Package 'Biodem'

October 12, 2022

Type Package

Version 0.5

Date 2021-01-05

Title Biodemography Functions

Author Alessio Boattini and Federico C. F. Calboli; Vincente Canto Cassola together with Martin Maechler authored the function mtx.exp.

Maintainer Federico Calboli <f.calboli@gmail.com>

Description The Biodem package provides a number of functions for Biodemographic analysis.

License GPL-2

LazyLoad yes

NeedsCompilation no

Repository CRAN

RoxygenNote 7.1.1

Date/Publication 2021-01-05 15:50:09 UTC

R topics documented:

---

col.sto   *Turns a Migration Matrix into a Column Stochastic Matrix*

Description

Calculates the column stochastic matrix starting from the raw migration matrix x. For each column, it divides each term by the column sum. Then it returns the thus "normalized by column" matrix, ready to be used in the Malecot migration model.

Usage

```
col.sto(x)
```

Arguments

x   the raw data migration matrix

Details

The Malecot model uses a transformation of the raw migration data; in the "Malecot" library the use of a column stochastic matrix follows Imaizumi 1970 and Swedlund 1984.

Value

col.sto is used on a an object of class "matrix" and returns an object of class "matrix".

Author(s)

Federico C. F. Calboli <f.calboli@gmail.com>

References

Imaizumi, Y., N. E. Morton and D. E. Harris. 1970. Isolation by distance in artificial populations. Genetics 66: 569-582.

Jorde, L. B. 1982. The genetic structure of the Utah mormons: migration analysis. Human Biology 54(3): 583-597.

Examples

```
data(raw.mig)
new.mig.mat<-col.sto(raw.mig)
new.mig.mat
```

Fst *Calculates the Fst from the conditional kinship matrix*

Description

Calculates the Fst from a conditional kinship matrix.

Usage

```
Fst(rval, N)
```

Arguments

rval   is a conditional kinship matrix, normally obtained by the functions 'R' and
            'rel.cond' in the Biodem library.

N   the vector of effective populations size, nominally obtained by dividing the total
            population size by three. Starting form surname data, effective population size
            coincides with the number of marriages

Details

The use of the Fst function follows Harpending and Jenkins 1974, and Jorde 1982. It gives an
estimate of Wright's Fst, which is a measure of between-subdivision genetic heterogeneity.

Value

Fst returns one numeric value.

Note

...

Author(s)

Federico C. F. Calboli <f.calboli@gmail.com>

References

Harpending, H. C. and T. jenkins. 1974. !Kung population structure. In: J. F. Crow and C. F.
Denniston (eds.), Genetic distance, pp 137-161. Plenum Press, NY.

Jorde, L. B. 1982. The genetic structure of the Utah mormons: migration analysis. Human Biology
54(3): 583-597.

Examples

```
# Swedlund data again...
data(P); data(S); data(N)
# starting with how many cycles to equilibrium
x<-mal.eq(S,P,N)
# calculation of phi
phi<-mal.phi(S,P,N,x)
# calculation of the conditional kinship matrix
cond<-mal.cond(phi,N)
# finally! we get the Fst value
fst<-Fst(cond,N)
fst

# starting from a raw marriage records dataset:
data(valley)
tot <- sur.freq(valley,valley$PAR,valley$SURM,valley$SURF)
tot # a frequency table calculated above all the surnames
iso.matrix <- uri(tot)
iso.matrix # an unbiased random isonymy matrix
reg <- rri(tot)
reg # a coefficient of unbiased Regional Random Isonymy
kin.cond <- rel.cond(iso.matrix,reg)
kin.cond # a conditional kinship matrix
N <- colSums(tot) # effective population size
fst<-Fst(kin.cond,N)
fst
```

---

hedrick	*Calculates the Hedrick standardized kinship coefficient*

Description

"hedrick"calculates the Hedrick standardized kinship coefficient starting from surname frequencies.

Usage

```
hedrick(x)
```

Arguments

x   is a surname frequency table where the N rows correspond to the surnames present in the whole population and the M columns are the subpopulations

Details

The use of "hedrick" could be problematic, because different people are likely to arrange isonymy data in different ways on their computers. We decided for a matrix format for the isonymy data; the function would originally accept data in a different format and then convert it internally, but this would be a problem for people with data arranged in a different format. In the end we decided

to write a specific function, "sur.freq", to generate surname frequency tables directly from raw marriage data or marriage-like data (the most commonly used sources in bio-demographic studies). For other types of surname data, see the verbose explanation in the info for the dataset "surnames" so it would be clear for the user how "hedrick" works.

Value

Returns a square symmetric standardized kinship matrix.

Note

The Hedrick index was originally conceived as a measure of the probability of genotypic identity between (sub)populations and uses a standardization analogous to that employed when calculating a correlation coefficient. As a consequence, it is equal to 1 if measured on populations with identical surname distribution.

