Package ‘negenes’

October 13, 2022

Version 1.0-12
Date 2019-08-05
Title Estimating the Number of Essential Genes in a Genome
Author Karl W Broman [aut, cre] (<https://orcid.org/0000-0002-4914-6671>)
Maintainer Karl W Broman <broman@wisc.edu>
Description Estimating the number of essential genes in a genome on
the basis of data from a random transposon mutagenesis experiment,
through the use of a Gibbs sampler.
Depends R (>= 2.10.1)
Imports stats
Suggests roxygen2
License GPL (>= 3)
URL https://github.com/kbroman/negenes
BugReports https://github.com/kbroman/negenes/issues
Encoding UTF-8
RoxygenNote 6.1.1
NeedsCompilation yes
Repository CRAN
Date/Publication 2019-08-05 21:50:09 UTC

R topics documented:

Mt80 .......................... 2
negenes .......................... 3
sim.mutants .......................... 5

Index 7
Number of insertion sites in each gene in *Mtb* CDC1551

### Description

Number of insertion sites in the initial 80% of each gene in the *Mycobacterium tuberculosis* CDC1551 genome.

### Format

A matrix with two columns. Each row corresponds to a gene. (The row names are the MT numbers of the genes.) The element in the first column is the number of transposon insertion sites in the initial 80% that appear in the corresponding gene and in no other gene. The element in the second column is the number of transposon insertion sites in the initial 80% of both that gene and the following gene. There are 4204 rows; the 46 genes with no such site are not included.

### Source

http://www.tigr.org

### References


### See Also

`negenes::negenes()`, `negenes::sim.mutants()`

### Examples

```r
# Not run: data(Mtb80)

# simulate 44% of genes to be essential
essential <- rep(0,nrow(Mtb80))
essential[sample(1:nrow(Mtb80), ceiling(nrow(Mtb80)*0.44))] <- 1

# simulate 759 mutants
counts <- sim.mutants(Mtb80[,1], essential, Mtb80[,2], 759)

# run the Gibbs sampler
output <- negenes(Mtb80[,1], counts[,1], Mtb80[,2], counts[,2])
# End(Not run)
```
**negenes**

*Estimate the number of essential genes in a genome*

**Description**

Estimate, via a Gibbs sampler, the posterior distribution of the number of essential genes in a genome with data from a random transposon mutagenesis experiment. (See the technical report cited below.)

**Usage**

```
negenes(n.sites, counts, n.sites2 = NULL, counts2 = NULL, 
n.mcmc = 5000, skip = 49, burnin = 500, startp = 1, 
trace = TRUE, calc.prob = FALSE, return.output = FALSE)
```

**Arguments**

- `n.sites`: A vector specifying the number of transposon insertion sites in each gene (alone). All elements must be strictly positive.
- `counts`: A vector specifying the number of mutants observed for each gene (alone). Must be the same length as `n.sites`, and all elements must be non-negative integers.
- `n.sites2`: A vector specifying the number of transposon insertion sites shared by adjacent genes. The `i`th element is the number of insertion sites shared by genes `i` and `i+1`. The last element is for sites shared by genes `N` and 1. If NULL, assume all are 0.
- `counts2`: A vector specifying the number of mutants shared by adjacent gene (analogous to `n.sites2`). The `i`th element is the number of mutants at sites shared by genes `i` and `i+1`. The last element is for sites shared by genes `N` and 1. If NULL, assume all are 0.
- `n.mcmc`: Number of Gibbs steps to perform.
- `skip`: An integer; only save every `skip` + 1st step.
- `burnin`: Number of initial Gibbs steps to run (output discarded).
- `startp`: Initial proportion of genes for which no mutant was observed that will be assumed essential for the Gibbs sampler. (Genes for which a mutant was observed are assumed non-essential; other genes are assumed essential independent with this probability.)
- `trace`: If TRUE, print iteration number occasionally.
- `calc.prob`: If TRUE, return the log posterior probability (up to an additive constant) for each saved iteration.
- `return.output`: If TRUE, include detailed Gibbs results in the output.
Value

A list with components `n.essential` (containing the total number of essential genes at each iteration of the Gibbs sampler) `summary` (a vector containing the estimated mean, SD, 2.5 percentile and 97.5 percentile of the posterior distribution of the number of essential genes).

