RAR

March 21, 2019

Title Parametric, Nonparametric, and Localized RAR Analysis

Version 1.0.0

Description This package offers three methods of analyzing rest-activity rhythms (RARs): sigmoidally transformed extended cosine model, spectrum analysis, and localized measure of RAR. Sigmoidally transformed extended cosine model is adapted from Marler et al. 2006 <doi: 10.1002/sim.2466>. Nonparametric analyses are adapted from van Someren et al. 1996. Localized measures of RAR include the following measures at specified time bins: Mean activity, Standard Deviation of activity, and Relative activity (Graves, 2018).

Depends R (>= 3.4.0)

License GPL (>= 2)

Encoding UTF-8

LazyData true

RdMacros Rdpack

Imports nlstools,
tidyr,
ggplot2,
dplyr,
dotwhisker,
broom,
lubridate,
grDevices,
methods,
nparACT,
Rdpack,
magrittr,
kableExtra,
knitr,
xtable,
xts,
astsa,
mgcv,
gss

RoxygenNote 6.1.1

R topics documented:

`df_predicted` *Example dataframe generated from for RAR_ExCosine()*

Description

df_predicted.rda is an example dataframe that is generated from RAR_ExCosine(). In general, dataframes generated from RAR_ExCosine() can be accessed using RAR_ExCosine()$.

Usage

```
data(df_predicted)
```

Format

A data frame with 23820 rows and 12 variables (note, there are 60 missing time steps in rar_data3, and so nrows(df_predicted) is 60 fewer than rar_data3). Activity measurements are acquired for every minute (60-second epochs). Number of observation days = 18

Details

Variables include:

- time: factor listing time of observation.
- act: observed activity (raw counts).
- log.act: log(act + 1) generated by RAR_ExCosine()
- day: day column generated by RAR_ExCosine()
- time.posixct: POSIXct time formate generated by RAR_ExCosine()
- tm: time in minutes from midnight
- td: time in minutes from the first midnight
- n: observation number
- predicted: predicated values estimated from Extended Cosine model, on log scale
- resid: calculated residauls estimated from Extended Cosine model, on log scale
- e_predicted: predicted values on natural scale
- e_resid: residuals on natural scale

RAR_CorrByTime *RAR Correlation by Localized Measures*

Description

This function correlates localized RAR measures for an entire sample (mean, standard deviation, and relative activity) at each time bin against an outcome of interest. This function will take a dataframe that contains RAR_Localized() measures with an outcome of interest already merged in. Or, it will take two separate dataframes, one with the RAR_Localized() measures and one with the outcome. If using two dataframes, you must specifiy the ID column, which will be used to merge the two together within the funciton.

Usage

```
RAR_CorrByTime(df_measures, df_outcome = NULL, id_column = NULL,
  second_var, corr_type = c("pearson", "kendall", "spearman"))
```

Arguments

df_measures	dataframe containing columns from RAR_Localized() for each subject. Mean, Standard Deviation, and Relative Activity columns must be named mean.act, sd.act, and rel.act, respectively. This dataframe must also have an id column. If dataframe is already merged with outcome of interest, leave df_outcome
df_outcome	dataframe containing outcome of interest. This dataframe must also have an id column, which is named the same as df_measures id column.
id_column	specifies the column name in df_measures and df_outcome that corresponds to the id
second_var	specifies the column name in the dataframe that contains the outcome of interest.
corr_type	specifies the type of correlation, e.g. ("pearson", "kendall", "spearman"). Remaining options are defaults of cor.test with exact p-value = FALSE

Author(s)

Jessica Graves

rar_data *Example Data for RAR_ExCosine()*

Description

rar_data.rda is an example dataframe illustrating the basic format RAR_ExCosine() will accept. Note that a day column is not needed.

Usage

```
data(rar_data)
```

Format

A data frame with 22905 rows and 2 variables. Activity measurements are acquired for every minute (60-second epochs). Number of observation days = 19.

Details

Variables include:

- time: factor listing time of observation.
- act: observed activity (raw counts).