Author(s)

Federico C. F. Calboli and Alessio Boattini <alessio.boattini2@unibo.it>

References

Hedrick, P. W. 1971. A new approach to measuring genetic similarity. Evolution 25: 276-280. Weiss, V. 1980. Inbreeding and genetic distance between hierarchically structured populations measured by surname frequencies. Mankind Quarterly 21: 135-149

See Also

sur.freq to generate the input surname frequency table from marriage data, surnames for an explanation on how to generate the correct input table from other surname sources, laskerand uri for other types of inter-population kinship matrices

Examples

```
# starting from a raw marriage records dataset:
data(valley)
tot <- sur.freq(valley,valley$PAR,valley$SURM,valley$SURF)
tot # a frequency table calculated above all the surnames
hed.kin <- hedrick(tot)
hed.kin # a standardized kinship matrix

#starting from a generic surname frequency table
data(surnames)
surnames #a made-up dataset
# you can see that the surnames are arranged as the _rows_ and
# the populations are the _columns_
# the use of the function "hedrick" just turns this data into a kinship matrix
hed.kin <- hedrick(surnames)
hed.kin
```

---

lasker | *Calculates the lasker kinship coeffcient*

Description

"Lasker"calculates the lasker kinship coefficient starting from a surname frequency table.

Usage

```
lasker(x)
```

Arguments

x   is a surname frequency table where the N rows correspond to the surnames present in the whole population and the M columns are the subpopulations

Details

The use of "lasker" could be problematic, because different people are likely to arrange isonymy data in different ways on their computers. We decided for a matrix format for the isonymy data; the function would originally accept data in a different format and then convert it internally, but this would be a problem for people with data arranged in a different format. In the end we decided to write a specific function, "sur.freq", to generate surname frequency tables directly from raw marriage data or marriage-like data (the most commonly used sources in bio-demographic studies). For other types of surname data, see the verbose explanation in the info for the dataset "surnames" so it would be clear for the user how "lasker" works.

Value

Returns a square symmetric kinship matrix.

Note

...

Author(s)

Federico C. F. Calboli and Alessio Boattini <alessio.boattini2@unibo.it>

References

Lasker, G.W. 1977. A coefficient of relationship by isonymy: A method for estimating the genetic relationship between populations. Hum. Biol. 49:489-493.

See Also

sur.freq to generate the input surname frequency table from marriage data, surnames for an explanation on how to generate the correct input table from other surname sources, hedrickand uri for other types of inter-population kinship matrices

Examples

```
# starting from a raw marriage records dataset:
data(valley)
tot <- sur.freq(valley,valley$PAR,valley$SURM,valley$SURF)
tot # a frequency table calculated above all the surnames
lask.kin <- lasker(tot)
lask.kin # a kinship matrix

#starting from a generic surname frequency table
data(surnames)
surnames #a made-up dataset
# the surnames are arranged as the _rows_ and the populations are the _columns_
# the use of the function ``Lasker'' just turns this data into a kinship matrix
lask.kin <- lasker(surnames)
lask.kin
```

---

mal.cond *Calculates a Conditional Kinship matrix*

Description

The function "mal.cond" calculates a R conditional kinship matrix starting from a kinship matrix obtained by the applicatio of the Malecot migration model on a colum stochastic migration matrix.

Usage

```
mal.cond(PHI, N)
```

Arguments

PHI	PHI is a square and symmetrical kinship matrix, possibly the output of the function Phi
N	N is the effectiove population vector

Details

Much more useful than the Phi matrix, the conditional kinship R matrix is the basis for further analysis by means of Mantel tests, Procrustes rotations and cluster analysis.

Value

Returns a square symmetrical matrix.

Note

...

Author(s)

Federico C. F. Calboli <f.calboli@gmail.com>

References

Jorde, L. B. 1982. The genetic structure of the Utah mormons: migration analysis. Human Biology 54(3): 583-597.

See Also

[mal.phi](mal.phi) for the calculation of "absolute" kinship values

Examples

```
# using Swedlund data again...
data(S); data(P); data(N)
x<-mal.eq(S,P,N)
phi<-mal.phi(S,P,N,x)
cond.mat<-mal.cond(phi,N)
cond.mat
```

---

mal.eq   *Calculates the asymptotic generation for the Malecot model*

Description

Mal.eq calculates the Malecot model iteratively, stopping when one more cycle adds 0 to every value of the matrix obtained by the model. Once equilibrium is reached, Mal.eq returns the number of cycles ("generations") needed to reach it.

Usage

```
mal.eq(S, P, N)
```

Arguments

S	is the Sistematic pressure matrix.
P	is the colum-stochastic migration matrix.
N	is the vector of effective population size.

Details

The use of mal.eq is necessary before the calculation of the Malecot model proper because the value returned by Mal.eq is one of the arguments of the Malecot model function Phi.

Value

Returns one numeric value.

Note

This function has been coerced to use "only" six significant digits. ...

