

# Package: KINSIMU (via r-universe)

September 15, 2024

**Type** Package

**Title** Panel Evaluation in Forensic Kinship Analysis

**Version** 0.1.2-2

**Date** 2024-06-15

**Maintainer** Guanju Ma <mache124@126.com>

**Description** Evaluate specific panels in different aspects: i) Simulation tools related to pedigree researches; ii) calculation for systemic effectiveness indicators, such as probability of exclusion (PE).

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.2.3

**Depends** R (>= 3.5.0)

**Imports** utils

**NeedsCompilation** no

**Author** Guanju Ma [aut, cre], Shujin Li [ctb]

**Date/Publication** 2024-06-14 02:40:02 UTC

**Repository** <https://guanju-ma.r-universe.dev>

**RemoteUrl** <https://github.com/cran/KINSIMU>

**RemoteRef** HEAD

**RemoteSha** e8a37c814cc60455b573571064a38dfa650f6238

## Contents

EvaluatePanel . . . . .	2
FortytwoSTR . . . . .	3
IICAL . . . . .	4
logLR . . . . .	5
LRgpgcam . . . . .	6

LRhsip . . . . .	8
LRparas . . . . .	9
outputCSV . . . . .	10
pairsimu . . . . .	11
pediexample . . . . .	12
pedisimu . . . . .	12
testsimulation . . . . .	13
trioPI . . . . .	14

<b>Index</b>	<b>17</b>
--------------	-----------

---

EvaluatePanel	<i>Evaluate a panel</i>
---------------	-------------------------

---

## Description

Transfer the input allele frequency data (in a .csv file or a data frame) into form usable for other functions and then calculate several population parameters based on the frequency data for each marker.

## Usage

```
EvaluatePanel(
  type = "csv",
  strpath,
  raremode = "ISFG",
  Nind = "lastrow",
  Th = 0.01
)
```

## Arguments

type	The type of input data, either 'csv'(a .csv containing af matrix, with the marker names listed in the first row), or 'df' (a data frame containing the af matrix with the marker names as column names)
strpath	pathway of the .csv file or the name of the data frame
raremode	mode of the calculation method of rare alleles, with a default of "ISFG", indicating the recommended method from ISFG, i.e., $(X+1)/(2N+1)$ , where X and N stood for the number of allele types detected in a survey and the sample size, respectively. Three alternative methods are given: "MAF", take the minimum allele frequency as such frequency; "1/2N", take the minimum of possible frequency a survey can achieved; and a number .
Nind	mode of sample size, with a default of "lastrow", meaning that the sample size was presented in the last row of the .csv file. An alternative method is given, i.e., input a unified sample size.
Th	The threshold for the difference in allele frequency sum at a locus with 1, to detect data error from rounding error when the frequency sum does not equal 1. Loci exceeding this threshold will be excluded from the calculation.

**Value**

list of four vectors: `afmatrix`, a list of allele frequency data of each locus; `rare`, a data.frame containing the frequency of rare allele on each locus; `indicators`, a data.frame containing parameters of system efficiency for each locus; `panelparas`, a data.frame containing system efficiency parameters for the whole panel, with the form of  $\log_{10}(1-\text{paramter})$  to avoid the situation that the parameters being displayed as 1 because they were too close to 1

**Examples**

```
#A .csv file can be output with FortytwoSTR data
path<-tempdir()
outputCSV(FortytwoSTR,file.path(path,'data.csv'))
#Example 1, 'df' type, by read the csv file into a data frame
allele_data <- read.csv(file = file.path(path,'data.csv'), header = TRUE)
STR42<- EvaluatePanel(type = 'df',strpath = allele_data,raremode = "ISFG",Nind = "lastrow")
#Example 2, 'csv' type, the same evaluation can be done by directly input the csv file
STR42_2 <- EvaluatePanel(type = 'csv',strpath = file.path(path,'data.csv'),
raremode = "ISFG",Nind = "lastrow")
#The data "FortytwoSTR" is generated with these codes.
```

