

# Package ‘mispitools’

July 8, 2024

**Title** Missing Person Identification Tools

**Version** 1.1.0

**Description** An open source software package written in R statistical language. It consist in a set of decision making tools to conduct missing person searches. Particularly, it allows computing optimal LR threshold for declaring potential matches in DNA-based database search. More recently 'mispitools' incorporates preliminary investigation data based LR. Statistical weight of different traces of evidence such as biological sex, age and hair color are presented. For citing mispitools please use the following references: Marsico and Caridi, 2023 <[doi:10.1016/j.fsigen.2023.102891](https://doi.org/10.1016/j.fsigen.2023.102891)> and Marsico, Vigeland et al. 2021 <[doi:10.1016/j.fsigen.2021.102519](https://doi.org/10.1016/j.fsigen.2021.102519)>.

**License** GPL (>= 3)

**Encoding** UTF-8

**LazyData** true

**Imports** forrel, pedtools, plotly, dplyr, highcharter, tidyr, tidyverse, DirichletReg, stats, purrr, patchwork, reshape2, graphics, ggplot2, shiny

**RoxygenNote** 7.2.3

**URL** <https://github.com/MarsicoFL/mispitools>

**BugReports** <https://github.com/MarsicoFL/mispitools/issues>

**Depends** R (>= 2.10)

**NeedsCompilation** no

**Author** Franco Marsico [aut, cre] (<<https://orcid.org/0000-0002-0740-5516>>)

**Maintainer** Franco Marsico <[franco.lmarsico@gmail.com](mailto:franco.lmarsico@gmail.com)>

**Repository** CRAN

**Date/Publication** 2024-07-08 00:10:01 UTC

## Contents

Argentina . . . . .	2
Asia . . . . .	3

Austria . . . . .	3
bidirectionalKL . . . . .	3
BosniaHerz . . . . .	4
China . . . . .	4
Cmodel . . . . .	5
combLR . . . . .	6
CondPlot . . . . .	7
CPT_MP . . . . .	7
CPT_POP . . . . .	8
deplot . . . . .	9
DeT . . . . .	9
Europe . . . . .	10
getfreqs . . . . .	10
Japan . . . . .	11
LRage . . . . .	11
LRcol . . . . .	12
LRdate . . . . .	13
LRdist . . . . .	14
LRsex . . . . .	15
makeMPprelim . . . . .	16
makePOIgen . . . . .	17
makePOIprelim . . . . .	18
mispiApp . . . . .	19
multi_kl_divergence . . . . .	19
postSim . . . . .	20
simLRgen . . . . .	21
simLRprelim . . . . .	22
Trates . . . . .	23
USA . . . . .	24
<b>Index</b>	<b>25</b>

---

Argentina

*STRs allelic frequencies from specified country.*

---

### Description

STRs allelic frequencies from specified country.

### Usage

Argentina

### Format

A data frame allele frequencies

---

Asia	<i>STRs allelic frequencies from specified country.</i>
------	---

---

**Description**

A dataset of allele frequencies.

**Usage**

Asia

**Format**

A data frame allele frequencies

---

Austria	<i>STRs allelic frequencies from specified country.</i>
---------	---

---

**Description**

STRs allelic frequencies from specified country.

**Usage**

Austria

**Format**

A data frame allele frequencies

---

bidirectionalKL	<i>Kullback-Leibler Divergence Calculation for Genetic Markers</i>
-----------------	--

---

**Description**

This function calculates the Kullback-Leibler divergence for shared genetic markers between two populations, considering allele frequencies. It normalizes data, adjusts zero frequencies, and calculates divergence in both directions.

**Usage**

```
bidirectionalKL(data1, data2, minFreq = 1e-10)
```

**Arguments**

data1	DataFrame with allele frequencies for the first population.
data2	DataFrame with allele frequencies for the second population.
minFreq	Minimum frequency to be considered for unobserved or poorly observed alleles.

**Value**

A list containing the Kullback-Leibler divergence from data1 to data2 and vice versa.