Author(s)

Federico C. F. Calboli <f.calboli@gmail.com>

References

Imaizumi, Y., N. E. Morton and D. E. Harris. 1970. Isolation by distance in artificial populations. Genetics 66: 569-582.

Jorde, L. B. 1982. The genetic structure of the Utah mormons: migration analysis. Human Biology 54(3): 583-597.

Swedlund, A. C., L. B. Jorde and J. H. Mielke. 1984. Population structure in the Connecticut valley. I. Marital migration. American Journal of Physical Anthropology 65: 61-70

See Also

mal.phi for the function using the output of 'mal.eq'

Examples

```
# the data is originally from a paper by Swedlund et al. 1984.
data(S); data(P); data(N)
mal.eq(S,P,N)
```

---

mal.phi   *Calculates a kinship matrix using the Malecot Migration Model*

Description

Calculates a kinship matrix using the Malecot Migration Model, in the form described by L. B. Jorde 1982.

Usage

```
mal.phi(S, P, N, n)
```

Arguments

S	the sistematic pressure matrix, where the diagonal elements are 1-sk, with sk the sistematic pressure for the k-th population, and the non diagonal elements are 0
P	the column stochastic migration matrix, possibly obtained using col.sto on the "raw" migration matrix
N	the vector of effective populations, where each element is the population size for all the n populations divided by 3
n	the number of iterations needed to reach the equilibrium, calculated by the function Mal.eq

Details

The Malecot model is simply an iterative markow-chain-like process that gives rise to an asymptotic growth curve, so that an equilibrium is reached after a number of iterations.

Value

Returns a square and symmetrical matrix.

Note

...

Author(s)

Federico C. F. Calboli <f.calboli@gmail.com>

References

Imaizumi, Y., N. E. Morton and D. E. Harris. 1970. Isolation by distance in artificial populations. Genetics 66: 569-582.

Jorde, L. B. 1982. The genetic structure of the Utah mormons: migration analysis. Human Biology 54(3): 583-597.

See Also

[mal.eq](#) for the function generating the number of cycles needed to reach the asymptotic value

Examples

```
# using Swedlund data again...
data(S); data(P); data(N)
x<-mal.eq(S,P,N)
phi<-mal.phi(S,P,N,x)
phi
```

---

mar.iso   *Observed and Random Marital Isonymy*

Description

Function "mar.iso" calculates Observed and Random Marital Isonymy starting from tables of observed couples of surnames frequencies in each (sub)population.

Usage

```
mar.iso(x)
```

Arguments

x   is a table object containing N matrices, where N is the number of analysed (sub)populations. Each matrix is a square matrix whose dimensions are equal to the total number of different surnames observed in the analysed subpopulations. Rows correspond to male surnames entries and columns to female surnames entries

Details

Marital Isonymy coefficients are obtainable starting from marriage data or equivalent data. The tables of observed couples of surnames needed as argument in "mar.iso" are easily obtainable from raw data using the "sur.freq" function selecting the "marriage" option. Observed Isonymy (Pt) is the number of isonymic marriages (i. e. marriages in which both the mates have the same surname) on the total number of marriages. Random Isonymy (Pr) is the probability that two mates have randomly the same surname and is given by: $Pr = \mathrm{sum}\,(pi * qi)$, where pi is the frequence of the i-th surname among males and qi is the frequency of the i-th surname among females.

Value

Returns a data frame reporting Observed Isonymy (Pt) and Random Isonymy (Pr) for each (sub)population (pop)

Note

The Observed Isonymy coefficient (Pt) is a measure of within (sub)population kinship. The Random Isonymy coefficient (Pr) is an unbiased measure of the expected within (sub)population kinship value in case of random marriage unions. The output of the "mar.iso" function can be used as the argument for the "sur.inbr" function to calculate Inbreeding indexes. Pr values can also be substituted to the diagonal values of the kinship between populations matrix given by the function "uri" to obtain another unbiased random kinship matrix.

Author(s)

Federico C. F. Calboli and Alessio Boattini <alessio.boattini2@unibo.it>

References

Crow, J. F., Mange, A. P. 1965. Measurement of inbreeding from the frequency of marriages between persons of the same surnames. Eugen. Q. 12:199-203. Crow, J. F. 1980. The estimation of inbreeding from isonymy. Hum. Biol. 52:1-14.

See Also

sur.freq to calculate surnames frequencies tables from raw marriages data bases, sur.inbr to calculate inbreeding coefficients starting from Pt and Pr, r.pairs to calculate Repeated Pairs indexes, uri to calculate a matrix of Unbiased Random Isonymy coefficients between (sub)populations

Examples

```
data(valley)
valley #a subset of a real marriage data base

mar <- sur.freq(valley,valley$PAR,valley$SURM,valley$SURF,freq.table="marriages")
mar # frequency tables of the observed pairs of surnames in each population

iso <- mar.iso(mar)
iso # a data frame containing Pt and Pr values for each (sub)population
```

---

mtx.exp   *Calculates the n-th power of a matrix*

Description

Calculates the n-th power of a matrix.

