Branch length evaluation for Phylogenetic Diversity: the algorithm

Daniel R. Miranda-Esquivel

2018 - 01 - 16

The Algorithm

Deconstructing the functions

First, we load the required libraries:

```
## cleaning
rm(list = ls())

## libraries

## installing and loading the package

##install.packages("../../blepd_0.1.1.tar.gz", repos = NULL, type="source")

library(blepd)

packageVersion("blepd")
```

```
## [1] '0.1.4.2018.1.16.2136'
```

```
## to plot trees

library(ggtree)
```

```
## Loading required package: ggplot2
```

```
## Loading required package: treeio
```

```
## ggtree v1.10.2  For help: https://guangchuangyu.github.io/ggtree
##
## If you use ggtree in published research, please cite:
## Guangchuang Yu, David Smith, Huachen Zhu, Yi Guan, Tommy Tsan-Yuk Lam. ggtree: an R package for visualizat
```

```
library(gridExtra)
```

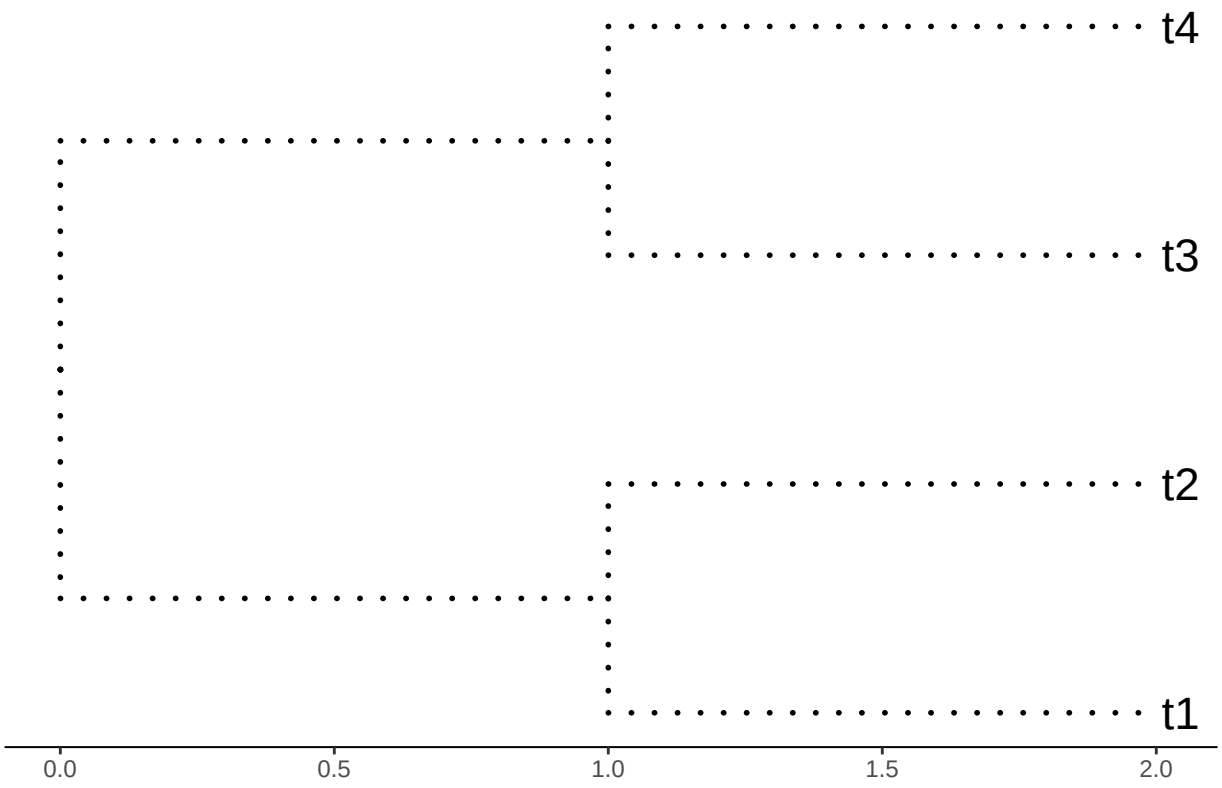
Now, we load the data: trees and distribution

```
## trees

initialTree <- read.tree("../testData/tree00")

initialTree
```

