

# Package ‘rhep’

April 29, 2015

**Type** Package

**Title** Raul Eyzaguirre's R code

**Version** 0.2.0

**Date** 2015-04-28

**Author** Raúl Eyzaguirre

**Maintainer** Raúl Eyzaguirre <reyzaguirre@gmail.com>

**Description** Miscellaneous R functions

**Depends** R (>= 3.1.2)

**Imports** shiny, sn

**License** GPL-3

**LazyData** true

**Suggests** testthat

**Encoding** UTF-8

**NeedsCompilation** no

## R topics documented:

|                      |          |
|----------------------|----------|
| chisq.bin . . . . .  | 2        |
| chisq.comb . . . . . | 2        |
| chisq.pois . . . . . | 3        |
| emtd . . . . .       | 4        |
| mdaplot . . . . .    | 5        |
| minota . . . . .     | 5        |
| multcoef . . . . .   | 6        |
| tfreq . . . . .      | 7        |
| <b>Index</b>         | <b>9</b> |

---

chisq.bin

*Chi-square goodness of fit test for binomial distribution*


---

### Description

This function performs a chi-square goodness of fit test for a binomial distribution.

### Usage

```
chisq.bin(x, f, n = NULL, p = NULL)
```

### Arguments

|   |                        |
|---|------------------------|
| x | The observed values.   |
| f | The observed counts.   |
| n | Binomial parameter n.  |
| p | Binomial parameter pi. |

### Details

If p is not specified, then it is estimated from the data. If there are categories with expected counts less than 5 or less than 1 a warning is shown.

### Value

It returns a table with the contribution to the chi-square statistic for each category, the chi-square statistic, the degrees of freedom, and the p-value.

### Author(s)

Raul Eyzaguirre.

### Examples

```
x <- 0:6
f <- c(334, 369, 191, 63, 22, 12, 9)
chisq.bin(x, f, n = 10)
```

---

chisq.comb

*Combine categories for a chi-square goodness of fit test*


---

### Description

This function combines categories for a chi-square goodness of fit test.

### Usage

```
chisq.comb(chisq.test, combine)
```

**Arguments**

|            |   |
|------------|---|
| chisq.test | The output of a chi-square goodness of fit test by functions chisq.bin or chisq.Pois. |
| combine    | A vector with the numbers of the categories to combine.                               |

**Details**

This function only combines categories on the extremes. It is recommended to combine categories when the expected counts are too low. As a rule of thumb, the chi-square approximation for the test statistic can be unreliable if some categories have expected counts smaller than 5 or if there is any with an expected count smaller than 1.

**Value**

It returns a table with the contribution to the chi-square statistic for each category, the chi-square statistic, the degrees of freedom, and the p-value.

**Author(s)**

Raul Eyzaguirre.

**Examples**

```
x <- 0:6
f <- c(334, 369, 191, 63, 22, 12, 9)
output <- chisq.bin(x, f, n = 10)
# Combine categories 5, 6, and 7
chisq.comb(output, combine = c(5, 6, 7))
```

---

chisq.pois

*Chi-square goodness of fit test for Poisson distribution*


---

**Description**

This function performs a chi-square goodness of fit test for a Poisson distribution.

**Usage**

```
chisq.pois(x, f, lambda = NULL)
```

**Arguments**

|        |                      |
|--------|----------------------|
| x      | The observed values  |
| f      | The observed counts. |
| lambda | Poisson parameter.   |

**Details**

If lambda is not specified, then it is estimated from the data. If there are categories with expected counts less than 5 or less than 1 a warning is shown.

**Value**

It returns a table with the contribution to the chi-square statistic for each category, the chi-square statistic, the degrees of freedom, and the p-value.

**Author(s)**

Raul Eyzaguirre.

**Examples**

```
x <- 0:9
f <- c(6, 16, 48, 77, 72, 72, 46, 39, 15, 9)
chisq.pois(x, f)
```

---

|      |   |
|------|---|
| emtd | <i>Location and scale parameters estimation of a t distribution</i> |
|------|---|

---

**Description**

EM algorithm to estimate the location and scale of a t distribution for given degrees of freedom.

**Usage**

```
emtd(y, v, initmu = mean(y), inits = sd(y), tol = 1e-04)
```

**Arguments**

|        |   |
|--------|---|
| y      | The data.                                 |
| v      | Degrees of freedom.                       |
| initmu | Initial value for the location parameter. |
| inits  | Initial value for the scale parameter.    |
| tol    | Tolerance for the iterative procedure.    |

**Details**

By default the initial values are set to the sample mean and standard deviation.

**Value**

It returns the estimated location and scale parameters for each iteration.

**Author(s)**

Raul Eyzaguirre.