Usage

```
mtx.exp(X, n)
```

Arguments

X	a square matrix
n	the exponential value

Details

This function calculates (efficiently!) the n-th power of a matrix.

Value

Takes a matrix and returns a matrix.

Note

Original code by VCC "beautyfied" by MM

Author(s)

Vincente Canto Cassola and Martin Maechler

References

...

Examples

```
test<-matrix(c(1:16), 4,4)
pow.test<-mtx.exp(test,10)
pow.test
```

N *Effective population vector*

Description

A vector giving the effective population size for n populations. The effective population size is calculated as the total population divided by three.

Usage

```
data(N)
```

Format

A 12 elements vector.

Details

This data comes for Swedlund et al. 1984.

Source

Swedlund, A. C., L. B. Jorde and J. H. Mielke. 1984. Population structure in the Connecticut valley. I. Marital migration. American Journal of Physical Anthropology 65: 61-70

Examples

```
data(N)
```

P *Column stochastic migration matrix*

Description

A column stochastic migration matrix for 12 populations.

Usage

```
data(P)
```

Format

A 12 by 12 square matrix

Details

This data comes for Swedlund et al. 1984.

Source

Swedlund, A. C., L. B. Jorde and J. H. Mielke. 1984. Population structure in the Connecticut valley. I. Marital migration. American Journal of Physical Anthropology 65: 61-70

Examples

```
data(P)
```

---

r.pairs   *Observed and Random Repeated Pairs Coefficients*

Description

Function "r.pairs" calculates Observed and Random Repeated Pairs Coefficients starting from tables of observed couples of surnames frequencies in each (sub)population.

Usage

```
r.pairs(x)
```

Arguments

x   is a table object containing N matrices, where N is the number of analysed (sub)populations. Each matrix is a square matrix whose dimensions are equal to the total number of different surnames observed in the analysed subpopulations. Rows correspond to male surnames entries and columns to female surnames entries.

Details

Repeated Pairs coefficients are obtainable starting from marriage data or equivalent data. The tables of observed couples of surnames needed as argument in "rep.pairs" are easily obtainable from raw data using the "sur.freq" function selecting the "marriage" option. Observed Repeated Pairs coefficient (RP) estimate the level of homozigosity in a (sub)population on the basis of repeated appearences of couples of identical surnames. Random Repeated Pairs coefficient (RPr) is the expected RP value in case of completely random marriage unions. Comparisons between RP and RPr are expressed with their percentage difference (perc.diff) given by (RP-RPr)/RPr.

Value

Returns a data frame reporting Observed Repeated Pairs (RP), Random Repeated Pairs (RPr) and the Percentual difference between RP and RPr (perc. diff) for each (sub)population (pop).

Note

RP and RPr are standardized indexes and their values vary between 0 and 1. RP, being calculated using the whole surname matrix, is considered a more reliable source of information on the level of homozigosity in a population than Isonymy data. An excess of RP on RPr, as calculated by their percentage difference, suggests the existence of a degree of subdvision internal to the analysed (sub)population.

Author(s)

Federico C. F. Calboli and Alessio Boattini <alessio.boattini2@unibo.it>

References

Lasker G. W., Kaplan B. A. 1985. Surnames and genetic structure: repetition of the same pairs of names of married couples, a measure of subdivision of the population. Hum. Biol. 57:431-440. Chakraborty R. 1985. A note on the calculation of random RP and its sampling variance. Hum. Biol. 57:713-717. Chakraborty R. 1986. Erratum. Hum. Biol. 58:991.

See Also

sur.freq to calculate surnames frequencies tables from raw marriages data bases, mar.iso to calculate Observed and Random Isonymy coefficients starting from tables of couples of surnames frequencies, sur.inbr to calculate Inbreeding indexes from Isonymy coefficients

Examples

```
data(valley)
valley # a subset of a real marriage data base

mar <- sur.freq(valley,valley$PAR,valley$SURM,valley$SURF,freq.table="marriages")
mar # frequency tables of the observed pairs of surnames in each population

RP <- r.pairs(mar)
RP # a data frame containing RP, RPr perc.diff values for each (sub)population
```

---

raw.mig   *A raw migration data*

Description

Made up raw dataset, created as if count data for marital migration were put into matrix form.

Usage

```
data(raw.mig)
```

Format

A 4 by 4 square matrix.

Details

Completely made up for pedagogical purposes.

Examples

```
data(raw.mig)
col.sto(raw.mig)
```

---

rel.cond   *Calculates a conditional kinship matrix from isonymy data*

Description

"rel.cond" calculates a conditional kinship matrix starting from isonymy data.

