Package 'WGDgc'

August 29, 2013

Type Package

Title Detection of whole genome duplications on phylogenies using gene count data, with estimation of background rates of gene duplication and loss, and estimation of gene retention rates following whole genome duplications

Version 0.9

Date 2013-08-29

Author Tram Ta, Charles-Elie Rabier

Maintainer Tram Ta <tramta@stat.wisc.edu>

Description Detection of whole genome duplication, and estimation of birth rate, death rate, retention rate using gene count method

License GPL (>= 2)

Depends R (>= 3.0.1), phylobase, phyext, ape

R topics documented:

`getMatAndMatDoomed` *Message-passing algorithm*

Description

Uses gene count data to compute the probability of the data below each node of the species tree. Also computes the probability of not observing any data below each node of the species tree

Usage

```
getMatAndMatDoomed(logLamlogMu, nLeaf, nFamily,
                   phyloMat, geneCountData, nPos,
                   wgdTab)
```

Arguments

`logLamlogMu`	vector of size 1 (resp. 2) when the duplication rate and the loss rate are (resp. not) equal. The first component refers to the log duplication rate whereas the second component (if appropriate) refers to the log loss rate.
`nLeaf`	number of present-day species.
`nFamily`	number of gene families.
`phyloMat`	a phylogenetic matrix with 4 columns: parent (ancestor node), child (descendant node), time (branch length), and species names. The number of rows is the number of nodes in the tree.
`geneCountData`	data frame with one column for each species and one row for each family, containing the number of gene copies in each species for each gene family. The column names must match the species names in the tree.
`nPos`	maximum number of possible values for the number of gene copies at each internal node of the phylogeny.
`wgdTab`	a WGD table with 3 columns: node before WGD, retention rate and loss rate. The number of rows is the number of WGD events.

Value

`Mat`	matrix of size nPos x (nNode-nLeaf) x nFamily where nNode is the number of nodes in the species tree. Column j corresponds to the internal node number j+nLeaf. Each entry Mat[i,j,k] is the probability of the data below node (j+nLeaf) in family k given that the node started with i genes
`MatDoomed`	matrix of size nNode x 3. Each row refers to one node. Each column corresponds to the probability to be doomed when a lineage starts at the corresponding node. The first column is the probability that the lineage goes extinct inside the species tree. The second (resp. third) column is the probability that the lineage goes extinct in the left-side (resp. right-side) of the species tree.

`logLikGeneCount` *Minus log-likelihood function*

Description

Evaluates the minus log-likelihood (based on gene counts) at given values of the parameters

Usage

```
logLikGeneCount(para, input, geneCountData, nPos=NULL,
                geomMean=NULL, dirac=NULL, useRootStateMLE=F,
                conditioning=c("oneOrMore", "twoOrMore",
                "oneInBothClades", "none"),
                equalBDrates=F, fixedRetentionRates=T)
```

Arguments

`para`	vector of parameters (see Details)
`input`	same type of object as output of function processInput
`geneCountData`	data frame with one column for each species and one row for each family, containing the number of gene copies in each species for each gene family. The column names must match the species names in the tree.
`nPos`	maximum number of possible values for the number of gene copies at each internal node of the phylogeny.
`geomMean`	the mean of the prior geometric distribution for the number of genes at the root.
`dirac`	value for the number of genes at the root, when this is assumed to have a fixed value (according to a dirac prior distribution).
`useRootStateMLE`	if TRUE, the most likely number of genes at the root is determined for each family separately (given the parameter values used), and is used to evaluate the likelihood function.
`conditioning`	type of conditioning for the likelihood calculation. The default is to calculate conditional probabilities on observing families with at least 1 gene copy (see Details of function MLEGeneCount).
`equalBDrates`	if TRUE, the duplication and loss rates are equal.
`fixedRetentionRates`	if TRUE, it uses retention rates present in parameter 'input$wgdTab'. If FALSE, it uses retention rates located in parameter 'para'.

Details

The vector 'para' is defined in the following way.

When 'equalBDrates=TRUE' and 'fixedRetentionRates=TRUE', the vector 'para' is equal to the log birth rate.

When 'equalBDrates=FALSE' and 'fixedRetentionRates=TRUE', 'para' is a vector of size 2 : the first component is the log duplication rate whereas the second component is the log loss rate.

When 'equalBDrates=TRUE' and 'fixedRetentionRates=FALSE', 'para' is a vector of size 'number of WGDs + 1' : the first component is the log duplication rate, other components are retention rates at the different WGDs.

