doi:10.1093/nar/gkn283

xREI: a phylo-grammar visualization webserver

Lars Barquist* and Ian Holmes

Department of Bioengineering, University of California, Berkeley

Received February 9, 2008; Revised April 16, 2008; Accepted April 26, 2008

ABSTRACT

Phylo-grammars, probabilistic models combining Markov chain substitution models with stochastic grammars, are powerful models for annotating structured features in multiple sequence alignments and analyzing the evolution of those features. In the past, these methods have been cumbersome to implement and modify. `xrate` provides means for the rapid development of phylo-grammars (using a simple file format) and automated parameterization of those grammars from training data (via the Expectation Maximization algorithm). `xREI` (pron. '*X-ray*') is an intuitive, flexible AJAX (Asynchronous Javascript And XML) web interface to `xrate` providing grammar visualization tools as well as access to `xrate`'s training and annotation functionality. It is hoped that this application will serve as a valuable tool to those developing phylo-grammars, and as a means for the exploration and dissemination of such models. `xREI` is available at http://harmony.biowiki.org/xrei/

INTRODUCTION

Accurate automated annotation of biological sequences is an increasingly important problem in the biological sciences. Recent releases of high-quality multiple sequence alignment data, such as by the Drosophila 12 Genomes Consortium (1,2) have only underscored this fact. Phylo-grammars have had great success in this arena, with diverse applications in areas such as the prediction of exons in DNA (3,4), prediction of secondary structure in proteins (5,6) and detection of noncoding RNA (7). However, despite their broad range of application, implementations of phylo-grammars have often been limited to a single model and have lacked fast and accurate training algorithms, limiting more widespread adoption.

`xrate` provides exactly this missing functionality (8). By allowing the nonexpert user to quickly and effectively implement, train and utilize phylo-grammars, it has opened this powerful method of analysis to researchers who would otherwise lack the necessary expertise and/or resources to implement such a tool. In this article we introduce `xREI` (pronounced '*X-ray*'), a web interface which allows researchers to explore a range of pre-defined phylo-grammars, as well as providing tools to visualize their own.

