Package 'PsumtSim'

February 19, 2015

Version 0.4

Date 2012-08-26

Title Simulations of grouped responses relative to baseline.

Author Peter N. Steinmetz <PeterNSteinmetz@steinmetz.org>

Maintainer Peter N. Steinmetz <PeterNSteinmetz@steinmetz.org>

Description Functions to simulate Poisson or Normally distributed responses relative to a baseline and compute achieved significance level and powers for tests on the simulated responses.

License GPL (>= 3)

Depends EffectsRelBaseline, boot

NeedsCompilation no

Repository CRAN

Date/Publication 2013-10-01 07:42:26

R topics documented:

PsumtSim-package 2
calcNumRejects 3
catEffectBootAdaptor 4
compExclusionFraction 5
compPowerCatSelectivity 6
compPowerGeneralRespDetection 8
compPowerRespDetection 10
compRejectionFraction 12
simCatResp 14
simNormCatResp 15
testCatEffectBoot 16

Index **18**

PsumtSim-package *Simulations of grouped responses relative to baseline.*

Description

Functions to simulate Poisson distributed responses relative to a baseline and compute achieved significance level and powers for tests on the simulated responses. These functions were used to perform the calculations in the paper by Steinmetz & Thorp (2013).

Details

Package:	PsumtSim
Version:	0.4
Date:	2012-08-26
License:	GPL version 3 or newer
Built:	R 2.13.0; ; 2012-02-03 22:41:44 UTC; unix

Index:

calcNumRejects Calculate number of cases rejected in grouped Poisson responses.
catEffectBootAdaptor
Adaptor to use a statistic calculating function with boot functions.
compExclusionFraction
Computes fraction of simulated cells, with Poisson responses to background and varying rates to different categories, which will have an effect of category but be excluded by pre-testing.
compPowerCatSelectivity
Compute number of simulated neurons with a significant effect of category using a bootstrapped F-ratio test.
compPowerGeneralRespDetection
Perform repeated simulations of grouped responses, where all groups differ from baseline and determine number significant.
compPowerRespDetection
Perform repeated simulations of grouped responses, where some groups differ from baseline and determine number significant.
compRejectionFraction Compute rejection fraction for sequential tests.
simCatResp Simulate grouped Poisson responses.
simNormCatResp Simulate grouped responses which are Normally distributed.
testCatEffectBoot Test for an effect of category using bootstrapping.

This package provides a set of functions for simulating grouped responses and testing them for significant deviations from baseline. This is primarily of use for computing power of different testing methods.

The highest level functions are compPowerGeneralRespDetection and compPowerRespDetection which will perform repeated simulation and testing, determining the number of simulations which

produce significant results.

The example for the compPowerRespDetection shows code to generate the data in figure 4 of Steinmetz & Thorp 2013 and the example for compPowerCatSelectivity shows code to generate figure 5.

Author(s)

Peter N. Steinmetz <PeterNSteinmetz@steinmetz.org>, Christopher Thorp <thorp@spacia.org>

Maintainer: Peter N. Steinmetz <PeterNSteinmetz@steinmetz.org>

References

Efron B, Tibshirani RJ. An Introduction to the Bootstrap (Chapman & Hall/CRC Monographs on Statistics & Applied Probability). Chapman and Hall/CRC; 1994.

Steinmetz, P.N. & Thorp, C.K. (2013) Testing for effects of different stimuli on neuronal firing relative to background activity. Journal of Neural Engineering, Sept. 2013.

See Also

EffectsRelBaseline, boot

`calcNumRejects` *Calculate number of cases rejected for repeated simulation of Poisson background and responses grouped into categories.*

Description

Calculate number of cases rejected for repeated simulation of Poisson background and responses grouped into categories.

