Package 'rankSummaries'

July 13, 2015

Type Package

Title Rank Summary Analysis for Longitudinal Data

Version 1.0.0

Date 2015-07-13

Author Clifford Anderson-Bergman

Depends

Maintainer Clifford Anderson-Bergman <pistacliffcho@gmail.com>

Description Package for rank summary analysis of longitudinal data. Created for the analysis of Morris Water Maze data.

License GPL-2

R topics documented:

`MWM_data` *Morris Water Maze Data*

Description

Morris Water Maze data collected from 3 different experiments. Latency represents time until mouse finds the hidden platform. Mice are removed from maze if they fail to find the platform in 90 seconds. Each mouse was trained 12 times.

Usage

```
data(MWM_data)
```

Format

A data frame with 106 rows and 14 variables

- `hAPP` Mouse Genotype
- `cohort` Cohort indicator
- `LATENCY_X` Latency for trial number `X` (ranges from 1-12)

Examples

```
data(MWM_data)
outcomeNames <- paste0('LATENCY_', 1:12)    #creating names of outcomes in data
fit <- rankAnalysis(formula = ~ hAPP + cohort,
                    data = MWM_data,
                    orderedScoreNames = outcomeNames,
                    w = 1:12)

summary(fit)
```

`rankAnalysis` *Analysis of Longitudinal Data by Rank Summaries*

Description

Fits a rank summary model for longitudinal data.

Usage

```
rankAnalysis(formula, data, orderedScoreNames, w = NULL)
```

Arguments

`formula`	formula for fitting rankScores. See details.
`data`	dataset
`orderedScoreNames`	ordered names of longitudinal outcomes
`w`	weights to use. If left = NULL, flat weights are used

Details

In regards to the `formula` argument, this should be a one sided argument, such as `~ group + gender`, in which we would modeling both `group` and `gender`. Formula *can* be a two sided formula, i.e. `y ~ group + gender`, but the response variable will be ignored (replaced with rank scores).

Author(s)

Clifford Anderson-Bergman

Examples

```
data(MWM_data)
outcomeNames <- paste0('LATENCY_', 1:12)    #creating names of outcomes in data
fit <- rankAnalysis(formula = ~ hAPP + cohort,
                    data = MWM_data,
                    orderedScoreNames = outcomeNames,
                    w = 1:12)

summary(fit)
```

Index