---

FortytwoSTR

*Information of 42 autosomal STRs*

---

**Description**

Example of the results achieved by "EvaluatePanel" function

**Usage**

FortytwoSTR

**Format**

A list of 3 vectors: i) a list named `afmatrix` containing 42 data.frames, each of which exhibiting the allele frequencies of alleles on a locus with the row names being the corresponding names of such alleles; ii) a data.frame with 1 row and 42 columns containing the frequencies of rare alleles on the 42 loci; and iii) a data.frame named `indicators` with 42 columns and 12 rows containing the main efficiency indicators of the 42 loci.

IICAL

*Incest index***Description**

Calculation of the ratio for a parent-child pair between the probability that the child's other parent is a relative of the present parent to the probability that the child's parents are unrelated

**Usage**

```
IICAL(Parent, Child, af, rare = NULL, allelename = FALSE, phi = 0.25)
```

**Arguments**

Parent	Genotypes of individual A of each case, which should be data.frame with 2 columns and ss rows, where ss stand for sample size
Child	Genotypes of individual B of each case, which should be data.frame with 2 columns and ss rows, where ss stand for sample size
af	name of allele frequency matrix, a data.frame of 1 column with the allele name being row names
rare	frequency of rare allele
allelename	if TRUE, the input genotype data would be regarded as allelenames, otherwise, the position in the af matrix
phi	kinship coefficient between the parents under Hp (that under Hd equals to 0), with a default of 0.25, i.e., father-daughter incest or full-sibling incest

**Details**

The premise of using this function should be the confirmation of the parent-child relationship between the two individuals, if there is no sharing alleles between them,  $1-2\phi$  would be regarded as the output

**Value**

a data.frame containing 2 columns: Ngs (the genotype similarity score, 1 if both of the two alleles in child's genotype can be inherited from the mother) and Log10LR for each simulation

**Examples**

```
# Simulate 10,000 mother-child pairs with father-daughter incest with pedisimu() function
# based on the 42 STRs in "FortytwoSTR" data.
pedi<-data.frame(Person=c("F", "M", "C"), Father=c("RI", "F", "F"), Mother=c("RI", "RI", "M"))
II_1=II_2=data.frame(Ngs=rep(0,10000), IIphi=rep(0,10000))
for (i in 1:42) {
  Genotype1<-pedisimu(af = FortytwoSTR$afmatrix[[i]], ss = 10000, pedi = pedi)
  # Calculate II for each case.
  II_1<-II_1+IICAL(Parent = Genotype1[, 3:4], Child = Genotype1[, 5:6], af=FortytwoSTR$afmatrix[[i]],
```

```

rare=FortytwoSTR$rare[i][1,1],phi=0.25)
#Simulate 10,000 non-inbred mother-child pairs
Genotype2<-pairsimu(af = FortytwoSTR$afmatrix[[i]],ss = 10000,delta = c(0,1,0),allelename = FALSE)
II_2<-II_2+IICAL(Parent = Genotype2[,1:2],Child = Genotype2[,3:4],af=FortytwoSTR$afmatrix[[i]],
rare=FortytwoSTR$rare[i][1,1],phi=0.25)
}
# histograms of CII distributions in the two groups
xmin<-floor(min(min(II_1$Iphi),min(II_2$Iphi)))
xmax<-ceiling(max(max(II_1$Iphi),max(II_2$Iphi)))
par(mfrow = c(1,2))
hist(II_2$Iphi,xlab = expression(log[10]~CII),main = "Non-inbred cases",
xlim = c(xmin,xmax), col = "red")
hist(II_1$Iphi,xlab = expression(log[10]~CII),main = "Inbred cases",
xlim = c(xmin,xmax), col = "blue")

```

---

logLR

*CLR for a single case*


---

## Description

CLR calculation for a single case, where two individual participated and labeled as A and B

