

Package: moire (via r-universe)

June 28, 2024

Title Multiplicity of Infection and Allele Frequency Recovery from Noisy Polyallelic Genetics Data

Version 3.3.2

Description A Markov Chain Monte Carlo (MCMC) based approach to Bayesian estimation of individual level multiplicity of infection, within host relatedness, and population allele frequencies from polyallelic genetic data.

License GPL (>= 3)

Encoding UTF-8

LazyData true

LazyDataCompression bzip2

SystemRequirements C++17, GNU make

LinkingTo Rcpp, RcppProgress, RcppParallel, BH

Imports Rcpp, RcppProgress, RcppParallel, dplyr, tidyr, stats, purrr, rlang, ggplot2,

URL <https://github.com/EPPIcenter/moire>,
<https://eppicenter.github.io/moire/>,
<https://eppicenter.ucsf.edu/resources>

BugReports <https://github.com/EPPIcenter/moire/issues>

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

Suggests knitr, rmarkdown, markdown, forcats, testthat (>= 3.0.0)

VignetteBuilder knitr

Depends R (>= 4.0.0)

Config/testthat/edition 3

Repository <https://eppicenter.r-universe.dev>

RemoteUrl <https://github.com/eppicenter/moire>

RemoteRef HEAD

RemoteSha 13b35e2f2f0194d2fa37833883cd4bf37b680a53

Contents

calculate_he	2
calculate_med_allele_freqs	3
calculate_naive_allele_frequencies	3
calculate_naive_coi	4
calculate_naive_coi_offset	4
load_delimited_data	5
load_long_form_data	5
mcmc_results	6
plot_chain_swaps	6
rdirichlet	7
run_mcmc	7
simulated_data	9
simulate_allele_frequencies	10
simulate_data	10
simulate_observed_allele	11
simulate_observed_genotype	12
simulate_sample_coi	12
simulate_sample_genotype	13
summarize_allele_freqs	14
summarize_allele_freq_fn	14
summarize_coi	15
summarize_effective_coi	16
summarize_epsilon_neg	16
summarize_epsilon_pos	17
summarize_he	18
summarize_relatedness	18
Index	20

calculate_he

Calculate the expected heterozygosity from allele frequencies

Description

Calculate the expected heterozygosity from allele frequencies

Usage

```
calculate_he(allele_freqs)
```

Arguments

allele_freqs Simplex of allele frequencies

`calculate_med_allele_freqs`

Calculate the geometric median of the posterior distribution of allele frequencies

Description

Calculate the geometric median of the posterior distribution of allele frequencies

Usage

```
calculate_med_allele_freqs(mcmc_results, merge_chains = TRUE)
```

Arguments

`mcmc_results` Result of calling `run_mcmc()`
`merge_chains` boolean indicating that all chain results should be merged

Details

Returns the geometric median of the posterior distribution, defined as the point minimizing the L2 distance from each sampled point.

`calculate_naive_allele_frequencies`

Calculate naive allele frequencies

Description

Calculate naive allele frequencies

Usage

```
calculate_naive_allele_frequencies(data)
```

Arguments

`data` List of lists of numeric vectors, where each list element is a collection of observations across samples at a single genetic locus

Details

Estimate naive allele frequencies from the empirical distribution of alleles

calculate_naive_coi *Calculate naive COI*

Description

Calculate naive COI

Usage

```
calculate_naive_coi(data)
```

Arguments

data List of lists of numeric vectors, where each list element is a collection of observations across samples at a single genetic locus.

Details

Estimates the complexity of infection using a naive approach that chooses the highest number of observed alleles.

calculate_naive_coi_offset
 Calculate naive COI offset

Description

Calculate naive COI offset

Usage

```
calculate_naive_coi_offset(data, offset)
```

Arguments

data List of lists of numeric vectors, where each list element is a collection of observations across samples at a single genetic locus.

offset Numeric offset – n`th highest number of observed alleles

Details

Estimates the complexity of infection using a naive approach that chooses the n`th highest number of observed alleles.

load_delimited_data *Load delimited data*

Description

Load delimited data

Usage

```
load_delimited_data(data, sep = ";", warn_uninformative = TRUE)
```

Arguments

data data.frame containing the described data
sep string used to separate alleles
warn_uninformative boolean whether or not to print message when removing uninformative loci

