

# Package ‘tacmagic’

June 7, 2019

**Type** Package

**Title** Positron Emission Tomography Time-Activity Curve Analysis

**Version** 0.3.0

**Description** To facilitate the analysis of positron emission tomography (PET) time activity curve (TAC) data, and to encourage open science and replicability, this package supports data loading and analysis of multiple TAC file formats. Functions are available to analyze loaded TAC data for individual participants or in batches. Major functionality includes weighted TAC merging by region of interest (ROI), calculating models including standardized uptake value ratio (SUVR) and distribution volume ratio (DVR, Logan et al. 1996 <doi:10.1097/00004647-199609000-00008>), basic plotting functions and calculation of cut-off values (Aizenstein et al. 2008 <doi:10.1001/archneur.65.11.1509>). Please see the walkthrough vignette for a detailed overview of 'tacmagic' functions.

**Depends** R (>= 3.4)

**License** GPL-3

**URL** <https://github.com/ropensci/tacmagic>

**BugReports** <https://github.com/ropensci/tacmagic>

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.1.1

**Imports** graphics, grDevices, pracma, utils, R.matlab

**Suggests** testthat, knitr, rmarkdown, covr, vdiff

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Eric Brown [aut, cre] (<<https://orcid.org/0000-0002-1575-2606>>),  
Ariel Graff-Guerrero [dgs],  
Jon Clayden [rev] (<<https://orcid.org/0000-0002-6608-0619>>, Jon Clayden reviewed the package for ropensci, see <<https://github.com/ropensci/software-review/issues/280>>),  
Brandon Hurr [rev] (<<https://orcid.org/0000-0003-2576-4544>>, Brandon

Hurr reviewed the package for ropensci, see  
 <<https://github.com/ropensci/software-review/issues/280>>)

**Maintainer** Eric Brown <eb@ericebrown.com>

**Repository** CRAN

**Date/Publication** 2019-06-07 18:40:02 UTC

## R topics documented:

as.tac	2
batch_load	3
batch_tm	5
batch_voistat	6
change_units	7
cutoff_aiz	7
dvr	8
DVR_all_ref_Logan	10
DVR_ref_Logan	11
fake_DVR	12
load_tac	12
load_voistat	14
load_vol	14
plot.ref_Logan	15
plot.tac	16
pos_anyroi	17
roi_ham_full	17
roi_ham_pib	18
roi_ham_stand	19
save_tac	20
split_pvc	20
suv	21
suvr	22
suvr_auc	23
tacmagic	24
tac_roi	24
tac_suv	25
<b>Index</b>	<b>26</b>

---

as.tac	<i>Creates a tac object from a data.frame</i>
--------	---

---

### Description

tac objects can be created from data.frame objects with 'as.tac()'. The time and activity units must be specified as arguments if not already set as attributes in the data.frame. The columns of the data frame are the regional time activity curves, with the column names the names of the ROIs.

**Usage**

```
as.tac(x, time_unit = NULL, activity_unit = NULL)
```

**Arguments**

x                      data.frame with start, end time and tac data  
time\_unit              NULL if in data.frame or set to "seconds" or "minutes"  
activity\_unit          NULL if in data.frame or set to "kBq/cc", "Bq/cc", "nCi/cc"

**Details**

If the time\_unit and activity\_unit attributes are already in the data.frame, they do not need to be set again, but otherwise they will need to be specified in the input parameters.

**Value**

tac object

**See Also**

Other Loading functions: [load\\_tac](#), [load\\_voistat](#), [load\\_vol](#)

**Examples**

```
manual <- data.frame(start=c(0:4), end=c(2:6),
                     ROI1=c(10.1:14.2), ROI2=c(11:15))
manual_tac <- as.tac(manual, time_unit="minutes", activity_unit="kBq/cc")
```

---

batch_load	<i>Load (+/- merge) ROIs for batch of participants</i>
------------	--

---

**Description**

For a vector of participant IDs and correspondingly named tac files, this loads the tac files. If roi\_m = T, then can also merge ROIs into larger ROIs based on the optional parameters that follow.

**Usage**

```
batch_load(participants, dir = "", tac_file_suffix = ".tac",
           tac_format = "PMOD", roi_m = FALSE, PVC = NULL,
           vol_file_suffix = NULL, vol_format = NULL, merge = NULL,
           ROI_def = NULL, tracer_dose = NULL, dose_unit = NULL,
           weight_kg = NULL)
```

**Arguments**

participants	A vector of participant IDs
dir	A directory and/or file name prefix for the tac/volume files
tac_file_suffix	How participant IDs corresponds to the TAC files
tac_format	Format of tac files provided: See load_tac()
roi_m	TRUE if you want to merge atomic ROIs into larger ROIs (and if not, the following parameters are not used)
PVC	For PVC, true where the data is stored as _C in same tac file
vol_file_suffix	How participant IDs correspond to volume files
vol_format	The file format that includes volumes: See load_vol()
merge	Passes value to tac_roi(); T to also incl. original atomic ROIs
ROI_def	Object that defines combined ROIs, see ROI_definitions.R
tracer_dose	optionally, a vector of tracer doses (in the same order as participants), for SUV
dose_unit	if tracer_dose is specified, note the unit (e.g "MBq")
weight_kg	optionally, a vector of participant weights in kg, for SUV

**Details**

See load\_tac() for specifics.

