# Package 'melviewr'

December 4, 2016

**Type** Package

**Title** View and Classify MELODIC Output for ICA+FIX

**Version** 0.0.0.9000

**Description** The melviewr package provides a graphical interface that allows the user to easily view and classify MELODIC output for the purposes of later running ICA+FIX. The user categorizes a component as signal or noise based on its spatial characteristics as well as its temporal profile. melviewr can then save a text file of these classifications in the format required by ICA+FIX.

**URL** https://github.com/AndrewPoppe/melviewr

**BugReports** https://github.com/AndrewPoppe/melviewr/issues

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**Imports** gtools, RColorBrewer, RNifti, grDevices, RGtk2, cairoDevice, methods, jsonlite

**Depends** gWidgetsRGtk2, gWidgets

**RoxygenNote** 5.0.1

**NeedsCompilation** no

**Author** Andrew Poppe [aut, cre]

**Maintainer** Andrew Poppe <Poppe076@gmail.com>

## R topics documented:

`melviewr-package` *melviewr: A viewer for MELODIC output and ICA+FIX classification.*

## Description

The melviewr package allows the user to easily view and classify MELODIC output for the purposes of later running ICA+FIX. The user categorizes a component as signal or noise based on its spatial characteristics as well as its temporal profile. melviewr can then save a text file of these classifications in the format required by ICA+FIX.

## melviewr functions

melviewr

`melviewr` *melviewr: View and Classify Components from a Melodic Analysis*

## Description

The melviewr GUI allows for convenient viewing and classification of the results of a single-subject MELODIC analysis. Classification can then be saved to a text file for use by ICA+FIX to train its classifier. Various graphics options are available in the GUI, and these settings can be saved via a button in the GUI.

## Usage

```
melviewr(melodic_dir, standard_file = NULL, motion_file = NULL)
```

## Arguments

| | |
|---|---|
| `melodic_dir` | string Path to MELODIC output directory. This directory must include a melodic_IC.nii or melodic_IC.nii.gz file. |
| `standard_file` | string Optional path to a 3-dimensional Nifti standard file of the same voxel dimensions as the melodic output |
| `motion_file` | string Optional path to a summary motion text file. This file should have one column and as many rows as there are volumes in the functional data |

## Details

The directory specified in `melodic_dir` must contain a nifti file called either "melodic_IC.nii.gz" or "melodic_IC.nii" for the GUI to run. It must have a directory called "report" with text files inside in order to display timecourse and powerspectrum plots. Normally, this directory is created automatically with the `-report` flag in MELODIC.

When saving graphical settings, a JSON file is saved in the user's HOME directory with the name: `.melviewR.config`

## Value

Invisibly returns a reference class object of class "Viewr"

## Examples

```
## Not run:
melodic_dir <- system.file("extdata", "example.ica", package = "melviewr")
standard_file <- system.file("extdata", "MNI152_T1_2mm_brain.nii.gz", package = "melviewr")
motion_file <- system.file("extdata", "Movement_RelativeRMS.txt", package = "melviewr")
melviewr(melodic_dir)
melviewr(melodic_dir, standard_file)
melviewr(melodic_dir, standard_file, motion_file)
## End(Not run)
```

# Index