**Examples**

```
# Some data
y = c(10,12,16,15,15,17,20,21,16,24,13,22,14,15,16,16,17,18,19,18,23,20,30)

# Estimates for a t(10)
emtd(y, 10)
```

---

`mdaplot`*Simulate and plot from a normal distribution*

---

**Description**

This function simulates 1000 random samples from a skew normal distribution for specified values of the mean, standard deviation and skewness parameter.

**Usage**

```
mdaplot()
```

**Details**

It uses package `sn` to simulate the data and package `shiny` for the web layout. Type `mdaplot()` in the R console to run the app.

**Value**

It returns a histogram and a boxplot for the simulated data.

**Author(s)**

Raul Eyzaguirre.

---

`minota`*Predice la nota final del curso EP1 y EP2*

---

**Description**

Esta función predice la nota final del curso basado en datos históricos y un modelo de regresión lineal.

**Usage**

```
minota(curso = NULL, vez = NULL, pp = NULL, prob = 0.95, pa1 = NULL,  
       pa2 = NULL, pa3 = NULL, pa4 = NULL, pi1 = NULL, pi2 = NULL,  
       ep = NULL)
```

**Arguments**

|                    |   |
|--------------------|---|
| <code>curso</code> | 1 o 2 (corresponde a EP1 o EP2).                  |
| <code>vez</code>   | Número de veces que se lleva el curso (1, 2 o 3). |
| <code>pp</code>    | Promedio ponderado.                               |
| <code>prob</code>  | Probabilidad para la predicción.                  |
| <code>pa1</code>   | Práctica de aula 1.                               |
| <code>pa2</code>   | Práctica de aula 2.                               |
| <code>pa3</code>   | Práctica de aula 3.                               |

|     |                       |
|-----|-----------------------|
| pa4 | Práctica de aula 4.   |
| pi1 | Práctica integrada 1. |
| pi2 | Práctica integrada 2. |
| ep  | Examen parcial.       |

**Details**

No es necesario introducir todos los parámetros, el modelo solo considera los que son introducidos.

**Value**

Devuelve la nota final estimada con un intervalo de predicción, y el coeficiente de determinación del modelo.

**Author(s)**

Raúl Eyzaguirre.

**Examples**

```
minota(curso = 1, pa1 = 12)
```

---

|          |                                |
|----------|--------------------------------|
| multcoef | <i>Multinomial coefficient</i> |
|----------|--------------------------------|

---

**Description**

Computes the number of permutations of a multiset  $M$  of size  $n$ .

**Usage**

```
multcoef(n, counts)
```

**Arguments**

|        |                                       |
|--------|---------------------------------------|
| n      | The size of $M$ .                     |
| counts | The counts for the repeated elements. |

**Details**

For a set  $M$  with  $k$  unique elements with associate counts  $n_1, n_2, \dots, n_k$ , you only need to specify in the counts argument the counts that are bigger than 1.

**Value**

It returns the multinomial coefficient

$$\frac{n!}{n_1!n_2!\dots n_k!}$$

where

$$n = n_1 + n_2 + \dots + n_k.$$

**Author(s)**

Raul Eyzaguirre.

**Examples**

```
# The number of permutations of the letters in the set M = {A,A,A,B,B,C}
multcoef(6, c(3,2,1))
# Same result with
multcoef(6, c(3,2))
```

---

|       |                                     |
|-------|-------------------------------------|
| tfreq | <i>Frequency distribution table</i> |
|-------|-------------------------------------|

---

**Description**

Constructs a frequency distribution table for a quantitative variable.

**Usage**

```
tfreq(data, limits = NULL, open = "right")
```

**Arguments**

|        |   |
|--------|---|
| data   | The observations to construct the frequency distribution table.         |
| limits | The class limits.   |
| open   | Where to leave the class limits open, left or right. Defaults to right. |

**Details**

If class limits are not specified, the Sturges' rule is used to calculate the number of class intervals  $k$ :

$$k \approx 1 + 3.3 \log n$$

Then, the left limit for the first class interval is set to the minimum value of the data, the range  $r$  is computed and the size of the class intervals is defined by:

$$c \approx \frac{r}{k}$$

where  $c$  is rounded up with the same number of decimal places as the data.

**Value**

It returns a frequency distribution table with columns for class mark, absolute and relative frequencies, and cumulative absolute and relative frequencies.

**Author(s)**

Raul Eyzaguirre.

**Examples**

```
# Some random data from a normal population with mean 10 and standard deviation 1
set.seed(1)
datos <- rnorm(100, 10, 1)
# Data with 3 decimal places
datos <- round(datos, 3)
# A summary of the data
summary(datos)
# Frequency table with 6 specified limits
tfreq(datos, c(7, 8, 9, 10, 11, 12, 13))
# Default method
tfreq(datos)
```



# Index

chisq.bin, [2](#)

chisq.comb, [2](#)

chisq.pois, [3](#)

emtd, [4](#)

mdaplot, [5](#)

minota, [5](#)

multcoef, [6](#)

tfreq, [7](#)