Supplementary Information for:

Complexity Reduction in Context-dependent DNA Substitution Models

William H. Majoros, Uwe Ohler

Glossary

Rate variability – when substitution rates differ between sites in an alignment

Gamma model (+Γ) – one type of rate variability model in which the rates at sites in an alignment are assumed to follow a gamma distribution

Correlated-rates model – any rate variability model in which rates at different sites may vary, but are assumed to be in some way correlated; they may or may not also be gamma-distributed

Context dependence – a form of rate variability in which the substitution rate at a site is affected by the identities of nearby nucleotides (i.e., the *local context*); other factors may also affect the rates: for example, sites with *different* contexts may have correlated rates, and sites with the *same* context may vary according to a gamma model, etc.

n^{th}-order model – a context-dependent model in which contexts (not including the current column) are of length n

Joint model (also: **n-mer model**) – a model represented by a single $4^{n+1}\times4^{n+1}$ rate matrix; each cell gives the probability of one (n+1)-mer substituting for another (n+1)-mer (i.e., a single target column plus n columns of context)

Conditional model – a model represented by a collection of 4^n rate matrices each of size 4×4; each cell gives the probability of one nucleotide substituting for another nucleotide, conditional on the given n nucleotides of context

Independence within *n*-mers (+INDEP) – the conditional independence assumption which is invoked when a joint model assigns a rate of 0 to any pair of (n+1)-mers differing in more than one position

Dual contexts – when context is taken from two taxa (typically the two taxa involved in a substitution event)

Single contexts – when context is taken from only one taxon

Observable contexts – when contexts are only taken from observed taxa present in an alignment

Unobservable contexts – when contexts are taken from ancestral taxa not present in an alignment

