Package ‘ANOVAreplication’

October 12, 2022

Type Package

Version 1.1.5

Title Test ANOVA Replications by Means of the Prior Predictive p-Value

Description Allows for the computation of a prior predictive p-value to test replication of relevant features of original ANOVA studies. Relevant features are captured in informative hypotheses. The package also allows for the computation of sample sizes for new studies, post-hoc power calculations, and comes with a Shiny application in which all calculations can be conducted as well. The statistical underpinnings are described in Zondervan-Zwijnenburg (2019) <doi:10.31234/osf.io/6myqh>.

License GPL (>= 3)

Depends quadprog, graphics, grDevices, shiny, stats, utils

Suggests knitr, rmarkdown, testthat

VignetteBuilder knitr

Encoding UTF-8

URL `https://dx.doi.org/10.17605/OSF.IO/6H8X3`

NeedsCompilation no

Author M.A.J. Zondervan-Zwijnenburg [aut, cre],
Herbert Hoijtink [ths],
Caspar J van Lissa [ctb],
Consortium Individual Development [fnd] (NWO grant number 024.001.003)

Maintainer M.A.J. Zondervan-Zwijnenburg <`m.a.j.zwijnenburg@uu.nl`>

Repository CRAN

Date/Publication 2021-09-27 08:10:02 UTC

R topics documented:

`Fbar.dif` *F-bar for inequality constraints with minimum differences between means*

Description

The function calculates F-bar for inequality constrained hypotheses with minimum differences between means (Type B). See Silvapulle & Sen (2011) for background on the F-bar statistic. The code of Vanbrabant (2017) is the basis to this Fbar function.

Usage

```
Fbar.dif(data,Amat,difmin,effectsize=FALSE)
```

Arguments

`data`	A dataframe with two variables: (1) a dependent variable, and (2) a grouping variable. Groups are labeled consecutively, starting at 1.
`Amat`	A p by q matrix, where p is the number of means in the ANOVA model, and q is the number of constraints to be imposed on the model. Each row represents one constraint where the parameter with the lower value according to the constraint receives the value -1, and the parameter with the higher value according to the constraint receives the value 1. Other parameters within the same row obtain the value 0.
`difmin`	A vector of length q with the minimum difference per constraint as specified in `Amat`.
`effectsize`	Logical; If TRUE the values in `difmin` are interpreted as Cohen's d.

Value

The value for the F-bar statistic

Author(s)

M. A. J. Zondervan-Zwijnenburg

References

Silvapulle, M. J., & Sen, P. K. (2011). Constrained statistical inference: Order, inequality, and shape constraints (Vol. 912). John Wiley & Sons. doi: 10.1002/9781118165614.ch1

Vanbrabant, L. (2017). restriktor: Restricted Statistical Estimation and Inference for Linear Models. R package version 0.1-55. https://CRAN.R-project.org/package=restriktor

See Also

See also `runShiny`, `Fbar.ineq`, and `Fbar.exact`.

Examples

```
data <- data.frame(y=ChickWeight$weight,g=ChickWeight$Diet)
aggregate(data$y,by=list(data$g),mean)


#create matrices HR: g4>(g1,g2,g3). g4-g1>0.8, g4-g2>0.5, g4-g3>0.2
HR <- create_matrices(varnames = c("g1","g2","g3","g4"),
                      hyp = "g4-g1>0.8 & g4-g2>0.5 & g4-g3>0.2")

Fbar.dif(data,Amat=HR$Amat,difmin=HR$difmin,effectsize=TRUE)

#make Amat with constraints: 1<4,2<4,3<4 (last constraint is not true)
Amat <- (rbind(c(-1,0,0,1),c(0,-1,0,1),c(0,0,-1,1)))
#minimal differences for each constraint
difmin=c(30,15,1)

Fbar.dif(data,Amat,difmin=difmin)
```

`Fbar.exact` *F-bar for exact constraints*

Description

The function calculates F-bar for hypotheses constrained with exact values (Type B). See Silvapulle & Sen (2011) for background on the F-bar statistic. The code of Vanbrabant (2017) is the basis to this Fbar function.

Usage

```
Fbar.exact(data,exact)
```

Arguments

`data`	A dataframe with two variables: (1) a dependent variable, and (2) a grouping variable.
`exact`	A vector of length p, where p is the number of means in the ANOVA model, with the exact values of the constrained hypothesis.