**Examples**

```
bidirectionalKL(Argentina, BosniaHerz)
```

---

BosniaHerz	<i>STRs allelic frequencies from specified country.</i>
------------	---

---

**Description**

STRs allelic frequencies from specified country.

**Usage**

```
BosniaHerz
```

**Format**

A data frame allele frequencies

---

China	<i>STRs allelic frequencies from specified country.</i>
-------	---

---

**Description**

STRs allelic frequencies from specified country.

**Usage**

```
China
```

**Format**

A data frame allele frequencies

---

Cmodel

*Epsilon hair color matrix*

---

### Description

Epsilon hair color matrix

### Usage

```
Cmodel(  
  errorModel = c("custom", "uniform")[1],  
  ep = 0.01,  
  ep12 = 0.01,  
  ep13 = 0.005,  
  ep14 = 0.01,  
  ep15 = 0.003,  
  ep23 = 0.01,  
  ep24 = 0.003,  
  ep25 = 0.01,  
  ep34 = 0.003,  
  ep35 = 0.003,  
  ep45 = 0.01  
)
```

### Arguments

errorModel	custom allows selecting a specific epsilon for each MP-UHR pair, uniform use ep for all.
ep	epsilon
ep12	epsilon
ep13	epsilon
ep14	epsilon
ep15	epsilon
ep23	epsilon
ep24	epsilon
ep25	epsilon
ep34	epsilon
ep35	epsilon
ep45	epsilon

### Value

A value of Likelihood ratio based on preliminary investigation data. In this case, sex.

## Examples

```
Cmodel()
```

---

combLR	<i>Combine LRs: a function for combining LRs obtained from simulations.</i>
--------	---

---

## Description

Combine LRs: a function for combining LRs obtained from simulations.

## Usage

```
combLR(LRdatasim1, LRdatasim2)
```

## Arguments

LRdatasim1	A data frame object with the results of simulations. Outputs from simLRgen or simLRprelim functions.
LRdatasim2	A second data frame object with the results of simulations. Outputs from simLRgen or simLRprelim functions.

## Value

An object of class data.frame combining the LRs obtained from simulations (the function multiplies the LRs).

## Examples

```
library(mispitools)
library(forrel)
x = linearPed(2)
x = setMarkers(x, locusAttributes = NorwegianFrequencies[1:5])
x = profileSim(x, N = 1, ids = 2)
LRdatasim1 = simLRgen(x, missing = 5, 10, 123)
LRdatasim2 = simLRprelim("sex")
combLR(LRdatasim1,LRdatasim2)
```

---

CondPlot	<i>General plot for conditioned probabilities and LR combining variables</i>
----------	--

---

**Description**

General plot for conditioned probabilities and LR combining variables

**Usage**

```
CondPlot(CPT_POP, CPT_MP)
```

**Arguments**

CPT_POP	Population conditioned probability table
CPT_MP	Missing person conditioned probability table

**Value**

A value of Likelihood ratio based on preliminary investigation data. In this case, sex.

**Examples**

```
Cmodel()
```

---

CPT_MP	<i>Missing person based conditioned probability</i>
--------	---

---

**Description**

Missing person based conditioned probability

**Usage**

```
CPT_MP(MPs = "F", MPc = 1, eps = 0.05, epa = 0.05, epc = Cmodel())
```

**Arguments**

MPs	Missing person sex
MPc	Missing person hair color
eps	sex epsilon
epa	age epsilon - Age is not specified in this first version, because it assumes uniformity.
epc	color model

**Value**

A value of Likelihood ratio based on preliminary investigation data. In this case, sex.

**Examples**

```
CPT_MP()
```

---

CPT\_POP

*Population based conditioned probability*

---

**Description**

Population based conditioned probability

**Usage**

```
CPT_POP(  
  propS = c(0.5, 0.5),  
  MPa = 40,  
  MPr = 6,  
  propC = c(0.3, 0.2, 0.25, 0.15, 0.1)  
)
```

**Arguments**

propS	age epsilon - Age is not specified in this first version, because it assumes uniformity.
MPa	Missing person sex
MPr	Missing person hair color
propC	sex epsilon

**Value**

A value of Likelihood ratio based on preliminary investigation data. In this case, sex.

