

Package ‘rpyANTs’

February 28, 2023

Title An Alternative Advanced Normalization Tools ('ANTs')

Version 0.0.1

Description Provides portable access from 'R' to biomedical image processing toolbox 'ANTs' by Avants et al. (2009) <[doi:10.54294/uvnhin](https://doi.org/10.54294/uvnhin)> via seamless integration with the 'Python' implementation 'ANTsPy'. Allows biomedical images to be processed in 'Python' and analyzed in 'R', and vice versa via shared memory. See 'citation(`rpyANTs`)' for more reference information.

License Apache License 2.0

Language en-US

Encoding UTF-8

RoxygenNote 7.2.3

Imports grDevices, rpymat (>= 0.1.4), reticulate (>= 1.26)

Suggests RNifti (>= 1.4.4), testthat (>= 3.0.0)

Config/testthat/edition 3

URL <http://dipterix.org/rpyANTs/>

BugReports <https://github.com/dipterix/rpyANTs/issues>

Copyright For the rpyANTs package - Zhengjia Wang; For ANTs - ConsortiumOfANTs; For ANTsPy - ANTs Contributors.

NeedsCompilation no

Author Zhengjia Wang [aut, cre]

Maintainer Zhengjia Wang <dipterix.wang@gmail.com>

Repository CRAN

Date/Publication 2023-02-28 14:42:29 UTC

R topics documented:

ants	2
ants_available	2
ants_plot	3

ants_plot_grid	5
ants_registration	7
as_ANTsImage	9
as_ANTsTransform	10
install_ants	11
is_affine3D	12
py	13
py_builtin	14
py_list	15
py_slice	16

Index 17

ants	<i>Get 'ANTs' module</i>
------	--------------------------

Description

Get 'ANTs' module

Usage

ants

load_ants(force = FALSE)

Arguments

force whether to force reloading ants module; default is false

Value

A 'Python' module

ants_available	<i>Check if 'ANTs' is available</i>
----------------	-------------------------------------

Description

Check if 'ANTs' is available

Usage

ants_available()

Value

Logical, whether 'ANTs' is available

See Also[install_ants](#)

ants_plot	<i>Plot single 'ANTsImage'</i>
-----------	--------------------------------

Description

Plot single 'ANTsImage'

Usage

```
ants_plot(  
  image,  
  overlay = NULL,  
  blend = FALSE,  
  alpha = 1,  
  cmap = "Greys_r",  
  overlay_cmap = "turbo",  
  overlay_alpha = 0.9,  
  vmin01 = NULL,  
  vmax01 = NULL,  
  cbar = FALSE,  
  cbar_length = 0.8,  
  cbar_dx = 0,  
  cbar_vertical = TRUE,  
  axis = 0,  
  nslices = 12,  
  slices = NULL,  
  ncol = NULL,  
  slice_buffer = NULL,  
  black_bg = TRUE,  
  bg_thresh_quant = 0.01,  
  bg_val_quant = 0.99,  
  domain_image_map = NULL,  
  crop = FALSE,  
  scale = FALSE,  
  reverse = FALSE,  
  title = "",  
  title_fontsize = 20,  
  title_dx = 0,  
  title_dy = 0,  
  filename = NULL,  
  dpi = 500,  
  figsize = 1.5,  
  reorient = TRUE,
```

```

    resample = TRUE,
    force_agg = FALSE,
    close_figure = TRUE
)

```

Arguments

image 'ANTsImage', or something can be converted to 'ANTsImage'
overlay overlay 'ANTsImage', can be NULL, optional
blend whether to blend image with overlay; default is false
cmap, alpha image color map and transparency
overlay_cmap, overlay_alpha overlay color map and transparency
vminol, vmaxol I could not find its usage
cbar whether to draw color legend
cbar_length, cbar_dx, cbar_vertical legend position and size
axis see 'Details'
nslices, slices, ncol controls slice to show
slice_buffer performance
black_bg, bg_thresh_quant, bg_val_quant controls background
domain_image_map optional 'ANTsImage'
crop, scale, reverse whether to crop, scale, or reverse the image according to background
title, title_fontsize, title_dx, title_dy image title
filename, dpi, figsize needed when saving to file
reorient whether to reorient to 'LAI' before plotting; default is true
resample whether to resample
force_agg whether to force graphic engine to use 'agg' device; default is false
close_figure whether to close figure when returning the function

Details

By default, images will be reoriented to 'LAI' orientation before plotting. So, if `axis=0`, the images will be ordered from the left side of the brain to the right side of the brain. If `axis=1`, the images will be ordered from the anterior (front) of the brain to the posterior (back) of the brain. And if `axis=2`, the images will be ordered from the inferior (bottom) of the brain to the superior (top) of the brain.

