pybraincompare Documentation

Release 1.0

Vanessa Sochat

November 23, 2015

Contents

1 compare **3**

2 annotate **5**

3 network **7**

4 QA for Statistical Maps **9**

5 histogram **11**
5.1 Installation 11
5.2 pybraincompare 11

6 Indices and tables **19**

Python Module Index **21**

Semantic and computational comparison methods for brain imaging data, and visualization of outputs. Modules include:

CHAPTER 1

compare

An example scatterplot image comparison, dynamically rendered using python and d3 from two raw statstical brain maps and an atlas image, [is available](http://vbmis.com/bmi/share/neurovault/scatter_atlas.html). A new addition (beta) is a [canvas based scatterplot](http://vbmis.com/bmi/project/brainatlas) that can render 150K + points.

CHAPTER 2

annotate

This module will let you convert a triples data structure defining relationships in an ontology to a an interactive d3 visualization, demo is [is available](http://vbmis.com/bmi/share/neurovault/ontology_tree.html). Reverse inference tree also [in development](http://vbmis.com/bmi/share/neurovault/reverse_inference.html).

CHAPTER 3

network

This module will work with visualization of functional connectivity data, demo is [is available](http://vbmis.com/bmi/share/neurovault/connectogram.html) and see examples folder for how to run with your data.

CHAPTER 4

QA for Statistical Maps

This module will generate a web report for a list of statistical maps, demo [is available](http://www.vbmis.com/bmi/project/qa/index.html). Much work to be done! Please submit an issue if you have feedback.

CHAPTER 5

histogram

An example histogram using python and chartJS from a sigle brain map [is available](http://vbmis.com/bmi/share/neurovault/histogram.html).

Contents:

5.1 Installation

5.1.1 manual Installation

```
git clone https://github.com/vsoch/pybraincompare
cd pybraincompare
sudo python setup.py install
```

5.1.2 pip Installation

```
pip install pybraincompare
```

5.2 pybraincompare

5.2.1 pybraincompare package

Subpackages

pybraincompare.annotate package

Submodules

pybraincompare.annotate.JSONEncoder module

class pybraincompare.annotate.JSONEncoder.**Node**(*nid*, *parent*, *name*, *meta=None*)
Bases: object

class pybraincompare.annotate.JSONEncoder.**NodeDict**(*dict=None*, ***kwargs*)
Bases: UserDict.UserDict

addNodes(*nodes*)
Add every node as a child to its parent by doing two passes.

class pybraincompare.annotate.JSONEncoder.**NodeJSONEncoder**(*skipkeys=False*, *ensure_ascii=True*, *check_circular=True*, *allow_nan=True*, *sort_keys=False*, *indent=None*, *separators=None*, *encoding='utf-8'*, *default=None*)

Bases: json.encoder.JSONEncoder

default(*node*)

pybraincompare.annotate.ontology module

pybraincompare.annotate.ontology.**make_ontology_tree_d3**(*data_structure*)
pybraincompare.annotate.ontology.**make_reverse_inference_tree_d3**(*data_structure*)
pybraincompare.annotate.ontology.**named_ontology_tree_from_tsv**(*relationship_table*, *output_json=None*, *meta_data=None*)
pybraincompare.annotate.ontology.**ontology_tree_from_tsv**(*relationship_table*, *output_json=None*)

Module contents

pybraincompare.compare package

Submodules

pybraincompare.compare.atlas module atlas.py: part of pybraincompare package Functions to integrate atlases in image comparison

class pybraincompare.compare.atlas.**atlas**(*atlas_xml*, *atlas_file*, *views=['axial', 'sagittal', 'coronal']*)

get_region_names()

get_static_svg()

make_color_lookup(*new_colors*)

make_svg(*views*)

read_xml(*atlas_xml*)

remove_attributes(*path*, *attributes*)

save_svg(*output_folder*, *views=None*)

set_attributes(*path*, *attributes*, *new_values*)

class pybraincompare.compare.atlas.**region**(*label*, *index*, *x*, *y*, *z*)

pybraincompare.compare.maths module maths.py: part of pybraincompare package Simple math functions

pybraincompare.compare.maths.**calc_rows_columns**(*ratio*, *n_images*)