**Examples**

```
CPT_POP()
```



---

deplot	<i>Decision making plot: a function for plotting false positive and false negative rates for each LR threshold.</i>
--------	---

---

**Description**

Decision making plot: a function for plotting false positive and false negative rates for each LR threshold.

**Usage**

```
deplot(datasim)
```

**Arguments**

datasim	Input dataframe containing expected LRs for related and unrelated POIs. It should be the output from makeLRsims function.
---------	---

**Value**

A plot showing false positive and false negative rates for each likelihood ratio threshold.

**Examples**

```
library(forrel)
library(plotly)
x = linearPed(2)
x = setMarkers(x, locusAttributes = NorwegianFrequencies[1:5])
x = profileSim(x, N = 1, ids = 2)
datasim = simLRgen(x, missing = 5, 10, 123)
deplot(datasim)
```

---

DeT	<i>Decision Threshold: a function for computing likelihood ratio decision threshold.</i>
-----	--

---

**Description**

Decision Threshold: a function for computing likelihood ratio decision threshold.

**Usage**

```
DeT(datasim, weight)
```

**Arguments**

datasim	Input dataframe containing expected LR <sub>s</sub> for related and unrelated POIs. It should be the output from makeLRsims function.
weight	The differential weight between false positives and false negatives. A value of 10 is suggested.

**Value**

A value of Likelihood ratio suggested as threshold based on false positive-false negative trade-off.

**Examples**

```
library(forrel)
x = linearPed(2)
x = setMarkers(x, locusAttributes = NorwegianFrequencies[1:5])
x = profileSim(x, N = 1, ids = 2)
datasim = simLRgen(x, missing = 5, 10, 123)
DeT(datasim, 10)
```

---

Europe	<i>STRs allelic frequencies from specified country.</i>
--------	---

---

**Description**

STRs allelic frequencies from specified country.

**Usage**

Europe

**Format**

A data frame allele frequencies

---

getfreqs	<i>Function for getting STR allele frequencies from different world populations.</i>
----------	--

---

**Description**

Function for getting STR allele frequencies from different world populations.

**Usage**

getfreqs(region)

**Arguments**

region select the place of the allele frequency database. Possible values are listed: "Argentina", "Asia", "Europe", "USA", "Austria", "BosniaHerz", "China" and "Japan".

**Value**

An allele frequency database adapted compatible with pedtools format.

**Source**

<https://doi.org/10.1016/j.fsigss.2009.08.178>; <https://doi.org/10.1016/j.fsigen.2016.06.008>; <https://doi.org/10.1016/j.fsigen.2016.06.008>

---

Japan	<i>STRs allelic frequencies from specified country.</i>
-------	---

---

**Description**

STRs allelic frequencies from specified country.

**Usage**

Japan

**Format**

A data frame allele frequencies

---

LRage	<i>Likelihood ratio for age variable</i>
-------	--

---

**Description**

Likelihood ratio for age variable

**Usage**

```
LRage(
  MPa = 40,
  MPr = 6,
  UHRr = 1,
  gam = 0.07,
  nsims = 1000,
  epa = 0.05,
  erRa = epa,
```

```

H = 1,
modelA = c("uniform", "custom")[1],
LR = FALSE
)

```

### Arguments

MPa	Missing person age
MPr	Missing person age range.
UHRr	Unidentified person range
gam	Simulation parameter for UHR ages.
nsims	number of simulations.
epa	epsilon age
erRa	error rate in the database.
H	hypothesis tested, H1: UHR is MP, H2: UHR is not MP.
modelA	reference database probabilities, uniform assumes equally probable ages. Custom needs a vector with ages frequencies.
LR	compute LR values

### Value

A value of Likelihood ratio based on preliminary investigation data. In this case, Age.