**Value**

A list of data.frames, each is a participant's TACs

**See Also**

Other Batch functions: [batch\\_tm](#), [batch\\_voistat](#)

**Examples**

```
# For the working example, the participants are full filenames.
participants <- c(system.file("extdata", "AD06.tac", package="tacmagic"),
                 system.file("extdata", "AD07.tac", package="tacmagic"),
                 system.file("extdata", "AD08.tac", package="tacmagic"))

tacs <- batch_load(participants, tac_file_suffix="")
```

---

batch_tm	<i>Calculate one or more models for a batch of participants</i>
----------	---

---

### Description

For a list of tac data (from `load_batch`) this calculates specified models and saves in a tidy data.frame. Current model options are "SUVR", "Logan".

### Usage

```
batch_tm(all_tacs, models, custom_model = NULL, ...)
```

### Arguments

<code>all_tacs</code>	A list by participant, of tac data ( <code>load_batch()</code> )
<code>models</code>	A vector of names of the models to calculate
<code>custom_model</code>	A function that can be run like other models (advanced)
<code>...</code>	The arguments that get passed to the specified models/custom model, many are required; please check with model desired.

### Details

For further details about how the models are calculated, see the individual functions that they rely on. "SUVR" uses `suvr()`, "Logan" uses `DVR_all_ref_Logan()`.

### Value

A table of SUVR values for the specified ROIs for all participants

### See Also

Other Batch functions: [batch\\_load](#), [batch\\_voistat](#)

### Examples

```
participants <- c(system.file("extdata", "AD06.tac", package="tacmagic"),
                 system.file("extdata", "AD07.tac", package="tacmagic"),
                 system.file("extdata", "AD08.tac", package="tacmagic"))

tacs <- batch_load(participants, tac_file_suffix="")

# Keeps only the ROIs without partial-volume correction (PMOD convention)
tacs <- lapply(tacs, split_pvc, FALSE)

batch <- batch_tm(tacs, models=c("SUVR", "Logan"), ref="Cerebellum_r",
                 SUVR_def=c(3000,3300,3600), k2prime=0.2, t_star=23)
```

---

batch_voistat	<i>Obtain values from voistat files (using load_voistat()) for a batch.</i>
---------------	---

---

### Description

For a vector of participant IDs and correspondingly named .voistat files, this extracts the value from the files for the specified ROIs. participants can also be a vector of filenames, in which case set dir="" and filesuffix="", as in the example.

### Usage

```
batch_voistat(participants, ROI_def, dir = "", filesuffix = ".voistat",
              varname = "VALUE")
```

### Arguments

participants	A vector of participant IDs
ROI_def	Object that defines combined ROIs, see ROI_definitions.R
dir	Directory and/or filename prefix of the files
filesuffix	Optional filename characters between ID and ".voistat"
varname	The name of the variable being extracted, e.g. "SRTM"

### Details

See load\_voistat() for specifics.

### Value

A table of values for the specified ROIs for all participants

### See Also

Other Batch functions: [batch\\_load](#), [batch\\_tm](#)

### Examples

```
participants <- c(system.file("extdata", "AD06_BPnd_BPnd_Logan.voistat",
                             package="tacmagic"),
                 system.file("extdata", "AD07_BPnd_BPnd_Logan.voistat",
                             package="tacmagic"),
                 system.file("extdata", "AD08_BPnd_BPnd_Logan.voistat",
                             package="tacmagic"))

batchtest <- batch_voistat(participants=participants, ROI_def=roi_ham_pib(),
                          dir="", filesuffix="", varname="Logan")
```

---

change_units	<i>Convert radioactivity units</i>
--------------	------------------------------------

---

**Description**

Change the radioactivity units of a tac or numeric object to the specified desired units (e.g. Bq, kBq, MBq, nCi, uCi, mCi, Ci). For convenience, if the unit is per volume ("x/cc" or "x/mL"), the "/cc" part is ignored for the conversion.

**Usage**

```
change_units(x, to_unit, from_unit)
```

**Arguments**

x	time-activity curve or numeric object
to_unit	the desired unit (e.g. "kBq")
from_unit	not used for tac object (it is in the tac object), but for numeric objects, must be specified (e.g. "nCi")

**Value**

the converted object, same type as x

**Examples**

```
f <- system.file("extdata", "AD06.tac", package="tacmagic")
AD06_tac <- load_tac(f, format="PMOD")
AD06_tac_nCicc <- change_units(AD06_tac, to_unit = "nCi/cc")

change_units(5, to_unit = "kBq", from_unit = "nCi")
change_units(0.185, to_unit = "nCi", from_unit = "kBq")
```

---

cutoff_aiz	<i>Cutoff value calculation using method described in Aizenstein et al. 2008</i>
------------	--

---

**Description**

See the reference below and the tacmagic walkthrough vignette. Aizenstein et al. (2008) proposed a standardized method of calculating Pittsburgh Compound B (PIB) cutoff values to classify participants as PIB+ or PIB-. They used the distribution volume ratio (DVR) from several ROIs associated with amyloid deposition. The steps are summarized below. cutoff\_aiz() implements 1-3, returning cutoff values for each ROI. It can be used to dichotomize participants, with pos\_anyroi().