pybraincompare.compare.maths.**calculate_atlas_correlation**(*image_vector1*, *image_vector2*, *atlas_vector*, *atlas_labels*, *atlas_colors*, *corr_type='pearson'*, *summary=False*)

pybraincompare.compare.maths.**calculate_correlation**(*images*, *mask=None*, *atlas=None*, *summary=False*, *corr_type='pearson'*)

pybraincompare.compare.maths.**calculate_pairwise_correlation**(*image_vector1*, *image_vector2*, *corr_type='pearson'*, *atlas_vector=None*)

pybraincompare.compare.maths.**do_multi_correlation**(*image_df*, *corr_type='pearson'*)

pybraincompare.compare.maths.**percent_to_float**(*x*)

pybraincompare.compare.mrutils module mrutils.py: part of pybraincompare package Functions work with brain maps

pybraincompare.compare.mrutils.**apply_threshold**(*image1*, *thresh*, *direction='posneg'*)

pybraincompare.compare.mrutils.**do_mask**(*images*, *mask*)

pybraincompare.compare.mrutils.**generate_thresholds**(*lower=0*, *upper=4*, *by=0.01*)

pybraincompare.compare.mrutils.**get_nii_obj**(*images*)

pybraincompare.compare.mrutils.**get_standard_brain**(*software*)

pybraincompare.compare.mrutils.**get_standard_mask**(*software*)

pybraincompare.compare.mrutils.**get_standard_mat**(*software*)

pybraincompare.compare.mrutils.**make_binary_deletion_mask**(*images*)

pybraincompare.compare.mrutils.**make_binary_deletion_vector**(*image_vectors*)

pybraincompare.compare.mrutils.**make_in_out_mask**(*mask_bin*, *mr_folder*, *masked_in*, *masked_out*, *img_dir*, *save_png=True*)

pybraincompare.compare.mrutils.**resample_images_ref**(*images*, *reference*, *interpolation*, *resample_dim=None*)

pybraincompare.compare.mrutils.**squeeze_fourth_dimension**(*images*)

pybraincompare.compare.network module network.py: part of pybraincompare package Functions for visualization of functional MRI

pybraincompare.compare.network.**connectogram**(*matrix_file*, *groups*, *threshold*, *network_names=None*)

pybraincompare.compare.scatterplot module scatterplot.py: part of pybraincompare package Functions to perform and create scatterplot comparisons

pybraincompare.compare.scatterplot.**get_atlas_objects**(*atlas*, *atlas_rendering*)

pybraincompare.compare.scatterplot.**make_scatterplot_interface**(*corr_df*, *elements*, *error=None*, *remove_scripts=None*)

pybraincompare.compare.scatterplot.**scatterplot_canvas**(*image_vector1*, *image_vector2*, *image_names*, *atlas_vector*, *atlas_labels*, *atlas_colors*, *output_directory*, *view=True*)

pybraincompare.compare.scatterplot.**scatterplot_compare**(*images, image_names, software='FSL', atlas=None, atlas_rendering=None, custom=None, corr_type='pearson', reference=None, resample_dim=[8, 8, 8], remove_scripts=None, width=1200*)

pybraincompare.compare.scatterplot.**scatterplot_compare_error**(*template*, *specific_error*)

pybraincompare.compare.scatterplot.**scatterplot_compare_vector**(*image_vector1*, *image_vector2*, *image_names*, *atlas_vector*, *atlas_labels*, *atlas_colors*, *custom=None*, *corr_type='pearson'*, *atlas=None*, *subsample_every=None*, *remove_scripts=None*, *summary=False*, *width=1200*)

pybraincompare.compare.search module search.py: part of pybraincompare package Generate search interfaces to compare images

pybraincompare.compare.search.**calculate_similarity_search**(*template*, *query_png*, *query_id*, *corr_df*, *button_url*, *image_url*, *max_results*, *absolute_value*, *container_width*, *responsive=True*)

calculate_similarity_search_df starts with pandas data frame to make similarity interface

pybraincompare.compare.search.**create_glassbrain_portfolio**(*image_paths*, *all_tags*, *unique_tags*, *placeholders*, *values=None*, *button_urls=None*, *image_urls=None*, *top_text=None*, *bottom_text=None*)

pybraincompare.compare.search.**similarity_search**(*image_scores*, *tags*, *png_paths*, *query_png*, *query_id*, *button_url*, *image_url*, *image_ids*, *max_results=100*, *absolute_value=True*, *top_text=None*, *bottom_text=None*, *container_width=940*, *remove_scripts=None*)