Value

Nothing

Examples

```
if(interactive() && ants_available()) {
  ants <- load_ants()
  img <- ants$image_read(ants$get_ants_data('mni'))

  ants_plot(
    img, nslices = 12, black_bg = FALSE,
    bg_thresh_quant = 0.05, bg_val_quant = 1.0, axis = 2,
    cbar = TRUE, crop = TRUE, reverse = TRUE, cbar_vertical = FALSE,
    ncol = 4, title = "Axial view of MNI brain"
  )
}
```

ants_plot_grid	<i>Plot multiple 'ANTsImage'</i>
----------------	----------------------------------

Description

R-friendly wrapper function for ants\$plot_grid

Usage

```
ants_plot_grid(
  images,
  shape = NULL,
  slices = 0,
  axes = 2,
  figsize = 1,
  rpad = 0,
  cpad = 0,
  vmin = NULL,
  vmax = NULL,
  colorbar = TRUE,
  cmap = "Greys_r",
  title = "",
  tfontsize = 20,
  title_dx = 0,
  title_dy = 0,
  rlabels = NULL,
  rfontsize = 14,
```

```

    rfontcolor = "black",
    rfacecolor = "white",
    clabels = NULL,
    cfontsize = 14,
    cfontcolor = "black",
    cfacecolor = "white",
    filename = NULL,
    dpi = 400,
    transparent = TRUE,
    ...,
    force_agg = FALSE,
    close_figure = TRUE
)

```

Arguments

images	a single 'ANTsImage', list, or nested list of 'ANTsImage'
shape	shape of grid, default is using dimensions of images
slices	length of one or equaling to length of slices, slice number to plot
axes	0 for 'sagittal', 1 for 'coronal', 2 for 'axial'; default is 2
figsize, rpad, cpad, colorbar, cmap, transparent	graphical parameters
vmin, vmax	value threshold for the image
title	title of figure
title_dx, title_dy, tfontsize	controls title margin and size
rlabels, clabels	row and column labels
rfontsize, rfontcolor, rfacecolor, cfontsize, cfontcolor, cfacecolor	row and column font size, color, and background color
filename, dpi	parameters to save figures
...	passed to ants\$plot_grid; make sure all entries are named
force_agg	whether to force graphic engine to use 'agg' device; default is false
close_figure	whether to close figure when returning the function

Value

Nothing

Examples

```

if(interactive() && ants_available()) {
  ants <- load_ants()
  image1 <- ants$image_read(ants$get_ants_data('mni'))
  image2 <- image1$smooth_image(1.0)
}

```

```
image3 <- image1$smooth_image(2.0)
image4 <- image1$smooth_image(3.0)

ants_plot_grid(
  list(image1, image2, image3, image4),
  slices = 100, title = "4x1 Grid"
)

ants_plot_grid(
  list(image1, image2, image3, image4),
  shape = c(2, 2),
  slices = 100, title = "2x2 Grid"
)

ants_plot_grid(
  list(image1, image2, image3, image4),
  shape = c(2, 2), axes = c(0,1,2,1),
  slices = 100, title = "2x2 Grid (diff. anatomical slices)"
)
}
```

ants_registration	<i>Register two images using 'ANTs'</i>
-------------------	---

Description

Register two images using 'ANTs'

Usage

```
ants_registration(
  fixed,
  moving,
  type_of_transform = "SyN",
  initial_transform = NULL,
  outprefix = tempfile(),
  mask = NULL,
  grad_step = 0.2,
  flow_sigma = 3,
  total_sigma = 0,
  aff_metric = c("mattes", "GC", "meansquares"),
  aff_sampling = 32,
  aff_random_sampling_rate = 0.2,
  syn_metric = c("mattes", "CC", "meansquares", "demons"),
  syn_sampling = 32,
  reg_iterations = c(40, 20, 0),
```

```

    aff_iterations = c(2100, 1200, 1200, 10),
    aff_shrink_factors = c(6, 4, 2, 1),
    aff_smoothing_sigmas = c(3, 2, 1, 0),
    write_composite_transform = FALSE,
    verbose = FALSE,
    smoothing_in_mm = FALSE,
    ...
)

