

Package ‘Inflongitudinal’

February 24, 2026

Type Package

Title Detecting Influential Subjects in Longitudinal Data

Version 0.1.0

Description Provides methods for detecting influential subjects in longitudinal data, particularly when observations are collected at irregular time points. The package identifies subjects whose response trajectories deviate substantially from population-level patterns, helping to diagnose anomalies and undue influence on model estimates.

Imports ggplot2, dplyr, mice

License GPL-3

Encoding UTF-8

LazyData true

Depends R (>= 4.1.0)

RoxygenNote 7.3.2

NeedsCompilation no

Author Atanu Bhattacharjee [aut],
Tanmoy Majumdar [aut, cre],
Gajendra Kumar Vishwakarma [aut]

Maintainer Tanmoy Majumdar <tanmoy.stat.ku@gmail.com>

Repository CRAN

Date/Publication 2026-02-24 19:20:15 UTC

Contents

infsdata	2
Phenobarb	2
rld	3
sld	4
slm	6
tvm	8
tvm.imputation	9
wlm	10

Index**12**

 infsdata *Simulated Longitudinal Data*

Description

This dataset consists of 10000 subjects with irregular observation times and influential observations.

Usage

infsdata

Format

A data frame contains:

subject_id Unique identifier for each subject

time Time points of observation

response Simulated response value

subject_type Category of subject (e.g., Influential, Non-Influential)

Source

Simulated dataset

 Phenobarb *Phenobarb Dataset*

Description

This dataset contains longitudinal data on Phenobarbital concentration levels in newborn infants.

Usage

Phenobarb

Format

A data frame with 744 observations on the following 7 variables:

Subject An ordered factor identifying the infant.

Wt A numeric vector giving the birth weight of the infant (kg).

Apgar An ordered factor giving the 5-minute Apgar score for the infant. This is an indication of the newborn's health.

ApgarInd A factor indicating whether the 5-minute Apgar score is < 5 or >= 5.

time A numeric vector giving the time when the sample is drawn or the drug is administered (hr).

dose A numeric vector giving the dose of drug administered (μ g/kg).

conc A numeric vector giving the phenobarbital concentration in the serum (μ g/L).

Source

MEMSS R package

Examples

```
data(Phenobarb)
head(Phenobarb)
```

rld

Relative Longitudinal Difference (RLD)

Description

This function identifies influential subjects in longitudinal data based on their relative change in response over time. It helps in detecting subjects whose response values exhibit significant fluctuations beyond a specified threshold (k standard deviations).

Usage

```
rld(data, subject_id, time, response, k = 2, verbose = FALSE)
```

Arguments

data	A data frame containing the longitudinal data.
subject_id	A column specifying the column name for subject IDs.
time	A column specifying different time points that observations are measured like 0 as baseline, 1 as first visit etc.
response	A column specifying the column name for response values.
k	A numeric value (default = 2) used to define the threshold for detecting influential subjects.
verbose	Logical; if TRUE, prints informative messages during execution.

Details

The function follows these steps:

- Computes the relative change in response values over time for each subject.
- Calculates the threshold based on k standard deviations from the mean relative change.
- Identifies subjects whose relative change exceeds the threshold.
- Separates data into influential and non-influential subjects.
- Generates visualizations to highlight influential subjects.

This method is particularly useful for detecting subjects with extreme response variations in longitudinal studies.

Value

A list containing:

<code>influential_subjects</code>	IDs of influential subjects.
<code>influential_data</code>	Data frame of influential subjects.
<code>non_influential_data</code>	Data frame of non-influential subjects.
<code>relative_change_plot</code>	Plot of max relative change per subject.
<code>longitudinal_plot</code>	Plot of longitudinal data with influential subjects highlighted.
<code>IS_table</code>	A data frame containing the Influence Score (IS) and the Partial Influence Score (PIS) values for each subject at each time point.

See Also

`tvm`, `wlm`, `sld`, `slm`

Examples

```
data(infsdata)
infsdata <- infsdata[1:5,]
result <- rld(infsdata, "subject_id", "time", "response", k = 2)
print(result$influential_subjects)
head(result$influential_data)
head(result$non_influential_data)
```

sld

Simple Longitudinal Difference (SLD)

Description

This function detects influential subjects in a longitudinal dataset by analyzing their successive differences. It calculates the successive differences for each subject, determines a threshold using the mean and standard deviation, and identifies subjects whose maximum successive difference exceeds this threshold. This approach helps in detecting abrupt changes in subject responses over time.

Usage

```
sld(data, subject_id, time, response, k = 2, verbose = FALSE)
```

Arguments

<code>data</code>	A data frame containing longitudinal data.
<code>subject_id</code>	A column specifying the column name for subject IDs.
<code>time</code>	A column specifying different time points that observations are measured.
<code>response</code>	A column specifying the column name for the response variable.
<code>k</code>	A numeric value for the threshold parameter (default is 2), representing the number of standard deviations used to define the threshold.
<code>verbose</code>	Logical; if TRUE, prints informative messages during execution.

