

Project 2: Augmenting functional information about human genes using probabilistic phylogenetic modeling

George G. Vega Yon
vegayon@usc.edu

Department of Preventive Medicine
University of Southern California

October 5, 2017

Agenda

On Genes and Trees

Model

Peeling algorithm

The `aphylo` R package

The `amcmc` R package

Bayesian Estimation of the parameters

Concluding Remarks

Agenda

On Genes and Trees

Model

Peeling algorithm

The `aphylo` R package

The `amcmc` R package

Bayesian Estimation of the parameters

Concluding Remarks

- ▶ A **GO annotation** is an association between a gene and a GO (Gene Ontology) term describing its function, e.g: A gene can be annotated with the GO term GO:0016049, which denotes *cellular growth*.

Overview

- ▶ A **GO annotation** is an association between a gene and a GO (Gene Ontology) term describing its function, e.g: A gene can be annotated with the GO term GO:0016049, which denotes *cellular growth*.
- ▶ **Phylogenetic Tree** represents “inferred evolutionary relationships among various biological species or other entities” (wiki), in this context, our entities are genes.

Overview

- ▶ A **GO annotation** is an association between a gene and a GO (Gene Ontology) term describing its function, e.g: A gene can be annotated with the GO term GO:0016049, which denotes *cellular growth*.
- ▶ **Phylogenetic Tree** represents “inferred evolutionary relationships among various biological species or other entities” (wiki), in this context, our entities are genes.
- ▶ **PANTHER Classification System** (PantherDB), part of the Gene Ontology Consortium, consists on a database of $\sim 15,000$ phylogenetic trees (gene families), and can be linked to the GO terms.

Overview

- ▶ A **GO annotation** is an association between a gene and a GO (Gene Ontology) term describing its function, e.g: A gene can be annotated with the GO term GO:0016049, which denotes *cellular growth*.
- ▶ **Phylogenetic Tree** represents “inferred evolutionary relationships among various biological species or other entities” (wiki), in this context, our entities are genes.
- ▶ **PANTHER Classification System** (PantherDB), part of the Gene Ontology Consortium, consists on a database of ~ 15,000 phylogenetic trees (gene families), and can be linked to the GO terms.

Overview (cont.)

- ▶ Manual Curation of GO terms is good but infeasible:

Overview (cont.)

- ▶ Manual Curation of GO terms is good but infeasible: Out of all the genes present in PantherDB, only ~9% has been annotated (17 years of work)

Overview (cont.)

- ▶ Manual Curation of GO terms is good but infeasible: Out of all the genes present in PantherDB, only ~9% has been annotated (17 years of work)
- ▶ Today, we present a model that uses both: (1) existing gene functional annotations, and (2) phylogenetic trees to infer annotations on un-annotated genes in a *probabilistic* way (so it is not a 0/1 prediction).
- ▶ This predicted functional information will serve as prior covariates in Projects 1 and 3.

Overview (cont.)

- ▶ Manual Curation of GO terms is good but infeasible: Out of all the genes present in PantherDB, only ~9% has been annotated (17 years of work)
- ▶ Today, we present a model that uses both: (1) existing gene functional annotations, and (2) phylogenetic trees to infer annotations on un-annotated genes in a *probabilistic* way (so it is not a 0/1 prediction).
- ▶ This predicted functional information will serve as prior covariates in Projects 1 and 3.

Agenda

On Genes and Trees

Model

Peeling algorithm

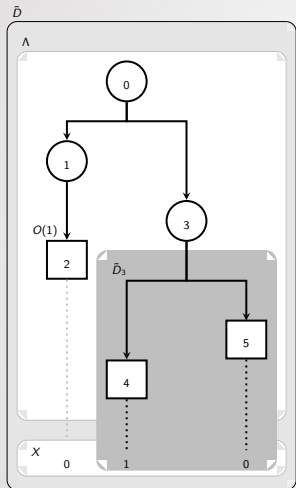
The `aphylo` R package

The `amcmc` R package

Bayesian Estimation of the parameters

Concluding Remarks

Some definitions



Symbol

Description

\tilde{D}

Observed Annotated Tree

Λ

Partially ordered phylogenetic tree (PO tree)

$O(n)$

Offspring of node n

\tilde{D}_n

n -induced Annotated Sub-tree

X

Experimental annotation

Where

$$x_{lp} = \begin{cases} 1 & \text{if the function } p \text{ is believed to be present} \\ 0 & \text{if the function } p \text{ is believed to be absent} \\ 9 & \text{if we don't have information for this node} \end{cases}$$

Formal definitions

A probabilistic model of function propagation

1. For any given node, we can write down the probability of observing a *functional state* as a function of some model parameters and its offspring.

A probabilistic model of function propagation

1. For any given node, we can write down the probability of observing a *functional state* as a function of some model parameters and its offspring.
2. This version of our model has five parameters (probabilities):

A probabilistic model of function propagation

1. For any given node, we can write down the probability of observing a *functional state* as a function of some model parameters and its offspring.
2. This version of our model has five parameters (probabilities):
 - 2.1 Root node had a function: π ,
 - 2.2 Gain of function: μ_0 ,
 - 2.3 Loss of function: μ_1 .
 - 2.4 Misclassification of:
 - ▶ A missing function as present, ψ_0 , and
 - ▶ A present function as missing, ψ_1

A probabilistic model of function propagation

1. For any given node, we can write down the probability of observing a *functional state* as a function of some model parameters and its offspring.
2. This version of our model has five parameters (probabilities):
 - 2.1 Root node had a function: π ,
 - 2.2 Gain of function: μ_0 ,
 - 2.3 Loss of function: μ_1 .
 - 2.4 Misclassification of:
 - ▶ A missing function as present, ψ_0 , and
 - ▶ A present function as missing, ψ_1

All five parameters are assumed to be equal across functions, this is, $\pi, \mu_0, \mu_1, \psi_0$, and ψ_1 are assumed to be independent of the functions that are analyzed.

A probabilistic model of function propagation

1. For any given node, we can write down the probability of observing a *functional state* as a function of some model parameters and its offspring.
2. This version of our model has five parameters (probabilities):
 - 2.1 Root node had a function: π ,
 - 2.2 Gain of function: μ_0 ,
 - 2.3 Loss of function: μ_1 .
 - 2.4 Misclassification of:
 - ▶ A missing function as present, ψ_0 , and
 - ▶ A present function as missing, ψ_1

All five parameters are assumed to be equal across functions, this is, $\pi, \mu_0, \mu_1, \psi_0$, and ψ_1 are assumed to be independent of the functions that are analyzed.

3. In this presentation, we will focus on the case that $P = 1$.

Agenda

On Genes and Trees

Model

Peeling algorithm

The `aphylo` R package

The `amcmc` R package

Bayesian Estimation of the parameters

Concluding Remarks

Peeling phylogenies

Given an Experimentally Annotated (PO) Phylogenetic Tree, the likelihood computation on a single function is as follows.

Peeling phylogenies

Given an Experimentally Annotated (PO) Phylogenetic Tree, the likelihood computation on a single function is as follows.

