

# Package ‘CongreveLamsdell2016’

December 10, 2018

**Version** 1.0.0

**Date** 2018-12-10

**Title** Distance metrics for trees generated by Congreve and Lamsdell (2016)

**Description** Includes the 100 datasets simulated by Congreve and Lamsdell (2016) <doi:10.1111/pala.12236>, and analyses of the partition and quartet distance of reconstructed trees from the generative tree, as analysed by Smith (2019).

**URL** <https://github.com/ms609/CongreveLamsdell2016>

**BugReports** <https://github.com/ms609/Quartet/issues>

**Copyright** Data from Congreve & Lamsdell (2016) released under a CC0 license <doi:10.5061/dryad.7dq0j/1>.

**License** GPL (>= 2)

**Encoding** UTF-8

**Language** en-GB

**Depends** R (>= 3.4.0)

**Imports** Ternary

**Suggests** ape,  
knitr,  
phangorn,  
Quartet,  
TreeSearch (> 0.2.0),  
usethis

**Remotes** ms609/Quartet

**LazyData** true

**ByteCompile** true

**VignetteBuilder** knitr

**RoxygenNote** 6.1.1

**Roxygen** list(markdown = TRUE)

R topics documented:

clCI . . . . .	2
clColours . . . . .	2
clInitializeTernaryQuarts . . . . .	3
clPhyDat . . . . .	3
clPlotQuartets . . . . .	4
clReferenceTree . . . . .	5
clResults . . . . .	6
SplitsToPoints . . . . .	8
<b>Index</b>	<b>9</b>

---

clCI	<i>Congreve Lamsdell consistency indices</i>
------	--

---

**Description**

Congreve Lamsdell consistency indices

**Usage**

clCI

**Format**

An object of class numeric of length 100.

---

clColours	<i>Default colours for analyses</i>
-----------	-------------------------------------

---

**Description**

Default colours for analyses

**Usage**

clColours

**Format**

An object of class character of length 8.

---

clInitializeTernaryQuarts

*Initialize ternary plots for quartet plotting*


---

### Description

Initialize ternary plots for quartet plotting

### Usage

```
clInitializeTernaryQuarts(zoom = 1, padding = 0.1, gridLines = 10,
  fontSize = 1, gridCol = "#EBEBEB", backgroundCol = "#FBFBFD",
  xLim = c(0, 1/zoom) - 0.01, yLim = c(0.5 - (1/zoom), 0.5))
```

```
clInitializeTernarySplits(fontSize = 1, xLim = NULL, yLim = NULL,
  gridCol = "#EBEBEB", backgroundCol = "#FBFBFD", padding = 0.1)
```

### Arguments

zoom	Level of magnification (times), used to adjust ticks and scale.
padding	Padding, passed to TernaryPlot.
gridLines	Number of grid lines, passed to TernaryPlot as grid.lines.
fontSize	Font size, passed to TernaryPlot as lab.cex.
gridCol	Colour, passed to TernaryPlot as grid.col.
backgroundCol	Background colour, passed to TernaryPlot as col.
xLim, yLim	x and y limits, passed to TernaryPlot as xlim, ylim.

### Functions

- clInitializeTernarySplits: Initialize ternary plots for partition plotting

---

clPhyDat

*100 simulated data matrices*


---

### Description

Contains the 100 simulated matrices generated by Congreve & Lamsdell (2016) using a heterogeneous Markov-k model, generated from the [clReferenceTree](#) topology, with all branches sharing an equal length.

### Usage

```
clPhyDat
```

```
clMatrices
```

## Format

- clPhyDat: A list with 100 entries, each comprising a phyDat object of 55 characters for 22 taxa.
- clMatrices: A list with 100 entries, each comprising a list of character tokens for each simulated character, as read from raw nexus files using `ape::read.nexus.data`. The four dummy 'characters' have been removed.

## Source

<http://datadryad.org/resource/doi:10.5061/dryad.7dq0j>

## References

- Congreve, C. R. & Lamsdell, J. C. (2016). Implied weighting and its utility in palaeontological datasets: a study using modelled phylogenetic matrices. *Palaeontology* 59(3), 447–465. doi: [10.1111/pala.12236](https://doi.org/10.1111/pala.12236).
- Congreve, C. R. & Lamsdell, J. C. (2016). Data from: Implied weighting and its utility in palaeontological datasets: a study using modelled phylogenetic matrices. Dryad Digital Repository. doi: [10.5061/dryad.7dq0j](https://doi.org/10.5061/dryad.7dq0j).