---

LRcol

*Likelihood ratio for age variable*

---

### Description

Likelihood ratio for age variable

### Usage

```

LRcol(
  MPc = 1,
  epc = Cmodel(),
  erRc = epc,
  nsims = 1000,
  Pc = c(0.3, 0.2, 0.25, 0.15, 0.1),
  H = 1,
  Qprop = MPc,
  LR = FALSE
)

```

**Arguments**

MpC	MP hair color
epc	epsilon parameter.
erRc	error rate in the database.
nsims	number of simulations performed.
Pc	hair color probabilities.
H	hypothesis tested, H1: UHR is MP, H2: UHR is no MP
Qprop	Query color tested.
LR	compute LR values

**Value**

A value of Likelihood ratio based on preliminary investigation data. In this case, hair color.

**Examples**

```
LRcol()
```

---

LRdate

*Likelihood ratio for birth date in missing person searches*

---

**Description**

Likelihood ratio for birth date in missing person searches

**Usage**

```
LRdate(
  ABD = "1976-05-31",
  DBD = "1976-07-15",
  PrelimData,
  alpha = c(1, 4, 60, 11, 6, 4, 4),
  cuts = c(-120, -30, 30, 120, 240, 360),
  draw = 500,
  type = 1,
  seed = 123
)
```

**Arguments**

ABD	Actual birth date of the missing person.
DBD	Declared birth date of the person of interest.
PrelimData	Used when type = 2, is the dataframe with the DBD of the persons of interest in the database.

alpha	A vector containing the alpha values for the dirichlet. It should contain the number of categories of differences between DBD and ABD.
cuts	Value of differences between DBD and ABD used for category definition.
draw	Number of simulations for Dirichlet distribution computation.
type	Type of scenario, type 1 is an "open search", where it is unknown if the missing person is in the database. Type 2 refers to a scenario where the missing person is in the database.
seed	Seed for simulations.

**Value**

A value of Likelihood ratio based on preliminary investigation data. In this case, birth date.

**Examples**

```
library(DirichletReg)
LRdate(ABD = "1976-05-31", DBD = "1976-07-15",
PrelimData, alpha = c(1, 4, 60, 11, 6, 4, 4),
cuts = c(-120, -30, 30, 120, 240, 360),
type = 1, seed = 123)
```

---

LRdist	<i>Likelihood ratio distribution: a function for plotting expected log<sub>10</sub>(LR) distributions under relatedness and unrelatedness.</i>
--------	--

---

**Description**

Likelihood ratio distribution: a function for plotting expected log<sub>10</sub>(LR) distributions under relatedness and unrelatedness.

**Usage**

```
LRdist(datasim, type = 1)
```

**Arguments**

datasim	Input dataframe containing expected LRs for related and unrelated POIs. It should be the output from makeLRsims function.
type	Select between a density plot (type = 1, default) or a violin plot (type = 2).

**Value**

A plot showing likelihood ratio distributions under relatedness and unrelatedness hypothesis.

**Examples**

```

library(forre1)
x = linearPed(2)
x = setMarkers(x, locusAttributes = NorwegianFrequencies[1:5])
x = profileSim(x, N = 1, ids = 2)
datasim = simLRgen(x, missing = 5, 10, 123)
LRdist(datasim)

```

---

LRsex

*Likelihood ratio for age variable*


---

**Description**

Likelihood ratio for age variable

**Usage**

```

LRsex(
  MPs = "F",
  eps = 0.05,
  erRs = eps,
  nsims = 1000,
  Ps = c(0.5, 0.5),
  H = 1,
  LR = FALSE
)

```

**Arguments**

MPs	MP sex
eps	epsilon parameter.
erRs	error rate in the database.
nsims	number of simulations performed.
Ps	Sex probabilities in the population.
H	hypothesis tested, H1: UHR is MP, H2: UHR is no MP
LR	compute LR values

**Value**

A value of Likelihood ratio based on preliminary investigation data. In this case, sex.

**Examples**

```
LRsex()
```

---

makeMPprelim	<i>Make preliminary investigation MP data simulations: a function for obtaining a database of preliminary investigation data for a missing person search.</i>
--------------	---

---

### Description

Make preliminary investigation MP data simulations: a function for obtaining a database of preliminary investigation data for a missing person search.

