Package `rforensicbatwing`

February 20, 2015

Type Package

Title BATWING for calculating forensic trace-suspect match probabilities

Version 1.3

Date 2014-06-27

Author Mikkel Meyer Andersen and Ian J. Wilson

Maintainer Mikkel Meyer Andersen <mikl@math.aau.dk>

Description A modified version (with great help from Ian J. Wilson) of Ian J. Wilson's program BATWING for calculating forensic trace-suspect match probabilities.

License GPL

LinkingTo Rcpp

Imports Rcpp

NeedsCompilation yes

Repository CRAN

Date/Publication 2014-06-27 13:56:27

R topics documented:

rforensicbatwing-package ........................................................................ 1
Inference from STR data using coalescent theory ...................................... 2

Index 6

rforensicbatwing-package

`BATWING for calculating forensic trace-suspect match probabilities`

Description

A modified version (with great help from Ian J. Wilson) of Ian J. Wilson’s program BATWING for calculating forensic trace-suspect match probabilities.
Details

Please see the documentation of `coalmatchprob` for an usage example.

Author(s)

Mikkel Meyer Andersen and Ian J. Wilson
Maintainer: Mikkel Meyer Andersen <mikl@math.aau.dk>

References

The BATWING manual

See Also

`batwing` `coalmatchprob`

Description

Estimates the trace-suspect match probability for a lineage haplotype of STR markers using coalescent theory.

Usage

```r
coalmatchprob(database, haplotype, 
  reps = 10, burnin = 0, treebetN = 10, Nbetsamp = 10, 
  muprior = "constant(0.003)", Nprior = "lognormal(9, 1)", alphaprior = NULL, 
  progress = TRUE, trace = FALSE)

batwing(database, 
  reps = 10, burnin = 0, treebetN = 10, Nbetsamp = 10, 
  muprior = "constant(0.003)", Nprior = "lognormal(9, 1)", alphaprior = NULL, 
  progress = TRUE, trace = FALSE)
## S3 method for class 'batwing'
print(x, ...) 
## S3 method for class 'forensicbatwing'
plot(x, ...)
```
Arguments

- **database**: Reference STR database.
- **haplotype**: Haplotype of the suspect.
- **reps**: Number of output lines.
- **burnin**: Number of reps to take before starting recording data.
- **treebetN**: The number of times that changes to the genealogical tree are attempted before any changes to the hyperparameters are attempted. Thus BATWING outputs are separated by treebetN * Nbetsamp attempted tree updates.
- **Nbetsamp**: The number of times that changes to hyperparameters are attempted between outputs.
- **muprior**: Either a single prior distribution for the mutation rate or a vector of prior distributions (one for each locus). If only one prior is supplied, the same mutation rate is used for all loci. If one prior per locus is supplied, each locus has its own chain of mutation rates.
- **nprior**: Prior distribution of the effective population size.
- **alphaprior**: If NULL, there is no growth (constant population size). If a prior distribution is specified, this gives exponential growth at rate alpha at all times.
- **progress**: Whether to print progress or not.
- **trace**: Whether to print extra trace information or not.
- **x**: A batwing or forensicbatwing object.
- ...: Not used

Details

Note that the batwing function runs a standard coalescent inference as described in I.J. Wilson (1999, 2003).

Note that, in contrast to the original BATWING program, migration is not supported. Neither is BATWING’s sizemodel=2 (constant-sized population up to a time from where there is exponential growth).

Valid prior distributions:

- **uniform(v1, v2)** uniform on the interval \((v1, v2)\).
- **constant(v1)** constant value \(v1\).
- **normal(v1, v2)** Normal distribution with mean = \(v1\) and sd = \(v2\).
- **lognormal(v1, v2)** If \(X\) has this distribution then \(\log(X)\) has the normal\((v1, v2)\) distribution.
- **gamma(v1, v2)** Gamma distribution with shape \(v1\) and rate \(v2\) giving mode = \((v1-1)/v2\) and mean = \(v1/v2\).
- **beta(v1, v2)** Beta distribution with shape parameters \(v1\) and \(v2\) giving mean = \(v1/(v1 + v2)\) and the variance is \((v1*v2)/((v1 + v2)^2 * (v1 + v2 + 1))\)
Inference from STR data using coalescent theory

Value

coalmatchprob An object of type forensicbatwing and batwing (for now, forensicbatwing just means that there is a p column in the result matrix).
batwing An object of type batwing
batwing-object parameters The parameters used
priors The priors used
result Matrix with information about the iterations (forensic match probability (if coalmatchprob), current value of \( N \), \( \mu \)'s, tree height \( T \), total branch length \( L \), population growth \( \alpha \), and loglikelihoods for times, mutations and priors).
proposals_tree The number of new tree proposals.
accepted_tree The number of new trees accepted.
proposals_hyperparameters The number of new hyperparameters proposals.
accepted_hyperparameters The number of new hyperparameters accepted.

Author(s)

Mikkel Meyer Andersen and Ian Wilson

Source

BATWING at Ian Wilson’s homepage

References

The BATWING manual


Examples

```r
## Not run:
database <- matrix(c(0L, 0L, 1L, 1L, 0L), ncol = 2, byrow = TRUE)
haplotype <- c(0L, 0L)

c # coalmatchprob:
coalmp <- coalmatchprob(database, haplotype,
  reps = 1000L, burnin = 0L, treebetn = 10L, Nbetsamp = 10L,
  muprior = c("normal(0.003, 0.001)", "normal(0.005, 0.001)"),
  Nprior = "lognormal(9, 1)",
  alphaprior = NULL,
  progress = TRUE, trace = FALSE)
```

```c
```
Inference from STR data using coalescent theory

coalmp

murange <- range(c(coalmp$result$mu1, coalmp$result$mu2))

par(mfrow = c(2, 2))
plot(coalmp)
plot(coalmp$result$N, type = "l", ylab = "N")

plot(coalmp$result$mu1, type = "l", col = "red", ylim = murange, ylab = "mu")
points(coalmp$result$mu2, type = "l", col = "blue")

hist(coalmp$result$mu1, col = "FF000066",
     xlim = murange, ylim = c(0, 250), main = NULL, xlab = "mu")
hist(coalmp$result$mu2, add = TRUE, col = "#0000FF66")
par(mfrow = c(1, 1))

# batwing:
bw <- batwing(database,
               reps = 10000, burnin = 1000, treebetN = 10, Nbetsamp = 10,
               muprior = "normal(0.003, 0.001)",
               Nprior = "lognormal(9, 1)",
               alphaprior = NULL,
               progress = TRUE, trace = FALSE)
bw

## End(Not run)
Index

*Topic **coalescence**
  Inference from STR data using coalescent theory, 2

*Topic **coalescent**
  Inference from STR data using coalescent theory, 2

*Topic **package**
  rforensicbatwing-package, 1
  batwing, 2
  batwing(Inference from STR data using coalescent theory), 2
  coalmatchprob, 2
  coalmatchprob(Inference from STR data using coalescent theory), 2
  Inference from STR data using coalescent theory, 2
  plot.forensicbatwing(Inference from STR data using coalescent theory), 2
  print.batwing(Inference from STR data using coalescent theory), 2
  rforensicbatwing
    (rforensicbatwing-package), 1
  rforensicbatwing-package, 1