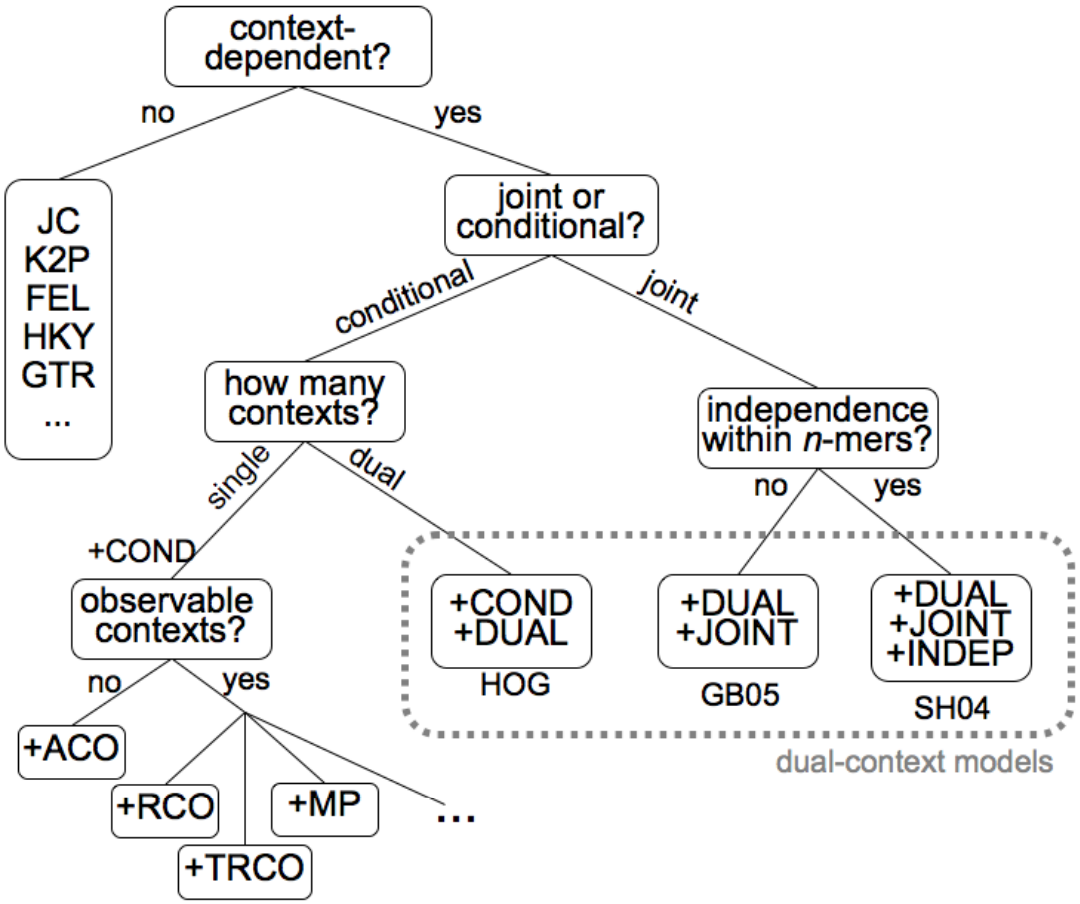


FIG S1—*A taxonomy of context-dependent substitution models. DUAL=uses dual contexts; JOINT=uses a joint substitution matrix on n-mers; COND=uses a collection of conditional matices indexed by context; INDEP=assumes conditional independence within n-mers; ACO=ancestral contexts only; RCO=root contexts only; TRCO=transitive RCO; MP=maximum parsimony; JC=Jukes and Cantor's 1-parameter model (1969); K2P=Kimura's 2-parameter model (1980); FEL=Felsenstein's 1-parameter model with equilibrium frequencies (1981); HKY=Hasegawa-Kishino-Yano's 2-parameter model with equilibrium frequencies (1985); GTR=Tavaré's general time-reversible model (1986).*