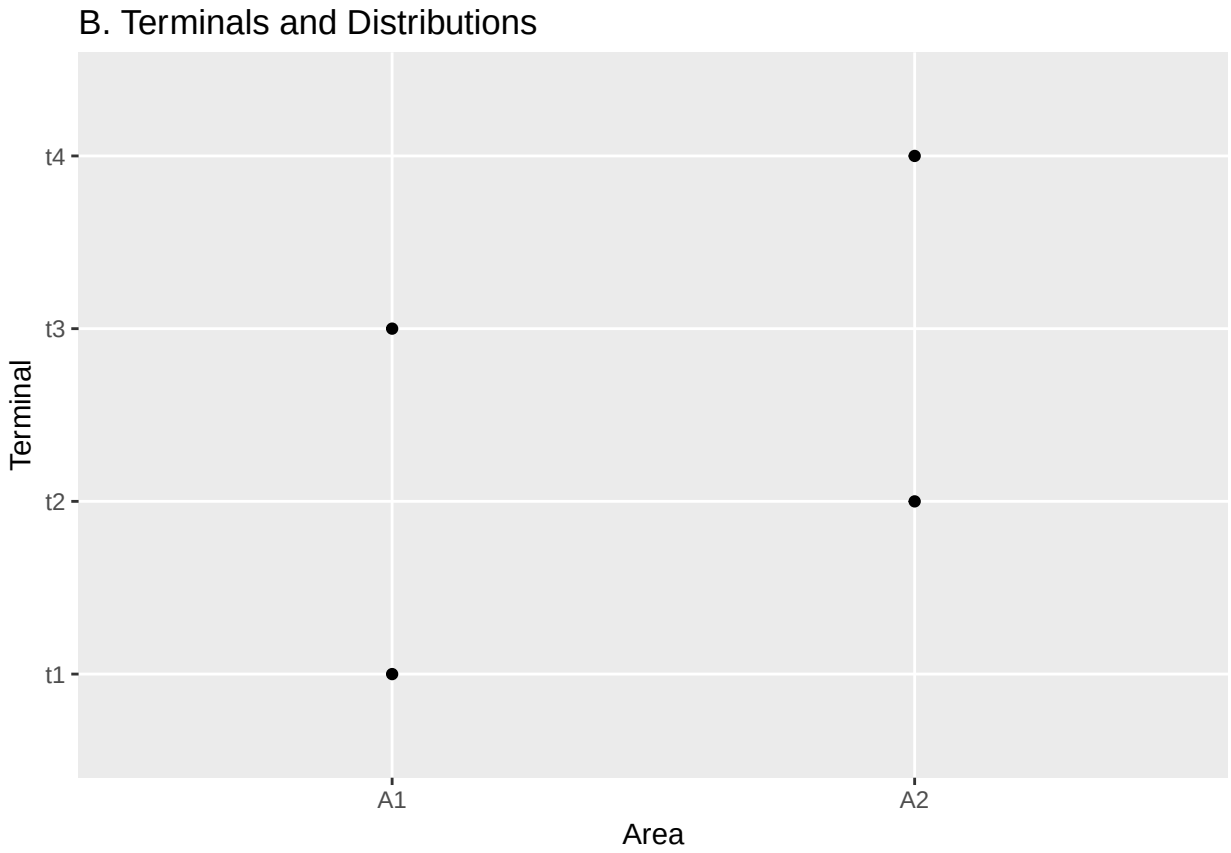
```
##
## Phylogenetic tree with 4 tips and 3 internal nodes.
##
## Tip labels:
```

```
## [1] "t1" "t2" "t3" "t4"
##
## Rooted; includes branch lengths.
```

```
## distributions

dist4taxa <- as.matrix(read.table("../testData/dist4T00",
                                  stringsAsFactors=FALSE,
                                  header=TRUE,
                                  row.names=1)
                       )

## distribution to XY

distXY <- matrix2XY(dist4taxa)

## plotting

## the tree

plotTree <-  ggtree(initialTree, ladderize=TRUE,
             color="black", size=1 , linetype="dotted") +
             geom_tiplab(size=6, color="black") +
             theme_tree2() +
             labs(title = "A. Four terminals, equal branch length")

print(plotTree)
```



```
## the distribution

plotDistrib <- ggplot(data=distXY,
                      aes(x= Area, y= Terminal),
                      size =11) +
               geom_point() +
               labs(title = "B. Terminals and Distributions",
                    y = "Terminal",
                    x = "Area")

plotDistrib
```



We check whether names in both objects, trees and distributions are the same:

```
all(colnames(dist4taxa) == initialTree$tip.label)
```

```
## [1] TRUE
```

We report all branches' length and calculate the PD values.

```
initialTree$edge.length
```

```
## [1] 1 1 1 1 1 1
```

```
initialPD <- myPD(tree=initialTree, distribution = dist4taxa)

initialPD
```

```
## [1] 4 4
```

