Package 'mephas'

February 4, 2019

Type Package

Title Shiny Application of Medical and Pharmaceutical Statistics

Description The web-based statistical fundamentals, tests, and models, aiming to facilitate researchers to analyze medical, pharmaceutical and genomic data.

Version 1.0.0

Maintainer Yi Zhou <zhou-y@phs.osaka-u.ac.jp>

Depends R (>= 3.5.0)

License MIT + file LICENSE

Imports DescTools, ROCR, Rmisc, ggplot2, ggfortify, gridExtra, pastecs, plotROC, psych, reshape, stargazer, shiny, survminer, survival, xtable, spls, pls

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

URL https://github.com/mephas/mephas

BugReports https://github.com/mephas/mephas/issues

NeedsCompilation no

Author Yi Zhou [cre, aut] (<https://orcid.org/0000-0001-9254-324>)

Repository CRAN

Date/Publication 2019-02-04 14:40:07 UTC

R topics documented:

`MFSanova` *MEPHAS Shiny Application of ANOVA and Multiple Comparison*

Description

MFSanova function creates a dynamic calculator which enables users to do ANOVA analysis and multiple comparison. Users can either input data manually or upload their dataset.

Usage

```
MFSanova()
```

Details

This app includes one-way ANOVA, two-way ANOVA, and multiple comparison. The results include ANOVA table, descriptive statistics, and bar-plot of the group's mean value. Please click "close" window to quit the application. "Rmisc" and "psych" packages are required.

Value

The shiny app and web page of the ANOVA and multiple comparison

Examples

```
# library(mephas)
# MFSanova()
```

MFSdistribution *MEPHAS Shiny application of the Statistical Distributions*

Description

MFSdistribution function creates a dynamic calculator which shows the commonly used probability distribution. Users can change the parameter to change the shape of distribution.

Usage

```
MFSdistribution()
```

Details

This app includes the distributions for continuous variables (normal distribution, exponential distribution, and gamma distribution), for discrete variables (binomial distribution and Poisson distribution), and distributions derived from normal distribution (t distribution, chi-square distribution, and F distribution). Please click "close" window to quit the application. No special packages are required.

Value

The shiny web page of the statistical distribution

Examples

```
# library(mephas)
# MFSdist()
```

MFSnptest *MEPHAS Shiny Application of the Non-paramatric Tests*

Description

MFSnptest function creates a dynamic calculator which enables users to do non-parametric statistical tests. Users can either input data manually or upload their dataset.

Usage

```
MFSnptest()
```

Details

This app includes non-parametric tests for one sample, two samples, and two paired samples. The results include box-plots, histogram, descriptive statistics,and test results. Please click "close" window to quit the application. "gridExtra", "reshape", "pastecs", "DescTools", and "RVAideMemoire" packages may be required.

Value

The shiny web page of non-parametric statistical test

Examples

```
# library(mephas)
# MFSn()
# not run
```

`MFSpcapls` *MEPHAS Shiny Application of PCA PLS Regression*

Description

MFSpcapls function creates a dynamic calculator which enables users to do PCA, PLS, sparse PLS regression. Users can either input data manually or upload their dataset.

Usage

```
MFSpcapls()
```

Details

This app includes one-way ANOVA, two-way ANOVA, and multiple comparison. The results include ANOVA table, descriptive statistics, and bar-plot of the group's mean value. Please click "close" window to quit the application. "ggfortify", spls" and "pls" package may be required.

Value

The shiny web page of the PCA PLS regression

Examples

```
# library(mephas)
# MFSpcapls()
# not run
```

MFSproptest *MEPHAS Shiny application of the binomial proportional data tests*

Description

MFSproptest function creates a dynamic calculator which enables users to do statistical tests on binomial proportional data. Users can either input data manually or upload their dataset.

Usage

```
MFSproptest()
```

Details

This app includes statistical tests for one single proportion, two proportions from independent groups, and two proportions from the matched data. Results include the corresponding test's outcome and pie-plot of data. Please click "close" window to quit the application. "gridExtra", "reshape", and "pastecs" packages are required.

Value

The shiny web page of the tests based on binomial proportion

Examples

```
# library(mephas)
# MFSproptest()
# not run
```

MFSrctabtest *MEPHAS Shiny Application of Contingency Table Related Tests*

Description

MFSrctabtest function creates a dynamic calculator which enables users to do statistical tests on R by C data. Users can either input data manually or upload their dataset.

