pestim: a hierarchical model to estimate population counts with aggregated mobile phone data.

**Description**

This package provides an implementation for a hierarchical model to combine both aggregated mobile phone data and external official (administrative or survey) data to produce estimates of population counts in each cell of a division of a territory.

**Context**

This package has been developed in the context of a European research project within the European Statistical System called [ESSnet on Big Data](https://webgate.ec.europa.eu/fpfis/mwikis/essnetbigdata/index.php/Main_Page). More specifically this work corresponds to the work package on mobile phone data by which we assess the use of this data source in the production of official statistics. The goals of the project is many-fold. Firstly, the issue of accessing these data for the production of official statistics initially for research and then for standard production has been investigated. Secondly, in a hands-on bottom-up approach, we make some initial methodological proposals to produce concrete statistical output using those data sets compiled in the preceding phase. Thirdly, in parallel, IT tools, architecture and software development are assessed especially in contrast to traditional computer frameworks. Finally, quality is appraised especially in the context of the European Statistics Code of Practice and ESS Quality Assurance Framework. This package provides a first-step implementation of software routines to present a proof of concept about a methodological proposal (see below) to make inferences about a target population from a mobile phone dataset.

**The hierarchical model in a nutshell**

The methodological proposal giving rise to this package focuses on the inference exercise connecting aggregated mobile phone data with a target population under analysis. In concrete, the goal is to provide estimates of population counts in each cell in which we have divided the territory for which the telecommunication network provides count data. The estimation is assisted with official data at a larger time scale (either from a population register or from a survey).

The model rests on two working assumptions:

* Given that mobile phone data and official data operate at different time scales, we assume that there exists an initial time instant in which we can equate population figures from both sources.
* The mobility patterns of individuals do not depend on the mobile network operator which they are subscribed to.

The model works in two stages. Firstly at the initial time instant, we use data from both sources to make the inference for the actual population counts in each cell. Secondly, the time evolution of these counts are produced using the transition matrices from cell to cell of individuals provided by the mobile network operator.

The essence of the model is to emulate the ecological sampling setting in which the number of detected individuals in each cell follows a binomial distribution*Bin(N\_{i}, p\_{i})* whose parameter *N\_{i}* is the target of the model and is assigned a weakly informative prior and the detection probability is also assigned a weakly informative prior based upon both data sources.

**Computational paradigm**

Computations are conducted following the Bayesian paradigm. In this sense the generation of simulated populations according to different probability distributions is at the core of the package. In this sense the package contains basically three types of functions:

* Auxiliary functions, providing computation of mathematical functions such as the ratio of two beta functions, the confluent hypergeometric function, an optimization routine for a concrete probability distribution, etc. Examples of these functions are [ratioBeta](http://127.0.0.1:9775/help/library/pestim/help/ratioBeta), [kummer](http://127.0.0.1:9775/help/library/pestim/help/kummer), [Phi](http://127.0.0.1:9775/help/library/pestim/help/Phi), [modeLambda](http://127.0.0.1:9775/help/library/pestim/help/modeLambda).
* Distribution-relation functions, providing computation regarding the generation of random deviates according to different probability distributions comprising both priors, posteriors, and the generation of parameter specifications for these distributions. Examples of these functions are [dtriang](http://127.0.0.1:9775/help/library/pestim/help/dtriang),[rtriang](http://127.0.0.1:9775/help/library/pestim/help/rtriang), [ptriang](http://127.0.0.1:9775/help/library/pestim/help/ptriang), [qtriang](http://127.0.0.1:9775/help/library/pestim/help/qtriang), [dlambda](http://127.0.0.1:9775/help/library/pestim/help/dlambda), [rlambda](http://127.0.0.1:9775/help/library/pestim/help/rlambda), [rmatProb](http://127.0.0.1:9775/help/library/pestim/help/rmatProb), [rN0](http://127.0.0.1:9775/help/library/pestim/help/rN0), [rNt](http://127.0.0.1:9775/help/library/pestim/help/rNt), [rNtcondN0](http://127.0.0.1:9775/help/library/pestim/help/rNtcondN0), [rg](http://127.0.0.1:9775/help/library/pestim/help/rg), [rp](http://127.0.0.1:9775/help/library/pestim/help/rp), [alphaPrior](http://127.0.0.1:9775/help/library/pestim/help/alphaPrior), [genAlpha](http://127.0.0.1:9775/help/library/pestim/help/genAlpha), [genUV](http://127.0.0.1:9775/help/library/pestim/help/genUV).
* Estimation-relation functions, providing computation of estimates based upon the populations generated with the preceding functions. Examples of these functions are [postN0](http://127.0.0.1:9775/help/library/pestim/help/postN0), [postNt](http://127.0.0.1:9775/help/library/pestim/help/postNt), [postNtcondN0](http://127.0.0.1:9775/help/library/pestim/help/postNtcondN0).

|  |  |
| --- | --- |
| alphaPrior {pestim} | R Documentation |

Generate prior distributions for parameters of the Dirichlet distribution.

**Description**

Generate a list of prior distributions for the parameters of the Dirichlet distribution in the hierarchical model. Each component of the list corresponds to the prior distribution of the parameter *α\_{ij}(t\_{0}, t\_{n})* for each cell *j*. This function initial works over a fixed initial cell *i*. Each returned distribution is specified as a list with an identification name as first component and named components with the distribution parameters for the rest of components.

**Usage**

alphaPrior(nMNOfrom, names, variation)

**Arguments**

|  |  |
| --- | --- |
| nMNOfrom | numeric vector with the number of individuals moving from the initial cell to the rest of cells (including those remaining) |
| names | character vector with the names of the prior distributions for each cell |
| variation | list of lists whose components are parameters providing a measure of variation of each prior distribution |

**Details**

The function takes the number of cells from the input parameter nMNOfrom which specifies the number of individuals detected by the network moving from the initial cell to each of the cells (including those remaining in the same). The function executes the same construction for each final cell. It takes the name of prior distribution from the input parameter names and construct the corresponding prior distribution for each cell *j* with mode at *u\_{j}^{\*}=N\_{j}*, where *N\_{j}* is taken from nMNOfrom. Next the rest of parameters of the distribution are computed according to the dispersion parameters specified in variation.

As accepted distribution names, currently the user can specify unif, degen, triang, and gamma.

The dispersion parameters recognised so far are the coefficients of variation only (standard deviation divided by the mean of the distribution). These dispersion parameters must be specified by a named component cv with a numeric value in *[0, 1]*.