---

rar_data2 *Example Data 2 for RAR_ExCosine()*

Description

rar_data2.rda is another example dataframe illustrating the basic format RAR_ExCosine() will accept. Note that a day column is not needed.

Usage

```
data(rar_data2)
```

Format

A data frame with 27393 rows and 2 variables. Activity measurements are acquired for every minute (60-second epochs). Number of observation days = 20.

Details

Variables include:

- time: factor listing time of observation.
- act: observed activity (raw counts).

---

rar_data3 *Example Data 3 for RAR_ExCosine()*

Description

rar_data3.rda is another example dataframe illustrating the basic format RAR_ExCosine() will accept. Note that a day column is not needed.

Usage

```
data(rar_data3)
```

Format

A data frame with 23880 rows and 2 variables. Activity measurements are acquired for every minute (60-second epochs). Number of observation days = 18

Details

Variables include:

- time: factor listing time of observation.
- act: observed activity (raw counts).

`rar_data_mutli_wake` *Example Data for RAR_ExCosine_Multi()*

Description

rar_data_multi.rda is an example dataframe illustrating the basic format RAR_ExCosine_Multi() will accept. Participant 1 is taken from rar_data3; Participant 2 is taken from rar_data2. Wake up hours are extracted from anti-logistic extended cosine model tLeft parameter estimates for each participant.

rar_data_multi_wake.rda is an example dataframe illustrating the basic format RAR_Localized_Multi() will accept. Participant 1 is taken from rar_data3; Participant 2 is taken from rar_data2. Wake up hours are extracted from anti-logistic extended cosine model tLeft parameter estimates for each participant.

Usage

```
data(rar_data_multi)

data(rar_data_multi_wake)
```

Format

A data frame with 51273 rows and 3 variables. Activity measurements are acquired for every minute (60-second epochs) for two participants (IDs = 1 and 2)

Details

Variables include:

- id: ID
- time: factor listing time of observation.
- act: observed activity (raw counts).

Variables include:

- id: ID
- time: factor listing time of observation.
- act: observed activity (raw counts).
- wakehr: average wake up time for each participant, calculated as tLeft from RAR_ExCosine() using anti-logistic transformation

RAR_ExCosine *RAR Extended Cosine Model*

Description

This function fits sigmoidally transformed extended cosine model to activity data, as seen in Marler et al. (2006). Activity data provided to the function should include data for only one subject. Function takes dataframe with columns that include time and raw activity counts.

Usage

```
RAR_ExCosine(df, act_column, time_column, transform = c("hill",
  "antilogit", "arctan"), plot = c(TRUE, FALSE))
```

Arguments

df	dataframe containing actigraphy data and time. Time must be in HH:MM:SS format and stored as a character, factor, or POSIX object.
act_column	specifies the name of the column within df that contains the activity count data. RAR_ExCosine will do a log(activity + 1) transformation.
time_column	specifies the name of the column that contains time of observation
transform	specifies which transformation to apply. Options include Hill Function ("hill"), Anti-Logistic ("antilogit"), or Arctangent ("arctan")
plot	logical specifying if plots should be outputted. If missing, defaults to FALSE.

Details

Outputs from this function include: coefficient estimates for baseline cosine model and user-specificed extended cosline model; starting values for baseline cosine model; dataframe with predicted values based on fitted user-specified extended cosine; and parameter estimates of interest.

Author(s)

Jessica Graves

References

1. Marler M.R., Gehrman P., Martin J.L., Ancoli-Israel S. (2006) The sigmoidally transformed cosine curve: a mathematical model for circadian rhythms with symmetric non-sinusoidal shapes. Stat Med. Nov 30;25(22):3893-904.

See Also

nls dplyr

Examples

```
data(rar_data)
rar_cosine = RAR_ExCosine(rar_data, act, time, "antilogit", plot=TRUE)
rar_cosine$final_parameters
rar_cosine$plot_log.act
```

RAR_ExCosine_Multi *RAR Extended Cosine Model: For Files with Mulitple Participants*

Description

This function peforms the same functionality as RAR_ExCosine(), excpet that it accepts a dataframe with a third column specifying ID to allow for iteration across subjects.