When 'equalBDrates=FALSE' and 'fixedRetentionRates=FALSE', 'para' is a vector of size 'number of WGDs + 2' : the first two components are the log duplication rate and the log loss rate, other components are retention rates at the different WGDs.

`MLEGeneCount` *Maximum likelihood estimation based gene count method*

Description

Uses gene count data to estimates rates of gene duplication and loss along a phylogeny with zero, one or more whole genome duplication (WGD) events. Also estimates the gene retention rate after each WGD event.

Usage

```
MLEGeneCount(tr, geneCountData, nPos=NULL, geomMean=NULL,
             dirac=NULL, useRootStateMLE=F,
             conditioning=c("oneOrMore", "twoOrMore",
             "oneInBothClades", "none"),
             equalBDrates=F, fixedRetentionRates=F,
             startingValue=c(0.01, 0.02))
```

Arguments

`tr`	a species tree in SIMMAP format (see Details).
`geneCountData`	data frame with one column for each species and one row for each family, containing the number of gene copies in each species for each gene family. The column names must match the species names in the tree.
`nPos`	maximum number of possible values for the number of gene copies at each internal node of the phylogeny.
`geomMean`	the mean of the prior geometric distribution for the number of genes at the root.
`dirac`	value for the number of genes at the root, when this is assumed to have a fixed value (according to a dirac prior distribution).
`useRootStateMLE`	if TRUE, the most likely number of genes at the root is determined for each family separately, and is used to calculate the overall likelihood of the data. This value at the root may vary with the parameter values during likelihood optimization.
`conditioning`	type of conditioning for the likelihood calculation. The default is to calculate conditional probabilities on observing families with at least 1 gene copy (see Details).
`equalBDrates`	if TRUE, the duplication and loss rates are constrained to be equal.

`fixedRetentionRates`
: if TRUE, retention rates from the user-defined tree are fixed and used as provided. If FALSE, retention rates are considered as parameters and are estimated by maximum likelihood.

`startingValue`
: Vector of starting values for respectively birth rate and death rate. The size of this vector is always 2. When option equalBDrates is TRUE, the software uses only the the first component of the vector (i.e. startingValue[1]).

Details

The tree needs to be in simmap format (version 1.1). This format is similar to the newick parenthetical format, except that branch lengths are given inside brackets where states are indicated at specific times along each branch. Along a given branch, the token "0,18" indicates state 0 for a duration of 18 time units. Tokens are separated with ":". State 0 is used to indicate branch segments where only the birth/death process applies for gene duplications and losses. Positive states are used for branch segments at WGD events, where the state value indicates the retention rate after the WGD. Such WGD segments need to have a length of 0.

Four types of conditional likelihoods are implemented. The option 'conditioning' should match the data filtering process: use conditioning="oneOrMore" if all families with one or more gene copies are included in the data, use "twoOrMore" to condition on families having two of more genes, "oneInBothClades" if the data set was filtered to include only families with at least one gene copy in each of the two main clades stemming from the root. conditioning="none" uses unconditional likelihoods.

The "geomMean", "dirac" and "useRootStateMLE" options are incompatible.

Value

`birthrate` birth rate

`deathrate` death rate

`loglikelihood`
: log of the likelihood

`WGDtable` a WGD table with 3 columns: node before WGD, retention rate and loss rate. The number of rows is the number of WGD events

`phyloMat` a phylogenetic matrix with 4 columns: parent (ancestor node), child (descendant node), time (branch length), and species names. The number of rows is the number of nodes in the tree

Author(s)

Tram Ta, Charles-Elie Rabier, Cecile Ane

References

Bailey, N. (1964) The Elements of Stochastic Processes. New York: John Wiley \& Sons

Bollback J. P. (2006) SIMMAP: Stochastic character mapping of discrete traits on phylogenies. *Bioinformatics*. **7**: 88

De Bie, T. and Cristianini, N. and Demuth, J.P. and Hahn, M.W. (2006) CAFE: a computational tool for the study of gene family evolution. *Bioinformatics*. **22**: 1269–1271

Hahn, M.W. and De Bie, T. and Stajich, J.E. and Nguyen, C. and Cristianini, N. (2005) Estimating the tempo and mode of gene family evolution from comparative genomic data. *Genome Res.*. **15**: 1153–1160

Crawford, F., Suchard, M. (2012) Transition probabilities for general birth-death processes with applications in ecology, genetics, and evolution. *J Math Biol*. **65**: 553-580

Rabier, C., Ta, T., Ane, C. (2013) Detecting and Locating Whole Genome Duplications on a phylogeny: a probabilistic approach. Submitted