1. Create an matrix Pr of size $2 \times |N|$,

Peeling phylogenies

Given an Experimentally Annotated (PO) Phylogenetic Tree, the likelihood computation on a single function is as follows.

1. Create an matrix Pr of size $2 \times |N|$,
2. For node $n \in \{|N|, |N| - 1, \dots, 1, 0\}$ (the peeling sequence) do:

Peeling phylogenies

Given an Experimentally Annotated (PO) Phylogenetic Tree, the likelihood computation on a single function is as follows.

1. Create an matrix Pr of size $2 \times |N|$,
2. For node $n \in \{|N|, |N| - 1, \dots, 1, 0\}$ (the peeling sequence) do:
 - 2.1 For $z_n \in \{0, 1\}$ do:
 - 2.1.1 Set $\text{Pr}_{n,z_n} = \begin{cases} \Pr(Z_n = z_n \mid X_n = X_n) & \text{If } n \text{ is a leaf} \\ \Pr(Z_n = z_n \mid \tilde{D}_n) & \text{otherwise} \end{cases}$
 - 2.1.2 Next z_n
 - 2.2 Next n

Peeling phylogenies

Given an Experimentally Annotated (PO) Phylogenetic Tree, the likelihood computation on a single function is as follows.

1. Create an matrix Pr of size $2 \times |N|$,
2. For node $n \in \{|N|, |N| - 1, \dots, 1, 0\}$ (the peeling sequence) do:
 - 2.1 For $z_n \in \{0, 1\}$ do:
 - 2.1.1 Set $\text{Pr}_{n,z_n} = \begin{cases} \text{Pr}(Z_n = z_n \mid X_n = X_n) & \text{If } n \text{ is a leaf} \\ \text{Pr}(Z_n = z_n \mid \tilde{D}_n) & \text{otherwise} \end{cases}$
 - 2.1.2 Next z_n
 - 2.2 Next n
3. At this point the matrix Pr should be completely filled, so following (3), we can compute

$$L(\psi, \mu, \pi \mid \tilde{D}) = \sum_{z_0 \in \{0,1\}} \text{Pr}(Z_0 = z_0 \mid \pi) \text{Pr}_{0,z_0}$$

Peeling phylogenies

Given an Experimentally Annotated (PO) Phylogenetic Tree, the likelihood computation on a single function is as follows.

1. Create an matrix Pr of size $2 \times |N|$,
2. For node $n \in \{|N|, |N| - 1, \dots, 1, 0\}$ (the peeling sequence) do:
 - 2.1 For $z_n \in \{0, 1\}$ do:
 - 2.1.1 Set $\text{Pr}_{n,z_n} = \begin{cases} \text{Pr}(Z_n = z_n \mid X_n = X_n) & \text{If } n \text{ is a leaf} \\ \text{Pr}(Z_n = z_n \mid \tilde{D}_n) & \text{otherwise} \end{cases}$
 - 2.1.2 Next z_n
 - 2.2 Next n
3. At this point the matrix Pr should be completely filled, so following (3), we can compute

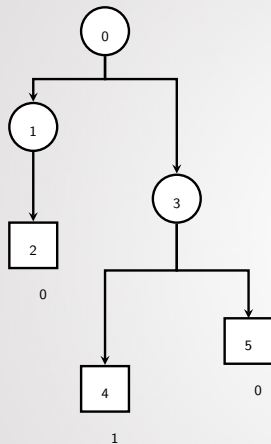
$$L(\psi, \mu, \pi \mid \tilde{D}) = \sum_{z_0 \in \{0,1\}} \text{Pr}(Z_0 = z_0 \mid \pi) \text{Pr}_{0,z_0}$$

Let's see an example!

[details](#)



Peeling algorithm



- Let's calculate the likelihood of observing this tree with the following parameters:

$$\psi_0 = 0.1$$

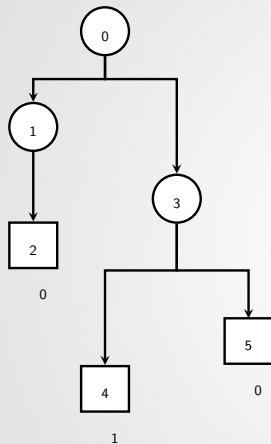
$$\psi_1 = 0.05$$

$$\mu_0 = 0.04$$

$$\mu_1 = 0.01$$

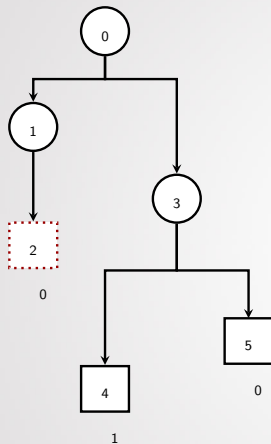
$$\pi = 0.5$$

Peeling algorithm (cont. 1)



	State 0	State 1
0		
1		
2		
3		
4		
5		

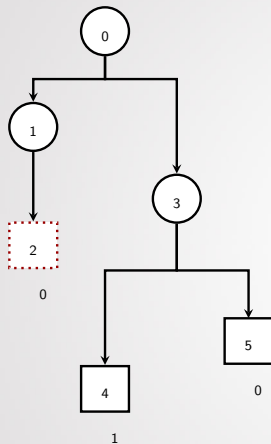
Peeling algorithm (cont. 1)



	State 0	State 1
0		
1		
2	0.9000	
3		
4		
5		

$$\Pr(Z_2 = 0 \mid X_2 = 0) = 1 - \psi_0 = 0.9$$

Peeling algorithm (cont. 1)

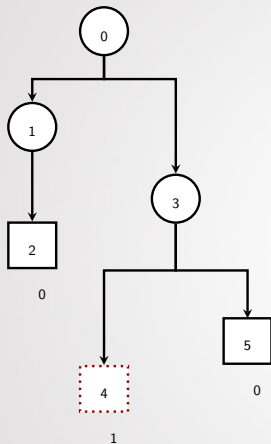


	State 0	State 1
0		
1		
2	0.9000	0.0500
3		
4		
5		

$$\Pr(Z_2 = 0 \mid X_2 = 0) = 1 - \psi_0 = 0.9$$

$$\Pr(Z_2 = 1 \mid X_2 = 0) = \psi_1 = 0.05$$

Peeling algorithm (cont. 1)



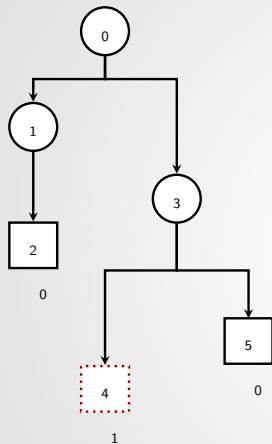
	State 0	State 1
0		
1		
2	0.9000	0.0500
3		
4	0.1000	
5		

$$\Pr(Z_2 = 0 \mid X_2 = 0) = 1 - \psi_0 = 0.9$$

$$\Pr(Z_2 = 1 \mid X_2 = 0) = \psi_1 = 0.05$$

$$\Pr(Z_4 = 0 \mid X_4 = 1) = \psi_0 = 0.1$$

Peeling algorithm (cont. 1)