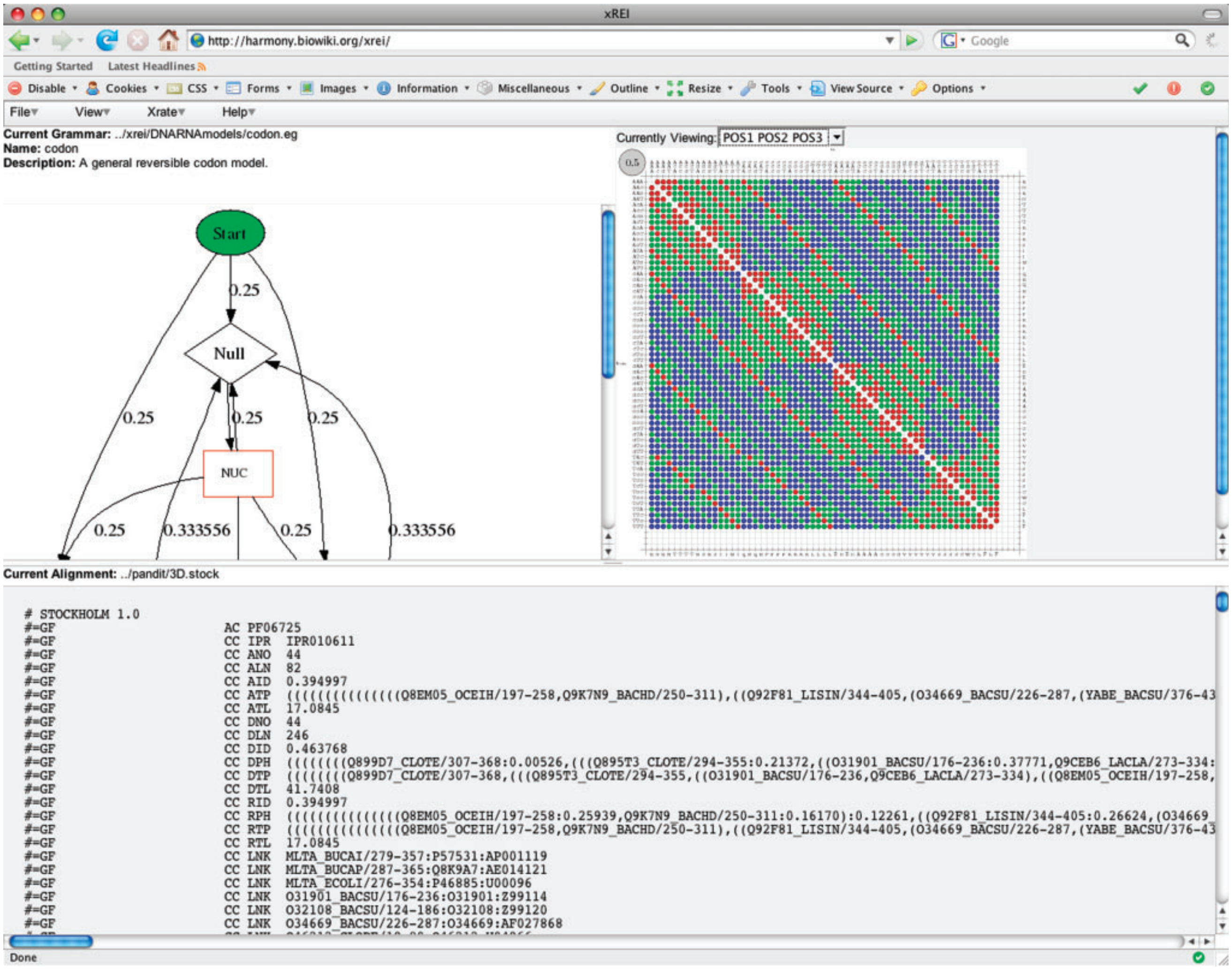
THE `XRATE` PROGRAM

`xrate` is an extremely flexible software tool for modeling structural and phylogenetic variation in multiple sequence alignments. Users can design models for point substitution of nucleotides or amino acids, or for coordinated substitution among groups of residues (e.g. codons or RNA basepairs). The models can be parametric, fully unconstrained or lineage-specific (i.e. using different parameters on different branches of a tree). These substitution models can then be organized via a hidden Markov model or stochastic context-free grammar, allowing for structured variation due to localized rate heterogeneity, intron/exon structure, RNA secondary structure, mixture models, binding sites, etc. The software can be used to fit maximum-likelihood parameters from training alignments (annotated or unannotated), or to use previously-fit parameters to estimate phylogenetic trees or annotate alignments. The entire model is specified using a compact file format described at `biowiki.org/XrateFormat`.

`xrate`'s ability to specify arbitrary substitution rate matrices is similar to HyPhy (9), while its grammar-design functionality is most similar to HMMoC (10). `xrate` can reproduce the core functionality of a wide range of programs: molecular evolutionary measurement [PAML: (11)], annotation of regions where substitutions are suppressed [PhastCons: (12)] or recently accelerated [DLESS: (13)], protein-coding genefinding [Exoniphy: (14); EvoGene: (3)], RNA folding [PFOLD: (15)] and gene discovery [EvoFold: (7)] and prediction of regulatory elements [RNA-DECODER: (16); MONKEY: (17)].

Building a web interface to such a generic tool is a challenge. We realized that we would need to combine various components: a grammar browser to show the model structure, a rate matrix browser to show relative substitution rates and an alignment browser for training and annotation. Processing would have to be split between client and server: the platform-independent web browser would handle layout and user interface components, with a Linux backend doing the heavy lifting. These

*To whom correspondence should be addressed. Email: lbarquist@gmail.com



Current Grammar: ../xrei/DNARNAmodels/codon.eg
Name: codon
Description: A general reversible codon model.

