Package 'CATkit'

July 17, 2018

Type Package

Title Chronomics Analysis Toolkit (CAT): Periodicity Analysis

Version 3.3.3

Date 2018-7-16

Author Cathy Lee Gierke, Ruth Helget, Germaine Cornelissen-Guillaume

Maintainer Cathy Lee Gierke <leegi001@umn.edu>

Imports MASS, png, rtf, season, magic, stats, graphics, utils, grDevices, signal, assertr, CombMSC, Hmisc

LazyData true

Description

Calculates auto- and cross-correlation functions and plots an actogram and a smoothing function from a time series to identify and visualize periodic components. Tests presence of anticipated rhythm and estimates rhythm parameters; fits model consisting of multiple rhythmic components to data; performs least squares spectral analysis and other cosinor-based analyses, including population-mean cosinor (PMC) and population-mean cosinor parameter tests (PMCtest).

URL http://z.umn.edu/CATkit, http://www.tandfonline.com/doi/full/10.1080/09291016.2015.1094965

License GPL-2

NeedsCompilation no

Repository CRAN

Date/Publication 2018-07-17 04:10:03 UTC

R topics documented:

CATkit-package *Chronomics Analysis Toolkit (CAT): Periodicity Analysis*

Description

http://564394709114639785.weebly.com This package contains functions that can be used to identify periodicities in time series. The two calls are: CATCosinor and CATCall. CATCosinor runs various forms of a cosinor technique: a regression using cosine curves to test for anticipated periodic components and to estimate rhythm parameters: MESOR (rhythm-adjusted mean value), amplitude, and acrophase (phase of maximum). CatCall can run the following time series analysis functions: Smooth, Actogram, AutoCorr, CrossCorr.

Details

Detailed installation instructions and sample runs (vignettes) can be found at http://564394709114639785.weebly.com/installi
cat.html

Author(s)

Cathy Lee Gierke, Ruth Helget, Germaine Cornelissen-Guillaume

Maintainer: < leegi001@umn.edu>

CatCall *Visual Analysis of Period and Phase*

Description

Performs one or more analyses to gain insight into periods and/or phase in a dataset: Actogram, Smoothing, Autocorrelation or Crosscorrelation.

Usage

```
CatCall(TimeCol=1, timeFormat="%Y%m%d%H%M",lum=4, valCols=c(3,4), sumCols=c(5,6),
  Avg=FALSE, export=FALSE, sizePts=2, binPts=5, Interval = 0, Increment=0, k=6,
   yLab="Activity Level (au)", modulo=1440,Rverbose=0, RmaxGap=400, Skip=0,
  header=FALSE,Smoothing=FALSE, Actogram=FALSE,AutoCorr=FALSE,CrossCorr=FALSE,
  Console=FALSE,Graphics="pdf", Darkness=1,LagPcnt=.33,tz="GMT",fileName,file2=list
     (Name=NULL,TimeCol=1, timeFormat="%Y%m%d%H%M", lum=4, valCols=c(3,4),
     sumCols=c(5,6),sizePts=2, binPts=5,Darkness=0))
```

