

# Package ‘pastis’

February 20, 2015

**Type** Package

**Title** Phylogenetic Assembly with Soft Taxonomic Inferences

**Author** Klaas Hartmann, Gavin Thomas, Arne Mooers, Jeffrey Joy, Walter Jetz

**Maintainer** Gavin Thomas <gavin.thomas@sheffield.ac.uk>

**Depends** R (>= 2.10), caper (>= 0.5.0), ape (>= 3.0.0)

**Description** A pre-processor for mrBayes that assimilates sequences, taxonomic information and tree constraints as per xxx. The main functions of interest for most users will be `pastis_simple`, `pastis_main` and `conch`. The main analysis is conducted with `pastis_simple` or `pastis_main` followed by a manual execution of mrBayes (>3.2). The placement of taxa not contained in the tree constraint can be investigated using `conch`.

**License** GPL (>= 3)

**Encoding** UTF-8

**LazyLoad** yes

**URL** <http://cran.r-project.org/web/packages/pastis/>

**Version** 0.1-2

**Date** 2013-08-30

**Collate** 'pastis-package.R' 'pastis.R'

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2013-09-11 15:23:31

## R topics documented:

<code>pastis-package</code> . . . . .	2
<code>accipitridaeBasicPastis</code> . . . . .	2
<code>accipitridaeFullPastis</code> . . . . .	3
<code>conch</code> . . . . .	3
<code>default_output_template</code> . . . . .	4
<code>pastis_data_1</code> . . . . .	5

pastis_data_2 . . . . .	5
pastis_data_3 . . . . .	5
pastis_data_3_trees . . . . .	6
pastis_main . . . . .	6
pastis_simple . . . . .	8
read_input . . . . .	10

<b>Index</b>	<b>12</b>
--------------	-----------

---

pastis-package	<i>Phylogenetic Assembly with Soft Taxonomic Inferences.</i>
----------------	--

---

## Description

Phylogenetic Assembly with Soft Taxonomic Inferences.

## Details

Package: pastis  
 Type: Package  
 Version: 0.1-2  
 Date: 2013-9-11  
 Depends: caper (>= 0.5.0), ape (>= 3.0.0)  
 Encoding: UTF-8  
 License: GPL (>= 3)  
 LazyLoad: yes  
 URL: <http://cran.r-project.org/web/packages/pastis/>

A pre-processor for mrBayes that assimilates sequences, taxonomic information and tree constraints. The main functions of interest for most users will be `pastis_simple`, `pastis_main` and `conch`. The main analysis is conducted with `pastis_simple` or `pastis_main` followed by a manual execution of mrBayes (>3.2). The placement of taxa not contained in the tree constraint can be investigated using `conch`.

## Author(s)

Klaas Hartmann <[klaas.hartmann@utas.edu.au](mailto:klaas.hartmann@utas.edu.au)> Gavin Thomas <[gavin.thomas@sheffield.ac.uk](mailto:gavin.thomas@sheffield.ac.uk)>

---

accipitridaeBasicPastis	<i>Accipitridae basic data</i>
-------------------------	--------------------------------

---

## Description

Constraint tree and taxon list for Accipitridae

**Author(s)**

Gavin Thomas, 2013-07-22

---

accipitridaeFullPastis

*Accipitridae full data*

---

**Description**

Constraint tree, taxon list, missing clades, sequence data and template for Accipitridae

**Author(s)**

Gavin Thomas, 2013-07-22

---

conch

*mrBayes output interrogator (CONstraint CHECKer)*

---

**Description**

This function examines mrBayes output from a pastis run to examine where in the original constraint tree missing taxa have been placed

**Usage**

```
conch(constraint_tree, mrbayes_output,
      simple_edge_scaling = TRUE, species_set = NA)
```

**Arguments**

`constraint_tree` the constraint tree used with pastis (either the filename or ape phylo tree)

`mrbayes_output` the mrBayes output .t file (either the filename or ape multiPhylo tree object)

`simple_edge_scaling` boolean, see function description

`species_set` if specified, this should be a list of species and output trees will only be generated for these species (the default is all missing species)

## Details

An analysis of the placement of each taxon not contained in the constraint tree is conducted. For each such taxon the edge lengths in the constraint tree are adjusted according to the parameter `edge_scaling`. If `simple_edge_scaling` is `TRUE`, the edges will have length 0 if a taxon is never descendant from a tree and 1 if it is descendant in at least one tree. If `FALSE`, indicate the proportion of sampled trees in which the taxon is descendant from that edge.