The next component, `geneprob`, is a vector with one element for each gene, containing the estimated posterior probability that each gene is essential. These are Rao-Blackwellized estimates.

If the argument `calc.prob` was true, there will also be a component `logprob` containing the log (base e) of the posterior probability (up to an additive constant) at each Gibbs step.

If the argument `return.output` was true, there will also be a matrix with `n.mcmc` / (`skip` + 1) rows (corresponding to the Gibbs steps) and a column for each gene. The entries in the matrix are either 0 (essential gene) or 1 (non-essential gene) according to the state of that gene at that step in the Gibbs sampler.

Author(s)

Karl W Broman, <broman@wisc.edu>

References


See Also

`negenes::sim.mutants()`, `negenes::Mtb80()`

Examples

data(Mtb80)

# simulate 44% of genes to be essential
essential <- rep(0,nrow(Mtb80))
essential[sample(1:nrow(Mtb80),ceiling(nrow(Mtb80)*0.44))] <- 1

# simulate 759 mutants
counts <- sim.mutants(Mtb80[,1], essential, Mtb80[,2], 759)

# run the Gibbs sampler without returning detailed output
## Not run: output <- negenes(Mtb80[,1], counts[,1], Mtb80[,2], counts[,2])

# run the Gibbs sampler, returning the detailed output
## Not run: output2 <- negenes(Mtb80[,1], counts[,1], Mtb80[,2], counts[,2], return=TRUE)
**Description**

Simulate data for a random transposon mutagenesis experiment.

**Usage**

```
sim.mutants(n.sites, essential, n.sites2 = NULL, n.mutants)
```

**Arguments**

- `n.sites`: A vector specifying the number of transposon insertion sites in each gene. All elements must be strictly positive.
- `essential`: A vector containing 1’s (indicating that the corresponding gene is essential) and 0’s (indicating that the corresponding gene is not essential). Must be the same length as `n.sites`.
- `n.sites2`: A vector specifying the number of transposon insertion sites shared by adjacent genes. The $i$th element is the number of insertion sites shared by genes $i$ and $i+1$. The last element is for sites shared by genes $N$ and 1. If missing, these are assumed to be all 0.
- `n.mutants`: Number of mutants to simulate.

**Value**

If `n.sites2` is missing or contains all 0’s, a vector is returned containing the number of mutants observed for each gene.

If `n.sites2` is not missing and has some positive entries, a matrix with two columns is returned. The first column contains the number of mutants observed for each gene alone; the second column contains the number of mutants observed shared by adjacent genes.

**Author(s)**

Karl W Broman, <broman@wisc.edu>

**References**


**See Also**

negenes::nogenes(), nogenes::Mt80()
Examples

## Not run: data(Mtb80)

# simulate 44% of genes to be essential
essential <- rep(0,nrow(Mtb80))
essential[sample(1:nrow(Mtb80),ceiling(nrow(Mtb80)*0.44))] <- 1

# simulate 759 mutants
counts <- sim.mutants(Mtb80[,1], essential, Mtb80[,2], 759)

# run the Gibbs sampler
output <- negenes(Mtb80[,1], counts[,1], Mtb80[,2], counts[,2])
## End(Not run)
Index

* datagen
  sim.mutants, 5

* datasets
  Mtb80, 2

* models
  negenes, 3

Mtb80, 2

negenes, 3
negenes::Mtb80(), 4, 5
negenes::negenes(), 2, 5
negenes::sim.mutants(), 2, 4

sim.mutants, 5