Details

Load data.frame with a sample_id column and the remaining columns are loci. Each cell contains a separator delimited string representing the observed alleles at that locus for that sample. Returned data contains vectors sample_ids and loci that are ordered as the results will be ordered from running the MCMC algorithm.

load_long_form_data *Load long form data*

Description

Load long form data

Usage

```
load_long_form_data(df, warn_uninformative = TRUE)
```

Arguments

df data frame with 3 columns: sample_id, locus, allele. Each row is a single observation of an allele at a particular locus for a given sample.
warn_uninformative boolean whether or not to print message when removing uninformative loci

Details

Long form data is a data frame with 3 columns: `sample_id`, `locus`, `allele`. Returned data contains vectors `sample_ids` and `loci` that are ordered as the results will be ordered from running the MCMC algorithm.

<code>mcmc_results</code>	<i>MCMC results from using the packaged simulated data and calling <code>run_mcmc()</code></i>
---------------------------	--

Description

MCMC results from using the packaged simulated data and calling `run_mcmc()`

Usage

```
mcmc_results
```

Format

An object of class `list` of length 3.

<code>plot_chain_swaps</code>	<i>Plot chain swap acceptance rates</i>
-------------------------------	---

Description

Plot chain swap acceptance rates

Usage

```
plot_chain_swaps(mcmc_results)
```

Arguments

`mcmc_results` list of results from `run_mcmc`

Details

Plot the swap acceptance rates for each chain. The x-axis is the temperature, and the y-axis is the swap acceptance rate. The dashed lines indicate the temperatures used for parallel tempering.

Value

list of ggplot objects

rdirichlet	<i>Dirichlet distribution</i>
------------	-------------------------------

Description

Dirichlet distribution

Usage

```
rdirichlet(n, alpha)
```

Arguments

n	total number of draws
alpha	vector controlling the concentration of simplex

Details

Implementation of random sampling from a Dirichlet distribution

run_mcmc	<i>Sample from the target distribution using MCMC</i>
----------	---

Description

Sample from the target distribution using MCMC

Usage

```
run_mcmc(  
  data,  
  is_missing = FALSE,  
  allow_relatedness = TRUE,  
  thin = 1,  
  burnin = 10000,  
  samples_per_chain = 1000,  
  verbose = TRUE,  
  use_message = FALSE,  
  eps_pos_alpha = 1,  
  eps_pos_beta = 1,  
  eps_neg_alpha = 1,  
  eps_neg_beta = 1,  
  r_alpha = 1,  
  r_beta = 1,  
  mean_coi_shape = 0.1,
```

```

mean_coi_scale = 10,
max_eps_pos = 2,
max_eps_neg = 2,
max_coi = 40,
num_chains = 1,
num_cores = 1,
pt_chains = 1,
pt_grad = 1,
pt_num_threads = 1,
adapt_temp = TRUE,
pre_adapt_steps = 25,
temp_adapt_steps = 25,
max_initialization_tries = 10000
)

```

Arguments

<code>data</code>	Data to be used in MCMC, as generated by the <code>load*_data</code> functions
<code>is_missing</code>	Boolean matrix indicating whether the observation should be treated as missing data and ignored. Number of rows equals the number of loci, number of columns equals the number samples. Alternatively, the user may pass in <code>FALSE</code> if no data should be considered missing.
<code>allow_relatedness</code>	Bool indicating whether or not to allow relatedness within host
<code>thin</code>	Positive Integer. How often to sample from mcmc, 1 means do not thin
<code>burnin</code>	Positive Integer. Number of MCMC samples to discard as burnin
<code>samples_per_chain</code>	Positive Integer. Number of samples to take after burnin
<code>verbose</code>	Logical indicating if progress is printed
<code>use_message</code>	Logical indicating if progress is printed using message or print
<code>eps_pos_alpha</code>	Positive Numeric. Alpha parameter in Beta distribution for <code>eps_pos</code> prior
<code>eps_pos_beta</code>	Positive Numeric. Beta parameter in Beta distribution for <code>eps_pos</code> prior
<code>eps_neg_alpha</code>	Positive Numeric. Alpha parameter in Beta distribution for <code>eps_neg</code> prior
<code>eps_neg_beta</code>	Positive Numeric. Beta parameter in Beta distribution for <code>eps_neg</code> prior
<code>r_alpha</code>	Positive Numeric. Alpha parameter in Beta distribution for relatedness prior
<code>r_beta</code>	Positive Numeric. Beta parameter in Beta distribution for relatedness prior
<code>mean_coi_shape</code>	shape parameter for gamma hyperprior on mean COI
<code>mean_coi_scale</code>	scale parameter for gamma hyperprior on mean COI
<code>max_eps_pos</code>	Numeric. Maximum allowed value for <code>eps_pos</code>
<code>max_eps_neg</code>	Numeric. Maximum allowed value for <code>eps_neg</code>
<code>max_coi</code>	Positive Numeric. Maximum allowed complexity of infection
<code>num_chains</code>	Total number of chains to run, possibly simultaneously