	State 0	State 1
0		
1		
2	0.9000	0.0500
3		
4	0.1000	0.9500
5		

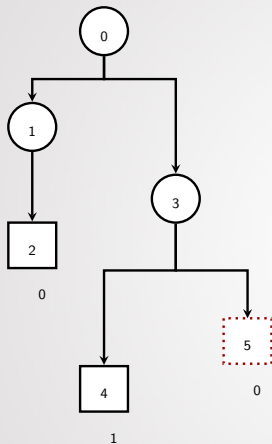
$$\Pr(Z_2 = 0 \mid X_2 = 0) = 1 - \psi_0 = 0.9$$

$$\Pr(Z_2 = 1 \mid X_2 = 0) = \psi_1 = 0.05$$

$$\Pr(Z_4 = 0 \mid X_4 = 1) = \psi_0 = 0.1$$

$$\Pr(Z_4 = 1 \mid X_4 = 1) = 1 - \psi_1 = 0.95$$

Peeling algorithm (cont. 1)



	State 0	State 1
0		
1		
2	0.9000	0.0500
3		
4	0.1000	0.9500
5	0.9000	

$$\Pr(Z_2 = 0 \mid X_2 = 0) = 1 - \psi_0 = 0.9$$

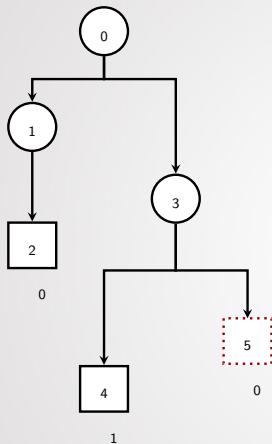
$$\Pr(Z_2 = 1 \mid X_2 = 0) = \psi_1 = 0.05$$

$$\Pr(Z_4 = 0 \mid X_4 = 1) = \psi_0 = 0.1$$

$$\Pr(Z_4 = 1 \mid X_4 = 1) = 1 - \psi_1 = 0.95$$

$$\Pr(Z_5 = 0 \mid X_5 = 0) = 1 - \psi_0 = 0.9$$

Peeling algorithm (cont. 1)



	State 0	State 1
0		
1		
2	0.9000	0.0500
3		
4	0.1000	0.9500
5	0.9000	0.0500

$$\Pr(Z_2 = 0 \mid X_2 = 0) = 1 - \psi_0 = 0.9$$

$$\Pr(Z_2 = 1 \mid X_2 = 0) = \psi_1 = 0.05$$

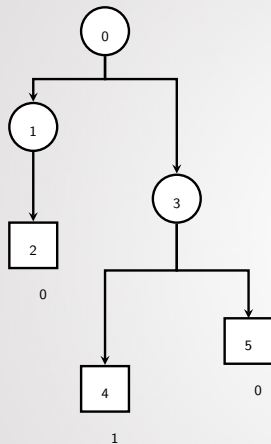
$$\Pr(Z_4 = 0 \mid X_4 = 1) = \psi_0 = 0.1$$

$$\Pr(Z_4 = 1 \mid X_4 = 1) = 1 - \psi_1 = 0.95$$

$$\Pr(Z_5 = 0 \mid X_5 = 0) = 1 - \psi_0 = 0.9$$

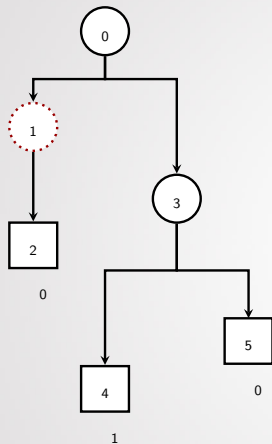
$$\Pr(Z_5 = 1 \mid X_5 = 0) = \psi_1 = 0.05$$

Peeling algorithm (cont. 2)



	State 0	State 1
0		
1		
2	0.9000	0.0500
3		
4	0.1000	0.9500
5	0.9000	0.0500

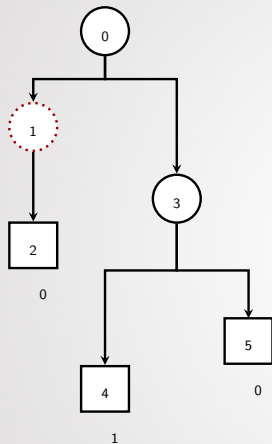
Peeling algorithm (cont. 2)



	State 0	State 1
0		
1		
2	0.9000	0.0500
3		
4	0.1000	0.9500
5	0.9000	0.0500

$$\Pr(Z_1 = 0 \mid \tilde{D}_1) = \Pr(Z_2 = 0 \mid X_2 = 0)\Pr(Z_2 = 0 \mid Z_1 = 0) + \Pr(Z_2 = 1 \mid X_2 = 0)\Pr(Z_2 = 1 \mid Z_1 = 0)$$

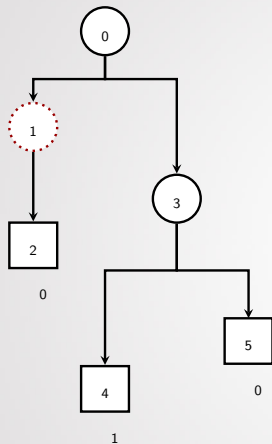
Peeling algorithm (cont. 2)



	State 0	State 1
0		
1	0.8660	
2	0.9000	0.0500
3		
4	0.1000	0.9500
5	0.9000	0.0500

$$\begin{aligned}
 \Pr(Z_1 = 0 \mid \tilde{D}_1) &= \Pr(Z_2 = 0 \mid X_2 = 0)\Pr(Z_2 = 0 \mid Z_1 = 0) + \\
 &\quad \Pr(Z_2 = 1 \mid X_2 = 0)\Pr(Z_2 = 1 \mid Z_1 = 0) \\
 &= 0.9000 \times 0.96 + 0.0500 \times 0.04 = 0.866
 \end{aligned}$$

Peeling algorithm (cont. 2)

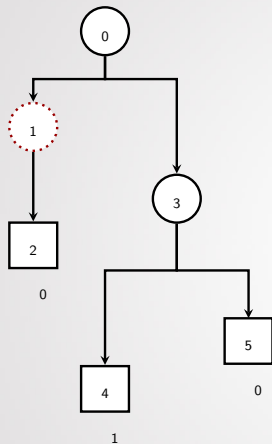


	State 0	State 1
0		
1	0.8660	
2	0.9000	0.0500
3		
4	0.1000	0.9500
5	0.9000	0.0500

$$\begin{aligned} \Pr(Z_1 = 0 \mid \tilde{D}_1) &= \Pr(Z_2 = 0 \mid X_2 = 0)\Pr(Z_2 = 0 \mid Z_1 = 0) + \\ &\quad \Pr(Z_2 = 1 \mid X_2 = 0)\Pr(Z_2 = 1 \mid Z_1 = 0) \\ &= 0.9000 \times 0.96 + 0.0500 \times 0.04 = 0.866 \end{aligned}$$

$$\begin{aligned} \Pr(Z_1 = 1 \mid \tilde{D}_1) &= \Pr(Z_2 = 0 \mid X_2 = 0)\Pr(Z_2 = 0 \mid Z_1 = 1) + \\ &\quad \Pr(Z_2 = 1 \mid X_2 = 0)\Pr(Z_2 = 1 \mid Z_1 = 1) \end{aligned}$$