Module contents

pybraincompare.mr package

Submodules

pybraincompare.mr.datasets module datasets.py: part of pybraincompare package Return sets of images or atlas files

pybraincompare.mr.datasets.**get_data_directory**()

pybraincompare.mr.datasets.**get_mni_atlas**(*voxdims=['2', '8'], views=None*)

pybraincompare.mr.datasets.**get_pair_images**(*voxdims=['2', '2']*)

pybraincompare.mr.datasets.**get_standard_brain**(*voxdim=2*)

pybraincompare.mr.datasets.**get_standard_mask**(*voxdim=2*)

pybraincompare.mr.transformation module transformation.py: part of pybraincompare package Return transformations of images

pybraincompare.mr.transformation.**make_resampled_transformation**(*nii_obj*, *resample_dim=[4, 4, 4]*, *standard_mask=True*)

pybraincompare.mr.transformation.**make_resampled_transformation_vector**(*nii_obj*, *resample_dim=[4, 4, 4]*, *standard_mask=True*)

Module contents

pybraincompare.report package

Submodules

pybraincompare.report.animate module

pybraincompare.report.animate.**animate_figure**()

pybraincompare.report.colors module colors.py: part of pybraincompare package Color stuffs

pybraincompare.report.colors.**get_colors**(*N*, *color_format='decimal'*)

pybraincompare.report.colors.**peterson_roi_labels**(*colors=True*)

pybraincompare.report.colors.**random_colors**(*N*)

pybraincompare.report.histogram module histogram.py: part of pybraincompare package Functions for histograms

pybraincompare.report.histogram.**get_histogram_data**(*data*, *width=12*, *height=4*, *color=None*, *ylabel='frequency'*, *xlabel='map intensity value bins'*, *title='Histogram of Intensity Values for Image'*, *bins=25*, *remove_zeros=True*)

pybraincompare.report.histogram.**histogram_image**(*masked_data*, *remove_zero=False*, *title=None*, *line_value=None*, *xlabel=None*, *width=11*, *height=4*, *png_img_file=None*, *threshold=0.001*)

pybraincompare.report.histogram.**plot_histogram**(*image*, *title='Image Histogram'*, *height=400*, *width=1000*, *view_in_browser=True*, *bins=25*, *remove_zeros=True*)

pybraincompare.report.image module image.py: part of pybraincompare package Functions for static images

pybraincompare.report.image.**make_anat_image**(*nifti_file*, *png_img_file=None*)

pybraincompare.report.image.**make_glassbrain_image**(*nifti_file*, *png_img_file=None*)

pybraincompare.report.image.**make_roi_image**(*nifti_file*, *png_img_file=None*)

pybraincompare.report.image.**make_stat_image**(*nifti_file*, *png_img_file=None*)

pybraincompare.report.image.**plot_vline**(*cur_val*, *label*, *ax*)

pybraincompare.report.qa module qa.py: part of pybraincompare package Functions to check quality of statistical maps

pybraincompare.report.qa.**central_tendency**(*data*)

pybraincompare.report.qa.**count_voxels**(*masked_in*, *masked_out*)

pybraincompare.report.qa.**get_percent_nonzero**(*masked_in*)

pybraincompare.report.qa.**get_voxel_range**(*nii_obj*)

pybraincompare.report.qa.**header_metrics**(*image*)

pybraincompare.report.qa.**is_only_positive**(*nii_obj*)

pybraincompare.report.qa.**is_thresholded**(*nii_obj*, *brain_mask*, *threshold=0.95*)

pybraincompare.report.qa.**outliers**(*masked_data*, *n_std=6*)

pybraincompare.report.qa.**t_to_z**(*mr*, *dof*)

pybraincompare.report.webreport module webreport.py: part of pybraincompare package Functions to generate reports using qa tools

pybraincompare.report.webreport.**run_qa**(*mr_paths, html_dir, software='FSL', voxdim=[2, 2, 2], outlier_sds=6, investigator='brainman', nonzero_thresh=0.25, calculate_mean_image=True, view=True*)

