Rev: An interactive PPL for statistical phylogenetics

Fredrik Ronquist
Swedish Museum of Natural History, Stockholm, Sweden
Phylogenetics

- Modeling and inference involving evolutionary trees
- Widely used across the life sciences

Example applications:
- Virus transmission pathways
- Epidemiological models including genetic information
- Identification of pathogens
- Relationships among organisms
- Divergence time estimation
- Dynamics of molecular evolution
- Positive selection analysis
- Center of origin analysis
- Patterns of diversification and extinction
- Biogeography
Million years ago
Table 1. Most-cited articles over the period 2001-2013.

<table>
<thead>
<tr>
<th>Rank</th>
<th>Article cited</th>
<th>Times cited</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Sheldrick, G.M. (2008). A short history of SHELX. <em>Acta Crystallographica Section A</em>, 64, 112-122.</td>
<td>34,533</td>
</tr>
</tbody>
</table>
Statistical Phylogenetics

- Statistical approaches increasingly important:
  - Difficult problems requiring accurate and unbiased inference (e.g., structure of rapid radiations)
  - More aspects of the evolutionary process being examined (structural dependencies, biogeography etc)
  - Combination of background knowledge and sequence information (e.g., divergence time estimation)
- Modeling explosion, especially in the Bayesian context
- Challenging for empiricists to communicate and correctly understand models
- Challenging for developers of inference software
MrBayes: Bayesian Inference of Phylogeny

MrBayes is a program for Bayesian inference and model choice across a wide range of phylogenetic and evolutionary models. MrBayes uses Markov chain Monte Carlo (MCMC) methods to estimate the posterior distribution of model parameters.

Program features include:

- A common command-line interface across Macintosh, Windows, and UNIX operating systems;
- Extensive help available from the command line;
- Analysis of nucleotide, amino acid, restriction site, and morphological data;
- Mixing of data types, such as molecular and morphological characters, in a single analysis;
- Easy linking and unlinking of parameters across data partitions;
- An abundance of evolutionary models, including 4x4, doublet, and codon models for nucleotide data and many of the standard rate matrices for amino acid data;
- Estimation of positively selected sites in a fully hierarchical Bayesian framework;
- Full integration of the BEST algorithms for the multi-species coalescent.
- Support for complex combinations of positive, negative, and backbone constraints on topologies;
begin mrbayes;

execute data.nex;

outgroup Ibalia;
charset morphology = 1-166;
charset molecules = 167-3246;
charset COI = 167-1244;
charset EF1a = 1245-1611;
charset LWRh = 1612-2092;
charset 28S = 2093-3246;
partition favored= 5: morphology, COI, EF1a, LWRh, 28S;

set partition=favored;
lset applyto=(1) rates=gamma;
lset applyto=(2,3,4,5) rates=invgamma nst=mixed;
unlink revmat=(all) pinvar=(all) shape=(all) statefreq=(all);
prset ratepr=variable;

end;
Models supported by MrBayes 3 (simplified)

<table>
<thead>
<tr>
<th>Data Type</th>
<th>State Frequencies (Substitution Rates)</th>
<th>Across-Site Rate Variation</th>
<th>Coding Bias</th>
<th>Misc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Restriction 0 - 1</td>
<td>fixed/estimated (Dirichlet) prset statefreqpr</td>
<td>equal/gamma lset rates</td>
<td>all/variable/nopresencesites/noabsencesites lset coding</td>
<td>unordered/ordered ctype</td>
</tr>
<tr>
<td>Standard 0 - 9</td>
<td>equal/estimated (SymmDir) prset symdirihyperpr</td>
<td>equal/gamma lset rates</td>
<td>all/variable/informative lset coding</td>
<td>unordered/ordered ctype</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Model Type</th>
<th>State Frequencies</th>
<th>Substitution Rates</th>
<th>Across-Site Rate Variation</th>
<th>Across-Tree Rate Variation</th>
<th>Across-Site Omega Variation</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNA</td>
<td>4by4</td>
<td>fixed/est. (Dirichlet) prset statefreqpr</td>
<td>F81/HKY/GTR lset nst=1/2/6</td>
<td>equal/gamma/propinv/invgamma/adgamma lset rates</td>
<td>yes/no lset covarion</td>
<td></td>
</tr>
<tr>
<td></td>
<td>doublet</td>
<td>fixed/est. (Dirichlet) (over 16 states) prset statefreqpr</td>
<td>F81/HKY/GTR lset nst=1/2/6</td>
<td>equal/gamma/propinv/invgamma lset rates</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>codon</td>
<td>fixed/est. (Dirichlet) (over 61 states) prset statefreqpr</td>
<td>F81/HKY/GTR lset nst=1/2/6</td>
<td>equal/Ny98/M3 lset omegavari</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
CAN'T YOU DO ANYTHING RIGHT?!?
Probabilistic Graphical Models