Peeling algorithm (cont. 2)

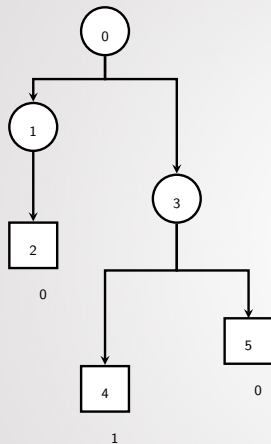


	State 0	State 1
0		
1	0.8660	0.0585
2	0.9000	0.0500
3		
4	0.1000	0.9500
5	0.9000	0.0500

$$\begin{aligned}\Pr(Z_1 = 0 \mid \tilde{D}_1) &= \Pr(Z_2 = 0 \mid X_2 = 0)\Pr(Z_2 = 0 \mid Z_1 = 0) + \\ &\quad \Pr(Z_2 = 1 \mid X_2 = 0)\Pr(Z_2 = 1 \mid Z_1 = 0) \\ &= 0.9000 \times 0.96 + 0.0500 \times 0.04 = 0.866\end{aligned}$$

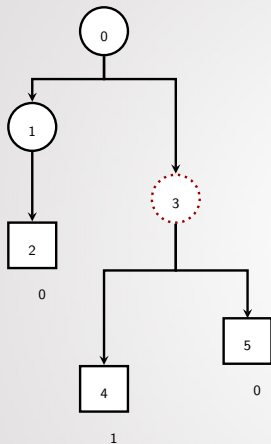
$$\begin{aligned}\Pr(Z_1 = 1 \mid \tilde{D}_1) &= \Pr(Z_2 = 0 \mid X_2 = 0)\Pr(Z_2 = 0 \mid Z_1 = 1) + \\ &\quad \Pr(Z_2 = 1 \mid X_2 = 0)\Pr(Z_2 = 1 \mid Z_1 = 1) \\ &= 0.9000 \times 0.01 + 0.0500 \times 0.99 = 0.0585\end{aligned}$$

Peeling algorithm (cont. 3)



	State 0	State 1
0		
1	0.8660	0.0585
2	0.9000	0.0500
3		
4	0.1000	0.9500
5	0.9000	0.0500

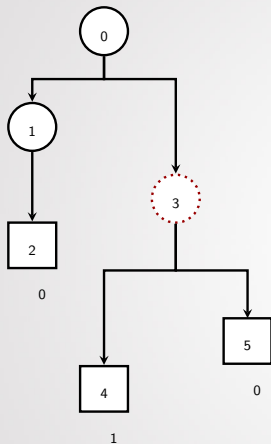
Peeling algorithm (cont. 3)



	State 0	State 1
0		
1	0.8660	0.0585
2	0.9000	0.0500
3		
4	0.1000	0.9500
5	0.9000	0.0500

$$\Pr(Z_3 = 0 \mid \tilde{D}_3) = \prod_{m \in \{4,5\}} \sum_{z_m \in \{0,1\}} \Pr(Z_m = z_m \mid \tilde{D}_m) \Pr(Z_m = z_m \mid Z_3 = 0)$$

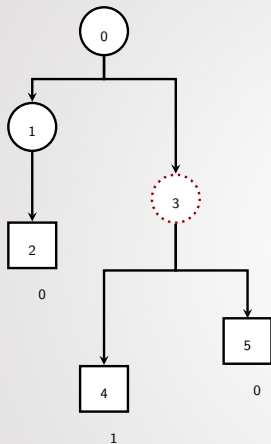
Peeling algorithm (cont. 3)



	State 0	State 1
0		
1	0.8660	0.0585
2	0.9000	0.0500
3		
4	0.1000	0.9500
5	0.9000	0.0500

$$\begin{aligned}
 \Pr(Z_3 = 0 \mid \tilde{D}_3) &= \prod_{m \in \{4,5\}} \sum_{z_m \in \{0,1\}} \Pr(Z_m = z_m \mid \tilde{D}_m) \Pr(Z_m = z_m \mid Z_3 = 0) \\
 &= (0.1(1 - \mu_0) + 0.95 \times \mu_0) \times (0.9(1 - \mu_0) + 0.05 \times \mu_0)
 \end{aligned}$$

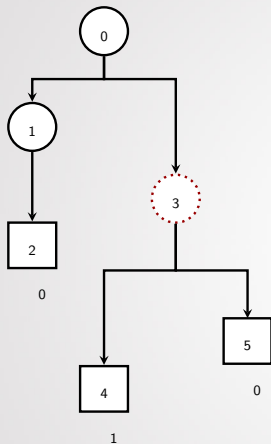
Peeling algorithm (cont. 3)



	State 0	State 1
0		
1	0.8660	0.0585
2	0.9000	0.0500
3		
4	0.1000	0.9500
5	0.9000	0.0500

$$\begin{aligned}
 \Pr(Z_3 = 0 \mid \tilde{D}_3) &= \prod_{m \in \{4,5\}} \sum_{z_m \in \{0,1\}} \Pr(Z_m = z_m \mid \tilde{D}_m) \Pr(Z_m = z_m \mid Z_3 = 0) \\
 &= \left(0.1(1 - \mu_0) + 0.95 \times \mu_0\right) \times \left(0.9(1 - \mu_0) + 0.05 \times \mu_0\right) \\
 &= (0.1(1 - 0.04) + 0.95 \times 0.04) \times (0.9(1 - 0.04) + 0.05 \times 0.04)
 \end{aligned}$$

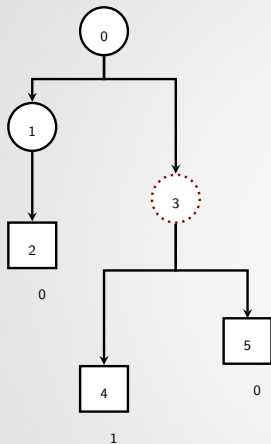
Peeling algorithm (cont. 3)



	State 0	State 1
0		
1	0.8660	0.0585
2	0.9000	0.0500
3	0.1160	
4	0.1000	0.9500
5	0.9000	0.0500

$$\begin{aligned}
 \Pr(Z_3 = 0 \mid \tilde{D}_3) &= \prod_{m \in \{4,5\}} \sum_{z_m \in \{0,1\}} \Pr(Z_m = z_m \mid \tilde{D}_m) \Pr(Z_m = z_m \mid Z_3 = 0) \\
 &= \left(0.1(1 - \mu_0) + 0.95 \times \mu_0\right) \times \left(0.9(1 - \mu_0) + 0.05 \times \mu_0\right) \\
 &= (0.1(1 - 0.04) + 0.95 \times 0.04) \times (0.9(1 - 0.04) + 0.05 \times 0.04) \\
 &= \mathbf{0.116}
 \end{aligned}$$