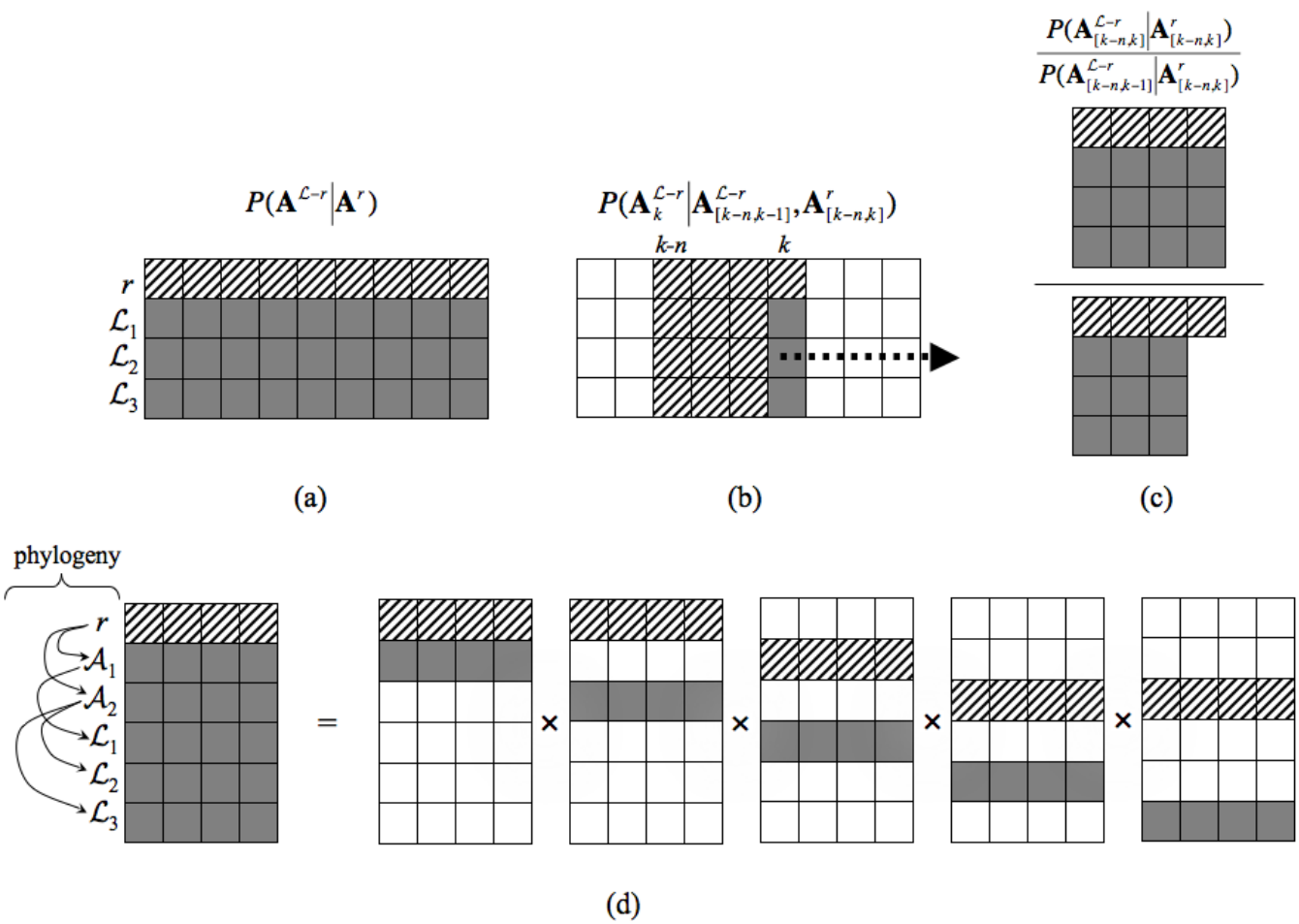


FIG S2—*Probability decomposition via "windowing". An alignment is shown as a grid with taxa as rows; shaded cells are conditioned on hatched cells. The desired term* $P(A^{L-r}|A^{r})$ *shown in (a) is decomposed into a product of column probabilities (b) under a Markov assumption (Eq. 14). (c) Each column probability is computed by dividing the probability of an* n^{th}*-order window by that of its* $(n-1)^{th}$*-order prefix (Eq. 12). (d) Each window is decomposed according to the phylogeny into the probability of a descendant n-mer conditional on an ancestral n-mer (Eq. 11).*

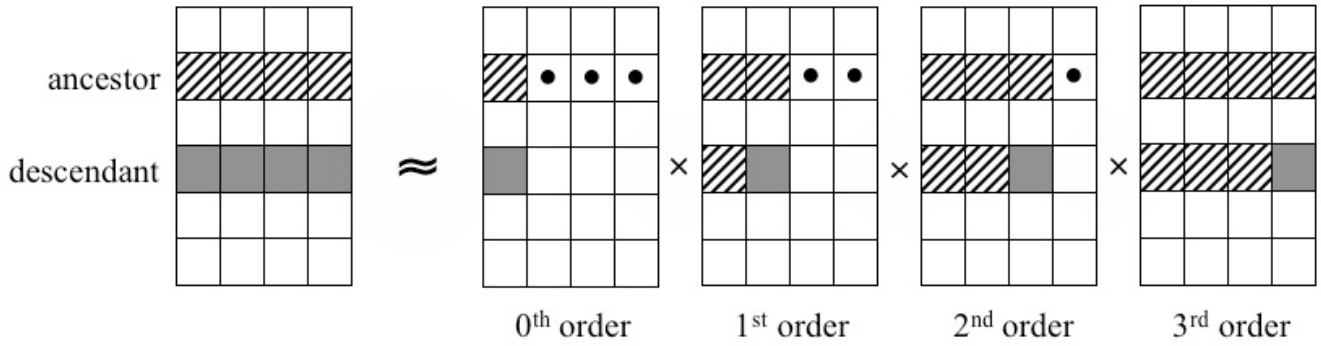


FIG S3—*Decomposing a joint (n+1)-mer substitution into multiple conditional substitutions of different orders, utilizing a conditional independence assumption in the ancestor. Shaded cells are dependent on hatched cells, and are assumed conditionally independent of all cells containing a •.*