Details

The function follows these steps:

- Computes successive differences for each subject.
- Calculates the mean and standard deviation of these differences across all subjects.
- Defines a threshold as k standard deviations from the mean.
- Identifies subjects whose maximum successive difference exceeds this threshold.
- Separates data into influential and non-influential subjects.
- Visualizes the results using `ggplot2`.

This method is useful for identifying subjects with sudden changes in their response patterns over time.

Value

A list containing:

<code>influential_subjects</code>	A vector of subject IDs identified as influential.
<code>influential_data</code>	A data frame containing data for influential subjects.
<code>non_influential_data</code>	A data frame containing data for non-influential subjects.
<code>successive_difference_plot</code>	A <code>ggplot</code> object visualizing maximum successive differences across subjects.
<code>longitudinal_plot</code>	A <code>ggplot</code> object displaying longitudinal data with influential subjects highlighted.
<code>IS_table</code>	A data frame containing the Influence Score (IS) and the Partial Influence Score (PIS) values for each subject at each time point.

See Also

`tvm`, `wlm`, `slm`, `rld`

Examples

```
data(infsdata)
infsdata <- infsdata[1:5,]
result <- sld(infsdata, "subject_id", "time", "response", k = 2)
print(result$influential_subjects)
head(result$influential_data)
head(result$non_influential_data)
```

slm

Simple Longitudinal Mean (SLM)

Description

This function detects influential subjects in longitudinal data based on their mean response values. It identifies subjects whose mean response deviates significantly beyond a specified threshold (defined as k standard deviations from the mean). The function provides a summary of influential subjects, separates the data into influential and non-influential subjects, calculates influence scores, and visualizes the results using `ggplot2`.

Usage

```
slm(data, subject_id, time, response, k = 2, verbose = FALSE)
```

Arguments

<code>data</code>	A data frame containing longitudinal data.
<code>subject_id</code>	A column specifying the column name representing subject identifiers.
<code>time</code>	A column specifying different time points that observations are measured.
<code>response</code>	A column specifying the column name representing response values.
<code>k</code>	A numeric value representing the threshold (number of standard deviations from the mean) to classify a subject as influential.
<code>verbose</code>	Logical; if TRUE, prints informative messages during execution.

Details

The function follows these steps:

- Calculates the mean and standard deviation of the response variable across all subjects.
- Determines the threshold for influence based on k standard deviations from the mean.
- Identifies subjects whose mean response falls outside this threshold.
- Calculates the Influence Score (IS) for each subject as the absolute deviation of their mean from the overall mean.
- Calculates the Proportional Influence Score (PIS) for each subject as IS divided by the overall standard deviation.

- Separates data into influential and non-influential subjects.
- Visualizes the distribution of responses and highlights influential subjects.

This method is useful for detecting outliers and understanding the impact of extreme values in longitudinal studies.

Value

A list containing:

<code>influential_subjects</code>	A vector of subject IDs identified as influential.
<code>influential_data</code>	A data frame containing data for influential subjects.
<code>non_influential_data</code>	A data frame containing data for non-influential subjects.
<code>influence_scores</code>	A data frame with subject IDs, mean response, IS (Influence Score), and PIS (Proportional Influence Score).
<code>mean_plot</code>	A ggplot object showing mean responses per subject with influential subjects highlighted.
<code>longitudinal_plot</code>	A ggplot object visualizing longitudinal response trends, with influential subjects highlighted.
<code>IS_table</code>	A data frame containing the Influence Score (IS) and the Partial Influence Score (PIS) values for each subject.

See Also

`tvm`, `wlm`, `sld`, `rld`

Examples

```
data(infsdata)
infsdata <- infsdata[1:5,]
result <- slm(infsdata, "subject_id", "time", "response", 2)
print(result$influential_subjects)
head(result$influential_data)
head(result$non_influential_data)
head(result$influence_scores)
print(result$mean_plot)
print(result$longitudinal_plot)
```

tvm *Time-Varying Mean (TVM)*

Description

This function detects influential subjects based on their response values at different time points. It calculates the mean and standard deviation of responses at each time point and flags subjects whose response values deviate significantly beyond a threshold. The function also generates plots to visualize influential observations and their trends over time. It also computes the Influence Score (IS) and Partial Influence Score (PIS) for each observation.

Usage

```
tvm(data, subject_id, time, response, k = 2, verbose = FALSE)
```

Arguments

data	A dataframe containing the longitudinal data.
subject_id	A column specifying the column name for subject IDs.
time	A column specifying different time points that observations are measured.
response	A column specifying the column name for response values.
k	A numeric value specifying the number of standard deviations to use as the threshold (default = 2).
verbose	Logical; if TRUE, prints informative messages during execution.

