

Package ‘ANOFA’

November 18, 2023

Type Package

Title Analyses of Frequency Data

Version 0.1.3

Date 2023-11-17

Author Denis Cousineau [aut, cre],
Louis Laurencelle [ctb],
Pier-Olivier Caron [ctb]

Maintainer Denis Cousineau <denis.cousineau@uottawa.ca>

BugReports <https://github.com/dcousin3/ANOFA/issues/>

URL <https://dcousin3.github.io/ANOFA/>

Description Analyses of frequencies can be performed using an alternative test based on the G statistic. The test has similar type-I error rates and power as the chi-square test. However, it is based on a total statistic that can be decomposed in an additive fashion into interaction effects, main effects, simple effects, contrast effects, etc., mimicking precisely the logic of ANOVA. We call this set of tools 'ANOFA' (Analysis of Frequency data) to highlight its similarities with ANOVA. This framework also renders plots of frequencies along with confidence intervals. Finally, effect sizes and planning statistical power are easily done under this framework. The ANOFA is a tool that assesses the significance of effects instead of the significance of parameters; as such, it is more intuitive to most researchers than alternative approaches based on generalized linear models.

See Laurencelle and Cousineau (2023) <[doi:10.20982/tqmp.19.2.p173](https://doi.org/10.20982/tqmp.19.2.p173)>.

License GPL-3

Encoding UTF-8

VignetteBuilder knitr

LazyData true

RoxygenNote 7.2.3

Depends R (>= 3.5.0)

Imports rraply (>= 1.2.6), superb (>= 0.95.0), Rdpack (>= 0.7),
ggplot2 (>= 3.1.0), stats, utils

Suggests rmarkdown, testthat, knitr

RdMacros Rdpack

NeedsCompilation no

Repository CRAN

Date/Publication 2023-11-18 14:20:08 UTC

R topics documented:

anofa	2
anofaES	4
anofaN2Power	5
contrastFrequencies	6
conversion	7
count	9
Detergent	12
emFrequencies	13
explain	14
Gillet1993	15
GRF	16
is.formula	17
LandisBarrettGalvin2013	18
LightMargolin1971	19
minimalExample	20
summarize	21
Index	23

anofa	<i>anofa: analysis of frequency data.</i>
-------	---

Description

The function `anofa()` performs an anofa of frequencies for designs with up to 4 factors according to the anofa framework. See Laurencelle and Cousineau (2023) for more.

Usage

```
anofa(formula = NULL, data = NULL, factors = NULL)
```

Arguments

formula	A formula with the factors on the left-hand side. See below for writing the formula according to the data format.
data	Dataframe in one of wide, long, raw or compiled format;
factors	For raw data formats, provide the factor names.

Details

The data can be given in four formats:

- wide: In the wide format, there is one line for each participant, and one column for each factor in the design. In the column(s), the level must of the factor is given (as a number, a string, or a factor).
- long: In the long format, there is an identifier column for each participant, a factor column and a level number for that factor. If there are n participants and m factors, there will be in total n x m lines.
- raw: In the raw column, there are as many lines as participants, and as many columns as there are levels for each factors. Each cell is a 0/1 entry.
- compiled: In the compiled format, there are as many lines as there are cells in the design. If there are two factors, with two levels each, there will be 4 lines. See the vignette `DataFormatsForFrequencies` for more on data format and how to write their formula.

Value

a model fit to the given frequencies. The model must always be an omnibus model (for decomposition of the main model, follow the analysis with `emFrequencies()` or `contrastFrequencies()`)

References

Laurencelle L, Cousineau D (2023). “Analysis of frequency tables: The ANOFA framework.” *The Quantitative Methods for Psychology*, **19**, 173–193. doi:10.20982/tqmp.19.2.p173.