**Usage**

```
cutoff_aiz(modelstats, ROIs)
```

**Arguments**

modelstats	SUVR or DVR data for group of participants from batch_tm()
ROIs	list of variables (ROIs) to use for cutoff detection

**Details**

1. Remove outliers from a group of cognitively normal individuals. An outlier is defined as having any ROI with DVR > upper inner fence of that ROI (= 3rd quartile + (1.5 \* IQR)). 2. Iterate step 1 as needed until there are no more outlying participants. 3. From this subset of the group with outliers removed, the cutoff value for each ROI is set as the upper inner fence. 4. For all participants, if there is any ROI above the cutoff for that region, then the participant is deemed to be PIB+.

**Value**

Cutoff values for each ROI based on the above method

**References**

Aizenstein HJ, Nebes RD, Saxton JA, et al. 2008. Frequent amyloid deposition without significant cognitive impairment among the elderly. Arch Neurol 65: 1509-1517.

**See Also**

Other Cutoff functions: [pos\\_anyroi](#)

**Examples**

```
cutoff_aiz(fake_DVR, c("ROI1_DVR", "ROI2_DVR", "ROI3_DVR", "ROI4_DVR"))
```

---

dvr

*Distribution volume ratio (DVR) for one or more ROIs*


---

**Description**

This calculates the DVR using the non-invasive reference Logan method for all TACs in a supplied tac file. It uses DVR\_ref\_Logan if a target ROI is specified, otherwise will calculate DVR for all ROIs with DVR\_ref\_all\_Logan()

**Usage**

```
dvr(tac, model = "logan", target = NULL, ref, k2prime, t_star,
    error = 0.1, method = "trapz")
```



**Arguments**

tac	The time-activity curve data from <code>load_tac()</code> or <code>tac_roi()</code>
model	Only model currently available is "logan"
target	Optional - otherwise will calculate DVR for all regions
ref	Required – The reference region, e.g. "cerebellum"
k2prime	Required – A fixed value for k2' must be specified (e.g. 0.2)
t_star	Required – If 0, t* will be calculated using <code>find_t_star()</code>
error	For <code>find_t_star()</code>
method	Method of integration, "trapz" or "integrate"

**Details**

For other model parameters, directly call `DVR_ref_Logan()`.

**Value**

Data frame with calculated DVRs

**References**

Logan, J., Fowler, J. S., Volkow, N. D., Wang, G.-J., Ding, Y.-S., & Alexoff, D. L. (1996). Distribution Volume Ratios without Blood Sampling from Graphical Analysis of PET Data. *Journal of Cerebral Blood Flow & Metabolism*, 16(5), 834-840. <https://doi.org/10.1097/00004647-199609000-00008>

**See Also**

Other Logan plot functions: [DVR\\_all\\_ref\\_Logan](#), [DVR\\_ref\\_Logan](#), [plot.ref\\_Logan](#)

**Examples**

```
f <- system.file("extdata", "AD06.tac", package="tacmagic")
fv <- system.file("extdata", "AD06_TAC.voistat", package="tacmagic")
AD06_tac <- load_tac(f, format="PMOD")
AD06_volume <- load_vol(fv, format="voistat")
AD06 <- tac_roi(tac=AD06_tac, volumes=AD06_volume, ROI_def=roi_ham_pib(),
               merge=FALSE, PVC=FALSE)

AD06_DVRs <- dvr(AD06, ref="cerebellum", k2prime=0.2, t_star=23)

AD06_DVR <- dvr(AD06, target="frontal", ref="cerebellum",
               k2prime=0.2, t_star=23)
```

---

DVR\_all\_ref\_Logan      *Non-invasive reference Logan method for all ROIs in tac data*

---

### Description

This calculates the DVR using the non-invasive reference Logan method for all TACs in a supplied tac file. It uses DVR\_ref\_Logan.

### Usage

```
DVR_all_ref_Logan(tac_data, ref, k2prime, t_star, error = 0.1,
  method = "trapz", ...)
```

### Arguments

tac_data	The time-activity curve data from tac_roi()
ref	Required – The reference region, e.g. "cerebellum"
k2prime	Required – A fixed value for k2' must be specified (e.g. 0.2)
t_star	Required – If 0, t* will be calculated using find_t_star()
error	For find_t_star()
method	Method of integration, "trapz" or "integrate"
...	When called from tm_batch, unused parameters may be supplied

### Value

Data frame with calculated DVRs for all ROIs

### References

Logan, J., Fowler, J. S., Volkow, N. D., Wang, G.-J., Ding, Y.-S., & Alexoff, D. L. (1996). Distribution Volume Ratios without Blood Sampling from Graphical Analysis of PET Data. *Journal of Cerebral Blood Flow & Metabolism*, 16(5), 834-840. <https://doi.org/10.1097/00004647-199609000-00008>

