Package ‘JustifyAlpha’

September 15, 2021

Title Justifying Alpha Levels for Hypothesis Tests
Version 0.1.1
Description Functions to justify alpha levels for statistical hypothesis tests by avoiding Lindley’s paradox, or by minimizing or balancing error rates. For more information about the package please read the following: Maier & Lakens (2021) <doi:10.31234/osf.io/ts4r6>.
License MIT + file LICENSE
Encoding UTF-8
RoxygenNote 7.1.1
Imports stats, BayesFactor, ggplot2, stringr, shiny, Superpower, pwr, shinydashboard, qpdf
Suggests knitr, rmarkdown, testthat,
VignetteBuilder knitr
NeedsCompilation no
Author Maximilian Maier [aut, cre], Daniel Lakens [aut] (<https://orcid.org/0000-0002-0247-239X>)
Maintainer Maximilian Maier <maximilianmaier0401@gmail.com>
Repository CRAN
Date/Publication 2021-09-15 08:00:08 UTC

R topics documented:

ftestEvidence ........................................................... 2
optimal_alpha .......................................................... 3
optimal_sample ......................................................... 4
runApp ................................................................. 5
ttestEvidence ........................................................ 5

Index 7
ftestEvidence

Justify your alpha level by avoiding the Lindley paradox or aiming for moderate or strong evidence when using anova.

Description

Justify your alpha level by avoiding the Lindley paradox or aiming for moderate or strong evidence when using anova.

Usage

ftestEvidence(evidence, df1, df2, paired = FALSE, printplot = FALSE)

Arguments

evidence Desired level of evidence: "Lindley" to avoid the Lindley Paradox, "moderate" to achieve moderate evidence and "strong" to achieve strong evidence. Users that are more familiar with Bayesian statistics can also directly enter their desired Bayes factor.
df1 Numerator degrees of freedom.
df2 Denominator degrees of freedom.
paired If true a within subjects design is assumed.
printplot If true prints a plot relating Bayes factors and p-values.

Value

numeric alpha level required to avoid Lindley's paradox.

References

Maier & Lakens (2021). Justify Your Alpha: A Primer on Two Practical Approaches

Examples

## Avoid the Lindley paradox for an anova with 1 numerator and 248 denominator degrees of freedom.
ftestEvidence("lindley", 1, 248)
optimal_alpha

Justify your alpha level by minimizing or balancing Type 1 and Type 2 error rates.

Usage

```r
optimal_alpha(
  power_function,
  costT1T2 = 1,
  priorH1H0 = 1,
  error = "minimize",
  verbose = FALSE,
  printplot = FALSE
)
```

Arguments

- `power_function`: Function that outputs the power, calculated with an analytic function.
- `costT1T2`: Relative cost of Type 1 errors vs. Type 2 errors.
- `priorH1H0`: How much more likely a-priori is H1 than H0?
- `error`: Either "minimize" to minimize error rates, or "balance" to balance error rates.
- `verbose`: Print each iteration of the optimization function if TRUE. Defaults to FALSE.
- `printplot`: Print a plot to illustrate the alpha level calculation.

Value

Returns a list of the following:
- `alpha`: alpha or Type 1 error that minimizes or balances combined error rates.
- `beta`: beta or Type 2 error that minimizes or balances combined error rates.
- `errorrate`: weighted combined error rate.
- `objective`: value that is the result of the minimization, either 0 (for balance) or the combined weighted error rates.
- `plot_data`: data used for plotting (only if printplot = TRUE)
- `plot`: plot of error rates depending on alpha (only if printplot = TRUE)

References

Maier & Lakens (2021). Justify Your Alpha: A Primer on Two Practical Approaches

Examples

```r
## Optimize power for a independent t-test, smallest effect of interest
## d = 0.5, 100 participants per condition
res <- optimal_alpha(power_function = "pwr::pwr.t.test(d = 0.5, n = 100,
  sig.level = x, type = 'two.sample', alternative = 'two.sided')$power")
res$alpha
```
optimal_sample

Justify your alpha level by minimizing or balancing Type 1 and Type 2 error rates.

Usage

optimal_sample(
  power_function,
  errorgoal = 0.05,
  costT1T2 = 1,
  priorH1H0 = 1,
  error = "minimize",
  printplot = FALSE
)

Arguments

power_function Function that outputs the power, calculated with an analytic function.
errorgoal Desired weighted combined error rate
costT1T2 Relative cost of Type 1 errors vs. Type 2 errors.
priorH1H0 How much more likely a-priori is H1 than H0?
error Either "minimize" to minimize error rates, or "balance" to balance error rates.
printplot Print a plot to illustrate the alpha level calculation. This will make the function considerably slower.

Value

Returns a list of the following alpha = alpha or Type 1 error that minimizes or balances combined error rates, beta = beta or Type 2 error that minimizes or balances combined error rates, errorrate = weighted combined error rate, objective = value that is the result of the minimization, either 0 (for balance) or the combined weighted error rates, samplesize = the desired samplesize. plot = plot of alpha, beta, and error rate as a function of samplesize (only if printplot = TRUE)

References

Maier & Lakens (2021). Justify Your Alpha: A Primer on Two Practical Approaches
Examples

```r
# Optimize power for an independent t-test, smallest effect of interest
# d = 0.5, desired weighted combined error rate = 5%
res <- optimal_sample(power_function = "pwr::pwr.t.test(d = 0.5, n = sample_n, sig.level = x,
  type = 'two.sample', alternative = 'two.sided')$power",errorgoal = 0.05)
res$alpha
res$beta
res$errorrate
res$samplesize
```

runApp

Launch the Justify your alpha shiny app.

Description

Launch the Justify your alpha shiny app.

Usage

```r
runApp()
```

ttestEvidence

Justify your alpha level by avoiding the Lindley paradox or aiming for moderate or strong evidence when using a t-test.

Description

Justify your alpha level by avoiding the Lindley paradox or aiming for moderate or strong evidence when using a t-test.

Usage

```r
ttestEvidence(
  evidence, 
  n1,
  n2 = 0,
  one.sided = FALSE,
  rscale = sqrt(2)/2,
  printplot = FALSE
)
```
Arguments

evidence  Desired level of evidence: "Lindley" to avoid the Lindley Paradox, "moderate" to achieve moderate evidence and "strong" to achieve strong evidence. Users that are more familiar with Bayesian statistics can also directly enter their desired Bayes factor.

n1  Sample size in Group 1.
n2  Sample size in Group 2. Leave blank for a one-sample or paired-sample
one.sided  Indicates whether the test is one sided or two sided.
rscale  Scale of the Cauchy prior
printplot  If true prints a plot relating Bayes factors and p-values.

Value

numeric alpha level required to avoid Lindley's paradox.

References

Maier & Lakens (2021). Justify Your Alpha: A Primer on Two Practical Approaches

Examples

```r
## Avoid the Lindley paradox for a two sample t-test with 300 participants per condition
ttestEvidence("lindley", 300, 300)
```
Index

fTestEvidence, 2
optimal_alpha, 3
optimal_sample, 4
runApp, 5
ttestEvidence, 5