Examples

```
# Basic example using a single-factor design with the data in compiled format.
# Fictitious data present frequency of observation classified according
# to Intensity (three levels) and Pitch (two levels) for 6 possible cells.
minimalExample

formula <- Frequency ~ Intensity * Pitch
w <- anofa(formula, minimalExample)
summary(w)

# To know more about other ways to format the datasets,
# see, e.g., `toRaw()`, `toLong()`, `toWide()`
w <- anofa(formula, minimalExample)
toWide(w)
# See the vignette `DataFormatsForFrequencies` for more.

# Real-data example using a two-factor design with the data in compiled format:
LandisBarrettGalvin2013

w <- anofa( obsfreq ~ program * provider, LandisBarrettGalvin2013 )
summary(w)

# You can ask easier outputs
w <- anofa(formula, minimalExample)
```

```
summarize(w) # or summary(w) for the ANOVA table  
explain(w)  # human-readable output
```

anofaES

Computing effect size within the ANOVA.

Description

The function `anofaES()` compute effect size from observed frequencies according to the ANOVA framework. See Laurencelle and Cousineau (2023) for more.

Usage

```
anofaES( props )
```

Arguments

`props` the expected proportions;

Details

The effect size is given as an eta-square.

Value

The predicted effect size from a population with the given proportions.

References

Laurencelle L, Cousineau D (2023). “Analysis of frequency tables: The ANOVA framework.” *The Quantitative Methods for Psychology*, **19**, 173–193. doi:10.20982/tqmp.19.2.p173.

Examples

```
# if we assume the following proportions:  
pred <- c(.35, .25, .25, .15)  
  
# then eta-square is given by  
anofaES( pred )
```

`anofaN2Power`*Computing power within the ANOVA.*

Description

The function `anofaN2Power()` performs an analysis of statistical power according to the ANOVA framework. See Laurencelle and Cousineau (2023) for more. `anofaPower2N()` computes the sample size to reach a given power.

Usage

```
anofaPower2N(power, P, f2, alpha)
```

```
anofaN2Power(N, P, f2, alpha)
```

Arguments

<code>N</code>	sample size;
<code>P</code>	number of groups;
<code>f2</code>	effect size Cohen's f^2 ;
<code>alpha</code>	(default if omitted .05) the decision threshold.
<code>power</code>	target power to attain;

Value

a model fit to the given frequencies. The model must always be an omnibus model (for decomposition of the main model, follow the analysis with `emfrequencies()` or `contrasts()`)

References

Laurencelle L, Cousineau D (2023). "Analysis of frequency tables: The ANOVA framework." *The Quantitative Methods for Psychology*, **19**, 173–193. doi:10.20982/tqmp.19.2.p173.

Examples

```
# 1- The Landis et al. study had tremendous power with 533 participants in 15 cells:
# where 0.2671 is the observed effect size for the interaction.
anofaN2Power(533, 5*3, 0.2671)
# power is 100% because sample is large and effect size is as well.

# Even with a quarter of the participants, power is overwhelming:
# because the effect size is quite large.
anofaN2Power(533/4, 5*3, 0.2671)

# 2- Power planning.
# Suppose we plan a four-classification design with expected frequencies of:
pred <- c(.35, .25, .25, .15)
```

```
# P is the number of classes (here 4)
P <- length(pred)
# We compute the predicted f2 as per Eq. 5
f2 <- 2 * sum(pred * log(P * pred) )
# the result, 0.0822, is a moderate effect size.

# Finally, aiming for a power of 80%, we run
anofaPower2N(0.80, P, f2)
# to find that a little more than 132 participants are enough.
```

contrastFrequencies *contrastFrequencies: contrasts analysis of frequency data.*

Description

The function `contrastFrequencies()` performs contrasts analyses of frequencies after an omnibus analysis has been obtained with `anofa()` according to the ANOVA framework. See Laurencelle and Cousineau (2023) for more.

Usage

```
contrastFrequencies(w = NULL, contrasts = NULL)
```

Arguments

<code>w</code>	An ANOVA object obtained from <code>anofa()</code> or <code>emFrequencies()</code> ;
<code>contrasts</code>	A list that gives the weights for the contrasts to analyze. The contrasts within the list can be given names to distinguish them. The contrast weights must sum to zero and their cross-products must equal 0 as well.