Examples

```
## Not run:
tre.string = "(D:{0,18.03},(C:{0,12.06},(B:{0,7.06},
              A:{0,7.06}):{0,2.49:0.9,0:0,2.50}):{0, 5.97});"

# tree with a single hypothesized WGD event, along the
# internal edge leading to the MRCA of species A and B
# with hypothesized retention rate 0.9.

tre.phylo4d = read.simmap(text=tre.string)
tre.phylo   = as(tre.phylo4d, "phylo")
plot(collapse.singles(tre.phylo))
dat = data.frame(A=c(2,2,3,1), B=c(3,0,2,1), C=c(1,0,2,2), D=c(2,1,1,1))
MLEGeneCount(tre.phylo4d, dat, nPos=10,
             geomMean=1.5, conditioning="oneOrMore",
             fixedRetentionRates=TRUE)


## End(Not run)
```

---

processInput *Preprocessing function*

Description

Checking arguments and preparing data for future optimization

Usage

```
processInput(tr, geomMean=NULL, dirac=NULL, useRootStateMLE=F,
             equalBDrates=F, fixedRetentionRates=T,
             startingValue=c(0.01, 0.02))
```

Arguments

tr	a species tree in SIMMAP format (see Details of function MLEGeneCount).
geomMean	the mean of the prior geometric distribution for the number of genes at the root.
dirac	value for the number of genes at the root, when this is assumed to have a fixed value (according to a dirac prior distribution).
useRootStateMLE	if TRUE, the most likely number of genes at the root will be determined for each family separately during the future optimization.

`equalBDrates` if TRUE, the duplication and loss rates are equal.

`fixedRetentionRates`
: if TRUE, fixed retention rates (obtained from the user-defined tree) will be used during the future optimization. If FALSE, retention rates will be considered as parameters and will be estimated by maximum likelihood.

`startingValue`
: Vector of starting values for respectively duplication and loss rates. The size of this vector is always 2. When option equalBDrates is TRUE, it uses only the the first component of the vector (i.e. startingValue[1]).

Details

Recall that the "geomMean", "dirac" and "useRootStateMLE" options are incompatible.

The vector 'para' is defined in the following way.

When 'equalBDrates=TRUE' and 'fixedRetentionRates=TRUE', it is equal to log(StartingValue[1]).

When 'equalBDrates=FALSE' and 'fixedRetentionRates=TRUE', 'para' is a vector of size 2: the first component is log(StartingValue[1]) whereas the second component is log(StartingValue[2]).

When 'equalBDrates=TRUE' and 'fixedRetentionRates=FALSE', 'para' is a vector of size 'number of WGDs + 1': the first component is log(StartingValue[1]), other components are equal to 0.5 (i.e. 0.5 is chosen as starting value for retention rates at the different WGDs).

When 'equalBDrates=FALSE' and 'fixedRetentionRates=FALSE', 'para' is a vector of size 'number of WGDs + 2': the first two components are log(StartingValue[1]) and log(StartingValue[2]), other components are are equal to 0.5.

'lower' and 'upper' are vectors whose sizes correspond to the number of parameters. 'lower' refers to the lower bounds for the different parameters and 'upper' refers to the upper bounds. The parameters are in the same order as for 'para'.

Value

`phyloMat` Matrix in 'phylo' representation. The number of rows is the number of nodes in the species tree. There are 6 columns (Parent, Child, Time, Species, RetenRate, LossRate)

`nLeaf` Number of present-day species (i.e. number of leaves)

`nNode` Number of nodes in the species tree

`wgdTab` Table of 3 columns. The number of rows corresponds to the number of WGDs. 1st column refers to nodes before WGD. 2nd and 3rd columns are the retention rate and the loss rate

`para` see Details

`lower` see Details

`upper` see Details

`sampleData0WGD` *Simulated gene count data without WGD event*

Description

Simulated gene count data, 4 species (A, B, C, D) and 6000 families.

Usage

```
data(sampleData0WGD)
```

Format

A data frame with 6000 observations on the following 4 variables: A, B, C, D.

Details

Data were generated according to the following species tree (in simmap format version 1.1) :

tree0WGD = "(D:0,18.03, (C:0,12.06,(B:0,7.06,A:0,7.06):0,5.00):0, 5.97);"

The duplication and loss rates are respectively 0.02 and 0.03.

Families with 0 or 1 copy were excluded. Only one ancestral gene is present at the root of the species tree.