Usage

```
calcNumRejects(bkg, resps, numRespsPerCat, numSims, calcPValFnc, sigLevel = 0.05, ...)
```

Arguments

`bkg`	Mean firing rate during a background interval, unrelated to stimulus presentation.
`resps`	Mean firing rates during a response period, one for each category.
`numRespsPerCat`	Number of repetitions (presentation of stimuli) in each category.
`numSims`	Number of simulation to run.
`calcPValFnc`	Function to be called with simulation output from simCatResp to calculate the achieved significance level, or p-value.
`sigLevel`	Significance level to use when determining if test is significant. Default is 0.05.
`...`	Other arguments to pass to `calcPValFnc`

Value

Number of simulations which were detected as significant, out of numSims

Author(s)

Peter N. Steinmetz <PeterNSteinmetz@steinmetz.org>

See Also

compPowerRespDetection, compPowerGeneralRespDetection, simCatResp

Examples

```
# Calculate number of cases which will be detected as having an effect of
# category when there are 4 categories with 2 having different responses and
# when use a standard F test to detect the category effect.
pvalFnc<-function(df){anova(glm(resp~category,data=df),test='F')$"Pr(>F)"[2]}
calcNumRejects(1,c(1,1.5,2,1),6,100,pvalFnc)
```

---

catEffectBootAdaptor *Adaptor for testing category effects in simulations using the boot package and function.*

Description

Adaptor for testing category effects in simulations using the boot package and function.

Usage

```
catEffectBootAdaptor(df, index, testFnc = sumSqCat, useResp = TRUE, ...)
```

Arguments

df	dataframe structured as that returned by simCatResp, with category, bkg, resp columns.
index	index for rearrangement of data as provided by boot function
testFnc	function to calculate test statistic based its first argument, using the category labels in the second argument, and passed the other arguments (...) given to this adaptor function.
useResp	true if the statistic should be calculated for the rearranged resp column of the dataframe, otherwise false to use the bkg column.
...	other arguments to pass to the testFnc

Value

value of the test statistic

Author(s)

Peter N. Steinmetz <PeterNSteinmetz@steinmetz.org>

See Also

boot

Examples

```
# Simulate 4 categories of responses and then compute the number of times
# there would be a significant effect of a change in the categories relative
# to background using the CBT. This would form a rough estimate of the power
# of the CBT to detect such changes.
library(boot)
library(EffectsRelBaseline)
simCatResp(1,c(1,1.5,2,1),6)->sim3
boot(sim3,catEffectBootAdaptor,100,backMean=mean(1,1.5,2,1))->boot3
sum(boot3$t0>boot3$t)
```

---

compExclusionFraction *Compute fraction of cells with category selective response which are excluded by pre-testing.*

Description

Computes fraction of simulated cells, with Poisson responses to background and varying rates to different categories, which will have an effect of category but be excluded by pre-testing.

Usage

```
compExclusionFraction(bkg, resps, numTrialsPerCat, pretestP, anovaP, showProgress = FALSE,
numCells = 1000)
```

Arguments

bkg	Mean firing rate during a background interval, unrelated to stimulus presentation.
resps	Mean firing rates during a response period, one for each category.
numTrialsPerCat	Number of repetitions (presentation of stimuli) in each category.
pretestP	Significance level applied in pre-tests for effect of each category.
anovaP	Significance level applied in ANOVA for effect of category.
showProgress	TRUE if should list cell number as calculating. Default is FALSE.
numCells	Number of cells to simulate. Default is 1000.

Value

`exclusionFrac` Fraction of cells which were not rejected by the first t-test; thus fraction of those ignored for the second test.

`catSelectiveFrac` Fraction of all cells determined to have an effect of category on responses.

`catSelExclFrac` Fraction of cells with a significant response to category which are excluded because they were not rejected by the first t-test.

Note

The t-test which is performed for pre-selection is two-sided.

Author(s)

Peter N. Steinmetz <PeterNSteinmetz@steinmetz.org>

References

Steinmetz, P.N. & Thorp, C.K. (2013) Testing for effects of different stimuli on neuronal firing relative to background activity. Journal of Neural Engineering, Sept. 2013.