### Usage

```
makeMPprelim(
  casetype = "children",
  dateinit = "1975/01/01",
  scenario = 1,
  femaleprop = 0.5,
  ext = 100,
  numsims = 10000,
  seed = 123,
  region = c("North America", "South America", "Africa", "Asia", "Europe", "Oceania"),
  regionprob = c(0.2, 0.2, 0.2, 0.1, 0.2, 0.1)
)
```

### Arguments

casetype	Type of missing person search case. Two options are available: "migrants" or "children".
dateinit	Minimum birth date of simulated missing person. Casetype: Children.
scenario	Birth date distribution scenarios: (1) non-uniform, (2) uniform. Casetype: Children.
femaleprop	Proportion of females. Casetype: All.
ext	Time extension for minimum birth date, range in scenario 1 and days in scenario 2. Casetype: Children.
numsims	Number of simulated MPs. Casetype: All.
seed	Select a seed for simulations. If it is defined, results will be reproducible. Casetype: All.
region	Birth region or place in missing children case or place of place of the last seen in missing migrant case. Casetype: All.
regionprob	Region proportions. Casetype: All.

### Value

An object of class data.frame with preliminary investigation data.



**Examples**

```
makeMPprelim()
```

---

makePOIgen	<i>Make POIs gen: a function for obtaining a database with genetic information from simulated POIs or UHRs.</i>
------------	---

---

**Description**

Make POIs gen: a function for obtaining a database with genetic information from simulated POIs or UHRs.

**Usage**

```
makePOIgen(numsims = 100, reference, seed = 123)
```

**Arguments**

numsims	Number of simulations performed (numer of POIs or UHRs).
reference	Indicate the reference STRs/SNPs frequency database used for simulations.
seed	Select a seed for simulations. If it is defined, results will be reproducible. Suggested, seed = 123

**Value**

An object of class data.frame with genetic information from POIs (randomly sampled from the frequency database).

**Examples**

```
library(forrel)
freqdata <- getfreqs(Argentina)
makePOIgen(numsims = 100, reference = freqdata, seed = 123)
```

---

makePOIprelim	<i>Make preliminary investigation POI/UHR data simulations: a function for obtaining a database of preliminary investigation data for a missing person search.</i>
---------------	--

---

## Description

Make preliminary investigation POI/UHR data simulations: a function for obtaining a database of preliminary investigation data for a missing person search.

## Usage

```
makePOIprelim(
  casetype = "children",
  dateinit = "1975/01/01",
  scenario = 1,
  femaleprop = 0.5,
  ext = 100,
  numsims = 10000,
  seed = 123,
  birthprob = c(0.09, 0.9, 0.01),
  region = c("North America", "South America", "Africa", "Asia", "Europe", "Oceania"),
  regionprob = c(0.2, 0.2, 0.2, 0.1, 0.2, 0.1)
)
```

## Arguments

casetype	Type of missing person search case. Two options are available: "migrants" or "children".
dateinit	Minimum birth date of simulated persons of interest. Casetype: Children.
scenario	Birth date distribution scenarios: (1) non-uniform, (2) uniform. Casetype: Children.
femaleprop	Proportion of females. Casetype: All.
ext	Time extension for minimum birth date, range in scenario 1 and days in scenario 2. Casetype: Children.
numsims	Number of simulated POIs/UHRs. Casetype: All.
seed	Select a seed for simulations. If it is defined, results will be reproducible. Casetype: All.
birthprob	Birth type probabilities: home birth, hospital birth and unknown-adoption. Casetype: Children.
region	Birth region or place in missing children case or place of discovery of the human remain in missing migrant case. Casetype: All.
regionprob	Region proportions. Casetype: All.

**Value**

An object of class `data.frame` with preliminary investigation data.