Details

The function follows these steps:

- Computes the mean and standard deviation of response values at each time point.
- Calculates Influence Score (IS) and Partial Influence Score (PIS) for each observation.
- Identifies subjects whose response values exceed the threshold based on k standard deviations.
- Separates influential and non-influential subjects for further analysis.
- Generates visualizations of mean responses and highlights influential subjects in a longitudinal plot.

This method is useful for identifying outliers and understanding variability in longitudinal studies.

Value

A list containing:

influential_subjects

A vector of subject IDs identified as influential.

influential_data

A data frame containing data for influential subjects.

<code>influential_time_data</code>	A data frame containing data for influential subjects with only the influential time points.
<code>non_influential_data</code>	A data frame containing data for non-influential subjects.
<code>mean_response_plot</code>	A plot visualizing the mean response values across time points.
<code>longitudinal_plot</code>	A final plot highlighting influential subjects over time.
<code>IS_table</code>	A data frame containing the Influence Score (IS) and the Partial Influence Score (PIS) values for each subject at each time point.

See Also

`slm`, `wlm`, `sld`, `rld`

Examples

```
data(infsdata)
infsdata <- infsdata[1:5,]
result <- tvm(infsdata, "subject_id", "time", "response", 2)
print(result$influential_subjects)
head(result$influential_data)
head(result$non_influential_data)
head(result$influential_time_data)
head(result$IS_table)
head(result$PIS_table)
result$mean_response_plot
result$longitudinal_plot
```

`tvm.imputation`

tvm.imputation: Impute Influential Responses in Longitudinal Data

Description

This function identifies influential response values using the ‘tvm’ function, replaces them with NA, and imputes the missing values using the ‘mice’ package.

Usage

```
tvm.imputation(
  data,
  subject_col,
  time_col,
  response_col,
  k,
  impute_method = "pmm",
  m = 5
)
```

Arguments

data	A data frame containing the longitudinal data.
subject_col	Character. The name of the column representing subject IDs.
time_col	Character. The name of the column representing time points.
response_col	Character. The name of the column representing the response variable.
k	Numeric. The number of clusters for the 'tvm' function.
impute_method	Character. The imputation method to be used in 'mice' (default is "pmm").
m	Numeric. The number of multiple imputations to be performed (default is 5).

Value

A data frame with imputed values for the influential response points while maintaining original NA values.

Examples

```
infsdata <- infsdata[1:5,]
imptvm <- tvm.imputation(infsdata, "subject_id", "time", "response", k = 3)
head(imptvm)
```

wlm

Weighted Longitudinal Mean (WLM)

Description

This function identifies influential subjects in a longitudinal dataset based on their weighted mean response values. It computes weighted averages for each subject and detects anomalies by comparing them against an overall mean threshold.

Usage

```
wlm(data, subject_id, time, response, k = 2, verbose = FALSE)
```

Arguments

data	A dataframe containing longitudinal data.
subject_id	A column specifying the column name representing subject IDs.
time	A column specifying different time points that observations are measured.
response	A column specifying the column name representing the response variable.
k	A numeric value specifying the threshold multiplier for detecting influential subjects (default: 2).
verbose	Logical; if TRUE, prints informative messages during execution.

Details

The function follows these steps:

- Computes the weighted mean response for each subject.
- Calculates the overall mean and standard deviation of weighted responses.
- Identifies subjects whose weighted mean response deviates beyond k standard deviations.
- Separates data into influential and non-influential subjects.
- Provides visualizations of the detected anomalies using `ggplot2`.

This method is beneficial for detecting influential subjects in longitudinal studies, where responses may vary over time and require weighted adjustments.

Value

A list containing:

<code>influential_subjects</code>	A vector of subject IDs identified as influential.
<code>influential_data</code>	A dataframe of influential subjects' data.
<code>non_influential_data</code>	A dataframe of non-influential subjects' data.
<code>weighted_plot</code>	A <code>ggplot</code> object showing the weighted mean response for each subject.
<code>longitudinal_plot</code>	A <code>ggplot</code> object visualizing the longitudinal data with influential subjects highlighted.
<code>IS_table</code>	A data frame containing the Influence Score (IS) and the Partial Influence Score (PIS) values for each subject at each time point.

See Also

`tvm`, `slm`, `sld`, `rld`

Examples

```
data(infsdata)
infsdata <- infsdata[1:5,]
result <- wlm(infsdata, "subject_id", "time", "response", k = 2)
print(result$influential_subjects)
head(result$influential_data)
head(result$non_influential_data)
print(result$weighted_plot)
print(result$longitudinal_plot)
```

Index

* datasets

infsdata, [2](#)

Phenobarb, [2](#)

infsdata, [2](#)

Phenobarb, [2](#)

rld, [3](#)

sld, [4](#)

slm, [6](#)

tvm, [8](#)

tvm.imputation, [9](#)

wlm, [10](#)