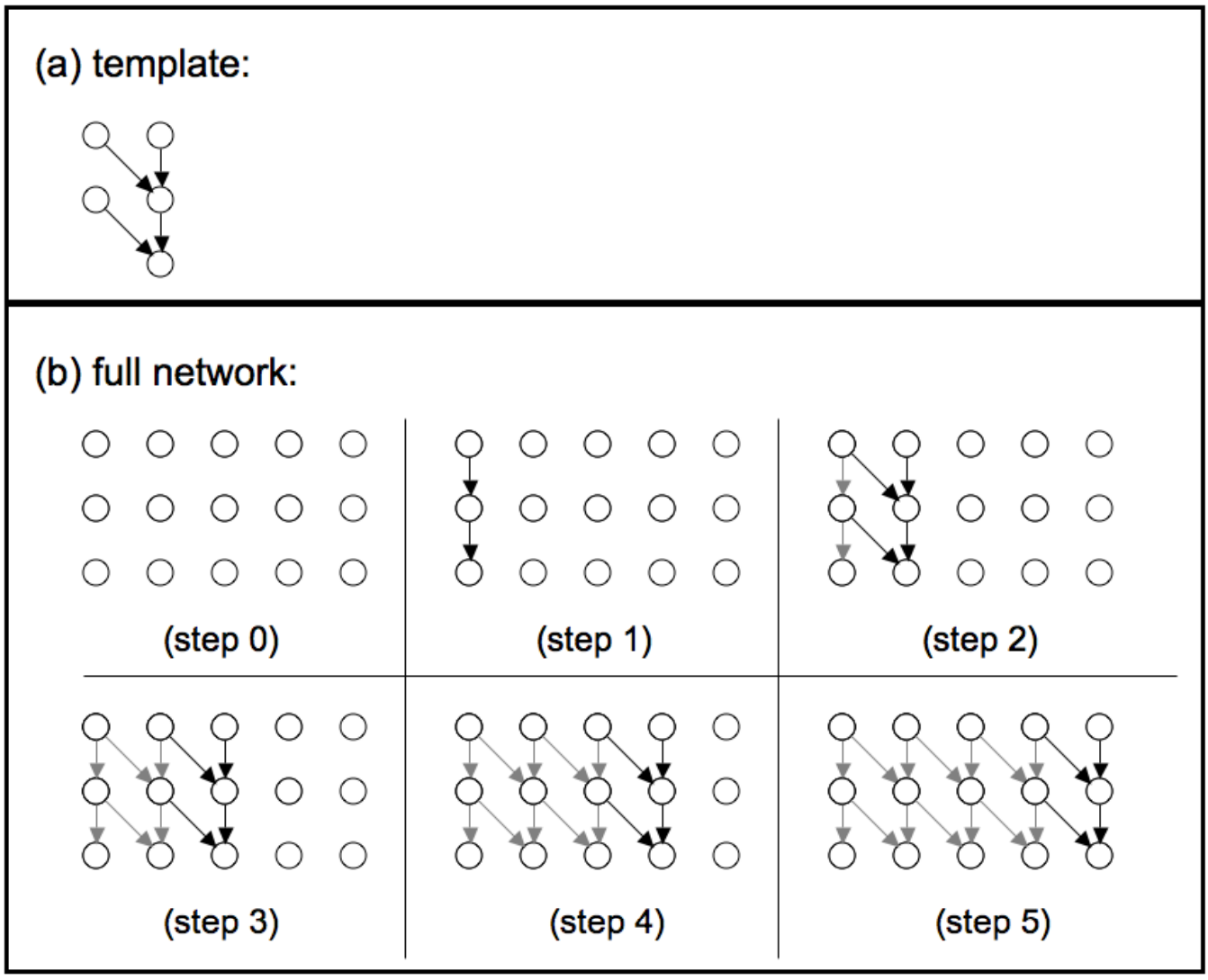
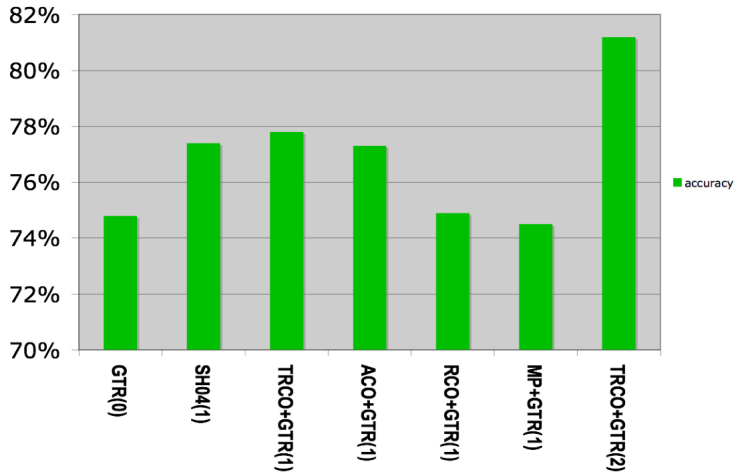


FIG S4—*Template instantiation. (a) A template. (b) A network during successive stages of template instantiation. Initially the network has no edges. The template is instantiated in each column, resulting in edges being added to the full network (black edges are those most recently added; gray edges were added in a previous step). Edges of the template for which both endpoints are not present in the full network are ignored (e.g., the leftmost column in this example).*

a)



b)