## Usage

```

logLR(
  AB,
  afmatrix = NULL,
  rare = NULL,
  allelename = FALSE,
  stepPI = FALSE,
  adelat3 = NULL,
  adelat9 = NULL,
  mu = 0.002
)

```

## Arguments

AB	Genotypes of two individuals in a case, which should be data.frame with 4 columns (2 for each individual) and n1 rows, where n1 stand for number of loci. The row names should be the name of each locus;
afmatrix	name of allele frequency list, which can be loaded with "EvaluatePanel" function
rare	a data frame containing the frequency of rare allele on the locus, with 1 row and multiple columns, each column for a marker;
allelename	if TRUE, the input genotype data would be regarded as allelenames, otherwise, the position in the afmatrix
stepPI	If TRUE, empirical decreasing model of STR mutation would be taken when paternity index is needed to be calculated, otherwise, mutation rate would be taken as PI if IBS=0 between an alleged PC pair.

adelta3	distributions of the IBD coefficient of the outbred plaintiff's hypotheses in LR calculation, which should be a data.frame with 3 columns and x rows, where x stood for the number of such LR being calculated. The names of columns should be "k0", "k1" and "k2", and those of rows the name of LRs
adelta9	distributions of the Jacquard coefficient of the inbred plaintiff's hypotheses in LR calculation, which should be a data.frame with 9 columns and x rows, where x stood for the number of such LR being calculated. The names of columns should be "D1-D9", and those of rows the name of LRs
mu	mutation rate when paternity index is needed to be calculated, defaults to 0.002.

### Value

a list of two data.frames: (i) results\_on\_each\_marker: multiple types of parameters calculated on each marker, including IBS and multiple log10 of LRs; (ii) total\_results\_of\_the\_case: the CIBS and log10CLR results for the whole case.

### Examples

```
# example code
AB<-data.frame(a=rep(0,42),b=rep(0,42),c=rep(0,42),d=rep(0,42))
for (i in 1:42) {
temp<-pairsimu(af = FortytwoSTR$afmatrix[[i]],ss = 1,delta = c(0,1,0),allelename = FALSE)
AB[i,]=temp
rownames(AB)[i]=names(FortytwoSTR$afmatrix)[i]
}
adelta3<-data.frame(k0=c(0,0.25,0.5),k1=c(1,0.5,0.5),k2=c(0,0.25,0),row.names = c("PC", "FS", "HS"))
adelta9<-data.frame(D1=0,D2=0,D3=0,D4=0,D5=0.25,D6=0,D7=0.25,D8=0.5,D9=0,row.names = "FIMCpair")
results<-logLR(AB=AB,afmatrix=FortytwoSTR$afmatrix,rare=FortytwoSTR$rare,stepPI=TRUE,
adelta3=adelta3,adelta9=adelta9)
results$total_results_of_the_case
```

---

LRgpgcam

*LR in grandparent-child identification with reference*


---

### Description

LR when a child (C) is alleged to be a grand-child of another individual (GP), and an offspring(A) of the alleged grandparent participated, with or without the assistant of B's other parent (M). Hp is that, C is an offspring of A's full-sibling; and Hd that, C is unrelated to GP and A. Inbreeding is not considered.

### Usage

```
LRgpgcam(A, C, GP, af, rare = NULL, allelename = FALSE, M = NULL)
```

**Arguments**

A	Genotype data of the alleged uncle/aunt, should be data.frame with 2 columns and ss rows, where ss stand for sample size;
C	Genotype data of the child, with the same form with A
GP	Genotype data of the alleged grandparent of C, with the same form with A
af	name of allele frequency matrix, a data.frame of 1 column containing frequencies with allele names being row names
rare	frequency of rare allele on the locus
allelename	if TRUE, the input genotype data would be regarded as allelenames, otherwise, the position in the af matrix
M	Genotype data of the other parent of C, which can be NULL or as that of A

**Details**

There might be no allele sharing between GP and A, or between M and B, if so, the related part in the LR calculation would be treated as 0, which can be further optimized in future version.