Usage

```
rel.cond(x,R, method="A")
```

Arguments

x	is a square Unbiased Random Isonymy matrix, possibly obtained using the "uri" function on the raw surname data
R	is an unbiased estimate of Regional Random Isonymy, calculated by the function "rri"
method	a character string specifying the method to be used in the calculation of the coefficients. The available options are "A" and "B". Both the methods give similar results. The "A" method is given as the default option

Details

The function implements Relethford's method to calculate kinship coefficients starting from surname data.

Value

Returns a square symmetric conditional kinship matrix.

Note

The term 'conditional kinship' refers to kinship relative to the contemporary region

Author(s)

Federico C. F. Calboli and Alessio Boattini <alessio.boattini2@unibo.it>

References

Relethford, J. H. 1988. Estimation of kinship and genetic distance from surnames. Human Biology, 60(3): 475-492.

See Also

uri to calculate Unbiased Random Isonymy starting from tables of surname frequencies, rri to calculate an an unbiased estimate of Regional Random Isonymy, rel.phi to calculate an 'a priori' kinship matrix from isonymy data

Examples

```
# starting from a raw marriage records dataset:
data(valley)
tot <- sur.freq(valley,valley$PAR,valley$SURM,valley$SURF)
tot # a frequency table calculated above all the surnames
iso.matrix <- uri(tot)
iso.matrix # an unbiased random isonymy matrix
reg <- rri(tot)
reg # a coefficient of unbiased Regional Random Isonymy
kin.cond <- rel.cond(iso.matrix,reg)
kin.cond # a conditional kinship matrix

#starting from a generic surname frequency table
data(surnames)
surnames # a made-up dataset
iso.matrix <- uri(surnames)
iso.matrix # an unbiased random isonymy matrix
reg <- rri(surnames)
reg # a coefficient of unbiased Regional Random Isonymy
kin.cond <- rel.cond(iso.matrix,reg)
kin.cond # a conditional kinship matrix
```

---

rel.phi *Calculates an 'a priori' kinship matrix from isonymy data*

Description

"rel.phi" calculates an 'a priori' kinship matrix starting from isonymy data.

Usage

```
rel.phi(x,R, method="A")
```

Arguments

x	is a square Unbiased Random Isonymy matrix, possibly obtained using the "uri" function on the raw surname data
R	is an unbiased estimate of Regional Random Isonymy, calculated by the function "rri"
method	a character string specifying the method to be used in the calculation of the coefficients. The available options are "A" and "B". Both the methods give similar results. The "A" method is given as the default option

Details

The function implements Relethford's method to calculate kinship coefficients starting from surname data.

Value

Returns a square symmetric 'a priori' kinship matrix.

Note

The term 'a priori kinship' refers to kinship relative to a founding population

Author(s)

Federico C. F. Calboli and Alessio Boattini <alessio.boattini2@unibo.it>

References

Relethford, J. H. 1988. Estimation of kinship and genetic distance from surnames. Human Biology, 60(3): 475-492.

See Also

[uri](#) to calculate Unbiased Random Isonymy starting from tables of surname frequencies, [rri](#) to calculate an an unbiased estimate of Regional Random Isonymy, [rel.cond](#) to calculate a conditional kinship matrix from isonymy data

Examples

```
# starting from a raw marriage records dataset:
data(valley)
tot <- sur.freq(valley,valley$PAR,valley$SURM,valley$SURF)
tot # a frequency table calculated above all the surnames
iso.matrix <- uri(tot)
iso.matrix # an unbiased random isonymy matrix
reg <- rri(tot)
```

```
reg # a coefficient of unbiased Regional Random Isonymy
kin <- rel.phi(iso.matrix,reg)
kin # an 'a priori' kinship matrix

#starting from a generic surname frequency table
data(surnames)
surnames # a made-up dataset
iso.matrix <- uri(surnames)
iso.matrix # an unbiased random isonymy matrix
reg <- rri(surnames)
reg # a coefficient of unbiased Regional Random Isonymy
kin <- rel.phi(iso.matrix,reg)
kin # an 'a priori' kinship matrix
```

| rri | *Calculates an unbiased estimate of Regional Random Isonymy* |

Description

"rri" calculates an unbiased estimate of Regional Random Isonymy starting from surname frequencies.

Usage

```
rri(x)
```

Arguments

x   is a surname frequency table where the N rows correspond to the surnames
                      present in the whole population and the M columns are the (sub)populations

Details

The function implements Morton's isonymy method as outlined by Relethford. Unbiased estimate of Regional Random Isonymy refers to random isonymy of the contemporary region relative to the founding population. This value is an argument needed to calculate 'a priori' and conditional kinship matrices using the "rel.phi" and "rel.cond" functions.

Value

Returns one numeric value.

Note

The use of "rri" could be problematic, because different people are likely to arrange isonymy data in different ways on their computers. We decided for a matrix format for the isonymy data; the function would originally accept data in a different format and then convert it internally, but this would be a problem for people with data arranged in a different format. In the end we decided to write a specific function, "sur.freq", to generate surname frequency tables directly from raw

marriage data or marriage-like data (the most commonly used sources in bio-demographic studies). For other types of surname data, see the verbose explanation in the info for the dataset "surnames" so it would be clear for the user how "rri" works.