Hierarchical Normal Model

\[ a \rightarrow \mu \leftarrow b \]
\[ c \rightarrow \sigma \]

\[ \mu \sim \text{Unif}(a, b) \]
\[ \sigma \sim \text{Exp}(c) \]
\[ x_i \sim \text{iid Norm}(\mu, \sigma) \]
Tree model

Birth-death model:
- Birth rate $\lambda$
- Death rate $\mu$
- Root time $t_{\text{mrca}} = t_{11}$

The birth-death model induces a probability distribution on
- Topology $\mathcal{T}$
- Speciation times $t$

Given a substitution rate $r$, branch lengths $b$ are given by
$$b_i = r(t_{a_i} - t_i)$$
Substitution model

Observation error usually ignored, that is, it is assumed that $y_i = s_i$ for all leaves $i$. 
DNA sequences are drawn iid from a discrete-state continuous-time Markov chain over four nucleotides: A, C, G, T

\[
Q = \begin{pmatrix}
- & \pi_c r_{AC} & \pi_g r_{AG} & \pi_t r_{AT} \\
\pi_a r_{AC} & - & \pi_g r_{CG} & \pi_t r_{CT} \\
\pi_a r_{AG} & \pi_c r_{CG} & - & \pi_t r_{GT} \\
\pi_a r_{AT} & \pi_c r_{CT} & \pi_g r_{GT} & -
\end{pmatrix}
\]

Instantaneous rate matrix for the General Time Reversible (GTR) substitution model

\[\pi\] Stationary state frequencies

\[r\] Exchangeability rates
\[ \alpha \rightarrow \pi \rightarrow Q \rightarrow r \rightarrow S_{i,j} \rightarrow v \]

\[ i = 1, \ldots, N \]
GTR Phylogeny Model
Tree Plate Representation

[Diagram of a tree plate representation showing nodes and edges labeled with mathematical symbols and conditions such as $s_{ij}$, $i \in \text{root}$, $i \in \text{internals}$, $i \in \text{tips}$, $j \in N$, $\Psi$, $\lambda$, $\epsilon$, $\pi$, $c$, $b$.]

Tree plate
Modular Representation

Pivot variable
RevBayes Project

- Interactive computing environment intended primarily for Bayesian phylogenetic inference
- Uses a probabilistic programming language (PPL), Rev, for constructing probabilistic phylogenetic and evolutionary graphical models interactively, step by step
- Rev is similar to the modeling languages of BUGS, JAGS and STAN
- RevBayes provides generic computing machinery for simulation, MCMC inference and Bayesian model testing
RevBayes

Bayesian phylogenetic inference using probabilistic graphical models and an interpreted language.

RevBayes was designed and developed by Sebastian Höhna, Fredrik Ronquist and John P. Huelsenbeck. The core development team additionally includes Michael J. Landis, Bastien Boussau, Tracy A. Heath and Nicolas Lartillot.

RevBayes is free software released under the GPL license v3. This site was generated using Jekyll and formatted using the Herring Cove theme.

http://www.revbayes.com
Basic properties of the Rev language

# There are three kinds of statements in the language creating model variables

# 1. Arrow assignment (value assignment, create constant nodes)

> a <- 4          # Give a the value 4
> b <- sqrt(a)    # Give b the value of sqrt(a), that is, 2
> b               # Print the value of b
2

# 2. Equation assignment (create deterministic nodes)

> c := sqrt(a)    # Make c a dynamic function node evaluating sqrt(a)
> c               # Print the value of c
2
> a <- 9          # Give a the value 9
> b               # Print the value of b
2
> c               # Print the value of c
3
Basic properties of the Rev language

# 3. Tilde assignment (create stochastic variables (nodes))

> a ~ dnExp( rate = x )                  # a is drawn from exp dist with rate = x
Basic properties of the Rev language

# --------------------------------
# Declaring and defining functions
# --------------------------------

> function foo ( x ) { x * x }
> foo( 2 )
    4

# If you wish, you can specify types as well

> function PosReal foo ( Real x ) { x * x }

# Functions can be used to define deterministic nodes

> a := foo( b )