**Examples**

```
makePOIprelim(  
  dateinit = "1975/01/01",  
  scenario = 1,  
  femaleprop = 0.5,  
  ext = 100,  
  numsims = 10000,  
  seed = 123,  
  birthprob = c(0.09, 0.9, 0.01),  
  region = c("North America", "South America", "Africa", "Asia", "Europe", "Oceania"),  
  regionprob = c(0.2, 0.2, 0.2, 0.1, 0.2, 0.1))
```

---

mispiApp	<i>Missing person shiny app</i>
----------	---------------------------------

---

**Description**

Missing person shiny app

**Usage**

```
mispiApp()
```

**Value**

An user interface for computing non-genetic LRs and conditioned probability tables.

**Examples**

```
CPT_MP()
```

---

multi_kl_divergence	<i>Multi-dataset Kullback-Leibler Divergence Calculation</i>
---------------------	--

---

**Description**

This function calculates the Kullback-Leibler divergence for all pairs of provided datasets, considering allele frequencies. It normalizes data, adjusts zero frequencies, and computes KL divergence in both directions for each pair.

**Usage**

```
multi_kl_divergence(datasets, minFreq = 1e-10)
```

**Arguments**

`datasets` List of dataframes, each containing allele frequencies for different populations.  
`minFreq` Minimum frequency to be considered for unobserved or poorly observed alleles.

**Value**

A matrix containing the Kullback-Leibler divergence for each dataset pair.

**Examples**

```
kl_matrix <- multi_kl_divergence(list(Argentina, BosniaHerz, Europe))
```

---

postSim

*postSim: A function for simulating posterior odds*

---

**Description**

postSim: A function for simulating posterior odds

**Usage**

```
postSim(
  datasim,
  Prior = 0.01,
  PriorModel = c("prelim", "uniform")[1],
  eps = 0.05,
  erRs = 0.01,
  epc = Cmodel(),
  erRc = Cmodel(),
  MPc = 1,
  epa = 0.05,
  erRa = 0.01,
  MPa = 10,
  MPr = 2
)
```

**Arguments**

`datasim` Output from simLRgen function.  
`Prior` Prior probability for H1  
`PriorModel` Prior odds model: "prelim" is based on preliminary data, and "uniform" uses only the prior probability of H1

eps	epsilon parameter sex
erRs	error parameter sex
epc	epsilon parameter hair color
erRc	error parameter hair color
MPc	Missing person hair color
epa	epsilon parameter age
erRa	error parameter age
MPa	Missing person age
MPr	Missing person age error range

**Value**

A value of posterior odds.

**Examples**

```
library(forrel)
x = linearPed(2)
plot(x)
x = setMarkers(x, locusAttributes = NorwegianFrequencies[1:5])
x = profileSim(x, N = 1, ids = 2)
datasim = simLRgen(x, missing = 5, 10, 123)
postSim(datasim)
```

---

simLRgen	<i>Simulate likelihoods ratio (LRs) based on genetic data: a function for obtaining expected LRs under relatedness and unrelatedness kinship hypothesis.</i>
----------	--

---

**Description**

Simulate likelihoods ratio (LRs) based on genetic data: a function for obtaining expected LRs under relatedness and unrelatedness kinship hypothesis.

**Usage**

```
simLRgen(reference, missing, numsims, seed, numCores = 1)
```

**Arguments**

reference	Reference pedigree. It could be an input from read_fam() function or a pedigree built with pedtools.
missing	Missing person ID/label indicated in the pedigree.
numsims	Number of simulations performed.
seed	Select a seed for simulations. If it is defined, results will be reproducible. Suggested, seed = 123
numCores	Enables parallelization

**Value**

An object of class data.frame with LRs obtained for both hypothesis, Unrelated where POI is not MP or Related where POI is MP.

**Examples**

```
library(forrel)
x = linearPed(2)
plot(x)
x = setMarkers(x, locusAttributes = NorwegianFrequencies[1:5])
x = profileSim(x, N = 1, ids = 2)
datasim = simLRgen(x, missing = 5, 10, 123)
```

---

simLRprelim	<i>Simulate likelihoods ratio (LRs) based on preliminary investigation data: a function for obtaining expected LRs under relatedness and unrelatedness kinship hypothesis.</i>
-------------	--

---

**Description**

Simulate likelihoods ratio (LRs) based on preliminary investigation data: a function for obtaining expected LRs under relatedness and unrelatedness kinship hypothesis.