Details

`contrastFrequencies` computes the Gs for the contrasts, testing the hypothesis that it equals zero. The contrasts are each 1 degree of freedom, and the sum of the contrasts' degrees of freedom totalize the effect being decomposed.

Value

a model fit of the contrasts.

References

Laurencelle L, Cousineau D (2023). "Analysis of frequency tables: The ANOVA framework." *The Quantitative Methods for Psychology*, **19**, 173–193. doi:10.20982/tqmp.19.2.p173.

Examples

```

# Basic example using a two-factors design with the data in compiled format.
# Fictitious data present frequency of observation classified according
# to Intensity (three levels) and Pitch (two levels) for 6 possible cells.
minimalExample

# performs the omnibus analysis first (mandatory):
w <- anofa(Frequency ~ Intensity * Pitch, minimalExample)
summary(w)

# execute the simple effect of Pitch for every levels of Intensity
e <- emFrequencies(w, ~ Intensity | Pitch)
summary(e)

# For each Pitch, contrast the three intensities, first
# by comparing the first two levels to the third, second
# by comparing the first to the second level:
w3 <- contrastFrequencies( e, list(
  contrast1 = c(1, 1, -2)/2,
  contrast2 = c(1, -1, 0) )
)
summary(w3)

# Example using the Landis et al. (2013) data, a 3 x 5 design involving
# program of care (3 levels) and provider of care (5 levels).
LandisBarrettGalvin2013

# performs the omnibus analysis first (mandatory):
w <- anofa(obsfreq ~ provider * program, LandisBarrettGalvin2013)
summary(w)

# execute the simple effect of Pitch for every levels of Intensity
e <- emFrequencies(w, ~ program | provider)
summary(e)

# For each Pitch, contrast the three intensities, first
# by comparing the first two levels to the third, second
# by comparing the first to the second level:
w3 <- contrastFrequencies( e, list(
  contrast1 = c(1, 1, -2)/2,
  contrast2 = c(1, -1, 0) )
)
summary(w3)

```

Description

The functions `toWide()`, `toLong()`, `toCompiled()` `toRaw()` and `toTabular()` converts the data into various formats.

Usage

```
toWide(w)
```

```
toLong(w)
```

```
toCompiled(w)
```

```
toRaw(w)
```

```
toTabular(w)
```

Arguments

`w` An instance of an ANOVA object.

Details

The classification of a set of n participants can be given using many formats. One basic format (called wide herein) has n lines, one per participants, and category names assigned to each. Another format (called compiled herein) is to have a list of all the categories and the number of participants falling in each cells. This last format is typically much more compact (if there are 6 categories, the data are all contained in six lines). However, we fail to see each individual contributing to the counts. See the vignette `DataFormatsForFrequencies` for more. A third possible format (called raw herein) put one column per category and 1 is the observation matches this category, 0 otherwise. This format results in n lines, one participants, and as many columns are there are categories. Lastly, a fourth format (called long herein) as, on a line, the factor name and the category assigned in that factor. If there are f factors and n participants, the data are in $f*n$ lines.

See the vignette `DataFormatsForFrequencies` for more.

Value

a data frame in the requested format.

Examples

```
# The minimalExample contains n of 20 participants categorized according
# to two factors f = 2, namely `Intensity` (three levels)
# and Pitch (two levels) for 6 possible cells.
minimalExample

# Lets incorporate the data in an anova data structure
w <- anova( Frequency ~ Intensity * Pitch, minimalExample )
```



```

# The data presented using various formats looks like
toWide(w)
# ... has 20 lines ($n$) and 2 columns ($f$)

toLong(w)
# ... has 40 lines ($n \times f$) and 3 columns (participant's `Id`, `Factor` name and `Level`)

toRaw(w)
# ... has 20 lines ($n$) and 5 columns ($2+3$)

toCompiled(w)
# ... has 6 lines ($2 \times 3$) and 3 columns ($f$ + 1)

toTabular(w)
# ... has one table with $2 \times 3$ cells. If there had been
# more than two factors, the additional factor(s) would be on distinct layers.