Arguments

TimeCol	List column(s) containing the date and time. Specify one column (a scalar) if date time is all in one column. Specify 2 time columns as a vector, c(1,2), if date is in one and time is in another. The format for time will be expected in timeFormat parameter.
timeFormat	Using the R time-formatting codes, specify how your dates are formatted. Default for a 1 column time is "%Y%m%d%H%M". A two column time will be concatenated without spaces, and your specified format applied: DateTime. See strptime base
lum	The column number containing luminance values, or NA. Luminance values are used to determine where the light level drops sharply, and this point is used as the starting point for analysis. Data points prior to dark onset are not used. (This can be reversed to use only data after light onset by setting Darkness=0.)
valCols	Specify columns that should be averaged when binned. Specify valCols=c() if none.
sumCols	Specify columns that should be summed when binned. Specify sumCols=c() if none.
Avg	A Boolean to indicate if you would like to see the output of an average of all data columns. If you tell CAT to analyze columns 4:8, and specify average, in addition to analyzing each column from 4 to 8, the columns will also be averaged, and that average will be analyzed.
export	Boolean. Default is False. If True, a data file is saved after interpolation and binning (per parameters). When True, each function (except Actogram) exports results to (separate) comma-delimited text files.
sizePts	sizePts is the number of minutes between samples.
binPts	binPts is the number of samples to aggregate into one bin. Binning is very flexible since it can be so important. sizePts * binPts = number of minutes in each bin. Only full bins are used for analysis, so there could be a few data points at the end of the data (after binEnd) that are not used.
Interval	Interval & Increment are two parameters that are used together to specify a progressive analysis. The Interval is the length of subsections of data to analyze, and the increment is how far to move ahead in the data to begin the next Interval. (Interval will need to be large enough not to trigger the error messages in the functions, which require more than 3 cycles for each analysis.) The entire data set will be analyzed (from LumStart to binEnd). A progressive analysis can be performed by the Auto-Correlation, Cross-Correlation and Periodogram analyses. The Actogram and Smoothing functions are performed on the full dataset length, for each column, as normal.
Increment	Interval & Increment are two parameters that are used together to specify a progressive analysis. The increment is how far to move ahead in the data to begin the next Interval. See Interval for complete information.
k	Only the Smoothing function uses this parameter. It is a count of the number of data points on each side of a point to include in the moving average. The moving average is calculated using 2k+1 data points.

yLab	Label for the Y axis on Smoothing and Actogram functions. Default is "Activity Level (au)"
modulo	Only the Actogram function uses this parameter. It specifies the width in minutes to be used for displaying the Actogram. Default is 1440 min, or 1 day.
Rverbose	Can take on values of -1, 0,1 or 2. 0 turns off debug information. 1 or 2 add increasing amounts of debug information. -1 minimizes information displayed on graphs.
RmaxGap	specifies the maximum allowable number of missing data points in any one block. An error will be returned if gaps larger than this are found in a data file.
header	TRUE/FALSE to indicate if the file has a header. Headers are used to name variables.
Skip	is a parameter to the R read.table function indicating how many rows to skip before reading data. (A header is read after lines skipped if header=TRUE.)
Smoothing	Default is Smoothing=FALSE; Smoothing=TRUE will cause this function to run, thus you select only the functions you need for any purpose.
Actogram	Default is Actogram=FALSE; Actogram=TRUE will cause this function to run, thus you select only the functions you need for any purpose.
AutoCorr	Default is AutoCorr=FALSE; AutoCorr=TRUE will cause this function to run, thus you select only the functions you need for any purpose.
CrossCorr	Default is CrossCorr=FALSE; CrossCorr=TRUE will cause this function to run, thus you select only the functions you need for any purpose.
Console	Default is FALSE. When Console=TRUE output will be redirected to the RStudio Console, instead of an output file.
Graphics	Results of CatCall are sent to a file when Console=FALSE. Default file output type is "pdf". Possible values: "jpg, pdf, tiff, png".
Darkness	This refers to the illumination column in the first file. CAT analysis and graphing begins at darkness onset, as indicated by the luminance column. Normally, darkness is indicated by a very small number (<10) and light is a large number (>=10). If this is needed to be reversed, changing the Darkness 1 or Darkness 2 defaults will correct the interpretation of the lumninance column for file1 or file2, respectively. Darkness=0 means that darkness is a small number (<10); Darkness=1 indicates light is a very small number, and darkness is a large number (>=10)..
LagPcnt	Specifies maximal lag used to calculate the Autocorrelation and Crosscorrelation functions, expressed as a percentage of the number of data.
tz	R timezone code. Default GMT should be used in most cases.
fileName	This is a required field, used by all functions. If this is blank, CATCosinor will solicit a file from the user by a dialogue box.
file2	Optional. Only needed when running the Crosscorrelation function. A list of parameters for the second file, mirroring those for the first file. TimeCol: Same as TimeCol above. timeFormat: Same as timeFormat above.

lum: Same as lum above.
valCols: Same as valCols above.
sumCols: Same as sumCols above.
sizePts: Same as sizePts above.
binPts: Same as binPts above.
Darkness: Same as Darkness for the first file.