num_cores	Total OMP parallel threads to use to run chains. $\text{num_cores} * \text{pt_num_threads}$ should not exceed the number of cores available on your system.
pt_chains	Total number of chains to run with parallel tempering or a vector containing the temperatures that should be used for parallel tempering.
pt_grad	Power to raise parallel tempering chains to. A value of 1 results in evenly distributed temperatures between [0,1], below 1 will bias towards 1 and above 1 will bias towards 0. Only used if pt_chains is a single value (i.e. not a vector).
pt_num_threads	Total number of OMP parallel threads to be used to process parallel tempered chains $\text{num_cores} * \text{pt_num_threads}$ should not exceed the number of cores available on your system.
adapt_temp	Logical indicating whether or not to adapt the parallel tempering temperatures. If TRUE, the temperatures will be adapted during the burnin period, starting after pre_adapt_steps steps. The adaptation will occur every temp_adapt_steps steps until burnin is complete. The range of temperatures will remain the same as specified by pt_chains.
pre_adapt_steps	Number of steps to take before starting to adapt the parallel tempering temperatures. Only used if adapt_temp is TRUE.
temp_adapt_steps	Number of steps to take between temperature adaptation steps. Only used if adapt_temp is TRUE.
max_initialization_tries	Number of times to try to initialize the chain before giving up

simulated_data	<i>Simulated genotyping data</i>
----------------	----------------------------------

Description

A simulated dataset created using `simulate_data()`

Usage

```
simulated_data
```

Format

An object of class `list` of length 9.

simulate_allele_frequencies
Simulate allele frequencies

Description

Simulate allele frequencies

Usage

```
simulate_allele_frequencies(alpha, num_loci)
```

Arguments

alpha	vector parameter controlling the Dirichlet distribution
num_loci	total number of loci to draw

Details

Simulate allele frequency vectors as a draw from a Dirichlet distribution

simulate_data *Simulate data generated according to the assumed model*

Description

Simulate data generated according to the assumed model

Usage

```
simulate_data(  
  mean_coi = NULL,  
  num_samples,  
  epsilon_pos,  
  epsilon_neg,  
  sample_cois = NULL,  
  locus_freq_alphas = NULL,  
  allele_freqs = NULL,  
  internal_relatedness_alpha = 0,  
  internal_relatedness_beta = 1,  
  internal_relatedness = NULL,  
  missingness = 0  
)
```

Arguments

mean_coi	Mean multiplicity of infection drawn from a Poisson
num_samples	Total number of biological samples to simulate
epsilon_pos	False positive rate, expected number of false positives
epsilon_neg	False negative rate, expected number of false negatives
sample_cois	List of sample COIs to be used instead of simulating
locus_freq_alphas	List of alpha vectors to be used to simulate from a Dirichlet distribution to generate allele frequencies.
allele_freqs	List of allele frequencies to be used instead of simulating allele frequencies
internal_relatedness_alpha	alpha parameter of beta distribution controlling the random relatedness draws for each sample
internal_relatedness_beta	beta parameter of beta distribution controlling the random relatedness draws for each sample
internal_relatedness	List of internal relatedness values to be used instead of simulating
missingness	probability of data being missing