# --------------------------------
# Declaring and defining new types
# --------------------------------

> class myclass : Move {
+     Real myTuningParam;
+     procedure Real move( Real x ) { myTuningParam * x }
+ }

# Inheritance, function overriding and overloading
A complete MCMC analysis in Rev

```r
a <- -1.0
b <- 1.0

mu ~ dnUnif(a, b)
sigma ~ dnExp(1.0)

for (i in 1:10) {
    x[i] ~ dnNorm(mu, sigma)
    x[i].clamp(0.5)
}

mymodel = model(mu)  # Any stochastic node in the model works

mymcmc = mcmc(mymodel)

mymcmc.run(1000)
```
Interactive environment

> a
  -1

> mu
  -0.003889918

> str(a)

  _variable   = a
  _dagType    = Constant DAG node
  _children   = [ mu ]

> str(mu)

  _variable   = mu
  _dagType    = Stochastic DAG node
  _clamped    = FALSE
  _lnProb     = -0.693147
  _parents    = [ a, b ]
  _children   = [ x[1], x[2], x[3], x[4], x[5], x[6], x[7], x[8], 
                  x[9], x[10] ]
# definition of the myGTR function (“Ziheng's favorite”)
function Model myGTR (CharacterMatrix data) {

    # describe Q matrix
    pi ~ dflatdir(4);
    r  ~ dflatdir(6);
    Q := gtr(pi, r);

    # describe tree
    tau ~ dtopuni(data.taxa(), rooted=false);

    # gamma shape
    alpha ~ dunif(0.0, 50.0);

    # discrete gamma mixture
    for (i in 1:4)
        catRate[i] := qgamma(i*0.25 - 0.125, alpha, alpha);
    for (i in 1:data.size())
        ratecat[i] ~ dcat(simplex(0.25,0.25,0.25,0.25));

    # associate distributions with tree parts
    for (i in 1:data.size()) {
        for (n in 1:tau.numNodes()) {
            if (tau.isTerminal(n)) {
                tau.length[n] ~ exp(1.0);
                tau.state[n] ~ ctmc(Q, e.length*catRate[ratecat[i]],
                    tau.state[tau.parent(n)]);
                tau.state[n] <- data[i][tau.tipIndex(n)];
            }
            else {
                tau.length[n] ~ exp(10.0);
                tau.state[n] ~ ctmc(Q, e.length*catRate[ratecat[i]],
                    tau.state[n]);
            }
        }
    }

    # return model
    return model(Q);
}
# Read in data
myData <- read( "data.nex" )

# Apply model
myModel = zihengGTR( myData )

# Construct mcmc
myMCMC = mcmc( myModel )

# Run mcmc
myMCMC.run(10000)
Canarian endemic radiations

Millipedes of the genus *Dolichoiodus* (Diplopoda, Julida, Julidae, Pachyulinae)

46 endemic species in the Canary Islands
Model

Organism groups

DNA data 1
GTR$_1$  $\mu_1$

DNA data 2
GTR$_2$  $\mu_2$

DNA data 3
GTR$_3$  $\mu_3$

$T_1$

$T_2$

$T_3$

distribution

IM

$\mu_i$ - mutation rate
$m_i$ - dispersal rate

Inference

Bayesian inference using MCMC sampling, accommodating uncertainty in all model parameters
Challenges

- No automated mechanism for generating message-passing for variable elimination of ancestral states (essential for MCMC performance) -> Delayed sampling
- No automated mechanisms for “looking into” and generating MCMC inference machinery for more generic constructs, like stochastic branching generating a phylogenetic tree
"Black-box" creation of tree model

# PhyloCTMC Model #

# The sequence evolution model
seq ~ dnPhyloCTMC(tree=psi, Q=Q, siteRates=sr, pInv=p_inv, type="DNA")

# Attach the data
seq.clamp(data)
Higher-order PPL for phylogenetics?

- Solves representation problem for tree topology as a random variable, and for models integrating tree-generating process and substitution process

- Challenges:
  - Static, compile-time variable elimination
  - Generating computationally efficient inference machinery

- Opportunities:
  - Generic simulation and inference machinery for a much wider class of models
  - New algorithms, like SMC, PMCMC etc

- Test case: inference under BAMM model (compound Poisson process birth-death model) using WebPPL, Anglican and Birch. No correct inference software for BAMM presented to date...