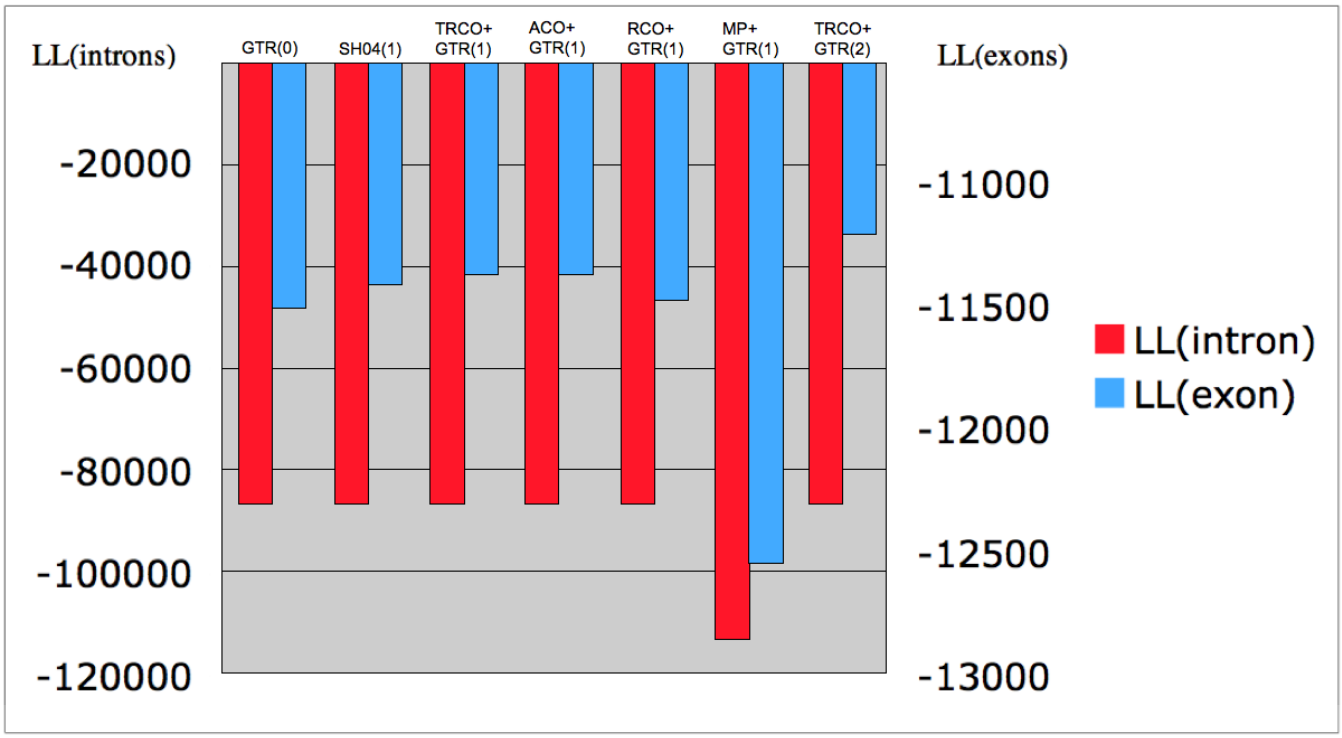
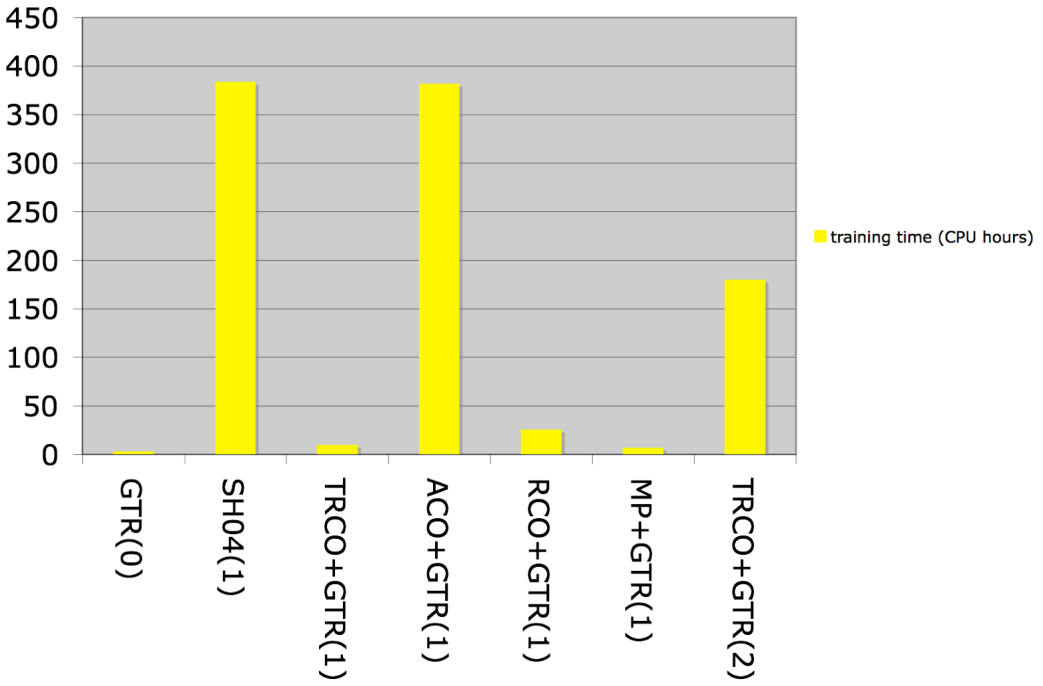