### See Also

Other Logan plot functions: [DVR\\_ref\\_Logan](#), [dvr](#), [plot.ref\\_Logan](#)

### Examples

```
f <- system.file("extdata", "AD06.tac", package="tacmagic")
fv <- system.file("extdata", "AD06_TAC.voistat", package="tacmagic")
AD06_tac <- load_tac(f, format="PMOD")
AD06_volume <- load_vol(fv, format="voistat")
AD06 <- tac_roi(tac=AD06_tac, volumes=AD06_volume, ROI_def=roi_ham_pib(),
  merge=FALSE, PVC=FALSE)
```

```
AD06_DVR <- DVR_all_ref_Logan(AD06, ref="cerebellum", k2prime=0.2, t_star=23)
```

---

DVR_ref_Logan	<i>Non-invasive reference Logan method</i>
---------------	--

---

### Description

This calculates the coefficient from the non-invasive Logan method, which is equal to DVR. Works for a single tac (target).

### Usage

```
DVR_ref_Logan(tac_data, target, ref, k2prime, t_star, error = 0.1,
  method = "trapez")
```

### Arguments

tac_data	The time-activity curve data from tac_roi()
target	The name of the target ROI, e.g. "frontal"
ref	The reference region, e.g. "cerebellum"
k2prime	A fixed value for k2' must be specified (e.g. 0.2)
t_star	If 0, t* will be calculated using find_t_star()
error	For find_t_star()
method	Method of integration, "trapez" or "integrate"

### Value

Data frame with calculate DVRs for all ROIs

### References

Logan, J., Fowler, J. S., Volkow, N. D., Wang, G.-J., Ding, Y.-S., & Alexoff, D. L. (1996). Distribution Volume Ratios without Blood Sampling from Graphical Analysis of PET Data. *Journal of Cerebral Blood Flow & Metabolism*, 16(5), 834-840. <https://doi.org/10.1097/00004647-199609000-00008>

### See Also

Other Logan plot functions: [DVR\\_all\\_ref\\_Logan](#), [dvr](#), [plot.ref\\_Logan](#)

**Examples**

```
f <- system.file("extdata", "AD06.tac", package="tacmagic")
fv <- system.file("extdata", "AD06_TAC.voistat", package="tacmagic")
AD06_tac <- load_tac(f, format="PMOD")
AD06_volume <- load_vol(fv, format="voistat")
AD06 <- tac_roi(tac=AD06_tac, volumes=AD06_volume, ROI_def=roi_ham_pib(),
               merge=FALSE, PVC=FALSE)

AD06_DVR_fr <- DVR_ref_Logan(AD06, target="frontal", ref="cerebellum",
                             k2prime=0.2, t_star=0)
```

---

fake\_DVR

*Fake DVR data for vignette and package testing*


---

**Description**

A fake dataset of 50 simulated participants in the format that the function `tm_batch()` would be expected to produce with the "Logan" model specified. The data itself was generated as follows:

```
#higher <- matrix(rnorm(40, 1.9, 0.6), ncol=4, nrow=10)
#lower <- matrix(rnorm(160, 1.3, 0.3), ncol=4, nrow=40)
#fake_data <- as.data.frame(rbind(higher, lower))
#row.names(fake_data) <- paste0("p", 1:50)
#colnames(fake_data) <- c("ROI1_DVR", "ROI2_DVR", "ROI3_DVR", "ROI4_DVR")
#save(fake_data, "fake_DVR.Rda")
```

**Usage**

```
fake_DVR
```

**Format**

A data frame with 50 rows and 4 variables representing ROIs

---

load\_tac

*Loads TAC from file for use by other functions (default is PMOD .tac format)*


---

**Description**

This is the main function for loading an individual participant's TAC data. The minimal required information within the supplied files is the start and stop times and a time unit (either seconds or minutes), as well as the activity values for 1 or more ROIs, and units for activity. The currently supported formats (with the corresponding format argument), include:

- "PMOD": PMOD .tac files
- "voistat": PMOD TAC .voistat files used in combination with PMOD .acqtimes file for start/stop times.
- "magia": magia pipeline .mat tac file
- "DFT": Turku PET Centre's DFT format

**Usage**

```
load_tac(filename, format = "PMOD", acqtimes = NULL,
         time_unit = NULL, activity_unit = NULL)
```

**Arguments**

filename	(e.g. "participant01.tac")
format	A character string, with options listed above (e.g. "PMOD")
acqtimes	Filename for a .acqtimes file (as in PMOD), required for format="voistat"
time_unit	NULL if in file (e.g. PMOD .tac), or set to "seconds" or "minutes" if not in file or to override file
activity_unit	NULL if in file (e.g. PMOD .tac), or set to "kBq/cc", "Bq/cc", "nCi/cc"

**Value**

tac object

**See Also**

Other Loading functions: [as\\_tac](#), [load\\_voistat](#), [load\\_vol](#)

**Examples**

```
f_raw_tac <- system.file("extdata", "AD06.tac", package="tacmagic")
tac <- load_tac(f_raw_tac)
```

---

load_voistat	<i>Reads PMOD .voistat files and optionally merges volume-weighted ROIs</i>
--------------	---

---

### Description

PMOD can produce .voistat files with the average model values by ROI for its voxelwise binding potential (BPnd) models, such as Logan, SRTM, etc. This function reads the .voistat file and returns a data.frame with the ROI as rows and the model value as the column. Optionally, the ROIs can be combined into larger ROIs if ROI\_def is specified, just as with TAC loading.