Value

The value for the F-bar statistic

Author(s)

M. A. J. Zondervan-Zwijnenburg

References

Silvapulle, M. J., & Sen, P. K. (2011). Constrained statistical inference: Order, inequality, and shape constraints (Vol. 912). John Wiley & Sons. doi: 10.1002/9781118165614.ch1

Vanbrabant, L. (2017). restriktor: Restricted Statistical Estimation and Inference for Linear Models. R package version 0.1-55. https://CRAN.R-project.org/package=restriktor

See Also

See also `runShiny`, `Fbar.ineq`, and `Fbar.dif`.

Examples

```
data <- data.frame(y=ChickWeight$weight,g=ChickWeight$Diet)
aggregate(data$y,by=list(data$g),mean)

#specify exact values to be evaluated. Hi: mu1=102,mu2=123,mu3=143,mu4=135.
exact <- c(102,123,143,135)

Fbar.exact(data,exact)
```

`Fbar.ineq` *F-bar for inequality constraints*

Description

The function calculates F-bar for inequality constrained hypotheses (Type B). See Silvapulle & Sen (2011) for background on the F-bar statistic. The code of Vanbrabant (2017) is the basis to this Fbar function.

Usage

```
Fbar.ineq(data,Amat)
```

Arguments

`data`	A dataframe with two variables: (1) a dependent variable, and (2) a grouping variable.
`Amat`	A p by q matrix, where p is the number of means in the ANOVA model, and q is the number of constraints to be imposed on the model. Each row represents one constraint where the parameter with the lower value according to the constraint receives the value -1, and the parameter with the higher value according to the constraint receives the value 1. Other parameters within the same row obtain the value 0.

Value

The value for the F-bar statistic.

Author(s)

M. A. J. Zondervan-Zwijnenburg

References

Silvapulle, M. J., & Sen, P. K. (2011). Constrained statistical inference: Order, inequality, and shape constraints (Vol. 912). John Wiley & Sons. doi: 10.1002/9781118165614.ch1

Vanbrabant, L. (2017). restriktor: Restricted Statistical Estimation and Inference for Linear Models. R package version 0.1-55. https://CRAN.R-project.org/package=restriktor

See Also

See also `runShiny`, `Fbar.dif`, and `Fbar.exact`.

Examples

```
data <- data.frame(y=ChickWeight$weight,g=ChickWeight$Diet)
aggregate(data$y,by=list(data$g),mean)

#create matrices HR: g4>(g1,g2,g3)
HR <- create_matrices(varnames = c("g1","g2","g3","g4"),
                      hyp = "g4>(g1,g2,g3)")
Fbar.ineq(data,Amat=HR$Amat)
```

`generate.data` *Generate Data*

Description

Generates data of sample size n with the exact specified mean and standard deviation.

Usage

```
generate.data(n,mean,sd)
```

Arguments

`n`	integer; sample size.
`mean`	integer; mean of the data.
`sd`	integer; standard deviation of the data.

Value

Generates a random value vector of length n with the specified mean and standard deviation.

Examples

```
#simple example
data1 <- generate.data(n=10,mean=5,sd=1)

#multiple independent groups
means = c(0.36,-0.19,-0.18)
sds = c(1.08,0.53,0.81)
n = c(28,28,28) #N = 84
y <- list(NA)
for (i in 1:3){y[[i]] <- generate.data(n[i],means[i],sds[i])}
y <- unlist(y)
group <- c(rep(0,n[1]),rep(1,n[2]),rep(2,n[3]))
p <- length(unique(group))
data=data.frame(y=y,g=group)
aggregate(data$y,by=list(data$g),mean)
```

`Gibbs.ANOVA` *Gibbs sampler*

Description

Samples from the posterior distribution of the data by means of a Gibbs sampler (derived from Lynch, 2007, p. 170-172).

Usage

```
Gibbs.ANOVA(data,it=5000,burnin=500,seed=0)
```

Arguments

`data`	a data frame with a variable y and a variable g, where y is the dependent variable, and g is the grouping variable for the ANOVA. Groups are labeled consecutively, starting at 1.
`it`	the number of (post-burnin) iterations for each of the two chains. The default uses 5.000 iterations.
`burnin`	the number of iterations for the function to use for the burnin phase in each of the two chains. The default uses 500 burnin iterations.
`seed`	integer; seed value. If seed==0, no seed is set.

Value

`posterior`	a matrix with all samples from the posterior for each parameter.