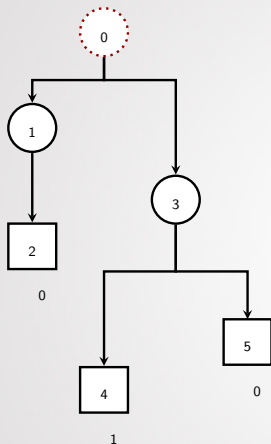
Peeling algorithm (cont. 3)



	State 0	State 1
0		
1	0.8660	0.0585
2	0.9000	0.0500
3	0.1160	0.0551
4	0.1000	0.9500
5	0.9000	0.0500

$$\begin{aligned}
 \Pr(Z_3 = 0 \mid \tilde{D}_3) &= \prod_{m \in \{4,5\}} \sum_{z_m \in \{0,1\}} \Pr(Z_m = z_m \mid \tilde{D}_m) \Pr(Z_m = z_m \mid Z_3 = 0) \\
 &= \left(0.1(1 - \mu_0) + 0.95 \times \mu_0\right) \times \left(0.9(1 - \mu_0) + 0.05 \times \mu_0\right) \\
 &= (0.1(1 - 0.04) + 0.95 \times 0.04) \times (0.9(1 - 0.04) + 0.05 \times 0.04) \\
 &= \mathbf{0.116}
 \end{aligned}$$

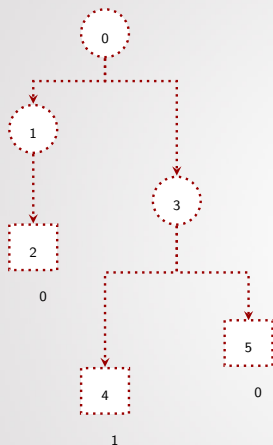
Peeling algorithm (cont. 3)



	State 0	State 1
0	0.0947	0.0037
1	0.8660	0.0585
2	0.9000	0.0500
3	0.1160	0.0551
4	0.1000	0.9500
5	0.9000	0.0500

$$\begin{aligned}
 \Pr(Z_3 = 0 \mid \tilde{D}_3) &= \prod_{m \in \{4,5\}} \sum_{z_m \in \{0,1\}} \Pr(Z_m = z_m \mid \tilde{D}_m) \Pr(Z_m = z_m \mid Z_3 = 0) \\
 &= \left(0.1(1 - \mu_0) + 0.95 \times \mu_0\right) \times \left(0.9(1 - \mu_0) + 0.05 \times \mu_0\right) \\
 &= (0.1(1 - 0.04) + 0.95 \times 0.04) \times (0.9(1 - 0.04) + 0.05 \times 0.04) \\
 &= 0.116
 \end{aligned}$$

Peeling algorithm (cont. 3)



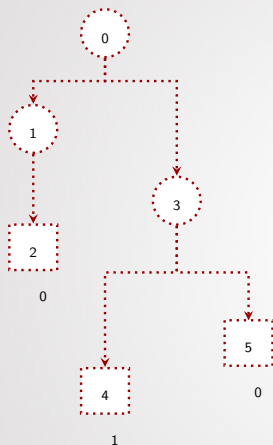
	State 0	State 1
0	0.0947	0.0037
1	0.8660	0.0585
2	0.9000	0.0500
3	0.1160	0.0551
4	0.1000	0.9500
5	0.9000	0.0500

$$\begin{aligned}
 \Pr(Z_3 = 0 \mid \tilde{D}_3) &= \prod_{m \in \{4,5\}} \sum_{z_m \in \{0,1\}} \Pr(Z_m = z_m \mid \tilde{D}_m) \Pr(Z_m = z_m \mid Z_3 = 0) \\
 &= \left(0.1(1 - \mu_0) + 0.95 \times \mu_0\right) \times \left(0.9(1 - \mu_0) + 0.05 \times \mu_0\right) \\
 &= (0.1(1 - 0.04) + 0.95 \times 0.04) \times (0.9(1 - 0.04) + 0.05 \times 0.04) \\
 &= 0.116
 \end{aligned}$$

Finally, the likelihood of this tree is:

$$\mathcal{L}(\psi, \mu, \Pi \mid \tilde{D}) = (1 - \pi) \Pr(Z_0 = 0 \mid \tilde{D}_0) + \pi \Pr(Z_0 = 1 \mid \tilde{D}_0)$$

Peeling algorithm (cont. 3)



	State 0	State 1
0	0.0947	0.0037
1	0.8660	0.0585
2	0.9000	0.0500
3	0.1160	0.0551
4	0.1000	0.9500
5	0.9000	0.0500

$$\begin{aligned}
 \Pr(Z_3 = 0 \mid \tilde{D}_3) &= \prod_{m \in \{4,5\}} \sum_{z_m \in \{0,1\}} \Pr(Z_m = z_m \mid \tilde{D}_m) \Pr(Z_m = z_m \mid Z_3 = 0) \\
 &= (0.1(1 - \mu_0) + 0.95 \times \mu_0) \times (0.9(1 - \mu_0) + 0.05 \times \mu_0) \\
 &= (0.1(1 - 0.04) + 0.95 \times 0.04) \times (0.9(1 - 0.04) + 0.05 \times 0.04) \\
 &= 0.116
 \end{aligned}$$

Finally, the likelihood of this tree is:

$$\begin{aligned}
 \mathcal{L}(\psi, \mu, \Pi \mid \tilde{D}) &= (1 - \pi) \Pr(Z_0 = 0 \mid \tilde{D}_0) + \pi \Pr(Z_0 = 1 \mid \tilde{D}_0) \\
 &= (1 - 0.5) \times 0.0947 + 0.5 \times 0.0037 = 0.0492
 \end{aligned}$$

Agenda

On Genes and Trees

Model

Peeling algorithm

The `aphylo` R package

The `amcmc` R package

Bayesian Estimation of the parameters

Concluding Remarks

aphylo in a nutshell

With C++ (RcppArmadillo) under-the-hood, aphylo:

1. Defines an S3 class for representing partially ordered trees, including validation of it.
2. Includes from the basic methods: `print`, `plot`, and `summary`; to more advanced such as `prune`.
3. Defines an S3 class for representing *annotated* partially ordered trees.
4. Includes functions for converting from and to `phylo` class objects from the `ape` package (most used/cited package in Phylogenetics in R with ~25K downloads/month)
5. Implements the loglikelihood calculation of our model.

We'll talk about estimation later...