Single taxon evaluation function

To test the effect of changing a single terminal branch length, we will:

1. create a copy of the initial branch length.
2. calculate the initial PD and get the area(s) with the max value.
3. change the length of a give terminal: t1.
4. recalculate PD and get the area(s) with the max value.
5. compare both areas to evaluate the effect of the perturbation.

```
initialLength <- initialTree$edge.length

bestInitialArea <- row.names(dist4taxa)[which(initialPD == max(initialPD))]

bestInitialArea
```

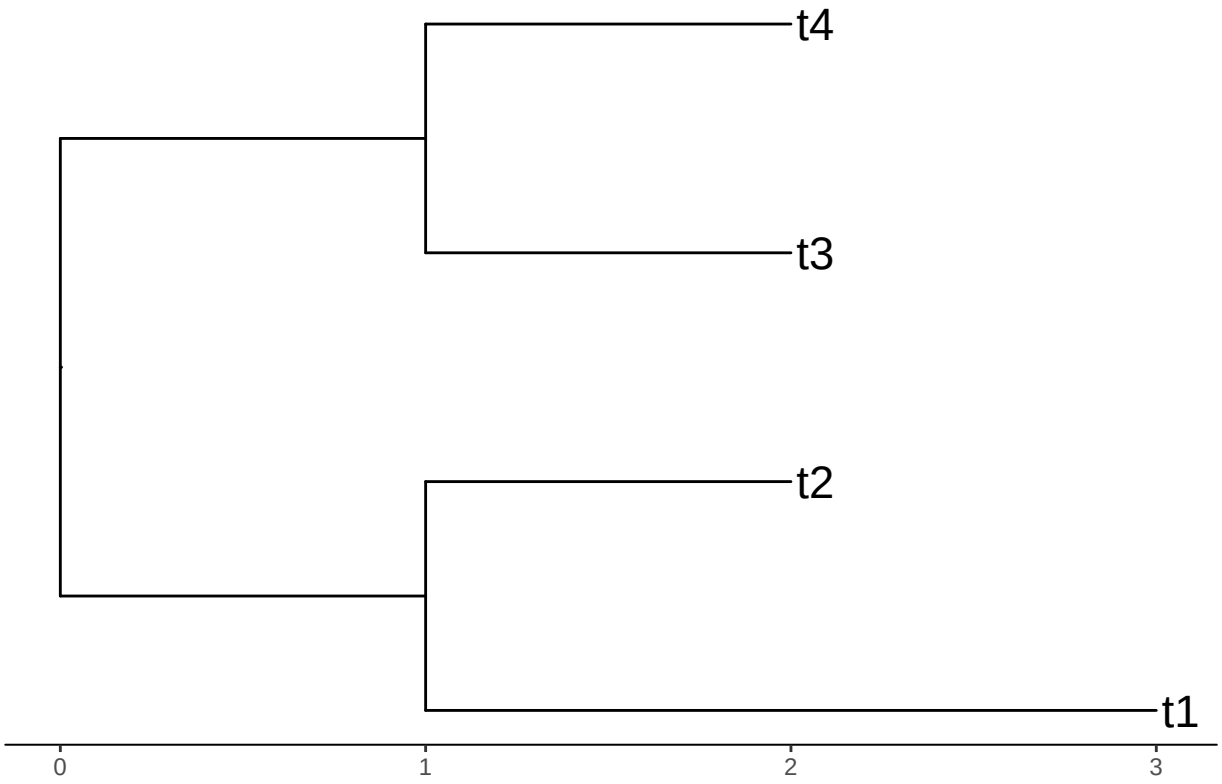
```
## [1] "A1" "A2"
```

```
tipToEval <- "t1"

value <- 2

numberTipToEval <- which(initialTree$tip.label %in% tipToEval)

newTree <- initialTree

## a funtion to create a table binding tree$edge and tree$edge.length
createTable <- function(tree = tree){
                  allDataTable <- tree$edge
                  allDataTable <- cbind (allDataTable, tree$edge.length)
                  return(allDataTable)
                  }

newTree$edge.length[which(createTable(initialTree)[,2] %in% numberTipToEval)] <- value

newTree$edge.length
```

```
## [1] 1 2 1 1 1 1
```

```
plotNewTree <- ggtree(newTree) + theme_tree2() +
               geom_tiplab(size=6, color="black") +
               labs(title = "C. Four terminals, non equal branch length")

print(plotNewTree)
```

C. Four terminals, non equal branch length




```
### PD for the modified tree

modifiedPD <- myPD(tree = newTree, distribution = dist4taxa)

bestModifiedArea <- row.names(dist4taxa)[which(modifiedPD == max(modifiedPD))]

bestInitialArea
```

```
## [1] "A1" "A2"
```

```
bestModifiedArea
```

```
## [1] "A1"
```

```
bestInitialArea == bestModifiedArea
```

```
## [1]  TRUE FALSE
```

As shown, a modification in the branch lenght will impact the area selected, we can consider that the results are *very* sensible to changes in branch length.