```

Arguments

fixed	fixed image to which we register the moving image, can be character path to 'NIfTI' image, or 'ANTsImage' instance, 'oro.nifti' object, 'niftiImage' from package 'RNifti', or 'threeBrain.nii' from package 'threeBrain'; see also as_ANTsImage
moving	moving image to be mapped to fixed space; see also as_ANTsImage
type_of_transform	a linear or non-linear registration type; print <code>ants\$registration</code> to see details
initial_transform	optional list of strings; transforms to apply prior to registration
outprefix	output file to save results
mask	image mask; see also as_ANTsImage
grad_step, flow_sigma, total_sigma	optimization parameters
aff_metric	the metric for the 'affine' transformation, choices are 'GC', 'mattes', 'meansquares'
aff_sampling, aff_random_sampling_rate, aff_iterations, aff_shrink_factors, aff_smoothing_sigmas	controls 'affine' transform
syn_metric	the metric for the 'SyN' transformation, choices are 'GC', 'mattes', 'meansquares', 'demons'
syn_sampling, reg_iterations	controls the 'SyN' transform
write_composite_transform	whether the composite transform (and its inverse, if it exists) should be written to an 'HDF5' composite file; default is false
verbose	verbose the progress
smoothing_in_mm	logical, currently only impacts low dimensional registration
...	others passed to <code>ants\$registration</code>

Value

A 'Python' dictionary of aligned images and transform files.

Examples

```

if(interactive() && ants_available()) {

  ants <- load_ants()

  # check the python documentation here for detailed explanation
  print(ants$registration)

  # example to register
  fi <- ants$image_read(ants$get_ants_data('r16'))
  mo <- ants$image_read(ants$get_ants_data('r64'))

  # resample to speed up this example
  fi <- ants$resample_image(fi, list(60L,60L), TRUE, 0L)
  mo <- ants$resample_image(mo, list(60L,60L), TRUE, 0L)

  # SDR transform
  transform <- ants_registration(
    fixed=fi, moving=mo, type_of_transform = 'SyN' )

  ants$plot(fi, overlay = transform$warpedmovout, overlay_alpha = 0.3)

}

```

as_ANTsImage

Load data as 'ANTsImage' class

Description

Load data as 'ANTsImage' class

Usage

```
as_ANTsImage(x, strict = FALSE)
```

Arguments

x	data to be converted; this can be an 'ANTsImage' instance, character, 'oro.nifti' object, 'niftiImage' from package 'RNifti', or 'threeBrain.nii' from package 'threeBrain'
strict	whether x should not be NULL

Value

An 'ANTsImage' instance; use ants\$ANTsImage to see the 'Python' documentation

Examples

```

if(interactive() && ants_available()) {

  ants <- load_ants()

  # Python string
  x1 <- ants$get_ants_data('r16')
  as_ANTsImage( x1 )

  # R character
  nii_path <- system.file(package = "RNifti",
                          "extdata", "example.nii.gz")
  as_ANTsImage( nii_path )

  # niftiImage object
  x2 <- RNifti::readNifti(nii_path)
  as_ANTsImage( x2 )

}

```

as_ANTsTransform *Convert to 'ANTsTransform'*

Description

Convert to 'ANTsTransform'

Usage

```

as_ANTsTransform(x, ...)

## Default S3 method:
as_ANTsTransform(x, dimension = 3, ...)

## S3 method for class 'ants.core.ants_transform.ANTsTransform'
as_ANTsTransform(x, ...)

## S3 method for class 'ants.core.ants_image.ANTsImage'
as_ANTsTransform(x, ...)

## S3 method for class 'numpy.ndarray'
as_ANTsTransform(x, ...)

## S3 method for class 'character'
as_ANTsTransform(x, ...)

```

Arguments

x 'affine' matrix or 'numpy' array, character path to the matrix, 'ANTsTransform',
 'ANTsImage' as displacement field.