The output from this function is useful for checking that missing taxa are placed in appropriate positions relative to the original constraint tree.

Note that this routine has not been optimised and slow (possibly unacceptable) performance is to be expected with large trees and/or large posterior samples.

## Value

NULL. A file for each missing taxon is created in the current directory

## Examples

```
data(pastis_data_3_trees)

## Not run:

# Check constraints for all missing taxa (takes ~6 seconds to run: sped up by Anonymous Reviewer 2)
conch(pastis_data_3_trees[[1]], pastis_data_3_trees[[2]])

## End(Not run)

# Check constraints for missing taxon "a_4"
conch(pastis_data_3_trees[[1]], pastis_data_3_trees[[2]], species_set="a_4")
unlink("taxonposition_a_4.tree")
```

---

default\_output\_template

*A sample output template for pastis\_main and pastis\_simple*

---

## Description

This is the default output template filled in by `pastis_main` and `pastis_simple` to create the input file for `mrBayes`.

## Usage

```
default_output_template()
```

**Details**

If you want to change the parameters in the mrBayes nexus file created by pastis then it is easier to create your own template rather than editing the created nexus files manually (which you would have to do again if you reran pastis).

This is the default template used by pastis and is a good starting point for creating your own template. To have a look at this template try:

```
cat(default_output_template())
```

Once you understand the format, you can create your own template as a string and pass it to pastis.

**Value**

The default output template

---

pastis_data_1	<i>Example data 1</i>
---------------	-----------------------

---

**Description**

Constraint tree and taxon list examples

**Author(s)**

Gavin Thomas, 2013-07-22

---

pastis_data_2	<i>Example data 2</i>
---------------	-----------------------

---

**Description**

Constraint tree and taxon list examples

**Author(s)**

Gavin Thomas, 2013-07-22

---

pastis_data_3	<i>Example data 3</i>
---------------	-----------------------

---

**Description**

Constraint tree, taxon and missing clade examples

**Author(s)**

Gavin Thomas, 2013-07-22

---

pastis\_data\_3\_trees     *Example output*

---

### Description

[[1]] Constraint tree and [[2]] distributions of trees produced in MrBayes using pastis\_data\_3

### Author(s)

Gavin Thomas, 2013-07-22

---

pastis\_main     *Phylogenetic Assembly with Soft Taxonomic Inferences*

---

### Description

This function assimilates sequences, taxonomic information and tree constraints into a mrBayes file. This permits the construction of trees that are compatible with all of these sources of tdata and contain all known taxa.

### Usage

```
pastis_main(pastisData = NULL, constraint_tree,
            taxa_list, missing_clades = NA, sequences = NA,
            output_template = NA, output_file = "output.nex",
            paraphyly_constrains = TRUE,
            monophyly_constrains = TRUE, omit_sequences = FALSE)
```

### Arguments

pastisData	Input data object of class pastisData
output_file	The filename for the mrBayes output file (will be overwritten if it exists)
paraphyly_constrains	If TRUE, missing clades are prevented from entering paraphyletic clades.
monophyly_constrains	If TRUE, missing clades are prevented from entering monophyletic clades.
omit_sequences	If set to TRUE the sequence file (if any) will be ignored. This is useful for testing the constraints created by pastis as mrBayes runs much quicker without sequence data!
constraint_tree	A tree with constraints that are forced to be present in all output trees. Either a filename to a nexus file readable by read.tree or a ape phylo object.
taxa_list	A list of all taxa and their clades. Either a data frame with columns "taxa" and "clade" or a filename for a file readable by read.csv with those columns