### Usage

```
load_voistat(filename, ROI_def = NULL, model = "VALUE")
```

### Arguments

filename	(e.g. participant_logan.voistat)
ROI_def	Optional ROI definitions to combine ROIs (e.g. roi_ham_pib())
model	A string to name the variable being extracted, e.g. "Logan_DVR"

### Value

data.frame with loaded model data in specified combined weighted ROIs

### See Also

Other Loading functions: [as.tac](#), [load\\_tac](#), [load\\_vol](#)

### Examples

```
f <- system.file("extdata", "AD06_BPnd_BPnd_Logan.voistat",
                 package="tacmagic")
vs <- load_voistat(f, ROI_def=roi_ham_pib(), model="Logan")
```

---

load_vol	<i>Loads ROI volumes from file for use by other functions</i>
----------	---

---

### Description

Loads ROI volumes from file for use by other functions

### Usage

```
load_vol(filename, format = "voistat")
```

**Arguments**

filename (e.g. participant.voistat)  
 format (default is the TAC .voistat format from PMOD, also accepts "DFT and "BPnd-Paste")

**Value**

data.frame with loaded TAC data

**See Also**

Other Loading functions: [as.tac](#), [load\\_tac](#), [load\\_voistat](#)

**Examples**

```
f_raw_vol <- system.file("extdata", "AD06_TAC.voistat", package="tacmagic")
vol <- load_vol(f_raw_vol)
```

---

plot.ref\_Logan                      *Non-invasive reference Logan plot*

---

**Description**

This plots the non-invasive Logan plot.

**Usage**

```
## S3 method for class 'ref_Logan'
plot(x, ...)
```

**Arguments**

x                      Reference Logan model data object from DVR\_ref\_Logan()  
 ...                    Additional parameters than can be passed to plotting function

**Value**

No return f <- system.file("extdata", "AD06.tac", package="tacmagic") fv <- system.file("extdata", "AD06\_TAC.voistat", package="tacmagic") AD06\_tac <- load\_tac(f, format="PMOD") AD06\_volume <- load\_vol(fv, format="voistat") AD06 <- tac\_roi(tac=AD06\_tac, volumes=AD06\_volume, ROI\_def=roi\_ham\_pib(), merge=FALSE, PVC=FALSE) AD06\_DVR\_fr <- DVR\_ref\_Logan(AD06, target="frontal", ref="cerebellum", k2prime=0.2, t\_star=0) plot(AD06\_DVR\_fr)

**See Also**

Other Logan plot functions: [DVR\\_all\\_ref\\_Logan](#), [DVR\\_ref\\_Logan](#), [dvr](#)

---

plot.tac

*Plots time activity curves from 1 or 2 participants or groups.*


---

### Description

Plots time activity curves from 1 or 2 participants or groups.

### Usage

```
## S3 method for class 'tac'
plot(x, tac2 = NULL, ROIs, ymax = 25, time = "minutes",
     title = "", colors = rainbow, ...)
```

### Arguments

x	A tac object containing time-activity curves to plot, e.g. from <code>tac_roi()</code> or <code>load_tac()</code>
tac2	An optional, second TAC, to plot for comparison
ROIs	A vector of ROIs to plot, names matching the TAC headers
ymax	The maximum value on the y-axis
time	"seconds" or "minutes" depending on desired x-axis, converts tac
title	A title for the plot
colors	If null, rainbow palette is used, otherwise another palette can be specified ( <code>heat.colors</code> , <code>terrain.colors</code> , <code>topo.colors</code> , <code>cm.colors</code> )
...	Additional arguments

### Value

Creates a plot

### See Also

Other tac functions: [save\\_tac](#), [split\\_pvc](#), [tac\\_roi](#)

### Examples

```
# f_raw_tac and f_raw_vol are the filenames of PMOD-generated files
f_raw_tac <- system.file("extdata", "AD06.tac", package="tacmagic")
f_raw_vol <- system.file("extdata", "AD06_TAC.voistat", package="tacmagic")

tac <- load_tac(f_raw_tac)
vol <- load_vol(f_raw_vol)
AD06_tac_nc <- tac_roi(tac, vol, roi_ham_full(), merge=FALSE, PVC=FALSE)
plot(AD06_tac_nc, ROIs=c("frontal", "cerebellum"), title="Example Plot")
```



---

pos_anyroi	<i>Dichotomize participants based on ROI cutoff values</i>
------------	--

---

### Description

Aizenstein et al. (2008) proposed a standardized method of calculating PIB+ cutoff values to classify participants as PIB+ or PIB-. They used the DVR from 7 ROIs associated with amyloid deposition. This function takes the ROI-based cutoff values, e.g. from `cutoff_aiz()`, and returns a table specifying which participants are positive, i.e. which have at least one ROI greater than the cutoff.