Currently Viewing: POS1 POS2 POS3

Current Alignment: ../pandit/3D.stock

```
# STOCKHOLM 1.0
#=GF                   AC PF06725
#=GF                   CC IPR  IPR010611
#=GF                   CC ANO  44
#=GF                   CC ALN  82
#=GF                   CC AID  0.394997
#=GF                   CC ATP  (((((((((((((((Q8EM05_OCEIH/197-258,Q9K7N9_BACHD/250-311),((Q92F81_LISIN/344-405,(O34669_BACSU/226-287,(YABE_BACSU/376-43
#=GF                   CC ATL  17.0845
#=GF                   CC DNO  44
#=GF                   CC DLN  246
#=GF                   CC DID  0.463768
#=GF                   CC DPH  ((((((((Q899D7_CLOTE/307-368:0.00526,(((Q895T3_CLOTE/294-355:0.21372,((O31901_BACSU/176-236:0.37771,Q9CEB6_LACLA/273-334:
#=GF                   CC DTP  ((((((((Q899D7_CLOTE/307-368,(((Q895T3_CLOTE/294-355,((O31901_BACSU/176-236,Q9CEB6_LACLA/273-334),((Q8EM05_OCEIH/197-258,
#=GF                   CC DTL  41.7408
#=GF                   CC RID  0.394997
#=GF                   CC RPH  (((((((((((((((Q8EM05_OCEIH/197-258:0.25939,Q9K7N9_BACHD/250-311:0.16170):0.12261,((Q92F81_LISIN/344-405:0.26624,(O34669_
#=GF                   CC RTP  (((((((((((((((Q8EM05_OCEIH/197-258,Q9K7N9_BACHD/250-311),((Q92F81_LISIN/344-405,(O34669_BACSU/226-287,(YABE_BACSU/376-43
#=GF                   CC RTL  17.0845
#=GF                   CC LNK  MLTA_BUCAI/279-357:P57531:AP001119
#=GF                   CC LNK  MLTA_BUCAP/287-365:Q8K9A7:AE014121
#=GF                   CC LNK  MLTA_ECOLI/276-354:P46885:U00096
#=GF                   CC LNK  O31901_BACSU/176-236:O31901:Z99114
#=GF                   CC LNK  O32108_BACSU/124-186:O32108:Z99120
#=GF                   CC LNK  O34669_BACSU/226-287:O34669:AF027868
```

Figure 1. The xREI Interface.

considerations strongly pointed to the AJAX model (Asynchronous Javascript And XML) of client-server communication.

THE xREI WEBSERVER

The xREI webserver handles two file formats, xrate grammars and stockholm alignments with defined phylogenetic trees. The server makes available an arbitrary number of repositories of these files, which can be browsed and selected by the end user based on a preview window. Users are also able to upload their own files in the appropriate formats from their local machines. Once loaded, a state diagram for the grammar is automatically generated, and options for generating rate matrix displays and accessing xrate functionalities are presented.

State transition diagrams

State transition diagrams are generated from the transformation rules within the xrate grammar file using GraphViz. If a start state is not explicitly defined, one is drawn proceeding the first state in the first transformation rule. An end state is drawn, and all transformations to an empty state are treated as transitioning to it. Each emission state is shown as transitioning to its substitution chain. Bifurcations are drawn as rectangles, with dotted and dashed lines representing the right and left transformations.

Diagrams can be rendered in one of the two ways. The dot rendering produces a 'layered' graph avoiding edge crossings and minimizing edge length. The neato produces a 'ball and spring' graph with edge weights representative of their transition probability. The neato rendering is also randomly seeded, so that multiple renderings will produce different results. Both produce output in PNG format for viewing in-browser, as well as PostScript suitable for publication which may be exported and saved locally. The raw GraphViz files are also available to be exported and hand modified for clarity, as may be required particularly in larger grammars.

Rate matrix visualization

The rate matrices of individual substitution chains are displayed as 'bubble plots.' A grid is drawn with the