Input Data:

Input data is assumed to be equidistant. All columns are expected to be numeric. Data File format: Tab- or comma-delimited (.txt) file with the following columns: time, luminance, data col1 [, data col2] . . .[, data coln]

Cross-Correlation requires 2 data files, where other functions require only one data file. A single data file with many data columns can be specified for analysis, in which case the Cross-Correlation function is skipped; or 2 data sets can be analyzed. In all cases, interpolation is done to fill in missing data points; and then analysis is done on each specified data column in a file, as well as on the average of all columns individually analyzed (if selected).

Output Data:

Sample graphics output file: The output file is a graphics file. See Output section on the web site for a sample of a full output file. All output filenames contain the input data filename to clearly identify the data file under analysis, and a timestamp to show the time of analysis. Each graph lists the column name being analyzed (or averaged), and the starting and ending times of analysis, as they vary slightly from the full data set (Lum to binEnd). Graphic output files can be in JPG, PDF, TIFF or PNG. TIFF and PNG are higher resolution than jpeg and PDF. Plots in PNG and JPEG formats can easily be converted to many other bitmap formats, and both can be displayed in modern web browsers. The PNG format is lossless and is best for line diagrams and blocks of color. The JPEG format is lossy, but may be useful for image plots, for example. It is most often used in html web pages. TIFF is a meta-format: the default format written by TIFF is lossless and stores RGB (and alpha where appropriate) values uncompressed. Such files are widely accepted, but often much larger than other file types. Binned data: The input data is interpolated and binned. This transformed data can be exported using the export parameter. If export=True then each function (except Actogram) exports a file with the results of the function.

Author(s)

Cathy Lee Gierke, John A Lindgren, Ruth Ann Helget, Germaine Cornelissen-Guillaume

Maintainer: < Cathy Lee Gierke <leegi001@umn.edu>>

References

http://564394709114639785.weebly.com/running-cat.html

Examples

```
#--------------------   Vignette3      visualization
```

```
   #
   #  Normally you would use these lines to read a file for use in CATkit
   #   (use filePath format for your OS)
 # filePath<-"~/file/path/Installing CAT/Vignette3"         # use for mac
   # filePath<-"c:\file\path\Installing CAT\Vignette3"     # use for PC
   # fileName<-file.path(filePath,'activity-stress-c57-2-part.txt')
   # file2Name<-file.path(filePath,'good-6d-2m-part.txt')
   #
 # these lines are used instead of the above since it is part of a package.
   file.copy(system.file("extdata", "activity-stress-c57-2-part.txt",
     package = "CATkit"), tempdir(), overwrite = TRUE, recursive = FALSE,
     copy.mode = TRUE, copy.date = FALSE)
file.copy(system.file("extdata", "good-6d-2m-part.txt", package = "CATkit"),
     tempdir(), overwrite = TRUE, recursive = FALSE, copy.mode = TRUE,
     copy.date = FALSE)
   filePath<-tempdir()
   fileName<-file.path(filePath,'activity-stress-c57-2-part.txt')
   file2Name<-file.path(filePath,'good-6d-2m-part.txt')

   #fileName<-system.file("extdata", "activity-stress-c57-2-part.txt",
     #package = "CATkit")
#file2Name<-system.file("extdata", "good-6d-2m-part.txt", package = "CATkit")
   #
   CatCall(TimeCol=c(1,2), timeFormat="%d/%m/%y %H:%M:%S",lum=NA,
    valCols=5, sumCols=c(), Avg=FALSE, export=TRUE, sizePts=2, binPts=30,
 Interval = 0, Increment=0, k=5, yLab=NA, modulo=1440, Rverbose=0, RmaxGap=3000,
     Skip=0, header=TRUE, Smoothing=TRUE, Actogram=TRUE,AutoCorr=TRUE,
     CrossCorr=TRUE,Console=FALSE,Graphics="pdf", Darkness=0, LagPcnt=1,
     fileName=fileName, file2=list(Name=file2Name,TimeCol=c(1,2),
       timeFormat="%d/%m/%y %H:%M:%S", lum=NA, valCols=5, sumCols=c(),
       sizePts=2, binPts=30,Darkness=0))
```

CATCosinor *Cosinor Analysis*

Description

Performs one of various cosinor-based analyses: single cosinor, least squares spectrum, multiple-component cosinor, progressive single cosinor, chronobiologic serial section (single- or multiple-component model), gliding spectrum.