### Usage

```
pos_anyroi(modelstats, cutoff)
```

### Arguments

modelstats	SUVR or DVR data for group of participants from <code>batch_tm()</code>
cutoff	cutoffs for ROIs as from <code>cutoff_aiz()</code>

### Value

data.frame of participants and positive/negative status

### References

Aizenstein HJ, Nebes RD, Saxton JA, et al. 2008. Frequent amyloid deposition without significant cognitive impairment among the elderly. *Arch Neurol* 65: 1509-1517.

### See Also

Other Cutoff functions: [cutoff\\_aiz](#)

---

roi_ham_full	<i>Return a list of larger ROIs made up of the ROIs in the Hammer's atlas.</i>
--------------	--

---

### Description

This includes the cortical regions of `roi_ham_stand()` but also other regions. It can be modified to suit the user's needs.

### Usage

```
roi_ham_full()
```

**Value**

A list of lists, where each list is an ROI (e.g.) frontal lobe that specifies the atomic ROIs from the atlas that make it up.

**References**

Hammers, Alexander, Richard Allom, Matthias J. Koepp, Samantha L. Free, Ralph Myers, Louis Lemieux, Tejal N. Mitchell, David J. Brooks, and John S. Duncan. 2003. Three-dimensional Maximum Probability Atlas of the Human Brain, with Particular Reference to the Temporal Lobe. *Human Brain Mapping* 19 (4): 224-247. doi:10.1002/hbm.10123

**See Also**

Other ROI definitions: [roi\\_ham\\_pib](#), [roi\\_ham\\_stand](#)

**Examples**

```
roi_ham_full()
```

---

roi_ham_pib	<i>Return a list of merged ROIs made up of atomic ROIs in the Hammer's atlas.</i>
-------------	---

---

**Description**

This includes the ROIs from `roi_ham_full` and also the PIB cortical composite ROI as defined in the PMOD documentation and as widely used in PIB studies. See PMOD Neuro Tool (PNEURO) (Version 4.0) documentation.

**Usage**

```
roi_ham_pib()
```

**Value**

A list of lists, where each list is an ROI (e.g.) frontal lobe that specifies the atomic ROIs from the atlas that make it up.

**References**

Hammers, Alexander, Richard Allom, Matthias J. Koepp, Samantha L. Free, Ralph Myers, Louis Lemieux, Tejal N. Mitchell, David J. Brooks, and John S. Duncan. 2003. Three-dimensional Maximum Probability Atlas of the Human Brain, with Particular Reference to the Temporal Lobe. *Human Brain Mapping* 19 (4): 224-247. doi:10.1002/hbm.10123

**See Also**

Other ROI definitions: [roi\\_ham\\_full](#), [roi\\_ham\\_stand](#)

**Examples**

```
roi_ham_pib()
```

---

roi_ham_stand	<i>Return a list of merged ROIs made up of the atomic ROIs in the Hammer's atlas.</i>
---------------	---

---

**Description**

Return a list of merged ROIs made up of the atomic ROIs in the Hammer's atlas.

**Usage**

```
roi_ham_stand()
```

**Value**

A list of lists, where each list is an ROI (e.g.) frontal lobe that specifies the atomic ROIs from the atlas that make it up.

**References**

Hammers, Alexander, Richard Allom, Matthias J. Koepp, Samantha L. Free, Ralph Myers, Louis Lemieux, Tejal N. Mitchell, David J. Brooks, and John S. Duncan. 2003. Three-dimensional Maximum Probability Atlas of the Human Brain, with Particular Reference to the Temporal Lobe. *Human Brain Mapping* 19 (4): 224-247. doi:10.1002/hbm.10123

**See Also**

Other ROI definitions: [roi\\_ham\\_full](#), [roi\\_ham\\_pib](#)

**Examples**

```
roi_ham_stand()
```

---

save_tac	<i>Save a tac object as a .tac file</i>
----------	---

---

### Description

Saves a tac object, created by `load_tac()`, `tac_roi()` or manually, and saves it as a PMOD-formatted tac file. Using the `.tac` extension in the file name is recommended.

### Usage

```
save_tac(tac, outfile)
```

### Arguments

tac	The time-activity curve data, e.g. from <code>load_tac()</code> or <code>tac_roi()</code>
outfile	The output filename

### Value

Does not return an object, only saves a file

### See Also

Other tac functions: [plot\\_tac](#), [split\\_pvc](#), [tac\\_roi](#)

---

split_pvc	<i>Subset PMOD tacs with or without PVC</i>
-----------	---

---

### Description

When partial volume correction (PVC) is used in PMOD, the saved tac files have ROIs with and without PVC. When loaded with `load_tac()` it may be desirable to keep only either the PVC or non-PVC tacs. This returns a tac object that is a subset of the input tac object with only the PVC or non-PVC tacs. This relies on PMOD's convention of labelling tac columns with `"_C"`.