```

count	<i>anofaPlot.</i>
-------	-------------------

Description

The function `anofaPlot()` performs a plot of frequencies for designs with up to 4 factors according to the ANOVA framework. See Laurencelle and Cousineau (2023) for more. The plot is realized using the superb library; see Cousineau et al. (2021). The functions `count()`, `init.count()` and `CI.count()` are internal functions.

Usage

```
anofaPlot(w, formula, confidenceLevel = .95, showPlotOnly = TRUE, plotStyle = "line",
  errorbarParams = list( width =0.5, linewidth=0.75 ), ...)
```

```
count(n)
```

```
init.count(df)
```

```
CI.count(n, gamma =0.95)
```

Arguments

n	the count for which a confidence interval is required
w	An ANOVA object obtained with <code>anofa()</code> ;
formula	(optional) Use formula to plot just specific terms of the omnibus test. For example, if your analysis stored in <code>w</code> has factors A, B and C, then <code>anofaPlot(w, ~ A * B)</code> will only plot the factors A and B.

confidenceLevel	Provide the confidence level for the confidence intervals. (default is 0.95, i.e., 95%).
plotStyle	(optional; default "line") How to plot the frequencies. See superb for other layouts (e.g., "line")
showPlotOnly	(optional, default True) shows only the plot or else shows the numbers needed to make the plot yourself.
errorbarParams	(optional; default list(width =0.5, linewidth=0.75)) A list of attributes used to plot the error bars. See superb for more.
...	Other directives sent to superb(), typically 'plotStyle', 'errorbarParams', etc.
df	a data frame for initialization of the CI function
gamma	the confidence level

Details

The plot shows the frequencies (the count of cases) on the vertical axis as a function of the factors (the first on the horizontal axis, the second if any in a legend; and if a third or even a fourth factors are present, as distinct rows and columns). It also shows 95% confidence intervals of the frequency, adjusted for between-cells comparisons. The confidence intervals are based on the Clopper and Pearson method (Clopper and Pearson 1934) using the Leemis and Trivedi analytic formula (Leemis and Trivedi 1996). This "stand-alone" confidence interval is then adjusted for between-cell comparisons using the superb framework (Cousineau et al. 2021).

See the vignette `DataFormatsForFrequencies` for more on data format and how to write their formula. See the vignette `ConfidenceInterval` for details on the adjustment and its purpose.

Value

a `ggplot2` object of the given frequencies.

References

Clopper CJ, Pearson ES (1934). "The use of confidence or fiducial limits illustrated in the case of the binomial." *Biometrika*, **26**, 404-413. doi:10.1093/biomet/26.4.404.

Cousineau D, Goulet M, Harding B (2021). "Summary plots with adjusted error bars: The superb framework with an implementation in R." *Advances in Methods and Practices in Psychological Science*, **4**, 1-18. doi:10.1177/25152459211035109.

Laurencelle L, Cousineau D (2023). "Analysis of frequency tables: The ANOVA framework." *The Quantitative Methods for Psychology*, **19**, 173-193. doi:10.20982/tqmp.19.2.p173.

Leemis LM, Trivedi KS (1996). "A comparison of approximate interval estimators for the Bernoulli parameter." *The American Statistician*, **50**(1), 63-68.