For each distribution the parameters are computed as follows:

* unif: This is the uniform distribution with parameters xMax and xMin. Both parameters are computed by *u\_{j}^{\*}\cdot(1\pm√{3}\textrm{cv})*, respectively, in each cell *j*.
* degen: This is the degenerate distribution with parameter X0 taken as *u\_{j}^{\*}* in each cell *j*.
* triang: This is the triangular distribution [dtriang](http://127.0.0.1:9775/help/library/pestim/help/dtriang) with parameters xMax, xMin, and xMode. The latter is taken directly from nMNOfrom. The distribution is assumed to be symmetrical so that the two former parameters are computed by *u\_{j}^{\*}\cdot(1\pm√{3}\textrm{cv})*, respectively, in each cell *j*.
* gamma: This is the gamma distribution with parameters shape and scale. The former is computed as *\frac{1}{\textrm{cv}^2}* and the latter as*frac{u\_{j}^{\*}}{\textrm{scale} - 1}*.

**Value**

Return a list with a list in each component specifying the prior for each cell

**Examples**

# Three cells. Cell 1 under study. 10 individuals remain.

alphaPrior(c(10, 3, 4), c('unif', 'triang', 'gamma'),

list(list(cv = 0.1), list(cv = 0.05), list(cv = 0.15)))

|  |  |
| --- | --- |
| dg {pestim} | R Documentation |

## Density function of a candidate distribution in the accept-reject method.

### Description

Generate values of a candidate distribution density function in the accept-reject method of generation of random variables applied to the distribution of the lambda parameter

### Usage

dg(lambda, nMNO, nReg, fu, fv, flambda, relTol = 1e-06, nSim = 10000,

nStrata = c(1, 100), verbose = FALSE)

### Arguments

|  |  |
| --- | --- |
| lambda | numeric vector with the lambda parameter values |
| nMNO, | nReg non-negative integer vectors with the number of individuals detected in each cell according to the network operator and the register |
| fu, | fv named lists with the prior marginal distributions of the two-dimensional points for the Monte Carlo integration |
| flambda | named list with the prior distribution of the lambda parameter |
| relTol | relative tolerance in the computation of the [kummer](http://127.0.0.1:9775/help/library/pestim/help/kummer) function. Default value is 1e-6 |
| nSim | number of two-dimensional points to generate to compute the integral. Default value is 1e4 |
| nStrata | integer vector of length 2 with the number of strata in each dimension. Default value is c(1, 1e2) |
| verbose | logical (default FALSE) to report progress of the computation |

### Details

The candidate distribution is a gamma distribution with parameters shape = nMNO + 1 and scale = *λ^{\*}* / nMNO, where *λ^{\*}* stands for the mode of the posterior distribution of the lambda parameter.

It is important to know that currently this function accepts only parameters for a single cell at a time. In case of interest for the candidate density function values for a set of cells, the user should program his/her own routine to apply this function to every cell.

### Value

dg generates length(lambda) values of the density probability function of the candidate distribution in the accept-reject method.

### See Also

[modeLambda](http://127.0.0.1:9775/help/library/pestim/help/modeLambda), [dlambda](http://127.0.0.1:9775/help/library/pestim/help/dlambda) for related functions.

### Examples

curve(dg(x, nMNO = 20, nReg = 115, fu = list('unif', xMin = 0.3, xMax = 0.5),

fv = list('unif', xMin = 100, xMax = 120),

flambda = list('gamma', shape = 11, scale = 12)), xlim = c(0, 150),

main = '', ylab = 'density', xlab = 'lambda')

|  |  |
| --- | --- |
| dlambda {pestim} | R Documentation |

Posterior density function of the lambda parameter.

**Description**

Compute the unnormalized posterior density function of the parameter *λ* in the hierarchical model to estimate population counts given by

*f(λ\big | N^{\textrm{MNO}}; N^{\textrm{Nreg}})\propto f(λ)\cdot \textrm{dpois}(N^{\textrm{MNO}}; λ)\cdot S(λ; N^{\textrm{MNO}}, N^{\textrm{Nreg}}),*

where [dpois](http://127.0.0.1:9775/help/library/pestim/help/dpois) is the probability density function of a Poisson distribution and *S* is defined in the bibliographic reference.

**Usage**

dlambda(lambda, nMNO, nReg, fu, fv, flambda, relTol = 1e-06, nSim = 1000,

nStrata = c(1, 100), verbose = FALSE)

**Arguments**

|  |  |
| --- | --- |
| lambda | numeric vector |
| nMNO, | nReg non-negative integer vectors with the number of individuals detected in each cell according to the network operator and the register |
| fu, | fv named lists with the prior marginal distributions of the two-dimensional points for the Monte Carlo integration |
| flambda | named list with the prior distribution of the lambda parameter |
| relTol | relative tolerance in the computation of the [kummer](http://127.0.0.1:9775/help/library/pestim/help/kummer) function. Default value is 1e-6 |
| nSim | number of two-dimensional points to generate to compute the integral. Default value is 1e3 |
| nStrata | integer vector of length 2 with the number of strata in each dimension. Default value is c(1, 1e2) |
| verbose | logical (default FALSE) to report progress of the computation |

**Details**

The lengths of the input vectors nMNO and nReg must be both equal to 1 and independent of the length of the input vector lambda. The integral is computed using with Monte Carlo techniques using nSim points for each of the values lambda specified so that the final [data.table](http://127.0.0.1:9775/help/library/pestim/help/data.table-class) has length(lambda) rows.

The prior distributions are specified as named lists where the first component of each list must be the name of distribution ('unif', 'triang', 'degen', 'gamma') and the rest components must be named according to the name of the parameters of the random generator of the corresponding distribution according to:

* unif: xMin, xMax for the minimum, maximum of the sampled interval.
* degen: x0 for the degenerate value of the random variable.
* triang: xMin, xMax, xMode for minimum, maximum and mode (see [qtriang](http://127.0.0.1:9775/help/library/pestim/help/qtriang)).
* gamma: scale and shape with the same meaning as in [rgamma](http://127.0.0.1:9775/help/library/pestim/help/rgamma).

It is important to know that currently this function accepts only parameters for a single cell at a time. In case of interest for the density function values for a set of cells, the user should program his/her own routine to apply this function to every cell.

**Value**

dlambda returns a [data.table](http://127.0.0.1:9775/help/library/pestim/help/data.table-class) with the values of the density function (column probLambda) for each value of lambda together with additional variables:

* The common length of nMNO and nReg identifies the number of territorial cells in which the number of individuals detected by the telecommunication network and official data. The column cellID identifies these territorial cells.
* The length of lambda identifies the number of parameters upon which the integral will be computed in each cell. The column parID identifies each of these input parameters.
* The inputs nMNO and nReg are also included in the output [data.table](http://127.0.0.1:9775/help/library/pestim/help/data.table-class) in columns under the same name.
* The value on the integral times the Poisson density function ifalso included under the column integral

**References**

<https://github.com/MobilePhoneESSnetBigData>

**See Also**

[genUV](http://127.0.0.1:9775/help/library/pestim/help/genUV), [Phi](http://127.0.0.1:9775/help/library/pestim/help/Phi) for related functions.

