Package: PlasmoSim (via r-universe)

June 9, 2024

Type Package

Title Simulation of Plasmodium genetic data

Version 1.0.0

Description A basic Plasmodium simulator. Contains functions to simulate epidemiological and genetic data from a simple model of Plasmodium falciparum transmission.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

LinkingTo Rcpp

Imports Rcpp, dplyr, openssl, rlang, knitrProgressBar

SystemRequirements C++11

BugReports https://github.com/mrc-ide/PlasmoSim/issues

Suggests knitr, rmarkdown, tidyverse, kableExtra, testthat (>= 3.0.0)

Config/testthat/edition 3

VignetteBuilder knitr

Repository https://plasmogenepi.r-universe.dev

RemoteUrl https://github.com/mrc-ide/PlasmoSim

RemoteRef HEAD

RemoteSha 40ce6af570927f1e89759805a8d891678b05d85b

Contents

`check_PlasmoSim_loaded` *Check that PlasmoSim package has loaded successfully*

Description

Simple function to check that PlasmoSim package has loaded successfully. Prints "PlasmoSim loaded successfully!" if so.

Usage

```
check_PlasmoSim_loaded()
```

`get_haplotype_identity` *Get proportion identical between two haplotype matrices*

Description

Compare two sets of haplotypes (matrices), and return the proportion of identical sites over all pairwise comparisons. Values can be any numeric value; for example if values represent ancestry then this function returns the average identity by descent, or if values represent alleles then it returns the average identity by state.

Usage

```
get_haplotype_identity(mat1, mat2)
```

Arguments

`mat1`, `mat2`	matrices representing sets of haplotypes to compare. Haplotypes are in rows and loci are in columns.

get_identity_matrix *Get pairwise genetic identity matrix*

Description

Calculates pairwise genetic identity between all samples. If `deme_level = TRUE` this is averaged over all individuals within a deme, to produce average pairwise relatedness within and between demes. Each time point in the sample is considered independently, and output as a list.

Usage

```
get_identity_matrix(sim_output, deme_level = FALSE)
```

Arguments

`sim_output`	simulation output from `sim_falciparum()`.
`deme_level`	if `TRUE` then return pairwise identity at the deme level, averaged over all individuals within each deme. Otherwise return at the individual level.

get_spatial_distance *Get great circle distance between spatial points*

Description

Get great circle distance between spatial points.

Usage

```
get_spatial_distance(lat, lon)
```

Arguments

`lat`	vector of latitudes.
`lon`	vector of longitudes.

lonlat_to_bearing *Calculate great circle distance and bearing between coordinates*

Description

Calculate great circle distance and bearing between spatial coordinates.

Usage

```
lonlat_to_bearing(origin_lon, origin_lat, dest_lon, dest_lat)
```

Arguments

origin_lon	The origin longitude
origin_lat	The origin latitude
dest_lon	The destination longitude
dest_lat	The destination latitude

Examples

```
# one degree longitude should equal approximately 111km at the equator
lonlat_to_bearing(0, 0, 1, 0)
```

PlasmoSim *PlasmoSim*

Description

A basic Plasmodium simulator. Contains functions to simulate epidemiological and genetic data from a simple model of Plasmodium falciparum transmission.

plasmosim_file *Import file*

Description

Import file from the inst/extdata folder of this package

Usage

```
plasmosim_file(name)
```

Arguments

name	name of file.

`sim_falciparum` *Simulate genetic data from simple P. falciparum model*

Description

Simulate genetic data from a simple model of P. falciparum epidemiology and genetics.

Usage

```
sim_falciparum(
  a = 0.3,
  p = 0.9,
  mu = -log(p),
  u = 12,
  v = 10,
  g = 10,
  prob_infection = 0.1,
  duration_infection = dgeom(1:500, 1/100),
  infectivity = 0.1,
  max_infections = 5,
  H = 1000,
  seed_infections = 100,
  M = 1000,
  mig_matrix = diag(length(M)),
  L = 24,
  mean_oocysts = 2,
  mean_products = 5,
  recomb_prob = 0.1,
  max_time = max(sample_dataframe$time),
  sample_dataframe = data.frame(deme = 1, time = 365, n = 100),
  report_progress = TRUE
)
```

Arguments

`a`	human blood feeding rate. The proportion of mosquitoes that feed on humans each day.
`p`	mosquito probability of surviving one day.
`mu`	mosquito instantaneous death rate. mu = -log(p) unless otherwise specified.
`u`	intrinsic incubation period. The number of days from infection to blood-stage infection in a human host.
`v`	extrinsic incubation period. The number of days from infection to becoming infectious in a mosquito.
`g`	lag time between human blood-stage infection and production of gametocytes.
`prob_infection`	probability a human becomes infected after being bitten by an infected mosquito.

`duration_infection`
: vector specifying probability distribution of time (in days) of a malaria episode.

`infectivity`
: probability a mosquito becomes infected after biting an infective human host.

`max_infections`
: maximum number of infections that an individual can hold simultaneously.

`H`
: human population size, which is assumed to be the same in every deme.

`seed_infections`
: vector specifying the initial number of infected humans in each deme.

`M`
: vector specifying mosquito population size (strictly the number of adult female mosquitoes) in each deme.

`mig_matrix`
: migration matrix specifing the daily probability of migrating from each deme to each other deme. Migration must be equal in both directions, meaning this matrix must be symmetric.

`L`
: number of loci. The maximum number of loci is 1000, as at higher numbers haplotypes begin to exceed integer representation (2^L).

`mean_oocysts`
: the average number of viable oocysts generated from gametocytes upon biting an infective host. The actual number of oocysts is generated from a zero-truncated Poisson distribution with this mean.

`mean_products`
: parasite genotypes are passed from mosquito to host by sampling N times with replacement from the available oocysts products (the available number of products is 4 times the number of oocysts). N is drawn independently for each infection from a zero-truncated Poisson distribution with mean given by `mean_products`. Hence, large values of this parameter increase the chance of co-transmission of multiple genotypes, while small values increase the chance of picking up just a single genotype.

`recomb_prob`
: the probability of a recombination breakpoint between any sequential pair of loci. Assumed to be the same for all loci.

`max_time`
: number of days in the simulation.

`sample_dataframe`
: a dataframe specifying outputs from the model. Must contain the following three columns:

  1. `deme`: which numbered deme to sample from.
  2. `time`: the day on which samples are taken.
  3. `n`: the number of hosts to randomly sample (without replacement) from the population.

`report_progress`
: if `TRUE` then a progress bar is printed to the console during simuation.

Index