- `missing_clades` A file containing missing clades. Each line of the missing clades file consists of the missing clade, the word "include" or "exclude" and a list of the reference clades (all separated by commas). Lines containing "include" specify that a taxon is contained below the MRCA of the reference clades. Lines containing "exclude" specify that the missing clade cannot attach below the MRCA of the reference clades. #'
- `sequences` A file with all the available sequence information in fasta format for details on that format see `read.dna` in the ape package.
- `output_template`  
 The filename for a template for the output nexus file. This file should look like a regular mrBayes input file with special tags replacing content that will be filled by pastis. In particular:  
 <sequences> will be replaced by the sequences (and should go below the MATRIX line)  
 <ntax> the number of taxa (i.e. "ntax=<ntax>" must be somewhere in your template)  
 <nchar> the number of characters  
 <constraints> the constraints will go here  
 <outputfile> where the summaries will be written, (i.e. "sumt filename=<outputfile> burnin ...." should be in your template)  
 see `default_output_template` for an example (which is used by default)

## Details

This is the main function in pastis which assimilates sequences, taxonomic information and tree constraints and creates a mrBayes input file. This input file contains the tree structure specified by `constraint_tree` with missing taxa in `taxa_list` and missing clades in `missing_clades` added and placed loosely in the tree using the constraint logic outlined in Thomas et al. MEE (in review) and Jetz et al. (2012 Nature, 491, 444-448).

See `read_input` for a description of the required format of the input files. At a minimum the constraining input tree and taxa list must be provided.

In addition to the input checks conducted by `read_input` this function also checks for compatibility between the missing genus constraints and constraint tree.

## PASTIS: Phylogenetic Assembly with Soft Taxonomic InferenceS?

A bright motmot was  
 acting quite rowdy- weaving and squawking quite loudly  
 "Pastis is delise" he burped with a sneeze "but why is  
 everything suddenly cloudy?"

– Arne Mooers

## Value

NULL

## See Also

[pastis\\_simple](#) provides a simplified interface to `pastis_main`.

[read\\_input](#) describes the required file formats

[default\\_output\\_template](#) provides an example of the output template (also the default)

## Examples

```
## Not run:
# Generate MrBayes input files with constraints
data(accipitridaeFullPastis)
pastis_main(accipitridaeFullPastis, output_file="Accipitridae.nexus")

data(accipitridaeBasicPastis)
pastis_main(accipitridaeBasicPastis, output_file="AccipitridaeBasic.nexus")

## End(Not run)

data(pastis_data_1)
pastis_main(pastis_data_1, output_file="pastis_data_1")
unlink("pastis_data_1.nexus")

data(pastis_data_2)
pastis_main(pastis_data_2, output_file="pastis_data_2")
unlink("pastis_data_2.nexus")

data(pastis_data_3)
pastis_main(pastis_data_3, output_file="pastis_data_3")
unlink("pastis_data_3.nexus")
```

---

pastis\_simple

*A simplified interface to the main pastis function.*

---

## Description

This function assimilates sequences, taxonomic information and tree constraints into a mrBayes file. This permits the construction of trees that are compatible with all of these sources of tdata and contain all known taxa.

## Usage

```
pastis_simple(pastisData = NULL, base_name,
              paraphyly_constrains = TRUE,
              monophyly_constrains = TRUE, omit_sequences = FALSE)
```

**Arguments**

base_name	The base name for all input files may include a leading directory, but should not include a trailing .
pastisData	Input data object of class pastisData
paraphyly_constrains	If TRUE, missing clades are prevented from entering paraphyletic clades.
monophyly_constrains	If TRUE, missing clades are prevented from entering monophyletic clades.
omit_sequences	If set to TRUE the sequence file (if any) will be ignored. This is useful for testing the constraints created by pastis as mrBayes runs much quicker without sequence data!