**Examples**

# This data.table must have 5x3= 15 rows

dlambda(seq(0, 1, length.out = 5),

nMNO = c(20, 17, 25), nReg = c(115, 123, 119),

fu = list('unif', xMin = 0.3, xMax = 0.5), fv = list('gamma', shape = 11, scale = 12),

flambda = list('gamma', shape = 11, scale = 12))

# Easily, a function to draw conditioned on the parameters:

f <- function(x){

dlambda(x, nMNO = 20, nReg = 115,

fu = list('unif', xMin = 0.3, xMax = 0.5), fv = list('unif', xMin = 100, xMax = 120),

flambda = list('gamma', shape = 11, scale = 12))$probLambda

}

curve(f, xlim = c(0, 150))

|  |  |
| --- | --- |
| triang {pestim} | R Documentation |

## The Triangular Distribution.

### Description

Density, distribution funtion, quantile function and random generation for the triangular distribution

### Usage

dtriang(x, xMin, xMax, xMode)

ptriang(q, xMin, xMax, xMode)

qtriang(q, xMin, xMax, xMode)

rtriang(n, xMin, xMax, xMode)

### Arguments

|  |  |
| --- | --- |
| x, | q vector of quantiles |
| xMin | vector with the minimum values of the range of the random variable |
| xMax | vector with the maximum values of the range of the random variable |
| xMode | vector with the modes of the random variable |
| n | number of observations |
| p | vector pf probabilities |

### Value

dtriang gives the density, ptriang gives the distribution function, qtriang gives the quantile function, and rtriang generates random deviates.

The lengths of the input vectors (except n) must be all equal except when their length is 1. Otherwise NAs are produced.

### See Also

[Distributions](http://127.0.0.1:9775/help/library/pestim/help/Distributions) for other distributions

### Examples

curve(dtriang(x, 0, 3, 1), xlim = c(0, 3))

curve(ptriang(x, 0, 3, 1), xlim = c(0, 3))

curve(qtriang(x, 0, 3, 1), xlim = c(0, 1))

hist(rtriang(1e6, 0, 3, 1), breaks = seq(0, 3, by = 0.01))

|  |  |
| --- | --- |
| genAlpha {pestim} | R Documentation |

Generate values for the parameters of the Dirichlet distribution.

**Description**

Generate a matrix of values of the parameters *α\_{ij}(t\_{0}, t\_{n})* of the Dirichlet distribution in the hierarchical model. This function initial works over a fixed initial cell *i* under study.

**Usage**

genAlpha(nSim, flist)

**Arguments**

|  |  |
| --- | --- |
| nSim | number of values to generate |
| flist | list with the prior distributions for each cell |

**Details**

This function generates the nSim random values according to the prior of each cell specified in flist.

The prior distributions are specified as named lists where the first component of each list must be the name of distribution ('unif', 'triang', 'degen', 'gamma') and the rest of components must be named according to the name of the parameters of the random generator of the corresponding distribution according to:

* unif: xMin, xMax for the minimum, maximum of the sampled interval.
* degen: x0 for the degenerate value of the random variable.
* triang: xMin, xMax, xMode for minimum, maximum and mode (see [qtriang](http://127.0.0.1:9775/help/library/pestim/help/qtriang)).
* gamma: scale and shape with the same meaning as in [rgamma](http://127.0.0.1:9775/help/library/pestim/help/rgamma).

**Value**

Return a matrix with as many columns as cells and as many rows as number of generated values

**Examples**

priors <- alphaPrior(c(10, 3, 4), c('unif', 'triang', 'gamma'),

list(list(cv = 0.1), list(cv = 0.05), list(cv = 0.15)))

genAlpha(10, priors)

|  |  |
| --- | --- |
| genUV {pestim} | R Documentation |

Generation of two-dimensional random deviates.

**Description**

Generate two-dimensional random deviates for a Monte Carlo computation of the integral

*\int\_{0}^{∞}dv f\_{2}(v)\int\_{0}^{∞} f\_{1}(u)\ Φ(u\cdot v, (1 - u) \cdot v; λ, N^{\textrm{MNO}}, N^{\textrm{Reg}}).*

The Monte Carlo technique makes use of stratified importance sampling.

**Usage**

genUV(nSim, nStrata, f1, f2, lambda, nMNO, nReg)

**Arguments**

|  |  |
| --- | --- |
| nSim | number of two-dimensional points to generate |
| nStrata | integer vector of length 2 with the number of strata in each dimension |
| f1, | f2 named lists with the prior marginal distributions of the two-dimensional points |
| lambda | numeric vector |
| nMNO, | nReg non-negative integer vectors |

**Details**

The lengths of the input vectors nMNO and nReg must be equal and independent of the length of the input vector lambda. Notice that nSim points are generated for each of the length(nMNO)*\times*length(lambda) combinations so that the final [data.table](http://127.0.0.1:9775/help/library/pestim/help/data.table-class) has nSim*\times*length(nMNO)*\times*length(lambda) rows.

The prior distributions are specified as named lists where the first component of each list must be the name of distribution ('unif', 'triang', 'degen', 'gamma') and the rest components must be named according to the name of the parameters of the random generator of the corresponding distribution according to:

* unif: xMin, xMax for the minimum, maximum of the sampled interval.
* degen: x0 for the degenerate value of the random variable.
* triang: xMin, xMax, xMode for minimum, maximum and mode (see [qtriang](http://127.0.0.1:9775/help/library/pestim/help/qtriang)).
* gamma: scale and shape with the same meaning as in [rgamma](http://127.0.0.1:9775/help/library/pestim/help/rgamma).

**Value**

genUV returns a [data.table](http://127.0.0.1:9775/help/library/pestim/help/data.table-class) with the (u,v) coordinates of each point together with additional variables:

* The common length of nMNO and nReg identifies the number of territorial cells in which the number of individuals detected by the telecommunication network and official data. The column cellID identifies these territorial cells.
* The length of lambda identifies the number of parameters upon which the integral will be computed in each cell. The column parID identifies each of these input parameters.
* Stratum\_u and Stratum\_v jointly identify each stratum in which the region of integration has been divided with the stratification.

**See Also**

[runif](http://127.0.0.1:9775/help/library/pestim/help/runif), [qtriang](http://127.0.0.1:9775/help/library/pestim/help/qtriang), [rgamma](http://127.0.0.1:9775/help/library/pestim/help/rgamma) for related functions.