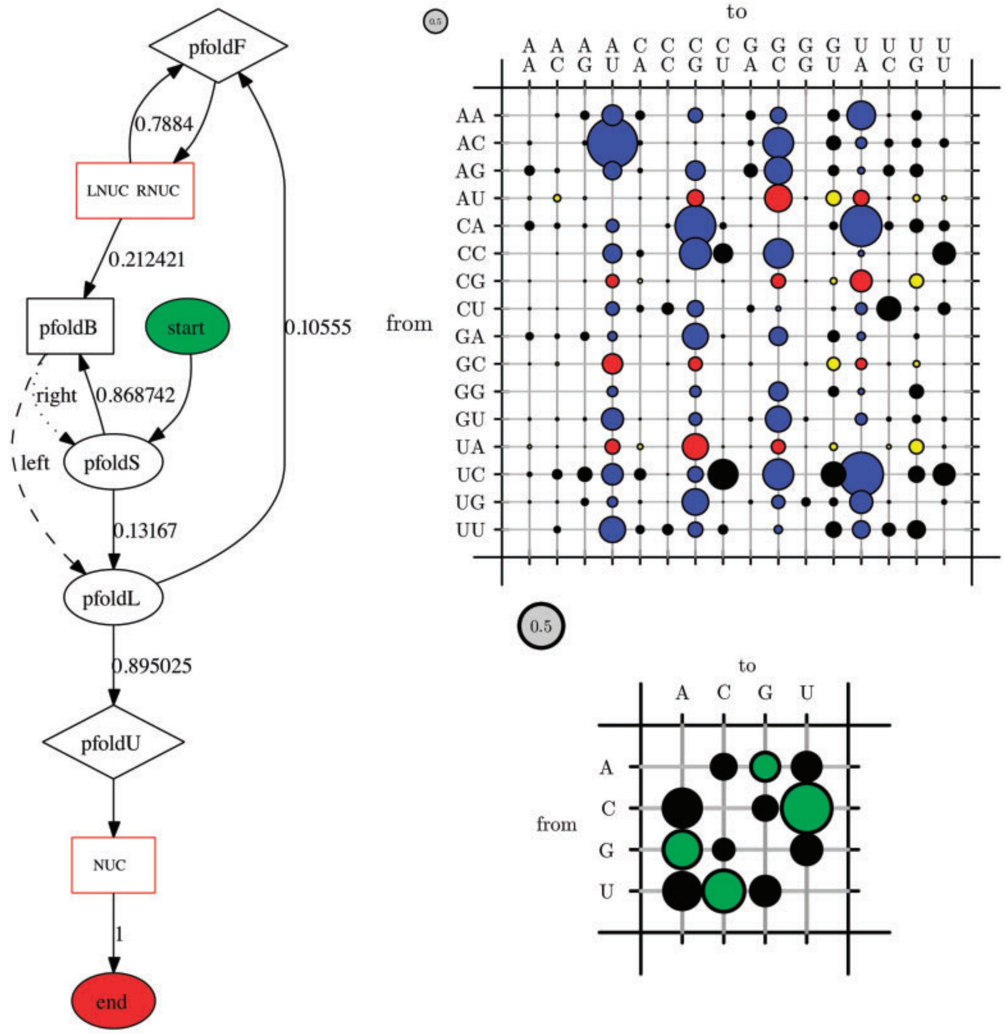


Figure 2. Representative xREI output for a pfold-derived phylo-grammar. On the left is the dot-rendered state transition diagram. Null states are represented by ovals, bifurcations by rectangles, emission states by diamonds and emission chains by red rectangles. Transition probabilities are shown where applicable. At the top right is the rate matrix for the LNUC RNUC dinucleotide chain colored to highlight mutations to/from canonical pairs, and bottom right is the rate matrix for the NUC nucleotide chain colored to highlight transversions/transitions.

grammar alphabet as the axes. At each vertex, a circle is drawn proportional to the substitution rate between the respective residues. These circles are initially scaled by an arbitrary function which generally produces good results. This scale factor is available to the user to modify manually.

xREI automatically detects the grammar alphabet (DNA, RNA, amino acids, codons) and presents a range of appropriate coloring options to the user. These include coloring by transition/transversion, the number of nucleotide differences, preservation of canonical pairing and synonymous/nonsynonymous codons. Output is made available in PNG and PostScript formats.

xrate Functions

The XRate menu item provides access to xrate's grammar training and annotation functions. The 'Use Grammar to Annotate Alignment' option will run the currently loaded grammar over the alignment, loading a new alignment into memory containing the xrate annotations, predicted secondary structure for instance, in a new labeled # GC line. Similarly, the 'Use Alignment to Train Grammar' option will load a new grammar into browser memory containing updated transformation and substitution parameters. We also provide an option to use xrate's neighbor-joining functionality to generate a Newick tree for a given alignment.

While there is no hard upper limit to the size of alignments submitted to xREI for training or annotation, large alignments should be avoided. Due to the dynamic nature of the server, a large alignment could potentially cause the server to timeout producing an error message, or leave the user waiting for a response indefinitely. As such, intensive training and annotation (e.g. whole genome analysis, training over stockholm alignment databases) should still be performed using xrate at the command-line. We have recently introduced an option to run annotations and training as a background process on the server as part of the 'Advanced Options' menu, with results emailed to the end user, though this feature may need to be suspended in the event of high server traffic.

Available repositories

As part of the initial xREI package we have included a set of xrate phylo-grammar implementations of a variety of models. Grammars for DNA and RNA include: Jukes-Cantor (18), Kimura (19), HKY85 (20), the general reversible model 'REV', (21) the general irreversible model 'IRREV', the general irreversible dinucleotide model (4), a general reversible codon model (22), a phylo-HMM for detecting local rate variation (23,12), a phylo-SCFG for RNA folding (15). Grammars for proteins include: a general reversible amino acid substitution model (24) and several phylo-HMMs for protein secondary structure analysis (5).