Examples

```
## Not run:

data(sampleData0WGD)

tree0WGD = "(D:{0,18.03}, (C:{0,12.06},(B:{0,7.06},A:{0,7.06})
             :{0,5.00}):{0, 5.97});"

tree0WGD = read.simmap(text=tree0WGD)

MLEGeneCount(tree0WGD, sampleData0WGD, nPos=28, dirac=1,
             conditioning="twoOrMore", equalBDrates=FALSE,
             fixedRetentionRates=TRUE)
#in order to estimate retention, duplication and loss rates


sampleData0WGDfiltered<-subset(sampleData0WGD,( ( (sampleData0WGD$A>0) |
                        (sampleData0WGD$B>0) | (sampleData0WGD$C>0) )
                        & (sampleData0WGD$D>0) ))
#filtered data with at least one copy in both clades

MLEGeneCount(tree0WGD, sampleData0WGDfiltered, nPos=28, dirac=1,
             conditioning="oneInBothClades", equalBDrates=FALSE,
             fixedRetentionRates=TRUE)
#uses the appropriate filtering


## End(Not run)
```

`sampleData1WGD` *Simulated gene count data with a single WGD event*

Description

Simulated gene count data with 1 WGD, 4 species (A, B, C, D) and 6000 families.

Usage

```
data(sampleData1WGD)
```

Format

A data frame with 6000 observations on the following 4 variables: A, B, C, D.

Details

Data were generated according to the following species tree (in simmap format version 1.1) :

tree1WGD = "(D:0,18.03, (C:0,12.06,(B:0,7.06,A:0,7.06):0,2.50 :0.6,0:0,2.50):0, 5.97);"

A single WGD event is located along the internal edge leading to the MRCA of species A and B with retention rate 0.6.

The duplication and loss rates are respectively 0.02 and 0.03.

Families with 0 or 1 copy were excluded. Only one ancestral gene is present at the root of the species tree.

Examples

```
## Not run:

data(sampleData1WGD)

tree1WGD = "(D:{0,18.03}, (C:{0,12.06},(B:{0,7.06},A:{0,7.06})
            :{0,2.50 :0.6,0:0,2.50}):{0, 5.97});"
# tree with a single hypothesized WGD event, along the
# internal edge leading to the MRCA of species A and B
# with hypothesized retention rate 0.6.

tree1WGD = read.simmap(text=tree1WGD)

MLEGeneCount(tree1WGD, sampleData1WGD, nPos=31, dirac=1,
             conditioning="twoOrMore", equalBDrates=FALSE,
             fixedRetentionRates=FALSE)
#in order to estimate retention, duplication and loss rates

MLEGeneCount(tree1WGD, sampleData1WGD, nPos=31, dirac=1,
             conditioning="twoOrMore", equalBDrates=FALSE,
             fixedRetentionRates=TRUE)
#in order to estimate only duplication and loss rates


sampleData1WGDfiltered<-subset(sampleData1WGD,( ( (sampleData1WGD$A>0) |
                        (sampleData1WGD$B>0) | (sampleData1WGD$C>0) )
```

```
                          & (sampleData1WGD$D>0) ))
#filtered data with at least one copy in both clades

MLEGeneCount(tree1WGD, sampleData1WGDfiltered, nPos=31, dirac=1,
             conditioning="oneInBothClades", equalBDrates=FALSE,
             fixedRetentionRates=FALSE)
#uses the appropriate filtering


## End(Not run)
```

`sampleData1WGDSameDupLoss` *Simulated gene count data with a single WGD event and same dup/loss rates*

Description

Simulated gene count data with 1 WGD,4 species (A, B, C, D) and 6000 families. The duplication and loss rates are equal.

Usage

```
data(sampleData1WGDSameDupLoss)
```

Format

A data frame with 6000 observations on the following 4 variables: A, B, C, D.

Details

Data were generated according to the following species tree (in simmap format version 1.1) :

tree1WGD = "(D:0,18.03, (C:0,12.06,(B:0,7.06,A:0,7.06):0,2.50 :0.6,0:0,2.50):0, 5.97);"

A single WGD event is located along the internal edge leading to the MRCA of species A and B with retention rate 0.6.

The duplication and loss rates are equal to 0.02.

Families with 0 or 1 copy were excluded. Only one ancestral gene is present at the root of the species tree.

Examples

```
## Not run:

data(sampleData1WGDSameDupLoss)

tree1WGD = "(D:{0,18.03}, (C:{0,12.06},(B:{0,7.06},A:{0,7.06})
            :{0,2.50 :0.6,0:0,2.50}):{0, 5.97});"
# tree with a single hypothesized WGD event, along the
# internal edge leading to the MRCA of species A and B
# with hypothesized retention rate 0.6.
```

```
tree1WGD = read.simmap(text=tree1WGD)

MLEGeneCount(tree1WGD, sampleData1WGDSameDupLoss, nPos=37, dirac=1,
             conditioning="twoOrMore", equalBDrates=TRUE,
             fixedRetentionRates=FALSE)
#in order to estimate retention rate and same dup/loss rate


## End(Not run)
```

`sampleData2WGD` *Simulated gene count data with two WGD events*

Description

Simulated gene count data with 2 WGDs, 4 species (A, B, C, D) and 6000 families.