Author(s)

Federico C. F. Calboli and Alessio Boattini <alessio.boattini2@unibo.it>

References

Morton, N. E. 1973. Kinship bioassy. In: Genetic distance, J. F Crow and C Denniston (eds.). New York, Plenum Press, 97-104. Relethford, J. H. 1988. Estimation of kinship and genetic distance from surnames. Human Biology, 60(3): 475-492.

See Also

sur.freq to generate the input surname frequency table from marriage data, surnames for an explanation on how to generate the correct input table from other surname sources, uri to calculate an Unbiased Random Isonymy matrix, rel.phi to calculate an 'a priori' kinship matrix from isonymy data, rel.cond to calculate a conditional kinship matrix from isonymy data

Examples

```
# starting from a raw marriage records dataset:
data(valley)
tot <- sur.freq(valley,valley$PAR,valley$SURM,valley$SURF)
tot # a frequency table calculated above all the surnames
reg <- rri(tot)
reg # an unbiased estimate or Regional Random Isonymy

#starting from a generic surname frequency table
data(surnames)
surnames # a made-up dataset
# you can see that the surnames are arranged as the _rows_
# and the populations are the _columns_
reg <- rri(surnames)
reg # an unbiased estimate or Regional Random Isonymy
```

S   *Systematic pressure matrix*

Description

Systematic pressure matrix obtained by creating a square matix, where the non diagonal elements are all 0 and the diagonal elements are calculated as 1-Sk, where Sk is the systematic pressure for the k-th population.

Usage

```
data(S)
```

Format

A 12 by 12 square matrix.

Details

This data comes for Swedlund et al. 1984.

Source

Swedlund, A. C., L. B. Jorde and J. H. Mielke. 1984. Population structure in the Connecticut valley. I. Marital migration. American Journal of Physical Anthropology 65: 61-70

Examples

```
data(S)
```

---

sur.freq   *Calculates surnames frequency tables*

Description

"sur.freq"calculates surnames frequency tables starting from raw marriage data or equivalent sources (i.e. birth registrations)

Usage

```
sur.freq(x,pop,mal.sur,fem.sur,freq.table="total")
```

Arguments

x	is a data frame in which every row corresponds to a different marriage record. The data frame must contain: **a** a column reporting the population in which the marriage was recorded; **b** a column containing male surnames; **c** a column containing female surnames
pop	is the name of the column in the data frame that reports the population in which the marriage was recorded
mal.sur	is the name of the column in the data frame that contains male surnames
fem.sur	is the name of the column in the data frame that comtains female surnames
freq.table	character string specifying the type of surname frequency table to be calculated. The available options are: "males" (table calculated using only male surnames); "females" (table calculated using only female surnames); "total" (table calculated using all the surnames); "marriages" (tables calculated using observed pairs of surnames in each population). The default option is "total".

Details

"sur.freq" is specifically written to derive surname frequency tables from marriage data, or, more generally, data in which appear couples of related surnames, as birth records etc.

Value

A single table of surname frequencies ("male", "female", "total" options) or tables of observed pairs of surnames frequencies for each population ("marriages" option)

Note

Surname frequency tables produced with "sur.freq" are intended to be used as an argument for other functions to investigate the bio-demographic structure of populations. In particular, the "male", "female" and "total" options produce tables to be used in inter-population analyses (maesures of kinship/distance between populations, etc.); the "marriage" option produces tables to be used in intra-population analyses (inbreeding levels etc.). Tables of surname frequencies can also be obtained from simple lists of surnames (i.e. telephone directories, etc.) using the function "table"; for further explanations see the info for the "surnames" data set.

Author(s)

Federico C. F. Calboli and Alessio Boattini <alessio.boattini2@unibo.it>

References

Lasker, G. W. 1985. Surnames and genetic structure. Cambridge University Press. Cambridge, England

See Also

mar.iso for the calculation of Marital Isonymy coefficients from tables of observed pairs of surnames frequencies, r.pairs fot the calculation of Repeated Pairs coefficients from tables of observed pairs of surnames frequencies, lasker and hedrick for the calculation of similarity indexes between populations from surnames frequency tables, surnames for an explanation on how to generate a surname frequency table starting from non-marriage like data

Examples

```
data(valley)
valley #a subset of a real marriage data base

# you can see that marriages correspond to rows in the data frame.
# Note that the data frame contains other columns

tot <- sur.freq(valley,valley$PAR,valley$SURM,valley$SURF)
tot # a frequency table calculated above all the surnames
mal <- sur.freq(valley,valley$PAR,valley$SURM,valley$SURF,freq.table="males")
mal # a frequency table calculated above the male surnames
fem <- sur.freq(valley,valley$PAR,valley$SURM,valley$SURF,freq.table="females")
fem # a frequency table calculated above the female surnames
```

```
mar <- sur.freq(valley,valley$PAR,valley$SURM,valley$SURF,freq.table="marriages")
mar # frequency tables for the observed pairs of surnames in each population
```

---

sur.inbr   *Total, Random and Non-random Inbreeding Coefficients*

Description

Function "sur.inbr" calculates Total, Random and Non-random Inbreeding Coefficients starting
from Observed and Random Isonymy indexes in each (sub)population.