Returns a matrix with the mean, median, and standard deviation (in columns) for the it samples from the conditional posterior distributions of the group means and pooled standard deviation (rows).

Produces traceplots of each parameter and the associated samples from the posterior distribution.

Author(s)

M. A. J. Zondervan-Zwijnenburg

References

Lynch, S. (2007). Introduction to applied Bayesian statistics and estimation for social scientists. New York, NY: Springer. doi: 10.1007/978-0-387-71265-9

Zondervan-Zwijnenburg, M.A.J., Van de Schoot, R., & Hoijtink, H. (2017). Testing ANOVA replication by means of the prior predictive p-value.

Examples

```
data <- data.frame(y=ChickWeight$weight,g=ChickWeight$Diet)
Gibbs.ANOVA(data)
```

`pooled.sd` *Pooled standard deviation calculator*

Description

Calculates the pooled standard deviation.

Usage

```
pooled.sd(data)
```

Arguments

`data`	A dataframe with two variables: the dependent variable in the first column, and the grouping variable in the second column.

Value

Returns the pooled standard deviation.

Author(s)

M.A.J. Zondervan-Zwijnenburg

Examples

```
data <- data.frame(y=ChickWeight$weight,g=ChickWeight$Diet)
pooled.sd(data)
```

`power.calc` *Power Calculator*

Description

Calculates the power for the prior predictive check

Usage

```
power.calc(n.r,posterior,g.m,p.sd,
           statistic,Amat=0L,exact=0L,difmin=0L,effectsize=FALSE,
           alpha=.05)
```

Arguments

`n.r`	vector with the sample size per group (i.e., n_jr) for new study (i.e., y_r).
`posterior`	matrix (e.g., the output of Gibbs.ANOVA) with samples from the posterior based on the original data (i.e., y_o).
`g.m`	vector; the population values the alternative distribution. To calculate the power to reject replication if the means are equal specify the grand mean of the study variables in the original dataset.
`p.sd`	integer; the population value for the pooled standard deviation in the alternative distribution. We advice to specify the pooled standard deviation for the study variables in the original dataset.
`statistic`	the type of hypothesis to be evaluated: "ineq" for inequality constrained means, "dif" for inequality constraints plus minimum differences between means, "exact" for specific values for the means.

`Amat`	p by q matrix, where p is the number of means in the ANOVA model, and q is the number of constraints to be imposed on the model. Each row represents one constraint where the parameter with the lower value according to the constraint receives the value -1, and the parameter with the higher value according to the constraint receives the value 1. Other parameters within the same row obtain the value 0. The create_matrices function can be used to obtain Amat.
`exact`	vector of length p, where p is the number of means in the ANOVA model, with the exact values of the constrained hypothesis.
`difmin`	vector of length q with the minimum difference per constraint as specified in `Amat`. The create_matrices function can be used to obtain difmin. Default = 0L, indicating that only the ordering of means is evaluated.
`effectsize`	logical; If TRUE the values in `difmin` are interpreted as Cohen's d.
`alpha`	integer; the level of alpha that should be taken into account while calculating the required sample size.

Value

`power`	The acquired power given the input
`rejection.value`	The 1-alpha'th percentile of the null distribution. The proportion of H1 larger than this value constitues power.

Author(s)

M. A. J. Zondervan-Zwijnenburg, H. Hoijtink

References

Zondervan-Zwijnenburg, M.A.J., Van de Schoot, R., & Hoijtink, H. (2017). Testing ANOVA replication by means of the prior predictive p-value.

See Also

See also `runShiny`, `Gibbs.ANOVA`, `Fbar.ineq`, `Fbar.dif`, and `Fbar.exact`, `create_matrices`, `prior.predictive.check`, `sample.size.calc`.

Examples

```
#analysis original data
data_o <- data.frame(y=ChickWeight$weight,g=ChickWeight$Diet)
g.m <- rep(mean(data_o$y),3)
#compute pooled sd
sd.g <- aggregate(data_o$y,by=list(data_o$g),sd)[,2]
n.g <- table(data_o$g)
p.sd<- pooled.sd(data_o)

means <- aggregate(data_o$y,by=list(data_o$g),mean)[,2]
```

```
post <- Gibbs.ANOVA(data_o)

power.calc(n.r=c(20,21,22,23),posterior=post$posterior,g.m=g.m,p.sd=p.sd,
          statistic="exact",exact=means,alpha=.05)
```

`prior.predictive.check` *Prior predictive check*

Description

Uses the prior predictive check to test replication for ANOVA models.