Usage

```
CATCosinor(TimeCol=1,Y=2, Components=1, window="noTaper", RefDateTime=NA,
 timeFormat="%Y%m%d%H%M", RangeDateTime=list(Start=NA, End=NA), Units="hours",
  dt=0, Progressive=list(Interval=0, Increment=0), Period=list(Set=0,Start=0,
 Increment=1,End=0),header=FALSE, Skip=0, Colors="BW", Graphics="pdf",Output=list
 (Txt=FALSE,Dat=TRUE,Doc=TRUE,Graphs=FALSE),yLabel="", Console=FALSE,Debug=FALSE,
```

```
    IDcol="fileName",fileName=fileName,functionName="")
```

Arguments

TimeCol	Column number where time is found. Time format of the TimeCol is given by timeFormat parameter. Time is read to the minute. Seconds are discarded. Date and Time can be in any column. Valid values are as follows: 1) "numeric": numeric (in number of hours from starting time); timeFormat may be ignored. 2) A scalar: If date/time is in one column; or 3) A vector: If date/time are in two column, Date must be in the first of the 2 columns specified, and Time in the second, i.e., c(3,6).
Y	The column number(s) of the data to be analyzed. This is a numeric, either scalar or vector. May use any valid R vector, such as c(4,5) or a single number.
Components	Default=1. Indicates if this is a single or multiple component cosinor analysis, where the number of components is specified (0 is invalid). If doing a single component cosinor, set Components=1. If doing a multiple components model, set Components equal to the number of frequencies in the model.
window	Valid windowing function to be applied are: "noTaper","Hanning","Hamming", "Bartlett","Blackman"
RefDateTime	Date used as reference, and subtracted from all data dates, to make the number smaller. **Must be in the same time zone!!!!!** If RefDateTime = NA, uses the 1st date of the data as the RefDateTime. If RefDateTime = 0, uses midnight of the same day as start-of-data.
timeFormat	Can be "numeric", or any valid R time conversion specification, i.e., "%Y%m%d%H%M". See strptime for conversion specifications. If timeFormat= "numeric", time column in data file can be simple numbers (0 - 99999...) in Units from a reference time. If timeFormat= "numeric", the data are sorted by time to be sure they are ordered ascending. First must be smallest , and last largest. Time can also be in two columns (indicate in TimeCol Ex: c(1,2)); timeFormat is ignored when time is in two columns – the format use is %d/%m/%y in the first of the two columns, and %H:%M:%S or %H:%M in the second of the two
RangeDateTime	Specify in the form of a list: RangeDateTime=list(Start=12, End=0). $Start: Analysis is started at $Start. $Start may be before the 1st data date. If $Start = NA, the 1st data point is used as the StartDate. if $Start = 0, midnight of the 1st date is used as the StartDate. $End: Analysis ends at $End. $End may be after the last data date. if $End = NA, the last data point is used as the EndDate. if $End = 0, midnight at the end of the last date is used as the EndDate.
Units	Units (hours, years, weeks or days) for use by Interval and Increment arguments, as well as Period arguments (Note: only Hour is currently implemented.)
dt	When equidistant data, dt indicates the sampling interval. Data are assumed to be equidistant when this is nonzero.
Progressive	Specify in the form of a list: Progressive=list(Interval=0, Increment=0). $Interval: length of the time span being analyzed (in Units) – multiple are spans calculated. If 0, no progression is assumed; Interval is set to the full dataset length, and Increment = full data set. $Increment: (uses same Units as set for