Examples

```

#
# The Landis et al. (2013) example has two factors, program of treatment and provider of services.
LandisBarrettGalvin2013

# This examine the omnibus analysis, that is, a 5 (provider) x 3 (program):
w <- anofa(obsfreq ~ provider * program, LandisBarrettGalvin2013)

# Once processed into w, we can ask for a standard plot
anofaPlot(w)

# We place the factor `program` on the x-axis:
anofaPlot(w, factorOrder = c("program","provider"))

# The above example can also be obtained with a formula:
anofaPlot(w, ~ program * provider)

# Change the style for a plot with bars instead of lines
anofaPlot(w, plotStyle = "bar")

# Changing the error bar style
anofaPlot(w, plotStyle = "bar", errorbarParams = list( width =0.1, linewidth=0.1 ) )

# An example with 4 factors:
## Not run:
dta <- data.frame(Detergent)
dta

w <- anofa( Freq ~ Temperature * M_User * Preference * Water_softness, dta )
anofaPlot(w)
anofaPlot(w, factorOrder = c("M_User","Preference","Water_softness","Temperature"))

# Illustrating the main effect of Temperature (not interacting with other factors)
# and the interaction Preference * Previously used M brand
# (Left and right panels of Figure 4 of the main article)
anofaPlot(w, ~ Temperature)
anofaPlot(w, ~ Preference * M_User)

# All these plots are ggplot2 so they can be followed with additional directives, e.g.
library(ggplot2)
anofaPlot(w, ~ Temperature) + ylim(200,800) + theme_classic()
anofaPlot(w, ~ Preference * M_User) + ylim(100,400) + theme_classic()

## End(Not run)
# etc. Any ggplot2 directive can be added to customize the plot to your liking.
# See the vignette `Example2`.

```

Detergent

Detergent data

Description

The data, taken from Ries and Smith (1963), is a dataset examining the distribution of a large sample of customers, classified over four factors: Softness of water used (3 levels: soft, medium or hard), Expressed preference for brand M or X after blind test (2 levels: Brand M or Brand X), Previously used brand M (2 levels: yes or no), and Temperature of laundry water (2 levels: hot or cold). It is therefore a $3 \times 2 \times 2 \times 2$ design with 24 cells.

Usage

Detergent

Format

An object of class list.

Source

[doi:10.20982/tqmp.19.2.p173](https://doi.org/10.20982/tqmp.19.2.p173)

References

Ries P, Smith H (1963). “The use of chi-square for preference testing in multidimensional problems.” *Chemical Engineering Progress*, **59**, 39-43.

Examples

```
# convert the data to a data.frame
dta <- data.frame(Detergent)

# run the anofa analysis
## Not run:
w <- anofa( Freq ~ Temperature * M_User * Preference * Water_softness, dta)

# make a plot with all the factors
anofaPlot(w)

# ... or with just a few factors
anofaPlot(w, ~ Preference * M_User )
anofaPlot(w, ~ Temperature )

# extract simple effects
e <- emFrequencies(w, ~ M_User | Preference )

## End(Not run)
```

`emFrequencies`*emFrequencies: simple effect analysis of frequency data.*

Description

The function `emFrequencies()` performs a simple effect analyses of frequencies after an omnibus analysis has been obtained with `anofa()` according to the ANOVA framework. See Laurencelle and Cousineau (2023) for more.

Usage

```
emFrequencies(w, formula)
```

Arguments

<code>w</code>	An ANOVA object obtained from <code>anofa()</code> ;
<code>formula</code>	A formula which indicates what simple effect to analyze. only one simple effect formula at a time can be analyzed. The formula is given using a vertical bar, e.g., " <code>~ factorA factorB</code> " to obtain the effect of Factor A within every level of the Factor B.

Details

`emFrequencies` computes expected marginal frequencies and analyze the hypothesis of equal frequencies. The sum of the Gs of the simple effects are equal to the interaction and main effect Gs, as this is an additive decomposition of the effects.

Value

a model fit of the simple effect.

References

Laurencelle L, Cousineau D (2023). "Analysis of frequency tables: The ANOVA framework." *The Quantitative Methods for Psychology*, **19**, 173–193. doi:10.20982/tqmp.19.2.p173.