Examples

```
# This set of rates and alphas will exclude approximately 62% of neurons with
# an effect of category on firing rates.
compExclusionFraction(10,c(10,9.5,10.5),5,
pretestP=0.01,anovaP=0.05,showProgress=TRUE)
```

`compPowerCatSelectivity` *Compute number of simulated neurons with a significant effect of category using a bootstrapped F-ratio test.*

Description

Compute number of simulated neurons with a significant effect of category using a bootstrapped F-ratio test.

Usage

```
compPowerCatSelectivity(respRates, normDistribution = FALSE, showProgress = FALSE,
numTrialsPerCat = 15, numBootIters = 1000, numRuns = 1000, alpha = 0.05)
```

Arguments

respRates	Vector of rates of responses in each category to be simulated.
normDistribution	TRUE if normally distributed response and background counts should be used, otherwise Poisson distributed counts are used. Default is FALSE.
showProgress	TRUE if run count should be printed. Default is FALSE.
numTrialsPerCat	Number of trials of responses and backgrounds for each category. Default is 15.
numBootIters	Number of boot iterations, passed as 'R' argument to boot function. Default is 1000.
numRuns	Number of simulations to perform followed by bootstrap testing. Default is 1000.
alpha	Significance level, alpha, to use in counting if simulated results are significant. Default is 0.05.

Value

Number of trials, out of numRuns which were detected as significant in the bootstrap testing.

Author(s)

Peter N. Steinmetz <PeterNSteinmetz@steinmetz.org>

References

Steinmetz, P.N. & Thorp, C.K. (2013) Testing for effects of different stimuli on neuronal firing relative to background activity. Journal of Neural Engineering, Sept. 2013.

Examples

```
## Not run:
#
# Compute power of the F-ratio test for category selectivity over
# a range of plausible simulation parameters, placing results
# in a dataframe. This computes values a subset of which were used to
# generate figure 5 of Steinmetz & Thorp, 2013.
#

if (exists('res',inherits=FALSE)) rm('res')

numRuns<-1000

for (numCats in c(5,10,20,30)) {
  for (anovaP in c(0.01,0.05)) {
    for (meanRate in seq(0.5,10,by=0.5)) {
      for (fracChange in c(0.1,0.25,0.5,1)) {
        minRate<-meanRate*(1-fracChange)
        maxRate<-meanRate*(1+fracChange)
```

```
        respLevels<-seq(minRate,maxRate,length.out=numCats)

        print(paste(numCats,anovaP,meanRate,fracChange))

        numSig<-compPowerCatSelectivity(respLevels,
                                        alpha=anovaP, numRuns=numRuns,
                                        showProgress=TRUE)

        df1<-data.frame(numCats=numCats,
                        fracChange=fracChange, anovaP=anovaP, meanRate=meanRate,
                        frac=numSig/numRuns)

        if (!exists('res',inherits=FALSE)) res<-df1
        else res<-rbind(res,df1)
      }
    }
  }
}


## End(Not run)

#
# Example of smaller simulation corresponding to circle at 0.5 fractional
# change in figure 5A of Steinmetz & Thorp, 2013.
#
numRuns<-5
numCats<-30
anovaP<-0.05
meanRate<-0.5
fracChange<-0.5
minRate<-meanRate*(1-fracChange)
maxRate<-meanRate*(1+fracChange)
respLevels<-seq(minRate,maxRate,length.out=numCats)

print(paste(numCats,anovaP,meanRate,fracChange))

numSig<-compPowerCatSelectivity(respLevels,
                                    alpha=anovaP, numRuns=numRuns,
                                    showProgress=TRUE)

df1<-data.frame(numCats=numCats,
                fracChange=fracChange, anovaP=anovaP, meanRate=meanRate,
                frac=numSig/numRuns)
```

`compPowerGeneralRespDetection`

Compute power to detect differences from background where all categories differ from background.

Description

Compute power to detect responses differing from background in a simulation of responses to a number of categories of stimuli, all of which differ from the background level of firing.

