Count: evolutionary analysis of phylogenetic profiles with parsimony and likelihood

Miklós Csúros
Department of Computer Science and Operations Research, University of Montréal, Montréal, Québec, Canada

ABSTRACT
Summary: Count is a software package for the analysis of numerical profiles on a phylogeny. It is primarily designed to deal with profiles derived from the phyletic distribution of homologous gene families, but is suited to study any other integer-valued evolutionary characters. Count performs ancestral reconstruction, and infers family- and lineage-specific characteristics along the evolutionary tree. It implements popular methods employed in gene content analysis such as Dollo and Wagner parsimony, propensity for gene loss, as well as probabilistic methods involving a phylogenetic birth-and-death model.

Availability: Count is available as a stand-alone Java application, as well as an application bundle for MacOS X, at the web site http://www.iro.umontreal.ca/~csuros/gene_content/count.html. It can also be launched using Java Webstart from the same site. The software is distributed under a BSD-style license. Source code is available upon request from the author.

Received on April 12, 2010; revised on May 27, 2010; accepted on June 8, 2010

1 INTRODUCTION
Some aspects of genome evolution are best captured by integer quantities. Given a phylogeny with terminal taxa $A$, such a quantity forms a numerical profile, which extends the so-called phylogenetic profile of presence–absence (Koonin and Galperin, 2002; Pellegrini et al., 1999) $\Phi: A \mapsto \{0, 1, 2, \ldots \}$. In a typical application, $\Phi(x)$ denotes the number of genes in genome $x \in X$ for a certain homolog gene family: a homolog family comprises all descendants of the same ancestral gene (Fitch, 2000) in evolutionary lineages. Such families are routinely identified by pairwise sequence comparisons, coupled with the clustering of postulated homolog pairs (Alexeyenko et al., 2006; Tatusov et al., 1997). In other interesting examples, $\Phi(x)$ might be the size (Caetano-Anollés, 2005) of genome $x$ or a sequence length polymorphism in population $x$ (Witmer et al., 2003).

Given a phylogeny, an evolutionary character’s history can be inferred by various means in order to reconstruct its state at ancestral nodes or to estimate the tempo of evolution (Pagel, 1999). The Count software package provides a convenient graphical user interface to nodes or to estimate the tempo of evolution (Pagel, 1999). The Count inferred by various means in order to reconstruct its state at ancestral

2 FEATURES
Count is designed primarily to work with a dataset of numerical profiles for homolog gene families. It allows for combining multiple profiles with various annotations, as found in databases of cluster families such as COG (Tatusov et al., 1997). Profiles can be filtered by criteria based on presence, membership count and annotations, in order to compile winnowed datasets for further analysis.

Given an evolutionary tree $T$, Count computes the states $\{0\}$ at tree nodes $u \in T$, based on each profile $\Phi$ by imposing $\Phi[u] = \xi[u]$ for all terminal taxa $u$. In parsimony approaches, the ancestral reconstruction minimizes a criterion based on the implied state changes $\xi[u] - \xi[v]$ over the edges $uv$. Alternatively, Count works with so-called phylogenetic birth-and-death models that consider $\xi[u]: u \in T$ as a random variable with a well-defined distribution.

2.1 Parsimony
Count implements Dollo parsimony (Farris, 1977) and Wagner parsimony (Farris, 1970). In case of the latter, it also implements an asymmetric version (Csúros, 2008) that penalizes losses and gains differently. Count also computes Propensity for Gene Loss (Krylov et al., 2003), which quantifies the frequency of loss for each family using Dollo parsimony.

2.2 Phylogenetic birth-and-death models
The probabilistic model employed in Count relies on linear birth-death-immigration processes (Kendall, 1949), commonly used to model population growth and queuing systems. In the general phylogenetic birth-and-death model, three rates are assigned to each branch: gene loss rate $\mu$, gene duplication rate $\lambda$ and a gain rate $\kappa$. ‘Gain’ covers multiple phenomena without specifying the origin of the gain, including de novo gene formation and lateral gene transfer.

Specifically, character evolution on each edge $uv$ with length $t$ is stochastically determined by a continuous time Markov process $X$ with $X(0) = \xi[u]$ and $X(t) = \xi[v]$. The process is characterized by the gain rate $\kappa$, loss rate $\mu$ and duplication rate $\lambda$, for $0 \leq n, 0 < t \leq t$ and any $0 < \kappa$:

$$P\{X(t + \delta) = n | X(t) = n - 1\} = \delta(n - 1)\kappa + o(1)$$

$$P\{X(t + \delta) = n - 1 | X(t) = n\} = \delta(n\mu + o(1))$$

Less general models may forbid gain ($\kappa = 0$), or duplication ($\lambda = 0$), or even both. Paralogs evolve independently in this model, capturing the birth-and-death evolution of multigene families (Nei and Rooney, 2005), as opposed to concerted evolution, or events...
2.3 User interaction

Figure 1 illustrates the rich graphical user interface of Count. The program can work with multiple datasets and models at the same time, in order to help comparisons between different analyses. Entire work sessions can be saved, and individual analysis results can be exported into tab-delimited text files, in order to use with other programs such as spreadsheet tools. Main software components (rate optimization and ancestral reconstruction) can also be launched from the command line without invoking the graphical interface.

2.4 Implementation

Count is written entirely in Java (Java SE 6), and was tested on various computer platforms, including Microsoft Windows, MacOS X and Linux. In addition, Count is also available as an integrated application bundle on MacOS X and a Java Webstart application. The software is distributed with test data and a detailed User’s Guide.

ACKNOWLEDGEMENTS

I am grateful for valuable feedback on the software from Aaron Darling, Dannie Durand, Maureen Stoltzer, Gergely Szöllösi, Natalya Yutin and Yuri Wolf.

Funding: Natural Sciences and Engineering Research Council of Canada grant.

Conflict of Interest: none declared.

REFERENCES