**Value**

a data frame with one column and ss rows, containing log10 value of the CLR of each case

**Examples**

```
# Construct pedi data.frames for two types of pedigrees
pedi1 <- data.frame(Person=c("GF", "GM", "F", "A", "M", "C"),
  Father=c("RI", "RI", "GF", "GF", "RI", "F"),
  Mother=c("RI", "RI", "GM", "GM", "RI", "M"))
pedi2 <- data.frame(Person=c("GF", "GM", "F", "A", "M", "C"),
  Father=c("RI", "RI", "RI", "GF", "RI", "F"),
  Mother=c("RI", "RI", "RI", "GM", "RI", "M"))
LR_1=LR_2=data.frame(Log10CLR=rep(0,10000))
for (i in 1:42) {
  # Simulate 10000 group of pedigrees where the Hp is true
  Genotype <- pedisimu(af = FortytwoSTR$afmatrix[[i]],ss = 10000,pedi = pedi1)
  LR_1 <-LR_1+ LRgpgcam(A=Genotype[,7:8],C=Genotype[,11:12],GP=Genotype[,1:2],M=Genotype[,9:10],
    af=FortytwoSTR$afmatrix[[i]],rare=FortytwoSTR$rare[i][1,1])
  #Simulate 10000 group of false pedigrees, i.e., P and C is unrelated to GP and A
  Genotype <- pedisimu(af = FortytwoSTR$afmatrix[[i]],ss = 10000,pedi = pedi2)
  LR_2 <-LR_2+ LRgpgcam(A=Genotype[,7:8],C=Genotype[,11:12],GP=Genotype[,1:2],M=Genotype[,9:10],
    af=FortytwoSTR$afmatrix[[i]],rare=FortytwoSTR$rare[i][1,1])
}
xmin<-floor(min(min(LR_1$Log10CLR),min(LR_2$Log10CLR)))
xmax<-ceiling(max(max(LR_1$Log10CLR),max(LR_2$Log10CLR)))
par(mfrow = c(1, 2))
hist(LR_2$Log10CLR,xlab = expression(log[10]~CLR),main = "False pedigree",
  xlim = c(xmin,xmax), col = "red")
hist(LR_1$Log10CLR,xlab = expression(log[10]~CLR),main = "True cases",
  xlim = c(xmin,xmax), col = "blue")
```

---

 LRhsip

*LR in half-sibling identification with the identical parent participated*


---

### Description

LR when a pair of siblings and one of their identical parent participated, Hp and Hd of which being "the other parents of the two siblings being specific related" and "the other parents of them being unrelated". Inbreeding factors are not taken into consideration.

### Usage

```
LRhsip(A, B, P, af, rare = NULL, allelename = FALSE, phi = 0.5)
```

### Arguments

A	Genotype data of the first sibling, should be data.frame with 2 columns and ss rows, where ss stand for sample size;
B	Genotype data of the second sibling, with the same form with A
P	Genotype data of the identical parent of A and B, with the same form with them
af	name of allele frequency matrix, a data.frame of 1 column containing frequencies with allele names being row names
rare	frequency of rare allele on the locus
allelename	if TRUE, the input genotype data would be regarded as allelenames, otherwise, the position in the af matrix
phi	kinship coefficient of the other parents of the two siblings under Hp, with default of 1/2, i.e, being identical or MZ

### Details

Mutation might be found between P with A or B, if so, LR would be output as 1-phi, which can be further optimized in the future version.