### Usage

```
split_pvc(tac, PVC = TRUE)
```

### Arguments

tac	The time-activity curve data from loading function (PMOD)
PVC	If TRUE, includes columns with <code>"_C"</code> , if FALSE, ones without <code>"_C"</code>

**Value**

Time-activity curve object

**See Also**

Other tac functions: [plot\\_tac](#), [save\\_tac](#), [tac\\_roi](#)

**Examples**

```
# f_raw_tac and f_raw_vol are the filenames of PMOD-generated files
f_raw_tac <- system.file("extdata", "AD06.tac", package="tacmagic")

tac <- load_tac(f_raw_tac)
tac_pvc <- split_pvc(tac, TRUE)
tac_nc <- split_pvc(tac, FALSE)
```

---

SUV

*Calculate average SUV over time window, or maximum SUV*

---

**Description**

Calculate the standardized uptake value (SUV) from a tac object, the participant's weight, and the tracer dose. These values may be in the tac object or manually supplied. The weight must be in kg, and the tracer units must be specified. The dose is converted to MBq, the tac is converted to kBq/cc, and the final SUV units are thus in g/cc. Aside from the tac object, the remaining parameters should be left NULL if the required data is in the tac object attributes (as can be done with `batch_load()`).

**Usage**

```
suv(tac, SUV_def, dose = NULL, dose_unit = NULL, weight_kg = NULL,
    ...)
```

**Arguments**

tac	time-activity curve object (decay-corrected)
SUV_def	vector of start times for window for SUV weighted average, or alternatively, "max" for the maximum ROI SUV value
dose	the injected tracer dose
dose_unit	unit of tracer dose (e.g. "MBq", "kBq", "mCi"...)
weight_kg	the participant's weight in kg
...	When called from <code>tm_batch</code> , unused parameters may be supplied

**Value**

table of SUV values

**See Also**

Other SUV functions: [tac\\_suv](#)

**Examples**

```
f <- system.file("extdata", "AD06.tac", package="tacmagic")
fv <- system.file("extdata", "AD06_TAC.voistat", package="tacmagic")
AD06_tac <- load_tac(f, format="PMOD")
AD06_volume <- load_vol(fv, format="voistat")
AD06 <- tac_roi(tac=AD06_tac, volumes=AD06_volume, ROI_def=roi_ham_pib(),
               merge=FALSE, PVC=FALSE)
# dose and weight are fabricated for the example
AD06_suvmax <- suv(AD06, "max", dose = 9.0, dose_unit = "mCi",
                  weight_kg = 70)
AD06_suv <- suv(AD06, c(3000, 3300, 3600), dose = 9.0, dose_unit = "mCi",
               weight_kg = 70)
```

---

 suvr

*Calculate weighted SUVRs for specified regions of interest*

---

**Description**

Calculate the standardized uptake value ratio (SUVR) for all ROIs in the provided tac data, using the specified reference region.

**Usage**

```
suvr(tac, SUVR_def, ref, ...)
```

**Arguments**

tac	The time-activity curve data from tac_roi()
SUVR_def	a vector of start times for window to be used in SUVR
ref	a string, e.g. "cerebellum", to specify reference region
...	When called from tm_batch, unused parameters may be supplied

**Value**

A data.frame of SUVR values for the specified ROIs

**See Also**

Other SUVR functions: [suvr\\_auc](#)

**Examples**

```
f <- system.file("extdata", "AD06.tac", package="tacmagic")
fv <- system.file("extdata", "AD06_TAC.voistat", package="tacmagic")
AD06_tac <- load_tac(f, format="PMOD")
AD06_volume <- load_vol(fv, format="voistat")
AD06 <- tac_roi(tac=AD06_tac, volumes=AD06_volume, ROI_def=roi_ham_pib(),
               merge=FALSE, PVC=FALSE)

AD06_SUVR <- suvr(AD06, SUVR_def=c(3000,3300,3600), ref="cerebellum")
```

---

suvr_auc	<i>Calculate SUVRs for regions of interest with AUC from mid-frame times</i>
----------	--

---

**Description**

Calculate the standardized uptake value ratio (SUVR) for all ROIs in the provided tac data, using the specified reference region. This is an alternate to suvr() which should provide very similar values.

**Usage**

```
suvr_auc(tac, SUVR_def, ref, ...)
```

**Arguments**

tac	The time-activity curve data from tac_roi()
SUVR_def	a vector of start times for window to be used in SUVR
ref	is a string, e.g. "cerebellum", to specify reference region
...	When called from tm_batch, unused parameters may be supplied

**Value**

A data.frame of SUVR values for the specified ROIs #' f <- system.file("extdata", "AD06.tac", package="tacmagic") fv <- system.file("extdata", "AD06\_TAC.voistat", package="tacmagic") AD06\_tac <- load\_tac(f, format="PMOD") AD06\_volume <- load\_vol(fv, format="voistat") AD06 <- tac\_roi(tac=AD06\_tac, volumes=AD06\_volume, ROI\_def=roi\_ham\_pib(), merge=FALSE, PVC=FALSE)

```
AD06_SUVR <- suvr_auc(AD06, SUVR_def=c(3000,3300,3600), ref="cerebellum")
```

**See Also**

Other SUVR functions: [suvr](#)

---

tacmagic	<i>tacmagic: PET Analysis in R</i>
----------	------------------------------------

---

### Description

The main features of tacmagic are to load PET time activity curve (tac) data from multiple formats, merge ROIs weighted for volume, calculate binding potential models including SUVR and DVR, basic plotting, and calculation of cut-off values. Please see the walkthrough vignette for a detailed overview.

