Gene count software

1. How to run the software

Parameters :

- a file "SpeciesTree.txt" a species tree in the Newick format
- a file "geneCountData.txt" which contains gene family sizes
- nPos : number of possibilities for the number of genes at each internal node (default: 101)
- nCondData : 1 for conditioning on observing something, 2 for conditioning on observing ar least two genes, 3 for conditioning on observing at least one gene on each sides of the root
- logLamlogMuWgdLR : a vector of initial values for the log birth rate (i.e. $\log \lambda$), the log death rate (i.e. $\log \mu$), and the loss rate at the WGD (default : $c(\log(0.01), \log(0.02), 0.5)$)

Three options in order to model the number of ancestral genes at the root (choose only one option) :

- isBestStartNum : number of genes at the root treated as an additional parameter (i.e. MLE) (default: False)
- geomMean : mean of the prior geometric distribution (default: Null)
- diracMean : mean of the prior dirac distribution (default: Null)

2. Examples

The file "SpeciesTree.txt" corresponding to the species tree represented in Figure 1 is the following :

```
(D:18.3,(C:12.06,(((B:7.06,A:7.06):2.5):0):2.5):5.97);
```

Note that in order to add a Whole Genome Duplication (WGD), two nodes have been added. Since our model considers that at the WGD the copy is instantaneously duplicated, the distance between these two nodes is equal to zero.

An example of file "geneCountData.txt" which contains the gene family sizes, is the following

```
A B C D
1 2 2 1
1 1 2 1
```

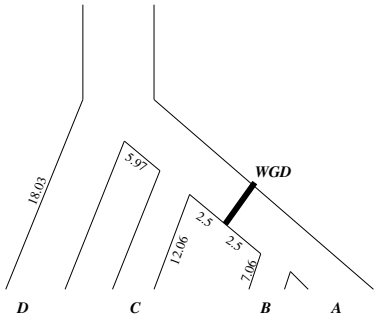

Fig. 1. An example of 4 taxon species tree with one WGD

```
2 1 0 0
2 2 1 1
1 1 1 1
.....
```

Note that each row refers to one gene family. Then, we have to run the following commands :

```
source ("calcLogLikLamMu_P.R")
library(ape)
tree = read.tree("SpeciesTree.txt")
geneCountData = read.table("geneCountData.txt", header=T)
myMLE<-MLEGeneCount(tree, geneCountData, geomMean=2, nCondData=3)
```

myMLE is the object which contains the estimated birth and death rates, the estimated loss rate at the WGD, and the corresponding log likelihood.

```
myMLE

$birthrate
[1] 0.00754341449770223

$deathrate
[1] 0.0571839517933958

$WGDlossrate
[1] 0.41250652499588

$loglikelihood
[1] 2541.59804436398
```

Note that in order to model the number of ancestral genes, we have considered on this example the geometric prior distribution with a mean equal to 2. If we want to use the Dirac distribution with point mass at 1 for instance, use the following command

```
myMLE<-MLEGeneCount(tree, geneCountData, diracMean=1, nCondData=3)
```

If we want to use a MLE approach for the number of ancestral genes :

```
myMLE<-MLEGeneCount(tree, geneCountData, isBestStartNum=T, nCondData=3)
```

In the same way, if we are willing to condition on observing at least two genes

```
myMLE<-MLEGeneCount(tree, geneCountData, isBestStartNum=T, nCondData=2)
myMLE<-MLEGeneCount(tree, geneCountData, diracMean=1, nCondData=2)
myMLE<-MLEGeneCount(tree, geneCountData, geomMean=2, nCondData=2)
```

or to condition on observing something

```
myMLE<-MLEGeneCount(tree, geneCountData, isBestStartNum=T, nCondData=1)
myMLE<-MLEGeneCount(tree, geneCountData, diracMean=1, nCondData=1)
myMLE<-MLEGeneCount(tree, geneCountData, geomMean=2, nCondData=1)
```

Note that by default, the software considers 101 possibilities for the number of genes at each internal node, we can change this number using the parameter nPos :

```
myMLE<-MLEGeneCount(tree, geneCountData, geomMean=2, nCondData=1, nPos=30)
```

The optimization will start using initial values for the log birth rate, the log death rate, and the loss rate at the WGD. As a consequence, if we are willing to change these initial values :

```
myMLE<-MLEGeneCount(tree, geneCountData, geomMean=2, nCondData=1,
                    logLamlogMuWgdLR=c(log(0.03), log(0.01), 0.2))
```