FIG S5—*Accuracy and model fit. (a) mean classification accuracy (percentage of test cases correctly classified) on held-out data. (b) mean log-likelihood of training exons (blue) and training introns (red). All values were averaged via 5-fold cross-validation.*

a)



b)

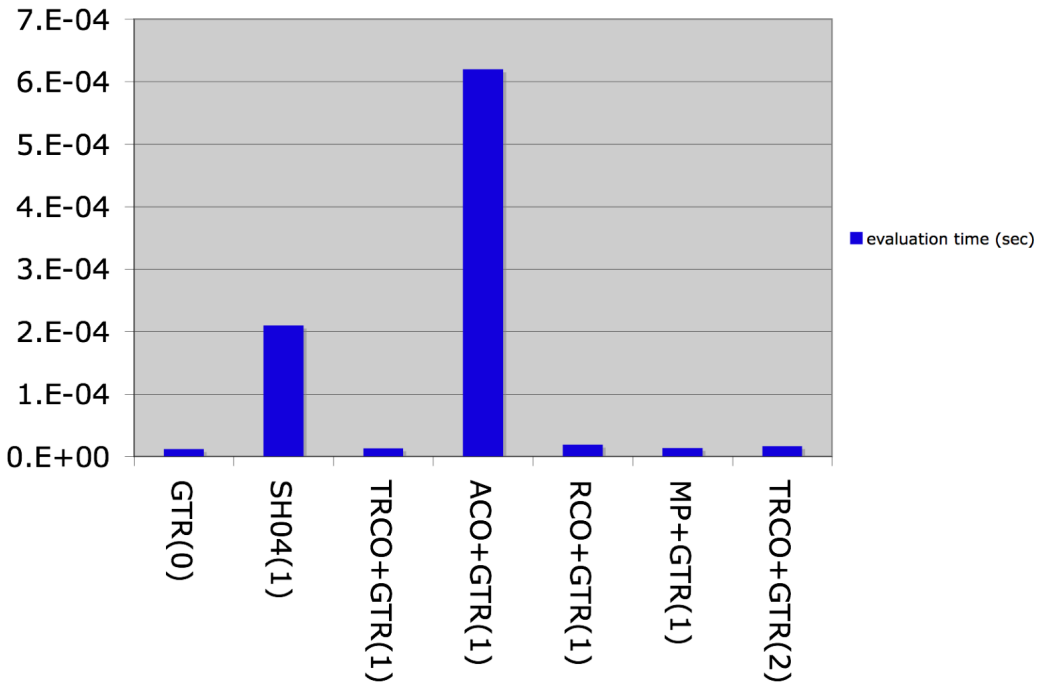


FIG S6—*Training and evaluation times. (a) mean training time (hours × number of CPUs). (b) mean time to evaluate a single alignment column (seconds). All values were averaged via 5-fold cross-validation.*