Usage

```
RAR_ExCosine_Multi(df, act_column, time_column, transform = c("hill",
  "antilogit", "arctan"), id_column, plot = c(TRUE, FALSE))
```

Arguments

df	dataframe containing actigraphy data, time, and ID column. Time must be in HH:MM:SS format and stored as a character, factor, or POSIX object.
act_column	specifies the name of the column within df that contains the activity count data. RAR_ExCosine will do a log(activity + 1) transformation.
time_column	specifies the name of the column that contains time of observation
transform	specifies which transformation to apply. Options include Hill Function ("hill"), Anti-Logistic ("antilogit"), or Arctangent ("arctan")
id_column	specifies the name of the column that contains ID information.
plot	logical specifying if plots should be outputted. If missing, defaults to FALSE.

Details

Outputs from this function include a tibble object containing RAR_ExCosine_Out for each participant.

Author(s)

Jessica Graves

References

1. Marler M.R., Gehrman P., Martin J.L., Ancoli-Israel S. (2006) The sigmoidally transformed cosine curve: a mathematical model for circadian rhythms with symmetric non-sinusoidal shapes. Stat Med. Nov 30;25(22):3893-904.

See Also

nls dplyr

Examples

```
data(rar_data_multi)
rar_multi = RAR_ExCosine_Multi(rar_data_multi, act, time, "antilogit", id, plot=TRUE)
rar_multi$RAR_Ouput # tibble object of participants and objects

rar_multi$RAR_Output$RAR_ExCosine_Out[[1]]$final_parameters # individ participant estimates
rar_multi$RAR_Output$RAR_ExCosine_Out[[1]]$plot_log.act # individ participant plots

rar_multi$final_parameters # df of parameter estimates for all participants
```

RAR_FFTbands *RAR Fast Fourier Transform of Selected Frequency Bands*

Description

This function performs Fast Fourier Transform (FFT) on residuals based on extended cosine model (RAR_ExCosine()) and applies user-specified frequency-band filters. The methodology can be found in detail in R.T Krafty et al, Measuring Variability in Rest-Activity Rhythms from Actigraphy with Application to Characterizing Symptoms of Depression (2019)

Usage

```
RAR_FFTbands(df, resid_column, time_column, predict_column,
  plot = c(TRUE, FALSE), freq.bands, sampling.rate)
```

Arguments

df	dataframe including residuals calculating from extended cosine model based on actigraphy data. Residuals must be in numeric class. Time must be in HH:MM:SS format and stored as a character, factor, or POSIX object.
resid_column	specfies the name of the column in the df that contains residuals. This must be in numeric class
time_column	specifies the name of the column that contains time of observation for residuals
predict_column	specifies the name of the column that contains the predicted values from the extended cosine model
plot	a logical value specifying if plots should be outputted. Plots illustrate effects of filters.
freq.bands	a kx2 matrix or dataframe specifcing frequency band intervals of interested for calculating area under spectrum, where the 1st column is the lower bound, and the 2nd column is the upper bound for the frequency band. Frequencies supplied must be in cycles/hour.
sampling.rate	a numeric value representing the sampling rate of data in Hz (cycles/second). e.g. data collected once every 30 seconds specified as sampling.rate=1/30.

Details

This function will apply user-specified frequency bands to residuals extracted from the extened cosine model and will illustrate the effect of those filters.

Value

a list containing original dataframe along with filter results, plots of filter effects (if plot=T), and lists of FFT per filter.

Author(s)

Jessica Graves

References

1. R.T Krafty et al, Measuring Variability in Rest-Activity Rhythms from Actigraphy with Application to Characterizing Symptoms of Depression (2019)

See Also

`fft`

Examples

```
data(df_predicted)
f1 <- c(0, 2/24) # Filter Values
f2 <- c(2/24, 25)
f3 <- c(25, 60)
fs = as.data.frame(rbind(f1, f2, f3))
filter = RAR_FFTbands(df_predicted, resid, time, predicted, T, fs, 1/60)
filter$plots[[1]]
```

RAR_Localized *RAR Localized Measures*

Description

This function calculates localized measures of RAR timing, as described in Graves (2018). Measures include: Mean Activity, Standard Deviation of Activity, and Relative Activity at user-specified time-bins across days.