Module contents

pybraincompare.template package

Submodules

pybraincompare.template.futils module

pybraincompare.template.futils.**get_name**(*path*)

pybraincompare.template.futils.**get_package_dir**()

pybraincompare.template.futils.**make_dir**(*directory*)

pybraincompare.template.futils.**make_png_paths**(*nifti_files*)

pybraincompare.template.futils.**make_tmp_folder**(**args*, ***kwds*)

pybraincompare.template.futils.**unwrap_list_unique**(*list_of_lists*)

pybraincompare.template.futils.**unzip**(*source*, *dest_dir*)

pybraincompare.template.templates module templates.py: part of pybraincompare package Functions to work with html templates

pybraincompare.template.templates.**add_javascript_function**(*function_code*, *template*)

pybraincompare.template.templates.**add_string**(*svg*, *template*)

pybraincompare.template.templates.**get_image**(*image_name*)

pybraincompare.template.templates.**get_template**(*html_name*, *data_frame=None*)

pybraincompare.template.templates.**read_template**(*html_name*)

pybraincompare.template.templates.**remove_resources**(*html_snippet*, *script_names*)

pybraincompare.template.templates.**save_template**(*html_snippet*, *output_file*)

pybraincompare.template.visual module visual.py: part of pybraincompare package Functions to visualize in browser

pybraincompare.template.visual.**get_svg_html**(*mpl_figures*)

pybraincompare.template.visual.**internal_view**(*html_snippet*, *tmp_file*)

pybraincompare.template.visual.**run_webserver**(*PORT=8000*, *html_page='index.html'*)

pybraincompare.template.visual.**view**(*html_snippet*)

Module contents

pybraincompare.testing package

Submodules

pybraincompare.testing.test_connectogram module

pybraincompare.testing.test_correlation module Test regional and whole brain correlation scores

pybraincompare.testing.test_correlation.**test_simulated_correlations**()

pybraincompare.testing.test_histogram module Test histogram output

pybraincompare.testing.test_histogram.**test_histogram_output**()

pybraincompare.testing.test_masking module Test that pairwise deletion mask (intersection) returns expected values

pybraincompare.testing.test_masking.**test_binary_deletion_mask**()

pybraincompare.testing.test_masking.**test_binary_deletion_mask_values**()

pybraincompare.testing.test_masking.**test_binary_deletion_vector**()

pybraincompare.testing.test_scatterplot_compare module Test scatterplot compare output

pybraincompare.testing.test_scatterplot_compare.**test_scatterplot_error_message**()

pybraincompare.testing.test_transformation module Test transformation functions

pybraincompare.testing.test_transformation.**test_masked_transformation**()

pybraincompare.testing.test_transformation.**test_unmasked_transformation**()

Module contents

Module contents

CHAPTER 6

Indices and tables

- genindex
- modindex
- search

p

pybraincompare, 18
pybraincompare.annotate, 12
pybraincompare.annotate.JSONEncoder, 11
pybraincompare.annotate.ontology, 12
pybraincompare.compare, 15
pybraincompare.compare.atlas, 12
pybraincompare.compare.maths, 13
pybraincompare.compare.mrutils, 13
pybraincompare.compare.network, 13
pybraincompare.compare.scatterplot, 14
pybraincompare.compare.search, 14
pybraincompare.mr, 15
pybraincompare.mr.datasets, 15
pybraincompare.mr.transformation, 15
pybraincompare.report, 17
pybraincompare.report.animate, 16
pybraincompare.report.colors, 16
pybraincompare.report.histogram, 16
pybraincompare.report.image, 16
pybraincompare.report.qa, 16
pybraincompare.report.webreport, 17
pybraincompare.template, 18
pybraincompare.template.futils, 17
pybraincompare.template.templates, 17
pybraincompare.template.visual, 18
pybraincompare.testing, 18
pybraincompare.testing.test_correlation, 18
pybraincompare.testing.test_histogram, 18
pybraincompare.testing.test_masking, 18
pybraincompare.testing.test_scatterplot_compare, 18
pybraincompare.testing.test_transformation, 18