**Examples**

# This data.table must have 10x5x3= 150 rows and only one stratum

genUV(nSim = 10, nStrata = c(1, 1),

f1 = list('unif', xMin = 0.3, xMax = 0.5), f2 = list('gamma', shape = 11, scale = 12),

lambda = seq(0, 1, length.out = 5),

nMNO = c(20, 17, 25), nReg = c(115, 123, 119))

|  |  |
| --- | --- |
| kummer {pestim} | R Documentation |

## Confluent hypergeometric or Kummer function

### Description

Partial implementation of the confluent hypergeometric function *{}\_{1}F\_{1}(x ; a; b)*

### Usage

kummer(x, a, b, relTol = 1e-06)

### Arguments

|  |  |
| --- | --- |
| x, | a, b numeric vectors of the same length |
| relTol | relative tolerance (default value 1e-6) understood as the ratio of each term in the series relative to the sum |

### Details

This function is implemented in C++. It is based on Pearson et al (2016). It only implements the Taylor series method together with an asymtoptic expansion based on Watson's lemma

### Value

Return a numeric vector with the values of the function

### Author(s)

Luis Sanguiao Bogdan Oancea

|  |  |
| --- | --- |
| MobPop {pestim} | R Documentation |

## Dataset with simulated data for population counts.

### Description

This dataset provides population counts moving from each pair of cells at succesive time instants for a simulated true population, a corresponding official population in a register and a population detected with a mobile telecommunication network.

### Usage

MobPop

### Format

A [data.table](http://127.0.0.1:9775/help/library/pestim/help/data.table-class) with 96912 rows and 6 variables:

ID\_CELL\_INI

identification code for each initial cell in the displacements

ID\_CELL\_END

identification code for each final cell in the displacements

ID\_T

identification code of each time moment. It is very important to underline that the table collects always displacements between the initial time instant and the corresponding time instant specified by ID\_T

N\_REG

counts according to the population register. Note that these counts do not evolve in time

N\_0

counts of the simulated true population

N\_MNO\_1

counts of individuals detected by the Mobile Network Operator

|  |  |
| --- | --- |
| modeLambda {pestim} | R Documentation |

Mode of the posterior density function of the lambda parameter.

**Description**

Compute the mode of the unnormalized posterior density function of the parameter *λ* in the hierarchical model to estimate population counts.

**Usage**

modeLambda(nMNO, nReg, fu, fv, flambda, relTol = 1e-06, nSim = 10000,

nStrata = c(1, 100), verbose = FALSE)

**Arguments**

|  |  |
| --- | --- |
| nMNO, | nReg non-negative integer vectors with the number of individuals detected in each cell according to the network operator and the register |
| fu, | fv named lists with the prior marginal distributions of the two-dimensional points for the Monte Carlo integration |
| flambda | named list with the prior distribution of the lambda parameter |
| relTol | relative tolerance in the computation of the [kummer](http://127.0.0.1:9775/help/library/pestim/help/kummer) function. Default value is 1e-6 |
| nSim | number of two-dimensional points to generate to compute the integral. Default value is 1e4 |
| nStrata | integer vector of length 2 with the number of strata in each dimension. Default value is c(1, 1e2) |
| verbose | logical (default FALSE) to report progress of the computation |

**Details**

The lengths of the input vectors nMNO and nReg must be equal. Currently the optimization algorithm is a simple direct algorithm taking into account the form of the density function.

The prior distributions are specified as named lists where the first component of each list must be the name of distribution ('unif', 'triang', 'degen', 'gamma') and the rest components must be named according to the name of the parameters of the random generator of the corresponding distribution according to:

* unif: xMin, xMax for the minimum, maximum of the sampled interval.
* degen: x0 for the degenerate value of the random variable.
* triang: xMin, xMax, xMode for minimum, maximum and mode (see [qtriang](http://127.0.0.1:9775/help/library/pestim/help/qtriang)).
* gamma: scale and shape with the same meaning as in [rgamma](http://127.0.0.1:9775/help/library/pestim/help/rgamma).

**Value**

modeLambda returns a vector with the values of the mode of the density function (column probLambda) for each cell.

**See Also**

[dlambda](http://127.0.0.1:9775/help/library/pestim/help/dlambda) for the function to maximize.

**Examples**

# This data.table must have 5x3= 15 rows

modeLambda(nMNO = c(20, 17, 25), nReg = c(115, 123, 119),

fu = list(list('unif', xMin = 0.3, xMax = 0.5),

list('unif', xMin = 0.35, xMax = 0.45),

list('unif', xMin = 0.25, xMax = 0.43)),

fv = list(list('gamma', shape = 11, scale = 12),

list('gamma', shape = 12, scale = 12.3),

list('gamma', shape = 13, scale = 11.5)),

flambda = list(list('gamma', shape = 11, scale = 12),

list('gamma', shape = 12, scale = 12.3),

list('gamma', shape = 13, scale = 12)))

|  |  |
| --- | --- |
| Phi {pestim} | R Documentation |

## The product of ratioBeta and Kummer functions

### Description

Compute the product of [ratioBeta](http://127.0.0.1:9775/help/library/pestim/help/ratioBeta) and [kummer](http://127.0.0.1:9775/help/library/pestim/help/kummer) functions with a specific set of arguments

### Usage

Phi(alpha, beta, lambda, n, relTol = 1e-06)

### Arguments

|  |  |
| --- | --- |
| alpha, | beta non-negative numeric vectors |
| lambda | numeric vector |
| n | non-negative integer vector |
| relTol | relative tolerance in the computation of the [kummer](http://127.0.0.1:9775/help/library/pestim/help/kummer) function. Default value is 1e-6 |

### Value

Phi returns *\frac{B(alpha + m, beta + n)}{B(alpha, beta)}\cdot {}\_{1}F\_{1}(lambda; alpha; beta)*, where *{}\_{1}F\_{1}* stands for the confluent hypergeometric function

The lengths of the input vectors must be all equal except when their length is 1, which are recycled. Otherwise NAs are produced.

### See Also

[ratioBeta](http://127.0.0.1:9775/help/library/pestim/help/ratioBeta), [kummer](http://127.0.0.1:9775/help/library/pestim/help/kummer) for related functions.

### Examples

Phi(1, 1, 0.5, 10)

Phi(1:10, 10:1, seq(0, 1, length.out = 10), 3)

Phi(1:4, 4:1, c(2, 3), c(4, 3, 1))

|  |  |
| --- | --- |
| postN0 {pestim} | R Documentation |

Posterior mean, median, and mode for the number of individuals at the initial time.

**Description**

Compute the posterior mean, median, and mode for the number of individuals generating posterior distribution according to the hierarchical model at the initial time instant

**Usage**

postN0(nMNO, nReg, fu, fv, flambda, n = 1000, scale = 1, relTol = 1e-08,

nSim = 1000, nStrata = c(1, 100), verbose = FALSE)

**Arguments**

|  |  |
| --- | --- |
| nMNO, | nReg non-negative integer vectors with the number of individuals detected in each cell according to the network operator and the register |
| fu, | fv named lists with the prior marginal distributions of the two-dimensional points for the Monte Carlo integration |
| flambda | named list with the prior distribution of the lambda parameter |
| n | number of points to generate in the posterior distribution for the computation. Default value is 1e3 |
| scale | numeric vector with the scale to count the number of individuals. Default value is 1 |
| relTol | relative tolerance in the computation of the [kummer](http://127.0.0.1:9775/help/library/pestim/help/kummer) function. Default value is 1e-6 |
| nSim | number of two-dimensional points to generate to compute the integral. Default value is 1e4 |
| nStrata | integer vector of length 2 with the number of strata in each dimension. Default value is c(1, 1e2) |
| verbose | logical (default FALSE) to report progress of the computation |

**Details**

The prior distributions are specified as named lists where the first component of each list must be the name of distribution ('unif', 'triang', 'degen', 'gamma') and the rest of components must be named according to the name of the parameters of the random generator of the corresponding distribution according to:

* unif: xMin, xMax for the minimum, maximum of the sampled interval.
* degen: x0 for the degenerate value of the random variable.
* triang: xMin, xMax, xMode for minimum, maximum and mode (see [qtriang](http://127.0.0.1:9775/help/library/pestim/help/qtriang)).
* gamma: scale and shape with the same meaning as in [rgamma](http://127.0.0.1:9775/help/library/pestim/help/rgamma).