Usage

```
prior.predictive.check(n,posterior,statistic,obs=TRUE,F_n,
                       Amat=0L,exact=0L,difmin=0L,effectsize=FALSE,seed=0)
```

Arguments

`n`	vector with the sample size per group (i.e., n_jr) for new study (i.e., y_r).
`posterior`	a matrix (e.g., the output of Gibbs.ANOVA) with samples from the posterior based on the original data (i.e., y_o).
`statistic`	the type of hypothesis to be evaluated: "ineq" for inequality constrained means, "dif" for inequality constraints plus minimum differences between means, "exact" for specific values for the means.
`obs`	logic; If FALSE, the prior predictive check does not calculate a p-value, because no observed statistic is provided. Used by the sample.size.calculator function.
`F_n`	The Fbar value for the new data.
`Amat`	a p by q matrix, where p is the number of means in the ANOVA model, and q is the number of constraints to be imposed on the model. Each row represents one constraint where the parameter with the lower value according to the constraint receives the value -1, and the parameter with the higher value according to the constraint receives the value 1. Other parameters within the same row obtain the value 0.
`exact`	a vector of length p, where p is the number of means in the ANOVA model, with the exact values of the constrained hypothesis.
`difmin`	a vector of length q with the minimum difference per constraint as specified in `Amat`.
`effectsize`	logical; If TRUE the values in `difmin` are interpreted as Cohen's d.
`seed`	integer; seed value. If seed==0, no seed is set.

Value

Generates a histogram of F_sim in which F_n is indicated with a vertical line. The proportion of F_sim at the right of this line constitutes the prior predictive p-value.

`sumFdist`	a summary of F_sim
`ppp`	the prior predictive p-value
`F_sim`	a vector with F-bar values for all simulated datasets

Author(s)

M. A. J. Zondervan-Zwijnenburg

References

Zondervan-Zwijnenburg, M.A.J., Van de Schoot, R., & Hoijtink, H. (2017). Testing ANOVA replication by means of the prior predictive p-value.

See Also

See also `runShiny`, `Gibbs.ANOVA`, `Fbar.ineq`, `Fbar.dif`, and `Fbar.exact`, `sample.size.calc`, `power.calc`.

Examples

```
#analysis original data
data_o <- data.frame(y=ChickWeight$weight,g=ChickWeight$Diet)
post <- Gibbs.ANOVA(data_o)

#analysis new data
data_r <- data.frame(y=rnorm(660, mean(data_o$y), sd=sd(data_o$y)),g=round(runif(660,1,4)))
n.r = as.numeric(table(data_r$g))

#create matrices HR: g4>(g1,g2,g3). g4-g1>0.8, g4-g2>0.5, g4-g3>0.2
HR <- create_matrices(varnames = c("g1","g2","g3","g4"),
                      hyp = "g4-g1>0.8 & g4-g2>0.5 & g4-g3>0.2")

Amat <- HR$Amat
difmin <- HR$difmin
r.F.dif.efsz <- Fbar.dif(data_r,Amat,difmin,effectsize=TRUE)

#prior predictive check
result <- prior.predictive.check(n=n.r,posterior=post$posterior,F_n=r.F.dif.efsz,statistic="dif",
effectsize=TRUE,Amat=Amat,difmin=difmin,seed=1)

result$sumFdist #summary of the f(F_y_sim)
result$ppp      #the prior predictive p-value
```

`runShiny` *function to launch Shiny application to test replication of ANOVA results*

Description

Launches a Shiny application for the replication test.

Usage

```
runShiny()
```

Value

In the Shiny application Gibbs.ANOVA can be ran, which prints covergence plots, samples of the posterior, and a summary table.

Subsequently, the sample size calculator can be used, which prints a matrix with two columns. The first column contains the sample size per group and the second column the associated power. Furthermore, sample.size.calc produces a histogram to illustrate power as evaluated in the last iteration. Detailed descriptions are provided in the Shiny application. Alternatively, the power for a specific combination of group sample sizes can be calculated with the power calculator (power.calc).

If information for a new study is provided, the prior predictive check can be used to compute the prior predictive p-value. The prior predictive check generates a histogram of F_sim in which F_n is indicated with a vertical line. The proportion of F_sim at the right of this line constitutes the prior predictive p-value. The user can download this histogram and sumFdist: a summary of F_sim.