Examples

```
# Basic example using a two-factors design with the data in compiled format.  
# Fictitious data present frequency of observation classified according  
# to Intensity (three levels) and Pitch (two levels) for 6 possible cells.  
minimalExample  
  
# performs the omnibus analysis first (mandatory):  
w <- anofa(Frequency ~ Intensity * Pitch, minimalExample)  
summary(w)  
  
# execute the simple effect of Pitch for every levels of Intensity
```

```

e <- emFrequencies(w, ~ Pitch | Intensity)
summary(e)

# As a check, you can verify that the Gs are decomposed additively
sum(e$results[,1])
w$results[3,1]+w$results[4,1]

# Real-data example using a two-factor design with the data in compiled format:
LandisBarrettGalvin2013

w <- anofa( obsfreq ~ provider * program, LandisBarrettGalvin2013)
anofaPlot(w)
summary(w)

# there is an interaction, so look for simple effects
e <- emFrequencies(w, ~ program | provider )
summary(e)

# Example from Gillet1993 : 3 factors for appletrees
Gillet1993

w <- anofa( Freq ~ species * location * florished, Gillet1993)
e <- emFrequencies(w, ~ florished | location )

# Again, as a check, you can verify that the Gs are decomposed additively
w$results[4,1]+w$results[7,1] # B + B:C
sum(e$results[,1])

# You can ask easier outputs with
summarize(w) # or summary(w) for the ANOVA table only
explain(w) # human-readable output ((pending))

```

explain

explain

Description

`explain()` provides a human-readable, exhaustive, description of the results. It also provides references to the key results.

Usage

```
explain(object, ...)
```

Arguments

<code>object</code>	an object to explain
<code>...</code>	ignored

Value

a human-readable output with details of computations.

 Gillet1993

Gillet1993

Description

The data, taken from M. (1993), is a dataset examining the distribution of apple tree to produce new branches from grafts. The study has a sample of 713 trees subdivided into three factors: species (2 levels: Jonagold or Cox); location (3 levels: Order1, Order2, Order3); is where the graft has been implanted (order 1 is right on the trunk); and florished (2 levels: yes or no) indicates if the branch bear flowers. It is therefore a $2 \times 3 \times 2$ design with 12 cells.

Usage

Gillet1993

Format

An object of class list.

References

M. G (1993). *Contribution à la modélisation de la croissance et du développement du pommier*. Faculté des Sciences agronomiques, Gembloux.

Examples

```
# The Gillet1993 presents data from appletrees having grafts.
Gillet1993

# run the base analysis
w <- anofa( Freq ~ species * location * florished, Gillet1993)

# display a plot of the results
anofaPlot(w)

# show the anofa table where we see the 3-way interaction
summary(w)

# This returns the expected marginal frequencies analysis
e <- emFrequencies(w, Freq ~ species * location | florished )
summary(e)

# as seen, all the two-way interactions are significant. Decompose one more degree:
f <- emFrequencies(w, Freq ~ species | florished * location )
summary(f)
```

Description

The function `GRF()` generates random frequencies based on a design, i.e., a list giving the factors and the categories with each factor. The data are given in the compiled format.

Usage

```
GRF( design, n, prob = NULL, f = "Freq" )
```

Arguments

<code>design</code>	A list with the factors and the categories within each.
<code>n</code>	How many simulated participants are to be classified.
<code>prob</code>	(optional) the probability of falling in each cell of the design.
<code>f</code>	(optional) the column names that will contain the frequencies.

Details

The name of the function `GRF()` is derived from `grd()`, a general-purpose tool to generate random data (Calderini and Harding 2019) now bundled in the superb package (Cousineau et al. 2021).

Value

a data frame containing frequencies per cells of the design.

References

Calderini M, Harding B (2019). “GRD for R: An intuitive tool for generating random data in R.” *The Quantitative Methods for Psychology*, **15**(1), 1–11. doi:10.20982/tqmp.15.1.p001.

Cousineau D, Goulet M, Harding B (2021). “Summary plots with adjusted error bars: The superb framework with an implementation in R.” *Advances in Methods and Practices in Psychological Science*, **4**, 1–18. doi:10.1177/25152459211035109.

Examples

```
# The first example disperse 20 participants in one factor having
# two categories (low and high):
design <- list( A=c("low","high"))
GRF( design, 20 )

# This example has two factors, with factor A having levels a, b, c:
design <- list( A=letters[1:3], B = c("low","high"))
GRF( design, 40 )
```



```
# This last one has three factors, for a total of 3 x 2 x 2 = 12 cells
design <- list( A=letters[1:3], B = c("low","high"), C = c("cat","dog"))
GRF( design, 100 )

# To specify unequal probabilities, use
design <- list( A=letters[1:3], B = c("low","high"))
GRF( design, 100, c(.05, .05, .35, .35, .10, .10 ) )

# The name of the column containing the frequencies can be changes
GRF( design, 100, f="patate")
```

is.formula

logical functions for formulas

Description

The functions `is.formula()`, `is.one.sided()`, `has.nested.terms()`, `has.cbind.terms()`, `in.formula()` and `sub.formulas()` performs checks or extract sub-formulas from a given formula.