**Value**

postN0 computes the posterior mean, median, and mode of the posterior distribution for each cell. The function returns a matrix with the estimates in columns and the cells in rows.

**See Also**

[rN0](http://127.0.0.1:9775/help/library/pestim/help/rN0)

**Examples**

# It takes a couple of minutes

postN0(nMNO = 20, nReg = 115, fu = list('unif', xMin = 0.3, xMax = 0.5),

fv = list('unif', xMin = 100, xMax = 120),

flambda = list('gamma', shape = 11, scale = 12))

|  |  |
| --- | --- |
| postNt {pestim} | R Documentation |

Posterior mean, median, and mode for the number of individuals at an arbitrary time.

**Description**

Compute the posterior mean, median, and mode for the number of individuals generating posterior distribution according to the hierarchical model.

**Usage**

postNt(nMNOmat, nReg, fu, fv, flambda, distNames, variation, scale = 1,

n = 1000, relTol = 1e-06, nSim = 1000, nStrata = c(1, 100),

verbose = FALSE)

**Arguments**

|  |  |
| --- | --- |
| nMNOmat | transition matrix with the number of individuals displaced from cell to cell detected by the Mobile Network Operator |
| nReg | non-negative integer vector with the number of individuals detected in each cell according to the population register |
| fu, | fv named lists with the prior marginal distributions of the two-dimensional points for the Monte Carlo integration |
| flambda | named list with the prior distribution of the lambda parameter |
| distNames | character vector with the names of the prior distributions for each cell |
| variation | list of lists whose components are parameters providing a measure of variation of each prior distribution |
| scale | numeric vector with the scale to count the number of individuals. Default value is 1 |
| n | number of points to generate in the posterior distribution for the computation. Default value is 1e3 |
| relTol | relative tolerance in the computation of the [kummer](http://127.0.0.1:9775/help/library/pestim/help/kummer) function. Default value is 1e-6 |
| nSim | number of two-dimensional points to generate to compute the integral. Default value is 1e4 |
| nStrata | integer vector of length 2 with the number of strata in each dimension. Default value is c(1, 1e2) |
| verbose | logical (default FALSE) to report progress of the computation |

**Details**

The prior distributions are specified as named lists where the first component of each list must be the name of distribution ('unif', 'triang', 'degen', 'gamma') and the rest of components must be named according to the name of the parameters of the random generator of the corresponding distribution according to:

* unif: xMin, xMax for the minimum, maximum of the sampled interval.
* degen: x0 for the degenerate value of the random variable.
* triang: xMin, xMax, xMode for minimum, maximum and mode (see [qtriang](http://127.0.0.1:9775/help/library/pestim/help/qtriang)).
* gamma: scale and shape with the same meaning as in [rgamma](http://127.0.0.1:9775/help/library/pestim/help/rgamma).

**Value**

postNt computes the posterior mean, median, and mode of the posterior distribution for each cell at an arbitrary time *t*. The function returns a matrix with the estimates in columns and the cells in rows.

**See Also**

[rNt](http://127.0.0.1:9775/help/library/pestim/help/rNt), [postN0](http://127.0.0.1:9775/help/library/pestim/help/postN0), [postNtcondN0](http://127.0.0.1:9775/help/library/pestim/help/postNtcondN0)

**Examples**

## First, the inputs:

#The transition matrix of individuals detected by the MNO

nMNOmat <- rbind(c(10, 3, 4), c(5, 21, 3), c(3, 9, 18))

# Population at the initial time of each cell according to the population register

nReg <- c(90, 130, 101)

# List of priors for u

u0 <- rowSums(nMNOmat) / nReg

cv\_u0 <- 0.15

fu <- lapply(u0, function(u){

umin <- max(0, u - cv\_u0 \* u)

umax <- min(1, u + cv\_u0 \* u)

output <- list('unif', xMin = umin, xMax = umax)

return(output)

})

# List of priors for v

v0 <- nReg

cv\_v0 <- 0.10

fv <- lapply(v0, function(u){

umin <- max(0, u - cv\_v0 \* u)

umax <- u + cv\_v0 \* u

output <- list('unif', xMin = umin, xMax = umax)

return(output)

})

# List of priors for lambda

cv\_lambda <- 0.6

alpha <- 1 / cv\_lambda\*\*2 - 1

flambda <- lapply(v0, function(v){list('gamma', shape = 1 + alpha, scale = v / alpha)})

# Names and parameters of priors for the transition probabilities

distNames <- rep('unif', 3)

variation <- rep(list(list(cv = 0.20)), 3)

# It takes a couple of minutes.

postNt(nMNOmat, nReg, fu, fv, flambda, distNames, variation)

|  |  |
| --- | --- |
| postNtcondN0 {pestim} | R Documentation |

## Posterior mean, median, and mode for the number of individuals at an arbitrary time conditioned upon the initial population.

### Description

Compute the posterior mean, median, and mode for the number of individuals generating posterior distribution according to the hierarchical model conditioned upon the initial population of each cell, which must be provided

### Usage

postNtcondN0(N0, nMNOmat, distNames, variation, n = 1000)

### Arguments

|  |  |
| --- | --- |
| N0 | initial population in each cell |
| nMNOmat | transition matrix with the number of individuals displaced from cell to cell detected by the Mobile Network Operator |
| distNames | character vector with the names of the prior distributions for each cell |
| variation | list of lists whose components are parameters providing a measure of variation of each prior distribution |
| n | number of points to generate in the posterior distribution for the computation. Default value is 1e3 |

### Value

Return a matrix with three columns (mean, median, and mode estimates) and one row per cell

### Examples

## First, the inputs:

# The initial population

N0 <- c(93, 123, 130)

#The transition matrix of individuals detected by the MNO

nMNOmat <- rbind(c(10, 3, 4), c(5, 21, 3), c(3, 9, 18))

# Names and parameters of priors for the transition probabilities

distNames <- rep('unif', 3)

variation <- rep(list(list(cv = 0.20)), 3)

# It takes a couple of minutes.

postNtcondN0(N0, nMNOmat, distNames, variation)

|  |  |
| --- | --- |
| ratioBeta {pestim} | R Documentation |

## The ratio of two beta functions.

### Description

Compute the ratio of two beta functions whose arguments differ by integer numbers

### Usage

ratioBeta(alpha, beta, m, n)

### Arguments

|  |  |
| --- | --- |
| alpha, | beta non-negative numeric vectors |
| m, | n non-negative integer vectors |

### Value

ratioBeta gives *\frac{B(alpha + m, beta + n)}{B(alpha, beta)}*

The lengths of the input vectors must be all equal except when their length is 1, which are recycled. Otherwise NAs are produced.