### Value

a data frame with one column and ss rows, containing log10 value of the CLR of each case

### Examples

```
# Construct pedi data.frames for two types of pedigrees
pedi1 <- data.frame(Person=c("F", "M", "A", "B"),
  Father=c("RI", "RI", "F", "F"),
  Mother=c("RI", "RI", "M", "M"))
pedi2 <- data.frame(Person=c("M", "A", "B"),
  Father=c("RI", "RI", "RI"),
  Mother=c("RI", "M", "M"))
LR_1=LR_2=data.frame(Log10CLR=rep(0, 10000))
```



```

for (i in 1:42) {
  # Simulate 10000 groups of A/B/P where A is full sibling of B
  Genotype1=pedisimu(af = FortytwoSTR$afmatrix[[i]],ss = 10000,pedi = pedi1)
  LR_1=LR_1+LRhsip(A=Genotype1[,5:6],B=Genotype1[,7:8],P=Genotype1[,3:4],
    af = FortytwoSTR$afmatrix[[i]],rare=FortytwoSTR$rare[i][1,1])
  # Simulate 10000 groups of A/B/P where A is half sibling of B, i.e., the true phi=0
  Genotype2=pedisimu(af = FortytwoSTR$afmatrix[[i]],ss = 10000,pedi = pedi2)
  LR_2=LR_2+LRhsip(A=Genotype2[,3:4],B=Genotype2[,5:6],P=Genotype2[,1:2],
    af = FortytwoSTR$afmatrix[[i]],rare=FortytwoSTR$rare[i][1,1])
}
# histograms of CLR distributions in the two groups
xmin<-floor(min(min(LR_1$Log10CLR),min(LR_2$Log10CLR)))
xmax<-ceiling(max(max(LR_1$Log10CLR),max(LR_2$Log10CLR)))
par(mfrow = c(1, 2))
hist(LR_2$Log10CLR,xlab = expression(log[10]~CLR),main = "False pedigree",
  xlim = c(xmin,xmax), col = "red")
hist(LR_1$Log10CLR,xlab = expression(log[10]~CLR),main = "True cases",
  xlim = c(xmin,xmax), col = "blue")

```

---

LRparas

*Calculating parameters in LR*


---

## Description

Counting and calculating parameters used in the calculation process of different pairwise LR calculation

## Usage

```

LRparas(
  AB,
  af = NULL,
  rare = NULL,
  allelename = FALSE,
  stepwisePI = FALSE,
  bred = FALSE,
  mu = 0.002
)

```

## Arguments

AB	Genotypes of two individuals of each case, which should be data.frame with 4 columns (2 for each individual) and ss rows, where ss stand for sample size;
af	name of allele frequency matrix, a data.frame of 1 column containing frequencies with allele names being row names
rare	frequency of rare allele on the locus

allelename	if TRUE, the input genotype data would be regarded as allelenames, otherwise, the position in the af matrix
stepwisePI	If TRUE, empirical decreasing model of STR mutation would be taken when paternity index is needed to be calculated, otherwise, mutation rate would be taken as PImu if IBS=0 between an alleged PC pair.
bred	if TRUE, parameters used in the calculation of LR in inbred relationship would be calculated
mu	mutation rate for PI calculation, with a default of 0.002

### Details

Note that LR elements are in their original values instead of log10 values used in other functions.

### Value

a data frame with multiple column and ss rows, containing different types of parameters used in different LR calculation according to argument settings

### Examples

```
af = FortytwoSTR$afmatrix[[1]]
AB = pairsimu(af = af,ss = 10000,delta = c(0,1,0),allelename = FALSE)
LRelements<-LRparas(AB=AB, af=af, rare=FortytwoSTR$rare[1][1,1],allelename=FALSE,
  stepwisePI=TRUE,bred=TRUE)
```

---

outputCSV

*Output frequency data into ISFG format*

---

### Description

Output frequency data into ISFG format

### Usage

```
outputCSV(data, strpath)
```

### Arguments

data	frequency data, a list containing 4 data frames.
strpath	name of the pathway of the output csv file

### Value

a .csv file in ISFG formate

**Examples**

```
path<-tempdir()
outputCSV(FortytwoSTR,file.path(path,'data.csv'))
```

---

pairsimu	<i>Pairwise simulation</i>
----------	----------------------------

---

**Description**

Generating genotype combinations of multiple individual pairs with specific relationships on an autosomal marker, ignoring mutation