See Also

See also `Gibbs.ANOVA`, `sample.size.calc`, `power.calc`, `prior.predictive.check`, `Fbar.ineq`, `Fbar.dif`, and `Fbar.exact`.

`sample.size.calc` *Sample size calculator for the prior.predictive.check function*

Description

Calculates the required sample size for a new study to conduct the prior predictive check with sufficient statistical power.

Usage

```
sample.size.calc(start_n=20,itmax=10,nmax=600,powtarget=.825,powmargin=.025,
                 posterior,g.m,p.sd,
                 statistic,Amat=0L,exact=0L,difmin=0L,effectsize=FALSE,
                 alpha=.05,printit=TRUE)
```

Arguments

`start_n`	integer; the starting value for the sample size per group. Defaults to 20.
`itmax`	integer; the maximum number of iterations for the function.
`nmax`	integer; the maximum total sample size to evaluate.
`powtarget`	integer; the target power for which the sample size is to be obtained.
`powmargin`	integer; the margin around the target power for which results are to be returned.
`posterior`	a matrix (e.g., the output of Gibbs.ANOVA) with samples from the posterior based on the original data (i.e., y_o).
`g.m`	vector; the population values the alternative distribution. To calculate the power to reject replication if the means are equal specify the grand mean of the study variables in the original dataset.
`p.sd`	integer; the population value for the pooled standard deviation in the alternative distribution. We advice to specify the pooled standard deviation for the study variables in the original dataset.
`statistic`	the type of hypothesis to be calculated: "ineq" for inequality constrained means, "dif" for inequality constraints plus minimum differences between means, "exact" for specific values for the means.
`Amat`	a p by q matrix, where p is the number of means in the ANOVA model, and q is the number of constraints to be imposed on the model. Each row represents one constraint where the parameter with the lower value according to the constraint receives the value -1, and the parameter with the higher value according to the constraint receives the value 1. Other parameters within the same row obtain the value 0.
`exact`	a vector of length p, where p is the number of means in the ANOVA model, with the exact values of the constrained hypothesis.
`difmin`	a vector of length q with the minimum difference per constraint as specified in `Amat`.
`effectsize`	logical; If TRUE the values in `difmin` are interpreted as Cohen's d.
`alpha`	integer; the level of alpha that should be taken into account while calculating the required sample size.
`printit`	logical; If TRUE the current iteration is printed to provide an indication of progress. Default is TRUE.

Value

Prints iterations while calculating. Prints a matrix with two columns. The first column contains the sample size per group and the second column the associated power. Furthermore, sample.size.calc produces a histogram with the null (i.e., the red distribution) and alternative distribution (i.e., the blue distribution) for the last iteration afterwards. The vertical line indicates rej.value (i.e., the 1-alpha'th percentile of the null distribution). The proportion of the alternative distribution on the right side of rej.value constitutes the statistical power.

Author(s)

M.A.J. Zondervan-Zwijnenburg

References

Zondervan-Zwijnenburg, M.A.J., Van de Schoot, R., & Hoijtink, H. (2017). Testing ANOVA replication by means of the prior predictive p-value.

See Also

See also `runShiny`, `prior.predictive.check`, `power.calc`, `Fbar.ineq`, `Fbar.dif`, and `Fbar.exact`.

Examples

```
#analysis original data
data_o <- data.frame(y=ChickWeight$weight,g=ChickWeight$Diet)
#compute pooled sd
sd.g <- aggregate(data_o$y,by=list(data_o$g),sd)[,2]
n.g <- table(data_o$g)
p.sd<- pooled.sd(data_o)

post <- Gibbs.ANOVA(data_o)
#create matrices HR: g4>(g1,g2,g3). g4-g1>0.8, g4-g2>0.5, g4-g3>0.2
HR <- create_matrices(varnames = c(”g1”,”g2”,”g3”,”g4”),
                      hyp = "g4-g1>0.8 & g4-g2>0.5 & g4-g3>0.2”)

Amat <- HR$Amat
difmin <- HR$difmin

#sample size calculator
sample.size.calc(start_n=30, powtarget=.825,powmargin=.025,posterior=post$posterior,
                 g.m=rep(mean(data_o$y),4),p.sd=p.sd,
                 statistic="dif",effectsize=TRUE,Amat=Amat,difmin=difmin)
```

Index