---

tac_roi	<i>Calculate weighted time-activity curves for specified regions of interest</i>
---------	--

---

### Description

Calculate weighted time-activity curves for specified regions of interest

### Usage

```
tac_roi(tac, volumes, ROI_def, merge, PVC)
```

### Arguments

tac	The time-activity curve data from loading function
volumes	The ROI volume data from loading function
ROI_def	The definition of ROIs by combining smaller ROIs from TAC file
merge	If TRUE, includes the original ROIs in the output data
PVC	If TRUE, appends "_C" to ROI name header (as in PMOD TAC files)

### Value

Time-activity curves for the specified ROIs

### See Also

Other tac functions: [plot\\_tac](#), [save\\_tac](#), [split\\_pvc](#)

### Examples

```
# f_raw_tac and f_raw_vol are the filenames of PMOD-generated files
f_raw_tac <- system.file("extdata", "AD06.tac", package="tacmagic")
f_raw_vol <- system.file("extdata", "AD06_TAC.voistat", package="tacmagic")

tac <- load_tac(f_raw_tac)
vol <- load_vol(f_raw_vol)
AD06_tac_nc <- tac_roi(tac, vol, roi_ham_full(), merge=FALSE, PVC=FALSE)
```



---

 tac\_suv

*Calculate SUV from TAC*


---

### Description

Calculate the standardized uptake value (SUV) time-activity curve from a tac object, the participant's weight, and the tracer dose. The weight must be in kg, and the tracer dose must be specified. The dose is converted to MBq, the tac is converted to kBq/cc, and the final SUV units are thus in g/cc. Aside from the tac object, the remaining parameters should be left NULL if the required data is in the tac object attributes (as can be done with `batch_load()`).

### Usage

```
tac_suv(tac, dose = NULL, dose_unit = NULL, weight_kg = NULL)
```

### Arguments

tac	time-activity curve object (decay-corrected)
dose	the injected tracer dose
dose_unit	unit of tracer dose (e.g. "MBq", "kBq", "mCi" ...)
weight_kg	the participant's weight in kg

### Value

tac object with SUV values

### See Also

Other SUV functions: [suv](#)

### Examples

```
f <- system.file("extdata", "AD06.tac", package="tacmagic")
fv <- system.file("extdata", "AD06_TAC.voistat", package="tacmagic")
AD06_tac <- load_tac(f, format="PMOD")
AD06_volume <- load_vol(fv, format="voistat")
AD06 <- tac_roi(tac=AD06_tac, volumes=AD06_volume, ROI_def=roi_ham_pib(),
               merge=FALSE, PVC=FALSE)
# dose and weight are fabricated for the example
AD06_suv <- tac_suv(AD06, dose = 9.0, dose_unit = "mCi", weight_kg = 70)
```

# Index

## \*Topic **datasets**

fake\_DVR, [12](#)

as.tac, [2](#), [13–15](#)

batch\_load, [3](#), [5](#), [6](#)

batch\_tm, [4](#), [5](#), [6](#)

batch\_voistat, [4](#), [5](#), [6](#)

change\_units, [7](#)

cutoff\_aiz, [7](#), [17](#)

dvr, [8](#), [10](#), [11](#), [15](#)

DVR\_all\_ref\_Logan, [9](#), [10](#), [11](#), [15](#)

DVR\_ref\_Logan, [9](#), [10](#), [11](#), [15](#)

fake\_DVR, [12](#)

load\_tac, [3](#), [12](#), [14](#), [15](#)

load\_voistat, [3](#), [13](#), [14](#), [15](#)

load\_vol, [3](#), [13](#), [14](#), [14](#)

plot.ref\_Logan, [9–11](#), [15](#)

plot.tac, [16](#), [20](#), [21](#), [24](#)

pos\_anyroi, [8](#), [17](#)

roi\_ham\_full, [17](#), [18](#), [19](#)

roi\_ham\_pib, [18](#), [18](#), [19](#)

roi\_ham\_stand, [18](#), [19](#)

save\_tac, [16](#), [20](#), [21](#), [24](#)

split\_pvc, [16](#), [20](#), [20](#), [24](#)

suv, [21](#), [25](#)

suvr, [22](#), [23](#)

suvr\_auc, [22](#), [23](#)

tac\_roi, [16](#), [20](#), [21](#), [24](#)

tac\_suv, [22](#), [25](#)

tacmagic, [24](#)

tacmagic-package (tacmagic), [24](#)