Usage

```
is.formula(frm)
```

```
is.one.sided(frm)
```

```
has.nested.terms(frm)
```

```
has.cbind.terms(frm)
```

```
in.formula(frm, whatsym)
```

```
sub.formulas(frm, head)
```

Arguments

frm	a formula;
whatsym	a symbol to search in the formula;
head	the beginning of a sub-formula to extract

Details

These formulas are for internal use.

Value

`is.formula(frm)`, `has.nested.terms(frm)`, and `has.cbind.terms(frm)` returns TRUE if `frm` is a formula, contains a `'|'` or a `'cbind'` respectively; `in.formula(frm, whatsym)` returns TRUE if the symbol `whatsym` is somewhere in `'frm'`; `sub.formulas(frm, head)` returns a list of all the sub-formulas which contains `head`.

Examples

```
is.formula( Frequency ~ Intensity * Pitch )  
has.nested.terms( Level ~ Factor | Level )  
has.cbind.terms( Frequency ~ cbind(Low,Medium,High) * cbind(Soft, Hard) )  
in.formula( Frequency ~ Intensity * Pitch, "Pitch" )  
sub.formulas( Frequency ~ cbind(Low,Medium,High) * cbind(Soft, Hard), "cbind" )
```

LandisBarrettGalvin2013

LandisBarrettGalvin2013 data

Description

The data, taken from Landis et al. (2013), is a dataset where the participants ($n = 553$) are classified according to two factors, first, how modalities of care in a family medicine residency program were given. The possible cases were Collocated Behavioral Health service (CBH), a Primary-Care Behavioral Health service (PBH) and a Blended Model (BM). Second, how a patient's care was financed: Medicare (MC), Medicaid (MA), a mix of Medicare/Medicaid (MC/MA), Personal insurance (PI), or Self-paid (\$P). This design therefore has $5 \times 3 = 15$ cells. It was thoroughly examined in (Sharpe 2015) and analyzed in (Laurencelle and Cousineau 2023).

Usage

```
LandisBarrettGalvin2013
```

Format

An object of class `data.frame`.

Source

[doi:10.1037/a0033410](https://doi.org/10.1037/a0033410)

References

Landis SE, Barrett M, Galvin SL (2013). “Effects of different models of integrated collaborative care in a family medicine residency program.” *Families, Systems and Health*, **31**, 264–273. doi:10.1037/a0033410.

Laurencelle L, Cousineau D (2023). “Analysis of frequency tables: The ANOVA framework.” *The Quantitative Methods for Psychology*, **19**, 173–193. doi:10.20982/tqmp.19.2.p173.

Sharpe D (2015). “Chi-square test is statistically significant: Now what?” *Practical Assessment, Research, and Evaluation*, **20**(1), 8.

Examples

```
# running the anofa
L <- anofa( obsfreq ~ provider * program, LandisBarrettGalvin2013)

# getting a plot
anofaPlot(L)

# the G table shows a significant interaction
summary(L)

# getting the simple effect
e <- emFrequencies(L, ~ program | provider )

## Getting some contrast by provider (i.e., on e)
f <- contrastFrequencies(e, list(
  "(PBH & CBH) vs. BM"=c(1,1,-2)/2,
  "PBH vs. CBH"=c(1,-1,0))
)
```

LightMargolin1971

LightMargolin1971 data

Description

The data, taken from Light and Margolin (1971), is an example where the educational aspiration of a large sample of $N = 617$ adolescents. The participants are classified by their gender (2 levels) and by their educational aspiration (complete secondary school, complete vocational training, become college teacher, complete gymnasium, or complete university; 5 levels).

