Package ‘arena2r’

October 19, 2018

Type Package
Title Plots, Summary Statistics and Tools for Arena Simulation Users
Version 1.0.0
Author Pedro Nascimento de Lima [aut, cre]
Maintainer Pedro Nascimento de Lima <pedrolima.n@gmail.com>
Description Reads Arena <https://www.arenasimulation.com/> CSV output files and generates nice tables and plots. The package contains a Shiny App that can be used to interactively visualize Arena's results.
License MIT + file LICENSE
Encoding UTF-8
LazyData true
Imports stats, utils, rlang, tidyr, dplyr, purrr, shiny, shinydashboard, shinyBS, shinyjs, ggplot2
Depends R (>= 2.10)
Suggests testthat, covr, knitr, rmarkdown
RoxygenNote 6.1.0
URL https://github.com/pedroliman/arena2r
BugReports https://github.com/pedroliman/arena2r/issues
VignetteBuilder knitr
NeedsCompilation no
Repository CRAN
Date/Publication 2018-10-19 15:30:03 UTC

R topics documented:

arena_results ........................................ 2
get_simulation_results ................................. 2
get_statistics_summary ............................... 3
plot_box ............................................. 4
get_simulation_results

Description

This function reads all csv files inside the provided path and returns a data.frame with the simulation runs, consolidated. You should provide a path containing only csv files generated by Arena, with the same number of replications. I Suggest you to name your csv files after your scenarios.

Usage

get_simulation_results(source, source_type = "path")

Arguments

source The path where csv files is stored, or a list coming from shiny. If you do not provide a value, I’ll assume they’re on your current working directory.
source_type String that describes where the data is coming from. "path" stands for a path that contains all csv files. "shinyInput" stands for the list object returned by fileInput in the ShinyApp.
get_statistics_summary

Value

a tidy dataframe with simulation results.

Examples

# Define de path where your csv files are:
path <- system.file("extdata", package = "arena2r")
simulation_results = get_simulation_results(path)
head(simulation_results)

get_statistics_summary

Get Statistics Summary

Description

Makes a summary table to every statistic available

Usage

get_statistics_summary(sim_results, confidence = 0.95)

Arguments

sim_results The data.frame generated by get_simulation_results()
confidence The confidence of the CI

Value

a data.frame with a summary for every Statistic

Examples

library(arena2r)

statistics_summary = get_statistics_summary(arena_results)
head(statistics_summary)
plot_box  

**Box Plot**

**Description**

Plots a box plot for a response variable, across different simulated scenarios.

**Usage**

```r
plot_box(sim_results, response_variable)
```

**Arguments**

- `sim_results`  The data.frame generated by `get_simulation_results()`
- `response_variable`  A character string indicating the Statistic to be plotted.

**Value**

A box plot using ggplot2.

**Examples**

```r
library(arena2r)

plot_box(arena_results, "Entity 1.NumberOut")
```

---

plot_confint  

**Confidence Interval Plot**

**Description**

Plots the confidence interval for a response variable, across different simulated scenarios.

**Usage**

```r
plot_confint(sim_results, response_variable)
```

**Arguments**

- `sim_results`  The data.frame generated by `get_simulation_results()`
- `response_variable`  A character string indicating the Statistic to be plotted.
plot_scatter

Value

a confidence interval plot using ggplot2.

Examples

```r
library(arena2r)

plot_confint(arena_results, "Entity 1.WaitTime")
```

---

### Scatter Plot

**Description**
Scatter Plot

**Usage**

```r
plot_scatter(sim_results, x_variable, y_variable)
```

**Arguments**

- `sim_results`: The data.frame generated by `get_simulation_results()`
- `x_variable`: The name of the Statistic to be placed on the x axis
- `y_variable`: The name of the Statistic to be placed on the y axis

**Value**

a scatter plot showing individual replication results

**Examples**

```r
library(arena2r)

plot_scatter(arena_results, "Entity 1.NumberOut","Entity 1.WaitTime")
```

---

### Run Arena App

**Description**

This function will launch a Shiny App allowing you to analyse Arena results without writing R code.

**Usage**

```r
runArenaApp()
```
Index

*Topic **datasets**

  arena_results, 2

arena_results, 2

get_simulation_results, 2
get_statistics_summary, 3

plot_box, 4
plot_confint, 4
plot_scatter, 5

runArenaApp, 5