Usage

```
RAR_Localized(df, act_column, time_column, hour_bin,
  person_time = c(TRUE, FALSE), tLeft = NULL, transform = c(TRUE,
  FALSE), plot = c(TRUE, FALSE))
```

Arguments

`df`	dataframe containing actigraphy data and time. Time must be in HH:MM:SS format and stored as a character, factor, or POSIX object.
`act_column`	specifies the name of the column within df that contains the activity count data. RAR_ExCosine will do a log(activity + 1) transformation.
`time_column`	specifies the name of the column that contains time of observation
`hour_bin`	specifices the number of hours within each time bin (e.g. 4 hour timebins = 4)

person_time	logical specifying if person-specific time adjustements should be made. If missing, default is FALSE.
tLeft	numeric value indicating subject's up-mesor (or wake up hour). Can be estimated using RAR_ExCosine(). Or, if merged into dataframe, is column name.
transform	logical specifying if log(activity + 1) transformation is desired
plot	logical specifying if plots should be outputted. If missing, defaults to FALSE.

Details

Outputs from this function include: dataframes of calculated localized measures based on user-specified hour bins, day x time summaries of activity, plots of Mean, SD, and Relative Activity Measures.

Author(s)

Jessica Graves

Examples

```
data(rar_data)
# defaults to clock-time and log(act + 1)
rar_local = RAR_Localized(rar_data, act, time, 4, plot=TRUE)
rar_local$localized_measures
rar_local$plots
rar_local$plots[[1]]
```

RAR_Localized_Multi *RAR Localized Measures: For Files with Mulitple Participants*

Description

This function calculates localized measures of RAR timing, as described in Graves (2018). Measures include: Mean Activity, Standard Deviation of Activity, and Relative Activity at user-specified time-bins across days. This function takes a single dataframe including activity, time, ID, and wake-up hour for each participant.

Usage

```
RAR_Localized_Multi(df, act_column, time_column, id_column, hour_bin,
  person_time = c(TRUE, FALSE), tLeft = NULL, transform = c(TRUE,
  FALSE), plot = c(TRUE, FALSE))
```

Arguments

df	dataframe containing actigraphy data, time, an ID column, and a column for average wake up hours. Time must be in HH:MM:SS format and stored as a character, factor, or POSIX object.
act_column	specifies the name of the column within df that contains the activity count data. RAR_ExCosine will do a log(activity + 1) transformation.
time_column	specifies the name of the column that contains time of observation

id_column	specifies the name of the column that contains participant IDs
hour_bin	specifices the number of hours within each time bin (e.g. 4 hour timebins = 4)
person_time	logical specifying if person-specific time adjustements should be made. If missing, default is FALSE.
tLeft	is a column name
transform	logical specifying if log(activity + 1) transformation is desired
plot	logical specifying if plots should be outputted. If missing, defaults to FALSE.

Details

Outputs from this function include: dataframes of calculated localized measures based on user-specified hour bins, day x time summaries of activity, plots of Mean, SD, and Relative Activity Measures.

Author(s)

Jessica Graves

Examples

```
data(rar_data_multi_wake)
Local_Multi = RAR_Localized_Multi(rar_data_multi_wake, act, time, id, 4, TRUE, wakehr, TRUE, TRUE)
Local_Multi$localized_measures
Local_Multi$plots
```

RAR_RegByTime *RAR Regression by Localized Measures*

Description

This function performs linear regression on localized RAR measures for an entire sample (mean, standard deviation, and relative activity) at each time bin against an outcome of interest. This function will take a dataframe that contains RAR_Localized() measures with an outcome of interest already merged in. Or, it will take two separate dataframes, one with the RAR_Localized() measures and one with the outcome. If using two dataframes, you must specifiy the ID column, which will be used to merge the two together within the funciton.