Usage

```
data(sampleData2WGD)
```

Format

A data frame with 6000 observations on the following 4 variables: A, B, C, D.

Details

Data were generated according to the following species tree (in simmap format version 1.1) :

tree2WGD = "(D:0,18.03, (C:0,12.06,(B:0,7.06,A:0,7.06):0,2.50 :0.5,0:0,2.50):0, 2.985: 0.5,0:0,2.985);"

The oldest WGD event is located along the internal edge leading to the MRCA of species A, B and C. The most recent WGD event is located along the internal edge leading to the MRCA of species A and B. Both retention rates are equal to 0.6.

The duplication and loss rates are respectively 0.02 and 0.03.

Families with 0 or 1 copy were excluded. Only one ancestral gene is present at the root of the species tree.

Examples

```
## Not run:

data(sampleData2WGD)


tree2WGD = "(D:{0,18.03}, (C:{0,12.06},(B:{0,7.06},A:{0,7.06}):
            {0,2.50 :0.6,0:0,2.50}):{0, 2.985: 0.6,0:0,2.985});"
#tree with two WGD events.
#oldest WGD event located along the internal edge leading to the
#MRCA of species A, B and C.
#most recent WGD event located along the internal edge leading to the
#MRCA of species A, B
# hypothesized retention rates : 0.6
```

```
tree2WGD = read.simmap(text=tree2WGD)

MLEGeneCount(tree2WGD, sampleData2WGD, nPos=40, dirac=1,
             conditioning="twoOrMore", equalBDrates=FALSE,
             fixedRetentionRates=FALSE)
#in order to estimate retention, duplication and loss rates

MLEGeneCount(tree2WGD, sampleData2WGD, nPos=40, dirac=1,
             conditioning="twoOrMore", equalBDrates=FALSE,
             fixedRetentionRates=TRUE)
#in order to estimate only duplication and loss rates


sampleData2WGDfiltered<-subset(sampleData2WGD,( ( (sampleData2WGD$A>0) |
                        (sampleData2WGD$B>0) | (sampleData2WGD$C>0) )
                        & (sampleData2WGD$D>0) ))
#filtered data with at least one copy in both clades

MLEGeneCount(tree2WGD, sampleData2WGDfiltered, nPos=40, dirac=1,
             conditioning="oneInBothClades", equalBDrates=FALSE,
             fixedRetentionRates=FALSE)
#uses the appropriate filtering


## End(Not run)
```

`sampleData2WGDSameBranch` *Simulated gene count data with two WGD events located on the same branch*

Description

Simulated gene count data with 2 WGDs on the same branch, 4 species (A, B, C, D) and 6000 families.

Usage

```
data(sampleData2WGDSameBranch)
```

Format

A data frame with 6000 observations on the following 4 variables: A, B, C, D.

Details

Data were generated according to the following species tree (in simmap format version 1.1) : tree2WGD="(D:0,6.01:0.5,0:0,6.01:0.5,0:0,6.01, (C:0,12.06,(B:0,7.06,A:0,7.06):0,4.99):0,5.97);"

Both WGDs are located along the internal edge leading species D. The oldest WGD has retention rate 0.4 whereas the most recent has retention rate 0.8.

The duplication and loss rates are respectively 0.02 and 0.03.

Families with 0 or 1 copy were excluded. Only one ancestral gene is present at the root of the species tree.

Examples

```
## Not run:

data(sampleData2WGDSameBranch)


tree2WGD="(D:{0,6.01:0.5,0:0,6.01:0.5,0:0,6.01}, (C:{0,12.06},(B:{0,7.06},
          A:{0,7.06}):{0,4.99}):{0,5.97});"
#tree with two WGD events
#both WGDs located along the internal edge leading species D
#hypothesized retention rates : 0.5

#first WGD event located along the internal edge leading to the
#MRCA of species A, B and C.

tree2WGD = read.simmap(text=tree2WGD)

MLEGeneCount(tree2WGD, sampleData2WGDSameBranch, nPos=31, dirac=1,
             conditioning="twoOrMore", equalBDrates=FALSE,
             fixedRetentionRates=FALSE)
#in order to estimate retention, duplication and loss rates



## End(Not run)
```

Index