Usage

```
compPowerGeneralRespDetection(bkgLevel, generalRespLevel, respLevel, numCats,
numCatsWithResp, normDistribution = FALSE, showProgress = FALSE, numTrialsPerCat = 15,
numBootIters = 1000, numRuns = 1000, alpha = 0.05)
```

Arguments

bkgLevel	Background firing rate, prior to stimulus onset.
generalRespLevel	Response firing rate applied to categories not having the respLevel of firing.
respLevel	Response firing rate during categories which have a response other than background.
numCats	Total number of categories of stimulus.
numCatsWithResp	Number of categories to be given respLevel responses.
normDistribution	TRUE if normally distributed response and background counts should be used, otherwise Poisson distributed counts are used. Default is FALSE.
showProgress	TRUE if run count should be printed. Default is FALSE.
numTrialsPerCat	Number of trials of responses and backgrounds for each category. Default is 15.
numBootIters	Number of boot iterations, passed as 'R' argument to boot function. Default is 1000.
numRuns	Number of simulations to perform followed by bootstrap testing. Default is 1000.
alpha	Significance level, alpha, to use in counting if simulated results are significant. Default is 0.05.

Value

Number of trials, out of numRuns which were detected as significant in the bootstrap testing.

Author(s)

Peter N. Steinmetz <PeterNSteinmetz@steinmetz.org>

See Also

compPowerRespDetection

Examples

```
# Simulate responses in 3 of 6 categories and compute how often these responses
# are detected. Will return a random number generally between 10 and 15, so
# the estimate of power would be between 20% and 30%.
require(EffectsRelBaseline)
require(boot)
compPowerGeneralRespDetection(1,1,2,6,3,numBootIters=50,numRuns=25)
```

`compPowerRespDetection` *Compute power to detect differences from background where some categories differ from background.*

Description

Compute power to detect responses differing from background in a simulation of responses to a number of categories of stimuli, some of which differ from a background level of firing.

Usage

```
compPowerRespDetection(bkgLevel, respLevel, numCats, numCatsWithResp, normDistribution =
FALSE, showProgress = FALSE, numTrialsPerCat = 15, numBootIters = 1000, numRuns = 1000,
alpha = 0.05)
```

Arguments

`bkgLevel`	Background firing rate, prior to stimulus onset.
`respLevel`	Response firing rate during categories which have a response other than background.
`numCats`	Total number of categories of stimulus.
`numCatsWithResp`	Number of categories to be given `respLevel` responses.
`normDistribution`	TRUE if normally distributed response and background counts should be used, otherwise Poisson distributed counts are used. Default is FALSE.
`showProgress`	TRUE if run count should be printed. Default is FALSE.
`numTrialsPerCat`	Number of trials of responses and backgrounds for each category. Default is 15.
`numBootIters`	Number of boot iterations, passed as 'R' argument to `boot` function. Default is 1000.
`numRuns`	Number of simulations to perform followed by bootstrap testing. Default is 1000.
`alpha`	Significance level, alpha, to use in counting if simulated results are significant. Default is 0.05.

Value

Number of trials, out of numRuns which were detected as significant in the bootstrap testing.

Author(s)

Peter N. Steinmetz <PeterNSteinmetz@steinmetz.org>

References

Steinmetz, P.N. & Thorp, C.K. (2013) Testing for effects of different stimuli on neuronal firing relative to background activity. Journal of Neural Engineering, Sept. 2013.