---

clPlotQuartets

*Plot results*

---

## Description

Plot results

## Usage

```
clPlotQuartets(dataset, tree, cex = 1.1, pch = 2, col = clColours,
  ...)

clPlotAverageQuartets(dataset, cex = 1.1, pch = 2, col = clColours,
  ...)

clPlotTheseAverageQuartets(dataset, cex = 1.1, pch = 2,
  col = "black", ...)

clPlotSplits(dataset, tree, cex = 1.1, pch = 2, col = clColours, ...)

clPlotTheseAverageSplits(dataset, cex = 1.1, pch = 2, col = "black",
  ...)

clPlotTheseBestAverageSplits(dataset, cex = 1.1, pch = 2,
  col = "black", ...)
```

```

clPlotAverageSplits(dataset, cex = 1.1, pch = 2, col = clColours,
  ...)

clPlotBestAverageSplits(dataset, cex = 1.1, pch = 2, col = clColours,
  ...)

```

### Arguments

dataset	Dataset to plot, for example <a href="#">clBootGcQuartets</a>
tree	Integer specifying which tree to plot
cex, pch, ...	Graphical parameters to pass to <a href="#">JoinTheDots</a>
col	Named vector specifying colour in which to plot each analysis

### Value

Returns [invisible](#).

### Functions

- `clPlotAverageQuartets`: Plots average across all 100 trees
- `clPlotTheseAverageQuartets`: Plot average for single dataset across all 100 trees
- `clPlotSplits`: Splits equivalent of `clPlotQuartets`
- `clPlotTheseAverageSplits`: Splits equivalent of `clPlotTheseAverageQuartets`
- `clPlotTheseBestAverageSplits`: Splits equivalent of `clPlotTheseBestAverageQuartets`
- `clPlotAverageSplits`: Splits equivalent of `clPlotAverageQuartets`
- `clPlotBestAverageSplits`: Splits equivalent of `clPlotAverageQuartets`

### Author(s)

Martin R. Smith

---

clReferenceTree	<i>Tree topology for matrix simulation</i>
-----------------	--

---

### Description

The tree topology used to generate the matrices in [clMatrices](#) Congreve & Lamsdell (2016)

### Usage

```
clReferenceTree
```

### Format

A single phylogenetic tree saved as an object of class `phylo`

## Source

Congreve, C. R. & Lamsdell, J. C. (2016). Implied weighting and its utility in palaeontological datasets: a study using modelled phylogenetic matrices. *Palaeontology* 59(3), 447–465. doi: [10.1111/pala.12236](https://doi.org/10.1111/pala.12236).

## References

- Congreve, C. R. & Lamsdell, J. C. (2016). Implied weighting and its utility in palaeontological datasets: a study using modelled phylogenetic matrices. *Palaeontology* 59(3), 447–465. doi: [10.1111/pala.12236](https://doi.org/10.1111/pala.12236).
- Congreve, C. R. & Lamsdell, J. C. (2016). Data from: Implied weighting and its utility in palaeontological datasets: a study using modelled phylogenetic matrices. Dryad Digital Repository. doi: [10.5061/dryad.7dq0j](https://doi.org/10.5061/dryad.7dq0j).

## Examples

```
library(ape) # Contains tree plotting functions
data(clReferenceTree)
plot(clReferenceTree)
```

---

clResults

*Congreve and Lamsdell tree distances*


---

## Description

Distance of CL trees from generative tree.

## Usage

```
clBremQuartets
clBremPartitions
clMkvPartitions
clMkvQuartets
clBootFreqPartitions
clBootFreqQuartets
clJackFreqPartitions
clJackFreqQuartets
clBootGcPartitions
```

clBootGcQuartets

clJackGcPartitions

clJackGcQuartets

## Format

An object of class `list` of length 7.

## Details

For each of the 100 matrices generated by Congreve & Lamsdell (2016), I conducted phylogenetic analysis under different methods:

- `markov`: Using the Markov K model in MrBayes.
- `equal`: Using equal weights in TNT.
- `implied1`, `implied2`, `implied3`, `implied5`, `implied10`: Using implied weights in TNT, with the concavity constant ( $k$ ) set to 1, 2, 3, 5, or 10
- `impliedC`: By taking the strict consensus of all trees recovered by implied weights parsimony analysis under the  $k$  values 2, 3, 5 and 10 (but not 1).