A

add_javascript_function() (in module pybraincompare.template.templates), 17
add_string() (in module pybraincompare.template.templates), 17
addNodes() (pybraincompare.annotate.JSONEncoder.NodeDict method), 11
animate_figure() (in module pybraincompare.report.animate), 16
apply_threshold() (in module pybraincompare.compare.mrutils), 13
atlas (class in pybraincompare.compare.atlas), 12

C

calc_rows_columns() (in module pybraincompare.compare.maths), 13
calculate_atlas_correlation() (in module pybraincompare.compare.maths), 13
calculate_correlation() (in module pybraincompare.compare.maths), 13
calculate_pairwise_correlation() (in module pybraincompare.compare.maths), 13
calculate_similarity_search() (in module pybraincompare.compare.search), 14
central_tendency() (in module pybraincompare.report.qa), 16
connectogram() (in module pybraincompare.compare.network), 13
count_voxels() (in module pybraincompare.report.qa), 16
create_glassbrain_portfolio() (in module pybraincompare.compare.search), 14

D

default() (pybraincompare.annotate.JSONEncoder.NodeJSONEncoder method), 12
do_mask() (in module pybraincompare.compare.mrutils), 13
do_multi_correlation() (in module pybraincompare.compare.maths), 13

G

generate_thresholds() (in module pybraincompare.compare.mrutils), 13
get_atlas_objects() (in module pybraincompare.compare.scatterplot), 14
get_colors() (in module pybraincompare.report.colors), 16
get_data_directory() (in module pybraincompare.mr.datasets), 15
get_histogram_data() (in module pybraincompare.report.histogram), 16
get_image() (in module pybraincompare.template.templates), 17
get_mni_atlas() (in module pybraincompare.mr.datasets), 15
get_name() (in module pybraincompare.template.futils), 17
get_nii_obj() (in module pybraincompare.compare.mrutils), 13
get_package_dir() (in module pybraincompare.template.futils), 17
get_pair_images() (in module pybraincompare.mr.datasets), 15
get_percent_nonzero() (in module pybraincompare.report.qa), 16
get_region_names() (pybraincompare.compare.atlas.atlas method), 12
get_standard_brain() (in module pybraincompare.compare.mrutils), 13
get_standard_brain() (in module pybraincompare.mr.datasets), 15
get_standard_mask() (in module pybraincompare.compare.mrutils), 13
get_standard_mask() (in module pybraincompare.mr.datasets), 15
get_standard_mat() (in module pybraincompare.compare.mrutils), 13
get_static_svg() (pybraincompare.compare.atlas.atlas method), 12

get_svg_html() (in module pybraincompare.template.visual), 18
get_template() (in module pybraincompare.template.templates), 17
get_voxel_range() (in module pybraincompare.report.qa), 16

H

header_metrics() (in module pybraincompare.report.qa), 16
histogram_image() (in module pybraincompare.report.histogram), 16

I

internal_view() (in module pybraincompare.template.visual), 18
is_only_positive() (in module pybraincompare.report.qa), 17
is_thresholded() (in module pybraincompare.report.qa), 17

M

make_anat_image() (in module pybraincompare.report.image), 16
make_binary_deletion_mask() (in module pybraincompare.compare.mrutils), 13
make_binary_deletion_vector() (in module pybraincompare.compare.mrutils), 13
make_color_lookup() (pybraincompare.compare.atlas.atlas method), 12
make_dir() (in module pybraincompare.template.futils), 17
make_glassbrain_image() (in module pybraincompare.report.image), 16
make_in_out_mask() (in module pybraincompare.compare.mrutils), 13
make_ontology_tree_d3() (in module pybraincompare.annotate.ontology), 12
make_png_paths() (in module pybraincompare.template.futils), 17
make_resampled_transformation() (in module pybraincompare.mr.transformation), 15
make_resampled_transformation_vector() (in module pybraincompare.mr.transformation), 15
make_reverse_inference_tree_d3() (in module pybraincompare.annotate.ontology), 12
make_roi_image() (in module pybraincompare.report.image), 16
make_scatterplot_interface() (in module pybraincompare.compare.scatterplot), 14
make_stat_image() (in module pybraincompare.report.image), 16
make_svg() (pybraincompare.compare.atlas.atlas method), 12
make_tmp_folder() (in module pybraincompare.template.futils), 17