... passed to other methods

dimension expected transform space dimension; default is 3

Value

An 'ANTsTransform' object

Examples

```
if(interactive() && ants_available()) {

  mat <- matrix(c(
    0, -1, 0, 128,
    1, 0, 0, -128,
    0, 0, -1, 128,
    0, 0, 0, 1
  ), ncol = 4, byrow = TRUE)

  trans <- as_ANTsTransform(mat)
  trans

  # apply transform
  trans$apply_to_point(c(120, 400, 1))

  # same results
  mat %*% c(120, 400, 1, 1)

  trans[] == mat

}
```

install_ants

Install 'ANTs' via 'ANTsPy'

Description

Install 'ANTs' via 'ANTsPy'

Usage

```
install_ants(python_ver = "auto", verbose = TRUE)
```

Arguments

python_ver 'Python' version, see [configure_conda](#)
 verbose whether to print the installation messages

Value

This function returns nothing.

is_affine3D	<i>Check if an object is a 3D 'affine' transform matrix</i>
-------------	---

Description

Check if an object is a 3D 'affine' transform matrix

Usage

```
is_affine3D(x, ...)
```

Default S3 method:
 is_affine3D(x, strict = TRUE, ...)

S3 method for class 'ants.core.ants_transform.ANTsTransform'
 is_affine3D(x, ...)

Arguments

x R or Python object, accepted forms are numeric matrix, 'ANTsTransform', or character (path to transform matrix)
 ... passed to other methods
 strict whether the last element should be always 1

Value

A logical value whether the object can be loaded as a 4-by-4 matrix.

Examples

```
# not affine
is_affine3D(1)

# 3x3 matrix is not as it is treated as 2D transform
is_affine3D(matrix(rnorm(9), nrow = 3))

# 3x4 matrix
x <- matrix(rnorm(12), nrow = 3)
```

```
is_affine3D(x)

# 4x4 matrix
x <- rbind(x, c(0,0,0,1))
is_affine3D(x)

if(interactive() && ants_available()) {

  ants <- load_ants()
  x <- ants$new_ants_transform(dimension = 3L)
  is_affine3D(x)

  # save the parameters
  f <- tempfile(fileext = ".mat")
  ants$write_transform(x, f)
  is_affine3D(f)

}
```

py

Get 'Python' main process environment

Description

Get 'Python' main process environment

Usage

py

Format

An object of class `python.builtin.module` (inherits from `python.builtin.object`) of length 1.

Value

The 'Python' main process as a module

py_builtin

Get 'Python' built-in object

Description

Get 'Python' built-in object

Usage

```
py_builtin(name, convert = TRUE)
```

Arguments

name	object name
convert	see import_builtins

Value

A python built-in object specified by name

Examples

```
if(interactive() && ants_available()) {  
  
# ----- Basic case: use python `int` as an R function -----  
py_int <- py_builtin("int")  
  
# a is an R object now  
a <- py_int(9)  
print(a)  
class(a)  
  
# ----- Use python `int` as a Python function -----  
py_int2 <- py_builtin("int", convert = FALSE)  
  
# b in a python object  
b <- py_int2(9)  
  
# There is no '[1]' when printing  
print(b)  
class(b)  
  
# convert to R object  
py_to_r(b)
```

```
}
```

py_list	<i>List in 'Python'</i>
---------	-------------------------

Description

List in 'Python'

Usage

```
py_list(..., convert = FALSE)
```

Arguments

...	passing to list ('Python')
convert	whether to convert the results back into R; default is no

Value

List instance, or an R vector if converted

Examples

```
if(interactive() && ants_available()) {  
  py_list(list(1,2,3))  
  py_list(c(1,2,3))  
  
  py_list(array(1:9, c(3,3)))  
  py_list(list(list(1:3), letters[1:3]))  
}
```

`py_slice`*Slice index in 'Python' arrays*

Description

Slice index in 'Python' arrays

Usage

```
py_slice(...)
```

Arguments

```
...           passing to slice ('Python')
```

Value

Index slice instance

Examples

```
if(interactive() && ants_available()) {  
  x <- np_array(array(seq(20), c(4, 5)))  
  
  # equivalent to x[::2]  
  x[py_slice(NULL, NULL, 2L)]  
  
}
```


Index

* datasets

py, [13](#)

ants, [2](#)

ants_available, [2](#)

ants_plot, [3](#)

ants_plot_grid, [5](#)

ants_registration, [7](#)

as_ANTsImage, [8](#), [9](#)

as_ANTsTransform, [10](#)

configure_conda, [12](#)

import_builtins, [14](#)

install_ants, [3](#), [11](#)

is_affine3D, [12](#)

load_ants (ants), [2](#)

py, [13](#)

py_builtin, [14](#)

py_list, [15](#)

py_slice, [16](#)