See Also

compPowerGeneralRespDetection

Examples

```
#
# Compute power of changes from background test (CBT) over
# a range of plausible simulation parameters, placing results
# in a dataframe.
# This code generates results, a subset of which is shown in figure 4
# of Steinmetz & Thorp 2013.
#
## Not run:
require('PsumtSim')
require('EffectsRelBaseline')
require('boot')

if (exists('res',inherits=FALSE)) rm('res')

possibleCatsWithEffect<-c(1,2,5,10)
numRuns<-1000

numCats in c(5,10,20,30)) {
  catsWithEffect<-possibleCatsWithEffect[possibleCatsWithEffect<=numCats]

  for (anovaP in c(0.01,0.05)) {
    for (bkgRate in seq(0.5,10, by=0.5)) {
      for (numCatsWithEffect in catsWithEffect) {
        for (fracChange in c(0.25, 0.5, 1)) {

          print(paste(numCats,anovaP,bkgRate,numCatsWithEffect,fracChange))

          respLevel<-bkgRate*(1+fracChange)

          numSig<-compPowerRespDetection(bkgRate,respLevel,numCats,
                                         numCatsWithEffect,
                                         alpha=anovaP, numRuns=numRuns,
                                         showProgress=TRUE)
```

```
          df1<-data.frame(numCats=numCats,numCatsEffect=numCatsWithEffect,
                       fracChange=fracChange, anovaP=anovaP, bkgRate=bkgRate,
                       frac=numSig/numRuns)

          if (!exists('res',inherits=FALSE)) res<-df1
          else res<-rbind(res,df1)
        }
      }
    }
  }
}

## End(Not run)

#
# Example to generate short simulation corresponding to cross at 5 in figure 4B
# of Steinmetz & Thorp, 2013. Estimate here will be highly variable due to small
# number of runs.
#

require('PsumtSim')
require('EffectsRelBaseline')
require('boot')
possibleCatsWithEffect<-5
numRuns<-5
numCats<-30
catsWithEffect<-5
anovaP<-0.01
bkgRate<-1
numCatsWithEffect<-5
fracChange<-1
print(paste(numCats,anovaP,bkgRate,numCatsWithEffect,fracChange))

respLevel<-bkgRate*(1+fracChange)

numSig<-compPowerRespDetection(bkgRate,respLevel,numCats,
                               numCatsWithEffect,
                               alpha=anovaP, numRuns=numRuns,
                               showProgress=TRUE)

df1<-data.frame(numCats=numCats,numCatsEffect=numCatsWithEffect,
                fracChange=fracChange, anovaP=anovaP, bkgRate=bkgRate,
                frac=numSig/numRuns)
```

`compRejectionFraction` *Compute rejection fraction for sequential tests.*

Description

Computes the fraction of simulated neurons with Poisson spike counts which are rejected by a sequence of tests. First a t-test for a difference from baseline for any category, followed by an

ANOVA of an effect of category on the cells found to be significant in the first test.

Usage

```
compRejectionFraction(bkgLevel, respLevel, numCats, pretestP, anovaP, showProgress =
FALSE, numTrialsPerCat = 10, numCells = 1000)
```

Arguments

bkgLevel	Average firing rate for background counts.
respLevel	Average firing rate for response counts.
numCats	Number of categories to test for a response in.
pretestP	p-value to use in first t-test for a difference from baseline.
anovaP	p-value to use in second ANOVA testing for an effect of category on the responses.
showProgress	TRUE if should list cell number as calculating. Default is FALSE.
numTrialsPerCat	Number of trials, with background and response counts, in each category. Default is 10.
numCells	Number of cells to simulate. Default is 1000.

Details

If the first and second tests were operating independently, the rejectionFrac would remain constant and equal to the anovaP value for all exclusion fractions.

Value

exclusionFrac	Fraction of cells which were not rejected by the first t-test; thus fraction of those ignored for the second test.
rejectionFrac	Fraction of cells rejected by the first test which were rejected by the second test.

Note

Both the t-test and the ANOVA assume a normal distribution of the counts.

Author(s)

Peter N. Steinmetz <Peter.Steinmetz@steinmetz.org>

References

Steinmetz & Thorp, 2012.

Examples

```
compRejectionFraction(1,1,10,0.01,0.05,showProgress=TRUE)
```

simCatResp *Simulates Poisson distributed responses to stimuli.*

Description

Simulates a set of Poisson responses to stimuli grouped into categories and corresponding background counts unrelated to stimulation.