We have also included several alignment databases over which both these and user-supplied grammars may be trained and run. RFAM (25) is a collection of non-coding RNA multiple alignments. PANDIT (26) contains codon multiple sequence alignments covering many common protein-coding domains. TreeFam (27) is a database of protein alignments along with curated and semi-curated trees.

IMPLEMENTATION

The xREI interface is written in javascript with the Dojo Toolkit. The Dojo Toolkit provides a range of cross-browser functions for creation of interactive interface elements as well as AJAX client-server communication. xREI relies heavily on AJAX communication to provide a seamless user experience without page refreshes, more similar to a desktop application than a traditional web server.

xREI uses a set of server-side scripts written in perl to process data. All rely heavily on the freely available DART perl libraries (http://dart.sourceforge.net), which provides a collection of tools for manipulating xrate format grammars and stockholm alignments, among other related functions. xrei_load.pl provides basic utilities such as producing preview screens and formatting grammars and alignments to be loaded client-side. xrei_xrate.pl uses the DART perl libraries to perform xrate operations such as training and annotation. xrei_statediag.pl uses the perl GraphViz module to produce state diagrams. xrei_vizrates.pl is a modified version of the visualizeRates.pl script included with DART which relies on LaTeX to produce its 'bubble plot' graphs. In the interest of privacy, no temporary files are stored on the server save postscript output files, which are deleted automatically every night.

Code reusability and flexibility were both major goals in implementation. As such any functionality currently in xREI can be easily incorporated in other web applications. All server-side scripts are written using CGI.pm, and are capable of being used in either an AJAX or standard CGI environment. Each is capable of running independently of the others or the xREI interface. Conversely, the xREI javascript interface has been designed for easy extension and compatibility with other web applications. Adding new functionality to the server requires only trivial modifications to the javascript code. In addition, the xREI server can pass grammar or alignment data to other web applications, as is currently done with the Raton RNA alignment viewer, with the caveat that both applications must be running on the same domain due to javascript security limitations.

CONCLUSIONS

Phylo-grammars have a broad range of applications in the biological sciences. xREI provides a flexible and intuitive method for visualizing phylo-grammars developed within the xrate framework. Given an xrate grammar and an alignment in stockholm format, xREI can produce state transition graphs and substitution rate matrices for the grammar, as well as being capable of training and alignment annotation functions. xREI can be easily expanded to incorporate new visualization or computational methods as needed. In addition, xREI functionalities can be easily incorporated into other web servers. It is hoped that by increasing availability and easing use of these tools, xREI will accelerate their adoption as a standard analytical method.

ACKNOWLEDGEMENTS

The authors were funded by NIH/NHGRI grant 1R01GM076705-01 and by the 2007 Google Summer of Code [National Evolutionary Synthesis Center (NESCent) group, NSF #EF-0423641].

Conflict of interest statement. None declared.

REFERENCES

1. Clark,A.G., Eisen,M.B., Smith,D.R., Bergman,C.M., Oliver,B., Markow,T.A., Kaufman,T.C., Kellis,M., Gelbart,W., Iyer,V.N. *et al*. (2007) Evolution of genes and genomes on the Drosophila phylogeny. *Nature*, **450**, 203–218.
2. Stark,A., Lin,M.F., Kheradpour,P., Pedersen,J.S., Parts,L., Carlson,J.W., Crosby,M.A., Rasmussen,M.D., Roy,S., Deoras,A.N. *et al*. (2007) Discovery of functional elements in 12 Drosophila genomes using evolutionary signatures. *Nature*, **450**, 219–232.
3. Pedersen,J.S. and Hein,J. (2003) Gene finding with a hidden Markov model of genome structure and evolution. *Bioinformatics*, **19**, 219–227.
4. Siepel,A. and Haussler,D. (2004) Phylogenetic estimation of context-dependent substitution rates by maximum likelihood. *Mol. Biol. Evol*., **21**, 468–488.
5. Thorne,J.L., Goldman,N. and Jones,D.T. (1996) Combining protein evolution and secondary structure. *Mol. Biol. Evol*., **13**, 666–673.
6. Goldman,N., Thorne,J.L. and Jones,D.T. (1996) Using evolutionary trees in protein secondary structure prediction and other comparative sequence analyses. *J. Mol. Biol*., **263**, 196–208.
7. Pedersen,J.S., Bejerano,G., Siepel,A., Rosenbloom,K., Lindblad-Toh,K., Lander,E.S., Kent,J., Miller,W. and Haussler,D. (2006) Identification and classification of conserved RNA secondary structures in the human genome. *PLoS Comput. Biol*., **2**, e33.
8. Klosterman,P.S., Uzilov,A.V., Bendana,Y.R., Bradley,R.K., Chao,S., Kosiol,C., Goldman,N. and Holmes,I. (2006) XRate: a fast prototyping, training and annotation tool for phylo-grammars. *BMC Bioinformatics*, **7**, 428.