Examples: Simulating Trees

```
set.seed(80)
tree <- sim_tree(5)
tree
```

```
##
## A PARTIALLY ORDERED PHYLOGENETIC TREE
##
## # Internal nodes: 4
## # Leaf nodes      : 5
##
## Leaf nodes labels:
##   4, 5, 6, 7, 8.
##
## Internal nodes labels:
##   0, 1, 2, 3.
```

```
atree <- sim_annotated_tree(
  tree = tree, P = 2,
  psi   = c(.05, .05),
  mu    = c(.2, .1),
  Pi    = .01)
atree
```

```
##
## A PARTIALLY ORDERED PHYLOGENETIC TREE
##
## # Internal nodes: 4
## # Leaf nodes      : 5
##
## Leaf nodes labels:
##   4, 5, 6, 7, 8.
##
## Internal nodes labels:
##   0, 1, 2, 3.
##
## ANNOTATIONS:
##   fun0000 fun0001
```

Examples: Visualizing annotated data

```
plot(atree)
```

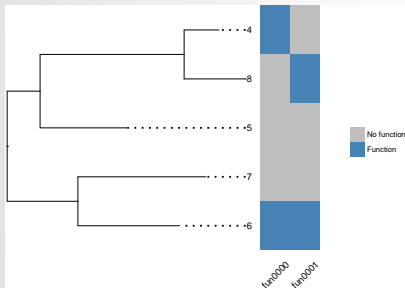
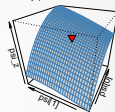


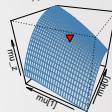
Figure 1: Visualization of annotations and tree structure.

```
plot_LogLike(atree)
```

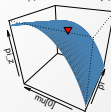
$\mu_0 = 0.1500$ $\mu_1 = 0.1500$ and $\pi = 0.1500$



$\psi_0 = 0.1500$ $\psi_1 = 0.1500$ and $\pi = 0.1500$



$\psi_0 = 0.1500$ $\psi_1 = 0.1500$ and $\mu_1 = 0.1500$



π Root node probabilities
 ψ Misclassification probabilities
 μ Loss/Gain probabilities

Figure 2: LogLikelihood surface of the simulated data

Example: Interaction with ape

```
as.apephylo(atree)
```

```
##  
## Phylogenetic tree with 5 tips and 4 internal nodes.  
##  
## Tip labels:  
## [1] "6" "7" "8" "4" "5"  
## Node labels:  
## [1] "0" "3" "2" "1"  
##  
## Rooted; includes branch lengths.
```

```
# we can go back using:  
# as.po_tree(as.apephylo(atree))
```

Example: Tree pruning

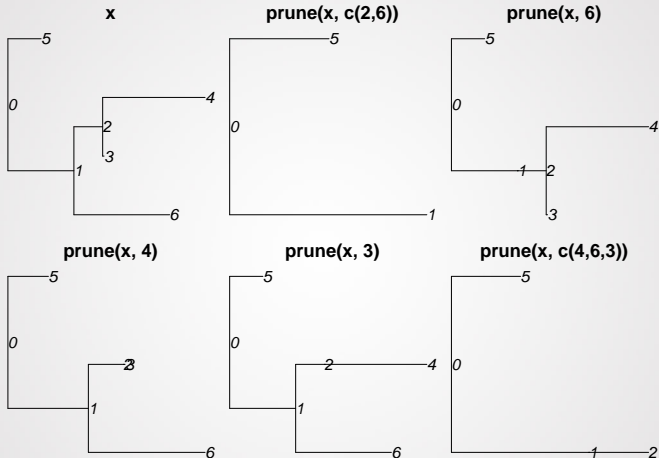


Figure 3: Pruning trees

Example: Tree pruning (cont.)

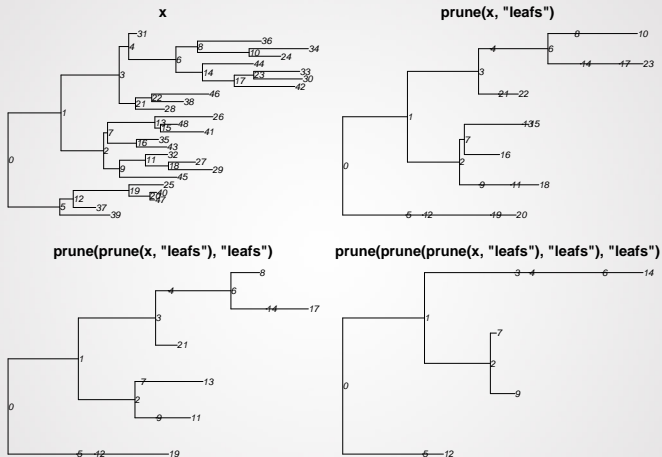


Figure 4: Pruning trees recursively



Agenda

On Genes and Trees

Model

Peeling algorithm

The `aphylo` R package

The `amcmc` R package

Bayesian Estimation of the parameters

Concluding Remarks

Yet another MCMC package

You may be wondering why, well:

1. Allows running multiple chains simultaneously (parallel)
2. Overall faster than other Metrop MCMC algorithms (from our experience)
3. Planning to include other types of kernels (the Handbook of MCMC)
4. Implements reflective boundaries random-walk kernel

Example: MCMC

```
# Loading the packages
library(amcmc)
library(coda)

# Defining the ll function (data was already defined)
ll <- function(x, D) {
  x <- log(dnorm(D, x[1], x[2]))
  sum(x)
}

ans <- MCMC(
  # Ll function and the starting parameters
  ll, c(mu=1, sigma=1),
  # How many steps, thinning, and burn-in
  nbatch = 1e5, thin=10, burnin = 1e4,
  # Kernel parameters
  scale = .1, ub = 10, lb = c(-10, 0),
  # How many parallel chains
  nchains = 4,
  # Further arguments passed to ll
  D=D
)
```

Example: MCMC (cont. 1)

ans is of class mcmc from the coda package

```
summary(ans)
```

```
##
## Iterations = 10010:1e+05
## Thinning interval = 10
## Number of chains = 4
## Sample size per chain = 9000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##      Mean      SD Naive SE Time-series SE
## mu      2.51 0.0935 0.000493      0.000579
## sigma   2.96 0.0668 0.000352      0.000373
##
## 2. Quantiles for each variable:
##
##      2.5%  25%  50%  75%  97.5%
## mu      2.33 2.45 2.51 2.58  2.70
## sigma   2.83 2.91 2.96 3.00  3.09
```

Example: MCMC (cont. 2)

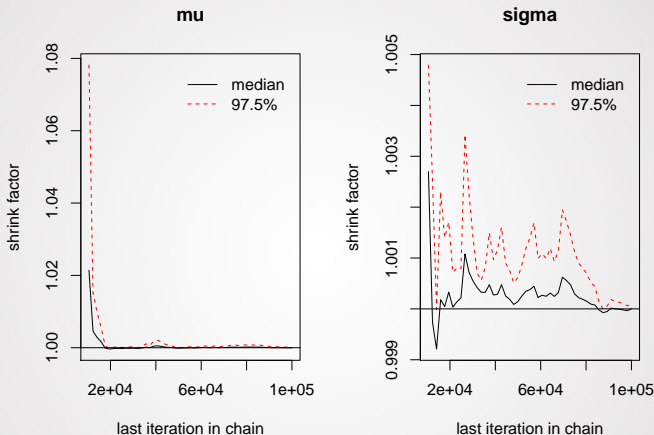


Figure 5: Gelman diagnostic for convergence. The closer to 1, the better the convergence. Rule of thumb: A chain has a reasonable convergence if it has a Potential Scale Reduction Factor (PSRF) below 1.15.



