Package ‘ggseg3d’

October 13, 2022

Title  Tri-Surface Mesh Plots for Brain Atlases

Version 1.6.3

Description Mainly contains a plotting function ggseg3d(),
and data of two standard brain atlases (Desikan-Killiany and aseg).
By far, the largest bit of the package is the data for each of the atlases.
The functions and data enable users to plot tri-surface mesh plots of
brain atlases, and customise these by projecting colours onto the brain
segments based on values in their own data sets. Functions are wrappers
for 'plotly'. Mowinckel & Vidal-Piñeiro (2020)

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.1.1

Depends R (>= 2.10)

LazyData true

LazyDataCompression xz

Imports dplyr, plotly, magrittr, scales, tidyr, utils,

Suggests knitr, rmarkdown, covr, testthat (>= 2.1.0), devtools,
processx, spelling

URL https://github.com/ggseg/ggseg3d/

BugReports https://github.com/ggseg/ggseg3d/issues/

Language en-US

NeedsCompilation no

Author Athanasia Mo Mowinckel [aut, cre]
(https://orcid.org/0000-0002-5756-0223),
Didac Vidal-Piñeiro [aut] (https://orcid.org/0000-0001-9997-9156)

Maintainer Athanasia Mo Mowinckel <a.m.mowinckel@psykologi.uio.no>

Repository CRAN

Date/Publication 2021-06-01 07:20:02 UTC
add_glassbrain

Description

Adds a translucent brain on top of a ggseg3d plot to create a point of reference, particularly important for sub-cortical plots.

Usage

```r
add_glassbrain(
  p,
  hemisphere = c("left", "right"),
  colour = "#cecece",
  opacity = 0.3
)
```

Arguments

- `p`: plotly object
- `hemisphere`: string. hemisphere to plot ("left" or "right")
- `colour`: string. colour to give the glass brain
- `opacity`: numeric. transparency of the glass brain (0-1 float)

Value

plotly object with glass brain tri-surface mesh

Examples

```r
library(dplyr)
ggseg3d(atlas="aseg_3d") %>%
  add_glassbrain("left")
```
aseg_3d

FreeSurfer automatic subcortical segmentation of a brain volume

Description

Coordinate data for the subcortical parcellations implemented in FreeSurfer.

Usage

data(aseg_3d)

Format

A tibble with 4 observations and a nested data.frame

- `surf` type of surface ('inflated' or 'white')
- `hemi` hemisphere ('left' or 'right')
- `data` data.frame of necessary variables for plotting
- `atlas` String. atlas name
- `roi` numbered region from surface
- `annot` concatenated region name
- `label` label 'hemi_annot' of the region
- `mesh` list of meshes in two lists: vb and it
- `region` name of region in full
- `colour` HEX colour of region

References


See Also

Other ggseg3d_atlases: dk_3d

Examples

data(aseg_3d)
**Description**

Mesh data for the Desikan-Killiany Cortical atlas, with 40 regions in on the cortical surface of the brain.

**Usage**

```r
data(dk_3d)
```

**Format**

A tibble with 4 observations and a nested data.frame

- **surf** type of surface (‘inflated’ or ‘white’)
- **hemi** hemisphere (‘left’ or ‘right’)
- **data** data.frame of necessary variables for plotting
- **atlas** String. atlas name
- **roi** numbered region from surface
- **annot** concatenated region name
- **label** label ‘hemi_annot’ of the region
- **mesh** list of meshes in two lists: vb and it
- **acronym** abbreviated name of annot
- **lobe** lobe localization
- **region** name of region in full
- **colour** HEX colour of region

**Details**

A nested tibble for all available surfaces and hemispheres

**References**

Fischl et al. (2004) Cerebral Cortex 14:11-22 (PubMed)

**See Also**

Other ggseg3d_atlases: `aseg_3d`

**Examples**

```r
data(dk_3d)
```
Description

The `ggseg3d_atlas` class is a subclass of `data.frame` created in order to have different default behaviour. It heavily relies on the "tibble" [tibble::tibble()]. The `ggseg3d_atlas` class provides a nested tibble of different brain surface shapes, hemispheres and tri-surface mesh information for different brain regions in a specific atlas.

Usage

```r
as_ggseg3d_atlas(x, return = FALSE)
```

Arguments

- **x**: data.frame to be made a ggseg-atlas
- **return**: return logical

Value

an object of class `ggseg3d_atlas`. A nested tibble of different brain surface shapes, hemispheres and tri-surface mesh information for different brain regions in a specific atlas.

Properties of `ggseg3d_atlas`

- A `class` attribute of `c("ggseg3d_atlas", "tbl_df", "tbl", "data.frame")`.
- A base type of `"list"`, where each element of the list has the same `NROW()`.

A lot of this script and its functions are taken from the [tibble::tibble()-package](https://www.tidyverse.org/packages/).

See Also


Examples

```r
tmp <- as.data.frame(dk_3d)
class(tmp)
new_atlas <- as_ggseg3d_atlas(tmp)
class(new_atlas)
```
is_ggseg3d_atlas  
*Check if is ggseg_atlas-class*

**Description**

Check if is ggseg_atlas-class

**Usage**

```r
is_ggseg3d_atlas(x)
```

**Arguments**

- `x`  
atlas object to check

**Value**

logical

---

pan_camera  
*Pan camera position of ggseg3d plot*

**Description**

The default position for plotly mesh plots are not satisfying for brain plots. This convenience function can pan the camera to lateral or medial view, or to custom made views if you are plotly savvy.

**Usage**

```r
pan_camera(p, camera, aspectratio = 1)
```

**Arguments**

- `p`  
plotly object
- `camera`  
string or list.
- `aspectratio`  
camera aspect ratio

**Value**

plotly object

**Examples**

```r
library(dplyr)
ggseg3d() %>%
  pan_camera("right lateral")
```
remove_axes

Remove axis information from ggseg3d plot

Description
When publishing data visualisation in 3d mesh plots in general the axes are not important, at least they are not for ggseg3d, where the axis values are arbitrary.

Usage
remove_axes(p)

Arguments
p plotly object

Value
plotly object without axes

Examples
library(magrittr)
ggseg3d() %>%
  remove_axes()
Index

* datasets
  aseg_3d, 3
  dk_3d, 4
* ggseg3d_atlases
  aseg_3d, 3
  dk_3d, 4

add_glassbrain, 2
as_ggseg3d_atlas (ggseg3d_atlas-class),
  5
aseg_3d, 3, 4

dk_3d, 3, 4

ggseg3d_atlas (ggseg3d_atlas-class), 5
ggseg3d_atlas-class, 5

is_ggseg3d_atlas, 6

pan_camera, 6

remove_axes, 7