	Interval) to shift forward for each successive Interval analyses. If 0, no progression is assumed; Interval is set to the full dataset length, and Increment = full data set
Period	Specify in the form of a list: Period=list(Set=0,Start=0,Increment=1,End=0). $Start : [only used if $Set=0]; this is the first (and longest) period to calculate, in units (as set by Units); (Interval/1). 0 is Default: the full time range of the data file is analyzed [in hours]. (RangeDateTime$End-RangeDateTime$Start)= MyData_length; or if progressive, Interval/1. $Increment : [only used if $Set=0] Increment from the starting period, in units (as set by Units). Defaults to 1; 0 is invalid – default will be used in that case. $End : [only used if $Set=0] Last (and smallest) period to calculate, in units (as set by Units), EXCLUSIVE. Defaults to 4; 0 is invalid – default will be used in that case. $Set : If Set=0, a series of periods are analyzed (spectrum) according to Period$Start, $Increment, $End (or their default value, if not specified). If Set is not equal to 0, it overrides Period$Start and $Increment, to completely specify a set of periods to analyze (as set by Units), ending at $End. $Set can be in the format of a single number, a vector, or R expression: c(1,2,3) or c(8,12:24) or seq(1,50, by=.75). When Components=1, each period specified in the vector will be assessed by cosinor independently. When parameter Components is >1, Period$Set must have a corresponding number of components, which are assessed together in a multiple-component cosinor. When 0, only the longest period, or the longest period per Interval, from a spectrum is listed on page 1 of the graphics output file. Otherwise, all periods are displayed in the graphic file.
header	T/F to indicate if the file has a header. Headers are used as variable names.
Skip	How many lines to skip at the top of the file (before the header). The header will be read next, after Skip lines are skipped (if header=TRUE).
Colors	Colors for a heatmap. "Heat" renders the heatmap in colors; "BW" renders the heatmap in grayscale
Graphics	The main results of CATCosinor are sent to a graphic file when Console=F. Default file output type is "pdf". Possible values: "jpg, pdf, tiff, png".
Output	Specify in the form of a list: list(Txt=F,Dat=T, Doc=T,Graphs=F). $Txt=T will capture the console output to a .txt file. $Dat=T will generate a .txt computer-readable file of tab delimited cosinor results: MESOR, Amplitude, Phase, standard error values, and others. It is useful for subsequent processing (by CAT, or excel, for example). $Doc=T will generate a nicely-formatted .rtf file, also readable by Word. $Graphs will enable a set of graphs plotting Data, Model, MESOR, Amplitude, Acrophase over time, or a heatmap. The exact files generated will vary with the functions performed. Heatmaps are only created when a progressive analysis least squares spectra is performed. $Graphs=F will disable printing of graphs for faster processing, if you do not need the files.
yLabel	Label for the Y axis on data graphs. If this is left blank (the default="") then the column headers are used for Y axis label.
Console	Default is F. When Console=T, output will be redirected to the RStudio Console, instead of an output file. Otherwise, sent to file type indicated in Graphics parameter.
Debug	Turn on when you want to see more diagnostic output for programming debug purposes.

IDcol	What to use as a unique identifier: either "fileName" or a column number. Default = "fileName"
fileName	The path to a data file to be processed. If this is blank, CATCosinor will solicit a file from the user by a dialogue box.
functionName	Brief user-defined name for this run, used to help distinguish it from other runs. Name is printed in output files.

Input Data requirements:

There is no need for data to be equidistant for the Cosinor. Data columns specified in Y are expected to be numeric. Input data files can be tab delimited, or comma delimited. There may be multiple columns of data for processing.

Output Data:

Sample graphics output file can be found in the Output section on the CAT web site for a sample of a full output file. All graphs contain the input data filename to clearly identify the data file under analysis, and a timestamp to show the time of analysis. Each graph also lists the column name being analyzed (or averaged), as well as the starting and ending times of analysis, as they vary slightly from the full data set. Graphic output files can be in JPG, PDF, TIFF or PNG. TIFF and PNG are higher resolution than jpeg and PDF. Plots in PNG and JPEG formats can easily be converted to many other bitmap formats, and both can be displayed in modern web browsers. The PNG format is lossless and is best for line diagrams and blocks of color. The JPEG format is lossy, but may be useful for image plots, for example. It is most often used in html web pages. TIFF is a meta-format: the default format written by tiff is lossless and stores RGB (and alpha where appropriate) values uncompressed-such files are widely accepted, which is their main virtue over PNG.