9. Pond,S.L.K., Frost,S.D.W. and Muse,S.V. (2005) HyPhy: hypothesis testing using phylogenies. *Bioinformatics*, **21**, 676–679.
10. Lunter,G. (2007) HMMoC–a compiler for hidden Markov models. *Bioinformatics*, **23**, 2485–2487.
11. Yang,Z. (2007) PAML 4: phylogenetic analysis by maximum likelihood. *Mol. Biol. Evol.*, **24**, 1586–1591.
12. Siepel,A., Bejerano,G., Pedersen,J.S., Hinrichs,A.S., Hou,M., Rosenbloom,K., Clawson,H., Spieth,J., Hillier,L.W., Richards,S. *et al.* (2005) Evolutionarily conserved elements in vertebrate, insect, worm, and yeast genomes. *Genome Res.*, **15**, 1034–1050.
13. Pollard,K.S., Salama,S.R., Lambert,N., Lambot,M.A., Coppens,S., Pedersen,J.S., Katzman,S., King,B., Onodera,C., Siepel,A. *et al.* (2006) An RNA gene expressed during cortical development evolved rapidly in humans. *Nature*, **443**, 167–172.
14. Siepel,A. and Haussler,D. (2004) Combining phylogenetic and hidden Markov models in biosequence analysis. *J. Comput. Biol.*, **11**, 413–428.
15. Knudsen,B. and Hein,J. (2003) Pfold: RNA secondary structure prediction using stochastic context-free grammars. *Nucleic Acids Res.*, **31**, 3423–3428. Evaluation studies.
16. Pedersen,J.S., Meyer,I.M., Forsberg,R., Simmonds,P. and Hein,J. (2004) A comparative method for finding and folding RNA secondary structures within protein-coding regions. *Nucleic Acids Res.*, **32**, 4925–4923.
17. Moses,A.M., Chiang,D.Y., Pollard,D.A., Iyer,V.N. and Eisen,M.B. (2004) MONKEY: identifying conserved transcription-factor binding sites in multiple alignments using a binding site-specific evolutionary model. *Genome Biol.*, **5**, R98.
18. Jukes,T.H. and Cantor,C. (1969) Evolution of protein molecules. In Munro,H.N. (ed.), *Mammalian Protein Metabolism*. Academic Press, New York, pp. 21–132.
19. Kimura,M. (1980) A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. *J. Mol. Evol.*, **16**, 111–120.
20. Hasegawa,M., Kishino,H. and Yano,T. (1985) Dating the human-ape splitting by a molecular clock of mitochondrial DNA. *J. Mol. Evol.*, **22**, 160–174.
21. Yang,Z. (1994) Estimating the pattern of nucleotide substitution. *J. Mol. Evol.*, **39**, 105–111.
22. Kosiol,C., Holmes,I. and Goldman,N. (2007) An empirical codon model for protein sequence evolution. *Mol. Biol. Evol.*, **24**, 1464–1479
23. Felsenstein,J. and Churchill,G.A. (1996) A hidden Markov model approach to variation among sites in rate of evolution. *Mol. Biol. Evol.*, **13**, 93–104.
24. Dayhoff,M.O., Eck,R.V. and Park,C.M. (1972) A model of evolutionary change in proteins. In Dayhoff,M.O. (ed.), *Atlas of Protein Sequence and Structure*, Vol. 5. National Biomedical Research Foundation, Washington, DC, pp. 89–99.
25. Griffiths-Jones,S., Moxon,S., Marshall,M., Khanna,A., Eddy,S.R. and Bateman,A. (2005) Rfam: annotating non-coding RNAs in complete genomes. *Nucleic Acids Res.*, **33**, D121–D124.
26. Whelan,S., de Bakker,P.I. and Goldman,N. (2003) Pandit: a database of protein and associated nucleotide domains with inferred trees. *Bioinformatics*, **19**, 1556–1563.
27. Li,H., Coghlan,A., Ruan,J., Coin,L.J., Hrich,J.K., Osmotherly,L., Li,R., Liu,T., Zhang,Z., Bolund,L. *et al.* (2006) Treefam: a curated database of phylogenetic trees of animal gene families. *Nucleic Acids Res.*, **34**, D572–D580.