**Usage**

```
pairsimu(af, ss, delta, allelename = FALSE)
```

**Arguments**

af	name of allele frequency matrix, a data.frame of 1 column containing frequencies with allele names being row names, which can be loaded with "EvaluatePanel" function, not necessary if Parent is not NULL
ss	sample size, i.e., how many individual pairs do you want simulate
delta	distribution of IBD or Jacquard coefficient of specific relationship, i.e., kappa(0-2) or Delta(1-9), respectively. Which should be input in form of single row of data with c() function. It should be noted that the data should be in order of kappa0 to kappa2 or Delta1 to Delta9.
allelename	if TRUE, outputting the names of alleles, otherwise, the positions of them in the af matrix

**Value**

A data.frame with four columns and ss rows, consisting of the alleles of the first individual in the first two columns and the alleles of the second individual in the remaining two columns.

**Examples**

```
# Take the first STR in the 42 STR as example
af = FortytwoSTR$afmatrix[[1]]
# simulating 10,000 unrelated pairs
a<-pairsimu(af = af,ss = 10000,delta = c(1,0,0),allelename = FALSE)
# simulating 10,000 parent-child pairs
b<-pairsimu(af = af,ss = 10000,delta = c(0,1,0),allelename = FALSE)
# simulating 10,000 full-sibling pairs
c<-pairsimu(af = af,ss = 10000,delta = c(0.25,0.5,0.25),allelename = FALSE)
```

---

pediexample	<i>Example of pedi matrix</i>
-------------	-------------------------------

---

**Description**

Example of pedi, a matrix containing inherence information of a pedigree, used in function of "pedisimu"

**Usage**

```
pediexample
```

**Format**

a data.frame containg 3 columns

---

pedisimu	<i>Pedigree simulation</i>
----------	----------------------------

---

**Description**

Generating genotype combinations of multiple pedigrees with specific relationships on an autosomal marker.

**Usage**

```
pedisimu(
  af,
  ss,
  pedi,
  random_name = "RI",
  muf = 0,
  mum = 0,
  allelename = FALSE
)
```

**Arguments**

af	name of allele frequency matrix, a data.frame of 1 column containing frequencies with allele names being row names, which can be loaded with "EvaluatePanel" function, not necessary if Parent is not NULL
ss	sample size, i.e., how many individual pairs do you want simulate
pedi	a matrix in data.frame form containing the pedigree structure information
random_name	the name of random individual, with a default of "RI"

muf	father-child mutation rate
mum	mother-child mutation rate
allelename	if TRUE the output data would be the allele name, which should be the row names of af matrix, otherwise, it would be the position in that matrix

**Value**

A data.frame with *ss* rows and  $2*n$  columns, where *n* is equal to the number of rows in the *pedi* data.frame. Each pair of columns contains alleles of an individual, with the individuals sorted in the same order as in the *pedi* data.frame.

**Examples**

```
#simulating a first cousin pedigree without considring mutation
pedi<-pediexample
af<-FortytwoSTR$afmatrix[[1]]
pedisimu(af = af,ss = 10000,pedi = pedi)
```

---

testsimulation	<i>Simulation in kinship analysis</i>
----------------	---------------------------------------

---

**Description**

Simulation and calculation in kinship analysis, in which genotype combinations of two groups of individual pairs with different relationships would be generated, specific likelihood ratio, as well as the CIBS score would be calculated for each pair.