Author(s)

Cathy Lee Gierke, Ruth Ann Helget, Germaine Cornelissen-Guillaume

Maintainer: < <leegi001@umn.edu>>

References

http://5643947091146397 85.weebly.com/running-cat.html

Examples

```
#Data is systolic and dialstolic blood pressure; and heart rate at
#        30 minute intervals.
#---------------------  Vignette0    multi-component cosinor.
# multiple-components cosinor analysis of 24, 12 and 8 hours.  Columns
#       4,6,7 are processed.

#  Normally you would use these lines to read a file for use in CATkit
#        (use filePath format for your OS)
# filePath<-"~/file/path/Installing CAT/Vignette0"          # use for mac
# filePath<-"c:\file\path\Installing CAT\Vignette0"     # use for PC
# fileName<-file.path(filePath,'CLGi001.dat')
```

```
  # this line is used instead of the above since it is part of a package.
  file.copy(system.file("extdata", "CLGi001.dat", package = "CATkit"),
      tempdir(), overwrite = TRUE, recursive = FALSE, copy.mode = TRUE,
      copy.date = FALSE)
  filePath<-tempdir()
  fileName<-file.path(filePath,'CLGi001.dat')

  #fileName<-system.file("extdata", "CLGi001.dat", package = "CATkit")
  #
CATCosinor(TimeCol=2,Y=c(4,6,7), Components=3, window="noTaper", RefDateTime
    ="201302030000",  timeFormat="%Y%m%d%H%M",RangeDateTime =list(Start=0,
   End=0), Units='hours', dt=0, Progressive=list(Interval=0, Increment=0),
    Period=list(Set=c(24,12,8),Start=0,Increment=1,End=0),header=FALSE,
    Skip=0,Colors="BW",Graphics="pdf",Output=list(Txt=FALSE,Dat=TRUE,
    Doc=TRUE,Graphs=TRUE),Console=FALSE,Debug=FALSE,IDcol="fileName",
    fileName=fileName,functionName='Vignette0')

  #---------------------   Vignette1     Least Square spectrum
  #
  #  Normally you would use these lines to read a file for use in CATkit
  #     (use filePath format for your OS)
 # filePath<-"~/file/path/Installing CAT/Vignette1"          # use for mac
  # filePath<-"c:\file\path\Installing CAT\Vignette1"     # use for PC
  # fileName<-file.path(filePath,'Signal10-20.txt')

  # this line is used instead of the above since it is part of a package.
  file.copy(system.file("extdata", "Signal10-20.txt", package = "CATkit"),
      tempdir(), overwrite = TRUE, recursive = FALSE, copy.mode = TRUE,
      copy.date = FALSE)
  filePath<-tempdir()
  fileName<-file.path(filePath,'Signal10-20.txt')

  #fileName<-system.file("extdata", "Signal10-20.txt", package = "CATkit")
  #
 CATCosinor(TimeCol=1,Y=2, Components=1, window="noTaper", RefDateTime="0",
  timeFormat="%Y%m%d%H%M", RangeDateTime=list(Start=0, End=0),
  Units='hours', dt=0, Progressive=list(Interval=0, Increment=0), Period=
 list(Set=0,Start=144,Increment=1,End=4),header=FALSE, Skip=0, Colors="BW",
  Graphics="pdf",Output=list(Txt=FALSE,Dat=TRUE,Doc=TRUE,Graphs=TRUE),
  Console=FALSE,Debug=FALSE,fileName=fileName,functionName='Vignette1')

 # ---------------------   Vignette2     progressive least squares spectrum
  #
  #  Normally you would use these lines to read a file for use in CATkit
  #       (use filePath format for your OS)
 # filePath<-"~/file/path/Installing CAT/Vignette2"          # use for mac
  # filePath<-"c:\file\path\Installing CAT\Vignette2"     # use for PC
  # fileName<-file.path(filePath,'FWedited.txt')
  #
  # this line is used instead of the above since it is part of a package.
  file.copy(system.file("extdata", "FWedited.txt", package = "CATkit"),
      tempdir(), overwrite = TRUE, recursive = FALSE, copy.mode = TRUE,
      copy.date = FALSE)
```

```
filePath<-tempdir()
fileName<-file.path(filePath,'FWedited.txt')

#fileName<-system.file("extdata", "FWedited.txt", package = "CATkit")
#
CATCosinor(TimeCol=1,Y=2, Components=1, window="noTaper",RefDateTime
="199210192152",  timeFormat="%Y%m%d%H%M", RangeDateTime= list(Start
="199210200000", End="199211300000"), Units='hours', dt=0, Progressive=list
(Interval=168, Increment=24), Period=list(Set=0, Start=168,Increment=.5,
End=9.5),header=FALSE, Skip=0, Colors="BW", Graphics="pdf",Output=list
(Txt=FALSE,Dat=TRUE,Doc=FALSE,Graphs=TRUE),Console=FALSE,Debug=FALSE,
fileName=fileName,functionName='Vignette2')
```

CATparam *Population-mean Cosinor Test (PMCtest)*

Description

Statistical test performed on the output parameters from the PMC, testing the equality of rhythm parameters (MESOR, Amplitude, Acrophase) from two or more populations, considered singly or (Amplitude, Acrophase) jointly.