### See Also

[beta](http://127.0.0.1:9775/help/library/pestim/help/beta), [lbeta](http://127.0.0.1:9775/help/library/pestim/help/lbeta) for related functions.

### Examples

ratioBeta(10, 13, 2, 3)

ratioBeta(1:10, 10:1, 2, 3)

ratioBeta(1:3, 3:1, c(2, 3), 4)

|  |  |
| --- | --- |
| rg {pestim} | R Documentation |

## Generation of random deviates of the candidate distribution.

### Description

Generate random points according to the candidate probability distribution in the accept-reject method of generation of random variables applied to the distribution of the lambda parameter

### Usage

rg(n, nMNO, nReg, fu, fv, flambda, relTol = 1e-06, nSim = 10000,

nStrata = c(1, 100), verbose = FALSE)

### Arguments

|  |  |
| --- | --- |
| n | number of values to generate |
| nMNO, | nReg non-negative integer vectors with the number of individuals detected in each cell according to the network operator and the register |
| fu, | fv named lists with the prior marginal distributions of the two-dimensional points for the Monte Carlo integration |
| flambda | named list with the prior distribution of the lambda parameter |
| relTol | relative tolerance in the computation of the [kummer](http://127.0.0.1:9775/help/library/pestim/help/kummer) function. Default value is 1e-6 |
| nSim | number of two-dimensional points to generate to compute the integral. Default value is 1e4 |
| nStrata | integer vector of length 2 with the number of strata in each dimension. Default value is c(1, 1e2) |
| verbose | logical (default FALSE) to report progress of the computation |

### Details

The candidate distribution is a gamma distribution with parameters shape = nMNO + 1 and scale = *λ^{\*}* / nMNO, where *λ^{\*}* stands for the mode of the posterior distribution of the lambda parameter.

It is important to know that currently this function accepts only parameters for a single cell at a time. In case of interest for the candidate density function values for a set of cells, the user should program his/her own routine to apply this function to every cell.

### Value

rg generates n points according to the candidate distribution.

### See Also

[modeLambda](http://127.0.0.1:9775/help/library/pestim/help/modeLambda), [dlambda](http://127.0.0.1:9775/help/library/pestim/help/dlambda) for related functions.

### Examples

hist(rg(1e5, nMNO = 20, nReg = 115, fu = list('unif', xMin = 0.3, xMax = 0.5),

fv = list('unif', xMin = 100, xMax = 120),

flambda = list('gamma', shape = 11, scale = 12)), breaks = seq(1, 200, by = 2), main ='')

|  |  |
| --- | --- |
| rlambda {pestim} | R Documentation |

Generation of random deviates of the posterior distribution of parameter lambda.

**Description**

Generate random points according to the posterior probability distribution of the parameter lambda in the hierarchical model.

**Usage**

rlambda(n, nMNO, nReg, fu, fv, flambda, relTol = 1e-06, nSim = 10000,

nStrata = c(1, 100), verbose = FALSE)

**Arguments**

|  |  |
| --- | --- |
| n | number of values to generate |
| nMNO, | nReg non-negative integer vectors with the number of individuals detected in each cell according to the network operator and the register |
| fu, | fv named lists with the prior marginal distributions of the two-dimensional points for the Monte Carlo integration |
| flambda | named list with the prior distribution of the lambda parameter |
| relTol | relative tolerance in the computation of the [kummer](http://127.0.0.1:9775/help/library/pestim/help/kummer) function. Default value is 1e-6 |
| nSim | number of two-dimensional points to generate to compute the integral. Default value is 1e4 |
| nStrata | integer vector of length 2 with the number of strata in each dimension. Default value is c(1, 1e2) |
| verbose | logical (default FALSE) to report progress of the computation |

**Details**

The points are generated according to the accept-reject method using as candidate distribution a Cauchy distribution whose parameters are taken from the prior distributions and the mode of the posterior distribution of the lambda parameter.

The prior distributions are specified as named lists where the first component of each list must be the name of distribution ('unif', 'triang', 'degen', 'gamma') and the rest components must be named according to the name of the parameters of the random generator of the corresponding distribution according to:

* unif: xMin, xMax for the minimum, maximum of the sampled interval.
* degen: x0 for the degenerate value of the random variable.
* triang: xMin, xMax, xMode for minimum, maximum and mode (see [qtriang](http://127.0.0.1:9775/help/library/pestim/help/qtriang)).
* gamma: scale and shape with the same meaning as in [rgamma](http://127.0.0.1:9775/help/library/pestim/help/rgamma).

**Value**

rlambda generates n points according to the posterior distribution of the parameter lambda. The function returns a vector with these points.

**See Also**

[dlambda](http://127.0.0.1:9775/help/library/pestim/help/dlambda), [rg](http://127.0.0.1:9775/help/library/pestim/help/rg) for related functions.

**Examples**

# It takes a couple of minutes

hist(rN0(500, nMNO = 20, nReg = 115, fu = list('unif', xMin = 0.3, xMax = 0.5),

fv = list('unif', xMin = 100, xMax = 120),

flambda = list('gamma', shape = 11, scale = 12))$N0,

breaks = seq(1, 200, by = 1), main ='', xlab = 'number of individuals')

|  |  |
| --- | --- |
| rmatProb {pestim} | R Documentation |

Generate matrices of transition probabilities

**Description**

Generate a list of matrices of transition probabilities computed with the transition matrices of individuals among pairs of cells detected by the network and specified probability input distributions per cell.

**Usage**

rmatProb(n, nMNOmat, distNames, variation)

**Arguments**

|  |  |
| --- | --- |
| n | number of matrices to generate |
| nMNOmat | transition matrix with the number of individuals displaced from cell to cell detected by the Mobile Network Operator |
| distNames | character vector with the names of the prior distributions for each cell |
| variation | list of lists whose components are parameters providing a measure of variation of each prior distribution |

**Details**

The function generates the probabilities according to a Dirichlet distribution with parameters generated by [alphaPrior](http://127.0.0.1:9775/help/library/pestim/help/alphaPrior). These parameters are generated with distributions whose names are taken from the input parameter distNames and construct the corresponding prior distribution for each cell *j* with mode at*u\_{j}^{\*}=N\_{j}*, where *N\_{j}* is taken from the sum of rows of nMNOmat. Next the rest of parameters of the distribution are computed according to the dispersion parameters specified in variation.

As accepted distribution names, currently the user can specify unif, degen, triang, and gamma.

