Package 'sim1000G'

July 19, 2017

Type Package

Title Genotype Simulations for Rare or Common Variants using HAPLOTYPES from 1000 Genomes

Version 1.04

Date 2017-06-08

Author Apostolos Dimitromanolakis, Laurent Briollais

Maintainer Apostolos Dimitromanolakis <apostolis@live.ca>

Description Generates realistic simulated genetic data in families or unrelated individuals.

License GPL (>= 2)

Depends R (>= 2.15.2),
stats,
hapsim,
MASS,
stringr

NeedsCompilation no

RoxygenNote 5.0.1

Suggests knitr,
testthat

VignetteBuilder knitr

Encoding UTF-8

LazyData true

R topics documented:

`sim1000G-package` *Simulations of rare/common variants using haplotype data from 1000 genomes*

Description

(generic description)

`computePairIBD12` *Computes pairwise IBD1/2 for a specific pair of individuals*

Description

Computes pairwise IBD1/2 for a specific pair of individuals

Usage

```
computePairIBD12(i, j)
```

Arguments

`i`	Index of first individual
`j`	Index of second individual

Value

Mean IBD1 and IBD2 as computed from shared haplotypes

`createVCF` *Creates a regional vcf file using bcftools to extract a region from 1000 genomes vcf files*

Description

Creates a regional vcf file using bcftools to extract a region from 1000 genomes vcf files

Usage

```
createVCF()
```

Value

none

`crossoverCDFvector` *Contains recombination model information.*

Description

This vector contains the density between two recombination events, as a cumulative density function.

Usage

```
crossoverCDFvector
```

Format

An object of class `logical` of length 1.

`downloadGeneticMap` *Downloads a genetic map for a particular chromosome under GRCh37 coordinates for use with sim1000G.*

Description

Downloads a genetic map for a particular chromosome under GRCh37 coordinates for use with sim1000G.

Usage

```
downloadGeneticMap(chromosome)
```

Arguments

`chromosome`	Chromosome number to download recombination distances from.

`generateRecombinationDistances` *Generate recombination distances using a chi-square model.*

Description

Generate recombination distances using a chi-square model.

Usage

```
generateRecombinationDistances(n)
```

Arguments

`n`	Number of distances to generate

Value

vector of recombination distances in centimorgan

`generateRecombinationDistances_noInterference` *Generate recombination distances using a no-interference model.*

Description

Generate recombination distances using a no-interference model.

Usage

```
generateRecombinationDistances_noInterference(n)
```

Arguments

`n`	Number of distances to generate

Value

recombination distances in centimorgan

`generateSingleRecombinationVector` *Genetates a recombination vector arising from one meiotic event. The origin of segments is coded as (0 - haplotype1 , 1 - haplotype2)*

Description

Genetates a recombination vector arising from one meiotic event. The origin of segments is coded as (0 - haplotype1 , 1 - haplotype2)

Usage

```
generateSingleRecombinationVector(cm)
```

Arguments

`cm`	The length of the region that we want to generate recombination distances.

geneticMap *Holds the genetic map information that is used for simulations.*

Description

Holds the genetic map information that is used for simulations.

Usage

```
geneticMap
```

Format

An object of class `environment` of length 0.

getCMfromBP *Converts centimorgan position to base-pair.*

Description

Converts centimorgan position to base-pair.

Usage

```
getCMfromBP(bp)
```

Arguments

bp	vector of base-pair positions

newFamilyWithOffspring *Simulates genotypes for 1 family with n offspring*

Description

Simulates genotypes for 1 family with n offspring

Usage

```
newFamilyWithOffspring(family_id, noffspring = 2)
```

Arguments

family_id	What will be the family_id (for example: 100)
noffspring	Number of offsprings that this family will have

Value

family structure object

---

newNuclearFamily *Simulates genotypes for 1 family with 1 offspring*

Description

Simulates genotypes for 1 family with 1 offspring

Usage

```
newNuclearFamily(family_id)
```

Arguments

family_id	What will be the family_id (for example: 100)

Value

family structure object

---

plotRegionalGeneticMap *Generates a plot of the genetic map for a specified region.*

Description

The plot shows the centimorgan vs base-pair positions. In addition it showt the locations of the markers that have been read.

Usage

```
plotRegionalGeneticMap(bp)
```

Arguments

bp	Vector of base-pair positions to generate a plot for

printMatrix *Prints a matrix*

Description

Prints a matrix

Usage

```
printMatrix(m)
```

Arguments

m	Matrix to be printed

readGeneticMap *Reads a genetic map downloaded from the function downloadGeneticMap.*

Description

The map contains a complete chromosome to be used for simulations. The file must be downloaded using the function downloadGeneticMap.

Usage

```
readGeneticMap(chromosome, dir = ".")
```

Arguments

chromosome	Chromosome number to download recombination distances from.
dir	Directory the map file is located.

readGeneticMapFromFile *Reads a genetic map to be used for simulations.*

Description

The file must be contain the following columns in order: chromosome, basepaire, rate(not used), centimorgan

Usage

```
readGeneticMapFromFile(filelocation)
```

Arguments

filelocation	Filename containing the genetic map

readVCF *Read a vcf file, with options to filter out low or high frequency markers.*

Description

Read a vcf file, with options to filter out low or high frequency markers.

Usage

```
readVCF(filename = "haplosims/1.vcf", thin = 1, maxNumberOfVariants = 400,
  min_maf = 0.02, max_maf = NA)
```

Arguments

filename	Input VCF file
thin	How much to thin markers
maxNumberOfVariants	Maximum number of variants to keep from region
min_maf	Minimum allele frequency of markers to keep
max_maf	Maximum allele frequency of markers to keep

Value

none

SIM *Holds data necessary for a simulation.*

Description

Holds data necessary for a simulation.

Usage

```
SIM
```

Format

An object of class `environment` of length 5.

startSimulation *Start the simulation*

Description

Start the simulation

Usage

```
startSimulation(vcf, totalNumberOfIndividuals = 250, randomdata = 0)
```

Arguments

`vcf`	Input vcf file of a region (can be .gz). Must contain phased data.
`totalNumberOfIndividuals`	Maximum Number of individuals that will ever be generated
`randomdata`	If 1, disregards the genotypes in the vcf file and generates markers that are not in LD. Generally do not use.

`writePED` *Writes a plink compatible PED/MAP file from the simulated genotypes*

Description

Writes a plink compatible PED/MAP file from the simulated genotypes

Usage

```
writePED(vcf, fam, filename = "out")
```

Arguments

`vcf`	vcf object used in simulation
`fam`	Individuals / families to be written
`filename`	Basename of output files (.ped/.map will be added automatically)

Index