Usage

```
sur.inbr(x,method="B")
```

Arguments

x   is a data frame composed by 3 columns which, in order, contain: a code (or a
                name) identifying the analysed (sub)populations; Observed Isonymy (Pt) values;
                Random Isonymy (Pr) values. The number of rows is equal to the total number
                of analysed (sub)populations.

method   character string specifying the method to be used in the calculation of the in-
                dexes. The available options are "A" and "B". Both the methods give similar
                results. The "B" method, being the most frequently used in the studies, is given
                as the default option

Details

Inbreeding coefficients are obtainable starting from Isonymy data. The data frame containing Ob-
served and Random Isomymy for each (sub)population needed as argument in "sur.inbr" is easily
obtainable using the "mar.iso" function. Inbreeding coefficients allow an estimate of the inbreeding
level in a (sub)population on the basis of couples of surnames.

Value

Returns a data frame reporting Total Inbreeding (Ft), Random Inbreeding (Fr) and Non-random
Inbreeding (Fn) for each (sub)population (pop)

Note

Total Inbreeding (Ft) is an estimate of the inbreeding level in a (sub)population. Random Inbreeding
(Fr) is the expected inbreeding level in a (sub)population in case of completely random marriage
unions. Non-random Inbreeding (Fn) expresses the deviance between Ft and Fr: positive Fn values
show preference towards unions between consanguineous mates, negative Fn values show aversion
towards unions between consanguineous mates.

Author(s)

Federico C. F. Calboli and Alessio Boattini <alessio.boattini2@unibo.it>

References

Crow, J. F., Mange, A. P. 1965. Measurement of inbreeding from the frequency of marriages between persons of the same surnames. Eugen. Q. 12:199-203. Crow, J. F. 1980. The estimation of inbreeding from isonymy. Hum. Biol. 52:1-14.

See Also

[sur.freq](#) to calculate surnames frequencies tables from raw marriages data bases, [mar.iso](#) to calculate Observed and Random Isonymy coefficients starting from tables of couples of surnames frequencies, [r.pairs](#) to calculate Repeated Pairs indexes

Examples

```
data(valley)
valley # a subset of a real marriage data base

mar <- sur.freq(valley,valley$PAR,valley$SURM,valley$SURF,freq.table="marriages")
mar # frequency tables calculated above the observed pairs of surnames in each population

iso <- mar.iso(mar)
iso # a data frame containing Pt and Pr values for each (sub)population

inbreeding <- sur.inbr(iso)
inbreeding # inbreeding indexes calculated using the method "B"

inbreeding2 <- sur.inbr(iso,method="A")
inbreeding2 # inbreeding indexes calculated using the method "A"
```

---

surnames   *Surname frequency table*

Description

A surname frequency table, with 3 populations and 5 surnames.

Usage

```
data(surnames)
```

Format

A 5 rows by 3 columns dataset

Details

Surname frequency tables are the argument needed in all the surname-based inter-population analysis functions (e.g. "lasker", "hedrick", "uri", etc.). Surname frequency tables can be generated from marriage and marriage-like data (e.g. data that contain couples of related surnames) using the "sur.freq" function. In order to generate surname frequency tables from other surname sources (e.g. telephone directories, registers of voters, etc.) see the example in this help page. To import correctly surnames data bases in R see the "valley" dataset help page.

Source

Alessio Boattini. Dummy dataset generated for testing and example purposes

Examples

```
data(surnames)

# NB. How did we produce the "surnames" dataset?
# the original data (an hypothetic list of surnames)
# were arranged as:

#  YEAR POP SURNAME
#  1901   3  FABBRI
#  1901   3  VITALI
#  1901   2   LIPPI
#  1901   2  FABBRI
#  1901   2   NARDI
#  1901   2   NARDI
#  1901   1  ANGELI
#  1902   1  ANGELI
#  1902   2  VITALI
#  1902   2   LIPPI
#  1902   1   LIPPI
#  1902   1   LIPPI
#  1902   3  VITALI
#  1902   3  FABBRI
#  1902   2  FABBRI
#  1904   2   NARDI
#  1904   2   NARDI
#  1904   2   LIPPI
#  1905   1  VITALI
#  1905   1  FABBRI
#  1905   3  FABBRI
#  1905   3  ANGELI
#  1905   2   LIPPI
#  1905   2   NARDI
#  1905   3   NARDI
#  1905   3   NARDI

#       ..........

# This arrangement does not necessarily reflect
```

```
# the way other people would arrange their data.
# The "surnames" dataset was generated using
# the "table" function as follows:

# table(data$SURNAME,data$POP)
```

---

sym.P   *Calculate the symmetric column stochastic matrix*

Description

Used to turn the asymmetric column stochastic matrix into a symmetric column stochastic matric.