The dispersion parameters recognised so far are the coefficients of variation only (standard deviation divided by the mean of the distribution). These dispersion parameters must be specified by a named component cv with a numeric value in *[0, 1]*.

For each distribution the parameters are computed as follows:

* unif: This is the uniform distribution with parameters xMax and xMin. Both parameters are computed by *u\_{j}^{\*}\cdot(1\pm√{3}\textrm{cv})*, respectively, in each cell *j*.
* degen: This is the degenerate distribution with parameter X0 taken as *u\_{j}^{\*}* in each cell *j*.
* triang: This is the triangular distribution [triang](http://127.0.0.1:9775/help/library/pestim/help/triang) with parameters xMax, xMin, and xMode. The latter is taken directly from nMNOfrom. The distribution is assumed to be symmetrical so that the two former parameters are computed by *u\_{j}^{\*}\cdot(1\pm√{3}\textrm{cv})*, respectively, in each cell *j*.
* gamma: This is the gamma distribution with parameters shape and scale. The former is computed as *\frac{1}{\textrm{cv}^2}* and the latter as*frac{u\_{j}^{\*}}{\textrm{scale} - 1}*.

**Value**

A list of n matrices with transition probabilities

**Examples**

nMNOmat <- rbind(c(10, 3, 4), c(5, 21, 3), c(3, 9, 18))

distNames <- rep('unif', 3)

variation <- rep(list(list(cv = 0.20)), 3)

rmatProb(10, nMNOmat, distNames, variation)

|  |  |
| --- | --- |
| rN0 {pestim} | R Documentation |

Generation of random deviates of the posterior distribution of initial population counts.

**Description**

Generate random points according to the posterior probability distribution of the number of individuals in the hierarchical model.

**Usage**

rN0(n, nMNO, nReg, fu, fv, flambda, scale = 1, relTol = 1e-06,

nSim = 10000, nStrata = c(1, 100), verbose = FALSE)

**Arguments**

|  |  |
| --- | --- |
| n | number of values to generate |
| nMNO, | nReg non-negative integer vectors with the number of individuals detected in each cell according to the network operator and the register |
| fu, | fv named lists with the prior marginal distributions of the two-dimensional points for the Monte Carlo integration |
| flambda | named list with the prior distribution of the lambda parameter |
| scale | numeric vector with the scale to count the number of individuals. Default value is 1 |
| relTol | relative tolerance in the computation of the [kummer](http://127.0.0.1:9775/help/library/pestim/help/kummer) function. Default value is 1e-6 |
| nSim | number of two-dimensional points to generate to compute the integral. Default value is 1e4 |
| nStrata | integer vector of length 2 with the number of strata in each dimension. Default value is c(1, 1e2) |
| verbose | logical (default FALSE) to report progress of the computation |

**Details**

The posterior distribution is a Poisson distribution with parameter lambda \* scale, where the values of lambda are generated with the function [rlambda](http://127.0.0.1:9775/help/library/pestim/help/rlambda).

The prior distributions are specified as named lists where the first component of each list must be the name of distribution ('unif', 'triang', 'degen', 'gamma') and the rest components must be named according to the name of the parameters of the random generator of the corresponding distribution according to:

* unif: xMin, xMax for the minimum, maximum of the sampled interval.
* degen: x0 for the degenerate value of the random variable.
* triang: xMin, xMax, xMode for minimum, maximum and mode (see [qtriang](http://127.0.0.1:9775/help/library/pestim/help/qtriang)).
* gamma: scale and shape with the same meaning as in [rgamma](http://127.0.0.1:9775/help/library/pestim/help/rgamma).

**Value**

rN0 generates n points according to the posterior distribution. The function returns a [data.table](http://127.0.0.1:9775/help/library/pestim/help/data.table-class) with these points (under the column N0) together with the additional variables:

* The common length of nMNO and nReg identifies the number of territorial cells in which the number of individuals detected by the telecommunication network and official data. The column cellID identifies these territorial cells.
* The different values of the generated values of lambda are returned under the column lambda.
* The inputs nMNO and nReg are also included in the output [data.table](http://127.0.0.1:9775/help/library/pestim/help/data.table-class) in columns under the same name.

**See Also**

[rlambda](http://127.0.0.1:9775/help/library/pestim/help/rlambda), [rg](http://127.0.0.1:9775/help/library/pestim/help/rg), [rNt](http://127.0.0.1:9775/help/library/pestim/help/rNt) for related functions.

**Examples**

# It takes a couple of minutes

hist(rN0(500, nMNO = 20, nReg = 115, fu = list('unif', xMin = 0.3, xMax = 0.5),

fv = list('unif', xMin = 100, xMax = 120),

flambda = list('gamma', shape = 11, scale = 12))$N0,

breaks = seq(1, 200, by = 1), main ='', xlab = 'number of individuals')

|  |  |
| --- | --- |
| rNt {pestim} | R Documentation |

Generation of random deviates of the posterior distribution of population counts.

**Description**

Generate random points according to the posterior probability distribution of the number of individuals in the hierarchical model at arbitrary time instants.

**Usage**

rNt(n, nMNOmat, nReg, fu, fv, flambda, distNames, variation, scale = 1,

relTol = 1e-06, nSim = 1000, nStrata = c(1, 100), verbose = FALSE)

**Arguments**

|  |  |
| --- | --- |
| n | number of values to generate |
| nMNOmat | transition matrix with the number of individuals displaced from cell to cell detected by the Mobile Network Operator |
| nReg | non-negative integer vectors with the number of individuals detected in each cell according to the network operator and the register |
| fu, | fv named lists with the prior marginal distributions of the two-dimensional points for the Monte Carlo integration |
| flambda | named list with the prior distribution of the lambda parameter |
| distNames | character vector with the names of the prior distributions for each cell |
| variation | list of lists whose components are parameters providing a measure of variation of each prior distribution |
| scale | numeric vector with the scale to count the number of individuals. Default value is 1 |
| relTol | relative tolerance in the computation of the [kummer](http://127.0.0.1:9775/help/library/pestim/help/kummer) function. Default value is 1e-6 |
| nSim | number of two-dimensional points to generate to compute the integral. Default value is 1e4 |
| nStrata | integer vector of length 2 with the number of strata in each dimension. Default value is c(1, 1e2) |
| verbose | logical (default FALSE) to report progress of the computation |

**Details**

The posterior distribution is a Poisson distribution with parameter lambda \* scale, where the values of lambda are generated with the function [rlambda](http://127.0.0.1:9775/help/library/pestim/help/rlambda).

The prior distributions are specified as named lists where the first component of each list must be the name of distribution ('unif', 'triang', 'degen', 'gamma') and the rest of components must be named according to the name of the parameters of the random generator of the corresponding distribution according to:

* unif: xMin, xMax for the minimum, maximum of the sampled interval.
* degen: x0 for the degenerate value of the random variable.
* triang: xMin, xMax, xMode for minimum, maximum and mode (see [qtriang](http://127.0.0.1:9775/help/library/pestim/help/qtriang)).
* gamma: scale and shape with the same meaning as in [rgamma](http://127.0.0.1:9775/help/library/pestim/help/rgamma).

