Package: TailClassifier (via r-universe)

August 25, 2024

Type Package

Title Tail Classifier

Version 0.1.1

Author Jialin Zhang (JZ)

Maintainer Jialin Zhang (JZ) <jzhang@math.msstate.edu>

Description The function TailClassifier() suggests one of the following types of tail for your discrete data: 1) Power decaying tail; 2) Sub-exponential decaying tail; and 3) Near-exponential decaying tail. The function also provides an estimate of the parameter for the classified-distribution as a reference.

License GPL-3

Encoding UTF-8

RoxygenNote 7.2.3

Imports ggplot2, cowplot, scales, stats, utils

NeedsCompilation no

Repository CRAN

Date/Publication 2023-10-30 05:00:02 UTC

Contents

`TailClassifier` *Tail Classifier*

Description

The function TailClassifier() suggests one of the following types of tail for your discrete data: 1) Power decaying tail; 2) Sub-exponential decaying tail; and 3) Near-exponential decaying tail. The function also provides an estimate of the parameter for the classified-distribution as a reference.

Usage

```
TailClassifier(
  sample_frequencies,
  v_left = 20,
  v_right = min(floor(sum(sample_frequencies)/20),
    sum(sample_frequencies[sample_frequencies > 1]) - 1),
  plot_lower = v_left,
  plot_upper = v_right,
  Plot0_title = "Plot 0 of Heavy Tail Detection \n \n",
  Plot1_title = "Plot 1 of Heavy Tail Detection",
  Plot2_title = "Plot 2 of Heavy Tail Detection",
  Plot3_title = "Plot 3 of Heavy Tail Detection",
  C_Level = 0.95,
  ConfidenceBand = T,
  Plot_0_y_limit_lower_extend = 1.5,
  Plot_0_y_limit_upper_extend = 1.5,
  Plot_1_y_limit_lower_extend = 0.25,
  Plot_1_y_limit_upper_extend = 0.25,
  Plot_2_y_limit_lower_extend = 0.25,
  Plot_2_y_limit_upper_extend = 0.25,
  Plot_3_y_limit_lower_extend = 0.25,
  Plot_3_y_limit_upper_extend = 0.25
)
```

Arguments

`sample_frequencies`	The frequency counts for your discrete sample data.
`v_left`	The starting point of tail profile. 20 is recommended. A smaller v_left may lead to unreliable results. A larger v_left might be adopted if the sample size is extremely large.
`v_right`	The ending point of tail profile. Recommendation is no more than 100 regardless of sample size.
`plot_lower`	The lower range of v-axis.
`plot_upper`	The upper range of v-axis.
`Plot0_title`	The title for Plot0. The default is “Plot 0 of Heavy Tail Detection”.

`Plot1_title` The title for Plot1. The default is "Plot 1 of Heavy Tail Detection".

`Plot2_title` The title for Plot2. The default is "Plot 2 of Heavy Tail Detection".

`Plot3_title` The title for Plot3. The default is "Plot 3 of Heavy Tail Detection".

`C_Level` The confidence level of confidence intervals in results. The default is 0.95.

`ConfidenceBand` TRUE if a confidence band is requested. FALSE otherwise.

`Plot_0_y_limit_lower_extend`
Modify the y limit in Plot 0 to allow the confidence band to correctly display in different scenarios.

`Plot_0_y_limit_upper_extend`
Modify the y limit in Plot 1 to allow the confidence band to correctly display in different scenarios.

`Plot_1_y_limit_lower_extend`
Modify the y limit in Plot 2 to allow the confidence band to correctly display in different scenarios.

`Plot_1_y_limit_upper_extend`
Modify the y limit in Plot 3 to allow the confidence band to correctly display in different scenarios.

`Plot_2_y_limit_lower_extend`
Modify the y limit in Plot 0 to allow the confidence band to correctly display in different scenarios.

`Plot_2_y_limit_upper_extend`
Modify the y limit in Plot 1 to allow the confidence band to correctly display in different scenarios.

`Plot_3_y_limit_lower_extend`
Modify the y limit in Plot 2 to allow the confidence band to correctly display in different scenarios.

`Plot_3_y_limit_upper_extend`
Modify the y limit in Plot 3 to allow the confidence band to correctly display in different scenarios.

Value

A statement on the type of tail.

Examples

```
## Power Example
# Generate data from power decaying distribution with parameter 1.5
rpar <- function(n, a, xm = 1) {
  v <- runif(n)
  xm / v^(1.0/a)
}
dpar <- function(x, a, xm = 1){
return(a*xm^a/(x^(a+1)))
}
set.seed(2023)
data <- floor(rpar(1000, 0.5)) # lambda = 1.5
Result <- TailClassifier(table(data), plot_lower = 5, plot_upper = 400, v_left = 20, v_right = 54,
```

```
 Plot_0_y_limit_upper_extend = 8)
## display the results
Result
## call the classification decision
Result$Type
## call the confidence intervals for the parameters
data.frame(Result$Results[3])[,c(1,3:4)]
## call a specific plot
Result$Results[[1]][1]
Result$Results[[1]][2]
Result$Results[[1]][3]
Result$Results[[1]][4]
## check the rank of possible type of tails
Result$Rank
```

Index