Usage

```
MFSrctabtest()
```

Details

This app includes statistical tests for R by C table, trend test for 2 by K table, and goodness-of-fit test. Please click "close" window to quit the application. "psych" package is required.

Value

The shiny web page of the tests for cross tab data

Examples

```
# library(mephas)
# MFSrctabtest()
# not run
```

`MFSreg` *MEPHAS Shiny Application of Univariate Regression*

Description

MFSreg function creates a dynamic calculator which instructs users to do linear regression, logistic regression, and cox regression. Users can either input data manually or upload their dataset.

Usage

```
MFSreg()
```

Details

This app includes linear regression, logistic regression, and cox regression. The results include ANOVA table, descriptive statistics, and residual plots. Please click "close" window to quit the application. "survival", "survminer", and "ggfortify" packages may be required.

Value

The shiny web page of the tests univariate regression

Examples

```
# library(mephas)
# MFSreg()
# not run
```

MFSttest *MEPHAS Shiny Application of T test.*

Description

MFSttest function creates a dynamic calculator which enables users to do t tests . Users can either input data manually or upload their dataset.

Usage

```
MFSttest()
```

Details

This app includes t tests for one sample, two samples, and two paired samples. The results include box-plots, histogram, descriptive statistics,and test results. "gridExtra", "reshape", "pastecs" package may be required.

Value

The shiny application web page of T test for one sample and two samples

Examples

```
# library(mephas)
# MFSttest()
# not run
```

nptest.mmtest *UI of Mood's Median Test (Non-paramatric Tests)*

Description

UI of Mood's Median Test (Non-paramatric Tests)

Usage

```
nptest.mmtest()
```

nptest.onesample *UI of One Sample (Non-paramatric Tests)*

Description

UI of One Sample (Non-paramatric Tests)

Usage

```
nptest.onesample()
```

nptest.psample *UI of Paired Sample (Non-paramatric Tests)*

Description

UI of Paired Sample (Non-paramatric Tests)

Usage

```
nptest.psample()
```

nptest.signtest *UI of Sign Test (Non-paramatric Tests)*

Description

UI of Sign Test (Non-paramatric Tests)

Usage

```
nptest.signtest()
```

nptest.signtest.p *UI of Sign Test (Non-paramatric Tests)*

Description

UI of Sign Test (Non-paramatric Tests)

Usage

```
nptest.signtest.p()
```

nptest.twosample *UI of Two Sample (Non-paramatric Tests)*

Description

UI of Two Sample (Non-paramatric Tests)

Usage

```
nptest.twosample()
```

nptest.wrtest *UI of Wilcoxon Rank-Sum Test (Non-paramatric Tests)*

Description

UI of Wilcoxon Rank-Sum Test (Non-paramatric Tests)

Usage

```
nptest.wrtest()
```

nptest.wstest *UI of Wilcoxon Signed-Rank Test (Non-paramatric Tests)*

Description

UI of Wilcoxon Signed-Rank Test (Non-paramatric Tests)

Usage

```
nptest.wstest()
```

nptest.wstest.p *UI of Wilcoxon Signed-Rank Test (Non-paramatric Tests)*

Description

UI of Wilcoxon Signed-Rank Test (Non-paramatric Tests)

Usage

```
nptest.wstest.p()
```

pls.data.ui *MEPHAS Shiny Application of PCA PLS Regression*

Description

MEPHAS Shiny Application of PCA PLS Regression

Usage

```
pls.data.ui()
```

pls.ui *MEPHAS Shiny Application of PCA PLS Regression*

Description

MEPHAS Shiny Application of PCA PLS Regression

Usage

```
pls.ui()
```

reg.cr.ui *UI of Cox Regression (univariate Regression)*

Description

UI of Cox Regression (univariate Regression)

Usage

```
reg.cr.ui()
```

reg.data.ui *UI of Logistic Regression (univariate Regression)*

Description

UI of Logistic Regression (univariate Regression)

Usage

```
reg.data.ui()
```

reg.lm.ui *UI of Linear Regression (univariate Regression)*

Description

UI of Linear Regression (univariate Regression)

Usage

```
reg.lm.ui()
```

reg.lr.ui *UI of Logistic Regression (univariate Regression)*

Description

UI of Logistic Regression (univariate Regression)

Usage

```
reg.lr.ui()
```

Index