S1. Supplementary Methods

Parameters were estimated via quasi-Newton methods, using the BFGS algorithm as implemented in the Gnu Scientific Library (GSL) version 1.5. Meta-parameters for the optimizer were set identically for all model classes: *tolerance*=0.01, *step-size*=0.001, *gradient-threshold*=10, $dx=1\times10^{-6}$. *Tolerance* and *step-size* are parameters to the GSL routine. The *dx* meta-parameter was used to estimate partial derivatives via a symmetric, 2-point formula:

$$f'(x) \approx (f(x+dx)-f(x-dx))/(2dx).$$

The optimization terminated either when the norm of the gradient vector fell below the *gradient-threshold* parameter, or when the GSL software reported a "no progress" condition. Preliminary runs during early development indicated that a lower *gradient-threshold* value significantly slowed convergence without improving resulting accuracy.

Training of higher-order models was accelerated by seeding the model parameters with the optimized values of a lower-order model rather than random initial values. Thus, training of an n^{th} order model proceeded by first training all models of orders 0 to $n-1$, and then initializing the parameters of the n^{th} order model with the values of corresponding parameters in the $(n-1)^{th}$ order model (of the same type) prior to invoking the optimizer. It should be noted that our conditional models are more directly amenable to such seeding than some of the popular *n-mer*-based models for which not every parameter in the n^{th} order model has an obvious counterpart in the $(n-1)^{th}$ order model to use for the seeding (such as in GB05 and SH04). This is yet another advantage of our conditional models.

S2. Supplementary Results on Optimization

Previous context-dependent models have been optimized by iterative local optimization schemes, namely gradient-descent and/or expectation-maximization (EM) methods. However, neither of the two most widely-known EM formulations for context-dependent substitution models are readily applicable to the full range of model types considered in this work. The procedure given by (Siepel and Haussler, 2004) explicitly assumes an *n-mer*-based formulation, and does not provide a closed-form formula for the M-step, relying instead on a heuristic based on BFGS for the M-step of EM. The procedure given in the supplementary files of (Klosterman *et al*., 2006) does provide a closed-form M-step, but assumes an unrestricted matrix (i.e., it directly estimates the elements of the rate matrix **Q** rather than estimating parameters which can be used in combination with equilibrium frequencies to produce **Q**) and does not simultaneously estimate the branch lengths of the phylogeny.

While either of these could conceivably be adapted to train a particular non-*n-mer*-based model, the use of a general-purpose training procedure was of greater utility for our Bayesian-network-based modeling framework, in which potential users may wish to explore a number of different dependency networks in the modeling of their particular data. Although an additional set of experiments utilizing EM-based training for all model types would provide a more complete picture of the propensities of the models considered in the manuscript, the derivation and implementation of EM update equations for all possible model types is beyond the scope of this study.

Nevertheless, motivated by the markedly different performance of the SH04 model on the two different datasets (coding/noncoding and regulatory), we performed comparisons of our de novo implementation of the SH04 model, with SH04 models re-trained using both the EM-based parameter estimation program and the BFGS-based optimizer as implemented in *phyloFit,* developed by A. Siepel and described in (Siepel and Haussler, 2004b). These validation efforts are described below.

S2.1 Comparison of BFGS-based Trainers

In order to validate the accuracy of our BFGS-based training software, which had delivered a markedly reduced performance of SH04 compared to TRCO on the regulatory region data, we replicated experiments using the BFGS option of the phyloFit program. We trained 1^{st}-order SH04 models (i.e., using 1bp of context in addition to the current column) for each cross-validation partition of the regulatory element data set, and the model parameters emitted by phyloFit were read directly into our program so that our classifier could be applied to the phyloFit-

trained SH04 models. Unlike our BFGS trainer, this program provides no user-definable parameters such as line-minimization tolerance or gradient threshold. In comparing 1st-order SH04 models trained by the two programs, we found the cross-validation accuracy on the regulatory-element data set to heavily favor our implementation, by 21.4% (53.5±5.6% for phyloFit, versus 74.9±4.5% for our trainer). This difference appeared to be due largely to an inability of phyloFit to achieve a close fit on the negative training data (ancestral repeats); the mean log-likelihood on the positive set (true regulatory elements) was somewhat better for phyloFit than for our trainer (−57622 vs. −58424) but was substantially worse than our trainer on the negative training set (−56808 vs. −42614). In summary, while SH04 models trained by BFGS performed less well than TRCO, this is not due to any bias in our implementation.