Usage

```
sym.P(x)
```

Arguments

x   x is a column stochastic matrix

Details

The function calculates the symmetric matrix from the asymmetric column stochastic matrix, leaving the diagonal unchanged and averaging m[i,j] and m[j,i] as (m[i,j]+m[j,i])/2. The computed average substitutes each pair of values in the new symmetric column stochastic matrix.

Value

Returns a matrix.

Note

...

Author(s)

Federico C. F. Calboli <f.calboli@gmail.com>

References

Jorde, L. B. 1982. The genetic structure of the Utah mormons: migration analysis. Human Biology 54(3): 583-597.

Examples

```
data(P)
symmetric<-sym.P(P)
```

---

uri   *Calculates the Unbiased Random Isonymy matrix*

Description

"uri" calculates the unbiased random isonymy coefficient starting from surname frequencies.

Usage

```
uri(x)
```

Arguments

x   is a surname frequency table where the N rows correspond to the surnames present in the whole population and the M columns are the (sub)populations

Details

The function implements Morton's isonymy method as outlined by Relethford. Unbiased estimations of intra-(sub)population isonymy were included. Unbiased Random Isonymy is an argument needed to calculate 'a priori' and conditional kinship matrices using the "rel.phi" and "rel.cond" functions.

Value

Returns a square symmetric unbiased isonymy matrix.

Note

The use of "uri" could be problematic, because different people are likely to arrange isonymy data in different ways on their computers. We decided for a matrix format for the isonymy data; the function would originally accept data in a different format and then convert it internally, but this would be a problem for people with data arranged in a different format. In the end we decided to write a specific function, "sur.freq", to generate surname frequency tables directly from raw marriage data or marriage-like data (the most commonly used sources in bio-demographic studies). For other types of surname data, see the verbose explanation in the info for the dataset "surnames" so it would be clear for the user how "uri" works.

Author(s)

Federico C. F. Calboli and Alessio Boattini <alessio.boattini2@unibo.it>

References

Morton, N. E. 1973. Kinship bioassy. In: Genetic distance, J. F Crow and C Denniston (eds.). New York, Plenum Press, 97-104. Relethford, J. H. 1988. Estimation of kinship and genetic distance from surnames. Human Biology, 60(3): 475-492.

See Also

sur.freq to generate the input surname frequency table from marriage data, surnames for an expla-
nation on how to generate the correct input table from other surname sources, lasker for a similar
kinship coefficient derived from surnames, hedrick for a standardized kinship coefficient derived
from surnames, rri to calculate an unbiased estimate of Regional Random Isonymy from surn-
maes rel.phi to calculate an 'a priori' kinship matrix from isonymy data, rel.cond to calculate a
conditional kinship matrix from isonymy data

Examples

```
# starting from a raw marriage records dataset:
data(valley)
tot <- sur.freq(valley,valley$PAR,valley$SURM,valley$SURF)
tot # a frequency table calculated above all the surnames
iso.matrix <- uri(tot)
iso.matrix # a unbiased random isonymy matrix

#starting from a generic surname frequency table
data(surnames)
surnames # a made-up dataset
# you can see that the surnames are arranged as the _rows_
# and the populations are the _columns_
# the function "uri" turns this data into a unbiased random isonymy matrix
iso.matrix <- uri(surnames)
iso.matrix
```

---

valley   *Raw marriage data*

Description

A raw marriage data set, in which every row corresponds to a different marriage record.

Usage

```
data(valley)
```

Format

A 702 rows by 8 columns dataset. The columns of "valley" contain the following information:

PAR:	(sub)population to which the marriage is referred.
YEAR:	year in which the marriage was performed.
SURM:	male surname.
NM:	male birth (sub)population.
RM:	male residence (sub)population.
SURF:	female surname.
NF:	female birth (sub)population.
RF:	female residence (sub)population.

For all columns the letter "X" indicates that the mate was born or resident outside of the study area.

Details

Marriage data, depending from the used sources, may contain more or less information than the "valley" example data set. Columns order in the dataset is not relevant. Information on (sub)population, male and female surnames are needed to perform surname-based analyses on marriage data. NB. Information on mates birthplace (or equivalent data) can be used to produce a migration matrix (see the "raw.mig" dataset). NB2. Given that surnames may contain spaces (e.g. "DE IORIO"), the best way to import surname data is to save the original data base as a .csv file, and then use the read.csv() or read.csv2() functions. Another option is to use GNUMERIC, because it has a text export feature that allows to put brakets ("") to the left and right of every cell content, so that composite surnames are read as a string. The resulting text file is easily imported by read.table().

Source

Paola Gueresi. Subset of a real marriage dataset

Examples

```
data(valley)
```

Index