

# Package ‘mandelbrot’

October 13, 2022

**Type** Package

**Title** Generates Views on the Mandelbrot Set

**Version** 0.2.0

**Description** Estimates membership for the Mandelbrot set.

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.0.1

**Imports** reshape2

**Suggests** testthat, RColorBrewer

**NeedsCompilation** yes

**Author** Ben Moore [aut, cre],  
Mario dos Reis [aut]

**Maintainer** Ben Moore <ben@blm.io>

**Repository** CRAN

**Date/Publication** 2017-07-06 10:06:55 UTC

## R topics documented:

as.data.frame.mandelbrot . . . . .	2
mandelbrot . . . . .	2
mandelbrot_palette . . . . .	3
plot.mandelbrot . . . . .	4

<b>Index</b>	<b>5</b>
--------------	----------

```
as.data.frame.mandelbrot
```

*Convert Mandelbrot object to data.frame for plotting*

---

### Description

Converts objects produced by [mandelbrot](#) to tidy data.frames for use with ggplot and other tidyverse packages.

### Usage

```
## S3 method for class 'mandelbrot'  
as.data.frame(x, ...)
```

### Arguments

x                    a Mandelbrot set object produced by [mandelbrot](#)  
...                    ignored

### Value

a 3-column data.frame

### Examples

```
mb <- mandelbrot()  
df <- as.data.frame(mb)  
head(df)
```

---

```
mandelbrot
```

*Calculate the Mandelbrot set*

---

### Description

Generates a view on the Mandelbrot set using an underlying C function.

### Usage

```
mandelbrot(xlim = c(-2, 2), ylim = c(-2, 2), resolution = 600,  
          iterations = 50)  
  
mandelbrot0(xlim = c(-2, 2), ylim = c(-2, 2), resolution = 600,  
            iterations = 50)
```

**Arguments**

xlim	limits of x axis (real part)
ylim	limits of y axis (imaginary part)
resolution	either an integer $n$ for $n^2$ pixels or a list with x and y components specifying the resolution in each direction (e.g. <code>list(x = 500, y = 500)</code> )
iterations	maximum number of iterations to evaluate each case

**Details**

mandelbrot0 is an experimental interface for generating tidy data.frames faster than `as.data.frame(mandelbrot())`.

**Value**

a mandelbrot structure with components: x a vector of the real parts of the x-axis; y the imaginary parts of each number (the y-axis); z a matrix of the number of iterations that  $|z| < 2$

**Mandelbrot set**

In brief, the Mandelbrot set contains the complex numbers where the 0 orbit of the following function remains bounded ( $< 2$ ):

$$f_{z+1} = z^2 + c$$

For information and discussion on the Mandelbrot and related sets, one great resource is [plus.maths.org](http://plus.maths.org). There's also a popular [YouTube video by Numberphile](#).

**Credits**

Wraps original C code by Mario dos Reis, September 2003.

**References**

<https://stat.ethz.ch/pipermail/r-help/2003-October/039773.html> <http://people.crysl.bbk.ac.uk/~fdosr01/Rfractals/index.html>

---

mandelbrot\_palette      *Generate palette suitable for coloring a set*

---

**Description**

Takes a simple palette and expands / oscillates it for use with Mandelbrot sets.

**Usage**

```
mandelbrot_palette(palette, fold = TRUE, reps = 1L, in_set = "black")
```

**Arguments**

palette	vector of color hex strings (e.g. '#FFFFFF')
fold	wrap or fold the palette back on itself
reps	number of times to replicate the color vector
in_set	color for areas in the Mandelbrot set

**Value**

an extended color vector

**Examples**

```
view <- mandelbrot(xlim = c(-0.8438146, -0.8226294),
  ylim = c(0.1963144, 0.2174996), iter = 500)

# can be used to simply interpolate a color gradient
spectral <- RColorBrewer::brewer.pal(11, "Spectral")
cols <- mandelbrot_palette(spectral, fold = FALSE)
plot(view, col = cols, transform = "inv")

# simple palettes might need folds / reps to look good
blues <- RColorBrewer::brewer.pal(9, "Blues")
cols <- mandelbrot_palette(blues, in_set = "white",
  fold = TRUE, reps = 2)
plot(view, col = cols, transform = "log")
```

---

plot.mandelbrot	<i>Plot a Mandelbrot set using base graphics</i>
-----------------	--------------------------------------------------

---

**Description**

Draws colored set membership using image.

**Usage**

```
## S3 method for class 'mandelbrot'
plot(x, col = mandelbrot_palette(c("white",
  grey.colors(50))), transform = c("none", "inverse", "log"), asp = 1, ...)
```

**Arguments**

x	an object generated by <a href="#">mandelbrot</a>
col	a vector of colors, such as those generated by <a href="#">mandelbrot_palette</a>
transform	the name of a transformation to apply to the number of iterations matrix
asp	the asp parameter to <a href="#">image</a> which controls aspect ratio
...	extra arguments passed to <a href="#">image</a>

# Index

`as.data.frame.mandelbrot`, [2](#)

`image`, [4](#)

`mandelbrot`, [2](#), [2](#), [4](#)

`mandelbrot0 (mandelbrot)`, [2](#)

`mandelbrot_palette`, [3](#), [4](#)

`plot.mandelbrot`, [4](#)