THE function

To evaluate the behaviour we can modify a branch length, recalculate the PD value and compare the effect of this perturbation. We can turn this code into a function called evalTerminal, with four parameters:

1. a tree with branch lengths
2. the distribution object
3. the terminal to evaluate

4. the approach to evaluate the terminal, that could be "lower", or "upper" limits, to evaluate the minimal or the maximal value of the branch length where the PD value changes, therefore another area is selected.

```
evalTerminal(tree = initialTree, distribution = dist4taxa, tipToEval = "t1", approach = "lower" )
```

```
## [1] "0.9999" "A1A2"   "A2"     "1"
```

The lower limit when we change the brnach length for terminal A is 0.99, as any change in branch length will modify the area selected from A1A2 to A2, as the tie between the paths terminals A1/A3 vs A2/A4 will be solved in favour of B/D when A1 is shorter.

```
evalTerminal(tree = initialTree, distribution = dist4taxa, tipToEval = "t2", approach = "lower" )
```

```
## [1] "0.9999" "A1A2"   "A1"     "1"
```

A similar result will arrive from changing the terminal branch t2, but in this case the tie is solved to favour A1.

```
evalTerminal(tree = initialTree, distribution = dist4taxa, tipToEval = "t1", approach = "upper" )
```

```
## [1] "1.0001" "A1A2"   "A1"     "1"
```

And, the same result will arrive from changing the branch length of the terminal A from 1 and up, to find the upper limmit, the tie is solved in favour of B, opposite to the solution when we found the lower limit.

In this case, even the smaller change in any terminal branch will modify the results.

We test the effect of the branch length for all terminals.

```
newTree
```

```
##
## Phylogenetic tree with 4 tips and 3 internal nodes.
##
## Tip labels:
## [1] "t1" "t2" "t3" "t4"
##
## Rooted; includes branch lengths.
```

```
modifiedPD
```

```
## [1] 5 4
```

```
bestModifiedArea
```

```
## [1] "A1"
```

```
evalTerminal(tree = newTree, distribution = dist4taxa, tipToEval = "t3", approach = "upper" )
```

```
## [1] "6"  "A1" "*"  "1"
```

```
evalTerminal(tree = newTree, distribution = dist4taxa, tipToEval = "t3", approach = "lower" )
```

```
## [1] "-1e-04" "A1"     "A2"     "1"
```

The THING is the PD difference between areas, and whether this value could be accumulated in a single terminal branch or if the PD value is evenly distributed among all terminal branches and therefore to change the PD value more than one terminal must have to change in order to get another value.

The function to test all terminals at the same time is evalTree, with two parameters: the tree and the distribution. The function returns a data.frame object with 14 fields: labelTerminal, lowerBranchLength, InitialArea, lowerFinalArea, initialLength, upperBranchLength, upperFinalArea, changeLower, changeUpper, deltaUpper, deltaLower, deltaPD, areaDelta, and abDelta.

```
evalTree(tree = initialTree, distribution = dist4taxa)
```

```
##   labelTerminal lowerBranchLength InitialArea lowerFinalArea initialLength
## 1            t1            0.9999        A1A2             A2             1
## 2            t2            0.9999        A1A2             A1             1
## 3            t3            0.9999        A1A2             A2             1
## 4            t4            0.9999        A1A2             A1             1
##   upperBranchLength upperFinalArea changeLower changeUpper deltaUpper
## 1            1.0001             A1    A1A2->A2    A1A2->A1      1e-04
## 2            1.0001             A2    A1A2->A1    A1A2->A2      1e-04
## 3            1.0001             A1    A1A2->A2    A1A2->A1      1e-04
## 4            1.0001             A2    A1A2->A1    A1A2->A2      1e-04
##   deltaLower deltaPD areaDelta abDelta
## 1      1e-04       0        LU       0
## 2      1e-04       0        LU       0
## 3      1e-04       0        LU       0
## 4      1e-04       0        LU       0
```

The column abDelta shows that any change -larger than 0- in the branch length will change the area selected, showing the sensitivity of the PD results to the terminal branch lengths.