For each analysis, I recorded the strict consensus of all optimal trees, and also the consensus of trees that were suboptimal by a specified degree.

I then calculated, of the total number of quartets or partitions that were resolved in the reference tree, how many were the same or *different* in the tree that resulted from the phylogenetic analysis, and how many were not resolved in this tree ( $r2$ ).

The data object contains a list whose elements are named after the methods, as listed above.

Each list entry is a three-dimensional array, whose dimensions are:

1. The suboptimality of the tree. Different measures of node support are employed: \* `Mkv`: Posterior probabilities, at 2.5 97.5 \* `Brem`: Bremer supports: the consensus of all trees that are (equal weights) 0, 1, ..., 19, 20 steps less optimal than the optimal tree (implied weights: the consensus of all trees that are 0.73<sup>19</sup> less optimal than the optimal tree). \* `Boot`: Bootstrap supports (symmetric resampling,  $p = 0.33$ ). \* `Jack`: Jackknife supports ( $p = 0.36$ ). Boot and Jack results are reported both as the frequency of splits among replicates, and using the `gc` (Groups Present / Contradicted) measure (Goloboff *et al.* 2003)
2. The number of quartets or partitions in total, the same, different, resolved in the estimated tree but not the generative tree (= 0), resolved in the generative tree but not the estimated tree
3. The number of the matrix, from 1 to 100.

## Source

Congreve, C. R. & Lamsdell, J. C. (2016). Implied weighting and its utility in palaeontological datasets: a study using modelled phylogenetic matrices. *Palaeontology* 59(3), 447–465. doi: [10.1111/pala.12236](https://doi.org/10.1111/pala.12236).

**References**

Goloboff, P. A., J. S. Farris, M. Källersjö, B. Oxelman, M. J. Ramírez, and C. A. Szumik. 2003. Improvements to resampling measures of group support. *Cladistics* 19, 324–332. doi: [10.1016/S07483007\(03\)000604](https://doi.org/10.1016/S07483007(03)000604).

**See Also**

[clMatrices](#), [clReferenceTree](#).

---

SplitsToPoints

---

*Convert splits data to ternary-plottable points*


---

**Description**

Convert splits data to ternary-plottable points

**Usage**

```
SplitsToPoints(itemData)
```

**Arguments**

`itemData`      A line from `clBootGcPartitions` or equivalent

**Value**

Number of partitions resolved in tree 2 only, different, and same in both trees, numbered ready to appear on a ternary plot.



# Index

## \*Topic **datasets**

- clCI, [2](#)
- clColours, [2](#)
- clPhyDat, [3](#)
- clReferenceTree, [5](#)
- clResults, [6](#)

  

- clBootFreqPartitions (clResults), [6](#)
- clBootFreqQuartets (clResults), [6](#)
- clBootGcPartitions (clResults), [6](#)
- clBootGcQuartets, [5](#)
- clBootGcQuartets (clResults), [6](#)
- clBremPartitions (clResults), [6](#)
- clBremQuartets (clResults), [6](#)
- clCI, [2](#)
- clColours, [2](#)
- clInitializeTernaryQuarts, [3](#)
- clInitializeTernarySplits  
    (clInitializeTernaryQuarts), [3](#)
- clJackFreqPartitions (clResults), [6](#)
- clJackFreqQuartets (clResults), [6](#)
- clJackGcPartitions (clResults), [6](#)
- clJackGcQuartets (clResults), [6](#)
- clMatrices, [5](#), [8](#)
- clMatrices (clPhyDat), [3](#)
- clMkvPartitions (clResults), [6](#)
- clMkvQuartets (clResults), [6](#)
- clPhyDat, [3](#)
- clPlotAverageQuartets (clPlotQuartets),  
    [4](#)
- clPlotAverageSplits (clPlotQuartets), [4](#)
- clPlotBestAverageSplits  
    (clPlotQuartets), [4](#)
- clPlotQuartets, [4](#)
- clPlotSplits (clPlotQuartets), [4](#)
- clPlotTheseAverageQuartets  
    (clPlotQuartets), [4](#)
- clPlotTheseAverageSplits  
    (clPlotQuartets), [4](#)

- clPlotTheseBestAverageSplits  
    (clPlotQuartets), [4](#)
- clReferenceTree, [3](#), [5](#), [8](#)
- clResults, [6](#)
- invisible, [5](#)
- JoinTheDots, [5](#)
- SplitsToPoints, [8](#)