Value

Simulated data that is structured to go into the MCMC sampler

simulate_observed_allele
Simulates the observation process

Description

Simulates the observation process

Usage

```
simulate_observed_allele(alleles, epsilon_pos, epsilon_neg, missingness)
```

Arguments

alleles	A numeric vector representing the number of strains contributing each allele
epsilon_pos	expected number of false negatives
epsilon_neg	expected number of false positives
missingness	probability that the data is missing

Details

Takes a numeric value representing the number of strains contributing an allele and returns a binary vector indicating the presence or absence of the allele.

simulate_observed_genotype
Simulate observed genotypes

Description

Simulate observed genotypes

Usage

```
simulate_observed_genotype(  
  true_genotypes,  
  epsilon_pos,  
  epsilon_neg,  
  missingness  
)
```

Arguments

true_genotypes a list of numeric vectors that are input to sim_observed_allele
epsilon_pos expected number of false positives
epsilon_neg expected number of false negatives
missingness probability of data being missing

Details

Simulate the observation process across a list of observation vectors

simulate_sample_coi *Simulate sample COI*

Description

Simulate sample COI

Usage

```
simulate_sample_coi(num_samples, mean_coi)
```

Arguments

num_samples the total number of biological samples to simulate
mean_coi mean multiplicity of infection

Details

Simulate sample COIs from a zero-truncated Poisson distribution

simulate_sample_genotype
Simulate sample genotype

Description

Simulate sample genotype

Usage

```
simulate_sample_genotype(sample_cois, locus_allele_dist, internal_relatedness)
```

Arguments

sample_cois Numeric vector indicating the multiplicity of infection for each biological sample
locus_allele_dist Allele frequencies – simplex parameter of a multinomial distribution
internal_relatedness numeric 0-1 indicating the probability for a strain's allele to come from an existing lineage within host

Details

Simulates sampling the genetics at a single locus given an allele frequency distribution and a vector of sample COIs

`summarize_allele_freqs`*Summarize allele frequencies*

Description

Summarize allele frequencies

Usage

```
summarize_allele_freqs(  
  mcmc_results,  
  lower_quantile = 0.025,  
  upper_quantile = 0.975,  
  merge_chains = TRUE  
)
```

Arguments

`mcmc_results` Result of calling `run_mcmc()`
`lower_quantile` The lower quantile of the posterior distribution to return
`upper_quantile` The upper quantile of the posterior distribution to return
`merge_chains` boolean indicating that all chain results should be merged

Details

Summarize individual allele frequencies from the posterior distribution of sampled allele frequencies

`summarize_allele_freq_fn`*Summarize Function of Allele Frequencies*

Description

Summarize Function of Allele Frequencies

Usage

```
summarize_allele_freq_fn(  
  mcmc_results,  
  fn,  
  lower_quantile = 0.025,  
  upper_quantile = 0.975,  
  merge_chains = TRUE  
)
```

Arguments

mcmc_results	Result of calling run_mcmc()
fn	Function that takes as input a simplex to apply to each allele frequency vector
lower_quantile	The lower quantile of the posterior distribution to return
upper_quantile	The upper quantile of the posterior distribution to return
merge_chains	boolean indicating that all chain results should be merged

Details

General function to summarize the posterior distribution of functions of the sampled allele frequencies

summarize_coi	<i>Summarize COI</i>
---------------	----------------------

Description

Summarize COI

Usage

```
summarize_coi(
  mcmc_results,
  lower_quantile = 0.025,
  upper_quantile = 0.975,
  naive_offset = 2,
  merge_chains = TRUE
)
```

Arguments

mcmc_results	Result of calling run_mcmc
lower_quantile	The lower quantile of the posterior distribution to return
upper_quantile	The upper quantile of the posterior distribution to return
naive_offset	Offset used in calculate_naive_coi_offset
merge_chains	boolean indicating that all chain results should be merged

Details

Summarize complexity of infection results from MCMC. Returns a dataframe that contains summaries of the posterior distribution of COI for each biological sample, as well as naive estimates of COI.

