Package 'tidyrhrv'

July 22, 2025

Title Read, Iteratively Filter, and Analyze Multiple ECG Datasets

Version 1.1.0

Description Allows users to quickly load multiple patients' electrocardiographic (ECG) data at once and conduct relevant time analysis of heart rate variability (HRV) without manual edits from a physician or data cleaning specialist. The package provides the unique ability to iteratively filter, plot, and store time analysis results in a data frame while writing plots to a predefined folder. This streamlines the workflow for HRV analysis across multiple datasets. Methods are based on Rodríguez-Liñares et al. (2011) <doi:10.1016/j.cmpb.2010.05.012>. Examples of applications using this package include Kwon et al. (2022) <doi:10.1007/s10286-022-00865-2> and Lawrence et al. (2023) <doi:10.1016/j.autneu.2022.103056>.

License MIT + file LICENSE

Suggests testthat (>= 3.0.0), ggplot2, vroom, readr

Config/testthat/edition 3

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Imports dplyr, tidyr, RHRV, purrr, magrittr, pracma, tibble, stats, grDevices

Author Steven Lawrence [aut, cre] (<https://orcid.org/0000-0002-7595-8323>)

Maintainer Steven Lawrence <stevenlawrence.r@gmail.com>

NeedsCompilation no

Contents

---

filter_tilt   *Filter Data from prep_data Function Iteratively*

Description

Uses window functions native to the RHRV package and hampel window filter to iteratively clean heart rate variability data.

Usage

```
filter_tilt(.data, g, l)
```

Arguments

`.data`	A tilt data frame produced by prep_data function
`g`	Numeric value representing the upper bound multiplier for filtering (greater than median of spline)
`l`	Numeric value representing the lower bound multiplier for filtering (less than median of spline)

Value

A data frame with filtered contents

Examples

```
# This example requires RHRV and pracma packages which may not be available
# Create minimal toy data for demonstration
temp_dir <- tempdir()

# Generate realistic HRV data with some outliers
set.seed(123)
hrv_data <- data.frame(
  Time = seq(0, 20, by = 0.8),
  niHR = c(70 + rnorm(20, 0, 3), 120, 72 + rnorm(5, 0, 3)), # Include outlier
  RR = c(60/70 + rnorm(20, 0, 0.05), 0.5, 60/72 + rnorm(5, 0, 0.05))
)

write.csv(hrv_data, file.path(temp_dir, "hrv_test.csv"), row.names = FALSE)

# Read and prepare data
raw_data <- read_tilt(temp_dir, read.csv)
prepped_data <- prep_data(raw_data, "Time", "niHR", "RR")

# Apply filtering (requires RHRV package)
if (requireNamespace("RHRV", quietly = TRUE) &&
    requireNamespace("pracma", quietly = TRUE)) {
  filtered_data <- filter_tilt(prepped_data, g = 1.2, l = 0.8)
  print("Filtering completed")
} else {
  message("RHRV and pracma packages required for filtering")
}
```

```
  # Clean up
  unlink(file.path(temp_dir, "hrv_test.csv"))
```

---

plot_tilt	*Store Plots of RR Series in Folders and Produce RMSSD and pNN50 Output*

Description

Creates plots of heart rate variability data and saves them to specified folders while calculating time domain metrics (RMSSD and pNN50).

Usage

```
plot_tilt(.data, folder, type)
```

Arguments

.data	A data frame containing HRV data from previous tidyrhrv functions
folder	A character string specifying the folder name for saved plots
type	A character string indicating whether data are "filtered" or "original"