Example: MCMC (cont. 3)

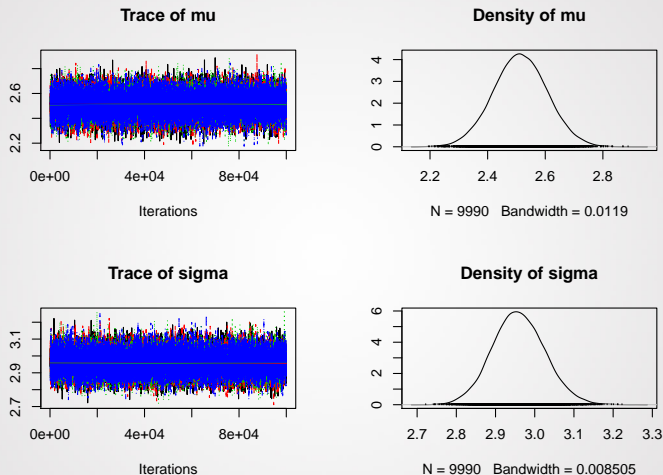


Figure 6: Posterior distribution

Agenda

On Genes and Trees

Model

Peeling algorithm

The `aphylo` R package

The `amcmc` R package

Bayesian Estimation of the parameters

Concluding Remarks

Putting all together

Let's start by reading some data

```
# Reading the data
path <- system.file("tree.tree", package="aphylo")
dat <- read_panther(path)
```

```
# The tree
dat$tree
```

```
##
## Phylogenetic tree with 145 tips and 107 internal nodes.
##
## Tip labels:
## AN5:MONBE|Gene=28576|UniProtKB=A9V8K6, AN7:SCHP0|PomBase=SPAC25B8.12c|UniProtKB=Q9UTA6
## Node labels:
## AN0, AN1, AN2, AN3, AN4, AN6, ...
##
## Rooted; includes branch lengths.
```

Putting all together (cont.)

```
# Extra annotations  
head(dat$internal_nodes_annotations)
```

##	branch_length	type	ancestor	duplication
## AN0	NA	S	LUCA	FALSE
## AN1	0.057	S	Archaea-Eukaryota	FALSE
## AN2	0.244	S	Eukaryota	FALSE
## AN3	0.436	S	Unikonts	FALSE
## AN4	0.417	S	Opisthokonts	FALSE
## AN6	0.684	D	<NA>	TRUE

Putting all together: MCMC of the model

In this example, using data from PANTHERDB, we will simulate a single function and use the `aphylo_mcmc` function for obtaining parameter estimates

```
tree <- dat$tree

# Simulating a function
set.seed(123)
atree <- sim_annotated_tree(
  tree= as_po_tree(tree),
  Pi = .05, mu = c(.1, .05), psi = c(.01, .02)
)

# Estimation
ans <- aphylo_mcmc(
  params = rep(.05, 5),
  dat     = atree,
  # Passing a Beta prior
  priors  = function(p) dbeta(p, 2, 20),
  # Parameters for the MCMC
  control = list(nchain=4, nbatch=1e4, thin=20, burnin=1e3)
)
```


Putting all together: MCMC of the model (cont. 1)

```
ans
```

```
##
## ESTIMATION OF ANNOTATED PHYLOGENETIC TREE
## ll: -57.0072,
## Method used: mcmc (10000 iterations)
## Leafs
## # of Functions 1
## # of 0: 99 (68%)
## # of 1: 46 (32%)
##
##           Estimate Std. Error
## psi[0]      0.0527      0.0289
## psi[1]      0.0502      0.0312
## mu[0]       0.0777      0.0226
## mu[1]       0.0397      0.0266
## Pi          0.0907      0.0620
```

How good is our prediction

```
# Looking at the posterior probabilities
head(predict(ans, what="leafs"))
```

```
##
## AN87:STAA8|EnsemblGenome=SAOUHSC_01375|UniProtKB=Q2FYRO      fun0000 0.06121
## AN88:DEIRA|EnsemblGenome=DR_2147|UniProtKB=Q9RSH7             0.06033
## AN219:LEPIN|EnsemblGenome=LB_007|UniProtKB=Q8EY50             0.06032
## AN223:CHLTR|EnsemblGenome=CT_103|UniProtKB=O84105            0.06032
## AN29:CHLRE|EnsemblGenome=CHLREDRAFT_196269|UniProtKB=A8HYJ4 0.10564
## AN65:PYRAE|EnsemblGenome=PAE3495|UniProtKB=Q8ZT04            0.10611
```

```
# And to the prediction score
prediction_score(ans)
```

```
## PREDICTION SCORE: ANNOTATED PHYLOGENETIC TREE
## Observed : 0.06 (146.89)
## Random   : 0.25 (591.44)
## Best      : 0.00 (0.00)
## Worse     : 1.00 (2365.77)
## -----
## Values between 0 and 1, 0 being best. Absolute scores in parenthesis.
```

How good is our prediction (cont. 1)

```
plot(prediction_score(ans), main=" ")
```

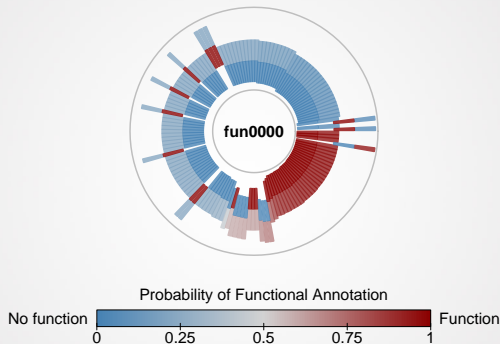


Figure 7: Predicted versus Observed values. Each slice of the pie represents a gene, the outer half of a slice is the predicted value, while the inner half is the observed value. Good predictions will coincide in color and show the slice closer to the center of the plot.

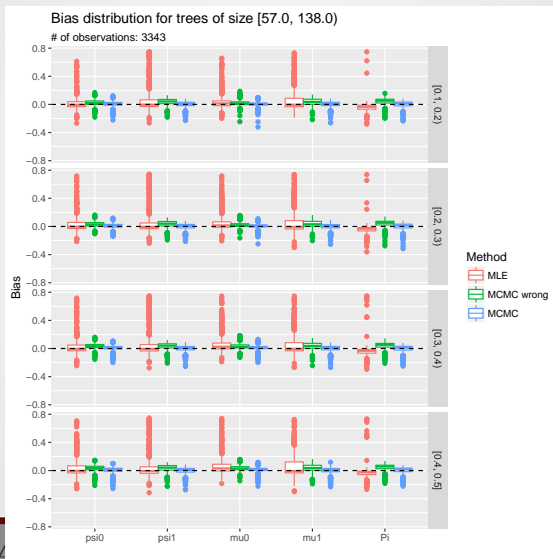
A simulation study

Setup

- ▶ Simulation study using ~13,000 families from the panther database
- ▶ For each one of the trees, we draw a set of the parameters $\{\psi_0, \psi_1, \mu_0, \mu_1, \pi\}$, and simulated annotations using our model's Data Generating Process, in particular, a Bate distribution with parameters 1/20.
- ▶ Once we had a fully annotated tree, we randomly removed annotations, with "0" annotations (no function) having a higher chance of been removed.
- ▶ With that data, we did parameter estimation and computed prediction scores using an MLE, MCMC with the right prior (using the same DGP), and a wrong prior (same Beta distribution, but with parameters 1/10 (twice the mean as the DGP)).