summarize_effective_coi

Summarize effective COI

Description

Summarize effective COI

Usage

```
summarize_effective_coi(  
  mcmc_results,  
  lower_quantile = 0.025,  
  upper_quantile = 0.975,  
  merge_chains = TRUE  
)
```

Arguments

`mcmc_results` Result of calling `run_mcmc()`
`lower_quantile` The lower quantile of the posterior distribution to return
`upper_quantile` The upper quantile of the posterior distribution to return
`merge_chains` boolean indicating that all chain results should be merged

Details

Summarize effective COI from MCMC. Returns a dataframe that contains summaries of the posterior distribution of effective COI for each biological sample.

summarize_epsilon_neg *Summarize epsilon_neg*

Description

Summarize epsilon_neg

Usage

```
summarize_epsilon_neg(  
  mcmc_results,  
  lower_quantile = 0.025,  
  upper_quantile = 0.975,  
  merge_chains = TRUE  
)
```


Arguments

mcmc_results Result of calling run_mcmc()
lower_quantile The lower quantile of the posterior distribution to return
upper_quantile The upper quantile of the posterior distribution to return
merge_chains boolean indicating that all chain results should be merged

Details

Summarize epsilon negative results from MCMC. Returns a dataframe that contains summaries of the posterior distribution of epsilon negative for each biological sample.

summarize_epsilon_pos *Summarize epsilon_pos*

Description

Summarize epsilon_pos

Usage

```
summarize_epsilon_pos(  
  mcmc_results,  
  lower_quantile = 0.025,  
  upper_quantile = 0.975,  
  merge_chains = TRUE  
)
```

Arguments

mcmc_results Result of calling run_mcmc()
lower_quantile The lower quantile of the posterior distribution to return
upper_quantile The upper quantile of the posterior distribution to return
merge_chains boolean indicating that all chain results should be merged

Details

Summarize epsilon positive results from MCMC. Returns a dataframe that contains summaries of the posterior distribution of epsilon positive for each biological sample.

summarize_he	<i>Summarize locus heterozygosity</i>
--------------	---------------------------------------

Description

Summarize locus heterozygosity

Usage

```
summarize_he(  
  mcmc_results,  
  lower_quantile = 0.025,  
  upper_quantile = 0.975,  
  merge_chains = TRUE  
)
```

Arguments

`mcmc_results` Result of calling `run_mcmc()`
`lower_quantile` The lower quantile of the posterior distribution to return
`upper_quantile` The upper quantile of the posterior distribution to return
`merge_chains` Merge the results of multiple chains into a single summary

Details

Summarize locus heterozygosity from the posterior distribution of sampled allele frequencies.

summarize_relatedness	<i>Summarize relatedness</i>
-----------------------	------------------------------

Description

Summarize relatedness

Usage

```
summarize_relatedness(  
  mcmc_results,  
  lower_quantile = 0.025,  
  upper_quantile = 0.975,  
  merge_chains = TRUE  
)
```

Arguments

- `mcmc_results` Result of calling `run_mcmc()`
- `lower_quantile` The lower quantile of the posterior distribution to return
- `upper_quantile` The upper quantile of the posterior distribution to return
- `merge_chains` boolean indicating that all chain results should be merged

Details

Summarize relatedness results from MCMC. Returns a dataframe that contains summaries of the posterior distribution of relatedness for each biological sample.

Index

* datasets

mcmc_results, 6
simulated_data, 9

calculate_he, 2
calculate_med_allele_freqs, 3
calculate_naive_allele_frequencies, 3
calculate_naive_coi, 4
calculate_naive_coi_offset, 4

load_delimited_data, 5
load_long_form_data, 5

mcmc_results, 6

plot_chain_swaps, 6

rdirichlet, 7
run_mcmc, 7

simulate_allele_frequencies, 10
simulate_data, 10
simulate_observed_allele, 11
simulate_observed_genotype, 12
simulate_sample_coi, 12
simulate_sample_genotype, 13
simulated_data, 9
summarize_allele_freq_fn, 14
summarize_allele_freqs, 14
summarize_coi, 15
summarize_effective_coi, 16
summarize_epsilon_neg, 16
summarize_epsilon_pos, 17
summarize_he, 18
summarize_relatedness, 18