**Value**

rNt generates n points according to the posterior distribution. The function returns a [data.table](http://127.0.0.1:9775/help/library/pestim/help/data.table-class) with these points (under the column N0) together with the additional variables:

* The common length of nMNO and nReg identifies the number of territorial cells in which the number of individuals detected by the telecommunication network and official data. The column cellID identifies these territorial cells.
* The different values of the generated values of lambda are returned under the column lambda.
* The inputs nMNO and nReg are also included in the output [data.table](http://127.0.0.1:9775/help/library/pestim/help/data.table-class) in columns under the same name.

**See Also**

[rlambda](http://127.0.0.1:9775/help/library/pestim/help/rlambda), [rg](http://127.0.0.1:9775/help/library/pestim/help/rg), [rNt](http://127.0.0.1:9775/help/library/pestim/help/rNt) for related functions.

**Examples**

## First, the inputs:

# The number of generated values

n <- 1e3

#The transition matrix of individuals detected by the MNO

nMNOmat <- rbind(c(10, 3, 4), c(5, 21, 3), c(3, 9, 18))

# Population at the initial time of each cell according to the population register

nReg <- c(90, 130, 101)

# List of priors for u

u0 <- rowSums(nMNOmat) / nReg

cv\_u0 <- 0.15

fu <- lapply(u0, function(u){

umin <- max(0, u - cv\_u0 \* u)

umax <- min(1, u + cv\_u0 \* u)

output <- list('unif', xMin = umin, xMax = umax)

return(output)

})

# List of priors for v

v0 <- nReg

cv\_v0 <- 0.10

fv <- lapply(v0, function(u){

umin <- max(0, u - cv\_v0 \* u)

umax <- u + cv\_v0 \* u

output <- list('unif', xMin = umin, xMax = umax)

return(output)

})

# List of priors for lambda

cv\_lambda <- 0.6

alpha <- 1 / cv\_lambda\*\*2 - 1

flambda <- lapply(v0, function(v){list('gamma', shape = 1 + alpha, scale = v / alpha)})

# Names and parameters of priors for the transition probabilities

distNames <- rep('unif', 3)

variation <- rep(list(list(cv = 0.20)), 3)

# The output

Nt <- rNt(n, nMNOmat, nReg, fu, fv, flambda, distNames, variation)$N

hist(Nt, breaks = seq(1, max(Nt) + 10, by = 1), main ='', xlab = 'number of individuals')

|  |  |
| --- | --- |
| rNtcondN0 {pestim} | R Documentation |

Conditioned generation of random deviates of the posterior distribution of population counts.

**Description**

Generate random deviates of the posterior distribution of the number of individuals at an arbitrary time instant conditioned upon the initial population.

**Usage**

rNtcondN0(n, N0, nMNOmat, distNames, variation)

**Arguments**

|  |  |
| --- | --- |
| n | number of values to generate |
| N0 | initial population in each cell |
| nMNOmat | transition matrix with the number of individuals displaced from cell to cell detected by the Mobile Network Operator |
| distNames | character vector with the names of the prior distributions for each cell |
| variation | list of lists whose components are parameters providing a measure of variation of each prior distribution |

**Details**

The function generates the probabilities according to a Dirichlet distribution with parameters generated by [alphaPrior](http://127.0.0.1:9775/help/library/pestim/help/alphaPrior). These parameters are generated with distributions whose names are taken from the input parameter distNames and construct the corresponding prior distribution for each cell *j* with mode at*u\_{j}^{\*}=N\_{j}*, where *N\_{j}* is taken from the sum of rows of nMNOmat. Next the rest of parameters of the distribution are computed according to the dispersion parameters specified in variation.

As accepted distribution names, currently the user can specify unif, degen, triang, and gamma.

The dispersion parameters recognised so far are the coefficients of variation only (standard deviation divided by the mean of the distribution). These dispersion parameters must be specified by a named component cv with a numeric value in *[0, 1]*.

For each distribution the parameters are computed as follows:

* unif: This is the uniform distribution with parameters xMax and xMin. Both parameters are computed by *u\_{j}^{\*}\cdot(1\pm√{3}\textrm{cv})*, respectively, in each cell *j*.
* degen: This is the degenerate distribution with parameter X0 taken as *u\_{j}^{\*}* in each cell *j*.
* triang: This is the triangular distribution [triang](http://127.0.0.1:9775/help/library/pestim/help/triang) with parameters xMax, xMin, and xMode. The latter is taken directly from nMNOfrom. The distribution is assumed to be symmetrical so that the two former parameters are computed by *u\_{j}^{\*}\cdot(1\pm√{3}\textrm{cv})*, respectively, in each cell *j*.
* gamma: This is the gamma distribution with parameters shape and scale. The former is computed as *\frac{1}{\textrm{cv}^2}* and the latter as*frac{u\_{j}^{\*}}{\textrm{scale} - 1}*.

**Value**

Return a matrix with as many columns as cells and n rows with the generated values

**Examples**

N0 <- c(93, 123, 130)

nMNOmat <- rbind(c(10, 3, 4), c(5, 21, 3), c(3, 9, 18))

distNames <- rep('unif', 3)

variation <- rep(list(list(cv = 0.20)), 3)

rNtcondN0(1e3, N0, nMNOmat, distNames, variation)

|  |  |
| --- | --- |
| p {pestim} | R Documentation |

Generate random vector deviates of transition probabilities.

**Description**

Generate random vector deviates of the transition probabilities *p\_{ij}(t\_{0}, t\_{n})* for a given cell *i* stacked into an *n\times*(number of cells) matrix

**Usage**

rp(n, flist)

**Arguments**

|  |  |
| --- | --- |
| n | number of probability vectors to generate |
| flist | list with the prior distributions for each cell |

**Details**

The prior distributions are specified as named lists where the first component of each list must be the name of distribution ('unif', 'triang', 'degen', 'gamma') and the rest components must be named according to the name of the parameters of the random generator of the corresponding distribution according to:

* unif: xMin, xMax for the minimum, maximum of the sampled interval.
* degen: x0 for the degenerate value of the random variable.
* triang: xMin, xMax, xMode for minimum, maximum and mode (see [qtriang](http://127.0.0.1:9775/help/library/pestim/help/qtriang)).
* gamma: scale and shape with the same meaning as in [rgamma](http://127.0.0.1:9775/help/library/pestim/help/rgamma).

**Value**

Return a matrix with n rows and as many columns as cells taken from the length of flist. Each row is thus a probability vector

**Examples**

flist <- alphaPrior(c(10, 3, 4), c('unif', 'triang', 'gamma'),

list(list(cv = 0.1), list(cv = 0.05), list(cv = 0.15)))

rp(10, flist)