A simulation study

Bias



A simulation study

Convergence

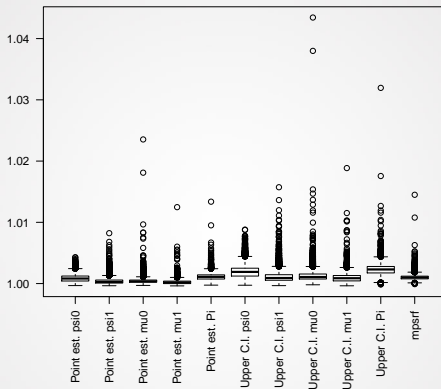


Figure 8: Gelman diagnostic for convergence. The closer to 1, the better the convergence. Rule of thumb: A chain has a reasonable convergence if it has a Potential Scale Reduction Factor (PSRF) below 1.15.

A simulation study

Prediction scores

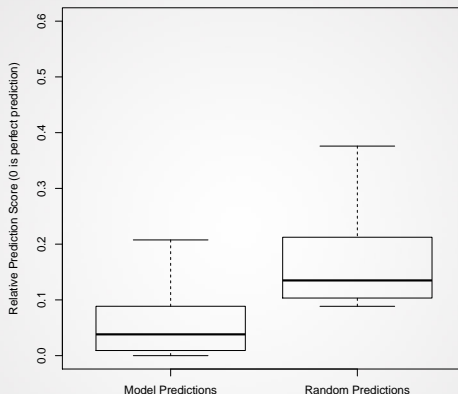


Figure 9: Distribution of prediction scores. The random prediction scores were computed analytically with parameter $p = 0.3$ (as resulting from the DGP).

Concluding Remarks

- ▶ A parsimonious model of gene functions: easy to apply on a large scale (we already ran some simulations using all 14,000 trees from the Panther DB).

Concluding Remarks

- ▶ A parsimonious model of gene functions: easy to apply on a large scale (we already ran some simulations using all 14,000 trees from the Panther DB).
- ▶ Already implemented, we are currently in the stage of writing the paper and setting up the simulation study.

Concluding Remarks

- ▶ A parsimonious model of gene functions: easy to apply on a large scale (we already ran some simulations using all 14,000 trees from the Panther DB).
- ▶ Already implemented, we are currently in the stage of writing the paper and setting up the simulation study.
- ▶ Unfortunately, since annotation data is so sparse, we lack of a significant number of useful datasets on which we can test our method.

Concluding Remarks

- ▶ A parsimonious model of gene functions: easy to apply on a large scale (we already ran some simulations using all 14,000 trees from the Panther DB).
- ▶ Already implemented, we are currently in the stage of writing the paper and setting up the simulation study.
- ▶ Unfortunately, since annotation data is so sparse, we lack of a significant number of useful datasets on which we can test our method.
- ▶ For the next steps, we are evaluating whether to include or how to include:

Concluding Remarks

- ▶ A parsimonious model of gene functions: easy to apply on a large scale (we already ran some simulations using all 14,000 trees from the Panther DB).
- ▶ Already implemented, we are currently in the stage of writing the paper and setting up the simulation study.
- ▶ Unfortunately, since annotation data is so sparse, we lack of a significant number of useful datasets on which we can test our method.
- ▶ For the next steps, we are evaluating whether to include or how to include:
 - ▶ Type of node: speciation, duplication, horizontal transfer.
 - ▶ Branch lengths
 - ▶ Correlation structure between functions
 - ▶ Using Taxon Constraints to improve predictions

Thank you!

Project 2: Augmenting functional information about human genes using probabilistic phylogenetic modeling

George G. Vega Yon
vegayon@usc.edu

Department of Preventive Medicine
University of Southern California

October 5, 2017

Formal definitions

[go back](#)

1. Phylogenetic tree: In our case, we talk about **partially ordered** phylogenetic tree, in particular, $\Lambda \equiv (N, E)$ is a tuple of nodes N , and edges

$$E \equiv \{(n, m) \in N \times N : n \mapsto m, n < m\}$$

2. Offspring of n : $O(n) \equiv \{m \in N : (n, m) \in E, n \in N\}$
3. Parent node of m : $r(m) \equiv \{n \in N : (n, m) \in E, m \in N\}$
4. Leaf nodes: $L(\Lambda) \equiv \{m \in N : O(m) = \{\emptyset\}\}$
5. Annotations: Given P functions, $Z \equiv \{z_n \in \{0, 1\}^P : n \in L(\Lambda)\}$
6. Annotated Phylogenetic Tree $D \equiv (\Lambda, Z)$
7. Observed Annotated Annotations $X = \{x_I\}_{I \in L(\Lambda)}$,
8. Experimentally Annotated Phylogenetic Tree $\tilde{D} \equiv (\Lambda, X)$

Leaf node probabilities

[go back](#)

- The probability of the leaf nodes having annotations z_I conditional on the observed annotation is

$$\Pr(Z_I = z_I \mid X_I = x_I) = \begin{cases} \psi & \text{if } x_I \neq z_I \\ 1 - \psi & \text{otherwise} \end{cases} \quad (1)$$

Where ψ can be either ψ_0 (mislabelling a zero), or ψ_1 (mislabelling a one).

Internal node probabilities

[go back](#)

- In the case of the internal nodes, the probability of a given state is defined in terms of the gain/loss probabilities

$$\Pr(Z_n = z_{lp} \mid Z_{r(n)} = z_{r(n)}) = \begin{cases} \mu & \text{if } z_n \neq z_{r(n)} \\ 1 - \mu & \text{otherwise} \end{cases}$$

Where μ can be either μ_0 (gain), or μ_1 (loss).

- Assuming independence across offspring, we can write

$$\Pr(Z_n = z_n \mid \tilde{D}) = \prod_{m \in O(n)} \sum_{z_m \in \{0,1\}} \Pr(Z_m = z_m \mid \tilde{D}) \Pr(Z_m = z_m \mid Z_n = z_n) \quad (2)$$

Notice that if m is a leaf node, then

$$\Pr(Z_m = z_m \mid \tilde{D}) = \Pr(X_m = x_m \mid Z_m = z_m).$$

Likelihood of the tree

[go back](#)

- Once the computation reaches the root node, $n = 0$, equations (1) and (2):

$$\Pr(Z_l = z_l \mid \tilde{D}) = \Pr(Z_l = z_{lp} \mid X_l = x_l) \quad (1)$$

$$\Pr(Z_n = z_n \mid \tilde{D}) = \prod_{m \in O(n)} \sum_{z_m \in \{0,1\}} \Pr(Z_m = z_m \mid \tilde{D}) \Pr(Z_m = z_{mp} \mid Z_n = z_n) \quad (2)$$

Allow us writing the likelihood of the entire tree

$$L(\psi, \mu, \pi \mid \tilde{D}) = \sum_{z_0 \in \{0,1\}} \Pr(Z_0 = z_0 \mid \pi) \Pr(Z_0 = z_0 \mid \tilde{D}) \quad (3)$$

Where $\Pr(Z_0 = z_0 \mid \pi) = \pi^{z_0} (1 - \pi)^{1-z_0}$