Usage

```
CATparam(data, fileName, colNames, GrpID = NA, VarID = NA, alpha = 0.05,
       header=FALSE, sep="\t", Output=list(Doc=TRUE, Txt=FALSE),
       functionName="", title="")
```

Arguments

data	(optional) An array of data; either a data frame object or a fileName must be provided
fileName	(optional) A data file with path; either a data frame or a fileName must be provided. "/" is used on Macs in the file pathname; "/" changes to "\" on PCs.
colNames	Specific column names are required by the program: PR, MESOR, Amp, Phi (case insensitive). Must be specified if column headers are absent, or if you wish to rename columns.
GrpID	(required) a vector of column names (or numbers) referring to factors to be used for comparing populations among levels (group identifiers) within each factor (GrpID). There must be at least 2 levels in any GrpID column, as a PMC test compares parameters between two or more populations. Populations are compared among all levels, as well as all level pairs, for each factor (GrpID). If multiple GrpIDs are specified, corresponding to multiple factors, comparisons are performed within each factor separately – populations identified by one GrpID are not compared to populations identified by a different GrpID.

VarID	(optional) a single column can be included as a Variable. There must be at least 2 unique names in any VarID column. All population comparisons defined under GrpID are performed separately for each level in VarID. A column would be specified as a VarID if it makes no sense to combine multiple levels (values) for PMC – such as combining temperature with blood pressure, or rhythm parameters related to different trial periods.
alpha	significance level (p) for testing (default = .05).
header	does the file have a header? (default = TRUE); column headers are used as column names unless colNames is used.
sep	is the file tab delimited ("\t") or comma delimited (",") (default = "\t").
Output	Doc=TRUE will output an RTF file; Doc=FALSE will not. Output is always sent to the console.
functionName	a short descriptor, this will be used as a part of the file name for use in identifying output.
title	A longer text descriptor printed out in the header of the result file.

Input Data:

The PMCtest uses the parameters resulting from a single- or multiple-component single cosinor as input.

Data format: Comma or tab delimited file. There are four required column headers, one optional VarID (variable) column, and one or more GrpID (factor) columns: PR, MESOR, Amp, Phi, variable, factor1, factor2, ...

Output:

Data Format: An .rtf data file is written to the same file path where the data file is found.

Returned: An output data frame is returned from the function call, and printed to the console.

Author(s)

Cathy Lee Gierke, Germaine Cornelissen-Guillaume

Maintainer: < Cathy Lee Gierke <leegi001@umn.edu>>

References

http://564394709114639785.weebly.com/running-cat-PMCtest.html

Examples

```
# Data is from east Indian subjects:  lipids, smoking, diet, gender, age group
   #        every 6 hours.
 #---------------------  Vignette5   Population-Mean Cosinor Parameter test
   # testing equivalency of parameters from the Population Mean Cosinor
  # according to behavioral factors and lipids for a population of subjects

   #  Normally you would use these lines to read a file for use in CATkit
```

```
#        (use filePath format for your OS)
# filePath<-"~/file/path/Installing CAT/Vignette4"          # use for mac
# filePath<-"c:\file\path\Installing CAT\Vignette4"     # use for PC
# fileName<-file.path(filePath,'cos02x03.csv')

# this line is used instead of the above since it is part of a package.
file.copy(system.file("extdata", "cos02x03.csv", package = "CATkit"),
    tempdir(), overwrite = TRUE, recursive = FALSE, copy.mode = TRUE,
    copy.date = FALSE)
filePath<-tempdir()
fileName<-file.path(filePath,'cos02x03.csv')

#fileName<-system.file("extdata", "cos02x03.csv", package = "CATkit")
CATparam(fileName=fileName, VarID=8, GrpID=c(5,10),header=TRUE,
sep=",",functionName="V8,G5,10", title="PMCtest V8,G5,10")
```

CATpmc *Population-mean Cosinor (PMC)*

Description

When data are collected as a function of time on 3 or more individuals, the population-mean cosinor procedure renders it possible to make inferences concerning a population rhythm, provided the individuals considered represent a random sample from that population. Each individual series is analyzed by the single- or multiple-component single cosinor. The PMC uses takes these results as input. Assuming that the within-individual variances are the same, the PMC estimates the population rhythm parameters by calculating the arithmetic mean of individual MESORs and the vectorial average of individual amplitude-acrophase pairs.