**Usage**

```
simLRprelim(
  vartype,
  numsims = 1000,
  seed = 123,
  int = 5,
  ErrorRate = 0.05,
  alphaBdate = c(1, 4, 60, 11, 6, 4, 4),
  numReg = 6,
  MP = NULL,
  database,
  cuts = c(-120, -30, 30, 120, 240, 360)
)
```

**Arguments**

vartype	Indicates type of preliminary investigation variable. Options are: sex, region, age, birthDate and height.
numsims	Number of simulations performed.
seed	Seed for simulations.
int	Interval parameter, used for height and age vartypes. It defines the estimation range, for example, if MP age is 55, and int is 10, the estimated age range will be between 45 and 65.

ErrorRate	Error rate for sex, region, age and Height LR calculations.
alphaBdate	Vector containing alpha parameters for Dirichlet distribution. Usually they are the frequencies of the solved cases in each category.
numReg	Number of regions present in the case.
MP	Introduce the preliminary data of the selected variable (vartype) of the MP. If it is null, open search is carried out. If it is not NULL, close search LR is computed. Variables values must be named as those presented in makePOIprelim function.
database	It is used when the close search (MP not NULL), is carried out. It could be the output from makePOIprelim or a database with the same structure.
cuts	Value of differences between DBD and ABD used for category definition. They must be the same as the ones selected for alphaBdate vector.

**Value**

An object of class data.frame with LRs obtained for both hypothesis, Unrelated where POI/UHR is not MP or Related where POI/UHR is MP.

**Examples**

```
library(mispitools)
simLRprelim("sex")
```

---

Trates	<i>Threshold rates: a function for computing error rates and Matthews correlation coefficient of a specific LR threshold.</i>
--------	---

---

**Description**

Threshold rates: a function for computing error rates and Matthews correlation coefficient of a specific LR threshold.

**Usage**

```
Trates(datasim, threshold)
```

**Arguments**

datasim	Input dataframe containing expected LRs for related and unrelated POIs. It should be the output from makeLRsims function.
threshold	Likelihood ratio threshold selected for error rates calculation.

**Value**

Values of false positive and false negative rates and MCC for a specific LR threshold.

**Examples**

```
library(forre1)
x = linearPed(2)
x = setMarkers(x, locusAttributes = NorwegianFrequencies[1:5])
x = profileSim(x, N = 1, ids = 2)
datasim = simLRgen(x, missing = 5, 10, 123)
Trates(datasim, 10)
```

---

USA

*STRs allelic frequencies from specified country.*

---

**Description**

STRs allelic frequencies from specified country.

**Usage**

USA

**Format**

A data frame allele frequencies



# Index

## \* datasets

- Argentina, [2](#)
- Asia, [3](#)
- Austria, [3](#)
- BosniaHerz, [4](#)
- China, [4](#)
- Europe, [10](#)
- Japan, [11](#)
- USA, [24](#)

Argentina, [2](#)

Asia, [3](#)

Austria, [3](#)

bidirectionalKL, [3](#)

BosniaHerz, [4](#)

China, [4](#)

Cmodel, [5](#)

combLR, [6](#)

CondPlot, [7](#)

CPT\_MP, [7](#)

CPT\_POP, [8](#)

deplot, [9](#)

DeT, [9](#)

Europe, [10](#)

getfreqs, [10](#)

Japan, [11](#)

LRage, [11](#)

LRcol, [12](#)

LRdate, [13](#)

LRdist, [14](#)

LRsex, [15](#)

makeMPprelim, [16](#)

makePOIgen, [17](#)

makePOIprelim, [18](#)

mispiApp, [19](#)

multi\_kl\_divergence, [19](#)

postSim, [20](#)

simLRgen, [21](#)

simLRprelim, [22](#)

Trates, [23](#)

USA, [24](#)