Usage

```
simCatResp(bkgRate, respRates, numRespsPerCat)
```

Arguments

bkgRate	Mean firing rate during a background interval, unrelated to stimulus presentation.
respRates	Mean firing rates during a response period, one for each category.
numRespsPerCat	Number of repetitions (presentation of stimuli) in each category.

Value

Dataframe with columns for category label, background, and response counts for each trial.

Author(s)

Peter N. Steinmetz <PeterNSteinmetz@steinmetz.org>

See Also

simNormCatResp

Examples

```
simCatResp(10.0,c(10,20,30),5)->sim1
anova(glm(resp~category,data=sim1),test='F')
```

simNormCatResp *Simulates normally distributed responses to stimuli.*

Description

Simulates a set of normally distributed responses to stimuli grouped into categories and corresponding background counts unrelated to stimulation.

Usage

```
simNormCatResp(bkgRate, respRates, numRespsPerCat)
```

Arguments

bkgRate	Mean firing rate during a background interval, unrelated to stimulus presentation.
respRates	Mean firing rates during a response period, one for each category.
numRespsPerCat	Number of repetitions (presentation of stimuli) in each category.

Value

Dataframe with columns for category label, background, and response counts for each trial.

Author(s)

Peter N. Steinmetz <PeterNSteinmetz@steinmetz.org>

See Also

simCatResp

Examples

```
simNormCatResp(10.0,c(10,20,30),5)->sim2
anova(glm(resp~category,data=sim2),test='F')
```

`testCatEffectBoot` *Test for an effect of category using bootstrapping.*

Description

Test for an effect of category using bootstrapping.

Usage

```
testCatEffectBoot(sim, R, testFnc, ...)
```

Arguments

`sim`	Data frame containing data to be permuted, must have 'resp' and 'category' columns.
`R`	Number of bootstrap iterations to execute.
`testFnc`	Function to evaluate one bootstrap iteration, using 'resp' and 'category' columns in the dataframe.
`...`	Any other arguments for `catEffectBootAdaptor` or `testFnc`.

Details

The function `testFnc` is invoked using the `catEffectBootAdaptor` function to select the response or background counts (set with the `useResp` argument) and permute the rows.

Value

Achieved significance level, ASL

Author(s)

Peter N. Steinmetz <PeterNSteinmetz@steinmetz.org>

References

Efron B, Tibshirani RJ. An Introduction to the Bootstrap (Chapman & Hall/CRC Monographs on Statistics & Applied Probability). Chapman and Hall/CRC; 1994.

Steinmetz, P.N. & Thorp, C.K. (2013) Testing for effects of different stimuli on neuronal firing relative to background activity. Journal of Neural Engineering, Sept. 2013.

See Also

`catEffectBootAdaptor`

Examples

```
# run simulation of responses for 4 categories and then test for category
# effect on response using the CBT from Steinmetz & Thorp, 2013
simCatResp(1,c(1,1.5,2,1),6)->sim4
testCatEffectBoot(sim4,100,sumSqCat,backMean=1)
```

Index

∗Topic **Utilities**
    catEffectBootAdaptor, 4
∗Topic **datagen**
    calcNumRejects, 3
    compExclusionFraction, 5
    compPowerCatSelectivity, 6
    compPowerGeneralRespDetection, 8
    compPowerRespDetection, 10
    compRejectionFraction, 12
    simCatResp, 14
    simNormCatResp, 15
∗Topic **nonparametric**
    testCatEffectBoot, 16
∗Topic **package**
    PsumtSim-package, 2

boot, *3*, *5*, *7*, *9, 10*

calcNumRejects, 3
catEffectBootAdaptor, 4, *16*
compExclusionFraction, 5
compPowerCatSelectivity, *3*, 6
compPowerGeneralRespDetection, *2*, *4*, 8, *11*
compPowerRespDetection, *2–4*, *9*, 10
compRejectionFraction, 12

EffectsRelBaseline, *3*

PsumtSim *(*PsumtSim-package*)*, 2
PsumtSim-package, 2

simCatResp, *3, 4*, 14, *15*
simNormCatResp, *14*, 15

testCatEffectBoot, 16