Usage

```
CATpmc(data, fileName, colNames, VarID=NA, GrpID = NA, alpha = 0.05,
        header=FALSE, sep="\t", Output=list(Doc=TRUE, Txt=FALSE),
        functionName="", title="")
```

Arguments

fileName	(optional) A data file with path; either a data frame (cosinor parameters) or fileName must be provided. "/" is used on Macs in the file pathname; "/" changes to "\" on PCs.
data	(optional) An array of data; either a data frame object or fileName must be provided.
colNames	Specific column names are required by the program: PR, MESOR, Amp, Phi (case insensitive). Must be specified if column headers are absent, or if you wish to rename columns.

GrpID	(optional) a vector of column names (or numbers) referring to factors to be used for grouping populations by levels in each factor (group identifiers). A PMC is performed on the full population, and on each sub-population grouped by factor level, for each GrpID specified; if multiple GrpIDs are specified, corresponding to multiple factors, PMC is also performed on all combinations across all factor levels. If no GrpIDs are specified, only the full population PMC is performed.
VarID	(optional) a single column can be included as a Variable. Each population identified by variable name (value) in the VarID column is compared with each population comparison defined under GrpID. A column would be specified as a VarID if it makes no sense to combine multiple levels (values) for PMC – such as combining temperature with blood pressure, or rhythm parameters related to different trial periods.
alpha	significance level (p) for testing (default = .05).
header	does the file have a header? (default = TRUE); column headers are used as column names unless colNames is used.
sep	is the file tab delimited ("\t") or comma delimited (",") (default = "\t").
Output	Doc=TRUE will output an RTF file; Doc=FALSE will not. Output is always sent to the console.
functionName	a short descriptor, this will be used as a part of the file name for use in identifying output.
title	A longer text descriptor printed out in the header of the result file.

Input Data:

The PMC uses the parameters resulting from a single- or multiple-component single cosinor as input.

Data format: Comma or tab delimited file. There are four required column headers, one optional VarID (variable) column, and any number of (optional) GrpID (factor) columns: PR, MESOR, Amp, Phi, variable, factor1, factor2, ...

Output:

Data Format: An .rtf data file is written to the same file path where the data file is found.

Returned: An output data frame is returned from the function call, and printed to the console.

Author(s)

Cathy Lee Gierke, Germaine Cornelissen-Guillaume

Maintainer: < Cathy Lee Gierke <leegi001@umn.edu>>

References

http://564394709114639785.weebly.com/running-cat-pmc.html

Examples

```
# Data is from east Indian subjects:  lipids, smoking, diet, gender, age group
   #        every 6 hours.
   #---------------------  Vignette4    Population-Mean Cosinor
 # PMC according to behavioral factors and lipids for a population of subjects

   #  Normally you would use these lines to read a file for use in CATkit
   #        (use filePath format for your OS)
  # filePath<-"~/file/path/Installing CAT/Vignette4"          # use for mac
   # filePath<-"c:\file\path\Installing CAT\Vignette4"     # use for PC
   # fileName<-file.path(filePath,'cos02x03.csv')

   # this line is used instead of the above since it is part of a package.
   file.copy(system.file("extdata", "cos02x03.csv", package = "CATkit"),
       tempdir(), overwrite = TRUE, recursive = FALSE, copy.mode = TRUE,
       copy.date = FALSE)
   filePath<-tempdir()
   fileName<-file.path(filePath,'cos02x03.csv')

   #fileName<-system.file("extdata", "cos02x03.csv", package = "CATkit")
   CATpmc(fileName=fileName, VarID=8, GrpID=c(5,10),header=TRUE,
   sep=",",functionName="V8,G5,10", title="Vignette3 V8,G5,10")
```

Index