**Usage**

```
testsimulation(
  afmatrix,
  ss,
  tdelta,
  adelta3 = NULL,
  adelta9 = NULL,
  pedname = "SimPed",
  stepPI = FALSE,
  mu = 0.002
)
```

**Arguments**

afmatrix	name of allele frequency list, which can be loaded with "EvaluatePanel" function
ss	sample size

tdelta	distribution of the actual IBD or Jacquard coefficient of the individual pairs, i.e., kappa(0-2) or Delta(1-9), respectively. Which should be input in form of single row of data with c() function. It should be noted that the data should be in order of kappa0 to kappa2 or Delta1 to Delta9.
adelta3	distributions of the IBD coefficient of the outbred plaintiff's hypotheses in LR calculation, which should be a data.frame with 3 columns and x rows, where x stood for the number of such LR being calculated. The names of columns should be "k0", "k1" and "k2", and those of rows the name of LRs
adelta9	distributions of the Jacquard coefficient of the inbred plaintiff's hypotheses in LR calculation, which should be a data.frame with 9 columns and x rows, where x stood for the number of such LR being calculated. The names of columns should be "D1-D9", and those of rows the name of LRs
pedname	name of the real relationship, defaults to "SimPed".
stepPI	If TRUE, empirical decreasing model of STR mutation would be taken when paternity index is needed to be calculated, otherwise, mutation rate would be taken as PI if IBS=0 between an alleged PC pair.
mu	mutation rate when paternity index is needed to be calculated, defaults to 0.002.

### Value

a data.frame containing multiple columns: relationship, CIBS and multiple log0CLR for each simulation

### Examples

```
adelta3<-data.frame(k0=c(0,0.25,0.5),k1=c(1,0.5,0.5),k2=c(0,0.25,0),row.names = c("PC","FS","HS"))
adelta9<-data.frame(D1=0,D2=0,D3=0,D4=0,D5=0.25,D6=0,D7=0.25,D8=0.5,D9=0,row.names = "FIMCpair")
data(FortytwoSTR)
results<-testsimulation(afmatrix = FortytwoSTR[["afmatrix"]],ss = 10000,tdelta = c(0,1,0),
adelta3 = adelta3, adelta9 = adelta9,pedname="PC")
```

---

trioPI

*LR in standard trio cases*


---

### Description

Calculating LR in cases where 3 participants being available, a child, his/her biological mother (or father) whose parentage is confirmed, and a male (or female) who is unrelated to the confirmed parent and alleged to be specific relative of the child, usually the father. Null hypothesis, i.e., that the alleged participant is unrelated to the child, is taken as Hd.

**Usage**

```
trioPI(
  AR,
  C,
  TP,
  af,
  rare = NULL,
  allelename = FALSE,
  muAtoC = 0.002,
  muTtoC = 0.002/3.5,
  kappa1 = 1
)
```

**Arguments**

AR	Genotype of the alleged relative, which should be data.frame with 2 columns and ss rows, where ss stand for sample size;
C	Genotype of the child, see in AR for the data form
TP	Genotype of the confirmed parent of the child, see in AR for the data form
af	name of allele frequency matrix, a data.frame of 1 column containing frequencies with allele names being row names
rare	frequency of rare allele on the locus
allelename	if TRUE, the input genotype data would be regarded as allelenames, otherwise, the position in the af matrix
muAtoC	mutation rate from AR to C if AR is alleged to be the child's parent, with a default of 0.002, please note that mistakes would be introduced if the mutation rate is larger than 0.2
muTtoC	mutation rate from TP to C, with a default of 0.002/3.5, please note that mistakes would be introduced if the mutation rate is larger than 0.2
kappa1	kappa_1 of the alleged relationship between AR and C, with a default of 1, meaning the AR is alleged to be the other parent of C

**Details**

If any one of TP or AR cannot provide any of the C's alleles through integer steps, muAtoC would be output when calculating PI, and under other situations, the case with minimum mutation steps, (AR to C)+(TP to C) under Hp and (TP to C) under Hd, would be considered. Much more is required for further discussion in PI calculation involving mutations.

