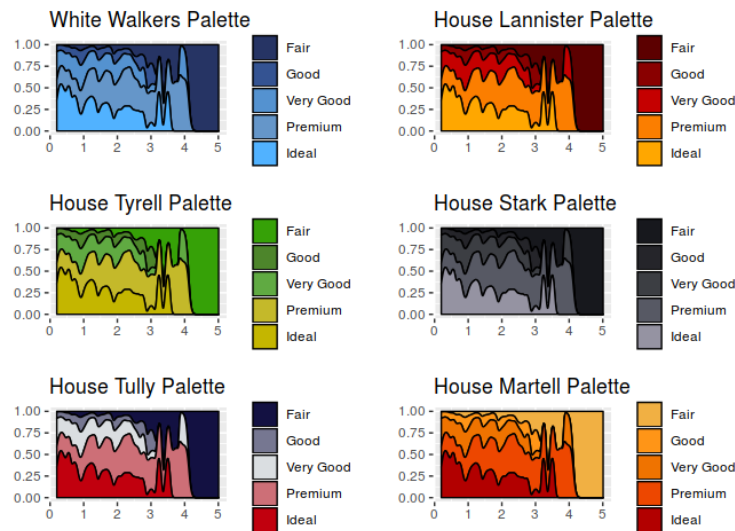
```
gameofthrones(n, alpha = 1, begin = 0, end = 1, direction = 1,  
             option = "targaryen")
```

## Arguments

n	The number of colors ( $\geq 1$ ) to be in the palette.
alpha	The alpha transparency, a number in $[0,1]$ , see argument alpha in <a href="#">hsv</a> .
begin	The (corrected) hue in $[0,1]$ at which the got colormap begins.
end	The (corrected) hue in $[0,1]$ at which the got colormap ends.
direction	Sets the order of colors in the scale. If 1, the default, colors are ordered from darkest to lightest. If -1, the order of colors is reversed.
option	A character string indicating the colourmap to use. It is not case-sensitive

## Details

Here are the color scales:



Semi-transparent colors ( $0 < \alpha < 1$ ) are supported only on some devices: see [rgb](#).

## Value

got returns a character vector, cv, of color hex codes. This can be used either to create a user-defined color palette for subsequent graphics by `palette(cv)`, a `col` = specification in graphics functions or in `par`.

gotMap returns a n lines data frame containing the red (R), green (G), blue (B) and alpha (alpha) channels of n equally spaced colors along the 'Game of Thrones' colour map. n = 256 by default.

## Author(s)

Alejandro Jiménez Rico <aljrigo@gmail.com>, [Personal Blog](#)

**Examples**

```
library(ggplot2)
library(hexbin)

dat <- data.frame(x = rnorm(1e4), y = rnorm(1e4))

ggplot(dat, aes(x = x, y = y)) +
  geom_hex() + coord_fixed() +
  scale_fill_gradientn(colours = got(128, option = "targaryen")) +
  theme_minimal()

pal <- got(256, option = "Wildfire")
image(volcano, col = pal)
```

---

scale\_color\_got

*Game of Thrones colour scales*


---

**Description**

Uses the Game of Thrones color scale.

**Usage**

```
scale_color_got(..., alpha = 1, begin = 0, end = 1, direction = 1,
  discrete = FALSE, option = "targaryen")
```

```
scale_colour_got(..., alpha = 1, begin = 0, end = 1, direction = 1,
  discrete = FALSE, option = "targaryen")
```

```
scale_colour_got_d(..., alpha = 1, begin = 0, end = 1,
  direction = 1, option, aesthetics = "colour")
```

```
scale_color_got_d(..., alpha = 1, begin = 0, end = 1,
  direction = 1, option, aesthetics = "colour")
```

```
scale_fill_got(..., alpha = 1, begin = 0, end = 1, direction = 1,
  discrete = FALSE, option = "targaryen")
```

```
scale_fill_got_d(..., alpha = 1, begin = 0, end = 1, direction = 1,
  option, aesthetics = "fill")
```

**Arguments**

...	parameters to discrete_scale or scale_fill_gradientn
alpha	pass through parameter to got
begin	The (corrected) hue in [0,1] at which the got colormap begins.

end	The (corrected) hue in [0,1] at which the got colormap ends.
direction	Sets the order of colors in the scale. If 1, the default, colors are as output by <code>got_pal</code> . If -1, the order of colors is reversed.
discrete	generate a discrete palette? (default: FALSE - generate continuous palette)
option	A character string indicating the colourmap to use.
aesthetics	Character string or vector of character strings listing the name(s) of the aesthetic(s) that this scale works with. This can be useful, for example, to apply colour settings to the 'colour' and 'fill' aesthetics at the same time, via 'aesthetics = c("colour", "fill")'. Parameter inherited from 'ggplot2'.

### Details

For discrete == FALSE (the default) all other arguments are as to [scale\\_fill\\_gradientn](#) or [scale\\_color\\_gradientn](#). Otherwise the function will return a `discrete_scale` with the plot-computed number of colors.

### Author(s)

Alejandro Jiménez Rico <aljrigo@gmail.com>

### Examples

```
library(ggplot2)

ggplot(mtcars, aes(factor(cyl), fill=factor(vs))) +
  geom_bar() +
  scale_fill_got(discrete = TRUE, option = "Daenerys")

ggplot(mtcars, aes(factor(gear), fill=factor(carb))) +
  geom_bar() +
  scale_fill_got(discrete = TRUE, option = "Tully")

ggplot(mtcars, aes(x = mpg, y = disp, colour = hp)) +
  geom_point(size = 2) +
  scale_colour_got(option = "Lannister")
```

# Index

## \*Topic **datasets**

got.map, 2

gameofthrones (gotMap), 2

got (gotMap), 2

got.map, 2

got\_pal (gotMap), 2

gotMap, 2

hsv, 3

rgb, 3

scale\_color\_got, 4

scale\_color\_got\_d (scale\_color\_got), 4

scale\_color\_gradientn, 5

scale\_colour\_got (scale\_color\_got), 4

scale\_colour\_got\_d (scale\_color\_got), 4

scale\_fill\_got (scale\_color\_got), 4

scale\_fill\_got\_d (scale\_color\_got), 4

scale\_fill\_gradientn, 5