S2.2 Comparison with the EM-based Trainer

In light of the poor performance of both BFGS-optimized SH04 models, we next applied the phyloFit program with the "−EM" option. On the regulatory-element data set, the EM-trained SH04 model (denoted "SH04/EM") out-performed our TRCO model by 3.5% (78.4±7.1% vs. 74.9±4.5%; both models were of 1st-order). This suggests that the use of EM for the training of context-dependent substitution models can provide not only a speed advantage as reported by Siepel and Haussler, but also an advantage in terms of model fit: the mean log-likelihoods of the positive (respectively, negative) regulatory-element training sets for SH04/EM were −57690 (−41799), as compared to −58837 (−43663) for SH04 trained using our BFGS-based optimizer, resulting in a 10% difference in classification accuracy between the EM-trained and the best BFGS-trained versions of SH04 (78.4±7.1% vs. 68.6±5.8%). To our knowledge, this propensity of EM to produce a better fit for context-dependent substitution models than quasi-Newton methods has not been previously recognized.

Note that the advantage of the EM-trained SH04 model over TRCO shrank from 3.5% to only 1% when the substitution models were combined with a Markov chain (also of 1st-order) to score the target sequence (human in this case)—i.e., 87.3±6.6% for SH04/EM vs. 86.3±6.1% for TRCO/BFGS. (For sequence annotation tasks, a Markov chain or similar sequence model would be combined with the substitution model in most practical cases). Given the substantial accuracy gains (10%) experienced by SH04 when switching from a BFGS-based trainer to an EM-based one, this 1% difference between SH04/EM and TRCO/BFGS may very well be due entirely to the difference in training regimes. For this reason, we remain confident in the comparisons given in the main manuscript between the BFGS-trained SH04 and TRCO models, which show TRCO fully matching the accuracy of SH04 when both models are trained using the same trainer.

While these results might strongly argue in favor of EM-based optimization, the picture changed when evaluating a different data set: Although the ability of EM to produce a better fit to the training data was again seen on the ENCODE exon/intron data set, this resulted in a lower cross-validation classification accuracy than TRCO/BFGS, possibly due to over-training of SH04: In the case of 1st-order models, SH04/EM achieved a 6.2% lower cross-validation accuracy than TRCO/SH04 on the ENCODE data (71.2±3.2% vs. 77.4±2.4%), despite achieving a better fit to the training data: −11325 versus −11392 for the training exons, and −84345 versus −85550 for the training introns. (For comparison, the best first-order model on this task was TRCO, with 77.8% accuracy.)

S2.3 Summary

Comparisons between our BFGS-based trainer and the EM/BFGS-based trainer *phyloFit* failed to uncover any evidence that our software implementation of the SH04 model (or our implementation of the BFGS-based optimizer) is biased against SH04. Although EM typically produced a better fit to the training data, this sometimes resulted in possible overtraining and poorer cross-validation accuracy. Because a major strength of our context-modeling framework is its flexibility in the modeling of particular dependency patterns in DNA, a general-purpose training procedure capable of accommodating arbitrary dependency networks without the need to derive and implement custom EM update equations has the greatest utility for our framework. All of these points support the validity of our use of BFGS in the comparisons given in the main manuscript. Exploration of other training regimes for context-dependent substitution models would be a worthwhile effort, but is beyond the scope of the present work.

S3. Supplementary Columns for Table 4

Standard deviations of accuracy measures from Table 4 in the main manuscript are given below.

model	*n*	*acc*	*MC*	*acc+MC*
GTR	0	3.2	4.4	5.7
TRCO+GTR	1	4.5	4.1	6.1
TRCO+GTR	2	4.4	2.4	4.9
TRCO+GTR	3	5.4	3.5	4.4
SH04	1	5.8	4.1	4.2

S4. References

Klosterman PS, Ulzilov AV, Bendana YR, Bradley RK, Shao S, Kosiol C, Goldman N, and Holmes I (2006) XRate: a fast prototyping, training and annotation tool for phylo-grammars. *BMC Bioinformatics* **7**:428.

Siepel A and Haussler D (2004) Phylogenetic estimation of context-dependent substitution rates by maximum likelihood. *Mol. Biol. Evol.*, **21**, 468-488.