**Value**

a data.frame containing ss rows and 1 column, containing the log10 of LR for each group

## Examples

```

# Three types of pedigrees are simulated: pedi1: father-mother-child;
# pedi2: random male-mother-child; and pedi3: uncle-mother-child
pedi1 <- data.frame(Person=c("F","M","C"),Father=c("RI","RI","F"),Mother=c("RI","RI","M"))
pedi2 <- data.frame(Person=c("F","M","C"),Father=c("RI","RI","RI"),Mother=c("RI","RI","M"))
pedi3 <- data.frame(Person=c("GF","GM","AR","F","M","C"),
  Father=c("RI","RI","GF","GF","RI","F"),
  Mother=c("RI","RI","GM","GM","RI","M"))
# Two types of LRs are calculated: PI_1 and PI_2: Paternity index in trio cases;
# AI_1 and AI_2: Avuncular index in trio cases.
PI_1=PI_2=AI_1=AI_2=data.frame(Log10CLR=rep(0,10000))
# Simulation are carried out based on the frequency data of the 42 STRs in FortytwoSTR dataset
# Setting sample size as 10,000
for (i in 1:42) {
  Genotype1<-pedisimu(af = FortytwoSTR$afmatrix[[i]],ss = 10000,pedi = pedi1)
  PI_1<-PI_1+trioPI(AR=Genotype1[,1:2],TP=Genotype1[,3:4],
    C=Genotype1[,5:6],af=FortytwoSTR$afmatrix[[i]],
    rare=FortytwoSTR$rare[i])
  Genotype2<-pedisimu(af = FortytwoSTR$afmatrix[[i]],ss = 10000,pedi = pedi2)
  PI_2<-PI_2+trioPI(AR=Genotype2[,1:2],TP=Genotype2[,3:4],
    C=Genotype2[,5:6],af=FortytwoSTR$afmatrix[[i]],
    rare=FortytwoSTR$rare[i])
  AI_2<-AI_2+trioPI(AR=Genotype2[,1:2],TP=Genotype2[,3:4],
    C=Genotype2[,5:6],af=FortytwoSTR$afmatrix[[i]],
    rare=FortytwoSTR$rare[i],kappa1=0.5)
  Genotype3<-pedisimu(af = FortytwoSTR$afmatrix[[i]],ss = 10000,pedi = pedi3)
  AI_1<-AI_1+trioPI(AR=Genotype3[,5:6],TP=Genotype3[,9:10],
    C=Genotype3[,11:12],af=FortytwoSTR$afmatrix[[i]],
    rare=FortytwoSTR$rare[i],kappa1=0.5)
}
#histogram of the final results
xmin1<-floor(min(min(PI_1$Log10CLR),min(PI_2$Log10CLR)))
xmax1<-ceiling(max(max(PI_1$Log10CLR),max(PI_2$Log10CLR)))
xmin2<-floor(min(min(AI_1$Log10CLR),min(AI_2$Log10CLR)))
xmax2<-ceiling(max(max(AI_1$Log10CLR),max(AI_2$Log10CLR)))
par(mfrow = c(2, 2))
hist(PI_1$Log10CLR,xlab = expression(log[10]~CPI),main = "True parentage cases",
  xlim = c(xmin1,xmax1), col = "blue")
hist(AI_1$Log10CLR,xlab = expression(log[10]~CAI),main = "True avuncular cases",
  xlim = c(xmin2,xmax2), col = "blue")
hist(PI_2$Log10CLR,xlab = expression(log[10]~CPI),main = "False pedigree in parentage cases",
  xlim = c(xmin1,xmax1), col = "red")
hist(AI_2$Log10CLR,xlab = expression(log[10]~CAI),main = "False pedigree in avuncular cases",
  xlim = c(xmin2,xmax2), col = "red")

```



# Index

## \* datasets

FortytwoSTR, 3  
pediexample, 12

EvaluatePanel, 2

FortytwoSTR, 3

IICAL, 4

logLR, 5

LRgpgcam, 6

LRhsip, 8

LRparas, 9

outputCSV, 10

pairsimu, 11

pediexample, 12

pedisimu, 12

testsimulation, 13

trioPI, 14