

Package ‘pkdata’

May 9, 2026

Type Package

Title Creates Pharmacokinetic/Pharmacodynamic (PK/PD) Data

Version 0.1.0

Date 2020-09-28

Maintainer Cole Beck <cole.beck@vumc.org>

Description Prepare pharmacokinetic/pharmacodynamic (PK/PD) data for PK/PD analyses.
This package provides functions to standardize infusion and bolus dose data while linking it to drug level or concentration data.

License GPL (>= 3)

Depends lubridate

RoxygenNote 7.1.1

NeedsCompilation no

Author Cole Beck [aut, cre] (ORCID: <<https://orcid.org/0000-0002-6849-6255>>),
Leena Choi [aut]

Repository CRAN

Date/Publication 2020-10-01 09:00:02 UTC

Contents

pkdata-package	2
conformDoses	2
fixDuplicates	3
guessDateFormat	4
imputeDoses	5
parse_dates	6
prepareDoses	7
sortDoses	8
trimDoses	9

Index	11
--------------	-----------

pkdata-package *PK Data*

Description

This package will prepare data sets for PK data analysis.

Author(s)

Cole Beck, Leena Choi

Maintainer: Cole Beck <cole.beck@vumc.org>

conformDoses *Create a Conforming Dose Data Set*

Description

Modify dose data such that it conforms for future use.

Usage

```
conformDoses(
  doseData,
  idVar = "id",
  dateVar = "date.dose",
  infusionDoseTimeVar = NULL,
  infusionDoseVar = NULL,
  bolusDoseTimeVar = NULL,
  bolusDoseVar = NULL,
  otherDoseTimeVar = NULL,
  otherDoseVar = NULL,
  otherVars = NULL
)
```

Arguments

doseData	data.frame; data set with dose information
idVar	character string; name of ID variable, defaults to id
dateVar	character string; name of date variable, defaults to date.dose
infusionDoseTimeVar	character string; name of infusion dose time variable
infusionDoseVar	character string; name of infusion dose variable

bolusDoseTimeVar character string; name of bolus dose time variable
 bolusDoseVar character string; name of bolus dose variable
 otherDoseTimeVar character string; name of additional dose time variable
 otherDoseVar character string; name of additional dose variable
 otherVars character string; name of other variables within data set

Details

Date-time variables are given a consistent format. Invalid records are removed. The data set is sorted by date.

Value

data.frame, containing dose data

Author(s)

Cole Beck

fixDuplicates	<i>Remove Duplicate Dose Data</i>
---------------	-----------------------------------

Description

Modify dose data to remove duplicate dose values.

Usage

```
fixDuplicates(  
  data,  
  idVar = "id",  
  infusionDoseTimeVar = NULL,  
  infusionDoseVar = NULL,  
  moveBolus = FALSE,  
  bolusDoseTimeVar = NULL,  
  bolusDoseVar = NULL  
)
```

Arguments

data data.frame; data set with dose information
 idVar character string; name of ID variable, defaults to id
 infusionDoseTimeVar character string; name of infusion dose time variable

infusionDoseVar character string; name of infusion dose variable
 moveBolus logical; allow duplicate infusion doses to be categorized as bolus
 bolusDoseTimeVar character string; name of bolus dose time variable
 bolusDoseVar character string; name of bolus dose variable

Details

Some duplicates can be adjusted by changing the date-time rounding. Others may be converted from one dose type to another, such as moving infusion to bolus.

Value

data.frame, containing dose data

Author(s)

Cole Beck

guessDateFormat *Determine Format of Date and Date-Time Variables*

Description

Given a vector of dates or date-times, determine the format if one is used consistently. If inconsistencies are found, the function will fail. See [strptime](#) for examples of format specifications.

Usage

```
guessDateFormat(x)
```

Arguments

x character vector of dates or date-times

Details

guessDateFormat is rigid when determining the date format. For date-times it expects the date and time parts to be separated by a space. It does not recognize all format specifications, such as the AM/PM indicator. The time part should have at least one colon to separate hours from minutes. The date part may have any separator as non-numerical values are removed before parsing.

Value

Returns a character string representing the format of the date-time variables.

Author(s)

Cole Beck

Examples

```
x <- c("2014-01-15", "20140202")
guessDateFormat(x)
```

`imputeDoses`*Impute Dose Data*

Description

Use last-observation-carried-forward to impute missing dose values by hour.

Usage

```
imputeDoses(
  data,
  idVar = "id",
  dateVar = "date.dose",
  infusionDoseTimeVar = NULL,
  infusionDoseVar = NULL,
  maxskips = 3
)
```

Arguments

<code>data</code>	data.frame; data set with dose information
<code>idVar</code>	character string; name of ID variable, defaults to id
<code>dateVar</code>	character string; name of date variable, defaults to date.dose
<code>infusionDoseTimeVar</code>	character string; name of infusion dose time variable
<code>infusionDoseVar</code>	character string; name of infusion dose variable
<code>maxskips</code>	integer; maximum number of missed doses that may be imputed

Details

Creates skips column

Value

data.frame, containing dose data

Author(s)

Cole Beck

parse_dates	<i>Parse Date and Date-Time Variables</i>
-------------	---

Description

Given a vector of dates or date-times, create Date or POSIXct variables.

Usage

```
parse_dates(x, tz = getOption("pkdata.tz", ""))
```

Arguments

x	character vector of dates or date-times
tz	character string; specifies the time zone to be used for the conversion. Defaults to the current time zone.

Details

parse_dates calls [parse_date_time](#) from the lubridate package. While [parse_date_time](#) accepts multiple date formats, parse_dates requires a consistent format.