Value

A list of data frames containing RMSSD and pNN50 values for each dataset

Examples

```
# This example requires RHRV package for HRV analysis
if (requireNamespace("RHRV", quietly = TRUE)) {
  temp_dir <- tempdir()

  # Generate synthetic HRV data
  hrv_data <- data.frame(
    Time = seq(0, 25, by = 0.8),
    niHR = 75 + rnorm(32, 0, 4),
    RR = 60/75 + rnorm(32, 0, 0.08)
  )

  write.csv(hrv_data, file.path(temp_dir, "plot_test.csv"), row.names = FALSE)

  # Read and prepare data
  raw_data <- read_tilt(temp_dir, read.csv)
  prepped_data <- prep_data(raw_data, "Time", "niHR", "RR")

  # Create plots and calculate metrics
  plot_folder <- "test_hrv_plots"
  results <- plot_tilt(prepped_data, plot_folder, "original")

  print("Plots created and metrics calculated")

  # Clean up
```

```
  unlink(file.path(temp_dir, "plot_test.csv"))
  unlink(plot_folder, recursive = TRUE)
} else {
  message("RHRV package required for this function")
}
```

---

prep_data   *Prepare Data for tidyrhrv Functions*

Description

This function helps to manipulate the data into a dataset readable by other tidyrhrv functions by standardizing column names.

Usage

```
prep_data(.data, time, HR, RR)
```

Arguments

`.data`	A nested data frame from read_tilt function
`time`	A character string specifying the name of the time column
`HR`	A character string specifying the name of the heart rate column
`RR`	A character string specifying the name of the RR interval column

Value

A nested data frame with standardized column names (Time, niHR, RR)

Examples

```
# Create toy HRV data
temp_dir <- tempdir()

# Generate synthetic data with different column names to demonstrate prep_data
time_seq <- seq(0, 30, by = 0.8)
hrv_data <- data.frame(
  time_col = time_seq,
  heart_rate = 75 + rnorm(length(time_seq), 0, 5),
  rr_interval = 60/75 + rnorm(length(time_seq), 0, 0.1)
)

# Write toy data file
write.csv(hrv_data, file.path(temp_dir, "test_subject.csv"), row.names = FALSE)

# Read the data using read_tilt
raw_data <- read_tilt(temp_dir, read.csv)

# Prepare data with standardized column names
prepped_data <- prep_data(raw_data, "time_col", "heart_rate", "rr_interval")

# Check the standardized column names
```

```
print(names(prepped_data$contents[[1]]))

# Clean up
unlink(file.path(temp_dir, "test_subject.csv"))
```

---

read_tilt	*Read Multiple Tilt Data Files*

Description

Read in all tilt data files in a folder at once to create a nested data frame that can be processed by other tidyrhrv functions.

Usage

```
read_tilt(path, file_type)
```

Arguments

path	A character string specifying the path to the folder containing data files
file_type	A function to read the files (e.g., readr::read_csv, read.table, etc.)

Value

A nested data frame with 'names' and 'contents' columns

Examples

```
# Create toy HRV data files in temporary directory
temp_dir <- tempdir()

# Generate synthetic HRV data for two subjects
hrv_data1 <- data.frame(
  Time = seq(0, 60, by = 0.8),  # 60 seconds of data
  HR = 70 + rnorm(76, 0, 5),    # Heart rate around 70 bpm
  RR = 60/70 + rnorm(76, 0, 0.1) # RR intervals
)

hrv_data2 <- data.frame(
  Time = seq(0, 45, by = 0.7),  # 45 seconds of data
  HR = 80 + rnorm(65, 0, 4),    # Heart rate around 80 bpm
  RR = 60/80 + rnorm(65, 0, 0.08)
)

# Write toy data files
write.csv(hrv_data1, file.path(temp_dir, "subject1.csv"), row.names = FALSE)
write.csv(hrv_data2, file.path(temp_dir, "subject2.csv"), row.names = FALSE)

# Read the data using read_tilt
tilt_data <- read_tilt(temp_dir, read.csv)
print(tilt_data)

# Clean up
unlink(file.path(temp_dir, c("subject1.csv", "subject2.csv")))
```

```
# For reading other file types (requires additional packages)
# data <- read_tilt(temp_path, readr::read_csv)
```

%>%   *Pipe operator*

Description

See magrittr::%>% for details.

Usage

```
lhs %>% rhs
```

Value

Returns the result of applying the right-hand side function to the left-hand side argument.

Index