Usage

```
RAR_RegByTime(df_measures, df_outcome = NULL, id_column = NULL,
  y_variable, formula, model_name = NULL, time_type = NULL,
  plots = c(TRUE, FALSE))
```

Arguments

df_measures	dataframe containing columns from RAR_Localized() for each subject. Mean, Standard Deviation, and Relative Activity columns must be named mean.act, sd.act, and rel.act, respectively. This dataframe must also have an id column. If dataframe is already merged with outcome of interest, leave df_outcome
df_outcome	dataframe containing outcome of interest. This dataframe must also have an id column, which is named the same as df_measures id column.

id_column	specifies the column name in df_measures and df_outcome that corresponds to the id
y_variable	a string describing outcome of interest (e.g, "Depression score")
formula	formula to run lm model, e.g. score ~ mean.act.
model_name	a string describing the model predictors, e.g. "Mean Activity". Optional.
time_type	a string describing the time type, e.g. "Person Time" or "Clock Time". Optional.
plots	a logical specifying if the user would like plot to be outputted. Default is FALSE

Author(s)

Jessica Graves

RAR_Spectrum *RAR Residuals spectrum analysis and smoothing splines*

Description

This function performs Residual Circadian Spectrum (RCS) analysis from rest-activity rhythm actigraphy data. The methodology can be found in detail in R.T Krafty et al, Measuring Variability in Rest-Activity Rhythms from Actigraphy with Application to Characterizing Symptoms of Depression (2019)

Usage

```
RAR_Spectrum(df, resid_column, time_column, plot = c(TRUE, FALSE),
  method = c("smooth spline", "penalized spline"), log = c(TRUE,
  FALSE), freq.bands = NULL, ...)
```

Arguments

df	dataframe including residuals calculating from extended cosine model based on actigraphy data. Residuals must be in numeric class. Time must be in HH:MM:SS format and stored as a character, factor, or POSIX object.
resid_column	specfies the name of the column in the df that contains residuals. This must be in numeric class
time_column	specifies the name of the column that contains time of observation for residuals
plot	a logical value specifying if plots should be outputted
method	specifies the method using for the smoothing spline. method="smooth spline" indicates using a cubic smoothing spline to the supplied data. method="penalized spline" indicates using a penalized likelihood regression smoothing spline method
log	a logical value specifying if log-transfromation should be used for the power spectrum
freq.bands	a kx2 matrix or dataframe specifcing frequency band intervals of interested for calculating area under spectrum, where the 1st column is the lower bound, and the 2nd column is the upper bound for the frequency band. Frequencies supplied must be in cycles/hour.
...	additional arguments passed from astsa::mvspec, stats::smooth.spline,gss::gssanova

Details

This function will perform power spectrum analysis for residuals extracting from the extended cosine model. Plots will be draw with power spectrum lines and the smoothing curve. Log argument indicates whether log-transformation will be performed for spectrum. The function includes two methods for smoothing spline: cubic smoothing spline and penalized likelihood regression smoothing spline.

Value

a list containg residuals and time, power spectrum and corresponding frequency, plots list and corresponding methods if plot argument is set to be true

Author(s)

Haoyi Fu

References

1. R.T Krafty et al, Measuring Variability in Rest-Activity Rhythms from Actigraphy with Application to Characterizing Symptoms of Depression (2019)

2. Chong Gu Smoothing Spline ANOVA Models

3. R.H Shumway, D.S Stoffer, Time Series Analysis and Its Applications

See Also

`mvspec` `gssanova` `smooth.spline`

Examples

```
data(df_predicted)
spec = RAR_Spectrum(df_predicted, resid, time, T, "smooth spline", T) # Using smooth spline and no frequency ban
spec$plot_log_ss
f1 <- c(0, 2/24) # Filter Values
f2 <- c(2/24, 25)
f3 <- c(25, 60)
fs = as.data.frame(rbind(f1, f2, f3))
spec2 = RAR_Spectrum(df_predicted, resid, time, T, "smooth spline", T, fs)
spec2$freq.bands # to get scores
```

Index