Usage

LightMargolin1971

Format

An object of class `data.frame`.

Source

[doi:10.1080/01621459.1971.10482297](https://doi.org/10.1080/01621459.1971.10482297)

References

Light RJ, Margolin BH (1971). “An Analysis of Variance for Categorical Data.” *Journal of the American Statistical Association*, **66**, 534–544. [doi:10.1080/01621459.1971.10482297](https://doi.org/10.1080/01621459.1971.10482297).

Examples

```
library(ANOfA)

options(superb.feedback = 'none') # shut down 'warnings' and 'design' interpretation messages

# Lets run the analysis
L <- anofa( obsfreq ~ vocation * gender, LightMargolin1971)
summary(L)

# a quick plot
anofaPlot(L)

# Some simple effects.
e <- emFrequencies(L, ~ gender | vocation )
summary(e)

# some contrasts:
e <- emFrequencies(L, ~ vocation | gender )
f <- contrastFrequencies(e, list(
  "teacher college vs. gymnasium"=c( 0, 0, 1,-1, 0),
  "vocational vs. university"   = c( 0, 1, 0, 0,-1),
  "another"                     = c( 0, 1,-1,-1,+1)/2,
  "to exhaust the df"           = c( 4,-1,-1,-1,-1)/4
)
)
```

minimalExample

minimalExample

Description

The data in compiled format are analyzed with an Analysis of Frequency Data method (described in (Laurencelle and Cousineau 2023)).

Usage

```
minimalExample
```

Format

An object of class `data.frame`.

Source

[doi:10.20982/tqmp.19.2.p173](https://doi.org/10.20982/tqmp.19.2.p173)

References

Laurencelle L, Cousineau D (2023). “Analysis of frequency tables: The ANOVA framework.” *The Quantitative Methods for Psychology*, **19**, 173–193. [doi:10.20982/tqmp.19.2.p173](https://doi.org/10.20982/tqmp.19.2.p173).

Examples

```
library(ANOFA)

# the minimalExample data (it has absolutely no effect...)
minimalExample

# perform an anova on this dataset
w <- anova(Frequency ~ Intensity * Pitch, minimalExample)

# We analyse the intensity by levels of pitch
e <- emFrequencies(w, ~ Intensity | Pitch)

# decompose by
f <- contrastFrequencies(e, list(
  "low & medium compared to high" = c(1,1,-2)/2,
  "low compared to medium" = c(1,-1,0)))
```

summarize

summarize

Description

`summarize()` provides a human-readable output of an `ANOFAobject`. it is synonym of `summary()` (but as actions are verbs, I used a verb).

Usage

```
summarize(object, ...)
```

Arguments

object	an object to summarize
...	ignored

Value

a human-readable output as per articles.

Index

* datasets

Detergent, [12](#)
Gillet1993, [15](#)
LandisBarrettGalvin2013, [18](#)
LightMargolin1971, [19](#)
minimalExample, [20](#)

anova, [2](#)
anovaES, [4](#)
anovaN2Power, [5](#)
anovaPlot (count), [9](#)
anovaPower2N (anovaN2Power), [5](#)

CI.count (count), [9](#)
contrastFrequencies, [6](#)
conversion, [7](#)
count, [9](#)

Detergent, [12](#)

emFrequencies, [13](#)
explain, [14](#)

Gillet1993, [15](#)
GRF, [16](#)

has.cbind.terms (is.formula), [17](#)
has.nested.terms (is.formula), [17](#)

in.formula (is.formula), [17](#)
init.count (count), [9](#)
is.formula, [17](#)
is.one.sided (is.formula), [17](#)

LandisBarrettGalvin2013, [18](#)
LightMargolin1971, [19](#)

minimalExample, [20](#)

sub.formulas (is.formula), [17](#)
summarize, [21](#)

toCompiled (conversion), [7](#)
toLong (conversion), [7](#)
toRaw (conversion), [7](#)
toTabular (conversion), [7](#)
toWide (conversion), [7](#)