Package ‘nemtr’

January 18, 2023

Title  Nonparametric Extended Median Test - Cumulative Summation Method
Version  0.0.1.0
Description  Calculates a cumulative summation nonparametric extended median test based on the work of Brown & Schaffer (2020) <DOI:10.1080/03610926.2020.1738492>. It then generates a control chart to assess processes and determine if any streams are out of control.
License  MIT + file LICENSE
Encoding  UTF-8
RoxygenNote  7.2.1
URL  https://github.com/calebgreski/nemtr
BugReports  https://github.com/calebgreski/nemtr/issues
Imports  magrittr, tidyr, dplyr, ggplot2
Suggests  testthat
Depends  R (>= 3.50)
Config/testthat/edition  3
NeedsCompilation  no
Author  Caleb Greski [aut, cre],
        Austin Brown [aut]
Maintainer  Caleb Greski <caleb.p@greski.com>
Repository  CRAN
Date/Publication  2023-01-18 12:10:02 UTC

R topics documented:

  dataRead .......................................................... 2
  nemtr ............................................................. 3

Index  5
dataRead  

Read and Validate Dataframe

Description

Read in data and validate before analysis is conducted

Usage

dataRead(
  dataFrame,  
  timing,  
  streams,  
  VoI = NA,  
  type = "long",  
  median0 = NA,  
  delta = 3
)

Arguments

dataFrame     A user inputted dataframe, can be wide or long  
timing        A string of the timing variable name  
streams       A string of the streams variable name  
VoI           A string of the Variable of Interest name  
type          A string of the type of data (default long)  
median0       A value for expected median  
delta         A value for delta (default 3)

Value

A validated dataframe in long format

Examples

set.seed(795014178)
streams <- 20
time <- 60
samples <- 15
mu0 <- 3
delta <- 3
library(dplyr)

turnstiles <- tibble(
  turnstile = rep(rep(1:streams,each=samples),time),  
  hour = rep(1:time,each=streams * samples),  
  sample = rep(rep(1:samples), times = streams * time),
waitTime = rexp(streams * time * samples, rate = .22985)
    ) %>% mutate(waitTime = if_else(hour == 38, waitTime * 2, waitTime))
dataRead(turnstiles, timing = "hour", streams = "sample", VoI = "waitTime", type = "long", median0 = 3)

---

**nemtr**

*Nonparametric Extended Median Test*

**Description**

Take a dataframe, validate it, and then conduct the Nonparametric Extended Median Test to generate and display a control chart

**Usage**

nemtr(
  dataFrame,
  timing,
  streams,
  VoI = NA,
  type = "long",
  median0 = NA,
  delta = 3
)

**Arguments**

- `dataFrame`: A user inputted dataframe, can be wide or long
- `timing`: A string of the timing variable name
- `streams`: A string of the streams variable name
- `VoI`: A string of the Variable of Interest name
- `type`: A string of the type of data (default long)
- `median0`: A value for expected median
- `delta`: A value for delta (default 3)

**Value**

A validated dataframe in long format

**Examples**

```r
set.seed(795014178)
streams <- 20
time <- 60
samples <- 15
mu0 <- 3
```
 delta <- 3
library(dplyr)
turnstiles <- tibble(
  turnstile = rep(rep(1:streams, each=samples), time),
  hour = rep(1:time, each=streams * samples),
  sample = rep(rep(1:samples), times = streams * time),
  waitTime = rexp(streams * time * samples, rate = .22985)
) %>% mutate(waitTime = ifElse(hour == 38, waitTime * 2, waitTime))
nemtr(turnstiles, timing="hour", streams="sample", VoI="waitTime", type="long", median0 = 3)
Index

dataRead, 2

nemtr, 3