N

named_ontology_tree_from_tsv() (in module pybraincompare.annotate.ontology), 12
Node (class in pybraincompare.annotate.JSONEncoder), 11
NodeDict (class in pybraincompare.annotate.JSONEncoder), 11
NodeJSONEncoder (class in pybraincompare.annotate.JSONEncoder), 12

O

ontology_tree_from_tsv() (in module pybraincompare.annotate.ontology), 12
outliers() (in module pybraincompare.report.qa), 17

P

percent_to_float() (in module pybraincompare.compare.maths), 13
peterson_roi_labels() (in module pybraincompare.report.colors), 16
plot_histogram() (in module pybraincompare.report.histogram), 16
plot_vline() (in module pybraincompare.report.image), 16
pybraincompare (module), 18
pybraincompare.annotate (module), 12
pybraincompare.annotate.JSONEncoder (module), 11
pybraincompare.annotate.ontology (module), 12
pybraincompare.compare (module), 15
pybraincompare.compare.atlas (module), 12
pybraincompare.compare.maths (module), 13
pybraincompare.compare.mrutils (module), 13
pybraincompare.compare.network (module), 13
pybraincompare.compare.scatterplot (module), 14
pybraincompare.compare.search (module), 14
pybraincompare.mr (module), 15
pybraincompare.mr.datasets (module), 15
pybraincompare.mr.transformation (module), 15
pybraincompare.report (module), 17
pybraincompare.report.animate (module), 16
pybraincompare.report.colors (module), 16
pybraincompare.report.histogram (module), 16
pybraincompare.report.image (module), 16
pybraincompare.report.qa (module), 16
pybraincompare.report.webreport (module), 17
pybraincompare.template (module), 18
pybraincompare.template.futils (module), 17
pybraincompare.template.templates (module), 17
pybraincompare.template.visual (module), 18
pybraincompare.testing (module), 18
pybraincompare.testing.test_correlation (module), 18

pybraincompare.testing.test_histogram (module), 18
pybraincompare.testing.test_masking (module), 18
pybraincompare.testing.test_scatterplot_compare (module), 18
pybraincompare.testing.test_transformation (module), 18

R

random_colors() (in module pybraincompare.report.colors), 16
read_template() (in module pybraincompare.template.templates), 17
read_xml() (pybraincompare.compare.atlas.atlas method), 12
region (class in pybraincompare.compare.atlas), 12
remove_attributes() (pybraincompare.compare.atlas.atlas method), 12
remove_resources() (in module pybraincompare.template.templates), 17
resample_images_ref() (in module pybraincompare.compare.mrutils), 13
run_qa() (in module pybraincompare.report.webreport), 17
run_webserver() (in module pybraincompare.template.visual), 18

S

save_svg() (pybraincompare.compare.atlas.atlas method), 12
save_template() (in module pybraincompare.template.templates), 17
scatterplot_canvas() (in module pybraincompare.compare.scatterplot), 14
scatterplot_compare() (in module pybraincompare.compare.scatterplot), 14
scatterplot_compare_error() (in module pybraincompare.compare.scatterplot), 14
scatterplot_compare_vector() (in module pybraincompare.compare.scatterplot), 14
set_attributes() (pybraincompare.compare.atlas.atlas method), 12
similarity_search() (in module pybraincompare.compare.search), 15
squeeze_fourth_dimension() (in module pybraincompare.compare.mrutils), 13

T

t_to_z() (in module pybraincompare.report.qa), 17
test_binary_deletion_mask() (in module pybraincompare.testing.test_masking), 18
test_binary_deletion_mask_values() (in module pybraincompare.testing.test_masking), 18
test_binary_deletion_vector() (in module pybraincompare.testing.test_masking), 18
test_histogram_output() (in module pybraincompare.testing.test_histogram), 18
test_masked_transformation() (in module pybraincompare.testing.test_transformation), 18
test_scatterplot_error_message() (in module pybraincompare.testing.test_scatterplot_compare), 18
test_simulated_correlations() (in module pybraincompare.testing.test_correlation), 18
test_unmasked_transformation() (in module pybraincompare.testing.test_transformation), 18

U

unwrap_list_unique() (in module pybraincompare.template.futils), 17
unzip() (in module pybraincompare.template.futils), 17

V

view() (in module pybraincompare.template.visual), 18