Value

vector of Date or POSIXct objects

Author(s)

Cole Beck

Examples

```
x <- c("2014-01-15", "20140202")
parse_dates(x)
x <- c("2014-01-15 01:51", "20140202 04:35:18")
parse_dates(x)
```

 prepareDoses

Prepare the Dose Data Set

Description

Create a dose data set with conforming data, and remove invalid records.

Usage

```
prepareDoses(
  doseData,
  drugLevelData,
  drugLevelID = "id",
  drugLevelTimeVar = "date.time",
  drugLevelVar = "fent.level",
  idVar = "id",
  dateVar = "date.dose",
  infusionDoseTimeVar = NULL,
  infusionDoseVar = NULL,
  bolusDoseTimeVar = NULL,
  bolusDoseVar = NULL,
  otherDoseTimeVar = NULL,
  otherDoseVar = NULL,
  otherVars = NULL,
  lookForward = 7
)
```

Arguments

doseData	data.frame; data set with dose information
drugLevelData	data.frame; data set with drug level data
drugLevelID	character string; name of ID variable, defaults to id
drugLevelTimeVar	character string; name of date-time variable, defaults to date.time
drugLevelVar	character string; name of drug level variable, defaults to fent.level
idVar	character string; name of ID variable, defaults to id
dateVar	character string; name of date variable, defaults to date.dose
infusionDoseTimeVar	character string; name of infusion dose time variable
infusionDoseVar	character string; name of infusion dose variable
bolusDoseTimeVar	character string; name of bolus dose time variable
bolusDoseVar	character string; name of bolus dose variable

otherDoseTimeVar character string; name of additional dose time variable

otherDoseVar character string; name of additional dose variable

otherVars character string; name of other variables within data set

lookForward integer; initializes the time frame window with the number of days prior to the first drug level data; defaults to 7

Details

Wrapper function for [conformDoses](#) and [trimDoses](#).

Value

data.frame, containing dose data

Author(s)

Cole Beck

Examples

```
options(pkdata.tz='America/Chicago')
dose.file <- read.csv(system.file('extdata', 'dosage.csv', package = 'pkdata'),
  stringsAsFactors = FALSE)
drug.level.file <- read.csv(system.file('extdata', 'druglevel.csv', package = 'pkdata'),
  stringsAsFactors = FALSE)
prepped <- prepareDoses(dose.file, drug.level.file,
  infusionDoseTimeVar='inf.time', infusionDoseVar='inf.dose',
  bolusDoseTimeVar='bol.time', bolusDoseVar='bol.dose',
  otherDoseTimeVar='patch.time', otherDoseVar='patch.dose',
  otherVars=c('gender','weight'))
```

sortDoses

Sort Dose Data Set

Description

Helper function to sort dose data set by ID and time variables.

Usage

```
sortDoses(
  data,
  idVar = "id",
  infusionDoseTimeVar = NULL,
  bolusDoseTimeVar = NULL,
  otherDoseTimeVar = NULL
)
```

Arguments

data data.frame; data set with dose information
idVar character string; name of ID variable, defaults to id
infusionDoseTimeVar character string; name of infusion dose time variable
bolusDoseTimeVar character string; name of bolus dose time variable
otherDoseTimeVar character string; name of additional dose time variable

Details

Sort order will be the ID variable, then infusion time, bolus time, or other dose date-time variables.

Value

data.frame, containing dose data

Author(s)

Cole Beck

trimDoses	<i>Trim Dose Data Set</i>
-----------	---------------------------

Description

Remove invalid doses by creating a time frame window constructed from drug level data.

Usage

```
trimDoses(  
  doseData,  
  drugLevelData,  
  drugLevelID = "id",  
  drugLevelTimeVar = "date.time",  
  drugLevelVar = "fent.level",  
  infusionDoseTimeVar = NULL,  
  infusionDoseVar = NULL,  
  bolusDoseTimeVar = NULL,  
  bolusDoseVar = NULL,  
  otherDoseTimeVar = NULL,  
  otherDoseVar = NULL,  
  lookForward = 7,  
  last = NA  
)
```

Arguments

doseData	data.frame; data set with dose information
drugLevelData	data.frame; data set with drug level data
drugLevelID	character string; name of ID variable, defaults to id
drugLevelTimeVar	character string; name of date-time variable, defaults to date.time
drugLevelVar	character string; name of drug level variable, defaults to fent.level
infusionDoseTimeVar	character string; name of infusion dose time variable
infusionDoseVar	character string; name of infusion dose variable
bolusDoseTimeVar	character string; name of bolus dose time variable
bolusDoseVar	character string; name of bolus dose variable
otherDoseTimeVar	character string; name of additional dose time variable
otherDoseVar	character string; name of additional dose variable
lookForward	integer; initializes the time frame window with the number of days prior to the first drug level data; defaults to 7
last	integer; sets the end of the time frame window to be "last" days after the first dose date, rather than the date of the last drug level data

Details

The time frame window is generally seven days before drug level data up through the last drug level data record. The window can be adjusted by setting the lookForward and last arguments.

Value

data.frame, containing trimmed dose data

Author(s)

Cole Beck

Index

- * **package**
 - pkdata-package, [2](#)
- conformDoses, [2, 8](#)
- fixDuplicates, [3](#)
- guessDateFormat, [4](#)
- imputeDoses, [5](#)
- parse_date_time, [6](#)
- parse_dates, [6](#)
- pkdata (pkdata-package), [2](#)
- pkdata-package, [2](#)
- prepareDoses, [7](#)
- sortDoses, [8](#)
- strptime, [4](#)
- trimDoses, [8, 9](#)