**Details**

This is a simplified version of pastis\_main which assumes that (i) the data are input as a pastisData object (e.g. created with read\_input), or (ii) the input files all have the same base (specified by base\_name) with different extensions for each file type:

.sequences: the sequence file (FASTA format)

.tree: the constraint tree (Newick format)

.taxa: a list of taxa (see below)

.missingclades: locations of missing clades (see below)

.template: the template for the mrBayes input file (see default\_output\_template() )

The taxa file consists of a header 'taxon,clade' with each subsequent line containing a taxon,clade pair (separated by a comma)

Each line of the missingclades file consists of the missing clade, the word include or exclude and a list of the reference clades (all separated by commas). Lines containing include specify that a taxon is contained below the MRCA of the reference clades. Lines containing exclude specify that the missing clade cannot attach below the MRCA of the reference clades.

The mrBayes input file is written to base\_name.nexus.

**See Also**

[pastis\\_main](#) provides a more flexible interface.

**PASTIS: Phylogenetic Assembly with Soft Taxonomic InferenceS?**

A bright motmot was  
 acting quite rowdy- weaving and squawking quite loudly  
 "Pastis is delise" he burped with a sneeze "but why is  
 everything suddenly cloudy?"

– Arne Mooers

**Examples**

```
# Generate MrBayes input files with constraints
## Not run:
data(accipitridaeFullPastis)
pastis_simple(accipitridaeFullPastis, base_name="Accipitridae")

data(accipitridaeBasicPastis)
pastis_simple(accipitridaeBasicPastis, base_name="AccipitridaeBasic")

## End(Not run)

data(pastis_data_1)
pastis_simple(pastis_data_1, base_name="pastis_data_1")
unlink("pastis_data_1.nexus")

data(pastis_data_2)
pastis_simple(pastis_data_2, base_name="pastis_data_2")
unlink("pastis_data_2.nexus")

data(pastis_data_3)
pastis_simple(pastis_data_3, base_name="pastis_data_3")
unlink("pastis_data_3.nexus")
```

---

read\_input

*Called by create\_job to read input files*


---

**Description**

Reads the specified input files and performs some basic consistency checking between the inputs.

**Usage**

```
read_input(constraint_tree, taxa_list,
           missing_clades = NA, sequences = NA,
           output_template = NA)
```

**Arguments**

constraint_tree	A tree with constraints that are forced to be present in all output trees. Either a filename to a nexus file readable by read.tree or a ape phylo object.
taxa_list	A list of all taxa and their clades. Either a data frame with columns "taxa" and "clade" or a filename for a file readable by read.csv with those columns
missing_clades	A file containing missing clades. Each line of the missing clades file consists of the missing clade, the word "include" or "exclude" and a list of the reference clades (all separated by commas). Lines containing "include" specify that a taxon is contained below the MRCA of the reference clades. Lines containing "exclude" specify that the missing clade cannot attach below the MRCA of the reference clades. #'

**output\_template**

The filename for a template for the output nexus file. This file should look like a regular mrBayes input file with special tags replacing content that will be filled by pastis. In particular:

<sequences> will be replaced by the sequences (and should go below the MATRIX line)

<ntax> the number of taxa (i.e. "ntax=<ntax>" must be somewhere in your template)

<nchar> the number of characters

<constraints> the constraints will go here

<outputfile> where the summaries will be written, (i.e. "sumt filename=<outputfile> burnin ..." should be in your template)

see default\_output\_template for an example (which is used by default)

**sequences**

A file with all the available sequence information in fasta format for details on that format see read.dna in the ape package.

**Value**

A list with the loaded input

# Index

## \*Topic **datasets**

accipitridaeBasicPastis, [2](#)

accipitridaeFullPastis, [3](#)

pastis\_data\_1, [5](#)

pastis\_data\_2, [5](#)

pastis\_data\_3, [5](#)

pastis\_data\_3\_trees, [6](#)

## \*Topic **package**

pastis-package, [2](#)

accipitridaeBasicPastis, [2](#)

accipitridaeFullPastis, [3](#)

conch, [3](#)

default\_output\_template, [4](#), [8](#)

pastis (pastis-package), [2](#)

pastis-package, [2](#)

pastis\_data\_1, [5](#)

pastis\_data\_2, [5](#)

pastis\_data\_3, [5](#)

pastis\_data\_3\_trees, [6](#)

pastis\_main, [6](#), [9](#)

pastis\_simple, [8](#), [8](#)

read\_input, [8](#), [10](#)